Sun Java System Messaging Server 6.3 관리 설명서

Sun Microsystems, Inc.
4150 Network Circle
Santa Clara, CA 95054
U.S.A.

부품 번호: 820–0512
2007년 6월 8일

070711@17466

목차

그림

표

코드 예

머리말

이 설명서에서는 Sun Java™ System Messaging Server 및 관련 소프트웨어 구성 요소를 관리하는 방법에 대해 설명합니다. Messaging Server는 개방형 인터넷 표준을 사용하여 모든 규모의 기업과 메시징 호스트의 전자 메일 요구 사항을 충족시키는 강력하고도 유연한 크로스 플랫폼 솔루션을 제공합니다.

이 문서의 개정 기록은 46 페이지 "Sun Java System Messaging Server 6.3 관리 설명서 개정 기록"을 참조하십시오.

대상

이 설명서는 사이트에서 Messaging Server를 관리하고 배포하는 일을 담당하는 사람을 대상으로 합니다. **Sun Java Communications Suite 5 Deployment Planning Guide**도 함께 참조해야 합니다.

본 설명서를 읽기 전에

본 설명서에서는 설명서를 읽는 대상이 Messaging Server 소프트웨어의 관리를 담당하며 다음 내용에 대해 잘 알고 있다고 가정합니다.

- 인터넷 및 WWW(World Wide Web)
- Messaging Server 프로토콜
- Sun Java System Directory Server 및 LDAP
- 시스템 관리 및 네트워킹 작업
- 일반 배포 구조

본 설명서의 구성

이 설명서는 다음 장과 부록으로 구성되어 있습니다.

표 P–1 본 설명서의 구성

장	설명
머리말	이 설명서 사용과 관련된 일반적인 내용입니다.
1 장	Messaging Server 기능을 설정하는 데 필요한 작업에 대해 설명합니다.
2 장	Messaging Server 5.2에서 이 버전의 Messaging Server로 업그레이드하는 방법에 대해 설명합니다.
3 장	Messaging Server와 함께 사용하기 위해 Veritas Cluster Server 및 Sun Cluster 고가용성 클러스터링 소프트웨어를 구성하는 방법에 대해 설명합니다.
4 장	일반적인 Messaging Server 작업에 대해 설명합니다.
5 장	POP, IMAP 및 HTTP 서비스를 지원하도록 서버를 구성하는 방법에 대해 설명합니다.
6 장	단일 사인 온(SSO)을 사용하는 방법에 대해 설명합니다.
7 장	표준 메일 프로토콜(POP, IMAP 및 SMTP)용 MMP(Messaging Multiplexor)에 대해 설명합니다.
8 장	MTA의 개념에 대해 설명합니다.
9 장	주소 변환 및 라우팅에 대해 설명합니다.
10 장	MTA 서비스와 구성에 대해 설명합니다.
11 장	imta.cnf 파일에서 다시 쓰기 규칙을 구성하는 방법에 대해 설명합니다.
12 장	MTA 구성 파일 imta.cnf에서 채널 키워드 정의를 사용하는 방법에 대해 설명합니다.
13 장	MTA에서 미리 정의된 채널 정의를 사용하는 방법에 대해 설명합니다.
14 장	스팸 및 바이러스 필터링 소프트웨어를 Messaging Server와 통합하고 구성하는 방법에 대해 설명합니다.
15 장	SMTP 대화 중에 위조된 전자 메일을 감지하여 거부할 수 있는 기술에 대해 설명합니다.
16 장	LMTP 작업 및 배포에 대해 설명합니다.
17 장	휴가 자동 회신 기법에 대해 설명합니다.
18 장	메일의 소스(보낸 사람, IP 주소 등) 또는 헤더 문자열에 기초하여 메일을 필터링하는 방법에 대해 설명합니다.
19 장	`conn_throttle.so`를 대체하면서 비슷한 기능을 제공하지만 Messaging Server 설치 전체로 확장하는 저장소 프로세스에 대해 설명합니다.
20 장	메시지 저장소와 해당 관리 인터페이스에 대해 설명합니다.

표 P–1 본 설명서의 구성 *(계속)*

장	설명
21 장	Messaging Server의 아카이브 개념에 대해 설명합니다.
22 장	Message Queue 서비스에서 클라이언트가 소비할 메시지를 생성하도록 JMQ 알림 플러그 인을 구성하는 방법에 대해 설명합니다.
23 장	Messaging Server에 대한 보안 및 액세스 제어를 구성하는 방법에 대해 설명합니다.
24 장	S/MIME을 관리하는 방법에 대해 설명합니다.
25 장	Messaging Server 로깅 기능에 대해 설명합니다.
26 장	MTA 문제 해결을 위한 일반 도구, 방법 및 절차에 대해 설명합니다.
27 장	Messaging Server의 모니터링에 대해 설명합니다.
부록 A	Messaging Server에서 SNMP 지원을 사용하는 방법에 대해 설명합니다.
부록 B	Messaging Server에서 ENS Publisher(Event Notification Service Publisher)를 사용하고 ENS(Event Notification Service)를 관리하는 방법에 대해 설명합니다.
부록 C	SMS(Short Message Service)를 구현하는 방법에 대해 설명합니다.
부록 D	설치를 계획할 수 있는 워크시트를 제공합니다.

Messaging Server 설명서 세트

다음 표에는 Messaging Server의 중요 설명서 세트에 포함된 설명서가 요약되어 있습니다.

표 P–2 Messaging Server 설명서

설명서 제목	내용
Sun Java System Messaging Server 6.3 Administration Reference	Messaging Server 명령, `configutil` 매개 변수, 구성 파일 및 옵션, 지원되는 표준에 대한 자세한 정보를 제공합니다.
Sun Java Communications Suite 5 Deployment Planning Guide	Messaging Server를 포함하여 Sun Java System Communications Services를 배포하는 데 필요한 정보가 들어 있습니다.
Sun Java System Delegated Administrator 6.4 관리 설명서	Sun Java System Communications Services Delegated Administrator를 구성 및 관리하는 방법에 대해 설명합니다. Delegated Administrator 명령에 대해서도 설명합니다.

표 P–2 Messaging Server 설명서 *(계속)*

설명서 제목	내용
Sun Java Communications Suite 5 Schema Migration Guide	System Messaging Server 및 Calendar Server에 대해 Sun Java System LDAP 디렉토리 데이터를 LDAP Schema 1에서 LDAP Schema 2로 마이그레이션하는 방법에 대해 설명합니다.
Sun Java Communications Suite 5 Event Notification Service Guide	Messaging Server 및 Calendar Server에 대한 ENS(Event Notification Service) 구조와 API를 설명합니다. 서버 설치를 사용자 정의하는 데 사용할 수 있는 ENS API에 대해 자세히 설명합니다.
Sun Java Communications Suite 5 릴리스 노트	Sun Java System Messaging Server를 릴리스할 당시 사용 가능한 중요한 정보가 포함되어 있습니다. 새 기능과 향상된 기능, 알려진 문제점과 제한 사항 및 기타 정보에 대해서도 설명합니다.
Sun Java Communications Suite 5 Schema Reference	Messaging Server 및 Calendar Server에 대한 스키마 정보의 참조로 사용됩니다.
Sun Java System Communications Express 6.3 관리 설명서	Communications Express 및 관련 소프트웨어 구성 요소를 관리하는 방법에 대해 설명합니다.
Sun Java System Communications Express 6.3 Customization Guide	Communications Express의 모양과 색감을 사용자 정의하는 방법에 대해 설명합니다. 자주 요청되는 사용자 정의를 수행하는 방법을 중점적으로 설명합니다.
Sun Java Enterprise System 5 Installation Guide for UNIX	Sun Java Enterprise System(Java ES) 소프트웨어를 설치하는 데 필요한 정보가 들어 있습니다.
Sun Java System Messaging Server 6 2005Q4 MTA Developer's Reference	Messaging Server MTA(Message Transfer Agent) SDK(Software Development Kit) 및 Callable Send 기능에 대해 설명합니다.
Sun Java Enterprise System Glossary	용어집입니다.
Sun Java Communications Suite 5 Documentation Center	Communications Suite 설명서에 대한 주제별 링크입니다.

또한 다음 URL을 사용하여 모든 Communications Services 제품에 해당하는 설명서를 확인할 수 있습니다. (http://www.sun.com/bigadmin/hubs/comms/)

관련 설명서

http://docs.sun.com 웹 사이트에서 Sun 기술 관련 설명서를 온라인으로 이용할 수 있습니다. 아카이브를 탐색하거나 특정 책 제목 또는 주제를 검색할 수 있습니다.

Messaging Server 배포와 관련된 기타 서버 설명서를 보려면 다음 위치로 이동하십시오.

- Access Manager 설명서: http://docs.sun.com/app/docs/coll/1292.2
- Calendar Server 설명서: http://docs.sun.com/app/docs/coll/1313.2
- Communications Express 설명서: http://docs.sun.com/app/docs/coll/1312.2
- Directory Server 설명서: http://docs.sun.com/app/docs/coll/1316.2
- Instant Messaging 설명서: http://docs.sun.com/app/docs/coll/1309.2
- Messaging Server 설명서: http://docs.sun.com/app/docs/coll/1312.2

기본 경로 및 파일 이름

다음 표에서는 본 설명서에 사용된 기본 경로와 파일 이름에 대해 설명합니다.

표 P–3 기본 경로 및 파일 이름

자리 표시자	설명	기본값
msg-svr-base	Messaging Server의 기본 설치 디렉토리를 나타냅니다. Messaging Server 기본 설치 디렉토리 및 제품 디렉토리는 특정 플랫폼에 따라 달라집니다.	Solaris 시스템: `/opt/SUNWmsgsr` Linux 시스템: `/opt/sun/messaging`

활자체 규약

다음 표에서는 본 설명서에 사용된 활자체 규약에 대해 설명합니다.

표 P–4 활자체 규약

서체	의미	예
`AaBbCc123`	명령, 파일 및 디렉토리의 이름, 그리고 컴퓨터 화면에 출력되는 내용입니다.	`.login` 파일을 편집하십시오. `ls -a`를 사용하여 모든 파일을 나열하십시오. `machine_name% you have mail.`
`AaBbCc123`	컴퓨터 화면 상의 출력과는 달리 사용자가 직접 입력하는 사항입니다.	`machine_name%` **`su`** `Password:`

표 P–4 활자체 규약 *(계속)*

서체	의미	예
AaBbCc123	명령줄 자리 표시자: 실제 이름이나 값으로 대체됩니다.	파일을 삭제하려면 `rm` *filename*을 입력하십시오.
AaBbCc123	책 제목, 새로 나오는 용어, 강조 표시할 단어입니다. (일부 강조된 항목은 온라인상에서 볼드로 표시됩니다.)	**사용자 설명서**의 6장을 읽으십시오. *cache*는 로컬로 저장된 복사본입니다. 파일을 저장하지 **마십시오**.

명령 예의 쉘 프롬프트

기본 시스템 프롬프트 및 수퍼유저 프롬프트는 다음 표와 같습니다.

표 P–5 쉘 프롬프트

쉘	프롬프트
UNIX 및 Linux 시스템의 C 쉘	`machine_name%`
UNIX 및 Linux 시스템의 C 쉘 수퍼유저	`machine_name#`
UNIX 및 Linux 시스템의 Bourne 쉘 및 Korn 쉘	`$`
UNIX 및 Linux 시스템의 Bourne 쉘 및 Korn 쉘 수퍼유저	`#`

기호 규칙

다음 표에서는 본 설명서에 사용된 기호에 대해 설명합니다.

표 P–6 기호 규칙

기호	설명	예	의미
`[ ]`	선택적 인수와 명령 옵션을 포함합니다.	`ls [-l]`	`-l` 옵션은 사용하지 않아도 됩니다.
`{ | }`	필수 명령 옵션에 대한 일련의 선택 항목을 포함합니다.	`-d {y|n}`	`-d` 옵션에는 `y` 인수나 `n` 인수를 사용해야 합니다.
`${ }`	변수 참조를 나타냅니다.	`${com.sun.javaRoot}`	`com.sun.javaRoot` 변수의 값을 참조합니다.
-	동시에 입력하는 여러 키를 결합합니다.	Ctrl-A	Ctrl 키를 누른 채로 A 키를 누릅니다.

표 P–6 기호 규칙 (계속)

기호	설명	예	의미
+	연속해서 입력하는 여러 키를 결합합니다.	Ctrl+A+N	Ctrl 키를 눌렀다가 놓은 다음 후속 키들을 누릅니다.
→	그래픽 사용자 인터페이스의 메뉴 항목 선택을 나타냅니다.	파일 → 새로 만들기 → 템플리트	파일 메뉴에서 새로 만들기를 선택합니다. 새로 만들기 하위 메뉴에서 템플리트를 선택합니다.

Sun 자원 온라인 액세스

docs.sun.com 웹 사이트에서 Sun 기술 관련 설명서를 온라인으로 이용할 수 있습니다. docs.sun.com 아카이브를 탐색하거나 특정 설명서 제목 또는 주제를 검색할 수 있습니다. 설명서는 PDF 및 HTML 형식의 온라인 파일로 이용할 수 있습니다. 장애가 있는 사용자도 지원 기술을 통해 두 가지 형식을 모두 읽을 수 있습니다.

다음과 같은 Sun 자원을 이용하려면 http://www.sun.com으로 이동하십시오.

- Sun 제품 다운로드
- 서비스 및 솔루션
- 지원(패치 및 업데이트 포함)
- 교육
- 리서치
- 커뮤니티(예: Sun 개발자 네트워크)

타사 웹 사이트 참조 사항

이 설명서에서는 추가 관련 정보를 제공하기 위해 타사 URL을 참조하기도 합니다.

주 – Sun은 이 설명서에 언급된 타사 웹 사이트의 가용성에 대해 책임지지 않습니다. Sun은 이러한 사이트나 자원을 통해 사용할 수 있는 내용, 광고, 제품 또는 기타 자료에 대해서는 보증하지 않으며 책임지지 않습니다. Sun은 해당 사이트 또는 자원을 통해 사용 가능한 내용, 제품 또는 서비스의 사용과 관련해 발생하거나 발생했다고 간주되는 손해나 손실에 대해 책임이나 의무를 지지 않습니다.

사용자 의견 환경

Sun은 설명서의 내용을 지속적으로 개선하고자 하며 사용자 여러분의 의견과 제안을 환영합니다. 의견을 공유하려면 http://docs.sun.com으로 이동하여 의견 보내기를 누르고 표시되는 온라인 양식에 전체 설명서 제목과 부품 번호를 입력합니다. 부품 번호는 설명서의 제목 페이지나 문서 URL에 있는 7자리 또는 9자리 숫자입니다.

Sun Java System Messaging Server 6.3 관리 설명서 개정 기록

버전	날짜	변경 설명
12	2007년 6월 8일	■ **Sun Java Communications Suite 5 릴리스 노트**에 **Sun Java Communications Suite 5 릴리스 노트**의 "MTA의 새로운 기능"을 추가 및 설명했습니다. ■ 버그 수정 및 기타 설명서 수정 사항을 추가했습니다.
11	2007년 4월 14일	■ 19 장 예를 추가했습니다. ■ Veritas Server Cluster 지원 버전을 3.5, 4.0, 4.1 및 5.0으로 업데이트했습니다. ■ 새 390 페이지 "12.12.5 스팸 필터 키워드"를 문서화했습니다. ■ 버그 수정 및 기타 설명서 수정 사항을 추가했습니다.
10	2007년 3월	이 기술 노트의 초기 릴리스입니다.

◆ ◆ ◆ 1 장

1 사후 설치 작업 및 레이아웃

이 장의 내용은 **Sun Java Communications Suite 5 Deployment Planning Guide**를 읽고 Sun Java™ Enterprise System 설치 프로그램을 사용하여 Messaging Server를 설치한 사용자를 대상으로 합니다. **Sun Java Communications Suite 5 Installation Guide**를 참조하십시오. 다음 작업을 수행하여 Messaging Server 기능을 설정할 수 있습니다. 사용자와 그룹을 준비 및/또는 마이그레이션하고 배포를 사용자 정의할 수 있습니다. 사용자 정의는 이 설명서의 뒷 장에 설명되어 있습니다. 준비는 **Sun Java System Delegated Administrator 6.4 관리 설명서**에 설명되어 있습니다.

이 장은 다음 내용으로 구성되어 있습니다.

1.1 UNIX 시스템 사용자와 그룹 만들기

시스템 사용자는 특정 서버 프로세스를 실행합니다. 사용자가 실행할 프로세스에 대해 적절한 권한을 갖도록 해당 사용자에게 권한을 부여해야 합니다.

모든 Sun Java System 서버에 대한 시스템 사용자 계정 및 그룹을 설정하고 해당 사용자가 소유한 디렉토리 및 파일에 대한 권한을 설정합니다. 그렇게 하려면 다음 단계를 수행합니다.

주 – 보안 상의 이유로 일부 배포에서는 서버마다 다른 시스템 관리자를 지정하는 것이 좋습니다. 이 작업은 서버마다 다른 시스템 사용자 및 그룹을 만들어 수행합니다. 예를 들어, Messaging Server의 시스템 사용자는 Web Server의 시스템 사용자와 다르고 Messaging Server 시스템 관리자는 Web Server를 관리할 수 없습니다.

▼ UNIX 시스템 사용자와 그룹 만들기

1. **수퍼유저로 로그인합니다.**
2. **시스템 사용자가 속하게 될 그룹을 만듭니다.**

   다음 예에서는 `mail` 그룹을 만듭니다.

   ```
   # groupadd mail
   ```

3. **시스템 사용자를 만든 다음 방금 전에 만든 그룹과 연결합니다. 또한 해당 사용자의 비밀번호를 설정합니다.**

   다음 예에서는 `mailsrv` 사용자를 만들고 이 사용자를 `mail` 그룹과 연결합니다.

   ```
   # useradd -g mail mailsrv
   ```

   `useradd` 및 `usermod` 명령은 `/usr/sbin`에 있습니다. 자세한 내용은 UNIX 설명서 페이지를 참조하십시오.

4. **또한 `/etc/groups` 및 `/etc/passwd` 파일을 확인하여 앞서 만든 시스템 그룹에 사용자가 추가되었는지 확인합니다.**

주 – Messaging Server를 설치하기 전에 UNIX 시스템 사용자와 그룹을 설정하지 않기로 한 경우에는 49 페이지 "1.3 Messaging Server 초기 런타임 구성 만들기"를 수행할 때 UNIX 시스템 사용자와 그룹을 지정할 수 있습니다.

1.2 Messaging Server 구성을 위해 Directory Server 준비

디렉토리 준비 및 디렉토리 준비 스크립트 comm_dssetup에 대한 자세한 내용은 설치 설명서에서 **Sun Java Communications Suite 5 Installation Guide**의 8 장, "Directory Preparation Tool (comm_dssetup.pl)"을 참조하십시오. 이 장에서는 LDAP Directory Server가 Messaging Server, Calendar Server 또는 Delegated Admin CLI 유틸리티 구성과 함께 작동하도록 구성하는 Directory Server 설정 스크립트(`comm_dssetup.pl`)를 실행하는 방법에 대해 설명합니다. `comm_dssetup.pl` 스크립트는 Directory Server에 새 스키마, 색인 및 데이터를 설정하여 Directory Server를 준비합니다. 새 Messaging Server 및 Communications Express 설치를 위해서는 이 스크립트를 실행해야 합니다. Directory Server에 속한 구성 요소 제품 중 하나를 업그레이드하는 경우에는 최신 `comm_dssetup.pl`을 실행하는 것도 좋습니다.

1.3 Messaging Server 초기 런타임 구성 만들기

초기 런타임 구성 프로그램은 Messaging Server를 설정하고 실행하는 데 필요한 구성을 제공합니다. 즉, 일반 기능의 Messaging Server 구성을 설정하도록 **초기 런타임 구성**을 만듭니다. 그렇게 하면 특정 사용자 정의를 만들 수 있는 기본 작업 구성이 제공됩니다. 이 프로그램은 한 번만 실행하면 됩니다. 이후에 이 프로그램을 또 실행하면 기존 구성을 덮어씁니다. 초기 런타임 구성을 수정하려면 이 설명서와 **Sun Java System Messaging Server 6.3 Administration Reference**에 설명된 구성 유틸리티를 사용하십시오.

1.3.1 Messaging Server 전제 조건

초기 런타임 구성 프로그램을 실행하기 전에 다음을 수행해야 합니다.

- Directory Server를 설치 및 구성합니다. **Sun Java Enterprise System 5 Installation Guide for UNIX**를 참조하십시오.
- `comm_dssetup.pl` 프로그램을 실행합니다. **Sun Java Communications Suite 5 Installation Guide**의 "Messaging Server Postinstallation Configuration"을 참조하십시오.
- 부록 D에 제공된 확인 목록에 Administration Server 및 Directory Server 설치 및 구성 매개 변수를 기록합니다.

1.3.2 Messaging Server 구성 확인 목록

Messaging Server 초기 런타임 구성 프로그램을 실행하는 경우 매개 변수를 표 D–3에 기록하십시오. 특정 질문에 답하려면 부록 D의 Directory Server 설치 확인 목록을 참조하십시오.

▼ configure 프로그램 실행 방법

다음은 Messaging Server 초기 런타임 구성을 구성하는 단계입니다.

1 **DNS가 제대로 구성되어 있는지 확인하고 로컬 서브넷에 없는 호스트로 라우팅하는 방법이 명확하게 지정되어 있는지 확인하십시오.**

- `/etc/defaultrouter`에 게이트웨이 시스템의 IP 주소가 포함되어 있어야 합니다. 이 주소는 로컬 서브넷에 있어야 합니다.
- `/etc/resolv.conf`가 존재하며 도달 가능한 DNS 서버와 도메인 접미어에 대한 올바른 항목이 포함되어 있습니다.
- `/etc/nsswitch.conf`에서 `hosts:` 및 `ipnodes:` 줄에 `files`, `dns` 및 `nis` 키워드가 추가되어 있습니다. 키워드 `files`는 `dns` 및 `nis` 앞에 있어야 합니다. 줄이 다음과 같은 경우

```
hosts:  nis dns files
ipnodes:  nis dns files
```

다음과 같이 변경해야 합니다.

```
hosts:  files nis dns
ipnodes:  files nis dns
```

- `/etc/hosts` 파일의 첫 번째 호스트 이름은 FQDN이어야 합니다.

  `/etc/hosts` 파일의 인터넷 호스트 테이블이 다음과 같은 경우

  ```
  123.456.78.910 budgie.west.sesta.com
  123.456.78.910 budgie loghost mailhost
  ```

  호스트의 IP 주소가 한 줄이 되도록 변경하십시오. 첫 번째 호스트 이름은 정규화된 도메인 이름이어야 합니다. 예를 들면 다음과 같습니다.

  ```
  123.456.78.910 budgie.west.sesta.com budgie loghost mailhost
  ```

- 다음 명령을 실행하여 줄이 올바르게 표시되는지 확인할 수 있습니다.

  ```
  # getent hosts ip_address
  # getent ipnodes ip_address
  ```

  줄이 올바르게 표시되면 IP 주소 뒤에 FQDN이 표시되고 그 뒤에 다른 값이 표시되어야 합니다. 예를 들면 다음과 같습니다.

  ```
  # getent hosts 192.18.126.103
  192.18.126.103  budgie.west.sesta.com budgie loghost mailhost
  ```

2 다음 명령으로 Messaging Server 초기 런타임 구성을 호출합니다.

msg-svr-base`/sbin/configure` *[flag]*

원격 시스템에서 Messaging Server를 구성하는 경우 `xhost`(1) 명령을 사용할 수 있습니다.

아래 표에서는 `configure` 프로그램에 설정할 수 있는 선택적 플래그에 대해 설명합니다.

플래그	설명
`-nodisplay`	명령줄 구성 프로그램을 호출합니다.
`-noconsole`	GUI 사용자 인터페이스 프로그램을 호출합니다.
`-state` *[statefile]*	자동 설치 파일을 사용합니다. `-nodisplay` 및 `-noconsole` 플래그와 함께 사용해야 합니다. 54 페이지 "자동 설치 수행"을 참조하십시오.

`configure` 명령을 실행하면 구성 프로그램이 시작됩니다.

3 시작합니다.

구성 프로그램의 첫 번째 패널은 저작권 페이지입니다. 다음을 선택하여 계속하거나 취소를 눌러 종료합니다. 경고를 받도록 관리 서버(Messaging Server 2005Q4 이전 버전에만 해당)를 구성하지 않은 경우에는 확인을 눌러 계속합니다.

4 정규화된 호스트 이름(FQHN)을 입력합니다.

이 시스템에서 Messaging Server가 작동합니다. Java Enterprise System 설치 프로그램을 사용하여 서버를 설치한 경우에는 실제 호스트 이름을 지정했을 수 있습니다. 그러나 클러스터 환경을 설치하는 중이면 논리 호스트 이름을 사용할 수 있습니다. 여기에서 원래 지정한 이름을 변경할 수 있습니다.

5 구성 및 데이터 파일을 저장할 디렉토리를 선택합니다.

Messaging Server 구성 및 데이터 파일을 저장할 디렉토리 *msg-svr-base*에 없는 경로 이름을 지정하십시오. 심볼릭 링크가 구성 및 데이터 디렉토리의 *msg-svr-base*에 생성됩니다. 이러한 심볼릭 링크에 대한 자세한 내용은 62 페이지 “1.11 사후 설치 디렉토리 레이아웃”을 참조하십시오.

이러한 파일을 저장할 충분한 디스크 공간이 있는지 확인합니다.

6 구성 요소를 로드하고 있음을 나타내는 작은 창이 표시됩니다.

몇 분 정도 소요됩니다

7 구성할 구성 요소를 선택합니다.

구성할 메시징 구성 요소를 선택합니다.

- Message Transfer Agent: 라우팅을 처리하고, 사용자 메일을 전송하며, SMTP 인증을 처리합니다. MTA는 호스트된 도메인, 도메인 별칭 및 서버측 필터에 대한 지원을 제공합니다.
- 메시지 저장소: 범용 메시지 저장소를 통해 통합된 메시징 서비스를 위한 기반을 제공합니다. 여러 프로토콜(HTTP, POP, IMAP)을 통해 메시지 저장소에 액세스할 수 있습니다. 메시지 저장소만 구성하는 경우에는 MTA도 선택해야 합니다.
- Webmail Server: HTTP 프로토콜이 메시지 저장소의 메시지를 검색하는 작업을 처리합니다. 이 구성 요소는 Communication Express에서 웹 기반 액세스를 제공할 때에도 사용됩니다.
- Messaging Multiplexor: 조직 내의 여러 메시징 서버 시스템에 대한 프록시 역할을 합니다. 사용자는 각 연결을 적절한 메일 서버로 리디렉션하는 멀티플렉서(Multiplexor) 서버에 연결합니다. 이 구성 요소는 기본적으로 활성화되지 않습니다. MMP와 메시지 저장소를 선택하면 같은 시스템에서 활성화됩니다. 따라서 구성 후 포트 번호를 변경하라는 경고 메시지가 나타납니다. 변경 방법에 대해서는 64 페이지 “1.12 사후 설치 포트 번호”를 참조하십시오.

  MMP를 구성하려면 7 장을 참조하십시오.

  구성할 구성 요소를 선택하고 구성하지 않을 구성 요소의 선택을 취소합니다.

8 구성한 파일을 소유할 사용자 아이디와 그룹을 입력합니다.

시스템 사용자와 그룹 설정에 대한 자세한 내용은 47 페이지 "1.1 UNIX 시스템 사용자와 그룹 만들기"를 참조하십시오.

9 Configuration Directory Server 패널

Configuration Directory LDAP URL, 관리자 및 비밀번호를 Administration Server 구성에서 가져와서 입력합니다. **(Messaging Server 6 2005Q4 이전 버전에 해당되며 그 이후 버전은 구성 데이터를 Directory Server에 저장하지 않고 Administration Server를 사용하지 않습니다.)**

Directory Server 설치에서 구성 서버 LDAP URL을 수집합니다. 표 D–1의 Directory Server 설치 워크시트를 참조하십시오.

Directory Manager는 Directory Server 및 이를 활용하는 모든 Sun Java System 서버(예: Messaging Server)에 대해 전반적인 관리자 권한이 있습니다. 또 Directory Server의 모든 항목에 대한 전체 관리 액세스 권한도 있습니다. 기본 및 권장 고유 이름(DN)은 `cn=Directory Manager`이고 Directory Server 구성 중에 설정됩니다.

주 – 기본값이 아닌 다른 값을 선택하면 Administration Server와 구성 Directory Server 사이의 불일치가 발생합니다. 이렇게 되면 사후 구성 단계를 수동으로 수행해야 합니다. 그러므로 확실히 다른 값이 필요한 지 확인하여 확인된 경우에만 이 항목을 수정하십시오.

---

10 사용자/그룹 Directory Server 패널

사용자 및 그룹 디렉토리 LDAP URL, 관리자 및 비밀번호를 입력합니다.

호스트에서 사용자/그룹 서버 LDAP URL 정보를 얻고 Directory Server 설치에서 포트 번호 정보를 얻습니다. 표 D–1의 Directory Server 설치 워크시트를 참조하십시오.

Directory Manager는 Directory Server 및 Directory Server를 사용하는 모든 Sun Java System 서버(예: Messaging Server)에 대해 전체적인 관리자 권한을 가지며 Directory Server의 모든 항목에 대해 완전한 관리 액세스 권한을 가집니다. 기본 및 권장 고유 이름(DN)은 `cn=Directory Manager`이고 Directory Server 구성 중에 설정됩니다.

복제된 Directory Server 인스턴스에 대해 설치를 하는 경우 마스터 디렉토리가 아니라 복제본의 자격 증명을 지정해야 합니다.

11 포스트마스터 전자 메일 주소

포스트마스터 전자 메일 주소를 입력합니다.

관리자가 주로 모니터할 주소를 선택합니다. 예를 들어, `siroe` 도메인의 포스트마스터의 경우 `pma@siroe.com`을 입력합니다. 이 주소의 첫 부분은 "Postmaster"가 될 수 .

전자 메일 주소의 사용자는 자동으로 생성되지 않습니다. 따라서 나중에 준비 도구를 사용하여 직접 만들어야 합니다.

12 관리자 계정의 비밀번호

서비스 관리자, 서버, 사용자/그룹 관리자, 최종 사용자 관리자 권한 및 PAB 관리자와 SSL 비밀번호에 사용될 초기 비밀번호를 입력합니다.

초기 런타임 구성 후 개별 관리자 계정에 대해 이 비밀번호를 변경할 수 있습니다. 자세한 내용은 101 페이지 "4.1 비밀번호 수정"을 참조하십시오.

13 기본 전자 메일 도메인

기본 전자 메일 도메인을 입력합니다.

이 전자 메일 도메인은 다른 도메인이 지정되지 않은 경우 사용되는 기본값입니다. 예를 들어, `siroe.com`이 기본 전자 메일 도메인인 경우 도메인 없이 사용자 아이디로 주소 지정된 메시지를 보낼 도메인입니다.

Sun LDAP Schema 2에 대한 사용자 및 그룹을 준비하기 위한 명령줄 인터페이스인 Delegated Administrator CLI를 사용하는 경우에는 구성 과정에서 동일한 기본 도메인을 지정합니다. 자세한 내용은 **Sun Java System Delegated Administrator 6.4 관리 설명서**를 참조하십시오.

14 조직 DN

사용자 및 그룹을 만들 조직 DN을 입력합니다. 기본값은 사용자/그룹 접미어 앞에 놓인 전자 메일 도메인입니다.

예를 들어, 사용자/그룹 접미어가 `o=usergroup`이고 전자 메일 도메인이 `siroe.com`인 경우 기본값은 `o=siroe.com, o=usergroup`입니다. 여기서 `o=usergroup`은 47 페이지 "1.1 UNIX 시스템 사용자와 그룹 만들기"에서 지정한 사용자/그룹 디렉토리 접미어입니다.

같은 사용자/그룹 디렉토리 접미어를 조직 DN으로 선택하면 호스트된 도메인을 만들려고 할 때 마이그레이션 문제가 발생할 수 있습니다. 초기 런타임 구성 도중 호스트된 도메인을 설정하려면 사용자/그룹 접미어의 한 수준 아래 DN을 지정하십시오.

15 구성 준비 완료

구성 프로그램이 시스템의 디스크 공간이 충분한지 확인한 다음 구성할 준비가 된 구성 요소를 개괄적으로 표시합니다.

메시징 구성 요소를 구성하려면 지금 구성을 선택합니다. 구성 변수를 변경하려면 뒤로를 선택합니다. 구성 프로그램을 종료하려면 취소를 선택합니다.

16 작업 시퀀스 시작, 시퀀스 완료 및 설치 요약 패널

최종 설치 요약 페이지에서 세부 정보를 선택하면 설치 상태를 볼 수 있습니다. 프로그램을 종료하려면 닫기를 선택합니다.

로그 파일이 *msg-svr-base*`/install/configure_`*YYYYMMDDHHMMSS*`.log`에 생성되며, 여기서 *YYYYMMDDHHMMSS*는 구성의 4자리 연도, 월, 일, 시, 분 및 초입니다.

이제 Messaging Server에 대한 초기 런타임 구성이 설정되었습니다. 구성 매개 변수를 변경하려면 이 설명서의 해당 부분을 참조하십시오.

Messaging Server를 시작하려면 다음 명령을 사용합니다.

```
/opt/SUNWmsgsr/sbin/start-msg
```

▼ 자동 설치 수행

Messaging Server 초기 런타임 구성 프로그램은 자동 설치 *state* 파일(saveState라고도 함)을 자동으로 만듭니다. 이 파일을 사용하여 Messaging Server Solaris 패키지가 설치된 배포 환경에 추가 Messaging Server 인스턴스를 빠르게 구성할 수 있습니다. 구성 질문에 대한 사용자의 모든 응답이 이 파일에 기록되어 있습니다.

자동 설치를 실행하면 configure 프로그램은 자동 설치 state 파일을 읽습니다. configure 프로그램은 Messaging Server의 이후 초기 런타임 구성에 대해 같은 설치 질문을 반복하지 않고 이 파일의 응답을 읽습니다. 새 설치에 state 파일을 사용하면 질문이 나타나지 않습니다. 대신 state 파일의 모든 응답이 새 설치 매개 변수로 자동으로 제공됩니다.

자동 설치 saveState *state* 파일은 *msg-svr-base*/install/configure_*YYYYMMDDHHMMSS* 디렉토리에 저장되어 있으며, 여기서 *YYYYMMDDHHMMSS*는 saveState 파일의 4자리 연도, 월, 일, 시, 분, 초를 나타냅니다.

자동 설치 *state* 파일을 사용하여 배포 환경 내 다른 시스템에 다른 Messaging Server 인스턴스를 구성하려면 다음 단계를 수행합니다.

1 자동 설치 *state* 파일을 새 설치를 수행하고 있는 시스템의 임시 영역에 복사합니다.

2 필요에 따라 자동 설치 *state* 파일을 검토하고 편집합니다.

state 파일의 일부 매개 변수와 지정 내용을 변경해야 할 수 있습니다. 예를 들어 새 설치의 기본 전자 메일 도메인은 *state* 파일에 기록된 기본 전자 메일 도메인과 다를 수 있습니다. *state* 파일에 나열된 매개 변수는 이 설치에 자동으로 적용된다는 점에 유의하십시오.

3 다음 명령을 실행하여 자동 설치 파일로 다른 시스템을 구성합니다.

msg-svr-base`/sbin/configure -nodisplay -noconsole -state \`
fullpath`/saveState`

여기서 *fullpath*는 saveState 파일이 있는 전체 디렉토리 경로입니다. 이 절의 단계 1을 참조하십시오.

주 – 자동 설치 프로그램을 실행하면 자동 설치에서 *msg-svr-base*/install/configure_*YYYYMMDDHHMMSS*/saveState 디렉토리 위치에 *state* 파일이 생성됩니다. 여기서 *YYYYMMDDHHMMSS*는 saveState 파일의 4자리 연도, 월, 일, 시, 분, 초입니다.

1.4 Directory Server 복제본에 대해 Messaging Server 설치

다음과 같은 제한 사항으로 인해 Directory Server 복제본에 대해 Messaging Server를 설치하지 못할 수 있습니다.

- Directory Server 마스터 자격 증명이 없는 경우
- Messaging Server가 Directory Server 마스터와 직접 통신할 수 없는 경우

▼ Directory Server 복제본에 대해 Messaging Server를 설치하는 방법

1 **Directory Server 복제본이 들어 있는 모든 Directory Server에 대해 `comm_dssetup.pl` 프로그램을 실행합니다(Sun Java Communications Suite 5 Installation Guide의 "Messaging Server Postinstallation Configuration" 참조).**

2 **49 페이지 "1.3 Messaging Server 초기 런타임 구성 만들기 "에서 설명하는 대로 복제된 Directory Server 자격 증명을 사용하여 Messaging `configure` 프로그램을 실행합니다.**

기본적으로 이 프로그램은 *msg-svr-base*`/sbin/configure`에 있습니다.

잘못된 권한 때문에 `configure` 프로그램은 Directory Server 관리자를 구성하다가 실패하게 됩니다. 하지만 Directory Server 복제본에 올바른 권한을 허용하는 데 필요한 *msg-svr-base*`/config/*.ldif` 파일이 생성됩니다.

3 **`*.ldif` 파일을 Directory Server 마스터로 옮깁니다.**

4 **`*.ldif` 파일에 대해 `ldapmodify` 명령을 실행합니다.**

`ldapmodify` 또는 *msg-svr-base* `/install/configure_`*YYYYMMDDHHMMSS*`.log`에 대한 자세한 내용은 Sun Java System Directory Server 설명서를 참조하십시오.

5 **`configure` 프로그램을 다시 실행합니다.**

이제 Directory Server 복제본(및 마스터)이 Messaging Server와 함께 사용할 수 있도록 구성되었습니다.

1.5 Messaging Server 준비 도구 설치

다음 절에서는 지원되는 준비 도구에 대한 설치 정보를 요약하여 설명합니다.

- 56 페이지 "1.5.1 메시징용 Schema 1 Delegated Administrator"
- 57 페이지 "1.5.2 LDAP 준비 도구"
- 47 페이지 "1.1 UNIX 시스템 사용자와 그룹 만들기"

1.5.1 메시징용 Schema 1 Delegated Administrator

Messaging Server에 사용할 수 있는 GUI 준비 도구는 두 가지로서 iPlanet Delegated Administrator(Sun LDAP Schema 1)와 Communications Services Delegated Administrator(Sun LDAP Schema 2)가 있습니다. 이 절에서는 iPlanet Delegated Administrator(Sun LDAP Schema 1)에 대해 설명합니다Communications Services Delegated Administrator(Sun LDAP Schema 2)에 대한 자세한 내용은 **Sun Java System Delegated Administrator 6.4 관리 설명서**를 참조하십시오.

iPlanet Delegated Administrator(Sun LDAP Schema 1)를 설치하려면 Sun Software 페이지에서 다운로드해야 합니다. 다운로드 위치 정보에 대해서는 Sun Java System 담당자에게 문의하십시오.

주 – iPlanet Delegated Administrator는 Messaging Server와 Web Server를 설치 및 구성한 후에만 설치할 수 있습니다. iPlanet Delegated Administrator 설치에 대한 자세한 내용은 iPlanet Delegated Administrator 설명서를 참조하십시오.

iPlanet Delegated Administrator는 기존 Messaging Server 5.x를 설치했고 현재 Messaging Server 6을 설치 중인 고객만 사용할 수 있습니다. Messaging Server 제품을 새로 구입하는 고객은 사용할 수 없습니다.

iPlanet Delegated Administrator는 Sun Java System Web Server 6.0(이전 Messaging Server 5.2에서만 번들로 포함됨)과 함께 사용해야 합니다. Web Server 6.1(Java Enterprise System 설치 프로그램에 번들됨)은 iPlanet Delegated Administrator와 함께 사용할 수 없습니다.

주 – 다음 제품을 설치할 때는 Java Enterprise System 설치 프로그램을 사용합니다. 이 제품 중 일부는 고유의 구성을 갖고 있지만 그 밖의 제품의 구성은 Java Enterprise System 설치/구성 프로그램에 포함되어 있습니다. 자세한 내용은 특정 제품 설명서를 참조하십시오.

▼ iPlanet Delegated Administrator 설치 방법

1. **Sun Java System Directory Server 5.2가 설치 및 구성되어 있어야 합니다.**
   자세한 내용은 해당 **Sun Java System Directory Server 설치 설명서**를 참조하십시오.

2. **Messaging Server를 설치 및 구성합니다.**
   Sun Java System Access Manager가 설치되지 않을 것이기 때문에 Messaging Server는 사용자가 Sun LDAP Schema 1을 사용하고 있다는 것을 감지합니다.

3. **이전 Messaging Server 5.2 번들에서 Sun Java System Web Server 6.0을 설치합니다.**
   Sun Java System Web Server 설명서 및 Sun Java System Delegated Administrator 설명서를 참조하십시오.

4 **메시징용 iPlanet Delegated Administrator 1.2 Patch 2를 설치합니다.**
최신 버전을 구하려면 Sun 기술 지원 담당자에게 문의하십시오.

iPlanet Delegated Administrator 설명서를 참조하십시오.

1.5.2 LDAP 준비 도구

Sun LDAP Schema 1 사용자 및 그룹은 LDAP Directory 도구를 사용하여 준비할 수 있습니다. Schema 2는 지원되지 않습니다.

▼ Schema 1 LDAP 준비 도구 설치 방법

1 **Directory Server가 아직 설치되어 있지 않은 경우에는 해당 서버를 설치하고 구성해야 합니다.**
자세한 내용은 **Sun Java Enterprise System 5 Installation Guide for UNIX**를 참조하십시오.

2 **Access Manager가 Directory Server의 데이터를 인식하도록 구성합니다.**
Access Manager가 LDAP 디렉토리의 데이터를 인식하도록 하려면 Access Manager 가 관리할 모든 조직, 그룹 및 사용자에 대한 항목에 특별한 객체 클래스를 추가해야 합니다. 아직 이렇게 하지 않은 경우 새 계정 준비를 시작하기 전에 먼저 이 작업을 수행하십시오. 이러한 객체 클래스를 디렉토리에 자동으로 추가하는 것을 돕기 위해 Access Manager 제품에는 샘플 스크립트가 포함되어 있습니다. 이러한 사후 설치 단계에 대한 자세한 내용은 **Sun Java System Access Manager Migration Guide**를 참조하십시오.

3 **이 설명서의 지침에 따라 Messaging Server를 설치 및 구성합니다.**
Messaging Server는 Access Manager가 설치되어 있는지 여부에 따라 현재 사용 중인 Sun Java System LDAP Schema를 식별합니다.

4 **Messenger Express의 메일 필터링을 활성화하려면 Sun Java System Web Server 6.1을 설치 및 구성하십시오.**
메일 필터링 활성화에 대한 자세한 내용은 62 페이지 “1.9 Messenger Express 및 Communications Express 메일 필터 구성”을 참조하십시오.

메일 필터링은 준비 도구는 아니지만 그 기능은 메시징용 Delegated Administrator의 이전 GUI 버전에 포함되어 있습니다.

5 **Sun Java System Messaging Server 설명서를 참조하여 LDAP 준비를 수행합니다.**
Sun LDAP Schema 1 LDAP 준비에 대해서는 **iPlanet Messaging Server 5.2 Provisioning Guide** 및 **Sun Java Communications Suite 5 Schema Reference**를 참조하십시오. **Schema Reference**에는 Sun LDAP Schema 1 및 v.2에 대한 객체 클래스와 속성이 있습니다.

1.6 SMTP 릴레이 차단

기본적으로 Messaging Server는 SMTP 릴레이 시도를 차단하도록 구성되어 있습니다. 즉, 인증되지 않은 외부 소스의 외부 주소로의 메시지 전송 시도를 거부합니다. 외부 시스템은 서버가 있는 호스트가 아닌 모든 시스템을 말합니다. 이 기본 구성은 다른 모든 시스템을 외부 시스템으로 간주하기 때문에 과도하게 SMTP 중계를 차단합니다.

설치한 뒤에는 사이트의 필요에 맞게 구성을 수동으로 수정하는 것이 중요합니다. 특히 Messaging Server가 SMTP 릴레이를 항상 허용해야 하는 자체 내부 시스템과 서브넷을 인식해야 합니다. 이 구성을 업데이트하지 않으면 MTA 구성을 테스트할 때 문제가 발생할 수 있습니다.

Messaging Server 시스템의 SMTP 서버를 통해 외부 주소로 지정된 메시지를 전송하려고 시도하는 IMAP 및 POP 클라이언트, 그리고 SMTP AUTH(SASL)를 사용하여 인증하지 않는 클라이언트의 전송 시도는 거부됩니다. 내부로 인식되는 시스템 및 서브넷은 일반적으로 *msg-svr-base*`/config/mappings` 파일에 포함되어 있는 `INTERNAL_IP` 매핑 테이블을 통해 제어됩니다.

예를 들어, IP 주소가 `192.45.67.89`인 Messaging Server 시스템의 경우 기본 `INTERNAL_IP` 매핑 테이블은 다음과 같습니다.

```
INTERNAL_IP

  $(192.45.67.89/32)  $Y
  127.0.0.1  $Y
  *  $N
```

첫 번째 항목은 `$(IP-pattern/significant-prefix-bits)` 구문을 사용하여 `192.45.67.89`의 전체 32비트와 일치하는 모든 IP 주소를 내부로 인식하도록 지정합니다. 두 번째 항목은 루프백 IP 주소 `127.0.0.1`을 내부로 인식합니다. 마지막 항목은 다른 모든 IP 주소가 내부로 인식되지 않도록 지정합니다.

마지막 `$N` 항목 앞에 추가 IP 주소 또는 서브넷을 지정하여 항목을 추가할 수 있습니다. 이러한 항목은 왼쪽에 IP 주소나 서브넷(`$(.../...)` 구문을 사용하여 서브넷 지정)을 지정하고 오른쪽에 `$Y`를 지정합니다. 또는 기존 `$(.../...)` 항목을 수정하여 더 일반적인 서브넷을 허용할 수 있습니다.

예를 들어, 동일한 샘플 사이트의 네트워크가 클래스 C 네트워크, 즉 `192.45.67.0` 서브넷을 모두 소유하는 네트워크인 경우 해당 사이트에서는 첫 번째 항목을 수정하여 매핑 테이블이 다음과 같도록 해야 합니다.

```
INTERNAL_IP

  $(192.45.67.89/24)  $Y
  127.0.0.1  $Y
  *  $N
```

또는 사이트가 192.45.67.80-192.45.67.99 범위 내의 IP 주소만 소유하는 경우 해당 사이트에서는 다음과 같은 매핑 테이블을 사용할 수 있습니다.

```
INTERNAL_IP

! Match IP addresses in the range 192.45.67.80-192.45.67.95
  $(192.45.67.80/28)  $Y
! Match IP addresses in the range 192.45.67.96-192.45.67.99
  $(192.45.67.96/30)  $Y
  127.0.0.1  $Y
  *  $N
```

msg-svr-base/sbin/ imsimta test-match 유틸리티는 IP 주소가 특정 $(.../...) 테스트 조건에 일치하는지 여부를 검사할 때 유용하게 사용할 수 있습니다. imsimta test -mapping 유틸리티는 INTERNAL_IP 매핑 테이블이 다양한 IP 주소 입력에 대해 원하는 결과를 반환하는지 검사할 때 매우 유용합니다.

INTERNAL_IP 매핑 테이블을 수정한 뒤에는 *msg-svr-base*/sbin/imsimta cnbuild 및 *msg-svr-base*/sbin/ imsimta restart 유틸리티를 실행해야 변경 사항이 적용됩니다.

매핑 파일 및 일반 매핑 테이블 형식과 imsimta 명령줄 유틸리티에 대한 자세한 내용은 **Sun Java System Messaging Server 6.3 Administration Reference**의 2 장, "Message Transfer Agent Command-line Utilities"를 참조하십시오. INTERNAL_IP 매핑 테이블에 대한 자세한 내용은 526 페이지 "18.6 SMTP 릴레이 추가"를 참조하십시오.

1.7 재부트 후 시작 활성화

부트 스크립트인 *msg-svr-base*/lib/ Sun_MsgSvr을 사용하여 시스템 재부트 후에 Messaging Server 시작을 활성화할 수 있습니다. 기본적으로 Messaging Server는 이 스크립트를 실행하지 않는 한 시스템 재부트 후에 다시 시작되지 않습니다. 또한 이 스크립트가 활성화되면 MMP를 시작할 수도 있습니다.

▼ 재부트 후에 Messaging Server를 활성화하는 방법

1 **Sun_MsgSvr 스크립트를 /etc/init.d 디렉토리에 복사합니다.**

2 **Sun_MsgSvr 스크립트에 대한 다음 소유권 및 액세스 모드를 변경합니다.**

소유권(chown(1M))	그룹 소유권(chgrp(1M))	액세스 모드(chmod(1M))
root(수퍼유저)	sys	0744

3 **/etc/rc2.d 디렉토리로 이동하여 다음과 같은 링크를 만듭니다.**

```
ln /etc/init.d/Sun_MsgSvr S92Sun_MsgSvr
```

4 **/etc/rc0.d 디렉토리로 이동하여 다음과 같은 링크를 만듭니다.**

```
ln /etc/init.d/Sun_MsgSvr K08Sun_MsgSvr
```

1.8 sendmail 클라이언트 처리

최종 사용자가 sendmail 클라이언트를 통해 메시지를 보내는 경우 프로토콜을 통해 해당 클라이언트를 지원하도록 Messaging Server를 구성할 수 있습니다. 사용자는 UNIX sendmail 클라이언트를 계속 사용할 수 있습니다.

sendmail 클라이언트와 Messaging Server 사이에 호환성을 만들려면 sendmail 구성 파일을 만들어 수정하면 됩니다.

주 – 새 sendmail 패치를 시스템에 적용할 때마다 61 페이지 "Solaris 9 플랫폼에서 sendmail 구성 파일을 만드는 방법"의 다음 지침에 설명된 대로 submit.cf 파일을 수정해야 합니다. Solaris 8의 경우에는 60 페이지 "Solaris 8에서 /usr/lib/sendmail의 올바른 버전을 구하는 방법"의 지침을 따르십시오.

이전 버전의 Messaging Server를 설치한 경우 /usr/lib/sendmail 이진 파일이 Messaging Server 제품의 구성 요소로 교체되었습니다. Messaging Server 6.0부터 현재 버전까지에서는 설치 중에 이 교체가 더 이상 필요하지 않습니다. 따라서 최신 sendmail 패치로부터 적절한 /usr/lib/sendmail 이진 파일 버전을 구해야 할 수 있습니다.

Solaris 9 플랫폼에서 sendmail은 setuid 프로그램이 아닌 setgid 프로그램입니다.

▼ Solaris 8에서 /usr/lib/sendmail의 올바른 버전을 구하는 방법

1 **/usr/lib/mail/cf 디렉토리에서 main-v7sun.mc 파일을 찾아 이 파일의 복사본을 만듭니다.**

이 절의 예에서는 sunone-msg.mc라는 복사본을 만들었습니다.

2 **sunone-msg.mc 파일에서 MAILER 매크로 앞에 다음 행을 추가합니다.**

```
FEATURE('nullclient', 'smtp:rhino.west.sesta.com')dnl
MASQUERADE_AS('west.sesta.com')dnl
define('confDOMAIN_NAME', 'west.sesta.com')dnl
```

49 페이지 "1.3 Messaging Server 초기 런타임 구성 만들기 "에서 설명하는 것처럼, 여기서 rhino.west.sesta.com은 localhost 이름이고 west.sesta.com은 기본 전자 메일 도메인입니다. HA 환경에서는 논리 호스트 이름을 사용합니다. 고가용성을 위한 논리 호스트 이름에 대한 자세한 내용은 3 장을 참조하십시오.

3 sunone-msg.mc 파일을 컴파일합니다.

```
/usr/ccs/bin/make sunone-msg.cf
```

sunone-msg.mc는 sunone-msg.cf를 출력합니다.

4 /etc/mail 디렉토리에 있는 기존 sendmail.cf 파일의 백업 복사본을 만듭니다.

a. /usr/lib/mail/cf/sunone-msg.cf를 복사한 다음 해당 파일의 이름을 sendmail.cf로 변경합니다.

b. 새 sendmail.cf 파일을 /etc/mail 디렉토리로 이동합니다.

▼ Solaris 9 플랫폼에서 sendmail 구성 파일을 만드는 방법

1 /usr/lib/mail/cf 디렉토리에서 submit.mc 파일을 찾아 해당 파일의 복사본을 만듭니다.
이 절의 예에서는 sunone-submit.mc라는 복사본을 만들었습니다.

2 sunone-submit.mc 파일에서 다음 행을

```
FEATURE("msp')dn
```

위 항목을 아래와 같이 변경하려면

```
FEATURE("msp', "rhino.west.sesta.com')dnl
```

여기서 rhino.west.sesta.com은 localhost 이름입니다.

49 페이지 "1.3 Messaging Server 초기 런타임 구성 만들기 "에서 설명하는 것처럼, 여기서 rhino.west.sesta.com은 localhost 이름이고 west.sesta.com은 기본 전자 메일 도메인입니다. HA 환경에서는 논리 호스트 이름을 사용합니다. 고가용성을 위한 논리 호스트 이름에 대한 자세한 내용은 3 장을 참조하십시오.

3 sunone-submit.mc 파일을 컴파일합니다.

```
/usr/ccs/bin/make sunone-submit.cf
```

sunone-submit.mc는 sunone-submit.cf를 출력합니다.

4 /etc/mail 디렉토리에 있는 기존 submit.cf 파일의 백업 복사본을 만듭니다.

a. /usr/lib/mail/cf/sunone-submit.cf 파일을 복사한 다음 해당 파일의 이름을 submit.cf로 변경합니다.

b. 새 submit.cf 파일을 /etc/mail 디렉토리로 이동합니다.

1.9 Messenger Express 및 Communications Express 메일 필터 구성

메일 필터는 Messenger Express 및 Communications Express를 통해 액세스할 수 있습니다. Communications Express만 사용할 때는 `.war` 파일을 배포할 필요가 없지만 Messenger Express 안의 메일 필터를 배포하려면 다음 명령을 실행해야 합니다.

*Web Server*를 웹 컨테이너로 사용하는 경우:

```
# cd web_svr_base/bin/https/httpadmin/bin
# ./wdeploy deploy -u /MailFilter -i https-srvr_instance \
-v https-virtual_srvr_instance msg_svr_base/SUNWmsgmf/MailFilter.war
```

*Application Server*를 웹 컨테이너로 사용하는 경우:

```
# cd app_svr_base/sbin
# ./asadmin
asadmin> deploy --user admin msg_svr_base/SUNWmsgmf/MailFilter.war
```

두 경우 모두 다음 `configutil` 매개 변수를 설정하고 `mshttpd`를 다시 시작합니다.

```
# cd msg_svr_base/sbin
# ./configutil -o "local.webmail.sieve.port" \
-v "WS_port_no|AS_port_no"
# ./stop-msg http
# ./start-msg http
```

최종 사용자에 대한 메일 필터 관련 정보는 Messenger Express 및 Communications Express 온라인 도움말 파일에서 볼 수 있습니다.

1.10 성능 및 조정

Sun Java Communications Suite 5 Deployment Planning Guide의 "Performance Considerations for a Messaging Server Architecture"를 참조하십시오.

1.11 사후 설치 디렉토리 레이아웃

Sun Java System Messaging Server를 설치하면 디렉토리 및 파일이 표 1–1에 설명된 구조대로 정렬됩니다. 이 표에는 일반적인 서버 관리 작업에서 가장 많이 사용되는 디렉토리와 파일만 나열되어 있습니다.

표 1–1 사후 설치 디렉토리 및 파일

디렉토리	기본 위치 및 설명
Messaging Server 기본	`/opt/SUNWmsgsr/`
(*msg_svr_base*)	(기본 위치)
	서버 프로그램, 구성, 유지 관리 및 정보 파일을 저장하는 Messaging Server 시스템의 전용 디렉토리입니다.
	Messaging Server 기본 디렉토리는 시스템당 하나만 있을 수 있습니다.
구성	*msg_svr_base*`/config/`
`config`	`imta.cnf` 및 `msg.conf`와 같은 모든 Messaging Server 구성 파일이 포함되어 있습니다.
	Solaris 및 Linux 플랫폼의 경우: 이 디렉토리는 초기 런타임 구성에서 지정한 데이터 및 구성 디렉토리의 `config` 하위 디렉토리(기본값: `/var/opt/SUNWmsgsr/`)에 심볼릭 링크로 연결됩니다.
로그	*msg_svr_base*`/log/`
`log`	`mail.log_current`와 같은 Messaging Server 로그 파일이 포함되어 있습니다.
	Solaris 및 Linux 플랫폼의 경우: 이 디렉토리는 초기 런타임 구성에서 지정한 데이터 및 구성 디렉토리의 `log` 하위 디렉토리(기본값: `/var/opt/SUNWmsgsr/`)에 심볼릭 링크로 연결됩니다.
데이터	*msg_svr_base*`/data/`
`data`	(필수 위치)
	데이터베이스, 구성, 로그 파일, 사이트 프로그램, 대기열, 저장소 및 메시지 파일이 포함되어 있습니다.
	`data` 디렉토리에는 `config` 및 `log` 디렉토리가 포함되어 있습니다.
	Solaris 및 Linux 플랫폼의 경우: 이 디렉토리는 초기 런타임 구성에서 지정한 데이터 및 구성 디렉토리(기본값: `/var/opt/SUNWmsgsr/`)에 심볼릭 링크로 연결됩니다.
시스템 관리자 프로그램	*msg_svr_base*`/sbin/`
`sbin`	(필수 위치)
	`imsimta`, `configutil`, `stop-msg`, `start-msg`, `uninstaller` 등과 같은 Messaging Server 시스템 관리자 실행 프로그램 및 스크립트가 포함되어 있습니다.

표 1–1 사후 설치 디렉토리 및 파일 *(계속)*

디렉토리	기본 위치 및 설명
라이브러리 `lib`	*msg_svr_base*`/lib/` (필수 위치) 공유 라이브러리, 개인 실행 프로그램과 스크립트, 데몬 및 사용자 정의가 불가능한 내용 데이터 파일이 포함되어 있습니다. 예를 들면 다음과 같습니다. (`imapd` 및 `qm_maint.hlp`).
SDK 포함 파일 `include`	*msg_svr_base*`/include/` (필수 위치) SDK (Software Development Kit)의 메시징 헤더 파일이 포함되어 있습니다.
예 `examples`	*msg_svr_base*`/examples/` (필수 위치) Messenger Express AUTH SDK와 같은 다양한 SDK의 예가 포함되어 있습니다.
설치 데이터 `install`	*msg_svr_base*`/install/` (필수 위치) 설치 로그 파일, 자동 설치 파일, 기본 구성 파일 및 초기 런타임 구성 로그 파일 등과 같은 설치 관련 데이터 파일이 포함되어 있습니다.

1.12 사후 설치 포트 번호

설치 및 초기 런타임 구성 프로그램을 실행하는 도중 다양한 서비스에 대한 포트 번호를 선택하게 됩니다. 이러한 포트 번호는 1부터 65535 사이의 임의의 숫자일 수 있습니다.

표 1–2에는 설치 후 지정되는 포트 번호가 나열되어 있습니다.

표 1–2 설치 도중 지정되는 포트 번호

포트 번호	서비스(`configutil` 매개 변수)
389	Directory Server를 설치하는 시스템의 표준 Directory Server LDAP 포트. 이 포트는 Directory Server 설치 프로그램에 지정되었습니다. (`local.ugldapport`)
110	표준 POP3 포트. 이 포트는 같은 시스템에 설치될 때 MMP 포트와 충돌을 일으킬 수 있습니다. (`service.pop.port`)
143	표준 IMAP4 포트. 이 포트는 같은 시스템에 설치될 때 MMP 포트와 충돌을 일으킬 수 있습니다. (`service.imap.port`)

표 1–2 설치 도중 지정되는 포트 번호 *(계속)*

포트 번호	서비스(configutil 매개 변수)
25	표준 SMTP 포트(service.http.smtpport)
80	Messenger Express HTTP 포트. 이 포트는 같은 시스템에 설치될 때 Web Server 포트와 충돌을 일으킬 수 있습니다. (service.http.port).
995	SSL에서의 POP3 포트. 암호화된 통신용으로 (service.pop.sslport)
993	SSL에서의 IMAP 포트. 암호화된 통신용으로 이 포트는 같은 시스템에 설치될 때 MMP 포트와 충돌을 일으킬 수 있습니다. (service.imap.sslport)
443	SSL에서의 HTTP 포트. 암호화된 통신용으로 (service.http.sslport)
7997	메시징 및 공동 작업 ENS(Event Notification Service) 포트
27442	내부 제품 통신을 위해 작업 제어기에서 사용하는 포트
49994	내부 제품 통신을 위해 Watcher에서 사용하는 포트. Watcher에 대한 자세한 내용은 **Sun Java System Messaging Server 관리 설명서**를 참조하십시오. (local.watcher.port)

같은 시스템에 여러 제품이 설치되어 있으면 포트 번호 충돌이 발생할 수 있습니다. 표 1–3에서는 잠재적인 포트 번호 충돌을 보여 줍니다.

표 1–3 잠재적 포트 번호 충돌

충돌하는 포트 번호	포트	충돌하는 포트
995	SSL을 통한 POP3	MMP POP3 프록시(SSL 사용)
143	IMAP 서버	MMP IMAP 프록시
110	POP3 서버	MMP POP3 프록시
993	SSL에서의 IMAP	MMP IMAP 프록시(SSL 사용)
80	Web Server 포트	Messenger Express

가능한 한, 포트 번호가 충돌하는 제품은 별도의 시스템에 설치해야 합니다. 그렇게 할 수 없는 경우에는 충돌하는 제품 중 하나의 포트 번호를 변경해야 합니다.

▼ 포트 번호 변경 방법

- **포트 번호를 변경하려면 configutil 유틸리티를 사용합니다.**
  전체 구문과 사용법에 대해서는 **Sun Java System Messaging Server 6.3 Administration Reference**의 “configutil”을 참조하십시오.

예 1–1 Messenger Express HTTP 포트 번호 변경

다음 예에서는 `service.http.port` `configutil` 매개 변수를 사용하여 Messenger Express HTTP 포트 번호를 `8080`으로 변경합니다.

```
# configutil -o service.http.port -v 8080
# stop-msg http
$start-msg http
```

◆ ◆ ◆ 2 장

2 Messaging Server 5.2에서 Sun Java System Messaging Server로 업그레이드

이 장의 정보는 **Sun Java Communications Suite 5 Upgrade Guide**로 옮겨졌습니다. 자세한 내용은 이 설명서를 참조하십시오. 641 페이지 "20.15 메일함을 새 시스템으로 이동 또는 마이그레이션"은 여전히 이 설명서에 포함되어 있습니다.

2.1 이동된 정보

Sun Java Communications Suite 5 Upgrade Guide를 참조하십시오.

◆ ◆ ◆ 3 장

3 고가용성 구성

이 절에서는 Veritas Cluster Server 또는 Sun Cluster 고가용성 클러스터링 소프트웨어를 구성하고 Messaging Server와 함께 사용할 준비를 하는 데 필요한 정보를 제공합니다. 이 장에서는 **Sun Java Communications Suite 5 Deployment Planning Guide**의 6 장, "Designing for Service Availability" 및 Veritas 또는 Sun Cluster Server 설명서에서 세부 계획, 설치 지침, 필수 패치 및 기타 필요한 정보에 대해 읽어본 것으로 간주합니다.

이 장은 다음 내용으로 구성되어 있습니다.

3.1 지원되는 버전

최신 지원 버전 및 플랫폼에 대한 자세한 내용은 **Sun Java Communications Suite 5 릴리스 노트**의 "이 Messaging Server 릴리스의 새로운 기능"을 참조하십시오.

3.2 고가용성 모델

여기에는 Messaging Server에서 사용 가능한 다양한 고가용성 모델이 있습니다. 기본적인 세 가지 모델은 다음과 같습니다.

이러한 각 모델에 대해서는 다음 하위 절에 자세히 설명되어 있습니다.

지원되는 모델은 HA 제품에 따라 다를 수 있습니다. 지원되는 모델에 대해서는 HA 설명서를 참조하십시오.

3.2.1 비대칭

기본 비대칭 또는 **핫 대기** 고가용성 모델은 두 개의 클러스터된 호스트 시스템 또는 **노드**로 구성되어 있습니다. 논리적 IP 주소 및 관련 호스트 이름이 두 노드 모두에 지정됩니다.

이 모델에서는 한 번에 하나의 노드만 활성화되므로, 백업 또는 핫 대기 노드는 대부분 유휴 상태로 유지됩니다. 두 노드 사이에 단일 공유 디스크 배열이 구성되고 활성 또는 **기본** 노드가 마스터로 지정됩니다. 메시지 저장소 분할 영역 및 MTA(Mail Transport Agent) 대기열이 이 공유 볼륨에 있습니다.

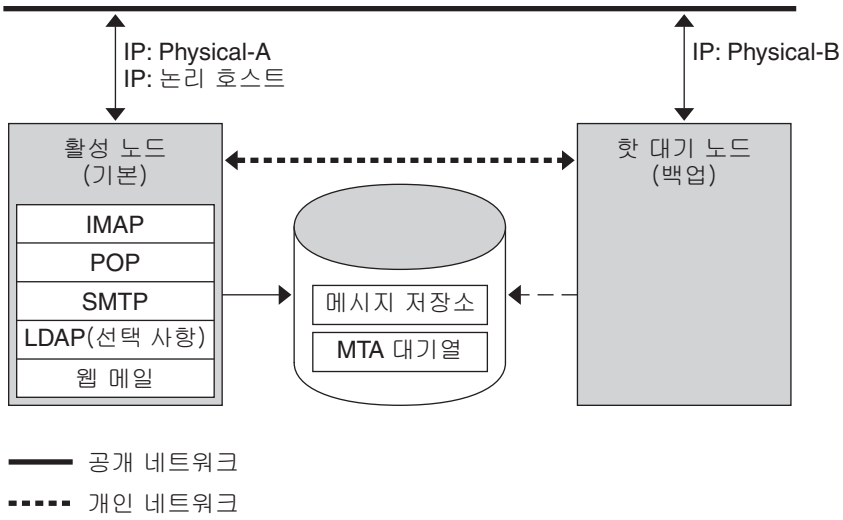


그림 3–1 비대칭 고가용성 모드

위 그림에서는 두 개의 물리적 노드인 `Physical-A`와 `Physical-B`를 보여 줍니다. 페일오버 전에는 `Physical-A`가 활성 노드입니다. 페일오버를 수행하면 `Physical-B`가 활성 노드가 되고 공유 볼륨이 전환되므로 `Physical-B`가 마스터로 지정됩니다. 모든 서비스는 `Physical-A`에서 중지되고 `Physical-B`에서 시작됩니다.

백업 노드가 기본 노드 전용으로 예약되어 있다는 점이 이 모델의 장점입니다. 또한 페일오버가 발생할 때 백업 노드에서 자원 경합이 발생하지 않습니다. 또한 이 모델은 백업 노드가 대부분 유휴 상태로 유지되므로 이 자원이 제대로 활용되지 않음을 의미하기도 합니다.

3.2.2 대칭

기본 대칭 또는 "이중 서비스" 고가용성 모델은 두 개의 호스팅 시스템으로 구성되며 각 시스템은 고유한 논리 IP 주소를 가집니다. 각 논리 노드는 하나의 물리 노드에 연결되며 두 개의 저장소 볼륨을 가진 하나의 디스크 배열을 제어합니다. 볼륨 중 하나는 로컬 메시지 저장소 분할 영역 및 MTA 대기열에 사용되고, 다른 하나는 파트너 메시지 저장소 분할 영역 및 MTA 대기열의 미러 이미지입니다.

다음 그림에서는 대칭 고가용성 모드를 보여 줍니다. 두 노드가 동시에 모두 활성화되며 각 노드는 다른 노드의 백업 노드 역할을 합니다. 일반적으로 각 노드는 Messaging Server의 인스턴스를 하나만 실행합니다.

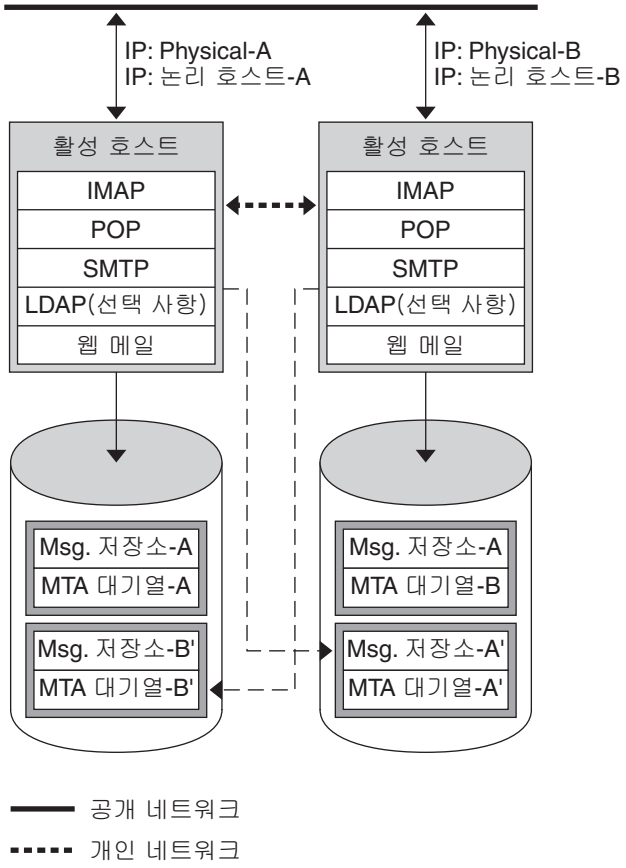


그림 3–2 대칭 고가용성 모드

페일오버를 수행하면 실패한 노드의 서비스가 종료되었다가 백업 노드에서 다시 시작됩니다. 이 시점에서는 백업 노드가 두 노드 모두에 대해 Messaging Server를 실행하고 두 개별 볼륨을 관리합니다.

두 노드가 동시에 활성화되므로 시스템 자원을 완전히 활용할 수 있다는 점이 이 모델의 장점입니다. 그러나 페일오버 중에는 두 노드 모두에서 Messaging Server에 대한 서비스가 실행되도록 백업 노드에서 많은 자원 경합이 발생합니다. 따라서 실패한 노드를 가능한 빨리 복구하여 서버를 이중 서비스 상태로 다시 전환해야 합니다.

또한 이 모델은 백업 저장소 배열을 제공합니다. 디스크 배열 오류가 발생할 경우 백업 노드의 서비스에서 중복 이미지를 선택할 수 있습니다.

대칭 모델을 구성하려면 공유 디스크에 공유 이진을 설치해야 합니다. 그렇게 하면 Messaging Server 패치 릴리스 중에 시스템을 업데이트할 수 있는 기능인 롤링 업그레이드가 금지될 수 있습니다. 이 기능은 이후에 릴리스될 예정입니다.

3.2.3 N+1(N Over 1)

N + 1 또는 "N over 1" 모델은 다중 노드 비대칭 구성에서 작동합니다. N개의 논리 호스트 이름과 N개의 공유 디스크 배열이 필요합니다. 단일 백업 노드가 모든 다른 노드의 핫 대기 노드로 예약되어 있습니다. 백업 노드는 N개의 노드에서 Messaging Server를 동시에 실행할 수 있습니다.

아래 그림에서는 기본 N + 1 고가용성 모델을 보여 줍니다.

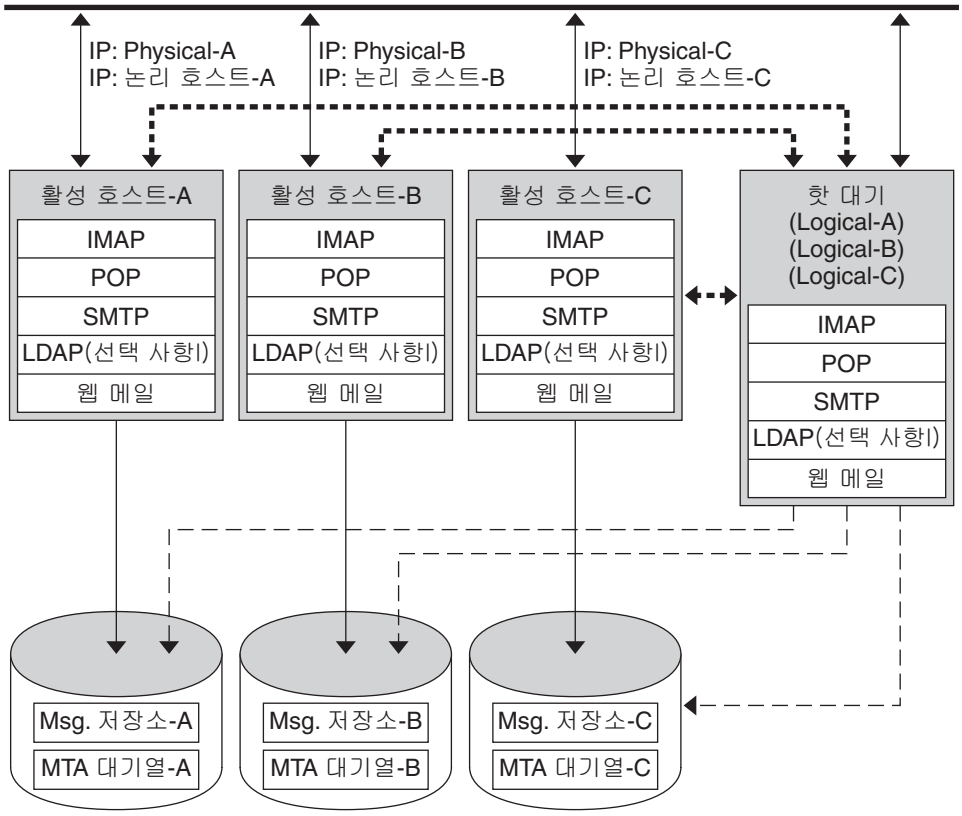


그림 3–3 N + 1 고가용성 모드

하나 이상의 활성 노드를 페일오버할 때 백업 노드는 실패한 노드의 권한을 선택합니다.

서버 로드가 여러 노드로 분산되고 하나의 백업 노드만으로 가능한 모든 노드 실패를 처리할 수 있다는 점이 N + 1 모델의 장점입니다. 따라서 시스템 유휴 비율이 1/N이 됩니다. 단일 비대칭 모델의 경우에는 1/1입니다.

N+1 모델을 구성하려면 로컬 디스크에만 이진을 설치해야 합니다. 즉, 대칭 모델과 마찬가지로 공유되지 않는 디스크에 이진을 설치해야 합니다. 현재 Messaging Server 설치 및 설정 프로세스에서는 대칭, 1+1 또는 N+1 비대칭/대칭 HA 솔루션의 공유 디스크에 이진을 강제로 설치합니다.

3.2.4 고가용성 모델 선택

다음 표에는 각 고가용성 모델의 장점과 단점이 요약되어 있습니다. 이 정보를 사용하면 배포에 적합한 모델을 결정하는 데 도움이 됩니다.

표 3–1 HA 모델 비교

모델	장점	단점	권장 사용자
비대칭	■ 간단한 비교 ■ 백업 노드가 100% 예약됨	시스템 자원이 완전히 활용되지 않음	향후에 확장할 계획이 있는 소규모 서비스 공급자
대칭	■ 우수한 시스템 자원 활용 ■ 고가용성	백업 노드의 자원 경합 HA에 완전 중복 디스크 필요	단일 서버 오류가 발생할 경우 성능 감소를 수용할 수 있는 소규모 기업 배포
N + 1	■ 로드 분산 ■ 쉬운 확장	관리 및 구성의 복잡성	자원을 구속하지 않고 배포해야 하는 대규모 서비스 공급자

3.2.5 시스템 중단 시간 계산

다음 표는 시스템 오류로 인해 지정된 날짜에 메시징 서비스를 사용할 수 없는 가능성을 보여 줍니다. 이러한 계산에서는 시스템 충돌이나 서버 중단으로 인해 평균적으로 3개월에 1일 동안 각 서버가 중단되고 각 저장 장치가 12개월에 1일 동안 중단된다고 가정합니다. 또한 이 계산에서는 두 노드가 동시에 중단될 수 있는 낮은 가능성을 무시합니다.

표 3–2 HA 중단 가능성

모델	서버 중단 시간 가능성
단일 서버(고가용성 아님)	Pr(중단) = (시스템 중단 4일 + 저장소 중단 1일)/365 = 1.37%
비대칭	Pr(중단) = (시스템 중단 0일 + 저장소 중단 1일)/365 = 0.27%
대칭	Pr(중단) = (시스템 중단 0일 + 저장소 중단 0일)/365 = (약 0)
N + 1 비대칭	Pr(중단) = (시스템 중단 5시간 + 저장소 중단 1일)/(365xN) = 0.27%/N

3.3 Messaging Server 고가용성 설치—개요

배포할 HA 모델을 선택한 후 Sun Cluster HA 또는 Veritas HA 중 하나를 선택합니다. 이 절에서는 예비 HA 배포 정보를 제공합니다. 이후의 절에서는 Sun Cluster 및 Veritas 고가용성 솔루션 관련 정보를 제공합니다.

3.3.1 클러스터 에이전트 설치

클러스터 에이전트는 클러스터 프레임워크에서 실행되는 Messaging Server 프로그램입니다.

Sun Cluster Messaging Server 에이전트(`SUNWscims`)는 Java Enterprise System 설치 프로그램을 통해 Sun Cluster를 선택한 경우에 설치됩니다. Veritas Cluster Messaging Server 에이전트(`SUNWmsgvc`)는 Sun Java Communications Suite CD의 Messaging Server `Product` 하위 디렉토리 `Solaris_sparc/Product/messaging_svr/Packages/SUNWmsgvc`에 있습니다. VCS 클러스터 에이전트를 설치하려면 `pkgadd`(1M) 명령을 사용해야 합니다.

3.3.2 Messaging Server 및 고가용성 지침

Messaging Server 및 고가용성(Veritas Cluster와 Sun Cluster 모두에 적용됨) 설치와 관련하여 몇 가지 알아두어야 할 사항이 있습니다.

- Messaging Server의 고가용성 기능은 기본적으로 설치되지 않습니다. Java Enterprise System 설치 프로그램의 사용자 정의 설치 메뉴에서 고가용성 구성 요소를 선택해야 합니다.
- 설치를 실행할 때 Messaging Server의 HA 논리 호스트 이름과 관련 IP 주소가 작동(예: 활성)하는지 확인합니다. 이렇게 하는 이유는 설치 과정에서 해당 이름과 주소를 사용하여 TCP 연결을 설정하기 때문입니다. Messaging Server의 HA 논리 호스트 이름이 현재 가리키고 있는 클러스터 노드에서 설치를 실행합니다.
- *msg_svr_base*는 공유 파일 시스템에 있어야 합니다. 그렇지 않으면 고가용성이 제대로 작동하지 않습니다. 예를 들어, 다른 노드에 페일오버한 이후에는 실패한 서버에 누적된 데이터가 서버에 더 이상 표시되지 않습니다.
- 초기 런타임 구성 중에 Messaging Server 호스트의 정규화된 이름을 묻는 메시지가 표시되면 Messaging Server에 대한 정규화된 HA 논리 호스트 이름을 지정해야 합니다. 설치 중에 이 논리 호스트 이름을 사용하여 TCP 연결을 시도합니다.
- `ha_ip_config`를 실행할 때 Messaging Server에 대한 IP 주소를 묻는 메시지가 표시되면 Messaging Server의 논리 호스트 이름에 연관된 IP 주소를 지정해야 합니다. 이 때, 물리적 호스트의 IP 주소를 사용하지 마십시오.
- Messaging Server의 현재 버전을 설치 및 구성하기 전에 클러스터링 소프트웨어를 설치해야 합니다. Messaging Server의 HA 논리 호스트 이름이 현재 가리키고 있는 클러스터 노드에서 설치를 실행합니다. 노드 이름을 묻는 메시지가 나타나면 클러스터 별칭을 사용합니다.

- Messaging Server 초기 런타임 구성(49 페이지 “1.3 Messaging Server 초기 런타임 구성 만들기 ” 참조)을 실행할 때 Messaging Server 클러스터의 정규화된 HA 논리 호스트 이름을 지정해야 합니다.
- 클러스터 호스트 이름을 사용하여 Messaging Server를 구성합니다. 그렇게 하지 않으면 클러스터 호스트 이름을 사용하여 또 다시 구성해야 합니다.

3.3.3 useconfig 유틸리티 사용

`useconfig` 유틸리티를 사용하면 단일 구성을 HA 환경의 여러 노드에서 공유할 수 있습니다. 이 유틸리티는 기존 구성을 업그레이드하거나 업데이트하는 용도로 사용할 수는 없습니다.

예를 들어 첫 번째 노드를 업그레이드하는 경우 Communications Suite 설치 프로그램을 통해 설치한 다음 Messaging Server를 구성합니다. 그런 다음 Communications Suite 설치 프로그램을 통해 Messaging Server 패키지를 설치할 두 번째 노드로 페일오버합니다. 하지만 초기 런타임 구성 프로그램(`configure`)을 다시 실행할 필요는 없습니다. 대신 `useconfig` 유틸리티를 사용할 수 있습니다.

유틸리티를 사용하려면 `useconfig`를 실행하여 이전 Messaging Server 구성을 가리키도록 합니다.

msg-svr-base`/sbin/useconfig install/configure_`*YYYYMMDDHHMMSS*

여기서 configure_*YYYYMMDDHHMMSS*는 이전 구성 설정 파일입니다.

새 노드의 경우 공유 디스크의 *msg-svr-base* `/data/setup` 디렉토리에서 `configure_`*YYYYMMDDHHMMSS*를 찾을 수 있습니다.

97 페이지 “3.5 Veritas Cluster Server 에이전트 설치” 및 76 페이지 “3.4 Sun Cluster 설치”에 대해 다음 절에서는 `useconfig` 유틸리티를 사용할 수 있는 시기에 대해 설명합니다.

3.4 Sun Cluster 설치

이 절에서는 Messaging Server를 Sun Cluster 고가용성(HA) 데이터 서비스로 설치 및 구성하는 방법에 대해 설명합니다. 다음 항목에 대해 설명합니다.

- 77 페이지 “3.4.1 Sun Cluster 요구 사항”
- 77 페이지 “3.4.2 HAStoragePlus 정보”
- 77 페이지 “3.4.3 Sun Cluster HAStorage 또는 HAStoragePlus를 사용하여 Messaging Server 구성 ”
- 94 페이지 “3.4.4 서버에서 IP 주소 바인딩”

Sun Cluster 설명서도 참조하십시오.

Veritas File System(VxFS)은 Sun Cluster 3.1에서 지원됩니다.

3.4.1 Sun Cluster 요구 사항

이 절에서는 다음을 가정합니다.

- Sun Cluster가 Solaris 운영 체제에 필수 패치와 함께 설치 및 구성되어 있습니다.
- Sun Cluster 에이전트 `SUNWscims`가 시스템에 설치되어 있습니다.
- 논리적 볼륨을 생성하는 경우 Solstice DiskSuite 또는 Veritas Volume Manager가 사용됩니다.

3.4.2 HAStoragePlus 정보

HAStoragePlus 자원 유형을 사용하여 Sun Cluster 환경에서 로컬 마운트 파일 시스템의 가용성을 높이는 것이 좋습니다. 페일오버 파일 시스템(FFS)이라고도 하는 로컬 파일 시스템은 전역 파일 시스템이라고도 하는 클러스터 파일 시스템(CFS)보다 더 높은 입력/출력 성능을 제공합니다. HAStoragePlus는 FFS와 CFS를 모두 지원합니다. 이와는 달리, HAStorage는 CFS만 지원합니다.

HAStoragePlus에는 다음과 같은 여러 장점이 있습니다.

- HAStoragePlus는 전역 파일 서비스 계층을 완벽히 우회합니다. 이런 특징은 디스크 입출력 사용량이 많은 데이터 서비스의 경우 성능을 크게 향상시킵니다.
- HAStoragePlus는 전역 파일 서비스 계층에서는 작동하지 않는 파일 시스템을 비롯하여, 모든 파일 시스템(UFS,VxFS 등)과 함께 사용할 수 있습니다. Solaris 운영 체제에서 지원되는 파일 시스템이라면 HAStoragePlus와 함께 사용할 수 있습니다.

데이터 서비스 자원 그룹에서 HAStorage 자원 또는 HAStoragePlus 자원 중 어느 것을 만들지 결정하려면 다음 기준을 고려하십시오.

- Sun Cluster 3.0 Release May 2002 또는 Sun Cluster 3.1을 사용 중인 경우 HAStoragePlus 사용
- Sun Cluster 3.0 December 2001 이전 버전을 사용 중인 경우 HAStorage 사용

HAStoragePlus에 대한 자세한 내용은 해당 Sun Cluster 문서(예: `http://docs.sun.com/app/docs/coll/573.10`)를 참조하십시오.

3.4.3 Sun Cluster HAStorage 또는 HAStoragePlus를 사용하여 Messaging Server 구성

이 절에서는 Sun Cluster용 Messaging Server에 대해 HAStorage 및 HAStoragePlus를 구성하는 방법에 대해 설명합니다. 첫 번째 절에서는 일반 단계에 대해 설명합니다. 이후의 절에서는 동기 및 비동기 배포에 고유한 예를 보여 줍니다.

HA를 구성한 뒤에는 94 페이지 “3.4.4 서버에서 IP 주소 바인딩”에서 HA 지원과 관련된 추가 단계를 검토하십시오.

다음 설명에서는 Messaging Server가 HA 논리 호스트 이름과 IP 주소로 구성되어 있다고 가정합니다. 물리적 호스트 이름은 HA 논리 호스트 이름인 `meadow`와 함께 `mars` 및 `venus`인 것으로 가정합니다. 그림 3–4는 Messaging Server HA 지원을 구성할 때 만들 다른 HA 자원의 중첩된 종속성을 나타냅니다.

주 – HAStorage 및 HAStoragePlus를 구성하는 방법을 설명하는 중 HAStoragePlus의 우수한 I/O 성능이 권장됩니다. 77 페이지 “3.4.2 HAStoragePlus 정보”를 참조하십시오.

이 절에는 다음과 같은 하위 절이 포함됩니다.

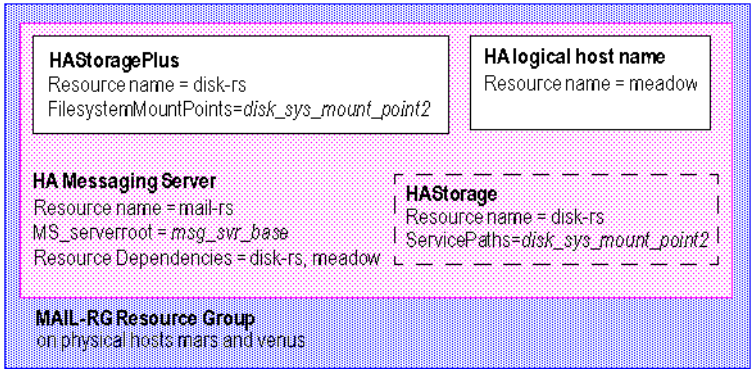


그림 3–4 간단한 Messaging ServerHA 구성

▼ Messaging Server를 Sun Cluster HAStorage 또는 HAStoragePlus와 함께 구성하는 방법—일반 예

이 절에서는 HA를 위해 Messaging Server를 구성하는 일반적인 단계에 대해 설명합니다. 이 단계를 검토한 후 다음 절의 해당 비대칭 또는 대칭 예를 참조하십시오. 이 지침에서는 물리적 호스트를 `mars` 및 `venus`라고 합니다. 논리적 호스트 이름은 `meadow`입니다.

그림 3–4는 Messaging Server HA 지원을 구성할 때 만들 다른 HA 자원의 중첩된 종속성을 나타냅니다.

1 **수퍼유저가 된 다음 콘솔을 엽니다.**

아래에 나오는 Sun Cluster 명령을 사용하려면 수퍼유저로 로그인해야 합니다. 또한 `/dev/console`에 출력되는 메시지를 보려면 콘솔이나 창이 필요합니다.

2 **모든 노드에서 필수 Messaging Sun Cluster Data Service Agents 패키지(SUNWscims)를 설치합니다.**

3 **클러스터의 각 노드에서 Messaging Server가 실행될 Messaging Server 런타임 사용자 및 그룹을 만듭니다.**

사용자 ID 및 그룹 ID 번호가 클러스터의 모든 노드에서 동일해야 합니다. **런타임 사용자** *ID*는 Messaging Server가 실행되는 사용자 이름입니다. 이 이름은 `root`가 아니어야 합니다. 기본값은 `mailsrv`입니다. **런타임 그룹** *ID*는 Messaging Server가 실행되는 그룹입니다. 기본값은 `mail`입니다.

`configure` 유틸리티에서 이러한 이름을 자동으로 만들 수 있지만, 이 장에서 설명한 것처럼 `configure`를 실행하기 전에 각 노드를 준비하는 과정에서 해당 이름을 만들 수도 있습니다. 런타임 사용자 및 그룹 ID 이름은

- `mailsrv`에 있어야 하고, 사용자가 선택한 이름은 클러스터의 모든 노드에서 `/etc/passwd`에 있어야 합니다.
- `mail`에 있어야 하고, 사용자가 선택한 이름은 클러스터의 모든 노드에서 `/etc/group`에 있어야 합니다.

47 페이지 “1.1 UNIX 시스템 사용자와 그룹 만들기”를 참조하십시오.

4 **필수 자원 유형을 Sun Cluster에 추가합니다.**

사용할 자원 유형을 Sun Cluster가 인식하도록 구성합니다. Messaging Server를 자원으로 등록하려면 다음 명령을 사용합니다.

```
# scrgadm -a -t SUNW.ims
```

HAStoragePlus를 자원 유형으로 등록하려면 다음 명령을 사용합니다.

```
# scrgadm -a -t SUNW.HAStoragePlus
```

HAStorage를 자원 유형으로 등록하려면 다음 명령을 사용합니다.

```
# scrgadm -a -t SUNW.HAStorage
```

5 **Messaging Server에 대한 페일오버 자원 그룹을 만듭니다.**

자원 그룹을 만들고 Messaging Server가 실행될 클러스터 노드에 표시되도록 만듭니다. 다음 명령은 `MAIL-RG`라는 자원 그룹을 만들고 클러스터 노드인 `mars`와 `venus`에 표시되도록 합니다.

```
# scrgadm -a -g MAIL-RG -h mars,venus
```

물론 자원 그룹에는 원하는 이름을 사용할 수 있습니다.

6 HA 논리 호스트 이름 자원을 만들어 온라인으로 전환합니다.

HA 논리 호스트 이름에 대한 자원을 만들고 활성화한 다음 자원 그룹에 추가합니다. 다음 명령은 논리 호스트 이름 meadow를 사용하여 이 작업을 수행합니다. -j 스위치를 생략했기 때문에 생성되는 자원의 이름 역시 meadow가 됩니다. meadow는 클라이언트가 자원 그룹의 서비스와 통신할 때 사용하는 논리 호스트 이름입니다.

```
# scrgadm -a -L -g MAIL-RG -l meadow
# scswitch -Z -g MAIL-RG
```

7 HAStorage 또는 HAStoragePlus 자원을 만듭니다.

그런 다음 Messaging Server가 종속된 파일 시스템에 대한 HA 저장소 또는 HAStoragePlus 자원 유형을 만들어야 합니다. 다음 명령은 disk-rs라는 이름의 HAStoragePlus 자원을 만들고 파일 시스템 *disk_sys_mount_point*는 이 자원의 제어를 받게 됩니다.

```
# scrgadm -a -j disk-rs -g MAIL-RG \
-t SUNW.HAStoragePlus \
-x FilesystemMountPoints=disk_sys_mount_point-1, disk_sys_mount_point-2 -x AffinityOn=True
```

SUNW.HAStoragePlus는 하나 이상의 데이터 서비스 자원에서 사용할 장치 그룹, 클러스터 및 로컬 파일 시스템을 나타냅니다. 한 자원이 SUNW.HAStoragePlus 유형의 자원을 자원 그룹에 추가하고 다른 자원과 SUNW.HAStoragePlus 자원 간의 종속성을 설정합니다. 이러한 종속성은 다음과 같은 경우에 데이터 서비스 자원을 온라인으로 전환합니다.

- 지정된 모든 장치 서비스가 사용 가능하고 배열(필요한 경우)된 경우
- 검사 후에 지정된 모든 파일 시스템이 마운트된 경우

FilesystemMountPoints 확장 등록 정보를 사용하여 전역 또는 로컬 파일 시스템을 지정할 수 있습니다. 즉, 모든 클러스터 노드 또는 단일 클러스터 노드에서 액세스할 수 있는 파일 시스템을 지정합니다. SUNW.HAStoragePlus 자원에서 관리되는 로컬 파일 시스템은 단일 클러스터 노드에 마운트되며 기본 장치가 Sun Cluster 전역 장치여야 합니다. 로컬 파일 시스템을 지정하는 SUNW.HAStoragePlus 자원은 선호도 전환이 활성화된 페일오버 자원 그룹에만 속할 수 있습니다. 따라서 이러한 로컬 파일 시스템을 페일오버 파일 시스템이라고 할 수 있습니다. 로컬 파일 시스템과 전역 파일 시스템의 마운트 지점을 모두 함께 지정할 수 있습니다.

/etc/vfstab 항목이 다음 조건을 모두 만족할 경우 마운트 지점이 FilesystemMountPoints 확장 등록 정보에 있는 파일 시스템이 로컬 파일 시스템으로 간주됩니다.

- 비전역 마운트 옵션
- 부트 시 마운트 플래그를 no로 설정합니다.

주 – SUNW.HAStoragePlus 자원 유형의 인스턴스는 전역 파일 시스템의 부트 시 마운트 플래그를 무시합니다.

HAStoragePlus 자원의 경우 FilesystemMountPoints의 쉼표로 구분된 목록은 Messaging Server가 종속되는 클러스터 파일 시스템(CFS) 또는 페일오버 파일 시스템(FFS)의 마운트 지점입니다. 위의 예에서는 두 개의 마운트 지점 *disk_sys_mount_point-1*과

*disk_sys_mount_point-2*가 지정되었습니다. 서버 중 하나에 해당 서버가 종속된 추가 파일 시스템이 있는 경우 추가 HA 저장소 자원을 만들고 단계 15에서 이 추가 종속성을 가리킵니다.

HAStorage의 경우 다음을 사용합니다.

```
# scrgadm -a -j disk-rs -g MAIL-RG \
-t SUNW.HAStorage
-x ServicePaths=disk_sys_mount_point-1, disk_sys_mount_point-2 -x AffinityOn=True
```

HAStorage 자원의 경우 `ServicePaths`의 쉼표로 분리된 목록은 Messaging Server가 종속된 클러스터 파일 시스템의 마운트 지점입니다. 위의 예에서는 두 개의 마운트 지점 *disk_sys_mount_point-1*과 *disk_sys_mount_point-2*가 지정되었습니다. 서버 중 하나에 해당 서버가 종속된 추가 파일 시스템이 있는 경우 추가 HA 저장소 자원을 만들고 단계 15에서 이 추가 종속성을 가리킵니다.

8 기본 노드에 필수 Messaging Server 패키지를 설치합니다. 나중에 구성 옵션을 선택합니다.

Communications Suite 설치 프로그램을 사용하여 Messaging Server 패키지를 설치합니다.

대칭 배포의 경우: Sun Cluster의 공유 디스크에 마운트된 파일 시스템에 Messaging Server 이진 및 구성 데이터를 설치합니다. 예를 들어, Messaging Server 이진은 `/disk_sys_mount_point-1/SUNWmsgsr`에 있고 구성 데이터는 `/disk_sys_mount_point-2/config`에 위치할 수 있습니다.

비대칭 배포의 경우: 로컬 파일 시스템의 Messaging Server 이진을 Sun Cluster의 각 노드에 설치합니다. 공유 디스크에 구성 데이터를 설치합니다. 예를 들어, 구성 데이터는 `/disk_sys_mount_point-2/config`에 위치할 수 있습니다.

9 Messaging Server를 구성합니다. 49 페이지 "1.3 Messaging Server 초기 런타임 구성 만들기"를 참조하십시오.

초기 런타임 구성에서 정규화된 호스트 이름을 묻는 메시지가 표시됩니다. 물리적 호스트 이름 대신 HA 논리 호스트 이름을 사용해야 합니다.

초기 런타임 구성 과정에서 49 페이지 "1.3 Messaging Server 초기 런타임 구성 만들기"에서 지정한 구성 디렉토리를 지정할 것을 요청합니다. HAStorage 또는 HAStoragePlus 자원의 공유 디스크 디렉토리 경로를 사용해야 합니다.

10 `ha_ip_config` 스크립트를 실행하여 `service.listenaddr`과 `service.http.smtphost`를 설정하고 `dispatcher.cnf`와 `job_controller.cnf` 파일을 고가용성에 대해 구성합니다.

스크립트를 사용하면 논리적 IP 주소가 물리적 IP 주소가 아니라 이 매개 변수와 파일에 대해 설정됩니다. 또한 watcher 프로세스가 활성화(`local.watcher.enable`을 1로 설정)되고 자동 재시작 프로세스가 활성화(`local.autorestart`를 1로 설정)됩니다.

스크립트 실행에 대한 자세한 내용은 94 페이지 "3.4.4 서버에서 IP 주소 바인딩"을 참조하십시오.

ha_ip_config 스크립트는 기본 노드에서만 실행해야 합니다.

11 imta.cnf 파일을 수정하고 모든 물리적 호스트 이름을 클러스터의 논리 호스트 이름으로 바꿉니다.

12 페일오버가 제대로 작동하는지 확인하기 위해 기본 클러스터 노드의 자원 그룹을 보조 클러스터 노드로 페일오버합니다.

자원 그룹을 다른 클러스터 노드로 수동으로 페일오버합니다. 페일오버를 수행할 노드에 대한 수퍼유저 권한이 있어야 합니다.

자원 그룹이 현재 어떤 노드에서 실행 중인지("online") 확인하려면 scstat 명령을 사용합니다. 예를 들어 자원 그룹이 mars에서 온라인인 경우 다음 명령을 사용하여 venus로 페일오버합니다.

```
# scswitch -z -g MAIL-RG -h venus
```

첫 번째 노드를 업그레이드하는 경우, Communications Suite 설치 프로그램을 통해 설치한 다음 Messaging Server를 구성합니다. 그런 다음 Communications Suite 설치 프로그램을 통해 Messaging Server 패키지를 설치할 두 번째 노드로 페일오버합니다. 하지만 초기 런타임 구성 프로그램(configure)을 다시 실행할 필요는 없습니다. 대신 useconfig 유틸리티를 사용할 수 있습니다.

13 보조 노드에 필수 Messaging Server 패키지를 설치합니다. 나중에 구성 옵션을 선택합니다.

두 번째 노드로 페일오버한 후 Communications Suite 설치 프로그램을 사용하여 Messaging Server 패키지를 설치합니다.

대칭 배포의 경우: Messaging Server를 설치하지 마십시오.

비대칭 배포의 경우: 로컬 파일 시스템의 Messaging Server 이진을 로컬 파일 시스템에 설치합니다.

14 클러스터의 두 번째 노드에서 useconfig를 실행합니다.

useconfig 유틸리티를 사용하면 단일 구성을 HA 환경의 여러 노드에서 공유할 수 있습니다. 초기 런타임 구성 프로그램(configure)을 실행할 필요는 없습니다. 대신 useconfig 유틸리티를 사용합니다(76 페이지 “3.3.3 useconfig 유틸리티 사용” 참조).

15 HA Messaging Server 자원을 만듭니다.

이제 HA Messaging Server 자원을 만들어 자원 그룹에 추가해야 합니다. 이 자원은 HA 논리 호스트 이름과 HA 디스크 자원에 종속됩니다.

HA Messaging Server 자원을 만들 때는 경로를 Messaging Server의 최상위 디렉토리인 *msg-svr-base* 경로로 지정해야 합니다. 이 작업은 다음 명령에 나와 있는 IMS_serverroot 확장 등록 정보를 사용하여 수행합니다.

```
# scrgadm -a -j mail-rs -t SUNW.ims -g MAIL-RG \
      -x IMS_serverroot=msg-svr-base \
```

```
-y Resource_dependencies=disk-rs,meadow
```

위의 명령은 *msg-svr-base* 디렉토리의 `IMS_serverroot`에 설치된 Messaging Server에 대해 `mail-rs`라는 이름의 HA Messaging Server 자원을 만듭니다. HA Messaging Server 자원은 HA 논리 호스트 이름 `meadow`와 HA 디스크 자원 `disk-rs`에 종속적입니다.

Messaging Server에 추가 파일 시스템 종속성이 있는 경우 이러한 파일 시스템에 대해 추가 HA 저장소 자원을 만들 수 있습니다. 추가 HA 저장소 자원 이름이 위 명령의 `Resource_dependencies` 옵션에 포함되도록 합니다.

16 Messaging Server 자원을 활성화합니다.

이제 HA Messaging Server 자원을 활성화하여 Messaging Server를 온라인으로 만듭니다. 이 작업을 수행하려면 다음 명령을 사용합니다.

```
# scswitch -e -j mail-rs
```

위의 명령은 MAIL-RG 자원 그룹의 `mail-rs` 자원을 활성화합니다. MAIL-RG 자원이 이미 온라인 상태가 되었기 때문에 위의 명령은 `mail-rs`도 온라인 상태로 만듭니다.

17 온라인 상태로 되었는지 확인합니다.

MAIL-RG 자원 그룹이 온라인인지 확인하려면 `scstat -pvv` 명령을 사용합니다.

또한 콘솔 장치에 표시된 출력에서 진단 정보를 살펴봅니다. `syslog` 파일, `/var/adm/messages`도 살펴봅니다. 자세한 디버깅 옵션과 정보는 93 페이지 “3.4.3.1 Sun Cluster에서 디버깅을 활성화하는 방법”을 참조하십시오.

▼ Sun Cluster 3.x에 대한 Messaging Server HA 지원 구성 해제 방법—일반 예

이 절에서는 Sun Cluster에 대한 HA 구성을 취소하는 방법에 대해 설명합니다. 간단한 구성 예(76 페이지 “3.4 Sun Cluster 설치”에서 설명)를 가정하여 설명합니다. 다른 구성에 대해서는 특정 명령(예: 단계 3)이 다를 수 있지만 논리적 순서는 같습니다.

1 수퍼유저가 됩니다.

다음 Sun Cluster 명령을 사용하려면 수퍼유저가 되어야 합니다.

2 자원 그룹을 오프라인 상태로 만듭니다.

자원 그룹의 모든 자원을 종료하려면 다음 명령을 실행합니다.

```
# scswitch -F -g MAIL-RG
```

이렇게 하면 자원 그룹 내의 모든 자원(예: Messaging Server 및 HA 논리 호스트 이름)이 종료됩니다.

3 **개별 자원을 비활성화합니다.**

그런 다음, 아래 명령을 사용하여 자원 그룹에서 자원을 하나씩 제거합니다.

```
# scswitch -n -j mail-rs
# scswitch -n -j disk-rs
# scswitch -n -j budgie
```

4 **자원 그룹에서 개별 자원을 제거합니다.**

자원이 비활성화되면 다음 명령으로 자원 그룹에서 자원을 하나씩 제거할 수 있습니다.

```
# scrgadm -r -j mail-rs
# scrgadm -r -j disk-rs
# scrgadm -r -j budgie
```

5 **자원 그룹을 제거합니다.**

자원 그룹에서 모든 자원이 제거되면 다음 명령으로 자원 그룹 자체를 제거할 수 있습니다.

```
# scrgadm -r -g MAIL-RG
```

6 **자원 유형을 제거합니다(선택 사항).**

클러스터에서 자원을 제거해야 하는 경우 다음 명령을 실행합니다.

```
# scrgadm -r -t SUNW.ims
# scrgadm -r -t SUNW.HAStoragePlus
```

▼ 2노드 대칭 Messaging Server를 구성하는 방법—예

이 예에서는 두 클러스터 노드의 물리적 호스트 이름이 `mars.red.siroe.com` 및 `venus.red.siroe.com`인 것으로 가정합니다. 설치 및 구성 디렉토리 위치는 고유해야 합니다. 각 노드의 설치 및 구성 디렉토리의 이름이 동일한 경우(예: `/opt/SUNWmsgsr` 및 `/var/opt/SUNWmsgsr`) 경합 문제가 발생할 수 있습니다. `venus`가 `mars`에 페일오버될 때 Messaging Server의 두 인스턴스가 동일한 설치 및 구성 디렉토리에서 완료될 경우 경합 문제가 발생합니다.

설치 디렉토리의 경우 `/opt/`*NodeMember*`/SUNWmsgsr` 형식을 사용하고 구성 디렉토리는 `/var/opt/`*NodeMember*`/SUNWmsgsr` 형식을 사용하여 설치 및 구성 디렉토리를 고유한 이름으로 지정하는 것이 좋습니다. 고유하기만 하면 어떤 디렉토리에도 이진 및 구성 데이터를 설치할 수 있습니다.

이 예에서는 두 클러스터 노드의 물리적 호스트 이름이 `mars.red.siroe.com` 및 `venus.red.siroe.com`인 것으로 가정합니다.

`mars.red.siroe.com`의 경우 이진은 `/opt/mars/SUNWmsgsr`에 설치되고 구성 데이터는 `/var/opt/mars/SUNWmsgsr`에 설치됩니다.

`venus.red.siroe.com`의 경우 이진은 `/opt/venus/SUNWmsgsr`에 설치되고 구성 데이터는 `/var/opt/venus/SUNWmsgsr`에 설치됩니다.

해당 논리 IP 주소를 가진 meadow 및 pasture라는 두 논리 호스트 이름이 있습니다. 예를 들어, 두 노드의 /etc/hosts 파일은 다음과 같습니다.

```
192.18.75.155 meadow.red.siroe.com meadow
192.18.75.157 pasture.red.siroe.com pasture
```

1 Messaging Server Sun Cluster 에이전트 패키지(SUNWscims)를 두 노드 모두에 설치합니다.

2 네 개의 파일 시스템을 만듭니다.

이러한 파일 시스템은 클러스터 파일 시스템이거나 로컬 파일 시스템(페일오버 파일 시스템)입니다.

```
/var/opt/mars/SUNWmsgsr
/var/opt/venus/SUNWmsgsr
/opt/mars/SUNWmsgsr
/opt/venus/SUNWmsgsr
```

이러한 파일 시스템을 공유 디스크에 마운트해야 합니다. 아래 예에서는 네 개의 클러스터 파일 시스템을 보여 줍니다. 아래 표시된 /etc/vfstab의 내용은 클러스터의 모든 노드에서 비슷해야 합니다.

```
# cat /etc/vfstab
#device device mount FS fsck mount mount to mount to fsck point type
pass at_boot options
/dev/md/penguin/dsk/d500 /dev/md/penguin/rdsk/d500 /opt/mars/SUNWmsgsr ufs 2 yes
logging,global
/dev/md/penguin/dsk/d400 /dev/md/penguin/rdsk/d400 /var/opt/mars/SUNWmsgsr ufs 2
yes logging,global
/dev/md/polarbear/dsk/d200 /dev/md/polarbear/rdsk/d200 /opt/venus/SUNWmsgsr ufs 2
yes  logging,global
/dev/md/polarbear/dsk/d300 /dev/md/polarbear/rdsk/d300 /var/opt/venus/SUNWmsgsr
ufs 2 yes logging,global
```

위에 표시된 네 개의 파일 시스템을 로컬 파일 시스템(페일오버 파일 시스템)으로 만들려면 부트 시 마운트 옵션을 no로 설정하고 마운트 옵션의 global 키워드를 제거합니다.

3 기본 노드 구성

a. 기본 노드에서 필수 자원 유형을 추가합니다.

그러면 사용할 자원 유형을 Sun Cluster가 인식하도록 구성됩니다. Messaging Server 및 HAStoragePlus 자원을 등록하려면 다음 명령을 사용합니다.

```
# scrgadm -a -t SUNW.HAStoragePlus
# scrgadm -a -t SUNW.ims
```

b. Messaging Server에 대해 MS_RG_MARS라는 페일오버 자원 그룹을 만듭니다.

```
# scrgadm -a -g MS_RG_MARS -h mars,venus
```

c. meadow라는 논리 호스트 이름 자원을 만든 후 자원 그룹에 추가하여 온라인으로 전환합니다.

```
# scrgadm -a -L -g MS_RG_MARS -l meadow
# scrgadm -c -j meadow -y R_description="LogicalHostname resource for meadow"
# scswitch -Z -g MS_RG_MARS
```

d. 앞에서 만든 파일 시스템을 사용하여 ms-hasp-mars라는 HAStoragePlus 자원을 만듭니다.

```
# scrgadm -a -j ms-hasp-mars -g MS_RG_MARS -t SUNW.HAStoragePlus -x
FileSystemMountPoints ="/opt/mars/SUNWmsgsr, /var/opt/mars/SUNWmsgsr" -x
AffinityOn=TRUE
```

e. HAStoragePlus 자원 활성화:

```
# scswitch -e -j ms-hasp-mars
```

4 Messaging Server를 기본 노드에 설치합니다.

Communications Suite 설치 프로그램을 사용하여 Messaging Server 패키지를 설치합니다. Messaging Server 이진 및 구성 데이터를 공유 파일 시스템에 설치하는지 확인합니다(단계 2 참조). 예를 들어, 이 Messaging Server 인스턴스의 경우 메시징 이진은 /opt/mars/SUNWmsgsr에 있고 구성 데이터는 /var/opt/mars/SUNWmsgsr에 있습니다.

5 Messaging Server를 기본 노드에서 설치하고 구성합니다(49 페이지 "1.3 Messaging Server 초기 런타임 구성 만들기" 참조).

초기 런타임 구성 프로그램에 정규화된 호스트 이름을 묻는 메시지가 표시됩니다. 논리 호스트 이름 meadow.red.siroe.com을 입력합니다. 또한 구성 디렉토리를 지정하라는 메시지가 표시됩니다. /var/opt/mars/SUNWmsgsr을 입력합니다.

6 기본 노드에서 ha_ip_config 스크립트를 실행하고 논리 IP 주소를 제공합니다.

이 스크립트는 기본 노드에서만 실행되고 보조 노드에서는 실행되지 않습니다. ha_ip_config 스크립트는 sbin 디렉토리 아래의 설치 디렉토리에 있습니다. 예를 들면 다음과 같습니다.

```
# /opt/mars/SUNWmsgsr/sbin/ha_ip_config

Please specify the IP address assigned to the HA logical host name.
Use dotted decimal form, a.b.c.d

Logical IP address: 192.18.75.155
# This value is the logical IP address of the logical hostname. Refer
# to the /etc/hosts file.
Please specify the path to the top level directory in which iMS is
installed.

iMS server root: /opt/mars/SUNWmsgsr
```

```
. . .

Updating the file /opt/mars/SUNWmsgsr/config/dispatcher.cnf
Updating the file /opt/mars/SUNWmsgsr/config/job_controller.cnf
Setting the service.listenaddr configutil parameter
Setting the local.snmp.listenaddr configutil parameter
Setting the service.http.smtphost configutil parameter
Setting the local.watcher.enable configutil parameter
Setting the local.autorestart configutil parameter
Setting the metermaid.config.bindaddr configutil parameters
Setting the metermaid.config.serveraddr configutil parameters
Setting the local.ens.port parameter
Configuration successfully updated
```

7 **imta.cnf 파일을 수정하고 모든 물리적 호스트 이름 항목(즉, mars)을 HA 논리 호스트 이름(meadow)으로 바꿉니다.**

8 **자원 그룹을 보조 노드(venus)에 페일오버합니다.**

페일오버 후 보조 노드(venus)를 구성합니다.

```
# scswitch -z -g MS_RG_VENUS -h mars
```

9 **보조 노드(venus)에서 useconfig 유틸리티를 실행합니다.** **76 페이지 "3.3.3 useconfig 유틸리티 사용"을 참조하십시오.**

초기 런타임 구성 프로그램(configure)을 실행하거나 Messaging Server 패키지를 설치할 필요는 없습니다.

다음 예에서 /var/opt/mars/SUNWmsgsr은 공유 구성 디렉토리입니다.

```
# useconfig /var/opt/mars/SUNWmsgsr/setup/configure_20061201124116
cp /var/opt/mars/SUNWmsgsr/setup/configure_20061201124116/Devsetup.properties
/opt/mars/SUNWmsgsr/lib/config-templates/Devsetup.properties
/usr/sbin/groupadd mail
/usr/sbin/useradd -g mail -d / mailsrv
/usr/sbin/usermod -G mail mailsrv
sed -e "s/local.serveruid/mailsrv/" -e "s/local.serveruid/mail/" -e "s:<msg·RootPath>:/opt/mars/SUNWmsgsr:"
/opt/mars/SUNWmsgsr/lib/config-templates/devtypes.txt.template >
/opt/mars/SUNWmsgsr/lib/config-templates/devtypes.txt
sed -e "s/local.serveruid/mailsrv/" -e "s/local.serveruid/mail/" -e
"s:<msg·RootPath>:/opt/mars/SUNWmsgsr:"
/opt/mars/SUNWmsgsr/lib/config-templates/config.ins.template >
/opt/mars/SUNWmsgsr/lib/config-templates/config.ins
/opt/mars/SUNWmsgsr/lib/devinstall -l sepadmsvr:pkgcfg:config -v -m -i
/opt/mars/SUNWmsgsr/lib/config-templates/config.ins
/opt/mars/SUNWmsgsr/lib/config-templates
/opt/mars/SUNWmsgsr/lib/jars /opt/mars/SUNWmsgsr/lib
devinstall returned 0
crle -c /var/ld/ld.config -s
 /usr/lib/secure:/opt/SUNWmsgsr/lib:/opt/venus/SUNWmsgsr/lib:/opt/mars/SUNWmsgsr/lib
```

```
-s /opt/mars/SUNWmsgsr/lib
See /opt/mars/SUNWmsgsr/install/useconfiglog_20061211155037 for more details
```

10 HA Messaging Server 자원을 만든 후 활성화합니다.

```
# scrgadm -a -j ms-rs-mars -t SUNW.ims -g MS_RG_MARS -x IMS_serverroot
=/opt/mars/SUNWmsgsr -y Resource_dependencies=meadow,ms-hasp-mars
# scswitch -e -j mail-rs-mars
```

위 명령은 Messaging Server에 대해 `ms-rs-mars`라는 HA Messaging Server 자원을 만듭니다. 이 HA Messaging Server 자원은 `/opt/mars/SUNWmsgsr`에 설치되고 HA 디스크 자원(앞에서 만든 파일 시스템)과 HA 논리 호스트 이름 `meadow`에 종속됩니다.

11 모든 항목이 제대로 작동하는지 확인합니다.

Messaging Server 자원을 기본 노드로 다시 페일오버합니다.

```
 # scswitch -z -g MAIL-RG -h mars
```

12 마찬가지로 Messaging Server의 두 번째 인스턴스에 대해 `venus`를 기본 노드로, `mars`를 보조(또는 대기) 노드로 사용하는 다른 페일오버 자원 그룹을 만듭니다.

이 자원 그룹에 대해 `venus`를 기본 노드로, `MS_RG_VENUS`를 자원 그룹으로, `pasture`를 논리 호스트 이름으로, `ms-hasp-venus`를 HAStoragePlus 자원으로 사용하여 3-10단계를 반복합니다. 명령은 다음과 같습니다.

자원 그룹 `MS_RG_VENUS` 만들기

```
# scrgadm -a -g MS_RG_VENUS -h venus,mars
```

`pasture`라는 논리 호스트 이름 자원을 만들어 자원 그룹에 추가한 다음 온라인으로 전환

```
# scrgadm -a -L -g MS_RG_VENUS -l pasture
# scrgadm -c -j pasture -y R_description="LogicalHostname resource for pasture"
# scswitch -Z -g MS_RG_VENUS
```

앞에서 만든 파일 시스템을 사용하여 `ms-hasp-venus`라는 HAStoragePlus 자원 만들기

```
# scrgadm -a -j ms-hasp-venus -g MS_RG_VENUS -t SUNW.HAStoragePlus -x
FileSystemMountPoints ="/opt/venus/SUNWmsgsr, /var/opt/venus/SUNWmsgsr" -x
AffinityOn=TRUE
```

HAStoragePlus 자원 활성화

```
# scswitch -e -j ms-hasp-venus
```

기본 노드에서 `ha_ip_config` 스크립트를 실행하고 논리 IP 주소 제공

```
# /opt/venus/SUNWmsgsr/sbin/ha_ip_config
```

HA Messaging Server 자원을 만든 후 활성화

```
# scrgadm -a -j ms-rs-venus -t SUNW.ims -g MS_RG_VENUS -x IMS_serverroot
=/opt/venus/SUNWmsgsr -y Resource_dependencies=pasture,ms-hasp-venus
# scswitch -e -j mail-rs-venus
```

자원 그룹을 보조 노드(venus)에 페일오버

```
# scswitch -z -g MS_RG_MARS -h venus
```

useconfig를 보조 노드(mars)에서 실행하려면 useconfig 유틸리티 실행

```
# useconfig /var/opt/venus/SUNWmsgsr/setup/configure_20061201124116
```

Messaging Server 자원을 기본 노드로 다시 페일오버하여 모든 사항이 제대로 작동하는지 확인

```
 # scswitch -z -g MAIL-RG -h venus
```

▼ HA 대칭 배포 구성 해제

구성 해제는 Messaging Server 또는 Sun Cluster를 업그레이드해야 하거나 Messaging Server를 제거해야 하는 경우에 수행됩니다. 여기서는 시스템이 이전 예를 사용하여 구성된 것으로 가정합니다.

첫 번째 단계에서는 클러스터에서 각 자원 그룹을 제거합니다. 이 예에는 MS_RG_MARS 및 MS_RG_VENUS의 두 자원 그룹이 있습니다. 두 그룹 모두 제거해야 합니다.

1 자원 그룹 MS_RG_MARS를 클러스터에서 제거합니다.

한 노드에서만 다음 명령을 사용합니다. 이 명령을 각 노드에서 실행할 필요는 없습니다.

a. 모든 클러스터 노드에서 자원 그룹을 오프라인으로 전환합니다.

```
# scswitch -F -g MS_RG_MARS
```

b. 해당하는 모든 Messaging Server 자원을 비활성화합니다.

```
# scswitch -n -j ms-rs-mars
# scswitch -n -j meadow
# scswitch -n -j ms-hasp-mars
```

c. 해당하는 모든 MS 자원을 제거합니다.

```
# scrgadm -r -j ms-rs-mars
# scrgadm -r -j meadow
# scrgadm -r -j ms-hasp-mars
```

d. 자원 그룹을 제거합니다.

```
scrgadm -r -g MS_RG_MARS
```

2 자원 그룹 MS_RG_VENUS를 클러스터에서 제거합니다.

한 노드에서만 다음 명령을 사용합니다. 이 명령을 각 노드에서 실행할 필요는 없습니다.

a. 모든 클러스터 노드에서 자원 그룹을 오프라인으로 전환합니다.

```
# scswitch -F -g MS_RG_VENUS
```

b. 해당하는 모든 Messaging Server 자원을 비활성화합니다.

```
# scswitch -n -j ms-rs-venus
# scswitch -n -j pasture
# scswitch -n -j ms-hasp-venus
```

c. 해당하는 모든 MS 자원을 제거합니다.

```
# scrgadm -r -j ms-rs-venus
# scrgadm -r -j pasture
# scrgadm -r -j ms-hasp-venus
```

d. 자원 그룹을 제거합니다.

```
scrgadm -r -g MS_RG_VENUS
```

3 사용하지 않는 자원 유형을 등록 취소합니다.

```
# scrgadm -r -t SUNW.HAStoragePlus
# scrgadm -r -t SUNW.ims
```

▼ 2노드 HA 비대칭 Messaging Server를 구성하는 방법—예

이 예에서는 물리적 호스트 이름이 daisy.red.siroe.com 및 lavender.red.siroe.com이고 논리 호스트 이름이 budgie인 두 개의 노드 클러스터가 있다고 가정합니다.

daisy.red.siroe.com의 경우 이진은 /opt/SUNWmsgsr에 설치되고 구성 데이터는 /var/opt/SUNWmsgsr에 설치됩니다.

논리 호스트 이름 budgie에 논리 IP 주소가 할당됩니다. 예를 들어, /etc/hosts 파일은 다음과 같습니다.

```
192.18.75.157 budgie.red.siroe.com budgie
```

1 Messaging Sun Cluster 에이전트(SUNWscims)를 두 노드 모두에 설치합니다.

2 파일 시스템을 만듭니다.

이 예에서는 /var/opt/SUNWmsgsr 파일 시스템이 공유 디스크에 마운트됩니다. 이 파일 시스템은 클러스터 파일 시스템이거나 로컬 파일 시스템(페일오버 파일 시스템)입니다.

3 기본 노드(daisy)를 구성합니다.

a. 기본 노드에서 필수 자원 유형을 추가합니다.

그러면 사용할 자원 유형을 Sun Cluster가 인식하도록 구성됩니다. Messaging Server 및 HAStoragePlus 자원을 등록하려면 다음 명령을 사용합니다.

```
# scrgadm -a -t SUNW.HAStoragePlus
# scrgadm -a -t SUNW.ims
```

b. Messaging Server 인스턴스에 대해 MS_RG_DAISY라는 자원 그룹을 만듭니다.

```
# scrgadm -a -g MS_RG_daisy -h daisy,lavender
```

c. meadow라는 논리 호스트 이름 자원을 만든 후 자원 그룹에 추가하여 온라인으로 전환합니다.

```
# scrgadm -a -L -g MS_RG_DAISY -l meadow
# scrgadm -c -j meadow -y R_description="LogicalHostname resource for meadow"
# scswitch -Z -g MS_RG_DAISY
```

d. 앞에서 만든 파일 시스템을 사용하여 ms-hasp-daisy라는 HAStoragePlus 자원을 만듭니다.

```
# scrgadm -a -j ms-hasp-daisy -g MS_RG_DAISY -t SUNW.HAStoragePlus -x
FileSystemMountPoints ="/var/opt/SUNWmsgsr" -x
AffinityOn=TRUE
```

e. HAStoragePlus 자원 활성화:

```
# scswitch -e -j ms-hasp-daisy
```

4 Messaging Server를 기본 노드에서 설치하고 구성합니다(49 페이지 "1.3 Messaging Server 초기 런타임 구성 만들기 " 참조).

초기 런타임 구성 프로그램에 정규화된 호스트 이름을 묻는 메시지가 표시됩니다. 논리 호스트 이름 meadow.red.siroe.com을 입력합니다. 또한 구성 디렉토리를 지정하라는 메시지가 표시됩니다. /var/opt/SUNWmsgsr을 입력합니다.

5 기본 노드에서 ha_ip_config 스크립트를 실행하고 논리 IP 주소를 제공합니다.

이 스크립트는 기본 노드에서만 실행되고 보조 노드에서는 실행되지 않습니다. ha_ip_config 스크립트는 sbin 디렉토리 아래의 설치 디렉토리에 있습니다. 예를 들면 다음과 같습니다.

```
# /opt/SUNWmsgsr/sbin/ha_ip_config

Please specify the IP address assigned to the HA logical host name.
Use dotted decimal form, a.b.c.d

Logical IP address: 192.18.75.155
# This value is the logical IP address of the logical hostname. Refer
# to the /etc/hosts file.
```

```
Please specify the path to the top level directory in which iMS is
installed.

iMS server root: /opt/SUNWmsgsr

. . .

Updating the file /opt/SUNWmsgsr/config/dispatcher.cnf
Updating the file /opt/SUNWmsgsr/config/job_controller.cnf
Setting the service.listenaddr configutil parameter
Setting the local.snmp.listenaddr configutil parameter
Setting the service.http.smtphost configutil parameter
Setting the local.watcher.enable configutil parameter
Setting the local.autorestart configutil parameter
Setting the metermaid.config.bindaddr configutil parameters
Setting the metermaid.config.serveraddr configutil parameters
Setting the local.ens.port parameter
Configuration successfully updated
```

6 **imta.cnf 파일을 수정하고 모든 물리적 호스트 이름 항목(daisy)을 HA 논리 호스트 이름(meadow)으로 바꿉니다.**

7 **자원 그룹을 보조 노드(lavender)에 페일오버합니다.**

페일오버 후 보조 노드(lavender)를 구성합니다.

```
# scswitch -z -g MS_RG_LAVENDER -h daisy
```

8 **보조 노드(lavender)에서 Messaging Server를 설치하고 useconfig 유틸리티를 실행합니다.** **76 페이지 "3.3.3 useconfig 유틸리티 사용"을 참조하십시오.**

초기 런타임 구성 프로그램(configure)을 실행할 필요는 없습니다.

다음 예에서 /var/opt/SUNWmsgsr은 공유 구성 디렉토리입니다.

```
# useconfig /var/opt/SUNWmsgsr/setup/configure_20061201124116
cp /var/opt/SUNWmsgsr/setup/configure_20061201124116/Devsetup.properties
/opt/SUNWmsgsr/lib/config-templates/Devsetup.properties
/usr/sbin/groupadd mail
/usr/sbin/useradd -g mail -d / mailsrv
/usr/sbin/usermod -G mail mailsrv
sed -e "s/local.serveruid/mailsrv/" -e "s/local.serveruid/mail/" -e "s:<msg·RootPath>:/opt/SUNWmsgsr:"
/opt/SUNWmsgsr/lib/config-templates/devtypes.txt.template >
/opt/SUNWmsgsr/lib/config-templates/devtypes.txt
sed -e "s/local.serveruid/mailsrv/" -e "s/local.serveruid/mail/" -e
"s:<msg·RootPath>:/opt/SUNWmsgsr:"
/opt//SUNWmsgsr/lib/config-templates/config.ins.template >
/opt/SUNWmsgsr/lib/config-templates/config.ins
/opt/SUNWmsgsr/lib/devinstall -l sepadmsvr:pkgcfg:config -v -m -i
/opt/SUNWmsgsr/lib/config-templates/config.ins
/opt/SUNWmsgsr/lib/config-templates
```

```
/opt/SUNWmsgsr/lib/jars /opt/SUNWmsgsr/lib
devinstall returned 0
crle -c /var/ld/ld.config -s
 /usr/lib/secure:/opt/SUNWmsgsr/lib:/opt/SUNWmsgsr/lib:/opt/SUNWmsgsr/lib
-s /opt/SUNWmsgsr/lib
See /opt/SUNWmsgsr/install/useconfiglog_20061211155037 for more details
```

9 HA Messaging Server 자원을 만든 후 활성화합니다.

```
# scrgadm -a -j ms-rs-daisy -t SUNW.ims -g MS_RG_DAISY -x IMS_serverroot
=/opt/SUNWmsgsr -y Resource_dependencies=meadow,ms-hasp-daisy
# scswitch -e -j mail-rs-daisy
```

위 명령은 Messaging Server에 대해 `ms-rs-daisy`라는 HA Messaging Server 자원을 만듭니다. 이 HAMessaging Server 자원은 `/opt/SUNWmsgsr`에 설치되고, HA 디스크 자원(앞에서 만든 파일 시스템)과 HA 논리 호스트 이름 `meadow`에 종속됩니다.

10 모든 항목이 제대로 작동하는지 확인합니다.

Messaging Server 자원을 기본 노드로 다시 페일오버합니다.

```
 # scswitch -z -g MAIL-RG -h daisy
```

3.4.3.1 Sun Cluster에서 디버깅을 활성화하는 방법

Messaging Server Data Service Sun Cluster 에이전트는 두 API를 사용하여 디버그 메시지를 기록합니다.

`scds_syslog_debug()`는 수준 1 시스템 로그에 디버깅 메시지를 기록합니다.

`scds_syslog()`는 `daemon.notice`, `daemon.info` 및 `daemon.error` 수준 시스템 로그에 메시지를 기록합니다.

모든 `syslog` 메시지에는 접두어가 다음과 같이 지정됩니다.

`SC[`*resourceTypeName*, *resourceGroupName*, *resourceName*,*methodName]*

예를 들면 다음과 같습니다.

```
Dec 11 18:24:46 mars SC[SUNW.ims,MS-RG,mail-rs,ims_svc_start]: [ID 831728daemon.debug]
Groupname mail exists.
Dec 11 18:24:46 mars SC[SUNW.ims,MS-RG,mail-rs,ims_svc_start]: [ID 383726daemon.debug]
Username mailsrv exists.
Dec 11 18:24:46 mars SC[SUNW.ims,MS-RG,mail-rs,ims_svc_start]: [ID 244341daemon.debug]
IMS_serverroot = /opt/mars/SUNWmsgsr
Dec 11 15:55:52 mars SC[SUNW.ims,MS_RG,MessagingResource,ims_svc_validate]:
[ID 855581daemon.error] Failed to get the configuration info
Dec 11 18:24:46 mars SC[SUNW.ims,MS-RG,mail-rs,ims_svc_start]: [ID 833212daemon.info]
Attempting to start the data service under process monitor facility.
```

Messaging Server 자원 유형 SUNW.ims의 메시지를 기록하려면 아래와 같이 /var/cluster 아래에 자원 유형 디렉토리를 만듭니다.

```
mkdir -p /var/cluster/rgm/rt/SUNW.ims
```

자원 유형 SUNW.ims에 대한 모든 디버깅 메시지를 보려면 클러스터의 모든 노드에서 다음 명령을 실행합니다.

```
echo 9 > /var/cluster/rgm/rt/SUNW.ims/loglevel
```

자원 유형 SUNW.iws에 대한 모든 디버깅 메시지를 억제하려면 클러스터의 모든 노드에서 다음 명령을 실행합니다.

```
echo 0 > /var/cluster/rgm/rt/SUNW.ims/loglevel
```

Sun Cluster 데이터 서비스의 디버그 메시지와 Messaging Server 에이전트의 가장 일반적인 디버깅 정보를 기록하려면 syslog.conf 파일을 편집합니다. 예를 들어, 모든 syslog 메시지를 /var/adm/clusterlog 파일에 기록하려면 syslog.conf 파일에 다음 행을 추가합니다.

```
daemon.debug /var/adm/clusterlog
```

그러면 모든 메시지가 다음 수준(emerg, alert, critical, error, warning, notice, information, debug)으로 기록됩니다. 자세한 내용은 syslog.conf 설명서 페이지를 참조하십시오.

이제 syslogd 데몬을 다시 시작합니다.

```
pkill -HUP syslogd
```

3.4.4 서버에서 IP 주소 바인딩

대칭형 또는 N + 1 고가용성 모델을 사용하는 경우 Sun Cluster Server를 Messaging Server와 함께 사용하기 위해 준비하려면 구성 과정에서 알아야 할 몇 가지 사항이 있습니다.

서버에서 실행 중인 Messaging Server에 올바른 IP 주소를 바인딩해야 합니다. 이는 HA 환경에서 Messaging Server를 제대로 구성하기 위해 필요합니다.

HA를 위해 Messaging Server를 구성하려면 Messaging Server가 바인딩하고 연결을 수신하는 인터페이스 주소를 구성해야 합니다. 기본적으로 서버는 사용 가능한 모든 인터페이스 주소에 바인딩합니다. 하지만 HA 환경에서는 서버가 HA 논리 호스트 이름과 연관된 인터페이스 주소에 바인딩되도록 해야 합니다.

따라서 특정 Messaging Server 인스턴스에 속한 서버가 사용하는 인터페이스 주소를 구성하기 위한 스크립트가 제공됩니다. 이 스크립트는 현재 IP 주소 또는 서버가 사용하는 HA 논리 호스트 이름과 연관될 IP 주소를 통해 인터페이스 주소를 식별합니다.

스크립트는 다음 구성 파일을 수정하거나 만들어서 구성을 변경합니다. 다음 파일의 경우

msg-svr-base/config/dispatcher.cnf

SMTP 및 SMTP Submit 서버의 INTERFACE_ADDRESS 옵션을 추가하거나 변경합니다. 다음 파일의 경우

msg-svr-base/config/job_controller.cnf

작업 제어기의 INTERFACE_ADDRESS 옵션을 추가하거나 변경합니다.

마지막으로 POP, IMAP 및 Messenger Express HTTP 서버에서 사용하는 configutil service.listenaddr 및 service.http.smtphost 매개 변수를 설정합니다.

원본 구성 파일이 있는 경우 해당 파일의 이름은 *.pre-ha로 변경됩니다.

스크립트를 다음과 같이 실행합니다.

▼ 서버에서 IP 주소 바인딩 방법

1. **수퍼유저가 됩니다.**
2. ***msg-svr-base*/sbin/ha_ip_config를 실행합니다.**
3. **스크립트에서는 아래와 같은 질문을 합니다. 질문에 대해 control-d를 입력하여 응답하면 스크립트 실행이 중단될 수 있습니다. 질문에 대한 기본 답변은 대괄호 [] 안에 나타납니다. 기본 답변을 사용하려면 Enter 키를 누릅니다.**

   a. **논리 IP 주소: Messaging Server에서 논리 호스트 이름에 할당되는 IP 주소를 지정합니다. IP 주소는 123.456.78.90과 같이 점으로 구분된 십진수 형식으로 지정해야 합니다.**

      논리적 IP 주소는 configutil 매개 변수 service.http.smtphost에서 자동으로 설정되며 클러스터의 메시징 시스템이 현재 어떤 시스템에서 실행 중인지 확인할 수 있게 해줍니다. 예를 들어 Messenger Express를 사용하는 경우 서버는 보내는 메일을 전송할 메일 호스트를 결정할 수 있습니다.

   b. **Messaging Server 기본(*msg-svr-base*): Messaging Server가 설치되는 최상위 디렉토리의 절대 경로를 지정합니다.**

   c. **위에서 선택한 사항중 일부를 변경하시겠습니까? 답변을 그대로 적용하고 구성을 변경하려면 "no"로 답합니다. 답변을 변경하려면 "yes"로 답합니다.**

주 – 또한 `ha_ip_config` 스크립트는 두 개의 새 프로세스인 `watcher`와 `msprobe`, 그리고 매개 변수 `local.autorestart`와 `local.watcher.enable`을 자동으로 활성화합니다. 이 새 매개 변수는 Messaging Server의 상태를 모니터하는 데 도움이 됩니다. 프로세스가 실패하거나 서비스가 응답하지 않으면 해당 오류를 나타내는 로그 메시지가 생성됩니다. 이제 클러스터 에이전트는 종료될 때마다 `watcher` 프로세스와 페일오버를 모니터합니다. Sun Cluster가 제대로 작동하려면 매개 변수를 활성화해야 합니다.

`watcher` 및 `msprobe` 프로세스에 대한 자세한 내용은 107 페이지 "4.5 실패했거나 응답이 없는 서비스의 자동 재시작"을 참조하십시오.

3.4.5 Messaging HA를 관리하는 데 유용한 Sun Cluster 명령

Messaging Server 자원 활성화

```
# scswitch -e -j messaging-resource
```

Messaging Server 자원 비활성화

```
# scswitch -n -j cal-resource
```

모든 자원과 자원 그룹 나열

```
# scstat -pvv
```

PMF에 의해 모니터링되는 프로세스인 PMF(Process Monitoring Facility) 태그 결정

```
# pmfadm -L
```

모든 자원 및 자원 그룹과 해당 상태 나열

```
# scstat -g
```

Sun Cluster 관리

```
scsetup
```

3.5 Veritas Cluster Server 에이전트 설치

Messaging Server는 Veritas Cluster Server 3.5, 4.0, 4.1 및 5.0과 함께 작동하도록 구성할 수 있습니다.

이 절차를 따르기 전에 Veritas Cluster Server 설명서를 검토하시기 바랍니다.

Communications Suite 설치 프로그램을 사용하여 Messaging Server를 설치하고 HA를 구성한 후에는 94 페이지 "3.4.4 서버에서 IP 주소 바인딩"을 검토하여 HA 지원의 구성과 관련된 추가 단계가 있는지 확인하십시오. 이 절에는 다음과 같은 하위 절이 포함됩니다.

3.5.1 Veritas Cluster Server 요구 사항

- 다음 지침(97 페이지 "3.5.2 VCS 설치 및 구성 지침")에서 설명한 대로 양쪽 노드에 Veritas Cluster 소프트웨어가 Messaging Server 소프트웨어와 함께 이미 설치 및 구성되어 있어야 합니다.

3.5.2 VCS 설치 및 구성 지침

다음 지침에서는 Veritas Cluster Server를 사용하여 Messaging Server를 HA 서비스로 구성하는 방법에 대해 설명합니다.

기본 `main.cf` 구성 파일은 `VCSweb` 응용 프로그램을 실행하는 `ClusterService`라는 자원 그룹을 설정합니다. 이 그룹에는 `csgnic` 및 `webip`와 같은 네트워크의 논리 호스트 IP 자원이 포함됩니다. 또한 이벤트 알림을 위한 `ntfr` 자원이 생성됩니다.

▼ Veritas Cluster Server를 사용하여 Messaging Server를 HA 서비스로 구성하는 방법

1 노드 중 하나에서 Cluster Explorer를 시작합니다.

이 Veritas Cluster Server 지침에서는 그래픽 사용자 인터페이스를 사용하여 Messaging Server를 HA 서비스로 구성한다고 가정합니다.

Cluster Explorer를 시작하려면 다음 명령을 실행합니다.

```
# /opt/VRTSvcs/bin/hagui
```

GUI를 사용하려면 `VRTScscm` 패키지가 설치되어 있어야 합니다.

2 Cluster Explorer를 사용하여 `MAIL-RG`라는 서비스 그룹을 추가합니다.

3 DiskGroup 유형의 s1ms_dg 디스크 그룹 자원을 MAIL-RG 서비스 그룹에 추가하고 활성화합니다.

4 Mount 유형의 s1ms_mt 마운트 자원을 MAIL-RG 서비스 그룹에 추가합니다.

a. 아직 활성화되지 않은 경우 링크 버튼을 눌러 자원 링크를 활성화합니다.

5 s1ms_mt와 s1ms_dg 사이에 링크를 만듭니다. s1ms_mt 자원을 활성화합니다.
그림에서는 종속성 트리를 설명합니다.

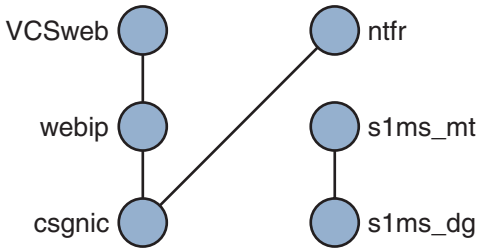


그림 3–5 Veritas Cluster Server 종속성 트리 1

6 Communications Suite 설치 프로그램을 실행하여 Messaging Server를 설치합니다.

a. 기본 노드(예: Node_A)에서 Messaging Server 초기 런타임 구성을 실행하여 Messaging Server를 설치합니다.

b. pkgadd(1M) 명령을 사용하여 Veritas Cluster Server 에이전트 패키지인 SUNWmsgvc(Sun Java Communications Suite CD의 Messaging Server Product 하위 디렉토리에 있음)를 설치합니다.
이제 Node_A에 Messaging Server와 Veritas 에이전트가 설치되었습니다.

7 백업 노드(예: Node_B)로 전환합니다.

8 Communications Suite 설치 프로그램을 실행하여 백업 노드(Node_B)에 Messaging Server를 설치합니다.

9 Messaging Server를 설치한 다음 useconfig 유틸리티를 사용하면 백업 노드(Node_B)에 초기 런타임 구성을 추가로 만들 필요가 없습니다. useconfig 유틸리티를 사용하면 단일 구성을 HA 환경의 여러 노드에서 공유할 수 있습니다. 이 유틸리티는 기존 구성을 업그레이드하거나 업데이트하는 용도로 사용할 수는 없습니다. 76 페이지 "3.3.3 useconfig 유틸리티 사용"을 참조하십시오.
이제 Node_B에 Veritas 에이전트가 설치되었습니다.

10 Veritas Cluster Server Cluster Manager에서 File 메뉴의 Import Types... 를 선택합니다. 그러면 파일 선택 상자가 표시됩니다.

11 **/etc/VRTSvcs/conf/config 디렉토리에서 MsgSrvTypes.cf 파일을 가져옵니다. 이 유형 파일을 가져옵니다. 이 파일을 찾으려면 클러스터 노드에 있어야 합니다.**

12 **이제 MsgSrv 유형의 자원(예: Mail)을 만듭니다. 이 자원을 사용하려면 논리 호스트 이름 등록 정보를 설정해야 합니다.**

13 **Mail 자원은 s1ms_mt 및 webip에 종속됩니다. 다음 종속성 트리에 표시되어 있는 것처럼 자원 간에 링크를 만듭니다.**

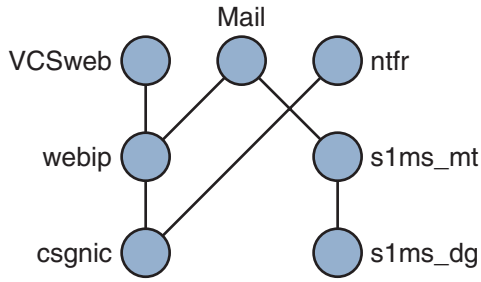


그림 3–6 Veritas Cluster 종속성 트리

a. **모든 자원을 활성화하고 Mail을 온라인 상태로 만듭니다.**

b. **모든 서버가 시작됩니다.**

14 **Node_A로 전환하여 HA 구성이 작동하는지 확인합니다.**

3.5.3 MsgSrv 속성

이 절에서는 mail 자원의 동작을 제어하는 추가 속성인 MsgSrv에 대해 설명합니다. Messaging Server를 Veritas Cluster Server와 함께 구성하려면 표 3–3를 참조하십시오.

표 3–3 Veritas Cluster Server 속성

속성	설명
FaultOnMonitorTimeouts	설정하지 않으면(=0), 모니터(감시) 시간 초과가 자원 오류로 간주되지 않습니다. 2로 설정할 것을 권장합니다. 모니터가 두 번 시간 초과되면 자원이 다시 시작되거나 페일오버됩니다.
ConfInterval	오류/재시작을 계산하는 시간 간격입니다. 서비스가 이 시간 동안 온라인 상태를 유지하면 이전 기록이 지워집니다. 권장 값은 600초입니다.
ToleranceLimit	모니터가 자원 FAULTED를 선언하기 전에 OFFLINE을 반환하는 횟수입니다. 이 값을 "0"(기본값)에 두는 것이 좋습니다.

3.6 고가용성 구성 해제

이 절에서는 고가용성의 구성을 해제하는 방법에 대해 설명합니다. 고가용성을 해제하려면 Veritas 또는 Sun Cluster 설명서의 지침을 따르십시오.

고가용성을 구성 해제하는 방법은 Veritas Cluster Server 또는 Sun Cluster 중 어떤 것을 제거하는지에 따라 달라집니다.

다음 항목에 대해 설명합니다.

- 100 페이지 “Veritas Cluster Server의 구성 해제 방법”

▼ Veritas Cluster Server의 구성 해제 방법

이 절에서는 Veritas Cluster Server에 대한 고가용성 구성 요소의 구성을 해제하는 방법에 대해 설명합니다.

1. **MAIL-RG 서비스 그룹을 오프라인으로 만들고 해당 그룹의 자원을 비활성화합니다.**
2. **mail 자원, logical_IP 자원 및 mountshared 자원 간의 종속성을 제거합니다.**
3. **MAIL-RG 서비스 그룹을 다시 온라인 상태로 전환하여 sharedg 자원을 사용할 수 있도록 합니다.**
4. **설치 도중 생성된 모든 Veritas Cluster Server 자원을 삭제합니다.**
5. **Veritas Cluster Server를 중지하고 두 노드에서 다음 파일을 제거합니다.**

   ```
   /etc/VRTSvcs/conf/config/MsgSrvTypes.cf
   /opt/VRTSvcs/bin/MsgSrv/online
   /opt/VRTSvcs/bin/MsgSrv/offline
   /opt/VRTSvcs/bin/MsgSrv/clean
   /opt/VRTSvcs/bin/MsgSrv/monitor
   /opt/VRTSvcs/bin/MsgSrv/sub.pl
   ```

6. **두 노드의 /etc/VRTSvcs/conf/config/main.cf 파일에서 Messaging Server 항목을 제거합니다.**
7. **두 노드에서 /opt/VRTSvcs/bin/MsgSrv/ 디렉토리를 제거합니다.**

◆ ◆ ◆ 4 장

4 일반 메시징 기능 구성

이 장에서는 서비스의 시작과 중지 및 디렉토리 액세스 구성 등과 같이 명령줄 유틸리티를 사용하여 수행하는 일반적인 Messaging Server 작업에 대해 설명합니다. 개별 Messaging Server 서비스에 한정된 작업(POP, IMAP, HTTP 및 SMTP)은 다음 장에서 설명합니다. 이 장은 다음 내용으로 구성되어 있습니다.

4.1 비밀번호 수정

초기 구성 중에 동일한 비밀번호를 갖는 관리자 수를 설정했기 때문에(49 페이지 "1.3 Messaging Server 초기 런타임 구성 만들기 " 참조) 해당 관리자의 비밀번호를 변경할 수 있습니다.

표 4–1에는 초기 런타임 구성 시 설정된 기본 비밀번호의 매개 변수와 이를 변경하기 위해 사용할 수 있는 유틸리티가 나열되어 있습니다. `configutil` 유틸리티를 사용하는 매개 변수의 전체 구문과 사용법은 **Sun Java System Messaging Server 6.3 Administration Reference**의 "configutil"을 참조하십시오.

표 4–1 Messaging Server 초기 런타임 구성 시 설정된 비밀번호

매개 변수	설명
`local.ugldapbindcred`	`configutil` 유틸리티를 통해 설정된 사용자/그룹 관리자의 비밀번호입니다.
`local.service.pab.ldappasswd`	`configutil` 유틸리티를 통해 설정된 PAB 검색용 바인드 DN에 의해 지정된 사용자의 비밀번호입니다.
키 파일의 SSL 비밀번호	`sslpassword.conf` 파일에 직접 설정된 비밀번호입니다.
서비스 관리자 자격 증명	LDAP 디렉토리에 직접 설정된 자격 증명입니다(`ldapmodify` 명령 사용).
Delegated Administrator에 대한 서비스 관리자	Sun LDAP Schema 1을 활성화했고 Delegated Administrator 유틸리티를 사용하는 경우 이 관리자의 비밀번호만 변경하면 됩니다. Delegated Administrator 서비스 관리자의 비밀번호를 변경하려면 LDAP 디렉토리(`ldapmodify` 명령 사용) 또는 Delegated Administrator UI를 수정하여 변경할 수 있습니다.
저장소 관리자	저장소 관리자의 비밀번호를 변경하려면 LDAP 디렉토리(`ldapmodify` 명령 사용)를 수정하여 변경할 수 있습니다.

다음 예에서는 `local.enduseradmincred` `configutil` 매개 변수를 사용하여 최종 사용자 관리자의 비밀번호를 변경합니다.

```
configutil -o local.enduseradmincred -v newpassword
```

4.2 메일 사용자, 메일링 목록 및 도메인 관리

모든 사용자, 메일 목록 및 도메인 정보는 LDAP 디렉토리에 항목으로 저장됩니다. LDAP 디렉토리에는 조직의 직원, 구성원, 클라이언트 또는 조직에 여러 방법으로 "속해 있는" 다양한 유형의 개인에 대한 광범위한 정보가 포함되어 있습니다. 이러한 개인들이 조직의 **사용자**를 구성합니다.

LDAP 디렉토리에서 사용자에 대한 정보는 효율적인 검색을 위해 구조화되어 있으며 각 사용자 항목은 속성 집합으로 식별됩니다. 사용자와 연관된 디렉토리 속성에는 사용자 아이디와 기타 아이디, 부서, 작업 분류, 물리적 위치, 관리자 이름, 직접 보고자 이름, 조직의 각 부분에 대한 액세스 권한 및 기타 여러 기본 설정 등이 포함될 수 있습니다.

전자 메시징 서비스를 갖춘 조직에서는 대부분의 사용자가 메일 계정을 보유하고 있습니다. Messaging Server의 경우 메일 계정 정보는 서버에 로컬로 저장되는 것이

아니라 LDAP 사용자 디렉토리의 일부입니다. 각 메일 계정에 대한 정보는 디렉토리의 사용자 항목에 첨부된 메일 속성으로 저장됩니다.

메일 사용자와 메일 목록을 만들고 관리하는 작업은 디렉토리에서 사용자와 메일 목록 항목을 만들고 수정하는 작업으로 구성됩니다. 이 작업은 Sun LDAP Schema 2용 Delegated Administrator 및 메시징용 iPlanet Delegated Administrator(Sun LDAP Schema 1용), Delegated Administrator 명령줄 유틸리티를 사용하거나 Sun LDAP Schema 1용 LDAP 디렉토리를 직접 수정하는 방식으로 수행합니다.

▼ Messaging Server에서 사용자 제거 방법

1 `commadmin user delete` **명령을 실행하여 사용자가 삭제된 것으로 표시합니다. Sun Java System Delegated Administrator 6.4 관리 설명서의 5 장,"명령줄 유틸리티"를 참조하십시오.**

2 **사용자로부터 서비스를 제거합니다.**

서비스는 메일함 또는 달력일 수 있습니다. 현재 버전의 Messaging Server에서는 이 프로그램을 `msuserpurge`라고 합니다. **Sun Java System Messaging Server 6.3 Administration Reference**의 "msuserpurge"를 참조하십시오. 달력 서비스에서는 프로그램이 `csclean`입니다. **Sun Java System Calendar Server 6.3 Administration Guide**를 참조하십시오.

3 `commadmin domain purge` **명령을 호출하여 사용자를 영구적으로 제거합니다.**

▼ Messaging Server에서 도메인 제거 방법

1 `commadmin domain delete` **명령을 실행하여 도메인이 삭제된 것으로 표시합니다. Sun Java System Delegated Administrator 6.4 관리 설명서의 5 장,"명령줄 유틸리티"를 참조하십시오.**

2 **해당 도메인의 사용자로부터 서비스를 제거합니다.**

서비스는 메일함 또는 달력일 수 있습니다. Messaging Server에서는 이 프로그램을 `msuserpurge`라고 합니다. **Sun Java System Messaging Server 6.3 Administration Reference**의 "msuserpurge"를 참조하십시오. 달력 서비스에서는 프로그램이 `csclean`입니다. *Sun Java System Calendar Server* **관리 설명서**를 참조하십시오.

3 `commadmin domain purge` **명령을 호출하여 사용자를 영구적으로 제거합니다.**

4.3 Sun ONE 콘솔로 Messaging Server 관리

Sun ONE 관리 콘솔은 Messaging Server에서 더 이상 지원되지 않습니다. 해당 명령줄 인터페이스를 사용하십시오.

4.4 서비스 시작 및 중지

서비스는 HA 환경에 설치되었는지 여부에 따라 다른 방식으로 시작 및 중지됩니다.

4.4.1 HA 환경에서 서비스 시작 및 중지

Messaging Server가 HA 제어 하에 실행 중일 때는 일반적인 Messaging Server 시작, 다시 시작 및 중지 명령으로 개별 Messaging Server 서비스를 제어할 수 없습니다. HA 환경에서 `stop-msg`를 시도하면 시스템에서는 HA 설정이 감지되었다는 경고 메시지와 함께 시스템을 올바로 중지하는 방법을 알려줍니다.

시작, 중지 및 다시 시작 명령은 아래 표에서 볼 수 있습니다. 다른 Messaging Server 서비스(예: SMTP)를 개별적으로 시작, 다시 시작 또는 중지시키는 특정 HA 명령은 없습니다. 하지만 `stop-msg` *service* 명령을 실행하여 `imap`, `pop`, `sched` 등의 개별 서비스를 중지하고 다시 시작할 수 있습니다.

Sun Cluster에서는 개별 자원을 세부적으로 제어할 수 있습니다. Sun Cluster는 Messaging Server를 자원으로 인식하기 때문에 `scswitch` 명령은 모든 Messaging Server 서비스에 적용됩니다.

표 4–2 Sun Cluster 3.0/3.1 환경에서 시작, 중지, 다시 시작

작업	개별 자원	전체 자원 그룹
시작	`scswitch -e -j` *resource*	`sscswitch -Z -g` *resource_group*
다시 시작	`scswitch -n -j` *resource* `scswitch -e -j` *resource*	`scswitch -R -g` *resource_group*
중지	`scswitch -n -j` *resource*	`scswitch -F -g` *resource_group*

표 4–3 Veritas 3.5, 4.0, 4.1 및 5.0 환경에서 시작, 중지, 다시 시작

작업	개별 자원	전체 자원 그룹
시작	`hares -online` *resource* `-sys` *system*	`hagrp -online` *group* `-sys` *system*
다시 시작	`hares -offline` *resource* `-sys` *system* `hares -online` *resource* `-sys` *system*	`hagrp -offline` *group* `-sys` *system* `hagrp -online` *group* `-sys` *system*

표 4–3 Veritas 3.5, 4.0, 4.1 및 5.0 환경에서 시작, 중지, 다시 시작 *(계속)*

작업	개별 자원	전체 자원 그룹
중지	`hares -offline` *resource* -sys *system*	`hagrp -offline` *group* -sys *system*

4.4.2 HA가 아닌 환경에서 서비스 시작 및 중지

msg-svr-base`/sbin/start-msg` 및 *msg-svr-base* `/sbin/stop-msg` 명령을 사용하여 명령줄에서 서비스를 시작하고 중지합니다. 명령 템플리트 *msg-svr-base* `/sbin/stop-msg` *service*(여기서 서비스는 `smtp`, `imap`, `pop`, `store`, `http`, `ens` 또는 `sched`)를 사용하여 서비스를 개별적으로 시작하고 중지할 수 있지만 이 설명서에서 설명하는 특정 작업을 제외하고는 권장되지 않습니다. 특정 서비스는 다른 서비스에 종속되므로 미리 지정된 순서대로 시작되어야 합니다. 서비스를 자체적으로 시작하려고 하면 복잡해질 수 있습니다. 따라서 `start-msg` 및 `stop-msg` 명령을 사용하여 모든 서비스를 함께 시작하고 중지해야 합니다.

주 – POP, IMAP 및 HTTP 등의 서비스를 시작하거나 중지하려면 먼저 해당 서비스들을 활성화해야 합니다. 자세한 내용은 118 페이지 “5.1.1 서비스 활성화/비활성화”를 참조하십시오.

중요: 서버 프로세스가 충돌하는 경우 충돌한 서버 프로세스에 의한 잠금을 기다리고 있는 다른 프로세스도 중지될 수 있습니다. 자동 재시작(107 페이지 “4.5 실패했거나 응답이 없는 서비스의 자동 재시작” 참조)을 사용하지 않는 경우 서버 프로세스가 충돌하면 모든 프로세스를 중지시킨 다음 모든 프로세스를 다시 시작해야 합니다. 여기에는 `stored`(메시지 저장소) 프로세스, `mboxutil`, `deliver`, `reconstruct`, `readership` 또는 `upgrade` 등과 같이 메시지 저장소를 수정하는 모든 유틸리티와 POP, IMAP, HTTP 및 MTA 프로세스가 포함됩니다.

▼ 메시징 서비스 시작, 종료 또는 상태 보기 방법

이 설명서의 다양한 부분에서 설명하는 특정 작업을 제외하고 개별 서비스를 종료하는 것은 권장되지 않습니다. 특정 서비스는 다른 서비스에 종속되므로 미리 지정된 순서대로 시작되어야 합니다. 서비스를 자체적으로 시작하려고 하면 복잡해질 수 있습니다. 따라서 `start-msg` 및 `stop-msg` 명령을 사용하여 모든 서비스를 함께 시작하고 중지해야 합니다.

- **`start-msg` 및 `stop-msg` 명령을 사용하여 메시징 서비스를 시작하거나 중지합니다. 예를 들면 다음과 같습니다.**

  msg-svr-base`/sbin/start-msg imap`

  msg-svr-base`/sbin/stop-msg pop`

  msg-svr-base`/sbin/stop-msg sched`

msg-svr-base`/sbin/stop-msg smtp`

서비스를 시작하거나 중지하려면 먼저 해당 서비스를 활성화해야 합니다. 106 페이지 "4.4.2.1 시작할 서비스 지정"을 참조하십시오.

주 – `start-msg` 및 `stop-msg` 명령은 SMTP 서버뿐 아니라 모든 MTA 서비스를 시작 및 중지합니다. MTA 서비스를 시작하거나 중지할 때 보다 세부적으로 제어하려면 디스패처 및 작업 제어기에 대해 `start/stop-msg` 명령을 사용할 수 있습니다. 자세한 내용은 **Sun Java System Messaging Server 6.3 Administration Reference**의 "start-msg" 및 **Sun Java System Messaging Server 6.3 Administration Reference**의 "stop-msg"를 참조하십시오.

4.4.2.1 시작할 서비스 지정

기본적으로 다음 서비스는 `start-msg`를 사용하여 시작합니다.

```
#./start-msg
Connecting to watcher ...
Launching watcher ...
Starting ens server .... 21132
Starting store server .... 21133
checking store server status ... ready
Starting imap server .... 21135
Starting pop server .... 21138
Starting http server .... 21141
Starting sched server .... 21143
Starting dispatcher server .... 21144
Starting job_controller server .... 21146
```

이러한 서비스는 `configutil` 매개 변수인 `service.imap.enable`, `service.pop.enable`, `service.http.enable`, `local.smsgateway.enable`, `local.snmp.enable`, `local.imta.enable`, `local.mmp.enable`, `local.ens.enable` 및 `local.sched.enable`을 활성화하거나 비활성화하여 제어할 수 있습니다. IMAP를 비활성화하려면 `service.imap.enable` 및 `service.imap.enablesslport`를 둘 다 `0`으로 설정해야 합니다. POP 및 HTTP의 경우에도 마찬가지입니다. 작동 방식에 대한 자세한 내용은 **Sun Java System Messaging Server 6.3 Administration Reference**의 "configutil Parameters"를 참조하십시오.

4.4.3 MTA 전용 모드에서 실행하는 Messaging Server 시작 및 중지

MTA 전용 시스템을 시작하려면 `imsched`도 시작해야 합니다. 이 작업에 앞서 예약된 작업 중 설치에 적합하지 않은 것을 모두 제거합니다.

imsched는 Messaging Server의 개별 구성 요소로서, 일부 Messaging Server를 시작하지 않을 경우 별도로 시작해야 합니다. start-msg imta 또는 start-msg smtp를 사용하여 MTA 전용 시스템을 시작할 경우, imsched 프로세스를 실행하지 않습니다.

MTA 모드에서만 메시징 서버를 실행하려면(store/imap/pop/http 프로세스 없음) 초기화 설치 후 메시징 서버 구성 도중 MTA를 설치/구성하도록 선택하거나(*msg_base*/sbin/configure), 다음 configutil 명령을 사용하여 메시지 저장소 및 mshttp 프로세스를 수동으로 비활성화하면 됩니다.

```
./configutil -o local.store.enable -v 0
./configutil -o service.http.enable -v 0
```

http 및 기타 store 프로세스를 비활성화한 경우 다음을 실행하여 메시징 서버를 시작할 수 있습니다.

```
# ./start-msg
bash-3.00# ./start-msg
Connecting to watcher ...
Launching watcher ... 4034
Starting ens server ... 4035
Starting sched server ... 4036
Starting dispatcher server .... 4038
Starting job_controller server .... 4042
```

imsched 및 imta를 비롯하여 해당되는 모든 프로세스가 시작됩니다. 이렇게 하면 sched 프로세스를 시작해야 한다는 것을 기억할 필요가 없습니다.

4.5 실패했거나 응답이 없는 서비스의 자동 재시작

Messaging Server는 서비스를 투명하게 모니터하고 서비스가 실패하거나 응답하지 않을 경우 즉, 서비스가 중지된 경우 서비스를 자동으로 다시 시작하는 watcher 및 msprobe라는 두 개의 프로세스를 제공합니다. watcher는 서버 실패를 모니터하고 msprobe는 서버 응답 시간을 검사하여 서버 중단을 모니터합니다. 서버가 실패하거나 요청에 대한 응답이 중지되면 자동으로 다시 시작됩니다. 표 4–4는 각 유틸리티에 의해 모니터링되는 서비스를 나타냅니다.

표 4–4 watcher 및 msprobe에서 모니터하는 서비스

watcher(크래시)	msprobe(응답하지 않은 보류)
IMAP, POP, HTTP, 작업 제어기, 디스패처, 메시지 저장소(stored), imsched, MMP. LMTP/SMTP 서버는 디스패처가 모니터하며 LMTP/SMTP 클라이언트는 job_controller가 모니터합니다.	IMAP, POP, HTTP, 작업 제어기, 메시지 저장소(stored), imsched, ENS, LMTP, SMTP

local.watcher.enable=on(기본값)으로 설정하면 프로세스 실패와 응답하지 않는 서비스를 모니터하여 특정 실패를 나타내는 default 로그 파일에 오류 메시지를 기록합니다. 자동 서버 재시작을 활성화하려면 configutil 매개 변수 local.autorestart를 yes로 설정합니다. 기본적으로 이 매개 변수는 no로 설정됩니다.

메시지 저장소 서비스 중 하나가 실패하거나 중지되면 시작 시 활성화된 모든 메시지 저장소 서비스가 다시 시작됩니다. 예를 들어 imapd가 실패하면 적어도 stored와 imapd가 다시 시작됩니다. POP 또는 HTTP 서버 등의 다른 메시지 저장소 서비스가 실행 중인 경우 해당 서비스는 그 실패 여부와 관계 없이 다시 시작됩니다.

메시지 저장소 유틸리티가 실패하거나 중지되어도 자동 재시작이 작동합니다. 예를 들어 mboxutil이 실패하거나 중지되면 시스템은 모든 메시지 저장소 서버를 자동으로 다시 시작합니다. 하지만 유틸리티는 다시 시작하지 않습니다. msprobe는 10분마다 실행됩니다. 서비스 및 프로세스 재시작은 10분 내에 최대 두 번 수행됩니다(local.autorestart.timeout을 사용하여 구성 가능).

local.autorestart의 yes 설정 여부에 관계 없이 시스템은 서비스를 모니터하여 실패 또는 무응답 오류 메시지를 해당 콘솔로 전송하고 *msg-svr-base*/data/log/ watcher는 기본 포트 49994를 통해 수신하지만 local.watcher.port를 사용하여 구성할 수도 있습니다.

watcher 로그 파일은 *msg-svr-base*/data/log/watcher에 생성됩니다. 이 로그 파일은 로깅 시스템(롤오버 또는 제거)에서 관리하는 것이 아니며 모든 서버의 시작과 중지를 기록합니다. 로그 예는 다음과 같습니다.

```
watcher process 13425 started at Tue Oct 21 15:29:44 2003

Watched 'imapd' process 13428 exited abnormally
Received request to restart:  store imap pop http
Connecting to watcher ...
Stopping http server 13440 .... done
Stopping pop server 13431 ... done
Stopping pop server 13434 ... done
Stopping pop server 13435 ... done
Stopping pop server 13433 ... done
imap server is not running
Stopping store server 13426 .... done
Starting store server .... 13457
checking store server status ...... ready
Starting imap server ..... 13459
Starting pop server ....... 13462
Starting http server ...... 13471
```

이 기능을 구성하는 방법에 대한 자세한 내용은 850 페이지 "27.8.9 msprobe 및 watcher 기능을 사용하여 모니터링"을 참조하십시오.

msprobe는 imsched로 제어됩니다. imsched가 충돌하면 watcher에서 이 이벤트를 검색하여 다시 시작(autorestart가 사용 가능한 경우)을 트리거합니다. 드물기는 하지만 imsched가 보류되는 경우 watcher에서 다시 시작하도록 하는 kill *imsched_pid*를 사용하여 imsched를 중지해야 할 수 있습니다.

4.5.1 고가용성 배포 시 자동 재시작

고가용성 배포 시 자동 재시작 기능을 사용하려면 다음 configutil 매개 변수를 설정해야 합니다.

표 4–5 HA 자동 재시작 매개 변수

매개 변수	설명/HA 값
local.watcher.enable	start-msg를 시작할 때 watcher를 활성화합니다. 기본값은 yes입니다.
local.autorestart	IMAP, POP, HTTP, 작업 제어기, 디스패처 및 MMP 서버를 포함하여 실패 또는 고정(응답 없음) 서버의 자동 재시작을 활성화합니다. 기본값은 No입니다.
local.autorestart.timeout	재시도 시간 초과 오류. 지정된 기간 내에 서버가 세 번 이상 실패하면 시스템은 서버 재시작 시도를 중지합니다. HA 시스템에서 이런 상황이 발생하면 Messaging Server가 종료되고 다른 시스템으로 페일오버됩니다. 값(초)은 msprobe 간격보다 더 긴 기간 값으로 설정해야 합니다. 아래 local.schedule.msprobe를 참조하십시오. 기본값은 600입니다.
local.schedule.msprobe	msprobe에서 일정을 실행합니다. crontab 스타일 일정 문자열(표 20–10 참조)입니다. 기본값은 5,15,25,35,45,55 * * * * lib/msprobe입니다. 비활성화하려면 local.schedule.msprobe.enable을 NO로 설정합니다.

4.6 자동 작업 예약

Messaging Server에서는 imsched라는 프로세스를 사용한 일반 작업 예약 기법을 제공합니다. Messaging Server 프로세스를 예약하기 위한 것입니다. 이 기능은 local.schedule.*taskname* configutil 매개 변수를 설정하여 활성화됩니다. 일정을 수정하려면 stop-msg sched 및 start-msg sched 명령을 사용하여 스케줄러를 다시 시작해야 합니다. 스케줄러 프로세스를 새로 고칠(refresh sched) 수도 있습니다.

이 매개 변수에는 명령과 해당 명령을 실행할 일정이 필요합니다. 형식은 다음과 같습니다.

configutil -o local.schedule.*taskname* -v "*schedule*"

*taskname*은 이 명령/일정 조합의 고유 이름입니다.

*schedule*의 형식은 다음과 같습니다.

minute hour day-of-month month-of-year day-of-week command args

*command args*는 Messaging Server 명령과 그 인수일 수 있습니다. 경로는 *msg-svr-base*에 대한 상대 경로이거나 절대 경로일 수 있습니다. 상대 경로의 예는 110 페이지 "4.6.2 미리 정의된 자동 작업"을 참조하십시오.

*minute hour day-of-month month-of-year day-of-week*는 명령을 실행하는 일정입니다. UNIX `crontab`의 형식을 따릅니다.

값은 공백이나 탭으로 구분하며 각각 0-59, 0-23, 1-31, 1-12 또는 0-6(0=일요일)의 값을 사용할 수 있습니다. 각 시간 필드에는 별표(유효한 모든 값), 쉼표로 구분된 값 목록 또는 하이픈으로 구분된 두 값의 범위를 사용할 수 있습니다. 일에는 한 달의 숫자와 요일을 모두 사용할 수 있으며 지정된 경우에는 둘 다 필요합니다. 예를 들어, 17일과 화요일로 설정하면 명령은 17일, 화요일에만 실행됩니다. 표 20–10을 참조하십시오.

스케줄러를 수정한 경우 `stop-msg sched` 및 `start-msg sched` 명령을 사용하여 스케줄러를 다시 시작해야 합니다. 또는 스케줄러 프로세스를 새로 고치면 됩니다.

```
refresh sched
```

예약된 작업을 비활성화하려면 다음을 실행합니다.

```
# configutil -o local.schedule.taskname.enable -v no
#  refresh sched
```

4.6.1 스케줄러의 예

오전 12:30 , 8:30 및 오후 4:30에 `imexpire`를 실행합니다.

```
# configutil -o local.schedule.rm_messages -v   30 0,8,16 * * * /opt/SUNWmsgsr/sbin/imexpire
```

20분마다 MTA 채널 대기열 메시지 카운터를 표시합니다.

```
# configutil -o local.schedule.counters -v   0,20,40 * * * * /opt/SUNWmsgsr/sbin/ims
# imta qm counters > /tmp/temp.txt
```

월요일부터 금요일까지 오전 12시에 `imsbackup`을 실행합니다.

```
# configutil -o local.schedule.msbackup -v   0 0 * * 1-5 /opt/SUNWmsgsr/sbin/imsbackup -f \
backupfile /primary
```

4.6.2 미리 정의된 자동 작업

설치할 때 Messaging Server가 미리 정의된 자동 작업을 생성, 예약 및 활성화합니다. 아래 내용을 참조하십시오.

다음 자동 작업이 메시지 저장소에 대해 설정되어 활성화됩니다.

```
local.schedule.expire = "0 23 * * * sbin/imexpire"
local.schedule.expire.enable = 1
local.schedule.snapshotverify = "0 0,4,8,12,16,20 * * * sbin/imdbverify -m"
local.schedule.snapshotverify.enable = 1
```

다음 자동 작업이 MTA에 대해 설정되어 활성화됩니다.

```
local.schedule.purge="0 0,4,8,12,16,20 * * * sbin/imsimta purge -num=5"
local.schedule.purge.enable = 1
local.schedule.return_job = "30 0 * * * lib/return_job"
local.schedule.return_job.enable = 1
```

다음 자동 작업이 메시지 저장소에 대해 설정되어 활성화됩니다.

```
local.schedule.msprobe = "5,15,25,35,45,55 * * * * lib/msprobe"
local.schedule.msprobe.enable = 1
```

4.7 인사 메시지 구성

Messaging Server에서는 새 사용자에게 보낼 전자 메일 인사 메시지를 작성할 수 있습니다.

▼ 새 사용자 인사 메시지 작성 방법

- **명령줄을 사용하여 새 사용자 인사 메시지를 만듭니다.**

  `configutil -o gen.newuserforms -v` *Message*

  여기서 *Message*는 최소한 제목 줄이 들어있는 헤더를 포함하고 그 뒤에 $$와 메시지 본문이 와야 합니다. $는 새 행을 나타냅니다.

  예를 들어, 이 매개 변수를 활성화하려면 다음 구성 변수를 설정해야 합니다.

  ```
  configutil -o gen.newuserforms -v 'Subject: Welcome!! $$ Sesta.com welcomes you
  to the premier internet experience in Dafandzadgad!
  ```

  사용 중인 쉘에 따라 $의 특수한 의미를 이스케이프하기 위해 $ 앞에 특수 문자를 추가해야 할 수 있습니다. $는 일반적으로 해당 쉘의 이스케이프 문자입니다.

4.7.1 도메인별 인사 메시지 설정 방법

호스트된 새 도메인을 만들 때마다 지원되는 언어에 대한 도메인별 인사 메시지를 만드는 것이 좋습니다. 그렇지 않을 경우 gen.newuserform에 의해 설정된 일반 인사 메시지가 보내집니다.

각 도메인의 새 사용자에 대해 인사 메시지를 설정할 수 있습니다. 인사 메시지는 사용자, 도메인 또는 사이트의 기본 언어에 따라 달라질 수 있습니다. 이 작업은 원하는 LDAP 도메인 항목에서 mailDomainWelcomeMessage 속성을 설정하여 수행합니다. 속성 구문은 다음과 같습니다.

```
mailDomainWelcomeMessage;lang-user_prefLang
mailDomainWelcomeMessage;lang-domain_prefLang
mailDomainWelcomeMessage;lang-gen.sitelanguage
```

다음 예는 영어를 사용하는 도메인 시작 메시지를 설정합니다.

```
mailDomainWelcomeMessage;lang-en: Subject: Welcome!! $$Welcome to the mail
system.
```

다음 예는 프랑스어를 사용하는 도메인 시작 메시지를 설정합니다.

```
mailDomainWelcomeMessage;lang-fr: Subject: Bienvenue!! $$Bienvenue a siroe.com!
```

위의 예를 사용할 때 다음을 가정합니다.

- 도메인은 siroe.com임
- 이 도메인에 새 사용자가 소속됨
- 이 사용자의 기본 언어는 LDAP 속성 `preferredlanguage`에서 지정한 대로 프랑스어임
- siroe.com에서는 위의 영어 및 프랑스어 환영 메시지를 사용할 수 있음
- 사이트 언어는 `gen.sitelanguage`에서 지정한 대로 en임

지원되는 로켈과 해당 언어 값 태그의 목록은 Directory Server Reference Manual (`http://docs.sun.com`)을 참조하십시오.

사용자가 처음 로그인하면 프랑스어 인사 메시지를 받게 됩니다. 프랑스어 시작 메시지를 사용할 수 없는 경우에는 영어 메시지가 사용됩니다.

4.7.1.1 인사 메시지 작동 원리

인사 메시지는 LDAP 속성 `mailDomainWelcomeMessage`와 configutil 매개 변수 `gen.newuserforms`를 둘 다 사용하여 설정할 수 있습니다. 다음은 메시지가 선택되는 순서를 가장 높은 우선 순위부터 보여 줍니다.

```
mailDomainWelcomeMessage;lang-user_prefLang
mailDomainWelcomeMessage;lang-domain_prefLang
mailDomainWelcomeMessage;lang-gen.sitelanguage
mailDomainWelcomeMessage
gen.newuserforms;lang-"$user-prefLang"
gen.newuserforms;lang-"$domain-prefLang"
gen.newuserforms;lang-"$gen.sitelanguage"
gen.newuserforms
```

알고리즘은 다음과 같이 실행됩니다. 도메인이 없는 경우(또는 도메인이 있더라도 도메인별 시작 메시지가 제공되지 않는 경우)에는 gen.newuserforms 매개 변수로 시작 메시지가 지정됩니다. 즉 사용자가 기본 언어를 설정하고(preferredlanguage LDAP 속성으로 설정) gen.newuserforms;lang-*user_prefLang*이 설정되어 있으면 사용자가 서버에 처음 로그인할 때 해당 메시지를 받게 됩니다.
gen.newuserforms;lang-*gen.sitelanguage*가 설정되 어 있고 preferredlanguage는 설정되어 있지 않지만 해당 사이트 언어가 설정된 경우(gen.sitelanguage 매개 변수 사용) 사용자는 해당 메시지를 받게 됩니다. 언어 태그 매개 변수가 설정되지 않고 태그되지 않은 gen.newuserforms가 설정된 경우 해당 메시지가 사용자에게 보내집니다. 위의 값 중 아무 것도 설정되지 않은 경우 사용자는 시작 메시지를 받을 수 없습니다.

사용자가 도메인에 있다고 가정하면 위에 설명된 것과 같이 사용자는 다음 목록 중 사용 가능한 것에 따라 지정된 순서대로 mailDomainWelcomeMessage;lang-xx 중 하나를 받을 수 있습니다.

예: 도메인이 siroe.com인 경우 도메인 기본 언어는 독일어(de)입니다. 하지만 이 도메인의 새 사용자 기본 언어는 터키어(tr)이고사이트 언어는 영어입니다. 다음 값을 사용할 수 있습니다(mailDomainWelcomeMessage는 siroe.com 도메인 속성).

```
mailDomainWelcomeMessage;lang-fr
mailDomainWelcomeMessage;lang-ja
gen.newuserforms;lang-de
gen.newuserforms;lang-en
gen.newuserforms
```

이 알고리즘에 따라 사용자에게 보내지는 메시지는 gen.newuserforms;lang-de가 됩니다.

4.8 사용자 기본 언어 설정

관리자는 사용자의 LDAP 항목에서 preferredLanguage 속성을 설정하여 GUI와 서버 생성 메시지에 사용할 기본 언어를 설정할 수 있습니다.

서버가 서버의 관리 도메인 외부에 있는 사용자에게 메시지를 보내는 경우에는 받는 메시지의 헤더에 지정된 기본 언어를 사용하여 응답하는 경우 이외에는 관리 도메인이 해당 사용자의 기본 언어를 알 수 없습니다. 헤더 필드(Accept-Language, Preferred-Language 또는 X-Accept-Language)는 사용자의 메일 클라이언트에 지정된 속성에 따라 설정됩니다.

기본 언어에 대해 여러 개의 설정이 있는 경우(예: 사용자에게 Directory Server에 저장된 기본 언어 속성도 있고 자신의 메일 클라이언트에 지정된 기본 언어도 있는 경우) 서버는 다음 순서로 기본 언어를 선택합니다.

1. 원본 메시지의 Accept-Language 헤더 필드
2. 원본 메시지의 Preferred-Language 헤더 필드

3. 원본 메시지의 X-Accept-Language 헤더 필드
4. 보내는 사람의 기본 언어 속성(LDAP 디렉토리에 있는 경우)

4.8.1 도메인 기본 언어 설정

도메인 기본 언어는 특정 도메인에 대해 지정된 기본 언어입니다. 예를 들어, mexico.siroe.com이라는 도메인에 대해 스페인어를 지정하려 할 수 있습니다. 관리자는 도메인의 LDAP 항목에서 preferredLanguage 속성을 설정하여 도메인 기본 언어를 설정할 수 있습니다.

▼ 사이트 언어 지정 방법

다음과 같이 서버의 기본 사이트 언어를 지정할 수 있습니다. 사이트 언어는 사용자 기본 언어가 설정되어 있지 않은 경우 언어별 메시지 버전을 보내는데 사용됩니다.

- **명령줄: 사이트 언어를 다음과 같이 지정합니다.**

  configutil -o gen.sitelanguage -v *value*

  여기서 *value*는 로컬 지원 언어 중 하나입니다. 지원되는 로켈과 언어 값 태그의 목록은 **Sun Java System Directory Server 5 2005Q1 Administration Guide**의 5장을 참조하십시오.

4.9 디렉토리 조회 사용자 정의

Messaging Server는 Sun Java System Directory Server와 같은 LDAP 기반 디렉토리 시스템이 없이는 작동되지 않습니다. Messaging Server에는 여러 가지 목적에 따라 디렉토리 액세스가 필요합니다. 예를 들면 다음과 같습니다.

- 메일 사용자 또는 메일 그룹의 계정 정보를 만들거나 업데이트할 때 해당 정보는 **사용자 디렉토리**라는 디렉토리에 저장됩니다.
- 메시지를 메일함으로 라우팅하고 메일함으로 전달할 때 Messaging Server는 사용자 디렉토리에서 보내는 사람이나 받는 사람에 대한 정보를 조회합니다.
- 메일 라우팅 조회에 대해 사용자를 인증할 때

Messaging Server가 사용자 및 그룹 조회를 위해 다른 사용자 디렉토리에 연결하는 것은 선택 사항입니다. 일반적으로 서버의 관리 도메인을 정의하는 사용자 디렉토리는 도메인의 모든 서버가 사용하는 디렉토리입니다.

▼ Messaging Server LDAP 사용자 조회 설정 수정 방법

- **사용자 디렉토리 연결 설정 명령은 아래에 표시되어 있지만 먼저 LDAP 및 PAB 비밀번호를 다음과 같이 설정합니다.**

- 구성 속성 `local.ugldapbinddn`에 지정된 사용자의 비밀번호를 수정합니다. 이 사용자 계정은 구성 속성 `local.ugldaphost`에 지정된 Directory Server에 있습니다.
- 속성 `local.service.pab.ldapbinddn` 및 `local.service.pab.ldaphost` 에 지정된 동일한 계정을 PAB 액세스에 사용하는 경우에는 `local.service.pab.ldappasswd`에 저장된 비밀번호를 업데이트해야 합니다.

Messaging Server별 디렉토리 설정의 사용 여부를 지정하는 경우

```
configutil -o local.ugldapuselocal -v [ yes | no ]
```

호스트 이름은 사용자 설치 시 사용자 정보가 있는 디렉토리의 호스트 시스템 이름입니다. 대부분 Messaging Server 호스트와는 다르지만 매우 소규모 설치인 경우에는 동일한 호스트일 수 있습니다. 사용자 조회를 위한 LDAP 호스트 이름을 지정하는 경우

```
configutil -o local.ugldaphost -v name[:port_number]
```

포트 번호는 Messaging Server가 사용자 조회를 위해 디렉토리에 액세스할 때 사용해야 하는 디렉토리 호스트의 포트 번호입니다. 이 번호는 디렉토리 관리자가 정의하며 반드시 기본 포트 번호(389)일 필요는 없습니다. 사용자 조회를 위한 LDAP 포트 번호를 지정하는 경우

```
configutil -o local.ugldapport -v number
```

기본 DN:검색 기본, 즉 사용자 조회를 위한 시작점을 나타내는 디렉토리 항목의 고유 이름입니다. 조회 프로세스의 속도를 높이려면 검색 기본이 디렉토리 트리에서 검색 중인 정보와 가능한 가까워야 합니다. 사용자 설치 디렉토리 트리에 "사람" 또는 "사용자" 분기가 있는 경우 이것이 좋은 시작점이 될 수 있습니다. 사용자 조회를 위한 LDAP 기본 DN을 지정하는 경우

```
configutil -o local.ugldapbasedn -v basedn
```

바인드 DN: Messaging Server가 조회를 위해 Directory Server에 연결할 때 자신을 나타내기 위해 사용하는 고유 이름입니다. 바인드 DN은 디렉토리의 사용자 부분에 대한 검색 권한을 갖고 있는 사용자 디렉토리 자체에 있는 항목의 고유 이름이어야 합니다. 익명 검색 액세스가 허용되는 디렉토리의 경우에는 이 항목을 비워둘 수 있습니다. 사용자 조회를 위한 LDAP 바인드 DN을 지정하는 경우

```
configutil -o local.ugldapbinddn -v binddn
```

4.10 암호화 설정

이 내용은 697 페이지 "23.5.2 SSL 사용 및 암호문 선택"에 설명되어 있으며, Messaging Server의 모든 보안 및 액세스 제어 항목에 대한 배경 정보가 포함되어 있습니다.

4.11 페일오버 LDAP 서버 설정

하나가 실패하면 다른 하나가 작동하도록 다음과 같이 사용자/그룹 디렉토리에 대한 LDAP 서버를 두 개 이상 지정할 수 있습니다.

▼ LDAP 서버 페일오버 설정 방법

1 **여러 LDAP 시스템에** `local.ugldaphost`**를 설정합니다. 예:**

```
configutil -o local.ugldaphost -v "server1 server2 ..."
```

2 `local.ugldapuselocal`**을** `yes`**로 설정합니다. 이 작업은 사용자/그룹 LDAP 구성 데이터가 로컬 구성 파일에 저장되도록 지정합니다. 그렇지 않은 경우에는 LDAP에 저장됩니다. 예:**

```
configutil -o local.ugldapuselocal -v yes
```

만약

목록의 첫 번째 서버가 실패하면 기존 LDAP 연결이 종료된 것으로 인식되어 새 연결이 생성됩니다. 새 LDAP 연결이 필요하면 LDAP 라이브러리는 모든 LDAP 서버를 나열된 순서대로 시도합니다.

◆ ◆ ◆ 5 장

5 POP, IMAP 및 HTTP 서비스 구성

Messaging Server는 메일함에 대한 클라이언트 액세스를 위해 POP3(Post Office Protocol 3), IMAP4(Internet Mail Access Protocol 4) 및 HTTP(HyperText Transfer Protocol)를 지원합니다. IMAP 및 POP는 모두 인터넷 표준 메일함 프로토콜입니다. 웹 사용 가능 전자 메일 프로그램인 Messenger Express를 사용하면 최종 사용자는 HTTP를 사용하는 인터넷에 연결된 컴퓨터 시스템에서 실행 중인 브라우저를 통해 자신의 메일함에 액세스할 수 있습니다.

이 장에서는 명령줄 유틸리티를 사용하여 이러한 서비스를 하나 이상 지원하도록 서버를 구성하는 방법에 대해 설명합니다.

SMTP(Simple Mail Transfer Protocol) 서비스 구성에 대한 자세한 내용은 10 장을 참조하십시오.

이 장은 다음 내용으로 구성되어 있습니다.

5.1 일반 구성

Messaging Server POP, IMAP 및 HTTP 서비스의 일반 기능을 구성하는 작업에는 서비스를 활성화/비활성화하고 포트 번호를 할당하며 선택적으로 연결 클라이언트에 보내진 서비스 배너를 수정하는 것이 포함됩니다. 이 절에서는 이에 대한 배경 정보를 제공합니다. 이러한 설정을 위해 따라야 하는 단계는 125 페이지 “5.5 POP 서비스 구성”, 126 페이지 “5.6 IMAP 서비스 구성” 및 130 페이지 “5.7 HTTP 서비스 구성”을 참조하십시오. 이 절은 다음과 같은 하위 절로 구성되어 있습니다.

5.1.1 서비스 활성화/비활성화

Messaging Server의 특정 인스턴스가 POP, IMAP 또는 HTTP 서비스를 활성화할지 여부를 제어할 수 있습니다. 이것은 서비스를 시작 및 중지하는 것과 다릅니다(104 페이지 “4.4 서비스 시작 및 중지” 참조). POP, IMAP 또는 HTTP 서비스가 작동하려면 해당 서비스를 사용 가능하게 한 다음 시작해야 합니다.

서비스를 사용 가능하게 하는 것은 서비스를 시작 또는 중지하는 것보다 “전역적인” 과정입니다. 예를 들어, 사용 가능 설정은 시스템 재부트 후에도 계속 유지되지만 이전에 “중지된” 서비스는 재부트 후에 다시 시작해야 합니다.

사용할 계획이 없는 서비스를 사용 가능하게 할 필요가 없습니다. 예를 들어, Messaging Server 인스턴스를 단지 MTA(mail transfer agent)로 사용할 경우 POP, IMAP 및 HTTP를 사용 불가능하게 해야 합니다. 이 인스턴스가 POP 서비스에만 사용될 경우 IMAP 및 HTTP를 사용 불가능하게 하고웹 기반 전자 메일에만 사용될 경우 POP 및 IMAP를 사용 불가능하게 해야 합니다.

서버 수준에서 서비스를 사용 가능/불가능하게 할 수 있습니다. 이 프로세스는 이 장과 106 페이지 “4.4.2.1 시작할 서비스 지정”에 설명되어 있습니다. 또한 LDAP 속성 `mailAllowedServiceAccess`를 설정하여 사용자 수준에서 서비스를 활성화/비활성화할 수도 있습니다.

5.1.2 포트 번호 지정

각 서비스에 대해 서버가 서비스 연결에 사용할 포트 번호를 지정할 수 있습니다.

- POP 서비스를 사용 가능하게 한 경우 서버가 POP 연결에 사용할 포트 번호를 지정 할 수 있습니다. 기본값은 110입니다.
- IMAP 서비스를 사용 가능하게 한 경우 서버가 IMAP 연결에 사용할 포트 번호를 지정 할 수 있습니다. 기본값은 143입니다.
- HTTP 서비스를 사용 가능하게 한 경우 서버가 HTTP 연결에 사용할 포트 번호를 지정 할 수 있습니다. 기본값은 80입니다.

경우에 따라서는 기본값이 아닌 포트 번호를 지정해야 할 수 있습니다. 예를 들어 두 개 이상의 서버 인스턴스가 단일 호스트 시스템에 있거나 IMAP 서버 및 Messaging Multiplexor 서버와 동일한 호스트 시스템을 사용하는 경우 등입니다. Multiplexor에 대한 자세한 내용은 7 장을 참조하십시오.

포트를 지정할 때는 다음 사항을 유의하십시오.

- 포트 번호는 1에서 65535 사이의 모든 숫자가 될 수 있습니다.
- 선택한 포트가 이미 사용 중이거나 다른 서비스에 예약되지 않았는지 확인합니다.

5.1.3 암호화된 통신을 위한 포트

Messaging Server는 SSL(Secure Sockets Layer) 프로토콜을 사용하여 IMAP, POP 및 HTTP 클라이언트와의 암호화된 통신을 지원합니다. Messaging Server의 SSL 지원에 대한 일반 정보는 686 페이지 "23.5 암호화 및 인증서 기반 인증 구성"을 참조하십시오.

5.1.3.1 SSL을 통한 IMAP

SSL을 통한 IMAP에 대해 기본(권장) 포트 번호(993)를 그대로 사용하거나 별개의 포트를 지정할 수 있습니다.

대부분의 현재 IMAP 클라이언트에 별개의 포트가 필요하므로 Messaging Server는 IMAP 및 SSL을 통한 IMAP에 대해 별개의 포트를 사용하는 옵션을 제공합니다. IMAP 및 SSL을 통한 IMAP 모두에 동일한 포트를 사용하는 통신은 새로운 표준으로 등장하고 있습니다. Messaging Server에 설치된 SSL 인증서가 있을 경우(688 페이지 "23.5.1 인증서 얻기" 참조) SSL을 통한 동일 포트 IMAP를 지원할 수 있습니다.

5.1.3.2 SSL을 통한 POP

POP에 대한 기본적인 별도 SSL 포트는 995입니다. "STLS" 명령을 사용하여 일반적인 POP 포트를 통해 SSL을 초기화할 수도 있습니다(125 페이지 "5.5 POP 서비스 구성" 참조).

5.1.3.3 SSL을 통한 HTTP

SSL을 통한 HTTP에 대해 기본 포트 번호(443)를 그대로 사용하거나 HTTP에 대해 다른 포트를 지정할 수 있습니다.

5.1.4 서비스 배너

클라이언트가 POP 또는 IMAP 포트에 처음 연결되면 서버는 식별 텍스트 문자열 클라이언트에게 보냅니다. 일반적으로 클라이언트의 사용자에게 표시되지 않는 이 서비스 배너는 서버를 Sun Java System Messaging Server로 식별하고 서버의 버전 번호를 제공합니다. 이 배너는 대부분 클라이언트 디버깅 또는 문제 해결 목적에 사용됩니다.

연결 클라이언트에 다른 메시지를 보내려는 경우 POP 또는 IMAP 서비스의 기본 배너를 바꿀 수 있습니다.

`configutil` 유틸리티(`service.imap.banner`, `service.pop.banner`)를 사용하여 서비스 배너를 설정합니다. `configutil`에 대한 자세한 구문 정보는 **Sun Java System Messaging Server 6.3 Administration Reference**를 참조하십시오.

5.2 로그인 요구 사항

메일을 검색하기 위해 POP, IMAP 또는 HTTP 서비스에 로그인하는 사용자를 허용하는 방법을 제어할 수 있습니다. 모든 서비스에 대해 비밀번호 기반 로그인을 허용하거나 IMAP 또는 HTTP 서비스에 대해 인증서 기반 로그인을 허용할 수 있습니다. 이 절에서는 이에 대한 배경 정보를 제공합니다. 이러한 설정을 위해 따라야 하는 단계는 125 페이지 “5.5 POP 서비스 구성”, 126 페이지 “5.6 IMAP 서비스 구성” 또는 130 페이지 “5.7 HTTP 서비스 구성”을 참조하십시오. 또한 POP 로그인에 대한 유효한 로그인 구분자를 지정할 수도 있습니다. 이 절은 다음과 같은 하위 절로 구성되어 있습니다.

▼ POP 클라이언트에 대한 로그인 구분자 설정

일부 메일 클라이언트는 @을 로그인 구분자(즉, uid@domain과 같은 주소의 @)로 허용하지 않습니다. 이러한 클라이언트의 예로는 Netscape Messenger 4.76, Netscape Messenger 6.0, Windows 2000의 Microsoft Outlook Express 등을 들 수 있습니다. 이에 대한 해결 방법은 다음과 같습니다.

1 **다음 명령으로 +를 유효한 구분자로 만듭니다.**

```
configutil -o service.loginseparator -v "@+"
```

2 **@이 아니라 +를 로그인 구분자로 사용하여 로그인해야 한다는 것을 POP 클라이언트 사용자에게 알립니다.**

5.2.1 도메인 이름을 사용하지 않고 로그인 허용

일반적인 로그인의 경우 사용자는 사용자 아이디를 입력하고 구분자와 도메인 이름을 입력한 다음 비밀번호를 입력합니다. 그러나 설치 도중에 지정하는 기본 도메인의 사용자는 도메인 이름이나 구분자를 입력하지 않고 로그인할 수 있습니다.

사용자 아이디만 사용하여(즉, 도메인 이름과 구분자를 사용하지 않고) 다른 도메인의 사용자가 로그인할 수 있게 하려면 sasl.default.ldap.searchfordomain을 0으로 설정합니다. 사용자 아이디는 전체 디렉토리 트리에서 고유해야 한다는 것에 주의하십시오. 사용자 아이디가 고유하지 않을 경우 도메인 이름 없이 로그인할 수 없습니다.

로그인하기 위해 사용자가 입력해야 하는 속성을 수정할 수 있습니다. 예를 들어, 사용자가 전화 번호(telephoneNumber) 또는 직원 번호(employeeID)를 사용하여 로그인할 수 있게 하려면 configutil 매개 변수 sasl.default.ldap.searchfilter에서 정의하는

LDAP 검색을 변경합니다. 이 매개 변수는 `inetDomainSearchFilter` 도메인별 속성의 전역 기본 설정이며 동일한 구문을 사용합니다.

이 매개 변수에 대한 자세한 내용은 **Sun Java System Messaging Server 6.3 Administration Reference**를 참조하십시오.

5.2.2 비밀번호 기반 로그인

일반적인 메시징 설치에서는 사용자가 자신의 POP, IMAP 또는 HTTP 메일 클라이언트에 비밀번호를 입력함으로써 메일함에 액세스합니다. 클라이언트가 비밀번호를 서버에 보내면 서버는 이를 사용하여 사용자를 인증합니다. 사용자가 인증되면 서버는 액세스 제어 규칙에 기초하여 사용자에게 해당 서버에 저장된 특정 메일함에 대한 액세스를 허용할 것인지 결정합니다.

비밀번호 로그인을 허용할 경우 사용자는 비밀번호를 입력하여 POP, IMAP 또는 HTTP에 액세스할 수 있습니다. 비밀번호 기반 또는 SSL 기반 로그인은 POP 서비스를 위한 유일한 인증 방법입니다. 비밀번호는 LDAP 디렉토리에 저장됩니다. 디렉토리 정책에 따라 적용되는 비밀번호 정책(예: 최소 길이)이 결정됩니다.

IMAP 또는 HTTP 서비스에 대한 비밀번호 로그인을 허용하지 않을 경우 비밀번호 기반 인증이 허용되지 않습니다. 이 경우 사용자는 다음 절에 설명된 대로 인증서 기반 로그인을 사용해야 합니다.

IMAP 및 HTTP 서비스에 대한 비밀번호 전송의 보안을 향상시키려면 비밀번호를 서버로 보내기 전에 암호화하도록 요구할 수 있습니다. 이렇게 하려면 로그인에 대한 최소 암호화 길이 요구 사항을 선택합니다.

- 0을 선택한 경우 암호화가 필요하지 않습니다. 비밀번호는 일반 텍스트로 보내지거나 클라이언트 정책에 따라 암호화됩니다
- 0이 아닌 값을 선택한 경우 클라이언트는 지정된 값 이상의 키 길이를 가진 암호화를 사용하여 서버와의 SSL 세션을 설정해야 합니다. 따라서 클라이언트가 보내는 모든 IMAP 또는 HTTP 사용자 비밀번호가 암호화됩니다.

서버에서 지원하는 최대값보다 큰 키 길이로 암호화하도록 클라이언트가 구성되었거나 클라이언트가 지원하는 최대값보다 큰 키 길이로 암호화하도록 서버가 구성된 경우 비밀번호 기반 로그인을 수행할 수 없습니다. 다양한 암호화 및 키 길이를 지원하도록 서버를 설정하는 방법에 대한 자세한 내용은 697 페이지 “23.5.2 SSL 사용 및 암호문 선택”을 참조하십시오.

5.2.3 인증서 기반 로그인

비밀번호 기반 인증 외에도 Sun Java System 서버는 디지털 인증서 검사를 통한 사용자 인증을 지원합니다. 이 경우, 클라이언트는 서버와의 SSL 세션을 설정할 때 비밀번호를 제공하는 대신 사용자의 인증서를 제공합니다. 인증서가 검증될 경우 사용자는 인증된 것으로 간주됩니다.

IMAP 또는 HTTP 서비스에 대한 인증서 기반 사용자 로그인을 허용하도록 Messaging Server를 설정하는 방법에 대한 자세한 내용은 699 페이지 “23.5.3 인증서 기반 로그인 설정”을 참조하십시오.

인증서 기반 로그인을 설정하는 데 필요한 작업을 수행한 경우 비밀번호 기반 로그인 및 인증서 기반 로그인이 모두 지원됩니다. 그런 다음 클라이언트가 SSL 세션을 설정하고 인증서를 제공할 경우 인증서 기반 로그인이 사용됩니다. 클라이언트는 SSL을 사용하지 않으며 클라이언트 인증서를 제공하지 않을 경우에는 대신 비밀번호를 보냅니다.

5.3 성능 매개 변수

Messaging Server의 POP, IMAP 및 HTTP 서비스에 대한 몇 가지 기본 성능 매개 변수를 설정할 수 있습니다. 하드웨어 용량과 사용자 기반에 따라 이러한 매개 변수를 조정하여 최대한의 서비스 효율성을 실현할 수 있습니다. 이 절에서는 이에 대한 배경 정보를 제공합니다. 이러한 설정을 위해 따라야 하는 단계는 125 페이지 “5.5 POP 서비스 구성”, 126 페이지 “5.6 IMAP 서비스 구성” 또는 130 페이지 “5.7 HTTP 서비스 구성”을 참조하십시오. 이 절은 다음과 같은 하위 절로 구성되어 있습니다.

5.3.1 프로세스 수

Messaging Server는 여러 실행 프로세스 간에 작업을 분할할 수 있으며 이렇게 하면 경우에 따라 효율성이 향상될 수 있습니다. 이 기능은 특히 서버 프로세스 수를 조정했을 때 하드웨어 프로세서 간에 여러 작업을 더 효율적으로 분산시킬 수 있는 다중 프로세스 서버 시스템에서 유용합니다.

그러나 여러 프로세스 간에 작업을 할당하고 특정 프로세스에서 다른 프로세스로 전환하는 것에는 성능 오버헤드가 존재합니다. 여러 프로세스를 사용하는 자체의 이점은 새 프로세스를 추가할수록 줄어듭니다. 대부분의 구성에 적용되는 간단한 경험상의 규칙은 서버 시스템의 하드웨어 프로세서당 프로세스를 하나씩 가지는 것입니다(최대한 네 개까지 가능). 실제의 최적 구성이 이와 다를 수 있으므로 이 경험상의 규칙을 단순히 고유한 분석을 위한 지침으로 활용해야 할 것입니다.

주: 일부 플랫폼에서는 성능에 영향을 줄 수 있는 플랫폼 특정의 일정한 프로세스별 제한(예: 최대 파일 설명자 수)을 극복하기 위해 프로세스 수를 늘릴 수 있습니다.

각 POP, IMAP 또는 HTTP 서비스에 대한 기본 프로세스 수는 1입니다.

5.3.2 프로세스당 연결 수

POP, IMAP 또는 HTTP 서비스가 유지 관리할 수 있는 동시 클라이언트 연결이 많아질수록 클라이언트에게 더 유리합니다. 사용할 수 있는 연결이 없기 때문에 서비스가 거부될 경우 클라이언트는 다른 클라이언트가 연결을 끊을 때까지 기다려야 합니다.

반면, 열려 있는 각 연결은 메모리 자원을 소비하며 서버 시스템의 입출력 하위 시스템에 대한 요청을 하기 때문에 서버가 지원하리라 예상할 수 있는 동시 세션 수에는 실제적인 제한이 있습니다. 서버 메모리나 입출력 용량을 증가시켜 이러한 제한을 늘릴 수도 있습니다.

IMAP, HTTP 및 POP는 이 점에 있어서 요구 사항이 다릅니다.

- IMAP 연결은 일반적으로 POP 및 HTTP 연결과 비교하여 오래 지속됩니다. 사용자가 IMAP에 연결하여 메시지를 다운로드할 경우 사용자가 종료하거나 연결 시간이 초과될 때까지 일반적으로 연결이 유지 관리됩니다. 이와 달리 POP 또는 HTTP 연결은 대개 POP 또는 HTTP 요청이 서비스되자마자 닫힙니다.
- IMAP 및 HTTP 연결은 일반적으로 POP 연결과 비교하여 매우 효율적입니다. 각 POP 재연결에는 사용자에 대한 재인증이 필요합니다. 이와 달리 IMAP 연결은 IMAP 세션 동안(로그인에서 로그아웃까지) 연결이 열려 있기 때문에 단일 인증만 필요합니다. HTTP 연결은 짧지만 각 HTTP 세션(로그인에서 로그아웃까지)에 여러 연결이 허용되므로 각 연결에 대해 사용자의 재인증이 필요하지 않습니다. 이러한 점에서 POP 연결은 IMAP 또는 HTTP 연결보다 훨씬 더 많은 성능 오버헤드를 발생시킵니다. Messaging Server는 특히, 열려 있지만 유휴 상태인 IMAP 연결과 여러 HTTP 연결을 통해 매우 낮은 오버헤드만이 필요하도록 설계되었습니다.

주 – HTTP 세션 보안에 대한 자세한 내용은 680 페이지 “23.2 HTTP 보안 정보”를 참조하십시오.

따라서 특정 시점의 특정 사용자 요구에 대해 Messaging Server는 POP 연결보다 더 많은 열려 있는 IMAP 및 HTTP 연결을 지원할 수 있습니다.

IMAP의 기본값은 4000이고 HTTP의 기본값은 프로세스당 6000개의 연결이며 POP의 기본값은 600입니다. 이러한 값은 일반적으로 구성된 서버 시스템이 처리할 수 있는 대략적으로 동일한 요구를 나타냅니다. 최적 구성이 이와 다를 수 있으므로 이러한 기본값을 단순히 일반적인 지침으로 사용해야 합니다.

일반적으로 활성 POP 연결은 활성 IMAP 연결보다 서버 리소스와 대역폭이 훨씬 더 많이 요구됩니다. 이는 IMAP 연결이 대부분의 시간에 유휴 상태인 것과 달리 POP 연결은 지속적으로 메시지를 다운로드하기 때문입니다. 따라서 POP에 대해 더 적은 수의 세션을 유지하는 것이 적합합니다. 반대로 POP 연결은 전자 메일을 다운로드하는 동안에만 지속되므로 활성 POP 사용자는 짧은 시간 동안만 연결되지만 IMAP 연결은 계속되는 메일 검사에서 연결된 상태로 유지됩니다.

5.3.3 프로세스당 스레드 수

여러 프로세스를 지원하는 것 외에도 Messaging Server는 여러 스레드 간에 작업을 분할하여 성능을 더욱 향상시킵니다. 서버의 스레드 사용은 실행 효율성을 크게 향상시키는데 이는 진행 중인 명령이 다른 명령의 실행을 저해하지 않기 때문입니다. 실행하는 동안에 필요에 따라 스레드는 설정된 최대 개수까지 작성 및 삭제됩니다.

동시에 실행되는 스레드가 많다는 것은 더 많은 클라이언트 요청을 지연 없이 처리할 수 있으며 이에 따라 더 많은 수의 클라이언트에게 신속하게 서비스할 수 있다는 것을 의미합니다. 그러나 스레드 간의 디스패칭으로 인해 성능 오버헤드가 발생하므로 서버가 사용할 수 있는 스레드 수에는 실제적인 제한이 존재합니다.

POP, IMAP 및 HTTP의 경우 기본 최대값은 프로세스당 250개의 스레드입니다. IMAP 및 HTTP의 기본 연결이 POP보다 많다는 사실에도 불구하고 이러한 기본값은 동일합니다. 수는 적지만 사용량이 많은 POP 연결과 동일한 최대 스레드 수를 사용하여 POP 연결보다 많은 수의 IMAP 및 HTTP 연결을 효율적으로 처리할 수 있는 것으로 알려져 있습니다. 실제의 최적 구성이 이와 다를 수 있지만 이러한 기본값으로 충분하기 때문에 값을 늘릴 필요는 없을 것입니다. 즉, 대부분의 설치에서 이러한 기본값은 적절한 성능을 제공합니다.

5.3.4 유휴 연결 해제

응답하지 않는 클라이언트의 연결에 사용된 시스템 자원을 재이용하기 위해 IMAP4, POP3 및 HTTP 프로토콜은 일정 시간 동안 유휴 상태였던 연결을 일방적으로 해제할 수 있는 기능을 서버에 제공합니다.

각 프로토콜 사양에서는 서버가 최소한의 시간 동안 유휴 연결을 열어두어야 합니다. 기본 시간은 POP는 10분, IMAP는 30분, HTTP는 3분입니다. 유휴 시간을 기본값보다 큰 값으로 늘릴 수 있지만 줄일 수는 없습니다.

POP 또는 IMAP 연결이 해제될 경우 새 연결을 설정하기 위해 사용자는 재인증되어야 합니다. 이와 달리 HTTP 연결이 해제될 경우 HTTP 세션이 계속 열려 있으므로 사용자를 재인증할 필요가 없습니다. HTTP 세션 보안에 대한 자세한 내용은 680 페이지 “23.2 HTTP 보안 정보”를 참조하십시오.

유휴 POP 연결은 일반적으로 클라이언트를 응답하지 않게 만드는 일부 문제(예: 충돌 또는 중지)로 인해 발생합니다. 반면, 유휴 IMAP 연결은 정상적인 상태입니다. IMAP 사용자가 일방적으로 연결이 끊기는 것을 방지하기 위해 IMAP 클라이언트는 일반적으로 30초 미만의 일정한 간격으로 IMAP 서버에 명령을 보냅니다.

5.3.5 HTTP 클라이언트 로그아웃

HTTP 세션은 여러 연결에서 지속될 수 있습니다. 연결이 해제될 때 HTTP 클라이언트는 로그아웃되지 않습니다. 그러나 지정된 기간(기본적으로 2시간) 동안 HTTP 세션이 유휴 상태이면 서버는 자동으로 HTTP 세션을 해제하고 클라이언트는 로그아웃됩니다.

세션이 해제되면 클라이언트의 세션 아이디가 더 이상 유효하지 않으므로 다른 세션을 설정하기 위해 클라이언트는 재인증되어야 합니다. HTTP 보안 및 세션 아이디에 대한 자세한 내용은 680 페이지 “23.2 HTTP 보안 정보”를 참조하십시오.

5.4 클라이언트 액세스 제어

Messaging Server에는 POP, IMAP 또는 HTTP 메시징 서비스와 SMTP에 대한 액세스를 어떤 클라이언트가 얻을 수 있는지 결정하는 액세스 제어 기능이 포함되어 있습니다. 다양한 기준에 기초하여 클라이언트에 대한 액세스를 허용 또는 거부하는 유연한 액세스 필터를 만들 수 있습니다.

클라이언트 액세스 제어는 Messaging Server의 중요한 보안 기능입니다. 클라이언트 액세스 제어 필터 작성에 대한 자세한 내용과 그 사용 예는 703 페이지 “23.7 POP, IMAP 및 HTTP 서비스에 대한 클라이언트 액세스 구성” 및 715 페이지 “23.9 SMTP 서비스에 대한 클라이언트 액세스 구성”을 참조하십시오.

5.5 POP 서비스 구성

`configutil` 명령을 사용하여 Messaging Server POP 서비스의 기본 구성을 수행할 수 있습니다. 이 절에서는 보다 일반적인 몇 가지 POP 서비스 옵션이 제공됩니다. 전체 목록은 **Sun Java System Messaging Server 6.3 Administration Reference**의 “configutil Parameters”에서 확인할 수 있습니다.

주 – POP 서비스의 경우 비밀번호 기반 로그인이 자동으로 사용 가능하게 됩니다.

자세한 내용은 다음을 참조하십시오.

- 118 페이지 “5.1.1 서비스 활성화/비활성화”
- 120 페이지 “POP 클라이언트에 대한 로그인 구분자 설정”
- 118 페이지 “5.1.2 포트 번호 지정”
- 123 페이지 “5.3.2 프로세스당 연결 수”
- 124 페이지 “5.3.4 유휴 연결 해제”
- 124 페이지 “5.3.3 프로세스당 스레드 수”
- 122 페이지 “5.3.1 프로세스 수”

POP 서비스를 사용 또는 사용하지 않으려면 다음을 수행합니다.

```
configutil -o service.pop.enable -v [ yes | no ]
```

포트 번호를 지정하려면 다음을 수행합니다.

```
configutil -o service.pop.port -v number
```

프로세스당 최대 네트워크 연결 수 설정 방법(123 페이지 “5.3.2 프로세스당 연결 수” 참조):

```
configutil -o service.pop.maxsessions -v number
```

연결에 대한 최대 유휴 시간 설정(124 페이지 “5.3.4 유휴 연결 해제” 참조):

```
configutil -o service.pop.idletimeout -v number
```

프로세스당 최대 스레드 수 설정(124 페이지 “5.3.3 프로세스당 스레드 수” 참조):

```
configutil -o service.pop.maxthreads -v number
```

최대 프로세스 수 설정 방법(122 페이지 “5.3.1 프로세스 수” 참조):

```
configutil -o service.pop.numprocesses -v number
```

SSL에서의 POP를 사용하려면 다음을 수행합니다.

```
configutil -o service.pop.enablesslport -v 1
configutil -o service.pop.sslport -v 995
```

SSL이 올바르게 구성된 경우 TLS도 지원됩니다.

프로토콜 시작 배너를 지정하려면 다음을 수행합니다.

```
configutil -o service.pop.banner -v banner
```

5.6 IMAP 서비스 구성

`configutil` 명령을 사용하여 Messaging Server IMAP 서비스의 기본 구성을 수행할 수 있습니다. 이 절에서는 더 일반적인 몇 가지 IMAP 서비스 옵션이 제공됩니다. 전체 목록은 **Sun Java System Messaging Server 6.3 Administration Reference**의 3 장, “Messaging Server Configuration”에서 확인할 수 있습니다. 자세한 내용은 다음을 참조하십시오.

- 118 페이지 “5.1.1 서비스 활성화/비활성화”
- 118 페이지 “5.1.2 포트 번호 지정”
- 121 페이지 “5.2.2 비밀번호 기반 로그인”
- 123 페이지 “5.3.2 프로세스당 연결 수”
- 124 페이지 “5.3.4 유휴 연결 해제”
- 124 페이지 “5.3.3 프로세스당 스레드 수”
- 122 페이지 “5.3.1 프로세스 수”
- 127 페이지 “5.6.1 IMAP IDLE 구성”

명령줄: 다음과 같이 명령줄에서 IMAP 속성에 대한 값을 설정할 수 있습니다.

IMAP 서비스를 사용 또는 사용하지 않으려면 다음을 수행합니다.

```
configutil -o service.imap.enable -v [ yes | no ]
```

포트 번호를 지정하려면 다음을 수행합니다.

```
configutil -o service.imap.port -v number
```

SSL을 통한 IMAP에 별개의 포트를 사용하려면 다음을 수행합니다.

```
configutil -o service.imap.enablesslport -v [ yes | no ]
```

SSL을 통한 IMAP에 사용할 포트 번호를 지정하려면 다음을 수행합니다.

```
configutil -o service.imap.sslport -v number
```

IMAP 서비스에 대한 비밀번호 로그인을 사용 또는 사용하지 않으려면 다음을 수행합니다.

```
configutil -o service.imap.plaintextmincipher -v value
```

*value*가 > 0이면 보안 계층(SSL 또는 TLS)이 활성화되지 않은 경우 일반 텍스트 비밀번호를 사용할 수 없게 됩니다. 사용자는 로그인하려면 네트워크상에서 비밀번호가 공개되는 것을 방지하는 SSL 또는 TLS를 클라이언트에서 사용 가능하게 해야 합니다. 기본값은 0입니다.

프로세스당 최대 네트워크 연결 수 설정(123 페이지 “5.3.2 프로세스당 연결 수” 참조):

```
configutil -o service.imap.maxsessions -v number
```

연결에 대한 최대 유휴 시간 설정(124 페이지 “5.3.4 유휴 연결 해제” 참조):

```
configutil -o service.imap.idletimeout -v number
```

프로세스당 최대 스레드 수 설정(124 페이지 “5.3.3 프로세스당 스레드 수” 참조):

```
configutil -o service.imap.maxthreads -v number
```

최대 프로세스 수 설정(122 페이지 “5.3.1 프로세스 수” 참조):

```
configutil -o service.imap.numprocesses -v number
```

프로토콜 시작 배너를 지정하려면 다음을 수행합니다.

```
configutil -o service.imap.banner -v banner
```

5.6.1 IMAP IDLE 구성

RFC 2177에 정의된 IMAP 지정에 대한 IMAP IDLE 확장을 사용하여 IMAP 서버는 새 메시지가 도착하거나 사용자의 메일함에 다른 업데이트가 적용될 경우 메일 클라이언트에게 알립니다. IMAP IDLE 기능의 장점은 다음과 같습니다.

- 메일 클라이언트가 받는 메일에 대해 IMAP 서버를 폴링할 필요가 없습니다.

  클라이언트가 폴링할 필요가 없으므로 IMAP 서버에서 작업 로드가 감소되고 서버의 성능이 향상됩니다. 클라이언트 폴링은 사용자에게 전달되는 메시지가 얼마되지 않거나 전혀 없는 경우 매우 낭비적입니다. 이 경우에도 클라이언트가 구성된 간격(일반적으로 5 또는 10분)에 따라 계속해서 폴링합니다.
- 메일 클라이언트는 새 메시지가 사용자의 메일함에 도착하면 거의 즉시 사용자에게 메시지를 표시합니다. 메시지 상태 변경도 거의 실시간으로 표시됩니다.

  IMAP 서버는 다음 IMAP 폴링 메시지를 대기할 필요 없이 새 메일 메시지나 업데이트된 메일 메시지를 클라이언트에게 즉시 알릴 수 있습니다. 대신 IMAP 서버는 새 메시지가 도착하거나 메시지 상태가 변경되면 즉시 알림을 받습니다. 그런 다음 서버는 IMAP 프로토콜을 통해 클라이언트에게 알립니다.

5.6.1.1 전제 조건

IMAP IDLE 기능은 ENS(Event Notification Service)를 사용하여 알림을 전파합니다. IMAP IDLE를 사용하려면 다음 ENS 구성 요소를 구성해야 합니다.

- 하나 이상의 호스트에서 `enpd` 서버 구성
- 모든 메시지 저장소 호스트에서 `iBiff` 알림 플러그 인 구성

Messaging Server에 대한 ENS 구성 방법에 대한 자세한 내용은 **Sun Java System Communications Services Event Notification Service Guide**를 참조하십시오.

`iBiff` 알림 플러그 인 구성에 대한 자세한 내용은 875 페이지 “B.1 Messaging Server에서 ENS Publisher 로드”을 참조하십시오.

▼ IMAP IDLE 구성 방법

1 **메시지 저장소를 실행하는 호스트의 연결만 허용하도록 `enpd` 서버를 구성합니다.**

연결을 메시지 저장소 호스트로 제한하려면 `ENS_ACCESS` 환경 변수를 설정합니다. 환경 변수는 `enpd`에 액세스할 수 있는 권한 목록을 설정합니다. 구문은 다음과 같습니다.

```
setenv ENS_ACCESS 'allowdeny ipaddress|mask;
allowdeny ipaddress|mask; ...'
```

여기서

allowdeny +(허용하도록 지정) 또는 —(거부하도록 지정)

ipaddress 점으로 구분된 십진수 형식의 IP 주소 지정

mask 점으로 구분된 십진수 형식의 IP 주소 마스크 지정

예:

다음 예에서는 로컬 호스트에만 액세스할 수 있습니다.

```
setenv ENS_ACCESS '+127.0.0.1|255.255.255.255'
```

다음 예에서는 로컬 호스트와 모든 IP 주소 192.168.0.*(192.168.0.17 제외)에 액세스할 수 있습니다.

```
setenv ENS_ACCESS '+192.168.0.1|255.255.255.0;+127.0.0.1|255.255.255.255; \
-192.168.0.17;255.255.255.255'
```

2 configutil 유틸리티를 실행하여 ENS 서버가 실행되고 있는 호스트의 이름을 지정합니다.

```
cd msg-svr-base
./configutil -o local.store.notifyplugin.enshost -v "ipaddress"
```

여기서 *ipaddress*는 ENS 호스트 시스템의 점으로 구분된 십진수 IP 주소를 지정합니다.

예:

```
cd msg-svr-base
./configutil -o local.store.notifyplugin.enshost -v "127.0.0.1"
```

3 ENS 알림에 사용할 이벤트 키를 지정합니다.

ENS 이벤트 키(ensEventKey)가 기본값으로 설정되어 있는 경우 IMAP IDLE가 작동하지 않습니다.

ensEventKey 값을 %M으로 끝나도록 구성해야 합니다. %M 문자열은 이벤트가 발생한 메일함의 이름으로 교체되는 대체 코드입니다.

다음 configutil 명령을 실행합니다.

```
./configutil -o local.store.notifyplugin.enseventkey -v "eventkey"
```

여기서 *eventkey*는 ENS에 사용되는 고유한 식별자입니다. 기본값은 enp://127.0.0.1/store입니다. 이벤트 키의 호스트 이름 부분은 ENS가 실행 중인 호스트를 결정하는 데 사용되지 않으며, 식별자의 일부일 뿐입니다.

예:

```
./configutil -o local.store.notifyplugin.enseventkey -v "enp://127.0.0.1/store/%M"
```

4 libibiff 알림 플러그 인 파일을 로드하여 Messaging Server용 ENS Publisher를 활성화합니다.

다음 configutil 명령을 실행합니다.

```
./configutil -o local.store.notifyplugin -v "msg-svr-base/lib/libibiff"
```

5 받는 메일함이 아니라 모든 사용자 메일함에서 알림을 전송할 수 있습니다.

기본적으로 알림은 받은 메일함에서 발생한 이벤트에 의해서만 생성됩니다. 그러나 IMAP IDLE RFC(2177)에는 모든 메일함에서 이벤트가 발생할 때마다 IDLE에서 클라이언트에게 알리도록 지정되어 있습니다.

RFC를 준수하려면 IMAP IDLE 기능에서 모든 메일함에 대해 알림을 활성화해야 합니다. 그렇지 않으면 IMAP 서버가 IDLE 기능을 광고하지 않습니다.

모든 메일함에 대해 알림을 구성하려면 `configutil` 명령 `noneinbox` 값을 1로 설정합니다.

```
./configutil -o local.store.notifyplugin.noneinbox.enable -v 1
```

여기서 `-v 1`은 모든 메일함에서 알림을 활성화합니다.

6 Messaging Server를 중지하고 다시 시작합니다.

```
cd msg-svr-base/sbin

./stop-msg

./start-msg
```

7 IMAP 서비스에 IDLE 기능이 포함되어 있는지 확인합니다. 텔넷을 사용하여 IMAP 호스트 및 포트에 연결합니다.

```
telnet IMAP_hostname port
```

예:

```
telnet myhost imap
trying 192.18.01.44 ...
connected to myhost.siroe.com

* OK [CAPABILITY IMAP4 IMAP4rev1 ACL QUOTA LITERAL+ NAMESPACE UIDPLUS
CHILDREN BINARY UNSELECT SORT LANGUAGE STARTTLS IDLE XSENDER X-NETSCAPE
XSERVERINFO X-SUN-SORT X-SUN-IMAP X-ANNOTATEMORE AUTH=PLAIN]
myhost.siroe.com IMAP4 service (Sun Java(tm) System
Messaging Server 6.3-0.05 (built Feb 7 2006))
```

5.7 HTTP 서비스 구성

Messaging Server는 Messenger Express 및 Communications Express라는 HTTP 메일 클라이언트를 지원합니다. POP 및 IMAP 클라이언트는 라우팅 또는 전달을 위해 Messaging Server MTA에 메일을 직접 보내지만 HTTP 클라이언트는 Webmail Server(mshttpd 또는 Messaging Server http 데몬이라고도 함)라는 특수 웹 서버에 메일을 보냅니다. 메시지의 대상에 따라 Webmail Server는 라우팅을 위해 아웃바운드 MTA에 메일을 직접 보내거나 IMAP를 사용하여 백엔드 메시지 저장소 중 하나에 메일을 보냅니다. 이는 그림 5–1에 표시되어 있습니다. Communications Express Server는 Webmail Server를 통해 요청을 라우팅합니다.

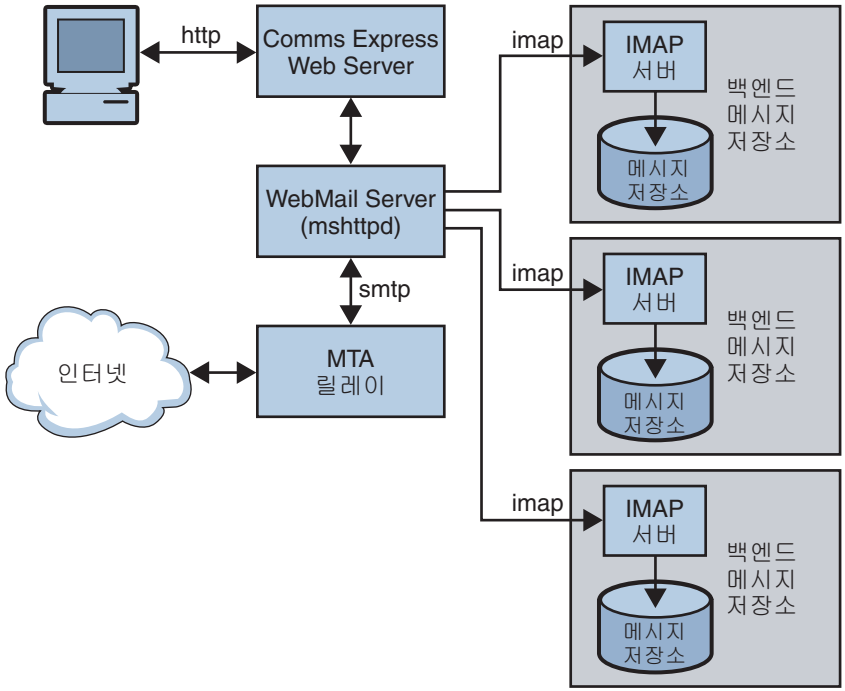


그림 5–1 HTTP 서비스 구성 요소

이전 버전에서는 Webmail Server가 메시지 저장소에 직접 액세스했습니다. 현재 버전에서는 IMAP 서버를 통해 메시지 저장소에 액세스합니다. 이 방법은 다음과 같은 장점이 있습니다.

- Messenger Express 및 Communications Express 클라이언트가 서로 다른 백엔드 메시지 저장소에 있는 공유 폴더에 액세스할 수 있습니다.
- Webmail Server를 각 백엔드 서버에 더 이상 설치할 필요가 없습니다.
- Webmail Server가 이전에는 MEM(Messenger Express Multiplexor)에서 수행했던 프런트엔드 서버 역할을 할 수 있습니다.
- MEM은 폐기되었으며 더 이상 사용되지 않습니다.
- 클라이언트 측에서는 사용자가 메시지 저장소에 없는 공유 폴더에 액세스할 수 있다는 점을 제외하고 변경된 사항이 없습니다.

이전 버전에서는 MEM이 HTTP 클라이언트 요청을 받아서 백엔드 메시지 저장소의 해당 Webmail Server에 전달합니다. 따라서 mshttpd 복사본을 모든 백엔드 서버에 설치해야 했습니다. 이 버전에서는 Webmail Server가 HTTP 클라이언트 전자 메일 요청을 받는 프런트엔드 서버 역할을 합니다. 또한 이러한 요청을 SMTP 또는 IMAP 호출로 변환한 다음 백엔드 메시지 저장소의 해당 IMAP 서버나 MTA에게 전달합니다. Messaging Server가 웹 기반 전자 메일에만 사용되는 경우 IMAP를 활성화해야 합니다.

5.7.1 HTTP 서비스 구성

대부분의 HTTP 구성 매개 변수는 POP 및 IMAP 서비스에 사용할 수 있는 매개 변수와 비슷합니다. 여기에는 연결 설정 및 프로세스 설정을 위한 매개 변수가 포함됩니다. 이 절에서는 더 일반적인 몇 가지 HTTP 서비스 옵션이 제공됩니다. 전체 목록은 **Sun Java System Messaging Server 6.3 Administration Reference**의 "configutil Parameters"에서 확인할 수 있습니다. 자세한 내용은 다음을 참조하십시오.

- 118 페이지 "5.1.1 서비스 활성화/비활성화"
- 118 페이지 "5.1.2 포트 번호 지정"
- 121 페이지 "5.2.2 비밀번호 기반 로그인"
- 123 페이지 "5.3.2 프로세스당 연결 수"
- 124 페이지 "5.3.4 유휴 연결 해제"
- 124 페이지 "5.3.5 HTTP 클라이언트 로그아웃"
- 124 페이지 "5.3.3 프로세스당 스레드 수"
- 122 페이지 "5.3.1 프로세스 수"

사용자가 액세스하는 각 IMAP 서버에 대해 Webmail Server는 IMAP 포트, SSL 사용 여부 및 사용자 로그인에 사용할 관리자 자격 증명을 알고 있어야 합니다. 이 작업을 수행하는 `configutil` 매개 변수는 다음과 같습니다.

`local.service.proxy.imapport[.`*hostname*`]` — 연결할 IMAP 포트입니다(기본값:143).

`local.service.proxy.imapssl` — SSL를 활성화합니다(기본값: 없음).

`local.service.proxy.admin[.`*hostname*`]` — 관리자 아이디입니다.

`local.service.proxy.adminpass[.`*hostname*`]` — 관리자 비밀번호입니다.

이러한 매개 변수는 전역적(모든 IMAP 백엔드 서버에 적용)으로 설정하거나, 옵션 이름에 백엔드의 정규화된 도메인 이름을 추가하여 IMAP 백엔드 서버별로 설정할 수 있습니다.

SSL을 통한 IMAP를 사용하려면 mshttpd를 SSL HTTP 서버로 구성하고 mshttpd 인증서 데이터베이스가 IMAP 백엔드의 CA를 신뢰해야 합니다. 또한 `service.http.sslusessl`을 활성화해야 합니다. IMAP를 실행하는 백엔드 메시지 저장소에서 자체 서명된 인증서(예: `generate-certDB`를 통해 생성된 인증서)를 사용하는 경우 해당 인증서를 프런트엔드 `mshttpd` 데몬 프록시에 추가해야 합니다.

`local.service.proxy.admin/pass`를 설정하지 않은 경우 로그인이 거부되고 **메일 서버를 사용할 수 없습니다.**라는 오류 메시지가 발생합니다. **관리자는 서버 로그에서 자세한 내용을 확인하십시오.**또한 http 로그에 누락된 구성 옵션이 나열됩니다.

명령줄에서 추가 HTTP 속성 값을 다음과 같이 설정할 수 있습니다.

HTTP 서비스를 활성화하거나 비활성화하려면 다음을 수행합니다.

```
configutil -o service.http.enable -v [ yes | no ]
```

기본적으로 HTTP 서비스는 라우팅 또는 전달을 위해 보내는 웹 메일을 로컬 MTA로 전송합니다. 예를 들어, 사이트가 호스팅 서비스이며 대부분의 수신자가 로컬 호스트 시스템과 다른 도메인에 있을 경우 메일을 원격 MTA로 보내도록 HTTP 서비스를 구성할 수 있습니다. 웹 메일을 원격 MTA로 보내려면 원격 호스트의 이름과 SMTP 포트 번호를 지정해야 합니다. 포트 번호를 지정하려면 다음을 수행합니다.

```
configutil -o service.http.port -v number
```

SSL을 통한 HTTP에 별개의 포트를 사용하려면 다음을 수행합니다.

```
configutil -o service.http.enablesslport -v [ yes | no ]
```

SSL을 통한 HTTP에 사용할 포트 번호를 지정하려면 다음을 수행합니다.

```
configutil -o service.http.sslport -v number
```

비밀번호 기반 로그인을 활성화하거나 비활성화하려면 다음을 수행합니다.

```
configutil -o service.http.plaintextmincipher -v value
```

*value*가 > 0이면 보안 계층(SSL 또는 TLS)이 활성화되지 않은 경우 일반 텍스트 비밀번호를 사용할 수 없게 됩니다. 사용자는 로그인하려면 네트워크상에서 비밀번호가 공개되는 것을 방지하는 SSL 또는 TLS를 클라이언트에서 사용 가능하게 해야 합니다. 기본값은 0입니다.

프로세스당 최대 네트워크 연결 수 설정(123 페이지 “5.3.2 프로세스당 연결 수” 참조):

```
configutil -o service.http.maxsessions -v number
```

연결에 대한 최대 유휴 시간 설정(124 페이지 “5.3.4 유휴 연결 해제” 참조)

```
configutil -o service.http.idletimeout -v number
```

클라이언트 세션에 대한 최대 유휴 시간 설정(124 페이지 “5.3.5 HTTP 클라이언트 로그아웃” 참조):

```
configutil -o service.http.sessiontimeout -v number
```

프로세스당 최대 스레드 수를 설정하려면 다음을 수행합니다.

```
configutil -o service.http.maxthreads -v number
```

최대 프로세스 수를 설정하려면 다음을 수행합니다.

```
configutil -o service.http.numprocesses -v number
```

HTTP 클라이언트가 첨부 파일이 있는 메시지를 생성하면 첨부 파일은 서버로 업로드되어 파일에 저장됩니다. HTTP 서비스는 라우팅 또는 전달을 위해 메시지를 MTA로 보내기 전에 첨부 파일을 검색하고 메시지를 생성합니다. 기본 첨부 파일 스풀 디렉토리를 사용하거나 대체 디렉토리를 지정할 수 있습니다. 또한 첨부 파일에 허용되는 최대 크기를 지정할 수도 있습니다. 클라이언트에 보내는 메일의 첨부 파일 스풀 디렉토리를 지정하려면 다음 명령을 사용합니다. 여기에는 base64로 인코딩된 모든 첨부 파일이 포함되며 base64 인코딩은 33%의 추가 공간이 필요하다는 점을 유의하십시오. 따라서 매개 변수의 제한이 5MB일 경우 메시지 하나와 첨부 파일의 최대 크기는 약 3.75MB가 됩니다.

`configutil -o service.http.spooldir -v` *dirpath*

최대 메시지 크기를 지정하려면 다음을 수행합니다.

`configutil -o service.http.maxmessagesize -v` *size*

여기에서 *size*는 바이트 수입니다. 여기에는 base64로 인코딩된 모든 첨부 파일이 포함되며 base64 인코딩은 33%의 추가 공간이 필요하다는 점을 유의하십시오. 따라서 매개 변수의 제한이 5MB일 경우 메시지 하나와 첨부 파일의 최대 크기는 약 3.75MB가 됩니다.

대체 MTA 호스트 이름을 지정하려면 다음을 수행합니다.

`configutil -o service.http.smtphost -v` *hostname*

대체 MTA 호스트 이름의 포트 번호를 지정하려면 다음을 수행합니다.

`configutil -o service.http.smtpport -v` *portnum*

◆ ◆ ◆ 6 장

6 단일 사인 온(SSO) 사용

단일 사인 온(SSO)은 최종 사용자가 한 번의 인증(사용자 아이디와 비밀번호를 사용하여 로그온)으로 여러 응용 프로그램에 액세스할 수 있는 기능입니다. Sun Java System Access Manager(이전 명칭은 Identity Server임)는 Sun Java System 서버에 대한 SSO에 사용되는 공식적인 게이트웨이입니다. 즉, 사용자가 다른 SSO 구성 서버에 액세스하려면 Access Manager에 로그인해야 합니다.

예를 들어, 제대로 구성된 경우 사용자는 Sun Java System Access Manager 로그인 화면에서 서명한 후 다른 창에서 다시 서명하지 않고도 Messenger Express에 액세스할 수 있습니다. 마찬가지로 Sun Java System Calendar Server를 제대로 구성한 경우 사용자는 Sun Java System Access Manager 로그인 화면에서 서명한 후 다른 창에서 다시 서명하지 않고도 해당 Calendar에 액세스할 수 있습니다.

Messaging Server에서는 SSO를 배포하는 두 가지 방법을 제공합니다. 첫 번째 방법은 Sun Java System Access Manager를 통한 방법이고 두 번째 방법은 통신 서버의 신뢰할 수 있는 원 기술을 통한 방법입니다. 신뢰할 수 있는 원을 사용하는 것은 레거시 SSO 구현 방법입니다. 이 방법은 Access Manager SSO에서는 사용할 수 없는 여러 기능을 제공하지만 이후의 모든 개발에서 Access Manager를 사용하게 될 것이므로 사용하지 않는 것이 좋습니다. 하지만 다음 절에서는 두 가지 방법을 모두 설명합니다.

6.1 Sun Java System 서버에 대한 Access Manager SSO

이 절에서는 Access Manager를 사용하는 SSO에 대해 설명합니다. 이 장은 다음 내용으로 구성되어 있습니다.

6.1.1 SSO 제한 사항 및 알림

- Messenger Express 세션은 Access Manager 세션이 유효한 경우에만 유효합니다. 사용자가 Access Manager에서 로그아웃하면 웹 메일 세션이 자동으로 닫힙니다(단일 사인 오프).
- 함께 작동하는 SSO 응용 프로그램은 동일한 DNS 도메인에 있어야 합니다. 이러한 도메인을 쿠키 도메인이라고도 합니다.
- SSO 응용 프로그램은 Access Manager 확인 URL(이름 지정 서비스)에 액세스해야 합니다.
- 브라우저에 쿠키가 있어야 합니다.

6.1.2 SSO를 지원하도록 Messaging Server 구성

네 개의 `configutil` 매개 변수가 Messaging Server SSO를 지원합니다. 이 네 개 중 `local.webmail.sso.amnamingurl` 하나만 Messaging Server에서 SSO를 사용하는 데 필요합니다. SSO를 사용하려면 이 매개 변수를 Access Manager에서 이름 지정 서비스를 실행하는 URL로 설정합니다. 일반적으로 이 URL은 http://*server*/amserver/namingservice입니다. 예:

```
configutil -o local.webmail.sso.amnamingurl -v
http://sca-walnut:88/amserver/namingservice
```

주 – Access Manager SSO는 이전 SSO 기법을 사용하는 `local.webmail.sso.enable`을 확인하지 않습니다. `local.webmail.sso.enable`을 `off` 또는 설정되지 않은 상태로 두어야 합니다. 그렇지 않으면 이전 SSO 기법에 필요한 구성 매개 변수가 없다는 경고 메시지가 기록됩니다.

`configutil` 명령을 사용하여 표 6–3에 표시된 SSO 구성 매개 변수를 수정할 수 있습니다.

표 6–1 Access Manager 단일 사인 온 매개 변수

매개 변수	설명
local.webmail.sso.amnamingurl	Access Manager가 이름 지정 서비스를 실행하는 URL입니다. Access Manager를 통해 단일 사인 온(SSO)하기 위해 필수적인 변수입니다. 일반적으로 이 URL은 `http://`*server*`/amserver/namingservice`입니다. 기본값: 설정 안 함

표 6–1 Access Manager 단일 사인 온 매개 변수 (계속)

매개 변수	설명
local.webmail.sso.amcookiename	Access Manager 쿠키 이름입니다. 기본적으로 Access Manager는 iPlanetDirectoryPro라는 쿠키에 세션 핸들을 저장합니다. Access Manager가 다른 쿠키 이름을 사용하도록 구성된 경우 Messaging Server에서 단일 사인 온(SSO) 수행 시 확인할 대상을 알 수 있도록 Messaging Server에서 해당 이름을 이 매개 변수로 구성해야 합니다. IS가 기본 구성으로 구성된 경우 기본값을 변경할 수 없습니다. 기본값: iPlanetDirectoryPro
local.webmail.sso.amloglevel	AMSDK 로깅 수준입니다. Messaging Server에 사용되는 SSO 라이브러리에는 Messaging Server와는 별도로 자체 로깅 기법이 있습니다. *msg-svr-base*/log의 http_sso라는 파일에 메시지가 기록됩니다. 기본적으로 *info* 이상의 로깅 수준을 가진 메시지만 기록되지만 로깅 수준을 1부터 5까지의 값(1 = errors, 2 = warnings, 3 = info, 4 = debug, 5 = maxdebug)으로 설정하여 로깅 수준을 높일 수 있습니다. 이 라이브러리에는 Messaging Server와 동일한 메시지 중요도 개념이 없으므로 수준을 *debug*로 설정하면 많은 의미 없는 데이터가 생성될 수 있습니다. 또한 http_sso 로그 파일이 일반 Messaging Server 로깅 코드에 의해 관리되지 않으며 정리 또는 롤오버되지 않습니다. 로그 수준을 기본값보다 높게 설정하는 경우 정리 작업은 시스템 관리자의 책임입니다. 기본값: 3
local.webmail.sso.singlesignoff	Messaging Server에서 Access Manager로의 단일 사인 오프입니다. Access Manager는 중앙 인증 기관이며 단일 사인 오프는 항상 Access Manager로부터 Messaging Server로 사용됩니다. 이 옵션을 사용하면 사이트에서 사용자가 웹 메일의 **로그아웃** 버튼을 눌러 Access Manager에서도 로그아웃할지 여부를 구성할 수 있습니다(일부 사용자 정의 작업 생략). 이 옵션은 기본적으로 사용됩니다. 이 옵션을 사용하지 않는 경우에는 로그아웃이 루트 문서를 참조하고 해당 루트 문서는 쿠키가 존재하고 유효한 이상 받은 메일함 디스플레이를 참조하기 때문에 사용자가 기본 웹 메일 클라이언트에서 로그아웃하면 자동으로 다시 로그인됩니다. 따라서, 사이트에서 이 옵션을 사용하지 않도록 선택하면 웹 메일 로그아웃시 발생되는 내용을 사용자 정의해야 합니다. 기본값: 예

6.1.3 SSO 문제 해결

SSO에 문제가 있는 경우 처음 수행할 작업은 오류에 대한 웹 메일 로그 파일 *msg-svr-base*/log/http를 검사하는 것입니다. 로깅 수준을 높이는 것도 유용할 수 있습니다(configutil -o logfile.http.loglevel -v debug). 이 작업이 도움이 되지 않는 경우 *msg-svr-base*/log/http_sso에서 amsdk 메시지를 검사한 다음 amsdk 로깅 수준을 높입니다(configutil -o local.webmail.sso.amloglevel -v 5). 새 로깅 수준을 적용하려면 서버를 다시 시작해야 합니다.

SSO에 여전히 문제가 있을 경우 로그인하는 동안 Access Manager와 Messaging Server 모두에서 정규화된 호스트 이름을 사용하는지 확인합니다. 쿠키는 동일한 도메인의 서버 간에만 공유되며 브라우저는 로컬 서버 이름에 대한 도메인이 무엇인지 알지 못하므로 브라우저에서 정규화된 이름을 사용해야 SSO가 작동합니다.

6.2 신뢰할 수 있는 원 SSO(레거시)

이 절에서는 신뢰할 수 있는 원 SSO에 대해 설명합니다. 이후의 모든 개발에서 Access Manager를 사용하게 될 것이므로 이 SSO 방법을 사용하지 않는 것이 좋습니다. 그러나 신뢰할 수 있는 원 SSO에서 사용할 수 있는 기능 중 일부는 현재 Access Manager SSO에서 사용할 수 없습니다. 이 절은 다음과 같은 하위 절로 구성되어 있습니다.

6.2.1 신뢰할 수 있는 원 SSO 개요 및 정의

SSO를 배포하기 전에 다음 용어에 대해 잘 알고 있어야 합니다.

- **SSO:** 단일 사인 온입니다. 하나의 응용 프로그램에 로그온하여 다른 응용 프로그램에 액세스할 수 있는 기능입니다. 사용자 아이디가 모든 응용 프로그램에서 동일합니다.
- **신뢰할 수 있는 응용 프로그램**. SSO 스키마(*SSO Prefix*)를 공유하고 서로 간에 쿠키 및 확인을 신뢰하는 응용 프로그램입니다. **피어** *SSO* **응용 프로그램**이라고도 합니다.
- **신뢰할 수 있는 원**. 신뢰할 수 있는 응용 프로그램 원입니다. 서로 동일한 SSO 접두어를 공유합니다.
- **SSO 접두어.** 동일한 신뢰할 수 있는 원 내의 다른 응용 프로그램이 생성한 쿠키를 찾는 데 사용할 수 있도록 SSO를 배포하는 개인이 정의하고 각 응용 프로그램에 알려진 문자열입니다. 다른 접두어를 사용하는 응용 프로그램은 동일한 원 내에 있지 않으므로 그러한 응용 프로그램 간에 이동하는 경우 다시 인증해야 합니다. 항상 그렇지는 않지만 구성 설정에서 접두어에 후행 -("-")가 명시되는 경우가 있습니다.
- **응용 프로그램 아이디**. (appid). SSO 원 내의 각 응용 프로그램에 대해 SSO를 배포하는 개인이 정의한 고유한 문자열입니다.
- **SSO 쿠키**. 브라우저에서 사용자가 일부 응용 프로그램에 인증되었음을 기억하는 데 사용하는 토큰입니다. 쿠키 이름 형식은 *SSO_prefix*-*application ID*입니다. 쿠키 값은 일반적으로 응용 프로그램에서 생성되는 세션 아이디인 SSO 키입니다.
- **쿠키 도메인.** 응용 프로그램이 쿠키를 보낼 수 있도록 제한된 도메인입니다. DNS 관점의 도메인입니다.

- **확인 URL.** 한 응용 프로그램에서 다른 응용 프로그램에 있는 쿠키를 확인하는 데 사용되는 URL입니다.

6.2.2 신뢰할 수 있는 원 SSO 응용 프로그램

SSO를 구현하기 전에 먼저 이 신뢰할 수 있는 원에 속하는 응용 프로그램을 고려해야 합니다. 이 신뢰할 수 있는 원에 포함될 수 있는 응용 프로그램은 Messenger Express, Calendar Express 및 이전 iPlanet Delegated Administrator for Messaging(Sun LDAP Schema 1만 지원하므로 권장되지 않음)입니다.

표 6–2에서는 SSO를 통해 서로 액세스할 수 있는 응용 프로그램을 보여 줍니다. 사용자의 관점에서 첫 번째 열의 응용 프로그램 중 하나에 로그인한 다음 사용자 아이디와 비밀번호를 다시 입력하지 않고 맨 위의 행에 있는 응용 프로그램에 액세스할 수 있다면 SSO가 적용되는 것입니다.

표 6–2 SSO 상호 운용성

대상:			
시작:	**Calendar Express**	**Messenger Express**	**Delegated Administrator**
Calendar Express	SSO	SSO	SSO
Messenger Express	SSO	해당 없음	SSO
Delegated Administrator	SSO	SSO	해당 없음

6.2.3 신뢰할 수 있는 원 SSO 제한

- 함께 작동하는 SSO 응용 프로그램은 동일한 도메인에 있어야 합니다.
- SSO 응용 프로그램은 서로의 SSO 확인 URL에 액세스할 수 있어야 합니다.
- 브라우저에서 쿠키를 지원해야 합니다.
- 보안상 브라우저가 실행되는 시스템에서는 SSO를 사용하지 마십시오.
- 다른 아이디로 전환하려면 브라우저를 다시 시작해야 합니다.
- Messenger Express와 Sun Java System Calendar Server 모두에서 단일 사인 오프가 사용된다고 가정할 때 Sun Java System Calendar Server에서 로그아웃한 경우에는 Messenger Express에 다시 로그인해야 합니다. Messenger Express에서 로그아웃한 경우에는 Sun Java System Calendar Server에 다시 로그인해야 합니다. 그러나 현재는 이 방법이 적용되지 않으므로 둘 중 하나에서 로그아웃한 후에도 다른 하나에서는 로그인한 상태를 유지할 수 있습니다.

6.2.4 신뢰할 수 있는 원 SSO 배포 시나리오 예

가장 단순한 SSO 배포 시나리오는 Messenger Express와 Delegated Administrator로만 구성됩니다. 동일한 신뢰할 수 있는 원 내에 포함되도록 동일한 SSO 접두어를 사용하여 동일한 시스템이나 다른 시스템에 Calendar Express를 추가하여 보다 복잡한 시나리오를 만들 수 있습니다. 그림 6–1에 표시되어 있습니다.

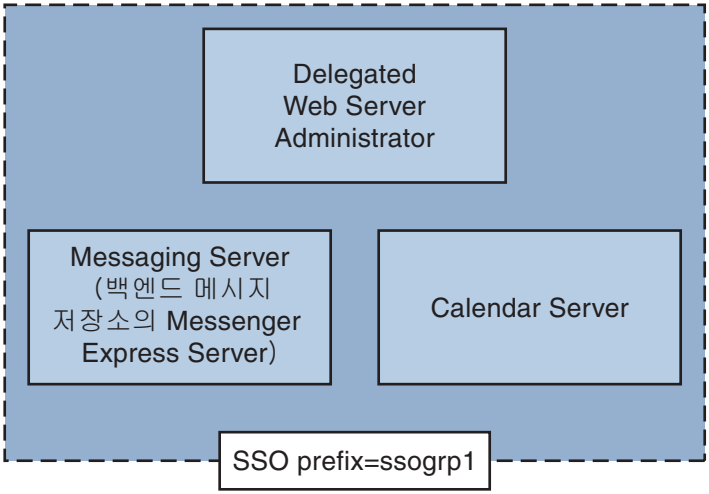


그림 6–1 단순 SSO 배포

복잡한 배포는 Webmail Server와 로드 밸런서를 포함할 수도 있습니다.

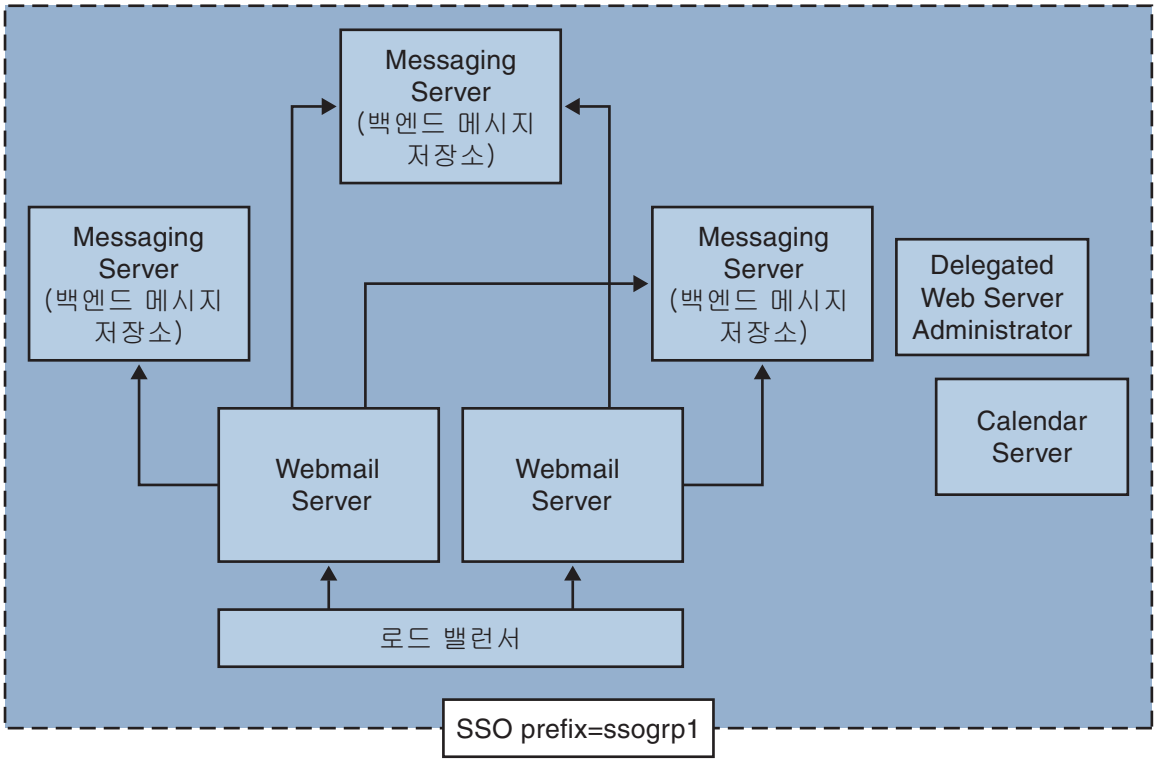


그림 6–2 복잡한 SSO 배포

6.2.5 신뢰할 수 있는 원 SSO 설정

이 절에서는 Messenger Express, Delegated Administrator 및 Calendar Manager에 대한 SSO 설정에 대해 설명합니다.

▼ Messenger Express, Delegated Administrator 및 Calendar Manager에 대해 SSO를 설정하는 방법

1 SSO에 대해 Messenger Express를 구성합니다.

a. 적절한 SSO `configutil` 매개 변수를 설정합니다.

Delegated Administrator가 있는 Messenger Express에 대해 단일 사인 온(SSO)을 사용하려면 구성 매개 변수를 다음과 같이 설정합니다(기본 도메인을 siroe.com으로 가정). 이러한 매개 변수는 표 6–3에 설명되어 있습니다. 루트 사용자여야 합니다. `cd`에서 *instance_root*

```
configutil -o local.webmail.sso.enable -v 1
configutil -o local.webmail.sso.prefix -v ssogrp1
```

ssogrp1은 Delegated Administrator에서 사용하는 기본 SSO 접두어입니다(다른 접두어 선택 가능). Delegated Administrator 및 Calendar Server 구성 시 이 기본값을 사용하면 입력 작업이 줄어듭니다.

```
configutil -o local.webmail.sso.id -v ims5
```

*ims5*는 다른 응용프로그램과 Messenger Express(ME)를 식별하기 위해 선택한 이름입니다.

```
configutil -o local.webmail.sso.cookiedomain -v ".siroe.com"
```

위의 도메인은 서버에 연결하는 ME/브라우저 클라이언트에 의해 사용 된 도메인과 반드시 일치해야 합니다. 따라서 이 서버의 호스트된 도메인이 xyz.com일 수도 있지만 DNS에서 실제 도메인을 사용해야 합니다. This value must start with a period.

```
configutil -o local.webmail.sso.singlesignoff -v 1
configutil -o local.sso.ApplicationID.verifyurl -v \
"http://ApplicationHost:port/VerifySSO?"
```

ApplicationID SSO 응용프로그램에 제공한 이름입니다(예: Delegated Administrator의 ida, Calendar Server의 ics50). 응용프로그램 호스트:포트는 호스트 및 응용프로그램의 포트 수입니다. 각각의 Messaging Server가 아닌 응용프로그램을 위한 행을 구성할 것입니다. 예:

```
configutil -o local.sso.ida.verifyurl -v \
"http://siroe.com:8080/VerifySSO?"
```

b. 구성을 변경한 후 Messenger Express http 서버를 다시 시작합니다.

```
cd instance_root./stop-msg http
./start-msg http
```

2 SSO에 대해 Directory Server를 구성합니다.

a. 디렉토리에 프록시 사용자 계정을 만듭니다.

프록시 사용자 계정을 사용하면 Delegated Administrator에서 프록시 인증을 위해 Directory Server에 바인드할 수 있습니다. 다음 LDIF 코드(proxy.ldif)를 사용하면 ldapadd를 통해 프록시 사용자 계정 항목을 만들 수 있습니다.

```
ldapadd -h mysystem.siroe.com -D "cn=Directory Manager" -w password -v -f
proxy.ldif
dn: uid=proxy, ou=people, o=siroe.com, o=isp
objectclass: top
objectclass: person
objectclass: organizationalperson
objectclass: inetorgperson
uid: proxy
givenname: Proxy
```

```
sn: Auth
cn: Proxy Auth
userpassword: proxypassword
```

b. 프록시 사용자 계정 인증을 위해 해당 ACI를 만듭니다.

`ldapmodify` 유틸리티를 사용하여 Delegated Administrator를 설치할 때 만든 각 접미어에 대한 ACI를 만듭니다.

`osiroot` - 사용자 데이터를 저장하기 위해 입력한 접미어입니다(기본값: `o=isp`). `osiroot`는 조직 트리의 루트입니다.

`dcroot` - 도메인 정보를 저장하기 위해 입력한 접미어입니다. 기본값은 `o=internet`입니다.

`osiroot` - 구성 정보를 저장하기 위해 입력한 접미어이며 사용자 데이터를 저장하기 위해 입력한 값과 동일해야 합니다.

다음은 앞에서 작성한 프록시 사용자의 `osiroot`에 대한 ACI 항목(`aci1.ldif`) 예입니다.

```
dn: o=isp
changetype: modify
add: aci
aci: (target="ldap:///o=isp")(targetattr="*")(version 3.0; acl
"proxy";allow (proxy) userdn="ldap:///uid=proxy, ou=people,
o=siroe.com, o=isp";)
```

```
ldapmodify -h siroe.com -D "cn=Directory Manager" -w password -v
-f aci1.ldif
```

다음과 같이 `dcroot`에 대해 비슷한 ACI 항목(`aci2.ldif`)을 만듭니다.

```
dn: o=internet
changetype: modify
add: aci
aci: (target="ldap:///o=internet")(targetattr="*")(version 3.0; acl
"proxy";allow (proxy) userdn="ldap:///uid=proxy, ou=people,
o=siroe.com, o=isp";)
```

```
ldapmodify -h siroe.com -D "cn=Directory Manager" -w password -v
-f aci2.ldif
```

3 Delegated Administrator를 구성합니다.

a. Delegated Administrator `resource.properties` 파일에 컨텍스트에 대한 프록시 사용자 자격 증명과 쿠키 이름을 추가합니다.

`resource.properties` 파일에서 다음 항목에 대한 주석 처리를 취소하고 수정합니다.

```
LDAPDatabaseInterface-ldapauthdn=Proxy_Auth_DN
LDAPDatabaseInterface-ldapauthpw=Proxy_Auth_Password
```

```
NDAAuth-singleSignOnId=SSO_Prefix-
NDAAuth-applicationId=DelAdminID
```

예를 들면 다음과 같습니다.

```
LDAPDatabaseInterface-ldapauthdn= uid=proxy,ou=people,o=cesta.com,o=isp
LDAPDatabaseInterface-ldapauthpw=proxypassword
NDAAuth-singleSignOnId=ssogrp1-
NDAAuth-applicationId=ida
```

`resource.properties` 파일은 다음 위치에 저장됩니다.

iDA_svr_base`/nda/classes/netscape/nda/servlet/`

b. 참여하는 서버의 확인 URL을 추가합니다.

수신하는 단일 사인 온(SSO) 쿠키를 확인하려면 Delegated Administrator에서 연결할 사람을 알고 있어야 합니다. 알려진 모든 참여 서버에 대한 확인 URL을 제공해야 합니다.

다음 예에서는 Messenger Express를 설치하고 해당 응용 프로그램 아이디가 `msg5`라고 가정합니다. Delegated Administrator `resource.properties` 파일을 편집하고 다음 항목을 추가합니다.

```
verificationurl-ssogrp1-msg5=http://webmail_hostname:port/VerifySSO?
verificationurl-ssogrp1-ida=http://iDA_hostname:port/VerifySSO?
verificationurl-ssogrp1-ics50=http://iCS_hostname:port/VerifySSO?
```

`resource.properties` 파일은 다음 디렉토리에 있습니다.

iDA_svr_base`/nda/classes/netscape/nda/servlet/`

4 Delegated Administrator 단일 사인 온(SSO) 쿠키 정보를 추가하고 UTF8 매개 변수 인코딩을 사용합니다.

a. Delegated Administrator에 대한 컨텍스트 식별자를 정의합니다.

`servlets.properties` 파일을 편집하고 `servlet.*.context=ims50` 텍스트에 포함된 모든 행의 주석을 취소합니다. 여기서 *는 모든 문자열입니다.

`servlets.properties` 파일은 다음 위치에 있습니다.

Web_Svr_Base`/https-`*instancename*`/config/`

b. Enterprise Server 구성에서 해당 컨텍스트에 대한 쿠키 이름을 지정합니다.

Enterprise Server `contexts.properties` 파일을 편집하고 파일의 맨 아래쪽에 있는 `#IDACONF-Start` 행 앞에 다음을 추가합니다.

```
context.ims50.sessionCookie=ssogrp1-ida
```

`contexts.properties` 파일은 다음 위치에 있습니다.

Web_Svr_Base`/https-`*instancename*`/config/`

c. ims5 컨텍스트에 대해 UTF8 매개 변수 인코딩을 사용합니다.

Enterprise Server 구성에서 ims5 컨텍스트에 대해 UTF8 매개 변수 인코딩을 사용하려면 Enterprise Server contexts.properties 파일에 다음 항목을 추가합니다.

```
context.ims50.parameterEncoding=utf8
```

5 Messenger Express를 다시 시작합니다.

단계 1a에서 2c에 설명된 대로 구성을 변경한 후에는 Messenger Express를 다시 시작해야 변경 내용이 적용됩니다.

Web_Svr_Base/**https-***instance_name*/**stop**
Web_Svr_Base/**https-***instancename*/**start**

6 이 SSO 그룹에서 Calendar를 배포하려면 Calendar Server를 구성합니다.

ics.conf를 편집하고 다음을 추가합니다.

```
sso.appid = "ics50"
sso.appprefix = "ssogrp1"
sso.cookiedomain = ".red.iplanet.com"
sso.enable = "1"
sso.singlesignoff = "true"
sso.userdomain = "mysystem.red.iplanet.com"
sso.ims5.url="http://mysystem.red.iplanet.com:80/VerifySSO?"
sso.ida.url=http://mysystem.red.iplanet.com:8080/VerifySSO?
```

7 Calendar Server를 다시 시작합니다.

```
start-cal
```

8 Messenger Express http 서버를 다시 시작합니다.

msg-svr-base/sbin/**stop-msg http**
msg-svr-base/sbin/**start-msg http**

6.2.6 Messenger Express의 신뢰할 수 있는 SSO 구성 매개 변수

145 페이지 “6.2.6 Messenger Express의 신뢰할 수 있는 SSO 구성 매개 변수”에 표시된 것처럼 configutil 명령을 사용하여 Messenger Express에 대한 단일 사인 온(SSO) 구성 매개 변수를 수정할 수 있습니다. configutil에 대한 자세한 내용은 **Sun Java System Messaging Server 6.3 Administration Reference**를 참조하십시오.

표 6–3 신뢰할 수 있는 원 단일 사인 온(SSO) 매개 변수

매개 변수	설명
`local.sso.` *appid*`.verifyurl`	피어 SSO 응용 프로그램에 대한 확인 URL 값을 설정합니다. *appid*는 해당 SSO 쿠키를 수락하는 피어 SSO 응용 프로그램의 응용 프로그램 아이디입니다. 예를 들어, Delegated Administrator에 대한 기본 *appid*는 `nda45`이고 실제 값은 Delegated Administrator`resource.properties` 파일 항목 `NDAAuth-applicationID`에 의해 지정됩니다. 신뢰할 수 있는 각 피어 SSO 응용 프로그램에 대해 정의된 하나의 매개 변수가 있어야 합니다. 확인 URL의 표준 형식은 다음과 같습니다. `http://`*nda-host:port* `/VerifySSO?` 여러 Webmail Server 및 메시지 저장소 서버(Messenger Express 실행) 또는 Calendar 프런트 엔드 앞에 로드 밸런서를 사용할 경우 `verifyurl`에 실제 호스트 이름을 사용하여 각 물리적 시스템에 대해 서로 **다른** `appid`를 지정해야 합니다. 그렇게 하면 쿠키를 확인하는 데 올바른 시스템이 사용됩니다.
`local.webmail.sso.cookiedomain`	이 매개 변수의 문자열 값은 Messenger Express HTTP 서버에서 설정한 모든 SSO 쿠키의 쿠키 도메인 값을 설정하는 데 사용됩니다. 기본값은 null입니다. 이 도메인은 Messenger Express 브라우저에서 서버에 액세스하는 데 사용한 DNS 도메인과 일치해야 합니다. 호스트된 도메인 이름이 아닙니다.
`local.webmail.sso.enable`	로그인 페이지를 가져올 때 클라이언트가 표시한 SSO 쿠키 허용 및 확인, 로그인에 성공한 경우 클라이언트에 SSO 쿠키 반환 및 쿠키 확인을 위해 다른 SSO 파트너의 요청에 회신 등을 포함하여 모든 단일 사인 온(SSO) 기능을 사용하거나 사용하지 않습니다. 0이 아닌 값을 설정하면 서버에서 모든 SSO 기능을 수행합니다. 0을 설정하면 서버에서 이러한 SSO 기능을 수행하지 않습니다. 기본값은 0입니다.
`local.webmail.sso.id`	Messenger Express HTTP 서버에서 설정한 SSO 쿠키의 서식을 지정할 때 이 매개 변수 문자열 값이 응용 프로그램 아이디 값으로 사용됩니다. 기본값은 null입니다. 이 값은 임의의 문자열입니다. 이 값은 `resource.properties` 파일에서 Delegated Administrator에 대해 지정한 값과 일치해야 합니다. `resource.properties`의 해당 항목은 다음과 같습니다. `Verifycationurl-XXX-YYY=http://webmailhost:webmailport/VerifySSO?` 여기서 `XXX`는 위에서 설정한 `local.webmail.sso.prefix` 값이고 `YYY`는 여기서 설정한 `local.webmail.sso.id` 값입니다.

표 6–3 신뢰할 수 있는 원 단일 사인 온(SSO) 매개 변수 *(계속)*

매개 변수	설명
`local.webmail.sso.prefix`	Messenger Express HTTP 서버에서 설정한 SSO 쿠키의 서식을 지정할 때 이 매개 변수 문자열 값이 접두어 값으로 사용됩니다. 이 접두어가 있는 SSO 쿠키만 서버에 인식되며 다른 SSO 쿠키는 모두 무시됩니다. 이 매개 변수에 null 값을 설정하면 서버에서 모든 SSO 기능을 효과적으로 비활성화할 수 있습니다. 기본값은 null입니다. 이 문자열은 후행 -가 없는 `resource.properties` 파일에서 Delegated Administrator에 사용된 문자열과 일치해야 합니다. 예를 들면 다음과 같습니다. `NDAAuth-singleSignOnID=ssogrp1-` 이 값을 `ssogrp1`로 설정해야 합니다.
`local.webmail.sso.singlesignoff`	0이 아닌 값으로 설정한 경우 이 매개 변수의 정수 값은 클라이언트가 로그아웃할 때 local.webmail.sso.prefix에 구성된 값과 일치하는 접두어 값을 갖는 클라이언트의 모든 SSO 쿠키를 지웁니다. 0으로 설정하면 Messenger Express는 클라이언트가 로그아웃할 때 해당 클라이언트의 SSO 쿠키만 지웁니다. 기본값은 0입니다.

◆ ◆ ◆ 7 장

7 멀티플렉서 서비스 구성 및 관리

이 장에서는 표준 메일 프로토콜(POP, IMAP 및 SMTP)용 MMP(Messaging Multiplexor)에 대해 설명합니다. 이전 릴리스에서는 Messenger Express 웹 인터페이스에 사용되는 Messenger Express Multiplexor에 대해 설명했지만 이 기능은 더 이상 필요하지 않습니다. 130 페이지 “5.7 HTTP 서비스 구성”을 참조하십시오.

이 장은 다음 내용으로 구성되어 있습니다.

7.1 멀티플렉서 서비스

멀티플렉서는 여러 메시지 저장소에 간접적으로 연결하는 데 사용할 수 있는 하나의 도메인 이름을 제공하기 때문에 수평 확장성(시스템을 추가하면 더 많은 사용자를 지원할 수 있는 기능)을 확보하는 데 필수적입니다. 멀티플렉서는 보안 이점도 제공할 수 있습니다.

MMP는 Messaging Server와 별도로 관리되지만 Messenger Express Multiplexing은 메시지 저장소 및 메시지 액세스 설치에 포함되어 있는 HTTP 서비스(`mshttpd`)에 내장되어 있습니다.

7.1.1 멀티플렉서의 장점

사용량이 많은 Messaging Server의 메시지 저장소는 크기가 매우 커질 수 있습니다. 따라서 사용자 메일함과 사용자 연결을 여러 서버로 분산시키면 용량과 성능을 향상시킬 수 있습니다. 또한 하나의 고용량의 큰 다중 프로세서 시스템보다는 여러 대의 작은 서버 시스템을 사용하는 것이 보다 비용면에서 효율적입니다.

메일 서버 설치의 크기가 커서 여러 메시지 저장소를 사용해야 하는 경우 멀티플렉서를 사용하면 여러 가지 장점이 있습니다. 사용자와 메시지 저장소를 간접적으로 연결하고 Messaging Server 간의 사용자 계정을 간단하게 재구성할 수 있으므로 다음과 같은 장점을 제공합니다.

- **간단한 사용자 관리**

  모든 사용자는 하나의 서버에 연결하기 때문에(POP, IMAP, SMTP 또는 웹 액세스용 멀티플렉서 시스템을 각각 분리하는 경우에는 하나 이상) 전자 메일 클라이언트를 미리 구성하고 일정한 로그인 정보를 모든 사용자에게 배포할 수 있습니다. 이를 통해 관리 작업이 단순화하고 오류가 발생할 수 있는 로그인 정보가 배포되는 가능성을 줄일 수 있습니다.

  특히 로드가 많은 상황에서는 여러 대의 멀티플렉서 서버를 동일한 구성으로 실행하고 DNS 라운드 로빈이나 로드 균형 조정 시스템을 사용하여 연결을 관리할 수 있습니다.

  멀티플렉서는 LDAP 디렉토리에 저장된 정보를 사용하여 각 사용자의 Messaging Server를 찾기 때문에 시스템 관리자는 사용자를 새 서버로 간단하게 옮길 수 있으며 사용자도 쉽게 알 수 있습니다. 관리자는 사용자의 메일함을 한 Messaging Server에서 다른 Messaging Server로 옮긴 다음 LDAP 디렉토리에서 사용자 항목을 업데이트할 수 있습니다. 사용자의 메일 주소, 메일함 액세스 및 기타 클라이언트 기본 설정은 변경되지 않습니다.

- **향상된 성능**

  메시지 저장소가 너무 커서 단일 시스템에 저장할 수 없는 경우 일부 메시지 저장소를 다른 시스템으로 옮겨서 로드의 균형을 조정할 수 있습니다.

  각 시스템마다 다른 클래스의 사용자를 할당할 수 있습니다. 예를 들어 프리미엄 사용자는 용량이 크고 성능도 강력한 시스템에 위치시킬 수 있습니다.

  멀티플렉서는 버퍼링을 수행하기 때문에 클라이언트 연결 속도가 늦더라도(예: 모뎀 사용) Messaging Server의 속도가 느려지지 않습니다.

- **비용 절감.** 멀티플렉서로 여러 Messaging Server를 효율적으로 관리할 수 있기 때문에 하나의 대형 시스템보다는 여러 대의 작은 서버 시스템을 구입하면 전체 비용을 줄일 수 있습니다.
- **뛰어난 확장성.** 멀티플렉서를 사용하면 구성을 쉽게 확장할 수 있습니다. 성능이나 저장소 용량의 필요가 증가하면 기존 투자를 교체할 필요 없이 시스템을 계속 추가할 수 있습니다.
- **사용자 중단 시간 최소화.** 멀티플렉서를 사용하여 여러 소형 저장소 시스템에서 대규모 사용자 기반으로 분산하면 사용자 중단 시간을 차단합니다. 즉, 개별 서버가 실패하더라도 해당 서버의 사용자에게만 영향이 미칩니다.
- **향상된 보안.** 멀티플렉서가 설치된 서버 시스템을 방화벽 시스템으로 사용할 수 있습니다. 모든 클라이언트 연결을 이 시스템을 통해 라우팅하면 외부 컴퓨터가 내부 메시지 저장소 시스템에 액세스하는 것을 제한할 수 있습니다. 멀티플렉서는 클라이언트와 암호화된 통신 및 암호화되지 않은 통신을 모두 지원합니다.

7.2 Messaging Multiplexor 정보

Sun Java System Messaging Multiplexor(MMP)는 여러 개의 백엔드 Messaging Server에 대한 연결의 단일 지점 역할을 하는 특수 Messaging Server입니다. Messaging Multiplexor를 사용하여 대규모 메시징 서비스 공급자는 POP 및 IMAP 사용자 메일함을 여러 시스템에서 분산시켜 메시지 저장소 용량을 늘릴 수 있습니다. 모든 사용자가 하나의 멀티플렉서 서버에 연결하고 이 서버가 각 연결을 적절한 Messaging Server로 리디렉션합니다.

여러 사용자에게 전자 메일 서비스를 제공하는 경우 Messaging Multiplexor를 설치 및 구성하여 전체 Messaging Server의 배열이 메일 사용자에게는 하나의 호스트로 표시되도록 할 수 있습니다.

Messaging Multiplexor는 Messaging Server의 일부로 제공됩니다. MMP는 Messaging Server 또는 기타 Sun Java System 서버를 설치할 때 함께 설치하거나 나중에 MMP만 따로 설치할 수 있습니다. MMP 지원 기능은 다음과 같습니다.

- 메일 클라이언트와의 암호화(SSL)된 통신 및 암호화되지 않은 통신 모두
- 클라이언트 인증서 기반 인증(153 페이지 “7.2.3 인증서 기반 클라이언트 인증”에서 설명함)
- 사용자 사전 인증(154 페이지 “7.2.4 사용자 사전 인증”에서 설명함)
- 다른 IP 주소에 수신하고 도메인 이름을 사용자 아이디에 자동으로 추가하는 가상 도메인(154 페이지 “7.2.5 MMP 가상 도메인”에서 설명함)
- 여러 서버에 다중 MMP 설치
- 향상된 LDAP 검색
- 레거시 POP 클라이언트용 POP before SMTP 서비스. 자세한 내용은 712 페이지 “23.8 POP before SMTP 사용”을 참조하십시오.

이 절은 다음과 같은 하위 절로 구성되어 있습니다.

7.2.1 Messaging Multiplexor의 작동 방식

MMP는 메일 사용자를 여러 서버 시스템으로 분산시키는 다중 스레드 서버입니다. MMP는 다른 서버 시스템(사용자 메일함이 있는 시스템)으로 지정된 수신 클라이언트 연결을 처리합니다. 클라이언트는 MMP 자체에 연결하며, MMP는 사용자에 대한 적절한 서버를 결정하고 해당 서버에 연결한 다음 클라이언트와 서버 사이에 데이터를

전달합니다. 이 기능을 통해 인터넷 서비스 공급자 및 기타 대규모 설치 조직에서는 메시지 저장소를 여러 시스템에 분산시킬 수 있으며(용량 증가), 사용자(효율성 증가)와 외부 클라이언트(보안 증가)에게는 하나의 메일 호스트로 보이게 합니다. 151 페이지 "7.2.1 Messaging Multiplexor의 작동 방식"은 MMP 설치에서 서버와 클라이언트가 상호간 어떻게 관련되어 있는지 보여 줍니다.

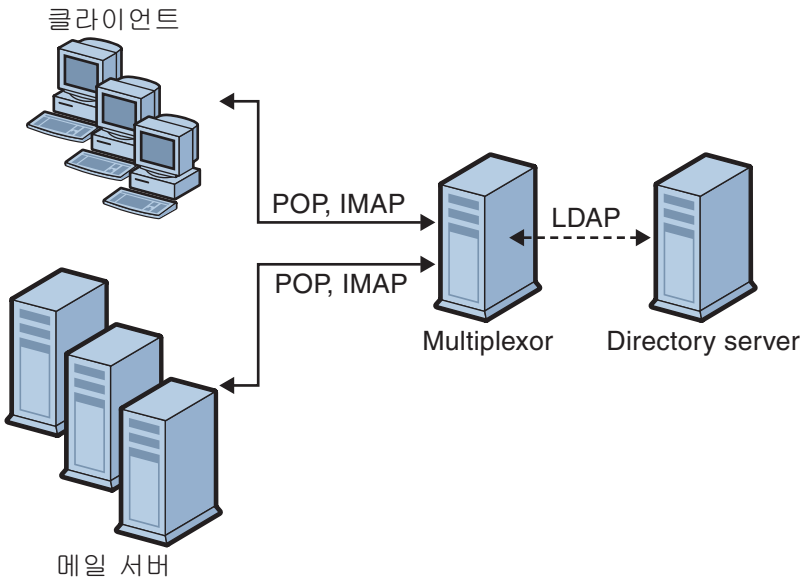


그림 7–1 MMP 설치 환경에서의 클라이언트와 서버

모든 POP, IMAP 및 SMTP 클라이언트를 Messaging Multiplexor와 함께 사용할 수 있습니다. MMP는 연결을 수용하고, LDAP 디렉토리 조회를 수행하고, 연결을 적절하게 라우팅합니다. 다른 메일 서버 설치와 마찬가지로 각 사용자에게는 특정 주소와 특정 Messaging Server의 메일함이 할당됩니다. 하지만 모든 연결은 MMP를 통해 경로가 지정됩니다.

다음은 사용자 연결을 설정하는 세부 단계입니다.

1. 사용자의 클라이언트가 MMP에 연결합니다. MMP는 예비 인증 정보(아이디)를 받습니다.
2. MMP는 Directory Server에 쿼리하여 사용자의 메일함이 포함된 Messaging Server를 결정합니다.
3. MMP는 적절한 Messaging Server에 연결하여 인증을 재수행한 다음 연결 기간 동안 전달 파이프 역할을 합니다.

7.2.2 암호화(SSL) 옵션

Messaging Multiplexor는 Messaging Server와 메일 클라이언트 사이의 암호화(SSL)된 통신 및 암호화되지 않은 통신을 모두 지원합니다. 현재 버전의 Messaging Server는 새 인증서 데이터베이스 형식(cert8.db)을 지원합니다.

SSL이 활성화된 경우 MMP는 STARTTLS를 지원하며 MMP가 SSL IMAP, POP 및 SMTP 연결에 대한 추가 포트에서 수신하도록 구성할 수도 있습니다.

IMAP, POP 및 SMTP 서비스에 대해 SSL 암호화를 활성화하려면 `ImapProxyAService.cfg`, `PopProxyAService.cfg` 및 `SmtpProxyAService.cfg` 파일을 각각 편집합니다. 또한 IMAP, POP 및 SMTP 서버 포트의 보안 여부에 관계 없이 이러한 모든 서버 포트 목록이 포함되게 하려면 `AService.cfg` 파일의 `default:ServiceList` 옵션을 수정해야 합니다. 자세한 내용은 159 페이지 “7.4 SSL로 MMP 구성”을 참조하십시오.

기본적으로 SSL 구성 매개 변수는 주석 처리되어 있으므로 SSL이 활성화되어 있지 않습니다. SSL을 활성화하려면 SSL 서버 인증서를 설치해야 합니다. 그런 다음 SSL 매개 변수의 주석 처리를 제거하고 적절하게 설정해야 합니다. SSL 매개 변수의 목록을 보려면 **Sun Java System Messaging Server 6.3 Administration Reference**의 “Encryption (SSL) Option”을 참조하십시오.

7.2.3 인증서 기반 클라이언트 인증

MMP는 인증서 매핑 파일(`certmap.conf`)을 사용하여 클라이언트의 인증을 사용자/그룹 Directory Server의 올바른 사용자와 일치시킬 수 있습니다.

인증서 기반 클라이언트 인증을 사용하려면 153 페이지 “7.2.2 암호화(SSL) 옵션”에 설명된 대로 SSL 암호화도 활성화해야 합니다.

또한 저장소 관리자도 구성해야 합니다. 메일 관리자를 사용할 수는 있지만 필요에 따라 사용 권한을 설정할 수 있도록 `mmpstore` 등의 고유한 사용자 아이디를 만드는 것이 좋습니다.

MMP는 `certmap` 플러그 인을 지원하지 않습니다. 대신 MMP에서는 `certmap.conf` 파일에 향상된 `DNComps` 및 `FilterComps` 속성 값 항목을 사용할 수 있습니다. 이러한 향상된 형식의 항목은 다음 형식을 사용합니다.

mapname`:DNComps` *FROMATTR=TOATTR**mapname*`:FilterComps` *FROMATTR=TOATTR*

인증서의 subjectDN에 있는 *FROMATTR* 값은 *TOATTR* =*value* 요소와 함께 LDAP 쿼리를 구성하는 데 사용할 수 있습니다. 예를 들어 subjectDN이 “`cn=Pilar Lorca, ou=pilar, o=siroe.com`”인 인증서는 다음 행을 사용하여 “`(uid=pilar)`”의 LDAP 쿼리에 매핑할 수 있습니다.

mapname`:FilterComps ou=uid`

▼ IMAP 또는 POP 서비스에 대한 인증서 기반 인증을 활성화하는 방법

1 **저장소 관리자로 사용할 사용자 아이디를 결정합니다.**
메일 관리자를 이 용도로 사용할 수 있는 경우 저장소 관리자의 고유한 사용자 아이디(예: `mmpstore`)를 만드는 것이 좋습니다.

2 **153 페이지 "7.2.2 암호화(SSL) 옵션"에 설명된 대로 SSL 암호화가 활성화되었는지 확인합니다.**

3 **구성 파일에 `certmap.conf` 파일의 위치를 지정하여 MMP가 인증서 기반 클라이언트 인증을 사용하도록 구성합니다.**

4 **692 페이지 "23.5.1.6 신뢰할 수 있는 CA의 인증서 설치"에 설명된 대로 적어도 하나의 신뢰할 수 있는 CA 인증서를 설치합니다.**

7.2.4 사용자 사전 인증

MMP는 수신되는 사용자로 디렉토리에 바인딩하고 결과를 로깅하여 사용자를 사전 인증할 수 있는 옵션을 제공합니다.

주 – 사용자 사전 인증을 활성화하면 서버 성능이 저하됩니다.

로그 항목의 형식은 다음과 같습니다.

date time `(sid 0x`*hex*`) user` *name* `pre-authenticated - client IP` *address,* `server IP` *address*

여기서 *date*는 `yyyymmdd`의 형식이고, *time*은 `hhmmss` 형식으로 서버에서 구성된 시간이고, *hex*는 16진수로 표시되는 세션 식별자(`sid`)이며, `user` *name*에는 가상 도메인(있는 경우)이 포함되며, IP 주소는 점으로 구분된 네 개의 번호 형식으로 되어 있습니다.

7.2.5 MMP 가상 도메인

MMP 가상 도메인은 서버 IP 주소와 연관된 구성 설정 세트입니다. 이 기능의 기본 용도는 각 서버 IP 주소에 대해 서로 다른 기본 도메인을 제공하는 것입니다.

사용자는 짧은 형식의 userID 또는 `user@domain` 형식의 정규화된 userID로 MMP에 인증할 수 있습니다. 짧은 형식의 userID를 제공하면 MMP는 지정된 경우 `DefaultDomain` 설정을 추가합니다. 그 결과 여러 개의 호스트된 도메인을 지원하는 사이트는 서버 IP 주소와 MMP 가상 도메인을 각 호스트된 도메인에 연결시켜 짧은 사용자 아이디를 사용할 수 있게 할 수 있습니다.

특정 호스트된 도메인에 대한 사용자 하위 트리를 찾는 경우 해당 도메인의 LDAP 도메인 트리 항목에서 `inetDomainBaseDN` 속성을 찾는 것이 좋습니다. 백엔드 메시지 저장소도 LDAP에서 사용자를 조회해야 하고 가상 도메인을 지원하지 않기 때문에 MMP의 `LdapUrl` 설정은 이러한 검색에 적합하지 않습니다.

Sun LDAP Schema 2가 활성화된 경우(**Sun Java Enterprise System 5 Installation Guide for UNIX** 및 **Sun Java Communications Suite 5 Schema Reference** 참조) 지정된 도메인의 사용자 하위 트리는 해당 도메인의 조직 노드 아래에 있는 하위 트리의 모든 사용자입니다.

가상 도메인을 활성화하려면 `VirtualDomainFile` 설정이 가상 도메인 매핑 파일에 대한 전체 경로를 지정하는 인스턴스 디렉토리에서 `ImapProxyAService.cfg`, `PopProxyAService.cfg` 또는 `SmtpProxyAService.cfg` 파일을 편집합니다.

가상 도메인 파일의 각 항목에 대한 구문은 다음과 같습니다.

`vdmap` *name IPaddr*
name:parameter value

여기서 *name*은 IP 주소를 구성 매개 변수와 연결시키는 데만 사용되며 원하는 모든 이름을 사용할 수 있습니다. 또한 *IPaddr*은 점으로 구분된 네 개의 번호 형식이며 *parameter* 및 *value* 쌍은 가상 도메인을 구성합니다. 설정 시 가상 도메인 구성 매개 변수 값은 전역 구성 매개 변수 값보다 우선합니다.

다음은 가상 도메인에 지정할 수 있는 구성 매개 변수입니다.

```
AuthCacheSize and AuthCacheSizeTTL
AuthService
BindDN and BindPass
CertMap
ClientLookup
CRAMs
DefaultDomain
DomainDelim
HostedDomains
LdapCacheSize and LdapCacheTTL
LdapURL
MailHostAttrs
PreAuth
ReplayFormat
RestrictPlainPasswords
StoreAdmin and StoreAdminPass
SearchFormat
TCPAccess
TCPAccessAttr
```

주 – LdapURL이 제대로 설정되어 있지 않으면 BindDN, BindPass, LdapCacheSize 및 LdapCacheTTL 설정이 무시됩니다.

이러한 구성 매개 변수에 대한 자세한 내용은 **Sun Java System Messaging Server 6.3 Administration Reference**를 참조하십시오.

7.2.6 SMTP 프록시 정보

MMP에는 기본적으로 비활성화되어 있는 SMTP 프록시가 포함되어 있습니다. 인터넷 메일 표준이 이미 SMTP(DNS MX 레코드)의 수평 확장을 위한 적절한 기법을 제공하므로 대부분의 사이트에는 SMTP 프록시가 필요하지 않습니다.

SMTP 프록시는 유용한 보안 기능을 제공합니다. 우선 SMTP 프록시는 POP 프록시와 통합되어 일부 레거시 POP 클라이언트에 필요한 POP before SMTP 인증 기능을 구현합니다. 자세한 내용은 **Sun Java Communications Suite 5 Deployment Planning Guide**의 "Using the MMP SMTP Proxy" 및 712 페이지 "23.8 POP before SMTP 사용"를 참조하십시오. 또한 SMTP 프록시를 사용하여 SSL 가속 하드웨어에 대한 투자를 최대화할 수 있습니다. 700 페이지 "23.5.4 SMTP 프록시를 사용하여 SSL 성능을 최적화하는 방법"을 참조하십시오.

7.3 Messaging Multiplexor 설정

Messaging Server의 초기 런타임 구성 중에 시스템에서 MMP를 구성할 것인지 여부를 결정하였습니다. Messaging Server와 같은 시스템에 설정하거나 별도의 시스템에 설정할 수 있습니다.

주 – MMP는 DNS 결과를 캐시하지 않습니다. Messaging Server 제품을 배포하려면 로컬 네트워크에 고품질 캐싱 DNS 서버가 필요합니다.

다음 절에서는 MMP의 설정 방법에 대해 설명합니다.

다음에서 MMP에 대한 추가 정보를 볼 수 있습니다.

- **Sun Java System Messaging Server 6.3 Administration Reference**의 5 장, "Messaging Multiplexor Configuration"

7.3.1 MMP를 구성하기 전에

MMP를 구성하기 전에 다음 작업을 수행합니다.

1. MMP를 구성할 시스템을 선택합니다. MMP 전용 시스템을 사용하는 것이 가장 좋습니다.

   주 – POP 또는 IMAP 서버가 실행 중인 시스템에서는 MMP를 활성화하지 않는 것이 좋습니다.

   Messaging Server와 같은 시스템에 MMP를 설치한 경우 POP와 IMAP 서버를 비표준 포트로 설정해야 합니다. 이렇게 하면 MMP와 Messaging Server 포트가 서로 충돌하지 않습니다.

   ---

2. MMP를 구성할 시스템에서 MMP가 사용할 UNIX 시스템 사용자를 만듭니다. 이 새 사용자는 UNIX 시스템 그룹에 속해야 합니다. 47 페이지 “1.1 UNIX 시스템 사용자와 그룹 만들기”를 참조하십시오.
3. 아직 설정하지 않은 경우 Messaging Server에 사용할 Directory Server와 호스트 시스템을 설정합니다. 48 페이지 “1.2 Messaging Server 구성을 위해 Directory Server 준비”를 참조하십시오.
4. 백엔드 서버에 앞서 MMP를 업그레이드할 경우 `ImapProxyAService.cfg`에서 `Capability` 옵션을 다음과 같이 설정하여 이전 백엔드의 `capability` 명령에 대한 응답과 일치시켜야 합니다.

   ```
   IMAP4 IMAP4rev1 ACL QUOTA LITERAL+ NAMESPACE UIDPLUS CHILDREN LANGUAGE
   XSENDER X-NETSCAPE XSERVERINFO
   ```

   위에서 편의상 줄 바꿈이 되었지만 실제 구성 값은 한 줄로 되어 있어야 합니다.

7.3.2 Multiplexor 구성

MMP를 구성하려면 Messaging Multiplexor를 활성화하는 옵션이 있는 Messaging Server 구성 프로그램을 사용해야 합니다. 구성 프로그램에 대한 자세한 내용은 49 페이지 “1.3 Messaging Server 초기 런타임 구성 만들기 ”를 참조하십시오.

▼ MMP 구성 방법

1 MMP를 설치 및 구성하려는 시스템에 Sun Java System Messaging Server를 설치합니다.

2 Messaging Server 초기 런타임 구성을 만들어 MMP를 구성합니다. 49 페이지 “1.3 Messaging Server 초기 런타임 구성 만들기 ”를 참조하십시오.

다음과 같은 예외가 있습니다. Messaging Server를 설치할 때 Messaging Multiplexor 옵션만 확인합니다.

7.3.3 Multiplexor 파일

Messaging Multiplexor 파일은 *msg-svr-base*/config 구성 파일 디렉토리에 저장됩니다. 표 7–1에 나열된 Messaging Multiplexor 구성 파일의 구성 매개 변수를 수동으로 편집해야 합니다. 모든 MMP 구성 매개 변수에 대한 자세한 내용은 **Sun Java System Messaging Server 6.3 Administration Reference**의 "Multiplexor Configuration Parameters"를 참조하십시오.

표 7–1 Messaging Multiplexor 구성 파일

파일	설명
PopProxyAService.cfg	POP 서비스에 사용되는 구성 변수를 지정하는 구성 파일
PopProxyAService-def.cfg	POP 서비스 구성 템플리트. 파일은 start-msg mmp로 초기 MMP가 시작되고 난 후에만 만들어집니다.
ImapProxyAService.cfg	IMAP 서비스에 사용되는 구성 변수를 지정하는 구성 파일
ImapProxyAService-def.cfg	IMAP 서비스 구성 템플리트. 파일은 start-msg mmp로 초기 MMP가 시작되고 난 후에만 만들어집니다.
AService.cfg	시작할 서비스를 지정하고 POP 및 IMAP 서비스 모두가 공유하는 몇 가지 옵션을 지정하는 구성 파일
AService-def.cfg	시작할 서비스를 지정하고 POP 및 IMAP 서비스가 공유하는 몇 가지 옵션을 지정하는 구성 템플리트. 파일은 start-msg mmp로 초기 MMP가 시작되고 난 후에만 만들어집니다.
SmtpProxyAService.cfg	SMTP Proxy 서비스에 사용되는 구성 변수를 지정하는 선택적 구성 파일. POP before SMTP를 활성화하는 경우에 필요합니다. POP before SMTP가 활성화되지 않은 경우에도 SSL 하드웨어를 최대한 지원하려는 경우 유용합니다. POP before SMTP에 대한 자세한 내용은 712 페이지 "23.8 POP before SMTP 사용"을 참조하십시오.
SmtpProxyAService-def.cfg	SMTP 프록시 서비스에 사용되는 구성 변수를 지정하는 구성 템플리트. 파일은 start-msg mmp로 초기 MMP가 시작되고 난 후에만 만들어집니다.

예를 들어 LogDir 및 LogLevel 매개 변수는 모든 구성 파일에서 찾을 수 있습니다. ImapProxyAService.cfg에서는 IMAP 관련 이벤트의 로깅 매개 변수를 지정하는 데 사용되며, 이와 유사하게 PopProxyAService.cfg의 해당 매개 변수는 POP 관련 이벤트의 로깅 매개 변수 구성에 사용됩니다. SmtpProxyAService.cfg에서는 SMTP 프록시 관련 이벤트에 대한 로깅을 지정하는 데 사용됩니다.

하지만 AService.cfg에서는 LogDir 및 LogLevel이 POP, IMAP 또는 SMTP 서비스를 시작하는 데 실패하는 경우와 같은 MMP 실패를 로깅하는 데 사용됩니다.

주 – MMP를 구성하거나 업그레이드하면 구성 템플리트 파일을 덮어씁니다.

7.3.4 Multiplexor 시작

Messaging Multiplexor의 인스턴스를 시작, 중지하거나 새로 고치려면 *msg-svr-base*/sbin 디렉토리에 있는 표 7–2의 다음 명령 중 하나를 사용합니다.

표 7–2 MMP 명령

옵션	설명
start-msg mmp	MMP를 시작합니다(다른 MMP가 이미 실행 중인 경우에도 시작).
stop-msg mmp	최근 시작한 MMP를 중지합니다.
refresh mmp	활성 연결을 중단하지 않고 이미 실행 중인 MMP의 구성을 새로 고칩니다.

7.3.5 기존 MMP 수정

MMP의 기존 인스턴스를 수정하려면 필요에 따라 ImapProxyAService.cfg 및/또는 PopProxyAService.cfg 구성 파일을 편집합니다. 이러한 구성 파일은 *msg-svr-base*/config 하위 디렉토리에 있습니다.

7.4 SSL로 MMP 구성

SSL을 사용하기 위해 MMP를 구성하려면 다음을 수행합니다.

주 – 여기서는 MMP가 메시지 저장소 또는 MTA가 없는 시스템에 설치되어 있는 것으로 가정합니다.

▼ SSL로 MMP 구성 방법

1. **SSL 서버 인증서를 설치합니다(686 페이지 "23.5 암호화 및 인증서 기반 인증 구성 " 참조).**
2. **ImapProxyAService.cfg 파일을 편집하고 관련 SSL 설정의 주석 처리를 제거합니다.**
3. **SSL 및 POP를 사용하려면 PopProxyAService.cfg 파일을 편집하고 관련 SSL 설정의 주석 처리를 제거합니다.**

   또는 AService.cfg 파일을 편집하고 ServiceList 설정의 110 뒤에 |995를 추가해야 합니다.

4 **BindDN 및 BindPass 옵션이 ImapProxyAService.cfg와 PopProxyAService.cfg 파일에 설정되어 있는지 확인합니다.**

또한 DefaultDomain 옵션을 기본 도메인(정규화되지 않은 아이디에 사용할 도메인)으로 설정해야 합니다.

서버측 SSL 지원만 필요한 경우에는 이로써 작업이 끝났습니다. *msg-svr-base*/sbin 디렉토리에서 다음 명령으로 MMP를 시작합니다.

```
start-msg mmp
```

5 **SSL을 사용하여 메시지를 수락하고 비 SSL을 사용하여 백엔드 메일 서버에 보내도록 MMP를 설정하는 방법:**

ImapProxyAService.cfg 또는 PopProxyAService.cfg에서 SSL Backside Port 옵션을 0으로 설정합니다.

6 **MMP와 백엔드 서버 간에 SSL을 사용하지 않으려면 SSLBacksidePort 옵션을 0으로 설정합니다.**

▼ 클라이언트 인증서 기반 로그인을 사용하여 MMP를 구성하는 방법

클라이언트 인증서 기반 로그인을 사용하려면 다음을 수행합니다.

1 **클라이언트 인증서 복사본과 이 복사본을 서명한 CA 인증서를 얻습니다.**

2 **CA 인증서를 신뢰할 수 있는 인증 기관으로 가져옵니다(688 페이지 "23.5.1 인증서 얻기" 참조).**

3 **Messaging Server 설치 도중 만든 저장소 관리자를 사용합니다.**

자세한 내용은 561 페이지 "20.4 저장소에 대한 관리자 액세스 지정"을 참조하십시오.

4 **MMP에 대한 certmap.conf 파일을 만듭니다. 예를 들면 다음과 같습니다.**

```
certmap default default
default:DNComps
default:FilterComps e=mail
```

이것은 LDAP 서버에서 메일 속성을 찾아서 인증서 DN의 e 필드와 일치하는 항목을 찾는다는 의미입니다.

5 **ImapProxyAService.cfg 파일을 편집하고 다음을 수행합니다.**

a. **CertMapFile을 certmap.conf로 설정합니다.**

b. **StoreAdmin 및 StorePass를 단계 3의 값으로 설정합니다.**

c. **UserGroupDN을 사용자 및 그룹 트리의 루트로 설정합니다.**

6 **POP3을 사용한 클라이언트 인증서가 필요한 경우 PopProxyAService.cfg 파일에 대해 단계 5를 반복합니다.**

7 **MMP가 이미 실행 중이 아닌 경우 *msg-svr-base*/sbin 디렉토리에서 다음 명령을 사용하여 실행합니다.**

```
start-msg mmp
```

8 **클라이언트 인증서를 클라이언트로 가져옵니다. Netscape™ Communicator에서 자물쇠(보안) 아이콘을 누른 다음 인증서 아래에서 사용자를 선택하고 인증서 가져오기...를 선택합니다. 그런 다음 지시에 따릅니다.**

주 – 어느 곳에서나 클라이언트 인증서를 사용하려면 모든 사용자가 이 단계를 수행해야 합니다.

7.4.1 샘플 토폴로지

가상의 Siroe Corporation에는 별도의 시스템에 두 개의 Messaging Multiplexor가 있으며 각각 여러 Messaging Server를 지원합니다. POP 및 IMAP 사용자 메일함은 Messaging Server 시스템에서 분산되어 있으며, 각 서버는 POP 또는 IMAP 전용 서버입니다. ImapProxyAService 항목을 ServiceList 설정에서 제거하면 POP 서비스에 대한 클라이언트 액세스를 제한할 수 있습니다. 마찬가지로 ServiceList 설정에서 PopProxyAService 항목을 제거하면 IMAP 서비스에 대한 클라이언트 액세스를 제한할 수 있습니다. 각 Messaging Multiplexor는 POP만 지원하거나 IMAP만 지원합니다. LDAP 디렉토리 서비스는 별도의 전용 시스템에 있습니다.

이 토폴로지는 아래 그림 7–2에서 볼 수 있습니다.

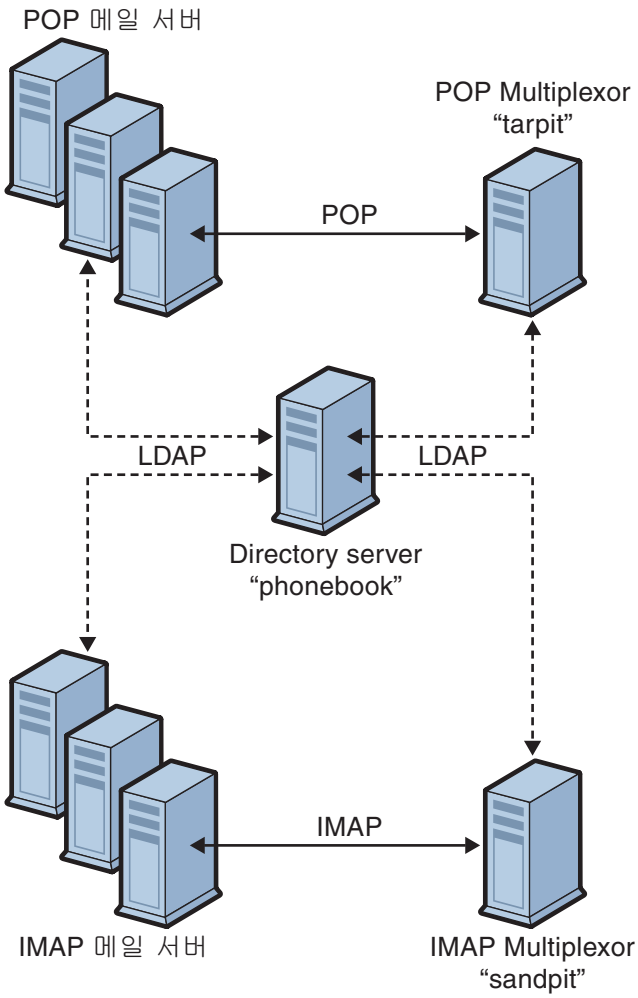


그림 7–2 여러 Messaging Server를 지원하는 여러 MMP

7.4.1.1 IMAP 구성 예

그림 7–2의 IMAP Messaging Multiplexor는 두 개의 프로세서가 있는 시스템인 `sandpit`에 설치되어 있습니다. 이 Messaging Multiplexor는 표준 포트에서 IMAP 연결(143)에 대기합니다. Messaging Multiplexor는 호스트 `phonebook`의 LDAP 서버와 사용자 메일함 정보를 통신하며, 적절한 IMAP 서버로 연결의 경로를 지정합니다. 이것은 IMAP 기능 문자열을 대체하고, 가상 도메인 파일을 제공하며, SSL 통신을 지원합니다.

`ImapProxyAService.cfg` 구성 파일은 다음과 같습니다.

```
default:LdapUrl ldap://phonebook.siroe.com/o=internet
default:LogDir /opt/SUNWmsgsr/config/log
default:LogLevel 5
default:BindDN "cn=Directory Manager"
default:BindPass secret
default:BacksidePort 143
default:Timeout 1800
default:Capability "IMAP4 IMAP4rev1 ACL QUOTA LITERAL+ NAMESPACE
UIDPLUS CHILDREN BINARY LANGUAGE XSENDER X-NETSCAPE XSERVERINFO"
default:SearchFormat (uid=%s)
default:SSLEnable yes
default:SSLPorts 993
default:SSLSecmodFile /opt/SUNWmsgsr/config/secmod.db
default:SSLCertFile /opt/SUNWmsgsr/config/cert8.db
default:SSLKeyFile /opt/SUNWmsgsr/config/key3.db
default:SSLKeyPasswdFile /opt/SUNWmsgsr/config/sslpassword.conf
default:SSLCipherSpecs all
default:SSLCertNicknames Siroe.com Server-Cert
default:SSLCacheDir /opt/SUNWmsgsr/config
default:SSLBacksidePort 993
default:VirtualDomainFile /opt/SUNWmsgsr/config/vdmap.cfg
default:VirtualDomainDelim @
default:ServerDownAlert "your IMAP server appears to be temporarily
out of service"
default:MailHostAttrs mailHost
default:PreAuth no
default:CRAMs no
default:AuthCacheSize 10000
default:AuthCacheTTL 900
default:AuthService no
default:AuthServiceTTL 0
default:BGMax 10000
default:BGPenalty 2
default:BGMaxBadness 60
default:BGDecay 900
default:BGLinear no
default:BGExcluded /opt/SUNWmsgsr/config/bgexcl.cfg
default:ConnLimits 0.0.0.0|0.0.0.0:20
default:LdapCacheSize 10000
default:LdapCacheTTL 900
default:HostedDomains yes
default:DefaultDomain Siroe.com
```

7.4.1.2 POP 구성 예

161 페이지 “7.4.1 샘플 토폴로지”에서 예로 든 POP Messaging Multiplexor는 4개의 프로세서가 있는 `tarpit` 시스템에 설치되어 있습니다. 이 Messaging Multiplexor는 표준

포트에서 POP 연결(110)을 수신합니다. Messaging Multiplexor는 호스트 phonebook의 LDAP 서버와 사용자 메일함 정보를 통신하며, 적절한 POP 서버로 연결의 경로를 지정합니다.

해당 PopProxyAService.cfg 구성 파일은 다음과 같습니다.

```
default:LdapUrl ldap://phonebook.siroe.com/o=internet
default:LogDir /opt/SUNWmsgsr/config/log
default:LogLevel 5
default:BindDN "cn=Directory Manager"
default:BindPass password
default:BacksidePort 110
default:Timeout 1800
default:SearchFormat (uid=%s)
default:SSLEnable no
default:VirtualDomainFile /opt/SUNWmsgsr/config/vdmap.cfg
default:VirtualDomainDelim @
default:MailHostAttrs mailHost
default:PreAuth no
default:CRAMs no
default:AuthCacheSize 10000
default:AuthCacheTTL 900
default:AuthService no
default:AuthServiceTTL 0
default:BGMax 10000
default:BGPenalty 2
default:BGMaxBadness 60
default:BGDecay 900
default:BGLinear no
default:BGExcluded /opt/SUNWmsgsr/config/bgexcl.cfg
default:ConnLimits 0.0.0.0|0.0.0.0:20
default:LdapCacheSize 10000
default:LdapCacheTTL 900
default:HostedDomains yes
default:DefaultDomain Siroe.com
```

7.5 MMP 작업

이 절에서는 기타 MMP 구성 작업에 대해 설명합니다. 다음과 같이 추가되었습니다.

7.5.1 MMP를 사용하여 메일 액세스 구성

MMP는 PORT_ACCESS 매핑 테이블을 사용하지 않습니다. 특정 IP 주소의 SMTP 연결을 거부하기를 원하고 MMP를 사용하는 경우 TCPAccess 옵션을 사용해야 합니다. 이 옵션 구문은 mailDomainAllowedServiceAccess와 동일합니다. **Sun Java Communications Suite 5 Schema Reference**를 참조하십시오. 이 항목은 704 페이지 "23.7.2 필터 구문"에도 설명되어 있습니다.

7.5.2 페일오버 MMP LDAP 서버 설정

하나의 LDAP 서버가 실패하면 다른 서버가 처리하도록 MMP에 대한 LDAP 서버를 두 개 이상 지정할 수 있습니다. PopProxyAservice.cfg 또는 ImapProxyAservice.cfg를 다음과 같이 수정합니다.

```
default:LdapUrl "ldap://ldap01 .yourdomain ldap02 .yourdomain/o=internet"
```

주 – 위 구성에서 호스트 이름 사이에 공백이 있어야 합니다.

◆ ◆ ◆ 8 장

8 MTA 개념

이 장에서는 MTA의 개념에 대해 설명합니다. 이 장은 다음 내용으로 구성되어 있습니다.

8.1 MTA 기능

*MTA(Message Transfer Agent)*는 Messaging Server의 구성 요소입니다(그림 8–1). 가장 기본적인 수준에서 MTA는 메시지 라우터입니다. MTA는 다른 서버에서 메시지를 수락하여 주소를 읽은 다음 최종 대상(일반적으로 사용자의 메일함)으로 가는 도중에 있는 다음 서버로 라우팅합니다.

수년 동안 많은 기능이 MTA에 추가되었으며 이에 따라서 MTA의 크기, 기능 및 복잡성이 증가했습니다. 이러한 MTA 기능은 중복되기는 하지만 일반적으로 다음과 같이 분류할 수 있습니다.

- **라우팅.** 메시지를 수락하여 필요에 따라(예: 별칭인 경우) 확장 또는 변환한 후에 다음 서버, 채널, 프로그램, 파일 등에 라우팅합니다. 라우팅 기능은 관리자가 메시지가 라우팅되는 방법에 대한 내부 및 외부 기법을 지정하는 수준까지 확장되었습니다. 예를 들어, SMTP 인증 사용, 다양한 SMTP 명령 및 프로토콜 사용, TCP/IP 또는 DNS 조회 지원, 작업 제출, 프로세스 제어 및 메시지 대기열 등을 지정할 수 있게 된 것입니다.
- **주소 다시 쓰기.** 봉투 주소는 흔히 라우팅 프로세스의 일부로 다시 쓰여지지만 봉투 또는 헤더 주소를 더 적절하거나 원하는 형태로 다시 쓸 수 있습니다.

- **필터링.** MTA는 주소, 도메인, 가능한 바이러스 및 스팸 내용, 크기, IP 주소, 헤더 내용 등에 기초하여 메시지를 필터링할 수 있습니다. 필터링된 메시지는 삭제, 거부 또는 수정되거나, 파일 또는 프로그램으로 보내지거나, 사용자 메일함으로 가는 도중의 다음 서버로 보내질 수 있습니다.
- **내용 수정.** 메시지 헤더와 내용을 수정할 수 있습니다. 예: 메시지를 특정 클라이언트나 특수한 문자 세트에서 읽을 수 있게 만들거나 스팸 또는 바이러스를 검사합니다.
- **감사.** 누가 언제, 어디서, 무엇을 제출했는지 추적합니다.

그림 8–2에 나온 여러 하위 구성 요소와 프로세스가 이러한 기능을 지원합니다. 이 장에서는 이러한 하위 구성 요소와 프로세스에 대해 설명합니다. 또한 시스템 관리자는 여러 도구를 사용하여 이러한 기능을 활성화하고 구성할 수 있습니다. 이러한 도구에는 MTA 옵션, `configutil` 매개 변수, 매핑 테이블, 키워드, 채널, 다시 쓰기 규칙 등이 포함됩니다. 이러한 도구에 대해서는 다음 MTA장에서 설명합니다.

- 10 장
- 11 장
- 12 장
- 13 장
- 14 장
- 16 장
- 17 장
- 18 장
- 23 장
- 25 장
- 26 장
- 27 장

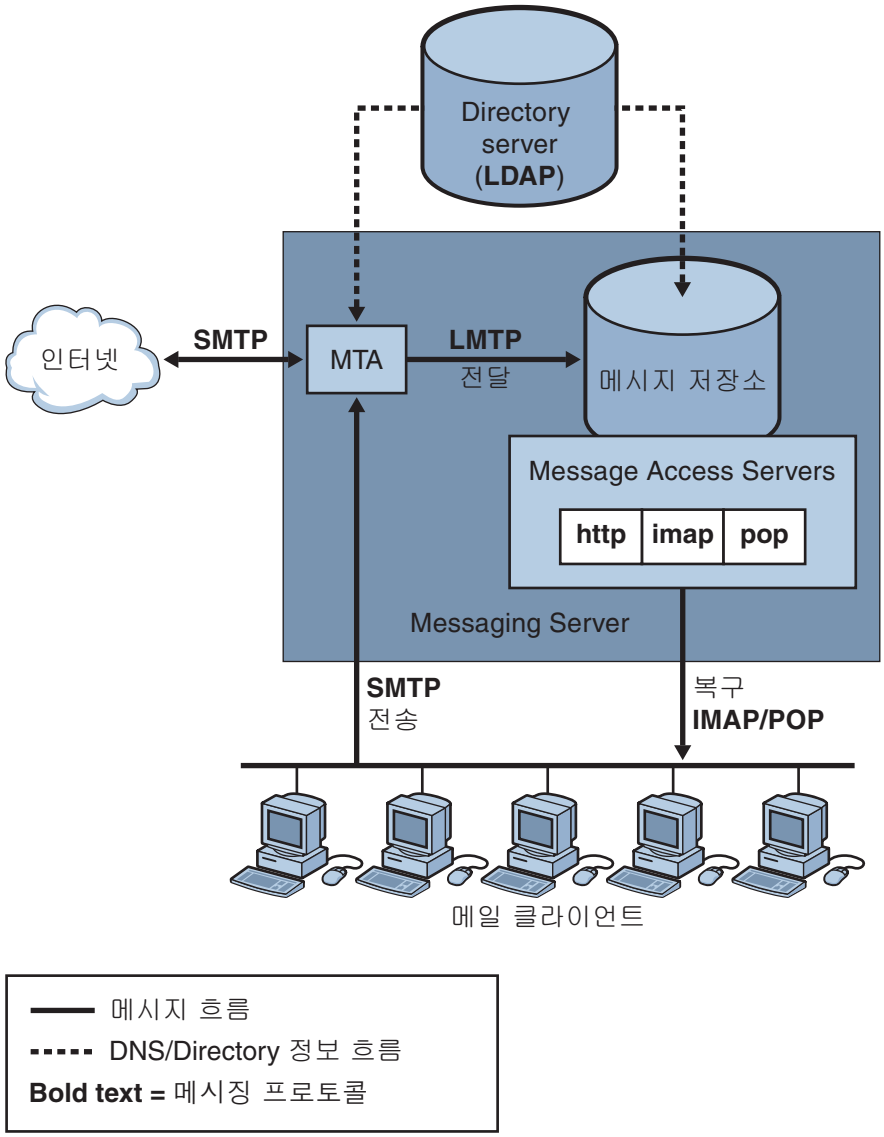


그림 8–1 Messaging Server, 단순화된 구성 요소 보기(Communications Express는 표시되지 않음)

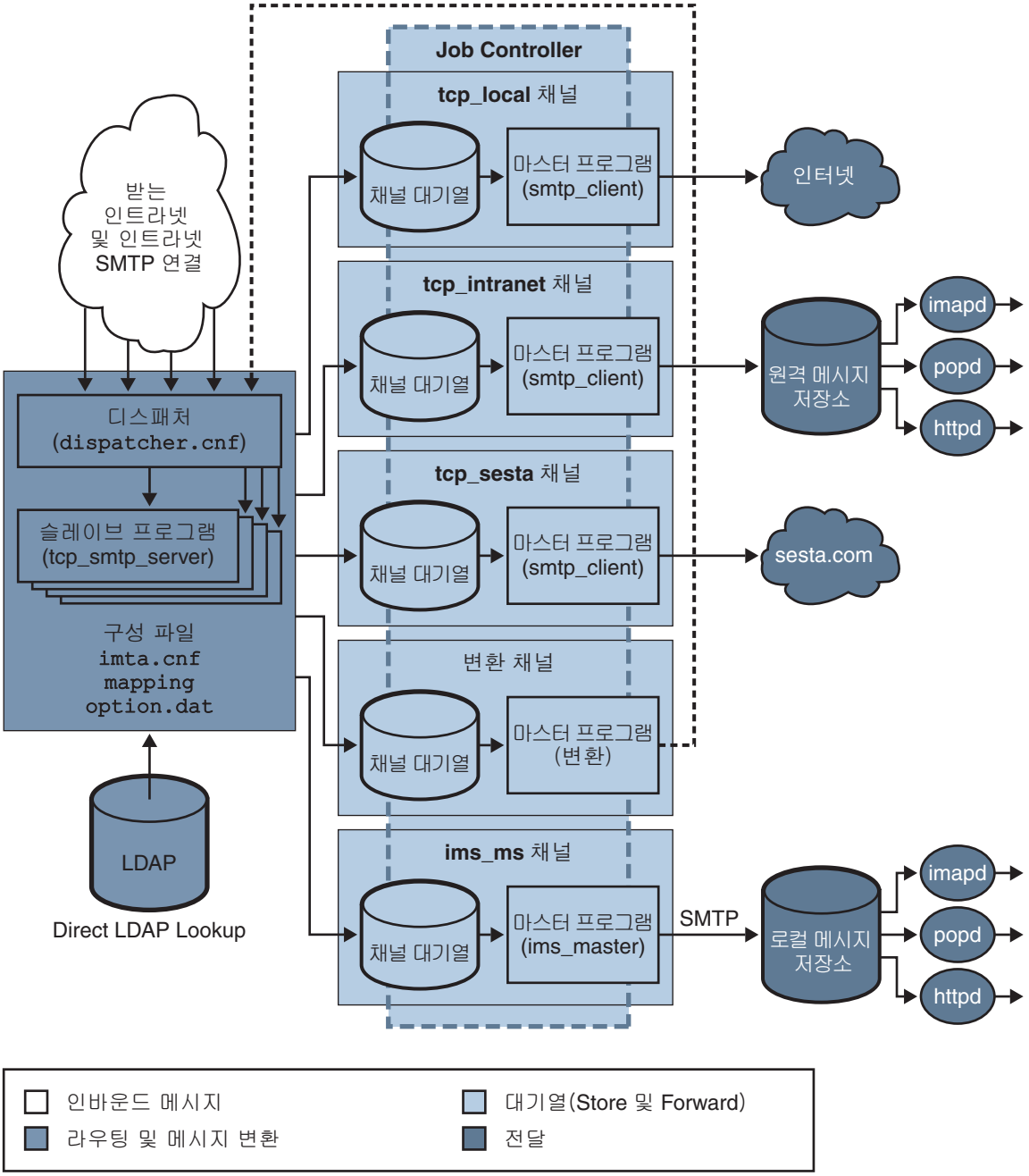

Job Controller
tcp_local 채널
채널 대기열
마스터 프로그램
(smtp_client)
인터넷
받는
인트라넷
및 인트라넷
SMTP 연결
tcp_intranet 채널
채널 대기열
마스터 프로그램
(smtp_client)
원격 메시지
저장소
imapd
popd
httpd
디스패처
(dispatcher.cnf)
슬레이브 프로그램
(tcp_smtp_server)
tcp_sesta 채널
채널 대기열
마스터 프로그램
(smtp_client)
sesta.com
구성 파일
imta.cnf
mapping
option.dat
변환 채널
채널 대기열
마스터 프로그램
(변환)
LDAP
Direct LDAP Lookup
ims_ms 채널
채널 대기열
마스터 프로그램
(ims_master)
SMTP
로컬 메시지
저장소
imapd
popd
httpd
인바운드 메시지
라우팅 및 메시지 변환
대기열(Store 및 Forward)
전달

그림 8–2 MTA 구조

8.2 MTA 구조 및 메시지 흐름 개요

이 절에서는 MTA 구조 및 메시지 흐름(그림 8–2)의 개요를 설명합니다. MTA는 상당히 복잡한 구성 요소이며, 이 그림은 시스템을 통과하는 메시지 흐름을 **단순하게** 표현한 것임을 유의하십시오. 실제로 이 그림은 시스템을 통과하는 모든 메시지 흐름을 정확하게 나타내지 않습니다. 그러나 개념적 논의를 위해서는 이 그림으로 충분할 것입니다.

8.2.1 디스패처 및 SMTP 서버(슬레이브 프로그램)

메시지는 SMTP 세션을 통해 인터넷 또는 인트라넷에서 MTA로 들어옵니다. MTA가 SMTP 연결에 대한 요청을 받으면 MTA **디스패처**(다중 스레드 연결 디스패칭 에이전트)는 SMTP 세션을 처리하기 위해 **슬레이브** 프로그램(`tcp_smtp_server`)을 실행합니다. 디스패처는 각 서비스에 대한 다중 스레드 프로세스 풀을 유지 관리합니다. 추가 세션이 요청되면 디스패처는 각 세션을 처리하기 위해 SMTP 서버 프로그램을 활성화합니다. 디스패처 프로세스 풀의 프로세스는 동시에 여러 연결을 처리할 수 있습니다. 디스패처와 슬레이브 프로그램은 서로 협력하여 각각의 받는 메시지에 대한 여러 다른 기능을 수행합니다. 세 가지 기본 기능은 다음과 같습니다.

- 메시지 차단 - 지정된 IP 주소, 메시지 주소, 포트, 채널, 헤더 문자열 등에 기초하여 메시지를 차단할 수 있습니다(18 장).
- 주소 변경 - 받는 `From:` 또는 `To:` 주소를 다른 형식으로 다시 쓸 수 있습니다.
- 채널 대기열에 포함 - 메시지를 보내야 하는 채널을 결정하기 위해 다시 쓰기 규칙을 통해 주소가 실행됩니다.

자세한 내용은 173 페이지 “8.3 디스패처”를 참조하십시오.

8.2.1.1 라우팅 및 주소 다시 쓰기

SMTP 서버가 메시지를 대기열에 포함시키지만 변환 채널 및 재처리 채널을 비롯한 여러 다른 채널도 이를 수행할 수 있습니다. 이 전달 단계가 진행하는 동안에 여러 작업이 수행되지만 기본 작업은 다음과 같습니다.

- 별칭 확장
- 다음 두 가지 작업을 수행하는 다시 쓰기 규칙을 통해 주소 실행
  - 주소의 도메인 부분을 적절한 형식으로 다시 쓰기
  - 적절한 채널 대기열로 메시지 보내기

채널

채널은 메시지 처리에 사용되는 기본 MTA 구성 요소입니다. 채널은 다른 시스템(예: 다른 MTA, 다른 채널 또는 로컬 메시지 저장소)과의 메시지 연결을 나타냅니다. 메일이 들어오면 메시지의 소스 및 대상에 따라 각기 다른 메시지에 다른 라우팅 및 처리가

필요합니다. 예를 들어, 로컬 메시지 저장소로 전달할 메일, 인터넷에 전달할 메일, 메일 시스템 내의 다른 MTA로 전달할 메일 등은 서로 다른 방식으로 처리됩니다. 채널은 각 연결에 필요한 처리 및 라우팅을 사용자 정의하기 위한 기법을 제공합니다. 기본 설치에서 대부분의 메시지는 인터넷, 인트라넷 및 로컬 메시지를 처리하는 채널로 이동합니다.

특정 상황을 위한 특수한 채널을 만들 수도 있습니다. 예를 들어, 특정 인터넷 도메인이 메시지를 매우 느린 속도로 처리하기 때문에 이 도메인으로 주소 지정된 메일이 MTA의 성능을 저하시킨다고 가정해 봅니다. 이 경우 느린 도메인으로 주소 지정된 메시지를 위한 특수한 처리를 제공하는 특정 채널을 만들어 이 도메인 병목 현상을 줄일 수 있습니다.

주소의 도메인 부분은 메시지를 대기열에 포함시킬 채널을 결정합니다. 도메인을 읽고 적절한 채널을 결정하는 기법을 다시 쓰기 규칙이라고 부릅니다(174 페이지 “8.4 다시 쓰기 규칙” 참조).

채널은 일반적으로 채널 대기열과 **마스터 프로그램**이라고 부르는 채널 처리 프로그램으로 구성됩니다. 슬레이브 프로그램이 메시지를 적절한 채널 대기열로 전달한 후 마스터 프로그램은 원하는 처리 및 라우팅을 수행합니다. 다시 쓰기 규칙과 마찬가지로 채널은 `imta.cnf` 파일에서 지정 및 구성합니다. 채널 항목의 예는 다음과 같습니다.

```
tcp_intranet smtp mx single_sys subdirs 20 noreverse maxjobs 7 SMTP_POOL
maytlsserver allowswitchchannel saslswitchchannel tcp_auth
tcp_intranet-daemon
```

이 경우에 첫 번째 단어 `tcp_intranet`은 채널 이름입니다. 마지막 단어는 채널 태그라고 부릅니다. 그 사이에 있는 단어는 채널 키워드라고 부르며 메시지가 처리되는 방법을 지정합니다. 수백 개의 다른 키워드를 사용하여 메시지를 다양한 방법으로 처리할 수 있습니다. 채널 키워드에 대한 자세한 내용은 12 장에 설명되어 있습니다.

메시지 전달

메시지가 처리된 후 마스터 프로그램은 메시지의 전달 경로를 따라 다음 정지 위치로 메시지를 보냅니다. 이 위치는 의도한 수신자의 메일함, 다른 MTA 또는 심지어 다른 채널이 될 수 있습니다. 다른 채널로 전달하는 것은 이 그림에 나와 있지 않지만 실제로는 흔히 볼 수 있습니다.

주소와 수신된 필드의 로컬 부분이 일반적으로 7비트 문자라는 점을 주의하시기 바랍니다. MTA는 이러한 필드에서 8비트 문자를 읽을 경우 각 8비트 문자를 별표로 바꿉니다.

8.3 디스패처

디스패처는 여러 다중 스레드 서버 프로세스가 SMTP 연결 서비스에 대한 역할을 공유할 수 있게 하는 다중 스레드 디스패칭 에이전트입니다. 디스패처를 사용하면 모두 동일한 포트에 대한 연결을 처리하는 여러 다중 스레드 SMTP 서버 프로세스를 동시에 실행할 수 있습니다. 또한 각 서버는 하나 이상의 활성 연결을 가질 수 있습니다.

디스패처는 자체 구성에 나열된 TCP 포트에 대한 중앙 수신기의 역할을 수행합니다. 연결이 설정된 후 디스패처는 정의된 각 서비스에 대해 하나 이상의 SMTP 서버 프로세스를 만들어 연결을 처리할 수 있습니다.

일반적으로 정의된 TCP 포트에 대한 연결을 수신하면 디스패처는 해당 포트의 사용 가능한 작업자 프로세스 풀에서 서비스를 검사하고 새 연결을 위한 최적의 후보를 선택합니다. 적절한 후보를 사용할 수 없는 경우 디스패처는 구성에서 허용하는 경우에 한하여 새 작업자 프로세스를 만들어 새 연결과 후속 연결을 처리할 수 있습니다. 또한 디스패처는 이후의 받는 연결을 예상하여 새 작업자 프로세스를 만들 수도 있습니다. 디스패처의 다양한 서비스 제어를 조정하고 특히 작업자 프로세스 수와 각 작업자 프로세스가 처리하는 연결 수를 제어하는 데 사용할 수 있는 여러 구성 옵션이 존재합니다.

자세한 내용은 229 페이지 “10.4.4 디스패처 구성 파일”을 참조하십시오.

8.3.1 서버 프로세스 작성 및 만료

디스패처 내의 자동 작업 관리 기능은 새 서버 프로세스의 작성과 오래된 또는 유휴 서버 프로세스의 만료를 제어합니다. 디스패처의 동작을 제어하는 기본 옵션은 `MIN_PROCS` 및 `MAX_PROCS`입니다. `MIN_PROCS`는 여러 서버 프로세스를 준비하고 받는 연결을 대기하여 보증된 서비스 수준을 제공합니다. 반면, `MAX_PROCS`는 주어진 서비스에 대해 동시에 활성화할 수 있는 서버 프로세스 수에 대한 상한값을 설정합니다.

최대한의 연결을 이미 처리하고 있거나 프로세스의 종료가 예약되었기 때문에 현재 실행 중인 서버 프로세스가 연결을 수락하지 못할 수 있습니다. 이 경우 디스패처는 이후의 연결을 지원하기 위해 추가 프로세스를 만들 수 있습니다.

`MIN_CONNS` 및 `MAX_CONNS` 옵션은 서버 프로세스 간에 연결을 분산시킬 수 있는 기법을 제공합니다. `MIN_CONNS`는 서버 프로세스를 “busy enough”(충분히 사용 중)로 플래그 지정하는 연결 수를 지정하고 `MAX_CONNS`는 서버 프로세스에 적용할 수 있는 “busiest”(최대한 사용 중)로 지정합니다.

일반적으로 디스패처는 현재 서버 프로세스 수가 `MIN_PROCS`보다 작거나 모든 기존 서버 프로세스가 “busy enough”(충분히 사용 중)이고 현재 활성화된 연결의 각 수가 최소한 `MIN_CONNS`인 경우 새 서버 프로세스를 만듭니다.

예를 들어, UNIX 시스템 `kill` 명령에 의해 서버 프로세스가 예기치 않게 종료할 경우 디스패처는 새 연결이 들어올 때와 마찬가지로 새 서버 프로세스를 만듭니다.

디스패처 구성에 대한 자세한 내용은 229 페이지 “10.4.4 디스패처 구성 파일”을 참조하십시오.

8.3.2 디스패처 시작 및 중지

디스패처를 시작하려면 다음 명령을 실행합니다.

```
start-msg dispatcher
```

이 명령은 디스패처가 관리하도록 구성된 MTA 구성 요소를 시작하기 위해 이전에 사용되던 다른 모든 `start-msg` 명령을 포함하므로 이러한 이전 명령은 더 이상 사용되지 않습니다. 특히 `imsimta start smtp`를 더 이상 사용해서는 안 됩니다. 폐기된 명령을 실행하려고 하면 MTA는 경고를 표시합니다.

디스패처를 종료하려면 다음 명령을 실행합니다.

```
stop-msg dispatcher
```

디스패처 종료 시에 서버 프로세스에서 수행되는 작업은 기본 TCP/IP 패키지에 따라 달라집니다. 디스패처에 적용되는 MTA 구성 또는 옵션을 수정할 경우 새 구성 또는 옵션이 적용되도록 디스패처를 다시 시작해야 합니다.

디스패처를 다시 시작하려면 다음 명령을 실행합니다.

```
imsimta restart dispatcher
```

디스패처를 다시 시작하면 현재 실행 중인 디스패처가 종료되고 새 디스패처가 즉시 시작됩니다.

8.4 다시 쓰기 규칙

다시 쓰기 규칙은 다음을 결정합니다.

- 주소의 도메인 부분을 적절한 또는 원하는 형식으로 다시 쓰는 방법
- 주소를 다시 쓴 후에 메시지를 대기열에 포함시켜야 하는 채널

각 다시 쓰기 규칙은 **패턴** 및 **템플리트**로 구성됩니다. 패턴은 주소의 도메인 부분에 대해 일치하는 문자열입니다. 템플리트는 도메인 부분이 패턴과 일치할 경우 수행되는 작업을 지정합니다. 템플리트는 1) 주소를 다시 쓰는 방법을 지정하는 명령 집합(즉, 제어 문자열) 및 2) 메시지를 보내야 하는 채널 이름의 두 부분으로 구성됩니다. 주소가 다시 작성된 후 의도한 수신자에게 전달되도록 메시지가 대상 채널의 대기열에 포함됩니다.

다시 쓰기 규칙의 예는 다음과 같습니다.

```
siroe.com $U%$D@tcp_siroe-daemon
```

siroe.com은 도메인 패턴입니다. siroe.com을 포함하는 주소를 가진 모든 메시지는 템플리트 명령($U%$D)에 따라 다시 작성됩니다. $U는 다시 작성된 주소가 같은 아이디를 사용하도록 지정합니다. %는 다시 작성된 주소가 같은 도메인 구분자를 사용하도록 지정합니다. $D는 다시 작성된 주소가 패턴에서 일치했던 같은 도메인 이름을 사용하도록 지정합니다. @tcp_siroe-daemon은 다시 작성된 주소를 가진 메시지가 tcp_siroe-daemon이라는 채널로 보내지도록 지정합니다. 자세한 내용은 11 장을 참조하십시오.

다시 쓰기 규칙의 구성에 대한 자세한 내용은 211 페이지 “10.2 MTA 구성 파일” 및 11 장을 참조하십시오.

8.5 채널

채널은 메시지를 처리하는 기본 MTA 구성 요소입니다. 채널은 다른 컴퓨터 시스템 또는 시스템 그룹과의 연결을 나타냅니다. 실제 하드웨어 연결 및/또는 소프트웨어 전송은 채널마다 크게 다를 수 있습니다.

채널은 다음 기능을 수행합니다.

- 메시지를 원격 시스템으로 전송하고 전송 후에는 대기열에서 메시지를 삭제합니다.
- 원격 시스템의 메시지를 수락하고 적절한 채널 대기열에 넣습니다.
- 메시지를 로컬 메시지 저장소에 전달합니다.
- 특수한 처리를 위해 메시지를 프로그램에 전달합니다.

메시지는 MTA로 들어올 때 채널에 의해 대기열에 포함되고, 나갈 때 대기열에서 제외됩니다. 일반적으로 메시지는 특정 채널을 통해 들어가고 다른 채널에 의해 나옵니다. 채널은 메시지를 대기열에서 제외하거나, 메시지를 처리하거나, 메시지를 다른 MTA 채널의 대기열에 포함시킬 수 있습니다.

이 절은 다음과 같은 하위 절로 구성되어 있습니다.

8.5.1 마스터 및 슬레이브 프로그램

항상 그런 것은 아니지만 채널은 일반적으로 마스터 및 슬레이브의 두 프로그램과 관련됩니다. 슬레이브 프로그램은 다른 시스템에서 메시지를 수락하고 채널의 메시지 대기열에 추가합니다. 마스터 프로그램은 채널에서 다른 시스템으로 메시지를 전송합니다.

예를 들어, SMTP 채널은 메시지를 전송하는 마스터 프로그램과 메시지를 받는 슬레이브 프로그램을 갖습니다. 이러한 프로그램은 각각 SMTP 클라이언트 및 서버입니다.

마스터 채널 프로그램은 일반적으로 MTA가 작업을 시작했던 보내는 연결을 담당합니다. 마스트 채널 프로그램은 다음을 수행합니다.

- 처리를 위해 로컬 요청에 응답하여 실행됩니다.
- 채널 메시지 대기열에서 메시지를 뺍니다.
- 대상 형식이 대기열에 넣은 메시지의 형식과 같지 않을 경우 필요에 따라 주소, 헤더 및 내용 변환을 수행합니다.
- 메시지의 네트워크 전송을 시작합니다.

슬레이브 채널 프로그램은 일반적으로 MTA가 외부 요청에 응답하는 받는 연결을 수락합니다. 슬레이브 채널 프로그램을 다음을 수행합니다.

- 외부 이벤트에 응답하여 또는 로컬 요청에 따라 실행됩니다.
- 메시지를 채널의 대기열에 넣습니다. 다시 쓰기 규칙을 통해 봉투 주소를 전달하는 방법으로 대상 채널이 결정됩니다.

예를 들어, 그림 8–3에는 두 개의 채널 프로그램(채널 1 및 채널 2)이 나와 있습니다. 채널 1의 슬레이브 프로그램은 원격 시스템으로부터 메시지를 받습니다. 이 프로그램은 주소를 확인하고 필요에 따라 다시 쓰기 규칙을 적용한 다음 다시 작성된 주소에 기초하여 해당 채널 메시지 대기열에 메시지를 포함시킵니다.

마스터 프로그램은 대기열에서 메시지를 제외시키고 메시지의 네트워크 전송을 시작합니다. 마스터 프로그램이 자신의 채널 대기열에서만 메시지를 제외시킬 수 있다는 것을 유의하십시오.

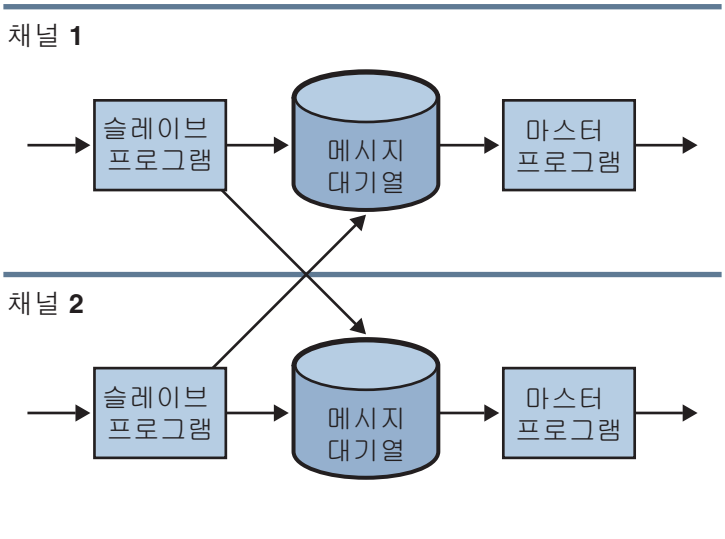


그림 8–3 마스터 및 슬레이브 프로그램

일반 채널이 마스터 및 슬레이브 프로그램을 모두 가지지만 경우에 따라서는 슬레이브 프로그램 **또는** 마스터 프로그램만 포함할 수도 있습니다. 예를 들어, Messaging Server와 함께 제공되는 `ims-ms` 채널은 그림 8–4에 나온 것처럼 로컬 메시지 저장소에 대해서만 메시지를 대기열에서 제외시키기 때문에 마스터 프로그램만 포함합니다.

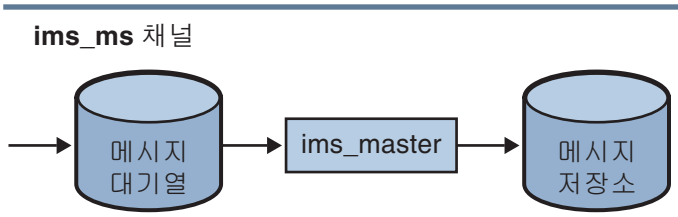


그림 8–4 ims-ms 채널

8.5.2 채널 메시지 대기열

모든 채널은 연관된 메시지 대기열을 가집니다. 메시지가 메시징 시스템으로 들어가면 슬레이브 프로그램은 메시지를 포함시킬 메시지 대기열을 결정합니다. 대기열에 넣은 메시지는 채널 대기열 디렉토리의 메시지 파일에 저장됩니다. 기본적으로 이러한 디렉토리는 *msg-svr-base*`/data/queue/` *channel*`/*`. 메시지 대기열 크기 지정에 대한 자세한 내용은 **Sun Java Communications Suite 5 Deployment Planning Guide**의 "Disk Sizing for MTA Message Queues"를 참조하십시오.

주의 – 문제가 발생할 수 있으므로 MTA 대기열 디렉토리에서 파일이나 디렉토리(즉, `imta_tailor` 파일의 `IMTA_QUEUE` 값)를 추가하지 않도록 합니다. MTA 대기열 디렉토리에 대해 별개의 파일 시스템을 사용할 경우 해당 마운트 지점 아래에 하위 디렉토리를 만들고 이 하위 디렉토리를 `IMTA_QUEUE` 값으로 지정합니다.

8.5.3 채널 정의

채널 정의는 다시 쓰기 규칙에 뒤이어 MTA 구성 파일 `imta.cnf`의 하단부에 표시됩니다(211 페이지 "10.2 MTA 구성 파일" 참조). 이 파일에 있는 첫 번째 빈 행은 다시 쓰기 규칙 섹션의 끝 부분이자 채널 정의의 시작 부분을 나타냅니다.

채널 정의는 채널 이름을 포함하며 이어서 채널 구성을 정의하는 선택적 키워드 목록과 메시지를 채널로 라우팅하기 위해 다시 쓰기 규칙에서 사용되는 고유한 채널 태그를 포함합니다. 채널 정의는 하나의 빈 행으로 구분됩니다. 채널 정의 안에 주석이 나타날 수 있지만 빈 행을 포함할 수는 없습니다.

```
[blank line]
! sample channel definition
```

Channel_Name keyword1 keyword2
Channel_Tag
[blank line]

채널 정의를 통틀어서 채널 호스트 테이블이라고 하며 개별 채널 정의를 채널 블록이라고 합니다. 예를 들어, 아래 예에서 채널 호스트 테이블은 세 개의 채널 정의 또는 블록을 포함합니다.

```
! test.cnf - An example configuration file.
!
! Rewrite Rules
      .
      .
      .

! BEGIN CHANNEL DEFINITIONS
! FIRST CHANNEL BLOCK
l
local-host

! SECOND CHANNEL BLOCK
a_channel defragment charset7 usascii
a-daemon

! THIRD CHANNEL BLOCK
b_channel noreverse notices 1 2 3
b-daemon
```

일반 채널 항목은 다음과 같이 나타납니다.

```
tcp_intranet smtp mx single_sys subdirs 20 noreverse maxjobs 7 SMTP_POOL
maytlsserver allowswitchchannel saslswitchchannel tcp_auth
tcp_intranet-daemon
```

이 경우에 첫 번째 단어 `tcp_intranet`은 채널 이름입니다. 마지막 단어 `tcp_intranet-daemon`을 **채널 태그**라고 부릅니다. 채널 태그는 메시지를 전송하기 위해 다시 쓰기 규칙에 사용되는 이름입니다. 채널 이름과 채널 태그 사이의 단어를 채널 **키워드**라고 하며, 메시지가 처리되는 방법을 지정합니다. 수백 개의 다른 키워드를 사용하여 메시지를 다양한 방법으로 처리할 수 있습니다. 채널 키워드의 전체 목록과 자세한 내용은 12 장을 참조하십시오.

채널 호스트 테이블은 Messaging Server가 사용할 수 있는 채널과 각 채널에 연관되는 시스템의 이름을 정의합니다.

UNIX 시스템에서 파일의 첫 번째 채널 블록은 항상 로컬 채널 `l`을 설명합니다. (로컬 채널 앞에 놓일 수 있는 `defaults` 채널은 예외입니다.) 로컬 채널은 라우팅 결정을 내리고 UNIX 메일 도구에 의해 보내진 메일을 전송하는 데 사용됩니다.

MTA 옵션 파일 option.dat에서 채널에 대한 전역 옵션을 설정하거나 채널 옵션 파일의 특정 채널에 대한 옵션을 설정할 수도 있습니다. 옵션 파일에 대한 자세한 내용은 230 페이지 “10.4.6 옵션 파일” 및 228 페이지 “10.4.2 TCP/IP(SMTP) 채널 옵션 파일”을 참조하십시오. 채널 구성에 대한 자세한 내용은 12 장을 참조하십시오. MTA 채널 작성에 대한 자세한 내용은 211 페이지 “10.2 MTA 구성 파일”을 참조하십시오.

8.6 MTA 디렉토리 정보

MTA는 각 메시지를 처리할 때 지원되는 사용자, 그룹 및 도메인에 대한 디렉토리 정보에 액세스해야 합니다. 이 정보는 LDAP 디렉토리 서비스에 저장됩니다. MTA는 LDAP 디렉토리에 직접 액세스합니다. 이에 대한 자세한 내용은 9 장에 설명되어 있습니다.

8.7 작업 제어기

메시지가 채널의 대기열에 배치될 때마다 작업 제어기는 해당 메시지를 전달하기 위해 실행 중인 작업이 있는지 확인합니다. 여기에는 새 작업 프로세스를 시작하거나, 스레드를 추가하거나, 단순히 작업이 이미 실행 중인지 확인하는 것이 포함됩니다. 채널 또는 풀에 대한 작업 제한에 도달하여 작업을 시작할 수 없을 경우 작업 제어기는 다른 작업이 종료할 때까지 기다립니다. 작업 제한을 더 이상 초과하지 않으면 작업 제어기는 다른 작업을 시작합니다.

채널 작업은 작업 제어기 내의 처리 풀 안에서 실행됩니다. 풀은 채널 작업이 실행되는 “장소”로 생각할 수 있습니다. 풀은 작업 세트가 풀 외부의 작업과 자원을 놓고 경쟁하지 않고도 작동할 수 있는 컴퓨팅 영역을 제공합니다. 풀에 대한 자세한 내용은 231 페이지 “10.4.8 작업 제어기 파일” 및 356 페이지 “12.5.4 채널 실행 작업의 처리 풀”을 참조하십시오.

채널에 대한 작업 제한은 maxjobs 채널 키워드에 의해 결정되며 풀에 대한 작업 제한은 풀의 JOB_LIMIT 옵션에 의해 결정됩니다.

Messaging Server는 일반적으로 모든 메시지를 즉시 전달하려고 시도합니다. 그러나 첫 번째 시도에서 메시지를 전달할 수 없는 경우 해당 backoff 키워드에 지정된 기간 동안 메시지가 지연됩니다. backoff 키워드에 지정된 시간이 경과하자마자 지연된 메시지를 전달할 수 있으며 필요한 경우 메시지를 처리하기 위해 채널 작업이 시작됩니다.

현재 처리 중인 메시지과 처리 대기 중인 메시지에 대한 작업 제어기의 메모리 내장 데이터 구조는 일반적으로 MTA 대기열 영역의 디스크에 저장된 전체 메시지 파일 집합을 반영합니다. 그러나 디스크의 메시지 파일 백로그가 작업 제어기의 메모리 내장 데이터 구조 크기 제한을 초과하기에 충분할 만큼 작성될 경우 작업 제어기는 디스크의 전체 메시지 파일 중 일부만 메모리에서 추적합니다. 작업 제어기는 메모리에서 추적 중인 메시지만 처리합니다. 메모리 내장 저장소를 비워야 할 정도로 많은 메시지가 전달된 경우 작업 제어기는 MTA 대기열 영역을 스캔하여 메시지 목록을 업데이트함으로써 메모리 내장 저장소를 자동으로 갱신합니다. 그런 다음 작업

제어기는 디스크에서 방금 검색한 추가 메시지 파일의 처리를 시작합니다. 작업 제어기는 MTA 대기열 영역에 대한 이러한 스캔 작업을 자동으로 수행합니다.

이전에는 작업 제어기가 대기열 디렉토리에 있는 모든 파일을 발견된 순서대로 읽었습니다. 지금은 여러 채널 대기열 디렉토리를 한 번에 읽습니다. 따라서 시작, 재시작 및 max_messages의 초과 이후에 보다 효과적인 작동이 수행됩니다. 한 번에 읽을 디렉토리 수는 작업 제어기 옵션 Rebuild_Parallel_Channel로 조절합니다. 1부터 100 사이의 값으로 설정할 수 있으며 기본값은 12입니다.

사이트에서 과도한 메시지 백로그가 정기적으로 발생할 경우 MAX_MESSAGES 옵션을 사용하여 작업 제어기를 조정할 수 있습니다. 작업 제어기가 더 많은 메모리를 사용할 수 있게 MAX_MESSAGES 옵션 값을 늘리면 메시지 백로그가 작업 제어기의 메모리 내장 캐시를 오버플로하는 경우를 줄일 수 있습니다. 또한 이 경우 작업 제어기가 MTA 대기열 디렉토리를 스캔해야 할 때와 관련된 오버헤드가 줄어듭니다. 그러나 작업 제어기가 메모리 내장 캐시를 다시 작성해야 할 경우 캐시가 더 크기 때문에 프로세스에 더 많은 시간이 걸린다는 것을 유의하십시오. 또한 작업 제어기는 시작 또는 재시작될 때마다 MTA 대기열 디렉토리를 스캔해야 하므로 과도한 메시지 백로그가 있다는 것은 그렇지 않을 때보다 작업 제어기를 시작 또는 재시작할 때 많은 오버헤드가 발생한다는 것을 의미합니다.

풀과 작업 제어기 구성에 대한 자세한 내용은 231 페이지 "10.4.8 작업 제어기 파일" 및 352 페이지 "12.5 메시지 처리 및 전달 구성"을 참조하십시오.

8.7.1 작업 제어기 시작 및 중지

작업 제어기를 시작하려면 다음 명령을 실행합니다.

```
start-msg job_controller
```

작업 제어기를 종료하려면 다음 명령을 실행합니다.

```
stop-msg job_controller
```

작업 제어기를 다시 시작하려면 다음 명령을 실행합니다.

```
imsimta restart job_controller
```

작업 제어기를 다시 시작하면 현재 실행 중인 작업 제어기가 종료되고 새 작업 제어기가 바로 시작됩니다.

◆ ◆ ◆ 9 장

9 MTA 주소 변환 및 라우팅

Messaging Server 6 2003Q4 이전에 Messaging Server는 LDAP 서버에 저장된 정보로 컴파일된 데이터베이스에서 모든 사용자, 도메인 및 그룹 데이터에 액세스했습니다. LDAP 서버에서 디렉토리 정보가 업데이트되면 데이터베이스 정보는 `dirsync`라는 프로그램과 동기화되었습니다. Messaging Server MTA는 이제 LDAP 디렉토리에 직접 액세스합니다. 이 장에서는 Direct LDAP 데이터 액세스를 사용한 MTA의 데이터 흐름에 대해 설명합니다. 이 장은 다음 내용으로 구성되어 있습니다.

9.1 Direct LDAP 알고리즘 및 구현

다음 절에서는 Direct LDAP 처리에 대해 설명합니다.

9.1.1 도메인의 로컬 여부 확인

user@domain 형식의 주소에서 시작되는 주소 변환 및 라우팅 프로세스는 우선 *domain*이 로컬인지 여부를 검사합니다. 이 절은 다음과 같은 하위 절로 구성되어 있습니다.

9.1.1.1 다시 쓰기 규칙 방법

주어진 문자열을 검사하여 로컬로 처리해야 하는 도메인인지 여부를 확인하는 기능이 MTA 다시 쓰기 규칙 방법에 추가되었습니다. 이 새로운 기능은 $V 또는 $Z 메타 문자에 의해 활성화됩니다. 이러한 새 메타 문자는 기존 $N, $M, $Q 및 $C 메타 문자와 구문적으로 유사합니다. 즉, 뒤에 패턴 문자열이 옵니다. $N, $M, $Q 및 $C의 경우 소스 또는 대상 채널에 대해 패턴이 일치됩니다. $V 및 $Z의 경우 패턴은 도메인이며 로컬인지 여부를 확인하는 검사가 수행됩니다. $V의 경우 로컬이 아닌 도메인에 대해 규칙이 실패하며 $Z의 경우 로컬 도메인에 대해 규칙이 실패합니다.

이러한 메타 문자의 처리는 다음 절차에 따라 구현됩니다.

1. Messaging Server는 현재 도메인이 디렉토리의 유효한 도메인 항목과 일치하는지 여부를 확인합니다. 항목이 없으면 단계 3으로 가십시오.
2. 도메인에 디렉토리의 항목이 있을 때는 LDAP_DOMAIN_ATTR_ROUTING_HOSTS MTA 옵션에서 지정한 속성(기본 mailRoutingHosts)이 해당 도메인 항목에서 검색됩니다. 이 속성이 존재할 경우 이 도메인의 사용자를 처리할 수 있는 호스트 집합이 나열됩니다. 이 목록은 local.hostname configutil 매개 변수에 지정된 호스트 및 local.imta.hostnamealiases configutil 매개 변수에 지정된 호스트 목록과 비교됩니다. 이러한 옵션은 각각 LDAP_LOCAL_HOST 및 LDAP_HOST_ALIAS_LIST MTA 옵션으로 무시할 수 있습니다. 일치하는 항목이 있거나 도메인에 속성이 존재하지 않을 경우 도메인은 로컬입니다. 일치하는 항목이 없으면 도메인은 로컬이 아닙니다.

   mailRoutingHosts 속성으로 인해 로컬이 아닌 것으로 간주되는 도메인의 처리는 ROUTE_TO_ROUTING_HOST MTA 옵션의 설정에 따라 달라집니다. 이 옵션이 0(기본값)으로 설정된 경우 주소는 단순히 로컬이 아닌 것으로 간주되며 MTA 다시 쓰기 규칙을 사용하여 라우팅을 결정합니다. 이 옵션이 1로 설정된 경우 LDAP_DOMAIN_ATTR_ROUTING_HOSTS MTA 옵션에 나열된 첫 번째 값으로 구성된 소스 경로가 주소의 앞에 놓입니다.
3. 도메인 항목을 찾을 수 없는 경우 도메인의 왼쪽에서 구성 요소를 제거하고 단계 1로 이동합니다. 구성 요소가 남아 있지 않으면 단계 4를 진행합니다.

   도메인 트리를 거슬러 올라가는 이 방법은 결과적으로 domain.com이 로컬로 인식될 경우 domain.com의 모든 하위 도메인이 로컬로 인식되게 합니다. 이 동작으로 바람직하지 않은 상황이 발생할 수 있으므로 동작을 제어하기 위한 MTA 옵션 DOMAIN_UPLEVEL이 제공됩니다. 특히 DOMAIN_UPLEVEL의 비트 0(값 = 1)이 지워진 경우 제거된 도메인 구성 요소로 재시도할 수 없게 됩니다. DOMAIN_UPLEVEL의 기본값은 0입니다.

4. 이제 부속 도메인 검사를 수행해야 합니다. 부속 도메인에는 도메인 항목이 없고, 오히려 하나 이상의 사용자 항목에 특별한 도메인 속성을 추가함으로써 지정됩니다. 후속 도메인 검사는 DOMAIN_MATCH_URL MTA 옵션에 지정된 LDAP URL을 사용하여 LDAP 검색을 시작하는 방법으로 수행됩니다. 이 옵션의 값은 다음과 같이 설정해야 합니다.

   ```
   ldap:///$B?msgVanityDomain?sub?(msgVanityDomain=$D)
   ```

   $B는 local.ugldapbasedn configutil 매개 변수의 값을 대체합니다(이는 디렉토리에서 사용자 트리의 기반임). LDAP_USER_ROOT MTA 옵션을 사용하여 특히 MTA에 대해 이 configutil 옵션의 값을 무시할 수 있습니다.

   이 검색의 실제 반환 값은 중요하지 않습니다. 중요한 것은 반환할 값이 존재하는지 여부입니다. 반환 값이 존재할 경우 도메인은 로컬로 간주되며 그렇지 않을 경우 도메인은 로컬이 아닌 것으로 간주됩니다.

9.1.1.2 도메인 로컬 여부의 도메인 맵 확인

디렉토리에서 유효한 도메인 항목을 찾기 위해 수행되는 단계가 무엇인지 잘 기억하는 것이 좋습니다. 이러한 단계는 스키마 수준별로 다릅니다. Sun LDAP Schema, v.1의 경우 이러한 단계는 다음과 같습니다.

1. 도메인을 도메인 트리의 기본 DN으로 변환합니다. 이 작업은 도메인을 일련의 dc 구성 요소로 변환한 다음 도메인 루트 접미어를 추가하는 방법으로 수행합니다. 기본 접미어는 service.dcroot configutil 매개 변수에서 얻습니다. 기본 접미어는 o=internet입니다. 따라서 a.b.c.d 형식의 도메인은 일반적으로 dc=a,dc=b,dc=c,dc=d,o=internet으로 변환됩니다. service.dcroot configutil 매개 변수는 LDAP_DOMAIN_ROOT MTA 옵션을 설정하여 무시할 수 있습니다.
2. 단계 1에서 발견된 기본 DN과 inetDomain 또는 inetDomainAlias 객체 클래스를 가진 항목을 찾습니다. 이 목적에 사용되는 검색 필터는 LDAP_DOMAIN_FILTER_SCHEMA1 MTA 옵션을 설정하여 무시할 수 있습니다. 이 옵션의 기본값은 (|(objectclass=inetDomain)(objectclass=inetdomainalias))입니다.
3. 아무 것도 발견되지 않을 경우 실패와 함께 종료합니다.
4. 항목의 객체 클래스를 inetDomain에서 찾은 경우, 해당 도메인 항목에 관련된 inetDomainBaseDn 속성이 있는지 확인하십시오. 이 속성이 있으면 사용자 항목에 대한 후속 검색에 사용되도록 저장한 다음 처리가 종료됩니다. 이 속성이 없으면 해당 항목은 도메인 별칭으로 가정되고 단계 5로 처리가 계속됩니다. inetDomainBaseDN 대신 MTA 옵션 LDAP_DOMAIN_ATTR_BASEDN을 사용할 수 있습니다.
5. 항목은 도메인 별칭이어야 합니다. aliasedObjectName 속성에서 참조한 새 항목을 찾아 단계 4로 돌아가십시오. aliasedObjectName 속성이 없으면 오류와 함께 처리가 종료됩니다. MTA 옵션 LDAP_DOMAIN_ATTR_ALIAS를 사용하여 aliasedObjectName 속성 사용을 대체할 수 있습니다.

   처리가 단계 4로 돌아가는 일은 한 번 정도만 발생할 수 있다는 점을 주의하십시오. 도메인 별칭에서 도메인 별칭을 가리키는 일은 허용되지 않습니다.

Sun LDAP Schema 2에서는 수행되는 작업이 훨씬 더 간단합니다. 디렉토리 도메인이 sunPreferredDomain 또는 associatedDomain 속성 값으로 표시되는 sunManagedOrganization 객체 클래스가 있는 항목을 검색합니다. 필요한 경우 sunPreferredDomain 및 associatedDomain 속성의 사용을 각각 MTA 옵션 LDAP_ATTR_DOMAIN1_SCHEMA2 및 LDAP_ATTR_DOMAIN2_SCHEMA2를 사용하여 무시할 수 있습니다. 검색은 service.dcroot configutil 매개 변수로 지정된 루트 하에서 수행됩니다. service.dcroot configutil 매개 변수는 LDAP_DOMAIN_ROOT MTA 옵션을 설정하여 무시할 수 있습니다. 아울러 스키마 2의 도메인 항목은 inetDomainBaseDn 속성을 가지지 않아도 됩니다. 해당 속성을 가지지 않은 경우 사용자 트리의 기본이 도메인 항목 자체인 것으로 간주됩니다.

두 가지 MTA 옵션이 사용자 기반 도메인 이름으로부터 보다 효율적으로 도메인을 조회하도록 지원합니다. LDAP_BASEDN_FILTER_SCHEMA1은 사용자 기반 도메인 이름 검색 시 Schema 1 도메인 식별에 사용되는 필터를 지정하는 문자열입니다. MTA 옵션이 지정된 경우, 기본값은 LDAP_DOMAIN_FILTER_SCHEMA1의 값입니다. 옵션이 지정되지 않은 경우 기본값은 (objectclass=inetDomain)입니다. LDAP_BASEDN_FILTER_SCHEMA2는 사용자 기반 도메인 검색 시 Schema 2 도메인 식별에 사용되는 추가 필터 요소를 지정하는 문자열입니다. MTA 옵션이 지정된 경우, 기본값은 LDAP_DOMAIN_FILTER_SCHEMA2입니다. 옵션이 지정되지 않은 경우 기본값은 빈 문자열입니다.

9.1.1.3 도메인 로컬 여부 정보의 캐싱

도메인 다시 쓰기 작업이 수행되는 빈도와 디렉토리 쿼리(특히 부속 도메인 검사)의 비용으로 인해 도메인에 대한 부정적 및 긍정적 표시를 모두 캐시해야 합니다. 이 작업은 동적으로 확장되는 메모리 내장의 개방형 체인 해시 테이블을 통해 구현됩니다. 캐시의 최대 크기는 DOMAIN_MATCH_CACHE_SIZE MTA 옵션(기본값 100000)으로 설정하며 캐시의 항목에 대한 시간 초과는 DOMAIN_MATCH_CACHE_TIMEOUT MTA 옵션(기본값 600초)으로 설정합니다.

9.1.1.4 오류 처리

이 프로세스 도중에 발생하는 임시 서버 오류를 신중하게 처리해야 하는데 이는 이 오류가 발생할 경우 주어진 도메인이 로컬인지 여부를 알 수 없기 때문입니다. 기본적으로 이러한 경우에는 두 가지 결과가 가능합니다.

1. 주소를 나중에 다시 시도하라는 임시(4*xx*) 오류를 클라이언트에게 반환합니다.
2. 주소를 수락하지만 재처리 채널에서 주소를 대기시켜 나중에 로컬로 다시 시도할 수 있게 합니다.

이러한 두 옵션이 모든 경우에 적합한 것은 아닙니다. 예를 들어, 결과 1은 원격 SMTP 중계와 통신할 때 적합합니다. 그러나 결과 2는 로컬 사용자로부터의 SMTP 제출을 처리할 때 적합합니다.

동일한 패턴을 가진 여러 규칙을 사용하여 일시적인 오류를 처리하는 것이 이론적으로 가능하지만 이러한 쿼리를 반복할 경우 발생하는 오버헤드는 캐시로도 처리할 수 없는 큰 부담이 됩니다. 이러한 이유로 도메인 다시 쓰기의 다음 규칙까지의 성공/실패 일치

모델은 적합하지 않습니다. 대신 MTA 옵션 DOMAIN_FAILURE에 지정된 특수한 템플리트가 도메인 조회 실패의 경우에 사용됩니다. $V 작업이 실패하면 이 템플리트는 처리 중인 현재 다시 쓰기 규칙 템플리트의 나머지 부분을 대체합니다.

9.1.1.5 도메인 검사 다시 쓰기 규칙을 위한 패턴

다른 다시 쓰기 규칙이 수행되기 전에 이 도메인 검사를 수행해야 합니다. 이 순서는 규칙의 왼쪽에서 특수한 $*를 사용하여 지정합니다. $* 패턴은 다른 모든 규칙에 앞서 검사됩니다.

9.1.1.6 모든 방법 사용

지금까지 설명된 모든 방법을 고려할 때 imta.cnf에서 필요한 새 다시 쓰기 규칙은 다음과 같습니다.

```
$*      $E$F$U%$H$V$H@localhost
```

또한 option.dat 파일에서 DOMAIN_FAILURE MTA 옵션 값은 다음과 같아야 합니다.

```
reprocess-daemon$Mtcp_local$1M$1~-error$4000000?Temporary lookup failure
```

이 다시 쓰기 규칙에서 localhost는 로컬 채널과 연관된 호스트 이름입니다. 여기에 표시된 DOMAIN_FAILURE 옵션 값이 기본값이므로 정상적인 환경에서 option.dat에 표시될 필요가 없습니다.

여기에서 순서는 특히 까다롭습니다. MTA는 주소가 재작성되었지만 경로가 아직 추가되기 전에 $V를 검사합니다. 따라서 일시적인 조회 실패가 발생할 경우에 MTA에서 경로를 변경할 수 있습니다. 보류 중인 채널 일치 검사는 삽입 지점이 변경될 때마다 적용되므로 $H초 후에 @이 검사를 호출합니다. 이 검사에 성공할 경우 템플리트의 나머지 부분이 적용되며 다시 쓰기 처리가 완료됩니다. 검사에 실패할 경우 다시 쓰기는 실패하며 적용 가능한 다음 다시 쓰기 규칙을 사용하여 다시 쓰기가 계속됩니다. 일시적인 오류로 인해 검사를 수행할 수 없는 경우 DOMAIN_FAILURE MTA 옵션에 지정된 값에서 템플리트 처리가 계속됩니다. 이 템플리트 값은 우선 라우팅 호스트를 reprocess-daemon으로 설정합니다. 그런 다음 템플리트는 MTA가 동일한 종류 또는 tcp_local의 재처리 채널을 처리하고 있는지 여부를 확인합니다. MTA가 이러한 채널을 처리하는 중이면 규칙이 계속 진행되어 라우팅 호스트를 잘못된 것으로 만들고 일시적인 오류를 결과로 지정합니다. MTA가 이러한 채널을 처리하는 중이 아니면 규칙이 잘리고 성공적으로 종료하므로 재처리 채널에 주소가 다시 쓰여집니다.

9.1.2 로컬 주소의 별칭 확장

주소가 로컬 채널과 연관된 것으로 확인되고 나면 해당 주소에 자동으로 별칭 확장이 적용됩니다. 별칭 확장 프로세스는 다음을 비롯하여 여러 정보 소스를 검사합니다.

1. 별칭 파일(컴파일된 구성의 일부)

2. 별칭 데이터베이스
3. 별칭 URL

검사되는 정확한 별칭 소스와 별칭 소스가 검사되는 순서는 option.dat 파일의 ALIAS_MAGIC MTA 옵션 설정에 따라 달라집니다. Direct LDAP의 경우 이 옵션을 8764로 설정합니다. 이것은 ALIAS_URL0 MTA 옵션에 지정된 URL이 우선적으로 검사되고 ALIAS_URL1 MTA 옵션에 지정된 URL이 검사된 다음 ALIAS_URL2 MTA 옵션에 지정된 URL이 검사되고 마지막으로 별칭 파일이 검사된다는 것을 의미합니다. 이 설정이 활성화되면 별칭 데이터베이스는 검사되지 않습니다.

다음 절에서는 별칭 확장에 대해 보다 자세히 소개합니다.

- 186 페이지 "9.1.2.1 LDAP URL을 사용한 별칭 검사"
- 186 페이지 "9.1.2.2 $V 메타 문자"
- 188 페이지 "9.1.2.3 URL에서 매핑 호출"
- 188 페이지 "9.1.2.4 $R 메타 문자"
- 189 페이지 "9.1.2.5 가져올 속성 결정"
- 189 페이지 "9.1.2.6 LDAP 오류 처리"
- 189 페이지 "9.1.2.7 LDAP 결과에 대한 유효성 검사"
- 190 페이지 "9.1.2.8 부속 도메인 지원"
- 190 페이지 "9.1.2.9 Catchall 주소 지원"

9.1.2.1 LDAP URL을 사용한 별칭 검사

LDAP에서 별칭을 검사하는 것은 두 개의 특수한 LDAP URL을 별칭 URL로 지정하여 구현합니다. 첫 번째 URL은 일반 사용자와 그룹을 처리하며 부속 도메인은 후속 별칭 URL에 의해 처리됩니다. 첫 번째 URL은 다음과 같이 ALIAS_URL0으로 지정됩니다.

```
ALIAS_URL0=ldap:///$V?*?sub?$R
```

9.1.2.2 $V 메타 문자

메타 문자 확장은 URL 조회 전에 발생합니다. ALIAS_URL0 값에 사용되는 두 개의 메타 문자는 $V 및 $R입니다.

$V 메타 문자는 주소의 도메인 부분을 기본 DN으로 변환합니다. 이것은 앞의 182 페이지 "9.1.1.1 다시 쓰기 규칙 방법" 절에서 설명한 $V 다시 쓰기 규칙 메타 문자에 의해 수행되는 초기 단계와 비슷합니다. $V 처리는 다음 단계로 구성됩니다.

1. 현재 도메인의 사용자 항목에 대한 기본 DN을 가져옵니다.
2. 현재 도메인과 연결된 정규 도메인을 가져옵니다. Sun LDAP Schema 1에서 정규 도메인 이름은 도메인 항목의 inetCanonicalDomainName 속성에 의해 제공됩니다(이 속성이 존재할 경우). 이 속성이 없을 경우 정규 이름은 실제 도메인 항목의 DN에서 명시적인 방식으로 생성되는 이름입니다. 현재 도메인이 별칭일 경우 이것은 현재 도메인과 다릅니다. 정규 이름을 저장하는 데 사용되는 이름 속성은 option.dat 파일의 LDAP_DOMAIN_ATTR_CANONICAL MTA 옵션으로 무시할 수 있습니다.

Sun LDAP Schema 2에서 정규 이름은 단순히 SunPreferredDomain 속성 값입니다.

중복 사용자 항목이 있는 도메인에 대한 정규 도메인 설정을 확인하는 유틸리티가 제공됩니다. **Sun Java System Messaging Server 6.3 Administration Reference**의 "imsimta test -domain"을 참조하십시오.

3. 기본 DN이 존재할 경우 이를 $V 대신에 URL로 대체합니다.
4. 이제 이 항목에 대한 적용 가능한 모든 호스트된 도메인이 확인됩니다. This is done by comparing either the canonical domain (if bit 2 (value = 4) of DOMAIN_UPLEVEL is clear) or the current domain (if bit 2 (value = 4) of DOMAIN_UPLEVEL is set) with the service.defaultdomain configutil parameter. 서로 일치하지 않을 경우 해당 항목은 호스트된 도메인의 구성원입니다. service.defaultdomain configutil 매개 변수는 option.dat 파일에서 LDAP_DEFAULT_DOMAIN MTA 옵션을 설정하여 무시할 수 있습니다.
5. 기본 DN 확인이 실패할 경우 도메인의 왼쪽에서 구성 요소를 제거하고 단계 1로 이동합니다. 구성 요소가 남아 있지 않을 경우 대체는 실패합니다.

$V는 또한 선택 사항인 숫자 인수를 허용합니다. 1로 설정된 경우(예: $1V) 도메인 트리의 도메인 확인 실패가 무시되고 local.ugldapbasedn configutil 옵션에서 지정한 사용자 트리의 기반이 반환됩니다.

도메인의 기본 DN을 검색하려는 시도가 성공할 경우 MTA는 또한 나중에 필요한 여러 유용한 도메인 속성을 검색합니다. 검색된 속성의 이름은 option.dat 파일의 다음 MTA 옵션에 의해 설정됩니다.

- LDAP_DOMAIN_ATTR_UID_SEPARATOR(기본값: domainUidSeparator)
- LDAP_DOMAIN_ATTR_SMARTHOST(기본값: mailRoutingSmartHost)
- LDAP_DOMAIN_ATTR_CATCHALL_ADDRESS(기본값: mailDomainCatchallAddress)
- LDAP_DOMAIN_ATTR_CATCHALL_MAPPING (기본값 없음)
- LDAP_DOMAIN_ATTR_BLOCKLIMIT(기본값: mailDomainMsgMaxBlocks)
- LDAP_DOMAIN_ATTR_REPORT_ADDRESS(기본값: mailDomainReportAddress)
- LDAP_DOMAIN_ATTR_STATUS(기본값: inetDomainStatus)
- LDAP_DOMAIN_ATTR_MAIL_STATUS(기본값: mailDomainStatus)
- LDAP_DOMAIN_ATTR_CONVERSION_TAG(기본값: mailDomainConversionTag)
- LDAP_DOMAIN_ATTR_FILTER(기본값: mailDomainSieveRuleSource)
- LDAP_DOMAIN_ATTR_DISK_QUOTA(기본값: 없음)
- LDAP_DOMAIN_ATTR_MESSAGE_QUOTA(기본값: 없음)
- LDAP_DOMAIN_ATTR_AUTOREPLY_TIMEOUT(기본값: 없음)
- LDAP_DOMAIN_ATTR_NOSOLICIT(기본값: 없음)
- LDAP_DOMAIN_ATTR_OPTIN(기본값: 없음)
- LDAP_DOMAIN_ATTR_RECIPIENTLIMIT(기본값: 없음)
- LDAP_DOMAIN_ATTR_RECIPIENTCUTOFF(기본값: 없음)
- LDAP_DOMAIN_ATTR_SOURCEBLOCKLIMIT(기본값: 없음)

9.1.2.3 URL에서 매핑 호출

간혹 도메인에서 기본 DN으로의 매핑이 다른 방식으로 수행되기도 합니다. 이러한 설정을 수용하기 위해 URL 확인 프로세스는 MAT 매핑을 호출하는 기능을 가집니다. 이 작업은 다음과 같은 일반적인 형식의 메타 문자 시퀀스를 통해 수행됩니다.

$|/*mapping-name*/ *mapping-argument*|

큰따옴표(")는 콜아웃을 시작 및 종료합니다. $의 바로 뒤에 오는 문자는 매핑 이름과 인수의 구분자이며 이 문자는 매핑 이름이나 인수에 사용되는 예상된 문자 값과 충돌하지 않도록 선택해야 합니다.

9.1.2.4 $R 메타 문자

$R 메타 문자는 URL을 위한 적절한 필터를 제공합니다. 이 메타 문자의 목적은 특정 사용자나 그룹에 대한 전자 메일 주소를 포함할 수 있는 모든 속성을 검색하는 필터를 생성하는 것입니다. 검색할 속성 목록은 configutil 매개 변수 `local.imta.mailaliases`에서 제공됩니다. 이 매개 변수를 설정하지 않을 경우 `local.imta.schematag` configutil 매개 변수가 검사되며 다음과 같이 이 매개 변수의 값에 따라 적절한 기본 속성 집합이 선택됩니다.

```
sims401 mail, rfc822mailalias

nms41 mail,mailAlternateAddress

ims50 mail,mailAlternateAddress, mailEquivalentAddress
```

`local.imta.schematag`의 값은 쉼표로 구분된 목록이 될 수 있습니다. 둘 이상의 스키마가 지원될 경우 중복 항목이 제거된 결합된 속성 목록이 사용됩니다. `LDAP_SCHEMATAG` MTA 옵션을 사용하여 특히 MTA에 대한 `local.imta.schematag`의 설정을 무시할 수 있습니다.

또한 이 필터는 원래 제공되었던 주소뿐만 아니라 로컬 부분이 동일하지만 실제로 도메인 트리에서 발견된 도메인(186 페이지 "9.1.2.2 $V 메타 문자" 절의 단계 2에서 저장)을 가진 주소를 검색합니다. 도메인 트리 조회의 반복 특성은 두 개의 주소가 다를 수 있다는 것을 의미합니다. 이 추가 검사는 `option.dat` 파일에서 `DOMAIN_UPLEVEL` MTA 옵션의 비트 1(값 = 2)을 통해 제어합니다. 이 비트를 설정하면 추가 주소 검사가 활성화됩니다. `DOMAIN_UPLEVEL`의 기본값은 0입니다.

예를 들어, 도메인 `siroe.com`이 도메인 트리에 있다고 가정해 봅니다. Sun LDAP Schema, v.1을 사용 중이라고 가정하면 주소 조회는 다음과 같습니다.

```
u@host1.siroe.com
```

$R 및 ims50 `schematag`의 확장 결과로 얻어지는 필터는 다음과 같습니다.

```
(|(mail=u@siroe.com)
    (mail=u@host1.siroe.com)
    (mailAlternateAddress=u@siroe.com)
```

```
    (mailAlternateAddress=u@host1.siroe.com)
    (mailEquivalentAddress=u@siroe.com)
    (mailEquivalentAddress=u@host1.siroe.com))
```

반면, DOMAIN_UPLEVEL이 3이 아니라 1로 설정된 경우 필터는 다음과 같을 것입니다.

```
(|(mail=u@host1.siroe.com)
       (mailAlternateAddress=u@host1.siroe.com)
       (mailEquivalentAddress=u@host1.siroe.com))
```

9.1.2.5 가져올 속성 결정

반환할 속성 목록에 대해 URL에서 *가 지정된 경우 이 별표는 MTA가 사용할 수 있는 속성 목록으로 대체됩니다. 이 목록은 MTA가 소비하는 옵션을 지정하는 다양한 MTA 옵션 설정으로부터 동적으로 만들어집니다.

9.1.2.6 LDAP 오류 처리

이 시점에서 결과 URL은 LDAP 검색을 수행하는 데 사용됩니다. 일정한 LDAP 오류가 발생한 경우 일시적인 오류 표시(SMTP의 4xx 오류)와 함께 처리가 종료됩니다. LDAP 작업이 성공적이지만 결과를 생성하지 못한 경우 LDAP_DOMAIN_ATTR_CATCHALL_ADDRESS MTA 옵션에서 검색된 도메인에 대한 catchall 주소 속성이 검사됩니다. 이 속성이 설정된 경우 해당 값이 현재 주소를 대체합니다.

catchall 주소 속성이 설정되지 않은 경우 LDAP_DOMAIN_ATTR_SMARTHOST MTA 옵션에서 검색된 도메인에 대한 smarthost 속성이 검사됩니다. 이 속성이 설정된 경우 다음 형식의 주소가 작성되고

@*smarthost*: *user*@*domain*

이 결과와 함께 별칭 처리가 성공적으로 종료합니다. 또한 LDAP_DOMAIN_ATTR_CONVERSION_TAG MTA 옵션에서 얻은 도메인에 대한 변환 태그(있을 경우)가 주소에 추가되어 smarthost로 전달하기 전에 변환이 수행될 수 있게 합니다. 도메인에 대해 catchall 주소나 smarthost가 존재하지 않을 경우 이 별칭 URL의 처리가 실패로 종료합니다.

9.1.2.7 LDAP 결과에 대한 유효성 검사

LDAP 검색이 결과를 반환한 후에 항목이 하나만 존재하는지 확인하기 위해 결과가 검사됩니다. 둘 이상의 항목이 존재할 경우 사용자 또는 그룹에 대한 올바른 객체 클래스, 삭제되지 않은 상태 및 UID(사용자인 경우)를 가지는지 확인하기 위해 각 항목이 검사됩니다. 이 검사를 통과하지 않은 항목은 무시됩니다. 이 검사에 의해 여러 항목이 하나로 줄어들 경우 처리가 계속 진행됩니다. 그렇지 않을 경우에는 중복 또는 모호한 디렉토리 오류가 반환됩니다.

9.1.2.8 부속 도메인 지원

ALIAS_URL0 검사는 기본 사용자나 호스트된 도메인의 사용자에 대해 수행됩니다. 이 검사가 실패할 경우 또한 부속 도메인 검사가 수행됩니다. 부속 도메인 검사에는 다음 별칭 URL이 사용됩니다.

```
ALIAS_URL1=ldap:///$B?*?sub?(&(msgVanityDomain=$D)$R)
```

9.1.2.9 Catchall 주소 지원

마지막으로 @*host* 형식의 catchall 주소에 대한 검사가 mailAlternateAddress 속성에서 수행되어야 합니다. 이 와일드카드 지정 형식은 호스트된 도메인과 부속 도메인 모두에서 허용되므로 이에 대한 적절한 별칭 URL은 다음과 같습니다.

```
ALIAS_URL2=ldap:///$1V?*?sub?(mailAlternateAddress=@$D)
```

주 – +* 하위 주소 대체 기법은 항상 Direct LDAP 모드에서 catch-all 주소를 사용하지만 문자열 대체는 전체 로컬 부분이 아닌 하위 주소에서만 이루어집니다. 이것은 이러한 구성을 사용할 때 원래 주소의 전체 로컬 부분이 catch-all 주소에 하위 주소로 연결되도록 변경되었습니다.

예를 들어, 주소 형식이 foo+bar@domain.com인 경우 domain.com 도메인에 로컬 사용자 foo가 없고 domain.com의 catch-all 주소가 bletch+*@example.com이면 결과 주소는 bletch+foo+bar@example.com이 됩니다. 이전에는 bletch+bar@example.com이 사용되었습니다.

9.1.3 LDAP 결과 처리

LDAP 별칭 결과 처리는 순서가 정해진 여러 단계를 통해 수행됩니다. 이러한 단계는 다음 절에서 설명됩니다.

- 200 페이지 "9.1.3.14 지연된 처리 제어"
- 200 페이지 "9.1.3.15 그룹 확장 속성"

9.1.3.1 객체 클래스 검사

별칭 검색에 성공할 경우 사용자나 그룹에 대한 적절한 객체 클래스 집합이 포함되었는지 확인하기 위해 항목의 객체 클래스가 검사됩니다. 사용자 및 그룹에 대한 필수 객체 클래스의 가능한 집합은 일반적으로 활성화된 schemata에 따라 달라지며 이것은 `local.imta.schematag` 설정에 의해 결정됩니다.

표 9–1은 다양한 `schematag` 값의 결과인 사용자 및 그룹 객체 클래스를 보여 줍니다.

표 9–1 다양한 `schematag` 값의 결과인 객체 클래스

schematag	사용자 객체 클래스	그룹 객체 클래스
sims40	`inetMailRouting+inetmailuser`	`inetMailRouting+inetmailgroup`
nms41	`mailRecipient + nsMessagingServerUser`	`mailGroup`
ims50	`inetLocalMailRecipient+inetmailuser`	`inetLocalMailRecipient + inetmailgroup`

이 표의 정보는 스키마 태그 처리의 나머지 부분과 마찬가지로 하드 코딩됩니다. 그러나 `option.dat` 파일에는 또한 사용자와 그룹에 대해 각각 다른 객체 클래스 집합을 지정하도록 설정할 수 있는 두 개의 MTA 옵션인 `LDAP_USER_OBJECT_CLASSES`와 `LDAP_GROUP_OBJECT_CLASSES`가 존재합니다.

예를 들어, `ims50,nms41`의 스키마 태그 설정은 다음 옵션 설정과 동등합니다.

```
LDAP_USER_OBJECT_CLASSES=inetLocalMailRecipient+inetmailuser,
mailRecipient+nsMessagingServerUser

LDAP_GROUP_OBJECT_CLASSES=inetLocalMailRecipient+inetmailgroup, mailGroup
```

사용자나 그룹에 적합한 올바른 객체 클래스 집합이 없을 경우 LDAP 결과는 간단하게 무시됩니다. 또한 MTA는 사용자나 그룹을 처리하고 있는지 확인하여 이 정보를 저장합니다. 저장된 이 정보는 나중에 반복적으로 사용됩니다.

여기에 설명된 객체 클래스 설정이 또한 실제 LDAP 검색 필터를 생성하는 데 사용되며 이 필터는 사용자나 그룹에 대한 올바른 객체 클래스를 항목이 갖고 있는지 확인하는 데 사용될 수 있다는 것에 주의합니다. 이 필터는 `$K` 메타 문자를 통해 액세스할 수 있습니다. 또한 이 필터는 채널 프로그램에서 사용할 수 있도록 MTA의 구성에 내부적으로 저장되며 `imsimta cnbuild -option` 명령이 사용되면 MTA 옵션 파일 `option.dat`에 `LDAP_UG_FILTER` 옵션으로 기록됩니다. 이 옵션은 단지 이 파일에 기록만 되며 MTA는 옵션 파일에서 절대로 이 옵션을 읽지 않습니다.

9.1.3.2 항목 상태 검사

다음은 항목의 상태가 검사됩니다. 두 가지 상태 속성이 존재하는데 하나는 항목에 대한 일반 속성이며 다른 하나는 메일 서비스에 대한 특수한 속성입니다.

표 9–2는 유효한 schemata에 따라 검사되는 schematag 항목의 일반 및 메일별 사용자 또는 그룹 속성을 보여 줍니다.

표 9–2 검사할 속성

schematag	유형	일반적인 문제	메일별
sims40	사용자	inetsubscriberstatus	mailuserstatus
sims40	그룹	없음	inetmailgroupstatus
nms41	사용자	없음	mailuserstatus
nms41	그룹	없음	없음
Messaging Server 5.0	사용자	inetuserstatus	mailuserstatus
Messaging Server 5.0	그룹	없음	inetmailgroupstatus

필요한 경우 option.dat 파일의 LDAP_USER_STATUS 및 LDAP_GROUP_STATUS MTA 옵션을 각각 사용하여 사용자와 그룹에 대한 대체 일반 상태 속성을 선택할 수 있습니다. 메일별 사용자 및 그룹 상태 속성은 LDAP_USER_MAIL_STATUS 및 LDAP_GROUP_MAIL_STATUS MTA 옵션으로 제어합니다.

이와 관련된 또 다른 요소는 도메인 자체에 대한 상태(LDAP_DOMAIN_ATTR_STATUS 및 LDAP_DOMAIN_ATTR_MAIL_STATUS)입니다. 모두 합쳐서 네 개의 상태 속성이 존재하며 이러한 상태 속성은 다음 순서로 고려되어 결합됩니다.

1. 도메인 상태
2. 도메인 메일 상태
3. 사용자 또는 그룹 상태
4. 메일 사용자 또는 그룹 상태

“active” 이외의 상태를 지정하는 첫 번째 속성은 다른 모든 것보다 우선합니다. 가능한 기타 상태 값은 “inactive”, “deleted”, “removed”, “disabled”, “hold” 및 “overquota”입니다.“Hold”, “disabled” 및 “removed” 상태는 메일 도메인이나 메일 사용자, 메일 그룹에만 지정될 수 있습니다. “Overquota” 상태는 메일 도메인 또는 메일 사용자 상태로만 지정할 수 있습니다.

특정 상태 속성이 존재하지 않을 경우 모든 상태는 기본적으로 “active”입니다. 알 수 없는 상태 값은 “inactive”로 .

네 가지 상태가 결합되면 사용자나 그룹은 “active”, “inactive”, “deleted”, “removed”, “disabled”, “hold” 및 “overquota”라는 상태가 될 수 있습니다.active 상태는 별칭 처리가 계속되도록 합니다. inactive 또는 overquota 상태의 경우 4xx(임시) 오류와 함께 주소가

즉시 거부됩니다. deleted, removed 및 disabled 상태의 경우 5xx(영구) 오류와 함께 주소가 즉시 거부됩니다. hold 상태는 상태 처리와 관련해서는 active로 처리되지만 나중에 전달 옵션이 고려될 때 단일 "hold" 항목을 포함하는 옵션 목록이 존재하는 모든 옵션보다 우선하도록 내부 플래그가 설정됩니다.

9.1.3.3 UID 검사

다음 단계는 항목의 UID를 고려하는 것입니다. 다양한 목적에 사용되는 UID는 모든 사용자 항목의 일부여야 하며 그룹 항목에 포함될 수 있습니다. UID가 없는 사용자 항목은 무시되며 이 별칭 URL의 처리가 실패로 종료합니다. 호스트된 도메인의 항목에 대한 UID는 실제 UID, 구분자 문자 및 도메인으로 구성될 수 있습니다. MTA는 실제 UID만을 원하기 때문에 나머지 항목은 제거되며 이 제거 작업에는 `option.dat` 파일의 `LDAP_DOMAIN_ATTR_UID_SEPARATOR` MTA 옵션에서 얻은 도메인 구분자 문자가 사용됩니다.

`uid`가 아닌 속성이 UID를 저장하는 데 사용된 드문 경우에는 `LDAP_UID` MTA 옵션을 사용하여 다른 속성을 사용하도록 강제할 수 있습니다.

9.1.3.4 메일 캡처

다음은 하나 이상의 메일 캡처 주소를 지정하는 데 사용되는 LDAP 속성이 검사됩니다. 이 목적에 사용되는 속성은 `LDAP_CAPTURE` MTA 옵션을 사용하여 지정해야 합니다. 기본값은 없습니다. 이 속성 값은 주소로 간주되며 특수한 "캡처" 알림이 생성되어 현재 메일을 첨부 파일로 포함하는 이러한 주소로 보내집니다. 또한 캡처 주소는 이후에 주소가 봉투의 From: 주소로 표시되는 경우에 주소 역방향 캐시를 시드하는 데 지시합니다.

9.1.3.5 역방향 캐시 시드

다음은 사용자 항목에 추가된 주 주소와 모든 별칭이 고려됩니다. 이 정보는 주소 역방향 캐시를 시드하는 데 사용되며 현재의 주소 변환 프로세스에서 특정한 역할을 수행하지는 않습니다. 우선 주 주소, 개인 이름, 수신자 제한, 수신자 경계 및 소스 블록 제한 속성이 고려됩니다. 주 주소는 일반적으로 "`mail`" 속성에 저장되지만 `LDAP_PRIMARY_ADDRESS` MTA 옵션을 적절하게 설정하여 다른 속성을 지정할 수 있습니다. (물론, 주 주소는 그 자체에 대해 반전됩니다.)다른 모든 속성의 경우에는 기본 속성이 존재하지 않습니다. 이러한 속성을 사용하려면 `LDAP_PERSONAL_NAME`(507 페이지 "17.4 휴가 자동 회신 속성" 참조), `LDAP_RECIPIENTLIMIT`, `LDAP_RECIPIENTCUTOFF`(385 페이지 "12.9.7 메시지 수신자 제한" 참조) 및 `LDAP_SOURCEBLOCKLIMIT`(382 페이지 "12.9.2 절대 메시지 크기 제한 지정" 참조) MTA 옵션으로 해당 속성을 지정해야 합니다. 또한 해당하는 도메인 수준 수신자 제한, 수신자 경계 및 소스 블록 제한 속성도 이 시점에서 고려됩니다. 사용자 수준 설정은 모든 도메인 수준 설정보다 우선합니다.

다음은 모든 보조 주소가 고려되고 각 보조 주소에 대한 캐시 항목이 작성됩니다. 보조 주소에는 역방향으로 진행되는 주소와 그렇지 않은 주소의 두 종류가 있습니다. 메일 캡처 요청을 모든 경우에 검사해야 하기 때문에 주소 역방향 캐시를 적절하게 시드하려면 두 종류를 모두 고려해야 합니다.

역방향이 적용되는 보조 주소는 일반적으로 mailAlternateAddress 속성에 저장되며 LDAP_ALIAS_ADDRESSES MTA 옵션을 설정하여 다른 옵션을 지정할 수 있습니다. 역방향이 적용되지 않는 보조 주소는 일반적으로 mailEquivalentAddress 속성에 저장되며 LDAP_EQUIVALENCE_ADDRESSES MTA 옵션으로 다른 속성을 지정할 수 있습니다.

9.1.3.6 메일 호스트 및 라우팅 주소

이제 mailhost 및 mailRoutingAddress 속성을 고려할 차례입니다. 고려되는 실제 속성은 각각 LDAP_MAILHOST 및 LDAP_ROUTING_ADDRESS MTA 옵션을 사용하여 무시할 수 있습니다. 이러한 속성은 서로 합쳐져서 주소를 지금 실행해야 하는지 아니면 다른 시스템으로 전달해야 하는지 여부를 결정합니다.

첫 번째 단계는 이 항목에 대해 mailhost가 의미가 있는지 여부를 확인하는 것입니다. 항목이 mailhost별 항목인지 여부를 확인하기 위해 해당 항목에 대해 활성화된 전달 옵션의 예비 검사가 수행됩니다. 특정 항목이 아닐 경우 mailhost 검사가 생략됩니다. 이 검사가 수행되는 방법을 이해하려면 196 페이지 “9.1.3.8 전달 옵션 처리”에 설명된 내용을 특히 # 플래그를 중심으로 참조하십시오.

사용자 항목의 경우 mailhost 속성은 로컬 시스템을 식별할 경우에만 실행됩니다. mailhost 속성은 local.hostname configutil 매개 변수의 값과 local.imta.hostnamealiases configutil 매개 변수에 지정된 값 목록에 대해 비교됩니다. 일치하는 항목이 있을 경우 mailhost 속성은 로컬 호스트를 식별하는 것으로 간주됩니다.

일치에 성공할 경우 별칭이 로컬로 실행될 수 있으며 별칭 처리가 계속된다는 것을 의미합니다. 일치에 실패할 경우에는 mailhost로 전달해야만 메일이 작동한다는 것을 의미합니다. 다음 형식의 새 주소가 생성되며

@mailhost:*user* @*domain*

별칭 확장 작업의 결과가 됩니다.

누락된 mailhost 속성의 처리는 항목이 사용자인지 아니면 그룹인지 여부에 따라 달라집니다. 사용자인 경우 mailhost가 필수적이므로 mailhost 속성이 존재하지 않을 경우 다음 형식의 새 주소가

@*smarthost*: *user*@*domain*

LDAP_DOMAIN_ATTR_SMARTHOST MTA 옵션에 지정된 도메인에 대한 스마트 호스트를 사용하여 생성됩니다. 도메인에 대한 스마트 호스트가 존재하지 않을 경우 오류가 보고됩니다.

반면, 그룹의 경우에는 mailhost가 필요하지 않으므로 누락된 mailhost는 그룹을 어디에서나 확장할 수 있다는 것을 의미하는 것으로 해석됩니다. 따라서 별칭 처리가 계속 진행됩니다.

mailRoutingAddress 속성은 유용한 최종 기능 하나를 추가합니다. 이 속성이 존재하면 별칭 처리는 결과적으로 mailRoutingAddress와 함께 종료합니다. 5.2 버전에서 mailHost 검사가 처음 수행된 후 라우팅 주소를 적용하려면 검사를 통과해야 합니다. 현재 버전 Messaging Server와 동일한 동작을 얻으려면 mailRoutingAddress 속성 형식이 다음과 같습니다. mailRoutingAddess: @mailhost:user@domain

9.1.3.7 기타 속성 지원

다음은 mailMsgMaxBlocks 속성이 고려됩니다. 우선 이 속성은 LDAP_DOMAIN_ATTR_BLOCKLIMIT MTA 옵션에서 반환된 도메인 블록 제한에 따라 최소화됩니다. 현재 메일의 크기가 이 제한을 초과하는 것으로 확인될 경우 별칭 처리는 크기 초과 오류와 함께 종료합니다. 크기가 알려지지 않거나 제한을 초과하지 않을 경우에는 제한이 저장되며 나중에 메일 자체가 검사될 때 해당 제한이 다시 검사됩니다. mailMsgMaxBlocks 사용은 LDAP_BLOCKLIMIT MTA 옵션을 사용하여 무시할 수 있습니다.

다음은 여러 속성이 평가 및 저장됩니다. 결국 이러한 속성은 ims_master 채널 프로그램에 사용되도록 대기열 파일 항목에 기록되며 이 채널 프로그램은 이러한 속성을 사용하여 저장소의 사용자 정보 캐시를 업데이트합니다. 개별 사용자에 대한 속성이 발견되지 않을 경우 도메인 수준 속성을 사용하여 기본값을 설정할 수 있습니다.

LDAP 항목이 사용자가 아니라 그룹에 대한 것이거나 LDAP 항목을 LDAP 디렉토리가 아니라 별칭 캐시에서 가져온 경우 이 단계를 건너뛰게 됩니다. 후자의 조건에 담겨 있는 논리는 이 정보를 자주 업데이트하는 것이 불필요하며 별칭 캐시를 사용하면 업데이트를 수행해야 할 경우에 대한 적절한 기준이 제공된다는 점입니다. 검색된 속성의 이름은 다양한 MTA 옵션에 의해 설정됩니다.

표 9–3은 검색된 디스크 할당량 및 메시지 할당량 속성을 설정하는 MTA 옵션을 보여 줍니다.

표 9–3 검색된 디스크 할당량 및 메시지 할당량 속성을 설정하는 MTA 옵션

MTA 옵션	속성
LDAP_DISK_QUOTA	mailQuota
LDAP_MESSAGE_QUOTA	mailMsgQuota

다음은 나중에 메타 문자 대체와 함께 사용할 수 있도록 여러 속성이 저장됩니다.

표 9–4는 MTA 옵션, 기본 속성 및 메타 문자를 보여 줍니다.

표 9–4 MTA 옵션, 기본 속성 및 메타 문자

MTA 옵션	기본 속성	메타 문자
LDAP_PROGRAM_INFO	mailProgramDeliveryInfo	$P

표 9–4 MTA 옵션, 기본 속성 및 메타 문자 (계속)

MTA 옵션	기본 속성	메타 문자
LDAP_DELIVERY_FILE	mailDeliveryFileURL	$F
LDAP_SPARE_1	기본값 없음	$1E $1G $E
LDAP_SPARE_2	기본값 없음	$2E $2G $G
LDAP_SPARE_3	기본값 없음	$3E $3G
LDAP_SPARE_4	기본값 없음	$4E $4G
LDAP_SPARE_5	기본값 없음	$5E $5G

추가 속성을 위한 예비 슬롯이 포함되는데 이러한 슬롯을 사용하여 사용자 정의된 주소 확장 기능을 작성할 수 있습니다.

다음은 mailconversiontag 속성과 연관된 모든 값이 현재의 변환 태그 집합에 추가됩니다. 이 속성의 이름은 LDAP_CONVERSION_TAG MTA 옵션을 사용하여 변경할 수 있습니다. 또한 도메인의 mailDomainConversionTag 속성과 연관된 모든 값도 추가됩니다.

9.1.3.8 전달 옵션 처리

다음은 mailDeliveryOption 속성이 검사됩니다. 이 속성의 이름은 LDAP_DELIVERY_OPTION MTA 옵션을 사용하여 변경할 수 있습니다. 이 속성은 여러 값을 가지며 별칭 변환 프로세스에 의해 생성되는 주소가 이 속성 값에 의해 결정됩니다. 또한 사용자와 그룹에 대해 허용되는 값이 다릅니다. 일반적으로 허용되는 값은 program, forward 및 hold입니다.사용자 전용 값은 mailbox, native, unix 및 autoreply이며 그룹 전용 값은 members, members_offline 및 file입니다.

mailDeliveryOption 속성이 적절한 주소로 변환되는 작업은 DELIVERY_OPTIONS MTA 옵션에 의해 제어됩니다. 이 옵션은 허용되는 각 mailDeliveryOption 값이 생성하는 주소뿐만 아니라 허용되는 mailDeliveryOption 값과 각 값이 사용자, 그룹 또는 둘 다에 적용될 수 있는지 여부를 지정합니다.

이 옵션 값은 deliveryoption=template 쌍의 쉼표로 구분된 목록으로 구성되며 각 쌍은 하나 이상의 선택적 단일 문자 접두어를 가집니다.

DELIVERY_OPTIONS 옵션의 기본값은 다음과 같습니다.

```
DELIVERY_OPTIONS=*mailbox=$M%$\\$2I$_+$2S@ims-ms-daemon, \
     &members=*,                                          \
     *native=$M@native-daemon,                            \
     /hold=@hold-daemon:$A,                               \
     *unix=$M@native-daemon,                              \
     &file=+$F@native-daemon,                             \
     &@members_offline=*,                                 \
```

```
program=$M%$P@pipe-daemon,                              \
#forward=**,                                            \
*^!autoreply=$M+$D@bitbucket
```

각 전달 옵션은 가능한 mailDeliveryOption 속성 값에 해당하며 해당 템플리트는 URL 처리에 사용되는 것과 동일한 메타 문자 대체 방법을 사용하여 결과 주소를 지정합니다.

표 9–5는 DELIVERY_OPTIONS 옵션에 사용할 수 있는 단일 문자 접두어를 보여 줍니다.

표 9–5 DELIVERY_OPTIONS MTA 옵션에 사용할 수 있는 단일 문자 접두어

접두어 문자	설명
@	메일이 프로세스 채널에 리디렉션되어야 한다는 사실을 보여주도록 플래그를 설정하십시오. 현재 사용자/그룹의 처리는 금지되어 있습니다. 플래그는 재처리 채널에서 보내지는 메일에 대해서는 무시됩니다.
*	전달 옵션이 사용자에게 적용됩니다.
&	전달 옵션이 그룹에 적용됩니다.
$	이 사용자 또는 그룹의 확장이 지연된다는 것을 나타내는 플래그를 설정합니다.
^	휴가 시작 및 종료 시간을 검사하여 이 전달 옵션이 실제로 유효한지 확인해야 한다는 것을 나타내는 플래그를 설정합니다.
#	이 전달 옵션의 확장이 항목의 지정된 메일 호스트에서 발생할 필요가 없다는 것을 나타내는 플래그를 설정합니다. 즉, 다음 항목은 메일 호스트에 독립적입니다. 이렇게 하면 MTA는 주어진 사용자 또는 그룹의 모든 전달 옵션이 메일 호스트에 독립적인지 확인할 수 있습니다. 이 조건이 충족되면 MTA는 메일을 메일 호스트에 전달하지 않고도 즉시 항목에 대한 작업을 수행할 수 있습니다.
/	이 전달 옵션에 의해 생성된 모든 주소가 보관된다는 것을 나타내는 플래그를 설정합니다. 이러한 수신자 주소를 포함하는 메시지 파일은 .HELD 확장자를 가집니다.
!	MTA가 자동 회신 작업을 내부적으로 처리해야 한다는 것을 나타내는 플래그를 설정합니다. 이 접두어는 자동 회신 전달 옵션에서만 사용되어야 합니다. 옵션 값은 메일을 bitbucket 채널로 전송해야 합니다.

* 및 &가 모두 존재하지 않을 경우 전달 옵션은 사용자 및 그룹 모두에 적용됩니다.

9.1.3.9 전달 옵션에 사용할 추가 메타 문자

MTA의 새로운 URL 템플리트 기능 사용을 지원하기 위해 여러 추가 메타 문자가 다음과 같이 추가되었습니다.

표 9–6은 전달 옵션에서 사용하기 위한 추가 메타 문자와 그 설명을 보여 줍니다.

표 9–6 전달 옵션에 사용할 추가 메타 문자

메타 문자	설명
$\	후속 텍스트를 소문자로 바꿉니다.

표 9–6 전달 옵션에 사용할 추가 메타 문자 *(계속)*

메타문자	설명
$^	후속 텍스트를 대문자로 바꿉니다.
$_	후속 텍스트에 대해 대소문자 변환을 수행하지 않습니다.
$*n*A	주소의 *n*번째 문자를 삽입합니다. 첫 번째 문자는 문자 0입니다. *n*이 생략될 경우 전체 주소가 대체됩니다. 이 메타 문자는 자동 회신 디렉토리 경로를 생성하는 데 사용됩니다.
$D	주소의 도메인 부분을 삽입합니다.
$*n*E	*n*번째 예비 속성의 값을 삽입합니다. *n*이 생략될 경우 첫 번째 속성이 사용됩니다.
$F	전달 파일의 이름(mailDeliveryFileURL 속성)을 삽입합니다.
$*n*G	*n*번째 예비 속성의 값을 삽입합니다. *n*이 생략될 경우 두 번째 속성이 사용됩니다.
$*n*H	원래 주소에서 0부터 시작하는 도메인의 *n*번째 구성 요소를 삽입합니다. *n*이 생략될 경우 기본값은 0입니다.
$*n*I	별칭과 연관된 호스트된 도메인을 삽입합니다. 이 메타 문자는 표 9–7에 의미가 설명되어 있는 정수 매개변수 *n*을 허용합니다.
$*n*J	0부터 시작하는 호스트 도메인의 *n*번째 부분을 삽입합니다. *n*의 기본값은 0입니다.
$*n*O	현재 주소와 연관된 소스 루트를 삽입합니다. 이 메타 문자는 표 9–7에 의미가 설명되어 있는 정수 매개변수 n을 허용합니다.
$K	사용자 또는 그룹에 대한 객체 클래스와 일치하는 LDAP 필터를 삽입합니다. LDAP_UG_FILTER 출력 전용 MTA 옵션의 설명을 참조하십시오.
$L	주소의 로컬 부분을 삽입합니다.
$*n*M	UID의 *n*번째 문자를 삽입합니다. 첫 번째 문자는 문자 0입니다. n이 생략될 경우 전체 UID가 대체됩니다.
$P	프로그램 이름(mailProgramDeliveryInfo 속성)을 삽입합니다.
$*n*S	현재 주소와 연관된 하위 주소를 삽입합니다. 이 메타 문자는 표 9–7에 의미가 설명되어 있는 정수 매개변수 n을 허용합니다.
$*n*U	현재 주소의 메일함 부분에서 따옴표가 없는 형식의 n번째 문자를 삽입합니다. 첫 번째 문자는 문자 0입니다. n이 생략될 경우 따옴표가 없는 전체 메일함이 대체됩니다.
$*n*X	메일 호스트의 *n*번째 구성 요소를 삽입합니다. *n*이 생략될 경우 전체 메일 호스트가 삽입됩니다.

표 9–7은 정수 매개 변수가 $nI 및 $nS 메타 문자의 동작을 수정하는 방법을 보여 줍니다.

표 9–7 $nI 및 $nS 메타 문자의 동작 수정을 제어하는 정수

정수	동작 설명
0	사용할 수 있는 값이 없을 경우 실패합니다(기본값).

표 9–7 $nI 및 $nS 메타 문자의 동작 수정을 제어하는 정수 *(계속)*

정수	동작 설명
1	사용할 수 있는 값이 있을 경우 해당 값을 삽입하고없을 경우에는 아무 것도 삽입하지 않습니다.
2	사용할 수 있는 값이 있을 경우 해당 값을 삽입하고없을 경우에는 아무 것도 삽입하지 않고 앞의 문자를 삭제합니다(이 특수한 동작은 `ims-ms` 채널에 필요함).
3	사용할 수 있는 값이 있을 경우 해당 값을 삽입하고없을 경우에는 아무 것도 삽입하지 않고 다음 문자를 무시합니다.

표 9–8은 메타 문자 외에 두 개의 특수한 템플리트 문자열을 보여 줍니다.

표 9–8 특수한 템플리트 문자열

특수한 템플리트 문자열	설명
*	그룹 확장을 수행합니다. 사용자 항목의 경우에는 이 값이 유효하지 않습니다.
**	`LDAP_FORWARDING_ADDRESS` MTA 옵션에 의해 명명된 속성을 확장합니다. 기본값은 `mailForwardingAddress`입니다.

그룹 확장의 경우 예를 들어, 사용자의 `mailDeliveryOption` 값이 `mailbox`로 설정되면 스트라이프된 UID, 백분율 기호 다음의 호스트된 도메인(적용 가능한 경우), 더하기 기호 다음의 하위 주소(지정된 경우), 마지막으로 `@ims-ms-daemon`으로 구성된 새 주소가 생성됩니다.

9.1.3.10 전달 옵션 기본값

이 시점에서 활성 전달 옵션의 목록이 비어 있을 경우 목록의 첫 번째 옵션(일반적으로 메일함)이 사용자에 대해 활성화되고 목록의 두 번째 옵션(일반적으로 구성원)이 그룹에 대해 활성화됩니다.

9.1.3.11 시작 및 종료 날짜 검사

전달 옵션 목록을 읽은 후에 시작 및 종료 날짜가 검사됩니다. 두 가지 속성이 존재하며 이러한 속성의 이름은 각각 `LDAP_START_DATE`(기본값: `vacationStartDate`) 및 `LDAP_END_DATE`(기본값: `vacationEndDate`) MTA 옵션에 의해 제어됩니다. 하나 이상의 활성 전달 옵션이 `^` 접두어 문자를 지정한 경우 이러한 옵션의 값이 현재 날짜에 대해 검사됩니다. 현재 날짜가 이러한 옵션에 지정된 범위를 벗어날 경우 `^` 접두어를 가진 전달 옵션이 활성 집합에서 제거됩니다. 자세한 내용은 507 페이지 “17.4 휴가 자동 회신 속성”을 참조하십시오.

9.1.3.12 Optin 및 Presence 속성

LDAP_OPTIN1 ~ LDAP_OPTIN8 MTA 옵션은 대상 주소를 기반으로 사용자 단위 스팸 필터 수신 선택 기능 값에 대한 LDAP 속성을 지정합니다. 옵션이 지정되고 속성이 존재할 경우 현재의 스팸 필터 수신 선택 기능 목록에 해당 속성이 추가됩니다. 또한 LDAP_DOMAIN_ATTR_OPTIN MTA 옵션에 의해 설정된 도메인 수준 속성이 설정하는 모든 값도 이 목록에 추가됩니다. LDAP_SOURCE_OPTIN1 ~ LDAP_SOURCE_OPTIN8은 그와 동등한 발송자 주소 기반의 사용자 단위 스팸 필터 optin을 제공합니다.

LDAP_PRESENCE MTA 옵션을 사용하면 사용자에 대한 존재 여부 정보를 반환하기 위해 확인할 수 있는 URL을 지정할 수 있습니다. 이 옵션이 지정되고 속성이 존재할 경우 시브(Sieve) 존재 여부 테스트와 함께 사용할 수 있도록 속성 값이 저장됩니다. 사용자 항목에 대한 값이 존재하지 않을 경우 LDAP_DOMAIN_ATTR_PRESENCE MTA 옵션에 의해 설정된 도메인 수준 속성이 이 URL의 소스로 사용됩니다.

9.1.3.13 시브(Sieve) 필터 처리

다음은 이 항목에 적용되는 시브(Sieve) 필터에 대해 mailSieveRuleSource 속성이 검사됩니다. 이 속성은 존재할 경우 이 시점에서 구문 분석 및 저장됩니다. 이 속성 값에 대한 두 가지 가능한 형식으로는 완전한 시브(Sieve) 스크립트를 포함하는 단일 값 형식과 각 값이 시브(Sieve) 스크립트를 포함하는 다중 값 형식이 있습니다. 후자의 형식은 웹 필터 생성 인터페이스에 의해 생성됩니다. 값의 순서를 지정하고 적절하게 값을 결합하기 위해 특수한 코드가 사용됩니다.

mailSieveRuleSource 속성 사용을 특별히 LDAP_FILTER MTA 옵션을 사용하여 무시할 수 있습니다.

9.1.3.14 지연된 처리 제어

다음은 mailDeferProcessing 속성이 검사됩니다. 이 속성은 LDAP_REPROCESS MTA 옵션을 사용하여 변경할 수 있습니다. 이 속성이 존재하고 no로 설정된 경우 처리는 정상적으로 처리됩니다. 그러나 이 속성이 yes로 설정되고 현재 소스 채널이 reprocess 채널이 아닐 경우 이 항목의 확장이 중지되며 원래 *user@domain* 주소가 단순히 reprocess 채널의 대기열에 포함됩니다. 이 속성이 존재하지 않을 경우 전달 옵션 처리와 연관된 지연된 처리 문자 접두어의 설정이 검사됩니다. 이에 대한 예는 196 페이지 "9.1.3.8 전달 옵션 처리" 절을 참조하십시오. 사용자에 대한 기본값은 no입니다. 그룹에 대한 기본값은 DEFER_GROUP_PROCESSING MTA 옵션에 의해 제어되며 이 옵션의 기본값은 1(yes)입니다. 사용자 항목의 경우 이 시점에서 별칭 처리가 완료됩니다.

9.1.3.15 그룹 확장 속성

여러 추가 속성이 그룹 확장과 연관되며 이 시점에서 처리되어야 합니다. 이러한 속성의 이름은 모두 다양한 MTA 옵션을 통해 구성할 수 있습니다.

표 9–9에는 기본 속성 이름, 속성 이름을 설정하기 위한 MTA 옵션 및 MTA가 속성을 처리하는 방법이 나열되어 있습니다. 이 표에 있는 요소의 순서는 다양한 그룹 속성이 처리되는 순서를 보여 줍니다. 정확한 작업을 위해서는 이 순서가 필수적입니다.

표 9–9 설정할 그룹 확장 기본 속성 및 MTA 옵션

기본 속성	(속성 이름을 설정하기 위한 MTA 옵션) 속성이 처리되는 방법
mgrpMsgRejectAction	(LDAP_REJECT_ACTION) 후속 액세스 검사가 실패할 경우 발생하는 작업을 제어하는 단일 값 속성입니다. 값은TOMODERATOR 하나만 정의되며 이 값은 설정된 경우 모든 액세스 실패를 mgrpModerator 속성에 의해 지정된 중재자로 리디렉션하도록 MTA에 지시합니다. 이 속성에 값(기본값 및 다른 모든 값)이 지정되면 오류가 보고되며 메일이 거부됩니다.
mailRejectText	(LDAP_REJECT_TEXT) 이 속성의 첫 번째 값에 저장된 텍스트의 첫 번째 행이 저장됩니다. 다음 인증 속성 중 하나로 인해 메일이 거부될 경우 이 텍스트가 반환됩니다. 이것은 텍스트가 SMTP 응답에 나타날 수 있으므로 값을 US-ASCII로 제한하여 현재 메시징 표준을 따르도록 해야 한다는 것을 의미합니다.
mgrpBroadcasterPolicy	(LDAP_AUTH_POLICY) 그룹에 보내는 데 필요한 인증 수준을 지정합니다. 가능한 토큰은 SMTP_AUTH_REQUIRED 또는 AUTH_REQ(두 가지 모두 SMTP AUTH 명령을 사용하여 그룹에 보내는 순서대로 보내는 사람을 식별해야 함을 의미) SMTP_AUTH_USED 및 AUTH_USED(SMTP_AUTH_REQUIRED 및 AUTH_REQ와 비슷하지만 게시자를 인증할 필요가 없음), PASSWORD_REQUIRED , PASSWD_REQUIRED 또는 PASSWD_REQ(세 가지 모두 mgrpAuthPassword 속성에 지정된 목록의 비밀번호가 메시지의 승인됨: 헤더 필드에 표시되어야 함을 의미), OR(이 목록에 대해 OR_CLAUSES MTA 옵션 설정을 1로 변경), AND(이 목록에 대해 OR_CLAUSES 옵션 설정을 0으로 변경), NO_REQUIREMENTS(작업하지 않음) 등입니다. 여러 값이 허용되고 각 값은 쉼표로 구분된 토큰 목록으로 구성될 수 있습니다. SMTP AUTH가 호출된 경우에는 또한 MAIL FROM 주소가 아니라 SASL 계층에 의해 제공된 전자 메일 주소에 대해 이후의 모든 권한 부여 검사가 수행된다는 것을 의미합니다.
mgrpAllowedDomain	(LDAP_AUTH_DOMAIN) 이 그룹에 메일을 제출하는 것이 허용된 도메인입니다. 0(기본값)으로 설정된 OR_CLAUSES 옵션과의 일치 실패는 액세스 검사가 실패했고 모든 후속 테스트를 수행하지 않는다는 의미입니다. 1로 설정된 OR_CLAUSES MTA 옵션과의 일치 실패는 "failure pending" 플래그를 설정합니다. 즉, 액세스 검사를 계속하려면 다른 액세스 검사가 성공해야 합니다. 전송자가 이미 LDAP_AUTH_URL과 일치되어 있으면 이러한 검사를 수행하지 않습니다. 값이 여러 개일 수 있고 글롭 와일드카드가 허용됩니다.
mgrpDisallowedDomain	(LDAP_CANT_DOMAIN) 이 그룹에 메일을 제출하는 것이 허용되지 않는 도메인입니다. 일치는 액세스 검사가 실패했고 모든 후속 검사가 수행되지 않는다는 의미입니다. 전송자가 이미 LDAP_AUTH_URL과 일치되어 있으면 이러한 검사를 수행하지 않습니다. 값이 여러 개일 수 있고 글롭 와일드카드가 허용됩니다.
mgrpAllowedBroadcaster	(LDAP_AUTH_URL) 이 그룹에 메일을 전송하는 것이 허용된 메일 주소를 식별하는 URL입니다. 값이 여러 개일 수 있습니다. 각 URL은 주소 목록으로 확장되며 현재 봉투의 From: 주소에 대해 각 주소가 지시합니다. 0(기본값)으로 설정된 OR_CLAUSES MTA 옵션과의 일치 실패는 액세스 검사가 실패했고 모든 후속 테스트를 수행하지 않는다는 의미입니다. 1로 설정된 OR_CLAUSES MTA 옵션과의 일치 실패는 "failure pending" 플래그를 설정합니다. 즉, 액세스 검사가 성공하기 위해서는 다른 액세스 검사가 성공해야 합니다. 일치하는 경우에도 후속 도메인 액세스 검사는 비활성화됩니다. 수행되는 확장은 모든 액세스 제어 검사를 비활성화하는 SMTP EXPN과 비슷합니다. mgrpallowedbroadcaster LDAP 속성 컨텍스트의 목록 확장은 전자 메일 주소를 저장하는 데 사용된 모든 속성(일반적으로 mail, mailAlternateAddress 및 mailEquivalentAddress)을 포함합니다. 이전에는 mail 속성만 반환되었으므로 대체 주소를 사용하여 해당 구성원으로 제한된 목록에 보낼 수 없었습니다.

표 9–9 설정할 그룹 확장 기본 속성 및 MTA 옵션 *(계속)*

기본 속성	(속성 이름을 설정하기 위한 MTA 옵션) 속성이 처리되는 방법
mgrpDisallowedBroadcaster	(LDAP_CANT_URL) 메일을 이 그룹에 전송하는 것이 허용되지 않는 메일 주소를 식별하는 URL입니다. 값이 여러 개일 수 있습니다. 각 URL은 주소 목록으로 확장되며 현재 봉투의 From: 주소에 대해 각 주소가 지시합니다. 일치는 액세스 검사가 실패했고 모든 후속 검사가 수행되지 않는다는 의미입니다. 수행되는 확장은 모든 액세스 제어 검사를 비활성화하는 SMTP EXPN과 비슷합니다.
mgrpMsgMaxSize	(LDAP_ATTR_MAXIMUM_MESSAGE_SIZE) 그룹에 전송할 수 있는 최대 메일 크기(바이트)입니다. 이 속성은 폐기되었지만 이전 버전과의 호환성을 위해 계속 지원됩니다. 대신, 새 mailMsgMaxBlocks 속성을 사용해야 합니다.
mgrpAuthPassword	(LDAP_AUTH_PASSWORD) 목록에 게시하는 데 필요한 비밀번호를 지정합니다. mgrpAuthPassword속성이 존재하면 재처리 통과가 강제됩니다. 메일이 재처리 채널의 대기열에 포함되면 헤더에서 가져온 비밀번호가 봉투에 포함됩니다. 그런 다음 재처리 도중에 해당 비밀번호는 봉투에서 가져와 이 속성에 대해 검사됩니다. 또한 실제로 사용되는 비밀번호만 헤더 필드에서 제거됩니다. 이 속성에서 OR_CLAUSES MTA 옵션은 다른 액세스 검사 속성의 경우와 같은 방식으로 작동됩니다.
mgrpModerator	(LDAP_MODERATOR_URL) 일련의 주소로 확장될 이 속성에 의해 제공된 URL 목록입니다. 이 주소 목록의 해석은 LDAP_REJECT_ACTION MTA 옵션의 설정에 따라 달라집니다. LDAP_REJECT_ACTION이 TOMODERATOR로 설정된 경우 이 속성은 액세스 검사 중 하나가 실패할 경우에 메일을 전송할 중재자 주소를 지정합니다. LDAP_REJECT_ACTION이 누락되거나 다른 값을 가진 경우 이 주소 목록은 봉투의 From: 주소와 비교됩니다. 일치하는 항목이 있을 경우 처리가 계속됩니다. 일치하는 항목이 없을 경우 이 속성에 지정된 모든 주소로 메일이 다시 전송됩니다. 이 속성을 확장하는 작업은 이 속성 값을 그룹에 대한 URL 목록으로 만드는 방법으로 구현됩니다. 그룹과 연관된 RFC822 주소 또는 DN의 모든 목록이 지워지며 그룹에 대한 전달 옵션은 members로 설정됩니다. 마지막으로 이 표에 나열된 후속 그룹 속성이 무시됩니다.
mgrpDeliverTo	(LDAP_GROUP_URL1) 확장될 경우 메일 목록 구성원 주소의 목록을 제공하는 URL 목록입니다.
memberURL	(LDAP_GROUP_URL2) 확장될 경우 메일 목록 구성원 주소의 또 다른 목록을 제공하는 URL의 다른 목록입니다.
uniqueMember	(LDAP_GROUP_DN) 그룹 구성원의 DN 목록입니다. DN은 전체 하위 트리를 지정할 수 있습니다. 고유한 구성원 DN은 LDAP URL에 이러한 DN을 내장하는 방법으로 확장됩니다. 사용할 정확한 URL은 GROUP_DN_TEMPLATE MTA 옵션에 의해 지정됩니다. 이 옵션의 기본값은 다음과 같습니다. ldap:///$A??sub?mail=* $A는 uniqueMember DN이 삽입되는 지점을 지정합니다.
mgrpRFC822MailMember	(LDAP_GROUP_RFC822) 이 목록에 있는 구성원의 메일 주소입니다.
rfc822MailMember	(LDAP_GROUP_RFC822) rfc822MailMember는 이전 버전과의 호환성을 위해 지원됩니다. rfc822MailMember와 mgrpRFC822MailMember 중에서 하나를 주어진 모든 그룹에 사용할 수 있습니다.
mgrpErrorsTo	(LDAP_ERRORS_TO) 봉투의 메일 발송자(MAIL FROM) 주소를 속성이 지정하는 주소로 설정합니다.

표 9–9 설정할 그룹 확장 기본 속성 및 MTA 옵션 (계속)

기본 속성	(속성 이름을 설정하기 위한 MTA 옵션) 속성이 처리되는 방법
mgrpAddHeader	(LDAP_ADD_HEADER) 속성에 지정된 헤더를 헤더 자르기 ADD 옵션으로 변환합니다.
mgrpRemoveHeader	(LDAP_REMOVE_HEADER) 지정된 헤더를 헤더 자르기 MAXLINES=-1 옵션으로 변환합니다.
mgrpMsgPrefixText	(LDAP_PREFIX_TEXT) 지정된 텍스트를 메일 텍스트(있을 경우)의 시작 부분에 추가합니다.
mgrpMsgSuffixText	(LDAP_SUFFIX_TEXT) 지정된 텍스트를 메일 텍스트(있을 경우)의 끝 부분에 추가합니다.
No Default	(LDAP_ADD_TAG) 지정된 텍스트의 제목을 검사하며 없을 경우 제목 필드의 첫 부분에 텍스트가 추가됩니다.

SMTP EXPN 명령: mgmanMemberVisibility 또는 확장의 일부인 특수한 그룹 확장에서는 최종 속성 하나가 검사됩니다. LDAP_EXPANDABLE MTA 옵션을 사용하여 검사할 다른 속성을 선택할 수 있습니다. 가능한 값에는anyone(누구나 그룹을 확장할 수 있다는 것을 의미), all 또는 true(확장이 허용되려면 사용자가 SASL로 성공적으로 인증되어야 한다는 것을 의미), 그리고 none(확장이 허용되지 않는다는 것을 의미)이 있습니다. 인식되지 않는 값은 none으로 해석됩니다. 이 속성이 존재하지 않을 경우 EXPANDABLE_DEFAULT MTA 옵션은 확장이 허용되는지 여부를 제어합니다.

별칭 항목은 도메인 항목과 비슷한 방식으로 캐시됩니다. 별칭 캐시를 제어하는 MTA 옵션은 ALIAS_ENTRY_CACHE_SIZE(기본값은 1000개 항목) 및 ALIAS_ENTRY_CACHE_TIMEOUT(기본값은 600초)입니다. 주어진 별칭에 대한 전체 LDAP 반환 값은 캐시에 보관됩니다.

별칭 항목의 네거티브 캐싱은 ALIAS_ENTRY_CACHE_NEGATIVE MTA 옵션에 의해 제어됩니다. 0이 아닌 값은 별칭 일치 실패의 캐싱을 활성화하고 값 0은 이 캐싱을 비활성화합니다. 별칭 항목의 네거티브 캐싱은 기본적으로 비활성화되는데 이것은 잘못된 주소가 실제로 반복적으로 지정될 가능성이 별로 없기 때문입니다. 또한 네거티브 캐싱은 디렉토리에 추가된 새 사용자를 제때에 인식하는 데 방해가 될 수 있습니다. 그러나 부속 도메인이 많이 사용될 경우 사이트는 별칭의 네거티브 캐싱을 다시 활성화하는 것을 고려해야 합니다. ALIAS_URL0에 지정된 URL에 의해 수행되는 검색은 성공할 가능성이 적습니다.

9.1.4 그룹 구성원 속성 구문 변경

LDAP 확장 결과를 매핑과 함께 사후 처리하도록 지원하는 기능이 추가되었습니다. 새로운 LDAP_URL_RESULT_MAPPING MTA 옵션을 사용하여 그룹 속성의 이름을 지정할 수 있습니다. 그러면 이 그룹 속성은 매핑의 이름을 지정합니다. 이 매핑은 mgrpDeliverTo 또는 memberURL 속성을 확장하여 반환한 모든 결과에 적용됩니다. 매핑 검사의 형식은 다음과 같습니다.

LDAP-URL|*LDAP-result*

$Y가 설정된 상태로 매핑이 반환될 경우, 매핑 결과 문자열은 별칭 처리를 위해 LDAP 결과를 대체합니다. $N이 설정된 상태에서 매핑이 반환될 경우, 결과를 건너뜁니다.

이 기법은 알맞은 전자 메일 주소를 포함하지 않은 그룹을 속성을 기반으로 정의할 때 사용할 수 있습니다. 예를 들어, 어떤 회사가 모든 사용자 항목에 호출기 번호를 포함시켰다고 가정합니다. 메시지에 특정 도메인을 접미어로 추가하면 전자 메일을 통해 이 번호로 메시지를 보낼 수 있습니다. 그러면 다음과 같이 그룹을 정의할 수 있습니다.

1. 디렉토리에서 새 `mgrpURLResultMapping` 속성을 정의하고 `LDAP_URL_RESULT_MAPPING` MTA 옵션을 이 속성의 이름으로 설정합니다.
2. 다음 속성으로 page-all 그룹을 정의합니다.

```
mgrpDeliverto: ldap:///o=usergroup?pagerTelephoneNumber?sub
mgrpURLResultMapping: PAGER-NUMBER-TO-ADDRESS
```

3. 매핑을 정의합니다.

```
PAGER-NUMBER-TO-ADDRESS
  *|*    "$1"@pagerdomain.com$Y
```

이 기법을 260 페이지 “10.12.1 메일링 목록으로 주소 지정된 메시지의 경우 LDAP 디렉토리에 대한 권한 부여 검사 최적화”에서 설명한 `PROCESS_SUBSTITUTION` 기법과 조합하면 더 흥미로운 결과를 얻을 수 있습니다. 예를 들어,

pager+user@*domain.com*

위와 같은 형식의 주소로 보내면 `user`라는 사용자에게 호출을 보내는 메타그룹을 쉽게 만들 수 있습니다.

9.2 주소 역방향

Direct LDAP를 사용한 주소 역방향은 `USE_REVERSE_DATABASE` 값 4로 시작되며 이 값은 역방향 데이터베이스를 사용하지 않도록 설정합니다. sleepycat 데이터베이스는 더 이상 사용되지 않으므로 `IMTA_TABLE:reverse.txt` 파일을 읽도록 `USE_TEXT_DATABASES`를 설정해야 합니다. 그런 다음 주소 역방향은 앞에서 설명한 라우팅 기능을 토대로 작성됩니다. 특히 이전 버전에서는 다음 형식의 역방향 URL 지정으로 시작되었습니다.

```
REVERSE_URL=ldap:///$V?mail?sub?$Q
```

`$V` 메타 문자는 별칭 URL과 관련하여 이미 설명되었습니다. 그러나 `$Q` 메타 문자는 별칭 URL에 사용되는 `$R` 메타 문자와 기능이 비슷하지만 특별히 주소 역방향에 사용하도록 되어 있습니다. `$R`과 달리 이 메타 문자는 주소 역방향의 후보인 주소가 포함된 속성을 검색하는 필터를 생성합니다. 검색할 속성 목록은 MTA 옵션 `LDAP_MAIL_REVERSES`에서 가져옵니다. 이 옵션이 설정되지 않은 경우 `local.imta.schematag configutil` 매개변수가 검사되고 그 값에 따라 적절한 기본 속성 집합이 선택됩니다.

주 – REVERSE_URL은 어떠한 이유로도 변경하지 않는 것이 좋습니다.

표 9–10에서는 local.imta.schematag 값과 선택된 기본 속성을 보여 줍니다.

표 9–10 local.imta.schematag 값과 속성

스키마 태그 값	속성
sims40	mail,rfc822mailalias
nms41	mail,mailAlternateAddress
ims50	mail,mailAlternateAddress

그러나 $Q는 사용하기에 더 이상 적합하지 않습니다. 메일 캡처와 다른 기능이 제대로 작동하도록 하기 위해 일치하는 항목이 발생했다는 사실 외에도 일치한 속성에 주의하도록 주소 역방향이 향상되었습니다. 이것은 $Q 대신에 $R을 사용하여 필터를 지정해야 한다는 것을 의미합니다. 또한 주소 역방향이 필요할 수 있는 속성 목록을 반환하는 $N 메타 문자가 추가되었습니다.

$N 값을 정확하게 제어할 수 없습니다. MTA가 주소 역방향에 대한 관련 속성의 고유한 하드 코드된(변경 예정) 목록에서 자동으로 구성됩니다. 다양한 LDAP_* 전역 MTA 옵션을 사용하여 원하는 속성의 이름에 해당하는 MTA를 변경할 경우 실제로는 LDAP에서 다른 속성을 가져오게 됩니다. 그러나 항상 관련 속성의 MTA 방법과 구문적으로 일치하는 속성입니다. 속성은 LDAP_CAPTURE(기본값 없음), LDAP_RECIPIENTLIMIT(기본값 없음), LDAP_RECIPIENTCUTOFF(기본값 없음), LDAP_SOURCEBLOCKLIMIT(기본값 없음), LDAP_SOURCE_CHANNEL(기본값 없음), LDAP_PERSONAL_NAME(기본값 없음), LDAP_SOURCE_CONVERSION_TAG(기본값 없음), LDAP_PRIMARY_ADDRESS(mail), LDAP_ALIAS_ADDRESSES(mailAlternateAddress), LDAP_EQUIVALENCE_ADDRESSES(mailEquivalentAddress) 및 LDAP_SPARE_* 속성입니다.

결과 옵션 값은 다음과 같습니다.

```
REVERSE_URL=ldap:///$V?$N?sub?$R
```

항상 그랬던 것처럼 local.imta.schematag는 쉼표로 구분된 목록이 될 수 있습니다. 둘 이상의 스키마가 지원될 경우 중복 항목이 제거된 결합된 속성 목록이 사용됩니다.

또한 이 필터는 원래 제공되었던 주소뿐만 아니라 로컬 부분이 동일하지만 실제로 도메인 트리에서 발견된 도메인(182 페이지 “9.1.1.1 다시 쓰기 규칙 방법”에서 저장)을 가진 주소를 검색합니다. 도메인 트리 조회의 반복 특성은 두 개의 주소가 다를 수 있다는 것을 의미합니다.

예를 들어, 도메인 siroe.com이 도메인 트리에 표시되며 MTA가 다음 주소를 찾는다고 가정해 봅니다.

u@host1.siroe.com

$R 및 ims50 스키마 태그의 확장 결과로 얻어지는 필터는 다음과 같습니다.

```
(|(mail=u@siroe.com)
(mail=u@host1.siroe.com)
(mailAlternateAddress=u@siroe.com)
(mailAlternateAddress=u@host1.siroe.com)
(mailEquivalentAddress=u@siroe.com)
(mailEquivalentAddress=u@host1.siroe.com))
```

역방향 조회는 다양한 속성을 반환하고, MTA는 메일 속성(정확하게 LDAP_PRIMARY_ADDRESS에 명명된 속성)을 주소 역방향에 대한 속성으로 사용해야 합니다. mailEquivalentAddress(정확하게 LDAP_EQUIVALENCE_ADDRESSES에 명명된 속성)도 허용됩니다.

URL이 생성된 후 LDAP 검색이 수행됩니다. 검색에 성공하면 LDAP는 여러 속성을 임의의 순서로 반환합니다. 검색에 실패하거나 오류가 발생하면 원래 주소가 변경되지 않습니다.

주소 역방향 작업이 수행되는 빈도(특히 메일 헤더에 표시될 수 있는 주소 수가 지정된 경우)와 관련 디렉토리 쿼리의 비용으로 인해 부정적 및 긍정적 결과를 모두 캐시해야 합니다. 이 작업은 동적으로 확장되는 메모리 내장의 개방형 체인 해시 테이블을 통해 구현됩니다. 캐시의 최대 크기는 REVERSE_ADDRESS_CACHE_SIZE MTA 옵션(기본값 100000)으로 설정하며 캐시의 항목에 대한 시간 초과는 REVERSE_ADDRESS_CACHE_TIMEOUT MTA 옵션(기본값 600초)으로 설정합니다. 캐시는 실제로 LDAP URL 및 LDAP 결과가 아니라 주소 자체를 저장합니다.

9.3 비동기 LDAP 작업

비동기 조회는 경우에 따라 성능 문제를 야기할 수 있는 큰 LDAP 결과 전체를 메모리에 저장할 필요가 없게 만듭니다. MTA는 MTA에 의한 다양한 유형의 조회를 비동기식으로 수행하는 기능을 제공합니다.

비동기 LDAP 조회의 사용은 LDAP_USE_ASYNC MTA 옵션에 의해 제어됩니다. 이 옵션은 비트 인코딩 값입니다. 각 비트는 설정된 경우 MTA 내의 특정 LDAP 사용과 함께 비동기 LDAP 조회 사용을 활성화합니다.

표 9–11에서는 option.dat 파일의 LDAP_USE_ASYNC MTA 옵션에 대한 비트 및 값 설정을 보여 줍니다.

표 9–11 LDAP_USE_ASYNC MTA 옵션에 대한 설정

비트	값	특정 LDAP 사용
0	1	LDAP_GROUP_URL1(mgrpDeliverTo) URL
1	2	LDAP_GROUP_URL2(memberURL) URL
2	4	LDAP_GROUP_DN(UniqueMember) DN
3	8	auth_list, moderator_list, sasl_auth_list 및 sasl_moderator_list 비지정 목록 매개 변수 URL
4	16	cant_list, sasl_cant_list 비지정 목록 매개 변수 URL
5	32	originator_reply 비지정 목록 매개 변수 URL
6	64	deferred_list, direct_list, hold_list, nohold_list 비지정 목록 매개 변수 URL
7	128	username_auth_list, username_moderator_list, username_cant_list 비지정 목록 매개 변수 URL
8	256	별칭 파일 목록 URL
9	512	별칭 데이터베이스 목록 URL
10	1024	LDAP_CANT_URL(mgrpDisallowedBroadcaster) 외부 수준 URL
11	2048	LDAP_CANT_URL 내부 수준 URL
12	4096	LDAP_AUTH_URL(mgrpAllowedBroadcaster) 외부 수준 URL
13	8192	LDAP_AUTH_URL 내부 수준 URL
14	16384	LDAP_MODERATOR_URL(mgrpModerator) URL

LDAP_USE_ASYNC MTA 옵션의 기본값은 0이며 이것은 비동기 LDAP 조회가 기본적으로 비활성화된다는 것을 의미합니다.

9.4 설정 요약

Direct LDAP를 활성화하려면 다음 MAT 옵션을 설정해야 합니다.

```
ALIAS_MAGIC=8764
ALIAS_URL0=ldap:///$V?*?sub?$R
USE_REVERSE_DATABASE=4
USE_DOMAIN_DATABASE=0
REVERSE_URL=ldap:///$V?mail?sub?$Q
```

부속 도메인을 지원하려는 경우 다음 추가 옵션을 설정해야 합니다.

```
DOMAIN_MATCH_URL=ldap:///$B?msgVanityDomain?sub? \
(msgVanityDomain=$D)
ALIAS_URL1=ldap:///$B?*?sub? (&(msgVanityDomain=$D)$R)
ALIAS_URL2=ldap:///$1V?*?sub?(mailAlternateAddress=@$D)
```

이러한 옵션의 마지막 부분은 호스트된 도메인과 부속 도메인 모두에서 와일드카드로 지정된 로컬 부분의 대소문자를 처리합니다. 와일드카드로 지정된 로컬 부분에 대한 지원은 필요하지만 부속 도메인 지원이 필요하지 않은 경우에는 다음 옵션을 사용해야 합니다.

```
ALIAS_URL1=ldap:///$V?*?sub?&(mailAlternateAddress=@$D)
```

`filter ssrd:$A` 절을 MTA 구성 파일(`imta.cnf`)의 ims-ms 채널 정의에서 제거해야 합니다.

9.5 동일한 의미로 서로 다른 여러 LDAP 속성 처리

이제 MTA는 동일한 의미로 서로 다른 여러 LDAP 속성을 처리할 수 있습니다. 이 기능은 이전 버전에서도 항상 지원되었던 동일한 속성에 대한 여러 값 처리 기능과는 다릅니다. 받는 속성에 대한 처리는 의미에 따라 달라집니다. 가능한 옵션은 다음과 같습니다.

1. 서로 다른 속성은 의미가 없으므로 사용자 항목을 잘못된 것으로 렌더링합니다. Mail Server 버전 6.2 이상에서 이 처리는 별도로 명시하지 않는 한 모든 속성의 기본값입니다.
2. 서로 다른 여러 속성을 지정한 경우에는 하나를 임의로 선택하여 사용합니다. `LDAP_AUTOREPLY_SUBJECT`, `LDAP_AUTOREPLY_TEXT` 및 `LDAP_AUTOREPLY_TEXT_INT`는 모두 6.2에서만 이 처리를 수신합니다. 6.3 이상에서는 507 페이지 “17.4 휴가 자동 회신 속성”에 설명된 처리를 받습니다. 6.3에서는 `LDAP_SPARE_3` 및 `LDAP_PERSONAL_NAME` 속성이 이 범주에 추가되었습니다. 6.2 이전 버전에서는 모든 속성이 이 방법으로 처리되었습니다.
3. 서로 다른 여러 속성은 의미가 없으므로 동등한 것으로 처리해야 합니다. 이 처리는 현재 `LDAP_CAPTURE`, `LDAP_ALIAS_ADDRESSES`, `LDAP_EQUIVALENCE_ADDRESSES` 및 `LDAP_DETOURHOST_OPTIN`에 적용됩니다. `LDAP_DETOURHOST_OPTIN` 속성은 6.3 버전에 처음으로 추가되었습니다.

◆ ◆ ◆ 10 장

10 MTA 서비스 및 구성 정보

이 장에서는 일반 MTA 서비스 및 구성에 대해 설명합니다. 더 구체적이고 자세한 설명은 다른 장에서 확인할 수 있습니다. 이 장은 다음 내용으로 구성되어 있습니다.

10.1 MTA 구성 컴파일

`imta.cnf`, `mappings`, `aliases` 또는 `option.dat`와 같은 MTA 구성 파일이 수정될 때마다 구성을 다시 컴파일해야 합니다. 이 명령은 구성 파일을 공유 메모리(UNIX) 또는 동적 링크 라이브러리(NT)의 단일 이미지로 컴파일합니다.

컴파일된 구성은 재로드 가능한 정적 및 동적 부분을 가집니다. 동적 부분이 변경되고 `imsimta reload`를 실행하면 실행 중인 프로그램에서 동적 데이터를 재로드합니다. 동적 부분은 매핑 테이블, 별칭 및 조회 테이블입니다.

구성 정보를 컴파일하는 주된 이유는 성능 때문입니다. 또한 컴파일된 구성이 사용 중일 때 구성 파일 자체가 "live" 상태가 아니므로 컴파일된 구성을 사용하면 구성 변경 사항을 더 편리하게 테스트할 수 있습니다.

MTA의 구성 요소(예: 채널 프로그램)는 구성 파일을 읽어야 할 때마다 컴파일된 구성이 존재하는지 먼저 확인합니다. 컴파일된 구성이 존재할 경우 실행 중인 프로그램에 이미지가 추가됩니다. 이미지 추가 작업이 실패할 경우 MTA는 텍스트 파일을 읽는 이전의 방법으로 돌아갑니다.

reverse, forward 또는 일반 데이터베이스를 변경한 경우 변경 내용을 적용하려면 imsimta reload 명령을 실행합니다. imta.cnf, mappings 파일, aliases, conversions 또는 option.dat 파일을 변경한 경우 이 변경 내용이 작업 제어기에 영향을 주지 않으면 imsimta cnbuild 뒤에 imsimta restart smtp 명령을 실행해야 합니다. dispatcher.cnf를 변경한 경우에는 imsimta restart dispatcher 명령을 실행해야 합니다. 컴파일된 구성에 포함된 구성 파일을 변경한 경우 이 변경 내용이 작업 제어기에만 영향을 주고 SMTP 서버에는 영향을 주지 않으면 일반적으로 imsimta cnbuild 및 imsimta restart job_controller 명령을 실행해야 합니다.

컴파일된 구성에 포함된 구성 파일을 변경한 경우 이 변경 내용이 SMTP 서버와 작업 제어기에 둘 다 영향을 주면 다음 명령을 실행해야 합니다.

```
imsimta cnbuild
imsimta restart smtp
imsimta restart job_controller
```

이러한 명령에 대한 자세한 내용은 **Sun Java System Messaging Server 6.3 Administration Reference**의 "MTA Commands"를 참조하십시오.

다음과 같은 경우에도 작업 제어기를 다시 시작해야 합니다.

- Controller 구성 파일인 job_controller.cnf 또는 job_controller.site나 job_controller.cnf에 포함된 임의의 파일을 변경한 경우
- imta.cnf 파일에 채널 키워드 풀인 maxjobs, master, slave, single, single_sys 또는 multiple을 추가하거나 사용을 변경한 경우. 대신 imsimta cache -change -thread_depth=...를 통해 imta.cnf에 threaddepth 채널 키워드를 추가하거나 변경할 수 있습니다.
- Controller에서 기존 채널 작업을 시간 초과할 때까지 기다리는 대신 마스터 채널 작업의 변경 내용을 즉시 적용하려면 MTA 구성 또는 채널 옵션 파일에 관련된(거의 모든) 변경 시(mappings 파일 또는 MTA 데이터베이스에 대한 변경: (1) conversion, process, reprocess 등의 "중간" 채널에 중요할 수도 있지만 일반적으로 아웃바운드 채널 작업에는 관련이 없습니다. (2) 이러한 중간 채널이 중요한 경우 작업 제어기를 다시 시작하지 않고 imsimta reload를 통해 mappings 파일 또는 데이터베이스 변경을 처리할 수 있습니다.)변경 내용을 즉시 적용하려는 요구와 작업 제어기를 다시 시작할 경우의 손상 간에 적절한 균형을 유지해야 하며 특정 종류의 작업이 더 오래 실행되는 기간도 고려해야 합니다.

MTA 구성에는 imta.cnf와 이 파일에 포함된 모든 파일(예: internet.rules), alias 파일, mappings 파일, conversions 파일, option.dat 파일(및 이 파일에 포함된 모든 파일), imta.filter, reverse, forward 및 일반 데이터 파일, 일부 configutil 매개 변수가 포함됩니다.

imta.cnf에 대한 위의 모든 변경(예: 채널 정의에서 키워드 추가/변경) 시에는 작업 제어기의 다시 시작 여부에 관계없이 기본 명령인 imsimta cnbuild도 필요합니다.

위의 조건 중 하나로 인해 다시 시작해야 하는 경우가 아니면 특히 대기열에 많은 메시지가 있는 경우 작업 제어기를 다시 시작하지 않도록 하십시오.

작업 제어기를 다시 시작하지 않아도 되는 경우가 많으며 작업 제어기를 다시 시작하면 메시지 재시도 횟수, 지연된 알림 메시지, 바운스된 메시지 등이 재설정되므로 imsimta refresh 명령을 사용하는 것은 권장되지 않습니다.

10.2 MTA 구성 파일

주 MTA 구성 파일은 imta.cnf입니다. 기본적으로 이 파일은 *msg-svr-base*/config/imta.cnf에 위치합니다. 이 파일은 채널 다시 쓰기 규칙뿐만 아니라 MTA 채널 정의를 포함합니다. 다시 쓰여진 대상 주소와 관련된 채널은 대상 채널이 됩니다. 일반적으로 기본 imta.cnf를 사용하면 시스템이 적절하게 작동합니다.

이 절에서는 MTA 구성 파일에 대해 간략하게 소개합니다. MTA 구성 파일을 구성하는 다시 쓰기 규칙과 채널 정의를 구성하는 방법에 대한 자세한 내용은 11 장 및 12 장을 참조하십시오.

MTA 구성 파일을 수정하여 사이트에서 사용되는 채널을 설정하고 어떤 채널이 다시 쓰기 규칙을 통해 어떤 종류의 주소를 담당하는지 지정합니다. 이 구성 파일은 주소 유형을 적절한 채널과 연관시키는 전송 경로(다시 쓰기 규칙)와 사용 가능한 전송 방법(채널)을 지정하여 전자 메일 시스템의 레이아웃을 설정합니다.

구성 파일은 도메인 다시 쓰기 규칙과 채널 정의의 두 부분으로 구성됩니다. 도메인 다시 쓰기 규칙은 파일에서 앞 부분에 나타나며 빈 행으로 채널 정의와 구분됩니다. 채널 정의는 통틀어서 채널 테이블이라고 합니다. 개별 채널 정의는 채널 블록을 구성합니다.

imta.cnf 구성 파일의 다음 예는 메시지를 적절한 채널로 라우팅하는 데 다시 쓰기 규칙이 사용되는 방법을 보여 줍니다. 가능한 한 간단하게 하기 위해 도메인 이름은 사용되지 않았습니다. 다시 쓰기 규칙은 구성 파일의 상반부에 나타나며 채널 정의는 그 뒤를 이어 구성 파일의 하반부에 나타납니다.

```
! test.cnf - An example configuration file.   (1)!
! This is only an example of a configuration file. It serves
! no useful purpose and should not be used in a real system.
!
! Part I: Rewrite rules
a     $U@a-daemon           (2)
b     $U@b-daemon
c     $U%c@b-daemon
d     $U%d@a-daemon
      (3)
```

```
! Part II: Channel definitions
l      (4)
local-host

a_channel defragment charset7 usascii      (5)
a-daemon

b_channel noreverse notices 1 2 3
b-daemon

</opt/SUNWmsgsr/msg-tango/table/internet.rules    (6)
```

다음 목록에는 위 구성 파일의 주요 항목(괄호로 묶인 굵은체의 숫자가 표시된)이 설명되어 있습니다.

1. 느낌표(!)는 주석 행을 포함하는 데 사용됩니다. 느낌표는 첫 번째 열에 표시되어야 합니다. 그 밖의 다른 위치에 표시된 느낌표는 **리터럴** 느낌표로 해석됩니다.
2. 다시 쓰기 규칙은 구성 파일의 상반부에 나타납니다. 다시 쓰기 규칙의 행에는 빈 행이 포함될 수 없습니다. 첫 번째 열에서 느낌표로 시작되는 주석 행은 허용됩니다.
3. 파일에 나타나는 첫 번째 빈 행은 다시 쓰기 규칙 섹션의 끝이자 채널 블록의 시작을 의미합니다. 이러한 정의를 통틀어서 MTA가 사용할 수 있는 채널 및 각 채널과 연관된 이름을 정의하는 **채널 호스트 테이블**이라고 합니다.
4. 처음에 표시되는 채널 블록은 일반적으로 로컬 또는 `l` 채널입니다. 그런 다음 빈 행으로 각 채널 블록이 서로 분리됩니다. (`l` 채널 앞에 나타날 수 있는 `defaults` 채널은 예외입니다.)
5. 일반 채널 정의는 채널 이름(`a_channel`), 채널 구성을 정의하는 일부 키워드(`defragment charset7 usascii`) 및 **채널 태그**라고도 부르는 라우팅 시스템(`a-daemon`)으로 구성됩니다.
6. 구성 파일에는 다른 파일의 내용이 포함될 수 있습니다. 첫 번째 열에 보다 작음 기호(<)가 있을 경우 해당 행의 나머지 부분은 파일 이름으로 간주되며 파일 이름은 항상 절대 및 전체 파일 경로여야 합니다. 이 경우 파일이 열리고 파일의 내용이 해당 지점에서 구성 파일에 결합됩니다. 포함 파일은 최대 3개 수준 깊이까지 중첩될 수 있습니다. 구성 파일이 세계 공용인 것처럼 구성 파일에 포함된 모든 파일도 세계 공용이어야 합니다.

표 10–1에서는 앞의 구성에 의해 일부 예제 주소가 라우팅되는 방법을 보여 줍니다.

표 10–1 주소 및 관련 채널

주소	다음 채널의 대기열에 넣음
u@a	a_channel
u@b	b_channel

표 10–1 주소 및 관련 채널 *(계속)*

주소	다음 채널의 대기열에 넣음
u@c	b_channel
u@d	a_channel

MTA 구성 파일에 대한 자세한 내용은 174 페이지 “8.4 다시 쓰기 규칙”, 177 페이지 “8.5.3 채널 정의” 및 11 장을 참조하십시오.

주 – imta.cnf 파일이 변경될 때마다 MTA 구성을 다시 컴파일해야 합니다. 209 페이지 “10.1 MTA 구성 컴파일”을 참조하십시오.

10.3 매핑 파일

MTA의 구성 요소는 대부분 테이블 조회 지향 정보를 사용합니다. 이 유형의 테이블은 입력 문자열을 출력 문자열로 변환, 즉 **매핑**하는 데 사용됩니다. **매핑 테이블**이라고 부르는 이러한 테이블은 대개 두 개의 열로 제공됩니다. 첫 번째(왼쪽) 열은 일치시킬 가능한 입력 문자열(패턴)을 제공하며 두 번째(오른쪽) 열은 입력 문자열이 매핑되는 결과 출력 문자열(템플리트)을 제공합니다. MTA 프로세스에서 사용하는 테이블과 사용 시기에 대한 자세한 내용은 표 10–2를 참조하십시오.

대부분의 MTA 데이터베이스(다른 유형의 MTA 데이터를 포함하는 데이터베이스로서 매핑 테이블과 혼동해서는 안 됨)는 바로 이 테이블 유형의 인스턴스입니다. 그러나 MTA 데이터베이스 파일은 와일드카드 조회 기능을 제공하지 않으므로 와일드카드 일치를 위해 전체 데이터베이스를 스캔해야 한다는 점에서 본질적으로 비효율적입니다.

MTA mappings 파일은 여러 매핑 테이블을 지원합니다. 와일드카드 기능뿐만 아니라 다단계 및 반복 매핑 방법이 제공합니다. 이 방식은 특히 항목 수가 많을 경우에 데이터베이스를 사용하는 것보다 컴퓨팅 작업이 많이 요구됩니다. 그러나 동일한 데이터베이스에서 대부분의 항목을 불필요하게 만드는 유연성이 있기 때문에 결과적으로 전체 오버헤드가 줄어들 수 있습니다.

매핑 테이블은 MTA mappings 파일에 저장됩니다. 이 파일은 MTA tailor 파일에서 IMTA_MAPPING_FILE 옵션으로 지정되며 기본적으로 *msg-svr-base*/config/mappings입니다. mappings 파일의 내용은 재로드 가능한 섹션의 일부로 컴파일된 구성에 통합됩니다(209 페이지 “10.1 MTA 구성 컴파일” 참조). 이 파일은 세계 공용이어야 합니다. 세계 공용 액세스를 허용하지 않을 경우 오류 동작이 발생합니다. mappings 파일이 변경될 때마다 MTA 구성을 다시 컴파일해야 합니다. 209 페이지 “10.1 MTA 구성 컴파일”을 참조하십시오.

표 10–2에는 이 설명서에서 다루는 매핑 테이블이 나열되어 있습니다.

표 10–2 Messaging Server 매핑 테이블

매핑 테이블	페이지 설명
AUTH_REWRITE	인증 작업(SASL)에서 얻은 주소 지정 정보를 사용하여 헤더 및 봉투 주소를 수정하기 위해 authrewrite 키워드와 함께 사용됩니다. 341 페이지 "12.4.3 TCP/IP 연결 및 DNS 조회 지원"을 참조하십시오.
CHARSET-CONVERSION	어떤 종류의 채널 간 문자 세트 변환 및 메시지 서식 재설정을 수행해야 하는지 지정하는 데 사용됩니다. 420 페이지 "13.6 문자 세트 변환 및 메시지 형식 다시 지정"을 참조하십시오.
COMMENT_STRINGS	주소 헤더 주석(괄호로 묶인 문자열)을 수정하는 데 사용됩니다. 368 페이지 "12.6.13 주소 헤더 행의 주석 처리"를 참조하십시오.
CONVERSIONS	변환 채널에 대한 메시지 트래픽을 선택하는 데 사용됩니다. 404 페이지 "13.5.2 변환 처리를 위한 트래픽 선택"을 참조하십시오.
FORWARD	별칭 파일이나 별칭 데이터베이스를 사용하여 수행하는 것과 비슷한 전달을 수행하는 데 사용됩니다. 243 페이지 "10.9.3 정방향 조회 테이블 및 FORWARD 주소 매핑"을 참조하십시오.
FROM_ACCESS	봉투의 From 주소에 기초하여 메일을 필터링하는 데 사용됩니다. To 주소가 부적절한 경우 이 테이블을 사용합니다. 512 페이지 "18.2.1 액세스 제어 매핑 테이블 - 작업"을 참조하십시오.
INTERNAL_IP	내부의 시스템과 서브넷을 인식하는 데 사용됩니다. 526 페이지 "18.6 SMTP 릴레이 추가"를 참조하십시오.
IP_ACCESS	소스 채널, 원격 서버의 IP 주소 수, 현재 시도 중인 IP 주소의 색인을 기준으로 받는 연결을 차단하는 데 사용됩니다. 523 페이지 "18.3.5 IP_ACCESS 매핑 테이블"을 참조하십시오.
MAIL_ACCESS	SEND_ACCESS 및 PORT_ACCESS 테이블에서 발견한 결합된 정보에 기초하여 받는 연결을 차단하는 데 사용됩니다. 512 페이지 "18.2.1 액세스 제어 매핑 테이블 - 작업"을 참조하십시오.
NOTIFICATION_LANGUAGE	알림 메시지를 사용자 정의 또는 현지화하는 데 사용됩니다. 246 페이지 "10.10 전달 상태 알림 메일 제어"를 참조하십시오.
ORIG_MAIL_ACCESS	ORIG_SEND_ACCESS 및 PORT_ACCESS 테이블에서 발견한 결합된 정보에 기초하여 받는 연결을 차단하는 데 사용됩니다. 512 페이지 "18.2.1 액세스 제어 매핑 테이블 - 작업"을 참조하십시오.
ORIG_SEND_ACCESS	봉투의 From 주소, 봉투의 To 주소, 소스 및 대상 채널에 기초하여 받는 연결을 차단하는 데 사용됩니다. 512 페이지 "18.2.1 액세스 제어 매핑 테이블 - 작업"을 참조하십시오.
PERSONAL_NAMES	개인 이름(꺽쇠로 구분된 주소 앞의 문자열)을 수정하는 데 사용됩니다. 369 페이지 "12.6.14 주소 헤더 행에서 개인 이름 처리"를 참조하십시오.

표 10–2 Messaging Server 매핑 테이블 *(계속)*

매핑 테이블	페이지 설명
PORT_ACCESS	IP 번호를 기준으로 받는 연결을 차단하는 데 사용됩니다. 512 페이지 "18.2.1 액세스 제어 매핑 테이블 - 작업"을 참조하십시오.
REVERSE	주소를 내부 형식에서 공개 광고 형식으로 변환하는 데 사용됩니다. 239 페이지 "10.9 주소를 내부 형식에서 공개 형식으로 변환"을 참조하십시오.
SEND_ACCESS	봉투의 From 주소, 봉투의 To 주소, 소스 및 대상 채널에 기초하여 받는 연결을 차단하는 데 사용됩니다. 512 페이지 "18.2.1 액세스 제어 매핑 테이블 - 작업"을 참조하십시오.
*SMS_Channel*_TEXT	사이트 정의 텍스트 변환에 사용됩니다. 892 페이지 "C.2.5 사이트 정의 텍스트 변환"을 참조하십시오.
X-ATT-NAMES	매핑 테이블에서 매개 변수 값을 검색하는 데 사용됩니다. 412 페이지 "13.5.3.5 변환 항목에서 매핑 테이블 호출"을 참조하십시오.
X-REWRITE-SMS-ADDRESS	로컬 SMS 주소 유효성 검사에 사용됩니다. 891 페이지 "C.2.4 사이트 정의 주소 유효성 검사 및 변환"을 참조하십시오.

10.3.1 매핑 파일의 파일 형식

mappings 파일은 일련의 개별 테이블로 구성됩니다. 각 테이블은 이름으로 시작되며 이름의 첫 번째 열에는 항상 알파벳 문자가 옵니다. 테이블 이름 뒤에는 필수적으로 빈 행이 오고 이어서 테이블의 항목이 나타납니다. 항목은 0개 이상의 들여쓰기 행으로 구성되며 각 항목 행은 하나 이상의 공백 또는 탭으로 구분된 두 개의 열로 구성됩니다. 항목 안의 모든 공백은 $ 문자를 사용하여 인용해야 합니다. 각 매핑 테이블 이름 뒤와 각 매핑 테이블 사이에 빈 행이 있어야 하며 단일 테이블의 항목 사이에는 빈 행이 표시될 수 없습니다. 주석은 첫 번째 열에서 느낌표(!)로 시작해야 합니다.

결과 형식은 다음과 같이 나타납니다.

TABLE1_NAME

```
pattern1-1    template1-1
pattern1-2    template1-2
pattern1-3    template1-3
   .             .
   .             .
   .             .
pattern1-n    template1-n
```

TABLE2_NAME

```
pattern2-1    template2-1
```

```
pattern2-2    template2-2
pattern2-3    template2-3
    .             .
    .             .
    .             .
pattern2-n    template2-n
       .
       .
       .
```

TABLE3_NAME

```
         .
         .
         .
```

매핑 테이블 TABLE2_NAME을 사용하는 응용 프로그램은 pattern2-2 문자열을 template2-2에 지정된 것으로 매핑합니다. 각 패턴 및 템플리트는 256자 및 1024자까지 포함할 수 있습니다. 매핑 파일에서 한 행의 최대 크기는 4096자입니다. 과도한 수의 항목이 막대한 양의 CPU와 메모리를 소비할 수 있지만 매핑에 표시될 수 있는 항목 수에는 제한이 없습니다. 252자를 초과하는 긴 행은 끝에 백슬래시(\)를 포함하여 계속 이어질 수 있습니다. 두 열 사이의 공백과 첫 번째 열 앞에 있는 공백은 생략할 수 없습니다.

중복된 매핑 테이블 이름은 mappings 파일에서 허용되지 않습니다.

10.3.1.1 매핑 파일에 다른 파일 포함

mappings 파일에 다른 파일이 포함될 수 있습니다. 이렇게 하려면 다음 형식의 행을 사용합니다.

```
<file-spec
```

이 행을 사용하면 포함이 나타나는 지점에서 file-spec 파일의 내용이 mappings 파일에 포함됩니다. 파일 지정은 전체 파일 경로(디렉토리 등)를 지정해야 합니다. 이 방식으로 포함된 모든 파일은 세계 공용이어야 합니다. 또한 이러한 포함된 mappings 파일에는 주석이 허용됩니다. 포함은 최대 3개 수준까지 중첩될 수 있습니다. 포함 파일은 mappings 파일이 로드될 때 동시에 로드됩니다. 즉, 포함 파일은 요청 시 로드되지 않으므로 포함 파일 사용과 관련하여 성능 또는 메모리가 절약되지는 않습니다.

10.3.2 매핑 작업

mappings 파일의 모든 매핑은 일관된 방식으로 적용됩니다. 특정 매핑과 다음 매핑 사이에서 변경되는 유일한 사항은 입력 문자열의 소스와 매핑 출력이 사용되는 대상입니다.

매핑 작업은 항상 입력 문자열과 매핑 테이블로부터 시작됩니다. 매핑 테이블의 항목은 테이블에 표시된 순서대로 위에서 아래로 한 번에 하나씩 스캔됩니다. 각 항목의 왼쪽은 패턴으로 사용되며 입력 문자열은 대소문자를 구분하지 않는 방식으로 해당 패턴과 비교됩니다. MTA 프로세스에서 사용하는 테이블과 사용 시기에 대한 자세한 내용은 표 10–2를 참조하십시오. 이 절은 다음과 같은 하위 절로 구성되어 있습니다.

- 217 페이지 “10.3.2.1 매핑 항목 패턴”
- 219 페이지 “10.3.2.2 IP 일치”
- 219 페이지 “10.3.2.3 매핑 항목 템플리트”

10.3.2.1 매핑 항목 패턴

패턴은 와일드카드 문자를 포함할 수 있습니다. 특히 일반적인 와일드카드 문자가 허용됩니다. 별표(*)는 0개 이상의 문자와 일치하며 각 백분율 기호(%)는 하나의 문자와 일치합 니다. 별표, 백분율 기호, 공백 및 탭은 앞에 달러 기호($)를 추가하여 인용할 수 있습니다. 별표 또는 백분율 기호를 인용하면 특별한 의미가 사라집니다. 영구적으로 패턴이나 템플리트로 끝나는 것을 방지하기 위해 공백과 탭을 인용해야 합니다. 리터럴 달러 기호 문자는 이중($$)으로 표시해야 하며 첫 번째 달러 기호가 두 번째 기호를 인용합니다.

표 10–3 매핑 패턴 와일드카드

와일드카드	**설명**
%	정확하게 하나의 문자와 일치합니다.
*	왼쪽에서 오른쪽으로의 “최대” 일치를 사용하여 0개 이상의 문자와 일치합니다.
뒤로 일치	**설명**
$ n*	n번째 와일드카드 또는 글롭과 일치합니다.
수정자	**설명**
$_	왼쪽에서 오른쪽으로의 “최소” 일치를 사용합니다.
$@	이어지는 와일드카드 또는 글롭의 “저장”을 해제합니다.
$^	이어지는 와일드카드 또는 글롭의 “저장”을 설정합니다. 기본값입니다.
글롭 와일드카드	**설명**

표 10–3 매핑 패턴 와일드카드 *(계속)*

$A%	하나의 알파벳 문자(A-Z 또는 a-z)와 일치합니다.
$A*	0개 이상의 알파벳 문자(A-Z 또는 a-z)와 일치합니다.
$B%	하나의 이진수(0 또는 1)와 일치합니다.
$B*	0개 이상의 이진수(0 또는 1)와 일치합니다.
$D%	하나의 십진수(0-9)와 일치합니다.
$D*	0개 이상의 십진수(0-9)와 일치합니다.
$H%	하나의 16진수(0-9 또는 A-F)와 일치합니다.
$H*	0개 이상의 16진수(0-9 또는 A-F)와 일치합니다.
$O%	하나의 8진수(0-7)와 일치합니다.
$O*	0 개 이상의 8진수(0-7)와 일치합니다.
$S%	하나의 기호 집합 문자(예: 0-9, A-Z, a-z, _, $)와 일치합니다.
$S*	0개 이상의 기호 집합 문자(예: 0-9, A-Z, a-z, _, $)와 일치합니다.
$T%	하나의 탭 또는 세로 탭이나 공백 문자와 일치합니다.
$T*	0개 이상의 탭 또는 세로 탭이나 공백 문자와 일치합니다.
$X%	$H%의 동의어입니다.
$X*	$H*의 동의어입니다.
$[c]%	문자 c와 일치합니다.
$[c]*	문자 c의 모든 경우와 일치합니다.
$[c_1 c_2 ... c_n]%	문자 c_1, c_2 또는 c_n과 정확하게 한 번 일치합니다.
$[c_1 c_2 ... c_n]*	문자 c_1, c_2 또는 c_n의 모든 경우와 일치합니다.
$[c_1 -c_n]%	c_1에서 c_n 범위에 있는 문자 하나와 일치합니다.
$[c_1 -c_n]*	c_1에서 c_n 범위에 있는 모든 문자와 일치합니다.
$< IPv4>	IPv4 주소와 일치하며 비트를 무시합니다.
$(IPv4)	IPv4 주소와 일치하며 접두어 비트를 유지합니다.
${IPv6}	IPv6 주소와 일치합니다.

글롭 내에서, 즉 $[. . .] 구조 내에서 백슬래시 문자(\)는 인용 문자입니다. 글롭 내에서 리터럴 하이픈(-) 또는 오른쪽 대괄호(])를 나타내려면 하이픈이나 오른쪽 대괄호를 백슬래시로 인용해야 합니다.

패턴의 다른 모든 문자는 해당 문자 자체를 표시 및 일치시킵니다. 특히 작은따옴표 및 큰따옴표 문자와 괄호는 매핑 패턴이나 템플리트에서 특별한 의미가 없는 보통 문자에 불과합니다. 따라서 유효하지 않은 주소나 부분 주소에 해당하는 항목을 쉽게 작성할 수 있습니다.

여러 수정자를 지정하거나 수정자와 뒤로 일치를 지정하려면 단지 하나의 달러 기호가 구문에 사용됩니다. 예를 들어, 뒤로 일치 자체를 저장하지 않고 처음 와일드카드를 뒤로 일치시키려면 $@$0이 아니라 $@0을 사용합니다.

`imsimta test -match` 유틸리티를 사용하여 매핑 패턴을 테스트하고 특히 패턴에서의 와일드카드 동작을 테스트할 수 있다는 점을 유의하십시오.

별표 와일드카드는 입력 문자열의 왼쪽에서 오른쪽으로 작동하여 항목을 최대한 일치시킵니다. 예를 들어, 입력 문자열 `a/b/c`가 패턴 `*/*`와 비교되면 왼쪽 별표는 `a/b`와 일치하고 오른쪽 별표는 나머지 `c`와 일치합니다.

`$_` 수정자는 패턴에서 왼쪽에서 오른쪽으로 작동하여 와일드카드 일치를 최소화므로 최소한의 가능한 일치만 일치로 간주됩니다. 예를 들어, 문자열 `a/b/c`가 패턴 `$_*/$_*`와 비교되면 왼쪽 `$_*`는 `a`와 일치하고 오른쪽 `$_*` 는 `b/c`와 일치합니다.

10.3.2.2 IP 일치

IPv4 접두어 일치의 경우 IP 주소나 서브넷이 지정되며 선택적으로 슬래시와 접두어의 비트 수(일치하는 항목을 비교할 때 고려되는)가 뒤에 올 수 있습니다. 예를 들어, 다음은 `123.45.67.0` 서브넷의 모든 항목과 일치합니다.

```
$(123.45.67.0/24)
```

IPv4 무시 비트 일치의 경우 IP 주소나 서브넷이 지정되며 선택적으로 슬래시와 비트 수(일치하는 항목을 검사할 때 무시되는)가 뒤에 올 수 있습니다. 예를 들어, 다음은 `123.45.67.0` 서브넷의 모든 항목과 일치합니다.

```
$<123.45.67.0/8>
```

다음 예는 `123.45.67.4`에서 `123.45.67.7`까지의 범위에 속한 모든 항목과 일치합니다.

```
$<123.45.67.4/2>
```

IPv6 일치는 IPv6 주소 또는 서브넷과 일치합니다.

10.3.2.3 매핑 항목 템플리트

주어진 항목의 패턴 비교에 실패할 경우 어떠한 작업도 수행되지 않으며 다음 항목의 스캔이 진행됩니다. 비교에 성공할 경우 항목의 오른쪽이 출력 문자열을 생성하기 위한 템플리트로 사용됩니다. 템플리트가 사용되면 입력 문자열은 템플리트에 제공된 지침으로부터 생성된 출력 문자열로 효과적으로 대체됩니다.

템플리트의 거의 모든 문자는 단순히 그대로 출력됩니다. 단, 달러 기호($)의 경우는 예외입니다.

달러 기호 뒤에 달러 기호, 공백 또는 탭이 오면 출력 문자열에 달러 기호, 공백 또는 탭이 생성됩니다. 이러한 문자는 모두 출력 문자열에 삽입하기 위해 인용해야 한다는 점을 유의하십시오.

달러 기호 뒤에 오는 숫자 *n*은 대체를 요청하며 달러 기호 뒤에 오는 알파벳 문자는 "메타 문자"라고 합니다.메타 문자 자체는 템플리트가 생성한 출력 문자열에 나타나지 않지만 특수한 일부 대체 또는 처리를 생성합니다. 특수한 대체 및 표준 처리 메타 문자의 목록은 표 10–4를 참조하십시오. 그 밖의 다른 메타 문자는 매핑별 응용 프로그램용으로 예약됩니다.

$C, $E, $L 또는 $R 메타 문자는 일치하는 패턴의 템플리트에 존재할 경우 매핑 프로세스에 영향을 주고 프로세스의 계속 또는 종료를 제어한다는 점을 유의하십시오. 즉, 한 항목의 출력이 다른 항목의 입력이 되는 반복 매핑 테이블 항목을 설정할 수 있습니다. 일치하는 패턴의 템플리트가 $C, $E, $L 또는 $R 메타 문자를 포함하지 않을 경우 $E(매핑 프로세스의 즉시 종료)가 가정됩니다.

무한 루프를 방지하기 위해 매핑 테이블의 반복 통과 횟수가 제한됩니다. 이전 통과보다 길거나 같은 패턴으로 통과가 다시 시작될 때마다 카운터가 증가합니다. 문자열의 길이가 이전 것보다 짧을 경우 카운터는 0으로 재설정됩니다. 카운터가 10을 초과하면 매핑을 반복하려는 요청은 무시됩니다.

표 10–4 매핑 템플리트 대체 및 메타 문자

대체 시퀀스	대체 대상
$n	0부터 시작하여 왼쪽에서 오른쪽으로 계산된 *n*번째 와일드카드 필드
$#...#	일련 번호 대체
$]...[	URL 조회, 결과에서 대체
$|...|	지정된 매핑 테이블을 제공된 문자열에 적용
${...}	일반 데이터베이스 대체
$}*domain*,*attribute*{	도메인별 속성에 액세스하는 기능을 추가합니다. *domain*은 해당 도메인이고 *attribute*는 도메인과 연관된 속성입니다. 도메인이 있고 속성을 가지고 있는 경우 초기 값이 매핑 결과로 대체됩니다. 하지만 속성이나 도메인이 없으면 매핑 항목이 실패합니다. *attributes*는 도메인 LDAP 속성이거나 아래에 정의되어 있는 특수한 속성일 수 있습니다. _base_dn_ - 도메인의 사용자 항목에 대한 기본 DN _domain_dn_ - 도메인 항목의 DN 그 자체 _domain_name_ - 도메인의 이름(별칭의 반대 개념) _canonical_name_ - 도메인과 관련된 정규 이름

표 10–4 매핑 템플리트 대체 및 메타 문자 *(계속)*

대체 시퀀스	대체 대상
$[...]	사이트 제공 루틴 호출(결과에서 대체)
메타 문자	**설명**
$C	다음 테이블 항목에서 시작하는 매핑 프로세스를 계속합니다. 이 항목의 출력 문자열을 매핑 프로세스의 새 입력 문자열로 사용합니다.
$E	지금 매핑 프로세스를 종료합니다. 이 항목의 출력 문자열을 매핑 프로세스의 최종 결과로 사용합니다. $+1E exits immediately without interpreting the rest of the template.
$L	다음 테이블 항목에서 시작하는 매핑 프로세스를 계속합니다. 이 항목의 출력 문자열을 매핑 프로세스의 새 입력 문자열로 사용합니다. 테이블의 모든 항목이 사용되고 나면 첫 번째 테이블 항목에서 시작하는 통과를 하나 더 만듭니다. 이후의 일치하는 항목에서는 $C, $E 또는 $R 메타 문자를 사용하여 이 조건을 무시할 수 있습니다.
$R	매핑 테이블의 첫 번째 항목에서 시작하는 매핑 프로세스를 계속합니다. 이 항목의 출력 문자열을 매핑 프로세스의 새 입력 문자열로 사용합니다.
$nA	위치 0에서 시작하여 현재 주소의 n번째 왼쪽 문자를 삽입합니다. n을 생략하면 전체 주소가 삽입됩니다.
$nX	0에서 시작하여 메일 호스트의 n번째 왼쪽 구성 요소를 삽입합니다. n을 생략하면 전체 메일 호스트가 삽입됩니다.
$?x?	매핑 항목이 x%의 시간 동안 성공합니다.
$\	후속 텍스트를 소문자로 강제 지정합니다.
$^	후속 텍스트를 대문자로 강제 지정합니다.
$_	후속 텍스트를 원래 대소문자로 유지합니다.
$=	대체된 후속 문자가 LDAP 검색 필터에 삽입하기 적합하게 인용되도록 합니다. 모두 대문자로 적용합니다.
$:*x*	지정된 플래그가 설정된 경우에만 일치합니다.
$;*x*	지정된 플래그가 지워진 경우에만 일치합니다.

이 절은 다음과 같은 하위 절로 구성되어 있습니다.

- 222 페이지 “와일드카드 필드 대체($n)”
- 222 페이지 “텍스트 대소문자 제어($\, $^, $_)”
- 222 페이지 “처리 제어($C, $L, $R, $E)”
- 223 페이지 “특수 플래그 검사”
- 223 페이지 “항목의 임의적 성공 또는 실패($?x?)”
- 223 페이지 “일련 번호 대체($#...#)”
- 224 페이지 “URL 대체, $]...[”

- 225 페이지 “매핑 테이블 대체($|...|)”
- 225 페이지 “일반 조회 테이블 또는 데이터베이스 대체(${...})”
- 225 페이지 “사이트 제공 루틴 대체($[...])”
- 226 페이지 “UTF-8 문자열 생성”

와일드카드 필드 대체($n)

숫자 n이 뒤에 오는 달러 기호는 패턴의 *n*번째 와일드카드와 일치했던 항목으로 대체됩니다. 와일드카드는 0부터 시작하여 번호가 매겨집니다. 예를 들어, 다음 항목은 입력 문자열 PSI%A::B와 일치하며 결과 출력 문자열 b@a.psi.siroe.com을 생성합니다.

```
PSI$%*::*    $1@$0.psi.siroe.com
```

또한 입력 문자열 PSI%1234::USER도 일치하여 USER@1234.psi.siroe.com을 출력 문자열로 생성합니다. 입력 문자열 PSIABC::DEF는 이 항목의 패턴과 일치하지 않으며 어떤 작업도 발생하지 않습니다. 즉, 이 항목으로부터 출력 문자열이 생성되지 않습니다.

텍스트 대소문자 제어($\, $^, $_)

메타 문자 $\는 후속 텍스트를 소문자로 강제 지정하고 $^는 후속 텍스트를 대문자로 강제 지정하며 $_는 후속 텍스트를 원래 대소문자로 유지합니다. 예를 들어, 이러한 메타 문자는 매핑을 사용하여 대소문자가 중요한 주소를 변환할 경우에 유용할 수 있습니다.

처리 제어($C, $L, $R, $E)

$C, $L, $R 및 $E 메타 문자는 매핑 프로세스의 종료 여부와 종료 시기를 제어하여 매핑 프로세스에 영향을 줍니다. 각 메타 문자는 다음과 같습니다.

- $C를 사용하면 매핑 프로세스가 다음 항목에서 계속되며 현재 항목의 출력 문자열이 매핑 프로세스의 새 입력 문자열로 사용됩니다.
- $L을 사용하면 매핑 프로세스가 다음 항목에서 계속되며 현재 항목의 출력 문자열이 매핑 프로세스의 새 입력 문자열로 사용됩니다. 일치하는 항목이 없을 경우 항목을 하나 더 만들어 첫 번째 테이블 항목에서 시작하는 테이블을 통과하도록 합니다. $C, $E 또는 $R 메타 문자를 가진 이후의 일치하는 항목은 이 조건을 무시합니다.
- $R을 사용하면 매핑 프로세스가 테이블의 첫 번째 항목에서 계속되며 현재 항목의 출력 문자열이 매핑 프로세스의 새 입력 문자열로 사용됩니다.
- $E를 사용하면 매핑 프로세스가 종료하며 이 항목의 출력 문자열이 최종 출력이 됩니다. $E가 기본값입니다.

매핑 테이블 템플리트는 왼쪽에서 오른쪽으로 스캔됩니다. “성공” 또는 “실패”할 수 있는 항목(예: 일반 데이터베이스 대체 또는 임의 값 제어 항목)에 $C, $L 또는 $R 플래그를 설정하려면 해당 항목의 왼쪽에 $C, $L 또는 $R 메타 문자를 추가합니다. 그렇지 않을 경우 항목의 나머지 부분이 실패하면 플래그가 표시되지 않습니다.

특수 플래그 검사

일부 매핑 검사에서는 특수한 플래그를 설정합니다. 이러한 플래그를 설정한 다음 $:, $; 테스트의 일반 매핑 테이블 기능을 사용하여 플래그의 존재/부재를 테스트할 수 있습니다. $:x는 플래그 x가 설정된 경우에만 항목이 일치하게 합니다. $;x는 플래그 x가 없는 경우에만 항목이 일치하게 합니다. 매핑 테이블에 적용될 수 있는 특수 플래그는 해당 매핑 테이블에 대한 설명을 참조하십시오. 표 18–2에서 $A, $T, $S, $F 및 $D를 참조하십시오.

플래그 검사가 성공하면 항목이 계속되고 종료되도록 하고 플래그 검사가 실패하면 매핑 프로세스가 계속되도록 하려는 경우 항목에서는 플래그 검사 왼쪽에 $C 메타 문자를 사용하고 플래그 검사 오른쪽에 $E 플래그를 사용해야 합니다.

항목의 임의적 성공 또는 실패($?x?)

매핑 테이블 항목의 $?x?메타 문자를 사용하면 *x*%의 시간 동안 항목이 "성공"합니다. 나머지 시간에는 항목이 "실패"하며 매핑 항목의 입력에 대한 출력이 변경되지 않은 채 출력으로 사용됩니다. (매핑에 따라 항목 실패의 결과가 처음에 일치하지 않은 항목과 반드시 같은 것은 아닙니다.)*x*는 성공 비율을 지정하는 실수여야 합니다.

예를 들어, IP 주소가 123.45.6.78인 시스템이 많은 양의 SMTP 전자 메일을 사이트로 전송하고 있으며 관리자가 그 속도를 줄이려는 경우 다음과 같은 방법으로 PORT_ACCESS 매핑 테이블을 사용할 수 있습니다. 여기에서 연결 시도의 25%만 허용하고 나머지 75%의 시도를 거부해야 한다고 가정해 봅니다. 다음 PORT_ACCESS 매핑 테이블은 $Y(연결 허용)를 가진 항목이 25%의 시간 동안만 성공하도록 $?25?를 사용합니다. 나머지 75%의 시간동안 이 항목이 실패하면 맨 앞의 $C로 인해 MTA는 다음 항목에서 매핑을 계속합니다. 결과적으로 SMTP 오류가 발생하고Try again later 메시지가 표시되면서 연결 시도가 거부됩니다.

```
PORT_ACCESS

   TCP|*|25|123.45.6.78|*          $C$?25?$Y
   TCP|*|25|123.45.6.78|*          $N45s$ 4.40$ Try$ again$ later
```

일련 번호 대체($#...#)

$#...# 대체는 MTA 시퀀스 파일에 저장된 값을 증가시키고 해당 값을 템플리트로 대체합니다. 매핑 테이블 출력에 고유한 한정자가 존재하는 것이 바람직한 경우(예: 매핑 테이블을 사용하여 파일 이름을 생성하는 경우) 이 대체를 사용하여 증가하는 고유 문자열을 생성할 수 있습니다.

다음 구문 중 하나를 사용할 수 있습니다.

$#*seq-file-spec*|*radix*|*width*|*m*#

$#*seq-file-spec*|*radix*|*width*#

$#*seq-file-spec*|*radix*#

$#*seq-file-spec*#

필수 *seq-file-spec* 인수는 이미 존재하는 MTA 시퀀스 파일에 대한 전체 파일 지정입니다. 선택적 *radix* 및 *width* 인수는 각각 시퀀스 값을 출력하는 데 적용할 기수와 출력할 자릿수를 지정합니다. 기본 기수는 10이며 허용 범위는 -36에서 36까지입니다. 예를 들어, 기수 36은 숫자 0,...,9,A,...,Z로 표현되는 값을 제공합니다. 기본적으로 시퀀스 값은 본래 너비로 인쇄되지만 지정된 너비에 더 많은 자릿수가 요구될 경우 올바른 자릿수가 되도록 출력의 왼쪽 부분이 0으로 채워집니다. 너비가 명시적으로 지정된 경우 기수도 명시적으로 지정되어야 한다는 점을 유의하십시오.

선택적 *m* 인수는 모듈러스입니다. 이 네 번째 인수를 지정하면 삽입되는 값은 파일 mod *m*에서 검색되는 시퀀스 번호입니다. 기본값은 모듈러스 작업을 수행하지 않는 것입니다.

위에서 언급한 것처럼 매핑에서 참조되는 MTA 시퀀스 파일은 이미 존재해야 합니다. MTA 시퀀스 파일을 만들려면 다음 UNIX 명령을 사용합니다.

touch *seq-file-spec*

또는

cat >*seq-file-spec*

매핑 테이블을 사용하여 액세스하는 일련 번호 파일은 세계 공용일 경우에만 제대로 작동합니다. 또한 이러한 일련 번호 파일을 사용하려면 `imta_tailor` 파일에서 `nobody`로 구성된 MTA 사용자 계정이 있어야 합니다.

URL 대체, $]...[

`$]`*url*`[` 형식의 대체는 특수하게 처리됩니다. *url*은 `file:` 및 `data:`를 포함하여 지원되는 모든 URL 유형이 될 수 있습니다.호스트와 포트가 생략된 표준 LDAP URL을 사용할 수 있으며, 대신 `LDAP_HOST` 및 `LDAP_PORT` 옵션을 사용하여 호스트와 포트를 지정합니다. 즉, LDAP URL은 다음과 같이 지정해야 합니다.

`ldap:///`*dn*`[?`*attributes*`[?`*scope*`?`*filter*`]]`

여기에서 대괄호 문자 [및]는 URL의 선택적 부분을 나타냅니다. *dn*은 필수 항목으로서 검색 기준을 지정하는 고유 이름입니다. 선택 항목인 URL의 *attributes*, *scope* 및 *filter* 부분은 반환할 정보를 더 구체화합니다. 즉, *attributes*는 이 LDAP 쿼리와 일치하는 LDAP 디렉토리 항목에서 반환될 속성을 지정합니다. *scope*는 `base`(기본값), `one` 또는 `sub`가 될 수 있으며 *filter*는 일치하는 항목의 특성을 설명합니다.

특정 LDAP URL 대체 시퀀스를 LDAP 쿼리 URL 내에서 사용할 수 있습니다. URL의 길이는 1024자까지 가능합니다. 매핑과 다른 매핑에 대한 매핑 호출로 만들어지는 표현식에도 이 사항이 적용됩니다.

매핑 테이블 대체($|...|)

$|*mapping* ;*argument*| 형식의 대체는 특수하게 처리됩니다. MTA는 MTA `mappings` 파일에서 *mapping*이라는 보조 매핑 테이블을 찾은 후 *argument*를 명명된 이 보조 매핑 테이블에 대한 입력으로 사용합니다. 명명된 보조 매핑 테이블은 존재해야 하며 성공할 경우 해당 출력에서 `$Y` 플래그를 설정해야 합니다. 명명된 보조 매핑 테이블이 존재하지 않거나 `$Y` 플래그를 설정하지 않을 경우 해당 보조 매핑 대체가 실패하고 원래 매핑 항목이 실패로 간주되어 원래 입력 문자열이 출력 문자열로 사용됩니다.

매핑 테이블 대체를 수행하는 매핑 테이블 항목에서 `$C`, `$R` 또는 `$L`과 같은 처리 제어 메타 문자를 사용하려는 경우 매핑 테이블 템플리트에서 매핑 테이블 대체의 왼쪽에 처리 제어 메타 문자를 두어야 한다는 점을 유의하십시오. 그렇지 않을 경우 매핑 테이블 대체가 "실패"하면 처리 제어 메타 문자가 표시되지 않습니다.

일반 조회 테이블 또는 데이터베이스 대체(${...})

$(*text*) 형식의 대체는 특수하게 처리됩니다. *text* 부분은 일반 조회 테이블이나 데이터베이스에 액세스하기 위한 키로 사용됩니다. 자세한 내용은 240 페이지 "10.9.1 MTA 텍스트 데이터베이스"를 참조하십시오. 테이블에서 *text*가 발견될 경우 테이블의 해당 템플리트가 대체됩니다. *text*가 테이블의 항목과 일치하지 않을 경우 입력 문자열이 변경되지 않은 채로 출력 문자열로 사용됩니다.

일반 조회 테이블을 사용하는 경우에는 MTA 옵션 `use_text_databases`의 낮은 순서 비트를 설정해야 합니다. 즉 기수로 설정합니다. `imsimta cnbuild`를 사용하여 컴파일을 수행하고 `imsimta reload`를 사용하여 재로드 가능한 데이터를 재로드하는 방식으로 `general.txt`의 변경 사항을 MTA 구성으로 컴파일해야 합니다.

일반 데이터베이스를 사용하는 경우 데이터베이스는 제대로 작동하기 위해 세계 공용이어야 합니다.

일반 테이블 대체를 수행하는 매핑 테이블 항목에서 $C, $R 또는 $L과 같은 처리 제어 메타 문자를 사용하려는 경우 매핑 테이블 템플리트에서 일반 테이블 대체의 왼쪽에 처리 제어 메타 문자를 두어야 합니다. 그렇지 않을 경우 일반 테이블 대체가 "실패"하면 처리 제어 메타 문자가 표시되지 않습니다.

사이트 제공 루틴 대체($[...])

$[*image*,*routine*,*argument*] 형식의 대체는 특수하게 처리됩니다. image, routine, argument 부분은 사용자 제공 루틴을 검색 및 호출하는 데 사용됩니다. UNIX 런타임에서 MTA는 `dlopen` 및 `dlsym`을 사용하여 공유 라이브러리 *image*로부터 *routine* 루틴을 동적으로 로드 및 호출합니다. 이어서 *routine* 루틴은 다음 인수 목록을 가진 함수로 호출됩니다.

```
status = routine (argument, arglength, result, reslength)
```

argument 및 result는 252바이트 길이의 문자열 버퍼입니다. argument 및 result는 포인터로 문자열에 전달됩니다(예: C에서는 char*로). arglength 및 reslength는 참조로 전달되는 서명된 긴 정수입니다. 입력의 경우 argument는 매핑 테이블 템플리트의 *argument* 문자열을 포함하고 arglength는 해당 문자열의 길이를 포함합니다. 반환 시에 결과 문자열은 result에 포함되고 그 길이는 reslength에 포함되어야 합니다. 그런 다음 이 결과 문자열은 매핑 테이블 템플리트에서 $[image,routine,argument]를 대체합니다. *routine* 루틴은 매핑 테이블 대체가 실패할 경우에는 0을 반환하고 성공할 경우에는 1을 반환해야 합니다. 대체가 실패할 경우 일반적으로 원래 입력 문자열이 그대로 출력 문자열로 사용됩니다.

사이트 제공 루틴 대체를 수행하는 매핑 테이블 항목에서 $C, $R 또는 $L과 같은 처리 제어 메타 문자를 사용하려는 경우 매핑 테이블 템플리트에서 사이트 제공 루틴 대체의 왼쪽에 처리 제어 메타 문자를 두어야 합니다. 그렇지 않을 경우 매핑 테이블 대체가 "실패"하면 처리 제어 메타 문자가 표시되지 않습니다.

사이트 제공 루틴 설명선 기법을 사용하면 MTA의 매핑 프로세스를 모든 종류의 복잡한 방법으로 확장할 수 있습니다. 예를 들어, PORT_ACCESS 또는 ORIG_SEND_ACCESS 매핑 테이블에서 특정한 유형의 로드 모니터링 서비스를 호출할 수 있으며 결과를 사용하여 연결이나 메시지를 수락할지 여부를 결정할 수 있습니다.

사이트 제공 공유 라이브러리 이미지 image는 세계 공용이어야 합니다.

UTF-8 문자열 생성

일반 매핑 테이블 기능의 유니코드 문자 값에서 UTF-8 문자열을 만들 수 있습니다. 유니코드 메타 문자의 순서는 다음 형식으로 나타납니다.

```
$&A0A0,20,A1A1&
```

이 형식에서 A0A0, 20 및 A1A1 위치에 문자가 포함되는 UTF-8 문자열을 만들어냅니다.

10.4 기타 MTA 구성 파일

imta.cnf 파일 외에도 Messaging Server는 MTA 서비스를 구성하는 데 도움이 되는 다른 여러 구성 파일을 제공합니다. 표 10–5에는 이러한 파일이 요약되어 있습니다. 이 절은 다음과 같은 하위 절로 구성되어 있습니다.

- 231 페이지 "10.4.7 조정 파일"
- 231 페이지 "10.4.8 작업 제어기 파일"

reverse, forward 또는 일반 데이터베이스를 변경한 경우 변경 내용을 적용하려면 imsimta reload 명령을 실행합니다(240 페이지 "10.9.1 MTA 텍스트 데이터베이스" 참조). job_controller에 영향을 주지 않도록 imta.cnf, mappings 파일, aliases, conversions 또는 option.dat 파일을 변경한 경우에는 imsimta cnbuild 뒤에 imsimta restart smtp 명령을 실행해야 합니다. dispatcher.cnf를 변경한 경우에는 imsimta restart dispatcher 명령을 실행해야 합니다. 컴파일된 구성에 포함된 구성 파일을 변경한 경우 이 변경 내용이 작업 제어기에만 영향을 주고 SMTP 서버에는 영향을 주지 않으면 일반적으로 imsimta cnbuild 및 imsimta restart job_controller 명령을 실행해야 합니다.

이러한 명령에 대한 자세한 내용은 **Sun Java System Messaging Server 6.3 Administration Reference**의 "MTA Commands"를 참조하십시오.

표 10–5 MTA 구성 파일

파일	설명
228 페이지 "10.4.1 별칭 파일"(필수)	디렉토리에 존재하지 않는 별칭을 구현합니다. *msg-svr-base*/config/aliases
228 페이지 "10.4.2 TCP/IP(SMTP) 채널 옵션 파일"(SMTP 옵션 파일이라고도 부름)	채널별 옵션을 설정합니다. *msg-svr-base*/config/*channel*_option
228 페이지 "10.4.3 변환 파일"	메시지 본문 부분의 변환을 제어하기 위해 변환 채널에 사용됩니다. *msg-svr-base*/config/conversions
229 페이지 "10.4.4 디스패처 구성 파일"(필수)	디스패처를 위한 구성 파일입니다. *msg-svr-base*/config/dispatcher.cnf
231 페이지 "10.4.8 작업 제어기 파일"(필수)	**작업 제어기**에 사용되는 구성 파일입니다. /*msg-svr-base*/config/job_controller.cnf
MTA 구성 파일(필수)	채널 정의뿐만 아니라 주소 다시 쓰기 및 라우팅에 사용됩니다. /*msg-svr-base*/config/imta.cnf
213 페이지 "10.3 매핑 파일"(필수)	매핑 테이블의 저장소입니다. /*msg-svr-base*/config/mappings
230 페이지 "10.4.6 옵션 파일"	전역 MTA 옵션 파일입니다. /*msg-svr-base*/config/option.dat
231 페이지 "10.4.7 조정 파일"(필수)	위치와 일부 조정 매개 변수를 지정하기 위한 파일입니다. /*msg-svr-base*/config/imta_tailor

표 10–5 MTA 구성 파일 (계속)

파일	설명
일반 조회 테이블(선택 사항)	일반 조회 기능은 일반 데이터베이스와 동일합니다. 재로드 가능한 컴파일된 구성의 일부입니다. 위치와 일부 조정 매개 변수를 지정하기 위한 파일입니다. */msg-svr-base*/config/general.txt
정방향 조회 테이블(선택 사항)	To: 주소에 대한 주소에 적용되지 정방향 데이터베이스와 동일하며 재로드 가능한 컴파일된 구성의 일부입니다. */msg-svr-base*/config/forward.txt
역방향 조회 테이블(선택 사항)	From: 주소에 대한 주소에 적용되지 역방향 데이터베이스와 동일하며재로드 가능한 컴파일된 구성의 일부입니다. */msg-svr-base*/config/reverse.txt

10.4.1 별칭 파일

별칭 파일 aliases는 디렉토리에서 설정되지 않은 별칭을 설정합니다. 특히 루트의 주소를 좋은 예로 들 수 있습니다. 이 파일에 설정된 별칭은 디렉토리에 동일한 별칭이 존재할 경우 무시됩니다. 별칭 및 aliases 파일에 대한 자세한 내용은 236 페이지 "10.5 별칭"을 참조하십시오.

aliases 파일을 변경한 후 변경 사항을 적용하려면 MTA를 다시 시작해야 합니다.

10.4.2 TCP/IP(SMTP) 채널 옵션 파일

TCP/IP 채널 옵션 파일은 TCP/IP 채널의 다양한 특성을 제어합니다. 채널 옵션 파일은 MTA 구성 디렉토리에 저장하고 *x*_option으로 이름을 지정해야 합니다. 여기서 *x*는 채널의 이름입니다. 예를 들어, *msg-svr-base*/config/tcp_local_option입니다. 자세한 내용은 333 페이지 "12.4.1 SMTP 채널 옵션 구성"을 참조하십시오. 모든 채널 옵션 키워드와 구문에 대한 전체 정보는 **Sun Java System Messaging Server 6.3 Administration Reference**를 참조하십시오.

10.4.3 변환 파일

변환 파일 conversions는 변환 채널이 MTA를 통과하는 메시지에 대해 변환을 수행하는 방법을 지정합니다. MTA 트래픽의 모든 하위 집합을 변환하도록 선택할 수 있으며 임의의 프로그램 또는 명령 프로시저 집합을 사용하여 변환 처리를 수행할 수 있습니다. MTA는 각 본문 부분에 대한 적절한 변환을 선택하기 위해 변환 파일을 확인합니다.

이 파일의 구문에 대한 자세한 내용은 401 페이지 "13.5 변환 채널"을 참조하십시오.

10.4.4 디스패처 구성 파일

디스패처 구성 파일 dispatcher.cnf는 디스패처 구성 정보를 지정합니다. 기본 구성 파일은 설치 시 작성되며 변경 없이 사용할 수 있습니다. 그러나 보안이나 성능상의 이유로 기본 구성 파일을 수정하려는 경우 dispatcher.cnf 파일을 편집하여 원하는 사항을 수정할 수 있습니다. 이에 대한 개념적 정보는 173 페이지 "8.3 디스패처"를 참조하십시오.

디스패처 구성 파일 형식은 다른 MTA 구성 파일의 형식과 비슷합니다. 옵션을 지정하는 행은 다음 형식을 가집니다.

option=value

*option*은 옵션의 이름이며 *value*는 옵션이 설정되는 문자열 또는 정수입니다. *option*이 정수 값을 가질 경우 *b%v* 형식의 표기법을 사용하여 기수를 지정할 수 있습니다. 여기에서 *b*는 기수 10으로 표현되는 기수이며 *v*는 기수 *b*로 표현되는 실제 값입니다. 이러한 옵션 지정은 다음 형식의 행을 사용하여 다음 옵션 설정이 적용되는 서비스에 해당하는 섹션으로 그룹화됩니다.

[SERVICE=*service-name*]

*service-name*은 서비스의 이름입니다. 이러한 섹션 태그 앞에 표시되는 초기 옵션 지정은 모든 섹션에 전역적으로 적용됩니다.

다음은 샘플 디스패처 구성 파일(dispatcher.cnf)입니다.

```
! The first set of options, listed without a [SERVICE=xxx]
! header, are the default options that will be applied to all
! services.
!
MIN_PROCS=0
MAX_PROCS=5
MIN_CONNS=5
MAX_CONNS=20
MAX_LIFE_TIME=86400
MAX_LIFE_CONNS=100
MAX_SHUTDOWN=2
!
! Define the services available to Dispatcher
!
[SERVICE=SMTP]
PORT=25
IMAGE=msg-svr-base/lib/tcp_smtp_server
LOGFILE=msg-svr-base/log/tcp_smtp_server.log
```

이 파일의 매개 변수에 대한 자세한 내용은 **Sun Java System Messaging Server 6.3 Administration Reference**를 참조하십시오.

10.4.5 매핑 파일

mappings 파일은 MTA가 입력 문자열을 출력 문자열로 매핑하는 방법을 정의합니다.

MTA의 구성 요소는 대부분 테이블 조회 지향 정보를 사용합니다. 일반적으로 이러한 종류의 테이블은 입력 문자열을 출력 문자열로 변환(즉, 매핑)하는 데 사용됩니다. 매핑 테이블이라고 부르는 이러한 테이블은 두 개의 열, 즉 가능한 입력 문자열을 제공하는 첫 번째(왼쪽) 열과 연관된 입력에 대한 결과 출력 문자열을 제공하는 두 번째(오른쪽) 열로 제공됩니다. 대부분의 MTA 데이터베이스는 이러한 매핑 테이블 유형의 인스턴스입니다. 그러나 MTA 데이터베이스 파일은 와일드카드 조회 기능을 제공하지 않으므로 와일드카드 일치를 위해 전체 데이터베이스를 스캔해야 한다는 점에서 본질적으로 비효율적입니다.

mappings 파일은 여러 매핑 테이블을 지원하기 위한 기능을 MTA에 제공합니다. 완전한 와일드카드 기능이 제공되는 것 외에도 다단계 및 반복 매핑 방법을 사용할 수 있습니다. 이 방식은 특히 항목 수가 많을 경우에 데이터베이스를 사용하는 것보다 컴퓨팅 작업이 많이 요구됩니다. 그러나 동일한 데이터베이스에서 대부분의 항목을 불필요하게 만드는 유연성이 있기 때문에 결과적으로 전체 오버헤드가 줄어들 수 있습니다.

imsimta test -mapping 명령을 사용하여 매핑 테이블을 테스트할 수 있습니다. mappings 파일 및 test -mapping 명령의 구문에 대한 자세한 내용은 213 페이지 “10.3 매핑 파일” 및 **Sun Java System Messaging Server 6.3 Administration Reference**를 참조하십시오.

mappings 파일을 변경한 후 MTA를 다시 시작하거나 imsimta reload 명령을 실행해야 합니다.

10.4.6 옵션 파일

옵션 파일 option.dat는 채널별 옵션과 달리 전역 MTA 옵션을 지정합니다.

옵션 파일을 사용하면 MTA에 전체적으로 적용되는 다양한 매개 변수의 기본값을 무시할 수 있습니다. 특히 옵션 파일은 구성 및 별칭 파일을 읽어오는 다양한 테이블의 크기를 설정하는 데 사용됩니다. 또한 옵션 파일을 사용하여 MTA가 수락하는 메시지의 크기를 제한하고 MTA 구성에 허용되는 채널 수를 지정하며 허용되는 다시 쓰기 규칙 수를 설정하는 등의 작업을 수행할 수 있습니다.

option.dat에서 #, ! 또는 ;으로 시작하는 행은 행이 계속된다는 것을 의미하는 후행 \가 바로 앞 행에 있는 경우에도 주석 행으로 처리됩니다. 이것은 이러한 문자를 포함할 수 있는 긴 옵션(특히 전달 옵션)에서 주의해야 한다는 것을 의미합니다.

일반적으로 # 또는 !로 시작하는 연속 행을 가지는 전달 옵션의 경우 이를 처리할 수 있는 안전하고 간단한 방법이 존재합니다.

옵션 파일의 구문에 대한 자세한 내용은 **Sun Java System Messaging Server 6.3 Administration Reference**를 참조하십시오.

10.4.7 조정 파일

조정 파일 imta_tailor는 다양한 MTA 구성 요소의 위치를 설정합니다. MTA가 제대로 작동하려면 imta_tailor 파일이 항상 *msg-svr-base*/config 디렉토리에 상주해야 합니다.

이 파일을 편집하여 특정 설치의 변경 사항을 반영할 수 있지만 이렇게 하려면 주의를 기울여야 합니다. 이 파일을 변경한 후에는 MTA를 다시 시작해야 합니다. MTA를 종료한 상태에서 변경을 수행하는 것이 더 바람직합니다.

주 – 꼭 필요한 경우가 아니면 이 파일을 편집해서는 안 됩니다.

이 파일에 대한 자세한 내용은 **Sun Java System Messaging Server 6.3 Administration Reference**를 참조하십시오.

10.4.8 작업 제어기 파일

작업 제어기는 메시지 전달을 위해 채널 작업을 작성 및 관리합니다이러한 채널 작업은 작업 제어기 내의 처리 풀 안에서 실행됩니다. 풀은 채널 작업이 실행되는 "장소"로 생각할 수 있습니다. 풀은 작업 세트가 풀 외부의 작업과 자원을 놓고 경쟁하지 않고도 작동할 수 있는 컴퓨팅 영역을 제공합니다. (작업 제어기의 개념과 채널 키워드 구성에 대한 자세한 내용은 179 페이지 "8.7 작업 제어기", 356 페이지 "12.5.4 채널 실행 작업의 처리 풀" 및 356 페이지 "12.5.5 서비스 작업 제한"을 참조하십시오.)

작업 제어기 파일 job_controller.cnf는 다음 채널 처리 정보를 지정합니다.

- 다양한 풀을 정의합니다.
- 모든 채널, 마스터 프로그램 이름 및 슬레이브 프로그램 이름(해당 사항이 있을 경우)을 지정합니다.

imta.cnf file에서 pool **키워드**를 사용하여 job_controller.cnf에서 정의된 프로세스 풀의 이름을 지정할 수 있습니다. 예를 들어, 샘플 job_controller.cnf 파일의 다음 조각은 MY_POOL 풀을 정의합니다.

```
[POOL=MY_POOL]
job_limit = 12
```

샘플 imta.cnf 파일의 다음 단편은 채널 블록에서 MY_POOL 풀을 지정합니다.

```
channel_x pool MY_POOL
channel_x-daemon
```

기본 풀 구성과 관련된 매개 변수를 수정하거나 추가 풀을 추가하려는 경우 job_controller.cnf 파일을 편집한 다음 작업 제어기를 중지했다가 다시 시작할 수 있습니다.

작업 제어기 구성 파일의 첫 번째 풀은 풀 이름을 지정하지 않는 모든 요청에 사용됩니다. MTA 구성 파일(imta.cnf)에 정의된 MTA 채널은 pool 채널 키워드 뒤에 풀 이름을 사용하여 처리 요청을 특정 풀을 향하도록 할 수 있습니다. 풀 이름은 작업 제어기 구성의 풀 이름과 일치해야 합니다. 작업 제어기가 요청된 풀 이름을 인식하지 않을 경우 요청은 무시됩니다.

초기 구성에서는 DEFAULT, LOCAL_POOL, IMS_POOL, SMTP_POOL 풀이 정의됩니다.

10.4.8.1 사용 예

일반적으로 일부 채널의 처리를 다른 채널의 처리와 차별화하려는 경우 추가 풀 정의를 작업 제어기 구성에 추가합니다. 또한 다른 특성을 가진 풀을 사용할 수도 있습니다. 예를 들어, 일부 채널에서 처리하도록 허용된 동시 요청의 개수를 제어해야 할 수 있습니다. 이렇게 하려면 작업 제한이 있는 새 풀을 만든 다음 pool 채널 키워드를 사용하여 해당 채널을 더 적절한 새 풀로 보냅니다.

풀 정의 외에도 작업 제어기 구성 파일에는 각 채널에 대해 작업 제어기가 요청을 처리하는 데 사용해야 하는 MTA 채널 및 명령 테이블이 포함되어 있습니다. 요청의 두 가지 유형은 "마스터"와 "슬레이브"입니다. 일반적으로 채널 마스터 프로그램은 채널의 MTA 메시지 대기열에 저장된 메시지가 있을 때 호출됩니다. 마스터 프로그램은 메시지를 대기열에서 제외합니다.

슬레이브 프로그램은 채널을 폴링하고 해당 채널에 대한 모든 인바운드 메시지를 가져오기 위해 호출됩니다. 거의 모든 MTA 채널에 마스터 프로그램이 있지만 슬레이브 프로그램은 없거나 불필요합니다. 예를 들어, TCP/IP를 통해 SMTP를 처리하는 채널은 슬레이브 프로그램을 사용하지 않는데, 그 이유는 네트워크 서비스인 SMTP 서버가 받는 SMTP 메시지를 수신하기 때문입니다. SMTP 채널의 마스터 프로그램은 MTA의 SMTP 클라이언트입니다.

채널과 관련된 대상 시스템이 한 번에 하나의 메시지만 처리할 수 있는 경우 작업 제한이 1인 새로운 유형의 풀을 만들어야 합니다.

```
[POOL=single_job]
job_limit=1
```

이와 달리 대상 시스템에 충분한 병행성이 있을 경우 작업 제한을 더 높은 값으로 설정할 수 있습니다.

예 10–1은 샘플 작업 제어기 구성 파일을 보여 줍니다. 사용 가능한 옵션은 표 10–6에 나와 있습니다.

예 10–1 UNIX의 샘플 작업 제어기 구성 파일

```
!MTA Job Controller configuration file
!
!Global defaults
tcp_port=27442          (1)
```

예 10–1 UNIX의 샘플 작업 제어기 구성 파일 *(계속)*

```
secret=never mind
slave_command=NULL      (2)
max_life_age=3600       (3)
!
!
!Pool definitions
!
[POOL=DEFAULT]          (4)
job_limit=10            (5)
!
[POOL=LOCAL_POOL]
job_limit=10
!
[POOL=IMS_POOL]
job_limit=1
!
[POOL=SMTP_POOL]
job_limit=1
!
!Channel definitions
!
!
[CHANNEL=l]             (6)
master_command=msg-svr-base/lib/l_master
!
[CHANNEL=ims-ms]
master_command=msg-svr-base/lib/ims_master
!
[CHANNEL=tcp_*]         (7)
master_command=msg-svr-base/lib/tcp_smtp_client
```

위 예의 괄호로 묶인 굵은체의 숫자가 표시된 주요 항목은 다음과 같습니다.

1. 이 전역 옵션은 작업 제어기가 요청을 수신하는 TCP 포트 번호를 정의합니다.
2. 후속 [CHANNEL] 섹션에 대한 기본 SLAVE_COMMAND를 설정합니다.
3. 후속 [CHANNEL] 섹션에 대한 기본 MAX_LIFE_AGE를 설정합니다.
4. 이 [POOL] 섹션은 DEFAULT라는 풀을 정의합니다.
5. 이 풀의 JOB_LIMIT를 10으로 설정합니다.
6. 이 [CHANNEL] 섹션은 l이라는 UNIX 로컬 채널에 적용됩니다. 이 섹션에 필요한 유일한 정의는 작업 제어기에서 이 채널을 실행하기 위해 사용하는 master_command입니다. 채널 이름에 와일드카드가 없기 때문에 채널은 정확하게 일치해야 합니다.
7. 이 [CHANNEL] 섹션은 이름이 tcp_*로 시작하는 모든 채널에 적용됩니다. 이 채널 이름은 와일드카드를 포함하므로 이름이 tcp_로 시작하는 모든 채널과 일치합니다.

추가 풀을 추가하는 예

작업 제어기는 메시지 전달을 위해 채널 작업을 작성 및 관리합니다이러한 채널 작업은 작업 제어기 내의 처리 풀 안에서 실행됩니다. 풀은 채널 작업이 실행되는 "장소"로 생각할 수 있습니다. 풀은 작업 세트가 풀 외부의 작업과 자원을 놓고 경쟁하지 않고도 작동할 수 있는 컴퓨팅 영역을 제공합니다. job_controller에서 설정된 작업 제한이 해당 풀에만 적용된다는 점에 유의하십시오. 따라서, 예를 들어 job_limit가 10으로 설정된 SMTP_POOL을 정의할 경우 특정 시점에 10개의 tcp_smtp 클라이언트 프로세스만 해당 풀에서 실행될 수 있습니다.

경우에 따라서는 추가 tcp_* 채널을 만드는 것이 필요할 수 있습니다(예: 특히 느린 메일 사이트를 위한 tcp 채널). 이러한 채널은 다른 풀에서 실행하는 것이 좋습니다. 이는 예를 들어, 10개의 다른 tcp_* 채널을 만들어 SMTP_POOL에서 모두 실행할 경우 특정 시점에 각 tcp_* 채널에 대해 하나의 tcp_smtp 클라이언트만 실행될 수 있기 때문입니다(job_limit가 10인 SMTP_POOL을 정의한 경우에 해당하며 메일이 모든 tcp_* 채널을 대상으로 하는지 여부에 따라 달라짐). 시스템의 로드량이 많고 다양한 tcp_* 채널로 나가길 기다리는 메시지를 모든 대기열이 갖고 있다고 가정하면 이는 비효율적입니다. 이러한 경우에는 경합이 발생하지 않도록 추가 tcp_* 채널에 대한 추가 풀을 정의할 수 있습니다.

예를 들어, 다음 tcp_* 채널을 설정한다고 가정해 봅니다.

```
tcp_yahoo smtp mx pool yahoo_pool keyword keyword keyword
tcp-yahoo-daemon

tcp_aol smtp mx keyword keyword keyword pool aol_pool
tcp-aol-daemon

tcp_hotmail smtp mx pool hotmail_pool keyword keyword keyword
tcp-hotmail-daemon
...
tcp_sun smtp mx pool sun_pool keyword keyword keyword
tcp-sun-daemon
```

각각의 새 채널에 대해 10개의 tcp_smtp_client 프로세스를 추가하기 위해 job_controller.cnf 파일에 다음을 추가할 수 있습니다.

```
[POOL=yahoo_pool]
job_limit=10

[POOL=aol_pool]
job_limit=10

[POOL=hotmail_pool]
job_limit=10

 ...
```

```
[POOL=sun_pool]
job_limit=10
```

풀에 대한 자세한 내용은 356 페이지 “12.5.4 채널 실행 작업의 처리 풀”를 참조하십시오.

표 10–6 작업 제어기 구성 파일 옵션

옵션	설명
일반 옵션	**설명**
INTERFACE_ADDRESS=*adapter*	작업 제어기가 바인드해야 하는 IP 주소 인터페이스를 지정합니다. 지정된 값(어댑터)은 ANY, ALL, LOCALHOST 또는 IP 주소 중 하나가 될 수 있습니다. 기본적으로 작업 제어기는 모든 주소에 바인드됩니다(ALL 또는 ANY를 지정하는 것과 동일). INTERFACE_ADDRESS=LOCALHOST를 지정하면 작업 제어기는 로컬 시스템 내의 연결만 수락합니다. 이 경우 작업 제어기에 의해 지원되는 시스템 간 작업이 없이 때문에 정상적인 작업에 영향을 주지 않습니다. 그러나 이 지정은 작업 제어기의 응답 여부를 HA 에이전트가 검사할 수 있는 HA 환경에서는 적합하지 않을 수 있습니다. Messaging Server가 실행 중인 시스템이 HA 환경에 있으며 “내부 네트워크” 어댑터 및 “외부 네트워크” 어댑터가 사용 중이고 높은 포트 번호에 대한 연결을 방화벽에서 차단할 수 있는지 확실하지 않을 경우, “내부 네트워크” 어댑터의 IP 주소를 지정하는 것을 고려해야 합니다.
MAX_MESSAGES=*integer*	작업 제어기는 메시지에 대한 정보를 메모리 내장 구조에서 유지합니다. 대량의 백로그가 작성될 경우에 대비하여 이 구조의 크기를 제한해야 할 수 있습니다. 백로그의 메시지 수가 여기에서 지정된 매개 변수를 초과할 경우 후속 메시지에 대한 정보를 메모리에 저장되지 않습니다. 메일 메시지는 항상 디스크에 기록되므로 손실되지 않지만 작업 제어기가 알고 있는 메시지 수가 이 숫자의 절반으로 줄어들 때까지 메시지 전달이 고려되지 않습니다. 이 시점에서 작업 제어기는 imsimta cache -sync 명령을 모방하는 대기열 디렉토리의 스캔을 수행합니다. 최소값은 10입니다. 기본값은 100000입니다.
SECRET=*file_spec*	작업 제어기로 보내진 요청을 보호하는 데 사용되는 공유 비밀입니다.
SYNCH_TIME=*time_spec*	작업 제어기는 가끔씩 디스크상의 대기열 파일을 스캔하여 누락된 파일을 검사합니다. 기본적으로 이 작업은 작업 제어기가 시작되고 4시간 후부터 4시간마다 수행됩니다. *time_spec*의 형식은 *HH*:*MM* /*hh*: *mm* 또는 /*hh*: *mm*입니다. 변수 *hh*.*mm*은 이벤트 사이의 간격을 나타내는 시간(*h*)과 분(*m*)을 나타냅니다. 변수 *HH*:*MM*은 이벤트가 처음 발생해야 하는 시간입니다. 예를 들어, 15:45/7:15를 지정하면 이벤트가 15:45에 처음 시작되어 그 후로 7시간 15분마다 발생합니다.
TCP_PORT=*integer*	작업 제어기가 요청 패킷을 수신해야 하는 TCP 포트를 지정합니다. 기본값이 시스템의 다른 TCP 응용 프로그램과 충돌하지 않을 경우 이 옵션을 변경하지 않습니다. 이 옵션을 변경할 경우 MTA 조정 파일 *msg-svr-base*/config/imta_tailor에서 해당 IMTA_JBC_SERVICE 옵션을 변경하여 일치시켜야 합니다. TCP_PORT 옵션은 전역적으로 적용되며 [CHANNEL] 또는 [POOL] 섹션에 있을 경우 무시됩니다.

표 10–6 작업 제어기 구성 파일 옵션 (계속)

옵션	설명
풀 옵션	**설명**
JOB_LIMIT=*integer*	풀이 동시에(병렬로) 사용할 수 있는 최대 프로세스 수를 지정합니다. JOB_LIMIT는 각 풀에 개별적으로 적용되므로 작업의 최대 총 개수는 모든 풀의 JOB_LIMIT 매개 변수를 더한 값입니다. 이 옵션은 기본적으로 섹션 외부에 설정된 경우 JOB_LIMIT를 지정하지 않는 모든 [POOL] 섹션에 사용됩니다. [CHANNEL] 섹션 안에 있을 경우 이 옵션은 무시됩니다.
채널 옵션	**설명**
MASTER_COMMAND=*file_spec*	채널을 실행하고 해당 채널에 대한 아웃바운드 메시지를 대기열에서 제외시키기 위해 작업 제어기가 작성한 UNIX 시스템 프로세스에 의해 실행되는 명령의 전체 경로를 지정합니다. 이 옵션은 기본적으로 섹션 외부에 설정된 경우 MASTER_COMMAND를 지정하지 않는 모든 [CHANNEL] 섹션에 사용됩니다. [POOL] 섹션 안에 있을 경우 이 옵션은 무시됩니다.
MAX_LIFE_AGE=*integer*	채널 마스터 작업의 최대 수명 시간을 초 단위로 지정합니다. 채널에 이 매개 변수가 지정되지 않은 경우 전역 기본값이 사용됩니다. 기본값이 지정되지 않은 경우 1800(30분)이 사용됩니다.
MAX_LIFE_CONNS=*integer*	최대 수명 매개 변수 외에 메시지가 있는지 작업 제어기에 물어볼 수 있는 횟수에 따라 채널 마스터 작업의 기대 수명이 제한됩니다. 채널에 이 매개 변수가 지정되지 않은 경우 전역 기본값이 사용됩니다. 기본값이 지정되지 않은 경우 300이 사용됩니다.
SLAVE_COMMAND=*file_spec*	채널을 실행하고 해당 채널에 대한 모든 인바운드 메시지를 폴링하기 위해 작업 제어기가 작성한 UNIX 시스템 프로세스에 의해 실행되는 명령의 전체 경로를 지정합니다. 대부분의 MTA 채널에는 SLAVE_COMMAND가 없습니다. 이러한 경우 예약된 값 NULL을 지정해야 합니다. 이 옵션은 기본적으로 섹션 외부에 설정된 경우 SLAVE_COMMAND를 지정하지 않는 모든 [CHANNEL] 섹션에 사용됩니다. [POOL] 섹션 안에 있을 경우 이 옵션은 무시됩니다.

10.5 별칭

MTA는 실제 사용자와 반드시 일치할 필요가 없는 로컬 시스템과 연관된 메일함 이름, 즉 **별칭**을 지원하는 기능을 제공합니다. 별칭은 메일링 목록 생성, 메일 전달 및 아이디에 대한 동의어 제공 등에 유용합니다. 별칭 지정이 처리되는 방법에 대한 자세한 내용은 186 페이지 “9.1.2.2 $V 메타 문자”를 참조하십시오.

aliases 파일 또는 별칭 데이터베이스에 정의되어 있는 이전 스타일의 메일링 목록은 이제 비지정 [capture] 매개 변수를 갖습니다. [capture] 매개 변수가 사용되는 경우, 이 매개 변수는 LDAP의 사용자나 그룹에 적용되는 LDAP_CAPTURE 속성에서 지정한 캡처 주소와 같은 의미의 캡처 주소를 지정합니다.

[envelope_from] 비지정 별칭 매개 변수, 지정 별칭 매개 변수의 오류 또는 mgrpErrorsTo LDAP 속성 값으로 제공된 값 "/"는 이제 메일링 목록의 의미를 유지하면서 받는 메시지의

주소에서 봉투의 From: 주소 사용으로 되돌리기 위한 요청으로 해석됩니다. 이는 원래의 보낸 사람에게 모든 형태의 목록 오류를 보고하는 메일링 목록을 설정하는 경우 유용합니다.

10.5.1 별칭 데이터베이스

별칭 데이터베이스 사용은 권장되지 않습니다. 대신, imsimta reload 명령을 사용하여 동적으로 재로드할 수 있는 aliases 파일을 사용합니다.

MTA는 디렉토리의 정보를 사용하여 별칭 데이터베이스를 만듭니다. 일반 별칭 파일이 참조될 때마다 별칭 데이터베이스가 한 번씩 참조됩니다. 그러나 별칭 데이터베이스는 일반 별칭 파일이 사용되기 전에 검사됩니다. 실제로 별칭 데이터베이스는 별칭 파일을 사용하기 전에 호출되는 일종의 주소 재작성기의 역할을 수행합니다.

주 – 데이터베이스 자체 형식은 개인적입니다. 데이터베이스를 직접 편집하려고 해서는 안 되며 필요한 모든 사항을 디렉토리에서 변경합니다.

10.5.2 별칭 파일

aliases 파일은 디렉토리에 설정되지 않은 별칭을 설정하는 데 사용됩니다. 특히 포스트마스터 별칭을 좋은 예로 들 수 있습니다. 이 파일에 설정된 별칭은 디렉토리에 동일한 별칭이 존재할 경우 무시됩니다. imsimta reload 명령을 실행하거나 MTA를 다시 시작하여 변경 사항을 활성화할 수 있습니다. 느낌표로 시작되는 모든 행은 주석으로 간주되어 무시됩니다. 또한 빈 행도 무시됩니다.

주 – Messaging Server는 주소 역방향 데이터베이스 및 특수한 매핑 테이블과 같은 주소 조작을 위한 다른 기능을 제공합니다. 그러나 최상의 성능을 위해서는 주소 조작을 수행할 수 있을 때마다 다시 쓰기 규칙을 사용하는 것이 좋습니다. 11 장을 참조하십시오.

이 파일의 물리적 행은 1024자로 제한됩니다. 백슬래시(\) 연결 문자를 사용하여 논리적 행을 여러 물리적 행으로 분할할 수 있습니다.

이 파일의 형식은 다음과 같습니다.

user@*domain*: *address* **(호스트된 도메인의 사용자인 경우)**

user@*domain*: *address* **(호스트되지 않은 도메인의 사용자인 경우. 예**: *default-domain)*

예를 들면 다음과 같습니다.

```
! A /var/mail/ user
inetmail@siroe.com: inetmail@native-daemon
```

```
! A message store user
ms_testuser@siroe.com: mstestuser@ims-ms-daemon
```

10.5.3 별칭 파일에 다른 파일 포함

주 aliases 파일에 다른 파일을 포함할 수 있습니다. 다음 형식의 행은 file-spec 파일을 읽도록 MTA에 지시합니다.

```
<file-spec
```

파일 지정은 완전한 파일 경로 지정이어야 하며 파일은 주 aliases 파일과 동일한 보호를 가져야 합니다. 예를 들어, 세계 공용이어야 합니다.

포함 파일의 내용은 해당 참조 지점에서 aliases 파일에 삽입됩니다. 포함 파일에 대한 참조를 파일의 실제 내용으로 대체하여 동일한 결과를 얻을 수 있습니다. 포함 파일의 형식은 주 aliases 파일 자체의 형식과 같습니다. 실제로 포함 파일 자체에 다른 파일이 포함될 수 있습니다. 최대 세 개 수준까지의 포함 파일 중첩이 허용됩니다.

10.6 명령줄 유틸리티

Messaging Server는 MTA에 대한 다양한 유지 관리, 테스트 및 관리 작업을 수행할 수 있는 여러 명령줄 유틸리티를 제공합니다. 예를 들어, imsimta cnbuild 명령을 사용하여 MTA 구성, 별칭, 매핑, 보안, 시스템 전체 필터 및 옵션 파일을 컴파일할 수 있습니다. MTA 명령줄 유틸리티에 대한 자세한 내용은 **Sun Java System Messaging Server 6.3 Administration Reference**를 참조하십시오.

10.7 SMTP 보안 및 액세스 제어

SMTP 보안 및 액세스 제어에 대한 자세한 내용은 18 장을 참조하십시오.

10.8 로그 파일

모든 MTA 특정 로그 파일은 로그 디렉토리(*msg-svr-base*/log)에 저장됩니다. 이 디렉토리는 MTA를 통과하는 메시지 트래픽을 설명하는 로그 파일과 특정 마스터 또는 슬레이브 프로그램에 대한 정보를 설명하는 로그 파일을 포함합니다.

MTA 로그 파일에 대한 자세한 내용은 25 장을 참조하십시오.

10.9 주소를 내부 형식에서 공개 형식으로 변환

주소 역방향 텍스트 데이터베이스(**역방향 텍스트 데이터베이스**라고도 함)와 REVERSE 매핑 테이블을 사용하여 주소를 내부 형식에서 공개 광고 형식으로 변환할 수 있습니다. 예를 들어 uid@mailhost.siroe.com은 siroe.com 도메인 내에서 유효한 주소일 수 있지만 외부에 공개하기에는 적합하지 않을 수 있습니다. 이 경우에는 내부 주소 대신에 firstname.lastname@siroe.com과 같은 공개 주소를 사용하는 것이 좋을 수도 있습니다.

Messaging Server는 aliases 파일 및 특수한 매핑 테이블과 같은 주소 조작을 위한 다른 기능을 제공합니다. 그러나 최상의 성능을 위해서는 주소 조작을 수행할 수 있을 때마다 다시 쓰기 규칙을 사용하는 것이 좋습니다. 11 장을 참조하십시오.

이 절은 다음과 같은 하위 절로 구성되어 있습니다.

역방향 텍스트 데이터베이스에서 각 사용자의 공개 주소는 디렉토리에 있는 사용자 항목의 mail 속성에 의해 지정됩니다.

역방향 텍스트 데이터베이스는 유효한 주소와 공개 주소 간의 매핑을 포함합니다. 자세한 내용은 240 페이지 “10.9.1 MTA 텍스트 데이터베이스”를 참조하십시오.

데이터베이스에서 주소가 발견될 경우 데이터베이스의 해당 오른쪽 부분이 주소로 대체됩니다. 주소가 발견되지 않을 경우 mappings 파일에서 REVERSE라는 매핑 테이블을 찾으려는 시도가 수행됩니다. 테이블이 존재하지 않거나 테이블의 항목이 일치하지 않을 경우 대체가 수행되지 않으며 다시 쓰기가 정상적으로 종료됩니다.

mappings 파일에서 REVERSE 매핑 테이블이 발견되고 주소가 매핑 항목과 일치할 경우 결과 문자열이 주소를 대체합니다(항목에서 $Y를 지정한 경우). $N이 지정된 경우에는 매핑 결과가 무시됩니다. 매핑 항목에서 $Y 외에 $D가 지정된 경우 결과 문자열은 역방향 데이터베이스에서 한 번 더 실행됩니다. 여기에서 일치하는 항목이 발생하면 데이터베이스의 템플리트가 매핑 결과(즉, 주소)를 대체합니다. 아래에는 일반 REVERSE 매핑 테이블 항목(즉, 모든 채널에 적용되는 항목)의 형식이 나와 있습니다. 플래그는 새 주소의 앞이나 끝에 올 수 있다는 점을 유의하십시오.

```
REVERSE

   OldAddress        $Y[Flags]NewAddress
```

아래에는 *channel-specific* 항목(즉, 특정 채널을 통과하는 메시지에서만 발생하는 매핑)의 형식이 나와 있습니다. 채널별 항목이 작동하려면 option.dat에서 use_reverse_database를 13으로 설정해야 한다는 점을 유의하십시오.

```
REVERSE

   source-channel|destination-channel|OldAddress  $Y[Flags]NewAddresS
```

표 10–7은 REVERSE 매핑 테이블 플래그를 보여 줍니다.

표 10–7 REVERSE 매핑 테이블 플래그

플래그	설명
$Y	출력을 새 주소로 사용합니다.
$N	주소가 변경되지 않고 그대로 유지됩니다.
$D	역방향 데이터베이스를 통해 출력을 실행합니다.
$A	역방향 데이터베이스 항목으로 패턴을 추가합니다.
$F	정방향 데이터베이스 항목으로 패턴을 추가합니다.
플래그 비교	**설명**
$:B	헤더(본문) 주소만 일치합니다.
$:E	봉투 주소만 일치합니다.
$:F	정방향 지정 주소만 일치합니다.
$:R	역방향 지정 주소만 일치합니다.
$:I	메시지 아이디만 일치합니다.

10.9.1 MTA 텍스트 데이터베이스

MTA에서 sleepycat 데이터베이스를 사용하면 Messaging Server 배포가 불안정해지기 때문에 사용되지 않습니다. (sleepycat이 빠른 시일 내에 제거되지는 않습니다.)따라서 대신 역방향, 정방향 및 일반 데이터베이스로 MTA 텍스트 데이터베이스를 사용해야 합니다.

텍스트 데이터베이스 설정 방법

1. 데이터가 포함된 텍스트 파일을 준비합니다.

   여기에는 `imsimta crdb`에서 사용하는 것과 같은 형식을 사용합니다. 행 하나에 한 항목을 입력하고 두 필드는 하나 이상의 공백으로 구분합니다. 파일 이름은 `imta_tailor` 에서 `IMTA_GENERAL_DATA`, `IMTA_REVERSE_DATA` 및 `IMTA_FORWARD_DATA` 옵션으로 지정되며 이들 옵션은 일반적으로 각각 *msg-svr-base*`/config/`의 `IMTA_TABLE:general.txt`, `IMTA_TABLE:reverse.txt` 및 `IMTA_TABLE:forward.txt`를 가리킵니다.

general.txt - 일반 데이터베이스 reverse.txt - 역방향 데이터베이스 forward.txt - 정방향 데이터베이스

2. USE_TEXT_DATABASES 옵션에서 적절한 비트를 설정합니다.

   비트 0(값 1) - 일반 데이터베이스에 텍스트 파일 사용 비트 1(값 2) - 역방향 데이터베이스에 텍스트 파일 사용 비트 2(값 4) - 정방향 데이터베이스에 텍스트 파일 사용

3. 원하는 데이터베이스를 활성화하기 위해 필요한 추가 옵션을 모두 설정합니다.

   예를 들어, USE_REVERSE_DATABASE, USE_FORWARD_DATABASE 등을 설정할 수 있습니다.

4. imsimta cnbuild를 실행합니다.
5. imsimta reload를 실행합니다.

USE_TEXT_DATABASES가 적절하지 않은 유일한 경우는 데이터가 심하게 동적일 때입니다. 그런 경우에는 기본 제공되는 데이터베이스 지원에 의존하지 않고 직접 MTA 플러그인을 작성하는 것이 더 좋을 수도 있습니다.

텍스트 데이터베이스가 적절하지 않은데 crdb(Sleepycat) 데이터베이스 지원을 사용하려는 경우에는 데이터베이스 사용 스타일을 구성하고 프로세스를 적절하게 업데이트하여 재컴파일, 재로드 또는 재시작을 수행하지 않고 imsimta crdb 또는 imsimta db로 데이터베이스를 업데이트할 수 있습니다. 하지만 이 작업이 제대로 적용되려면 imsimta crdb를 사용하여 기존 항목을 추가 또는 업데이트만 할 수 있는 상황이거나 데이터가 일련의 추가/삭제/변경 작업으로 구성되어 있어야 합니다. 데이터를 이 방식으로 구성하지 않으면(일반적인 경우) 업데이트를 수행할 때 전체 데이터베이스를 대체해야 하기 때문에 텍스트 데이터베이스를 사용하는 것이 더 좋습니다.

10.9.2 주소 역방향 제어 설정

reverse 및 noreverse 채널 키워드와 MTA 옵션 USE_REVERSE_DATABASE 및 REVERSE_ENVELOPE을 사용하여 주소 역방향을 적용할 시기와 방법에 대한 세부 사항을 제어합니다. 기본적으로 주소 역방향 작업은 단지 역방향 지정 주소가 아니라 모든 주소에 적용됩니다.

주소 역방향은 REVERSE_ENVELOPE 시스템 옵션 값(기본값: 1-on, 0-off)의 설정에 따라서 활성화 또는 비활성화됩니다.

대상 채널의 noreverse 는 메시지의 주소에 주소 역방향이 적용되지 않도록 지정합니다. reverse는 주소 역방향을 적용하도록 지정합니다. 자세한 내용은 366 페이지 “12.6.9 역방향 데이터베이스의 채널별 사용”을 참조하십시오.

USE_REVERSE_DATABASE는 MTA가 주소 역방향 데이터베이스와 REVERSE 매핑을 대체 주소의 소스로 사용하는지 여부를 제어합니다. 값 0은 채널에서 주소 역방향이 사용되지 않는다는 것을 의미합니다. 기본값인 5는 단지 역방향 지정 주소가 아니라 모든 주소(MTA 주소 다시 쓰기 프로세스에 의해 재작성된 후)에 주소 역방향을 적용하도록

지정합니다. 값 13은 역방향 지정 주소만이 아니라 reverse 채널 키워드를 가진 주소(MTA 주소 다시 쓰기 프로세스에 의해 재작성된 후)에 주소 역방향을 적용하도록 지정합니다. 더 세부적인 주소 역방향 작업은 USE_REVERSE_DATABASE 옵션의 비트 값을 설정하여 지정할 수 있습니다. 자세한 내용은 **Sun Java System Messaging Server 6.3 Administration Reference** 의 "Option File Format and Available Options"를 참조하십시오.

REVERSE_ENVELOPE 옵션은 주소 역방향이 메시지 헤더 주소뿐만 아니라 봉투의 From 주소에 적용되는지 여부를 제어합니다.

구체적인 효과에 대해서는 Sun Java System Messaging Server Administration Reference에서 이러한 옵션과 키워드에 대한 자세한 설명을 참조하십시오.

10.9.2.1 일반 역방향 매핑 예

다음 예는 일반 REVERSE 매핑을 보여 줍니다. 여기서는 siroe.com의 내부 주소가 user@mailhost.siroe.com 형식을 갖는 것으로 가정합니다. 그러나 아이디 공간은 user@host1.siroe.com 및 user@host2.siroe.com이 siroe.com의 모든 호스트에 대해 동일한 사용자를 지정하는 것으로 간주됩니다. 다음 REVERSE 매핑은 주소 역방향 데이터베이스와 함께 사용될 수 있습니다.

```
REVERSE

   *@*.siroe.com        $0@siroe.com$Y$D
```

이 예에서 *name*@*anyhost*.siroe.com 형식의 주소는 *name*@siroe.com으로 변경됩니다. $D 메타 문자는 주소 역방향 데이터베이스를 참조하도록 합니다. 주소 역방향 텍스트 데이터베이스는 다음 형식의 항목을 포함해야 합니다.

```
user@mailhost.siroe.com     first.last@siroe.com
```

10.9.2.2 채널별 역방향 매핑 예

기본적으로 주소 역방향 데이터베이스는 라우팅 가능성 범위가 메일 서버 도메인으로 설정된 경우에 사용됩니다. 채널별 REVERSE 매핑 테이블 항목의 예는 다음과 같습니다.

```
REVERSE

   tcp_*|tcp_local|binky@macho.siroe.com    $D$YRebecca.Woods@siroe.com
```

이 항목은 tcp_local의 대상 채널에서 나가는 tcp_* 소스 채널을 가진 모든 메일에 대해 binky@macho.siroe.com 형식의 주소를 Rebecca.Woods@siroe.com으로 변경하도록 MTA에 지시합니다.

주 – 채널별 역방향 매핑을 사용하려면 option.dat에서 USE_REVERSE_DATABASE 옵션을 13으로 설정해야 합니다(기본값=5)

10.9.3 정방향 조회 테이블 및 FORWARD 주소 매핑

주소 역방향은 봉투의 To: 주소에 적용되지 않습니다. 이는 메일이 메일 시스템을 통과할 때 봉투의 To: 주소가 계속해서 재작성 및 수정된다는 확실한 이유가 있기 때문입니다. 라우팅의 전체 목표는 봉투의 To: 주소를 점차적으로 시스템 및 메일함별 형식으로 변환하는 것입니다. 주소 역방향의 정형화 기능은 전반적으로 봉투의 To: 주소에 적용되지

어떠한 경우든 MTA에서 풍부한 기능을 사용하여 봉투의 To: 주소에 적용되지 별칭 파일, 별칭 데이터베이스 및 일반 조회 테이블이 바로 이 기능을 정확하게 제공합니다.

MTA에서는 또한 패턴 기반 전달, 소스별 전달 또는 주소 자동 등록과 같은 특수한 전달 목적에 사용되는 정방향 조회 테이블과 FORWARD 매핑을 사용할 수 있습니다. 정방향 조회 테이블과 FORWARD 매핑은 주로 특수한 종류의 주소 전달에 사용하도록 되어 있다는 점을 유의하십시오. 즉, 대부분의 주소 전달은 MTA의 다른 전달 기법 중 하나를 사용할 때 더 효율적으로 수행됩니다.

봉투의 To: 주소에 대한 다양한 대체 기법은 역방향 조회 테이블과 동일한 기능을 제공하지만 역방향 매핑과 동일한 기능에 대해서는 아직 언급된 것이 없습니다. 경우에 따라서는 봉투의 To: 주소에 대한 매핑 기능이 유용하고 바람직할 수 있습니다.

10.9.3.1 FORWARD 매핑 테이블

FORWARD 매핑 테이블은 패턴을 기반으로 하는 전달 기능을 제공하며 소스 고유 전달을 위한 기법도 제공합니다. FORWARD 매핑 테이블이 매핑 파일에 있으면 각 봉투의 To: 지시합니다. 이 매핑이 존재하지 않거나 매핑의 항목이 일치하지 않을 경우 변경이 수행되지 않습니다.

주소가 매핑 항목과 일치할 경우 매핑 결과가 테스트됩니다. 항목에 $Y가 지정될 경우 결과 문자열이 봉투의 To: 주소를 대체하며 $N이 지정될 경우 매핑 결과를 무시합니다. 추가 플래그 목록은 표 10–8을 참조하십시오.

표 10–8 FORWARD 출력 매핑 테이블 플래그 설명

플래그	설명
$D	다시 쓰기 프로세스를 통해 출력을 다시 실행합니다.
$G	정방향 조회 테이블을 통해 출력을 실행합니다(정방향 조회 테이블이 사용 가능하게 된 경우).

표 10–8 FORWARD 출력 매핑 테이블 플래그 설명 *(계속)*

플래그	설명
$H	추가 정방향 조회 테이블이나 FORWARD 매핑 조회를 사용 불가능하게 합니다.
$I	메시지를 .HELD 파일로 보관합니다.
$N	주소가 변경되지 않고 그대로 유지됩니다.
$Y	출력을 새 주소로 사용합니다.

FORWARD 매핑(있을 경우)은 정방향 조회 테이블이 조회되기 전에 참조됩니다. FORWARD 매핑이 일치하고 플래그 $G가 있을 경우 정방향 조회 테이블에 대해 FORWARD 매핑의 결과가 검사됩니다(USE_FORWARD_DATABASE의 적절한 설정을 통해 정방향 조회 테이블이 사용 가능하게 된 경우). 채널 고유 정방향 조회 테이블 사용이 지정된 경우 정방향 조회 테이블에서 조회하기 전에 FORWARD 매핑의 결과에 소스 주소와 소스 채널이 접두어로 추가된다는 점을 유의하십시오. 일치하는 FORWARD 매핑 항목에 $D가 지정된 경우 FORWARD 매핑의 결과와 선택적 정방향 테이블 조회가 MTA 주소 다시 쓰기 프로세스를 통해 다시 실행됩니다. 일치하는 FORWARD 매핑 항목에 $H가 지정된 경우 $D 사용으로 인해 발생하는 후속 주소 다시 쓰기 프로세스 동안에 추가 FORWARD 매핑이나 데이터베이스 조회가 수행되지 않습니다.

다음 입력 플래그를 FORWARD 매핑에서 사용할 수 있습니다. 이전에는 여러 *_ACCESS 매핑에서만 사용 가능했습니다.

표 10–9 FORWARD 입력 매핑 테이블 플래그 설명

플래그	설명
$A	연결 인증에 SASL가 사용됩니다.
$D	이 수신자에 대해 NOTIFY=DELAYS가 활성 상태입니다.
$E	받는 연결에서 ESMTP/EHLO를 사용합니다.
$F	이 수신자에 대해 NOTIFY=FAILURES가 활성 상태입니다.
$L	받는 연결에서 LMTP/LHLO를 사용합니다.
$S	이 수신자에 대해 NOTIFY=SUCCESSES가 활성 상태입니다.
$T	연결 보호에 SSL/TLS가 사용됩니다.

아래 예는 복잡한 REVERSE 및 FORWARD 매핑의 사용을 보여 줍니다. 여기에서는 mr_local 채널과 연관된 am.sigurd.innosoft.com이라는 시스템 또는 의사 도메인이 일반적인 형식의 RFC 822 주소를 생성한다고 가정합니다.

```
"lastname, firstname"@am.sigurd.example.com
```

또는

```
"lastname,firstname"@am.sigurd.example.com
```

이러한 주소는 완전히 유효하지만 RFC 822 구문 규칙을 완벽하게 따르지 않는 전자 메일 프로그램(예: 인용된 주소를 제대로 처리하지 않은 전자 메일 프로그램)에서는 흔히 혼동을 일으킵니다. 결과적으로 인용이 필요하지 않는 주속 형식이 더 많은 전자 메일 프로그램에서 작동합니다. 이러한 형식 중 하나는 다음과 같습니다.

```
firstname.lastname@am.sigurd.example.com
```

복잡한 FORWARD 및 REVERSE 매핑 예

REVERSE

```
 *|mr_local|"*,$ *"@am.sigurd.example.com $Y"$1,$ $2"@am.sigurd.example.com
 *|mr_local|"*,*"@am.sigurd.example.com   $Y"$1,$ $2"@am.sigurd.example.com
 *|*|"*,$ *"@am.sigurd.example.com        $Y$3.$2@am.sigurd.example.com
 *|*|"*,*"@am.sigurd.example.com          $Y$3.$2@am.sigurd.example.com
 *|mr_local|*.*@am.sigurd.example.com     $Y"$2,$ $1"@am.sigurd.example.com
 *|*|*.*@am.sigurd.example.com            $Y$2.$3@am.sigurd.example.com
```

FORWARD

```
 "*,$ *"@am.sigurd.example.com            $Y"$0,$ $1"@am.sigurd.example.com
 "*,*"@am.sigurd.example.com              $Y"$0,$ $1"@am.sigurd.example.com
 *.*@am.sigurd.example.com                $Y"$1,$ $0"@am.sigurd.example.com
```

따라서 위 예에 나온 샘플 매핑 테이블은 (1) 이러한 세 개의 주소 형식을 모두 사용할 수 있도록 허용하고, (2) 원래 형식의 주소만 mr_local 채널에 제공하고 필요에 따라 형식을 변환하며, (3) 인용되지 않은 새 형식의 주소만 다른 모든 채널에 제공하고 필요에 따라 형식을 변환하는 세 가지 목적을 가집니다. 위의 REVERSE 매핑에서는 MTA 옵션 USE_REVERSE_DATABASE에 비트 3이 설정된 것으로 가정합니다.

10.9.3.2 정방향 조회 테이블

주소 전달이 자동 등록되거나 소스별로 고유해야 할 경우 정방향 조회 테이블을 사용할 수 있습니다. 간단한 메시지 전달에는 일반적으로 정방향 조회 테이블을 사용하는 것이 적합하지 않으며 aliases 파일 또는 별칭 조회 테이블이 이러한 전달을 수행하는 데 더 효율적인 방법이라는 점에 유의하십시오. 기본적으로 정방향 조회 테이블은 전혀 사용되지 않으므로 USE_FORWARD_DATABASE 옵션을 통해 명시적으로 활성화해야 합니다. 정방향 테이블 조회는 주소 다시 쓰기 이후, 별칭 확장이 수행된 후, 그리고 임의의 FORWARD 매핑이 검사된 후에 수행됩니다. 정방향 테이블 조회에 성공할 경우 MTA 주소 다시 쓰기 프로세스를 통해 대체된 결과 주소가 다시 실행됩니다.

정방향 조회 테이블에는 메모리 내장 해시 테이블과 기본 텍스트 데이터베이스의 두 가지 기법을 사용할 수 있습니다. 테이블의 크기가 너무 크지 않을 경우 해시 테이블이 권장됩니다(1,000은 너무 큰 것이 아니지만 100,000은 너무 크다고 할 수 있습니다). 해시 테이블은 use_text_databaseS 옵션에서 비트 2(값 4)를 설정하고

use_forward_database를 설정하여 활성화합니다. 해시 테이블은 *msg-svr-base*/configure/forward.txt에서 읽어오고 구성의 재로드 가능 부분으로 컴파일되며 imsimta reload 명령에 의해 강제로 활성 MTA 프로세스로 재로드될 수 있습니다.

소스 텍스트 파일의 형식은 기본적으로 다음과 같습니다.

```
user1@domain1 changedmailbox1@changeddomain1
user2@domain2 changedmailbox@changeddomain2
```

하지만 USE_FORWARD_DATABASE 옵션의 비트 2를 설정하여 정방향 텍스트 데이터베이스의 소스별 사용을 활성화한 경우 소스 텍스트 파일 형식은 다음과 같습니다.

```
source-channel|source-address|original-address changed-address
```

예를 들어, 다음과 같은 항목은

```
tcp_limited|bob@blue.com|helen@red.com  "helen of troy"@siroe.com
```

bob@blue.com에서 메일이 오고 대기열을 넣는 채널이 tcp_limited인 경우에만 To: 주소 helen@red.com을 "helen of troy"@siroe.com에 매핑합니다.

정방향 텍스트 데이터베이스에 대한 자세한 내용은 240 페이지 "10.9.1 MTA 텍스트 데이터베이스"를 참조하십시오.

10.10 전달 상태 알림 메일 제어

전달전달 상태 알림 또는 **상태 알림**은 MTA가 보낸 사람이나 포스트마스터(선택 사항)로 보내는 전자 메일 상태 메일입니다. Messaging Server를 사용하면 내용이나 언어별로 알림 메일을 사용자 정의할 수 있습니다. 또한 각 유형의 전달 상태(예: FAILED, BOUNCED, TIMEDOUT 등)에 대해 다른 메일을 만들 수 있습니다. 이외에도 특정 채널에서 보내지는 메일에 대한 상태 알림을 만들 수 있습니다.

기본적으로 상태 알림은 *msg-svr-base* /config/locale/C 디렉토리에 저장됩니다(*msg-svr-base*/config/imta_tailor 파일의 IMTA_LANG 설정에서 지정됨). 파일 이름은 다음과 같습니다.

return_bounced.txt, return_delivered.txt return_header.opt, return_timedout.txt, return_deferred.txt, return_failed.txt, return_prefix.txt, return_delayed.txt, return_forwarded.txt, return_suffix.txt

*.txt 파일의 메일 텍스트는 한 행당 78자로 제한되어야 합니다. 이러한 파일은 Messaging Server를 업그레이드할 때 덮어쓰게 되므로 이러한 파일을 직접 변경하면 안 됩니다. 이러한 파일을 수정하고 유일한 알림 메일 템플리트 집합(return_*.txt)으로 사용하려면 파일을 새 디렉토리로 복사하여 해당 위치에서 편집합니다. 그런 다음 이러한 템플리트를 포함하는 새 디렉토리를 가리키도록 imta_tailor 파일에서

IMTA_LANG 옵션을 설정합니다. 각 언어에 대한 집합이 필요한 경우처럼 여러 집합의 알림 파일이 필요한 경우 NOTIFICATION_LANGUAGE 매핑 테이블을 설정해야 합니다.

이 절은 다음과 같은 하위 절로 구성되어 있습니다.

10.10.1 상태 알림 생성 및 수정

단일 알림 메시지는 세 파일 집합 return_prefix.txt + return_*ActionStatus*.txt + return_suffix.txt로부터 생성됩니다.

알림을 사용자 정의하거나 현지화하려면 각 로켈 및/또는 사용자 정의에 대해 전체 return_*.txt 파일을 만들어서 별도의 디렉토리에 저장합니다. 예를 들어 프랑스어 알림 파일을 한 디렉토리에 저장해 놓고 스페인어 알림 파일을 또 다른 디렉토리에 저장하고 원치 않는 대량 전자 메일 채널에 대한 알림을 다른 디렉토리에 저장할 수 있습니다.

주 – 이 릴리스에는 프랑스어, 독일어, 스페인어 샘플 파일이 포함되어 있습니다. 자신의 고유한 요구에 맞도록 이러한 파일을 수정할 수 있습니다.

일본어와 같은 더블바이트 언어의 경우 텍스트를 일본어로 생성한 다음 해당 텍스트를 ASCII인 것처럼 표시하여 % 문자를 검사합니다. % 문자가 잘못 들어가 있을 경우 이를 %%으로 대체합니다.

상태 알림 메일 집합의 형식과 구조는 다음과 같습니다.

1. return_prefix.txt는 본문의 소개 부분뿐만 아니라 해당하는 헤더 텍스트를 제공합니다. 미국 영어 로켈에 대한 기본값은 다음과 같습니다.

   ```
   Content-type: text/plain; charset=us-asci
   Content-language: EN-US

   This report relates to a message you sent with the following
   header fields: %H
   ```

   미국 ASCII가 아닌 상태 알림 메일의 경우 charset 매개 변수와 Content-Language 헤더 값을 적절하게 변경해야 합니다. 예를 들어, 현지화된 프랑스어 파일의 경우 이러한 값은 ISO-8859-1 및 fr입니다. %H는 표 10–10에 정의된 헤더 대체 시퀀스입니다.
2. return_<*ActionStatus*>.txt는 상태별 텍스트를 포함합니다. *ActionStatus*는 메일의 MTA 상태 유형을 나타냅니다. 예를 들어, return_failed.txt의 기본 텍스트는 다음과 같습니다.

```
Your message cannot be delivered to the following recipients:%R
```

return_bounced.txt의 기본 텍스트는 다음과 같습니다.

```
Your message is being returned. It was forced to return bythe postmaster.

The recipient list for this message was:%R
```

3. return_suffix.txt는 결과 텍스트를 포함합니다. 기본적으로 이 파일은 비어 있습니다.

표 10–10 알림 메일 대체 시퀀스

대체	정의
%H	메일의 헤더로 확장됩니다.
%C	메일을 대기열에 포함했던 단위 수[1]로 확장됩니다.
%L	메시지가 반환되기 전에 대기열에 남아 있었던 메일의 단위 수[1]로 확장됩니다.
%F	메시지가 대기열에 머무르는 것이 허용되는 단위 수[1]로 확장됩니다.
%S [%s]	이전에 확장된 숫자 값이 1이 아닐 경우 문자 S 또는 s로 확장됩니다. 예: 메시지가 대기열에 포함되었던 일 수에 따라 "%C day%s"는 "1day" 또는 "2days"로 확장될 수 있습니다.
%U [%u]	사용 중인 시간 단위 Hour [hour] 또는 Day [day]로 확장됩니다. 예: 메시지가 대기열에 포함되었던 일 또는 시간과 MTA 옵션 RETURN_UNITS에 따라 "%C %U%s"가 "2 days" 또는 "1 hour"로 확장될 수 있습니다. RETURN_UNITS=1(시간)을 설정했으며 사이트에서 현지화된 상태 알림 메일을 사용할 경우 return_delayed.txt 및 return_timedout.txt를 편집하여 영어가 아닌 모든 언어에 대해 "days"를 hours로 대체해야 합니다. 예를 들어, 프랑스어는 jour(s)를 heure(s)로,독일어는 Tag(e)를 Stunde(n)로, 스페인어는 día(s)를 hora(s)로 바꿔야 합니다.
%R	메일 수신자의 목록으로 확장됩니다.
%%	% (텍스트는 문자 세트에 상관 없이 대체 시퀀스에 대해 바이트 단위로 스캔된다는 것에 주의합니다. 더블바이트 문자 세트를 사용하는 중이면 % 기호가 잘못 들어가 있지 않은지 검사합니다.)

[1] 단위는 MTA 옵션 파일의 RETURN_UNITS 옵션에 의해 정의되며 시간 또는 일(기본값)이 될 수 있습니다.

10.10.2 전달 상태 알림 메일 사용자 정의 및 현지화

전달 상태 알림 메일을 현지화하여 여러 다른 언어로 여러 사용자에게 메일을 반환할 수 있습니다. 예를 들어, 프랑스어를 선호하는 사용자에게 프랑스어 알림을 반환할 수 있습니다.

상태 알림 메일을 현지화 또는 사용자 정의하는 것은 다음 두 단계로 구성됩니다.

1. 현지화/사용자 정의된 return_*.txt 메일 파일 집합을 만들어 각 집합을 별도의 디렉토리에 저장합니다. 이에 대해서는 247 페이지 "10.10.1 상태 알림 생성 및 수정"에 설명되어 있습니다.
2. NOTIFICATION_LANGUAGE 매핑 테이블을 설정합니다.

msg-svr-base/config/mappings에 있는 NOTIFICATION_LANGUAGE 매핑 테이블은 **원본 메일**(알림을 보내는 원인이 되는 메일)의 속성(예: 언어, 국가, 도메인 또는 주소)에 따라 현지화되거나 사용자 정의된 알림 메일 파일 집합을 사용하도록 지정합니다.

상태 알림 유형, 소스 채널, 기본 언어, 반송 주소 및 첫 번째 수신자를 결정하기 위해 원래 보낸 사람의 메시지가 구문 분석됩니다. 테이블의 구성 방법에 따라 이러한 속성 중 하나 이상에 기초하여 알림 파일 집합이 선택됩니다.

NOTIFICATION_LANGUAGE 매핑 테이블의 형식은 다음과 같습니다. 인쇄상의 이유로 샘플 항목에서는 줄 바꿈되었지만 실제 항목은 한 행에 표시됩니다.

```
NOTIFICATION_LANGUAGE

 dsn-type-list|source-channel|preferred-language|return-address \
|first-recipient $Idirectory-spec
```

- dsn-type-list는 쉼표로 구분된 전달 상태 알림 유형의 목록입니다. 여러 유형을 지정할 경우 공백 없이 쉼표로 이러한 유형을 구분해야 합니다(공백을 사용할 경우 매핑 테이블 항목의 패턴이 종료됨). 이러한 유형은 다음과 같습니다.
  - failed - 일반 영구 오류 메일(예: 해당 사용자 없음)입니다. return_failed.txt 파일이 사용됩니다.
  - bounced - 수동 "바운스"와 함께 사용되는 알림 메일입니다.포스트마스터에 의해 수행됩니다. return_bounced.txt 파일이 사용됩니다.
  - timedout - MTA가 전달이 허용되는 시간 내에 메일을 전달하지 못했습니다. 메일을 반환하고 있습니다. return_timedout.txt 파일이 사용됩니다.
  - delayed - MTA가 메일을 전달하지 못했지만 계속해서 전달을 시도할 것입니다. return_delayed.txt 파일이 사용됩니다.
  - deferred - "delayed"와 비슷한 전달 실패 알림이지만 MTA가 전달 시도를 계속할 기간이 표시되지 않습니다. return_deferred.txt 파일이 사용됩니다.
  - forwarded - 메일에 대한 전달 수신 확인이 요청되었지만 이러한 수신 확인을 지원하지 않는 시스템에 메시지가 전달되었습니다. return_forwarded.txt 파일이 사용됩니다.
- source-channel은 알림 메일을 생성하는 채널(즉, 메시지가 현재 대기열이 포함된 채널)입니다. 예를 들어, 메시지 저장소의 전달 대기열에 대한 ims-ms, 아웃바운드 SMTP 대기열에 대한 tcp_local 등이 있습니다.
- preferred-language는 처리할 메일(알림이 생성된 메일)이 표현되는 언어입니다. 이 정보의 소스는 우선적으로 accept_language 필드입니다. 해당 필드가 없는 경우 Preferred-language: 헤더 필드와 X-Accept-Language: 헤더 필드가 사용됩니다. 표준 언어 코드 값의 목록은 *msg-svr-base*/config/languages.txt 파일을 참조하십시오.

  이 필드는 비어 있지 않을 경우 Preferred-language: 또는 X-Accept-language: 헤더 행에 지정된 메일의 메일 발송자가 됩니다. 이에 따라 이 필드에 무의미한 문자가 나타날 수 있습니다.

- return-address는 원래 메일의 봉투의 From: **주소**입니다. 이 주소는 알림 메시지가 보내지는 봉투 주소이므로 사용할 언어를 나타내는 표시 기호일 수 있습니다.
- first-recipient는 원래 메일을 보내도록 지정된 봉투의 To: 주소(메시지가 여러 수신자에 대해 실패한 경우 첫 번째 주소)입니다. 예를 들어, "dan@siroe.com으로 보내는 메시지를 발송할 수 없습니다." 알림에서 보고되는 봉투의 To: 주소는 dan@siroe.com입니다.
- directory-spec은 매핑 테이블 검사가 일치할 경우 사용할 return_*.txt 파일을 포함하는 디렉토리입니다. 디렉토리 지정 앞에 $I가 와야 한다는 것에 주의합니다.

  예를 들어, 프랑스어 알림 파일(return_*.txt)을 /lc_messages/table/notify_french/ 디렉토리에 저장하고 스페인어 알림 파일을 /lc_messages/table/notify_spanish/ 디렉토리의 return_*.txt 파일에 저장하는 사이트는 아래와 같은 테이블을 사용할 수 있습니다. 각 항목이 하나 이상의 공백으로 시작해야 하며 항목 사이에 빈 행이 올 수 없다는 것에 주의합니다.

```
NOTIFICATION_LANGUAGE

! Preferred-language: header value specified
!
    *|*|fr|*|*     $I/lc_messages/table/notify_french/
    *|*|es|*|*     $IIMTA_TABLE/notify_spanish/
    *|*|en|*|*     $I/imta/lang/
!
! If no Preferred-language value, then select notification based on the
! country code in the domain name. EX: PF=French Polynesia; BO=Bolivia
!
    *|*|*|*.fr|*   $I/imta/table/notify_french/
    *|*|*|*.fx|*   $I/imta/table/notify_french/
    *|*|*|*.pf|*   $I/imta/table/notify_french/
    *|*|*|*.tf|*   $I/imta/table/notify_french/
    *|*|*|*.ar|*   $I/imta/table/notify_spanish/
    *|*|*|*.bo|*   $I/imta/table/notify_spanish/
    *|*|*|*.cl|*   $I/imta/table/notify_spanish/
    *|*|*|*.co|*   $I/imta/table/notify_spanish/
    *|*|*|*.cr|*   $I/imta/table/notify_spanish/
    *|*|*|*.cu|*   $I/imta/table/notify_spanish/
    *|*|*|*.ec|*   $I/imta/table/notify_spanish/
    *|*|*|*.es|*   $I/imta/table/notify_spanish/
    *|*|*|*.gp|*   $I/imta/table/notify_spanish/
    *|*|*|*.gt|*   $I/imta/table/notify_spanish/
    *|*|*|*.gy|*   $I/imta/table/notify_spanish/
    *|*|*|*.mx|*   $I/imta/table/notify_spanish/
    *|*|*|*.ni|*   $I/imta/table/notify_spanish/
    *|*|*|*.pa|*   $I/imta/table/notify_spanish/
    *|*|*|*.ve|*   $I/imta/table/notify_spanish/
```

주 – 알림 언어 매핑이 사용 가능하도록 기본 `mappings.locale` 파일이 설치 시 함께 제공되어 `mappings` 파일에 포함됩니다. 알림 언어 매핑을 사용 불가능하게 하려면 다음과 같이 포함 줄을 주석 처리합니다.

```
! <IMTA_TABLE:mappings.locale
```

(파일에서 주석을 읽은 다음 자신의 요구에 맞게 적절하게 수정합니다.)

10.10.3 생성된 알림 국제화

DSN(Delivery Status Notification) 및 MDN(Message Disposition Notification) 모두에 두 옵션 파일을 사용할 수 있습니다. 생성된 알림의 국제화를 더 유연하게 수행할 수 있게 하는 이러한 파일은 다음과 같습니다.

```
IMTA_LANG:return_option.dat (DSN)IMTA_LANG:disposition_option.dat (MDN)
```

표 10–11에는 이러한 파일에 사용할 수 있는 옵션이 설명되어 있습니다.

표 10–11 DSN(Delivery Status Notification) 및 MDN(Message Disposition Notification) 옵션

옵션	설명
`DAY`(DSN)	`RETURN_UNITS=0`(기본값)을 설정하는 경우 `%U` 또는 `%u` 대체에 대해 삽입되는 텍스트입니다. 영어의 "Day" 또는 "day" 각각을 대체하는 기본적인 경우와는 달리 `%U` 및 `%u` 간에는 차이점이 없습니다.
`DIAGNOSTIC_CODE`(DSN)	DSN의 첫 번째 부분에서 수신자별 섹션의 구성에 사용되는 "`Diagnostic code:`" 텍스트보다 우선합니다. 이 필드는 DSN의 첫 번째 부분에 사용되는 문자 세트와 같은 문자 세트로 지정해야 합니다.
`HOUR`(DSN)	`RETURN_UNITS=1`을 설정하는 경우 `%U` 또는 `%u` 대체에 대해 삽입되는 텍스트입니다. 영어의 "Hour" 또는 "hour" 각각을 대체하는 기본적인 경우와는 달리 `%U` 및 `%u` 간에는 차이점이 없습니다.
`n.n.n`(DSN)	DSN의 수신자별 부분을 구성할 때 이름이 수신자별 숫자 상태와 일치하는 옵션이 있는지 확인하는 검사가 수행됩니다. 일치하는 항목이 있는 경우 해당 텍스트가 DSN에 삽입됩니다. 또한 위에서 설명한 `REASON` 옵션이 길이가 0인 결과를 생성하는 경우 `REASON` 필드는 삽입되지 않습니다.
`ORIGINAL_ADDRESS`(DSN)	DNS의 첫 번째 부분에서 수신자별 섹션의 구성에 사용되는 "Original address:"텍스트보다 우선합니다. 이 필드는 DSN의 첫 번째 부분에 사용되는 문자 세트와 같은 문자 세트로 지정해야 합니다.
`REASON`(DSN)	DSN의 첫 번째 부분에서 수신자별 섹션의 구성에 사용되는 "`Reason:`" 텍스트보다 우선합니다. 이 필드는 DSN의 첫 번째 부분에 사용되는 문자 세트와 같은 문자 세트로 지정해야 합니다.

표 10–11 DSN(Delivery Status Notification) 및 MDN(Message Disposition Notification) 옵션 (계속)

옵션	설명
RECIPIENT_ADDRESS(DSN)	DNS의 첫 번째 부분에서 수신자별 섹션의 구성에 사용되는 "Recipient address:"텍스트보다 우선합니다. 이 필드는 DSN의 첫 번째 부분에 사용되는 문자 세트와 같은 문자 세트로 지정해야 합니다.
RETURN_PERSONAL(DSN 및 MDN)	From: 필드와 함께 사용되는 개인 이름 필드보다 우선합니다. 이 필드는 RFC 2047로 인코딩되어야 합니다. 이 옵션을 지정하지 않을 경우 RETURN_PERSONAL MTA 옵션에 설정된 값이 사용됩니다.
SUBJECT(DSN 및 MDN)	Subject: 우선합니다. 이 값은 알림에서 자체의 제목 필드를 제공하지 않은 경우에만 사용됩니다. 이 필드는 RFC 2047로 인코딩되어야 합니다. 이 옵션을 사용하지 않고 알림에서 제목을 제공하지 않은 경우 적절한 제목이 생성됩니다.
TEXT_CHARSET (MDN)	MDN의 첫 번째 부분과 제목의 문자 세트 텍스트를 변환해야 합니다. 기본값은 변환을 수행하지 않는 것입니다.

10.10.4 추가 상태 알림 메일 기능

앞의 절에서는 상태 알림 메일을 설정하기 위한 필수 절차에 대해 설명했습니다. 다음 절에서는 추가 기능에 대해 설명합니다.

- 252 페이지 "10.10.4.1 큰 메일의 내용 반환 차단"
- 252 페이지 "10.10.4.2 상태 알림 메일의 포함 헤더에서 미국 ASCII가 아닌 문자 제거"
- 253 페이지 "10.10.4.3 알림 메일 전달 간격 설정"
- 253 페이지 "10.10.4.4 상태 알림 메일에 변경된 주소 포함"
- 254 페이지 "10.10.4.5 포스트마스터에 대해 상태 알림 메일을 전송, 차단 및 지정"

10.10.4.1 큰 메일의 내용 반환 차단

일반적으로 메시지가 바운스 또는 차단되면 메일 내용이 보낸 사람과 알림 메일의 로컬 도메인 포스트마스터에게 반환됩니다. 이것은 매우 큰 여러 메시지가 반환될 경우 자원에 적지 않은 부담이 될 수 있습니다. 일정 크기 이상의 메일 내용이 반환되는 것을 차단하려면 MTA 옵션 파일에서 CONTENT_RETURN_BLOCK_LIMIT 옵션을 설정합니다.

MTA는 봉투 반송 주소와 연결된 블록 제한을 가져오며, 지정된 반송 정책이 없는 상태에서 메시지 크기가 블록 제한을 초과하면 RET=HDRS를 설정합니다. 그러면 대용량 메시지에 대한 배달 실패 보고서가 배달되지 않는 상황이 방지됩니다. 이 변경과 연결된 새로운 옵션이나 설정은 없습니다.

10.10.4.2 상태 알림 메일의 포함 헤더에서 미국 ASCII가 아닌 문자 제거

인터넷 메일 헤더의 원시 형식은 미국 ASCII가 아닌 문자를 허용하지 않습니다. 미국 ASCII가 아닌 문자가 메일 헤더에 사용된 경우 이러한 문자는 RFC 2047에 설명된 "MIME 헤더 인코딩"을 통해 인코딩됩니다. 따라서 전자 메일의 중국어 "Subject" 행은 실제로 다음과 같이 나타납니다.

```
Subject: =?big5?Q?=A4j=AB=AC=A8=B1=AD=B1=B0=D3=F5=A5X=AF=B2?=
```

이 경우 헤더를 표시할 때 인코딩을 제거하는 작업을 전자 메일 클라이언트가 수행합니다.

%H 템플리트가 헤더를 알림 메일의 본문에 복사하므로 인코딩된 헤더 텍스트가 정상적으로 표시됩니다. 그러나 제목의 문자 세트(이 경우에는 "big5")가 return_prefix.txt의 Content-Type 헤더 문자 세트 매개 변수에 있는 문자 세트와 일치할 경우 Messaging Server는 인코딩을 제거합니다. 이러한 문자 세트가 일치하지 않을 경우 Messaging Server는 인코딩을 그대로 둡니다.

10.10.4.3 알림 메일 전달 간격 설정

키워드: notices, nonurgentnotices, normalnotices, urgentnotices

전달할 수 없는 메시지는 보낸 사람에게 반환하기 전에 지정된 시간 동안 주어진 채널 대기열에 보관됩니다. 또한 Messaging Server가 전달을 시도하는 동안에 일련의 상태/경고 메시지가 보낸 사람에게 반환될 수 있습니다. 메일 간의 시간과 간격은 notices, nonurgentnotices, normalnotices 또는 urgentnotices 키워드를 사용하여 지정할 수 있습니다. 예를 들면 다음과 같습니다.

```
notices 1 2 3
```

모든 메일에 대해 1일 및 2일 후에 일시적인 실패 상태 알림 메시지가 보내집니다. 3일 후에도 여전히 전달되지 않은 경우 해당 메시지는 메일 발송자에게 반환됩니다.

```
urgentnotices 2,4,6,8
```

높은 우선 순위를 가진 메일에 대해 2일, 4일 및 6일 후에 일시적인 실패 알림이 보내집니다. 8일 후에도 여전히 전달되지 않은 경우 해당 메시지는 메일 발송자에게 반환됩니다.

MTA 옵션 파일의 RETURN_UNITS 옵션에서는 시간(1) 또는 일(0) 단위를 지정할 수 있다는 것에 주의합니다. 기본값은 (0)일입니다. RETURN_UNITS=1을 설정할 경우 반송 작업을 1시간마다 실행하고 알림을 1시간마다 가져오도록 예약해야 합니다. 반송 작업이 1시간마다 실행되면 또한 1시간마다 mail.log* 파일을 롤오버합니다. mail.log 파일이 1시간마다 롤오버되는 것을 방지하려면 imta.tailor 파일의 IMTA_RETURN_SPLIT_PERIOD 조정 파일 옵션을 24로 설정합니다. 반송 작업 예약은 local.schedule.return_job configutil 매개 변수에서 제어합니다. 하지만 기본적으로 이 명령은 정기적으로 실행됩니다(110 페이지 "4.6.2 미리 정의된 자동 작업" 참조).

notices 키워드가 지정되지 않은 경우 기본값은 로컬 l 채널에 대한 notices 설정을 사용하는 것입니다. 로컬 채널에 대한 설정이 없을 경우 notices 3, 6, 9, 12가 기본값으로 사용됩니다.

10.10.4.4 상태 알림 메일에 변경된 주소 포함

키워드: includefinal, suppressfinal, useintermediate

MTA가 알림 메일(바운스 메일, 전달 수신 확인 메일 등)을 생성할 때 "원래" 형식의 수신자 주소와 MTA에서 사용할 수 있는 변경된 "최종" 형식의 수신자 주소가 모두 존재할 수 있습니다. 알림 메일의 수신자(알림 메시지가 관련된 원래 메일을 보낸 사람)가 대개 원래 형식을 인식하므로 MTA는 항상 원래 형식(존재할 경우)을 알림 메일에 포함합니다.

includefinal 및 suppressfinal 채널 키워드는 MTA가 또한 최종 형식의 주소를 포함하는지 여부를 제어합니다. 내부 메일함 이름을 외부에서 볼 수 없도록 "숨기는" 사이트에서는 최종 형식의 주소를 포함하지 않을 수 있습니다. 이러한 사이트는 원래 "외부" 형식의 주소만 상태 알림 메일에 포함하기를 원할 것입니다. 기본값은 최종 형식의 수신자 주소를 포함하는 includefinal입니다. suppressfinal을 사용하면 원래 주소 형식이 존재할 경우 최종 주소 형식이 상태 알림 메일에 표시되지 않습니다.

useintermediate 키워드는 목록이 확장되었지만 사용자 메일함 이름이 생성되기 전에 만들어진 중간 형식의 주소를 사용합니다. 이 정보를 사용할 수 없는 경우 최종 형식이 사용됩니다.

10.10.4.5 포스트마스터에 대해 상태 알림 메일을 전송, 차단 및 지정

기본적으로 실패 및 경고 상태 알림 메일의 복사본은 오류 반환 및 경고가 빈 Errors-to: 헤더 행 또는 빈 봉투의 From: 주소로 완전히 억제되지 않은 경우 포스트마스터에게 보내집니다. 다음 절과 표 10–12에 설명된 여러 채널 키워드를 사용하면 포스트마스터에 대한 알림 메일 전달을 더 세부적으로 제어할 수 있습니다. 이 절은 다음과 같은 하위 절로 구성되어 있습니다.

반환되는 실패 메일

키워드: sendpost, nosendpost, copysendpost, errsendpost

채널 프로그램은 장기적인 서비스 실패나 잘못된 주소로 인해 메일을 전달하지 못할 수 있습니다. 이 경우 MTA 채널 프로그램은 메일을 전달할 수 없는 이유에 대한 설명과 함께 메일을 보낸 사람에게 반환합니다. 또한 선택적으로 모든 실패 메일의 복사본이 로컬 포스트마스터에게 보내집니다. 이것은 메일 실패를 모니터링하는 데 유용하지만 포스트마스터가 처리해야 하는 과도한 양의 트래픽을 발생시킬 수 있습니다. 표 10–12를 참조하십시오.

경고 메일

키워드:warnpost, nowarnpost, copywarnpost, errwarnpost

메일을 반환하는 것 외에도 MTA는 전달되지 않은 메일에 대한 자세한 경고를 보낼 수 있습니다. 경우에 따라 채널 프로그램이 실패한 전달 시도 이후에 경고 메일을 생성할 수 있지만 이러한 경고는 일반적으로 notices 채널 키워드 설정에 기초한 시간 초과로 인해 발생합니다. 경고 메시지는 무엇이 잘못되었는지와 전달 시도가 얼마나 오랫동안 계속되었는지에 대한 설명을 포함합니다. 또한 대부분의 경우 경고 메시지는 문제가 된 메일의 헤더와 처음 몇 개의 행을 포함합니다.

선택적으로 모든 경고 메일의 복사본을 로컬 포스트마스터에게 보낼 수 있습니다. 이것은 포스트마스터가 처리해야 하는 많은 양의 트래픽을 발생시키지만 다양한 대기열의 상태를 모니터하는 데 유용할 수 있습니다. 경고 메일을 포스트마스터에게 보내는 것을 제어하기 위해 `warnpost`, `copywarnpost`, `errwarnpost` 및 `nowarnpost` 키워드가 사용됩니다. (표 10–12 참조).

빈 봉투 반송 주소

키워드: `returnenvelope`

`returnenvelope` 키워드는 비트 플래그 집합으로 해석되는 단일 정수 값을 가집니다. 비트 0(값 = 1)은 MTA에 의해 생성된 반송 알림이 빈 봉투 주소 또는 로컬 포스트마스터의 주소로 작성되는지 여부를 제어합니다. 이 비트를 설정하면 로컬 마스터 주소가 사용되고 이 비트를 지우면 빈 주소가 사용됩니다.

주 – RFC 1123에는 빈 주소를 사용하도록 명시되어 있지만 일부 시스템은 빈 봉투의 From: 주소를 적절하게 처리하지 않으므로 이 옵션을 사용하는 것이 필요할 수 있습니다.

비트 1(값 = 2)은 MTA가 모든 빈 봉투 주소를 로컬 포스트마스터의 주소로 대체하는지 여부를 제어합니다. 이 비트는 RFC 821, RFC 822 또는 RFC 1123을 따르지 않는 비호환 시스템을 수용하는 데 사용됩니다.

비트 2(값 = 4)는 구문적으로 잘못된 반송 주소를 사용할 수 없게 합니다.

비트 3(값 = 8)은 `mailfromdnsverify` 키워드와 동일합니다.

포스트마스터에게 반환되는 메일 내용

키워드:`postheadonly`, `postheadbody`

채널 프로그램이나 정기적인 메일 반송 작업이 포스트마스터와 원래의 보낸 사람 모두에게 메일을 반환할 경우 포스트마스터 복사본은 전체 메시지나 헤더가 될 수 있습니다. 포스트마스터 복사본을 단지 헤더로 제한하면 사용자 메일의 프라이버시 수준이 향상됩니다. 그러나 일반적으로 포스트마스터와 시스템 관리자는 원할 경우 `root` 시스템 권한을 사용하여 메일 내용을 읽을 수 있으므로 이러한 제한만으로 메일 보안이 보장되지는 않습니다. 표 10–12를 참조하십시오.

채널별 포스트마스터 주소 설정

키워드: aliaspostmaster, returnaddress, noreturnaddress, returnpersonal, noreturnpersonal

기본적으로 MTA가 바운스 또는 상태 알림 메일을 생성할 때 사용되는 포스트마스터의 반송 주소는 postmaster@*local-host*입니다. 여기서 *local-host*는 공식 로컬 호스트 이름(로컬 채널에 있는 이름)이고 포스트마스터 개인 이름은 "MTA e-Mail Interconnect"입니다. 잘못된 포스트마스터 주소를 선택할 경우 급격한 루핑과 많은 오류 메시지가 발생할 수 있으므로 주의해야 합니다.

RETURN_ADDRESS 및 RETURN_PERSONAL 옵션을 사용하면 포스트마스터 주소 및 개인 이름에 대한 MTA 시스템 기본값을 설정할 수 있습니다. 또는 채널별 제어를 원할 경우 returnaddress 및 returnpersonal 채널 키워드를 사용할 수 있습니다. returnaddress 및 returnpersonal은 각각 포스트마스터 주소와 포스트마스터 개인 이름을 지정하는 필수 인수를 가집니다. 기본적으로 noreturnaddress와 noreturnpersonal이 지정되며 이것은 기본값이 사용되어야 한다는 것을 의미합니다. 기본값은 RETURN_ADDRESS 및 RETURN_PERSONAL 옵션을 통해 지정하거나 이러한 옵션이 설정되지 않은 경우 보통의 기본값으로 지정됩니다.

aliaspostmaster 키워드가 채널에 있을 경우 사용자 이름 postmaster(소문자, 대문자 또는 대소문자 혼합)로 주소 지정된 모든 메시지는 postmaster@*local-host*로 리디렉션됩니다. 여기서 *local-host*는 공식 로컬 호스트 이름(로컬 채널의 이름)입니다. 인터넷 표준에 따르면 메일을 수락하는 DNS의 모든 도메인이 메일을 수신하는 유효한 포스트마스트 계정을 가져야 한다는 것에 주의합니다. 따라서 여러 다른 도메인에 대해 별개의 포스트마스터 계정을 설정하는 대신 포스트마스터의 책임을 중앙 집중화하려는 경우 이 키워드가 유용할 수 있습니다. 즉, returnaddress가 MTA에서 포스트마스터의 알림 메일을 생성할 때 사용되는 반송 포스트마스터 주소를 제어하는 것과 달리 aliaspostmaster는 포스트마스터로 주소 지정된 메일을 MTA에서 처리하는 방법에 영향을 줍니다.

표 10–12 포스트마스터 및 보낸 사람에게 알림 메일을 보내는 데 사용되는 키워드

키워드	설명
반환되는 메일 내용	**알림에 대한 주소 지정**
notices	알림을 보낸 후 메일을 반환되기 전에 경과할 수 있는 시간을 지정합니다.
nonurgentnotices	낮음 우선 순위를 갖는 메일에 대해 알림을 보낸 후 메일을 반환되기 전에 경과할 수 있는 시간을 지정합니다.
normalnotices	중간 우선 순위를 갖는 메일에 대해 알림을 보낸 후 메일을 반환되기 전에 경과할 수 있는 시간을 지정합니다.
urgentnotices	높음 우선 순위를 갖는 메일에 대해 알림을 보낸 후 메일을 반환되기 전에 경과할 수 있는 시간을 지정합니다.

표 10–12 포스트마스터 및 보낸 사람에게 알림 메일을 보내는 데 사용되는 키워드 *(계속)*

키워드	설명
반환되는 메일	**반환되는 메일에 대한 실패 알림 처리 방법**
sendpost	실패한 모든 메시지의 복사본을 포스트마스터에게 보냅니다.
copysendpost	전송 실패 메일에서 메일 발송자 주소가 비어 있지 않는 한 실패 알림 복사본을 포스트마스터에게 보냅니다. 이 경우 포스트마스터는 실제로 바운스 또는 알림인 메일을 제외하고 실패한 모든 메일의 복사본을 갖게 됩니다.
errsendpost	메시지 발송자에게 알림을 보낼 수 없는 경우 실패 알림 복사본을 포스트마스터에게 보냅니다. nosendpost가 지정된 경우 실패한 메시지는 포스트마스터에게 보내지지 않습니다.
nosendpost	전달이 실패한 모든 메시지의 복사본을 포스트마스터에게 보내지 않습니다.
경고 메일	**경고 메일 처리 방법**
warnpost	경고 메시지의 복사본을 포스트마스터에게 보냅니다. 빈 Warningsto: 헤더 또는 빈 봉투의 From: 주소로 경고가 완전히 억제되지 않은 경우 기본값은 경고 복사본을 포스트마스터에게 보내는 것입니다.
copywarnpost	전달되지 않은 메시지에서 보낸 사람 주소가 비어 있지 않는 한 경고 메시지 복사본을 포스트마스터에게 보냅니다.
errwarnpost	메시지 발송자에게 알림을 보낼 수 없는 경우 경고 메시지 복사본을 포스트마스터에게 보냅니다.
nowarnpost	경고 메시지의 복사본을 포스트마스터에게 보내지 않습니다.
반환되는 메일 내용	**포스트마스터에게 전체 메일을 보낼 것인지 아니면 단순히 헤더만 보낼 것인지 지정**
postheadonly	헤더만 포스트마스터에게 반환됩니다. 포스트마스터 복사본을 단지 헤더로 제한하면 사용자 메일의 프라이버시 수준이 향상됩니다. 그러나 포스트마스터와 시스템 관리자는 원할 경우 root 시스템 권한을 사용하여 메일 내용을 읽을 수 있으므로 이러한 제한만으로 메일 보안이 보장되지는 않습니다.
postheadbody	헤더와 메시지 내용을 모두 반환합니다.
반환되는 메일 내용	**알림에 대한 주소 지정**
includefinal	전달 알림에 최종 수신자 주소 형식을 포함합니다.

표 10–12 포스트마스터 및 보낸 사람에게 알림 메일을 보내는 데 사용되는 키워드 *(계속)*

키워드	설명
returnenvelope	빈 봉투 반송 주소 사용을 제어합니다. returnenvelope 키워드는 비트 플래그 집합으로 해석되는 단일 정수 값을 가집니다. 비트 0(값 = 1)은 MTA에 의해 생성된 반송 알림이 빈 봉투 주소 또는 로컬 포스트마스터의 주소로 작성되는지 여부를 제어합니다. 이 비트를 설정하면 로컬 마스터 주소가 사용되고 이 비트를 지우면 빈 주소가 사용됩니다. 비트 1(값 = 2)은 MTA가 모든 빈 봉투 주소를 로컬 포스트마스터의 주소로 대체하는지 여부를 제어합니다. 이 비트는 RFC 821, RFC 822 또는 RFC 1123을 따르지 않는 비호환 시스템을 수용하는 데 사용됩니다. 비트 2(값 = 4)는 구문적으로 잘못된 반송 주소를 사용할 수 없게 합니다. 비트 3(값 = 8)은 mailfromdnsverify 키워드와 동일합니다.
suppressfinal	원본 주소 형식이 있는 경우 알림 메시지에서 최종 주소 형식을 생략합니다.
useintermediate	목록을 확장한 이후 사용자 메일함 이름이 생성되기 이전에 생성되는 중간 주소 형식을 사용합니다. 이 정보를 사용할 수 없는 경우 최종 형식이 사용됩니다.
반환되는 메일 내용	**알림에 대한 주소 지정**
aliaspostmaster	공식 채널 이름에서 포스트마스터 아이디로 주소 지정된 메시지는 postmaster@local-host로 리디렉션됩니다. 여기서 local-host는 로컬 호스트 이름(로컬 채널에 있는 이름)입니다.
returnaddress	로컬 포스트마스터에 대한 반송 주소를 지정합니다.
noreturnaddress	RETURN_ADDRESS 옵션 값을 포스트마스터 주소 이름으로 사용합니다.
returnpersonal	로컬 포스트마스터에 대한 개인 이름을 설정합니다.
noreturnpersonal	RETURN_PERSONAL 옵션 값을 포스트마스터 개인 이름으로 사용합니다.

10.11 MDN(Message Disposition Notification) 제어

MDN(Message Disposition Notifications)은 MTA가 보낸 사람 및/또는 포스트마스터에게 보내는 전자 메일 보고서로서 메일의 전달 처리를 보고합니다. 예를 들어, 시브(Sieve) 필터에 의해 메시지가 거부된 경우 MDN이 보낸 사람에게 보내집니다. MDN은 또한 읽음 확인, 확인, 수신 알림 또는 전달 확인이라고도 합니다. 시브(Sieve) 스크립트 언어는 일반적으로 메일 필터링 및 휴가 메일에 사용됩니다.

10.11.1 MDN(Message Disposition Notification) 메일 사용자 정의 및 현지화

MDN을 수정 및 현지화하기 위한 지침은 여기에 설명된 약간의 차이점을 제외하고 전달 상태 알림 메일을 사용자 정의 및 현지화하기 위한 지침과 비슷합니다. 248 페이지 "10.10.2 전달 상태 알림 메일 사용자 정의 및 현지화" 및 251 페이지 "10.10.3 생성된 알림 국제화"를 참조하십시오.

매핑(DISPOSITION_LANGUAGE 매핑이라고 함)은 상태 알림을 국제화하는 데 사용되는 notification_language 매핑 테이블(248 페이지 "10.10.2 전달 상태 알림 메일 사용자 정의 및 현지화" 참조)과 비슷합니다.

그러나 이 매핑에 대한 MDN 검사는 다음 형식을 가집니다.

```
type|modifiers|source-channel|header-language|return|recipient
```

여기서

type은 배포 유형이며 displayed, dispatched, processed, deleted, denied 또는 failed 중 하나를 사용할 수 있습니다.

modifiers는 쉼표로 구분된 처리 수정자 목록입니다. 현재 목록은 error, warning, superseded 및 expired입니다.

source-channel은 MDN을 생성하는 소스 채널입니다.

header-language는 accept-language, preferred-language 또는 x-accept-language 중 하나로 지정된 언어입니다. MTA는 이러한 옵션 중에서 존재하는 첫 번째 옵션을 사용합니다.

return은 알림이 반환되는 주소입니다.

recipient는 배포 대상 주소입니다.

처리 매핑의 결과는 세로 막대(|)로 구분되는 둘 또는 세 개의 정보로 구성됩니다. 첫 번째 정보는 배포 알림의 템플리트 파일이 있는 디렉토리입니다. 두 번째 정보는 독립형 배포 텍스트에 적용될 문자 세트입니다. (이 정보는 처리, 특히 휴가 시브(Sieve) 작업에 대한 :mime 매개 변수 사용이나 자동 회신 에코에 의해 생성된 처리에서 템플리트 파일을 사용하지 않으며, 따라서 이러한 파일로부터 문자 세트를 상속할 수 없기 때문에 필요합니다.)마지막으로 세 번째 정보는 알림에 대한 대체 제목 줄입니다. 이 정보는 $T 플래그 역시 매핑에 의해 설정된 경우에만 사용됩니다.

다음의 추가 템플리트 파일이 MDN을 구성하는 데 사용됩니다.

```
disposition_deleted.txt disposition_failed.txtdisposition_denied.txt
disposition_prefix.txtdisposition_dispatched.txt
disposition_processed.txtdisposition_displayed.txt
disposition_suffix.txtdisposition_option.opt
```

이러한 템플리트 파일은 상태 알림 메일에 대한 다양한 return_*.txt 파일과 비슷한 방식으로 사용됩니다. *.txt 파일의 메일 텍스트는 한 행당 78자로 제한되어야 합니다.

10.12 MTA 성능 최적화

이 절에서는 기타 MTA 최적화에 대해 설명합니다. 이 장은 다음 내용으로 구성되어 있습니다.

- 260 페이지 "10.12.1 메일링 목록으로 주소 지정된 메시지의 경우 LDAP 디렉토리에 대한 권한 부여 검사 최적화"

10.12.1 메일링 목록으로 주소 지정된 메시지의 경우 LDAP 디렉토리에 대한 권한 부여 검사 최적화

메일링 목록으로 주소 지정된 메시지의 경우, 메타 문자 대체를 사용하여 LDAP 디렉토리에 대한 권한 부여 검사를 줄일 수 있습니다.

메타 문자 대체는 mgrpModerator, mgrpAllowedBroadcaster 및 mgrpDisallowedBroadcaster 속성에서 지정할 수 있습니다. 특히 주소와 관련된 다양한 메타 문자 시퀀스(전체 주소를 나타내는 $A, 메일함 부분을 나타내는 $U, 도메인 부분을 나타내는 $D)는 현재 봉투 From: 주소를 참조하며, 어떤 경우에는 URL 반환 결과를 일치 가능성이 있는(또는 반드시 일치하는) 항목으로 제한할 때 사용할 수도 있습니다. 그러면 권한 부여 검사가 훨씬 더 효율적으로 수행될 수 있습니다.

새로운 MTA 옵션 PROCESS_SUBSTITUTIONS는 URL을 지정하는 다양한 LDAP 속성에서 대체가 수행되는지 여부를 제어합니다. 이는 비트 인코딩 값이며, 비트는 다음과 같이 정의됩니다.

비트	값	설명
0	1	설정된 경우 mgrpDisallowedBroadcaster에서 대체를 활성화합니다.
1	2	설정된 경우 mgrpAllowedBroadcaster에서 대체를 활성화합니다.
2	4	설정된 경우 mgrpModerator에서 대체를 활성화합니다.
3	8	설정된 경우 mgrpDeliverTo에서 대체를 활성화합니다.
4	16	memberURL에서 대체를 활성화합니다.

PROCESS_SUBSTITUTIONS MTA 옵션의 기본값은 0입니다. 이는 모든 대체가 기본적으로 비활성화되는 것을 의미합니다.

LDAP 조회를 통해 정의되고 목록상의 누구든지 게시가 허용되는 동적 목록을 예로 들 수 있습니다. 그러한 경우, 일반적으로 다음과 같은 속성으로 목록을 정의합니다.

```
mgrpAllowedBroadcaster:
ldap:///o=Sesta,c=US??sub?(&(objectClass=inetMailUser)(objectClass=inetOrgPerson))
mgrpDeliverTo:
ldap:///o=Sesta,c=US??sub?(&(objectClass=inetMailUser)(objectClass=inetOrgPerson))
```

그러나 이렇게 정의하면 권한 부여 검사를 위해 한 번 그리고 실제 수신자 목록 작성을 위해 한 번, 총 두 번 목록을 확장하게 됩니다. 따라서 이 작업은 서버 사용량이 매우 많습니다. 반면에, 현재 봉투 `From:` 주소를 포함하는 항목만 권한 부여 검사에서 반환하도록 제한을 추가한다면 훨씬 더 효율적일 것입니다. 먼저 `PROCESS_SUBSTITUTION` 설정을 2로 변경하고, 다음 항목을 설정할 수 있습니다.

```
mgrpAllowedBroadcaster:
ldap:///o=Sesta,c=US??sub?(&(objectClass=inetMailUser)(objectClass=inetOrgPerson)
(mail=$A)
mgrpDeliverTo:
ldap:///o=Sesta,c=US??sub?(&(objectClass=inetMailUser)(objectClass=inetOrgPerson))
```

이 예에서는 Sesta US의 모든 사용자 항목이 아니라 보낸 사람의 항목에 대해서만 브로드캐스트 권한 부여 검사가 수행됩니다. 그러면 디렉토리 서버에서 수행해야 할 작업이 색인화된 단일 일치 및 단일 반환 값으로 줄어듭니다. 다른 방법은 전체 목록을 반환하고 MTA가 일치 작업을 수행하게 하는 것입니다.

대체에 사용 가능한 정보는 그 속성이 권한 부여 검사에 사용되는지 또는 실제 목록 확장에 사용되는지에 따라 달라집니다. 권한 부여 속성의 경우 전체 주소(`$A`), 도메인(`$D`), 호스트(`$H`) 및 로컬 부분(`$L`)가 모두 인증된 보낸 사람 주소로부터 파생됩니다. 목록 확장 속성의 경우, 이러한 모든 대체 값은 목록을 지정한 봉투 수신자 주소에서 파생됩니다. 그러나 두 경우 모두, 하위 주소 대체(`$S`)는 현재 봉투 수신자 주소로부터 파생됩니다.

목록 확장 URL에서 하위 주소 정보에 액세스할 수 있는 기능은 **메타 그룹**, 즉 서로 다른 그룹을 하나의 전체 모음으로 만드는 단일 그룹 항목을 정의할 수 있도록 해줍니다. 예를 들어 다음의 값이 `mgrpDeliverTo`이고

```
mgrpDeliverTo: ldap:///o=usergroup?mail?sub?(department=$S)
```

해당 `PROCESS_SUBSTITUTIONS` 값이 8인 그룹의 경우 `group+department@domain.com` 형식의 주소를 사용하여 지정된 부서의 모든 구성원에게 메일을 보낼 수 있습니다. 하위 주소가 너무 어려운 경우 전달 매핑과 같은 기법을 사용하여 구문을 변경할 수 있습니다.

◆ ◆ ◆ 11장

11 다시 쓰기 규칙 구성

이 장에서는 imta.cnf 파일에서 다시 쓰기 규칙을 구성하는 방법에 대해 설명합니다. 10장을 아직 읽지 않은 경우 이 장을 읽기 전에 10장을 읽어 보십시오.

이 장은 다음 내용으로 구성되어 있습니다.

Messaging Server의 주소 다시 쓰기 기능은 주소의 호스트 또는 도메인 부분을 조작 및 변경하기 위한 기본 기능입니다. Messaging Server는 별칭, 주소 역방향 데이터베이스 및 특수한 매핑 테이블과 같은 주소 조작을 위한 다른 기능을 제공합니다. 그러나 최상의 성능을 위해서는 주소 조작을 수행할 수 있을 때마다 다시 쓰기 규칙을 사용하는 것이 좋습니다.

11.1 시작하기 전에

imta.cnf 파일의 다시 쓰기 규칙을 변경하면 시작 시에 구성 데이터를 한 번만 로드하는 모든 프로그램이나 채널(예: SMTP 서버)을 imsimta restart 명령을 사용하여 다시 시작해야 합니다. 컴파일된 구성을 사용하는 경우 재컴파일을 수행한 다음 다시 시작해야 합니다.

구성 정보 컴파일과 프로그램 시작에 대한 자세한 내용은 **Sun Java System Messaging Server 6.3 Administration Reference**를 참조하십시오.

11.2 다시 쓰기 규칙 구조

다시 쓰기 규칙은 MTA 구성 파일 `imta.cnf`의 상반부에 표시됩니다. 구성 파일의 각 규칙은 하나의 행으로 나타납니다. 규칙 사이에 주석이 올 수 있지만 빈 행은 허용되지 않습니다. 다시 쓰기 규칙은 빈 행으로 끝나며 그 뒤에 채널 정의가 옵니다. 아래 예는 구성 파일의 다시 쓰기 규칙 섹션을 보여 줍니다.

```
! test.cnf - An example configuration file.
!
! This is only an example of a configuration file. It serves
! no useful purpose and should not be used in a real system.
!
a.com   $U@a-host
b.org   $U@b-host
c.edu   $U%c@b-daemon
d.com   $U%d@a-daemon

! Begin channel definitions
```

다시 쓰기 규칙은 패턴과 그 뒤에 오는 등가 문자열 또는 **템플리트**로 구성됩니다. 두 부분을 공백으로 구분해야 하며 각 부분 내에서는 공백이 허용되지 않습니다. 다시 쓰기 규칙의 구조는 다음과 같습니다.

pattern template

pattern

도메인 이름에서 검색할 문자열을 나타냅니다. 표 11–3에서 패턴은 `a.com`, `b.org`, `c.edu` 및 `d.com`입니다.

패턴이 주소의 도메인 부분과 일치할 경우 다시 쓰기 규칙이 주소에 적용됩니다. 패턴과 템플리트 사이를 공백으로 구분해야 합니다. 패턴 구문에 대한 자세한 내용은 265 페이지 “11.3 다시 쓰기 규칙 패턴 및 태그”를 참조하십시오.

template

다음 중 하나입니다.

UserTemplate%*DomainTemplate*@*ChannelTag*[*controls*]
UserTemplate@*ChannelTag*[*controls*]
UserTemplate%*DomainTemplate*[*controls*]
UserTemplate@*DomainTemplate*@*ChannelTag*[**controls**]
UserTemplate@*DomainTemplate*@*SourceRoute*@*ChannelTag*[*controls*]

여기서

*UserTemplate*는 주소의 사용자 부분이 다시 작성되는 방법을 지정합니다. 대체 시퀀스를 사용하여 원래 주소의 일부나 데이터베이스 조회의 결과를 나타낼 수 있습니다. 대체

시퀀스는 다시 작성된 주소를 생성하기 위해 자신이 나타내는 대상으로 대체됩니다. 표 11–4에서는 $U 대체 시퀀스가 사용됩니다. 자세한 내용은 276 페이지 “11.6 템플리트 대체 및 다시 쓰기 규칙 제어 시퀀스”를 참조하십시오.

*DomainTemplate*는 주소의 도메인 부분이 다시 작성되는 방법을 지정합니다. *UserTemplate*와 마찬가지로 *DomainTemplate*는 대체 시퀀스를 포함할 수 있습니다.

*ChannelTag*는 이 메시지가 전송될 채널을 나타냅니다. (모든 채널 정의는 채널 이름뿐만 아니라 채널 태그를 포함해야 합니다. 채널 태그는 일반적으로 다시 쓰기 규칙과 채널 정의에 모두에 표시됩니다.)

*controls*를 사용하여 규칙의 적용 가능성을 제한할 수 있습니다. 일부 제어 시퀀스는 규칙의 시작 부분에 있어야 하며 다른 제어는 규칙의 끝 부분에 있어야 합니다. 제어에 대한 자세한 내용은 276 페이지 “11.6 템플리트 대체 및 다시 쓰기 규칙 제어 시퀀스”를 참조하십시오.

템플리트 구문에 대한 자세한 내용은 268 페이지 “11.4 다시 쓰기 규칙 템플리트”를 참조하십시오.

11.3 다시 쓰기 규칙 패턴 및 태그

이 절은 다음과 같은 하위 절로 구성되어 있습니다.

대부분의 다시 쓰기 규칙 패턴은 해당 호스트와만 일치하는 특정 호스트 이름이나 전체 하위 도메인의 모든 호스트/도메인과 일치하는 하위 도메인 패턴으로 구성됩니다.

예를 들어, 다음 다시 쓰기 규칙 패턴은 지정된 호스트와만 일치하는 특정 호스트 이름을 포함합니다.

```
host.siroe.com
```

다음 다시 쓰기 규칙 패턴은 전체 하위 도메인의 모든 호스트 또는 도메인과 일치하는 하위 도메인 패턴을 포함합니다.

```
.siroe.com
```

그러나 이 패턴은 정확한 호스트 이름 `siroe.com`과 일치하지 않습니다. 정확한 호스트 이름 `siroe.com`과 일치하려면 별개의 `siroe.com` 패턴이 필요합니다.

MTA는 특정 호스트 이름부터 시작한 다음 특수성을 줄이기 위해 이름을 점차적으로 일반화화여 호스트/도메인 이름의 다시 쓰기를 시도합니다. 이것은 더 특수한 다시 쓰기

규칙 패턴의 사용이 더 일반적인 다시 쓰기 규칙 패턴 사용보다 선호된다는 것을 의미합니다. 예를 들어, 다음 다시 쓰기 규칙 패턴이 구성 파일에 존재한다고 가정해 봅니다.

```
hosta.subnet.siroe.com
.subnet.siroe.com
.siroe.com
```

다시 쓰기 규칙 패턴에 기초하여 `jdoe@hosta.subnet.siroe.com` 주소는 `hosta.subnet.siroe.com` 다시 쓰기 규칙 패턴과 일치하고 `jdoe@hostb.subnet.siroe.com` 주소는 `.subnet.siroe.com` 다시 쓰기 규칙 패턴과 일치하며 `jdoe@hostc.siroe.com` 주소는 `.siroe.com` 다시 쓰기 규칙 패턴과 일치합니다.

특히 하위 도메인 다시 쓰기 규칙 패턴을 통합하는 다시 쓰기 규칙이 인터넷상의 사이트에서 일반적으로 사용됩니다. 이러한 사이트는 일반적으로 고유한 내부 호스트와 서브넷을 위한 여러 다시 쓰기 규칙을 가지며 최상위 인터넷 도메인에 대한 다시 쓰기 규칙을 `internet.rules`(*msg-svr-base*`/config/internet.rules`) 파일의 구성에 포함합니다.

특정 다시 쓰기 규칙을 통해 처리되는 내부 호스트 대상이 아니라 인터넷 대상에 대한 메시지가 제대로 재작성되고 보내는 TCP/IP 채널로 라우팅되도록 `imta.cnf` 파일이 다음을 포함하는지 확인합니다.

- 최상위 인터넷 도메인과 일치하는 패턴이 포함된 다시 쓰기 규칙
- 이러한 패턴과 일치하는 주소를 보내는 TCP/IP 채널로 다시 쓰는 템플리트

```
!    Ascension Island
 .AC                    $U%$H$D@TCP-DAEMON
. [text
.     removed for
.             brevity]
!      Zimbabwe
 .ZW                    $U%$H$D@TCP-DAEMON
```

IP 도메인 리터럴은 비슷한 계층 일치 패턴을 따르지만 왼쪽에서 오른쪽이나 아니라 오른쪽에서 왼쪽으로 일치합니다. 예를 들어, 다음 패턴은 정확하게 IP 리터럴 `[1.2.3.4]`와만 일치합니다.

```
[1.2.3.4]
```

다음 패턴은 `1.2.3.0` 서브넷의 모든 항목과 일치합니다.

```
[1.2.3.]
```

이미 설명했던 더 일반적인 종류의 호스트 또는 하위 도메인 다시 쓰기 규칙 패턴 외에도, 다시 쓰기 규칙은 표 11–1에 요약되어 있으며 다음 하위 절에서 설명하는 여러 특수한 패턴을 사용할 수 있습니다.

표 11–1 다시 쓰기 규칙의 특수한 패턴 요약

패턴	설명/사용
$*	모든 주소와 일치합니다. 이 규칙은 지정된 경우 파일에서의 위치에 상관 없이 가장 먼저 시도됩니다.
$%	백분율 핵 규칙입니다. A%B 형식의 모든 호스트/도메인 지정과 일치합니다.
$!	뱅 스타일 규칙입니다. B!A 형식의 모든 호스트/도메인 지정과 일치합니다.
[ ]	IP 리터럴 모두 일치 규칙입니다. 모든 IP 도메인 리터럴과 일치합니다.
.	모든 호스트/도메인 지정과일치합니다(예: `joe@[129.165.12.11]`).

이러한 특수한 패턴 외에도 Messaging Server는 또한 다시 쓰기 규칙 패턴에 표시될 수 있는 **태그**의 개념을 가집니다. 이러한 태그는 주소를 여러 번 다시 쓸 수 있으며 이전 다시 쓰기에 기초하여 주소와 일치할 다시 쓰기 규칙을 제어함으로써 후속 다시 쓰기를 구별해야 하는 경우에 사용됩니다. 자세한 내용은 268 페이지 “11.3.4 태그된 다시 쓰기 규칙 집합”을 참조하십시오.

11.3.1 백분율 핵과 일치시키는 규칙

MTA는 `A%B` 형식의 주소 다시 쓰기가 실패할 경우 해당 주소를 `A%B@localhost` 형식으로 처리하기 전에 하나의 추가 규칙을 시도합니다. (이러한 주소 형식에 대한 자세한 내용은 268 페이지 “11.4 다시 쓰기 규칙 템플리트”를 참조하십시오.) 이 규칙은 백분율 기호를 포함하는 로컬 부분의 다시 쓰기가 다른 방법(아래 설명된 모두 일치 규칙 포함)으로 실패할 경우에만 활성화됩니다.

백분율 핵 규칙은 특수한 내부 의미를 백분율 핵 주소에 할당하는 데 유용합니다.

11.3.2 뱅 스타일(UUCP) 주소와 일치시키는 규칙

MTA는 `B!A` 형식의 주소 다시 쓰기가 실패할 경우 해당 주소를 `B!A@localhost` 형식으로 처리하기 전에 하나의 추가 규칙을 시도합니다. 이 추가 규칙은 **뱅 스타일 규칙**입니다. 패턴은 `$!`이며 절대 변경되지 않습니다. 이 규칙은 느낌표를 포함하는 로컬 부분의 다시 쓰기가 다른 방법(아래 설명된 기본 규칙 포함)으로 실패할 경우에만 활성화됩니다.

뱅 스타일 규칙은 UUCP 시스템과 라우팅에 대한 포괄적인 지식을 바탕으로 UUCP 스타일 주소를 시스템으로 라우팅하도록 강제 지정하는 데 사용할 수 있습니다.

11.3.3 모든 주소와 일치시키는 규칙

특수 패턴 "."(마침표)는 다른 규칙이 일치하지 않거나 호스트/도메인 지정을 채널 테이블에서 찾을 수 없을 경우 모든 호스트/도메인 지정과 일치합니다. 즉, "." 규칙은 주소 다시 쓰기가 다른 방법으로 실패한 경우 마지막 수단으로 사용됩니다.

주 – 대체 시퀀스와 관련하여 모두 일치 규칙이 일치하고 해당 템플리트가 확장되면 `$H`는 전체 호스트 이름으로 확장되고 `$D`는 마침표 "."로 확장됩니다. 따라서 `$D`는 모두 일치 규칙 템플리트에서 제한적으로 사용됩니다.

11.3.4 태그된 다시 쓰기 규칙 집합

다시 쓰기 프로세스가 진행되면서 다른 규칙 집합을 적용하는 것이 적합할 수 있습니다. 이렇게 하려면 다시 쓰기 규칙 태그를 사용합니다. 현재 태그는 구성 파일 또는 도메인 데이터베이스에서 패턴이 조회되기 전에 각 패턴의 앞에 놓입니다. 다시 쓰기 규칙 템플리트의 `$T` 대체 문자열을 사용하여 일치하는 임의의 다시 쓰기 규칙에 의해 태그가 변경될 수 있습니다(아래에 설명됨).

태그는 다소 고정적일 수 있습니다. 즉, 설정된 태그는 단일 주소에서 추출되는 모든 호스트에 계속 적용됩니다. 이것은 임의의 태그를 사용한 후에 적절한 태그 값으로 시작하는 대체 규칙을 제공할 때 신중해야 한다는 것을 의미합니다. 일반적으로 태그는 매우 전문화된 응용 프로그램에서만 사용되므로 이것은 실제로 거의 문제가 되지 않습니다. 주소 다시 쓰기가 끝나면 태그는 기본 태그(빈 문자열)로 재설정됩니다.

기본적으로 모든 태그 값은 세로 막대(|)로 끝납니다. 이 문자는 일반 주소에서 사용되지 않으므로 패턴의 나머지 부분에서 태그를 자유롭게 나타냅니다.

11.4 다시 쓰기 규칙 템플리트

다음 절에서는 다시 쓰기 규칙의 템플리트 형식에 대해 더욱 자세하게 설명합니다. 표 11–2에는 템플리트 형식이 요약되어 있습니다.

표 11–2 다시 쓰기 규칙의 템플리트 형식 요약

템플리트	사용
A%B	A는 새 사용자/메일함 이름이 되고 B는 새 호스트/도메인 지정이 되며 다시 쓰기를 다시 수행합니다. 269 페이지 "11.4.2 반복되는 다시 쓰기 템플리트, A%B"
A@B	A%B@B로 처리됩니다. 269 페이지 "11.4.1 일반 다시 쓰기 템플리트, A%B@C 또는 A@B"

표 11–2 다시 쓰기 규칙의 템플리트 형식 요약 (계속)

템플리트	사용
A%B@C	A는 새 사용자/메일함 이름이 되고 B는 새 호스트/도메인 지정이 되며 호스트 C와 연관된 채널로 라우팅합니다. 269 페이지 “11.4.1 일반 다시 쓰기 템플리트, A%B@C 또는 A@B”
A@B@C	A@B@C@C로 처리됩니다. 270 페이지 “11.4.3 지정된 경로 다시 쓰기 템플리트, A@B@C@D 또는 A@B@C”
A@B@C@D	A는 새 사용자/메일함 이름이 되고 B는 새 호스트/도메인 지정이 되며 C를 소스 경로로 삽입하고 호스트 D와 연관된 채널로 라우팅합니다. 270 페이지 “11.4.3 지정된 경로 다시 쓰기 템플리트, A@B@C@D 또는 A@B@C”

11.4.1 일반 다시 쓰기 템플리트, A%B@C 또는 A@B

다음 템플리트는 가장 일반적인 형식의 템플리트입니다. 주소의 사용자 부분과 주소의 도메인 부분에 규칙이 적용된 다음 새 주소를 사용하여 메시지를 특정 채널(*ChannelTag*로 표시)로 라우팅합니다.

UserTemplate%*DomainTemplate*@*ChannelTag*[*controls*]

다음 형식의 템플리트는 그 적용에 있어 가장 일반적인 형식의 템플리트와 동일합니다. 그러나 이 형식의 템플리트는 *DomainTemplate*와 *ChannelTag*가 동일한 경우에만 가능합니다.

UserTemplate@*ChannelTag*[*controls*]

11.4.2 반복되는 다시 쓰기 템플리트, A%B

다음 템플리트 형식은 규칙이 적용된 후에 추가 다시 쓰기가 필요한 메타 규칙에 사용됩니다. 규칙이 적용된 후 전체 다시 쓰기 프로세스가 새 결과 주소에서 반복됩니다. (다른 모든 다시 쓰기 규칙 형식은 규칙이 적용된 후 프로세스를 종료합니다.)

UserTemplate%*DomainTemplate*[*controls*]

예를 들어, 다음 규칙은 주소 끝에서 모든 `.removable` 도메인을 제거합니다.

```
.removable $U%$H
```

이러한 반복 규칙을 사용할 때에는 매우 주의해야 합니다. 함부로 사용하면 “규칙 루프”가 발생하기 때문입니다.따라서 메타 규칙은 반드시 필요한 경우에만 사용해야 합니다. `imsimta test -rewrite` 명령을 사용하여 메타 규칙을 테스트합니다. `test -rewrite` 명령에 대한 자세한 내용은 **Sun Java System Messaging Server 6.3 Administration Reference**를 참조하십시오.

11.4.3 지정된 경로 다시 쓰기 템플리트, A@B@C@D 또는 A@B@C

다음 템플리트는 주소에 *ChannelTag*가 소스 경로로 삽입된다는 점을 제외하면 더 일반적인 *UserTemplate*%*DomainTemplate*@*ChannelTag* 템플리트와 같은 방식으로 작동합니다(첫 번째 구분자 문자가 다르다는 점에 주의). 그런 다음 메시지가 *ChannelTag*로 라우팅됩니다.

UserTemplate@*DomainTemplate*@*Source-Route*
@*ChannelTag*[*controls*]

다시 작성된 주소는 @*route* :*user*@*domain* 이 됩니다. 또한 다음 템플리트도 유효합니다.

UserTemplate@*DomainTemplate*@*ChannelTag*[*controls*]

예를 들어, 다음 규칙은 `jdoe@com1` 주소를 `@siroe.com:jdoe@com1` 소스 라우팅 주소로 다시 씁니다. 채널 태그는 `siroe.com`이 됩니다.

```
com1 $U@com1@siroe.com
```

11.4.4 다시 쓰기 규칙 템플리트의 대소문자 구분

다시 쓰기 규칙의 패턴과 달리 템플리트의 대소문자는 유지됩니다. 이것은 다시 쓰기 규칙을 사용하여 대소문자를 구분하는 메일 시스템에 인터페이스를 제공할 때 필요합니다. 주소에서 추출된 자료를 대체하는 $U 및 $D와 같은 대체 시퀀스도 원래 대소문자를 유지한다는 것에 주의합니다.

대체된 자료에 특정 대소문자를 사용하도록 강제 지정하는 것이 바람직한 경우(예: UNIX 시스템에서 메일함에 소문자를 사용하도록 강제 지정하는 경우) 특수한 대체 시퀀스를 템플리트에 사용하여 대체된 자료에 원하는 대소문자를 적용할 수 있습니다. 특히, $\는 대체된 이후의 자료를 소문자로 강제하고 $^는 대문자로 강제하며 $_는 원래 대소문자를 사용하도록 지정합니다.

예를 들어, 다음 규칙을 사용하여 `unix.siroe.com` 주소에 대해 메일함을 소문자로 강제 지정할 수 있습니다.

```
unix.siroe.com    $\$U$_%unix.siroe.com
```

11.5 MTA가 다시 쓰기 규칙을 주소에 적용하는 방법

다음 단계는 MTA가 다시 쓰기 규칙을 특정 주소에 적용하는 방법을 설명합니다.

1. MTA는 주소에서 첫 번째 호스트 또는 도메인 지정을 추출합니다.

   주소는 다음과 같이 둘 이상의 호스트 또는 도메인 이름을 지정할 수 있습니다.

   ```
   jdoe%hostname@siroe.com.
   ```

2. 첫 번째 호스트 또는 도메인 이름을 식별한 후에 MTA는 패턴이 호스트 또는 도메인 이름과 일치하는 다시 쓰기 규칙을 스캔하는 검색을 수행합니다.
3. 일치하는 다시 쓰기 규칙이 발견되면 MTA는 해당 규칙의 템플리트 부분에 따라 주소를 다시 씁니다.
4. 마지막으로 MTA는 채널 태그를 각 채널과 연관된 호스트 이름과 비교합니다.

   일치하는 항목이 발견된 경우 MTA는 연관된 채널의 대기열에 메시지를 넣고, 그렇지 않을 경우 다시 쓰기 프로세스가 실패합니다. 일치하는 채널이 로컬 채널일 경우 별칭 데이터베이스와 별칭 파일을 조회하여 주소의 일부 추가 다시 쓰기가 발생할 수 있습니다.

이러한 단계는 다음 하위 절에 더욱 자세하게 설명되어 있습니다.

주 – 기존 채널에 속하지 않은 채널 태그를 사용하면 주소가 이 규칙과 일치하는 메시지가 바운스됩니다. 즉, 일치하는 메시지를 라우팅할 수 없게 됩니다.

이 절은 다음과 같은 하위 절로 구성되어 있습니다.

11.5.1 단계 1. 첫 번째 호스트 또는 도메인 지정 추출

주소 다시 쓰기 프로세스는 첫 번째 호스트나 도메인 지정을 주소에서 추출하는 것으로 시작됩니다. (RRFC 822 주소 규칙에 익숙하지 않은 경우에는 다음 내용을 이해할 수 있도록 이 표준을 읽어보는 것이 좋습니다.)주소의 호스트/도메인 지정이 스캔되는 순서는 다음과 같습니다.

1. 소스 경로의 호스트(왼쪽에서 오른쪽으로 읽음)
2. “at” 기호(@) 오른쪽에 나타나는 호스트
3. 마지막 단일 백분율 기호(%) 오른쪽에 있는 호스트

4. 첫 번째 느낌표(!)

주소 다시 쓰기를 수행하는 채널에서 bangoverpercent 키워드가 적용될 경우(즉, 메시지를 대기열에 넣으려고 시도하는 채널 자체가 bangoverpercent 채널 키워드로 표시된 경우) 마지막 두 개 항목의 순서가 바뀝니다.

표 11–3에는 몇 개의 주소와 첫 번째로 추출할 수 있는 호스트 이름의 예가 나와 있습니다.

표 11–3 추출된 주소 및 호스트 이름

주소	첫 번째 호스트 도메인 지정	설명
user@a	a	"단순 형식의" 도메인 이름
user@a.b.c	a.b.c	"정규화된" 도메인 이름(FQDN)
user@[0.1.2.3]	[0.1.2.3]	"도메인 리터럴"
@a:user@b.c.d	a	단순 형식의 도메인 이름, 즉 "route"
@a.b.c:user@d.e.f	a.b.c	소스 라우팅 주소입니다. 경로 부분이 정규화됩니다.
@[0.1.2.3]:user@d.e.f	[0.1.2.3]	소스 라우팅 주소입니다. 경로 부분이 도메인 리터럴입니다.
@a,@b,@c:user@d.e.f	a	a, b, c 순서의 라우팅을 가진 소스 라우팅 주소입니다.
@a,@[0.1.2.3]:user@b	a	경로 부분에 도메인 리터럴이 있는 소스 라우팅 주소입니다.
user%A@B	B	이 비표준 라우팅 형식은 "백분율 핵"
user%A	A	
user%A%B	B	
user%%A%B	B	
A!user	A	"뱅 스타일" 주소 지정. 일반적으로 UUCP에 사용됩니다.
A!user@B	B	
A!user%B@C	C	
A!user%B	B	nobangoverpercent 키워드가 활성화됩니다. 기본값입니다.
A!user%B	A	bangoverpercent 키워드가 활성화됩니다.

RFC 822는 주소에 있는 느낌표(!)와 백분율 기호(%)의 해석을 다루지 않습니다. at 기호(@)가 존재하지 않을 경우 백분율 기호는 관례상 at 기호와 동일한 방식으로 해석되며, 따라서 Messaging Server MTA에 이 규칙이 적용됩니다.

반복되는 백분율 기호의 특수한 해석은 백분율 기호를 로컬 아이디의 일부로 허용하는 데 사용됩니다. 이것은 일부 외국 메일 시스템 주소를 처리하는 데 유용할 수 있습니다. 느낌표에 대한 해석은 RFC 976의 "뱅 스타일" 주소 규칙을 따르며 Messaging Server MTA에서 UUCP 주소를 사용할 수 있게 합니다.

이러한 해석의 순서는 RFC 822 또는 RFC 976에 지정되어 있지 않으므로 `bangoverpercent` 및 `nobangoverpercent` 키워드를 사용하여 다시 쓰기를 수행하는 채널에 의해 적용되는 순서를 제어할 수 있습니다. 대부분 기본값이 "표준"이지만 경우에 따라서는 대체 설정이 유용할 수도 있습니다.

주 – 주소에서는 느낌표(!) 또는 백분율 기호(%)를 사용하지 않는 것이 좋습니다.

11.5.2 단계 2. 다시 쓰기 규칙 스캔

첫 번째 호스트 또는 도메인 지정이 주소에서 추출된 후 MTA는 다시 쓰기 규칙을 참조하여 그 처리 방법을 결정합니다. 호스트/도메인 지정은 각 규칙의 패턴 부분(즉, 각 규칙의 왼쪽)과 비교됩니다. 이 비교는 RFC 822에 명시된 대로대소문자를 구분하지 않습니다. MTA에서는 대소문자를 구분하지 않지만 가능하면 대소문자를 유지합니다.

호스트 또는 도메인 지정이 어떠한 패턴과도 일치하지 않으면(이러한 경우를 "어떠한 규칙과도 일치하지 않음"이라고 함) 호스트 또는 도메인 지정의 첫 번째 부분, 즉 첫 번째 마침표의 앞 부분(대개 호스트 이름)이 제거되고 별표(*)로 대체된 다음, 구성 파일 다시 쓰기 규칙에서만 결과 호스트 또는 도메인 지정을 찾으려는 또 다른 시도가 수행됩니다(도메인 데이터베이스는 참조되지 않음).

이 시도가 실패하면 첫 번째 부분이 제거되고 프로세스가 반복됩니다. 또 다시 실패할 경우에는 다음 부분(일반적으로 하위 도메인)이 제거되고 다시 쓰기 프로세스가 처음에는 별표를 사용하고 그 다음에는 별표 없이 다시 시도됩니다. 별표를 포함하는 모든 검사는 구성 파일 다시 쓰기 규칙 테이블에서만 수행되며 도메인 데이터베이스는 검사되지 않습니다. 이 프로세스는 일치하는 항목이 발견되거나 전체 호스트 또는 도메인 지정이 사용될 때까지 계속됩니다. 이 절차는 가장 구체적인 도메인을 우선 일치시키고 점차 덜 구체적이면서 더 일반적인 도메인을 일치시키는 방향으로 진행됩니다.

일치 절차의 알고리즘에 대한 자세한 내용은 다음과 같습니다.

- 호스트/도메인 지정이 비교 문자열 `spec_1` 및 `spec_2`의 초기 값으로 사용됩니다(예: `spec_1 = spec_2 = a.b.c`).
- 일치하는 항목이 발견될 때까지 비교 문자열 `spec_1`이 구성 파일에 이어 도메인 데이터베이스에 있는 각 다시 쓰기 규칙의 패턴 부분과 비교됩니다. 일치하는 항목이 발견되면 일치 절차가 종료합니다.

- 일치하는 항목이 발견되지 않을 경우 spec_2의 별표가 아닌 맨 왼쪽 부분이 별표로 변환됩니다. 예를 들어, spec_2가 a.b.c인 경우 *.b.c 로 변경되고 spec_2가 *.b.c인 경우 *.*.c로 변경됩니다. 일치하는 항목이 발견되면 일치 절차가 종료됩니다.
- 일치하는 항목이 발견되지 않을 경우 선행 마침표를 포함한 비교 문자열 spec_1의 첫 번째 부분이 제거됩니다. spec_1이 한 부분으로만 구성된 경우(예: .c 또는 c) 문자열은 마침표 "."로 바뀝니다. 결과 문자열 spec_1의 길이가 0이 아닌 경우에는 단계 1로 돌아갑니다. 결과 문자열의 길이가 0인 경우(예: 이전에 "."인 경우) 조회 프로세스가 실패하고 일치 절차가 종료됩니다.

예를 들어, dan@sc.cs.siroe.edu 주소를 다시 작성한다고 가정해 봅니다. 이 경우 MTA는 다음 패턴을 주어진 순서대로 찾습니다.

```
sc.cs.siroe.edu
*.cs.siroe.edu
.cs.siroe.edu
*.*.siroe.edu
.siroe.edu
*.*.*.edu
.edu
*.*.*.*
.
```

11.5.3 단계 3. 템플리트에 따라 주소 다시 쓰기

호스트/도메인 지정은 다시 쓰기 규칙과 일치할 경우 규칙의 템플리트 부분에 따라 다시 작성됩니다. 템플리트는 다음 세 가지 정보를 지정합니다.

1. 주소의 새 아이디
2. 주소의 새 호스트/도메인 지정
3. 이 주소에 대한 메시지를 보내야 하는 기존 MTA 채널을 식별하는 채널 태그

11.5.4 단계 4. 다시 쓰기 프로세스 완료

호스트/도메인 지정이 다시 작성된 후 다음 작업 중 하나가 수행될 수 있습니다.

- 채널 태그가 로컬 채널 및 routelocal 채널 키워드로 표시된 채널과 관련이 없거나 주소에 추가 호스트/도메인 지정이 없을 경우 다시 작성된 지정은 다시 쓰기를 위해 추출된 원래 지정을 대체하는 주소로 대체되고 다시 쓰기 프로세스가 종료합니다.
- 채널 태그가 로컬 채널이나 routelocal로 표시된 채널과 일치하거나 주소에 추가 호스트/도메인 지정이 있을 경우 다시 작성된 주소를 버리고 원래(초기) 호스트/도메인 지정이 주소에서 제거되며 새 호스트/도메인 지정이 주소에서 추출된 후 전체 프로세스가 반복됩니다. 다시 쓰기는 모든 호스트/도메인 지정이 사라지거나 로컬이 아닌 채널 또는 routelocal이 아닌 채널을 통한 경로가 발견될 때까지

계속됩니다. 이 반복 기법은 MTA가 소스 라우팅에 대한 지원을 제공하는 방법입니다. 실제로 로컬 시스템 및 routelocal 시스템을 통과하는 여분의 경로는 이 프로세스에 의해 주소에서 제거됩니다.

11.5.5 다시 쓰기 규칙 실패

호스트 도메인 지정이 다시 쓰기 규칙이 일치하는 데 실패하거나 기본 규칙이 존재하지 않을 경우 MTA는 지정을 "있는 그대로" 사용합니다. 예를 들어, 원래 지정은 새 지정과 라우팅 시스템 모두가 됩니다. 주소에 유효하지 않은 호스트/도메인 지정이 있을 경우 라우팅 시스템이 임의 채널과 연관된 어떠한 시스템 이름과도 일치하지 않을 때 이점이 감지되며 메시지가 바운스됩니다.

11.5.6 다시 쓰기 후의 구문 검사

다시 쓰기 규칙이 주소에 적용된 후 추가 구문 검사가 수행되지 않습니다. RFC 822를 따르지 않는 형식으로 주소를 변환하는 데 다시 쓰기 규칙이 사용될 수 있다는 점에서 일부러 이러한 검사가 수행되지 않는 것입니다. 그러나 이것은 또한 구성 파일의 실수로 인해 MTA에서 잘못된 또는 유효하지 않은 주소를 가진 메시지가 보내질 수 있다는 것을 의미합니다.

11.5.7 도메인 리터럴 처리

도메인 리터럴은 다시 쓰기 프로세스 동안 특수하게 처리됩니다. 주소의 도메인 부분에 있는 도메인 리터럴이 다시 쓰기 규칙 패턴과 일치하지 않을 경우 이러한 리터럴은 마침표로 구분하고 대괄호로 묶은 문자열 그룹으로 해석됩니다. 맨 오른쪽의 문자열이 제거되며 검색이 반복됩니다. 이 작업이 수행되지 않을 경우 빈 대괄호만 남을 때까지 다음 문자열이 제거됩니다. 빈 대괄호에 대한 검색이 실패하면 전체 도메인 리터럴이 제거되고 도메인 주소의 다음 섹션(있을 경우)에서 다시 쓰기가 진행됩니다. 도메인 리터럴의 내부 처리에서 별표는 사용되지 않습니다. 전체 도메인 리터럴이 별표로 대체될 경우 별표 수는 도메인 리터럴의 요소 수에 해당합니다.

일반 도메인 또는 호스트 지정과 마찬가지로 도메인 리터럴은 또한 가장 구체적인 순서에서 가장 일반적인 순서로 시도됩니다. 해당 패턴이 일치하는 첫 번째 규칙은 호스트 또는 도메인 지정을 다시 쓰는 데 사용되는 규칙입니다. 규칙 목록에 두 개의 동일한 패턴이 있을 경우 먼저 표시된 규칙이 사용됩니다.

예를 들어, `dan@[128.6.3.40]` 주소를 다시 작성한다고 가정해 봅니다. 다시 쓰기 프로세스는 `[128.6.3.40]`, `[128.6.3.]`, `[128.6.]`, `[128.]`, `[]`, `[*.*.*.*]`를 차례로 찾은 다음 마지막으로 모두 일치 규칙 `"."`을 찾습니다.

11.6 템플리트 대체 및 다시 쓰기 규칙 제어 시퀀스

대체는 문자열을 다시 작성된 주소(사용된 특정 대체 시퀀스에 의해 결정되는 값)에 삽입하여 아이디나 주소를 다시 쓰는 데 사용됩니다. 이 절은 다음과 같은 하위 절로 구성되어 있습니다.

예를 들어, 다음 템플리트에서 `$U`는 대체 시퀀스입니다. 이 대체 시퀀스는 템플리트의 출력에서 다시 쓰는 주소의 *username* 부분을 대체합니다. 따라서 이 템플리트로 `jdoe@mailhost.siroe.com`이 다시 작성된 경우 `$U`가 원래 주소의 *username* 부분인 `jdoe`로 대체되어 결과 출력은 `jdoe@siroe.com`이 됩니다.

```
$U@siroe.com
```

제어 시퀀스는 특정 다시 쓰기 규칙의 적용 가능성에 추가 조건을 부과합니다. 다시 쓰기 규칙의 패턴 부분이 검사되는 호스트 또는 도메인 지정과 일치해야 할 뿐만 아니라 다시 쓰는 주소의 다른 측면에서 제어 시퀀스에 의해 설정된 조건을 충족해야 합니다. 예를 들어, 다시 쓰는 주소는 $E 제어 시퀀스가 있을 경우 봉투 주소여야 하며 $F 제어 시퀀스가 있을 경우 정방향 지정 주소여야 합니다. 다음 다시 쓰기 규칙은`user@siroe.com` 형식의 (다시 쓰기) 봉투의 To: 주소에만 적용됩니다.

```
siroe.com $U@mail.siroe.com$E$F
```

도메인 또는 호스트 지정이 다시 쓰기 규칙의 패턴 부분과 일치하지만 규칙 템플리트의 제어 시퀀스가 부과한 모든 조건을 충족하지 않을 경우 다시 쓰기 규칙은 실패하고 다른 적용 가능한 규칙을 찾기 위해 다시 쓰기 프로세스가 계속됩니다.

표 11–4에는 템플리트 대체와 제어 시퀀스가 요약되어 있습니다.

표 11–4 다시 쓰기 규칙 템플리트 대체 및 제어 시퀀스 요약

대체 시퀀스	대체 대상
$D	일치한 도메인 지정 부분
$H	호스트/도메인 지정의 일치하지 않는 부분(패턴에 있는 점의 왼쪽)
$L	도메인 리터럴의 일치하지 않는 부분(패턴 리터럴에 있는 점의 오른쪽)
$U	원래 주소의 아이디
$nA	위치 0에서 시작하여 현재 주소의 n번째 왼쪽 문자를 삽입합니다. n을 생략하면 전체 주소가 삽입됩니다.
$nX	0에서 시작하여 메일 호스트의 n번째 왼쪽 구성 요소를 삽입합니다. n을 생략하면 전체 메일 호스트가 삽입됩니다.
$0U	원래 주소에서 모든 하위 주소를 뺀 로컬 부분(아이디)
$1U	원래 주소의 로컬 부분(아이디)에 있는 하위 주소(있을 경우)
$$	리터럴 달러 기호($) 삽입
$%	리터럴 백분율 기호(%) 삽입
$@	리터럴 at 기호(@) 삽입
$\	모두 소문자로 적용
$^	모두 대문자로 적용
$_	원래 대소문자 사용
$=	대체된 후속 문자가 LDAP 검색 필터에 삽입하기 적합하게 인용되도록 합니다.
$W	임의의 고유 문자열로 대체
$]...[	LDAP 검색 URL 조회
$.	임시 LDAP 조회에 실패한 경우 매핑 항목의 결과로 처리되는 문자열을 지정합니다.
$(text)	일반 데이터베이스 대체(조회에 실패할 경우 규칙이 실패함)
${...}	지정된 매핑을 제공된 문자열에 적용
$[...]	사용자 제공 루틴 호출(결과로 대체)
$&*n*	일치하지 않는(와일드카드로 표시된) 호스트의 *n*번째부분(0부터 시작하여 왼쪽에서 오른쪽으로 계산)
$!*n*	일치하지 않는(와일드카드로 표시된) 호스트의 *n*번째부분(0부터 시작하여 오른쪽에서 왼쪽으로 계산)
$**n*	일치하는 패턴의 *n*번째 부분(0부터 시작하여 왼쪽에서 오른쪽으로 계산)
$#*n*	일치하는 패턴의 *n*번째 부분(0부터 시작하여 오른쪽에서 왼쪽으로 계산)

표 11–4 다시 쓰기 규칙 템플리트 대체 및 제어 시퀀스 요약 *(계속)*

대체 시퀀스	대체 대상
$*n*D	일치한 도메인 지정 부분(0부터 시작하는 n번째 맨 왼쪽 부분부터 대체)
$*n*H	일치하지 않는 호스트/도메인 지정 부분(0부터 시작하는 n번째 맨 왼쪽 부분부터 대체)
제어 시퀀스	다시 쓰기 규칙에 미치는 영향
$1M	채널이 내부 재처리 채널일 경우에만 적용됩니다.
$1N	채널이 내부 재처리 채널이 아닐 경우에만 적용됩니다.
$1~	보류 중인 모든 채널 일치 검사를 수행합니다. 검사가 실패할 경우 현재 다시 쓰기 규칙 템플리트의 처리가 성공적으로 종료합니다.
$A	호스트가 at 기호의 오른쪽에 있을 경우 적용됩니다.
$B	헤더/본문 주소에만 적용됩니다.
$C *channel*	*channel*로 보낼 경우 실패합니다.
$E	봉투 주소에만 적용됩니다.
$F	정방향 지정(예: To:) 주소에만 적용됩니다.
$M *channel*	*channel*이 주소를 다시 쓰는 경우에만 적용됩니다.
$N *channel*	*channel*이 주소를 다시 쓰는 경우 실패합니다.
$P	호스트가 백분율 기호의 오른쪽에 있을 경우 적용됩니다.
$Q *channel*	*channel*로 보낼 경우 적용됩니다.
$R	역방향 지정(예: From:) 주소에만 적용됩니다.
$S	호스트가 소스 경로에서 온 경우 적용됩니다.
$T*newtag*	다시 쓰기 규칙 태그를 newtag로 설정합니다.
$V*host*	호스트 이름이 LDAP 디렉토리(DC 트리에 있거나 가상 도메인으로 존재)에 정의되지 않은 경우 실패합니다. LDAP 검색의 시간이 초과되면 호스트 이름 다음의 문자 바로 뒤부터 시작하는 다시 쓰기 패턴의 나머지 부분은 MTA 옵션 문자열 `DOMAIN_FAILURE`로 대체됩니다.
$X	호스트가 느낌표의 왼쪽에 있을 경우 적용됩니다.
$Z*host*	호스트 이름이 LDAP 디렉토리(DC 트리에 있거나 가상 도메인으로 존재)에 정의된 경우 실패합니다. LDAP 검색의 시간이 초과되면 호스트 이름 다음의 문자 바로 뒤부터 시작하는 다시 쓰기 패턴의 나머지 부분은 MTA 옵션 문자열 `DOMAIN_FAILURE`로 대체됩니다.
$*n*T	기본 `ALIAS_MAGIC` 설정을 대체합니다. 여기서 *n*은 `ALIAS_MAGIC` MTA 옵션의 적절한 값입니다. 별칭 확장 중에 규칙이 일치할 때 도메인의 설정을 대체합니다.
$?*errmsg*	다시 쓰기가 실패할 경우 기본 오류 메시지 대신에 *errmsg*를 반환합니다. 오류 메시지는 US ASCII여야 합니다.

표 11–4 다시 쓰기 규칙 템플리트 대체 및 제어 시퀀스 요약 *(계속)*

대체 시퀀스	대체 대상
$*number*?*errmsg*	다시 쓰기가 실패할 경우 기본 오류 메시지 대신에 *errmsg*를 반환하고 SMTP 확장 오류 코드를 *a.b.c*로 설정합니다. ■ *a*는 *number*/ 1000000(첫 번째 자리 값) ■ *b*는 (*number*/1000) 나머지 1000(두 번째에서 네 번째 자리의 값) ■ *c*는 *number* 나머지 1000(마지막 세 개 자리의 값) 다음 예는 오류 코드를 3.45.89로 설정합니다. `$3045089?the snark is a boojum`

11.6.1 아이디 및 하위 주소 대체, $U, $0U, $1U

템플리트의 모든 $U는 원래 주소의 아이디(RFC 822 "로컬 부분")로 대체됩니다. a."b" 형식의 사용자 이름은 "a.b"로 대체된다는 것에 주의합니다. 이는 RFC2822에서 RFC 822의 이전 구문을 더 이상 사용하지 않고 앞으로는 후자의 형식이 필수적으로 사용될 것이기 때문입니다.

템플리트의 모든 $0U는 하위 주소 및 하위 주소 표시 문자(+)를 빼고 원래 주소의 아이디로 대체됩니다. 템플리트의 모든 $1U는 원래 주소의 하위 주소 및 하위 주소 표시 문자(있을 경우)로 대체됩니다. 따라서 $0U 및 $1U가 아이디의 보완적 부분이며 $0U$1U는 $U와 동일하다는 것에 주의합니다.

11.6.2 호스트/도메인 및 IP 리터럴 대체, $D, $H, $nD, $nH, $L

모든 $H는 규칙에 의해 일치되지 않은 호스트/도메인 지정 부분으로 대체됩니다. 모든 $D는 다시 쓰기 규칙에 의해 일치된 호스트/도메인 지정 부분으로 대체됩니다. $nH 및 $nD 문자는 0부터 계산되는 n번째 맨 왼쪽 부분에서 일반 $H 또는 $D 부분을 대체합니다. 즉, $nH 및 $nD는 각각 $H 또는 $D 대체의 맨 왼쪽에서 1부터 계산되는 n개의 부분을 생략합니다. 특히, $0H는 $H와 동일하며 $0D는 $D와 동일합니다.

예를 들어, `jdoe@host.siroe.com` 주소가 다음 다시 쓰기 규칙과 일치한다고 가정해 봅니다.

```
host.siroe.com     $U%$1D@TCP-DAEMON
```

결과 주소는 `TCP-DAEMON`을 보내는 채널로 사용하는 `jdoe@siroe.com`입니다. 여기에서 $D는 일치한 전체 도메인 `host.siroe.com`으로 대체될 것이지만 $1D는 첫 번째 항목(`siroe`)으로부터 일치하는 부분으로 대체되므로 `siroe.com`으로 대체됩니다.

$L은 다시 쓰기 규칙에 의해 일치되지 않은 도메인 리터럴 부분을 대체합니다.

11.6.3 리터럴 문자 대체, $$, $%, $@

$, % 및 @ 문자는 일반적으로 다시 쓰기 규칙 템플리트에서 메타 문자입니다. 이러한 문자의 리터럴 삽입을 수행하려면 달러 문자 $를 사용하여 해당 문자를 인용합니다. 즉, $$는 단일 달러 기호 $로 확장되고 $%는 단일 백분율 %(이 경우 백분율은 템플리트 필드 구분자로 해석되지 않음)로 확장되며 $@은 단일 at 기호 @(마찬가지로 필드 구분자로 해석되지 않음)으로 확장됩니다.

11.6.4 LDAP 쿼리 URL 대체, $]...[

`$]ldap-url[` 형식의 대체는 LDAP 쿼리 URL로 해석되며 LDAP 쿼리 결과가 대체됩니다. 표준 LDAP URL은 호스트와 포트가 생략된 채로 사용됩니다. 호스트와 포트는 대신 `msg.conf` 파일(`local.ldaphost` 및 `local.ldapport` 속성)에서 지정됩니다.

즉, LDAP URL은 다음과 같이 지정해야 하며 여기에서 대괄호 문자 [ ]는 URL의 선택적 부분을 나타냅니다.

```
ldap:///dn[?attributes[?scope?filter]]
```

`dn`은 필수 항목으로서 검색 기준을 지정하는 고유 이름입니다. URL의 선택적 속성, 범위 및 필터 부분은 반환할 정보를 더욱 구체화합니다. 다시 쓰기 규칙의 경우 반환 지정을 위한 속성은 `mailRoutingSystem` 속성(또는 이와 유사한 속성)이 될 수 있습니다. 범위는 base(기본값), one 또는 sub가 될 수 있으며 필터는 `mailDomain` 값이 다시 쓰는 도메인과 일치하는 객체의 반환을 요청하는 것이 될 수 있습니다.

LDAP 디렉토리 스키마가 `mailRoutingSystem` 및 `mailDomain` 속성을 포함할 경우 주어진 주소를 라우팅할 시스템을 결정하기 위한 다시 쓰기 규칙은 다음과 같이 나타날 수 있습니다. 여기에서 LDAP URL 대체 시퀀스 $D는 현재 도메인 이름을 생성된 LDAP 쿼리로 대체하는 데 사용됩니다.

```
.siroe.com \
  $U%$H$D@$]ldap:///o=siroe.com?mailRoutingSystem?sub? \
  (mailDomain=$D)
```

백슬래시 문자는 쉽게 알아볼 수 있도록 하나의 논리적 다시 쓰기 규칙 행을 계속해서 두 번째 물리적 행으로 이어주는 데 사용됩니다. 표 11–5에는 LDAP URL 대체 시퀀스가 나열되어 있습니다.

표 11–5 LDAP URL 대체 시퀀스

대체 시퀀스	설명
`$$`	리터럴 $ 문자

표 11–5 LDAP URL 대체 시퀀스 *(계속)*

대체 시퀀스	설명
$.	임시 LDAP 조회에 실패한 경우 매핑 항목의 결과로 처리되는 문자열을 지정합니다. 기본적으로 현재 규칙의 적용 기간에만 임시 실패 문자열이 설정된 상태로 있습니다. "$.."를 사용하여 기본 상태로 돌아갈 수 있습니다. 이 기본 상태에서는 임시 실패 문자열이 설정되지 않았으며, 임시 LDAP 실패 발생 시 매핑 항목이나 다시 쓰기 규칙 실패가 발생합니다. 디렉토리 항목 일치 실패 이외의 모든 오류는 임시 오류로 간주됩니다. 일반적으로 잘못된 LDAP URL로 인한 오류와 디렉토리 서버 구성 문제로 인한 오류는 구분할 수 없습니다.
$~*account*	사용자 계정의 홈 디렉토리
$A	주소
$D	도메인 이름
$H	호스트 이름(정규화된 도메인 이름의 첫 번째 부분)
$L	~ 또는 _와 같은 특수한 선행 문자를 뺀 아이디
$S	하위 주소
$U	아이디

MTA는 이제 다시 쓰기 규칙 및 매핑에서 수행된 조회에서 URL 결과를 캐시합니다. 이 새로운 URL 결과 캐시는 두 개의 새로운 MTA 옵션, 즉 URL_RESULT_CACHE_SIZE(기본값은 10000개 항목) 및 URL_RESULT_CACHE_TIMEOUT(기본값은 600초)에 의해 제어됩니다.

11.6.5 일반 데이터베이스 대체, $(...)

$(텍스트) 형식의 대체는 특수하게 처리됩니다. 텍스트 부분은 특수한 일반 데이터베이스에 액세스하기 위한 키로 사용됩니다. 이 데이터베이스는 *msg-svr-base*/config/imta_tailor 파일의 IMTA_GENERAL_DATABASE 옵션으로 지정하는 파일(일반적으로 *msg-svr-base*/db/generaldb.db)로 구성됩니다.

데이터베이스에서 “text-string”이 발견될 경우 데이터베이스의 해당 템플리트가 대체됩니다. "text-string"이 데이터베이스의 항목과 일치하지 않을 경우 다시 쓰기 프로세스는 실패합니다(다시 쓰기 규칙이 처음에 일치하지 않은 것과 같음). 대체에 성공할 경우 추가 대체를 위해 데이터베이스에서 추출된 템플리트가 다시 스캔됩니다. 그러나 참조가 무한대로 반복되는 것을 방지하기 위해 추출된 템플리트의 추가 $(텍스트) 대체는 금지됩니다.

예를 들어, jdoe@siroe.siroenet 주소가 다음 다시 쓰기 규칙과 일치한다고 가정해 봅니다.

```
.SIROENET $($H)
```

이 경우 텍스트 문자열 `siroe`가 일반 데이터베이스에서 조회되고 조회 결과(있을 경우)가 다시 쓰기 규칙의 템플리트에 사용됩니다. `siroe`를 조회한 결과가 `$u%eng.siroe.com@siroenet`이라고 가정해 봅니다. 그러면 템플리트 출력은 `jdoe@eng.siroe.com`(즉, 아이디 = `jdoe`, 호스트/도메인 지정 = `eng.siroe.com`)이 되고 라우팅 시스템은 `siroenet`이 됩니다.

일반 텍스트 데이터베이스가 있는 경우 제대로 작동하려면 세계 공용이어야 합니다. 자세한 내용은 240 페이지 “10.9.1 MTA 텍스트 데이터베이스”를 참조하십시오.

11.6.6 지정된 매핑 적용, ${...}

.SIROENET $($H) `${mapping,argument}` 형식의 대체는 MTA 매핑 파일에서 매핑을 찾아 적용하는 데 사용됩니다. `mapping` 필드는 사용할 매핑 테이블의 이름을 지정하며 `argument`는 매핑에 전달할 문자열을 지정합니다. 매핑은 존재해야 하며 성공적일 경우 해당 출력에 $Y 플래그를 설정해야 합니다. 매핑이 존재하지 않거나 $Y를 설정하지 않을 경우 다시 쓰기가 실패합니다. 성공적일 경우 매핑 결과가 현재 위치에서 템플리트에 병합되고 다시 확장됩니다.

이 기법을 통해 MTA 다시 쓰기 프로세스는 복잡한 여러 방법으로 확장될 수 있습니다. 예를 들어, 주소의 아이디 부분을 선택적으로 분석 및 수정할 수 있습니다(일반적으로 MTA 다시 쓰기 프로세스에서 불가능한 기능임).

11.6.7 사용자 제공 루틴 대체, $[...]

`$[`*image*`,`*routine*`,`*argument*`]` 형식의 대체는 사용자 제공 루틴을 검색 및 호출하는 데 사용됩니다. UNIX의 런타임에서 MTA는 `dlopen` 및 `dlsym`을 사용하여 공유 라이브러리 이미지에서 지정된 루틴을 동적으로 로드 및 호출합니다. 그런 다음 루틴은 다음 인수 목록과 함께 함수로 호출됩니다.

`status :=` *routine* `(`*argument*`,` *arglength*`,` *result*`,` *reslength*`)`

argument 및 *result*는 252바이트 길이의 문자열 버퍼입니다. UNIX에서 *argument* 및 *result*는 문자열에 대한 포인터로 전달됩니다(예: C에서는 `char*`로 전달됨). *arglength* 및 *reslength*는 참조에 의해 전달되는 서명된 긴 정수입니다. 입력 시에 *argument*는 다시 쓰기 규칙 템플리트의 인수 문자열을 포함하며 *arglength*는 해당 문자열의 길이를 포함합니다. 반환 시에 결과 문자열은 *result*에 포함되고 그 길이는 *reslength*에 포함되어야 합니다. 그런 다음 해당 결과 문자열은 다시 쓰기 규칙 템플리트의 "$[*image*,*routine*,*argument*]"를 대체합니다. 루틴은 다시 쓰기 규칙이 실패할 경우 0을 반환하고 성공할 경우 -1을 반환합니다.

이 기법을 통해 다시 쓰기 프로세스는 모든 종류의 복잡한 방법으로 확장될 수 있습니다. 예를 들어, 일부 유형의 이름 서비스에 대한 호출을 수행하고 그 결과를 사용하여 주소를 일정한 방식으로 변경할 수 있습니다. 호스트`siroe.com`에 대한 정방향 지정 주소(예: To:

주소)에 대해 다음 다시 쓰기 규칙을 사용하여 디렉토리 서비스 조회를 수행할 수 있습니다. 285 페이지 “11.6.12 방향 및 위치별 다시 쓰기 규칙($B, $E, $F, $R)”에 설명된 $F는 정방향 지정 주소에 대해서만 이 규칙이 사용되도록 합니다.

```
siroe.com $F$[LOOKUP_IMAGE,LOOKUP,$U]
```

정방향 지정 주소 jdoe@siroe.com은 이 다시 쓰기 규칙과 일치할 경우 LOOKUP_IMAGE(UNIX의 공유 라이브러리)가 메모리에 로드된 다음 LOOKUP 루틴이 인수 매개 변수로 jdoe와 함께 호출되게 합니다. 그런 다음 LOOKUP 루틴은 결과 매개 변수에서 다른 주소, 즉 John.Doe%eng.siroe.com과 다시 쓰기 규칙이 성공했음을 나타내는 값 -1을 반환할 수 있습니다. 결과 문자열의 백분율 기호(269 페이지 “11.4.2 반복되는 다시 쓰기 템플리트, A%B” 참조)는 John.Doe@eng.siroe.com을 다시 쓸 주소로 사용하여 다시 쓰기 프로세스가 다시 시작되게 합니다.

UNIX 시스템에서 사이트 제공 공유 라이브러리 이미지는 세계 공용이어야 합니다.

11.6.8 단일 필드 대체, $&, $!, $*, $#

단일 필드 대체는 다시 쓰는 호스트/도메인 지정에서 단일 하위 도메인 부분을 추출합니다. 표 11–6에는 사용 가능한 단일 필드 대체가 나와 있습니다.

표 11–6 단일 필드 대체

제어 시퀀스	사용
$&n	호스트 지정에서 n번째 요소(n=0,1,2,..,9)를 대체합니다(일정한 와일드카드와 일치하지 않았거나 일치한 부분). 요소는 점으로 구분되며 왼쪽의 첫 번째 요소가 요소 0입니다. 요청한 요소가 없으면 다시 쓰기가 실패합니다.
$!n	호스트 지정에서 n번째 요소(n=0,1,2,..,9)를 대체합니다(일정한 와일드카드와 일치하지 않았거나 일치한 부분). 요소는 점으로 구분되며 오른쪽의 첫 번째 요소가 요소 0입니다. 요청한 요소가 없으면 다시 쓰기가 실패합니다.
$*n	도메인 지정에서 n번째 요소(n=0,1,2,..,9)를 대체합니다(패턴의 명시적 텍스트와 일치하지 않은 부분). 요소는 점으로 구분되며 왼쪽의 첫 번째 요소가 요소 0입니다. 요청한 요소가 없으면 다시 쓰기가 실패합니다.
$#n	도메인 지정에서 n번째 요소(n=0,1,2,..,9)를 대체합니다(패턴의 명시적 텍스트와 일치하지 않은 부분). 요소는 점으로 구분되며 오른쪽의 첫 번째 요소가 요소 0입니다. 요청한 요소가 없으면 다시 쓰기가 실패합니다.

jdoe@eng.siroe.com 주소가 다음 다시 쓰기 규칙과 일치한다고 가정해 봅니다.

```
*.SIROE.COM      $U%$&0.siroe.com@mailhub.siroe.com
```

이 경우 템플리트의 결과는 jdoe@eng.siroe.com이 되며 mailhub.siroe.com이 라우팅 시스템으로 사용됩니다.

11.6.9 고유 문자열 대체

$W 제어 시퀀스를 사용할 때마다 대문자 및 숫자로 구성된 반복되지 않는 고유한 텍스트 문자열이 삽입됩니다. 반복되지 않는 주소 정보를 생성해야 할 경우에 $W가 유용합니다.

11.6.10 소스 채널별 다시 쓰기 규칙($M, $N)

특정 소스 채널에 대해서만 작동하는 다시 쓰기 규칙을 지정할 수 있습니다. 이것은 단순 형식의 이름이 다음과 같이 두 개의 의미를 가질 경우에 유용합니다.

1. 특정 채널에 도착하는 메시지에 나타날 경우
2. 다른 채널에 도착하는 메시지에 나타날 경우

소스 채널별 다시 쓰기는 사용 중인 채널 프로그램과 rules 및 norules 채널 키워드와 연관됩니다. 다시 쓰기를 수행 중인 MTA 구성 요소와 연관된 채널에서 norules가 지정된 경우 채널별 다시 쓰기 검사가 수행되지 않습니다. 이러한 채널에 rules가 지정된 경우에는 채널별 규칙 검사가 적용됩니다. rules 키워드가 기본값입니다.

소스 채널별 다시 쓰기는 주어진 주소와 일치하는 채널과 연관되지 않으며 다시 쓰기를 수행하는 MTA 구성 요소와 이 구성 요소의 채널 테이블 항목에만 의존합니다.

채널별 다시 쓰기 검사는 규칙의 템플리트 부분에 있는 $N 또는 $M 제어 시퀀스의 존재에 의해 트리거됩니다. $N 또는 $M 뒤에서부터 at 기호(@), 백분율 기호(%) 또는 후속 $N, $M, $Q, $C, $T 또는 $?의 앞 부분에 있는 문자는 채널 이름으로 해석됩니다.

예를 들어, $M*channel*의 경우 *channel*이 현재 다시 쓰기를 수행 중이 아니면 규칙이 실패하고 $N*channel*의 경우 *channel*이 다시 쓰기를 수행 중이면 규칙이 실패합니다. 여러 $M 및 $N 절을 지정할 수 있습니다. 여러 $M 절 중 하나가 일치하면 규칙이 성공하며 여러 $N 절 중 하나가 일치하면 규칙이 실패합니다.

11.6.11 대상 채널별 다시 쓰기 규칙($C, $Q)

메시지를 대기열에 넣는 채널에 의존하여 다시 쓰기 규칙이 적용될 수 있습니다. 이것은 일부 호스트에 대한 두 개의 이름, 즉 특정 호스트 그룹에 알려진 이름과 다른 호스트 그룹에 알려진 이름이 존재할 경우에 유용합니다. 다른 채널을 사용하여 메시지를 각 그룹에 보내면 주소를 다시 작성하여 각 그룹에 알려진 이름으로 호스트를 참조할 수 있습니다.

대상 채널별 다시 쓰기는 메시지를 대기열에서 빼고 처리하는 채널과 해당 채널의 rules 및 norules 채널 키워드와 연관됩니다. 대상 채널에 norules가 지정된 경우 채널별 다시 쓰기 검사가 수행되지 않습니다. 대상 채널에 rules가 지정된 경우 채널별 규칙 검사가 적용됩니다. rules 키워드가 기본값입니다.

대상 채널별 다시 쓰기는 주어진 주소와 일치하는 채널과 연관되지 않습니다. 대상 채널별 다시 쓰기는 메시지의 봉투의 To: 주소에만 의존합니다. 메시지를 대기열에 넣을 때 대상 채널을 결정하기 위해 메시지의 envolope To: 주소가 우선적으로 다시

작성됩니다. envelope To: 주소를 다시 쓰는 동안 $C 및 $Q 제어 시퀀스는 무시됩니다. envelope To: 주소가 다시 작성되고 대상 채널이 결정되면 메시지와 연관된 다른 주소가 다시 작성될 때 $C 및 $Q 제어 시퀀스가 적용됩니다.

대상 채널별 다시 쓰기 검사는 규칙의 템플리트 부분에 있는 $C 또는 $Q 제어 시퀀스의 존재에 의해 트리거됩니다. $C 또는 $Q 뒤에서부터 at 기호(@), 백분율 기호(%) 또는 후속 $N, $M, $C, $Q, $T 또는 $?의 앞 부분에 있는 문자는 채널 이름으로 해석됩니다.

예를 들어, $Q*channel*의 경우 *channel*이 대상이 아니면 규칙이 실패하고 $C*channel*의 경우 *channel*이 대상이면 규칙이 실패합니다. 여러 $Q 및 $C 절을 지정할 수 있습니다. 여러 $Q 절 중 하나가 일치하면 규칙이 성공하며 여러 $C 절 중 하나가 일치하면 규칙이 실패합니다.

11.6.12 방향 및 위치별 다시 쓰기 규칙($B, $E, $F, $R)

경우에 따라 봉투 주소 또는 헤더 주소에만 적용되는 다시 쓰기 규칙을 지정해야 합니다. 제어 시퀀스 $E는 다시 쓰는 주소가 봉투 주소가 아닐 경우 다시 쓰기가 실패하도록 지정하며 $B는 다시 쓰는 주소가 메시지 헤더나 본문에서 오지 않은 경우 다시 쓰기가 실패하도록 지정합니다. 이러한 시퀀스는 다시 쓰기에 다른 영향을 미치지 않으며 다시 쓰기 규칙 템플리트의 임의 위치에 나타날 수 있습니다.

주소를 또한 방향별로 범주화할 수 있습니다. 정방향 지정 주소는 To:, Cc:, Resent-to: 또는 대상을 나타내는 다른 헤더 또는 봉투 행에서 시작되는 주소입니다. 역방향 지정 주소는 소스를 나타내는 From:, Sender: 또는 Resent-From:과 같은 주소입니다. 제어 시퀀스 $F는 주소가 정방향 지정일 경우에 다시 쓰기를 적용하도록 지정하며 $R은 주소가 역방향 지정일 경우에 다시 쓰기를 적용하도록 지정합니다.

11.6.13 호스트 위치별 다시 쓰기($A, $P, $S, $X)

경우에 따라 주소에서 호스트 이름이 나타나는 위치가 중요한 다시 쓰기가 필요할 수 있습니다. 호스트 이름은 다음과 같이 주소의 여러 다른 컨텍스트에 나타날 수 있습니다.

- 소스 경로 내
- at 기호(@)의 오른쪽
- 로컬 부분에 있는 백분율 기호(%)의 오른쪽
- 로컬 부분에 있는 느낌표의 왼쪽

정상적인 경우라면 호스트 이름은 표시 위치에 상관 없이 동일하게 처리되어야 합니다. 일부 상황에서는 특수한 처리가 필요할 수 있습니다.

주소에서의 호스트 위치에 기초한 일치를 제어하기 위해 네 개의 제어 시퀀스가 사용됩니다.

- $S는 규칙이 소스 경로에서 추출된 호스트와 일치할 수 있도록 지정합니다.
- $A는 규칙이 @ 기호의 오른쪽에 있는 호스트와 일치할 수 있도록 지정합니다.

- $P는 규칙이 % 기호의 오른쪽에 있는 호스트와 일치할 수 있도록 지정합니다.
- $X는 규칙이 느낌표(!)의 왼쪽에 있는 호스트와 일치할 수 있도록 지정합니다.

호스트가 지정된 위치와 다른 위치에 있을 경우 규칙은 실패합니다. 이러한 시퀀스는 하나의 다시 쓰기 규칙에서 결합될 수 있습니다. 예를 들어, $S 및 $A가 지정된 경우 규칙은 소스 경로나 at 기호의 오른쪽에 지정된 호스트와 일치합니다. 이러한 시퀀스를 전혀 지정하지 않는 것은 시퀀스를 모두 지정되는 것과 같습니다. 즉, 위치에 상관 없이 규칙이 일치할 수 있습니다.

11.6.14 현재 태그 값 변경, $T

$T 제어 시퀀스는 현재 다시 쓰기 규칙 태그를 변경하는 데 사용됩니다. 구성 파일과 도메인 데이터베이스에서 다시 쓰기 규칙 패턴이 조회되기 전에 다시 쓰기 규칙 태그가 모든 다시 쓰기 규칙 패턴의 앞에 놓입니다. $T의 뒤에서부터 at 기호, 백분율 기호, $N, $M, $Q, $C, $T 또는 $?의 앞 부분에 있는 텍스트는 새 태그로 간주됩니다.

태그는 특정 구성 요소가 발견되었을 때 주소의 전체 특성이 변경되는 특수한 주소 지정 형식을 처리하는 데 유용합니다. 예를 들어, 소스 경로에서 발견될 경우 특정 호스트 이름 internet을 주소에서 제거해야 하며 TCP-DAEMON 채널에 대해 결과 주소를 강제로 일치시켜야 한다고 가정해 봅니다.

이것은 다음과 같은 규칙으로 구현할 수 있습니다(localhost가 로컬 호스트의 공식 이름이라고 가정함).

```
internet              $S$U@localhost$Tmtcp-force|

mtcp-force|.          $U%$H@TCP-DAEMON
```

첫 번째 규칙은 특정 호스트 이름 internet과 일치합니다(소스 경로에 표시될 경우). 이 규칙은 internet을 로컬 채널에 대해 강제로 일치시켜 주소에서 제거되도록 합니다. 그런 다음 다시 쓰기 태그가 설정됩니다. 다시 쓰기가 진행되지만 태그로 인해 일반 규칙은 일치하지 않습니다. 마지막으로 태그와 함께 기본 규칙이 시도되고 이 집합의 두 번째 규칙이 적용되어 다른 조건에 상관 없이 TCP-DAEMON 채널에 대해 주소를 강제로 일치시킵니다.

11.6.15 다시 쓰기와 관련된 오류 메시지 제어($?)

MTA는 다시 쓰기 및 채널 일치가 실패할 경우 기본 오류 메시지를 제공합니다. 특별한 상황에서는 이러한 메시지를 변경하는 기능이 유용할 수 있습니다. 예를 들어, 누군가가 이더넷 라우터 상자에 메일을 보내려고 시도할 경우 일반적인 "잘못된 호스트/도메인이 지정됨"라는 메시지보다 "라우터가 메일을 수락할 수 없음"이라는 메시지가 더 정확하게 의미를 표시합니다.

특수한 제어 시퀀스를 사용하여 규칙이 실패할 경우 인쇄할 오류 메시지를 변경할 수 있습니다. $? 시퀀스는 오류 메시지를 지정하는 데 사용됩니다. $?의 뒤에서부터 at 기호(@), 백분율 기호(%), $N, $M, $Q, $C, $T 또는 $?의 앞 부분에 있는 텍스트는 다시 쓰기의 결과가 임의의 채널과 일치하는 데 실패할 경우 인쇄될 오류 메시지의 텍스트로 간주됩니다. 오류 메시지의 설정은 "고정적"이며 다시 쓰기 프로세스 동안 계속 적용됩니다.

$?를 포함하는 규칙은 다른 규칙과 마찬가지로 작동합니다. $?만 포함하는 특수한 규칙의 경우 다른 방식으로 처리됩니다. 즉, 주소의 메일함 또는 호스트 부분을 변경하지 않고 다시 쓰기 프로세스가 종료하며 호스트는 채널 테이블에서 있는 그대로 조회됩니다. 이 조회는 실패할 것이며 이에 따라 오류 메시지가 결과로 반환됩니다.

예를 들어, MTA 구성 파일의 최종 다시 쓰기 규칙이 다음과 같다고 가정해 봅니다.

```
. $?Unrecognized address; contact postmaster@siroe.com
```

이 예에서 실패할 수 있는 인식되지 않은 호스트 또는 도메인 지정은 실패하는 과정에서 오류 메시지 `Unrecognized address; contact postmaster@siroe.com`을 생성합니다.

11.7 많은 수의 다시 쓰기 규칙 처리

MTA는 항상 `imta.cnf` 파일에서 모든 다시 쓰기 규칙을 읽어 해시 테이블의 메모리에 저장합니다. 컴파일된 구성을 사용하면 정보가 필요할 때마다 구성 파일을 읽는 것과 관련된 오버헤드가 사라집니다. 해시 테이블은 여전히 모든 다시 쓰기 규칙을 메모리에 저장하는 데 사용됩니다. 이 방법은 다시 쓰기 규칙 수가 적거나 보통인 경우에 적합합니다. 그러나 일부 사이트에는 10,000개 이상의 다시 쓰기 규칙이 필요할 수 있으며 이로 인해 과도한 양의 메모리가 소비될 수 있습니다.

MTA는 보조 색인 데이터 파일에 많은 수의 다시 쓰기 규칙을 저장하기 위한 선택적 기능을 제공하여 이 문제를 해결합니다. 일반 구성 파일을 읽을 때마다 MTA는 도메인 데이터베이스의 존재를 확인합니다. 도메인 데이터베이스가 존재할 경우 구성 파일에서 발견된 규칙에서 시도된 일치가 실패할 때마다 이 데이터베이스가 열리고 참조됩니다. 구성 파일에서 특정 규칙이 발견되지 않을 경우에만 도메인 데이터베이스가 검사되므로 규칙을 항상 구성 파일에 추가하여 데이터베이스에서 이를 무시할 수 있습니다. 기본적으로 도메인 데이터베이스는 호스트 도메인과 연관된 다시 쓰기 규칙을 저장하는 데 사용됩니다. IMTA_DOMAIN_DATABASE 속성은 `imta_tailor` 파일에 저장됩니다. 데이터베이스의 기본 위치는 *msg-svr-base*`/data/db/domaindb.db`입니다.

주 – 이 필드를 수동으로 편집해서는 안 됩니다.

11.8 다시 쓰기 규칙 테스트

imsimta test -rewrite 명령을 사용하여 다시 쓰기 규칙을 테스트할 수 있습니다. -noimage 한정자를 사용하면 새 구성을 다시 컴파일하기 전에 구성 파일의 변경 사항을 테스트할 수 있습니다.

이 유틸리티를 -debug 한정자와 함께 사용하여 몇 개의 주소를 다시 쓰는 것이 유용할 수 있습니다. 이 작업은 주소를 다시 쓰는 방법을 단계별로 보여 줍니다. 예를 들어, 다음 명령을 실행합니다.

```
% imsimta test -rewrite -debug joe@siroe.com
```

imsimta test -rewrite 유틸리티에 대한 자세한 내용은 **Sun Java System Messaging Server 6.3 Administration Reference**를 참조하십시오.

11.9 다시 쓰기 규칙 예

다음 예는 샘플 다시 쓰기 규칙과 샘플 주소가 이러한 규칙에 의해 다시 작성되는 방법을 보여 줍니다.

다음 예에 나온 다시 쓰기 규칙이 시스템 SC.CS.SIROE.EDU에 대한 구성 파일에 포함되었다고 가정해 봅니다.

```
sc                      $U@sc.cs.siroe.edu
sc1                     $U@sc1.cs.siroe.edu
sc2                     $U@sc2.cs.siroe.edu
*                       $U%$&0.cs.siroe.edu
*.cs                    $U%$&0.cs.siroe.edu
*.cs.siroe              $U%$&0.cs.siroe.edu
*.cs.siroe.edu          $U%$&0.cs.siroe.edu@ds.adm.siroe.edu
sc.cs.siroe.edu         $U@$D
sc1.cs.siroe.edu        $U@$D
sc2.cs.siroe.edu        $U@$D
sd.cs.siroe.edu         $U@sd.cs.siroe.edu
.siroe.edu              $U%$H.siroe.edu@cds.adm.siroe.edu
.edu                    $U@$H$D@gate.adm.siroe.edu
[]                      $U@[$L]@gate.adm.siroe.edu
```

표 11–7은 몇 가지 샘플 주소와 다시 쓰기 규칙에 따라 이러한 주소가 다시 작성 및 라우팅되는 방법을 보여 줍니다.

표 11–7 샘플 주소 및 다시 쓰기

초기 주소	다시 작성된 주소	라우팅 대상
user@sc	user@sc.cs.siroe.edu	sc.cs.siroe.edu
user@sc1	user@sc1.cs.siroe.edu	sc1.cs.siroe.edu
user@sc2	user@sc2.cs.siroe.edu	sc2.cs.siroe.edu
user@sc.cs	user@sc.cs.siroe.edu	sc.cs.siroe.edu
user@sc1.cs	user@sc1.cs.siroe.edu	sc1.cs.siroe.edu
user@sc2.cs	user@sc2.cs.siroe.edu	sc2.cs.siroe.edu
user@sc.cs.siroe	user@sc.cs.siroe.edu	sc.cs.siroe.edu
user@sc1.cs.siroe	user@sc1.cs.siroe.edu	sc1.cs.siroe.edu
user@sc2.cs.siroe	user@sc2.cs.siroe.edu	sc2.cs.siroe.edu
user@sc.cs.siroe.edu	user@sc.cs.siroe.edu	sc.cs.siroe.edu
user@sc1.cs.siroe.edu	user@sc1.cs.siroe.edu	sc1.cs.siroe.edu
user@sc2.cs.siroe.edu	user@sc2.cs.siroe.edu	sc2.cs.siroe.edu
user@sd.cs.siroe.edu	user@sd.cs.siroe.edu	sd.cs.siroe.edu
user@aa.cs.siroe.edu	user@aa.cs.siroe.edu	ds.adm.siroe.edu
user@a.eng.siroe.edu	user@a.eng.siroe.edu	cds.adm.siroe.edu
user@a.cs.sesta.edu	user@a.cs.sesta.edu	gate.adm.siroe.edu—route inserted
user@b.cs.sesta.edu	user@b.cs.sesta.edu	gate.adm.siroe.edu— route inserted
user@[1.2.3.4]	user@[1.2.3.4]	gate.adm.siroe.edu—route inserted

기본적으로, 이러한 다시 쓰기 규칙의 내용을 살펴보면 우선호스트 이름이 단순 형식 이름(`sc`, `sc1` 또는 `sc2`) 중 하나이거나 전체 이름(`sc.cs.siroe.edu` 등) 중 하나일 경우 이를 전체 이름으로 확장하고 대상으로 라우팅합니다. 한 부분으로 된 단순 형식 이름에 `cs.cmu.edu`를 추가하고 다시 시도합니다. 뒤에 `.cs`가 오는 한 부분을 뒤에 `.cs.siroe.edu`가 오는 한 부분으로 변환하고 다시 시도합니다. 또한 `.cs.siroe`를 `.cs.siroe.edu`로 변환하고 다시 시도합니다.

이름이 `sd.cs.siroe.edu`(직접 연결되는 일부 시스템)인 경우 이를 다시 작성하고 자체에서 라우팅합니다. 호스트 이름이 `.cs.siroe.edu` 하위 도메인에 있는 것이면 이를 `ds.cs.siroe.edu`(`.cs.siroe.edu` 하위 도메인의 게이트웨이)로 라우팅합니다. 호스트 이름이 `.siroe.edu` 하위 도메인에 있는 것이면 이를 `cds.adm.siroe.edu`(`.siroe.edu` 하위 도메인의 게이트웨이)로 라우팅합니다. 호스트 이름이 `.edu` 최상위 도메인 있는 것이면 이를 `gate.adm.siroe.edu`(메시지를 적절한 대상으로 라우팅할 수 있음)로 라우팅합니다. 도메인 리터럴이 사용될 경우 이를 또한 `gate.adm.siroe.edu`로 보냅니다.

앞의 예와 마찬가지로 대부분의 다시 쓰기 규칙 적용은 주소의 아이디 또는 메일함 부분을 변경하지 않습니다. 주소의 아이디 부분을 변경하는 기능은 RFC 822를 따르지 않는 전자 메일 프로그램(즉, 호스트/도메인 지정의 일부로 주소의 아이디 부분을 채워야 하는 전자 메일 프로그램)과 인터페이스하기 위해 MTA를 사용할 때 사용됩니다. 이 기능을 사용할 때는 매우 주의해야 합니다.

◆ ◆ ◆ 12 장

12 채널 정의 구성

이 장에서는 MTA 구성 파일 `imta.cnf`에서 채널 키워드 정의를 사용하는 방법에 대해 설명합니다. 이 장을 읽기 전에 10 장, 177 페이지 “8.5.3 채널 정의” 및 211 페이지 “10.2 MTA 구성 파일”을 읽어 보십시오. 이 장은 다음 내용으로 구성되어 있습니다.

주 – `imta.cnf`에서 채널 정의를 변경하는 경우 `imsimta restart` 명령을 사용하여 시작할 때 한 번만 구성 데이터를 로드하는 프로그램 또는 채널(예: SMTP 서버)을 다시 시작해야 합니다. 컴파일된 구성을 사용하는 경우 재컴파일을 수행한 다음 다시 시작해야 합니다. 구성 정보 컴파일과 프로그램 시작에 대한 자세한 내용은 **Sun Java System Messaging Server 6.3 Administration Reference**를 참조하십시오.

12.1 채널 기본값 구성

많은 구성은 모든 채널 또는 거의 모든 채널에서 다양한 채널 키워드 반복을 포함합니다. 그러한 구성의 유지 관리는 귀찮고 오류가 발생하기 쉽습니다. 일부 구성을 단순화하기 위해 다양한 채널의 기본값으로 사용할 키워드를 지정할 수 있습니다.

예를 들어, 구성 파일의 다음 행은 해당 행 다음에 있는 모든 채널 블록이 해당 행에 지정된 키워드를 상속함을 나타냅니다.

```
defaults keyword1 keyword2 keyword3 ...
```

defaults 행은 실제로 채널을 지정하지 않은 채 키워드 기본값을 변경하는 특수 채널 블록으로 간주할 수 있습니다. 또한, defaults 행에는 추가 채널 블록 정보 행이 필요하지 않습니다(무시하도록 지정된 경우).

지정 가능한 defaults 행 수에 대한 제한이 없습니다. 즉, 여러 기본값 행의 효과가 최근에 발생한 앞 행(위에서 아래로 읽음)에 누적됩니다.

구성 파일의 특정 지점(예: 외부 파일에 있는 채널 블록의 독립형 섹션의 시작 부분)에서 시작하여 defaults 행의 효과를 무조건적으로 제거하는 것이 좋을 수도 있습니다. nodefaults 행이 이러한 목적으로 제공됩니다. 예를 들어, 구성 파일에 다음 행을 삽입하면 이전의 기본값 채널에서 지정된 모든 설정이 무효화되고 구성이 기본값을 지정하지 않은 경우에 적용되는 상태로 돌아갑니다.

```
nodefaults
```

일반 채널 블록과 마찬가지로 빈 행을 사용하여 각 defaults 또는 nodefaults 채널 블록을 다른 채널 블록과 구분해야 합니다. defaults 및 nodefaults 채널 블록은 구성 파일에서 로컬 채널 앞에 표시될 수 있는 유일한 채널 블록입니다. 그러나, 다른 채널 블록과 마찬가지로 이 채널 블록도 마지막 다시 쓰기 규칙 뒤에 표시해야 합니다.

12.2 채널 키워드(알파벳순)

다음 표는 알파벳순으로 나열된 키워드 목록입니다.

표 12–1 채널 키워드(알파벳순)

키워드	추가 정보
733	361 페이지 “12.6.1 주소 유형 및 규칙”
822	361 페이지 “12.6.1 주소 유형 및 규칙”
addreturnpath	367 페이지 “12.6.11 Return-path: 헤더 행 생성”

표 12–1 채널 키워드(알파벳순) *(계속)*

키워드	추가 정보
addresssrs	486 페이지 “15.7 SRS(Sender Rewriting Scheme)를 사용하여 SPF에서 전달된 메일 처리”
addrsperfile	384 페이지 “12.9.4 할당량이 초과된 사용자에 대한 메일 전달 처리”
Aliasdetourhost	391 페이지 “12.12.6 주소 검증 후와 확장 전의 라우팅”
aliaslocal	369 페이지 “12.6.15 별칭 파일 및 별칭 데이터베이스 검사 지정”
aliaspostmaster	255 페이지 “포스트마스터에게 반환되는 메일 내용”
allowetrn	337 페이지 “12.4.2.3 ETRN 명령 지원”
allowswitchchannel	347 페이지 “12.4.3.8 받는 메일을 위한 대체 채널(전환 채널)”
alternatechannel	382 페이지 “12.9.3 크기 제한 또는 수신자 수 제한을 초과하는 메시지 대상 다시 지정”
alternateblocklimit	382 페이지 “12.9.3 크기 제한 또는 수신자 수 제한을 초과하는 메시지 대상 다시 지정”
alternatelinelimit	382 페이지 “12.9.3 크기 제한 또는 수신자 수 제한을 초과하는 메시지 대상 다시 지정”
alternaterecipientlimit	382 페이지 “12.9.3 크기 제한 또는 수신자 수 제한을 초과하는 메시지 대상 다시 지정”
authrewrite	341 페이지 “12.4.3 TCP/IP 연결 및 DNS 조회 지원”
backoff	355 페이지 “12.5.3 전달에 실패한 메시지에 대한 재시도 간격 지정”
bangoverpercent	363 페이지 “12.6.3 주소에 라우팅 정보 추가”
bangstyle	361 페이지 “12.6.1 주소 유형 및 규칙”
bidirectional	354 페이지 “12.5.1 채널 방향 설정”
blocketrn	337 페이지 “12.4.2.3 ETRN 명령 지원”
blocklimit	382 페이지 “12.9.2 절대 메시지 크기 제한 지정”
cacheeverything	343 페이지 “12.4.3.2 채널 연결 정보 캐싱”
cachefailures	343 페이지 “12.4.3.2 채널 연결 정보 캐싱”
cachesuccesses	343 페이지 “12.4.3.2 채널 연결 정보 캐싱”
caption	395 페이지 “12.12.9 모니터링 프레임워크에 대한 채널 화면 표시 설정”
channelfilter	390 페이지 “12.12.4 메일함 필터 파일 위치 지정”
charset7	339 페이지 “12.4.2.7 문자 세트 레이블링 및 8비트 데이터”
charset8	339 페이지 “12.4.2.7 문자 세트 레이블링 및 8비트 데이터”

표 12–1 채널 키워드(알파벳순) (계속)

키워드	추가 정보
charsetesc	339 페이지 “12.4.2.7 문자 세트 레이블링 및 8비트 데이터”
checkehlo	336 페이지 “12.4.2.2 EHLO 명령 지원”
chunkingclient	350 페이지 “12.4.6 SMTP 청크 지원”
chunkingserver	350 페이지 “12.4.6 SMTP 청크 지원”
commentinc	368 페이지 “12.6.13 주소 헤더 행의 주석 처리”
commentmap	368 페이지 “12.6.13 주소 헤더 행의 주석 처리”
commentomit	368 페이지 “12.6.13 주소 헤더 행의 주석 처리”
commentstrip	368 페이지 “12.6.13 주소 헤더 행의 주석 처리”
commenttotal	368 페이지 “12.6.13 주소 헤더 행의 주석 처리”
connectalias	364 페이지 “12.6.5 메시지를 대기열에서 제거할 때 주소 다시 쓰기”
connectcanonical	364 페이지 “12.6.5 메시지를 대기열에서 제거할 때 주소 다시 쓰기”
copysendpost	254 페이지 “반환되는 실패 메일”
copywarnpost	254 페이지 “경고 메일”
daemon	348 페이지 “12.4.3.10 대상 호스트 선택”
datefour	374 페이지 “12.7.4 두 자리 또는 네 자리로 날짜 변환”
datetwo	374 페이지 “12.7.4 두 자리 또는 네 자리로 날짜 변환”
dayofweek	374 페이지 “12.7.5 날짜의 요일 지정”
defaulthost	364 페이지 “12.6.6 불완전한 주소를 수정할 때 사용할 호스트 이름 지정”
defaultmx	346 페이지 “12.4.3.5 TCP/IP MX 레코드 지원”
defaultnameservers	346 페이지 “12.4.3.6 이름 서버 조회”
deferralrejectlimit	395 페이지 “12.12.8 잘못된 RCPT TO: 주소에 대한 제한 설정 ”
deferred	354 페이지 “12.5.2 지연 전달 날짜 구현”
defragment	377 페이지 “12.8.2 메시지/부분 메시지 자동 조각 모음”
dequeue_removeroute	371 페이지 “12.6.18 소스 경로 제거”
description	395 페이지 “12.12.9 모니터링 프레임워크에 대한 채널 화면 표시 설정”
destinationfilter	390 페이지 “12.12.4 메일함 필터 파일 위치 지정”
destinationnosolicit	395 페이지 “12.12.7 NO-SOLICIT SMTP 확장 지원”
destinationspamfilterX	390 페이지 “12.12.5 스팸 필터 키워드”

표 12–1 채널 키워드(알파벳순) *(계속)*

키워드	추가 정보
destinationspamfilterXoptin	390 페이지 “12.12.5 스팸 필터 키워드”
destinationsrs	486 페이지 “15.7 SRS(Sender Rewriting Scheme)를 사용하여 SPF에서 전달된 메일 처리”
disabledestinationspamfilterX	390 페이지 “12.12.5 스팸 필터 키워드”
disableetrn	337 페이지 “12.4.2.3 ETRN 명령 지원”
disablesourcespamfilterX	390 페이지 “12.12.5 스팸 필터 키워드”
dispositionchannel	389 페이지 “12.12.1 프로세스 채널 대체”
disconnectbadauthlimit	381 페이지 “12.9.1 성공하지 못한 인증 시도에 대한 제한”
disconnectbadcommandlimit	387 페이지 “12.10.3 세션 제한 설정”
domainetrn	337 페이지 “12.4.2.3 ETRN 명령 지원”
domainvrfy	338 페이지 “12.4.2.4 VRFY 명령 지원”
dropblank	366 페이지 “12.6.8 잘못된 빈 수신자 헤더 스트라이핑”
ehlo	336 페이지 “12.4.2.2 EHLO 명령 지원”
eightbit	339 페이지 “12.4.2.7 문자 세트 레이블링 및 8비트 데이터”
eightnegotiate	339 페이지 “12.4.2.7 문자 세트 레이블링 및 8비트 데이터”
eightstrict	339 페이지 “12.4.2.7 문자 세트 레이블링 및 8비트 데이터”
errsendpost	254 페이지 “반환되는 실패 메일”
errwarnpost	254 페이지 “경고 메일”
expandchannel	359 페이지 “12.5.9 여러 주소 확장”
expandlimit	359 페이지 “12.5.9 여러 주소 확장”
expnallow	338 페이지 “12.4.2.5 EXPN 지원”
expndisable	338 페이지 “12.4.2.5 EXPN 지원”
expndefault	338 페이지 “12.4.2.5 EXPN 지원”
exproute	363 페이지 “12.6.3 주소에 라우팅 정보 추가”
fileinto	390 페이지 “12.12.4 메일함 필터 파일 위치 지정”
filesperjob	356 페이지 “12.5.5 서비스 작업 제한”
filter	390 페이지 “12.12.4 메일함 필터 파일 위치 지정”
forwardcheckdelete	344 페이지 “12.4.3.3 역방향 DNS 조회”

표 12–1 채널 키워드(알파벳순) (계속)

키워드	추가 정보
forwardchecknone	344 페이지 “12.4.3.3 역방향 DNS 조회”
forwardchecktag	344 페이지 “12.4.3.3 역방향 DNS 조회”
header_733	361 페이지 “12.6.1 주소 유형 및 규칙”
header_822	361 페이지 “12.6.1 주소 유형 및 규칙”
header_uucp	361 페이지 “12.6.1 주소 유형 및 규칙”
headerlabelalign	375 페이지 “12.7.7 헤더 맞춤 및 접기”
headerlimit	385 페이지 “12.9.8 헤더 크기 제한”
headerlinelength	375 페이지 “12.7.7 헤더 맞춤 및 접기”
headerread	372 페이지 “12.7.2 선택한 메시지 헤더 행 제거”
headertrim	372 페이지 “12.7.2 선택한 메시지 헤더 행 제거”
holdexquota	384 페이지 “12.9.4 할당량이 초과된 사용자에 대한 메일 전달 처리”
holdlimit	359 페이지 “12.5.9 여러 주소 확장”
identnone	344 페이지 “12.4.3.4 IDENT 조회”
identnonelimited	344 페이지 “12.4.3.4 IDENT 조회”
identnonenumeric	344 페이지 “12.4.3.4 IDENT 조회”
identnonesymbolic	344 페이지 “12.4.3.4 IDENT 조회”
identtcp	344 페이지 “12.4.3.4 IDENT 조회”
identtcplimited	344 페이지 “12.4.3.4 IDENT 조회”
identtcpsymbolic	344 페이지 “12.4.3.4 IDENT 조회”
ignoreencoding	377 페이지 “12.8.1 Encoding 헤더 행 무시”
ignoremessageencoding	381 페이지 “12.8.5 멀티파트 및 Message/RFC822 부분의 cContent-transfer-encoding 필드 해석”
ignoremultipartencoding	381 페이지 “12.8.5 멀티파트 및 Message/RFC822 부분의 cContent-transfer-encoding 필드 해석”
immnonurgent	354 페이지 “12.5.2 지연 전달 날짜 구현”
improute	363 페이지 “12.6.3 주소에 라우팅 정보 추가”
includefinal	253 페이지 “10.10.4.4 상태 알림 메일에 변경된 주소 포함”
inenttcpnumeric	344 페이지 “12.4.3.4 IDENT 조회”

표 12–1 채널 키워드(알파벳순) *(계속)*

키워드	추가 정보
inner	372 페이지 “12.7.1 포함 헤더 다시 쓰기”
innertrim	372 페이지 “12.7.2 선택한 메시지 헤더 행 제거”
interfaceaddress	343 페이지 “12.4.3.1 TCP/IP 포트 번호 및 인터페이스 주소”
interpretencoding	377 페이지 “12.8.1 Encoding 헤더 행 무시”
interpretmessageencoding	381 페이지 “12.8.5 멀티파트 및 Message/RFC822 부분의 cContent-transfer-encoding 필드 해석”
interpretmultipartencoding	381 페이지 “12.8.5 멀티파트 및 Message/RFC822 부분의 cContent-transfer-encoding 필드 해석”
language	376 페이지 “12.7.10 헤더의 기본 언어 설정”
lastresort	346 페이지 “12.4.3.7 마지막 Resort 호스트”
linelength	380 페이지 “12.8.4 메시지 행 길이 제한 적용”
linelimit	382 페이지 “12.9.2 절대 메시지 크기 제한 지정”
localvrfy	338 페이지 “12.4.2.4 VRFY 명령 지원”
logging	387 페이지 “12.11.1 로깅 키워드”
logheader	387 페이지 “12.11.1 로깅 키워드”
loopcheck	388 페이지 “12.11.3 Loopcheck 설정”
mailfromdnsverify	339 페이지 “12.4.2.6 DNS 도메인 확인”
master	354 페이지 “12.5.1 채널 방향 설정”
master_debug	388 페이지 “12.11.2 디버깅 키워드”
maxblocks	379 페이지 “12.8.3 대용량 메시지 자동 조각화”
maxheaderaddrs	374 페이지 “12.7.6 긴 헤더 행 자동 분할”
maxheaderchars	374 페이지 “12.7.6 긴 헤더 행 자동 분할”
maxjobs	356 페이지 “12.5.5 서비스 작업 제한”
maxlines	379 페이지 “12.8.3 대용량 메시지 자동 조각화”
maxprocchars	375 페이지 “12.7.7 헤더 맞춤 및 접기”
maysaslserver	349 페이지 “12.4.4 SMTP 인증, SASL 및 TLS”
maytls	351 페이지 “12.4.8 전송 계층 보안”
maytlsclient	351 페이지 “12.4.8 전송 계층 보안”

표 12–1 채널 키워드(알파벳순) (계속)

키워드	추가 정보
maytlsserver	351 페이지 “12.4.8 전송 계층 보안”
missingrecipientpolicy	365 페이지 “12.6.7 수신자 헤더 행 없이 메시지 적법화”
msexchange	351 페이지 “12.4.7 Microsoft Exchange 게이트웨이 채널 지정”
multiple	384 페이지 “12.9.4 할당량이 초과된 사용자에 대한 메일 전달 처리”
mustsaslserver	349 페이지 “12.4.4 SMTP 인증, SASL 및 TLS”
musttls	351 페이지 “12.4.8 전송 계층 보안”
musttlsclient	351 페이지 “12.4.8 전송 계층 보안”
musttlsserver	351 페이지 “12.4.8 전송 계층 보안”
mx	346 페이지 “12.4.3.5 TCP/IP MX 레코드 지원”
namelengthlimit	385 페이지 “12.9.6 일반 및 Filename Content-type 및 Content-disposition 매개변수의 길이 제어”
nameservers	346 페이지 “12.4.3.6 이름 서버 조회”
noaddresssrs	486 페이지 “15.7 SRS(Sender Rewriting Scheme)를 사용하여 SPF에서 전달된 메일 처리”
noaddreturnpath	367 페이지 “12.6.11 Return-path: 헤더 행 생성”
nobangoverpercent	363 페이지 “12.6.3 주소에 라우팅 정보 추가”
noblocklimit	382 페이지 “12.9.2 절대 메시지 크기 제한 지정”
nocache	343 페이지 “12.4.3.2 채널 연결 정보 캐싱”
nochannelfilter	390 페이지 “12.12.4 메일함 필터 파일 위치 지정”
nochunkingclient	350 페이지 “12.4.6 SMTP 청크 지원”
nochunkingserver	350 페이지 “12.4.6 SMTP 청크 지원”
nodayofweek	374 페이지 “12.7.5 날짜의 요일 지정”
nodefaulthost	364 페이지 “12.6.6 불완전한 주소를 수정할 때 사용할 호스트 이름 지정”
nodeferred	354 페이지 “12.5.2 지연 전달 날짜 구현”
nodefragment	377 페이지 “12.8.2 메시지/부분 메시지 자동 조각 모음”
nodestinationfilter	390 페이지 “12.12.4 메일함 필터 파일 위치 지정”
nodestinationsrs	486 페이지 “15.7 SRS(Sender Rewriting Scheme)를 사용하여 SPF에서 전달된 메일 처리”
nodropblank	366 페이지 “12.6.8 잘못된 빈 수신자 헤더 스트라이핑”

표 12–1 채널 키워드(알파벳순) *(계속)*

키워드	추가 정보
noehlo	336 페이지 “12.4.2.2 EHLO 명령 지원”
noexproute	363 페이지 “12.6.3 주소에 라우팅 정보 추가”
noexquota	384 페이지 “12.9.4 할당량이 초과된 사용자에 대한 메일 전달 처리”
nofileinto	390 페이지 “12.12.4 메일함 필터 파일 위치 지정”
nofilter	390 페이지 “12.12.4 메일함 필터 파일 위치 지정”
noheaderread	372 페이지 “12.7.2 선택한 메시지 헤더 행 제거”
noheadertrim	372 페이지 “12.7.2 선택한 메시지 헤더 행 제거”
noimproute	363 페이지 “12.6.3 주소에 라우팅 정보 추가”
noinner	372 페이지 “12.7.1 포함 헤더 다시 쓰기”
noinnertrim	372 페이지 “12.7.2 선택한 메시지 헤더 행 제거”
nolinelimit	382 페이지 “12.9.2 절대 메시지 크기 제한 지정”
nologging	387 페이지 “12.11.1 로깅 키워드”
noloopcheck	388 페이지 “12.11.3 Loopcheck 설정”
nomailfromdnsverify	339 페이지 “12.4.2.6 DNS 도메인 확인”
nomaster_debug	388 페이지 “12.11.2 디버깅 키워드”
nomsexchange	341 페이지 “12.4.3 TCP/IP 연결 및 DNS 조회 지원”
nomx	346 페이지 “12.4.3.5 TCP/IP MX 레코드 지원”
norandomemx	346 페이지 “12.4.3.5 TCP/IP MX 레코드 지원”
nosourcesrs	486 페이지 “15.7 SRS(Sender Rewriting Scheme)를 사용하여 SPF에서 전달된 메일 처리”
nonurgentbackoff	355 페이지 “12.5.3 전달에 실패한 메시지에 대한 재시도 간격 지정”
nonurgentblocklimit	358 페이지 “12.5.7 크기 기반 메시지 우선 순위”
nonurgentnotices	253 페이지 “10.10.4.3 알림 메일 전달 간격 설정”
noreceivedfor	367 페이지 “12.6.12 봉투의 To: 및 From: 주소에서 Received: 헤더 행 구성”
noreceivedfrom	367 페이지 “12.6.12 봉투의 To: 및 From: 주소에서 Received: 헤더 행 구성”
noremotehost	364 페이지 “12.6.6 불완전한 주소를 수정할 때 사용할 호스트 이름 지정”
norestricted	366 페이지 “12.6.10 제한된 메일함 인코딩 사용”
noreturnaddress	255 페이지 “포스트마스터에게 반환되는 메일 내용”

표 12–1 채널 키워드(알파벳순) (계속)

키워드	추가 정보
noreturnpersonal	255 페이지 “포스트마스터에게 반환되는 메일 내용”
noreverse	366 페이지 “12.6.9 역방향 데이터베이스의 채널별 사용”
normalbackoff	355 페이지 “12.5.3 전달에 실패한 메시지에 대한 재시도 간격 지정”
normalblocklimit	358 페이지 “12.5.7 크기 기반 메시지 우선 순위”
normalnotices	253 페이지 “10.10.4.3 알림 메일 전달 간격 설정”
norules	370 페이지 “12.6.17 채널별 다시 쓰기 규칙 검사 사용”
nosasl	349 페이지 “12.4.4 SMTP 인증, SASL 및 TLS”
nosaslserver	349 페이지 “12.4.4 SMTP 인증, SASL 및 TLS”
nosaslswitchchannel	349 페이지 “12.4.4 SMTP 인증, SASL 및 TLS”
nosendetrn	337 페이지 “12.4.2.3 ETRN 명령 지원”
nosendpost	254 페이지 “반환되는 실패 메일”
noservice	360 페이지 “12.5.10 서비스 변환 사용”
noslave_debug	388 페이지 “12.11.2 디버깅 키워드”
nosmtp	336 페이지 “12.4.2.1 채널 프로토콜 선택 및 행 종결 기호”
nosourcefilter	390 페이지 “12.12.4 메일함 필터 파일 위치 지정”
noswitchchannel	347 페이지 “12.4.3.8 받는 메일을 위한 대체 채널(전환 채널)”
notices	253 페이지 “10.10.4.3 알림 메일 전달 간격 설정”
notificationchannel	389 페이지 “12.12.1 프로세스 채널 대체”
notls	351 페이지 “12.4.8 전송 계층 보안”
notlsclient	351 페이지 “12.4.8 전송 계층 보안”
notlsserver	351 페이지 “12.4.8 전송 계층 보안”
novrfy	338 페이지 “12.4.2.4 VRFY 명령 지원”
nowarnpost	254 페이지 “경고 메일”
nox_env_to	373 페이지 “12.7.3 X-Envelope-to 헤더 행 생성/제거”
parameterlengthlimit	385 페이지 “12.9.6 일반 및 Filename Content-type 및 Content-disposition 매개변수의 길이 제어”
percentonly	363 페이지 “12.6.3 주소에 라우팅 정보 추가”
percents	361 페이지 “12.6.1 주소 유형 및 규칙”

표 12–1 채널 키워드(알파벳순) *(계속)*

키워드	추가 정보
personalinc	369 페이지 “12.6.14 주소 헤더 행에서 개인 이름 처리”
personalmap	369 페이지 “12.6.14 주소 헤더 행에서 개인 이름 처리”
personalomit	369 페이지 “12.6.14 주소 헤더 행에서 개인 이름 처리”
personalstrip	369 페이지 “12.6.14 주소 헤더 행에서 개인 이름 처리”
pool	356 페이지 “12.5.4 채널 실행 작업의 처리 풀”
port	343 페이지 “12.4.3.1 TCP/IP 포트 번호 및 인터페이스 주소”
postheadbody	255 페이지 “포스트마스터에게 반환되는 메일 내용”
postheadonly	255 페이지 “포스트마스터에게 반환되는 메일 내용”
randommx	346 페이지 “12.4.3.5 TCP/IP MX 레코드 지원”
receivedfor	367 페이지 “12.6.12 봉투의 To: 및 From: 주소에서 Received: 헤더 행 구성”
receivedfrom	367 페이지 “12.6.12 봉투의 To: 및 From: 주소에서 Received: 헤더 행 구성”
recipientcutoff	385 페이지 “12.9.7 메시지 수신자 제한”
recipientlimit	385 페이지 “12.9.7 메시지 수신자 제한”
rejectsmtplonglines	384 페이지 “12.9.5 1000자가 넘는 행이 있는 SMTP 메일 처리”
remotehost	364 페이지 “12.6.6 불완전한 주소를 수정할 때 사용할 호스트 이름 지정”
restricted	366 페이지 “12.6.10 제한된 메일함 인코딩 사용”
returnaddress	255 페이지 “포스트마스터에게 반환되는 메일 내용”
returnenvelope	255 페이지 “빈 봉투 반송 주소”
returnpersonal	255 페이지 “포스트마스터에게 반환되는 메일 내용”
reverse	366 페이지 “12.6.9 역방향 데이터베이스의 채널별 사용”
routelocal	364 페이지 “12.6.4 명시적 라우팅 주소의 다시 쓰기 사용 안 함”
rules	370 페이지 “12.6.17 채널별 다시 쓰기 규칙 검사 사용”
saslswitchchannel	349 페이지 “12.4.4 SMTP 인증, SASL 및 TLS”
sendetrn	337 페이지 “12.4.2.3 ETRN 명령 지원”
sendpost	254 페이지 “반환되는 실패 메일”
sensitivitycompanyconfidential	376 페이지 “12.7.9 민감도 검사”
sensitivitynormal	376 페이지 “12.7.9 민감도 검사”
sensitivitypersonal	376 페이지 “12.7.9 민감도 검사”

표 12–1 채널 키워드(알파벳순) (계속)

키워드	추가 정보
sensitivityprivate	376 페이지 “12.7.9 민감도 검사”
service	360 페이지 “12.5.10 서비스 변환 사용”
sevenbit	339 페이지 “12.4.2.7 문자 세트 레이블링 및 8비트 데이터”
silentetrn	337 페이지 “12.4.2.3 ETRN 명령 지원”
single	384 페이지 “12.9.4 할당량이 초과된 사용자에 대한 메일 전달 처리”
single_sys	348 페이지 “12.4.3.10 대상 호스트 선택”
slave	354 페이지 “12.5.1 채널 방향 설정”
slave_debug	388 페이지 “12.11.2 디버깅 키워드”
smtp	336 페이지 “12.4.2.1 채널 프로토콜 선택 및 행 종결 기호”
smtp_cr	336 페이지 “12.4.2.1 채널 프로토콜 선택 및 행 종결 기호”
smtp_crlf	336 페이지 “12.4.2.1 채널 프로토콜 선택 및 행 종결 기호”
smtp_crorlf	336 페이지 “12.4.2.1 채널 프로토콜 선택 및 행 종결 기호”
smtp_lf	336 페이지 “12.4.2.1 채널 프로토콜 선택 및 행 종결 기호”
sourceblocklimit	382 페이지 “12.9.2 절대 메시지 크기 제한 지정”
sourcecommentinc	368 페이지 “12.6.13 주소 헤더 행의 주석 처리”
sourcecommentmap	368 페이지 “12.6.13 주소 헤더 행의 주석 처리”
sourcecommentomit	368 페이지 “12.6.13 주소 헤더 행의 주석 처리”
sourcecommentstrip	368 페이지 “12.6.13 주소 헤더 행의 주석 처리”
sourcecommenttotal	368 페이지 “12.6.13 주소 헤더 행의 주석 처리”
sourcefilter	390 페이지 “12.12.4 메일함 필터 파일 위치 지정”
sourcenosolicit	395 페이지 “12.12.7 NO-SOLICIT SMTP 확장 지원”
sourcepersonalinc	369 페이지 “12.6.14 주소 헤더 행에서 개인 이름 처리”
sourcepersonalmap	369 페이지 “12.6.14 주소 헤더 행에서 개인 이름 처리”
sourcepersonalomit	369 페이지 “12.6.14 주소 헤더 행에서 개인 이름 처리”
sourcepersonalstrip	369 페이지 “12.6.14 주소 헤더 행에서 개인 이름 처리”
sourceroute	361 페이지 “12.6.1 주소 유형 및 규칙”
sourcespamfilterX	390 페이지 “12.12.5 스팸 필터 키워드”
sourcespamfilterXoptin	390 페이지 “12.12.5 스팸 필터 키워드”

표 12–1 채널 키워드(알파벳순) (계속)

키워드	추가 정보
sourcesrs	486 페이지 “15.7 SRS(Sender Rewriting Scheme)를 사용하여 SPF에서 전달된 메일 처리”
streaming	340 페이지 “12.4.2.8 프로토콜 스트리밍”
subaddressexact	370 페이지 “12.6.16 하위 주소 처리”
subaddressrelaxed	370 페이지 “12.6.16 하위 주소 처리”
subaddresswild	370 페이지 “12.6.16 하위 주소 처리”
subdirs	386 페이지 “12.10.2 여러 하위 디렉토리로 채널 메시지 대기열 분산”
submit	389 페이지 “12.12.2 채널 작업 유형”
suppressfinal	253 페이지 “10.10.4.4 상태 알림 메일에 변경된 주소 포함”
switchchannel	347 페이지 “12.4.3.8 받는 메일을 위한 대체 채널(전환 채널)”
threaddepth	359 페이지 “12.5.8 SMTP 채널 스레드”
tlsswitchchannel	351 페이지 “12.4.8 전송 계층 보안”
transactionlimit	358 페이지 “12.5.6 연결 트랜잭션 제한 설정”
truncatesmtplonglines	384 페이지 “12.9.5 1000자가 넘는 행이 있는 SMTP 메일 처리”
unrestricted	366 페이지 “12.6.10 제한된 메일함 인코딩 사용”
urgentbackoff	355 페이지 “12.5.3 전달에 실패한 메시지에 대한 재시도 간격 지정”
urgentblocklimit	358 페이지 “12.5.7 크기 기반 메시지 우선 순위”
urgentnotices	253 페이지 “10.10.4.3 알림 메일 전달 간격 설정”
useintermediate	253 페이지 “10.10.4.4 상태 알림 메일에 변경된 주소 포함”
user	389 페이지 “12.12.3 파이프 채널”
userswitchchannel	347 페이지 “12.4.3.9 사용자 또는 도메인 설정을 기준으로 소스 채널 전환”
uucp	361 페이지 “12.6.1 주소 유형 및 규칙”
viaaliasoptional	371 페이지 “12.6.19 반드시 별칭을 통해 주소 지정”
viaaliasrequired	371 페이지 “12.6.19 반드시 별칭을 통해 주소 지정”
vrfyallow	338 페이지 “12.4.2.4 VRFY 명령 지원”
vrfydefault	338 페이지 “12.4.2.4 VRFY 명령 지원”
vrfyhide	338 페이지 “12.4.2.4 VRFY 명령 지원”
warnpost	254 페이지 “경고 메일”

표 12–1 채널 키워드(알파벳순) (계속)

키워드	추가 정보
wrapsmtplonglines	384 페이지 “12.9.5 1000자가 넘는 행이 있는 SMTP 메일 처리”
x_env_to	373 페이지 “12.7.3 X-Envelope-to 헤더 행 생성/제거”

12.3 채널 키워드 범주화(기능별)

다음 표는 범주화한 키워드 목록입니다. 표와 범주는 다음과 같습니다.

- 표 12–2 주소 처리 키워드
- 표 12–3 첨부 파일 및 MIME 처리
- 표 12–4 문자 세트 및 8비트 데이터
- 표 12–5 MTA 대기열 영역에서 파일 만들기
- 표 12–6 헤더 키워드
- 표 12–7 받는 채널 일치 및 전환 키워드
- 표 12–8 로깅 및 디버깅 채널 키워드
- 표 12–9 긴 주소 목록 또는 헤더 채널 키워드
- 표 12–10 메일함 필터 채널 키워드
- 표 12–11 NO-SOLICIT SMTP 확장 지원 키워드
- 표 12–12 알림 및 포스트마스터 메시지 키워드
- 표 12–13 제어 및 작업 전송 처리 키워드
- 표 12–14 민감도 제한 키워드
- 표 12–15 메시지, 사용자 할당량, 권한 및 인증 시도의 제한 키워드
- 표 12–16 SMTP 인증, SASL 및 TLS 키워드
- 표 12–17 SMTP 명령 및 프로토콜 키워드
- 표 12–18 TCP/IP 연결 및 DNS 조회 지원 키워드
- 표 12–19 기타 키워드

표 12–2 주소 처리 키워드

키워드	페이지 정의
주소 처리	
733	봉투에 percents와 동의어인 % 라우팅을 사용합니다. 361 페이지 “12.6.1 주소 유형 및 규칙”
822	361 페이지 “12.6.1 주소 유형 및 규칙” 봉투에 sourceroute와 동일한 소스 루트를 사용합니다.
addreturnpath	367 페이지 “12.6.11 Return-path: 헤더 행 생성” Return-path: 헤더를 이 채널의 대기열에 포함된 메시지에 추가합니다.

표 12–2 주소 처리 키워드 (계속)

키워드	페이지 정의
aliaslocal	369 페이지 “12.6.15 별칭 파일 및 별칭 데이터베이스 검사 지정”
	별칭 파일 및 별칭 데이터베이스에서 다시 쓴 주소를 조회합니다.
authrewrite	341 페이지 “12.4.3 TCP/IP 연결 및 DNS 조회 지원”
	MTA에서 인증된 메시지 발송자 정보(사용 가능한 경우)를 헤더로 전파하도록 하기 위해 소스 채널에서 사용됩니다.
bangoverpercent	363 페이지 “12.6.3 주소에 라우팅 정보 추가”
	A!B%C를 A!(B%C)로 그룹화합니다.
bangstyle	361 페이지 “12.6.1 주소 유형 및 규칙”
	봉투에 `uucp`와 동의어인 UUCP ! 라우팅을 사용합니다.
defaulthost	364 페이지 “12.6.6 불완전한 주소를 수정할 때 사용할 호스트 이름 지정”
	주소를 완성하기 위해 사용할 도메인 이름을 지정합니다.
dequeue_removeroute	371 페이지 “12.6.18 소스 경로 제거”
	봉투의 `To:` 주소에서 소스 경로를 제거합니다.
exproute	363 페이지 “12.6.3 주소에 라우팅 정보 추가”
	주소가 원격 시스템에 전달될 때 명시적 라우팅이 필요합니다.
holdlimit	359 페이지 “12.5.9 여러 주소 확장”
	봉투의 수신자 주소 수가 이 제한을 초과할 경우 메시지를 보관합니다.
improute	363 페이지 “12.6.3 주소에 라우팅 정보 추가”
	이 채널 주소에 대한 암시적 라우팅이 필요합니다.
missingrecipientpolicy	365 페이지 “12.6.7 수신자 헤더 행 없이 메시지 적법화”
	수신자 헤더가 없는 메시지를 적법화(헤더 추가)하는 방법에 대한 정책을 설정합니다.
noaddreturnpath	367 페이지 “12.6.11 Return-path: 헤더 행 생성”
	메시지를 대기열에 포함할 때 `Return-path:` 헤더를 추가하지 않습니다.
nobangoverpercent	363 페이지 “12.6.3 주소에 라우팅 정보 추가”
	A!B%C를 (A!B)%C로 그룹화합니다.
nodefaulthost	364 페이지 “12.6.6 불완전한 주소를 수정할 때 사용할 호스트 이름 지정”
	주소를 완성하기 위해 사용할 도메인 이름을 지정하지 않습니다.
noexproute	363 페이지 “12.6.3 주소에 라우팅 정보 추가”
	이 채널의 주소에 대한 명시적 라우팅이 필요하지 않습니다.

표 12–2 주소 처리 키워드 (계속)

키워드	페이지 정의
noimproute	363 페이지 “12.6.3 주소에 라우팅 정보 추가” 이 채널의 주소에 대한 암시적 라우팅이 필요하지 않습니다.
noreceivedfrom	367 페이지 “12.6.12 봉투의 To: 및 From: 주소에서 Received: 헤더 행 구성” 원본 봉투의 From: 주소를 포함하지 않고 `Received:` 헤더 행을 지시합니다.
noremotehost	364 페이지 “12.6.6 불완전한 주소를 수정할 때 사용할 호스트 이름 지정” 주소를 완성하기 위해 로컬 호스트의 도메인 이름을 기본 도메인 이름으로 사용합니다.
norestricted	366 페이지 “12.6.10 제한된 메일함 인코딩 사용” `unsrestricted`와 동일합니다.
noreverse	366 페이지 “12.6.9 역방향 데이터베이스의 채널별 사용” 메시지의 주소를 역방향 주소 처리에서 제외합니다.
norules	370 페이지 “12.6.17 채널별 다시 쓰기 규칙 검사 사용” 이 채널에 대해 채널별 다시 쓰기 규칙 검사를 실행하지 않습니다.
percentonly	363 페이지 “12.6.3 주소에 라우팅 정보 추가” 뱅 경로를 무시합니다. 봉투에 % 라우팅을 사용합니다.
percents	361 페이지 “12.6.1 주소 유형 및 규칙” 봉투에 733과 동의어인 % 라우팅을 사용합니다.
remotehost	364 페이지 “12.6.6 불완전한 주소를 수정할 때 사용할 호스트 이름 지정” 주소를 완성하기 위해 원격 호스트의 이름을 기본 도메인 이름으로 사용합니다.
restricted	366 페이지 “12.6.10 제한된 메일함 인코딩 사용” 채널이 인코딩을 필요로 하는 메일 시스템에 연결됩니다.
reverse	366 페이지 “12.6.9 역방향 데이터베이스의 채널별 사용” 주소를 주소 역방향 데이터베이스 또는 REVERSE 매핑에 대해 검사합니다.
routelocal	364 페이지 “12.6.4 명시적 라우팅 주소의 다시 쓰기 사용 안 함” 채널에 주소를 다시 쓸 때 MTA가 주소의 명시적 라우팅을 “단락”하게 합니다.
rules	370 페이지 “12.6.17 채널별 다시 쓰기 규칙 검사 사용” 이 채널에 대해 채널별 다시 쓰기 규칙 검사를 실행합니다.
sourceroute	361 페이지 “12.6.1 주소 유형 및 규칙” 822와 동의어입니다.

표 12–2 주소 처리 키워드 *(계속)*

키워드	페이지 정의
subaddressexact	370 페이지 “12.6.16 하위 주소 처리” 항목 일치 작업 중에 특수 하위 주소 처리를 수행하지 않습니다. 하위 주소를 포함하여 전체 메일함이 항목과 일치해야 별칭이 일치하는 것으로 간주됩니다.
subaddressrelaxed	370 페이지 “12.6.16 하위 주소 처리” MTA는 전체 일치와 name+* 형식의 일치를 차례로 조사한 다음 이름 부분에 대한 일치를 추가로 확인해야 합니다.
subaddresswild	370 페이지 “12.6.16 하위 주소 처리” MTA는 전체 하위 주소를 포함한 정확한 일치를 조사한 다음 name+* 형식의 항목을 조사해야 합니다.
unrestricted	366 페이지 “12.6.10 제한된 메일함 인코딩 사용” MTA에 RFC 1137 인코딩 및 디코딩을 수행하지 않도록 지시합니다.
uucp	361 페이지 “12.6.1 주소 유형 및 규칙” 봉투에 bangstyle과 동의어인 UUCP! 라우팅을 사용합니다.
viaaliasoptional	371 페이지 “12.6.19 반드시 별칭을 통해 주소 지정” 채널과 일치하는 최종 수신자 주소를 별칭을 통해 생성할 필요가 없습니다.
viaaliasrequired	371 페이지 “12.6.19 반드시 별칭을 통해 주소 지정” 채널과 일치하는 최종 수신자 주소를 별칭을 통해 생성해야 합니다.

표 12–3 첨부 파일 및 MIME 처리

키워드	정의
defragment	377 페이지 “12.8.2 메시지/부분 메시지 자동 조각 모음” 채널에 대기 중인 부분 메시지를 조각 모음 채널 대기열에 대신 넣습니다.
ignoreencoding	377 페이지 “12.8.1 Encoding 헤더 행 무시” 받는 메시지에서 `Encoding:` 헤더를 무시합니다.
ignoremessageencoding	381 페이지 “12.8.5 멀티파트 및 Message/RFC822 부분의 cContent-transfer-encoding 필드 해석” 받는 메시지의 message/rfc822 부분에 있는 content-transfer-encoding 필드를 무시합니다.
ignoremultipartencoding	381 페이지 “12.8.5 멀티파트 및 Message/RFC822 부분의 cContent-transfer-encoding 필드 해석” 받는 메시지의 멀티파트 부분에 있는 content-transfer-encoding 필드를 무시합니다.
interpretencoding	377 페이지 “12.8.1 Encoding 헤더 행 무시” 필요한 경우 받는 메시지에서 `Encoding:` 헤더를 해석합니다.

표 12–3 첨부 파일 및 MIME 처리 *(계속)*

키워드	정의
interpretmessageencoding	381 페이지 “12.8.5 멀티파트 및 Message/RFC822 부분의 cContent-transfer-encoding 필드 해석” 받는 메시지의 message/rfc822 부분에 있는 content-transfer-encoding 필드를 해석합니다.
interpretmultipartencoding	381 페이지 “12.8.5 멀티파트 및 Message/RFC822 부분의 cContent-transfer-encoding 필드 해석” 받는 메시지의 멀티파트 부분에 있는 content-transfer-encoding 필드를 해석합니다.
nodefragment	377 페이지 “12.8.2 메시지/부분 메시지 자동 조각 모음” 조각 모음을 사용하지 않습니다.

표 12–4 문자 세트 및 8비트 데이터

키워드	정의
charset7	339 페이지 “12.4.2.7 문자 세트 레이블링 및 8비트 데이터” 7비트 텍스트 메시지과 관련된 기본 문자 세트입니다.
charset8	339 페이지 “12.4.2.7 문자 세트 레이블링 및 8비트 데이터” 8비트 텍스트 메시지와 관련된 기본 문자 세트입니다.
charsetesc	339 페이지 “12.4.2.7 문자 세트 레이블링 및 8비트 데이터” 이스케이프 문자를 포함하는 7비트 텍스트 메시지와 관련된 기본 문자 세트입니다.
eightbit	339 페이지 “12.4.2.7 문자 세트 레이블링 및 8비트 데이터” 채널에서 8비트 문자를 지원합니다.
eightnegotiate	339 페이지 “12.4.2.7 문자 세트 레이블링 및 8비트 데이터” 채널에서 가능할 경우 8비트 전송 사용을 협상합니다.
eightstrict	339 페이지 “12.4.2.7 문자 세트 레이블링 및 8비트 데이터” 협상되지 않은 8비트 데이터를 포함하는 메시지를 거부합니다.
sevenbit	339 페이지 “12.4.2.7 문자 세트 레이블링 및 8비트 데이터” 8비트 문자를 지원하지 않으므로 8비트 문자를 인코딩해야 합니다.

표 12–5 MTA 대기열 영역에서 파일 만들기

키워드	페이지 정의
addrsperfile	384 페이지 “12.9.4 할당량이 초과된 사용자에 대한 메일 전달 처리” 채널 대기열에서 단일 메시지 파일과 연결될 수 있는 최대 수신자 수를 제한합니다.

표 12–5 MTA 대기열 영역에서 파일 만들기 *(계속)*

키워드	페이지 정의
expandchannel	359 페이지 "12.5.9 여러 주소 확장" expandlimit 적용으로 인해 지연된 확장을 수행할 채널을 지정합니다.
expandlimit	359 페이지 "12.5.9 여러 주소 확장" 주소 수가 이 제한을 초과할 경우 받는 메시지를 "오프라인"으로 처리합니다.
multiple	384 페이지 "12.9.4 할당량이 초과된 사용자에 대한 메일 전달 처리" 메시지 파일의 수신자 수를 제한하지 않습니다. SMTP 채널의 기본값은 99입니다.
single	384 페이지 "12.9.4 할당량이 초과된 사용자에 대한 메일 전달 처리" 채널의 각 대상 주소에 대해 별도의 메시지 복사본을 만듭니다.
single_sys	384 페이지 "12.9.4 할당량이 초과된 사용자에 대한 메일 전달 처리" 사용된 각 대상 시스템에 대해 단일 메시지 복사본을 만듭니다.
subdirs	386 페이지 "12.10.2 여러 하위 디렉토리로 채널 메시지 대기열 분산" 채널 대기열의 메시지를 분산시킬 하위 디렉토리 수를 지정합니다.

표 12–6 헤더 키워드

키워드	정의
authrewrite	341 페이지 "12.4.3 TCP/IP 연결 및 DNS 조회 지원" MTA에서 인증된 메시지 발송자 정보(사용 가능한 경우)를 헤더로 전파하도록 하기 위해 소스 채널에서 사용됩니다.
commentinc	368 페이지 "12.6.13 주소 헤더 행의 주석 처리" 메시지 헤더 행의 주석을 그대로 둡니다.
commentmap	368 페이지 "12.6.13 주소 헤더 행의 주석 처리" `COMMENT_STRINGS` 매핑 테이블을 통해 메시지 헤더 행에 주석 문자열을 실행합니다.
commentomit	368 페이지 "12.6.13 주소 헤더 행의 주석 처리" 메시지 헤더 행에서 주석을 제거합니다.
commentstrip	368 페이지 "12.6.13 주소 헤더 행의 주석 처리" 메시지 헤더 행의 주석 필드에서 문제가 있는 문자를 제거합니다.
commenttotal	368 페이지 "12.6.13 주소 헤더 행의 주석 처리" Received: 헤더 행을 제외한 모든 헤더 행에서 주석(괄호 안의 내용)을 제거합니다. 권장되지 않습니다.

표 12–6 헤더 키워드 (계속)

키워드	정의
datefour	374 페이지 “12.7.4 두 자리 또는 네 자리로 날짜 변환” 모든 연도 필드를 네 자리로 확장합니다.
datetwo	374 페이지 “12.7.4 두 자리 또는 네 자리로 날짜 변환” 네 자리 날짜에서 앞의 두 자리를 제거하여두 자리 날짜를 사용해야 하는 메일 시스템과 호환되게 합니다. 그 밖의 다른 목적으로 사용해서는 안 됩니다.
dayofweek	374 페이지 “12.7.5 날짜의 요일 지정” 요일 정보를 보존하고 이 정보를 날짜 및 시간 헤더에 추가합니다(없는 경우).
defaulthost	364 페이지 “12.6.6 불완전한 주소를 수정할 때 사용할 호스트 이름 지정” 주소를 완성하기 위해 사용할 도메인 이름을 지정합니다.
deletemessagehash	376 페이지 “12.7.11 Message-hash: 헤더 제어” 기존 Message-hash: 필드를 모두 삭제합니다.
dropblank	366 페이지 “12.6.8 잘못된 빈 수신자 헤더 스트라이핑” 받는 메시지에서 잘못된 빈 헤더를 제거합니다.
generatemessagehash	376 페이지 “12.7.11 Message-hash: 헤더 제어” 대상 채널에서 지정한 경우 메시지에 Message-hash: 헤더 필드를 삽입합니다.
header_733	361 페이지 “12.6.1 주소 유형 및 규칙” 메시지 헤더에 % 라우팅을 사용합니다.
header_822	361 페이지 “12.6.1 주소 유형 및 규칙” 메시지 헤더에 소스 경로를 사용합니다.
headerlabelalign	375 페이지 “12.7.7 헤더 맞춤 및 접기” 이 채널의 대기열에 포함된 메시지 헤더에 대한 맞춤 점을 제어합니다. 맞춤 점은 정수 값 인수를 가집니다.
headerlinelength	375 페이지 “12.7.7 헤더 맞춤 및 접기” 이 채널의 대기열에 포함된 헤더 행의 길이를 제어합니다.
headerread	372 페이지 “12.7.2 선택한 메시지 헤더 행 제거” 원본 메시지 헤더를 처리하기 전에 옵션 파일의 헤더 자르기 규칙을 메시지 대기열에 포함된 메시지 헤더에 적용합니다.
headertrim	372 페이지 “12.7.2 선택한 메시지 헤더 행 제거” 원본 메시지 헤더를 처리한 후에 옵션 파일의 헤더 자르기 규칙을 메시지 헤더에 적용합니다.

표 12–6 헤더 키워드 *(계속)*

키워드	정의
header_uucp	361 페이지 “12.6.1 주소 유형 및 규칙” 헤더에 ! 라우팅을 사용합니다.
inner	372 페이지 “12.7.1 포함 헤더 다시 쓰기” 메시지를 구문 분석하고 내부 헤더를 다시 씁니다.
innertrim	372 페이지 “12.7.2 선택한 메시지 헤더 행 제거” 옵션 파일의 헤더 자르기 규칙을 내부 메시지 헤더에 적용합니다.
keepmessagehash	376 페이지 “12.7.11 `Message-hash:` 헤더 제어” 기존 `Message-hash:` 필드를 모두 유지합니다.
language	376 페이지 “12.7.10 헤더의 기본 언어 설정” 헤더에 기본 언어를 지정합니다.
maxheaderaddrs	374 페이지 “12.7.6 긴 헤더 행 자동 분할” 한 행에 표시할 수 있는 주소 수를 제어합니다.
maxheaderchars	374 페이지 “12.7.6 긴 헤더 행 자동 분할” 한 행에 표시할 수 있는 문자 수를 제어합니다.
missingrecipientpolicy	365 페이지 “12.6.7 수신자 헤더 행 없이 메시지 적법화” 수신자 헤더가 없는 메시지를 적법화(헤더 추가)하는 방법에 대한 정책을 설정합니다.
nodayofweek	374 페이지 “12.7.5 날짜의 요일 지정” 날짜 및 시간 헤더에서 요일을 제거하여이 정보를 처리할 수 없는 메일 시스템과 호환되게 합니다. 그 밖의 다른 목적으로 사용해서는 안 됩니다.
nodefaulthost	364 페이지 “12.6.6 불완전한 주소를 수정할 때 사용할 호스트 이름 지정” 주소를 완성하기 위해 사용할 도메인 이름을 지정하지 않습니다.
nodropblank	366 페이지 “12.6.8 잘못된 빈 수신자 헤더 스트라이핑” 받는 메시지에서 잘못된 빈 헤더를 제거하지 않습니다.
noheaderread	372 페이지 “12.7.2 선택한 메시지 헤더 행 제거” 옵션 파일의 헤더 자르기 규칙을 적용하지 않습니다.
noheadertrim	372 페이지 “12.7.2 선택한 메시지 헤더 행 제거” 옵션 파일의 헤더 자르기 규칙을 적용하지 않습니다.
noinner	372 페이지 “12.7.1 포함 헤더 다시 쓰기” 내부 메시지 헤더 행을 다시 쓰지 않습니다.

표 12–6 헤더 키워드 (계속)

키워드	정의
noinnertrim	372 페이지 “12.7.2 선택한 메시지 헤더 행 제거” 내부 메시지 헤더에 헤더 자르기를 적용하지 않습니다.
noreceivedfor	367 페이지 “12.6.12 봉투의 To: 및 From: 주소에서 Received: 헤더 행 구성” 봉투 수신자 정보를 포함하지 않고 `Received:` 헤더 행을 구성합니다.
noreceivedfrom	367 페이지 “12.6.12 봉투의 To: 및 From: 주소에서 Received: 헤더 행 구성” 원본 봉투의 From: 주소를 포함하지 않고 `Received:` 헤더 행을 지시합니다.
noremotehost	364 페이지 “12.6.6 불완전한 주소를 수정할 때 사용할 호스트 이름 지정” 주소를 완성하기 위해 로컬 호스트의 도메인 이름을 기본 도메인 이름으로 사용합니다.
noreverse	366 페이지 “12.6.9 역방향 데이터베이스의 채널별 사용” 채널에 대기 중인 메시지에서 역방향 주소 처리를 사용하지 않습니다.
norules	370 페이지 “12.6.17 채널별 다시 쓰기 규칙 검사 사용” 이 채널에 대해 채널별 다시 쓰기 규칙 검사를 실행하지 않습니다.
nox_env_to	373 페이지 “12.7.3 X-Envelope-to 헤더 행 생성/제거” `X-Envelope-to` 헤더 행을 제거합니다.
personalinc	369 페이지 “12.6.14 주소 헤더 행에서 개인 이름 처리” 메시지 헤더 행의 개인 이름 필드를 그대로 둡니다.
personalmap	369 페이지 “12.6.14 주소 헤더 행에서 개인 이름 처리” PERSONAL_NAMES 매핑 테이블을 통해 개인 이름을 실행합니다.
personalomit	369 페이지 “12.6.14 주소 헤더 행에서 개인 이름 처리” 메시지 헤더 행에서 개인 이름 필드를 제거합니다.
personalstrip	369 페이지 “12.6.14 주소 헤더 행에서 개인 이름 처리” 헤더 행의 개인 이름 필드에서 문제가 있는 문자를 제거합니다.
receivedfor	367 페이지 “12.6.12 봉투의 To: 및 From: 주소에서 Received: 헤더 행 구성” 메시지의 주소가 한 명의 봉투 수신자로 지정된 경우 해당 봉투의 `To:` 주소를 메시지가 구성하는 Received: 헤더 행에 포함합니다.
receivedfrom	367 페이지 “12.6.12 봉투의 To: 및 From: 주소에서 Received: 헤더 행 구성” MTA에서 봉투의 `From:` 주소를 변경한 경우, 받는 메시지에 대한 `Received:` 헤더 행을 구성할 때 봉투의 원래 `From:` 주소를 포함합니다.

표 12–6 헤더 키워드 *(계속)*

키워드	정의
remotehost	364 페이지 “12.6.6 불완전한 주소를 수정할 때 사용할 호스트 이름 지정” 주소를 완성하기 위해 원격 호스트의 이름을 기본 도메인 이름으로 사용합니다.
restricted	366 페이지 “12.6.10 제한된 메일함 인코딩 사용” 채널이 이 인코딩을 필요로 하는 메일 시스템에 연결됩니다.
reverse	366 페이지 “12.6.9 역방향 데이터베이스의 채널별 사용” 주소를 주소 역방향 데이터베이스 또는 REVERSE 매핑에 대해 검사합니다.
rules	370 페이지 “12.6.17 채널별 다시 쓰기 규칙 검사 사용” 이 채널에 대해 채널별 다시 쓰기 규칙 검사를 실행합니다.
sensitivitycompanyconfidential	374 페이지 “12.7.9 민감도 검사” `Companyconfidential`이 메시지에 대한 최대 민감도 제한으로 적용됩니다.
sensitivitynormal	376 페이지 “12.7.9 민감도 검사” `Normal`이 메시지에 대한 최대 민감도 제한으로 적용됩니다.
sensitivitypersonal	376 페이지 “12.7.9 민감도 검사” `Personal`이 메시지에 대한 최대 민감도 제한으로 적용됩니다.
sensitivityprivate	376 페이지 “12.7.9 민감도 검사” `Private`이 메시지에 대한 최대 민감도 제한으로 적용됩니다.
sourcecommentinc	368 페이지 “12.6.13 주소 헤더 행의 주석 처리” 받는 메시지 헤더 행의 주석을 그대로 둡니다.
sourcecommentmap	368 페이지 “12.6.13 주소 헤더 행의 주석 처리” 소스 채널을 통해 헤더 행에 주석 문자열을 실행합니다.
sourcecommentomit	368 페이지 “12.6.13 주소 헤더 행의 주석 처리” 받는 메시지 헤더 행(예: To:, From:, Cc: 등과 같은 주소 지정 헤더)에서 주석을 제거하도록 MTA에 지시합니다.
sourcecommentstrip	368 페이지 “12.6.13 주소 헤더 행의 주석 처리” 받는 헤더 행의 주석 필드에서 문제가 있는 문자를 제거합니다.
sourcecommenttotal	368 페이지 “12.6.13 주소 헤더 행의 주석 처리” 받는 메시지에서 주석(괄호 안의 내용)을 제거합니다.
sourcepersonalinc	369 페이지 “12.6.14 주소 헤더 행에서 개인 이름 처리” 받는 메시지 헤더 행의 개인 이름을 그대로 둡니다.

표 12–6 헤더 키워드 *(계속)*

키워드	정의
sourcepersonalmap	369 페이지 “12.6.14 주소 헤더 행에서 개인 이름 처리” 소스 채널을 통해 개인 이름을 실행합니다.
sourcepersonalomit	369 페이지 “12.6.14 주소 헤더 행에서 개인 이름 처리” 받는 메시지 헤더 행에서 개인 이름 필드를 제거합니다.
sourcepersonalstrip	369 페이지 “12.6.14 주소 헤더 행에서 개인 이름 처리” 받는 메시지 헤더 행의 개인 이름 필드에서 문제가 있는 문자를 제거합니다.
unrestricted	366 페이지 “12.6.10 제한된 메일함 인코딩 사용” MTA에 RFC 1137 인코딩 및 디코딩을 수행하지 않도록 지시합니다.
x_env_to	373 페이지 “12.7.3 X-Envelope-to 헤더 행 생성/제거” `X-Envelope-to` 헤더 행을 생성합니다.

표 12–7 받는 채널 일치 및 전환 키워드

키워드	정의
allowswitchchannel	347 페이지 “12.4.3.8 받는 메일을 위한 대체 채널(전환 채널)” `switchchannel` 채널에서 이 채널로 전환을 허용합니다.
nosaslswitchchannel	349 페이지 “12.4.4 SMTP 인증, SASL 및 TLS” SASL 인증에 성공할 때 이 채널로 전환하지 않습니다.
noswitchchannel	347 페이지 “12.4.3.8 받는 메일을 위한 대체 채널(전환 채널)” 채널 전환을 수행하지 않습니다.
switchchannel	347 페이지 “12.4.3.8 받는 메일을 위한 대체 채널(전환 채널)” 서버 채널에서 원본 호스트와 연결된 채널로 전환합니다.
saslswitchchannel	349 페이지 “12.4.4 SMTP 인증, SASL 및 TLS” 클라이언트가 SASL을 성공적으로 사용하면 받는 연결이 지정한 채널로 전환되게 합니다.
tlsswitchchannel	351 페이지 “12.4.8 전송 계층 보안” TLS 협상이 성공하면 다른 채널로 전환합니다.
userswitchchannel	347 페이지 “12.4.3.9 사용자 또는 도메인 설정을 기준으로 소스 채널 전환” 사용자 또는 도메인 설정을 기준으로 소스 채널을 전환합니다.

표 12–8 로깅 및 디버깅 채널 키워드

키워드	정의
로깅	387 페이지 “12.11.1 로깅 키워드” 로그 파일에 대기열에서 제거되거나 포함된 메시지에 대해 기록하고 특정 채널에 대한 로깅을 활성화합니다.
loopcheck	388 페이지 “12.11.3 Loopcheck 설정” MTA가 자체적으로 통신하는지 확인하도록 SMTP EHLO 응답 배너에 문자열을 넣습니다.
master_debug	388 페이지 “12.11.2 디버깅 키워드” 채널의 마스터 프로그램 출력에 디버깅 출력을 만듭니다.
nologging	387 페이지 “12.11.1 로깅 키워드” 로그 파일에 대기열에 포함되거나 대기열에서 제거된 메시지에 대해 기록하지 않습니다.
noloopcheck	388 페이지 “12.11.3 Loopcheck 설정” SMTP EHLO 응답 배너에 문자열을 넣지 않습니다.
nomaster_debug	388 페이지 “12.11.2 디버깅 키워드” 채널의 마스터 프로그램 출력에 디버깅 출력을 만들지 않습니다.
noslave_debug	388 페이지 “12.11.2 디버깅 키워드” 슬레이브 디버깅 출력을 생성하지 않습니다.
slave_debug	388 페이지 “12.11.2 디버깅 키워드” 슬레이브 디버깅 출력을 생성합니다.

표 12–9 긴 주소 목록 또는 헤더 채널 키워드

키워드	정의
expandchannel	359 페이지 “12.5.9 여러 주소 확장” expandlimit 적용으로 인해 지연된 확장을 수행할 채널을 지정합니다.
expandlimit	359 페이지 “12.5.9 여러 주소 확장” 주소 수가 이 제한을 초과할 경우 받는 메시지를 “오프라인”으로 처리합니다.
holdlimit	359 페이지 “12.5.9 여러 주소 확장” 주소 수가 이 제한을 초과할 경우 메시지를 보관합니다.
maxprocchars	375 페이지 “12.7.7 헤더 맞춤 및 접기” 처리 및 다시 쓰기 가능한 최대 길이 헤더입니다.

표 12–10 메일함 필터 채널 키워드

키워드	정의
channelfilter	390 페이지 “12.12.4 메일함 필터 파일 위치 지정” 채널 필터 파일의 위치이며 `destinationfilter`와 동일합니다.
destinationfilter	보내는 메시지에 적용되는 채널 필터 파일의 위치입니다.
destinationspamfilterX	390 페이지 “12.12.5 스팸 필터 키워드” 스팸 필터링 소프트웨어 X를 통해 이 채널을 대상으로 하는 메시지를 실행합니다. 스팸 필터링 소프트웨어 매개 변수를 허용하지 않습니다.
destinationspamfilterXoptin	390 페이지 “12.12.5 스팸 필터 키워드” 스팸 필터링 소프트웨어 X를 통해 이 채널을 대상으로 하는 메시지를 실행합니다.
disabledestinationspamfilterX	390 페이지 “12.12.5 스팸 필터 키워드” 이 채널을 대상으로 하는 메시지에 대해 스팸 필터 X를 비활성화합니다.
disablesourcespamfilterX	390 페이지 “12.12.5 스팸 필터 키워드” 이 채널에서 보내는 메시지에 대해 스팸 필터 X를 비활성화합니다.
fileinto	390 페이지 “12.12.4 메일함 필터 파일 위치 지정” 메일함 필터 `fileinto` 작업을 적용할 때 주소에 미치는 영향을 지정합니다.
filter	390 페이지 “12.12.4 메일함 필터 파일 위치 지정” 사용자 필터 파일의 위치를 지정합니다.
nochannelfilter	390 페이지 “12.12.4 메일함 필터 파일 위치 지정” 보내는 메시지에 대해 채널 필터링을 수행하지 않습니다. `nodestinationfilter`라고도 합니다.
nodestinationfilter	390 페이지 “12.12.4 메일함 필터 파일 위치 지정” 보내는 메시지에 대해 채널 필터링을 수행하지 않습니다.
nofileinto	390 페이지 “12.12.4 메일함 필터 파일 위치 지정” 메일함 필터 fileinto 연산자가 영향을 미치지 않습니다.
nofilter	390 페이지 “12.12.4 메일함 필터 파일 위치 지정” 사용자 메일함 필터링을 수행하지 않습니다.
nosourcefilter	390 페이지 “12.12.4 메일함 필터 파일 위치 지정” 받는 메시지에 대해 채널 필터링을 수행하지 않습니다.
sourcefilter	390 페이지 “12.12.4 메일함 필터 파일 위치 지정” 받는 메시지에 대해 채널 필터 파일의 위치를 지정합니다.

표 12–10 메일함 필터 채널 키워드 *(계속)*

키워드	정의
sourcespamfilterX	390 페이지 "12.12.5 스팸 필터 키워드" 스팸 필터링 소프트웨어 X를 통해 이 채널에서 수신한 메시지를 실행합니다. 스팸 필터링 소프트웨어 매개 변수를 허용하지 않습니다.
sourcespamfilterXoptin	390 페이지 "12.12.5 스팸 필터 키워드" 스팸 필터링 소프트웨어 X를 통해 이 채널에서 수신한 메시지를 실행합니다. 스팸 필터링 소프트웨어 매개 변수를 허용합니다.

표 12–11 NO-SOLICIT SMTP 확장 지원 키워드

키워드	정의
sourcenosolicit	395 페이지 "12.12.7 NO-SOLICIT SMTP 확장 지원" 이 채널이 전송하는 메일에서 차단될 요청 필드 값의 쉼표로 구분된 목록을 지정합니다.
destinationnosolicit	395 페이지 "12.12.7 NO-SOLICIT SMTP 확장 지원" 이 채널의 대기열에 포함된 메일에서 허용되지 않는 요청 필드 값의 쉼표로 구분된 목록을 지정합니다.

표 12–12 알림 및 포스트마스터 메시지 키워드

키워드	정의
(전체 알림 절차는 246 페이지 "10.10 전달 상태 알림 메일 제어" 페이지 참조)	
aliaspostmaster	255 페이지 "포스트마스터에게 반환되는 메일 내용" 공식 채널 이름에서 포스트마스터 아이디로 주소 지정된 메시지는 postmaster@local-host로 리디렉션됩니다. 여기서 local-host는 로컬 호스트 이름(로컬 채널에 있는 이름)입니다.
copysendpost	254 페이지 "반환되는 실패 메일" 전송 실패 메시지에서 보낸 사람 주소가 비어 있지 않는 한 실패 알림 복사본을 포스트마스터에게 보냅니다.
copywarnpost	254 페이지 "경고 메일" 전달되지 않은 메시지에서 보낸 사람 주소가 비어 있지 않는 한 경고 메시지 복사본을 포스트마스터에게 보냅니다.
errsendpost	254 페이지 "반환되는 실패 메일" 메시지 발송자에게 알림을 보낼 수 없는 경우 실패 알림 복사본을 포스트마스터에게 보냅니다.
errwarnpost	254 페이지 "경고 메일" 메시지 발송자에게 알림을 보낼 수 없는 경우 경고 메시지 복사본을 포스트마스터에게 보냅니다.

표 12–12 알림 및 포스트마스터 메시지 키워드 (계속)

키워드	정의
includefinal	253 페이지 “10.10.4.4 상태 알림 메일에 변경된 주소 포함” 전달 알림에 최종 수신자 주소 형식을 포함합니다.
nonurgentnotices	253 페이지 “10.10.4.3 알림 메일 전달 간격 설정” 우선 순위가 낮은 메시지에 대해 알림을 보내고 메시지를 반환하기 전에 경과할 수 있는 시간을 지정합니다.
noreturnaddress	255 페이지 “포스트마스터에게 반환되는 메일 내용” RETURN_ADDRESS 옵션 값을 포스트마스터 주소 이름으로 사용합니다.
noreturnpersonal	255 페이지 “포스트마스터에게 반환되는 메일 내용” RETURN_PERSONAL 옵션 값을 포스트마스터 개인 이름으로 사용합니다.
normalnotices	253 페이지 “10.10.4.3 알림 메일 전달 간격 설정” 우선 순위가 중간인 메시지에 대해 알림을 보내고 메시지를 반환하기 전에 경과할 수 있는 시간을 지정합니다.
nosendpost	254 페이지 “반환되는 실패 메일” 전달이 실패한 모든 메시지의 복사본을 포스트마스터에게 보내지 않습니다.
notices	253 페이지 “10.10.4.3 알림 메일 전달 간격 설정” 알림을 보낸 후 메시지가 반환되기 이전에 경과할 수 있는 시간을 지정합니다.
nowarnpost	254 페이지 “경고 메일” 경고 메시지의 복사본을 포스트마스터에게 보내지 않습니다.
postheadbody	255 페이지 “포스트마스터에게 반환되는 메일 내용” 헤더와 메시지 내용을 모두 반환합니다.
postheadonly	255 페이지 “포스트마스터에게 반환되는 메일 내용” 헤더만 포스트마스터에게 반환됩니다.
returnaddress	255 페이지 “포스트마스터에게 반환되는 메일 내용” 로컬 포스트마스터에 대한 반송 주소를 지정합니다.
returnenvelope	255 페이지 “빈 봉투 반송 주소” 빈 봉투 반송 주소 사용을 제어합니다.
returnpersonal	255 페이지 “포스트마스터에게 반환되는 메일 내용” 로컬 포스트마스터에 대한 개인 이름을 설정합니다.

표 12–12 알림 및 포스트마스터 메시지 키워드 *(계속)*

키워드	정의
sendpost	254 페이지 "반환되는 실패 메일" 실패한 모든 메시지의 복사본을 포스트마스터에게 보냅니다.
suppressfinal	253 페이지 "10.10.4.4 상태 알림 메일에 변경된 주소 포함" 원본 주소 형식이 있는 경우 알림 메시지에서 최종 주소 형식을 생략합니다.
urgentnotices	253 페이지 "10.10.4.3 알림 메일 전달 간격 설정" 우선 순위가 높은 메시지에 대해 알림을 보내고 메시지를 반환하기 전에 경과할 수 있는 시간을 지정합니다.
useintermediate	253 페이지 "10.10.4.4 상태 알림 메일에 변경된 주소 포함" 목록을 확장한 이후 사용자 메일함 이름이 생성되기 이전에 생성되는 중간 주소 형식을 사용합니다.
warnpost	254 페이지 "경고 메일" 경고 메시지의 복사본을 포스트마스터에게 보냅니다.

표 12–13 제어 및 작업 전송 처리 키워드

키워드	정의
(기능에 대한 자세한 내용은 352 페이지 "12.5 메시지 처리 및 전달 구성" 참조)	
backoff	355 페이지 "12.5.3 전달에 실패한 메시지에 대한 재시도 간격 지정" 전달되지 않은 메시지의 재전달 시도 간격입니다. `normalbackoff`, `nonurgentbackoff`, `urgentbackoff` 키워드로 대체될 수 있습니다.
bidirectional	354 페이지 "12.5.1 채널 방향 설정" 마스터 및 슬레이브 프로그램에서 사용되는 채널입니다.
deferred	354 페이지 "12.5.2 지연 전달 날짜 구현" `Deferred-delivery:` 헤더 행을 인식하고 수락합니다.
expandchannel	359 페이지 "12.5.9 여러 주소 확장" expandlimit 적용으로 인해 지연된 확장을 수행할 채널을 지정합니다.
expandlimit	359 페이지 "12.5.9 여러 주소 확장" 주소 수가 이 제한을 초과할 경우 받는 메시지를 "오프라인"으로 처리합니다.
filesperjob	356 페이지 "12.5.5 서비스 작업 제한" 단일 작업에서 처리할 대기열 항목의 수입니다.

표 12–13 제어 및 작업 전송 처리 키워드 *(계속)*

키워드	정의
immnonurgent	354 페이지 “12.5.2 지연 전달 날짜 구현” 높음, 중간 및 낮음 우선 순위 메시지를 제출하면 바로 전달을 시작합니다.
master	354 페이지 “12.5.1 채널 방향 설정” 마스터 프로그램에서 사용되는 채널(master)입니다.
maxjobs	356 페이지 “12.5.5 서비스 작업 제한” 채널에 대해 동시에 실행될 수 있는 최대 작업 수입니다.
nodeferred	354 페이지 “12.5.2 지연 전달 날짜 구현” Deferred-delivery: 헤더 행을 수락하지 않음을 지정합니다.
nonurgentbackoff	355 페이지 “12.5.3 전달에 실패한 메시지에 대한 재시도 간격 지정” 낮음 우선 순위 메시지의 재전달 시도 간격입니다.
nonurgentblocklimit	358 페이지 “12.5.7 크기 기반 메시지 우선 순위” 이 크기 이상인 메시지의 우선 순위를 낮음(두 번째 우선 순위 클래스) 이하로 지정합니다. 즉, 해당 메시지는 항상 다음 정기 작업이 처리되는 동안 대기한 후 처리됩니다.
normalbackoff	355 페이지 “12.5.3 전달에 실패한 메시지에 대한 재시도 간격 지정” 중간 우선 순위 메시지의 재전달 시도 간격입니다.
normalblocklimit	358 페이지 “12.5.7 크기 기반 메시지 우선 순위” 이 크기 이상인 메시지의 우선 순위를 낮음으로 지정합니다.
noservice	360 페이지 “12.5.10 서비스 변환 사용” 이 채널로 받는 메시지에 대한 서비스 변환을 CHARSET-CONVERSION을 통해 활성화해야 합니다.
pool	356 페이지 “12.5.4 채널 실행 작업의 처리 풀” 채널에 대한 풀을 지정합니다. 현재 채널에 대한 전달 작업을 풀링해야 하는 풀 이름이 뒤에 와야 합니다.
service	360 페이지 “12.5.10 서비스 변환 사용” CHARSET-CONVERSION 항목에 관계 없이 서비스 변환을 무조건적으로 사용합니다.
slave	354 페이지 “12.5.1 채널 방향 설정” 슬레이브 프로그램에서 사용되는 채널(slave)입니다.
threaddepth	359 페이지 “12.5.8 SMTP 채널 스레드” 다중 스레드 SMTP 클라이언트를 사용하여 새 스레드를 트리거하는 메시지 수입니다.
transactionlimit	연결당 허용되는 메시지 수를 제한합니다.

표 12–13 제어 및 작업 전송 처리 키워드 *(계속)*

키워드	정의
urgentbackoff	355 페이지 “12.5.3 전달에 실패한 메시지에 대한 재시도 간격 지정” 높음 우선 순위 메시지의 재전달 시도 간격입니다.
urgentblocklimit	358 페이지 “12.5.7 크기 기반 메시지 우선 순위” 이 크기 이상인 메시지의 우선 순위를 중간으로 지정합니다.
user	389 페이지 “12.12.3 파이프 채널” 파이프 채널에서 실행할 아이디를 나타내는 데 사용됩니다.

표 12–14 민감도 제한 키워드

키워드	정의
sensitivitycompanyconfidential	374 페이지 “12.7.9 민감도 검사” 메시지에 적용되는 최대 민감도 제한입니다.
sensitivitynormal	376 페이지 “12.7.9 민감도 검사” `Normal`이 메시지에 대한 최대 민감도 제한으로 적용됩니다.
sensitivitypersonal	376 페이지 “12.7.9 민감도 검사” `Personal`이 메시지에 대한 최대 민감도 제한으로 적용됩니다.
sensitivityprivate	376 페이지 “12.7.9 민감도 검사” `Private`이 메시지에 대한 최대 민감도 제한으로 적용됩니다.

표 12–15 메시지, 사용자 할당량, 권한 및 인증 시도의 제한 키워드

키워드	정의
alternatechannel	382 페이지 “12.9.3 크기 제한 또는 수신자 수 제한을 초과하는 메시지 대상 다시 지정 ” alternateblocklimit, alternatelinelimit 및 alternaterecipientlimit에 대한 대체 대상 채널입니다.
alternateblocklimit	382 페이지 “12.9.3 크기 제한 또는 수신자 수 제한을 초과하는 메시지 대상 다시 지정 ” 메시지를 alternativechannel로 보내기 이전에 메시지의 블록 수 제한을 지정합니다.
alternatelinelimit	382 페이지 “12.9.3 크기 제한 또는 수신자 수 제한을 초과하는 메시지 대상 다시 지정 ” 메시지를 alternativechannel로 보내기 이전에 메시지의 행 수 제한을 지정합니다.
alternaterecipientlimit	382 페이지 “12.9.3 크기 제한 또는 수신자 수 제한을 초과하는 메시지 대상 다시 지정 ” 메시지를 alternativechannel로 보내기 이전에 메시지의 수신자 수 제한을 지정합니다.

표 12–15 메시지, 사용자 할당량, 권한 및 인증 시도의 제한 키워드 *(계속)*

키워드	정의
blocklimit	382 페이지 “12.9.2 절대 메시지 크기 제한 지정” 각 메시지에 허용되는 최대 MTA 블록 수입니다.
disconnectbadauthlimit	381 페이지 “12.9.1 성공하지 못한 인증 시도에 대한 제한” 세션의 연결이 끊어지기 전에 세션에 허용되는 인증 시도의 실패 횟수를 제한합니다.
disconnectbadcommandlimit	387 페이지 “12.10.3 세션 제한 설정” 잘못된 세션 명령의 수를 제한합니다.
disconnectrecipientlimit	387 페이지 “12.10.3 세션 제한 설정” 세션 수신자의 수를 제한합니다.
disconnectrejectlimit	387 페이지 “12.10.3 세션 제한 설정” 거부된 수신자의 수를 제한합니다.
disconnecttransactionlimit	387 페이지 “12.10.3 세션 제한 설정” 트랜잭션 수를 제한합니다.
headerlimit	385 페이지 “12.9.8 헤더 크기 제한” 가장 외부에 있는 주 메시지 헤더의 최대 크기를 제한합니다.
holdexquota	384 페이지 “12.9.4 할당량이 초과된 사용자에 대한 메일 전달 처리” 할당량을 초과한 사용자의 메시지를 보관합니다.
holdlimit	359 페이지 “12.5.9 여러 주소 확장” 주소 수가 이 제한을 초과할 경우 받는 메시지를 보관합니다.
linelength	380 페이지 “12.8.4 메시지 행 길이 제한 적용” 채널별로 허용되는 최대 메시지 행 길이를 제한합니다.
linelimit	382 페이지 “12.9.2 절대 메시지 크기 제한 지정” 각 메시지에 허용되는 최대 행 수입니다.
maxblocks	379 페이지 “12.8.3 대용량 메시지 자동 조각화” 메시지에 허용되는 최대 블록 수를 지정합니다.
maxlines	379 페이지 “12.8.3 대용량 메시지 자동 조각화” 메시지에 허용되는 최대 행 수를 지정합니다.
nameparameterlengthlimit	385 페이지 “12.9.6 일반 및 Filename Content-type 및 Content-disposition 매개 변수의 길이 제어” name content-type 및 filename content-disposition 매개 변수가 잘리는 지점을 제어합니다.

표 12–15 메시지, 사용자 할당량, 권한 및 인증 시도의 제한 키워드 *(계속)*

키워드	정의
noblocklimit	382 페이지 "12.9.2 절대 메시지 크기 제한 지정" 각 메시지에 허용되는 MTA 블록 수를 제한하지 않습니다.
noexquota	384 페이지 "12.9.4 할당량이 초과된 사용자에 대한 메일 전달 처리" 할당량을 초과하는 사용자의 메시지를 메시지 발송자에게 반환합니다.
nolinelimit	382 페이지 "12.9.2 절대 메시지 크기 제한 지정" 각 메시지에 허용되는 행 수에 지정된 제한이 없습니다.
nonurgentblocklimit	358 페이지 "12.5.7 크기 기반 메시지 우선 순위" 이 크기 이상인 메시지의 우선 순위를 낮음(두 번째 우선 순위 클래스) 이하로 지정합니다. 즉, 해당 메시지는 항상 다음 정기 작업이 처리되는 동안 대기한 후 처리됩니다.
normalblocklimit	358 페이지 "12.5.7 크기 기반 메시지 우선 순위" 이 크기 이상인 메시지의 우선 순위를 낮음으로 지정합니다.
parameterlengthlimit	385 페이지 "12.9.6 일반 및 Filename Content-type 및 Content-disposition 매개 변수의 길이 제어" 일반 content-type 및 content-disposition 매개 변수가 잘리는 지점을 조절합니다.
recipientcutoff.	385 페이지 "12.9.7 메시지 수신자 제한" 수신자가 이 값을 초과할 경우 메시지를 거부합니다.
recipientlimit	385 페이지 "12.9.7 메시지 수신자 제한" 메시지에 허용되는 수신자 주소 수를 제한합니다.
rejectsmtplonglines	384 페이지 "12.9.5 1000자가 넘는 행이 있는 SMTP 메일 처리" 1000자(CRLF 포함)보다 긴 행을 포함하는 메시지를 거부합니다.
sourceblocklimit	382 페이지 "12.9.2 절대 메시지 크기 제한 지정" 각 받는 메시지에 허용되는 최대 MTA 블록 수입니다.
truncatesmtplonglines	384 페이지 "12.9.5 1000자가 넘는 행이 있는 SMTP 메일 처리" 1000자를 초과하는 행을 자릅니다.
wrapsmtplonglines	384 페이지 "12.9.5 1000자가 넘는 행이 있는 SMTP 메일 처리" 1000자를 초과하는 행을 줄 바꿈합니다.
urgentblocklimit	358 페이지 "12.5.7 크기 기반 메시지 우선 순위" 이 크기 이상인 메시지의 우선 순위를 중간으로 지정합니다.

표 12–16 SMTP 인증, SASL 및 TLS 키워드

키워드	정의
(기능에 대한 자세한 내용은 349 페이지 “12.4.4 SMTP 인증, SASL 및 TLS” 참조)	
authrewrite	341 페이지 “12.4.3 TCP/IP 연결 및 DNS 조회 지원” MTA에서 인증된 메시지 발송자 정보(사용 가능한 경우)를 헤더로 전파하도록 하기 위해 소스 채널에서 사용됩니다.
maysaslserver	349 페이지 “12.4.4 SMTP 인증, SASL 및 TLS” 클라이언트가 SASL 인증 사용을 시도하도록 허용합니다.
maytls	351 페이지 “12.4.8 전송 계층 보안” MTA가 받는 연결에 TLS를 제공하고 보내는 연결에 TLS를 시도하도록 합니다.
maytlsclient	351 페이지 “12.4.8 전송 계층 보안” TLS를 지원하는 SMTP 서버에 메시지를 보낼 경우 MTA SMTP 클라이언트가 TLS 사용을 시도합니다.
maytlsserver	351 페이지 “12.4.8 전송 계층 보안” MTA SMTP 서버에서 STARTTLS 확장 지원을 광고하고 메시지를 받을 때 TLS 사용을 허용합니다.
msexchange	351 페이지 “12.4.7 Microsoft Exchange 게이트웨이 채널 지정” TCP/IP 채널에서 사용되며 이 채널이 Microsoft Exchange 게이트웨이 및 클라이언트와 통신하는 채널임을 MTA에 알려줍니다.
mustsaslserver	349 페이지 “12.4.4 SMTP 인증, SASL 및 TLS” SMTP 서버는 원격 클라이언트가 성공적으로 인증되지 않는 경우 메시지가 수락되지 않습니다.
musttls	351 페이지 “12.4.8 전송 계층 보안” 보내는 연결과 받는 연결 모두에서 TLS 사용을 강제합니다.
musttlsclient	351 페이지 “12.4.8 전송 계층 보안” MTA SMTP 클라이언트는 메시지를 보낼 때 TLS 사용을 강제합니다(MTA가 STARTTLS 명령을 실행하고 해당 명령이 성공해야 함).
musttlsserver	351 페이지 “12.4.8 전송 계층 보안” MTA SMTP 서버에서 STARTTLS 확장 지원을 광고하고 메시지를 받을 때 TLS 사용을 강제합니다.
nomsexchange	341 페이지 “12.4.3 TCP/IP 연결 및 DNS 조회 지원” 기본값입니다.
nosasl	349 페이지 “12.4.4 SMTP 인증, SASL 및 TLS” SASL 인증이 허용 또는 시도되지 않습니다.

표 12–16 SMTP 인증, SASL 및 TLS 키워드 *(계속)*

키워드	정의
nosaslserver	349 페이지 “12.4.4 SMTP 인증, SASL 및 TLS” SASL 인증이 허용되지 않습니다.
notls	351 페이지 “12.4.8 전송 계층 보안” TLS가 허용 또는 시도되지 않습니다.
notlsclient	351 페이지 “12.4.8 전송 계층 보안” 보내는 연결에서 MTA SMTP 클라이언트가 TLS 사용을 시도하지 않습니다(보내는 연결 중에 STARTTLS 명령이 실행되지 않음).
notlsserver	351 페이지 “12.4.8 전송 계층 보안” 받는 연결에서 MTA SMTP 서버의 TLS 사용이 허용되지 않습니다(SMTP 서버에서 STARTTLS 확장을 광고하지 않고 명령 자체가 허용되지도 않음).
saslswitchchannel	349 페이지 “12.4.4 SMTP 인증, SASL 및 TLS” 클라이언트가 SASL을 성공적으로 사용하면 받는 연결이 지정한 채널로 전환되게 합니다.
tlsswitchchannel	351 페이지 “12.4.8 전송 계층 보안” 클라이언트가 TLS 협상에 성공할 경우 받는 연결이 지정한 채널로 전환되게 합니다. 전환할 채널을 지정하는 필수 값을 사용합니다.

표 12–17 SMTP 명령 및 프로토콜 키워드

키워드	정의
(기능에 대한 자세한 내용은 333 페이지 “12.4.2 SMTP 명령 및 프로토콜 지원” 참조)	
allowetrn	337 페이지 “12.4.2.3 ETRN 명령 지원” ETRN 명령을 수락합니다.
blocketrn	337 페이지 “12.4.2.3 ETRN 명령 지원” ETRN 명령을 차단합니다.
checkehlo	336 페이지 “12.4.2.2 EHLO 명령 지원” SMTP 응답 배너를 검사하여 EHLO 또는 HELO 중 어떤 것을 사용할지 여부를 결정합니다.
chunkingclient	350 페이지 “12.4.6 SMTP 청크 지원” Enable server chunking support (default).
chunkingserver	350 페이지 “12.4.6 SMTP 청크 지원” 서버 청크 지원을 활성화합니다(기본값).

표 12–17 SMTP 명령 및 프로토콜 키워드 *(계속)*

키워드	정의
disableetrn	337 페이지 “12.4.2.3 ETRN 명령 지원” ETRN SMTP 명령 지원을 사용하지 않습니다.
domainetrn	337 페이지 “12.4.2.3 ETRN 명령 지원” 도메인을 지정하는 ETRN 명령만 수락합니다.
domainvrfy	338 페이지 “12.4.2.4 VRFY 명령 지원” 전체 주소를 사용하여 VRFY 명령을 실행합니다.
ehlo	336 페이지 “12.4.2.2 EHLO 명령 지원” 초기 연결에 SMTP EHLO 명령을 사용합니다.
eightbit	339 페이지 “12.4.2.7 문자 세트 레이블링 및 8비트 데이터” 채널에서 8비트 문자를 지원합니다.
eightnegotiate	339 페이지 “12.4.2.7 문자 세트 레이블링 및 8비트 데이터” 채널에서 가능할 경우 8비트 전송 사용을 협상합니다.
eightstrict	339 페이지 “12.4.2.7 문자 세트 레이블링 및 8비트 데이터” 협상되지 않은 8비트 데이터를 포함하는 메시지를 거부합니다.
expnallow	338 페이지 “12.4.2.5 EXPN 지원” `DISABLE_EXPAND` SMTP 채널 옵션을 사용하여 SMTP 서버 수준에서 사용 불가능으로 설정했더라도 `EXPN`을 허용합니다.
expndisable	338 페이지 “12.4.2.5 EXPN 지원” `EXPN`을 무조건적으로 사용 불가능하게 합니다.
expndefault	338 페이지 “12.4.2.5 EXPN 지원” SMTP 서버에 사용하도록 설정된 경우 `EXPN`을 허용합니다.
localvrfy	338 페이지 “12.4.2.4 VRFY 명령 지원” 로컬 주소를 사용하여 VRFY 명령을 실행합니다.
mailfromdnsverify	339 페이지 “12.4.2.6 DNS 도메인 확인” `MAIL FROM:` 명령에 사용된 도메인이 DNS에 있는지 여부를 확인합니다.
nochunkingclient	350 페이지 “12.4.6 SMTP 청크 지원” 서버 청크 지원을 비활성화합니다.
nochunkingserver	350 페이지 “12.4.6 SMTP 청크 지원” 서버 청크 지원을 비활성화합니다.

표 12–17 SMTP 명령 및 프로토콜 키워드 (계속)

키워드	정의
noehlo	336 페이지 “12.4.2.2 EHLO 명령 지원” EHLO 명령을 사용하지 않습니다.
nomailfromdnsverify	339 페이지 “12.4.2.6 DNS 도메인 확인” MAIL FROM: 명령에 사용된 도메인이 DNS에 있는지 여부를 확인하지 않습니다.
nosendetrn	337 페이지 “12.4.2.3 ETRN 명령 지원” ETRN 명령을 보내지 않습니다.
nosmtp	336 페이지 “12.4.2.1 채널 프로토콜 선택 및 행 종결 기호” SMTP 프로토콜을 지원하지 않습니다. 기본값입니다.
novrfy	338 페이지 “12.4.2.4 VRFY 명령 지원” VRFY 명령을 실행하지 않습니다.
sendetrn	337 페이지 “12.4.2.3 ETRN 명령 지원” ETRN 명령을 보냅니다.
sevenbit	339 페이지 “12.4.2.7 문자 세트 레이블링 및 8비트 데이터” 8비트 문자를 지원하지 않으므로 8비트 문자를 인코딩해야 합니다.
silentetrn	337 페이지 “12.4.2.3 ETRN 명령 지원” 채널 정보를 반향하지 않고 ETRN 명령을 수락합니다.
smtp	336 페이지 “12.4.2.1 채널 프로토콜 선택 및 행 종결 기호” SMTP 프로토콜을 지원합니다. smtp 키워드는 모든 SMTP 채널에 필수입니다(이 키워드는 smtp_crlf와 동일함).
smtp_cr	336 페이지 “12.4.2.1 채널 프로토콜 선택 및 행 종결 기호” 후행 줄 바꿈(LF) 없이 캐리지 리턴(CR)을 사용하여 행을 종료합니다.
smtp_crlf	336 페이지 “12.4.2.1 채널 프로토콜 선택 및 행 종결 기호” 행을 캐리지 리턴(CR) 줄 바꿈(LF) 시퀀스로 종료해야 합니다.
smtp_crorlf	336 페이지 “12.4.2.1 채널 프로토콜 선택 및 행 종결 기호” 행을 캐리지 리턴(CR), 줄 바꿈(LF) 시퀀스 또는 전체 CRLF를 사용하여 종료해야 합니다.
smtp_lf	336 페이지 “12.4.2.1 채널 프로토콜 선택 및 행 종결 기호” 선행 CR 없이 줄 바꿈(LF)을 사용하여 행을 종료합니다.
streaming	340 페이지 “12.4.2.8 프로토콜 스트리밍” 채널에 연결된 프로토콜에 사용되는 프로토콜 스트리밍의 범위를 제어합니다.

표 12–17 SMTP 명령 및 프로토콜 키워드 *(계속)*

키워드	정의
vrfyallow	338 페이지 “12.4.2.4 VRFY 명령 지원”
	VRFY 명령에 대한 정보 응답을 제공합니다.
vrfydefault	338 페이지 “12.4.2.4 VRFY 명령 지원”
	채널의 HIDE_VERIFY 옵션 설정에 따라 VRFY 명령에 대한 기본 응답을 제공합니다.
vrfyhide	338 페이지 “12.4.2.4 VRFY 명령 지원”
	SMTP VRFY 명령에 대한 위장 응답을 제공합니다.

표 12–18 TCP/IP 연결 및 DNS 조회 지원 키워드

키워드	정의
TCP/IP 연결 및 DNS 조회 지원	
(기능에 대한 자세한 내용은 341 페이지 “12.4.3 TCP/IP 연결 및 DNS 조회 지원” 참조)	
cacheeverything	343 페이지 “12.4.3.2 채널 연결 정보 캐싱”
	모든 연결 정보를 캐시합니다.
cachefailures	343 페이지 “12.4.3.2 채널 연결 정보 캐싱”
	연결 실패 정보만 캐시합니다.
cachesuccesses	343 페이지 “12.4.3.2 채널 연결 정보 캐싱”
	연결 성공 정보만 캐시합니다.
connectalias	364 페이지 “12.6.5 메시지를 대기열에서 제거할 때 주소 다시 쓰기”
	수신자 주소에 나열된 모든 호스트에 전달합니다.
connectcanonical	364 페이지 “12.6.5 메시지를 대기열에서 제거할 때 주소 다시 쓰기”
	MTA가 연결되는 시스템에 대한 호스트 별칭에 연결합니다.
daemon	348 페이지 “12.4.3.10 대상 호스트 선택”
	봉투 주소에 관계 없이 특정 호스트 시스템에 연결합니다.
defaultmx	346 페이지 “12.4.3.5 TCP/IP MX 레코드 지원”
	채널이 네트워크에서 MX 조회를 수행할지 여부를 결정합니다.
defaultnameservers	346 페이지 “12.4.3.6 이름 서버 조회”
	TCP/IP 스택의 이름 서버 선택을 참조합니다.

표 12–18 TCP/IP 연결 및 DNS 조회 지원 키워드 *(계속)*

키워드	정의
forwardcheckdelete	344 페이지 “12.4.3.3 역방향 DNS 조회” 역방향 DNS 조회를 수행한 경우 반환된 이름에 대한 정방향 조회를 수행하여 반환된 IP 번호가 원본과 일치하는지 확인합니다. 원본과 일치하지 않는 경우 이름을 삭제하고 IP 주소를 사용합니다.
forwardchecknone	344 페이지 “12.4.3.3 역방향 DNS 조회” DNS 역방향 조회 후 정방향 조회를 수행하지 않습니다.
forwardchecktag	344 페이지 “12.4.3.3 역방향 DNS 조회” 역방향 DNS 조회를 수행한 경우 반환된 이름에 대한 정방향 조회를 수행하여 반환된 IP 번호가 원본과 일치하는지 확인합니다. 원본과 일치하지 않는 경우 이름에 * 태그를 붙입니다.
identnone	344 페이지 “12.4.3.4 IDENT 조회” IDENT 조회를 수행하지 않고, IP를 호스트 이름으로 변환하고, 호스트 이름과 IP 주소를 모두 `Received:` 헤더에 포함합니다.
identnonelimited	344 페이지 “12.4.3.4 IDENT 조회” IDENT 조회를 수행하지 않고, IP를 호스트 이름으로 변환하지만 채널 전환 중에 호스트 이름을 사용하지 않고, 호스트 이름과 IP 주소를 모두 `Received:` 헤더에 포함합니다.
identnonenumeric	344 페이지 “12.4.3.4 IDENT 조회” IDENT 조회를 수행하지 않거나 IP를 호스트 이름으로 변환하지 않습니다.
identnonesymbolic	344 페이지 “12.4.3.4 IDENT 조회” IDENT 조회를 수행하지 않고 IP를 호스트 이름으로 변환하고 호스트 이름만 `Received:` 헤더에 포함합니다.
identtcp	344 페이지 “12.4.3.4 IDENT 조회” 받는 SMTP 연결에서 IDENT 조회를 수행하고 IP를 호스트 이름으로 변환하고 호스트 이름과 IP 주소를 모두 `Received:` 헤더에 포함합니다.
identtcplimited	344 페이지 “12.4.3.4 IDENT 조회” 받는 SMTP 연결에 IDENT 조회를 수행하고 IP를 호스트 이름으로 변환하지만 채널 전환 중에 호스트 이름을 사용하지 않습니다. 호스트 이름과 IP 주소를 `Received:` 헤더에 포함합니다.
indenttcpnumeric	344 페이지 “12.4.3.4 IDENT 조회” 받는 SMTP 연결에서 IDENT 조회를 수행하지만 IP를 호스트 이름으로 변환하지 않습니다.
identtcpsymbolic	344 페이지 “12.4.3.4 IDENT 조회” 받는 SMTP 연결에서 IDENT 조회를 수행하고 IP를 호스트 이름으로 변환하고 호스트 이름만 `Received:` 헤더에 포함합니다.

표 12–18 TCP/IP 연결 및 DNS 조회 지원 키워드 *(계속)*

키워드	정의
interfaceaddress	343 페이지 “12.4.3.1 TCP/IP 포트 번호 및 인터페이스 주소” 지정된 TCP/IP 인터페이스 주소에 바인딩합니다.
lastresort	346 페이지 “12.4.3.7 마지막 Resort 호스트” 마지막 Resort 호스트를 지정합니다.
mailfromdnsverify	339 페이지 “12.4.2.6 DNS 도메인 확인” `MAIL FROM:` 명령에 사용된 도메인이 DNS에 있는지 여부를 확인합니다.
mx	346 페이지 “12.4.3.5 TCP/IP MX 레코드 지원” TCP/IP 네트워크 및 소프트웨어가 MX 레코드 조회를 지원합니다.
nameservers	346 페이지 “12.4.3.6 이름 서버 조회” TCP/IP 스택의 자체 이름 서버 선택을 참조하는 대신 참조할 이름 서버 목록을 지정합니다. `nameservers`에는 이름 서버에 대한 공백으로 구분된 IP 주소 목록이 필요합니다.
nocache	343 페이지 “12.4.3.2 채널 연결 정보 캐싱” 연결 정보를 캐시하지 않습니다.
nomailfromdnsverify	339 페이지 “12.4.2.6 DNS 도메인 확인” `MAIL FROM:` 명령에 사용된 도메인이 DNS에 있는지 여부를 확인하지 않습니다.
nomx	346 페이지 “12.4.3.5 TCP/IP MX 레코드 지원” TCP/IP 네트워크가 MX 조회를 지원하지 않습니다.
nonrandomemx	346 페이지 “12.4.3.5 TCP/IP MX 레코드 지원” MX 조회를 수행하고 반환된 항목을 같은 우선 순위로 임의화하지 않습니다.
port	343 페이지 “12.4.3.1 TCP/IP 포트 번호 및 인터페이스 주소” SMTP 연결에 대한 기본 포트 번호를 지정합니다. 표준 포트는 25입니다.
randommx	346 페이지 “12.4.3.5 TCP/IP MX 레코드 지원” MX 조회를 수행하고 반환된 항목을 같은 우선 순위로 임의화합니다.
single	348 페이지 “12.4.3.10 대상 호스트 선택” 채널의 각 대상 주소에 대해 별도의 메시지 복사본을 만들도록 지정합니다.
single_sys	348 페이지 “12.4.3.10 대상 호스트 선택” 사용된 각 대상 시스템에 대해 단일 메시지 복사본을 만듭니다.
threaddepth	359 페이지 “12.5.8 SMTP 채널 스레드” 다중 스레드 SMTP 클라이언트를 사용하여 새 스레드를 트리거하는 메시지 수입니다.

표 12–19 기타 키워드

키워드	정의
addresssr	486 페이지 “15.7 SRS(Sender Rewriting Scheme)를 사용하여 SPF에서 전달된 메일 처리” SRS 인코딩을 제어합니다.
deferralrejectlimit	395 페이지 “12.12.8 잘못된 RCPT TO: 주소에 대한 제한 설정 ” 잘못된 RCPT TO: 주소 수에 제한을 설정합니다.
caption	395 페이지 “12.12.9 모니터링 프레임워크에 대한 채널 화면 표시 설정” 모니터링 프레임워크에 대한 짧은 채널 화면 표시 문자열을 설정합니다.
description	395 페이지 “12.12.9 모니터링 프레임워크에 대한 채널 화면 표시 설정” 모니터링 프레임워크에 대한 채널 화면 표시 문자열을 설정합니다.
destinationsrs	486 페이지 “15.7 SRS(Sender Rewriting Scheme)를 사용하여 SPF에서 전달된 메일 처리” SRS 인코딩을 제어합니다.
dispositionchannel	389 페이지 “12.12.1 프로세스 채널 대체” 처음에 전달 상태 알림(DSN)을 대기열에 포함하기 위한 장소로 프로세스 채널을 대체합니다.
destinationfilter	390 페이지 “12.12.4 메일함 필터 파일 위치 지정” 보내는 메시지에 적용할 채널 수준 필터를 지정하기 위해 일반 MTA 채널에서 사용됩 니다.
filter	390 페이지 “12.12.4 메일함 필터 파일 위치 지정” 필터 파일 위치를 설명하는 필수 URL 인수를 가집니다.
noaddresssrs	486 페이지 “15.7 SRS(Sender Rewriting Scheme)를 사용하여 SPF에서 전달된 메일 처리” SRS 인코딩을 제어합니다.
nodestinationfilter	390 페이지 “12.12.4 메일함 필터 파일 위치 지정” 어느 채널 방향에 대해서도 채널 메일함 필터를 사용하지 않습니다.
nodestinationsrs	486 페이지 “15.7 SRS(Sender Rewriting Scheme)를 사용하여 SPF에서 전달된 메일 처리” SRS 인코딩을 제어합니다.
nosourcefilter	390 페이지 “12.12.4 메일함 필터 파일 위치 지정” 소스 채널에 대해 채널 메일함 필터를 사용하지 않습니다.
nosourcesrs	486 페이지 “15.7 SRS(Sender Rewriting Scheme)를 사용하여 SPF에서 전달된 메일 처리” SRS 인코딩을 제어합니다.
nofilter	390 페이지 “12.12.4 메일함 필터 파일 위치 지정” 기본값이며 채널에 대해 사용자 메일함 필터를 사용하지 않는다는 것을 의미합니다.

표 12–19 기타 키워드 (계속)

키워드	정의
notificationchannel	389 페이지 “12.12.1 프로세스 채널 대체” 처음에 MDN(Message Disposition Notification)을 대기열에 포함하기 위한 장소로 프로세스 채널을 대체합니다.
sourcefilter	390 페이지 “12.12.4 메일함 필터 파일 위치 지정” 받는 메시지에 적용할 채널 수준 필터를 지정하기 위해 일반 MTA 채널에서 사용됩니다.
sourcesrs	486 페이지 “15.7 SRS(Sender Rewriting Scheme)를 사용하여 SPF에서 전달된 메일 처리” SRS 인코딩을 제어합니다.
submit	389 페이지 “12.12.2 채널 작업 유형” 채널을 제출 전용 채널로 표시하는 데 사용됩니다.
user	389 페이지 “12.12.3 파이프 채널” 파이프 채널에서 실행할 아이디를 나타내는 데 사용됩니다.

12.4 SMTP 채널 구성

설치 유형에 따라 Messaging Server는 설치 시 여러 SMTP 채널을 제공합니다(아래 표 참조). 이러한 채널은 SMTP over TCP/IP를 구현합니다. 다중 스레드 TCP SMTP 채널은 디스패처의 제어에 따라 실행되는 다중 스레드 SMTP 서버를 포함합니다. 보내는 SMTP 메일은 `tcp_smtp_client` 채널 프로그램에서 처리되고 필요한 경우 작업 제어기의 제어에 따라 실행됩니다.

표 12–20 SMTP 채널

채널	정의
`tcp_local`	원격 SMTP 호스트로부터 인바운드 메시지를 받습니다. 스마트 호스트/방화벽 구성을 사용하는지 여부에 따라 아웃바운드 메시지를 원격 SMTP에 직접 보내거나 스마트 호스트/방화벽 시스템으로 보냅니다.
`tcp_intranet`	인트라넷 내에서 메시지를 주고 받습니다.
`tcp_auth`	`tcp_local`에 대한 전환 채널로 사용되며 중계 차단 제한을 피하도록 인증된 사용자를 tcp_auth 채널로 전환합니다.
`tcp_submit`	예약된 제출 포트 587에서 메시지 제출(일반적으로 사용자 에이전트로부터)을 허용합니다(RFC 2476 참조).
`tcp_tas`	IA 특수 채널이 통합 메시징을 수행하는 사이트에서 사용됩니다.

이 절에 설명된 대로 채널 키워드를 추가하거나 제거하여 이러한 채널의 정의를 수정하거나 새 채널을 만들 수 있습니다. 또한, 옵션 파일을 사용하여 TCP/IP 채널의

다양한 특성을 제어할 수 있습니다. 옵션 파일은 MTA 구성 디렉토리(*msg-svr-base*/config)에 저장하고 *x*_*option*으로 이름을 지정해야 합니다. 여기서 *x*는 채널의 이름입니다. 자세한 내용은 **Sun Java System Messaging Server 6.3 Administration Reference**의 "Option File"을 참조하십시오.

이 절은 다음 내용으로 구성되어 있습니다.

12.4.1 SMTP 채널 옵션 구성

TCP/IP 채널 옵션 파일은 TCP/IP 채널의 다양한 특성을 제어합니다. 채널 옵션 파일은 MTA 구성 디렉토리에 저장하고 *x*_option으로 이름을 지정해야 합니다. 여기서 *x*는 채널의 이름입니다. 예: */msg-svr-base*/config/tcp_local_option

옵션 파일은 하나 이상의 키워드와 관련 값으로 구성됩니다. 예를 들어, 옵션 파일에 DISABLE_EXPAND 키워드를 포함시키고 값을 1로 설정하여 메일링 목록 확장을 비활성화할 수 있습니다.

다른 옵션 파일 키워드를 사용하여 다음을 수행할 수 있습니다.

- 메시지당 허용되는 수신자 수에 대한 제한 설정(ALLOW_RECIPIENTS_PER_TRANSACTION)
- 연결당 허용되는 메시지 수에 대한 제한 설정(ALLOW_TRANSACTIONS_PER_SESSION)
- MTA 로그 파일에 기록되는 정보 유형 세부 조정(LOG_CONNECTION, LOG_TRANPORTINFO)
- 클라이언트 채널 프로그램에 허용되는 최대 동시 아웃바운드 연결 수 지정(MAX_CLIENT_THREADS)

모든 채널 옵션 키워드 및 구문에 대한 자세한 내용은 **Sun Java System Messaging Server 6.3 Administration Reference**를 참조하십시오.

12.4.2 SMTP 명령 및 프로토콜 지원

SMTP 채널이 EHLO, ETRN, EXPN, VRFY 등과 같은 특정 SMTP 명령을 지원하는지 여부를 지정할 수 있습니다. 또한, 채널이 DNS 도메인 확인을 지원하는지 여부, 채널에서 행 종결 기호로 사용되는 문자 등을 지정할 수 있습니다. 이 절은 다음 내용으로 구성되어 있습니다.

표 12–21은 이 절에서 설명하는 키워드를 요약해서 보여 줍니다.

표 12–21 SMTP 명령 및 프로토콜 키워드

채널 키워드	설명
프로토콜 선택 및 행 종결 기호	**채널이 SMTP 프로토콜을 지원하는지 여부를 지정하고 행 종결 기호로 사용되는 문자 시퀀스를 지정합니다.**
`smtp`	SMTP 프로토콜을 지원합니다. `smtp` 키워드는 모든 SMTP 채널에 필수입니다(이 키워드는 `smtp_crorlf`와 동일함).
`nosmtp`	SMTP 프로토콜을 지원하지 않습니다. 기본값입니다.
`smtp_cr`	후행 줄 바꿈(LF) 없이 캐리지 리턴(CR)을 사용하여 행을 종료합니다.
`smtp_crlf`	행을 캐리지 리턴(CR) 줄 바꿈(LF) 시퀀스로 종료해야 합니다.
`smtp_lf`	선행 CR 없이 줄 바꿈(LF)을 사용하여 행을 종료합니다.
`smtp_crorlf`	행을 캐리지 리턴(CR), 줄 바꿈(LF) 시퀀스 또는 전체 CRLF를 사용하여 종료해야 합니다.
EHLO 키워드	**채널에서 EHLO 명령을 처리하는 방법을 지정합니다.**
`ehlo`	초기 연결에 SMTP EHLO 명령을 사용합니다.
`checkehlo`	SMTP 응답 배너를 검사하여 EHLO 또는 HELO 중 어떤 것을 사용할지 여부를 결정합니다.
`noehlo`	EHLO 명령을 사용하지 않습니다.
ETRN 키워드	**채널에서 ETRN 명령(대기열 처리 요청)을 처리하는 방법을 지정합니다.**
`allowetrn`	ETRN 명령을 수락합니다.
`blocketrn`	ETRN 명령을 차단합니다.
`domainetrn`	도메인을 지정하는 ETRN 명령만 수락합니다.
`silentetrn`	채널 정보를 반향하지 않고 ETRN 명령을 수락합니다.
`sendetrn`	ETRN 명령을 보냅니다.

표 12–21 SMTP 명령 및 프로토콜 키워드 (계속)

채널 키워드	설명
nosendetrn	ETRN 명령을 보내지 않습니다.
VRFY 키워드	**채널에서 VRFY 명령을 처리하는 방법을 지정합니다.**
domainvrfy	전체 주소를 사용하여 VRFY 명령을 실행합니다.
localvrfy	로컬 주소를 사용하여 VRFY 명령을 실행합니다.
novrfy	VRFY 명령을 실행하지 않습니다.
vrfyallow	VRFY 명령에 대한 정보 응답을 제공합니다.
vrfydefault	채널의 HIDE_VERIFY 옵션 설정에 따라 VRFY 명령에 대한 기본 응답을 제공합니다.
vrfyhide	SMTP VRFY 명령에 대한 위장 응답을 제공합니다.
EXPN 키워드	채널에서 EXPN 키워드를 처리하는 방법을 지정합니다.
expnallow	DISABLE_EXPAND SMTP 채널 옵션을 사용하여 SMTP 서버 수준에서 사용 불가능으로 설정했더라도 EXPN을 허용합니다.
expndisable	EXPN을 무조건적으로 사용 불가능하게 합니다.
expndefault	SMTP 서버에 사용하도록 설정된 경우 EXPN을 허용합니다. 기본값입니다.
DNS 도메인 확인	**채널에서 DNS 도메인 확인을 수행하는지 여부를 지정합니다.**
mailfromdnsverify	MAIL FROM: 명령에 사용된 도메인이 DNS에 있는지 여부를 확인합니다.
nomailfromdnsverify	MAIL FROM: 명령에 사용된 도메인이 DNS에 있는지 여부를 확인하지 않습니다.
문자 세트 및 8비트 데이터	**채널에서 8비트 데이터를 처리하는 방법을 지정합니다. (주:이러한 키워드는 일반적으로 SMTP 채널에서 사용되지만 모든 종류의 채널과 관련될 수 있습니다.)**
charset7	7비트 텍스트 메시지과 관련된 기본 문자 세트입니다.
charset8	8비트 텍스트 메시지와 관련된 기본 문자 세트입니다.
charsetesc	이스케이프 문자를 포함하는 7비트 텍스트 메시지와 관련된 기본 문자 세트입니다.
eightbit	채널에서 8비트 문자를 지원합니다.
eightnegotiate	채널에서 가능할 경우 8비트 전송 사용을 협상합니다.
eightstrict	채널은 잘못된 8비트 데이터를 포함하는 메시지를 거부해야 합니다.
sevenbit	채널에서 8비트 문자를 지원하지 않으므로 8비트 문자를 인코딩해야 합니다.
프로토콜 스트리밍	**사용할 채널에 대한 프로토콜 스트리밍 범위를 지정합니다.**
streaming	채널에 연결된 프로토콜에 사용되는 프로토콜 스트리밍의 범위를 제어합니다.

12.4.2.1 채널 프로토콜 선택 및 행 종결 기호

키워드: smtp, nosmtp, smtp_crlf , smtp_cr, smtp_crorlf, smtp_lf

smtp 및 nosmtp 키워드는 채널이 SMTP 프로토콜을 지원하는지 여부를 지정합니다. smtp 키워드 또는 이 키워드의 변형 중 하나가 모든 SMTP 채널에 필수입니다.

smtp_crlf, smtp_cr, smtp_crorlf 및 smtp_lf 키워드는 SMTP에서 MTA가 행 종결 기호로 사용하는 문자 시퀀스를 지정하는 데 사용될 수 있습니다. smtp_crlf 키워드는 행을 캐리지 리턴(CR) 줄 바꿈(LF) 시퀀스로 종료해야 함을 의미합니다. smtp_lf 또는 smtp 키워드는 선행 CR 없이 LF를 사용할 수 있음을 의미합니다. 마지막으로 smtp_cr은 후행 LF 없이 CR을 사용할 수 있음을 의미합니다. 이러한 옵션은 받는 메일의 처리에만 영향을 미칩니다.

SMTP 표준은 행 종결 기호로 CRLF를 필요로 하기 때문에 MTA는 항상 표준 CRLF 시퀀스를 생성합니다. 여러 가지 smtp 키워드는 MTA에서 추가 비표준 행 종결 기호를 수락할지 여부만 제어합니다. 예를 들어, MTA가 적법한 SMTP 메시지만 수락하고 비표준행 종결 기호가 있는 메시지를 거부하게 하려면 smtp_crlf를 지정할 수 있습니다.

12.4.2.2 EHLO 명령 지원

키워드: ehlo, noehlo, checkehlo

추가 명령 협상을 허용하도록 SMTP 프로토콜이 확장되었습니다(RFC 1869). 이를 위해 RFC 821의 HELO 명령을 대체하는 새 EHLO 명령이 사용됩니다. 확장 SMTP 서버는 지원하는 확장 목록을 제공하여 EHLO에 응답합니다. 확장되지 않은 서버가 알 수 없는 명령 오류를 반환하면 클라이언트는 이전의 HELO 명령을 대신 보냅니다.

이 폴백 전략은 일반적으로 확장된 서버와 확장되지 않은 서버 모두에 잘 적용됩니다. 그러나 RFC 821에 따라 SMTP를 구현하지 않은 서버에서는 문제가 발생할 수 있습니다. 특히, 일부 비호환 서버에서는 알 수 없는 명령을 수신할 경우 연결을 끊는 것으로 알려져 있습니다.

SMTP 클라이언트는 서버에서 HELO를 수신할 때 연결을 끊는 경우 다시 연결한 다음 EHLO를 사용하는 전략을 구현합니다. 원격 서버에서 EHLO를 수신할 때 연결을 끊고 문제 상태로 전환하는 경우에는 이 전략을 적용할 수 없습니다.

이러한 상황에 대처하기 위해 ehlo, noehlo 및 checkehlo 채널 키워드가 제공됩니다. ehlo 키워드는 모든 초기 연결 시도에서 EHLO 명령을 사용하도록 MTA에 지시합니다. noehlo 키워드는 모든 EHLO 명령 사용을 비활성화합니다. checkehlo 키워드는 문자열 "ESMTP"에 대해 원격 SMTP 서버가 반환하는 응답 배너를 테스트합니다. 이 문자열이 있는 경우 EHLO가 사용되고, 없는 경우 HELO가 사용됩니다. 기본 동작은 모든 초기 연결 시도에서 EHLO를 사용하는 것입니다. 배너 행에 문자열 "fire away"가 없는 경우에는 HELO가 사용됩니다. ehlo 및 checkehlo 키워드로부터 발생되는 동작 사이에 위치하는 이 기본 동작에 해당하는 키워드가 없습니다.

12.4.2.3 ETRN 명령 지원

키워드: allowetrn , blocketrn , disableetrn , domainetrn , silentetrn , sendetrn, nosendetrn, novrfy

RFC 1985에 정의되어 있는 ETRN 명령은 SMTP 클라이언트와 서버가 상호 작용하여 서버에서 지정된 호스트로 보낼 메시지 대기열 처리를 시작할 수 있도록 SMTP 서비스를 확장합니다.

SMTP 클라이언트는 ETRN을 사용하여 원격 SMTP 서버에서 SMTP 클라이언트에 보낼 메시지 대기열의 처리를 시작하도록 요청할 수 있습니다. 즉, ETRN을 사용하면 원격 SMTP 시스템에서 수신하는 메시지에 대해 원격 SMTP 시스템 "폴링"을 구현할 수 있습니다. 이 방법은 서로 일시적으로만 연결하는 시스템에 유용할 수 있습니다. 예를 들어, 전화 접속을 통해서만 인터넷에 연결하는 다른 사이트에 대해 보조 MX(Mail eXchange) 호스트로 설정되는 사이트가 있습니다. 이 명령을 사용하여 원격 전화 접속 서버의 메일 전달 요청을 허용합니다.

SMTP 클라이언트는 SMTP ETRN 명령줄에서 메시지를 보낼 시스템의 이름(일반적으로 SMTP 클라이언트 시스템 자체 이름)을 지정합니다. 원격 SMTP 서버가 ETRN 명령을 지원하는 경우 해당 서버는 개별 프로세스를 트리거하여 명명된 시스템에 다시 연결하고 해당 명명된 시스템에 대해 전달 대기 중인 메시지를 보냅니다.

ETRN 명령에 대한 응답

allowetrn, blocketrn, domainetrn 및 silentetrn 키워드는 보내는 SMTP 클라이언트가 MTA에 MTA 대기열의 메시지를 전달하도록 요청하는 ETRN 명령을 실행할 때 MTA 응답을 제어합니다.

기본적으로 MTA는 모든 ETRN 명령을 수락하려고 시도합니다. 즉, allowetrn 키워드가 사용됩니다. 채널 정의에 blocketrn 키워드를 포함하여 MTA에서 ETRN 명령을 수락하지 않도록 지정할 수 있습니다.

silentetrn 키워드를 포함하여 도메인과 일치하고 MTA에서 실행할 채널의 이름을 반향하지 않고 모든 ETRN 명령을 수락하도록 지정할 수 있습니다. domainetrn 키워드는 MTA가 도메인을 지정하는 ETRN 명령만 사용하도록 지정합니다. 또한, 도메인과 일치하고 MTA에서 실행할 채널의 이름을 에코하지 않도록 지정합니다.

disableetrn은 ETRN 명령을 전혀 지원하지 않습니다. ETRN은 SMTP 서버에서 지원되는 명령으로 광고되지 않습니다.

ETRN 명령 보내기

sendetrn 및 nosendetrn 채널 키워드는 MTA가 SMTP 연결을 시작할 때 ETRN 명령을 보내는지 여부를 제어합니다. 기본값은 nosendetrn이며 MTA가 ETRN 명령을 보내지 않음을 의미합니다. sendetrn 키워드는 원격 SMTP 서버가 ETRN을 지원할 경우 ETRN 명령을 보내도록 MTA에 지시합니다. sendetrn 키워드는 메시지가 전달 시도를 받도록 요청하는 시스템 이름 앞에 입력해야 합니다.

12.4.2.4 VRFY 명령 지원

키워드: domainvrfy, localvrfy, vrfyallow, vrfydefault, vrfyhide

VRFY 명령을 사용하면 SMTP 클라이언트가 SMTP 서버에 특정 아이디에 대한 메일이 있는지 확인하라는 요청을 서버에 보낼 수 있습니다. VRFY 명령은 RFC 821에 정의되어 있습니다.

서버는 사용자가 로컬 사용자인지 여부, 메일 전달 여부 등을 나타내는 응답을 보냅니다. 250 응답은 아이디가 로컬임을 나타내고, 251 응답은 아이디가 로컬이 아니지만 서버가 메시지를 전달할 수 있음을 나타냅니다. 서버 응답에는 메일함 이름이 포함되어 있습니다.

VRFY 명령 보내기

정상적인 환경에서는 VRFY 명령을 SMTP 대화 상자의 일부로 실행할 이유가 없습니다. SMTP RCPT TO 명령은 VRFY 명령과 동일한 기능을 수행하고 해당 오류를 반환해야 합니다. RCPT TO에 주소를 받아들인 다음 나중에 바운스할 수 있는 서버가 존재하지만, 이러한 서버는 VRFY 명령의 일부로 보다 집중적인 검사를 수행합니다.

기본적으로 MTA는 VRFY 명령을 보내지 않습니다(novrfy 키워드 사용).

필요한 경우 채널 정의에 domainvrfy 또는 localvrfy 키워드를 포함하여 SMTP VRFY 명령을 실행하도록 MTA를 구성할 수 있습니다. domainvrfy 키워드는 전체 주소(user@host)를 인수로 갖는 VRFY 명령이 실행되게 합니다. localvrfy 키워드는 MTA에서 주소의 로컬 부분(user)만 사용하여 VRFY 명령이 실행되게 합니다.

VRFY 명령에 대한 응답

vrfyallow, vrfydefault 및 vrfyhide 키워드는 보내는 SMTP 클라이언트가 SMTP VRFY 명령을 실행할 때 SMTP 서버의 응답을 제어합니다.

vrfyallow 키워드는 자세한 정보 응답을 실행하도록 MTA에 지시합니다. vrfydefault는 HIDE_VERIFY=1 채널 옵션이 지정되어 있지 않은 경우 자세한 정보 응답을 제공하도록 MTA에 지시합니다. vrfyhide 키워드는 모호한 응답만 생성하도록 MTA에 지시합니다. 이러한 키워드는 동일한 SMTP 서버를 통해 처리되는 모든 받는 TCP/IP 채널에 일반적으로 적용되는 HIDE_VERIFY 옵션과 달리 VRFY 응답을 채널 단위로 제어할 수 있습니다.

12.4.2.5 EXPN 지원

키워드: expnallow , expndisable , expndefault

expnallow는 DISABLE_EXPAND SMTP 채널 옵션을 사용하여 SMTP 서버 수준에서 사용 불가능으로 설정했더라도 EXPN을 허용합니다. expndisable은 EXPN을 무조건적으로 사용 불가능하게 합니다. expndefault는 SMTP 서버에 사용하도록 설정된 경우 EXPN을 허용합니다(기본값). 확장은 목록 단위로 비활성화할 수 있지만 서버 수준에서 비활성화하면 목록별 설정이 부적절하게 됩니다.

12.4.2.6 DNS 도메인 확인

키워드: mailfromdnsverify, nomailfromdnsverify

받는 TCP/IP 채널에서 mailfromdnsverify를 설정하면 MTA는 DNS의 항목이 SMTP MAIL FROM 명령에 사용된 도메인에 있는지 확인하여 해당 항목이 없는 경우 메시지를 거부합니다. 기본값인 nomailfromdnsverify는 확인이 수행되지 않음을 의미합니다. 반송 주소 도메인에 대해 DNS 확인을 수행하면 일부 유효한 메시지가 거부될 수 있습니다. 예를 들어, 도메인 이름을 아직 등록하지 않은 합법적인 사이트에서 또는 DNS에 잘못된 정보가 있는 경우 거부됩니다. 이는 RFC 1123, Requirements for Internet Hosts에 명시되어 있는 전자 메일 수락 및 전달에 있어 관대함을 요구하는 조항에 위반됩니다. 그러나 몇몇 사이트에서는 존재하지 않는 도메인으로부터 위조된 전자 메일 주소로 UBE(Unsolicited Bulk Email)를 받는 경우 그러한 검사를 수행할 수 있습니다.

COM 및 ORG 최상위 도메인에서 DNS 와일드카드 항목이 소개되어 mailfromdnsverify가 별로 유용하지 않게 되었기 때문에 mailfromdnsverify 코드가 수정되었습니다. DNS가 하나 이상의 A 레코드를 반환하면 이러한 값은 새 MTA 옵션 BLOCKED_MAIL_FROM_IPS에 지정된 도메인 리터럴과 비교됩니다. 일치하는 값이 발견되면 도메인은 유효하지 않은 것으로 간주됩니다. 올바른 동작을 복원하기 위한 현재의 올바른 설정은 다음과 같습니다.

```
BLOCKED_MAIL_FROM_IPS=[64.94.110.11]
```

이 옵션의 기본값은 빈 문자열입니다.

12.4.2.7 문자 세트 레이블링 및 8비트 데이터

키워드: charset7, charset8, charsetesc, sevenbit, eightbit, eightnegotiate, eightstrict

문자 세트 레이블링

MIME 사양은 일반 텍스트 메시지에 사용되는 문자 세트를 레이블링하는 기법을 제공합니다. 특히 charset= 매개 변수를 Content-type: 의 헤더 행으로 지정할 수 있습니다. US-ASCII(기본값), ISO-8859-1, ISO-8859-2 등을 포함하여 다양한 문자 세트 이름이 MIME에 정의되어 있습니다.

일부 기존 시스템과 사용자 에이전트는 이러한 문자 세트 레이블 생성을 위한 기법을 제공하지 않기 때문에 일반 텍스트 메시지가 제대로 레이블링되지 않을 수 있습니다. charset7, charset8 및 charsetesc 채널 키워드는 문자 세트 레이블링이 부족한 메시지 헤더에 삽입할 문자 세트 이름을 지정할 수 있도록 채널 단위 기법을 제공합니다. 각 키워드에는 문자 세트 이름을 지정하는 단일 인수가 필요합니다. 이름의 유효성은 검사하지 않습니다. MTA 테이블 디렉토리에 있는 charsets.txt 문자 세트 정의 파일에 지정된 문자 세트에 대해서만 문자 세트 변환을 수행할 수 있습니다. 가능하면 이 파일에 정의된 이름을 사용해야 합니다.

charset7 문자 세트 이름은 메시지에 7비트 문자만 포함되어 있는 경우에 사용되며, charset8 문자 세트 이름은 메시지에 8비트 데이터가 있는 경우에 사용되고, charsetesc는 7비트 데이터만 포함하는 메시지에 이스케이프 문자도 포함하게 되는 경우에 사용됩니다. 해당 키워드를 지정하지 않으면 Content-type: 헤더 행으로 어떤 문자 세트도 지정되지 않습니다.

또한, charset8 키워드는 8비트 문자는 무조건적으로 부적합한 메시지 헤더에서 8비트 문자의 MIME 인코딩을 제어합니다. MTA는 일반적으로 charset8 값이 지정되어 있지 않은 경우 메시지 헤더에 있는 (잘못된) 8비트 데이터를 UNKNOWN charset으로 레이블링하여 MIME 인코딩합니다.

이러한 문자 세트 사양은 기존 레이블을 대체하지 않습니다. 즉, 메시지에 문자 세트 레이블이 이미 있거나 메시지 유형이 텍스트가 아닌 경우에는 적용되지 않습니다. 이 문자 세트 사양은 MTA 로컬 채널을 다음과 같이 레이블링하는 데 적합합니다.

```
l ... charset7 US-ASCII charset8 ISO-8859-1 ...
hostname
```

메시지에 Content-type 헤더가 없는 경우에 추가됩니다. 또한, 이 키워드는 MIME-version: 헤더 행을 추가합니다(없는 경우).

charsetesc 키워드는 이스케이프 문자를 포함하는 일본어 또는 한국어 문자 세트를 사용하는 레이블링되지 않은 메시지를 받는 채널에서 특히 유용합니다.

8비트 데이터

일부 전송에서는 127(십진수)보다 더 큰 서수 값을 갖는 문자의 사용을 제한합니다. 일부 SMTP 서버는 높은 비트를 제거하기 때문에 이 8비트 범위에 속하는 문자를 사용하는 메시지를 제대로 해석하지 못합니다.

Messaging Server는 문제가 있는 8비트 문자가 메시지에 직접 표시되지 않도록 해당 메시지를 자동으로 인코딩하는 기능을 제공합니다. 이 인코딩은 sevenbit 키워드를 지정하여 지정된 채널의 대기열에 포함된 모든 메시지에 적용할 수 있습니다. 이러한 제한이 없는 경우 채널에 eightbit 표시를 해야 합니다.

"원격 SMTP 서버가 8비트를 허용하는 SMTP 확장을 지원한다고 명시하지 않은 경우" SMTP 프로토콜에서 8비트를 사용할 수 없습니다.확장 SMTP와 같은 일부 전송에서는 8비트 문자를 전송할 수 있는지 확인할 수 있도록 협상을 지원하는 경우도 있습니다. 따라서 협상에 실패할 경우 eightnegotiate 키워드를 사용하여 메시지를 인코딩하도록 채널에 지시하는 것이 좋습니다. 이 키워드는 모든 채널의 기본값입니다. 이 경우 협상을 지원하지 않는 채널에서는 8비트 데이터를 전송할 수 없는 것으로 간주합니다.

eightstrict 키워드는 협상되지 않은 8비트 데이터가 포함된 받는 메시지를 거부하도록 Messaging Server에 지시합니다.

12.4.2.8 프로토콜 스트리밍

키워드: streaming

일부 메일 프로토콜에서는 스트리밍 작업을 지원합니다. 이것은 MTA가 한 번에 여러 작업에 대한 명령을 실행하고 각 작업에 대한 응답이 일괄적으로 도착할 때까지 기다릴 수 있음을 의미합니다. `streaming` 키워드는 채널과 연결된 프로토콜에서 사용되는 프로토콜 스트리밍의 범위를 제어합니다. 이 키워드에는 정수 매개 변수가 필요합니다. 매개 변수를 해석하는 방법은 사용 중인 프로토콜에 따라 다릅니다.

정상적인 환경에서는 SMTP 파이프라인 확장을 사용하여 사용 가능한 스트리밍 범위를 협상합니다. 따라서, 정상적인 환경에서는 이 키워드를 절대 사용하지 마십시오.

사용 가능한 스트리밍 값의 범위는 0부터 3까지입니다. 값 0은 스트리밍을 지정하지 않고, 값 1은 RCPT TO 명령 그룹을 스트리밍하고, 값 2는 MAIL FROM/RCPT TO를 스트리밍하고, 값 3은 HELO/MAIL FROM/RCPT TO 또는 RSET/MAIL FROM/RCPT TO 스트리밍을 사용합니다. 기본값은 0입니다.

12.4.3 TCP/IP 연결 및 DNS 조회 지원

서버에서 TCP/IP 연결 및 주소 조회를 처리하는 방법에 대한 정보를 지정할 수 있습니다. 이 절은 다음 내용으로 구성되어 있습니다.

표 12–22는 이 절에서 설명하는 TCP/IP 연결 및 DNS 조회 키워드를 보여줍니다.

표 12–22 TCP/IP 연결 및 DNS 조회 키워드

채널 키워드	설명
포트 선택 및 인터페이스 주소	**SMTP 연결에 대한 기본 포트 번호 및 인터페이스 주소를 지정합니다.**
`port`	SMTP 연결에 대한 기본 포트 번호를 지정합니다. 표준 포트는 25입니다.
`interfaceaddress`	지정된 TCP/IP 인터페이스 주소에 바인딩합니다.
캐시 키워드	**연결 정보를 캐시하는 방법을 지정합니다.**
`cacheeverything`	모든 연결 정보를 캐시합니다.
`cachefailures`	연결 실패 정보만 캐시합니다.

표 12–22 TCP/IP 연결 및 DNS 조회 키워드 *(계속)*

채널 키워드	설명
cachesuccesses	연결 성공 정보만 캐시합니다.
nocache	연결 정보를 캐시하지 않습니다.
역방향 DNS 조회	**받는 SMTP 연결에서 역방향 DNS 조회를 처리하는 방법을 지정합니다.**
forwardcheckdelete	역방향 DNS 조회를 수행한 경우 반환된 이름에 대한 정방향 조회를 수행하여 반환된 IP 번호가 원본과 일치하는지 확인합니다. 원본과 일치하지 않는 경우 이름을 삭제하고 IP 주소를 사용합니다.
forwardchecknone	DNS 역방향 조회 후 정방향 조회를 수행하지 않습니다.
forwardchecktag	역방향 DNS 조회를 수행한 경우 반환된 이름에 대한 정방향 조회를 수행하여 반환된 IP 번호가 원본과 일치하는지 확인합니다. 원본과 일치하지 않는 경우 이름에 * 태그를 붙입니다.
IDENT 조회/DNS 역방향 조회	**받는 SMTP 연결에서 IDENT 조회 및 DNS 역방향 조회를 처리하는 방법을 지정합니다.**
identnone	IDENT 조회를 수행하지 않고 IP를 호스트 이름으로 변환하고 호스트 이름과 IP 주소를 모두 Received: 헤더에 포함합니다.
identnonelimited	IDENT 조회를 수행하지 않고 IP를 호스트 이름으로 변환하지만 채널 전환 중에 호스트 이름을 사용하지 않고 호스트 이름과 IP 주소를 모두 Received: 헤더에 포함합니다.
identnonenumeric	IDENT 조회를 수행하지 않거나 IP를 호스트 이름으로 변환하지 않습니다.
identnonesymbolic	IDENT 조회를 수행하지 않고 IP를 호스트 이름으로 변환하고 호스트 이름만 Received: 헤더에 포함합니다.
identtcp	받는 SMTP 연결에서 IDENT 조회를 수행하고 IP를 호스트 이름으로 변환하고 호스트 이름과 IP 주소를 모두 Received: 헤더에 포함합니다.
identtcplimited	받는 SMTP 연결에 IDENT 조회를 수행하고 IP를 호스트 이름으로 변환하지만 채널 전환 중에 호스트 이름을 사용하지 않습니다. 호스트 이름과 IP 주소를 모두 Received: 헤더에 포함합니다.
indenttcpnumeric	받는 SMTP 연결에서 IDENT 조회를 수행하지만 IP를 호스트 이름으로 변환하지 않습니다.
identtcpsymbolic	받는 SMTP 연결에서 IDENT 조회를 수행하고 IP를 호스트 이름으로 변환하고 호스트 이름만 Received: 헤더에 포함합니다.
MX 레코드 지원 및 TCP/IP 이름 서버	**채널에서 MX 레코드 조회를 지원하는지 여부와 지원 방법을 지정합니다.**
mx	TCP/IP 네트워크 및 소프트웨어가 MX 레코드 조회를 지원합니다.
nomx	TCP/IP 네트워크가 MX 조회를 지원하지 않습니다.
defaultmx	채널이 네트워크에서 MX 조회를 수행할지 여부를 결정합니다.
randommx	MX 조회를 수행하고 반환된 항목을 같은 우선 순위로 임의화합니다.

표 12–22 TCP/IP 연결 및 DNS 조회 키워드 *(계속)*

채널 키워드	설명
nonrandomemx	MX 조회를 수행하고 반환된 항목을 같은 우선 순위로 임의화하지 않습니다.
nameservers	TCP/IP 스택의 자체 이름 서버 선택을 참조하는 대신 참조할 이름 서버 목록을 지정합니다. nameservers에는 이름 서버에 대한 공백으로 구분된 IP 주소 목록이 필요합니다.
defaultnameservers	TCP/IP 스택의 이름 서버 선택을 참조합니다.
lastresort	마지막 Resort 호스트를 지정합니다.
전환 키워드	**받는 메일에 대한 대체 채널 선택을 제어합니다.**
allowswitchchannel	switchchannel 채널에서 이 채널로 전환을 허용합니다.
noswitchchannel	서버 채널을 유지합니다. 원본 호스트와 연결된 채널을 전환하지 않고, 채널 전환을 허용하지 않습니다.
switchchannel	서버 채널에서 원본 호스트와 연결된 채널로 전환합니다.
userswitchchannel	사용자 또는 도메인 설정을 기준으로 소스 채널을 전환합니다.
tlsswitchchannel	TLS 협상이 성공하면 다른 채널로 전환합니다.
saslswitchchannel	SASL 인증이 성공할 경우 다른 채널로 전환합니다.
대상 호스트 선택 및 메시지 복사본 저장소	**대상 호스트 시스템과 메시지 복사본의 저장 방법을 지정합니다.**
daemon	봉투 주소에 관계 없이 특정 호스트 시스템에 연결합니다.
single	채널의 각 대상 주소에 대해 별도의 메시지 복사본을 만들도록 지정합니다.
single_sys	사용된 각 대상 시스템에 대해 단일 메시지 복사본을 만듭니다.

12.4.3.1 TCP/IP 포트 번호 및 인터페이스 주소

키워드: port, interfaceaddress

SMTP over TCP/IP 채널은 메시지를 보낼 때 일반적으로 포트 25에 연결합니다. port 키워드를 사용하여 비표준 포트에 연결하도록 SMTP over TCP/IP 채널에 지시할 수 있습니다. 이 키워드는 SMTP 연결을 위해 MTA가 수신하는 포트를 제어하는 PORT 디스패처 옵션을 보완합니다.

interfaceaddress 키워드는 TCP/IP 채널이 아웃바운드 연결에 대한 소스 주소로 바인딩하는 주소를 제어합니다. 즉, 여러 인터페이스 주소를 갖는 시스템에서 이 키워드는 MTA가 SMTP 메시지를 보낼 때 소스 IP 주소로 사용하는 주소를 제어합니다. 이 키워드는 받는 연결 및 메시지를 받기 위해 TCP/IP 채널이 수신하는 인터페이스 주소를 제어하는 INTERFACE_ADDRESS 디스패처 옵션을 보완합니다.

12.4.3.2 채널 연결 정보 캐싱

키워드: cacheeverything, nocache, cachefailures, cachesuccesses

SMTP 프로토콜을 사용하는 채널은 이전 연결 시도 내역이 들어 있는 캐시를 유지 관리합니다. 이 캐시를 사용하면 액세스할 수 없는 호스트에 여러 번 다시 연결할 필요가 없으므로 시간이 낭비되거나 다른 메시지가 지연되지 않습니다. 캐시는 프로세스 단위 캐시이며 아웃바운드 SMTP 전달 채널을 한 번 실행하는 동안에만 지속됩니다.

캐시는 일반적으로 연결 성공과 연결 실패를 모두 기록합니다. 성공한 연결 시도는 후속 실패를 오프셋하기 위해 기록됩니다. 그러나, 이전에는 성공했으나 지금은 실패한 호스트에서는 연결을 시도한 적이 없거나 이전에 실패했던 호스트에서처럼 다른 연결을 시도하기 이전에 지연 기간이 보장되지 않습니다.

MTA에 사용되는 캐싱 전략이 모든 상황에 적합한 것은 아닙니다. 따라서 MTA 캐시 조정을 위해 채널 키워드가 제공됩니다.

`cacheeverything` 키워드는 모든 형식의 캐싱을 사용하며 기본값입니다. `nocache` 키워드는 모든 캐싱을 사용하지 않습니다.

`cachefailures` 키워드는 연결 실패 캐싱만 사용하며 `cacheeverything` 키워드의 경우보다 재시도가 제한됩니다. 마지막으로 `cachesuccesses`는 연결 성공만 캐시합니다. 이 마지막 키워드는 SMTP 채널의 `nocache` 키워드와 효과가 동일합니다.

12.4.3.3 역방향 DNS 조회

키워드: `forwardchecknone`, `forwardchecktag`, `forwardcheckdelete`

`forwardchecknone`, `forwardchecktag` 및 `forwardcheckdelete` 채널 키워드는 역방향 DNS 조회 수행 효과를 수정할 수 있습니다. 이러한 키워드는 MTA가 DNS 역방향 조회를 사용하여 찾은 IP 이름에 대해 정방향 조회를 수행하는지 여부를 제어할 수 있으며, 그러한 정방향 조회를 요청받을 경우 IP 이름 정방향 조회가 연결의 원본 IP 번호와 일치하지 않을 때 MTA에서 수행할 작업을 지정합니다.

`forwardchecknone` 키워드는 기본값이며 정방향 조회가 수행되지 않음을 의미합니다. `forwardchecktag` 키워드는 각 역방향 조회 이후에 정방향 조회를 수행하고, 정방향 조회를 사용하여 찾은 IP 번호가 원본 연결의 IP 번호와 일치하지 않을 경우 해당 IP 이름에 별표(*) 태그를 붙이도록 MTA에 지시합니다. `forwardcheckdelete` 키워드는 각 역방향 조회 이후에 정방향 조회를 수행하고 해당 이름에 대한 정방향 조회가 원본 연결 IP 주소와 일치하지 않는 경우 역방향 조회에서 반환된 이름을 무시(삭제)하도록 MTA에 지시합니다. 이 경우 MTA는 원본 IP 주소를 대신 사용합니다.

주 – 순방향 조회가 원본 IP 주소와 일치하지 않는 것이 정상인 사이트가 있습니다. 이러한 사이트에서는 여러 IP 주소에 보다 "일반적인" IP 이름을 사용합니다.

12.4.3.4 IDENT 조회

키워드: `identnone`, `identnonelimited`, `identtnonnumeric`, `identnonesymbolic`, `identtcp` , `identtcpnumeric`, `identtcpsymbolic`, `identtcplimited`

IDENT 키워드는 MTA에서 IDENT 프로토콜을 사용하여 연결 및 조회를 처리하는 방법을 제어합니다. IDENT 프로토콜은 RFC 1413에 설명되어 있습니다.

identtcp, identtcpsymbolic 및 identtcpnumeric 키워드는 IDENT 프로토콜을 사용하여 연결 및 조회를 수행하도록 MTA에 지시합니다. IDENT 프로토콜에서 가져온 정보(일반적으로 SMTP에 연결한 아이디)는 메시지의 Received: 헤더에 다음과 같이 삽입됩니다.

- identtcp는 받는 IP 번호에 해당하는 호스트 이름(DNS 역방향 조회에서 보고됨)과 IP 번호 자체를 삽입합니다.
- identtcpsymbolic은 받는 IP 번호에 해당하는 호스트 이름(DNS 역방향 조회에서 보고됨)을 삽입합니다. IP 번호 자체는 Received:header.
- identtcpnumeric은 실제 받는 IP 번호를 삽입합니다. IP 번호에 대한 DNS 역방향 조회는 수행되지 않습니다.

주 – identtcp, identtcpsymbolic 또는 identtcpnumeric에 의한 IDENT 조회가 유효하려면 원격 시스템에서 IDENT 서버를 실행해야 합니다.

IDENT 쿼리를 시도하면 성능이 감소될 수 있음에 유의하십시오. 점차적으로 라우터는 자신이 구성하지 않은 포트에 대해 시도된 연결에 "블랙홀"을 생성합니다. IDENT 쿼리에서 이러한 현상이 발생할 경우 MTA는 연결 시간 초과(TCP/IP 스택 제어 시간 초과, 일반적으로 1분 또는 2분)가 발생할 때까지는 다시 수신하지 않습니다.

identtcp, indenttcplimited 또는 identtcpsymbolic을 identtcpnumeric과 비교하는 경우 또 다른 성능 요소가 있습니다. identtcp, identtcplimited 또는 identtcpsymbolic으로 호출되는 DNS 역방향 조회는 사용자에게 보다 친숙한 호스트 이름을 가져오기 위해 추가 오버헤드를 발생시킵니다.

identnone 키워드는 IDENT 조회를 사용하지 않지만 IP를 호스트 이름으로 변환하도록 지정하고 IP 번호와 호스트 이름 모두 메시지의 Received: 헤더에 포함합니다.

identnonesymbolic 키워드는 IDENT 조회를 사용하지 않지만 IP를 호스트 이름으로 변환하며 호스트 이름만 메시지의 Received: 헤더에 포함합니다.

identnonenumeric 키워드는 이 IDENT 조회를 사용하지 않고 일반 DNS 역방향 조회에서 IP 번호를 호스트 이름으로 변환하지 못하게 하며, Received: 헤더에서 사용자가 쉽게 접근하기 어려운 정보를 삭제하여 성능을 향상시킵니다. 기본값입니다.

identtcplimited 및 identnonelimited 키워드는 IDENT 조회, 역방향 DNS 조회 및 Received: 헤더에 표시되는 정보에 있어서 각각 identtcp 및 identnone과 동일한 효과를 가집니다. identtcplimited 또는 identnonelimited에서는 DNS 역방향 조회의 호스트 이름 확인 성공 여부에 관계 없이 switchchannel 키워드의 사용에 따른 채널 전환의 기초로 항상 IP 문자 주소를 사용한다는 점만 다릅니다.

12.4.3.5 TCP/IP MX 레코드 지원

키워드: mx, nomx, defaultmx , randommx, nonrandommx

TCP/IP 네트워크에서 MX(메일 전달) 레코드 사용을 지원하는 경우도 있고 지원하지 않는 경우도 있습니다. 일부 TCP/IP 채널 프로그램에서는 MTA 시스템이 연결된 네트워크에서 MX 레코드를 제공하지 않는 경우 MX 레코드를 사용하지 않도록 구성할 수 있습니다. mx, nomx, defaultmx, randommx, nonrandommx 키워드는 MX 레코드 지원을 제어합니다.

randommx 키워드는 MX 조회를 수행하고 우선 순위가 동일한 MX 레코드 값을 임의의 순서로 처리하도록 지정합니다. nonrandommx 키워드는 MX 조회를 수행하고 우선 순위가 동일한 MX 레코드 값을 받은 순서대로 처리하도록 지정합니다.

mx 키워드는 nonrandommx 키워드에 해당하며 이후의 릴리스에서는 randommx에 해당하는 키워드로 변경될 수 있습니다. nomx 키워드는 MX 조회를 사용하지 않습니다. defaultmx 키워드는 네트워크에서 MX 레코드를 지원할 경우 mx를 사용하도록 지정합니다. defaultmx 키워드는 MX 조회를 지원하는 채널의 기본값입니다.

12.4.3.6 이름 서버 조회

키워드: nameservers, defaultnameservers

이름 서버 조회를 수행할 경우 TCP/IP 스택의 자체 이름 서버 선택을 참조하는 대신 nameservers 채널 키워드를 사용하여 이름 서버 목록을 지정할 수 있습니다. nameservers 키워드에는 다음 예에 표시된 것처럼 이름 서버에 대한 공백으로 구분된 IP 주소 목록이 필요합니다.

```
nameservers 1.2.3.1 1.2.3.2
```

기본값인 defaultnameservers는 TCP/IP 스택의 자체 이름 서버 선택을 사용함을 의미합니다.

UNIX에서 이름 서버 조회를 수행하지 않게 하려면 nsswitch.conf 파일을 수정하면 됩니다. NT에서는 TCP/IP 구성을 수정하십시오.

12.4.3.7 마지막 Resort 호스트

키워드: lastresort

lastresort 키워드는 다른 모든 연결 시도가 실패할 경우에 연결할 호스트를 지정하는 데 사용됩니다. 실제로 이 키워드는 마지막 Resort MX 레코드 역할을 합니다. 또한 SMTP 채널에서만 유효합니다.

이 키워드에는 "마지막 Resort 시스템"의 이름을 지정하는 단일 매개 변수가 필요합니다.예를 들면 다음과 같습니다.

```
tcp_local single_sys smtp mx lastresort mailhub.siroe.com
TCP-DAEMON
```

12.4.3.8 받는 메일을 위한 대체 채널(전환 채널)

키워드: switchchannel, allowswitchchannel, noswitchchannel. 349 페이지 "12.4.4 SMTP 인증, SASL 및 TLS"에 있는 saslswitchchannel, 351 페이지 "12.4.8 전송 계층 보안"에 있는 tlsswitchchannel 및 347 페이지 "12.4.3.9 사용자 또는 도메인 설정을 기준으로 소스 채널 전환"에 있는 userswitchchannel을 참조하십시오.

switchchannel, allowswitchchannel, noswitchchannel 키워드는 받는 메일을 위한 대체 채널 선택을 제어합니다.

MTA는 원격 시스템으로부터 받는 연결을 수락할 때 연결에 사용할 채널을 선택해야 합니다. 일반적으로 이 결정은 사용된 전송을 기반으로 합니다. 예를 들어, 받는 SMTP over TCP/IP 연결은 tcp_local 채널에 자동으로 연결됩니다.

서로 다른 특성을 갖는 여러 보내는 채널을 사용하여 동일한 전송을 통해 서로 다른 여러 시스템을 처리할 경우에는 이 규칙이 적용되지 않습니다. 이러한 경우 받는 연결은 보내는 연결과 동일한 채널에 연결되지 않기 때문에 해당 채널 특성이 원격 시스템과 연결되지 않습니다.

switchchannel 키워드를 사용하면 이 문제를 해결할 수 있습니다. 서버가 사용하는 초기 채널에서 switchchannel을 지정하면 연결(원본) 호스트의 IP 주소와 채널 테이블을 비교하여 일치하는 경우 원본 채널이 적절하게 변경됩니다. IP 주소가 일치하지 않거나 원래의 기본 받는 채널에 대한 일치가 발견되는 경우 MTA는 DNS 역방향 조회를 통해 찾은 호스트 이름을 사용하여 선택적으로 일치를 시도할 수 있습니다. 소스 채널이 switchchannel 또는 allowswitchchannel(기본값) 표시가 있는 채널로 변경될 수 있습니다. noswitchchannel 키워드는 채널 전환이 수행되지 않음을 지정합니다.

서버가 연결하는 채널이 아닌 다른 채널에 지정한 switchchannel 키워드는 기본적으로 적용되지 않습니다. 현재는 switchchannel 키워드가 SMTP 채널에만 적용되며 switchchannel을 적용할 수 있는 다른 채널이 없습니다.

12.4.3.9 사용자 또는 도메인 설정을 기준으로 소스 채널 전환

키워드: userswitchchannel. 347 페이지 "12.4.3.8 받는 메일을 위한 대체 채널(전환 채널)"에 있는 switchchannel을 참조하십시오.

이제 사용자 또는 도메인 설정을 기준으로 소스 채널을 전환할 수 있습니다. 여기에는 다음과 같은 세 가지 새로운 설정이 사용됩니다.

1. 새로운 userswitchchannel 채널 키워드. 사용자 채널을 전환하려면 초기 소스 채널에 이 키워드가 있어야 합니다.
2. 새로운 MTA 옵션 LDAP_DOMAIN_ATTR_SOURCE_CHANNEL은 전환할 채널의 이름이 포함된 도메인 수준 속성의 이름을 지정합니다.
3. 새로운 MTA 옵션 LDAP_SOURCE_CHANNEL은 전환할 채널의 이름이 포함된 사용자 수준 속성의 이름을 지정합니다.

또한, 전환되는 채널을 전환이 허용되도록 설정해야 합니다. 즉, 해당 채널을 noswitchchannel 키워드로 표시할 수 없습니다. MAIL FROM 주소를 다시 작성하여 반환된 정보를 기준으로 전환이 적용됩니다. MAIL FROM 주소는 위조하기 쉬우므로 이 기능을 사용할 때에는 특히 주의해서 사용해야 합니다.

12.4.3.10 대상 호스트 선택

키워드: daemon, single, single_sys

daemon 키워드의 해석과 사용은 적용되는 채널의 유형에 따라 다릅니다.

daemon 키워드는 SMTP 채널에서 대상 호스트 선택을 제어하는 데 사용됩니다.

일반적으로 채널은 처리할 메시지의 봉투 주소에 나열되는 모든 호스트에 연결됩니다. daemon 키워드는 봉투 주소에 관계 없이 특정 원격 시스템(일반적으로 방화벽 또는 메일 허브 시스템)에 대신 연결하도록 채널에 지시하는 데 사용됩니다. 실제 원격 시스템 이름은 다음 예에 표시된 것처럼 daemon 키워드 바로 뒤에 표시되어야 합니다.

```
tcp_firewall smtp mx daemon firewall.acme.com
TCP-DAEMON
```

daemon 키워드 뒤의 인수가 정규화된 도메인 이름이 아닌 경우 해당 인수가 무시되고 채널이 해당 채널의 공식 호스트에 연결됩니다. 공식 호스트는 채널과 연관된 정규화된 호스트 이름이며 세 개 행 채널 블록의 두 번째 행에서 지정할 수 있습니다.

```
tcp_firewall smtp mx daemon router
firewall.acme.com
TCP-DAEMON
```

또한 공식 호스트를 두 개 행 채널 블록에서 TCP-DAEMON 뒤에 지정하여 아웃바운드 연결 자체를 특정 호스트로 식별하게 할 수 있습니다.

```
tcp_firewall smtp mx daemon router
TCP-DAEMON firewall.acme.com
```

방화벽 또는 게이트웨이 시스템 이름을 공식 호스트 이름으로 지정할 경우 다음 예에 표시된 것처럼 daemon 키워드에 지정된 인수가 일반적으로 라우터로 지정됩니다.

```
tcp_firewall smtp mx daemon router
firewall.acme.com
TCP-DAEMON
```

기타 관련 키워드로는 single 및 single_sys가 있습니다. single 키워드는 채널의 각 대상 주소에 대해 별도의 메시지 복사본을 만들도록 지정합니다. single_sys 키워드는 사용된 각 대상 시스템에 대해 단일 메시지 복사본을 만듭니다. 사용된 키워드에 관계 없이 메시지가 대기열에 있는 각 채널에 대해 해당 메시지 복사본을 하나 이상 만듭니다.

12.4.4 SMTP 인증, SASL 및 TLS

키워드: maysaslserver , mustsaslserver , nosasl, nosaslserver, saslswitchchannel, nosaslswitchchannel)

Messaging Server가 SASL(Simple Authentication and Security Layer)을 사용한 SMTP 서버 인증을 지원하는지 여부를 제어할 수 있습니다. SASL은 RFC 2222에 정의되어 있습니다. SASL, SMTP 인증 및 보안에 대한 자세한 내용은 23 장을 참조하십시오.

maysaslserver, mustsaslserver, nosasl, nosaslserver, switchchannel 및 saslswitchchannel 채널 키워드는 TCP/IP 채널과 같은 SMTP 채널이 SMTP 프로토콜에서 SASL(SMTP AUTH)을 사용하도록 구성하는 데 사용됩니다.

nosasl이 기본값이며 SASL 인증이 허용 또는 시도되지 않음을 의미합니다. 또한, SASL 인증이 허용되지 않음을 의미하는 nosaslserver를 포함합니다. maysaslserver를 지정하면 SMTP 서버에서 클라이언트가 SASL 인증 사용을 시도하도록 허용합니다. mustsaslserver를 지정하면 SMTP 서버에서 클라이언트가 SASL 인증을 사용하려고 시도하지만 SMTP 서버는 원격 클라이언트가 성공적으로 인증되지 않는 경우 메시지를 수락하지 않습니다.

saslswitchchannel을 사용하면 클라이언트가 SASL을 성공적으로 사용할 경우에 받는 연결이 지정한 채널로 전환되게 합니다. 전환할 채널을 지정하는 필수 값을 사용합니다.

12.4.5 헤더의 SMTP AUTH에서 인증된 주소 사용

키워드: authrewrite

authrewrite 채널 키워드 및 관련 AUTH_REWRITE 매핑 테이블을 사용하면 인증 작업에서 얻은 주소 지정 정보를 사용하여 헤더 및 봉투 주소를 수정할 수 있습니다. 특히 인증된 전자 메일 주소를 제공하도록 SASL 인증을 구성할 수 있습니다. 일반적으로 SMTP AUTH 정보가 사용되지만 이 정보는 FROM_ACCESS 매핑을 통해 대체될 수 있습니다. authrewrite 키워드는 표 12–23에 따라 필수 비트 값을 가집니다.

표 12–23 authrewrite 비트 값

비트	값	설명
0	1	아무것도 변경하지 않습니다(기본값).
1	2	인증 작업에서 제공된 주소를 포함하는 Sender: 또는 Resent-sender: 헤더 필드를 추가합니다. 다른 resent- 필드가 있는 경우 Resent- 변형이 사용됩니다.
2	4	AUTH 메일 발송자를 포함하는 Sender: 헤더 필드를 추가합니다.

표 12–23 authrewrite 비트 값 (계속)

비트	값	설명
3	8	다음과 같은 형태의 AUTH_REWRITE라는 매핑 테이블에서 검사를 생성합니다. *mail-from*|*sender*|*from*|*auth-sender* 여기서 *mail-from*은 봉투의 From: 주소이고 *sender*는 Sender: 또는 Resent-sender: 헤더 필드의 주소이며 *from*은 From: 또는 Resent-From: 헤더 필드의 주소이고 *auth-sender*는 인증 작업에서 제공된 주소입니다. 결과는 AUTH_REWRITE 매핑을 통해 실행됩니다. 매핑은 세로 막대(|)로 구분된 항목 목록을 반환합니다. 이러한 항목은 다음 플래그를 설정하여 순서대로 소비됩니다. $J $K 메시지 봉투의 From: 주소를 바꿉니다. $Y $T 적절한 Sender: 또는 Resent-sender: 헤더 필드를 추가합니다. $N 메시지를 거부합니다. 결과를 매핑하면 오류 메시지의 텍스트가 제공됩니다. 텍스트가 제공되지 않으면 invalid originator address used라는 오류 메시지가 표시됩니다. $Z 적절한 From: 또는 Resent-from: 헤더 필드를 추가합니다. (일반적으로 From: 필드를 대체하는 것은 아주 나쁜 방법입니다.) 헤더에 다른 Resent- 필드가 있는 경우 Resent- 변형이 사용됩니다.
4	16	인증에서 인증된 주소를 제공하지 않은 경우라도 AUTH_REWRITE 매핑을 적용합니다. 비트가 지워지면 인증된 주소를 사용할 수 있을 경우 매핑만 적용됩니다.
5	32	AUTH_REWRITE 매핑 검사의 시작 부분에 소스 채널을 포함합니다. 이 부분과 남은 정보는 |로 구분됩니다. 비트가 지워지면 채널이 포함되지 않습니다.

주의 – $Z 플래그는 봉투와 헤더 주소를 수정하기 위해 합법적으로 사용하는 경우는 적으므로 높은 수준으로 제한되어야 합니다.

12.4.6 SMTP 청크 지원

키워드: chunkingclient, chunkingserver, nochunkingclient, nochunkingserver

SMTP 청크(RFC 3030)에 대한 지원이 SMTP 클라이언트와 서버 모두에 추가되었습니다. 이 지원은 기본적으로 활성화됩니다. 청크 허용 여부를 제어할 때 4개의 새 채널 키워드를 사용할 수 있습니다. 이러한 키워드는 다음과 같습니다.

- chunkingclient - 클라이언트 청크 지원을 활성화합니다(기본값).
- chunkingserver - 서버 청크 지원을 활성화합니다(기본값).
- nochunkingclient - 클라이언트 청크 지원을 비활성화합니다.
- nochunkingserver - 서버 청크 지원을 비활성화합니다.

지정된 메시지 전송에 청크가 사용되었는지 여부를 나타내도록 로그 파일 작업 필드가 확장되었습니다. 특히 청크가 사용될 경우 C가 추가됩니다. 청크가 작동하려면 ESMTP를 사용해야 하므로 일반적으로 EEC 또는 DEC와 같은 필드 값이 표시됩니다.

12.4.7 Microsoft Exchange 게이트웨이 채널 지정

키워드: msexchange, nomsexchange

TCP/IP 채널에서 msexchange 채널 키워드를 사용하여 이 채널이 Microsoft Exchange 게이트웨이 및 클라이언트와 통신하는 채널임을 MTA에 알려줄 수 있습니다. 이 채널 키워드를 maysaslserver 또는 mustsaslserver 키워드를 통해 SASL을 사용하는 받는 TCP/IP 채널에 넣을 경우 MTA의 SMTP 서버가 "잘못된" 형식(새로 수정된 AUTH 사양 대신 올바른 ESMTP 사용과 호환되지 않는 원본의 ESMTP AUTH 사양을 기반으로 함)을 사용하여 AUTH를 광고하게 합니다. 예를 들어, 일부 Microsoft Exchange 클라이언트는 올바른 AUTH 형식을 인식하지 않고 잘못된 AUTH 형식만 인식합니다.

또한, msexchange 채널 키워드는 끊어진 TLS 명령을 광고 및 인식하게 합니다.

기본값은 nomsexchange입니다.

12.4.8 전송 계층 보안

키워드: maytls, maytlsclient, maytlsserver, musttls, musttlsclient, musttlsserver, notls, notlsclient, notlsserver, tlsswitchchannel

maytls, maytlsclient, maytlsserver, musttls, musttlsclient , musttlsserver, notls, notlsclient, notlsserver 및 tlsswitchchannel 채널 키워드는 TCP/IP 채널과 같은 SMTP 기반 채널에 의한 SMTP 프로토콜에서 TLS 사용을 구성하는 데 사용됩니다.

기본값은 notls이고 TLS가 허용 또는 시도되지 않음을 의미합니다. 보내는 연결에서 MTA SMTP 클라이언트가 TLS 사용을 시도하지 않음(보내는 연결 중에 STARTTLS 명령 실행 안 됨)을 의미하는 notlsclient 키워드와 받는 연결에서 MTA SMTP 서버의 TLS 사용이 허용되지 않음(SMTP 서버에서 STARTTLS 확장을 광고하지 않고 명령 자체도 허용 안 됨)을 의미하는 notlsserver 키워드를 포함합니다.

maytls를 지정하면 MTA가 받는 연결에 TLS를 제공하고 보내는 연결에서 TLS를 시도합니다. TLS를 지원하는 SMTP 서버에 메시지를 보낼 때 MTA SMTP 클라이언트가 TLS 사용을 시도함을 의미하는 maytlsclient 키워드와 MTA SMTP 서버가 STARTTLS 확장 지원을 광고하고 메시지를 받을 때 TLS 사용을 허용함을 의미하는 maytlsserver 키워드를 포함합니다.

TLS로 작업할 경우 다음 조건을 충족시켜야 합니다.

- mailsrv 계정에서 파일에 액세스할 수 있도록 인증서의 보호/소유권을 설정해야 합니다.

- mailsrv 계정에서 인증서가 저장되는 디렉토리 내의 파일에 액세스할 수 있도록 해당 디렉토리에서 보호/소유권을 설정해야 합니다.

musttls를 지정하면 MTA가 보내는 연결과 받는 연결 모두에서 TLS 사용을 주장하게 되므로 TLS 사용 협상에 실패할 경우 원격 시스템과 전자 메일을 교환할 수 없습니다. MTA SMTP 클라이언트가 메시지를 보낼 때 TLS 사용을 주장하고 TLS 사용 협상(MTA가 STARTTLS 명령을 실행하고 해당 명령이 성공해야 함)에 실패한 SMTP 서버에 메시지를 보내지 않음을 의미하는 musttlsclient를 포함합니다. 또한, MTA SMTP 서버가 STARTTLS 확장 지원을 광고하고 메시지를 받을 때 TLS 사용을 강제하고 TLS 사용 협상에 실패한 클라이언트에서 보낸 메시지를 받지 않음을 의미하는 musttlsserver를 포함합니다.

tlsswitchchannel 키워드를 사용하면 클라이언트가 SASL 협상에 성공할 경우 받는 연결이 지정한 채널로 전환됩니다. 전환할 채널을 지정하는 필수 값을 사용합니다.

12.5 메시지 처리 및 전달 구성

서버가 특정 기준에 따라 메시지 전달을 시도하도록 구성할 수 있습니다. 또한, 서비스 작업 처리 제한, 새 SMTP 채널 스레드 생성 시기 등과 같은 작업 처리 매개 변수를 지정할 수 있습니다. 이 절은 다음 내용으로 구성되어 있습니다.

메시지 처리 및 전달에 대한 개념 정보는 표 12–24를 참조하십시오.

352 페이지 “12.5 메시지 처리 및 전달 구성”은 이 절에서 설명하는 키워드를 요약해서 보여 줍니다.

표 12–24 메시지 처리 및 전달 키워드

키워드	정의
즉시 전달	**메시지 즉시 전달에 대한 사양을 정의합니다.**
immnonurgent	높음, 중간 및 낮음 우선 순위 메시지를 제출하면 바로 전달을 시작합니다.
채널 방향	**채널을 사용하는 프로그램 유형을 지정합니다.**

표 12–24 메시지 처리 및 전달 키워드 *(계속)*

키워드	정의
bidirectional	마스터 및 슬레이브 프로그램에서 사용되는 채널입니다.
master	마스터 프로그램에서 사용되는 채널(`master`)입니다.
slave	슬레이브 프로그램에서 사용되는 채널(`slave`)입니다.
지연 전달	**지연 작업 전달에 대한 사양을 정의합니다.**
`backoff`	지연 메시지의 재전달 시도 간격을 지정합니다. `normalbackoff`, `nonurgentbackoff`, `urgentbackoff`에 의해 대체될 수 있습니다.
`deferred`	`Deferred-delivery:` 헤더 행을 수락하고 인식합니다.
`nodeferred`	기본값입니다. `Deferred-delivery:` 헤더 행을 수락하지 않음을 지정합니다.
`nonurgentbackoff`	낮음 우선 순위 메시지의 재전달 시도 간격입니다.
`normalbackoff`	중간 우선 순위 메시지의 재전달 시도 간격입니다.
`urgentbackoff`	높음 우선 순위 메시지의 재전달 시도 간격입니다.
크기 기반 메시지 우선 순위	**메시지 크기를 기반으로 메시지의 우선 순위를 정의합니다.**
`nonurgentblocklimit`	이 크기 이상인 메시지의 우선 순위를 낮음(두 번째 우선 순위 클래스) 이하로 지정합니다. 즉, 해당 메시지는 항상 다음 정기 작업이 처리되는 동안 대기한 후 처리됩니다.
`normalblocklimit`	이 크기 이상인 메시지의 우선 순위를 낮음으로 지정합니다.
`urgentblocklimit`	이 크기 이상인 메시지의 우선 순위를 중간으로 지정합니다.
채널 실행 작업의 처리 풀	**작업 우선 순위와 지연 수준이 서로 다른 메시지 처리를 위한 풀을 지정합니다.**
`pool`	채널이 실행되는 풀을 지정합니다.
`after`	채널이 실행되기 이전의 시간 지연을 지정합니다.
서비스 작업 제한	**서비스 작업 수와 작업당 처리할 최대 메시지 파일 수를 지정합니다.**
`maxjobs`	채널에 대해 동시에 실행될 수 있는 최대 작업 수를 지정합니다.
`filesperjob`	단일 작업에서 처리할 대기열 항목 수를 지정합니다.
SMTP 채널 스레드	
`threaddepth`	다중 스레드 SMTP 클라이언트를 사용하여 새 스레드를 트리거하는 메시지 수입니다.
다중 주소 확장	**수신자가 여러 명인 메시지의 처리를 정의합니다.**
`expandlimit`	주소 수가 이 제한을 초과할 경우 받는 메시지를 "오프라인"으로 처리합니다.
`expandchannel`	expandlimit 적용으로 인해 지연된 확장을 수행할 채널을 지정합니다.
`holdlimit`	주소 수가 이 제한을 초과할 경우 받는 메시지를 보관합니다.

표 12–24 메시지 처리 및 전달 키워드 *(계속)*

키워드	정의
트랜잭션 제한	**연결 트랜잭션 제한을 지정합니다.**
transactionlimit	연결당 허용되는 메시지 수를 제한합니다.
전달할 수 없는 메시지 알림	**전달할 수 없는 메시지 알림을 보내는 시기를 지정합니다.**
notices	알림을 보낸 후 메시지가 반환되기 이전에 경과할 수 있는 시간을 지정합니다.
nonurgentnotices	우선 순위가 낮은 메시지에 대해 알림을 보내고 메시지를 반환하기 전에 경과할 수 있는 시간을 지정합니다.
normalnotices	우선 순위가 중간인 메시지에 대해 알림을 보내고 메시지를 반환하기 전에 경과할 수 있는 시간을 지정합니다.
urgentnotices	우선 순위가 높은 메시지에 대해 알림을 보내고 메시지를 반환하기 전에 경과할 수 있는 시간을 지정합니다.

12.5.1 채널 방향 설정

키워드: master, slave, bidirectional

세 키워드를 사용하여 채널이 마스터 프로그램(master), 슬레이브 프로그램(slave) 또는 두 프로그램 모두(bidirectional) 중 어느 것에서 사용되는지 지정합니다. 이 키워드를 지정하지 않을 경우 기본값은 bidirectional입니다. 이러한 키워드는 메시지를 채널의 대기열에 넣을 때 MTA가 전달 작업을 시작하는지 여부를 지정합니다.

이러한 키워드의 사용은 해당 채널 프로그램의 기본 특성을 반영합니다. MTA가 지원하는 다양한 채널 설명은 이러한 키워드를 사용해야 하는 시기와 위치를 나타냅니다.

12.5.2 지연 전달 날짜 구현

키워드: deferred, nodeferred, immnonurgent

deferred 채널 키워드는 Deferred-delivery: 헤더 행을 인식하고 준수합니다. deferred 전달 날짜가 미래 날짜인 메시지는 해당 날짜가 만료되어 반환되거나 지연 전달 날짜가 될 때까지 채널 대기열에 보관됩니다. Deferred-delivery: 헤더 행의 형식 및 작업에 대한 자세한 내용은 RFC 1327을 참조하십시오.

기본값은 nodeferred 키워드입니다. 지연 메시지 처리에 대한 지원이 RFC 1327에 규정되어 있지만 지연 메시지 처리를 실제로 구현하면 메일 시스템을 디스크 할당량의 확장으로 효과적으로 사용할 수 있습니다.

키워드 immnonurgent는 높음, 중간 및 낮음 우선 순위 메시지를 제출하면 바로 전달을 시작합니다.

12.5.3 전달에 실패한 메시지에 대한 재시도 간격 지정

키워드: backoff , nonurgentbackoff, normalbackoff, urgentbackoff, notices

기본적으로 전달에 실패한 메시지에 대한 전달 재시도 간격은 메시지의 우선 순위에 따라 다릅니다. 전달 시도 사이의 기본 간격(분)은 아래에 나와 있습니다. 우선 순위 뒤의 첫 번째 숫자는 최초 전달 실패 후 첫 번째 전달 재시도가 시도될 때까지의 시간(분)을 나타냅니다.

```
urgent: 30, 60, 60, 120, 120, 120, 240
normal: 60, 120, 120, 240, 240, 240, 480
nonurgent: 120, 240, 240, 480, 480, 480, 960
```

우선 순위가 높은 메시지의 경우 최초 전달 실패 후 30분 뒤, 첫 번째 전달 재시도 후 60분 뒤, 두 번째 재시도 후 60분 뒤, 세 번째 재시도 후 120분 뒤 등의 간격으로 재시도가 시도됩니다. 지정된 마지막 시도 이후의 재시도는 동일한 간격으로 반복됩니다. 따라서, 우선 순위가 높은 메시지는 이러한 전달 재시도가 240분 간격으로 수행됩니다.

전달 시도는 notices, nonurgentnotices, normalnotices 또는 urgentnotices 키워드에 지정된 기간 동안 계속됩니다. 전달이 성공적으로 이루어질 수 없으면 **전달 실패 알림**이 생성되고 메시지는 보낸 사람에게 반환됩니다. notices 키워드에 대한 자세한 내용은 253 페이지 “10.10.4.3 알림 메일 전달 간격 설정”을 참조하십시오.

backoff 키워드를 사용하면 다양한 우선 순위를 갖는 메시지에 대한 전달 재시도 간격 설정을 사용자 정의할 수 있습니다. nonurgentbackoff는 우선 순위가 낮은 메시지에 대한 간격을 지정하고, normalbackoff는 우선 순위가 보통인 메시지에 대한 간격을 지정하고, urgentbackoff는 우선 순위가 높은 메시지에 대한 간격을 지정합니다. 이러한 키워드를 지정하지 않으면 우선 순위에 관계 없이 backoff에 의해 모든 메시지에 대한 간격이 지정됩니다.

예를 들면 다음과 같습니다.

```
urgentbackoff "pt30m" "pt1h" "pt2h" "pt3h" "pt4h" "pt5h" "pt8h" "pt16h"
```

여기서 우선 순위가 높은 메시지의 경우 최초 전달 실패 후 30분 뒤, 첫 번째 전달 시도 후 1시간 뒤(최초 실패 후 1시간 30분 뒤), 두 번째 전달 시도 후 2시간 뒤, 세 번째 시도 후 3시간 뒤, 네 번째 시도 후 4시간 뒤, 다섯 번째 시도 후 5시간 뒤, 여섯 번째 시도 후 8시간 뒤, 일곱 번째 전달 시도 후 16시간 뒤에 각각 전달 재시도가 수행됩니다. 후속 시도는 notices 키워드에 지정된 기간까지 16시간마다 수행됩니다. 전달이 성공적으로 이루어질 수 없으면 전달 실패 알림이 생성되고 메시지는 보낸 사람에게 반환됩니다. 간격 구문은 ISO 8601P에 나와 있으며 Sun Java System Messaging Server Administration Reference에도 설명되어 있습니다.

다음 예에서,

```
normalbackoff "pt30m" "pt1h" "pt8h" "p1d" "p2d” "p1w"
```

우선 순위가 보통인 메시지의 경우 최초 전달 실패 후 30분 뒤, 첫 번째 전달 시도 후 1시간 뒤, 두 번째 시도 후 8시간 뒤, 세 번째 시도 후 1일 뒤, 네 번째 시도 후 2일 뒤, 다섯 번째 시도 후 1주일 뒤에 전달 재시도가 각각 수행되고 이후에는 `notices` 키워드에 지정된 기간까지 1주일마다 전달 재시도가 반복됩니다. 전달이 성공적으로 이루어질 수 없으면 전달 실패 알림이 생성되고 메시지는 보낸 사람에게 반환됩니다.

마지막 예에서,

```
backoff "pt30m" "pt120m" "pt16h" "pt36h" "p3d"
```

모든 실패한 메시지의 경우 `nonurgentbackoff`, `normalbackoff` 또는 `urgentbackoff` 키워드에 의해 대체되지 않는 한, 메시지 우선 순위에 관계 없이 최초 전달 실패 후 30분 뒤, 첫 번째 재시도 후 2시간 뒤, 두 번째 재시도 후 16시간 뒤, 세 번째 재시도 후 36시간 뒤, 네 번째 재시도 후 3일 뒤에 전달 재시도가 각각 수행되고 이후에는 `notices` 키워드에 지정된 기간까지 3일마다 전달 재시도가 수행됩니다. 전달이 성공적으로 이루어질 수 없으면 전달 실패 알림이 생성되고 메시지는 보낸 사람에게 반환됩니다.

12.5.4 채널 실행 작업의 처리 풀

키워드: `pool`

채널을 동일한 풀 내에서 실행하여 여러 채널이 자원을 공유하도록 구성할 수 있습니다. 다른 채널을 특정 채널에 전용인 풀에서 실행하도록 구성할 수 있습니다. 각 풀 내에서 메시지는 메시지 우선 순위에 따라 서로 다른 처리 대기열에 자동으로 정렬됩니다. 풀 내에서 우선 순위가 높은 메시지가 우선 순위가 낮은 메시지보다 먼저 처리됩니다. 358 페이지 “12.5.7 크기 기반 메시지 우선 순위”를 참조하십시오.

`pool` 키워드를 사용하면 작업이 만들어지는 풀을 채널 단위로 선택할 수 있습니다. `pool` 키워드는 현재 채널에 대한 전달 작업을 풀링해야 하는 풀 이름 앞에 와야 합니다. 풀 이름은 11자를 초과할 수 없습니다.

작업 제어기 개념 및 구성에 대한 자세한 내용은 231 페이지 “10.4.8 작업 제어기 파일”, 231 페이지 “10.4.8 작업 제어기 파일” 및 356 페이지 “12.5.5 서비스 작업 제한”을 참조하십시오.

12.5.5 서비스 작업 제한

키워드: `maxjobs`, `filesperjob`

메시지가 채널의 대기열에 배치될 때마다 작업 제어기는 해당 메시지를 전달하기 위해 실행 중인 작업이 있는지 확인합니다. 여기에는 새 작업 프로세스를 시작하거나, 스레드를 추가하거나, 단순히 작업이 이미 실행 중인지 확인하는 것이 포함됩니다. 단일 서비스 작업으로 모든 메시지 전달을 확인하지 못할 수도 있습니다. 작업 제어기 개념 및 구성에 대한 자세한 내용은 231 페이지 “10.4.8 작업 제어기 파일”, 356 페이지 “12.5.4 채널 실행 작업의 처리 풀” 및 179 페이지 “8.7 작업 제어기”를 참조하십시오.

특정 설치의 경우 메시지 전달을 위해 시작할 프로세스 및 스레드에 대한 합리적인 최대 수가 있습니다. 이 최대 수는 프로세서 수, 디스크 속도, 연결 특징 등의 요인에 따라 다릅니다. MTA 구성에서는 다음을 제어할 수 있습니다.

- 지정된 채널에 대해 실행할 최대 프로세스 수(maxjobs 채널 키워드)
- 채널 집합에 대해 시작할 최대 프로세스 수(작업 제어기 구성 파일의 관련 풀 섹션에 있는 JOB_LIMIT 매개 변수)
- 새 스레드 또는 프로세스를 시작하기 전에 받은 대기열에 포함된 메시지 수(threaddepth 채널 키워드)
- 일부 채널의 경우 지정된 전달 프로그램 내에서 실행되는 최대 스레드 수(채널 옵션 파일에 있는 max_client_threads 매개 변수)

지정된 채널에 대해 실행을 시작하는 최대 프로세스 수는 채널에 설정된 maxjobs의 최대값이며 채널이 실행되는 풀에 대해 설정된 JOB_LIMIT입니다.

메시지를 처리해야 한다고 가정합니다. 일반적으로 작업 제어기는 새 프로세스를 다음과 같이 시작합니다.

- 채널에 대해 실행 중인 프로세스가 없고 풀 작업 제한에 도달하지 않은 경우 작업 제어기는 새 프로세스를 시작합니다.
- 채널 프로그램이 단일 스레드이거나 스레드 제한에 도달하고 백로그가 threaddepth에 지정된 스레드 수의 배 이상으로 증가하고 채널 및 풀 작업 제한에 도달하지 않은 경우 작업 제어기는 새 프로세스를 시작합니다.
- 채널 프로그램이 다중 스레드이고 스레드 제한에 도달하지 않았으며, 메시지 백로그가 threaddepth의 2배 이상으로 증가한 경우 새 스레드가 시작됩니다.

특히, SMTP 채널의 경우 서로 다른 호스트에 대한 대기열에 메시지가 포함될 경우 새 스레드 또는 프로세스가 시작됩니다. 따라서, SMTP 채널의 경우 작업 제어기는 새 프로세스를 다음과 같이 시작합니다. 메시지를 처리해야 한다고 가정합니다.

- SMTP 채널에 대해 실행 중인 프로세스가 없고 풀 제한에 도달하지 않은 경우 작업 제어기는 새 프로세스를 시작합니다.
- 스레드 제한(MAX_CLIENT_THREADS)에 도달하고, 아직 사용되고 있지 않은 호스트의 대기열에 메시지가 포함되고, 채널(maxjobs) 및 풀 작업 제한(JOB_LIMIT)에 도달하지 않은 경우 새 프로세스가 시작됩니다.
- 스레드 제한에 도달하지 않고 아직 사용되고 있지 않은 호스트의 대기열에 메시지가 포함될 경우 새 스레드가 시작됩니다.
- 스레드 제한에 도달하지 않았으며 메시지가 대기열에 포함되어 해당 호스트에 대한 메시지 백로그가 threaddepth의 2배 이상으로 증가할 경우 새 스레드가 시작됩니다.

359 페이지 “12.5.8 SMTP 채널 스레드”를 참조하십시오.

filesperjob 키워드를 사용하여 MTA에서 추가 서비스 작업을 만들 수 있습니다. 이 키워드는 여러 서비스 작업을 만들어 처리하기 전에 연결된 채널로 보내야 하는 대기열 항목(파일) 수를 지정하는 단일의 양의 정수 매개 변수를 가집니다. 0보다 작거나 같은

값을 지정하면 하나의 서비스 작업만 대기열에 포함하라는 요청으로 해석됩니다. 키워드를 지정하지 않으면 0의 값을 지정한 것과 같습니다. 이 키워드의 효과는 최대화됩니다. 즉, 계산된 높은 숫자가 실제로 만들어지는 서비스 작업 수가 됩니다.

filesperjob 키워드는 실제 대기열 항목 또는 파일 수를 지정된 값으로 나눕니다. 지정된 메시지의 대기열 항목 수는 single 및 single_sys 키워드의 사용, 메일링 목록의 헤더 수정 작업 사양 등 많은 요소에 의해 제어됩니다.

maxjobs 키워드는 동시에 실행될 수 있는 총 서비스 작업 수에 대한 최대값을 지정합니다. 이 키워드는 정수 값이 뒤에 와야 합니다. 계산된 서비스 작업 수가 이 값보다 더 큰 경우 maxjobs 작업만 실제로 만들어집니다. maxjobs를 지정하지 않은 경우 이 값의 기본값은 100입니다. 일반적으로 maxjobs는 채널이 사용되는 서비스 풀에 관계 없이 동시에 실행될 수 있는 총 작업 수보다 작거나 같은 값으로 설정됩니다.

12.5.6 연결 트랜잭션 제한 설정

키워드: transactionlimit

transactionlimit는 연결당 허용되는 메시지 수를 제한합니다. 이 키워드를 사용하여 다음과 같은 방법으로 공격자를 차단할 수 있습니다.

공격자가 SMTP를 통해 연결한 다음 많은 RCPT TO 명령을 보내 합법적인 전자 메일 주소를 추측하려고 시도할 수 있습니다. 트랜잭션에 허용되는 유효하지 않은 RCPT TO 수를 제한하여 그런 공격을 차단할 수 있습니다. 공격자는 SMTP 세션에 허용되는 트랜잭션 수를 제한할 수 있는 transactionlimit가 있는 여러 트랜잭션을 사용하여 응답할 수 있습니다. 공격자가 여러 세션을 사용할 수 있지만 과도한 비용이 듭니다. 연결 억제를 사용하여 대부분의 경우에 실제로 많은 비용이 들게 하는 다양한 방법으로 세션 수를 제한할 수 있습니다.

이 비용은 우리 쪽에 부과되는 비용이지만, 일부 SMTP 클라이언트는 수신자 제한, 트랜잭션 제한 또는 두 가지 모두에 잘못된 반응을 나타내는 경우도 있습니다. 이러한 클라이언트에 대한 예외를 만들어야 합니다. 그러나, TCP 채널 옵션은 SMTP 서버에 무조건적으로 적용됩니다. 해결책은 채널 키워드와 switchchannel을 사용하여 문제가 있는 에이전트의 경로를 더 큰 제한이 있는 채널로 지정하는 것입니다.

12.5.7 크기 기반 메시지 우선 순위

키워드: urgentblocklimit, normalblocklimit, nonurgentblocklimit

urgentblocklimit, normalblocklimit 및 nonurgentblocklimit 키워드를 사용하여 MTA에 크기 기반 메시지의 우선 순위를 낮추도록 지시할 수 있습니다. 이러한 키워드는 메시지를 처리할 때 작업 제어기가 적용하는 우선 순위에 영향을 미칩니다.

12.5.8 SMTP 채널 스레드

키워드: threaddepth,

다중 스레드 SMTP 클라이언트는 각 스레드에 대한 대상별로 보내는 메시지를 정렬합니다. threaddepth 키워드를 사용하면 다중 스레드 SMTP 클라이언트에 한 스레드에서 지정된 수의 메시지만 처리하도록 지시하여 대상이 같은 메시지(일반적으로 한 스레드에서 모두 처리됨)일 경우에도 추가 스레드를 사용하게 할 수 있습니다. 이 키워드의 기본값은 10입니다.

채널의 백로그가 threaddepth의 배 이상으로 증가할 때마다 작업 제어기는 해당 채널의 대기열에 포함된 메시지 처리를 전담하는 처리량을 높이려고 시도합니다. 다중 스레드 채널의 경우 작업 제어기는 해당 채널에 대한 메시지를 처리하는 작업에서 새 스레드를 시작하게 합니다. 또는 모든 작업이 해당 채널에 대해 허용된 최대 수의 스레드(tcp_* 채널에 대한 옵션의 MAX_CLIENT_THREADS)를 갖는 경우 새 프로세스를 시작합니다. 단일 스레드 채널의 경우 새 프로세스를 시작합니다. 채널에 대한 작업 제한(maxjobs) 또는 풀에 대한 작업 제한(JOB_LIMIT)에 도달한 경우 작업 제어기는 새 작업을 시작하지 않습니다.

기본적으로 threaddepth는 적극적인 작업을 예약하는 방법을 제어합니다. 다음의 두 가지 다른 상황을 고려해 보겠습니다.

(1) 일반(아웃바운드) SMTP 채널

(2) 스마트 호스트에 전달하는 SMTP 채널

작업 제어기는 특정 채널을 대상으로 하는 메시지를 대상 호스트별로 정렬하고 이러한 대상 호스트의 백로그에 기초하여 메시지를 처리하기 위한 작업을 예약합니다.

첫 번째 경우는 많은 수의 대상 호스트가 있고 대상 호스트의 백로그가 대부분 작습니다. 실행되는 스레드의 수가 많으며 aol, yahoo, hotmail 등과 같이 대량의 트래픽이 있을 수 있는 대상 호스트를 제외하고는 모두 제대로 작동합니다. 스레드 깊이가 128인 경우 백로그가 128에 도달하면 yahoo에 전달되는 두 번째 스레드만 가져오게 됩니다. 이는 바람직하지 않습니다.

두 번째 경우는 대상 호스트가 하나만 존재하며 해당 호스트에 많은 수의 스레드를 전달하는 것이 바람직합니다. 어느 경우든 기본값 10은 너무 작을 수 있습니다.

threaddepth를 사용하면 채널이 연결하는 SMTP 서버에서 여러 동시 연결을 처리할 수 있을 때 데몬 라우터 TCP/IP 채널(단일의 특정 SMTP에 연결하는 TCP/IP 채널)에서 다중 스레딩을 수행하는 데 특히 유용합니다.

12.5.9 여러 주소 확장

키워드: expandlimit, expandchannel, holdlimit

대부분의 채널은 각 인바운드 메시지 전송에서 여러 수신자 주소 사양을 지원합니다. 단일 메시지에 많은 수신자 주소가 있는 사양에서는 메시지 전송 처리가 지연될 수 있습니다(온라인 지연). 너무 오래 지연될 경우 네트워크 시간 초과가 발생하여 메시지 제출 시도가 반복되거나 기타 문제가 발생할 수 있습니다.

MTA는 단일 메시지에 대해 지정된 수보다 더 많은 주소를 지정할 경우 지연(오프라인) 처리를 강제하는 특수 기능을 제공합니다. 메시지 처리 지연은 온라인 지연을 대폭 줄일 수 있습니다. 그러나 처리 오버헤드가 지연되지만 완전히 제거되지는 않습니다.

예를 들어, 일반 reprocessing 채널과 expandlimit 키워드를 함께 사용하면 이 특수 기능이 활성화됩니다. expandlimit 키워드는 지연 처리 이전에 채널에서 받은 메시지에 허용되는 주소 수를 지정하는 정수 인수를 가집니다. expandlimit 키워드를 지정하지 않은 경우 기본값은 무제한입니다. 값이 0이면 채널에서 수신하는 모든 주소에서 지연 처리를 수행합니다.

로컬 채널 또는 reprocessing 채널 자체에는 expandlimit 키워드를 지정하면 안 됩니다. 이러한 키워드를 지정하면 예상치 못한 결과가 발생할 수 있습니다.

지연 처리를 수행하는 데 실제로 사용되는 채널은 expandchannel 키워드를 사용하여 지정할 수 있습니다. expandchannel을 지정하지 않은 경우 reprocessing 채널이 기본적으로 사용되지만 다른 reprocessing 또는 processing 채널을 사용하는 것이 유용한 경우도 있습니다. expandchannel을 통해 지연 처리를 위한 채널을 지정하는 경우 해당 채널이 reprocessing 또는 processing 채널이어야 합니다. 다른 종류의 채널 사양은 예측할 수 없는 결과를 초래할 수 있습니다.

expandlimit 키워드를 적용하려면 reprocessing 채널이나 지연 처리를 수행하는 데 사용되는 모든 채널을 MTA 구성 파일에 추가해야 합니다. MTA 구성 유틸리티를 사용하여 구성을 작성한 경우 reprocessing 채널이 이미 있어야 합니다.

지나치게 큰 수신자 주소 목록은 UBE(Unsolicited Bulk Email) 특성을 가질 수 있습니다. holdlimit 키워드는 수신자가 지정된 수보다 많은 채널에서 수신하는 메시지에 .HELD 메시지 표시를 하고 reprocess 채널 또는 expandchannel 키워드를 통해 지정한 모든 채널의 대기열에 포함하도록 MTA에 지시합니다. 이러한 파일은 MTA 포스트마스터가 수동으로 처리할 때까지 reprocess 대기열에 처리되지 않은 상태로 유지됩니다.

12.5.10 서비스 변환 사용

키워드: service, noservice

service 키워드는 CHARSET-CONVERSION 항목에 관계 없이 서비스 변환을 무조건적으로 사용합니다. noservice 키워드를 설정하면 이 채널에 수신되는 메시지에 대해 CHARSET-CONVERSION을 통해 서비스 변환을 사용해야 합니다.

12.6 주소 처리 구성

이 절에서는 주소 처리를 수행하는 키워드에 대해 설명합니다. 이 장은 다음 내용으로 구성되어 있습니다.

12.6.1 주소 유형 및 규칙

키워드: 822, 733, uucp, header_822, header_733, header_uucp

이 키워드 그룹은 채널에서 지원하는 주소 유형을 제어합니다. 전송 계층(메시지 봉투)에 사용되는 주소와 메시지 헤더에 사용되는 주소를 구분합니다.

12.6.1.1 822(sourceroute)

소스 경로 봉투 주소입니다. 이 채널은 소스 경로를 포함하여 전체 RFC 822 형식의 봉투 주소 지정 규칙을 지원합니다. `sourceroute` 키워드를 822와 동의어로 사용할 수도 있습니다. 이 키워드는 다른 봉투 주소 유형 키워드를 지정하지 않은 경우의 기본값입니다.

12.6.1.2 733(percents)

백분율(%) 기호 봉투 주소입니다. 이 채널은 소스 경로를 제외한 전체 RFC 822 형식의 봉투 주소 지정을 지원합니다. 소스 경로는 백분율(%) 기호 규칙을 대신 사용하여 다시 작성해야 합니다. `percents` 키워드는 733의 동의어로 사용할 수 있습니다.

주 – SMTP 채널에서 733 주소 규칙을 사용하면 이러한 규칙이 SMTP 봉투의 전송 계층 주소에 적용됩니다. 그렇게 하면 RFC 821을 위반할 수 있습니다. 733 주소 규칙은 반드시 필요한 경우에만 사용하십시오.

12.6.1.3 uucp(bangstyle)

뱅 스타일 봉투 주소입니다. 이 채널은 RFC 976 뱅 스타일의 주소 규칙을 준수하는 주소를 봉투에 사용합니다. 이 채널의 예로는 UUCP 채널이 있습니다. `bangstyle` 키워드는 uucp의 동의어로 사용할 수 있습니다.

12.6.1.4 header_822

소스 경로 헤더 주소입니다. 이 채널은 소스 경로를 포함하여 전체 RFC 822 형식의 헤더 주소 지정 규칙을 지원합니다. 다른 헤더 주소 유형 키워드를 지정하지 않을 경우의 기본값입니다.

12.6.1.5 header_733

백분율(%) 기호 헤더 주소입니다. 이 채널은 소스 경로를 제외한 RFC 822 형식의 헤더 주소 지정을 지원합니다. 소스 경로는 백분율(%) 기호 규칙을 대신 사용하여 다시 작성해야 합니다.

주 – 메시지 헤더에 733 주소 규칙을 사용하면 RFC 822 및 RFC 976 위반입니다. 채널이 소스 경로 주소를 처리할 수 없는 시스템에 연결된다고 확신하는 경우에만 이 키워드를 사용하십시오.

12.6.1.6 header_uucp

UUCP 또는 뱅 스타일의 헤더 주소입니다. 이 키워드를 사용하면 RFC 976 위반이므로 이 키워드는 사용하지 않는 것이 좋습니다.

12.6.2 ! 및 %를 사용하는 주소 해석

키워드: `bangoverpercent`, `nobangoverpercent`, `percentonly`

주소는 항상 RFC 822 및 RFC 976에 따라 해석되지만 이러한 표준에 맞게 주소가 지정되지 않은 복합 주소를 처리할 경우 애매할 수 있습니다. 특히 `A!B%C` 형식의 주소는 다음과 같이 해석될 수 있습니다.

- `A`는 라우팅 호스트로 해석되고 `C`는 최종 대상 호스트로 해석됩니다.

또는

- `C`는 라우팅 호스트로 해석되고 `A`는 최종 대상 호스트로 해석됩니다.

RFC 976은 메일 프로그램이 두 번째 규칙을 사용하여 주소를 해석할 수 있음을 의미하지만, 그런 해석이 반드시 필요하다는 의미는 아닙니다. 첫 번째 해석 방법이 더 좋은 경우도 있습니다.

bangoverpercent 키워드는 첫 번째 A!(B%C) 해석을 적용합니다. nobangoverpercent 키워드는 두 번째(A!B)%C 해석을 적용합니다. 기본값은 nobangoverpercent입니다.

주 – 이 키워드는 A!B@C 형식의 주소 처리에 영향을 미치지 않습니다. 이러한 주소는 항상 (A!B)@C로 처리됩니다. 이러한 처리는 RFC 822와 RFC 976 모두에서 필수입니다.

---

percentonly 키워드는 뱅 경로를 무시합니다. 이 키워드를 설정하면 %이 라우팅으로 해석됩니다.

12.6.3 주소에 라우팅 정보 추가

키워드: exproute, noexproute, improute, noimproute

MTA가 처리하는 주소 지정 모델에서는 모든 시스템이 다른 나머지 시스템의 주소와 다른 시스템에 연결하는 방법을 알고 있는 것으로 가정합니다. 불행하게도 이러한 가정이 모든 경우에 적용되는 것은 아닙니다. 채널이 다른 나머지 시스템에서는 알지 못하는 하나 이상의 시스템(예: 개인 TCP/IP 네트워크의 내부 시스템)에 연결하는 경우가 있습니다. 이 채널의 시스템 주소는 사이트 외부의 원격 시스템에는 적합하지 않을 수 있습니다. 그러한 주소에 회신할 수 있으려면 로컬 시스템을 통해 메시지 경로를 지정하도록 원격 시스템에 알려주는 소스 경로가 해당 주소에 포함되어 있어야 합니다. 그러면 로컬 시스템이 메시지의 경로를 이러한 시스템으로 (자동) 지정할 수 있습니다.

exproute 키워드("explicit routing"의 약어)는 주소를 원격 시스템에 전달할 때 연결된 채널에 명시적 라우팅이 필요함을 MTA에 알려 줍니다. 이 키워드를 한 채널에 지정하면 MTA가 해당 채널과 일치하는 모든 헤더 주소와 모든 봉투의 From: 주소에 로컬 시스템의 이름 또는 로컬 시스템의 현재 별칭을 포함하는 라우팅 정보를 추가합니다. 기본값인 noexproute는 라우팅 정보를 추가하지 않도록 지정합니다.

EXPROUTE_FORWARD 옵션을 사용하여 exproute 작업을 역방향 주소로 제한할 수 있습니다. 다른 시나리오는 MTA가 자체적으로 라우팅을 수행할 수 없는 채널을 통해 시스템에 연결하는 경우입니다. 이 경우 부적합한 시스템에 연결되는 채널에 보낸 메일에 해당 주소가 사용되는 시기를 나타내는 라우팅이 다른 채널과 연결된 모든 주소에 있어야 합니다.

암시적 라우팅과 improute 키워드는 이 상황을 처리하는 데 사용됩니다. MTA는 다른 채널과 일치하는 모든 주소가 improute 표시 채널에 보낸 메일에 사용될 경우 라우팅이 필요함을 알고 있습니다. 기본값인 noimproute는 지정된 채널에서 보내는 메시지의 주소에 라우팅 정보를 추가하지 않도록 지정합니다. IMPROUTE_FORWARD 옵션을 사용하여 improute 작업을 역방향 주소로 제한할 수 있습니다.

exproute 및 improute 키워드의 사용을 절제해야 합니다. 이러한 키워드는 길고 복잡한 주소를 만들기 때문에 다른 시스템에 사용되는 지능적인 라우팅 체계를 손상시킬 수 있습니다. 명시적 라우팅과 암시적 라우팅을 지정된 경로와 혼동해서는 안 됩니다. 지정된 경로는 라우팅 정보를 다시 쓰기 규칙에서 주소로 삽입하는 데 사용됩니다. 이 키워드는 특수 A@B@C 다시 쓰기 규칙 템플리트에 의해 활성화됩니다.

지정된 경로를 활성화하면 헤더와 봉투에 있는 모든 주소에 적용됩니다. 지정된 경로는 특정 다시 쓰기 규칙에 의해 활성화되며 일반적으로 현재 사용 중인 채널과는 관련이 없습니다. 다시 말해서, 명시적 라우팅과 암시적 라우팅은 채널 단위로 제어되며 삽입된 경로 주소는 항상 로컬 시스템입니다.

12.6.4 명시적 라우팅 주소의 다시 쓰기 사용 안 함

키워드: routelocal

routelocal 채널 키워드는 채널에 주소를 다시 쓸 때 MTA가 주소에 명시적 라우팅을 "단락"하게 합니다. 명시적 라우팅 주소(!, % 또는 @ 문자 사용)는 단순화됩니다.

내부 TCP/IP 채널과 같은 "내부" 채널에 이 키워드를 사용하면 SMTP 릴레이 차단 구성을 단순화할 수 있습니다.

명시적 % 또는 다른 라우팅이 필요한 채널에서는 이 키워드를 사용하지 마십시오.

12.6.5 메시지를 대기열에서 제거할 때 주소 다시 쓰기

키워드: connectalias, connectcanonical

MTA는 일반적으로 채널 대기열에 메시지를 넣을 때 주소를 다시 씁니다. 메시지를 대기열에서 제거하는 동안에는 추가 다시 쓰기를 수행하지 않습니다. 그렇게 하면 채널 대기열에 이전 이름으로 주소가 지정된 메시지가 있을 경우 호스트 이름을 변경하면 문제가 발생할 수 있습니다.

connectalias 키워드는 수신자 주소에 나열된 모든 호스트에 전달하도록 MTA에 지시하는 기본값입니다. connectcanonical 키워드는 MTA가 연결되는 시스템의 호스트 별칭에 연결하도록 MTA에 지시합니다.

12.6.6 불완전한 주소를 수정할 때 사용할 호스트 이름 지정

키워드: remotehost, noremotehost, defaulthost, nodefaulthost

MTA는 잘못 구성되거나 호환되지 않는 메일 프로그램 및 SMTP 클라이언트로부터 도메인 이름이 포함되지 않은 주소를 받는 경우가 있습니다. 이 경우 MTA는 이러한 주소를 올바른 주소로 만든 후에 전달을 시도합니다. 이를 위해 MTA는 주소에 도메인 이름을 추가(예: @siroe.com을 mrochek에 추가)합니다.

도메인 이름이 없는 봉투의 To: 주소의 경우 MTA는 항상 로컬 호스트 이름을 추가해야 한다고 가정합니다. 그러나 From: 주소와 같은 다른 주소에서는 MTA SMTP 서버의 경우 두 가지 이상의 도메인 이름 선택 항목(로컬 MTA 호스트 이름과 클라이언트 SMTP가 보고한 원격 호스트 이름)이 있습니다. 세 번째 선택 항목(해당 채널에서 추가할 특정 도메인 이름)이 있는 경우도 있습니다. 처음 두 선택 항목이 특정 빈도로 계속해서 발생할 경우 그 중 하나가 올바를 수 있습니다. 잘못 구성된 SMTP 클라이언트를 처리할 경우 원격 호스트 도메인 이름을 사용하는 것이 좋습니다. SMTP가 메시지를 게시하는 데 사용하는 POP 또는 IMAP 클라이언트와 같은 경량 원격 메일 클라이언트를 처리할 경우 로컬 호스트 도메인 이름을 사용하면 안 됩니다. POP 또는 IMAP와 같은 경량 원격 메일 클라이언트의 경우 클라이언트에 로컬 호스트의 도메인 이름이 아닌 해당 클라이언트의 특정 도메인 이름이 있어야 합니다. 그런 다음 다른 특정 도메인 이름을 추가할 수 있습니다. MTA가 채널 단위로 선택하도록 허용하는 것이 가장 좋습니다.

noremotehost 채널 키워드는 로컬 호스트 이름을 사용하도록 지정합니다. 기본값은 noremotehost 키워드입니다.

defaulthost 채널 키워드는 받는 사용자 아이디에 추가할 특정 호스트 이름을 지정하는 데 사용됩니다. 또한, 해당 채널에서 수신하는 주소(봉투의 From: 및 헤더)를 완성하는 데 사용할 도메인 이름이 뒤에 와야 합니다. Submit 채널의 경우 defaulthost 키워드의 첫 번째 인수가 봉투의 To: 주소에도 적용됩니다. 선택적 두 번째 도메인 이름(하나 이상의 점이 있음)을 봉투의 To: 주소를 완성하는 데 사용하도록 지정할 수 있습니다. 기본값은 nodefaulthost입니다.

앞의 347 페이지 “12.4.3.8 받는 메일을 위한 대체 채널(전환 채널)” 절에서 설명한 것처럼 switchchannel 키워드를 사용하여 받는 SMTP 연결을 특정 채널에 연결할 수 있습니다. 이 기능을 사용하면 원격 메일 클라이언트를 적절하게 처리될 수 있는 특정 채널에서 그룹화할 수 있습니다. 또한, MTA 호스트에서 네트워크 차원 문제를 해결하려고 시도하는 것보다는 비호환 클라이언트를 여러 개 사용 중인 경우에도 표준 호환 원격 메일 클라이언트를 배포하는 것이 더 간단합니다.

12.6.7 수신자 헤더 행 없이 메시지 적법화

키워드: missingrecipientpolicy

RFC 822(인터넷) 메시지에는 To:, Cc: 또는 Bcc: 헤더 행과 같은 수신자 헤더 행이 있어야 합니다. 이러한 헤더 행이 없는 메시지는 잘못된 것입니다. 그럼에도 불구하고 sendmail의 많은 이전 버전과 같은 일부 손상된 사용자 에이전트 및 메일 프로그램에서는 잘못된 메시지를 생성합니다.

missingrecipientpolicy 키워드는 그러한 메시지를 사용하는 방법을 지정하는 정수 값을 취합니다. 기본값은 이 키워드가 명백히 표시되지 않은 경우 1(잘못된 메시지를 변경하지 않고 전달)이 됩니다.

표 12–25 missingrecipientpolicy 값

값	작업
0	RFC 2822에서 권장하는 가장 좋은 방법을 사용합니다. 현재는 값 1과 동등합니다.
1	잘못된 메시지를 변경하지 않은 상태로 전달합니다.
2	봉투의 To: 수신자를 `To:` 헤더 행에 넣습니다.
3	모든 봉투의 To: 수신자를 `Bcc:` 헤더 행에 넣습니다.
4	그룹 구조(예: ";") `To:` 헤더 행을 "To: Recipients not specified: ;"로 생성합니다.
5	빈 `Bcc:` 헤더 행을 생성합니다.
6	메시지를 거부합니다.

`MISSING_RECIPIENT_POLICY` 옵션을 사용하여 이 동작에 대한 MTA 시스템 기본값을 설정할 수 있습니다. 초기 Messaging Server 구성에서는 `MISSING_RECIPIENT_POLICY`가 1로 설정됩니다.

12.6.8 잘못된 빈 수신자 헤더 스트라이핑

키워드: `dropblank`, `nodropblank`

RFC 822(인터넷) 메시지에서 `To:`, `Resent-To:`, `Cc:` 또는 `Resent-Cc:` 헤더에는 하나 이상의 주소가 포함되어 있어야 합니다. 이러한 헤더는 빈 값을 가질 수 없습니다. 그럼에도 불구하고 일부 메일 프로그램은 잘못된 헤더를 생성할 수 있습니다. `dropblank` 채널 키워드를 소스 채널에 지정한 경우 MTA는 받는 메시지에서 잘못된 빈 헤더를 제거합니다.

12.6.9 역방향 데이터베이스의 채널별 사용

키워드: `reverse`, `noreverse`

`reverse` 키워드는 채널 대기열에 포함된 메시지의 주소를 주소 역방향 데이터베이스 또는 `REVERSE` 매핑을 통해 검색하여 수정(있는 경우)하도록 지시합니다. `noreverse`는 채널 대기열에 포함된 메시지의 주소를 주소 역방향 처리에서 제외합니다. 기본값은 `reverse` 키워드입니다. 239 페이지 "10.9 주소를 내부 형식에서 공개 형식으로 변환"을 참조하십시오.

12.6.10 제한된 메일함 인코딩 사용

키워드: `restricted`, `unrestricted`

일부 메일 시스템에서는 RFC 822에서 허용하는 모든 형식의 주소를 처리하는 데 어려움이 있습니다. 이러한 공통된 예로는 잘못된 구성 파일을 갖는 sendmail 기반 메일 프로그램이 있습니다. 따옴표가 있는 로컬 부분 또는 메일함 사양이 문제의 원인이 되는 경우가 많습니다.

```
"smith, ned"@siroe.com
```

이러한 것이 문제의 주된 원인이며 RFC 1137에는 이러한 문제를 해결하기 위한 방법론이 나와 있습니다. 기본적인 방법은 주소에서 따옴표를 제거한 다음 따옴표가 필요한 문자를 atom에 허용된 문자로 매핑하는 변환을 적용하는 것입니다. 여기에 사용되는 atom에 대한 정의는 RFC 822를 참조하십시오. 예를 들면 선행 주소는 다음과 같습니다.

```
smith#m#_ned@siroe.com
```

`restricted` 채널 키워드는 채널이 이 인코딩을 필요로 하는 시스템에 연결됨을 MTA에 알려줍니다. 그러면 MTA는 채널에 메시지가 기록될 때 헤더와 봉투 주소 모두에서 따옴표가 있는 로컬 부분을 인코딩합니다. 채널의 받는 주소는 자동으로 디코딩됩니다. `unrestricted` 키워드는 MTA에 RFC 1137 인코딩 및 디코딩을 수행하지 않도록 지시합니다. 기본값은 `unrestricted` 키워드입니다.

주 – 따옴표가 있는 로컬 부분을 적용할 수 없는 시스템에 연결하는 채널에는 `restricted` 키워드를 적용해야 합니다. 따옴표가 있는 로컬 부분을 실제로 생성하는 채널에는 이 키워드를 적용할 수 없습니다. 그런 주소를 생성할 수 있는 채널은 해당 주소를 처리할 수 있다고 가정합니다.

12.6.11 Return-path: 헤더 행 생성

키워드: `addreturnpath`, `noaddreturnpath`

일반적으로 `Return-path:` 헤더 행을 추가하는 것은 최종 전달을 수행하는 채널에서 담당합니다. ims-ms 채널과 같은 일부 채널의 경우 채널에서 `Return-path:` 헤더를 추가하는 것보다 MTA에서 추가하는 것이 더 효과적입니다. `addreturnpath` 키워드는 `Return-path:` 헤더가 이 채널의 대기열에 포함될 때 이 경로를 추가합니다.

12.6.12 봉투의 To: 및 From: 주소에서 Received: 헤더 행 구성

키워드: `receivedfor`, `noreceivedfor`, `receivedfrom`, `noreceivedfrom`

`receivedfor` 키워드는 메시지의 주소가 한 명의 봉투 수신자로만 지정된 경우 해당 봉투의 To: 주소를 `Received:` 헤더 행에 포함하도록 MTA에 지시합니다. 기본값은 `receivedfor` 키워드입니다. `noreceivedfor` 키워드는 봉투 주소 정보를 포함하지 않고 `Received:` 헤더 행을 구성하도록 MTA에 지시합니다.

receivedfrom 키워드는 특정 종류의 메일 목록 확장으로 인해 MTA에서 봉투의 From: 주소를 변경한 경우, 받는 메시지에 대한 Received: 헤더 행을 구성할 때 원래의 From: 주소를 포함하도록 MTA에 지시합니다. 기본값은 receivedfrom입니다. noreceivedfrom 키워드는 원래 From: 주소를 포함하지 않고 Received: 헤더 행을 구성하도록 MTA에 지시합니다.

12.6.13 주소 헤더 행의 주석 처리

키워드: commentinc, commentmap, commentomit, commentstrip, commenttotal, sourcecommentinc, sourcecommentmap, sourcecommentomit, sourcecommentstrip, sourcecommenttotal

MTA는 필요한 경우에만 헤더 행의 내용을 해석합니다. 그러나 주소를 포함하는 모든 등록된 헤더 행의 구문을 분석하여 축약형 주소를 제거한 다음 올바른 주소로 변환해야 합니다. 이 프로세스 중에 헤더 행을 다시 작성할 때 주석(괄호로 묶인 문자열)을 추출하여 수정하거나 제외할 수 있습니다.

이 동작은 commentinc, commentmap, commentomit, commentstrip 및 commenttotal 키워드를 사용하여 제어합니다. commentinc 키워드는 헤더 행에 주석을 유지하도록 MTA에 지시하는 기본값입니다. commentomit 키워드는 To:, From:, Cc: 헤더 행과 같은 주소 지정 헤더에서 주석을 제거하도록 MTA에 지시합니다.

commenttotal 키워드는 Received: 헤더 행을 제외한 모든 헤더 행에서 주석을 제거하도록 MTA에 지시합니다. 이 키워드는 대부분 유용하지 않거나 권장되지 않습니다. commentstrip은 모든 주석 필드에서 nonatomic 문자를 제거하도록 MTA에 지시합니다. commentmap 키워드는 COMMENT_STRINGS매핑 테이블을 통해 주석 문자열을 실행합니다.

소스 채널에서 이 동작은 sourcecommentinc, sourcecommentmap, sourcecommentomit, sourcecommentstrip 및 sourcecommenttotal 키워드에 의해 제어됩니다. sourcecommentinc 키워드는 헤더 행에 주석을 유지하도록 MTA에 지시하는 기본값입니다. sourcecommentomit 키워드는 To:, From:, Cc:, From:, Cc: 등과 같은 주소 지정 헤더)에서 주석을 제거하도록 MTA에 지시합니다. sourcecommenttotal 키워드는 Received: 헤더를 제외한 모든 헤더에서 주석을 제거하도록 MTA에 지시합니다. 따라서, 이 키워드는 대부분 유용하지 않거나 권장되지 않습니다. 마지막으로 sourcecommentstrip 키워드는 모든 주석 필드에서 nonatomic 문자를 스트라이프하도록 MTA에 지시합니다. sourcecommentmap 키워드는 소스 채널을 통해 주석 문자열을 실행합니다.

이러한 키워드는 모든 채널에 적용될 수 있습니다.

COMMENT_STRINGS 매핑 테이블의 구문은 다음과 같습니다.

(*comment_text*) | *address*

항목 템플리트가 $Y 플래그를 설정하는 경우 원래 주석이 지정된 텍스트(괄호로 묶임)로 변경됩니다.

12.6.14 주소 헤더 행에서 개인 이름 처리

키워드: personalinc, personalmap, personalomit, personalstrip, sourcepersonalinc, sourcepersonalmap, sourcepersonalomit, sourcepersonalstrip

다시 쓰기 프로세스 중에 주소를 포함하는 모든 등록된 헤더 행의 구문을 분석하여 축약형 주소를 제거한 다음 올바른 주소로 변환해야 합니다. 이 프로세스 중에 헤더 행을 다시 작성할 때 개인 이름(대괄호로 구분된 주소 앞의 문자열)을 추출하여 선택적으로 수정하거나 제외할 수 있습니다.

이 동작은 personalinc, personalmap, personalomit 및 personalstrip 키워드를 사용하여 제어합니다. personalinc 키워드는 헤더에 개인 이름을 유지하도록 MTA에 지시하는 기본값입니다. personalomit 키워드는 모든 개인 이름을 제거하도록 MTA에 지시합니다. personalstrip 키워드는 모든 개인 이름 필드에서 nonatomic 문자를 제거하도록 MTA에 지시합니다. personalmap 키워드는 PERSONAL_NAMES 매핑 테이블을 통해 개인 이름을 실행하도록 MTA에 지시합니다.

소스 채널에서 이 동작은 sourcepersonalinc, sourcepersonalmap, sourcepersonalomit 또는 sourcepersonalstrip 키워드에 의해 제어됩니다. sourcepersonalinc 키워드는 헤더에 개인 이름을 유지하도록 MTA에 지시하는 기본값입니다. sourcepersonalomit 키워드는 모든 개인 이름을 제거하도록 MTA에 지시합니다. 마지막으로 sourcepersonalstrip은 모든 개인 이름 필드에서 nonatomic 문자를 제거하도록 MTA에 지시합니다. sourcepersonalmap 키워드는 소스 채널을 통해 개인 이름을 실행하도록 MTA에 지시합니다.

이러한 키워드는 모든 채널에 적용될 수 있습니다.

PERSONAL_NAMES 매핑 테이블 검사 구문은 다음과 같습니다.

personal_name | *address*

템플리트에 $Y 플래그를 설정하면 원래의 개인 이름이 지정된 텍스트로 변경됩니다.

12.6.15 별칭 파일 및 별칭 데이터베이스 검사 지정

키워드: aliaslocal

일반적으로 로컬 채널(UNIX의 l 채널)에 다시 작성된 주소만 별칭 파일과 별칭 데이터베이스에서 조회됩니다. aliaslocal 키워드를 채널에 배치하여 별칭 파일과 별칭 데이터베이스에서 해당 채널에 다시 작성된 주소를 조회할 수도 있습니다. 만든 조회 검사의 정확한 형식은 ALIAS_DOMAINS 옵션에 의해 제어됩니다.

12.6.16 하위 주소 처리

키워드: subaddressexact, subaddressrelaxed, subaddresswild

하위 주소의 개념 관련 배경으로 원시 및 ims-ms 채널은 주소의 로컬 부분(메일함 부분)에서 + 문자를 해석합니다. *name+subaddress@domain* 형식 주소에서 MTA는 더하기(+) 문자 뒤의 메일함 부분을 하위 주소로 간주합니다. 원시 채널은 하위 주소를 추가 정보로 취급하고 해당 하위 주소에 관계 없이 계정 이름에 실제로 전달합니다. ims-ms 채널은 하위 주소를 전달할 폴더 이름으로 해석합니다.

하위 주소는 로컬 채널(UNIX의 L 채널)에 의한 별칭 조회, aliaslocal 키워드로 표시한 채널에 의한 별칭 조회, 디렉토리 채널에 의한 메일함 조회 등에도 영향을 미칩니다. 그런 일치를 위해 정확한 하위 주소 처리를 구성할 수 있습니다. 주소를 항목과 비교할 경우 MTA는 항상 하위 주소를 포함한 전체 메일함에서 정확한 일치를 먼저 검사합니다. 그런 다음 MTA가 추가 검사를 수행하는지 여부를 구성할 수 있습니다.

subaddressexact 키워드는 항목 일치 중에 특수 하위 주소 처리를 수행하지 않도록 MTA에 지시합니다. 하위 주소를 포함하여 전체 메일함이 항목과 일치해야 별칭이 일치하는 것으로 간주됩니다. 추가 비교(와일드카드 비교 또는 하위 주소를 제외한 비교)가 수행되지 않습니다. subaddresswild 키워드는 전체 하위 주소를 포함한 정확한 일치를 조사한 다음 name+* 형식 항목을 조사하도록 MTA에 지시합니다. subaddressrelaxed 키워드는 정확한 일치를 조사하고 name+* 형식 일치를 조사한 다음 이름 부분에 대해서만 일치를 조사하도록 MTA에 지시합니다. subaddressrelaxed를 사용하면 다음 형식의 별칭 항목이 name 또는 name+subaddress와 일치되고, 일반 이름이 새 이름으로 변환되며, name+subaddress가 newname+subaddress로 변환됩니다. 기본값은 subaddressrelaxed 키워드입니다.

```
name:   newname+*
```

따라서 subaddresswild 키워드 또는 subaddressrelaxed 키워드는 별칭 또는 디렉토리 채널이 사용 중이지만 임의의 하위 주소를 사용하여 주소가 지정된 메일을 받으려는 경우에 유용할 수 있습니다. 이러한 키워드를 사용하면 주소에서 단일 하위 주소 변형마다 별도의 항목을 둘 필요가 없습니다.

이러한 키워드는 로컬 채널(UNIX의 L 채널) 또는 aliaslocal 키워드로 표시한 채널에만 적용됩니다.

표준 Messaging Server 구성은 실제로 subaddressrelaxed 동작을 하는 L 채널에서 중계됩니다(기본적으로 다른 키워드를 명시하지 않은 경우).

12.6.17 채널별 다시 쓰기 규칙 검사 사용

키워드: rules, norules

rules 키워드는 이 채널에 대해 채널별 다시 쓰기 규칙 검사를 실행하도록 MTA에 지시합니다. 기본값입니다. norules 키워드는 이 채널을 검사하지 않도록 MTA에 지시합니다. 이 두 키워드는 일반적으로 디버깅하는 데 사용되고 실제 응용 프로그램에서는 거의 사용되지 않습니다.

12.6.18 소스 경로 제거

키워드: dequeue_removeroute

dequeue_removeroute 키워드는 메시지를 대기열에서 제거할 때 소스 경로를 봉투의 To: 주소에서 제거합니다. 이 키워드는 현재 tcp-* 채널에서만 구현됩니다. 또한, 소스 경로를 올바르게 처리하지 못하는 시스템에 메시지를 전송할 때 유용합니다.

12.6.19 반드시 별칭을 통해 주소 지정

키워드: viaaliasoptional, viaaliasrequired

viaaliasrequired는 채널과 일치하는 최종 수신자 주소를 별칭을 통해 생성해야 함을 지정합니다. 최종 수신자 주소는 별칭 확장(해당하는 경우)이 수행된 이후의 일치를 참조합니다. 주소를 MTA에 수신자 주소로 직접 전달할 수 없습니다. 즉, 주소를 채널에 다시 쓰는 것만으로는 부족합니다. 채널에 다시 쓴 후 별칭을 통해 주소를 확장해야 채널과 일치하는 것으로 간주됩니다.

예를 들어, 로컬 채널에서 viaaliasrequired 키워드를 사용하여 임의의 계정(예: UNIX 시스템에서 임의의 원시 Berkeley 메일함)에 전달하지 못하게 할 수 있습니다.

기본값은 viaaliasoptional이며 별칭을 사용하여 채널과 일치하는 최종 수신자 주소를 생성할 필요가 없음을 의미합니다.

12.6.20 수신자 주소 처리

키워드: acceptalladdresses, acceptvalidaddresses

acceptvalidaddresses가 기본값이며, SMTP 대화 중에 수신자 오류가 즉시 보고되는 MTA 표준 동작에 해당합니다. 채널에 키워드 acceptalladdresses가 지정된 경우 SMTP 대화 중에 모든 수신자 주소가 허용됩니다. 잘못된 주소는 나중에 DSN이 보내집니다.

12.7 헤더 처리 구성

이 절에서는 헤더 및 봉투 정보를 처리하는 키워드에 대해 설명합니다. 이 장은 다음 내용으로 구성되어 있습니다.

12.7.1 포함 헤더 다시 쓰기

키워드: noinner, inner

필요한 경우에만 헤더 행의 내용을 해석합니다. MIME 메시지는 메시지 내부에 메시지를 포함하는 기능이 있기 때문에 여러 메시지 헤더 집합을 포함할 수 있습니다(메시지/RFC822). MTA는 일반적으로 가장 외부에 있는 메시지 헤더 집합만 해석하고 다시 씁니다. 메시지의 내부 헤더에도 헤더 다시 쓰기를 적용하도록 MTA에 선택적으로 지시할 수 있습니다.

이 동작은 noinner 및 inner 키워드에 의해 제어됩니다. noinner 키워드는 내부 메시지 헤더 행을 다시 쓰지 않도록 MTA에 지시하는 기본값입니다. inner 키워드는 메시지의 구문을 분석하고 내부 헤더를 다시 쓰도록 MTA에 지시합니다. 이러한 키워드는 모든 채널에 적용될 수 있습니다.

12.7.2 선택한 메시지 헤더 행 제거

키워드: headertrim, noheadertrim, headerread, noheaderread, innertrim, noinnertrim

MTA는 메시지에서 선택한 메시지 헤더 행을 잘라내거나 제거하기 위한 채널 단위 기능을 제공합니다. 이 작업은 채널 키워드와 관련 헤더 옵션 파일을 조합하거나 두 가지를 모두 사용하여 수행됩니다. 헤더 옵션 파일 형식에 대해서는 **Sun Java System Messaging Server 6.3 Administration Reference**의 "Header Option Files"에 설명되어 있습니다.

headertrim 키워드는 **원본 메시지 헤더를 처리한 후에** 채널과 연결된 헤더 옵션 파일을 참조하여 해당 대상 채널의 대기열에 포함된 메시지의 헤더를 적절하게 잘라내도록 MTA에 지시합니다. noheadertrim 키워드는 헤더 자르기를 수행하지 않습니다. 기본값은 noheadertrim 키워드입니다.

innertrim 키워드는 내부 메시지 부분(내장된 MESSAGE/RFC822 부분)에서도 헤더 자르기를 수행하도록 MTA에 지시합니다. 기본값인 noinnertrim 키워드는 내부 메시지 부분에서 헤더 자르기를 수행하지 않도록 MTA에 지시합니다.

headerread 키워드는 **원본 메시지 헤더를 처리하기 전에** 채널과 연결된 헤더 옵션 파일을 참조하여 해당 소스 채널에 의해 대기열에 포함된 메시지의 헤더를 적절하게 잘라내도록 MTA에 지시합니다. 반면에 headertrim 헤더 자르기는 메시지를 처리한 이후에 적용되며 소스 채널이 아니라 대상 채널입니다. noheaderread 키워드는 메시지 대기열이 포함된 헤더 자르기를 수행하지 않습니다. 기본값은 noheaderread입니다.

headeromit 및 headerbottom 키워드와 달리 headertrim 및 headerread 키워드는 모든 채널에 적용될 수 있습니다. 메시지에서 필수 헤더 정보를 스트라이프하면 MTA 작업이 잘못 수행될 수 있습니다. 따라서 제거하거나 제한할 헤더를 선택할 경우 각별히 주의하십시오. 이 기능은 선택한 헤더 행을 제거하거나 제한해야 할 경우를 위해 제공됩니다.

주의 – 메시지에서 헤더 정보를 스트라이프하면 MTA 작업이 잘못 수행될 수 있습니다. 따라서 제거하거나 제한할 헤더를 선택할 경우 주의하십시오. 이러한 키워드는 선택한 헤더 행을 제거하거나 제한해야 할 경우를 위해 제공됩니다. 헤더 행을 자르거나 제거하기 전에 해당 헤더의 사용법을 이해하고 제거의 함축된 의미를 고려해야 합니다.

headertrim 및 innertrim 키워드의 헤더 옵션 파일에는 채널이 있는 *channel*_headers.opt 형식 이름과 헤더 옵션 파일이 연결되는 채널 이름이 있습니다. 마찬가지로 headerread 키워드의 헤더 옵션 파일에는 channel_read_headers.opt 형식의 이름이 있습니다. 이러한 파일은 MTA 구성 디렉토리인 *instance_root*/config/에 저장됩니다.

12.7.3 X-Envelope-to 헤더 행 생성/제거

키워드: x_env_to, nox_env_to

x_env_to 및 nox_env_to 키워드는 특정 채널의 대기열에 포함된 메시지 복사본에서 X-Envelope-to 헤더 행의 생성 또는 억제를 제어합니다. single 키워드로 표시한 채널에서 x_env_to 키워드는 이러한 헤더를 생성하고 nox_env_to 키워드는 대기열에 포함된 메시지에서 해당 헤더를 제거합니다. 기본값은 nox_env_to입니다.

이렇게 하는 것은 거의 의미가 없겠지만, 이제 채널에 single을 설정하지 않고도 x_env_to 키워드를 사용할 수 있습니다.

12.7.4 두 자리 또는 네 자리로 날짜 변환

키워드: datefour, datetwo

원본 RFC 822 사양은 메시지 헤더의 날짜 필드에서 두 자리 연도를 호출합니다. 이 연도 표시는 이후에 RFC 1123에서 네 자리로 변경되었지만, 일부 이전 메일 시스템에서는 네 자리 날짜를 사용할 수 없습니다. 또한, 일부 최신 메일 시스템에서는 두 자리 날짜를 더 이상 받아들일 수 없습니다.

주 – 두 형식을 모두 처리할 수 없는 시스템은 표준 위반입니다.

---

datefour 및 datetwo 키워드는 메시지 헤더 날짜에서 MTA의 연도 필드 처리를 제어합니다. 기본값인 datefour 키워드는 모든 연도 필드를 네 자리로 확장하도록 MTA에 지시합니다. 50보다 작은 두 자리 날짜에는 2000이 추가되고 50보다 큰 값에는 1900이 추가됩니다.

주의 – datetwo 키워드는 네 자리 날짜에서 앞의 두 자리를 제거하도록 MTA에 지시합니다. 이 키워드는 두 자리 날짜를 필요로 하는 호환되지 않는 메일 시스템과 호환성을 제공하기 위한 것입니다. 이외의 다른 목적으로 사용해서는 안 됩니다.

12.7.5 날짜의 요일 지정

키워드: dayofweek, nodayofweek

RFC 822 사양은 메시지 헤더의 날짜 필드에서 선행 요일 사양에 허용됩니다. 요일 정보를 적용할 수 없는 시스템도 있습니다. 따라서 요일 정보를 헤더에 표시하면 매우 유용하기는 하지만 일부 시스템에서는 이 정보를 포함하기를 꺼리게 됩니다.

dayofweek 및 nodayofweek 키워드는 MTA의 요일 정보 처리를 제어합니다. 기본값인 dayofweek 키워드는 요일 정보를 유지하고 날짜 및 시간 헤더에 이 정보를 추가(없는 경우)하도록 MTA에 지시합니다.

주의 – nodayofweek 키워드는 날짜 및 시간 헤더에서 선행 요일 정보를 제거하도록 MTA에 지시합니다. 이 키워드는 이 정보를 제대로 처리할 수 없는 호환되지 않는 메일 시스템과 호환성을 제공하기 위한 것입니다. 이외의 다른 목적으로 사용해서는 안 됩니다.

12.7.6 긴 헤더 행 자동 분할

키워드: maxheaderaddrs, maxheaderchars

일부 메시지 전송 프로그램 특히, 일부 sendmail 구현에서는 긴 헤더 행을 제대로 처리할 수 없습니다. 이로 인해 헤더가 손상되지는 않지만 잘못된 메시지 거부를 초래할 수 있습니다. 이는 총체적인 표준 위반이지만 일반적인 문제이기도 합니다.

MTA는 긴 헤더 행을 독립된 여러 헤더 행으로 분할하기 위한 채널 단위 기능을 제공합니다. `maxheaderaddrs` 키워드는 한 행에 표시할 수 있는 주소 수를 제어합니다. `maxheaderchars` 키워드는 한 행에 표시할 수 있는 문자 수를 제어합니다. 두 키워드 모두 연관된 제한을 지정하는 단일 정수 매개 변수를 필요로 합니다. 기본적으로 헤더 행의 길이와 표시할 수 있는 주소 수에는 제한이 적용되지 않습니다.

12.7.7 헤더 맞춤 및 접기

키워드: `headerlabelalign`, `headerlinelength`

`headerlabelalign` 키워드는 이 채널의 대기열에 포함된 메시지 헤더에 대한 맞춤 점을 제어합니다. 맞춤 점은 정수 값 인수를 가집니다. 맞춤 점은 헤더 내용이 정렬되는 여백입니다. 예를 들어, 맞춤 점이 10인 샘플 헤더 행의 모양은 다음과 같습니다.

```
To:       joe@siroe.com
From:     mary@siroe.com
Subject:  Alignment test
```

기본 `headerlabelalign`은 0이고 헤더가 정렬되지 않습니다. `headerlinelength` 키워드는 이 채널의 대기열에 포함된 메시지 헤더 행의 길이를 제어합니다. 이 키워드보다 더 긴 행은 RFC 822 접기 규칙에 따라 접힙니다.

이러한 키워드는 메시지 대기열에 있는 메시지 헤더의 형식만 제어합니다. 헤더의 실제 디스플레이는 일반적으로 사용자 에이전트에 의해 제어됩니다. 또한, 인터넷을 통해 전송할 경우 일반적으로 헤더의 형식이 다시 지정되기 때문에 이러한 키워드는 메시지 헤더의 형식을 다시 지정하지 못하는 단순 사용자 에이전트와 함께 사용하더라도 가시적인 효과가 없을 수 있습니다.

12.7.8 최대 길이 헤더 지정

키워드: `maxprocchars`

많은 주소를 포함하는 긴 헤더 행을 처리하려면 많은 시스템 자원을 소비할 수 있습니다. `maxprocchars` 키워드는 MTA가 처리하고 다시 쓸 수 있는 최대 길이 헤더를 지정하는 데 사용됩니다. 이 키워드보다 더 긴 헤더를 지닌 메시지도 허용되고 전달됩니다. 긴 헤더 행을 어떠한 식으로도 다시 쓰지 않는다는 점만 차이가 납니다. 단일 정수 인수가 필요하며 기본값은 모든 길이의 헤더를 처리합니다.

12.7.9 민감도 검사

키워드: sensitivitynormal, sensitivitypersonal, sensitivityprivate, sensitivitycompanyconfidential

민감도 검사 키워드는 채널에 허용될 수 있는 메시지 민감도에 대한 최대값을 설정합니다. 기본값은 sensitivitycompanyconfidential이며 모든 민감도의 메시지가 허용됩니다. Sensitivity: 헤더가 없는 메시지는 보통(가장 낮은 민감도)으로 간주됩니다. 그런 키워드에 의해 지정된 것보다 더 높은 민감도를 갖는 메시지는 채널의 대기열에 포함되면 거부되고 다음과 같은 오류 메시지가 표시됩니다.

```
message too sensitive for one or more paths used
```

MTA는 이러한 민감도 검사를 수신자 단위가 아니라 메시지 단위로 수행합니다. 한 수신자의 대상 채널에서 민감도 검사에 실패할 경우 해당 메시지는 해당 민감도 채널에 연결된 수신자뿐만 아니라 모든 수신자에게 전달됩니다.

12.7.10 헤더의 기본 언어 설정

키워드: language

헤더의 인코딩된 단어는 특정 언어를 가질 수 있습니다. language 키워드는 기본 언어를 지정합니다.

12.7.11 Message-hash: 헤더 제어

키워드: generatemessagehash, keepmessagehash, deletemessagehash

이 키워드는 메시지의 Message-hash: 헤더를 제어합니다. 대상 채널에 Generatemessage를 지정하면 메시지에 Message-hash: 헤더 필드가 삽입됩니다. Keepmessagehash를 사용하면 기존 Message-hash: 필드가 모두 유지됩니다. Deletemessagehash를 사용하면 기존 Message-hash: 필드가 모두 삭제됩니다. 기본값은 Deletemessagehash입니다.

Message-Hash: 필드에 들어가는 값은 메시지의 해시입니다. 새로운 MTA 옵션 몇 가지로 해시 생성 방법을 제어합니다.

MESSAGE_HASH_ALGORITHM - 해시 알고리즘. 다음 중 하나를 지정할 수 있습니다. md2, md4, md5(기본값), sha1, md128(RIPE-MD128) 또는 md160(RIPE-MD160).

MESSAGE_HASH_FIELDS - 헤더에서 해시까지 순서대로 표시한 쉼표로 구분된 필드 목록입니다. 알려진 헤더 필드는 무엇이든 지정할 수 있습니다. 이 옵션을 지정하지 않으면 기본값으로 "message-id,from,to,cc,bcc, resent-message-id,resent-from,resent-to,resent-cc,resent-bcc, subject,content-id,content-type,content-description"이 ”

12.8 첨부 파일 및 MIME 처리

이 절에서는 첨부 파일 및 MIME 처리를 수행하는 키워드에 대해 설명합니다. 이 장은 다음 내용으로 구성되어 있습니다.

12.8.1 Encoding 헤더 행 무시

키워드: `ignoreencoding`, `interpretencoding`

MTA는 `Yes CHARSET-CONVERSION`을 사용하여 다양한 비표준 메시지 형식을 `MIME`으로 변환할 수 있습니다. 특히, RFC 1154 형식에서는 비표준 `Encoding:` 헤더 행을 사용할 수 있습니다. 일부 게이트웨이에서는 이 헤더 행에 잘못된 정보를 생성하므로 이러한 헤더 행을 무시해야 할 경우도 종종 있습니다. `ignoreencoding` 키워드는 `Encoding:` 헤더 행을 무시하도록 MTA에 지시합니다.

주 – MTA에 `CHARSET-CONVERSION`이 사용되지 않는 경우 이러한 헤더는 항상 무시됩니다. `interpretencoding` 키워드가 기본값이며 이 키워드는 `Encoding:` 헤더 행에 주의하도록 MTA에 지시합니다(다르게 지시되지 않을 경우).

12.8.2 메시지/부분 메시지 자동 조각 모음

키워드: `defragment`, `nodefragment`

MIME 표준은 메시지를 더 작은 여러 부분으로 분할하기 위한 메시지/부분 내용 유형을 제공합니다. 이 기능은 크기 제한이 있는 네트워크를 선회하거나 메시지 조각화에서 "검사점 지정" 형식을 제공할 수 있는 불안정한 네트워크를 선회해야 하는 경우에 유용합니다. 그렇게 하면 메시지 전송 중에 네트워크 오류가 발생하더라도 중복된 작업을 줄일 수 있습니다. 메시지가 대상에 도착한 이후에 자동으로 재어셈블할 수 있도록 각 부분에 정보가 포함됩니다.

`defragment` 채널 키워드 및 조각 모음 채널을 사용하여 MTA에서 메시지를 재어셈블할 수 있습니다. 채널에 `defragment` 표시가 있는 경우 채널의 대기열에 포함된 부분 메시지가 조각 모음 채널 대기열에 대신 포함됩니다. 모든 부분이 도착하면 메시지가 다시 작성되어 대상 위치로 보내집니다. `nodefragment`는 이 특수 처리를 사용하지 않습니다. 기본값은 `nodefragment` 키워드입니다.

12.8.2.1 조각 모음 채널

대상 채널에 defragment 키워드가 있는 경우 메시지는 조각 모음 채널로 라우팅됩니다. 즉, MTA에서 일반적으로 메시지의 대기열에 포함된 대상 채널에 defragment 키워드가 있으면 MTA는 메시지 구조 "내부를 살펴 보고"(MIME 구문 분석) 구조가 MIME 메시지 조각인 것이 확인되면 메시지를 일반 대상 채널로 직접 전송하는 대신 조각 모음 채널로 라우팅합니다.

조각 모음 데이터베이스에는 각 메시지 조각을 받은 호스트를 나타내는 정보를 비롯하여 MTA로 들어오는 메시지 조각에 대한 정보가 포함됩니다. 처음으로 조각을 받고 조각 모음 데이터베이스에 기록하고 나면 같은 조각 모음 데이터베이스를 사용하여 다른 시스템에 수신된 메시지의 다른 모든 부분이 첫 부분을 받은 호스트로 라우팅됩니다. 예를 들면 다음과 같습니다.

1. message/partial; id=123; part=x는 대상/아웃바운드 채널이 있을 위치에 defragment 키워드가 있기 때문에 호스트 1에 도착한 다음 호스트 1에 있는 조각 모음 채널로 라우팅됩니다.
2. 호스트 1에 있는 조각 모음 채널에서는 조각 모음 데이터베이스를 검사하여 이 메시지의 다른 부분이 도착했는지 확인합니다. 도착한 다른 부분이 없으면 조각 모음 채널(호스트 1)은 이 부분을 조각 모음 데이터베이스에 넣고 해당 부분이 호스트 1에 있는 것으로 표시합니다.
3. message/partial; id=123; part=y는 대상/아웃바운드 채널이 있을 위치에 defragment 키워드가 있기 때문에 호스트 2에 도착한 다음 호스트 2에 있는 조각 모음 채널로 라우팅됩니다.
4. 호스트 2에 있는 조각 모음 채널에서는 조각 모음 데이터베이스를 검사하고 이 메시지의 x 부분이 이미 호스트 1에 저장되어 있는 것을 확인합니다. 조각 모음 채널은 메시지 조각을 호스트 1(소스를 @host1이 포함된 주소로 라우팅)로 전송합니다.
5. message/partial' id=123; part=y가 호스트 1에 도착하여 조각 모음 채널로 라우팅되면 조각 모음 채널이 실행되고 해당 부분을 데이터베이스에 입력하는 방식으로 계속됩니다.

조각화된 메시지의 나머지 부분은 모두 메시지의 첫 부분(처음 받은 부분, part=1일 필요는 없음)을 받은 호스트로 전달합니다. 메시지는 호스트의 조각 모음 채널에서 재어셈블되며, 조각 모음된 메시지(또는 조각 모음 시간이 초과된 경우 각 조각이 그대로 전송)이 실제 대상 채널로 전송됩니다. 각 메시지의 "첫" 부분을 받는 호스트에 따라 메시지의 조각 모음에 대한 로드 균형 조정이 이루어집니다.

12.8.2.2 조각 모음 채널 보존 시간

메시지는 제한된 시간 동안만 조각 모음 채널 대기열에 보존됩니다. 첫 번째 배달 실패 알림을 보내도록 지정된 시간의 1/2이 경과하면 메시지의 다양한 부분을 재어셈블하지 않고 보냅니다. 이 시간 값 선택은 조각 모음 채널 대기열의 메시지에 대한 배달 실패 알림을 보내지 않게 합니다.

notices 채널 키워드는 배달 실패 알림을 보내기 전에 경과할 수 있는 시간을 제어하며, 부분적으로 보내기 전에 메시지가 보존되는 시간을 제어합니다. notices 키워드 값을 가능한 조각 모음에 대해 메시지를 보존하려는 시간의 2배로 설정합니다. 예를 들어, notices 값을 4로 설정하면 메시지 조각 모음이 2일 동안 보존됩니다.

```
defragment notices 4
DEFRAGMENT-DAEMON
```

12.8.2.3 조각 모음 및 휴가 캐싱에 NFS 기반 파일 시스템 사용

조각 모음 및 휴가 캐싱에는 NFS 기반 파일 시스템이 자주 사용됩니다. 그 용도 중 하나는 여러 MTA 시스템이 모두 동일한 조각 모음 캐시를 공유하도록 지정하여 여러 시스템 사이에서 조각 모음 데이터베이스를 공유하는 것입니다. 이렇게 하려면 각 시스템의 msg-svr-base/config/*defragment_cache*에서 공유 NFS 디스크 상의 공유 조각 모음 데이터베이스로 사용할 파일로 가는 링크를 만듭니다.

어느 경우에나 휴가 및 조각 모음 캐시에 적절한 NFS 파일 의미를 지원하는 NFS 서버(특히 Solaris NFS와 같이 잠금 요청을 인식하는 서버)를 사용할 수 있습니다. NFS를 사용하는 경우에는 소프트 마운트 옵션을 사용합니다. (기본값은 하드 마운트입니다.)mount timeo 옵션으로 제어되는 시간 초과 값을 비교적 짧게 설정하는 것도 좋습니다(mount_nfs(1M) 설명서 페이지 참조).

NFS 하드 마운트를 사용하는 경우 NFS가 다운되면 여러 시스템의 조각 모음 채널이 중지됩니다. 소프트 마운트를 사용하면 조각 모음 채널이 중지되지 않지만, 조각 모음 캐시를 열 수 없기 때문에 다른 호스트의 조각 모음 채널과 서로 통신할 수 없습니다. 드문 경우이긴 하지만, 모든 메시지의 조각들이 같은 호스트에 처음 도착하는 경우에는 호스트의 조각 모음 채널에서 메시지를 재어셈블하여 계속 전송할 수 있습니다. 그보다는 조각들이 서로 다른 호스트에 도착하여 어셈블되지 않고 관련된 조각 모음 채널의 보존 기간이 만료된 후에 별도의 조각으로 전송될 가능성이 큽니다.

12.8.3 대용량 메시지 자동 조각화

키워드: maxblocks, maxlines

일부 메일 시스템 또는 네트워크 전송 프로그램은 특정 크기 제한을 초과하는 메시지를 처리할 수 없습니다. MTA는 채널 단위로 제한을 적용하는 기능을 제공합니다. 설정된 제한보다 큰 메시지는 여러 개의 작은 메시지로 자동으로 분할(조각화)됩니다. 그런 조각화에 사용되는 내용 유형은 message/partial이며, 동일한 메시지의 각 부분이 서로 연결된 다음 받는 메일 프로그램에 의해 자동으로 재어셈블되도록 고유한 아이디 매개변수가 추가됩니다.

maxblocks 및 maxlines 키워드는 자동 조각화가 활성화되는 크기 제한을 적용하는 데 사용됩니다. 이 두 키워드의 뒤에는 단일의 정수 값이 있어야 합니다. maxblocks 키워드는 메시지에 허용되는 최대 블록 수를 지정합니다. MTA 블록은 일반적으로

1024바이트이지만 MTA 옵션 파일의 BLOCK_SIZE 옵션으로 변경할 수 있습니다. maxlines 키워드는 메시지에 허용되는 최대 행 수를 지정합니다. 필요한 경우 이 두 제한을 동시에 적용할 수 있습니다.

메시지 헤더는 메시지 크기에 어느 정도까지는 포함됩니다. 메시지 헤더는 여러 메시지로 분할될 수 없고 지정된 크기 제한을 초과할 수 없기 때문에 메시지 헤더 크기에는 매우 복잡한 기법이 사용됩니다. 이 논리는 MTA 옵션 파일의 MAX_HEADER_BLOCK_USE 및 MAX_HEADER_LINE_USE 옵션에 의해 제어됩니다.

MAX_HEADER_BLOCK_USE를 사용하여 0에서 1까지의 실수를 지정합니다. 기본값은 0.5입니다. 메시지 헤더는 메시지에 소비할 수 있는 총 블록 수(maxblocks 키워드로 지정)에서 이만큼을 차지할 수 있습니다. 메시지 헤더가 긴 경우 MTA는 MAX_HEADER_BLOCK_USE와 maxblocks를 곱한 값을 헤더 * MAX_HEADER_BLOCK_USE의 크기로 사용합니다(헤더 크기는 실제 헤더 크기와 maxblocks 중 작은 값).

예를 들어, maxblocks가 10이고 MAX_HEADER_BLOCK_USE가 기본값 0.5인 경우 5블록보다 더 큰 메시지 헤더는 5블록 헤더로 취급되고, 메시지의 크기가 5블록 이하일 경우 조각화되지 않습니다. 값이 0인 경우에는 헤더가 메시지 크기 제한에서 무시됩니다.

값이 1인 경우 헤더에 사용 가능한 최대 크기까지 사용할 수 있습니다. 각 조각은 제한을 초과하는지 여부에 관계 없이 항상 메시지 내용의 한 행 이상을 포함하고 있어야 합니다. MAX_HEADER_LINE_USE는 maxlines 키워드와 비슷한 방식으로 동작합니다.

12.8.4 메시지 행 길이 제한 적용

키워드: linelength

SMTP 사양은 최대 1,000바이트를 포함하는 텍스트 행에 사용할 수 있습니다. 보다 엄격한 행 길이 제한이 적용되는 전송 프로그램도 있습니다. linelength 키워드는 채널 단위로 최대 허용 가능한 메시지 행 길이를 제한하는 기법을 제공합니다. 지정된 채널의 대기열에 포함되고 행 길이가 해당 채널에 지정된 제한보다 더 긴 메시지는 자동으로 인코딩됩니다.

MTA에서 사용할 수 있는 다양한 인코딩은 항상 행 길이를 80자 미만으로 줄입니다. 그런 인코딩을 수행한 후 해당 디코딩 필터를 적용하여 원본 메시지를 복구할 수 있습니다.

주 – 인코딩은 행 길이를 80자 미만으로 줄일 수만 있습니다. 80자 미만의 행 길이 값을 지정하면 명시된 제한에 맞는 길이의 행이 생성되지 않을 수 있습니다.

---

linelength 키워드는 전송을 위해 데이터 인코딩에서 "소프트" 줄 바꿈을 수행하게 합니다. 인코딩은 일반적으로 수신하는 쪽에서 디코딩하여 원래의 "긴" 행을 복구합니다. "하드" 줄 바꿈에 대한 자세한 내용은 표 13–7의 "레코드, 텍스트"를 참조하십시오.

12.8.5 멀티파트 및 Message/RFC822 부분의 cContent-transfer-encoding 필드 해석

키워드: `interpretmultipartencoding`, `ignoremultipartencoding` , `interpretmessageencoding`, `ignoremessageencoding`

MIME 사양은 멀티파트 또는 message/rfc822 부분에서 7비트, 8비트 및 이진을 제외한 content-transfer-encoding을 사용할 수 없게 합니다. 그러나 일부 에이전트는 이 사양을 위반하고 멀티파트 및 message/rfc822 개체를 인코딩하고 있습니다. 이에 따라 MTA는 그러한 인코딩을 허용 및 제거하는 코드를 갖고 있습니다. 최근, 이와 다른 표준 위반이 등장했는데 quoted-printable 또는 base63 값을 갖는 content-transfer-encoding 필드가 존재하지만 실제로는 그 부분이 인코딩되지 않는 것입니다. MTA에서 그러한 메시지를 디코딩하려고 시도하면 예상하는 대로 빈 메시지가 만들어집니다.

이 문제가 있는 메시지가 많아졌기 때문에 이를 해결하기 위해 두 쌍의 채널 키워드가 새로 추가되었습니다. 즉 멀티파트 및 message/rfc822 부분에서 content-transfer-encoding 필드의 해석을 활성화하거나 비활성화할 수 있습니다. 첫 번째 쌍은 `interpretmultipartencoding`과 `ignoremultipartencoding`이며, 두 번째 쌍은 `interpretmessageencoding`과 `ignoremessageencoding`입니다. 기본값은 `interpretmultipartencoding`과 `interpretmessageencoding` 입니다.

12.9 메시지, 할당량, 수신자 및 인증 시도의 제한

이 절에서는 메시지 크기 제한, 사용자 할당량 및 권한을 설정하는 키워드에 대해 설명합니다. 이 장은 다음 내용으로 구성되어 있습니다.

12.9.1 성공하지 못한 인증 시도에 대한 제한

키워드: `disconnectbadauthlimit`

이 키워드는 세션의 연결이 끊어지기 전에 세션에 허용되는 인증 시도의 실패 횟수를 제한하는 데 사용할 수 있습니다. 이 옵션의 기본값은 3입니다.

12.9.2 절대 메시지 크기 제한 지정

키워드: blocklimit, noblocklimit, linelimit, nolinelimit, sourceblocklimit

조각화를 사용하면 메시지를 작은 여러 조각으로 자동으로 분할할 수 있지만 관리상의 목적(예: 서비스 거부 공격 방지)으로 정의한 제한보다 더 큰 메시지는 거부하는 것이 좋을 수도 있습니다.

blocklimit, linelimit 및 sourceblocklimit 키워드는 절대 크기 제한을 적용하는 데 사용됩니다. 각 키워드의 뒤에는 단일의 정수 값이 있어야 합니다.

blocklimit 키워드는 메시지에 허용되는 최대 블록 수를 지정합니다. MTA는 이 값보다 더 많은 블록을 포함하는 메시지를 채널의 대기열에 넣으려는 시도를 거부합니다. MTA 블록은 일반적으로 1024바이트이지만 MTA 옵션 파일의 BLOCK_SIZE 옵션으로 변경할 수 있습니다.

sourceblocklimit 키워드는 받는 메시지에 허용되는 최대 블록 수를 지정합니다. MTA는 이 값보다 더 많은 블록을 포함하는 메시지를 채널에 제출하려는 시도를 거부합니다. 다시 말해서 blocklimit는 대상 채널에 적용되고 sourceblocklimit는 소스 채널에 적용됩니다. MTA 블록은 일반적으로 1024바이트이지만 MTA 옵션 파일의 BLOCK_SIZE 옵션으로 변경할 수 있습니다.

사용자 LDAP 속성을 MTA 옵션 LDAP_SOURCEBLOCKLIMIT에 지정하고 이 속성을 보낸 사람의 LDAP 항목에 추가하여 보낸 사람 단위로 소스 블록 제한을 지정할 수도 있습니다. 또한 소스 블록 제한이 보낸 사람의 도메인을 기반으로 지원되기도 합니다. 도메인 LDAP 속성을 MTA 옵션 LDAP_DOMAIN_ATTR_SOURCEBLOCKLIMIT에 지정하고 이 속성을 보낸 사람의 도메인 LDAP 항목에 추가합니다. 이러한 값은 모두 기본값이 없습니다.

linelimit 키워드는 메시지에 허용되는 최대 행 수를 지정합니다. MTA는 이 행 수보다 더 많은 행을 포함하는 메시지를 채널의 대기열에 넣으려는 시도를 거부합니다. 필요한 경우 blocklimit 키워드와 linelimit 키워드를 동시에 적용할 수 있습니다.

MTA 옵션 LINE_LIMIT 및 BLOCK_LIMIT를 사용하여 모든 채널에 비슷한 제한을 적용할 수 있습니다. 이러한 제한은 모든 채널에 적용할 수 있다는 장점이 있습니다. 따라서 MTA 서버는 메시지 수신자 정보를 가져오기 전에 해당 제한을 메일 클라이언트에게 알릴 수 있습니다. 이 기능은 일부 프로토콜에서 메시지 거부 프로세스를 단순화합니다.

기본값은 nolinelimit 및 noblocklimit 채널 키워드이며 제한이 적용되지 않음을 의미합니다. 전역 제한은 LINE_LIMIT 또는 BLOCK_LIMIT MTA 옵션을 통해 적용됩니다.

12.9.3 크기 제한 또는 수신자 수 제한을 초과하는 메시지 대상 다시 지정

키워드: alternatechannel, alternateblocklimit , alternatelinelimit, alternaterecipientlimit

MTA는 수신자 수, 메시지 크기, 메시지 행 수 등에 대한 지정된 제한을 초과하는 메시지의 대상을 대체 대상 채널로 다시 지정하는 기능을 제공합니다. 이 기능은 대상 채널에 포함될 수 있는 alternatechannel, alternateblocklimit, alternatelinelimit 및 alternaterecipientlimit 채널 키워드 집합으로 구현됩니다. alternatechannel 키워드는 사용할 대체 채널의 이름을 지정하는 단일 인수를 가집니다. 다른 키워드는 각각 해당 임계값을 지정하는 정수 인수를 갖습니다. 이러한 임계값을 초과하는 메시지는 원본 대상 채널 대신 대체 채널의 대기열에 포함됩니다.

다음 채널 블록 예에서는 인터넷의 tcp_local 채널로 이동해야 하는 5,000개 이상의 블록을 가진 대용량 메시지가 tcp_big 채널로 대신 이동됩니다.

```
tcp_local smtp ...other keywords... alternatechannel tcp_big alternateblocklimit 5
tcp-daemon


tcp_big smtp ...rest of keywords...
tcp-big-daemon
```

다음 예는 alternate* 채널 키워드를 사용하는 방법을 보여 줍니다.

- 대용량 메시지를 지연된 시간이나 한가한 시간에 전달하려면 alternatechannel(예: tcp_big)이 실행되는 시간을 제어할 수 있습니다.

  한 가지 방법은 imsimta qm 유틸리티의 STOP channel_name 및 START channel_name 명령을 사용하는 것입니다. 이러한 명령은 작업 제어기가 실행하는 사용자 정의 정기 작업 또는 cron 작업을 통해 주기적으로 실행됩니다.
- 작업 제어기가 대용량 메시지가나 수신자가 많은 메시지를 자체 풀에서 처리하려면 alternatechannel을 사용할 수도 있습니다.

  대용량 메시지가나 많은 수신자를 갖는 메시지는 원격 SMTP 서버에서 처리하고 받는 데 더 많은 시간이 걸리기 때문에 대용량 메시지가 작은 메시지의 전달을 지연하지 않도록 작은 메시지가나 수신자가 적은 메시지를 그러한 메시지와 구분할 수 있습니다.

  작업 제어기의 정기 메시지 일정 예약 및 스레드 및 프로세스에 대한 메시지 할당 작업은 대부분의 구성에서 허용됩니다.
- 대용량 메시지 또는 수신자가 많은 메시지에 대한 특수 TCP/IP 채널 시간 초과 값을 설정하려면 alternatechannel을 사용할 수 있습니다.

  특히, 특수 TCP/IP 채널 시간 초과 값 설정은 대용량 메시지 또는 수신자가 많은 메시지를 받는 데 지나치게 많은 시간이 걸리는 원격 호스트에 메시지를 보내려는 경우에 유용할 수 있습니다.

  대부분의 구성에서는 기본 자동 시간 초과 조정으로 충분합니다. 사용자는 기본값에서 값을 조정할 수만 있고 특수 채널은 사용할 수 없습니다. 특히 **Sun Java System Messaging Server 6.3 Administration Reference** 채널 옵션을 참조하십시오.

- 매우 긴 메시지에 대해 특수 MIME 메시지 조각화를 수행하려면 alternatechannel 및 alternateblocklimit 채널 키워드를 maxblocks 채널 키워드와 함께 사용할 수 있습니다.

  일반적으로 지정한 크기를 넘는 메시지를 조각화하려면 일반 아웃바운드 TCP/IP 채널에 원하는 maxblocks 크기를 입력합니다. maxblocks 채널 키워드는 조각화를 수행하는 임계값이자 조각을 만들 크기입니다.

  더 큰 임계값을 트리거하고 더 작은 조각을 만들려면 아웃바운드 TCP/IP 채널에서 alternatechannel 및 alternateblocklimit를 사용할 수 있습니다. 그런 다음 대체 채널에서 maxblock 크기를 사용하여 특정 크기를 넘는 메시지를 조각화할 수 있습니다.

- alternatechannel을 특수 필터링과 함께 사용할 수 있습니다. 예를 들어, 수신자가 많은 메시지는 스팸인 경우 내용을 보다 주의깊게 조사해야 합니다. 보내는 채널을 기준으로 서로 다른 필터링을 수행할 수 있습니다(390 페이지 "12.12.4 메일함 필터 파일 위치 지정"의 destinationfilter 채널 키워드 참조).

  변환 채널을 통해 상대적으로 자원을 많이 사용하는 스캔(예: 바이러스 필터링)을 수행할 경우 매우 큰 메시지에서 자원 문제가 발생할 수 있습니다. 대체 변환 채널을 사용할 수 있습니다. 또는 보내는 채널을 기준으로 일반 변환 채널 내에서 특수 변환 절차를 수행할 수 있습니다.

- 대용량의 보내는 메시지를 자체 채널로 보내려면 alternatechannel을 사용하여 mail.log* 파일 분석 또는 카운터 표시에서 제외할 수 있습니다.

  또한, 전달 통계를 주의깊게 분석할 경우 자체 채널에서 대용량 메시지를 처리하는 것이 좋습니다. 원격 SMTP 호스트에 보내는 대용량 메시지가나 수신자가 많은 메시지는 처리를 완료하는 데 더 많은 시간이 걸리므로 대용량 메시지에 대해서는 일반 메시지와 다른 전달 통계를 생성하기 때문입니다.

12.9.4 할당량이 초과된 사용자에 대한 메일 전달 처리

키워드: holdexquota, noexquota

noexquota 및 holdexquota 키워드는 디스크 할당량이 초과된 Berkeley 메일함 사용자(UNIX) 즉, uid 대체 채널에 전달되는 사용자로 주소가 지정된 메시지의 처리를 제어합니다.

noexquota는 할당량을 초과한 사용자에게 보내는 메시지를 메시지 발송자에게 반환하도록 MTA에 지시합니다. holdexquota는 할당량을 초과한 사용자에게 보내는 메시지를 보관하도록 MTA에 지시합니다. 그런 메시지는 전달이 가능할 때까지 또는 시간이 초과되어 메시지 반환 작업에 의해 보낸 사람에게 반환될 때까지 MTA 대기열에 남아 있습니다.

12.9.5 1000자가 넘는 행이 있는 SMTP 메일 처리

키워드: rejectsmtplonglines, wrapsmtplonglines, truncatesmtplonglines

rejectsmtplonglines는 SMTP에서 허용되는 1000자(CRLF 포함)보다 긴 행이 포함된 메시지를 거부하는 옵션을 추가합니다. 이 영역의 다른 옵션에는 아주 긴 행을 줄 바꿈하는 wrapsmtplonglines, 아주 긴 행을 자르는 truncatesmtplonglines 등이 있습니다. 이러한 키워드는 둘 다 제출에 사용되는 초기 채널(예: tcp_local)에 적용해야 합니다. 이후에 전환되는 채널에는 영향을 주지 않습니다.

12.9.6 일반 및 Filename Content-type 및 Content-disposition 매개 변수의 길이 제어

키워드: parameterlengthlimit 및 nameparameterlengthlimit

parameterlengthlimit는 일반 content-type 및 content-disposition 매개 변수가 잘리는 지점을 제어합니다. 기본값은 1024입니다. nameparameterlengthlimit는 name content-type 및 filename content-disposition 매개 변수가 잘리는 지점을 제어합니다. 기본값은 128입니다. 메시지에서 MIME 처리가 수행되지 않을 경우 가장 외부에 있는 메시지 헤더만 처리됩니다. MIME 처리는 inner 키워드나 문자 세트 변환 사용 등과 같은 다양한 방법으로 사용 가능하게 할 수 있습니다.

12.9.7 메시지 수신자 제한

키워드: recipientlimit 및 recipientcutoff

recipientlimit는 메시지에 대해 허용할 총 수신자 주소 수를 지정합니다. recipientcutoff는 MTA에 제공된 수신자의 총 수를 지정된 값과 비교합니다. 이 값이 지정된 제한을 초과하면 메시지 전달이 허용되지 않습니다. 두 키워드는 모두 단일 정수 인수를 가집니다. 해당 채널 키워드를 지정하지 않은 경우 두 키워드의 기본값은 제한이 없습니다.

보낸 사람이나 보낸 사람의 도메인에 수신자 제한을 설정할 수도 있습니다. 이렇게 하려면 적절한 MTA 옵션(LDAP_RECIPIENTLIMIT, LDAP_RECIPIENTCUTOFF, LDAP_DOMAIN_ATTR_RECIPIENTLIMIT, LDAP_DOMAIN_ATTR_RECIPIENTCUTOFF)에 사용자 또는 도메인 LDAP 속성을 지정하고 해당 속성을 보낸 사람의 사용자 항목이나 도메인 항목에 추가합니다.

12.9.8 헤더 크기 제한

키워드: headerlimit

가장 최상의 주 메시지 헤더의 최대 크기에 제한을 부과합니다. 주 메시지 헤더가 지정된 한계에 이르면 자동으로 잘립니다. 전역 MTA 옵션 HEADER_LIMIT를 설정하면 이 옵션이 이러한 채널 수준 제한을 대체합니다. 기본값은 제한 없음입니다.

12.10 MTA 대기열에서 파일 만들기

이 절에서는 MTA 대기열에서 파일 만들기를 지정하여 디스크 자원을 제어할 수 있는 키워드에 대해 설명합니다. 이 장은 다음 내용으로 구성되어 있습니다.

12.10.1 메시지의 여러 주소 처리 방법 제어

키워드: `multiple`, `addrsperfile`, `single`, `single_sys`

MTA에서는 대기열에 포함된 각 메시지에 여러 대상 주소를 표시할 수 있습니다. 일부 채널 프로그램은 수신자가 한 명이거나, 수신자의 수가 제한되었거나, 메시지 복사본당 하나의 대상 시스템이 있는 메시지만 처리할 수 있습니다. 예를 들어, SMTP 채널 마스터 프로그램은 지정된 트랜잭션에 있는 단일 원격 호스트에 대해서만 연결을 설정하기 때문에 해당 호스트의 주소만 처리할 수 있습니다. 그럼에도 불구하고 모든 SMTP 트래픽에 단일 채널이 일반적으로 사용됩니다.

또 다른 예로 일부 SMTP 서버는 한 번에 처리할 수 있는 수신자 수에 대한 제한을 적용할 수 있지만 이 오류 유형을 처리할 수 없습니다.

`multiple`, `addrsperfile`, `single` 및 `single_sys` 키워드를 사용하여 여러 주소를 처리하는 방법을 제어할 수 있습니다. `single` 키워드는 채널의 각 대상 주소에 대해 별도의 메시지 복사본을 만들어야 함을 의미합니다. tcp_* 채널에 `single`을 사용하면 작업 제어기가 트래픽을 관리하는 방법이 변경되고 일반적인 SMTP 시나리오에 적절하지 않기 때문에 권장되지 않습니다. `single_sys` 키워드는 사용된 각 대상 시스템에 대해 단일의 메시지 복사본을 만듭니다. 기본값인 `multiple` 키워드는 전체 채널에 대해 단일의 메시지 복사본을 만듭니다.

주 – 사용된 키워드에 관계 없이 메시지가 대기열에 있는 각 채널에 대해 해당 메시지 복사본을 하나 이상 만듭니다.

`addrsperfile` 키워드는 채널 대기열의 단일 메시지 파일에 연결될 수 있는 최대 수신자 수에 대한 제한을 적용하여 단일 작업에서 처리되는 수신자 수를 제한하는 데 사용됩니다. 이 키워드에는 메시지 파일에 허용되는 최대 수신자 주소 수를 지정하는 단일의 정수 인수가 필요합니다. 이 수에 도달하면 MTA는 자동으로 추가 메시지 파일을 생성하여 해당 주소를 수용합니다. 기본 `multiple` 키워드는 일반적으로 메시지 파일에 수신자 수 제한을 적용하지 않습니다. SMTP 채널의 기본값은 99입니다.

12.10.2 여러 하위 디렉토리로 채널 메시지 대기열 분산

키워드: `subdirs`

기본적으로 채널의 대기열에 포함된 모든 메시지는 *msg_svr_base*/queue/ *channel-name* 디렉토리에 파일로 저장됩니다. 여기서 *channel-name*은 채널의 이름입니다. 그러나, 많은 수의 메시지를 처리하고 처리 대기 중에 대용량 메시지 파일 저장소를 생성하는 채널(예: TCP/IP 채널)의 경우 해당 메시지 파일을 여러 하위 디렉토리로 분산하여 파일 시스템의 성능을 향상시킬 수 있습니다. subdirs 채널 키워드의 뒤에는 채널에 대한 메시지를 분산할 하위 디렉토리의 수를 지정하는 정수가 와야 합니다. 예를 들면 다음과 같습니다.

```
tcp_local single_sys smtp subdirs 10
```

12.10.3 세션 제한 설정

키워드: disconnectbadcommandlimit, disconnectrecipientlimit, disconnectrejectlimit, disconnecttransactionlimit

네 개의 새로운 채널 키워드는 일정한 수의 오류가 감지된 후에 SMTP 서버가 클라이언트와의 연결을 끊을 수 있는 기능을 제공합니다.

disconnectrecipientlimit - 세션 수신자의 수를 제한합니다.

disconnectrejectlimit - 거부된 수신자의 수를 제한합니다.

disconnecttransactionlimit - 트랜잭션 수를 제한합니다.

disconnectbadcommandlimit - 잘못된 명령의 수를 제한합니다.

이러한 키워드는 모두 세션 제한입니다. disconnectbadcommandlimit를 제외하고 이러한 제한은 모두 MAIL FROM 또는 RSET 명령이 실행될 때 확인됩니다. 이러한 제한 중 하나라도 초과하면 서버는 4xy 오류를 표시하고 연결을 끊습니다. 잘못된 명령 제한은 잘못된 명령이 실행될 때 확인된다는 점만 다릅니다.

12.11 로깅 및 디버깅 구성

이 절에서는 로깅 및 디버깅 키워드에 대해 설명합니다.

12.11.1 로깅 키워드

키워드: logging , nologging, logheader

MTA는 대기열에 포함될 때와 대기열에서 제거될 때 각 메시지를 로깅하는 기능을 제공합니다. logging 및 nologging 키워드는 메시지 로깅을 채널 단위로 제어합니다.

기본적으로 초기 구성에서는 모든 채널에 대해 로깅을 설정합니다. 채널 정의에서 nologging 키워드를 대체하여 특정 채널에 대해 로깅을 비활성화할 수 있습니다.

logheader는 채널 단위로 LOG_HEADER MTA 옵션을 무시합니다. 기본값인 0은 메시지 헤더 로깅을 사용하지 않도록 설정합니다. 자세한 내용은 **Sun Java System Messaging Server 6.3 Administration Reference**의 "Option File"을 참조하십시오.

로깅에 대한 자세한 내용은 25 장을 참조하십시오.

12.11.2 디버깅 키워드

키워드: master_debug, slave_debug, nomaster_debug, noslave_debug

일부 채널 프로그램에는 추가 진단 출력을 생성하여 디버깅을 도와주는 선택 코드가 포함되어 있습니다. 채널 단위로 이 디버깅 출력을 생성하는 데 사용하는 두 채널 키워드가 제공됩니다. 마스터 프로그램에서 디버깅 출력을 생성하는 master_debug 키워드와 슬레이브 프로그램에서 디버깅 출력을 생성하는 slave_debug 키워드입니다. nomaster_debug 및 noslave_debug에 따라 두 디버깅 출력 유형은 기본적으로 사용되지 않습니다.

활성화된 디버깅 출력은 채널 프로그램과 연결된 로그 파일에서 끝납니다. 로그 파일의 위치는 프로그램에 따라 다릅니다. 로그 파일은 일반적으로 로그 디렉토리에 보관됩니다. 마스터 프로그램의 로그 파일 이름은 일반적으로 *x*_master.log 형식입니다. 여기서 *x*는 채널 이름입니다. 슬레이브 프로그램의 로그 파일 이름은 일반적으로 *x*_slave.log 형식입니다.

UNIX에서 l 채널에 대해 master_debug 및 slave_debug를 활성화하면 MTA 디버그 정보가 들어 있는 현재 디렉토리(디렉토리에 쓰기 권한이 있는 경우. 쓰기 권한이 없는 경우 디버그 출력이 stdout으로 이동됨)에 imta_sendmail.log-*uniqueid* 파일이 생성됩니다.

12.11.3 Loopcheck 설정

키워드: loopcheck, noloopcheck

loopcheck 키워드는 MTA가 자체적으로 통신하는지 확인하도록 SMTP EHLO 응답 배너에 문자열을 넣습니다. loopcheck를 설정하면 SMTP 서버가 XLOOP 확장을 광고합니다.

XLOOP를 지원하는 SMTP 서버와 통신할 때 MTA의 SMTP 클라이언트는 광고된 문자열을 MTA 값과 비교하고 클라이언트가 실제로 SMTP 서버와 통신 중인 경우 메시지를 즉시 바운스합니다.

12.12 기타 키워드

이 절에서는 기타 키워드에 대해 설명합니다. 이 장은 다음 내용으로 구성되어 있습니다.

12.12.1 프로세스 채널 대체

키워드: `notificationchannel`, `dispositionchannel`

이러한 키워드는 각각 초기에 전달 상태 알림(DSN) 및 MDN(Message Disposition Notification)을 대기열에 포함하기 위한 장소로 프로세스 채널을 대체합니다. 명명된 채널이 없으면 Messaging Server는 다시 프로세스 채널을 사용합니다.

`notificationchannel`은 초기에 전달 상태 알림(DSN)을 대기열에 포함하기 위한 장소로 프로세스 채널을 대체합니다. 명명된 채널이 없으면 Messaging Server는 다시 프로세스 채널을 사용합니다.

`dispositionchannel`은 초기에 MDN(Message Disposition Notification)을 대기열에 포함하기 위한 장소로 프로세스 채널을 대체합니다. 명명된 채널이 없으면 Messaging Server는 다시 프로세스 채널을 사용합니다.

12.12.2 채널 작업 유형

키워드: `submit`

Messaging Server는 RFC 2476의 메시지 제출 프로토콜을 지원합니다. `submit` 키워드를 사용하여 채널을 제출 전용 채널로 표시할 수 있습니다. 이 키워드는 SMTP 서버가 메시지 제출 전용으로 사용되는 특수 포트에서 실행하는 대부분의 TCP/IP 채널에 유용합니다. RFC 2476은 포트 587을 이러한 메시지 제출용으로 사용하도록 구성합니다.

12.12.3 파이프 채널

키워드: `user`

`user` 키워드는 파이프 채널에서 실행할 아이디를 나타내는 데 사용됩니다.

user에 대한 인수는 일반적으로 소문자로 사용되지만 인수가 따옴표로 묶여 있는 경우 원래의 문자가 그대로 유지됩니다.

12.12.4 메일함 필터 파일 위치 지정

키워드: filter, nofilter, channelfilter, nochannelfilter, destinationfilter, nodestinationfilter, sourcefilter, nosourcefilter, fileinto, nofileinto)

filter 키워드는 원시 및 ims-ms 채널에서 해당 채널에 대한 사용자 필터 파일의 위치를 지정하는 데 사용됩니다. 이 키워드는 필터 파일 위치를 설명하는 필수 URL 인수를 갖습니다. 기본값은 nofilter이며 해당 채널에 대해 사용자 메일함 필터가 사용되지 않음을 의미합니다.

sourcefilter 및 destinationfilter 키워드는 일반 MTA 채널에서 받는 메시지와 보내는 메시지 각각에 적용할 채널 수준 필터를 지정하는 데 사용할 수 있습니다. 이러한 키워드는 채널 필터 파일 위치를 설명하는 필수 URL 인수를 갖습니다. 기본값은 nosourcefilter 및 nodestinationfilter이며 어느 채널 방향에 대해서도 채널 메일함 필터를 사용하지 않음을 의미합니다.

이전의 channelfilter 및 nochannelfilter 키워드는 각각 destinationfilter 및 nodestinationfilter의 동의어입니다.

현재 ims-ms 채널과 LMTP 채널에서만 지원되는 fileinto 키워드는 메일함 필터 fileinto 연산자를 적용할 때 주소를 변경하는 방법을 지정합니다. ims-ms 채널에서의 일반적인 사용법은 다음과 같습니다.

```
fileinto $U+$S@$D
```

위의 키워드는 폴더 이름을 원본 주소에 하위 주소로 삽입하여 원래의 하위 주소를 대체하도록 지정합니다.

LMTP 채널에서의 일반적인 사용법은 다음과 같습니다.

```
fileinto @$4O:$U+$S@$D
```

여기서 $40는 4와 영문자 O입니다(숫자 0이 아님).

12.12.5 스팸 필터 키워드

키워드: destinationspamfilter*X*optin, sourcespamfilter*X*optin, disabledestinationspamfilter*X*, disablesourcespamfilter*X*

destinationspamfilter*X*optin은 사용자나 도메인에서 LDAP_OPTIN*X* LDAP 속성을 사용하여 서비스를 지정하지 않더라도 이 채널을 대상으로 하는 모든 메시지가 필터링 소프트웨어 X를 통해 실행되도록 지정합니다. 필터링 소프트웨어 X는 option.dat의

spamfilter*X*_library에서 정의합니다. optin 매개 변수 앞에는 키워드가 오고, 사용 가능한 매개 변수는 필터링 프로그램에 따라 다릅니다. 예를 들어, Brightmail 매개 변수는 spam, virus 또는 spam, virus입니다. SpamAssassin 매개 변수는 spam입니다.

sourcespamfilter*X*optin 은 이 채널로부터 수신된 모든 메시지가 스팸 필터링 소프트웨어 X를 통해 실행하도록 지정합니다. 필터링 소프트웨어 X는 option.dat의 spamfilter*X*_library에서 정의합니다. 시스템 차원 기본 매개 변수 앞에는 키워드가 오고 사용 가능한 매개 변수는 필터링 프로그램에 따라 다릅니다. switchchannel이 적용되는 경우 이 키워드가 switched-to 채널에 포함됩니다.

sourcespamfilter*X* 및 destinationspamfilter*X*는 destinationspamfilter*X*optin 및 destinationspamfilter*X*optin과 동일한 기능을 수행하지만, optin 매개 변수를 허용하지 않습니다. 매개 변수를 전달하지 않고 단순히 활성화되거나 활성화되지 않는 필터링 소프트웨어와 함께 사용됩니다.

disabledestinationspamfilter *X*는 이 채널을 대상으로 하는 메시지에 대해 스팸 필터 X를 비활성화합니다. 스팸 필터 X를 활성화한 채널로부터 메시지를 수신한 경우(예: destinationspamfilter *X*optin) 또는 사용자나 도메인 LDAP 항목에서 optin 속성을 사용하여 필터를 활성화한 경우, 이 키워드는 스팸 필터 X를 비활성화합니다.

disablesourcespamfilter*X*는 이 채널로부터 수신한 메시지에 대해 스팸 필터 X를 비활성화합니다. 스팸 필터 X를 활성화하는 채널로 메시지를 보내는 경우(예: destinationspamfilter *X*optin) 또는 사용자나 도메인 LDAP 항목에서 optin 속성을 사용하여 필터를 활성화한 경우, 이 키워드는 스팸 필터 X를 비활성화합니다.

이러한 키워드를 사용하는 방법에 대한 자세한 내용은 437 페이지 “채널 수준 필터링 지정”을 참조하십시오.

12.12.6 주소 검증 후와 확장 전의 라우팅

키워드: aliasdetourhost, aliasoptindetourhost

aliasdetourhost 및 aliasoptindetourhost를 사용하면 호스트된 사용자의 mailHost 속성 값을 소스 채널별로 대체할 수 있습니다. 특히 aliasdetourhost는 일반적으로 로컬(이 시스템에서 호스트되는) 사용자를 대상으로 하는 메시지를 라우팅할 때 특정 종류의 처리를 위해 별도의 호스트로 “우회”하는 데 사용됩니다. 메시지를 원래 호스트에서 확인(해당 주소가 정당한 로컬 주소인지)하고 처리 호스트로 우회한 다음 확장 및 전달을 위해 원래 호스트로 되돌려 보낼 수 있습니다. aliasdetourhost를 언급하는 경우에는 사용자가 LDAP를 따르는 속성을 통해 선택된 경우에만 우회한다는 점을 제외하고 aliasdetourhost와 비슷하게 작동하는 aliasoptindetourhost도 설명합니다.

aliasdetourhost는 채널 및 타사 필터링 호스트를 더 적절하게 구성하고 일종의 “중간 필터링”을 사용할 수 있게 합니다. aliasdetourhost는 일반적으로 대체변환 채널과 함께

사용됩니다. 대체 변환 채널이 원격 수신자의 라우팅에 영향을 주는 데 사용되는 것과는 달리, aliasdetourhost는 로컬(이 시스템에서 호스팅되는) 사용자의 라우팅에 영향을 주는 데 사용됩니다.

aliasdetourhost의 인수는 호스트 또는 도메인 이름이거나 호스트/도메인 지정입니다. 다시 쓰기 규칙을 사용하여 호스트 이름, IP 리터럴 주소 및 채널 태그(암시적으로 호스트 이름으로 간주되는)를 처리할 수 있습니다. 소스 채널에서 이 키워드를 지정하는 경우 태그 정보가 처리된 후 메일 호스트 정보가 확인되기 직전에 LDAP에 저장된 주소의 별칭 확장이 중지됩니다. 이 때 메시지가 aliasdetourhost 값으로 보내지고 처리된 주소가 성공적으로 완료되지만 이러한 작업은 주소 검증이 끝나고 별칭 확장이 수행되기 전에 발생합니다.

aliasdetourhost를 사용하여 변환 채널 필터링과 관련된 다양한 문제를 방지하는 예는 다음과 같습니다. 여기서는 시스템이 프런트엔드 MTA 및 백엔드 메시지 저장소를 사용하여 설정되었다고 가정합니다. 사용자의 전달 옵션은 forward 및 mailbox로 설정되어 있습니다. MTA에서는 바이러스 백신/스팸 시스템을 위해 대체 변환 채널을 사용합니다. 이 사용자에게 메시지가 도착하면 MTA 별칭이 확장되고 두 명의 수신자(하나는 로컬, 다른 하나는 원격)가 생성됩니다. 원격 수신자의 복사본은 직접 전송됩니다. 반면, 로컬 수신자의 복사본은 변환 채널로 이동하여 스캔된 다음 반환됩니다. 그런 다음 별칭 확장이 다시 적용되어 원격 수신자의 두 번째 복사본이 생성되며 로컬 수신자의 복사본은 정상적으로 전달됩니다. 최종적으로 원격 수신자에 대한 두 개의 복사본과 로컬 수신자에 대한 하나의 복사본이 생성됩니다.

aliasdetourhost를 사용하는 채널에서는 로컬로 호스트된 사용자에 대해 대체 변환 채널을 사용하지 않고(단, 다른 수신자에 대해서는 여전히 대체 변환 채널을 사용할 수 있음) 다음 작업을 수행할 수 있습니다.

- 메시지를 수락합니다.
- 메시지를 외부 스팸/바이러스 필터에 라우팅합니다.
- 주소 확장 및 전달을 위해 메시지를 다시 수락합니다.

예 1:

타사 스캐너가 MTA에서 별개의 호스트에 실행 중이라고 가정합니다. 다음 예는 메시지를 수락하기 전에 수신자 주소 검증을 수행하는 기능을 유지하면서도 가짜 복제본을 만들지 않고 사용자 항목을 전달할 수 있게 합니다.

1. 새 tcp_scanner 채널을 만듭니다.

   daemon 키워드를 해당 채널에 넣어 필터링 시스템을 가리킵니다. enqueue_removeroute도 이 채널에 추가합니다. tcp_scanner 채널은 imta.cnf에서 다음과 같이 표시됩니다.

```
tcp_scanner smtp mx single_sys subdirs 20 noreverse maxjobs 7
pool SMTP_POOL daemon my_a-v_filter.siroe.com enqueue_removeroute
tcp_scanner-daemon
```

2. 스캔할 모든 인바운드 소스 tcp 채널(tcp_local, tcp_submit, tcp_intranet, tcp_auth 등)에서 aliasDetourHost tcp_scanner-daemon을 tcp_local에 추가합니다. 다음은 tcp_local 및 tcp_submit에 대한 예입니다.

```
! tcp_local
tcp_local smtp mx single_sys remotehost inner switchchannel
identnonenumeric subdirs 20 maxjobs 7 pool SMTP_POOL maytlsserver
maysaslserver saslswitchchannel tcp_auth missingrecipientpolicy 0
aliasdetourhost tcp_scanner-daemon
tcp-daemon

! tcp_submit
tcp_submit submit smtp mx single_sys mustsaslserver maytlsserver
missingrecipientpolicy 4 aliasdetourhost tcp_scanner-daemon
tcp_submit-daemon
```

   aliasdetourhost(tcp_scanner-daemon)의 인수는 새 채널 tcp_scanner의 공식 호스트 이름입니다.

3. tcp_scanner 채널을 통해 스캔 시스템에서 메시지를 다시 수신하기 위해 다시 쓰기 규칙을 작성합니다.

   ```
   [1.2.3.4] $E$R$U[1.2.3.4]@tcp_scanner-daemon
   ```

   여기서 1.2.3.4는 스캐너 시스템의 IP 주소입니다.

   이 다시 쓰기 규칙이 없으면 메시지가 다른 tcp* 소스 채널을 통해 들어오며 모든 메시지에 aliasdetourhost가 있기 때문에 메시지가 다시 스캔됩니다. 루프가 발생합니다.

4. 구성을 다시 컴파일하고 디스패처를 다시 시작합니다.

```
#imsimta cnbuild
#imsimta restart dispatcher
```

예 2:

타사 스캐너가 MTA와 동일한 호스트에서 실행 중이지만 다른 포트를 수신한다고 가정합니다. 메일이 포트 10024에서 수락되며 포트 10025에서 릴레이된다고 가정합니다.

1. 새 tcp_scanner 채널을 만듭니다.

```
! tcp_scanner
tcp_scanner smtp nomx single_sys identnonenumeric subdirs 20 maxjobs
7 pool SCAN_POOL daemon 127.0.0.1 port 10024 enqueue_removeroute
tcp_scanner-daemon
```

2. 스캔할 모든 인바운드 소스 tcp 채널(tcp_local, tcp_submit, tcp_intranet 등)에서 aliasDetourHost tcp_scanner-daemon을 tcp_local에 추가합니다. 다음은 tcp_local 및 tcp_submit에 대한 예입니다.

```
! tcp_local
tcp_local smtp mx single_sys remotehost inner switchchannel
identnonenumeric subdirs 20 maxjobs 7 pool SMTP_POOL maytlsserver
maysaslserversaslswitchchannel tcp_auth missingrecipientpolicy 0
aliasdetourhost tcp_scanner-daemon
tcp-daemon

! tcp_submit
tcp_submit submit smtp mx single_sys mustsaslserver maytlsserver
missingrecipientpolicy 4 aliasdetourhost tcp_scanner-daemon
tcp_submit-daemon
```

3. tcp_scanner 채널을 통해 아웃바운드 메일을 다시 라우팅하기 위해 mappings 파일에 추가합니다.

```
CONVERSIONS

in-chan=tcp_scanner;out-chan=*;CONVERT     No
in-chan=tcp_*;out-chan=tcp_local;CONVERT   Yes,Channel=tcp_scanner
```

4. SMTP_POOL 아래의 job_controller.cnf에서 동시 스캔 수에 대한 제한을 추가합니다.

   스캔 소프트웨어에도 제한이 있어야 하지만 Messaging Server가 메시지를 수락하지 않을 경우에 스캐너로 메일을 전송하지 않도록 하기 위해서 이 설정을 동일하게 유지하는 것이 좋습니다.

```
!
[POOL=SCAN_POOL]
job_limit=2
!
```

5. 새 서비스를 dispatcher.cnf에 추가하여 특정 포트에서 스캐너로부터 되돌아오는 메일을 수락하고 메일이 다시 스캔되지 않도록 tcp_scan에서 메일을 가져옵니다.

```
!
[SERVICE=SMTP_SCANNING]
INTERFACE_ADDRESS=127.0.0.1
PORT=10025
IMAGE=IMTA_BIN:tcp_smtp_server
LOGFILE=IMTA_LOG:tcp_smtp_server.log
STACKSIZE=2048000
PARAMETER=CHANNEL=tcp_scanner
!
```

6. 구성을 다시 컴파일하고 디스패처를 다시 시작합니다.

```
# imsimta cnbuild
# imsimta restart job_controller
# imsimta restart dispatcher
```

12.12.6.1 aliasoptindetourhost

이제 다음 기능을 사용하여 사용자별로 aliasdetourhost를 적용할 수 있습니다.

- aliasoptindetourhost 채널 키워드. 사용자가 다음 속성을 통해 선택된 경우에만 우회가 발생한다는 점을 제외하면 aliasdetourhost의 기능과 비슷합니다. 키워드 값은 우회에 사용할 수 있는 호스트를 쉼표로 구분하여 표시한 목록입니다.
- 사용자를 우회에 선택하는 데 사용된 속성 이름을 지정하는 LDAP_DETOURHOST_OPTIN MTA 옵션(소스 채널에 aliasoptindetourhost가 설정되어 있다고 가정). 이 속성 값에 마침표가 포함되어 있으면 이 값을 우회에 사용할 수 있는 호스트의 목록과 비교하고 목록에서 처음으로 일치하는 호스트를 선택하여 우회를 적용합니다. 값에 마침표가 포함되어 있지 않은 경우에는 첫 번째 우회 호스트를 무조건 사용합니다.
- SPAMFILTERx_NULL_OPTIN과 비슷한 ALIASDETOURHOST_NULL_OPTIN MTA 옵션(표 14–1 참조). LDAP_DETOURHOST_OPTIN 속성에 사용된 경우 제외되는 속성과 동일하게 취급되는 특수 값을 지정합니다. 기본값은 ""이며 빈 속성 값이 무시됨을 의미합니다.

12.12.7 NO-SOLICIT SMTP 확장 지원

키워드: sourcenosolicit 및 destinationnosolicit

draft-malamud-no-soliciting-07.txt에 설명된 NO-SOLICIT SMTP 확장은 Messaging Server에서 제안된 표준으로 구현되었습니다. 다음 채널 키워드를 사용하여 이 기능을 제어할 수 있습니다.

sourcenosolicit는 이 채널이 전송하는 메일에서 차단될 요청 필드 값의 쉼표로 구분된 목록을 지정합니다. 이 값 목록은 NO-SOLICIT EHLO 응답에 표시됩니다. 글롭 스타일 와일드카드를 이러한 값에서 사용할 수 있지만 와일드카드를 포함하는 값은 EHLO 알림에 표시되지 않습니다.

destinationnosolicit는 이 채널의 대기열에 포함된 메일에서 허용되지 않는 요청 필드 값의 쉼표로 구분된 목록을 지정합니다.

12.12.8 잘못된 RCPT TO: 주소에 대한 제한 설정

키워드: deferralrejectlimit

단일 세션에서 허용되는 잘못된 RCPT TO: 주소의 수에 제한을 설정합니다. 지정된 수의 To: 주소가 거부되면 모든 후속 수신자(잘못되었는지 여부에 상관없이)는 4xx 오류와 함께 거부됩니다. ALLOW_REJECTIONS_BEFORE_DEFERRAL SMTP 채널 키워드와 동일한 기능을 채널 단위로 제공합니다.

12.12.9 모니터링 프레임워크에 대한 채널 화면 표시 설정

키워드: caption 및 description

이 키워드는 따옴표로 묶은 문자열을 인수로 간주하며, 모니터링 프레임워크 콘솔에서 채널 화면 표시에 쓰입니다. 캡션이나 설명이 존재하지 않으면 모니터링 프레임워크 에이전트는 채널 이름으로부터 하나를 만듭니다.

◆ ◆ ◆ 13장

13 미리 정의된 채널 사용

Messaging Server를 처음 설치하면 여러 채널이 미리 정의되어 있습니다(표 13–1 참조). 이 장에서는 MTA에서 미리 정의된 채널 정의를 사용하는 방법에 대해 설명합니다.

10장을 아직 읽지 않은 경우 이 장을 읽기 전에 10장을 읽어 보십시오. `imta.cnf` 파일의 다시 쓰기 규칙 구성에 대한 자세한 내용은 11장을 참조하십시오.

이 장은 다음 내용으로 구성되어 있습니다.

`defaults` 채널에 대해서는 292 페이지 "12.1 채널 기본값 구성"에 설명되어 있습니다.

13.1 미리 정의된 채널

아래 표에서는 미리 정의된 채널 중 일부를 보여 줍니다.

표 13–1 미리 정의된 채널

채널	정의
`defaults`	여러 채널의 기본값이 되는 키워드를 지정하는 데 사용됩니다. 292 페이지 "12.1 채널 기본값 구성"을 참조하십시오.
`l`	UNIX 전용입니다. 라우팅 결정을 내리고 UNIX 메일 도구를 사용하여 메일을 전송하는 데 사용됩니다.
`ims-ms`	로컬 저장소에 메일을 최종 전달합니다.

표 13–1 미리 정의된 채널 (계속)

채널	정의
native	UNIX 전용입니다. 메일을 /var/mail에 전달합니다. (Messaging Server는 /var/mail 액세스를 지원하지 않습니다. 사용자가 UNIX 도구를 사용하여 /var/mail 저장소에서 메일을 액세스해야 합니다.)
pipe	사이트에서 제공하는 프로그램 또는 스크립트를 통해 전달을 수행하는 데 사용됩니다. pipe 채널에서 실행되는 명령은 imsimta 프로그램 인터페이스를 통해 관리자가 제어합니다.
reprocessprocess	이러한 채널은 지연된 오프라인 메시지를 처리하는 데 사용됩니다. reprocess 채널은 일반적으로 소스 또는 대상 채널로 표시되지 않고 process 채널은 다른 MTA 채널과 마찬가지로 표시됩니다.
defragment	MIME 조각화된 메시지를 재어셈블할 수 있습니다.
conversion	MTA를 통해 주고 받는 메시지에 대해 본문 부분 단위 변환을 수행합니다.
bitbucket	삭제해야 하는 메시지에 사용됩니다.
inactive/deleted	디렉토리에 비활성/삭제됨으로 표시된 사용자의 메시지를 처리하는 데 사용됩니다. 일반적으로 메시지를 바운스하고 메시지를 보낸 사람에게 사용자 정의 바운스 메시지를 반환합니다.
hold	사용자의 메시지를 보관하는 데 사용됩니다. 예를 들어, 사용자가 한 메일 서버에서 다른 메일 서버로 마이그레이션하는 경우에 이러한 사용자의 메일을 보관합니다.
sms	SMS 게이트웨이에 단방향 전자 메일 지원을 제공합니다.
tcp_local tcp_intranet tcp_auth tcp_submit tcp_tas	TCP/IP를 통해 SMTP를 구현합니다. 다중 스레드 TCP SMTP 채널은 디스패처의 제어에 따라 실행되는 다중 스레드 SMTP 서버를 포함합니다. 보내는 SMTP 메일은 tcp_smtp_client 채널 프로그램에서 처리되고 필요한 경우 작업 제어기의 제어에 따라 실행됩니다. tcp_local은 원격 SMTP 호스트로부터 인바운드 메시지를 받습니다. 스마트 호스트/방화벽 구성을 사용하는지 여부에 따라 아웃바운드 메시지를 원격 SMTP에 직접 보내거나 스마트 호스트/방화벽 시스템으로 보냅니다. 때때로 tcp_local은 프록시나 방화벽을 통해 원격 SMTP 호스트로부터 메일을 받습니다. 또한 내부 중계 활동에 사용되는 경우도 있습니다. tcp_intranet은 인트라넷에서 메시지를 보내고 받습니다. tcp_auth는 tcp_local에 대한 전환 채널로 사용되며 인증된 사용자를 tcp_auth 채널로 전환하여 릴레이 차단 제한을 방지합니다. tcp_submit는 예약된 제출 포트 587에서 메시지 제출(주로 사용자 에이전트로부터)을 허용합니다(RFC 2476 참조). tcp_tas는 사이트에서 통합 메시징을 수행하는 데 사용되는 특수 채널입니다.

13.2 파이프 채널을 사용하여 메시지를 프로그램에 전달

받는 메일이 메일함 대신 프로그램에 전달되게 할 수 있습니다. 예를 들어, 사용자는 받는 메일을 메일 정렬 프로그램으로 보내기를 원할 수 있습니다. pipe 채널은 사이트에서 제공하는 사용자 단위 프로그램을 사용하여 메시지 전달을 수행합니다.

프로그램 전달을 쉽게 수행하려면 pipe 채널에서 호출 가능하도록 먼저 프로그램을 등록해야 합니다. imsimta program 유틸리티를 사용하여 등록합니다. 이 유틸리티는 pipe 채널에서 호출이 가능하도록 등록하는 각 명령에 고유한 이름을 제공합니다. 그런 다음 최종 사용자는 메소드 이름을 mailprogramdeliveryinfo LDAP 속성 값으로 지정할 수 있습니다.

예를 들어, UNIX 명령 myprocmail을 사용자가 호출할 수 있는 프로그램으로 추가하려면 먼저 다음 예에 표시된 것처럼 imsimta program 유틸리티를 사용하여 명령을 등록합니다. 이 예에서는 -d *username* 인수를 사용하여 procmail 프로그램을 실행하고 사용자로 실행되는 myprocmail 프로그램을 등록합니다.

```
imsimta program -a -m myprocmail -p procmail -g "-d %s" -e user
```

programs 디렉토리 *msg-svr-base*/data/site-programs에 실행 파일이 존재하는지 확인합니다. 또한 실행 권한이 "others"로 설정되었는지도 ”

사용자가 프로그램에 액세스하려면 해당 사용자의 LDAP 항목에 다음 속성과 값이 포함되어 있어야 합니다.

```
maildeliveryoption: program
mailprogramdeliveryinfo: myprocmail
```

imsimta program 유틸리티에 대한 자세한 내용은 **Sun Java System Messaging Server 6.3 Administration Reference**를 참조하십시오.

대체 전달 프로그램은 다음 종료 코드 및 명령줄 인수 제한을 준수해야 합니다.

종료 코드 제한. pipe 채널에서 호출되는 전달 프로그램은 메시지를 대기열에서 제외할지, 나중에 처리하도록 전달할지 또는 메시지를 반환할지 여부를 채널이 알 수 있도록 의미 있는 오류 코드를 반환해야 합니다.

하위 프로세스가 종료 코드 0(EX_OK)으로 끝나는 경우 해당 메시지가 성공적으로 전달되고 MTA 대기열에서 제거되는 것으로 간주합니다. 하위 프로세스가 종료 코드 71, 74, 75 또는 79(EX_OSERR, EX_IOERR, EX_TEMPFAIL 또는 EX_DB)로 끝나는 경우 일시적인 오류가 발생하여 메시지 전달이 지연되는 것으로 간주합니다. 다른 종료 코드가 반환되는 경우 해당 메시지는 메시지 발송자에게 전달 불가능 메시지로 반환됩니다. 이러한 종료 코드는 sysexits.h 시스템 헤더 파일에 정의됩니다.

명령줄 인수. 전달 프로그램은 변수 인수 %s뿐 아니라 고정 인수를 가질 수 있습니다. 이를 사용하여 사용자가 실행한 프로그램의 아이디를 나타내거나 포스트마스터인 “inetmail”이 실행하는 프로그램의 아이디+도메인을 나타낼 수 있습니다.예를 들어, 다음 명령줄은 procmail 프로그램을 사용하여 수신자의 메일을 전달합니다.

```
/usr/lib/procmail -d %s
```

13.3 원시(/var/mail) 채널 구성

옵션 파일을 사용하여 원시 채널의 다양한 특성을 제어할 수 있습니다. 이 원시 채널 옵션 파일은 MTA 구성 디렉토리에 저장하고 native_option(예: *msg-svr-base*/config/native_option)으로 이름을 지정해야 합니다.

옵션 파일은 여러 행으로 구성됩니다. 각 행에는 하나의 옵션에 대한 설정값이 포함되어 있습니다. 옵션의 형식은 다음과 같습니다.

option=value

*value*는 옵션의 요구 사항에 따라 문자열 또는 정수일 수 있습니다.

표 13–2 로컬 채널 옵션

옵션	설명
FORCE_CONTENT_LENGTH (0 또는 1, UNIX 전용)	FORCE_CONTENT_LENGTH=1이면 MTA는 원시 채널에 전달된 메시지에 Content-length: 헤더 행을 추가하여 "From"이 행의 시작 부분에 있을 경우 채널에서 ">From" 구문을 사용하지 않게 합니다. 그렇게 하면 로컬 UNIX 메일이 Sun의 최신 메일 도구와 호환되지만 다른 UNIX 메일 도구와는 호환되지 않을 수 있습니다.
FORWARD_FORMAT(문자열)	사용자의 .forward 파일 위치를 지정합니다. %u 문자열은 각 사용자 아이디에서 대체됨을 나타내고, %h 문자열은 각 사용자의 홈 디렉토리에서 대체됨을 나타냅니다. 이 옵션을 명확하게 지정하지 않을 경우 기본 동작은 다음과 같습니다. FORWARD_FORMAT=%h/.forward
REPEAT_COUNT (integer) SLEEP_TIME (integer)	MTA가 새 메일을 전달하려고 시도할 때 다른 프로세스에 의해 사용자의 새 메일 파일이 잠긴 경우 이러한 옵션을 사용하여 원시 채널 프로그램이 수행하는 시도 횟수와 빈도를 제어할 수 있습니다. 지정된 횟수만큼의 시도 후에도 파일을 열 수 없는 경우 메시지를 원시 대기열에 그대로 두고 해당 원시 채널이 다음에 실행될 때 새 메시지를 다시 전달하려고 시도합니다. REPEAT_COUNT 옵션은 채널 프로그램이 메일 파일을 열기 위해 시도하는 횟수를 제어합니다. REPEAT_COUNT 기본값은 30(30회 시도)입니다. SLEEP_TIME 옵션은 채널 프로그램이 시도 간에 대기하는 시간(초)을 제어합니다. SLEEP_TIME 기본값은 2(시도 간격: 2초)입니다.
SHELL_TIMEOUT(정수)	.forward의 사용자 쉘 명령이 완료될 때까지 채널이 대기하는 시간(초)을 제어합니다. 이러한 시간 초과가 발생하면 "*user* 쉘 명령 *command* 완료 시간 초과"와 비슷한 오류 메시지와 함께 원래의 보낸 사람에게 메시지가 반환됩니다.기본값은 600(10분)입니다.

표 13–2 로컬 채널 옵션 (계속)

옵션	설명
SHELL_TMPDIR(디렉토리별)	로컬 채널에서 쉘 명령에 전달할 때 임시 파일을 만드는 위치를 제어합니다. 기본적으로 그런 임시 파일은 사용자의 홈 디렉토리에 만들어집니다. 관리자는 이 옵션을 사용하여 임시 파일을 다른(단일) 디렉토리에 만들도록 선택할 수 있습니다. 예를 들면 다음과 같습니다. SHELL_TMPDIR=/tmp

13.4 보관 채널을 사용하여 메시지를 일시적으로 보관

보관 채널은 새 메일 받기가 일시적으로 금지된 수신자의 메시지를 보관하는 데 사용됩니다. 사용자의 이름이 변경되었거나 메일함이 다른 메일 호스트 또는 도메인으로 이동된 경우에 메시지를 보관할 수 있습니다. 다른 이유로 메시지를 일시적으로 보관할 수도 있습니다.

메시지를 보관할 경우 메시지를 reprocess 채널로 이동하는 데 사용되는 것과 동일한 기법으로 메시지를 *msg-svr-base*/queue/hold 디렉토리의 보관 채널로 이동합니다. 그렇게 해도 봉투의 To: 주소는 변경되지 않습니다. 메시지는 *msg-server*/queue/hold 디렉토리의 보관 채널 대기열에 *ZZxxx*.HELD 파일로 작성됩니다. 이렇게 하면 메시지가 작업 제어기에 의해 표시되지 않고 "보관"됩니다..HELD 파일의 목록을 보려면 imsimta qm dir -held 명령을 사용합니다. 이러한 메시지를 선택한 다음 imsimta qm release 명령을 사용하여 해제할 수 있습니다. 메시지를 해제하면 메시지 이름이 *ZZxxx*.00으로 변경되고 작업 제어기에 알림이 전달됩니다. 그러면 보관 채널과 연결된 마스터 프로그램 reprocess.exe에서 해당 메시지를 처리합니다. 일반적인 다시 쓰기 방법을 사용하여 메시지 및 To: 주소를 처리합니다.

imsimta qm 명령에 대한 자세한 내용은 **Sun Java System Messaging Server 6.3 Administration Reference**의 "imsimta qm"을 참조하십시오.

13.5 변환 채널

conversion 채널을 사용하면 MTA를 통과하는 메시지 흐름에서 임의의 본문을 본문 부분 단위로 처리할 수 있습니다. 본문 부분은 메시지와 다릅니다. 예를 들어, 메일은 한 첨부 파일에 여러 본문 부분을 포함할 수 있습니다. 또한, 본문 부분은 MIME 헤더에서 지정하고 설명합니다.이 프로세스는 사이트에서 제공하는 프로그램이나 명령 프로시저에 의해 수행될 수 있으며 텍스트 또는 이미지를 다른 형식으로 변환, 바이러스 스캔, 언어 변환 등과 같은 작업을 수행할 수 있습니다. 변환할 다양한 메일 유형의 MTA 트래픽을 선택한 다음 각 유형의 메일 본문 부분에 대해 특정 프로세스 및 프로그램을 지정할 수 있습니다.

이 장을 학습하려면 채널 개념을 잘 알고 있어야 합니다(175 페이지 "8.5 채널" 참조). conversion 채널을 사용한 바이러스 스캔과 관련된 자세한 내용은 Messaging Server

설명서 웹 사이트(http://docs.sun.com/db/coll/S1_MsgTechNotes (http://docs.sun.com/db/coll/S1_MsgTechNotes))의 아래쪽에 있는 현재 버전의 Messaging Server Technical Notes를 참조하십시오.

변환 채널 구현은 A) 처리할 메일 트래픽 선택 및 B) 메일 처리 방법 지정으로 구성됩니다. 이러한 절차에 대해서는 나중에 자세히 설명합니다.

주 – 기본 변환 채널은 MTA 구성 파일(`imta.cnf`)에 자동으로 만들어집니다. 이 채널을 그대로 사용할 수 있으므로 수정할 필요가 없습니다.

이 절은 다음과 같은 하위 절로 구성되어 있습니다.

13.5.1 MIME 개요

변환 채널을 사용하면 MIME(Multipurpose Internet Mail Extensions) 헤더 행을 광범위하게 사용할 수 있습니다. 메일 구성 및 MIME 헤더 필드에 대한 지식이 필요합니다. MIME에 대한 자세한 내용은 RFC 1806, 2045 - 2049 및 2183 (http://www.faqs.org/rfcs/)을 참조하십시오. 편의를 위해 여기서는 MIME에 대한 간단한 개요를 제공합니다.

13.5.1.1 메일 구성

간단한 메일은 헤더와 본문으로 구성됩니다. 헤더는 메일의 맨 위쪽에 있으며 날짜, 제목, 보낸 사람, 수신자 등과 같은 특정 제어 정보가 포함되어 있습니다. 본문은 헤더 뒤의 첫 번째 빈 행 다음의 모든 내용입니다. MIME는 여러 본문 부분 및 본문 부분 내에 중첩된 본문 부분을 포함할 수 있는 보다 복잡한 메시지를 구성하는 방법을 지정합니다. 이러한 메시지를 다중 부분 메시지라고 합니다. 앞에서 설명한 것처럼 변환 채널은 메시지를 본문 부분 단위로 처리합니다.

13.5.1.2 MIME 헤더

MIME 규격은 본문 부분에 대한 헤더 행 집합을 정의합니다. 여기에는 `MIME-Version`, `Content-type`, `Content-Transfer-Encoding`, `Content-ID` 및 `Content-disposition`이 포함됩니다. 변환 채널은 `Content-type` 및 `Content-disposition` 헤더를 가장 많이 사용합니다. 다음은 일부 MIME 헤더 행의 예입니다.

```
Content-type: APPLICATION/wordperfect5.1;name=Poem.wpc
Content-transfer-encoding: BASE64
Content-disposition: attachment; filename=Poem.wpc
Content-description: "Project documentation Draft1 wordperfect format"
```

주 – MIME 헤더 행은 `To:`, `Subject:` 및 `From:` 등과 같은 일반적인 비 MIME 헤더 행과 다릅니다. 변환 채널의 경우 MIME 헤더 행은 기본적으로 Content-로 시작합니다.

Content-type 헤더

MIME `Content-Type` 헤더는 본문 부분의 내용을 설명합니다. `Content-Type` 헤더 형식의 예를 들면 다음과 같습니다.

`Content-type:` *type/ subtype*; *parameter1=value*; *parameter2=value...*

*type*은 본문 부분의 내용 유형을 설명합니다. 유형의 예로는 `Text`, `Multipart`, `Message`, `Application`, `Image`, `Audio`, `Video` 등이 있습니다.

*subtype*은 내용 유형을 자세히 설명합니다. 각 `Content-type`에는 자체 하위 유형 집합이 있습니다. 예를 들어,`text/plain`, `application/octet-stream`, `image/jpeg` 등이 있습니다. MIME 메일에 대한 내용 하위 유형은 IANA(Internet Assigned Numbers Authority)에 의해 지정되고 나열됩니다. 목록 복사본은 http://www.iana.org/assignments/media-types에 있습니다.

*parameter*는 Content-type/subtype 쌍에 한정됩니다. `charset` 및 `name` 매개 변수의 예를 들면 다음과 같습니다.

```
Content-type: text/plain; charset=us-ascii
Content-type: application/msword; name=temp.doc
```

`charset` 매개 변수는 텍스트 메일의 문자 세트를 지정합니다. `name` 매개 변수는 데이터를 파일로 작성할 경우 사용할 파일 이름을 제안합니다.

주 – `Content-Type` 값, `subtypes` 및 매개 변수 이름은 대/소문자를 구분합니다.

Content-disposition 헤더

MIME `Content-disposition` 헤더는 본문 부분에 대한 표시 정보를 제공합니다. 이 헤더를 첨부 파일에 추가하여 첨부 파일의 본문 부분을 표시할지(`inline`) 복사할 파일 이름으로 표시할지(`attachment`) 여부를 지정하는 경우도 있습니다. `Content-disposition` 헤더의 형식은 다음과 같습니다.

`Content-disposition:` *disposition_type; parameter1=value;parameter2=value...*

*disposition_type*은 일반적으로 `inline`(본문 부분 표시) 또는 `attachment`(저장할 파일로 표시)입니다. `Attachment`에는 일반적으로 저장된 파일에 대한 이름을 제안하는 값이 있는 `filename` 매개 변수가 있습니다.

Content-disposition 헤더에 대한 자세한 내용은 RFC2183을 참조하십시오.

13.5.2 변환 처리를 위한 트래픽 선택

다른 MTA 채널과 달리 변환 채널은 일반적으로 주소 또는 MTA 다시 쓰기 규칙에 지정되어 있지 않습니다. 대신 CONVERSIONS 매핑 테이블(imta_tailor 파일의 IMTA_MAPPING_FILE 매개 변수에 의해 지정됨)을 사용하여 변환 채널에 메시지를 보냅니다. 테이블 항목의 형식은 다음과 같습니다.

IN-CHAN=*source-channel* ;OUT-CHAN=*destination-channel*;CONVERT *Yes/No*

MTA는 각 메시지를 처리할 때 CONVERSIONS 매핑 테이블(있는 경우)을 검사합니다. *source-channel*이 메시지를 가져온 채널이고 *destination-channel*이 메시지를 받을 채널인 경우 CONVERT 앞의 작업이 수행됩니다. Yes일 경우 MTA는 *destination-channel*에서 가져온 메시지를 변환 채널로 전환합니다. 일치하는 항목이 발견되지 않는 경우 메시지가 일반 대상 채널의 대기열에 포함됩니다.

주 – CONVERSIONS 매핑 테이블에 관계 없이 user@conversion.*localhostname* 또는 user@conversion 형식의 주소가 변환 채널을 통해 라우팅됩니다.

다음 예에서는 모든 외부 메일(인터넷을 통해 주고 받는 메일)을 변환 채널로 라우팅합니다.

```
CONVERSIONS

   IN-CHAN=tcp_local;OUT-CHAN=*;CONVERT   Yes
   IN-CHAN=*;OUT-CHAN=tcp_local;CONVERT   Yes
```

첫 번째 행은 tcp_local 채널에서 가져온 메시지가 처리됨을 지정합니다. 두 번째 행은 tcp_local 채널로 보낸 메일도 처리됨을 지정합니다. tcp_local 채널은 인터넷을 통해 주고 받는 모든 메시지를 처리합니다. 기본값은 변환 채널을 통해 전달하지 않는 것이기 때문에 다른 메시지가 변환 채널을 통해 전달되지 않습니다.

이 테이블은 기본적인 테이블이므로 보다 많이 사용자 정의된 구성(예: 다중 outbound-to-the-Internet tcp_* 채널을 사용하거나 다중 inbound-from-the-Internet tcp_* 채널을 사용하는 사이트)이 있는 사이트에는 충분하지 않을 수 있습니다.

13.5.3 변환 처리 제어

이 절에서는 변환 처리를 제어하는 방법에 대해 설명합니다. 이 절은 다음 내용으로 구성되어 있습니다.

- 407 페이지 “13.5.3.1 변환 채널 정보 흐름”

- 407 페이지 “13.5.3.2 변환 채널 환경 변수 사용”
- 411 페이지 “13.5.3.3 변환 채널 출력 옵션 사용”
- 412 페이지 “13.5.3.4 포함된 MESSAGE/RFC822 부분의 헤더”
- 412 페이지 “13.5.3.5 변환 항목에서 매핑 테이블 호출”

메시지를 변환 채널로 보내면 본문 부분 단위로 처리됩니다. 처리는 `imta_tailor` 파일의 `IMTA_CONVERSION_FILE` 옵션에 지정된 MTA `conversions` 파일(기본값: *msg-svr-base*`/conversions`)에 의해 제어됩니다. `conversions` 파일은 1) 처리할 본문 부분의 유형을 지정하고 2) 처리 방법을 제어하는 항목으로 구성됩니다.

각 항목은 여러 `name=`*value* 매개 변수 절을 포함하는 하나 이상의 행으로 구성됩니다. 매개 변수 절의 값은 MIME 규칙을 따릅니다. 마지막 행을 제외한 모든 행은 세미콜론(`;`)으로 끝나야 합니다. 이 파일의 물리적 행은 252자로 제한됩니다. 백슬래시(\) 연결 문자를 사용하여 논리적 행을 여러 물리적 행으로 분할할 수 있습니다. 세미콜론으로 끝나지 않는 행이나 하나 이상의 빈 행 또는 두 가지 모두를 사용하여 항목을 종료합니다.

다음은 `conversion` 파일 항목의 간단한 예입니다.

예 13–1 `conversions` 파일 항목

```
out-chan=ims-ms; in-type=application; in-subtype=wordperfect5.1;
  out-type=application; out-subtype=msword; out-mode=block;
  command="/usr/bin/convert -in=wordp -out=msword 'INPUT_FILE' 'OUTPUT_FILE'"
```

`out-chan=ims-ms; in-type=application; in-subtype=wordperfect5.1` 절은 본문 부분을 규정합니다. 즉, 변환할 부분의 유형을 지정합니다. 각 부분의 헤더를 읽고 해당 헤더의 `Content-Type:`과 기타 헤더 정보를 추출합니다. 그런 다음 `conversion` 파일의 항목을 처음부터 끝까지 순서대로 검사하여 `in-*` 매개 변수가 있는지 확인하고, `OUT-CHAN` 매개 변수(있는 경우)를 검사합니다. 이러한 매개 변수가 모두 처리할 본문 부분의 해당 정보와 일치하면 `command=` 또는 `delete=` 절에 지정된 변환이 수행되고 `out-*` 매개 변수가 설정됩니다.

일치하지 않는 경우 해당 부분이 다음에 있는 `conversions` 파일 항목에 대해 일치됩니다. 모든 본문 부분을 스캔하여 처리한 다음(규정된 일치가 있다고 가정) 메시지를 다음 채널로 보냅니다. 일치하지 않는 경우 처리 작업을 수행하지 않고 메시지를 다음 채널로 보냅니다.

`out-chan=ims-ms`는 `ims-ms` 채널이 대상인 메일 부분만 변환하도록 지정합니다. `in-type=application` 및 `in-subtype=wordperfect5.1`은 메일 부분에 대한 MIME `Content-type` 헤더가 `application/wordperfect5.1`이 되도록 지정합니다.

추가 `in-*` 매개 변수를 사용하여 메일 부분을 자세히 규정할 수 있습니다. 표 13–6을 참조하십시오. 위 항목은 다음 MIME 헤더 행이 있는 메일 부분에 대한 변환 작업을 트리거합니다.

```
Content-type: APPLICATION/wordperfect5.1;name=Draft1.wpc
Content-transfer-encoding: BASE64
Content-disposition: attachment; filename=Draft1.wpc
Content-description: "Project documentation Draft1 wordperfect format"
```

예 13–1에서 세 conversion 파일 규정 매개 변수 뒤의 두 매개 변수 out-type=application 및 out-subtype=msword는 "처리된" 본문 부분에 첨부할 대체 MIME 헤더 행을 지정합니다. out-type=application 및 out-subtype=msword는 보내는 메일의 MIME Content-type/subtype이 application/msword가 되도록 지정합니다.

in-type 매개 변수와 out-type 매개 변수가 동일하고 보내는 본문 부분의 원본 MIME 레이블이 변환 채널의 기본값이기 때문에 out-type=application은 필요하지 않습니다. 추가 출력 매개 변수를 사용하여 보내는 본문 부분에 대한 추가 MIME 레이블을 지정할 수 있습니다.

out-mode=block(예 13–1)은 사이트에서 제공하는 프로그램이 반환할 파일 유형을 지정합니다. 즉, 파일 저장 방법과 반환된 파일에서 변환 채널을 다시 읽는 방법을 지정합니다. 예를 들어, html 파일은 텍스트 모드로 저장되고 .exe 프로그램 또는 zip 파일은 블록/이진 모드로 저장됩니다. 모드는 읽을 파일이 특정 저장소 형식에 속하는지를 설명하는 방법입니다.

예 13–1의 마지막 매개 변수는 본문 부분에 대해 수행할 작업을 다음과 같이 지정합니다.

```
command="/usr/bin/convert -in=wordp -out=msword 'INPUT_FILE' 'OUTPUT_FILE'"
```

command= 매개 변수는 프로그램이 본문 부분에서 실행됨을 지정합니다. /usr/bin/convert는 가상 명령 이름이고 -in=wordp 및 -out=msword는 입력 텍스트 및 출력 텍스트의 형식을 지정하는 가상 명령줄 인수입니다. INPUT_FILE 및 OUTPUT_FILE은 원래의 본문 부분이 포함된 파일과 프로그램에서 변환된 본문 부분을 저장하는 파일을 지정하는 변환 채널 환경 매개 변수(407 페이지 "13.5.3.2 변환 채널 환경 변수 사용" 참조)입니다.

주 – 일반 변환 항목에 의해 외부 메일 헤더를 포함하는 파일이 요청되는 경우 봉투 생성자 및 수신자 정보는 x-envelope-from 및 x-envelope-to 필드로 제공됩니다.

본문 부분에서 명령을 실행하지 않고 command 매개 변수를 DELETE=1로 대체하면 메일 부분을 간단하게 삭제할 수 있습니다.

주 – conversions 파일을 수정할 때마다 구성을 다시 컴파일해야 합니다(209 페이지 "10.1 MTA 구성 컴파일" 참조).

13.5.3.1 변환 채널 정보 흐름

정보 흐름은 다음과 같습니다. 본문 부분이 포함된 메시지가 변환 채널로 전달됩니다. 변환 채널에서 메시지를 구문 분석하여 한 부분씩 처리합니다. 그런 다음 본문 부분을 규정합니다. 즉, MIME 헤더 행을 **규정 매개 변수**와 비교하여 메시지를 처리할지 여부를 결정합니다. 본문 부분이 규정되면 변환 처리가 시작됩니다. MIME 또는 본문 부분 정보를 변환 스크립트로 전달해야 할 경우 해당 정보가 **정보 전달 매개 변수**에 지정된 환경 변수(407 페이지 "13.5.3.2 변환 채널 환경 변수 사용")로 저장됩니다.

이 지점에서 **작업 매개 변수**에 지정된 작업이 본문 부분에 대해 수행됩니다. 일반적으로 본문 부분을 삭제하거나 스크립트에 포함된 프로그램으로 전달하는 작업입니다. 스크립트는 본문 부분을 처리한 다음 사후 처리 메일로 재어셈블할 수 있도록 변환 채널로 다시 보냅니다. 또한, 변환 채널 **출력 옵션**을 사용하여 정보를 변환 채널로 보낼 수 있습니다. 이 정보는 출력 본문 부분에 추가할 새 MIME 헤더 행, 메시지를 보낸 사람에게 반환할 오류 텍스트, 작업(메일 바운스, 삭제, 보관 등)을 시작하도록 MTA에 지시하는 특수 지시문 등입니다.

마지막으로 변환 채널은 출력 본문 부분의 헤더 행을 **출력 매개 변수**에 지정된 대로 바꿉니다.

13.5.3.2 변환 채널 환경 변수 사용

메일 본문 부분에 대한 작업을 수행할 경우 사이트에서 제공하는 프로그램을 통해 MIME 헤더 행 정보 또는 전체 본문 부분을 전달하는 것이 좋습니다. 예를 들어, 프로그램에는 메일 본문 부분 외에도 `Content-type` 및 `Content-disposition` 헤더 행 정보가 필요할 수 있습니다. 일반적으로 사이트에서 제공하는 프로그램의 기본 입력은 파일에서 읽은 메일 본문 부분입니다. 본문 부분을 처리한 후 프로그램은 변환 채널에서 읽을 수 있도록 본문 부분을 파일에 기록해야 합니다. 이러한 유형의 정보 전달은 변환 채널 환경 변수를 사용하여 수행됩니다.

`parameter-symbol-*` 매개 변수 또는 미리 정의된 채널 환경 변수(411 페이지 "13.5.3.3 변환 채널 출력 옵션 사용" 참조)를 사용하여 `conversions` 파일에 환경 변수를 만들 수 있습니다.

다음 `conversions` 파일 항목 및 수신 헤더는 환경 변수를 사용하여 MIME 정보를 사이트에서 제공하는 프로그램에 전달하는 방법을 나타냅니다.

`conversions` 파일 항목:

```
in-channel=*; in-type=application; in-subtype=*;
  parameter-symbol-0=NAME; parameter-copy-0=*;
  dparameter-symbol-0=FILENAME; dparameter-copy-0=*;
  message-header-file=2; original-header-file=1;
  override-header-file=1; override-option-file=1;
  command="/bin/viro-scan500.sh "INPUT_FILE' "OUTPUT_FILE'"
```

수신 헤더:

```
Content-type: APPLICATION/msword; name=Draft1.doc
Content-transfer-encoding: BASE64
Content-disposition: attachment; filename=Draft1.doc
Content-description: "Project documentation Draft1 msword format"
```

in-channel=*; in-type=application; in-subtype=*는 유형이 application인 입력 채널의 메일 본문 부분을 처리하도록 지정합니다.

parameter-symbol-0=NAME은 첫 번째 Content-type 매개 변수 값(이 예의 경우 Draft1.doc)을 NAME이라는 환경 변수에 저장하도록 지정합니다.

parameter-copy-0=*는 입력 본문 부분의 모든 Content-type 매개 변수를 출력 본문 부분에 복사하도록 지정합니다.

dparameter-symbol-0=FILENAME은 첫 번째 Content-disposition 매개 변수 값(이 예의 경우 Draft1.doc)을 FILENAME이라는 환경 변수에 저장하도록 지정합니다.

dparameter-copy-0=*는 입력 본문 부분의 모든 Content-disposition 매개 변수를 출력 본문 부분에 복사하도록 지정합니다.

message-header-file=2는 메일의 원본 헤더(가장 외부에 있는 메일 헤더) 전체를 MESSAGE_HEADERS 환경 변수에 지정된 파일에 기록하도록 지정합니다.

original-header-file=1은 포함된 MESSAGE/RFC822 부분의 원본 헤더를 ORIGINAL_HEADERS 환경 변수에 기록하도록 지정합니다.

override-header-file=1은 포함된 MIME 부분의 원본 MIME 헤더 행을 무시하고 OUTPUT_HEADERS 환경 변수에 지정된 파일에서 MIME 헤더를 읽도록 지정합니다. $OUTPUT_HEADERS는 변환을 실행할 때 즉석에서 만들어지는 임시 파일입니다. 사이트에서 제공하는 프로그램은 이 파일을 사용하여 변환 프로세스 중에 변경된 MIME 헤더 행을 저장합니다. 그런 다음 변환 채널은 본문 부분을 재어셈블할 때 이 파일에서 MIME 헤더 행을 읽습니다. MIME 헤더 행만 수정할 수 있습니다. 다른 일반적인 비 MIME 헤더 행은 변환 채널에서 변경할 수 없습니다.

override-option-file=1은 변환 채널이 OUTPUT_OPTIONS 환경 변수를 통해 명명된 파일에서 **변환 채널 옵션**을 읽도록 지정합니다. 411 페이지 "13.5.3.3 변환 채널 출력 옵션 사용"을 참조하십시오.

command="msg-svr-base/bin/viro-scan500.sh"는 메일 본문 부분에서 실행할 명령을 지정합니다.

표 13–3 변환 채널 환경 변수

환경 변수	설명
ATTACHMENT_NUMBER	현재 부분의 첨부 파일 수입니다. ATTACHMENT-NUMBER 변환 일치 매개 변수와 같은 형식을 사용합니다.

표 13–3 변환 채널 환경 변수 (계속)

환경 변수	설명
CONVERSION_TAG	활성 변환 태그의 현재 목록입니다. TAG 변환 일치 매개 변수에 해당합니다.
INPUT_CHANNEL	메시지를 변환 채널의 대기열에 포함시킨 채널입니다. IN-CHANNEL 변환 일치 매개 변수에 해당합니다.
INPUT_ENCODING	본문 부분에 원래 표시된 부분을 인코딩합니다.
INPUT_FILE	원본 본문 부분이 포함된 파일 이름입니다. 사이트에서 제공하는 프로그램은 이 파일을 읽어야 합니다.
INPUT_HEADERS	본문 부분의 원본 헤더 행이 포함된 파일 이름입니다. 사이트에서 제공하는 프로그램은 이 파일을 읽어야 합니다.
INPUT_TYPE	입력 메일 부분의 MIME Content-type입니다.
INPUT_SUBTYPE	입력 메일 부분의 MIME 내용 하위 유형입니다.
INPUT_DESCRIPTION	입력 메일 부분의 MIME content-description입니다.
INPUT_DISPOSITION	입력 메일 부분의 MIME content-disposition입니다.
MESSAGE_HEADERS	포함된 메일의 가장 외부에 있는 원본 헤더(본문 부분 아님) 또는 최신 포함된 MESSAGE/RFC822 부분의 헤더를 포함하는 파일 이름입니다. 사이트에서 제공하는 프로그램은 이 파일을 읽어야 합니다.
OUTPUT_CHANNEL	메시지를 보내는 채널입니다. OUT-CHANNEL 변환 일치 매개 변수에 해당합니다.
OUTPUT_FILE	사이트에서 제공하는 프로그램이 출력을 저장하는 파일 이름입니다. 사이트에서 제공하는 프로그램은 이 파일을 만들고 써야 합니다.
OUTPUT_HEADERS	사이트에서 제공하는 프로그램이 포함 부분에 대한 MIME 헤더 행을 저장하는 파일의 이름입니다. 사이트에서 제공하는 프로그램은 이 파일을 만들고 써야 합니다. 파일에는 마지막 행으로 빈 행이 오는 실제 MIME 헤더 행(option=value 행 아님)이 포함되어 있어야 합니다. 또한, MIME 헤더 행만 수정할 수 있습니다. 다른 일반적인 비 MIME 헤더 행은 변환 채널에서 변경할 수 없습니다.
OUTPUT_OPTIONS	사이트에서 제공하는 프로그램이 변환 채널 옵션을 읽어야 하는 파일의 이름입니다. 411 페이지 “13.5.3.3 변환 채널 출력 옵션 사용”을 참조하십시오.
PART_NUMBER	현재 부분의 부품 번호입니다. PART-NUMBER 변환 일치 매개 변수와 같은 형식을 사용합니다.
PART_SIZE	처리할 부분의 크기(바이트)입니다.

메일 변환 태그

메일 변환 태그는 특정한 받는 사람이나 보낸 사람과 연관된 특수한 태그입니다. 메시지가 전달될 때 변환 채널 프로그램에서 변환 태그를 볼 수 있으며 특수한 처리를 위해 변환 태그를 사용할 수 있습니다. 변환 태그는 LDAP 디렉토리에 저장됩니다.

메일 변환 태그를 다음과 같이 사용할 수 있습니다. 관리자는 메일 변환 태그 값이 harmonica인 선택된 사용자를 설정할 수 있습니다. 그런 다음 관리자는 해당 메일 처리 시에 태그와 harmonica 값을 감지하는 변환 채널을 설정합니다. 해당 태그와 값이 감지되면 프로그램에서 몇 가지 임의의 기능을 수행합니다.

메일 변환 태그는 사용자 또는 도메인 단위로 설정할 수 있습니다. 도메인 수준의 수신자 LDAP 속성은 MailDomainConversionTag(MTA 옵션 LDAP_DOMAIN_ATTR_CONVERSION_TAG를 사용하여 수정할 수 있음)입니다. 사용자 수준의 수신자 LDAP 속성은 MailConversionTag(MTA 옵션 LDAP_CONVERSION_TAG를 사용하여 수정할 수 있음)입니다. 이러한 두 속성은 값이 여러 개일 수 있으며 각 값에 다른 태그를 지정할 수 있습니다. 특정 수신자와 연관된 태그 집합은 누적됩니다. 즉, 도메인 수준에서 설정된 태그가 사용자 수준에서 설정된 태그와 결합됩니다.

보낸 사람 기반의 변환 태그는 MTA 옵션 LDAP_SOURCE_CONVERSION_TAG 및 LDAP_DOMAIN_ATTR_SOURCE_CONVERSION_TAG를 사용하여 설정할 수 있습니다. 이 옵션은 이러한 소스 주소와 연관된 변환 태그에 대해 각각 사용자 및 도메인 수준 LDAP 속성을 지정합니다. 이러한 옵션에는 기본 속성이 없습니다.

시스템 시브(Sieve)에서 addconversiontag와 setconversiontag라는 새로운 두 작업을 사용할 수 있습니다. 둘 다 단일 인수, 즉 문자열이나 변환 태그의 목록을 허용합니다. addconversiontag는 현재 태그 목록에 변환 태그를 추가하며, setconversiontag는 새 태그를 추가하기 전에 기존 목록을 비웁니다. setconversiontag를 사용하여 나머지 모든 변환 태그 설정 기법을 취소할 수 있도록 이러한 작업은 매우 늦게 수행됩니다. 따라서 시브(Sieve) 필터에 변환 태그를 포함시킬 수 있습니다.

시브(Sieve) 봉투 테스트에서는 봉투 필드 지정자 값으로 conversiontag를 허용합니다. 이 테스트에서는 현재의 태그 목록을 한번에 하나씩 검사합니다. :count 수정자가 지정된 경우 활성 변환 태그의 수를 확인할 수 있습니다. 이 봉투 테스트 유형은 시스템 시브(Sieve)로 제한됩니다. 또한 이 테스트에서는 시브(Sieve) 처리 이전에 존재했던 태그 세트만 확인합니다. setconversiontag 및 addconversiontag 작업의 결과는 표시되지 않습니다.

다양한 매핑 검사에 변환 태그 정보 포함시키기

새로운 MTP 옵션인 INCLUDE_CONVERSIONTAG가 추가되어 다양한 매핑 검사에 변환 태그 정보를 포함시키는 것을 선택적으로 활성화할 수 있습니다. 이는 비트 인코딩 값입니다. 아래 표와 같이 비트가 할당됩니다. 항상 현재 태그 집합이 쉼표로 구분된 목록으로 검사에 나타납니다.

위치	값	매핑
0	1	CHARSET_CONVERSION - ;CONVERT 앞에 ;TAG= 필드로 추가됩니다.
1	2	CONVERSION - ;CONVERT 앞에 ;TAG= 필드로 추가됩니다.

위치	값	매핑
2	4	FORWARD - 현재 주소의 바로 앞에 추가됩니다(| delim).
3	8	ORIG_SEND_ACCESS - 검사의 끝에 추가됩니다(| delim).
4	16	SEND_ACCESS - 검사의 끝에 추가됩니다(| delim).
5	32	ORIG_MAIL_ACCESS - 검사의 끝에 추가됩니다(| delim).
6	64	MAIL_ACCESS - 검사의 끝에 추가됩니다(| delim).

13.5.3.3 변환 채널 출력 옵션 사용

변환 채널 출력 옵션(표 13–4)은 변환 스크립트에서 변환 채널로 정보 및 특수 지시문을 전달하는 데 사용되는 동적 변수입니다. 예를 들어, 본문 부분을 처리하는 동안 스크립트가 메시지를 바운스하고 메일에 바이러스가 있다는 오류 텍스트를 반환되는 메일에 추가하도록 요청하는 특수 지시문을 보낼 수 있습니다.

원하는 변환 항목에 OVERRIDE-OPTION-FILE=1을 설정하여 출력 옵션을 시작합니다. 그러면 출력 옵션이 필요에 따라 스크립트에서 설정되고 환경 변수 파일 OUTPUT_OPTIONS에 저장됩니다. 스크립트가 본문 부분 처리를 완료하면 변환 채널이 OUTPUT_OPTIONS 파일에서 옵션을 읽습니다.

OUTPUT_OPTION 변수는 변환 채널이 옵션을 읽는 파일의 이름입니다. 일반적으로 이 파일을 즉석에서 만들어지는 임시 파일로 사용하여 정보를 전달합니다. 아래 예는 출력 옵션을 사용하여 바이러스 메일을 보낸 사람에게 오류 메시지를 반환하는 스크립트입니다.

```
/usr/local/bin/viro_screen2k $INPUT_FILE   # run the virus screener

if [ $? -eq 1 ]; then
   echo "OUTPUT_DIAGNOSTIC='Virus found and deleted.'" > $OUTPUT_OPTIONS
   echo "STATUS=178029946" >> $OUTPUT_OPTIONS
else
   cp $INPUT_FILE $OUTPUT_FILE # Message part is OK
fi
```

이 예에서는 시스템 진단 메시지와 상태 코드를 $OUTPUT_OPTIONS에 정의된 파일에 추가합니다. $OUTPUT_OPTIONS 임시 파일에는 다음과 비슷한 내용이 표시됩니다.

```
OUTPUT_DIAGNOSTIC="Virus found and deleted."
STATUS=178029946
```

OUTPUT_DIAGNOSTIC='Virus found and deleted' 행은 메시지에 Virus found and deleted 텍스트를 추가하도록 변환 채널에 지시합니다.

178029946은 *msg-svr-base*/include/deprecated/pmdf_err.h에 있는 각 pmdf_err.h 파일의 PMDF__FORCERETURN 상태입니다. 이 상태 코드는 메시지를 보낸 사람에게 다시 바운스하도록 변환 채널에 지시합니다. 특수 지시문 사용에 대한 자세한 내용은 414 페이지 "13.5.4 변환 채널 출력을 사용하여 메시지 바운스, 삭제, 보관 또는 재시도"를 참조하십시오

다음은 전체 출력 옵션 목록입니다.

표 13–4 변환 채널 출력 옵션

옵션	설명
OUTPUT_TYPE	출력 메시지 부분의 MIME 내용 유형입니다.
OUTPUT_SUBTYPE	출력 메시지 부분의 MIME 내용 하위 유형입니다.
OUTPUT_DESCRIPTION	출력 메시지 부분의 MIME 내용 설명입니다.
OUTPUT_DIAGNOSTIC	메시지가 변환 채널에 의해 강제로 바운스될 경우 보낸 사람에게 전달되는 메시지의 일부로 포함되는 텍스트입니다.
OUTPUT_DISPOSITION	출력 메시지 부분의 MIME content-disposition입니다.
OUTPUT_ENCODING	출력 메시지 부분에 사용할 MIME 내용 전송 encoding입니다.
OUTPUT_MODE	변환 채널에서 출력 메시지 부분을 쓸 때 사용하는 MIME Mode 즉, 수신자가 출력 메시지 부분을 읽을 때 사용하는 모드입니다.
STATUS	변환기의 종료 상태입니다. 일반적으로 변환 채널에서 일부 작업을 시작하는 특수 지시문입니다. 전체 지시문 목록은 *msg_svr_base*/include/deprecated/pmdf_err.h를 참조하십시오.

13.5.3.4 포함된 MESSAGE/RFC822 부분의 헤더

메시지 부분에서 변환을 수행할 때 변환 채널은 포함된 MESSAGE/RFC822 부분의 헤더에 액세스합니다. 포함된 MESSAGE/RFC822 부분이 없는 경우 메시지 헤더에 액세스합니다. 헤더의 정보는 사이트에서 제공하는 프로그램에 유용할 수 있습니다.

ORIGINAL-HEADER-FILE=1이 있는 항목을 선택하면 포함된 MESSAGE/RFC822 부분의 모든 원본 헤더 행이 ORIGINAL_HEADERS 환경 변수에 표시된 파일에 기록됩니다. OVERRIDE-HEADER-FILE=1인 경우 변환 채널은 ORIGINAL_HEADERS 환경 변수에 표시된 파일의 내용을 읽어 해당 포함 부분의 헤더로 사용합니다.

13.5.3.5 변환 항목에서 매핑 테이블 호출

out-parameter-* 값은 이름이 중복 지정된 매핑 테이블에서 저장 및 검색될 수 있습니다. 이 기능은 첨부 파일이 postscript, msword, text 중 어느 것인지에 관계 없이 att.dat와 같은 일반적인 이름으로 모든 첨부 파일을 보내는 클라이언트가 첨부 파일의 이름을 변경하여 보낼 때 유용합니다. 이 방법은 다른 클라이언트(예: Outlook)가 확장명을 읽어 해당 부분을 열 수 있도록 이 부분의 레이블을 다시 지정하는 일반적인 방법입니다.

매핑 테이블에서 매개 변수 값을 검색하는 구문은 다음과 같습니다.

```
"mapping-table-name:mapping-input[$Y, $N]'
```

$Y는 매개 변수 값을 반환합니다. 일치하는 내용이 없거나 $N이 반환될 경우 변환 파일 항목의 해당 매개 변수가 무시되거나 빈 문자열로 처리됩니다. 일치 항목이 부족하거나 $N이 반환될 경우에는 변환 항목이 중지되지 않습니다.

다음 매핑 테이블을 참조하십시오.

```
X-ATT-NAMES

   postscript       temp.PS$Y
   wordperfect5.1   temp.WPC$Y
   msword           temp.DOC$Y
```

위 매핑 테이블에서 첨부 파일에 있는 특정 파일 이름을 일반 파일 이름으로 대체하면 변환 항목이 다음과 같이 표시됩니다.

```
out-chan=tcp_local; in-type=application; in-subtype=*;
   in-parameter-name-0=name; in-parameter-value-0=*;
   out-type=application; out-subtype='INPUT-SUBTYPE';
   out-parameter-name-0=name;
   out-parameter-value-0="'X-ATT-NAMES:\\'INPUT_SUBTYPE\\''";
   command="cp  "INPUT_FILE'  "OUTPUT_FILE'"
```

위의 예에서 `out-chan=tcp_local; in-type=application; in-subtype=*`는 `application/*`(*는 하위 유형의 수행 작업을 지정함)의 `content-type` 헤더를 사용하여 `tcp_local` 채널에서 처리할 메시지를 가져오도록 지정합니다.

`in-parameter-name-0=name; in-parameter-value-0=*`는 메시지에 `name=*` 매개 변수 유형(*는 매개 변수 값의 수행 작업을 지정함)이 있어야 함을 지정합니다)

`out-type=application;`은 사후 처리 메시지의 MIME `Content-type` 매개 변수가 `application`임을 지정합니다.

`out-subtype='INPUT-SUBTYPE';`은 사후 처리 본문 부분의 MIME MIME `subtype` 매개 변수가 입력 `subtype`의 원본 값인 `INPUT-SUBTYPE` 환경 변수가 되도록 지정합니다. 따라서

```
Content-type: application/xxxx; name=foo.doc
```

위 항목을 아래와 같이 변경하려면

```
Content-type: application/msword; name=foo.doc
```

다음을 사용합니다.

```
out-type=application; out-subtype=msword
```

`out-parameter-name-0=name;`은 출력 본문 부분의 첫 번째 MIME Content-type 매개 변수가 `name=` 유형임을 지정합니다.

out-parameter-value-0='X-ATT-NAMES:\\'INPUT_SUBTYPE\\'';은 첫 번째 MIME subtype 매개 변수 값을 사용하여 X-ATT-NAMES 매핑 테이블에서 subtype 일치 항목을 검색하도록 지시합니다. 일치 항목이 발견되면 name 매개 변수는 X-ATT-NAMES 매핑 테이블에 지정된 새 값을 받습니다. 따라서, 매개 변수가 msword 유형인 경우 name 매개 변수는 temp.DOC가 됩니다.

13.5.4 변환 채널 출력을 사용하여 메시지 바운스, 삭제, 보관 또는 재시도

이 절에서는 변환 채널 옵션을 사용하여 메시지를 바운스, 삭제 또는 보관하는 방법을 설명합니다. 기본 절차는 다음과 같습니다.

1. 해당 변환 파일 항목에서 OVERRIDE-OPTION-FILE=1을 설정합니다. 이 설정은 OUTPUT_OPTIONS 파일에서 출력 옵션을 읽도록 변환 채널에 지시합니다.
2. 변환 스크립트를 사용하여 특정 메시지 본문 부분에 필요한 작업을 결정합니다.
3. 스크립트에서 OUTPUT_OPTIONS 파일에 STATUS=*directive_code* 옵션을 작성하여 해당 작업에 대한 특수 지시문을 지정합니다.

전체 특수 지시문 목록은 *msg-svr-base*/include/deprecated/pmdf_err.h를 참조하십시오. 변환 채널에 일반적으로 사용되는 특수 지시문은 다음과 같습니다.

표 13–5 변환 채널에 일반적으로 사용되는 특수 지시문

이름	16진수 값	10진수 값
PMDF__FORCEHOLD	0x0A9C86AA	178030250
PMDF__FORCERETURN	0x0A9C857A	178029946
PMDF__FORCEDELETE	0x0A9C8662	178030178
PMDF__FORCEDISCARD	0x0A9C86B3	178030259
PMDF__AGN	0x0A9C809A	178028698

이러한 지시문의 기능의 예는 다음과 같습니다.

13.5.4.1 메시지를 바운스하는 방법

변환 채널을 사용하여 메시지를 바운스하려면 해당 conversions 파일 항목에서 OVERRIDE-OPTION-FILE=1을 설정하고 변환 스크립트에 다음 행을 추가합니다.

```
echo "STATUS=178029946" >> $OUTPUT_OPTIONS
```

바운스된 메시지에 간단한 텍스트 문자열을 추가하려면 변환 스크립트에 다음 행을 추가합니다.

```
echo OUTPUT_DIAGNOSTIC=text-string >> $OUTPUT_OPTIONS
```

여기서 텍스트 문자열은 “The message sent from your machine contained a virus which has been removed. Be careful about executing email attachments.”과 같습니다.

13.5.4.2 메시지 전부 또는 부분 조건부 삭제

메시지 부분을 포함된 내용에 따라 조건적으로 삭제하는 것이 좋을 수 있습니다. 조건적 삭제는 출력 옵션을 사용하여 수행할 수 있습니다. 반대로 DELETE=1 변환 매개 변수 절은 메시지 부분을 무조건 삭제합니다.

출력 옵션을 사용하여 메시지 부분을 삭제하려면 해당 변환 파일 항목에 OVERRIDE-OPTION-FILE=1을 설정하고 변환 스크립트에 다음 행을 추가합니다.

```
echo "STATUS=178030178" >> $OUTPUT_OPTIONS
```

마찬가지로 메시지 전체를 삭제하려면 다음을 사용할 수 있습니다.

```
echo "STATUS=178030259" >> $OUTPUT_OPTIONS
```

13.5.4.3 메시지를 보관하는 방법

메시지 부분을 포함된 내용에 따라 조건적으로 보관하는 것이 좋을 수 있습니다. 출력 옵션을 사용하여 메시지 부분을 삭제하려면 해당 변환 파일 항목에 OVERRIDE-OPTION-FILE=1을 설정하고 변환 스크립트에 다음 행을 추가합니다.

```
echo "STATUS=178030250" >> $OUTPUT_OPTIONS
```

이 행은 변환 채널이 메시지를 변환 채널 대기열에 .HELD 파일로 보관해야 하도록 요청합니다.

13.5.4.4 메시지를 다시 처리하는 방법

변환기 스크립트에 임시 자원 문제가 발생하면(시스템에서 외부 서버에 연결할 수 없거나, 필요한 파일이 잠겨 있는 등) PMDF_AGN을 사용하여 변환 채널에서 임시 오류가 발생한 메시지의 처리를 고려하게 만들 수 있습니다. MTA는 mail.log_current에 "Q" 상태 메시지를 기록하고 메시지를 변환 채널에 보관한 다음 나중에 다시 처리를 시도합니다.

변환 스크립트에 다음 행을 추가합니다.

```
echo "STATUS=178028698" >> $OUTPUT_OPTIONS
```

13.5.5 변환 채널 예

아래 예에 표시된 CONVERSIONS 매핑 및 변환 규칙 세트는 tcp_docuprint 가상 채널로 보낸 GIF, JPEG 및 BITMAP 파일을 PostScript로 자동으로 변환하게 합니다. 이러한 여러 변환에서는 가상 /usr/bin/ps-converter.sh를 사용하여 해당 변환을 수행합니다. WordPerfect 5.1 파일을 Microsoft Word 파일로 변환하는 추가 규칙이 포함되어 있습니다.

```
CONVERSIONS

    IN-CHAN=*;OUT-CHAN=tcp_docuprint;CONVERT    Yes

out-chan=ims-ms; in-type=application; in-subtype=wordperfect5.1;
  out-type=application; out-subtype=msword; out-mode=block;
  command="/bin/doc-convert -in=wp -out=msw   'INPUT_FILE'  'OUTPUT_FILE'"

out-chan=tcp_docuprint; in-type=image; in-subtype=gif;
  out-type=application; out-subtype=postscript; out-mode=text;
  command="/bin/ps-convert -in=gif -out=ps   'INPUT_FILE'  'OUTPUT_FILE'"

out-chan=tcp_docuprint; in-type=image; in-subtype=jpeg;
  out-type=application; out-subtype=postscript; out-mode=text;
  command="/bin/ps-convert -in=jpeg -out=ps  'INPUT_FILE'  'OUTPUT_FILE'"

out-chan=tcp_docuprint; in-type=image; in-subtype=bitmap;
  out-type=application; out-subtype=postscript; out-mode=text;
  command="/bin/ps-convert -in=bmp -out=ps   'INPUT_FILE'  'OUTPUT_FILE'"
```

변환 매개 변수는 다음과 같습니다.

표 13–6 변환 매개 변수

매개 변수	설명
매개 변수 규정(변환하기 전에 메시지가 일치해야 하는 매개 변수 지정)	
OUT-CHAN,OUT-CHANNEL	변환을 위해 일치시킬 출력 채널(와일드카드 허용)입니다. 메시지가 해당 채널의 대상으로 지정된 경우에만 이 항목에 지정된 변환이 수행됩니다.
IN-CHAN,IN-CHANNEL	변환을 위해 일치시킬 입력 채널(와일드카드 허용)입니다. 지정된 채널에서 메시지가 전달된 경우에만 이 항목에 지정된 변환이 수행됩니다.
IN-TYPE	변환을 위해 일치시킬 입력 MIME 유형(와일드카드 허용)입니다. 이 필드가 본문 부분의 MIME 유형과 일치하는 경우에만 지정된 변환이 수행됩니다.
IN-SUBTYPE	변환을 위해 일치시킬 입력 MIME 하위 유형(와일드카드 허용)입니다. 이 필드가 본문 부분의 MIME 하위 유형과 일치하는 경우에만 이 항목에 지정된 변환이 수행됩니다.

표 13–6 변환 매개 변수 *(계속)*

매개 변수	설명
IN-PARAMETER-NAME-*n*	변환을 위해 일치해야 하는 입력 MIME Content-Type 매개 변수 이름입니다(*n* = 0, 1, 2...). 이 매개 변수를 IN-PARAMETER-VALUE-*n*과 함께 사용하여 이름 및 보유한 값을 기준으로 매개 변수를 명확하게 식별할 수 있습니다.
IN-PARAMETER-VALUE-*n*	변환을 위해 일치해야 하는 해당 IN-PARAMETER-NAME의 입력 MIME Content-Type 매개 변수 값입니다. 이 필드가 본문 부분의 Content-Type 매개 변수 목록에 있는 해당 매개 변수와 일치하는 경우에만 이 항목에 지정된 변환이 수행됩니다. 와일드카드가 허용됩니다.
IN-PARAMETER-DEFAULT-*n*	매개 변수가 없는 경우 입력 MIME Content-Type 매개 변수의 기본값입니다. 이 값은 해당 매개 변수가 본문 부분에 지정되어 있지 않은 경우 IN-PARAMETER-VALUE-*n* 테스트를 위한 기본값으로 사용됩니다.
IN-DISPOSITION	변환을 위해 일치시킬 입력 MIME Content-Disposition입니다.
IN-DPARAMETER-NAME-*n*	변환을 위해 일치해야 하는 입력 MIME Content-Disposition 매개 변수 이름입니다(*n* = 0, 1, 2...). 이 매개 변수를 IN-DPARAMETER-VALUE-*n*과 함께 사용하여 이름 및 보유한 값을 기준으로 매개 변수를 명확하게 식별할 수 있습니다.
IN-DPARAMETER-VALUE-*n*	변환을 위해 일치해야 하는 해당 IN-DPARAMETER-NAME의 입력 MIME Content-Disposition 매개 변수 값입니다. 이 필드가 본문 부분의 Content-Disposition: 매개 변수 목록에 있는 해당 매개 변수와 일치하는 경우에만 이 항목에 지정된 변환이 수행됩니다. 와일드카드가 허용됩니다.
IN-DPARAMETER-DEFAULT-*n*	매개 변수가 없는 경우의 입력 MIME Content-Disposition 매개 변수의 기본값입니다. 이 값은 해당 매개 변수가 본문 부분에 지정되어 있지 않은 경우 IN-DPARAMETER-VALUE-*n* 테스트를 위한 기본값으로 사용됩니다.
IN-DESCRIPTION	변환을 위해 일치시킬 입력 MIME Content-Description입니다.
IN-SUBJECT	포함 MESSAGE/RFC822 부분에서 가져온 입력 Subject입니다.
TAG	메일 목록 CONVERSION_TAG 매개 변수에 의해 설정되는 입력 태그입니다.
출력 매개 변수(본문 부분의 변환 후 출력 설정 지정)	
OUT-TYPE	입력 유형과 다를 경우의 출력 MIME 유형입니다.
OUT-SUBTYPE	입력 하위 유형과 다를 경우의 출력 MIME 하위 유형입니다.
OUT-PARAMETER-NAME-*n*	출력 MIME Content-Type 매개 변수 이름입니다(*n* = 0, 1, 2...).
OUT-PARAMETER-VALUE-*n*	OUT-PARAMETER-NAME-*n*에 해당하는 출력 MIME Content-Type 매개 변수 값입니다.
PARAMETER-COPY-*n*	입력 본문 부분의 Content-Type 매개 변수 목록에서 출력 본문 부분의 Content-Type: 매개 변수 목록에 복사할 Content-Type 매개 변수 목록입니다(*n*=0, 1, 2...). IN-PARAMETER-NAME-*n* 절과 일치할 경우 동일한 MIME 매개 변수 이름을 사용하여 복사합니다.
OUT-DISPOSITION	입력 MIME Content-Disposition과 다를 경우의 출력 MIME Content-Disposition입니다.

표 13–6 변환 매개 변수 *(계속)*

매개 변수	설명
OUT-DPARAMETER-NAME-*n*	출력 MIME Content-Disposition 매개 변수 이름입니다(*n*=0, 1, 2...).
OUT-DPARAMETER-VALUE-*n*	OUT-DPARAMETER-NAME-*n*에 해당하는 출력 MIME Content-Disposition 매개 변수 값입니다.
DPARAMETER-COPY-*n*	입력 본문 부분의 Content-Disposition: 매개 변수 목록에서 출력 본문 부분의 Content-Disposition: 매개 변수 목록에 복사할 Content-Disposition: 매개 변수 목록입니다(*n* = 0, 1, 2,...). IN-PARAMETER-NAME-*n* 절과 일치할 경우 MIME 매개 변수 이름을 인수로 사용하여 복사합니다. 와일드카드를 인수로 사용할 수 있습니다. 특히 * 인수는 모든 원본 Content-Disposition: 매개 변수를 복사함을 의미합니다.
OUT-DESCRIPTION	입력 MIME Content-Description과 다를 경우의 출력 MIME Content-Description입니다.
OUT-MODE	변환된 파일을 읽고 저장하는 데 사용하는 모드입니다. BLOCK(이진 및 실행 파일) 또는 TEXT여야 합니다.
OUT-ENCODING	메시지를 재어셈블할 때 변환된 파일에 적용할 인코딩입니다.
작업 매개 변수(메시지 부분에서 수행할 작업 지정)	
COMMAND	변환을 수행하기 위해 실행할 명령입니다. 변환을 수행하기 위해 실행할 명령입니다. 이 매개 변수는 필수입니다. 명령을 지정하지 않으면 항목이 무시됩니다. \ 대신 /를 사용하여 경로를 지정합니다(예: command="D:/tmp/mybat.bat").
DELETE	0 또는 1입니다. 이 플래그를 설정하면 메시지 부분이 삭제됩니다. 이 부분이 메시지의 유일한 부분인 경우 빈 단일 텍스트 부분이 대체됩니다.
RELABEL	RELABEL=1인 경우 MIME 레이블을 출력 매개 변수에 지정된 레이블로 다시 지정합니다. Relabel=0인 경우에는 레이블을 다시 지정하지 않습니다. 일반적으로 레이블 다시 지정은 레이블이 잘못 지정된 부분에서 수행됩니다. 예를 들어,Content-type: application/octet-stream에서 Content-type: application/msword로 레이블을 다시 지정합니다. 그렇게 하면 해당 부분을 파일로 저장하여 프로그램에서 열 필요 없이 해당 부분을 "두 번 눌러" 열 수 있습니다.
SERVICE-COMMAND	SERVICE-COMMAND=command는 전체 MIME 메시지(MIME 헤더 및 내용 본문 부분)에 적용할 사이트에서 제공하는 절차를 실행합니다. 다른 CHARSET-CONVERSION 작업 또는 변환 채널 작업과 달리 service-command는 MIME 역어셈블리, 디코딩, 재인코딩 및 재어셈블리를 자체적으로 수행합니다. 이 플래그는 변환 채널을 처리하는 동안 항목이 무시되게 합니다. SERVICE-COMMAND 항목은 문자 세트 변환 처리 중에 대신 수행됩니다. \ 대신 /를 사용하여 경로를 지정합니다(예: command="D:/tmp/mybat.bat").
정보 전달 매개 변수(사이트에서 제공하는 프로그램을 통해 정보를 전달하는 데 사용됨)	

표 13–6 변환 매개 변수 *(계속)*

매개 변수	설명
DPARAMETER-SYMBOL-*n*	Content-disposition 매개 변수 값(있는 경우)이 저장되는 환경 변수입니다(*n* = 0, 1, 2,...). 각 DPARAMETER-SYMBOL-*n*은 Content-Disposition: 매개 변수 목록에서 순서대로(n=0이 첫 번째 매개 변수, n=2가 두 번째 매개 변수 등) 추출되어 사이트에서 제공하는 프로그램을 실행하기 전에 지정된 환경 변수에 저장됩니다.
PARAMETER-SYMBOL-*n*	Content-disposition 매개 변수 값(있는 경우)이 저장되는 환경 변수입니다(*n* = 0, 1, 2...). 각 PARAMETER-SYMBOL-*n*은 Content-Type: 매개 변수 목록에서 순서대로(n=0이 첫 번째 매개 변수, n=2가 두 번째 매개 변수 등) 추출되어 사이트에서 제공하는 프로그램을 실행하기 전에 동일한 이름의 환경 변수에 저장됩니다. IN-PARAMETER-NAME-*n* 절과 일치할 경우 MIME 매개 변수를 변환할 변수 이름을 인수로 사용합니다.
MESSAGE-HEADER-FILE	메시지 원본 헤더의 모두 또는 일부를 MESSAGE_HEADERS 환경 변수에서 지정한 파일에 쓰거나 전혀 쓰지 않습니다. 이 값을 1로 설정하면 포함된 본문 부분의 원본 헤더를 MESSAGE_HEADERS 환경 변수에서 지정한 파일에 씁니다. 이 값을 2로 설정하면 메시지 원본 헤더(가장 바깥에 있는 메시지 헤더) 전체를 해당 파일에 씁니다.
ORIGINAL-HEADER-FILE	0 또는 1입니다. 이 값을 1로 설정하면 포함된 MESSAGE/RFC822 부분(본문 부분이 아님)의 원본 헤더를 ORIGINAL_HEADERS 환경 변수에 표시된 파일에 씁니다.
OVERRIDE-HEADER-FILE	0 또는 1입니다. 이 값을 1로 설정하는 경우 변환 채널은 포함된 MIME 부분의 원본 헤더 행을 무시하고 OUTPUT_HEADERS 환경 변수에서 MIME 헤더 행을 읽습니다.
OVERRIDE-OPTION-FILE	OVERRIDE-OPTION-FILE=1인 경우 변환 채널은 OUTPUT_OPTIONS 환경 변수에서 옵션을 읽습니다.
PART-NUMBER	점으로 구분된 정수(*a. b. c...*)입니다. MIME 본문 부분의 부품 번호입니다.

13.5.6 아랍어 문자 세트 자동 감지

아랍어 문자 세트를 자동으로 감지하기 위한 새로운 auto_ef 프로그램이 추가되었습니다.

변환 채널에서 auto_ef 프로그램을 호출하여 레이블링되지 않았거나 잘못 레이블링된 아랍어 문자 세트의 텍스트 메시지를 대부분 자동으로 감지하고 레이블링할 수 있습니다. 이러한 레이블링되지 않았거나 잘못 레이블링된 메시지는 일반적으로 Yahoo나 Hotmail에서 아랍어로 보낸 메시지입니다.

문자 세트 레이블링을 제대로 하지 못하면 많은 메일 클라이언트에서 메시지를 올바로 표시하지 못합니다.

메시지에 MIME 내용 유형 헤더가 있으면 auto_ef 프로그램은 텍스트/일반 내용 유형의 헤더만 검사하고 처리합니다. 메시지가 MIME 내용 유형 헤더로 레이블링되어 있지 않은 경우에는 auto_ef는 텍스트/일반 내용 유형을 무조건 추가합니다.

이 프로그램을 활성화하거나 사용 가능하도록 설정하려면 다음을 수행해야 합니다.

▼ 아랍어 문자 세트 자동 감지 방법

1 ***msg_svr_base*/config 디렉토리에 있는 매핑 파일을 편집하여 선택에 따라 소스 및 대상 채널에 대해 변환 채널을 활성화합니다. 인터넷에서 로컬 사용자에게 오는 모든 메시지에 대해 변환 채널을 활성화하려면 매핑 파일에 다음과 유사한 섹션을 추가합니다.**

```
CONVERSIONS

   IN-CHAN=tcp*;OUT-CHAN=ims-ms;CONVERT YES
```

IN 및 OUT 채널은 구성에 따라 다릅니다. 릴레이 MTA에 배포하려면 해당 구성에 맞게 채널을 수정해야 합니다. 예를 들면 다음과 같습니다.

```
IN-CHAN=tcp*;OUT-CHAN=tcp*;CONVERT YES
```

또는 다음과 같이 모든 채널에 대해 설정할 수 있습니다.

```
IN-CHAN=*;OUT-CHAN=*;CONVERT YES
```

2 **Messaging Server 사용자가 소유하고 읽을 수 있는 *msg_svr_base*/config 디렉토리에 다음 내용을 포함하는 변환 파일을 만듭니다.**

```
!
in-channel=*; out-channel=*;
  in-type=text; in-subtype=*;
  parameter-copy-0=*; dparameter-copy-0=*;
  original-header-file=1; override-header-file=1;
  command="msg-svr-base
/lib/arabicdetect.sh"
!
```

3 **다음 명령으로 MTA 구성을 컴파일합니다.**

msg-svr-base/sbin/imsimta cnbuild

4 **다음 명령을 사용하여 다시 시작합니다.**

msg-svr-base /sbin/imsimta restart

13.6 문자 세트 변환 및 메시지 형식 다시 지정

이 절에서는 MTA에서 내부적으로 수행한 문자 집합과 형식 지정, 레이블 지정의 변환에 대해 설명합니다. 이 절에 나와 있는 예 중 일부는 DEC VMS 같이 오래되었거나 더 이상 사용되지 않는 기술 또는 d 채널을 사용합니다. 이러한 기술이 오래된 것이라 해도 DEC나 d 채널에만 사용되는 예를 제시한 것은 아니며 변환 기술의 작동 방법을 설명하는 데 있어서는 여전히 유효한 예라 할 수 있습니다. 향후 릴리스에서는 이러한 예를 업데이트할 예정입니다.

Messaging Server의 가장 기본적인 매핑 테이블 중 하나는 문자 세트 변환 테이블입니다. 이 테이블의 이름은 CHARSET-CONVERSION입니다. 이 테이블은 수행할 채널 간 문자 세트 변환 및 메시지 형식 다시 지정 종류를 지정하는 데 사용됩니다.

여러 시스템에서는 문자 세트 변환 또는 메시지 형식 다시 지정을 수행할 필요가 없으므로 이 테이블이 필요하지 않습니다. 그러나, 문자 변환을 수행해야 하는 상황이 발생할 수 있습니다. 예를 들어 일본어 OpenVMS를 실행하는 사이트는 DEC Kanji와 현재 인터넷에서 사용되는 ISO-2022 Kanji 사이에 변환해야 할 필요가 있습니다. 변환이 사용될 또 다른 경우로서, 다국어 문자가 너무 많이 사용되었기 때문에 DEC 다국어 문자 집합(DEC-MCS)과 MIME에 사용되도록 지정된 ISO-8859-1 문자 집합 사이에 약간의 차이가 발생할 수 있고 이 두 집합 간에 실제 변환이 필요할 수 있습니다.

또한 CHARSET-CONVERSION 매핑 테이블을 사용하여 메시지의 형식을 변경할 수도 있습니다. 많은 비 MIME 형식을 MIME 형식으로 변환하는 기능이 제공됩니다. 또한 MIME 인코딩과 구조를 변경할 수 있습니다. 이러한 옵션은 MIME 또는 MIME의 일부 하위 집합만 지원하는 시스템에 메시지를 릴레이할 때 사용됩니다. 마지막으로 MIME을 비 MIME 형식으로 변환하는 기능이 제공되는 경우도 가끔씩 있습니다.

MTA는 다음과 같은 두 가지 방법으로 CHARSET-CONVERSION 매핑 테이블을 감시합니다. 첫 번째 감시 방법은 MTA가 메시지 형식을 다시 지정해야 하는지 여부를 확인하고 그렇게 해야 할 경우 사용할 형식 지정 옵션을 결정하는 데 사용됩니다. 형식 다시 지정을 지정하지 않으면 MTA는 특정 문자 세트 변환을 확인하지 않습니다. 이 첫 번째 감시 방법에 사용되는 입력 문자열의 형식은 일반적으로 다음과 같습니다.

IN-CHAN=*in-channel*;OUT-CHAN=*out-channel*;CONVERT

여기서 *in-channel*은 소스 채널(메시지를 가져온 채널)의 이름이고 *out-channel*은 대상 채널(메시지가 이동하는 채널)의 이름입니다. 일치가 발생할 경우 쉼표로 구분된 키워드 목록이 결과 문자열로 표시됩니다. 키워드는 표 13–7에 나열되어 있습니다.

표 13–7 HARSET-CONVERSION 매핑 테이블 키워드

키워드	설명
Always	*out-channel*로 이동하기 전에 변환 채널을 통해 메시지를 전달하더라도 변환을 수행합니다.
Appledouble	다른 MacMIME 형식을 Appledouble 형식으로 변환합니다.
Applesingle	다른 MacMIME 형식을 Applesingle 형식으로 변환합니다.
BASE64	MIME 인코딩을 BASE64로 전환합니다. 이 키워드는 이미 인코딩된 메시지 부분에만 적용됩니다. Content-transfer-encoding: 7비트 또는 8비트를 가진 메시지는 특수한 인코딩이 필요하지 않으므로 이 BASE64 옵션이 아무 영향을 미치지 않습니다.
Binhex	다른 MacMIME 형식 또는 Macintosh 유형 및 Mac 작성자 정보를 포함하는 부분을 Binhex 형식으로 변환합니다.

표 13–7 HARSET-CONVERSION 매핑 테이블 키워드 *(계속)*

키워드	설명
Block	MacMIME 형식 부분에서 데이터 포크만 추출합니다.
Bottom	메시지/rfc822 본문 부분(전달된 메시지)을 메시지 내용 부분과 헤더로 "결합"합니다.
Delete	전달된 헤더를 삭제하여 메시지/rfc822 본문 부분(전달된 메시지)을 메시지 내용 부분으로 "결합"합니다.
Level	메시지에서 중복 멀티파트 수준을 제거합니다.
Macbinary	다른 MacMIME 형식 또는 Macintosh 유형 및 Macintosh 작성자 정보를 포함하는 부분을 Macbinary 형식으로 변환합니다.
No	변환을 비활성화합니다.
QUOTED-PRINTABLE	MIME 인코딩을 QUOTED-PRINTABLE로 전환합니다.
Record,Text	텍스트/일반 부분을 80자에서 줄 바꿈합니다.
Record,Text= n	텍스트/일반 부분을 n자에서 줄 바꿈합니다.
RFC1154	메시지를 RFC 1154 형식으로 변환합니다.
Top	메시지/rfc822 본문 부분(전달된 메시지)을 헤더 부분과 메시지 내용 부분으로 "결합"합니다.
UUENCODE	MIME 인코딩을 X-UUENCODE로 전환합니다.
Yes	변환을 활성화합니다.

13.6.1 문자 세트 변환

MTA가 메시지 형식을 다시 지정해야 하는지를 감시하여 발견하는 경우 메시지의 각 부분을 확인합니다. 텍스트 부분이 있으면 해당 문자 세트 매개 변수를 사용하여 두 번째 감시를 생성합니다. MTA는 변환이 필요하다고 확인된 경우에만 두 번째 감시를 수행합니다. 이 두 번째 감시의 입력 문자열은 다음과 같습니다.

`IN-CHAN=`*in-channel*`;OUT-CHAN=`*out-channel*`;IN-CHARSET=`*in-char-set*

in-channel 및 *out-channel*은 앞의 예와 동일하고 *in-char-set*는 문제가 있는 특정 부분과 연결된 문자 세트의 이름입니다. 첫 번째 감시에서 키워드가 일치하여 메시지 형식이 다시 지정(예: MIME 구조 변경)되더라도 이 두 번째 감시에서 일치 항목이 발견되지 않으면 문자 세트 변환이 수행되지 않습니다. 일치 항목이 발견될 경우 다음과 같은 형식의 문자열이 생성됩니다.

`OUT-CHARSET=`*out-char-set*

여기서 *out-char-set*는 in-char-set를 변환해야 하는 문자 세트의 이름을 지정합니다. 이러한 문자 세트는 모두 MTA 테이블 디렉토리에 있는 문자 세트 정의 테이블 charsets.txt에서 정의해야 합니다. 문자 세트를 이 파일에 제대로 정의하지 않으면 변환이 수행되지 않습니다. 이 파일에는 수 백개의 문자 세트가 정의되어 있고 현재 사용 중인 대부분의 문자 세트가 이 파일에 정의되어 있기 때문에 이러한 경우는 흔하지 않습니다. charsets.txt 파일에 대한 자세한 내용은 imsimta chbuild(UNIX 및 NT) 유틸리티 설명을 참조하십시오.

모든 조건이 충족되면 MTA는 문자 세트 매핑을 작성하고 변환을 수행합니다. 변환된 메시지 부분의 레이블은 변환된 문자 세트의 이름으로 다시 지정됩니다.

Charset-conversion 매핑이 확장되어 다음과 같은 여러 추가 기능을 제공합니다.

- IN-CHARSET 옵션은 매핑 항목의 출력 템플리트에서 지정할 수 있습니다. 이 옵션이 있을 경우 인코딩된 단어에서 지정된 문자 세트를 무시합니다.
- 정수 0 또는 1을 가지는 RELABEL-ONLY 옵션을 지정할 수 있습니다. 이 옵션의 값이 1인 경우 OUT-CHARSET은 단순히 IN-CHARSET을 대체하고 레이블을 다시 지정하지는 않습니다.
- IN-CHARSET 옵션을 사용하여 입력 문자 세트를 *로 설정하면 문자 세트가 "감지되어" 적절한 레이블이 결정됩니다.

예 13–2 ISO-8859-1과 UTF-8 사이의 변환

로컬로 ISO-8859-1이 사용된다고 할 때, 인터넷에서 사용하기 위해서는 UTF-8로 변환되어야 합니다. 특히, 인터넷 연결이 tcp_local을 통한 것이라고 가정할 때 tcp_internal과 ims-ms는 내부 메시지가 만들어져 전달되는 위치입니다. 아래는 이러한 변환을 수행하는 CHARSET-CONVERSION 표입니다. 각 IN-CHAN 항목은 한 행에 있어야 합니다. 이를 나타내기 위해 역슬래시(\)가 사용됩니다.

```
CHARSET-CONVERSION

 IN-CHAN=tcp_internal;OUT-CHAN=tcp_local;CONVERT                Yes
 IN-CHAN=tcp_local;OUT-CHAN=tcp_internal;CONVERT                Yes
 IN-CHAN=tcp_local;OUT-CHAN=ims-ms;CONVERT                      Yes
 IN-CHAN=*;OUT-CHAN=*;CONVERT                                   No
 IN-CHAN=tcp_internal;OUT-CHAN=tcp_local;IN-CHARSET=ISO-8859-1 OUT-CHARSET=UTF-8
 IN-CHAN=tcp_local;OUT-CHAN=tcp_internal;IN-CHARSET=UTF-8 OUT-CHARSET=ISO-8859-1
 IN-CHAN=tcp_local;OUT-CHAN=ims-ms;IN-CHARSET=UTF-8       OUT-CHARSET=ISO-8859-1
```

예 13–3 EUC-JP와 ISO-2022-JP 사이의 변환

아래 CHARSET-CONVERSION 표는 EUC-JP의 로컬 사용과 ISO 2022 기반 JP 코드 사이의 변환을 지정합니다.

```
CHARSET-CONVERSION
```

예 13–3 EUC-JP와 ISO-2022-JP 사이의 변환 *(계속)*

```
IN-CHAN=ims-ms;OUT-CHAN=ims-ms;CONVERT                    No
IN-CHAN=tcp_internal;OUT-CHAN=ims-ms;CONVERT              No
IN-CHAN=tcp_internal;OUT-CHAN=tcp_internal;CONVERT        No
IN-CHAN=tcp_internal;OUT-CHAN=*;CONVERT                   Yes
IN-CHAN=*;OUT-CHAN=ims-ms;CONVERT                         Yes
IN-CHAN=*;OUT-CHAN=tcp_internal;CONVERT                   Yes
IN-CHAN=tcp_internal;OUT-CHAN=*;IN-CHARSET=EUC-JP      OUT-CHARSET=ISO-2022-JP
IN-CHAN=*;OUT-CHAN=ims-ms;IN-CHARSET=ISO-2022-JP        OUT-CHARSET=EUC-JP
IN-CHAN=*;OUT-CHAN=tcp_internal;IN-CHARSET=ISO-2022-JP  OUT-CHARSET=EUC-JP
```

13.6.2 메시지 형식 다시 지정

위에서 설명한 것처럼 CHARSET-CONVERSION 매핑 테이블은 MIME과 여러 해당 메일 형식 사이에서 첨부 파일을 변환하는 데도 사용됩니다.

다음 절에서는 CHARSET-CONVERSION 매핑 테이블에서 적용할 수 있는 메시지 형식 다시 지정의 몇 가지 다른 예를 제공합니다.

13.6.2.1 비 MIME 이진 첨부 파일 변환

특정 형식의 메일 또는 Microsoft Mail(MSMAIL) SMTP 게이트웨이에서 가져온 메시지 등과 같은 특정 비표준(비 MIME) 형식 메일은 메시지 처리에 관계된 채널에 대해 CHARSET-CONVERSION을 활성화할 경우 MIME 형식으로 자동으로 변환됩니다. tcp_local 채널은 일반적으로 Microsoft Mail SMTP 게이트웨이를 통해 가져온 메시지에 대한 수신 채널이며, 다음은 로컬 사용자에게 전달된 메시지에 대한 변환을 활성화합니다.

```
CHARSET-CONVERSION

  IN-CHAN=tcp_local;OUT-CHAN=ims-ms;CONVERT          Yes
```

또한 채널에 대한 항목을 다른 로컬 메일 시스템에 추가하려고 할 수 있습니다. 예를 들어 다음은 tcp_internal 채널에 대한 항목입니다.

```
CHARSET-CONVERSION

  IN-CHAN=tcp_local;OUT-CHAN=l;CONVERT               Yes
  IN-CHAN=tcp_local;OUT-CHAN=tcp_internal;CONVERT    Yes
```

모든 채널을 포함하려면 OUT-CHAN=ims-ms 대신 OUT-CHAN=*를 지정하면 됩니다. 그러나, 이렇게 하면 tcp_local 채널에 수신되는 모든 메시지를 특정 채널에 바운드하지 않고 꼼꼼하게 검사하게 되므로 메시지 처리 오버헤드가 증가할 수 있습니다.

무엇보다도 그러한 무분별한 변환은 시스템을 통해 단순히 전달하기만 하면 되는(메시지 봉투와 관련 전송 정보 이외에 다른 변환을 수행할 필요 없이 단순히 전송만

하면 되는 상황) 변환 메시지로 인해 시스템이 정렬되지 않은 복잡한 상태로 될 수 있습니다(사용자의 시스템은 그렇지 않을 수도 있음).

MIME을 Microsoft Mail SMTP 게이트웨이에서 인식할 수 있는 형식으로 변환하려면 Microsoft Mail SMTP 게이트웨이에 대한 MTA 구성에서 별도의 채널(예: `tcp_msmail`)을 사용하고 매핑 파일에 다음을 입력합니다.

```
CHARSET-CONVERSION

  IN-CHAN=*;OUT-CHAN=tcp_msmail;CONVERT        RFC1154
```

13.6.2.2 MIME 헤더 레이블 다시 지정

일부 사용자 에이전트 또는 게이트웨이에서는 내용에 비해 부족하지만 세부 MIME 헤더를 구성하는 데는 충분한 정보를 제공하는 MIME 헤더를 사용하여 메시지를 보낼 수 있습니다. 그러한 사용자 에이전트 또는 게이트웨이를 적절하게 구성하는 것이 가장 좋지만 해당 구성을 직접 제어할 수 없는 경우 MIME 헤더를 보다 유용하게 다시 구성하도록 MTA에 요청할 수 있습니다.

`CHARSET-CONVERSION` 매핑 테이블에 대한 첫 번째 감시에서 `Yes` 또는 `Always` 키워드를 생성하는 경우 MTA는 `conversions` 파일이 있는지를 확인합니다. 변환 파일이 있는 경우 MTA는 해당 파일에서 `RELABEL=1`인 항목을 조사하여 이러한 항목이 있는 경우 해당 항목에 지정된 MIME 레이블을 다시 지정합니다. 변환 파일 항목에 대한 자세한 내용은 404 페이지 “13.5.3 변환 처리 제어”를 참조하십시오.

예를 들어 다음과 같은 `CHARSET-CONVERSION` 테이블과

```
CHARSET-CONVERSION

  IN-CHAN=tcp_local;OUT-CHAN=tcp_internal;CONVERT            Yes
```

MTA `conversion` 파일 항목을

```
out-chan=ims-ms; in-type=application; in-subtype=octet-stream;
  in-parameter-name-0=name; in-parameter-value-0=*.ps;
  out-type=application; out-subtype=postscript;
  parameter-copy-0=*; relabel=1

out-chan=ims-ms; in-type=application; in-subtype=octet-stream;
  in-parameter-name-0=name; in-parameter-value-0=*.msw;
  out-type=application; out-subtype=msword;
     parameter-copy-0=* relabel=1
```

조합하면 `tcp_local` 채널에 도착하고 `ims-ms` 채널에 라우팅된 메시지가 나타나게 되는데, 이는 원래 application/octet-stream이라는 MIME 레이블 지정을 사용하여 도착하되 파일 이름 매개 변수에 `ps` 또는 `msw`라는 확장명이 있는 메시지로서 각각 application/postscript 또는 application/msword라는 레이블이 붙여집니다. 이러한 보다

세부적인 레이블 지정은 원본 사용자 에이전트 또는 게이트웨이에서 수행되었어야 하는 내용입니다. 이러한 레이블 다시 지정은 MIME-CONTENT-TYPES-TO-MR 매핑 테이블과 함께 사용할 때 특히 유용합니다. MIME-CONTENT-TYPES-TO-MR 매핑 테이블은 결과로 나타나는 MIME 형식을 다시 적합한 MRTYPE 태그로 변환하는 데 사용되며 해당 태그가 최적 상태로 기능하기 위해서는 MIME 레이블을 정확히 지정해야 합니다. 모든 내용 유형에 application/octet-stream으로만 레이블이 지정된 경우 MIME-CONTENT-TYPES-TO-MR 매핑 테이블은 모든 형식을 무조건 MRTYPE이라는 한 가지 태그로만 변환할 수 있습니다.

위의 예와 함께 다음이 포함된 MIME-CONTENT-TYPES-TO-MR 매핑 테이블 항목을 사용할 때

```
APPLICATION/POSTSCRIPT        PS
APPLICATION/MSWORD              MW
```

레이블 지정을 수행하면

```
Content-type: application/octet-stream; name=stuff.ps
```

다음과 같이 레이블이 다시 지정되고

```
Content-type: application/postscript
```

MRTYPE 태그로 변환됨으로써 PS 메시지 라우터가 PostScript를 예상할 수 있게 됩니다.

일반 이진 데이터에 대해 반대 방향으로 레이블 다시 지정(특정 MIME 첨부 파일 레이블 지정을 application/octet-stream으로 "다운그레이드")하는 것이 유용할 경우가 있습니다. 특히, 특정 MIME 레이블 지정의 "다운그레이드"는 mime_to_x400 채널(PMDF-X400) 또는 xapi_local 채널(PMDF-MB400)의 convert_octet_stream 채널 키워드와 함께 사용하여 모든 이진 MIME 첨부 파일을 X.400 bodypart 14 형식으로 강제 설정하는 경우가 많습니다.

예를 들어, 다음과 같은 CHARSET-CONVERSION 매핑 테이블과

```
CHARSET-CONVERSION

    IN-CHAN=*;OUT-CHAN=mime_to_x400*;CONVERT Yes
```

CHARSET-CONVERSION 매핑 테이블과 다음 PMPF 변환 파일 항목을

```
out-chan=mime_to_x400*; in-type=application; in-subtype=*;
   out-type=application; out-subtype=octet-stream; relabel=1

out-chan=mime_to_x400*; in-type=audio; in-subtype=*;
   out-type=application; out-subtype=octet-stream; relabel=1

out-chan=mime_to_x400*; in-type=image; in-subtype=*;
   out-type=application; out-subtype=octet-stream; relabel=1

out-chan=mime_to_x400*; in-type=video; in-subtype=*;
   out-type=application; out-subtype=octet-stream; relabel=1
```

조합하면 mime_to_x400* 채널로 가는 모든 메시지에 대해 특정 MIME 첨부 파일 레이블 지정이 일반 application/octet-stream 레이블 지정으로 다운그레이드되고 그에 따라 convert_octet_stream이 적용됩니다.

13.6.2.3 MacMIME 형식 변환

Macintosh 파일에는 Macintosh 특정 정보가 들어 있는 자원 포크와 다른 플랫폼에서 사용 가능한 데이터가 들어 있는 데이터 포크의 두 부분이 있습니다. 따라서, Macintosh 파일 전송은 더욱 복잡하게 수행됩니다. Macintosh 파일 부분 전송에는 서로 다른 네 가지 형식이 공통적으로 사용됩니다. Applesingle, Binhex 및 Macbinary의 세 형식은 한 부분으로 함께 인코딩되는 Macintosh 자원 포크와 Macintosh 데이터 포크로 구성됩니다. 네 번째 형식인 Appledouble은 자원 포크와 데이터 포크가 별도의 부분에 존재하는 멀티파트 형식입니다. 따라서, Appledouble이 Macintosh 이외의 플랫폼에 가장 유용한 형식입니다. 이 경우 비 Macintosh 응용 프로그램에서는 자원 포크 부분은 무시하고 데이터 포크 부분만 사용할 수 있습니다. 다른 형식은 Macintosh에 메시지를 보낼 경우에 특히 유용합니다.

MTA는 이러한 다양한 Macintosh 형식을 변환할 수 있습니다. CHARSET-CONVERSION 키워드 Appledouble, Applesingle, Binhex 또는 Macbinary는 다른 MacMIME 구조 부분을 각각 multipart/appledouble, application/applefile, application/mac-binhex40 또는 application/macbinary MIME 구조로 변환하도록 MTA에 지시합니다. 또한 Binhex 또는 Macbinary 키워드는 MIME Content-type: 헤더에 X-MAC-TYPE 및 X-MAC-CREATOR 매개 변수를 포함하는 비 MacMIME 형식 부분의 지정된 형식으로 변환하도록 적용됩니다. CHARSET-CONVERSION 키워드 Block은 MacMIME 형식 부분에서 자원 포크는 삭제하고 데이터 포크만 추출하도록 MTA에 지시합니다. 이렇게 하면 정보가 손실되므로 일반적으로 Appledouble을 사용하는 것이 좋습니다.

예를 들어 다음 CHARSET-CONVERSION 테이블은 MTA에게 VMS MAIL 메일함이나 GroupWise 사서함으로 전달할 때는 Appledouble 형식으로 변환하고 메시지 라우터 채널로 전달할 때는 Macbinary 형식으로 변환하도록 명령합니다.

```
CHARSET-CONVERSION
   IN-CHAN=*;OUT-CHAN=l;CONVERT              Appledouble
   IN-CHAN=*;OUT-CHAN=wpo_local;CONVERT      Appledouble
   IN-CHAN=*;OUT-CHAN=tcp_internal;CONVERT   Macbinary
```

Appledouble 형식으로의 변환은 이미 MacMIME 형식 중 하나로 된 부분에만 적용됩니다. Macbinary 형식으로의 변환은 이미 MacMIME 형식 중 하나로 된 부분에만 적용되거나 X-MAC-TYPE 및 X-MAC-CREATOR 매개 변수를 MIME Content-type: 헤더에 포함한 비 MacMIME 부분에만 적용됩니다.

Appledouble 또는 Block 형식으로 변환할 경우 MAC-TO-MIME-CONTENT-TYPES 매핑 테이블을 사용하여 원본 Macintosh 파일의 Macintosh 작성자 및 Macintosh 입력 정보에 따라 Appledouble 부분 또는 Block 부분의 데이터 포크에 넣을 특정 MIME 레이블을 지정할 수 있습니다. 이 테이블의 검사에는 *format*|*type*| *creator*|*filename*이라는 형식이

사용되는데 여기서 format은 SINGLE, BINHEX 또는 MACBINARY 중 하나이고 type 및 creator는 각각 16진수인 Macintosh 유형 및 Macintosh 작성자 정보이며 filename은 파일 이름입니다.

예를 들어, `ims-ms` 채널에 보낼 때 MACBINARY 또는 BINHEX 부분에서 변환된 MS Word 또는 PostScript 문서에 대해 특정 MIME 레이블을 사용하도록 Appledouble로 변환하는 테이블은 다음과 같습니다.

```
CHARSET-CONVERSION

  IN-CHAN=*;OUT-CHAN=ims-ms;CONVERT     Appledouble


MAC-TO-MIME-CONTENT-TYPES

! PostScript
    MACBINARY|45505346|76677264|*     APPLICATION/POSTSCRIPT$Y
    BINHEX|45505346|76677264|*        APPLICATION/POSTSCRIPT$Y
! Microsoft Word
    MACBINARY|5744424E|4D535744|*     APPLICATION/MSWORD$Y
    BINHEX|5744424E|4D535744|*        APPLICATION/MSWORD$Y
```

매핑 항목의 템플리트(오른쪽)에는 지정된 레이블 지정을 수행하도록 설정된 $Y 플래그가 있어야 합니다. 추가 첨부 파일 형식에 대한 샘플 항목은 MTA 테이블 디렉토리의 `mac_mappings.sample` 파일을 참조하십시오.

비 MacMIME 형식 부분을 Binhex 또는 Macbinary 형식으로 변환하려면 해당 부분은(매개 변수 값이 있을 경우) X-MAC-TYPE 및 X-MAC-CREATOR MIME Content-type:을 가져야 합니다. MIME 레이블 다시 지정을 사용하여 해당 매개 변수가 없는 부분에 이러한 매개 변수를 제공할 수 있습니다.

13.6.3 서비스 변환

MTA의 변환 서비스 기능을 사용하면 사이트에서 제공하는 절차에 따라 메시지를 처리하여 새로운 형식의 메시지를 생성할 수 있습니다. 개별 MIME 메시지 부분의 내용에 적용되는 위에서 설명한 `CHARSET-CONVERSION` 작업 또는 `conversion` 채널과 달리 변환 서비스는 전체 MIME 메시지 부분(MIME 헤더 및 내용)과 전체 MIME 메시지에 대해 수행됩니다. 다른 `CHARSET-CONVERSION` 작업 또는 변환 채널 작업과 달리 변환 서비스는 MIME 역어셈블리, 디코딩, 재인코딩 및 재어셈블리를 자체적으로 수행합니다.

다른 `CHARSET-CONVERSION` 작업과 마찬가지로 변환 서비스도 `CHARSET-CONVERSION` 매핑 테이블을 통해 활성화됩니다. `CHARSET-CONVESION` 매핑 테이블에 대한 첫 번째 감시에서 `Yes` 또는 `Always` 키워드를 생성하는 경우 MTA는 MTA `conversions` 파일이 있는지 확인합니다. `conversions` 파일이 있는 경우 MTA는 해당 파일에서 `SERVICE-COMMAND`를

지정하는 항목을 조사하여 그러한 항목이 발견되면 해당 항목을 실행합니다. conversions 파일 항목의 형식은 다음과 같습니다.

```
in-chan=channel-pattern;
  in-type=type-pattern; in-subtype=subtype-pattern;
  service-command=command
```

핵심적인 사항은 명령 문자열입니다. 명령 문자열은 서비스 변환을 수행하기 위해 실행해야 하는 명령(예: 문서 변환기 호출)입니다. 명령은 서비스를 제공할 메시지 텍스트가 들어 있는 입력 파일을 처리하여 새 메시지 텍스트가 포함된 출력 파일을 생성해야 합니다. UNIX에서 명령은 성공할 경우 0으로 끝나고 그렇지 않은 경우 0이 아닌 다른 값으로 끝납니다.

예를 들어, 다음과 같은 CHARSET-CONVERSION 테이블과

```
CHARSET-CONVERSION

IN-CHAN=bsout_*;OUT-CHAN=*;CONVERT Yes
```

UNIX MTA conversions 파일 항목을

```
in-chan=bsout_*; in-type=*; in-subtype=*;
service-command="/pmdf/bin/compress.sh compress $INPUT_FILE $OUTPUT_FILE"
```

조합하면 BSOUT 채널의 모든 메시지가 압축됩니다.

환경 변수는 메시지의 봉투 수신자 주소 목록이 포함된 파일 이름과 입력 및 출력 파일의 이름을 전달하는 데 사용됩니다. 이러한 환경 변수의 이름은 다음과 같습니다.

- INPUT_FILE - 처리할 입력 파일의 이름
- OUTPUT_FILE - 처리할 출력 파일의 이름
- INFO_FILE - 봉투 수신자 주소가 포함된 파일의 이름

표준 명령줄 대체를 사용하여 이러한 세 환경 변수 값을 명령줄로 대체할 수 있습니다. UNIX의 경우 변수 이름 앞에 $ 문자를 표시합니다. 예를 들어 INPUT_FILE과 OUTPUT_FILE에 a.in 및 a.out 값이 있을 때 UNIX에서 다음 선언은

```
in-chan=bsout_*; in-type=*; in-subtype=*;
 service-command="/pmdf/bin/convert.sh $INPUT_FILE $OUTPUT_FILE"
```

다음 명령을 실행합니다.

```
/pmdf/bin/convert.sh a.in a.out
```

◆ ◆ ◆ 14장

14 Messaging Server에 스팸 및 바이러스 필터링 프로그램 통합

이 장에서는 스팸 및 바이러스 필터링 소프트웨어를 Messaging Server에 통합하고 구성하는 방법에 대해 설명합니다. 이 장에 설명되어 있는 스팸/바이러스 필터링 기술은 변환 채널(401 페이지 "13.5 변환 채널" 참조)에서 제공하는 기술보다 훨씬 강력합니다. Messaging Server에서는 Symantec Brightmail AntiSpam, SpamAssassin, Milter 및 스팸 방지/ICAP(Internet Content Adaptation Protocol, RFC 3507)를 지원하는 바이러스 백신 프로그램, 특히 Symantec AntiVirus Scan Engine을 지원합니다.

주 – 이 장에서 **스팸 방지** 또는 **스팸 필터링** 기능은 **바이러스 백신** 또는 **바이러스 필터링** 기능을 의미하기도 합니다. 일부 제품(Brightmail)은 두 기능을 모두 제공하지만 다른 제품은 스팸 필터링만 제공하거나(SpamAssassin) 바이러스 필터링만 제공합니다(Symantec AntiVirus Scan Engine). 또한 일반적으로 구성 매개 변수에는 `spam`이 사용된다는 점에 유의하십시오.

이 장은 다음 내용으로 구성되어 있습니다.

14.1 Messaging Server에 스팸 필터링 프로그램 통합—작동 원리

Messaging Server의 관점에서 스팸 방지 솔루션은 거의 동일한 기능을 수행합니다.

1. Messaging Server가 메시지 복사본을 스팸 필터링 소프트웨어로 보냅니다.
2. 스팸 필터링 소프트웨어가 메시지를 분석하여 스팸인지 여부에 대한 답신을 보냅니다. SpamAssassin과 같은 일부 프로그램은 메시지가 스팸일 가능성을 숫자 등급으로 나타낸 **스팸 점수**를 답신과 함께 보냅니다.
3. Messaging Server가 답신을 읽고 메시지에 대한 시브(Sieve) 작업을 수행합니다(439 페이지 "14.2.3 스팸 메시지에 대해 수행할 작업 지정" 참조).

스팸 필터링 프로그램은 프로토콜을 통해 MTA와 상호 작용합니다. 이러한 프로토콜은 ICAP 기반 프로그램(예: Symantec AntiVirus Scan Engine)에서와 같이 표준 프로토콜이거나, Brightmail에서와 같이 전용 프로토콜이거나, Spam Assassin에서와 같이 표준이 아닌 프로토콜일 수 있습니다. 각 프로토콜에는 MTA와 상호 작용하기 위한 소프트웨어 후크가 필요합니다. Brightmail 및 SpamAssassin은 Messaging Server와 상호 작용할 수 있는 최초의 스팸 필터링 프로그램입니다. 이제 MTA에서는 ICAP를 사용하는 프로그램도 지원합니다.

14.2 타사 스팸 필터링 프로그램 배포 및 구성

Messaging Server에서 타사 필터링 소프트웨어를 배포하려면 다음의 다섯 가지 작업을 수행해야 합니다.

1. **배포할 스팸 필터링 프로그램과 이 스팸 필터링 프로그램을 배포할 서버의 수를 지정합니다.** Messaging Server에서는 최대 여덟 개의 다른 스팸/바이러스 프로그램을 사용하여 받는 메시지를 필터링할 수 있습니다. 이러한 프로그램은 별도의 시스템에서 실행할 수도 있고 단일 시스템 배포에서 Messaging Server와 동일한 시스템에서 실행하거나 2계층 배포에서 MTA와 동일한 시스템에서 실행할 수도 있습니다. 필요한 서버 수는 메시지 로드, 하드웨어 성능 및 기타 요인에 따라 달라집니다. 사이트의 하드웨어 요구 사항을 확인하려면 스팸 필터링 소프트웨어 설명서를 참조하거나 담당자에게 문의하십시오.
2. **스팸 필터링 소프트웨어를 설치 및 구성합니다.** 자세한 내용은 스팸 필터링 소프트웨어 설명서를 참조하거나 담당자에게 문의하십시오.
3. **필터링 클라이언트 라이브러리를 로드하고 구성합니다.** 여기에는 MTA `option.dat` 파일에서 클라이언트 라이브러리 및 구성 파일을 지정하고 필터링 소프트웨어의 구성 파일에서 원하는 옵션을 설정하는 것이 포함됩니다. 433 페이지 "14.2.1 스팸 필터링 소프트웨어 클라이언트 라이브러리 로드 및 구성"을 참조하십시오.
4. **필터링할 메시지를 지정합니다.** 사용자, 도메인 또는 채널별로 메시지를 필터링할 수 있습니다. 434 페이지 "14.2.2 필터링할 메시지 지정"을 참조하십시오.

5. **스팸 처리 방법을 지정합니다.** 스팸 삭제, 폴더에 정리, 제목 줄에 태그 지정 등을 수행할 수 있습니다. 439 페이지 “14.2.3 스팸 메시지에 대해 수행할 작업 지정”을 참조하십시오.

주 – 이전 버전의 Messaging Server에서는 Brightmail 필터링 기술만 지원했기 때문에 키워드와 옵션에 `sourcebrightmail` 또는 `Brightmail_config_file`과 같은 이름이 붙었습니다. 이러한 키워드와 옵션을 `sourcespamfilter` 또는 `spamfilter_config_file`과 같은 보다 일반적인 이름으로 변경했습니다. 이전 Brightmail 이름은 호환성을 위해 그대로 유지됩니다.

14.2.1 스팸 필터링 소프트웨어 클라이언트 라이브러리 로드 및 구성

각 스팸 필터링 프로그램에서 Messaging Server에 클라이언트 라이브러리 파일과 구성 파일을 제공해야 합니다. 클라이언트 라이브러리의 로드 및 구성에는 다음 두 가지 작업이 포함됩니다.

- `option.dat` 파일에서 스팸 필터링 소프트웨어 라이브러리 경로(`spamfilter`*X*`_library`) 및 구성 파일(`spamfilter`*X*`_config_file`) 지정. 이러한 옵션 외에도 스팸 필터링 LDAP 속성과 스팸 메시지에 대해 사용할 시브(Sieve) 작업을 지정하는 데 사용되는 여러 다른 옵션이 있습니다.
- 스팸 필터링 소프트웨어 구성 파일에서 원하는 옵션 지정. 각 스팸 필터링 프로그램은 서로 다른 구성 파일 및 구성 옵션을 가집니다. 이에 대해서는 스팸 필터링 소프트웨어 절이나 필터링 소프트웨어 설명서에 설명되어 있습니다. 443 페이지 “14.3 Symantec Brightmail 스팸 방지 사용” 및 461 페이지 “14.5 SAVSE(Symantec Anti-virus Scanning Engine) 사용”을 참조하십시오.

14.2.1.1 스팸 필터링 소프트웨어 라이브러리 경로 지정

Messaging Server는 메시지에 대해 최대 여덟 개의 다른 필터링 시스템을 호출할 수 있습니다. 예를들어, Symantec AntiVirus Scan Engine 및 SpamAssassin을 통해 메시지를 실행할 수 있습니다. 각 필터링 소프트웨어는 1에서 4까지의 번호로 식별됩니다. 이 번호는 여러 스팸 필터 옵션, LDAP 속성 및 채널 키워드의 일부로 표시되며 *X*가 `sourcespamfilter`*X*`optin` 또는 `spamfilter`*X*`_config_file`과 같이 필터 식별 번호로 사용됩니다. 키워드나 옵션 이름에서 식별 번호를 생략하면 기본값이 1이 됩니다.

다음 `option.dat` 설정은 Messaging Server가 Symantec AntiVirus Scan Engine 및 SpamAssassin을 통해 메시지를 필터링하도록 지정합니다.

```
spamfilter1_library=Symantec_Library_File
spamfilter1_config_file=Symantec_Config_File
spamfilter2_library=SpamAssassin_Library_File
spamfilter2_config_file=SpamAssassin_Config_File
```

다른 옵션이나 키워드를 사용하여 시스템을 구성할 경우 옵션이나 키워드의 끝에서 해당 번호를 사용합니다. 예를 들어, `sourcespamfilter2optin`은 SpamAssassin을 나타내고 `sourcespamfilter1optin`은 Symantec AntiVirus Scan Engine을 나타냅니다. 번호를 순서대로 사용할 필요는 없습니다. 예를 들어, Symantec AntiVirus Scan Engine을 일시적으로 사용하지 않으려면 `spamfilter1_library` 구성 파일을 주석 처리할 수 있습니다.

14.2.2 필터링할 메시지 지정

Messaging Server에 스팸 필터링 소프트웨어가 설치되어 실행할 준비가 되면 필터링할 메시지를 지정해야 합니다. 사용자, 도메인 또는 채널별로 메시지를 필터링하도록 Messaging Server를 구성할 수 있습니다. 다음 절에는 이러한 각 시나리오가 설명되어 있습니다.

- 434 페이지 "사용자 수준 필터링 지정"
- 435 페이지 "14.2.2.1 사용자 수준 필터 예"

주 – *optin* 표현식은 사용자, 도메인 또는 채널이 메일 필터링을 받도록 선택되었음을 의미합니다.

▼ 사용자 수준 필터링 지정

사용자 단위로 필터링을 지정하는 것이 바람직할 수 있습니다. 예를 들어, 스팸이나 바이러스 필터링을 ISP 고객에게 프리미엄 서비스로 제공한 경우 이 서비스를 받을 사용자와 받지 않을 사용자를 지정할 수 있습니다. 사용자 필터링을 위한 일반적인 단계는 다음과 같습니다.

1 스팸 필터링 소프트웨어를 활성화하는 사용자 LDAP 속성을 지정합니다.

`option.dat`에 `LDAP_OPTIN`*X* 옵션을 설정합니다. 예:

```
LDAP_OPTIN1=SymantecAV
LDAP_OPTIN2=SpamAssassin
```

주 – 기본적으로 `SymantecAV` 또는 `SpamAssassin`과 같은 속성은 스키마에 존재하지 않습니다. 어떤 새 속성을 사용하든지 디렉토리 스키마에 이를 추가해야 합니다. 자세한 내용은 해당 Directory Server 설명서를 참조하십시오.

---

2 스팸 필터링을 받을 사용자 항목의 필터 속성을 설정합니다.

필터 속성의 값은 다중 값이며 서버에 따라 다릅니다. 단계 1의 예를 사용할 경우 항목은 다음과 같습니다.

```
SymantecAV: virus
SpamAssassin: spam
```

바이러스와 스팸을 모두 필터링할 수 있는 Brightmail과 같은 프로그램의 경우 유효한 값은 spam 및 virus입니다. 다중 값 속성으로 사용되는 각 값에는 개별 속성 항목이 필요합니다. 예를 들어, Brightmail의 필터 속성이 Brightmail로 설정된 경우 항목은 다음과 같습니다.

```
Brightmail: spam
Brightmail: virus
```

14.2.2.1 사용자 수준 필터 예

이 예에서는 Brightmail을 사용한다고 가정합니다. 또한 option.dat 파일에서 LDAP_OPTIN1이 Brightmail로 설정되어 있다고 가정합니다. Otis Fanning이라는 사용자는 자신의 사용자 항목에서 Brightmail 속성을 spam 및 virus로 설정했습니다. 따라서, Brightmail은 그의 메일에서 스팸과 바이러스를 필터링합니다. 435 페이지 “14.2.2.1 사용자 수준 필터 예”에서는 Otis Fanning에 대한 Brightmail 사용자 항목을 보여줍니다.

예 14–1 Brightmail에 대한 LDAP 사용자 항목 예

```
dn: uid=fanning,ou=people,o=sesta.com,o=ISP
objectClass: person
objectClass: organizationalPerson
objectClass: inetOrgPerson
objectClass: inetUser
objectClass: ipUser
objectClass: inetMailUser
objectClass: inetLocalMailRecipient
objectClass: nsManagedPerson
objectClass: userPresenceProfile
cn: Otis Fanning
sn: fanning
initials: OTF
givenName: Otis
pabURI: ldap://ldap.siroe.com:389/ou=fanning,ou=people,o=sesta.com,o=isp,o=pab
mail: Otis.Fanning@sesta.com
mailAlternateAddress: ofanning@sesta.com
mailDeliveryOption: mailbox
mailHost: manatee.siroe.com
uid: fanning
dataSource: iMS 5.0 @(#)ims50users.sh 1.5a 02/3/00
userPassword: password
inetUserStatus: active
mailUserStatus: active
mailQuota: -1
mailMsgQuota: 100
Brightmail: virus
Brightmail: spam
```

Symantec AntiVirus Scan Engine 및 SpamAssassin을 사용하는 경우 항목은 다음과 같습니다.

```
SymantecAV: virus
SpamAssassin: spam
```

443 페이지 “14.3 Symantec Brightmail 스팸 방지 사용”, 448 페이지 “14.4 SpamAssassin 사용” 또는 461 페이지 “14.5 SAVSE(Symantec Anti-virus Scanning Engine) 사용”을 참조하십시오.

▼ 도메인 수준 필터링 지정

필터링을 받을 도메인을 지정할 수 있습니다. 예를 들어 스팸 방지 또는 바이러스 백신 필터링을 ISP 도메인 고객에게 프리미엄 서비스로 제공한 경우 이 기능을 사용할 수 있습니다. 도메인 필터링을 지정하는 일반적인 단계는 다음과 같습니다.

1 필터링 소프트웨어를 활성화하는 도메인 LDAP 속성을 지정합니다.

`option.dat`에서 `LDAP_DOMAIN_ATTR_OPTIN`*X* 옵션을 설정합니다. 예:

```
LDAP_DOMAIN_ATTR_OPTIN1=SymantecAV
LDAP_DOMAIN_ATTR_OPTIN2=SpamAssassin
```

주 – 기본적으로 `SymantecAV` 또는 `SpamAssassin`과 같은 속성은 스키마에 존재하지 않습니다. 어떤 새 속성을 사용하든지 디렉토리 스키마에 이를 추가해야 합니다. 자세한 내용은 해당 Directory Server 설명서를 참조하십시오.

---

2 스팸 필터링을 받을 도메인 항목에서 필터 속성을 설정합니다.

필터 속성의 값은 다중 값이며 서버에 따라 다릅니다. 단계 1의 예를 사용할 경우 항목은 다음과 같습니다.

```
SymantecAV: virus
SpamAssassin: spam
```

바이러스와 스팸을 모두 필터링할 수 있는 Brightmail과 같은 프로그램의 경우 유효한 값은 `spam` 및 `virus`입니다. 다중 값 속성으로 사용되는 각 값에는 개별 속성 값 항목이 필요합니다. 예를 들어, `LDAP_DOMAIN_ATTR_OPTIN1`이 `Brightmail`로 설정된 경우 항목은 다음과 같습니다.

```
Brightmail: spam
Brightmail: virus
```

도메인 수준 필터링 예

이 예에서는 Brightmail을 사용한다고 가정합니다. 또한 `option.dat` 파일에서 `LDAP_DOMAIN_ATTR_OPTIN1`이 `Brightmail`로 설정되어 있다고 가정합니다. Sun LDAP Schema 1에 대한 DC 트리의 `sesta.com` 도메인 항목에 Brightmail 속성이 `spam` 및 `virus`로 설정되어 있습니다. Sun LDAP Schema 2의 도메인 항목에서도 스팸 필터링을 받도록 `Brightmail`을 설정합니다.

Brightmail은 sesta.com으로 보낸 모든 메일에서 스팸과 바이러스를 필터링합니다. 436 페이지 “도메인 수준 필터링 예”는 다음과 같습니다.

예 14–2 Brightmail에 대한 LDAP 도메인 항목 예

```
dn: dc=sesta,dc=com,o=internet
objectClass: domain
objectClass: inetDomain
objectClass: mailDomain
objectClass: nsManagedDomain
objectClass: icsCalendarDomain
description: DC node for sesta.com hosted domain
dc: sesta
inetDomainBaseDN: o=sesta.com,o=isp
inetDomainStatus: active
mailDomainStatus: active
mailDomainAllowedServiceAccess: +imap, pop3, http:*
mailRoutingHosts: manatee.siroe.com
preferredMailHost: manatee.siroe.com
mailDomainDiskQuota: 100000000
mailDomainMsgQuota: -1
mailClientAttachmentQuota: 5
Brightmail: spam
Brightmail: virus
```

Symantec AntiVirus Scan Engine 및 SpamAssassin을 사용하는 경우 항목은 다음과 같습니다.

```
SymantecAV: virus
SpamAssassin: spam
```

자세한 내용과 추가 예는 443 페이지 “14.3 Symantec Brightmail 스팸 방지 사용”, 448 페이지 “14.4 SpamAssassin 사용” 또는 461 페이지 “14.5 SAVSE(Symantec Anti-virus Scanning Engine) 사용”을 참조하십시오.

▼ 채널 수준 필터링 지정

소스 및 대상 채널별로 필터링을 지정하면 스팸 필터링의 융통성과 세부 수준이 향상됩니다. 예를 들어, 다음과 같이 필터링을 지정할 수 있습니다.

- 특정 MTA 릴레이에서 백엔드 메시지 저장소로 보낸 메시지만
- 특정 MTA에서 받는 모든 메일
- 특정 MTA에서 보내는 모든 메일
- 특정 MTA에서 받는 메일과 보내는 메일

Messaging Server에서는 소스 또는 대상 채널을 기준으로 필터링을 지정할 수 있습니다. 이 작업을 수행하는 기법은 390 페이지 "12.12.5 스팸 필터 키워드"에 설명된 채널 키워드입니다. 다음 예에서는 채널 수준 필터링을 설정하는 방법을 설명합니다.

1 **백엔드 메시지 저장소 호스트에 메시지를 보내는 모든 인바운드 SMTP 서버의** `imta.cnf` **파일에 다시 쓰기 규칙을 추가합니다. 예:**

```
msg_store1.siroe.com $U@msg_store1.siroe.com
```

2 `destinationspamfilter`*X*`optin` **키워드를 사용하여 다시 쓰기 규칙에 해당하는 채널을 추가합니다. 예:**

```
tcp_msg_store1 smtp subdirs 20 backoff "pt5m" "pt10" "pt30" \
"pt1h" "pt2h" "pt4h" maxjobs 1 pool IMS_POOL \
fileinto $U+$S@$D destinationspamfilter1optin spam
msg_store1.siroe.com
```

채널 수준 필터링 예

이 예는 필터링 프로그램이 번호 1로 지정되었다고 가정합니다. 스팸 필터링에 사용 가능한 키워드는 390 페이지 "12.12.5 스팸 필터 키워드"를 참조하십시오.

▼ MTA 중계에서 백엔드 메시지 저장소로 필터링하는 방법

이 예에서는 MTA 중계에서 백엔드 메시지 저장소(`msg_store1.siroe.com`)로 오는 모든 메일에서 스팸 및 바이러스를 필터링합니다.

1 **메시지를 백엔드 메시지 저장소 호스트로 보내는 다시 쓰기 규칙을** `imta.cnf` **파일에 추가합니다. 예:**

```
msg_store1.siroe.com   $U@msg_store1.siroe.com
```

2 `destinationspamfilter`*X*`optin` **키워드를 사용하여 다시 쓰기 규칙에 해당하는 채널을 추가합니다. 예:**

```
tcp_msg_store1 smtp subdirs 20 backoff "pt5m" "pt10" "pt30" "pt1h" \
"pt2h" "pt4h" maxjobs 1 pool IMS_POOL fileinto $U+$S@$D \
destinationspamfilter 1optin spam,virus
msg_store1.siroe.com
```

예 2. MTA를 통해 전달하는 모든 받는 메일에서 스팸을 필터링합니다(일반적으로 모든 받는 메일은 `tcp_local` 채널을 통해 전달됨).

```
tcp_local smtp mx single_sys remotehost inner switchchannel \
identnonelimited subdirs 20 maxjobs 7 pool SMTP_POOL \
maytlsserver maysaslserver saslswitchchannel tcp_auth \
sourcespamfilter1optin spam
tcp-daemon
```

예 3. MTA를 통해 인터넷으로 전달하는 모든 보내는 메일을 필터링합니다(일반적으로 인터넷으로 보내는 모든 메시지는 tcp_local 채널을 통해 전달됨).

```
tcp_local smtp mx single_sys remotehost inner switchchannel \
identnonelimited subdirs 20 maxjobs 7 pool SMTP_POOL \
maytlsserver maysaslserver saslswitchchannel tcp_auth \
destinationspamfilter1optin spam tcp-daemon
```

예 4. MTA를 통해 전달하는 모든 받는 메일과 보내는 메일을 필터링합니다.

```
tcp_local smtp mx single_sys remotehost inner switchchannel \
identnonelimited subdirs 20 maxjobs 7 pool SMTP_POOL \
maytlsserver maysaslserver saslswitchchannel tcp_auth \
sourcespamfilter1optin spam destinationspamfilter1optin spam
tcp-daemon
```

예 5. 사용자별 수신 선택 기능을 사용하지 않고 2계층 시스템에서 로컬 메시지 저장소를 대상으로 하는 모든 메일을 필터링합니다.

```
ims-ms smtp mx single_sys remotehost inner switchchannel \
identnonelimited subdirs 20 maxjobs 7 pool SMTP_POOL \
maytlsserver maysaslserver saslswitchchannel tcp_auth \
destinationspamfilter1optin spam
tcp-daemon
```

예 6. 모든 받는 메일과 보내는 메일에서 스팸과 바이러스를 필터링합니다(소프트웨어가 스팸과 바이러스를 모두 필터링하는 것으로 가정).

```
tcp_local smtp mx single_sys remotehost inner switchchannel \
identnonelimited subdirs 20 maxjobs 7 pool SMTP_POOL \
maytlsserver maysaslserver saslswitchchannel tcp_auth \
destinationspamfilter1optin spam,virus sourcespamfilter1optin \
spam,virus
tcp-daemon
```

14.2.3 스팸 메시지에 대해 수행할 작업 지정

스팸 필터링 프로그램은 메시지를 분석하여 Messaging Server의 현재 버전에 스팸 여부에 대한 답신을 보냅니다. 그러면 Messaging Server에서 해당 메시지에 대한 작업을 수행합니다. 작업은 시브(Sieve) 메일 필터링 언어를 사용하여 지정됩니다. 메시지 삭제, 메시지를 폴더에 정리, 헤더 추가, 제목 줄에 태그 추가 등의 작업을 수행할 수 있습니다. if-then-else 문이 있는 복잡한 시브(Sieve) 스크립트를 사용할 수도 있습니다.

주 – 전체 시브(Sieve) 구문은 시브(Sieve) 사양 3028을 참조하십시오. 또한 http//www.cyrusoft.com/sievettp://www.cyrusoft.com/sieve/ (http://www.cyrusoft.com/sieve/)를 참조하십시오.

시브(Sieve) 스크립트는 표 14–1에서 설명한 MTA 스팸 필터 옵션(option.dat)을 통해 지정됩니다. 기본 스팸 필터 작업 옵션은 SpamfilterX_null_action(null 값이 스팸 답신 값으로 반환될 때 실행하도록 시브(Sieve) 규칙 지정) 및 SpamfilterX_string_action(문자열이 스팸 답신으로 반환될 때 실행하도록 시브(Sieve) 규칙 지정)입니다.

스팸 필터링 프로그램은 일반적으로 MTA에 해당 메시지가 스팸임을 나타내는 문자열 또는 null 값을 반환합니다. 메시지가 스팸일 확률을 나타내는 숫자 등급인 스팸 점수를 반환하는 프로그램도 있습니다. 이 점수는 작업 시퀀스의 일부로 사용될 수 있습니다. 다음 예는 필터링된 메시지에 대해 수행할 작업을 지정하는 방법을 보여줍니다. 각 예에서 필터링 프로그램이 번호 1로 지정되었다고 가정합니다.

예 1: 답신 값이 null인 스팸 메시지를 SPAM_CAN 파일로 저장합니다.

```
spamfilter1_null_action=data:,require "fileinto"; fileinto "SPAM_CAN  ;
```

문자열을 반환하는 스팸 메시지에 대해 동일한 작업을 수행할 수 있습니다.

```
spamfilter1_string_action=data:,require "fileinto"; fileinto "SPAM_CAN  ;
```

예 2: 답신이 문자열인 스팸 메시지를 MTA에 반환된 해당 답신 문자열로 명명된 파일에 저장합니다($U가 수행하는 작업). 즉, 반환된 답신 문자열이 spam일 경우 메시지는 spam이라는 파일에 저장됩니다.

```
spamfilter1_null_action=data:,require "fileinto"; fileinto "$U  ;
```

예 3: 답신 값이 문자열인 스팸 메시지를 삭제합니다.

```
spamfilter1_string_action=data:,discard
```

null 값을 반환하는 스팸 메시지에 대해 동일한 작업을 수행할 수 있습니다.

```
spamfilter1_null_action=data:,require "fileinto"; fileinto "SPAM_CAN  ;
```

예 4. 아래 행은 문자열 답신 값을 통해 스팸으로 확인된 각 메시지에 Spam-test: FAIL 헤더를 추가합니다.

```
spamfilter1_string_action=data:,require ["addheader"];addheader "Spam-test:
FAIL";
```

예 5. 이 행은 문자열을 반환하는 스팸 메시지의 제목 줄에 [PROBABLE SPAM] 문자열을 추가합니다.

```
spamfilter1_string_action=data:,addtag “[PROBABLE SPAM]”;
```

예 6. 이 행은 헤더에 resent-from 및 User-1이 포함되어 있는 경우 문자열 답신 값으로 간주하고 스팸 메시지를 testspam 메일함에 파일로 저장합니다. 해당 헤더가 없는 경우 메시지를 spam 파일에 저장합니다.

```
spamfilter1_string_action=data:,require "fileinto"; \
  if header :contains ["resent-from"] ["User-1"] {  \
  fileinto "testspam"; \
  } else {  \
  fileinto "spam";};
```

대부분의 스팸 필터 소프트웨어로 답신 문자열을 구성할 수 있기 때문에 반환되는 문자열에 따라 다른 작업을 지정할 수 있습니다. 이 작업은 일치하는 `spamfilter`*X*`_verdict_`*n* 및 `spamfilter`*X*`_action_`*n* 옵션 쌍을 사용하여 수행할 수 있습니다.

예 7. 이 일치 쌍 옵션은 반환된 답신 문자열이 `remove`인 스팸 메시지를 삭제합니다.

```
spamfilter1_verdict_0=remove
spamfilter1_action_0=data:,discard
```

스팸 답신 문자열 지정 방법에 대한 자세한 내용은 해당 스팸 필터링 소프트웨어 절을 참조하십시오.

표 14–1 MTA 스팸 필터 옵션(option.dat)

Spam Assassin에 대한 MTA 옵션	설명
`Spamfilter` *X*`_config_file`	필터링 소프트웨어 X 구성 파일의 전체 파일 경로와 이름을 지정합니다. 기본값: 없음
`Spamfilter` *X*`_library`	필터링 소프트웨어 X 공유 라이브러리의 전체 파일 경로와 이름을 지정합니다. 기본값: 없음
`Spamfilter` *X*`_optional`	필터링 라이브러리 X가 보고한 특정 실패가 일시적인 처리 실패로 간주되는지 무시되는지 여부를 제어합니다. `0`은 스팸 필터링 문제로 일시적인 처리 실패가 발생함을 지정합니다. `1`은 모든 이벤트가 아니라 일부 이벤트에서 스팸 필터 처리를 건너뛰어 필터링 라이브러리가 실패합니다. 특히, 시스템이 라이브러리 코드에 값을 반환하지 않고 고착 상태가 될 경우 MTA의 일부 부분도 함께 고착될 수 있습니다. -2와 2는 스팸 필터 플러그 인에서 보고한 문제의 이벤트에 syslog 메시지를 보낸다는 점만 제외하고 각각 0 및 1과 같습니다. `3`은 스팸 필터 오류가 발생하여 메시지를 수락하지만 reprocess 채널에서 메시지를 대기시켜 나중에 처리할 수 있도록 합니다. `4`는 `3`과 같지만 스팸 필터 임시 실패를 `syslog`에도 기록합니다. 기본값: `0`
`LDAP_optin` *X*	사용자 단위로 필터링 소프트웨어 X를 활성화하는 데 사용되는 LDAP 속성의 이 름을 지정합니다. 필터링은 대상 주소를 기반으로 합니다. 즉, 이 속성을 가진 사용자에게 전송된 메시지를 스팸으로 필터링합니다. 이 속성은 `inetMailUser` objectclass의 속성이어야 합니다. 속성 자체는 여러 값을 가질 수 있으며 대/소문자를 구분합니다. SpamAssassin의 경우 이 값은 소문자 `spam`이어야 합니다. 기본값: 없음

표 14–1 MTA 스팸 필터 옵션(option.dat) *(계속)*

Spam Assassin에 대한 MTA 옵션	설명
LDAP_SOURCE_OPTIN*X*	LDAP_SOURCE_OPTIN1 ~ LDAP_SOURCE_OPTIN8에서는 LDAP_optin*X*에 해당하는, 발송자 주소 기반 사용자 단위 스팸 필터 옵션 값을 제공합니다. 즉, 이 사용자가 발송하는 메일은 스팸으로 필터링됩니다.
LDAP_domain_attr_optin *X*	도메인 단위로 필터링 소프트웨어 X를 활성화하는 데 사용되는 LDAP 속성의 이름을 지정합니다. 대상 도메인에 적용됩니다. mailDomain objectclass에 있어야 한다는 점을 제외하고 LDAP_optin과 동일합니다. 기본값: 없음
Spamfilter *X*_null_optin	LDAP_optin *X* 또는 LDAP_domain_attr_optin*X*에서 정의한 속성 값으로 발견된 경우 MTA에서 해당 속성이 없는 것처럼 간주하는 문자열을 지정합니다. 즉, 해당 항목에 대한 필터링을 비활성화합니다. 434 페이지 "14.2.2 필터링할 메시지 지정"을 참조하십시오. 기본값: 빈 문자열. 빈 optin 속성은 기본적으로 무시됩니다(빈 optin 속성이 빈 optin 목록을 갖는 필터링을 트리거했던 iPlanet Messaging Server 5.2와 달라진 내용입니다. spamfilter*X*_null_optin을 실제로 표시된 적이 없는 문자열로 설정하여 5.2 동작을 복원할 수 있습니다.)
Spamfilter *X*_null_action	필터링 소프트웨어 X 답신이 null로 반환될 경우에 메시지에서 수행할 작업을 지정하는 시브(Sieve) 규칙을 정의합니다. 파일 URL을 사용하여 시브(Sieve) 문을 외부적으로 저장할 수 있습니다. 예를 들면 다음과 같습니다. file:///var/opt/SUNWmsgsr/config/null_action.sieve . 또한, 시브(Sieve) 거부 작업을 사용하여 스팸을 거부하지 않기 때문에 주소를 사용하여 스팸을 보낸 적이 있는 악의 없는 사용자에게 배달 실패 알림을 전달합니다. 기본값: data:,discard;
Spamfilter*X*_string_action	답신이 문자열일 경우에 메시지에서 수행할 작업을 지정하는 시브(Sieve) 규칙을 정의합니다. 파일 URL을 사용하여 시브(Sieve) 문을 외부적으로 저장할 수 있습니다. 예를 들면 다음과 같습니다. file:///var/opt/SUNWmsgsr/config/null_action.sieve. 또한, 시브(Sieve) reject 작업을 사용하여 스팸을 거부하지 않기 때문에 서버를 사용하여 스팸을 보낸 적이 있는 악의 없는 사용자에게 배달 실패 알림을 전달합니다. 기본값: data:,require "fileinto"; fileinto "$U; $U는 verdict가 반환한 문자열입니다.
spamfilter*X*_verdict_ *n*	spamfilter*X*_verdict_*n* 및 spamfilter *X*_action_*n* 옵션은 일치하는 쌍입니다. 여기서 *n*은 0에서 9 사이의 숫자입니다. 이러한 옵션을 사용하면 임의의 답신 문자열에 대한 시브(Sieve) 필터를 지정할 수 있습니다. 이 작업은 spamfilter*X*_verdict_*n* 및 spamfilter*X*_action_*n*을 각각 답신 문자열 및 시브(Sieve) 필터로 설정하여 수행합니다. 여기서 *n*은 0에서 9까지의 정수입니다. 예를 들어 사이트에 "reject" 답신이 있는 경우 다음을 지정하여 시브(Sieve) 거부 작업을 발생시킵니다. spamfilter1_verdict_*0*=reject spamfilter1_action_*0*=data:,require "reject"; reject "Rejected by spam filter"; 모든 spamfilter*X*_verdict_*n* 옵션과 해당 작업 옵션의 기본값은 빈 문자열입니다. 기본값: 없음

표 14–1 MTA 스팸 필터 옵션(option.dat) (계속)

Spam Assassin에 대한 MTA 옵션	설명
spamfilter*X*_action_*n*	spamfilter*X*_verdict_*n*을 참조하십시오. 기본값: 없음
spamfilter*X*_final	일부 필터링 라이브러리에는 수신자 주소를 기반으로 작업을 수행하는 기능이 있습니다. spamfilter*X*_final은 필터링 라이브러리에 전달되는 수신자 주소의 종류를 지정합니다. 0은 중간 주소가 사용되게 지정하고 1은 최종 수신자 주소 형식을 보냅니다. 기본값: 0
optin_user_carryover	전달은 스팸 필터 처리를 위한 시도입니다. forward 전달 옵션을 지정하고 다른 사용자의 전달 주소를 지정하는 사용자 항목을 고려합니다. 또한, 사용자 항목은 특정 필터링 종류에 대한 optin으로 설정됩니다. 전달된 메시지에 필터링을 적용하시겠습니까? 다시 말해서 특정 사용자에게 올바른 필터링 선택이 다른 사용자에게는 올바른 선택이 아닐 수 있습니다. 즉, 필터링 작업 제거가 사이트의 보안 정책을 위반하는 수단으로 사용될 수 있습니다. 모든 경우에서 어떤 답변도 옳지 않기 때문에 OPTIN_USER_CARRYOVER는 전달 중에 사용자/별칭 항목의 스팸 필터링 optin 목록이 다른 사용자/별칭 항목에 적용되는 방법을 제어합니다. 이는 비트 인코딩 값입니다. 비트 값의 의미는 다음과 같습니다. 비트 0(값 1). 각 LDAP 사용자 항목이 이전의 활성 사용자/도메인 optin을 무조건적으로 대체합니다. 비트 1(값 2). 사용자의 도메인에 optin 속성이 있는 경우 활성화된 이전 사용자/도메인/별칭 optin을 대체합니다. 비트 2(값 4). 사용자에게 optin 속성이 있는 경우 활성화된 이전 사용자/도메인/별칭 optin을 대체합니다. 비트 3(값 8). [optin] 비지정 매개 변수에 지정된 optin이 활성화된 이전 사용자/도메인/별칭 optin을 대체합니다. 기본값: 0(사용자가 다른 사용자에게 전달하는 전달 옵션이 있는 경우 optin이 누적됩니다. 기본값은 전달 중에 사이트 보안 정책을 적용하고 다른 설정은 적용되지 않게 합니다.)

14.3 Symantec Brightmail 스팸 방지 사용

Brightmail 솔루션은 전자 메일 서버에 다운로드되는 실시간 스팸 방지 및 바이러스 백신 규칙 업데이트와 Brightmail Server로 구성됩니다. 아래의 절 외에도 **Configuring Brightmail with Sun Java System Messaging Server**를 참조하십시오.

14.3.1 Brightmail 작업 방법

Brightmail Server는 고객 사이트에 배포됩니다. Brightmail에는 인터넷을 통한 새로운 스팸 감지를 위한 전자 메일 감시 설정이 있습니다. Brightmail 기술자들은 이러한 스팸을 실시간으로 차단하기 위한 사용자 정의 규칙을 작성합니다. 이러한 규칙은 Brightmail Server에 실시간으로 다운로드됩니다. Brightmail 데이터베이스는 업데이트되고 Brightmail Server는 지정된 사용자 또는 도메인의 전자 메일에 대해 이 데이터베이스 필터를 실행합니다.

14.3.1.1 Brightmail 구조

그림 14–1에서는 Brightmail 구조에 대해 설명합니다.

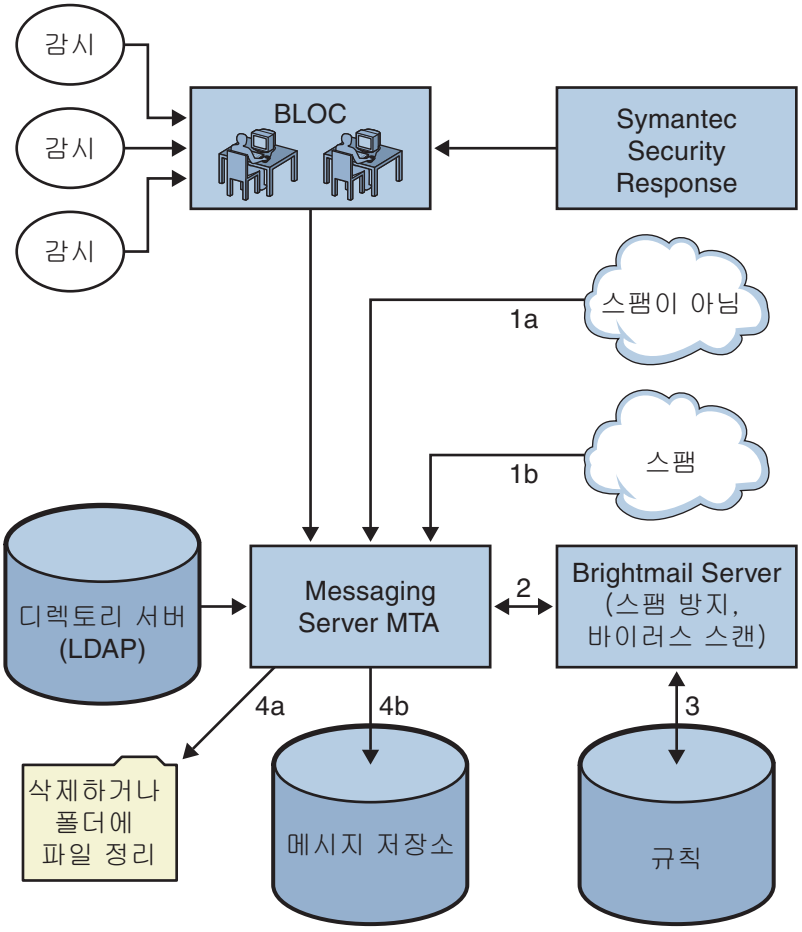


그림 14–1 Brightmail 및 Messaging Server 구조

BLOC(Brightmail Logistics and Operations Center)에서 전자 메일 감시를 통해 스팸을 받는 경우 연산자는 해당 스팸 방지 규칙을 즉시 생성하여 Brightmail 고객 시스템에 다운로드되게 합니다. 마찬가지로 Symantec Security Response 실시간 바이러스 규칙을 Brightmail을 통해 받습니다. 이러한 규칙은 고객의 Brightmail Server에서 스팸 및 바이러스를 찾아내는 데 사용됩니다.

MTA는 Brightmail SDK를 사용하여 Brightmail Server와 통신합니다. MTA는 Brightmail의 응답에 따라 메시지를 발송합니다. MTA는 메일 (1a) 또는 (1b)를 받은 후 Brightmail Server로 보냅니다(2). Brightmail Server는 해당 규칙과 데이터를 사용하여 메시지가 스팸 또는 바이러스인지 확인하고(3) MTA에 답신을 보냅니다. 답신에 따라 MTA는 (4a) 폴더에서 메시지 또는 파일을 삭제하거나 (4b) 일반적으로 대상에 전달합니다.

Brightmail SDK는 타사 소프트웨어이기 때문에 본사의 설치 키트에는 제공되지 않습니다. Brightmail SDK 및 서버 소프트웨어는 고객이 Brightmail Inc.를 통해 직접 구입해야 합니다. MTA에는 Brightmail 통합을 위해 Brightmail SDK를 로드할지 여부와 로드 위치를 알려주는 구성 설정이 있습니다.

SDK가 로드되면 여러 요소와 세부(granularity - Brightmail에서 현재 프로세싱이 *optin*임을 지정하는 데 사용되는 용어) 수준으로 Brightmail 메시지 처리를 결정합니다. 이는 다음 기준에 따라 지정됩니다.

- 소스 채널 또는 대상 채널이 Brightmail에 대해 활성화되는지 여부(`imta.cnf`)
- (`imta.cnf`)에서 선택한 서비스에 대한 채널 기본값이 있는지 여부
- 도메인별 수신 선택 기능(optin)이 있는지 여부(LDAP)
- 사용자별 수신 선택 기능(optin)이 있는지 여부(LDAP)

특정 메시지 수신자에 대해 위의 optin 값과 기본값을 결합합니다. 그렇게 하면 스팸과 바이러스 모두에 대해 채널 기본값이 이미 지정되어 있는 경우 각 사용자 optin 값을 제공할 필요가 없습니다. 즉, 시스템 관리자가 모든 사용자에 대해 스팸 및 바이러스 필터링을 수행하도록 결정할 경우 스팸 또는 바이러스에 대해 수신 선택 기능(optin)을 사용자에게 제공할 필요가 없습니다. 처리를 중단할 수 있는 방법은 없습니다. 즉, 시스템이나 도메인 수신 선택 기능(optin)을 통해 이미 선택된 서비스는 취소할 수 없습니다. 이것은 또한 서비스에 대해 수신 선택 기능(optin)을 사용하고 메일을 다른 주소로 전달한 경우, 사용자를 대신하여 필터링이 수행된 후에 해당 주소가 메일을 받게 된다는 것을 의미합니다.

바이러스 감지 또는 스팸 감지의 두 서비스만 제공됩니다. Brightmail은 “content-filtering” 서비스도 제공하지만 이 기능은 시브(Sieve)를 사용하여 제공되므로 Brightmai에서 시브(Sieve) 필터링을 수행할 수 있도록 추가된 값이 없습니다.

메시지에 바이러스가 있는 것으로 확인되면 바이러스를 삭제하고 치료된 메시지를 MTA에 다시 제출하도록 Brightmail Server를 구성할 수 있습니다. 다시 제출된 치료된 메시지에 원본 메시지에 대한 정보가 없을 경우 원하지 않은 부작용이 발생하기 때문에 치료된 메시지를 다시 MTA로 제출하도록 Brightmail을 구성하지 않는 것이 좋습니다. 메시지가 스팸인 경우 MTA는 Brightmail의 답신과 구성을 함께 사용하여 메시지에

표시되는 내용은 결정할 수 있습니다. 메시지를 삭제하거나, 폴더에 정리하거나, 제목 줄에 스팸 또는 바이러스 태그를 지정하거나, 시브(Sieve) 규칙에 전달하거나, INBOX에 전달할 수 있습니다.

Brightmail Server는 MTA와 동일한 시스템에 위치하거나 별도의 시스템에 위치할 수 있습니다. 실제로, 하나 이상의 MTA에 서비스를 제공하는 Brightmail Server 그룹을 가질 수 있습니다. Brightmail SDK는 Brightmail 구성 파일을 사용하여 사용할 Brightmail Server를 결정합니다.

14.3.2 Brightmail 요구 사항 및 성능 고려 사항

- Brightmail Server는 Solaris 운영 체제에서 실행해야 합니다.
- Brightmail에서 스팸 검사와 바이러스 검사를 모두 구현하는 경우 MTA 메시지 처리 능력이 50%까지 감소될 수 있습니다. MTA 처리 능력을 유지하려면 각 MTA에 대해 두 대의 Brightmail Server가 필요합니다.
- SpamAssassin에는 사용자 단위로 다양한 필터링을 수행할 수 있는 기능이 있지만 동일한 메시지에 한 번에 두 개의 서로 다른 필터링 기준을 적용할 수는 없습니다. 따라서, SpamAssassin은 시스템 차원 필터링만 허용합니다. 개별 사용자에 대한 사용자 정의 필터링은 사용할 수 없습니다.

14.3.3 Brightmail 배포

Brightmail을 배포하려면 다음 단계를 수행합니다.

- **Brightmail을 설치하고 구성합니다.** 설치 및 구성 정보는 Brightmail 소프트웨어 설명서를 참조하거나 담당자에게 문의하십시오. 선택한 Brightmail 구성 옵션은 447 페이지 “14.3.4 Brightmail 구성 옵션”에 표시되어 있지만 자세한 최신 정보는 Brightmail 설명서를 참조하십시오.
- **Brightmail 클라이언트 라이브러리를 로드하고 구성합니다.** 이 작업에는 Brightmail 클라이언트 라이브러리 `libbmiclient.so`와 MTA에 대한 구성 파일 `config`를 지정하는 것이 포함됩니다. 433 페이지 “14.2.1 스팸 필터링 소프트웨어 클라이언트 라이브러리 로드 및 구성”을 참조하십시오.
- **스팸을 필터링할 메시지를 지정합니다.** 사용자, 도메인 또는 채널별로 메시지를 필터링할 수 있습니다. 434 페이지 “14.2.2 필터링할 메시지 지정”을 참조하십시오.
- **스팸 메시지에 대해 수행할 작업을 지정합니다.** 스팸 삭제, 폴더에 정리, 제목 줄에 태그 지정 등을 수행할 수 있습니다. 439 페이지 “14.2.3 스팸 메시지에 대해 수행할 작업 지정”을 참조하십시오.
- **기타 MTA 필터 구성 매개 변수를 원하는 대로 설정합니다.** 표 14–1을 참조하십시오.

14.3.4 Brightmail 구성 옵션

선택한 Brightmail 구성 파일 옵션은 표 14–2에 표시되어 있습니다. Brightmail 구성 파일 환경 옵션의 전체 목록은 Brightmail에서 구할 수 있습니다. 옵션과 값은 대소문자를 구분하지 않습니다.

표 14–2 선택된 Brightmail 구성 파일 옵션

Brightmail 옵션	설명
`blSWPrecedence`	지정된 메시지에 여러 답신이 있을 수 있습니다. 이 옵션은 우선 순위를 지정합니다. 이 옵션을 `virus-spam`으로 지정한 경우 메시지에서 바이러스 검사를 먼저 수행한 다음 스팸 검사를 수행합니다. 답신은 하이픈(-)으로 구분됩니다. Sun Java System Messaging Server에서 Brightmail을 사용할 경우에 권장되는 설정입니다.
`blSWClientDestinationDefault`	스팸 또는 바이러스 대신 일반적인 메시지 전달 방법을 지정하므로 답신이 없습니다. 이 메시지를 일반적인 방법으로 전달하려면 `inbox`를 값으로 지정합니다. 기본값은 없습니다.
`blSWLocalDomain`	이 속성은 로컬로 간주되는 도메인을 지정합니다. 로컬로 간주되는 여러 도메인을 지정하는 여러 행으로 된 이 속성이 있을 수 있습니다. 로컬 도메인과 외부 도메인은 답신에 대한 서로 다른 두 가지 처리를 지정하기 위해 구분하여 사용됩니다. `blSWClientDestinationLocal` 및 `blSWClientDestinationForeign`을 참조하십시오. 예를 들어, 다음을 지정할 수 있습니다. `blSWLocalDomain=siroe.com`

`blSWClientDestinationLocal`	이 속성은 로컬 도메인에 대한 답신과 작업 쌍을 지정합니다. 일반적으로 스팸과 바이러스에 대해 각각 한 행씩 두 개의 행이 있습니다. 값은 `verdict|action` 형식입니다. 예를 들면 다음과 같습니다.
	
	`blSWClientDestinationLocal=spam|spambox`
	
	`blSWClientDestinationLocal=virus|`
	
	"null" 작업(|의 오른쪽에 아무 것도 없음)에 대한 기본 Brightmail 해석은 메일 삭제입니다. 따라서, 위의 예에서 `virus` 답신이 있는 경우 메시지가 삭제됩니다. 답신이 `spam`인 경우 메시지를 `spambox`라는 폴더에 정리합니다. 메시지가 스팸 또는 바이러스가 아닌 경우 답신이 일치하지 않고 위의 `blSWClientDestinationDefault` 설정을 기준으로 메일을 전달합니다.
	
	MTA에서 서버를 사용하거나 별도의 Brightmail Server를 사용할 경우 `Brightmail_verdict_n`, `Brightmail_action_n`, `Brightmail_null_action` 및 `Brightmail_string_action` MTA 옵션을 사용하여 정의된 작업은 Brightmail Server에서 반환되는 작업 및 답신에 우선합니다. 이 예의 경우 MTA에서 다른 `Brightmail_null_action`을 사용하여 Virus 작업(삭제)을 대체하거나 `Brightmail_verdict_0=spambox` 및 `Brightmail_action_0=data:,require "fileinto";fileinto "Junk";`를 사용하여 메일을 `spambox` 대신 `Junk` 폴더에 정리할 수 있습니다.

표 14–2 선택된 Brightmail 구성 파일 옵션 (계속)

Brightmail 옵션	설명
blSWClientDesintationForeign	로컬이 아닌 도메인에 있는 사용자에게 적용된다는 점을 제외하고 위의 blSWClientDestinationLocal과 형식 및 해석이 동일합니다.
blSWUseClientOptin	Sun Java System Messaging Server에서 사용할 경우 이 속성을 항상 TRUE로 설정합니다.
blswcServerAddress	ip:port[,ip:port,...] 형식을 사용하여 하나 이상의 Brightmail Server의 IP 주소와 포트 번호를 지정합니다.

14.4 SpamAssassin 사용

이 절은 다음과 같은 하위 절로 구성되어 있습니다.

14.4.1 SpamAssassin 개요

Messaging Server는 스팸을 식별하는 데 사용되는 프리웨어 메일 필터인 SpamAssassin의 사용을 지원합니다. SpamAssassin는 Perl에서 작성된 라이브러리와 SpamAssassin을 메시징 시스템에 통합하는 데 사용될 수 있는 일련의 응용 프로그램과 유틸리티로 구성됩니다.

SpamAssassin은 메일 헤더 및 본문 정보에 대한 일련의 테스트를 수행하여 모든 메시지에 대한 점수를 계산합니다. 테스트가 성공하거나 실패할 때마다 true(스팸) 또는 false(스팸 아님) 답신이 렌더링됩니다. 점수는 양의 실수 또는 음의 실수입니다. 점수가 특정 임계값(일반적으로 5.0)을 초과하면 스팸으로 간주됩니다. 다음은 SpamAssassin 결과 문자열의 예입니다.

True ; 18.3 / 5.0

True는 메시지가 스팸임을 나타냅니다. 18.3은 SpamAssassin 점수입니다. 5.0은 임계값입니다.

SpamAssassin은 자세히 구성할 수 있습니다. 언제든지 테스트를 추가하거나 제거하여 기존의 테스트 점수를 조정할 수 있습니다. 이 작업은 다양한 구성 파일을 통해 수행됩니다. SpamAssassin에 대한 자세한 내용은 SpamAssassin 웹 사이트를 참조하십시오.

Brightmail 스팸 및 바이러스 스캔 라이브러리를 호출하는 데 사용한 것과 동일한 기법을 사용하여 SpamAssassin `spamd` 서버에 연결할 수 있습니다. Messaging Server에 제공되는 모듈을 `libspamass.so`라 합니다.

14.4.2 SpamAssassin/Messaging Server 작동 원리

`spamd`는 SpamAssassin의 데몬 버전이며 MTA에서 호출할 수 있습니다. `spamd`는 소켓에서 요청을 수신하고 메시지 테스트를 위한 하위 프로세스를 생성합니다. 하위 프로세스는 메시지를 처리하여 결과를 돌려 보낸 후 종료됩니다. 이론적으로 하위 프로세스 간에 코드 자체가 공유되기 때문에 포크는 효율적인 프로세스이어야 합니다.

SpamAssassin 설치에서 클라이언트 부분인 `spamc`는 사용되지 않습니다. 이 기능은 Messaging Server의 일부인 `libspamass.so`라는 공유 라이브러리에 의해 대신 수행됩니다. `libspamass.so`는 Brightmail SDK에서와 동일한 방법으로 로드됩니다.

MTA의 관점에서 SpamAssassin과 Brightmail을 거의 투명하게 전환하여 스팸을 필터링할 수 있습니다. 그렇지만 서로 간에 기능이 동일하지 않기 때문에 완전히 투명하지는 않습니다. 예를 들어, Brightmail은 바이러스도 필터링할 수 있지만 SpamAssassin은 스팸을 필터링하는 데만 사용됩니다. 또한, 두 소프트웨어 패키지가 반환하는 결과 또는 **답신**도 다릅니다. SpamAssassin은 점수를 제공하는 반면 Brightmail은 답신 이름을 제공하기 때문에 구성 상에 여러 가지 차이점이 있습니다.

MTA와 통합된 SpamAssassin을 사용할 경우 SpamAssassin에서 점수와 답신만 반환됩니다. 메시지 자체는 수정되지 않습니다. 즉, 헤더 추가, 제목 줄 수정 등과 같은 작업은 시브(Sieve) 스크립트를 통해 수행해야 합니다. 또한 `mode` 옵션을 사용하면 답신을 가리키도록 반환되는 문자열을 지정할 수 있습니다. 문자열로 선택할 수 있는 것은 null이나 기본값, SpamAssassin 결과 문자열, `verdict` 문자열 등입니다. 자세한 내용은 458 페이지 "14.4.7 SpamAssassin 옵션"을 참조하십시오.

14.4.3 SpamAssassin 요구 사항 및 사용 시 고려 사항

- SpamAssassin은 무료입니다. 소프트웨어와 설명서는 http//www.spamassassin.org에서 구할 수 있습니다.
- SpamAssassin을 조정 및 구성하여 스팸을 매우 정확하게 감지할 수 있습니다. 이러한 조정은 사용자와 SpamAssassin 커뮤니티가 직접 수행해야 합니다. Messaging Server는 SpamAssassin에서 수행할 수 있는 작업을 제공하거나 향상시키지 않습니다.
- 특정 번호를 사용할 수 없는 동안 SpamAssassin은 Brightmail보다 처리 능력이 떨어집니다.
- MTA와 통합된 SpamAssassin은 사용자, 도메인 또는 채널에 대해 사용할 수 있습니다.
- Vipul Razor 또는 DCC(Distributed Checksum Clearinghouse)와 같은 다른 온라인 데이터베이스를 사용하도록 SpamAssassin을 구성할 수 있습니다.

- Messaging Server는 SSL(Secure Socket Layer) 버전의 `libspamass.so`를 제공하지 않지만 openSSL을 사용하도록 SpamAssassin을 작성할 수 있습니다.
- Perl 5.6 이상이 필요합니다.

14.4.3.1 SpamAssassin을 실행하는 장소

SpamAssassin은 자체 시스템, 단일 시스템 배포의 Messaging Server와 동일한 시스템 또는 2계층 배포의 MTA와 동일한 시스템에서 실행할 수 있습니다. LMTP(Local Mail Transfer Protocol)가 MTA와 메시지 저장소 사이에서 사용되는 경우 MTA에서 필터링을 호출해야 합니다. 메시지 저장소에서는 필터링을 호출할 수 없습니다. MTA와 메시지 저장소 간에 SMTP가 사용되는 경우 각 시스템이나 별도의 타사 시스템에서 실행할 수 있습니다.

SpamAssassin을 실행하는 서버 그룹을 사용하려면 해당 그룹의 앞에 로드 밸런서를 사용해야 합니다. MTA는 SpamAssassin 서버에 대해 단일 주소로만 구성됩니다.

14.4.4 SpamAssassin 배포

SpamAssassin을 배포하려면 다음 단계를 수행합니다.

- **SpamAssassin을 설치하고 구성합니다.** 설치 및 구성 정보는 SpamAssassin 소프트웨어 설명서를 참조하십시오. 458 페이지 "14.4.7 SpamAssassin 옵션"을 참조하십시오.
- **SpamAssassin 클라이언트 라이브러리를 로드 및 구성합니다.** 이 작업에는 클라이언트 라이브러리 `libspamass.so`와 MTA에 대한 구성 파일(만들어야 함)을 지정하는 것이 포함됩니다. 433 페이지 "14.2.1 스팸 필터링 소프트웨어 클라이언트 라이브러리 로드 및 구성"을 참조하십시오.
- **스팸을 필터링할 메시지를 지정합니다.** 사용자, 도메인 또는 채널별로 메시지를 필터링할 수 있습니다. 434 페이지 "14.2.2 필터링할 메시지 지정"을 참조하십시오.
- **스팸 메시지에 대해 수행할 작업을 지정합니다.** 스팸 삭제, 폴더에 정리, 제목 줄에 태그 지정 등을 수행할 수 있습니다. 439 페이지 "14.2.3 스팸 메시지에 대해 수행할 작업 지정"을 참조하십시오.
- **기타 필터 구성 매개 변수를 원하는 대로 설정합니다.** 표 14–1을 참조하십시오.

14.4.5 SpamAssassin 구성 예

이 절에서는 일반적인 몇 가지 SpamAssassin 구성 예를 설명합니다.

- 451 페이지 "스팸을 별도의 폴더에 정리"
- 452 페이지 "스팸 메시지에 SpamAssassin 점수가 포함된 헤더 추가"
- 453 페이지 "SpamAssassin 결과 문자열을 제목 줄에 추가"

주 – 이 예에서는 많은 옵션과 키워드를 사용합니다. 390 페이지 “12.12.5 스팸 필터 키워드” 및 표 14–1을 참조하십시오.

▼ 스팸을 별도의 폴더에 정리

이 예에서는 로컬 메시지 저장소에서 받는 메시지를 테스트하여 스팸을 spam이라는 폴더에 정리합니다. 처음 세 단계는 순서에 관계없이 수행할 수 있습니다.

1 SpamAssassin 구성 파일을 만듭니다.

이 파일의 이름과 위치는 단계 2에서 지정합니다. spamassassin.opt라는 이름을 사용하는 것이 좋습니다. 이 파일은 다음과 같은 행으로 구성되어 있습니다.

```
host=127.0.0.1
port=2000
mode=0
verdict=spam
debug=1
```

host 및 port는 spamd가 실행 중인 시스템의 이름과 spamd가 받는 요청을 수신하는 포트를 지정합니다. mode=0은 메시지가 스팸으로 간주될 때 verdict에 지정된 문자열이 반환되도록 지정합니다. debug=1은 SpamAssassin 라이브러리에서 디버깅을 설정합니다. 표 14–3을 참조하십시오.

2 option.dat 파일에 다음 행을 추가합니다.

```
! for Spamassassin
spamfilter1_config_file=/opt/SUNWmsgsr/config/spamassassin.opt
spamfilter1_library=/opt/SUNWmsgsr/lib/libspamass.so
spamfilter1_optional=1
spamfilter1_string_action=data:,require "fileinto"; fileinto "$U";
```

spamfilter1_config_file은 SpamAssassin 구성 파일을 지정합니다.

spamfilter1_library는 SpamAssassin 공유 라이브러리를 지정합니다.

spamfilter1_optional=1은 spamd에 의한 장애가 있더라도 MTA가 작업을 계속하도록 지정합니다.

spamfilter1_string_action은 스팸 메시지에 대해 수행할 시브(Sieve) 작업을 지정합니다.

이 예에서는 기본값이 이미 data:,require "fileinto"; fileinto "$U";로 설정되어 있기 때문에 spamfilter1_string_action은 필요하지 않습니다. 이 행은 스팸 메시지를 폴더로 보내도록 지정합니다. 폴더 이름은 SpamAssassin에서 반환되는 스팸 답신 값입니다. SpamAssassin에서 반환되는 값은 spamassassin.opt의 verdict 옵션에 의해 지정됩니다. 단계 1을 참조하십시오. 이 경우 폴더 이름은 spam입니다.

3 필터링할 메시지를 지정합니다.

로컬 메시지 저장소로 들어오는 모든 메시지를 필터링하려면 `ims-ms` 채널에 `destinationspamfilter` *X*`optin spam` 키워드를 추가하여 `imta.cnf` 파일을 변경합니다.

```
!
! ims-ms
ims-ms defragment subdirs 20 notices 1 7 14 21 28 backoff "pt5m" "pt10m"
"pt30m" "pt1h"  "pt2h" "pt4h" maxjobs 4 pool IMS_POOL fileinto
$U+$S@$D destinationspamfilter1optin spam
ims-ms-daemon
```

4 구성을 다시 컴파일하고 서버를 다시 시작합니다. MTA만 다시 시작해야 합니다. `stop-msg`는 실행할 필요가 없습니다.

```
# imsimta cnbuild
# imsimta restart
```

5 `spamd` 데몬을 시작합니다. 이 작업은 다음과 같은 일반적인 명령 형식을 사용하여 수행됩니다.

```
spamd -d
```

`spamd` 기본값은 로컬 시스템으로부터의 연결만 허용하는 것입니다. SpamAssassin과 Messaging Server가 서로 다른 시스템에서 실행 중인 경우 이 구문이 필요합니다.

`spamd -d -i` *listen_ip_address* `-A` *allowed_hosts*

여기서 *listen_ip_address*는 수신할 주소이고 *allowed_hosts*는 이 `spamd` 인스턴스에 연결할 수 있는 인증된 호스트 또는 네트워크(IP 주소 사용)의 목록입니다.

주 – `spamd`가 모든 주소에서 수신하게 하려면 0.0.0.0을 `-i` *listen_ip_address*와 함께 사용할 수 있습니다. 시스템의 IP 주소를 변경할 때 `spamfilter` *X*`_verdict_`*n*으로 인해 명령 스크립트를 변경할 필요가 없기 때문에 모든 주소를 수신하는 것이 좋습니다.

▼ 스팸 메시지에 SpamAssassin 점수가 포함된 헤더 추가

이 예에서는 SpamAssassin에 의해 스팸으로 확인된 메시지에 `Spam-test:` *result string* 헤더를 추가합니다. 다음은 헤더 예입니다.

```
Spam-test: True ; 7.3 / 5.0
```

여기서 `Spam-test:`는 리터럴이고 그 뒤의 모든 항목은 결과 문자열입니다. `True`는 스팸임을 의미하고 `false`는 스팸이 아님을 의미합니다. `7.3`은 SpamAssassin 점수입니다. `5.0`은 임계값입니다. 이 결과는 특정 점수 이상 또는 사이의 메일을 파일로 저장하거나 삭제할 수 있는 시브(Sieve) 필터를 설정할 때 유용합니다.

또한, `USE_CHECK`를 `0`으로 설정하면 답신 문자열과 함께 일치하는 SpamAssassin 테스트 목록이 반환됩니다. 표 14–3의 USE_CHECK를 참조하십시오.

1 **필터링할 메시지를 지정합니다. 이 내용은 451 페이지 "스팸을 별도의 폴더에 정리"의 단계 3에 설명되어 있습니다.**

2 **SpamAssassin 구성 파일을 만듭니다.**

이 파일의 이름과 위치는 `spamfilter_config`*X*`_file`을 사용하여 지정합니다(다음 단계 참조). 이 파일은 다음과 같은 행으로 구성되어 있습니다.

```
host=127.0.0.1
port=2000
mode=1
field=
debug=1
```

`host` 및 `port`는 `spamd`가 실행 중인 시스템의 이름과 `spamd`가 받는 요청을 수신하는 포트를 지정합니다. `mode=1`은 메시지가 스팸으로 확인되는 경우 SpamAssassin 결과 문자열을 반환하도록 지정합니다. `field=`는 SpamAssasin 결과 문자열의 문자열 접두어를 지정합니다. 이 예에서는 시브(Sieve) 스크립트로 지정하기 때문에 접두어가 필요하지 않습니다. `debug=1`은 SpamAssassin 라이브러리에서 디버깅을 설정합니다.

3 **`option.dat` 파일에 다음 행을 추가합니다.**

```
!for Spamassassin
spamfilte1r_config_file=/opt/SUNWmsgsr/config/spamassassin.opt
spamfilter1_library=/opt/SUNWmsgsr/lib/libspamass.so
spamfilter1_optional=1
spamfilter1_string_action=data:,require ["addheader"];addheader "Spam-test: $U";
```

앞의 예와 마찬가지로 처음 세 옵션은 SpamAssassin 구성 파일과 공유 라이브러리를 지정하고 공유 라이브러리에 오류가 있을 경우 MTA 작업을 계속하도록 지정합니다. 다음 행은

```
spamfilter1_string_action=data:,require ["addheader"];addheader "Spam-test:
$U";
```

스팸 메시지에 헤더를 추가하도록 지정합니다. 헤더에는 SpamAssassin에서 반환되는 문자열 앞에 Spam-text:라는 문자 접두어가 붙습니다. 이전 단계에서 mode=1로 지정했기 때문에 SpamAssassin 결과 문자열이 반환됩니다. 예를 들면 다음과 같습니다. True; 7.3/5.0입니다.

4 **구성을 다시 컴파일하고 서버를 다시 시작한 다음 `spamd` 데몬을 시작합니다.**

450 페이지 "14.4.5 SpamAssassin 구성 예"를 참조하십시오.

▼ SpamAssassin 결과 문자열을 제목 줄에 추가

SpamAssassin 결과 문자열을 제목 줄에 추가하여 SpamAssassin 점수로 메시지를 읽을지 여부를 결정할 수 있습니다. 예를 들면 다음과 같습니다.

Subject: [SPAM True ; 99.3 / 5.0] Free Money At Home with Prescription Xanirex!

USE_CHECK를 0으로 설정하면 답신 문자열과 함께 일치하는 SpamAssassin 테스트 목록이 반환됩니다(458 페이지 “14.4.7 SpamAssassin 옵션” 참조). 이 목록의 길이가 매우 길 수 있으므로 USE_CHECK를 1로 설정하는 것이 가장 좋습니다.

1 필터링할 메시지를 지정합니다.

이 내용은 451 페이지 “스팸을 별도의 폴더에 정리”의 단계 3에 설명되어 있습니다.

2 SpamAssassin 구성 파일을 만듭니다.

이 단계는 451 페이지 “스팸을 별도의 폴더에 정리”에 설명되어 있습니다. mode=1은 메시지가 스팸으로 확인되는 경우 SpamAssassin 결과 문자열을 반환하도록 지정합니다.

```
host=127.0.0.1
port=2000
mode=1
debug=1
```

host 및 port는 spamd가 실행 중인 시스템의 이름과 spamd가 받는 요청을 수신하는 포트를 지정합니다. mode=1은 메시지가 스팸으로 확인되는 경우 SpamAssassin 결과 문자열을 반환하도록 지정합니다. debug=1은 SpamAssassin 라이브러리에서 디버깅을 설정합니다.

3 option.dat 파일에 다음 행을 추가합니다.

```
!for Spamassassin
spamfilter1_config_file=/opt/SUNWmsgsr/config/spamassassin.opt
spamfilter1_library=/opt/SUNWmsgsr/lib/libspamass.so
spamfilter1_optional=1
spamfilter1_string_action=data:,addtag “[SPAM detected: $U]”;
```

앞의 예와 마찬가지로 처음 세 옵션은 SpamAssassin 구성 파일과 공유 라이브러리를 지정하고 공유 라이브러리에 오류가 있을 경우 MTA 작업을 계속하도록 지정합니다. 다음 행은

```
spamfilter1_string_action=data:,addtag “[SPAM detected $U]”;
```

Subject: 행에 태그를 추가하도록 지정합니다. SpamAssassin에서 반환되는 “[*result string*]” 앞의 field 문자열(기본값: Spam-Test) 앞에 SPAM detected라는 문자 접두어가 있습니다. 450 페이지 “14.4.5 SpamAssassin 구성 예”에서 mode=1로 지정했기 때문에 SpamAssassin 결과 문자열이 반환됩니다. 따라서, 제목 줄은 다음과 비슷합니다. 따라서, 제목 줄은 다음과 비슷합니다.

```
Subject: [SPAM detected Spam-Test: True ; 11.3 / 5.0] Make Money!
```

다음과 같이 addheader와 addtag를 함께 사용할 수도 있습니다.

```
spamfilter1_string_action=data:,require ["addheader"];addtag "[SPAM detected
$U]";addheader "Spamscore: $U";
```

다음과 비슷한 메시지를 볼 수도 있습니다.

```
Subject: [SPAM detected Spam-Test: True ; 12.3 / 5.0] Vigaro Now!Spamscore:
Spam-Test: True ; 12.3 / 5.0
```

spamassassin.opt에서 field=를 설정하여 Spam-Test의 기본값을 제거합니다. 더 명확한 메시지가 반환됩니다.

```
Subject: [SPAM True ; 91.3 / 5.0] Vigaro Now!Spamscore: True ; 91.3 / 5.0
```

4 구성을 다시 컴파일하고 서버를 다시 시작한 다음 spamd 데몬을 시작합니다.

451 페이지 "스팸을 별도의 폴더에 정리"를 참조하십시오.

▼ SpamAssassin 점수를 기준으로 메시지를 필터링하는 방법

이 예에서는 SpamAssassin 점수를 기준으로 메시지를 필터링하는 방법을 보여줍니다. 여기서는 spamadjust 및 spamtest 시브(Sieve) 필터 작업을 사용합니다. 이 예에서는 SpamAssassin 점수를 포함하는 헤더가 모든 메시지에 추가됩니다. SpamAssassin 소프트웨어 관리자는 이 헤더를 사용하여 SpamAssassin을 조정하고 스팸 전자 메일 감지 성능을 향상시킬 수 있습니다. 메시지의 SpamAssassin 점수가 5에서 10까지인 경우에는 메시지가 사용자 계정의 spam 폴더에 필터링됩니다. 메시지의 SpamAssassin 점수가 10보다 큰 경우에는 메시지가 삭제됩니다. 기본적으로 SpamAssassin에서는 점수가 5 이상인 메시지를 스팸으로 간주합니다.

1 필터링할 메시지를 지정합니다.

이 내용은 451 페이지 "스팸을 별도의 폴더에 정리"의 단계 3에 설명되어 있습니다.

2 SpamAssassin 구성 파일을 만듭니다.

이 파일의 이름과 위치는 spamfilter_configX_file을 사용하여 지정합니다(다음 단계 참조). 이 파일은 다음과 같은 행으로 구성되어 있습니다.

```
debug=1
host=127.0.0.1
port=783
mode=2
field=
```

host 및 port는 spamd가 실행 중인 시스템의 이름과 spamd가 받는 요청을 수신하는 포트를 지정합니다. mode=2는 점수에 관계 없이 항상 SpamAssassin 결과 문자열을 반환하도록 지정합니다. field=는 SpamAssasin 결과 문자열의 문자열 접두어를 지정합니다. 이 예에서는 시브(Sieve) 스크립트로 지정하기 때문에 접두어가 필요하지 않습니다. debug=1은 SpamAssassin 라이브러리에서 디버깅을 설정합니다.

3 option.dat 파일에 다음 행을 추가합니다.

```
! For SpamAssassin
spamfilter1_config_file=/opt/SUNWmsgsr/config/spamassassin.opt
spamfilter1_library=/opt/SUNWmsgsr/lib/libspamass.so
```

```
spamfilter1_optional=1
spamfilter1_string_action=data:, require ["addheader","spamtest"]; \
spamadjust "$U"; addheader "Spam-test: $U"
```

앞의 예와 마찬가지로 처음 세 행은 SpamAssassin 구성 파일과 공유 라이브러리를 지정하고 공유 라이브러리에 오류가 있을 경우 MTA 작업을 계속하도록 지정합니다. 마지막 두 행에서는 `spamtest` 작업에 사용되는 SpamAssassin(`$U`) 반환 문자열에서 SpamAssassin 점수를 추출하고 모든 메시지에 스팸 점수 헤더를 추가하도록 지정합니다(예: `Spam-test: True; 7.3/5.0`).

4 **스팸 점수를 기준으로 전자 메일을 처리하는 채널 수준 필터를 만듭니다.**

540 페이지 “채널 수준 필터 만들기”를 참조하십시오. 해당 파일에 다음 규칙을 추가합니다.

```
require ["spamtest","relational","comparator-i;ascii-numeric","fileinto"];
if spamtest :value "ge" :comparator "i;ascii-numeric" "10" {discard;}
elsif spamtest :value "ge" :comparator "i;ascii-numeric" "5" {fileinto "spam";}
else {keep;}
```

두 번째 행에서는 SpamAssassin 점수가 10 이상인 경우 스팸 전자 메일을 삭제합니다. 세 번째 행에서는 점수가 5 이상인 경우 전자 메일을 사용자의 "spam" 폴더에 파일로 보관합니다. 마지막 행 `else {keep;}`에서는 점수가 5 미만인 모든 메시지를 보관합니다.

5 **구성을 다시 컴파일하고 서버를 다시 시작한 다음 `spamd` 데몬을 시작합니다.**

451 페이지 “스팸을 별도의 폴더에 정리”의 마지막 단계를 참조하십시오.

14.4.6 SpamAssassin 테스트

SpamAssassin을 테스트하려면 `spamassassion.opt` 파일에서 `debug=1`을 설정합니다. `imta.cnf`에서 채널별 `master_debug` 또는 `slave_debug`를 설정할 필요는 없습니다. 그런 다음 테스트 사용자에게 테스트 메시지를 보냅니다. *msg-svr-base* `/data/log/tcp_local_slave.log*` 파일은 다음과 비슷한 행으로 구성됩니다.

```
15:15:45.44: SpamAssassin callout debugging enabled; config
/opt/SUNWmsgsr/config/spamassassin.opt
15:15:45.44: IP address 127.0.0.1 specified
15:15:45.44: Port 2000 selected
15:15:45.44: Mode 0 selected
15:15:45.44: Field "Spam-Test: " selected
15:15:45.44: Verdict "spam" selected
15:15:45.44: Using CHECK rather than SYMBOLS
15:15:45.44: Initializing SpamAssassin message context
...
15:15:51.42: Creating socket to connect to SpamAssassin
15:15:51.42: Binding SpamAssassin socket
15:15:51.42: Connecting to SpamAssassin
```

```
15:15:51.42: Sending SpamAssassin announcement
15:15:51.42: Sending SpamAssassin the message
15:15:51.42: Performing SpamAssassin half close
15:15:51.42: Reading SpamAssassin status
15:15:51.67: Status line: SPAMD/1.1 0 EX_OK
15:15:51.67: Reading SpamAssassin result
15:15:51.67: Result line: Spam: False ; 1.3 / 5.0
15:15:51.67: Verdict line: Spam-Test: False ; 1.3 / 5.0
15:15:51.67: Closing connection to SpamAssassin
15:15:51.73: Freeing SpamAssassin message context
```

로그 파일이 위와 비슷한 행으로 구성되어 있지 않거나 spamd가 실행되고 있지 않은 경우 마지막 마침표(.)가 SMTP 서버로 보내진 후에 SMTP 대화 상자에 다음 오류 메시지가 반환됩니다.

```
452 4.4.5 Error writing message temporaries - Temporary scan failure: End
message status = -1
```

또한, spamfilter1_optional=1(권장)이 option.dat에 설정되어 있는 경우 메시지가 허용되지만 필터링되지는 않습니다. 스팸 필터링이 활성화되지 않은 경우와 동일하며 tcp_local_slave.log*에 다음 행이 나타납니다.

```
15:35:15.69: Creating socket to connect to SpamAssassin
15:35:15.69: Binding SpamAssassin socket
15:35:15.69: Connecting to SpamAssassin
15:35:15.69: Error connecting socket: Connection refused
15:35:15.72: Freeing SpamAssassin message context
```

SMTP 서버에서 전체 메시지를 수신한 후(마지막 "."가 SMTP 서버로 보내진 후), SMTP 서버가 메시지를 수락했음을 보낸 사람이 알기 전에 SpamAssassin이 호출됩니다.

Mail-SpamAssassin-2.60과 같은 디렉토리에서 sample-spam.txt를 사용하여 샘플 스팸 메시지를 보내는 다른 테스트를 수행합니다. 이 메시지에는 내부에 다음과 같은 특수 텍스트 문자열이 있습니다.

```
XJS*C4JDBQADN1.NSBN3*2IDNEN*GTUBE-STANDARD-ANTI-UBE-TEST-EMAIL*C.34X
```

해당 tcp_local_slave.log*는 다음과 같은 내용으로 구성되어 있습니다.

```
16:00:08.15: Creating socket to connect to SpamAssassin
16:00:08.15: Binding SpamAssassin socket
16:00:08.15: Connecting to SpamAssassin
16:00:08.15: Sending SpamAssassin announcement
16:00:08.15: Sending SpamAssassin the message
16:00:08.15: Performing SpamAssassin half close
16:00:08.15: Reading SpamAssassin status
16:00:08.43: Status line: SPAMD/1.1 0 EX_OK
```

```
16:00:08.43: Reading SpamAssassin result
16:00:08.43: Result line: Spam: True ; 1002.9 / 5.0
16:00:08.43: Verdict line: Spam-Test: True ; 1002.9 / 5.0
16:00:08.43: Closing connection to SpamAssassin
16:00:08.43: Mode 0 verdict of spam
16:00:08.43: Mode 0 verdict of spam
16:00:08.47: Freeing SpamAssassin message context
```

mail.log_current 파일의 해당 항목은 다음과 비슷합니다. 메시지가 spam이라는 폴더에 정리되어 있음을 의미하는 메일 주소의 +spam 부분에 주의하십시오.

```
15-Dec-2003 15:32:17.44 tcp_intranet ims-ms E 1 morchia@siroe.com rfc822;
morchia morchia+spam@ims-ms-daemon 15-Dec-2003 15:32:18.53
ims-ms D 1 morchia@siroe.com rfc822;morchia morchia+spam@ims-ms-daemon
```

14.4.7 SpamAssassin 옵션

이 절에는 SpamAssassin 옵션 표가 포함되어 있습니다.

표 14–3 SpamAssassin 옵션(spamassassin.opt)

옵션	설명	기본값
debug	libspamass.so에서 디버깅을 설정할지 여부를 지정합니다. spamd에 대한 디버깅은 spamd를 호출하는 명령줄에 의해 제어됩니다. 정수 값으로 설정합니다. 0은 설정 해제를, 1은 설정을 나타내며, 2 이상인 설정 값은 spamd에서 받은 내용을 그대로 보고합니다.	0
field	SpamAssassin 결과 문자열 접두어를 지정합니다. SpamAssassin 결과는 다음과 비슷합니다. `Spam-Test: False ; 0.0 / 5.0 Spam-Test: True ; 27.7 / 5.0` field 옵션을 사용하면 결과의 Spam-Test: 부분을 변경할 수 있습니다. 빈 field 값을 지정하는 경우 ":"이 제거된다는 점을 유의하십시오. *USE_CHECK*를 0으로 설정하면 다음과 비슷한 결과 문자열이 표시됩니다. Spam-test: False ; 0.3 / 4.5 ; HTML_MESSAGE,NO_REAL_NAME Spam-test: True ; 8.8 / 4.5 ; NIGERIAN_BODY, NO_REAL_NAME,PLING_PLING,RCVD_IN_SBL,SUBJ_ALL_CAPS	"Spam-test"
host	spamd가 실행 중인 시스템의 이름입니다.	localhost

표 14–3 SpamAssassin 옵션(spamassassin.opt) *(계속)*

옵션	설명	기본값
mode	SpamAssassin 필터 결과를 답신 정보로 변환하는 것을 제어합니다. 즉, 메시지를 처리한 후 반환되는 답신 정보를 지정합니다. 다음과 같은 네 가지 모드를 사용할 수 있습니다. 자세한 내용은 460 페이지 "14.4.7.1 SpamAssassin mode 옵션"을 참조하십시오. 0 - 메시지가 스팸일 경우 **답신 문자열**(verdict 옵션에 의해 지정됨)을 반환합니다. MTA 옵션 spamfilter *X*_string_action은 verdict 문자열이 반환될 경우 작업할 내용을 지정하는 데 사용될 수 있습니다. 아래에 정의된 verdict 옵션이 비어 있거나 지정되지 않았고 메시지가 스팸이면 *null* **답신**이 반환됩니다. MTA 옵션 spamfilter*X*_null_action은 null 답신이 반환될 경우에 수행할 작업을 지정합니다. 스팸이 아닌 경우 *SpamAssassin* **기본 답신 문자열**을 반환합니다. 기본 답신은 항상 작업을 수행하지 않고 정상적으로 전달됨을 의미합니다. 1 - 메시지가 스팸인 경우 SpamAssassin **결과 문자열**을 반환합니다. 스팸이 아닌 경우 *SpamAssassin* **기본 답신 문자열**을 반환합니다. 기본 답신은 항상 작업을 수행하지 않고 정상적으로 전달됨을 의미합니다. SpamAssassin 결과 문자열은 True; 6.5 / 7.3과 비슷합니다. 2- 메시지가 스팸인지 여부에 관계 없이 SpamAssassin 결과 문자열이 반환된다는 점을 제외하고 모드 1과 동일합니다. 기본 또는 null 답신이 반환되지 않으며 verdict 옵션이 사용되지 않습니다. 3 - 메시지가 스팸인 경우에는 SpamAssassin 결과 문자열을 반환하고 스팸이 아닌 경우에는 verdict 옵션에 지정된 verdict 문자열을 반환합니다. spamfilter*X*_verdict_*n* 및 spamfilter*X*_action_*n* 일치 쌍을 사용하여 SpamAssassin 결과 문자열의 작업을 제어할 수 있습니다. spamfilter*X*_string_action을 사용하여 verdict 문자열의 작업을 제어할 수 있습니다.	0
port	spamd가 수신 요청을 수신하는 포트 번호입니다.	783
USE_CHECK	1 - spamd CHECK 명령을 사용하여 SpamAssassin 점수를 반환합니다. 0 - SYMBOLS 명령을 사용하여 일치하는 SpamAssassin 테스트의 점수와 목록을 반환합니다. 2.55 이전 버전 SpamAssassin에서는 이 옵션을 사용하면 시스템이 중단되거나 다른 문제가 발생할 수 있습니다. 위의 *field*를 참조하십시오.	
SOCKS_HOST	문자열입니다. 중간 SOCKS 서버의 이름을 지정합니다. 이 옵션을 지정하면 ICAP가 직접 연결되지 않고 지정된 SOCKS 서버를 통해 연결됩니다.	""
SOCKS_PORT	중간 SOCKS 서버가 실행 중인 포트를 지정합니다.	1080
SOCKS_PASSWORD	SOCKS 서버를 통해 연결을 설정할 때 사용할 비밀번호(문자열)를 지정합니다. 아이디/비밀번호가 필요한지 여부는 SOCKS 서버 구성에 따라 다릅니다.	""
SOCKS_USERNAME	SOCKS 서버를 통해 연결을 설정할 때 사용할 아이디(문자열)를 지정합니다.	""

표 14–3 SpamAssassin 옵션(spamassassin.opt) (계속)

옵션	설명	기본값
USERNAME_MAPPING	플러그 인이 MTA로부터 수신자 주소를 받을 때 주소 정보로 검사할 매핑의 이름을 지정합니다. 검사 형식은 다음과 같습니다. *current-username*|*current-recipient-address*|*current-optin-string* 매핑에서 $Y 플래그를 설정할 경우 출력 문자열은 spamd에 전달할 업데이트된 사용자 이름으로 간주됩니다.	""
verdict	MODE 0에 사용되는 답신 문자열을 지정합니다.	“”

14.4.7.1 SpamAssassin mode 옵션

메시지를 처리한 후, SpamAssassin에서 메시지가 스팸인지 아닌지를 파악합니다. mode를 사용하면 답신을 나타내도록 반환되는 문자열을 지정할 수 있습니다. 선택할 수 있는 옵션으로는 null이나 기본값, SpamAssassin 결과 문자열, verdict 옵션으로 지정된 verdict 문자열이 있습니다. (**기본값**은 null이나 SpamAssassin 결과 문자열, verdict에서 지정한 문자열 중 어느 것도 아니지만 기타 비구성 결과 문자열이라는 점에 유의하십시오.)아래 테이블은 mode 작업에 대한 개요입니다.

표 14–4 SpamAssassin mode 옵션의 문자열 반환

verdict\ 설정	스팸	mode=0	mode=1	mode=2	mode=3
verdict="" (설정되지 않음)	예	null	SpamAssassin 결과	SpamAssassin 결과	SpamAssassin 결과
	아니요	기본값	기본값	SpamAssassin 결과	기본값
verdict=string	예	verdict **문자열**	SpamAssassin 결과	SpamAssassin 결과	SpamAssassin 결과
	아니요	기본값	기본값	SpamAssassin 결과	verdict **문자열**

첫 번째 열은 verdict 옵션이 설정되어 있는지 여부를 나타냅니다. 두 번째 열은 해당 메시지가 스팸인지 여부를 나타냅니다. 모드 열은 여러 모드에 문자열이 반환되었음을 나타냅니다. 예를 들어, verdict가 설정되어 있지 않고 mode가 0으로 설정되어 있으며 메시지가 스팸이 아닌 경우에는 기본 문자열이 반환됩니다. verdict가 YO SPAM!으로, mode가 0으로 설정되어 있고 메시지가 스팸인 경우에는 YO SPAM! 문자열이 반환됩니다.

14.5 SAVSE(Symantec Anti-virus Scanning Engine) 사용

이 절에서는 SAVSE를 배포하는 방법에 대해 설명하지만 다른 ICAP 지원 스팸 방지/바이러스 백신 프로그램을 배포하는 데에도 도움이 됩니다. 이 절은 다음과 같은 하위 절로 구성되어 있습니다.

14.5.1 SAVSE 개요

SAVSE는 TCP/IP 서버 응용 프로그램 및 통신 API(Application Programming Interface)로서, 바이러스 스캐닝 서비스를 제공합니다. 네트워크 인프라 장치를 통해 서비스되거나 저장되는 트래픽을 방지하도록 특별히 설계되었기 때문에 모든 주요 파일 형식(모바일 코드와 압축 파일 형식 포함)에 있는 바이러스와 웜, 트로이 목마 등을 감지하고 방지합니다. 자세한 내용은 Symantec 웹 사이트를 참조하십시오.

주 – 현재 버전의 Messaging Server는 SAVSE 스캔 기능만 지원합니다. 복구를 지원하거나 기능을 삭제하지 않습니다.

14.5.2 SAVSE 요구 사항 및 사용 시 고려 사항

이는 Symantec으로부터 별도의 라이센스를 받은 제품입니다.

SAVSE 구성에서는 스캔 모드만 지원되며 스캔과 복구 모드 또는 스캔과 삭제 모드는 지원되지 않습니다.

14.5.2.1 SAVSE 실행 위치

SAVSE 또는 ICAP를 지원하는 다른 서버는 자체의 별도 시스템, 단일 시스템 배포의 Messaging Server와 동일한 시스템 또는 2계층 배포의 MTA와 동일한 시스템에서 실행할 수 있습니다. LMTP(Local Mail Transfer Protocol)가 MTA와 메시지 저장소 사이에서 사용되는 경우 MTA에서 필터링을 호출해야 합니다. 메시지 저장소에서는 필터링을 호출할 수 없습니다. MTA와 메시지 저장소 간에 SMTP가 사용되는 경우 각 시스템이나 별도의 타사 시스템에서 실행할 수 있습니다.

SAVSE를 실행하는 서버 그룹을 사용하려면 해당 그룹의 앞에 로드 밸런서를 사용해야 합니다. MTA는 SAVSE 서버에 대해 단일 주소로만 구성됩니다.

14.5.3 SAVSE 배포

SAVSE를 배포하려면 다음 단계를 수행합니다.

- **SAVSE를 설치하고 구성합니다.** 설치 및 구성 정보는 Symantec 소프트웨어 설명서를 참조하십시오. 464 페이지 "14.5.5 SAVSE 옵션"을 참조하십시오.
- **SAVSE 클라이언트 라이브러리를 로드 및 구성합니다.** 이 작업에는 클라이언트 라이브러리 `libicap.so`와 MTA에 대한 구성 파일(만들어야 함)을 지정하는 것이 포함됩니다. 433 페이지 "14.2.1 스팸 필터링 소프트웨어 클라이언트 라이브러리 로드 및 구성"을 참조하십시오.
- **바이러스를 필터링할 메시지를 지정합니다.** 사용자, 도메인 또는 채널별로 메시지를 필터링할 수 있습니다. 434 페이지 "14.2.2 필터링할 메시지 지정"을 참조하십시오.
- **바이러스 메시지에 대해 수행할 작업을 지정합니다.** 바이러스 삭제, 폴더에 정리, 제목 줄에 태그 지정 등의 작업을 수행할 수 있습니다. 439 페이지 "14.2.3 스팸 메시지에 대해 수행할 작업 지정"을 참조하십시오.
- **기타 필터 구성 매개 변수를 원하는 대로 설정합니다.** 표 14–1 439 페이지 "14.2.3 스팸 메시지에 대해 수행할 작업 지정"을 참조하십시오.

14.5.4 SAVSE 구성 예

다음 예에서는 로컬 메시지 저장소에서 받는 메시지를 테스트하여 바이러스가 첨부된 메시지를 삭제합니다. 처음 세 단계는 순서에 관계없이 수행할 수 있습니다.

▼ SAVSE 구성 방법

1 SAVSE 구성 파일을 만듭니다.

이 파일의 이름과 위치는 다음 단계에서 지정합니다. 여기에서 사용되는 이름은 `SAVSE.opt`입니다. 다음은 이 파일의 예입니다.

```
host=127.0.0.1
port=1344
mode=0
verdict=virus
debug=1
```

`host` 및 `port`는 SAVSE 프로그램이 실행 중인 시스템의 이름과 이 프로그램이 수신 요청을 수신하는 포트(SAVSE의 기본 포트는 1344임)를 지정합니다. `mode=0`은 메시지가 스팸으로 간주될 때 `verdict`에 지정된 문자열(이 경우 단어 `virus`)이 반환되도록 지정합니다. `debug=1`은 디버깅을 설정합니다. ICAP 구성 매개 변수에 대한 설명은 464 페이지 "14.5.5 SAVSE 옵션"을 참조하십시오.

2 option.dat 파일을 만듭니다. 예:

```
! for Symantex Anti-virus Scan Engine
spamfilter1_config_file=/opt/SUNWmsgsr/config/SAVSE.opt
spamfilter1_library=/opt/SUNWmsgsr/lib/libicap.so
spamfilter1_optional=1
spamfilter1_string_action=data:,discard
```

spamfilter1_config_files는 SAVSE 구성 파일을 지정합니다.

spamfilter1_library는 SAVSE 공유 라이브러리의 위치를 지정합니다.

spamfilter1_optional=1은 SAVSE 프로그램에 오류가 있더라도 MTA가 작업을 계속하도록 지정합니다.

spamfilter1_string_action은 스팸 메시지에 대해 수행할 시브(Sieve) 작업을 지정합니다. 이 값은 바이러스가 있는 메시지를 삭제하도록 지정하며이 값은 기본값이기 때문에 값을 바꾸지 않는 한 지정할 필요가 없습니다.

3 **필터링할 메시지를 지정합니다.**

로컬 메시지 저장소로 들어오는 모든 메시지를 필터링하려면 ims-ms 채널에 destinationspamfilter1optin spam 키워드를 추가하여 imta.cnf 파일을 변경합니다.

```
!
! ims-ms
ims-ms defragment subdirs 20 notices 1 7 14 21 28 backoff "pt5m" "pt10m"
"pt30m" "pt1h"  "pt2h" "pt4h" maxjobs 4 pool IMS_POOL fileinto
$U+$S@$D destinationspamfilter1optin virus
ims-ms-daemon
```

4 **구성을 다시 컴파일하고 서버를 다시 시작합니다. MTA만 다시 시작해야 합니다.** stop-msg**는 실행할 필요가 없습니다.**

```
# imsimta cnbuild
# imsimta restart
```

5 **SAVSE가 시작되는지 확인합니다.**

자동으로 시작되어야 하지만, 그렇지 않은 경우는 **/etc/init.d/symcscna start**

14.5.4.1 가능한 다른 구성

mode를 0으로 설정하면 spamfilterX_null_option을 사용하여 스팸으로 확인된 메시지를 특정 폴더에 정리하는 등의 다른 작업을 수행하도록 지정할 수 있습니다. 예를 들면 다음과 같습니다.

```
spamfilter1_null_option=data:,require "fileinto"; fileinto "VIRUS";
```

감염된 메시지를 폴더에 정리하는 것은 대부분의 경우 좋지 않습니다.

mode를 1로 설정하여 작업을 시작할 수도 있습니다. 예를 들어, mode를 1로 설정하고 MTA에서 spamfilter*X*_string_action 옵션을 다음과 같이 설정하여 스팸 결과를 거부 메시지에 포함할 수 있습니다.

```
spamfilter1_string_action=data:,require "reject"; reject "Message contained a
virus [$U]";
```

fileinto와 마찬가지로 reject 작업을 사용하여 바이러스를 처리하는 것은 바이러스가 보낸 사람에게 다시 전송되므로 좋은 방법이 아닙니다.

option.dat 파일에 한 행을 추가하여 스팸 메시지 헤더에 태그를 추가할 수도 있습니다. 예:

```
spamfilter1_string_action=data:,addtag “[SPAM detected!]”;
```

메시지에 바이러스가 있는 것으로 확인되었는지 여부에 상관없이 작업을 수행해야 할 경우에는 mode를 2로 설정할 수 있습니다. 나중에 테스트할 수 있는 헤더 필드를 추가하는 것을 모드 2 적용의 예로 볼 수 있습니다.

```
spamfilterX_string_action=data:,require ["addheader"];addheader "$U"
```

14.5.5 SAVSE 옵션

SAVSE 옵션 파일은 아주 일반적인 ICAP 옵션 파일입니다. 이 파일의 이름과 위치는 option.dat의 spamfilter*X*_config_file에 설정합니다. 이 파일은 option=value 형태의 행으로 구성됩니다. 유일한 필수 옵션은 HOST입니다. 이 옵션은 ICAP 필터링 서버가 실행 중인 시스템의 이름으로 설정해야 합니다. ICAP 서버가 로컬 호스트에서 실행 중인 경우에도 이 옵션을 설정해야 합니다. 이 옵션 파일은 아래와 같습니다.

표 14–5 ICAP 옵션

옵션	설명	기본값
debug	ICAP 인터페이스 모듈에서 디버그 출력을 활성화하거나 비활성화합니다. 0 또는 1입니다.	0
field	ICAP 결과 접두어를 지정합니다. SAVSE 결과 문자열은 다음과 비슷합니다. Virus-Test: False Virus-Test: True; W32.Mydoom.A@mm.enc 이 옵션은 결과의 Virus-Test: 부분을 변경할 수 있는 방법을 제공합니다. 빈 field 값을 지정하는 경우 “:“이 제거된다는 점을 유의하십시오.	Virus-test
host	ICAP 필터링 서버가 실행 중인 시스템의 이름입니다.	localhost

표 14–5 ICAP 옵션 *(계속)*

옵션	설명	기본값
mode	ICAP 필터 결과를 답신 정보로 변환하는 것을 제어합니다. 즉, 메시지를 처리한 후 반환되는 문자열 정보를 지정합니다. 다음과 같은 네 가지 모드를 사용할 수 있습니다. 자세한 내용은 466 페이지 "14.5.5.1 ICAP mode 옵션"을 참조하십시오. 0 - 메시지에 바이러스가 있는 경우 **답신 문자열**(verdict 옵션에 의해 지정됨)을 반환합니다. MTA 옵션 spamfilter*X*_string_action은 verdict 문자열이 반환될 경우 작업할 내용을 지정하는 데 사용될 수 있습니다. verdict 옵션이 비어 있거나 지정되지 않은 경우 *null* **답신**이 반환됩니다. MTA 옵션 spamfilter *X*_null_action은 기본 작업을 무시함으로써 메시지를 버리게 될 경우와 null 답신이 반환될 경우 수행할 작업을 지정하는 데 사용될 수 있습니다. 메시지에 바이러스가 있지 않은 경우 기본 답신이 반환됩니다. 기본 문자열은 구성할 수 없으며 항상 작업을 수행하지 않고 정상적으로 전달됨을 의미합니다. 1 - 메시지에 바이러스가 있을 경우 ICAP **결과 문자열**을 반환합니다. 메시지에 바이러스가 있지 않은 경우 기본 답신이 반환됩니다. 기본 문자열은 항상 작업을 수행하지 않고 정상적으로 전달됨을 의미합니다. 다음은 SAVSE 결과 문자열의 두 가지 예입니다. VIRUS TEST: FALSEVIRUS-TEST: TRUE; W32.Mydoom.A@mm.enc 2 - 무조건적으로 ICAP 결과 문자열을 답신으로 반환합니다. 기본 또는 null 답신이 반환되지 않으며 verdict 옵션이 사용되지 않습니다. 이 설정은 메시지에 바이러스가 있는 것으로 확인되었는지 여부에 상관없이 작업을 수행해야 할 경우에 사용됩니다. 나중에 테스트할 수 있는 헤더 필드를 추가하는 것을 모드 2 적용의 예로 볼 수 있습니다. spamfilter*X*_string_action=data:,require ["addheader"];addheader "$U" 3 - 메시지에 바이러스가 있는 경우 ICAP 결과 문자열이 반환되고, 그렇지 않으면 verdict 옵션에서 지정한 verdict 문자열이 반환됩니다. 이 설정은 바이러스가 확인된 경우 특정 작업을 수행하고 그렇지 않은 경우 다른 작업을 수행해야 하는 경우에 사용됩니다. spamfilter*X*_verdict_*n* 및 spamfilter*X*_action_*n* 일치 쌍을 사용하여 ICAP 결과 문자열의 작업을 제어할 수 있습니다. spamfilter *X*_string_action을 사용하여 verdict 문자열의 작업을 제어할 수 있습니다.	0
port	ICAP 서버가 실행 중인 포트 번호를 지정합니다.	1344
SOCKS_HOST	문자열입니다. 중간 SOCKS 서버의 이름을 지정합니다. 이 옵션을 지정하면 ICAP가 직접 연결되지 않고 지정된 SOCKS 서버를 통해 연결됩니다.	""
SOCKS_PORT	정수입니다. 중간 SOCKS 서버가 실행 중인 포트를 지정합니다.	1080
SOCKS_PASSWORD	문자열입니다. SOCKS 서버를 통해 연결을 설정할 때 사용할 비밀번호를 지정합니다. 아이디/비밀번호가 필요한지 여부는 SOCKS 서버 구성에 따라 다릅니다.	""
SOCKS_USERNAME	문자열입니다. SOCKS 서버를 통해 연결을 설정할 때 사용할 아이디를 지정합니다.	""
verdict	MODE 0과 3에 사용되는 답신 문자열을 지정합니다.	""

14.5.5.1 ICAP mode 옵션

메시지를 처리한 후, SASVE와 같은 ICAP 바이러스 방지 프로그램은 메시지에 바이러스가 있는지 여부를 파악합니다. `mode`를 사용하면 이 답신을 나타내는 ICAP 프로그램에서 반환하는 문자열을 지정할 수 있습니다. 선택할 수 있는 옵션으로는 *null*, *default*, *ICAP* **결과 문자열**, `verdict` **문자열**(`verdict` 옵션으로 지정) 등이 있습니다. **기본값**은 null이나 ICAP 결과 문자열, `verdict`로 지정된 문자열이 아니지만 프로그램에서 반환한 기타 비구성 문자열이라는 점을 유의하십시오. 아래 테이블은 `mode` 작업에 대한 개요입니다.

표 14–6 ICAP 모드 옵션의 답신 문자열 반환

verdict\설정	바이러스	mode=0	mode=1	mode=2	mode=3
verdict="" (설정되지 않음)	예	null	ICAP 결과	ICAP 결과	ICAP 결과
	아니요	기본값	기본값	ICAP 결과	기본값
verdict=string	예	verdict **문자열**	ICAP 결과	ICAP 결과	ICAP 결과
	아니요	기본값	기본값	ICAP 결과	verdict **문자열**

첫 번째 열은 `verdict` 옵션이 설정되어 있는지 여부를 나타냅니다. 두 번째 열은 메시지에 바이러스가 있는지 여부를 나타냅니다. 모드 열은 여러 모드에 문자열이 반환되었음을 나타냅니다. 예를 들어, `verdict`가 설정되어 있지 않고 `mode`가 0으로 설정되어 있으며 메시지에 바이러스가 없을 경우 ICAP 프로그램에서 기본값을 반환합니다. `verdict`가 `WARNING VIRUS!`로, `mode`가 0으로 설정되어 있고 메시지에 바이러스가 있으면 ICAP 프로그램에서 `WARNING VIRUS!`라는 문자열을 반환합니다.

14.6 ClamAV 사용

Messaging Server는 무료로 사용할 수 있는 인기 있는 타사 바이러스 스캐너인 ClamAV를 사용하여 바이러스와 트로이 목마에 감염된 메시지를 감지합니다. 새로 만들어진 바이러스를 감지하기 위해 ClamAV에서 사용하는 바이러스 서명은 ClamAV 소프트웨어 패키지에 제공되는 `freshclam` 유틸리티를 사용하여 자동으로 업데이트할 수 있습니다.

ClamAV에 대한 자세한 내용은 ClamAV 웹 사이트를 참조하십시오.

- 467 페이지 “14.6.1 ClamAV/Messaging Server 작동 원리”
- 467 페이지 “14.6.2 ClamAV 요구 사항 및 사용 시 고려 사항”
- 467 페이지 “14.6.3 ClamAV 배포”
- 468 페이지 “ClamAV를 사용하여 바이러스나 트로이 목마에 감염된 전자 메일의 Jettison 수행”
- 469 페이지 “14.6.4 ClamAV 테스트”
- 470 페이지 “14.6.5 ClamAV 옵션”

14.6.1 ClamAV/Messaging Server 작동 원리

Messaging Server에 통합된 ClamAV에서는 ClamAV 패키지의 일부로 제공되는 `clamd` 데몬을 활용합니다. `clamd`는 소켓에서 메시지 처리 요청을 수신하는 다중 스레드 프로세스입니다. 이 프로세스는 메시지를 처리한 후에 응답을 반환하고 연결을 닫습니다. ClamAV 설치에서 클라이언트 부분인 `clamdscan`은 사용되지 않습니다. 이 기능은 Messaging Server의 일부인 `libclamav.so`라는 공유 라이브러리에서 수행됩니다.

`libclamav.so`는 Brightmail SDK에서와 동일한 방법으로 로드됩니다.

14.6.2 ClamAV 요구 사항 및 사용 시 고려 사항

ClamAV는 자체 시스템, 단일 시스템 배포의 Messaging Server와 동일한 시스템 또는 2계층 배포의 MTA와 동일한 시스템에서 실행할 수 있습니다. LMTP(Local Mail Transfer Protocol)가 MTA와 메시지 저장소 사이에서 사용되는 경우 MTA에서 필터링을 호출해야 합니다. 메시지 저장소에서는 필터링을 호출할 수 없습니다. MTA와 메시지 저장소 사이에서 SMTP를 사용하는 경우 두 곳 모두에서 SMTP를 호출할 수 있습니다.

ClamAV를 실행 중인 서버 그룹을 사용하려면 해당 그룹의 앞에 로드 밸런서를 사용합니다. MTA는 ClamAV 서버에 대해 단일 주소로만 구성됩니다.

기타 고려 사항

- ClamAV는 무료입니다. 소프트웨어와 설명서는 `http://clamav.net`에서 구할 수 있습니다.
- MTA와 통합된 ClamAV는 사용자, 도메인 또는 채널에 대해 사용할 수 있습니다.
- ClamAV 패키지는 바이러스 서명을 정기적으로 업데이트하는 유틸리티를 제공합니다. 이 유틸리티를 `freshclam`이라고 합니다. 자세한 내용은 ClamAV 패키지 설명서를 참조하십시오.
- Messaging Server 2006Q4 이상에는 기본적으로 `libclamav.so` 라이브러리가 포함되어 있습니다.

14.6.3 ClamAV 배포

ClamAV를 배포하려면 다음 단계를 수행합니다.

- **ClamAV를 설치하고 구성합니다.** 설치 및 구성 정보는 ClamAV 소프트웨어 설명서를 참조하십시오. 470 페이지 “14.6.5 ClamAV 옵션”을 참조하십시오.
- **ClamAV 클라이언트 라이브러리를 로드하고 구성합니다.** 이 작업에는 클라이언트 라이브러리 `libclamav.so`와 MTA에 대한 구성 파일(만들어야 함)을 지정하는 것이 포함됩니다. 433 페이지 “14.2.1 스팸 필터링 소프트웨어 클라이언트 라이브러리 로드 및 구성”을 참조하십시오.
- **스팸을 필터링할 메시지를 지정합니다.** 사용자, 도메인 또는 채널별로 메시지를 필터링할 수 있습니다. 434 페이지 “14.2.2 필터링할 메시지 지정”을 참조하십시오.

- **바이러스 메시지에 대해 수행할 작업을 지정합니다.** 439 페이지 “14.2.3 스팸 메시지에 대해 수행할 작업 지정”을 참조하십시오.
- **기타 필터 구성 매개 변수를 원하는 대로 설정합니다.** 470 페이지 “14.6.5 ClamAV 옵션”을 참조하십시오.

▼ ClamAV를 사용하여 바이러스나 트로이 목마에 감염된 전자 메일의 Jettison 수행

다음 예에서는 ClamAV에서 바이러스나 트로이 목마가 포함된 것으로 발견된 모든 메시지에 대해 jettison을 수행합니다. 답신 문자열은 사용되지 않습니다.

1 ClamAV 구성 파일을 만듭니다.

이 파일의 이름과 위치는 단계 2에서 지정됩니다. `clamav.opt`라는 이름을 사용하는 것이 좋습니다. 이 파일은 다음과 같은 행으로 구성되어 있습니다.

```
# more /opt/SUNWmsgsr/config/clamav.opt
! ClamAV Settings
debug=1
host=127.0.0.1
port=3310
mode=1
```

`debug=1`은 ClamAV 라이브러리에서 디버깅을 설정합니다.

`host` 및 `port`는 `clamd`가 실행 중인 시스템의 이름과 `clamd`가 받는 요청을 수신하는 포트를 지정합니다.

`mode=1`은 바이러스에 감염된 전자 메일이 감지된 경우 ClamAV 플러그 인이 ClamAV 결과 문자열을 답신으로 반환하도록 지정합니다.

2 `option.dat` 파일을 수정합니다.

`option.dat` 파일에 다음 행을 추가합니다.

```
! ClamAV settings
spamfilter2_config_file=/opt/SUNWmsgsr/config/clamav.opt
spamfilter2_library=/opt/SUNWmsgsr/lib/libclamav.so
spamfilter2_string_action=data:,require ["jettison"]; jettison;
```

`spamfilter2_config_file`은 ClamAV 구성 파일을 지정합니다.

`spamfilter2_library`는 ClamAV 공유 라이브러리를 지정합니다.

`spamfilter2_string_action`은 바이러스에 감염된 전자 메일에 대해 수행할 시브(Sieve) 작업을 지정합니다.

3 **필터링할 메시지를 지정합니다.**

로컬 메시지 저장소로 들어오는 모든 메시지를 필터링하려면 `ims-ms` 채널에 `destinationspamfilterXoptin` 바이러스 키워드를 추가하여 `imta.cnf` 파일을 변경합니다.

```
!
! ims-ms
ims-ms defragment subdirs 20 notices 1 7 14 21 28 backoff "pt5m" "pt10m"
"pt30m" "pt1h" "pt2h" "pt4h" maxjobs 4 pool IMS_POOL fileinto
$U+$S@$D destinationspamfilter2optin virus
ims-ms-daemon
```

4 **구성을 다시 컴파일하고 서버를 다시 시작합니다.**

MTA만 다시 시작해야 합니다. stop-msg는 실행할 필요가 없습니다.

```
 # imsimta cnbuild
# imsimta restart
```

5 `clamd` **데몬을 시작합니다.**

14.6.4 ClamAV 테스트

ClamAV를 테스트하려면 먼저 `clamav.opt` 파일에서 `debug=1`을 설정합니다. (`imta.cnf`.에서 채널별 `master_debug` 또는 `slave_debug`를 설정할 필요는 없습니다.) 그런 다음 테스트 사용자에게 EICAR 바이러스 문자열(http://www.eicar.org/anti_virus_test_file.htm)이 포함된 파일 첨부를 보냅니다. 이 문자열은 실제 바이러스를 첨부하지 않고도 바이러스 스캐너를 트리거하여 전자 메일을 바이러스에 감염된 것으로 인식할 수 있도록 설계되었습니다.

```
X5O!P%@AP[4\PZX54(P^)7CC)7}$EICAR-STANDARD-ANTIVIRUS-TEST-FILE!$H+H*
```

테스트 로그를 검토합니다. `msg-svr-base/data/log/tcp_local_slave.log*` 파일은 다음과 비슷한 행으로 구성됩니다.

```
10:39:00.85: ClamAV callout debugging enabled;
config /opt/SUNWmsgsr/config/clamav.opt
10:39:00.85: IP address 127.0.0.1 specified
10:39:00.85: Port 3310 selected
10:39:00.85: Mode 1 selected
10:39:00.85: Field "Virus-Test: " selected
10:39:00.85: Verdict "" selected
10:39:00.85: Initializing ClamAV message context
 ...
10:39:00.85: Creating socket to connect to clamd server
10:39:00.85: Binding clamd socket
10:39:00.85: Connecting to clamd server
```

```
10:39:00.85: Sending ClamAV STREAM request
10:39:00.85: Retrieving ClamAV STREAM response
10:39:00.85: STREAM response: PORT 2003
10:39:00.85: Creating socket to connect to clamd server data port
10:39:00.85: Binding clamd data socket
10:39:00.85: Connecting to clamd server data port
10:39:00.85: Sending ClamAV the message
10:39:00.85: Closing ClamAV data connection
10:39:00.85: Reading ClamAV result
10:39:00.87: Result line: stream: Eicar-Test-Signature FOUND
10:39:00.87: Scan result: Message is infected
10:39:00.87: Verdict line: Virus-Test: True ; Eicar-Test-Signature
10:39:00.87: Closing connection to ClamAV
10:39:00.87: Mode 1 verdict of Virus-Test: True ; Eicar-Test-Signature
10:39:00.87: Mode 1 verdict of Virus-Test: True ; Eicar-Test-Signature
 ...
10:39:00.87: Freeing ClamAV message context
```

로그 파일이 위와 비슷한 행으로 구성되어 있지 않거나 clamd가 실행되고 있지 않은 경우 마지막 마침표(.)가 SMTP 서버로 보내진 후에 SMTP 대화 상자에 다음 오류 메시지가 반환됩니다.

```
452 4.4.5 Error writing message temporaries - Error
connecting to ClamAV server
```

14.6.5 ClamAV 옵션

ClamAV 옵션 파일은 option=value 형식의 행으로 구성된 일반적인 메시징 서버 스타일의 옵션 파일입니다. 유일한 필수 옵션은 HOST입니다. 이 옵션은 clamd가 실행 중인 시스템의 이름으로 설정해야 합니다. clamd가 로컬 호스트에서 실행 중인 경우에도 이 옵션을 설정해야 합니다.

이 옵션 파일에 사용할 수 있는 추가 옵션은 다음과 같습니다.

표 14–7 ClamAV 옵션

옵션	설명	기본값
DEBUG	ClamAV 인터페이스 모듈에서 디버그 출력을 활성화하거나 비활성화합니다. (clamd 자체의 디버그 출력은 clamd 명령줄에 있는 옵션에 의해 제어됩니다.)값이 클수록 디버깅 출력이 많아집니다. 0을 설정하면 출력이 없습니다. 1은 기본 디버깅을 제공합니다. 2는 clamd에서 TCP 트래픽 로깅을 추가합니다.	0

표 14–7 ClamAV 옵션 *(계속)*

옵션	설명	기본값

FIELD	ClamAV 결과 문자열 접두어를 지정합니다. ClamAV 결과 문자열은 일반적으로 다음 중 하나와 비슷합니다.
	`Virus-Test: False`
	`Virus-Test: True ; Worm.Mydoom.I`
	`FIELD` 옵션을 사용하면 결과의 `Virus-Test` 부분을 변경할 수 있습니다. 빈 `FIELD` 값을 지정하면 ":"도 제거된다는 점에 주의해야 합니다.
		"Virus-Test"

MESSAGE_BUFFER_SIZE	`clamdscan/clamd` 인터페이스의 특성으로 인해, ClamAV 플러그 인은 ClamAV로 메시지를 보내기 전에 메모리 버퍼에 메시지를 보관해야 합니다. 메모리 버퍼의 크기는 이 옵션에 의해 제어됩니다. 기본값은 1,048,576자입니다. 이 크기보다 긴 메시지는 잘리며 전체가 ClamAV로 전송되지 않습니다. 모든 메시지를 완전히 스캔하려면 이 값은 MTA가 받을 수 있는 최대 메시지 크기를 반영해야 합니다. 이 값을 줄이면 바이러스 스캔 시간을 절약할 수 있지만, 바이러스가 감지되지 않은 채 통과할 가능성이 있습니다.	1048576
`MODE`	ClamAV 결과를 답신 정보로 변환하는 것을 제어합니다. 서로 다른 네 가지 모드를 사용할 수 있습니다. 0 - 바이러스가 있는 메시지를 발견한 경우 `VERDICT` 옵션에 지정된 답신 문자열이 반환되고, 그렇지 않으면 기본 답신이 반환됩니다. `VERDICT` 옵션이 비어 있거나 지정되어 있지 않으면 null 답신이 반환됩니다. 1 - 바이러스가 있는 메시지를 발견한 경우 ClamAV 결과이 답신으로 반환되고, 그렇지 않으면 기본 답신이 반환됩니다. 2 - 무조건적으로 ClamAV 결과 문자열을 답신으로 반환합니다. 기본 또는 null 답신이 반환되지 않으며 `VERDICT` 옵션이 사용되지 않습니다. 3 - 메시지에 바이러스가 있는 경우 ClamAV 결과 문자열이 반환되고, 그렇지 않으면 `VERDICT` 옵션에서 지정한 답신 문자열이 반환됩니다.	0
`PORT`	`clamd`가 실행 중인 포트를 지정합니다.	3310
`SOCKS_HOST`	중간 SOCKS 서버의 이름을 지정합니다. 이 옵션을 지정하면 `clamd`가 직접 연결되지 않고 지정된 SOCKS 서버를 통해 연결됩니다.	3310
`SOCKS_PORT`	중간 SOCKS 서버가 실행 중인 포트를 지정합니다.	1080
`SOCKS_PASSWORD`	SOCKS 서버를 통해 연결을 설정할 때 사용할 비밀번호를 지정합니다. 아이디/비밀번호가 필요한지 여부는 SOCKS 서버 구성에 따라 다릅니다.	""
`SOCKS_USERNAME`	SOCKS 서버를 통해 연결을 설정할 때 사용할 아이디를 지정합니다.	""
`VERDICT`	모드 0과 3에 사용되는 답신 문자열을 지정합니다.	""

14.7 시브(Sieve) 확장 지원

표준 시브(Sieve) 함수 외에도 Messaging Server는 addheader, addtag, spamtest 및 spamadjust를 비롯한 여러 확장에 대한 지원을 제공합니다. addheader 및 addtag는 452 페이지 “스팸 메시지에 SpamAssassin 점수가 포함된 헤더 추가” 및 453 페이지 “SpamAssassin 결과 문자열을 제목 줄에 추가”에 설명되어 있습니다.

관리자는 이러한 확장을 사용하여 서로 다른 임계값을 설정하고 SpamAssassin 답신을 대체하는 목록을 설정할 수 있습니다. 특정 메시지를 받는 사용자에 따라 두 가지를 결합하여 서로 다른 임계값을 지정할 수 있습니다. spamadjust는 비표준 작업입니다. spamtest는 ftp://ftp.isi.edu/in-notes/rfc3685.txt (ftp://ftp.isi.edu/in-notes/rfc3685.txt)에 설명되어 있습니다.

addtag에서 여러 제목 줄 태그 추가를 구분하는 데 사용되는 내부 구분자 문자가 공백에서 세로 막대로 바뀌었습니다. 따라서 일부 스팸 필터에서 필요한 대로 공백이 포함된 태그를 추가할 수 있습니다. 예를 들어, 이전에는 addtag "[Probable Spam]"이 addtag "[Probable" 및 add tag "spam]"을 의미했습니다. 이제는 "[Probable Spam]"이라는 단일 태그로 간주됩니다. 이 변경으로 세로 막대는 태그에 사용할 수 없습니다.

spamtest를 사용하면 시브(Sieve) [RELATIONAL] 확장을 "i;ascii-numeric" 구분 문자와 함께 사용하여 SpamAssassin 점수를 특정 값과 비교할 수 있습니다. SpamAssassin 점수는 일반적으로 실수이지만 spamtest는 점수를 근사 정수로 반올림하여 0과 10 사이의 정수 값으로 강제합니다. 0 미만인 값은 0으로 반올림되고 10을 초과하는 값은 10으로 내림합니다. Messaging Server에서 유지 관리되는 텍스트 문자열을 첨부하여 spamtest 테스트에 표시되는 테스트 문자열을 생성합니다. :percent는 spamtest에서 지원됩니다(SIEVE Email Filtering: Spamtest and Virustest Extensions draft-ietf-sieve-spamtestbis-05 (http://www.ietf.org/internet-drafts/draft-ietf-sieve-spamtestbis-05.txt) 참조).

spamadjust는 현재 스팸 점수를 조정하는 데 사용됩니다. 이 작업은 실수 값에 대해 스캔되는 단일 문자열 인수를 가져옵니다. 이 값은 현재 스팸 점수를 조정하는 데 사용됩니다. 전체 문자열이 현재 점수 텍스트 문자열에 첨부됩니다. 아래에 표시된 예에서 문자열은 "undisclosed recipients""

여러 spamadjust 작업이 허용되고 각 작업이 현재 점수에 추가됩니다. 점수 값은 항상 0에서 시작합니다. 서명된 숫자값을 허용하여 현재 점수를 내리거나 높일 수 있습니다. spamadjust의 require 절이 없으므로 spamtest 확장이 대신 나열되어야 합니다.

예를 들어, SpamAssassin MODE 설정 2와 함께 사용할 수 있는 spamadjust는 다음과 같습니다.

```
spamfilterX_string_action=data:,require ["spamtest"];spamadjust "$U";
```

시스템 수준 시브(Sieve) 필터는 특정 헤더 유형(있는 경우)을 검사한 다음 SpamAssassin 점수에 5를 더하여 SpamAssassin 점수를 수정할 수 있습니다.

```
require "spamtest";
if header :contains ["to", "cc", "bcc", "resent-to", "resent-cc",
   "resent-bcc"]  ["<undisclosed recipients>", "undisclosed.recipients"]
{spamadjust "+5 undisclosed recipients";}
```

마지막으로 사용자 수준 시브(Sieve) 스크립트는 결과 값을 테스트하고 스팸으로 확인된 메시지를 삭제하고, 스팸일 확률이 있는 메시지를 파일로 저장하고, 메시지를 로컬 도메인의 주소에서 통과하도록 허용할 수 있습니다.

```
require ["spamtest", "relational",
"comparator-i;ascii-numeric", "fileinto"];
if anyof (address :matches "from" ["*@siroe.com", "*@*.siroe.com"])
    {keep;}
elsif spamtest :value "ge" :comparator "i;ascii-numeric" "8"
    {discard;}
elsif spamtest :value "ge" :comparator "i;ascii-numeric" "5"
    {fileinfto "spam-likely";}
else
   {keep;}
```

14.8 Milter 사용

이 절은 다음과 같은 하위 절로 구성되어 있습니다. 473 페이지 “14.8 Milter 사용”

14.8.1 Milter 개요

Milter는 Sendmail Content Management API의 약식 이름으로, 이 API를 사용하여 작성한 소프트웨어를 나타내기도 합니다. Milter는 MTA를 통해 전달되는 메시지의 유효성을 검사, 수정 또는 차단하는 데에 사용되는 타사 소프트웨어의 플러그 인 인터페이스를 제공합니다. Milter는 메시지의 연결(IP) 정보, 봉투 프로토콜 요소, 메시지 헤더 및/또는 메시지 본문 내용을 처리하고 메시지 수신자, 헤더 및 본문을 수정할 수 있습니다. 필터링은 스팸 거부, 바이러스 필터링 및 내용 제어에 사용할 수 있습니다. 일반적으로 Milter는 사이트 전체의 필터링 문제를 확장 가능한 방식으로 해결할 방법을 찾습니다. 원래 sendmail용으로 설계되어 sendmail용으로 작성된 Milter는 Messaging Server에도 사용할 수 있지만 몇 가지 제한이 적용됩니다(아래 참조). Milter에 대한 자세한 내용은 인터넷을 참조하십시오.

14.8.2 Milter/Messaging Server 작동 원리

Milter는 메시지에서 수행되는 작업을 제어합니다. Messaging Server는 434 페이지 "14.2.2 필터링할 메시지 지정"에 설명된 방법을 사용하여 Milter로 작업을 수행할 메시지를 제어합니다.

sendmail에서 Milter는 sendmail 자체의 지원 코드와 별도의 `libmilter` 라이브러리로 구성됩니다. 필터 작성자는 필터를 `libmilter`에 링크하여 서버를 만듭니다. 그런 다음 sendmail이 Milter 서버에 연결되도록 구성합니다.

Messaging Server는 Milter 인터페이스의 sendmail측을 에뮬레이트하는 라이브러리를 제공합니다. 따라서 sendmail용으로 작성된 Milter를 Messaging Server에서 사용할 수 있습니다.

여기에서 몇 가지 주의할 사항이 있습니다. Milter 프로토콜에는 텍스트 및 이진 요소가 복잡하게 혼합되어 있으며 설명서가 잘 구비되어 있지 않습니다. 또한, Milter의 의미는 sendmail에서 메시지를 처리하는 방식과 밀접한 연관이 있습니다. 특히, Milter는 sendmail 구성에 정의된 매크로 중 일부에 액세스할 수 있으며 실제로 자주 액세스합니다. Messaging Server의 Milter 클라이언트 라이브러리에서는 충분한 sendmail 매크로 집합을 제공하려고 시도하지만, 현재 구현되지 않은 sendmail 구성의 특정 부분에 의존하도록 Milter를 작성할 수도 있습니다. 따라서 네트워크에서 임의로 추출한 Milter는 해당 클라이언트 라이브러리에서 작동할 수도 있고 작동하지 않을 수도 있습니다. 문제가 발생하는 경우 문제 해결을 위해 최선을 다하겠지만, 모든 Milter에 대해 성공을 보장할 수는 없습니다.

14.8.3 Milter 요구 사항 및 사용 시 고려 사항

Milter 서버는 별도의 독립적인 서버에서 실행할 수도 있고 Messaging Server와 같은 시스템에서 실행할 수도 있으며, MTA와 같은 시스템에 단일 시스템 배포 또는 2계층 배포로 실행할 수도 있습니다. MTA와 메시지 저장소 사이에서 LMTP를 사용하는 경우에는 MTA에서 필터링을 호출해야 합니다. 메시지 저장소에서는 필터링을 호출할 수 없습니다. MTA와 메시지 저장소 간에 SMTP가 사용되는 경우 각 시스템이나 별도의 타사 시스템에서 실행할 수 있습니다.

Messaging Server는 여러 Milter 서버에 대한 연결을 지원합니다. 여러 IP 주소로 변환되는 도메인 이름을 지정하면 연결이 작동될 때까지 DNS에서 받은 순서대로 모든 이름에 대한 연결을 시도합니다. 일부 DNS 서버는 반환되는 주소의 순서를 무작위로 지정하는 기능을 지원하며, 이를 통해 원시 로드 균형 조정/페일오버 기능을 제공합니다.

14.8.3.1 지원되는 Milter 메시지 수정 작업

Milter 인터페이스는 현재 헤더 추가(`SMFIF_ADDHDRS`), 헤더 변경 또는 삭제(`SMFIF_CHGHDRS`) 및 메시지 검역소(`SMFIF_QUARANTINE`) 기능을 지원합니다. 메시지 본문 변경(`SMFIF_CHGBODY`), 수신자 추가(`SMFIF_ADDRCPT`) 및 수신자 삭제(`SMFIF_DELRCPT`)는 현재 지원되지 않습니다.

14.8.3.2 Milter 인터페이스에서 제공되는 매크로

Milter 인터페이스에는 현재 다음과 같은 매크로가 정의되어 있습니다.

`$j` Received: 헤더 필드의 by 절에 텍스트를 넣습니다. Messaging Server에서는 `RECEIVED_DOMAIN` MTA 옵션을 통해 제어됩니다. 해당 옵션을 설정하지 않은 경우에는 대신 `local` 채널에 있는 공식 호스트가 사용됩니다.

`${client_addr}` 점으로 구분된 네 개의 번호 값으로 표현되는 SMTP 클라이언트의 IP 주소입니다. SMTP over TCP를 사용하는 경우에만 설정됩니다.

`$i` 현재 메시지의 대기열 아이디입니다. Messaging Server는 각 세션에 대해 고유한 아이디를 생성합니다. 이 아이디가 `$i` 매크로에 표시됩니다.

`${mail_addr}` 현재 트랜잭션의 `MAIL FROM` 주소입니다.

`${mail_host}` 현재 트랜잭션의 `MAIL FROM` 주소에서 호스트 부분입니다.

`${rcpt_addr}` 현재 트랜잭션의 `RCPT TO` 주소입니다.

`${rcpt_host}` 현재 `RCPT TO` 주소의 호스트 부분입니다.

▼ Milter 배포

Milter를 배포하려면 다음 단계를 수행합니다.

1. **원하는 작업을 수행할 Milter를 가져와서 구성합니다.**
   정보 얻기 및 구성에 대한 자세한 내용은 해당 Milter 설명서를 참조하십시오.

2. **Milter 클라이언트 라이브러리를 로드하고 구성합니다. 433 페이지 "14.2.1 스팸 필터링 소프트웨어 클라이언트 라이브러리 로드 및 구성"을 참조하십시오.**

   a. **클라이언트 라이브러리 `libmilter.so`의 경로를 지정합니다. Milter 구성 파일의 경로 및 이름을 지정합니다.**
      예:

      ```
      spamfilter1_library=/opt/SUNWmsgsr/lib/libmilter.so
      spamfilterX_config_file=/opt/SUNWmsgsr/lib/milter.opt
      ```

   b. **원하는 옵션을 사용하여 Milter 구성 파일을 만듭니다.**
      Milter 옵션 파일은 `option=value` 형식의 행으로 구성됩니다. 두 개의 필수 옵션은 `HOST`와 `PORT`입니다. `HOST`는 Milter 서버를 실행하는 시스템의 이름으로 설정해야 하며, `PORT`는 Milter 서버가 수신하도록 구성된 포트로 설정해야 합니다. TCP/IP 연결만 지원되며, UNIX 도메인 소켓은 지정하거나 사용할 수 없습니다.

      이 옵션 파일에서 몇 개의 옵션을 추가로 사용할 수도 있습니다.

DEBUG(정수, 기본값 0) — Milter 클라이언트 라이브러리의 디버그 출력을 활성화하거나 비활성화합니다. 값이 클수록 디버깅 출력이 많아집니다. 0을 설정하면 출력이 없습니다. 1은 기본 디버깅을 제공합니다. 2는 TCP 트래픽 로깅을 추가합니다. (Milter 서버의 디버그 출력은 일반적으로 서버 시작에 사용되는 명령줄의 설정에 따라 제어됩니다. 대부분의 Milter는 syslog에 대한 직접 디버그 출력 기능만 제공합니다.)

TIMEOUT(정수, 기본값 3600) — Milter 연결과 관련된 작업의 시간 초과 값을 100분의 1초 단위로 지정합니다. 6.3 이상 버전에서 사용할 수 있습니다.

SOCKS_HOST(문자열, 기본값 "") — 중간 SOCKS 서버의 이름을 지정합니다. 이 옵션을 지정하면 Milter가 직접 연결되지 않고 지정된 SOCKS 서버를 통해 연결됩니다.

SOCKS_PORT(정수, 기본값 1080) — 중간 SOCKS 서버가 실행되는 포트를 지정합니다.

SOCKS_PASSWORD(문자열, 기본값 "") — SOCKS 서버를 통한 연결에 사용할 비밀번호를 지정합니다. 아이디/비밀번호가 필요한지 여부는 SOCKS 서버 구성에 따라 다릅니다.

SOCKS_USERNAME(문자열, 기본값 "") — SOCKS 서버를 통한 연결에 사용할 아이디를 지정합니다.

3 **Milter로 보낼 메시지를 지정합니다.**

사용자, 도메인 또는 채널별로 메시지를 필터링할 수 있습니다. 434 페이지 “14.2.2 필터링할 메시지 지정”을 참조하십시오.

4 **option.dat 파일에서 spamfilter*X*_string_action 옵션을 설정합니다.**

```
spamfilterX_string_action=data:,$M
```

이 설정은 무조건적으로 사용되지만 Milter가 제대로 작동하려면 MTA 옵션 파일에 있어야 합니다.

14.9 기타 스팸 방지 및 서비스 거부 기술

스팸 및 바이러스 필터링 소프트웨어를 시스템에 추가하는 것이 사용자 메일함에 들어오는 스팸 및 바이러스를 줄일 수 있는 가장 효과적인 방법입니다. 그러나 Messaging Server는 스팸 필터링을 지원하는 그 밖의 다양한 기술과 방법을 제공합니다. 이러한 기술은 스팸 필터링 이외의 용도로도 자주 사용되므로 이 책 전반에 걸쳐 다뤄집니다. 다음은 스팸 방지 및 서비스 거부 기술을 설명하는 절입니다.

스팸 방지 기술:

- 477 페이지 “14.9.1 스팸 방지 기술: SMTP 배너 보내기 지연”
- 391 페이지 “12.12.6 주소 검증 후와 확장 전의 라우팅”
- 15 장
- 18 장
- 703 페이지 “23.7 POP, IMAP 및 HTTP 서비스에 대한 클라이언트 액세스 구성”

- 339 페이지 “12.4.2.6 DNS 도메인 확인”
- 359 페이지 “12.5.9 여러 주소 확장”
- 541 페이지 “18.14 MTA 차원 필터 만들기”
- 529 페이지 “18.7 SMTP 릴레이 차단 구성”

서비스 거부 기술:

- 19 장
- 382 페이지 “12.9.2 절대 메시지 크기 제한 지정”
- 524 페이지 “18.3.6 MTA에 대해 지정된 IP 액세스 연결 제한”
- 836 페이지 “27.4.1 메시지 대기열 크기 모니터링”
- 837 페이지 “27.4.3 인바운드 SMTP 연결 모니터링”

14.9.1 스팸 방지 기술: SMTP 배너 보내기 지연

유용한 스팸 차단 전략 중 하나는 0.5초 정도의 짧은 시간에 SMTP 배너 보내기를 지연했다가 입력 버퍼를 지우고 최종적으로 배너를 보내는 것입니다. 이 방법이 효과적인 이유는 많은 스팸 클라이언트가 표준과 호환되지 않고, 서버가 보내는 응답을 무시하면서 연결이 열리는 즉시 SMTP 명령을 내보내기 시작하기 때문입니다. 이 기능이 사용 가능한 경우 그렇게 작동하는 스팸 클라이언트에서는 SMTP 대화 중에 처음 몇 개의 명령이 손실되며 대화의 나머지는 잘못된 것으로 렌더링됩니다.

이제 Messaging Server에서 이 기능이 구현되었습니다. `BANNER_PURGE_DELAY` SMTP 채널 옵션을 배너 제거 및 전송 전 지연 시간(1/100초)으로 설정하여 이 기능을 무조건적으로 사용 가능하게 할 수 있습니다. 값이 0이면 지연과 제거가 모두 사용 불가능하게 됩니다.

또한 `PORT_ACCESS` 매핑을 사용하여 이 기능을 제어할 수 있습니다. 템플리트에서 `$D`를 지정하면 필수 SMTP `auth rulset` 및 영역 그리고 선택적 응용 프로그램 정보 추가 이후에 템플리트 결과로부터 추가 인수를 읽게 됩니다. 이 값은 `BANNER_PURGE_DELAY` 값과 동일한 의미의 정수이어야 합니다. 모든 `PORT_ACCESS` 매핑 설정은 `BANNER_PURGE_DELAY` SMTP 채널 옵션을 대체합니다.

◆ ◆ ◆ 15 장

15 SPF(Sender Policy Framework)를 사용하여 위조된 전자 메일 처리

스팸 제작자와 전자 메일 사기범은 가짜 도메인 이름과 전자 메일 주소 또는 합법적인 도메인 이름과 전자 메일 주소를 사용하여 전자 메일을 위조함으로써 사용자가 알고 있는 회사에서 온 메일인 것처럼 위장합니다. 예를 들어, president@whitehouse.gov와 같은 전자 메일 주소를 사용해서 전자 메일을 보내면 사용자는 실제로 메일이 그 주소에서 왔다고 믿을 수 있습니다. 전자 메일을 위조하는 기법은 사용자가 신청하지 않은 메일을 열게 만들거나 엉뚱한 곳에 정보를 제공하게 만드는 일에 사용됩니다. 또한, 스팸 발송자들은 전자 메일을 RBL 목록에 없는 합법적인 도메인에서 보내고 싶어 합니다.

SPF(Sender Policy Framework)는 SMTP 대화 중에 위조된 전자 메일을 감지하여 거부할 수 있는 기술입니다. SPF는 도메인 이름을 사용할 수 있는 호스트를 도메인에서 명시적으로 인증하도록 허용하는 프로토콜입니다. 이 인증 정보를 확인하도록 받는 호스트를 구성할 수도 있습니다. 이러한 방식으로 SPF는 위조된 전자 메일의 수를 줄일 수 있습니다.

15.1 작동 원리

Messaging Server에 메시지가 오면 MTA는 SPF 쿼리를 통해 주소가 실제로 주소에 있는 도메인으로부터 왔는지 확인합니다. SPF 쿼리에서는 DNS를 참조하여 메시지의 도메인(*domain*)에 속한 TXT 레코드를 찾습니다. *Domain*은 HELO 또는 EHLO(`spfhelo` 채널 키워드를 사용하는 경우) 인수로 지정한 도메인 이름이거나 MAIL FROM: 명령으로 지정한 발송자 주소의 도메인 이름입니다(보통 @ 문자 뒤에 오는 부분). 도메인 이름이

지정되지 않았거나 도메인 이름을 사용할 수 없는 경우에는 HELO/EHLO를 사용하는 동안 지정된 값이 *domain*으로 사용됩니다. 대부분의 ISP는 도메인과 일치하는 인증된 IP 목록을 배포합니다. IP 주소가 도메인 이름과 일치하지 않는 경우에는 메시지가 위조된 것으로 간주됩니다.

주 – 보통은 DNS를 쿼리하기 전에 SPF_LOCAL 매핑 테이블에서 일치하는 도메인을 확인합니다. 일치하는 항목이 발견되면 해당 항목을 먼저 사용합니다.

매핑 테이블에서 찾은 레코드에 `redirect=`*domain* 절이 포함되어 있으면 DNS 쿼리를 통해 도메인으로 리디렉션해서 반복적인 매핑 파일 확인을 건너뜁니다.

결과 TXT 레코드의 예는 다음과 같습니다.

```
v=spf1 +mx a:colo.siroe.com/28 -all
```

이 RFC에서 지원되는 SPF 레코드에는 `v=spf1` 토큰이 필요합니다.

`+mx`는 MX 레코드에서 *domain*을 검사하고 이 SMTP 연결의 소스 IP 주소가 *domain*의 MX 쿼리 결과로 얻은 IP 주소 중 하나와 일치하는지 확인하도록 지시합니다. 일치하는 항목이 있는 경우 표시되는 `+`는 결과가 `Pass`임을 의미합니다.

`a:colo.siroe.com/28`은 `colo.siroe.com`의 A 레코드를 검사한 후 28비트만을 비교하여(255.255.255.240에 대해 마스크 처리됨) 이 SMTP 연결의 소스 IP 주소가 A 레코드에 지정된 것과 동일한 CIDR 서브넷에 있는지 확인합니다. 한정자가 지정되지 않았기 때문에 기본값은 결과가 `Pass`임을 의미하는 `+`입니다.

마지막으로 `-all`은 다른 모든 부분을 일치시키고 결과로 `Fail`을 결정합니다. SPF 레코드에 대한 자세한 내용은 `http://www.ietf.org/rfc/rfc4408.txt`에 있는 RFC 4408을 참조하십시오.

SPF 처리 결과는 여러 항목 중 하나일 수 있습니다. 아래 표에는 결과와 그 설명이 표시됩니다.

표 15–1 SPF 처리 결과

결과	설명
`Pass`	조회가 성공하여 SPF 레코드를 찾았으며 레코드에서 발송 시스템의 *domain* 사용이 인증되었음을 확인했습니다.
`Fail`	조회 결과 SPF 레코드를 찾았지만 SMTP 트랜잭션 중에 SMTP 클라이언트의 *domain* 사용 권한이 레코드에서 명시적으로 거부되었습니다. SPF 구현의 기본 동작은 5xx 회신을 표시하며 SMTP 명령을 거부하는 것입니다.

표 15–1 SPF 처리 결과 (계속)

결과	설명
SoftFail	조회 결과 일치하는 SPF 레코드를 찾았으며 레코드에서 SMTP 클라이언트의 *domain* 사용 인증이 거부되었지만 거부가 덜 명확해서 바로 실패로 확인되지는 않았습니다. 구현의 기본 동작은 메시지를 받지만 SoftFail을 Received-SPF: 헤더에 표시하여 시브(Sieve) 처리 등의 이후 평가에 반영하는 것입니다.
Neutral	SPF 레코드에서 SMTP 클라이언트의 *domain* 사용 인증을 요구하지 않습니다. 메시지는 받습니다. 사양에 따라 Neutral은 아래의 None과 같이 처리해야 합니다.
None	일치하는 SPF 레코드를 찾지 못했기 때문에 SPF 처리가 수행되지 않았습니다.
PermError	SPF 처리 중에 SPF 레코드의 구문 오류, 중첩된 SPF 레코드 처리 중의 DNS 실패(include: 기법이나 redirect= 수정자로 인한) 또는 중첩된 SPF 레코드 처리 중에 구성된 SPF 처리 제한 초과 등의 영구적인 오류가 발생했습니다. 기본 동작은 5xx 회신을 표시하며 SMTP 명령을 거부하는 것입니다.
TempError	SPF 처리 중에 임시 오류가 발생했으며, SPF 레코드를 쿼리하는 동안 발생한 DNS 시간 초과 때문일 가능성이 큽니다. 기본 동작은 4xx 회신을 표시하며 SMTP 명령을 거부하는 것입니다.

SPF 처리가 완료되고 나면 메시지에 Received-SPF: 헤더가 기록되어 SPF 처리의 결과를 나타냅니다. 이 헤더는 시브(Sieve) 처리 중에 쿼리하여 이후의 고려에 반영할 수 있습니다. option.dat 파일에서 MTA 옵션 MM_DEBUG를 활성화한 경우(>0) 본격적인 디버깅을 사용할 수 있습니다.

15.2 제한

SPF는 스팸 방지를 위해 사용할 수 있는 도구 중 하나에 불과하며, 모든 문제를 처리할 수는 없습니다. 스팸 발송자가 도메인을 만들고 도메인이 합법적인 것처럼 보이게 만드는 SPF TXT 레코드를 추가하는 일은 쉽습니다. 한편, SPF 실패를 피할 수 있는 TXT 레코드는 많지만, 잘 알려진 ISP에서 위조된 전자 메일이 오는 경우 SPF에서 효과적으로 감지할 수 있습니다.

15.3 배포 전 고려 사항

모든 메시지에 대해 DNS 쿼리를 수행해야 하기 때문에 매우 빠른 DNS 서버를 갖추는 것이 중요합니다.

15.4 기술 설정

SPF 기술 설정은 두 단계를 통해 이루어집니다.

- 받는 TCP 채널에 채널 키워드를 포함합니다. 보통은 tcp_local 채널이 사용되지만 `tcp_local`에서 다른 채널로 채널 전환을 허용하는 경우에는 다른 채널을 사용할 수도 있습니다. 표 15–2를 참조하십시오.
- `option.dat` 파일에서 옵션을 설정합니다. 표 15–3을 참조하십시오.

15.5 참조 정보

이 절에서는 SPF 채널 키워드와 SPF MTA 옵션에 대한 참조 정보를 제공합니다. SPF 지원은 받는 tcp_* 채널(보통은 tcp_local)에 적용되는 네 개의 채널 키워드를 통해 구현됩니다. 다음 표에는 키워드와 해당 설명이 나와 있습니다.

표 15–2 SPF 키워드

키워드	설명
`spfnone`	SPF 처리를 비활성화합니다.
`spfhelo`	HELO 또는 EHLO의 인수로 지정된 도메인의 SPF 처리를 활성화합니다.
`spfmailfrom`	MAIL FROM:을 받은 후에 발송자 봉투 주소에 제공되는 도메인 이름의 SPF 처리를 활성화합니다.
`spfrcptto`	RCPT TO:를 받은 후에 발송자 봉투 주소에 제공되는 도메인 이름의 SPF 처리를 활성화합니다. RCPT TO: 명령이 실행되고 수신자가 다른 방식으로 유효한 수신자인 것이 확인될 때까지 SMTP 트랜잭션이 지연된다는 점을 제외하면 `spfmailfrom`와 처리가 같습니다.

주 – `spfmailfrom`과 `spfrcptto`는 혼동되기 쉬운 키워드이기 때문에 채널에는 두 키워드 중 하나만 지정해야 합니다. 하지만 `spfhelo`를 `spfmailfrom` 또는 `spfrcptto`와 함께 사용하면 두 종류의 SPF 검사를 모두 수행할 수 있습니다.

SPF 처리를 제한하고 다양한 SPF 결과(`Fail`, `SoftFail`, `PermError` 및 `TempError` 포함)에 따라 SMTP 명령을 받을 것인지, 4xx 회신과 함께 실패할 것인지(임시 실패) 또는 5xx 회신과 함께 실패할 것인지(영구 실패) 제어하는 추가 지원이 있습니다.

option.dat에서 다음 MTA 옵션을 사용하여 SPF 처리에 제한을 적용할 수 있습니다.

표 15–3 SPF 제한 옵션

옵션	설명
SPF_MAX_RECURSION	include: 또는 redirect=로 중첩된 SPF 레코드에 허용되는 반복 횟수를 지정합니다. 이 제한을 초과하면 PermError가 발생합니다. 기본값: 10(RFC에서 필수)
SPF_MAX_DNS_QUERIES	DNS 조회가 필요한 기법 또는 수정자의 수를 지정합니다(include:, a:, mx:, ptr:, exists:, redirect= 및 exp= 포함). 제한은 실제 조회 수로 계산되는 것이 아니며 한 기법에서 여러 DNS 쿼리가 발생할 수 있습니다. 이 제한을 초과하면 PermError가 발생합니다. 기본값: 10(RFC에서 필수)
SPF_MAX_TIME	SPF 처리를 완료할 때까지 허용되는 시간(초)을 지정합니다. 이 값을 초과하면 TempError가 발생합니다. 기본값은 RFC 권장 사항보다 덜 엄격하게 지정됩니다. 기본값: 45

또한, option.dat에서 다음 MTA 옵션을 구성하여 SPF 결과(Fail, SoftFail, PermError 및 TempError)에 대한 회신으로 SMTP 서버의 동작을 제어할 수 있습니다. 각 결과에 대해 SMTP 서버는 2xx(성공) 응답, 4xx(임시 실패) 또는 5xx(영구 실패)를 반환할 수 있습니다. 또한 Fail 및 SoftFail의 경우 MTA는 SPF 결과를 "모든" 기법의 결과와 기타 방법으로 명시적으로 참조된 일치로 구분할 수 있습니다. 그런 다음 특정 결과와 SPF 레코드의 기본 결과로 구분할 수 있습니다. 이 모든 옵션에서 유효한 값은 2, 4 또는 5입니다. 2, 4 또는 5 값은 특정 SPF 상태를 가져온 결과로 SMTP에서 나타나는 2xx, 4xx 또는 5xx 응답에 해당됩니다. 따라서, 예를 들어 SPF_SMTP_STATUS_FAIL=2와 SPF 레코드가 "-a:192.168.1.44"(이쪽 IP 주소)를 사용하여 명시적으로 차단하는 경우 5xx 응답을 보내기 전에 대신 "250 OK"로 주소를 받습니다.

표 15–4 SPF 실패 및 오류 옵션

옵션	설명
SPF_SMTP_STATUS_FAIL	SPF 레코드의 일치 값이 "-all" 외의 기법이 플래그로 표시된 "-"인 경우에 사용됩니다. 기본값: 5
SPF_SMTP_STATUS_FAIL_ALL	일치하는 기법이 "-all"인 경우에 사용됩니다. 기본값: 5

표 15–4 SPF 실패 및 오류 옵션 (계속)

옵션	설명
SPF_SMTP_STATUS_SOFTFAIL	SPF 레코드의 일치 값이 "~all" 외의 기법이 플래그로 표시된 "~"인 경우에 사용됩니다. 기본값: 2
SPF_SMTP_STATUS_SOFTFAIL_ALL	일치하는 기법이 "~all"인 경우에 사용됩니다. 기본값: 2
SPF_SMTP_STATUS_TEMPERROR	임시 실패가 있는 경우에 사용되며, 보통 DNS 처리 문제와 관련됩니다. 기본값: 4
SPF_SMTP_STATUS_PERMERROR	영구 실패가 있는 경우에 사용되며, 보통 영구 실패는 SPF 처리 중에 구문 오류나 기타 기술 오류가 발견된 경우에 발생합니다. (로컬 오류가 아닌 오류로 인해 발생합니다.) 기본값: 5

15.6 spfquery를 사용하여 SPF 테스트

이 테스트 유틸리티를 사용하여 SPF 처리를 테스트할 수 있습니다.

주 – spfquery는 SPF 구성을 테스트하지 않습니다. 여기서는 SPF 처리를 활성화하는 경우에 반환되는 내용을 테스트합니다.

요구 사항: Messaging Server 이진 실행 액세스 권한과 루트 또는 mailsrv 등의 라이브러리에 대한 액세스 권한이 있는 사용자로 실행해야 합니다.

위치: *msg-svr-base*/sbin/

15.6.1 구문

```
spfquery [-i ip-address] [-s sender-email] [-h helo-domain]
  [-e none | neutral | pass | fail | temperror | permerror] [-v] [-V] [?] domain
```

다음 표에는 spfquery 옵션과 해당 설명이 표시되어 있습니다.

표 15–5 spfquery 옵션

옵션	설명
`-i` *ip address*	SPF 쿼리의 원격 주소로 사용할 IP 주소를 지정합니다. 기본값은 `127.0.0.1`입니다. 이 옵션은 `--ip-address`가 될 수도 있습니다.
`-s` *domain*	`MAIL FROM:`에 지정된 것처럼 사용되는 전자 메일 주소입니다. 기본값: postmaster@*domain*. 이 옵션은 `--sender`로도 지정할 수 있습니다.
`-h` *helo-domain*	HELO 도메인에 지정된 것처럼 사용되는 도메인 이름입니다. 이 도메인 자체는 확인되지 않았으며 매크로 처리의 보충 정보로 제공됩니다. 기본값은 *domain*에 지정한 값과 같습니다. 이 옵션은 `--helo-domain`이 될 수도 있습니다.
`-e` *result*	`spfquery`는 SPF 처리의 결과를 예상한 것과 비교하여 결과가 다른 경우 메시지를 인쇄하고 `0`이 아닌 반환 값과 함께 `spfquery`를 종료합니다. 결과는 `none`, `neutral`, `pass`, `fail`, `softfail`, `temperror` 또는 `permerror`가 될 수 있습니다. 이 옵션은 `--expect`로도 지정할 수 있습니다.
`-v`	SPF 처리 중에 세부 정보 표시 출력을 활성화합니다. 이 옵션은 `--verbose`로도 지정할 수 있습니다.
`-V`	SPF 라이브러리의 현재 버전을 인쇄합니다. 이 옵션은 `--version`으로도 지정할 수 있습니다.
`-?`	이 사용 정보를 인쇄합니다. 이 옵션은 `--help`로도 지정할 수 있습니다.

15.6.2 디버깅을 사용하는 경우의 예

```
# /opt/SUNWmsgsr/sbin/spfquery -v -i 192.168.1.3 11.spf1-test.siroe.com
    Running SPF query with:
      IP address: 192.168.1.3
          Domain: 11.spf1-test.siroe.com
          Sender: postmaster@11.spf1-test.siroe.com (local-part: postmaster)
     HELO Domain: 11.spf1-test.siroe.com

    15:30:04.33: ----------------------------------------------------------------
    15:30:04.33: SPFcheck_host called:
    15:30:04.33:       source ip = 192.168.1.3
    15:30:04.33:          domain = 11.spf1-test.siroe.com
    15:30:04.33:          sender = postmaster@11.spf1-test.siroe.com
    15:30:04.33:      local_part = postmaster
    15:30:04.33:     helo_domain = 11.spf1-test.siroe.com
    15:30:04.33:
    15:30:04.33:   Looking up "v=spf1" records for 11.spf1-test.siroe.com
    15:30:04.35:     DNS query status: Pass
    15:30:04.35:       "v=spf1 mx:spf1-test.siroe.com                  -all"
    15:30:04.35:
    15:30:04.35:   Parsing mechanism: " mx : spf1-test.siroe.com"
    15:30:04.35:     Assuming a Pass prefix
```

```
15:30:04.35:     Processing macros in spf1-test.siroe.com
15:30:04.35:     Comparing against 192.168.1.3
15:30:04.35:     Looking for MX records for spf1-test.siroe.com
15:30:04.41:       mx02.spf1-test.siroe.com:
15:30:04.41:         192.0.2.22 - No match
15:30:04.41:         192.0.2.21 - No match
15:30:04.41:         192.0.2.20 - No match
15:30:04.41:         192.0.2.23 - No match
15:30:04.41:       mx01.spf1-test.siroe.com:
15:30:04.42:         192.0.2.13 - No match
15:30:04.42:         192.0.2.11 - No match
15:30:04.42:         192.0.2.12 - No match
15:30:04.42:         192.0.2.10 - No match
15:30:04.42:       mx03.spf1-test.siroe.com:
15:30:04.42:         192.0.2.32 - No match
15:30:04.42:         192.0.2.30 - No match
15:30:04.42:         192.0.2.31 - No match
15:30:04.42:         192.168.1.3 - Matched
15:30:04.42:   Mechanism matched; returning Pass
15:30:04.42:
15:30:04.42:   Parsing mechanism: "- all : " (not evaluated)
15:30:04.42:
15:30:04.42: SPFcheck_host is returning Pass
15:30:04.42: ----------------------------------------------------------------
```

15.7 SRS(Sender Rewriting Scheme)를 사용하여 SPF에서 전달된 메일 처리

위에서 설명한 대로, SPF는 메일 FROM: (봉투의 From) 주소에서 도메인과 연결된 특수 TXT 레코드를 조회하여 전자 메일 위조를 방지하는 기법입니다. 실제로는 여러 DNS 조회가 수행될 수 있는 이 작업은 결과적으로 도메인으로부터 메일을 보낼 수 있는 권한이 부여된 IP 주소의 목록을 만듭니다. 이 목록에 대해 SMTP 클라이언트의 IP 주소를 검사하며, 해당 IP 주소가 발견되지 않으면 메시지가 사기로 간주될 수 있습니다. SPF 지원은 Messaging Server 버전 6.3에서 구현되었습니다.

SPF는 메일 전달 서비스를 제공하는 사이트(예: 졸업생에게 서비스를 제공하는 대학교 또는 회원들에게 서비스를 제공하는 전문 기관)에서 심각한 문제가 됩니다. 전달자는 임의의 발송자로부터 온 메일을 보내게 됩니다. 이 발송자에는 SPF 정책을 구현한 발송자와 전달 시스템의 IP 주소를 도메인의 주소를 사용하도록 허용된 주소로 나열하지 않은 발송자도 포함될 수 있습니다.

SRS(Sender Rewriting Scheme)는 이 문제에 대한 해결책을 제시합니다. SRS는 전달자의 도메인을 사용하여 원래 보낸 사람의 주소를 새 주소로 캡슐화하여 작동합니다. 전달자 자체의 도메인만 SPF 검사를 위해 공개됩니다. 이 주소를 사용할 경우 메일(일반적으로 알림)을 전달자에게 라우팅하며, 전달자는 주소 캡슐화를 제거한 다음 실제 대상에게 메시지를 보냅니다.

물론 주소 캡슐화가 완전히 새로운 개념은 아닙니다. 소스 경로가 RFC 822에 정의되었으며, 백분율 핵 라우팅 및 뱅 경로와 마찬가지로 이러한 기능을 제공합니다. 그러나 이러한 기법 사용을 허용하면 시스템이 열린 중계가 되므로 지금의 인터넷 환경에서는 문제가 될 수 있습니다.

SRS는 캡슐화 형식에 키가 지정된 해시와 타임스탬프를 추가하여 이 문제를 처리합니다. 이 주소는 일정한 기간 동안만 유효하며, 그 기간 이후에는 사용할 수 없습니다. 해시는 타임스탬프나 캡슐화된 주소가 수정되는 것을 방지합니다.

또한 SRS는 주소 길이를 지나치게 늘리지 않으면서 멀티홉 전달을 처리하는 기법을 제공합니다. 그러기 위해서는 SRS 주소 형식 지정 중 일부가 SRS를 구현하는 모든 시스템에서 동일한 방식으로 수행되어야 합니다.

SRS 지원은 6.3P1 릴리스에 대해 구현되었습니다. 다음 MTA 옵션이 추가되었습니다.

- `SRS_DOMAIN`이 옵션은 SRS 주소에서 사용할 도메인으로 설정되어야 합니다. 이 도메인으로 보낸 전자 메일은 해당 도메인에 대해 SRS 작업을 수행할 수 있는 시스템으로 라우팅되어야 합니다. SRS 처리는 일반 주소 처리 위에 중첩으로 처리되므로, 사이트가 주 도메인을 SRS 도메인으로 사용하는 것을 방지할 수 없습니다.
- `SRS_SECRETS`이는 SRS 주소 인코딩 및 디코딩에 사용되는 비밀 키를 쉼표로 구분한 목록입니다. 목록 중 첫 번째 키는 무조건적으로 인코딩에 사용됩니다. 디코딩의 경우, 다른 해시 값을 생성하기 위해 각각의 값을 시도합니다. 해시 중 일치하는 것이 있으면 디코딩 작업이 진행됩니다.

  여러 키를 사용할 수 있으므로 서비스 중단 없이 비밀을 변경하는 것이 가능합니다. 두 번째 키를 추가하고, 이전에 발급한 주소가 모두 시간 초과될 때까지 기다린 다음 첫 번째 키를 제거합니다.
- `SRS_MAXAGE`선택적으로 메시지가 시간 초과되기 전 경과될 시간(일)을 지정합니다. 이 옵션이 지정되지 않은 경우 기본값은 14일입니다.

  선택된 SRS 도메인을 위해 전자 메일을 처리하는 모든 시스템은 SRS 처리에 대해 구성되고 세 가지 SRS 옵션이 모두 동일하게 설정되어야 합니다.

이 옵션만 설정하면 SRS 주소 디코딩이 가능해집니다. 인코딩은 또 다른 문제로서, 전달 작업과 연관된 봉투 `From:` 주소에 인코딩되어야 합니다. SRS 인코딩은 여섯 개의 새로운 채널 키워드, 즉 `addresssrs`, `noaddresssrs`, `destinationsrs`, `nodestinationsrs`, `sourcesrs` 및 `nosourcesrs`에 의해 제어됩니다.

SRS 인코딩이 수행되려면 세 가지 조건이 충족되어야 합니다.

(1) 현재 소스 채널이 `sourcesrs`로 표시되어야 합니다. (`nosourcesrs`가 기본값임).

(2) 현재 대상 채널이 `destinationsrs`로 표시되어야 합니다(`nodestinationsrs`가 기본값임).

(3) 현재 주소는 다시 쓰여질 경우 `addresssrs`로 표시된 채널과 일치해야 합니다(`noaddress`가 기본값임).

이 조건이 모두 충족될 경우에만 인코딩이 이루어집니다. 가장 단순한 설정은 모든 메시지가 tcp_local 채널에서 들어오고 나가고 로컬이 아닌 주소는 모두 SRS 처리가 필요한 순수 전달 설정입니다. 그러한 설정에서는 tcp_local이 세 키워드, 즉 sourcesrs, destinationsrs 및 addresssrs로 표시됩니다.

마지막으로, imsimta test -rewrite는 어떤 주소가 입력되든지 SRS 인코딩 및 디코딩 결과를 표시하도록 향상되었습니다. 예를 들어, foo@example.com이라는 주소는 다음과 유사한 출력을 생성할 수 있습니다.

```
SRS encoding = SRS0=dnG=IS=example.com=foo@example.org
```

이 인코딩된 주소를 다시 쓰면 다음과 같은 출력을 생성합니다.

```
SRS decoding = foo@example.com
```

또한 imsimta test -rewrite는 SRS 디코딩 중에 발생하는 모든 오류를 표시합니다.

◆ ◆ ◆ 16 장

16 LMTP 전달

Sun Java System Messaging Server MTA는 다중 계층 Messaging Server 배포가 사용되는 상황에서 LMTP(RFC 2033에 정의된 Local Mail Transfer Protocol)를 사용할 수 있습니다. 인바운드 중계와 백엔드 메시지 저장소를 사용하는 이 시나리오에서는 중계가 주소 확장 및 전달 방법(예: 자동 회신, 전달)뿐만 아니라 메일링 목록 확장을 담당합니다. 기본적으로 백엔드 저장소에 대한 전달은 백엔드 시스템이 LDAP 디렉토리에서 수신자 주소를 다시 조회해야 하는 SMTP를 통해 이루어지므로 전체 MTA 방법이 사용됩니다. 빠르고 효율적인 전달을 위해 MTA는 SMTP 대신 LMTP를 사용하여 메시지를 백엔드 저장소에 전달할 수 있습니다. Sun Java System Messaging Server의 LMTP 서버는 일반적인 용도의 LMTP 서버가 아니라 릴레이와 백엔드 메시지 저장소 사이의 개인 프로토콜 역할을 합니다. 설명의 단순화를 위해 2계층 배포를 포함하는 예를 사용합니다.

주 – LMTP는 다중 계층 배포에서 사용하도록 설계되었기 때문에 단일 시스템 배포에서는 사용할 수 없습니다. 또한 Messaging Server의 LMTP 서비스는 다른 LMTP 서버나 다른 LMTP 클라이언트와 함께 작동하도록 설계되지 않았습니다.

이 장은 다음 내용으로 구성되어 있습니다.

16.1 LMTP 전달 기능

MTA의 LMTP 서버가 백엔드 메시지 저장소에 메시지를 전달하는 데 보다 효율적인 이유는 다음과 같습니다.

- 백엔드 저장소의 로드가 감소합니다.

  중계는 수평적으로 확장 가능하지만 백엔드 저장소는 그럴 수 없기 때문에 가능한 중계에서 프로세스를 처리하는 것이 좋습니다.

- LDAP 서버의 로드가 감소합니다.

  LDAP 인프라가 대용량 메시징 배포의 제한 요소가 되는 경우가 있습니다.

- 메시지 대기열의 수가 감소합니다.

  대기열을 중계와 백엔드 저장소 모두에 두면 메시징 배포 관리자가 손실된 메시지를 찾기가 훨씬 어려워집니다.

16.2 LMTP를 사용하지 않는 2계층 배포의 메시징 처리

그림 16–1은 LMTP를 사용하지 않는 2계층 배포 시나리오의 다음 메시지 처리 설명을 그림 형식으로 설명합니다.

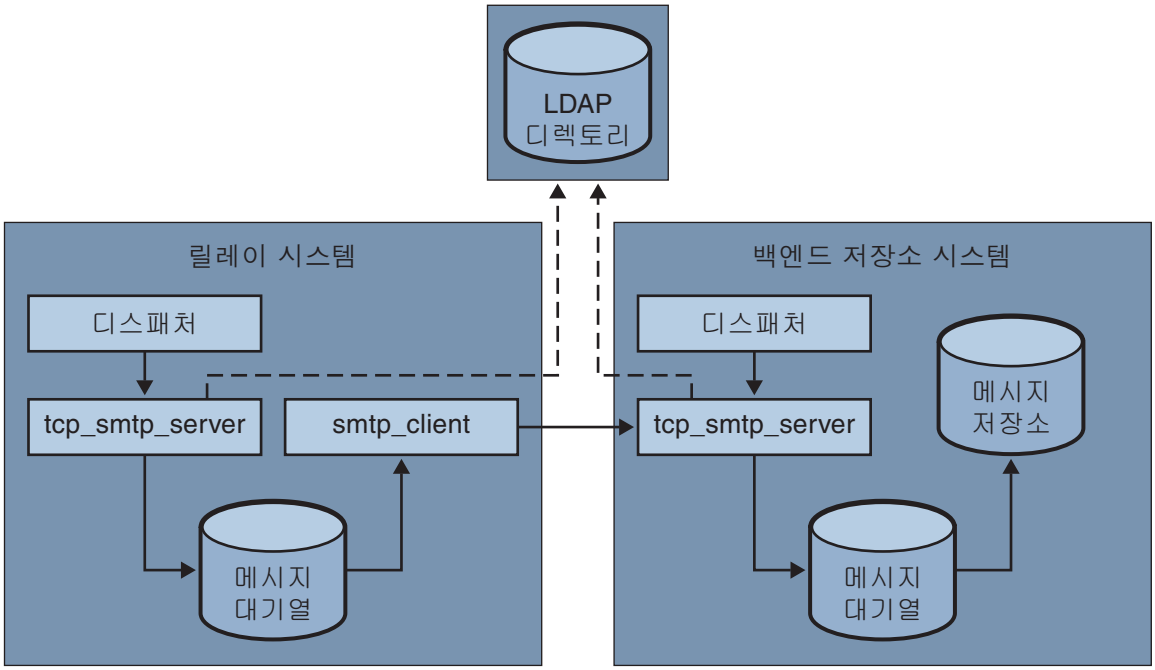


그림 16–1 LMTP를 사용하지 않는 2계층 배포

LMTP 없이 저장소 시스템의 앞면에 릴레이가 있는 2수준 배포에서 인바운드 메시지 처리는 릴레이 시스템의 디스패처가 선택하고 `tcp_smtp_server` 프로세스에 전달되는 SMTP 포트에 대한 연결에서 시작됩니다. 이 프로세스에서는 인바운드 메시지에 대해 다음을 포함한 많은 작업을 수행합니다.

- 디렉토리에서 사용자 조회
- 사용자가 이 전자 메일 배포로 호스팅되는 도메인 내에 있는지 확인
- 사용자가 도메인에 유효한 사용자인지 확인
- 메일 봉투 주소를 `@mailhost:user@domain`으로 다시 쓰기
- 메일 호스트에 전달할 메시지 대기열에 포함시키기

그러면 `smtp_client` 프로세스가 대기열에서 메일 메시지를 선택하여 메일 호스트에 보냅니다. 메일 호스트에서도 비슷한 프로세스가 수행됩니다. 디스패처에서 SMTP에 대한 연결을 선택하여 `tcp_smtp_server` 프로세스에 전달합니다. 이 프로세스에서는 다음을 포함하여 많은 메시지 작업을 수행합니다.

- 디렉토리에서 사용자 조회
- 사용자가 이 전자 메일 배포로 호스팅되는 도메인 내에 있는지 확인
- 사용자가 도메인에 유효한 사용자인지 확인
- 메시지를 `ims_ms` 채널에 전달하도록 메일 봉투 주소 다시 쓰기
- 저장소에 전달할 메시지 대기열에 포함시키기

그런 다음 `ims_ms` 프로세스가 메일 메시지를 선택하여 저장소에 전달하려고 시도합니다. 이 시나리오에서는 대기열에 포함시키는 프로세스가 두 번 수행되고 각 MTA에서 LDAP 조회를 수행합니다.

16.3 LMTP를 사용하는 2계층 배포의 메시징 처리

491 페이지 “16.3 LMTP를 사용하는 2계층 배포의 메시징 처리”는 LMTP를 사용하는 2계층 배포 시나리오의 다음 메시지 처리를 그림 형식으로 설명합니다.

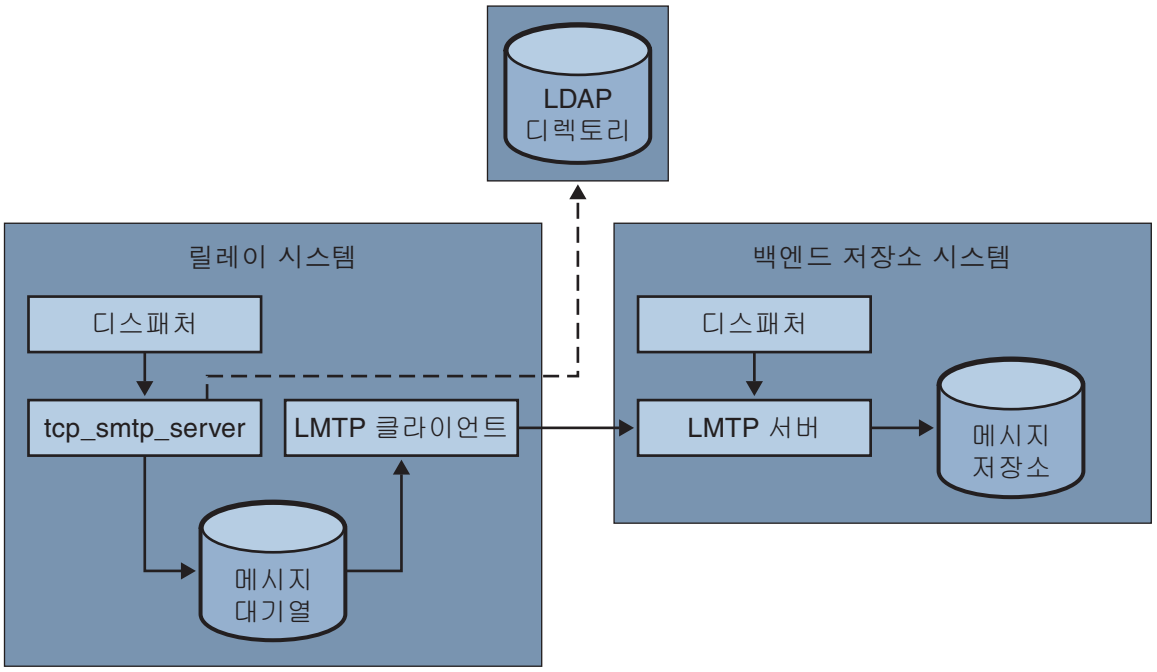


그림 16–2 LMTP를 사용하는 2계층 배포

LMTP를 배치하고 디스패처에서 중계 시스템의 SMTP에 대한 연결을 선택하여 `tcp_smtp_server` 프로세스에 전달합니다. 이 프로세스에서는 인바운드 메시지에 대해 다음을 포함한 많은 작업을 수행합니다.

- 디렉토리에서 사용자 조회
- 사용자가 이 전자 메일 배포로 호스팅되는 도메인 내에 있는지 확인
- 사용자가 도메인에 유효한 사용자인지 확인
- 사용자에 대한 메일함을 호스팅하는 백엔드 메시지 저장소 시스템 결정
- 메일 호스트에 전달할 메시지 대기열에 포함시키기

저장소 시스템에서는 디스패처가 LMTP 포트에 대한 연결을 수신한 다음 `lmtp_server` 프로세스에 전달합니다. 그런 다음 LMTP가 메시지를 사용자의 메일함이나 UNIX 원시 메일함에 넣습니다. 메시지가 성공적으로 전달되면 메시지가 중계 시스템의 대기열에서 제거됩니다. 성공적으로 전달되지 않은 경우 메시지가 중계 시스템에 그대로 남아 있습니다. 메시지 저장소의 LMTP 프로세스에서는 주소 또는 메시지 처리를 위해 MTA 방법을 사용하지 않습니다.

16.4 LMTP 개요

대부분의 경우 MTA 자체는 기본적으로 백엔드 서버에 존재하지 않을 수 있습니다. 필요한 유일한 MTA 구성 요소는 다음과 같습니다.

- 디스패처
- `libimta`
- LMTP 서버
- `imta.cnf`파일
- `mappings` 파일
- `imta.tailor` 파일

디스패처에는 MTA 구성 파일이 필요한데, 이러한 파일은 너무 짧을 수 있습니다. 디스패처를 백엔드 서버에서 실행해야 LMTP 서버를 시작할 수 있습니다. 디스패처와 LMTP 서버는 `libimta`의 다양한 기능을 사용하기 때문에 백엔드 서버에 위치할 필요가 없습니다.

LMTP 서버는 일반적인 MTA 대기열에 포함 또는 대기열에서 제외 기능, 헤더 처리 또는 주소 변환 작업을 수행하지 않습니다. 중계 시스템이 모든 메시지 내용 조작을 수행합니다. 이러한 조작을 통해 메시지를 메시지 저장소에 전달할 형식으로 표시하고 저장소에 필요한 형식으로 된 전달 주소를 표시합니다. 사용자 할당량과 같이 메시지를 저장소에 전달할 때 일반적으로 사용할 수 있는 추가 수신자 정보는 수신자 주소와 함께 LMTP 매개 변수로 표시됩니다. 전달이 실패할 경우 메시지가 중계 시스템의 LMTP 대기열에 그대로 포함되어 있습니다.

16.5 LMTP 전달 구성

LMTP 전달 기법은 릴레이 시스템과 백엔드 저장소 모두에서 구성해야 합니다. 릴레이 시스템에서는 저장소에 전달할 메시지가 LMTP 채널에 전달되도록 `DELIVERY_OPTIONS` MTA 옵션(`option.dat`에서)을 변경해야 합니다. 백엔드 저장소는 디스패처를 사용하여 구성해야 하지만 작업 제어기는 필요하지 않습니다. 디스패처는 LMTP 서버를 실행하도록 구성해야 합니다.

일반적인 다중 계층 배포에서는 서로 다른 백엔드 메시지 저장소 시스템에 사용자가 제공됩니다. 이 백엔드 시스템 중 하나 이상에 LMTP가 설정되지 않을 수 있으므로 프런트엔드 릴레이에서는 LMTP를 인식하는 저장소 시스템을 알고 있어야 합니다. 이는 LMTP 전달을 수락하도록 구성된 메시지 저장소를 명시적으로 지정하는 일반 데이터베이스 기능을사용하면 가능합니다.

▼ LMTP를 사용하는 인바운드 MTA 중계 구성

LMTP를 사용하도록 인바운드 MTA 릴레이를 구성하려면 다음을 수행합니다.

1 **imta.cnf 파일을 수정하여 LMTP 다시 쓰기 규칙을 다음과 같이 변경합니다.**

```
! lmtp
.lmtp   $E$F$U%$H.lmtp@lmtpcs-daemon
.lmtp   $B$F$U%$H@$H@lmtpcs-daemon
!
! lmtp native
.lmtpn  $E$F$U%$H.lmtpn@lmtpcn-daemon
.lmtpn  $B$F$U%$H@$H@lmtpcn-daemon
!
```

2 **DELIVERY_OPTIONS 메일함을 다음과 같이 설정합니다.**

```
#*mailbox=@$X.LMTP:$M%$\$2I$_+$2S@lmtpcs-daemon
```

3 **원시 DELIVERY_OPTIONS 절을 다음과 같이 설정합니다.**

```
#*native=@$X.LMTPN:$M+$2S@native-daemon
```

4 **채널 키워드 multigate connectcanonical을 각 tcp_lmtp* 채널 블록에 추가합니다.**

5 **다음 채널 키워드를 tcp_lmtpcs 채널에 추가합니다.**

```
fileinto @$4O:$U+$S@$D
```

위 키워드에서 'O'는 숫자 0이 아닌 대문자 O입니다.

6 **받는 MTA 릴레이 구성 설정은 다음과 같습니다.**

option.dat의 DELIVERY_OPTIONS 항목은 다음과 같습니다.

```
!-----------------------------------------
! Modified DELIVERY_OPTIONS to activate LMTP
! delivery from a frontend to the backend store
!-----------------------------------------
!
DELIVERY_OPTIONS=\
    #*mailbox=@$X.LMTP:$M%$\$2I$_+$2S@lmtpcs-daemon,\
    #&members=*,\
    #*native=@$X.LMTPN:$M+$2S@native-daemon,\
    #*unix=@$X.LMTPN:$M,\
    #*file=@$X.LMTPN:+$F,\
    #&@members_offline=*,\
    #/hold=@hold-daemon:$A,\
    #program=$M%$P@pipe-daemon,\
    #forward=**,\
    #*^!autoreply=$M+$D@bitbucket
!
```

변경이 끝난 후 수정된 imta.cnf 다시 쓰기 규칙은 다음과 같습니다.

```
! lmtp
.lmtp   $E$F$U%$H.lmtp@lmtpcs-daemon
.lmtp   $B$F$U%$H@$H@lmtpcs-daemon
!
! lmtp native
.lmtpn  $E$F$U%$H.lmtpn@lmtpcn-daemon
.lmtpn  $B$F$U%$H@$H@lmtpcn-daemon
!
```

변경된 채널 블록은 다음과 같아야 합니다.

```
!
! tcp_lmtpcs (LMTP client - store)
tcp_lmtpcs defragment lmtp  multigate connectcanonical \
   fileinto @$4O:$U+$S@$D port 225 nodns single_sys \
   subdirs 20 maxjobs 7 pool SMTP_POOL dequeue_removeroute
lmtpcs-daemon

!
! tcp_lmtpcn (LMTP client - native)
tcp_lmtpcn defragment lmtp multigate connectcanonical port 226 \
   nodns single_sys subdirs 20 maxjobs 7 pool SMTP_POOL \
   dequeue_removeroute
lmtpcn-daemon
```

16.5.1 LMTP 및 최소 MTA와 함께 백엔드 저장소를 구성하는 방법

LMTP를 통해 메시지를 받는 경우 백엔드 저장소에서는 최소 MTA만 필요합니다. 디스패처, 작업 제어기 및 단순 MTA 구성이 필요하며 특히 MTA 구성의 유일한 중요 부분을 구성하는 dispatcher.cnf, job_controller.cnf 및 mappings 파일이 필요합니다.

dispatcher.cnf 파일에는 다음이 포함되어 있어야 합니다.

```
! VERSION=1.1
! IMTA default dispatcher configuration file
!
! Global defaults
!
MIN_PROCS=1
MAX_PROCS=10
MIN_CONNS=30
MAX_CONNS=50
MAX_SHUTDOWN=2
```

```
MAX_LIFE_TIME=86400
MAX_LIFE_CONNS=10000
MAX_IDLE_TIME=600
HISTORICAL_TIME=0
!
! rfc 2033 LMTP server - store
!
[SERVICE=LMTPSS]
PORT=225
IMAGE=IMTA_BIN:tcp_lmtp_server
LOGFILE=IMTA_LOG:tcp_lmtpss_server.log
PARAMETER=CHANNEL=tcp_lmtpss
STACKSIZE=2048000
! Uncomment the following line and set INTERFACE_ADDRESS to an
! appropriate host IP (dotted quad) if the dispatcher needs to
! listen on a specific interface (e.g. in a HA environment).
! INTERFACE_ADDRESS=!
! rfc 2033 LMTP server - native
!
[SERVICE=LMTPSN]
PORT=226
IMAGE=IMTA_BIN:tcp_lmtpn_server
LOGFILE=IMTA_LOG:tcp_lmtpsn_server.log
PARAMETER=CHANNEL=tcp_lmtpsn
STACKSIZE=2048000
```

기본적으로 dispatcher.cnf 파일의 LMTP 서비스는 주석 처리됩니다. LMTP가 작동하려면 이러한 주석 처리를 제거해야 합니다.

MAX_CONNS, MAX_PROCS, MAX_LIFE_CONNS 및 MAX_LIFE_TIME의 일반 디스패처 옵션도 지정할 수 있지만 그럴 경우 하드웨어에 맞게 설정해야 합니다.

PORT_ACCESS 매핑이 중요합니다. 백엔드 서버에 대한 LMTP 구현은 Sun Java System Messaging Server 릴레이 시스템과 백엔드 저장소 사이의 개인 프로토콜로 사용됩니다. PORT_ACCESS 매핑을 사용하여 그런 릴레이만 이러한 서비스에 연결될 수 있도록 확인해야 합니다. 매핑 파일의 모양은 다음과 같습니다.

```
PORT_ACCESS

  TCP|*|225|192.18.74.206|* $Y
  TCP|*|226|192.18.74.206|* $Y
  TCP|*|225|192.18.74.129|* $Y
  TCP|*|226|192.18.74.129|* $Y
  TCP|*|*|*|*   $N500$ Do$ not$ connect$ to$ this$ machine
```

위 IP 주소는 LMTP 서버 및 클라이언트 IP 주소입니다. 이 PORT_ACCESS 매핑 테이블에 지정된 샘플 IP 주소를 백엔드 저장소에 연결되는 네트워크에 있는 릴레이 시스템의 IP 주소로 바꾸어야 합니다.

imta.cnf 파일이 있어야 하지만 이것만으로 완벽한 구성이 이뤄지지는 않습니다. 최소 imta.cnf 파일은 다음 채널 정의로 구성됩니다.

```
!
! IMTA configuration file
!
! tcp_lmtpss (LMTP server - store)
tcp_lmtpss lmtp
tcp_lmtpss-daemon

!
! tcp_lmtpsn (LMTP server - native)
tcp_lmtpsn lmtp
tcp_lmtpsn-daemon
```

기본적으로 LMTP 채널 정의는 주석 처리됩니다. LMTP를 작동하려면 LMTP의 주석 처리를 제거해야 합니다.

설치 시 생성되는 기본 job_controller.cnf 파일을 사용할 수 있습니다. 이 파일을 수정할 필요가 없습니다.

16.5.2 LMTP를 통해 메시지 저장소와 전체 MTA를 갖는 백엔드 시스템에 메시지를 보내도록 중계 구성

백엔드 저장소에 MTA의 전체 기능을 제공하면서 LMTP를 사용하여 로드를 절약해야 하는 경우가 있습니다. 예를 들어, 백엔드 저장소에서 프로그램을 전달할 수 있습니다. 이 경우에는 위의 494 페이지 “LMTP를 사용하는 인바운드 MTA 중계 구성”에서 설명한 대로 릴레이를 구성해야 합니다.

16.5.3 전체 MTA가 있는 백엔드 메시지 저장소 시스템의 LMTP 구성

백엔드 저장소 메시징 시스템 구성은 LMTP를 사용하여 저장소에 직접 전달하는 구성에서 dispatcher.cnf 파일의 끝에 다음 행이 추가되는 점만 다릅니다.

```
! rfc 2033 LMTP server - store
![SERVICE=LMTPSS]
PORT=225
IMAGE=IMTA_BIN:tcp_lmtp_server
LOGFILE=IMTA_LOG:tcp_lmtpss_server.log
PARAMETER=CHANNEL=tcp_lmtpss
STACKSIZE=2048000
! Uncomment the following line and set INTERFACE_ADDRESS to an
```

```
! appropriate host IP (dotted quad) if the dispatcher needs to
! listen on a specific interface (e.g. in a HA environment).
!INTERFACE_ADDRESS=
!
! rfc 2033 LMTP server - native
!
[SERVICE=LMTPSN]
PORT=226
IMAGE=IMTA_BIN:tcp_lmtpn_server
LOGFILE=IMTA_LOG:tcp_lmtpsn_server.log
PARAMETER=CHANNEL=tcp_lmtpsn
STACKSIZE=2048000
! Uncomment the following line and set INTERFACE_ADDRESS to an
! appropriate host IP (dotted quad) if the dispatcher needs to
! listen on a specific! interface (e.g. in a HA environment).
!INTERFACE_ADDRESS=
!
```

기본적으로 `dispatcher.cnf` 파일의 LMTP 서비스는 주석 처리됩니다. LMTP가 작동하려면 이러한 주석 처리를 제거해야 합니다. 또한 LMTP 포트 번호는 예일 뿐이므로 사용자가 선택한 임의의 번호가 될 수 있습니다.

백엔드 저장소를 LMTP에 대해서만 구성할 경우에는 위에서 설명한 `dispatcher.cnf` 파일과 동일합니다. 또한, 매핑 파일에는 LMTP 전용 백엔드 저장소에 대해 설명한 `PORT_ACCESS` 매핑이 필요합니다.

16.5.4 LMTP 메시지 데이터에 대한 응답 시 4.2.1 메일함 사용 중 오류 처리

LMTP 채널 옵션 `MAILBOX_BUSY_FAST_RETRY`가 1(기본값)로 설정된 경우, LMTP 메시지 데이터에 대한 응답 시 4.2.1 메일함 사용 중 오류는 임의의 짧은 간격 후에 메시지를 재시도하여 처리됩니다. 일반 메시지 backoff 값은 적용되지 않습니다. 이 옵션을 0으로 설정하면 이 동작이 비활성화됩니다.

16.6 구현된 LMTP 프로토콜

이 절에서는 샘플 LMTP 대화 상자를 제공하여 해당 대화 상자에 표시되는 내용을 설명합니다. 중계 시스템의 LMTP 클라이언트는 표준 LMTP 프로토콜을 사용하여 백엔드 저장소의 MLTP 서버와 대화합니다. 프로토콜은 특정 방법으로 사용됩니다. 예를 들면 다음과 같습니다.

```
---> LHLO
<--- 250 OK
```

LHLO 메시지에 대한 작업이 수행되지 않습니다. 회신은 항상 250 OK입니다.

```
---> MAIL FROM: address size=messageSizeInBytes
<--- 250 OK
```

메시지 발송자 주소를 검사하거나 변환하지 않습니다. size= 매개 변수는 전달할 메시지의 크기(바이트)를 지정합니다. 이 값은 프로토콜에 표시되는 메시지의 정확한 크기입니다. 정확한 메시지 크기가 꼭 필요한 것은 아니지만 대개 실제 메시지 크기는 이 크기를 초과하지 않습니다. LMTP 서버는 메시지를 받도록 메모리에 이 크기의 버퍼를 할당합니다.

```
---> RCPT TO: uid+folder@domain xquota=size,number xdflg=xxx
<--- 250 OK
```

받을 때는 수신자 주소를 확인하지 않지만 나중에 사용할 수 있도록 수신자 목록이 작성됩니다. 주소의 @domain 부분은 주 도메인의 uids에서는 생략되고 +folder 부분은 선택 사항입니다. 이 형식은 MTA의 메시지 저장소 채널에 사용되는 것과 동일한 주소 형식입니다.

xquota= 매개 변수는 최대 총 크기와 최대 메시지 수로 구성되는 사용자의 메시지 할당량을 지정합니다. MTA는 사용자에 대한 LDAP 조회를 통해 주소 변환을 수행하는 동안 검색되는 이 메시지 할당량 정보를 제공합니다. 이 정보는 메시지 저장소의 할당량 정보를 디렉토리와 동기화된 상태로 유지하는 데 사용됩니다. 할당량 정보를 가져오는 것은 성능에 영향을 미치지 않습니다.

xdflg= 매개 변수는 비트 필드로 해석되는 숫자를 지정합니다. 이러한 비트 수는 메시지가 전달되는 방법을 제어합니다. 예를 들어, 비트 값을 2로 설정하면 사용자에 대해 할당량이 초과하더라도 메시지가 전달됩니다. (xdflg는 내부 매개 변수이며 포함된 비트가 예고 없이 변경되거나 추가될 수 있다는 것에 주의합니다. Sun 서버에서 이 확장을 사용하는 다른 클라이언트나 일부 다른 서버에서 이 매개 변수를 사용하는 Sun 클라이언트나 모두 지원되지 않습니다.)

이 상호 작용은 수신자마다 한 번씩 여러 번 반복될 수 있습니다.

```
--->DATA
---> <the message text>
--->.
```

그런 다음 LMTP 클라이언트가 SMTP에서처럼 전체 메시지를 점으로 표시하여 보냅니다. 메시지는 한 행에 점(.) 하나로 끝납니다. 메시지 크기가 초과될 경우 LMTP 서버는 다음을 보냅니다.

```
<--- 500 message too big
```

그런 다음 연결을 종료합니다.

메시지가 올바르게 전달되면 LMTP 서버는 RCPT TO: 행에 지정된 각 수신자에 대한 상태를 LMTP 클라이언트에게 다시 보냅니다. 예를 들어, 메시지가 성공적으로 전달될 경우 다음과 같은 응답을 받습니다.

```
<--- 250 2.5.0 address OK
```

여기서 address는 RCPT TO: 행에 표시된 주소입니다.

변환은 다른 MAIL FROM: 행에서 반복되거나 다음 상호 작용으로 종료될 수 있습니다.

```
---> quit
<--- 221 OK
```

표 16–1에서는 각 수신자에 대해 가능한 상태 코드를 나타냅니다. 이 3열 테이블의 첫 번째 열에는 짧은 코드가 표시되고, 두 번째 열에는 긴 코드가 표시되며, 세 번째 열에는 상태 텍스트가 표시됩니다. 2.x.x 상태 코드는 성공 코드이고, 4.x.x 코드는 재시도 가능한 오류이고, 5.x.x 코드는 재시도할 수 없는 오류입니다.

표 16–1 수신자에 대한 LMTP 상태 코드

짧은 코드	긴 코드	상태 텍스트
250	2.5.0	확인
420	4.2.0	메일함 잠김
422	4.2.2	할당량 초과
420	4.2.0	잘못된 메일함 형식
420	4.2.0	메일함 지원 안 함
430	4.3.0	IMAP IOERROR
522	5.2.2	지속적인 할당량 초과
523	5.2.3	길이가 너무 긴 메시지
511	5.1.1	메일함 없음
560	5.6.0	메시지에 null 포함
560	5.6.0	메시지에 nl 포함
560	5.6.0	메시지에 잘못된 헤더 있음
560	5.6.0	메시지에 빈 행 없음

그렇지 않으면 메일함, 원시(UNIX) 및 파일에 대한 전달 옵션이 변경된 것입니다. 이러한 규칙의 목적은 메시지가 해당 LMTP 채널을 통해 백엔드 서버에 전달되도록 주소를 생성하는 것입니다. 생성된 주소는 라우팅된 원본 주소이며 그 형식은 다음과 같습니다.

`@sourceroute:`*localpart*`@`*domain*

◆ ◆ ◆ 17장

17 휴가 자동 메시지 회신

자동으로 생성된 전자 메일 응답(자동 회신), 특히 휴가 메시지에 대해 MTA는 *MDN*(Message Disposition Notification) 및 시브(Sieve) 스크립트 언어를 사용합니다. MDN은 메시지의 전달 처리를 보고하기 위해 MTA가 보낸 사람 및/또는 포스트마스터에게 보내는 전자 메일입니다. MDN은 또한 읽음 확인, 확인, 수신 알림 또는 전달 확인이라고도 합니다. 시브(Sieve)는 메일 필터를 만드는 데 사용되는 간단한 스크립트 언어입니다. Messaging Server 5.x와 달리 문자 세트로 ISO-2022-JP 대신 UTF-8이 사용됩니다.

이 장에서는 휴가 자동 회신 기법에 대해 설명합니다. 대부분의 경우 기본 구성은 수정할 필요가 없지만 백엔드 메시지 저장소가 아니라 MTA 릴레이 시스템에서 휴가 처리를 수행하도록 시스템을 구성해야 할 수 있습니다.

이 장은 다음 내용으로 구성되어 있습니다.

17.1 휴가 자동 회신 개요

휴가 시브(Sieve) 스크립트는 다양한 LDAP 휴가 속성으로부터 자동으로 생성됩니다(507 페이지 "17.4 휴가 자동 회신 속성" 참조). 또한 부가적인 유연성을 위해 이러한 스크립트를 명시적으로 지정할 수도 있습니다. 기본 휴가 추적 기법은 다양한 보낸 사람에게 회신을 보낸 시간을 추적하는 파일 집합(원하는 수신자 당 하나)입니다.

주 – 휴가 메시지의 문자 세트가 UTF-8로 변경되었습니다.

기본적으로 MTA는 백엔드 저장소 시스템에서 휴가를 평가합니다. 그러나 MTA 릴레이가 백엔드 저장소처럼 많은 작업을 처리하는 것은 아니기 때문에 백엔드 저장소

대신 메일 릴레이 시스템에서 휴가 메일을 평가하도록 하여 성능을 높일 수 있습니다. 이 기능을 사용하면 각각의 릴레이에서 서로 다른 메시지를 처리하기 때문에 원하는 것보다 더 많은 수의 휴가 응답이 보내질 수 있습니다. 휴가 메시지를 원하는 수보다 더 많이 보내지 않게 하려면 릴레이 간에 파일 추적을 공유할 수 있습니다. 이 방법도 충분하지 않다고 생각될 경우에는 휴가 메시지가 백엔드 저장소 시스템에서 항상 평가되게 할 수 있습니다.

17.2 자동 회신 구성

전달 주소가 일련의 패턴을 통해 생성됩니다. 사용되는 패턴은 `mailDeliveryOption` 속성에 정의된 값에 따라 다릅니다. 전달 주소는 각각의 유효한 `mailDeliveryOption`에 대해 생성됩니다. 패턴은 MTA 옵션 `DELIVERY_OPTIONS`에 의해 `option.dat` 파일에 정의됩니다. `option.dat` 파일의 `DELIVERY_OPTIONS`에 있는 기본 자동 회신 규칙은 다음과 같습니다.

```
*^!autoreply=$M+$D@bitbucket
```

MTA는 자동 회신 `DELIVERY_OPTION` MTA 옵션에 “^” 기호를 기록합니다. 그렇게 하면 MTA가 휴가 날짜를 확인하게 됩니다. 현재 날짜가 휴가 날짜 범위에 속하는 경우 처리가 계속되며, MTA는 자동 회신 `DELIVERY_OPTION`에 "!"를 기록합니다. 그런 다음 사용자가 입력하는 다양한 자동 회신 LDAP 속성을 기반으로 휴가 시브(Sieve) 스크립트를 만듭니다. 자동 회신 규칙에는 접두어 문자 ”!’, ”#’, ”^’ 및 ”*’가 있습니다.

메일함 전달 옵션에 ”!’ 플래그를 표시할 수 있습니다. 그럴 경우 휴가 스크립트 생성이 무조건적으로 사용됩니다. ”^’ 플래그에 의해 추가로 금지할 수 있도록 자동 회신 방법이 별도의 전달 옵션에 의해 활성화됩니다. 이 단계에서 날짜를 확인하는 것이 시브(Sieve) 논리를 사용하는 것보다 더 효과적입니다.

표 17–1에는 첫 번째 열의 자동 회신 규칙과 두 번째 열의 해당 정의에 사용되는 접두어 문자가 표시되어 있습니다.

표 17–1 DELIVERY_OPTIONS의 자동 회신 규칙에 사용되는 접두어 문자

접두어 문자	정의
!	자동 회신 시브(Sieve) 스크립트 생성을 사용합니다.
#	릴레이에서 프로세스를 처리하도록 허용합니다.
^	휴가 날짜에 옵션을 평가하도록 표시된 경우에만 옵션을 평가합니다.
@	다양한 메시지 헤더 필드와 봉투의 `From:` 주소에 연결된 LDAP 항목으로부터 원하는 언어 정보를 추출합니다. 이 정보를 정확한 시간에 사용하려면 자동 회신을 사용할 때 reprocess 채널을 통해 메시지를 전달해야 합니다. `autoreply` 전달 옵션에 @ 플래그를 추가하면 됩니다. 채널 홉을 추가하면 메시지 처리 오버헤드가 커집니다.

자동 회신 규칙은 bitbucket 채널의 대상 주소를 지정합니다. 자동 회신이 생성되면 이 방법에 의해 메일이 전달된 것으로 간주되지만 MTA 방법을 사용하려면 전달 주소가 필요합니다. bitbucket 채널에 전달되는 내용은 삭제됩니다.

17.2.1 백엔드 시스템에서 자동 회신 구성

DELIVERY_OPTIONS의 기본 자동 회신 규칙은 사용자에게 서비스를 제공하는 메일 서버에서 자동 회신을 처리합니다. 백엔드 저장소 시스템에서 휴가 메시지가 평가되도록 하려면 아무 것도 구성할 필요가 없습니다. 기본 동작입니다.

▼ 릴레이에서 자동 회신 구성 방법

성능 향상을 위해 백엔드 저장소 시스템 대신 릴레이에서 휴가를 평가하려면 option.dat 파일을 편집하고 DELIVERY_OPTIONS의 자동 회신 규칙 앞에 # 문자를 붙입니다.

1 **an 편집기를 사용하여 option.dat 파일을 엽니다.**

2 **자동 회신 규칙이 다음과 같이 표시되도록 DELIVERY_OPTIONS 옵션을 추가하거나 변경합니다.**

```
#*^!autoreply=$M+$D@bitbucket
```

기본 DELIVERY_OPTIONS 옵션은 다음과 같습니다.

```
DELIVERY_OPTIONS=*mailbox=$M%$\$2I$_+$2S@ims-ms-daemon, \
 &members=*, \
 *native=$M@native-daemon, \
 /hold=@hold-daemon:$A, \
 *unix=$M@native-daemon, \
 &file=+$F@native-daemon, \
 &@members_offline=* \
 ,program=$M%$P@pipe-daemon, \
 #forward=**, \
 *^!autoreply=$M+$D@bitbucket
```

이렇게 하면 릴레이에서 프로세스를 처리할 수 있습니다. MTA가 릴레이에서 자동 회신을 수행하도록 지정한 경우 특정 사용자가 최근에 휴가 알림 메시지를 보냈는지 여부를 각 릴레이에서 독립적으로 추적하도록 하거나 또는 이 정보를 릴레이 간에 공유하도록 할 수 있습니다. 전자의 경우가 더 간단합니다. 특히, 휴가 알림 메시지를 보내는 횟수가 문제가 되지 않는 경우에 그렇습니다. 휴가 알림 메시지를 보내는 빈도 규칙을 엄격하게 적용하려면 릴레이 간에 정보를 공유해야 합니다. 릴레이 간에 정보를 공유하려면 파일이 NFS에 마운트되어 있어야 합니다. NFS 마운팅에 대한 자세한 내용은 379 페이지 “12.8.2.3 조각 모음 및 휴가 캐싱에 NFS 기반 파일 시스템 사용”을 참조하십시오.

이 파일의 위치는 VACATION_TEMPLATE 옵션에 의해 제어됩니다. option.dat 파일에서 이 옵션을 /<path>/%A로 설정해야 합니다. 여기서 <path>는 여러 릴레이 시스템 간에 공유되는 디렉토리의 경로입니다. 템플리트는 file:URL이어야 하며, $U를 사용하여 사용자 이름을 대체해야 합니다. 기본 설정은 다음과 같습니다.

```
VACATION_TEMPLATE=file:///opt/SUNWmsgsr/data/vacation/$3I/$1U/$2U/$U.vac
```

메타 문자에 대한 설명은 표 9–6을 참조하십시오.

주 – 이제 휴가 파일 템플리트에 UID에 대한 액세스 권한이 있으므로 사용자 의 UID를 기반으로 작성될 휴가 파일에 경로를 허용하게 됩니다. 추가적으로, 휴가 파일 경로를 확인하는 데 사용된 주소는 사용자의 메일 속성에 저장된 주소이며, 현재 수신자 주소는 이전에 사용되었습니다.

17.3 휴가 자동 회신 작동 원리

휴가 작업을 호출하면 다음과 같이 작동합니다.

1. Sun Java System Messaging Server에서 휴가 작업이 시스템 수준 시브(Sieve) 스크립트가 아니라 사용자 수준으로 수행되었는지 확인합니다. 휴가가 시스템 수준 스크립트에서 사용되는 경우 오류가 발생합니다.
2. "휴가 알림 없음" 내부 MTA 플래그가 선택됩니다. 이 플래그를 설정하면 프로세스가 종료되고 휴가 알림이 보내지지 않습니다.
3. 메시지에 대한 반송 주소가 선택됩니다. 이 주소가 비어 있으면 프로세스가 종료되고 휴가 알림이 보내지지 않습니다.
4. MTA가 :addresses 태그 인수에 지정된 사용자 주소 또는 추가 주소가 현재 메시지의 To:, Cc:, Resent-to: 또는 Resent-cc: 헤더 필드에 표시되는지 확인합니다. 헤더 필드에 주소가 없는 경우 프로세스가 종료되고 휴가 알림이 보내지지 않습니다.
5. Messaging Server는 :subject 인수와 이유 문자열을 구성합니다. 이 문자열은 현재 메시지의 반송 주소와 함께 이전 휴가 응답의 사용자 단위 레코드에 대해 확인됩니다. :days 인수에 허용된 시간 내에 응답이 이미 보내진 경우 프로세스가 종료되고 응답이 보내지지 않습니다.
6. Messaging Server는 :subject 인수, 이유 문자열 및 :mime 인수로부터 휴가 알림을 만듭니다. 이 응답 메시지에 다음과 같은 두 기본 형식을 사용할 수 있습니다.
   - RFC 2298에 지정된 형식의 메시지 배포 알림(첫 번째 부분에 이유 텍스트가 포함되어 있음)
   - 단일 부분 텍스트 회신. 이 형식은 "회신" 자동 회신 모드 속성 설정을 지원하는 데만 사용됩니다.

Messenger Express를 통해 휴가 메시지가 구성된 경우 mailautoreplymode가 자동으로 reply로 설정된다는 것에 주의하십시오.

"휴가 알림 없음" MTA 플래그는 기본적으로 선택되어 있지 않습니다. 이 플래그는 비표준 novacation 작업을 사용하여 시스템 수준 시브(Sieve) 스크립트로 설정할 수 있습니다. novacation 시브(Sieve) 작업은 시스템 수준 시브(Sieve) 스크립트에서만 허용됩니다. 사용자 수준 스크립트에서 이 플래그를 사용하면 오류가 발생합니다. 이 작업을 사용하여 휴가 회신에 대한 사이트 차원 제한(예: 하위 문자열 "MAILER-DAEMON"이 포함된 주소에 대한 회신 차단)을 구현할 수 있습니다.

사용자 단위 응답별 정보는 로컬 사용자 당 하나씩 일반 텍스트 파일에 저장됩니다. 이러한 파일의 위치 및 이름 지정 방법은 VACATION_TEMPLATE MTA 옵션 설정에 지정됩니다. 이 옵션을 file: URL로 설정해야 합니다.

이러한 파일은 자동으로 유지 관리되며 VACATION_CLEANUP 정수 MTA 옵션 설정에 의해 제어됩니다. 이러한 파일 중 하나가 열릴 때마다 현재 시간(초) modulo 값이 계산됩니다. 결과가 0이면 파일이 스캔되고 모든 만료된 항목이 제거됩니다. 옵션 기본값은 200이고 200번에 한 번씩 정리가 수행됨을 의미합니다.

이러한 일반 텍스트 파일을 읽고 쓰는 데 사용되는 방법은 NFS에 대해 올바르게 작업을 수행할 수 있는 방식으로 디자인됩니다. 그렇게 하면 여러 MTA에서 공통 파일 시스템에서 단일 파일 집합을 공유할 수 있습니다.

17.4 휴가 자동 회신 속성

휴가 작업에 사용되는 사용자 LDAP 디렉토리 속성 집합은 다음과 같습니다.

- MTA 옵션 LDAP_AUTOREPLY_ADDRESSES에서 정의한 속성

  이 속성은 시브(Sieve) 휴가에 :addresses 인수를 생성하는 기능을 제공합니다. 이 옵션은 기본적으로 값이 없습니다. 이 속성은 여러 값을 가질 수 있습니다. 각 값은 :addresses 휴가 매개 변수에 전달할 개별 주소를 지정합니다.

- LDAP_PERSONAL_NAME에서 정의한 속성

  별칭 처리는 이 속성에 지정된 개인 이름 정보를 추적하며 이 정보를 사용하여 생성되는 모든 MDN 또는 휴가 회신의 From: 필드를 구성합니다. 개인 정보가 노출되지 않도록 주의해서 사용합니다.

- vacationStartDate

  휴가 시작 날짜 및 시간입니다. 이 값은 YYYYMMDDHHMMSSZ 형식입니다. 이 값은 GMT로 정규화됩니다. 자동 회신은 현재 시간이 이 속성에 지정된 시간 이후인 경우에만 생성되어야 합니다. 이 속성이 없는 경우 시작 날짜가 적용되지 않습니다. LDAP_START_DATE MTA 옵션을 다른 속성 이름으로 설정하여 이 정보에 대해 다른 속성을 조사하도록 MTA에 명령할 수 있습니다.

  시브(Sieve) 스크립트를 생성한 코드에서 이 속성을 읽고 확인합니다. 현재 날짜가 휴가 시작 날짜 이전이면 휴가 프로세스가 중지됩니다. 현재 시브(Sieve)에 날짜/시간 테스트 및 비교 기능이 없기 때문에 스크립트 자체로는 이 속성을 처리할 수 없습니다.

- vacationEndDate

  휴가 종료 날짜 및 시간입니다. 이 값은 YYYYMMDDHHMMSSZ 형식입니다. 이 값은 GMT로 정규화됩니다. 자동 회신은 현재 시간이 이 속성에 지정된 시간 이전인 경우에만 생성되어야 합니다. 이 속성이 없는 경우 종료 날짜가 적용되지 않습니다. LDAP_END_DATE MTA 옵션을 다른 속성 이름으로 설정하여 이 정보에 대해 다른 속성을 조사하도록 MTA에 명령할 수 있습니다.

  시브(Sieve) 스크립트를 생성한 코드에서 이 속성을 읽고 확인합니다. 현재 날짜가 휴가 종료 날짜 이후이면 휴가 프로세스가 중지됩니다. 현재 시브(Sieve)에 날짜/시간 테스트 및 비교 기능이 없기 때문에 스크립트 자체에서는 이 속성을 처리할 수 없습니다.

- mailAutoReplyMode

  사용자 메일 계정에 대한 자동 회신 모드를 지정합니다. 이 속성에 유효한 값은 다음과 같습니다.

  - echo - 추가된 mailAutoReplyText 또는 mailAutoReplyTextInternal 텍스트와 함께 원본 메시지 텍스트를 반향하는 멀티파트를 만듭니다.
  - reply - 원래의 보낸 사람에게 mailAutoReplyText 또는 mailAutoReplyTextInternal에 지정된 대로 단일 부분 회신을 보냅니다.

    이러한 모드는 시브(Sieve) 스크립트에 휴가 작업에 대한 비표준 :echo 및 :reply 인수로 표시됩니다. echo는 원본 메시지를 반환된 내용에 포함하는 "처리된" MDN(Message Disposition Notification)을 생성합니다. reply는 회신 텍스트만 포함되는 순수 회신을 생성합니다. 잘못된 값이 휴가 작업에 대한 인수로 매니페스트되지 않으므로 원본 메시지의 헤더만 포함된 MDN이 생성됩니다. 에코 자동 회신 모드를 선택하면 최근에 회신을 보낸 방법에 관계 없이 모든 메시지에 대해 자동 회신을 보냅니다.

    LDAP_AUTOREPLY_MODE MTA 옵션을 다른 속성 이름으로 설정하여 이 정보에 다른 속성을 사용하도록 MTA에 명령할 수 있습니다.

- mailAutoReplySubject

  자동 응답에 사용할 제목 필드의 내용을 지정합니다. UTF-8 문자열이어야 합니다. 이 값은 휴가 작업에 :subject 인수로 전달됩니다. LDAP_AUTOREPLY_SUBJECT MTA 옵션을 다른 속성 이름으로 설정하여 이 정보에 다른 속성을 사용하도록 MTA에 명령할 수 있습니다.

- mailAutoReplyText

  수신자의 도메인에 있는 사용자를 제외한 모든 보낸 사람에게 자동 회신 텍스트를 보냅니다. 이 옵션을 지정하지 않는 경우 외부 사용자는 휴가 메시지를 받지 못합니다. LDAP_AUTOREPLY_TEXT MTA 옵션을 다른 속성 이름으로 설정하여 이 정보에 다른 속성을 사용하도록 MTA에 명령할 수 있습니다.

- mailAutoReplyTextInternal

수신자 도메인에 있는 보낸 사람에게 자동 회신 텍스트를 보냅니다. 이 옵션을 지정하지 않는 경우 내부 사용자가 메시지 자동 회신 텍스트 메시지를 받게 됩니다. LDAP_AUTOREPLY_TEXT_INT MTA 옵션을 다른 속성 이름으로 설정하여 이 정보에 다른 속성을 사용하도록 MTA에 명령할 수 있습니다.

MTA는 mailAutoReplyText 또는 mailAutoReplyTextInternal 속성 값을 휴가 작업에 이유 문자열로 전달합니다.

- mailAutoReplyTimeOut

  지정된 메일 보낸 사람에게 연속 자동 응답을 보내는 기간(시간)입니다. mailAutoReplyMode=reply인 경우에만 사용됩니다. 값이 0이면 메시지를 받을 때마다 응답을 보냅니다. 이 값은 휴가 작업에 대한 비표준 :hours 인수로 변환됩니다. 일반적으로 시브(Sieve) 휴가 작업은 이러한 목적으로 :days 인수만을 지원하며 0 값을 허용하지 않습니다.

  이 속성이 사용자 항목에 표시되지 않으면 AUTOREPLY_TIMEOUT_DEFAULT MTA 옵션에서 기본 시간 초과 값을 가져옵니다. LDAP_AUTOREPLY_TIMEOUT MTA 옵션을 설정하여 이 정보에 다른 속성을 사용하도록 MTA에 명령할 수 있습니다.

MTA는 언어 태그가 다른 여러 LDAP 속성과 속성 값 중에서 선택하여, 사용할 정확한 값을 결정할 수 있습니다. 사용되는 언어 태그를 봉투의 보낸 사람 주소와 연결된 기본 언어 정보와 비교합니다. 현재 이 처리를 받는 속성은 LDAP_AUTOREPLY_SUBJECT(보통 mailAutoReplySubject), LDAP_AUTOREPLY_TEXT(보통 mailAutoReplyText), LDAP_AUTOREPLY_TEXT_INT(보통 mailAutoReplyTextInternal), LDAP_SPARE_4, LDAP_SPARE_5, LDAP_PREFIX_TEXT 및 LDAP_SUFFIX_TEXT입니다.

각 속성 값에 다른 언어 태그 값이 지정됩니다. 서로 다른 값에 같은 태그 값이 있는 경우에는 그 중에서 무작위로 선택됩니다.

17.5 기타 자동 회신 작업 및 문제

이 절에서는 구성 절에서 다루지 않은 자동 회신 작업 및 문제에 대해 설명합니다.

17.5.1 Sun Mail Server가 아닌 서버에서 자동 전달된 전자 메일에 대해 자동 회신 메시지 보내기

MTA가 Sun 시스템이 아닌 시스템으로부터 자동 전달된 메시지를 받을 경우 자동 회신 문제가 발생할 수 있습니다. 예를 들어 sesta.com에 홈 계정이 있는 고객이 siroe.com의 업무용 계정으로 메시지를 자동 전달하도록 해당 계정을 설정했고, siroe.com에서 Messaging Server를 사용하며, 이 사용자가 휴가 메시지에 자동 회신하도록 계정을 설정한 경우 Messaging Server에서 휴가 메시지를 보내는 데 문제가 생깁니다.

이러한 문제는 sesta.com 메일 서버가 봉투 주소를 user@sesta.com에서 user@siroe.com으로 변경하지만 헤더는 바꾸지 않아 user@sesta.com으로 남아 있기 때문에 발생합니다. MTA는 메시지를 수신하면 헤더 주소만 확인하고 이 주소를 LDAP

사용자 디렉토리의 주소와 일치시키려고 합니다. 자동 회신을 설정한 누군가와 일치할 경우 휴가 메시지가 보내집니다. user@sesta.com과 일치하는 LDAP 주소가 없기 때문에 휴가 메시지가 보내지지 않습니다. 문제는 실제 주소가 헤더가 아니라 봉투에 있다는 것입니다.

자동 전달을 수행하는 원격 시스템에서 알고 있는 수신자의 주소가 로컬 시스템에서 그 사용자의 주소로 알려져 있지 않으므로, 필요할 때 휴가 회신이 보내질 수 있도록 수신자가 로컬 시스템에 그러한 주소를 알리는 방법이 필요합니다.

시브(Sieve) vacation 작업에 대한 :addresses 인수는 이 기능을 제공합니다. 이 인수는 그러한 점검을 수행할 수 있도록 수신자에 해당하는 주소의 목록을 받습니다. MTA 옵션 LDAP_AUTOREPLY_ADDRESSES에서 정의한 속성은 사용자의 LDAP 항목에서 그러한 주소를 지정할 수 있도록 해줍니다.

Sun Mail Server가 아닌 서버로부터 자동 전달된 메시지에 대해 자동 회신 기능을 제공하려면, 해당 사용자나 관리자는 LDAP_AUTOREPLY_ADDRESSES에서 정의하는 속성으로 해당 메시지가 전달될 수 있는 전자 메일 주소를 설정해야 합니다.

◆ ◆ ◆ 18장

18 메일 필터링 및 액세스 제어

이 장에서는 소스(보낸 사람, IP 주소 등) 또는 헤더 문자열을 기반으로 메일을 필터링하는 방법에 대해 설명합니다. 두 개의 메일 필터링 기법, 즉 매핑 테이블을 사용한 MTA에 대한 액세스 제어와 시브(Sieve) 서버측 규칙(SSR)이 사용됩니다.

매핑 테이블을 사용하여 MTA에 대한 액세스를 제한하면 From: 및 To: 주소, IP 주소, 포트 번호, 소스 또는 대상 채널 등에 기초하여 메시지를 필터링할 수 있습니다. 매핑 테이블은 SMTP 중계를 활성화 또는 비활성화할 수 있게 합니다. 시브(Sieve)는 헤더에서 발견된 문자열에 기초하여 메시지를 필터링할 수 있게 하는 메일 필터링 스크립트이며메시지 본문에서는 작동하지 않습니다.

봉투 수준 제어가 필요한 경우에는 매핑 테이블을 사용하여 메일을 필터링하고 헤더 기반 제어가 필요한 경우에는 시브(Sieve) 서버측 규칙을 사용합니다.

이 장은 다음 두 부분으로 구성됩니다.

511 페이지 “18.1 1부. 매핑 테이블”. 관리자가 특정 매핑 테이블을 구성하여 MTA 서비스에 대한 액세스를 제어할 수 있게 합니다. 관리자는 Messaging Server를 통해 메일을 주고 받을 수 있는 사람과 그렇지 않은 사람을 제어할 수 있습니다.

536 페이지 “18.9 2부. 메일함 필터”. 사용자와 관리자가 메시지 헤더에서 찾은 문자열을 기초로 메시지를 필터링하고, 필터링된 이런 메시지에 수행할 작업을 지정할 수 있습니다. 시브(Sieve) 필터 언어를 사용하여 채널, MTA 또는 사용자 수준에서 필터링할 수 있습니다.

18.1 1부. 매핑 테이블

1부는 다음 내용으로 구성되어 있습니다.

- 526 페이지 “18.5 액세스 제어 매핑 테스트”
- 526 페이지 “18.6 SMTP 릴레이 추가”
- 529 페이지 “18.7 SMTP 릴레이 차단 구성”
- 534 페이지 “18.8 많은 수의 액세스 항목 처리”

18.2 매핑 테이블을 사용한 액세스 제어

특정 매핑 테이블을 구성하여 메일 서비스에 대한 액세스를 제어할 수 있습니다. 이러한 매핑 테이블을 사용하면 메일을 전송 및/또는 수신할 수 있거나 그렇게 할 수 없는 사람을 제어할 수 있습니다. 표 18–1에는 이 절에 설명된 매핑 테이블에 나열되어 있습니다. FROM_ACCESS, MAIL_ACCESS 및 ORIG_MAIL_ACCESS 매핑에 제공되는 응용 프로그램 정보 문자열에는 HELO/EHLO SMTP 명령에서 요구된 시스템 이름도 포함됩니다. 이 이름은 문자열 끝에 표시되며 슬래시(/)로 나머지 문자열(일반적으로 "SMTP*")과 구분합니다. 요구된 시스템 이름은 일부 웜 및 바이러스를 차단하는 데 유용하게 사용될 수 있습니다.

18.2.1 액세스 제어 매핑 테이블 - 작업

모든 매핑 테이블과 마찬가지로 액세스 제어 매핑 테이블은 동일한 일반 형식을 가집니다(213 페이지 “10.3 매핑 파일” 참조). 즉, 매핑 테이블 이름이 맨 앞에 오고 그 뒤에 공백이 있으며 맨 뒤에 하나 이상의 매핑 항목이 오는 형식입니다. 매핑 항목은 왼쪽에 있는 **검색 패턴**과 오른쪽에 있는 **템플리트**로 구성됩니다. 검색 패턴은 특정 메시지를 필터링하며 템플리트는 메시지에 대해 수행할 작업을 지정합니다. 예를 들면 다음과 같습니다.

```
SEND_ACCESS

 *|Elvis1@sesta.com|*|*      $Y
 *|Nelson7@sesta.com|*|*     $Y
 *|AkiraK@sesta.com|*|*      $Y
 *|*@sesta.com|*|*           $NMail$ Blocked
```

다음 예는 sesta.com 도메인에서 Elvis1, Nelson, AkiraK 의 전자 메일을 제외한 모든 전자 메일을 차단합니다.

액세스 제어 매핑 항목의 검색 패턴은 세로 막대(|)로 구분된 여러 검색 기준으로 구성됩니다. 검색 기준의 순서는 액세스 매핑 테이블에 따라 다르며 이후의 절에서 설명합니다. 예를 들어 SEND_ACCESS 매핑 테이블의 검색 형식은 다음과 같습니다.

src-channel|*from-address*|*dst-channel*|*to-address*

여기서 *src-channel*은 메시지가 대기 중인 채널, *from-address*는 메시지를 보낸 사람의 주소, *dst-channel*은 메시지가 대기될 채널, *to-address*는 메시지 주소가 지정된 주소입니다. 이 네 필드에 별표를 사용하면 해당 필드는 모든 채널 또는 주소와 일치하게 됩니다.

주 – mappings 파일을 수정할 때마다 구성을 다시 컴파일해야 합니다(209 페이지 “10.1 MTA 구성 컴파일” 참조).

표 18–1 액세스 제어 매핑 테이블

매핑 테이블	설명
SEND_ACCESS(516 페이지 “18.3.1 SEND_ACCESS 및 ORIG_SEND_ACCESS 테이블” 참조)	봉투의 From 주소와 봉투의 To 주소, 소스 및 대상 채널을 기준으로 받는 연결을 차단하는 데 사용됩니다. To 주소는 다시 쓰기, 별칭 확장 등이 수행된 뒤 검사됩니다.
ORIG_SEND_ACCESS(516 페이지 “18.3.1 SEND_ACCESS 및 ORIG_SEND_ACCESS 테이블” 참조)	봉투의 From 주소와 봉투의 To 주소, 소스 및 대상 채널을 기준으로 받는 연결을 차단하는 데 사용됩니다. To 주소는 다시 쓴 다음, 별칭 확장 전에 검사됩니다.
MAIL_ACCESS(518 페이지 “18.3.2 MAIL_ACCESS 및 ORIG_MAIL_ACCESS 매핑 테이블” 참조)	SEND_ACCESS 및 PORT_ACCESS 테이블에서 발견한 결합된 정보(PORT_ACCESS에서 발견한 IP 주소 및 포트 번호 정보와 결합된 SEND_ACCESS에서 발견된 채널 및 주소 정보)에 기초하여 받는 연결을 차단하는 데 사용됩니다.
ORIG_MAIL_ACCESS(518 페이지 “18.3.2 MAIL_ACCESS 및 ORIG_MAIL_ACCESS 매핑 테이블” 참조)	ORIG_SEND_ACCESS 및 PORT_ACCESS 테이블에서 발견한 결합된 정보(PORT_ACCESS에서 발견한 IP 주소 및 포트 번호 정보와 결합된 ORIG_SEND_ACCESS에서 발견된 채널 및 주소 정보)에 기초하여 받는 연결을 차단하는 데 사용됩니다.
FROM_ACCESS(519 페이지 “18.3.3 FROM_ACCESS 매핑 테이블” 참조)	봉투의 From 주소를 기준으로 메일을 필터링하는 데 사용됩니다. To 주소가 부적절한 경우 이 테이블을 사용합니다.
PORT_ACCESS(521 페이지 “18.3.4 PORT_ACCESS 매핑 테이블” 참조)	IP 번호를 기준으로 받는 연결을 차단하는 데 사용됩니다.
IP_ACCESS	소스 채널, 원격 서버의 IP 주소 수, 현재 시도 중인 IP 주소의 색인을 기준으로 받는 연결을 차단하는 데 사용됩니다. 523 페이지 “18.3.5 IP_ACCESS 매핑 테이블”을 참조하십시오.

SEND_ACCESS 및 ORIG_SEND_ACCESS에 사용 가능한 주소 및 채널 정보와 IP 주소와 포트 번호 정보를 포함하여 PORT_ACCESS 매핑 테이블을 통해 사용 가능한 모든 정보를 사용할 수 있는 경우 MAIL_ACCESS 및 ORIG_MAIL_ACCESS 매핑이 가장 일반적입니다.

18.3 액세스 제어 매핑 테이블 플래그

이 절은 다음과 같은 하위 절로 구성되어 있습니다.

- 523 페이지 “18.3.5 IP_ACCESS 매핑 테이블”
- 524 페이지 “18.3.6 MTA에 대해 지정된 IP 액세스 연결 제한”

표 18–2에는 SEND_ACCESS, ORIG_SEND_ACCESS, MAIL_ACCESS, ORIG_MAIL_ACCESS 및 FROM_ACCESS 매핑 테이블에 관련된 액세스 매핑 플래그가 표시되어 있습니다. PORT_ACCESS 매핑 테이블은 이와 약간 다른 플래그 집합을 지원합니다(표 18–3 참조).

인수가 있는 플래그에서는 인수를 표에 나온 읽기 순서에 따라 정렬해야 합니다. 예를 들면 다음과 같습니다.

```
ORIG_SEND_ACCESS

  tcp_local|*|tcp_local|*     $N$D30|Relaying$ not$ allowed
```

이 경우에 올바른 순서는 지연 기간 다음에 거부 문자열이 오는 것입니다. 플래그 자체는 임의의 순서가 될 수 있습니다. 따라서 다음 항목의 결과는 동일합니다.

```
30|Relaying$ not$ allowed$D$N
$N30|Relaying$ not$ allowed$D
30|$N$DRelaying$ not$ allowed
```

표 18–2 액세스 매핑 플래그

플래그	설명
$A	SASL이 사용된 경우 설정합니다. 223 페이지 “특수 플래그 검사”를 참조하십시오.
$B	메시지를 bitbucket으로 리디렉션합니다.
$D	전달 지연 확인이 요청된 경우(FROM_ACCESS에서는 사용할 수 없음) 설정합니다. 223 페이지 “특수 플래그 검사”를 참조하십시오.
$E	EHLO 명령이 실행/허용되어 ESMTP가 사용된 경우에 설정합니다(FROM_ACCESS에서는 사용할 수 없음). 223 페이지 “특수 플래그 검사”를 참조하십시오.
$F	전달 실패 확인이 요청된 경우(FROM_ACCESS에서는 사용할 수 없음) 설정합니다. 223 페이지 “특수 플래그 검사”를 참조하십시오.
$H	메시지를 .HELD 파일로 보관합니다.
$L	LMTP를 사용한 경우에 설정합니다(FROM_ACCESS에서는 사용할 수 없음). 223 페이지 “특수 플래그 검사”를 참조하십시오.
$S	전달 성공 확인이 요청된 경우(FROM_ACCESS에서는 사용할 수 없음) 설정합니다. 223 페이지 “특수 플래그 검사”를 참조하십시오.
$T	TLS가 사용된 경우 설정합니다. 223 페이지 “특수 플래그 검사”를 참조하십시오.

표 18–2 액세스 매핑 플래그 (계속)

플래그	설명
$U	ORIG_SEND_ACCESS, SEND_ACCESS, ORIG_MAIL_ACCESS 및 MAIL_ACCESS에서 사용될 경우 매핑의 시작 부분에서 단일 정수 인수를 가져오며 이에 따라 MM_DEBUG 값을 설정합니다. 또한 가능한 경우 채널 수준 디버깅도 사용 가능하게 합니다. 결과적으로 소스 IP 주소, 원래 주소, 수신자 주소 등의 항목에 기초하여 디버깅을 사용 가능하게 합니다.
$Y	액세스를 허용합니다.
$V	모든 수신자에 대해 삭제를 수행합니다.
$Z	모든 수신자에 대해 jettison을 수행합니다.
$!	FROM_ACCESS에서만 사용 가능합니다. 이 메시지에 대한 휴가 메시지 보내기를 비활성화합니다. 즉 novacation 플래그를 설정합니다. 이는 시스템/채널 시브(Sieve)에서 novacation을 명시적으로 설정하는 것과 동일한 결과를 얻을 수 있습니다. 그러면 해당 메시지에 적용되었을 후속 vacation 작업을 대체(적용 방지)합니다.
인수가 있는 플래그, 인수 읽기 순서에 따라 + (이 목록을 알파벳 순서로 정렬하지 마십시오!)	
$U*integer*	매핑의 시작 부분에서 단일 정수 인수를 가져오며 이에 따라 MM_DEBUG를 설정합니다. 또한 가능한 경우 채널 수준 디버깅도 사용 가능하게 합니다. 결과적으로 소스 IP 주소, 원래 주소, 수신자 주소 등의 항목에 기초하여 디버깅을 사용 가능하게 할 수 있습니다.
$J*address*	원본 봉투의 From: 주소를 지정된 *address*로 대체합니다.
$K*address*	* ++ 원래의 Sender: 주소를 지정된 *address*로 대체합니다.
$I*user*|*identifier*	지정된 사용자의 그룹 아이디를 확인합니다.
$<*string*	+++ 검사가 일치하면 *string*을 syslog(UNIX, user.notice 기능 및 심각도) 또는 이벤트 로그(NT)로 보냅니다.+++
$>*string*	+++ 액세스가 거부되면 *string*을 syslog(UNIX, user.notice 기능 및 심각도) 또는 이벤트 로그(NT)로 보냅니다.
$D*delay*	*delay* 시간 간격(1/100초)에 대한 지연 응답입니다. 양수 값을 사용하면 트랜잭션의 각 명령에 지연이 적용되며, 음수 값을 사용하면 주소 전달(FROM_ACCESS 테이블에 대한 SMTP MAIL FROM: 명령, 다른 테이블에 대한 SMTP RCPT TO: 명령)에만 지연이 적용됩니다.
$T*tag*	*tag* 접두어가 사용됩니다.
$A*header*	메시지에 헤더 행 *header*를 추가합니다.
$G*conversion_tag*	ORIG_SEND_ACCESS, SEND_ACCESS, ORIG_MAIL_ACCESS 및 MAIL_ACCESS에서 사용될 경우 매핑 결과에서 값을 읽고 이를 현재 수신자에게 적용될 변환 태그 집합으로 처리합니다. FROM_ACCESS와 함께 사용될 경우 변환 태그가 모든 수신자에게 적용됩니다. $G의 위치는 매핑에서 읽은 인수 시퀀스에서 $A(헤더 주소) 다음입니다. 409 페이지 “메일 변환 태그”를 참조하십시오.

표 18–2 액세스 매핑 플래그 *(계속)*

플래그	설명
$S*x,y,z*	* 세로 막대(|)로 구분된 추가 인수를 매핑 결과에서 읽습니다. 이 인수는 쉼표로 구분된 하나에서 세 개의 정수 값으로 구성됩니다. 첫 번째 값은 트랜잭션에 대한 최소 `blocklimit`를 새로 설정하고, 두 번째 값은 최소 `ecipientlimit`를 새로 설정하며, 세 번째 값은 최소 `recipientcutoff`를 새로 설정합니다. 이 인수는 모든 캡처 인수를 읽은 후에 매핑 결과에서 읽습니다. 382 페이지 “12.9.2 절대 메시지 크기 제한 지정”을 참조하십시오.
$X*error-code*	메시지 거부 시 지정된 *error-code* 확장 SMTP 오류 코드를 발행합니다.
$,*spamadjust_arg*	액세스 매핑 테이블에서 시브(Sieve) `spamadjust` 작업을 수행할 수 있습니다. 인수는 spamadjust 인수와 동일한 형식을 취합니다. 이러한 매핑의 일부는 수신자별 기준으로 적용된다는 점에 유의하십시오. 수행된 spamadjust 작업은 모든 수신자에 적용됩니다.
$N*string*	선택적 오류 텍스트 *string*을 사용하여 액세스를 거부합니다.
$F*string*	*$N string*에 대한 동의어, 즉 선택적 오류 텍스트 *string*을 사용하여 액세스를 거부합니다.

* `FROM_ACCESS` 테이블에만 사용할 수 있습니다.

+ 인수가 있는 여러 개의 플래그를 사용하려면 인수를 세로 막대 문자 |로 구분하고 이 테이블에 나열된 순서대로 인수를 배치합니다.

++ `$K` 플래그가 `FROM_ACCESS` 매핑 테이블에 적용되려면 소스 채널에 `authrewrite` 키워드가 포함되어야 합니다.

+++ 문제가 있는 보낸 사람을 처리할 때는 서비스 거부 공격을 방지하기 위해 `$D` 플래그를 사용하는 것이 좋습니다. 특히 모든 `$>` 항목 또는 액세스를 거부하는 `$<` 항목에는 `$D`를 사용하는 것이 좋습니다.

18.3.1 SEND_ACCESS 및 ORIG_SEND_ACCESS 테이블

`SEND_ACCESS` 및 `ORIG_SEND_ACCESS` 매핑 테이블을 사용하여 메일을 보내거나, 받거나, 둘 모두를 할 수 있는 사람과 할 수 없는 사람을 제어할 수 있습니다. 액세스 검사는 메시지 봉투의 `From:` 주소와 봉투의 `To:` 주소에서 사용 가능하며 메시지를 전송한 채널과 대상 채널을 알 수 있습니다.

`SEND_ACCESS` 또는 `ORIG_SEND_ACCESS` 매핑 테이블이 있으면 MTA를 통과하여 전달되는 모든 메시지의 각 수신자에 대해 MTA는 다음 형식의 문자열로 테이블을 스캔합니다(세로 막대 문자 | 사용).

src-channel|*from-address*|*dst-channel*|*to-address*

*src-channel*은 메시지가 대기 중인 채널, *from-address*는 메시지를 보낸 사람의 주소, *dst-channel*은 메시지가 대기될 채널, *to-address*는 메시지 주소가 지정된 주소입니다. 이 네 필드에 별표를 사용하면 해당 필드는 모든 채널 또는 주소와 일치하게 됩니다.

여기서 주소는 봉투 주소, 즉 봉투의 `From:` 주소와 봉투의 `To:` 주소입니다. `SEND_ACCESS`의 경우 봉투의 `To:` 주소가 다시 쓰기, 별칭 확장 등이 수행된 후 검사되고, `ORIG_SEND_ACCESS`의 경우 원래 지정된 봉투의 `To:` 주소를 다시 쓴 다음, 그리고 별칭 확장 전에 검사됩니다.

검색 문자열이 패턴(즉, 테이블 항목의 왼쪽)과 일치하면 매핑의 결과 출력이 검사됩니다. 출력에 플래그 $Y 또는 $y가 포함된 경우 해당 `To:` 주소에 대한 대기가 허용됩니다. 출력에 플래그 $N, $n, $F 또는 $f가 포함되어 있으면 해당 주소에 대한 대기가 거부됩니다. 거부된 경우 선택적 거부 텍스트가 매핑 출력에 표시될 수 있습니다. 이 문자열은 MTA가 표시하는 거부 오류에 포함됩니다. 문자열이 출력되지 않으면($N, $n, $F 또는 $f 플래그 제외) 기본 거부 텍스트가 사용됩니다. 추가 플래그에 대한 설명은 513 페이지 “18.3 액세스 제어 매핑 테이블 플래그”를 참조하십시오.

MTA 옵션 `ACCESS_ORCPT`를 1로 설정하면 원래 수신자(ORCPT) 주소를 포함하는 `SEND_ACCESS`, `ORIG_SEND_ACCESS`, `MAIL_ACCESS` 및 `ORIG_MAIL_ACCESS` 매핑 테이블로 전달되는 검사 값에 수직 막대로 구분된 필드가 또 하나 추가됩니다. 메시지에 ORCPT 주소가 없으면 수정되지 않은 원래 RCPT TO: 주소가 대신 사용됩니다. 기본값은 0이고 검사 값은 마지막에 있습니다.

src-channel|*from-address*|*dst-channel*|*to-address*|*ORCPT_address*

다음 예에서는 mail, Pine 등의 UNIX 사용자 에이전트가 보낸 메일(로컬, `l`, 채널 및 메시지에서 인터넷으로 전송)이 일종의 TCP/IP 채널로 나갑니다. 여기서는 포스트마스터를 제외한 이러한 로컬 사용자가 인터넷으로 메일을 보낼 수 없지만 인터넷에서 메일을 받을 수는 있다고 가정합니다. 이 경우 아래 예에 표시된 `SEND_ACCESS` 매핑 테이블을 사용하는 것이 이러한 제한을 적용하는 한 가지 방법이 됩니다. 매핑 테이블에서 로컬 호스트 이름을 `sesta.com`으로 가정합니다. 채널 이름 "`tcp_*`"에서 와일드카드가 사용되어 가능한 모든 TCP/IP 채널 이름(예: `tcp_local`)과 일치합니다.

예 18–1 SEND_ACCESS 매핑 테이블

```
SEND_ACCESS

   *|postmaster@sesta.com|*|*     $Y
   *|*|*|postmaster@sesta.com     $Y
   l|*@sesta.com|tcp_*|*          $NInternet$ postings$ are$ not$ permitted
```

거부 메시지에 공백을 입력하려면 달러 기호를 사용합니다. 달러 기호가 없으면 거부가 일찍 완료되어 “Internet postings are not permitted” 대신 “Internet”만 표시됩니다. 이 예에서는 PC 기반 메일 시스템이나 POP 또는 IMAP 클라이언트등 “로컬” 게시의 소스에 대한 다른 가능성은 무시합니다.

주 – MTA 거부 오류 텍스트를 메시지를 보내려는 사용자에게 실제로 표시할 것인지 여부는 메시지를 보내려는 클라이언트가 결정합니다. SEND_ACCESS를 사용하여 받는 SMTP 메시지를 거부하는 경우 MTA는 선택적 거부 텍스트를 비롯하여 SMTP 거부 코드만 발행합니다. 즉, 이 정보를 사용하여 원래 보낸 사람에게 보낼 바운스 메시지를 구성할 것인지는 SMTP 클라이언트가 결정합니다.

18.3.2 MAIL_ACCESS 및 ORIG_MAIL_ACCESS 매핑 테이블

MAIL_ACCESS 매핑 테이블은 SEND_ACCESS와 PORT_ACCESS 매핑 테이블의 수퍼 세트입니다. 여기에서는 SEND_ACCESS의 채널과 주소 정보를 PORT_ACCESS의 IP 주소 및 포트 번호 정보와 조합합니다. 마찬가지로, ORIG_MAIL_ACCESS 매핑 테이블은 ORIG_SEND_ACCESS와 PORT_ACCESS 매핑 테이블의 수퍼 세트입니다. MAIL_ACCESS의 검사 문자열 형식은 다음과 같습니다.

port-access-probe-info|*app-info*|*submit-type*|*send_access-probe-info*

마찬가지로 ORIG_MAIL_ACCESS의 검사 문자열 형식은 다음과 같습니다.

port-access-probe-info|*app-info*|*submit-type*|*orig_send_access-probe-info*

여기서 받는 SMTP 메시지의 경우 *port-access-probe-info*는 보통 PORT_ACCESS 매핑 테이블 검사에 포함된 모든 정보로 구성되는 반면, 그 외의 경우에는 비어 있는 상태가 됩니다. *app-info*에는 HELO/EHLO SMTP 명령에서 요구한 시스템 이름이 포함됩니다. 이 이름은 문자열 끝에 표시되며 슬래시(/)로 나머지 문자열(일반적으로 "SMTP*")과 구분합니다. 요구된 시스템 이름은 일부 웜 및 바이러스를 차단하는 데 유용하게 사용될 수 있습니다. *submit-type*은 Messaging Server로 메시지가 전송된 방법에 따라 MAIL, SEND, SAML, SOML 중 하나가 될 수 있습니다. 일반적으로 그 값은 MAIL이며 이는 메시지로 전송된다는 의미입니다. 즉, 브로드캐스트 요청(또는 조합된 브로드캐스트/메시지 요청)이 SMTP 서버로 전송된 경우 SEND, SAML 또는 SOML이 발생할 수 있습니다. 또한 MAIL_ACCESS 매핑의 경우 *send-access-probe-info*는 일반적으로 SEND_ACCESS 매핑 테이블 검사에 포함된 모든 정보로 구성됩니다. 마찬가지로 ORIG_MAIL_ACCESS 매핑의 경우에도 *orig-send-access-probe-info*는 일반적으로 ORIG_SEND_ACCESS 매핑 테이블 검사에 포함되는 모든 정보로 구성됩니다.

MTA 옵션 ACCESS_ORCPT를 1로 설정하면 원래 수신자(ORCPT) 주소를 포함하는 SEND_ACCESS, ORIG_SEND_ACCESS, MAIL_ACCESS 및 ORIG_MAIL_ACCESS 매핑 테이블로 전달되는 검사 값에 수직 막대로 구분된 필드가 또 하나 추가됩니다. 메시지에 ORCPT 주소가 없으면 수정되지 않은 원래 RCPT TO: 주소가 대신 사용됩니다. 기본값은 0이고 검사 값은 마지막에 있습니다. 예:

port-access-probe-info|*app-info*|*submit-type*|*send_access-probe-info*|*ORCPT_address*

받는 TCP/IP 연결 정보를 채널 및 주소 정보와 동일한 매핑 테이블에서 사용할 수 있는 경우에는 특정 IP 주소에서 보낸 메일에 표시되도록 허용되는 봉투의 From: 주소를 지정하는 등의 제어를 보다 편리하게 수행할 수 있습니다. 이렇게 하면 전자 메일 위조의 가능성을 줄이거나 사용자가 자신의 POP 및 IMAP 클라이언트의 From: 주소를 적절하게 설정하도록 유도할 수 있습니다. 예를 들어, 봉투의 From: 주소 vip@siroe.com이 IP 주소 1.2.3.1 및 1.2.3.2에서 받는 메시지에만 나타나도록 하고 1.2.0.0 서브넷에 있는 시스템으로부터 받는 메시지의 봉투의 From: 주소는 siroe.com에서 보낸 것으로 하려면 아래 예에 표시된 대로 MAIL_ACCESS 매핑 테이블을 사용할 수 있습니다.

예 18–2 MAIL_ACCESS 매핑 테이블

```
MAIL_ACCESS

! Entries for vip's two systems
!
  TCP|*|25|1.2.3.1|*|SMTP*|MAIL|tcp_*|vip@siroe.com|*|*  $Y
  TCP|*|25|1.2.3.2|*|SMTP*|MAIL|tcp_*|vip@siroe.com|*|*  $Y
!
! Disallow attempts to use vip's From: address from other
! systems
!
  TCP|*|25|*|*|SMTP*|MAIL|tcp_*|vip@siroe.com|*|*  \
      $N500$ Not$ authorized$ to$ use$ this$ From:$ address
!
! Allow sending from within our subnet with siroe.com From:
! addresses
!
  TCP|*|25|1.2.*.*|*|SMTP*|MAIL|tcp_*|*@siroe.com|*|*  $Y
!
! Allow notifications through
!
  TCP|*|25|1.2.*.*|*|SMTP*|MAIL|tcp_*||*|*  $Y
!
! Block sending from within our subnet with non-siroe.com
! addresses
!
  TCP|*|25|1.2.*.*|*|SMTP*|MAIL|tcp_*|*|*|*  \
     $NOnly$ siroe.com$ From:$ addresses$ authorized
```

18.3.3 FROM_ACCESS 매핑 테이블

FROM_ACCESS 매핑 테이블은 메일을 보낼 수 있는 사용자를 제어하거나 인증된 주소를 가진 From: 주소를 무시하는 데 사용할 수 있습니다.

FROM_ACCESS 매핑 테이블에 대한 입력 검사 문자열은 MAIL_ACCESS 매핑 테이블에서 대상 채널과 주소를 제외하고 인증된 보낸 사람 정보(사용 가능한 경우)를 추가한 것과

같습니다. 따라서 FROM_ACCESS 매핑 테이블이 있는 경우 Messaging Server는 시도되는 각 메시지 전송에 대해 다음 형식의 문자열을 가진 테이블을 검색합니다(세로 막대 문자 | 사용 주의).

port-access-probe-info|*app-info*|*submit-type*|*src-channel*|*from-address*|*auth-from*

여기서 받는 SMTP 메시지의 경우 *port-access-probe-info*는 보통 PORT_ACCESS 매핑 테이블 검사에 포함된 모든 정보로 구성되는 반면, 그 외의 경우에는 비어 있는 상태가 됩니다. *app-info*에는 HELO/EHLO SMTP 명령에서 요구한 시스템 이름이 포함됩니다. 이 이름은 문자열 끝에 표시되며 슬래시(/)로 나머지 문자열(일반적으로 "SMTP*")과 구분합니다. 요구된 시스템 이름은 일부 웜 및 바이러스를 차단하는 데 유용하게 사용될 수 있습니다. *submit-type*은 MTA로 메시지가 전송된 방법에 따라 MAIL, SEND, SAML, SOML 중 하나가 될 수 있습니다. 일반적으로 그 값은 MAIL이며 이는 메시지로 전송된다는 의미입니다. 즉, 브로드캐스트 요청(또는 조합된 브로드캐스트/메시지 요청)이 SMTP 서버로 전송된 경우 SEND, SAML 또는 SOML이 발생할 수 있습니다. *src-channel*은 메시지를 보낸(메시지를 대기열에 넣는) 채널, *from-address*는 메시지를 최초로 보낸 사람의 주소이며 *auth-from*은 인증된 보낸 사람 주소(이 정보가 사용 가능한 경우)이고 인증된 정보를 사용할 수 없는 경우에는 비어 있습니다.

검사 문자열이 패턴(즉, 테이블 항목의 왼쪽)과 일치하면 매핑의 결과 출력이 검사됩니다. 출력에 플래그 $Y 또는 $y가 포함된 경우 해당 To: 주소에 대한 대기가 허용됩니다. 출력에 플래그 $N, $n, $F 또는 $f가 포함되어 있으면 해당 주소에 대한 대기가 거부됩니다. 거부된 경우 선택적 거부 텍스트가 매핑 출력에 표시될 수 있습니다. 이 문자열은 Messaging Server가 표시하는 거부 오류에 포함될 수 있습니다. 문자열이 출력되지 않으면($N, $n, $F 또는 $f 플래그 제외) 기본 거부 텍스트가 사용됩니다. 추가 플래그에 대한 설명은 513 페이지 “18.3 액세스 제어 매핑 테이블 플래그”를 참조하십시오.

FROM_ACCESS는 메시지 발송자를 기준으로 전송 가능한 메시지를 허용할지 여부를 결정하는 것 외에도 봉투의 From: 주소를 $J 플래그를 통해 변경하거나 authrewrite 채널 키워드(받은 메시지의 Sender: 헤더 주소 추가)의 결과를 $K 플래그를 통해 수정하는 데 사용할 수도 있습니다. 예를 들어, 이 매핑 테이블을 사용하여 다음과 같이 원래의 봉투의 From: 주소를 인증된 주소로 바꿀 수 있습니다.

예 18–3 FROM_ACCESS 매핑 테이블

```
FROM_ACCESS

  *|SMTP*|*|tcp_auth|*|        $Y
  *|SMTP*|*|tcp_auth|*|*       $Y$J$4
```

FROM_ACCESS 매핑 테이블을 사용하여 일부 소스 채널의 0이 아닌 값에 대해 authrewrite를 설정한 결과를 수정할 때 인증된 주소가 글자 그대로 사용되는 경우에는 FROM_ACCESS를 사용하지 않아도 됩니다.

예를 들어, tcp_local 채널에 authrewrite 2를 설정한 경우에는 authrewrite만으로도 이 결과를 얻을 수 있기 때문에(인증된 주소를 그대로 추가) FROM_ACCESS 매핑 테이블이 필요하지 않습니다.

```
FROM_ACCESS

   *|SMTP*|*|tcp_auth|*|      $Y
   *|SMTP*|*|tcp_auth|*|*     $Y$K$4
```

하지만 FROM_ACCESS의 실제 용도는 아래 예에 표시된 대로 보다 복잡하고 세밀한 변경을 허용하는 것입니다. Sender: 헤더 행(SMTP AUTH 인증 전송자 주소 표시)을 받는 메시지에 추가하려는 경우에는 authrewrite 키워드만 사용해도 됩니다. 하지만 SMTP AUTH 인증 전송자 주소가 봉투의 From: 주소와 다른 경우에만 Sender: 헤더 행 등을 받는 메시지에 추가하는(즉, 주소가 일치하는 경우에는 Sender: 헤더 행) 것으로 가정하고, 또한 봉투의 From: 에 선택적 하위 주소 정보가 포함되어 있다는 이유만으로 SMTP AUTH와 봉투의 From:을 서로 다른 것으로 간주하지 않는 것으로 가정합니다.

```
FROM_ACCESS

! If no authenticated address is available, do nothing
  *|SMTP*|*|tcp_auth|*| $Y
! If authenticated address matches envelope From:, do nothing
  *|SMTP*|*|tcp_auth|*|$3* $Y
! If authenticated address matches envelope From: sans
! subaddress, do nothing
  *|SMTP*|*|tcp_auth|*+*@*|$3*@$5* $Y
! Fall though to...
! ...authenticated address present, but didn?t match, so force
! Sender: header
  *|SMTP*|*|tcp_auth|*|* $Y$K$4
```

FROM_ACCESS의 $(메타 문자는 주소를 결과 문자열로부터 읽고 현재 대체 포스트마스터 주소를 대체하는 데 사용하도록 지정합니다. $)는 동일한 효과를 제공하되 대체 포스트마스터 주소가 매핑을 호출하기 전에 설정되면 안되는 제약 조건이 있습니다. 따라서 특정 포스트마스터 주소가 로컬이 아닌 도메인의 주소와 함께 사용될 수 있습니다. 도메인 포스트마스터 주소는 물론 로컬에서 정의된 도메인에서만 작동합니다. 대체 주소는 $N/$F 실패 결과를 읽기 전에 FROM_ACCESS 결과에서 읽은 (현재) 마지막 문자열입니다.

18.3.4 PORT_ACCESS 매핑 테이블

디스패처는 IP 주소와 포트 번호를 기반으로 선택적으로 받는 연결을 수락하거나 거부할 수 있습니다. 디스패처는 시작 시에 PORT_ACCESS라는 매핑 테이블을 찾습니다. 이 테이블이 있으면 디스패처는 연결 정보를 다음 형식으로 구성합니다.

TCP|*server-address*|*server-port*|*client-address*|*client-port*

디스패처는 모든 PORT_ACCESS 매핑 항목에 대응시키려 시도합니다. 매핑 결과에 $N 또는 $F가 포함되어 있으면 연결이 즉시 닫힙니다. 매핑의 다른 결과는 연결이 수락되는 것을 나타냅니다. 거부 메시지 다음에 선택적으로 $N 또는 $F가 올 수 있습니다. $N 또는 $F가 오는 경우 메시지는 닫히기 직전에 해당 연결로 다시 보내질 수 있습니다. CRLF 종결자는 연결로 다시 보내지기 전에 문자열에 추가됩니다.

주 – MMP는 PORT_ACCESS 매핑 테이블을 사용하지 않습니다. 특정 IP 주소의 SMTP 연결을 거부하기를 원하고 MMP를 사용하는 경우 TCPAccess 옵션을 사용해야 합니다. 165 페이지 "7.5.1 MMP를 사용하여 메일 액세스 구성"을 참조하십시오. 매핑 테이블을 사용하여 SMTP 연결을 제어하려면 INTERNAL_IP 매핑 테이블을 사용합니다(528 페이지 "18.6.1 외부 사이트에 대한 SMTP 릴레이 허용" 참조).

$< 플래그 다음에 선택적 문자열이 있으면 매핑 검사가 일치하는 경우 Messaging Server는 문자열을 syslog(UNIX) 또는 이벤트 로그(NT)로 보냅니다. $> 플래그 다음에 선택적 문자열이 오면 액세스가 거부된 경우 Messaging Server는 syslog(UNIX) 또는 이벤트 로그(NT)로 보냅니다. LOG_CONNECTION MTA 옵션이 비트 1로 설정되고 $N 플래그가 설정되어 연결이 거부된 경우 $T 플래그를 지정하면 "T" 항목이 연결 로그에 기록됩니다. LOG_CONNECTION MTA 옵션의 비트 4가 설정된 경우에는 사이트 제공 텍스트가 PORT_ACCESS 항목에 제공되어 "C" 연결 로그 항목에 포함될 수 있습니다. 이러한 텍스트를 지정하려면 항목의 오른쪽에 두 개의 세로 막대 문자를 넣고 그 뒤에 원하는 텍스트를 입력합니다. 표 18–3에는 사용 가능한 플래그가 나열되어 있습니다.

이전 버전의 Messaging Server(6.2 이전)에서 PORT_ACCESS 매핑은 디스패처와 달리 LOG_CONNECTION MTA 옵션의 비트 4(값 16)가 설정되었거나, SMTP auth가 활성화되었거나, 둘 다인 경우에만 SMTP 서버에서 다시 평가되었습니다. 또, AUTH, EHLO 또는 HELO 명령이 실행된 경우에만 평가가 실행되었습니다. 지금은 이런 사항이 변경되었습니다. PORT_ACCESS는 이제 SMTP 서버 스레드가 시작되면 배너가 전송되기 전에 무조건 평가됩니다. PORT_ACCESS는 diff로 다시 평가할 수 있습니다.

표 18–3 PORT_ACCESS 매핑 플래그

플래그	설명
$D	추가 인수를 템플리트 결과에서 필수 SMTP auth rulset 및 영역 그리고 선택적 응용 프로그램 정보 추가 다음에 읽게 합니다. 이 값은 BANNER_PURGE_DELAY 값과 동일한 의미의 정수이어야 합니다. 즉, 배너를 제거하고 보내기 전 지연 시간(1/100초)을 지정합니다. 값이 0이면 지연과 제거가 모두 비활성화됩니다. 모든 PORT_ACCESS 매핑 설정은 BANNER_PURGE_DELAY SMTP 채널 옵션을 대체합니다. 이 스팸 방지 기능 사용에 대한 자세한 내용은 477 페이지 "14.9.1 스팸 방지 기술: SMTP 배너 보내기 지연"을 참조하십시오.
$Y	액세스를 허용합니다.

표 18–3 PORT_ACCESS 매핑 플래그 *(계속)*

플래그	설명
$U	채널 수준 디버깅을 선택적으로 활성화합니다.
인수가 있는 플래그, 인수 읽기 순서에 따라+	
$< 문자열	검사가 일치하는 경우 syslog (UNIX) 또는 이벤트 로그(NT)에 문자열을 보냅니다.
$> 문자열	액세스가 거부되는 경우 syslog (UNIX) 또는 이벤트 로그(NT)에 문자열을 보냅니다.
$N 문자열	선택적 오류 텍스트 문자열을 사용하여 액세스를 거부합니다.
$F 문자열	$N 문자열에 대한 동의어, 즉 선택적 오류 텍스트 문자열을 사용하여 액세스를 거부합니다.
$T 텍스트	LOG_CONNECTION MTA 옵션이 비트 1(값 2)로 설정되고 $N 플래그가 설정되어 연결이 거부된 경우 $T 플래그를 지정하면 "T" 항목이 연결 로그에 기록됩니다. T 로그 항목에는 전체 매핑 결과 문자열($N 및 해당 문자열)이 포함됩니다.

+ 인수가 있는 여러 개의 플래그를 사용하려면 인수를 세로 막대 문자 |로 구분하고 이 테이블에 나열된 순서대로 인수를 배치합니다.

예를 들어, 다음 매핑은 설명하는 텍스트가 없이 추출되어 거부된 특정 호스트를 제외한 단일 네트워크로부터의 SMTP 연결(포트 25, 일반 SMTP 포트)만 수락합니다.

```
PORT_ACCESS

  TCP|*|25|192.123.10.70|*  $N500
  TCP|*|25|192.123.10.*|*   $Y
  TCP|*|25|*|*              $N500$ Bzzzt$ thank$ you$ for$ playing.
```

PORT_ACCESS 매핑 테이블을 변경한 뒤에는 디스패처를 다시 시작해야 디스패처에 변경 내용이 적용됩니다. 컴파일된 MTA 구성을 사용하는 경우에는 먼저 구성을 다시 컴파일하여 변경 내용을 컴파일된 구성에 통합시켜야 합니다.

PORT_ACCESS 매핑 테이블은 특별히 IP 기반 거부를 수행하기 위한 것입니다. 메일 주소 수준 일반 제어의 경우 SEND_ACCESS 또는 MAIL_ACCESS 매핑 테이블이 보다 적합합니다.

18.3.5 IP_ACCESS 매핑 테이블

IP_ACCESS 매핑 테이블을 사용하면 MTA에서 연결하려는 IP 주소에 대한 최종 확인을 수행할 수 있습니다. 그러면 연결 시도가 중단되거나 리디렉션될 수 있습니다. 연결을 허용하지 않아야 할 대상 IP 주소에 대한 보안 문제가 있거나, bogus로 알려진 대상 IP 주소(127.0.0.1 등)에 대한 연결을 피하거나, lastresort 키워드 효과(346 페이지 "12.4.3.7 마지막 Resort 호스트" 참조)와 비슷한 다른 대상 IP 주소로 페일오버하는 등의 특수한 경우에 유용할 수 있습니다.

이 액세스 매핑은 원격 서버에 대한 연결을 열려고 시도하기 전에 SMTP 클라이언트 작업을 수행하는 동안 참조됩니다. 매핑 검사의 형식은 다음과 같습니다.

```
source-channel|address-count|address-current|ip-current|hostname
```

`source-channel`은 메시지의 대기열을 해제 중인 채널입니다. `address-count`는 원격 서버의 총 IP 주소 수입니다. `address-current`는 시도 중인 현재 IP 주소의 색인입니다. `ip-current`는 현재 IP 주소입니다. `hostname`은 원격 서버의 심볼릭 이름입니다. 아래 표는 이 테이블의 플래그를 보여줍니다.

표 18–4 IP_ACCESS 매핑 테이블 플래그

플래그	설명
$N	"잘못된 호스트/도메인 오류"가 발생한 메시지를 즉시 거부합니다.제공되는 텍스트는 거부 사유로 기록되지만 DSN에 포함되지는 않습니다.
$I	연결을 시도하지 않고 현재 IP를 건너뜁니다.
$A	현재 IP 주소를 매핑 결과로 대체합니다.

18.3.6 MTA에 대해 지정된 IP 액세스 연결 제한

특정 주소가 MTA에 연결될 수 있는 빈도를 제한하려면 19 장를 참조하십시오. 특정 IP 주소로 연결을 제한하는 기능은 서비스 거부 공격에 사용되는 과도한 연결을 방지하는 데 유용합니다. 이전에는 Port Access 매핑 테이블에서 `conn_throttle.so` 공유 라이브러리를 사용하여 이 기능을 수행했습니다. `conn_throttle.so`의 기능을 새롭게 향상시킬 계획은 없으며, MeterMaid로 대체하는 것이 더 효과적입니다.

`conn_throttle.so`는 특정 IP 주소가 MTA에 너무 자주 연결하는 것을 제한하기 위해 `PORT_ACCESS` 매핑 테이블에 사용되는 공유 라이브러리입니다. 모든 구성 옵션은 다음과 같이 연결 억제 공유 라이브러리에 대한 매개 변수로 지정됩니다.

`$[`*msg-svr-base*`/lib/conn_throttle.so, throttle,` *IP-address*`,`*max-rate* `]`

*IP-address*는 원격 시스템의 점으로 구분된 십진수 형식의 주소이며, *max-rate*는 이 IP 주소에 대한 최대 분당 연결 비율입니다.

루틴 이름 `throttle_p`를 루틴 축소 버전의 `throttle` 대신 사용할 수 있습니다. `throttle_p`는 향후 기존에 너무 많이 연결했던 연결을 거부하게 됩니다. 최대 비율이 100인데 분당 250번의 연결이 시도된 경우에는 해당 분 내에 처음 100번의 연결 시도 후 원격 사이트가 차단되며 그 다음 1분 동안에도 차단됩니다. 즉, 매 분마다 시도된 전체 연결 수에서 최대 비율을 빼서 전체 연결 수가 최대 비율보다 크면 원격 시스템이 차단됩니다.

지정된 IP 주소가 분당 최대 연결 비율을 초과하지 않으면 공유 라이브러리 호출이 실패합니다.

해당 비율을 초과하면 호출에 성공하지만 아무 것도 반환하지 않습니다. 이 작업은 다음 예와 같이 $C/$E 조합으로 수행됩니다.

```
PORT_ACCESS
  TCP|*|25|*|* \
$C$[msg-svr-base/lib/conn_throttle.so,throttle,$1,10] \
$N421$ Connection$ not$ accepted$ at$ this$ time$E
```

여기서

`$C`는 다음 테이블 항목에서 시작한 매핑 프로세스를 계속하여 이 항목의 출력 문자열을 매핑 프로세스에 대한 새 입력 문자열로 사용합니다.

`$[`*msg-svr-base*`/lib/conn_throttle.so,throttle,$1,10]`은 `throttle`을 라이브러리 루틴, `$1`을 서버 IP 주소, 그리고 10을 분당 연결 임계값으로 사용하는 라이브러리 호출입니다.

`$N421$ Connection$ not$ accepted$ at$ this$ time`은 액세스를 거부하고 "Connection not accepted at this time"이라는 메시지와 함께 421 SMTP 코드(임시 부정 완료)를 반환합니다.

`$E`는 이제 매핑 프로세스를 닫습니다. 이 항목의 출력 문자열을 매핑 프로세스의 최종 결과로 사용합니다.

18.4 액세스 제어가 적용되는 경우

Messaging Server는 가능한 빨리 액세스 제어 매핑을 검사합니다. 정확한 작업 수행 시기는 사용 중인 전자 메일 프로토콜에 따라 다릅니다(검사해야 하는 정보가 사용 가능한 경우).

SMTP 프로토콜의 경우 MAIL FROM: 명령에 대한 응답으로 `FROM_ACCESS` 거부가 수행된 후 보내는 측에서 수신자 정보나 메시지 데이터를 보낼 수 있습니다. 보내는 측에서 메시지 데이터를 보내기 전에 RCPT TO: 명령에 대한 응답으로 `SEND_ACCESS` 또는 `MAIL_ACCESS` 거부가수행됩니다. SMTP 메시지가 거부되면 Messaging Server는 메시지 데이터를 수락하거나 볼 수 없으므로 이러한 거부 수행으로 인한 오버헤드가 최소화됩니다.

여러 개의 액세스 제어 매핑 테이블이 있으면 Messaging Server는 이들 모두를 검사합니다. 즉, `FROM_ACCESS`, `SEND_ACCESS`, `ORIG_SEND_ACCESS`, `MAIL_ACCESS` 및 `ORIG_MAIL_ACCESS` 매핑 테이블에 모두 영향을 받을 수 있습니다.

18.5 액세스 제어 매핑 테스트

imsimta test -rewrite 유틸리티(특히 -from, -source_channel, -sender 및 -destination_channel 옵션과 함께 사용 시)는 액세스 제어 매핑을 테스트할 때 유용합니다. 자세한 내용은 **Sun Java System Messaging Server 6.3 Administration Reference**의 "imsimta test"를 참조하십시오. 아래 예는 샘플 SEND_ACCESS 매핑 테이블 및 그 검사 결과를 보여 줍니다.

```
MAPPING TABLE:

SEND_ACCESS

  tcp_local|friendly@siroe.com|l|User@sesta.com     $Y
  tcp_local|unwelcome@varrius.com|l|User@sesta.com  $NGo$ away!

PROBE:

$ TEST/REWRITE/FROM="friendly@siroe.com" -
_$ /SOURCE=tcp_local/DESTINATION=l User@sesta.com
...
Submitted address list:
  l
    User (SESTA.COM) *NOTIFY FAILURES* *NOTIFY DELAYS* Submitted
notifications list:

$ TEST/REWRITE/FROM="unwelcome@varrius.com" -
_$ /SOURCE=tcp_local/DESTINATION=l User@sesta.com
...
Submitted address list:
Address list error -- 5.7.1 Go away! User@sesta.com

Submitted notifications list:
```

18.6 SMTP 릴레이 추가

기본적으로 Messaging Server는 SMTP 릴레이 시도를 차단하도록 구성되어 있습니다. 즉, 인증되지 않은 외부 소스의 외부 주소로의 메시지 전송 시도를 거부합니다. 외부 시스템은 서버가 있는 호스트가 아닌 모든 시스템을 말합니다. 이 기본 구성은 다른 모든 시스템을 외부 시스템으로 간주하기 때문에 과도하게 SMTP 중계를 차단합니다.

Messaging Server 시스템의 SMTP 서버를 통해 외부 주소로 지정된 메시지를 전송하려고 시도하는 IMAP 및 POP 클라이언트, 그리고 SMTP AUTH(SASL)를 사용하여 인증하지 않는 클라이언트의 전송 시도는 거부됩니다. 따라서 사용자 구성을 수정하여 중계를 항상 수락하는 자체 내부 시스템과 서브넷을 인식하도록 할 수 있습니다.

내부로 인식되는 시스템과 서브넷은 *msg-svr-base*/config/mappings 파일에 포함된 INTERNAL_IP 매핑 테이블을 통해 제어됩니다.

예를 들어, IP 주소가 123.45.67.89인 Messaging Server 시스템에서 기본 INTERNAL_IP 매핑 테이블은 다음과 같이 나타납니다.

```
INTERNAL_IP

   $(123.45.67.89/32)   $Y
   127.0.0.1            $Y
   *   $N
```

여기서 첫 번째 항목은 $(IP-pattern/signicant-prefix-bits) 구문을 사용하여 123.45.67.89의 32비트와 완전히 일치하는 모든 IP 주소를 내부로 인식하도록 지정합니다. 두 번째 항목은 루프백 IP 주소 127.0.0.1을 내부로 인식합니다. 마지막 항목은 다른 모든 IP 주소가 내부로 인식되지 않도록 지정합니다. 모든 항목 앞에는 적어도 하나의 공백이 있어야 합니다.

마지막 $N 항목 앞에 추가 IP 주소 또는 서브넷을 지정하여 항목을 추가할 수 있습니다. 이러한 항목은 왼쪽에 IP 주소나 서브넷($(.../...구문을 사용하여 서브넷 지정)을 지정하고 오른쪽에 $Y를 지정합니다. 또는 기존 $(.../...) 항목을 수정하여 더 일반적인 서브넷을 허용할 수 있습니다.

예를 들어, 동일한 샘플 사이트의 네트워크가 클래스 C 네트워크, 즉 123.45.67.0 서브넷을 모두 소유하는 네트워크인 경우 해당 사이트에서는 주소 일치에 사용되는 비트 수를 변경하여 첫 번째 항목을 수정해야 합니다. 아래 매핑 테이블에서는 32비트를 24비트로 수정합니다. 이렇게 하면 클래스 C 네트워크의 모든 클라이언트에서 SMTP 중계 서버를 통해 메일을 중계할 수 있습니다.

```
INTERNAL_IP

   $(123.45.67.89/24)   $Y
   127.0.0.1   $Y
   *   $N
```

또는 사이트가 123.45.67.80-123.45.67.99 범위 내의 IP 주소만 소유하는 경우 해당 사이트는 다음을 사용할 수 있습니다.

```
INTERNAL_IP

! Match IP addresses in the range 123.45.67.80-123.45.67.95
   $(123.45.67.80/28) $Y
! Match IP addresses in the range 123.45.67.96-123.45.67.99
   $(123.45.67.96/30) $Y
   127.0.0.1 $Y
   * $N
```

imsimta test -match 유틸리티는 IP 주소가 특정 $(.../...) 테스트 조건에 일치하는지 여부를 검사할 때 유용하게 사용할 수 있습니다. imsimta test -mapping 유틸리티는 INTERNAL_IP 매핑 테이블이 다양한 IP 주소 입력에 대해 원하는 결과를 반환하는지 검사할 때 매우 유용합니다.

INTERNAL_IP 매핑 테이블을 수정한 뒤에는 imsimta restart 명령(컴파일된 구성을 실행하고 있지 않은 경우) 또는 imsimta cnbuild 명령 뒤에 imsimta restart smtp 명령(컴파일된 구성을 실행하는 경우)을 실행해야 변경 사항이 적용됩니다.

매핑 테이블과 일반적인 매핑 테이블 형식 및 imsimta 명령줄 유틸리티에 대한 자세한 내용은 Messaging Server Reference Manual을 참조하십시오.

18.6.1 외부 사이트에 대한 SMTP 릴레이 허용

위에서 설명한 것처럼 모든 내부 IP 주소를 INTERNAL_IP 매핑 테이블에 추가해야 합니다. 다른 시스템/사이트에서 SMTP 릴레이를 허용하려는 경우 가장 간단한 방법은 해당 시스템/사이트를 INTERNAL_IP 매핑 테이블에 사용자의 실제 내부 IP 주소와 함께 포함시키는 것입니다.

다른 시스템/사이트를 실제 내부 시스템/사이트로 인식시키지 않으려는 경우(예를 들어 로깅이나 다른 제어 목적을 위해 **실제 내부 시스템**과 **릴레이 권한을 가진 내부가 아닌 시스템**을 구분하려는 경우) 다른 방법으로 시스템을 구성할 수 있습니다.

한 가지 방법은 다른 시스템에서 보내는 메시지를 받는 특별 채널을 설정하는 것입니다. 이렇게 하려면 기존 *tcp_internal*과 유사한 *tcp_friendly* 채널을 공식 호스트 이름 *tcp_friendly-daemon*으로 만들고 다른 시스템 IP 주소가 나열된 INTERNAL_IP 매핑 테이블과 유사한 FRIENDLY_IP 매핑 테이블을 만듭니다. 그런 후 다음과 같은 현재 다시 쓰기 규칙 바로 뒤에

```
! Do mapping lookup for internal IP addresses
[]    $E$R${INTERNAL_IP,$L}$U%[$L]@tcp_intranet-daemon
```

다음과 같이 다시 쓰기 규칙을 새로 추가합니다.

```
! Do mapping lookup for "friendly", non-internal IP addresses
[]     $E$R${FRIENDLY_IP,$L}$U%[$L]@tcp_friendly-daemon
```

또 다른 방법은 위의 ORIG_SEND_ACCESS 매핑 테이블에 다음 형식의 새로운 최종 $N 항목을 추가하고

```
tcp_local|*@siroe.com|tcp_local|*     $Y
```

(여기서 siroe.com은 다른 도메인의 이름) 다음 형식의 ORIG_MAIL_ACCESS 매핑 테이블을 추가하는 것입니다.

```
ORIG_MAIL_ACCESS

   TCP|*|25|$(match-siroe.com-IP-addresses)|*|SMTP*|MAIL|   \
tcp_local|*@siroe.com|tcp_local|*      $Y
   TCP|*|*|*|*|SMTP*|MAIL|tcp_local|*|tcp_local|* $N
```

여기서 $(...) IP 주소 구문은 이전 절에서 설명한 것과 같은 구문입니다. ORIG_SEND_ACCESS 검사는 주소가 정상인 경우 지속되므로 계속 수행할 수 있으며 또한 IP 주소가 siroe.com IP 주소에 해당하는 경우에 한해 보다 엄격한 ORIG_MAIL_ACCESS 검사를 수행할 수 있습니다.

18.7 SMTP 릴레이 차단 구성

액세스 제어 매핑을 사용하여 다른 사용자가 Messaging Server 시스템을 통해 SMTP 메일을 릴레이하지 못하도록 막을 수 있습니다. 예를 들어, 다른 사용자가 메일 시스템을 사용하여 대량 전자 메일을 수백 수천의 인터넷 메일함으로 릴레이하지 못하도록 할 수 있습니다.

기본적으로 Messaging Server는 로컬 POP 및 IMAP 사용자에 의한 릴레이를 포함하여 모든 SMTP 릴레이 작업을 차단합니다.

적합한 로컬 사용자에게 릴레이를 허용하면서 인증되지 않은 릴레이를 차단하려면 Messaging Server에서 이 두 클래스의 사용자를 서로 구분할 수 있도록 구성해야 합니다. 예를 들어, POP나 IMAP를 사용하는 로컬 사용자의 경우 Messaging Server가 SMTP 릴레이 역할을 수행합니다.

SMTP 릴레이를 차단하려면 다음이 가능해야 합니다.

- 내부 메일과 외부 메일 구분
- 531 페이지 “18.7.2 인증된 사용자의 메일 구분”
- 532 페이지 “18.7.3 메일 릴레이 금지”

내부 호스트 및 클라이언트에서 SMTP 릴레이를 사용하려면 “내부” IP 주소나 서브넷을 INTERNAL_IP 매핑 테이블에 추가해야 합니다.

18.7.1 MTA의 내부 메일과 외부 메일 구분 방법

메일 릴레이 작업을 차단하려면 MTA는 먼저 사용자 사이트에서 전송된 내부 메일과 외부 인터넷에서 전송되어 사용자 시스템을 경유하여 다시 인터넷으로 나가는 외부 메일을 구분할 수 있어야 합니다. 내부 메일은 허용하고 외부 메일은 차단하려 합니다. 인바운드 SMTP 채널(일반적으로 tcp_local 채널이며 기본적으로 설정됨)에서 switchchannel 키워드를 사용하여 구분할 수 있습니다.

switchchannel 키워드를 사용하여 SMTP 서버가 들어오는 SMTP 연결에 연관된 실제 IP 주소를 조사합니다. Messaging Server는 이 IP 주소와 다시 쓰기 규칙을 결합하여 도메인

내에서 보낸 SMTP와 도메인 외부로부터의 연결을 구분합니다. 그런 다음 이 정보는 내부 메시지 트래픽과 외부 메시지 트래픽을 분리하는 데 사용될 수 있습니다.

아래 설명된 MTA 구성은 기본적으로 서버가 내부 메시지 트래픽과 외부 메시지 트래픽을 구분할 수 있도록 설정됩니다.

- 구성 파일에서 로컬 채널 바로 앞에 `noswitchchannel` 키워드가 지정된 `defaults` 채널이 옵니다.

  ```
  ! final rewrite rules
  defaults noswitchchannel
  ! Local store
  ims-ms ...
  ```

- 받는 TCP/IP 채널은 `switchchannel`과 `remotehost` 키워드를 지정하며, 예를 들면 다음과 같습니다.

  ```
  tcp_local smtp single_sys mx switchchannel remotehost
  TCP-DAEMON
  ```

- 받는 TCP/IP 채널 정의 뒷부분은 이름은 다르지만 비슷한 채널입니다. 예를 들면 다음과 같습니다.

  ```
  tcp_intranet smtp single_sys mx allowswitchchannel routelocal
  tcp_intranet-daemon
  ```

  `routelocal` 채널 키워드는 채널에 주소를 다시 쓸 때 MTA가 해당 주소의 모든 명시적 라우팅을 이 채널을 통해 "단락"하도록 하여 명시적으로 소스 라우팅된 주소를 통해 내부 SMTP 호스트를 거친 루핑 방식으로 릴레이 시도를 차단합니다.

위 구성 설정으로 도메인 내에서 생성된 SMTP 메일은 `tcp_intranet` 채널을 통해 들어옵니다. 다른 모든 SMTP 메일은 `tcp_local` 채널을 통해 들어옵니다. 이렇게 해당 메일이 들어오는 채널을 기준으로 내부 메일과 외부 메일이 구분됩니다.

이 작업의 작동 방식에 대해 알아보겠습니다. 여기서 핵심은 `switchchannel` 키워드이며,`tcp_local` 채널에 적용됩니다. 서버는 메시지가 SMTP 서버에 들어오면 키워드를 통해 받는 연결과 연관된 소스 IP 주소를 검사합니다. 서버는 받는 연결의 리터럴 IP 주소에 대해 역방향 지정 봉투 다시 쓰기를 시도하여 연관된 채널을 찾습니다. 소스 IP 주소가 `INTERNAL_IP` 매핑 테이블 내의 IP 주소나 서브넷과 일치하는 경우 해당 매핑 테이블을 호출하는 다시 쓰기 규칙을 통해 해당 주소가 `tcp_intranet` 채널로 다시 쓰여집니다.

`tcp_intranet` 채널은 `allowswitchchannel` 키워드로 표시되기 때문에 메시지는 `tcp_intranet` 채널로 전환되어 해당 채널로 들어갑니다. 메시지가 `INTERNAL_IP` 매핑 테이블에 없는 IP 주소의 시스템에서 들어오는 경우 역방향 지정 봉투 다시 쓰기로 `tcp_local` 또는 다른 채널로 다시 씁니다. 하지만 `tcp_intranet` 채널로는 다시 쓰지 않으며 다른 모든 채널은 기본적으로 `noswitchchannel`로 표시되어 있으므로 메시지는 다른 채널로 전환되지 않고 `tcp_local` 채널로 남아 있게 됩니다.

주 – "tcp_local" 문자열을 사용하는 모든 매핑 테이블이나 변환 파일 항목은 사용법에 따라 "tcp_*" 또는 "tcp_intranet"으로 변경해야 할 수 있습니다.

18.7.2 인증된 사용자의 메일 구분

사이트에는 물리적 네트워크의 일부가 아닌 "로컬" 클라이언트 사용자가 있을 수 있습니다. 이러한 사용자가 메시지를 전송하면 외부 IP 주소(예: 임의의 인터넷 서비스 제공자)로부터 메시지가 전송됩니다. 사용자가 SASL 인증을 수행할 수 있는 메일 클라이언트를 사용하는 경우 이러한 인증된 연결을 다른 외부 연결과 구분할 수 있습니다. 따라서 인증되지 않은 릴레이 전송 시도는 거부되는 반면 인증된 전송은 허용됩니다. 인바운드 SMTP 채널(일반적으로 tcp_local 채널)에 saslswitchchannel 키워드를 사용하여 인증된 연결과 인증되지 않은 연결을 구분할 수 있습니다.

saslswitchchannel 키워드는 전환할 채널을 지정하는 인수를 취합니다. SMTP 보낸 사람이 인증에 성공하면 해당 전송 메시지는 지정된 전환 대상 채널에서 오는 것으로 간주됩니다.

▼ 인증된 전송 구분을 추가하는 방법

1 구성 파일에 고유 이름을 가진 새 TCP/IP 채널 정의를 추가합니다. 예를 들면 다음과 같습니다.

```
tcp_auth smtp single_sys mx mustsaslserver noswitchchannel TCP-INTERNAL
```

이 채널은 정규 채널 전환을 허용하지 않아야 합니다(즉, 이전 기본 행에서 명시적 또는 암시적으로 noswitchchannel이 있어야 함). 이 채널에는 mustsaslserver가 있어야 합니다.

2 다음 예에 표시된 대로 maysaslserver 및 saslswitchchannel tcp_auth를 추가하여 tcp_local 채널을 수정합니다.

```
tcp_local smtp mx single_sys maysaslserver saslswitchchannel \
tcp_auth switchchannel
|TCP-DAEMON
```

이 구성을 사용하면 로컬 비밀번호로 인증할 수 있는 사용자가 보낸 SMTP 메일이 tcp_auth 채널에 들어갈 수 있습니다. 내부 호스트에서 보낸 인증되지 않은 SMTP 메일은 여전히 tcp_internal 채널로 들어옵니다. 다른 모든 SMTP 메일은 tcp_local로 들어옵니다.

18.7.3 메일 릴레이 금지

이 예에서는인증되지 않은 사용자가 시스템을 통해 SMTP 메일을 릴레이하지 못하도록 하는 것을 설명합니다. 우선 로컬 사용자는 SMTP 메일을 릴레이할 수 있어야 합니다. 예를 들어, POP 및 IMAP 사용자는 Messaging Server를 사용하여 메일을 보냅니다. 로컬 사용자는 물리적으로 메시지가 내부 IP 주소에서 들어오는 로컬이거나, 물리적으로는 원격이지만 로컬 사용자로 인증이 가능한 사용자일 수 있습니다.

인터넷 상에 있는 임의의 사용자가 해당 서버를 릴레이로 사용하지 못하게 하려 합니다. 다음 절에서 설명하는 구성을 사용하면 이러한 사용자 클래스를 구분하고 올바른 클래스를 차단할 수 있습니다. 특히 `tcp_local` 채널을 통해 들어오고 같은 채널을 통해 나가는 메일을 차단하려 합니다. 이를 위해 `ORIG_SEND_ACCESS` 매핑 테이블이 사용됩니다.

`ORIG_SEND_ACCESS` 매핑 테이블을 사용하여 소스 채널과 대상 채널을 기반으로 트래픽을 차단할 수 있습니다. 이 경우 `tcp_local` 채널을 통해 송수신되는 트래픽을 차단해야 합니다. 이 기능은 다음 `ORIG_SEND_ACCESS` 매핑 테이블로 구현됩니다.

```
ORIG_SEND_ACCESS

   tcp_local|*|tcp_local|*        $NRelaying$ not$ permitted
```

이 예에서 해당 항목은 메시지가 `tcp_local` 채널에 들어가서 바로 해당 채널로 다시 나올 수 없도록 지정합니다. 즉, 이 항목은 외부 메일이 SMTP 서버로 들어와서 곧바로 인터넷으로 릴레이되는 것을 방지합니다.

ims-ms 채널과 일치하는 주소(하지만 별칭이나 메일링 목록 정의를 통해 다시 외부 주소로 확장될 수 있는 주소)를 차단할 수 있도록 `SEND_ACCESS` 매핑 테이블 대신 `ORIG_SEND_ACCESS` 매핑 테이블이 사용됩니다. `SEND_ACCESS` 매핑 테이블을 사용할 때는 외부 사용자가 다시 외부 사용자로 확장되는 메일링 목록을 보내거나 메시지를 다시 외부 주소로 전달하는 사용자에게 보낼 수 있도록 하려면 길이를 늘여야 합니다.

18.7.4 SMTP 릴레이 차단에 RBL 검사를 포함한 DNS 조회 사용

Messaging Server에는 유효한 DNS 이름을 가진 주소에서 전송된 메일만 전달되도록 하는 여러 방법이 있습니다. 가장 간단한 방법은 `tcp_local` 채널에 `mailfromdnsverify` 채널 키워드를 지정하는 것입니다.

Messaging Server는 `ORIG_MAIL_ACCESS`에서 다음 규칙을 사용하여 유효한 DNS 이름을 가진 주소에서 전송된 메일만 전달되도록 하는 `dns_verify` 프로그램도 제공합니다.

```
ORIG_MAIL_ACCESS

  TCP|*|*|*|*|SMTP*|MAIL|*|*@*|*|*  \
```

```
$[msg-svr-base/lib/dns_verify.so,  \
dns_verify,$7|$$y|$$NInvalid$ host:$ $$7$ -$ %e]
```

위 예에서 줄 바꿈은 이러한 매핑 항목에서 구문적으로 매우 중요합니다. 다음 행으로 진행하려면 백슬래시 문자를 사용해야 합니다.

또한 `dns_verify` 이미지를 사용하여 받는 연결을 RBL(Realtime Blackhole List), MAPS(Mail Abuse Prevention System), DUL(Dial-up User List) 또는 ORBS(Open Relay Behavior-modification System) 목록 등에 대해 검사하여 UBE로부터 보호할 수 있습니다. 새 `mailfromdnsverify` 키워드와 마찬가지로 `dns_verify` 콜아웃을 수행하는 대신 "보다 간단한 구성" 방법으로 이러한 검사를 수행할 수도 있습니다. 보다 간단한 방법은 `dispatcher.cnf` 파일에 `DNS_VERIFY_DOMAIN` 옵션을 사용하는 것입니다. 예를 들어, `[SERVICE=SMTP]` 섹션에서 검사하려는 다양한 목록에 대한 옵션의 인스턴스를 설정합니다.

```
[SERVICE=SMTP]
PORT=25
! ...rest of normal options...
DNS_VERIFY_DOMAIN=sbl-xbl.spamhaus.org.
DNS_VERIFY_DOMAIN=list.dsbl.org.
...etc...
```

이 경우 메시지는 SMTP 수준에서 거부됩니다. 즉 메시지는 SMTP 대화 도중 거부되므로 MTA로 전송되지 않습니다. 이 방법의 단점은 내부 사용자가 보낸 메시지를 포함하여 모든 받는 SMTP 메시지를 검사한다는 것입니다. 따라서 효율성이 떨어지며 인터넷 연결이 중지되면 문제가 발생할 수 있습니다. 그 대안은 `PORT_ACCESS` 매핑 테이블 또는 `ORIG_MAIL_ACCESS` 매핑 테이블로부터 `dns_verify`를 호출하는 것입니다. `PORT_ACCESS` 매핑 테이블에는 로컬 내부 IP 주소나 메시지 전송자를 검사하지 않는 초기 항목과 다른 모든 사용자에 대해 원하는 검사를 수행하는 후기 항목을 지정할 수 있습니다. 또는 `ORIG_MAIL_ACCESS` 매핑 테이블에서 tcp_local 채널로 받는 메시지에만 검사를 적용하려는 경우에는 내부 시스템/클라이언트로부터 받는 메시지에 대해 해당 검사를 건너뛸 수 있습니다. `dns_verify`를 가리키는 항목을 사용하는 예는 다음과 같습니다.

```
PORT_ACCESS

! Allow internal connections in unconditionally
  *|*|*|*|* $C$|INTERNAL_IP;$3|$Y$E
! Check other connections against RBL list
  TCP|*|25|*|* \
$C$[msg-svr-base/lib/dns_verify.so,\
dns_verify_domain_port,$1,sbl-xbl.spamhaus.org.]EXTERNAL$E
ORIG_MAIL_ACCESS

  TCP|*|25|*|*|SMTP*|*|tcp_local|*@*|*|*  \
$C$[msg-svr-base/lib/dns_verify.so,\
dns_verify_domain,$1,sbl-xbl.spamhaus.org.]$E
```

18.7.4.1 DNS 기반 데이터베이스 지원

dns_verify 프로그램은 원치 않는 대량 전자 메일을 보낼 수 있는 받는 SMTP 연결을 확인하는 데 사용되는 DNS 기반 데이터베이스를 지원합니다. 공개적으로 사용 가능한 DNS 데이터베이스 중 일부는 일반적으로 이러한 용도로 사용되는 TXT 레코드를 포함하지 않을 수 있습니다. 대신 A 레코드만 포함합니다.

일반 설정에서 특정 IP 주소에 대한 DNS의 TXT 레코드에는 메시지를 거부할 때 SMTP 클라이언트로 반환하기에 적합한 오류 메시지가 포함되어 있습니다. 하지만 TXT 레코드가 없고 A 레코드가 있는 경우 Messaging Server 5.2 이전의 dns_verify 버전에서는 "*No error text available*"이라는 메시지를 반환했습니다.

이제 dns_verify에서는 사용 가능한 TXT 레코드가 없는 경우에 사용되는 기본 텍스트를 지정하는 옵션을 제공합니다. 예를 들어, 다음 PORT_ACCESS 매핑 테이블에서는 이 옵션을 사용하는 방법을 보여 줍니다.

```
PORT_ACCESS

   *|*|*|*|* $C$|INTERNAL_IP;$3|$Y$E  \
   TCP|*|25|*|*   \
$C$[<msg-svr-base/lib/dns_verify.so \
,dns_verify_domain_port,$1,dnsblock.siroe.com,Your$ host$ ($1)$ \
found$ on$ dnsblock$ list]$E
    * $YEXTERNAL
```

이 예에서 원격 시스템이 dnsblock.siroe.com 도메인의 쿼리에 있지만 TXT 레코드를 사용할 수 없는 경우에는 "*Your host a.b.c.d found on dnsblock list.*"

18.8 많은 수의 액세스 항목 처리

매핑 테이블에서 많은 수의 항목을 사용하는 사이트는 특정 조회에 대해 일반 텍스트 데이터베이스를 호출하는 몇 개의 일반적인 와일드카드 항목이 매핑 테이블에 포함되도록 구성해야 합니다. 매핑 테이블에 많은 수의 항목이 직접 존재하는 것보다 특정 조회에 대해 일반 텍스트 데이터를 호출하는 매핑 테이블 항목이 몇 개 있는 것이 더 효율적입니다.

특별한 경우 인터넷 전자 메일을 보내고 받을 수 있는 사용자별로 제어하려는 사이트가 있을 수 있습니다. 이러한 제어는 ORIG_SEND_ACCESS 등의 액세스 매핑 테이블을 사용하여 편리하게 구현될 수 있습니다. 이 때 대량의 특정 정보(예: 특정 주소)를 일반 텍스트 데이터베이스에 저장하고 매핑 테이블 항목을 일반 텍스트 데이터베이스로 적절하게 호출할 수 있도록 하면 효율성과 성능이 크게 향상될 수 있습니다.

예를 들어, 다음 ORIG_SEND_ACCESS 매핑 테이블을 살펴보십시오.

```
ORIG_SEND_ACCESS

! Users allowed to send to Internet
```

```
!
  *|adam@siroe.com|tcp_local|*    $Y
  *|betty@siroe.com|tcp_local|*   $Y
! ...etc...
!
! Users not allowed to send to Internet
!
  *|norman@siroe.com|tcp_local|*  $NInternet$ access$ not$ permitted
  *|opal@siroe.com|tcp_local|*    $NInternet$ access$ not$ permitted
! ...etc...
!
! Users allowed to receive from the Internet
!
  tcp_*|*|*|adam@siroe.com        $Y
  tcp_*|*|*|betty@siroe.com       $Y
! ...etc...
!
! Users not allowed to receive from the Internet
!
  tcp_*|*|*|norman@siroe.com      $NInternet$ e-mail$ not$ accepted
  tcp_*|*|*|opal@siroe.com        $NInternet$ e-mail$ not$ accepted
! ...etc...
```

테이블에 각 사용자가 개별적으로 입력된 매핑 테이블을 사용하는 것보다 더 효율적인 설정(수만 명의 사용자 항목이 있는 경우 특히 더 효율적임)이 아래 예에 나와 있습니다. 이 예에서는 일반 데이터베이스의 소스 텍스트 파일과 샘플 ORIG_SEND_ACCESS 매핑 테이블을 볼 수 있습니다. 설정 정보는 240 페이지 “10.9.1 MTA 텍스트 데이터베이스”를 참조하십시오.

DATABASE ENTRIES

```
SEND|adam@domain.com    $Y
SEND|betty@domain.com   $Y
! ...etc...
SEND|norman@domain.com  $NInternet$ access$ not$ permitted
SEND|opal@domain.com    $NInternet$ access$ not$ permitted
! ...etc...
RECV|adam@domain.com    $Y
RECV|betty@domain.com   $Y
! ...etc...
RECV|norman@domain.com  $NInternet$ e-mail$ not$ accepted
RECV|opal@domain.com    $NInternet$ e-mail$ not$ accepted
```

MAPPING TABLE

```
ORIG_SEND_ACCESS
```

```
! Check if may send to Internet
!
  *|*|*|tcp_local       $C${SEND|$1}$E
!
! Check if may receive from Internet
!
  tcp_*|*|*|*           $C${RECV|$3}$E
```

이 예에서 일반 데이터베이스 왼쪽에(그리고 이에 따라 매핑 테이블에 의해 생성된 일반 데이터베이스 검사에) 임의 문자열 SEND| 및 RECV|를 사용하면 두 가지 종류의 검사를 구분할 수 있습니다. 표시된 대로 일반 텍스트 데이터베이스 검사 앞뒤에 $C와 $E 플래그를 붙이는 것은 일반 데이터베이스에 대한 일반적인 매핑 테이블 호출입니다.

위의 예에서는 일반 텍스트 데이터베이스 항목에 대한 간단한 매핑 테이블 검사를 보여 줍니다. 보다 복잡한 검사를 수행하는 매핑 테이블도 일반 텍스트 테이블을 사용하여 효율성을 높일 수 있습니다.

18.9 2부. 메일함 필터

메일함 필터(시브(Sieve) 필터라고도 함)는 메시지 헤더에 지정된 문자열을 포함한 메시지를 필터링 하고 이러한 메일 메시지에 지정한 작업을 적용합니다. 관리자는 채널이나 MTA를 통해 사용자에게 가는 메일 스트림을 필터링합니다. Messaging Server 필터는 서버에 저장되며 서버에 의해 평가됩니다. 따라서 이를 서버측 규칙(SSR)이라고도 합니다.

이 부분은 다음 내용으로 구성되어 있습니다.

18.10 시브(Sieve) 필터 지원

Messaging Server 필터는 시브(Sieve) 필터링 언어(Draft 9 of the Sieve Internet Draft)를 기반으로 합니다. 시브(Sieve) 구문과 의미에 대한 자세한 내용은 RFC3028을 참조하십시오. 또한, Messaging Server는 다음의 시브(Sieve) 확장도 지원합니다.

- **jettison**. 메시지를 자동으로 삭제한다는 점에서는 discard와 비슷하지만, 암시적 보관을 취소할 뿐 다른 작업을 수행하지 않는 discard와 달리 jettison은 discard가 수행되도록 강제 설정합니다. 이러한 동작상의 차이는 여러 시브(Sieve) 필터가

포함된 경우에만 중요합니다. 예를 들어 시스템 수준 discard는 명시적으로 keep을 지정한 사용자 시브(Sieve) 필터에 의해 대체될 수 있지만 시스템 수준 jettison은 사용자 시브(Sieve)에서 수행한 모든 것을 대체합니다.

- **가장 시브(Sieve) 필터.** 한 사용자가 다른 사용자에 대해 시브(Sieve) 필터를 지정할 수 있는 방법을 제공합니다. 이 LDAP 옵션에서 제어하는 사용자 항목에 두 가지 LDAP 속성을 사용합니다.
  - LDAP_PARENTAL_CONTROLS - Yes 또는 No라는 문자열 값을 포함하는 속성을 지정합니다. Yes는 가장 시브(Sieve)가 이 항목에 적용됨을 의미하고 No는 그러한 시브(Sieve)가 적용되지 않음을 의미합니다. 기본값은 없습니다.
  - LDAP_FILTER_REFERENCE - 가장 시브(Sieve)를 찾을 수 있는 디렉토리 항목을 가리키는 DN을 포함한 속성을 지정합니다. 기본값은 없습니다.

    가장 시브(Sieve)가 포함되어 있는 항목에는 다음 MTA 옵션에서 지정한 두 가지 속성이 있어야 합니다.
  - LDAP_HOH_FILTER - 가장 시브(Sieve)를 포함하는 속성을 지정합니다. 이 옵션의 기본값은 mailSieveRuleSource입니다.
  - LDAP_HOH_OWNER - 가장의 소유자 전자 메일 주소를 포함하는 속성을 지정합니다. 이 옵션의 기본값은 mail입니다.

    두 속성 모두 가장 시브(Sieve)의 작동에 필요합니다.
- 이제 시브(Sieve) 리디렉션에서 세 개의 헤더 필드를 추가할 수 있습니다.

  ```
  resent-date: date-of-resend-operation
  resent-to: address-specified-in-redirect
  resent-from: addres-of-sieve-owner
  ```

  리디렉션에 대한 새로운 :resent 및 :noresent 인수는 이러한 필드의 추가 여부를 제어할 때 사용할 수 있습니다. 어느 인수도 지정되지 않으면 시스템 기본값이 사용됩니다. 시스템 기본값은 새로운 SIEVE_REDIRECT_ADD_RESENT MTA 옵션을 통해 제어됩니다. 이 옵션을 1로 설정하면 :noresent가 사용되지 않는 한 이러한 필드가 생성됩니다. 0으로 설정하면 :resent가 사용된 경우에만 필드가 생성됩니다. 이 옵션의 기본값은 1이며 이는 일반 리디렉션의 경우 기본적으로 필드가 생성됨을 의미합니다.
- 시브(Sieve) 리디렉션은 세 가지 새로운 인수로 향상되었습니다.

  :resetmailfrom - 봉투 FROM: 주소를 현재 시브(Sieve) 소유자의 주소로 재설정합니다.

  :keepmailfrom - 원본 메시지의 봉투 FROM:주소를 유지합니다.

  :notify - 리디렉션된 메시지에 대해 새로운 알림 플래그의 집합을 지정합니다. 알림 플래그 목록을 제공하는 단일 매개 변수가 필요합니다. DSN SMTP 확장의 NOTIFY 매개 변수에서 허용하는 동일한 플래그 집합이 SUCCESS , FAILURE, DELAY 및 NEVER 에도 허용됩니다. 이 플래그는 시브(Sieve) 목록으로 지정됩니다. 예를 들면, 다음과 같습니다.

  ```
  redirect :notify ["SUCCESS","FAILURE"] "foo@example.com";
  ```

:notify가 FAILURE, DELAY의 일반 SMTP 기본값으로 지정되지 않을 경우, 기본값이 됩니다. :notify가 지정되지 않은 경우 :keepmailfrom이 기본값입니다. :notify가 지정된 경우 기본값은 :resetmailfrom으로 바뀝니다. 추가 예외 사항은 SUCCESS 플래그를 지정할 경우 무조건적으로 :resetmailfrom을 사용해야 한다는 것입니다.

18.11 시브(Sieve) 필터링 개요

시브(Sieve) 필터는 메시지 헤더에 있는 문자열에 따라 메일 메시지에 적용되는 하나 이상의 조건부 작업으로 구성됩니다. 관리자는 채널 수준 필터와 MTA 차원 필터를 만들어서 원하지 않는 메일의 전달을 방지할 수 있습니다. 사용자는 Messenger Express를 사용하여 자신의 메일함에 사용자별 필터를 만들 수 있습니다. 구체적인 지침은 Messenger Express 온라인 도움말을 참조하십시오.

서버는 다음 우선 순위에 따라 필터를 적용합니다.

1. 사용자 수준 필터

   개인 메일함이 메시지를 명시적으로 수락하거나 거부하면 해당 메시지에 대한 필터 처리가 종료됩니다. 하지만 수신자에게 메일함 필터가 없거나 사용자의 메일함 필터가 해당 메시지에 명시적으로 적용되지 않는 경우에는 Messaging Server가 채널 수준 필터를 적용합니다. 사용자별 필터가 설정됩니다.

2. 채널 수준 필터

   채널 수준 필터가 메시지를 명시적으로 수락하거나 거부하면 해당 메시지에 대한 필터 처리가 종료됩니다. 그렇지 않으면 Messaging Server가 MTA 차원 필터(있는 경우)를 적용합니다.

3. MTA 차원 필터

기본적으로 각 사용자에게는 메일함 필터가 없습니다. 사용자가 Messenger Express 인터페이스를 사용하여 하나 이상의 필터를 만들면 해당 필터가 디렉토리에 저장되어 디렉토리 동기화 프로세스 도중 MTA에 의해 검색됩니다.

18.12 사용자 수준 필터 만들기

사용자별 메일 필터는 특정 사용자의 메일함을 대상으로 하는 메시지에 적용됩니다. 사용자별 메일 필터는 Messenger Express를 통해서만 만들 수 있습니다.

18.13 채널 수준 필터 만들기

채널 수준 필터는 채널에 대기된 각 메시지에 적용됩니다. 이러한 필터의 일반적 용도는 특정 채널을 통과하는 메시지를 차단하는 것입니다.

표 18–5 `filter` 채널 키워드 URL 패턴 대체 태그(대소문자 무시)

태그	의미
*	그룹 확장을 수행합니다.
**	속성 `mailForwardingAddress`를 확장합니다. 여러 전달 주소를 생성할 수 있는 값이 여러 개인 속성일 수 있습니다.
$$	$ 문자 대체입니다.
$\	후속 텍스트를 소문자로 바꿉니다.
$^	후속 텍스트를 대문자로 바꿉니다.
$_	후속 텍스트에 대해 대소문자 변환을 수행하지 않습니다.
$~	주소의 로컬 부분과 연관된 홈 디렉토리에 대한 파일 경로를 대체합니다.
$1S	`$S`와 비슷하지만 하위 주소를 사용할 수 없는 경우 아무 것도 삽입하지 않습니다.
$2S	`$S`와 비슷하지만 하위 주소를 사용할 수 없는 경우 아무 것도 삽입하지 않으며 선행 문자를 삭제합니다.
$3S	`$S`와 비슷하지만 하위 주소를 사용할 수 없는 경우 아무 것도 삽입하지 않으며 후행 문자를 무시합니다.
$A	주소 local-part@ host.domain을 대체합니다.
$D	host.domain을 대체합니다.
$E	두 번째 예비 속성 값, `LDAP_SPARE_1`을 삽입합니다.
$F	전달 파일의 이름(`mailDeliveryFileURL` 속성)을 삽입합니다.
$G	두 번째 예비 속성 값, `LDAP_SPARE_2`를 삽입합니다.
$H	호스트를 대체합니다.
$I	호스트된 도메인(`domainUidSeparator`에 의해 지정된 구분자의 오른쪽에 있는 UID 일부)을 삽입합니다. 호스트된 도메인을 사용할 수 없는 경우 실패합니다.
$1I	`$I`와 비슷하지만 호스트된 도메인을 사용할 수 없는 경우 아무 것도 삽입하지 않습니다.
$2I	`$I`와 비슷하지만 호스트된 도메인을 사용할 수 없는 경우 아무 것도 삽입하지 않고 선행 문자를 삭제합니다.
$3I	`$I`와 비슷하지만 호스트된 도메인을 사용할 수 없는 경우 아무 것도 삽입하지 않고 후행 문자를 무시합니다.

표 18–5 filter 채널 키워드 URL 패턴 대체 태그(대소문자 무시) *(계속)*

태그	의미
$L	로컬 부분을 대체합니다.
$M	호스트된 도메인을 제거하고 UID를 삽입합니다.
$P	메소드 이름(mailProgramDeliveryInfo 속성)을 삽입합니다.
$S	현재 주소와 연관된 하위 주소를 삽입합니다. 하위 주소는 하위 주소 구분자 뒤에 있는 원래 주소의 일부 사용자 부분입니다. 여기서 구분자는 일반적으로 +이지만 MTA 옵션 SUBADDRESS_CHAR로 지정할 수 있습니다. 하위 주소를 지정하지 않으면 실패합니다.
$U	현재 주소의 메일함 부분을 삽입합니다. 이것은 @ 기호 왼쪽에 있는 주소 전체이거나 하위 주소 구분자 + 앞에 있는 주소의 왼쪽 부분입니다.

▼ 채널 수준 필터 만들기

1 시브(Sieve)를 사용하여 필터를 작성합니다.

2 필터를 다음 디렉토리에 있는 파일에 저장합니다.

msg-svr-base/config/*file* .filter

이 파일은 세계 공용이어야 하며 MTA의 uid가 소유해야 합니다.

3 채널 구성에 다음을 포함합니다.

destinationfilter file:IMTA_TABLE:*file* .filter

4 구성을 다시 컴파일하고 디스패처를 다시 시작합니다.

필터 파일의 변경 내용은 다시 컴파일하거나 디스패처를 다시 시작하지 않아도 적용됩니다.

destinationfilter 채널 키워드를 통해 해당 채널의 대기열에 **포함된** 메시지에 대한 메시지 필터링을 사용할 수 있습니다. sourcefilter 채널 키워드를 통해 채널에 **의해**(로부터) 대기된 메시지에 대한 메시지 필터링을 사용할 수 있습니다. 이러한 키워드에는 채널과 연관된 해당 채널 필터 파일에 대한 경로를 지정하는 하나의 필수 매개 변수가 있습니다.

destinationfilter 채널 키워드 구문은 다음과 같습니다.

destinationfilter *URL-pattern*The syntax for the sourcefilter channel keyword is:

sourcefilter *URL-pattern*where *URL-pattern* is a URL specifying the path to the filter file for the channel in questionIn the following example, *channel-name* is the name of the channel.

destinationfilter file:///usr/tmp/filters/*channel-name*.filter

filter 채널 키워드를 통해 해당 채널에 대한 메시지 필터링을 사용할 수 있습니다. 키워드에는 채널을 통해 메일을 받는 각 봉투 수신자와 연관된 필터 파일의 경로를 지정하는 하나의 필수 매개 변수가 있습니다.

filter 채널 키워드의 구문은 다음과 같습니다.

```
filter URL-pattern
```

*URL-pattern*은 특별한 대체 시퀀스를 처리한 후 경로를 특정 수신 주소에 대한 필터 파일로 지정하는 URL입니다. *URL-pattern*은 특별 대체 시퀀스 발생 시 이를 포함할 수 있으며, 이 시퀀스는 수신 주소(해당 local-part@host.domain)에서 추출된 문자열로 대체될 수 있습니다. 이러한 대체 시퀀스는 표 18–5에 나와 있습니다.

fileinto 키워드는 메일함 필터 fileinto 연산자가 적용되었을 때 주소를 변경하는 방법을 지정합니다. 다음 예에서는 폴더 이름이 다음과 같이 원래 있던 하위 주소를 대체하면서 원래 주소의 하위 주소로 삽입되어야 한다는 것을 지정합니다.

```
fileinto $U+$S@$D
```

18.14 MTA 차원 필터 만들기

MTA 차원 필터는 MTA에 대해 대기된 모든 메시지에 적용됩니다. 이 필터의 일반적 용도는 메일의 대상에 관계 없이 원하지 않는 대량 전자 메일이나 기타 원하지 않는 메시지를 차단하는 것입니다. MTA 필터를 만들려면 다음을 수행합니다.

▼ MTA 차원 필터 만들기

1. **시브(Sieve)를 사용하여 필터를 작성합니다.**

2. **다음 파일에 해당 필터를 저장합니다.**

   msg-svr-base/config/imta.filter

   이 필터는 모두가 읽을 수 있어야 하며이 파일이 있으면 자동으로 사용됩니다.

3. **구성을 다시 컴파일하고 디스패처를 다시 시작합니다.**

   컴파일된 구성을 사용하면 MTA 차원 필터 파일은 컴파일된 구성에 통합됩니다.

18.14.1 제거된 메시지를 FILTER_DISCARD 채널 외부로 라우팅

기본적으로 메일함 필터를 통해 제거된 메시지는 즉시 시스템에서 제거(삭제)됩니다. 하지만 사용자가 처음 메일함 필터를 설정할 때나(또는 실수로) 디버깅을 위해 삭제 작업이 일정 시간 동안 지연되도록 할 수 있습니다.

메일함 필터에 의해 제거된 메시지를 시스템에 일시 보관한 후 나중에 삭제하려면 먼저 다음 예에 표시된 대로 삭제할 때까지 메시지를 보관할 기간(일반적으로 일 수)을 지정하는 notices 채널 키워드와 함께 filter_discard 채널을 MTA 구성에 추가합니다.

```
filter_discard notices 7
FILTER-DISCARD
```

그런 다음 MTA 옵션 파일에서 FILTER_DISCARD=2 옵션을 설정합니다. filter_discard 대기열에 있는 메시지는 사용자의 개인 휴지통 폴더의 확장된 범위에 들어 있는 것으로 간주해야 합니다. 따라서 filter_discard 대기열에 있는 메시지에 대한 경고 메시지는 전송되지 않으며 바운스 또는 반환 요청 시에도 보낸 사람에게 반환되지 않습니다. 이러한 메시지에 대해 수행 가능한 유일한 작업은 최종 알림 값이 만료되거나 imsimta return 등의 유틸리티를 사용하여 수동 바운스가 요청된 경우 해당 메시지를 영구적으로 삭제하는 것입니다.

Messaging Server 6 2004Q2 전에는 jettison 시브(Sieve) 작업에서 filter_discard 채널 사용을 FILTER_DISCARD MTA 옵션으로 제어했습니다. 이제는 FILTER_DISCARD 설정에서 기본값을 가져오는 FILTER_JETTISON 옵션으로 제어합니다. FILTER_DISCARD의 기본값은 1입니다(discard를 bitbucket 채널로 전달).

18.15 사용자 수준 필터 디버그

사용자가 시브(Sieve) 필터가 제대로 작동하지 않는다고 불평할 경우 여러 단계를 수행하여 필터를 디버깅할 수 있습니다. 여기에서 이러한 단계에 대해 설명합니다.

▼ 사용자 수준 필터 디버그

1 fileinto 필터링이 작동하려면 imta.cnf 파일에서 ims-ms 채널이 다음과 같이 표시되어 있어야 합니다.

```
fileinto $U+$S@$D
```

2 사용자의 LDAP 항목에서 사용자 수준 필터를 가져옵니다.

사용자 수준 필터는 MailSieveRuleSource 속성 아래의 LDAP 항목에 저장됩니다. ldapsearch 명령을 사용하여 검색하려는 경우 이러한 필터가 base64 인코딩되어 있으므로 -Bo 스위치를 사용하여 출력을 디코딩해야 합니다.

```
./ldapsearch -D "cn=directory manager" -w password -b
"o=alcatraz.sesta.com,o=isp" -Bo uid=test
```

또한 아래 설명된 imsimta test -rewrite 명령을 사용하면 디코딩이 자동으로 수행됩니다.

3 사용자 필터가 MTA에 표시되는지 확인합니다.

다음 명령을 실행합니다.

```
# imsimta test -rewrite -filter -debug user@sesta.com
```

이렇게 하면 앞 단계에서 검색한 사용자의 시브(Sieve) 필터가 출력되어야 합니다. 필터가 표시되지 않으면 LDAP 항목이 필터를 반환하지 않는 이유를 찾아야 합니다. `imsimta test -rewrite` 출력에 필터가 표시되면 MTA가 사용자의 필터를 인식하는 것입니다. 다음 단계에서는 `imsimta test -expression` 명령을 사용하여 필터 해석을 테스트합니다.

4 `imsimta test -exp`**를 사용하여 사용자 필터를 디버깅합니다. 다음 정보가 필요합니다.**

a. `mailSieveRuleSource` **속성에 있는 사용자의 시브(Sieve) 언어 문. 위 단계를 참조하십시오.**

b. 필터를 트리거한 것으로 여겨지는 rfc2822 메시지

c. 필터가 메시지에 대해 수행할 것으로 예상되는 작업에 대한 설명

5 사용자의 `mailSieveRuleSource: values`**를 기반으로 시브(Sieve) 언어 문을 포함하는 텍스트 파일(예:** `temp.filter`**)을 만듭니다. 예:**

```
require "fileinto";
if anyof(header :contains
["To","Cc","Bcc","Resent-to","Resent-cc",
   "Resent-bcc"] "commsqa"){
   fileinto "QMSG";
}
```

예상 결과: `commsqa`가 이 메시지의 수신자일 경우 메시지를 `QMSG`라는 폴더에 정리합니다.

6 사용자가 제공한 rfc2822 메시지 파일의 내용을 포함하는 `test.msg`**라는 텍스트 파일을 만듭니다.**

사용자 메시지 저장소 영역의 `.msg` 파일을 사용하거나 사용자가 제공한 rfc2822 메시지 파일의 내용을 포함하는 `test_rfc2822.msg`라는 텍스트 파일을 만들 수 있습니다.

7 `imsimta test -exp` **명령을 사용합니다.**

```
# imsimta test -exp -mm -block -input=temp.filter -message=test_rfc2822.msg
```

8 출력을 검사합니다.

`imsimta test -exp` 명령의 마지막 줄은 시브(Sieve) 해석의 결과를 표시합니다. 이 결과는 다음과 같습니다.

```
Sieve Result: []
or this:
Sieve Result: [action]
```

여기서 *action*은 이 메시지에서 시브(Sieve) 필터를 적용한 결과로 수행되는 작업입니다.

필터 기준이 일치하면 몇 가지 작업이 결과로 표시됩니다. 필터 기준이 일치하지 않으면 빈 시브(Sieve) 결과가 표시되며 시브(Sieve) 필터에 논리적 오류가 있거나 .msg 파일에 일치하는 정보가 포함되지 않은 것입니다. 다른 오류가 발생할 경우에는 시브(Sieve) 스크립트에 구문 오류가 있는 것이므로 이를 디버깅해야 합니다.

출력에 대한 자세한 내용은 544 페이지 "18.15.1 imsimta test -exp 출력"을 참조하십시오.

9 필터 구문이 유효하고 결과가 올바를 경우 다음 단계는 tcp_local_slave.log 디버그 로그 파일을 검사하는 것입니다.

테스트하는 메시지 파일과 전송되는 메시지 파일이 다를 수 있습니다. 무엇이 수신되는지 확인하는 방법은 tcp_local_slave.log 파일을 검사하는 것뿐입니다. 이 로그에는 MTA로 보내는 메시지와 이 메시지에 필터를 적용하는 방법이 표시되어 있습니다.

tcp_local_slave.log 디버그 파일을 가져오는 방법은 388 페이지 "12.11.2 디버깅 키워드"의 slave_debug 키워드를 참조하십시오.

18.15.1 imsimta test -exp 출력

imsimta test -exp의 전체 명령은 다음과 같습니다.

```
# imsimta test -exp -mm -block -input=temp.filter -message=rfc2822.msg
```

출력 예는 다음과 같습니다.

예 18–4 imsimta test -exp 출력

```
# imsimta test -exp -mm -block -input tmp.filter -message=rfc2822.msg
Expression: if header :contains ["to"] ["pamw"]       (1)
Expression: {
Expression: redirect "usr3@sesta.com";
Expression: keep;
Expression: }
Expression:
Expression: Dump: header:2000114;0  3  1  :contains  1  "to"  1
"pamw"  if  8  ;
Dump: redirect:2000121;0  1  1  "usr3@sesta.com"  ;  keep:2000117;0 (2)
Dump: 0
Result: 0
Filter result: [ redirect "usr3@sesta.com" keep ]    (3)
```

1) Expression: 출력 행은 tmp.filter 텍스트 파일에서 읽고 구문 분석될 필터를 표시합니다. 이러한 행은 스크립트를 디버깅하는 데 그다지 유용하지 않습니다.

2) Dump: 출력 행은 시브(Sieve) 문을 해석하는 컴퓨터의 결과입니다. 오류가 표시되지 않아야 하며 출력이 입력과 일치하는 것으로 보여야 합니다. 예를 들어, 이 덤프에서 필터 파일 `redirect "usr3@sesta.com";`의 행과 같은 단어 `redirect`, `usr3@sesta.com`을 표시해야 합니다.

일치하는 텍스트가 표시되지 않은 경우에는 신경을 써야 합니다. 그렇지 않은 경우에는 스크립트를 디버깅하는 데 그다지 유용하지 않습니다.

3) 출력의 맨 아래에 `Filter result:` 문이 나타납니다. 앞에서 언급한 것처럼 다음과 같은 두 가지 결과가 가능합니다.

`Sieve Result: []` 또는`Sieve Result: [`*action*`]`

여기서 action은 시브(Sieve) 스크립트가 수행하는 작업입니다. 경우에 따라서는 빈 결과를 예상할 수도 있습니다. 예를 들어, discard 필터의 경우에는 테스트하는 모든 `.msg` 파일을 항상 삭제하지 않는지 테스트해야 합니다. 예를 들어 대괄호 사이에 작업이 있는 경우

```
Filter result: [ fileinto "QMSG" keep]
```

`rfc2822.msg` 파일의 텍스트가 필터 기준과 일치했다는 것을 의미합니다. 이 특정 예에서 필터는 메일을 `QMSG` 폴더에 파일로 저장하고 복사본을 받은 메일함에 보관합니다. 이 경우의 결과 작업은 `fileinto` 및 `keep`입니다.

필터를 테스트할 때 두 결과 모두에 대해 여러 `.msg` 파일을 테스트해야 합니다. 필터와 일치하는 메시지가 필터링되는지, 일치시키지 않으려는 메시지가 필터링되지 않는지 항상 테스트해야 합니다.

와일드카드 일치의 경우에는 `:contains`가 아니라 `:matches` 테스트를 사용해야 한다는 것에 주의합니다. 예를 들어, `from=*@sesta.com`을 일치시키려면 `:matches`를 사용해야 합니다. 그렇지 않으면 테스트 조건을 전혀 만족하지 않으므로 테스트가 실패합니다.

18.15.2 imsimta test -exp 구문

`imsimta test -exp`는 지정된 RFC2822 메시지에 대해 시브(Sieve) 언어 문을 테스트하고 필터 결과를 표준 출력으로 보냅니다.

구문은 다음과 같습니다.

```
imsimta test -exp -mm -block -input=Sieve_language_scriptfile
-message=rfc2822_message_file
```

여기서

`-block`은 전체 입력을 단일 시브(Sieve) 스크립트로 처리합니다. 기본값은 각 행을 별개의 스크립트로 처리하고 별개로 평가하는 것입니다. 시브(Sieve)는 파일의 끝에 도달한 경우에만 평가됩니다.

`-input=`*Sieve_file*은 시브(Sieve) 스크립트를 포함하는 파일입니다. 기본값은 stdin에서 테스트 스크립트 행이나 스크립트 블록을 읽는 것입니다.

`-message=`*message_file*은 시브(Sieve) 스크립트를 테스트할 RFC 2822 메시지를 포함하는 텍스트 파일입니다. 이 파일은 반드시 RFC 2822 메시지여야 하며 대기열 파일이 될 수 없습니다(zz*.00 파일이 아님).

이 명령은 활성화될 경우 스크립트 정보를 읽어 테스트 메시지의 컨텍스트에서 평가한 다음 결과를 기록합니다. 결과에는 스크립트의 최종 문을 평가한 결과뿐만 아니라 수행되는 작업도 표시됩니다.

유용한 추가 한정자는 다음과 같습니다.

`-from=`*address*는 봉투 테스트에 사용할 봉투의 `from:` 주소를 지정합니다. 기본값은 `RETURN_ADDRESS` MTA 옵션에 지정된 값을 사용하는 것입니다.

`-output=`*file*은 결과를 *file*에 기록합니다. 기본값은 스크립트 평가 결과를 stdout에 기록하는 것입니다.

◆ ◆ ◆ 19장

19 MeterMaid를 사용하여 받는 연결 억제

MeterMaid는 IP 주소 SMTP 봉투 주소 모니터링 등을 통해 연결과 트랜잭션의 중앙 집중식 측정 및 관리를 수행할 수 있는 서버입니다. 기능적으로 MeterMaid는 특정 IP 주소가 MTA와 연결 가능한 빈도를 제한할 때 사용할 수 있습니다. 특정 IP 주소와의 연결을 제한하는 기능은 서비스 거부 공격에 사용되는 과도한 연결을 방지하는 데 유용합니다. MeterMaid는 `conn_throttle.so`와 비슷한 기능을 제공하면서 이를 대체하지만 Messaging Server 설치 전체에 걸쳐 그 기능을 확장합니다. `conn_throttle.so`의 기능을 새롭게 향상시킬 계획은 없으며, MeterMaid로 대체하는 것이 더 효과적입니다.

이 절은 다음과 같은 하위 절로 구성되어 있습니다.

19.1 기술 개요

`conn_throttle.so`는 특정 IP가 최근에 너무 자주 연결되어 잠시 무시할 필요가 있는 시점을 확인하기 위해 받는 연결의 메모리 내장 테이블을 사용하는 MTA 매핑 테이블의 콜아웃으로 쓰이는 공유 라이브러리입니다. 메모리 내장 테이블을 사용하면 성능에 도움이 되지만, 각 서버의 개별 프로세스가 자체 테이블을 유지하기 때문에 비용이 커집니다.

대부분의 경우 `conn_throttle.so` 콜아웃은 각 시스템의 단일 프로세스인 디스패처에서 액세스하는 `PORT_ACCESS` 매핑으로 수행됩니다. 서버 당 별도의 테이블이 있는 것 외에는 비용이 없습니다.

MeterMaid로 인해 주로 향상된 부분은 Messaging Server 환경의 모든 시스템과 프로세스에서 액세스할 수 있는 단일 억제 정보 저장소를 MeterMaid가 유지한다는 점입니다. MeterMaid는 계속해서 메모리 내장 데이터베이스에 데이터를 저장하여

성능을 극대화합니다. MeterMaid를 다시 시작하면 이전에 저장한 정보가 모두 손실되지만 보통 데이터의 수명이 매우 짧기 때문에 다시 시작하는 경우(거의 수행되지 않음)에 발생하는 비용은 매우 적습니다.

19.2 작동 원리

MeterMaid의 구성은 msg.conf에 저장되며 configutil에서 유지 관리됩니다.

MTA에서 check_metermaid.so를 사용하면 매핑 테이블 콜아웃을 통해 MeterMaid에 액세스할 수 있습니다. 이 기능은 어느 *_ACCESS 테이블에서나 호출할 수 있습니다. PORT_ACCESS 테이블에서 호출한 경우에는 연결의 IP 주소를 기반으로 제한을 확인하는 데에 사용할 수 있습니다. 이 방법은 오래된 conn_throttle.so를 대체하여 MeterMaid를 구현하는 경우에 가장 흔히 사용되는 방법입니다. 다른 *_ACCESS 테이블에서 호출하는 경우 MeterMaid는 봉투 발신 또는 봉투 수신 주소와 IP 주소 등의 다른 데이터를 제한하는 데에도 사용할 수 있습니다.

check_metermaid.so에는 시작점이 하나만 정의됩니다. throttle 루틴에서는 MeterMaid에 연결하고 쉼표로 구분된 두 후속 인수를 제공합니다. 첫 인수는 데이터 검사 기준으로 사용할 테이블의 이름이며, 둘째 인수는 검사할 데이터입니다.

검사 결과 확인된 특정 데이터가 해당 테이블에서 할당량을 초과한 것이 밝혀지면 check_metermaid.so에서 매핑 엔진이 입력을 계속 처리할 수 있도록 "success"를 반환합니다. 그 후에 항목의 나머지 부분을 사용하여 할당량을 초과한 이 연결을 처리합니다.

```
PORT_ACCESS

  *|*|*|*|* $C$|INTERNAL_IP;$3|$Y$E
  *|*|*|*|* $C$:A$[/opt/SUNWmsgsr/lib/check_metermaid.so,throttle,tablename,$3]$N421$ \
Connection$ declined$ at$ this$ time$E
  *          $YEXTERNAL
```

매핑 테이블 항목에서 check_metermaid.so 호출 앞에 있는 $:A 플래그 테스트에 주의하십시오. 디스패처에서 검사에 A 플래그를 설정하면서 PORT_ACCESS를 확인할 때 MeterMaid 검사만 수행하기 위한 것입니다.

19.3 MeterMaid의 Configutil 매개 변수

MeterMaid의 구성은 msg.conf에 저장되며 configutil에서 유지 관리됩니다. 다음은 현재 MeterMaid에서 지원하는 설정입니다. 기본값은 괄호 안에 있습니다. MeterMaid 매개 변수의 전체 목록은 **Sun Java System Messaging Server 6.3 Administration Reference**의 "configutil Parameters"를 참조하십시오.

- local.metermaid.enable Watcher에서 MeterMaid를 시작하고 제어할 수 있도록 MeterMaid 데몬을 실행하는 시스템에서 이 설정을 yes로 설정해야 합니다.

- `logfile.metermaid.*`. 이 설정은 IMAP, POP 및 기타 서비스에서 사용되는 것과 같습니다. 기본적으로 MeterMaid는 `msg-svr-base/data/log/metermaid`에 로그 파일을 기록합니다.
- `metermaid.config.listenaddr`(`INADDR_ANY`). MeterMaid를 바인드할 주소입니다. 대부분의 시스템에서는 기본값을 변경할 필요가 없지만, 홈이 여럿인 시스템이나 HA 시스템의 경우에는 여기에 적절한 주소를 지정하는 것이 좋습니다.
- `metermaid.config.maxthreads`(20). MeterMaid 서버는 다중 스레드이며 작업을 예약하는 스레드 풀을 유지 관리합니다. 이 값은 MeterMaid에서 사용되는 스레드의 최대 수를 설정합니다. CPU가 5개 이상인 시스템에서 이 값을 늘리면 전체 처리량이 늘어납니다.
- `metermaid.config.port`(63837). MeterMaid에서 연결을 수신하며 MeterMaid 클라이언트가 연결하는 포트입니다.
- `metermaid.config.secret`(기본값 없음, 값 입력 필요). MeterMaid는 받는 연결을 인증하기 위해 클라이언트가 MeterMaid에 연결한 후 보내는 공유 비밀을 사용합니다.
- `metermaid.config.serverhost`(기본값 없음, 값 입력 필요). 클라이언트가 연결하는 호스트 이름 또는 IP 주소입니다. `metermaid.config.listenaddr`와 같은 수도 있지만 Messaging Server 환경에 있는 특정 시스템 하나로 클라이언트를 보내는 특정 값이 있는 경우가 많습니다.

이 설정은 `check_metermaid` 클라이언트에서 사용됩니다.

- `metermaid.mtaclient.connectfrequency` (15) `connectfrequency` 초마다 연결을 시도합니다. MeterMaid와 연결할 필요가 있는 클라이언트는 MeterMaid가 사용 불가능한 상태에서 계속 연결이 시도되지 않도록 이를 내부 억제로 사용합니다. 클라이언트가 MeterMaid와 통신할 수 없는 경우에는 MeterMaid가 이 연결을 차단하지 않은 것을 알리는 MTA 매핑 엔진에 "fail" 상태를 반환합니다.

  예를 들어, `check_metermaid.so`가 MeterMaid와의 연결을 시도하지만 어떤 이유로 실패한 경우 `metermaid.mtaclient.connectfrequency` 에서 지정한 대로 다음 N초간은 추가적인 시도가 이루어지지 않습니다. MeterMaid가 작동하지 않는 경우, `check_metermaid.so`가 너무 자주 MeterMaid와의 연결을 시도하지 않게 해줍니다.
- `metermaid.mtaclient.connectwait`(5). 클라이언트가 MeterMaid와의 연결을 기다리는 경우(초기 연결 또는 이미 설정된 연결 재사용), `connectwait` 초간 기다린 후에 "fail" 상태를 반환하고 이 연결이 계속되도록 허용합니다.
- `metermaid.mtaclient.debug`(no). 이 옵션이 활성화되어 있으면 서버나 SMTP 서버의 스레드별 로그 파일에 클라이언트의 디버깅 정보가 출력됩니다.
- `metermaid.mtaclient.maxconns`(3). 다중 스레드 서버를 지원하기 위해 클라이언트에서 MeterMaid에 대한 연결 풀을 유지 관리할 수 있습니다. 그러면 통신 중의 동시 처리를 늘릴 수 있습니다. 하지만 MeterMaid에서 수행되는 내부 잠금 때문에 특정 테이블에 대한 액세스는 한 번에 하나의 요청으로 제한되므로, 한 프로세스에서 여러 연결을 사용할 경우 이점이 제한될 수 있습니다.
- `metermaid.mtaclient.readwait`(10). MeterMaid와 통신할 때 클라이언트는 `readwait`초간 기다린 다음 fail 상태를 반환하고 이 연결이 계속되도록 허용합니다.

마지막으로, 억제 테이블은 표시된 것과 같이 msg.conf에도 정의됩니다. 각 구성 매개 변수에 있는 *는 정의되는 특정 테이블의 이름을 나타냅니다. 예를 들어, internal이라는 테이블의 경우 첫 매개 변수는 metermaid.table.internal.data_type이 됩니다.

- metermaid.table.*.data_type(string). MeterMaid는 테이블에서 문자열과 ipv4라는 두 가지 데이터를 지원할 수 있습니다. 문자열 데이터는 항목 당 255바이트로 제한되며 대소문자를 구분하거나 구분하지 않는 함수를 사용하여 비교할 수 있습니다(아래 metermaid.table.*.options 참조).
- metermaid.table.*.max_entries(1000). MeterMaid에서 각 테이블을 초기화할 때 이만큼의 항목을 미리 할당합니다. MeterMaid는 오래된 항목이 만료되지 않은 경우에도 자동으로 리사이클합니다. 새 연결을 받은 경우 MeterMaid는 가장 오래 전에 액세스한 항목을 다시 사용합니다. 사이트에서 quota_time 중에 받은 연결을 캐시하기에 충분하도록 높은 값을 지정해야 합니다.
- metermaid.table.*.options는 테이블의 동작 또는 특성을 정의하는 키워드를 포함하는, 쉼표로 분리된 목록입니다. 유효한 키워드는 다음과 같습니다.
  - nocase — 데이터 작업에서 모든 비교는 대소문자를 구분하지 않는 비교 함수를 통해 이루어집니다. (문자열 데이터의 경우에만 이 옵션이 유효합니다.)
  - penalize — quota_time초가 지나면 일반적으로 억제에서 연결 수를 0으로 재설정하지만, 축소 옵션을 사용하는 경우 억제에서는 quota_time동안 추가 연결 시도가 축소되도록 연결 수를 할당량만큼 줄입니다(0보다 작지는 않음). 예를 들어, 할당량이 5이고 quota_time이 60인 경우 시스템에서 처음 1분간 12개의 연결 시도를 받으면 처음 5개의 연결은 승인하고 나머지 7개의 연결은 거부합니다. 60초가 지난 후 특정 주소에 대한 연결 수는 7개로 줄지만 할당량보다는 높은 값으로 유지되어 연결 시도는 거부됩니다. 다시 60초가 지나는 동안 새로운 연결 시도가 이루어지지 않으면 연결 수는 다시 2로 줄며 MeterMaid에서 다시 연결 시도를 허용하게 됩니다.
- metermaid.table.*.quota(100). 연결을 받으면 할당량을 기준으로 수를 계산합니다. quota_time초 동안 받은 연결의 수가 이 값을 초과하면 MeterMaid에서 연결을 거부합니다. (받는 연결에 미치는 실제 효과는 매핑 테이블에서 제어하며 결과는 추가 조사, 지연 또는 연결 거부 등이 될 수 있습니다.)
- metermaid.table.*.quota_time(60). 이 값은 quota를 기준으로 연결 수를 계산하는 시간(초)을 지정합니다. 이 시간이 지나고 나면 이 테이블의 type에 따라 받는 주소에 대해 계산된 연결 수가 감소합니다.
- metermaid.table.*storage(hash). MeterMaid에서는 hash와 splay의 두 가지 저장 방법을 사용할 수 있습니다. 기본 해시 테이블 방법을 사용하는 것이 좋지만 경우에 따라 스프레이 트리에서 더 빠른 조회를 제공할 수도 있습니다.
- metermaid.table.*.type (throttle). 현재 MeterMaid에서 지원되는 유일한 테이블 유형은 throttle입니다. 이 테이블 유형에서는 데이터(보통 IP 주소)를 추적하여 quota_time초 동안 받는 연결을 quota개로 억제합니다.

19.4 과도한 IP 주소 연결을 Metermaid를 사용하여 제한—예

이 예에서는 MeterMaid를 사용하여 분당 10개 연결로 IP 주소를 억제합니다. 참고로, 매핑 파일에서 이와 동일한 conn_throttle.so 설정은 다음과 같습니다.

```
PORT_ACCESS
   *|*|*|*|*  $C$|INTERNAL_IP;$3|$Y$E
   *|*|*|*|*  $C$[/opt/SUNWmsgsr/lib/conn_throttle.so,throttle,$3,10]\
$N421$ Connection$ declined$ at$ this$ time$E
   *          $YEXTERNAL
```

이 PORT_ACCESS 매핑 테이블은 외부 연결의 경우 분당 10개 이하의 속도로 제한하도록 conn_throttle.so를 구현합니다.

두 기술의 근본적인 차이점 중 하나는 MeterMaid가 억제를 위해 속도 제한과 같은 세부 정보를 직접 매핑 테이블에 구성하지 않고 configutil 매개 변수를 이 설정에 사용한다는 것입니다. 이 예에 대해서는 아래에서 설명합니다.

1. **시스템 중 하나를 MeterMaid 서버 호스트로 지정합니다.**

   이 시스템에서 다음 configutil 매개 변수를 설정합니다.

   ```
   local.metermaid.enable -v TRUE
   ```

   클라이언트와 MeterMaid 서버 간의 통신을 확인하는 데 사용되는 인증 비밀번호를 설정합니다.

   ```
   configutil -o metermaid.config.secret -v password
   ```

2. **억제 테이블을 정의합니다.**

   MeterMaid의 억제 동작은 작동 특성을 정의하는 명명된 억제 테이블에 의해 결정됩니다. 분당 10개 연결의 속도로 억제하는 테이블을 정의하려면 다음 매개 변수를 설정합니다.

   ```
   configutil -o metermaid.table.ext_throttle.data_type -v ipv4
   configutil -o metermaid.table.ext_throttle.quota -v 10
   ```

   *ext_throttle*은 억제 테이블의 이름입니다. ipv4는 데이터 유형 Internet Protocol 버전 4 주소 표현입니다. 10은 할당량(연결 한도)입니다.

3. **MeterMaid 시스템에서 MeterMaid를 시작합니다.**

   ```
   # start-msg metermaid
   ```

4. **MTA가 MeterMaid를 사용하여 억제를 수행하는 시스템에서 MeterMaid 호스트와 비밀번호를 지정합니다.**

   이는 필수 항목입니다.

```
configutil -o metermaid.config.secret -v MeterMaid_Password
configutil -o metermaid.config.serverhost -v name_or_ipaddress_of_MetermaidHost
```

5. **MeterMaid** PORT_ACCESS **테이블을 설정합니다.**

   이 테이블은 동등한 conn_throttle.so 설정과 비슷합니다.

```
PORT_ACCESS

   *|*|*|*|*  $C$|INTERNAL_IP;$3|$Y$E
   *|*|*|*|*  $C$:A$[/opt/SUNWmsgsr/lib/check_metermaid.so,throttle,\
ext_throttle,$3] $N421$ Connection$ declined$ at$ this$ time$E
   *          $YEXTERNAL
```

   첫 번째 행은 연결을 시도하는 IP 주소가 내부 주소인지 확인합니다. 내부 주소인 경우 연결을 허용합니다. 두 번째 행은 MeterMaid를 통해 IP 주소를 실행하며, 너무 자주 연결한 경우 연결을 거부합니다. 세 번째 행은 그 밖의 다른 연결을 허용하지만 EXTERNAL이라는 플래그를 지정합니다.

   check_metermaid.so에 대한 이 호출은 conn_throttle.so에 대한 콜아웃과 유사합니다. check_metermaid.so의 함수가 동일합니다. throttle과 해당 인수는 단지 metermaid.table*tablename*을 사용하여 구성한 테이블 이름과 검사할 IP 주소($3)입니다. conn_throttle.so처럼 이 함수도 metermaid.table.ext_throttle.quota에서 지정한 한도에 도달하면 *success*를 반환합니다. 따라서 매핑 항목 행의 나머지가 처리될 수 있으며, 원격 SMTP 클라이언트에게 메시지(421 SMTP 코드, 임시 부정 완료, **이 시점에 허용되지 않는 연결**)가 보내지고 디스패처는 연결 종료를 지시받습니다.

   $:A 때문에 이 행은 디스패처로부터 호출되는 경우에만 처리됩니다. 이 설정이 없으면 check_metermaid.so에 대한 호출은 PORT_ACCESS 매핑 테이블 검사도 수행하는 tcp_smtp_server 프로세스의 컨텍스트에서도 발생합니다. 따라서 MeterMaid는 받는 연결 각각을 두 번씩 계산합니다.

이는 conn_throttle.so를 대체하도록 MeterMaid를 설정하는 기본 구성입니다. 이 항목에 대한 자세한 내용은 217 페이지 “10.3.2 매핑 작업” 및 521 페이지 “18.3.4 PORT_ACCESS 매핑 테이블”을 참조하십시오.

19.4.1 기타 유용한 MeterMaid 옵션

경우에 따라서는 두 가지 추가 구성 옵션이 유용할 수 있습니다. conn_throttle.so 공유 라이브러리에는 throttle_p 함수도 있습니다. 이 함수는 기본 60초를 넘은 기간 동안 결과를 적용하여 한도를 초과한 연결을 축소합니다. MeterMaid에서도 MeterMaid 서버 시스템에 다음 옵션을 구성하여 이와 동일한 동작을 사용할 수 있습니다.

```
configutil -o metermaid.table.ext_throttle.options -v penalize
```

그러면 metermaid.table.ext_throttle.quota에 설정된 값을 초과하는 연결 시도에 대해 연결을 축소할 수 있도록 ext_throttle 테이블의 동작이 변경됩니다.

또 다른 옵션은 많은 수의 연결을 수신하는 시스템과 관련 있습니다. MeterMaid는 분산 MTA 환경 전반에 걸쳐 연결 추적이 가능하므로, MeterMaid의 내부 메모리 내장 데이터베이스에 보존되는 연결 수 제한이 MTA 환경의 전체 볼륨에 충분하지 않을 수 있습니다. 기본값은 테이블당 1000개 항목이지만, MTA 환경 전체에 걸쳐 분당 연결 1000개를 초과할 것으로 예상된다면 다음 구성 옵션을 통해 이 값을 늘릴 수 있습니다.

```
configutil -o metermaid.table.ext_throttle.max_entries -v max_entries
```

60초 동안 *max_entries*에 도달하더라도 MeterMaid는 가장 오래되고 덜 쓰인 항목을 자동 삭제합니다. 따라서 더 자주 연결되는 시스템은 MeterMaid 테이블에 남아 계산에 포함되므로 효과적인 억제를 수행하기에 충분한 정보가 유지됩니다.

◆ ◆ ◆ 20장

20 메시지 저장소 관리

이 장에서는 메시지 저장소와 메시지 저장소의 관리 인터페이스에 대해 설명합니다. 이 장은 다음 내용으로 구성되어 있습니다.

20.1 개요

메시지 저장소는 특정 Messaging Server 인스턴스에 대한 사용자 메일함을 포함합니다. 메일함, 폴더 및 로그 파일 수가 늘어나면 메시지 저장소의 크기가 늘어납니다. 메일함의 크기(디스크 할당량)를 제한하고 허용되는 총 메시지 수의 한도를 지정하며 저장소의 메시지에 대한 에이징 정책을 설정하여 저장소 크기를 제어할 수 있습니다.

시스템에 다른 사용자를 추가하면 디스크 저장소 요구 사항이 증가합니다. 서버가 지원하는 사용자 수에 따라 메시지 저장소는 하나 또는 여러 개의 물리적 디스크가 필요할 수 있습니다. 이러한 추가 디스크 공간을 시스템에 통합하는 방법에는 두 가지가 있습니다. 가장 쉬운 방법은 메시지 저장소 분할 영역을 추가하는 것입니다(602 페이지 “20.10 메시지 저장소 분할 영역 구성” 참조).

마찬가지로 여러 호스트된 도메인을 지원하는 경우 하나의 큰 도메인에서 서버 인스턴스를 전담하도록 할 수 있습니다. 이 구성을 사용하면 특정 도메인에 대한 저장소 관리자를 지정할 수 있습니다. 또한 다른 분할 영역을 추가하여 메시지 저장소를 확장할 수 있습니다.

Messaging Server는 메시지 저장소의 관리를 위해 표 20–1에 설명된 일련의 명령줄 유틸리티를 제공합니다. 이러한 유틸리티 사용에 대한 자세한 내용은 605 페이지 “20.11 메시지 저장소 유지 관리 절차 수행” 및 **Sun Java System Messaging Server 6.3 Administration Reference**를 참조하십시오.

표 20–1 메시지 저장소 명령줄 유틸리티

유틸리티	설명
`configutil`	저장소의 구성 매개 변수를 설정 및 수정합니다.
`deliver`	메일을 IMAP 또는 POP 메일 클라이언트가 액세스할 수 있는 메시지 저장소로 직접 전달합니다.
`hashdir`	특정 사용자의 메시지 저장소를 포함하는 디렉토리를 식별합니다.
`imsconnutil`	메시지 저장소의 사용자 액세스를 모니터합니다.
`imexpire`	관리자가 지정한 기준(예: 기간)에 따라 메시지 저장소에서 메시지를 자동으로 제거합니다.
`iminitquota`	LDAP 디렉토리에서 할당량 제한을 다시 초기화하고 사용 중인 디스크 공간을 다시 계산합니다.
`imsasm`	사용자 메일함의 저장과 복구를 처리합니다.
`imsbackup`	저장된 메시지를 백업합니다.
`imsexport`	Messaging Server 메일함을 UNIX `/var/mail` 형식 폴더로 내보냅니다.
`imsrestore`	백업된 메시지를 복원합니다.
`imscripter`	IMAP 서버 프로토콜 스크립트 도구입니다. 명령이나 명령 시퀀스를 실행합니다.
`mboxutil`	메일함을 나열, 작성, 삭제, 이름 변경 또는 이동을 수행하고 할당량 사용을 보고합니다.
`mkbackupdir`	백업 디렉토리를 만들어 메시지 저장소의 정보와 동기화합니다.
`MoveUser`	한 메시징 서버에서 다른 메시징 서버로 사용자의 계정을 이동합니다.
`imquotacheck`	메시지 저장소의 각 사용자에 대한 총 메일함 크기를 계산하고 이 크기를 지정된 할당량과 비교합니다. 현지화된 버전의 `imquotacheck` 알림은 % 및 $ 기호를 잘못 변환합니다. 인코딩을 수정하려면 메시지 파일에서 모든 $와 %를 각각 \24와 \25로 바꿉니다.
`readership`	공유 IMAP 폴더에서 readership 정보를 수집합니다.

표 20–1 메시지 저장소 명령줄 유틸리티 (계속)

유틸리티	설명
reconstruct	손상된 메일함을 다시 구성합니다.
stored	백그라운드 및 일상 작업을 수행하고 디스크에 저장된 메시지를 정리 및 삭제합니다.

20.2 메시지 저장소 디렉토리 레이아웃

그림 20–1에는 서버 인스턴스의 메시지 저장소 디렉토리 레이아웃이 나와 있습니다. 메시지 저장소는 메일함 내용을 신속하게 액세스할 수 있도록 설계되었습니다. 저장소 디렉토리는 표 20–2에 설명되어 있습니다.

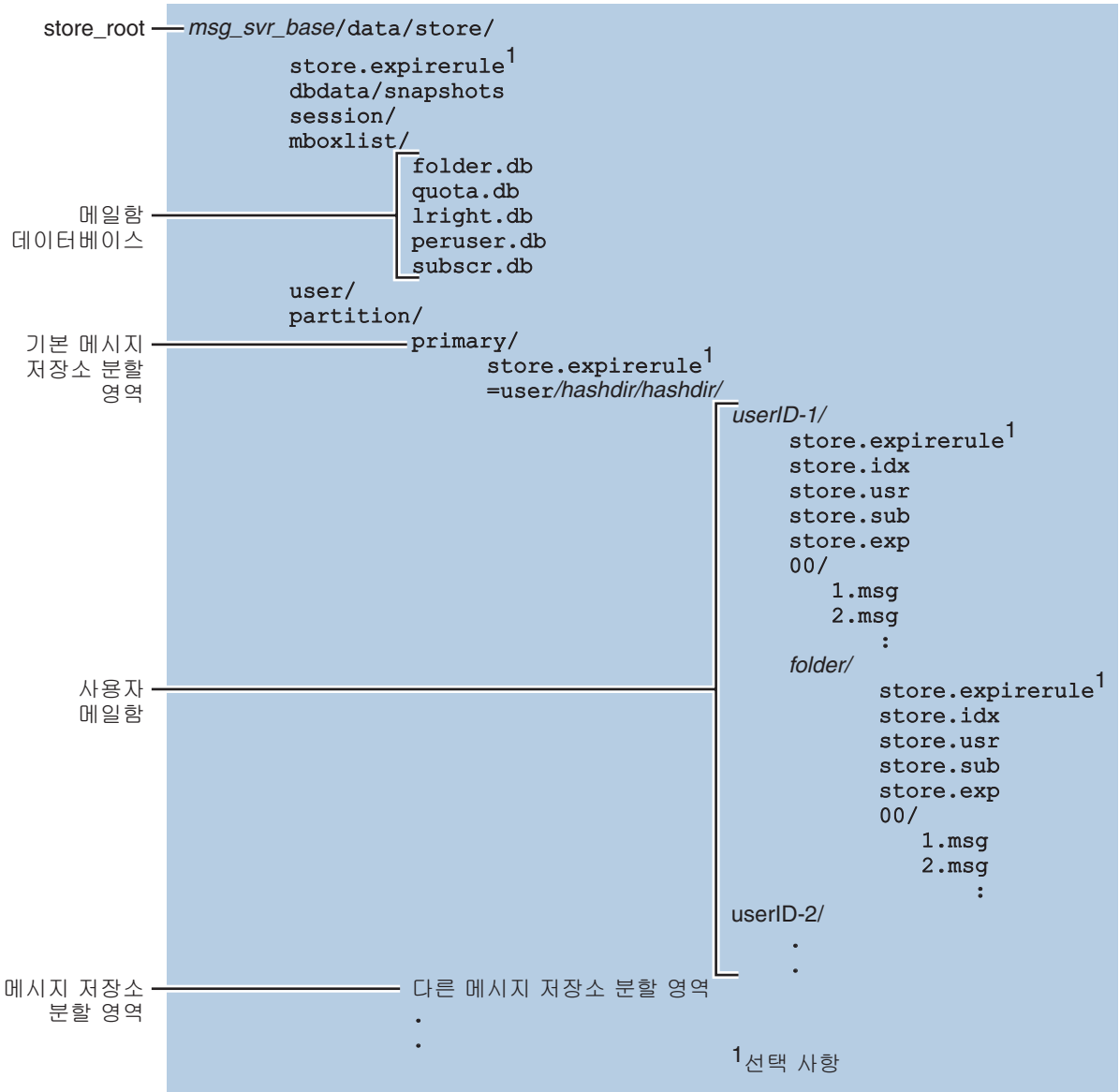


그림 20–1 메시지 저장소 디렉토리 레이아웃

메시지 저장소는 여러 메일함 데이터베이스 및 사용자 메일함으로 구성됩니다. 메일함 데이터베이스는 사용자, 메일함, 분할 영역, 할당량 및 기타 메시지 저장소 관련 데이터에 대한 정보로 구성됩니다. 사용자 메일함은 사용자의 메시지와 폴더를 포함합니다. 메일함은 전적으로 메시지 저장소를 저장하는 **디스크** 분할 영역의 한 영역인 **메시지 저장소 분할 영역**에 저장됩니다. 자세한 내용은 602 페이지 “20.10 메시지 저장소 분할

영역 구성"을 참조하십시오. 메시지 저장소 분할 영역은 디스크 분할 영역과 다르지만 유지 관리가 용이하도록 각 메시지 저장소 분할 영역에 대해 하나의 디스크 분할 영역을 가지는 것이 좋습니다.

INBOX와 같은 메일함은 *store_root*에 위치합니다. 예를 들어, 샘플 디렉토리 경로는 다음과 같을 수 있습니다.

store_root/partition/primary/=user/53/53/=mack1

아래 표에서는 메시지 저장소 디렉토리를 설명합니다.

표 20–2 메시지 저장소 디렉토리 설명

위치	내용/설명
msg-svr-base	기본값: /opt/SUNWmsgsr 서버 프로그램, 구성, 유지 관리 및 정보 파일을 포함하는 Messaging Server 시스템상의 디렉토리입니다.
store_root	*msg-svr-base*/data/store 메시지 저장소의 최상위 디렉토리입니다. mboxlist, user 및 partition 하위 디렉토리를 포함합니다.
./store.expirerule	자동 메시지 제거 규칙(만료 규칙)을 포함합니다. 이 선택적 파일의 위치는 다를 수 있습니다. 592 페이지 "20.9 자동 메시지 제거(만료 및 제거) 기능 설정 방법"을 참조하십시오.
store_root/dbdata/snapshots	stored가 정기적으로 만드는 메시지 저장소 데이터베이스 백업 스냅샷입니다.
store_root/mboxlist/	메일함 및 할당량 관련 정보를 저장하는 메일함 데이터베이스(Berkeley DB)를 포함합니다. folder.db는 메일함이 저장된 분할 영역의 이름, ACL, store.idx의 일부 정보 복사본 등을 비롯하여 메일함에 대한 정보를 포함합니다. folder.db에는 각 메일함별로 한 개의 항목이 있습니다. quota.db는 할당량 및 할당량 사용에 대한 정보를 포함합니다. quota.db에는 각 사용자별로 한 개의 항목이 있습니다. lright.db는 acl 조회 권한별 폴더에 대한 색인입니다. peruser.db는 사용자별 플래그에 대한 정보를 포함합니다. 이 플래그는 특정 사용자가 메시지를 보았거나 삭제했는지 여부를 나타냅니다. subscr.db는 사용자 가입에 대한 정보를 포함합니다.
store_root/session/	활성 메시지 저장소 프로세스 정보를 포함합니다.
store_root/user/	사용되지 않습니다.

표 20–2 메시지 저장소 디렉토리 설명 *(계속)*

위치	내용/설명
store_root/partition/	메시지 저장소 분할 영역을 포함합니다. 기본 primary 분할 영역이 만들어집니다. 정의하는 다른 모든 분할 영역을 이 디렉토리에 넣습니다.
store_root/partition/primary/=user/	분할 영역의 하위 디렉토리에 모든 사용자 메일함을 포함합니다. 메일함은 빠른 검색을 위해 해시 구조에 저장됩니다. 특정 사용자의 메일함을 포함하는 디렉토리를 찾으려면 hashdir 유틸리티를 사용합니다.
.../=user/*hashdir*/ *hashdir*/*userid* /	아이디가 *userid*인 사용자에 대한 최상위 메일 폴더이며,사용자의 받은 메일함입니다. 기본 도메인의 경우 *userid*는 *uid*이고 호스트된 도메인의 경우 *userid*는 *uid@domain*입니다. 받는 메시지는 이 메일 폴더로 전달됩니다.
.../*userid*/*folder*	Messaging Server의 사용자 정의 폴더입니다.
.../*userid*/store.idx	/*userid*/ 디렉토리에 저장된 메일에 대해메시지 수, 이 메일함에 사용된 디스크 할당량, 메일함이 마지막으로 추가된 시간, 메시지 플래그, 헤더 및 MIME 구조를 비롯한 각 메시지의 변수 길이 정보, 각 메시지의 크기 등과 같은 정보를 제공하는 색인입니다. 이 색인은 또한 각 사용자에 대한 mboxlist 정보와 할당량 정보의 백업 복사본을 포함합니다.
.../*userid*/store.usr	폴더에 액세스한 사용자 목록을 포함합니다. 목록의 각 사용자에 대해 사용자가 폴더에 액세스한 마지막 시간, 사용자가 본 메시지 목록 및 사용자가 삭제한 메시지 목록에 대한 정보를 포함합니다.
.../*userid*/store.sub	사용자 가입에 대한 정보를 포함합니다.
.../*userid*/store.exp	정리되었지만 디스크에서 제거되지는 않은 메시지 파일의 목록을 포함합니다. 이 파일은 정리된 메시지가 있는 경우에만 나타납니다.
.../*userid*/*nn*/ or .../*userid*/*folder*/*nn*/	*nn*은 *message_id*.msg 형식의 메시지를 포함하는 해시 디렉토리입니다. *nn*은 00에서 99 사이의 숫자가 될 수 있으며 *message_id*도 숫자입니다. 예: 1에서 99 사이의 메시지는 .../00 디렉토리에 저장됩니다. 첫 번째 메시지는 1.msg이고 두 번째 메시지는 2.msg, 세 번째 메시지는 3.msg입니다. 100에서 199 사이의 메시지는 01 디렉토리에 저장되고 9990에서 9999 사이의 메시지는 99 디렉토리에 저장되며 이와 같이 10000에서 10099 사이의 메시지는 00 디렉토리에 저장됩니다.

20.2.1 유효한 폴더 이름 및 유효하지 않은 폴더 이름

다음은 유효한 IMAP 폴더 문자와 유효하지 않은 IMAP 폴더 문자입니다.

유효한 IMAP 폴더 문자: <space> ! " # $ & ' ( ) + , - . / 0-9 : ; < = > @ A-Z [\ ] ^ _ ‘ a-z { | } ~

유효하지 않은 IMAP 폴더 문자: % * ?

public/이라는 폴더와 같이 특정 문자 순서가 거부될 수도 있습니다. 또, 이 제한은 영어 로켈을 사용하는 경우에 적용됩니다. 일본어 등의 다른 언어에서는 인코딩된 문자 세트를 사용합니다.

20.3 메시지 저장소에서 메시지를 제거하는 방법

메시지는 메시지 저장소에서 다음 세 단계를 거쳐 제거됩니다.

1. **삭제**. 클라이언트가 메시지 플래그를 **삭제**로 설정합니다. 이 시점에 메시지가 제거 표시되지만 클라이언트는 삭제 플래그를 제거하여 메시지를 복원할 수 있습니다. 두 번째 클라이언트가 있을 경우 삭제된 플래그는 바로 두 번째 클라이언트부터 표시되지 않을 수 있습니다. `configutil` 매개 변수 `local.imap.immediateflagupdate`를 설정하여 즉시 플래그 업데이트를 사용할 수 있습니다.
2. **정리**. 메일함에서 메시지가 제거됩니다. 기술적으로는 메시지가 메시지 저장소 색인 파일 `store.idx`에서 제거되는 것입니다. 메시지 자체는 여전히 디스크상에 존재하지만 메시지가 정리되고 나면 클라이언트가 더 이상 메시지를 복원할 수 없습니다.

   만료는 특수한 경우의 정리를 의미합니다. 메시지 크기, 기간 등과 같은 관리자가 정의한 일련의 제거 기준을 따르는 메시지가 정리됩니다. 592 페이지 “20.9 자동 메시지 제거(만료 및 제거) 기능 설정 방법”을 참조하십시오.
3. **제거**. `imexpire` 유틸리티는 기본적으로 매일 오후 11시에 정리된 메시지를 디스크에서 제거합니다. 이는 메시지 만료 일정을 제어하는 `local.schedule.expire`와 제거 유예 기간(그 전까지 메시지가 제거되지 않는 기간)을 제어하는 `store.cleanupage`로 구성할 수 있습니다. 오래된 버전의 MTA 로그 파일을 정리하는 `imsimta purge` 명령 및 `configutil` 매개 변수 `local.schedule.purge`와는 다릅니다.

20.4 저장소에 대한 관리자 액세스 지정

메시지 저장소 관리자는 사용자 메일함을 보고 모니터링하며 메시지 저장소에 대한 액세스 제어를 지정할 수 있습니다. 저장소 관리자는 모든 서비스(POP, IMAP, HTTP 또는 SMTP)에 대한 프록시 인증 권한을 가지므로 모든 사용자의 권한을 사용하여 모든 서비스에 인증될 수 있습니다. 이러한 권한을 사용하여 저장소 관리자는 저장소 관리를 위한 일정한 유틸리티를 실행할 수 있습니다. 예를 들어, 저장소 관리자는 `MoveUser`를 사용하여 사용자 계정과 메일함을 특정 시스템에서 다른 시스템으로 이동할 수 있습니다.

이 절에서는 Messaging Server 설치의 메시지 저장소에 대한 저장소 권한을 허가하는 방법에 대해 설명합니다.

주 – 다른 사용자가 저장소에 대한 관리자 권한을 가질 수도 있습니다. 예를 들어, 일부 관리자가 이러한 권한을 가질 수 있습니다.

다음 하위 절에 설명된 대로 관리자 작업을 수행할 수 있습니다.

- 562 페이지 "관리자 항목 추가 방법"
- 562 페이지 "관리자 항목 수정"
- 562 페이지 "관리자 항목 삭제"
- 563 페이지 "20.4.1 관리자에 의한 경우를 제외하고 메일함 삭제 또는 이름 바꾸기 차단"

▼ 관리자 항목 추가 방법

- **명령줄: 명령줄에서 관리자 항목을 추가하려면 다음을 수행합니다.**

  ```
  configutil -o store.admins -v "adminlist"
  ```

  여기서 *adminlist*는 공백으로 구분된 관리자 아이디 목록입니다. 여러 관리자를 지정할 경우 목록을 따옴표로 묶어야 합니다. 또한, 관리자는 서비스 관리자 그룹의 구성원이어야 합니다(LDAP 사용자 항목에서 `memberOf: cn=Service Administrators,ou=Groups,o=usergroup`).

▼ 관리자 항목 수정

- **명령줄. 명령줄에서 메시지 저장소 관리자 UID 목록의 기존 항목을 수정하려면 다음을 수행합니다.**

  ```
  configutil -o store.admins -v "adminlist"
  ```

  여기서 *adminlist*는 공백으로 구분된 관리자 아이디 목록입니다. 여러 관리자를 지정할 경우 목록을 따옴표로 묶어야 합니다. 또한, 관리자는 서비스 관리자 그룹의 구성원이어야 합니다(LDAP 사용자 항목에서 `memberOf: cn=Service Administrators,ou=Groups,o=usergroup`).

▼ 관리자 항목 삭제

- **명령줄. 명령줄에서 저장소 관리자를 삭제하려면 다음과 같이 관리자 목록을 편집할 수 있습니다.**

  ```
  configutil -o store.admins -v "adminlist"
  ```

여기서 *adminlist*는 공백으로 구분된 관리자 아이디 목록입니다. 여러 관리자를 지정할 경우 목록을 따옴표로 묶어야 합니다. 또한, 관리자는 서비스 관리자 그룹의 구성원이어야 합니다(LDAP 사용자 항목에서 `memberOf: cn=Service Administrators,ou=Groups,o=usergroup`).

20.4.1 관리자에 의한 경우를 제외하고 메일함 삭제 또는 이름 바꾸기 차단

관리자에 의한 경우를 제외하고 일부 메일함을 삭제하거나 수정하는 것을 차단할 수 있습니다. 다음 절차에서 이 작업 방법에 대해 설명합니다. 관리자가 아닌 사람이 보호된 메일함을 삭제 또는 수정하거나 이름을 바꾸려고 하면 `mailbox is pinned`라는 오류 메시지가 표시됩니다.

다음 형식을 사용하여 `local.store.pin configutil` 변수를 설정합니다.

```
configutil -o local.store.pin -v "mailbox1"%"mailbox2"%"mailbox 3"
```

여기서 *mailbox1*, *mailbox2* 및 *mailbox 3*은 보호할 메일함이고(메일함 이름에 공백을 사용할 수 있음) `%`는 각 메일함 사이의 구분자입니다.

20.5 공유 폴더 정보

그룹 또는 **공유 폴더**는 지정된 사용 권한에 따라 다른 사용자와 그룹이 읽고 삭제하며 메시지를 추가할 수 있다는 점을 제외하고 다른 모든 메일 폴더와 같습니다. 일반적인 끌어서 놓기, 시브(Sieve) 필터 또는 다음 형식으로 직접 메시지를 보내 공유 폴더에 메시지를 추가할 수 있습니다. *uid+folder@domain*.

아래의 예는 carol.fanning@siroe.com에서 소유한 **개인 공유 폴더**, `crafts_club`으로 전자 메일을 보내기 위한 주소입니다.

```
carol.fanning+crafts_club@siroe.com
```

다음 예는 `tennis@siroe.com`이라는 **공개 공유 폴더**로 전자 메일을 보내기 위한 주소입니다.

```
public+tennis@siroe.com
```

공유 폴더는 특정 주제에 대한 지속적인 전자 메일 대화를 시작, 공유 및 보관하는 데 유용합니다. 예를 들어, 한 그룹의 소프트웨어 개발자가 특정 프로젝트의 개발을 논의하기 위해 `mosaic_voices`라는 공유 폴더를 만들 수 있습니다. 메시지를 보내거나 `mosaic_voices` 폴더에 넣으면 공유 폴더에 대한 사용 권한이 있는 모든 사용자(개별 주소나 그룹 주소로 사용 권한을 추가할 수 있음)가 이 메일함을 열고 메시지를 읽을 수 있습니다.

공유 폴더는 사용자의 메일함에서 Shared Folders라는 폴더 아래에 표시됩니다. 예를 들면 다음과 같습니다.

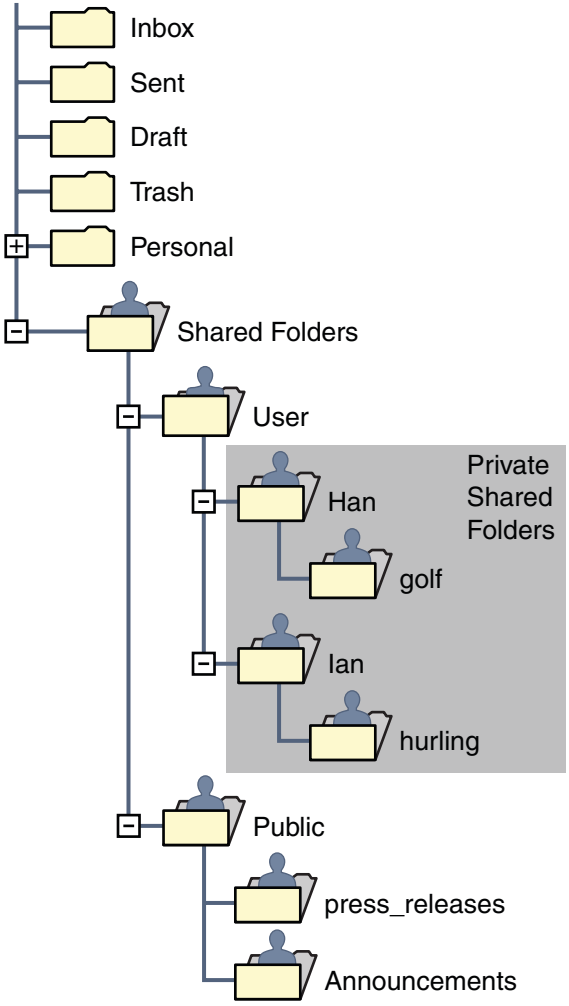


그림 20–2 메일 클라이언트에서 본 공유 메일 폴더 목록의 예

공유 폴더에는 다음 두 종류가 있습니다.

- **개인 공유 폴더** – Communications Express 또는 공유 폴더 만들기를 지원하는 다른 메일 클라이언트를 사용하여 특정 사용자가 만들고 소유하는 공유 폴더입니다. 자세한 내용은 Communications Express 도움말 화면을 참조하십시오. 개인 공유 폴더는 Shared Folders/User 메일 폴더 디렉토리에 있습니다.
- **공개 공유 폴더** – 공개 공유 폴더는 메일 관리자가 만들며 소유자가 없습니다. 공개 폴더의 전자 메일 주소는 다음과 같습니다.

`public+`*foldername*`@`*domain*

예를 들어, 회사 내의 특정 이익 집단에 대한 정보를 게시하기 위해 `public+software_dev@siroe.com`과 같은 폴더를 필요로 할 수 있습니다. 관련 직원에게 이 공개 폴더에 대한 액세스 권한이 부여됩니다. 공개 공유 폴더는 `Shared Folders/Public` 메일 폴더 디렉토리에 있습니다.

일반적으로 공유 폴더는 특정 메시지 저장소의 사용자만 사용할 수 있습니다. 그러나 Messaging Server에서는 여러 메시지 저장소에서 액세스할 수 있는 특수한 공유 폴더를 만들 수 있습니다. 이러한 폴더를 **분산 공유 폴더**라고 부릅니다. 자세한 내용은 570 페이지 "20.6.4 분산 공유 폴더 설정"을 참조하십시오.

20.6 공유 폴더 작업

이 절에서는 공유 폴더와 관련된 다음 관리자 작업에 대해 설명합니다.

▼ 개인 공유 폴더의 공유 속성 지정

1 개인 공유 폴더는 사용자가 만듭니다.

많은 메일 클라이언트에서 개인 공유 폴더 만들기를 지원합니다. Communications Express에서 시도해 볼 수 있습니다.

2 개인 공유 폴더의 공유 매개 변수를 설정합니다.

다음 `configutil` 매개 변수가 지원됩니다.

`store.privatesharedfolders.restrictanyone` - 활성화되면(1) 일반 사용자는 개인 공유 폴더에 대한 권한을 `anyone`으로 설정할 수 없습니다. 기본값: 0

`store.privatesharedfolders.restrictdomain` - 활성화되면(1) 일반 사용자는 개인 폴더를 도메인 외부 사용자와 공유할 수 없습니다. 기본값: 0

`store.privatesharedfolders.shareflags` - 0이면 여러 사용자들 간에 플래그를 공유할 수 없습니다. 1이면 여러 사용자들 간에 플래그를 공유할 수 있습니다. 기본값: 0

`store.publicsharedfolders.user` - 공개 공유 폴더 소유자의 사용자 아이디입니다. 보통 `public`입니다. 기본값: NULL(설정되지 않음)

▼ 공개 공유 폴더 만들기

공개 폴더는 LDAP 데이터베이스와 `readership` 명령에 대한 액세스가 필요하기 때문에 시스템 관리자가 만들어야 합니다.

1 **모든 공개 폴더의 컨테이너 역할을 수행하는 `public`이라는 LDAP 사용자 항목을 추가합니다(563 페이지 "20.5 공유 폴더 정보" 참조).**

예:

```
dn: cn=public,ou=people,o=sesta.com,o=ISP
objectClass: person
objectClass: organizationalPerson
objectClass: inetOrgPerson
objectClass: inetUser
objectClass: ipUser
objectClass: inetMailUser
objectClass: inetLocalMailRecipient
objectClass: nsManagedPerson
objectClass: userPresenceProfile
cn: public
mail: public@sesta.com
mailDeliveryOption: mailbox
mailHost: manatee.siroe.com
uid: public
inetUserStatus: active
mailUserStatus: active
mailQuota: -1
mailMsgQuota: 100
```

2 **`mboxutil` 명령줄 유틸리티를 사용하여 `public` 계정 내에 폴더를 만듭니다.**

예를 들어, gardening이라는 공개 폴더를 만듭니다.

```
mboxutil -c user/public/gardening
```

3 **폴더의 이름을 설정합니다.**

보통 `public`으로 지정합니다. 다음은 폴더 이름을 `public`으로 설정하는 명령입니다.

```
configutil -o store.publicsharedfolders.user –v public
```

4 **사용자와 공유 폴더에 대한 액세스 권한을 지정합니다.**

`readership` 명령을 사용하여 사용자와 해당 액세스 권한을 지정합니다. 예를 들어, 다음 명령은 `gardening` 공개 폴더에 대한 조회, 읽기 및 게시 권한을 `sesta.com`의 모든 사용자에게 할당합니다.

```
readership -s user/public/gardening anyone@sesta.com lrp
```

readership 사용 방법에 대한 자세한 내용은 568 페이지 "20.6.2 공유 폴더의 액세스 제어 권한 설정 또는 변경"을 참조하십시오.

20.6.1 전자 메일 그룹을 사용하여 공유 폴더 추가

공유 폴더는 일반적으로 Communications Express를 사용하여 공유 폴더 목록에 사용자를 추가하거나 앞에서 설명한 것처럼 공개 공유 폴더를 만들어 작성합니다. 그러나 그룹에 속한 모든 사용자가 공유 폴더에 액세스할 수 있도록 공유 폴더 목록에 전자 메일 그룹(메일 배포 목록)을 추가할 수도 있습니다. 예를 들어, `tennis@sesta.com`이라는 그룹에 25명의 구성원이 있으며 공유 폴더를 만들어 이 그룹 주소로 전송된 모든 전자 메일을 저장하기로 결정했습니다.

▼ 공유 폴더에 전자 메일 그룹을 추가하는 방법

공유 폴더에 전자 메일 그룹을 추가하려면 시스템 관리자 권한이 필요합니다.

1. **폴더를 만듭니다.(이미 폴더를 만든 경우에는 이 단계를 건너뜁니다.)**

   일반적으로 이 작업은 그룹의 구성원 중 한 명이 수행해야 합니다. 그렇지 않으면 다음 명령을 사용하여 대신 폴더를 만들 수 있습니다.

   ```
   mboxutil -c user/gregk/gardening
   ```

   `gregk`는 공유 폴더 소유자의 `uid`입니다. `gardening`은 공유 폴더의 이름입니다.

2. **이 그룹 공유 폴더를 액세스할 모든 구성원의 사용자 항목에 속성-값 쌍인 `aclGroupAddr` *group_name@domain*을 추가합니다.**

   위의 예를 사용하면 공유 폴더에 대한 액세스 권한을 할당할 각 사용자 항목에 다음 속성-값 쌍을 추가합니다.

   ```
   aclGroupAddr: tennis@sesta.com
   ```

   그룹 항목에 `memberURL` 속성을 사용하여 동적으로 그룹을 만든 경우에는 해당 구성원에 이미 이 속성이 있습니다. 이 속성의 URL 값은 다음과 같습니다.

   ```
   memberURL: ldap:///o=sesta.com??sub?(&(aclGroupAddr=tennis@sesta.com)
   (objectclass=inetmailuser))
   ```

   (인쇄상의 이유로 샘플 항목에서는 줄 바꿈되었지만 실제 항목은 한 행에 표시됩니다.)

3. **그룹과 공유 폴더에 대한 액세스 권한을 지정합니다.**

   `readership` 명령을 사용하여 이 작업을 수행할 수 있습니다. 위의 예를 사용하면 다음 명령은 `gardening` 공개 폴더에 대한 조회, 읽기 및 게시 권한을 tennis@sesta.com의 구성원에게 할당합니다.

   ```
   readership -s user/gregk/tennis tennis@sesta.com lrp
   ```

   readership 사용 방법에 대한 자세한 내용은 568 페이지 "20.6.2 공유 폴더의 액세스 제어 권한 설정 또는 변경"을 참조하십시오.

20.6.2 공유 폴더의 액세스 제어 권한 설정 또는 변경

사용자는 Communications Express 인터페이스를 사용하여 공유 폴더에 대한 액세스 제어를 설정하거나 변경할 수 있습니다. 관리자는 `readership` 명령줄 유틸리티를 사용하여 공유 폴더에 대한 액세스 제어를 설정하거나 변경할 수 있습니다. 이 명령의 형식은 다음과 같습니다.

`readership -s` *`foldername identifier rights_chars`*

여기서 *foldername*은 권한을 설정할 공개 폴더의 이름이고 *identifier*는 권한을 할당하는 개인 또는 그룹이며 *rights_chars*는 할당할 권한입니다. 각 문자의 의미는 표 20–3을 참조하십시오.

주 – anyone은 특수 식별자입니다. anyone에 대한 액세스 권한은 모든 사용자에게 적용됩니다. 마찬가지로 anyone@*domain*에 대한 액세스 권한은 동일한 도메인의 모든 사용자에게 적용됩니다.

표 20–3 ACL 권한 문자

문자	설명
l	lookup– 사용자가 공유 폴더를 보고 가입할 수 있습니다. (허용되는 IMAP 명령: LIST 및 LSUB).
r	read– 사용자가 공유 폴더를 읽을 수 있습니다(허용되는 IMAP 명령: 폴더에서 SELECT, CHECK, FETCH, PARTIAL, SEARCH, COPY).
s	seen– 세션에서 사용자가 본 정보를 보관하도록 시스템에 지시합니다. (IMAP STORE SEEN 플래그 설정).
w	write– 사용자가 read로 표시하고 메시지를 삭제할 수 있습니다. (SEEN 및 DELETED가 아닌 IMAP STORE 플래그 설정).
i	insert– 사용자가 한 폴더에서 다른 폴더로 전자 메일을 복사하고 이동할 수 있습니다. (허용되는 IMAP 명령: 폴더로 APPEND, COPY).
p	post– 사용자가 공유 폴더 전자 메일 주소로 메일을 보낼 수 있습니다. (IMAP 명령이 필요하지 않음).
c	create– 사용자가 새 하위 폴더를 만들 수 있습니다. (허용되는 IMAP 명령: CREATE).
d	delete– 사용자가 공유 폴더에서 항목을 삭제할 수 있습니다. (허용되는 IMAP 명령: EXPUNGE, STORE DELETED 플래그 설정)
a	administer– 사용자에게 관리 권한이 있습니다. (허용되는 IMAP 명령: SETACL).

20.6.2.1 예

sesta 도메인에 있는 모든 사용자에게 golftournament라는 공개 폴더에 대한 조회, 읽기 및 전자 메일 표시 권한(게시 권한 제외)을 주려면 다음 명령을 실행합니다.

```
readership -s User/public/golftournament anyone@sesta lwr
```

메시지 저장소에 있는 모든 사용자에게 동일한 액세스 권한을 할당하려면 다음 명령을 실행합니다.

```
readership -s User/public/golftournament anyone lwr
```

조회, 읽기, 전자 메일 표시 및 게시 권한을 그룹에 할당하려면 다음 명령을 실행합니다.

```
readership -s User/public/golftournament group=golf@sesta.com lwrp
```

이 폴더에 대한 관리자 및 게시 권한을 개별 사용자 jdoe에게 할당하려는 경우 다음 명령을 실행합니다.

```
readership -s User/public/golftournament jdoe@sesta.com lwrpa
```

공개 폴더에 대한 개별 사용자 또는 그룹 액세스를 거부하려면 userid에 접두어 대시를 사용합니다. 예를 들어, jsmith에 대한 조회, 읽기 및 쓰기 권한을 거부하려면 다음 명령을 실행합니다.

```
readership -s User/public/golftournament -jsmith@sesta.com lwr
```

개별 사용자 또는 그룹 액세스를 거부하려면 ACL 권한 문자에 접두어 대시를 사용합니다. 예를 들어, jsmith에 대한 게시 권한을 거부하려면 다음 명령을 실행합니다.

```
readership -s User/public/golftournament jsmith@sesta.com -p
```

주 – *uid+folder@domain* 주소를 사용하여 공유 폴더에 메시지를 게시하려면 readership 명령과 함께 p(post) 액세스 권한을 사용해야 합니다. 568 페이지 “20.6.2 공유 폴더의 액세스 제어 권한 설정 또는 변경”을 참조하십시오.

20.6.3 공유 폴더 목록을 사용 가능 또는 사용 불가능하게 하기

서버는 구성 옵션 local.store.sharedfolders의 설정에 따라 LIST 명령에 응답할 때 공유 폴더를 반환하거나 반환하지 않습니다. 이 옵션을 비활성화하려면 off로 설정합니다. 기본적으로 이 옵션은 활성화됩니다(on으로 설정).

SELECT 및 LSUB 명령은 이 옵션의 영향을 받지 않습니다. LSUB 명령은 공유 폴더를 비롯하여 가입한 모든 폴더를 반환합니다. 사용자는 자신이 소유하거나 가입한 공유 폴더를 SELECT할 수 있습니다.

20.6.4 분산 공유 폴더 설정

일반적으로 공유 폴더는 특정 메시지 저장소의 사용자만 사용할 수 있습니다. 그러나 Messaging Server에서는 여러 메시지 저장소에서 액세스할 수 있는 **분산 공유 폴더**를 만들 수 있습니다. 즉, 분산 공유 폴더에 대한 액세스 권한은 메시지 저장소 그룹 내의 모든 사용자에게 부여될 수 있습니다. 단, 웹 메일 클라이언트(Messenger Express와 같은 HTTP 액세스 클라이언트)는 원격 공유 폴더 액세스를 지원하지 않습니다. 사용자는 폴더를 나열하여 가입할 수 있지만 내용을 보거나 변경할 수는 없습니다.

분산 공유 폴더는 다음이 필요합니다.

- 메시지 저장소 `userid`가 메시지 저장소 그룹에서 고유해야 합니다.
- 배포에서 디렉토리 데이터가 동일해야 합니다.

원격 메시지 저장소(공유 폴더를 보유하지 않는 메시지 저장소)는 표 20–4에 나열된 구성 변수를 설정하여 프록시 서버로 구성해야 합니다.

표 20–4 분산 공유 폴더 구성을 위한 변수

이름	값	데이터 형식
`local.service.proxy.serverlist`	메시지 저장소 서버 목록	공백으로 구분된 문자열
`local.service.proxy.admin`	기본 저장소 관리자 로그인 이름	문자열
`local.service.proxy.adminpass`	기본 저장소 관리자 비밀번호	문자열
`local.service.proxy.admin.`*hostname*	특정 호스트의 저장소 관리자 로그인 이름	문자열
`local.service.proxy.adminpass.`*hostname*	특정 호스트의 저장소 관리자 비밀번호	문자열

20.6.4.1 분산 공유 폴더 설정—예

그림 20–3은 StoreServer1, StoreServer2 및 StoreServer3이라는 세 메시지 저장소 서버의 분산 폴더 예를 보여 줍니다.

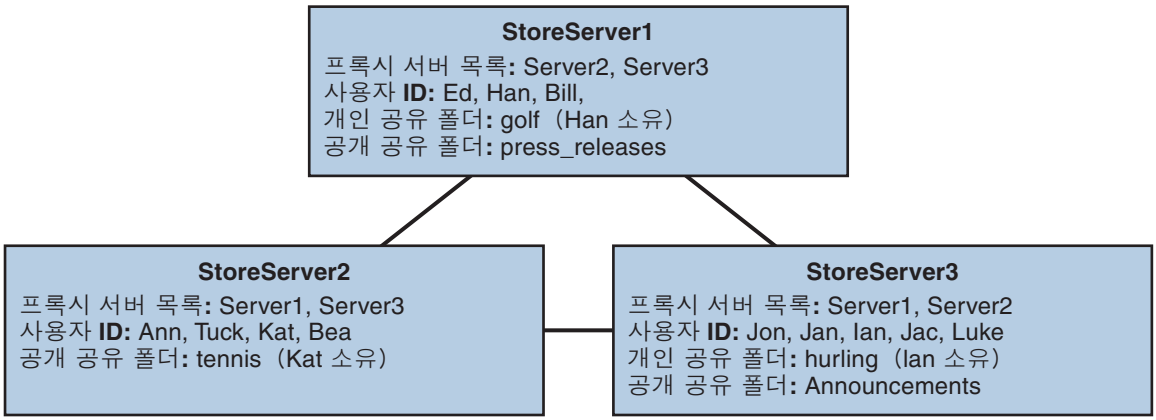


그림 20–3 분산 공유 폴더—예

이러한 서버는 표 20–4에 나온 변수의 설정을 통해 서로 간에 피어 프록시 메시지 저장소로 연결됩니다. 각 서버에는 개인 공유 폴더 *golf*(Han 소유), *tennis*(Kat 소유) 및 *hurling*(Luke 소유)이 있습니다. 또한 *press_releases* 및 *Announcements*라는 두 개의 공개 공유 폴더가 존재합니다. 세 서버의 사용자는 이러한 세 개의 공유 폴더에 액세스할 수 있습니다. 그림 20–2는 Ed의 공유 폴더 목록을 보여 줍니다. 다음은 이 구성의 각 서버에 대한 ACL의 예입니다.

```
$ StoreServer1 :> imcheck -d lright.db
Ed: user/Han/golf
Ian: user/Han/golf
anyone: user/public/press_releases
```

```
$ StoreServer2 :> imcheck -d lright.db
Jan: user/Kat/tennis
Ann: user/Kat/tennis
anyone: user/public+Announcements user/public+press_releases
```

```
$ StoreServer3 :> imcheck -d lright.db
Tuck: user/Ian/hurling
Ed: user/Ian/hurling
Jac: user/Ian/hurling
anyone: user/public/Announcements
```

20.6.5 공유 폴더 데이터 모니터 및 유지 관리

readership 명령줄 유틸리티를 사용하면 folder.db, peruser.db 및 lright.db 파일에 보관되는 공유 폴더 데이터를 모니터 및 유지 관리할 수 있습니다. folder.db는 ACL의 복사본을 보유하는 각 폴더에 대한 레코드를 가집니다. peruser.db는 각 사용자 및 메일함에 대한 항목을 가지며 이 항목은 다양한 플래그 설정과 사용자가 임의 폴더에 마지막으로 액세스한 날짜를 나열합니다. lright.db는 모든 사용자와 사용자가 조회 권한을 가진 공유 폴더를 나열합니다.

readership 명령줄 유틸리티에는 다음 옵션이 있습니다.

표 20–5 readership 옵션

옵션	설명
-d days	지정된 날짜 동안 해당 폴더를 선택한 사용자 수에 대한 보고서를 공유 폴더별로 반환합니다.
-p months	지정된 개월 수 동안 공유 폴더를 선택하지 않는 사용자에 대해 peruser.db에서 데이터를 제거합니다.
-l	lright.db의 데이터를 나열합니다.
-s *folder_identifier_rights*	지정된 폴더에 대한 액세스 권한을 설정합니다. folder.db와 lright.db를 업데이트합니다.

다양한 옵션을 사용하여 다음 기능을 수행할 수 있습니다.

- 572 페이지 "20.6.5.1 공유 폴더 사용 모니터"
- 573 페이지 "20.6.5.2 사용자 및 사용자의 공유 폴더 나열"
- 573 페이지 "20.6.5.3 비활성 사용자 제거"
- 573 페이지 "20.6.5.4 액세스 권한 설정"

20.6.5.1 공유 폴더 사용 모니터

공유 폴더를 액세스하는 활성 사용자 수를 확인하려면 다음 형식의 명령을 실행합니다.

```
readership -d days
```

여기서 *days*는 검사할 일 수입니다. 이 옵션은 활성 사용자 목록이 아니라 그 수를 반환한다는 점에 주의하십시오.

예: 마지막 30일 이내에 공유 폴더를 선택한 사용자 수를 확인하려면 다음 명령을 실행합니다.

```
readership -d 30
```

20.6.5.2 사용자 및 사용자의 공유 폴더 나열

사용자와 사용자가 액세스할 수 있는 공유 폴더를 나열하려면 다음 명령을 실행합니다.

```
imcheck -d lright.db
```

출력 예는 다음과 같습니다.

```
$ imcheck -d lright.db
group=lee-staff@siroe.com: user/user2/lee-staff
richb: user/golf user/user10/Drafts user/user2/lee-staff user/user10/Trash
han1: user/public+hurling@siroe.com user/golf
gregk: user/public+hurling@siroe.com user/heaving user/tennis
```

20.6.5.3 비활성 사용자 제거

지정된 기간 동안 공유 폴더에 액세스하지 않은 비활성 사용자를 제거하려면 다음 명령을 실행합니다.

```
readership -p months
```

여기서 *months*는 검사할 개월 수입니다.

예: 지난 6개월 동안 공유 폴더에 액세스하지 않은 사용자를 제거합니다.

```
readership -p 6
```

20.6.5.4 액세스 권한 설정

새 공유 폴더에 대한 액세스 권한을 할당하거나 현재 공유 폴더에서 액세스 권한을 변경할 수 있습니다.

이 명령으로 액세스 권한을 설정하는 방법의 예는 568 페이지 “20.6.2 공유 폴더의 액세스 제어 권한 설정 또는 변경”을 참조하십시오.

20.7 메시지 유형 관리

이 절은 다음 내용으로 구성되어 있습니다.

20.7.1 메시지 유형 개요

통합 메시징 응용 프로그램은 텍스트 메시지, 음성 메일, 팩스 메일, 이미지 데이터 및 기타 데이터 형식을 포함하여 여러 유형의 메시지를 받고, 보내고, 저장하고, 관리할 수 있습니다. 메시지 저장소를 사용하면 서로 다른 메시지 유형을 63개까지 정의할 수 있습니다.

메시지를 유형별로 조작하는 방법 중 하나는 메시지를 개별 폴더에 유형별로 구성하는 것입니다.

메시지 유형 기능을 사용할 수 있기 때문에 서로 다른 여러 메시지 유형을 개별 메일함 폴더에서 유지 관리할 필요가 없습니다. 메시지 유형을 구성하고 나면 저장 위치에 관계없이 메시지 저장소에서 확인할 수 있습니다. 따라서 같은 폴더에 서로 다른 메시지 유형을 저장할 수 있습니다. 또한 다음 작업을 수행할 수 있습니다.

- 메시지 유형의 사용 추적
- 메시지 유형별로 그룹화된 알림 전송
- 같은 폴더 또는 다른 폴더에 저장된 여러 메시지 유형에 서로 다른 할당량을 설정 및 관리
- 각 메시지 유형에 고유하게 구성된 기준에 따라 한 폴더에서 다른 폴더로 메시지 이동
- 각 메시지 유형에 구성된 기준에 따라 메시지 만료

20.7.1.1 메시지 유형 구성 계획

통합 메시징 응용 프로그램에서 이질적인 형식의 데이터에는 Messaging Server가 데이터를 저장 및 관리할 수 있도록 표준 인터넷 메시지 헤더가 지정됩니다. 예를 들어 최종 사용자의 전화로 음성 메일을 전송하면 전화 프런트엔드 시스템이 받는 음성 메일에 메시지 헤더를 추가하여 메시지 저장소로 전달합니다.

서로 다른 여러 유형의 메시지를 인식하고 관리하려면 통합 메시징 시스템의 모든 구성 요소에서 같은 메시지 유형 정의와 헤더 필드를 사용하여 메시지를 식별해야 합니다.

메시지 유형을 지원하도록 메시지 저장소를 구성하려면 먼저 다음을 수행해야 합니다.

- 사용할 메시지 유형 계획
- 각 메시지 유형 정의 결정
- 사용할 헤더 필드 결정

예를 들어, 응용 프로그램에 전화 메시지가 포함된 경우 이 메시지 유형을 "multipart/voice-message"로 정의하고 내용 유형 헤더 필드를 사용하여 메시지 유형을 확인할 수 있습니다.

그런 다음 메시지 저장소로 전달되는 각 전화 메시지에 다음 헤더 정보를 추가하도록 전화 프런트엔드 시스템을 구성합니다.

```
Content-Type: multipart/voice-message
```

다음으로는 뒤의 절에 있는 설명에 따라 multipart/voice-message 메시지 유형을 인식하도록 메시지 저장소를 구성합니다.

20.7.1.2 메시지 유형 정의 및 사용

multipart/voice-message와 같은 고유한 정의를 지정하여 메시지 유형을 정의합니다. 기본적으로 메시지 저장소는 내용 유형 헤더 필드를 읽고 메시지 유형을 결정합니다. 원하는 경우 다른 헤더 필드를 구성하여 메시지 유형을 식별할 수 있습니다.

메시지 저장소에서는 대소문자를 무시하고 내용 유형(또는 기타 지정된) 헤더 필드를 읽습니다. 즉, 헤더의 대소문자 조합이 예상된 조합과 다른 경우에도 메시지 저장소에서 헤더 필드를 유효한 것으로 승인합니다.

메시지 저장소는 헤더 필드에 있는 메시지 유형 이름만 읽습니다. 추가 인수 또는 매개변수는 무시합니다.

메시지 유형을 정의하려면 configutil 유틸리티를 사용하여 store.messagetype 매개변수의 값을 설정합니다. 자세한 지침은 575 페이지 "메시지 유형을 구성하는 방법"을 참조하십시오.

메시지 유형을 구성하면 메시지 저장소에서 지정된 유형의 메시지를 식별하고 조작할 수 있습니다. 이 단계가 통합 메시징 응용 프로그램에서 메시지 유형을 관리하기 위해 필요한 첫 단계입니다.

메시지 저장소에서 제공하는 메시지 유형 기능의 장점을 모두 활용하려면 다음 작업 중 일부 또는 전부를 수행해야 합니다.

- JMQ 알림 플러그 인을 구성하고 메시지 유형의 상태를 추적하는 알림을 검색할 Message Queue 클라이언트를 작성
- 각 메시지 유형에 적용되는 할당량 루트 구성
- 메시지 유형에 따라 메시지가 만료 및 제거되도록 만료 규칙을 작성하고 LDAP 속성 값 설정

이러한 작업에 대한 요약은 다음 절을 참조하십시오.

- 579 페이지 "20.7.3 메시지 유형에 해당하는 알림 메시지 전송"
- 579 페이지 "20.7.4 메시지 유형별로 할당량 관리"
- 581 페이지 "20.7.5 메시지 유형별 메시지 만료"

▼ 메시지 유형을 구성하는 방법

메시지 유형을 구성하려면 configutil 유틸리티를 사용하여 메시지 유형을 정의하고 식별하는 store.messagetype 매개 변수 값을 설정합니다.

1 **store.messagetype.enable 매개 변수를 on으로 설정하여 메시지 유형을 활성화합니다.**

이 configutil 매개 변수를 사용하면 메시지 저장소에서 메시지 유형을 식별 및 조작할 수 있습니다. 개별 메시지 유형을 구성하려면 먼저 이 매개 변수를 설정해야 합니다.

예를 들어, 다음 명령을 입력합니다.

```
configutil -o store.messagetype.enable -v 1
```

2 store.messagetype.*x* 매개 변수를 설정하여 메시지 유형을 정의 및 식별합니다.

변수 *x*는 메시지 저장소에서 이 특정 메시지 유형을 식별합니다. 변수 *x*는 0보다 크고 64보다 작은 정수여야 합니다. 고유한 정수를 사용하여 반복적으로 이 매개 변수를 구성하면 메시지 유형을 63개까지 정의할 수 있습니다.

유형을 설명하는 텍스트 문자열을 사용하여 메시지 유형의 값을 정의합니다.

예를 들어 텍스트 메시지 유형을 정의하려면 다음 명령을 입력합니다.

```
configutil -o store.messagetype.1 -v text/plain
```

음성 메시지 유형을 정의하려면 다음 명령을 입력합니다.

```
configutil -o store.messagetype.2 -v multipart/voice-message
```

3 store.messagetype.*x*.flagname 매개 변수를 설정하여 메시지 유형에 플래그 이름을 지정합니다.

이 매개 변수는 메시지 유형을 식별하는 고유한 플래그를 만듭니다. 이 유형의 메시지가 처음 메시지 저장소에 도착할 때마다 플래그가 자동으로 설정되며 제거될 때까지 메시지와 연결된 상태로 남습니다. 플래그 이름 값은 메시지 유형을 설명하는 텍스트 문자열입니다. store.messagetype.*x* 매개 변수로 설정한 값과 같은 필요는 없습니다.

변수 *x*는 store.messagetype.*x* 매개 변수로 정의한 메시지 유형의 정수 아이디입니다.

예를 들어 앞의 단계에서 구성한 메시지 유형에 플래그 이름을 정의하려면 다음 명령을 입력합니다.

```
configutil -o store.messagetype.1.flagname -v text

configutil -o store.messagetype.2.flagname -v voice_message
```

4 store.messagetype.*x*.quotaroot 매개 변수를 설정하여 메시지 유형의 할당량 루트 이름을 구성합니다.

이 매개 변수를 사용하면 할당량 기능에서 이 메시지 유형의 할당량 루트를 식별 및 관리할 수 있습니다. 매개 변수 값은 이름, 즉 메시지 유형을 설명하는 텍스트 문자열입니다. store.messagetype.*x* 매개 변수로 설정한 값과 같을 필요는 없습니다.

변수 *x*는 store.messagetype.*x* 매개 변수로 정의한 메시지 유형의 정수 아이디입니다.

이 매개 변수가 구성되어 있으면 지정한 메시지 유형에 적용되는 할당량을 설정할 수 있습니다. 자세한 내용은 579 페이지 “20.7.4 메시지 유형별로 할당량 관리”를 참조하십시오.

예를 들어, 앞의 단계에서 구성한 메시지 유형에 대해 할당량 루트 사용을 활성화하려면 다음 명령을 입력합니다.

```
configutil -o store.messagetype.1.quotaroot -v text

configutil -o store.messagetype.2.quotaroot -v voice
```

5 **메시지 유형을 식별하기 위한 대체 헤더 필드를 구성하려면** `store.messagetype.header` **매개 변수를 설정합니다.**

기본적으로 메시지 저장소는 내용 유형 헤더 필드를 읽고 메시지 유형을 결정합니다. `store.messagetype.header` 매개 변수는 메시지 유형 식별을 위해 다른 헤더 필드를 사용하려는 경우에만 구성합니다. 이 매개 변수 값은 텍스트 문자열입니다.

예를 들어 `X-Message-Type`이라는 필드를 사용하려면 다음 명령을 입력합니다.

```
configutil -o store.messagetype.header -v X-Message-Type
```

20.7.2 IMAP 명령의 메시지 유형

메시지 유형에 `store.messagetype.`*x*`.flagname` 매개 변수를 구성할 때, 메시지 유형을 식별하는 고유 플래그를 만듭니다. 이 플래그는 최종 사용자가 수정할 수 없습니다.

Messaging Server는 메시징 유형 플래그를 IMAP 클라이언트에 사용자 플래그로 제공합니다. 메시지 유형을 사용자 플래그에 매핑하면 메일 클라이언트에서 간단한 IMAP 명령을 사용하여 메시지 유형별로 메시지를 조작할 수 있습니다.

예를 들어, 다음 작업을 수행할 수 있습니다.

- IMAP FETCH FLAGS 명령을 사용하여 메시지 유형 플래그 이름을 클라이언트에 사용자 정의 플래그로 표시합니다.

  IMAP FETCH FLAGS 명령의 사용 샘플은 아래에 있는 예 20–1을 참조하십시오.
- IMAP SEARCH 명령에서 메시지 유형 플래그를 키워드로 사용합니다.

  IMAP SEARCH 명령의 사용 샘플은 아래에 있는 예 20–1을 참조하십시오.

메시지 유형 사용자 플래그는 읽기 전용입니다. IMAP 명령으로 수정할 수 없습니다.

다음 예에서는 아래에 표시된 값으로 메시지 유형 `configutil` 매개 변수를 구성하는 경우를 가정합니다.

```
store.messagetype.enable = yes

store.messagetype.1 = text/plain
store.messagetype.1.flagname = text
store.messagetype.1.quotaroot = text

store.messagetype.2 = multipart/voice-message
store.messagetype.2.flagname = voice_message
store.messagetype.2.quotaroot = voice
```

예 20–1 메시지 유형 configutil 구성을 기반으로 한 IMAP FETCH 세션

다음 IMAP 세션에서는 현재 선택한 메일함의 메시지를 불러옵니다.

```
2 fetch 1:2 (flags rfc822)
* 1 FETCH (FLAGS (\Seen text) RFC822 {164}

Date: Wed, 8 July 2006 03:39:57 -0700 (PDT)
From: bob.smith@siroe.com
To: john.doe@siroe.com
Subject:  Hello
Content-Type: TEXT/plain; charset=us-ascii


* 2 FETCH (FLAGS (\Seen voice_message) RFC822 {164}

Date: Wed, 8 July 2006 04:17:22 -0700 (PDT)
From: sally.lee@siroe.com
To: john.doe@siroe.com
Subject:  Our Meeting
Content-Type: MULTIPART/voice-message; ver=2.0

2 OK COMPLETED
```

앞의 예에서는 두 개의 메시지를 불러오며, 하나는 텍스트 메시지이고 하나는 음성 메일입니다.

메시지 유형 플래그는 store.messagetype.*.flagname 매개 변수를 사용하여 구성한 형식으로 표시됩니다.

내용 유형 헤더 필드는 메시지 유형을 식별합니다. 메시지 유형 이름은 받는 메시지에서 받은 그대로 표시됩니다. 여기에는 대문자와 소문자를 혼합하여 사용하며 charset=us-ascii와 같은 메시지 유형 인수를 포함합니다.

예 20–2 메시지 유형 configutil 구성을 기반으로 한 IMAP SEARCH 세션

다음 IMAP 세션에서는 현재 선택한 메일함의 음성 메시지를 검색합니다.

```
3 search keyword voice_message
* SEARCH 2 4 6
3 OK COMPLETED
```

앞의 예에서 메시지 2, 4, 6은 음성 메시지입니다. 검색에 사용되는 키워드는 store.messagetype.2.flagname 매개 변수 값인 voice_message입니다.

20.7.3 메시지 유형에 해당하는 알림 메시지 전송

알림은 텍스트 메시지, 음성 메시지 및 이미지 데이터 등의 여러 메시지 유형에 대해 상태 정보를 전달할 수 있습니다. Messaging Server에서는 Sun Java System Message Queue를 사용하여 메시지 유형에 해당하는 알림 정보를 보냅니다. Message Queue용 JMQ 알림 플러그 인 구성에 대한 자세한 내용은 Message Queue 22 장을 참조하십시오.

JMQ 알림 플러그 인에서 특정 메시지 유형을 인식할 수 있게 하려면 `store.messagetype.x.flagname` 매개 변수를 포함한 `store.messagetype` 매개 변수를 구성해야 합니다. 자세한 내용은 575 페이지 "메시지 유형을 구성하는 방법"을 참조하십시오.

메시지 유형을 구성하고 나면 JMQ 알림 메시지에서 특정 메시지 유형을 식별할 수 있습니다. 메시지 유형별로 알림 메시지를 해석하고 각 유형에 대한 상태 정보를 메일 클라이언트로 전달하도록 Message Queue 클라이언트를 작성할 수 있습니다.

JMQ 알림 기능에서는 현재 메일함에 있는 메시지의 수를 메시지 유형별로 계산합니다. 수 값이 하나 전달되는 대신 각 메시지 유형의 수를 지정하는 배열이 알림 메시지와 함께 전달됩니다.

예를 들어, `NewMsg` 알림 메시지는 새 음성 메일 메시지 7개와 새 텍스트 메시지 4개가 사용자의 받은 메일함에 있다고 알리는 데이터를 전달할 수 있습니다.

메시지 유형별 알림 전송에 대한 자세한 내용은 670 페이지 "22.3.3 특정 메시지 유형의 알림"을 참조하십시오.

20.7.4 메시지 유형별로 할당량 관리

메시지 유형에 대해 할당량을 설정할 경우 해당 값을 **할당량 루트**에 포함합니다. 할당량 루트는 사용자에 대한 할당량을 지정합니다. 여기서는 특정 메시지 유형 및 메일함 폴더에 다른 할당량을 지정할 수 있으며, 유형으로 정의되지 않은 모든 나머지 메시지 유형, 폴더 및 메시지에 적용되는 기본 할당량을 지정할 수 있습니다.

할당량 설정 및 관리에 대한 자세한 내용은 584 페이지 "20.8.2 할당량 작동 원리"를 참조하십시오.

20.7.4.1 메시지 유형 할당량을 설정하기 전에

메시지 유형에 할당량을 설정하기 전에 다음 매개 변수를 구성해야 합니다.

- 각 메시지 유형에 대해 `store.messagetype.x.quotaroot` 매개 변수를 설정합니다. 자세한 내용은 575 페이지 "메시지 유형을 구성하는 방법"을 참조하십시오.
- `store.typequota.enable` 매개 변수를 `on`으로 설정합니다.

  예를 들어, 다음 명령을 입력합니다.

```
configutil -o store.typequota.enable -v 1
```

20.7.4.2 메시지 유형 할당량 설정 방법

다음 중 한 가지 방법을 사용하여 메시지 유형의 할당량을 설정합니다.

- LDAP 속성 `mailQuota` 또는 `mailMsgQuota`(또는 둘 다)를 사용하여 사용자에 대한 메시지 유형 할당량을 설정합니다.

  이 속성에 할당량 루트를 설정하는 방법에 대한 자세한 내용은 **Sun Java Communications Suite 5 Schema Reference**의 3 장, "Messaging Server and Calendar Server Attributes"에 있는 `mailQuota` 및 `mailMsgQuota` 항목을 참조하십시오.

- `mailQuota` 및 `mailMsgQuota` 속성을 설정하지 않은 경우 모든 개별 사용자에 적용되는 기본 메시지 유형 할당량을 설정합니다.

  기본 할당량을 설정하려면 `store.defaultmessagequota` 또는 `store.defaultmailboxquota` 매개 변수(또는 둘 다)를 사용합니다.

  이 매개 변수에 할당량 루트를 설정하는 방법에 대한 자세한 내용은 587 페이지 "20.8.4 메시지 저장소 할당량 구성"을 참조하십시오.

`configutil` 매개 변수나 위에 표시된 LDAP 속성을 사용하여 메시지 유형에 대한 할당량을 설정하는 경우 `store.messagetype.`*x*`.quotaroot` 매개 변수를 사용하여 지정한 할당량 루트를 사용해야 합니다.

20.7.4.3 메시지 유형 할당량 루트의 예

이 절에 설명된 예에서는 사용자 `joe`에 대해 다음 할당량을 설정합니다.

- 기본 메일함 저장소 할당량은 40M입니다.
- 기본 메일함 메시지 할당량은 5000입니다.
- Archive 폴더의 저장소 할당량은 100M입니다.
- 텍스트 메시지 유형의 저장소 할당량은 10M입니다.
- 텍스트 메시지 유형의 메시지 할당량은 2000입니다.
- 음성 메시지 유형의 저장소 할당량은 10M입니다.
- 음성 메시지 유형의 메시지 할당량은 200입니다.

이 할당량 루트는 다른 모든 폴더 및 메시지 유형을 합한 것보다(60M) Archive 폴더에 더 큰 저장소(100M)를 허용합니다. 또, Archive 폴더에는 메시지 제한이 없습니다. 이 예에서 아카이브에 문제가 되는 것은 저장소 제한 뿐입니다.

메시지 유형에는 저장소 및 메시지 수 할당량이 모두 있습니다.

메시지 유형 할당량은 Archive 폴더에 저장되었는지 또는 다른 폴더에 저장되었는지에 관계 없이 해당 유형의 모든 메시지를 합한 것에 적용됩니다.

기본 메일함 할당량은 텍스트 또는 음성 메시지 유형이 아니고 Archive 폴더에 저장되지 않는 모든 메시지에 적용됩니다. 즉, 메시지 유형 할당량과 Archive 할당량은 기본 메일함 할당량의 일부로 계산되지 않습니다.

이 예에서 할당량 루트를 설정하려면 다음 단계를 수행합니다.

1. 다음과 같이 store.messagetype.*x*.quotaroot 매개 변수를 구성합니다.

```
store.messagetype.1.quotaroot = text

store.messagetype.2.quotaroot = voice
```

2. 다음과 같이 사용자 joe에 대해 mailQuota 속성을 구성합니다.

```
mailQuota: 20M;#text%10M;#voice%10M;Archive%100M
```

3. 다음과 같이 사용자 joe에 대해 mailMsgQuota 속성을 구성합니다.

```
mailMsgQuota: 5000;#text%2000;#voice%200
```

getquotaroot IMAP 명령을 실행하면 다음과 같이 결과 IMAP 세션에 사용자 joe의 메일함에 해당되는 모든 할당량 루트가 표시됩니다.

```
1 getquotaroot INBOX
* QUOTAROOT INBOX user/joe user/joe/#text user/joe/#voice
* QUOTA user/joe (STORAGE 12340 20480 MESSAGE 148 5000)
* QUOTA user/joe/#text (STORAGE 1966 10240 MESSAGE 92 2000)
* QUOTA user/joe/#voice (STORAGE 7050 10240 MESSAGE 24 200)

2 getquotaroot Archive
* QUOTAROOT user/joe/Archive user/joe/#text user/joe/#voice
* QUOTA user/joe/Archive (STORAGE 35424 102400)
* QUOTA user/joe/#text (STORAGE 1966 10240 MESSAGE 92 2000)

* QUOTA user/joe/#voice (STORAGE 7050 10240 MESSAGE 24 200)
```

20.7.5 메시지 유형별 메시지 만료

만료 및 제거 기능을 사용하면 만료 규칙에 정의된 기준에 따라 메시지를 한 폴더에서 다른 폴더로 옮기고, 아카이브에 보관하고, 메시지 저장소에서 제거할 수 있습니다. 이 작업은 imexpire 유틸리티를 사용하여 수행합니다.

imexpire 유틸리티는 관리자가 실행하기 때문에 할당량 적용을 무시합니다.

만료 규칙을 작성하고 imexpire 유틸리티를 사용하는 방법에 대한 자세한 내용은 592 페이지 “20.9 자동 메시지 제거(만료 및 제거) 기능 설정 방법”을 참조하십시오.

각기 다른 기준에 따라 다른 유형의 메시지가 만료되도록 만료 규칙을 작성할 수 있습니다.

만료 기능은 만료 기준 설정에 선택할 수 있는 많은 옵션을 제공하여 매우 유연합니다. 이 절에서는 서로 다른 기준에 따라 텍스트 및 음성 메시지가 만료되는 한 가지 예에 대해 설명합니다.

이 예에서는 다음과 같이 텍스트 및 음성 메시지 유형을 구성한 경우를 가정합니다.

```
store.messagetype.1 = text/plain

store.messagetype.2 = multipart/voice-message
```

또한 메시지 저장소가 내용 유형 헤더 필드를 읽고 메시지 유형을 결정하도록 구성된 경우를 가정합니다.

예 20–3 서로 다른 메시지 유형을 만료하는 샘플 규칙

```
TextInbox.folderpattern: user/%/INBOX
TextInbox.messageheader.Content-Type: text/plain
TextInbox.messagedays: 365
TextInbox.action: fileinto:Archive


VoiceInbox.folderpattern: user/%/INBOX
VoiceInbox.messageheader.Content-Type: multipart/voice-message
VoiceInbox.savedays: 14
VoiceInbox.action: fileinto:OldMail

VoiceOldMail.folderpattern: user/%/OldMail
VoiceOldMail.messageheader.Content-Type: multipart/voice-message
VoiceOldMail.savedays: 30
VoiceOldMail.action: fileinto:Trash

Trash.folderpattern: user/%/Trash
Trash.savedays: 7
Trash.action: discard
```

이 예에서 텍스트 메시지와 음성 메시지는 서로 다른 방식으로 만료되며 다음과 같이 서로 다른 일정을 따릅니다.

- 텍스트 메시지는 메시지 저장소에 도착한 후 1년이 지나면 사용자의 받은 메일함에서 사용자의 Archive 폴더로 이동합니다.
- 음성 메일은 2주가 지나면 받은 메일함에서 OldMail 폴더로 이동합니다. 사용자가 음성 메시지를 저장하면 저장된 날짜가 재설정되고, 메시지는 새로운 날짜로부터 2주가 지난 후에 이동합니다.
- 음성 메일은 30일이 지나면 OldMail 폴더에서 Trash 폴더로 이동합니다. 사용자는 OldMail 폴더에 음성 메시지를 저장할 수 있습니다. 그러면 메시지 제거가 새로 저장된 날짜로부터 30일 후로 연기됩니다.
- 모든 유형의 메시지는 Trash 폴더로 이동한 후 7일이 지나면 삭제됩니다.

  만료 규칙은 음성 메일을 지운 편지함으로 자동으로 옮깁니다. 텍스트 메시지는 사용자가 메시지를 삭제하면 지운 편지함으로 이동합니다.

예 20–3 서로 다른 메시지 유형을 만료하는 샘플 규칙 *(계속)*

주: savedays 규칙을 사용하면 메시지를 저장한 후 지정된 일 수가 지났을 때 메시지가 만료됩니다. 일반적인 음성 메일 시스템에서 사용자는 음성 메일 메뉴로 음성 메일을 저장할 수 있습니다. 텍스트 메시지의 경우 메시지는 폴더로 이동하면 저장됩니다. messagedays 규칙을 사용하면 메시지가 메시지 저장소에 처음 도착한 후 지정된 일 수가 지나면 메시지가 저장된 폴더나 이동 빈도에 관계 없이 만료됩니다.

20.8 메시지 저장소 할당량 정보

전자 메일과 음성 메일이 폭주하면 IMAP 메일함이 매우 커질 수 있습니다. 사용자 또는 도메인에서 보관할 수 있는 메시지의 수나 메시지에 사용할 수 있는 디스크 공간에 대한 메시지 저장소 할당량 제한, 즉 **할당량**이 특정 폴더 또는 특정 메시지 유형에 대해 지정됩니다. 할당량은 메시지 저장소 사용을 제한하거나 줄이기 위해 사용됩니다. 이 절은 다음 내용으로 구성되어 있습니다.

자세한 내용은 609 페이지 “20.11.4 할당량 제한 모니터”를 참조하십시오.

20.8.1 할당량 개요

할당량은 특정 사용자 또는 도메인에 대해 설정할 수 있으며 메시지 수 또는 바이트 수를 기준으로 설정할 수 있습니다. 특정 폴더 및 메시지 유형에 대해 설정할 수도 있습니다. 메시지 유형 할당량을 사용하면 메시지 유형에 대해 제한을 지정할 수 있습니다. 예를 들어 음성 메일과 전자 메일이 있습니다. 폴더 할당량은 사용자의 폴더 크기를 바이트 수 또는 메시지 수로 제한합니다. 예를 들어, Trash 폴더에 할당량을 설정할 수 있습니다. Messaging Server를 사용하면 도메인과 사용자의 기본 할당량과 사용자 정의 할당량을 설정할 수 있습니다.

할당량을 설정하고 나면 할당량을 초과했거나 할당량에 도달해 가는 사용자 또는 도메인에 대해 시스템이 응답하는 방식도 구성할 수 있습니다. 응답 중 한 가지로 사용자에게 **할당량 알림**을 보내는 것이 있습니다. 다른 응답으로는 할당량을 초과한 경우에 메시지 저장소에 대한 메시지 전달을 중지하는 것이 있습니다. 이를 **할당량 적용**이라고 하며 보통 지정된 **유예 기간**이 지난 후에 발생합니다. 유예 기간은 적용이 일어나기 전에 메일함이 할당량을 초과한 상태로 유지될 수 있는 기간을 말합니다. 할당량 초과로 인해 메시지 전달이 중지되는 경우 받는 메시지는 다음 중 하나가 발생할 때까지 MTA 대기열에 남아 있습니다.

- 사용자 메시지의 크기 또는 수가 더 이상 할당량을 초과하지 않으면 MTA에서 메시지를 전달합니다.

- 배달되지 않은 메시지가 지정된 **유예 기간**보다 오래 MTA 대기열에 남아 있으면 메시지가 보낸 사람에게 돌아갑니다. 591 페이지 "20.8.4.5 유예 기간 설정"을 참조하십시오.
- 메시지가 최대 메시지 대기 시간보다 오래 메시지 대기열에 남아 있습니다. 이 값은 `notices` MTA 채널 키워드(253 페이지 "10.10.4.3 알림 메일 전달 간격 설정" 참조)에 의해 제어됩니다.

  예를 들어, 유예 기간이 2일로 설정되어 있고 할당량을 1일 동안 초과한 경우 새 메시지는 계속 들어와 메시지 대기열에 보관되며 전달 시도도 계속됩니다. 2일 이후에 메시지가 보낸 사람에게 바운스됩니다.

사용자가 메시지를 삭제 및 정리하거나 서버에서 설정된 만료 정책에 따라 메시지를 삭제하면 디스크 공간을 사용할 수 있게 됩니다(592 페이지 "20.9 자동 메시지 제거(만료 및 제거) 기능 설정 방법" 참조).

20.8.1.1 Telephony Application Server 예외 사항

통합 메시징 요구 사항을 지원하기 위해 Messaging Server는 메시지 저장소에서 부과한 할당량 제한을 무시하는 기능을 제공합니다. 이 기능은 TAS(Telephony Application Server)라는 특정 에이전트에서 받은 메시지가 전달되도록 합니다. TAS가 받은 메시지는 할당량 제한에 상관 없이 메시지가 저장소로 전달되도록 하는 특수한 MTA 채널을 통해 라우팅됩니다. 난해한 용법이기는 하지만 전화 통신 응용 프로그램에 사용할 수 있습니다. TAS 채널 구성에 대한 자세한 내용은 Sun 메시징 담당자에게 문의하십시오.

메시지 유형별 할당량은 통합 메시징을 사용하는 전화 통신 응용 프로그램에 유용합니다. 예를 들어 대화 텍스트 및 음성 메일이 섞인 메시지가 사용자의 메일함에 저장되어 있는 경우 관리자는 다른 유형의 메시지에 대해 다른 할당량을 설정할 수 있습니다. 사용자의 전자 메일에 할당량이 하나 지정되고, 음성 메일에 다른 할당량이 지정될 수 있습니다.

20.8.2 할당량 작동 원리

사용자 정의된 사용자 및 도메인 할당량은 LDAP 사용자 및 도메인 항목에 할당량 속성을 추가하여 지정합니다. 할당량 기본값, 알림 정책, 적용 및 유예 기간은 `configutil` 매개변수에 지정되거나 `imquotacheck` 유틸리티를 사용하여 지정됩니다.

사용자가 할당량을 초과하는지 확인하기 위해 Messaging Server는 우선 개별 사용자에 대한 할당량이 설정되었는지 검사합니다. 할당량이 설정되지 않은 경우 Messaging Server는 모든 사용자에 설정된 기본 할당량을 확인합니다. 사용자의 경우 할당량은 사용자의 폴더 전체에 있는 바이트 또는 메시지 수를 모두 누적한 값에 적용됩니다. 도메인의 경우 할당량은 특정 도메인에 있는 모든 사용자의 바이트 또는 메시지 수를 모두 누적한 값에 적용됩니다. 메시지 유형의 경우 할당량은 해당 메시지 유형의 바이트 또는 메시지 수를 모두 누적한 값에 적용됩니다. 폴더의 경우 할당량은 사용자의 폴더의 바이트 또는 메시지 수를 모두 누적한 값에 적용됩니다.

사용자의 메일함 트리에 다음과 같은 할당량 값을 지정할 수 있습니다.

- 사용자의 메일함에 있는 특정 폴더의 할당량 값
- 음성 메일이나 텍스트 메시지와 같은 특정 메시지 유형에 대한 할당량 값. (메시지 유형 할당량은 사용자의 메일함에 있는 모든 폴더에서 해당 유형의 메시지에 적용됩니다.)
- 할당량이 명시적으로 지정되지 않은 사용자 메일함 내의 모든 폴더 및 메시지 유형에 적용되는 기본 할당량 값

사용자에 대해 여러 할당량 값을 지정하는 경우 다음 지침이 적용됩니다.

- 할당량은 서로 중복되지 않습니다. 예를 들어 특정 메시지 유형 또는 폴더에 대한 할당량이 있는 경우 해당 유형이나 폴더의 메시지는 기본 할당량 계산에 포함되지 않습니다. 각 메시지는 한 할당량에 대해서만 계산됩니다.
- 사용자 메일함 전체의 총 할당량은 기본, 유형 및 폴더에 지정된 모든 할당량 값을 합한 것과 같습니다.
- 메시지 유형 할당량은 폴더 할당량보다 우선적으로 사용됩니다. 예를 들어, 사용자의 `memos` 폴더에 한 할당량이 지정되어 있고 음성 메시지에 다른 할당량이 지정되어 있는 경우를 가정할 수 있습니다. 이제 사용자가 `memos` 폴더에 8개의 음성 메시지를 저장하는 경우를 가정합니다. 8개의 메시지는 음성 메일 할당량에 대해 계산되며 `memos` 폴더 할당량에서 제외됩니다.

할당량 속성과 `configutil` 매개 변수의 변경 사항은 자동으로 적용되지만 정보가 캐시에 저장되기 때문에 즉시 적용되는 것은 아니며, 변경 사항이 완전히 적용될 때까지 시간이 걸릴 수 있습니다. Messaging Server에는 변경 사항을 즉시 업데이트하는 **Sun Java System Messaging Server 6.3 Administration Reference**의 "iminitquota" 명령이 있습니다.

`imquotacheck` 유틸리티를 사용하면 지정된 할당량을 기준으로 메시지 저장소 사용을 확인할 수 있습니다.

20.8.3 메시지 저장소 할당량 속성 및 매개 변수

이 절에서는 주요 메시지 저장소 할당량 속성과 `configutil` 매개 변수를 나열합니다. 이는 기능 인터페이스의 개요를 제공하기 위한 것입니다. 이 속성과 매개 변수에 대한 자세한 내용은 해당 참조 설명서를 참조하십시오.

다음 표에는 할당량 속성이 나열되어 있습니다. **Sun Java Communications Suite 5 Schema Reference**를 참조하십시오.

표 20–6 메시지 저장소 할당량 속성

속성	설명
`mailQuota`	사용자의 메일함에 허용되는 디스크 공간(바이트)입니다.

표 20–6 메시지 저장소 할당량 속성 *(계속)*

속성	설명
`mailMsgQuota`	사용자에게 허용되는 최대 메시지 수입니다. 이 값은 저장소에 있는 모든 폴더에 대한 누적 개수입니다.
`mailUserStatus`	메일 사용자의 상태입니다. 가능한 값에는 `active`, `inactive`, `deleted`, `hold`, `overquota` 등이 있습니다.
`mailDomainDiskQuota`	허용되는 디스크 공간(바이트)으로 도메인에 있는 모든 메일함에 대한 누적 개수입니다.
`mailDomainMsgQuota`	도메인에 허용되는 최대 메시지 수 즉, 저장소에 있는 모든 메일함에 대한 개수 합계입니다.
`mailDomainStatus`	메일 도메인의 상태입니다. 값 및 기본값은 `mailUserStatus`와 동일합니다.

다음 표에는 할당량 매개 변수가 나열되어 있습니다. 자세한 최신 내용은 **Sun Java System Messaging Server 6.3 Administration Reference**의 3 장, “Messaging Server Configuration”을 참조하십시오.

표 20–7 메시지 저장소 `configutil` 매개 변수

매개 변수	설명
store.quotaenforcement	할당량 적용을 활성화합니다. Off이면 할당량 데이터베이스가 계속 업데이트되 지만 메시지가 항상 전달됩니다. 기본값: On입니다.
store.quotanotification	할당량 알림을 활성화 합니다. 기본값: OFF입니다.
store.defaultmailboxquota	저장소 기본 할당량(바이트)입니다. 기본값: -1(제한 없음)입니다.
store.defaultmessagequota	저장소 기본 할당량(메시지 수)입니다. 숫자 값입니다. 기본값: -1(제한 없음)입니다.
store.quotaexceededmsg	할당량 경고 메시지입니다. 이 값이 없으면 알림을 보내지 않습니다. 기본값: 없습니다.
store.quotaexceededmsginterval	할당량 초과 알림을 보내는 간격(일)입니다. 기본값: 7
store.quotagraceperiod	메일함이 할당량 초과된 후 메일함의 메일을 보낸 사람에게 돌려보낼 때까지 허용된 시간(시)입니다. 시간 수입니다. 기본값: 120입니다.
store.quotawarn	할당량 경고 임계값. 클라이언트에게 할당량 경고를 보내기 전에 초과한 할당량 비율입니다. 기본값: 90입니다.
local.store.quotaoverdraft	Netscape Messaging Server에서 마이그레이션된 시스템과의 호환성을 제공하는 데 사용됩니다. ON인 경우 디스크 사용량이 할당량을 초과하는 메시지 전달을 허용합니다. 사용자가 할당량을 초과하면 메시지가 지연되거나 바운스되고 할당량 경고 메시지가 발송되며 할당량 유예 기간 타이머가 시작됩니다. 기본값은 메시지 저장소가 임계값에 도달할 때 할당량 경고 메시지를 보냅니다. 기본값: Off입니다. 하지만 store.overquotastatus가 설정된 경우에는 on으로 간주되며 그렇지 않은 경우에는 사용자가 할당량을 초과할 수 없고 `overquotastatus` 가 사용되지 않습니다.

표 20–7 메시지 저장소 configutil 매개 변수 (계속)

매개 변수	설명
local.store.overquotastatus	메시지가 MTA의 대기열에 포함되기 전에 할당량 적용을 활성화합니다. 그렇게 하면 MTA 대기열이 가득 차지 않습니다. 설정하는 경우 사용자가 아직 할당량을 초과하지 않았지만 받는 메시지로 인해 사용자가 할당량을 초과하게 되면 메시지가 전달되지만 mailuserstatus LDAP 속성이 overquota로 설정되므로 MTA에서 더 이상의 메시지를 수락하지 않습니다. 기본값: off입니다.

메시지 저장소 할당량에는 두 개의 유틸리티도 포함되어 있습니다. **Sun Java System Messaging Server 6.3 Administration Reference**의 "iminitquota"는 할당량 설정을 초기화합니다. 즉, 할당량 속성과 configutil 매개 변수는 이 명령을 실행하고 나면 자동으로 적용됩니다. 이 명령을 실행하지 않아도 변경 사항이 적용되기는 하지만, 정보가 캐시에 저장된 후 실제로 적용될 때까지 약간의 시간이 걸리기 때문에 즉시 적용되지는 않습니다.

imquotacheck 유틸리티를 사용하면 지정된 할당량을 기준으로 메시지 저장소 사용량을 확인할 수 있습니다.

20.8.4 메시지 저장소 할당량 구성

이 절에서는 다음 작업에 대해 설명합니다.

20.8.4.1 기본 사용자 할당량 지정

기본 할당량은 LDAP 항목에 개별 할당량이 설정되지 않은 사용자에게 적용됩니다. 이 프로세스는 두 단계로 구성됩니다. 1) 사용자 기본 할당량을 지정한 다음 2) 기본 할당량에 바인드되는 사용자를 지정합니다. 다음 예에서는 기본 사용자 할당량을 설정하는 방법을 보여 줍니다. 자세한 매개 변수 정보는 **Sun Java System Messaging Server 6.3 Administration Reference**의 3 장, "Messaging Server Configuration"을 참조하십시오.

메시지 크기에 대한 기본 사용자 디스크 할당량(바이트)을 지정하려면:

```
configutil -o store.defaultmailboxquota -v [ -1 | number ]
```

여기서 -1은 할당량이 없는 것을 나타내고(메시지 사용 제한 없음) *number*는 바이트 수를 나타냅니다.

전체 메시지 수에 대한 기본 사용자 할당량을 지정하려면:

```
configutil -o store.defaultmessagequota -v [ -1 | number ]
```

여기서 -1은 할당량이 없는 것을 나타내고(메시지 제한 없음) *number*는 메시지 수를 나타냅니다.

특정 사용자의 기본 할당량을 지정하려면:

기본 메시지 저장소 할당량을 사용하는 사용자 항목에서 mailQuota 속성을 -2로 설정합니다. mailQuota가 지정되어 있지 않으면 시스템 기본 할당량이 사용됩니다.

20.8.4.2 개별 사용자 할당량 지정

각 사용자에 개별 할당량을 지정할 수 있습니다. 사용자별 할당량을 설정하려면 사용자의 LDAP 항목에 **Sun Java Communications Suite 5 Schema Reference**의 "mailQuota" 또는 **Sun Java Communications Suite 5 Schema Reference**의 "mailMsgQuota" 속성을 설정합니다(자세한 내용은 **Sun Java System Messaging Server 6.3 Administration Reference**의 "configutil Parameters" 참조). 다음 예에서는 사용자 할당량을 설정하는 방법을 보여 줍니다.

시스템 기본 할당량을 지정하려면 LDAP 항목에 mailQuota를 추가하거나 –2로 설정하지 마십시오.

할당량을 1,000개의 메시지로 설정하려면 mailMsgQuota를 1000으로 설정합니다.

할당량을 2MB로 설정하려면 mailQuota를 2M 또는 2000000으로 설정합니다.

할당량을 2GB로 설정하려면 mailQuota를 2G 또는 2000000000 또는 2000M으로 설정합니다.

할당량을 2GB로, 음성 메일 할당량을 20MB로, Archive 폴더 할당량을 100MB로 지정하려면 다음을 수행합니다.

```
mailQuota: 2G;#voice%20M;Archive%100M
```

2GB의 할당량은 사용자의 메일함에서 명시적으로 할당량이 지정되지 않은 모든 폴더를 나타냅니다. 이 예에서는 Archive 폴더에 있는 메시지와 voice 유형의 메시지가 제외됩니다. 100MB의 할당량에는 Archive 폴더에 속한 모든 폴더의 메시지가 포함됩니다.

20.8.4.3 도메인 할당량 지정

도메인에 디스크 공간 또는 메시지 할당량을 설정할 수 있습니다. 이 할당량은 특정 도메인에 있는 모든 사용자에 대한 바이트 또는 메시지 수를 모두 누적한 값입니다. 도메인 할당량을 설정하려면 원하는 LDAP 도메인 항목에서 **Sun Java Communications Suite 5 Schema Reference**의 "mailDomainDiskQuota" 또는 **Sun Java Communications Suite 5 Schema Reference**의 "mailDomainMsgQuota" 속성을 설정합니다..

할당량을 1,000개의 메시지로 설정하려면 mailDomainMsgQuota를 1000으로 설정합니다.

할당량을 2MB로 설정하려면 mailDomainDiskQuota를 2M 또는 2000000으로 설정합니다.

할당량을 2GB로 설정하려면 mailDomainDiskQuota를 2G 또는 2000000000 또는 2000M으로 설정합니다.

▼ 할당량 알림 설정 방법

할당량 알림은 사용자가 할당량에 가까와질 때 경고 메시지를 보내는 프로세스입니다. 이 기능을 사용하려면 세 가지 단계를 수행해야 합니다.

1 할당량 알림 활성화

명령줄에서 다음을 실행합니다.

```
configutil -o store.quotanotification -v [ yes | no ]
```

메시지가 설정되지 않은 경우 할당량 경고 메시지가 사용자에게 보내지지 않습니다.

2 할당량 경고 메시지 정의

경고 메시지는 디스크 할당량을 초과하려 하는 사용자에게 전송되는 메시지입니다. 명령줄에서 할당량 경고 메시지를 정의하려면 다음을 수행합니다.

```
configutil -o store.quotaexceededmsg -v 'message'
```

메시지는 RFC 822 형식이어야 합니다. 즉, 메시지는 최소한 제목 줄이 들어있는 헤더를 포함하고 그 뒤에 $$와 메시지 본문이 와야 합니다. '$'는 새 행을 나타냅니다. 사용 중인 쉘에 따라 $의 특수한 의미를 이스케이프하기 위해 $ 앞에 \를 추가해야 할 수 있습니다. $는 일반적으로 해당 쉘의 이스케이프 문자입니다. 예:

```
configutil -o store.quotaexceededmsg -v   Subject: WARNING: User quota
exceeded$$User quota threshold exceeded - reduce space used.'
```

또한 다음 변수가 지원됩니다.

[ID] - 사용자 아이디

[DISKUSAGE] - 디스크 사용

[NUMMSG] - 메시지 수

[PERCENT] - store.quotawarn 비율

[QUOTA] - mailquota 속성

[MSGQUOTA] - mailmsgquota 속성

다음은 이러한 변수를 사용한 예입니다.

```
configutil -o store.quotaexceededmsg -v   Subject: Overquota
Warning$$[ID],$$Your mailbox size has exceeded [PERCENT] of its alloted
```

```
quota.$Disk Usage: [DISKUSAGE]$Number of Messages: [NUMMSG]$Mailquota:
[QUOTA]$Message Quota: [MSGQUOTA]$$-Postmaster'
```

3 경고 메시지를 보내는 빈도 지정

다음 매개 변수를 설정합니다.

```
configutil -o store.quotaexceededmsginterval -v number
```

여기서 *number*는 일 수를 나타냅니다. 예를 들어, 3은 메시지가 3일마다 보내진다는 것을 의미합니다.

4 할당량 임계값 지정

할당량 임계값은 클라이언트에서 경고를 보내기 전에 초과되는 할당량의 비율입니다. 사용자의 디스크 사용량이 지정된 임계값을 초과하면 서버에서 사용자에게 경고 메시지를 보냅니다.

주 – `local.store.quotaoverdraft=on`이면 `store.quotawarn`으로 설정한 임계값에 상관없이 사용자의 디스크 사용량이 할당량의 100%를 초과할 때까지 전자 메일 알림이 트리거되지 않습니다.

클라이언트가 IMAP ALERT 기법을 지원하는 IMAP 사용자의 경우 사용자가 메일함을 선택할 때마다 사용자의 화면에 메시지가 표시되고 IMAP 로그에 메시지가 기록됩니다.

명령줄에서 할당량 임계값을 지정하려면 다음을 수행합니다.

```
configutil -o store.quotawarn -v number
```

여기서 *number*는 허용되는 할당량의 비율을 나타냅니다.

20.8.4.4 할당량 적용 활성화 또는 비활성화

기본적으로 사용자 또는 도메인은 할당량 초과 알림을 받는 것 외에(설정된 경우) 다른 효과 없이 할당량을 초과할 수 있습니다. 할당량 적용은 디스크 사용량이 할당량 수준 이하로 떨어질 때까지 추가 메시지를 받지 않도록 메일함을 잠급니다.

할당량 적용을 활성화 또는 비활성화하려면 다음을 수행합니다.

```
configutil -o store.quotaenforcement -v [ on | off]
```

할당량 초과 메시지가 MTA 대기열에 저장되고 보낸 사람에게 메시지가 배달되지 않았지만 나중에 다시 배달 시도가 있을 것임을 나타내는 알림이 전송됩니다. 유예 기간이 만료되어 모든 메시지가 보낸 사람에게 되돌아가거나, 디스크 사용량이 할당량 아래로 떨어지고 메시지가 MTA 대기열에서 제외되고 메시지 저장소로 배달될 수 있을 때까지 배달 재시도가 계속됩니다. 메시지 대기열로 보내기 전에 할당량을 초과한 메시지를 돌려보내려면 다음 명령줄을 사용합니다.

```
configutil -o store.overquotastatus -v on
```

도메인 수준에서 할당량 적용을 활성화하는 방법

특정 도메인에 할당량을 적용하려면 다음 명령을 사용합니다.

```
imquotacheck -f -d domain
```

모든 도메인에 대해 사용하려면 `-d` 옵션을 제외합니다. 도메인이 할당량을 초과하면 `maildomainstatus` 속성이 `overquota`로 설정되어 해당 도메인에 대한 모든 전달이 중지됩니다. 도메인이 `overquota`가 아닌 경우 값은 `active`로 설정됩니다.

할당량 적용 비활성화

할당량을 비활성화했지만 사용자 할당량이 적용되는 것으로 표시되는 경우 다음 매개변수를 확인합니다.

다음 `configutil` 매개 변수가 off이거나 설정되어 있지 않아야 합니다.

- `store.quotaenforcement`
- `local.store.overquotastatus`
- `local.store.quotaoverdraft`

`store.overquotastatus`가 `on`일 경우 `store.quotaoverdraft`는 항상 `on`인 것으로 간주됩니다. 그렇지 않을 경우 사용자가 할당량을 초과하여 거부를 트리거하는 일이 없습니다. 또한 `store.quotaoverdraft`가 `on`이면 할당량보다 작은 하나의 메시지만 사용자에게 허용됩니다. 즉, 사용자의 할당량보다 큰 메시지는 허용되지 않습니다.

이 매개 변수를 변경한 후에는 메시징 서비스를 다시 시작해야 합니다.

다음 메시지 저장소 속성을 활성화해야 합니다.

- `maildomainstatus`
- `mailuserstatus`

할당량 적용 구성에 상관없이 메일함 할당량보다 큰 메시지는 바운스됩니다.

20.8.4.5 유예 기간 설정

유예 기간은 메시지를 보낸 사람에게 다시 바운스하기 전까지 메일함이 할당량(디스크 공간 또는 메시지 수)을 초과할 수 있는 기간을 지정합니다. 유예 기간은 메시지가 메시지 대기열에 보관되는 기간이 아니라 메시지 대기열에 있는 메시지를 비롯하여 모든 받는 메시지가 바운스되기 전까지 메일함이 할당량을 초과할 수 있는 기간입니다. 자세한 내용은 555 페이지 “20.1 개요”를 참조하십시오. 유예 기간은 사용자가 할당량 임계값에 도달하여 경고를 받으면 시작됩니다. 589 페이지 “할당량 알림 설정 방법”을 참조하십시오.

명령줄에서 할당량 유예 기간을 지정하려면 다음을 수행합니다.

```
configutil -o store.quotagraceperiod -v number
```

여기서 *number*는 시간을 나타냅니다.

20.8.4.6 Netscape Messaging Server 할당량 호환 모드

Netscape Messaging Server에서는 디스크 사용량이 할당량을 초과한 경우 메시지 전달을 지연 또는 바운스하고 할당량 초과 알림을 보낸 다음 유예 기간을 시작했습니다. Messaging Server는 이 동작을 유지하는 `local.store.quotaoverdraft` 매개 변수를 제공합니다.

`ON`으로 설정하면 디스크 사용량이 할당량을 초과할 때까지 메시지가 전달됩니다. 할당량을 초과하면 메시지가 지연되고(메시지는 MTA 메시지 대기열에 보관되지만 메시지 저장소로 전달되지 않음) 할당량 초과 경고 메시지가 사용자에게 보내지며 유예 기간이 시작됩니다. 유예 기간은 할당량 초과 메시지가 바운스될 때까지 메일함의 할당량이 초과되어 있는 기간을 결정합니다. 기본값은 메시지 저장소가 임계값에 도달할 때 할당량 경고 메시지를 보냅니다. 이 매개 변수의 기본값은 `Off`입니다.

20.9 자동 메시지 제거(만료 및 제거) 기능 설정 방법

자동 메시지 제거 기능(만료 및 제거라고도 함)은 관리자가 정의한 일련의 기준을 기반으로 메시지 저장소에서 메시지를 자동으로 제거합니다. 이전 메시지가나 과도하게 큰 메시지, 보았거나 삭제한 메시지, 특정 Subject: 행을 가진 메시지 등을 자동으로 제거하는 데이 기능을 사용할 수 있습니다. 이 기능은 다음 제거 기준을 허용합니다.

- 폴더(메일함), 사용자, 도메인, 전체 메시지 저장소 또는 특정 분할 영역별
- 메일함의 메시지 수
- 메일함의 총 크기
- 메시지가 메일함에 남아 있는 기간(일)
- 메시지 크기와 유예 기간(크기를 초과한 메시지가 제거되기 전에 메시지 저장소에 남아 있는 일 수)
- 메시지에 **조회** 또는 **삭제됨** 플래그가 지정되었는지 여부
- 헤더 문자열

이 기능은 메시지를 정리 및 제거하는 `imexpire` 유틸리티에 의해 수행됩니다. 메시지 제거 프로세스에 대한 자세한 내용은 561 페이지 “20.3 메시지 저장소에서 메시지를 제거하는 방법”을 참조하십시오.

주 – 서버는 경고 없이 메시지를 제거하므로 자동 메시지 제거 정책을 사용자에게 알리는 것이 중요합니다. 메시지가 갑작스럽게 제거되면 사용자와 관리자가 매우 당황할 수 있습니다.

- 593 페이지 “20.9.1 imexpire 작동 원리”
- 593 페이지 “20.9.2 자동 메시지 제거 기능 배포”

20.9.1 imexpire 작동 원리

imexpire는 명령줄에서 호출하거나 imsched 데몬에 의해 자동으로 실행되도록 예약할 수 있습니다. 관리자는 store.expirerule 이라는 파일에 만료 규칙 집합을 지정합니다. 이 파일은 메시지 제거 기준을 지정합니다. 각각 규칙 범위와 관련된 디렉토리에 저장된 여러 개의 파일이 있을 수 있습니다. 즉, 전체 메시지 저장소에 전역적으로 적용되는 규칙과 특정 분할 영역에 적용되는 규칙, 그리고 사용자에 적용되는 규칙이 각각 다른 디렉토리에 저장됩니다.

주 – configutil 명령과 store.expire.*attribute* 매개 변수를 사용하여 전역 만료 규칙을 지정할 수도 있지만 store.expirerule을 사용하여 규칙을 지정하는 것이 더 좋습니다. configutil을 사용하여 너무 많은 규칙을 만들면 성능 문제가 발생할 수 있습니다.

---

imexpire는 시작 시에 모든 만료 규칙을 로드합니다. 기본적으로 imexpire는 분할 영역당 하나의 스레드를 만듭니다. 각 스레드는 할당된 분할 영역 아래의 사용자 폴더 목록을 거치는 과정에서 로컬 만료 규칙 파일을 로드합니다. 이 만료 기능은 각 폴더에 적용 가능한 만료 규칙에 대해 해당 폴더를 검사하고 필요에 따라 메시지를 정리합니다. 메일함 디렉토리에 store.exp 파일이 있고 store.cleanupage 구성 매개 변수에 지정된 시간보다 오래 정리/만료된 메시지가 있을 경우 제거 기능은 메시지 해시 디렉토리의 모든 메시지 파일을 영구적으로 제거하고 store.exp 파일에서 UID 레코드를 제거합니다.

또한 *msg-svr-base*/config/에 있는 expire_exclude_list라는 파일에 한 행씩 사용자 아이디를 추가하여 지정된 사용자를 만료 규칙에서 제외할 수 있습니다.

20.9.2 자동 메시지 제거 기능 배포

자동 메시지 제거는 다음 세 단계로 구성됩니다.

1. 자동 메시지 제거 정책을 정의합니다. 자동으로 제거할 메시지는 무엇입니까? 메시지가 자동으로 제거될 사용자, 폴더, 도메인 및 분할 영역은 무엇입니까? 제거 기준을 정의하는 크기, 메시지 기간 및 헤더는 무엇입니까? 제거할 메시지 범위를 정의합니다. 593 페이지 “20.9.2.1 자동 메시지 제거 정책 정의”을 참조하십시오.
2. 이 정책을 구현하기 위한 imexpire 규칙을 지정합니다. 594 페이지 “20.9.2.2 자동 메시지 제거 정책을 구현하는 규칙 설정 방법”을 참조하십시오.
3. imexpire 일정을 지정합니다. 600 페이지 “20.9.2.3 자동 메시지 제거 및 로깅 수준 예약”을 참조하십시오.

20.9.2.1 자동 메시지 제거 정책 정의

제거 기준을 지정하여 자동 메시지 제거 정책을 정의합니다. imexpire에서는 다음 기준을 제거에 사용할 수 있습니다.

메시지 기간. *X*일보다 오래된 메시지를 자동으로 제거합니다(속성: messagedays).

메시지 수. X개의 메시지를 초과하는 폴더의 메시지를 자동으로 제거합니다(속성: `messagecount`).

크기를 초과하는 메시지의 기간. Y일의 유예 기간 후에 X바이트를 초과한 메시지를 자동으로 제거합니다(속성: `messagesize` 및 `messagesizedays`).

조회 및 삭제됨 메시지 플래그. 조회 또는 **삭제됨** 플래그가 설정된 메시지를 자동으로 제거합니다. 이 기준은 `"and"` 및 `"or"`로 설정할 수 있습니다. `or`로 설정된 경우 메시지의 조회/삭제 플래그는 다른 기준에 상관 없이 메시지를 자동으로 삭제합니다. `and`로 설정된 경우 다른 모든 기준을 충족하면서 메시지의 조회/삭제 플래그를 설정해야 합니다. (속성: `seen` 및 `deleted`).

메시지의 헤더 필드. 메시지 제거 기준으로 사용할 헤더와 문자열을 지정할 수 있습니다(예: "Subject: Work from Home!").

메시지 폴더. 메시지를 제거할 폴더를 지정할 수 있습니다(속성: `folderpattern`). 이 속성은 수정된 UTF-7 문자 세트만 사용합니다.

주 – `imexpire`에서는 메시지를 읽은 후로 경과한 시간에 따라 메시지를 삭제하거나 보존하도록 허용하지 않습니다. 예를 들어, 200일 동안 읽지 않은 메시지를 제거하도록 지정할 수 없습니다.

자동 메시지 제거 정책의 예

예 1: 1,000개의 메시지를 초과하는 폴더에서 365일이 지난 모든 메시지를 제거합니다.

예 2: siroe.com 도메인에서 180일이 지난 메시지를 제거합니다.

예 3: **삭제됨**으로 표시된 모든 메시지를 제거합니다.

예 4: **조회** 표시가 있고 30일이 지났으며 100KB보다 크고 폴더의 메시지 수가 1,000개를 초과하며 `X-spam` 헤더가 있는 메시지를 sesta.com에서 제거합니다.

20.9.2.2 자동 메시지 제거 정책을 구현하는 규칙 설정 방법

이전 절에서 정의한 자동 메시지 제거 정책을 구현하려면 `imexpire` 규칙을 설정해야 합니다. `store.expirerule` 파일에 규칙을 포함하면 규칙이 설정됩니다. 다음은 두 개의 전역 `store.expirerule` 규칙을 보여 주는 예입니다.

```
Rule1.regexp: 1
Rule1.folderpattern: user/.*/trash
Rule1.messagedays: 2
Rule2:regexp: 1
Rule2.folderpattern: user/.*
Rule2.messagedays: 14
```

이 예에서 Rule 1은 휴지통 폴더의 모든 메시지가 2일 후에 제거되도록 지정합니다. Rule 2는 메시지 저장소의 모든 메시지가 14일 후에 제거되도록 지정합니다.

이 절은 다음과 같은 하위 절로 구성되어 있습니다.

만료 규칙 지침

이 절에서는 `store.expirerule` 파일 규칙에 대한 지침을 설정합니다.

주 – 이전의 Messaging Server 릴리스에서는 `configutil` 매개 변수 `store.expirerule.`*attribute*를 사용하여 만료 규칙을 설정할 수 있었습니다(**Sun Java System Messaging Server 6.3 Administration Reference**의 "configutil Parameters" 참조).이는 여전히 유효하지만 헤더 제약 조건을 사용하는 만료 규칙(예: 특정 제목 줄을 가진 메시지를 만료하는 것)은 지원되지 않습니다. 어떤 경우에서든 `store.expirerule`을 사용하여 모든 만료 규칙을 지정하는 것이 가장 좋습니다.

- 규칙은 `store.expirerule`이라는 파일에 지정됩니다.
- 동일한 규칙을 사용하여 여러 만료 기준을 지정할 수 있습니다(위의 예를 참조).
- 규칙은 전체 메시지 저장소(전역 규칙), 메시지 저장소 분할 영역, 사용자 또는 폴더에 적용할 수 있습니다.
  - 전역 규칙은 *msg-svr-base*`/config/store.expirerule`에 저장됩니다.

    주 – 각 전역 규칙을 모든 메일함에 대해 검사하므로 지정한 전역 규칙의 수에 따라 처리 오버헤드가 발생할 수 있습니다. 이런 이유로 인해 전역 규칙 파일에서 분할 영역, 메일함 또는 사용자 규칙을 지정해서는 안 됩니다. 일반적으로 필요한 수보다 많은 만료 규칙을 이 파일에 포함시키지 않도록 해야 합니다.

  - 분할 영역 규칙은 *store_root*`/partition/`*partition_name*`/store.expirerule` 에 저장됩니다.
  - 사용자 규칙은 *store_root*`/partition/`*partition_name*`/`*userid*`/store.expirerule`에서 지정하거나 `folderpattern` 규칙이 `user/`*userid*`/.*`가 되도록 지정하여 만들 수 있습니다.
  - 폴더 규칙은 *store_root*`/partition/`*partition_name*`/`*userid*`/`*folder*`/store.expirerule`에서 지정하거나 `folderpattern` 규칙이 `user/`*userid*`/`*folder*가 되도록 지정하여 만들 수 있습니다.
  - *rule_name*을 사용하는 여러 개의 비전역 규칙(사용자, 폴더, 분할 영역)은 Messaging Server 6.2p4 릴리스 이후에만 구현되었습니다.

- 여러 만료 규칙을 메일함에 동시에 적용할 수 있습니다. 메일함에 대한 만료 규칙은 전역 규칙 및 로컬 규칙으로 구성됩니다. 로컬 규칙은 같은 디렉토리와 해당하는 모든 하위 폴더의 메일함에 적용됩니다.
- `imexpire`는 메일함에 대해 지정된 배타적 규칙이 없을 경우 메일함에 적용되는 모든 만료 규칙을 통합합니다(표 20–8 참조). 결과 규칙 집합은 적용 가능한 모든 규칙을 기반으로 하는 가장 제한적인 만료 정책을 나타냅니다. 예를 들어, 메시지가 만료되는 최대 메시지 기간을 규칙 X가 10일로 지정하고 규칙 Y가 5일로 지정할 경우 통합 결과는 5일이 됩니다.

표 20–8 `imexpire` 속성

속성	설명(속성 값)
`action`	만료 규칙에 걸린 메시지에서 수행할 작업을 지정합니다. 가능한 값은 다음과 같습니다. `discard`는 메시지를 삭제합니다. 기본값입니다. `report` 작업은 메일함 이름과 uid-validity 및 uid를 stdout에 출력합니다. `archive`는 Sun Compliance 및 Content Management System에 메시지를 아카이브로 보관한 다음 메시지를 삭제합니다. `fileinto:` *folder* 작업은 메시지를 특정 폴더에 넣습니다. 공유 폴더 접두사를 사용하면 메시지를 다른 사용자가 소유한 폴더에 넣을 수 있습니다.
`exclusive`	해당 규칙이 배타적인지 여부를 지정합니다. `exclusive`로 지정된 경우 해당 규칙만 지정된 메일함에 적용되며 다른 모든 규칙은 무시됩니다. 둘 이상의 배타적인 규칙이 존재할 경우 마지막으로 로드된 규칙이 사용됩니다. 예를 들어, 전역 및 로컬 배타적인 규칙을 지정할 경우 로컬 규칙이 사용됩니다. 둘 이상의 배타적인 전역 규칙이 있을 경우 `configutil`에서 나열한 마지막 전역 규칙이 사용됩니다. (`1`/`0`)
`folderpattern`	해당 규칙의 영향을 받는 폴더를 지정합니다. 형식은 *store_root* `/partition/*/` 디렉토리를 나타내는 `user/`로 시작해야 합니다. 표 20–9를 참조하십시오. (POSIX 정규 표현식).
`messagecount`	폴더의 최대 메시지 수입니다. 추가 메시지가 전달되면 가장 오래된 메시지가 정리됩니다. (정수)
`foldersize`	추가 메시지가 전달되었을 때 가장 오래된 메시지가 정리되기 전까지의 최대 폴더 크기입니다. (바이트 단위 정수)
`messagedays`	메시지가 정리되기 전까지의 메시지 기간(일)입니다. (정수)
`messagesize`	메시지가 정리되는 것으로 표시되기 전까지의 메시지의 최대 크기(바이트)입니다. (정수)
`messagesizedays`	유예 기간입니다. 크기를 초과한 메시지가 폴더에 남아 있어야 하는 일 수입니다. (정수)
`messageheader.`*header*	제거할 메시지를 표시하는 헤더 필드와 문자열을 지정합니다. 값은 대소문자를 구분하지 않으며 정규 표현식은 인식되지 않습니다. 예: `Rule1.messageheader.Subject: Get Rich Now!` *Expires* 및 *Expiry-Date* 헤더의 경우 `imexpire`는 이러한 헤더 필드로 지정한 날짜 값이 `messagedays` 속성보다 오래 되었으면 메시지를 제거합니다. 여러 개의 만료 헤더 필드를 지정한 경우에는 가장 이른 만료 날짜가 사용됩니다. (문자열)

표 20–8 imexpire 속성 (계속)

속성	설명(속성 값)
regexp	규칙 작성에 UNIX 정규 표현식을 사용 가능하게 합니다. (1 또는 0) 지정하지 않으면 IMAP 표현식이 사용됩니다.
savedays	메시지가 정리될 때까지 폴더에 저장되는 시간(일 수)입니다.
seen	seen은 사용자가 메시지를 열었을 때 시스템에 의해 설정되는 메시지 상태 플래그입니다. seen 속성이 and로 설정된 경우 메시지를 보는 것 **외에도** 다른 기준을 충족해야 규칙이 적용됩니다. seen 속성이 or로 설정된 경우 메시지를 보았거나 **또는** 다른 기준을 충족하면 규칙이 적용됩니다. (and/or)
sieve	메시지 선택 기준을 지정하는 시브(Sieve) 규칙입니다. 예: Rule17.sieve: header :contains Subject Vigara
deleted	deleted는 사용자가 메시지를 삭제했을 때 시스템에 의해 설정되는 메시지 상태 플래그입니다. deleted 속성이 and로 설정된 경우 메시지를 삭제한 것 **외에도** 다른 기준을 충족해야 규칙이 적용됩니다. deleted 속성이 or로 설정된 경우 메시지를 삭제했거나 **또는** 다른 기준을 충족하면 규칙이 적용됩니다. (**and/or**)

현지화된 메일함 이름

IMAP 프로토콜은 메일함 이름에 수정된 UTF-7 인코딩을 사용하도록 지정합니다. Messaging Server는 메일함 이름을 현지화할 수 있도록 외부 인터페이스에서 현지화된 문자 세트를 지원합니다. 하지만 내부에서 시스템은 현지화된 이름을 UTF-7로 변환합니다. 따라서 클라이언트에 현지화된 메일함 이름이 있는 폴더는 그에 해당되는 UTF-7로 된 메일함 파일 이름을 가집니다(IMAP 오류 메시지에서는 메일함 이름을 현지화된 문자 세트가 아닌 UTF-7로 출력)

일반적으로 메일함 이름이 필요한 대부분의 메시지 저장소 유틸리티는 현지화된 문제 세트로 된 이름을 사용하지만 다른 문자 세트를 사용할 수 있는 옵션 플래그가 있을 수도 있습니다. 이러한 유틸리티에는 reconstruct, mboxutil, imsbackup, imsrestore 및 hashdir가 있습니다. 하지만 imexpire는 folderpattern 속성으로 지정되는 메일함 이름이 UTF-7로 되어 있어야 합니다. 현지화된 이름은 사용할 수 없습니다.

imexpire에 적절한 folderpattern을 얻으려면 현지화된 메일함 이름을 수정된 UTF-7 이름으로 변환해야 할 수도 있습니다. mboxutil -E 명령을 사용하여 다음과 같이 수행하면 됩니다.

```
$ mboxutil -l -p user/user1/*
msgs  Kbytes  last msg        partition  quotaroot  mailbox
|
77     27     2006/9/9 3:21   primary    10240      user/kat/INBOX
0      0      -               primary               user/kat/箱

$ mboxutil -l -E MUTF-7 -p user/user1/*
msgs  Kbytes  last msg        partition  quotaroot  mailbox

77     27     2006/9/9 3:21   primary    10240      user/kat/INBOX
0      0      -               primary               user/kat/&V4NXPnux-
```

첫 번째 `mboxutil`은 현지화된 파일 이름을 나타냅니다. 두 번째 `mboxutil`은 수정된 UTF-7의 파일 이름을 나타냅니다. IMAP list 명령을 사용할 수도 있습니다.

```
2 list "" *
* LIST (\NoInferiors) "/" INBOX
* LIST (\HasNoChildren) "/" &V4NXPnux-
```

텍스트 형식으로 imexpire 규칙 설정

자동 메시지 제거 규칙은 `store.expirerule` 파일에서 규칙을 지정하여 설정합니다. `store.expirerule` 파일에는 한 행에 하나씩 만료 기준이 있습니다. 전역 규칙 구성 파일(*msg-svr-base*`/data/store/store.expirerule`)의 만료 기준 형식은 다음과 같습니다.

rule_name.*attribute*: *value*

사용자 또는 메일함 규칙 구성 파일의 만료 규칙 형식은 다음과 같습니다.

attribute: *value*

예 20–4에서는 *msg-svr-base*`/config/store.expirerule`의 전역 만료 규칙 집합을 보여 줍니다.

Rule 1은 다음과 같이 전역 만료 정책(즉, 모든 메시지에 적용되는 정책)을 설정합니다.

- 규칙 작성에 UNIX 정규 표현식을 사용 가능하게 합니다.
- 3일이 지난 100,000바이트를 초과하는 메시지를 제거합니다.
- 사용자가 삭제한 메시지를 제거합니다.
- Subject: 헤더에 "Vigara Now!" 또는 "XXX Porn!" 문자열이 있는 메시지를 제거합니다.
- 모든 폴더의 메시지 수를 1,000개로 제한합니다. 메시지가 1,000개를 넘으면 시스템은 폴더에서 가장 오래된 메시지를 제거하여 합계를 1,000개로 유지합니다.

- 365일이 지난 모든 메시지를 제거합니다.

Rule 2는 호스트된 도메인 `siroe.com`에서 사용자에 대한 자동 메시지 제거 정책을 설정합니다. 이 규칙은 메일함 크기를 1MB로 제한하고 삭제된 메시지를 제거하며 14일이 지난 메시지를 제거합니다.

Rule 3은 `f.dostoevski` 사용자의 `inbox` 폴더에 있는 메시지에 대한 자동 메시지 제거 정책을 설정합니다. 이 규칙은 제목 줄에 "On-line Casino" 라는 표현이 있는 메시지를 제거합니다.

예 20–4 `imexpire` 규칙 예

```
Rule1.regexp: 1
Rule1.folderpattern: user/.*
Rule1.messagesize: 100000
Rule1.messagesizedays: 3
Rule1.deleted: or
Rule1.Subject: Vigara Now!
Rule1.Subject: XXX Porn!
Rule1.messagecount: 1000
Rule1.messagedays: 365
Rule2.regexp: 1
Rule2.folderpattern: user/.*@siroe.com/.*Rule2.exclusive: 1
Rule2.deleted: or
Rule2.messagedays: 14
Rule2.messagecount: 1000
Rule3.folderpattern: user/f.dostoevski/inboxRule3.Subject: *On-line Casino*
```

imexpire 폴더 패턴 설정

POSIX 정규 표현식을 사용하여 `imexpire` 속성 `regex`를 1로 설정함으로써 폴더 패턴을 지정할 수있습니다. 지정하지 않으면 IMAP 표현식이 사용됩니다. `user/`로 시작되고 뒤에 패턴이 나오는 형식이어야 합니다. 표 20–9에서는 다양한 폴더의 폴더 패턴을 보여줍니다)

표 20–9 정규 표현식을 사용한 `imexpire` 폴더 패턴

폴더 패턴	범위
`user/`*`userid`*`/.*`	*userid*의 모든 폴더에 있는 모든 메시지에 적용됩니다.
`user/`*`userid`*`/Sent`	`Sent` 폴더에 있는 *userid*의 메시지에 규칙을 적용합니다.
`user/.*`	메시지 저장소 전체에 규칙을 적용합니다.
`user/.*/trash`	모든 사용자의 `trash` 폴더에 규칙을 적용합니다.

표 20–9 정규 표현식을 사용한 imexpire 폴더 패턴 *(계속)*

폴더 패턴	범위
user/.*@siroe.com/.*	호스트된 도메인 siroe.com의 폴더에 규칙을 적용합니다.
user/[^@]*/.*	기본 도메인에 있는 폴더에 규칙을 적용합니다.

20.9.2.3 자동 메시지 제거 및 로깅 수준 예약

자동 메시지 제거는 imsched 예약 데몬에 의해 활성화됩니다. 기본적으로 imsched는 매일 23시에 imexpire를 호출하여 메시지를 정리 및 제거합니다. 이 일정은 표 20–10에 설명된 configutil 매개 변수 local.schedule.expire 및 store.cleanupage를 설정하여 사용자 정의할 수 있습니다.

메시지 저장소가 큰 경우 만료와 제거를 완료하는 데 시간이 걸릴 수 있으므로 이러한 프로세스를 실행하는 빈도를 실험하여 결정하는 것이 필요할 수 있습니다. 예를 들어, 만료/제거 주기가 10시간일 경우 만료 및 제거를 하루에 한 번씩 실행하도록 기본 일정을 세우지는 않을 것입니다. 일정은 imexpire 명령 및 자동 작업 예약 매개 변수를 사용하여 만료 및 제거됩니다(109 페이지 “4.6 자동 작업 예약” 참조). 예를 들면 다음과 같습니다.

```
configutil -o local.schedule.expire -v "0 1 * * 6 /opt/SUNWmsgsr/sbin/imexpire -e"
configutil -o local.schedule.mspurge -v "0 23 * * * /opt/SUNWmsgsr/sbin/imexpire -c"
```

이 예에서 메시지는 토요일 오전 1시에 만료되고 매일 밤 11시에 제거됩니다. 제거 일정이 설정되어 있지 않으면 imexpire에서 만료 후에 제거를 수행합니다.

표 20–10 만료 및 제거 configutil 로그 및 예약 매개 변수

매개 변수	설명

local.schedule.expire	imexpire를 실행하는 간격입니다. 다음 UNIX crontab 형식을 사용합니다. *minute hour day-of-month month-of-year day-of-week*
	
	값은 공백이나 탭으로 구분하며 각각 0-59, 0-23, 1-31, 1-12 및 0-6(0=일요일)의 값을 사용할 수 있습니다. 각 시간 필드에는 별표(유효한 모든 값), 쉼표로 구분된 값 목록 또는 하이픈으로 구분된 두 값의 범위를 사용할 수 있습니다. 날짜는 일과 요일 모두를 사용하여 지정할 수 있지만 둘 다 일치하는 경우가 매우 드물기 때문에 일반적이지 않습니다. 일과 요일을 모두 지정한 경우에는 둘 다 필요합니다. 예를 들어, 17일과 화요일을 설정하면 두 값이 모두 true가 되어야 합니다.
	
	imexpire에 -e 및 -c 플래그를 사용하여 각각 만료만 하도록 또는 제거만 하도록 할 수도 있습니다. **Sun Java System Messaging Server 6.3 Administration Reference**의 "imexpire"를 참조하십시오.
	
	간격 예:
	
	1) 오전 12:30 , 8:30 및 오후 4:30에 imexpire를 실행합니다.
	
	`30 0,8,16 * * * /opt/SUNWmsgsr/sbin/imexpire`
	
	2) 주중 아침 3:15에 imexpire를 실행합니다.
	
	`15 3 * * 1-5 /opt/SUNWmsgsr/sbin/imexpire`
	
	3) 월요일에만 imexpire를 실행합니다.
	
	`0 0 * * 1 /opt/SUNWmsgsr/sbin/imexpire`
	
	기본값:
	
	`0 23 * * * /opt/SUNWmsgsr/sbin/imexpire`
	
	비활성화하려면local.schedule.expire.enable을 NO로 설정합니다.

store.cleanupage	만료 또는 정리된 메시지가 purge에 의해 영구적으로 제거된 전까지의 기간(시간)입니다. 기본값: 없음
local.store.expire.loglevel	다음과 같이 로그 수준을 지정합니다. 1 = 전체 만료 세션의 요약을 기록합니다. 2 = 만료된 메일함별로 하나씩 메시지를 기록합니다. 3 = 만료된 메일별로 하나씩 메시지를 기록합니다. 기본값: 1

imexpire 로깅 수준 설정

imexpire는 완료 시에 기본 로그 파일에 대한 요약을 기록합니다. 명령줄에서 만료가 호출될 경우 -v(verbose) 및 -d(debug) 옵션을 사용하여 자세한 상태/디버그 메시지를 stderr에 기록하도록 imexpire에 지시할 수 있습니다. imexpire가 imsched에 의해 호출될 경우 configutil 매개 변수 local.store.expire.loglevel을 여러 다른 로깅 수준에 대해 1,2 또는 3으로 설정할 수 있습니다. Loglevel 1은 기본값으로 전체 만료 세션의 요약을 기록합니다. Loglevel 2는 만료된 메일함별로 하나씩의 메시지를 기록합니다. 마지막으로 Loglevel 3은 만료된 메시지별로 하나씩의 메시지를 기록합니다.

자동 메시지 제거에서 지정된 사용자 제외

msg-svr-base /config/에 있는 expire_exclude_list라는 파일에 한 행씩 사용자 아이디를 추가하여 지정된 사용자를 만료 규칙에서 제외할 수 있습니다. 또는 사용자의 메일함에 배타적인 더미 만료 규칙을 구성합니다.

20.10 메시지 저장소 분할 영역 구성

메일함은 전적으로 메시지 저장소를 저장하는 디스크 분할 영역의 한 영역인 메시지 저장소 분할 영역에 저장됩니다. 메시지 저장소 분할 영역은 디스크 분할 영역과 다르지만 유지 관리가 용이하도록 각 메시지 저장소 분할 영역에 대해 하나의 디스크 분할 영역과 하나의 파일 시스템을 가지는 것이 좋습니다. 메시지 저장소 분할 영역은 특별히 메시지 저장소로 지정된 디렉토리입니다.

사용자 메일함은 기본적으로 *store_root*/partition/ 디렉토리에 저장됩니다(557 페이지 “20.2 메시지 저장소 디렉토리 레이아웃” 참조). partition 디렉토리는 하나 또는 여러 개의 분할 영역을 포함할 수 있는 논리 디렉토리입니다. 시작 시에 partition 디렉토리는 primary 분할 영역이라는 하위 분할 영역을 포함합니다.

필요에 따라 분할 영역을 partition 디렉토리에 추가할 수 있습니다. 예를 들어, 단일 디스크를 분할하여 다음과 같이 사용자를 구성할 수 있습니다.

store_root/partition/mkting/
store_root/partition/eng/
store_root/partition/sales/

디스크 저장소 요구 사항이 늘어나면 이러한 분할 영역을 다른 물리적 디스크 드라이브에 매핑할 수 있습니다.

한 디스크의 메일함 수를 제한해야 합니다. 여러 디스크로 메일함을 분산시키면 메시지 전달 시간이 향상됩니다(SMTP 수락율을 변경할 필요는 없음). 디스크별로 할당하는 메일함 수는 디스크 용량과 각 사용자에게 할당되는 디스크 공간에 따라 다릅니다. 예를 들어, 사용자별로 더 적은 디스크 공간을 할당할 경우 디스크별로 더 많은 메일함을 할당할 수 있습니다.

메시지 저장소에 여러 개의 디스크가 필요한 경우에는 RAID(Redundant Array of Inexpensive Disks) 기술을 사용하여 편리하게 여러 디스크를 관리할 수 있습니다. RAID 기술을 사용하면 일련의 디스크에서 데이터를 분산시킬 수 있지만 디스크가 하나의 논리 볼륨으로 나타나므로 관리가 간단해집니다. 또한 오류 복구 목적으로 저장소를 복제하기 위해(즉, 중복을 위해) RAID 기술을 사용할 수도 있습니다.

주 – 디스크 액세스를 향상시키려면 메시지 저장소와 메시지 대기열이 별개의 디스크에 상주해야 합니다.

20.10.1 분할 영역 추가

분할 영역을 추가할 때에는 디스크상에 분할 영역이 저장되는 절대 물리 경로와 논리 이름(분할 영역 별명이라고 함)을 지정합니다.

분할 영역 별명을 사용하면 물리 경로에 상관 없이 사용자를 논리 분할 영역 이름에 매핑할 수 있습니다. 사용자 계정을 설정하고 사용자의 메시지 저장소를 지정할 때 분할 영역 별명을 사용할 수 있습니다. 입력하는 이름은 소문자를 사용하는 알파벳 이름이어야 합니다.

분할 영역을 작성 및 관리하려면 서버를 실행하는 데 사용되는 사용자 아이디가 물리 경로에 지정된 위치에 쓸 수 있는 권한을 가져야 합니다.

주 – 분할 영역을 추가한 후에 서버를 중지했다가 다시 시작하여 구성 정보를 갱신해야 합니다.

▼ 메시지 저장소 분할 영역 추가 방법

- **명령줄, 명령줄에서 저장소에 분할 영역을 추가하려면 다음을 수행합니다.**

  ```
  configutil -o store.partition.nickname.path -v path
  ```

  여기서 *nickname*은 분할 영역의 논리 이름이고 *path*는 분할 영역이 저장되는 절대 경로 이름을 나타냅니다.

  기본 주 분할 영역의 경로를 지정하려면 다음을 수행합니다.

  ```
  configutil -o store.partition.primary.path -v path
  ```

20.10.2 메일함을 다른 디스크 분할 영역으로 이동

기본적으로 메일함은 `primary` 분할 영역에 만들어집니다. 분할 영역이 가득 차면 추가 메시지를 저장할 수 없습니다. 이 문제는 다음의 여러 방법으로 해결할 수 있습니다.

- 사용자 메일함의 크기를 줄입니다.

- 볼륨 관리 소프트웨어를 사용하는 경우 디스크를 추가합니다.
- 추가 분할 영역을 만들고(603 페이지 "20.10.1 분할 영역 추가" 참조) 메일함을 새 분할 영역으로 이동합니다.

볼륨 관리 소프트웨어를 사용하여 시스템에 다른 디스크 공간을 추가하는 것이 사용자에게 가장 투명한 절차이기 때문에 가능하면 이 방법을 사용하는 것이 좋습니다. 그러나 메일함을 다른 분할 영역으로 이동할 수도 있습니다.

▼ 메일함을 다른 디스크 분할 영역으로 이동

1 **마이그레이션하는 도중에 사용자가 메일함과 연결되지 않게 합니다. 이렇게 하려면 사용자에게 로그오프하고 메일함을 이동하는 동안 메일함을 사용하지 않도록 지시하거나 사용자가 로그오프한 후에 POP, IMAP 및 HTTP 서비스를 허용하지 않도록 `mailAllowedServiceAccess` 속성을 설정합니다. Sun Java Communications Suite 5 Schema Reference의 "mailAllowedServiceAccess"를 참조하십시오.**

주 – POP, IMAP 및 HTTP 액세스를 허용하지 않도록 `mailAllowedServiceAccess`를 설정해도 메일함에 대한 열린 연결이 끊기지 않습니다. 따라서 메일함을 이동하기 전에 모든 연결이 닫혔는지 확인해야 합니다.

---

2 **다음 명령을 사용하여 사용자 메일함을 이동합니다.**

`mboxutil -r user/<`*userid*`>/INBOX user/<`*userid*`>/INBOX < `*partition_name*

예:

```
mboxutil -r user/ofanning/INBOX user/ofanning/INBOX secondary
```

3 **이동한 사용자의 LDAP 항목에서 `mailMessageStore` 속성을 새 분할 영역의 이름으로 설정합니다.**

예: `mailMessageStore: secondary`

4 **이제 메시지 저장소 연결이 허용된다는 것을 사용자에게 알립니다. 해당하는 경우 `mailAllowedServiceAccess` 속성을 변경하여 POP, IMAP 및 HTTP 서비스를 허용합니다.**

20.10.3 기본 메시지 저장소 분할 영역 정의 변경

기본 분할 영역은 사용자를 만들 때 사용자 항목에 `mailMessageStore` LDAP 속성을 지정하지 않은 경우에 사용되는 분할 영역입니다. 기본 분할 영역이 필요하지 않도록 사용자의 메시지 저장소 분할 영역을 지정하는 `mailMessageStore` LDAP 속성을 모든 사용자 항목에 지정해야 합니다. 또한 로드 균형 조정이나 기타 이유 때문에 기본 분할 영역을 변경하면 **안 됩니다**. 기본 분할 영역 정의에 의존하는 사용자가 있는 상태에서 기본 분할 영역을 변경하는 것은 적절하지 않으며 위험합니다.

기본 분할 영역을 꼭 변경해야 할 경우에는 configutil 매개 변수 store.defaultpartition을 사용하여 기본값 정의를 변경하기 전에 이전의 기본 분할 영역(남겨진 것)의 모든 사용자가 mailMessageStore 속성을 현재 분할 영역(더 이상 기본값이 아닌)으로 설정해야 합니다.

20.11 메시지 저장소 유지 관리 절차 수행

이 절에서는 메시지 저장소의 유지 관리 및 복구 작업을 수행하는 데 사용되는 유틸리티에 대해 설명합니다. 관리자는 항상 포스트마스터 메시지를 읽어 서버가 보낼 수 있는 주의와 경고를 확인해야 합니다. 또한 로그 파일에서 서버의 작동 상태에 대한 정보를 모니터해야 합니다. 로그 파일에 대한 자세한 내용은 25 장을 참조하십시오.

이 절은 다음 내용으로 구성되어 있습니다.

20.11.1 메시지 저장소에 물리적 디스크 추가

Messaging Server 메시지 저장소는 특정 Messaging Server 인스턴스에 대한 사용자 메일함을 포함합니다. 메일함, 폴더 및 로그 파일 수가 늘어나면 메시지 저장소의 크기가 늘어납니다.

시스템에 다른 사용자를 추가하면 디스크 저장소 요구 사항이 증가합니다. 서버가 지원하는 사용자 수에 따라 메시지 저장소는 하나 또는 여러 개의 물리적 디스크가 필요할 수 있습니다. Messaging Server를 사용하면 필요에 따라 저장소를 추가할 수 있습니다. 저장소를 추가하는 한 가지 방법은 저장 장치를 사용하는 것입니다. Messaging Server를 사용하여 Network Appliance 저장 장치를 구성하는 방법에 대한 자세한 내용은 **Using NetApp Filers with Sun Java System Messaging Server Message Store**를 참조하십시오.

20.11.2 메일함 관리

이 절에서는 mboxutil, hashdir, readership과 같은 메일함 관리 및 모니터링 유틸리티에 대해 설명합니다.

20.11.2.1 mboxutil 유틸리티

mboxutil 명령을 사용하여 메일함에 대한 일반적인 유지 관리 작업을 수행합니다. mboxutil 작업은 다음을 포함됩니다.

- 메일함 나열
- 고아 및 비활성 메일함 나열 및 제거
- 메일함 작성
- 메일함 이름 바꾸기
- 한 분할 영역에서 다른 분할 영역으로 메일함 이동
- 메일함 정리
- 정리된 후 제거되지 않은 메시지 복원
- 개인의 메일함 가입과 더 이상 존재하지 않는 미가입 메일함 나열
- 또한 mboxutil 명령을 사용하여 할당량에 대한 정보를 볼 수 있습니다. 자세한 내용은 609 페이지 “20.11.4 할당량 제한 모니터”를 참조하십시오.

주 – mboxutil 프로세스를 실행 도중에 종료해서는 안 된다는 것을 유의하십시오. SIGKILL(kill -9)을 사용하여 중지할 경우 모든 서버를 다시 시작하고 복구를 수행해야 할 수 있습니다.

자세한 구문 및 사용 요구 사항은 **Sun Java System Messaging Server 6.3 Administration Reference**의 “mboxutil”을 참조하십시오.

예

모든 사용자의 모든 메일함을 나열하려면 다음을 수행합니다.

```
mboxutil -l
```

모든 메일함을 나열하고 또한 경로와 ACL 정보를 포함하려면 다음을 수행합니다.

```
mboxutil -l -x
```

사용자 daphne에 대해 INBOX라는 기본 메일함을 만들려면 다음을 수행합니다.

```
mboxutil -c user/daphne/INBOX
```

사용자 delilah에 대해 projx라는 메일 폴더를 삭제하려면 다음을 수행합니다.

```
mboxutil -d user/delilah/projx
```

사용자 druscilla에 대해 INBOX라는 기본 메일함과 **모든 메일 폴더**를 삭제하려면 다음을 수행합니다.

```
mboxutil -d user/druscilla/INBOX
```

사용자 desdemona에 대해 메일 폴더 memos의 이름을 memos-april로 바꾸려면 다음을 수행합니다.

```
mboxutil -r user/desdemona/memos user/desdemona/memos-april
```

사용자 dimitria에 대한 메일 계정을 새 분할 영역으로 이동하려면 다음을 수행합니다.

```
mboxutil -r user/dimitria/INBOX user/dimitria/INBOX partition
```

여기서 *partition*은 새 분할 영역의 이름을 지정합니다.

사용자 dimitria에 대해 personal이라는 메일 폴더를 새 분할 영역으로 이동하려면 다음을 수행합니다.

```
mboxutil -r user/dimitria/personal user/dimitria/personal partition
```

20.11.2.2 고아 계정 제거

고아 계정(LDAP에 해당 항목이 없는 메일함)을 검색하려면 다음 명령을 사용합니다.

```
mboxutil -o
```

명령 출력은 다음과 같습니다.

```
mboxutil: Start checking for orphaned mailboxes
user/annie/INBOX
user/oliver/INBOX
mboxutil: Found 2 orphaned mailbox(es)
mboxutil: Done checking for orphaned mailboxes
```

고아 메일함을 삭제하여 스크립트 파일로 변환될 수 있는 고아 메일함을 나열하는 orphans.cmd라는 이름의 파일을 만들려면 다음 명령을 사용합니다.

```
mboxutil -o -w orphans.cmd
```

명령 출력은 다음과 같습니다.

```
mboxutil: Start checking for orphaned mailboxes
mboxutil: Found 2 orphaned mailbox(es)
mboxutil: Done checking for orphaned mailboxes
```

다음 명령을 사용하여 고아 파일을 삭제합니다.

```
mboxutil -d -f orphans.cmd
```

20.11.2.3 hashdir 유틸리티

메시지 저장소의 메일함은 빠른 검색을 위해 해시 구조에 저장됩니다. 결과적으로 특정 사용자의 메일함을 포함하는 디렉토리를 찾으려면 `hashdir` 유틸리티를 사용합니다.

이 유틸리티는 특정 계정의 메시지 저장소를 포함하는 디렉토리를 식별합니다. 이 유틸리티는 메시지 저장소에 상대적인 경로(예: `d1/a7/`)를 보고합니다. 이 경로는 사용자 아이디 기반 디렉토리의 바로 앞에 있는 디렉토리 수준에 상대적입니다. 이 유틸리티는 경로 정보를 표준 출력으로 보냅니다.

예를 들어, 사용자 `crowe`에 대한 메일함의 상대 경로를 찾으려면 다음을 수행합니다.

```
hashdir crowe
```

20.11.2.4 readership 유틸리티

`readership` 유틸리티는 공유 IMAP 폴더의 메시지를 읽은 메일함 소유자 이외의 사용자 수를 보고합니다.

IMAP 폴더 소유자는 폴더의 메시지를 읽는 권한을 다른 사용자에게 부여할 수 있습니다. 다른 사람에게 액세스가 허용되는 폴더를 **공유 폴더**라고 합니다. 관리자는 `readership` 유틸리티를 사용하여 공유 폴더를 액세스하는 소유자 이외의 사용자 수를 확인할 수 있습니다.

이 유틸리티는 모든 메일함을 스캔한 후 공유 폴더별로 한 행씩의 출력을 생성하여 읽은 사람 수(뒤에 공백과 메일함 이름이 옴)를 보고합니다.

각 읽은 사람은 이전의 지정된 일 수 동안 공유 폴더를 선택했던 고유한 인증 아이디입니다. 자신의 고유한 메일함을 읽을 때는 사용자가 계산되지 않습니다. 폴더 소유자 외에 최소 한 명 이상의 읽은 사람이 존재하지 않을 경우 개인 메일함은 보고되지 않습니다.

예를 들어, 다음 명령은 마지막 15일 동안에 공유 IMAP 폴더를 선택한 모든 아이디를 읽은 사람으로 계산합니다.

```
readership -d 15
```

20.11.3 최대 메일함 크기

메일함의 최대 크기는 약 100만 개의 메시지에 해당됩니다. 메시지가 이보다 많아지면 더 이상 메시지가 사용자에게 전달되지 않으며 메시지 저장소 성능에 문제가 생길 수 있습니다. 자세한 내용은 641 페이지 “20.14.4.7 메일함 오버플로 때문에 사용자 메일이 전달되지 않음”을 참조하십시오.

20.11.4 할당량 제한 모니터

imqutoacheck를 사용하여 할당량 사용 및 제한을 모니터링한 다음 정의된 할당량과 제한을 나열하는 보고서를 생성하고 할당량 사용과 관련된 정보를 제공합니다. 할당량과 사용량 수치는 KB로 보고됩니다. 이 유틸리티는 메일함 크기를 사용자의 할당량과 비교할 수도 있습니다. 옵션으로 지정된 할당량 비율을 초과한 사용자에게 전자 메일 알림을 보낼 수 있습니다.

주 – imquotacheck에서 일부 기능이 변경되었습니다. Messaging Server 6.x에서 imquotacheck 유틸리티가 quotacheck 유틸리티를 대체했습니다. Messaging Server 5.x에서는 quotacheck 유틸리티를 사용하여 사용자 목록을 검색할 때 quotacheck가 로컬 mboxlist 데이터베이스를 검색했습니다. 이 기능은 mboxutil 유틸리티의 검색 기능과 중복되었습니다.

Messaging Server 6.x에서는 이 중복된 기능이 imquotacheck 유틸리티에서 제거되었습니다. imquotacheck를 사용하여 사용자 검색을 수행할 경우 로컬 mboxlist 데이터베이스가 아닌 LDAP 디렉토리에 대해 검색이 수행됩니다. 로컬 mboxlist 데이터베이스에서 사용자 목록을 검색하려면 mboxutil 유틸리티를 사용합니다.

할당량이 규칙 파일의 최소 임계값을 초과하는 모든 사용자에 대한 사용량을 나열하려면 다음을 수행합니다.

```
imquotacheck
```

도메인 siroe.com에 대한 할당량 정보를 나열합니다.

```
imquotacheck -d siroe.com
```

기본 규칙 파일에 따라 모든 사용자에게 알림을 보내려면 다음을 수행합니다.

```
imquotacheck -n
```

지정된 *rulefile*, *myrulefile* 및 지정된 메일 템플리트 파일 *mytemplate.file*에 따라 모든 사용자에게 알림을 보내려면 다음을 수행합니다(자세한 내용은 **Sun Java System Messaging Server 6.3 Administration Reference**의 “imquotacheck” 참조).

```
imquotacheck -n -r myrulefile -t mytemplate.file
```

모든 사용자에 대한 사용량을 나열하고 규칙 파일을 무시하려면 다음을 수행합니다.

```
imquotacheck -i
```

사용자 user1에 대한 폴더별 사용량을 나열하려면 다음을 수행합니다(규칙 파일 무시).

```
imquotacheck -u user1 -e
```

20.11.5 디스크 공간 모니터

시스템이 디스크 공간과 분할 영역 사용을 모니터해야 하는 빈도와 경고를 보내야 하는 상황을 지정할 수 있습니다. 자세한 내용은 833 페이지 “27.3.2 디스크 공간 모니터링”을 참조하십시오.

20.11.6 stored 데몬

stored 데몬은 메시지 저장소에 대해 다음과 같은 유지 관리 작업을 수행합니다.

- 검사점 데이터베이스 트랜잭션 수행.
- 교착 상태 감지 및 교착 상태에 빠진 데이터베이스 트랜잭션의 롤백
- 시작 시에 임시 파일 및 잠금 파일 제거
- 데이터베이스 스냅샷 아카이브 생성
- 필요에 따른 데이터베이스 복구(630 페이지 “20.14.2 메시지 저장소 시작 및 복구” 참조)

임의의 서버 데몬이 충돌된 경우 stored를 비롯한 모든 데몬을 중지했다가 다시 시작해야 합니다.

20.11.7 동일한 메시지의 중복 저장에 따른 저장소 크기 줄이기

한 메시지가 여러 수신자에 전송될 때 해당 메시지는 각 수신자의 메일함마다 놓이게 됩니다. 일부 메시징 시스템에서는 각 수신자의 메일함에 같은 메시지의 복사본을 별도로 저장합니다. 그러나 이와 반대로, Sun Java System Messaging Server에서는 해당 메시지가 있는 메일함의 수에 관계 없이 메시지 사본을 하나만 유지합니다. 이는 해당 메시지를 포함하는 메일함에 메시지에 대한 하드 링크를 작성하는 방법으로 이루어집니다.

다른 메시징 시스템을 Sun Java Messaging Server로 마이그레이션할 때는 마이그레이션 과정을 통해 이러한 여러 메시지 복사본이 복사될 수 있습니다. 이는 규모가 큰 메시지 저장소의 경우 불필요하게 많은 메시지가 중복되는 것을 의미합니다. 또한 일반적인 서버 작업(예: IMAP append 작업이나 기타 소스의 작업)에 같은 메시지 사본이 여러 개 누적될 수 있습니다.

Messaging Server는 과도한 메시지 복사본을 제거하고 하나의 복사본에 대한 하드 링크로 대체하는 relinker라는 이름의 새 명령을 제공합니다.

20.11.7.1 relinker작동 원리

재연결 기능은 이 명령이나 실시간 모드에서 실행할 수 있습니다. relinker 명령을 실행할 때는 이 명령이 메시지 저장소 분할 영역 전체를 스캔하고 MD5 메시지 다이제스트 저장소(하드 링크에 해당)를 작성 또는 업데이트하며 필요한 하드 링크를 만듭니다.

다이제스트 저장소는 메시지 저장소의 메시지에 대한 하드 링크로 이루어져 있습니다. 디렉토리 계층 *partition_path*/=md5에 저장됩니다. 이 디렉토리는 사용자 메일함 계층 *partition_path*/=user(그림 20–1 참조)와 병행합니다. 다이제스트 저장소의 메시지는 MD5 다이제스트에서 고유하게 식별합니다. 예를 들어, fredb/00/1.msg의 다이제스트가 4F92E5673E091B43415FFFA05D2E47인 경우 *partition*/=user/*hashdir*/*hashdir*/=fredb/00/1.msg는 *partition*/=md5/*hashdir*/*hashdir*/4F92E5673E091B43415FFFA05D2E47EA.msg에 연결됩니다. 다른 메일함에 이와 같은 메시지가 있을 때(예: *partition_path*/=user/*hashdir*/*hashdir*/gregk/00/17.msg) 해당 메시지는 *partition_path*/=md5/4F/92/4F92E5673E091B43415FFFA05D2E47EA.msg에 대해 하드 링크로 연결됩니다. 그림 20–4에 표시되어 있습니다.

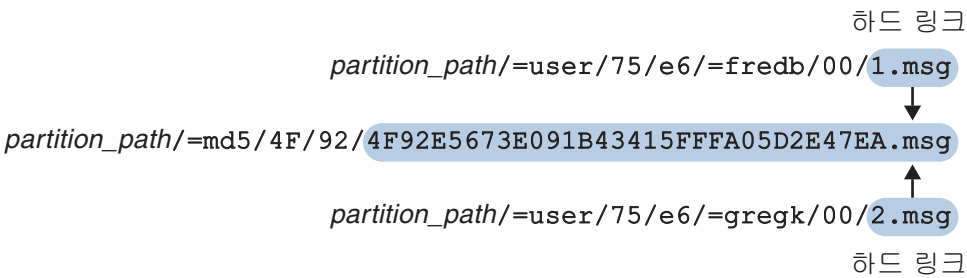


그림 20–4 메시지 저장소 다이제스트 저장소

이 메시지의 경우 링크 수는 3이 됩니다. 두 메시지 모두 fredb 및 gregk의 메일함에서 삭제되면 링크 수는 하나가 되고 해당 메시지를 삭제할 수 있습니다.

relinker 프로세스는 같은 기능에 대해 실시간으로 실행될 수 있습니다. 자세한 내용은 613 페이지 “20.11.7.3 실시간 모드에서 relinker 사용”을 참조하십시오.

20.11.7.2 명령줄 모드에서 relinker 사용

relinker는 메시지 저장소 분할 영역 전체를 스캔하고 MD5 메시지 저장소(하드 링크에 해당)를 작성 또는 업데이트하며 과도한 메시지 파일을 삭제합니다. relinker가 저장소 분할 영역을 스캔한 후에는 재연결 이전과 이후 해당 분할 영역의 크기와 고유한 메시지의 수에 대한 통계를 산출합니다. 이미 해시된 저장소에서 실행 속도를 높이려면 relinker는 아직 =md5에 있지 않은 메시지의 다이제스트만 계산합니다. 전체 다이제스트 저장소를 지우는 옵션도 있습니다(사용자 메시지함에는 영향을 주지 않음).

이 명령의 구문은 다음과 같습니다.

relinker [-P *partitionname*] [-d]

여기서 *partitionname*은 처리될 분할 영역(기본값: 모든 분할 영역)을 지정하고 -d는 다이제스트 저장소가 삭제됨을 지정합니다. 출력 예는 다음과 같습니다.

```
# relinker

Processing partition: primary
Scanning digest repository...
Processing user directories.............................
-----------------------------------------------------------
Partition statistics           Before            After
-----------------------------------------------------------
Total messages                 4531898         4531898
Unique messages                4327531         3847029
Message digests in repository        0         3847029
Space used                       99210Mb          90481Mb
Space savings from single-copy    3911Mb          12640Mb
-----------------------------------------------------------


# relinker -d
Processing partition: primary
Purging digest repository...
-----------------------------------------------------------
Partition statistics                 Before         After
-----------------------------------------------------------
Message digests in repository       3847029             0
-----------------------------------------------------------
```

특히 저장소에 메시지가 하나도 없는 처음에는 relinker 실행에 시간이 많이 걸릴 수 있습니다. 이는 모든 메시지마다 다이제스트를 계산(relinker 기준이 모든 메시지를 포함하도록 구성된 경우)해야 하기 때문입니다. relinker 기준 구성에 대한 자세한 내용은 613 페이지 “20.11.7.4 relinker 구성”을 참조하십시오. 예를 들어, 100GB의 메시지 저장소를 처리하는 데는 6시간 정도가 걸립니다. 그러나 런타임 재연결을 활성화한 경우(613 페이지 “20.11.7.3 실시간 모드에서 relinker 사용” 참조)에는 relinker 명령을 실행할 필요가 없습니다.

relinker 명령줄 모드가 배타적으로 사용되고 런타임 옵션이 아닌 경우에는 다이제스트 저장소(=md5)를 제거해야 합니다. 그 외의 경우에는 저장소에서 제거된 메시지(=user)에 해당 다이제스트 저장소의 연결이 그대로 있으므로(고아 메시지가 됨) 사용 가능한 디스크 공간이 되지 않습니다. relinker를 한 번만 실행할 수 있는 저장소의 일회 최적화를 수행하는 경우(예: 마이그레이션 후)에는 relinker -d를 사용하여 전체 저장소를 삭제합니다. 반복된 제거의 경우(마이그레이션 도중)에는 relinker 명령을 반복적으로 수행하는 것만으로도 충분합니다. 이 명령을 실행할 때마다 만료된 메시지나 고아 메시지를 저장소에서 제거하기 때문입니다.

처리되는 다른 분할 영역마다 병행하여 relinker의 여러 인스턴스를 실행(-p 옵션 사용)하는 것이 안전합니다. 메시지는 같은 파티션 안에서만 재연결됩니다.

20.11.7.3 실시간 모드에서 relinker 사용

relinker 기능은 configutil 매개 변수 local.store.relinker.enabled를 yes로 설정하면 실시간 모드에서 활성화할 수 있습니다. 실시간 모드에서 relinker를 사용하면 구성된 relinker 기준(613 페이지 "20.11.7.4 relinker 구성")에 맞게 배달된(또는 복원되거나 IMAP 추가된) 모든 메시지의 다이제스트를 계산한 다음 해당 다이제스트가 이미 존재하는지 여부를 저장소에서 확인하게 됩니다. 이 다이제스트가 존재하는 경우에는 해당 메시지의 새 사본을 만들지 않고 그에 대한 링크를 대상 메일함에 만듭니다. 다이제스트가 없을 때는 메시지를 만들고 나중에 그에 대한 링크를 저장소에 추가합니다.

stored는 각 분할 영역의 다이제스트 저장소를 스캔하고 링크 수가 한 개인 메시지(즉, relinker 기준에 맞지 않는 메시지)를 제거합니다. 스캔은 구성 가능한 시간 동안 한 번에 한 디렉토리씩 수행됩니다. 이로써 I/O 로드가 공평하게 분산되고 다른 서버 작업에 별다른 영향을 주지 않게 됩니다. 기본적으로 제거 주기는 24시간이며 이는 저장소에서 메시지가 삭제되거나 구성된 최대 캐시 사용 기간을 초과해도 최대 24시간까지는 그대로 남아 있다는 의미입니다. 이 작업은 relinker 실시간 모드가 활성화되어 있는 경우 사용 가능합니다.

20.11.7.4 relinker 구성

표 20–11은 relinker 기준 설정에 사용되는 매개 변수를 보여 줍니다.

표 20–11 relinker configutil 매개 변수

매개 변수	설명
local.store.relinker.enabled	추가 코드 및 stored 제거에서 메시지의 실시간 재연결을 활성화합니다. relinker 명령줄 도구는 이 옵션이 꺼져 있더라도 실행할 수 있습니다. 그러나 stored가 저장소를 제거하지 않으므로 이 작업에서는 relinker -d를 사용해야 합니다. 이 옵션을 설정하면 디스크 공간이 절감되는 대신 메시지 배달 성능이 떨어집니다. 기본값: no
local.store.relinker.maxage	저장소에 메시지가 저장될 최대 캐시 사용 기간(시간)으로, relinker 명령줄에서 고려합니다. -1은 제한이 없다는 뜻으로 저장소에서 고아 메시지만 삭제합니다. relinker의 경우 이는 캐시 사용 기간에 관계 없이 기존 메시지를 처리한다는 의미가 됩니다. 값을 적게 할수록 저장소가 작아지므로 relinker 또는 stored 삭제가 더 빨리 실행되고 디스크 공간을 더 빨리 재생 이용할 수 있습니다. 반면, 값을 크게 할수록 긴 시간(예를 들어, 사용자가 같은 메시지를 며칠 간격으로 저장하거나 며칠 또는 몇 주에 걸쳐 마이그레이션을 실행할 때)에 걸쳐 메시지 재연결이 중복됩니다. 기본값: 24
local.store.relinker.minsize	런타임 또는 명령줄 relinker에서 고려하는 메시지의 최소 크기. 0이 아닌 값을 설정하면 저장소는 작아지는 대신 규모가 작은 메시지에 대한 relinker 혜택은 늘어납니다. 기본값: 0

표 20–11 relinker configutil 매개 변수 (계속)

매개 변수	설명
local.store.relinker.purgecycle	전체 stored 삭제 주기의 대략적인 지속 시간. 실제 지속 시간은 저장소의 각 디렉토리를 스캔하는 데 걸리는 시간에 따라 달라집니다. 값이 적을수록 더 많은 I/O를 사용하게 되고 값이 클수록 디스크 공간의 재생 이용이 느려집니다. 0은 디렉토리 사이에 일시 중지 없이 계속적으로 삭제가 실행된다는 의미입니다. -1은 stored에서 제거가 실행되지 않는다는 뜻입니다. 따라서 제거는 relinker -d 명령을 사용하여 수행해야 합니다. 기본값: 24

20.12 메시지 저장소 백업 및 복원

메시지 저장소 백업 및 복원은 가장 일반적이고 중요한 관리 작업 중 하나입니다. 이 작업은 메시지 저장소의 모든 메시지와 폴더를 백업하는 것으로 구성됩니다. 다음과 같은 문제가 발생했을 때 데이터가 손실되지 않도록 메시지 저장소에 대한 백업 및 복원 정책을 구현해야 합니다.

- 시스템 충돌
- 하드웨어 오류
- 메시지 또는 메일함의 우발적 삭제
- 시스템 재설치 또는 업그레이드 시의 문제
- 자연 재해(예: 지진, 화재, 태풍)
- 사용자 마이그레이션

imsbackup 및 imsrestore 명령줄 유틸리티나 Legato Networker™를 사용하는 통합 솔루션을 사용하여 메시지 저장소 백업 및 복원을 수행할 수 있습니다.

Messaging Server는 단일 복사본 백업 절차를 제공합니다. 특정 메시지를 포함하는 사용자 폴더 수에 상관 없이 백업 도중 메시지 파일은 처음 발견된 메시지 파일을 사용하여 한 번만 백업됩니다. 두 번째 메시지 복사본은 첫 번째 메시지 파일의 이름에 대한 링크로 백업되며 그 다음 복사본도 마찬가지입니다. imsbackup은 메시지 파일의 장치와 색인 노드를 색인으로 사용하여 모든 메시지의 해시 테이블을 유지 관리합니다. 단, 이 방법은 데이터 복원 시 고려해야 할 사항이 있습니다. 자세한 내용은 619 페이지 “20.12.5 부분 복원 시의 고려 사항”을 참조하십시오.

주 – 모든 메시지 파일과 디렉토리를 백업함으로써 메시지 저장소 백업과 복원을 수행할 수도 있습니다. 625 페이지 “20.12.9 메시지 저장소 재해 복구 및 복원”을 참조하십시오.

이 절에는 다음과 같은 하위 절이 포함됩니다.

20.12.1 메일함 백업 정책 만들기

백업 정책은 다음 몇 가지 요인에 영향을 받습니다.

20.12.1.1 작업량이 가장 많은 시간대

시스템에 대한 백업 일정을 예약할 때에는 작업량이 가장 많은 시간대에 시스템 로드를 줄일 수 있도록 작업량이 가장 많은 시간대를 피해야 합니다. 예를 들어, 백업은 오전 2시와 같은 새벽에 실행되도록 예약하는 것이 가장 적합합니다.

20.12.1.2 전체 및 증분 백업

증분 백업(617 페이지 “증분 백업” 참조)은 변경된 데이터의 저장소를 스캔하고 변경된 사항만 백업합니다. 전체 백업은 전체 메시지 저장소를 백업합니다. 증분 백업과 달리 전체 백업은 시스템이 전체 백업을 수행하는 빈도를 결정해야 합니다. 일반적으로 증분 백업을 일상적인 유지 관리 절차로 수행하면서 일주일에 한 번씩 전체 백업을 수행합니다.

20.12.1.3 병렬 또는 직렬 백업

사용자 데이터를 여러 디스크에 저장할 경우 필요에 따라 사용자 그룹을 병렬로 백업할 수 있습니다. 시스템 자원에 따라 병렬 백업은 전반적인 백업 절차의 속도를 높일 수 있습니다. 하지만 백업이 서버 성능에 미치는 영향을 최소화하려는 경우에는 직렬 백업을 사용합니다. 병렬 백업 또는 직렬 백업 사용 여부는 시스템 로드, 하드웨어 구성, 사용 가능한 테이프 드라이브 수 등에 따라 달라질 수 있습니다.

20.12.2 백업 그룹 만들기

백업 그룹은 정규식에 의해 정의되는 임의의 사용자 메일함 집합입니다. 사용자 메일함을 백업 그룹으로 정리하면 보다 유연한 백업 관리를 정의할 수 있습니다.

예를 들어, 사용자 아이디가 A-L로 시작하는 사용자를 포함하는 첫 번째 백업 그룹, 사용자 아이디가 M-Z로 시작하는 사용자를 포함하는 두 번째 백업 그룹, 사용자 아이디가 숫자로 시작하는 사용자를 포함하는 세 번째 백업 그룹의 세 가지 백업 그룹을 만들 수 있습니다. 관리자는 이러한 백업 그룹을 사용하여 메일함을 병렬로 백업하거나 특정 날짜에 일정 그룹만 백업하고 다른 날짜에 다른 그룹을 백업할 수 있습니다.

백업 그룹과 관련하여 다음 몇 가지 사항에 유념해야 합니다.

1. 백업 그룹은 메일 사용자의 임의 **가상** 그룹이며 보기와 달리 메시지 저장소 디렉토리(그림 20–1)에 정확하게 매핑되지 않습니다.
2. 관리자가 UNIX 정규식을 사용하여 백업 그룹을 정의합니다.
3. 정규식은 *msg-svr-base*/config/backup-groups.conf 구성 파일에 정의됩니다.
4. 백업 그룹은 imsbackup 및 imsrestore에서 참조될 경우 다음 경로 형식을 사용합니다. /*partition_name*/*backup_group*

backup-groups.conf의 형식은 다음과 같습니다.

group_name=definition
group_name=definition
.
.
.

위 단락에 설명된 예에 따라 다음 정의를 사용하여 세 개의 백업 그룹을 만듭니다.

```
groupA=[a-l].*
groupB=[m,-z].*
groupC=[0-9].*
```

이제 imsbackup 및 imsrestore를 여러 수준에서 사용할 수 있습니다. 다음과 같이 백업 명령을 사용하여 전체 메시지 저장소를 백업/복원할 수 있습니다.

imsbackup -f *device* /

groupA의 모든 사용자에 대한 모든 메일함을 백업하려면 다음 명령을 사용합니다.

imsbackup -f *device* /*partition*/groupA

기본 분할 영역을 primary라고 합니다.

20.12.2.1 미리 정의된 백업 그룹

Messaging Server에는 backup-groups 구성 파일을 만들지 않고 사용할 수 있는 하나의 미리 정의된 백업 그룹이 포함되어 있습니다. 이 그룹은 user라고 하며 모든 사용자를 포함합니다. 예를 들어, 다음 명령은 primary 분할 영역의 모든 사용자를 백업합니다.

```
imsbackup -f backupfile /primary/user
```

20.12.3 Messaging Server 백업 및 복원 유틸리티

데이터를 백업 및 복원하기 위해 Messaging Server는 `imsbackup` 및 `imsrestore` 유틸리티를 제공합니다. `imsbackup` 및 `imsrestore` 유틸리티는 Legato Networker와 같은 일반 용도의 도구에 있는 고급 기능을 포함하지 않습니다. 예를 들어, 이러한 유틸리티는 테이프 자동 변환기에 대한 매우 제한적인 지원을 제공하고 단일 저장소를 동시에 여러 장치에 기록할 수 없습니다. 포괄적인 백업은 Legato Networker와 같은 일반화된 도구에 대한 플러그 인을 통해 실현됩니다. Legato Networker 사용에 대한 자세한 내용은 621 페이지 "20.12.6 Legato Networker 사용"을 참조하십시오.

20.12.3.1 imsbackup 유틸리티

`imsbackup`을 사용하면 메시지 저장소에서 선택한 내용을 자기 테이프, UNIX 파이프 또는 일반 파일을 비롯한 모든 직렬 장치에 기록할 수 있습니다. 백업이나 백업의 일부를 나중에 `imsrestore` 유틸리티를 사용하여 복구할 수 있습니다. `imsbackup`의 출력을 `imsrestore`로 파이프할 수 있습니다.

다음 예에서는 전체 메시지 저장소를 `/dev/rmt/0`으로 백업합니다.

```
imsbackup -f /dev/rmt/0 /
```

여기에서는 사용자 아이디 `joe`의 메일함을 `/dev/rmt/0`으로 백업합니다.

```
imsbackup -f /dev/rmt/0 /primary/user/joe
```

다음 예에서는 백업 그룹 `groupA`에 정의된 모든 사용자의 모든 메일함을 `backupfile`에 백업합니다(615 페이지 "20.12.2 백업 그룹 만들기" 참조).

```
imsbackup -f- /primary/groupA > backupfile
```

증분 백업

다음 예는 2004년 5월 1일 1:10 PM부터 현재까지 저장된 메시지를 백업합니다. 기본값은 날짜에 상관없이 모든 메시지를 백업하는 것입니다.

```
imsbackup -f /dev/rmt/0 -d 20040501:131000 /
```

이 명령은 기본 차단 요소 20을 사용합니다. `imsbackup` 명령의 전체 구문 설명은 **Sun Java System Messaging Server 6.3 Administration Reference**를 참조하십시오.

20.12.3.2 imsrestore 유틸리티

백업 장치에서 메시지를 복원하려면 `imsrestore` 명령을 사용합니다. 예를 들어, 다음 명령은 `backupfile` 파일에서 `user1`의 메시지를 복원합니다.

```
imsrestore -f backupfile /primary/user1
```

`imsbackup` 명령의 전체 구문 설명은 **Sun Java System Messaging Server 6.3 Administration Reference**를 참조하십시오.

20.12.4 백업 수행 시 대량 메일 제외

백업 작업을 수행할 때 백업에서 제외할 메일함을 지정할 수 있습니다. 중요하지 않은 메시지를 늘릴 수 있는 대량 또는 휴지통 메일함을 제외함으로써 백업 세션을 단순화하고 작업을 완료하는 데 필요한 시간을 줄이며 백업 데이터를 저장하는 데 필요한 디스크 공간을 최소화할 수 있습니다.

메일함을 제외하려면 `configutil` 매개 변수 `local.store.backup.exclude`에 대한 값을 지정합니다.

하나의 메일함이나 "%" 문자로 구분된 메일함 목록을 지정할 수 있습니다. "%"는 메일함 이름에 사용할 수 없는 문자입니다. 예를 들어, 다음 값을 지정할 수 있습니다.

```
Trash
```

```
Trash%Bulk Mail%Third Class Mail
```

첫 번째 예에서는 폴더 `Trash`가 제외됩니다. 두 번째 예에서는 폴더 `Trash`, `Bulk Mail` 및 `Third Class Mail`이 제외됩니다.

백업 유틸리티는 `local.store.backup.exclude` 매개 변수에 지정된 폴더를 제외하고 사용자 메일함의 모든 폴더를 백업합니다.

이 기능은 Messaging Server 백업 유틸리티, Legato Networker 및 타사 백업 소프트웨어와 함께 작동합니다.

작업 도중에 전체 논리 이름을 지정하여 `local.store.backup.exclude` 설정을 대체하거나 제외된 메일함을 백업할 수 있습니다. 휴지통 폴더가 제외되었다고 가정합니다. 다음을 지정하여 휴지통도 백업할 수 있습니다.

```
/primary/user/user1/trash
```

그러나 다음과 같이 지정할 경우에는

```
/primary/user/user1
```

휴지통 폴더가 제외됩니다.

20.12.5 부분 복원 시의 고려 사항

부분 복원은 메시지 저장소의 일부만 복원하는 것이고 전체 복원은 전체 메시지 저장소를 복원하는 것입니다. 메시지 저장소는 단일 복사본 메시지 시스템을 사용합니다. 즉, 메시지의 단일 복사본만 단일 파일로 저장소에 저장됩니다. 메시지를 여러 메일함으로 보낼 때와 같은 메시지의 다른 인스턴스는 해당 복사본에 대한 링크로 저장됩니다. 따라서 메시지를 복원할 때 고려해야 할 사항이 있습니다. 예를 들면 다음과 같습니다.

- **전체 복원.** 전체 복원 도중에 연결된 메시지는 계속해서 연결된 메시지 파일과 동일한 색인 노드를 가리킵니다.
- **부분 백업/복원.** 부분 백업 및 복원 도중에는 메시지 저장소의 단일 복사본 특성이 유지되지 않을 수 있습니다.

다음 예에서는 부분 복원 작업을 수행할 때 여러 사용자가 사용하는 메시지에 발생하는 변화를 보여 줍니다. 세 명의 사용자 A , B 및 C에 속하는 모두 동일한 다음 세 개의 메시지가 있다고 가정합니다.

```
A/INBOX/1
B/INBOX/1
C/INBOX/1
```

예 1. 첫 번째 예에서는 시스템이 다음과 같이 부분 백업 및 전체 복원 절차를 수행합니다.

1. 사용자 B 및 C의 메일함을 백업합니다.
2. 사용자 B 및 C의 메일함을 삭제합니다.
3. 단계 1의 백업 데이터를 복원합니다.

이 예에서는 `B/INBOX/1` 및 `C/INBOX/1`에 새 색인 노드 번호가 할당되며 메시지 데이터가 디스크의 새 위치에 기록됩니다. 하나의 메시지만 복원되며 두 번째 메시지는 첫 번째 메시지에 대한 하드 링크입니다.

예 2. 이 예에서는 시스템이 다음과 같이 전체 백업 및 부분 복원 작업을 수행합니다.

1. 전체 백업을 수행합니다.
2. 사용자 A의 메일함을 삭제합니다.
3. 사용자 A의 메일함을 복원합니다.

A/INBOX/1에 새 색인 노드 번호가 할당됩니다.

예 3. 이 예에서는 여러 번의 부분 복원 시도가 필요할 수 있습니다.

1. 전체 백업을 수행합니다.

   `B/INBOX/1` 및 `C/INBOX/1`은 `A/INBOX/1`에 대한 링크로 백업됩니다.
2. 사용자 A 및 B의 메일함을 삭제합니다.
3. 사용자 B의 메일함을 복원합니다.

   복원 유틸리티는 관리자에게 `A/INBOX`를 먼저 복원할 것을 요청합니다.

4. 사용자 A 및 B의 메일함을 복원합니다.
5. 사용자 A의 메일함을 삭제합니다(선택 사항).

주 – 부분 복원으로 모든 메시지가 복원되게 하려면 `imsbackup` 명령을 `-i` 옵션과 함께 실행합니다. `-i` 옵션은 필요한 경우 모든 메시지를 여러 번 백업합니다.

백업 장치를 검색할 수 있는 경우(예: 드라이브 또는 테이프) `imsrestore`는 `A/INBOX/1`을 포함하는 위치를 검색하여 `B/INBOX/1`로 복원합니다. 백업 장치를 검색할 수 있는 경우(예: UNIX 파이프) `imsrestore`는 객체 아이디와 파일에 의존하는(연결된) 객체의 아이디를 기록하며 관리자는 `-r` 옵션과 함께 `imsrestore`를 다시 호출하여 누락된 메시지 참조를 복원해야 합니다.

20.12.5.1 증분 백업된 메일함에서 메시지을 복원하는 방법

증분 백업된 메일함에서 메시지를 복원할 때 해당 메일함이 메시지를 복원할 서버에 있는 경우 간단하게 `imesrestore`를 실행하여 메시지를 복원할 수 있습니다. 그러나 증분 백업된 메일함에서 메시지를 복원할 때 해당 메일함이 더 이상 없는 경우에는 다른 복원 절차를 따라야 합니다.

메시지 저장소 서버에 없는 메일함에 메시지를 복원하려면 다음 절차 중 하나를 사용합니다.

- 복원 작업 도중에 사용자에 대한 메시지 전달을 비활성화합니다. 이렇게 하려면 LDAP 속성 `mailDeliveryOption`을 `hold`로 설정합니다.
- `imesrestore`를 사용하기 전에 `mboxutil -c` 명령을 사용하여 메일함을 만듭니다.

증분 백업을 복원할 때 이러한 지침을 따라야 하는 이유는 다음과 같습니다. 메일함이 삭제되거나 마이그레이션되면 `imsrestore` 유틸리티는 백업 아카이브에 저장된 메일함 고유 아이디 유효성과 메시지 고유 아이디(UID)를 사용하여 메일함을 다시 만듭니다.

이전에는 `imsrestore`가 삭제되었거나 마이그레이션된 메일함을 다시 만들 때 새 UID 유효성을 메일함에 할당하고 새 UID를 메시지에 할당했습니다. 이 경우 캐시된 메시지를 가진 클라이언트는 메일함 UID 유효성과 메시지 UID를 다시 동기화해야 합니다. 클라이언트가 새 데이터를 다시 다운로드해야 하므로 서버에서 작업 로드가 증가합니다.

새 `imsrestore` 동작의 경우에는 클라이언트 캐시가 동기화된 상태로 유지되며 복원 프로세스가 투명하게 작동하므로 성능에 부정적인 영향이 없습니다.

메일함이 있는 경우 `imsrestore`는 새 UID를 복원된 메시지에 할당하므로 새 UID와 이미 기존 메시지에 할당된 UID의 일관성이 유지됩니다. UID 일관성을 보장하기 위해 `imsrestore`는 복원 작업 도중에 메일함을 잠급니다. 그러나 `imsrestore`는 이제 새 UID 값을 할당하는 대신에 백업 아카이브의 메일함 UID 유효성과 메시지 UID를 사용하므로 증분 백업 및 복원을 수행할 경우 UID가 일관되지 않을 수 있습니다.

imsbackup 유틸리티의 -d 날짜 옵션을 사용하여 증분 백업을 수행할 경우 복원 작업을 완료하기 위해 imsrestore를 여러 번 호출해야 할 수 있습니다. 증분 백업이 수행되었으면 최신 전체 백업과 이후의 모든 증분 백업을 복원해야 합니다.

복원 작업 사이에 새 메시지를 메일함에 전달할 수 있지만 이 경우에는 메시지 UID가 일관되지 않을 수 있습니다. UID의 비일관성을 방지하려면 위에 설명된 작업 중 하나를 수행해야 합니다.

20.12.6 Legato Networker 사용

Messaging Server에는 Legato Networker와 같은 타사 백업 도구와의 인터페이스를 제공하는 백업 API가 포함되어 있습니다. 물리적 메시지 저장소 구조와 데이터 형식은 백업 API 내에서 캡슐화됩니다. 메시지 저장소와 직접 상호 작용하는 백업 API는 메시지 저장소의 논리적 뷰를 백업 서비스에 제공합니다. 백업 서비스는 메시지 저장소의 개념적 표시를 사용하여 백업 객체를 저장 및 복원합니다.

Messaging Server는 Legato Networker의 save 및 recover 명령으로 호출하여 메시지 저장소 데이터를 백업 및 복원할 수 있는 ASM(Application Specific Module)을 제공합니다. 호출된 ASM은 이어서 Messaging Server imsbackup 및 imsrestore 유틸리티를 호출합니다.

주 – 이 절에서는 Messaging Server 메시지 저장소와 함께 Legato Networker를 사용하는 방법에 대해 설명합니다. Legato Networker 인터페이스를 이해하려면 Legato 설명서를 참조하십시오.

▼ Legato Networker를 사용하여 데이터를 백업하는 방법

1. **/usr/lib/nsr/imsasm에서 msg-srv-base/lib/msg/imsasm에 대한 심볼릭 링크를 만듭니다.**

2. **Sun 또는 Legato에서 nsrfile 이진 파일의 복사본을 얻어 다음 디렉토리에 복사합니다.**

   /usr/bin/nsr

   이전 버전 Networker (5.x)를 사용 중인 경우에만 필요합니다. Networker 6.0 이상에서는 nsrfile이 /usr/bin/nsr에 자동으로 설치됩니다.

3. **그룹별로 사용자를 백업하려면 다음 단계를 수행합니다.**

   a. **615 페이지 "20.12.2 백업 그룹 만들기"에 설명된 대로 백업 그룹 파일을 만듭니다.**

   b. **구성을 확인하려면 mkbackupdir.sh를 실행합니다.**

      mkbackupdir.sh에 의해 작성된 디렉토리 구조를 확인합니다. 이 구조는 표 20–4에 나온 것과 비슷해야 합니다.

backup-groups.conf 파일을 지정하지 않을 경우 백업 프로세스는 모든 사용자에 대해 기본 백업 그룹 ALL을 사용합니다.

4 백업 전에 mkbackupdir.sh 스크립트를 호출하기 위해 /nsr/res/ 디렉토리에서 저장 그룹에 대한 res 파일을 만듭니다. 표 20–4에 예가 나와 있습니다.

주 – 이전 버전의 Legato Networker에서는 saveset 이름이 64자로 제한됩니다. 이 디렉토리 이름과 메일함의 논리 이름을 합친 것(예: /primary/groupA/fred)이 64자 이상인 경우 mkbackupdir.sh -p를 실행해야 합니다. 따라서 mkbackupdir.sh의 -p 옵션에 짧은 경로 이름을 사용해야 합니다. 예를 들어, 다음 명령은 /backup 디렉토리 아래에 백업 이미지를 만듭니다.

```
mkbackupdir.sh -p /backup
```

중요백업 디렉토리는 메시지 저장소 소유자(예: mailsrv)가 쓸 수 있어야 합니다.

표 20–6에서는 샘플 백업 그룹 디렉토리 구조를 보여 줍니다.

```
/backup/primary/groupA/amy
                       /bob
                       /carly
                /groupB/mary
                       /nancy
                       /zelda
                /groupC/123go
                       /1bill
                       /354hut
```

아래 예에서는 /nsr/res 디렉토리에 있는 IMS.res라는 샘플 res 파일을 보여 줍니다.

```
type: savepnpc;
precmd: "echo mkbackupdir started",
   "/usr/siroe/server5/msg-siroe/bin/mkbackupdir.sh -p /backup";
pstcmd: "echo imsbackup Completed";
timeout: "12:00 pm";
```

이제 다음과 같이 Legato Networker 인터페이스를 실행할 준비가 되었습니다.

5 필요한 경우 Messaging Server 저장 그룹을 만듭니다.

a. nwadmin을 실행합니다.

b. 사용자 정의 | 그룹 | 만들기를 선택합니다.

6 **다음과 같이 savepnpc를 백업 명령으로 사용하여 백업 클라이언트를 만듭니다.**

a. **saveset를 mkbackupdir로 만든 디렉토리로 설정합니다.**

단일 세션 백업의 경우 /backup을 사용합니다.

병렬 백업의 경우 /backup/*server*/*group*을 사용합니다.

615 페이지 "20.12.2 백업 그룹 만들기"에 정의된 대로 이미 *group*이 만들어졌는지 확인합니다.

백업 세션 수에 대해서도 병렬을 설정해야 합니다.

621 페이지 "Legato Networker를 사용하여 데이터를 백업하는 방법"을 참조하십시오.

7 **그룹 제어 | 시작을 선택하여 백업 구성을 테스트합니다.**

예: Networker에서 백업 클라이언트 만들기

Networker에서 백업 클라이언트를 만들려면 nwadmin에서 클라이언트 | 클라이언트 설치 | 만들기를 선택합니다.

```
Name: siroe
Group: IMS
Savesets:/backup/primary/groupA
   /backup/secondary/groupB
   /backup/tertiary/groupC
         .
         .
Backup Command:savepnpc
Parallelism: 4
```

20.12.6.1 Legato Networker를 사용하여 데이터 복원

Legato Networker nwrecover 인터페이스나 recover 명령줄 유틸리티를 사용하여 데이터를 복구할 수 있습니다. 다음 예에서는 사용자 a1의 INBOX를 복구합니다.

```
recover -a -f -s siroe /backup/siroe/groupA/a1/INBOX
```

다음 예에서는 전체 메시지 저장소를 복구합니다.

```
recover -a -f -s siroe /backup/siroe
```

20.12.7 Legato를 제외한 타사 소프트웨어 사용

Messaging Server는 두 개의 메시지 저장소 백업 솔루션인 명령줄 imsbackup 및 Solstice Backup(Legato Networker)을 제공합니다. 단일 imbackup을 실행하여 전체 메시지 저장소를 백업하는 대량 메시지 저장소는 매우 많은 시간이 소요될 수 있습니다. Legato

솔루션은 여러 백업 장치에서의 동시 백업 세션을 지원합니다. 동시 백업으로 백업 시간을 대폭 단축시킬 수 있습니다(시간당 25GB의 데이터 백업 가능).

타사 동시 백업 소프트웨어(예: Netbackup)를 사용하는 경우 다음 방법을 사용하여 백업 소프트웨어를 Messaging Server와 통합할 수 있습니다.

▼ Legato를 제외한 타사 소프트웨어 사용

1 **사용자를 그룹으로 분할하고(615 페이지 "20.12.2 백업 그룹 만들기" 참조) *msg-svr-base*/config/ 디렉토리 아래에 backup-groups.conf 파일을 만듭니다.**

주 – 이 백업 솔루션에는 추가 디스크 공간이 필요합니다. 모든 그룹을 동시에 백업하려는 경우 디스크 공간 요구 사항은 메시지 저장소 크기의 2배입니다. 디스크 공간이 충분치 않을 경우에는 사용자를 더 작은 그룹으로 분할한 다음 그룹 집합을 한꺼번에group1 - group5, group6 - group10 등과 같이 백업합니다. 백업 후에 그룹 데이터 파일을 제거합니다.

---

2 **imsbackup을 실행하여 스테이징 영역에서 각 그룹을 파일로 백업합니다.**

명령은 imsbackup -f <device> /<instance>/<group>입니다.

여러 imsbackup 프로세스를 동시에 실행할 수 있습니다. 예를 들면 다음과 같습니다.

```
# imsbackup -f- /primary/groupA > /bkdata/groupA &
# imsbackup -f- /primary/groupB > /bkdata/groupB &
. . .
```

imsbackup은 큰 파일을 지원하지 않으므로 백업 데이터가 2GB 이상일 경우 -f- 옵션을 사용하여 데이터를 stdout에 기록한 다음 출력을 파일로 파이프해야 합니다.

3 **타사 백업 소프트웨어를 사용하여 스테이징 영역(이 예에서는 /bkdata)에서 그룹 데이터 파일을 백업합니다.**

4 **사용자를 복원하려면 사용자의 그룹 파일 이름을 식별하고 테이프에서 해당 파일을 복원한 다음 imsrestore를 사용하여 데이터 파일에서 사용자를 복원합니다.**

imsrestore는 큰 파일을 지원하지 않습니다. 데이터 파일이 2GB 이상일 경우 다음 명령을 사용합니다.

```
# cat /bkdata/groupA | imsrestore -f- /primary/groupA/andy
```

20.12.8 백업 및 복원 문제 해결

이 절에서는 일반적인 백업 및 복원 문제와 그 해결 방법에 대해 설명합니다.

- **문제**: imsrestore 또는 imsasm을 사용하여 폴더 또는 받은 메일함을 복원할 때 해당 폴더의 모든 메시지가 현재 폴더에 추가됩니다. 이로 인해서 해당 폴더에 메시지의 복사본이 여러 개 있게 됩니다.

솔루션: imsasm 스크립트에 imsrestore의 -i 플래그가 설정되어 있지 않도록 합니다.

- **문제**: 메일 폴더에 추가된 새 메시지만 증분 백업하려고 하는데 전체 폴더가 백업됩니다. 새 메시지만 백업하려면 어떻게 해야 합니까?

  솔루션: imsbackup에 -d *datetime* 플래그를 설정합니다. 이렇게 하면 지정된 날짜 및 시간에서 현재까지 저장된 메시지가 백업됩니다. 기본값은 날짜와 상관없이 모든 메시지를 백업하는 것입니다.

20.12.9 메시지 저장소 재해 복구 및 복원

재해는 메일함 하나 또는 메일함 집합이 아닌 전체 메시지 저장소에 심각한 문제가 발생한 경우를 말합니다. 즉, 메시지 저장소 서버의 모든 데이터가 손실되는 경우가 이에 해당합니다. 다음의 손실 데이터를 복원하면 메시지 저장소 재해를 완전히 복원할 수 있습니다.

- 모든 메시지 저장소 데이터. 이 데이터는 614 페이지 "20.12 메시지 저장소 백업 및 복원"에서 설명하는 절차를 사용하여 백업할 수 있습니다. 파일 시스템 백업 방법을 사용하는 경우에는 다음 데이터를 백업해야 합니다.
  - 모든 메시지 저장소 분할 영역
  - msg-svr-base/data/store/mboxlist에 있는 메시지 저장소 데이터베이스 파일
  - msg-svr-base/data/store/dbdata/snapshots에 있는 메시지 저장소 데이터베이스 스냅샷(메시지 저장소 데이터베이스 스냅샷 파일의 위치는 configutil 매개 변수 local.store.snapshotpath로 구성 가능)
- 구성 데이터. msg-svr-base/data/config에 있는 로컬 구성 파일 포함

재해 복구를 위해 메시지 저장소를 백업하려면 파일 시스템 스냅샷 도구를 사용하여 파일 시스템의 스냅샷을 만듭니다. 스냅샷은 **반드시 한 시점의** 파일 시스템 스냅샷이어야 합니다. 그렇지 않으면 mboxlist 백업을 사용할 수 없습니다(전체 데이터베이스 스냅샷에서 mboxlist 데이터베이스를 복원해야 함).

모든 데이터(메시지 저장소 분할 영역, 데이터베이스 파일 등)를 같은 시점에 캡처하는 것이 가장 좋지만 그럴 수 없는 경우에는 다음과 같은 순서로 데이터를 백업해야 합니다.

1. 데이터베이스 스냅샷
2. 데이터베이스 파일
3. 메시지 저장소 분할 영역
4. 구성 데이터

메시지 저장소 분할 영역과 데이터베이스 파일을 같은 시점의 스냅샷으로 백업하지 않은 경우에는 파일 시스템 스냅샷을 복원한 후에 reconstruct -m을 실행합니다. 그러면 데이터베이스와 저장소 분할 영역이 동기화됩니다.

20.13 사용자 액세스 모니터링

Messaging Server는 IMAP, POP 및 http를 통해 사용자의 메시지 저장소 액세스를 모니터링할 수 있는 `imsconnutil` 명령을 제공합니다. 또한 사용자의 마지막 로그인 및 로그아웃을 확인할 수 있습니다. 이 명령은 메시지 저장소별로 작동하므로 여러 메시지 저장소에 대해 사용할 수 없습니다.

주 – 이 기능 또는 기타 Messaging Server 기능을 사용하여 사용자의 전자 메일을 모니터링, 읽기 또는 액세스할 때 해당 법률이나 규칙을 위반하거나 고객 정책 또는 계약을 위반할 경우 책임이 따를 수 있습니다.

이 명령을 사용하려면 시스템 사용자(`mailsrv`)가 루트로 액세스해야 하며 구성 변수 `local.imap.enableuserlist`, `local.http.enableuserlist`, `local.enablelastaccess`를 1로 설정해야 합니다.

IMAP 또는 웹 메일 클라이언트를 통해 현재 로그온한 사용자를 나열하려면 다음 명령을 사용합니다.

 # imsconnutil -c

메시지 저장소에 있는 모든 사용자의 마지막 IMAP, POP, 또는 Messenger Express 액세스(로그인 및 로그아웃)를 나열하려면 다음을 사용합니다.

 # imsconnutil -a

다음 명령은 1) 지정된 사용자가 IMAP나 Messenger Express 또는 mshttp를 통해 연결된 임의의 클라이언트(POP 사용자는 대개 연결 상태를 유지하지 않으므로 POP는 해당되지 않음)를 사용하여 현재 로그온했는지 확인하고 2) 사용자가 마지막으로 로그온 및 로그오프한 시간을 나열하는 두 가지 작업을 수행합니다.

 # imsconnutil -c -a -u *user_ID*

다음 명령을 사용하여 사용자 목록이 파일에 한 행당 하나씩 입력될 수 있습니다.

 # imsconnutil -c -a -f *filename*

또한 `-s` 플래그를 사용하여 특정 서비스(`imap` 또는 `http`)를 지정할 수도 있습니다. 예를 들어, 특정 사용자 아이디가 IMAP에 로그온했는지 여부를 나열하려면 다음 명령을 사용합니다.

 # imsconnutil -c -s imap -u *user_ID*

`-k` 옵션은 IMAP IDLE가 구성된 경우에만 작동합니다. `imsconnutil` 구문에 대한 자세한 설명은 **Sun Java System Messaging Server 6.3 Administration Reference** 의 “imsconnutil”을 참조하십시오.

다음은 몇 가지 출력 예입니다.

```
$ ./imsconnutil -a -u soroork

UID     IMAP last access    HTTP last access    POP last access
================================================================================
ed   08/Jul/2003:10:49:05   10/Jul/2003:14:55:52  ---NOT-RECORDED---

$ ./imsconnutil -c

IMAP
UID    TIME                 AUTH            TO                  FROM
================================================================================
ed   17/Jun/2003:11:24:03  plain     172.58.73.45:193   129.157.12.73:2631
bil  17/Jun/2003:04:28:43  plain     172.58.73.45:193   129.158.16.34:2340
mia  17/Jun/2003:09:36:54  plain     172.58.73.45:193   192.18.184.103:3744
jay  17/Jun/2003:05:38:46  plain     172.58.73.45:193   129.159.18.123:3687
pau  17/Jun/2003:12:23:28  plaintext 172.58.73.45:193   192.18.194.83:2943
ton  17/Jun/2003:05:38:46  plain     172.58.73.45:193   129.152.18.123:3688
ani  17/Jun/2003:12:26:40  plaintext 172.58.73.45:193   192.18.164.17:1767
ani  17/Jun/2003:12:25:17  plaintext 172.58.73.45:193   129.150.17.34:3117
jac  17/Jun/2003:12:26:32  plaintext 172.58.73.45:193   129.150.17.34:3119
ton  17/Jun/2003:12:25:32  plaintext 172.58.73.45:193   192.18.148.17:1764
================================================================================
10 users were logged in to imap.
Feature is not enabled for http.
--------------------------------------------------------------------------------
```

20.14 메시지 저장소 문제 해결

이 절에서는 메시지 저장소를 능동적으로 유지 관리하기 위한 지침을 제공합니다. 또한 이 절에서는 메시지 저장소가 손상되었거나 예기치 않게 종료된 경우 사용할 수 있는 다른 메시지 저장소 복구 절차에 대해 설명합니다. 이 추가 메시지 저장소 복구 절차에 대한 절은 633 페이지 “20.14.3 메일함 및 메일함 데이터베이스 복구”를 확장한 것입니다.

이 절을 읽기 전에 이 장과 함께 Sun Java System Messaging Server Administration Reference의 명령줄 유틸리티 및 `configutil`에 대한 장을 검토하는 것이 좋습니다. 이 절은 다음 내용으로 구성되어 있습니다.

- 627 페이지 “20.14.1 표준 메시지 저장소 모니터링 절차”
- 630 페이지 “20.14.2 메시지 저장소 시작 및 복구”
- 633 페이지 “20.14.3 메일함 및 메일함 데이터베이스 복구”
- 637 페이지 “20.14.4 일반 문제 및 해결 방법”

20.14.1 표준 메시지 저장소 모니터링 절차

이 절에서는 메시지 저장소의 표준 모니터링 절차에 대해 간략하게 설명합니다. 이러한 절차는 일반적인 메시지 저장소 검사, 테스트 및 표준 유지 관리에 유용합니다.

자세한 내용은 841 페이지 "27.7 메시지 저장소 모니터링"을 참조하십시오.

20.14.1.1 하드웨어 공간 검사

메시지 저장소에는 충분한 추가 디스크 공간과 하드웨어 자원이 있어야 합니다. 메시지 저장소가 디스크 및 하드웨어 공간의 최대 한도에 가까이 도달하면 메시지 저장소 내에 문제가 발생할 수 있습니다.

디스크 공간 부족은 메일 서버 문제 및 오류의 가장 일반적인 원인 중 하나입니다. 메시지 저장소에 쓰기 위한 공간이 없을 경우 메일 서버에서 오류가 발생합니다. 또한 사용 가능한 디스크 공간이 일정한 임계값 아래로 내려가면 메시지 전달, 로깅 등과 관련된 문제가 발생합니다. `stored` 프로세스의 정리 기능이 실패하고 삭제된 메시지가 메시지 저장소에서 정리되지 않으면 디스크 공간이 급속도로 줄어들 수 있습니다.

디스크 공간 모니터링에 대한 자세한 내용은 610 페이지 "20.11.5 디스크 공간 모니터" 및 841 페이지 "27.7 메시지 저장소 모니터링"을 참조하십시오.

20.14.1.2 로그 파일 검사

로그 파일을 검사하여 메시지 저장소 프로세스가 구성된 대로 실행되는지 확인합니다. Messaging Server는 지원되는 각각의 주요 프로토콜 또는 서비스(SMTP, IMAP, POP 및 HTTP)에 대한 별도의 로그 파일집합을 만듭니다. *msg-svr-base*`/log/` 디렉토리에서 로그 파일을 볼 수 있습니다. 정기적으로 로그 파일을 모니터해야 합니다.

로깅이 서버 성능에 영향을 줄 수 있다는 것을 유의하십시오. 더 자세한 로깅을 지정할수록 일정한 시간 동안 로그 파일이 차지하는 디스크 공간이 더 많아집니다. 따라서 효과적이면서 실제적인 로그 회전, 만료 및 백업 정책을 서버에 정의해야 합니다. 서버의 로깅 정책 정의에 대한 자세한 내용은 25 장을 참조하십시오.

20.14.1.3 원격 측정을 사용하여 사용자 IMAP/POP/Webmail 세션 검사

Messaging Server는 사용자의 전체 IMAP, POP 또는 HTTP 세션을 파일로 캡처할 수 있는 원격 측정이라는 기능을 제공합니다. 이 기능은 클라이언트 문제를 디버깅하는 데 유용합니다. 예를 들어, 사용자가 메시지 액세스 클라이언트가 제대로 작동하지 않는다고 불평할 경우 이 기능을 사용하여 액세스 클라이언트와 Messaging Server 사이의 상호 작용을 추적할 수 있습니다.

POP 세션을 캡처하려면 다음 디렉토리를 만듭니다.

msg-svr-base`/data/telemetry/`*pop_or_imap_or_http*`/`*userid*

POP 세션을 캡처하려면 다음 디렉토리를 만듭니다.

msg-svr-base`/data/telemetry/pop/`*userid*

IMAP 세션을 캡처하려면 다음 디렉토리를 만듭니다.

msg-svr-base`/data/telemetry/imap/`*userid*

Webmail 세션을 캡처하려면 다음 디렉토리를 만듭니다.

msg-svr-base/data/telemetry/http/*userid*

디렉토리는 Messaging Server 사용자 아이디에서 소유하며 쓰기를 수행할 수 있어야 합니다.

Messaging Server는 이 디렉토리에서 세션당 하나의 파일을 만듭니다. 출력 예는 다음과 같습니다.

```
LOGIN redb 2003/11/26 13:03:21
>0.017>1 OK User logged in
<0.047<2 XSERVERINFO MANAGEACCOUNTURL MANAGELISTSURL MANAGEFILTERSURL
>0.003>* XSERVERINFO MANAGEACCOUNTURL {67}
http://redb@cuisine.blue.planet.com:800/bin/user/admin/bin/enduser
MANAGELISTSURL NIL MANAGEFILTERSURL NIL
2 OK Completed
<0.046<3 select "INBOX"
>0.236>* FLAGS (\Answered flagged draft deleted \Seen $MDNSent Junk)
* OK [PERMANENTFLAGS (\Answered flag draft deleted \Seen $MDNSent Junk \*)]
* 1538 EXISTS
* 0 RECENT
* OK [UNSEEN 23]
* OK [UIDVALIDITY 1046219200]
* OK [UIDNEXT 1968]
3 OK [READ-WRITE] Completed
<0.045<4 UID fetch 1:* (FLAGS)
>0.117>* 1 FETCH (FLAGS (\Seen) UID 330)
* 2 FETCH (FLAGS (\Seen) UID 331)
* 3 FETCH (FLAGS (\Seen) UID 332)
* 4 FETCH (FLAGS (\Seen) UID 333)
* 5 FETCH (FLAGS (\Seen) UID 334)
<etc>
```

원격 측정 기록을 비활성화하려면 만들었던 디렉토리를 옮기거나 제거합니다.

20.14.1.4 stored 프로세스 검사

stored 함수는 메시지 데이터베이스의 교착 상태 및 트랜잭션 작업, 에이징 정책 적용, 디스크에 저장된 메시지 정리 및 삭제와 같은 여러 중요한 작업을 수행합니다. stored의 실행이 중지되면 Messaging Server에서 문제가 발생합니다. start-msg가 실행될 때 stored가 시작되지 않을 경우 다른 프로세스는 시작되지 않습니다.

- stored 프로세스가 실행 중인지 확인합니다. imcheck를 실행합니다.
- *store_root*/mboxlist에서 작성된 로그 파일을 확인합니다.
- 기본 로그 파일 *msg-svr-base*/log/default/default에서 stored 메시지를 확인합니다.
- stored 프로세스에서 다음 기능 중 하나를 시도할 때마다 *msg-svr-base*/config/ 디렉토리에 있는 다음 파일의 타임스탬프가 업데이트되는지 확인합니다.

표 20–12 stored 작업

stored 작업	기능
stored.ckp	데이터베이스 검사점이 시작될 때 수정됩니다. 약 1분마다 시간이 기록됩니다.
stored.lcu	데이터베이스 로그가 정리될 때마다 수정됩니다. 약 5분마다 시간이 기록됩니다.
stored.per	사용자 단위 db 쓰기가 생성될 때마다 수정됩니다. 1시간에 한 번씩 시간이 기록됩니다.

stored 프로세스에 대한 자세한 내용은 **Sun Java System Messaging Server 6.3 Administration Reference**의 610 페이지 “20.11.6 stored 데몬” 장을 참조하십시오.

stored 함수 모니터링에 대한 자세한 내용은 841 페이지 “27.7 메시지 저장소 모니터링”을 참조하십시오.

20.14.1.5 데이터베이스 로그 파일 검사

데이터베이스 로그 파일은 *store_root* /mboxlist 디렉토리에 있는 sleepycat 트랜잭션 검사점 지정 로그 파일을 나타냅니다. 로그 파일이 누적될 경우 데이터베이스 검사점 지정이 발생하지 않습니다. 일반적으로 단일 기간 동안 둘 또는 세 개의 데이터베이스 로그 파일이 존재합니다. 파일이 더 많이 있을 경우는 문제가 발생한 것일 수 있습니다.

20.14.1.6 사용자 폴더 검사

사용자 폴더를 검사하려는 경우 모든 사용자 폴더를 검토하고 오류를 보고하는 reconstruct -r -n(재귀적 수정 없음) 명령을 실행할 수 있습니다. reconstruct 명령에 대한 자세한 내용은 633 페이지 “20.14.3 메일함 및 메일함 데이터베이스 복구”를 참조하십시오.

20.14.1.7 코어 파일 검사

코어 파일은 프로세스가 예기치 않게 종료된 경우에만 존재합니다. 특히 메시지 저장소에 문제가 있을 경우 이러한 파일을 검토하는 것이 중요합니다. Solaris에서는 coreadm을 사용하여 core 파일 위치를 구성합니다.

20.14.2 메시지 저장소 시작 및 복구

메시지 저장소 데이터는 메시지, 색인 데이터 및 메시지 저장소 데이터베이스로 구성됩니다. 이 데이터는 상당히 견고하지만 가끔 시스템에 메시지 저장소 데이터 문제가 발생할 수 있습니다. 이러한 문제는 기본 로그 파일에서 나타나며 그 대부분이 항상 투명하게 수정됩니다. 아주 드물게 로그 파일의 오류 메시지가 reconstruct 유틸리티를 실행해야 한다는 것을 나타낼 수 있습니다. 또한 메시지를 보호하기 위한

마지막 수단으로 614 페이지 “20.12 메시지 저장소 백업 및 복원”에 설명된 백업 및 복원 프로세스가 사용됩니다. 이 절에서는 stored 의 자동 시작 및 복구 프로세스를 중심으로 설명합니다.

메시지 저장소는 이전에 관리자가 담당하던 많은 복구 작업을 자동화합니다. 이러한 작업은 시작 시에 메시지 저장소 데몬 stored에 의해 수행되며 필요에 따라 데이터베이스 스냅샷 및 자동 고속 복구를 포함합니다. stored는 메시지 저장소의 데이터베이스를 철저하게 검사하여 문제가 감지된 경우 이를 자동으로 복구합니다.

stored는 또한 상태 메시지를 통해 포괄적인 데이터베이스 상태 분석을 기본 로그에 제공하여 메시지 저장소에 대해 수행된 복구 작업과 메시지 저장소를 작동시키기 위한 자동 시도를 보고합니다.

20.14.2.1 자동 시작 및 복구—작동 원리

stored 데몬은 다른 메시지 저장소 프로세스보다 먼저 시작됩니다. 이 데몬은 메시지 저장소 데이터베이스를 초기화하고 필요한 경우 복구합니다. 메시지 저장소 데이터베이스는 폴더, 할당량, 가입 및 메시지 플래그 정보를 보관합니다. 데이터베이스는 로깅 및 트랜잭션 가능하여 이미 복구가 내장되어 있습니다. 또한 일부 데이터베이스 정보는 각 폴더의 메시지 색인 영역에서 중복하여 복사됩니다.

데이터베이스는 상당히 견고하지만 가끔씩 손상될 수 있으며 stored는 대부분의 경우 이 문제를 투명하게 복구합니다. 그러나 stored를 다시 시작할 때마다 기본 로그 파일을 검사하여 추가 관리 개입이 필요하지 않은지 확인해야 합니다. 데이터베이스의 추가 재작성이 필요한 경우 로그 파일의 상태 메시지에 reconstruct를 실행하라는 내용이 나타납니다.

메시지 저장소 데이터베이스를 열기 전에 stored는 무결성을 분석하고 **경고** 범주에 속하는 상태 메시지를 기본 로그로 보냅니다. 일부 메시지는 관리자에게 유용하며 다른 일부는 내부 분석에 사용되는 코딩된 데이터로 구성됩니다. stored는 문제를 감지하면 데이터베이스를 수정하고 재시작을 시도합니다.

데이터베이스가 열리면 stored는 나머지 서비스를 시작할 수 있다는 것을 알립니다. 자동 수정이 실패할 경우 기본 로그의 메시지는 수행할 작업을 지정합니다. 632 페이지 “reconstruct가 필요하다는 것을 지정하는 오류 메시지”를 참조하십시오.

이전 릴리스에서는 stored에서 복구 프로세스를 구현하는 데 매우 오래 걸려서 관리자가 stored를 "중단"된 것으로 여기기도 했습니다.이제 이러한 긴 복구 유형은 제거되었으며 stored는 1분 이내에 최종 상태를 확인해야 합니다. 그러나 stored가 스냅샷 복구와 같은 복구 기술을 사용해야 할 경우 프로세스는 몇 분 정도가 소요될 수 있습니다.

대부분의 경우 복구가 수행된 후에 데이터베이스는 최신 상태로 업데이트되며 다른 작업은 필요하지 않습니다. 그러나 일부 복구는 메시지 저장소의 중복 데이터를 동기화하기 위해 reconstruct -m이 필요합니다. 이러한 내용도 기본 로그에 표시되므로 시작 후에 기본 로그를 모니터하는 것이 중요합니다. 메시지 저장소가 시작되어 정상적으로 실행되는 것처럼 보이는 경우에도 reconstruct와 같은 요청된 모든 작업을 실행해야 합니다.

로그 파일을 읽어야 하는 또 다른 이유는 처음에 데이터베이스를 손상시킨 원인을 확인하는 데 있습니다. stored는 시스템상의 문제와 무관하게 메시지 저장소를 사용하도록 설계되었지만 데이터베이스 손상이 숨겨진 더 큰 문제의 일부일 수 있으므로 그 원인을 확인하는 것이 필요합니다.

reconstruct가 필요하다는 것을 지정하는 오류 메시지

이 절에서는 reconstruct를 실행해야 하는 오류 메시지 유형에 대해 설명합니다.

오류 메시지가 메일함 오류를 나타내면 reconstruct <mailbox>를 실행합니다. 예:

```
"Invalid cache data for msg 102 in mailbox user/joe/INBOX. Needs reconstruct"

"Mailbox corrupted, missing fixed headers: user/joe/INBOX"

"Mailbox corrupted, start_offset beyond EOF: user/joe/INBOX"
```

오류 메시지가 데이터베이스 오류를 나타내면 reconstruct -m을 실행합니다. 예:

```
"Removing extra database logs. Run reconstruct -m soon after startup to resync
redundant data"

"Recovering database from snapshot. Run reconstruct -m soon after startup to
resync redundant data"
```

데이터베이스 스냅샷

스냅샷은 데이터베이스의 핫 백업으로 stored에서 손상된 데이터베이스를 몇 분 안에 투명하게 복원하기 위해 사용합니다. 이것은 다른 영역에 저장된 중복된 정보에 의존하는 reconstruct를 사용하는 것보다 훨씬 더 빠릅니다.

메시지 저장소 데이터베이스 스냅샷—작동 원리

mboxlist 디렉토리에 있는 데이터베이스의 스냅샷은 기본적으로 24시간에 한 번씩 자동으로 생성됩니다. 스냅샷은 기본적으로 store 디렉토리의 하위 디렉토리에 복사됩니다. 언제든지 기본적으로 다섯 개의 스냅샷(라이브 데이터베이스 하나, 스냅샷 세 개, 데이터 베이스/제거된 복사본 하나)이 있습니다. 데이터베이스/제거된 복사본이 가장 최신 버전이며 mboxlist 데이터베이스 디렉토리의 removed 하위 디렉토리로 보내지는 데이터베이스의 긴급 복사본입니다.

현재 데이터베이스가 손상된 것으로 확인되어 복구 프로세스에서 이를 제거하기로 결정하면 stored는 가능한 경우 해당 데이터베이스를 removed 디렉토리로 이동합니다. 이렇게 하면 필요에 따라 데이터베이스를 분석할 수 있습니다.

데이터 이동은 일주일에 한 번만 발생합니다. 따라서 이미 데이터베이스 복사본이 있는 경우에는 stored에서 저장소를 만들 때마다 데이터베이스 복사본을 교체하지 않습니다. stored는 removed 디렉토리의 데이터가 1주일보다 오래된 경우에만 교체를 수행합니다. 이것은 문제가 있는 원래 데이터베이스가 계속적인 시작으로 인해 너무 빨리 대체되는 것을 방지합니다.

메시지 저장소 데이터베이스 스냅샷 간격 및 위치 지정

결합된 데이터베이스와 스냅샷의 5배에 해당하는 공간이 있어야 합니다. 관리자는 스냅샷을 별개의 디스크에서 실행되도록 다시 구성하고 시스템 요구에 맞게 조정하는 것이 좋습니다.

stored가 시작 시에 데이터베이스 문제를 감지할 경우 최적의 스냅샷이 자동으로 복구됩니다. 세 개의 스냅샷 변수는 스냅샷 파일 위치, 스냅샷 촬영 간격, 저장되는 스냅샷 수 등과 같은 매개 변수를 설정할 수 있습니다. 이러한 configutil 매개 변수는 표 20–13에 나와 있습니다.

스냅샷 간격이 너무 작으면 시스템에 자주 부담을 주게 되며 데이터베이스의 문제가 스냅샷으로 복사될 가능성이 더 커집니다. 스냅샷 간격이 너무 크면 스냅샷을 가져왔을 때 갖고 있던 상태를 데이터베이스가 계속 보유하는 상황이 발생할 수 있습니다.

스냅샷 간격으로 1일이 권장되지만 시스템상에 문제가 수일 동안 지속되거나 문제가 존재하기 전의 시점으로 되돌아가려는 경우 일주일 이상의 간격이 유용할 수 있습니다.

stored는 데이터베이스를 모니터하여 데이터베이스가 완전하지 않다고 의심될 경우 최신 스냅샷을 거부하며 그 대신에 가장 안정적인 최신 스냅샷을 검색합니다. 하루 전의 스냅샷을 검색할 수 있다는 사실에도 불구하고 시스템은 보다 최신의 중복 데이터가 있는 경우 이를 사용하며 이전 스냅샷 데이터를 무시합니다.

따라서 스냅샷의 궁극적인 역할은 시스템을 가능한 최신 상태로 유지하고 데이터를 즉석에서 재작성하려고 시도하는 시스템의 나머지 부분에 대한 부담을 줄여 주는 것입니다.

표 20–13 메시지 저장소 데이터베이스 스냅샷 매개 변수

매개 변수	설명
local.store.snapshotpath	메시지 저장소 데이터베이스 스냅샷 파일의 위치입니다. 기존 절대 경로 또는 store 디렉토리에 대한 상대 경로입니다. 기본값: dbdata/snapshots
local.store.snapshotinterval	스냅샷 간격(분)입니다. 유효한값1 - 46080 기본값: 분일
local.store.snapshotdirs	보관되는 다른 스냅샷 수입니다. 유효한값2 -367 기본값: 3

20.14.3 메일함 및 메일함 데이터베이스 복구

하나 이상의 메일함이 손상되면 reconstruct 유틸리티를 사용하여 메일함 또는 메일함 데이터베이스를 다시 작성하고 모든 불일치를 복구할 수 있습니다.

reconstruct 유틸리티는 하나 이상의 메일함이나 마스터 메일함 파일을 다시 작성하고 모든 불일치를 복구합니다. 이 유틸리티를 사용하면 메시지 저장소에서 거의 모든 형태의 데이터 손상을 복구할 수 있습니다. 632 페이지 “reconstruct가 필요하다는 것을 지정하는 오류 메시지”를 참조하십시오.

표 20–14에는 reconstruct 옵션이 나열되어 있습니다. 구문 및 사용 요구 사항에 대한 자세한 내용은 **Sun Java System Messaging Server 6.3 Administration Reference**의 “reconstruct”를 참조하십시오.

표 20–14 reconstruct 옵션

옵션	설명
-e	재구성하기 전에 store.exp 파일을 제거합니다. 이렇게 하면 저장 프로세스에서 정리하지 않은 제거된 메시지에 대한 모든 내부 저장소 레코드가 제거됩니다. -i 또는 -e 옵션은 폴더가 실제로 재구성될 때만 작동하므로 이러한 옵션을 사용할 때 -f 옵션을 사용하는 것도 유용합니다. 마찬가지로 재구성이 아닌 검사를 수행하는 -n 옵션을 사용할 경우 -i 및 -e 옵션이 작동하지 않습니다. reconstruct가 손상을 감지하지 못하는 경우 reconstruct -e를 실행하면 제거된 메시지가 복구되지 않습니다. -f는 재구성을 실행합니다.
-i	재구성 전에 store.idx 파일 길이를 0으로 설정합니다. -i 또는 -e 옵션은 폴더가 실제로 재구성될 때만 작동하므로 이러한 옵션을 사용할 때 -f 옵션을 사용하는 것도 유용합니다. 마찬가지로 재구성이 아닌 검사를 수행하는 -n 옵션을 사용할 경우 -i 및 -e 옵션이 작동하지 않습니다.
-f	reconstruct를 수행하여 메일함을 수정합니다.
-l	lright.db를 재구성합니다.
-m	일관성 검사를 수행하고 필요한 경우 메일함 데이터베이스를 복구합니다. 이 옵션은 스풀 영역에서 찾은 모든 메일함을 검사하고 메일함 데이터베이스에서 적절하게 항목을 추가 또는 제거합니다. 데이터베이스에서 항목을 추가 또는 제거할 때마다 표준 출력 파일에 메시지가 인쇄됩니다. 특히 folder.db, quota.db 및 lright.db를 수정합니다.
-n	메일함을 수정하지 않고 메시지 저장소만 검사합니다. 메일함 이름을 제공하지 않을 경우 -n 옵션을 단독으로 사용할 수 없습니다. 메일함 이름을 제공하지 않을 때는 -n 옵션을 -r 옵션과 함께 사용해야 합니다. -r 옵션은 -p 옵션과 함께 사용할 수 있습니다. 예를 들어, 다음 명령은 모두 유효합니다. reconstruct -n user/dulcinea/INBOX reconstruct -n -r reconstruct -n -r -p primary reconstruct -n -r user/dulcinea/
-o	폐기되었습니다. mboxutil -o를 참조하십시오.
-o -d *filename*	폐기되었습니다. mboxutil -o를 참조하십시오.

표 20–14 reconstruct 옵션 (계속)

옵션	설명
-p *partition*	-p 옵션은 -m 옵션과 함께 사용되며 재구성의 범위를 지정한 분할 영역으로 제한합니다. -p 옵션을 지정하지 않을 경우 reconstruct에서 모든 분할 영역이 기본값이 됩니다. 특히 folder.db 및 quota.db를 수정하지만 lright.db는 수정하지 않습니다. 이는 lright.db를 수정하려면 메시지 저장소의 모든 사용자에 대한 acl을 스캔해야 하기 때문입니다. 모든 분할 영역에 대해 이 작업을 수행하는 것은 그리 효율적이지 않습니다. lright.db를 수정하려면 reconstruct -l을 실행합니다. 분할 영역 이름을 지정하며 전체 경로 이름을 사용하면 안 됩니다.
-q	할당량 하위 시스템의 모든 불일치(예: 잘못된 할당량 루트를 가진 메일함 또는 잘못된 할당량 사용이 보고된 할당량 루트)를 수정합니다. 다른 서버 프로세스가 실행되는 동안 -q 옵션을 실행할 수 있습니다.
-r [*mailbox*]	지정된 메일함의 분할 영역에 대한 일관성 검사를 복구 및 수행합니다. -r 옵션은 또한 지정된 메일함 내의 모든 하위 메일함을 복구합니다. 메일함 인수 없이 -r을 지정할 경우 사용자 분할 영역 디렉토리에 있는 모든 메일함의 스풀 영역이 복구됩니다.
-u *user*	-u 옵션은 -m 옵션과 함께 사용되며 재구성의 범위를 지정한 사용자로 제한합니다. -u 옵션은 -p 옵션과 함께 사용해야 합니다. -u 옵션을 지정하지 않을 경우 모든 분할 영역이나 -p 옵션을 사용하여 지정한 분할 영역이 reconstruct에서 기본값이 됩니다. 분할 영역 이름을 지정하며 전체 경로 이름을 사용하면 안 됩니다.

20.14.3.1 메일함 재작성

메일함을 다시 작성하려면 -r 옵션을 사용합니다. 다음 경우에 이 옵션을 사용해야 합니다.

- 메일함에 액세스하면 “System I/O error” 또는 “Mailbox has an invalid format” 오류 중 하나가 반환됩니다.
- 메일함에 액세스할 때 서버 충돌이 발생하는 경우
- 스풀 디렉토리에서 파일이 추가 또는 제거된 경우

reconstruct -r은 우선 일관성 검사를 실행합니다. 이 검사는 모든 비일관성을 보고하며 문제가 감지된 경우에만 재작성을 수행합니다. 결과적으로 이 릴리스에서 reconstruct 유틸리티의 성능이 향상됩니다.

다음 예에 설명된 대로 reconstruct를 사용할 수 있습니다.

사용자 daphne에 속하는 메일함의 스풀 영역을 다시 작성하려면 다음 명령을 사용합니다.

```
reconstruct -r user/daphne
```

메일함 데이터베이스에 나열된 모든 메일함의 스풀 영역을 다시 작성하려면 다음 명령을 사용합니다.

```
reconstruct -r
```

대용량 메시지 저장소의 경우 메일함 데이터베이스에 나열된 모든 메일함의 스풀 영역을 다시 작성하는 것이 아주 오래 걸릴 수 있으므로 이 옵션은 신중하게 사용해야 합니다. 637 페이지 "20.14.3.3 reconstruct 성능"을 참조하십시오. 저장소에 여러 디스크를 사용하는 것이 보다 나은 오류 복구 방법일 수 있습니다. 디스크가 하나가 중지되었다고 전체 저장소가 중지되지는 않습니다. 디스크가 손상된 경우 다음과 같이 -p 옵션을 사용하여 저장소의 일부만 다시 작성하면 됩니다.

```
reconstruct -r -p subpartition
```

primary 분할 영역에 있을 경우에만 명령줄 인수에 나열된 메일함을 다시 작성하려면 다음 명령을 수행합니다.

```
reconstruct -p primary mbox1 mbox2 mbox3
```

primary 분할 영역에 있는 모든 메일함을 다시 작성할 필요가 없을 경우 다음 명령을 사용합니다.

```
reconstruct -r -p primary
```

reconstruct를 실행하여 일관성 검사를 수행하지 않고 폴더를 다시 작성하려면 -f 옵션을 사용합니다. 예를 들어, 다음 명령을 실행하여 사용자 폴더 daphne를 다시 구성합니다.

```
reconstruct -f -r user/daphne
```

모든 메일함을 수정하지 않고 검사하려면 다음과 같이 -n 옵션을 사용합니다.

```
reconstruct -r -n
```

20.14.3.2 메일함 검사 및 복구

메일함 데이터베이스의 고급 일관성 검사와 복구를 수행하려면 다음 명령을 사용합니다.

```
reconstruct -m
```

기본 분할 영역의 일관성 검사와 복구를 수행하려면 다음 명령을 사용합니다.

```
reconstruct -p primary -m
```

주 – -P 및 -m 플래그와 함께 reconstruct를 실행하면 lright.db가 수정되지 않습니다. 이는 lright.db를 수정하려면 메시지 저장소의 모든 사용자에 대한 ACL을 스캔해야 하기 때문입니다. 모든 분할 영역에 대해 이 작업을 수행하는 것은 그리 효율적이지 않습니다. lright.db를 수정하려면 reconstruct -l을 실행합니다.

---

john이라는 개별 사용자의 메일함에 대한 일관성 검사와 복구를 수행하려면 다음 명령을 사용합니다.

```
reconstruct -p primary -u john -m
```

다음 경우에 -m 옵션을 사용해야 합니다.

- 하나 이상의 디렉토리가 저장소 스풀 영역에서 제거되어 메일함 데이터베이스 항목 또한 제거해야 하는 경우
- 하나 이상의 디렉토리가 저장소 스풀 영역에서 복원되어 메일함 데이터베이스 항목을 또한 추가해야 하는 경우
- stored -d 옵션으로 데이터베이스의 일관성을 유지할 수 없는 경우

  stored -d 옵션으로 데이터베이스의 일관성을 유지할 수 없는 경우에는 다음 단계를 나열된 순서대로 수행합니다.

  - 모든 서버를 종료합니다.
  - *store_root*/mboxlist의 모든 파일을 제거합니다.
  - 서버 프로세스를 다시 시작합니다.
  - reconstruct -m을 실행하여 스풀 영역 내용에서 새 메일함 데이터베이스를 작성합니다.

20.14.3.3 reconstruct 성능

reconstruct가 작업을 수행하는 데 걸리는 시간은 다음 요소에 따라 달라집니다.

- 수행하는 작업의 종류 및 선택한 옵션
- 디스크 성능
- reconstruct -m 실행 시의 폴더 수
- reconstruct -r 실행 시의 메시지 수
- 메시지 저장소의 전체 크기
- 시스템이 실행 중인 다른 프로세스 및 시스템 사용량
- 진행 중인 POP, IMAP, HTTP 또는 SMTP 활동이 있는지 여부

reconstruct -r 옵션은 초기 일관성 검사를 수행합니다. 이 검사는 다시 작성해야 하는 폴더 수에 따라 reconstruct 성능을 향상시킵니다.

약 2400명의 사용자와 85GB의 메시지 저장소가 있으며 서버에 동시 POP, IMAP 또는 SMTP 활동이 있는 시스템에서 다음 성능이 확인되었습니다.

- reconstruct -m은 약 1시간이 걸렸습니다.
- reconstruct -r -f는 약 18시간이 걸렸습니다.

주 – 진행 중인 POP, IMAP, HTTP 또는 SMTP 활동을 서버에서 수행하지 않을 경우 reconstruct 작업에는 훨씬 더 적은 시간이 소요될 수 있습니다.

20.14.4 일반 문제 및 해결 방법

이 절에서는 다음과 같은 일반적인 메시지 저장소 문제와 해결 방법에 대해 설명합니다.

20.14.4.1 Linux - Messaging Server 패치 120230-08 IMAP, POP 및 HTTP 서버가 프로세스 당 세션 초과 때문에 시작되지 않음

이 패치를 설치한 후 Messaging Server를 설치하려 하면 IMAP, POP 및 HTTP 서버가 시작되지 않고 다음 예와 같은 오류 로그를 보낼 수 있습니다.

```
http server - log:
[29/May/2006:17:44:37 +051800] usg197 httpd[6751]: General Critical: Not enough file
descriptors to support 6000 sessions per process; Recommend ulimit -n 12851 or 87
sessions per process.

pop server - log:
[29/May/2006:17:44:37 +051800] usg197 popd[6749]: General Critical: Not enough file
descriptors to support 600 sessions per process; Recommend ulimit -n 2651 or 58
sessions per process.

Once these values setting in /opt/sun/messaging/sbin/configutil then imap server
failed to start

imap server - log:
[29/May/2006:17:44:37 +051800] usg197 imapd[6747]: General Critical: Not enough
file descriptors to support 4000 sessions per process; Recommend ulimit -n 12851
or 58 sessions per process.
```

세 서버 세션 모두에 적절한 수의 파일 설명자를 설정합니다. 다음과 비슷한 행을 `/etc/sysctl.conf`에 추가하고 `sysctl -p`를 사용하여 파일을 다시 읽으면 추가 파일 설명자를 사용할 수 있습니다.

```
fs.file-max = 65536
```

또한 다음과 같은 행을 `/etc/security/limits.conf`에 추가해야 합니다.

```
*   soft  nofile  65536
*   hard  nofile  65536
```

20.14.4.2 메일 페이지를 로드하지 않는 Messenger Express 또는 Communications Express

사용자가 Messenger Express 페이지 또는 Communications Express 메일 페이지를 로드할 수 없는 경우 압축 후에 데이터가 손상되었을 수 있습니다. 이 문제는 시스템에서 오래된 프록시 서버를 배포한 경우에 종종 발생할 수 있습니다. 이 문제를 해결하려면 `local.service.http.gzip.static` 및 `local.service.http.gzip.dynamic`을 `0`으로 설정하여 데이터 압축을 비활성화해 보십시오. 문제가 해결되면 프록시 서버를 업데이트할 수 있습니다.

20.14.4.3 와일드카드 패턴을 사용하는 명령이 작동하지 않음

UNIX 쉘의 경우 일부는 와일드카드 매개 변수에 따옴표가 필요하지만 일부는 그렇지 않습니다. 예를 들어, C 쉘은 와일드카드(*, ?)를 파일로 포함하는 인수 확장을 시도하며 일치하는 항목이 없으면 실패합니다. 이러한 패턴 일치 인수를 `mboxutil`과 같은 명령에 전달하려면 따옴표로 묶어야 할 수 있습니다.

예를 들면 다음과 같습니다.

```
mboxutil -l -p user/usr44*
```

Bourne 쉘에서 작동하지만 tsch 및 C 쉘에서는 실패합니다. 이러한 쉘에는 다음이 필요합니다.

```
mboxutil -l -p "user/usr44*"
```

와일드카드 패턴을 사용하는 명령이 작동하지 않을 경우 해당 쉘의 와일드카드를 따옴표로 묶어야 하는지 여부를 확인합니다.

20.14.4.4 알 수 없거나 잘못된 분할 영역

메일함을 방금 만든 새 분할 영역으로 이동했거나 Messaging Server를 갱신 또는 다시 시작하지 않은 경우 Messenger Express에서 "알 수 없는/잘못된 분할 영역"이라는 메시지가 표시될 수 있습니다. 이 문제는 새 분할 영역에서만 발생합니다. 이제 추가 사용자 메일함을 이 새 분할 영역에 추가할 경우 Messaging Server를 갱신 또는 다시 시작할 필요가 없습니다.

20.14.4.5 사용자 메일함 디렉토리 문제

사용자 저장소 손상이 몇몇의 사용자로 제한되고 시스템에 대한 전역 손상이 없을 경우 사용자 메일함 문제가 존재합니다. 다음 지침은 사용자 메일함 디렉토리 문제를 식별, 분석 및 해결하기 위한 프로세스를 제시합니다.

1. 로그 파일, 오류 메시지 또는 관찰된 모든 비정상적인 동작을 검토합니다.
2. 정보와 기록을 계속 디버깅하려면 전체 *store_root*`/mboxlist/` 사용자 디렉토리를 메시지 저장소 외부의 다른 위치로 복사합니다.

3. 문제를 일으키는 사용자 폴더를 찾으려면 `reconstruct -r -n` 명령을 실행합니다. `reconstruct`를 사용하여 폴더를 찾을 수 없는 경우 `folder.db`에 폴더가 존재하지 않을 수 있습니다.

   `reconstruct -r -n` 명령을 사용하여 폴더를 찾을 수 없는 경우 `hashdir` 명령을 사용하여 위치를 확인합니다. `hashdir`에 대한 자세한 내용은 **Sun Java System Messaging Server 6.3 Administration Reference**의 Messaging Server Command-line Utilities 장에 있는 608 페이지 “20.11.2.3 hashdir 유틸리티” 및 `hashdir` 유틸리티를 참조하십시오.
4. 폴더를 찾은 후에는 파일과 권한을 검사하고 적절한 파일 크기를 확인합니다.
5. `reconstruct -r`(`-n` 옵션 없이)을 사용하여 메일함을 다시 작성합니다.
6. 관찰된 문제를 `reconstruct`에서 감지하지 못하는 경우 `reconstruct -r -f` 명령을 실행하여 메일 폴더를 다시 구성할 수 있습니다.
7. 폴더가 `mboxlist` 디렉토리(*store_root*`/mboxlist`)에 존재하지 않지만 `partition` 디렉토리(*store_root*`/partition`)에 존재할 경우 전역 불일치가 존재할 수 있습니다. 이 경우 `reconstruct -m` 명령을 실행해야 합니다.
8. 이전 단계들로 문제가 해결되지 않을 경우 `store.idx` 파일을 제거하고 `reconstruct` 명령을 다시 실행할 수 있습니다.

주의 – `reconstruct` 명령으로 검색할 수 없는 파일에 문제가 있다고 확신하는 경우에만 `store.idx` 파일을 제거해야 합니다.

9. 문제가 특정 메시지로 한정된 경우 해당 메시지 파일을 메시지 저장소 외부의 다른 위치로 복사하고 `mailbox/` 디렉토리에서 `reconstruct -r` 명령을 실행해야 합니다.
10. 폴더가 디스크(*store_root*`/partition/` 디렉토리)에 존재하지만 데이터베이스(*store_root*`/mboxlist/` 디렉토리)에는 확실하게 없을 경우 `reconstruct -m` 명령을 실행하여 메시지 저장소 일관성을 확인합니다.

`reconstruct` 명령에 대한 자세한 내용은 633 페이지 “20.14.3 메일함 및 메일함 데이터베이스 복구”를 참조하십시오.

20.14.4.6 저장소 데몬이 시작되지 않음

`stored`가 시작되지 않고 다음 오류 메시지가 반환됩니다.

```
# msg-svr-base/sbin/start-msg

msg-svr-base: Starting STORE daemon ...Fatal error: Cannot
find group in name service
```

이 오류 메시지는 `local.servergid`에 구성된 UNIX 그룹을 찾을 수 없다는 것을 나타냅니다. `Stored` 및 다른 유틸리티에서는 해당 `gid`가 이 그룹으로 설정되어야 합니다. 경우에 따라 실수로 `local.servergid`에 의해 정의된 그룹이 삭제될 수 있습니다. 이런

경우에는 삭제된 그룹을 만들고 mailsrv를 그룹에 추가한 다음 *instance_root*와 해당 파일의 소유권을 mailsrv 및 해당 그룹으로 변경합니다.

20.14.4.7 메일함 오버플로 때문에 사용자 메일이 전달되지 않음

메시지 저장소의 store.idx 파일에는 2GB의 엄격한 제한이 적용됩니다. 이는 단일 메일함(폴더)에 약 100만 개의 메시지가 들어가는 정도에 해당됩니다. store.idx 파일이 2GB를 초과하려 할 정도로 메일함이 커지면 사용자는 새 전자 메일을 받지 못하게 됩니다. 또한 imapd, popd, mshttpd와 같이 해당 메일함을 처리하는 다른 프로세스의 성능도 저하될 수 있습니다.

이 문제가 발생하면 mail.log_current에 다음과 같은 오류가 표시됩니다.

```
05-Oct-2005 16:09:09.63 ims-ms Q 7 ...System I/O error.Administrator, check
server log for details.System I/O error.
```

그 외에도 MTA 로그 파일에는 다음과 같은 오류가 표시됩니다.

```
[05/Oct/2005:16:09:09 +0900] jmail ims_master[20745]:Store Error:Unable to
append cache for user/admin:File too large
```

사용자의 메시지 저장소 디렉토리에 있는 파일을 조회하거나 imta 로그 파일에서 자세한 메시지를 확인하면 이 문제가 발생한 것을 확실히 알 수 있습니다.

즉시 필요한 조치는 파일의 크기를 줄이는 것입니다. 일부 메일을 삭제하거나 다른 메일함으로 옮기십시오. mboxutil -r을 사용하여 폴더 이름을 변경할 수도 있고 mboxutil -d를 사용하여 폴더를 삭제할 수도 있습니다(606 페이지 "20.11.2.1 mboxutil 유틸리티" 참조).

장기적으로는 사용자에게 메일함 크기 제한에 대해 알리거나, 에이징 정책(592 페이지 "20.9 자동 메시지 제거(만료 및 제거) 기능 설정 방법" 참조)과 할당량 정책(583 페이지 "20.8 메시지 저장소 할당량 정보" 참조)을 구현하거나, local.store.maxmessages를 설정하여 메일함 제한을 설정하거나(**Sun Java System Messaging Server 6.3 Administration Reference**의 "configutil Parameters" 참조), 아카이브 시스템을 설정하거나, 메일함 크기를 제어 범위 내로 유지하기 위한 조치를 취해야 합니다.

20.15 메일함을 새 시스템으로 이동 또는 마이그레이션

한 Messaging Server 시스템의 기존 메일함을 다른 Messaging Server 시스템으로 이동해야 할 수 있습니다. 이러한 경우는 다음과 같습니다.

- Sun Messaging Server가 아닌 시스템에서 Sun Java System Messaging Server로 마이그레이션하는 경우
- 한 물리적 서버의 메일함을 다른 물리적 서버로 이동하는 경우

Messaging Server는 한 시스템의 메일함을 다른 시스템으로 이동하는 여러 가지 방법을 제공합니다. 방법마다 장단점이 있으며, 여기에 대해서는 아래 절에서 설명합니다. 이 방법에 대한 자세한 내용은 다음 절을 참조하십시오.

- 642 페이지 "20.15.1 온라인 상태에서 다른 Messaging Server로 사용자 메일함 마이그레이션"
- 644 페이지 "온라인 상태로 사용자 메일함을 다른 Messaging Server로 마이그레이션하는 방법"
- 648 페이지 "IMAP 클라이언트를 사용하여 메일함을 이동하는 방법"
- 650 페이지 "`moveuser` 명령을 사용하여 메일함을 이동하는 방법"
- 651 페이지 "`imsimport` 명령을 사용하여 메일함을 이동하는 방법"

20.15.1 온라인 상태에서 다른 Messaging Server로 사용자 메일함 마이그레이션

이 절차를 사용하여 메시지 저장소를 이전 버전 Messaging Server에서 최신 버전으로 마이그레이션하거나 메일함을 다른 Sun Messaging Server 메시지 저장소로 이동할 수 있습니다. 이 절차는 iPlanet Messaging Server 5.0 이상에서 수행해야 합니다. 이전 버전 Messaging Server 또는 Sun Microsystems가 아닌 메시지 저장소에서 메시지를 이동할 때는 이 절차를 사용할 수 없습니다.

이 절차로 메일함을 이동할 경우의 장점은 다음과 같습니다.

- 시스템 관리자가 사용자의 개입 없이 이전 원본 시스템에서 새로운 대상 시스템으로 메일함을 이동합니다.
- 이 프로세스는 다른 프로세스보다 더 빠릅니다.
- 전체 분할 영역을 이동하는 경우 재연결 작업이 필요하지 않습니다.
- 두 Messaging Server 시스템 모두 활성화되어 온라인 상태로 유지됩니다.
- 메시지 저장소에 있는 모든 메일함이나 해당 메시지의 하위 집합을 마이그레이션할 수 있습니다. 이 절차는 증분 마이그레이션을 가능하게 합니다.

이 절차로 메일함을 이동할 경우의 단점은 다음과 같습니다.

- 이 방법은 Sun Messaging Server가 아닌 시스템에서 작동하지 않습니다.
- 마이그레이션 중인 사용자가 메일함 마이그레이션을 완료할 때까지는 자신의 메일함에 대한 액세스 권한이 없습니다.
- 이 방법은 복잡하고 많은 시간이 소요될 수 있습니다.

20.15.1.1 증분 메일함 마이그레이션

증분 마이그레이션은 메시지 저장소를 다른 시스템으로 안전하고 효율적으로 이동하거나 새 시스템으로 업그레이드할 수 있는 등 다양한 이점을 제공합니다. 증분 마이그레이션을 사용하면 이전 백엔드 메시지 저장소와 함께 새 백엔드 메시지 저장소

시스템을 구성할 수 있습니다. 그런 다음 새 시스템을 테스트하고, 친분이 있는 사용자에게 마이그레이션한 다음 새 시스템을 다시 테스트할 수 있습니다. 새 시스템과 구성이 편리하고 마이그레이션 절차에 익숙한 경우 실제 상업용 사용자를 마이그레이션할 수 있습니다. 이러한 사용자를 개별 백업 그룹으로 분할하여 마이그레이션 중에 이 그룹의 구성원만 잠시 오프라인 상태로 전환할 수 있습니다.

온라인 증분 마이그레이션은 업그레이드가 실패할 경우 시스템 전체에서 백아웃을 계획할 필요가 없다는 또 다른 장점이 있습니다. 백아웃은 시스템에 대해 수행한 변경을 취소하여 원래의 작업 상태로 되돌리는 절차입니다. 마이그레이션을 수행할 때 실패에 대한 계획을 수립해야 합니다. 즉, 마이그레이션의 모든 단계에서 이전 작업 상태로 되돌릴 계획을 세워야 합니다.

오프라인 마이그레이션은 모든 마이그레이션 단계를 완료하고 서비스를 다시 실행할 때까지는 마이그레이션이 성공했는지 확인할 수 없다는 문제가 있습니다. 따라서 시스템이 작동하지 않고 빠르게 수정할 수 없는 경우 수행한 모든 단계에 대해 백아웃 절차를 수행해야 합니다. 이 작업은 매우 힘들고 시간이 많이 소요될 수 있으며, 작업을 수행하는 동안 사용자는 오프라인 상태로 유지됩니다.

온라인 증분 마이그레이션에서는 다음과 같은 기본 단계를 수행합니다.

1. 이전 시스템과 함께 새 시스템을 구축하여 두 시스템이 독립적으로 작동할 수 있게 합니다.

2. 이전 시스템이 새 시스템과 공존하도록 구성합니다.

3. 친분이 있는 사용자 그룹을 마이그레이션하고 새 시스템을 테스트하며 이전 시스템과의 공존 상태를 테스트합니다.

4. 이전 시스템의 사용자를 여러 그룹으로 분할하고 원하는 경우 각 그룹을 새 그룹으로 마이그레이션합니다.

5. 이전 시스템을 역어셈블합니다.

두 시스템이 공존하기 때문에 새 시스템으로 마이그레이션하기 전에 새 시스템을 테스트하고 익숙해질 시간이 있습니다. 원하지 않더라도 백아웃 절차를 수행해야 하는 경우에는 2단계와 4단계에 대해서만 계획해야 합니다. 2단계는 사용자의 데이터를 건드리지 않기 때문에 쉽게 반전됩니다. 4단계에서는 백아웃을 수행하여 사용자의 상태를 활성 상태로 되돌리고 메일 호스트 속성을 이전 호스트로 되돌립니다. 시스템 차원의 백아웃이 필요하지 않습니다.

20.15.1.2 온라인 마이그레이션 개요

온라인 상태에서 메일함을 마이그레이션하는 프로세스는 매우 간단합니다. 메일함으로 전송 중인 메시지(MTA 채널 대기열에서 전달을 위해 대기 중)가 마이그레이션 중에 손상되지 않도록 하는 과정은 매우 복잡합니다. 한 가지 해결 방법으로 마이그레이션 과정에서 전송된 메시지를 **보관** 상태로 유지하고 다양한 채널 대기열의 메시지가

전달될 때까지 대기합니다. 그러나 시스템 문제나 특정 사용자의 할당량 초과로 메시지가 대기열에 고착될 수 있습니다. 그럴 경우 문제를 해결한 다음 메일함을 마이그레이션해야 합니다.

다양한 방법으로 메시지 손실 가능성을 줄이고 메시지가 채널 대기열에 고착되지 않도록 할 수 있지만, 그럴 경우 절차가 복잡해지는 단점이 있습니다.

절차에서 수행되는 단계의 순서와 필요성은 모든 메일함에 전달되는 모든 메시지가 손실되는지 여부와 배포에 따라 다릅니다. 이 절에서는 단계와 관련한 이론과 개념에 대해 설명합니다. 사용자는 각 단계를 이해하고 지정된 배포에 대해 수행할 각 단계와 순서를 결정해야 합니다. 다음은 메일함 이동 프로세스의 개요입니다. 이 프로세스는 배포에 따라 다를 수 있습니다.

1. 이동 중인 메일함에 대한 사용자 액세스를 차단합니다.
2. 이동 중인 메일함으로 전달되는 메시지를 임시로 보관합니다.
3. 메시지가 채널 대기열에 고착되지 않는지 확인합니다.
4. 사용자의 메일 호스트 속성을 새 메일함 위치로 변경합니다.
5. 메일함을 새 위치로 이동합니다.
6. 전달을 위해 보관 중인 메시지를 새 메일함으로 릴리스하고 마이그레이션된 메일함으로 전달할 받는 메시지를 활성화합니다.
7. 이전 메시지 저장소를 검사하여 마이그레이션 이후에 전달된 메시지가 있는지 확인합니다.
8. 메일함에 대한 사용자 액세스 차단을 해제합니다.

▼ 온라인 상태로 사용자 메일함을 다른 Messaging Server로 마이그레이션하는 방법

시작하기 전에 이 마이그레이션 유형에 대한 요구 사항은 다음과 같습니다.

- 원본(이전) Messaging Server와 대상(새) Messaging Server 모두에서 `stored`를 실행해야 합니다.
- 원본 시스템과 대상 시스템이 공존할 경우 두 시스템 간에 메시지를 라우팅할 수 있어야 합니다. 이렇게 해야 전달 상태 알림 메시지를 대상 시스템에서 생성하여 원본 시스템으로 전달하는 등의 작업을 수행할 수 있습니다.

주 – 일부 단계는 Messaging Server를 최신 버전으로 업그레이드하는 경우에만 적용됩니다. 이 단계는 다른 메시지 저장소로 메일함 마이그레이션만 수행하는 경우에는 적용되지 않을 수도 있습니다. 전체 시스템 마이그레이션에 적용되는 단계는 별도의 설명이 있습니다.

---

1 **원본 시스템에서 backup-groups.conf 파일을 사용하여 이동할 사용자 항목을 동일한 백업 그룹 수로 분할합니다.**

이 단계는 해당 절차의 뒤에 나오는 메일함 마이그레이션 단계 8에 대한 준비 과정입니다. 자세한 내용은 615 페이지 "20.12.2 백업 그룹 만들기"을 참조하십시오.

파일에 사용자 아이디를 넣고 imsbackup 명령에 -u 옵션을 사용할 수도 있습니다.

2 **이동 대상인 사용자에게 이동이 완료될 때까지 메일함에 액세스할 수 없음을 알립니다.**

데이터를 이동하기 전에 이동할 사용자가 메일 시스템에서 로그아웃했는지 확인합니다. (626 페이지 "20.13 사용자 액세스 모니터링" 참조)

3 **백엔드 메시지 저장소와 MMP 시스템에서 인증 캐시 시간 초과를 0으로 설정하고 MTA에서 ALIAS_ENTRY_CACHE_TIMEOUT 옵션을 0으로 설정합니다.**

a. **이동할 메일함이 있는 백엔드 메시지 저장소에서 인증 캐시 시간 초과를 0으로 설정합니다.**

```
configutil -o service.authcachettl -v 0
```

이 단계와 단계 7(mailUserStatus를 hold로 변경)에서는 마이그레이션 중에 사용자의 메일함 액세스를 즉시 차단합니다.

b. **모든 MMP에서 LDAP 및 인증 캐시 시간 초과를 0으로 설정합니다.**

ImapProxyAService.cfg 및 PopProxyAService.cfg는 LdapCacheTTL과 AuthCacheTTL을 모두 0으로 설정합니다.

c. **마이그레이션할 메일함에 메시지를 삽입하는 MTA를 호스트하는 모든 Messaging Server에서 ALIAS_ENTRY_CACHE_TIMEOUT 옵션을 0으로 설정합니다.**

마이그레이션 중인 메일함에 메시지를 삽입하는 MTA를 호스트하는 Messaging Server는 일반적으로 백엔드 메시지 저장소입니다. 그러나 시스템에서 LMTP를 사용하는 경우에는 해당 시스템이 인바운드 MTA입니다. 구성을 확인합니다.

/msg_svr_base /config/option.dat에서 ALIAS_ENTRY_CACHE_TIMEOUT을 재설정하면 MTA가 캐시를 우회하고 LDAP 항목을 직접 조사하므로 중간 채널 대기열(예: conversion 또는 reprocess 채널)에 만료된 캐시 정보가 아니라 이동 중인 사용자의 새 mailUserStatus(hold)가 표시됩니다. ALIAS_ENTRY_CACHE_TIMEOUT은 option.dat에 있습니다.

d. **캐시가 재설정된 시스템을 다시 시작합니다.**

변경 사항을 적용하려면 시스템을 다시 시작해야 합니다. 자세한 내용은 104 페이지 "4.4 서비스 시작 및 중지"를 참조하십시오.

4 **원본 Messaging Server와 대상 Messaging Server가 모두 실행 중인지 확인합니다.**

원본 Messaging Server는 받는 메시지를 새 대상 서버에 라우팅할 수 있어야 합니다.

5 메일함을 이동할 모든 사용자 항목에서 LDAP 속성 mailUserStatus를 active에서 hold로 변경합니다.

속성을 변경하면 받는 메시지가 hold 대기열에 보관되고 IMAP, POP 및 HTTP를 통한 액세스가 금지됩니다. 일반적으로 사용자는 사용자 그룹으로 이동됩니다. 단일 도메인의 모든 메일함을 이동할 경우 mailDomainStatus 속성을 사용할 수 있습니다.

mailUserStatus에 대한 자세한 내용은 **Sun Java Communications Suite 5 Schema Reference**의 "mailUserStatus"를 참조하십시오.

6 마이그레이션 중인 메일함에 전달된 메시지가 ims-ms 또는 tcp_lmtp* 채널 대기열(LMTP가 배포된 경우)에 고착되지 않는지 확인합니다.

다음 명령을 사용하여 메시지가 채널 대기열 디렉토리 트리에 있고 마이그레이션할 사용자에 대해 *held* 상태인지(.HELD 파일을 보려면) 확인합니다.

```
imsimta qm directory -to=<user_address_to_be_migrated> -directory_tree

imsimta qm directory -to=<user_address_to_be_migrated> -held -directory_tree
```

대기열에 메시지가 있는 경우 나중에 위 명령을 실행하여 MTA가 해당 메시지를 대기열에서 해제하는지 확인합니다. 대기열에서 해제되지 않은 메시지가 있는 경우 마이그레이션을 수행하기 전에 이 문제를 해결해야 합니다. 이 문제는 드물기는 하지만 수신자 메일함의 할당량이 초과된 경우, 사용자가 로그인한 후 메시지를 이동 중이라서 메일함이 차단된 경우, LMTP 백엔드 서버가 응답하지 않는 경우, 네트워크 또는 이름 서버 문제 등이 원인일 수 있습니다.

7 이동할 사용자 항목과 메일 그룹 항목*에서 LDAP 속성 mailHost를 변경합니다.

ldapmodify 명령을 사용하여 항목을 새 메일 서버로 변경합니다. Messaging Server 또는 Directory Server와 함께 제공된 ldapmodify를 사용합니다. Solaris OS ldapmodify 명령은 사용하지 마십시오.

* 이전 메일 호스트가 종료된 경우에는 메일 그룹 항목에서 mailHost 속성을 변경해야 합니다. 이 속성을 새 메일 호스트로 변경하거나 속성을 완전히 제거할 수 있습니다. 메일 그룹은 선택적으로 mailHost를 가질 수 있습니다. mailHost를 갖는다는 것은 해당 호스트만 그룹 확장을 수행할 수 있다는 의미이고, mailHost를 생략한다는 것은(보다 일반적인 경우) 모든 MTA가 그룹 확장을 수행할 수 있다는 의미입니다. 메일 그룹 항목에는 마이그레이션할 메일함이 없고 일반적으로 mailhost 속성도 없습니다.

mailhost에 대한 자세한 내용은 **Sun Java Communications Suite 5 Schema Reference**의 "mailHost"를 참조하십시오.

8 메일함 데이터를 원본 Messaging Server 메시지 저장소에서 대상 Messaging Server 메시지 저장소로 이동하고 시작된 시간을 기록합니다.

imsbackup 유틸리티를 사용하여 메일함을 백업하고 imsrestore 유틸리티를 사용하여 새 Messaging Server에 복원합니다. 예를 들어, oldmail.siroe.com이라는 Messaging Server 5.2

시스템의 메일함을 newmail.siroe.com으로 마이그레이션하려면 oldmail.siroe.com에서 다음 명령을 실행합니다.

```
/server-root/bin/msg/store/bin/imsbackup -f- /instance/group      \
| rsh newmail.siroe.com /opt/SUNWmsgsr/lib/msg/imsrestore.sh   \
-f- -c y -v 1
```

여러 백업 및 복원 세션(그룹마다 하나씩)을 동시에 실행하여 새 메시지 저장소로의 전송 속도를 최대화할 수 있습니다. imsbackup 및 imsrestore 유틸리티에 대한 자세한 내용은 **Sun Java System Messaging Server 6.3 Administration Reference**의 "Command Descriptions"과 614 페이지 "20.12 메시지 저장소 백업 및 복원"을 참조하십시오.

주 – 나중의 전달 확인을 위해 imsbackup이 실행된 시간의 타임스탬프를 기록합니다.

---

9 (시스템 업그레이드를 위한 조건부 단계) 메일함 마이그레이션을 이전 버전 Messaging Server에서 현재 버전으로 업그레이드하는 과정의 일부로 수행하는 경우 현재 버전 Messaging Server를 시스템에 대한 새 기본 Messaging Server로 설정합니다.

oldmail.siroe.com의 DNS A 레코드를 변경하여 newmail.siroe.com(이전에 oldmail.siroe.com에서 호스트되던 도메인을 담당하는 서버)을 가리키도록 합니다.

10 새 메시지 저장소에 대한 사용자 액세스를 활성화합니다.

LDAP 속성 mailUserStatus 또는 mailDomainStatus(해당하는 경우)를 hold 상태로 변경되기 이전의 값(예: active)으로 설정합니다.

11 모든 원본 Messaging Server에서 메시지를 *held* 상태에서 해제합니다.

받는 메시지를 보관 중인 시스템은 다음 명령을 실행하여 모든 사용자 메시지를 릴리스해야 합니다.

```
imsimta qm release -channel=hold -scope
```

여기서 *scope*는 all(모든 메시지 릴리스), user(사용자 아이디) 또는 domain(사용자가 있는 도메인)입니다.

12 인증 캐시 시간 초과 및 ALIAS_ENTRY_CACHE_TIMEOUT 옵션을 기본값이나 원하는 값으로 재설정하고 시스템을 다시 시작합니다.

이제 마이그레이션해야 할 모든 사용자 메일함을 마이그레이션했습니다. 계속하기 전에 이전 시스템에서 LDAP에 새 항목이 mailhost로 생성되지 않았는지 확인하고, 생성된 항목이 있는 경우 해당 항목을 마이그레이션합니다. 또한 준비 시스템을 수정하여 해당 항목이 만들어지는지 확인합니다.

preferredmailhost 속성을 새 메일 호스트의 이름으로 변경할 수도 있습니다.

백엔드 메시지 저장소에 대해 인증 캐시 시간 초과를 다음과 같이 설정합니다.

```
configutil -o service.authcachettl -v 900
```

MMP의 경우 ImapProxyAService.cfg와 PopProxyAService.cfg에서 LdapCacheTTL과 AuthCacheTTL 옵션을 900으로 설정합니다.

MTA의 경우 ALIAS_ENTRY_CACHE_TIMEOUT 옵션을 600으로 설정합니다. ALIAS_ENTRY_CACHE_TIMEOUT은 option.dat에 있습니다.

변경 사항을 적용하려면 시스템을 다시 시작해야 합니다. 자세한 내용은 104 페이지 "4.4 서비스 시작 및 중지"를 참조하십시오.

13 사용자 클라이언트가 새 메일 서버를 가리키는지 확인합니다.

업그레이드가 끝나면 사용자가 메일 클라이언트 프로그램을 통해 새 서버를 가리키도록 합니다. 이 예에서는 oldmail.siroe.com에서 newmail.siroe.com을 가리킵니다.

대안은 MMP(Messaging Multiplexor)를 사용하는 것입니다. 그러면 사용자가 새 메일 서버를 클라이언트로 직접 가리킬 필요가 없습니다. MMP는 LDAP 사용자 항목에 저장된 mailHost 속성으로부터 정보를 가져와서 클라이언트를 새 서버로 자동으로 리디렉션합니다.

14 모든 작업이 완료되면 마이그레이션 이후에 이전 메시지 저장소로 전달된 메시지가 없는지 확인합니다.

이전 메시지 저장소로 이동한 후 mboxutil -l을 실행하여 메일함을 나열합니다. 마지막 메시지 전달 타임스탬프를 확인합니다. 마이그레이션 타임스탬프(imsbackup 명령을 실행한 날짜 스탬프) 이후에 전달된 메시지가 있는 경우 백업 및 복원 명령을 사용하여 해당 메시지를 마이그레이션합니다. 준비 단계를 수행했기 때문에 마이그레이션 이후에 전달된 메시지가 표시되는 경우는 극히 드뭅니다.

이론적으로는 메시지가 notices 채널 키워드에 지정된 날짜 또는 시간 동안 대기열에 고착될 수 있습니다(253 페이지 "10.10.4.3 알림 메일 전달 간격 설정" 참조).

15 새 메시지 저장소에서 중복 메시지를 제거하고 relinker 명령을 실행합니다.

이 명령은 새 메시지 저장소에서 디스크 공간을 비울 수 있습니다. 610 페이지 "20.11.7 동일한 메시지의 중복 저장에 따른 저장소 크기 줄이기"를 참조하십시오.

16 마이그레이션한 원본 저장소에서 이전 메시지를 제거하고 이전 저장소의 데이터베이스에서 사용자를 삭제합니다.

mboxutil -d 명령을 실행합니다. (606 페이지 "20.11.2.1 mboxutil 유틸리티" 참조)

▼ IMAP 클라이언트를 사용하여 메일함을 이동하는 방법

메시지를 다른 Messaging Server로 마이그레이션해야 하는 경우 언제든지 이 절차를 사용할 수 있습니다. 이 방법으로 메일함을 이동하기 전에 장점과 단점을 고려하십시오.

IMAP 클라이언트를 사용하여 메일함을 이동할 경우의 장점은 다음과 같습니다.

- 이 방법을 사용하여 Sun Messaging Server가 아닌 시스템에서 Sun Java System Messaging Server로 마이그레이션할 수 있습니다. 메일함을 다른 물리적 서버로 이동하는 데 이 방법을 사용할 수도 있습니다.
- 시스템 관리자가 새 메일 서버나 메시지 저장소를 설정한 후 사용자가 메일함을 새 시스템으로 이동해야 합니다.
- 메일함 이동 프로세스가 비교적 간단합니다.
- 사용자의 메일함 액세스를 비활성화할 필요는 없습니다.

IMAP 클라이언트를 사용하여 메일함을 이동할 경우의 단점은 다음과 같습니다.

- 이전 시스템과 새 시스템이 동시에 실행되고 있어야 하며 사용자가 액세스할 수 있어야 합니다.
- 총체적으로 이 방법은 다른 방법보다 메일함을 이동하는 데 오랜 시간이 걸립니다.
- 메일함을 새 시스템으로 이동할 책임이 사용자에게 있습니다.
- 재연결 작업을 수행할 때까지 새 메시지 저장소의 크기가 이전 메시지 저장소보다 훨씬 큽니다.

1 **새 Messaging Server를 설치하고 구성합니다.**

2 **`local.store.relinker`를 enable로 설정합니다.**

이렇게 하면 동일한 메시지의 중복 저장으로 인한 새 시스템의 메시지 저장소 크기가 줄어듭니다. 자세한 내용은 610 페이지 “20.11.7 동일한 메시지의 중복 저장에 따른 저장소 크기 줄이기”를 참조하십시오.

3 **새 Messaging Server에서 사용자를 준비합니다.**

Delegated Administrator를 사용하여 이 작업을 수행할 수 있습니다. 새 시스템에서 사용자가 준비되면 새로 도착하는 메시지가 새 INBOX로 전달됩니다.

4 **사용자가 메일 클라이언트에서 새 Messaging Server 메일함과 이전 메일함을 모두 표시하도록 구성하게 합니다.**

여기에는 클라이언트에서 새 전자 메일 계정을 설정하는 작업이 포함될 수 있습니다. 자세한 내용은 메일 클라이언트 설명서를 참조하십시오.

5 **사용자에게 이전 Messaging Server의 폴더를 새 Messaging Server로 끌어다 놓도록 지시합니다.**

6 **모든 메일함이 새 시스템으로 마이그레이션되었는지 사용자에게 확인한 다음 이전 시스템에서 사용자 계정을 종료합니다.**

▼ moveuser 명령을 사용하여 메일함을 이동하는 방법

메시지를 다른 Messaging Server로 마이그레이션해야 하는 경우 언제든지 이 절차를 사용할 수 있습니다. 이 절차는 Sun Messaging Server가 아닌 시스템의 IMAP 메일함을 Sun Java System Messaging Server로 마이그레이션하는 데 유용합니다. 이 방법으로 메일함을 이동하기 전에 장점과 단점을 고려하십시오.

moveuser 명령을 사용하여 메일함을 이동할 경우의 장점은 다음과 같습니다.

- 이전 시스템의 메일함을 새 시스템으로 이동할 책임이 전적으로 시스템 관리자에게 있습니다. 사용자는 어떤 작업도 수행할 필요가 없습니다.
- 모든 IMAP 서버에서 작동합니다.

moveuser 명령을 사용하여 메일함을 이동할 경우의 단점은 다음과 같습니다.

- 이전 시스템과 새 시스템이 동시에 실행되고 있어야 하며 사용자가 액세스할 수 있어야 합니다.
- 이 방법은 IMAP가 아닌 다른 방법보다 메일함을 이동하는 데 오랜 시간이 걸립니다.
- 메일함을 이동하는 동안 사용자의 메일함 액세스를 비활성화해야 합니다.
- 재연결 작업을 수행할 때까지 새 메시지 저장소의 크기가 이전 메시지 저장소보다 훨씬 큽니다.

1 **새 Messaging Server를 설치하고 구성합니다.**

2 **local.store.relinker를 enable로 설정합니다.**

이렇게 하면 동일한 메시지의 중복 저장으로 인한 새 시스템의 메시지 저장소 크기가 줄어듭니다. 자세한 내용은 610 페이지 “20.11.7 동일한 메시지의 중복 저장에 따른 저장소 크기 줄이기”를 참조하십시오.

3 **Messaging Server로 받는 메시지를 중지합니다.**

사용자 속성 mailUserStatus를 hold로 설정합니다.

4 **필요한 경우 새 Messaging Server에서 사용자를 준비합니다.**

이전 버전의 Messaging Server에서 마이그레이션하는 경우 동일한 LDAP 디렉토리와 서버를 사용할 수 있습니다. moveuser는 각 사용자 항목에서 mailhost 속성을 변경합니다.

5 **moveuser 명령을 실행합니다.**

Directory Server siroe.com의 계정 정보를 기반으로, host1의 모든 사용자를 host2로 이동합니다.

```
MoveUser -l \
"ldap://siroe.com:389/o=siroe.com???(mailhost=host1.domain.com)" \
-D "cn=Directory Manager" -w password -s host1 -x admin \
-p password -d host2 -a admin -v password
```

`moveuser` 명령에 대한 자세한 내용은 **Sun Java System Messaging Server 6.3 Administration Reference**의 "MoveUser"를 참조하십시오.

6 **새 메시지 저장소에 대한 사용자 액세스를 활성화합니다.**

`mailUserStatus` LDAP 속성을 `active`로 설정합니다.

7 **이전 시스템을 종료합니다.**

▼ `imsimport` 명령을 사용하여 메일함을 이동하는 방법

이 절차는 특히 UNIX `/var/mail` 형식 폴더의 메일함을 Sun Java System Messaging Server 메시지 저장소로 이동하는 데 사용됩니다. 그러나 마이그레이션하는 Messaging Server에서 IMAP 메시지 저장소를 UNIX `/var/mail` 형식으로 변환할 수 있으면 `imsimport` 명령을 사용하여 메시지를 Sun Java System Messaging Server로 마이그레이션할 수 있습니다. 이 방법으로 메일함을 이동하기 전에 장점과 단점을 고려하십시오.

`imsimport` 명령을 사용하여 메일함을 이동할 경우의 장점은 다음과 같습니다.

- 이전 시스템의 메일함을 새 시스템으로 이동할 책임이 전적으로 시스템 관리자에게 있습니다. 사용자는 어떤 작업도 수행할 필요가 없습니다.

`imsimport` 명령을 사용하여 메일함을 이동할 경우의 단점은 다음과 같습니다.

- 이 방법은 IMAP가 아닌 다른 방법보다 메일함을 이동하는 데 오랜 시간이 걸립니다.
- 메일함을 이동하는 동안 사용자의 메일함 액세스를 비활성화해야 합니다.
- 재연결 작업을 수행할 때까지 새 메시지 저장소의 크기가 이전 메시지 저장소보다 훨씬 큽니다.

1 **새 Messaging Server를 설치하고 구성합니다.**

2 **`local.store.relinker`를 enable로 설정합니다.**

이렇게 하면 동일한 메시지의 중복 저장으로 인한 새 시스템의 메시지 저장소 크기가 줄어듭니다. 자세한 내용은 610 페이지 "20.11.7 동일한 메시지의 중복 저장에 따른 저장소 크기 줄이기"를 참조하십시오.

3 **필요한 경우 새 Messaging Server에서 사용자를 준비합니다.**

Delegated Administrator를 사용하여 이 작업을 수행할 수 있습니다. 아직 새 시스템으로 전환하지 마십시오.

4 **새 메시지 저장소와 이전 메시지 저장소에 대한 사용자 액세스를 모두 비활성화합니다.**

`mailUserStatus` LDAP 속성을 `hold`로 설정합니다. 사용자 메일이 보관 대기열로 전송되고 IMAP, POP 및 HTTP를 통한 메일함 액세스가 차단됩니다. 저장소 서버의 MTA 및 메시지 액세스 서버는 이 요구 사항을 따라야 합니다. 이 설정은 다른 모든 `mailDeliveryOption` 설정을 대체합니다.

5 **기존 메일 서버의 메시지 저장소가** `/var/mail` **형식이 아니면 메시지 저장소를** `/var/mail` **파일로 변환합니다.**

타사 메일 서버 설명서를 참조하십시오.

6 `imsimport` **명령을 실행합니다.**

예를 들면 다음과 같습니다.

```
imsimport -s /var/mail/joe -d INBOX -u joe
```

`imsimport` 명령에 대한 자세한 내용은 **Sun Java System Messaging Server 6.3 Administration Reference**의 "imsimport"를 참조하십시오.

7 **메시지 저장소에 대한 사용자 액세스를 활성화합니다.**

`mailUserStatus` LDAP 속성을 `active`로 설정합니다.

8 **새 메시지 저장소에 대한 사용자 액세스를 활성화합니다.**

9 **이전 시스템을 종료합니다.**

◆ ◆ ◆ 21장

21 메시지 아카이브

이 장에서는 Messaging Server의 아카이브 개념에 대해 설명합니다. 아카이브 시스템을 설정하는 방법에 대한 지침은 제공하지 않습니다. 배포에 대한 자세한 내용은 **Message Archiving Using the Sun Compliance and Content Management Solution**을 참조하십시오. 배포 정보를 별도의 문서에 두는 이유는 업데이트를 좀더 신속하게 구성하기 위해서입니다.

이 장은 다음 내용으로 구성되어 있습니다.

21.1 아카이브 개요

메시지 아카이브 시스템은 Messaging Server와 다른 별도의 시스템에 받고 보내는 메시지를 전부 또는 지정된 것만 저장합니다. 보내고, 받고, 삭제하고, 이동한 모든 메시지를 아카이브 시스템에 저장하고 보관할 수 있습니다. 아카이브 메시지는 전자 메일 사용자가 수정하거나 제거할 수 없기 때문에 받는 메시지와 보내는 메시지의 무결성이 유지됩니다. 메시지 아카이브는 레코드 유지에 유용하며, 메시지 저장소 관리에도 유용합니다. 예를 들어, 아카이브를 사용하여 메시지 백업을 수행하거나 오래된 메시지를 비싼 메시지 저장소에서 덜 비싼 아카이브 저장소로 옮기는 경우가 있을 수 있습니다.

아카이브 메시지는 별도의 아카이브 소프트웨어 GUI 클라이언트나 Messaging Server를 통해 액세스할 수 있습니다. 아카이브 메시지는 삭제되지 않기 때문에, Messaging Server에서 메시지가 삭제된 경우 아카이브 클라이언트를 사용하여 삭제된 메시지를 검색할 수 있습니다. 하지만 아카이브 메시지는 Messaging Server의 경우와 같은 방식으로 메일함 폴더에 저장되지 않습니다.

Messaging Server에서 아카이브 메시지에 액세스할 수 있도록 시스템을 설정할 수도 있습니다. 예를 들어, 2년이 넘은 메시지를 아카이브로 보관하도록 시스템을 설정할 수 있습니다. 그러면 메시지 본문이 메시지 저장소 대신 아카이브 시스템에 있게 됩니다. 사용자가 보기에는 메시지가 일반 전자 메일 메시지와 다르게 보이지 않습니다. 같은

헤더와 제목 정보가 나타나지만(이 부분은 메시지 저장소에 저장), 메시지 본문은 메시지 저장소에서 필요할 때 아카이브 서버로부터 다운로드됩니다. 따라서 아카이브 서버에서 메시지를 다운로드하는 동안 약간의 지연이 생길 수 있습니다. 또, 아카이브 메시지는 전자 메일 클라이언트에서 검색할 수 없습니다. 검색은 아카이브 GUI에서 수행해야 합니다.

21.1.1 메시지 아카이브 시스템: 컴플라이언스 및 운영

아카이브에는 컴플라이언스 및 운영의 두 가지 유형이 있습니다. 호환 아카이브는 검색 가능한 전자 메일 레코드를 유지할 법적인 의무가 있는 경우에 사용됩니다. MTA로 들어오는 선택된 전자 메일(사용자, 도메인, 채널, 받는 메일, 보내는 메일 등으로 선택)이 메시지 저장소나 인터넷에 전달되기 전에 아카이브 시스템에 복사됩니다. 스팸 및 바이러스 필터링을 수행하기 전이나 수행한 후에 아카이브를 실행하도록 설정할 수 있습니다.

작업 아카이브는 메일 관리용으로 사용됩니다. 예를 들면 다음과 같습니다.

- 덜 사용하는(오래된) 메시지를 저장 비용이 낮은 저장소를 사용하는 아카이브 시스템으로 옮겨 Messaging Server 메시지 저장소의 저장소 사용량 절감
- 데이터 백업 대체

컴플라이언스 및 운영 아카이브는 상호 배타적인 관계가 아닙니다. 즉, 컴플라이언스 및 운영 아카이브를 모두 사용하도록 시스템을 설정할 수 있습니다.

◆ ◆ ◆ 22장

22 JMQ 알림 플러그 인을 구성하여 Message Queue에서 사용할 메시지 생성

이 장에서는 JMQ 알림 플러그인을 구성하여 Message Queue 서비스에서 클라이언트가 소비할 메시지를 생성하는 방법에 대해 설명합니다.

이 장은 다음 내용으로 구성되어 있습니다.22 장

22.1 JMQ 알림 개요

Messaging Server 알림 플러그 인을 사용하면 메시징 서비스 또는 이벤트 서비스에 알림 메시지를 전달할 수 있습니다. 메시징 서비스에서 소비자(클라이언트 인터페이스)에게 알림을 전송하면 이 소비자는 메시지를 필터링하여 지정된 사용자에게 전달합니다.

예를 들어, 사용자의 메일함에 새 전자 메일이 도착하면 알림 플러그 인은 메시징 서비스에 알림 메시지를 전달합니다. 메시징 서비스의 구성 요소인 메시지 소비자는 알림을 받은 후 사용자의 전자 메일 클라이언트(Communications Express 또는 Mozilla 메일 등)로 보냅니다. 그러면 전자 메일 클라이언트가 사용자의 컴퓨터 화면에 팝업 메시지 "새 메시지가 수신되었습니다."를 표시할 수 있습니다.

또 다른 예로, 사용자의 메일함이 할당량을 초과한 경우 알림 플러그 인에서는 할당량 초과 알림 메시지를 생성합니다. 메시지 소비자는 사용자와 이벤트에 대해 알아야 할 관리자에게 경고를 보냅니다.

22.1.1 두 개의 알림 메시징 서비스

서로 다른 두 개의 메시징 서비스에 알림을 전달하도록 Messaging Server를 구성할 수 있습니다.

- Sun Java System Message Queue 3.6 2005Q4

- Event Notification Service

Message Queue 서비스는 JMS(Java Messaging Service) 사양을 구현하며 메시지 브로커, 메시지를 생성하거나 소비하는 클라이언트를 만드는 인터페이스, 관리 서비스 및 제어를 제공합니다. Message Queue는 라우팅 및 전달 기능, 프로토콜, 메시지 형식에 JMS 표준을 사용합니다.

Event Notification Service는 Messaging Server 및 Sun Java System Calendar Server와 함께 번들로 제공되는 구성 요소입니다. 이 서비스는 게시/가입 구조를 사용하여 이벤트 알림을 보내고 받는 전용 서비스입니다.

Message Queue, Event Notification Service 또는 두 서비스 모두에 대해 알림 생성자를 구성할 수 있습니다.

주 – 이 장에서는 Message Queue에 대해서만 알림을 구성하는 방법에 대해 설명합니다.

Event Notification Service에 대한 자세한 내용은 *Sun Java System Communications Suite Event Notification Service Guide*를 참조하십시오.

22.1.2 알림 플러그 인

Messaging Server에서 Message Queue 또는 Event Notification Service로 보내는 알림을 생성하려면 해당 서비스의 플러그 인을 구성해야 합니다.

- JMQ 알림 플러그 인을 사용하면 Message Queue 브로커에 알림 메시지를 전달할 수 있습니다.
- iBiff 플러그 인을 사용하면 Event Notification Service에 알림 이벤트를 게시할 수 있습니다.

iBiff 플러그 인을 로드하고 Event Notification Service를 구성하는 방법을 보려면 *Sun Java System Messaging Server* **관리 설명서**의 "부록 B: Messaging Server에서 Event Notification Service 관리"를 참조하십시오.

22.1.3 JMQ 알림 사용의 장점

JMQ 알림 플러그 인을 Message Queue와 함께 사용하면 다음과 같은 장점을 얻을 수 있습니다.

- Message Queue는 JMS 표준을 구현합니다.
- Message Queue를 사용하면 **항목**이나 **대기열** 또는 두 가지 전달 방법 모두에 메시지를 생성할 수 있습니다. 간략한 정의는 657 페이지 “22.1.3.1 항목 또는 대기열에 게시”를 참조하십시오.

- Message Queue는 메시지 배포를 수행하는 동안, 특히 메시지를 대기열에 생성할 때 향상된 로드 균형 조정 기능을 제공합니다.
- JMQ 알림 플러그 인을 사용하면 알림 플러그 인을 다섯 개까지 구성할 수 있습니다. 서로 다른 여러 플러그 인에서 항목, 대기열, Event Notification Service 등에 메시지를 생성할 수 있습니다. 자세한 내용은 657 페이지 "22.1.3.2 여러 개의 JMQ 알림 플러그 인 사용"을 참조하십시오.
- Message Queue는 안정적인 알림 전달 기능을 제공합니다.

  예를 들어 영구 플래그를 사용하는 메시지를 생성하도록 JMQ 알림 플러그 인을 구성하면 소비자가 메시지를 받을 때까지 메시지가 Message Queue 브로커에 남습니다. 서버가 다운되는 경우에도 메시지를 검색하여 적절한 소비자가 사용할 수 있도록 메시지가 저장됩니다.

22.1.3.1 항목 또는 대기열에 게시

항목과 대기열은 서로 다른 메시징 전달 패턴을 사용하며 각각 Message Queue 서비스에 구성할 수 있습니다.

항목.메시지 생성자가 항목으로 메시지를 보낼 때 게시/가입 구조가 사용됩니다. 이 브로드캐스트 패턴에서 생성자는 항목 대상으로 메시지를 보냅니다. 이 항목 대상에 가입할 수 있는 소비자의 수에는 제한이 없습니다. 항목에 가입된 각 소비자는 자체 메시지 복사본을 가집니다. 항목에 가입된 소비자가 없는 경우에는 메시지가 삭제됩니다.

Event Notification Service도 게시/가입 구조를 사용하며 Message Queue에 정의된 항목 패턴과 비슷합니다.

대기열.메시지 생성자가 대기열에 메시지를 보낼 때 지점간 구조가 사용됩니다. 이 패턴에서는 생성자가 한 소비자만 메시지를 받을 수 있는 대기열 대상으로 메시지를 보냅니다. 대기열에서 오는 메시지를 여러 소비자가 기다리는 경우에는 가입자 중 하나만 메시지를 받습니다. 메시지를 기다리는 소비자가 없는 경우에는 메시지가 시간 초과되거나 사용자가 대기열에 관심을 보일 때까지 메시지가 보관됩니다.

대기열에 메시지를 생성하면 여러 소비자 사이에 메시지 로드를 분산할 수 있습니다.

22.1.3.2 여러 개의 JMQ 알림 플러그 인 사용

한 개에서 다섯 개까지 알림 플러그 인을 구성할 수 있습니다.

Messaging Server는 다음과 같은 기본 위치에 플러그 인 라이브러리를 제공합니다.

```
/opt/SUNWmsgsr/lib/libjmqnotify
```

`configutil` 유틸리티를 사용하여 플러그 인의 매개 변수를 지정하고 플러그 인이 실행 코드의 라이브러리를 가리키도록 합니다.

두 개 이상의 플러그 인을 지정한 경우 각 플러그 인은 다른 메시지와는 독립적으로 알림 메시지를 생성합니다. 예를 들어, delete-message 매개 변수로 두 개의 플러그 인을 구성하고 사용자의 메일함에서 메시지를 삭제한 경우, 두 플러그 인 모두가 알림 메시지를 생성합니다.

여러 개의 플러그 인을 구성하면 목적에 따라 다른 메시지 배포 패턴을 사용할 수 있습니다. 예를 들어, 각각 다음에 메시지를 생성하도록 서로 다른 세 개의 플러그 인을 구성할 수 있습니다.

- 대기열(Message Queue 사용)
- 항목(Message Queue 사용)
- Event Notification Service

22.1.3.3 알림 플러그 인의 매개 변수 구성

구성하는 각 플러그 인에 대해 별도의 `configutil` 매개 변수 집합을 정의해야 합니다.

매개 변수에 따라 두 가지 정보가 결정됩니다.

- 생성할 알림 메시지의 유형. 예를 들어, `LogUser` 매개 변수를 사용하면 사용자가 로그인하거나 로그아웃할 때마다 알림 메시지가 전송됩니다.
- Message Queue에 필요한 구성 정보. 예를 들어 `jmqHost` 매개 변수는 Message Queue 브로커가 실행 중인 호스트의 IP 주소를 식별합니다.

플러그 인을 구성하는 방법에 대한 지침은 660 페이지 "JMQ 알림 플러그 인 구성 방법"을 참조하십시오.

22.2 JMQ 알림 서비스 구성

이 절에서는 JMQ 알림 플러그 인이 전체 Message Queue 서비스 컨텍스트에 맞게 들어가는 방식에 대해 간략하게 설명합니다. 그런 다음 JMQ 알림 플러그 인을 구성하는 자세한 지침을 제공합니다.

22.2.1 JMQ 알림 서비스 계획

JMQ 알림 플러그 인은 Message Queue 서비스의 한 부분에 불과합니다. 메시징 서비스에는 메시지를 소비하는 클라이언트와 Message Queue 인프라(브로커, 관리 구성 요소 등)도 포함됩니다.

다음 단계는 Messaging Server를 지원하는 Message Queue 서비스를 만들기 위해 수행할 작업에 대한 간략한 설명입니다.

1. **알림 메시지 서비스를 설계합니다.**

Messaging Server 설치에 필요한 알림 메시지를 정의합니다. 메시지 서비스 개발 라이프사이클의 계획 및 설계 단계는 이 장에서 다루는 범위를 벗어납니다. 하지만 JMQ 알림 플러그 인을 구성하기 전에 다음과 같은 설계 관련 질문을 생각해 봐야 합니다.

- 알림으로 생성할 메시지 이벤트는 무엇입니까? 사용 가능한 알림 메시지의 목록은 667 페이지 "22.3.1 알림 메시지"를 참조하십시오.
- 메시지를 대기열, 항목, 둘 다 중 어디에 생성하려고 합니까?
- Message Queue 서비스뿐만 아니라 고유 Event Notification Service를 사용하려고 합니까?

이런 질문을 생각해 보면 알림 플러그 인을 하나만 구성할 것인지 여러 개를 구성할 것인지, 그리고 각 플러그인을 어떻게 구성할 것인지 결정하는 데에 도움이 됩니다.

2. **Message Queue 제품을 설치, 구성 및 배포합니다.**

   Message Queue 설치에 대한 자세한 내용은 *Sun Java System Message Queue Installation Guide*를 참조하십시오.

   Message Queue 구성 및 배포에 대한 자세한 내용은 *Sun Java System Message Queue* **관리 설명서**를 참조하십시오.

3. **JMQ 알림 메시지를 소비할 Message Queue 클라이언트를 하나 이상 작성합니다.**

   클라이언트는 Message Queue 클라이언트 API 요구 사항에 맞아야 합니다. 다음 경로에서 C로 작성된 간단한 클라이언트 소스 코드의 예를 찾을 수 있습니다.

   ```
   /opt/SUNWmsgsr/examples/jmqsdk/
   ```

   소스 파일 이름은 `jmqclient.c`입니다.

   이 클라이언트 소스 코드는 `libjmqnotify` 라이브러리에 있는 매개 변수로 정의된 JMQ 알림 메시지로부터 메시지를 받습니다. 그런 다음 메시지 출력을 `stdout`으로 보냅니다.

   C 또는 Java로 Message Queue 클라이언트를 작성하는 방법에 대한 자세한 내용은 *Sun Java System Message Queue Developer's Guide for C Clients* 또는 *Sun Java System Message Queue Developer's Guide for Java Clients*를 참조하십시오.

4. **알림 메시지를 생성하도록 JMQ 알림 플러그 인을 구성 및 활성화합니다.**

   이 장의 나머지 부분에서는 알림 플러그 인을 구성하는 방법에 대해 설명합니다.

5. **런타임 Message Queue 클라이언트를 구성 및 시작합니다.**

   런타임 Message Queue 클라이언트 배포에 대한 자세한 내용은 *Sun Java System Message Queue* **관리 설명서**를 참조하십시오.

▼ JMQ 알림 플러그 인 구성 방법

이 절차에서는 먼저 알림을 생성하는 메시지 이벤트를 구성합니다. 다음으로는 Message Queue에 필요한 정보를 지정합니다. 마지막으로(단계 9) 플러그 인 라이브러리 이름 다음에 매개 변수를 지정하여 플러그 인 이름을 구성합니다.

```
'/opt/SUNWmsgsr/lib/libjmqnotify$plug-in_name'
```

플러그 인 이름을 지정하지 않으면 기본적으로 `jmqnotify`가 사용됩니다.

시작하기 전에 다음 제품을 설치, 구성 및 배포해야 합니다.

- Sun Java System Messaging Server
- Sun Java System Message Queue 3.6 SP3 2005Q4 이상

주 – 다음 단계에서 구성하는 대부분의 `configutil` 매개 변수는 선택 사항입니다. 기본값 목록은 표 22–2를 참조하십시오.

---

1 알림 메시지 매개 변수를 구성합니다.

플러그 인에 포함할 각 알림 메시지 유형에 대해 `configutil` 유틸리티와 함께 `local.store.notifyplugin` 명령을 사용합니다.

예를 들어 새 메시지에 대해 알림을 사용하려면 다음을 입력합니다.

```
configutil -o local.store.notifyplugin.jmqnotify.NewMsg.enable -v 1
```

여기서 *jmqnotify*는 플러그 인의 이름이며

`-v 1`은 이 메시지의 알림을 활성화합니다. `0` 값은 이 메시지의 알림을 비활성화합니다.

모든 JMQ 알림 메시지의 목록은 667 페이지 “22.3.1 알림 메시지”를 참조하십시오.

JMQ 알림 메시지를 활성화하는 `configutil` 매개 변수의 정의는 *Sun Java System Messaging Server Administration Reference*의 "3장: Messaging Server Configuration"을 참조하십시오.

알림 메시지 중에는 두 개 이상의 `configutil` 매개 변수를 사용하여 추가 기능이 있는 메시지를 활성화하는 것도 있습니다. 예를 들어, 일부 메시지의 알림 텍스트에는 메시지 헤더가 포함되어 있습니다. 이런 메시지의 구성 방법에 대한 지침은 676 페이지 “newflags 및 oldflags 등록 정보의 구문”을 참조하십시오.

주 – 구성하는 각 플러그 인에 대해 별도로 매개 변수를 구성해야 합니다.

따라서 `jmq1`과 `jmq2`라는 두 매개 변수를 구성한 경우 두 플러그 인 모두에서 새 메시지 알림을 활성화하려면 `local.store.notifyplugin` 명령을 두 번 실행해야 합니다.

```
configutil -o local.store.notifyplugin.jmq1.NewMsg.enable -v 1

configutil -o local.store.notifyplugin.jmq2.NewMsg.enable -v 1
```

2 Message Queue 대상(브로커)이 실행 중인 호스트를 지정합니다.

예를 들어, 다음 명령을 입력합니다.

```
configutil -o local.store.notifyplugin.jmqnotify.jmqHost -v "127.0.0.1"
```

여기서 *jmqnotify*는 플러그 인의 이름이며

`"127.0.0.1"`은 Message Queue 브로커의 호스트 시스템 IP 주소입니다.

3 Message Queue 브로커의 포트를 지정합니다.

예를 들어, 다음 명령을 입력합니다.

```
configutil -o local.store.notifyplugin.jmqnotify.jmqPort -v "7676"
```

여기서 *jmqnotify*는 플러그 인의 이름이며

`"7676"`은 Message Queue 브로커의 포트입니다.

4 서비스에 메시지를 생성할 권한이 있는 Message Queue 사용자의 사용자 아이디와 비밀번호를 지정합니다.

예를 들어, 다음 명령을 입력합니다.

```
configutil -o local.store.notifyplugin.jmqnotify.jmqUser -v "guest"

configutil -o local.store.notifyplugin.jmqnotify.jmqPwd -v "%$#a62t&"
```

여기서 *jmqnotify*는 플러그 인의 이름이며

`"guest"`와 `"%$#a62t&"`는 각각 Message Queue 사용자의 사용자 아이디와 비밀번호입니다.

5 대상의 유형(항목 또는 대기열)과 메시지를 보낼 대상의 이름을 구성합니다.

다음 단계를 수행합니다.

a. 대상이 항목인지 대기열인지 지정합니다.

예를 들어, 다음 명령을 입력합니다.

```
configutil -o local.store.notifyplugin.jmqnotify.DestinationType -v "queue"
```

여기서 *jmqnotify*는 플러그 인의 이름이며

"queue"는 대상이 대기열이 되도록 지정합니다. 이 매개 변수에 허용되는 값은 "queue"와 "topic"입니다.

b. 대상 이름을 지정합니다.

예를 들어, 다음 명령 중 하나를 입력합니다.

```
configutil -o local.store.notifyplugin.jmqnotify.jmqQueue -v "JES-MS"
```

또는

```
configutil -o local.store.notifyplugin.jmqnotify.jmqTopic -v "JES-MS"
```

여기서 *jmqnotify*는 플러그 인의 이름이며

jmqQueue 또는 jmqTopic은 대상 유형을 나타냅니다. jmqQueue 및 jmqTopic 매개 변수는 서로 동의어이며 동시에 사용할 수 없습니다. 한 플러그 인에는 이 매개 변수 중 하나만 사용할 수 있습니다.

"JES-MS"는 메시지를 보낼 대기열 또는 항목의 이름 예입니다.

6 메시지 우선 순위를 지정합니다.

예를 들어, 다음 명령을 입력합니다.

```
configutil -o local.store.notifyplugin.jmqnotify.Priority -v 3
```

여기서 *jmqnotify*는 플러그 인의 이름이며

-v 3은 이 플러그 인으로 생성되는 메시지에 지정할 Message Queue 우선 순위입니다.

Priority의 기본값은 4입니다.

7 Message Queue 브로커에 메시지를 보관하는 기간(밀리초)을 지정합니다.

예를 들어, 다음 명령을 입력합니다.

```
configutil -o local.store.notifyplugin.jmqnotify.ttl -v 100
```

여기서 *jmqnotify*는 플러그 인의 이름이며

-v 100은 Message Queue 서비스에서 메시지를 전달하거나 삭제하기 전에 100밀리초 동안 보관하도록 지정합니다. 0 값은 메시지를 영구적으로 보관하며 시간 초과가 발생하지 않음을 의미합니다.

8 메시지의 지속성을 지정합니다.

예를 들어, 다음 명령을 입력합니다.

```
configutil -o local.store.notifyplugin.jmqnotify.Persistent -v 1
```

여기서 *jmqnotify*는 플러그 인의 이름이며

-v 1은 Message Queue 서비스에 영구 메시지를 사용하도록 지정합니다. 허용되는 값은 1(영구)과 0(비영구)입니다.

9 플러그 인 이름을 지정합니다.

기본 이름으로 플러그 인 하나를 구성하려면 플러그 인 라이브러리의 정규화된 이름을 입력하거나 라이브러리 이름과 해당 플러그 인 매개 변수를 입력합니다.

```
configutil -o local.store.notifyplugin -v /opt/SUNWmsgsr/lib/libjmqnotify
```

또는

```
configutil -o local.store.notifyplugin -v '/opt/SUNWmsgsr/lib/libjmqnotify$jmqnotify'
```

여기서 `libjmqnotify`는 라이브러리 이름이며

`jmqnotify`는 플러그 인 매개 변수의 기본 이름입니다.

달러 기호($)를 사용하여 라이브러리 이름과 매개 변수를 분리합니다.

전체 값을 작은 따옴표로 묶어야 합니다('*value*'). 그렇게 하지 않으면 쉘에서 달러 기호를 해석합니다.

기본 플러그 인에서 읽는 `configutil` 매개 변수는 다음과 같은 이름을 가집니다.

```
local.store.notifyplugin.jmqnotify.*
```

`jmq42` 등의 다른 플러그 인 이름을 구성하려면 다음 명령을 입력합니다.

```
configutil -o local.store.notifyplugin -v '/opt/SUNWmsgsr/lib/libjmqnotify$jmq42'
```

`jmq42` 플러그 인에서 읽는 `configutil` 매개 변수는 다음과 같은 이름을 가집니다.

```
local.store.notifyplugin.jmq42.*
```

▼ 여러 개의 플러그 인 구성 방법

1 만들려는 각 플러그 인에 대해 별도의 JMQ 알림 매개 변수 집합을 구성합니다.

예를 들어, `jmq1`과 `jmq2`라는 두 개의 플러그 인을 구성하는 경우를 가정합니다. 새 메시지 알림은 두 플러그 인 모두에 대해 활성화하고 제거된 메시지 알림은 `jmq2` 플러그 인에 대해서만 활성화하려 합니다. 이 경우 다음과 같이 `local.store.notifyplugin` 명령을 세 번 실행합니다.

```
configutil -o local.store.notifyplugin.jmq1.NewMsg.enable -v 1

configutil -o local.store.notifyplugin.jmq2.NewMsg.enable -v 1

configutil -o local.store.notifyplugin.jmq2.PurgeMsg.enable -v 1
```

플러그 인이 Message Queue 서비스와 통신하도록 하는 매개 변수도 지정해야 합니다.

모든 알림 매개 변수를 구성하는 단계별 지침은 660 페이지 "JMQ 알림 플러그 인 구성 방법"을 참조하십시오.

2 플러그 인 이름을 구성합니다.

jmq1과 jmq2라는 두 개의 플러그 인을 구성하려면 다음 명령을 입력합니다.

```
configutil -o local.store.notifyplugin
-v '/opt/SUNWmsgsr/lib/libjmqnotify$jmq1$$/opt/SUNWmsgsr/ \
lib/libjmqnotify$jmq2'
```

이 예에서는 플러그 인 라이브러리의 인스턴스 두 개가 실행됩니다.

달러 기호 하나($)를 사용하여 라이브러리 이름과 플러그 인 이름을 지정하는 매개 변수를 분리합니다.

달러 기호 두 개($$)를 사용하여 첫째 플러그 인 인스턴스를 다음 인스턴스와 분리합니다.

전체 값을 작은 따옴표로 묶어야 합니다('*value*'). 그렇게 하지 않으면 쉘에서 달러 기호를 해석합니다.

이 예에서 첫째 인스턴스는 jmq1이라는 이름의 매개 변수에서 구성을 작성합니다.

```
local.store.notify.jmq1.*
```

둘째 인스턴스는 jmq2라는 이름의 매개 변수에서 구성을 작성합니다.

```
local.store.notify.jmq2.*
```

22.2.2 두 개 이상의 configutil 매개 변수를 사용하는 알림 메시지 지정

대부분의 알림 메시지에서는 local.store.notifyplugin 명령을 하나만 실행하여 메시지를 지정합니다.

하지만 다음 알림 메시지는 두 개 이상의 local.store.notifyplugin 명령으로 구성할 수 있습니다.

1. NewMsg
2. UpdateMsg
3. DeleteMsg
4. MsgFlags

다음 절차에서는 이런 알림 메시지를 설정하는 방법에 대해 설명합니다.

▼ 메시지 헤더와 메시지 본문에 새 메시지 및 업데이트된 메시지 알림 구성

새로운 전자 메일 메시지나 업데이트된 메시지가 있는 경우에 보내는 알림 메시지의 텍스트에 메시지 헤더와 메시지 본문을 추가할 수 있습니다.

메시지 헤더 및 메시지 본문을 포함하는 것은 선택 사항입니다. 둘 다 포함할 수도 있고, 하나만 포함할 수도 있고, 둘 다 포함하지 않을 수도 있습니다. 기본값은 메시지 헤더나 메시지 본문 없이 메시지를 보내는 것입니다.

1 새 메시지 또는 업데이트된 메시지 알림을 지정합니다.

```
configutil -o local.store.notifyplugin.jmqnotify.NewMsg.enable -v 1

configutil -o local.store.notifyplugin.jmqnotify.UpdateMsg.enable -v 1
```

여기서 *jmqnotify*는 플러그 인의 이름이며

`-v 1`은 이런 메시지의 알림을 활성화합니다. `0` 값은 알림을 비활성화합니다.

2 다음 예와 같이 `maxHeaderSize` 매개 변수에 0보다 큰 값을 지정합니다.

```
configutil -o local.store.notifyplugin.jmqnotify.maxHeaderSize -v 1024
```

여기서 *jmqnotify*는 플러그 인의 이름이며

`1024`는 보낼 헤더의 최대 크기입니다. `maxHeaderSize`의 기본값은 `0`이며, 이 값을 설정하면 메시지와 함께 헤더 정보를 보내지 않습니다.

3 다음 예와 같이 `maxBodySize` 매개 변수에 0보다 큰 값을 지정합니다.

```
configutil -o local.store.notifyplugin.jmqnotify.maxBodySize -v 1024
```

여기서 *jmqnotify*는 플러그 인의 이름이며

`5120`은 보낼 메시지 본문의 최대 크기입니다. `maxBodySize`의 기본값은 `0`이며, 이 값을 설정하면 메시지와 함께 본문을 보내지 않습니다.

▼ 메시지 헤더가 있는 삭제된 메시지 알림 구성 방법

전자 메일 메시지가 삭제된 경우 알림 메시지 텍스트에 메시지 헤더를 추가할 수 있습니다.

메시지 헤더를 포함하는 것은 선택 사항입니다. 기본값은 메시지 헤더 없이 알림을 보내는 것입니다.

1 전자 메일 메시지가 삭제되면 전송되도록 알림을 활성화합니다.

```
configutil -o local.store.notifyplugin.jmqnotify.DeleteMsg.enable -v 1
```

여기서 *jmqnotify*는 플러그 인의 이름이며

`-v 1`은 이 메시지의 알림을 활성화합니다. `0` 값은 알림을 비활성화합니다.

2 `ExpungeHeaders` 매개 변수를 지정합니다.

```
configutil -o local.store.notifyplugin.jmqnotify.ExpungeHeaders -v 1
```

여기서 *jmqnotify*는 플러그 인의 이름이며

`-v 1`은 삭제된 메시지 알림과 함께 메시지 헤더가 전송되도록 합니다. `ExpungeHeaders`의 기본값은 `0`입니다. 이 값을 설정하면 삭제된 메시지 알림에 헤더 정보가 함께 전송되지 않습니다.

`DeleteMsg` 메시지가 메시지 헤더를 전달하도록 `ExpungeHeaders` 매개 변수를 구성해야 합니다.

3 다음 예와 같이 `maxHeaderSize` 매개 변수에 0보다 큰 값을 지정합니다.

```
configutil -o local.store.notifyplugin.jmqnotify.maxHeaderSize -v 1024
```

여기서 *jmqnotify*는 플러그 인의 이름이며

`1024`는 보낼 헤더의 최대 크기입니다. `maxHeaderSize`의 기본값은 `0`이며, 이 값을 설정하면 메시지와 함께 헤더 정보를 보내지 않습니다.

22.2.2.1 메시지 상태 변경에 대한 알림 구성

전자 메일 메시지의 상태가 변경된 경우에 알림 메시지를 보내도록 구성할 수 있습니다.

메시지 플래그 알림으로 전달되는 정보

다음과 같은 이유로 전자 메일 메시지의 상태 플래그가 변경될 때마다 메시지 플래그 알림이 생성됩니다.

- 응답
- 플래그 지정
- 삭제
- 조회(읽음)
- 초안 작성

메시지 플래그 알림이 전송될 때 알림에는 다음 등록 정보가 포함됩니다.

- 상태가 변경되기 전에 전자 메일 메시지에 설정된 플래그
- 상태가 변경된 후에 전자 메일 메시지에 설정된 플래그

이 정보는 5자로 된 문자열인 두 개의 등록 정보 `oldflags`와 `newflags`로 전달됩니다.

이 두 등록 정보의 값에 대한 설명은 676 페이지 “newflags 및 oldflags 등록 정보의 구문”을 참조하십시오.

메시지 플래그 알림에 필요한 Configutil 매개 변수

메시지 플래그 알림을 활성화하려면 다음 `configutil` 매개 변수를 구성해야 합니다.

- `local.store.notifyplugin.MsgFlags`
- `local.store.notifyplugin.*.MsgFlags.enable`

첫째 `MsgFlags` 매개 변수는 IMAP 서버와 메시지 저장소에서 상태 플래그의 변경 값을 식별 및 추적하여 이 정보를 알림 메시지에 전달할 수 있도록 설정합니다.

이 매개 변수는 모든 알림 플러그 인에 적용됩니다. 따라서 메시지 플래그 알림을 사용하는 알림 플러그 인이 **있는** 경우에는 매개 변수를 활성화해야 합니다. 메시지 플래그 알림을 사용하는 플러그 인이 없는 경우에는 이 매개 변수를 비활성화(기본값)해야 합니다.

둘째 매개 변수 `*.MsgFlags.enable`을 사용하면 특정 플러그 인 라이브러리에 대해 메시지 플래그 알림을 전송할 수 있습니다.

주 – 메시지 플래그에 대해 알림을 활성화하려면 두 매개 변수를 모두 구성해야 합니다.

▼ 메시지 상태 플래그가 변경된 경우의 알림 활성 방법

1 **메시지 플래그 알림으로 상태 플래그를 추적하고 상태 정보가 전달되도록 설정합니다.**

```
configutil -o local.store.notifyplugin.MsgFlags -v 1
```

여기서 `-v 1`은 메시지 플래그 알림과 함께 메시지 플래그 정보가 전송되도록 설정합니다. `0` 값은 이 알림을 비활성화합니다.

2 **특정 플러그 인에서 메시지 플래그 알림을 보낼 수 있도록 설정합니다.**

```
configutil -o local.store.notifyplugin.jmqnotify.MsgFlags.enable -v 1
```

여기서 *jmqnotify*는 플러그 인의 이름이며

`-v 1`은 이 플러그 인에 대해 메시지 플래그 알림을 활성화합니다. `0` 값은 알림을 비활성화합니다.

22.3 JMQ 알림 메시지 및 등록 정보

이 절에서는 다음 항목에 대해 설명합니다.

22.3.1 알림 메시지

메시지 저장소에서 발생하는 다양한 종류의 이벤트에 대해 알림 메시지를 생성할 수 있습니다. 예를 들어, 사용자가 로그인하면 `Login` 메시지를 생성하여 Message Queue 브로커에 전달할 수 있습니다.

configutil 매개 변수는 생성할 각 메시지 종류를 지정합니다. 메시지를 생성하는 이벤트는 다양한 configutil 매개 변수를 구성하여 결정합니다. configutil 매개 변수는 하나 이상의 JMQ 알림 플러그 인 라이브러리에서 참조됩니다.

모든 메시지는 대상 유형이 "topic" 또는 "queue" 중 어느 것으로 설정되었는지에 따라 항목 또는 대기열에 전달됩니다. Message Queue 대상을 구성하는 방법에 대한 자세한 내용은 660 페이지 "JMQ 알림 플러그 인 구성 방법"을 참조하십시오.

각 메시지는 다음 메시지 헤더로 식별됩니다.

MQ_MESSAGE_TYPE_HEADER_PROPERTY

JMQ 알림 플러그 인은 다음 표에 표시된 메시지를 지원합니다.

이런 메시지를 활성화하는 configutil 매개 변수의 목록은 671 페이지 "22.3.4 configutil 매개 변수의 기본값"을 참조하십시오.

표 22–1 JMQ 알림 메시지

알림 메시지	설명
DeleteMsg	"삭제됨"으로 표시된 메시지가 메일함에서 제거됩니다. IMAP expunge와 동일합니다.
Login	사용자가 IMAP, HTTP 또는 POP에서 로그인했습니다. (이 메시지는 configutil 매개 변수 local.store.notifyplugin.*.LogUser.enable로 활성화됩니다.)
Logout	사용자가 IMAP, HTTP 또는 POP에서 로그아웃했습니다. (이 메시지는 configutil 매개 변수 local.store.notifyplugin.*.LogUser.enable로 활성화됩니다.)
MsgFlags	메시지의 메시지 플래그가 변경되었습니다. 이 메시지와 함께 이전 플래그와 새 플래그가 전달됩니다.
NewMsg	시스템에서 사용자의 메일함으로 새 메시지를 받았습니다. 메시지 헤더와 본문을 포함할 수 있습니다.
OverQuota	사용자의 메일함이 할당량(diskquota, msgquota) 중 하나를 초과했기 때문에 작업이 실패했습니다. MTA 채널은 할당량이 변경되거나 사용자의 메일함 수가 할당량 아래로 떨어질 때까지 메시지를 보관합니다. MTA에 보관되어 있는 동안 메시지가 만료되면 정리됩니다.
PurgeMsg	서버 프로세스 imexpire에 의해 메일함에 있는 메시지가 정리되었습니다(날짜 만료). 이것이 서버측 정리이고 DeleteMsg가 클라이언트측 정리입니다. 진정한 의미의 제거라고는 할 수 없습니다.
ReadMsg	메일함에 있는 메시지를 읽었습니다. (IMAP 프로토콜에서 메시지가 조회로 표시되었습니다.)

표 22–1 JMQ 알림 메시지 (계속)

알림 메시지	설명
TrashMsg	IMAP 또는 HTTP에 의해 메시지가 삭제 표시되었습니다. 메일 클라이언트의 구성에 따라 폴더에 있는 메시지가 사용자에게 계속 보일 수도 있습니다. 정리를 수행하면 메시지가 폴더에서 제거됩니다.
UnderQuota	할당량이 할당량 초과 상태에서 정상 상태로 돌아왔습니다.
UpdateMsg	IMAP 작업에 의해 메시지가 메일함에 추가되었습니다. 예를 들어, 사용자가 전자 메일 메시지를 메일함에 복사했습니다. 메시지 헤더와 본문을 포함할 수 있습니다.

22.3.2 알림 메시지의 규칙 및 지침

다음 규칙 및 지침이 지원되는 알림 메시지에 적용됩니다.

- 대부분의 알림 메시지 텍스트는 하나의 빈 공간입니다. (Message Queue에서 빈 메시지 본문이 허용되지 않기 때문에 빈 공간을 사용합니다.)예외는 다음과 같습니다.
  - maxHeaderSize 매개 변수로 구성하는 경우 NewMsg, UpdateMsg 및 DeleteMsg 메시지에는 메시지 헤더를 포함할 수 있습니다. maxHeaderSize를 0보다 큰 값으로 설정해야 합니다.

    DeleteMsg 메시지에 메시지 헤더를 포함하려면 ExpungeHeaders 매개 변수 값도 1로 설정해야 합니다.
  - maxBodySize 매개 변수로 구성하는 경우 NewMsg 및 UpdateMsg 메시지에는 메시지 본문을 포함할 수 있습니다. maxBodySize를 0보다 큰 값으로 설정해야 합니다.

    NewMsg 및 UpdateMsg의 경우에는 기본적으로 메시지 본문이 전달되지 않습니다(꺼짐). 따라서 Message Queue의 오버로드가 방지됩니다. 다른 메시지에는 메시지 본문이 포함되지 않습니다.
- INBOX만 변경된 경우 또는 INBOX와 다른 모든 폴더가 변경된 경우에 알림 메시지를 생성할 수 있습니다. 다음 구성 매개 변수가 INBOX만 변경된 경우(값 = 0) 또는 INBOX와 다른 모든 폴더가 변경된 경우(값 = 1) 허용됩니다.

  ```
  local.store.notifyplugin.jmqnotify.noneInbox.enable
  ```

  기본 설정은 INBOX에서만(값 = 0) 메시지를 생성하는 것입니다.

  폴더를 선택하는 기법은 없습니다. 변수를 활성화하면(값 = 1) 모든 폴더가 포함됩니다.
- NewMsg 알림은 "서버에서 메시지를 받아 메시지 대기열에 넣은 후"가 아니라 사용자 메일함에 메시지가 보관된 후에만 발생합니다.
- POP3 클라이언트 액세스에 대해서는 메시지가 생성되지 않습니다.

- XNOTNOTIFY를 실행하면 모든 메시지를 억제할 수 있습니다. 예를 들어 관리에만 사용되던 IMAP 스크립트(사용자에게는 알리지 않음)에서 이를 실행하면 모든 메시지를 억제할 수 있습니다.

22.3.3 특정 메시지 유형의 알림

알림은 텍스트 메시지, 음성 메시지 및 이미지 데이터 등의 여러 메시지 유형에 대해 상태 정보를 전달할 수 있습니다. 같은 메일 폴더에 이러한 서로 다른 메시지 유형을 저장하려는 경우도 많습니다. 예를 들어, 사용자 휴대폰의 받은 메일함에 새로운 텍스트 메시지와 음성 메시지가 도착하는 경우가 있습니다.

이런 메시지 유형을 구성하려면 `store.messagetype.enable`과 같은 `configutil` 명령을 사용합니다. 메시지 유형의 구성 및 관리에 대한 자세한 내용은 "18장: 메시지 저장소 관리"에 있는 "메시지 유형 관리"를 참조하십시오.

메시지 유형을 구성하고 나면 JMQ 알림 메시지에서 특정 메시지 유형을 식별할 수 있습니다. 메시지 유형별로 알림 메시지를 해석하고 각 유형에 대한 상태 정보를 메일 클라이언트로 전달하도록 Message Queue 클라이언트를 작성할 수 있습니다.

예를 들어, 서로 다른 여러 유형의 새 메시지가 사용자의 메일함에 도착하는 경우를 가정할 수 있습니다. `NewMsg` 알림 메시지는 예를 들어 새 음성 메일 메시지 7개와 새 텍스트 메시지 4개가 사용자의 받은 메일함에 있다고 알리는 데이터를 전달할 수 있습니다.

다음 알림 메시지는 특정 메시지 유형을 추적하는 정보를 전달할 수 있습니다.

```
NewMsg
UpdateMsg
ReadMsg
TrashMsg
DeleteMsg
PurgeMsg
OverQuota
UnderQuota
```

JMQ 알림 기능에서는 현재 메일함에 있는 메시지의 수를 메시지 유형별로 계산합니다. 수 값이 하나 전달되는 대신 각 메시지 유형의 수를 지정하는 배열이 알림 메시지와 함께 전달됩니다.

메시지별 개수 값은 `numMsgs` 등록 정보를 통해 알림 메시지와 함께 전달됩니다. `ReadMsg` 및 `TrashMsg` 알림 메시지의 경우 조회된 메시지의 수(`numSeen`)와 삭제됨으로 표시된 수(`numDeleted`) 역시 메시지 유형별로 계산됩니다.

주 – Event Notification Service는 메시지 유형을 지원하지 않습니다. 메시지 유형에 대한 정보를 전달하려면 JMQ 알림 플러그 인을 사용하십시오.

22.3.4 configutil 매개 변수의 기본값

알림 메시지와 Message Queue에 필요한 구성 정보는 `configutil` 매개 변수로 구성됩니다.

표 22–2에는 이런 매개 변수와 그 기본값이 나와 있습니다.

`configutil` 매개 변수에 대한 자세한 정의는 *Sun Java System Messaging Server Administration Reference*의 "3장: Messaging Server Configuration"을 참조하십시오.

표 22–2 `configutil` 매개 변수 및 기본값

`configutil` 매개 변수	기본값
`local.store.notifyplugin.*.maxBodySize`	0 — 비활성화
`local.store.notifyplugin.*.maxHeaderSize`	0 — 비활성화
`local.store.notifyplugin.*.NewMsg.enable`	1 — 활성화
`local.store.notifyplugin.*.UpdateMsg.enable`	1 — 활성화
`local.store.notifyplugin.*.ReadMsg.enable`	1 — 활성화
`local.store.notifyplugin.*.DeleteMsg.enable`	1 — 활성화
`local.store.notifyplugin.*.PurgeMsg.enable`	1 — 활성화
`local.store.notifyplugin.*.LogUser.enable`	1 — 활성화
`local.store.notifyplugin.*.MsgFlags.enable`	0 — 비활성화
`local.store.notifyplugin.*.noneInBox.enable`	0 — 비활성화
`local.store.notifyplugin.*.jmqHost`	“127.0.0.1”
`local.store.notifyplugin.*.jmqPort`	7676
`local.store.notifyplugin.*.jmqTopic`	“JES-MS”
`local.store.notifyplugin.*.jmqQueue`	“JES-MS”
`local.store.notifyplugin.*.jmqUser`	“guest”
`local.store.notifyplugin.*.jmqPwd`	“guest”
`local.store.notifyplugin.*.destinationtype`	“topic”

표 22–2 configutil 매개 변수 및 기본값 (계속)

configutil 매개 변수	기본값
local.store.notifyplugin.*.Priority	4
local.store.notifyplugin.*.ttl	0 — 메시지가 시간 초과되지 않음을 나타냅니다.
local.store.notifyplugin.*.Persistent	1 — 활성화

22.3.5 알림 메시지 등록 정보

모든 메시지는 등록 정보에 정의된 추가 정보를 전달합니다. 메시지마다 다른 등록 정보가 있습니다. 예를 들어, NewMsg는 새 메시지의 IMAP uid를 나타냅니다.

22.3.5.1 표준 알림 메시지 등록 정보

표 22–3에서는 표준 알림 메시지 등록 정보에 대해 설명합니다. 이 등록 정보는 모든 JMS 메시지에 있습니다.

표 22–3 표준 알림 메시지 등록 정보

등록 정보	데이터 유형	설명
hostname	ConstMQString	메시지를 생성한 시스템의 호스트 이름입니다.
pid	MQInt32	메시지를 생성한 프로세스의 ID입니다.
process	ConstMQString	메시지를 생성한 프로세스의 이름을 지정합니다.
timestamp	MQFloat64	기준 시점(GMT로 1970년 1월 1일 자정)으로부터 지난 시간(밀리초)을 지정합니다.

22.3.5.2 특정 알림 메시지에만 해당되는 등록 정보

표 22–4에서는 특정 알림 메시지와 함께 전달되는 등록 정보에 대해 설명합니다.

각 메시지에는 아래 표에 표시된 등록 정보의 일부가 포함됩니다. 각 메시지에 연결된 등록 정보의 목록은 표 22–5를 참조하십시오.

표 22–4 특정 알림 메시지에만 해당되는 등록 정보

등록 정보	데이터 유형	설명
client	ConstMQString	메시지에 연결된 Message Queue 클라이언트의 IP 주소입니다.

표 22–4 특정 알림 메시지에만 해당되는 등록 정보 (계속)

등록 정보	데이터 유형	설명
diskquota	MQInt32	메시지에 연결된 사용자의 디스크 공간 할당량(KB)입니다. 값이 -1로 설정되면 할당량이 없는 것을 나타냅니다.
diskquotaused	MQInt32	메시지와 연결된 사용자가 사용하는 디스크 공간 크기(KB)입니다.
hdrLen	MQInt32	메시지 헤더의 크기입니다. 메시지 본문에 있는 헤더가 잘렸을 수도 있기 때문에 해당 헤더의 크기와 이 값이 다를 수도 있습니다.
imapUid	MQInt32	메시지와 연결된 IMAP uid 등록 정보입니다.
lastUid	MQInt32	메일함에 사용된 마지막 IMAP uid 값입니다.
mailboxName	ConstMQstring	이벤트와 연결된 메시지 저장소 메일함 이름입니다. mailboxName에는 다음 형식 중 하나가 지정됩니다. 여기서 uid는 사용자의 고유 식별자입니다. uid — 기본(주) 도메인에 있는 사용자의 받은 메일함을 식별합니다. uid@domain — 호스트된 도메인에 있는 사용자의 받은 메일함을 식별합니다. uid/mailboxname — 기본 도메인에 있는 사용자의 최상위 메일함을 식별합니다. uid@domain/mailboxname — 호스트된 도메인에 있는 사용자의 최상위 메일함을 식별합니다. uid/foldername/mailboxname — 기본 도메인에 있는 사용자 폴더의 메일함을 식별합니다. uid@domain/foldername/mailboxname — 호스트된 도메인에 있는 사용자 폴더의 메일함을 식별합니다.
msgquota	MQInt32	최대 메시지 수에 대한 사용자의 할당량입니다. 값이 -1로 설정되면 할당량이 없는 것을 나타냅니다.
newflags	ConstMQString	현재 작업으로 인해 플래그가 변경된 후에 사용자의 메일함 메시지에 설정되는 플래그입니다. MsgFlags 알림 메시지가 생성된 경우에는 항상 이 등록 정보가 oldflags와 함께 존재합니다. newflags의 구문과 값은 이 표 아래 있는 676 페이지 “newflags 및 oldflags 등록 정보의 구문”을 참조하십시오.

표 22–4 특정 알림 메시지에만 해당되는 등록 정보 *(계속)*

등록 정보	데이터 유형	설명
numDeleted	MQInt32	메일함에서 삭제됨으로 표시된 메시지의 수입니다. 이 수는 메일함 소유자가 삭제한 메시지의 수를 계산합니다. 다른 사용자가 메일함에 액세스할 수 있는 경우 메일함에서 다른 사용자가 수행하는 작업은 이 수에 포함되지 않습니다. (다른 사용자의 작업으로 DeleteMsg 등의 알림이 트리거될 수는 있습니다.)
numDeleted*nn*	MQInt32	메일함에서 삭제됨으로 표시된 메시지의 총 수를 나타내며, 각 메시지 유형별로 지정됩니다. 메시지 유형이 구성된 경우 numDeleted*nn* 등록 정보는 각 메시지 유형 *nn*에 대한 수를 전달합니다. numDeleted 등록 정보는 항상 전달됩니다. 이 등록 정보는 모든 유형을 포함하여 삭제됨으로 표시된 모든 메시지의 총 수를 계산합니다. 예를 들어 20개의 메시지가 삭제됨으로 표시된 경우 10개가 유형 3이고, 7개가 유형 16이고, 나머지 메시지의 유형은 인식되지 않았다면 알림과 함께 다음 등록 정보 및 수 값이 전달됩니다. numDeleted=20 numDeleted3=10 numDeleted16=7
numMsgs	MQInt32	현재 메일함에 있는 총 메시지 수입니다.
numMsgs*nn*	MQInt32	현재 메일함에 있는 메시지의 총 수를 나타내며 각 메시지 유형별로 지정됩니다. 메시지 유형이 구성된 경우 numMsgs*nn* 등록 정보는 각 메시지 유형 *nn*에 해당하는 개수를 전달합니다. numMsgs 등록 정보는 항상 전달됩니다. 이 등록 정보는 모든 유형을 포함하여 메일함에 있는 모든 메시지의 총 수를 나타냅니다. 예를 들어, 현재 메일함에 20개의 메시지가 있는 경우 10개가 유형 3이고, 7개가 유형 16이고, 나머지 메시지의 유형은 인식되지 않았다면 알림과 함께 다음 등록 정보 및 수 값이 전달됩니다. numMsgs=20 numMsgs3=10 numMsgs16=7

표 22–4 특정 알림 메시지에만 해당되는 등록 정보 *(계속)*

등록 정보	데이터 유형	설명
numSeen	MQInt32	메일함에서 조회(읽음)로 표시된 메시지의 수입니다. 이 수는 메일함 소유자가 읽은 메시지의 수를 계산합니다. 다른 사용자가 메일함에 액세스할 수 있는 경우 메일함에서 다른 사용자가 수행하는 작업은 이 수에 포함되지 않습니다. (다른 사용자의 작업으로 ReadMsg 등의 알림이 트리거될 수는 있습니다).
numSeen*nn*	MQInt32	메일함에서 조회(읽음)으로 표시된 메시지의 총 수를 나타내며, 각 메시지 유형별로 지정됩니다. 메시지 유형이 구성된 경우 numSeen*nn* 등록 정보는 각 메시지 유형 *nn*에 해당하는 개수를 전달합니다. numSeen 등록 정보는 항상 전달됩니다. 여기서는 모든 유형을 총괄하여 조회로 표시된 모든 메시지의 수를 나타냅니다. 예를 들어 20개의 메시지가 조회로 표시된 경우 10개가 유형 3이고, 7개가 유형 16이고, 나머지 메시지의 유형은 인식되지 않았다면 알림과 함께 다음 등록 정보 및 수 값이 전달됩니다. numSeen=20 numSeen3=10 numSeen16=7
numSeenDeleted	MQInt32	메일함에서 조회(읽음)으로 표시되고 삭제됨으로 표시된 메시지의 수입니다. 이 값은 읽음으로 표시되고 메일함 소유자에 의해 삭제된 메시지의 수를 나타냅니다. 다른 사용자가 메일함에 액세스할 수 있는 경우 메일함에서 다른 사용자가 수행하는 작업은 이 수에 포함되지 않습니다. (하지만 다른 사용자의 작업으로 ReadMsg 및 DeleteMsg와 같은 알림이 트리거될 수는 있습니다).

표 22–4 특정 알림 메시지에만 해당되는 등록 정보 *(계속)*

등록 정보	데이터 유형	설명

`numSeenDeleted`*nn*	MQInt32	메일함에서 조회(읽음)으로 표시되고 삭제됨으로 표시된 메시지의 총 수를 나타내며, 각 메시지 유형별로 지정됩니다. 메시지 유형이 구성된 경우 `numSeenDeleted`*nn* 등록 정보는 각 메시지 유형 *nn*에 해당하는 개수를 전달합니다.
		
		`numSeenDeleted` 등록 정보는 항상 전달됩니다. 이 등록 정보는 모든 유형을 포함하여 읽음 및 삭제됨으로 표시된 모든 메시지의 총 수를 나타냅니다.
		
		예를 들어 20개의 메시지가 읽음 및 삭제됨으로 표시된 경우 10개가 유형 3이고, 7개가 유형 16이고, 나머지 메시지의 유형은 인식되지 않았다면 알림과 함께 다음 등록 정보 및 수 값이 전달됩니다.
		
		`numSeenDeleted=20`
		
		`numSeenDeleted3=10`
		
		`numSeenDeleted16=7`

`oldflags`	ConstMQString	현재 작업으로 인해 플래그가 변경되기 전에 사용자의 메일함 메시지에 설정되었던 플래그입니다. `MsgFlags` 알림 메시지가 생성된 경우에는 항상 이 등록 정보가 `newflags`와 함께 존재합니다. `oldflags`의 구문과 값은 이 표 아래 있는 676 페이지 “newflags 및 oldflags 등록 정보의 구문”을 참조하십시오.
`quotaRoot`	ConstMQString	이 값은 사용자 이름, 폴더 이름 또는 메시지 유형이 될 수 있습니다.
`size`	MQInt32	메시지의 크기입니다. 본문은 보통 메시지가 잘린 버전이기 때문에 이 값은 메시지 본문의 크기와 다를 수 있습니다.
`uidValidity`	MQInt32	IMAP uid 유효성 등록 정보입니다.
`userid`	ConstMQString	메시지와 연결된 사용자 아이디입니다.

주 – 가입자는 메시지 참조를 구문 분석할 때 기록되지 않은 등록 정보를 허용해야 합니다. 그러면 이후에 새 등록 정보가 추가되어도 호환성을 유지할 수 있습니다.

newflags 및 oldflags 등록 정보의 구문

`newflags` 및 `oldflags` 등록 정보는 5자로 된 문자열입니다. 문자열에는 다음 값이 있어야 합니다.

- /answered 플래그가 설정된 경우 첫 문자는 "A"입니다. 설정되지 않은 경우 비어 있습니다(" ").
- /flagged 플래그가 설정된 경우 둘째 문자는 "F"입니다. 설정되지 않은 경우 비어 있습니다(" ").
- /deleted 플래그가 설정된 경우 셋째 문자는 "D"입니다. 설정되지 않은 경우 비어 있습니다(" ").
- /seen 플래그가 설정된 경우 넷째 문자는 "S"입니다. 설정되지 않은 경우 비어 있습니다(" ").
- /draft 플래그가 설정된 경우 다섯째 문자는 "R"입니다. 설정되지 않은 경우 비어 있습니다(" ").

22.3.5.3 각 알림 메시지와 함께 전달되는 등록 정보

표 22–5에서는 각 알림 메시지와 연결된 등록 정보가 표시되어 있습니다.

예를 들어 TrashMsg 메시지에 적용되는 등록 정보를 보려면 "ReadMsg, TrashMsg"의 열 헤더를 살펴 봅니다.TrashMsg 메시지에는 표준 등록 정보 외에도 mailboxName, numMsgs, uidValidity, numSeen 및 numDeleted를 사용할 수 있습니다.

표 22–5 각 알림 메시지와 함께 전달되는 등록 정보

등록 정보	NewMsg, UpdateMsg	ReadMsg, TrashMsg	DeleteMsg, PurgeMsg	MsgFlags	Login, Logout	OverQuota, UnderQuota
client	아니요	아니요	아니요	아니요	예	아니요
diskquota	아니요	아니요	아니요	아니요	아니요	예
diskquotaused	아니요	아니요	아니요	아니요	아니요	예
hdrLen	예	아니요	아니요	예	아니요	아니요
hostname	예	예	예	예	예	예
imapUid	예	아니요	예	예	아니요	아니요
lastUid	아니요	아니요	예	아니요	아니요	아니요
mailboxName	예	예	예	예	아니요	아니요
msgquota	아니요	아니요	아니요	아니요	아니요	예
newflags	아니요	아니요	아니요	예	아니요	아니요
numDeleted	예	예	예	아니요	아니요	아니요
numDeleted*n*	예*	예*	예*	아니요	아니요	아니요
numMsgs	예	예	예	아니요	아니요	예

표 22–5 각 알림 메시지와 함께 전달되는 등록 정보 *(**계속**)*

등록 정보	NewMsg, UpdateMsg	ReadMsg, TrashMsg	DeleteMsg, PurgeMsg	MsgFlags	Login, Logout	OverQuota, UnderQuota
numMsgs*n*	예*	예*	예*	아니요	아니요	아니요
numSeen	예	예	예	아니요	아니요	아니요
numSeen*n*	예*	예*	예*	아니요	아니요	아니요
numSeenDeleted	예	예	예	아니요	아니요	아니요
numSeenDeleted*n*	예*	예*	예*	아니요	아니요	아니요
oldflags	아니요	아니요	아니요	예	아니요	아니요
Owner	아니요	예	아니요	아니요	아니요	아니요
pid	예	예	예	예	예	예
process	예	예	예	예	예	예
quotaRoot	아니요	아니요	아니요	아니요	아니요	예
size	예	아니요	아니요	아니요	아니요	아니요
timestamp	예	예	예	예	예	예
uidValidity	예	예	예	예	아니요	아니요
userid	아니요	예	아니요	아니요	예	예

주 – * numDeleted*n*, numMsgs *n*, numSeen*n* 및 numSeenDeleted*n* 등록 정보는 메시지 저장소에 메시지 유형이 정의된 경우에만 알림과 함께 전달됩니다.

◆ ◆ ◆ 23장

23 보안 및 액세스 제어 구성

Messaging Server는 메시지를 가로챌 수 없게 하고 침입자가 사용자 또는 관리자로 가장하는 것을 금지하며 특정 사용자에게만 메시징 시스템의 특정 부분에 대한 액세스를 허용할 수 있는 다양하고 유연한 보안 기능을 지원합니다.

Messaging Server 보안 구조는 Sun Java System 서버 전체의 보안 구조 중 일부입니다. 이 구조는 최대한의 상호 운용성과 일관성을 위해 업계 표준 및 공개 프로토콜에 기반을 둡니다.

이 장은 다음 내용으로 구성되어 있습니다.

23.1 서버 보안 정보

서버 보안에는 광범위한 주제가 포함됩니다. 대부분의 기업에서는 허가된 사용자만 서버에 액세스하고, 비밀번호나 아이디의 손상을 방지하며, 사용자가 통신 중에 다른 사람을 나타내지 않도록 하고, 필요 시 비밀리에 통신이 이루어지도록 보장하는 것이 메시징 시스템의 중요한 요구 사항입니다.

서버 통신의 보안이 다양한 방식으로 손상될 수 있기 때문에 보안을 향상시키는 여러 방법이 존재합니다. 이 장에서는 암호화, 인증 및 액세스 제어를 설정하는 방법에 대해 중점적으로 설명합니다. 다음과 같은 Messaging Server의 보안 관련 주제가 이 장에서 다루어집니다.

- **사용자 아이디 및 비밀번호 로그인:** IMAP, POP, HTTP 또는 SMTP에 로그인하려면 사용자 아이디와 비밀번호를 입력해야 하며 보낸 사람 인증을 메시지 수신자에게 전송하려면 SMTP 비밀번호 로그인을 사용해야 합니다.
- **암호화 및 인증:** 통신 및 인증 클라이언트를 암호화하기 위해 TLS 및 SSL 프로토콜을 사용하도록 서버를 설정합니다.
- **관리자 액세스 제어:**액세스 제어 기능을 사용하여 Messaging Server 및 일부 개별 작업에 대한 액세스를 위임합니다.
- **TCP 클라이언트 액세스 제어:** 필터링 기술을 사용하여 서버의 POP, IMAP, HTTP 및 인증된 SMTP 서비스에 연결할 수 있는 클라이언트를 제어합니다.

Messaging Server와 관련된 모든 보안 및 액세스 문제가 이 장에서 설명되지는 않습니다. 다른 곳에서 다루어지는 보안 주제는 다음과 같습니다.

- **물리적 보안:** 서버 시스템의 보안을 물리적으로 유지하기 위한 준비가 없을 경우 소프트웨어 보안은 무의미할 수 있습니다.
- **메시지 저장소 액세스:** Messaging Server에 대한 메시지 저장소 관리자 집합을 정의할 수 있습니다. 이러한 관리자는 메일함을 확인 및 모니터하고 메일함에 대한 액세스를 제어할 수 있습니다. 자세한 내용은 20 장을 참조하십시오.
- **최종 사용자 계정 구성:** 최종 사용자 계정 정보는 주로 Delegated Administrator 제품을 사용하여 관리할 수 있습니다.
- **원하지 않는 대량 전자 메일 필터링(UBE):**18 장을 참조하십시오.
- S/MIME(Secure Multipurpose Internet Mail Extensions)에 대해서는 24 장에서 설명합니다.

다양한 보안 주제를 다루는 매우 많은 문서가 존재합니다. 여기서 언급된 주제에 대한 추가 배경 정보와 다른 보안 관련 정보는 http://docs.sun.com의 설명서 웹 사이트를 참조하십시오.

23.2 HTTP 보안 정보

Messaging Server는 사용자 아이디/비밀번호 인증, 클라이언트 인증서 인증 및 Access Manager를 지원합니다. 그러나 클라이언트와 서버 간의 네트워크 연결을 프로토콜이 처리하는 방법에서 몇 가지 차이점이 있습니다.

POP, IMAP 또는 SMTP 클라이언트가 Messaging Server에 로그인하면 연결이 설정되고 세션이 만들어집니다. 세션이 끝날 때까지, 즉 로그인에서 로그아웃까지 연결이 지속됩니다. 새 연결을 설정할 때 클라이언트는 서버에 대해 재인증되어야 합니다.

HTTP 클라이언트가 Messaging Server에 로그인할 때 서버는 고유한 세션 아이디를 클라이언트에게 제공합니다. 클라이언트는 이 세션 아이디를 사용하여 세션 도중에 여러 연결을 설정합니다. HTTP 클라이언트는 각 연결에 대해 재인증될 필요가 없습니다. 즉, 세션이 해제되고 새 세션을 설정하려는 경우에만 클라이언트 재인증이 필요합니다. (지정된 기간 동안 HTTP 세션이 유휴 상태일 경우 서버가 자동으로 HTTP 세션을 해제하며 클라이언트는 자동으로 로그아웃됩니다. 이 기간의 기본값은 2시간입니다.)

HTTP 세션의 보안을 향상시키기 위해 다음 기술이 사용됩니다.

- 세션 아이디는 특정 IP 주소로 바인드됩니다.
- 각 세션 아이디는 관련된 시간 초과 값을 가집니다. 세션 아이디는 지정된 기간 동안 사용되지 않을 경우 무효화됩니다.
- 서버가 열려 있는 모든 세션 아이디의 데이터베이스를 보유하므로 클라이언트가 아이디를 위조할 수 없습니다.
- 세션 아이디는 URL에 저장되지만 쿠키 파일에는 저장되지 않습니다.

향상된 연결 성능을 위한 구성 매개 변수 지정에 대한 자세한 내용은 5 장을 참조하십시오.

Access Manager에 대한 자세한 내용은 6 장을 참조하십시오.

23.3 인증 기법 구성

인증 기법은 클라이언트가 자신의 아이디를 서버에 대해 입증하는 특정 방법입니다. Messaging Server는 SASL(Simple Authentication and Security Layer) 프로토콜에서 정의되는 인증 방법을 지원하며 인증서 기반 인증을 지원합니다. SASL 기법은 이 절에 설명되어 있습니다. 인증서 기반 인증에 대한 자세한 내용은 686 페이지 “23.5 암호화 및 인증서 기반 인증 구성 ”을 참조하십시오.

Messaging Server는 비밀번호 기반 인증을 위한 다음 SASL 인증 방법을 지원합니다.

- **PLAIN** - 이 기법은 사용자의 일반 비밀번호를 네트워크를 통해 전달하므로 도청에 취약합니다.

  SSL을 사용하여 도청 문제를 줄일 수 있다는 것에 주의합니다. 자세한 내용은 686 페이지 “23.5 암호화 및 인증서 기반 인증 구성 ”을 참조하십시오.
- **DIGEST-MD5** - RFC 2831에 정의된 챌린지/응답 인증 기법입니다. (DIGEST-MD5는 아직 Messaging Multiplexor에서 지원하지 않습니다.)

주 – 이 기능은 더 이상 사용되지 않으며 이후 릴리스에서 제거될 것입니다.

- **CRAM-MD5** - APOP와 비슷하지만 다른 프로토콜에서 사용하기에도 적합한 챌린지/응답 인증 기법입니다. RFC 2195에 정의되어 있습니다.

- **APOP** - POP3 프로토콜에만 사용할 수 있는 챌린지/응답 인증 기법입니다. RFC 1939에 정의되어 있습니다.
- **LOGIN** - PLAIN과 동일하며 SMTP 인증의 예비 표준 구현과의 호환성을 위해서만 존재합니다. 기본적으로 이 기법은 SMTP에 대해서만 사용 가능하게 됩니다.

챌린지/응답 인증 기법을 사용하면 서버는 요청 문자열을 클라이언트에게 보냅니다. 클라이언트는 사용자의 비밀번호와 해당 챌린지의 해시로 응답합니다. 클라이언트의 응답이 서버의 고유한 해시와 일치할 경우 사용자가 인증됩니다. 해시는 취소할 수 없으므로 사용자의 비밀번호가 네트워크를 통해 보내질 때 공개되지 않습니다.

주 – POP, IMAP 및 SMTP 서비스는 모든 SASL 기법을 지원합니다. HTTP 서비스는 일반 텍스트 비밀번호 기법만 지원합니다.

표 23–1은 몇 개의 SALS 및 SASL 관련 `configutil` 매개 변수를 보여 줍니다. `configutil` 매개 변수의 최신 전체 목록은 **Sun Java System Messaging Server 6.3 Administration Reference**의 “configutil Parameters”를 참조하십시오.

표 23–1 일부 SASL 및 SASL 관련 `configutil` 매개 변수

매개 변수	설명
`sasl.default.ldap.has_plain_passwords`	디렉토리에 일반 텍스트 비밀번호가 저장되는지 나타내며 APOP, CRAM-MD5 및 DIGEST-MD5를 사용 가능하게 하는 부울입니다. 기본값: False
sasl.default.transition_criteria	더 이상 지원 또는 사용되지 않습니다. `sasl.default.auto_transition`을 참조하십시오.
sasl.default.auto_transition	부울입니다. 이 매개 변수가 설정되고 사용자가 일반 텍스트 비밀번호를 제공할 경우 비밀번호 저장 형식이 Directory Server에 대한 기본 비밀번호 저장 방법으로 전환됩니다. 일반 텍스트 비밀번호를 APOP, CRAM-MD5 또는 DIGEST-MD5로 마이그레이션하는 데 사용할 수 있습니다. 기본값: False
service.imap.allowanonymouslogin	IMAP에 사용하기 위해 SASL ANONYMOUS 기법을 사용 가능하게 합니다. 기본값: False

표 23–1 일부 SASL 및 SASL 관련 configutil 매개 변수 (계속)

매개 변수	설명
service.{imap|pop|http}.plaintextmincipher	이 매개 변수가 > 0이면 보안 계층(SSL 또는 TLS)이 활성화되지 않은 경우 일반 텍스트 비밀번호를 사용할 수 없게 됩니다. 사용자는 로그인하려면 네트워크상에서 비밀번호가 공개되는 것을 방지하는 SSL 또는 TLS를 클라이언트에서 사용 가능하게 해야 합니다. MMP는 동일한 옵션 "RestrictPlainPasswords"를 가집니다. 주의: Messaging Server의 5.2 릴리스는 SSL 또는 TLS에 의해 협상된 암호문 강도에 대해 실제로 값을 검사합니다. 이 옵션을 단순화하고 일반적인 사용을 더 잘 반영하기 위해 이 기능이 제거되었습니다. 기본값: 0
sasl.default.mech_list	사용 가능하게 할 SASL 기법의 공백으로 구분된 목록입니다. 비어 있지 않을 경우 이 옵션은 sasl.default.ldap.has_plain_passwords 및 service.imap.allowanonymouslogin 옵션을 모두 무시합니다. 이 옵션은 모든 프로토콜(imap, pop, smtp)에 적용됩니다. 기본값: False
sasl.default.ldap.searchfilter	도메인의 inetDomainSearchFilter에 지정되지 않은 경우 사용자를 조회하는 데 사용되는 기본 검색 필터입니다. 구문은 inetDomainSearchFilter(스키마 설명서 참조)와 동일합니다. 기본값: (&(uid=%U)(objectclass=inetmailuser))
sasl.default.ldap.searchfordomain	기본적으로 인증 시스템은 도메인 조회 규칙에 따라(즉, 필요에 따라) LDAP에서 도메인을 조회한 다음 사용자를 조회합니다. 그러나 이 옵션이 기본값 "1"이 아니라 "0"으로 설정된 경우 도메인 조회는 수행되지 않으며 sasl.default.ldap.searchfilter를 사용한 사용자 검색이 local.ugldapbasedn에 지정된 LDAP 트리에서 직접 수행됩니다. 이것은 레거시 단일 도메인 스키마와의 호환성을 위해 제공되지만 심지어 소규모 회사에서도 여러 도메인에 대한 지원이 필요한 합병이나 사명 변경이 발생할 수 있으므로 새 배포에는 사용하지 않는 것이 좋습니다.

23.3.1 일반 텍스트 비밀번호에 대한 액세스 구성

CRAM-MD5, DIGEST-MD5 또는 APOP SASL 인증 방법을 사용하려면 사용자의 일반 텍스트 비밀번호에 대한 액세스가 필요합니다. 다음 단계를 수행해야 합니다.

1. 비밀번호를 일반 텍스트로 저장하도록 Directory Server를 구성합니다.
2. Directory Server가 일반 텍스트 비밀번호를 사용한다는 것을 인식하도록 Messaging Server를 구성합니다.

▼ Directory Server를 구성하여 일반 텍스트 비밀번호를 저장하는 방법

CRAM-MD5, DIGEST-MD5 또는 APOP 기법을 사용하려면 Directory Server를 구성하여 비밀번호를 일반 텍스트로 저장하게 해야 합니다. Directory Server 6 이전 버전을

사용하는 경우 다음 지침이 적용됩니다. 6 이상 버전의 경우 최신 Directory Server 설명서(**Sun Java System Directory Server Enterprise Edition 6.0 Administration Guide**)를 :

1. **Directory Server 콘솔에서 구성할 Directory Server를 엽니다.**
2. **구성 탭을 누릅니다.**
3. **왼쪽 표시 영역에서 데이터를 엽니다.**
4. **오른쪽 표시 영역에서 비밀번호를 누릅니다.**
5. **비밀번호 암호화 드롭다운 목록에서 "일반 텍스트"를 선택합니다.**

주 – 이 변경 사항은 앞으로 만들 사용자에만 영향을 줍니다. 이 변경 이후에 기존 사용자는 자신의 비밀번호를 전환하거나 재설정해야 합니다.

23.3.1.1 Messaging Server에 일반 텍스트 비밀번호 구성

이제 Messaging Server를 구성하여 Directory Server가 일반 텍스트 비밀번호를 검색할 수 있다는 것을 인식하도록 할 수 있습니다. 이렇게 하면 Messaging Server는 APOP, CRAM-MD5 및 DIGEST-MD5를 안전하게 광고할 수 있습니다.

```
configutil -o sasl.default.ldap.has_plain_passwords -v 1
```

값을 0으로 설정하여 이러한 챌린지/응답 SASL 기법을 사용 불가능하게 할 수 있습니다.

주 – 기존 사용자는 비밀번호를 재설정하거나 마이그레이션할 때까지 APOP, CRAM-MD5 또는 DIGEST-MD5를 사용할 수 없습니다(사용자 전환 참조).

MMP는 CRAM과 동등한 옵션을 가집니다.

23.3.2 사용자 전환

`configutil`을 사용하여 사용자 전환에 대한 정보를 지정할 수 있습니다. 적절한 항목을 갖고 있지 않은 기법으로 클라이언트가 인증을 시도하거나 사용자 비밀번호가 변경되는 경우를 예로 들 수 있습니다.

```
configutil -o sasl.default.auto_transition -v value
```

value의 경우 다음 중 하나를 지정할 수 있습니다.

- `no` 또는 `0` - 비밀번호를 전환하지 않습니다. 기본값입니다.
- `yes` 또는 `1` - 비밀번호를 전환합니다.

사용자를 성공적으로 전환하려면 사용자 비밀번호 속성에 대한 쓰기 권한을 Messaging Server에 허용하는 ACI를 Directory Server에서 설정해야 합니다. 이렇게 하려면 다음 단계를 수행합니다.

▼ 사용자 전환

. Directory Server 6 이전 버전을 사용하는 경우 다음 지침이 적용됩니다. 6 이상 버전의 경우 최신 Directory Server 설명서(**Sun Java System Directory Server Enterprise Edition 6.0 Administration Guide**)를

1. **콘솔에서 구성할 Directory Server를 엽니다.**
2. **디렉토리 탭을 누릅니다.**
3. **사용자/그룹 트리의 기본 접미어를 선택합니다.**
4. **객체 메뉴에서 액세스 권한을 선택합니다.**
5. **"Messaging Server 최종 사용자 관리자 쓰기 액세스 권한"에 대한 ACI를 선택(두 번 누름)합니다.**
6. **ACI 속성을 누릅니다.**
7. **`userpassword` 속성을 기존 속성 목록에 추가합니다.**
8. **확인을 누릅니다.**

   `sasl.default.mech_list`를 사용하여 SASL 기법의 목록을 사용 가능하게 할 수 있습니다. 비어 있지 않을 경우 이 옵션은 `sasl.default.ldap.has_plain_passwords` 및 `service.imap.allowanonymouslogin` 옵션을 모두 무시합니다. 이 옵션은 모든 프로토콜(imap, pop, smtp)에 적용됩니다.

23.4 사용자 비밀번호 로그인

Messaging Server에 로그인하여 메일을 주고 받으려는 사용자에게 비밀번호 제출을 요구하는 것은 무단 액세스를 방지하는 첫 번째 방법입니다. Messaging Server는 IMAP, POP, HTTP 및 SMTP 서비스에 대한 비밀번호 기반 로그인을 지원합니다.

23.4.1 IMAP, POP 및 HTTP 비밀번호 로그인

기본적으로 내부 사용자는 Messaging Server에서 메시지를 검색하기 위해 비밀번호를 제출해야 합니다. 관리자는 POP, IMAP 및 HTTP 서비스에 대한 비밀번호 로그인을 별개로 사용 가능 또는 불가능하게 합니다. POP, IMAP 및 HTTP 서비스의 비밀번호 로그인에 대한 자세한 내용은 121 페이지 “5.2.2 비밀번호 기반 로그인”을 참조하십시오.

사용자 비밀번호는 사용자의 클라이언트 소프트웨어에서 서버로 일반 텍스트 또는 암호화된 형식으로 전송할 수 있습니다. 클라이언트와 서버가 둘 다 SSL을 사용 가능하게 구성되고 697 페이지 "23.5.2 SSL 사용 및 암호문 선택"에 설명된 것처럼 필요한 강도의 암호화를 지원할 경우 암호화가 수행됩니다.

사용자 아이디와 비밀번호는 설치 시의 LDAP 사용자 디렉토리에 저장됩니다. 최소 길이와 같은 비밀번호 보안 조건은 디렉토리 정책 요구 사항에 의해 결정되며 Messaging Server 관리의 일부가 아닙니다.

인증서 기반 로그인은 비밀번호 기반 로그인의 대안입니다. 이 장에서 SSL의 나머지 내용을 다루면서 인증서 기반 로그인이 설명됩니다. 699 페이지 "23.5.3 인증서 기반 로그인 설정"을 참조하십시오.

챌린지/응답 SASL 기법은 일반 텍스트 비밀번호 로그인의 또 다른 대안입니다.

23.4.2 SMTP 비밀번호 로그인

기본적으로 사용자는 메시지를 보내기 위해 Messaging Server의 SMTP 서비스에 연결할 때 비밀번호를 제출할 필요가 없습니다. 그러나 관리자는 인증된 SMTP를 사용할 수 있도록 SMTP에 대한 비밀번호 로그인을 사용 가능하게 할 수 있습니다.

인증된 *SMTP*는 클라이언트를 서버에 대해 인증할 수 있는 SMTP 프로토콜의 확장입니다. 이 인증에는 메시지가 수반됩니다. 인증된 SMTP는 주로 이동 중이거나 홈 ISP를 사용하는 로컬 사용자가 열린 중계(다른 사용자가 남용할 수 있는)를 만들지 않고 메일을 전달하도록 허용하기 위해 사용됩니다. 클라이언트는 서버에 대해 인증되도록 "AUTH" 명령을 사용합니다.

SMTP 비밀번호 로그인과 이에 따라 인증된 SMTP를 사용 가능하게 하는 방법은 349 페이지 "12.4.4 SMTP 인증, SASL 및 TLS"를 참조하십시오.

SSL 암호화를 함께 사용하거나 사용하지 않으면서 인증된 SMTP를 사용할 수 있습니다.

23.5 암호화 및 인증서 기반 인증 구성

이 절에는 다음과 같은 하위 절이 포함됩니다.

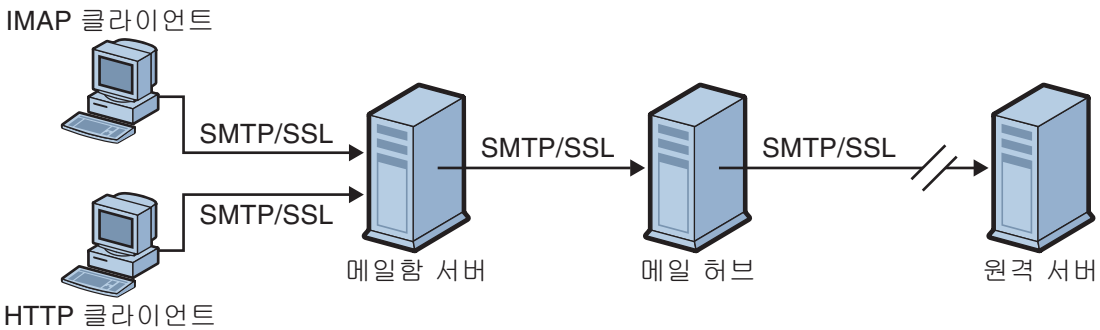
Messaging Server는 클라이언트 및 서버의 암호화된 통신과 인증서 기반 인증을 위해 TLS(Transport Layer Security) 프로토콜(또는 SSL(Secure Sockets Layer) 프로토콜로 알려져 있음)을 사용합니다. Messaging Server는 SSL 버전 3.0 및 3.1을 지원합니다. TLS는 SSL과 완전히 호환되며 필요한 모든 SSL 기능을 포함합니다.

SSL에 대한 배경 정보는 **Managing Servers With iPlanet Console 5.0**의 *Introduction to SSL*을 참조하십시오. SSL은 **Managing Servers With iPlanet Console 5.0**의 *Introduction to Public-Key Cryptography*에 설명되어 있는 공개 키 암호화의 개념에 기초합니다.

Messaging Server와 클라이언트 간의 메시지나 Messaging Server와 다른 서버 간의 메시지 전송이 암호화될 경우 통신에서 도청이 발생할 가능성이 거의 없습니다. 또한 연결하는 클라이언트가 인증될 경우 침입자가 클라이언트를 가장(스푸핑)할 가능성이 거의 없습니다.

SSL은 IMAP4, HTTP, POP3 및 SMTP의 응용 프로그램 계층 아래에 있는 프로토콜 계층의 기능을 수행합니다. SMTP 및 SMTP/SSL은 같은 포트를 사용하고 HTTP 및 HTTP/SSL에는 다른 포트가 필요하며 IMAP 및 IMAP/SSL과 POP 및 POP/SSL은 같은 포트나 다른 포트를 사용할 수 있습니다. 그림 23–1에 나온 것처럼 SSL은 보내는 메시지와 받는 메시지 모두에 대해 특정 메시지 통신 단계에서 작동합니다.

A. 보내는 메시지



B. 받는 메시지

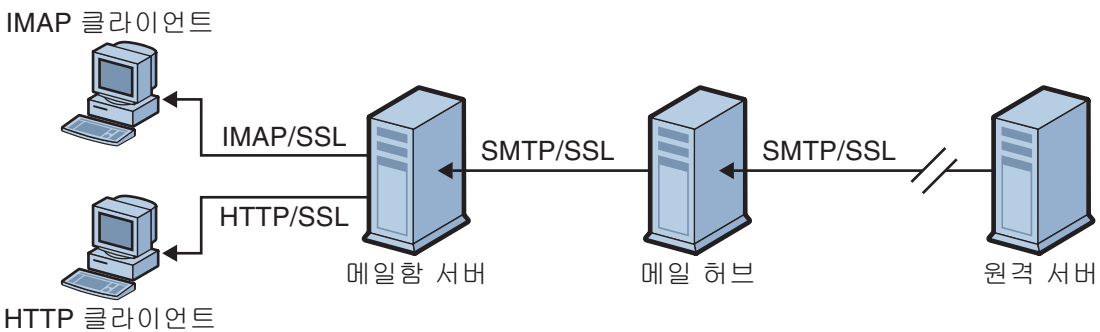


그림 23–1 Messaging Server와의 암호화된 통신

SSL은 홉 간의 암호화를 제공하지만 메시지는 각 중간 서버에서 암호화되지 않습니다.

주 – 보내는 메시지 대한 암호화를 사용하려면 `maytls`, `musttls` 등의 `tls` 채널 키워드를 포함하도록 채널 정의를 수정해야 합니다. 자세한 내용은 351 페이지 "12.4.8 전송 계층 보안"을 참조하십시오.

SSL 연결을 설정하는 과정에서 발생하는 추가 오버헤드가 서버의 성능에 부담을 줄 수 있다는 것에 주의합니다. 따라서 메시징 설치를 디자인하고 성능을 분석할 경우 서버 용량과 보안 요구 사항 간에 적절히 균형을 맞추는 것이 필요합니다.

23.5.1 인증서 얻기

SSL을 암호화에 사용하는지 아니면 인증에 사용하는지 여부에 상관 없이 Messaging Server에 대한 서버 인증서를 얻어야 합니다. 인증서는 해당 서버를 클라이언트와 다른 서버에 대해 식별합니다. 이 절의 뒷 부분에서 설명하는 대로 `msgcert` 명령을 사용하여 인증서를 얻는 방법이 가장 효과적입니다. 이전 `certutil` 명령을 계속 사용할 수는 있지만 훨씬 더 복잡하고 국제화되지 않았습니다. `certutil`에 대한 자세한 내용은 686 페이지 "23.5 암호화 및 인증서 기반 인증 구성" 및 http://www.mozilla.org/projects/security/pki/nss/tools/certutil.html을 참조하십시오.

이 절은 다음과 같은 하위 절로 구성되어 있습니다.

23.5.1.1 내부 및 외부 모듈 관리

서버 인증서는 데이터를 암호화 및 해독하는 데 사용되는 숫자인 키 쌍의 소유권과 유효성을 설정합니다. 서버 인증서와 키 쌍은 서버의 신원을 나타내며 서버 내부이거나 외부의 이동식 하드웨어 카드(스마트 카드)가 될 수 있는 인증서 데이터베이스에 저장됩니다.

Sun Java System 서버는 PKCS(Public-Key Cryptography System) #11 API를 따르는 모듈을 사용하여 키 및 인증서 데이터베이스에 액세스합니다. 특정 하드웨어 장치의 PKCS #11 모듈은 일반적으로 해당 제공자로부터 얻을 수 있으며 Messaging Server에 설치한 후에만 Messaging Server에서 해당 장치를 사용할 수 있습니다. 미리 설치된 "Netscape Internal PKCS # 11 Module"은 서버 내부의 인증서 데이터베이스를 사용하는 단일 내부 소프트웨어 토큰을 지원합니다.

인증서 사용을 위해 서버를 설정하는 작업에는 인증서와 해당 키를 위한 데이터베이스를 만들고 PKCS #11 모듈을 설치하는 것이 포함됩니다. 외부 하드웨어 토큰을 사용하지 않을 경우 서버에서 내부 데이터베이스를 만들고 Messaging Server의 일부인 내부 기본 모듈을 사용합니다. 외부 토큰을 사용할 경우 하드웨어 스마트 카드 판독기를 연결하고 해당 PKCS #11 모듈을 설치합니다.

주 – 다음 절에서는 콘솔 또는 Directory Server 콘솔에 대해 설명합니다. 이 명칭은 Directory Server 6 이전 버전에 사용되는 용어입니다. 6 이상 버전에서는 그래픽 사용자 인터페이스를 Directory Server Control Center라고 합니다. 자세한 내용은 최신 Directory Server 설명서(**Sun Java System Directory Server Enterprise Edition 6.0 Administration Guide**)를 참조하십시오.

내부 및 외부 PKCS #11 모듈을 모두 콘솔을 통해 관리할 수 있습니다. PKCS #11 모듈을 설치하려면 다음을 수행합니다.

1. 하드웨어 카드 판독기를 Messaging Server 호스트 시스템에 연결하고 드라이버를 설치합니다.
2. `msg-svr-base/sbin`에 있는 modutil을 사용하여 설치된 드라이버용 PKCS #11 모듈을 설치합니다.

하드웨어 암호화 가속기 설치. 암호화를 위해 SSL을 사용할 경우 하드웨어 암호화 가속기를 설치하여 메시지를 암호화 및 해독하는 서버의 성능을 향상시킬 수 있습니다. 일반적으로 암호화 가속기는 서버 시스템에 영구적으로 설치된 하드웨어 보드와 소프트웨어 드라이버로 구성됩니다. Messaging Server는 PKCS #11 API를 따르는 가속기 모듈을 지원합니다. (이러한 모듈은 기본적으로 고유한 키를 저장하지 않으며 이를 위한 내부 데이터베이스를 사용하는 하드웨어 토큰입니다.) 가속기를 설치하려면 우선 하드웨어와 드라이버를 제조업체에서 지정한 대로 설치한 다음 하드웨어 인증서 토큰과 마찬가지로 PKCS #11 모듈을 설치하여 설치를 완료합니다.

23.5.1.2 비밀번호 파일 만들기

SSL이 사용 가능하게 되는 대부분의 Sun Java System 서버에서는 키 쌍을 해독하는 데 필요한 비밀번호를 제공하라는 메시지가 시작 시에 관리자에게 표시됩니다. 그러나 Messaging Server에서는 비밀번호를 여러 번 입력(최소한 세 개의 서버 프로세스에 필요함)하는 불편함을 없애고 무인 서버의 다시 시작을 용이하게 만들기 위해 비밀번호를 비밀번호 파일에서 읽습니다. 비밀번호는 `msgcert generate_certdb` 명령을 사용하여 인증서 데이터베이스를 만들 때 생성됩니다.

비밀번호 파일의 이름은 `sslpassword.conf`이며 *msg-svr-base*`/config/` 디렉토리에 위치합니다. 이 파일의 항목은 다음 형식을 갖는 개별 행입니다.

```
moduleName:password
```

여기에서 *moduleName*은 사용할 내부 또는 외부 PKCS #11 모듈의 이름이며 *password*는 모듈의 키 쌍을 해독하는 비밀번호입니다. 비밀번호는 일반(암호화되지 않은) 텍스트로 저장됩니다.

Messaging Server는 내부 모듈 및 기본 비밀번호를 위한 다음과 같은 단일 항목을 가지는 기본 버전의 비밀번호 파일을 제공합니다.

```
Internal (Software) Token:netscape!
```

내부 인증서를 설치할 때 기본값이 아닌 비밀번호를 지정할 경우 비밀번호 파일의 위 행을 편집하여 지정된 비밀번호를 반영해야 합니다. 외부 모듈을 설치할 경우 이에 대해 지정한 모듈 이름과 비밀번호를 포함하는 새 행을 파일에 추가해야 합니다.

주의 – 서버 시작 시에 모듈 비밀번호를 묻는 메시지가 관리자에게 표시되지 않으므로 서버에 대한 관리자 액세스 제어와 서버 호스트 시스템 및 해당 백업의 물리적 보안을 적절하게 하는 것이 특히 중요합니다.

23.5.1.3 인증서 얻기 및 관리

SSL을 암호화에 사용하는지 아니면 인증에 사용하는지 여부에 상관 없이 Messaging Server에 대한 서버 인증서를 얻어야 합니다. 인증서는 해당 서버를 클라이언트와 다른 서버에 대해 식별합니다. 인증서 얻기 및 관리를 위한 기본 기법은 `msgcert`를 사용하는 것입니다. Administration Server를 설치한 경우에는 관리 콘솔을 사용할 수도 있습니다.

이 절의 나머지 부분에서는 `msgcert` 사용 방법에 대해 설명합니다.

23.5.1.4 `msgcert` 정보

`msgcert`를 사용하면 인증서 요청 생성, 인증서 데이터베이스에 인증서 추가, 데이터베이스에 있는 인증서 나열 등과 같은 작업을 수행할 수 있습니다. 자세한 내용을 보려면 명령줄에 다음을 입력하십시오.

msg-svr-base`/sbin/msgcert --help`

그러면 아래와 같이 표시됩니다.

```
# ./msgcert --help

Usage: msgcert SUBCMD [GLOBAL_OPTS] [SUBCMD_OPTS] [SUBCMD_OPERANDS]
Manages the Messaging Servers Certificate Database
The accepted values for SUBCMD are:

add-cert              Adds a certificate to the certificate database
add-selfsign-cert     Creates and adds a selfsign certificate to the
                      certificate database
export-cert           Exports a certificate and its keys from the database
```

```
generate-certDB       Creates Messaging Server Databases cert8.db key3.db
                      secmod.db and sslPassword
import-cert           Adds a new certificate and its keys to the cert database
import-selfsign-cert  Adds a new selfsign certificate and its keys to the
                      cert database
list-certs            Lists all certificates in the Certificate database
remove-cert           Removes a certificate from the database
renew-cert            Renews a certificate
renew-selfsign-cert   Renews a selfsign certificate
request-cert          Generates a certificate request
show-cert             Displays a certificate

The accepted value for GLOBAL_OPTS is:-?, --help
                Displays SUBCMD help

NOTE: You must stop all the TLS or SSL-enabled servers before making any
changes to the Certificate Database.
```

위에 표시된 각 하위 명령은 특정 인증서 관리 기능을 수행합니다. 다음을 입력하면 이러한 하위 명령과 해당 기능에 대한 자세한 내용을 볼 수 있습니다.

`msgcert` *SUBCMD* `–help`

이 절의 나머지 부분에서는 일부 공통된 인증서 관리 절차에 대해 설명합니다.

23.5.1.5 인증서 관리

이 절에서는 Messaging Server에서 SSL 인증서를 관리하는 방법에 대해 설명합니다. Messaging Server에서 SSL을 실행하려면 외부 인증 기관(CA)을 포함하는 PKI(Public Key Infrastructure) 솔루션이나 자체 서명된 인증서를 사용해야 합니다. PKI 솔루션의 경우 공개 키와 개인 키를 모두 포함하는 CA 서명된 서버 인증서가 필요합니다. 이 인증서는 하나의 Messaging Server에만 한정됩니다. 또한 공개 키가 포함된 신뢰할 수 있는 CA 인증서가 필요합니다. 신뢰할 수 있는 CA 인증서가 있으면 해당 CA의 모든 서버 인증서가 신뢰됩니다. 이 인증서를 CA 루트 키 또는 루트 인증서라고도 합니다.

인증서 데이터베이스 비밀번호 구성

인증서를 관리할 때 인증서 비밀번호를 입력하거나 비밀번호 파일을 지정할 필요는 없습니다. 비밀번호를 -W 인수로 간단히 전달할 수 있습니다. 예:

```
echo "password22" > /tmp/certdbpwd
echo "password22" > /tmp/certdbpwd
# ./msgcert list-certs -W /tmp/certdbpwd
```

▼ 기본 자체 서명된 인증서를 사용하여 Messaging Server 인증서 데이터베이스 만들기

1 Messaging Server 인증서 데이터베이스를 만들려면 다음 명령을 실행합니다.

```
msgcert generate-certDB
```

이 명령은 CERT_PW_FILE에서 인증서 데이터베이스 비밀번호를 읽습니다(기본값: 비밀번호 요청).

2 다음 명령을 사용하여 이 인증서를 볼 수 있습니다.

```
msgcert show-cert Server-Cert
```

▼ 자체 서명된 인증서 관리

테스트를 위해 인증서를 사용할 경우 자체 서명된 인증서를 사용할 수 있습니다. 배포 구성에서는 신뢰할 수 있는 인증 기관(CA) 인증서를 사용할 수 있습니다. Directory Server 관리 콘솔을 사용하여 이 작업을 수행할 수도 있습니다.

1 인증서 데이터베이스를 만들 때 기본 자체 서명된 인증서가 자동으로 제공됩니다. 자체 서명된 인증서를 기본값이 아닌 설정으로 사용하려면 msgcert add-selfsign-cert 명령을 사용합니다. 예:

```
msgcert add-selfsign-cert --name siroe --org comms --org-unit Messaging
--city SantaClara --state ca --country us MySelfSigned-Cert
```

자체 서명된 인증서는 3개월 동안 유효합니다.

2 자체 서명된 인증서가 만료되면 다음 명령을 사용하여 인증서를 갱신합니다.

```
msgcert renew-selfsign-cert cert_alias
```

23.5.1.6 신뢰할 수 있는 CA의 인증서 설치

./msgcert add-cert를 사용하여 인증 기관의 인증서를 설치합니다. CA 인증서는 CA 자체의 신원을 검증합니다. 서버는 클라이언트 및 다른 서버를 인증하는 과정에서 이러한 CA 인증서를 사용합니다.

예를 들어, 비밀번호 기반 인증 외에 인증서 기반 클라이언트 인증이 가능하도록 설정한 경우(157페이지의 "인증서 기반 로그인 설정" 참조) 클라이언트가 제공할 수 있는 인증서를 발급하는 신뢰할 수 있는 모든 CA의 CA 인증서를 설치해야 합니다. 이러한 CA는 조직 내부에 대한 것이거나 민간 또는 정부 기관이나 다른 회사 등 외부에 대한 것일 수 있습니다. 인증을 위한 CA 인증서 사용에 대한 자세한 내용은 **Managing Servers With iPlanet Console 5.0**에서 *Introduction to Public-Key Cryptography*를 참조하십시오.

Messaging Server를 설치하면 여러 상용 CA에 대한 CA 인증서가 기본적으로 포함되어 있습니다. 다른 상용 CA를 추가해야 하거나 회사에서 내부 용도의 고유한 CA를 개발(Sun Java System Certificate Server 사용)하는 중이면 추가 CA 인증서를 얻어 설치해야 합니다.

주 – Messaging Server에서 자동으로 제공되는 CA 인증서는 처음에 클라이언트 인증서에 대해 신뢰할 수 있는 것으로 표시되지 않습니다. 따라서 이러한 CA에 의해 발급된 클라이언트 인증서를 신뢰하려면 트러스트 설정을 편집해야 합니다. 자세한 내용은 688 페이지 "23.5.1 인증서 얻기"를 참조하십시오.

다음 절차에서는 Messaging Server에서 사용할 CA 서명된 서버 및 신뢰할 수 있는 CA 인증서를 요청하고 설치하는 과정에 대해 설명합니다.

▼ CA 서명된 서버 인증서 요청

Directory Server 관리 콘솔을 사용하여 이 작업을 수행할 수도 있습니다.

1 CA 서명된 서버 인증서 요청을 생성합니다.

```
msgcert request-cert [-W CERT_PW_FILE] {-S DN|--name NAME [--org ORG] [--org-unit ORG-UNIT]
   [--city CITY] [--state STATE] [--country COUNTRY] } [-F FORMAT] [-o OUTPUT_FILE]
```

다음은 CA 서명된 서버 인증서 요청의 예입니다. 이 예에서는 이진 형식의 인증서를 반환합니다.

```
./msgcert request-cert --name aqua --org siroe --org-unit Messaging -o my_ca_signed_request_cert
```

ASCII 형식의 인증서를 반환하려면 명령을 다음과 같이 사용합니다.

```
./msgcert request-cert --name aqua --org siroe --org-unit Messaging -F ascii -o my_casigned_request_cert
```

인증 기관에서는 서버를 완벽하게 식별하기 위해 일반적으로 이 예에 표시된 모든 속성을 요구합니다. 각 속성에 대한 설명을 보려면 `./msgcert request-cert --help`를 입력합니다. `msgcert request-cert`를 사용하여 인증서를 요청하는 경우 ASCII를 출력 형식으로 지정하지 않는 한 결과 인증서 요청은 이진 인증서 요청입니다. ASCII를 지정한 경우 결과 인증서 요청은 PEM 형식의 PKCS #10 인증서 요청입니다. PEM은 RFC 1421-1424에 지정되고 base64 인코딩된 인증서 요청을 ASCII 문자로 표시하는 데 사용되는 Privacy Enhanced Mail 형식입니다. 요청 내용은 다음 예와 비슷합니다.

```
-----BEGIN NEW CERTIFICATE REQUEST-----
MIIBdTCB3wIBADA2MRIwEAYDVQQLEwlNZXNzYWdpbmcxDjAMBgNVBAoTBXNpcm9l
MRAwDgYDVQQDEwdhcXVhdGljMIGfMA0GCSqGSIb3DQEBAQUAA4GNADCBiQKBgQDt
KEh5Fnj/h9GEu18Da6DkJpcNShkwxanjnKs2883ZoUV5Sp4pN7U6Vfbh0414WXZh
D26m3t81q9b9h47Klkf0pW1X3BB6LOjGOHSt2VoNBI8n3hJ6XiN2zYbrlLTgdKuo
y0YrSG/kHFnqKghikag9O/Ox+cwD+mpjl2QnsPZgswIDAQABoAAwDQYJKoZIhvcN
AQEEBQADgYEArqgWQIwNZDC2d3EZawI23Wj9o6Pyvu9J1rkb+NYgIEnNp9jugxqX
F326N0ABLdHXXNX/2ZvC5TKOgS4RidTBM89N9xJvokmvRGfc+1x80uxy474YdNlZ
s+nP8AYo9dW9mrLOammozx9HLPSVYNFp4FxekgV2n8QG7WC5rkN5bCE=
-----END NEW CERTIFICATE REQUEST-----
```

2 절차에 따라 인증서 요청을 인증 기관에 전송합니다.

인증 기관 인증서를 얻는 프로세스는 사용하는 인증 기관에 따라 다릅니다. 일부 상업용 CA는 인증서를 자동으로 다운로드할 수 있는 웹 사이트를 제공합니다. 그 외의 CA는 요청 시 인증서를 전자 메일로 전송합니다.

요청을 보낸 후 CA에서 인증서로 응답할 때까지 기다려야 합니다. 요청에 대한 응답 시간은 각각 다릅니다. 예를 들어, CA가 회사 내부에 있는 경우 CA가 요청에 응답하는 데 1-2일이 걸릴 수 있습니다. 선택한 CA가 회사 외부에 있는 경우 CA가 요청에 응답하는 데 몇 주일이 걸릴 수도 있습니다.

3 인증 기관으로부터 수신한 인증서를 저장합니다.

인증서를 안전한 위치에 백업해 두어야 합니다. 인증서가 손실될 경우 백업 파일을 사용하여 인증서를 다시 설치할 수 있습니다. 인증서를 텍스트 파일로 저장할 수 있습니다. PEM 형식의 PKCS #11 인증서는 다음 예와 비슷합니다.

```
-----BEGIN CERTIFICATE-----
MIICjCCAZugAwIBAgICCEEwDQYJKoZIhKqvcNAQFBQAwfDELMAkGA1UEBhMCVVMx
IzAhBgNVBAoGlBhbG9a2FWaWxsZGwSBXaWRnZXRzLCBJbmMuMR0wGwYDVQQLExRX
aWRnZXQgTW3FrZXJzICdSJyBVczEpMCcGAx1UEAxgVGVzdCBUXN0IFRlc3QgVGVz
dCBUZXN0IFlc3QgQ0EswHhcNOTgwMzEyMDIzMzUWhcNOTgwMzI2MDIzMpzU3WjBP
MQswCYDDVQQGEwJVUzEoMCYGA1UEChMfTmV0c2NhcGUgRGlyZN0b3J5VIFB1Ymxp
Y2F0aW9uczEWMB4QGA1UEAxMNZHVgh49dq2tLNvbjTBaMA0GCSqGSIb3DQEBAQUA
A0kAMEYkCQCksMR/aLGdfp4m0OiGgijG5KgOsyRNvwGYW7kfW+8mmijDtZaRjYNj
jcgpF3VnlbxbclX9LVjjNLC5737XZdAgEDozYwpNDARBglghkgBhvhCEAQEEBAMC
APAwHkwYDVR0jBBgwFAU67URjwCaGqZHUpSpdLxlzwJKiMwDQYJKoZIhQvcNAQEF
BQADgYEAJ+BfVem3vBOPBveNdLGfjlb9hucgmaMcQa9FA/db8qimKT/ue9UGOJqL
bwbMKBBopsDn56p2yV3PLIsBgrcuSoBCuFFnxBnqSiTS7YiYgCWqWaUA0ExJFmD6
6hBLseqkSWulk+hXHN7L/NrViO+7zNtKcaZLlFPf7d7j2MgX4Bo=
-----END CERTIFICATE-----
```

▼ CA 서명된 서버 인증서 및 신뢰할 수 있는 CA 인증서 추가

Directory Server 관리 콘솔을 사용하여 이 작업을 수행할 수도 있습니다.

1 다음 명령을 사용하여 CA 서명된 서버 인증서를 추가합니다.

```
msgcert add-cert cert_alias cert_file
```

여기서 *cert_alias*는 인증서를 식별하기 위해 제공하는 이름이고, *cert_file*은 PEM 형식의 PKCS #11 인증서를 포함하는 텍스트 파일입니다.

예를 들어, CA 서명된 서버 인증서를 설치하려면 다음과 비슷한 명령을 사용할 수 있습니다.

```
msgcert add-cert /my_cert/server-cert-file
```

이제 인증서가 설치되었지만 아직 신뢰되지는 않습니다. CA 서명된 서버 인증서를 신뢰하려면 인증 기관 인증서를 설치해야 합니다.

2 다음 명령을 사용하여 신뢰할 수 있는 인증 기관 인증서를 추가합니다.

```
msgcert add-cert -C cert_alias cert_file
```

-C 옵션은 인증서가 신뢰할 수 있는 인증 기관 인증서임을 나타냅니다.

예를 들어, 인증 기관의 신뢰할 수 있는 인증서를 설치하려면 다음 명령을 사용할 수 있습니다.

```
msgcert add-cert -C CA-cert /my_cert/ca-cert-file
```

3 선택적으로 다음 명령을 사용하여 설치한 인증서를 확인합니다.

모든 서버 인증서를 나열하여 별칭 및 유효 날짜와 같은 정보를 표시하려면 다음 명령을 사용합니다.

```
msgcert list-certs
```

Messaging Server에는 ./msgcert generate-CertDB를 통해 생성된 경우 Server-Cert라는 기본 인증서가 있습니다. 텍스트 Same as issuer는 기본 인증서가 자체 서명된 서버 인증서임을 나타냅니다. 예를 들면 다음과 같습니다.

```
# ./msgcert list-certs
Enter the certificate database password:
Alias          Valid from       Expires on       Self-   Issued by             Issued to
                                                 signed
------------   ---------------  --------------- ------  --------------------- ------------------------  ------
SelfSignedCrt 2006/07/28 12:58  2006/10/28 12:58   y     CN=SFO,L=SC,ST=ca,C=us  Same as issuer
Server-Cert   2006/07/28 07:47  2006/10/28 07:47   y     CN=perseids             Same as issuer
2 certificates found
```

신뢰할 수 있는 CA 인증서를 나열하려면 다음 명령을 사용합니다.

```
msgcert list-certs -C
```

인증서 만료 날짜를 포함하여 인증서 세부 정보를 보려면 다음 명령을 사용합니다.

```
msgcert show-cert cert_alias
```

예를 들어, 자체 서명된 인증서를 표시하려면 다음 명령을 사용합니다.

```
# ./msgcert show-cert MySelfSigned-Cert
Enter the certificate database password:
Certificate:
    Data:
        Version: 3 (0x2)
        Serial Number:
            00:83:35:37:94
        Signature Algorithm: PKCS #1 MD5 With RSA Encryption
        Issuer:
            "CN=siroe,O=comms,OU=Messaging,L=SantaClara,ST=ca,C=us"
        Validity:
```

```
            Not Before: Fri Jul 28 19:58:31 2006
            Not After : Sat Oct 28 19:58:31 2006
        Subject:
            "CN=siroe,O=comms,OU=Messaging,L=SantaClara,ST=ca,C=us"
        Subject Public Key Info:
            Public Key Algorithm: PKCS #1 RSA Encryption
            RSA Public Key:
                Modulus:
                    aa:9d:3d:23:b2:59:39:f3:77:c8:69:7f:b0:d1:ac:d2:
                    4e:81:c8:51:0f:27:6f:a1:21:4b:a9:27:46:d7:0f:b4:
                    c8:44:86:32:5e:4f:2f:1c:2f:a9:b8:a3:49:b5:b8:ab:
                    51:a8:a5:ba:1c:e8:90:7d:46:67:f9:a7:44:c5:1d:24:
                    e6:bd:e8:8f:07:b4:5a:68:41:b1:19:f2:ea:98:ba:25:
                    55:b8:ba:9c:af:bb:43:c3:c0:8f:14:a7:4c:2b:50:b4:
                    ac:df:b5:cd:68:de:a6:14:9d:68:77:d3:8b:7f:de:c0:
                    5d:35:d7:55:8d:b5:c3:14:2a:60:a9:bf:de:96:90:a9
                Exponent: 65537 (0x10001)
    Signature Algorithm: PKCS #1 MD5 With RSA Encryption
    Signature:
        15:86:f1:cc:85:c9:08:0f:ff:d3:56:d8:e2:c8:ea:3c:
        8e:45:36:be:8b:b0:7d:2f:e9:cd:e3:b4:ad:8c:70:59:
        c8:a5:14:da:9c:fa:7f:70:86:64:34:0b:21:ae:c4:28:
        d2:f5:94:5c:a6:78:0f:d9:fd:fc:c5:5e:37:49:25:a9:
        bc:12:59:cb:fb:4e:e9:d4:8a:8d:3d:41:12:ae:f1:7f:
        8d:d3:10:ac:fb:33:51:5d:0c:1b:dc:23:5f:95:d5:6d:
        c6:1d:e5:ed:13:8b:16:41:89:5b:4d:de:c0:c7:56:a2:
        48:82:38:32:5a:99:d5:21:20:c5:0d:5c:ea:0c:84:aa
    Fingerprint (MD5):
        EF:76:A3:6C:09:4E:BC:6B:87:76:A3:35:70:1F:B2:C4
    Fingerprint (SHA1):
        BB:1C:20:4B:79:3A:F1:49:F0:83:FB:CC:9C:56:10:D3:06:97:AA:07

    Certificate Trust Flags:
        SSL Flags:
            Valid CA
            Trusted CA
            User
            Trusted Client CA
        Email Flags:
            User
        Object Signing Flags:
            User
```

▼ 만료된 CA 서명된 서버 인증서 갱신

CA 서명된 서버 인증서(공개 및 개인 키)가 만료된 경우 다음 절차에 따라 인증서를 갱신할 수 있습니다. Directory Server 관리 콘솔을 사용하여 이 작업을 수행할 수도 있습니다.

1. **인증 기관으로부터 업데이트된 CA 서명된 서버 인증서를 얻습니다.**

2. **업데이트된 인증서를 받은 다음 인증서를 설치합니다.**

   ```
   msgcert renew-cert cert_alias cert_file
   ```

▼ CA 서명된 서버 인증서 내보내기 및 가져오기

경우에 따라 나중에 인증서를 다른 시스템(예: 다른 호스트)으로 가져올 수 있도록 인증서를 내보낼 수도 있습니다. Directory Server 관리 콘솔을 사용하여 이 작업을 수행할 수도 있습니다.

1. **인증서를 내보냅니다.**

   ```
   msgcert export-cert [-o OUTPUT_FILE] CERT_ALIAS
   ```

   예를 들면 다음과 같습니다.

   ```
   $ ./msgcert export-cert -o /tmp/first-certificate "First Certificate"
   $./msgcert export-cert -o /tmp/first-server-certificate Server-Cert
   Choose the PKCS#12 file password:
   Confirm the PKCS#12 file password:
   $ls /tmp
   first-server-certificate
   /tmp/first-certificate
   ```

2. **인증서를 가져옵니다.**

   ```
   $ msgcert import-cert  CERT_FILE
   ```

   예를 들어, 인증서를 가져오려면 다음 명령을 사용합니다.

   ```
   $ msgcert import-cert /tmp/first-server-certificate
   Enter the PKCS#12 file password:
   $
   ```

23.5.2 SSL 사용 및 암호문 선택

콘솔을 사용하여 SSL을 사용 가능하게 하고 Messaging Server가 클라이언트와의 암호화된 통신에 사용할 수 있는 암호화 암호문 집합을 선택할 수 있습니다. msgcert 유틸리티를 사용하여 SSL 인증서를 설치하고 해당 configutil을 실행하거나 해당 서비스에 대해 SSL을 활성화하는 데 필요한 구성 파일을 편집할 수도 있습니다.

23.5.2.1 암호문 정보

암호문은 암호화 프로세스에서 데이터를 암호화 및 해독하는 데 사용되는 알고리즘입니다. 일부 암호문은 다른 암호문보다 강력한데 이는 이러한 암호문으로 스크램블된 메시지를 권한 없는 사용자가 해독하는 것이 더 어렵다는 것을 의미합니다.

암호문은 키(긴 번호)를 데이터에 적용하는 방법으로 데이터에서 작동합니다. 일반적으로 암호문이 암호화 도중에 사용하는 키가 더 길수록 적절한 암호화 키 없이 데이터를 해독하는 것이 어려워집니다.

클라이언트는 Messaging Server와의 SSL 연결을 시작할 때 암호화에 사용할 선호되는 암호문과 키 길이를 서버에 알려줍니다. 모든 암호화된 통신에서 클라이언트와 서버는 동일한 암호문을 사용해야 합니다. 일반적으로 다양한 암호문 및 키 조합이 사용되기 때문에 서버는 암호화를 유연하게 지원해야 합니다. Messaging Server는 암호문 및 키 길이 조합을 최대 6개까지 지원할 수 있습니다.

표 23–2에는 SSL 3.0과 함께 사용하는 Messaging Server에서 지원하는 암호문이 나열되어 있습니다. 이 표에 요약된 정보의 자세한 내용은 **Managing Servers with iPlanet Console**의 *Introduction to SSL* 절에서 확인할 수 있습니다.

표 23–2 Messaging Server의 SSL 암호문

암호문	설명
128비트 암호화 및 MD5 메시지 인증을 사용하는 RC4	가장 빠른 암호화 암호문(RSA에 의한)이며 강도가 매우 높은 암호문 및 암호화 키의 조합입니다.
168비트 암호화 및 SHA 메시지 인증을 사용하는 Triple DES	더 느린 암호화 암호문(미국 정보 표준)이지만 강도가 가장 높은 암호문 및 암호화 키의 조합입니다.
56비트 암호화 및 SHA 메시지 인증을 사용하는 DES	더 느린 암호화 암호문(미국 정보 표준)이며 강도가 보통인 암호문 및 암호화 키의 조합입니다.
40비트 암호화 및 MD5 메시지 인증을 사용하는 RC4	가장 빠른 암호화 암호문(RSA에 의한)이며 강도가 낮은 암호문 및 암호화 키의 조합입니다.
40비트 암호화 및 MD5 메시지 인증을 사용하는 RC2	더 느린 암호화 암호문(RSA에 의한)이며 강도가 낮은 암호문 및 암호화 키의 조합입니다.
암호화 없음, MD5 메시지 인증만 사용	암호화가 없으며 메시지 다이제스트만 인증에 사용됩니다.

특정 암호문을 사용하지 않을 중요한 이유가 없을 경우 모든 암호문을 지원해야 합니다. 그러나 일부 국가에서는 수출법에 따라 특정 암호화 암호문의 사용이 제한된다는 것에 주의합니다. 또한 미국 수출 제한법이 완화되기 전에 만들어진 일부 클라이언트 소프트웨어는 더 높은 강도의 암호화를 사용할 수 없습니다. 40비트 암호문이 우발적인 도청을 방지할 수 있지만 보안되지는 않으므로 적극적인 공격을 차단하지 않는다는 것에 주의합니다.

SSL을 사용 가능하게 하고 암호화 암호문을 선택하려면 다음 명령줄 단계를 따릅니다.

인증서를 지정하려면 다음을 수행합니다.

```
configutil -o encryption.rsa.nssslpersonalityssl -v certname
```

또한 SSL 서버 인증서 별명에 대한 서비스별 구성 설정이 있습니다. 새 configutil 설정은 다음과 같습니다.

`local.imta.sslnicknames`(SMTP 및 Submit 서버용), `local.imap.sslnicknames`(IMAP 서버용), `local.pop.sslnicknames`(POP 서버용), `local.http.sslnicknames`(웹 메일 서버용)

이러한 설정은 의미가 동일하며 `encryption.rsa.nssslpersonalityssl` 설정을 대체합니다. 특히, 이 설정은 쉼표로 분리된 NSS 인증서 별명 목록입니다. 목록에 허용되는 별명이 여러 개 있더라도 각 별명은 서로 다른 유형의 인증서(예: RSA 인증서 및 DSS 인증서)를 참조해야 하므로 설정은 거의 항상 한 개의 별명입니다. NSS 소프트웨어 토큰 또는 기본 토큰을 검색하는 경우 별명이 비정규화될 수 있습니다. 또는 지정된 보안 모듈에서 해당 별명을 검색하는 경우 *security-module*:*nickname*" 형식을 사용할 수 있습니다. 이는 기본 NSS 데이터베이스 이외의 위치나 하드웨어 토큰에 저장된 인증서에 필요합니다.

또한 제품에서 여러 NSS 소프트웨어 토큰을 사용할 수 없습니다. 특히 IMAP, POP, SMTP 및 HTTP용으로는 `cert8.db`, `key3.db` 및 `secmod.db`가 하나만 제공됩니다. NSS에서는 그렇지 않습니다.

주 – 보내는 메시지에 대해 SSL 암호화를 사용하려면 채널 정의를 수정하여 `maytls`, `musttls` 등의 `tls` 채널 키워드를 포함해야 합니다. 자세한 내용은 351 페이지 "12.4.8 전송 계층 보안"을 참조하십시오.

23.5.3 인증서 기반 로그인 설정

비밀번호 기반 인증 외에도 Sun Java System 서버는 디지털 인증서 검사를 통한 사용자 인증을 지원합니다. 인증서 기반 인증에서 클라이언트는 서버와의 SSL 세션을 설정하고 사용자의 인증서를 서버로 제출합니다. 그런 다음 서버는 제출된 인증서가 진짜인지 여부를 평가합니다. 인증서가 검증될 경우 사용자는 인증된 것으로 간주됩니다.

인증서 기반 로그인을 사용하도록 Messaging Server를 설정하려면 다음을 수행합니다.

▼ 인증서 기반 로그인 설정

1. **서버의 서버 인증서를 얻습니다. 자세한 내용은 688 페이지 "23.5.1 인증서 얻기"를 참조하십시오.**

2. **인증서 설정 마법사를 실행하여 서버가 인증할 사용자에게 인증서를 발급하는 신뢰할 수 있는 모든 인증 기관의 인증서를 설치합니다. 자세한 내용은 692 페이지 "23.5.1.6 신뢰할 수 있는 CA의 인증서 설치"를 참조하십시오.**

   서버의 데이터베이스에 최소한 하나 이상의 신뢰할 수 있는 CA가 있을 경우 서버는 각 연결 클라이언트로부터 클라이언트 인증서를 요청한다는 것에 주의합니다.

3 **SSL을 설정합니다. 자세한 내용은 697 페이지 "23.5.2 SSL 사용 및 암호문 선택"을 참조하십시오.**

4 **(선택 사항) 제출된 인증서의 정보에 기초하여 서버가 LDAP 사용자 디렉토리를 적절하게 검색하도록 서버의 `certmap.conf` 파일을 편집합니다.**

사용자의 인증서에 있는 전자 메일 주소가 사용자의 디렉토리 항목에 있는 전자 메일 주소와 일치하며 검색을 최적화하거나 사용자 항목의 인증서에 대해 제출된 인증서를 검증할 필요가 없을 경우에는 `certmap.conf` 파일을 편집할 필요가 없습니다.

`certmap.conf`의 형식과 변경할 수 있는 사항에 대한 자세한 내용은 *Managing Servers with iPlanet Console*에서 SSL 장을 참조하십시오.

이러한 단계를 수행하고 나면 사용자가 IMAP 또는 HTTP에 로그인할 수 있도록 클라이언트가 SSL 세션을 설정할 때 Messaging Server는 클라이언트로부터 사용자의 인증서를 요청합니다. 클라이언트에 의해 제출된 인증서가 서버에서 신뢰할 수 있는 것으로 설정한 CA가 발급했으며 인증서의 아이디가 사용자 디렉토리의 항목과 일치할 경우 사용자가 인증되며 액세스가 허가됩니다(해당 사용자를 제어하는 액세스 제어 규칙에 따라).

인증서 기반 로그인을 사용 가능하게 하기 위해 비밀번호 기반 로그인을 허용하지 않을 필요는 없습니다. 비밀번호 기반 로그인이 허용되는 기본 상태에서 이 절에 설명된 작업을 수행할 경우 비밀번호 기반 및 인증서 기반 로그인이 모두 지원됩니다. 이 경우 클라이언트가 SSL 세션을 설정하고 인증서를 제공할 경우 인증서 기반 로그인이 사용됩니다. 클라이언트가 SSL을 사용하지 않거나 인증서를 제공하지 않을 경우 서버는 비밀번호를 요청합니다.

23.5.4 SMTP 프록시를 사용하여 SSL 성능을 최적화하는 방법

SMTP 프록시는 SMTP 프로토콜에 추가 대기 시간을 야기하므로 대부분의 사이트는 SMTP 프록시를 사용해서는 안 됩니다. 그러나 SMTP 연결을 보호하기 위해 SSL에 많이 의존하는 대규모 사이트는 SSL 및 프록시 외에는 일체 수행하지 않는 서버에서 모든 프로토콜에 대해 모든 SSL 작업을 수행하여 SSL 가속기 하드웨어에 대한 투자를 극대화하길 원할 수 있습니다. SMTP 프록시를 사용하면 메시지 대기열을 별개의 MTA 시스템에 두면서 SSL을 프런트엔드 프록시 서버에서 처리할 수 있습니다. 이러한 방법으로 각 작업에 대해 최적화된 하드웨어를 별도로 구성 및 구입할 수 있습니다.

SMTP 프록시를 설치하는 방법은 **Sun Java Communications Suite 5 Deployment Planning Guide**의 "Using the MMP SMTP Proxy" 및 712 페이지 "23.8 POP before SMTP 사용"을 참조하십시오.

23.6 Messaging Server에 대한 관리자 액세스 구성

이 절의 내용은 주로 Sun Java System LDAP Schema v. 1과 관련됩니다. 이 절은 다음과 같은 하위 절로 구성되어 있습니다.

이 절에서는 Messaging Server에 대한 액세스 권한을 서버 관리자가 어떻게 얻을 수 있는지 제어하는 방법에 대해 설명합니다. 주어진 Messaging Server와 특정 Messaging Server 작업에 대한 관리 액세스는 위임된 서버 관리의 컨텍스트 내에서 발생합니다.

위임된 서버 관리는 대부분의 Sun Java System 서버가 갖고 있는 기능으로서 특정 관리자가 개별 서버 또는 서버 기능에 대한 선택적 액세스를 다른 관리자에게 제공하는 기능을 말합니다. 이 장에서는 위임된 서버 작업을 간단하게 요약하여 설명합니다. 자세한 내용은 *Managing Servers with iPlanet Console*에서 서버 관리 위임에 대한 장을 참조하십시오.

23.6.1 위임된 관리 계층

네트워크에 Sun Java System 서버를 처음 설치하면 설치 프로그램은 LDAP 사용자 디렉토리에 구성 관리자 그룹이라는 그룹을 자동으로 만듭니다. 기본적으로 구성 관리자 그룹의 구성원은 네트워크의 모든 호스트와 서버에 대한 무제한적인 액세스 권한을 가집니다.

구성 관리자 그룹은 Messaging Server에 대한 위임된 관리를 구현(Sun Java System LDAP Schema v. 1이 사용될 경우)하기 위해 작성할 수 있는 다음과 같은 액세스 계층의 최상위에 위치합니다.

1. **구성 관리자.**Sun Java System 서버의 네트워크에 대한 "수퍼유저"입니다. 모든 자원에 대한 전체 액세스 권한을 가집니다.
2. **서버 관리자.** 도메인 관리자는 각 유형의 서버를 관리하기 위해 그룹을 만들 수 있습니다. 예를 들어, 관리 도메인이나 전체 네트워크에서 모든 Messaging Server를 관리하기 위해 메시징 관리자 그룹을 만들 수 있습니다. 이러한 그룹의 구성원은 다른 서버는 제외하고 해당 관리 도메인의 모든 Messaging Server에 액세스할 수 있습니다.
3. **작업 관리자.** 마지막으로 위 관리자는 모두 단일 Messaging Server 또는 Messaging Server 집합에 대한 제한된 액세스 권한을 가진 그룹을 만들거나 개별 사용자를 지정할 수 있습니다. 이러한 작업 관리자는 서버를 시작 또는 중지하거나 특정 서비스의 로그를 액세스하는 등의 제한된 특정 서버 작업만 수행할 수 있습니다.

콘솔은 관리자가 다음 작업을 수행할 수 있는 편리한 인터페이스를 제공합니다.

- "서버 전체에 대한 액세스 제공"(다음)에 설명된 대로 특정 Messaging Server에 대한 액세스를 그룹이나 개인에게 허가합니다.

- 702 페이지 "23.6.2 특정 작업에 대한 액세스 제한"에 설명된 대로 특정 Messaging Server의 특정 작업에 대한 액세스를 제한합니다.

▼ 서버 전체에 대한 액세스 제공

이 절에서는 특정 Messaging Server 인스턴스에 대한 액세스 권한을 사용자나 그룹에 제공하는 방법에 대해 설명합니다.

1. **액세스를 제공할 Messaging Server에 액세스할 수 있는 관리자로서 콘솔에 로그인합니다.**
2. **콘솔 창에서 해당 서버를 선택합니다.**
   콘솔 메뉴에서 객체를 선택한 다음 액세스 권한 설정을 선택합니다.
3. **서버를 액세스할 수 있는 사용자 및 그룹 목록을 추가하거나 편집합니다.**
   보다 자세한 지침은 *Managing Servers with iPlanet Console*에서 서버 관리 위임에 대한 장을 참조하십시오)

   특정 Messaging Server에 액세스할 수 있는 개인 및 그룹 목록을 설정한 후에는 다음에 설명된 것처럼 ACI를 사용하여 특정 서버 작업을 해당 목록의 특정 개인 또는 그룹에 위임할 수 있습니다.

23.6.2 특정 작업에 대한 액세스 제한

관리자는 일반적으로 하나 이상의 관리 작업을 수행하기 위해 서버에 연결합니다. 일반 관리 작업은 콘솔에서 Messaging Server 작업 양식에 나열됩니다.

기본적으로 특정 Messaging Server에 대한 액세스는 모든 해당 작업에 대한 액세스를 의미합니다. 그러나 작업 양식의 각 작업은 추가된 액세스 제어 명령(ACI) 집합을 가질 수 있습니다. 서버는 작업에 대한 액세스를 연결된 사용자(서버 전체에 대한 액세스 권한을 이미 갖고 있는 사용자)에게 제공하기 전에 이러한 ACI를 참조합니다. 실제로 서버는 사용자가 권한을 갖고 있는 작업만 작업 양식에 표시합니다.

Messaging Server에 액세스할 수 있는 경우 임의의 작업(즉, 액세스할 수 있는 작업)에 대한 ACI를 작성 및 편집함으로써 다른 사용자나 그룹이 가질 수 있는 작업 액세스 권한을 제한할 수 있습니다.

▼ 사용자 또는 그룹의 작업 액세스 제한 방법

1. **제한된 액세스를 제공할 Messaging Server에 액세스할 수 있는 관리자로서 콘솔에 로그인합니다.**
2. **서버를 열고 작업 텍스트를 눌러 서버의 작업 양식에서 작업을 선택합니다.**

3 **편집 메뉴에서 액세스 권한 설정을 선택하고 액세스 규칙 목록을 추가 또는 편집하여 사용자나 그룹에 원하는 종류의 액세스를 제공합니다.**

4 **다른 작업에 대해 적절하게 이 과정을 반복합니다.**

보다 자세한 지침은 *Managing Servers with iPlanet Console*에서 서버 관리 위임에 대한 장을 참조하십시오)

ACI와 ACI 작성 방법에 대한 자세한 내용은 *Managing Servers with iPlanet Console*에서 서버 관리 위임에 대한 장에 자세히 설명되어 있습니다.

23.7 POP, IMAP 및 HTTP 서비스에 대한 클라이언트 액세스 구성

이 절에는 다음과 같은 하위 절이 포함됩니다.

Messaging Server는 서버에 액세스할 수 있는 클라이언트를 광범위하고 세부적으로 제어할 수 있도록 IMAP, POP 및 HTTP 서비스에 대한 정교한 서비스별 액세스 제어를 지원합니다.

대기업이나 인터넷 서비스 공급자에 대한 메시징 서비스를 관리하는 중이면 이러한 기능은 시스템에서 스팸 발송자 및 DNS 스푸퍼를 차단하고 네트워크의 일반 보안을 향상시키는 데 도움을 줄 수 있습니다. 특히 원하지 않은 대량 전자 메일을 제어하는 방법은 18 장을 참조하십시오.

주 – IP 주소별로 액세스를 제어하는 것이 기업의 중요한 문제가 아닐 경우 이 절에 설명된 필터를 만들 필요가 없습니다. 단지 최소한의 액세스 제어만 필요한 경우에는 710 페이지 "23.7.3.2 대부분 허용"에서 이를 설정하는 방법에 대한 지침을 참조하십시오.

23.7.1 클라이언트 액세스 필터의 작동 방법

Messaging Server 액세스 제어 기능은 서비스되는 TCP 데몬과 동일한 포트에서 수신하는 프로그램입니다. 즉, 이 기능은 액세스 필터를 사용하여 클라이언트 신원을 확인하며 클라이언트가 필터링 프로세스를 통과할 경우 데몬에 대한 액세스 권한을 클라이언트에게 제공합니다.

Messaging Server TCP 클라이언트 액세스 제어 시스템은 필요한 경우 처리 과정의 일부로서 소켓 종점 주소에 대한 다음 분석을 수행합니다.

- 두 종점 모두에 대한 역방향 DNS 조회(이름 기반 액세스 제어를 수행하기 위해)
- 두 종점 모두에 대한 정방향 DNS 조회(DNS 스푸핑을 감지하기 위해)
- `Identd` 콜백(클라이언트 종점의 사용자가 클라이언트 호스트에 알려져 있는 검사하기 위해)

시스템은 **필터**라고 부르는 액세스 제어문에 대해 이 정보를 비교하여 액세스를 허가 또는 거부할지 여부를 결정합니다. 각 서비스에 대해 별도의 허용 필터 및 거부 필터 집합이 액세스를 제어합니다. 허용 필터는 액세스를 명시적으로 허가하며 거부 필터는 액세스를 명시적으로 금지합니다.

클라이언트가 서비스에 대한 액세스를 요청하면 액세스 제어 시스템은 다음 조건을 사용하여 각 서비스의 필터에 대해 클라이언트의 주소 또는 이름 정보를 순서대로 비교합니다.

- 일치하는 첫 번째 항목에서 검색이 중지됩니다. 허용 필터는 거부 필터 이전에 처리되므로 허용 필터가 우선 순위를 가집니다.
- 클라이언트 정보가 해당 서비스의 허용 필터와 일치할 경우 액세스가 허가됩니다.
- 클라이언트 정보가 해당 서비스의 거부 필터와 일치할 경우 액세스가 거부됩니다.
- 허용 또는 거부 필터와 일치하는 항목이 없을 경우 액세스가 허가됩니다. 단, 허용 필터만 있고 거부 필터가 없을 경우 일치하는 항목이 없으면 액세스가 거부됩니다.

여기에서 설명된 필터 구문은 간단한 방식으로 여러 다른 종류의 액세스 제어 정책을 구현할 수 있을 정도로 충분히 유연합니다. 거의 배타적인 허용 또는 거부를 사용하여 대부분의 정책을 구현할 수 있지만 허용 필터와 거부 필터를 둘 다 임의로 조합하여 사용할 수 있습니다.

다음 절에서는 필터 구문을 자세하게 설명하고 사용 예를 제공합니다. 711 페이지 “23.7.4 서비스에 대한 액세스 필터 만들기” 절에서는 액세스 필터를 만들기 위한 절차에 대해 설명합니다.

23.7.2 필터 구문

필터 구문은 서비스 정보와 클라이언트 정보를 모두 포함합니다. 서비스 정보는 서비스 이름, 호스트 이름 및 호스트 주소를 포함할 수 있습니다. 클라이언트 정보는 호스트 이름, 호스트 주소 및 아이디를 포함할 수 있습니다. 서버 및 클라이언트 정보는 둘 다 와일드카드 이름이나 패턴을 포함할 수 있습니다.

가장 간단한 형식의 필터는 다음과 같습니다.

service: *hostSpec*

여기에서 *service*는 서비스 이름(예: `smtp`, `pop`, `imap` 또는 `http`)이고 *hostSpec*은 호스트 이름, IP 주소 또는 액세스를 요청하는 클라이언트는 나타내는 와일드카드 이름 또는

패턴입니다. 필터가 처리될 때 액세스를 요구하는 클라이언트가 *client*와 일치할 경우 *service*에 지정된 서비스에 대한 액세스가 해당 필터 유형에 따라 허용되거나 거부됩니다. 다음은 몇 가지 예입니다.

```
imap: roberts.newyork.siroe.com
pop: ALL
http: ALL
```

이러한 필터가 허용 필터일 경우 첫 번째 필터는 IMAP 서비스에 대한 액세스를 `roberts.newyork.siroe.com` 호스트에 허가하고 두 번째 및 세 번째 필터는 각각 POP 및 HTTP 서비스에 대한 액세스를 모든 클라이언트에게 허가합니다. 이러한 필터가 거부 필터일 경우 이러한 클라이언트는 서비스에 대한 액세스가 거부됩니다. `ALL`과 같은 와일드카드 이름에 대한 자세한 내용은 706 페이지 “23.7.2.1 와일드카드 이름”을 참조하십시오.

필터가 더 일반적인 형식을 가질 경우 필터의 서버 또는 클라이언트 정보는 다음과 같이 다소 복잡할 수 있습니다.

serviceSpec: *clientSpec*

여기에서 *serviceSpec*은 *service* 또는 *service@hostSpec*이 될 수 있고 *clientSpec*은 *hostSpec* 또는 *user@hostSpec*이 될 수 있습니다. *user*는 액세스를 요구하는 클라이언트 호스트와 연관된 사용자 아이디 또는 와일드카드 이름입니다. 다음 두 개의 예를 가정해 봅니다.

```
pop@mailServer1.siroe.com: ALL
imap: srashad@xyz.europe.siroe.com
```

여기에서 이러한 필터가 거부 필터일 경우 첫 번째 필터는 `mailServer1.siroe.com` 호스트에서 SMTP 서비스에 대한 액세스를 모든 클라이언트에 대해 거부합니다. 두 번째 필터는 `xyz.europe.siroe.com` 호스트에서 사용자 `srashad`의 IMAP 서비스에 대한 액세스를 거부합니다. 이러한 확장된 서버 및 클라이언트 지정을 사용하는 시기에 대한 자세한 내용은 708 페이지 “23.7.2.4 서버 호스트 지정” 및 708 페이지 “23.7.2.5 클라이언트 사용자 아이디 지정”을 참조하십시오.

마지막으로 가장 일반적인 형식의 필터는 다음과 같습니다.

serviceList: *clientList*

여기에서 *serviceList*는 하나 이상의 *serviceSpec* 항목으로 구성되며 *clientList*는 하나 이상의 *clientSpec* 항목으로 구성됩니다. *serviceList* 및 *clientList* 내의 개별 항목은 공백 및/또는 쉼표로 구분됩니다.

여기에서는 필터가 처리될 때 액세스를 요구하는 클라이언트가 *clientList*의 *clientSpec* 항목 중 하나와 일치할 경우 *serviceList*에 지정된 모든 서비스에 대한 액세스가 해당 필터 유형에 따라 허용되거나 거부됩니다. 다음은 이에 대한 한 예입니다.

```
pop, imap, http: .europe.siroe.com .newyork.siroe.com
```

이 필터가 허용 필터일 경우 `europe.siroe.com` 또는 `newyork.siroe.com` 도메인에 있는 모든 클라이언트는 POP, IMAP 및 HTTP 서비스에 대한 액세스가 허가됩니다. 선행 점이나 다른 패턴을 사용하여 도메인 또는 서브넷을 지정하는 방법에 대한 자세한 내용은 707 페이지 "23.7.2.2 와일드카드 패턴"을 참조하십시오.

또한 다음 구문을 사용할 수도 있습니다.

"+" 또는 "-" *serviceList*:*$*next_rule*

+(허용 필터)는 클라이언트 목록에 대해 데몬 목록 서비스가 허가된다는 것을 의미합니다.

-(거부 필터)는 클라이언트 목록에 대해 서비스가 거부된다는 것을 의미합니다.

*(와일드카드 필터)는 모든 클라이언트가 이러한 서비스를 사용하도록 허용합니다.

$는 규칙을 구분합니다.

다음 예는 모든 클라이언트에서 여러 서비스를 사용 가능하게 합니다.

```
+imap,pop,http:*
```

다음 예는 여러 규칙을 표시하지만 각 규칙은 서비스 이름을 하나만 가지도록 단순화되며 클라이언트 목록에 와일드카드를 사용합니다. 이는 LDIF 파일에서 액세스 제어를 지정하기 위해 가장 일반적으로 사용되는 방법입니다.

```
+imap:ALL$+pop:ALL$+http:ALL
```

다음 예는 사용자에 대해 모든 서비스를 거부하는 방법을 보여 줍니다.

```
-imap:*$-pop:*$-http:*
```

23.7.2.1 와일드카드 이름

다음 와일드카드 이름을 사용하여 서비스 이름, 호스트 이름이나 주소, 또는 아이디를 나타낼 수 있습니다.

표 23–3 서비스 필터의 와일드카드 이름

와일드카드 이름	설명
`ALL, *`	범용 와일드카드입니다. 모든 이름과 일치합니다.
`LOCAL`	모든 로컬 호스트(이름에 점 문자가 포함되지 않은 호스트)와 일치합니다. 그러나 설치가 정규 이름만 사용할 경우 로컬 호스트 이름은 점을 포함하므로 이 와일드카드와 일치하지 않습니다.

표 23–3 서비스 필터의 와일드카드 이름 *(계속)*

와일드카드 이름	설명
UNKNOWN	이름이 알려지지 않은 모든 사용자나 이름이나 주소가 알려지지 않은 모든 호스트와 일치합니다. 다음과 같이 이 와일드카드 이름을 신중하게 사용합니다. 임시 DNS 서버 문제로 인해 호스트 이름을 사용하지 못할 수 있습니다. 이러한 경우 UNKNOWN을 사용하는 모든 필터는 모든 클라이언트 호스트와 일치합니다. 통신하는 네트워크 유형을 소프트웨어에서 식별할 수 없으면 네트워크 주소를 사용할 수 없습니다. 이러한 경우 UNKNOWN을 사용하는 모든 필터는 해당 네트워크의 모든 클라이언트 호스트와 일치합니다.
KNOWN	이름이 알려진 모든 사용자나 이름 **및** 주소가 알려진 모든 호스트와 일치합니다. 다음과 같이 이 와일드카드 이름을 신중하게 사용합니다. 일시적인 DNS 서버 문제로 인해 호스트 이름을 사용하지 못할 수 있습니다. 이러한 경우 KNOWN을 사용하는 모든 필터를 모든 클라이언트 호스트에서 실패합니다. 통신하는 네트워크 유형을 소프트웨어에서 식별할 수 없으면 네트워크 주소를 사용할 수 없습니다. 이러한 경우 KNOWN을 사용하는 모든 필터는 해당 네트워크의 모든 클라이언트 호스트에서 실패합니다.
DNSSPOOFER	DNS 이름이 고유한 IP 주소와 일치하는 않는 모든 호스트와 일치합니다.

23.7.2.2 와일드카드 패턴

다음 패턴을 서비스나 클라이언트 주소에서 사용할 수 있습니다.

- 점 문자(.)로 시작하는 문자열. 이름의 마지막 구성 요소가 지정된 패턴과 일치할 경우 호스트 이름이 일치합니다. 예를 들어, 와일드카드 패턴 `.siroe.com`은 도메인 `siroe.com`의 모든 호스트와 일치합니다.
- 점 문자(.)로 끝나는 문자열. 첫 번째 숫자 필드가 지정된 패턴과 일치할 경우 호스트 주소가 일치합니다. 예를 들어, 와일드카드 패턴 `123.45.`는 서브넷 `123.45.0.0`에 있는 모든 호스트의 주소와 일치합니다.
- `n.n.n.n/m.m.m.m` 형식의 문자열. 이 와일드카드 패턴은 *net/mask* 쌍으로 해석됩니다. *net*이 주소 및 *mask*의 비트 AND와 같은 경우 호스트 주소가 일치합니다. 예를 들어, 패턴 `123.45.67.0/255.255.255.128`은 `123.45.67.0`에서 `123.45.67.127`까지의 모든 주소와 일치합니다.

23.7.2.3 EXCEPT 연산자

액세스 제어 시스템은 단일 연산자를 지원합니다. EXCEPT 연산자를 사용하면 *serviceList* 또는 *clientList*에 여러 항목이 있을 경우 일치하는 이름이나 패턴에 대한 예외를 만들 수 있습니다. 예를 들어, 다음 표현식은

list1 EXCEPT *list2*

*list1*과 일치하는 항목이 또한 *list2*와 일치하지 **않을** 경우에 일치한다는 것을 의미합니다.

다음은 이에 대한 한 예입니다.

```
ALL: ALL EXCEPT isserver.siroe.com
```

이 예는 거부 필터인 경우 `isserver.siroe.com` 호스트 시스템에 있는 것을 제외하고 모든 클라이언트에 대해 모든 서비스의 액세스를 거부합니다.

`EXCEPT` 절은 중복될 수 있습니다. 다음 표현식은

```
list1 EXCEPT list2 EXCEPT list3
```

다음과 같은 것으로 평가됩니다.

```
list1 EXCEPT (list2 EXCEPT list3)
```

23.7.2.4 서버 호스트 지정

서버 호스트 이름이나 주소 정보를 *serviceSpec* 항목에 포함하여 요청되는 특정 서비스를 필터에서 추가로 식별할 수 있습니다. 이러한 경우 항목의 형식은 다음과 같습니다.

```
service@hostSpec
```

Messaging Server 호스트 시스템이 다른 인터넷 호스트 이름을 가진 여러 인터넷 주소에 대해 설정된 경우 이 기능을 사용할 수 있습니다. 서비스 공급자인 경우에는 이 기능을 사용하여 다른 액세스 제어 규칙을 가진 여러 도메인을 단일 서버 인스턴스에서 호스트할 수 있습니다.

23.7.2.5 클라이언트 사용자 아이디 지정

RFC 1413에 설명된 대로 `identd` 서비스를 지원하는 클라이언트 호스트 시스템의 경우 클라이언트의 사용자 아이디를 필터의 *clientSpec* 항목에 포함하여 특정 클라이언트 요청 서비스를 추가로 식별할 수 있습니다. 이러한 경우 항목의 형식은 다음과 같습니다.

```
user@hostSpec
```

여기에서 *user*는 클라이언트의 `identd` 서비스(또는 와일드카드 이름)에 의해 반환되는 사용자 아이디입니다.

필터에서 클라이언트 아이디를 지정하는 것은 유용할 수 있지만 다음과 같은 사항에 주의해야 합니다.

- `identd` 서비스는 인증이 아닙니다. 클라이언트 시스템이 손상된 경우 이 서비스가 반환하는 클라이언트 사용자 아이디를 신뢰할 수 없습니다. 일반적으로 특정 사용자 아이디를 사용하지 않으며 `ALL`, `KNOWN` 또는 `UNKNOWN` 와일드카드 이름만 사용합니다.
- `identd`는 대부분의 최신 클라이언트 시스템에서 지원되지 않으므로 현대적인 배포에서는 거의 가치가 없습니다. 이후 버전에서는 `identd` 지원을 제거하는 것이 고려되고 있으므로 이 기능이 필요한 경우 Sun Java System에 따로 알릴 필요가 있을 것입니다.

- 사용자 아이디 조회는 시간이 걸립니다. 따라서 모든 사용자에 대한 조회를 수행하면 identd를 지원하지 않는 클라이언트의 액세스가 느려질 수 있습니다. 선택적 사용자 아이디 조회로 이 문제를 줄일 수 있습니다. 예를 들어, 다음과 같은 규칙은

  serviceList: @xyzcorp.com ALL@ALL

  도메인 xyzcorp.com의 사용자를 사용자 아이디 조회를 수행하지 않고 비교하지만 다른 모든 시스템에서는 사용자 아이디 조회를 수행합니다.

경우에 따라 사용자 아이디 조회 기능은 클라이언트 호스트에서 권한 없는 사용자의 공격으로부터 보호하는 데 도움이 될 수 있습니다. 예를 들어, 일부 TCP/IP 구현에서는 rsh(원격 쉘 서비스)를 사용하는 침입자가 신뢰할 수 있는 클라이언트 호스트를 가장할 수 있습니다. 클라이언트 호스트가 ident 서비스를 지원할 경우 사용자 아이디 조회를 사용하여 이러한 공격을 감지할 수 있습니다.

23.7.3 필터 예

이 절의 예는 액세스 제어의 다양한 방법을 보여 줍니다. 이러한 예에서는 허용 필터가 거부 필터보다 먼저 처리되고 일치하는 항목이 발견될 때 검색이 종료하며 일치하는 항목이 전혀 없을 경우 액세스가 허가된다는 것에 주의합니다.

여기에 나열된 예는 IP 주소가 아니라 호스트 및 도메인 이름을 사용합니다. 주소와 넷마스크 정보를 필터에 포함하여 이름 서비스 실패를 대비한 안정성을 향상시킬 수 있다는 것에 주의합니다.

23.7.3.1 대부분 거부

이 경우에는 액세스가 기본적으로 거부됩니다. 명시적으로 허가된 호스트만 액세스가 허용됩니다.

기본 정책(액세스 없음)은 다음과 같은 평범하고 단순한 거부 파일을 통해 구현됩니다.

```
ALL: ALL
```

이 필터는 허용 필터에 의해 액세스가 명시적으로 허가되지 않은 모든 클라이언트에 대한 모든 서비스를 거부합니다. 그런 다음 허용 필터는 다음과 같을 수 있습니다.

```
ALL: LOCAL @netgroup1

ALL: .siroe.com EXCEPT externalserver.siroe.com
```

첫 번째 규칙은 로컬 도메인(즉, 호스트 이름에 점이 없는 모든 호스트)의 모든 호스트와 netgroup1 그룹의 구성원에 대해 액세스를 허가합니다. 두 번째 규칙은 선행 점 와일드카드 패턴을 사용하여 externalserver.siroe.com 호스트를 제외하고 siroe.com 도메인에 있는 모든 호스트의 액세스를 허가합니다.

23.7.3.2 대부분 허용

이 경우에는 액세스가 기본적으로 허가됩니다. 명시적으로 지정된 호스트만 액세스가 거부됩니다.

기본 정책(액세스 허가)은 허용 필터를 불필요하게 만듭니다. 원하지 않는 클라이언트는 다음과 같이 거부 필터에 명시적으로 나열됩니다.

```
ALL: externalserver.siroe1.com, .siroe.asia.com
ALL EXCEPT pop: contractor.siroe1.com, .siroe.com
```

첫 번째 필터는 특정 호스트와 특정 도메인에 대해 모든 서비스를 거부합니다. 두 번째 필터는 특정 호스트와 특정 도메인의 POP 액세스만 허가합니다.

23.7.3.3 스푸핑된 도메인에 대한 액세스 거부

필터에서 DNSSPOOFER 와일드카드 이름을 사용하여 호스트 이름 스푸핑을 감지할 수 있습니다. DNSSPOOFER를 지정하면 액세스 제어 시스템은 정방향 또는 역방향 DNS 조회를 수행하여 클라이언트가 제공한 호스트 이름이 실제 IP 주소와 일치하는지 확인합니다. 다음은 거부 필터의 예입니다.

```
ALL: DNSSPOOFER
```

이 필터는 IP 주소가 해당 DNS 호스트 이름과 일치하지 않는 모든 원격 호스트에 대한 모든 서비스를 거부합니다.

23.7.3.4 가상 도메인에 대한 액세스 제어

단일 서버 인스턴스가 여러 IP 주소 및 도메인 이름과 연관된 가상 도메인을 메시징 설치에서 사용할 경우 허용 및 거부 필터를 조합하여 각 가상 도메인에 대한 액세스를 제어할 수 있습니다. 예를 들어, 다음 허용 필터를

```
ALL@msgServer.siroe1.com: @.siroe1.com
ALL@msgServer.siroe2.com: @.siroe2.com
...
```

다음 거부 필터와 결합하여 사용할 수 있습니다.

```
ALL: ALL
```

각 허용 필터는 domain*N* 내의 호스트만 IP 주소가 msgServer.siroe*N* .com에 해당하는 서비스에 연결되도록 허용합니다. 다른 모든 연결은 거부됩니다.

23.7.3.5 웹 메일 액세스를 허용하면서 IMAP 액세스 제어

사용자의 웹 메일 액세스는 허용하고 IMAP 액세스는 금지하려면 다음과 같은 필터를 만듭니다.

+imap:*access_server_host*, *access_server_host*

이 필터는 액세스 서버 호스트의 IMAP**만** 허용합니다. `service.imap.domainallowed`를 사용하여 IMAP 서버 수준에서 필터를 설정하거나 LDAP 속성을 사용하여 도메인/사용자 수준에서 필터를 설정할 수 있습니다.

23.7.4 서비스에 대한 액세스 필터 만들기

IMAP, POP 또는 HTTP 서비스에 대한 허용 및 거부 필터를 만들 수 있습니다. 또한 이러한 필터를 SMTP 서비스에 대해 만들 수 있지만 인증된 SMTP 세션에만 적용된다는 점에서 이것은 거의 가치가 없습니다. 18 장을 참조하십시오.

▼ 필터를 만드는 방법

- **명령줄. 다음과 같이 명령줄에서 액세스 및 거부 필터를 지정할 수도 있습니다.**

  서비스에 대한 액세스 필터를 작성 또는 편집하려면 다음을 수행합니다.

  `configutil -o service.`*service*`.domainallowed -v` *filter***여기에서** *service***는** `pop`, `imap` **또는** `http`**이고** *filter***는** 704 페이지 "23.7.2 필터 구문"**에 설명된 구문 규칙을 따릅니다**.

  서비스에 대한 거부 필터를 작성 또는 편집하려면 다음을 수행합니다.

  ```
  configutil -o service.service.domainnotallowed -v filter
  ```

  여기에서 *service*는 `pop`, `imap` 또는 `http`이고 *filter*는 704 페이지 "23.7.2 필터 구문"에 설명된 구문 규칙을 따릅니다. 다양한 예는 709 페이지 "23.7.3 필터 예"를 참조하십시오.

23.7.5 HTTP 프록시 인증에 대한 액세스 필터 만들기

모든 저장소 관리자는 모든 서비스에 대해 프록시 인증을 수행할 수 있습니다. 저장소 관리자에 대한 자세한 내용은 561 페이지 "20.4 저장소에 대한 관리자 액세스 지정"을 참조하십시오. 모든 사용자는 해당 클라이언트 호스트가 프록시 인증 액세스 필터를 통해 액세스가 허가된 경우 HTTP 서비스에 한하여 프록시 인증을 수행할 수 있습니다.

프록시 인증을 사용하면 포털 사이트 등의 다른 서비스에서 사용자를 인증하고 인증서를 HTTP 로그인 서비스로 전달할 수 있습니다. 예를 들어, 포털 사이트가 여러 서비스를 제공하며 그 중 하나가 Messenger Express 웹 기반 전자 메일이라고 가정해 봅니다. 이 경우 최종 사용자는 HTTP 프록시 인증 기능을 사용하여 포털 서비스에 한 번만 인증되면 됩니다. 즉, 전자 메일을 액세스하기 위해 다시 인증될 필요가 없습니다. 포털 사이트는 클라이언트와 서비스 간의 인터페이스로 작동하는 로그인 서버를 구성해야 합니다. Messenger Express 인증을 위한 로그인 서버의 구성을 돕기 위해 Sun Java System은 Messenger Express용 인증 SDK를 제공합니다.

이 절에서는 IP 주소별로 HTTP 프록시 인증을 허용하기 위해 허용 필터를 만드는 방법에 대해 설명합니다. 로그인 서버를 설정하는 방법이나 Messenger Express 인증 SDK를

사용하는 방법은 이 절에서 설명하지 않습니다. Messenger Express에 맞게 로그인 서버를 설정하고 인증 SDK를 사용하는 방법에 대한 자세한 내용은 Sun Java System 담당자에게 문의하십시오.

▼ HTTP 프록시 인증에 대한 액세스 필터 만들기

- **명령줄. 다음과 같이 명령줄에서 HTTP 서비스에 대한 프록시 인증을 위한 액세스 필터를 지정합니다.**

  ```
  configutil -o service.service.proxydomainallowed -v filter
  ```

  여기에서 *filter*는 704 페이지 "23.7.2 필터 구문"에 설명된 구문 규칙을 따릅니다.

23.8 POP before SMTP 사용

SMTP 인증 또는 *SMTP Auth*(RFC 2554)는 SMTP 릴레이 서버 보안을 제공하는 데 선호되는 방법입니다. SMTP Auth는 인증된 사용자만 MTA를 통해 메일을 보낼 수 있도록 허용합니다. 그러나 일부 레거시 클라이언트는 *POP before SMTP*에 대한 지원만 제공합니다. 이러한 시스템에서는 아래 설명된 것처럼 POP before SMTP를 사용 가능하게 할 수 있습니다. 그러나 가능하면 사용자에게 POP before SMTP를 사용하는 대신 POP 클라이언트를 업그레이드하도록 권장하는 것이 좋습니다. 사이트에서 POP before SMTP가 배포되고 나면 사용자는 인터넷 보안 표준을 따르지 못하는 클라이언트에 의존하게 됩니다. 따라서 최종 사용자가 해킹의 위험에 노출될 뿐만 아니라 성공적인 최근 POP 세션의 IP 주소를 추적 및 조정하면서 발생하는 불가피한 성능 저하로 인해 사이트 속도가 느려집니다.

Messaging Server 의 POP before SMTP 구현은 SIMS 또는 Netscape Messaging Server에서와 완전히 다릅니다. POP 및 SMTP 프록시 모두를 가지도록 Messaging Multiplexor(MMP)를 구성하여 POP before SMTP를 지원합니다. SMTP 클라이언트가 SMTP 프록시에 연결하면 프록시는 최근 POP 인증의 메모리 내장 캐시를 검사합니다. 동일한 클라이언트 IP 주소의 POP 인증이 발견되면 SMTP 프록시는 메시지를 로컬 및 비로컬 수신자 모두에게 보낼 수 있게 허용해야 한다는 것을 SMTP 서버에 알립니다.

▼ SMTP 프록시 설치 방법

SMTP 프록시 사용에 대한 자세한 내용은 **Sun Java Communications Suite 5 Deployment Planning Guide**의 "Using the MMP SMTP Proxy"를 참조하십시오.

1. **MMP(Messaging Multiplexor)를 설치합니다**

   자세한 내용은 **Sun Java Communications Suite 5 Installation Guide**를 참조하십시오.

2. **MMP에서 SMTP 프록시를 사용 가능하게 합니다.**

   다음 문자열을

msg-svr-base/lib/SmtpProxyAService@25|587

msg-svr-base/config/AService.cfg 파일의 ServiceList 옵션에 추가합니다. 이 옵션은 길이가 한 줄이며 줄 바꿈을 포함할 수 없습니다.

주 – MMP가 업그레이드되면 MMP에 대한 네 개의 기존 구성 파일에 해당하는 네 개의 새 파일이 만들어집니다. 이러한 새 파일은 다음과 같습니다.

AService-def.cfg, ImapProxyAService-def.cfg, PopProxyAService-def.cfg 및 SmtpProxyAService-def.cfg

이러한 파일은 설치 프로그램에 의해 만들어지며, 문서에 설명된 네 개의 구성 파일은 설치하는 동안에 만들어지거나 영향을 받지 않습니다. MMP는 시작되면 일반 구성 파일(현재 문서화되어 있는)을 찾습니다. 일반 구성 파일을 찾지 못한 경우 MMP는 각 *AService-def.cfg 파일을 해당 *AService.cfg 파일 이름에 복사하는 것을 시도합니다.

---

3 각 SMTP 릴레이 서버에서 SMTP 채널 옵션 파일 tcp_local_option의 PROXY_PASSWORD 옵션을 설정합니다.

SMTP 프록시는 SMTP 서버에 연결되면 실제 클라이언트 IP 주소와 다른 연결 정보를 SMTP 서버에 알려 SMTP 서버가 중계 차단 및 다른 보안 정책(POP before SMTP 인증 포함)을 제대로 적용할 수 있게 해야 합니다. 이것은 보안에 민감한 작업으로서 반드시 인증되어야 합니다. MMP SMTP 프록시 및 SMTP 서버 모두에서 구성되는 프록시 비밀번호는 다른 사람이 기능을 남용할 수 없게 합니다.

예: PROXY_PASSWORD=*A_Password*

4 MMP에서 SMTP 서버에 연결하기 위해 사용하는 IP 주소가 INTERNAL_IP 매핑 테이블에서 "내부" 주소로 처리되지 않는지 확인합니다.

INTERNAL_IP 매핑 테이블에 대한 자세한 내용은 18 장의 526 페이지 “18.6 SMTP 릴레이 추가”를 참조하십시오.

5 POP before SMTP를 지원하도록 SMTP 프록시를 구성합니다.

a. *msg-svr-base*/config/SmtpProxyAService.cfg 구성 파일을 편집합니다.

다음 SMTP 프록시 옵션은 IMAP 및 POP 프록시 옵션(7 장 및 **Sun Java System Messaging Server 6.3 Administration Reference**의 “Encryption (SSL) Option” 절에서 이러한 옵션에 대한 설명 참조)과 똑같이 작동합니다.

LdapURL , LogDir, LogLevel, BindDN , BindPass, Timeout, Banner , SSLEnable, SSLSecmodFile, SSLCertFile, SSLKeyFile, SSLKeyPasswdFile, SSLCipherSpecs, SSLCertNicknames, SSLCacheDir , SSLPorts, CertMapFile, CertmapDN, ConnLimits, TCPAccess

위에 나열되지 않은 다른 MMP 옵션(BacksidePort 옵션 포함)은 현재 SMTP 프록시에 적용되지 않습니다.

다음 다섯 개의 옵션을 추가합니다.

SmtpRelays는 라운드 로빈 중계에 사용할 SMTP 중계 서버 호스트 이름(선택적 포트 포함)의 공백으로 구분된 목록입니다. 이러한 중계는 XPROXYEHLO 확장을 지원해야 합니다. 이 옵션은 필수이며 기본값은 없습니다.

예: `default:SmtpRelays manatee:485 gonzo mothra`

SmtpProxyPassword는 SMTP 중계 서버에 대한 소스 채널 변경 사항을 허가하는 데 사용되는 비밀번호입니다. 이 옵션은 필수이고 기본값은 없으며 SMTP 서버의 PROXY_PASSWORD 옵션과 일치해야 합니다.

예: `default:SmtpProxyPassword` *A_Password*

EhloKeywords 옵션은 기본 집합 외에 클라이언트에게 전달할 프록시에 대한 EHLO 확장 키워드 목록을 제공합니다. MMP는 SMTP 중계에 의해 반환된 EHLO 목록에서 인식되지 않은 모든 EHLO 키워드를 제거합니다. EhloKeywords는 이 목록에서 제거하지 않아야 하는 추가 EHLO 키워드를 지정합니다. 기본값은 비어 있지만 8BITMIME, PIPELINING, DSN, ENHANCEDSTATUSCODES, EXPN, HELP, XLOOP, ETRN, SIZE, STARTTLS, AUTH 키워드는 SMTP 프록시에서 지원되므로 이 옵션에 나열할 필요가 없습니다.

다음은 드물게 사용되는 "TURN" 확장을 사용하는 사이트에서 사용할 수 있는 예입니다.

예: `default:EhloKeywords TURN`

PopBeforeSmtpKludgeChannel 옵션은 허가된 POP before SMTP 연결에 사용할 MTA 채널의 이름으로 설정됩니다. 기본값은 비어 있으며 POP before SMTP를 사용 가능하게 하려는 사용자에 대한 일반 설정은 tcp_intranet입니다. 이 옵션은 SSL 성능을 최적화하는 데 필요하지 않습니다(700 페이지 "23.5.4 SMTP 프록시를 사용하여 SSL 성능을 최적화하는 방법" 참조).

예: `default:PopBeforeSmtpKludgeChannel tcp_intranet`

ClientLookup 옵션은 기본적으로 no입니다. yes로 설정된 경우 클라이언트 IP 주소에 대한 DNS 역방향 조회가 무조건 수행되므로 SMTP 중계 서버에서 이 작업을 수행할 필요가 없습니다. 이 옵션은 호스트된 도메인별로 설정할 수 있습니다.

예: `default:ClientLookup yes`

b. PopProxyAService.cfg 구성 파일에서 PreAuth 및 AuthServiceTTL 옵션을 설정합니다. 이 옵션은 SSL 성능을 최적화하는 데 필요하지 않습니다. 700 페이지 "23.5.4 SMTP 프록시를 사용하여 SSL 성능을 최적화하는 방법"을 참조하십시오.

이러한 옵션은 POP 인증 후에 사용자가 메일을 전달할 수 있도록 허가된 시간(초)을 지정합니다. 일반 설정은 900-1800(15-30분)입니다.

예:

```
default:PreAuth yes
default:AuthServiceTTL 900
```

c. 목록에서 다음 항목을 시도하기 전에 MMP가 SMTP 중계의 응답을 대기하는 시간(초)을 선택적으로 지정할 수 있습니다.

기본값은 10(초)입니다. SMTP 중계에 대한 연결이 실패한 경우 MMP는 페일오버 시간 초과에 해당하는 시간(분) 동안 해당 중계를 시도하지 않습니다(따라서 페일오버 시간 초과가 10초이고 중계가 실패한 경우 MMP는 해당 중계를 10분 동안 다시 시도하지 않음).

예: `default:FailoverTimeout 10`

23.9 SMTP 서비스에 대한 클라이언트 액세스 구성

SMTP 서비스에 대한 클라이언트 액세스 구성에 대한 자세한 내용은 18 장을 참조하십시오.

23.10 SSL을 통한 사용자/그룹 디렉토리 조회

MTA, MMP 및 IMAP/POP/HTTP 서비스를 위해 SSL을 통한 사용자/그룹 디렉토리 조회를 수행할 수 있습니다. 전제 조건은 Messaging Server가 SSL 모드에서 구성되어야 한다는 것입니다. 이 기능을 활성화하려면 configutil 매개 변수를 설정해야 합니다. 즉, `local.service.pab.ldapport`는 `636`으로, `local.ugldapport`는 `636`으로, `local.ugldapusessl`은 `1`로 설정합니다.

◆ ◆ ◆ 24장

24 Communications Express Mail용 S/MIME 관리

Secure/Multipurpose Internet Mail Extension(S/MIME) 3.1을 Sun Java System Communications Express Mail에서 사용할 수 있습니다. S/MIME을 사용하도록 설정한 Communications Express Mail 사용자는 Communications Express Mail, Microsoft Outlook Express 및 Mozilla 메일 시스템의 다른 사용자와 서명되었거나 암호화된 메시지를 교환할 수 있습니다.

Communications Express Mail에서 S/MIME을 사용하는 방법에 대한 자세한 내용은 온라인 도움말을 참조하십시오. 이 장에서는 S/MIME을 관리하는 방법에 대해 설명합니다. 이 장은 다음 내용으로 구성되어 있습니다.

24.1 S/MIME이란?

S/MIME은 Communications Express Mail 사용자에게 다음 기능을 제공합니다.

- 보내는 메일 메시지에 대한 디지털 서명을 만들어 메시지를 받는 사람이 메시지가 손상되지 않았으며 메시지를 보낸 사람으로부터 받았음을 확인할 수 있게 합니다.
- 메시지가 수신자의 메일함에 도착하기 전에 다른 사람이 메시지 내용을 확인, 변경 또는 사용하는 것을 방지하기 위해 나가는 메일 메시지를 암호화합니다.

- 인증서 해지 목록(CRL)을 포함하는 프로세스를 사용하여 서명된 받는 메시지의 디지털 서명을 확인합니다.
- 수신자가 메시지의 내용을 읽을 수 있도록 암호화된 받는 메시지를 자동으로 해독합니다.
- 서명 및 암호화된 메시지를 Communications Express Mail 및 Mozilla 메일 시스템 같은 S/MIME 호환 클라이언트를 사용하는 다른 사용자와 교환합니다.

24.1.1 알아야 할 개념

S/MIME을 올바로 관리하려면 다음 개념에 익숙해야 합니다.

- 플랫폼의 기본 관리 절차
- LDAP(Lightweight Directory Access Protocol) 디렉토리의 구조 및 사용
- LDAP 디렉토리의 항목 추가 또는 수정
- Sun Java System Directory Server의 구성 프로세스
- 다음에 대한 개념 및 용도
  - 보안 통신 회선을 위한 SSL(Secure Socket Layer)
  - 디지털 서명된 전자 메일 메시지
  - 암호화된 전자 메일 메시지
  - 브라우저의 로컬 키 저장소
  - 스마트 카드와 이를 사용하기 위한 소프트웨어 및 하드웨어
  - 개인 공개 키 쌍 및 해당 인증서
  - 인증 기관(CA)
  - 키와 해당 인증서 확인
  - 인증서 해지 목록(CRL). (743 페이지 “24.9.2 CRL에 대해 인증서 확인 시기” 참조)

24.2 필수 소프트웨어 및 하드웨어 구성 요소

이 절에서는 S/MIME과 함께 Communications Express Mail을 사용하기 위한 필수 하드웨어 및 소프트웨어에 대해 설명합니다. S/MIME 구성을 시도하기 전에 서버와 클라이언트 시스템에 올바른 버전의 소프트웨어를 모두 설치해야 합니다.

표 24–1에는 Communications Express Mail에 액세스하는 클라이언트 시스템의 필수 소프트웨어 및 하드웨어가 나열되어 있습니다.

표 24–1 클라이언트 시스템의 필수 하드웨어 및 소프트웨어

구성 요소	설명
운영 체제	■ Microsoft Windows 98, 2000 또는 XP

표 24–1 클라이언트 시스템의 필수 하드웨어 및 소프트웨어 *(계속)*

구성 요소	설명
브라우저	■ Windows용 Microsoft Internet Explorer 버전 6 SP2 ■ Windows 2000 및 Windows 98용 Microsoft Internet Explorer 버전 6 SP1(2004년 12월 1일 당시의 최신 패치 포함)
Sun 소프트웨어	Sun Java 2 Runtime Environment, Standard Edition, 버전 1.4.2_03 이상(1.5는 아님)
인증서가 포함된 개인-공개 키	인증서가 포함된 하나 이상의 개인 및 공개 키 쌍. 인증서가 필요하며 표준 X.509 v3 형식이어야 합니다. S/MIME 기능을 사용하게 될 각 Communications Express Mail 사용자에 대해 CA로부터 키와 인증서를 얻습니다. 키와 해당 인증서는 클라이언트 시스템이나 스마트 카드에 저장됩니다. 공개 키와 인증서는 Directory Server가 액세스할 수 있는 LADP 디렉토리에도 저장됩니다. 키 인증서를 CA가 유지 관리하는 인증서 해지 목록(CRL)과 비교하여 키의 유효성을 추가로 확인하려면 CRL이 시스템의 일부여야 합니다. 743 페이지 “24.9.2 CRL에 대해 인증서 확인 시기”을 참조하십시오.
스마트 카드 소프트웨어(키와 인증서를 스마트 카드에 저장할 경우에만 필요함)	■ ActivCard Gold(현재 이름은 ActiveIdentity) 버전 2.1 또는 3.0 또는 ■ NetSign 버전 3.1
스마트 카드 판독기	클라이언트 시스템 및 스마트 카드 소프트웨어가 지원하는 스마트 카드 판독 장치 모델

표 24–2에는 서버 시스템의 필수 Sun Microsystems 소프트웨어가 나열되어 있습니다.

표 24–2 서버 시스템의 필수 소프트웨어

Sun 구성 요소	설명
메일 서버	Sun Java System Messaging Server 6 5 릴리스 이상(Solaris 버전 8 또는 9 및 Sun SPARC 시스템)
LDAP 서버	Sun Java System Directory Server 5 2004Q2 이상
Java	Java 2 Runtime Environment, Standard Edition, 버전 1.4.2 이상
Access Manager	(Schema 2에서 배포하는 경우) - Sun Java System Access Manager 6 2005Q1 및 Communications Express - Sun Java System Communications Express 6 2005Q1 이상

24.3 S/MIME 사용을 위한 요구 사항

Messaging Server를 설치한 후 Communications Express Mail 사용자가 서명 및 암호화 기능을 즉시 사용할 수 있는 것은 아닙니다. 사용자가 S/MIME을 사용할 수 있으려면 이 절에서 설명하는 요구 사항이 충족되어야 합니다.

24.3.1 개인 및 공개 키

S/MIME을 사용할 각 Communications Express Mail 사용자에게 표준 X.509 v3 형식의 인증서를 포함하여 하나 이상의 개인 및 공개 키 쌍을 발급해야 합니다. 확인 프로세스에 사용되는 인증서는 다른 메일 사용자에게 키가 실제로 키를 사용하는 사람에게 속해 있다는 것을 보장합니다. 사용자는 둘 이상의 키 쌍과 관련 인증서를 가질 수 있습니다.

키와 해당 인증서는 조직 내에서 발급 받거나 타사 공급업체에게 구입합니다. 키와 인증서 발급 방법에 상관없이 발급하는 조직을 인증 기관(CA)이라고 합니다.

키 쌍과 해당 인증서는 다음 두 가지 방법으로 저장됩니다.

- 스마트 카드에 저장

  이러한 카드는 일반 신용 카드와 비슷하며 신용 카드를 사용할 때와 마찬가지로 메일 사용자가 적절하게 보호해야 합니다. 스마트 카드에서 개인 키 정보를 읽으려면 메일 사용자의 컴퓨터(클라이언트 시스템)에 연결된 특수한 카드 판독기가 필요합니다. 자세한 내용은 720 페이지 “24.3.2 스마트 카드에 저장된 키”를 참조하십시오.

- 메일 사용자 컴퓨터(클라이언트 시스템)의 로컬 키 저장소에 저장

  메일 사용자의 브라우저에는 키 저장소가 있습니다. 또한 브라우저는 키 쌍과 인증서를 키 저장소로 다운로드하기 위한 명령도 제공합니다. 자세한 내용은 721 페이지 “24.3.3 클라이언트 시스템에 저장된 키”를 참조하십시오.

24.3.2 스마트 카드에 저장된 키

인증서를 포함하는 개인 공개 키 쌍을 스마트 카드에 저장할 경우 카드 판독기를 메일 사용자의 컴퓨터에 올바로 연결해야 합니다. 또한 카드 판독 장치에는 소프트웨어가 필요합니다. 카드 판독 장치와 해당 소프트웨어는 장비를 판매하는 공급업체가 제공합니다.

카드 판독 기능이 있는 시스템에는 사실상 두 부분이 있습니다. 한 부분은 하드웨어 카드 판독기와 해당 드라이버입니다. 두 번째 부분은 실제 카드입니다. 카드는 일반적으로 서로 다른 공급업체에서 제공하며 카드 판독을 위한 드라이버가 필요합니다. 모든 카드가 지원되는 것은 아닙니다. 지원되는 스마트 카드(ActiveCard(현재 이름은 ActiveIdentity) 및 NetSign) 목록은 표 24–1을 참조하십시오.

제대로 설치되었다면, 메일 사용자는 보내는 메시지에 대한 디지털 서명을 만들려 할 때 스마트 카드를 판독 장치에 삽입합니다. 스마트 카드 비밀번호가 확인되면 Communications Express Mail에서 메일에 서명하기 위해 개인 키에 액세스할 수 있습니다. 지원되는 스마트 카드와 판독 장치에 대한 자세한 내용은 718 페이지 “24.2 필수 소프트웨어 및 하드웨어 구성 요소”를 참조하십시오.

스마트 카드 공급업체가 제공하는 라이브러리가 사용자의 컴퓨터에 있어야 합니다. 자세한 내용은 739 페이지 “24.8 클라이언트 시스템의 키 액세스 라이브러리”를 참조하십시오.

24.3.3 클라이언트 시스템에 저장된 키

키 쌍과 인증서를 스마트 카드에 저장하지 않을 경우 메일 사용자의 컴퓨터(클라이언트 시스템)에 있는 로컬 키 저장소에 보관해야 합니다. 메일 사용자의 브라우저에서 키 저장소를 제공하고 키 쌍과 인증서를 키 저장소에 다운로드하기 위한 명령도 제공합니다. 브라우저에 따라 키 저장소를 비밀번호로 보호하기도 합니다.

로컬 키 저장소를 지원하려면 브라우저 공급업체가 제공하는 라이브러리가 사용자의 컴퓨터에 있어야 합니다. 자세한 내용은 739 페이지 “24.8 클라이언트 시스템의 키 액세스 라이브러리”를 참조하십시오.

24.3.4 LDAP 디렉토리에 공개 키 게시

모든 공개 키와 인증서는 Sun Java System Directory Server가 액세스할 수 있는 LDAP 디렉토리에도 저장해야 합니다. 이러한 작업을 S/MIME 메시지를 작성하는 다른 메일 사용자가 사용할 수 있도록 공개 키를 게시한다고 합니다.

보낸 사람과 받는 사람의 공개 키는 암호화된 메시지의 암호화/해독 프로세스에 사용됩니다. 공개 키 인증서는 디지털 서명에 사용된 공개 키를 검증하는 데 사용됩니다.

`ldapmodify`를 사용하여 공개 키와 인증서를 게시하는 방법은 749 페이지 “24.11 인증서 관리”를 참조하십시오.

24.3.5 메일 사용자에게 S/MIME 사용 권한 부여

서명된 또는 암호화된 메시지를 작성하려면 유효한 Communications Express Mail 사용자에게 그렇게 할 수 있는 권한이 있어야 합니다. 이 작업을 수행하려면 사용자의 LDAP 항목에 대한 `mailAllowedServiceAccess` 또는 `mailDomainAllowedServiceAccess` LDAP 속성을 사용합니다. 이러한 속성을 사용하여 개인 또는 도메인 단위로 S/MIME의 메일 사용자를 포함시키거나 제외시킬 수 있습니다.

자세한 내용은 748 페이지 “24.10 S/MIME 기능을 사용할 수 있는 권한 부여”를 참조하십시오.

24.3.6 여러 언어 지원

메일 메시지에 영어만 사용하는 Communications Express Mail 사용자는 라틴어가 아닌 언어(예: 중국어)의 문자를 포함하는 S/MIME 메시지를 읽지 못할 수 있습니다. 이러한 상황이 발생하는 한 가지 이유는 사용자 시스템에 설치된 JRE(Java 2 Runtime Environment)의 `/lib` 디렉토리에 `charsets.jar` 파일이 없기 때문입니다.

기본 JRE 설치 프로세스를 사용하여 영어 버전의 JRE를 다운로드한 경우 `charsets.jar` 파일이 설치되지 않습니다. 그러나 기본 설치의 다른 모든 언어 항목을 선택하면 `charsets.jar`가 설치됩니다.

charsets.jar 파일이 /lib 디렉토리에 설치되게 하려면 사용자 정의 설치를 사용하여 영어 버전의 JRE를 설치하라고 사용자에게 알립니다. 설치 중에 사용자가 "추가 언어 지원" 옵션을 선택해야 합니다.

24.4 Messaging Server 설치 후 시작

이 절에서는 S/MIME 애플릿이란 무엇이며 Communications Express Mail의 S/MIME 을 설정하는 기본 구성 절차에 대해 설명합니다. 구성 프로세스에는 S/MIME 애플릿의 매개 변수와 Messaging Server의 옵션을 설정하는 작업이 포함됩니다.

24.4.1 S/MIME 애플릿

개인 및 공개 키를 확인하는 여러 절차와 함께 메시지에 서명하거나, 메시지를 암호화하거나, 메시지를 해독하는 프로세스는 S/MIME 애플릿이라고 부르는 특수한 애플릿에 의해 처리 됩니다. S/MIME 기능을 구성하려면 smime.conf 파일의 매개 변수와 Messaging Server 옵션을 사용합니다. 그림 24–1은 S/MIME 애플릿과 다른 시스템 구성 요소와의 관계를 보여 줍니다.

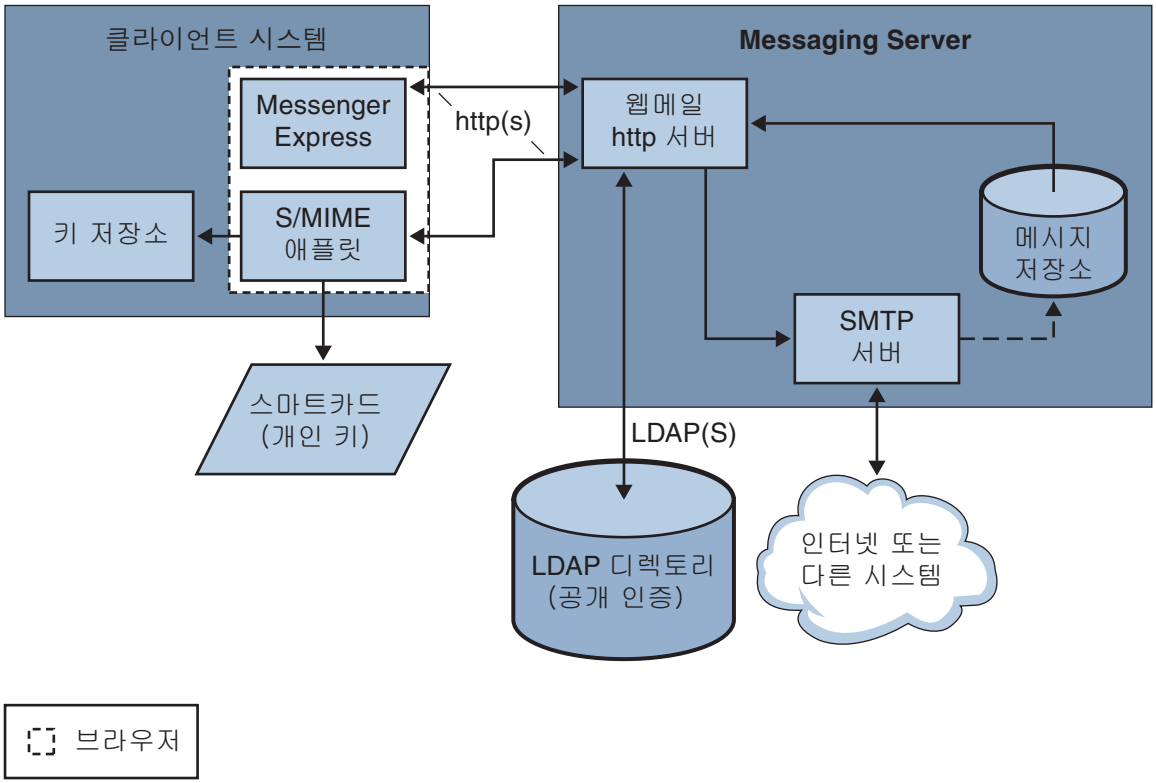


그림 24–1 S/MIME 애플릿

24.4.1.1 처음으로 로그인

S/MIME을 사용할 수 있는 권한을 가진 Communications Express Mail 사용자가 Messaging Server에 처음 로그인하면 S/MIME 애플릿에 대한 일련의 특수한 프롬프트 가 표시됩니다. 프롬프트에서 Yes 또는 Always를 선택하면 S/MIME 애플릿이 사용자의 컴퓨터에 다운로드됩니다. 사용자가 Communications Express Mail을 로그아웃할 때까지 S/MIME 애플릿이 컴퓨터에 남아 있습니다.

자세한 내용은 749 페이지 “24.11 인증서 관리”를 참조하십시오.

24.4.1.2 S/MIME 애플릿 다운로드

사용자 컴퓨터에서 JRE(Java 2 Runtime Environment)에 대한 캐싱이 활성화되어 있지 않은 경우 사용자가 Communications Express Mail에 로그인할 때마다 S/MIME 애플릿이 다운로드됩니다. 캐싱이 활성화되면 초기 다운로드 후에 S/MIME 애플릿의 복사본이 사용자의 시스템에 저장되므로 사용자가 로그인할 때마다 애플릿이 다운로드되지 않습니다.

캐싱은 성능을 향상시키므로 사용자에게 Java 2 Runtime Environment 버전 1.4.*x*에 대한 캐싱을 활성화하는 다음 단계를 수행하도록 지시할 수 있습니다.

▼ Java 2 Runtime Environment, 버전 1.4에 대한 캐싱을 활성화하는 방법

1. **Windows 제어판으로 이동합니다.**
2. **Java 플러그인 아이콘(Java 2 Runtime Environment)을 두 번 누릅니다.**
3. **캐시 탭을 누릅니다.**
4. **캐싱 사용 확인란을 선택합니다.**
5. **적용을 누릅니다.**

   S/MIME 애플릿이 다운로드된 후에도 사용자는 이 애플릿을 인식하지 못합니다. 이 애플릿에는 Communications Express Mail에서 메시지 서명, 암호화 또는 해독을 수행한다는 것이 표시됩니다. 또한 사용자는 오류 메시지가 나타나지 않는 한 개인 또는 공개 키 확인 프로세스를 인식하지 못합니다. 자세한 내용은 741 페이지 "24.9 개인 및 공개 키 확인"을 참조하십시오.

24.4.2 기본 S/MIME 구성

S/MIME의 구성 파일인 `smime.conf`에는 각 S/MIME 매개 변수에 대한 설명 주석과 예가 들어 있습니다. `smime.conf` 파일은 Messaging Server의 *msg-svr-base*`/config/` 디렉토리에 있습니다. 여기서 *msg-svr-base*는 Messaging Server가 설치된 디렉토리입니다.

다음 절차에는 S/MIME 기능을 구성하는 데 필요한 최소한의 단계가 포함되어 있습니다.

▼ S/MIME 구성 방법

1. **Messaging Server를 설치한 후 Communications Express Mail의 기본 기능이 작동 하고 있는지 확인합니다.**
2. **아직 없는 경우, S/MIME 기능을 사용할 권한이 있는 모든 메일 사용자에 대해 표준 X.509 v3 형식의 인증서와 함께 개인 공개 키 쌍을 만들거나 얻습니다.**
3. **키와 인증서를 위해 스마트 카드를 사용할 경우**

   a. **스마트 카드를 메일 사용자에게 배포합니다.**

   b. **Communications Express Mail에 액세스하는 각 클라이언트 시스템에 스마트 카드 판독 장치와 소프트웨어를 올바로 설치합니다.**

4 **브라우저의 로컬 키 저장소를 사용하여 키와 인증서를 저장할 경우 키 쌍과 인증서를 로컬 키 저장소에 다운로드하는 방법을 메일 사용자에게 알려줍니다.**

5 **스마트 카드나 로컬 키 저장소를 지원하려면 올바른 라이브러리가 클라이언트 시스템에 있어야 합니다.** **739 페이지 "24.8 클라이언트 시스템의 키 액세스 라이브러리"를 참조하십시오.**

6 **S/MIME을 지원하도록 LDAP 디렉토리를 설정합니다.**

a. **CA의 모든 인증서를 인증 기관의 고유 이름을 사용하여 Directory Server가 액세스할 수 있는 LDAP 디렉토리에 저장합니다. 이러한 인증서의 LDAP 속성은** `cacertificate;binary`**입니다. 인증서를 저장한 디렉토리 정보를 기록해 둡니다. 이후의 단계에서 이 정보가 필요합니다.**

LDAP 디렉토리 정보를 지정하는 예는 표 24–3의 `trustedurl`을, LDAP 디렉토리 검색에 대한 자세한 내용은 749 페이지 "24.11 인증서 관리"를 참조하십시오.

b. **Directory Server가 액세스할 수 있는 LDAP 디렉토리에 공개 키와 인증서를 저장합니다. 공개 키와 인증서에 대한 LDAP 속성은** `usercertificate;binary`**입니다. 인증서를 저장한 디렉토리 정보를 기록해 둡니다. 이후의 단계에서 이 정보가 필요합니다.**

LDAP 디렉토리 정보를 지정하는 예는 표 24–3의 `certurl`을, LDAP 디렉토리 검색에 대한 자세한 내용은 749 페이지 "24.11 인증서 관리"를 참조하십시오.

c. **S/MIME 메시지를 주고 받는 모든 사용자에게 자신의 사용자 항목에서 LDAP 필터와 함께 S/MIME을 사용할 수 있는 권한이 주어졌는지 확인합니다. 필터는** `mailAllowedServiceAccess` **또는** `mailDomainAllowedServiceAccess` **LDAP 속성을 사용하여 정의합니다.**

주: 기본적으로 `mailAllowedServiceAccess` 또는 `mailDomainAllowedServiceAccess`를 사용하지 않은 경우 `smime`를 비롯한 모든 서비스가 허용됩니다. 이러한 속성에 서비스를 명시적으로 지정할 경우 메일 사용자에게 S/MIME 기능을 사용할 수 있는 권한을 제공하려면 `smime`뿐만 아니라 `http` 및 `smtp` 서비스도 지정해야 합니다.

자세한 내용은 748 페이지 "24.10 S/MIME 기능을 사용할 수 있는 권한 부여"를 참조하십시오.

7 **사용 가능한 텍스트 편집기를 사용하여** `smime.conf` **파일을 편집합니다. 매개 변수 구문은 이 파일의 시작 부분에 있는 주석을 참조하십시오.**

`smime.conf`의 모든 텍스트와 매개 변수 예는 주석 문자(#)로 시작됩니다. 필요한 매개 변수를 `smime.conf`에 추가하거나 매개 변수 예를 파일의 다른 부분에 복사하고 해당 값을 변경할 수 있습니다. 예를 복사하여 편집할 경우 행의 시작 부분에 있는 # 문자를 제거해야 합니다.

다음 매개 변수를 각각 하나의 행으로 파일에 추가합니다.

a. `trustedurl`(표 24–3 참조) -- **CA의 인증서를 찾기 위한 LDAP 디렉토리 정보로 설정합니다. 단계 a에서 저장한 정보를 사용합니다.**

b. `certurl`(표 24–3 참조) -- **공개 키와 인증서를 찾기 위한 LDAP 디렉토리 정보로 설정합니다. 단계 b에서 저장한 정보를 사용합니다.**

c. `usersertfilter`(표 24–3 참조) -- `smime.conf` **파일의 값 예로 설정합니다. 대부분의 경우 값 예를 사용하면 됩니다. 예를 복사하고 행의 시작 부분에서 # 문자를 삭제합니다.**

이 매개 변수는 키 쌍을 다른 메일 주소에 할당할 때 사용자의 모든 개인 공개 키 쌍을 찾을 수 있도록 Communications Express Mail 사용자의 주, 대체 및 이와 동일한 전자 메일 주소에 대한 필터 정의를 지정합니다.

d. `sslrootcacertsurl`(표 24–3 참조) -- **S/MIME 애플릿과 Messaging Server 사이의 통신 연결에 SSL을 사용하는 경우 Messaging Server의 SSL 인증서를 확인하는 데 사용되는 CA의 인증서를 찾기 위해 LDAP 디렉토리 정보와 함께** `sslrootcacertsurl`**을 설정합니다. 자세한 내용은 738 페이지 "24.7 SSL을 사용하여 인터넷 연결 보안"을 참조하십시오.**

`checkoverssl`(표 24–3 참조) -- S/MIME 애플릿과 Messaging Server 사이의 통신 연결에 SSL을 사용하지 않을 경우 `0`으로 설정합니다.

e. `crlenable`(표 24–3 참조) -- **CRL 확인을 수행하려면** `smime.conf` **파일에 다른 매개 변수를 추가해야 하므로 지금은** `0`**으로 설정하여 CRL 확인을 비활성화합니다.**

f. `logindn` **및** `loginpw`(표 24–3 참조) -- **공개 키 및 CA 인증서가 포함된 LDAP 디렉토리에 액세스하기 위해 인증이 필요한 경우 이러한 매개 변수를 읽기 권한을 가진 LDAP 항목의 고유 이름과 비밀번호로 설정합니다.**

주:`crlmappingurl`, `sslrootcacertsurl` 또는 `trustedurl` 매개 변수에 지정된 LDAP 정보를 사용하여 LDAP 디렉토리에 액세스할 때마다 `logindn` 및 `loginpw`의 값이 사용됩니다. 자세한 내용은 730 페이지 "24.5 smime.conf 파일의 매개 변수" 및 728 페이지 "24.4.3 자격 증명을 사용하여 LDAP에서 공개 키, CA 인증서 및 CRL 액세스"를 참조하십시오.

LDAP 디렉토리에 액세스하는 데 인증이 필요하지 않은 경우 `logindn` 및 `loginpw`를 설정하지 마십시오.

8 `configutil`**을 사용하여 Messaging Server 옵션을 설정합니다.**

a. `local.webmail.smime.enable` -- `1`**로 설정합니다.**

b. `local.webmail.cert.enable` -- **CRL에 대해 인증서를 확인하려는 경우** `1`**로 설정합니다.**

자세한 내용은 737 페이지 "24.6 Messaging Server 옵션"을 참조하십시오.

9 이제 Communications Express Mail이 S/MIME 기능을 사용하도록 구성되었습니다. 다음 단계를 수행하여 S/MIME 기능이 작동하는지 확인합니다.

a. Messaging Server를 다시 시작합니다.

b. Messaging Server 로그 파일 *msg-svr-base* /log/http에서 S/MIME과 관련된 진단 메시지를 확인합니다.

c. S/MIME에 대한 문제가 감지된 경우 진단 메시지를 통해 구성 매개 변수의 문제를 수정하는 방법을 확인할 수 있습니다.

d. 필요한 구성 매개 변수를 수정합니다.

e. Messaging Server의 로그 파일에 S/MIME에 대한 진단 메시지가 더 이상 존재하지 않을 때까지 단계 a - d까지 반복합니다.

f. 다음 단계를 수행하여 S/MIME 기능이 작동하는지 확인합니다.

i. 클라이언트 시스템에서 Messaging Server에 로그인합니다. S/MIME 애플릿에 대한 특수 프롬프트에 Yes 또는 Always로 대답합니다. 749 페이지 "24.11 인증서 관리"를 참조하십시오.

ii. 자신에게 보내는 짧은 메시지를 작성합니다.

iii. 작성 창의 맨 아래에서 암호화 확인란을 선택하여(선택되어 있지 않은 경우) 메시지를 암호화합니다.

iv. 보내기를 눌러 암호화된 메시지를 자신에게 보냅니다. 이때 키와 인증서 기법이 대부분 작동해야 합니다.

v. 암호화된 메시지에 문제가 있을 경우 대개 smime.conf 파일의 LDAP 디렉토리 정보에 사용한 값이나 LDAP 디렉토리에 키와 인증서가 저장된 방법에 문제의 원인이 있을 수 있습니다. Messaging Server 로그에서 추가 진단 메시지를 확인합니다.

아래 표에 요약되어 있는 나머지 S/MIME 매개 변수는 S/MIME 환경을 추가로 구성하는 데 사용할 수 있는 여러 옵션을 제공합니다. 매개 변수에 대한 자세한 내용은 730 페이지 "24.5 smime.conf 파일의 매개 변수"를 참조하십시오.

S/MIME의 필수 매개 변수	스마트 카드 및 로컬 키 저장소를 위한 매개 변수	CRL 확인을 위한 매개 변수	초기 설정 및 보안 연결을 위한 매개 변수
certurl*	-p pattern	checkoverssl	alwaysencrypt

S/MIME의 필수 매개 변수	스마트 카드 및 로컬 키 저장소를 위한 매개 변수	CRL 확인을 위한 매개 변수	초기 설정 및 보안 연결을 위한 매개 변수
logindn		crlaccessfail	alwayssign
loginpw		crldir	sslrootcacertsurl
trustedurl*		crlenable	
usercertfilter*		crlmappingurl	
		crlurllogindn	
		crlurlloginpw	
		crlusepastnextupdate	
		readsigncert	
		revocationunknown	
		sendencryptcert	
		sendencryptcertrevoked	
		readsigncert	
		sendsigncertrevoked	
		timestampdelta	

* 이러한 매개 변수에는 기본값이 없기 때문에 값을 지정해야 합니다.

24.4.3 자격 증명을 사용하여 LDAP에서 공개 키, CA 인증서 및 CRL 액세스

S/MIME에 필요한 공개 키, CA 인증서 및 CRL을 LDAP 디렉토리에 저장할 수 있습니다(앞의 절 참조). 키, 인증서 및 CRL은 LDAP의 단일 URL 또는 여러 URL에서 액세스할 수 있습니다. 예를 들어, CRL을 하나의 URL에 저장하고 공개 키와 인증서를 다른 URL에 저장할 수 있습니다. Messaging Server에서는 원하는 CRL이나 인증서 정보를 포함하는 URL과 이러한 URL에 액세스할 수 있는 항목의 DN 및 비밀번호를 지정할 수 있습니다. 이러한 DN/비밀번호 자격 증명은 선택 사항입니다. 아무 것도 지정하지 않을 경우 먼저 HTTP 서버 자격 증명으로 LDAP 액세스를 시도하고 이것이 실패할 경우 `anonymous`로 액세스를 시도합니다.

두 쌍의 `smime.conf` 자격 증명 매개 변수, 즉 `logindn` 및 `loginpw`와 `crlurllogindn` 및 `crlurlloginpw`를 설정하여 원하는 URL에 액세스할 수 있습니다.

`logindn` 및 `loginpw`는 `smime.conf`의 모든 URL에 사용되는 자격 증명입니다. 이러한 매개 변수는 `certurl` 및 `trustedurl` 매개 변수에 지정된 공개 키, 해당 인증서 및 CA 인증서에 대한 읽기 권한이 있는 LDAP 항목의 DN과 비밀번호를 지정합니다.

crlurllogindn 및 crlurlloginpw는 매핑 테이블의 결과 URL에 대한 읽기 권한이 있는 LDAP 항목의 DN과 비밀번호를 지정합니다(자세한 내용은 743 페이지 "24.9.3 CRL 액세스" 참조). 이러한 자격 증명이 허용되지 않을 경우 LDAP 액세스가 거부되며 다른 자격 증명으로 다시 시도되지 않습니다. 두 매개 변수를 모두 지정하거나 둘 다 비워두어야 합니다. 이러한 매개 변수는 인증서로부터 직접 가져온 URL에는 적용되지 않습니다.

24.4.3.1 특정 URL의 비밀번호 설정

Messaging Server에서는 certUrl , trustedUrl, crlmappingUrl, sslrootcacertsUrl과 같은 smime.conf URL에 액세스하기 위한 DN/비밀번호 쌍을 명시적으로 정의할 수 있습니다.

구문은 다음과 같습니다.

url_type URL[|*URL_DN* | *URL_password*]

예:

```
trustedurl==ldap://mail.siroe.com:389/cn=Directory Manager, ou=people,
o=siroe.com,o=ugroot?cacertificate?sub?(objectclass=certificationauthority) |
cn=Directory manager | boomshakalaka
```

24.4.3.2 LDAP 자격 증명 사용 요약

이 절에서는 LDAP 자격 증명의 사용에 대해 요약합니다.

- 모든 LDAP 자격 증명은 선택 사항입니다. 아무 것도 지정하지 않을 경우 먼저 HTTP 서버 자격 증명으로 LDAP 액세스를 시도하고 이것이 실패할 경우 anonymous 액세스를 시도합니다.

  두 쌍의 smime.conf 매개 변수를 지정할 수 있는 두 개의 URL 집합에 대한 자격 증명으로 사용합니다.

  logindn 및 loginpw - smime.conf의 모든 URL

  crlurllogindn 및 crlurlloginpw - 매핑 테이블의 모든 URL

  이것들을 기본 LDAP 자격 증명 쌍이라고 합니다.
- smime.conf 또는 매핑 CRL URL을 통해 지정된 모든 URL에 선택적 로컬 LDAP 자격 증명 쌍을 지정할 수 있습니다.
- 자격 증명은 지정된 순서대로 확인됩니다.

  로컬 LDAP 자격 증명 쌍 - 지정된 경우 한 번만 시도됩니다.

  기본 LDAP 자격 증명 쌍 - 지정된 경우 로컬 LDAP 자격 증명 쌍이 없으면 한 번만 시도됩니다.

  서버 - 로컬 LDAP 자격 증명 쌍과 기본 LDAP 자격 증명 쌍이 모두 지정되지 않은 경우 처음으로 시도됩니다.

  4) anonymous - 서버가 실패하거나 아무 것도 지정하지 않은 경우에만 마지막에 시도됩니다.

- URL에 로컬 LDAP 자격 증명 쌍이 지정된 경우 이 자격 증명 쌍이 처음에 사용됩니다. 액세스에 실패하면 액세스가 거부됩니다.
- URL에 로컬 LDAP 자격 증명 쌍이 지정되지 않은 경우 해당하는 기본 LDAP 자격 증명 쌍이 사용됩니다. 액세스에 실패하면 액세스가 거부됩니다.

24.5 smime.conf 파일의 매개 변수

`smime.conf` 파일은 Messaging Server의 *msg-svr-base*`/config/` 디렉토리에 있습니다. 여기서 *msg-svr-base*는 Messaging Server가 설치된 디렉토리입니다. 이 파일의 모든 텍스트 및 매개 변수 예는 주석 문자(#)로 시작됩니다.

매개 변수를 원하는 값과 함께 `smime.conf`에 추가하거나 매개 변수 예를 편집할 수 있습니다. 예를 사용할 경우 파일의 다른 부분에 예를 복사하고 매개 변수의 값을 편집한 다음 행의 시작 부분에 있는 # 문자를 제거합니다.

Messaging Server를 설치한 후에 사용 가능한 텍스트 편집기를 사용하여 `smime.conf`를 편집합니다. 표 24–3에 설명된 매개 변수는 대소문자를 구분하지 않으며 다른 언급이 없는 한 설정할 필요가 없습니다.

표 24–3 smime.conf 파일의 S/MIME 구성 매개 변수

매개 변수	용도
alwaysencrypt	S/MIME을 사용할 권한이 있는 모든 Communications Express Mail 사용자에 대해 모든 보내는 메시지를 자동으로 암호화할 것인지 여부에 대한 초기 설정을 제어합니다. 각 Communications Express Mail 사용자는 표 24–5에 설명된 확인란을 사용하여 자신의 메시지에 대해 이 매개 변수 값을 무시할 수 있습니다. 다음 값 중 하나를 선택합니다. `0` - 메시지를 암호화하지 않습니다. Communications Express Mail 내의 암호화 확인란이 선택되지 않은 상태로 표시됩니다. 기본값입니다. `1` - 메시지를 항상 암호화합니다. Communications Express Mail 내의 암호화 확인란이 선택된 상태로 표시됩니다. 예: `alwaysencrypt==1`

표 24–3 smime.conf 파일의 S/MIME 구성 매개 변수 *(계속)*

매개 변수	용도
alwayssign	S/MIME을 사용할 권한이 있는 모든 Communications Express Mail 사용자에 대해 모든 보내는 메시지를 자동으로 서명할 것인지 여부에 대한 초기 설정을 제어합니다. 각 Communications Express Mail 사용자는 표 24–5에 설명된 확인란을 사용하여 자신의 메시지에 대해 이 매개 변수 값을 무시할 수 있습니다. 다음 값 중 하나를 선택합니다. `0` - 메시지에 서명하지 않습니다. Communications Express Mail 내의 서명 확인란이 선택되지 않은 상태로 표시됩니다. 기본값입니다. `1` - 메시지에 항상 서명합니다. Communications Express Mail 내의 서명 확인란이 선택된 상태로 표시됩니다. 예: `alwaysensign==1`
certurl	Communications Express Mail 사용자의 공개 키와 인증서를 찾기 위한 LDAP 디렉토리 정보를 지정합니다(공개 키의 LDAP 속성은 `usercertificate;binary`임). 인증서에 대한 자세한 내용은 749 페이지 “24.11 인증서 관리”를 참조하십시오. 이 매개 변수는 Messaging Server에 의해 서비스되는 모든 사용자가 포함된 LDAP 디렉 토리 정보 트리(DIT)의 사용자/그룹에서 최상위 노드를 가리켜야 합니다. 이러한 점은 특히 도메인이 둘 이상인 사이트의 경우 중요합니다. 고유 이름이 단일 도메인의 사용자를 포함하는 하위 트리가 아니라 사용자/그룹 트리의 루트 고유 이름이어야 합니다. 이 매개 변수는 반드시 설정해야 하는 필수 매개 변수입니다. 예: `certurl==ldap://mail.siroe.com:389/ou=people,o=siroe.com,o=ugroot`
checkoverssl	CRL에 대해 키의 인증서를 확인할 때 SSL 통신 연결을 사용하는지 여부를 제어합니다. 자세한 내용은 738 페이지 “24.7 SSL을 사용하여 인터넷 연결 보안”을 참조하십시오. 다음 값 중 하나를 선택합니다. `0` - SSL 통신 연결을 사용하지 않습니다. `1` - SSL 통신 연결을 사용합니다. 기본값입니다. CRL 확인이 유효한 상태에서 프록시 서버를 사용하면 문제가 발생할 수 있습니다. 745 페이지 “24.9.4 프록시 서버 및 CRL 확인”을 참조하십시오.

표 24–3 smime.conf 파일의 S/MIME 구성 매개 변수 (계속)

매개 변수	용도
crlaccessfail	Messaging Server가 CRL 액세스를 여러 번 시도했다가 실패한 후에 CRL 액세스를 다시 시도하기까지 기다리는 시간을 지정합니다. 이 매개 변수에는 기본값이 없습니다. **구문:** `crlaccessfail==`*number_of_failures* :*time_period_for_failures*: *wait_time_before_retry* 여기서 *number_of_failures*는 Messaging Server가 *time_period_for_failures*에 지정된 시간 간격 동안 CRL 액세스에 실패할 수 있는 횟수입니다. 값이 0보다 커야 합니다. *time_period_for_failures*는 Messaging Server가 CRL 액세스 시도에 실패한 횟수를 계산하는 시간(초)입니다. 값이 0보다 커야 합니다. *wait_time_before_retry*는 Messaging Server가 지정된 간격 동안 제한된 실패 횟수에 도달한 후에 CRL에 다시 액세스하려고 시도하기까지 대기하는 시간(초)입니다. 값이 0보다 커야 합니다. 예: `crlaccessfail==10:60:300` 이 예에서 Messaging Server는 1분 동안 10번까지 CRL 액세스에 실패할 수 있습니다. 그런 다음 5분을 기다렸다가 CRL 액세스를 다시 시도합니다. 747 페이지 “24.9.7 CRL 액세스 문제”를 참조하십시오.
crldir	Messaging Server가 CRL을 디스크에 다운로드하는 디렉토리 정보를 지정합니다. 기본값은 *msg-svr-base*`/data/store/mboxlist` 입니다. 여기서 *msg-svr-base*는 Messaging Server가 설치된 디렉토리입니다. 자세한 내용은 745 페이지 “24.9.5 오래된 CRL 사용”을 참조하십시오.
crlenable	CRL에 대해 인증서를 확인하는지 여부를 제어합니다. CRL에 일치하는 것이 있을 경우 인증서가 해지된 것입니다. `smime.conf` 파일의 `send*revoked` 매개 변수 값은 해지된 인증서가 있는 키를 Communications Express Mail이 거부하는지 아니면 사용하는지 여부를 지정합니다. 자세한 내용은 741 페이지 “24.9 개인 및 공개 키 확인”을 참조하십시오. 다음 값 중 하나를 선택합니다. `0`- 각 인증서를 CRL에 대해 확인하지 않습니다. `1`- 각 인증서를 CRL에 대해 확인합니다. 기본값입니다. Messaging Server의 `local.webmail.cert.enable` 옵션이 `1`로 설정되었는지 확인합니다. 그렇지 않으면 `crlenable`을 `1`로 설정하는 경우에도 CRL 확인이 수행됩니다.

표 24–3 smime.conf 파일의 S/MIME 구성 매개 변수 *(계속)*

매개 변수	용도
crlmappingurl	CRL 매핑 정의를 찾기 위한 LDAP 디렉토리 정보를 지정합니다. 이 매개 변수는 매핑 정의가 있는 경우에만 필요합니다. 자세한 내용은 743 페이지 "24.9.3 CRL 액세스"를 참조하십시오. 이 매개 변수에는 기본값이 없습니다. 또한 URL에 액세스할 수 있는 DN과 비밀번호를 추가할 수도 있습니다. 구문: `crlmappingurl` *URL*[|*URL_DN*|*URL_password*] 예: `crlmappingurl==ldap://mail.siroe.com:389/cn=XYZ Messaging, ou=people, o=mail.siroe.com,o=isp?msgCRLMappingRecord?sub?( objectclass=msgCRLMappingTable) | cn=Directory Manager | pAsSwOrD`
crlurllogindn	CRL 매핑 정의에 대한 읽기 권한을 가진 LDAP 항목의 고유 이름을 지정합니다. 항목을 인증서에서 직접 가져올 경우에는 지정하지 않습니다. 자세한 내용은 904페이지의 "CRL 액세스"를 참조하십시오. `crllogindn` 및 `crlloginpw`의 값을 지정하지 않을 경우 Messaging Server는 HTTP 서버의 로그인 값을 사용하여 LDAP 디렉토리에 액세스합니다. 액세스에 실패할 경우 Messaging Server는 LDAP 디렉토리에 익명으로 액세스를 시도합니다. 예: `crllogindn==cn=Directory Manager`
crlurlloginpw	`crllogindn` 매개 변수의 고유 이름에 대한 비밀번호를 ASCII 텍스트로 지정합니다. `crllogindn` 및 `crlloginpw`의 값을 지정하지 않을 경우 Messaging Server는 HTTP 서버의 로그인 값을 사용하여 LDAP 디렉토리에 액세스합니다. 액세스에 실패할 경우 Messaging Server는 LDAP 디렉토리에 익명으로 액세스를 시도합니다. 예: `crlloginpw==zippy`
crlusepastnextupdate	현재 날짜가 CRL의 next-update 필드에 지정된 날짜 이후인 경우에 CRL을 사용할지 여부를 제어합니다. 자세한 내용은 745 페이지 "24.9.5 오래된 CRL 사용"을 참조하십시오. 다음 값 중 하나를 선택합니다. `0` - 오래된 CRL을 사용하지 않습니다. `1` - 오래된 CRL을 사용합니다. 기본값입니다.

표 24–3 smime.conf 파일의 S/MIME 구성 매개 변수 (계속)

매개 변수	용도
logindn	certurl 및 trustedurl 매개 변수에 지정된 LDAP 디렉토리에 있는 공개 키 및 해당 인증서와 CA 인증서에 대한 읽기 권한을 가진 LDAP 항목의 고유 이름을 지정합니다. logindn 및 loginpw의 값을 지정하지 않을 경우 Messaging Server는 HTTP 서버의 로그인 값을 사용하여 LDAP 디렉토리에 액세스합니다. 액세스에 실패할 경우 Messaging Server는 LDAP 디렉토리에 익명으로 액세스를 시도합니다. 예: logindn==cn=Directory Manager
loginpw	logindn 매개 변수의 고유 이름에 대한 비밀번호를 ASCII 텍스트로 지정합니다. logindn 및 loginpw의 값을 지정하지 않을 경우 Messaging Server는 HTTP 서버의 로그인 값을 사용하여 LDAP 디렉토리에 액세스합니다. 액세스에 실패할 경우 Messaging Server는 LDAP 디렉토리에 익명으로 액세스를 시도합니다. 예: loginpw==SkyKing
platformwin	Windows 플랫폼에서 스마트 카드나 로컬 키 저장소를 사용할 때 필요한 하나 이상의 라이브러리 이름을 지정합니다. 클라이언트 시스템에서 기본값이 작동하지 않을 경우에만 이 매개 변수를 변경합니다. 기본값은 다음과 같습니다. platformwin==CAPI:library=capibridge.dll; 자세한 내용은 739 페이지 “24.8 클라이언트 시스템의 키 액세스 라이브러리”를 참조하십시오.
readsigncert	메시지를 읽을 때 S/MIME 디지털 서명을 확인하기 위해 공개 키의 인증서를 CRL에 대해 확인할지 여부를 제어합니다. 메시지의 디지털 서명을 만들기 위해 개인 키가 사용되지만 개인 키를 CRL에 대해 확인할 수 없습니다. 따라서 개인 키와 연관된 공개 키의 인증서가 CRL에 대해 확인됩니다. 741 페이지 “24.9 개인 및 공개 키 확인”을 참조하십시오. 다음 값 중 하나를 선택합니다. 0 - 인증서를 CRL에 대해 확인하지 않습니다. 1 - 인증서를 CRL에 대해 확인합니다. 기본값입니다.
revocationunknown	인증서를 CRL에 대해 확인할 때 모호한 상태가 반환될 경우 수행할 작업을 지정합니다. 이 경우에 인증서가 유효한지 아니면 해지된 상태인지는 확실하지 않습니다. 자세한 내용은 741 페이지 “24.9 개인 및 공개 키 확인”을 참조하십시오. 다음 값 중 하나를 선택합니다. ok - 인증서를 유효한 것으로 간주합니다. revoked - 인증서를 해지된 것으로 간주합니다. 기본값입니다.

표 24–3 smime.conf 파일의 S/MIME 구성 매개 변수 (계속)

매개 변수	용도
sendencryptcert	보내는 메시지를 암호화하는 데 사용되는 공개 키의 인증서를 사용 전에 CRL에 대해 확인하는지 여부를 제어합니다. 741 페이지 “24.9 개인 및 공개 키 확인”을 참조하십시오. 다음 값 중 하나를 선택합니다. `0` - 인증서를 CRL에 대해 확인하지 않습니다. `1` - 인증서를 CRL에 대해 확인합니다. 기본값입니다.
sendencryptcertrevoked	보내는 메시지를 암호화하는 데 사용되는 공개 키의 인증서가 해지된 경우 수행할 작업을 지정합니다. 자세한 내용은 741 페이지 “24.9 개인 및 공개 키 확인”을 참조하십시오. 다음 값 중 하나를 선택합니다. `allow` - 공개 키를 사용합니다. `disallow` - 공개 키를 사용하지 않습니다. 기본값입니다.
sendsigncert	개인 키를 사용하여 보내는 메시지의 디지털 서명을 만들 수 있는지 여부를 결정하기 위해 공개 키의 인증서를 CRL에 대해 확인하는지 여부를 제어합니다. 디지털 서명에 대해 개인 키가 사용되지만 개인 키를 CRL에 대해 확인할 수 없습니다. 따라서 개인 키와 연관된 공개 키의 인증서를 CRL에 대해 확인합니다. 자세한 내용은 741 페이지 “24.9 개인 및 공개 키 확인”을 참조하십시오. 다음 값 중 하나를 선택합니다. `0` - 인증서를 CRL에 대해 확인하지 않습니다. `1` - 인증서를 CRL에 대해 확인합니다. 기본값입니다.
sendsigncertrevoked	개인 키가 해지된 상태인 것으로 확인되었을 때 수행할 작업을 결정합니다. 메시지의 디지털 서명을 만들 때 개인 키가 사용되지만 개인 키를 CRL에 대해 확인할 수는 없습니다. 따라서 개인 키와 연관된 공개 키의 인증서를 CRL에 대해 확인합니다. 공개 키 인증서가 해지되면 해당 개인 키도 해지됩니다. 자세한 내용은 741 페이지 “24.9 개인 및 공개 키 확인”을 참조하십시오. 다음 값 중 하나를 선택합니다. `allow` -해지된 상태의 개인 키를 사용합니다. `disallow` - 해지된 상태의 개인 키를 사용하지 않습니다. 기본값입니다.

표 24–3 smime.conf 파일의 S/MIME 구성 매개 변수 (계속)

매개 변수	용도
sslrootcacertsurl	Messaging Server의 SSL 인증서를 확인하는 데 사용되는 유효한 CA의 인증서를 찾기 위한 고유 이름과 LDAP 디렉토리 정보를 지정합니다. Messaging Server에서 SSL이 활성화된 경우 이 매개 변수를 반드시 지정해야 합니다. 자세한 내용은 738 페이지 "24.7 SSL을 사용하여 인터넷 연결 보안"을 참조하십시오. 클라이언트 응용 프로그램에서 모든 요청을 수신하는 프록시 서버에 대한 SSL 인증서가 있을 경우 이러한 SSL 인증서에 대한 CA 인증서도 이 매개 변수가 가리키는 LDAP 디렉 토리에 있습니다. 또한 이 URL에 액세스할 수 있는 DN과 비밀번호를 추가할 수도 있습니다. 구문: `crlmappingurl` *URL*[|*URL_DN*|*URL_password*] 예: `sslrootcacertsurl==ldap://mail.siroe.com:389/cn=SSL Root CA Certs,ou=people,o=siroe.com,o=isp? cacertificate;binary?base? (objectclass=certificationauthority)|cn=Directory Manager | pAsSwOrD`
timestampdelta	공개 키의 인증서를 CRL에 대해 확인할 때 메시지의 보낸 시간이나 받은 시간을 사용할지를 결정하는 데 사용되는 시간 간격(초)을 지정합니다. 매개 변수 기본값인 0은 Communications Express Mail에서 항상 받은 시간을 사용하도록 지시합니다. 자세한 내용은 746 페이지 "24.9.6 사용할 메시지 시간 지정"을 참조하십시오. 예: `timestampdelta==360`
trustedurl	유효한 CA의 인증서를 찾기 위한 고유 이름과 LDAP 디렉토리 정보를 지정합니다. 필수 매개 변수입니다. 또한 이 URL에 액세스할 수 있는 DN과 비밀번호를 추가할 수도 있습니다. 구문: `crlmappingurl` *URL*[|*URL_DN*|*URL_password*] 예: `trustedurl==ldap://mail.siroe.com:389/cn=Directory Manager, ou=people, o=siroe.com,o=ugroot?cacertificate?sub? (objectclass=certificationauthority)|cn=Directory Manager | pAsSwOrD`

표 24–3 smime.conf 파일의 S/MIME 구성 매개 변수 *(계속)*

매개 변수	용도
usercertfilter	키 쌍을 다른 메일 주소에 할당할 때 사용자의 모든 개인 공개 키 쌍을 찾을 수 있도록 Communications Express Mail 사용자의 주, 대체 및 이와 동일한 전자 메일 주소에 대한 필터 정의를 지정합니다. 이 매개 변수는 필수이며 기본값이 없습니다.

24.6 Messaging Server 옵션

S/MIME에 적용되는 세 가지 Messaging Server 옵션을 설정하려면 Messaging Server가 설치된 시스템에서 다음 작업을 수행합니다.

▼ S/MIME에 적용되는 Messaging Server 옵션을 설정하는 방법

1 루트로 로그인하여 다음을 입력합니다.

```
# cd msg-svr-base/sbin
```

여기서 *msg-svr-base*는 Messaging Server가 설치된 디렉토리입니다.

2 다음 표에 설명된 Messaging Server 옵션을 시스템에 맞게 설정합니다. `configutil` **유틸리티를 사용하여 이러한 옵션을 설정합니다. 별도의 언급이 없을 경우 옵션을 설정할 필요가 없습니다.**

매개 변수	용도
`local.webmail.cert.enable`	CRL 확인을 처리하는 프로세스가 CRL 확인을 수행해야 하는지 여부를 제어합니다. `0` - 프로세스가 인증서를 CRL에 대해 확인하지 않습니다. 기본값입니다. `1` - 프로세스가 인증서를 CRL에 대해 확인합니다. `1`로 설정할 경우 `smime.conf` 파일의 `crlenable` 매개 변수가 `1`로 설정되었는지 확인합니다.
`local.webmail.cert.port`	CRL 통신에 사용할 Messaging Server가 실행되는 시스템의 포트 번호를 지정합니다. 이 포트는 해당 시스템에서 대해서만 로컬로 사용됩니다. 값은 1024보다 커야 하며 기본값은 55443입니다. 기본 포트 번호가 이미 사용 중이면 이 옵션을 반드시 지정해야 합니다.

매개 변수	용도
`local.webmail.smime.enable`	Communications Express Mail 사용자가 S/MIME 기능을 사용할 수 있는지 여부를 제어합니다. 다음 값 중 하나를 선택합니다. `0` - 시스템이 올바른 소프트웨어 및 하드웨어 구성 요소로 구성된 경우에도 Communications Express Mail 사용자가 S/MIME 기능을 사용할 수 없습니다. 기본값입니다. `1` - 사용 권한이 있는 Communications Express Mail 사용자가 S/MIME 기능을 사용할 수 있습니다. 예: `configutil -o local.webmail.smime.enable -v 1`

24.7 SSL을 사용하여 인터넷 연결 보안

다음 표에 요약된 것처럼 Messaging Server에서는 Communications Express Mail에 영향을 주는 인터넷 연결에 SSL(Secure Socket Layer)을 사용할 수 있습니다.

대상 연결	설명
Messaging Server 및 Communications Express Mail 간의 연결	SSL을 사용하여 이 연결을 보안하려면 Messaging Server에 대한 관리 작업이 필요합니다. Communications Express Mail 사용자는 브라우저에서 Messaging Server에 대한 URL 정보를 입력할 때 HTTP가 아니라 HTTPS 프로토콜을 사용해야 합니다. 738 페이지 "24.7.1 Messaging Server 및 Communications Express Mail 간의 연결 보안"을 참조하십시오.
Messaging Server 및 S/MIME 애플릿 간 연결	공개 키 인증서를 CRL에 대해 확인할 경우 S/MIME 애플릿이 Messaging Server와 직접 통신해야 합니다. SSL을 사용하여 이 연결을 보안하려면 `smime.conf` 파일에서 `sslrootcacertsurl` 및 `checkoverssl`을 설정하는 것 외에도 Messaging Server에 대한 관리 작업이 필요합니다. 739 페이지 "24.7.2 Messaging Server 및 S/MIME 애플릿 간의 연결 보안"을 참조하십시오.

24.7.1 Messaging Server 및 Communications Express Mail 간의 연결 보안

Messaging Server에서는 Messaging Server와 Communications Express Mail 간의 인터넷 연결에 SSL(Secure Socket Layer)을 사용할 수 있습니다. Messaging Server에서 SSL을 설정한 후 Communications Express에서 SSL을 구성합니다. **Sun Java System Communications Express 6.3 관리 설명서**를 참조하십시오. Communications Express Mail 사용자는 브라우저에서 HTTPS 프로토콜을 사용하여 Communications Express URL을 지정합니다.

HTTPS://*hostname*.*domain*:*secured_port*

즉, HTTP 프로토콜(HTTP://*hostname* .*domain*: *unsecure_port*)을 사용하지 않습니다. Communications Express 로그인 창이 표시되면 창 아래의 잠금 위치에 보안 링크가 있음을 나타내는 잠금 아이콘이 보입니다.

Messaging Server에 대한 SSL 구성은 686 페이지 "23.5 암호화 및 인증서 기반 인증 구성"을 참조하십시오.

24.7.2 Messaging Server 및 S/MIME 애플릿 간의 연결 보안

공개 키 인증서를 CRL에 대해 확인할 경우 S/MIME 애플릿이 Messaging Server와 직접 통신해야 합니다.

▼ SSL을 사용하여 통신 연결 보안을 유지하는 방법

1. **관리 작업을 수행하여 Messaging Server에서 SSL을 구성합니다. 686 페이지 "23.5 암호화 및 인증서 기반 인증 구성"을 참조하십시오.**

2. **smime.conf 파일에서 sslrootcacertsurl 매개 변수를 설정하여 루트 SSL CA 인증서를 찾기 위한 정보를 지정합니다. 이러한 CA 인증서는 Messaging Server와 S/MIME 애플릿 간에 SSL 연결을 설정할 때 Messaging Server의 SSL 인증서를 확인하는 데 사용됩니다.**

3. **smime.conf 파일에서 checkoverssl 매개 변수를 1로 설정합니다. 이 Messaging Server 옵션은 Messaging Server와 S/MIME 애플릿 간의 연결에 SSL이 사용되는지 여부를 지정합니다. checkoverssl을 1로 설정하면 Communications Express Mail 사용자가 Messenger Server의 URL을 지정하는 방법(HTTP 또는 HTTPS)에 상관없이 Messaging Server 및 S/MIME 애플릿 간의 연결이 SSL을 사용하여 보안됩니다.**

주 – Messaging Server와 Communications Express Mail 등의 클라이언트 응용 프로그램 간에 프록시 서버를 사용할 수 있습니다. 프록시 서버를 보안된 통신 연결과 함께 사용하거나 보안된 통신 연결 없이 사용하는 방법은 745 페이지 "24.9.4 프록시 서버 및 CRL 확인"을 참조하십시오.

24.8 클라이언트 시스템의 키 액세스 라이브러리

메일 사용자가 개인 공개 키 쌍과 인증서를 스마트 카드에 보관하는지 아니면 브라우저의 로컬 키 저장소에 보관하는지 여부에 상관없이 클라이언트 시스템에는 저장 방법을 지원하기 위한 키 액세스 라이브러리가 존재해야 합니다.

이러한 라이브러리는 스마트 카드 및 브라우저 공급업체가 제공합니다. 클라이언트 시스템에 올바른 라이브러리가 있는지 확인하고 `smime.conf` 파일에서 올바른 플랫폼 매개 변수에 라이브러리 이름을 지정해야 합니다. 선택할 수 있는 매개 변수는 다음과 같습니다.

- `platformwin`(PC에서 실행 중인 Microsoft Windows의 경우)

클라이언트 시스템에 설치된 라이브러리를 알고 있는 경우 이러한 라이브러리만 지정하거나 설치된 라이브러리가 확실하지 않을 경우 특정 플랫폼과 공급업체의 모든 라이브러리 이름을 지정할 수 있습니다. S/MIME 애플릿이 지정한 이름 중에서 필요한 라이브러리를 찾지 못할 경우 S/MIME 기능이 작동하지 않습니다.

하나 이상의 라이브러리 파일 이름을 지정하는 구문은 다음과 같습니다.

```
platform_parameter==vendor:library=library_name;...
```

여기서

*platorm_parameter*는 Communications Express Mail에 액세스하는 클라이언트 시스템의 플랫폼에 대한 매개 변수 이름입니다. 다음 이름 중 하나를 선택합니다(platformwin).
`platformwin`

*vendor*는 스마트 카드나 브라우저의 공급업체를 지정합니다. 다음 리터럴 중 하나를 선택합니다.

`cac`(ActivCard 또는 NetSign 스마트 카드의 경우)

`capi`(CAPI가 포함된 Internet Explorer의 경우)

`mozilla`(네트워크 보안 서비스가 포함된 Mozilla의 경우)

*library_name*은 라이브러리 파일 이름을 지정합니다. 공급업체 및 운영 체제에 대한 라이브러리 이름은 표 24–4를 참조하십시오.

표 24–4 클라이언트 시스템의 특수 라이브러리

스마트 카드 또는 브라우저 공급업체	운영 체제	라이브러리 파일 이름
	Windows	`acpkcs211.dll`
CAPI(Cryptographic Application Programming Interface)가 포함된 Internet Explorer	Windows	`capibridge.dll`
	Windows	`softokn3.dll`
	Windows	core32.dll

24.8.1 예

다음 예에서는 Windows 플랫폼에 대한 하나의 스마트 카드 라이브러리, 하나의 Internet Explorer 라이브러리 및 하나의 Mozilla 라이브러리를 지정합니다.

```
platformwin==CAC:library=acpkcs211.dll;CAPI:library=capibridge.dll;
MOZILLA:library=softokn3.dll;
```

24.9 개인 및 공개 키 확인

Communications Express Mail은 개인 키 또는 공개 키를 사용하기 전에 그림 24–2에 표시된 확인 테스트를 통과해야 합니다. 이 절의 나머지 부분에서는 공개 키 인증서를 CRL에 대해 확인하는 작업을 자세히 설명합니다.

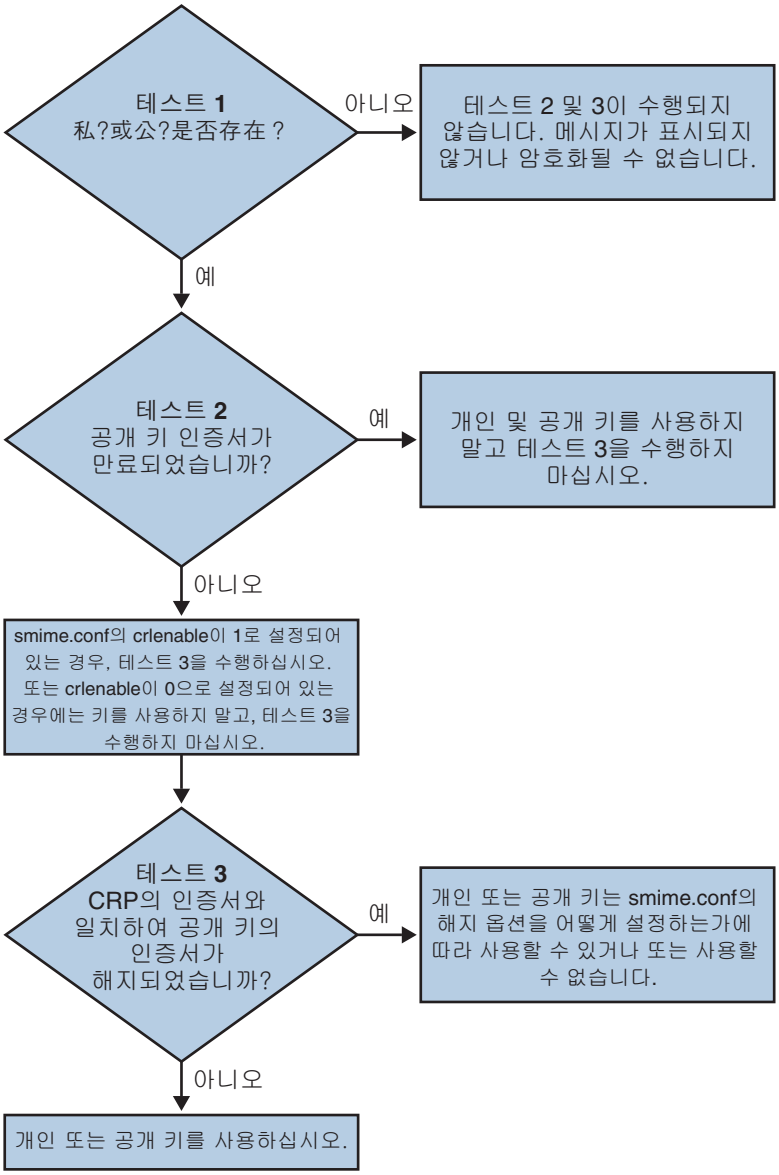


그림 24–2 개인 및 공개 키 확인

24.9.1 사용자의 개인 키 또는 공개 키 찾기

Communications Express Mail 사용자가 여러 개인 공개 키 쌍과 여러 전자 메일 주소(주, 대체 또는 별칭 주소)를 갖고 있는 경우 키가 이러한 주소와 연결될 수 있습니다. 이 경우

S/MIME 애플릿이 확인을 위해 모든 키를 찾는 것이 중요합니다. smime.conf 파일에서 usercertfilter 매개 변수를 사용하여 공개 키 인증서를 CRL에 대해 확인할 때 키 소유자의 메일 주소 목록을 만드는 필터를 정의합니다. 자세한 내용은 730 페이지 “24.5 smime.conf 파일의 매개 변수”의 usercerfilter를 참조하십시오.

24.9.2 CRL에 대해 인증서 확인 시기

인증서 해지 목록, 즉 CRL은 키 쌍과 인증서를 발급하는 CA가 유지 관리하는 해지된 인증서 목록입니다. CRL 확인이 사용 가능한 경우 시스템은 인증서 요청이 있을 때마다 CRL에서 해당 인증서가 해지되었는지 확인합니다.

smime.conf 파일에서 crlenable이 1로 설정되어 있는 경우에는 만료된 키가 발견될 때 CRL 테스트가 수행됩니다. 공개 키의 인증서를 CRL에 대해 확인합니다. 각 CA에 대해 CRL이 하나만 있을 수 있지만 동일한 CRL이 여러 위치에 있을 수 있습니다.

S/MIME 애플릿이 확인을 수행하라는 요청을 보내면 Messaging Server가 인증서를 CRL에 대해 확인합니다. 공개 키 인증서는 공개 키를 검증하기 위해 사용됩니다. 개인 키는 소유자만 사용할 수 있도록 비밀로 유지되기 때문에 개인 키를 CRL에 대해 직접 확인할 수 없습니다. 따라서 개인 키가 올바른지 확인하려면 키 쌍의 공개 키 인증서가 사용됩니다. 공개 키의 인증서가 CRL 테스트를 통과하면 연관된 개인 키도 테스트를 통과합니다.

소유자가 회사를 그만두거나 스마트 카드를 분실할 경우와 같은 다양한 이유 때문에 인증서가 해지될 수 있습니다.

다음과 같은 세 가지 경우에 인증서를 CRL에 대해 확인합니다.

- 보내는 메시지가 서명된 경우

  sendsigncert를 0으로 설정하지 않았거나 crlenable을 0으로 설정하지 않은 경우 S/MIME 애플릿은 항상 이 확인을 수행합니다.
- 서명된 받는 메시지를 읽을 경우

  readsigncert를 0으로 설정하지 않았거나 crlenable을 0으로 설정하지 않은 경우 S/MIME 애플릿은 항상 이 확인을 수행합니다.
- 보내는 메시지가 암호화된 경우

  sendencryptcert을 0으로 설정하지 않았거나 crlenable를 0으로 설정하지 않은 경우 S/MIME 애플릿은 항상 이 확인을 수행합니다.

24.9.3 CRL 액세스

인증서에는 Messaging Server가 CRL을 찾기 위해 사용하는 0개 이상의 URL(배포 지점이라고도 함)이 포함됩니다. 인증서에 CRL URL이 없을 경우 CRL에 대해 확인할 수 없으며 진짜 상태를 알지 못한 채 개인 또는 공개 키를 사용하여 메시지를 서명하거나 암호화하게 됩니다.

Messaging Server가 사용할 수 있는 모든 URL을 시도한 후 CRL을 찾거나 액세스하지 못할 경우 인증서 상태가 알 수 없는 것으로 간주됩니다. 상태를 알 수 없는 개인 키 또는 공개 키를 사용할지 여부는 `revocationunknown`의 설정에 따라 결정됩니다.

각 CA에 대해 하나의 CRL만 지원되지만 동일한 CRL의 여러 복사본이 사용자의 공개 키 인증서 간에 다른 URL로 표시되는 다른 위치에 존재할 수 있습니다. Messaging Server는 CRL에 액세스할 때까지 인증서의 모든 URL 위치를 시도합니다.

정기적으로 CA에서 최신 CRL을 원하는 위치에 다운로드하여 CRL의 여러 복사본에 최적으로 액세스할 수 있도록 관리할 수 있습니다. 인증서에 포함된 URL을 변경할 수 없지만 인증서의 URL을 CRL 정보가 포함된 새 URL로 매핑하여 새 CRL 위치를 사용하도록 Messaging Server를 리디렉션할 수 있습니다. 다음 구문을 사용하여 LDAP 디렉토리에서 하나 이상의 매핑 정의 목록을 만듭니다(표 24–3 참조).

`msgCRLMappingRecord=`*`url_in_certificate`*`==`*`new_url`*`[|`*`url_login_DN`*`|`*`url_login_password`*`]`

url_in_certificate는 CRL을 찾기 위한 이전 정보가 포함된 인증서의 URL입니다. new_url은 새 CRL 정보가 포함된 새 URL입니다. *url_login_DN* 및 *url_login_password*에 액세스할 수 있는 항목의 DN과 비밀번호입니다. 두 옵션은 모두 선택 사항이며 지정하는 경우 새 URL 액세스에 대해서만 사용됩니다.

DN과 비밀번호가 실패할 경우 LDAP 액세스가 거부되며 다른 자격 증명을 사용하여 다시 시도되지 않습니다. 이러한 로그인 자격 증명은 LDAP URL에 대해서만 유효합니다. `smmime.conf`에서 `crlurllogindn` 및 `crlurlloginpw`를 사용할 경우 매핑 레코드에 로그인 DN과 비밀번호를 지정할 필요가 없습니다. 728 페이지 “24.4.3 자격 증명을 사용하여 LDAP에서 공개 키, CA 인증서 및 CRL 액세스”를 참조하십시오.

한 계층의 매핑만 허용됩니다. 인증서의 다른 URL을 동일한 새 URL에 매핑할 수 있지만 인증서 URL을 여러 새 URL에 할당할 수는 없습니다. 예를 들어, 다음 매핑 목록은 유효하지 않습니다.

```
msgCRLMappingRecord=URL12==URL45
msgCRLMappingRecord=URL12==URL66
msgCRLMappingRecord=URL12==URL88
msgCRLMappingRecord=URL20==URL90
msgCRLMappingRecord=URL20==URL93
```

다음 예는 올바른 매핑 목록입니다.

```
msgCRLMappingRecord=URL12==URL45
msgCRLMappingRecord=URL14==URL66
msgCRLMappingRecord=URL88==URL66
msgCRLMappingRecord=URL201==URL90
msgCRLMappingRecord=URL202==URL93
```

LDAP 디렉토리에서 매핑 정의를 만든 후에 `smime.conf` 파일의 `crlmappingurl`을 사용하여 이러한 정의를 찾기 위한 디렉토리 정보를 지정합니다. 730 페이지 “24.5 smime.conf 파일의 매개 변수”를 참조하십시오.

24.9.4 프록시 서버 및 CRL 확인

시스템에서 클라이언트 응용 프로그램과 Messaging Server 간에 프록시 서버를 사용할 경우 CRL 확인을 수행하기 위해 S/MIME 애플릿을 올바르게 구성했더라도 CRL 확인이 차단될 수 있습니다. 이러한 문제가 발생하면 유효한 키 인증서가 해지 또는 알 수 없는 상태라는 것을 경고하는 오류 메시지가 Communications Express Mail 사용자에게 표시됩니다.

다음과 같은 상황에서 이 문제가 발생합니다.

- 다음 구성 값을 사용하여 CRL 확인을 요청한 경우
  - `smime.conf` 파일의 `crlenable` 매개 변수를 `1`로 설정
  - Messaging Server의 `local.webmail.cert.enable` 옵션을 `1`로 설정
- S/MIME 애플릿과 프록시 서버 간의 통신 연결이 SSL로 보안되지 않았지만 `smime.conf` 파일의 `checkoverssl` 매개 변수가 1로 설정되었기 때문에 `S/MIME` 애플릿이 보안 연결을 예상하는 경우

이 문제를 해결하려면 다음 작업을 수행합니다.

1. 클라이언트 시스템과 프록시 서버 간의 통신 연결을 SSL을 사용한 보안 연결로 설정하고 모든 구성 값을 그대로 둡니다. 또는,
2. 통신 연결을 보안되지 않은 상태로 두고 `checkoverssl`을 `0`으로 설정합니다.

자세한 내용은 738 페이지 “24.7 SSL을 사용하여 인터넷 연결 보안”을 참조하십시오.

24.9.5 오래된 CRL 사용

S/MIME 애플릿이 확인을 수행하라는 요청을 보내면 Messaging Server가 인증서를 CRL에 대해 확인합니다. Messaging Server는 인증서를 확인할 때마다 CRL을 메모리로 다운로드하는 대신 CRL 복사본을 디스크에 다운로드하고 해당 복사본을 인증서 확인에 사용합니다. 모든 CRL에는 지정한 날짜 이후에는 최신 버전의 CRL을 사용하도록 지정하는 next-update 필드가 있습니다. next-update 날짜는 CRL 사용의 만료 날짜나 시간 제한으로 간주할 수 있습니다. CRL은 next-update 날짜가 지나면 오래된 것으로 간주되며 Messaging Server는 다음 번에 인증서를 검사할 때 최신 버전의 CRL을 다운로드합니다.

S/MIME 애플릿이 인증서를 CRL에 대해 확인하도록 요청할 때마다 Messaging Server는 다음 작업을 수행합니다.

1. 현재 날짜를 CRL의 next-update 날짜와 비교합니다.
2. CRL이 오래된 경우 Messaging Server는 최신 버전의 CRL을 다운로드하여 디스크의 오래된 CRL을 대체한 다음에 확인 작업을 진행합니다. 그러나 최신 버전의 CRL을 찾을 수 없거나 다운로드할 수 없으면 `smime.conf` 파일의 `crlusepastnextupdate` 값에 따라 수행할 작업을 결정합니다.

3. crlusepastnextupdate가 0으로 설정된 경우 오래된 CRL이 사용되지 않으며 해당 인증서는 모호한 상태가 됩니다. S/MIME 애플릿은 smime.conf의 revocationunknown 값에 따라 다음과 같이 수행할 작업을 결정합니다.

   a. revocationunknown이 ok로 설정된 경우 인증서는 유효한 것으로 간주되며 개인 또는 공개 키를 메시지의 서명이나 암호화에 사용합니다.

   b. revocationunknown이 revoked로 설정된 경우 인증서는 유효하지 않은 것으로 간주되고 개인 또는 공개 키를 메시지 서명이나 암호화에 사용하지 않으며 키를 사용할 수 없다는 오류 메시지를 메일 사용자에게 표시됩니다.

   crlusepastnextupdate가 1로 설정된 경우 S/MIME 애플릿이 오래된 CRL을 계속 사용하므로 Communications Express Mail 내에서 처리 중단이 발생하지 않지만 이러한 상황을 알리기 위해 Messaging Server 로그 파일에 메시지가 기록됩니다.

인증서를 CRL에 대해 확인할 때 이러한 순서의 이벤트가 계속됩니다. Messaging Server가 최신 버전의 CRL을 제때에 smime.conf 파일의 설정에 따라 다운로드할 수 있으면 메일 처리가 중단 없이 계속됩니다. Messaging Server 로그를 정기적으로 확인하여 오래된 CRL이 사용 중임을 나타내는 메시지가 반복되는지 확인합니다. 최신 CRL을 다운로드할수 없는 경우 액세스할 수 없는 이유를 조사해야 합니다.

24.9.6 사용할 메시지 시간 지정

timestampdelta 매개 변수는 주로 다음 용도로 사용됩니다.

1. 메시지가 대상에 도착하는 데 오래 걸리는 상황을 처리하는 경우. 이 경우 메시지를 보낼 때 키가 유효했더라도 보낸 사람의 키가 잘못된 키로 간주되었을 수 있습니다.
2. 보낸 시간을 속일 수 있으므로 메시지의 보낸 시간에 대한 신뢰를 제한하려는 경우

모든 메시지는 다음 두 가지 시간과 관련되어 있습니다.

- 메시지를 보낸 시간(메시지 헤더 세부 정보의 날짜 행에 있음)
- 메시지가 대상에 도착한 시간(메시지 헤더 세부 정보의 마지막 받은 날짜 행에 있음)

주 – \메시지의 보낸 사람 필드 오른쪽에 있는 삼각형 아이콘을 누르면 메시지 헤더 세부 정보를 볼 수 있습니다.

메시지를 보낼 때 유효했던 인증서가 메시지가 대상에 도착할 때 해지되거나 만료될 수 있습 니다. 이 경우에는 인증서의 유효성을 검사할 때 보낸 시간과 받은 시간 중에서 어떤 시간 을 사용해야 하는지가 중요합니다. 보낸 시간을 사용하면 메시지를 보낼 때 인증서가 유효했는지 여부를 확인합니다. 그러나 항상 보낸 시간을 사용하면 메시지가 대상에 도착하는 데 오래 걸릴 수 있다는 사실을 고려하지 못합니다.

smime.conf 파일의 timestampdelta 매개 변수를 사용하여 CRL 확인에 사용할 시간에 영향을 줄 수 있습니다. 이 매개 변수는 초를 나타내는 양의 정수로 설정합니다. 받은 시간에서 timestampdelta 값을 뺀 시간이 보낸 시간보다 앞설 경우 보낸 시간이

사용됩니다. 그렇지 않을 경우에는 받은 시간이 사용됩니다. timestampdelta의 값이 작을수록 받은 시간이 더 자주 사용됩니다. timestampdelta를 설정하지 않으면 항상 받은 시간이 사용됩니다. 표 24–3의 timestampdelta를 참조하십시오.

24.9.7 CRL 액세스 문제

네트워크 또는 서버 문제와 같은 다양한 이유 때문에 Messaging Server가 인증서를 CRL에 대해 확인하려고 할 때 CRL을 사용하지 못할 수 있습니다. 이 경우에 Messaging Server가 계속 CRL에 액세스하려고 시도하느라 시간을 낭비하게 하는 대신 smime.conf 파일의 crlaccessfail 매개 변수를 사용하여 CRL 액세스를 시도하는 빈도를 관리함으로써 Messaging Server가 다른 작업을 수행하게 할 수 있습니다.

crlaccessfail을 사용하여 다음을 정의합니다.

- 실패한 시도의 횟수(시도가 실패할 때마다 Messaging Server 로그에 오류 메시지가 기록됨)
- 실패한 시도 횟수를 세는 기간
- 새 CRL 액세스 주기를 시도하기 전에 기다리는 시간

매개 변수의 구문과 예는 표 24–3의 crlaccessfail을 참조하십시오.

24.9.8 인증서가 해지된 경우

공개 키의 인증서가 CRL의 항목과 일치하지 않으면 개인 또는 공개 키가 보내는 메시지의 서명이나 암호화에 사용됩니다. 인증서가 CRL의 항목과 일치하거나 인증서의 상태를 알 수 없으면 개인 또는 공개 키는 해지된 것으로 간주됩니다. 기본적으로 Communications Express Mail은 해지된 인증서가 포함된 키를 보내는 메시지의 서명이나 암호화에 사용하지 않습니다. 수신자가 메시지를 읽을 때 서명된 메시지의 개인 키가 해지된 경우 수신자에게 서명을 신뢰할 수 없다는 경고 메시지가 표시됩니다.

원할 경우 smime.conf 파일에서 다음 매개 변수를 사용하여 해지된 모든 인증서에 대한 여러 기본 정책을 변경할 수 있습니다.

- sendsigncertrevoked를 allow로 설정하면 공개 키 인증서가 해지되어 해지된 것으로 간주되는 개인 키를 사용하여 보내는 메시지에 서명합니다.
- sendencryptcertrevoked를 allow로 설정하면 해지된 인증서가 있는 공개 키를 사용하여 보내는 메시지를 암호화합니다.
- revocationunknown을 ok로 설정하면 상태를 알 수 없는 인증서를 유효한 것으로 처리하며 개인 또는 공개 키를 보내는 메시지의 서명이나 암호화에 사용합니다.

24.10 S/MIME 기능을 사용할 수 있는 권한 부여

Communications Express Mail을 통해 사용할 수 있는 여러 메일 서비스를 사용할 수 있는 권한을 LDAP 필터를 사용하여 부여하거나 거부할 수 있습니다. 필터는 `mailAllowedServiceAccess` 또는 `mailDomainAllowedServiceAccess` LDAP 속성을 사용하여 정의합니다. 일반적으로 필터는 다음 세 가지 방법 중 하나로 작동합니다.

- 필터를 사용하지 않을 경우 모든 서비스에 대한 권한이 모든 사용자에게 부여됨
- 지정된 서비스 이름에 대한 권한이 일련의 사용자에게 명시적으로 부여됨(서비스 이름 목록 앞에 더하기 기호(+)가 있음)
- 지정된 서비스 이름에 대한 권한이 일련의 사용자에 대해 명시적으로 거부됨(서비스 이름 목록 앞에 음수 기호(-)가 있음)

S/MIME의 필수 메일 서비스 이름은 `http`, `smime` 및 `smtp`입니다. Communications Express Mail 사용자 중에서 S/MIME 사용을 제한해야 하는 경우 적절한 LDAP 속성 구문과 서비스 이름을 사용하여 필터를 만듭니다. 이러한 속성은 LDAP 명령을 사용하여 만들거나 수정합니다.

24.10.1 S/MIME 권한 예

1. 다음 예는 한 명의 Communications Express Mail 사용자가 S/MIME 기능에 액세스할 수 없게 합니다.

```
mailAllowedServiceAccess: -smime:*$+imap,pop,http,smtp:*
```

또는

```
mailAllowedServiceAccess: +imap,pop,http,smtp:*
```

2. 다음 예는 도메인의 모든 Communications Express Mail 사용자가 S/MIME 기능에 액세스할 수 없게 합니다.

```
mailDomainAllowedServiceAccess: -smime:*$+imap:*$+pop:*$+smtp:*$+http:*
```

또는

```
mailDomainAllowedServiceAccess: +imap:*$+pop:*$+smtp:*$+http:*
```

자세한 내용은 704 페이지 “23.7.2 필터 구문”을 참조하십시오.

24.11 인증서 관리

다음 예에서는 대부분 ldapsearch 및 ldapmodify 명령을 사용하여 LDAP 디렉토리에서 사용자 키와 인증서를 검색합니다. 이러한 명령은 Directory Server와 함께 제공됩니다. 이러한 명령에 대한 자세한 내용은 *Sun ONE Directory Server Resource Kit Tools Reference* 릴리스를 참조하십시오.

24.11.1 LDAP 디렉토리의 CA 인증서

다음 예에서는 인증 기관의 인증서를 LDAP 디렉토리에 추가합니다. 이러한 인증서의 디렉토리 구조는 이미 존재합니다. 인증서와 인증서가 속하는 LDAP 항목은 add-root-CA-cert.ldif라는 .ldif 파일에 입력합니다. Base64 인코딩된 텍스트로 입력해야 하는 인증서 정보를 제외하고 이 파일의 모든 텍스트는 ASCII 텍스트로 입력합니다.

```
dn: cn=SMIME Admin,ou=people,o=demo.siroe.com,o=demo
objectClass: top
objectClass: person
objectClass: organizationalPerson
objectClass: inetOrgPerson
objectClass: certificationAuthority
cn: RootCACerts
sn: CA
authorityRevocationList: novalue
certificateRevocationList: novalue
cacertificate;binary:: MFU01JTUUEjAQBgNVBAsTCU1zZ1NlcnZlcjcMBoGA1UEAxMTydG
QGEwJVUzEOMAwGA1UEMFUJTUUxEjAQBgNVBAsTCU1zZ1NlcnZlcjEMBoGA1UEAxMTQ2VydG
aFw0wNjAxMwODAwMDBaM267hgbX9FExCzAJByrjgNVBAk9STklBMQwCgYDVQQVHR8EgaQwg
YTAlVMRMQYDVQQIEwpDQUxJRk9STklBMQwwCgYDVQQKEwww3ltgYz11lzAdBgNVBpYSE9Vc
5yZWQaddWlm899XBsYW5ldC5jb20wgZ8wDQYJoGBAK1mUTy8vvnOFg4mlHjkghytQUR1k8l
5mvWRf77ntm5mGXRD3XMU4OciUq6zUfIg3ngvxlLyERTIqjUS8HQU4R5pvj+rrVgsAGjggE
+FNAJmtOV2A3wMyghqkVPNDP3Aqq2fkcn4va3C5nRNAYxNNVE84JJ0H3jyPDXhMBlQU6vQn
weMBAAjggEXMIIBEzARBglghkgBhCAQEEBApqlSai4mfuvjh02SQkoPMNDAgTwMB8GA1UdI
QYMBaAEd38IK05AHreiU9OYc6vNMOwZMIGsBgNVHR8EgaQwgaEwb6BtoGuGaWxkYXA6Lyht
bmcucmVkLmlbGFuZXQuY29tL1VJD1DXJ0aWZpY2F0ZSBNYW5hZ2VyLE9VPVBlb3BsZSxPPW
aWxxYT9jZXJ0aZpY2jdu2medXRllkghytQURYFNrkuoCygKoYoaHR0cDovL3Bla2kghytQU
Zy5yZWQuaXBsYW5lC5jb20vcGVranLmNybDAeBgNVHREEFzAVgRNwb3J0aWEuc2hhb0BzdW
4uY29tMA0GCxLm78freCxS3Pp078jyTaDci1AudBL8+RrRUQvxsMJfZeFED+Uuf10Ilt6kw
Tc6W5UekbirfEZGAVQIzlt6DQJfgpifGLvtQ60Kw==
```

ldapmodify 명령을 사용하여 CA 인증서를 LDAP 디렉토리에 추가합니다.

```
# ldapmodify -a -h demo.siroe.com -D "cn=Directory Manager" -w mypasswd -v
 -f add-root-CA-cert.ldif
```

smime.conf의 trustedurl 매개 변수 값은 LDAP 디렉토리에서 CA 인증서의 위치를 지정합니다. 예 1의 경우 trustedurl은 다음으로 설정됩니다.

```
trustedurl==ldap://demo.siroe.com:389/cn=SMIME Admin, ou=people,
o=demo.siroe.com,o=demo?cacertificate;binary?sub?
(objectclass=certificationAuthority)
```

24.11.2 LDAP 디렉토리의 공개 키 및 인증서

이 예에서는 메일 사용자의 공개 키와 인증서를 LDAP 디렉토리에 추가합니다. 이 예에서는 LDAP 디렉토리에 메일 사용자가 이미 있다고 가정합니다. 키와 인증서 및 이들이 속한 LDAP 항목은 `add-public-cert.ldif`라는 `.ldif` 파일에 입력합니다. Base64 인코딩된 텍스트로 입력해야 하는 키와 인증서 정보를 제외하고 이 파일의 모든 텍스트는 ASCII 텍스트로 입력합니다.

```
dn: uid=JohnDoe,ou=People, o=demo.siroe.com,o=demo
changetype: modify
replace: usercertificate
usercertificate;binary:: MFU01JTUUxEjAQBgNVBAsT1zZ1NlcnZlcjMBoGA1UEAxMTydG
QGEwJVUzEAwGA1hMFU01JTUUxEjAQBgNVBAsTCU1zZ1NlcnZlcjEcMBoGA1UEAxMTQ2VydG
aFw0wNjAxMTODAwaM267hgbX9FExCzAJBgwyrjgNVBAk9STklBMQwwCgYDVQQVHR8EgaQwg
AlVzMRMwEQYDVQQIDQUxJRk9STklBMQwwCgYDVQQKEwww3ltgoOYz11lzAdBgNVBpYSE9Vc
5yZWaddiiWlm899XBsYW5ldb20wgZ8wDQYJoGBAK1mUTy8vvO2nOFg4mlHjkghytQUR1k8l
5mvgcWL77ntm5mGXRD3XMU4OcizUfIg3ngvxlLKLyERTIqjUS8HQU4R5pvj+rrVgsAGjggE
+FG9NAqtOV2A3wMyghqkVPNDP3Aqq2BYfkcn4va3RNAYxNNVE84JJ0H3jyPDXhMBlQU6vQn
1NAgMBGjggEXMIIBEzARBglghkgBhvhCAQEEBApqlSai4mfuvjh02SQMNDAgTwMB8GA1UdI
QYMBaEd38IK05AHreiU9OYc6v+ENMOwZMIGsBgNVHR8EgaQwgaEwb6BuGaWxkYXA6Lyht74
tpbmcmVkLmlwbGFuZXQuY29tL1VJRD1DZXJ0aWZpY2F0ZSBNYW5hZ2V9VPVBlb3BsZSxPPW
1haWxT9jZXJ0aWZpY2jdu2medXRllHjkghytQURYFNrkuoCygKoYoaHDovL3Bla2kghytQU
luZy5WQuaXBsYW5ldC5jb20vcGVraW5nLmNybDAeBgNVHREEFzAVgRNw0aWEuc2hhb0BzdW
4uY29A0GCxLm78UfreCxS3Pp078jyTaDv2ci1AudBL8+RrRUQvxsMJfZD+Uuf10Ilt6kwhm
Tc6W5UekbirfEZGAVQIzlt6DQJfgpifGLvtQ60Kw==
```

`ldapmodify` 명령을 사용하여 공개 키와 인증서를 LDAP 디렉토리에 추가합니다.

```
# ldapmodify -a -h demo.siroe.com -D "cn=Directory Manager" -w mypasswd -v
 -f add-public-cert.ldif
```

`smime.conf`의 `certurl` 매개 변수 값은 LDAP 디렉토리에서 공개 키와 해당 인증서의 위치를 지정합니다. 예 2의 경우 `certurl`은 다음으로 설정됩니다.

```
certurl==ldap://demo.siroe.com:389/ou=people, o=demo.siroe.com,
o=demo?userCertificate;binary?sub?
```

24.11.3 키와 인증서가 LDAP 디렉토리에 있는지 확인

다음 예에서는 LDAP 디렉토리에서 CA 인증서와 공개 키 및 해당 인증서를 검색하는 것을 보여 줍니다.

24.11.3.1 하나의 CA 인증서 검색

다음 예에서 -b 옵션에 정의된 기본 DN cn=SMIME admin, ou=people,o=demo.siroe.com,o=demo objectclass=*는 LDAP 디렉토리에 있는 하나의 CA 인증서를 설명합니다. 디렉토리에서 찾는 경우 ldapsearch는 인증서에 대한 정보를 ca-cert.lidf 파일에 반환합니다.

```
# ldapsearch -L -h demo.siroe.com -D "cn=Directory Manager" -w mypasswd -b
"cn=SMIME admin, ou=people,o=demo.siroe.com,o=demo" "objectclass=*"
> ca-cert.ldif
```

아래 예에서는 ca-cert.ldif 파일의 검색 결과를 보여 줍니다. 파일 내용의 형식은 ldapsearch의 -L 옵션을 사용한 결과입니다.

```
# more ca-cert.ldif
dn: cn=SMIME admin,ou=people,o=demo.siroe.com,o=demo
objectClass: top
objectClass: person
objectClass: organizationalPerson
objectClass: inetOrgPerson
objectClass: certificationAuthority
cn: RootCACerts
cn: SMIME admin
sn: CA
authorityRevocationList: novalue
certificateRevocationList: novalue
cacertificate;binary:: MFU01JTUUxEjAQBgNVBAsTCU1zZNlcnZlcjcMBoGA1UEAxMTydG
QGEwJVEOMAwGA1UEChMFU0UUxEjAQBgNVBAsTCU1zZ1NlcnZlcjEcMBoGA1UEAxMTQ2VydG
aFw0jAxMTIwODAwMDBaM267X9FExCzAJBgwyrjgNVBAk9STklBMQwwCgYDVQQVHR8EgaQwg
YlVzMRMwEQYDVQQIEwpDQUx9STklBMQwwCgYDVQQKEwww3ltgoOYz11lzAdBgNVBpYSE9Vc
5yQuaddiiWlm899XBsYW5ljb20wgZ8wDQYJoGBAK1mUTy8vvO2nOFg4mlHjkghytQUR1k8l
5mcWRfL77ntm5mGXRD3XMciUq6zUfIg3ngvxlLKLyERTIqjUS8HQU4R5pvj+rrVgsAGjggE
+FNAJmqtOV2A3wMyghqkDP3Aqq2BYfkcn4va3C5nRNAYxNNVE84JJ0H3jyPDXhMBlQU6vQn
1NABAAGjggEXMIIBEzglghkgBhvhCAQEEBApqlSai4mfuvjh02SQkoPMNDAgTwMB8GA1UdI
QYMAFEd38IK05AHreOYc6v+ENMOwZMIGsBgNVHR8EgaQwgaEwb6BtoGuGaWxkYXA6Lyht74
tpbucmVkLmlwbGFuZY29tL1VJRD1DZXJ0aWZpY2F0ZSBNYW5hZ2VyLE9VPVBlb3BsZSxPPW
1haWYT9jZXJ0aWZpdu2medXRllHjkghytQURYFNrkuoCygKoYoaHR0cDovL3Bla2kghytQU
luZyZWQuaXBsYW5ldb20vcGVraW5nLmNybDAeBgNVHREEFzAVgRNwb3J0aWEuc2hhb0BzdW
4uYtMA0GCxLm78Ufre3Pp078jyTaDv2ci1AudBL8+RrRUQvxsMJfZeFED+Uuf10Ilt6kwhm
Tc6W5UekbirfEZGAVQIzlt6DQJfgpifGLvtQ60Kw==
```

여러 공개 키 검색

다음 예에서 -b 옵션에 정의된 기본 DN o=demo.siroe.com,o=demo objectclass=*는 LDAP 디렉토리의 기본 DN과 그 아래에서 발견된 모든 공개 키와 인증서를 usergroup.ldif 파일로 반환하도록 합니다.

```
# ldapsearch -L -h demo.siroe.com -D "cn=Directory Manager" -w mypasswd
-b "o=demo.siroe.com,o=demo" "objectclass=*" > usergroup.ldif
```

하나의 공개 키 검색

다음 예에서 -b 옵션에 정의된 기본 DN uid=JohnDoe, ou=people,o=demo.siroe.com,o=demo objectclass=*는 LDAP 디렉토리에 있는 하나의 공개 키와 해당 인증서를 설명합니다.

```
# ldapsearch -L -h demo.siroe.com -D "cn=Directory Manager" -w mypasswd -b
"uid=JohnDoe, ou=people,o=demo.siroe.com,o=demo" "objectclass=*" > public-key.ldif
```

아래 예는 public-key.ldif 파일의 검색 결과를 보여 줍니다. 파일 내용의 형식은 ldapsearch의 -L 옵션을 사용한 결과입니다.

```
# more public-key.ldif
dn: uid=sdemo1, ou=people, o=demo.siroe.com, o=demo
objectClass: top
objectClass: person
objectClass: organizationalPerson
objectClass: siroe-am-managed-person
objectClass: inetOrgPerson
objectClass: inetUser
objectClass: ipUser
objectClass: userPresenceProfile
objectClass: inetMailUser
objectClass: inetLocalMailRecipient
objectClass: icsCalendarUser
objectClass: sunUCPreferences
mail: JohnDoe@demo.siroe.com
mailHost: demo.siroe.com
.
.
uid: JohnDoe
.
.
mailUserStatus: active
inetUserStatus: active
.
.
usercertificate;binary:: MFU01JTUUxEjAQBgNBAsTCU1zZ1NlcnZjcMBoGA1UEAxMTydG
QGEwJEOwGA1UEChMFU01JTUUxEjAQBgNVBAsTCU1zZ1NlcnZlcjEcMBoGA1UEAxMTQ2VydG
aFw0MTIwODAwMDBaM267hgbX9FExCzAJBgwyrjgNVBAk9STklBMQwwCgYDVQQVHR8EgaQwg
YTAlVEQYDVQQIEwpDQUxJRk9STklBMQwwCgYDVQQKEwww3ltgoOYz11lzAdBgNVBpYSE9Vc
5yZWQdWlm899XBsYW5ldC5jb20wgZ8wDQYJoGBAK1mUTy8vvO2nOFg4mlHjkghytQUR1k8l
5mvgc7ntm5mGXRD3XMU4OciUq6zUfIg3ngvxlLKLyERTIqjUS8HQU4R5pvj+rrVgsAGjggE
+FG9NmV2A3wMyghqkVPNDP3Aqq2BYfkcn4va3C5nRNAYxNNVE84JJ0H3jyPDXhMBlQU6vQn
1NAgMAgEXMIIBEzARBglghkgBhvhCAQEEBApqlSai4mfuvjh02SQkoPMNDAgTwMB8GA1UdI
```

```
QYMBaEdK05AHreiU9OYc6v+ENMOwZMIGsBgNVHR8EgaQwgaEwb6BtoGuGaWxkYXA6Lyht74
tpbucmVkwbGFuZXQuY29tL1VJRD1DZXJ0aWZpY2F0ZSBNYW5hZ2VyLE9VPVBlb3BsZSxPPW
1haxYT9jZaWZpY2jdu2medXRllHjkghytQURYFNrkuoCygKoYoaHR0cDovL3Bla2kghytQU
luZyZWQuaYW5ldC5jb20vcGVraW5nLmNybDAeBgNVHREEFzAVgRNwb3J0aWEuc2hhb0BzdW
4u9tMA0GC78UfreCxS3Pp078jyTaDv2ci1AudBL8+RrRUQvxsMJfZeFED+Uuf10Ilt6kwhm
Tc6W5UekbirfEZGAVQIzlt6DQJfgpifGLvtQ60Kw==
.
.
```

24.11.4 네트워크 보안 서비스 인증서

네트워크 보안 서비스(NSS)에 사용되는 여러 인증서는 LDAP 데이터베이스가 아닌 자체 데이터베이스에 저장됩니다. 이러한 인증서와 관련 CRL을 데이터베이스에 저장하기 위해 Messaging Server에서는 `certutil` 및 `crlutil`의 두 가지 유틸리티를 제공합니다. 이러한 유틸리티를 사용하여 데이터베이스를 검색할 수도 있습니다.

`certutil`에 대한 자세한 내용은 **Sun Java System Directory Server 관리 설명서**(http://docs.sun.com/doc/817-2012)를 참조하십시오. 또한 `crlutil`과 함께 제공되는 도움말 텍스트를 참조하십시오(유틸리티를 인수 없이 실행하면 유틸리티의 온라인 도움말을 볼 수 있음).

24.12 Communications Express S/MIME 최종 사용자 정보

이 절에서는 최종 사용자를 위한 정보를 제공합니다. 이 절은 다음 내용으로 구성되어 있습니다.

24.12.1 처음으로 로그인

메일 사용자가 Communications Express Mail에 처음으로 로그인하면 S/MIME 애플릿 과 관련된 특수한 프롬프트가 표시됩니다.

24.12.1.1 Windows용 프롬프트

Windows 98, 2000 또는 XP에서 Communications Express Mail에 처음 로그인하면 다음 프롬프트가 표시됩니다.

1. JRE(Java 2 Runtime Environment)가 컴퓨터(클라이언트 시스템)에 설치되지 않은 경우 다음과 비슷한 프롬프트가 표시됩니다.

```
Do you want to install and run “Java Plug-in 1.4.2_03 signed on 11/20/03 and
distributed by Sun Microsystems, Inc.”?Publisher authenticity verified by:
VeriSign Class 3 Code Signing 2001 CA
```

Yes를 누르고 다음 프롬프트에 따라 JRE를 설치합니다.

주 – 영어 언어 지원을 원하지만 라틴어가 아닌 문자(예: 중국어)를 포함하는 S/MIME 메시지도 읽고 싶은 경우에는 컴퓨터의 `/lib` 디렉토리에 `charsets.jar` 파일이 있어야 합니다.

`charsets.jar` 파일이 `/lib` 디렉토리에 설치되도록 하려면 사용자 정의 설치를 사용하여 영어 버전의 JRE를 설치합니다. 설치 중에 "추가 언어 지원" 옵션을 선택합니다.

자세한 내용은 721 페이지 “24.3.6 여러 언어 지원”을 참조하십시오.

마지막 설치 프롬프트에서 "마침"을 누릅니다. 컴퓨터를 다시 시작하고 Communications Express Mail에 다시 로그인합니다.

2. 다음과 같은 프롬프트가 표시됩니다.

   ```
   Do you want to trust the signed applet distributed by “Sun Microsystems,
   Inc.”?Publisher authenticity verified by: Thawte Consulting cc
   ```

   다음 응답 중 하나를 누릅니다.

   - Communications Express Mail 세션에 S/MIME 애플릿을 허용하려면 예를 누릅니다. 로그인할 때마다 프롬프트가 표시됩니다.
     - S/MIME 애플릿을 거부하려면 아니오를 누릅니다. S/MIME 기능을 사용할 수 없습니다.
     - 현재 및 이후의 모든 Communications Express Mail 세션에 S/MIME 애플릿을 허용하려면 항상을 누릅니다. 프롬프트가 다시 표시되지 않습니다.
3. 다음과 같은 프롬프트가 표시됩니다.

   ```
   Do you want to trust the signed applet distributed by “sun microsystems,
   inc.”?Publisher authenticity verified by: VeriSign, Inc.
   ```

   다음 응답 중 하나를 누릅니다.

   - Communications Express Mail 세션에 S/MIME 애플릿을 허용하려면 예를 누릅니다. 로그인할 때마다 프롬프트가 표시됩니다.
   - S/MIME 애플릿을 거부하려면 아니오를 누릅니다. S/MIME 기능을 사용할 수 없습니다.
   - 현재 및 이후의 모든 Communications Express Mail 세션에 S/MIME 애플릿을 허용하려면 항상을 누릅니다. 프롬프트가 다시 표시되지 않습니다.

24.12.2 서명 및 암호화 설정

모든 사용자의 보내는 메시지를 다음 중 어떤 방법으로 처리할지 제어하기 위해 설정할 수 있는 초기 서명 및 암호화 설정이 있습니다.

- 자동으로 서명
- 자동으로 암호화
- 자동으로 서명 및 암호화

또한 초기 설정은 Communications Express Mail 창의 맨 아래와 옵션 - 설정 창에 있는 서명 및 암호화 확인란을 선택한(기능 설정) 또는 선택하지 않은(기능 해제) 상태로 표시할지 여부를 제어합니다. `smime.conf` 파일에서 `alwaysencrypt` 및 `alwayssign` 매개 변수를 사용하여 이러한 초기 설정을 지정합니다.

메일 사용자에게 메일의 초기 설정을 변경할 수 있다고 알려주십시오. Communications Express Mail에 로그인한 후 사용자는 하나의 메시지에 대한 설정을 임시로 대체하거나 모든 메시지에 대한 설정을 지속적으로 대체할 수 있습니다.

표 24–5에는 확인란 사용에 대해 요약되어 있습니다.

표 24–5 Communications Express Mail의 서명 및 암호화 확인란

확인란 텍스트	위치	Communications Express Mail 사용자가 수행하는 작업
메시지에 서명	메시지 작성, 전달 또는 회신에 사용되는 Communications Express Mail 창의 맨 아래에 있습니다.	■ 현재 메시지에 서명하려면 선택합니다. ■ 현재 메시지에 서명하지 않으려면 선택 취소합니다.
메시지 암호화	메시지 작성, 전달 또는 회신에 사용되는 Communications Express Mail 창의 맨 아래에 있습니다.	■ 현재 메시지를 암호화하려면 선택합니다. ■ 현재 메시지를 암호화하지 않으려면 선택 취소합니다.
보내는 모든 메시지에 서명	Communications Express Mail 옵션 설정 창에서 보안 메시징 옵션 아래에 있습니다.	■ 모든 메시지에 자동으로 서명하려면 선택합니다. ■ 모든 메시지에 자동으로 서명하지 않으려면 선택 취소합니다. 주: "메시지에 서명" 확인란을 사용하여 "보내는 모든 메시지에 서명"설정을 메시지별로 대체할 수 있습니다.

표 24–5 Communications Express Mail의 서명 및 암호화 확인란 (계속)

확인란 텍스트	위치	Communications Express Mail 사용자가 수행하는 작업
보내는 모든 메시지 암호화	Communications Express Mail 옵션 설정 창에서 보안 메시징 옵션 아래에 있습니다.	■ 모든 메시지를 자동으로 암호화하려면 선택합니다. ■ 모든 메시지를 자동으로 암호화하지 않으려면 선택 취소합니다. 주: "메시지 암호화" 확인란을 사용하여 "보내는 모든 메시지 암호화" 설정을 메시지별로 대체할 수 있습니다.

24.12.3 Java 콘솔 활성화

Communications Express Mail 사용자가 서명된 메시지와 암호화된 메시지를 처리할 때 S/MIME 애플릿에서 다양한 작업 메시지를 Java 콘솔에 기록할 수 있습니다. Java 콘솔 메시지는 메일 사용자가 보고한 문제를 해결하는 데 도움이 될 수 있습니다. 그러나 사용자의 LDAP 항목의 `inetMailUser` 객체 클래스에 `nswmExtendedUserPrefs` 속성을 추가하여 사용자에 대해 Java 콘솔을 활성화한 경우에만 작업 메시지가 생성됩니다. 예를 들면 다음과 같습니다.

```
nswmExtendedUserPrefs: meSMIMEDebug=on
```

이렇게 하면 Communications Express Mail의 성능이 크게 저하될 수 있으므로 모든 메일 사용자에 대해 Java 콘솔을 항상 활성화하지는 마십시오.

◆ ◆ ◆ 25장

25 로깅 관리

이 장에서는 Messaging Server MTA, 메시지 저장소 및 서비스의 로깅 기능에 대해 개괄적으로 설명합니다. 또한 이 장에서는 이러한 로깅 기능을 관리하는 방법에 대해 설명합니다.

이 장은 다음 내용으로 구성되어 있습니다.

25.1 로깅 개요

로깅은 시스템이 시스템 서비스에 대한 시간이 기록되고 레이블이 지정된 정보를 제공하는 수단입니다. 로깅에서는 시스템의 현재 스냅샷뿐 아니라 기록 보기도 제공합니다.

Messaging Server 로그 파일을 이해하고 사용하면 다음을 수행할 수 있습니다.

- 메시지 크기, 메시지 전달 속도, MTA를 통해 전달되는 메시지 수 등의 메시지 통계 수집
- 추세 파악
- 용량 계획 상호 연계
- 문제 해결

예를 들어, 사용자 수가 증가하여 더 많은 디스크 저장소를 사이트에 추가해야 할 경우 Messaging Server 로그 파일을 사용하여 시스템 수요의 증가 비율을 확인하고 필요한 새 디스크 저장소의 양을 계획할 수 있습니다.

또한 Messaging Server 로그를 사용하여 하루 동안의 메시징 패턴을 파악할 수 있습니다. 매일 최고 부하가 발생하는 시점을 파악하면 용량 계획을 수행하는 데 도움이 됩니다.

또한 로깅은 사용자 문제를 해결하는 데 도움이 됩니다. 예를 들어, 사용자가 예상한 메일 메시지를 받지 못할 경우 Messaging Server 로깅 기능을 사용하여 사용자의 메일 메시지를 추적할 수 있습니다. 이렇게 함으로써 메시지가 자동으로 필터링되어 SPAM 폴더로 보내졌기 때문에 도착하지 않았음을 확인할 수도 있습니다.

25.1.1 로깅 데이터의 유형

일반적으로 로깅은 두 가지 유형의 정보를 제공합니다.

- 작업 데이터
- 오류 조건(이벤트 로깅이라고도 함)

대부분의 경우 Messaging Server 로깅에서는 작업 데이터를 제공합니다. 이 작업 데이터에는 메시지가 시스템에 들어온 날짜와 시간, 메시지의 보낸 사람 및 받는 사람, 메시지가 디스크에 기록된 시간, 이후에 메시지가 디스크에서 제거되고 사용자의 메일함에 삽입된 시간 등의 정보가 포함됩니다.

또한 Messaging Server 로깅에서는 이벤트 로깅 데이터도 제공합니다. 이벤트 로깅 데이터를 얻으려면 다른 로그 파일에서 여러 항목을 모아야 합니다. 그런 다음 메시지 아이디와 같은 고유한 상수를 사용하여 시스템을 통해 지점 간에 전달된 메시지의 주기를 검색하고 연관시킬 수 있습니다.

25.1.2 Messaging Server 로그 파일의 유형

Messaging Server 로깅은 세 가지 유형의 로그 파일로 구성됩니다.

1. **MTA 로그.** 앞에서 설명한 Message Transfer Agent에 대한 작업 데이터를 제공합니다.
2. **오류 로그.** 이 오류 로그는 MTA 디버그 로그이며 MTA 하위 구성 요소 로그(즉, 작업 제어기, 디스패처 등)입니다.
3. **메시지 저장소 및 서비스 로그.**http 서버, mshttpd, imap, pop 서비스 및 Admin 서비스의 메시지를 제공합니다. 이 로그의 형식은 처음 두 로그 유형과 다릅니다.

다음 표에는 다양한 유형의 로그 파일이 나열되어 있습니다. 기본적으로 로그 파일은 *msg-svr-base*/data/log 디렉토리에 있습니다. 각 로그 파일의 유형을 개별적으로 사용자 정의하고 볼 수 있습니다.

표 25–1 Messaging Server 로그 파일

로그 파일 유형	로그 파일 설명	기본 이름
MTA	날짜 및 시간 정보, 대기열에 포함 및 대기열에서 제외 정보 등을 비롯하여 MTA를 통과하는 메시지 트래픽에 대한 정보를 보여 줍니다.	mail.log, mail.log_current, mail.log_yesterday

표 25–1 Messaging Server 로그 파일 (계속)

로그 파일 유형	로그 파일 설명	기본 이름
연결	전자 메일을 보내기 위해 이 시스템에 연결하는 원격 시스템(MTA)을 포함합니다.	connection.log
카운터	채널별로 송수신된 메시지에 관한 메시지 추세를 포함합니다.	counters
작업 제어기	마스터, 작업 제어기, 보낸 사람 및 대기열에서 제외 채널 프로그램에 대한 데이터를 포함합니다.	job_controller.log
디스패처	디스패처에 관한 오류를 포함합니다. 디스패처 디버깅을 설정하면 정보가 증가합니다.	dispatcher.log
채널	채널에 관한 오류를 기록합니다. master_debug 및 slave_debug 키워드는 채널 디버깅을 설정하여 채널 로그 파일의 자세한 표시 수준을 늘립니다. 정보 수준과 유형은 option.dat에서 여러 *_DEBUG MTA 옵션을 사용하여 제어합니다.	*channelname*_master.log*(예: tcp_local_master.log*) *channelname*_slave.log*(예: tcp_local_slave.log*)
IMAP	이 서버의 IMAP4 활동에 관련된 기록 이벤트가 포함됩니다.	imap, imap.*sequenceNum.timeStamp*
POP	이 서버의 POP3 활동에 관련된 기록 이벤트가 포함됩니다.	pop, pop.*sequenceNum.timeStamp*
HTTP	이 서버의 HTTP 활동에 관련된 기록 이벤트가 포함됩니다.	http, http.*sequenceNum.timeStamp*
기본값	명령줄 유틸리티 및 기타 프로세스 등과 같은 이 서버의 다른 활동에 관련된 기록 이벤트가 포함됩니다.	default, default.*sequenceNum.timeStamp*
msgtrace	메시지 저장소에 대한 추적 정보를 포함합니다. 파일이 매우 빠른 속도로 아주 커질 수 있습니다. 적절히 모니터하십시오.	msgtrace
watcher	프로세스 실패와 응답하지 않는 서비스(표 4–4 참조)를 모니터하고 특정 실패를 나타내는 오류 메시지를 기록합니다.	watcher

여기서

sequenceNum - 로그 파일 디렉토리에 있는 다른 로그 파일과 비교하여 이 로그 파일의 생성 순서를 지정하는 정수를 지정합니다. 일련 번호가 높은 로그 파일은 이 번호가 낮은 로그 파일보다 최신 파일입니다. 일련 번호는 롤오버되지 않습니다. 즉 서버 설치부터 시작하여 서버 수명 동안 계속 증가합니다.

timeStamp - 파일 생성 날짜 및 시간을 지정하는 큰 정수를 지정합니다. 이 값은 1970년 1월 1일 자정부터 시작하여 계산한 초의 수인 표준 UNIX 시간으로 표시됩니다.

예를 들어 imap.63.915107696라는 이름의 로그 파일은 1998년 12월 31일 12:34:56 PM에 생성되었으며 IMAP 로그 파일의 디렉토리에 만들어진 63번째 로그 파일입니다.

타임스탬프를 사용하여 개방형 일련 번호 지정의 조합을 통해, 분석을 위한 파일의 회전, 만료 및 선택에 더 큰 유연성을 가질 수 있습니다. 자세한 내용은 790 페이지 "25.4.3 서비스 로깅 옵션 정의 및 설정"을 참조하십시오.

25.1.3 여러 로그 파일에서 메시지 추적

시스템에서 메시지가 전달되는 방법과 여러 로그 파일에 정보가 기록되는 시점은 아래에 설명되어 있습니다. 이러한 설명을 통해 Message Server의 로그 파일을 사용하여 문제를 해결하는 방법을 이해할 수 있습니다. 그림 8–2를 참조하십시오.

1. 원격 호스트가 메시징 호스트의 TCP 소켓에 연결하여 SMTP 서비스를 요청합니다.
2. MTA 디스패처가 요청에 응답하고 메시징 호스트의 SMTP 서비스에 연결을 넘겨줍니다.

   MTA는 모듈식으로 설계되므로 작업 제어기 및 SMTP 서비스 디스패처를 포함하는 일련의 프로세스로 구성됩니다. 디스패처는 받는 TCP 연결을 가져와 SMTP 서비스로 보냅니다. SMTP 서비스는 메시지를 디스크의 채널 영역에 기록합니다. SMTP 서비스는 메시지의 봉투 매개 변수(예: 보낸 사람 및 받는 사람)를 인식합니다. 시스템의 구성 항목에서 시스템이 속한 대상 채널을 알려줍니다.
3. 디스패처가 스레드를 포크했고 특정 IP 주소에서 받는 연결에 대해 스레드를 사용할 수 있게 만들었다고 dispatcher.log 파일에 기록합니다.
4. SMTP 서버가 원격 호스트에서 SMTP 서버에 연결하여 메시지를 보낼 때 발생한 일에 대한 기록을 tcp_smtp_server.log 파일에 기록합니다. 이 로그 파일은 디스패처가 호스트 IP의 SMTP 서버로 보낼 때 작성됩니다.
5. SMTP 서버가 tcp_intranet과 같은 채널 프로그램의 디스크에 있는 대기열 영역에 메시지를 기록하고 작업 제어기에 알립니다.
6. 작업 제어기가 채널 프로그램에 연결합니다.
7. 채널 프로그램이 메시지를 전달합니다.

   채널마다 고유한 로그 파일이 있습니다. 그러나 이러한 로그에서는 일반적으로 채널의 시작 및 중지를 표시합니다. 추가 정보를 얻으려면 채널에 대한 디버그 수준을 활성화해야 합니다. 그러나 이로 인해서 시스템이 느려지고 진행 중인 문제가 더 모호해질 수 있으므로 실제 문제가 발생했을 경우에만 디버그 수준을 활성화해야 합니다.

주 – 효율성을 위해 채널이 기존 프로세스에 대해 이미 실행 중인 경우 시스템은 새 메시지가 들어와도 새 채널 프로세스를 만들지 않습니다. 현재 실행 중인 프로세스에서 새 메시지를 가져갑니다.

---

8. 메시지가 다른 호스트, 다른 TCP 연결 등의 다음 홉으로 전달됩니다. 이 정보는 `connection.log` 파일에 기록됩니다.

   SMTP 서버가 디스크의 대기열 영역에 메시지를 기록하는 것과 동시에 메시지를 담당하는 채널이 `mail.log_current` 또는 `mail.log` 파일에 레코드를 기록합니다. 이 레코드는 메시지가 대기열에 포함된 날짜와 시간, 보낸 사람, 받는 사람 등의 정보를 보여 줍니다. 자세한 내용은 771 페이지 "25.3.4 MTA 메시지 로깅 예"를 참조하십시오. 메시지를 추적하는 데 가장 유용한 파일은 `mail.log_current` 파일입니다.

25.2 로깅 관리를 위한 도구

`configutil` 명령을 사용하여 Messaging Server 로그 파일 작성 및 관리를 위한 정책을 사용자 정의할 수 있습니다.

메시지 저장소에 대해 지정하는 설정은 기록되는 이벤트와 이벤트 수에 영향을 미칩니다. 로그 파일을 분석할 때 이러한 설정 및 기타 특성을 사용하여 기록되는 이벤트를 자세히 검색할 수 있습니다.

MTA는 구성 파일에서 정보를 지정하여 MTA 로깅을 구성하는 별개의 로깅 기능을 사용합니다.

Messaging Server에서 제공하지 않는 로그 분석 및 보고서 생성 기능을 사용하려면 다른 도구를 사용해야 합니다. 로그 파일은 텍스트 편집기나 표준 시스템 도구로 조작할 수 있습니다.

정규 표현식 구문 분석을 지원하는 스크립트 가능 텍스트 편집기를 사용하면 이 장에서 설명하는 모든 조건을 기준으로 로그 항목을 추출하고 검색할 수 있으며, 결과를 정렬하거나 합계나 기타 통계를 낼 수도 있습니다.

UNIX 환경에서는 UNIX `syslog` 파일을 조작하기 위해 개발된 기존 보고서 생성 도구를 수정하고 사용할 수도 있습니다. 공개 도메인 `syslog` 조작 도구를 사용하려면 다른 날짜/시간 형식과 Messaging Server 로그 항목에는 있지만 `syslog` 항목에는 없는 두 개의 추가 구성 요소(*facility* 및 *logLevel*)를 고려하여 수정해야 합니다.

25.3 MTA 메시지 및 연결 로그 관리

MTA는 각 메시지가 대기열에 포함되고 제외될 때마다 로깅할 수 있는 기능을 제공합니다. 또한 디스패처 오류 및 디버깅 출력도 제공합니다.

이 절은 다음과 같은 하위 절로 구성되어 있습니다.

- 771 페이지 “25.3.4 MTA 메시지 로깅 예”
- 784 페이지 “25.3.5 디스패처 디버깅 활성화”

로깅은 채널별로 제어하거나 모든 채널의 메시지 활동이 기록되도록 지정할 수 있습니다. 초기 구성에서는 모든 채널에 대해 로깅이 비활성화됩니다.

자세한 내용은 766 페이지 “25.3.2 MTA 로깅 활성화”를 참조하십시오.

로깅을 활성화하면 MTA는 메시지가 MTA 채널을 통과할 때마다 *msg-svr-base*/data/log/mail* 파일에 항목을 기록합니다. 이러한 로그 항목은 MTA를 통해(또는 특정 채널을 통해) 전달된 메시지의 수에 대한 통계를 얻으려는 경우에 유용합니다. 또한 이러한 로그 항목을 사용하여 메시지의 전송/전달 여부와 시점 등의 다른 문제를 조사 할 수 있습니다.

매일밤 자정 정도에 실행되는 메시지 반환 작업은 기존 mail.log_yesterday를 누적 로그 파일인 mail.log에 추가하고, 현재 mail.log_current 파일의 이름을 mail.log_yesterday로 바꾼 다음 새 mail.log_current 파일을 시작합니다. 또한 메시지 반환 작업은 모든 connection.log* 파일에 대해 비슷한 작업을 수행합니다.

MTA는 현재 파일을 유지하기 위해 자동 롤오버를 수행하지만 파일 백업, 파일 자르기, 파일 삭제 등의 작업에 대한 정책을 결정하기 위해 누적 mail.log 파일을 관리해야 합니다.

로그 파일 관리 방법에 대해 고려할 때는 MTA의 주기적 반환 작업이 사이트에서 제공하는 *msg-svr-base*/bin/daily_cleanup 프로시저(있을 경우)를 실행한다는 것에 주의합니다. 따라서 일부 사이트는 매주 한 번 기존 mail.log 파일의 이름을 바꾸는 등의 자체적인 정리 절차를 제공할 수 있습니다.

주 – 로깅을 활성화하면 mail.log가 지속적으로 증가하므로 확인하지 않는 경우 디스크 공간을 모두 차지해버릴 수 있습니다. 따라서 이 파일의 크기를 주기적으로 모니터하여 불필요한 내용은 삭제해야 합니다. 또한 필요에 따라 전체 파일을 삭제하여 다른 버전이 생성되도록 할 수도 있습니다.

25.3.1 MTA 로그 항목 형식 이해

MTA 로그 파일은 ASCII 텍스트로 기록됩니다. 기본적으로 각 로그 파일 항목에는 아래 예와 같이 8개나 9개의 필드가 포함됩니다.

```
16-Feb-2007 14:54:13.72 tcp_local ims-ms EE 1 adam@sesta.com
rfc822;marlowe@siroe.com marlowe@ims-ms-daemon
```

로그 항목에는 다음이 표시됩니다.

1. 항목이 작성된 날짜와 시간(예: 16-Feb-2007 14:54:13.72)
2. 소스 채널의 채널 이름(이 예의 경우 tcp_local)

3. 대상 채널의 채널 이름(이 예의 경우 ims-ms)(SMTP 채널의 경우 LOG_CONNECTION이 활성화되어 있으면, 플러스 기호(+)는 SMTP 서버에 대한 인바운드를 나타내고 마이너스 기호(-)는 SMTP 클라이언트를 통한 아웃바운드를 나타냅니다.)
4. 항목의 유형(이 예의 경우 EE). 항목은 단일 작업 코드로 구성되거나(표 25–2 참조) 하나의 작업 코드나 하나 이상의 수정자 코드로 구성됩니다(표 25–3 참조). 항목의 형식은 다음과 같습니다.

   <*action_code*> <*zero or more optional modifiers*>

   예를 들어, 로깅 항목 코드 EEC는 전자 메일이 ESMTP(수정자 E) 및 SMTP 청크(수정자 C)를 사용하여 대기열에 포함(작업 코드 E)되었음을 의미합니다. 현재 사용된 작업과 수정자 코드에 대한 자세한 내용은 아래 표를 참조하십시오.
5. 메시지의 크기(이 예의 경우 1). 기본적으로 KB로 표현되지만 MTA 옵션 파일에 BLOCK_SIZE 키워드를 사용하여 이 기본값을 변경할 수 있습니다. 이 필드에서 파일 크기 대신 페이지 수를 기록하도록 SMS 채널을 구성할 수 있습니다. 938 페이지 “LOG_PAGE_COUNT”를 참조하십시오.
6. 봉투의 From: 주소(이 예의 경우 adam@sesta.com). 알림 메시지와 같이 봉투의 From: 주소가 비어 있는 메시지의 경우에는 이 필드가 비어 있습니다.
7. 봉투의 To: 주소의 원래 형식(이 예의 경우 marlowe@siroe.com).
8. 봉투의 To: 주소의 활성(현재) 형식(이 예의 경우 marlowe@ims-ms-daemon).
9. 전달 상태(SMTP 채널 전용)

다음 세 표에서는 로깅 항목 코드를 설명합니다.

표 25–2 로깅 항목 작업 코드

항목	설명
B	SMTP 서버로 보낸 잘못된 명령. 수신자 주소 필드에는 거부된 명령이 포함되고 진단 필드에는 SMTP 서버가 제공한 응답이 포함됩니다. MTA 채널 옵션 MAX_B_ENTRIES는 지정된 세션에 기록되는 잘못된 명령의 수를 제어합 니다. 기본값은 10입니다.
D	대기열에서 제외 성공
E	대기열에 포함
J	대기열에 포함 시도 거부(슬레이브 채널 프로그램에 의한 거부)
K	수신자 메시지 거부됨. 보낸 사람이 NOTIFY=NEVER DSN 플래그 설정을 요청하거나 메시지가 시간 초과하거나 메시지를 수동으로 반환하는 경우(예: imsimta qm “delete” 명령은 각 수신자에 대해 항상 “K” 레코드를 생성하고 qm “return” 명령은 “R” 레코드 대신 “K” 레코드 생성). 보낸 사람의 요청에 따라 보낸 사람에게 알림을 보내지 않았음을 나타냅니다. rejection/time-out과 동일한 종류이지만 실패한 메시지에 대한 새 알림 메시지(원래의 보낸 사람에게 보냄)이 생성되는 “R” 레코드와 비교될 수 있습니다.
Q	대기열에서 제외 일시적으로 실패

표 25–2 로깅 항목 작업 코드 *(계속)*

항목	설명
R	대기열에서 제외 시도에서 수신자 주소 거부(마스터 채널 프로그램에 의한 거부) 또는 실패/바운스 메시지의 생성
V	트랜잭션이 비정상적으로 중단될 때마다 나타나는 경고 메시지. 대기열에 포함된 수신자 주소별로 하나의 "V" 레코드가 있습니다.
W	메시지가 아직 전달되지 않았지만 아직 대기열에서 시도 중에 있음을 원래 전송자에게 알려주기 위해 전송되는 경고 메시지
Z	일부 수신자는 성공했지만 이 수신자는 일시적으로 성공하지 못했습니다. 모든 수신자의 원본 메시지 파일이 대기열에서 제외되었으며 대신 이 수신자와 다른 성공하지 못한 수신자를 위한 새 메시지 파일이 곧 대기열에 포함됩니다.

다음 표에서는 로깅 항목 수정자 코드를 설명합니다.

표 25–3 로깅 항목 수정자 코드

항목	설명
A	SASL 인증이 사용됩니다.
C	청크가 사용되었습니다. 청크가 작동하려면 ESMTP를 사용해야 하므로 일반적으로 `EEC` 또는 `DEC`와 같은 필드 값이 표시됩니다.
E	EHLO 명령이 실행/허용되었으며 이로 인해 ESMTP가 사용되었습니다.
L	LMTP가 사용되었습니다.
S	TLS/SSL이 사용되었습니다. S 트랜잭션 로그 항목은 이제 채널과 연관된 다양한 전송된 메시지 개수를 증가시킵니다.

`LOG_CONNECTION`을 활성화한 경우(**Sun Java System Messaging Server 6.3 Administration Reference**의 “Option File Format and Available Options” 참조) 추가 작업 코드 세트가 사용됩니다. 아래 내용을 참조하십시오.

표 25–4 SMTP 채널의 `LOG_CONNECTION` 작업 코드 + 또는 - 항목

항목	설명
C	연결 끊김뒤이어 진단 필드가 표시됩니다. connection.log_current(하나의 로그 파일이 사용되는 경우에는 mail.log_current)에 작성됩니다. 연결이 끊긴 이유를 기록하는 데 사용됩니다. 특히, 연결이 끊긴 이유가 일부 세션이 연결 끊기 제한에 도달했기 때문인 경우 이 사실이 진단 필드에 표시됩니다.
O	연결 열림

표 25–4 SMTP 채널의 LOG_CONNECTION 작업 코드 + 또는 - 항목 *(계속)*

항목	설명
U	SMTP 인증 성공 및 실패를 기록합니다. 형식은 다른 O 항목 및 C 항목의 경우와 같습니다. 특히, 응용 프로그램 필드와 전송 정보 필드가 동일한 순서로 표시됩니다. 사용자 이름이 알려져 있으면 사용자 이름 필드에 기록됩니다. LOG_CONNECTION MTA 옵션의 Bit 7(값 128)이 이를 제어합니다.
X	연결 거부됨
Y	연결이 설정되기 전에 연결 시도가 실패했음
I	ETRN 명령이 수신됨

MTA 옵션 파일에서 LOG_CONNECTION, LOG_FILENAME, LOG_MESSAGE_ID , LOG_NOTARY , LOG_PROCESS 및 LOG_USERNAME을 모두 활성화하면 형식은 아래 예와 같이 됩니다. (인쇄상의 이유로 샘플 로그 항목에서는 행이 바뀌어졌지만 실제 로그 항목은 한 행에 표시됩니다.)

```
16-Feb-2007 15:04:01.14 2bbe.5.3 tcp_local ims-ms
EE 1 service@siroe.com rfc822;adam@sesta.com
adam@ims-ms-daemon 20 /opt/SUNWmsgsr/data/queue/ims-ms/000/ZZf0r2i0HIaY1.01
<0JDJ00803FAON200@mailstore.siroe.com> mailsrv
siroe.com (siroe.com [192.160.253.66])
```

위에서 설명한 것 이외의 추가 필드는 다음과 같습니다.

1. 점(.) 문자와 카운트가 뒤에 붙은 프로세스 아이디(16진수로 표현됨). 다중 스레드 채널 항목인 경우(예: tcp_* 채널 항목) 프로세스 아이디와 카운트 사이에 스레드 아이디도 있습니다. 이 예에서 프로세스 아이디는 2bbe.5.3입니다.
2. 정수로 표현된 메시지의 NOTARY(전달 수신 요청) 플래그(이 예의 경우 20)
3. MTA 대기열 영역의 파일 이름(이 예의 경우 /opt/SUNWmsgsr/data/queue/ims-ms/000/ZZf0r2i0HIaY1.01).
4. 메시지 아이디(이 예의 경우 <0JDJ00803FAON200@mailstore.siroe.com>).
5. 실행 프로세스의 이름(이 예의 경우 mailsrv). UNIX에서 SMTP 서버 등의 디스패처 프로세스로 일반적으로 mailsrv SASL이 사용되지 않은 경우, 이 예의 경우 *service@siroe.com)입니다.
6. 연결 정보(이 예의 경우 siroe.com (siroe.com [192.160.253.66]). 연결 정보는 HELO/EHLO 행(받는 SMTP 메시지)에서 전송 시스템이 나타내는 이름 또는 대기열에 포함 채널의 공식 호스트 이름(다른 종류의 채널) 등의 전송 시스템이나 채널 이름으로 구성됩니다. TCP/IP 채널의 경우 전송 시스템의 실제 이름, 즉 DNS 역조회 및/또는 IP 주소에 의해 보고되는 심볼릭 이름은 ident* 채널 키워드에 의해 제어되어 괄호 안에 표시될 수 있습니다. 344 페이지 “12.4.3.4 IDENT 조회”를 참조하십시오. 이 샘플에서는 이러한 키워드 중 하나를 사용한 것으로 가정합니다. 이 경우 DNS와 IP 주소에서 발견된 이름을 모두 표시하는 기본 identnone 키워드를 사용합니다.

25.3.2 MTA 로깅 활성화

단지 몇 개의 특정 MTA 채널에 대한 통계만 수집하려면 해당 MTA 채널에 대해서만 로깅 채널 키워드를 활성화합니다. 모든 MTA 채널에 대한 로깅을 활성화하는 사이트가 많이 있습니다. 특히 문제를 추적하는 경우 문제 진단의 첫 번째 단계로 예상하거나 의도한 채널로 메시지가 전달되는지 알아내고 모든 채널에 대해 로깅을 활성화하면 이러한 문제의 진단에 도움이 됩니다.

▼ 특정 채널에서 MTA 로깅을 활성화하는 방법

1. **`imta.cnf` 파일을 편집합니다.**
   이 파일은 `/opt/SUNWmsgsr/config` 디렉토리에 있습니다.

2. **특정 채널에 대한 로깅을 활성화하려면 `logging` 키워드를 채널 정의에 추가합니다. 예를 들면 다음과 같습니다.**

   channel-name keyword1 keyword2 **`logging`**

   또한 로그 파일의 디렉토리 경로, 로그 수준 등의 여러 구성 매개 변수도 설정할 수도 있습니다. 786 페이지 "25.4 메시지 저장소, Admin 및 Default 서비스 로그 관리"를 참조하십시오.

▼ 모든 채널에서 MTA 로깅을 활성화하는 방법

1. **`imta.cnf` 파일을 편집합니다.**
   이 파일은 `/opt/SUNWmsgsr/config` 디렉토리에 있습니다.

2. **로깅 키워드를 `defaults` 채널 구성 파일에 추가합니다(292 페이지 "12.1 채널 기본값 구성" 참조). 예를 들면 다음과 같습니다.**

```
defaults logging notices 1 2 4 7 copywarnpost copysendpost postheadonly
noswitchchannel immnonurgent maxjobs 7 defaulthost siroe.com siroe.com

!
! delivery channel to local /var/mail store
l subdirs 20 viaaliasrequired maxjobs 7 pool LOCAL_POOL
mailhost.siroe.com
```

25.3.3 추가 MTA 로깅 옵션 지정

로깅이 활성화되면 기본 정보가 항상 제공될 뿐 아니라 MTA 옵션 파일에 다양한 `LOG_*` MTA 옵션을 구성하여 선택적 정보 필드가 추가로 포함되도록 할 수 있습니다. IMTA 조정 파일(*msg-svr-base*`/config/imta_tailor`)에서 `IMTA_OPTION_FILE` 옵션을 사용하여 지정한 파일에서 MTA 옵션 파일을 지정합니다. 기본적으로 이 파일은 *msg-svr-base*`/config/option.dat`입니다.

MTA Option 파일에 대한 자세한 내용은 **Sun Java System Messaging Server 6.3 Administration Reference**의 “Option File”을 참조하십시오.

이 절은 다음과 같은 하위 절로 구성되어 있습니다.

▼ MTA 로그를 syslog에 보내는 방법

1 MTA 옵션 파일을 편집합니다.

2 LOG_MESSAGES_SYSLOG 옵션을 1로 설정합니다.

값이 0이면 syslog 알림 생성이 비활성화됩니다. 값이 0이 아니면 syslog 알림 생성이 활성화되며, 절대값이 syslog 우선 순위 및 기능 마스크를 제어합니다. 양수 값은 syslog 알림과 일반 mail.log* 항목을 의미합니다. 음수 값(권장되지 않음)은 syslog 알림만 의미하며 일반 mail.log* 항목을 비활성화합니다. 값 0이 기본값이며 syslog(이벤트 로그) 로깅이 수행되지 않음을 나타냅니다.

▼ 로그 항목의 형식을 제어하는 방법

1 MTA option.dat 파일을 편집합니다.

2 LOG_FORMAT 옵션을 설정합니다.

- 1은 (기본값) 표준 형식입니다.
- 2는 null이 아닌 형식을 요청합니다. 빈 주소 필드가 "<>" 문자열로 변환됩니다”
- 3은 계산된 형식을 요청합니다. 모든 가변 길이 필드 앞에 N이 붙습니다. 여기서 N은 필드의 문자 수입니다.
- 4는 로그 항목을 XML 호환 형식으로 기록합니다. 항목 로그 항목은 여러 속성을 포함하고 하위 요소를 갖지 않은 단일의 XML 요소로 표시됩니다. 현재는 en(대기열에 포함된/대기열에서 제거된 항목), co(연결 항목) 및 he(헤더 항목)의 세 요소가 정의되어 있습니다.

  대기열에 포함/대기열에서 제거(en) 요소는 다음과 같은 속성을 가질 수 있습니다.

```
ts - time stamp (always present)
no - node name (present if LOG_NODE=1)
pi - process id (present if LOG_PROCESS=1)
sc - source channel (always present)
dc - destination channel (always present)
ac - action (always present)
sz - size (always present)
so - source address (always present)
od - original destination address (always present)
de - destination address (always present)
rf - recipient flags (present if LOG_NOTARY=1)
fi - filename (present if LOG_FILENAME=1)
ei - envelope id (present if LOG_ENVELOPE_ID=1)
mi - message id (present if LOG_MESSAGE_ID=1)
us - username (present if LOG_USERNAME=1)
ss - source system (present if bit 0 of LOG_CONNECTION
     is set and source system information is available)
se - sensitivity (present if LOG_SENSITIVITY=1)
pr - priority (present if LOG_PRIORITY=1)
in - intermediate address (present if LOG_INTERMEDIATE=1)
ia - initial address (present if bit 0 of LOG_INTERMEDIATE
     is set and intermediate address information is available)
fl - filter (present if LOG_FILTER=1 and filter information
     is available)
re - reason (present if LOG_REASON=1 and reason string is set)
di - diagnostic (present if diagnostic info available)
tr - transport information (present if bit 5 of LOG_CONNECTION
     is set and transport information is available)
ap - application information (present if bit 6 of LOG_CONNECTION
     is set and application information is available)
qt - the amount of time a message has spent in the queue (LOG_QUEUE_TIME=1)
```

다음은 샘플 en 항목입니다.

```
<en ts="2004-12-08T00:40:26.70" pi="0d3730.10.43" sc="tcp_local"
dc="l" ac="E" sz="12" so="info-E8944AE8D033CB92C2241E@whittlesong.com"
od="rfc822;ned+2Bcharsets@mauve.sun.com"
de="ned+charsets@mauve.sun.com" rf="22"
fi="/path/ZZ01LI4XPX0DTM00IKA8.00" ei="01LI4XPQR2EU00IKA8@mauve.sun.com"
mi="<11a3b401c4dd01$7c1c1ee0$1906fad0@elara>" us=""
ss="elara.whittlesong.com ([208.250.6.25])"
in="ned+charsets@mauve.sun.com" ia="ietf-charsets@innosoft.com"
fl="spamfilter1:rvLiXh158xWdQKa9iJ0d7Q==, addheader, keep"/>
```

이 항목은 편의상 줄 바꿈되어 있지만 실제 로그 파일 항목은 항상 한 줄로 표시됩니다.

연결(co) 항목은 다음과 같은 속성을 가질 수 있습니다.

```
ts - time stamp (always present, also used in en entries)
no - node name (present if LOG_NODE=1, also used in en entries)
pi - process id (present if LOG_PROCESS=1, also used in en entries)
sc - source channel (always present, also used in en entries)
dr - direction (always present)
ac - action (always present, also used in en entries)
tr - transport information (always present, also used in en entries)
ap - application information (always present, also used in en entries)
mi - message id (present only if message id info available,
     also used in en entries)
us - username (present only if username information available, also
     used in en entries)
di - diagnostic (present only if diagnostic information available,
     also used in en entries)
ct - the amount of time a message has spent in the queue (LOG_QUEUE_TIME=1,
also used in en entries)
```

다음은 샘플 co 항목입니다.

```
<co ts="2004-12-08T00:38:28.41" pi="1074b3.61.281" sc="tcp_local" dr="+"
ac="O" tr="TCP|209.55.107.55|25|209.55.107.104|33469" ap="SMTP"/>
```

헤더(he) 항목은 다음과 같은 속성을 가집니다.

```
ts - time stamp (always present, also used in en entries)
no - node name (present if LOG_NODE=1, also used in en entries)
pi - process id (present if LOG_PROCESS=1, also used in en entries)
va - header line value (always present)
```

다음은 샘플 he 항목입니다.

```
<he ts="2004-12-08T00:38:31.41" pi="1074b3.61.281" va="Subject: foo"/>
```

▼ 로그 메시지 항목을 연관시키는 방법

1 MTA 옵션 파일을 편집합니다.

2 LOG_MESSAGE_ID 옵션을 1로 설정합니다.
값 0이 기본값이며 메시지 아이디가 mail.log 파일에 저장되지 않음을 나타냅니다.

▼ 메시지가 대기열에 보관되어 있었던 시간을 기록하는 방법

1 **MTA 옵션 파일을 편집합니다.**

2 **LOG_QUEUE_TIME 옵션을 1로 설정합니다.**
이 옵션은 메시지가 대기열에 보관되어 있었던 시간을 기록합니다. 대기열 시간은 정수 값(초)으로 기록되며,응용 프로그램 정보 문자열 바로 뒤에 비 XML 형식 로그로 표시됩니다. 이 값에 대한 XML 형식 로그의 속성 이름은 qt입니다.

▼ 메시지 전달 재시도 횟수를 식별하는 방법

1 **MTA 옵션 파일을 편집합니다.**

2 **LOG_FILENAME 옵션을 1로 설정합니다.**
이 옵션을 사용하면 특정 메시지 파일의 전달이 재시도된 횟수를 곧바로 쉽게 알 수 있습니다. 또한 이 옵션을 사용하면 MTA가 여러 수신자를 대상으로 하는 메시지를 디스크 상에서 개별 메시지 파일 복사본으로 분할하거나 분할하지 않은 경우를 식별할 수 있습니다.

▼ TCP/IP 연결을 기록하는 방법

1 **MTA 옵션 파일을 편집합니다.**

2 **LOG_CONNECTION 옵션을 설정합니다.**
이 옵션을 사용하면 MTA는 메시지 트래픽 뿐만 아니라 TCP/IP 연결을 기록합니다. 연결 로그 항목은 기본적으로 mail.log* 파일에 기록됩니다. 또는 연결 로그 항목을 connection.log* 파일에 기록할 수도 있습니다. 자세한 내용은 SEPARATE_CONNECTION_LOG 옵션을 참조하십시오.

▼ 항목을 connection.log 파일에 기록하는 방법

1 **MTA 옵션 파일을 편집합니다.**

2 **SEPARATE_CONNECTION_LOG 옵션을 1로 설정합니다.**
연결 로그 항목을 connection.log 파일에 대신 기록하도록 지정하려면 이 옵션을 사용합니다. 기본값 0을 사용하면 연결 로깅이 MTA 로그 파일에 저장됩니다.

▼ 프로세스 아이디로 로그 메시지를 연관시키는 방법

1 **MTA 옵션 파일을 편집합니다.**

2 **LOG_PROCESS 옵션을 설정합니다.**
이 옵션을 LOG_CONNECTION과 함께 사용하면 연결 항목과 메시지 항목을 프로세스 아이디로 연관시킬 수 있습니다.

▼ 메시지를 대기열에 포함시키는 프로세스에 연관된 사용자 아이디를 mail.log 파일에 저장하는 방법

1 **MTA 옵션 파일을 편집합니다.**

2 **LOG_USERNAME 옵션을 설정합니다.**
이 옵션은 메시지를 대기열에 포함시키는 프로세스에 연관된 아이디를 mail.log 파일에 저장할지 여부를 제어합니다. SASL(SMTP AUTH)이 사용된 SMTP 제출의 경우 아이디 필드는 인증된 아이디(앞에 별표 문자가 접두사로 붙음)가 됩니다.

25.3.4 MTA 메시지 로깅 예

MTA 메시지 파일에 기록되는 정확한 필드 형식 및 필드 목록은 설정하는 로깅 옵션에 따라 다릅니다. 이 절에서는 일반적인 로그 항목의 몇 가지 예를 보여 줍니다. 이 절은 다음과 같은 하위 절로 구성되어 있습니다.

추가적인 옵션 필드에 대한 설명은 766 페이지 “25.3.3 추가 MTA 로깅 옵션 지정”을 참조하십시오.

주 – 인쇄상의 이유로 로그 파일 항목이 여러 행으로 표시되어 있지만 실제 로그 파일 항목은 한 항목당 한 행으로 표시됩니다.

로그 파일을 검토하는 경우, 일반적인 시스템에서는 많은 메시지가 한 번에 처리된다는 점에 유의하십시오. 일반적으로 특정 메시지에 관련된 항목은 같은 시간에 처리되고 있는 다른 메시지와 관련된 항목 간에 섞여 있습니다. 기본 로깅 정보는 MTA를 통해 이동하는 전체 메시지를 이해하는 데 적합합니다.

같은 메시지에 관련된 특정 항목을 같은 수신자에게 연관시키려면 LOG_MESSAGE_ID를 활성화합니다. 특정 메시지를 MTA 대기열 영역의 특정 파일과 연관시키거나, 아직 성공적으로 대기열에서 제외되지 않은 메시지의 배달 시도가 몇 번 있었는지 알아내려면 LOG_FILENAME을 활성화합니다. SMTP 메시지의 경우(TCP/IP 채널을 통해 처리됨) 원격 시스템의 TCP 연결을 전송된 메시지와 연관시키려면 LOG_PROCESS와 LOG_CONNECTION의 몇 가지 수준을 활성화합니다.

25.3.4.1 MTA 로깅 예: 사용자가 보내는 메시지 전송

아래 예는 로컬 사용자가 보내는 TCP/IP 채널(예: 인터넷)로 메시지를 전송하는 경우 볼 수 있는 로그 항목의 기본 예를 보여 줍니다. 이 예에서는 LOG_CONNECTION이 활성화되어 있습니다. (1)과 (2)로 표시된 행은 하나의 항목입니다. 실제 로그 파일에서는 한 행으로 표시됩니다. 마찬가지로 (3) - (7)로 표시된 행도 하나의 항목이며 실제로 한 행으로 표시됩니다.

예 25–1 MTA 로깅: 로컬 사용자가 보내는 메시지 전송

```
16-Feb-2007 15:41:32.36 tcp_intranet tcp_local    EE 1        (1)
adam@sesta.com rfc822;marlowe@siroe.com marlowe@siroe.com     (2)
siroe.com (siroe.com [192.160.253.66])

16-Feb-2007 15:41:34.73 tcp_local                 DE 1        (3)
adam@sesta.com rfc822;marlowe@siroe.com marlowe@siroe.com     (4)
thor.siroe.com dns;thor.siroe.com

(TCP|206.184.139.12|2788|192.160.253.66|25)                   (5)

(thor.siroe.com ESMTP Sendmail ready Thu 15 Feb 2007 21:37:29 -0700 [MST])     (6)

smtp;250 2.1.5 <marlowe@siroe.com>... Receipt ok              (7)
```

1. 이 행은 한(1) 블록 메시지의 tcp_intranet 채널부터 tcp_local 채널까지의 대기열에 포함 날짜 및 시간을 ESMTP(EE)로 표시합니다.
2. 이 부분은 로그 파일에서 (1)과 같은 행의 일부이며 여기서는 인쇄 편의상 별도의 행으로 표시했습니다. 봉투의 From: 주소(이 경우 adam@sesta.com) 및 봉투의 To: 주소의 원래 버전과 현재 버전(이 경우 marlowe@siroe.com)을 표시합니다.
3. 이 행은 한(1) 블록 메시지의 tcp_local 채널의 대기열에서 제외 날짜와 시간을 ESMTP(DE)로 표시합니다. 즉 tcp_local 채널에 의한 일부 원격 SMTP 서버로의 성공적 전송을 표시합니다.
4. 또한 봉투의 From: 주소, 원래 봉투의 To: 주소 및 봉투의 To: 주소의 현재 형식을 표시합니다.

예 25–1 MTA 로깅: 로컬 사용자가 보내는 메시지 전송 *(계속)*

5. 연결이 이루어진 실제 시스템의 이름이 DNS에서 `thor.siroe.com`이며, 로컬 전송 시스템의 IP 주소가 206.184.139.12이고 포트 2788에서 전송되고, 원격 대상 시스템의 IP 주소가 192.160.253.66이고 원격 대상 시스템의 연결 포트는 포트 25임을 보여 줍니다.
6. 원격 SMTP 서버의 SMTP 배너 행을 표시합니다.
7. 이 주소에 대해 반환된 SMTP 상태 코드를 표시합니다. 250은 기본 SMTP 성공 코드이며 이 원격 SMTP 서버는 확장된 SMTP 상태 코드와 추가 텍스트로 응답합니다.

25.3.4.2 MTA 로깅 예: 옵션 로깅 필드 포함

이 예는 예 25–3과 비슷한 로깅 항목을 보여 주지만, `LOG_FILENAME=1` 및 `LOG_MESSAGE_ID=1`을 설정하여 파일 이름(아래 1 및 3)과 메시지 아이디(아래 2 및 4)를 표시합니다.. 특히 메시지 아이디를 사용하여 항목과 메시지를 연관시킬 수 있습니다.

예 25–2 MTA 로깅 – 옵션 로깅 필드 포함

```
16-Feb-2007 15:41:32.36 tcp_intranet tcp_local    EE 1
adam@sesta.com rfc822;marlowe@siroe.com marlowe@siroe.com
/opt/SUNWmsgsr/data/queue/tcp_local/002/ZZf0r4i0Wdy51.01      (1)
<0JDJ00D02IBWDX00@sesta.com>                                  (2)
siroe.com (siroe.com [192.160.253.66])

16-Feb-2007 15:41:34.73 tcp_local                 DE 1
adam@sesta.com rfc822;marlowe@siroe.com marlowe@siroe.com
/opt/SUNWmsgsr/data/queue/tcp_local/002/ZZf0r4i0Wdy51.01      (3)
<0JDJ00D02IBWDX00@sesta.com>                                  (4)
thor.siroe.com dns;thor.siroe.com
(TCP|206.184.139.12|2788|192.160.253.66|25)
(thor.siroe.com ESMTP Sendmail ready at Thu, 15 Feb 2007 21:37:29 -0700 [MST])
smtp;250 2.1.5 <marlowe@siroe.com>... Recipient ok
```

25.3.4.3 MTA 로깅 예 – 목록으로 전송

이 예는 `LOG_FILENAME=1`, `LOG_MESSAGE_ID=1` 및 `LOG_CONNECTION=1`을 활성화하여 여러 수신자에게 전송하는 예를 보여 줍니다. 여기서 사용자 `adam@sesta.com`은 MTA 메일링 목록 `test-list@sesta.com`으로 전송하였고 이 메일링 목록은 `bob@sesta.com`, `carol@varrius.com` 및 `david@varrius.com`으로 확장됩니다. 원래 봉투의 `To:` 주소는 각 수신자에 대해 `test-list@sesta.com`이고, 현재 봉투의 `To:` 주소는 각각의 해당 주소입니다. 두 개의 별도 파일(`l` 채널에 대해 하나, `tcp_local` 채널에서 나가는 파일 하나)이 관련되어 있지만 메시지 아이디는 동일하게 유지됩니다.

예 25–3 MTA 로깅 – 목록으로 전송

```
20-Feb-2007 14:00:16.46 tcp_local    tcp_local    EE 1
adam@sesta.com rfc822;test-list@sesta.com carol@varrius.com
/opt/SUNWmsgsr/data/queue/tcp_local/004/ZZf0r2D0yuej4.01
<0JDQ00706R0FX100@sesta.com>
siroe.com (siroe.com [192.160.253.66])

20-Feb-2007 14:00:16.47 tcp_local    tcp_local    EE 1
adam@sesta.com rfc822;test-list@sesta.com david@varrius.com
/opt/SUNWmsgsr/data/queue/tcp_local/004/ZZf0r2D0yuej4.01
<0JDQ00706R0FX100@sesta.com>
siroe.com (siroe.com [192.160.253.66])

20-Feb-2007 14:00:16.48 tcp_local    ims-ms       EE 1
adam@sesta.com rfc822;test-list@sesta.com bob@ims-ms-daemon
/opt/SUNWmsgsr/data/queue/ims-ms/008/ZZf0r2D0yuej6.01
<0JDQ00706R0FX100@sesta.com>
siroe.com (siroe.com [192.160.253.66])

20-Feb-2007 14:00:16.68 ims-ms                    D 1
adam@sesta.com rfc822;test-list@sesta.com bob@ims-ms-daemon
/opt/SUNWmsgsr/data/queue/ims-ms/008/ZZf0r2D0yuej6.01
<0JDQ00706R0FX100@sesta.com>

20-Feb-2007 14:00:17.73 tcp_local                 DE 1
adam@sesta.com rfc822;test-list@sesta.com carol@varrius.com
/opt/SUNWmsgsr/data/queue/tcp_local/004/ZZf0r2D0yuej4.01
<0JDQ00706R0FX100@sesta.com>
gw.varrius.com dns;gw.varrius.com (TCP|206.184.139.12|2788|192.160.253.66|25)
(gw.varrius.com -- SMTP Sendmail)
smtp;250 2.1.5 <carol@varrius.com >... Recipient ok

20-Feb-2007 14:00:17.75 tcp_local                 DE 1
adam@sesta.com rfc822;test-list@sesta.com david@varrius.com
/opt/SUNWmsgsr/data/queue/tcp_local/004/ZZf0r2D0yuej4.01
<0JDQ00706R0FX100@sesta.com>
gw.varrius.com dns;gw.varrius.com (TCP|206.184.139.12|2788|192.160.253.66|25)
(gw.varrius.com -- SMTP Sendmail)
smtp;250 2.1.5 <david@varrius.com>... Recipient ok
```

25.3.4.4 MTA 로깅 – 존재하지 않는 도메인으로 전송

이 예는 존재하지 않는 도메인(여기에서는 very.bogus.comcom)으로의 전송 시도를 보여 줍니다. 즉, MTA의 다시 쓰기 규칙에 의해 존재하지 않는 것으로 알려지지 않으며 MTA가 보내는 TCP/IP 채널에 일치하는 도메인 이름으로 전송을 시도합니다. 이 예에서는 MTA 옵션을 LOG_FILENAME=1 및 LOG_MESSAGE_ID=1로 설정한 것으로 가정합니다.

TCP/IP 채널이 실행되어 DNS에서 도메인 이름을 검사할 때 DNS는 이름이 존재하지 않는다는 오류를 반환합니다. (5)에 있는 "rejection" 항목(R)과 (6)에 있는 유효한 도메인 이름이 아니라는 오류를 반환하는 DNS를 주의하십시오.

메시지가 제출된 뒤 주소가 거부되었기 때문에 MTA는 원래 전송자에게 바운스 메시지를 생성합니다. MTA는 새 거부 메시지를 원래의 전송자(1)에게 보내도록 대기열에 포함시키고 복사본은 포스트마스터에게 전송한 다음(4) 원래의 아웃바운드 메시지를 삭제합니다((5)에 있는 R 항목).

바운스 메시지 등의 알림 메시지에는 봉투의 From: 필드가 빈 공간으로 표시되어 있는 빈 봉투의 From: 주소((2) 및 (8))가 있습니다. MTA가 생성한 바운스 메시지의 초기 포함된 대기열은 새 알림 메시지의 아이디와 원래 메시지의 아이디(3)를 보여 줍니다. 이러한 정보가 항상 MTA에 사용 가능한 것은 아니지만 기록할 수 있는 경우에는 아웃바운드 실패 메시지에 해당하는 로그 항목을 결과 알림 메시지에 해당하는 로그 항목에 연관시킬 수 있도록 해줍니다. 이러한 알림 메시지는 프로세스 채널의 대기열에 포함되고 그런 다음 적절한 대상 채널(7)의 대기열에 포함됩니다.

예 25–4 MTA 로깅 – 존재하지 않는 도메인으로 전송

```
20-Feb-2007 14:17:07.77 tcp_intranet tcp_local    E 1
adam@sesta.com rfc822;user@very.bogus.com user@very.bogus.com
/opt/SUNWmsgsr/data/queue/tcp_local/008/ZZf0r2D0CVaL0.00
<0JDQ00903RS89T00@sesta.com>
siroe.com (siroe.com [192.160.253.66])

20-Feb-2007 14:17:08.24 tcp_local    process      E 1       (1)
rfc822;adam@sesta.com adam@sesta.com                        (2)
/opt/SUNWmsgsr/data/queue/process/ZZf0r2D0CVbR0.00
<0JDQ00904RSK9Z00@sesta.com>,<0JDQ00903RS89T00@sesta.com>   (3)
tcp-daemon.mailhost.sesta.com

20-Feb-2007 14:17:08.46 tcp_local    process      E 1       (4)
rfc822;postmaster@sesta.com postmaster@sesta.com
/opt/SUNWmsgsr/data/queue/process/ZZf0r2D0CVbR1.00
<0JDQ00906RSK9Z00@sesta.com>,<0JDQ00903RS89T00@sesta.com>
tcp-daemon.mailhost.sesta.com

20-Feb-2007 14:17:08.46 tcp_local                 R 1       (5)
adam@sesta.com rfc822;user@very.bogus.com user@very.bogus.com
/opt/SUNWmsgsr/data/queue/tcp_local/008/ZZf0r2D0CVaL0.00
<0JDQ00903RS89T00@sesta.com>
Illegal host/domain name found                              (6)
(TCP active open: Failed gethostbyname() on very.bogus.com, resolver errno = 1)

20-Feb-2007 14:17:09.21 process      ims-ms       E 3       (7)
rfc822;adam@sesta.com adam@ims-ms-daemon                    (8)
/opt/SUNWmsgsr/data/queue/ims-ms/018/ZZf0r2D0CVbS1.00
```

예 25–4 MTA 로깅 – 존재하지 않는 도메인으로 전송 *(계속)*

```
<0JDQ00904RSK9Z00@sesta.com>
process-daemon.mailhost.sesta.com

20-Feb-2007 14:17:09.72 process      ims-ms       E 3
rfc822;postmaster@sesta.com postmaster@ims-ms-daemon
/opt/SUNWmsgsr/data/queue/ims-ms/014/ZZf0r2D0CVbS2.00
<0JDQ00906RSK9Z00@sesta.com>
process-daemon.mailhost.sesta.com

20-Feb-2007 14:17:09.73 ims-ms                    D 3
rfc822;adam@sesta.com adam@ims-ms-daemon
/opt/SUNWmsgsr/data/queue/ims-ms/018/ZZf0r2D0CVbS1.00
<0JDQ00904RSK9Z00@sesta.com>

20-Feb-2007 14:17:09.84 ims-ms                    D 3
rfc822;postmaster@sesta.com postmaster@ims-ms-daemon
/opt/SUNWmsgsr/data/queue/ims-ms/014/ZZf0r2D0CVbS2.00
<0JDQ00906RSK9Z00@sesta.com>
```

25.3.4.5 MTA 로깅 예 – 존재하지 않는 원격 사용자에게 전송

이 예는 원격 시스템의 잘못된 주소로 전송을 시도하는 예입니다. 이 예에서는 MTA 옵션 설정이 LOG_FILENAME=1 및 LOG_MESSAGE_ID=1이고 채널 옵션 설정이 LOG_BANNER=1 및 LOG_TRANSPORTINFO=1인 것으로 가정합니다. (1)에 있는 거부 항목 (R)에 주의하십시오. 하지만 예 25–4의 거부 항목과는 대조적으로 여기에 있는 거부 항목은 원격 시스템으로 연결되었음을 보여주고 원격 SMTP 서버, (2) 및 (3)에 의해 발생한 SMTP 오류 코드를 보여줍니다. (2)에 있는 정보가 포함된 이유는 채널 옵션이 LOG_BANNER=1 및 LOG_TRANSPORTINFO=1로 설정되었기 때문입니다.

예 25–5 MTA 로깅 – 존재하지 않는 원격 사용자에게 전송

```
26-Feb-2007 13:56:35.16 tcp_intranet tcp_local    EE 1
adam@sesta.com rfc822;nonesuch@siroe.com nonesuch@siroe.com
/opt/SUNWmsgsr/data/queue/tcp_local/000/ZZf0s690a3mf2.01
<0JE100J08UU24H00@sesta.com>
siroe.com (siroe.com [192.160.253.66])

26-Feb-2007 13:56:35.19 tcp_local    process      E 1
rfc822;adam@sesta.com adam@sesta.com
/opt/SUNWmsgsr/data/queue/process/ZZf0s690a3ml2.00
<0JE100J09UUB4N00@sesta.com>,<0JE100J08UU24H00@sesta.com>
tcp-daemon.mailhost.sesta.com

26-Feb-2007 13:56:35.20 tcp_local    process      E 1
rfc822;postmaster@sesta.com postmaster@sesta.com
```

예 25–5 MTA 로깅 – 존재하지 않는 원격 사용자에게 전송 *(계속)*

```
/opt/SUNWmsgsr/data/queue/process/ZZf0s690a3ml3.00
<0JE100J0BUUB4N00@sesta.com>,<0JE100J08UU24H00@sesta.com>
tcp-daemon.mailhost.sesta.com

26-Feb-2007 13:56:35.20 tcp_local                RE 1          (1)
adam@sesta.com rfc822;nonesuch@siroe.com nonesuch@siroe.com
/opt/SUNWmsgsr/data/queue/tcp_local/000/ZZf0s690a3mf2.01
<0JE100J08UU24H00@sesta.com>
thor.siroe.com dns;thor.siroe.com
(TCP|206.184.139.12|2788|192.160.253.66|25)                  (2)
(thor.siroe.com -- Server ESMTP [Sun Java System Messaging
Server 6.2-8.01 [built Feb 16 2007]])
smtp;550 5.1.1 unknown or illegal alias: nonesuch@siroe.com  (3)

26-Feb-2007 13:56:35.62 process      ims-ms       E 4
rfc822;adam@sesta.com adam@ims-ms-daemon
/opt/SUNWmsgsr/data/queue/ims-ms/003/ZZf0s690a3mm5.00
<0JE100J09UUB4N00@sesta.com>
process-daemon.mailhost.sesta.com

26-Feb-2007 13:56:36.07 process      ims-ms       E 4
rfc822;postmaster@sesta.com postmaster@ims-ms-daemon
/opt/SUNWmsgsr/data/queue/ims-ms/016/ZZf0s690a3nm7.01
<0JE100J0BUUB4N00@sesta.com>
process-daemon.mailhost.sesta.com

26-Feb-2007 13:56:35.83 ims-ms                   D 4
rfc822;adam@sesta.com adam@ims-ms-daemon
/opt/SUNWmsgsr/data/queue/ims-ms/003/ZZf0s690a3mm5.00
<0JE100J09UUB4N00@sesta.com>

26-Feb-2007 13:56:36.08 ims-ms                   D 4
rfc822;postmaster@sesta.com postmaster@ims-ms-daemon
/opt/SUNWmsgsr/data/queue/ims-ms/016/ZZf0s690a3nm7.01
<0JE100J0BUUB4N00@sesta.com>
```

25.3.4.6 MTA 로깅 예 – 원격측의 메시지 제출 시도 거부

이 예는 MTA가 원격 메시지 제출 시도를 거부할 때 발생하는 로그 파일 항목을 보여 줍니다. 이 예에서는 선택적인 LOG_* 옵션을 활성화하지 않은 것으로 간주하므로 항목에 기본적인 필드가 기록됩니다. 특히 LOG_CONNECTION 옵션을 활성화하면 J 항목 등과 같은 추가 필드가 표시됩니다.이 경우에는 다음을 비롯한 ORIG_SEND_ACCESS 매핑을 사용하여 SMTP 릴레이 차단을 설정한(529 페이지 “18.7 SMTP 릴레이 차단 구성” 참조) MTA에 대한 예입니다.

```
ORIG_SEND_ACCESS

! ...numerous entries omitted...
!
   tcp_local|*|tcp_local|*   $NRelaying$ not$ permitted
```

여기서 alan@very.bogus.com은 내부 주소가 아닙니다. 따라서 원격 사용자 harold@varrius.com이 MTA 시스템을 통해 원격 사용자 alan@very.bogus.com에게 중계는 시도는 거부됩니다.

예 25–6 MTA 로깅 – 원격측의 메시지 제출 시도 거부

```
26-Feb-2007 14:10:06.89 tcp_local                JE 0  (1)
harold@varrius.com rfc822; alan@very.bogus.com                (2)
530 5.7.1 Relaying not allowed: alan@very.bogus.com           (3)
```

1. 이 로그는 MTA가 원격측의 메시지 제출 시도를 거부한 날짜와 시간을 보여 줍니다. 거부는 J 레코드에서 표시합니다. MTA 채널이 거부된 메시지를 전송하려고 시도하는 경우는 예 25–4 및 예 25–5에서 볼 수 있는 것처럼 R 레코드에서 표시합니다.

   주 – 로그에 기록된 마지막 J 레코드에는 지정된 세션에 대한 마지막 레코드라는 표시가 있습니다. 또한 현재 버전의 Messaging Server에서는 J 레코드 수에 제한이 없습니다.

   ---

2. 시도된 봉투의 From: 및 To: 주소가 표시됩니다. 이 경우 원래 봉투의 To: 정보를 사용할 수 없으므로 해당 필드가 비어 있습니다.
3. 해당 항목에는 MTA가 원격측(전송을 시도한 보낸 사람)에게 발행한 SMTP 오류 메시지가 포함됩니다.

25.3.4.7 MTA 로깅 예 – 복수 전달 시도

이 예는 첫 번째 시도에서 메시지를 배달을 할 수 없어서 MTA가 메시지 전송을 여러 번 시도한 경우의 로그 파일 항목입니다. 이 예에서는 옵션을 LOG_FILENAME=1 및 LOG_MESSAGE_ID=1로 설정한 것으로 가정합니다.

예 25–7 MTA 로깅 – 복수 전달 시도

```
26-Feb-2007 14:38:16.27 tcp_intranet tcp_local    EE 1          (1)
adam@sesta.com rfc822;user@some.org user@some.org
/opt/SUNWmsgsr/data/queue/tcp_local/001/ZZf0s690kN_y0.00
<0JE100L05WRJIC00@sesta.com>

26-Feb-2007 14:38:16.70 tcp_local                 Q 1                   (2)
adam@sesta.com rfc822;user@some.org user@some.org
/opt/SUNWmsgsr/data/queue/tcp_local/001/ZZf0s690kN_y0.00               (3)
```

예 25–7 MTA 로깅 – 복수 전달 시도 *(계속)*

```
<0JE100L05WRJIC00@sesta.com>
TCP active open: Failed connect() 192.1.1.1:25  Error: no route to host  (4)

...several hours worth of entries...

26-Feb-2007 16:58:11.20 tcp_local                Q 1                    (5)
adam@sesta.com rfc822;user@some.org user@some.org
/opt/SUNWmsgsr/data/queue/tcp_local/001/ZYf0s690kN_y0.01                (6)
<0JE100L05WRJIC00@sesta.com>
TCP active open: Failed connect() 192.1.1.1:25    Error: no route to host

...several hours worth of entries...

26-Feb-2007 19:15:12.11 tcp_local                Q 1
adam@sesta.com rfc822;user@some.org user@some.org
/opt/SUNWmsgsr/data/queue/tcp_local/001/ZXf0s690kN_y0.00                 (7)
<0JE100L05WRJIC00@sesta.com>
TCP active open: Failed connect() 192.1.1.1:25  Error: Connection refused (8)

...several hours worth of entries...

26-Feb-2007 22:41:12.63 tcp_local                DE 1                    (9)
adam@sesta.com rfc822;user@some.org user@some.org
/opt/SUNWmsgsr/data/queue/tcp_local/001/ZXf0s690kN_y0.00
<0JE100L05WRJIC00@sesta.com>
host.some.org dns;host.some.org (TCP|206.184.139.12|2788|192.1.1.1|25)
(All set, fire away)
smtp;250 2.1.5 <user@some.org >... Recipient ok
```

1. 메시지는 `tcp_internal` 채널로 보내집니다. 일반적으로 POP 또는 IMAP 클라이언트 또는 MTA를 SMTP 릴레이로 사용하여 조직 내의 다른 호스트가 보내는 것이며, MTA는 이것을 보내는 `tcp_local` 채널의 대기열에 포함시킵니다.
2. Q 항목에 표시되어 있는 것처럼 첫 번째 전달 시도는 실패했습니다.
3. 이는 `ZZ*` 파일 이름에서 볼 수 있는 첫 번째 전달 시도입니다.
4. TCP/IP 패키지가 원격측으로의 경로를 찾을 수 없어서 이 전달 시도는 실패했습니다. 예 25–4와는 반대로 DNS가 대상 도메인 이름인 `some.org`의 문제를 나타내는 것이 아니라, "no route to host" 오류 메시지가 전송측과 수신측 사이에 어떤 네트워크 문제가 있음을 나타냅니다.
5. MTA 주기적 작업이 다음에 전달을 재시도하면 또 실패하게 됩니다.
6. 이제 파일 이름은 두 번째 시도임을 나타내는 `ZY*`로 바뀝니다.
7. 세 번째로 시도가 실패한 경우 파일 이름은 `ZX*`입니다.

예 25–7 MTA 로깅 – 복수 전달 시도 *(계속)*

8. 다음에 주기적 작업이 재시도하는 전달이 실패하면, 이번에는 TCP/IP 패키지가 원격 SMTP 서버에 도달할 수 없다고 표시하는 것이 아니라 원격 SMTP 서버가 연결을 설정하지 않음을 나타냅니다. 원격측에서 네트워크 문제는 해결했지만 아직 SMTP 서버를 실행하지 않았을 수 있습니다. 즉, 해당 SMTP 서버가 다른 메시지를 처리할 수 없어서 MTA가 연결을 시도할 때 그 연결을 설정하지 못했을 수 있습니다.
9. 마지막으로 메시지가 대기열에서 제외됩니다.

25.3.4.8 MTA 로깅 – 변환 채널을 통해 라우팅된 받는 SMTP 메시지

이 예는 메시지가 변환 채널을 통해 라우팅되는 경우를 보여 줍니다. 해당 사이트에는 다음과 같은 CONVERSIONS 매핑 테이블이 있는 것으로 가정합니다.

```
CONVERSIONS
   IN-CHAN=tcp_local;OUT-CHAN=ims-ms;CONVERT Yes
```

이 예에서는 옵션을 LOG_FILENAME=1 및 LOG_MESSAGE_ID=1로 설정한 것으로 가정합니다.

예 25–8 MTA 로깅 – 변환 채널을 통해 라우팅된 받는 SMTP 메시지

```
26-Feb-2007 15:31:04.17 tcp_local    conversion   EE 1   (1)
amy@siroe.edu rfc822;bert@sesta.com bert@ims-ms-daemon
/opt/SUNWmsgsr/data/queue/conversion/ZZf0s090wFwx2.01
<0JE100206Z7J5F00@siroe.edu>

26-Feb-2007 15:31:04.73 conversion   ims-ms       E 1          (2)
amy@siroe.edu rfc822;bert@sesta.com bert@ims-ms-daemon
/opt/SUNWmsgsr/data/queue/ims-ms/007/ZZf0s090wMwq1.00
<0JE100206Z7J5F00@siroe.edu>

26-Feb-2007 15:31:04.73 conversion               D 1          (3)
amy@siroe.edu rfc822;bert@sesta.com bert@ims-ms-daemon
/opt/SUNWmsgsr/data/queue/conversion/ZZf0s090wFwx2.01
<0JE100206Z7J5F00@siroe.edu>

26-Feb-2007 15:31:04.73 ims-ms                   D 1          (4)
amy@siroe.edu rfc822;bert@sesta.com bert@ims-ms-daemon
/opt/SUNWmsgsr/data/queue/ims-ms/007/ZZf0s090wMwq1.00
<0JE100206Z7J5F00@siroe.edu>
```

1. 외부 사용자 amy@siroe.edu가 보낸 메시지는 ims-ms 채널 수신자 bert@sesta.com으로 주소 지정됩니다. 하지만 CONVERSIONS 매핑 항목 때문에 메시지는 초기에 변환 채널의 대기열에 포함됩니다(ims-ms 채널로 직접 전송되지 않음).
2. 변환 채널이 실행되고 메시지를 ims-ms 채널 대기열에 포함시킵니다.

예 25–8 MTA 로깅 – 변환 채널을 통해 라우팅된 받는 SMTP 메시지 *(계속)*

3. 그런 다음에는 변환 채널이 메시지를 대기열에서 제외할 수 있습니다(오래된 메시지 파일 삭제).
4. 그리고 마지막으로 ims-ms 채널이 메시지를 대기열에서 제외(전달)합니다.

25.3.4.9 MTA 로깅 예: 아웃바운드 연결 로깅

이 예는 LOG_CONNECTION=3을 통해 연결 로깅이 활성화된 경우 보내는 메시지에 대한 로그 출력을 보여 줍니다. 이 예에서도 LOG_PROCESS=1, LOG_MESSAGE_ID=1 및 LOG_FILENAME=1이라고 가정합니다. 이 예는 사용자 adam@sesta.com이 세 명의 수신자(bobby@hosta.sesta.com, carl@hosta.sesta.com 및 dave@hostb.sesta.com)에게 같은 메시지를 보내는 경우를 보여 줍니다(각 메시지 사본의 메시지 아이디는 동일함). 이 예에서는 일반적으로 채널이 그러하듯이 single_sys 채널 키워드로 표시된 tcp_local 채널로 메시지가 나가는 것으로 가정합니다. 따라서 (1), (2), (3)에서 볼 수 있듯이 각 수신자 집합에 대해 별도의 메시지 파일이 디스크에 별도의 호스트 이름으로 생성됩니다. 여기서 bobby@hosta.sesta.com 및 carl@hosta.sesta.com 수신자는 같은 메시지 파일에 저장되지만 dave@hostb.sesta.com 수신자는 다른 메시지 파일에 저장됩니다.

예 25–9 MTA 로깅: 아웃바운드 연결 로깅

```
28-Feb-2007 09:13:19.18 409f.3.1 tcp_intranet tcp_local    EE 1
adam@sesta.com rfc822;bobby@hosta.sesta.com bobby@hosta.sesta.com
/opt/SUNWmsgsr/data/queue/tcp_local/000/ZZf0s4g0G2Zt0.00           (1)
<0JE500C0371HRJ00@sesta.com>
siroe.com (siroe.com [192.160.253.66])

28-Feb-2007 09:13:19.18 409f.3.1 tcp_intranet tcp_local    EE 1
adam@sesta.com rfc822;carl@hosta.sesta.com carl@hosta.sesta.com
/opt/SUNWmsgsr/data/queue/tcp_local/000/ZZf0s4g0G2Zt0.00           (2)
<0JE500C0371HRJ00@sesta.com>
siroe.com (siroe.com [192.160.253.66])

28-Feb-2007 09:13:19.19 409f.3.2 tcp_intranet tcp_local    EE 1
adam@sesta.com rfc822;dave@hostb.sesta.com dave@hostb.sesta.com
/opt/SUNWmsgsr/data/queue/tcp_local/004/ZZf0s4g0G2Zt1.00           (3)
<0JE500C0371HRJ00@sesta.com>
siroe.com (siroe.com [192.160.253.66])

28-Feb-2007 09:13:19.87 40a5.2.0 tcp_local    -  O                 (4)
TCP|206.184.139.12|5900|206.184.139.66|25
SMTP/hostb.sesta.com/mailhub.sesta.com                             (5)

28-Feb-2007 09:13:20.23 40a5.3.4 tcp_local    -  O                 (6)
TCP|206.184.139.12|5901|206.184.139.70|25
```

예 25–9 MTA 로깅: 아웃바운드 연결 로깅 *(계속)*

```
SMTP/hosta.sesta.com/hosta.sesta.com                                    (7)

28-Feb-2007 09:13:20.50 40a5.2.5 tcp_local                 DE 1
adam@sesta.com rfc822;bobby@hosta.sesta.com bobby@hosta.sesta.com
/opt/SUNWmsgsr/data/queue/tcp_local/000/ZZf0s4g0G2Zt0.00
<0JE500C0371HRJ00@sesta.com>
hosta.sesta.com dns;hosta.sesta.com                                     (8)
(TCP|206.184.139.12|5901|206.184.139.70|25)
(hosta.sesta.com -- Server ESMTP [Sun Java System Messaging Server
6.2-8.01 [built Feb 16 2007]])
smtp;250 2.1.5 bobby@hosta.sesta.com and options OK.

28-Feb-2007 09:13:20.50 40a5.2.5 tcp_local                 DE 1
adam@sesta.com rfc822;carl@hosta.sesta.com carl@hosta.sesta.com
/opt/SUNWmsgsr/data/queue/tcp_local/000/ZZf0s4g0G2Zt0.00
<0JE500C0371HRJ00@sesta.com>
hosta.sesta.com dns;hosta.sesta.com
(TCP|206.184.139.12|5901|206.184.139.70|25)
(hosta.sesta.com -- Server ESMTP [Sun Java System Messaging Server
6.2-8.01 [built Feb 16 2007]])
smtp;250 2.1.5 carl@hosta.sesta.com and options OK.

28-Feb-2007 09:13:20.50 40a5.2.6 tcp_local    -  C             (9)
TCP|206.184.139.12|5901|206.184.139.70|25
SMTP/hosta.sesta.com/hosta.sesta.com

28-Feb-2007 09:13:21.13 40a5.3.7 tcp_local                 DE 1
adam@sesta.com rfc822;dave@hostb.sesta.com dave@hostb.sesta.com
/opt/SUNWmsgsr/data/queue/tcp_local/004/ZZf0s4g0G2Zt1.00
<0JE500C0371HRJ00@sesta.com>
mailhub.sesta.com dns;mailhub.sesta.com
(TCP|206.184.139.12|5900|206.184.139.66|25)
(mailhub.sesta.com ESMTP Sendmail ready at Tue, 27 Feb 2007 22:19:40 GMT)
smtp;250 2.1.5 <dave@hostb.sesta.com>... Recipient ok

28-Feb-2007 09:13:21.33 40a5.3.8 tcp_local    -  C             (10)
TCP|206.184.139.12|5900|206.184.139.66|25
SMTP/hostb.sesta.com/mailhub.sesta.com
```

1. 메시지가 첫 번째 수신자의 대기열에
2. 포함됩니다. 그리고 두 번째 수신자의 대기열에
3. 포함됩니다. 그리고 세 번째 수신자의 대기열에 포함됩니다.
4. LOG_CONNECTION=3으로 설정하면 MTA가 이 항목을 기록합니다. 마이너스 기호(-)는 이 항목이 보내는 연결을 참조한다는 것을 나타냅니다. 0는 이 항목이 연결 열기에 해당한다는 것을 나타냅니다. 이러한 별도의 연결을 열 때 다중 스레드 TCP/IP 채널에

예 25–9 MTA 로깅: 아웃바운드 연결 로깅 *(계속)*

같은 프로세스가 사용되기 때문에(열기는 스레드 2 및 스레드 3에 의해 수행되지만) 여기서 프로세스 아이디는 모두 40a5임을 알 수 있습니다.

5. 두 개의 개별적인 원격 시스템에 연결해야 하기 때문에 별도 스레드의 다중 스레드 SMTP 클라이언트는 각각(이 항목에서는 첫 번째, 7에 표시된 두 번째)에 대한 연결을 엽니다. 이 항목 부분은 전송 및 대상 IP 번호와 포트 번호를 표시하며 초기 호스트 이름 및 DNS 조회로 발견된 호스트 이름을 표시합니다. SMTP/ *initial-host*/*dns-host* 절에서는 초기 호스트 이름이 모두 표시되며, 초기 호스트 이름에 대한 DNS MX 레코드 조회를 수행한 뒤에 사용된다는 것을 알 수 있습니다. mailhub.sesta.com은 hostb.sesta.com의 MX 서버입니다.
6. 다중 스레드 SMTP 클라이언트는 동일한 프로세스를 통해 별도의 스레드에서 두 번째 시스템에 대한 연결을 엽니다.
7. 두 개의 개별적인 원격 시스템에 연결해야 하기 때문에 별도 스레드의 다중 스레드 SMTP 클라이언트는 각각(이 항목에서는 두 번째, 5에 표시된 첫 번째)에 대한 연결을 엽니다. 이 항목 부분은 전송 및 대상 IP 번호와 포트 번호를 표시하며 초기 호스트 이름 및 DNS 조회로 발견된 호스트 이름을 표시합니다. 이 예에서는 hosta.sesta.com 시스템이 직접 메시지를 수신합니다.
8. 특정 연결 항목 이외에도 LOG_CONNECTION=3으로 설정하면 여기에서 예로 표시하는 대로 정규 메시지 항목에 연결 관련 정보가 포함됩니다.
9. LOG_CONNECTION=3으로 설정하면 MTA는 이 항목을 기록합니다. 메시지가 대기열에 포함된 뒤(이 예에서는 bobby와 carl 메시지) 이 항목의 C에 표시하는 대로 연결이 닫힙니다.
10. 메시지 전달(이 예에서는 dave)이 완료되었기 때문에 mailhub.sesta.com 연결이 닫혔습니다.

25.3.4.10 MTA 로깅 예: 인바운드 연결 로깅

이 예는 LOG_CONNECTION=3을 통해 연결 로깅이 활성화된 경우 받는 SMTP 메시지에 대한 로그 출력을 보여 줍니다.

예 25–10 MTA 로깅 – 인바운드 연결 로깅

```
28-Feb-2007 11:50:59.10 tcp_local + O                  (1)
TCP|206.184.139.12|25|192.160.253.66|1244 SMTP         (2)

28-Feb-2007 11:51:15.12 tcp_local    ims-ms    EE 1
service@siroe.com rfc822;adam@sesta.com adam@ims-ms-daemon
THOR.SIROE.COM (THOR.SIROE.COM [192.160.253.66])       (3)

28-Feb-2007 11:51:15.32 ims-ms                  D 1
service@siroe.com rfc822;adam@sesta.com adam@ims-ms-daemon
```

예 25–10 MTA 로깅 – 인바운드 연결 로깅 *(계속)*

```
28-Feb-2007 11:51:15.66 tcp_local + C                  (4)
TCP|206.184.139.12|25|192.160.253.66|1244 SMTP
```

1. 원격 시스템이 연결을 엽니다. `O` 문자는 이 항목이 연결 열기에 관련되어 있음을 나타냅니다. `+` 문자는 이 항목이 받는 연결에 관련되어 있음을 나타냅니다.
2. 연결에 대한 IP 번호와 포트가 표시됩니다. 이 항목에서 수신 시스템(로그 파일 항목을 만드는 시스템)의 IP 주소는 206.184.139.12이고 포트 25로 연결됩니다. 송신 시스템의 IP 주소는 192.160.253.66이고 포트 1244에서 전송됩니다.
3. 받는 TCP/IP 채널(`tcp_local`)에서 `ims-ms` 채널 수신자로 메시지를 대기열에 포함하는 항목에서는 `LOG_CONNECTION=3`이 활성화되어 있기 때문에 기본값 이외의 정보를 포함할 수 있습니다. 특히 전송 시스템이 HELO 또는 EHLO 행에 표시한 이름, 연결 IP 번호에 대한 DNS 역조회로 발견된 전송 시스템 이름 및 전송 시스템의 IP 주소가 모두 기록되어 있습니다. 이 기능에 영향을 주는 채널 키워드에 대한 설명은 12 장 동작을 참조하십시오.
4. 인바운드 연결이 닫힙니다. `C` 문자는 이 항목이 연결 닫기에 관련되어 있음을 나타냅니다. `+` 문자는 이 항목이 받는 연결에 관련되어 있음을 나타냅니다.

25.3.5 디스패처 디버깅 활성화

디스패처 오류 및 디버깅 출력(활성화된 경우)은 MTA 로그 디렉토리의 `dispatcher.log` 파일에 기록됩니다. 디스패처 구성 정보는 *msg-svr-base* `/config/dispatcher.cnf` 파일에 지정됩니다. 기본 구성 파일은 설치 시 작성되며 변경 없이 사용할 수 있습니다. 그러나 보안이나 성능상의 이유로 기본 구성 파일을 수정하려는 경우 `dispatcher.cnf` 파일을 편집하여 원하는 사항을 수정할 수 있습니다.

표 25–5 디스패처 디버깅 비트

비트			사용
	16진수 값	10진수 값	
0	x 00001	1	기본 서비스 디스패처 주 모듈 디버깅
1	x 00002	2	추가 서비스 디스패처 주 모듈 디버깅
2	x 00004	4	서비스 디스패처 구성 파일 로깅
3	x 00008	8	기본 서비스 디스패처 기타 디버깅
4	x 00010	16	기본 서비스 디버깅
5	x 00020	32	추가 서비스 디버깅
6	x 00040	64	프로세스 관련 서비스 디버깅

표 25–5 디스패처 디버깅 비트 (계속)

비트			사용
	16진수 값	10진수 값	
7	x 00080	128	사용되지 않습니다.
8	x 00100	256	기본 서비스 디스패처 및 프로세스 통신 디버깅
9	x 00200	512	추가 서비스 디스패처 및 프로세스 통신 디버깅
10	x 00400	1024	패킷 수준 통신 디버깅
11	x 00800	2048	사용되지 않습니다.
12	x 01000	4096	기본 작업자 프로세스 디버깅
13	x 02000	8192	추가 작업자 프로세스 디버깅
14	x 04000	16384	추가 작업자 프로세스 디버깅(특히 연결 핸드오프)
15	x 08000	32768	사용되지 않습니다.
16	x 10000	65536	서비스 디스패처 I/O에 대한 기본 작업자 프로세스 디버깅
17	x 20000	131072	서비스 디스패처 I/O에 대한 기타 작업자 프로세스 디버깅
20	x 100000	1048576	기본 통계 디버깅
21	x 200000	2097152	추가 통계 디버깅
24	x 1000000	16777216	dispatcher.log 파일에 대한 PORT_ACCESS 거부 기록

▼ 디스패처 오류 디버깅 출력을 활성화하는 방법

1 dispatcher.cnf 파일을 편집합니다.

2 DEBUG 옵션을 -1로 설정합니다.

또한 32비트 디버그 마스크를 16진수로 정의하는 논리 또는 환경 변수 IMTA_DISPATCHER_DEBUG(UNIX)를 FFFFFFFF 값으로 설정할 수 있습니다. 위의 표에는 각 비트의 의미가 설명되어 있습니다.

▼ 디스패처 매개 변수 설정 방법(Solaris)

디스패처 구성 파일에서 제공되는 디스패처 서비스는 다양한 시스템 매개 변수의 요구 사항에 영향을 미칩니다. 시스템의 힙 크기(datasize)는 디스패처의 스레드 스택을 사용하기에 충분해야 합니다.

1 힙 크기(기본 datasize)를 표시하려면 다음 중 하나를 사용합니다.

csh 명령

```
# limit
```

ksh 명령

```
# ulimit -a
```

Solaris 유틸리티

```
# sysdef
```

2 **각 디스패처 서비스에 대해 STACKSIZE*MAX_CONNS를 계산한 다음 각 서비스에 대해 계산된 값을 모두 더합니다. 시스템의 힙 크기는 이 숫자의 두 배 이상이 되어야 합니다.**

25.4 메시지 저장소, Admin 및 Default 서비스 로그 관리

이 절에서는 메시지 저장소(POP, IMAP 및 HTTP), Admin 및 Default 서비스에 대한 로깅을 설명합니다. (표 25–1 참조).

이 서비스에 대해 로그 설정을 지정하고 로그를 봅니다. 지정하는 설정은 기록되는 이벤트 및 기록되는 이벤트의 수에 영향을 미칩니다. 로그 파일을 분석할 때 이러한 설정 및 기타 특성을 사용하여 기록되는 이벤트를 자세히 검색할 수 있습니다.

이 절에는 다음과 같은 하위 절이 포함됩니다.

25.4.1 서비스 로그 특징 이해

이 절에서는 메시지 저장소 및 관리 서비스에 대한 로깅 수준, 기록되는 이벤트 범주, 로그의 파일 이름 규칙, 로그 파일 디렉토리 등과 같은 로그 특징을 설명합니다.

25.4.1.1 로깅 수준

로깅의 수준 또는 우선 순위는 로깅 작업 수행의 세밀도를 정의합니다. 우선 순위 수준이 높을수록 세밀도가 떨어집니다. 즉, 우선 순위가 높은(높은 심각도) 이벤트가 기록됩니다. 수준이 낮으면 세밀도가 높아집니다. 즉, 더 많은 이벤트가 로그 파일에 기록됩니다.

`logfile.`*service*`.loglevel` 구성 매개 변수(790 페이지 “25.4.3 서비스 로깅 옵션 정의 및 설정” 참조)를 사용하여 각 서비스(POP, IMAP, HTTP, Admin 및 Default)마다 별도의 로깅

수준을 설정할 수 있습니다. 또한 로깅 수준을 사용하여 로그 이벤트에 대한 검색을 필터링할 수도 있습니다. 사용 가능한 수준은 표 표 25–6에서 설명합니다. 이러한 로깅 수준은 UNIX `syslog` 기능에 의해 정의된 수준의 하위 집합입니다.

표 25–6 저장소 및 관리 서비스의 로깅 수준

수준	설명
Critical	최소의 로깅 세밀도입니다. 서버 문제 또는 중요한 조건이 발생(예: 서버가 메일함에 액세스할 수 없거나 서버를 실행하려면 라이브러리가 필요한 경우)할 때마다 이벤트가 로그에 기록됩니다.
Error	오류 조건(예: 클라이언트나 다른 서버에 대한 연결 시도가 실패한 경우)이 발생할 때마다 이벤트가 로그에 기록됩니다.
Warning	경고 조건이 발생할 때마다(예: 클라이언트가 보낸 통신을 서버가 인식할 수 없는 경우) 이벤트가 로그에 기록됩니다.
Notice	알림(일반적이지만 중요한 조건)이 발생할 때마다(예: 사용자 로그인 실패 또는 세션 종료시) 이벤트가 로그에 기록됩니다. 이는 기본 로그 수준입니다.
Information	수행되는 모든 중요 작업(예: 사용자가 성공적으로 로그온했거나 메일함을 만들거나 메일함 이름을 변경한 경우)에 대한 이벤트를 로그에 기록합니다.
Debug	가장 세밀한 로깅입니다. 디버깅 용도에만 적합합니다. 문제를 나타내기 위해 각 프로세스나 작업 내의 개별 단계에서 이벤트가 로그에 기록됩니다.

특정 로깅 수준을 선택하면 해당 수준과 그보다 높은(세밀도는 더 낮은) 모든 수준에 해당하는 이벤트가 기록됩니다. 기본 로깅 수준은 `Notice`입니다.

주 – 로깅을 세밀하게 지정할수록 로그 파일이 차지하는 디스크 공간은 많아집니다. 이에 대한 지침은 790 페이지 “25.4.3 서비스 로깅 옵션 정의 및 설정”을 참조하십시오.

25.4.1.2 기록되는 이벤트의 범주

지원되는 각 서비스 또는 프로토콜 내에서 Messaging Server는 기능 또는 기능 영역에 따라 기록되는 이벤트를 범주화합니다. 기록되는 모든 이벤트에는 해당 이벤트를 생성한 기능의 이름이 포함됩니다. 이러한 범주는 검색 중에 이벤트를 필터링하는 데 도움이 됩니다. 표 25–7에는 로깅을 위해 Messaging Server가 인식하는 범주가 나열되어 있습니다.

표 25–7 로그 이벤트가 발생하는 범주

기능	설명
일반적인 문제	이 프로토콜 또는 서비스에 관련된 구분되지 않은 작업입니다.

표 25–7 로그 이벤트가 발생하는 범주 (계속)

기능	설명
LDAP	LDAP 디렉토리 데이터베이스에 액세스하는 Messaging Server에 관련된 작업입니다.
Network	네트워크 연결에 관련된 작업(소켓 오류가 이 범주에 속함)입니다.
Account	사용자 계정에 관련된 작업(사용자 로그인이 이 범주에 속함)입니다.
Protocol	프로토콜별 명령에 관련된 프로토콜 수준 작업(POP, IMAP 또는 HTTP 기능에서 반환되는 오류가 이 범주에 속함)입니다.
Stats	서버 통계의 수집에 관련된 작업입니다.
Store	메시지 저장소 액세스에 관련된 낮은 수준의 작업(읽기/쓰기 오류가 이 범주에 속함)입니다.

로그 검색에서 범주를 필터로 사용하는 예에 대해서는 792 페이지 “25.4.4 서비스 로그 검색 및 보기”를 참조하십시오.

25.4.1.3 서비스 로그 파일 디렉토리

기록되는 모든 서비스에는 로그 파일이 저장되는 하나의 디렉토리가 할당됩니다. 모든 POP 로그 파일과 기타 모든 서비스의 로그 파일과 마찬가지로 모든 IMAP 로그 파일이 함께 저장됩니다. 각 디렉토리의 위치를 정의할 수 있으며 디렉토리에 허용되는 로그 파일의 최대 크기와 수를 지정할 수 있습니다.

로그 파일을 모두 저장할 수 있을 만큼 저장소 용량이 충분한지 확인하십시오. 로그 데이터는 용량이 매우 커질 수 있습니다(특히 낮은 로깅 수준에서).

또한 모든 로그 파일 디렉토리가 백업되고 오버로드되지 않도록 로깅 수준, 로그 회전, 로그 만료 및 서버 백업 정책을 적절하게 정의해야 합니다. 그렇지 않으면 정보가 손실될 수 있습니다. 790 페이지 “25.4.3 서비스 로깅 옵션 정의 및 설정”을 참조하십시오.

25.4.2 서비스 로그 파일 형식 이해

Messaging Server에 의해 생성된 모든 메시지 저장소와 관리 서비스 로그 파일의 내용 형식은 서로 동일합니다. 로그 파일은 여러 행의 텍스트 파일로, 각 행이 기록된 하나의 이벤트를 설명합니다. 지원되는 각 서비스에 대한 모든 이벤트 설명에는 다음과 같은 일반 형식이 있습니다.

dateTime hostName processName[*pid*]: *category logLevel*: *eventMessage*

표 25–8에는 로그 파일 구성 요소가 나열되어 있습니다. 이러한 이벤트 설명의 형식은, 날짜/시간 형식이 다르고 형식에 두 개의 추가 구성 요소(*category*와 *logLevel*)가 추가된다는 점을 제외하면 UNIX `syslog` 기능에 의해 정의된 것과 동일합니다.

표 25–8 저장소 및 관리 로그 파일 구성 요소

구성 요소	정의
dateTime	이벤트가 기록된 날짜 및 시간이며, *dd/mm/yyyy hh:mm:ss* 형식으로 표현됩니다. 여기서 시간대 필드는 GMT로부터의 *+/-hhmm*으로 표현됩니다. 예를 들면 다음과 같습니다. `02/Jan/1999:13:08:21 -0700`
hostName	서버가 실행되고 있는 호스트 시스템의 이름입니다(예: `showshoe`). **주:**호스트에 두 개 이상의 Messaging Server 인스턴스가 있는 경우 프로세스 아이디(pid)를 사용하여 각 인스턴스의 기록된 이벤트를 구분할 수 있습니다.
processName	이벤트를 생성한 프로세스의 이름입니다(예: `cgi_store`).
pid	이벤트를 생성한 프로세스의 프로세스 아이디입니다(예: `18753`).
category	이벤트가 속하는 범주(예: `General`)입니다(예 25–5 참조).
logLevel	이벤트가 표시되는 로깅 수준(예: `Notice`)입니다(예 25–4 참조).
eventMessage	이벤트별 설명 메시지로 길이는 임의적일 수 있습니다(예: `Log created (894305624)`).

다음은 기록된 세 개의 이벤트 예입니다.

```
02/May/1998:17:37:32 -0700 showshoe cgi_store[18753]:
 General Notice:
   Log created (894155852)

04/May/1998:11:07:44 -0400 xyzmail cgi_service[343]: General Error:
   function=getserverhello|port=2500|error=failed to connect

03/Dec/1998:06:54:32 +0200 SiroePost imapd[232]: Account Notice:
   close [127.0.0.1] [unauthenticated] 1998/12/3 6:54:32
   0:00:00 0 115 0
```

IMAP와 POP 이벤트 항목은 세 개의 번호로 끝날 수 있습니다. 위의 예에서는 0 115 0입니다. 첫 번째 번호는 클라이언트가 전송한 바이트 수이고, 두 번째 번호는 서버가 전송한 바이트 수이고, 세 번째 숫자는 선택한 메일함(POP의 경우 항상 1)입니다.

Log Viewer 창에서 로그 파일을 볼 때는 특정 로깅 수준이나 범주 또는 특정 프로세스 아이디 등과 같은 이벤트의 특정 구성 요소를 검색하여, 표시되는 이벤트를 제한할 수 있습니다. 자세한 내용은 792 페이지 “25.4.4 서비스 로그 검색 및 보기”를 참조하십시오.

각 로그 항목의 이벤트 메시지에는 기록되는 이벤트 유형에 한정된 형식이 사용됩니다. 즉, 각 서비스는 이벤트 메시지에 표시되는 내용을 정의합니다. 대부분의 이벤트 메시지는 간단하며 설명적입니다. 다른 메시지는 좀 더 복잡할 수 있습니다.

25.4.3 서비스 로깅 옵션 정의 및 설정

관리에 필요한 가장 적합한 메시지 저장소와 관리 서비스 로깅 구성을 정의할 수 있습니다. 이 절에서는 최적의 구성과 정책을 결정하는 데 도움이 되는 문제에 대해 설명하고 이를 구현하는 방법에 대해 설명합니다.

25.4.3.1 유연한 로깅 구조

로그 파일의 이름 지정 스키마(*service.sequenceNum.timeStamp*)는 유연한 로그 회전과 백업 정책을 지정하는 데 도움이 됩니다. 각 서비스에 대한 이벤트가 서로 다른 파일에 기록되기 때문에 문제를 빠르게 차단할 수 있습니다. 또한 파일 이름의 일련 번호가 계속 커지고 타임스탬프가 항상 고유하기 때문에 제한된 일련 번호 집합을 모두 사용한 후 새 로그 파일이 기존 로그 파일을 덮어쓰지 않습니다. 대신 보다 유연한 수명 제한, 파일 수 또는 전체 저장소 용량에 도달하였을 때만 기존 로그 파일을 삭제하거나 덮어쓰게 됩니다.

Messaging Server는 관리 작업을 단순화하고 백업 작업을 수월히 하는 로그 파일의 자동 회전 기능을 지원합니다. 이후의 기록되는 이벤트를 저장하기 위해 수동으로 현재 로그 파일을 지우고 새 파일을 만들 필요가 없습니다. 서버를 중단하거나 새 로그 파일을 시작하도록 서버에 수동으로 알리지 않고도 디렉토리에서 현재 로그 파일을 제외한 모든 파일을 언제든지 백업할 수 있습니다.

로깅 정책 설정 시 전체 로그 저장소 제한, 최대 로그 파일 수, 개별 파일 크기, 최대 파일 수명 및 로그 파일 회전 비율을 제어하는 옵션을 설정할 수 있습니다(각 옵션별).

25.4.3.2 원하는 옵션 계획

로그 파일의 회전이나 삭제를 시작할 수 있는 두 개 이상의 제한을 설정해야 합니다. 먼저 도달한 제한이 작업을 제어합니다. 예를 들어 최대 로그 파일 크기가 3.5MB이고 매일 새 로그가 생성되도록 지정한 경우, 로그 데이터가 24시간 동안 3.5MB 이상 작성되면 매일 하나 이상의 로그 파일이 생성됩니다. 또한 최대 로그 파일 수가 10이고 최대 수명이 8일인 경우, 로그 회전이 더 빠르게 수행되면 8일이 되기 전에 10개의 파일이 생성되므로 로그 파일의 수명 제한에는 도달하지 않게 됩니다.

Messaging Server 관리 로그에 대해 제공되는 다음 기본값을 계획의 시작점으로 사용할 수 있습니다.

디렉토리의 최대 로그 파일 수: 10

최대 로그 파일 크기: 2MB

모든 로그 파일에 허용되는 총 최대 크기: 20MB

허용되는 빈 디스크 최대 공간: 5MB

로그 롤오버 시간: 1일

만료 전 최대 수명: 7일

로깅 수준: Notice

이 구성에서는 서버 관리 로그 데이터가 매일 약 2MB 정도 누적될 것이며, 1주에 한 번 백업하고, 관리 로그의 저장소에 할당된 총 크기는 최소한 25MB가 되는 것으로 가정합니다. 로깅 수준이 더 세밀한 경우 이 설정이 충분하지 않을 수 있습니다.

POP, IMAP 또는 HTTP 로그의 경우에도 처음에는 동일한 값을 사용할 수 있습니다. 모든 서비스에 대략 여기에 표시된 기본값과 동일한 로그 저장소 요구 사항이 있는 경우 처음에는 총 로그 저장소 용량을 약 150MB로 설정해야 합니다. 이는 일반적인 저장소 요구 사항이며 실제 요구 사항은 환경에 따라 크게 다를 수 있습니다.

25.4.3.3 로깅 옵션 이해

명령줄을 사용하여 메시지 저장소 로깅 구성을 제어하는 옵션을 구성할 수 있습니다.

이러한 옵션의 최적 설정은 로그 데이터가 누적되는 비율에 따라 다릅니다. 1MB의 저장소 공간에는 4,000개에서 10,000개의 로그 항목이 들어갈 수 있습니다. Notice와 같은 보다 세밀한 로깅 수준에서는, 작업량이 중간 정도인 서버가 매주 수백 MB의 로그 데이터를 생성할 수 있습니다. 이 경우 다음과 같은 방법을 사용할 수 있습니다.

- 저장소 제한에 일치하는 로깅 수준 즉, 평가 수준을 설정하면 로그 데이터는 저장소 제한을 평가하는 데 사용되는 비율과 비슷하게 누적됩니다.
- 검색 성능에 영향이 없도록 로그 파일 크기를 정의합니다. 또한 회전 일정과 총 저장소 제한에 맞게 조정합니다. 로그 항목이 누적되는 비율이 지정되면, 그 최대값은 회전이 자동으로 수행될 때 예상되는 누적 양보다 약간 크게 설정합니다. 그리고 최대 파일 크기와 최대 파일 수를 곱한 값이 전체 저장소 제한과 거의 일치하도록 해야 합니다.

  예를 들어 IMAP 로그 회전이 매일 수행되며 IMAP 로그 데이터의 예상 누적량이 일일 3MB이고 IMAP 로그의 전체 저장소 제한이 25MB인 경우, 최대 IMAP 로그 파일 크기는 3.5MB로 설정할 수 있습니다. 이 예에서 일부 로그 데이터가 너무 빠르게 누적되어 모든 로그 파일이 최대 크기에 도달하고 최대 로그 파일 수에 도달하면 일부 로그 데이터가 손실될 수 있습니다.
- 서버 백업이 매주 수행되며 IMAP 로그 파일을 매일 회전하는 경우 최대 IMAP 로그 파일 수를 약 10(개별 로그 크기 제한이 초과되는 경우의 회전을 고려한 값)으로 지정하고 최대 수명은 7-8일로 지정할 수 있습니다.
- 전체 저장소 제한은 하드웨어 용량 내에서 선택하고 해당 서버의 백업 일정에 맞게 조정해야 합니다. 로그 데이터의 예상 누적 비율을 평가하고 안전 계수를 추가한 다음 총 저장소 제한이 서버 백업 사이의 간격을 초과하지 않도록 정의합니다.

  예를 들어 매일 평균 3MB의 IMAP 로그 파일 데이터가 누적되고 서버 백업이 매주 수행되는 경우, IMAP 로그의 저장소 제한을 25 - 30MB로 설정할 수 있습니다(디스크 저장소 용량이 충분한 경우).

- 안전을 위해, 로그 파일이 저장될 볼륨에 허용되는 빈 디스크의 최소 공간을 선택해야 합니다. 이렇게 하면 로그 파일 크기 이외의 요소로 인해 볼륨이 꽉 차게 되면 꽉 찬 디스크에 로그 데이터를 쓰려고 할 때 발생하는 실패가 발생하기 전에 오래된 로그 파일이 삭제됩니다.

25.4.4 서비스 로그 검색 및 보기

로그 파일은 메시지 저장소와 관리 로그 데이터를 볼 수 있는 기본 인터페이스를 제공합니다. 특정 서비스에 대한 로그 파일은 날짜 순으로 나열됩니다. 검색할 로그 파일을 선택한 다음에는 검색 매개 변수를 지정하여 개별 이벤트에 대한 검색 범위를 좁힐 수 있습니다.

25.4.4.1 검색 매개 변수

다음은 로그 데이터를 보기 위해 지정할 수 있는 유용한 검색 매개 변수입니다.

- **시간.** 이벤트를 검색할 특정 시간의 시작과 끝을 지정하거나 검색할 일 수(현재 날짜 이전)를 지정할 수 있습니다. 서버 충돌 시간 또는 알고 있는 다른 발생 시간까지 기록된 이벤트를 확인할 시간 범위를 지정할 수 있습니다. 또는 현재 로그 파일 중 오늘의 이벤트만 보도록 범위를 지정할 수도 있습니다.
- **로깅 수준.** 로깅 수준(786 페이지 "25.4.1.1 로깅 수준" 참조)을 지정할 수 있습니다. 예를 들어 서버 다운의 원인을 알려면 Critical을 선택하고 실패한 프로토콜 호출을 찾으려면 Error를 선택합니다.
- **기능.** 문제가 포함된 기능을 지정할 수 있습니다(787 페이지 "25.4.1.2 기록되는 이벤트의 범주" 참조). 예를 들어 디스크 오류 때문에 서버가 크래시된 것으로 생각되면 Store를 선택하고, IMAP 프로토콜 명령 오류에 문제가 있는 경우 Protocol을 선택합니다.
- **텍스트 검색 패턴.** 보다 세밀한 검색을 위해 텍스트 검색 패턴을 지정할 수 있습니다. 알고 있는 이벤트 시간, 프로세스 이름, 프로세스 아이디 및 이벤트 메시지의 일부(예: 원격 호스트 이름, 함수 이름, 오류 번호 등) 등과 같이 와일드카드 형식의 검색으로 표현할 수 있는 이벤트의 모든 구성 요소(788 페이지 "25.4.2 서비스 로그 파일 형식 이해" 참조)를 포함시킬 수 있습니다.

  검색 패턴에는 다음의 특수 문자 및 와일드카드 문자가 포함될 수 있습니다.

* 모든 문자 세트(예: `*.com`)

? 모든 단일 문자(예: `199?`)

[*nnn*] *nnn* 집합에 속하는 모든 문자(예: `[aeiou]`)

[^*nnn*] *nnn* 집합에 속하지 않는 모든 문자(예: `[^aeiou]`)

[*n-m*] *n-m* 범위의 모든 문자(예: `[A-Z]`)

[^*n-m*] *n-m* 범위에 속하지 않는 모든 문자(예: `[^0-9]`)

\ 이스케이프 문자: *, ?, [또는] 앞에 입력하여 리터럴로 사용

주: 검색은 대소문자를 구분합니다.

로그를 볼 때 로깅 수준과 기능을 조합하는 예는 다음과 같습니다.

- Account 기능(및 Notice 수준)을 지정하여 실패한 로그인을 표시합니다. 그러면 잠재적 보안 침해를 조사할 때 유용합니다.
- Network 기능(및 모든 로깅 수준)을 지정하여 연결 문제를 조사합니다.
- 모든 기능(및 Critical 로깅 수준)을 조사하여 서버의 기능에 관련된 기본적인 문제를 찾습니다.

25.4.5 서비스 로그 작업

이 절에서는 로그 검색 및 보기를 위한 `configutil` 명령을 사용하여 서비스 로그에 대한 작업을 수행하는 방법에 대해 설명합니다. 이 장은 다음 내용으로 구성되어 있습니다.

▼ 서비스 로그를 syslog에 전송하는 방법

- **`configutil` 명령을 `syslogfacility` 옵션과 함께 실행합니다.**

  `configutil -o logfile.` *service*`.syslogfacility -v` *value*

  여기서 *service*는 `admin`, `pop`, `imap`, `imta` 또는 `http` 이고 *value*는 `user`, `mail`, `daemon`, `local0` ~ `local7` 또는 none입니다.

  값을 설정하면 메시지는 설정 값에 따라 `syslog` 기능에 기록되며 다른 모든 로그 파일 서비스 옵션은 무시됩니다. 옵션이 설정되지 않았거나 값이 `none`이면 로깅은 Messaging Server 로그 파일을 사용합니다.

25.4.5.1 HTTP 로깅 비활성화 방법

시스템이 HTTP 메시지 액세스 즉, 웹 메일을 지원하지 않는 경우 다음 변수를 설정하여 HTTP 로깅을 비활성화할 수 있습니다. 시스템이 웹 메일을 지원해야 하는 경우(예: Messenger Express) 이 변수를 설정하지 마십시오.

- 다음 configutil 명령을 실행합니다.

```
configutil -o service.http.enable -v no
configutil -o service.http.enablesslport -v no
```

▼ 서버 로그 수준을 설정하는 방법

- **다음 configutil 명령을 실행합니다.**

  `configutil -o logfile.`*service* `.loglevel -v` *level*

  여기서 *service*는 admin, pop, imap, imta 또는 http이고 *loglevel*은 Nolog, Critical, Error, Warning, Notice, Information 또는 Debug입니다.

▼ 서버 로그 파일의 디렉토리 경로를 지정하는 방법

- **다음 configutil 명령을 실행합니다.**

  `configutil -o logfile.`**service**`.logdir -v` **dirpath**

▼ 각 서비스 로그의 최대 파일 크기를 지정하는 방법

- **다음 configutil 명령을 실행합니다.**

  `configutil -o logfile.`**service**`.maxlogfilesize -v` **size**

  여기서 *size*는 바이트 수를 지정합니다.

▼ 서비스 로그 회전 일정을 지정하는 방법

- **다음 configutil 명령을 실행합니다.**

  `configutil -o logfile.`**service**`.rollovertime -v` **number**

  여기서 *number*는 초를 지정합니다.

▼ 디렉토리당 서비스 로그 파일의 최대 수를 지정하는 방법

- **다음 configutil 명령을 실행합니다.**

  `configutil -o logfile.`**service**`.maxlogfiles -v` **number**

  여기서 *number*는 로그 파일의 최대 수를 지정합니다.

▼ 저장소 제한을 지정하는 방법

- **다음 `configutil` 명령을 실행합니다.**

  ```
  configutil -o logfile.service.maxlogsize -v number
  ```

  여기서 *number*는 바이트 수를 지정합니다.

▼ 유지할 빈 디스크 공간의 최소 크기를 지정하는 방법

- **다음 `configutil` 명령을 실행합니다.**

  ```
  configutil -o logfile.service.minfreediskspace -v number
  ```

  여기서 *number*는 바이트 수를 지정합니다.

25.4.5.2 로그 파일이 만료되는 시기를 지정하는 방법

```
configutil -o logfile.service.expirytime -v number
```

여기서 *number*는 초를 지정합니다.

25.4.6 메시지 저장소 로깅에 메시지 추적 사용

MTA가 메시지를 추적하는 것과 비슷한 방법으로 메시지 저장소 로깅을 사용하여 메일 아이디로 메시지를 추적할 수 있습니다. 이 방식으로 메시지를 추적하면 메시지의 수명 주기의 중요 이벤트를 추적할 수 있습니다.

메시지 저장소 로그의 메시지를 추적하려면 일반 로깅 구성 외에도 메시지 추적을 구성해야 합니다. 기본적으로 메시지 추적은 활성화되지 않습니다.

주 – 메시지 추적에는 많은 양의 디스크 공간이 필요합니다. 디스크 공간이 충분하지 않을 경우 이 기능을 활성화하지 마십시오.

메시지 저장소 로깅에서는 다음 작업을 추적할 수 있습니다.

- 추가 - 메시지 저장소 라이브러리가 메시지를 폴더에 추가하는 기본 방법입니다. 추가 추적은 메시지 저장소에 들어오는 메시지를 보여 줍니다.
- 가져오기 - 최종 사용자에 대한 메시지나 메시지의 일부분을 검색하는 IMAP 명령입니다. 메시지 추적의 경우 서비스에서 최종 사용자가 읽을 메시지를 검색하는 경우로까지 그 의미가 확장됩니다.

  메시지 추적에서는 종종 메시지 헤더를 읽을 때 추적을 사용하지 않을 수 있으므로 본문 가져오기가 메시지 본문의 일부를 검색할 때를 나타냅니다.
- 정리: 이 경우 임의의 서비스가 사용자 폴더에서 메시지를 제거하는 때를 나타내도록 확장되는 IMAP 용어입니다.

▼ 메시지 추적을 활성화하는 방법

- **다음 `configutil` 명령을 실행합니다.**

  ```
  configutil -o local.mstrace.active -v   yes
  ```

  메시지 추적 정보는 각 프로세스의 기본 로그에 기록됩니다. IMAP 가져오기는 `imap` 로그 파일에 표시됩니다. ims_master 추가는 `ims_master` 채널 로그 파일에 표시됩니다.

▼ 메시지 추적을 단일 로그 파일로 리디렉션하는 방법

- **메시지 추적 로깅을 단일 "msgtrace" 로그 파일로 리디렉션하려면 `configutil` 명령을 사용하여 로그 파일 매개 변수를 구성해야 합니다. 다른 로그 파일과 달리 msgtrace 로그 파일은 로컬로 구성됩니다. 예를 들면 다음과 같습니다.**

  ```
  configutil -o "local.logfile.msgtrace.buffersize" -v "0"
  configutil -o "local.logfile.msgtrace.expirytime" -v "604800"
  configutil -o "local.logfile.msgtrace.flushinterval" -v "60"
  configutil -o "local.logfile.msgtrace.logdir" -v "/opt/SUNWmsgsr/data/log"
  configutil -o "local.logfile.msgtrace.loglevel" -v "Information"
  configutil -o "local.logfile.msgtrace.logtype" -v "NscpLog"
  configutil -o "local.logfile.msgtrace.maxlogfiles" -v "10"
  configutil -o "local.logfile.msgtrace.maxlogfilesize" -v "2097152"
  configutil -o "local.logfile.msgtrace.maxlogsize" -v "20971520"
  configutil -o "local.logfile.msgtrace.minfreediskspace" -v "5242880"
  configutil -o "local.logfile.msgtrace.rollovertime" -v "86400"
  ```

▼ 메시지 추적 로깅을 구성 해제하는 방법

- **msgtrace 로그 파일을 구성 해제하려면 `configutil` 명령을 사용하여 해당 구성에 대한 모든 참조를 제거합니다. 예를 들면 다음과 같습니다.**

  ```
  configutil -o "local.logfile.msgtrace.buffersize" -v ""
  configutil -o "local.logfile.msgtrace.expirytime" -v ""
  configutil -o "local.logfile.msgtrace.flushinterval" -v ""
  configutil -o "local.logfile.msgtrace.logdir" -v ""
  configutil -o "local.logfile.msgtrace.loglevel" -v ""
  configutil -o "local.logfile.msgtrace.logtype" -v ""
  configutil -o "local.logfile.msgtrace.maxlogfiles" -v ""
  configutil -o "local.logfile.msgtrace.maxlogfilesize" -v ""
  configutil -o "local.logfile.msgtrace.maxlogsize" -v ""
  configutil -o "local.logfile.msgtrace.minfreediskspace" -v ""
  configutil -o "local.logfile.msgtrace.rollovertime" -v ""
  ```

▼ LMTP 로깅을 구성하는 방법

- **LMTP를 사용하고 단일 "msgtrace" 로그 파일을 사용하지 않을 경우에는 마찬가지로 `tcp_lmtp_server` 로그 파일을 로컬로 구성해야 합니다. LMTP를 사용하지 않거나, 메시지 추적을 사용하지 않거나, "msgtrace" 로그 파일에서 메시지 추적을 사용할 경우에는 LMTP 메시지 저장소측 로그를 초기화할 필요가 없습니다. LMTP는 이미 MTA 정보를 별도로 기록합니다. 예를 들면 다음과 같습니다.**

```
configutil -o "local.logfile.tcp_lmtp_server.buffersize" -v "0"
configutil -o "local.logfile.tcp_lmtp_server.expirytime" -v "604800"
configutil -o "local.logfile.tcp_lmtp_server.flushinterval" -v "60"
configutil -o "local.logfile.tcp_lmtp_server.logdir" -v \
   "/opt/SUNWmsgsr/data/log"
configutil -o "local.logfile.tcp_lmtp_server.loglevel" -v "Information"
configutil -o "local.logfile.tcp_lmtp_server.logtype" -v "NscpLog"
configutil -o "local.logfile.tcp_lmtp_server.maxlogfiles" -v "10"
configutil -o "local.logfile.tcp_lmtp_server.maxlogfilesize" -v "2097152"
configutil -o "local.logfile.tcp_lmtp_server.maxlogsize" -v "20971520"
configutil -o "local.logfile.tcp_lmtp_server.minfreediskspace" \
   -v "5242880"
configutil -o "local.logfile.tcp_lmtp_server.rollovertime" -v "86400"
```

25.4.7 메시지 저장소 로깅 예

메시지 저장소 로그 파일에 기록되는 정확한 필드 형식 및 필드 목록은 설정하는 로깅 옵션에 따라 다릅니다. 이 기능은 클라이언트 문제를 디버깅하는 데 유용합니다. 예를 들어, 사용자가 메시지 액세스 클라이언트가 제대로 작동하지 않는다고 불평할 경우 이 기능을 사용하여 액세스 클라이언트와 Messaging Server 사이의 상호 작용을 추적할 수 있습니다. 628 페이지 "20.14.1.3 원격 측정을 사용하여 사용자 IMAP/POP/Webmail 세션 검사" 을 참조하십시오.

25.4.8 메시지 저장소 로깅 예

메시지 저장소 로그 파일에 기록되는 정확한 필드 형식 및 필드 목록은 설정하는 로깅 옵션에 따라 다릅니다. 이 절에서는 일반적인 로그 항목의 몇 가지 예를 보여 줍니다.

25.4.8.1 메시지 저장소 로깅 예: 잘못된 비밀번호

사용자가 잘못된 비밀번호를 입력하면 "사용자 없음" 메시지와 달리 "인증" 실패가 기록됩니다. 보안상의 이유 때문에 "사용자 없음" 메시지가 클라이언트에게 전달되지만 기록되는 것은 실제 이유(잘못된 비밀번호)입니다.

예 25–11 메시지 저장소 로깅 – 잘못된 비밀번호

```
 [30/Aug/2004:16:53:05 -0700] vadar imapd[13027]: Account Notice: badlogin:
[192.18.126.64:40718] plaintext user1 authentication failure
```

25.4.8.2 메시지 저장소 로깅 – 비활성화된 계정

다음 예는 비활성화된 계정으로 인해 사용자가 로그인할 수 없는 이유를 보여 줍니다. 또한 비활성화된 계정이 "(inactive)" 또는 "(hold)"로 구분됩니다."

예 25–12 메시지 저장소 로깅 – 비활성화된 계정

```
[30/Aug/2004:16:53:31 -0700] vadar imapd[13027]: Account Notice: badlogin:
[192.18.126.64:40720] plaintext user3 account disabled (hold)
```

25.4.8.3 메시지 저장소 로깅 예: 추가된 메시지

다음 예는 메시지가 폴더에 추가될 때마다 발생하는 추가 메시지를 보여 줍니다. 메시지 저장소 로그는 `ims_master` 및 `lmtp` 채널을 통해 메시지 저장소에 들어오는 모든 메시지를 기록합니다. 사용자 아이디, 폴더, 메시지 크기 및 메시지 아이디의 "추가"가 기록됩니다.

예 25–13 메시지 저장소 로깅 – 추가

```
[31/Aug/2004:16:33:14 -0700] vadar ims_master[13822]: Store Information:append:
user1:user/user1:659:<Roam.SIMC.2.0.6.1093995286.11265.user1@vadar.siroe.com>
```

25.4.8.4 메시지 저장소 로깅 예: 클라이언트가 검색한 메시지

클라이언트가 메시지를 검색하면 메시지 저장소 로그는 "가져오기" 메시지를 기록합니다. 메시지 저장소 로그는 하나 이상의 본문 부분에 대한 클라이언트의 모든 가져오기를 기록합니다. 사용자 아이디, 폴더 및 메시지 아이디의 "가져오기"가 기록됩니다.

예 25–14 메시지 저장소 로깅 – 클라이언트가 검색한 메시지

```
[31/Aug/2004:15:55:26 -0700] vadar imapd[13729]: Store Information:
fetch:user1:user/user1:<Roam.SIMC.2.0.6.1093051161.3655.user1@vad.siroe.com>
```

25.4.8.5 메시지 저장소 로깅 예: 폴더에서 제거된 메시지

예 25–15 메시지 저장소 로깅 예: 폴더에서 제거된 메시지

IMAP 또는 POP 메시지가 폴더에서 제거되지만 시스템에서는 제거되지 않을 경우 메시지 저장소는 "정리" 메시지를 기록합니다. 이 메시지는 사용자가 정리하든 유틸리티가 정리하든 상관없이 기록됩니다. 폴더 및 메시지 아이디의 "정리"가 기록됩니다.

```
31/Aug/2004:16:57:36 -0700] vadar imexpire[13923]: Store Information:
expunge:user/user1:<Roam.SIMC.2.0.6.1090458838.2929.user1@vadar.siroe.com>
```

25.4.8.6 메시지 저장소 로깅 예: 중복된 로그인 메시지

하나의 msgtrace 로그 파일에 대해 메시지 추적을 구성할 경우 imap 및 pop 로그 파일에 표시되는 일반 "로그인" 메시지가 msgtrace 파일에서 중복됩니다. 일반 로그인 메시지는 다음과 같습니다.

예 25–16 메시지 저장소 로깅 – 로그인

```
[30/Aug/2004:16:53:13 -0700] vadar imapd[13027]: Account Information: login
[192.18.126.64:40718] user1 plaintext
```

◆ ◆ ◆ 26 장

26 MTA 문제 해결

이 장에서는 MTA(Message Transfer Agent) 문제 해결에 대한 일반 도구, 방법 및 절차에 대해 설명합니다. 이 장은 다음 내용으로 구성되어 있습니다.

관련 항목인 모니터링 절차는 27 장에 설명되어 있습니다.

주 – 이 장을 읽기 전에 본 설명서의 5장부터 10장과 **Sun Java System Messaging Server Administration Reference**의 MTA 구성 및 명령줄 유틸리티에 대한 장을 검토하십시오.

26.1 문제 해결 개요

MTA 관련 문제 해결의 첫 단계 중 하나는 진단을 시작할 위치를 결정하는 것입니다. 문제에 따라 로그 파일에서 오류 메시지를 볼 수 있습니다. 다른 상황에서는 모든 표준 MTA 프로세스를 확인하고 MTA 구성을 검토하거나 개별 채널을 시작 및 중지할 수 있습니다. 사용 방법에 상관 없이 MTA 관련 문제 해결 시에는 다음 질문을 고려하십시오.

- 구성 또는 환경 문제(예: 디스크 공간 또는 할당량 문제)로 인해 메시지를 받을 수 없었습니까?
- 메시지가 메시지 대기열에 놓였을 때 디스패처 및 작업 제어기와 같은 MTA 서비스가 제공되었습니까?
- 네트워크 연결성 또는 라우팅 문제로 인해 메시지가 원격 시스템에 고착되거나 잘못 라우팅되었습니까?
- 메시지가 메시지 대기열에 놓이기 전후에 문제가 발생했습니까?

위 질문은 이 장의 다음 절에 설명되어 있습니다.

26.2 표준 MTA 문제 해결 절차

이 절에서는 MTA에 대한 표준 문제 해결 절차에 대해 간략하게 설명합니다. 문제가 오류 메시지를 생성하지 않거나 오류 메시지에서 충분한 진단 정보를 제공하지 않을 경우 또는 MTA의 일반 상태 확인, 테스트 및 표준 유지 관리를 수행할 경우에는 다음 절차를 따릅니다.

26.2.1 MTA 구성 확인

`imsimta test -rewrite` 유틸리티를 사용하여 주소 구성을 테스트합니다. 이 유틸리티를 사용하여 실제로 메시지를 보내지 않고도 MTA의 주소 재작성 및 채널 매핑을 테스트할 수 있습니다. 자세한 내용은 **Sun Java System Messaging Server 6.3 Administration Reference**의 2 장, "Message Transfer Agent Command-line Utilities"를 참조하십시오.

일반적으로 유틸리티는 메시지가 대기할 채널뿐 아니라 적용될 주소 재작성을 표시합니다. 단, MTA 구성에 구문 오류가 있는 경우 유틸리티는 오류 메시지를 발생합니다. 출력이 예상과 다를 경우에는 구성을 수정해야 할 수 있습니다.

26.2.2 메시지 대기열 디렉토리 확인

일반적으로 *msg-svr-base*`/data/queue/`와 같은 MTA 메시지 대기열 디렉토리에 메시지가 있는지 확인합니다. `imsimta qm`과 같은 명령줄 유틸리티를 사용하여 MTA 메시지 대기열 디렉토리 아래에 예상한 메시지 파일이 있는지 확인합니다. `imsimta qm`에 대한 자세한 내용은 **Sun Java System Messaging Server 6.3 Administration Reference**의 "imsimta qm" 및 849 페이지 "27.8.6 imsimta qm 카운터"를 참조하십시오.

`imsimta test -rewrite`의 출력이 올바르게 표시되면 메시지가 실제로 MTA 메시지 대기열 하위 디렉토리에 놓이는지 확인합니다. 확인 시에는 메시지 로깅을 사용 가능하게 합니다(MTA 로깅에 대한 자세한 내용은 761 페이지 "25.3 MTA 메시지 및 연결 로그 관리" 참조). 그런 다음 / *msg-svr-base*`/log/` 디렉토리에서 mail.log_current 파일을 확인합니다. 메시지가 MTA 메시지 대기열 하위 디렉토리에 놓이는지 확인하기 위해 메시지 아이디를 사용하여 특정 메시지를 추적할 수 있습니다. 메시지를 찾을 수 없는 경우에는 파일 디스크 공간 또는 디렉토리 사용 권한에 문제가 있을 수 있습니다.

26.2.3 중요 파일의 소유권 확인

Messaging Server를 설치 시에는 메일 서버 사용자 계정(기본값은 mailsrv)을 선택해야 합니다. 다음 디렉토리, 하위 디렉토리 및 파일은 이 계정에서 소유해야 합니다.

msg-svr-base/data/queue/
msg-svr-base/data/log
msg-svr-base/data/tmp

예를 들어 다음 UNIX 시스템 명령은 이 디렉토리의 보호 및 소유권 확인에 사용될 수 있습니다.

```
ls -l -p -d /opt/SUNWmsgsr/data/queue
drwxr-x---   2 mailsrv  mail 512 Jan  4 16:09 /opt/SUNWmsgsr/data/queue/

ls -l -p -d /opt/SUNWmsgsr/data/log
drwxr-x---   2 mailsrv  mail  3072 Feb 16 12:07 /opt/SUNWmsgsr/data/log/

ls -l -p -d /opt/SUNWmsgsr/data/tmp
drwxr-x---   2 mailsrv  mail   512 Feb 16 12:55 /opt/SUNWmsgsr/data/tmp/
```

다음과 같은 UNIX 시스템의 명령을 사용하여 MTA 계정이 *msg-svr-base*/data/queue의 파일을 소유하는지 확인합니다.

```
ls -l -p -R /opt/SUNWmsgsr/data/queue
```

26.2.4 작업 제어기 및 디스패처 실행 확인

MTA 작업 제어기는 대부분의 보내는(마스터) 채널 작업을 포함하여 MTA 프로세스 작업의 실행을 처리합니다.

MTA의 다중 스레드 SMTP 채널과 같은 일부 MTA 채널은 받는 메시지를 처리하는 상주 서버 프로세스를 포함합니다. 이 서버는 채널에 대한 슬레이브(받는) 방향을 처리합니다. MTA 디스패처는 이러한 MTA 서버를 만듭니다. 디스패처 구성 옵션은 서버의 사용 가능성과 만들어진 서버의 수 및 각 서버가 처리할 수 있는 연결 수를 제어합니다.

작업 제어기 및 디스패처가 있는지 확인하고 MTA 서버 및 실행 중인 처리 작업이 있는지 보려면 imsimta process 명령을 사용합니다. 유휴 상태에서 명령은 job_controller 및 dispatcher 프로세스를 수행해야 합니다. 예를 들면 다음과 같습니다.

```
# imsimta process
USER      PID S VSZ    RSS   STIME    TIME     COMMAND
mailsrv 9567 S 18416 9368  02:00:02  0:00  /opt/SUNWmsgsr/lib/tcp_smtp_server
mailsrv 6573 S 18112 5720  Jul_13    0:00  /opt/SUNWmsgsr/lib/job_controller
mailsrv 9568 S 18416 9432  02:00:02  0:00  /opt/SUNWmsgsr/lib/tcp_smtp_server
mailsrv 6574 S 17848 5328  Jul_13    0:00  /opt/SUNWmsgsr/lib/dispatcher
```

작업 제어기가 없는 경우 */msg-svr-base*/data/queue 디렉토리의 파일은 백업되고 메시지가 전달되지 않습니다. 디스패처가 없으면 SMTP 연결을 수신할 수 없습니다.

imsimta process에 대한 자세한 내용은 **Sun Java System Messaging Server 6.3 Administration Reference**의 "imsimta process"를 참조하십시오.

또한 imsimta qm jobs를 사용하면 현재 작업 제어기에서 관리 중인 모든 활성 및 보류 중 배달 처리 작업을 채널별로 나열할 수 있습니다. 성공적으로 배달한 메시지 파일 수, 후속 배달 시도가 요청된 메시지 파일 수와 같은 추가 누적 정보가 각 채널에 대해 제공됩니다. 명령 구문은 다음과 같습니다.

```
jobs [-[no]hosts] [-[no]jobs] [-[no]messages] [channel-name]
```

작업 제어기 또는 디스패처가 모두 없는 경우에는 */msg-svr-base*/data/log에서 dispatcher.log-* 또는 job_controller.log-* 파일을 검토해야 합니다.

로그 파일이 존재하지 않거나 오류가 표시되지 않는 경우에는 start-msg 명령을 사용하여 프로세스를 시작합니다. 자세한 내용은 **Sun Java System Messaging Server 6.3 Administration Reference**의 "start-msg"를 참조하십시오.

주 – 시스템에서 실행해야 할 프로그램을 수행(exec())하기 전에 먼저 자식 프로세스를 포크(fork()) 처리하는 경우가 아닌 경우, imsimta process를 실행할 때 디스패처나 작업 제어기의 여러 인스턴스가 표시되지 않습니다. 그러나 이러한 중복의 시간 프레임은 매우 작습니다.

26.2.5 로그 파일 확인

MTA 처리 작업이 제대로 실행되지만 메시지가 메시지 대기열 디렉토리에 남아 있는 경우 로그 파일에 무슨 문제가 있는지 확인할 수 있습니다. 모든 MTA 로그 파일은 */msg-svr-base*/log 디렉토리에 만들어집니다. 다양한 MTA 처리 작업에 대한 로그 파일 이름 형식은 표 26–1에 표시됩니다.

표 26–1 MTA 로그 파일

파일 이름	로그 파일 내용
*channel*_master.log-*uniqueid*	*channel*에 대한 마스터 프로그램(주로 클라이언트)의 출력입니다.
*channel*_slave.log-*uniqueid*	*channel*에 대한 슬레이브 프로그램(주로 서버)의 출력입니다.
dispatcher.log-*uniqueid*	디스패처 디버깅입니다. 이 로그는 디스패처 DEBUG 옵션의 설정과 관계없이 만들어집니다. 하지만 자세한 디버깅 정보를 얻으려면 DEBUG 옵션을 0이 아닌 값으로 설정해야 합니다.
imta	ims-ms 채널 오류 메시지로 전달에 문제가 있는 경우 나타납니다.

표 26–1 MTA 로그 파일 *(계속)*

파일 이름	로그 파일 내용
job_controller.log-*uniqueid*	작업 제어기 로깅입니다. 이 로그는 작업 제어기 DEBUG 옵션의 설정과 관계없이 만들어집니다. 하지만 자세한 디버깅 정보를 얻으려면 DEBUG 옵션을 0이 아닌 값으로 설정해야 합니다.
tcp_smtp_server.log-*uniqueid*	tcp_smtp_server에 대한 디버깅입니다. 이 로그에 있는 정보는 메시지가 아닌 서버에만 해당됩니다.
return.log-*uniqueid*	주기적인 MTA 메시지 바운서 작업에 대한 출력을 디버그하며 option.dat에 return_debug 옵션이 사용되는 경우 이 로그 파일이 만들어집니다.

주 – 각 로그 파일은 고유 아이디(*uniqueid*)를 가지도록 만들어져 동일한 채널에서 만든 이전 로그를 덮어쓰지 못하게 합니다. 특정 로그 파일을 찾기 위해 imsimta view 유틸리티를 사용할 수 있습니다. 또한 imsimta purge 명령을 사용하여 오래된 로그 파일을 제거할 수 있습니다. 하지만 기본적으로 이 명령은 정기적으로 실행됩니다(110 페이지 “4.6.2 미리 정의된 자동 작업” 참조). 자세한 내용은 **Sun Java System Messaging Server 6.3 Administration Reference**의 “imsimta purge”를 참조하십시오.

*channel*_master.log-*uniqueid* 및 *channel*_slave.log- *uniqueid* 로그 파일은 다음 상황에서 만들어질 수 있습니다.

- 현재 구성에 오류가 있습니다.
- master_debug 또는 slave_debug 키워드가 imta.cnf 파일의 채널에 설정됩니다.
- mm_debug가 0이 아닌 값(mm_debug > 0)으로 option.dat 파일(*/msg-svr-base* /config/ 디렉토리)에 설정됩니다.

채널 마스터 및 슬레이브 프로그램 디버깅에 대한 자세한 내용은 **Sun Java System Messaging Server Administration Reference**를 참조하십시오.

26.2.6 수동으로 채널 프로그램 실행

MTA 전달 문제 진단 시에는 MTA 전달 작업을 수동으로 실행하는 것이 좋으며 특히 하나 이상의 채널에 대한 디버깅을 사용 가능하게 한 후에 유용합니다.

imsimta submit 명령은 MTA 작업 제어기에 채널을 실행할 것을 알립니다. If debugging is enabled for the channel in question, imsimta submit will create a log file in directory */msg-svr-base*/log as shown in 표 26–1.

imsimta run 명령은 현재 출력이 사용자의 단말기로 전송되는 활성 프로세스 상태의 채널에 대한 아웃바운드 전달을 수행합니다. 이는 작업 전송 자체에 문제가 있다고 의심되는 경우 작업을 전송하는 것보다 편리할 수 있습니다.

주 – 채널을 수동으로 실행하려면 작업 제어기가 실행 중이어야 합니다.

`imsimta submit` 및 `imsimta run` 명령에 대한 구문, 옵션, 매개 변수 및 그 예에 대한 정보는 **Sun Java System Messaging Server 6.3 Administration Reference**의 "Command Descriptions"를 참조하십시오.

26.2.7 개별 채널 시작 및 중지

경우에 따라 개별 채널을 시작 및 중지하면 메시지 대기열 문제의 진단 및 디버그가 더 쉬워질 수 있습니다. 메시지 대기열을 중지하면 대기 메시지를 검사하여 루프 또는 스팸 공격의 존재 여부를 확인할 수 있습니다.

▼ 특정 채널에 대한 아웃바운드 처리(대기열에서 제외) 중지 방법

1 **`imsimta qm stop` 명령을 사용하여 특정 채널을 중지합니다. 이렇게 하면 작업 제어기를 중지하지 않아도 되며 해당 구성을 다시 컴파일하지 않아도 됩니다. 다음 예에서는 `conversion` 채널이 중지됩니다.**

```
imsimta qm stop conversion
```

2 **처리를 계속하려면 `imsimta qm start` 명령을 사용하여 채널을 다시 시작합니다. 다음 예에서는 `conversion` 채널이 시작됩니다.**

```
imsimta qm start conversion
```

`imsimta qm start` 및 `imsimta qm stop` 명령에 대한 자세한 내용은 **Sun Java System Messaging Server 6.3 Administration Reference**의 "imsimta qm"을 참조하십시오.

26.2.7.1 특정 도메인 또는 IP 주소에서 인바운드 처리(채널 대기열에 포함) 중지

클라이언트 호스트에게 임시 SMTP 오류를 반환하면서 특정 도메인 또는 IP 주소에 대한 인바운드 메시지 처리를 중지하려면 다음 프로세스 중 하나를 실행할 수 있습니다. 이렇게 하면 메시지가 사용자 시스템에 보존되지 않습니다. 511 페이지 "18.1 1부. 매핑 테이블"를 참조하십시오.

- 특정 호스트 또는 도메인 이름에 대한 인바운드 처리를 중지하려면 MTA 매핑 파일(보통 */msg-svr-base*/`config/mappings`)의 ORIG_SEND_ACCESS 매핑 테이블에 다음 액세스 규칙을 추가합니다.

```
ORIG_SEND_ACCESS

  *|*@sesta.com|*|*        $X4.2.1|$NHost$ temporarily$ blocked
```

이 프로세스를 사용하면 보내는 사람의 원격 MTA가 시스템에 메시지를 보존하며 인바운드 처리를 다시 시작할 때까지 계속해서 주기적으로 메시지를 재전송합니다.

- 특정 IP 주소에 대한 인바운드 처리를 중지하려면 MTA 매핑 파일(보통 */msg-svr-base*/config/mappings)의 PORT_ACCESS 매핑 테이블에 다음 액세스 규칙을 추가합니다.

```
PORT_ACCESS

    TCP|*|25|IP_address_to_block|*    $N500$ can't$ connect$ now
```

도메인 또는 IP 주소에서 인바운드 처리를 다시 시작하려면 매핑 테이블에서 이 규칙을 제거하고 구성을 다시 컴파일해야 합니다. 추가로 각 매핑 테이블에 대한 고유 오류 메시지를 만들 수 있습니다. 이렇게 하면 어떤 매핑 테이블이 사용되는지 확인할 수 있습니다.

26.2.8 MTA 문제 해결 예

이 절에서는 특정 MTA 문제를 단계적으로 해결하는 방법에 대해 설명합니다. 이 예에서 메일을 받는 사람은 전자 메일의 첨부 파일을 받지 못했습니다. 주: MIME 프로토콜 용어와 동일하게 이 절에서는 "첨부 파일"을 "메시지 부분"이라고 합니다. 앞서 말한 문제 해결 기술은 메시지 부분이 사라진 위치 및 이유를 확인하기 위해 사용됩니다(802 페이지 "26.2 표준 MTA 문제 해결 절차" 참조). 다음 단계를 사용하면 MTA를 통해 메시지가 거쳐간 경로를 확인할 수 있습니다. 또한 메시지가 메시지 대기열에 들어가기 전이나 후에 사라졌는지 여부를 확인할 수 있습니다. 이렇게 하려면 채널을 수동으로 중지 및 실행하여 관련 파일을 캡처해야 합니다.

주 – 채널을 통해 메시지를 수동으로 실행하는 경우에는 작업 제어기가 실행 중이어야 합니다.

26.2.8.1 메시지 경로에서 채널 확인

메시지 경로에 어떤 채널이 있는지 확인하면 master_debug 및 slave_debug 키워드를 해당 채널에 적용할 수 있습니다. 이 키워드는 채널의 마스터 및 슬레이브 로그 파일에서 디버깅 출력을 생성하고 마스터 및 슬레이브 디버깅 정보는 메시지 부분이 사라진 지점을 확인하는 데 도움을 줍니다.

1. log_message_id=1 in the option.dat file in directory */msg-svr-base* /config. 디렉토리의 option.dat 파일에 추가합니다. 이 매개 변수와 함께 message ID: 헤더 행이 mail.log_current 파일에 표시됩니다.
2. imsimta cnbuild를 실행하여 구성을 다시 컴파일합니다.
3. imsimta restart dispatcher를 실행하여 SMTP 서버를 다시 시작합니다.
4. 최종 사용자가 메시지 부분이 있는 메시지를 재전송하도록 합니다.

5. 메시지가 통과하는 채널을 결정합니다.

   채널을 확인할 수 있는 방법은 많지만 다음 방법을 사용하는 것이 좋습니다.

   a. UNIX 플랫폼의 경우 `grep` 명령을 사용하여 `mail.log_current` msg-svr-base 디렉토리의 `mail.log_current` 파일에서 message ID: 헤더 행을 찾습니다.

   b. message ID: 헤더 행을 찾은 경우 대기열에 포함 및 대기열에서 제외 레코드를 찾아 메시지 경로를 확인합니다. 로깅 입력 코드에 대한 자세한 내용은 762 페이지 "25.3.1 MTA 로그 항목 형식 이해"를 참조하십시오. 그 예는 다음 E 및 D 레코드를 참조하십시오.

```
29-Aug-2001 10:39:46.44  tcp_local conversion        E 2 ...
29-Aug-2001 10:39:46.44  conversion tcp_intranet     E 2 ...
29-Aug-2001 10:39:46.44  tcp_intranet                  D 2 ...
```

왼쪽에 있는 채널이 소스 채널이고 오른쪽에 있는 채널이 대상 채널입니다. 이 예에서 E 및 D 레코드는 메시지 경로가 `tcp_local` 채널에서 `conversion` 채널로 이동한 다음 최종적으로 `tcp_intranet` 채널로 이동했음을 나타냅니다.

26.2.8.2 수동으로 채널을 시작 및 중지하여 데이터 수집

이 절에서는 채널을 수동으로 시작 및 중지하는 방법에 대해 설명합니다. 자세한 내용은 806 페이지 "26.2.7 개별 채널 시작 및 중지"를 참조하십시오. 메시지 경로에서 채널을 수동으로 시작 및 중지하면 MTA 프로세스의 각 단계에서 메시지 및 로그 파일을 저장할 수 있습니다. 이 파일은 나중에 810 페이지 "메시지 정지 지점 확인 방법"에 사용됩니다.

▼ 수동으로 채널을 시작 및 중지하는 방법

1 **실질적인 디버깅 정보를 제공하려면 `mm_debug=5`를 /*msg-svr-base*/`config` 디렉토리의 `option.dat` 파일에 설정합니다.**

2 **`slave_debug` 및 `master_debug` 키워드를 /*msg-svr-base*/`config` 디렉토리에 있는 `imta.cnf` 파일의 해당 채널에 추가합니다.**

   a. 메시지 부분이 있는 메시지를 보내는 원격 시스템에서 인바운드 채널(또는 초기 대화 중에 메시지가 전환되는 모든 채널)에 `slave_debug` 키워드를 사용합니다. 이 예에서 `slave_debug` 키워드는 `tcp_local` 채널에 추가됩니다.

   b. 메시지가 통과되고 807 페이지 "26.2.8.1 메시지 경로에서 채널 확인"에서 확인된 다른 채널에 `master_debug` 키워드를 추가합니다. 이 예에서 `master_debug` 키워드는 `conversion` 및 `tcp_intranet` 채널에 추가됩니다.

   c. `imsimta restart dispatcher` 명령을 실행하여 SMTP 서버를 다시 시작합니다.

3 imsimta qm stop 및 imsimta qm start 명령을 사용하여 특정 채널을 수동으로 시작 및 중지합니다. 이 키워드 사용에 대한 자세한 내용은 806 페이지 "26.2.7 개별 채널 시작 및 중지"를 참조하십시오.

4 메시지 파일을 캡처하는 프로세스를 시작하려면 최종 사용자가 메시지 부분이 있는 메시지를 재전송하도록 합니다.

5 메시지가 채널에 입력될 때 imsimta qm stop 명령에 의해 중지된 경우에는 채널에서 메시지가 중지됩니다. 자세한 내용은 단계 3을 참조하십시오.

   a. 메시지 경로에서 다음 채널을 수동으로 실행하기 전에 메시지 파일을 복사하고 이름을 바꿉니다. 다음 UNIX 플랫폼 예를 참조하십시오.

      ```
      # cp ZZ01K7LXW76T7O9TD0TB.00 ZZ01K7LXW76T7O9TD0TB.KEEP1
      ```

      일반적으로 메시지 파일은 */msg-svr-base*/data/queue/*destination_channel*/001과 유사한 디렉토리에 상주합니다. *destination_channel*은 메시지가 다음으로 통과(tcp_intranet 등)하는 채널입니다. 하위 디렉토리(001, 002 등)를 *destination_channel* 디렉토리에 만들려면 채널에 subdirs 키워드를 추가합니다.

   b. 메시지가 처리된 순서를 확인하려면 메시지를 트랩 및 복사할 때마다 메시지 확장자에 번호를 지정하는 것이 좋습니다.

6 채널에서 메시지 처리를 계속하고 메시지 경로에서 다음 대상 채널로 대기열에 포함합니다. 이 작업을 수행하려면 imsimta qm start 명령을 사용합니다.

7 */msg_svr_base*/log 디렉토리에 있는 해당 채널 로그 파일(예: tcp_intranet_master.log-*)을 복사하여 저장합니다. 추적하는 메시지에 대한 데이터를 가진 해당 로그 파일을 선택합니다. 복사한 파일을 채널에 넣을 때 해당 메시지의 타임스탬프 및 제목 헤더와 일치하도록 합니다. tcp_intranet_master.log-*의 예에서는 파일이 삭제되지 않도록 파일을 tcp_intranet_master.keep로 저장할 수 있습니다.

8 메시지가 해당 최종 대상에 도달할 때까지 단계 5-7을 반복합니다.

   단계 7에서 복사한 로그 파일은 단계 5에서 복사한 메시지 파일과 상관 관계가 있어야 합니다. 예를 들어, 누락된 메시지 부분 시나리오에서 모든 채널을 중지한 경우 conversion_master.log-* 및 tcp_intranet_master.log-* 파일을 저장합니다. 또한 소스 채널 로그 파일인 tcp_local_slave.log-*도 저장합니다. 추가로 다음과 같이 각 대상 채널로부터 해당 메시지 파일의 복사본을 저장합니다. conversion 채널에서 ZZ01K7LXW76T7O9TD0TB.KEEP1 및 tcp_intranet 채널에서 ZZ01K7LXW76T7O9TD0TB.KEEP2 파일을 저장합니다.

9 메시지 및 로그 파일을 복사한 후 디버깅 옵션을 제거합니다.

   a. */msg-svr-base*/config 디렉토리에 있는 imta.cnf 파일의 해당 채널에서 slave_debug 및 master_debug 키워드를 제거합니다.

b. /*msg-svr-base*/config **디렉토리의** option.dat **파일에서** mm_debug=0**을 재설정하고** log_message_id=1**을 제거합니다.**

c. imsimta cnbuild**를 사용하여 구성을 다시 컴파일합니다.**

d. imsimta restart dispatcher **명령을 실행하여 SMTP 서버를 다시 시작합니다.**

▼ 메시지 정지 지점 확인 방법

1 채널 프로그램 시작 및 중지가 완료되면 문제 해결에 사용할 수 있는 다음과 같은 파일을 가지게 됩니다.

a. 각 채널 프로그램에서 메시지 파일의 모든 복사본(예: ZZ01K7LXW76T7O9TD0TB.KEEP1)

b. tcp_local_slave.log-* **파일**

c. 각 대상 채널에 대한 *channel*_master.log-* **파일 집합**

d. 메시지 경로를 표시하는 mail.log_current **레코드 집합**

모든 파일은mail.log_current 레코드에서 message ID: 헤더 행과 일치하는 타임스탬프 및 메시지 아이디 값을 가져야 합니다. 메시지가 보낸 사람에게 다시 반송될 경우는 예외이며 반송된 메시지는 원본 메시지와는 다른 아이디 값을 가지게 됩니다.

2 tcp_local_slave.log-* **파일을 검사하여 메시지가 메시지 대기열에 들어갔을 때 메시지 부분을 가지고 있었는지 확인합니다.**

SMTP 대화 상자 및 데이터를 확인하여 클라이언트 시스템에서 무엇을 보냈는지 봅니다.

메시지 부분이 tcp_local_slave.log-* 파일에 표시되지 않았다면 메시지가 MTA에 놓이기 전에 문제가 발생한 것입니다. 그 결과 메시지가 메시지 부분 없이 대기열에 포함되었습니다. 이 경우 보낸 사람의 원격 SMTP 서버 또는 보낸 사람의 클라이언트 시스템에서 문제가 발생했을 수 있습니다.

3 메시지 파일의 복사본을 조사하여 메시지 부분이 어디서 변경 또는 누락되었는지 확인합니다.

메시지 파일에서 메시지 부분이 변경 또는 누락되었음이 표시되면 이전 채널의 로그 파일을 검사합니다. 예를 들어, tcp_intranet 채널에 놓이는 메시지의 메시지 부분이 변경 또는 누락된 경우 conversion_master.log-* 파일을 확인합니다.

4 **메시지의 최종 대상을 확인합니다.**

`tcp_local_slave.log` 메시지 파일(예: `ZZ01K7LXW76T7O9TD0TB.KEEP1`) 및 *channel*`_master.log-*` 파일에서 메시지 부분이 변경되지 않은 것으로 확인되면 MTA는 메시지를 변경하지 않았으며 메시지 부분은 해당 최종 대상으로 가는 경로의 다음 단계에서 사라진 것입니다.

최종 대상이 `ims-ms` 채널(메시지 저장소)인 경우, 메시지 부분이 전송 과정 도중이나 이후에 누락되는지를 확인하기 위해 메시지를 서버에서 클라이언트 시스템으로 다운로드할 수 있습니다. 대상 채널이 `tcp_*` 채널인 경우에는 메시지 경로의 MTA로 이동해야 합니다. 이 MTA를 Messaging Server MTA라고 가정하면 전체 문제 해결 프로세스를 반복해야 합니다(807 페이지 "26.2.8.1 메시지 경로에서 채널 확인", 808 페이지 "26.2.8.2 수동으로 채널을 시작 및 중지하여 데이터 수집" 및 이 절 참조). 사용자가 관리하는 MTA가 아닌 경우 문제를 보고한 사용자가 해당 사이트에 문의해야 합니다.

26.3 일반 MTA 문제 및 솔루션

이 절에서는 MTA 구성 및 작업에 대한 일반 문제 및 솔루션을 나열합니다.

26.3.1 TLS 문제

SMTP 대화 중에 `STARTTLS` 명령은 다음 오류를 반환합니다.

```
454 4.7.1 TLS library initialization failure
```

pop/imap에 액세스에 대한 인증서를 설치 및 작동 중에 있다면 다음 사항을 확인하십시오.

- `mailsrv` 계정이 파일을 액세스할 수 있도록 인증서의 보호/소유권을 설정해야 합니다.

- mailsrv 계정에서 인증서가 저장되는 디렉토리 내의 파일에 액세스할 수 있도록 해당 디렉토리에서 보호/소유권을 설정해야 합니다.

보호를 변경하고 인증서를 설치한 후에는 다음을 실행해야 합니다.

```
stop-msg dispatcher
start-msg dispatcher
```

다시 시작해도 좋지만 시스템을 완전히 종료하고 인증서를 설치한 다음 다시 시작하는 것이 좋습니다.

26.3.2 영향력이 없는 구성 파일 또는 MTA 데이터베이스에 대한 변경 사항

구성, 매핑, 전환, 보안, 옵션 또는 별칭 파일에 대한 변경이 적용되지 않는 경우 다음 단계를 수행했는지 확인합니다.

1. 구성을 다시 컴파일합니다(imsimta cnbuild 실행).
2. 적절한 프로세스(예: imsimta restart dispatcher)를 다시 시작합니다.
3. 모든 클라이언트 연결을 다시 설정합니다.

26.3.3 MTA에서 보내는 메일은 전송하지만 받는 메일을 수신하지 않음

대부분의 MTA 채널은 받는 메시지를 수신하는 슬레이브 또는 채널 프로그램에 의존합니다. MTA가 지원하는 일부 전송 프로토콜(TCP/IP 및 UUCP 등)의 경우 해당 표준 서버 대신 MTA 슬레이브 프로그램을 활성화해야 합니다. Messaging Server 설치의 일환으로 원시 sendmail SMTP 서버가 MTA SMTP 서버로 대체됩니다.

다중 스레드 SMTP 서버의 경우 디스패처에서 SMTP 서버의 시작을 제어합니다. 디스패처가 SMTP 서비스에 대해 1 이상의 MIN_PROCS 값을 사용하도록 구성된 경우, 적어도 하나 이상(가능한 경우 SMTP 서비스에 대한 MAX_PROCS 값에 따라 그 이상으로)의 SMTP 서버 프로세스가 항상 실행 중이어야 합니다. imsimta process 명령을 사용하여 SMTP 서버 프로세스의 존재 여부를 확인할 수 있습니다. 자세한 내용은 **Sun Java System Messaging Server 6.3 Administration Reference**의 "imsimta process"를 참조하십시오.

26.3.4 디스패처(SMTP Server)가 시작하지 않음

디스패처가 시작되지 않는 경우에는 먼저 dispatcher.log-*에서 관련 오류 메시지를 확인합니다. 로그에서 /tmp/.SUNWmsgsr.dispatcher.socket 파일을 만들거나 액세스하는 문제를 표시하는 경우, /tmp 보호가 1777로 설정되어 있는지 확인합니다. 사용 권한에 다음과 같이 표시됩니다.

```
drwxrwxrwt 8 root sys 734 Sep 17 12:14 tmp/
.
```

또한 SUNWmsgsr.dispatcher.socket 파일의 ls -l을 수행하고 적합한 소유권을 확인하십시오. 예를 들어, root에서 만든 경우에는 inetmail에서 액세스할 수 없습니다.

SUNWmsgsr.dispatcher.file을 제거하지 말고 누락된 경우에는 만들지 마십시오. 디스패처에서 해당 파일을 만듭니다. 보호가 1777로 설정되지 않은 경우에는 디스패처가 소켓 파일을 만들고 액세스할 수 없기 때문에 시작 또는 다시 시작하지 않습니다. 또한 Messaging Server와 연관되지 않은 다른 문제들이 발생할 수 있습니다.

26.3.5 받는 SMTP 연결의 시간 초과

받는 SMTP 연결의 시간 초과는 흔히 시스템 자원 및 해당 할당과 연관되어 있습니다. 다음 기술을 사용하여 받는 SMTP 연결의 시간 초과 원인을 확인할 수 있습니다.

▼ 받는 SMTP 연결의 시간 초과 원인을 확인하는 방법

1 **허용된 동시에 받는 SMTP 연결 수를 확인합니다. 이는 SMTP 서비스에 대한 MAX_PROCS 및 MAX_CONNS 디스패처 설정이 제어하며 허용되는 동시 연결 수는 MAX_PROCS*MAX_CONNS입니다. 시스템 자원이 충분하고 이 수가 사용량에 비해 너무 적은 경우 수를 늘릴 수 있습니다.**

2 **다른 기술로는 TELNET 세션을 여는 것이 있습니다.**

다음 예에서는 사용자가 127.0.0.1 포트 25에 연결합니다. 연결되면 220 배너가 반환됩니다. 예를 들면 다음과 같습니다.

```
telnet 127.0.0.1 25
Trying 127.0.0.1...
Connected to 127.0.0.1.
Escape character is '^]'.
220 budgie.sesta.com --Server ESMTP (Sun Java System Messaging Server 6.1
(built May  7 2001))
```

사용자가 해당 포트에 연결되고 220 배너를 수신하지만 추가 명령(예: ehlo 및 mail from)이 응답을 부정하지 않는 경우, imsimta test -rewrite를 실행하여 구성이 올바른지 확인합니다.

3 **220 배너의 응답 시간이 느리고 SMTP 서버에 pstack 명령이 실행 중인 경우 다음 iii_res* 함수가 표시됩니다(이 함수는 이름 확인 조회가 수행 중임을 나타냄).**

```
febe2c04 iii_res_send (fb7f4564, 28, fb7f4de0, 400, fb7f458c, fb7f4564) +
42c febdfdcc iii_res_query (0, fb7f4564, c, fb7f4de0, 400, 7f) + 254
```

그런 다음 호스트는 localhost/127.0.0.1과 같은 일반 쌍에서도 역방향 이름 확인 조회를 실행해야 할 수 있습니다. 이와 같은 성능 저하를 방지하려면 /etc/nsswitch.conf 파일에서 사용자 호스트 조회 순서를 다시 정렬해야 합니다. 그러기 위해서는 /etc/nsswitch.conf 파일에서 다음 행을

```
hosts: dns nis [NOTFOUND=return] files
```

아래와 같이 변경합니다.

```
hosts: files dns nis [NOTFOUND=return]
```

/etc/nsswitch.conf 파일에서 이러한 변경 작업을 수행하면 여러 SMTP 서버가 불필요한 조회를 수행하는 대신 더 적은 수의 SMTP 서버에서 메시지를 처리하므로 성능이 향상됩니다.

4 **또한 slave_debug 키워드를 주로 tcp_local 및 tcp_intranet과 같은 TCP/IP 메일을 통해 받는 SMTP를 처리하는 채널에 입력할 수 있습니다. 그런 다음 최근의 tcp_local_slave.log-*uniqueid* 파일을 검토하여 시간 초과된 메시지의 특성을 확인합니다. 예를 들어, 수신자가 많은 받는 메시지가 시간 초과되는 경우 채널에 expandlimit 키워드를 사용하는 것이 좋습니다.**

시스템이 오버로드되고 지나치게 확장된 경우 시간 초과를 완전히 방지할 수는 없습니다.

26.3.6 메시지가 대기열에서 제외되지 않음

TCP/IP 전달 중에 발생하는 오류는 보통 일시적이며 문제가 발생하면 일반적으로 MTA에서 메시지를 보관하고 주기적으로 다시 보내려는 시도를 합니다. 다른 호스트 연결이 정상적으로 작동할 때 대규모 네트워크가 특정 호스트에서 주기적으로 중단되는 것은 정상입니다. 문제를 확인하려면 로그 파일에서 전달 시도와 관련된 오류를 검사합니다. "smtp_open의 치명적 오류"와 같은 오류 메시지가 표시될 수 있습니다. 이런 오류는 보편적이며 보통 일시적인 네트워크 문제와 연관되어 있습니다. TCP/IP 네트워크 문제를 디버그하려면 PING, TRACEROUTE 및 NSLOOKUP과 같은 유틸리티를 사용합니다.

다음 예는 메시지가 xtel.co.uk에 전달 시 대기열에서 대기하는 이유를 확인하기 위해 사용자가 사용할 수 있는 단계를 보여 줍니다. 메시지가 대기열에서 제외되지 않는 이유를 확인하려면 MTA가 TCP/IP에서 SMTP 메일 전달에 사용하는 단계를 다시 만들 수 있습니다.

```
% nslookup -query=mx xtel.co.uk       (Step 1)

Server: LOCALHOST
Address: 127.0.0.1

Non-authoritative answer:
```

```
XTEL.CO.UK  preference = 10, mail exchanger = nsfnet-relay.ac.uk      (Step 2)

% telnet nsfnet-relay.ac.uk 25      (Step 3)
Trying... [128.86.8.6]
telnet: Unable to connect to remote host: Connection refused
```

1. NSLOOKUP 유틸리티를 사용하여 이 호스트에 어떤 MX 레코드가 있는지(있는 경우) 확인합니다. MX 레코드가 없으면 호스트에 직접 연결해야 합니다. MX 레코드가 있으면 지정된 MX 중계에 연결해야 합니다. 명시적으로 연결하지 않도록 구성된 경우를 제외하면 MTA는 우선적으로 MX 정보를 적용합니다. 346 페이지 “12.4.3.5 TCP/IP MX 레코드 지원”을 참조하십시오.
2. 이 예에서 DNS(도메인 이름 서비스)는 `xtel.co.uk`에 대해 지정된 MX 중계의 이름을 반환합니다. 이 호스트는 MTA가 실제로 연결할 호스트입니다. 둘 이상의 MX 중계가 나열된 경우 MTA는 각 MX 레코드를 기본값이 낮은 것부터 오름차순으로 연속해서 시도합니다.
3. 원격 호스트에 연결된 경우 SMTP 서버 포트 25의 TELNET을 사용하여 인바운드 SMTP 연결을 허용하는지 확인합니다.

주 – 포트를 지정하지 않고 TELNET을 사용하는 경우에는 원격 호스트가 일반 TELNET 연결을 허용합니다. 하지만 위의 경우 원격 호스트가 SMTP 연결을 허용한다는 것을 의미하지는 않습니다. 많은 시스템이 일반 TELNET 연결은 허용하지만 SMTP 연결을 거부하며 또한 그 반대로 SMTP 연결을 허용하지만 TELNET 연결은 거부합니다. 따라서 항상 SMTP 포트에 대해 테스트를 수행해야 합니다.

이전 예에서 원격 호스트는 SMTP 포트에 대한 연결을 거부합니다. 이는 MTA가 메시지 전달에 실패하는 이유입니다. 원격 호스트의 잘못된 구성 또는 원격 호스트에서의 일부 자원 고갈로 인해 연결이 거부될 수 있습니다. 이런 경우 문제를 로컬에서 해결할 수 없습니다. 일반적으로 MTA가 메시지를 계속해서 시도하도록 합니다.

DNS를 사용하지 않는 TCP/IP 네트워크에 Messaging Server가 실행 중인 경우에는 처음 두 단계를 건너뛸 수 있습니다. 대신 TELNET을 사용하여 해당 호스트에 직접 액세스할 수 있습니다. 호스트 이름은 MTA에서 사용하는 이름과 같아야 합니다. 호스트 이름을 확인하려면 MTA의 최근 시도에서 관련 로그 파일을 봅니다. 호스트 파일을 사용하는 경우에는 호스트 이름 정보가 올바른지 확인합니다. 호스트 이름 대신 DNS를 사용해야 합니다.

TCP/IP 호스트에 대한 연결을 테스트할 때 대화식 테스트 사용에 문제가 발생하지 않는 경우 문제는 MTA가 마지막 메시지 전달을 시도할 때 간단히 해결되었을 수 있습니다. 메시지가 대기열에서 제외되는지 확인하려면 적절한 채널에서 `imsimta submit tcp_channel`을 다시 실행합니다.

26.3.6.1 새 채널 만들기

원격 도메인이 정지되고 이 서버로 주소가 지정된 메일 양이 너무 커서 전달할 수 없는 메시지가 보내는 채널 대기열을 채우는 경우가 있습니다. MTA는 이러한 메시지를 주기적으로 다시 전달하려고 시도하며(재시도 빈도와 횟수는 backoff 키워드를 사용하여 구성 가능) 정상적인 상황에서는 어떤 작업도 수행할 필요가 없습니다. 그러나 대기열에 있는 메시지 수가 너무 많으면 모든 채널 작업이 전달할 수 없는 메시지의 백로그를 처리하기 때문에 다른 메시지가 적시에 전달되지 못할 수 있습니다.

이 경우 이러한 메시지를 자체 작업 제어기 풀에서 실행되는 새 채널로 다시 라우팅할 수 있습니다. 이렇게 하면 처리를 위한 경합이 발생하지 않으며 다른 채널이 해당 메시지를 전달할 수 있습니다. 이 절차에 대해서는 아래에서 설명합니다. siroe.com이라는 도메인을 사용한다고 가정합니다.

▼ 새 채널 만들기

1 tcp_siroe-daemon이라는 새 채널을 만들고 pool 키워드에 대한 새 값을 추가합니다.

/msg-svr-base/config/imta.cnf의 채널 블록 섹션에 채널이 작성됩니다. 이 채널은 정기적인 보내는 tcp_* 채널에서 동일한 채널 키워드를 가져야 합니다. 일반적으로 이 채널은 모든 아웃바운드(인터넷) 트래픽을 처리하는 tcp_local입니다. siroe.com이 인터넷상에 있으므로 이 채널이 에뮬레이트됩니다. 새 채널은 다음과 같을 수 있습니다.

```
tcp_siroe smtp nomx single_sys remotehost inner allowswitchchannel     \
dentnonenumeric subdirs 20 maxjobs 7 pool SMTP_SIROE maytlsserver      \
maysaslserver saslswitchchannel tcp_auth missingrecipientpolicy 0      \
tcp_siroe-daemon
```

새 키워드-값 쌍의 풀인 SMTP_SIROE를 확인합니다. 이것은 이 채널로 보내는 메시지가 SMTP_SIROE 풀의 컴퓨터 자원만 사용하도록 지정합니다. 또한 새 채널의 앞뒤에 빈 행이 필요하다는 것에 주의하십시오.

2 imta.cnf 파일의 다시 쓰기 규칙 섹션에 두 개의 다시 쓰기 규칙을 추가하여 siroe.com을 대상으로 하는 전자 메일을 새 채널로 보냅니다.

새 다시 쓰기 규칙은 다음과 같습니다.

```
siroe.com      $U%$D@tcp_siroe-daemon
.siroe.com       $U%$H$D@tcp_siroe-daemon
```

이러한 다시 쓰기 규칙은 siroe.com을 대상으로 하는 메시지(host1.siroe.com 또는 hostA.host1.siroe.com과 같은 주소 포함)를 해당 공식 호스트 이름이 tcp_siroe-daemon인 새 채널로 보냅니다. $U%$D 및 $U%$H$D 규칙의 다시 쓰기 부분은 메시지의 원래 주소를 유지합니다. $U는 원래 주소에서 아이디를 복사합니다. %는 아이디와 도메인 사이에 있는 구분자 @입니다. $H는 패턴에서 점 왼쪽에 있는 호스트/도메인 지정의 일치하지 않는 부분을 복사합니다. $D는 일치한 도메인 지정 부분을 복사합니다.

3 SMTP_SIROE라는 새 작업 제어기 풀을 정의합니다.

/msg-svr-base/config/job_controller.cnf에 다음을 추가합니다.

```
[POOL=SMTP_SIROE]
job_limit=10
```

이렇게 하면 최대 10개의 작업을 동시에 실행할 수 있는 SMTP_SIROE라는 메시지 자원 풀이 작성됩니다. 이 풀 정의와 다른 항목 사이에 빈 행이 있으면 제거합니다. 작업 및 풀에 대한 자세한 내용은 179 페이지 “8.7 작업 제어기”를 참조하십시오.

4 MTA를 다시 시작합니다.

다음 명령을 실행합니다. `imsimta cnbuild;imsimta restart`

이 명령은 구성을 다시 컴파일하고 작업 제어기와 디스패처를 다시 시작합니다.

이 예에서는 인터넷 사용자가 `siroe.com`이라는 특정 원격 사이트로 다수의 전자 메일을 보냅니다. 어떤 이유로 `siroe.com`이 일시적으로 받는 SMTP 연결을 허용할 수 없어 전자 메일을 전달할 수 없게 됩니다. (이러한 상황이 자주 발생합니다.)

`siroe.com`을 대상으로 하는 전자 메일이 들어오면 보내는 채널 대기열인 `tcp_local`에 전달할 수 없는 메시지가 채워집니다. MTA는 이러한 메시지를 주기적으로 다시 전달하려고 시도하며(재시도 빈도와 횟수는 `backoff` 키워드를 사용하여 구성 가능) 정상적인 상황에서는 어떤 작업도 수행할 필요가 없습니다.

그러나 대기열에 있는 메시지 수가 너무 많으면 모든 채널 작업이 `siroe.com` 메시지의 백로그를 처리하기 때문에 다른 메시지가 적시에 전달되지 못할 수 있습니다. 이 경우 `siroe.com` 메시지를 자체 작업 제어기 풀에서 실행되는 새 채널로 다시 라우팅할 수 있습니다(179 페이지 “8.7 작업 제어기” 참조). 이렇게 하면 `siroe.com` 메시지가 사용하는 처리 자원을 경합하지 않아도 다른 채널이 해당 메시지를 전달할 수 있습니다. 새 채널을 만들어 이 문제를 해결하는 방법은 아래에서 설명합니다.

26.3.7 MTA 메시지가 전달되지 않음

메시지 전송 문제 외에도 메시지 대기열에서 메시지를 처리할 수 없도록 하는 두 가지 일반적인 문제가 있습니다.

1. 대기열 캐시가 대기열 디렉토리의 메시지와 동기화되지 않습니다. 전달 대기 중인 MTA 대기열 하위 디렉토리의 메시지 파일은 메모리 내장 대기열 캐시에 놓입니다. 채널 프로그램이 실행되는 경우에는 이 대기열 캐시에 문의하여 해당 대기열에서 어떤 메시지를 전달할지 결정합니다. 대기열에 메시지 파일이 있지만 해당 대기열 캐시 항목이 없는 경우도 있습니다.
   a. 대기열 캐시에 특정 파일이 있는지 확인하려면 `imsimta cache -view` 유틸리티를 사용하고 파일이 대기열 캐시에 없는 경우 대기열 캐시를 동기화해야 합니다.

대기열 캐시는 보통 4시간마다 동기화됩니다. 필요하면 `imsimta cache -sync` 명령을 사용하여 캐시를 수동으로 다시 동기화할 수 있습니다. 이 채널 프로그램은 일단 동기화되면 새 메시지가 처리된 후에도 처리되지 않는 원본 메시지를 처리합니다. 기본값(4시간)을 변경하려면 대기열 캐시의 동기화 빈도를 반영하는 *timeperiod*가 있는 `sync_time=`*timeperiod*를 추가하여 *msg-svr-base*`/config` 디렉토리의 `job_controller.cnf` 파일을 수정합니다. *timeperiod*는 30분보다 커야 합니다. 다음 예에서는 `sync_time=02:00`을 `job_controller.cnf`의 전역 기본 섹션에 추가하여 동기화를 2시간으로 수정합니다.

```
! VERSION=5.0
!IMTA job controller configuration file
!
!Global defaults
tcp_port=27442
secret=N1Y9[HzQKW
slave_command=NULL
sync_time=02:00
```

`imsimta cache -sync`를 실행한 후 `imsimta submit` *channel*을 실행하여 메시지의 백로그를 모두 지웁니다. 메시지의 백로그가 큰(1000 이상) 경우 채널을 모두 지우는 데 긴 시간이 필요할 수도 있다는 점에 유의해야 합니다.

요약된 대기열 캐시 정보에 대해서는 `imsimta qm -maint dir -database -total`을 실행합니다.

b. 대기열 캐시를 동기화한 후에도 메시지가 전달되지 않으면 작업 제어기를 다시 시작해야 합니다. 그러려면 `imsimta restart job_controller` 명령을 사용합니다.

작업 제어기를 다시 시작하면 디스크에서 메시지 대기열의 메시지 데이터 구조가 재구성됩니다.

주의 – 작업 제어기를 다시 시작하는 것은 최후의 수단으로 다른 방법을 모두 사용해 본 후에 수행되어야 합니다.

작업 제어기에 대한 자세한 내용은 179 페이지 “8.7 작업 제어기”를 참조하십시오.

2. 해당 처리 로그 파일을 만들 수 없기 때문에 채널 처리 프로그램을 실행할 수 없습니다. 액세스 권한, 디스크 공간 및 할당량을 확인하십시오.

26.3.8 메시지 루핑

MTA가 메시지 루핑을 감지하면 해당 메시지는 .HELD 파일로 취급되어 보류됩니다. 819 페이지 “26.3.8.1 .HELD 메시지 진단 및 정리”를 참조하십시오. 특정 경우에는 MTA에서 감지할 수 없는 메시지 루프가 발생할 수 있습니다.

첫 번째 단계로 메시지 루핑의 원인을 확인합니다. 해당 채널에 대해 문제 메시지 파일이 MTA 대기열 영역, 문제 메시지와 연관된 MTA 메일 로그 항목(해당 채널에 대한 MTA 구성 파일에서 `logging` 채널 키워드를 활성화한 경우) 및 MTA 채널 디버그 로그 파일에 있는 동안 문제 메시지 파일의 복사본을 검토해야 합니다. 문제 메일에 대한 From: 및 To: 주소와 받은 날짜헤더 행 및 메시지 구조(메시지 내용의 캡슐화 유형)를 확인하는 것은 발생할 수 있는 메시지 루프 유형을 정확히 아는데 도움이 됩니다.

보다 일반적인 경우는 다음과 같습니다.

1. 포스트마스터 주소가 손상되었습니다.

   MTA는 포스트마스터 주소로 전자 메일을 받도록 합니다. 포스트마스터로 보낸 메시지가 루핑되는 경우에는 메시지를 받을 수 있는 계정을 가르키는 적절한 포스트마스터 주소가 구성되어 있는지 확인합니다.

2. Received: 헤더 행을 제거하면 MTA에서 메시지 루프를 감지할 수 없습니다.

   정상적인 메시지 루프 감지는 Received: 주석(괄호 안의 내용)을 제거합니다. Received: 헤더 행이 제거되면 (해당 시스템에서 명시적으로 또는 방화벽과 같은 다른 시스템에서) 메일 루프를 적절히 감지하는 것을 방해할 수 있습니다. 이 시나리오에서는 원하지 않는 헤더 행이 제거되지 않도록 합니다. 또한, 근본적인 메시지 루핑 원인을 조사합니다. 가능한 원인으로는 시스템 이름 할당 문제, 해당 이름의 변형을 인식하지 못하게 구성된 시스템 문제, DNS 문제, 해당 시스템의 인증 주소 지정 정보 없음 또는 사용자 주소 전달 오류등이 있습니다.

3. 다른 메시징 시스템이 알림 메시지를 잘못 처리하면 알림 메시지에 대한 응답으로 다시 캡슐화된 메시지가 생성됩니다.

   인터넷 표준은 메시지 루프를 방지하기 위해 알림 메시지(전달되는 메시지 또는 반송되는 메시지에 대한 보고서)에 From: 주소가 공백인 봉투를 요구합니다. 하지만 일부 메시징 시스템은 이러한 알림 메시지를 제대로 처리하지 않습니다. 알림 메시지를 전달 또는 튕기는 경우 이 메시징 시스템은 새 From: 주소에서 추출된 SMS 대상 주소의 숫자가 아닌 모든 문자를 스트라이프하려면 이 옵션을 지시합니다. 이 봉투를 삽입하면 메시지 루프가 발생할 수 있습니다. 해결책은 알림 메시지를 제대로 처리하지 못하는 메시징 시스템을 수정하는 것입니다.

26.3.8.1 .HELD 메시지 진단 및 정리

MTA에서 메시지 배달과 관련하여 심각한 문제를 발견한 경우, 그 메시지는 */msg-svr-base*`/data/queue/`*channel*에서 `.HELD` 접미어가 붙은 파일에 저장됩니다. 예를 들면 다음과 같습니다.

```
% ls
ZZ0HXZ00G0EBRBCP.HELD
ZZ0HY200C0O6LGHU.HELD
ZZ0HYA006LP66O3H.HELD
ZZ0HZ7003EOQSE37.HELD
```

`.HELD` 파일이 생성되는 대표적인 이유 세 가지는 다음과 같습니다.

- 메시지가 루핑합니다. 일종의 `Received:` 헤더 라인이 만들어지면서 메시지가 루핑하고 있음을 MTA가 감지합니다.
- 사용자 또는 도메인 상태가 `hold`로 설정됩니다. 이 메시지는 일부 유지 관리 절차가 수행되는 동안(예: 사용자 메일함 이동 중) MTA 관리자가 일부러 분리해 놓은 메시지입니다.
- 의심스러운 메시지입니다. 의심 임계값에 도달했으며, MTA 관리자가 나중에 수동으로 검사하도록 보류해 놓은 메시지입니다. 구성된 최대 봉투 수신자 수(359 페이지 "12.5.9 여러 주소 확장"의 `holdlimit` 채널 키워드 참조)를 초과했거나, 해당 메시지가 의심스러워 **Sun Java System Messaging Server 6.3 Administration Reference**의 "imsimta qclean", **Sun Java System Messaging Server 6.3 Administration Reference**의 "clean" 또는 **Sun Java System Messaging Server 6.3 Administration Reference**의 "hold" 명령을 실행했거나, 시브(Sieve) 스크립트에서 `hold` 작업을 사용한 경우, 메시지가 `.HELD`가 될 수 있습니다.

루핑에 의한 Messages .HELD

서버 또는 채널 사이에 바운스되는 메시지를 루핑한다고 합니다. 일반적으로 각 서버 또는 채널은 메시지 전달에 대한 책임이 다른 서버 또는 채널에 있다고 생각하기 때문에 메시지 루프가 발생합니다. 루핑 메시지는 보통 많은 수의 `*Received:` 헤더 행을 포함하고 있습니다. `Received:` 헤더 행은 메시지 루프의 정확한 경로를 나타냅니다. 그러한 헤더 행에 나타나는 호스트 이름과 수신자 주소 정보(예: `for recipient` 절이나 `(ORCPT recipient)` 주석)를 자세히 확인합니다. 메시지 루프가 생기는 원인 중 하나는 사용자의 실수입니다.

예를 들어, 최종 사용자는 서로 다른 두 개의 메일 호스트에서 서로에게 메시지를 전달하도록 옵션을 설정할 수 있습니다. 최종 사용자는 `sesta.com` 계정에서 `varrius.com` 계정으로 메일이 전달되도록 합니다. 그 후 최종 사용자가 이 설정을 사용 가능하게 한 사실을 잊고 `varrius.com` 계정에서 `sesta.com` 계정으로 메일이 전달되도록 설정합니다.

또한 MTA 구성 결함으로 인해 루프가 발생할 수 있습니다. 예를 들어, MTA 호스트 X는 `mail.sesta.com`에 대한 메시지가 호스트 Y로 간다고 생각합니다. 하지만 호스트 Y는 `mail.sesta.com`에 대한 메시지를 호스트 X가 처리해야 한다고 생각하기 때문에 메시지를 호스트 X에게 반환합니다.

이런 경우 MTA는 메시지를 무시하고 더 이상의 전달을 시도하지 않습니다. 이러한 문제가 발생하면 메시지를 튕기는 서버나 채널을 알기 위해 메시지의 헤더 행을 확인합니다. 필요에 따라 항목을 수정합니다.

메시지 루프의 또 다른 대표적인 원인은 MTA가 자신의 이름 중 하나로 인식하지 않는(인식하도록 구성되지 않은) 네트워크 이름을 사용하여 MTA 호스트로 주소 지정된 메시지를 수신하는 것입니다. 이 문제를 해결하려면 MTA가 **자신의** 이름으로 인식하는 이름 목록에 추가 이름을 추가합니다. 메시지가 루핑 중이라고 MTA가 판단하는 임계값은 구성 가능합니다. `MAX_*RECEIVED_LINES` `option.dat` 옵션(**Sun Java System Messaging Server 6.3 Administration Reference**의 "Option File Format and Available Options")을 참조하십시오. 또한 MTA는 선택적으로 구성 가능합니다. 임계값 초과로

인해 메시지가 강제로 .HELD 상태가 될 때마다 syslog 알림을 생성하려면 HELD_SNDOPR 전역 MTA 옵션을 참조하십시오. Received count exceeded; message held. 라는 syslog 메시지가 존재하면 그러한 상황임을 알 수 있습니다.

Sun Java System Messaging Server 6.3 Administration Reference의 "release"를 실행하거나 다음 단계를 수행하여 .HELD 메시지를 다시 보낼 수 있습니다.

1. .HELD 확장자 이름을 00외의 두 자리 숫자로 바꿉니다(예: .HELD에서 .06으로).

주 – .HELD 파일의 이름을 바꾸기 전에 메시지가 루핑을 중단해야 합니다.

---

2. imsimta cache -sync를 실행합니다. 이 명령을 실행하면 캐시가 업데이트됩니다.
3. imsimta submit *channel* 또는 imsimta run *channel*을 실행합니다.

Received: 헤더 행이 누적되기 때문에 메시지가 다시 .HELD로 표시될 가능성이 있으므로 이 단계를 여러 번 수행해야 할 수 있습니다. 문제가 여전히 계속되는 경우 앞에서 설명한 것처럼 동일한 채널 아래 *.HELD 파일이 다시 만들어집니다. 문제가 해결되었다면 메시지는 대기열에서 삭제되고 배달됩니다.

배달 시도 없이 메시지가 삭제될 수 있게 하려면 **Sun Java System Messaging Server 6.3 Administration Reference**의 "clean"을 참조하십시오.

사용자 또는 도메인 hold 상태에 의한 Messages .HELD

사용자 또는 도메인의 hold 상태로 인한 Messages .HELD 및 그러한 이유로 생겨난 messages .HELD는 보관 채널의 대기열 영역에 저장됩니다. 즉, 보관 채널의 대기열 영역에 있는 .HELD 메시지 파일은 사용자나 도메인 상태에 의한 .HELD로 간주할 수 있습니다.

의심스러운 특성에 의한 Messages .HELD

의심스러운 특성 때문에 생긴 Messages .HELD는 그 특성을 나타냅니다. 사이트에서 **의심스러운 것으로** 규정한 모든 것이 이 특성에 해당될 수 있습니다. MTA 관리자는 이러한 구성 선택 사항과 작업을 알고 있어야 합니다. 그러나 이 MTA의 유일한 관리자 또는 원래 관리자가 아닌 경우, holdlimit 채널 키워드 사용이 구성되었는지(359 페이지 "12.5.9 여러 주소 확장"), MTA 매핑 파일의 주소 기반 *_ACCESS 매핑 테이블에서 $H를 사용하는지 또는 시스템 시브(Sieve) 파일(시스템 수준 imta.filter 파일 또는 sourcefilter나 destinationfilter 채널 키워드를 사용하여 구성, 명명된 채널 수준 시브(Sieve) 필터, 390 페이지 "12.12.4 메일함 필터 파일 위치 지정" 참조)에서 hold 작업을 사용하는지 MTA 구성에서 확인합니다. 그리고 최근에 수동 명령줄 메시지 보관을 수행했는지(예: imsimta qm clean 명령을 통해) 동료 MTA 관리자에게 문의합니다. 또한 시스템 시브(Sieve) 필터에서 또는 사용자 개인 시브(Sieve) 필터에서의 시브(Sieve) 필터 hold 작업 적용이 선택적으로 기록될 수 있습니다. 자세한 내용은 LOG_FILTER 전역 MTA 옵션(**Sun Java System Messaging Server 6.3 Administration Reference**의 "Option File Format and Available Options" 참조)을 참조하십시오.

26.3.9 받은 메시지가 인코딩됨

MTA에서 보낸 메시지는 인코딩된 형식으로 수신됩니다. 예를 들면 다음과 같습니다.

```
Date: Wed, 04 Jul 2001 11:59:56 -0700 (PDT)
From: "Desdemona Vilalobos" <Desdemona@sesta.com>
To: santosh@varrius.com
Subject: test message with 8bit data
MIME-Version: 1.0
Content-type: TEXT/PLAIN; CHARSET=ISO-8859-1
Content-transfer-encoding: QUOTED-PRINTABLE

2=00So are the Bo=F6tes Void and the Coal Sack the same?=
```

이러한 메시지는 MTA 디코더 명령인 `imsimta decode`를 통해 읽을 때 인코딩되지 않은 상태로 표시됩니다. 자세한 내용은 **Sun Java System Messaging Server Administration Reference**를 참조하십시오.

SMTP 프로토콜은 RFC 821에 설명된 것과 같이 ASCII 문자(7비트 문자 세트)의 전송만을 허용합니다. SMTP를 사용한 8비트 문자의 검토되지 않은 전송은 유효하지 않으며 일부 SMTP 서버에 다양한 문제를 일으키는 것으로 알려져 있습니다. 예를 들어, SMTP 서버는 연산 관련 루프로 이동할 수 있습니다. 메시지가 계속해서 다시 보내집니다. 8비트 문자는 서버 충돌을 일으킬 수 있습니다. 결국 8비트 문자 세트는 8비트 데이터를 처리할 수 없는 브라우저 및 메일함을 복잡하게 만들 수 있습니다.

SMTP 클라이언트는 8비트 데이터를 포함하는 메시지를 처리할 때 보낸 사람에게 전달할 수 없는 것으로 메시지를 반환하거나, 메시지를 코드화하거나, RFC 821을 직접 위반하여 메시지 보내기와 같은 세 가지 옵션만을 가지고 있습니다. 하지만 MIME 및 SMTP 확장자의 발명으로 이제 ASCII 문자 세트를 사용하여 8비트 데이터 코드화에 사용할 수 있는 표준 인코딩 옵션이 있습니다.

이전 예에서 수신자는 TEXT/PLAIN의 MIME 내용 유형으로 인코딩된 메시지를 받았으며, 원격 SMTP 서버(MTA SMTP 클라이언트가 메시지를 전송한 서버)는 8비트 데이터의 전송을 지원하지 않았습니다. 하지만 원본 메시지가 8비트 문자를 포함하고 있기 때문에 MTA가 메시지를 인코딩해야 했습니다.

26.3.10 서버측 규칙(SSR)이 작동하지 않음

필터는 메시지 메시지에 적용할 하나 이상의 조건부 작업으로 구성되어 있습니다. 필터는 서버에 저장 및 평가되므로 흔히 서버측 규칙(SSR)이라고 합니다.

이 절에서는 다음 SSR 관련 항목에 대해 설명합니다.

- 823 페이지 “26.3.10.1 사용자 SSR 규칙 테스트”
- 823 페이지 “26.3.10.2 일반 구문 문제”

542 페이지 “18.15 사용자 수준 필터 디버그”를 참조하십시오.

26.3.10.1 사용자 SSR 규칙 테스트

- MTA의 사용자 필터를 확인하려면 다음 명령을 사용합니다.

```
# imsimta test -rewrite -debug -filter user@domain
```

출력에서는 다음 정보를 찾습니다.

```
mmc_open_url called to open ssrf:user@ims-ms
   URL with quotes stripped: ssrd: user@ims-ms
Determined to be a SSRD URL.
   Identifier: user@ims-ms-daemon
Filter successfully obtained.
```

- 추가로 필터의 적용 방법을 보기 위해 `tcp_local` 채널에 `slave_debug` 키워드를 추가할 수 있습니다. 결과는 `tcp_local_slave.log` 파일에 표시됩니다. 충분한 디버깅 정보를 얻으려면 */msg-svr-base*`/config` 디렉토리의 `option.dat` 파일에 `mm_debug=5`를 추가해야 합니다.

26.3.10.2 일반 구문 문제

- 필터에 구문 문제가 있는 경우 `tcp_local_slave.log-*` 파일에서 다음 메시지를 찾습니다.

  `Error parsing filter expression:...`

  - 필터 상태가 양호하면 출력 끝에 필더 정보가 표시됩니다.
  - 필터 상태가 양호하지 않으면 출력 끝에 다음 오류가 표시됩니다.`Address list error -- 4.7.1 Filter syntax error: desdaemona@sesta.com`

    또한 필터 상태가 양호하지 않으면 SMTP RCPT TO 명령에서 다음과 같은 임시 오류 응답 코드를 반환합니다.

```
RCPT TO: user@domain
452 4.7.1 Filter syntax error
```

26.3.11 메일 보내기 버튼을 누른 후 응답이 느림

사용자가 메시지를 보낼 때 지연이 발생하는 경우 메시지 대기열 디스크가 충분히 크게 지정되지 않아 디스크 입력/출력이 줄었기 때문일 수 있습니다. 사용자가 전자 메일 클라이언트에서 보내기 버튼을 누를 때 MTA는 메시지가 메시지 대기열에 적용되어야만 메시지 수신을 완전히 수락합니다. 메시지 대기열 크기 지정에 대한 자세한 내용은 설명서 에서 확인할 수 있습니다.

26.3.12 받은 필드 또는 주소의 로컬 부분에 있는 별표

이제 MTA는 구성한 받은 필드뿐만 아니라 주소의 로컬 부분에서 8비트 문자(단순히 ASCII 문자가 아니라)를 검사하고 이 문자를 별표로 바꿉니다.

26.4 일반 오류 메시지

MTA가 시작되지 않으면 명령줄에 일반 오류 메시지가 표시됩니다. 이 절에서는 일반 오류 메시지를 설명하고 진단합니다.

주 – 고유한 사용자 MTA 구성을 진단하려면 `imsimta test -rewrite -debug` 유틸리티를 사용하여 사용자의 MTA 주소 다시 쓰기 및 채널 매핑 프로세스를 검사합니다. 이 유틸리티를 사용하여 메시지를 실제로 보내지 않고도 해당 구성을 확인할 수 있습니다. 802 페이지 “26.2.1 MTA 구성 확인”을 참조하십시오.

또한 MTA 하위 구성 요소는 이 장에서 설명하지 않은 다른 오류 메시지를 표시할 수도 있습니다. 명령줄 유틸리티 및 구성에 대한 장은 **Sun Java System Messaging Server Administration Reference**를 참조하고 각 하위 구성 요소에 대한 자세한 내용은 5장에서 10장을 참조하십시오. 이 절에서는 다음 오류 유형에 대해 설명합니다.

- 824 페이지 “26.4.1 mm_init 오류”
- 827 페이지 “26.4.2 컴파일된 구성 버전이 일치하지 않는 경우”
- 828 페이지 “26.4.3 스왑 공간 오류”
- 828 페이지 “26.4.4 파일 열기 또는 만들기 오류”
- 828 페이지 “26.4.5 유효하지 않은 호스트/도메인 오류”
- 829 페이지 “26.4.6 SMTP 채널 오류, os_smtp_* 오류”

26.4.1 mm_init 오류

mm_init 오류는 일반적으로 MTA 구성 문제를 나타냅니다. `imsimta test -rewrite` 유틸리티를 실행하면 이러한 오류가 표시됩니다. `imsimta cnbuild`, 채널, 서버 또는 브라우저와 같은 다른 유틸리티에서도 이와 같은 오류를 반환합니다.

일반적으로 발생하는 mm_init 오류는 다음과 같습니다.

- 825 페이지 “26.4.1.1 bad equivalence for alias. . .”
- 825 페이지 “26.4.1.2 cannot open alias include file. . .”
- 825 페이지 “26.4.1.3 duplicate aliases found. . .”
- 825 페이지 “26.4.1.4 duplicate host in channel table. . .”
- 825 페이지 “26.4.1.5 duplicate mapping name found. . .”
- 826 페이지 “26.4.1.6 mapping name is too long. . .”
- 826 페이지 “26.4.1.7 error initializing ch_ facility compiled character set version mismatch”

26.4.1.1 bad equivalence for alias. . .

별칭 파일 항목의 오른쪽의 서식 지정이 잘못되었습니다.

26.4.1.2 cannot open alias include file. . .

별칭 파일에 포함된 파일을 열 수 없습니다.

26.4.1.3 duplicate aliases found. . .

두 개의 별칭 파일 항목의 왼쪽이 동일합니다. 중복된 별칭을 찾아서 제거해야 합니다. 행 번호 XXX인 `error line #XXX` 오류 메시지를 찾습니다. 해당 행에서 중복된 별칭을 수정할 수 있습니다.

26.4.1.4 duplicate host in channel table. . .

이 오류 메시지는 MTA 구성에 공식 호스트 이름이 같은 두 개의 채널 정의가 있다는 것을 표시합니다.

사용자 구성 파일(`imta.cnf`)의 다시 쓰기 규칙(위쪽)에 추가로 생긴 빈 행으로 인해 MTA는 나머지 구성 파일을 채널 정의로 해석하게 됩니다. 파일의 맨 처음 행이 빈 행이 아니어야 합니다. 동일한 패턴(왼쪽)의 다시 쓰기 규칙이 많으므로 MTA는 이 규칙을 고유하지 않은 공식 호스트 이름을 가진 채널 정의로 해석합니다. 모든 중복된 공식 호스트 이름을 가진 채널 정의 및 파일의 위(다시 쓰기 규칙)쪽에 있는 잘못된 모든 빈 행에 대해 MTA 구성을 확인합니다.

26.4.1.5 duplicate mapping name found. . .

이 메시지는 두 개의 매핑 테이블이 같은 이름을 가지고 있다는 것을 나타내며 중복된 매핑 테이블 중 한 개는 제거되어야 합니다. 하지만 매핑 파일의 서식 지정 오류로 인해 MTA에서 무관한 것을 매핑 테이블 이름으로 잘못 해석할 수도 있습니다. 예를 들어, 매핑 테이블 항목을 적절하게 들여쓰지 않으면 MTA에서 항목의 왼쪽이 실질적인 매핑 테이블 이름인 것으로 잘못 생각할 수 있습니다. 일반 형식 매핑 테이블을 검사하고 매핑 테이블 이름을 확인합니다.

주 – 빈 행은 매핑 테이블 이름을 가진 모든 행의 앞뒤에 있어야 합니다. 하지만 어떤 빈 행도 매핑 테이블의 항목 간에 산재해 있으면 안 됩니다.

26.4.1.6 mapping name is too long. . .

이 오류는 매핑 테이블 이름이 너무 길어서 줄여야 함을 의미합니다. 매핑 파일의 서식 지정 오류로 인해 MTA에서 무관한 것을 매핑 테이블 이름으로 잘못 해석할 수도 있습니다. 예를 들어, 매핑 테이블 항목을 적절하게 들여쓰지 않으면 MTA에서 항목의 왼쪽이 실질적인 매핑 테이블 이름인 것으로 잘못 생각할 수 있습니다. 매핑 파일 및 매핑 테이블 이름을 확인합니다.

26.4.1.7 error initializing ch_ facility compiled character set version mismatch

이 메시지가 표시되면 `imsimta chbuild` 명령을 통해 컴파일된 문자 세트 테이블을 다시 컴파일하고 다시 설치해야 합니다. 자세한 내용은 **Sun Java System Messaging Server 6.3 Administration Reference**의 "imsimta chbuild"를 참조하십시오.

26.4.1.8 error initializing ch_ facility no room in. . .

일반적으로 이 오류 메시지는 MTA 문자 세트 내부 테이블의 크기를 조정해야 한다는 것을 의미하며 다음 명령을 통해 컴파일된 문자 세트 테이블을 다시 만듭니다.

```
imsimta chbuild -noimage -maximum -option
imsimta chbuild
```

위와 같이 변경하기 전에는 아무것도 다시 컴파일하거나 다시 시작하지 않도록 합니다. `imsimta chbuild`에 대한 자세한 내용은 **Sun Java System Messaging Server 6.3 Administration Reference**의 "imsimta chbuild"를 참조하십시오.

26.4.1.9 local host alias or proper name too long for system. . .

이 오류는 로컬 호스트 별칭 또는 해당 이름이 너무 길다는 것을 나타냅니다(채널 블록에서 두 번째 또는 후속 이름 중 하나의 오른쪽). 하지만 초기 MTA 구성 파일(예: 다시 쓰기 규칙의 추가적인 빈 행)의 일부 구문 오류로 인해 MTA에서 무관한 것을 채널 정의로 잘못 해석할 수도 있습니다. 구성 파일의 표시된 행을 확인하는 것 외에도 다른 구문 오류에 대해 위의 해당 행을 확인합니다. 특히 MTA에서 이 오류를 표시하는 행이 다시 쓰기 규칙으로 사용되는 경우 반드시 그 위의 추가적인 빈 행을 확인해야 합니다.

26.4.1.10 no equivalence addresses for alias. . .

별칭 파일에 있는 항목의 오른쪽(번역 값)이 없습니다.

26.4.1.11 no official host name for channel. . .

이 오류는 채널 정의 블록에 필수적인 두 번째 행(공식 호스트 이름 행)이 없다는 것을 나타냅니다. 채널 정의 블록에 대한 자세한 내용은 **Sun Java System Messaging Server Administration Reference**의 MTA 구성 및 명령줄 유틸리티 장 및 12 장을 참조하십시오. 각 채널 정의 블록 전후에는 빈 행이 필요하지만 채널 정의의 채널 이름과 공식 호스트 이름 행 사이에 빈 행이 있어서는 안 됩니다. 또한 빈 행은 MTA 구성 파일의 다시 쓰기 규칙 부분에 허용되지 않습니다.

26.4.1.12 official host name is too long

채널(채널 정의 블록의 두 번째 행)의 공식 호스트 이름 길이는 128자의 8진수로 제한됩니다. 채널에 더 긴 공식 호스트 이름을 사용하려면 이를 자리 표시자 이름으로 줄인 다음 다시 쓰기 규칙을 사용하여 긴 이름을 짧은 공식 호스트 이름에 일치시킵니다. l(로컬) 채널 호스트 이름을 사용하면 이 시나리오를 볼 수 있습니다. 예를 들면 다음과 같습니다.

```
Original l Channel:
!delivery channel to local /var/mail store
l subdirs 20 viaaliasrequired maxjobs 7 pool LOCAL_POOL
walleroo.pocofronitas.thisnameismuchtoolongandreallymakesnosensebutitisan
example.monkey.gorilla.orangutan.antidisestablismentarianism.newt.salaman
der.lizard.gecko.komododragon.com

Create Place Holder:
!delivery channel to local /var/mail store
l subdirs 20 viaaliasrequired maxjobs 7 pool LOCAL_POOL
newt

Create Rewrite Rule:
newt.salamander.lizard.gecko.komododragon.com    $U%$D@newt
```

l(로컬) 채널을 사용하는 경우에는 REVERSE 매핑 테이블을 사용해야 합니다. 사용법 및 구문에 대한 자세한 내용은 **Sun Java System Messaging Server Administration Reference**의 MTA configuration 장을 참조하십시오.

초기 MTA 구성 파일의 특정 구문 오류(예: 다시 쓰기 규칙의 추가적인 빈 행)로 인해 MTA에서 무관한 것을 채널 정의로 잘못 해석할 수 있습니다. 이로 인해 의도된 다시 쓰기 규칙이 공식 호스트 이름으로 해석될 수 있습니다. 구성 파일의 표시된 행을 확인하는 것 외에도 다른 구문 오류에 대해 위의 해당 행을 확인합니다. 특히 MTA에서 이 오류를 표시하는 행이 다시 쓰기 규칙으로 사용되는 경우 반드시 그 위의 추가적인 빈 행을 확인해야 합니다.

26.4.2 컴파일된 구성 버전이 일치하지 않는 경우

imsimta cnbuild 유틸리티의 기능 중 하나는 신속하게 로드되는 이미지에 MTA 구성 정보를 컴파일하는 것입니다. 컴파일 형식은 엄격히 정의되며 MTA의 버전에 따라 상당한 차이가 있습니다. 사소한 변경 사항이 패치 릴리스의 일부로 발생할 수 있습니다.

이와 같은 변경 사항이 발생하면 호환되지 않는 형식을 감지할 수 있도록 내부 버전 필드도 변경됩니다. 호환되지 않는 형식이 감지되면 MTA 구성 요소가 위의 오류와 같이 정지합니다. 이 문제에 대한 해결책은 imsimta cnbuild 명령을 사용하여 새로 컴파일된 구성을 생성하는 것입니다.

또한 업데이트된 구성 정보를 얻을 수 있도록 imsimta restart 명령을 사용하여 모든 상주 MTA 서버 프로세스를 다시 시작하는 것도 좋은 방법입니다.

26.4.3 스왑 공간 오류

제대로 작동하게 하려면 사용자의 메시징 시스템에 충분한 스왑 공간을 구성하는 것이 중요합니다. 사용자의 구성에 따라 필수 스왑 공간 크기가 다릅니다. 일반적인 조정 권장 사항으로는 스왑 공간의 크기가 적어도 주 기억 장치 크기의 3배여야 합니다.

다음은 스왑 공간이 없음을 알리는 오류 메시지입니다.

```
jbc_channels: chan_execute [1]: fork failed: Not enough space
```

이 오류를 작업 제어기 로그 파일에서 볼 수도 있습니다. 다른 스왑 공간 오류는 사용자 구성에 따라 다릅니다.

다음 명령을 사용하여 사용한 스왑 공간과 남은 스왑 공간 크기를 확인할 수 있습니다.

- Solaris 시스템: `swap -s` (at the time MTA processes are busy), `ps -elf`, or `tail /var/adm/messages`
- 시스템 `swapinfo` or `tail /var/adm/syslog/syslog.log`

26.4.4 파일 열기 또는 만들기 오류

메시지를 보내려면 MTA는 MTA 메시지 대기열 디렉토리에서 구성 파일을 읽거나 메시지 파일을 만듭니다. 구성 파일은 MTA 또는 MTA의 SKD에 대해 쓰여진 모든 프로그램으로 읽을수 있어야 합니다. 설치하는 동안 적절한 사용 권한이 이 파일에 할당됩니다. 구성 파일을 만드는 MTA 유틸리티 및 절차도 사용 권한을 할당합니다. 시스템 관리자가 해당 파일을 보호하는 경우에는 다른 권한있는 사용자 또는 일부 사이트별 절차를 통해 MTA에서 구성 정보를 읽지 못할 수 있습니다. 이런 경우 "파일 열기" 오류 또는 예기치 않은 동작이 발생합니다. `imsimta test -rewrite` 유틸리티는 구성 파일 읽기에 문제가 발생하면 추가 정보를 보고합니다. **Sun Java System Messaging Server 6.3 Administration Reference**의 “imsimta test”을 참조하십시오.

MTA가 권한이 있는 계정에서는 작동하고 권한이 없는 계정에서는 작동하지 않는 것처럼 보이는 경우에는 MTA 테이블 디렉토리의 파일 사용 권한이 문제의 원인일 수 있습니다. 구성 파일 및 해당 디렉토리에서 사용 권한을 확인합니다. 803 페이지 “26.2.3 중요 파일의 소유권 확인”을 참조하십시오.

일반적으로 "파일 만들기" 오류는 MTA 메시지 대기열 디렉토리에서 메시지 파일을 만드는 중에 발생하는 문제를 나타냅니다. 파일 만들기 문제를 진단하려면 802 페이지 “26.2.2 메시지 대기열 디렉토리 확인”을 참조하십시오.

26.4.5 유효하지 않은 호스트/도메인 오류

브라우저를 통해 주소가 MTA에 제공되는 경우 이 오류가 나타날 수 있습니다. 또는, 오류 반환 메일 메시지의 일부로 오류가 지연되고 반환될 수 있습니다. 두 경우 모두 이

오류 메시지는 MTA가 지정된 호스트에게 메시지를 전달할 수 없다는 것을 나타냅니다. 지정된 호스트에게 메시지를 보낼 수 없는 이유를 확인하려면 다음 문제 해결 절차를 수행해야 합니다.

- 해당 주소가 잘못되었거나, 잘못 옮겨졌거나, 존재하지 않는 호스트 또는 도메인의 이름을 사용하지 않았는지 확인합니다.
- `imsimta test -rewrite` 유틸리티를 통해 해당 주소를 실행합니다. 이 유틸리티에서도 해당 주소에 대해 "유효하지 않은 호스트/도메인" 오류를 반환하는 경우, MTA에는 `imta.cnf` 파일 및 관련 파일에 해당 주소를 처리할 수 있는 규칙이 없는 것입니다. MTA를 올바르게 구성하고 모든 구성 질문에 대해 적절하게 응답했으며 최신 구성 정보를 유지하는지 확인합니다.
- `imsimta test -rewrite`에서 주소에 대한 오류가 발생하지 않으면 MTA는 주소 처리 방법을 결정할 수 있지만 네트워크 전송이 해당 주소를 받아들이지 않습니다. 추가 세부 사항은 전달 시도에서 적절한 로그 파일을 확인할 수 있습니다. 잘못 구성된 도메인 이름 서버가 문제를 발생시킬 수 있지만 일시 네트워크 라우팅 또는 이름 서비스 오류는 오류 메시지를 반환하면 안 됩니다.
- 인터넷을 사용하는 경우 MX 레코드 조회를 지원하는 TCP/IP 채널을 제대로 구성했는지 확인합니다. 인터넷 상에서는 많은 도메인 주소에 직접 액세스할 수 없으며 메일 시스템에서 MX 항목을 제대로 해결해야 합니다. 인터넷을 사용하고 TCP/IP가 MX 레코드를 지원하도록 구성된 경우 MX 지원이 사용 가능하도록 MTA를 구성해야 합니다. 자세한 내용은 341 페이지 “12.4.3 TCP/IP 연결 및 DNS 조회 지원”을 참조하십시오. TCP/IP 패키지가 MX 레코드 조회를 지원하도록 구성되지 않은 경우에는 MX 전용 도메인에 연결할 수 없습니다.

26.4.6 SMTP 채널 오류, os_smtp_* 오류

다음 오류는 반드시 MTA 오류인 것은 아닙니다. os_smtp_open, os_smtp_read, 및 os_smtp_write 등 os_smtp_* 오류이러한 오류는 MTA가 네트워크 계층에서 발생한 문제를 보고할 때 생성됩니다. 예를 들어, os_smtp_open 오류는 원격측 네트워크 연결을 열 수 없다는 것을 의미합니다. MTA가 주소 지정 오류나 채널 구성 오류로 인해 잘못된 시스템에 연결되도록 구성되어 있을 수 있습니다. 일반적으로 `os_smtp_*` errors는 DNS 또는 네트워크 연결 문제로 인해, 특히 이전에 작업 채널 또는 주소였다면 발생합니다. `os_smtp_read` 또는 `os_smtp_write` 오류는 보통 다른 쪽에서 연결을 중단하거나 네트워크 문제로 인해 연결이 중단되었다는 것을 나타냅니다.

네트워크 및 DNS 문제는 일시적인 경우가 많습니다. 따라서 가끔 발생하는 os_smtp_* 오류는 신경쓰지 않아도 됩니다. 하지만 이 오류가 지속적으로 나타난다는 것은 기본 네트워크 문제를 나타내는 것일 수 있습니다.

특정 os_smtp_* 오류에 대한 자세한 내용을 얻으려면 해당 채널에서 디버깅을 활성화합니다. 시도된 SMTP 대화의 세부 사항을 표시하는 디버그 채널 로그 파일을 조사합니다. 특히 SMTP 대화 중에 언제 네트워크 문제가 발생했는지 확인합니다. 그

시간으로 네트워크 또는 원격측 문제의 유형을 알 수도 있습니다. 경우에 따라 네트워크 수준 디버깅(예: TCP/IP 패킷 추적)을 수행하여 보내거나 받은 메시지를 확인할 수 있습니다.

◆ ◆ ◆ 27 장

27 Messaging Server 모니터링

일반적으로 제대로 계획 및 구성된 서버는 관리자의 광범위한 개입 없이 작동합니다. 그러나 서버에서 문제의 징후를 모니터하는 것은 관리자가 해야 할 일입니다. 이 장에서는 Messaging Server의 모니터링에 대해 설명합니다. 이 장은 다음 내용으로 구성되어 있습니다.

문제 해결 절차는 26 장에서 확인할 수 있습니다.

27.1 자동 모니터링 및 재시작

Messaging Server는 서비스를 투명하게 모니터하고 서비스가 실패하거나 응답하지 않는 경우(즉, 서비스가 중지되거나 멈춘 경우) 서비스를 자동으로 다시 시작하는 방법을 제공합니다. Messaging Server는 IMAP, POP, HTTP, 작업 제어기, 디스패처 및 MMP 서버를 비롯한 모든 메시지 저장소, MTA 및 MMP 서비스를 모니터할 수 있지만SMS 또는 TCP/SNMP 서버와 같은 다른 서비스는 모니터하지 않습니다. TCP/SNMP는 작업 제어기에서 모니터링합니다. 107 페이지 "4.5 실패했거나 응답이 없는 서비스의 자동 재시작" 및 850 페이지 "27.8.9 msprobe 및 watcher 기능을 사용하여 모니터링"을 참조하십시오.

27.2 일상적인 모니터링 작업

일상적으로 수행해야 하는 가장 중요한 작업은 포스트마스터 메일 검사, 로그 파일 모니터링 및 `stored` 유틸리티 설정입니다. 아래에서는 이러한 작업에 대해 설명합니다.

27.2.1 포스트마스터 메일 검사

Messaging Server에는 포스트마스터 전자 메일용으로 설정된 관리 메일링 목록이 미리 정의되어 있습니다. 이 메일링 목록에 속한 모든 사용자는 포스트마스터로 주소 지정된 메일을 자동으로 받게 됩니다.

포스트마스터 메일에 대한 규칙은 RFC822에서 정의됩니다. RFC822에 따르면 모든 전자 메일 사이트에서 postmaster라는 이름의 사용자 또는 메일링 목록으로 주소 지정된 메일을 수락해야 하며 이 주소로 보내진 메일은 실제 당사자에게 전달되어야 합니다. postmaster@host.domain으로 보내진 모든 메일은 포스트마스터 계정 또는 메일링 목록으로 보내집니다.

일반적으로 포스트마스터 주소는 사용자가 메일 서비스에 대한 전자 메일을 보내야 하는 곳입니다. 포스트마스터는 예를 들어, 서버 응답 시간에 대한 메일을 로컬 사용자로부터 받거나 서버로 메일을 보내는 데 문제가 있다는 내용의 메일을 다른 서버 관리자로부터 받을 수 있습니다. 포스트마스터 메일은 매일 확인해야 합니다.

특정 오류 메시지를 포스트마스터 주소로 보내도록 서버를 구성할 수도 있습니다. 예를 들어, MTA가 메시지를 라우팅 또는 전달하지 못할 경우 포스트마스터 주소로 보내진 전자 메일을 통해 알림을 받을 수 있습니다. 또한 예외적인 상황에 대한 경고(디스크 공간 부족, 서버 응답 실패 등에 대한)를 포스트마스터에게 보낼 수도 있습니다.

27.2.2 로그 파일 모니터링 및 유지 관리

Messaging Server는 지원되는 각각의 주요 프로토콜이나 서비스(SMTP, IMAP, POP 및 HTTP)에 대한 별도의 로그 파일 집합을 생성합니다. 이러한 로그 파일은 *msg-svr-base*`/data/log`에 위치합니다. 로그 파일은 정기적으로 모니터링해야 하며, 특히 서버에 문제가 있는 경우에는 이러한 모니터링이 더욱 필요합니다.

로깅이 서버 성능에 영향을 줄 수 있다는 것을 유의하십시오. 더 자세한 로깅을 지정할수록 일정한 시간 동안 로그 파일이 차지하는 디스크 공간이 더 많아집니다. 따라서 효과적이면서 실제적인 로그 회전, 만료 및 백업 정책을 서버에 정의해야 합니다. 서버의 로깅 정책 정의에 대한 자세한 내용은 25 장을 참조하십시오.

27.2.3 msprobe 유틸리티 설정

`msprobe` 유틸리티는 자동으로 모니터링을 수행하고 기능을 다시 시작합니다. 자세한 내용은 850 페이지 “27.8.9 msprobe 및 watcher 기능을 사용하여 모니터링”을 참조하십시오.

27.3 시스템 성능 모니터링

이 장에서는 Messaging Server 모니터링에 초점을 맞추고 있지만 서버가 상주하는 시스템도 모니터링해야 합니다. 적절하게 구성된 서버는 잘못 구성된 시스템에서 제대로 작동할 수 없으며 하드웨어가 전자 메일 로드를 감당할 만큼 성능이 충분하지 않다는 서버 오류의 증상이 나타날 수 있습니다. 시스템 성능을 모니터하기 위한 절차가 플랫폼마다 차이가 있고 해당 플랫폼의 시스템 설명서를 참조할 필요가 있다는 점에서 이 장에서 이러한 절차를 자세하게 다루지는 않습니다. 여기에서는 성능 모니터링을 위한 다음 절차에 대해 설명합니다.

- 833 페이지 "27.3.1 종단간 메시지 전달 시간 모니터링"
- 833 페이지 "27.3.2 디스크 공간 모니터링"
- 836 페이지 "27.3.3 CPU 사용 모니터링"

27.3.1 종단간 메시지 전달 시간 모니터링

전자 메일은 제때 전달되어야 합니다. 이는 서비스 계약 요구 사항일 뿐 아니라 메일을 가능한 신속하게 전달하는 것은 바람직한 정책입니다. 종단간의 느린 메일 전달은 여러 가지 원인으로 인해 발생할 수 있습니다. 예를 들어, 서버가 제대로 작동하지 않거나, 일정 기간 동안 과도한 메시지 로드가 발생했거나, 기존 하드웨어 자원이 용량을 초과했을 수 있습니다.

27.3.1.1 느린 종단간 메시지 전달 시간의 증상

메일을 전달하는 데 평소보다 오래 걸립니다.

27.3.1.2 종단간 메시지 전달 시간 모니터

- 메시지를 보내고 받는 임의의 기능을 사용합니다. 서버 홉 간의 헤더 시간과 출발 시점 및 검색 간의 시간을 비교합니다. 842 페이지 "27.8.1 immonitor-access"를 참조하십시오.

27.3.2 디스크 공간 모니터링

부족한 디스크 공간은 메일 서버 문제와 오류의 가장 일반적인 원인 중 하나입니다. MTA 대기열 또는 메시지 저장소에 쓰기 위한 공간이 없을 경우 메일 서버에 오류가 발생합니다. 또한 로그 파일이 모니터링 및 정리되지 않을 경우 모든 디스크 공간이 채워질 때까지 로그 파일의 크기가 증가할 수 있습니다.

새 메시지가 메일함에 전달되면 메시지 저장소 분할 영역의 크기가 증가합니다. 예를 들어, 메시지 저장소 할당량이 적용되지 않을 경우 메시지 저장소가 분할 영역에 사용할 수 있는 디스크 공간을 초과할 수 있습니다. 디스크 공간 부족이 발생하는 또 다른 원인으로는 MTA 메시지 대기열이 너무 커지는 것을 들 수 있습니다. 또한 로그 파일

모니터링 기능에 문제가 있거나 로그 파일이 관리가 불가능할 정도로 커지는 경우도 원인이 될 수 있습니다. LDAP, MTA 및 Message Access와 같은 다양한 로그 파일이 존재하며 이러한 로그 파일은 각각 다른 디스크에 저장될 수 있다는 것을 유의하십시오.

27.3.2.1 디스크 공간 문제의 증상

공간이 부족해지는 디스크나 분할 영역에 따라 여러 다른 증상이 발생할 수 있습니다. MTA 대기열이 오버플로되고 SMTP 연결을 거부하거나, 메시지가 ims_master 대기열에 남아 있으면서 메시지 저장소로 전달되지 않거나, 로그 파일이 오버플로될 수 있습니다.

메시지 저장소 분할 영역이 모두 채워지면 메시지 액세스 데몬이 실패할 수 있으며 메시지 저장소 데이터가 손상될 수 있습니다. `imexpire` 및 `reconstruct`와 같은 메시지 저장소 유지 보수 유틸리티는 손상을 복구하고 디스크 사용량을 줄일 수 있습니다. 그러나 이러한 유틸리티를 사용하려면 추가 디스크 공간이 필요하며 전체 디스크를 차지하는 분할 영역을 복구하려면 잠재적으로 중단 시간이 발생할 수 있습니다.

27.3.2.2 디스크 공간 모니터

시스템 구성에 따라 다양한 디스크와 분할 영역을 모니터링해야 할 수 있습니다. 예를 들어, MTA 대기열, 메시지 저장소 및 로그 파일이 각기 다른 디스크/분할 영역에 상주할 수 있습니다. 이 경우 각 공간에 대한 모니터링이 필요하며 각 공간을 모니터하는 방법이 다를 수 있습니다.

Messaging Server는 메시지 저장소 디스크 사용량을 모니터링하고 분할 영역이 사용 가능한 모든 디스크 공간을 차지하는 것을 방지하기 위한 특정 방법을 제공합니다.

다음 단계를 수행하여 메시지 저장소의 디스크 공간 사용을 모니터링할 수 있습니다.

- 메시지 저장소 디스크 사용량을 모니터링하기 위한 매개 변수를 설정합니다.
- 디스크 사용량 임계값에 도달하면 메시지 저장소 분할 영역을 잠급니다.

자세한 내용은 뒤이어 나오는 834 페이지 “메시지 저장소 모니터링” 및 835 페이지 “메시지 저장소 분할 영역 모니터링” 절을 참조하십시오.

메시지 저장소 모니터링

메시지 저장소의 디스크 사용은 용량의 75%를 초과하지 않도록 하는 것이 좋습니다. `configutil` 유틸리티로 다음 경보 속성을 구성하여 메시지 저장소 디스크 사용을 모니터할 수 있습니다.

- `alarm.diskavail.msgalarmstatinterval`
- `alarm.diskavail.msgalarmthreshold`
- `alarm.diskavail.msgalarmwarninginterval`
- `alarm.diskavail.msgalarmdescription`

이러한 매개 변수를 설정함으로써 시스템이 디스크 공간을 모니터하는 빈도와 경고를 보내야 하는 상황을 지정할 수 있습니다. 예를 들어, 디스크 공간을 600초 간격으로 모니터하려는 경우 다음 명령을 지정합니다.

```
configutil -o alarm.diskavail.msgalarmstatinterval -v 600
```

사용 가능한 디스크 공간이 20% 이하로 내려갈 때마다 경고를 받으려면 다음 명령을 지정합니다.

```
configutil -o alarm.diskavail.msgalarmthreshold -v 20
```

이러한 매개 변수에 대한 자세한 내용은 표 27–6을 참조하십시오.

메시지 저장소 분할 영역 모니터링

분할 영역이 사용 가능한 디스크 공간의 지정된 비율보다 많은 공간을 차지할 경우 메시지 저장소 분할 영역에 메시지가 전달되지 않게 할 수 있습니다. 두 개의 `configutil` 매개 변수를 설정하여 이 기능을 활성화하고 디스크 사용 임계값을 지정합니다.

이 기능을 사용하면 메시지 저장소 데몬이 분할 영역의 디스크 사용량을 모니터링합니다. 디스크 사용량이 증가하면 저장소 데몬은 분할 영역을 더 자주(100분에 한 번에서 1분에 한 번에까지 범위) 동적으로 검사합니다.

디스크 사용이 지정된 임계값보다 높아지면 저장소 데몬은 다음을 수행합니다.

- 분할 영역을 잠급니다. 받는 메시지는 MTA 메시지 대기열에서 보관되지만 메시지 저장소 분할 영역의 메일함에 전달되지 않습니다.
- 메시지를 기본 로그 파일에 기록합니다.
- 전자 메일 알림을 포스트마스터에게 보냅니다. (`configutil` 매개 변수 `alarm.msgalarmnoticercpt`를 설정하여 전자 메일의 수신자를 변경할 수 있습니다.)

디스크 사용량이 임계값 아래로 내려가면 분할 영역의 잠금이 해제되고 메시지가 다시 저장소로 전달됩니다.

`configutil` 매개 변수는 다음과 같습니다.

- `local.store.checkdiskusage`는 분할 영역 모니터링 기능을 활성화합니다.

  허용되는 값: `yes`, `no`

  기본값: `yes`

- `local.store.diskusagethreshold`는 디스크 사용량 임계값을 지정합니다. `local.store.diskusagethreshold`의 값은 1 - 99%입니다.

  기본값: 99

분할 영역을 다시 나누거나 로컬 메시지 저장소에 더 많은 디스크 공간을 할당할 수 있는 기회를 얻을 수 있도록 디스크 사용량 임계값의 비율을 충분히 낮게 설정해야 합니다.

예를 들어, 분할 영역이 시간당 2%의 비율로 디스크 공간을 채우며 로컬 메시지 저장소의 추가 디스크 공간을 할당하는 데 1시간이 걸린다고 가정합니다. 이 경우에는 디스크 사용량 임계값을 98%보다 낮은 값으로 설정해야 합니다.

MTA 대기열 및 로깅 공간 모니터링

MTA 대기열 디스크 및 로깅 공간 디스크 사용을 모니터링해야 합니다.

로깅 공간 관리에 대한 자세한 내용은 25 장을 참조하십시오. 예를 들어, `mail.log` 파일을 모니터링하는 방법을 보려면 761 페이지 "25.3 MTA 메시지 및 연결 로그 관리"를 참조하십시오.

27.3.3 CPU 사용 모니터링

CPU 사용량이 많다는 것은 해당 사용 수준에 맞는 CPU 용량이 부족하거나 일부 프로세스가 적절한 한도 이상의 CPU 주기를 사용 중임을 의미합니다.

27.3.3.1 CPU 사용 문제의 증상

시스템 응답 시간이 저하되고 사용자가 로그인하는 데 시간이 오래 걸리며 전달 속도가 느려집니다.

27.3.3.2 CPU 사용 모니터

CPU 사용을 모니터하는 작업은 플랫폼별로 차이가 있습니다. 관련된 플랫폼 설명서를 참조하십시오.

27.4 MTA 모니터링

이 절은 다음과 같은 하위 절로 구성되어 있습니다.

27.4.1 메시지 대기열 크기 모니터링

메시지 대기열이 과도하게 커지는 것은 메시지가 전달되지 않거나, 메시지 전달이 지연되거나, 시스템이 전달할 수 있는 것보다 빠른 속도로 메시지가 도착하기 때문일 수 있습니다. 이 문제는 시스템에 쇄도하는 막대한 수의 메시지로 인한 서비스 거부 공격이나 작업 제어기가 실행되지 않는 등의 여러 이유로 발생할 수 있습니다.

메시지 대기열에 대한 자세한 내용은 177 페이지 "8.5.2 채널 메시지 대기열", 814 페이지 "26.3.6 메시지가 대기열에서 제외되지 않음" 및 817 페이지 "26.3.7 MTA 메시지가 전달되지 않음"을 참조하십시오.

27.4.1.1 메시지 대기열 문제의 증상

- 디스크 공간 사용량이 증가합니다.
- 사용자가 메시지를 제때에 받지 못합니다.
- 메시지 대기열 크기가 비정상적으로 커집니다.

27.4.1.2 메시지 대기열의 크기 모니터

메시지 대기열을 모니터링하는 최선의 방법은 `imsimta qm` 및 `imsimta summarize`을 사용하는 것입니다. 849 페이지 “27.8.6 imsimta qm 카운터”를 참조하십시오.

또한 대기열 디렉토리 *msg-svr-base*`/data/queue/`)의 파일 수를 모니터링할 수도 있습니다. 파일 수는 사이트마다 차이가 있으므로 "너무 많다"고 판단하는 데 기준이 되는 내역을 작성해야 합니다.2주 이상 동안에 대기열 파일의 크기를 기록하여 대략적인 평균을 구하는 방법으로 이러한 기준 내역을 작성할 수 있습니다.

27.4.2 전달 실패 비율 모니터링

전달 실패는 메시지를 외부 사이트로 전달하려는 시도가 실패한 것입니다. 전달 실패 비율이 높다는 것은 DNS 서버를 사용할 수 없거나 원격 서버에서 연결 응답 시간이 초과하는 등의 네트워크 문제가 존재한다는 것을 나타낼 수 있습니다.

27.4.2.1 전달 실패 비율의 증상

외부적인 증상은 없습니다. 다수의 `Q` 레코드가 `mail.log_current`에 나타납니다.

27.4.2.2 전달 실패의 비율 모니터

전달 실패는 로깅 항목 코드 `Q`로 MTA 로그에 기록됩니다. *msg-svr-base* `/data/log/mail.log_current` 파일에서 레코드를 확인합니다. 예:

```
mail.log:06-Oct-2003 00:24:03.66 501d.0b.9 ims-ms Q 5 durai.balusamy@Sun.COM
rfc822;durai.balusamy@Sun.COM durai@ims-ms-daemon
<00ce01c38bda$c7e2b240$6501a8c0@guindy> Mailbox is busy
```

27.4.3 인바운드 SMTP 연결 모니터링

특정 IP 주소에서 인바운드 SMTP 연결 수가 비정상적으로 증가한다는 것은 다음을 의미할 수 있습니다.

- 외부 사용자가 메일 전달을 시도하고 있습니다.
- 외부 사용자가 서비스 거부 공격을 시도하고 있습니다.

27.4.3.1 인증되지 않은 SMTP 연결의 증상

- **메일을 전달하는 외부 사용자**: 외부적인 증상은 없습니다.

- **서비스 거부 공격**: 메시지 요청으로 SMTP 서버를 오버로드하려는 외부 시도가 있습니다.

27.4.3.2 인바운드 SMTP 연결 모니터

- **메일을 전달하는 외부 사용자:***msg-svr-base*/log/mail.log_current에서 로깅 항목 코드 J(거부된 전달)가 있는 레코드를 확인합니다. 원격 IP 주소의 로깅을 설정하려면 다음 행을 option.dat 파일에 추가합니다.

  ```
  log_connection=1
  ```

  이 기능을 사용 가능하게 하면 약간의 성능 저하가 발생한다는 점을 유의하십시오.
- **서비스 거부 공격:**netstat 명령을 실행하고 SMTP 포트(기본값: 25)에서 연결을 검사하여 어떤 사용자가 얼마나 많이 SMTP 서버에 연결되어 있는지 확인할 수 있습니다. 예:

```
Local address       Remote address                                State
192.18.79.44.25     192.18.78.44.56035    32768   0  32768   0    CLOSE_WAIT
192.18.79.44.25     192.18.136.54.57390    8760   0  24820   0    ESTABLISHED
192.18.79.44.25     192.18.26.165.48508   33580   0  24820   0    TIME_WAIT
```

시스템에서 특정 읽기 작업이 비정상적인지 확인할 수 있으려면 우선 적절한 수의 SMTP 연결과 상태(ESTABLISHED, CLOSE_WAIT 등)를 확인해야 한다는 것을 유의하십시오.

다수의 연결이 SYN_RECEIVED 상태로 있을 경우 네트워크 연결이 끊어졌거나 서비스 거부 공격이 발생한 것일 수 있습니다. 또한 SMTP 서버 프로세스의 수명이 제한됩니다. 이것은 dispatcher.cnf 파일에서 MTA 구성 변수 MAX_LIFE_TIME을 통해 제어합니다. 기본값은 86,400초(1일)입니다. 마찬가지로 MAX_LIFE_CONNS는 서버 프로세스가 수명 한도 내에서 처리할 수 있는 최대 연결 수를 지정합니다. 특정 SMTP 서버가 오래 지속될 경우 이를 조사하는 것이 필요할 수 있습니다.

27.4.4 디스패처 및 작업 제어기 프로세스 모니터링

MTA가 작동하려면 디스패처 및 작업 제어기 프로세스가 실행 중이어야 합니다. 각 종류의 프로세스는 하나만 존재해야 합니다.

27.4.4.1 디스패처 및 작업 제어기 프로세스가 다운된 경우의 증상

디스패처가 다운되었거나 자원이 부족할 경우 SMTP 연결이 거부됩니다.

작업 제어기가 다운된 경우 대기열 크기가 증가합니다.

27.4.4.2 디스패처 및 작업 제어기 프로세스 모니터

dispatcher 및 job_controller라는 프로세스가 존재하는지 확인합니다. 803 페이지 “26.2.4 작업 제어기 및 디스패처 실행 확인”을 참조하십시오.

27.5 LDAP Directory Server 모니터링

이 절은 다음과 같은 하위 절로 구성되어 있습니다.

27.5.1 slapd 모니터링

LDAP Directory Server(slapd)는 메시징 시스템에 대한 디렉토리 정보를 제공합니다. slapd가 다운될 경우 시스템이 제대로 작동하지 않습니다. slapd 응답 시간이 너무 느릴 경우 로그인 속도나 LDAP 조회가 필요한 다른 트랜잭션에 영향을 줍니다.

27.5.1.1 slapd 문제의 증상

- 클라이언트 POP, IMAP 또는 웹 메일 인증이 실패하거나 예상보다 느려집니다.
- MTA가 제대로 작동하지 않습니다.

27.5.1.2 slapd 모니터

- ns-slapd 프로세스가 실행 중인지 확인합니다.
- slapd-*instance*/logs/에서 slapd 로그 파일 access 및 errors를 확인합니다.
- 사용자를 검색하는 동안 ns-slapd 응답 시간을 확인합니다.
- 842 페이지 “27.8.1 immonitor-access”를 참조하십시오.

27.6 메시지 액세스 모니터링

이 절은 다음과 같은 하위 절로 구성되어 있습니다.

27.6.1 imapd, popd 및 httpd 모니터링

이러한 프로세스는 IMAP, POP 및 웹 메일 서비스에 대한 액세스를 제공합니다. 이러한 프로세스가 실행 중이 아니거나 응답하지 않을 경우 서비스는 제대로 작동하지 않습니다. 서비스가 실행 중이지만 오버로드된 경우 모니터링을 수행하여 문제를 감지하고 더 적절하게 서비스를 구성할 수 있습니다.

27.6.1.1 imapd, popd 및 httpd 문제의 증상

연결이 거부되며 시스템의 연결 속도가 너무 느려집니다. 예를 들어, IMAP가 실행 중이 아닐 때 IMAP에 직접 연결하려고 시도하면 다음과 같은 메시지가 나타납니다.

```
telnet 0 143 Trying 0.0.0.0... telnet: Unable to connect to remote host:
Connection refused
```

클라이언트와의 연결을 시도할 경우에는 다음과 같은 메시지가 표시됩니다.

```
Client is unable to connect to the server at the location you have specified. The
server may be down or busy.
```

27.6.1.2 imapd, popd 및 httpd 모니터

- watcher 및 msprobe를 사용하여 모니터할 수 있습니다. 107 페이지 "4.5 실패했거나 응답이 없는 서비스의 자동 재시작" 및 850 페이지 "27.8.9 msprobe 및 watcher 기능을 사용하여 모니터링"을 참조하십시오.
- SNMP를 사용하여 모니터할 수 있습니다.

  SNMP가 설정된 경우 SNMP는 이러한 프로세스를 모니터하는 매우 적절한 방법이 됩니다. 부록 A을 참조하십시오. 서버 정보는 네트워크 서비스 모니터링 MIB에 있습니다.
- 로그 파일을 검사합니다.

  msg-svr-base/log/*service* 디렉토리를 확인합니다. 여기서 *service*는 http, IMAP 또는 POP가 될 수 있습니다. 이 디렉토리에는 여러 로그 파일이 존재합니다. 파일 이름 중 하나는 *service*의 이름(imap, pop, http)이며 나머지는 서비스 이름 외에 일련 번호와 날짜가 서비스 이름에 연결되어 있습니다. 예를 들면 다음과 같습니다.

  ```
  imap imap.29.1010221593 imap.31.1010394412 imap.33.1010567224
  ```

  서비스 이름만 가진 파일이 가장 최신 로그입니다. 나머지 파일은 일련 번호(여기에서는 29, 31, 33)로 순서가 정해지며 일련 번호가 가장 높은 파일이 그 다음의 최신 로그입니다. (25 장 참조)

  서버가 종료된 경우 다음과 같은 메시지가 나타날 수 있습니다.

  imap.12.1065431243:[07/Oct/2003:01:15:43 -0700] gotmail-2 imapd[20525]: General Warning: Sun Java System Messaging Server IMAP4 6.1 (built Sep 24 2003) shutting down
- counterutil을 사용하여 검사할 수 있습니다. 843 페이지 "27.8.3 counterutil" 및 **Sun Java System Messaging Server 6.3 Administration Reference**의 "counterutil"을 참조하십시오.
- 플랫폼별 명령을 실행하여 imapd, popd 및 httpd 프로세스가 실행 중인지 확인합니다. 예를 들어, Solaris에서는 ps 명령을 사용하여 imapd, popd 및 mshttpd를 찾을 수 있습니다.
- 852 페이지 "27.8.9.1 경보 메시지"에 설명된 서버 응답 구성 매개 변수를 설정하여 지정된 서버 성능 임계값에 대한 경보를 설정할 수 있습니다.
- 842 페이지 "27.8.1 immonitor-access"를 참조하십시오.

27.7 메시지 저장소 모니터링

메시지는 데이터베이스에 저장됩니다. 디스크상의 사용자 배포, 메일함 크기 및 디스크 요구 사항은 저장소 성능에 영향을 미칩니다. 이러한 내용은 다음 절에서 설명합니다.

27.7.1 stored 모니터링

stored는 메시지 데이터베이스의 교착 상태 및 트랜잭션 작업을 수행하고 에이징 정책을 적용하며 디스크에 저장된 메시지를 정리 및 지우는 등의 중요한 여러 작업을 수행합니다. stored의 실행이 중지되면 Messaging Server에서 결과적으로 문제가 발생합니다. start-msg가 실행될 때 stored가 시작되지 않을 경우 다른 프로세스는 시작되지 않습니다. stored에 대한 자세한 내용은 **Sun Java System Messaging Server 6.3 Administration Reference**의 "stored"를 참조하십시오.

27.7.1.1 stored 문제의 증상

외부적인 증상은 없습니다.

27.7.1.2 stored 모니터

- stored 프로세스가 실행 중인지 확인합니다. stored는 *msg-svr-base*/data/proc에서 pidfile.store라는 pid 파일을 작성 및 업데이트합니다. pid 파일은 복구 시에 init 상태를 표시하고 준비가 되었을 때 ready 상태를 표시합니다. 예를 들면 다음과 같습니다.

  ```
  231: cat store
  28250
  ready
  ```

  첫 번째 행의 숫자는 stored의 프로세스 아이디입니다.

  ```
  232: ps -eaf | grep stored
  inetuser 28250 1 0 Jan 05 ? 8:44
  /opt/SUNWmsgsr/lib/stored -d
  ```

- *msg-svr-base*/store/mboxlist에서 작성된 로그 파일을 확인합니다. 모든 로그 파일이 직접적인 stored 문제로 인해 작성된 것은 아니라는 점을 유의하십시오. imapd가 중지되거나 데이터베이스 문제가 있을 경우에도 로그 파일이 작성될 수 있습니다.
- *msg-svr-base*/config의 다음 파일에서 타임스탬프를 확인합니다.

  stored.ckp - 검사점 지정 시도가 있을 때마다 수정됩니다. 1분마다 시간이 기록되도록 해야 합니다. stored.lcu - db 로그가 정리될 때마다 수정됩니다. 5분마다 시간이 기록되도록 해야 합니다. stored.per - 사용자 단위 db 쓰기가 생성될 때마다 수정됩니다. 60분마다 시간이 기록되도록 해야 합니다.

- 기본 로그 파일 *msg-svr-base*/log/default/default에서 stored 메시지를 확인합니다.
- watcher 및 msprobe를 사용하여 모니터할 수 있습니다. 107 페이지 "4.5 실패했거나 응답이 없는 서비스의 자동 재시작" 및 850 페이지 "27.8.9 msprobe 및 watcher 기능을 사용하여 모니터링"을 참조하십시오.

27.7.2 메시지 저장소 데이터베이스 잠금의 상태 모니터링

데이터베이스 잠금의 상태는 다른 서버 프로세스에 의해 유지됩니다. 이러한 데이터베이스 잠금은 메시지 저장소의 성능에 영향을 줄 수 있습니다. 교착 상태의 경우 메시지가 적절한 속도로 저장소에 삽입되지 않으며 ims-ms 채널 대기열의 크기가 결과적으로 더 증가합니다. 이는 대기열을 백업해야 할 정당한 이유가 되며, 따라서 문제를 진단하기 위해서는 대기열 길이의 내역을 갖고 있는 것이 유용합니다.

27.7.2.1 메시지 저장소 데이터베이스 잠금 문제의 증상

트랜잭션 수가 누적되며 해결되지 않습니다.

27.7.2.2 메시지 저장소 데이터베이스 잠금 모니터

imcheck -s (used to be counterutil -o db_lock) 명령을 사용합니다.

27.8 모니터링을 위한 유틸리티와 도구

다음 도구를 모니터링에 사용할 수 있습니다.

27.8.1 immonitor-access

immonitor-access 는 메일 전달(SMTP 서버), 메시지 액세스 및 저장(POP 및 IMAP 서버), 디렉토리 서비스(LDAP 서버) 및 HTTP 서버와 같은 Messaging Server 구성 요소/프로세스의 상태를 모니터링합니다. 이 유틸리티는 다양한 서비스의 응답 시간과 메시지를 전송 및 검색하는 데 걸린 총 왕복 시간을 측정합니다. 디렉토리 서비스는

디렉토리에서 지정된 사용자를 조회하고 응답 시간을 측정하는 방법으로 모니터링합니다. 메일 전달은 메시지(SMTP)를 보내는 방법으로 모니터하며 메시지 액세스 및 저장은 메시지를 검색하는 방법으로 모니터링합니다. HTTP 서버에 대한 모니터링은 HTTP 서버가 작동하여 실행 중인지 확인하는 것으로 제한됩니다.

자세한 지침은 **Sun Java System Messaging Server 6.3 Administration Reference**의 "immonitor-access"를 참조하십시오.

27.8.2 imcheck

`imcheck -s`를 사용하여 로그 및 트랜잭션을 포함한 데이터베이스 통계를 모니터링합니다.

27.8.3 counterutil

이 유틸리티는 다른 시스템 카운터에서 얻은 통계를 제공합니다. 사용 가능한 카운터 객체의 최신 목록은 다음과 같습니다.

```
# /opt/SUNWmsgsr/sbin/counterutil -l
Listing registry (/opt/SUNWmsgsr/data/counter/counter)
numobjects = 11
refcount = 1
created = 25/Sep/2003:02:04:55 -0700
modified = 02/Oct/2003:22:48:55 -0700
     entry = alarm
     entry = diskusage
     entry = serverresponse     entry = imapstat
     entry = httpstat
     entry = popstat
     entry = cgimsg
```

각 항목은 카운터 객체를 나타내며 해당 객체에 대해 유용한 여러 카운트를 제공합니다. 이 절에서는 `alarm`, `diskusage`, `serverresponse`, `popstat`, `imapstat` 및 `httpstat` 카운터 객체에 대해서만 설명합니다. `counterutil` 명령 사용에 대한 자세한 내용은 **Sun Java System Messaging Server 6.3 Administration Reference**의 "counterutil"을 참조하십시오.

27.8.3.1 counterutil 출력

`counterutil`은 다양한 플래그를 가집니다. 이 유틸리티의 명령 형식은 다음과 같습니다.

`counterutil -o` *CounterObject* `-i 5 -n 10`

여기서

`-o` *CounterObject*는 카운터 객체 `alarm`, `diskusage`, `serverresponse`, `popstat`, `imapstat` 및 `httpstat`를 나타냅니다.

`-i 5`는 5초 간격을 지정합니다.

`-n 10`은 반복 횟수(기본값: 무한대)를 나타냅니다.

`counterutil`의 사용 예는 다음과 같습니다.

```
# counterutil -o imapstat -i 5 -n 10
Monitor counteroobject (imapstat)
registry /gotmail/iplanet/server5/msg-gotmail/counter/counter opened
counterobject imapstat opened

count = 1 at 972082466 rh = 0xc0990 oh = 0xc0968

global.currentStartTime [4 bytes]: 17/Oct/2000:12:44:23 -0700
global.lastConnectionTime [4 bytes]: 20/Oct/2000:15:53:37 -0700
global.maxConnections [4 bytes]: 69
global.numConnections [4 bytes]: 12480
global.numCurrentConnections [4 bytes]: 48
global.numFailedConnections [4 bytes]: 0
global.numFailedLogins [4 bytes]: 15
global.numGoodLogins [4 bytes]: 10446
...
```

27.8.3.2 counterutil을 사용한 경보 통계

이러한 경보 통계는 `stored`에 의해 보내진 경보를 나타냅니다.경보 카운터는 다음 통계를 제공합니다.

표 27–1 `counterutil alarm` 통계

접미어	설명
`alarm.countoverthreshold`	임계값을 초과한 횟수입니다.
`alarm.countwarningsent`	보내진 경고 수입니다.
`alarm.current`	현재 모니터링되는 값입니다.
`alarm.high`	기록된 값 중에서 가장 높은 값입니다.
`alarm.low`	기록된 값 중에서 가장 낮은 값입니다.
`alarm.timelastset`	현재 값이 마지막으로 설정된 시간입니다.
`alarm.timelastwarning`	경고가 마지막으로 보내진 시간입니다.
`alarm.timereset`	재설정이 마지막으로 수행된 시간입니다.
`alarm.timestatechanged`	경보 상태가 마지막으로 변경된 시간입니다.

표 27–1 counterutil alarm 통계 (계속)

접미어	설명
alarm.warningstate	경보 상태(yes(1) 또는 no(0))입니다.

27.8.3.3 counterutil을 사용한 IMAP, POP 및 HTTP 연결 통계

현재 IMAP, POP 및 HTTP 연결 수, 실패한 로그인 수, 시작 시점부터의 총 연결 수 등에 대한 정보를 얻으려면 `counterutil -o` *CounterObject* `-i 5 -n 10` 명령을 사용합니다. 여기에서 *CounterObject*는 카운터 객체 `popstat`, `imapstat` 또는 `httpstat`를 나타냅니다. `imapstat` 접미어의 의미는 표 27–2에 나와 있습니다. `popstat` 및 `httpstat` 객체는 같은 형식과 구조로 동일한 정보를 제공합니다.

표 27–2 counterutil imapstat 통계

접미어	설명
currentStartTime	현재 IMAP 서버 프로세스의 시작 시간입니다.
lastConnectionTime	새 클라이언트가 마지막으로 수락된 시간입니다.
maxConnections	IMAP 서버에 의해 처리되는 최대 동시 연결 수입니다.
numConnections	현재 IMAP 서버에 의해 서비스되는 총 연결 수입니다.
numCurrentConnections	현재 활성 연결의 수입니다.
numFailedConnections	현재 IMAP 서버에 의해 서비스되는 실패한 연결 수입니다.
numFailedLogins	현재 IMAP 서버에 의해 서비스되는 실패한 로그인 수입니다.
numGoodLogins	현재 IMAP 서버에 의해 서비스되는 성공적인 로그인 수입니다.

27.8.3.4 counterutil을 사용한 디스크 사용 통계

명령: `counterutil -o diskusage` 명령은 다음 정보를 생성합니다.

표 27–3 counterutil diskstat 통계

접미어	설명
diskusage.availSpace	디스크 분할 영역에서 사용할 수 있는 총 공간입니다.
diskusage.lastStatTime	마지막으로 통계를 가져온 시간입니다.
diskusage.mailPartitionPath	메일 분할 영역 경로입니다.
diskusage.percentAvail	사용할 수 있는 디스크 분할 영역 공간의 비율입니다.
diskusage.totalSpace	디스크 분할 영역의 총 공간입니다.

27.8.3.5 서버 응답 통계

명령: `counterutil -o serverresponse` 명령은 다음 정보를 생성합니다. 이 정보는 서버가 실행 중인지 확인하고 서버가 얼마나 빨리 응답하는지 확인하는 데 유용합니다.

표 27–4 `counterutil serverresponse` 통계

접미어	설명
`http.laststattime`	http 서버 응답이 마지막으로 확인된 시간입니다.
`http.responsetime`	http에 대한 응답 시간입니다.
`imap.laststattime`	imap 서버 응답이 마지막으로 확인된 시간입니다.
`imap.responsetime`	imap에 대한 응답 시간입니다.
`pop.laststattime`	pop 서버 응답이 마지막으로 확인된 시간입니다.
`pop.responsetime`	pop에 대한 응답 시간입니다.

27.8.4 로그 파일

Messaging Server는 SMTP, IMAP, POP 및 HTTP에 대한 이벤트 레코드를 기록합니다. Messaging Server 로그 파일을 작성 및 관리하기 위한 정책은 사용자 정의할 수 있습니다.

로깅이 서버 성능에 영향을 주므로 서버에 부담을 주기 전에 로깅을 매우 신중하게 고려해야 합니다. 자세한 내용은 25 장을 참조하십시오.

27.8.5 imsimta 카운터

MTA는 메일 모니터링 MIB, RFC 1566에 기초하여 각 활성 채널에 대한 메시지 트래픽 카운터를 증가시킵니다. 채널 카운터는 전자 메일 시스템의 추세와 상태를 나타내는 데 도움을 줍니다. 채널 카운터는 메시지 트래픽의 정확한 계산을 제공하도록 설계되지는 않았습니다. 정확한 계산을 보려면 25 장에 설명된 대로 MTA 로깅을 확인합니다.

MTA 채널 카운터는 사용 가능한 최소 경량 기법을 사용하여 구현되므로 가능한 한 실제 작업에 미치는 영향이 최소화됩니다. 채널 카운터는 그 이상을 시도하지 않습니다. 즉, 섹션을 매핑하려는 시도가 실패할 경우 정보가 기록되지 않고 섹션의 잠금 중 하나를 거의 즉각적으로 얻을 수 없을 경우 정보가 기록되지 않으며 시스템이 종료할 경우 메모리 내장 섹션에 포함된 정보가 영원히 손실됩니다.

`imsimta counters -show` 명령은 MTA 채널 메시지 통계(아래 참조)를 제공합니다. 시간이 지나면 최소값에 주의하면서 이러한 카운터를 검사해야 합니다. 일부 채널의 경우 최소값은 실제로 음수일 수 있습니다. 음수 값은 카운터가 0이 되었을 때(클러스터 전반의 카운터 데이터베이스 작성 시) 채널에 대해 대기 중인 메시지가 존재했다는 것을 의미합니다. 이러한 메시지가 대기열에서 빠지면 채널의 관련 카운터가 감소하므로

결과적으로 음수 최소값이 생성됩니다. 이러한 카운터의 경우 올바른 "절대값"은 초기화 이후부터 카운터가 지니고 있는 최소값이 아니라 현재 값입니다.

```
Channel          Messages    Recipients    Blocks
-------          --------    ----------    -------
tcp_local
   Received       29379        79714      982252                    (1)
   Stored            61          113       -2004                    (2)
   Delivered      29369        79723      983903 (29369 first time)       (3)
   Submitted      13698        13699       18261                    (4)
   Attempted          0            0           0                    (5)
   Rejected           1           10           0                    (6)
   Failed           104          104        4681                    (7)

   Queue time/count        16425/29440 = 0.56                       (8)
   Queue first time/count  16425/29440 = 0.56                       (9)

   Total In Assocs           297637
   Total Out Assocs           28306
```

1) Received는 tcp_local이라는 채널의 대기열에 포함된 메시지 수입니다. 즉, 다른 채널에 의해 tcp_local의 대기열에 포함된 메시지(mail.log* 파일의 E 레코드)입니다.

2) Stored는 채널 대기열에 저장된 전달할 메시지 수입니다.

3) Delivered는 tcp_local 채널에 의해 처리된(대기열에서 제외된) 메시지 수입니다. 즉, mail.log* 파일의 D 레코드입니다. 대기열에서 제외하는 작업은 성공적인 전달(즉, 다른 채널의 대기열에 포함)에 해당하거나 보낸 사람에게 반송되는 메시지로 인한 작업에 해당할 수 있습니다. 일반적으로 이 값은 Received에서 Stored를 뺀 숫자입니다.

MTA는 또한 처음 시도할 때 대기열에서 제외된 메시지 수를 추적하며 이 수는 괄호로 표시됩니다.

4) Submitted는 tcp_local 채널에 의해 다른 채널의 대기열에 포함된 메시지 수(mail.log 파일의 E 레코드)입니다.

5) Attempted는 대기열에서 빼는 도중에 일시적인 문제를 경험한 메시지 수(즉, mail.log* 파일의 Q 또는 Z 레코드)입니다.

6) Rejected는 시도된 대기열에 포함 작업 중에서 거부된 작업 수(즉, mail.log* 파일의 J 레코드)입니다.

7) Failed는 시도된 대기열에서 빼기 작업 중에서 실패한 작업 수(즉, mail.log* 파일의 R 레코드)입니다.

8) Queue time/count는 전달된 메시지가 대기열에 있는 평균 시간입니다. 여기에는 처음 시도에서 전달된 메시지((9) 참조)와 추가 전달 시도가 필요했던 메시지(대기열에 여유 공간이 생길 때까지 오랜 시간을 기다린 메시지)가 모두 포함됩니다.

9) Queue `first time/count`는 처음 시도에서 전달된 메시지가 대기열에 있는 평균 시간입니다.

제출된 메시지 수가 전달된 메시지 수보다 많을 수 있다는 것을 유의하십시오. 이것은 채널이 대기열에서 제외하는(전달하는) 각 메시지가 대기열에 포함되는(제출되는) 최소한 하나 이상의 새 메시지가 되기 때문에 흔히 발생하는 일입니다. 예를 들어, 메시지에 다른 채널을 통해 도달하는 두 명의 수신자가 있는 경우 대기열에 포함 작업은 두 개가 필요합니다. 또는 메시지가 바운스될 경우 복사본 하나가 보낸 사람에게 되돌아가고 다른 복사본 하나가 포스트마스터에게 보내질 수 있습니다. 이 경우 일반적으로 제출 작업은 두 개가 될 것입니다(두 복사본이 동일한 채널을 통해 도달하지 않을 경우).

`Submitted` 및 `Delivered` 간의 연결은 채널 유형에 따라 바뀌는 것이 더 일반적입니다. 예를 들어, 변환 채널에서는 메시지가 일부 다른 임의 채널에 의해 대기열에 포함되고 나면 변환 채널이 해당 메시지를 처리하여 또 다른 채널의 대기열에 포함시킨 다음 자신의 대기열에서 제외되었다는 것을 메시지에 표시합니다. 각 개별 메시지는 다음 경로를 가집니다.

```
elsewhere -> conversion E record Received
conversion -> elsewhere E record Submitted
conversion              D record Delivered
```

그러나 "전달"이 아니라 두 개의 개별 부분(슬레이브 및 마스터)을 가지는 `tcp_local`과 같은 채널의 경우 `Submitted` 및 `Delivered` 사이에 연결이 없습니다. `Submitted` 카운터는 `tcp_local` 채널의 SMTP 서버 부분과 관련되며 `Delivered` 카운터는 `tcp_local` 채널의 SMTP 클라이언트 부분과 관련됩니다. 이들은 완전히 별개인 두 개의 프로그램이며 각각을 통과하는 메시지가 완전히 다를 수 있습니다.

SMTP **서버로 제출되는 메시지**:

```
tcp_local -> elsewhere E record Submitted
```

SMTP **클라이언트를 통해 다른** *SMTP* **호스트로 보내지는 메시지**:

```
elsewhere -> tcp_local E record Received
tcp_local             D record Delivered
```

채널이 대기열에서 제외하는(전달하는) 메시지는 대기열에 포함되는(제출되는) 최소한 하나 이상의 새 메시지가 됩니다. 예를 들어, 메시지에 다른 채널을 통해 도달하는 두 명의 수신자가 있는 경우 대기열에 포함 작업은 두 개가 필요합니다. 또는 메시지가 바운스될 경우 복사본 하나가 보낸 사람에게 되돌아가고 다른 복사본 하나가 포스트마스터에게 보내질 수 있습니다. 이 경우에는 일반적으로 두 복사본이 동일한 채널을 통해 도달할 것입니다.

27.8.5.1 UNIX 및 NT에서의 구현

성능상의 이유로 인해 MTA를 실행하는 노드는 공유 메모리 섹션(UNIX) 또는 공유 파일 매핑 객체(NT)를 사용하여 채널 카운터 캐시를 메모리에서 유지합니다. 노드의 프로세스가 대기열에서 메시지를 포함시키거나 제외시킬 때 이 메모리 내장 캐시의 카운터가 업데이트됩니다. 채널이 실행될 때 내장 메모리 섹션이 존재하지 않을 경우 이 섹션은 자동으로 만들어집니다. (또한 내장 메모리 섹션이 존재하지 않을 경우 `imta start` 명령은 이 섹션을 만듭니다.)

`imta counters -clear` 또는 `imta qm counters clear` 명령을 사용하면 카운터를 0으로 재설정할 수 있습니다.

27.8.6 imsimta qm 카운터

`imsimta qm counters` 유틸리티는 MTA 채널 대기열 메시지 카운터를 표시합니다. 이 유틸리티를 실행하려면 루트 또는 `mailsrv`여야 합니다. 출력 필드는 846 페이지 “27.8.5 imsimta 카운터”에 설명된 것과 동일합니다. **Sun Java System Messaging Server 6.3 Administration Reference**의 “imsimta counters”를 참조하십시오.

예:

```
# imsimta counters -create
# imsimta qm counters show
Channel               Messages   Recipients Blocks
--------------------- ---------- ---------- ----------
tcp_intranet
   Received             13077      13859     264616
   Stored                  92         91       -362
   Delivered            12985      13768     264978
   Submitted             2594       2594       3641
...
```

MTA를 다시 시작할 때마다 **`# imsimta counters -create`**를 실행해야 합니다.

27.8.7 SNMP를 사용한 MTA 모니터링

Messaging Server는 SNMP(Simple Network Management Protocol)를 통한 시스템 모니터링을 지원합니다. Sun Net Manager 또는 HP OpenView(이 제품에서 제공되지 않음)와 같은 SNMP 클라이언트(경우에 따라 **네트워크 관리자**라고 부름)를 사용하면 Messaging Server의 일정 부분을 모니터할 수 있습니다. 자세한 내용은 부록 A을 참조하십시오.

27.8.8 메일함 할당량 검사를 위한 imquotacheck

imquotacheck 유틸리티를 사용하여 메일함 할당량 사용과 제한을 모니터할 수 있습니다. imquotacheck 유틸리티는 정의된 할당량과 제한을 나열하는 보고서를 생성하며 할당량 사용에 대한 정보를 제공합니다.

예를 들어, 다음 명령은 모든 사용자 할당량 정보를 나열합니다.

```
% imquotacheck
--------------------------------------------------------------------------
Domain red.siroe.com (diskquota = not set msgquota = not set) quota usage
--------------------------------------------------------------------------
diskquota         size(K)    %use    msgquota      msgs    %use    user
# of domains = 1
# of users = 705

no quota          50418             no quota      4392             ajonk
no quota              5             no quota      2                andrt
no quota         355518             no quota      2500             ansri
 ...
```

다음 예에서는 sorook이라는 사용자의 할당량 사용을 보여 줍니다.

```
% imquotacheck -u sorook
--------------------------------------------------------------------------
quota usage for user sorook
--------------------------------------------------------------------------
diskquota      size(K)    %use    msgquota      msgs     %use    user

no quota       1487              no quota      305              sorook
```

27.8.9 msprobe 및 watcher 기능을 사용하여 모니터링

Messaging Server는 여러 시스템 서비스를 모니터하기 위해 watcher 및 msprobe라는 두 가지 프로세스를 제공합니다. watcher는 서버 충돌을 관찰하고 필요에 따라 다시 시작하며 msprobe는 서버 중지(응답하지 않음)를 모니터합니다. 특히 msprobe는 다음을 모니터합니다.

- **서버 응답 시간**. msprobe는 프로토콜 명령을 사용하여 활성화된 서버에 연결하고 응답 시간을 측정합니다. 응답 시간이 경고 알림 임계값을 초과하는 경우 경고 메시지가 서버로 전송되고(852 페이지 “27.8.9.1 경보 메시지” 참조), 서버 응답 시간이 지정한 시간 초과 기간을 넘은 경우에는 서버가 다시 시작됩니다. 서버 응답 시간은 카운터 데이터베이스에 표시되고 기본 로그 파일에 기록됩니다. counterutil은 서버 응답 시간 통계를 표시하는 데 사용됩니다(843 페이지 “27.8.3 counterutil”).

msprobe는 imap, pop, http, cert, job_controller, smtp, lmtp, mmp 및 ens 서버를 모니터합니다. smtp 또는 lmtp가 응답하지 않을 때는 디스패처가 다시 시작됩니다. ens는 자동으로 다시 시작되지 않습니다.

- **디스크 사용**. msprobe는 모든 메시지 저장소 분할 영역의 디스크 사용을 확인합니다. 특히 메시지 저장소 mboxlist 데이터베이스 디렉토리와 MTA 대기열 디렉토리를 검사합니다. 디스크 사용이 구성된 임계값을 초과할 때는 경고 메시지가 전송됩니다. 디스크 크기와 사용은 카운터 데이터베이스에 표시되고 기본 로그 파일에 기록됩니다. 관리자는 counterutil 유틸리티(843 페이지 "27.8.3 counterutil" 참조)를 사용하여 디스크 사용 통계를 표시할 수 있습니다.
- **메시지 저장소 mboxlist 데이터베이스 로그 파일 누적**. 로그 파일 누적은 mboxlist 데이터베이스 오류를 나타냅니다. msprobe는 활성 로그 파일의 개수를 세고 그 수가 임계값보다 클 경우 msprobe에서 default 로그 파일에 중요 오류 메시지를 기록하여 관리자에게 서버를 다시 시작하도록 알립니다. autorestart가 활성화되어 있으면(local.autorestart를 yes로 설정) 저장소 데몬이 다시 시작됩니다.

watcher 및 msprobe는 표 27–5에 있는 configutil 옵션으로 제어됩니다. 자세한 내용은 107 페이지 "4.5 실패했거나 응답이 없는 서비스의 자동 재시작"을 참조하십시오.

표 27–5 msprobe 및 watcher configutil 옵션

옵션	설명
local.autorestart	서버 자동 재시작 활성화. 실패하거나 중지된 서비스를 자동으로 다시 시작합니다. 기본값: 아니요
local.autorestart.timeout	재시도 시간 초과 오류. 지정된 시간 내에 서버가 세 번 이상 실패하면 시스템은 서버 재시작 시도를 중지합니다. 이 값(초)은 msprobe 간격(local.schedule.msprobe)보다 길게 설정해야 합니다. 기본값: 600초
local.probe.service.timeout	다시 시작하기 전 특정 서버에 대한 시간 초과. *service*는 imap, pop, http, cert, job_controller, smtp, lmtp, mmp 또는 ens가 될 수 있습니다. 기본값: service.readtimeout 사용
local.probe.service.warningthreshold	경고 메시지가 default 로그 파일에 기록되기 전 특정 서버가 응답하지 않는 시간(초). *service*는 imap, pop, http, cert, job_controller, smtp, lmtp, mmp 또는 ens가 될 수 있습니다. 기본값: local.probe.warningthreshold
local.probe.warningthreshold	경고 메시지가 default 로그 파일에 기록되기 전 서버가 응답하지 않는 시간(초). 기본값: 5초
local.queuedir	대기열 크기가 alarm.diskavail.msgalarmthreshold에서 정의한 임계값을 초과하는 경우 검사할 MTA 대기열 디렉토리. 기본값: 없음

표 27–5 msprobe 및 watcher configutil 옵션 (계속)

옵션	설명
service.readtimeout	서버를 다시 시작하기 전 해당 서버가 응답하지 않는 시간. local.schedule.msprobe를 참조하십시오. 기본값: 10초
local.schedule.msprobe	msprobe에서 일정을 실행합니다. crontab 스타일 일정 문자열(표 20–10 참조). 이 값은 기본적으로 자동으로 설정됩니다. 110 페이지 “4.6.2 미리 정의된 자동 작업”을 참조하십시오. 비활성화하려면 local.schedule.msprobe.enable을 NO로 설정합니다.
local.watcher.enable	서비스 실패를 모니터하는 watcher를 활성화합니다. IMAP, POP, HTTP, 작업 제어기, 디스패처, 메시지 저장소(stored), imsched 및 MMP. LMTP/SMTP 서버는 디스패처가 모니터하며 LMTP/SMTP 클라이언트는 job_controller가 모니터합니다. 특정 실패에 대해 오류 메시지를 기본 로그 파일에 기록합니다. 기본값: on

27.8.9.1 경보 메시지

msprobe는 지정된 조건을 경고하도록 포스트마스터에게 전자 메일 형식으로 경보를 보낼 수 있습니다(840 페이지 “27.6.1.2 imapd, popd 및 httpd 모니터” 참조). 다음은 일정한 임계값을 초과할 때 보내지는 샘플 전자 메일 경보입니다.

```
Subject:    ALARM: server response time in seconds of “ldap_siroe.com_389” is 10
Date:    Tue, 17 Jul 2001 16:37:08 -0700 (PDT)
From:    postmaster@siroe.com
To:     postmaster@siroe.com

Server instance: /opt/SUNWmsgsr
Alarmid: serverresponse
Instance: ldap_siroe_europa.com_389
Description: server response time in seconds
Current measured value (17/Jul/2001:16:37:08 -0700): 10
Lowest recorded value: 0
Highest recorded value: 10
Monitoring interval: 600 seconds
Alarm condition is when over threshold of 10
Number of times over threshold: 1
```

msprobe가 디스크 및 서버 성능을 모니터하는 빈도와 경보를 보내는 상황을 지정할 수 있습니다. 이렇게 하려면 configutil 명령을 사용하여 경보 매개 변수를 설정합니다. 표 27–6은 기본 설정과 함께 유용한 경보 매개 변수를 보여 줍니다. **Sun Java System Messaging Server 6.3 Administration Reference**의 “configutil Parameters”를 참조하십시오.

표 27–6 유용한 경보 메시지 `configutil` 매개 변수

매개 변수	설명(괄호 안의 값이 기본값임)
larm.msgalarmnoticehost	(localhost) 경고 메시지를 보낼 시스템입니다.
alarm.msgalarmnoticeport	(25) 경보 메시지를 보낼 때 연결할 SMTP 포트입니다.
alarm.msgalarmnoticercpt	(Postmaster@localhost) 경보 알림을 받는 사람입니다.
alarm.msgalarmnoticesender	(Postmaster@localhost) 경보를 보낸 사람의 주소입니다.
alarm.diskavail.msgalarmdescription	(percentage mail partition diskspace available.)디스크 사용 경보의 설명 필드에 사용되는 문자열.
alarm.diskavail.msgalarmstatinterval	(3600) 디스크 가용성 검사의 간격(초)입니다. 디스크 사용 검사를 사용하지 않으려면 0으로 설정합니다.
alarm.diskavail.msgalarmthreshold	(10) 디스크 공간 가용성 비율로서 이 비율 아래로 내려가면 경보가 보내집니다.
alarm.diskavail.msgalarmthresholddirection	(-1) 경보가 디스크 공간 가용성이 임계값보다 작을 때 발생하는지(-1) 아니면 임계값보다 클 때 발생하는지(1) 여부를 지정합니다.
alarm.diskavail.msgalarmwarninginterval	(24). 디스크 가용성 경보가 반복되는 간격(시간)입니다.
alarm.serverresponse.msgalarmdescription	(server response time in seconds.)서버 응답 경보의 설명 필드에 사용되는 문자열.
alarm.serverresponse.msgalarmstatinterval	(600) 서버 응답 검사의 간격(초)입니다. 서버 응답 검사를 사용하지 않으려면 0으로 설정합니다.
alarm.serverresponse.msgalarmthreshold	(10) 서버 응답 시간(초)이 이 값을 초과할 경우 경보가 발생합니다.
alarm.serverresponse.msgalarmthresholddirection	(1) 경보가 서버 응답 시간이 임계값보다 클 때 발생하는지(1) 아니면 임계값보다 작을 때 발생하는지(-1) 여부를 지정합니다.
alarm.serverresponse.msgalarmwarninginterval	(24) 서버 응답 경보가 반복되는 간격(시간)입니다.

◆ ◆ ◆ 부 록 A

SNMP 지원

Messaging Server는 SNMP(Simple Network Management Protocol)를 통한 시스템 모니터링을 지원합니다. Sun Net Manager 또는 HP OpenView(이 제품에서 제공되지 않음)와 같은 SNMP 클라이언트(경우에 따라 **네트워크 관리자**라고 부름)를 사용하면 Messaging Server의 일정 부분을 모니터할 수 있습니다. Messaging Server 모니터링에 대한 자세한 내용은 27 장을 참조하십시오.

이 장에서는 Messaging Server에서 SNMP 지원을 사용하는 방법에 대해 설명합니다. 또한, SNMP에 의해 제공되는 정보 유형에 대한 개요를 제공합니다. 하지만 SNMP 클라이언트에서 이 정보를 보는 방법에 대해서는 설명하지 않습니다. SNMP 클라이언트를 사용하여 SNMP 기반 정보를 보는 방법에 대한 자세한 내용은 SNMP 클라이언트 설명서를 참조하십시오. 또한, 이 문서에는 Messaging Server SNMP 구현에서 사용 가능한 일부 데이터에 대해 설명되어 있지만 전체 MIB 정보는 RFC 2788 (`http://www.faqs.org/rfcs/rfc2788.html`) 및 RFC 2789 (`http://www.faqs.org/rfcs/rfc2788.html`)에서 이용할 수 있습니다.

이 장은 다음 내용으로 구성되어 있습니다.

A.1 SNMP 구현

Messaging Server는 Network Services Monitoring MIB(RFC 2788)와 Mail Monitoring MIB(RFC 2789)의 두 표준화된 MIB를 구현합니다. Network Services Monitoring MIB는 POP, IMAP, HTTP 및 SMTP 서버와 같은 네트워크 서비스 모니터링을 위해 제공됩니다. Mail Monitoring MIB는 MTA 모니터링을 위해 제공됩니다. Mail Monitoring MIB를사용하면 각 MTA 채널의 활성 상태와 비활성 상태를 모두 모니터할 수 있습니다. 활성 정보에는 주로 현재 대기열에 포함된 메시지와 열린 네트워크 연결(예: 대기열에

있는 메시지 개수, 열린 네트워크 연결의 소스 IP 주소)에 대한 정보가 있고, 비활성 정보에는 누적 합계(예: 처리된 총 메시지 수, 총 인바운드 연결)가 제공됩니다.

주 – 전체 Messaging Server SNMP 모니터링 정보 목록은 RFC 2788 및 RFC 2789를 참조하십시오.

SNMP는 Solaris 및 Red Hat Linux를 실행하는 플랫폼에서 지원됩니다. Solaris 9 운영 체제의 Messaging Server는 SEA(Solstice Enterprise Agents)를 사용합니다. Solaris 10 운영 체제부터 Messaging Server는 오픈 소스 Net-SNMP 모니터링 프레임워크를 지원하며 Solaris 9 OS SEA(Solstice Enterprise Agents) 기술을 레거시(단종 조치) 상태로 분류합니다. 또한 Net-SNMP는 Linux 플랫폼에서 널리 사용됩니다. Messaging Server는 Solaris 10 이상과 Linux 플랫폼에서 Net-SNMP 기반 SNMP 하위 에이전트를 사용합니다.

Net-SNMP 프레임워크를 채택하여 Messaging Server의 SNMP 하위 에이전트는 다음과 같은 새로운 기능을 제공합니다.

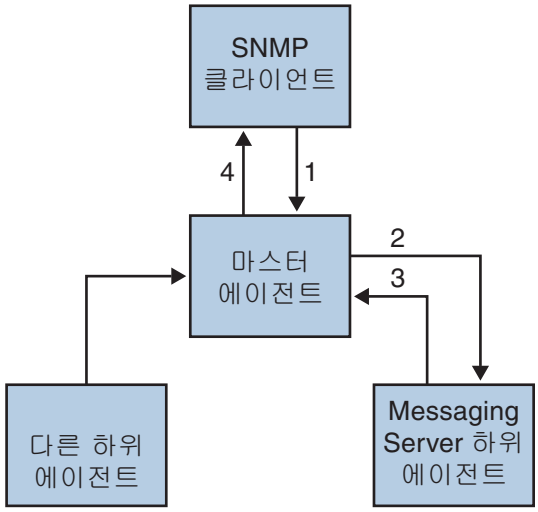
- SNMP 버전 2c 및 3을 지원합니다. 이 지원은 Net-SNMP 프레임워크에서 제공합니다. 이전 SNMP 기술인 Solstice Enterprise Agents는 SNMP 버전 1만 지원했습니다. 향상된 보안 기능과 액세스 제어는 이 두 SNMP 버전의 주요 이점입니다.
- 하위 에이전트를 "독립형" SNMP 에이전트로 실행되도록 구성할 수 있습니다. 그러면 사이트에서 동일한 시스템에서 실행 중인 다양한 SNMP 에이전트를 격리시킬 수 있습니다.
- 동일한 시스템에서 실행 중인 여러 Messaging Server "인스턴스"를 동시에 모니터링할 수 있습니다. 이 지원은 위 두 항목 중 하나 또는 SNMP 버전 3 "컨텍스트 이름"을 사용하여 제공됩니다. 이 지원을 사용하면 페일오버 클러스터에서 Messaging Server의 SNMP 모니터링이 가능합니다.

Messaging Server SNMP 지원 제한은 다음과 같습니다.

- 각 호스트 컴퓨터에서는 Solaris 9 OS의 SNMP를 통해 Messaging Server 인스턴스를 하나씩만 모니터링할 수 있습니다.
- SNMP 지원은 모니터링 전용입니다. SNMP 관리는 지원되지 않습니다.
- SNMP 트랩이 구현되지 않습니다. RFC 2788은 트랩을 사용하지 않는 유사한 기능을 제공합니다.

A.1.1 Messaging Server에서의 SNMP 작업

Messaging Server SNMP 프로세스는 시작할 때 플랫폼의 원시 SNMP 마스터 에이전트를 사용하여 자체 등록되는 SNMP 하위 에이전트입니다. 클라이언트의 SNMP 요청은 마스터 에이전트로 전달됩니다. 마스터 에이전트는 Messaging Server가 대상인 모든 요청을 Messaging Server 하위 에이전트 프로세스에 전달합니다. Messaging Server 하위 에이전트 프로세스에서는 해당 요청을 처리하고 마스터 에이전트를 통해 클라이언트에게 응답을 릴레이합니다. 이 프로세스는 그림 A–1에 표시되어 있습니다.



1. SNMP 클라이언트는 정보 요청을 마스터 에이전트에 전달합니다.
2. 마스터 에이전트는 Messaging Server 하위 에이전트에 요청을 전달합니다.
3. Messaging Server 하위 에이전트는 마스터 에이전트에게 응답합니다.
4. 마스터 에이전트는 SNMP 클라이언트에게 응답합니다.

그림 A–1 SNMP 정보 흐름

A.2 Solaris 9에서 Messaging Server에 대한 SNMP 지원 구성

SNMP 모니터링으로 인한 오버헤드는 매우 작지만 Messaging Server는 SNMP 지원을 비활성화한 상태로 제공됩니다. SNMP 지원을 사용하려면 다음 명령을 실행합니다.

```
# su user-id-for-ims
# configutil -o local.snmp.enable -v 1
# start-msg snmp
```

SNMP를 활성화하면 매개 변수를 지정하지 않더라도 `start-msg` 명령이 다른 Messaging Server 프로세스와 함께 SNMP 하위 에이전트 프로세스를 자동으로 시작합니다.

Messaging Server SNMP 하위 에이전트를 작동하려면 Solaris 원시 SNMP 마스터 에이전트를 실행해야 합니다. Solaris 원시 SNMP 마스터 에이전트는 일반적으로 Solaris 부트 절차의 일부로 시작되는 `snmpdx` 데몬입니다.

SNMP 하위 에이전트는 수신할 UDP 포트를 자동으로 선택합니다. 필요한 경우 다음 명령을 사용하여 하위 에이전트에 고정 UDP 포트를 지정할 수 있습니다.

```
# configutil -o local.snmp.port -v port-number
```

나중에 포트 번호로 값 0을 지정하여 이 설정을 취소할 수 있습니다. 기본 설정인 값 0은 Messaging Server에서 하위 에이전트가 사용 가능한 UDP 포트를 자동으로 선택할 수 있도록 합니다.

`/etc/snmp/conf` 디렉토리에는 두 개의 SNMP 하위 에이전트 구성 파일인 SNMP 액세스 제어 정보를 포함하는 `ims.acl`과 SNMP MIB OID 등록 정보를 포함하는 `ims.reg`가 있습니다.

일반적으로 이러한 파일은 편집할 필요가 없습니다. Messaging Server에서 제공하는 MIB는 읽기 전용이며 `ims.reg` 파일에서 포트 번호를 지정할 필요가 없습니다. 포트 번호를 지정하는 경우 `configutil` 유틸리티를 사용하여 다른 포트를 설정할 때까지는 해당 포트가 적용됩니다. 그럴 경우 하위 에이전트에서는 `configutil`을 사용하여 설정한 포트 번호를 사용합니다. 파일을 편집하는 경우 변경 내용을 적용하려면 SNMP 하위 에이전트를 중지하고 다시 시작해야 합니다.

```
# stop-msg snmp
# start-msg snmp
```

주 – Messaging Server에서 SNMP 지원을 활성화한 상태에서 Solaris 10 OS에서 생성한 SNMP 쿼리는 기본 포트 16161에 연결되어야 합니다. 예를 들어, 오픈 소스 SNMP 도구 *snmpwalk*를 사용하여 Messaging Server에 대한 네트워크/메일 통계를 쿼리하는 경우 `-p 16161` 옵션을 사용합니다.

A.3 Solaris 10 OS에 대한 SNMP 지원 구성

기본적으로 SNMP 모니터링은 Messaging Server에서 비활성화됩니다. 기본 Messaging Server 구성에서 제공되는 서비스 수를 최소화하려면 이 기본값을 선택합니다. 이 기본값을 SNMP 모니터링을 사용하여 성능 감소가 발생한다는 의미로 해석하지 마십시오. 실제로 Messaging Server의 SNMP 지원은 매우 적은 자원을 사용하며 메시징 서버에 가장 적은 영향을 줍니다. 결론적으로 Messaging Server의 SNMP 지원을 사용하려면 구성 단계를 한 번 수행해야 합니다. 또한 Messaging Server와 같은 하위 에이전트를 실행하려면 플랫폼 Net-SNMP 마스터 에이전트 `snmpd`의 기본 구성을 변경해야 합니다. 이러한 변경에 대해서는 다음 절에서 설명합니다.

A.3.1 Net-SNMP 구성

Messaging Server의 Net-SNMP 기반 SNMP 하위 에이전트는 AgentX 프로토콜을 사용하여 플랫폼의 SNMP 마스터 에이전트(RFC 2741)와 통신합니다. AgentX 프로토콜을 사용하도록 Net-SNMP 마스터 에이전트 `snmpd`를 구성해야 합니다. 그렇게 하려면 플랫폼의 `snmpd.conf` 파일에 다음과 같은 행이 있어야 합니다.

```
master agentx
```

이 행이 없으면 해당 행을 추가한 다음 snmpd 데몬을 다시 시작합니다. 데몬에 SIGHUP 신호를 보내는 것으로는 충분하지 않습니다. snmpd 데몬을 다시 시작한 후 snmpd가 AgentX 통신을 위해 만드는 UNIX 도메인 소켓을 찾습니다. Solaris 및 Linux 시스템에서 이 소켓은 기본적으로 /var/agentx/master라는 특수 파일로 표시됩니다. 해당 위치와 이름은 snmpd.con 파일을 통해 변경할 수 있습니다.

Solaris 10 OS snmpd 구성은 아래와 같이 표시됩니다.

```
%cp /etc/sma/snmp/snmpd.conf /etc/sma/snmp/snmpd.conf.save
% cat >> /etc/sma/snmp/snmpd.conf
# Messaging Server's subagent requires the AgentX protocol
master agentx
^D
% cat >> /etc/sma/snmp/snmpd.conf
% ls -al /var/agentx/
srwxrwxrwx 1 root root 0 Aug 9 13:58 /var/agentx/master
```

또한 Red Hat Enterprise Linux AS 3 시스템에서 기본 snmpd.conf 파일은 "공개" SNMP 커뮤니티에서 볼 수 있는 정보를 제한합니다. 따라서 해당 제한을 제거하거나 Messaging Server의 하위 에이전트에서 제공하는 MIB를 포함하도록 확장해야 합니다. 최초 테스트를 위해 확장할 것을 권장합니다. 그렇게 하려면 아래 표시된 대로 OID 하위 트리 mib-2.27 및 mib-2.28를 "systemview"라는 보기에 포함시킵니다. 실제 배포를 위해 각 사이트에서 전체 보안 정책을 고려해야 합니다. SNMP 하위 에이전트에서 제공하는 정보는 "읽기 전용"입니다.

```
% cp /etc/snmp/snmpd.conf /etc/snmp/snmpd.conf.save
% cat >>/etc/snmp/snmpd.conf
# Messaging Server's subagent requires the AgentX protocol
master agentx
# Messaging Server's subagent exports mib-2.27 and .28
# Add the mib-2.27 and .28 OID subtrees to the systemview
view systemview included .1.3.6.1.2.1.27
view systemview included .1.3.6.1.2.1.28
^D
% /sbin/service snmpd restart
% ls -al /var/agentx/master
srwxr-xr-x 1 root root 0 Aug 8 21:20 /var/agentx/master
```

SNMP v3 컨텍스트 이름을 사용하여 동일한 호스트 컴퓨터에서 동시에 실행 중인 다양한 Messaging Server 인스턴스의 MIB를 구별할 경우 SNMP v3 쿼리에 사용할 하나 이상의 SNMP v3 사용자 아이디와 비밀번호를 구성해야 합니다.

A.3.2 Messaging Server 하위 에이전트 구성

Messaging Server SNMP 하위 에이전트의 기본 작업을 위해서는 하위 에이전트를 활성화한 다음 수동 시작 명령을 한 번만 실행하면 됩니다. 그 이후에는 Messaging Server가 시작되거나 중지될 때마다 하위 에이전트도 함께 시작되거나 중지됩니다. Solaris와 Linux 모두에서 이 구성을 적용하는 데 필요한 명령은 다음과 같습니다.

```
% configutil -o local.snmp.enable -v 1
% start-msg snmp
```

이 명령을 실행한 후 명령줄에서 `snmpwalk` 명령을 사용하여 하위 에이전트를 테스트할 수 있습니다. Solaris 및 Linux에 해당하는 예제 아래의 스크린 샷을 참조하십시오. `rfc2248.txt` 및 `rfc2249.txt` 파일은 Network Services 및 MTA MIB의 복사본입니다. Solaris 시스템에서는 이러한 파일이 `/etc/sma/snmp/mibs/` 디렉토리에 `NETWORK-SERVICES-MIB.txt` 및 `MTA-MIB.txt` 이름으로 존재할 수도 있습니다. 이러한 파일을 `snmpwalk` 도구에 제공할 필요는 없지만, 그렇게 하면 `snmpwalk`에서 숫자 객체 식별자(OID) 대신 각 MIB 변수의 이름을 인쇄할 수 있습니다.

Solaris에서의 기본 테스트:

```
% D=/opt/SUNWmsgsr/examples/mibs /usr/sfw/bin/snmpwalk -v 1 -c public \
    -m +$D/rfc2248.txt:$D/rfc2249.txt 127.0.0.1 mib-2.27
NETWORK-SERVICES-MIB::applName.1 = STRING: /opt/SUNWmsgsr MTA on mail.siroe.com
...
% D=/opt/SUNWmsgsr/examples/mibs /usr/sfw/bin/snmpwalk -v 1 -c public \
     -m +$D/rfc2248.txt:$D/rfc2249.txt 127.0.0.1 mib-2.28
MTA-MIB::mtaReceivedMessages.1 = Counter32: 1452
MTA-MIB::mtaStoredMessages.1 = Gauge32: 21
...
```

Linux에서의 기본 테스트:

```
% export D=/opt/sun/messaging/examples/mibs
% /usr/bin/snmpwalk -v 1 -c public \
     -m +$D/rfc2248.txt:$D/rfc2249.txt 127.0.0.1 mib-2.27
NETWORK-SERVICES-MIB::applName.1 = STRING: /opt/sun/messaging MTA on mail.siroe.com
...
% /usr/bin/snmpwalk -v 1 -c public \
     -m +$D/rfc2248.txt:$D/rfc2249.txt 127.0.0.1 mib-2.28
MTA-MIB::mtaReceivedMessages.1 = Counter32: 21278
MTA-MIB::mtaStoredMessages.1 = Gauge32: 7
...
```

A.3.3 독립형 SNMP 에이전트로 실행

Messaging Server의 SNMP 하위 에이전트를 독립형 SNMP 에이전트로 실행되도록 구성하기 전에 SNMP 요청을 수신하는 데 사용할 이더넷 인터페이스와 UDP 포트를 결정해야 합니다. 기본적으로 SNMP 하위 에이전트는 UDP 포트 161을 사용하여 사용 가능한 모든 이더넷 인터페이스를 수신합니다. 대부분의 경우 플랫폼의 SNMP 마스터 에이전트인 `snmpd`를 방해하지 않도록 포트 번호를 변경할 수 있습니다. HA 페일오버와 같은 일부 환경에서는 이더넷 인터페이스를 사용 가능한 모든 인터페이스(`INADDR_ANY`)에서 IP 주소로 식별되는 특정 인터페이스로 변경할 수도 있습니다. 이더넷 인터페이스 및 UDP 포트의 두 개념은 `local.snmp.listenaddr` 및 `local.snmp.port` 옵션을 통해 제어됩니다.

이더넷 인터페이스와 UPD 포트를 선택하면 `local.snmp.standalone` 옵션의 값이 설정되고 하위 에이전트가 다시 시작됩니다. 그러면 하위 에이전트가 `snmpd` 및 모든 하위 에이전트에 독립적으로 SNMP 에이전트로 작동합니다.

예를 들어, IP 주소 10.53.1.37를 사용하여 이더넷 인터페이스의 UDP 포트 9161을 수신하는 독립형 에이전트로 실행하려면 다음과 같은 명령을 실행합니다.

독립형 에이전트로 실행하도록 구성:

```
% configutil -o local.snmp.port -v 9161
% configutil -o local.snmp.listenaddr -v 10.53.1.37
% configutil -o local.snmp.standalone -v 1
% stop-msg snmp
% start-msg snmp
% snmpwalk -v 1 -c public 10.53.1.37:9161 .
SNMPv2-SMI::mib-2.27.1.1.2.1 = STRING: "/opt/SUNWmsgsr MTA on mail.siroe.com"
...
```

A.3.4 여러 Messaging Server 인스턴스 모니터링

여기서는 동일한 호스트 컴퓨터에서 실행 중인 여러 Messaging Server 인스턴스를 모니터링하는 두 가지 기술에 대해 설명합니다. 독립형 모드에서 하위 에이전트를 실행하는 첫 번째 기술은 Messaging Server의 개별 인스턴스가 호스트 컴퓨터 사이에서 동적으로 이동할 수 있는 고가용성(HA) 페일오버 구성에 적합합니다. SNMP v3 컨텍스트 이름을 사용하는 두 번째 기술은 Messaging Server의 여러 인스턴스가 단일 시스템으로 한정되고 SNMP 모니터링 소프트웨어에 의해 폴링되는 IP 주소 수를 제한하려는 경우(예: 모니터링 소프트웨어 라이센스 비용이 IP 주소 단위로 부과되는 경우)에 몇 가지 제한된 이점을 제공합니다. 이 두 번째 기술은 HA 페일오버 설정에서도 사용될 수 있지만 독립형 모드 기술과 같은 수의 IP 주소만큼만 폴링해야 합니다.

A.3.5 고가용성 페일오버를 위해 독립형 에이전트 사용

Messaging Server의 SNMP 모니터링이 요구되는 고가용성 페일오버 설정에서는 861 페이지 "A.3.3 독립형 SNMP 에이전트로 실행"에서 설명한 것처럼 Messaging Server의 SNMP 하위 에이전트를 독립형 에이전트로 실행하는 것이 좋습니다. 하위 에이전트가 독립형 모드로 실행되는 경우 Messaging Server의 각 HA 인스턴스는 `local.snmp.listenaddr` 옵션이 해당 인스턴스의 페일오버 IP 주소 값으로 설정되어야 합니다. 관리를 간소화하기 위해 각 인스턴스는 동일한 UDP 포트를 사용하지만 각 물리적 클러스터 호스트에서 실행 중인 `snmpd` 데몬에서 사용되는 포트와 구별되어야 합니다. 일반적으로 이러한 데몬은 UDP 포트 161을 사용하므로 `local.snmp.port` 옵션을 통해 다른 포트 번호를 명시적으로 지정합니다.

Messaging Server의 SNMP 지원을 여기에 권장된 대로 구성한 경우 모니터링 스테이션은 인스턴스가 실행 중인 물리적 클러스터 호스트에 상관없이 페일오버 IP 주소 또는 호스트 이름을 통해 Messaging Server의 각 인스턴스를 모니터링할 수 있습니다. 또한 Messaging Server의 독립형 SNMP 에이전트는 각각 인스턴스의 고유한 페일오버 IP 주소로 식별되는 해당 가상 이더넷 인터페이스만 수신하기 때문에 서로 충돌하지 않습니다. 이러한 가상 이더넷 인터페이스는 HA 페일오버 프레임워크에서 자동으로 생성됩니다. UDP 포트를 신중하게 선택했기 때문에 에이전트가 클러스터 내의 시스템에서 실행 중인 `snmpd` 데몬과 충돌하지 않습니다.

A.3.6 SNMP v3 컨텍스트 이름을 통해 여러 인스턴스 구별

861 페이지 "A.3.3 독립형 SNMP 에이전트로 실행"에서 설명한 것처럼 Messaging Server의 SNMP 지원을 독립형 모드로 사용할 때의 단점은 없지만, 일부 사이트에서는 동일한 시스템에서 동시에 실행 중인 여러 Messaging Server 인스턴스를 모니터링하는 기능을 유지하면서 많은 기존의 하위 에이전트 모드를 사용하는 것이 더 좋을 수도 있습니다. 예를 들어, 라이센스 모델에 의해 폴링될 수 있는 IP 주소 수가 제한되는 SNMP 모니터링 시스템이 있습니다. 이렇게 하려면 `local.snmp.standalone`을 0으로 설정한 상태에서 Messaging Server의 SNMP 하위 에이전트를 계속 실행합니다. 또는 `local.snmp.enablecontextname` 옵션을 0이 아닌 값으로 지정하여 각 Messaging Server 인스턴스가 고유한 SNMP v3 컨텍스트 이름을 사용하도록 구성합니다. `service.defaultdomain` 값과 다른 컨텍스트 이름을 지정하려면 `local.snmp.contextname` 옵션을 사용하여 원하는 이름을 설정합니다. Messaging Server의 각 SNMP 하위 에이전트 인스턴스가 다시 시작되면 SNMP v3 쿼리를 통해 해당 컨텍스트 이름을 포함하는 인스턴스를 모니터링할 수 있습니다. 동일한 시스템에서 실행되는 두 Messaging Server 인스턴스의 MIB는 인스턴스의 SNMP v3 컨텍스트 이름을 통해 구별되므로 MIB 객체 식별자(OID) 충돌이 발생하지 않습니다.

A.3.7 Messaging Server의 Net-SNMP 기반 SNMP 하위 에이전트 옵션

다음은 Messaging Server의 Net-SNMP 기반 SNMP 하위 에이전트에만 적용되는 옵션입니다. 이 하위 에이전트는 Solaris 10 이상을 실행하는 Solaris 플랫폼과 Linux 플랫폼에서 사용됩니다. 아래 설명된 옵션은 Solaris 9 이하의 운영 체제를 실행하는 Solaris 플랫폼용으로 제공된 레거시 SNMP 하위 에이전트에는 적용되지 않습니다.

아래 설명하는 옵션은 `configutil` 옵션입니다. 따라서 다음과 같은 형식의 명령을 사용하여 해당 값을 검사합니다.

```
% configutil -o option-name
```

여기서 *option-name*은 값을 표시할 옵션의 이름입니다. 옵션의 값을 설정하거나 변경하려면 다음과 같은 형식의 명령을 사용합니다.

```
% configutil -o option-name -v option-value
```

여기서 *option-value*는 설정할 값입니다. 이 옵션에 대한 변경 사항을 적용하려면 다시 시작해야 합니다.

```
% stop-msg snmp
% start-msg snmp
```

다음은 각 옵션에 대한 설명과 해당 옵션의 기본값입니다.

표 A–1 SNMP 하위 에이전트 옵션

옵션(기본값)	설명
local.snmp.enable (0)	Messaging Server SNMP 하위 에이전트는 Messaging Server에서 정상적인 시작 및 종료 절차를 수행하는 중에 하위 에이전트를 자동으로 중지하고 시작하도록 이 옵션의 값을 1 또는 `true`로 설정한 경우에만 실행됩니다. 기본적으로 이 옵션은 0으로 설정되어 하위 에이전트 작업을 비활성화합니다. 하위 에이전트를 활성화하려면 861 페이지 "A.3.3 독립형 SNMP 에이전트로 실행"에 설명한 것처럼 플랫폼의 마스터 에이전트가 적절하게 구성되어 있어야 합니다.

표 A–1 SNMP 하위 에이전트 옵션 (계속)

local.snmp.standalone (0)	Messaging Server의 SNMP 지원은 일반적으로 SNMP 하위 에이전트로 실행되며, 플랫폼의 SNMP 마스터 에이전트 snmpd를 통해 SNMP 요청을 받습니다. 이 작업 모드는 기본값이며 이 옵션의 값을 0 또는 false로 지정하여 선택합니다. 그러나 861 페이지 "A.3.3 독립형 SNMP 에이전트로 실행"에서 설명한 것처럼 하위 에이전트는 "독립형" 모드로 실행되어 snmpd와 독립적으로 SNMP 에이전트로 작동할 수 있습니다. 독립 모드에서 실행할 경우 하위 에이전트(현재의 SNMP 에이전트)는 각각 local.snmp.listenaddr 및 local.snmp.port 옵션에 지정된 이더넷 인터페이스 및 UDP 포트에서 SNMP 요청을 직접 수신합니다. 독립 모드로 실행하려면 이 옵션의 값을 1 또는 TRUE로 지정합니다. 독립 모드로 실행하면 시스템에서 실행 중인 다른 SNMP 마스터나 하위 에이전트를 방해하지 않습니다.
local.snmp.listenaddr (INADDR_ANY)	독립 모드로 실행할 때 SNMP 요청을 수신할 이더넷 인터페이스의 호스트 이름 또는 IP 주소입니다. 기본적으로 사용 가능한 모든 인터페이스가 수신됩니다. 이 작업은 값 INADDR_ANY를 지정하는 것과 일치합니다. 인터페이스에 연결된 IP 주소나 호스트 이름을 지정하여 선택할 수 있는 인터페이스도 있습니다. 인터페이스는 물리적 인터페이스 또는 가상 인터페이스입니다. 이 옵션은 local.snmp.standalone이 0 또는 FALSE로 설정된 경우 무시됩니다.
local.snmp.cachettl (30)	캐시된 모니터링 데이터의 수명(TTL, 초)입니다. 이 옵션은 하위 에이전트가 동일한 모니터링 데이터를 보고하는 시간을 제어합니다. 이 시간이 경과하면 Messaging Server에서 가져온 새 정보로 해당 데이터를 새로 고칩니다. 메시지 루프 정보를 제외한 모든 데이터는 기본적으로 30초 이하로 캐싱됩니다. 루프 정보는 .HELD 파일을 스캔하여 결정되며 10분마다 한 번만 업데이트됩니다. 그 이유는 모든 디스크 내장 메시지 대기열을 스캔하는 자원 비용 때문입니다. 하위 에이전트는 모니터링 데이터를 지속적으로 업데이트하지 않습니다. 모니터링 데이터는 SNMP 요청을 수신하고 캐시된 데이터가 만료된 경우(즉, TTL이 지난 경우)에만 업데이트됩니다. TTL을 30초로 설정하고 SNMP 요청이 5분마다 생성되는 경우 하위 에이전트는 SNMP 요청이 있을 때마다 Messaging Server에서 새 데이터를 가져옵니다. 즉, Messaging Server의 데이터를 5분마다 한 번만 가져옵니다. 반면에 SNMP 요청이 10초마다 생성되는 경우에는 하위 에이전트는 29초 경과된 캐시된 데이터가 있는 일부 요청에 응답하고 Messaging Server는 30초마다 한 번만 폴링됩니다.

표 A–1 SNMP 하위 에이전트 옵션 *(계속)*

local.snmp.servertimeout (5)	하위 에이전트는 각 서비스에 대한 TCP 연결을 실제로 열고 프로토콜 교환을 실행하여 각 모니터된 서비스의 작업 상태를 결정합니다. 이 시간 초과 값(초)은 하위 에이전트가 프로토콜 교환 중에 각 단계에 대한 응답을 대기하는 시간을 제어합니다. 기본적으로 5초의 시간 초과 값이 사용됩니다.
local.snmp.directoryscan (1)	이 옵션을 사용하면 하위 에이전트가 디스크 내장 메시지 대기열에서 `.HELD` 메시지 파일 및 가장 오래된 메시지 파일을 스캔할지 여부를 제어할 수 있습니다. 이 정보는 `mtaGroupLoopsDetected`, `mtaGroupOldestMessageStored` 및 `mtaGroupOldestMessageId` MIB 변수와 일치합니다. 이 옵션 값을 1 또는 `true`로 설정하면 이 정보 캐시가 유지되고 필요에 따라 업데이트됩니다. 이러한 특정 MIB 변수와 관련 없는 수천 개의 대기열 메시지가 있는 사이트에서는 이 옵션 값을 0 또는 `false`로 설정하는 것이 좋습니다.
local.snmp.enablecontextname (0)	하위 에이전트는 SNMP v3 **컨텍스트 이름**으로 MIB를 등록할 수 있습니다. 그럴 경우 SNMP 요청에서 컨텍스트 이름을 지정하는 SNMP v3 클라이언트만 MIB를 요청할 수 있습니다. 여러 독립 하위 에이전트에서 컨텍스트 이름을 사용하여 동일한 OID 트리 아래(즉, 동일한 SNMP 마스터 에이전트 아래)에 Network Services 및 MTA MIB를 등록할 수 있습니다. 자세한 내용은 861 페이지 “A.3.4 여러 Messaging Server 인스턴스 모니터링”을 참조하십시오. SNMP v3 컨텍스트 이름 사용을 활성화하려면 이 옵션의 값을 1 또는 `true`로 지정합니다. 그러면 하위 에이전트는 기본적으로 컨텍스트 이름으로 `service.defaultdomain` 옵션 값을 사용합니다. 컨텍스트 이름으로 다른 값을 사용하려면 `local.snmp.contextname` 옵션을 사용합니다.
local.snmp.contextname (service.defaultdomain)	`local.snmp.enablecontextname`을 사용하여 SNMP v3 컨텍스트 이름 사용을 활성화한 경우 이 옵션을 사용하여 하위 에이전트에서 MIB에 대해 사용하는 컨텍스트 이름을 명시적으로 설정할 수 있습니다. 이 옵션에 대해 제공되는 값은 문자열 값이고 SNMP v3 컨텍스트 이름으로 사용하기에 적합해야 합니다. 이 옵션은 `local.snmp.enablecontextname` 값이 0 또는 `false`인 경우 무시됩니다.

A.4 SNMP 클라이언트로부터 모니터링

RFC 2788 (`http://www.faqs.org/rfcs/rfc2788.html`) 및 RFC 2789 (`http://www.faqs.org/rfcs/rfc2788.html`)의 기본 OID는 다음과 같습니다.

mib-2.27 = 1.3.6.1.2.1.27

mib-2.28 = 1.3.6.1.2.1.28

이러한 두 OID에서 SNMP 클라이언트를 가리키고 "공개" SNMP 커뮤니티로 액세스합니다.

MIB 복사본을 SNMP 클라이언트에 로드하려는 경우 MIB의 ASCII 복사본은 *msg-svr-base*/lib/config-templates 디렉토리에 rfc2788.mib 및 rfc2789.mib 파일 이름으로 존재합니다. 이러한 MIB을 SNMP 클라이언트 소프트웨어에 로드하는 것과 관련된 지침은 SNMP 클라이언트 소프트웨어 설명서를 참조하십시오. 일부 이전 SNMP 클라이언트에서는 이러한 MIB에 사용되는 SnmpAdminString 데이터 유형을 인식하지 못할 수 있습니다. 그럴 경우 동일한 디렉토리에 있는 rfc2248.mib 및 rfc2249.mib 파일을 사용합니다.

A.5 Messaging Server의 SNMP 정보

이 절에서는 SNMP를 통해 제공되는 Messaging Server 정보를 요약합니다. 이 절은 다음 내용으로 구성되어 있습니다.

자세한 내용은 RFC 2788 (http://www.faqs.org/rfcs/rfc2788.html) 및 RFC 2789 (http://www.faqs.org/rfcs/rfc2788.html)의 개별 MIB 테이블을 참조하십시오. RFC/MIB 용어로 메시징 서비스(MTA, HTTP 등)는 **응용 프로그램**(appl), Messaging Server 네트워크 연결은 **연결**(assoc), MTA 채널은 *MTA* **그룹**(mtaGroups)입니다.

두개 이상의 Messaging Server 인스턴스를 동시에 모니터할 수 있는 플랫폼의 경우 applTable에 여러 MTA 및 서버 집합이 있고 다른 테이블에 여러 MTA가 있을 수 있습니다.

주 – MIB에 보고되는 누적 값(예: 전달된 총 메시지 수, 총 IMAP 연결 수 등)은 재부트 후에 다시 0으로 설정됩니다.

사이트별로 임계값과 중요 모니터링 값이 다릅니다. 정상적인 SNMP 클라이언트에서는 추세 분석을 수행하여 기록된 추세와 편차가 발생할 경우 경고를 보냅니다.

A.5.1 applTable

applTable은 서버 정보를 제공합니다. MTA 행 하나와 각각의 서버(WebMail HTTP, IMAP, POP, SMTP 및 SMTP Submit, 활성화된 경우)에 대한 추가적인 행이 제공되는

1차원적 테이블입니다. 이 테이블에서는 버전 정보, 가동 시간, 현재 작업 상태(실행, 중단, 정체), 현재 연결 수, 총 누적 연결 수 및 기타 관련 데이터를 제공합니다.

다음은 applTable(mib-2.27.1.1)의 데이터 예입니다.

applTable:

```
applName.1  = mailsrv-1  MTA on mailsrv-1.west.sesta.com          (1)
applVersion.1 = 5.1
applUptime.1 = 7322                                (2)
applOperStatus.1 = up                              (3)
applLastChange.1 = 7422                            (2)
applInboundAssociations.1 =                        (5)
applOutboundAssociations.1 =                       (2)
applAccumulatedInboundAssociations.1 = 873
applAccumulatedOutboundAssociations.1 = 234
applLastInboundActivity.1 = 1054822                (2)
applLastOutboundActivity.1 = 1054222               (2)
applRejectedInboundAssociations.1 = 0              (4)
applFailedOutboundAssociations.1 = 17
applDescription.1 = Sun Java System Messaging Server 6.1
applName.2     1 = mailsrv-1 HTTP WebMail svr. mailsrv-1.sesta.com      (1)
...
applName.3 = mailsrv-1 IMAP server on mailsrv-1.west.sesta.com
...
applName.4 = mailsrv-1 POP server on mailsrv-1.west.sesta.com
...
applName.5 = mailsrv-1 SMTP server on mailsrv-1.west.sesta.com
...
applName.6 = mailsrv-1 SMTP Submit server on mailsrv-1.west.sesta.com
...
```

주:

1. 응용 프로그램(.appl*) 접미어(.1, .2 등)는 행 번호 applIndex입니다. applIndex에서 값 1은 MTA, 값 2는 HTTP 서버 등을 나타냅니다. 따라서 이 예에서 테이블의 첫 번째 행에는 MTA에 관한 데이터가 제공되고 두 번째 행에는 POP 서버에 관한 데이터가 제공됩니다.

   등호 뒤에 있는 이름은 모니터할 Messaging Server 인스턴스의 이름입니다. 이 예에서 인스턴스 이름은 mailsrv-1입니다.

2. 이러한 값은 SNMP 타임스탬프 값이고 이벤트 시간의 sysUpTime 값입니다. 즉, sysUpTime은 SNMP 마스터 에이전트가 시작된 후의 시간(1/100초)입니다.

3. HTTP, IMAP, POP, SMTP 및 SMTP Submit 서버의 작업 상태는 구성된 TCP 포트를 통해 해당 서버에 실제로 연결한 다음 해당 프로토콜(예: HTTP의 경우 HEAD 요청 및 응답, SMTP의 경우 HELO 명령 및 응답 등)을 통해 간단한 작업을 수행하여 확인합니다. 이 연결 시도에서 각 서버의 상태(up(1), down(2) 또는 congested(4))가 결정됩니다.

이러한 검사는 서버에 일반 인바운드 연결로 표시되고 각 서버에 대한 applAccumulatedInboundAssociations MIB 변수 값에 포함됩니다.

MTA의 경우 작업 상태는 작업 제어기의 작업 상태입니다. MTA가 실행 상태로 표시되면 작업 제어기가 실행 중인 것입니다. MTA가 중단 상태로 표시되면 작업 제어기가 중단된 것입니다. 이 MTA 작업 상태는 MTA의 서비스 디스패처 상태와는 별개입니다. MTA의 작업 상태는 실행 또는 중단 값으로만 나타납니다. 작업 제어기에 "congested"(정체)라는 개념은 있지만 MTA 상태에 나타나지는 않습니다.

4. HTTP, IMAP 및 POP 서버의 경우 applRejectedInboundAssociations MIB 변수는 거부된 인바운드 연결 시도 횟수가 아니라 실패한 로그인 시도 횟수를 나타냅니다.

A.5.1.1 applTable 사용

나열된 각 응용 프로그램에 대한 서버 상태(applOperStatus) 모니터링은 각 서버 모니터링의 핵심입니다.

applLastInboundActivity가 마지막 MTA 인바운드 작업을 표시한 이후 많은 시간이 경과하면 일부 연결이 끊어져 연결되지 않을 수 있습니다. applOperStatus=2(중단)인 경우 모니터되는 서비스가 중단 상태입니다. applOperStatus=1(실행)인 경우 다른 곳에 문제가 있을 수 있습니다.

A.5.2 assocTable

이 테이블은 MTA에 네트워크 연결 정보를 제공합니다. 이 테이블은 각 활성 네트워크 연결에 대한 정보를 제공하는 2차원 테이블입니다. 다른 서버에 대한 연결 정보는 제공되지 않습니다.

다음은 applTable(mib-2.27.2.1)의 데이터 예입니다.

assocTable:

```
    assocRemoteApplication.1.1  = 129.146.198.167          (1)
    assocApplicationProtocol.1.1 = applTCPProtoID.25       (2)
    assocApplicationType.1.1 = peerinitiator(3)            (3)
    assocDuration.1.1 = 400                                (4)
...
```

주:

.x.y 접미어(1.1)에서 x는 응용 프로그램 색인인 applIndex이며 보고되는 applTable의 응용 프로그램을 나타냅니다. 이 경우에는 MTA입니다. y는 보고되는 응용 프로그램에 대한 각 연결을 나열합니다.

1. 원격 SMTP 클라이언트의 소스 IP 주소입니다.

2. 네트워크 연결에 사용되는 프로토콜을 나타내는 OID입니다. aplTCPProtoID는 TCP 프로토콜을 나타냅니다. .n 접미어는 사용 중인 TCP 포트를 나타내고 .25는 TCP 포트 25를 통해 통신하는 프로토콜인 SMTP를 나타냅니다.
3. 원격 SMTP 클라이언트가 사용자 에이전트(UA)인지 다른 MTA인지 알 수 없습니다. 마찬가지로 하위 에이전트는 항상 peer-initiator를 보고하는 반면 ua-initiator는 절대 보고하지 않습니다.
4. 이 값은 SNMP TimeInterval이며 시간 단위(1/100초)를 사용합니다. 이 예에서 연결은 4초 동안 열립니다.

A.5.2.1 assocTable 사용

이 테이블은 활성 문제를 진단하는 데 사용됩니다. 예를 들어, 갑자기 200,000 인바운드 연결이 발생하는 경우 이 테이블을 사용하여 해당 연결의 시작 위치를 알 수 있습니다.

A.5.3 mtaTable

applTable의 각 MTA마다 하나의 행을 갖는 1차원 테이블입니다. 각 행에는 mtaGroupTable의 선택 변수에 대한 해당 MTA 내의 모든 채널(그룹이라고 함)의 합계가 제공됩니다.

다음은 applTable(mib-2.28.1.1)의 데이터 예입니다.

mtaTable:

```
mtaReceivedMessages.1 = 172778
mtaStoredMessages.1 = 19
mtaTransmittedMessages.1 = 172815
mtaReceivedVolume.1 = 3817744
mtaStoredVolume.1 = 34
mtaTransmittedVolume.1 = 3791155
mtaReceivedRecipients.1 = 190055
mtaStoredRecipients.1 = 21
mtaTransmittedRecipients.1 = 3791134
mtaSuccessfulConvertedMessages.1 = 0      (1)
mtaFailedConvertedMessages.1 = 0
mtaLoopsDetected.1 = 0                    (2)
```

주:

.x 접미어(.1)는 applTable에서 이 응용 프로그램에 대한 행 번호를 제공합니다. 이 예에서 .1은 이 데이터가 applTable에 있는 첫 번째 응용 프로그램에 대한 데이터임을 나타냅니다. 따라서, 이 데이터는 MTA에 관한 데이터입니다.

1. 변환 채널의 경우 0이 아닌 값만 가집니다.

2. 현재 MTA의 메시지 대기열에 저장된 .HELD 메시지 파일 수를 계산합니다.

A.5.3.1 mtaTable 사용

mtaLoopsDetected가 0이 아니면 루핑 메일 문제가 있는 것입니다. 이럴 경우 MTA 대기열에서 .HELD 파일을 찾아 진단하여 문제를 해결합니다.

시스템에서 변환 채널을 사용하여 바이러스 스캔을 수행하고 감염된 메시지를 거부하는 경우 mtaSuccessfulConvertedMessages는 다른 변환 오류와 함께 감염된 메시지 수를 제공합니다.

A.5.4 mtaGroupTable

이 2차원 테이블은 applTable의 각 MTA에 대한 채널 정보를 제공합니다. 이 정보는 저장(대기열에 포함된) 및 전달된 메일 메시지의 수와 같은 데이터를 포함합니다. 각 채널에 대한 저장된 메시지 수 mtaGroupStoredMessages 모니터링은 값이 비정상적으로 커서 메일이 대기열에 백업되고 있는 경우에 중요합니다.

다음은 mtaGroupTable(mib-2.28.2.1)의 데이터 예입니다.

mtaGroupTable:

```
mtaGroupName.1.1 = tcp_intranet                    1
        ...
mtaGroupName.1.2 = ims-ms
        ...
mtaGroupName.1.3 = tcp_local
    mtaGroupDescription.1.3 = mailsrv-1 MTA tcp_local channel
    mtaGroupReceivedMessages.1.3 = 12154
    mtaGroupRejectedMessages.1.3 = 0
    mtaGroupStoredMessages.1.3 = 2
    mtaGroupTransmittedMessages.1.3 = 12148
    mtaGroupReceivedVolume.1.3 = 622135
    mtaGroupStoredVolume.1.3 = 7
    mtaGroupTransmittedVolume.1.3 = 619853
    mtaGroupReceivedRecipients.1.3 = 33087
    mtaGroupStoredRecipients.1.3 = 2
    mtaGroupTransmittedRecipients.1.3 = 32817
    mtaGroupOldestMessageStored.1.3 = 1103
    mtaGroupInboundAssociations.1.3 = 5
    mtaGroupOutboundAssociations.1.3 = 2
    mtaGroupAccumulatedInboundAssociations.1.3 = 150262
    mtaGroupAccumulatedOutboundAssociations.1.3 = 10970
    mtaGroupLastInboundActivity.1.3 = 1054822
    mtaGroupLastOutboundActivity.1.3 = 1054222
    mtaGroupRejectedInboundAssociations.1.3 = 0
```

```
mtaGroupFailedOutboundAssociations.1.3 = 0
mtaGroupInboundRejectionReason.1.3 =
mtaGroupOutboundConnectFailureReason.1.3 =
mtaGroupScheduledRetry.1.3 = 0
mtaGroupMailProtocol.1.3 = applTCPProtoID.25
mtaGroupSuccessfulConvertedMessages.1.3 = 03          2
mtaGroupFailedConvertedMessages.1.3 = 0
mtaGroupCreationTime.1.3 = 0
mtaGroupHierarchy.1.3 = 0
mtaGroupOldestMessageId.1.3 = <01IFBV8AT8HYB4T6UA@red.iplanet.com>
mtaGroupLoopsDetected.1.3 = 0                         3
mtaGroupLastOutboundAssociationAttempt.1.3 = 1054222
```

주:

.x.y 접미어(예: 1.1, 1.2. 1.3)에서 x는 응용 프로그램 색인인 applIndex이며 보고되는 applTable의 응용 프로그램을 나타냅니다. 이 경우에는 MTA입니다. y는 MTA의 각 채널을 나열합니다. 이 열거 색인 mtaGroupIndex는 mtaGroupAssociationTable 및 mtaGroupErrorTable 테이블에도 사용됩니다.

1. 보고되는 채널의 이름이며 이는 tcp_intranet 채널입니다.
2. 변환 채널의 경우 0이 아닌 값만 가집니다.
3. 현재 이 채널의 메시지 대기열에 저장된 .HELD 메시지 파일 수를 계산합니다.

A.5.4.1 mtaGroupTable 사용

Rejected 및 *Failed*에 대한 추세 분석은 잠정적인 채널 문제를 확인하는 데 유용할 수 있습니다.

mtaGroupStoredVolume의 비율이 mtaGroupStoredMessages로 갑자기 증가하면 대용량 정크 메일이 해당 대기열 주위로 바운스되는 것을 의미할 수 있습니다.

mtaGroupStoredMessages로 갑자기 증가하는 것은 요청하지 않은 대량 전자 메일을 받거나 어떤 이유로 인해 전달이 실패함을 나타낼 수 있습니다.

mtaGroupOldestMessageStored 값이 전달할 수 없는 메시지 알림(notices 채널 키워드)에 사용된 값보다 몇 배 이상 큰 경우 이는 바운스 처리를 통해 처리할 수 없는 메시지를 나타낼 수 있습니다. 바운스는 mtaGroupOldestMessageStored > (최대 기간 + 24시간)를 테스트로 사용할 수 있도록 밤에 수행됩니다.

mtaGroupLoopsDetected가 0보다 큰 경우 메일 루프가 감지됩니다.

A.5.5 mtaGroupAssociationTable

해당 항목이 assocTable에 대한 색인이 되는 3차원 테이블입니다. applTable의 각 MTA에는 2차원 하위 테이블이 있습니다. 이 2차원 하위 테이블에는 해당 MTA의 각 채널에 대한 행이 하나씩 있습니다. 각 채널에는 해당 채널이 현재 진행 중인 각 활성 네트워크 연결에 대한 항목이 있습니다. 해당 항목 값은 assocTable(항목 값에 의해 색인화되고 MTA의 applIndex 색인이 참조된)에 대한 색인입니다. assocTable에 표시된 항목은 채널에 보관된 네트워크 연결입니다.

즉, mtaGroupAssociationTable 테이블은 assocTable에 표시된 네트워크 연결을 mtaGroupTable의 해당 채널에 연결합니다.

다음은 mtaGroupAssociationTable(mib-2.28.3.1)의 데이터 예입니다.

mtaGroupAssociationTable:

```
mtaGroupAssociationIndex.1.3.1 = 1      1
mtaGroupAssociationIndex.1.3.2 = 2
mtaGroupAssociationIndex.1.3.3 = 3
mtaGroupAssociationIndex.1.3.4 = 4
mtaGroupAssociationIndex.1.3.5 = 5
mtaGroupAssociationIndex.1.3.6 = 6
mtaGroupAssociationIndex.1.3.7 = 7
```

주:

.x.y.z 접미어에서 x는 응용 프로그램 색인인 applIndex이며 보고되는 applTable의 응용 프로그램을 나타냅니다. 이 경우에는 MTA입니다. y는 보고되는 mtaGroupTable의 채널을 나타냅니다. 이 예에서 3은 tcp_local 채널을 나타냅니다. z는 채널로 또는 채널에서 열린 연결을 나열합니다.

1. 여기서 값은 assocTable에 대한 색인입니다. 특히 x와 이 값은 각각 assocTable의 applIndex 및 assocIndex 색인 값이 됩니다. 또는 assocTable(applIndex 무시)의 첫 번째 행은 tcp_local 채널에 의해 제어되는 네트워크 연결을 설명합니다.

A.5.6 mtaGroupErrorTable

메시지 전달을 시도하는 동안 각 MTA의 각 채널에서 발생된 임시 및 영구 오류의 개수를 제공하는 다른 3차원 테이블입니다. 색인 값이 4000000인 항목은 임시 오류이고 색인 값이 5000000인 항목은 영구 오류입니다. 임시 오류가 발생하면 나중에 전달 시도할 수 있도록 대기열에 메시지를 다시 포함하고, 영구 오류가 발생하면 메시지는 거부되거나 전달할 수 없는 메시지로 반환됩니다.

다음은 mtaGroupErrorTable(mib-2.28.5.1)의 데이터 예입니다.

mtaGroupErrorTable:

```
mtaGroupInboundErrorCount.1.1.4000000     1 = 0
mtaGroupInboundErrorCount.1.1.5000000 = 0
mtaGroupInternalErrorCount.1.1.4000000 = 0
mtaGroupInternalErrorCount.1.1.5000000 = 0
mtaGroupOutboundErrorCount.1.1.4000000 = 0
mtaGroupOutboundErrorCount.1.1.5000000 = 0

mtaGroupInboundErrorCount.1.2.4000000     1 = 0
...

mtaGroupInboundErrorCount.1.3.4000000     1 = 0
...
```

주:

1. .x.y.z 접미어에서 x는 응용 프로그램 색인인 applIndex이며 보고되는 applTable의 응용 프로그램을 나타냅니다. 이 경우에는 MTA입니다. y는 보고되는 mtaGroupTable의 채널을 나타냅니다. 이 예에서 1은 tcp_intranet 채널을 지정하고 2는 ims-ms 채널을, 3은 tcp_local 채널을 지정합니다. 마지막으로 z는 4000000 또는 5000000이며 각각 해당 채널에 대한 메시지 전달 시도 중에 발생한 임시 오류 및 영구 오류의 개수를 나타냅니다.

A.5.6.1 mtaGroupErrorTable 사용

오류 개수가 크게 증가하면 비정상적인 전달 문제를 나타낼 수 있습니다. 예를 들어, tcp_channel이 크게 증가하면 DNS 또는 네트워크 문제를 나타낼 수 있습니다. ims_ms 채널이 크게 증가할 경우 메시지를 메시지 저장소로 전달하는 데 문제(예: 분할 영역이 꽉 참, stored 문제 등)가 있을 수 있습니다.

◆ ◆ ◆ 부 록 B

Messaging Server에서 Event Notification Service 관리

이 부록에서는 Messaging Server에서 Event Notification Service Publisher(ENS Publisher)를 사용하고 ENS(Event Notification Service)를 관리할 때 필요한 사항에 대해 설명합니다.

이 장/부록은 다음 내용으로 구성되어 있습니다.

ENS 및 ENS API에 대한 자세한 내용은 **Sun Java Communications Suite 5 Event Notification Service Guide**를 참조하십시오.

B.1 Messaging Server에서 ENS Publisher 로드

ENS(Event Notification Service)는 기본 게시 및 가입 서비스입니다. ENS는 Sun Java System 응용 프로그램이 특정 유형의 **이벤트**를 수집하기 위한 중앙 지점으로 사용하는 디스패처 역할을 합니다. 이벤트는 하나 이상의 자원 등록 정보 값에 대한 변경 사항입니다. 이러한 유형의 이벤트가 발생할 때 그 사실을 알고자 하는 모든 응용 프로그램은 이벤트 순서를 식별하고 알림과 가입을 일치시키는 ENS에 등록합니다.

ENS와 iBiff(Messaging Server용 ENS Publisher)는 Messaging Server부터 함께 제공됩니다. ENS는 기본적으로 사용되지만 iBIFF는 로드되지 않습니다. 875 페이지 “B.1 Messaging Server에서 ENS Publisher 로드”를 참조하십시오.

Messaging Server에서 알림에 가입하려면 Messaging Server 호스트에서 `libibiff` 파일을 로드한 다음 Messaging Server를 중지했다가 다시 시작해야 합니다.

▼ Messaging Server에서 ENS Publisher 로드

명령줄에서 다음 단계를 수행합니다. 이 단계에서 Messaging Server 설치 디렉토리의 위치는 *msg-svr-base*이며 Messaging Server 사용자는 *inetuser*입니다. 이 변수의 일반적인 값은 각각 `/opt/SUNWmsgsr` 및 `mailsrv`입니다.

1. **`mailsrv`로 `configutil` 유틸리티를 실행하여 `libibiff` 파일을 로드합니다.**

   `cd` *msg-svr-base*
   `./configutil -o "local.store.notifyplugin" -v "`*msg-svr-base***`/lib/libibiff"`**

2. **`root`로 Messaging Server를 중지했다가 다시 시작합니다.**

   `cd` *msg-svr-base* **`/sbin`**

   `./stop-msg`

   `./start-msg`

3. **이제 ENS를 통해 알림을 받을 수 있습니다.** 876 페이지 "B.2 샘플 Event Notification Service 프로그램 실행"을 참조하십시오.

B.2 샘플 Event Notification Service 프로그램 실행

Messaging Server에는 알림 수신 방법을 익힐 수 있는 샘플 프로그램이 포함되어 있습니다. 이 샘플 프로그램은 *msg-svr-base*`/examples` 디렉토리에 있습니다.

▼ 샘플 ENS 프로그램 실행

1. *msg-svr-base*`/examples` **디렉토리로 이동합니다.**

2. **`Makefile.sample` 파일을 사용하여 C 컴파일러로 `apub`와 `asub` 예를 컴파일합니다.**
   msg-svr-base`/examples` **디렉토리가 포함되도록 라이브러리 검색 경로를 설정합니다.**

3. **프로그램 컴파일이 끝나면 별도 창에서 다음과 같이 실행할 수 있습니다.**

   `apub localhost 7997`

   `asub localhost 7997`

   `apub` 창에 입력하는 내용은 모두 `asub` 창에 나타납니다. 또한 기본 설정을 사용할 경우 모든 iBiff 알림이 `asub` 창에 나타납니다.

4. **iBiff에서 발행한 알림을 받으려면 `asub.c`와 유사한 프로그램을 작성합니다.**
   샘플 프로그램 및 ENS용 프로그램을 직접 작성하는 것에 대한 자세한 내용은 **Sun Java Communications Suite 5 Event Notification Service Guide**를 참조하십시오.

주 – 라이브러리 검색 경로에 *msg-svr-base*/lib 디렉토리가 포함되도록 설정한 다음에는 더 이상 Directory Server를 중지하고 시작할 수 없습니다. 해결 방법은 라이브러리 검색 경로에서 해당 항목을 제거하는 것입니다.

B.3 Event Notification Service 관리

ENS의 관리 작업에는 서비스의 시작과 중지, ENS용 iBiff Publisher의 동작을 제어하기 위한 구성 매개 변수 변경이 포함됩니다.

B.3.1 ENS 시작 및 중지

start-msg ens 및 stop-message ens 명령을 사용하여 ENS 서버를 시작 및 중지할 수 있습니다. 이 명령을 실행하려면 root가 되어야 합니다.

- ENS를 시작하려면 다음을 수행합니다.

  msg-svr-base **/sbin/start-msg ens**
- ENS를 중지하려면 다음을 수행합니다.

  msg-svr-base **/sbin/stop-msg ens**

▼ ENS 시작 및 중지

- **ENS를 시작하려면 다음을 수행합니다.**

  msg-svr-base **/sbin/start-msg ens**
  - ENS를 중지하려면 다음을 수행합니다.

    *msg-svr-base***/sbin/stop-msg ens**

B.3.2 Event Notification Service 구성 매개 변수

몇 가지의 구성 매개 변수가 iBiff의 동작을 제어합니다. configutil 유틸리티 프로그램을 사용하여 이러한 매개 변수를 설정합니다.

표 B–1 iBiff 구성 매개 변수

매개 변수	설명
local.store.notifyplugin.maxHeaderSize	알림과 함께 전송될 헤더의 최대 크기(바이트)를 지정합니다. 기본값은 0바이트입니다.

표 B–1 iBiff구성 매개 변수 *(계속)*

매개 변수	설명
`local.store.notifyplugin.maxBodySize`	알림과 함께 전송될 본문의 최대 크기(바이트)를 지정합니다. 기본값은 0바이트입니다.
`local.store.notifyplugin.`*eventType* `.enable`	특정 이벤트 유형이 알림을 생성하는 경우에 지정합니다. 유효한 값은 1(활성화)과 0(비활성화)입니다. 기본값은 1이며, `local.store.notifyplugin.ReadMsg.enable`을 0으로 설정하면 `ReadMsg` 알림이 비활성화됩니다.
`local.store.notifyplugin.ensHost`	ENS 서버의 호스트 이름을 지정합니다. 기본값은 `127.0.0.1`입니다.
`local.store.notifyplugin.ensPort`	ENS 서버의 TCP 포트를 지정합니다. 기본값은 `7997`입니다.
`local.store.notifyplugin.ensEventKey`	ENS 알림에 사용할 이벤트 키를 지정합니다. 기본값은 `enp://127.0.0.1/store`입니다. 이벤트 키의 호스트 이름 부분은 ENS 호스트를 결정하는 데 사용되지 않으며, ENS가 사용하는 고유 식별자일 뿐입니다. 이 키는 가입자가 이 키에 일치하는 이벤트에 대한 알림을 받기 위해 가입해야 하는 키입니다.

◆ ◆ ◆ 부록 C

SMS(Short Message Service)

이 장에서는 Sun™ ONE Messaging Server에서 SMS(Short Message Service)를 구현하는 방법에 대해 설명합니다. 이 장은 다음 내용으로 구성되어 있습니다.

C.1 소개

Sun Java System Messaging Server는 SMS(Short Message Service)를 사용하여 ETM(Email-To-Mobile) 및 MTE(Mobile-To-Email) 메시징을 구현합니다. SMS는 단방향(ETM만 해당) 또는 양방향(ETM 및 MTE 모두)이 되도록 구성할 수 있습니다. 단방향 서비스만 사용하려면 SMS 채널을 추가 및 구성해야 합니다. 양방향 서비스를 사용하려면 SMS 채널을 추가 및 구성하는 것 외에도 SMS 게이트웨이 서버를 구성해야 합니다.

단방향 및 양방향 SMS의 경우 생성된 SMS 메시지는 SMPP(Short Message Peer to Peer) 프로토콜을 사용하여 SMSC(Short Message Service Center)로 전송됩니다. 특히 SMSC는 TCP/IP를 지원하는 V3.4 이상의 SMPP 서버를 제공해야 합니다.

그림 C–1은 단방향 및 양방향 SMS의 논리적 메시지 흐름을 보여 줍니다.

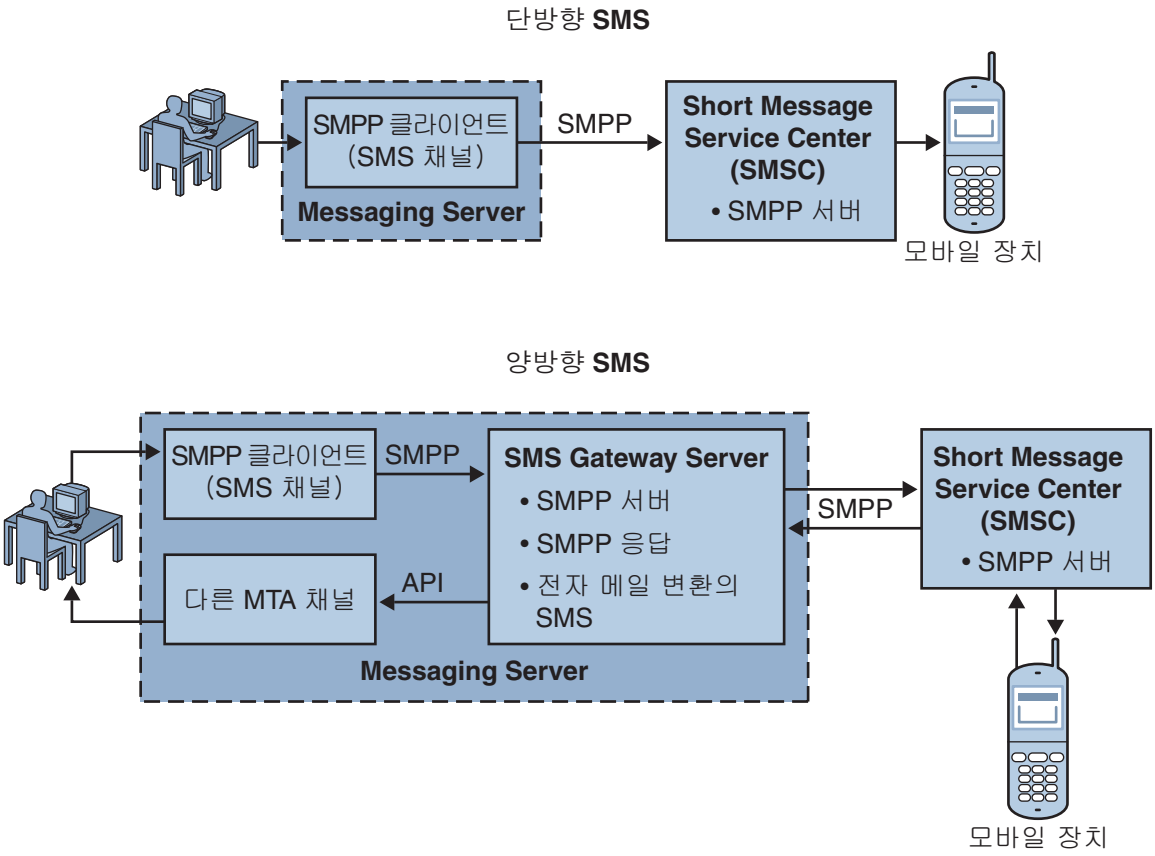


그림 C–1 단방향 및 양방향 SMS의 논리적 흐름

C.1.1 단방향 SMS

단방향 서비스를 사용하기 위해 Messaging Server는 원격 SMSC와 통신하는 SMPP 클라이언트(MTA SMS 채널)를 구현합니다. SMS 채널은 883 페이지 "C.2.2 전자 메일에서 SMS로의 변환 프로세스"에 설명된 대로 대기열에 포함된 전자 메일을 SMS 메시지로 변환합니다.

SMS 채널은 이러한 방식으로 작동하여(SMPP) ESME(External Short Message Entity)의 기능을 수행합니다.

C.1.1.1 양방향 SMS

양방향 SMS를 사용하여 메일 서버에서 원격 장치로 전자 메일을 보낼 뿐만 아니라 원격 장치로부터 그리고 원격 장치가 보낸 메일 응답을 받을 수 있습니다.

양방향 SMS 서비스를 사용하려면 이전 항목에 설명된 MTA SMS 채널(SMPP 클라이언트)과 SMS 게이트웨이 서버가 모두 필요합니다. Sun Java System Messaging Server는 일반 설치 프로세스의 일부로 SMS 게이트웨이 서버를 설치하며 사용자는 이 서버를 구성해야 합니다. SMS 게이트웨이 서버는 다음 두 가지 기능을 수행합니다.

- SMPP 중계

  SMS 게이트웨이 서버는 MTA SMS 채널과 SMSC 간의 투명한 SMPP 클라이언트 역할을 수행합니다. SMS 게이트웨이 서버는 또한 중계 역할을 수행하면서 중계된 메시지에 대한 고유한 SMS 주소를 생성하고 원격 SMSC에 의해 반환된 메시지 아이디를 저장하여 나중에 SMS 알림 메시지와 상호 작용시킵니다.

- SMPP 서버

  SMS 게이트웨이 서버는 모바일에서 전송된 SMS 메시지, 이전 전자 메일에 대한 응답 및 SMS 알림을 받기 위한 SMTP 서버의 역할을 수행합니다. SMS 게이트웨이 서버는 변환 프로세스를 정의하는 프로필을 사용하여 SMS 메시지에서 대상 전자 메일 주소를 추출합니다. 프로필은 또한 이전에 보낸 ETM(Email-To-Mobile) 메시지에 응답하여 원격 SMSC가 반환한 알림 메시지를 처리하는 방법을 설명합니다.

주 – Sun Java System Messaging Server는 Windows 플랫폼에서 양방향 SMS를 지원하지 않습니다.

C.1.2 요구 사항

이 설명서는 Logica CMG의 SMPP 사양과 사용자 SMSC용 SMPP 설명서를 읽은 것으로 가정합니다.

SMS를 구현하려면 다음 요구 사항을 충족해야 합니다.

- Sun Java System Messaging Server 6 이상(단방향 SMS는 iPlanet Messaging Server 5.2에서도 구현됩니다.)
- SMSC는 TCP/IP를 통해 SMPP V3.4 이상을 지원해야 하며 Messaging Server를 실행하는 호스트와 SMSC 간에 TCP/IP 연결이 존재해야 합니다.

SMS 게이트웨이 서버에 대한 저장소 계획 정보는 950 페이지 “C.6 SMS 게이트웨이 서버 저장소 요구 사항”을 참조하십시오.

C.2 SMS 채널 작동 원리

SMS 채널은 대기 중인 전자 메일 메시지를 SMS 메시지로 변환한 다음 전달을 위해 SMSC에 전송하는 멀티스레드 채널입니다.

이 절은 다음과 같은 채널 작업 항목으로 구성되어 있습니다.

C.2.1 전자 메일을 채널로 전송

896 페이지 "C.3 SMS 채널 구성"으로 SMS 채널이 구성되면 하나 이상의 호스트 이름이 해당 채널과 연관됩니다. 여기서는 설명을 위해 호스트 이름 `sms.siroe.com`을 채널과 연관된 호스트 이름으로 가정합니다. 이 경우 전자 메일은 다음 형식의 주소를 가진 채널로 전송됩니다.

```
local-part@sms.siroe.com
```

여기서 `local-part`는 SMS 대상 주소(예: 휴대폰 번호, 호출기 아이디 등) 또는 다음 형식의 속성 값 쌍 목록입니다.

```
/attribute1=value1/attribute2=value2/.../@sms.siroe.com
```

인식되는 속성 이름과 그 사용법은 표 C–1에 나와 있습니다. 이러한 속성은 일부 채널 옵션에 대한 수신자별 제어를 허용합니다.

표 C–1 SMS 속성

속성 이름	속성 값 및 사용법
`ID`	SMS 메시지를 전송할 SMS 대상 주소(예: 휴대폰 번호, 호출기 아이디 등)입니다. 이 속성과 관련 값은 반드시 존재해야 합니다.
`FROM`	SMS 소스 주소입니다. 옵션 `USE_HEADER_FROM=0`일 경우 무시됩니다.
`FROM_NPI`	지정된 `NPI` 값을 사용합니다. 옵션 `USE_HEADER_FROM=0`일 경우 무시됩니다.
`FROM_TON`	지정된 `TON` 값을 사용합니다. 옵션 `USE_HEADER_FROM=0`일 경우 무시됩니다.
`MAXLEN`	해당 수신자에 대해 생성된 SMS 메시지에 포함할 최대 총 바이트(즉, 8비트 바이트)입니다. `MAXLEN`과 904 페이지 "MAX_MESSAGE_SIZE" 채널 옵션에 지정된 값 중에서 작은 값이 사용됩니다.

표 C–1 SMS 속성 (계속)

속성 이름	속성 값 및 사용법
MAXPAGES	해당 수신자에 대해 전자 메일을 분할할 최대 SMS 메시지 수입니다. MAXPAGES와 904 페이지 "MAX_PAGES_PER_MESSAGE" 채널 옵션에 지정된 값 중에서 작은 값이 사용됩니다.
NPI	ID 속성으로 지정된 대상 SMS 주소에 대한 NPI(Numeric Plan Indicator) 값을 지정합니다. 이 속성에 허용되는 값에 대한 자세한 내용은 907 페이지 "DEFAULT_DESTINATION_NPI" 채널 옵션의 설명을 참조하십시오. 이 속성을 사용할 경우 이 속성 값은 DEFAULT_DESTINATION_NPI 채널 옵션이 제공한 값보다 우선합니다.
PAGELEN	해당 수신자에 대해 단일 SMS 메시지에 포함할 최대 바이트 수입니다. 이 값과 904 페이지 "MAX_PAGE_SIZE" 채널 옵션에 지정된 값 중에서 최소값이 사용됩니다.
TO	ID의 동의어입니다.
TO_NPI	NPI의 동의어입니다.
TO_TON	TON의 동의어입니다.
TON	ID 속성으로 지정된 대상 SMS 주소에 대한 TON(Type of Number) 값을 지정합니다. 이 속성에 허용되는 값에 대한 자세한 내용은 908 페이지 "DEFAULT_DESTINATION_TON" 채널 옵션의 설명을 참조하십시오. 이 속성을 사용할 경우 이 속성 값은 DEFAULT_DESTINATION_TON 채널 옵션이 제공한 값보다 우선합니다.

다음은 몇 가지 주소 예입니다.

```
123456@sms.siroe.com
/id=123456/@sms.siroe.com
/id=123456/maxlen=100/@sms.siroe.com
/id=123456/maxpages=1/@sms.siroe.com
```

전자 메일 주소의 SMS 대상 주소 부분에 대해 변환, 유효성 검사 및 기타 작업을 수행하는 방법은 891 페이지 "C.2.4 사이트 정의 주소 유효성 검사 및 변환"을 참조하십시오.

C.2.2 전자 메일에서 SMS로의 변환 프로세스

전자 메일을 원격 사이트로 보내려면 전자 메일을 원격 SMSC가 이해할 수 있는 SMS 메시지로 변환해야 합니다. 이 절에서는 SMS 채널의 대기열에 포함된 전자 메일을 하나 이상의 SMS 메시지로 변환하는 과정에 대해 설명합니다. 아래에 설명된 것처럼 옵션을 사용하여 생성되는 SMS 메시지의 최대 개수, 이러한 SMS 메시지의 최대 총 길이, 모든 단일 SMS 메시지의 최대 크기를 제어할 수 있습니다. 전자 메일의 텍스트 부분(즉, MIME 텍스트 컨텐트 유형)만 사용되며 변환되는 부분의 최대 개수를 제어할 수도 있습니다.

전자 메일 메시지의 헤더 행과 텍스트 부분에 사용된 문자 세트는 모두 유니코드로 변환된 다음 적절한 SMS 문자 세트로 변환됩니다.

SMS_TEXT 매핑 테이블(892 페이지 "C.2.5 사이트 정의 텍스트 변환" 참조)이 없을 경우 SMS 채널의 대기열에 포함된 전자 메일 메시지는 그림 C–2에 설명된 처리를 거칩니다.

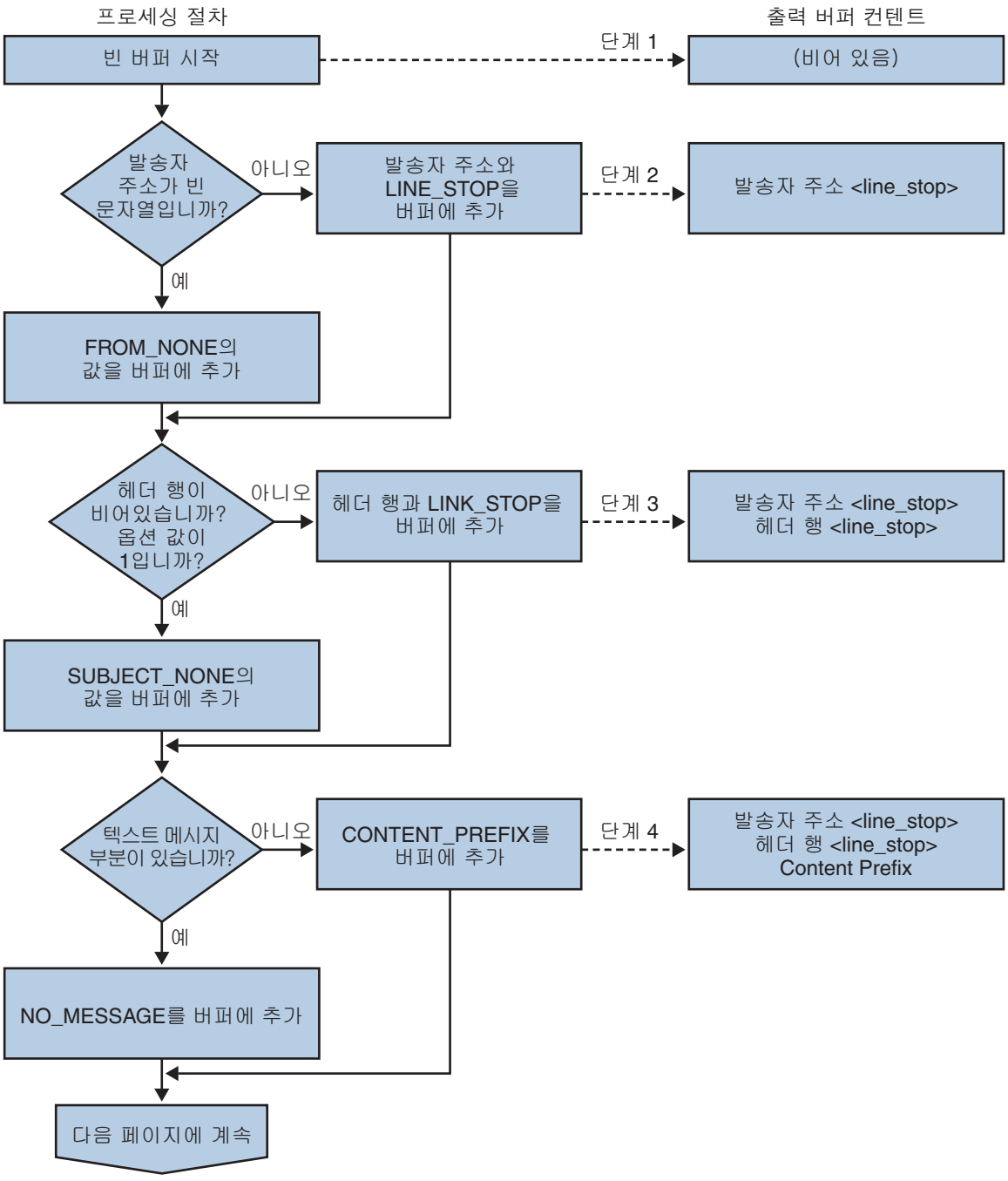


그림 C–2 SMS 채널 전자 메일 처리

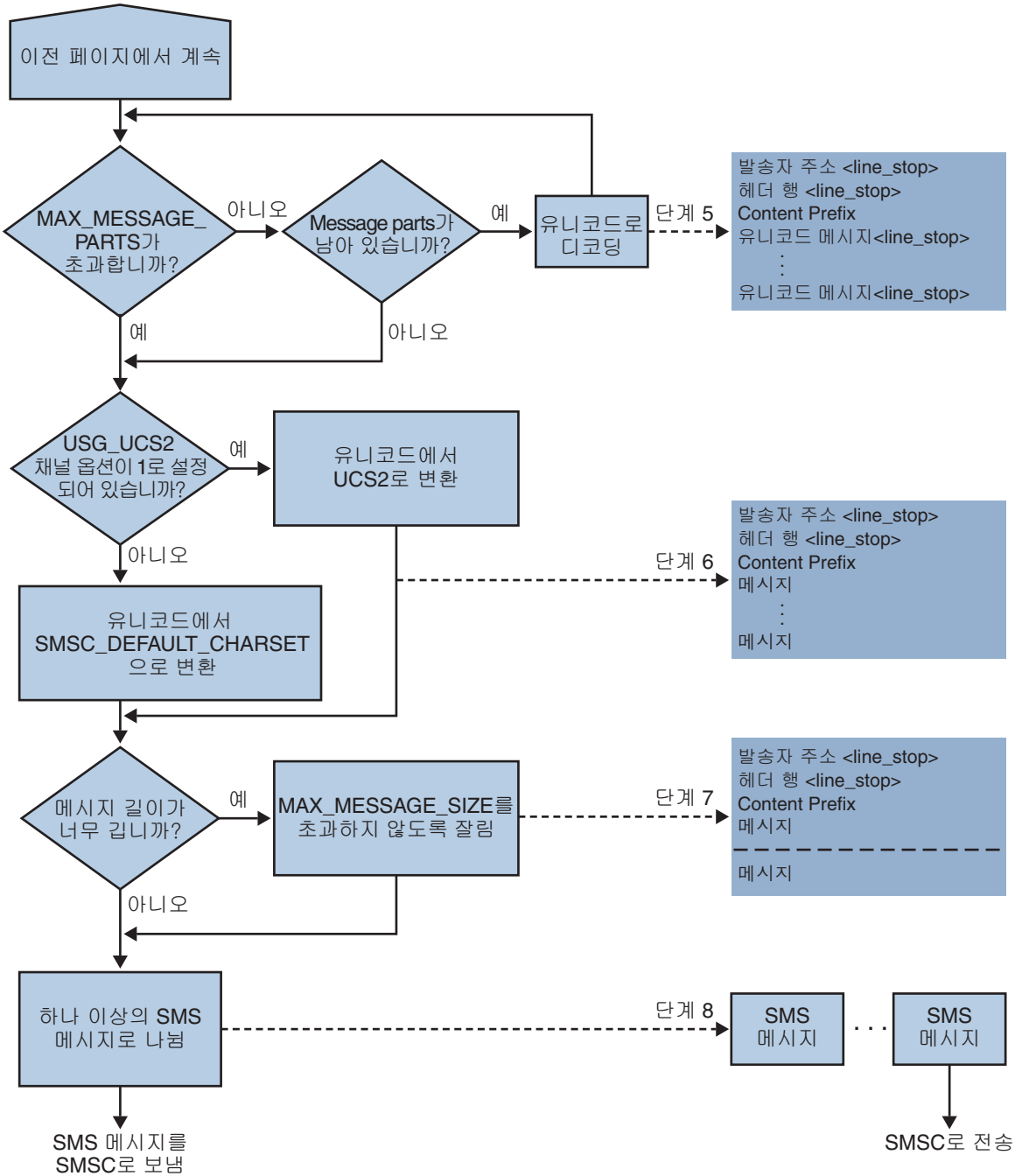


그림 C–3 SMS 채널 전자 메일 처리(계속)

다음 단계는 번호가 매겨진 그림 C–2의 상자에 해당합니다.

1. 빈 출력 버퍼가 시작됩니다. 버퍼에 사용되는 문자 세트는 유니코드입니다.
2. 전자 메일 메시지 발송자 주소를 선호하는 순서로 내림차순으로 나열한 다음 5개의 소스 중 하나에서 가져옵니다.

   1. `Resent-from:`
   2. `From:`
   3. `Resent-sender:`
   4. `Sender:`
   5. `Envelope From:`

   메일 발송자 주소가 빈 문자열일 경우 917 페이지 “FROM_NONE” 채널 옵션 값이 대신 버퍼에 추가됩니다.

   그러나 메일 발송자 주소가 빈 문자열이 아닐 경우 917 페이지 “FROM_FORMAT” 채널 옵션의 처리 결과와 `LINE_STOP` 채널 옵션 값이 출력 버퍼에 추가됩니다.

   `Resent-from:` 및 `Resent-sender:` 헤더 행은 906 페이지 “USE_HEADER_RESENT” 옵션 값이 1인 경우에만 고려됩니다. 그렇지 않으면 `Resent-` 헤더 행은 무시됩니다.
3. `Subject:` 헤더 행이 존재하지 않거나 비어 있을 경우 918 페이지 “SUBJECT_NONE” 옵션 값이 출력 버퍼에 추가됩니다.

   그렇지 않을 경우 918 페이지 “SUBJECT_FORMAT” 옵션의 처리 결과와 917 페이지 “LINE_STOP” 채널 옵션 값이 출력 버퍼에 추가됩니다.
4. 텍스트 메시지 부분이 없을 경우 918 페이지 “NO_MESSAGE” 채널 옵션 값이 출력 버퍼에 추가됩니다.

   텍스트 메시지 부분이 없을 경우 916 페이지 “CONTENT_PREFIX” 채널 옵션 값이 출력 버퍼에 추가됩니다.

   텍스트 메시지 부분이 아닌 부분은 무시됩니다.
5. 각 텍스트 부분에 대해 `MAX_MESSAGE_PARTS` 한도에 도달하지 않은 동안에 텍스트 부분은 유니코드로 디코딩되고 `LINE_STOP` 채널 옵션 값과 함께 버퍼에 추가됩니다.
6. 그런 다음 결과 출력 버퍼는 유니코드에서 SMSC의 기본 문자 세트 또는 UCS2(UTF-16)로 변환됩니다. SMSC의 기본 문자 세트는 905 페이지 “SMSC_DEFAULT_CHARSET로 변환” 옵션으로 지정합니다.
7. 변환된 문자열은 904 페이지 “MAX_MESSAGE_SIZE” 바이트를 초과하지 않도록 잘립니다.
8. 883 페이지 “C.2.2 전자 메일에서 SMS로의 변환 프로세스”에서 변환된 문자열은 하나 이상의 SMS 메시지로 나뉘며 하나의 SMS 메시지는 `MAX_PAGE_SIZE` 바이트를 초과하지 않습니다. 최대한 904 페이지 “MAX_PAGES_PER_MESSAGE”에 지정된 수만큼 SMS 메시지가 생성됩니다.

주 – 전자 메일 메시지가 여러 수신자를 가질 수 있으므로 4페이지의 "전자 메일을 채널로 전송"에 설명된 MAXLEN, MAXPAGES 또는 PAGELEN 속성을 사용하는 각 수신자 주소에 대해 단계 6에서 단계 8까지 수행해야 할 수 있습니다.

C.2.2.1 샘플 전자 메일 처리

예를 들어, 채널의 기본 설정을 가진 전자 메일 메시지는 다음과 같습니다.

```
From: John Doe
To: 1234567@sms.siroe.com
Subject: Today's meeting
Date: Fri, 26 March 2001 08:17

The staff meeting is at 14:30 today in the big conference room.
```

다음 SMS 메시지로 변환됩니다.

```
jdoe@siroe.com (Today's meeting) The staff meeting is at 14:30 today in the big
conference room.
```

다음과 같이 일련의 옵션 집합 설정이 다른 경우에는

```
CONTENT_PREFIX=Msg:
FROM_FORMAT=From:${pa}
SUBJECT_FORMAT=Subj:$s
```

다음 결과가 생성됩니다.

```
From:John Doe Subj:Today's meeting Msg:The staff meeting is at 14:30 today in the
big conference room.
```

C.2.3 SMS 메시지 전송 프로세스

전자 메일이 하나 이상의 SMS 메시지로 변환된 후(각 수신자에 대해 다른 집합을 가질 수 있음) SMS 메시지는 대상 SMSC로 전송됩니다. 이러한 전송은 TCP/IP를 통한 SMPP V3.4를 사용하여 수행됩니다. SMS 채널과 연관된 공식 호스트 이름으로 사용하기 위해 SMPP 서버의 호스트 이름(SMPP_SERVER)을 가져오며 사용할 TCP 포트(SMPP_PORT)가 port 채널 키워드를 통해 지정됩니다.

처리할 메시지가 있으면 채널이 시작됩니다. 채널은 송신기로서 SMPP 서버에 바인드하여 913 페이지 "C.3.3.4 SMPP 옵션"에 설명된 ESME_ 채널 옵션으로 지정한 자격 증명을 제공합니다. 표 C–2에는 BIND_TRANSMITTER PDU(Protocol Data Unit)에서 설정되는 필드와 해당 값이 나열되어 있습니다.

표 C–2 BIND_TRANSMITTER PDU에서 생성되는 필드

필드	값
system_id	914 페이지 “ESME_SYSTEM_ID” 채널 옵션이며 기본값은 빈 문자열입니다.
password	914 페이지 “ESME_PASSWORD” 채널 옵션이며 기본값은 빈 문자열입니다.
system_type	914 페이지 “ESME_SYSTEM_TYPE” 채널 옵션이며 기본값은 빈 문자열입니다.
interface_version	SMPP V3.4를 나타내는 0x34입니다.
addr_ton	913 페이지 “ESME_ADDRESS_TON”이며 기본값은 알 수 없는 TON을 나타내는 0x00입니다.
addr_npi	913 페이지 “ESME_ADDRESS_NPI”이며 기본값은 알 수 없는 NPI를 나타내는 0x00입니다.
addr_range	914 페이지 “ESME_IP_ADDRESS” 채널 옵션이며 기본값은 빈 문자열입니다.

채널이 다중 스레드되는 점에 주의하십시오. 보낼 메일의 수에 따라 대기열에서 제외 스레드가 채널에서 여러 개 실행 중일 수 있습니다. (여러 채널 프로세스가 실행 중일 수도 있습니다.)각 스레드는 BIND_TRANSMITTER를 수행한 후 해당 TCP/IP 연결에서 전송해야 할 모든 SMS 메시지를 보낸 다음 UNBIND를 보내고 연결을 닫습니다. 잠재적 재사용을 위해 유휴 기간 동안 연결을 열어두려는 시도는 수행되지 않습니다. 원격 SMPP 서버가 스로틀(throttle) 오류를 되돌려 보낼 경우 UNBIND가 실행되고 TCP/IP 연결이 닫힌 후에 새 연결과 BIND가 설정됩니다. 원격 SMPP 서버가 SMS 메시지 전송이 끝나기 전에 UNBIND를 보낼 경우에도 비슷한 동작이 발생합니다.

그런 다음 SMPP SUBMIT_SM PDU가 사용되어 SMS 메시지를 제출합니다. 영구 오류(예: ESME_RINVDSTADR)가 반환될 경우 전자 메일은 전달할 수 없는 것으로 반환됩니다. 임시 오류가 반환될 경우 전자 메일은 나중에 전달을 시도하기 위해 다시 대기열에 포함됩니다. 영구 오류는 SMS 대상 주소가 잘못된 경우처럼 특정 조건이 무한정 존재하며 반복된 전달 시도가 실제적인 효과가 없는 오류를 말합니다. 이와 달리 임시 오류는 서버 다운이나 서버 정체와 같이 특정 조건이 머지 않아 없어질 것으로 보이는 오류입니다.

USE_HEADER_FROM 옵션 값이 1일 경우 제출된 SMS 메시지의 소스 주소가 설정됩니다. 사용되는 값은 원본 전자 메일에서 파생되며 모든 응답을 전송해야 할 가장 가능성 있는 (전자 메일) 주소가 선택됩니다. 이에 따라 선호하는 순서로 내림차순으로 나열한 다음 7개의 소스 중 하나에서 소스 주소를 가져옵니다.

1. Resent-reply-to:
2. Resent-from:
3. Reply-to:

4. From:
5. Resent-sender:
6. Sender:
7. Envelope From:

Resent-reply-to: 및 Reply-to: 헤더 행은 906 페이지 “USE_HEADER_REPLY_TO” 옵션 값이 1인 경우에만 고려됩니다. Resent-reply-to:, Resent-from: 및 Resent-sender: 헤더 행은 906 페이지 “USE_HEADER_RESENT” 옵션 값이 1인 경우에만 고려됩니다. 이것은 모든 옵션 값이 1이어야 Resent-reply-to: 헤더 행이 고려된다는 의미입니다.이러한 두 옵션의 기본값은 모두 0입니다. 따라서 4, 6 및 7 항목만 기본 구성에서 고려됩니다.
마지막으로 SMS 메시지의 소스 주소가 20바이트로 제한되므로 선택된 소스 주소는 이 제한을 초과할 경우 잘립니다.

표 C–3은 SUBMIT_SM PDU에서 설정되는 필수 필드를 보여 줍니다.

표 C–3 SUBMIT_SM PDU에서 생성되는 필수 필드

필드	값
service_type	911 페이지 “DEFAULT_SERVICE_TYPE”채널 옵션이며 기본값은 빈 문자열입니다.
source_addr_ton	912 페이지 “DEFAULT_SOURCE_TON” 채널 옵션이며 USE_HEADER_FROM=1일 경우 이 필드는 일반적으로 영숫자 TON을 나타내는 값 0x05를 가집니다. 그렇지 않을 경우에는 국가별 TON을 나타내는 기본값 0x01이 사용됩니다.
source_addr_npi	911 페이지 “DEFAULT_SOURCE_NPI” 채널 옵션이며 기본값은 0x00입니다.
source_addr	USE_HEADER_FROM=0일 경우 911 페이지 “DEFAULT_SOURCE_ADDRESS” 채널 옵션입니다. 그렇지 않을 경우에는 전자 메일 메시지의 발송자를 나타내는 영숫자 문자열입니다.
dest_addr_ton	TON 주소 지정 속성 또는 908 페이지 “DEFAULT_DESTINATION_TON” 채널 옵션입니다. 기본값은 국가별 TON을 나타내는 0x01입니다.
dest_addr_npi	NPI 주소 지정 속성 또는 911 페이지 “DEFAULT_SOURCE_NPI” 채널 옵션입니다. 기본값은 알 수 없는 NPI를 나타내는 0x00입니다.
dest_addr	전자 메일 봉투의 To: 주소의 로컬 부분에서 파생된 대상 SMS 주소입니다. 882 페이지 “C.2.1 전자 메일을 채널로 전송”을 참조하십시오.
esm_class	단방향 SMS의 경우 기본 SMSC 메시지 유형인 저장 및 전달 모드를 나타내는 0x03으로 설정되며 응답 경로를 설정하지 않습니다. 양방향 SMS 메시지의 경우 0x83으로 설정됩니다.

표 C–3 SUBMIT_SM PDU에서 생성되는 필수 필드 (계속)

필드	값
protocol_id	0x00이며 CDMA 및 TDMA에 사용되지 않습니다. GSM의 경우 0x00은 인터넷은 연결되어 있지 않지만 SME 간 프로토콜이 있음을 나타냅니다.
priority_flag	GSM 및 CDMA의 경우 0x00이고 TDMA의 경우 0x01입니다. 모두 일반 우선 순위를 나타냅니다. 909 페이지 “DEFAULT_PRIORITY” 채널 옵션에 대한 설명을 참조하십시오.
schedule_delivery_time	즉시 전달을 나타내는 빈 문자열입니다.
validity_period	912 페이지 “DEFAULT_VALIDITY_PERIOD” 채널 옵션이며 기본값은 SMSC의 기본값을 사용해야 한다는 것을 나타내는 빈 문자열입니다.
registered_delivery	등록된 전달이 없음을 나타내는 0x00입니다.
replace_if_present_flag	모든 이전 SMS 메시지를 대체해야 한다는 것을 나타내는 0x00입니다.
data_coding	SMSC 기본 문자 세트의 경우 0x00이며 UCS2 문자 세트의 경우 0x08입니다.
sm_default_msg_id	미리 정의된 메시지를 사용하지 않는다는 것을 나타내는 0x00입니다.
sm_length	SMS 메시지의 길이와 내용입니다. 자세한 내용은 883 페이지 “C.2.2 전자 메일에서 SMS로의 변환 프로세스”를 참조하십시오.
short_message	SMS 메시지의 길이와 내용입니다. 자세한 내용은 883 페이지 “C.2.2 전자 메일에서 SMS로의 변환 프로세스”를 참조하십시오.

표 C–4는 SUBMIT_SM PDU의 선택 필드를 보여 줍니다.

표 C–4 SUBMIT_SM PDU에서 생성되는 선택적 필드

필드	값
privacy	910 페이지 “DEFAULT_PRIVACY” 채널 키워드에 대한 설명을 참조하십시오. 기본값은 전자 메일 메시지에 Sensitivity: 헤더 행 값이 없으면 이러한 필드를 제공하지 않는 것입니다.
sar_refnum	913 페이지 “USE_SAR” 채널 키워드에 대한 설명을 참조하십시오. 기본값은 이러한 필드를 제공하지 않는 것입니다.
sar_total	위의 sar_refnum을 참조하십시오.
sar_seqnum	위의 sar_refnum을 참조하십시오.

전송할 SMS 메시지가 더 이상 없거나(메시지 대기열이 비어 있거나) 914 페이지 “MAX_PAGES_PER_BIND”를 초과할 때까지 채널은 SMPP 서버에 바인드되어 있습니다. 후자의 경우 전송할 추가 SMS 메시지가 남아 있으면 새 연결이 설정되고 바인드 작업이 수행됩니다.

SMS 채널이 다중 스레드되는 점에 주의하십시오. 채널의 각 처리 스레드는 SMPP 서버와의 고유한 TCP 연결을 유지 관리합니다. 예를 들어, 각각 전송할 SMS 메시지를 가진 세 개의 처리 스레드가 있을 경우 채널은 SMPP 서버에 대한 세 개의 열린 TCP 연결을 가집니다. 각 연결은 송신기로 SMPP 서버에 바인드합니다. 또한 지정된 모든 처리 스레드는 미해결 SMS 전송을 한 번에 하나씩만 가집니다. 즉, 지정된 스레드는 SMS 메시지를 전송한 다음 다른 SMS 메시지를 전송하기 전에 전송 응답(즉, SUBMIT_SM_RESP PDU)을 기다립니다.

C.2.4 사이트 정의 주소 유효성 검사 및 변환

사이트는 882 페이지 “C.2.1 전자 메일을 채널로 전송”에 설명된 수신자 전자 메일 주소로 인코딩되는 SMS 대상 주소에 유효성 검사 또는 변환을 적용할 수 있습니다. 예를 들어, 사이트에서 다음을 수행할 수 있습니다.

- 숫자가 아닌 문자 스트라이프(예: 800.555.1212를 8005551212로 변환)
- 접두어 추가(예: 8005551212를 +18005551212로 변환)
- 정확성 검증(예: 123은 너무 짧음)

처음 두 개의 작업은 특히 912 페이지 “DESTINATION_ADDRESS_NUMERIC” 및 913 페이지 “DESTINATION_ADDRESS_PREFIX” 채널 옵션으로 수행할 수 있습니다. 일반적으로 이 세 작업 및 기타 작업은 매핑테이블(다시 쓰기 규칙의 매핑 테이블 콜아웃 또는 FORWARD 매핑 테이블)을 사용하여 구현할 수 있습니다. 다시 쓰기 규칙의 매핑 테이블 콜아웃을 사용하면 사이트 정의 오류 응답으로 주소를 거부하는 기능을 비롯하여 가장 뛰어난 유연성이 제공됩니다. 이 절의 나머지 부분에서는 다시 쓰기 규칙의 매핑 테이블 콜아웃을 사용하는 이러한 방식에 대해 설명합니다.

대상 주소가 10 또는 11자리의 숫자로만 되고 문자열 "+1"을 접두어로 붙여야 하는 것으로 가정합니다. 이는 다음 다시 쓰기 규칙을 사용하여 수행할 수 있습니다.

```
sms.siroe.com       ${X-REWRITE-SMS-ADDRESS,$U}@sms.siroe.com
sms.siroe.com        $?Invalid SMS address
```

위에서 첫 번째 다시 쓰기 규칙은 X-REWRITE-SMS-ADDRESS라는 사이트 정의 매핑 테이블로 콜아웃됩니다. 검사를 위해 전자 메일 주소의 로컬 부분이 이 매핑 테이블로 전달됩니다. 매핑 프로세스에서 로컬 부분을 허용할 경우에는 주소가 수락되어 SMS 채널에 다시 기록됩니다. 매핑 프로세스가 로컬 부분을 허용하지 않을 경우에는 다음 다시 쓰기 규칙이 적용됩니다. 다음 다시 쓰기 규칙이 $? 다시 쓰기 규칙이므로 "Invalid SMS address"라는 오류 텍스트와 함께 주소가 거부됩니다.

아래에는 X-REWRITE-SMS-ADDRESS 매핑 테이블이 나와 있습니다. 이 매핑 테이블은 속성값 쌍 목록 형식이나 단순히 원시 SMS 대상 주소로 되어 있는 로컬 부분에 대한 필수 검증 단계를 수행합니다.

```
X-VALIDATE-SMS-ADDRESS

! Iteratively strip any non-numeric characters
   $_*$[$ -/:-~]%*  $0$2$R
! Accept the address if it is of the form 1nnnnnnnnnn or nnnnnnnnnn
! In accepting it, ensure that we output +1nnnnnnnnnn
   1%%%%%%%%%%      +1$0$1$2$3$4$5$6$7$8$9$Y
   %%%%%%%%%%       +1$0$1$2$3$4$5$6$7$8$9$Y
! We didn't accept it and consequently it's invalid
   *                 $N

X-REWRITE-SMS-ADDRESS
    */id=$_*/*       $C$0/id=$|X-VALIDATE-SMS-ADDRESS;$1|/$2$Y$E
    */id=$_*/*       $N
    *                $C$|X-VALIDATE-SMS-ADDRESS;$0|$Y$E
    *                $N
```

위와 같이 설정된 경우 912 페이지 "DESTINATION_ADDRESS_NUMERIC" 옵션 값이 0(기본값)인지 확인합니다. 그렇지 않을 경우 "+"가 SMS 대상 주소에서 스트라이프됩니다.

C.2.5 사이트 정의 텍스트 변환

사이트는 변환 규칙 테이블을 사용하여 883 페이지 "C.2.2 전자 메일에서 SMS로의 변환 프로세스"에 설명된 단계 1-6을 사용자 정의할 수 있습니다. 이러한 규칙은 MTA 매핑 파일의 매핑 테이블을 통해 지정됩니다.

매핑 테이블 이름은 *SMS_Channel*_TEXT여야 하며 여기에서 *SMS_Channel*은 SMS 채널의 이름입니다(예: 채널 이름이 sms인 경우 SMS_TEXT 또는 채널 이름이 sms_mway인 경우 SMS_MWAY_TEXT).

이 매핑 테이블에서 두 가지 유형의 항목을 만들 수 있습니다. 그러나 이러한 항목의 형식을 설명하기 전에 매핑 파일의 사용 방법을 이해하는 것이 중요합니다. 매핑 파일의 사용 방법을 아는 것은 이러한 항목을 생성 및 사용하는 방법을 이해하는 데 있어 필수적입니다. 이러한 두 유형의 항목이 설명된 후에는 매핑 테이블 예가 제공됩니다.

두 가지 유형의 항목은 다음과 같습니다.

- 893 페이지 "C.2.5.1 메시지 헤더 항목"
- 893 페이지 "C.2.5.2 메시지 본문 항목"

C.2.5.1 메시지 헤더 항목

이러한 항목은 SMS 메시지에 포함해야 하는 메시지 헤더 행과 이러한 메시지 헤더 행을 축약 또는 변환하는 방법을 지정합니다. 헤더 행은 이러한 항목 중 하나에 의해 길이가 0이 아닌 문자열로 성공적으로 매핑된 경우에만 생성할 SMS 메시지에 포함됩니다. 각 항목은 다음 형식을 가집니다.

H|*pattern replacement-text*

메시지 헤더 행은 패턴과 일치할 경우 매핑 파일의 패턴 일치 및 문자열 교체 기능을 사용하여 교체 텍스트 replacement-text로 교체됩니다. 그런 다음 메타 문자 $Y가 교체 텍스트에 지정된 경우 헤더 행 매핑의 최종 결과는 SMS 메시지에 포함됩니다. 헤더 행이 패턴 문자열과 일치하지 않거나 길이가 0인 문자열에 매핑되거나 교체 텍스트에 $Y 메타 문자가 지정되지 않은 경우 SMS 메시지에서 헤더 행이 생략됩니다. 다음 두 항목은

```
H|From:* F:$0$Y
H|Subject:* S:$0$Y
```

From: 및 Subject: 헤더 행을 SMS 메시지에 From: 및 Subject:의 약어 F: 및 S:로 포함시킵니다. 다음 항목은

```
H|Date:* H|D:$0$R$Y
H|D:*,*%19%%*:*:* H|D:$0$ $5:$6$R$Y
```

Date: 헤더 행을 허용 및 매핑되도록 합니다. 예를 들어, 다음 헤더 행은

```
Date: Wed, 16 Dec 1992 16:13:27 -0700 (PDT)
```

다음과 같이 변환됩니다.

```
D: Wed 16:13
```

매우 복잡한 반복 매핑을 작성할 수 있습니다. 사용자 정의 필터를 설정하려는 사이트는 우선 매핑 파일의 작동 방법을 이해해야 할 수 있습니다. 항목의 오른쪽에 있는 H|는 원할 경우 생략할 수 있습니다. 반복 매핑 집합에 필요한 테이블 항목 수를 줄이기 위해 오른쪽에 H|가 허용됩니다.

C.2.5.2 메시지 본문 항목

이러한 항목은 메시지 본문의 각 행에 적용할 매핑을 설정합니다. 메시지 본문의 각 행은 작성될 SMS 메시지에 통합되기 전에 이러한 매핑을 통해 전달됩니다. 이러한 항목은 다음 형식을 가집니다.

B|*pattern* B| *replacement-text*

메시지 본문의 행은 *pattern* 패턴과 일치할 경우 교체 텍스트 *replacement-text*로 교체됩니다. 마찬가지로 이 기능을 사용하여 매우 복잡한 반복 매핑을 생성할 수 있습니다. 항목의 오른쪽에 있는 B|는 원할 경우 생략할 수 있습니다.

C.2.5.3 SMS 매핑 테이블 예

예 C–1에는 SMS_TEXT 매핑 테이블 예가 나와 있습니다. 각 행의 끝에 있는 괄호 안의 숫자는 바로 뒤에 오는 894 페이지 "설명 텍스트" 절의 항목 번호에 해당합니다.

예 C–1 SMS_TEXT 매핑 테이블 예

```
SMS_TEXT

   H|From:*         H|F:$0$R$Y       (1)
   H|Subject:*      H|S:$0$R$Y       (1)
   H|F:*<*>*        H|F:$1$R$Y       ()
   H|F:*(*)*        H|F:$0$2$R$Y     (2)
   H|F:*"*"*        H|F:$0$2$R$Y     (3)
   H|F:*@*          H|F:$0$R$Y       (4)
   H|%:$ *          H|$0:$1$R$Y      (5)
   H|%:*$           H|$0:$1$R$Y      (5)
   H|%:*$ $ *       H|$0:$1$ $2$R$Y  (6)
   B|*--*           B|$0-$1$R        (7)
   B|*..*           B|$0.$1$R        (7)
   B|*!!*           B|$0!$1$R        (7)
   B|*??*           B|$0?$1$R        (7)
   B|*$ $ *         B|$0$ $1$R       (6)
   B|$ *            B|$0$R           (5)
   B|*$             B|$0$R           (5)
```

설명 텍스트

이 절에서는 위의 SMS_TEXT 매핑 테이블 예에 포함된 항목에 대해 설명합니다.

위 예에서는 매핑의 반복 적용을 구현 및 제어하기 위해 메타 문자 $R이 사용됩니다. 이러한 매핑을 반복함으로써 강력한 필터링이 수행됩니다. 예를 들어, 단일 선행 또는 후행 공백을 제거하거나(6) 두 개의 공백을 하나의 공백으로 줄이는(7) 간단한 매핑은 서로 결합되어 모든 선행 및 후행 공백을 스트라이프하고 연속된 여러 공백을 모두 하나의 공백으로 줄이는 필터가 됩니다. 이러한 필터링은 각 SMS 메시지의 크기를 줄이는 데 도움이 됩니다.

1. 이 두 항목은 From: 및 Subject: 헤더 행을 SMS 메시지에 포함시킵니다. From: 및 Subject:는 각각 약어 F: 및 S:로 표시됩니다. 일부 다른 항목이 From: 및 Subject: 헤더 행에 영향을 줄 수 있습니다.

   이 항목은 <...> 패턴을 포함하는 From: 헤더 행을 대괄호 안의 텍스트만으로 줄입니다. 예를 들면 다음과 같습니다.

   ```
   F: "John C. Doe" <jdoe@siroe.com> (Hello)
   ```

   다음 행으로 교체됩니다.

   ```
   F: jdoe@siroe.com
   ```

2. 이 항목은 From: 헤더 행의 (...) 패턴을 포함하여 그 안에 있는 모든 것을 제거합니다. 예를 들면 다음과 같습니다.

   ```
   F: "John C. Doe" <jdoe@siroe.com> (Hello)
   ```

   다음 행으로 교체됩니다.

   ```
   F: "John C. Doe" <jdoe@siroe.com>
   ```

3. 이 항목은 From: 헤더 행의 "..." 패턴을 포함하여 그 안에 있는 모든 것을 제거합니다. 예를 들면 다음과 같습니다.

   ```
   F: "John C. Doe" <jdoe@siroe.com> (Hello)
   ```

   다음 행으로 교체됩니다.

   ```
   F: <jdoe@siroe.com> (Hello)
   ```

4. 이 항목은 From: 헤더 행의 at 기호(@)를 포함하여 그 오른쪽에 있는 모든 것을 제거합니다. 예를 들면 다음과 같습니다.

   ```
   F: "John C. Doe" <jdoe@siroe.com> (Hello)
   ```

   다음 행으로 교체됩니다.

   ```
   F: "John C. Doe" <jdoe@
   ```

5. 이러한 네 개의 항목은 메시지 헤더와 본문의 행에서 선행 및 후행 공백을 제거합니다.
6. 이러한 두 개의 항목은 메시지 헤더와 본문의 행에서 두 개의 공백을 하나의 공백으로 줄입니다.
7. 이러한 네 개의 항목은 이중 대시, 마침표, 느낌표 및 물음표를 일치하는 문자 하나로 줄입니다. 마찬가지로 이것은 SMS 메시지의 바이트를 줄이는 데 도움이 됩니다.

항목의 순서는 매우 중요합니다. 예를 들어, 순서가 정해진 메시지에서 메시지의 본문은 From: 헤더 행입니다.

```
From: "John C. Doe" (Hello)
```

다음과 같이 줄어듭니다.

```
jdoe
```

이 작업은 다음과 같은 단계로 수행됩니다.

1. 다음과 같이 From: 헤더 행에서 시작합니다.

   ```
   From: "John C. Doe" (Hello)
   ```

   첫 번째 매핑 항목의 패턴이 일치하여 다음 결과를 생성합니다.

   ```
   F: "John C. Doe" (Hello)
   ```

   결과 문자열의 $R 메타 문자로 인해 결과 문자열이 다시 매핑됩니다.
2. 마지막 단계의 결과 문자열에 매핑이 적용되어다음을 생성합니다.

   ```
   F: jdoe@siroe.com
   ```

매핑의 $R로 인해 전체 매핑 세트가 이 단계의 결과에 다시 적용됩니다.

3. 다음으로 매핑이 적용되어 다음이 생성됩니다.

   ```
   F: jdoe
   ```

   매핑의 $R로 인해 전체 매핑 세트가 이 단계의 결과에 다시 적용됩니다.

4. 다음으로 매핑이 적용되어 다음이 생성됩니다.

   ```
   F:jdoe
   ```

   매핑의 $R로 인해 전체 매핑 세트가 이 단계의 결과에 다시 적용됩니다.

5. 일치하는 다른 항목이 없으므로 다음 최종 결과 문자열이

   ```
   F:jdoe
   ```

   SMS 메시지에 통합됩니다.

주 – `imsimta` 테스트 매핑 유틸리티를 사용하여 매핑 테이블을 테스트할 수 있습니다. 예를 들면 다음과 같습니다.

```
# imsimta test -mapping -noimage_file -mapping_file=test.txt
Enter table name: SMS_TEXT
Input string: H|From: "John C. Doe"  (Hello)
Output string: H|F:jdoe
Output flags: [0,1,2,89]
Input string: ^D
#
```

`imsimta test` 유틸리티에 대한 자세한 내용은 **Sun Java System Messaging Server 6.3 Administration Reference**의 "imsimta test"를 참조하십시오.

C.3 SMS 채널 구성

이 절에서는 단방향(Email-To-Mobile) 및 양방향(Email-To-Mobile 및 Mobile-To-Email) 기능을 모두 사용하도록 SMS 채널을 설정하는 방법에 대해 설명합니다. 923 페이지 "C.3.7 양방향 SMS를 위한 SMS 채널 구성"에 언급된 내용을 제외하고 SMS 채널은 단방향 및 양방향 기능 모두에 대해 동일하게 설정됩니다.

이 절은 다음 내용으로 구성되어 있습니다.

C.3.1 SMS 채널 추가

Messaging Server 구성에 SMS 채널을 추가하려면 다음 두 단계를 수행해야 합니다.

1. 897 페이지 "C.3.1.1 채널 정의 및 다시 쓰기 규칙 추가".
2. 899 페이지 "C.3.2 SMS 채널 옵션 파일 만들기".

모든 상황에서 설정해야 하는 채널 옵션이 있는 것은 아니지만 914 페이지 "ESME_PASSWORD", 914 페이지 "ESME_SYSTEM_ID", 904 페이지 "MAX_PAGE_SIZE", 912 페이지 "DEFAULT_SOURCE_TON" 및 908 페이지 "DEFAULT_DESTINATION_TON" 옵션 중 하나 이상을 설정해야 할 수도 있습니다. 또한 설명된 바와 같이 `imta.cnf` 파일의 채널 정의나 채널 옵션 파일을 통해 SMPP 서버의 호스트 이름 또는 IP 주소와 TCP 포트를 설정해야 합니다.

둘 이상의 SMS 채널을 구성하여 다른 SMS 채널에 다른 특성을 제공할 수 있습니다. 여러 SMS 채널 사용에 대한 자세한 내용은 920 페이지 "C.3.4 SMS 채널 추가"를 참조하십시오.

한 가지 주의해야 할 사항은 `imta.cnf` 파일을 변경할 경우 다시 컴파일해야 한다는 것입니다. 단순히 채널 옵션 파일을 변경할 경우에는 다시 컴파일할 필요가 없습니다.

또한 채널 변경 사항 적용 시간은 변경 사항에 따라 달라질 수 있다는 점에 주의하십시오. 대부분의 채널 옵션 변경 사항은 변경 이후에 시작된 모든 채널에서 적용되며 작업 제어기에서 종종 새 채널을 시작하므로 이것은 거의 즉시 적용되는 것처럼 보일 수 있습니다. 일부 변경 사항은 재컴파일을 수행하고 SMTP 서버를 다시 시작할 때까지 적용되지 않습니다. 이러한 옵션은 채널 자체가 실행될 때가 아니라 메시지가 채널의 대기열에 포함될 때 처리됩니다.

C.3.1.1 채널 정의 및 다시 쓰기 규칙 추가

채널 정의 및 다시 쓰기 규칙을 추가하려면 다음을 수행합니다.

▼ 채널 정의 및 다시 쓰기 규칙 추가 방법

1 SMS 채널을 MTA의 구성에 추가하기 전에 채널 이름을 선택해야 합니다. 채널 이름은 `sms` 또는 `sms_x`가 될 수 있으며 여기서 `x`는 1바이트에서 36바이트 사이의 길이를 가진 대소문자가 구분된 임의의 문자열입니다(예: `sms_mway`).

2 채널 정의를 추가하려면 `installation-directory/config/` 디렉토리에 있는 `imta.cnf` 파일을 편집합니다. 파일의 맨 아래에 빈 행과 다음과 같은 두 행을 추가합니다.

```
channel-name port p threaddepth t \
  backoff "pt2m" "pt5m" "pt10m" "pt30m" notices 1
smpp-host-name
```

여기서 *channel-name*은 사용자가 선택한 채널 이름이고 *p*는 SMPP 서버가 수신하는 TCP 포트이며 *t*는 각 전달 프로세스의 최대 동시 SMPP 서버 연결 수입니다. 마지막으로 *smpp-host-name*은 SMPP 서버를 실행하는 시스템의 호스트 이름입니다.

예를 들어, 다음과 같이 채널 정의를 지정할 수 있습니다.

```
sms_mway port 55555 threaddepth 20 \
backoff "pt2m" "pt5m" "pt10m" "pt30m" notices 1
smpp.siroe.com
```

threaddepth를 계산하는 방법에 대한 지침은 899 페이지 “C.3.1.2 동시 연결 수 제어”를 참조하십시오.

backoff 및 notices 채널 키워드에 대한 자세한 내용은 921 페이지 “C.3.5 전달 재시도 빈도 조정”을 참조하십시오.

smpp-host-name에 호스트 이름 대신 IP를 지정하려면 도메인 리터럴을 지정합니다. 예를 들어, IP 주소가 127.0.0.1인 경우 smpp-host-name에 [127.0.0.1]을 지정합니다. 또는 915 페이지 “SMPP_SERVER” 채널 옵션을 사용할 것을 고려합니다.

주 – Sun Java System Messaging Server 6.1의 경우 master 채널 키워드는 더 이상 사용되지 않습니다. 이 키워드가 있을 경우 무시됩니다.

---

3 채널 정의가 추가된 후에는 파일의 상단 부분으로 가서 다음 형식의 다시 쓰기 규칙을 추가합니다.

```
smpp-host-name $u@smpp-host-name
```

예를 들면 다음과 같습니다.

```
smpp.siroe.com $u@smpp.siroe.com
```

4 imta.cnf 파일을 저장합니다.

5 imsimta cnbuild 명령을 사용하여 구성을 다시 컴파일합니다.

6 imsimta restart dispatcher 명령을 사용하여 SMTP 서버를 다시 시작합니다.

7 위 구성에서는 전자 메일 주소를 id@smpp-host-name(예: 123456@smpp.siroe.com)으로 지정하여 전자 메일을 채널로 전송합니다. 주소 지정에 대한 자세한 내용은 883 페이지 “C.2.2 전자 메일에서 SMS로의 변환 프로세스”를 참조하십시오.

8 선택적으로 SMPP 서버의 호스트 이름을 사용자가 볼 수 없게 하거나 다른 호스트 이름을 동일한 채널과 연관시키려는 경우 다시 쓰기 규칙을 추가합니다. 예를 들어, host-name-1 및 host-name-2를 채널과 연관시키려면 다음을 다시 쓰기 규칙에 추가합니다.

```
host-name-1 $U%host-name-1@smpp-host-name
host-name-2 $U%host-name-2@smpp-host-name
```

예를 들어, SMPP 서버의 호스트 이름이 smpp.siroe.com이지만 사용자에게 전자 메일 주소를 id@sms.sesta.com으로 지정하게 하려면 다음 다시 쓰기 규칙을 추가합니다.

```
sms.sesta.com $U%sms.sesta.com@smpp.siroe.com
```

915 페이지 “SMPP_SERVER” 및 915 페이지 “SMPP_PORT” 채널 옵션은 채널의 공식 호스트 이름과 port 채널 키워드 설정을 무시합니다. SMPP_PORT 옵션이 사용되면 또한 port 키워드를 사용할 필요가 없습니다. 이러한 두 옵션은 적용된 후에 구성을 다시 컴파일하지 않고도 변경할 수 있다는 이점이 있습니다. SMPP_SERVER 옵션의 추가 사용에 대해서는 920 페이지 “C.3.4 SMS 채널 추가”에 설명되어 있습니다.

C.3.1.2 동시 연결 수 제어

threaddepth 채널 키워드는 전달 프로세스 내의 각 전달 스레드에 할당할 메시지 수를 제어합니다. 허용되는 총 동시 연결 수를 계산하려면SMPP_MAX_CONNECTIONS 옵션과 job_limit(SMPP_MAX_CONNECTIONS * job_limit) 옵션의 값을 곱합니다. 915 페이지 “SMPP_MAX_CONNECTIONS” 옵션은 전달 프로세스의 최대 전달 스레드 수를 제어합니다. job_limit 옵션은 채널이 실행되는 작업 제어기 처리 풀에 대해 최대 동시 전달 프로세스 수를 제어합니다.

총 동시 연결 수를 제한하려면 이러한 옵션 중 하나 또는 둘 다를 적절하게 조정해야 합니다. 예를 들어, 원격 SMPP 서버가 단일 연결만 허용할 경우 SMPP_MAX_CONNECTIONS 및 job_limit를 모두 1로 설정해야 합니다. 값을 조정할 때는 job_limit가 1을 초과하도록 허용하는 것이 좋습니다.

C.3.2 SMS 채널 옵션 파일 만들기

일반적으로 채널 옵션 파일은 채널 작업에 필요한 사이트별 매개 변수를 포함합니다. SMS에는 채널 옵션 파일이 필요하지 않습니다. 현재 설치에 채널 옵션 파일이 필요한 경우에는 installation-directory/config/ 디렉토리의 텍스트 파일에 이를 저장합니다. 다른 채널 옵션 파일과 마찬가지로 파일 이름의 형식은 다음과 같습니다.

```
channel_name_option
```

예를 들어, 채널 이름이 sms_mway인 경우 채널 옵션 파일은 다음과 같습니다.

```
installation-directory/config/sms_mway_option
```

각 옵션은 파일에서 다음 형식을 사용하여 단일 행에 포함됩니다.

```
option_name=option_value
```

예를 들면 다음과 같습니다.

```
PROFILE=GSM
SMSC_DEFAULT_CHARSET=iso-8859-1
USE_UCS2=1
```

사용 가능한 SMS 채널 옵션 목록과 각 옵션에 대한 설명은 뒤에 나오는 900 페이지 “C.3.3 사용 가능한 옵션”을 참조하십시오.

C.3.3 사용 가능한 옵션

SMS 채널은 넓은 의미에서 다음과 같은 6개의 범주로 구분되는 여러 옵션을 포함합니다.

- **전자 메일에서 *SMS*로의 변환**: 전자 메일에서 SMS로의 변환 프로세스를 제어하는 옵션입니다.
- ***SMS* 게이트웨이 서버 옵션**: 게이트웨이 프로필 옵션입니다.
- ***SMS* 필드**: 생성된 SMS 메시지의 SMS별 필드를 제어하는 옵션입니다.
- ***SMPP* 프로토콜**: TCP/IP를 통한 SMPP 프로토콜 사용과 관련된 옵션입니다.
- **현지화**: SMS 메시지에 삽입되는 텍스트 필드의 현지화를 허용하는 옵션입니다.
- **기타**: 디버그 및 로깅 옵션입니다.

이러한 옵션은 아래 표에 요약되어 있으며 이후의 절에 보다 자세하게 설명되어 있습니다.

표 C–5 SMS 채널 옵션

전자 메일에서 SMS로의 변환 옵션		
옵션(페이지 번호)	설명	기본값
903 페이지 “GATEWAY_NOTIFICATIONS”	전자 메일 알림 메시지를 SMS 메시지로 변환할지 여부를 지정합니다.	`0`
903 페이지 “MAX_MESSAGE_PARTS”	전자 메일에서 추출할 메시지 부분의 최대 개수입니다.	`2`
904 페이지 “MAX_MESSAGE_SIZE”	전자 메일에서 추출할 최대 바이트 수입니다.	`960`
904 페이지 “MAX_PAGE_SIZE”	단일 SMS 메시지에 포함할 최대 바이트 수입니다.	`160`
904 페이지 “MAX_PAGES_PER_MESSAGE”	전자 메일을 분할할 최대 SMS 메시지 수입니다.	`6`
905 페이지 “ROUTE_TO”	SMS 메시지를 지정된 IP 호스트 이름으로 라우팅합니다.	
905 페이지 “SMSC_DEFAULT_CHARSET로 변환”	SMSC에 사용되는 기본 문자 세트입니다.	`US-ASCII`
905 페이지 “USE_HEADER_FROM”	SMS 소스 주소를 설정합니다.	`0`
906 페이지 “USE_HEADER_PRIORITY”	전자 메일 메시지 헤더에서 우선 순위 정보의 사용을 제어합니다.	`1`
906 페이지 “USE_HEADER_REPLY_TO”	SMS 소스 주소를 생성할 때 `Reply-to:` 헤더 행의 사용을 제어합니다.	`0`
906 페이지 “USE_HEADER_RESENT”	발송자 정보를 생성할 때 `Resent-*:` 헤더 행 사용을 제어합니다.	`0`

표 C–5 SMS 채널 옵션 (계속)

906 페이지 “USE_HEADER_SENSITIVITY”	전자 메일 메시지 헤더에서 개인 정보의 사용을 제어합니다.	1
907 페이지 “USE_UCS2”	적용 가능한 경우 SMS 메시지에서 UCS2 문자 세트를 사용합니다.	1
SMS 게이트웨이 서버 옵션		
907 페이지 “GATEWAY_PROFILE”	SMS 게이트웨이 서버의 구성 파일 sms_gateway.cnf에 구성된 게이트웨이 프로필 이름과 일치합니다.	해당 없음
SMS 필드 옵션		
907 페이지 “DEFAULT_DESTINATION_NPI”	SMS 대상 주소의 기본 NPI입니다.	0x00
908 페이지 “DEFAULT_DESTINATION_TON”	SMS 대상 주소의 기본 TON입니다.	0x01
909 페이지 “DEFAULT_PRIORITY”	SMS 메시지의 기본 우선 순위 설정입니다.	0=GSM, CDMA 1=TDMA
910 페이지 “DEFAULT_PRIVACY”	SMS 메시지의 기본 개인 정보 값 플래그입니다.	-1
911 페이지 “DEFAULT_SERVICE_TYPE”	전송된 SMS 메시지와 연관된 SMS 응용 프로그램 서비스입니다.	해당 없음
911 페이지 “DEFAULT_SOURCE_ADDRESS”	기본 SMS 소스 주소입니다.	0
911 페이지 “DEFAULT_SOURCE_NPI”	SMS 소스 주소의 기본 NPI입니다.	0x00
912 페이지 “DEFAULT_SOURCE_TON”	SMS 소스 주소의 기본 TON입니다.	0x01
912 페이지 “DEFAULT_VALIDITY_PERIOD”	SMS 메시지의 기본 유효 기간입니다.	해당 없음
912 페이지 “DESTINATION_ADDRESS_NUMERIC”	대상 SMS 주소를 0 - 9개의 문자로만 구성되도록 줄입니다.	0
913 페이지 “DESTINATION_ADDRESS_PREFIX”	대상 SMS 주소의 접두어로 사용할 텍스트 문자열입니다.	해당 없음
913 페이지 “PROFILE”	사용할 SMS 프로필입니다.	GSM
913 페이지 “USE_SAR”	SMS sar_ 필드를 사용하는 여러 SMS 메시지의 순서를 정합니다.	0

표 C–5 SMS 채널 옵션 *(계속)*

SMPP 프로토콜 옵션		
913 페이지 “ESME_ADDRESS_NPI”	SMPP 서버에 바인드할 때 지정할 ESME NPI입니다.	0x00
913 페이지 “ESME_ADDRESS_TON”	SMPP 서버에 바인드할 때 지정할 ESME TON입니다.	0x00
914 페이지 “ESME_IP_ADDRESS”	Sun Java System Messaging Server를 실행하는 호스트의 IP 주소입니다.	해당 없음
914 페이지 “ESME_PASSWORD”	SMPP 서버에 바인드할 때 제시할 비밀번호입니다.	해당 없음
914 페이지 “ESME_SYSTEM_ID”	바인드할 때 SMSC에 제시할 시스템 아이디입니다.	해당 없음
914 페이지 “ESME_SYSTEM_TYPE”	바인드할 때 SMSC에 제시할 시스템 유형입니다.	해당 없음
914 페이지 “MAX_PAGES_PER_BIND”	SMPP 서버와의 단일 세션 도중 전송할 최대 SMS 메시지 수입니다.	1024
914 페이지 “REVERSE_ORDER”	멀티파트 SMS 메시지의 전송 시퀀스입니다.	0
915 페이지 “SMPP_MAX_CONNECTIONS”	최대 동시 SMPP 서버 연결 수입니다.	20
915 페이지 “SMPP_PORT”	단방향 SMS의 경우 SMPP 서버가 수신하는 TCP 포트입니다. 양방향 SMS의 경우 SMPP 중계를 위해 LISTEN_PORT에서 사용하는 것과 동일한 TCP 포트입니다.	해당 없음
915 페이지 “SMPP_SERVER”	단방향 SMS의 경우 연결할 SMPP 서버의 호스트 이름입니다. 양방향 SMS의 경우 SMS 게이트웨이 서버의 호스트 이름이나 IP 주소를 가리키도록 설정합니다. SMPP 릴레이의 LISTEN_INTERFACE_ADDRESS 옵션을 사용할 경우 지정된 네트워크 인터페이스 주소와 연관된 호스트 이름이나 IP 주소를 사용해야 합니다.	해당 없음
915 페이지 “TIMEOUT”	SMPP 서버에서 읽기 및 쓰기 완료 시의 시간 초과입니다.	30
현지화 옵션		
916 페이지 “CONTENT_PREFIX”	전자 메일의 내용을 소개하는 텍스트입니다.	Msg:
916 페이지 “DSN_DELAYED_FORMAT”	전달 지연 알림에 대한 서식 지정 문자열입니다.	빈 문자열
916 페이지 “DSN_FAILED_FORMAT”	전달 실패 알림에 대한 서식 지정 문자열입니다.	설명 참조
917 페이지 “DSN_RELAYED_FORMAT”	중계 알림에 대한 서식 지정 문자열입니다.	설명 참조
917 페이지 “DSN_SUCCESS_FORMAT”	성공한 전달 알림에 대한 서식 지정 문자열입니다.	설명 참조
917 페이지 “FROM_FORMAT”	전자 메일 발송자를 나타내기 위해 표시할 텍스트입니다.	$a

표 C–5 SMS 채널 옵션 (계속)

917 페이지 “FROM_NONE”	메일 발송자가 없을 경우 표시할 텍스트입니다.	해당 없음
917 페이지 “LANGUAGE”	(i-default) 텍스트 필드를 선택할 언어 그룹입니다.	i-default
917 페이지 “LINE_STOP”	전자 메일에서 추출한 각 행의 끝에 포함할 텍스트입니다.	공백 문자
918 페이지 “NO_MESSAGE”	메시지에 내용이 없음을 나타내는 텍스트입니다.	]no message]
918 페이지 “SUBJECT_FORMAT”	전자 메일의 제목을 나타내기 위해 표시할 텍스트입니다.	$s
918 페이지 “SUBJECT_NONE”	전자 메일의 제목이 없을 경우 표시할 텍스트입니다.	해당 없음

기타 옵션		
918 페이지 “DEBUG”	세부 정보 디버그 출력을 사용 가능하게 합니다.	6
938 페이지 “LISTEN_CONNECTION_MAX”	모든 SMPP 중계 및 서버 인스턴스화에서 허용할 최대 동시 인바운드 TCP 연결 수입니다.	10,000
938 페이지 “LOG_PAGE_COUNT”	mail.log 파일의 메시지 크기 필드에 기록된 값이 블록이 아니라 페이지 수가 되도록 제어합니다.	0

C.3.3.1 전자 메일에서 SMS로의 변환 옵션

다음 옵션은 전자 메일에서 SMS 메시지로의 변환을 제어합니다. 해당 옵션의 값 범위는 괄호 안에 있습니다. 일반적으로 지정된 전자 메일을 하나 이상의 SMS 메시지로 변환할 수 있습니다. 883 페이지 “C.2.2 전자 메일에서 SMS로의 변환 프로세스”를 참조하십시오.

GATEWAY_NOTIFICATIONS

(*0* **또는** *1*) 전자 메일 알림을 SMS 알림으로 변환할지 여부를 지정합니다. 전자 메일 알림 메시지는 RFC 1892, 1893 및 1894를 준수해야 합니다. 기본값은 0입니다.

GATEWAY_NOTIFICATIONS=0일 경우 이러한 알림은 무시되며 SMS 알림으로 변환되지 않습니다.

알림을 SMS 알림으로 변환할 수 있게 하려면 GATEWAY_NOTIFICATIONS=1을 설정합니다. 이 옵션이 1로 설정되면 현지화 옵션(DSN_*_FORMAT)은 SMS 메시지로 변환되고 게이트웨이를 통해 전송할 알림 유형(성공, 실패, 지연, 중계)을 제어합니다. 알림 유형의 값이 빈 문자열이면 해당 유형 알림이 SMS 메시지로 변환되지 않습니다.

MAX_MESSAGE_PARTS

(**정수**) 멀티파트 전자 메일을 SMS 메시지로 변환할 때 텍스트 부분 중에서 MAX_MESSAGE_PARTS만큼의 처음 부분만 변환됩니다. 나머지 부분은 무시됩니다. 기본적으로 MAX_MESSAGE_PARTS는 2입니다. 메시지 부분의 개수를 제한하지 않으려면

-1을 지정합니다. 값이 0으로 지정되면 SMS 메시지에 포함되는 메시지 내용이 없습니다. 이것은 전자 메일의 헤더 행(예: `Subject:`)만 사용하여 SMS 메시지를 생성하는 효과를 가집니다.

텍스트와 첨부 파일을 모두 포함하는 전자 메일은 일반적으로 두 부분으로 구성됩니다. 또한 일반 텍스트 메시지 부분만 변환되며, 다른 모든 MIME 컨텐트 유형은 무시됩니다.

MAX_MESSAGE_SIZE

(**정수**, *= 10*) 이 옵션을 사용하면 전자 메일에서 생성되는 SMS 메시지에 포함될 총 바이트 수의 상한값을 지정할 수 있습니다. 특히 하나 이상의 생성된 SMS 메시지에 최대 `MAX_MESSAGE_SIZE` 바이트가 사용됩니다. 모든 추가 바이트는 무시됩니다.

기본적으로 960바이트의 상한값이 지정됩니다. `MAX_MESSAGE_SIZE=960`에 해당합니다. 임의의 바이트 수를 허용하려면 값을 0으로 지정합니다.

사용되는 바이트 수는 전자 메일 메시지를 유니코드에서 SMSC의 기본 문자 세트나 UCS2로 변환한 후에 결정됩니다. 이것은 UCS2의 경우 각 UCS2 문자의 길이가 최소 2바이트 이상이므로 960바이트의 `MAX_MESSAGE_SIZE`는 최대 480개의 문자를 생성한다는 것을 의미합니다.

`MAX_MESSAGE_SIZE` 및 904 페이지 “MAX_PAGES_PER_MESSAGE” 옵션은 모두 결과 SMS 메시지의 전체 크기를 제한하는 동일한 목적으로 사용됩니다. 실제로 904 페이지 “MAX_PAGE_SIZE”=960 및 904 페이지 “MAX_PAGE_SIZE”=160은 `MAX_PAGES_PER_MESSAGE=6`을 나타냅니다. 서로 다른 두 개의 옵션이 존재하는 이유는 무엇입니까? 그것은 단일 SMS 메시지의 최대 크기인 `MAX_PAGE_SIZE`를 고려할 필요 없이 페이지의 전체 크기나 수를 제어할 수 있어야 하기 때문입니다. 이 기능은 채널 옵션 파일에서는 중요하지 않을 수 있지만 882 페이지 “C.2.1 전자 메일을 채널로 전송”에 설명된 882 페이지 “C.2.1 전자 메일을 채널로 전송” 또는 882 페이지 “C.2.1 전자 메일을 채널로 전송” 주소 지정 속성을 사용하는 경우에는 중요합니다.

마지막으로 `MAX_MESSAGE_SIZE` 및 `MAX_PAGE_SIZE * MAX_PAGES_PER_MESSAGE`의 두 제한값 중 보다 작은 값이 사용됩니다.

MAX_PAGE_SIZE

(**정수**, *>= 10*) 단일 SMS 메시지에서 허용하는 최대 바이트 수는 `MAX_PAGE_SIZE` 옵션을 사용하여 제어합니다. 기본적으로 160바이트가 사용됩니다. 이것은 `MAX_PAGE_SIZE=160`에 해당합니다.

MAX_PAGES_PER_MESSAGE

(**정수**, *1 - 255*) 주어진 전자 메일에 대해 생성할 최대 SMS 메시지 수를 이 옵션을 사용하여 제어합니다. 실제로 이 옵션은 전자 메일을 잘라 `MAX_PAGES_PER_MESSAGE` SMS 메시지에 들어가는 전자 메일의 해당 부분만 SMS 메시지로 변환합니다. 자세한 내용은 904 페이지 “MAX_PAGE_SIZE” 옵션에 대한 설명을 참조하십시오.

기본적으로 MAX_PAGES_PER_MESSAGE는 904 페이지 “MAX_MESSAGE_SIZE”를 904 페이지 “MAX_PAGE_SIZE”로 나눈 값 이나 1보다 큰 값으로 설정됩니다.

ROUTE_TO

(**문자열**, *IP* **호스트 이름**, *1-64***바이트**) 프로필을 대상으로 하는 모든 SMS 메시지는 다음 형식의 전자 메일 주소를 사용하여 지정된 IP 호스트 이름으로 다시 라우팅됩니다.

```
SMS-destination-address@route-to
```

여기서 SMS-destination-address는 SMS 메시지의 대상 주소이며 route-to는 이 옵션에서 지정되는 IP 호스트 이름입니다. SMS 메시지의 전체 내용이 결과 전자 메일의 내용으로 보내집니다. PARSE_RE_* 옵션은 무시됩니다.

주 – PARSE_RE_* 및 ROUTE_TO 옵션을 동시에 사용할 수 없습니다. 동일한 게이트웨이 프로필에서 두 옵션을 함께 사용하는 것은 구성 오류입니다.

SMSC_DEFAULT_CHARSET로 변환

(*string*) 이 옵션을 사용하여 SMSC의 기본 문자 세트를 지정할 수 있습니다. 다음 파일에서 제공되는 문자 세트 이름을 사용합니다.

```
installation-directory/config/charsets.txt
```

이 옵션을 지정하지 않을 경우 US-ASCII가 사용됩니다. charsets.txt에 사용되는 니모닉 이름은 동일한 디렉토리의 charnames.txt에 정의됩니다.

전자 메일을 처리할 때 헤더 행과 텍스트 메시지 부분은 우선 디코딩된 다음 유니코드로 변환됩니다. 그런 다음 907 페이지 “USE_UCS2” 옵션 값과 SMS 메시지가 기본 SMSC 문자 세트에 없는 최소한 하나 이상의 도형 문자를 포함하는지 여부에 따라 데이터는 SMSC의 기본 문자 세트나 UCS2로 변환됩니다. UCS2 문자 세트는 유니코드의 16비트 인코딩으로 흔히 UTF-16으로 불립니다.

USE_HEADER_FROM

(**정수**, *0-2*) From: 주소를 SMSC에 전달하려면 이 옵션을 설정합니다. 값은 From: 주소를 가져온 위치와 형식을 나타냅니다. 표 C–6에서는 허용 가능한 값과 해당 의미를 보여 줍니다.

표 C–6 USE_HEADER_FROM 값

값	설명
0	SMS 소스 주소가 From: 주소에서 설정되지 않습니다. 발견된 속성값 쌍을 사용합니다.

표 C–6 USE_HEADER_FROM 값 (계속)

값	설명
1	MS 소스 주소를 from-local@from-domain으로 설정합니다. 여기서 From: 주소는@from-route:from-local@from-domain입니다.
2	SMS 소스 주소를 from-local로 설정합니다. 여기서 From: 주소는@from-route:from-local@from-domain입니다.

USE_HEADER_PRIORITY

(*0* **또는** *1*) 이 옵션은 RFC 822 Priority: 헤더 행의 처리를 제어합니다. Priority: 헤더 행의 정보는 결과 SMS 메시지의 우선 순위 플래그를 설정하여 909 페이지 "DEFAULT_PRIORITY" 옵션에서 지정된 기본 SMS 우선 순위를 무시하는 데 사용됩니다. 이 경우는 USE_HEADER_PRIORITY=1에 해당합니다. RFC 822 Priority: 헤더 행을 사용하지 않으려면 USE_HEADER_PRIORITY=0을 지정합니다.

SMS 우선 순위 플래그 처리에 대한 자세한 내용은 DEFAULT_PRIORITY 옵션에 대한 설명을 참조하십시오.

USE_HEADER_REPLY_TO

(*0* **또는** *1*) USE_HEADER_FROM=1인 경우 이 옵션은 Reply-to: 또는 Resent-reply-to: 헤더 행이 SMS 소스 주소를 사용되는지 여부를 제어합니다. 기본적으로 Reply-to: 및 Resent-reply-to: 헤더 행은 무시됩니다. 이것은 옵션 값 0에 해당합니다. 이러한 헤더 행 사용을 고려하려면 옵션 값 1을 사용합니다.

RFC 2822에서는 Reply-to: 및 Resent-reply-to: 헤더 행에 매핑하는 데만 사용됩니다.

USE_HEADER_RESENT

(*0* **또는** *1*) USE_HEADER_FROM=1이면 이 옵션은 Resent- 헤더 행을 SMS 소스 주소로 사용하도록 고려할지 여부를 제어합니다. 기본적으로 Resent- 헤더 행은 무시됩니다. 이것은 옵션 값 0에 해당합니다. 이러한 헤더 행 사용을 고려하려면 옵션 값 1을 사용합니다.

RFC 2822에서는 Resent- 헤더 행을 사용하지 않습니다.

USE_HEADER_SENSITIVITY

(*0* **또는** *1*) USE_HEADER_SENSITIVITY 옵션은 RFC 822 Sensitivity: 헤더 행의 처리를 제어합니다. 기본적으로, Sensitivity: 헤더 행의 정보는 결과 SMS 메시지의 개인 정보 플래그를 설정하여 910 페이지 "DEFAULT_PRIVACY" 옵션에서 지정된 기본 SMS 개인 정보를 무시하는 데 사용됩니다. 이것은 기본값이며 USE_HEADER_SENSITIVITY=1에 해당합니다. RFC 822 Sensitivity: 헤더 행을 사용하지 않으려면 USE_HEADER_SENSITIVITY=0을 지정합니다.

SMS 개인 정보 플래그 처리에 대한 자세한 내용은 910 페이지 “DEFAULT_PRIVACY” 옵션에 대한 설명을 참조하십시오.

USE_UCS2

(*0* **또는** *1*) 채널은 생성되는 SMS 메시지에서 적절한 경우 UCS2 문자 세트를 사용합니다. 이것은 기본 동작이며 `USE_UCS2=1`에 해당합니다. UCS2 문자 세트를 사용하지 않으려면 `USE_UCS2=0`을 지정합니다. 문자 세트 문제에 대한 자세한 내용은 905 페이지 “SMSC_DEFAULT_CHARSET로 변환” 옵션에 대한 설명을 참조하십시오.

표 C–7 USE_UCS2에 대해 유효 값

USE_UCS2 값	결과
1(기본값)	가능한 경우 SMSC 기본 문자 세트가 사용됩니다. 원본 전자 메일이 SMSC 기본 문자 세트에 도형 문자를 포함하지 않을 경우 UCS2 문자 세트가 사용됩니다.
0	항상 SMSC 기본 문자 세트가 사용됩니다. 해당 문자 세트에서 사용할 수 없는 도형 문자는 니모닉으로 나타납니다(예: AE-ligature의 경우 "AE").

C.3.3.2 SMS 게이트웨이 서버 옵션

GATEWAY_PROFILE

SMS 게이트웨이 서버 구성 파일 `sms_gateway.cnf`에 있는 게이트웨이 프로필의 이름입니다.

C.3.3.3 SMS 옵션

다음 옵션을 사용하면 생성된 SMS 메시지에서 SMS 필드를 지정할 수 있습니다.

DEFAULT_DESTINATION_NPI

(**정수**, *0 - 255*) 기본적으로 대상 주소에는 NPI(Numeric Plan Indicator) 값으로 0이 할당됩니다. 이 옵션을 사용하면 0부터 255까지 범위의 대체 정수 값이 할당될 수 있습니다. 다음 표 C–8에서는 일반적인 NPI 값을 보여 줍니다.

표 C–8 Numeric Plan Indicator 값

값	설명
0	알 수 없음
1	ISDN (E.163, E.164)
3	데이터(X.121)

표 C–8 Numeric Plan Indicator 값 *(계속)*

값	설명
4	텔렉스(F.69)
6	육상 이동 통신(E.212)
8	국가
9	Private
10	ERMES
14	IP 주소(인터넷)
18	WAP 클라이언트 아이디
>= 19	정의되지 않음

이 옵션 값은 다음 세 가지 방법 중 하나로 지정할 수 있습니다.

- 10진수 값(예: 10)
- 접두어 "0x"가 있는 16진수 값(예: 0x0a)
- 대소문자를 구분하지 않는 다음 텍스트 문자열 중 하나(괄호 안은 연관된 10진수 값): data(3), default(0), e.163(1), e.164(1), e.212(6), ermes(10), f.69(4), Internet(14), ip(14), isdn(1), land-mobile(6), national(8), private(9), telex(4), unknown(0), wap(18), x.121(3)

DEFAULT_DESTINATION_TON

(**정수**, *0 - 255*) 기본적으로 대상 주소에는 TON(Type of Number) 지정자 값으로 0이 할당됩니다. 이 옵션을 사용하면 0부터 255까지 범위의 대체 정수 값이 할당될 수 있습니다. 다음 표 C–9에서는 일반적인 TON 값을 보여 줍니다.

표 C–9 일반 TON 값

값	설명
0	알 수 없음
1	국제
2	국가
3	네트워크별
4	가입자 번호
5	영숫자
6	축약

표 C–9 일반 TON 값 *(계속)*

값	설명
>=7	정의되지 않음

이 옵션 값은 다음 세 가지 방법 중 하나로 지정할 수 있습니다.

- 10진수 값(예: 10)
- 접두어 "0x"가 있는 16진수 값(예: 0x0a)
- 대소문자를 구분하지 않는 다음 텍스트 문자열 중 하나(괄호 안은 연관된 10진수 값): abbreviated(6), alphanumeric(5), default(0), international(1), national(2), network-specific(3), subscriber(4), unknown(0)

DEFAULT_PRIORITY

*(**정수**, 0 - 255)* SMS 메시지는 필수 우선 순위 필드를 가집니다. 다음 표 C–10에서는 SMS 우선 순위 값이 해석되는 방법을 보여 줍니다.

표 C–10 각 SMS 프로필 유형에 대해 해석되는 SMS 우선 순위 값

값	GSM	TDMA	CDMA
0	낮음	대량	중간
1	우선 순위	중간	대화식
2	우선 순위	Urgent	Urgent
3	우선 순위	매우 높음	긴급

이 옵션을 사용하면 SMS 메시지에 할당되는 기본 우선 순위를 지정할 수 있습니다. 값을 지정하지 않을 경우 기본 우선 순위 0이 `PROFILE=GSM` 및 `CDMA`에 사용되며 우선 순위 1이 913 페이지 “PROFILE”=TDMA에 사용됩니다.

906 페이지 “USE_HEADER_PRIORITY”=1이고 전자 메일에 RFC 822 `Priority:` 헤더 행이 있는 경우 해당 헤더 행에 지정된 우선 순위를 대신 사용하여 결과 SMS 메시지의 우선 순위를 설정합니다. 특히 `USE_HEADER_PRIORITY=0`일 경우에는 SMS 우선 순위 플래그는 항상 `DEFAULT_PRIORITY` 옵션에 따라 설정되며 RFC 822 `Priority:` 헤더 행은 항상 무시됩니다. `USE_HEADER_PRIORITY=1`이고 전자 메일 메시지에 RFC 822 `Priority:` 헤더 행이 있는 경우 해당 헤더 행에 지정된 우선 순위를 대신 사용하여 결과 SMS 메시지의 우선 순위를 설정합니다. 해당 헤더 행이 존재하지 않을 경우 `DEFAULT_PRIORITY` 옵션을 사용하여 SMS 우선 순위 플래그를 설정합니다.

다음 표는 RFC 822 `Priority:` 헤더 행 값을 SMS 우선 순위 플래그로 변환하는 데 사용되는 매핑을 보여 줍니다.

표 C–11 Priority 헤더를 SMS 우선 순위 플래그로 변환하기 위한 매핑

RFC 822	SMS 우선 순위 플래그		
우선순위값	GSM	TDMA	CDMA
세 번째	낮음(0)	대량(0)	중간(0)
두 번째	낮음(0)	대량(0)	중간(0)
낮음	낮음(0)	대량(0)	중간(0)
중간	낮음(0)	중간(1)	중간(0)
Urgent	우선 순위(1)	높음(2)	높음(2)

DEFAULT_PRIVACY

(**정수**, *-1, 0 - 255*) SMS 메시지에서 개인 정보 플래그를 설정할지 여부 및 사용할 값은 DEFAULT_PRIVACY와 906 페이지 "USE_HEADER_SENSITIVITY" 옵션으로 제어합니다. 기본적으로 DEFAULT_PRIVACY에는 -1 값이 사용됩니다. 다음 표 C–12에서는 DEFAULT_PRIVACY 및 906 페이지 "USE_HEADER_SENSITIVITY" 옵션을 다양한 값으로 설정한 결과를 보여 줍니다.

표 C–12 DEFAULT_PRIVACY 및 USE_HEADER_SENSITIVITY에 대한 결과 값

DEFAULT_PRIVACY	USE_HEADER_SENSITIVITY	결과
-1	0	SMS 개인 정보 플래그가 SMS 메시지에서 절대 설정되지 않습니다.
n >= 0	0	SMS 개인 정보 플래그가 항상 값 n으로 설정됩니다. RFC 822 Sensitivity: 헤더 행은 항상 무시됩니다.
-1(기본값)	1(기본값)	원본 전자 메일 메시지에 RFC 822 Sensitivity: 헤더 행이 있을 경우에만 SMS 메시지의 개인 정보 플래그가 설정됩니다. 이 경우 SMS 개인 정보 플래그는 Sensitivity: 헤더 행 값에 해당하도록 설정됩니다. 기본값입니다.
n >= 0	1	SMS 메시지의 개인 정보 플래그는 원본 전자 메일 메시지의 RFC 822 Sensitivity: 헤더 행과 일치하도록 설정됩니다. 전자 메일에 Sensitivity: 헤더 행이 없을 경우 SMS 개인 정보 플래그의 값은 n으로 설정됩니다.

다음 표 C–13에서는 개인 정보 값의 SMS 해석을 보여 줍니다.

표 C–13 개인 정보 값의 SMS 해석

값	설명
0	제한 없음
1	제한됨
2	기밀
3	비밀
>= 4	정의되지 않음

RFC 822 `Sensitivity:` 헤더 행 값을 SMS 개인 정보 값으로 변환하는 데 사용되는 매핑은 다음 표 C–14에 나와 있습니다.

표 C–14 `Sensitivity` 헤더를 SMS 우선 순위 값으로 변환하기 위한 매핑

RFC 822 Sensitivity: 값	SMS 개인 정보 값
Personal	1(제한됨)
Private	2(기밀)
회사 기밀	3(비밀)

DEFAULT_SERVICE_TYPE

(**문자열**, *0 - 5***바이트**) 채널에 의해 생성되는 SMS 메시지와 연관시킬 서비스 유형입니다. 기본적으로 서비스 유형은 지정되지 않습니다(즉, 길이가 0인 문자열이 사용됨). 일반적인 몇 가지 서비스 유형으로는 `CMT`(Cellular Messaging), `CPT`(Cellular Paging), `VMN`(Voice Mail Notification), `VMA`(Voice Mail Alerting), `WAP`(Wireless Application Protocol) 및 `USSD`(Unstructured Supplementary Data Services)가 있습니다.

DEFAULT_SOURCE_ADDRESS

(**문자열**, *0 - 20***바이트**) 전자 메일에서 생성되는 SMS 메시지에 사용할 소스 주소입니다. `USE_HEADER_FROM=1`이면 전자 메일 메시지 발송자 주소가 이 옵션으로 지정한 값보다 우선한다는 것에 주의합니다. 기본적으로 값을 사용하지 않도록 `0`이 설정됩니다.

DEFAULT_SOURCE_NPI

(**정수**, *0 - 255*) 기본적으로 소스 주소에는 NPI 값으로 0이 할당됩니다. 이 옵션을 사용하면 0부터 255까지 범위의 대체 정수 값이 할당될 수 있습니다. 일반 NPI 값을 보여주는 표는 907 페이지 “DEFAULT_DESTINATION_NPI” 옵션 설명을 참조하십시오.

DEFAULT_SOURCE_TON

(**정수**, *0 - 255*) 기본적으로 소스 주소에는 TON 지정자 값으로 0이 할당됩니다. 이 옵션을 사용하면 0부터 255까지 범위의 대체 정수 값이 할당될 수 있습니다. 일반 TON 값을 보여주는 표는 908 페이지 "DEFAULT_DESTINATION_TON" 옵션 설명을 참조하십시오.

DEFAULT_VALIDITY_PERIOD

(**문자열**, *0 - 252* **바이트**) 기본적으로 SMS 메시지에는 상대적인 유효 기간이 지정되지 않으며 대신 SMSC의 기본값이 사용됩니다. 상대적인 다른 유효 기간을 지정하려면 이 옵션을 사용합니다. 값은 초, 분, 시 또는 일 단위로 지정할 수 있습니다. 다음 표 C–15에는 이 옵션의 여러 값에 대한 형식과 설명이 나와 있습니다.

표 C–15 DEFAULT_VALIDITY_PERIOD의 형식 및 값

형식	설명
nnn	암시적 초 단위(예: 604800)
*nnn*s	초 단위(예: 604800s)
*nnn*m	분 단위(예: 10080m)
*nnn*h	시간 단위(예: 168h)
*nnn*d	일 단위(예: 7d)

0, 0s, 0m, 0h 또는 0d 지정을 사용하여 SMSC의 기본 유효 기간을 선택할 수 있습니다. 즉, 0, 0s, 0m, 0h 또는 0d 지정이 사용되면 생성된 SMS 메시지의 유효 기간에 빈 문자열이 지정됩니다.

이 옵션은 UTC 형식의 값을 허용하지 않습니다.

DESTINATION_ADDRESS_NUMERIC

(*0* **또는** *1*) 전자 메일 봉투의 `To:` 주소에서 추출된 SMS 대상 주소의 숫자가 아닌 모든 문자를 스트라이프하려면 이 옵션을 사용합니다. 예를 들어, 다음 봉투의 `To:` 주소는

```
"(800) 555-1212"@sms.siroe.com
```

다음과 같이 줄어듭니다.

```
8005551212@sms.siroe.com
```

이 스트라이핑을 사용하려면 해당 옵션의 값을 1로 지정합니다. 기본적으로 이 스트라이핑은 사용 불가능하며 옵션 값 0에 해당합니다. 스트라이핑이 사용 가능한 경우 대상 주소 접두어가 913 페이지 "DESTINATION_ADDRESS_PREFIX" 옵션을 통해 추가되기 전에 스트라이핑이 수행됩니다.

DESTINATION_ADDRESS_PREFIX

(**문자열**) 경우에 따라 고정된 텍스트 문자열(예: "+")을 모든 SMS 대상 주소의 접두어로 사용해야 하는 경우도 있습니다. 이 옵션을 사용하면 이러한 접두어를 지정할 수 있습니다. 지정된 접두어는 모든 SMS 대상 주소(해당 접두어가 없는)에 추가됩니다. 912 페이지 "DESTINATION_ADDRESS_NUMERIC" 옵션에 의해 스트라이프되는 것을 방지하기 위해 이 옵션은 `DESTINATION_ADDRESS_NUMERIC` 옵션 다음에 적용됩니다.

PROFILE

(**문자열**) SMSC와 함께 사용할 SMS 프로필을 지정합니다. 가능한 값은 `GSM`, `TDMA` 및 `CDMA`입니다. 지정하지 않을 경우 GSM이 사용됩니다. 이 옵션은 909 페이지 "DEFAULT_PRIORITY" 및 910 페이지 "DEFAULT_PRIVACY"와 같은 다른 채널 옵션의 기본값을 선택하는 데만 사용됩니다.

USE_SAR

(0 **또는** *1)* 아주 큰 전자 메일을 여러 SMS 메시지로 분리해야 할 수 있습니다. 이 경우 개별 SMS 메시지는 선택적으로 SMS sar_ 필드를 사용하여 순서 지정 정보를 추가할 수 있습니다. 이로 인해 "분할된" SMS 메시지가 생성되며 수신 단말기는 이러한 SMS 메시지를 단일 SMS 메시지로 재어셈블할 수 있습니다. 해당되는 경우 `USE_SAR=1`을 지정하여 이 순서 지정 정보가 추가됨을 나타냅니다. 기본값은 순서 지정 정보를 추가하지 않는 것이며 이것은 `USE_SAR=0`에 해당합니다.

`USE_SAR=1`을 지정하면 914 페이지 "REVERSE_ORDER" 옵션이 무시됩니다.

C.3.3.4 SMPP 옵션

다음 옵션을 사용하면 SMPP 프로토콜 매개 변수를 지정할 수 있습니다. 문자열 "`ESME_`"로 시작하는 이름을 가진 옵션은 MTA가 ESME(External Short Message Entity)로 작동할 때, 즉 MTA가 SMS 메시지를 SMPP 서버의 관련 SMSC에 전송하기 위해 SMPP 서버에 바인드할 때 MTA를 식별하는 역할을 수행합니다.

ESME_ADDRESS_NPI

*(***정수***, 0 - 255)* 기본적으로 바인드 작업은 ESME NPI 값으로 알 수 없는 NPI를 나타내는 0을 지정합니다. 이 옵션을 사용하면 0부터 255까지 범위의 대체 정수 값이 할당될 수 있습니다. 일반 NPI 값을 보여주는 표는 907 페이지 "DEFAULT_DESTINATION_NPI" 옵션 설명을 참조하십시오.

ESME_ADDRESS_TON

*(***정수***, 0 - 255)* 기본적으로 바인드 작업은 ESME TON 값으로 0을 지정합니다. 이 옵션을 사용하면 0부터 255까지 범위의 대체 정수 값이 할당될 수 있습니다. 일반 TON 값을 보여주는 표는 908 페이지 "DEFAULT_DESTINATION_TON" 옵션 설명을 참조하십시오.

ESME_IP_ADDRESS

(**문자열**, *0 - 15* **바이트**) SMPP 서버에 바인드할 때 BIND PDU는 클라이언트의(즉, ESME) 주소 범위가 IP 주소임을 나타냅니다. 이것은 TON과 NPI를 각각 0x00 및 0x0d로 지정하는 방법으로 수행합니다. 그런 다음 주소 범위 필드의 값이 SMS 채널을 실행하는 호스트의 IP 주소로 설정됩니다. IP 주소를 점으로 구분된 십진수 형식(예: 127.0.0.1)으로 지정합니다.

ESME_PASSWORD

(**문자열**, *0 - 8***바이트**) SMPP 서버에 바인드할 때 비밀번호가 필요할 수 있습니다. 그럴 경우 이 옵션을 사용하여 비밀번호를 지정합니다. 기본적으로 길이가 0인 비밀번호 문자열이 제공됩니다.

ESME_SYSTEM_ID

(**문자열**, *0 - 15* **바이트**) SMPP 서버에 바인드할 때 MTA의 시스템 아이디가 제공될 수 있습니다. 기본적으로 시스템 아이디는 지정되지 않습니다(즉, 길이가 0인 문자열이 사용됨). 시스템 아이디를 지정하려면 이 옵션을 사용합니다.

ESME_SYSTEM_TYPE

(**문자열**, *0 - 12***바이트**) SMPP 서버에 바인드할 때 MTA의 시스템 유형이 제공될 수 있습니다. 기본적으로 시스템 유형은 지정되지 않습니다(즉, 길이가 0인 문자열이 사용됨).

MAX_PAGES_PER_BIND

(**정수**, >= *0*) 일부 SMPP 서버는 단일 바운드 세션 도중 전송되는 최대 SMS 메시지 수를 제한할 수 있습니다. 이러한 상황에서 이 옵션을 사용하면 단일 세션 동안에 전송할 최대 SMS 메시지 수를 지정할 수 있습니다. 지정된 한도에 도달하면 채널은 바인드 해제를 수행하고 TCP/IP 연결을 닫았다가 다시 연결한 후에 바인드를 다시 수행합니다.

기본적으로 MAX_PAGES_PER_BIND에 값 1024가 사용됩니다. 채널은 또한 ESME_RTHROTTLED 오류를 감지하고 이에 따라 채널의 단일 실행 동안에 MAX_PAGES_PER_BIND를 조정합니다.

REVERSE_ORDER

(*0* **또는** *1*) 전자 메일 메시지에서 둘 이상의 SMS 메시지를 생성할 경우 이러한 SMS 메시지를 순차적 순서(REVERSE_ORDER=0) 또는 역순차적 순서(REVERSE_ORDER=1)로 SMSC에 전송할 수 있습니다. 역순차적 순서는 수신 단말기에서 마지막으로 받은 메시지를 가장 먼저 표시할 경우에 유용합니다. 이 경우 마지막으로 받은 메시지는 전자 메일의 마지막 부분이 아니라 첫 번째 부분이 됩니다. 기본적으로 REVERSE_ORDER=1이 사용됩니다.

913 페이지 “USE_SAR”=1을 지정하면 이 옵션이 무시됩니다.

SMPP_MAX_CONNECTIONS

(**정수**, *1 - 50*) 이 옵션은 프로세스당 최대 동시 SMPP 연결 수를 제어합니다. 각 연결에 연관된 스레드가 있으므로 이 옵션은 또한 프로세스당 최대 "작업자" 스레드 수를 제한합니다. 기본적으로 `SMPP_MAX_CONNECTIONS=20`입니다.

SMPP_PORT

(**정수**, *1 - 65535*) SMPP 서버가 수신하는 TCP 포트는 이 옵션이나 `port` 채널 키워드를 사용하여 지정할 수 있습니다. 이 포트 번호는 이러한 두 기법 중 하나를 통해 지정해야 합니다. 두 기법 모두를 사용하여 지정할 경우 `SMPP_PORT` 옵션을 사용하여 설정한 값이 우선합니다. 이 옵션에는 기본값이 없습니다.

양방향 SMS의 경우 SMPP 중계를 위한 `LISTEN_PORT`와 동일한 포트인지 확인합니다.

SMPP_SERVER

(**문자열**, *1 - 252* **바이트**) 단방향 SMS의 경우 연결할 SMPP 서버의 IP 호스트 이름은 채널과 연관된 공식 호스트 이름입니다(즉, MTA 구성에서 채널 정의의 두 번째 행에 표시된 호스트 이름). 이 옵션을 사용하면 채널 정의에 지정된 이름을 무시하는 다른 호스트 이름이나 IP 주소를 지정할 수 있습니다. IP 주소를 지정할 때는 점으로 구분된 십진수 표기법(예: 127.0.0.1)을 사용합니다.

양방향 SMS의 경우 SMS 게이트웨이 서버의 호스트 이름이나 IP 주소를 가리키도록 설정합니다. SMPP 릴레이의 `LISTEN_INTERFACE_ADDRESS` 옵션을 사용할 경우 지정된 네트워크 인터페이스 주소와 연관된 호스트 이름이나 IP 주소를 사용해야 합니다.

TIMEOUT

(**정수**, *>= 2*) SMPP 서버에 데이터를 쓰거나 SMPP 서버로부터 데이터를 받기 위해 기다릴 때 기본적으로 30초의 시간 초과가 사용됩니다. 다른 시간 초과 값을 초 단위로 지정하려면 `TIMEOUT` 옵션을 사용합니다. 지정된 값은 최소 1초 이상이어야 합니다.

C.3.3.5 현지화 옵션

SMS 메시지 구성 시 SMS 채널은 SMS 메시지에 포함할 여러 고정 텍스트 문자열을 가집니다. 예를 들어, 이러한 문자열은 전자 메일의 `From:` 주소 및 `Subject:` 헤더 행을 소개합니다. 이 절에 설명된 채널 옵션을 사용하면 이러한 문자열의 버전을 다른 언어로 지정한 다음 채널의 기본 언어를 지정할 수 있습니다. 예 C–2에서는 옵션 파일의 언어 부분을 보여 줍니다.

예 C–2 채널 옵션 파일의 언어 지정 부분

```
LANGUAGE=default-language

[language=i-default]
```

예 C–2 채널 옵션 파일의 언어 지정 부분 *(계속)*

```
FROM_PREFIX=From:
SUBJECT_PREFIX=Subj:
CONTENT_PREFIX=Msg:
LINE_STOP= NO_MESSAGE=[no message]
REPLY_PREFIX=Re:

[language=en]
FROM_PREFIX=From:
SUBJECT_PREFIX=Subj:
CONTENT_PREFIX=Msg:
LINE_STOP=
NO_MESSAGE=[no message]
REPLY_PREFIX=Re:
  ...
```

각 `[language=x]` 블록 내에서 해당 언어와 관련된 현지화 옵션을 지정할 수 있습니다. 이 블록 안에 특정 옵션을 지정하지 않을 경우 해당 옵션의 전역 값이 사용됩니다. `[language=x]` 블록 외부에 지정된 현지화 옵션은 해당 옵션의 전역 값을 설정합니다.

아래 나열된 옵션의 경우 US-ASCII 또는 UTF-8 문자 세트를 사용하여 문자열 값을 지정해야 합니다. US-ASCII 문자 세트는 UTF-8 문자 세트의 특수한 경우입니다.

CONTENT_PREFIX

(**문자열**, *0 - 252* **바이트**) SMS 메시지에서 전자 메일 메시지의 내용 앞에 포함되는 텍스트 문자열입니다. 기본 전역 값은 US-ASCII 문자열 "`Msg:`"입니다.

DSN_DELAYED_FORMAT

*(***문자열**, *0-256***자***)* 전달 지연 알림에 대한 서식 지정 문자열입니다. 기본적으로 이 옵션에는 빈 문자열이 사용되며 이 경우 지연 알림이 SMS로 변환되지 않습니다. 이 옵션을 적용하려면 903 페이지 “GATEWAY_NOTIFICATIONS”가 `1`로 설정되어야 합니다. `GATEWAY_NOTIFICATIONS=0`일 경우 이 옵션은 무시됩니다.

DSN_FAILED_FORMAT

*(***문자열**, *0-256***자***)* 영구 전달 실패 알림의 서식 지정 문자열입니다. 이 옵션의 기본값은 다음 문자열입니다.

```
Unable to deliver your message to $a; no further delivery attempts will be
made.
```

실패한 알림이 변환되는 것을 방지하려면 이 옵션에 빈 문자열을 지정합니다. 이 옵션을 적용하려면 903 페이지 “GATEWAY_NOTIFICATIONS”가 `1`로 설정되어야 합니다. `GATEWAY_NOTIFICATIONS=0`일 경우 이 옵션은 무시됩니다.

DSN_RELAYED_FORMAT

*(**문자열**, 0-256**자**)* 릴레이 알림의 서식 지정 문자열입니다. 기본값은 다음 문자열입니다.

```
Your message to $a has been relayed to a messaging system which may not
provide a final delivery confirmation
```

중계 알림이 변환되는 것을 방지하려면 이 옵션에 빈 문자열을 지정합니다. 이 옵션을 적용하려면 903 페이지 “GATEWAY_NOTIFICATIONS”가 1로 설정되어야 합니다. GATEWAY_NOTIFICATIONS=0일 경우 이 옵션은 무시됩니다.

DSN_SUCCESS_FORMAT

*(**문자열**, 0-256**자**)* 성공한 전달 알림의 서식 지정 문자열입니다. 기본값은 다음 문자열입니다.

```
Your message to $a has been delivered
```

성공한 전달 알림이 변환되는 것을 방지하려면 이 옵션에 빈 문자열을 지정합니다. 이 옵션을 적용하려면 903 페이지 “GATEWAY_NOTIFICATIONS”가 1로 설정되어야 합니다. GATEWAY_NOTIFICATIONS=0일 경우 이 옵션은 무시됩니다.

FROM_FORMAT

(**문자열**, *0 - 252* **바이트**) SMS 메시지에 삽입할 메일 발송자 정보의 서식 지정을 위한 서식 지정 템플리트입니다. 기본 전역 값은 메일 발송자의 전자 메일 주소로 대체되는 US-ASCII 문자열 "$a"입니다. 919 페이지 “C.3.3.6 서식 지정 템플리트”를 참조하십시오.

FROM_NONE

(**문자열**, *0 - 252* **바이트**) 표시할 메일 발송자 주소가 없을 경우 SMS 메시지에 포함할 텍스트 문자열입니다. 기본 전역 값은 빈 문자열입니다.

사이트는 일반적으로 메일 발송자의 주소가 없는 전자 메일을 거부하므로 이 옵션은 거의 사용되지 않습니다.

LANGUAGE

(**문자열**, *0 - 40* **바이트**) 텍스트 문자열을 선택할 기본 언어 그룹입니다. 값을 지정하지 않을 경우 호스트의 기본 로켈 지정에서 언어가 파생됩니다. 호스트의 로켈 지정을 사용할 수 없거나 "C"에 해당할 경우 i-default가 사용됩니다. i-default는 “전세계 사용자를 대상으로 하는 영어 텍스트"에)

LINE_STOP

(**문자열**, *0 - 252* **바이트**) SMS 메시지에서 전자 메일로부터 추출한 행 사이에 포함할 텍스트 문자열입니다. 기본 전역 값은 US-ASCII 공백 문자(" ")입니다.

NO_MESSAGE

(**문자열**, *0 - 252* **바이트**) 전자 메일에 내용이 없음을 나타내기 위해 SMS 메시지에 포함할 텍스트 문자열입니다. 기본 전역 값은 US-ASCII 문자열 "[no message]"입니다.

SUBJECT_FORMAT

(**문자열**, *0 - 252* **바이트**) SMS 메시지에 표시할 Subject: 헤더 행 내용의 서식 지정을 위한 서식 지정 템플리트입니다. 이 옵션의 전역 기본 값은 US-ASCII 문자열 "($s)"입니다. 자세한 내용은 919 페이지 “C.3.3.6 서식 지정 템플리트”를 참조하십시오.

Subject: 헤더 행이 없거나 해당 헤더 행의 내용이 비어 있는 경우의 처리 방법은 SUBJECT_NONE 옵션을 참조하십시오.

SUBJECT_NONE

(**문자열**, *0 - 252* **바이트**) 원본 전자 메시지에 Subject: 헤더 행이 없거나 Subject: 헤더 행의 값이 빈 문자열일 경우 표시할 텍스트 문자열입니다. 이 옵션의 기본 전역 값은 빈 문자열입니다.

DEBUG

(**정수, 비트 마스크**) 디버그 출력을 사용 가능하게 합니다. 기본값은 경고 및 오류 메시지를 선택하는 6입니다. 0이 아닌 모든 값은 채널 정의에서 master_debug를 지정하는 것과 동일하게 채널 자체에 대한 디버그 출력을 사용 가능하게 합니다.표 C–16에는 DEBUG 비트 마스크의 비트 값이 정의되어 있습니다.

표 C–16 DEBUG 비트 마스크

비트	값	설명
0-31	-1	매우 자세한 출력
0	1	정보 메시지
1	2	경고 메시지
3	4	오류 메시지
3	8	서브루틴 호출 추적
4	16	해시 테이블 진단
5	32	I/O 진단, 수신
6	64	I/O 진단, 전송
7	128	SMS에서 전자 메일로의 변환 진단(모바일 원본 및 SMS 알림)

표 C–16 DEBUG 비트 마스크 *(계속)*

비트	값	설명
8	256	PDU 진단, 헤더 데이터
9	512	PDU 진단, 본문 데이터
10	1024	PDU 진단, 유형 길이 값 데이터
11	2048	옵션 처리(모든 옵션 설정을 로그 파일로 보냄)

C.3.3.6 서식 지정 템플리트

917 페이지 “FROM_FORMAT”, 918 페이지 “SUBJECT_FORMAT” 및 모든 `DSN_*` 채널 옵션으로 지정하는 서식 지정 템플리트는 리터럴 텍스트 및 대체 시퀀스의 조합을 포함하는 UTF-8 문자열입니다. 다음과 같은 샘플 전자 메일 주소를 가정합니다.

```
Jane Doe <user@siroe>
```

다음 표 C–17에서는 인식되는 대체 시퀀스를 보여 줍니다.

표 C–17 대체 시퀀스

시퀀스	설명
`$a`	메일 발송자의 전자 메일 주소에 있는 로컬 및 도메인 부분(예: "user@siroe")으로 교체됩니다.
`$d`	메일 발송자의 전자 메일 주소에 있는 도메인 부분(예: "domain")으로 교체됩니다.
`$p`	메일 발송자의 전자 메일 주소에 있는 구 부분(예: "Jane Doe")으로 교체됩니다.
`$s`	`Subject:` 헤더 행의 내용으로 교체합니다.
`$u`	메일 발송자의 전자 메일 주소에 있는 로컬 부분(예: "user")으로 교체됩니다.
`\x`	리터럴 문자 "x"로 교체됩니다.

예를 들어, 다음 서식 지정 템플리트는

```
From: $a
```

다음 텍스트 문자열을 생성합니다.

```
From: user@siroe
```

다음 구조는

```
${xy:alternate text}
```

시퀀스 x와 연관된 텍스트로 대체하는 데 사용할 수 있습니다. 해당 텍스트가 빈 문자열인 경우 시퀀스 y와 연관된 텍스트가 대신 사용됩니다. 또한 위 구조는 해당 테스트가 빈 문자열인 경우 대체 텍스트로 대체하는 데 사용할 수 있습니다. 예를 들어, 다음과 같은 서식 지정 템플리트를 가정합니다.

```
From: ${pa:unknown sender}
```

구 부분을 포함하는 메일 발송자의 전자 메일 주소의 경우

```
John Doe <jdoe@siroe.com>
```

이 템플리트에서는 다음을 생성합니다.

```
From: John Doe
```

그러나 구가 없는 다음 주소의 경우

```
jdoe@siroe.com
```

이 템플리트는 다음을 생성합니다.

```
From: jdoe@siroe.com
```

빈 메일을 보낸 발송자 주소의 경우에는 다음을 생성합니다.

```
From: unknown sender
```

C.3.4 SMS 채널 추가

둘 이상의 SMS 채널을 가지도록 MTA를 구성할 수 있습니다. 이는 일반적으로 다음 두 가지 이유 때문입니다.

1. 다른 SMPP 서버와 통신하려는 경우

   이것은 매우 간단하여 단순히 추가 SMS 채널을 구성에 추가하면 됩니다. 이 과정에서 추가 SMS 채널에 (a)다른 채널 이름을 제공하고 (b) 다른 호스트 이름을 연관시킵니다. 예를 들면 다음과 같습니다.

   ```
   sms_mway port 55555 threaddepth 20
   smpp.siroe.com

   sms_ace port 777 threaddepth 20
   sms.ace.net
   ```

   새로운 다시 쓰기 규칙은 필요하지 않습니다. 직접적으로 일치되는 다시 쓰기 규칙이 없을 경우 Messaging Sever는 연관된 호스트 이름을 가진 채널을 찾습니다. 예를 들어, `user@host.domain`으로 표시될 경우 Messaging Sever는 "host.domain"이라는 이름의 채널을 찾습니다. 이러한 채널이 발견되면 메시지는 해당 채널로 라우팅됩니다. 그렇지 않을 경우 서버는 다시 쓰기 규칙에서 ".domain"을 찾고, 없을 경우 점(".") 규칙을 찾습니다. 다시 쓰기 규칙에 대한 자세한 내용은 11 장을 참조하십시오.

2. 다른 채널 옵션을 사용하여 동일한 SMPP 서버와 통신하려는 경우

   다른 채널 옵션을 사용하여 동일한 SMPP 서버와 통신하려면 각 채널 정의의 915 페이지 “SMPP_SERVER” 채널 옵션에서 동일한 SMPP 서버를 지정합니다.

   이 기법을 사용해야 하는 이유는 두 개의 다른 채널이 동일한 공식 호스트 이름(즉, 채널 정의의 두 번째 행에 나열된 호스트 이름)을 가질 수 없기 때문입니다. 이러한 채널이 동일한 SMPP 서버와 통신할 수 있게 하려면 해당 채널 옵션 파일의 SMPP_SERVER에서 동일한 SMPP 서버를 지정하는 별개의 채널 두 개를 정의합니다.

   예를 들어, 다음 채널 정의와

```
sms_mway_1 port 55555 threaddepth 20
SMS-DAEMON-1

sms_mway_2 port 55555 threaddepth 20
SMS-DAEMON-2
```

   다음 다시 쓰기 규칙이 있다고 가정합니다.

```
sms-1.siroe.com $u%sms-1.siroe.com@SMS-DAEMON-1
sms-2.siroe.com $U%sms-2.siroe.com@SMS-DAEMON-2
```

   이 경우에 동일한 SMPP 서버를 사용하기 위해 이러한 두 채널은 각각 해당 채널 옵션 파일에서 915 페이지 “SMPP_SERVER”=smpp.siroe.com을 지정합니다.

C.3.5 전달 재시도 빈도 조정

SMPP 서버에 도달할 수 없는 경우와 같이 일시적인 오류로 인해 SMS 메시지를 전달하지 못하면 전자 메일은 전달 대기열에 남아 있다가 나중에 다시 시도됩니다. 따로 구성하지 않은 경우에는 작업 제어기에서 1시간 동안 전달을 다시 시도하지 않습니다. 그러나 SMS 메시징의 경우 1시간은 기다리는 시간으로 너무 깁니다. 이러한 경우에는 SMS 채널과 함께 backoff 채널 키워드를 사용하여 전달 시도에 보다 빠른 일정을 지정하는 것이 좋습니다. 예를 들면 다음과 같습니다.

```
sms_mway port 55555 threaddepth 20 \
  backoff "pt2m" "pt5m" "pt10m" "pt30m" notices 1
smpp.siroe.com
```

위 설정에서는 첫 시도가 있은 후 2분, 5분 및 10분 후에 한 번씩 재전달이 시도되며 이러한 시도가 모두 실패할 경우 마지막으로 30분마다 재전달이 시도됩니다. notices 1 채널 키워드는 하루가 지나도록 메시지를 전달할 수 없는 경우 해당 메시지를 전달할 수 없는 것으로 반환합니다.

C.3.6 샘플 단방향 구성(MobileWay)

MTA SMS 채널은 모든 SMPP V3.4 호환 SMPP 서버에서 사용할 수 있습니다. 이 절에서는 구성 예를 제시하기 위해 MobileWay SMPP 서버에서 사용하도록 SMS 채널을 구성하는 방법에 대해 설명합니다. MobileWay(http://www.mobileway.com/ (http://www.mobilway.com))는 전역 데이터 및 SMS 연결의 주요 공급업체입니다. MobileWay를 통해 SMS 트래픽을 라우팅하면 전세계 대부분의 주요 SMS 네트워크상에 있는 SMS 가입자에 도달할 수 있습니다.

MobileWay를 통해 SMPP 계정을 요청할 때는 다음 질문에 답하라는 메시지가 나타날 수 있습니다.

- SMPP 클라이언트의 IP 주소: 인터넷 상의 다른 도메인에 표시된 것처럼 Messaging Server 시스템의 IP 주소를 제공합니다.
- 기본 유효 기간: 전송하는 SMS 메시지에 유효 기간이 지정되지 않을 경우 MobileWay에서 사용하는 SMS 유효 기간입니다. 이 유효 기간이 만료될 때까지 전달할 수 없는 SMS 메시지는 무시됩니다. 적절한 값(예: 2일, 7일 등)을 제공합니다.
- 창 크기: SMPP 클라이언트가 전송하는 최대 SMS 메시지 수입니다. SMPP 클라이언트는 이 수를 초과할 경우 추가 SMS 메시지를 전송하는 것을 중지하고 SMPP 서버의 응답을 기다립니다. 1의 메시지 값을 제공해야 합니다.
- 표준 시간대: Messaging Server 시스템이 작동하는 표준 시간대를 지정합니다. 표준 시간대는 GMT에서 오프셋으로 지정해야 합니다.
- 시간 초과: 단방향 SMS 메시징과는 관련이 없습니다.
- 아웃바운드 요청에 대한 IP 주소 및 TCP 포트: 단방향 SMS 메시징과는 관련이 없습니다.

위 질문에 대한 대답을 MobileWay에 제공하면 SMPP 서버와 통신하는 데 필요한 SMPP 계정 및 정보가 다음과 같이 제공됩니다.

```
Account Address: a.b.c.d:p
Account Login: system-id
Account Passwd: secret
```

Account Address 필드는 연결할 MobileWay SMPP 서버의 IP 주소, `a.b.c.d` 및 TCP 포트 번호, P입니다. 이러한 값을 915 페이지 “SMPP_SERVER” 및 915 페이지 “SMPP_PORT” 채널 옵션에 사용합니다. 계정 로그인 및 비밀번호는 각각 914 페이지 “ESME_SYSTEM_ID” 및 914 페이지 “ESME_PASSWORD” 채널 옵션에 사용할 값입니다. 이 정보를 사용하여 채널 옵션 파일에 다음을 포함해야 합니다.

```
SMPP_SERVER=a.b.c.d
SMPP_PORT=p
ESME_SYSTEM_ID=system-id
ESME_PASSWORD=secret
```

이제 MobileWay와 상호 작용하기 위해 다음과 같은 두 개의 추가 옵션 설정이 필요합니다.

```
ESME_ADDRESS_TON=0x01
DEFAULT_DESTINATION_TON=0x01
```

`imta.cnf` 파일의 다시 쓰기 규칙은 다음과 같이 나타날 수 있습니다.

```
sms.your-domain $u@sms.your-domain
```

또한 `imta.cnf` 파일의 채널 정의는 다음과 같이 나타날 수 있습니다.

```
sms_mobileway
sms.your-domain
```

일단 채널 옵션 파일, 다시 쓰기 규칙 및 채널 정의가 제대로 되면 테스트 메시지를 보낼 수 있습니다. MobileWay는 다음과 같은 형식의 국제 주소 지정을 요구합니다.

```
+<country-code><subscriber-number>
```

예를 들어, 가입자 번호가 (800) 555-1212인 북미 가입자에게 테스트 메시지를 보내려면 전자 메일 주소를 다음과 같이 지정합니다.

```
+18005551212@sms.your-domain
```

C.3.6.1 디버깅

채널을 디버그하려면 채널 정의에서 `master_debug` 채널 키워드를 지정합니다. 예를 들면 다음과 같습니다.

```
sms_mway port 55555 threaddepth 20 \
backoff "pt2m" "pt5m" "pt10m" "pt30m" notices 1 master_debug
```

`master_debug` 채널 키워드를 사용하면 채널 작업에 대한 기본 진단 정보가 채널의 로그 파일에 출력됩니다. 채널에서 수행한 SMPP 트랜잭션에 대한 자세한 진단 정보를 원할 경우에는 또한 해당 채널의 옵션 파일에서 다음을

```
DEBUG=-1
```

지정합니다.

C.3.7 양방향 SMS를 위한 SMS 채널 구성

SMS 채널 구성에 대한 일반 지침은 이전 896 페이지 “C.3 SMS 채널 구성” 절부터 설명된 항목을 참조하십시오. 다음 표 C–18에 나열된 사항을 제외하고 SMS 채널을 마치 원격 SMSC에 직접 연결하는 것처럼 구성합니다.

표 C–18 양방향 구성 예외

예외	설명
master 채널 키워드	master 채널 키워드를 제거합니다(있을 경우). SMS 채널 구성에 더 이상 필요하지 않습니다.
SMPP_SERVER	SMS 게이트웨이 서버의 호스트 이름이나 IP 주소를 가리키도록 설정합니다. SMPP 릴레이의 LISTEN_INTERFACE_ADDRESS 옵션(935 페이지 "C.5.7 구성 옵션" 참조)을 사용할 경우 지정된 네트워크 인터페이스 주소와 연관된 호스트 이름이나 IP 주소를 사용해야 합니다.
SMPP_PORT	SMPP 중계(932 페이지 "C.5.5.2 SMPP 중계" 참조)를 인스턴스화하는 데 사용되는 LISTEN_PORT 설정과 동일한 TCP 포트가 사용됩니다.
DEFAULT_SOURCE_ADDRESS	값을 선택한 다음 이 주소를 다시 게이트웨이 SMPP 서버로 라우팅하도록 원격 SMSC를 구성합니다. SMS 채널의 옵션 파일에서 이 옵션을 사용하여 선택한 값을 지정합니다.
GATEWAY_PROFILE	게이트웨이 프로필 이름과 일치하도록 설정합니다. 931 페이지 "C.5.5.1 게이트웨이 프로필"을 참조하십시오.
USE_HEADER_FROM	0으로 설정합니다.

다른 모든 채널은 SMS 채널 설명서에 설명된 대로 구성되어야 합니다.

929 페이지 "C.5.1 양방향 SMS 라우팅 설정"에 언급된 대로 원격 SMSC의 경우 LISTEN_PORT 옵션에 지정된 TCP 포트 번호를 사용하여 DEFAULT_SOURCE_ADDRESS 채널 옵션에 정의된 SMS 주소를 게이트웨이의 SMPP 서버에 라우팅하도록 구성해야 합니다. LISTEN_PORT를 지정하는 방법은 932 페이지 "C.5.5.3 SMPP 서버"를 참조하십시오.

여러 SMS 채널에서 동일한 SMPP 중계를 사용할 수 있습니다. 마찬가지로 여러 SMS 채널의 SMS 응답과 알림을 처리하기 위해 하나의 SMPP 서버 또는 게이트웨이만 필요합니다. 여러 중계, 서버 및 게이트웨이 프로필을 구성하는 기능은 구성 옵션을 통해 여러 다른 사용 특성을 구현하는데 필요합니다.

C.4 SMS 게이트웨이 서버 작동 이론

SMS 게이트웨이 서버는 모바일에서 전송된 SMS 메시지를 정확한 전자 메일 주소와 일치시키는 기법을 통해 양방향 SMS를 용이하게 합니다. 이 절은 다음과 같은 SMS 게이트웨이 서버 항목으로 구성되어 있습니다.

- 925 페이지 "C.4.1 SMS 게이트웨이 서버 기능"
- 925 페이지 "C.4.2 SMPP 중계 및 서버의 동작"
- 926 페이지 "C.4.3 원격 SMPP에서 게이트웨이 SMPP로의 통신"
- 927 페이지 "C.4.4 SMS 중계 및 알림 처리"

C.4.1 SMS 게이트웨이 서버 기능

SMS 게이트웨이 서버는 SMPP 중계와 서버의 기능을 동시에 수행합니다. SMS 게이트웨이 서버는 각 기능의 여러 "인스턴스화"를 가지도록 구성할 수 있습니다. 예를 들어, 각각 다른 TCP 포트나 네트워크 인터페이스를 수신하고 다른 원격 SMPP 서버로 중계하는 세 개의 다른 SMPP 중계를 가지도록 구성할 수 있습니다. 마찬가지로 각각 TCP 포트 및 네트워크 인터페이스의 다른 조합을 수신하는 네 개의 다른 SMPP 서버를 가지도록 SMS 게이트웨이 서버를 구성할 수 있습니다.

SMS 게이트웨이 서버는 SMS 메시지를 전자 메일로 보내기 위한 0개 이상의 게이트웨이 프로필을 가질 수 있습니다. 각 게이트웨이 프로필은 해당 프로필과 일치하는 대상 SMS 주소, SMS 메시지에서 대상 전자 메일 주소를 추출하는 방법, SMS에서 전자 메일로의 변환 프로세스가 가진 다양한 특성 등을 설명합니다. SMS 중계 또는 서버를 통해 SMS 게이트웨이 서버에 제공된 각 SMS 메시지는 각 프로필과 비교됩니다. 일치하는 항목이 발견되면 해당 메시지는 전자 메일로 라우팅됩니다.

마지막으로 게이트웨이 프로필은 또한 이전 ETM(Email-To-Mobile) 메시지에 응답하여 원격 SMSC가 반환한 알림 메시지를 처리하는 방법을 설명합니다.

C.4.2 SMPP 중계 및 서버의 동작

SMPP 중계로 작동할 때 SMS 게이트웨이 서버는 로컬 SMPP 클라이언트의 모든 요청을 원격 SMPP 서버로 중계한 다음 원격 서버의 응답을 다시 중계하면서 가능한 투명하게 작업을 시도합니다. 그러나 다음 두 가지 경우는 예외입니다.

- 로컬 SMPP 클라이언트가 구성된 게이트웨이 프로필 중 하나와 일치하는 SMS 대상 주소를 가진 메시지를 전송하면 전송된 SMS 메시지는 직접 전자 메일로 되돌려 보내집니다. 즉, SMS 메시지는 원격 SMPP 서버로 중계되지 않습니다.
- 로컬 또는 원격 SMPP 클라이언트가 이전에 SMPP 중계에 의해 생성된 고유한 SMS 소스 주소와 일치하는 SMS 대상 주소를 가진 메시지를 전송하면 해당 SMS 메시지는 이전에 중계된 메시지에 대한 응답입니다. 이 응답은 원본 메시지의 발송자에게 다시 전송됩니다.

일반적으로 SMS 게이트웨이 서버는 생성되는 고유한 SMS 소스 주소가 게이트웨이 프로필 중 하나와 일치하도록 구성됩니다.

주 – SMS 게이트웨이 서버의 SMPP 중계는 정규화된 Sun Java System SMPP 클라이언트, 즉 Sun Java System Messaging Server의 SMS 채널에서만 사용하도록 되어 있습니다. 따라서 임의의 SMPP 클라이언트에서는 사용되지 않습니다.

SMPP 서버로 작동할 때 SMS 게이트웨이 서버는 다음과 같은 세 가지 경우에 SMS 메시지를 전자 메일로 전송합니다.

- SMS 메시지가 모바일에서 전송되었으며 게이트웨이 프로필과 일치할 경우

- SMS 메시지가 모바일에서 전송되었으며 SMS 대상 주소가 이전에 생성된 고유한 SMS 소스 주소와 일치할 경우
- SMS 메시지가 이전에 SMS 게이트웨이 서버의 SMPP 릴레이에 의해 릴레이된 ETM(Email-To-Mobile) 메시지에 해당하는 SMS 알림인 경우

다른 모든 SMS 메시지는 SMPP 서버에 의해 거부됩니다.

C.4.3 원격 SMPP에서 게이트웨이 SMPP로의 통신

원격 SMPP 클라이언트는 PDU(Protocol Data Unit)를 사용하여 게이트웨이 SMPP 서버와 통신합니다. 원격 SMPP 클라이언트는 게이트웨이 SMPP 서버가 응답하는 요청 PDU를 보냅니다. 게이트웨이 SMPP 서버는 동기식으로 작동합니다. 게이트웨이 SMPP 서버는 연결된 원격 SMPP 클라이언트의 다음 요청 PDU를 처리하기 전에 요청 PDU에 대한 응답을 완료합니다.

다음 표 C–19에는 게이트웨이 SMPP 서버가 처리하는 요청 PDU와 게이트웨이 SMPP 서버의 응답이 나열되어 있습니다.

표 C–19 SMPP 서버 PDU(Protocol Data Unit)

요청 PDU	SMPP 서버 응답
BIND_TRANSMITTERBIND_TRANSCEIVERUNBIND	적절한 응답 PDU로 응답합니다. 인증 자격 증명은 무시됩니다.
OUTBIND	게이트웨이 SMPP 서버가 BIND_RECEIVER PDU를 되돌려 보냅니다. 표시된 인증 자격 증명은 무시됩니다.
SUBMIT_SMDATA_SM	고유한 SMS 소스 주소나 게이트웨이 프로필의 SELECT_RE 설정을 사용하여 대상 SMS 주소와 일치시키려고 시도합니다. 둘 다 일치하지 않을 경우 ESME_RINVDSTADR 오류와 함께 PDU가 거부됩니다.
DELIVER_SM	기록 레코드에서 대상 SMS 주소나 수신 확인된 메시지 아이디를 찾으려고 시도합니다. 둘 다 일치하지 않을 경우 ESME_RINVMSGID 오류를 반환합니다.
BIND_RECEIVER	지원되지 않습니다. ESME_RINVCMDID 오류와 함께 GENERIC_NAK PDU를 반환합니다.
SUBMIT_MULTI	지원되지 않습니다. ESME_RINVCMDID 오류와 함께 GENERIC_NAK PDU를 반환합니다.
REPLACE_SM	지원되지 않습니다. ESME_RINVCMDID 오류와 함께 GENERIC_NAK PDU를 반환합니다.
CANCEL_SM	지원되지 않습니다. ESME_RINVCMDID 오류와 함께 GENERIC_NAK PDU를 반환합니다.
QUERY_SM	지원되지 않습니다. ESME_RINVCMDID 오류와 함께 GENERIC_NAK PDU를 반환합니다.
QUERY_LAST_MSGS	지원되지 않습니다. ESME_RINVCMDID 오류와 함께 GENERIC_NAK PDU를 반환합니다.

표 C–19 SMPP 서버 PDU(Protocol Data Unit) (계속)

요청 PDU	SMPP 서버 응답
QUERY_MSG_DETAILS	지원되지 않습니다. ESME_RINVCMDID 오류와 함께 GENERIC_NAK PDU를 반환합니다.
ENQUIRE_LINK	ENQUIRE_LINK_RESP PDU를 반환합니다.
ALERT_NOTIFICATION	허용되지만 무시됩니다.

C.4.4 SMS 중계 및 알림 처리

SMS 게이트웨이 서버는 해당 SMPP 중계를 통해 중계된 각 SMS 메시지의 기록 레코드를 유지 관리합니다. 기록 데이터를 사용하는 이유는 전자 메일 메시지를 SMS로 전송할 때 일반적으로 메일 발송자의 전자 메일 주소를 SMS 소스 주소로 변환할 수 없기 때문입니다. 이 경우 모든 SMS 응답과 알림이 이 SMS 소스 주소로 전송되므로 문제가 발생합니다. 이 문제는 자동으로 생성된 고유한 SMS 소스 주소를 중계된 메시지에서 사용하는 방법으로 해결합니다. 그런 다음 원격 SMSC는 이러한 SMS 소스 주소를 다시 게이트웨이 SMPP 서버로 라우팅하도록 구성됩니다.

기록 데이터는 메시지 아이디와 생성된 고유한 SMS 소스 주소의 메모리 내장 해시 테이블로 표시됩니다. 또한 이 데이터는 연관된 전자 메일 원본 데이터와 함께 디스크에 저장됩니다. 이러한 디스크 기반 저장소는 일련의 파일로써 각 파일은 트랜잭션의 HASH_FILE_ROLLOVER_PERIOD 기간(초 단위이며 기본값은 30분)을 나타냅니다. 각 파일은 RECORD_LIFETIME 기간(초 단위이며 기본값은 3일) 동안 보관됩니다. 기록 데이터의 메모리 내장 및 디스크 내장 자원 요구 사항에 대한 자세한 내용은 **Sun Java Communications Suite 5 Deployment Planning Guide**를 참조하십시오.

각 레코드는 다음 세 가지 구성 요소를 가집니다.

- 전자 메일 원본 데이터(예: 봉투의 From: 및 To: 주소). MTA SMS 채널에서 메시지를 전송할 때 이 데이터를 제공합니다.
- SMPP 중계에 의해 생성되며 중계된 SMS 메시지에 삽입되는 고유한 SMS 소스 주소
- 원격 SMSC의 SMPP 서버에서 전송을 수락할 때 반환하는 수신 확인된 결과 메시지 아이디

C.4.4.1 SMS 응답의 라우팅 프로세스

게이트웨이 SMPP 중계 및 서버는 기록 레코드를 사용하여 SMS 응답, 알림 및 모바일에서 전송된 메시지를 처리합니다. SMS 메시지가 SMPP 중계 또는 서버에 제공될 때 다음 라우팅 프로세스를 따릅니다.

1. SMPP 중계가 이전에 생성한 일치하는 고유한 SMS 소스 주소가 있는지 확인하기 위해 SMS 대상 주소가 기록 레코드와 비교됩니다. 일치하는 항목이 발견된 경우 단계 6으로 이동합니다.

2. 일치하는 항목이 없지만 메시지가 SMS 알림(SMPP DELIVER_SM PDU)일 경우 수신 확인된 메시지 아이디(존재할 경우)가 기록 레코드와 비교됩니다. 일치하는 항목이 발견된 경우 단계 8로 이동합니다. [SMS 게이트웨이 서버는 실제로 이러한 항목을 SMPP 릴레이 또는 SMPP 서버에 제공하는 것을 허용합니다.]
3. 일치하는 항목이 없을 경우 대상 SMS 주소가 구성된 각 게이트웨이 프로필의 SELECT_RE 옵션 표현식과 비교됩니다. 일치하는 항목이 발견된 경우 단계 9로 이동합니다.
4. 일치하는 항목이 없고 SMS 메시지가 게이트웨이 SMPP 중계에 제공된 경우 메시지는 원격 SMPP 서버로 중계됩니다.
5. 일치하는 항목이 없고 SMS 메시지가 게이트웨이 SMPP 서버에 제공된 경우 해당 메시지가 잘못된 것으로 결정되어 SMPP 응답 PDU에서 오류 응답이 반환됩니다. 전자 메일에서 SMS로의 경우 최종적으로 NDN(Non Delivery Notification)이 생성됩니다.
6. 일치하는 고유한 SMS 소스 주소가 발견된 경우 SMS 메시지를 추가로 검사하여 응답 또는 알림 메시지인지 확인합니다. 알림 메시지가 되려면 수신 확인된 메시지 아이디가 있는 SUBMIT_SM PDU여야 합니다. 그렇지 않을 경우 해당 메시지는 응답으로 간주됩니다.
7. 응답일 경우 SMS 메시지는 기록 레코드의 원본 전자 메일 정보를 사용하여 전자 메일로 변환됩니다.
8. 알림일 경우 SMS 메시지는 RFC 1892-1894에 따라 전자 메일 DSN(Delivery Status Notification)으로 변환됩니다. 이 때 원본 전자 메일 메시지의 ESMTP NOTIFY 플래그(RFC 1891)가 적용됩니다. 예를 들어, SMS 메시지가 "성공" DSN이지만 원본 전자 메일 메시지가 "실패" 알림만 요청한 경우 SMS 알림은 무시됩니다.
9. 대상 SMS 주소가 구성된 게이트웨이 프로필의 SELECT_RE 옵션과 일치할 경우 SMS 메시지는 모바일에서 전송된 메시지로 간주되며 해당 게이트웨이 프로필에 대한 PARSE_RE_n 규칙에 따라 다시 전자 메일로 변환됩니다. 변환이 실패할 경우 SMS 메시지가 잘못된 것이므로 오류 응답이 반환됩니다.

C.5 SMS 게이트웨이 서버 구성

이 절에서는 ETM(Email-To-Mobile) 및 MTE(Mobile-To-Email) 기능을 모두 사용하도록 SMS 게이트웨이 서버를 설정하는 방법에 대해 설명합니다. 이 절은 다음 내용으로 구성되어 있습니다.

C.5.1 양방향 SMS 라우팅 설정

MTA와 SMSC 간에 양방향 전자 메일 및 SMS 라우팅을 설정하려면 다음 3단계 과정을 수행하는 것이 좋습니다.

- 929 페이지 “C.5.1.1 SMS 주소 접두어 설정”– SMS 주소 접두어를 선택합니다. 10자 이하인 모든 접두어를 사용할 수 있습니다.
- 929 페이지 “C.5.1.2 게이트웨이 프로필 설정”– SMS 게이트웨이 서버에 사용하기 위해 접두어를 반전시킵니다(게이트웨이 프로필을 설정함으로써).
- 930 페이지 “C.5.1.3 SMSC 구성”– 접두어로 시작하는 SMS 대상 주소를 SMS 게이트웨이 SMPP 서버로 라우팅하도록 SMSC를 구성합니다. 모바일에서 전송한 전자 메일은 접두어만 가집니다. 응답과 알림은 정확하게 10개의 십진수가 뒤에 오는 접두어를 가집니다.

C.5.1.1 SMS 주소 접두어 설정

MTA SMS 채널에 의해 생성되는 소스 SMS 주소는 선택된 SMS 주소 접두어와 일치하도록 설정해야 합니다. 그러려면 다음을 설정합니다.

- MTA SMS 채널 옵션

  `USE_HEADER_FROM=0`

  `DEFAULT_SOURCE_ADDRESS=`*prefix*

  첫 번째 설정은 채널이 전자 메일에 포함된 정보에서 SMS 소스 주소를 설정하지 않도록 합니다. 두 번째 설정은 다른 소스에서 설정되지 않은 경우 SMS 소스 주소가 선택된 접두어로 설정되도록 합니다.
- 해당 접두어를 전자 메일을 수락하여 라우팅할 SMS 대상 주소로 인식합니다. 그러려면 `SELECT_RE` 게이트웨이 프로필 옵션을 다음과 같이 지정합니다.

  `SELECT_RE=`*prefix*

C.5.1.2 게이트웨이 프로필 설정

릴레이된 모든 SMS 소스 주소가 고유하도록 SMS 게이트웨이 서버의 게이트웨이 프로필을 설정해야 합니다. 이것은 기본 설정이지만 게이트웨이 프로필 옵션 `MAKE_SOURCE_ADDRESSES_UNIQUE=1`을 지정하여 명시적으로 설정할 수 있습니다. 결과적으로 다음 형식의 중계된 SMS 소스 주소가 만들어집니다.

`prefixnnnnnnnnnn`

여기서 nnnnnnnnnn은 고유한 10자리 십진수가 됩니다.

C.5.1.3 SMSC 구성

마지막으로 접두어(단순히 접두어만 또는 접두어와 10자리 숫자)와 일치하는 모든 SMS 대상 주소를 SMS 게이트웨이 서버의 SMPP 서버로 라우팅하도록 SMSC를 구성해야 합니다. 이러한 라우팅을 위한 정규식은 다음과 유사합니다.

```
prefix([0-9]{10,10}){0,1}
```

여기서 prefix는 DEFAULT_SOURCE_ADDRESS의 값이고 [0-9]는 10자리 숫자에 허용되는 값이며 {10, 10}은 최소한 10자리 및 최대한 10자리 숫자가 존재한다는 것을 지정하고 마지막으로 {0, 1}은 0개 또는 1개의 10자리 숫자가 존재할 수 있다는 것을 지정합니다.

C.5.2 SMS 게이트웨이 서버 활성화/비활성화

- SMS 게이트웨이 서버를 사용하려면 configutil 매개 변수 local.smsgateway.enable의 값을 1로 설정해야 합니다. 다음 구성 유틸리티 명령을 사용하여 이 값을 설정합니다.

  ```
  # configutil -o local.smsgateway.enable -v 1
  ```

- 게이트웨이 서버를 사용하지 않으려면 다음 명령을 사용하여 local.smsgateway.enable의 값을 0으로 설정합니다.

  ```
  # configutil -o local.smsgateway.enable -v 0
  ```

C.5.3 SMS 게이트웨이 서버 시작 및 중지

SMS 게이트웨이 서버가 사용 가능하게 된 후 다음 명령을 사용하여 서버를 시작 및 중지할 수 있습니다.

```
# start-msg sms
```

및

```
# stop-msg sms
```

C.5.4 SMS 게이트웨이 서버 구성 파일

SMS 게이트웨이 서버가 작동하려면 구성 파일이 필요합니다. 구성 파일은 UTF-8을 사용하여 인코딩되는 유니코드 텍스트 파일이며 ASCII 텍스트 파일이 될 수 있습니다. 구성 파일의 이름은 다음과 같아야 합니다.

installation-directory/config/sms_gateway.cnf

구성 파일의 각 옵션 설정은 다음 형식을 가집니다.

```
option-name=option-value
```

옵션 그룹의 일부인 옵션은 다음 형식으로 표시됩니다.

```
[group-type=group-name]
option-name-1=option-value-1
option-name-2=option-value-2
...
option-name-n=option-value-n
```

C.5.5 게이트웨이 서버에서 ETM(Email-To-Mobile) 구성

양방향 SMS의 ETM(Email-To-Mobile) 부분을 구현하려면 다음을 구성해야 합니다.

- 931 페이지 "C.5.5.1 게이트웨이 프로필"
- 932 페이지 "C.5.5.2 SMPP 중계"
- 932 페이지 "C.5.5.3 SMPP 서버"

C.5.5.1 게이트웨이 프로필

ETM(Email-To-Mobile) 게이트웨이 프로필을 구성하려면 다음 단계를 수행합니다.

▼ ETM(Email-To-Mobile) 게이트웨이 프로필 구성 방법

1 게이트웨이 프로필을 SMS 게이트웨이 서버 구성 파일에 추가합니다.

옵션 그룹을 추가하려면 다음 형식을 사용합니다.

```
[GATEWAY_PROFILE=profile_name]
option-name-1=option-value-1
option-name-2=option-value-2a
...
option-name-n=option-value-n
```

위 형식에서 게이트웨이 프로필 이름 `profile_name`의 길이는 11바이트를 초과해서는 안 됩니다. 이 이름은 SMS 채널 옵션 파일의 `GATEWAY_PROFILE` 채널 옵션에 대한 이름과 동일해야 합니다. 이 이름은 대소문자를 구분하지 않습니다. 유효한 채널 옵션 목록은 900 페이지 "C.3.3 사용 가능한 옵션"을 참조하십시오.

2 게이트웨이 프로필 옵션(예: `SMSC_DEFAULT_CHARSET`)을 원격 SMSC의 특성과 일치하도록 설정합니다.

3 다른 게이트웨이 프로필 옵션을 SMS 채널의 전자 메일 특성과 일치하도록 설정합니다.

게이트웨이 프로필 옵션에 대한 자세한 내용은 943 페이지 "C.5.11 게이트웨이 프로필 옵션"을 참조하십시오.

4 CHANNEL 옵션을 설정합니다.

해당 값을 MTA SMS 채널 이름으로 설정합니다.

게이트웨이를 통해 알림이 전자 메일로 보내지면 이 채널 이름을 사용하여 결과 전자 메일이 MTA의 대기열에 포함됩니다.

C.5.5.2 SMPP 중계

SMPP 중계를 구성하려면 다음 단계를 완료합니다.

▼ SMPP 릴레이 구성 방법

1 SMPP 중계 인스턴스화(옵션 그룹)를 SMS 게이트웨이 서버의 구성 파일에 추가합니다.

옵션 그룹을 추가하려면 다음 형식을 사용합니다.

```
[SMPP_RELAY=relay_name]
option-name-1=option-value-1
option-name-2=option-value-2
...
option-name-n=option-value-n
```

임의의 이름을 릴레이 이름으로 사용할 수 있습니다. 단, 해당 이름이 동일한 구성 파일 내의 다른 SMPP 중계 인스턴스화에 사용되어서는 안 됩니다.

2 LISTEN_PORT 옵션을 설정합니다.

SMS 채널의 SMPP_PORT 옵션에 사용되는 값은 릴레이의 LISTEN_PORT 옵션에 사용되는 것과 일치해야 합니다. LISTEN_PORT의 경우 다른 SMPP 중계 또는 서버 인스턴스화에 사용되지 않으며 동일한 컴퓨터에서 실행 중인 다른 서버에서도 사용되지 않는 TCP 포트 번호를 선택합니다.

3 SERVER_HOST 옵션을 설정합니다.

릴레이의 SERVER_HOST 옵션은 원격 SMSC의 SMPP 서버에 대한 호스트 이름을 제공해야 합니다. 호스트 이름 대신에 IP 주소를 사용할 수 있습니다.

4 SERVER_PORT 옵션을 설정합니다.

릴레이의 SERVER_PORT 옵션은 원격 SMSC의 SMPP 서버에 대한 TCP 포트를 제공해야 합니다.

모든 SMPP 릴레이 옵션에 대한 자세한 내용은 939 페이지 “C.5.9 SMPP 중계 옵션”을 참조하십시오.

C.5.5.3 SMPP 서버

SMPP 서버를 구성하려면 다음 단계를 완료합니다.

▼ SMPP 서버 구성 방법

1 SMPP 서버 인스턴스화(옵션 그룹)를 SMS 게이트웨이 서버의 구성 파일에 추가합니다.

옵션 그룹을 추가하려면 다음 형식을 사용합니다.

```
[SMPP_SERVER=server_name]
option-name-1=option-value-1
option-name-2=option-value-2...
option-name-n=option-value-n
```

임의의 이름을 서버 이름으로 사용할 수 있습니다. 단, 해당 이름이 동일한 구성 파일 내의 다른 SMPP 서버 인스턴스화에 사용되어서는 안 됩니다.

2 LISTEN_PORT 옵션을 설정합니다.

다른 서버나 중계 인스턴스화에 사용되지 않는 TCP 포트 번호를 선택합니다. 또한 이 포트 번호는 동일한 컴퓨터의 다른 서버에 사용되면 안 됩니다.

이 TCP 포트를 사용하여 SMPP를 통해 알림을 SMS 게이트웨이 서버 시스템으로 라우팅하도록 원격 SMSC를 구성해야 합니다.

모든 SMPP 서버 옵션에 대한 자세한 내용은 942 페이지 "C.5.10 SMPP 서버 옵션"을 참조하십시오.

C.5.6 MTE(Mobile-To-Email) 작업 구성

MTE(Mobile-To-Email) 기능을 구성하려면 다음 두 가지 구성 단계를 수행해야 합니다.

여러 게이트웨이 프로필을 동일한 SMPP 서버 인스턴스화를 사용할 수 있습니다. 실제로 동일한 SMPP 서버 인스턴스화를 ETM(Email-To-Mobile) 및 MTE(Mobile-To-Email) 응용 프로그램에 모두 사용할 수 있습니다.

C.5.6.1 MTE(Mobile-To-Email) 게이트웨이 프로필 구성

모바일 원본의 경우 게이트웨이 프로필은 두 가지 주요 정보 즉, 해당 프로필을 대상으로 하는 SMS 메시지를 식별하는 방법과 이러한 메시지를 전자 메일로 변환하는 방법을 제공합니다. 이 프로필은 SELECT_RE 옵션을 추가할 경우 ETM(Email-To-Mobile)에 사용되는 프로필과 동일할 수 있습니다.

게이트웨이 프로필을 구성하려면 다음 단계를 수행합니다.

▼ 게이트웨이 프로필 구성 방법

1 게이트웨이 프로필(옵션 그룹)을 SMS 게이트웨이 서버의 구성 파일에 추가합니다.

옵션 그룹을 추가하려면 다음 형식을 사용합니다.

```
[GATEWAY_PROFILE=profile_name]
option-name-1=option-value-1
option-name-2=option-value-2
...
option-name-n=option-value-n
```

11자 이하의 모든 이름을 프로필 이름으로 사용할 수 있습니다. 단, 해당 이름을 동일한 구성 파일 내의 다른 게이트웨이 프로필에서 사용해서는 안 됩니다.

2 각 게이트웨이 프로필에 대해 지정해야 하는 SELECT_RE 옵션을 설정합니다.

이 옵션 값은 SMS 대상 주소가 비교되는 ASCII 정규 표현식입니다. SMS 대상 주소가 정규 표현식과 일치할 경우 일치하는 프로필에 설명된 특성을 사용하여 SMS 메시지가 게이트웨이를 통해 전자 메일로 보내집니다.

겹치는 SMS 주소 집합을 가진 여러 게이트웨이 프로필(예: 주소 000과 일치하는 프로필과 다른 모든 3자리 주소와 일치하는 또 다른 프로필)을 구성할 수 있습니다. 그러나 SMS 메시지가 하나의 게이트웨이 프로필(일치하는 첫 번째 것)로만 전달되므로 여러 게이트웨이 프로필을 구성하는 것을 피해야 합니다. 또한 이러한 프로필이 비교되는 순서는 정의되지 않습니다.

3 CHANNEL 옵션을 설정합니다.

해당 값은 MTA SMS 채널의 이름이어야 합니다.

모든 모바일 원본 옵션에 대한 자세한 내용은 943 페이지 "C.5.11 게이트웨이 프로필 옵션"을 참조하십시오.

C.5.6.2 MTE(Mobile-To-Email) SMPP 서버 구성

SMPP 서버 추가는 ETM(Email-To-Mobile) SMPP 서버의 경우와 같습니다(932 페이지 "C.5.5.3 SMPP 서버" 참조).

SMS 트래픽을 게이트웨이 SMPP 서버로 라우팅하도록 원격 SMSC를 구성해야 합니다. 이렇게 하려면 MTE(Mobile-To-Email) 트래픽을 라우팅하기 위해 SMSC가 사용하는 SMS 대상 주소가 게이트웨이 프로필 옵션 SELECT_RE에 설정된 값이어야 합니다.

예를 들어, SMS 주소 000이 MTE(Mobile-To-Email) 트래픽에 사용될 경우 SMS 대상 주소 000에 대한 트래픽을 게이트웨이 SMPP 서버로 라우팅하도록 SMSC를 구성해야 합니다. 게이트웨이 프로필은 옵션 설정 SELECT_RE=000을 사용해야 합니다.

C.5.7 구성 옵션

이 절에서는 SMS 게이트웨이 서버 구성 파일 옵션에 대해 자세히 설명합니다. 이 절에 나오는 표에는 사용 가능한 모든 구성 옵션이 간단한 설명과 함께 나열되어 있습니다. 전역 옵션, SMPP 중계 옵션, SMPP 서버 옵션 및 SMS 게이트웨이 서버 프로필 옵션에 대한 표가 제공됩니다.

이 절은 사용 가능한 모든 구성 옵션에 대한 자세한 설명을 제공하는다음 하위 절로 구성되어 있습니다.

- 935 페이지 “C.5.8 전역 옵션”

  전역 옵션은 모든 옵션 그룹의 앞에 오도록 구성 파일의 맨 위에 두어야 합니다. 나머지 옵션은 옵션 그룹 안에 있어야 합니다.
- 939 페이지 “C.5.9 SMPP 중계 옵션”
- 942 페이지 “C.5.10 SMPP 서버 옵션”
- 943 페이지 “C.5.11 게이트웨이 프로필 옵션”

C.5.8 전역 옵션

SMS 게이트웨이 서버에는 현재 다음 세 가지 범주의 전역 옵션이 있습니다.

- 936 페이지 “C.5.8.1 스레드 조정 옵션”
- 936 페이지 “C.5.8.2 기록 데이터 조정”
- 937 페이지 “C.5.8.3 기타”

모든 전역 옵션은 지정된 모든 옵션 그룹의 앞에 오도록 구성 파일의 맨 위에 지정해야 합니다. 표 C–20에는 모든 전역 구성 옵션이 나열되어 있습니다.

표 C–20 전역 옵션

옵션	기본값	설명
938 페이지 “DEBUG”	6	생성된 진단 출력의 유형을 선택합니다.
937 페이지 “HISTORY_FILE_DIRECTORY”		기록 데이터 파일의 절대 디렉토리 경로입니다.
937 페이지 “HISTORY_FILE_MODE”	0770	기록 데이터 파일에 대한 권한입니다.
937 페이지 “HISTORY_FILE_ROLLOVER_PERIOD”	30분	기록 데이터의 동일한 파일에 쓸 수 있는 최대 시간입니다.
938 페이지 “LISTEN_CONNECTION_MAX”	10,000	모든 SMPP 중계 및 서버 인스턴스화에서의 최대 동시 인바운드 연결 수입니다.
937 페이지 “RECORD_LIFETIME”	3일	기록 데이터 아카이브에 있는 레코드의 수명입니다.

표 C–20 전역 옵션 *(계속)*

옵션	기본값	설명
936 페이지 “THREAD_COUNT_INITIAL”	10개	작업자 스레드의 초기 수입니다.
936 페이지 “THREAD_COUNT_MAXIMUM”	50개	작업자 스레드의 최대 수입니다.
936 페이지 “THREAD_STACK_SIZE”	64KB	각 작업자 스레드의 스택 크기입니다.

C.5.8.1 스레드 조정 옵션

각 인바운드 TCP 연결은 SMPP 세션을 나타냅니다. 세션 처리는 스레드 풀의 작업자 스레드에 의해 수행됩니다. 세션 처리가 I/O 요청이 완료되기를 기다려야 할 경우 작업 스레드는 세션을 대기시키며 수행할 다른 작업이 작업 스레드에 제공됩니다. I/O 요청이 완료되면 풀의 사용 가능한 작업자 스레드에 의해 세션이 다시 시작됩니다.

다음 옵션을 사용하여 이 작업자 스레드 프로세스 풀을 조정할 수 있습니다. 936 페이지 “THREAD_COUNT_INITIAL”, 936 페이지 “THREAD_COUNT_MAXIMUM”, 936 페이지 “THREAD_STACK_SIZE”.

THREAD_COUNT_INITIAL

*(**정수**, > 0)* 초기에 작업자 스레드 풀에 대해 만들어지는 스레드 수입니다. 메모리 내장 기록 데이터를 관리하는 데 사용되는 전용 스레드(두 개의 스레드) 및 받는 TCP 연결을 수신하는 데 사용되는 전용 스레드(SMS 게이트웨이 서버가 수신하는 TCP 포트/인터페이스 주소 쌍마다 스레드 하나씩)는 이 수에 포함되지 않습니다. `THREAD_COUNT_INITIAL`의 기본값은 10개입니다.

THREAD_COUNT_MAXIMUM

*(**정수**, >= `THREAD_COUNT_INITIAL`)* 작업자 스레드 풀에 허용되는 최대 스레드 수입니다. 기본값은 50개입니다.

THREAD_STACK_SIZE

*(**정수**, > 0)* 작업자 스레드 풀의 각 작업자 스레드에 대한 스택 크기(바이트)입니다. 기본값은 65,536바이트(64KB)입니다.

C.5.8.2 기록 데이터 조정

SMS 메시지가 중계되면 원격 수신 SMPP 서버에 의해 생성된 메시지 아이디가 메모리 내장 해시 테이블에 저장됩니다. 또한 이 메시지 아이디와 함께 원래 전자 메일에 대한 정보가 저장됩니다. 그 후에 메시지 아이디가 SMS 알림에 의해 참조될 경우 이 정보를 검색할 수 있습니다. 이어서 검색된 정보를 사용하여 SMS 알림을 적절한 전자 메일 수신자에게 보낼 수 있습니다.

메모리 내장 해시 테이블은 전용 스레드에 의해 디스크에 저장됩니다. 결과 디스크 파일은 "기록 파일"이라고 합니다. 이러한 기록 파일은 SMS 게이트웨이 서버를 다시 시작한 후 메모리 내장 해시 테이블을 복원하는 데 필요한 데이터를 비휘발성 형태로 저장하고 잠재적으로 긴 데이터를 디스크에 저장하여 가상 메모리를 절약하는 두 가지 역할을 수행합니다. 각 기록 파일은 HASH_FILE_ROLLOVER_PERIOD(초) 동안만 데이터가 기록되며 이 기간 후에는 파일이 닫히고 새 기록 파일이 만들어집니다. 기록 파일은 RECORD_LIFETIME(초)을 초과할 경우 디스크에서 삭제됩니다.

다음 옵션을 사용하여 기록 파일을 조정할 수 있습니다. 937 페이지 “HISTORY_FILE_DIRECTORY”, 937 페이지 “HISTORY_FILE_MODE”, 937 페이지 “HISTORY_FILE_ROLLOVER_PERIOD”, 937 페이지 “RECORD_LIFETIME”.

HISTORY_FILE_DIRECTORY

*(**문자열**, **절대 디렉토리 경로**)* 기록 파일을 기록할 디렉토리의 절대 경로입니다. 디렉토리 경로는 존재하지 않을 경우 새로 만들어집니다. 이 옵션의 기본값은 다음과 같습니다.

msg-svr-base/data/sms_gateway_cache/

사용할 디렉토리는 충분한 속도의 디스크 시스템에 존재하며 예상 저장소에 충분한 여유 공간을 가져야 합니다. 저장소 계획 정보에 대해서는 950 페이지 “C.6 SMS 게이트웨이 서버 저장소 요구 사항”을 참조하십시오.

HISTORY_FILE_MODE

*(**정수**, ***8*진수 값**)* 기록 파일과 연관된 파일 권한입니다. 기본값은 0770(8진수)입니다.

HISTORY_FILE_ROLLOVER_PERIOD

*(**정수**, **초**)* HASH_FILE_ROLLOVER_PERIOD(초)마다 현재 기록 파일이 닫히고 새 기록 파일이 만들어집니다. 기본값은 1800초(30분)입니다.

RECORD_LIFETIME

*(**정수**, **초** > 0)* 기록 레코드의 수명(초)입니다. 이 수명보다 오래된 레코드는 메모리에서 제거됩니다. 즉, 이 수명보다 오래된 기록 파일은 디스크에서 삭제됩니다. 기본값은 259,200초(3일)입니다. 메모리에 저장된 레코드는 메모리 내장 데이터를 관리하는 전용 스레드에 의해 완전히 제거됩니다. 이러한 제거는 HASH_FILE_ROLLOVER_PERIOD(초)마다 발생합니다. 디스크의 파일은 새 기록 파일을 여는 것이 필요할 때 제거됩니다.

C.5.8.3 기타

다음은 기타 옵션입니다.

- 938 페이지 “DEBUG”
- 938 페이지 “LISTEN_CONNECTION_MAX”

■ 938 페이지 “LOG_PAGE_COUNT”

DEBUG

*(**정수**, **비트 마스크**)* 디버그 출력을 사용 가능하게 합니다. 기본값은 경고 및 오류 메시지를 선택하는 6입니다.

표 C–21에는 DEBUG 비트 마스크의 비트 값이 정의되어 있습니다.

표 C–21 DEBUG 비트 마스크

비트	값	설명
0-31	-1	매우 자세한 출력
0	1	정보 메시지
1	2	경고 메시지
3	4	오류 메시지
3	8	서브루틴 호출 추적
4	16	해시 테이블 진단
5	32	I/O 진단, 수신
6	64	I/O 진단, 전송
7	128	SMS에서 전자 메일로의 변환 진단(모바일 원본 및 SMS 알림)
8	256	PDU 진단, 헤더 데이터
9	512	PDU 진단, 본문 데이터
10	1024	PDU 진단, 유형 길이 값 데이터
11	2048	옵션 처리(모든 옵션 설정을 로그 파일로 보냄)

LISTEN_CONNECTION_MAX

*(**정수**, >= 0)* 모든 SMPP 중계 및 서버 인스턴스화에서 허용할 최대 동시 인바운드 TCP 연결 수입니다. 값 0은 연결 수에 전역 제한이 없음을 나타냅니다. 그러나 지정된 중계 또는 서버 인스턴스화에 의해 중계 또는 서버별 제한이 있을 수 있습니다. 기본값: 10,000

LOG_PAGE_COUNT

(0, 1, 2) LOG_PAGE_CONT SMS 채널 옵션은 logging 채널 키워드를 사용하여 채널에 대해 로깅이 활성화된 경우에만 적용됩니다. 로깅이 활성화되면 이 옵션은 mail.log 파일의 메시지 크기 필드에 기록된 값을 제어합니다. 일반적으로 이 필드는 기본 메시지 파일의 블록 크기를 나타냅니다. LOG_PAGE_COUNT의 값이 0이 아니면, 전송된 페이지 수가 로그 파일의 이 필드에 대신 기록됩니다.

0 - 기본 메시지 파일의 블록 크기를 기록합니다. 이는 LOG_PAGE_COUNT가 지정되지 않은 경우 기본 동작입니다.

1 - 전체 메시지를 수신자에게 성공적으로 전송한 경우 보내진 페이지 수를 기록합니다. 그렇지 않으면 수신자에게 일부 페이지가 보내졌더라도 페이지 수를 0으로 기록합니다.

2 - 전체 메시지가 보내졌는지 여부와 상관 없이 수신자에게 보내진 페이지 수를 기록합니다.

LOG_PAGE_COUNT=1과 LOG_PAGE_COUNT=2 의 차이는 메시지가 여러 페이지로 전송될 만큼 큰 경우에만 적용됩니다. 그 경우 모든 페이지를 전송하기 전에 오류가 발생할 가능성이 있습니다. 예를 들어, MTA와 원격 SMPP 서버 간의 네트워크가 다운될 수 있습니다. 그 경우 나중에 해당 메시지를 다시 전송하도록 시도합니다. 시도할 때마다 이전에 보낸 페이지를 보내지 않은 페이지와 함께 다시 보냅니다. 사이트에서는 이러한 실패한 배달 시도 중에 제대로 보내진 페이지 수를 기록할 것인지 여부를 선택할 수 있습니다.

C.5.9 SMPP 중계 옵션

SMS 게이트웨이 서버는 해당 SMPP 중계의 여러 인스턴스화를 가질 수 있으며 각 인스턴스화는 수신되는 TCP 포트 및 인터페이스에 대해 다른 특성 부분을 가집니다. SMPP 중계가 수신하는 각 네트워크 인터페이스 및 TCP 포트 쌍에 대해 다르게 지정함으로써 고유한 특성을 포함시킬 수 있습니다. 이러한 특성은 이 절에 설명된 옵션을 사용하여 지정합니다.

각 인스턴스화는 다음 형식의 옵션 그룹 안에 포함되어야 합니다.

```
[SMPP_RELAY=relay-name]
option-name-1=option-value-1
option-name-2=option-value-2
...
option-name-n=option-value-n
```

문자열 relay-name은 단순히 이 인스턴스화를 다른 인스턴스화와 구별하는 역할을 수행합니다.

표 C–22에는 SMPP 릴레이 구성 옵션이 나열되어 있습니다.

표 C–22 SMPP 중계 옵션

옵션	기본값	설명
940 페이지 “C.5.9.1 LISTEN_BACKLOG”	255	인바운드 SMPP 클라이언트 연결의 연결 백로그입니다.
940 페이지 “LISTEN_CONNECTION_MAX”		동시 인바운드 연결의 최대 수입니다.

표 C–22 SMPP 중계 옵션 (계속)

옵션	기본값	설명
940 페이지 “LISTEN_INTERFACE_ADDRESS”		인바운드 SMPP 클라이언트 연결의 네트워크 인터페이스입니다.
941 페이지 “LISTEN_PORT”		인바운드 SMPP 클라이언트 연결의 TCP 포트입니다.
941 페이지 “LISTEN_RECEIVE_TIMEOUT”	`600 s`	인바운드 SMPP 클라이언트 연결의 읽기 시간 초과입니다.
941 페이지 “LISTEN_TRANSMIT_TIMEOUT”	`120 s`	인바운드 SMPP 클라이언트 연결의 쓰기 시간 초과입니다.
941 페이지 “MAKE_SOURCE_ADDRESSES_UNIQUE”	`1`	중계된 SMS 소스 주소를 고유하게 만들고 응답 가능하게 합니다.
941 페이지 “SERVER_HOST”		중계할 SMPP 서버의 호스트 이름 또는 IP 주소입니다.
941 페이지 “SERVER_PORT”		중계할 SMPP 서버의 TCP 포트입니다.
941 페이지 “SERVER_RECEIVE_TIMEOUT”	`600 s`	아웃바운드 SMPP 서버 연결의 읽기 시간 초과입니다.
942 페이지 “SERVER_TRANSMIT_TIMEOUT”	`120 s`	아웃바운드 SMPP 서버 연결의 쓰기 시간 초과입니다.

C.5.9.1 LISTEN_BACKLOG

(**`[0,255]`의 정수**) TCP 스택이 허용하는 인바운드 SMPP 클라이언트 연결의 연결 백로그입니다. 기본값은 255입니다.

LISTEN_CONNECTION_MAX

(**정수**, *>= 0*) 이 SMPP 중계 인스턴스화에 허용되는 동시 인바운드 TCP 연결의 최대 수입니다. 이 값은 전역 `LISTEN_CONNECTION_MAX` 설정을 초과할 경우 무시됩니다.

LISTEN_INTERFACE_ADDRESS

문자열, **`"INADDR_ANY"` 또는 점으로 구분된 십진수 *IP* 주소**) 인바운드 SMPP 클라이언트 연결의 수신할 네트워크 인터페이스에 대한 IP 주소입니다. 문자열 “`INADDR_ANY`”(사용 가능한 모든 인터페이스)이거나 점으로 구분된 십진수 형식의 IP 주소입니다(예: 193.168.100.1). 기본값은 “`INADDR_ANY`” 입니다. 클러스터된 HA 구성은 이 값을 HA 논리 IP 주소에 해당하도록 설정해야 합니다.

LISTEN_PORT

*(**정수**, *TCP* **포트 번호**)* 인바운드 SMPP 클라이언트 연결을 수락하기 위해 바인드할 TCP 포트입니다. 이 옵션 지정은 필수 사항이며 기본값은 없습니다. 또한 이 서비스에 대한 IANA(Internet Assigned Numbers Authority) 할당이 없습니다.

LISTEN_RECEIVE_TIMEOUT

*(**정수**, **초** > 0)* SMPP 클라이언트로부터 데이터를 읽는 동안 기다릴 때 허용되는 시간 초과입니다. 기본값은 600초(10분)입니다.

LISTEN_TRANSMIT_TIMEOUT

*(**정수**, **초** > 0)* 데이터를 SMPP 클라이언트로 보낼 때 허용되는 시간 초과입니다. 기본값은 120초(2분)입니다.

MAKE_SOURCE_ADDRESSES_UNIQUE

*(0 **또는** 1)* 기본적으로 SMPP 중계는 고유한 10자리 문자열을 각 SMS 소스 주소에 추가합니다. 그런 다음 결과 SMS 소스 주소는 다른 기록 데이터와 함께 저장됩니다. 결과는 SMS 사용자가 응답할 수 있는 고유한 SMS 주소입니다. SMPP 서버는 SMS 대상 주소로 사용될 때 이 주소를 감지한 다음 SMS 메시지를 올바른 전자 메일 발송자에게 보냅니다.

단방향 SMS에 대해 이러한 고유한 SMS 소스 주소가 생성되지 않게 하려면 이 옵션의 값으로 0을 지정합니다.

SERVER_HOST

*(**문자열**, *TCP* **호스트 이름 또는 점으로 구분된 십진수** *IP* **주소**)* SMPP 클라이언트 트래픽을 중계할 SMPP 서버입니다. 호스트 이름이나 IP 주소를 지정할 수 있습니다. 이 옵션 지정은 필수 사항이며 기본값은 없습니다.

SERVER_PORT

*(**정수**, *TCP* **포트 번호**)* 중계할 원격 SMPP 서버의 TCP 포트입니다. 이 옵션 지정은 필수 사항이며 기본값은 없습니다. 이 서비스에 대한 IANA 할당은 없습니다. 따라서 SNPP에 대한 IANA 할당과 혼동하지 않도록 주의합니다.

SERVER_RECEIVE_TIMEOUT

*(**정수**, **초** > 0)* SMPP 서버로부터 데이터를 읽는 동안 기다릴 때 허용되는 시간 초과입니다. 기본값은 600초(10분)입니다.

SERVER_TRANSMIT_TIMEOUT

*(**정수**, **초** > 0)* 데이터를 SMPP 서버로 보낼 때 허용되는 시간 초과입니다. 기본값은 120초(2분)입니다.

C.5.10 SMPP 서버 옵션

SMS 게이트웨이 서버는 해당 SMPP 서버의 여러 인스턴스화를 가질 수 있으며 각 인스턴스화는 수신되는 TCP 포트 및 인터페이스에 대해 다른 특성 부분을 가집니다. SMPP 서버가 수신하는 각 네트워크 인터페이스 및 TCP 포트 쌍에 대해 다르게 지정함으로써 고유한 특성을 포함할 수 있습니다. 이러한 특성은 이 절에 설명된 옵션을 사용하여 지정합니다.

각 인스턴스화는 다음 형식의 옵션 그룹 안에 포함되어야 합니다.

```
[SMPP_SERVER=server-name]
option-value-1=option-value-1
option-value-2=option-value-2
...
option-name-n=option-value-n
```

문자열 `server-name`은 단순히 이 인스턴스화를 다른 인스턴스화와 구별하는 역할을 수행합니다.

표 C–23에는 SMPP 서버 구성 옵션이 나열되어 있습니다.

표 C–23 SMPP 서버 옵션

옵션	기본값	설명
943 페이지 “C.5.10.1 LISTEN_BACKLOG”	`255`	인바운드 SMPP 서버 연결의 연결 백로그입니다.
943 페이지 “LISTEN_CONNECTION_MAX”		동시 인바운드 연결의 최대 수입니다.
943 페이지 “LISTEN_INTERFACE_ADDRESS”		인바운드 SMPP 서버 연결의 네트워크 인터페이스입니다.
943 페이지 “LISTEN_PORT”		인바운드 SMPP 서버 연결의 TCP 포트입니다.
943 페이지 “LISTEN_RECEIVE_TIMEOUT”	`600 s`	인바운드 SMPP 서버 연결의 읽기 시간 초과입니다.
943 페이지 “LISTEN_TRANSMIT_TIMEOUT”	`120 s`	인바운드 SMPP 서버 연결의 쓰기 시간 초과입니다.

C.5.10.1 LISTEN_BACKLOG

(*[0,255]*의 정수) TCP 스택이 허용하는 인바운드 SMPP 클라이언트 연결의 연결 백로그입니다. 기본값은 255입니다.

LISTEN_CONNECTION_MAX

*(*정수, *>= 0)* 이 SMPP 서버 인스턴스화에 허용되는 동시 인바운드 TCP 연결의 최대 수입니다. 이 값은 전역 `LISTEN_CONNECTION_MAX` 설정을 초과할 경우 무시됩니다.

LISTEN_INTERFACE_ADDRESS

*(*문자열, `"INADDR_ANY"` 또는 점으로 구분된 십진수 *IP* 주소*)* 인바운드 SMPP 클라이언트 연결의 수신할 네트워크 인터페이스에 대한 IP 주소입니다. 문자열 "`INADDR_ANY`"(사용 가능한 모든 인터페이스)이거나 점으로 구분된 십진수 형식의 IP 주소입니다(예: 193.168.100.1). 기본값은 "`INADDR_ANY`" 입니다.

LISTEN_PORT

*(*정수, *TCP* 포트 번호*)* 인바운드 SMPP 클라이언트 연결을 수락하기 위해 바인드할 TCP 포트입니다. 이 옵션 지정은 필수 사항이며 기본값은 없습니다. 이 서비스에 대한 IANA 할당이 없습니다.

LISTEN_RECEIVE_TIMEOUT

*(*정수, 초 *> 0)* SMPP 클라이언트로부터 데이터를 읽는 동안 기다릴 때 허용되는 시간 초과입니다. 기본값은 600초(10분)입니다.

LISTEN_TRANSMIT_TIMEOUT

*(*정수, 초 *> 0)* 데이터를 SMPP 클라이언트로 보낼 때 허용되는 시간 초과입니다. 기본값은 120초(2분)입니다.

C.5.11 게이트웨이 프로필 옵션

0개 이상의 게이트웨이 프로필이 존재할 수 있습니다. SMS 게이트웨이 서버의 구성 파일에서 각 게이트웨이 프로필은 옵션 그룹 내에서 다음 형식으로 선언됩니다.

```
[GATEWAY_PROFILE=profile-name]
option-name-1=option-value-1
option-name-2=option-value-2
...
option-name-n=option-value-n
```

문자열 `profile-name`은 단순히 해당 프로필을 다른 원본 프로필과 구별하는 역할을 수행합니다.

표 C–24에는 SMS 게이트웨이 서버 프로필 옵션이 나열되어 있습니다.

표 C–24 SMS 게이트웨이 서버 프로필 옵션

옵션	기본값	설명
944 페이지 "C.5.11.1 CHANNEL"	sms	메시지를 대기열에 포함하는 채널입니다.
944 페이지 "EMAIL_BODY_CHARSET"	US-ASCII	전자 메일 본문의 문자 세트입니다.
945 페이지 "EMAIL_HEADER_CHARSET"	US-ASCII	전자 메일 헤더의 문자 세트입니다.
945 페이지 "FROM_DOMAIN"		전자 메일을 다시 SMS로 라우팅하기 위한 도메인 이름입니다.
945 페이지 "PARSE_RE_0, PARSE_RE_1, ..., PARSE_RE_9"		SMS 메시지 텍스트를 구문 분석하기 위한 정규 표현식입니다.
946 페이지 "PROFILE"	GSM	GSM, TDMA 또는 CDMA에서 작동하는SMS 프로필입니다.
947 페이지 "SELECT_RE"		플러그 인을 선택하기 위한 정규 표현식입니다.
947 페이지 "SMSC_DEFAULT_CHARSET로 변환"	US-ASCII	SMSC의 기본 문자 세트
947 페이지 "USE_SMS_PRIORITY"	0	전자 메일에 대한 게이트웨이 SMS 우선 순위 플래그입니다.
948 페이지 "USE_SMS_PRIVACY"	0	전자 메일에 대한 게이트웨이 SMS 개인 정보 표시기입니다.

C.5.11.1 CHANNEL

*(**문자열**, 1-40**자**)* 전자 메일을 대기열에 포함시키는 데 사용되는 MTA 채널의 이름입니다. 지정하지 않으면 "sms"로 간주됩니다. 지정한 채널을 MTA의 구성에 정의해야 합니다.

EMAIL_BODY_CHARSET

*(**문자열**, **문자 세트 이름**)* 전자 메일 메시지의 본문에 삽입하기 전에 SMS 텍스트를 변환하기 위한 문자 세트입니다. 필요한 경우 변환된 텍스트는 MIME 인코딩됩니다. 기본값은 US-ASCII입니다. SMS 메시지가 charset에서 사용할 수 없는 도형 문자를 포함할 경우 이러한 도형 문자는 수신자에게 의미가 있거나 없을 수 있는 니모닉 문자로 변환됩니다.

MTA에 알려진 문자 세트 목록은 다음 파일에서 확인할 수 있습니다.

```
installation-directory/config/charsets.txt
```

EMAIL_HEADER_CHARSET

*(**문자열**, **문자 세트 이름**)* RFC 822 `Subject:` 헤더 행에 삽입하기 전에 SMS 텍스트를 변환하기 위한 문자 세트입니다. 필요한 경우 변환된 문자열은 MIME 인코딩됩니다. 기본값은 US-ASCII입니다. SMS 메시지가 charset에서 사용할 수 없는 도형 문자를 포함할 경우 이러한 도형 문자는 수신자에게 의미가 있거나 없을 수 있는 니모닉 문자로 변환됩니다.

FROM_DOMAIN

*(**문자열**, *IP* **호스트 이름**, *1-64***자**)* 전자 메일 메시지 봉투의 `From:` 주소를 생성할 때 SMS 소스 주소에 추가할 도메인 이름입니다. 지정된 이름은 전자 메일을 다시 SMS로 라우팅하기 위한 올바른이름이어야 합니다(예: MTA SMS 채널과 연관된 호스트 이름). 지정하지 않을 경우 `CHANNEL` 옵션으로 지정된 채널의 공식 호스트 이름이 사용됩니다.

PARSE_RE_0, PARSE_RE_1, ..., PARSE_RE_9

*(**문자열**, *UTF-8* **정규 표현식**)* 전자 메일의 모바일 원본의 경우 게이트웨이 프로필은 SMS 메시지의 텍스트에서 대상 전자 메일 주소를 추출해야 합니다. 이것은 하나 이상의 POSIX 호환 정규 표현식(RE)으로 수행됩니다. 대상 전자 메일 주소를 생성하는 일치하는 항목이 발견되거나 정규 표현식 목록이 모두 사용될 때까지 각 정규 표현식은 SMS 메시지의 텍스트를 평가합니다.

주 – `PARSE_RE_*` 및 `ROUTE_TO` 옵션을 동시에 사용할 수 없습니다. 동일한 게이트웨이 프로필에서 두 옵션을 함께 사용하는 것은 구성 오류입니다.

각 정규 표현식은 POSIX와 호환되어야 하며 UTF-8 문자 세트로 인코딩되어야 합니다. 정규 표현식은 문자열 0을 대상 주소로 출력해야 합니다. 선택적으로 `Subject:` 헤더 행에 사용할 텍스트를 문자열 1로 출력하고, 메시지 본문에 사용할 텍스트를 문자열 2로 출력할 수 있습니다. 정규 표현식에서 "사용"하지 않는 모든 텍스트는 문자열 2로 출력된 텍스트에 이어서 메시지 본문에 사용될 수 있습니다.

정규 표현식은 PARSE_RE_0, PARSE_RE_1, ...PARSE_RE_9의 순서로 시도됩니다. 지정된 정규 표현식이 없을 경우 다음 기본 정규 표현식이 사용됩니다.

```
[ \t]*([^\( ]*)[ \t]*(?:\(([^\)]*)\))?[ \t]*(.*)
```

이 기본 정규 표현식은 다음 구성 요소로 나뉩니다.

```
[ \t]*
```

선행 공백 문자(`SPACE` 및 `TAB`)를 무시합니다.

```
([^\( ]*)
```

대상 전자 메일 주소입니다. 첫 번째로 보고되는 문자열입니다.

```
[ \t]*
```

공백 문자를 무시합니다.

```
(?:\(([^\)]*)\))?
```

괄호로 묶인 선택적 제목 텍스트입니다. 두 번째로 보고되는 문자열입니다. `?:`는 바깥쪽 괄호가 문자열을 보고하지 않게 합니다. 이러한 괄호는 단순히 후행 `?`에 대해 해당 내용을 단일 RE로 그룹화하는 데 사용됩니다. 후행 `?`는 RE 구성 요소가 0번 또는 1번만 일치하도록 하며 표현식 {0,1}과 같습니다.

```
[ \t]*
```

공백 문자를 무시합니다.

```
(.*)
```

메시지 본문에 대한 나머지 텍스트입니다. 세 번째로 보고되는 문자열입니다.

예를 들어, 위 정규 표현식을 사용할 경우 다음 샘플 SMS 메시지는

```
dan@sesta.com(Testing)This is a test
```

다음 전자 메일을 생성합니다.

```
To: dan@sesta.com
Subject: Testing

This is a test
```

두 번째 예로 다음 SMS 메시지는

```
sue@sesta.com This is another test
```

다음을 생성합니다.

```
To: sue@sesta.com

This is another test
```

SMS 메시지는 이러한 정규 표현식으로 평가되기 전에 유니코드의 UTF-16 인코딩으로 변환됩니다. 그런 다음 변환된 텍스트는 이전에 UTF-8에서 UTF-16으로 변환된 정규 표현식으로 평가됩니다. 이어서 평가 결과는 `Subject:` 텍스트의 대상 전자 메일 주소 `EMAIL_HEADER_CHARSET`(있을 경우)와 메일 본문의 `EMAIL_BODY_CHARSET`(있을 경우)에 대한 US-ASCII로 변환됩니다.

PROFILE

*(**문자열**, "GSM", "TDMA" **또는** "CDMA")* 가정할 SMS 프로필입니다. 현재 이 정보는 SMS 우선 순위 플래그를 RFC 822 `Priority:` 헤더 행으로 매핑하기 위해 사용됩니다. 결과적으로 `USE_SMS_PRIORITY`가 기본값인 0일 경우 이 옵션은 아무 효과가 없습니다.

SELECT_RE

*(**문자열**, *US-ASCII* **정규 표현식**)* 각 SMS 메시지의 SMS 대상 주소와 비교할 US-ASCII POSIX 호환 정규 표현식입니다. SMS 메시지의 대상 주소가 이 RE와 일치할 경우 SMS 메시지는 이 게이트웨이 프로필에 따라 게이트웨이를 통해 전자 메일로 보내집니다.

SMS 메시지의 대상 주소가 US-ASCII 문자 세트로 지정되므로 이 정규 표현식도 US-ASCII로 표현되어야 합니다.

SMSC_DEFAULT_CHARSET로 변환

*(**문자열**, **문자 세트 이름**)* 원격 SMSC에 사용되는 기본 문자 세트의 이름입니다. 이 옵션에는 일반적으로 US-ASCII 및 UTF-16-BE(USC2)가 지정됩니다. 지정하지 않을 경우 US-ASCII가 사용됩니다.

USE_SMS_PRIORITY

*(**정수**, *0* **또는** *1)* 기본적으로 `USE_SMS_PRIORITY=0`이며 이 경우 SMS 메시지의 우선 순위 플래그가 무시되어 전자 메일 메시지와 함께 보내지지 않습니다. 우선 순위 플래그를 전자 메일과 함께 전달하려면 `USE_SMS_PRIORITY=1`을 지정합니다. 표 C–25에는 우선 순위 플래그가 전자 메일과 함께 전달될 경우의 SMS에서 전자 메일로의 매핑이 나와 있습니다.

표 C–25 SMS에서 전자 메일로의 우선 순위 플래그 매핑

SMS 프로필	SMS 우선 순위 플래그	전자 메일 우선 순위 헤더 행
GSM	0(낮음)	헤더 행 없음(Normal을 의미)
	1, 2, 3(우선 순위)	Urgent
TDMA	0(대량)	Nonurgent
	1(중간)	헤더 행 없음(Normal을 의미)
	2(높음)	Urgent
	3(매우 높음)	Urgent
CDMA	0(중간)	헤더 행 없음(Normal을 의미)
	1(대화식)	Urgent
	2(높음)	Urgent
	3(긴급)	Urgent

전자 메일 `Priority:` 헤더 행 값은 `Nonurgent`, `Normal` 및 `Urgent`입니다.

USE_SMS_PRIVACY

*(**정수**, 0 **또는** 1)* 기본적으로 `USE_SMS_PRIVACY=0`이며 이 경우 SMS 개인 정보 표시가 무시되어 전자 메일 메시지와 함께 보내지지 않습니다. 이 정보를 전자 메일과 함께 전달하려면 `USE_SMS_PRIVACY=1`을 지정합니다. 표 C–26에는 개인 정보 플래그가 전자 메일과 함께 전달될 경우 SMS에서 전자 메일로의 매핑이 나와 있습니다.

표 C–26 SMS에서 전자 메일로의 개인 정보 플래그 매핑

SMS 개인 정보 플래그	전자 메일 민감도 헤더 행
0(제한되지 않음)	헤더 행 없음
1(제한됨)	`Personal`
2(기밀)	`Private`
3(비밀)	`Company-confidential`

전자 메일 `Sensitivity:` 헤더 행 값은 `Personal`, `Private` 및 `Company-confidential`입니다.

C.5.12 양방향 SMS의 구성 예

동작에 대한 가정

이 예에서는 다음 동작을 원한다고 가정합니다.

- 다음 주소로 지정된 전자 메일이

  `sms-id@sms.domain.com`

  다음 SMS 주소로 전송되고

  `sms-id`

  `000`nnnnnnnnnn 범위의 고유한 SMS 소스 주소가 주어집니다.
- SMS 주소 `000`으로 지정된 모바일 SMS 메시지를 SMS 메시지 텍스트의 시작 부분에서 추출된 전자 메일 주소와 함께 게이트웨이를 통해 전자 메일로 보냅니다.

  예를 들어, SMS 메시지 텍스트가 다음과 같은 경우

  `jdoe@domain.com Interested in a movie?`

  "Interested in a movie?" 메시지가 jdoe@domain.com으로 보내집니다.
- `000`nnnnnnnnnn으로 보낸 SMS 알림을 게이트웨이를 통해 전자 메일로 보내고 수신 확인되는 메일 발송자에게 전송합니다.

이 동작을 수행하기 위해 다음 가정과 지정이 적용됩니다.

추가 가정 및 지정

- MTA의 SMS 채널은 도메인 이름 `sms.domain.com`을 사용합니다.

- SMS 게이트웨이 서버는 호스트 gateway.domain.com에서 실행되며 다음을 사용합니다.
  - SMPP 중계를 위한 TCP 포트 503
  - SMPP 서버를 위한 TCP 포트 504
- 원격 SMSC의 SMPP 서버는 호스트 `smpp.domain.com`에서 실행되며 TCP 포트 377을 수신합니다.
- 원격 SMSC의 기본 문자 세트는 UCS2(UTF-16이라고도 함)입니다.

SMS 채널 구성

위 동작을 적용하기 위해 `imta.cnf` 파일에서 다음 SMS 채널 구성을 사용할 수 있습니다(다음 행을 파일의 맨 아래에 추가).

```
(blank line)
sms
sms.domain.com
```

SMS 채널 옵션 파일

그러면 채널 옵션 파일 `sms_option`에 다음 설정이 포함됩니다.

```
SMPP_SERVER=gateway.domain.com
SMPP_PORT=503
USE_HEADER_FROM=0
DEFAULT_SOURCE_ADDRESS=000
GATEWAY_PROFILE=sms1
SMSC_DEFAULT_CHARSET=UCS2
```

SMS 게이트웨이 서버 구성

마지막으로 게이트웨이 서버 구성 파일 `sms_gateway.cnf`는 다음과 같이 표시되어야 합니다.

```
HISTORY_FILE_DIRECTORY=/sms_gateway_cache/
[SMPP_RELAY=relay1]
LISTEN_PORT=503SERVER_HOST=smpp.domain.com
SERVER_PORT=377

[SMPP_SERVER=server1]
LISTEN_PORT=504

[GATEWAY_PROFILE=sms1]
SELECT_RE=000([0-9]{10,10}){0,1}
SMSC_DEFAULT_CHARSET=UCS2
```

이 구성 테스트

테스트할 SMSC가 없을 경우 약간의 루프백 테스트를 수행할 수 있습니다. sms_option 파일에서 몇 가지 추가 설정을 사용하면 위 구성에 대해 간단한 루프백 테스트를 수행할 수 있습니다.

C.5.12.1 추가 sms_option 파일 설정

sms_option 파일의 추가 설정은 다음과 같습니다.

```
! So that we don't add text to the body of the SMS message
FROM_FORMAT=
SUBJECT_FORMAT=
CONTENT_PREFIX=
```

이러한 설정이 없으면 다음을 포함하는 전자 메일은

```
user@domain.com (Sample subject) Sample text
```

다음 SMS 메시지로 변환됩니다.

```
From:user@domain.com Subject:Sample Subject Msg:Sample text
```

이것은 다음과 같은 ETM(Email-To-Mobile) 코드에서 예상한 형식이 아닙니다.

```
user@domain.com (Sample subject) Sample text
```

따라서 루프백 테스트를 위해서는 FROM_FORMAT, SUBJECT_FORMAT 및 CONTENT_PREFIX 옵션에 빈 문자열을 지정해야 합니다.

루프백 테스트 수행

다음과 같이 000@sms.domain.com으로 주소 지정된 테스트 전자 메일을 보냅니다.

```
user@domain.com (Test message) This is a test message which should loop back
```

그 결과 이 전자 메일 메시지를 전자 메일 수신자 user@domain.com으로 다시 라우팅해야 합니다. 테스트를 위해 DNS 또는 호스트 테이블에 sms.domain.com을 추가했는지 확인합니다.

C.6 SMS 게이트웨이 서버 저장소 요구 사항

SMS 게이트웨이 서버에 필요한 자원의 양을 결정하려면 표 C–27의 요구 사항을 토대로 얻어진 숫자와 함께 초당 예상되는 릴레이 메시지 수와 RECORD_LIFETIME 설정을 사용합니다.

표 C–27에서는 기록 데이터, SMPP 릴레이 및 SMPP 서버에 대한 요구 사항을 보여 줍니다.

표 C–27 SMS 게이트웨이 서버 저장소 요구 사항

구성 요소	요구 사항
메모리 내장 기록 레코드	릴레이된 각 메시지는 33+m+s바이트의 가상 메모리가 필요합니다. 여기서 m은 메시지의 SMS 메시지 아이디의 길이(1 <= m <= 64)이고 s는 메시지의 SMS 소스 주소의 길이(1 <= s<= 20)입니다. MAKE_SOURCE_ADDRESS_UNIQUE=0이면 16+m바이트만 사용됩니다. 64비트 운영 체제의 경우 레코드당 49+m+s바이트의 가상 메모리가 사용됩니다[MAKE_SOURCE_ADDRESS_UNIQUE=0일 경우 24+m]. 또한 힙 할당자가 실제로 각 레코드에 대해 더 큰 가상 메모리를 할당할 수 있습니다. 최대 레코드 수는 430억개입니다(2**32-1). 해시 테이블은 레코드 수가 1680만개(2**24) 이하인 경우 약 16MB를 사용하고 6710만개(2**26) 이하인 경우 약 64MB를 사용하며 6710만개 이상인 경우 약 256MB를 사용합니다. 64비트 운영 체제의 경우 메모리 소비량을 두 배로 늘립니다. 각 레코드 자체에 필요한 메모리 소비량 외에도 이러한 소비량이 요구됩니다.
디스크 내장 기록 데이터	중계된 각 메시지에는 평균적으로 다음 수의 바이트가 필요합니다. 81+m+2s+3a+ S+2i 여기서 ■ m은 SMS 메시지 아이디의 평균 길이이며 1<=m<=64입니다. ■ s는 SMS 소스 주소의 평균 길이이며 1 <= s <= 20입니다. ■ a는 전자 메일 주소의 평균 길이이며 3 <= a <= 129입니다. ■ S는 Subject: 헤더 행의 평균 길이이며 0 <= S<= 80입니다. ■ i는 전자 메일 봉투 아이디의 평균 길이이며 0 <= i <= 129입니다. 특정 레코드의 크기는 메시지 봉투의 From: 및 To: 주소의 길이, 봉투 및 메시지 아이디의 길이, Subject: 헤더 행의 길이에 따라 달라집니다. 최대 레코드 길이는 910바이트입니다. MAKE_SOURCE_ADDRESS_UNIQUE=0이 사용될 때 각 레코드의 크기(바이트)는 다음과 같습니다.78+m+3a+S+2 i.
SMPP 중계	중계된 각 세션은 두 개의 소켓 즉,로컬 클라이언트를 가진 소켓과 원격 서버를 가진 소켓을 사용합니다. 32비트 운영 체제에서는 연결당 약 1KB의 가상 메모리가 사용되고 64비트 운영 체제에서는 약 2KB의 가상 메모리가 사용됩니다.
SMPP 서버	각 받는 연결은 하나의 TCP 소켓을 사용합니다. 32비트 운영 체제에서는 연결당 약 1KB의 가상 메모리가 사용되고 64비트 운영 체제에서는 약 2KB의 가상 메모리가 사용됩니다.

예를 들어, 초당 평균 50개의 메시지가 중계될 것으로 예상할 경우 SMS 소스 주소의 길이는 13바이트, SMS 메시지 아이디의 일반 길이는 12바이트, 전자 메일 주소는 24바이트, Subject: 행은 40바이트, 전자 메일 및 봉투 아이디는 각각 40바이트이고, 기록 데이터는 7일 동안 보관되며, 그 이후에는 다음과 같습니다.

- 저장할 기록 레코드가 3024만 개이며 이러한 레코드는 각각 메모리와 디스크에서 평균 58바이트 및 311바이트의 길이를 가집니다.
- 기록 레코드의 메모리 내장 소비량은 약 1.70GB(1.6GB + 64MB)입니다.
- 디스크 내장 저장소는 약 8.76GB입니다.

모든 디스크 내장 요구 사항을 처리하기 위해 충분한 디스크가 제공될 수 있지만 32비트 시스템의 가상 메모리 요구 사항은 약 2GB로 엄격하게 제한됩니다. 필요한 가상 메모리나 디스크 저장소의 양을 줄이려면 `RECORD_LIFETIME` 옵션을 사용하여 레코드를 보유하는 시간을 줄입니다.

◆ ◆ ◆ 부 록 D

설치 워크시트

이 부록에서는 설치를 계획할 수 있는 워크시트를 제공합니다. 다음 워크시트가 포함되어 있습니다.

D.1 Directory Server 설치

Java Enterprise System 설치 프로그램 또는 이전 설치를 통해 Directory Server를 설치했습니다. 표 D–1(Communications Suite Deployment Planning Guide에 표시된 워크시트의 복제본)에 Directory Server 설치 및 구성 매개 변수를 기록합니다. Administration Server 및 Messaging Server를 설치하고 구성할 때 이 매개 변수가 필요합니다.

표 D–1 Directory Server 설치 매개 변수

매개 변수:	설명:	예:	사용:	사용자의 답변:
디렉토리 설치 루트	서버 프로그램, 구성, 유지 관리 및 정보 파일을 저장하는 Directory Server 시스템의 전용 디렉토리입니다.	`/var/mps/serverroot/`	`comm_dssetup.pl` Perl 스크립트 48 페이지 “1.2 Messaging Server 구성을 위해 Directory Server 준비”를 참조하십시오.	

표 D–1 Directory Server 설치 매개 변수 (계속)

매개 변수:	설명:	예:	사용:	사용자의 답변:
호스트	호스트 이름은 IP 호스트 이름이며 "짧은 형식"의 호스트 이름(예: `fiddle`)이거나 정규화된 호스트 이름입니다. 정규화된 호스트 이름은호스트 이름과 도메인 이름의 두 부분으로 구성됩니다.	`fiddle.west.sesta.com`	Administration Server 구성 48 페이지 “1.2 Messaging Server 구성을 위해 Directory Server 준비”를 참조하십시오.	
LDAP 디렉토리 포트 번호	LDAP Directory Server의 기본값은 `389`입니다.	`389`	Administration Server 구성 및 Messaging Server 구성 48 페이지 “1.2 Messaging Server 구성을 위해 Directory Server 준비” 및 49 페이지 “1.3 Messaging Server 초기 런타임 구성 만들기 ”를 참조하십시오.	
관리자 아이디 및 비밀번호	구성 정보를 담당하는 관리자입니다. 관리자의 비밀번호입니다.	`Admin` PaSsWoRd	Administration Server 구성 48 페이지 “1.2 Messaging Server 구성을 위해 Directory Server 준비”를 참조하십시오.	
사용자 및 그룹 트리 접미어	디렉토리 트리의 최상위에 있는 LDAP 항목의 고유 이름으로서 이 위치 아래에 사용자 및 그룹 데이터가 저장됩니다.	`o=usergroup`)	`comm_dssetup.pl` Perl 스크립트 48 페이지 “1.2 Messaging Server 구성을 위해 Directory Server 준비”를 참조하십시오.	

표 D–1 Directory Server 설치 매개 변수 (계속)

매개 변수:	설명:	예:	사용:	사용자의 답변:
디렉토리 관리자 DN 및 비밀번호	UNIX의 수퍼유저 사용자에 해당하는 권한 있는 디렉토리 관리자입니다. 일반적으로 이 관리자는 사용자와 그룹 데이터의 관리를 담당합니다. 디렉토리 관리자의 비밀번호입니다.	`cn=Directory Manager` `pAsSwOrD`	`comm_dssetup.pl` Perl 스크립트 및 Messaging Server 구성 48 페이지 “1.2 Messaging Server 구성을 위해 Directory Server 준비” 및 49 페이지 “1.3 Messaging Server 초기 런타임 구성 만들기”를 참조하십시오.	
관리 도메인	관리 제어의 영역입니다.	`System Lab`	Administration Server 구성 48 페이지 “1.2 Messaging Server 구성을 위해 Directory Server 준비”를 참조하십시오.	

D.2 Directory Server 설정 스크립트(comm_dssetup.pl)

Messaging Server 구성을 위한 Directory Server를 준비하기 위해 Directory Server 설정 스크립트(`comm_dssetup.pl`)를 실행하는 경우 표 D–2에 설치 매개 변수를 기록하십시오. Messaging Server 초기 런타임 구성 시 이 매개 변수가 필요합니다.

표 D–2 `comm_dssetup.pl` 스크립트 매개 변수

매개 변수	설명	예	사용자의 답변:
서버 루트	서버 프로그램, 구성, 유지 관리 및 정보 파일을 저장하는 Directory Server의 전용 설치 루트입니다.	`/var/mps/serverroot/`	
서버 인스턴스	대부분의 기능을 담당하는 LDAP Directory Server 데몬 또는 서비스입니다. 특정 배포의 경우 사용자와 그룹을 유지 관리하기 위한 전용 인스턴스를 지정하고 구성에 대해서는 별도의 인스턴스를 지정해야 할 수 있습니다.	`slapd-varrius`	

표 D–2 comm_dssetup.pl 스크립트 매개 변수 (계속)

매개 변수	설명	예	사용자의 답변:
DC 루트	2 트리 DIT 준비 모델(Sun LDAP Schema 1 또는 Sun LDAP Schema 2 호환 모드)을 사용하려는 경우 DC 트리는 로컬 DNS 구조를 미러하며, 시스템에서 사용자와 그룹 데이터 항목이 포함된 조직 트리에 대한 색인으로 사용됩니다.	o=internet)	
사용자 및 그룹 기본 접미어	사용자 및 그룹 항목에 대한 이름 공간이 포함된 조직 트리의 최상위 항목입니다.	o=usergroup)	
디렉토리 관리자 DN 및 비밀번호	조직 트리의 사용자 및 그룹 데이터를 담당하는 관리자입니다. 설치 프로그램에서 지정한 내용과 같아야 합니다. 디렉토리 관리자 DN의 비밀번호입니다.	cn=Directory Manager pAsSwOrD	

D.3 Messaging Server 초기 런타임 구성

Messaging Server 초기 런타임 구성 프로그램을 실행하는 경우 설치 매개 변수를 표 D–3에 기록하십시오. 또한 특정 질문에 답하기 위해 953 페이지 “D.1 Directory Server 설치” 확인 목록을 참조해야 할 수 있습니다.

표 D–3 초기 런타임 구성 매개 변수

매개 변수	설명	예	사용자의 답변:
구성 및 데이터 디렉토리	모든 Messaging Server 구성 파일이 포함됩니다. *msg-svr-base*/data 디렉토리가 이 디렉토리에 심볼릭 링크됩니다.	/var/mps/SUNmsgsr/	

표 D–3 초기 런타임 구성 매개 변수 *(계속)*

매개 변수	설명	예	사용자의 답변:
UNIX 시스템 사용자	실행하려는 프로세스에 대해 적절한 권한을 갖도록 하기 위해 시스템 사용자에게 지정되는 특정 권한입니다. 이 시스템 사용자는 Administration Server 초기 런타임 구성에서 지정한 사용자와 같으면 안 됩니다.	`mailsrv`	
UNIX 시스템 그룹	특정 UNIX 시스템 사용자가 속해 있는 그룹입니다. 이 시스템 그룹은 Administration Server 초기 런타임 구성에서 지정한 그룹과 같으면 안 됩니다.	`mail`	

구성 디렉토리 LDAP URL, 디렉토리 관리자 및 비밀번호	Configuration Directory Server, LDAP URL, 바인드 DN 및 비밀번호입니다.	`ldap://fiddle.west.sesta.com:389`
		`cn=Directory Manager`
		`PaSsWoRd`	

사용자 및 그룹 디렉토리 LDAP URL, 디렉토리 관리자 및 비밀번호	사용자 및 그룹 Directory Server, LDAP URL, 바인드 DN 및 비밀번호입니다.
	사용자 및 그룹 디렉토리와 구성 디렉토리를 별도로 지정하는 것이 좋습니다.
		`ldap://fiddle.west.sesta.com:389`
		`cn=Directory Manager`
		`PaSsWoRd`	

포스트마스터 전자 메일 주소	포스트마스터 메일을 모니터할 관리자의 전자 메일 주소입니다. 주소는 정규화된 주소여야 하며 해당 주소와 연결된 메일함이 있는 유효한 주소여야 합니다.	`pma@siroe.com`	
관리자 계정의 비밀번호	서비스 관리자, 사용자/그룹 관리자, 최종 사용자 관리자 권한 및 PAB 관리자와 SSL 비밀번호에 사용할 비밀번호입니다.	`paSSwoRD`	
기본 전자 메일 도메인	지정된 도메인이 없을 때 사용되는 전자 메일 기본값입니다.	`siroe.com`	

표 D–3 초기 런타임 구성 매개 변수 (계속)

매개 변수	설명	예	사용자의 답변:
기본 전자 메일 도메인의 조직 이름	조직이 속해 있으며 조직 트리를 구성하는 데 사용되는 조직 이름입니다.	예를 들어, 조직 이름이 Engineering인 경우 `siroe.com`(기본 전자 메일 도메인)의 모든 사용자는 LDAP DN `o=Engineering, o=usergroup` 아래에 배치됩니다. 사용자 및 그룹 디렉토리 접미어는 `comm_dssetup.pl`에 지정되어 있습니다.	

용어집

용어집

이 설명서 세트에 사용되는 전체 용어 목록은 **Sun Java Enterprise System Glossary**을 참조하십시오.

색인

B

C

D

E

F

G

H

J

K

L

M

N

O

P

Q

T

U

V

W

X

가

개

게

경

고

명

모

문

바

반

받

방

배