관리자 설명서

Sun™ ONE Web Server

버전 6.1

817-7510
2004 년 4 월

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

목차

본 설명서 정보

이 설명서에서는 Sun™ Open Net Environment (Sun ONE) Web Server 6.1 을 구성 및 관리하는 방법에 대하여 설명합니다 . 이 설명서는 기업 엔터프라이즈에서 WWW(World Wide Web) 를 통하여 클라이언트 - 서버 응용 프로그램을 더 많은 사람들에게 제공하려는 IT(Information Technology) 관리자를 위한 것입니다 .

이 서문의 내용 :

이 설명서의 내용

이 설명서에서는 Sun ONE Web Server 를 구성하고 관리하는 방법에 대하여 설명합니다 . 서버를 구성한 후 , 서버를 유지 보수하는 데 이 설명서를 사용합니다 .

서버를 설치하면 서버 루트 디렉토리의 `/manual/https/ag` 에 이 설명서의 HTML 형식 버전이 저장됩니다 . 기본으로 서버 루트 디렉토리는 `C:\Sun\WebServer6.1\` 또는 `/opt/SunWwbsvr` 입니다 .

이 설명서의 구성

이 설명서는 다섯 개의 부분과 용어집 및 전체 색인으로 구성됩니다 . Sun ONE Web Server 6.1 을 처음 사용하는 경우에는 제 1 부 " 서버 기본 " 에서 제품의 개요를 살펴보십시오 . 이 버전의 Sun ONE Web Server 에 이미 익숙한 경우에는 제 1 부 , " 서버 기본 " 을 대충 살펴보고 제 2 부 , "Administration Server 사용 " 으로 계속할 수 있습니다 .

Administration Server 사용에 대한 기초 지식이 갖추어지면 제 3 부 , " 구성 , 모니터링 및 성능 조정 , " 을 참조하십시오 . 여기에는 Sun ONE Web Server 를 구성하고 모니터하는 예제가 포함되어 있습니다 . 제 4 부 , " 가상 서버 및 서비스 관리 " 에서는 프로그램 및 구성 스타일 사용에 대하여 설명합니다 .

마지막으로부록 에서는 다음을 포함하여 다양한 주제에 대하여 설명하는 참조 내용이 있습니다 . 하이퍼 텍스트 전송 프로토콜 (HTTP), 서버 구성 파일 , ACL 파 일 , 국제화 문제 , 서버 확장 및 Sun ONE Web Server 사용자 인터페이스 참조 등 다시 찾아볼 수 있는 다양한 제목에 대한 참조가 들어 있습니다 . 참고로 사용자 인터페이스 부록은 온라인 버전으로만 제공됩니다 .

제 1 부 : 서버 기본

이 부분에서는 Sun ONE Web Server 의 개요를 살펴봅니다 . 다음과 같은 장이 포함됩니다 .

- 제 1 장 , "Sun ONE Web Server 소개 " 에서는 Sun ONE Web Server 에 대한 개요를 설명합니다 .
- 제 2 장 , "Sun ONE Web Server 관리 " 에서는 Administration Server 를 이용하여 Sun ONE Web Server 를 관리하는 방법에 대하여 설명합니다 .

제 2 부 : Administration Server 사용

이 부분에서는 Administration Server 를 사용하여 Sun ONE Web Server 를 관리하는 방법에 대한 개념 및 절차를 자세히 살펴봅니다 . 다음과 같은 장이 포함됩니다 .

- 제 3 장 , " 사용자 및 그룹 관리 " 에서는 Administration Server Preferences Users and Groups 폼을 사용하여 Sun ONE Web Server 를 구성하는 방법에 대하여 설명합니다 .

- 제 4 장 , " 웹 컨테이너 및 웹 응용 프로그램용 J2EE 기반 보안 " 에서는 SUN ONE Web Server 보안을 구성하는 방법과 ACL 기반 액세스 제어 및 Java™ 2 Platform, Enterprise Edition(J2EE™)/ 서브릿 기만 인증 및 권한 등 두 가지 보안 모델에 대하여 설명합니다 .
- 제 5 장 , " 관리 기본 설정 " 에서는 Administration Server Preferences and Global Settings 폼을 사용하여 Sun ONE Web Server 를 구성하는 방법에 대하여 설명합니다 .
- 제 6 장 , " 인증서 및 키 사용 " 에서는 인증서와 공용 키를 사용하여 보안을 향상시키는 방법에 대하여 설명합니다 . 이 장을 읽기 전에 공용키 암호화와 SSL(Secure Sockets Layer) 프로토콜의 기본적인 지식이 있어야 합니다 . 이들 개념에는 암호화 및 복호화 , 키 , 디지털 인증서 및 서명 , SSL 암호화 , 암호 , SSL 핸드셰이크의 주요 단계 등이 포함됩니다 .
- 제 7 장 , " 서버 클러스터 관리 " 에서는 서버 클러스터링의 개념을 설명하고 이를 사용하여 서버 사이에서 구성을 공유하는 방법에 대하여 설명합니다 .

제 3 부 : 구성 및 모니터링

이 부분에는 Server Manager 를 사용하여 Sun ONE Web Server 를 구성하고 모니터링하는 방법 예제가 포함됩니다 . 다음과 같은 장이 포함됩니다 .

- 제 8 장 , " 서버 기본설정 구성 " 에서는 Sun ONE Web Server 용 서버 기본 설정을 구성하는 방법에 대하여 설명합니다 .
- 제 9 장 , " 서버 액세스 제어 " 에서는 서버의 일정 부분에 액세스할 수 있는 사용자를 지정하는 방법에 대하여 설명합니다 .
- 제 10 장 , " 로그 파일 사용 " 에서는 하이퍼 텍스트 전송 프로토콜 (HTTP) 를 사용하거나 , 로그 파일을 기록 및 검토하거나 , 또는 운영 체제와 함께 제공되는 성능 모니터 도구를 사용하여 Sun ONE Web Server 를 모니터하는 방법에 대하여 설명합니다 .
- 제 11 장 , " 모니터 서비스 " 에서는 SNMP(Simple Network Management Protocol) 를 사용하여 Sun ONE Web Server 를 모니터하는 방법에 대하여 설명합니다 .
- 제 12 장 , " 이름 지정 구성과 리소스 " 에서는 서버에서 JNDI(Jave Naming and Description Interface) 리소스를 구성하고 데이터베이스 연결을 포함하는 방법에 대하여 설명합니다 .

제 4 부 : 가상 서버 및 서비스 관리

이 부분에는 Server Manager 를 사용하여 스타일을 프로그램하고 구성하는 데 대한 내용이 있습니다 . 다음과 같은 장이 포함됩니다 .

- 제 13 장 , " 가상 서버 사용 " 에서는 Sun ONE Web Server 를 사용하여 가상 서버를 설정하고 관리하는 방법에 대하여 설명합니다 .
- 제 14 장 , " 가상 서버 만들기 및 구성 " 에서는 각각의 가상 서버를 만들고 구성하는 방법에 대하여 설명합니다 .
- 제 15 장 , " 프로그램으로 서버 확장 " 에서는 서버에 Java 애플릿 , CGI 프로그램 , JavaScript 응용 프로그램 및 기타 플러그 인 등을 설치하는 방법에 대하여 설명합니다 .
- 제 16 장 , " 내용 관리 " 에서는 서버의 컨텐트를 구성하고 관리하는 방법에 대하여 설명합니다 .
- 제 17 장 , " 구성 스타일 적용 " 에서는 Sun One Web Server 에서 구성 스타일을 사용하는 방법에 대하여 설명합니다 .
- 제 18 장 , " 검색 사용 " 에서는 서버에 있는 문서의 컨텐츠와 속성을 검색하는 방법에 대하여 설명합니다 . 또한 사용자 커뮤니티에 맞게 텍스트 검색 인터페이스를 사용자 정의하는 방법에 대하여 설명합니다 .
- 제 19 장 , "WebDAV 를 사용하는 웹 게시 " 에서는 웹 게시와 맞춤 협력 웹 저작이 가능한 WebDAV 프로토콜을 사용할 수 있도록 가상 서버를 구성하는 방법에 대하여 설명합니다 .

제 5 부 : 부록

이 부분에는 나중에 찾아 볼 수 있는 다양한 참조 자료에 대한 부록이 포함되어 있습니다 . 다음과 같은 부록이 포함됩니다 .

- 부록 A, " 명령줄 유틸리티 " 에는 사용자 인터페이스 화면 대신 명령줄 유틸리티를 사용하는 방법이 있습니다 .
- 부록 B, "Hypertext Transfer Protocol" 에서는 몇 가지 HTTP 기본 개념을 간단히 소개했습니다 .
- 부록 C, "ACL 파일 구문 " 에서는 액세스 제어 목록 (ACL) 파일과 해당 구문을 설명합니다 .
- 부록 D, " 국제화 및 현지화용 지원 " 에서는 Sun ONE Web Server 의 해외 버전에 대하여 설명합니다 .

또한 용어집에는 Sun ONE Web Server 관리자에게는 생소할 수 있으나 자주 사용되는 용어가 있습니다 .

Sun ONE Web Server 설명서 사용

Sun ONE Web Server 설명서는 다음 웹사이트에서 PDF 와 HTML 형식으로 사용할 수 있습니다 .

http://docs.sun.com/db/prod/s1websrv#hic

Sun ONE Web Server 설명서에서 설명하는 작업과 개념의 목록은 다음 표와 같습니다 .

표 1) Sun ONE Web Server 설명서 과정표

내용	해당 부분
소프트웨어 및 설명서에 대한 최신 정보	*릴리스 노트*
Sun ONE Web Server 시작 , 서버 기본 및 기능을 소개하는 실습 포함 (처음 사용자에게 권장)	*시작 설명서*
설치 및 이전 작업 수행 : • Sun ONE Web Server 및 다양한 구성요소의 설치 , 지원 플랫폼 및 환경 • Sun ONE Web Server 4.1 또는 6.0 에서 Sun ONE Web Server 6.1 로 이전	*설치 및 이전 설명서*

표 1) Sun ONE Web Server 설명서 과정표

내용	해당 부분
다음의 관리 작업 수행 : • Administration 및 명령줄 인터페이스 사용 • 서버 기본설정 구성 • 서버 인스턴스 사용 • 서버 작동 모니터링 및 로깅 • 인증서 및 공용 키 암호화를 사용하여 서버 보안 • 액세스 제어 구성으로 서버 보안 • Java(TM) 2 Platform, Enterprise Edition(J2EE(TM) 플랫폼) 보안 기능 • 응용 프로그램 구현 • 가상 서버 관리 • 성능 요구에 맞추어 서버 작업 부하 정의 및 시스템의 규모 설정 • 서버 문서의 컨텐트 및 속성 검색 , 텍스트 검색 인터페이스 작성 • 컨텐트 압축용으로 서버 구성 • WebDAV 를 사용한 웹 게시 및 컨텐트 저작용으로 서버 구성	*관리자 설명서*
프로그래밍 기법 및 API 를 사용하여 다음 작업 수행 : • Sun ONE Web Server 확장 및 수정 • 클라이언트 요청에 대한 응답으로 동적 컨텐트 생성 • 서버의 컨텐트 수정	*Programmer's Guide*
사용자 정의 NSAPI(Netscape Server Application Programmer's Interface) 플러그인 생성	*NSAPI Programmer's Guide*
Sun ONE Web Server 에서 서브릿 및 JSP(TM)(JavaServer Pages(TM)) 기법 구현	*Programmer's Guide to Web Applications*
구성 파일 편집	*Administrator's Configuration File Reference Guide*

표 1) Sun ONE Web Server 설명서 과정표

내용	해당 부분
Sun ONE Web Server 를 조정하여 성능 최적화	*Performance Tuning, Sizing, and Scaling Guide*

문서 규약

이 부분에서는 이 설명서에서 사용된 표기 규약에 대하여 설명합니다 .

- **파일 및 디렉토리 경로**에는 UNIX 형식 (디렉토리 이름 사이를 슬래시 (/) 로 구분) 이 사용됩니다 . Windows 버전의 경우 디렉토리 경로는 같으나 디렉토리 사이를 역슬래시 (\) 로 구분합니다 .
- **URL** 에는 다음의 형식을 사용합니다 .

  `http://`*`server.domain/path/file`*`.html`

  이 URL 에서 **server** 는 응용 프로그램이 실행되는 서버 이름이며 , **domain** 은 사용하는 인터넷 도메인 이름 , **path** 는 서버의 디렉토리 구조 , **file** 은 각각의 파일 이름입니다 . URL 에서 기울임체로 표기된 항목은 자리 표시자입니다 .
- **글꼴 형식** :
  - `monospace` 글꼴은 예제 코드와 코드 목록, API 및 언어 요소(함수 이름이나 클래스 이름 등), 파일 이름 , 경로 이름 , 디렉토리 이름 , HTML 태그 등에 사용됩니다 .
  - *기울임체*는 코드 변수에 사용됩니다 .
  - 또한 책제목 , 강조 , 변수 및 자리 표시 , 문자 그대로의 뜻을 전달하기 위한 단어 등에도 *기울임체*를 사용합니다 .
  - **굵은체** 는 문단을 시작할 때 , 또는 문자 그대로의 뜻을 전달하는 데 사용된 단어를 표시할 때 사용합니다 .
- 이 문서에서 **설치 디렉토리**는 *install_dir* 로 표기합니다 .

UNIX 기반 플랫폼의 경우 기본 *install_dir*:

```
/opt/SUNWwbsvr/
```

Windows 의 경우 :

```
C:\Sun\WebServer6.1
```

제품 지원

시스템에 문제가 발생한 경우에는 다음의 메커니즘을 통하여 고객 지원에 문의하십시오 .

- 온라인 지원 웹 사이트 :

  `http://www.sun.com/supportraining/`

제 1 부

서버 기본

1 장

Sun ONE Web Server 소개

이 장에서는 Sun ONE Web Server 를 소개하고 몇 가지 기본적인 서버 개념에 대하여 알아봅니다 . 이 장을 통하여 Sun ONE Web Server 의 작동 원리를 알아보십시오 .

이 장의 내용 :

Sun ONE Web Server

Sun ONE Web Server 6.1 은 개방형 표준을 기반으로 구축된 복수 프로세스 , 복수 스레드 보안 웹 서버입니다 . 모든 규모의 기업을 위하여 높은 성능 , 신뢰도 , 확장성 및 관리 용이성을 제공합니다 .

이 부분에서는 Sun ONE Web Server 의 기능에 대하여 설명하고 수행할 수 있는 몇 가지 기본 관리 작업을 소개합니다 . 이 부분의 주제 :

Sun ONE Web Server 6.1 의 새로운 기능

Sun ONE Web Server 6.1 에 포함된 새로운 기능 :

Java Servlet 2.3 및 JSP(JavaServer Pages) 1.2 지원

Sun ONE Web Server 6.1 에는 Java™ 2 Platform, Enterprise Edition(J2EE™) 호환 Java™ Servlet 2.3 및 JSP™(JavaServer Pages™) 표준 구현이 포함됩니다. J2EE 호환 웹 컨테이너는 Java™ 기술 표준과 호환되는 웹 응용 프로그램을 디자인하고 배포할 수 있는 유연성과 신뢰성을 제공합니다 . 웹 응용 프로그램은 가상 서버 단위로 구현할 수 있습니다 .

이 기술에 대한 자세한 내용은 다음의 자료를 참조하십시오 .

Java Servlets
http://java.sun.com/products/servlet/index.jsp

Java Servlet 2.3 사양
http://java.sun.com/products/servlet/download.html

JavaServer Pages
http://java.sun.com/products/jsp/index.jsp

Sun ONE Web Server 에서 서브릿 및 JSP 를 구현하는 방법에 대한 내용은 Sun ONE Web Server 6.1 *Programmer's Guide to Web Applications* 를 참조하십시오 .

JDK 1.4.1_03 지원

Sun ONE Web Server 6.1 은 JDK™(Java Developer's Kit) 1.4.1_03 을 지원합니다 . 이 JDK 는 Web Server 와 함께 번들로 제공되며 설치할 때 (설치를 선택하는 경우) 함께 설치됩니다 . 또한 Web Server 를 설치한 후 보유하고 있는 JDK 를 따로 설치할 수 있습니다 . Administration Server 와 Java 및 서브릿 지원을 사용하려는 경우 반드시 JDK 를 설치해야 합니다 .

WebDAV 지원

Sun ONE Web Server는 WebDAV(Web-based Distributed Authoring and Versioning) 프로토콜을 지원합니다 . 이 프로토콜을 사용하면 다음 기능을 포함하여 협력 웹 게시가 가능합니다 .

RFC 2581 표준 및 RFC 2581 클라이언트와의 상호 운용성

- 웹 게시용 보안 및 액세스 제어
- 파일 시스템 기반 WebDAV 컬렉션 및 리소스에 대한 기본 게시 작업

WebDAV 는 컨텐츠 메타데이터 , 이름 공간 관리 및 덮어 쓰기 방지 등을 통합 지원합니다 . 이러한 기술은 WebDAV 를 지원하는 다양한 저작 도구와 조합되어 협력 환경용으로 이상적인 개발 플랫폼을 제공합니다 .

NSAPI 필터 지원

Sun ONE Web Server 6.1 은 NSAPI(Netscape Server Application Programmer's Interface) 가 NSAPI 필터를 지원할 수 있도록 확장합니다 .

필터를 사용하면 HTTP 요청 및 응답 스트림을 사용자에 맞게 처리할 수 있으므로 하나의 기능이 다른 기능에 제시되거나 다른 기능에 의하여 생성된 컨텐츠를 가로채거나 수정할 수 있습니다 . 예를 들어 플러그인이 NSAPI 필터를 설치하여 다른 플러그인의 SAF(Server Application Function) 이 생성한 XML 페이지를 가로챈 후 , 이 XML 페이지를 해당 클라이언트에 맞게 HTML, XHTML 또는 WAP 페이지로 변환할 수 있습니다 . 또한 , NSAPI 필터는 클라이언트로부터 받은 데이터를 다른 플로그인에 보내기 전에 압축할 수 있습니다 .

더 자세한 내용은 Sun ONE Web Server 6.1 *NSAPI Programmer's Guide* 를 참조하십시오 .

HTTP 압축

Sun ONE Web Server 6.1 은 컨텐츠 압축을 지원하므로 하드웨어 비용의 증가 없이 클라이언트와 서버에 더 많은 컨텐츠를 더 빠르게 제공할 수 있습니다 . 내용 압축은 내용 다운로드 시간을 단축시키므로 전화 접속 및 높은 수준의 트래픽 연결 사용자는 더 많은 혜택을 누릴 수 있습니다 .

더 자세한 내용은 Sun ONE Web Server 6.1 *관리자 설명서를 참조하십시오* .

새로운 검색 엔진 지원

Sun ONE Web Server 6.1 은 텍스트 전용 색인화 및 검색을 제공하는 새로운 Java 기반 검색 엔진을 지원합니다 . 이 검색 기능을 사용하면 서버에서 문서를 검색하고 결과를 웹 페이지에 표시할 수 있습니다 . 서버 관리자는 사용자의 검색에 대한 문서 색인을 만들 수 있으며 사용자는 자신의 요구에 맞추어 검색 인터페이스를 만들 수 있습니다 .

더 자세한 내용은 Sun ONE Web Server 6.1 *관리자 설명서* 를 참조하십시오 .

보안 기능 향상

Sun ONE Web Server 6.1 의 새로운 기능을 사용하면 보통 파일 인증을 통하여 액세스를 제한할 수 있습니다 . 또한 Sun ONE Web Server 6.1 은 이전 버전의 Web Server 와 달리 Java Security Manager 를 지원합니다 . Java Security Manager 는 제품을 설치할 때 기본적으로 사용하지 않도록 설정됩니다 . `server.xml` 에에 대한 자세한 내용은 Sun ONE Web Server 6.1 *Administrator's Configuration File Reference* 를 참조하십시오 .

JNDI 지원

Sun ONE Web Server 6.1 은 JNDI(Java Naming and Directory Interface™) 를 지원하므로 다양한 종류의 기업 이름 지정 및 디렉토리 서비스에 대한 매끄러운 연결을 제공합니다 .

JDBC 지원

Sun ONE Web Server 에서는 JDBC™(Java Database Connectivity) 를 바로 사용할 수 있으며 광범위한 업계 표준 및 사용자 정의 JDBC 드라이버를 지원합니다 .

Sun ONE Studio 5 지원

Sun ONE Web Server 6.1 은 Sun™ ONE Studio 5, Standard Edition 을 지원합니다 . Sun ONE Studio 기술은 Sun 의 강력하며 확장 가능한 Java 기술 개발자용 통합 개발 환경 (IDE) 입니다 . Sun ONE Studio 5 는 NetBean™ 소프트웨어를 기반으로 만들어졌으며 Sun ONE 플랫폼과 통합되었습니다 . (Sun ONE Web Server 6.1 은 또한 NetBeans 3.5 및 3.5.1 을 지원합니다 .)

Sun ONE Studio 지원은 Sun ONE Web Server 6.1 이 지원하는 모든 플랫폼에서 사용할 수 있습니다 . Web Server 용 플러그인은 다음과 같은 방법으로 구할 수 있습니다 .

- Sun ONE Web Server 6.1 미디어 키트의 Companion CD.
- Sun ONE Studio 의 AutoUpdate 기능 사용
- Sun ONE Web Server 의 다운로드 센터 : http://www.sun.com/software/download/inter_ecom.html

Sun ONE Web Server 6.1 용 Sun ONE Studio 5 플러그인은 오직 로컬 Web Server 에서만 작동한다는 점에 유의해야 합니다 . (즉 , IDE 와 Web Server 가 동일한 컴퓨터에 있는 경우에만 작동합니다 .)

Sun ONE Web Server 6.1 용 Sun ONE Studio 5 의 작동은 Sun™ ONE Application Server 7 의 작동과 같습니다 . Sun ONE Studio 5 에서 웹 응용 프로그램 기능을 사용하는 데 대한 자세한 내용은 다음 위치의 자습서를 참조하십시오 .
http://developers.sun.com/tools/javatools/documentation/s1s5/cdshop.pdf

Sun ONE Web Server 6.1 인스턴스를 기본으로 설정한 후 , 자습서에 설명한 것과 동일하게 조치합니다 .

또한 NetBeans 자습서를 참조하십시오 .
http://usersguide.netbeans.org/tutorials/webapps/index.html

Sun ONE Studio 에 대한 자세한 내용은 다음을 방문하십시오 .
http://www.sun.com/software/sundev/jde/

NSS 3.3.5 및 NSPR 4.1.5 지원

Sun ONE Web Server 6.1 은 NSS(Network Security Services) 3.3.5 및 NSPR(Netscape Portable Runtime) 4.1.5 를 지원합니다 .

PHP 호환성

Sun ONE Web Server 6.1 은 다기능성의 널리 사용되는 Open Source 웹 스크립트 언어인 PHP 와 호환됩니다 . PHP(Hypertext Preprocessor 의 약자) 는 모든 주요 운영 체제에서 작동합니다 .

Sun ONE Web Server 6.1 에는 PHP 버전 4.3.2 를 권장합니다 . Sun ONE Web Server 용 PHP 관련 설치 및 구성에 대한 자세한 내용은 다음을 참조하십시오 .
http://www.php.net/manual/en/install.netscape-enterprise.php

고급 하드웨어 가속기 암호화 지원

Sun ONE Web Server 6.1 은 Sun™ Crypto Accelerator 1000 용 하드웨어 가속기 지원을 제공합니다 . Sun Crypto Accelerator 1000 은 암호화 가속기 보드로 Web Server 의 SSL 성능을 향상시킵니다 .

Start on Boot 옵션

UNIX 플랫폼의 경우 Sun ONE Web Server 6.1 에 Start on Boot 옵션이 채택되었으므로 시스템을 부팅할 때 Web Server 가 자동으로 시작되도록 구성할 수 있습니다 . 더 자세한 내용은 Sun ONE Web Server 6.1 *설치 및 이전 설명*서를 참조하십시오 .

추가 기능

복수 프로세스 및 프로세스 모니터 , 장애 복구 , 자동 복구 및 동적 로그 순환 등을 지원합니다 .

Sun ONE Web Server 운용 및 관리

다음과 같은 사용자 인터페이스를 통하여 Sun ONE Web Server 를 관리할 수 있습니다 .

- Sun ONE Web Server Administration Server
- Server Manager
- Class Manager
- Virtual Server Manager

이전 릴리스에서 Web Server 와 기타 Netscape 서버는 Administration Server 라고 하는 단일 서버에 의하여 관리되었습니다 . 4.x 릴리스의 경우 " 관리 서버 " 는 단순히 Sun ONE Web Server 에 추가되는 인스턴스가 되었으며 Sun ONE Web Server Administration Server 또는 Administration Server 라고 합니다 . Administration Server 를 사용하여 모든 Sun ONE Web Server 인스턴스를 관리할 수 있습니다 . 더 자세한 내용은 "Administration Server" 페이지 37 를 참조하십시오 .

참고	또한 구성 파일을 편집하거나 명령줄 유틸리티를 사용하여 직접 관리 작업을 수행할 수 있습니다 .

Sun ONE Web Server 의 개별 인스턴스를 관리하는 경우 Server Manager 를 이용할 수 있습니다 . 더 자세한 내용은 "Server Manager" 페이지 38 를 참조하십시오 .

가상 서버를 관리하려면 Class Manager 를 사용합니다 . 더 자세한 내용은 "Class Manager" 페이지 39 를 참조하십시오 .

Sun ONE Web Server 구성

Sun ONE Web Server 는 다양한 기능을 사용 또는 사용 안 함으로 설정하거나 , 개별 클라이언트 요청에 대한 응답 방식을 결정하거나 , 서버에서 실행되며 서버의 작업과 통합되는 프로그램을 작성할 수 있도록 구성됩니다 . 이를 구분하는 지시 (명령) 는 구성 파일에 저장됩니다 . Sun ONE Web Server 는 시작할 때와 클라이언트 요청이 있을 때 구성 파일을 읽어 선택사항을 원하는 서버 작업과 매핑합니다 .

이 파일에 대한 자세한 내용은 Sun ONE Web Server 6.1 *Administrator's Configuration File Reference* 를 참조하십시오 .

Administration Server

Administration Server 는 모든 Sun ONE Web Server 를 구성하는데 사용하는 폼이 들어 있는 웹 기반 서버입니다 .

Sun ONE Web Server 를 설치한 후 브라우저를 사용하여 Administration Server 로 이동하고 , 해당 폼을 사용하여 Sun ONE Web Server 를 구성합니다 . 폼을 제출하면 Administration Server 가 해당 관리 대상의 서버에 대한 구성을 변경합니다 .

Administration Server 페이지로 이동하는 데 사용하는 URL 은 Sun ONE Web Server 를 설치할 때 Administration Server 용으로 선택한 컴퓨터 호스트 이름과 포트 번호에 따라 달라집니다 . 예를 들어 , Administration Server 를 포트 1234 에 설치한 경우 URL 은 다음과 같게 됩니다 .

```
http://myserver.sun.com:1234/
```

폼을 사용하기 전에 Administration Server 에 인증을 요구하는 프롬프트가 표시됩니다 . 따라서 사용자 이름과 암호를 입력해야 합니다 . Sun ONE Web Server 를 컴퓨터에 설치할 때 " 수퍼유저 " 사용자 이름과 암호를 설정했을 것입니다 . 전형적인 인증 화면은 그림에 보이는 것과 같습니다 .

설치 후 , 분산된 관리를 이용하여 여러 사람에게 Administration Server 의 다양한 폼에 액세스하는 권한을 부여할 수 있습니다 . 분산 관리에 대한 더 자세한 내용은 제 5 장 , " 관리 기본 설정 " 의 " 복수 관리자 허용 " 페이지 100 을 참조하십시오 .

Administration Server 용 설정은 오른쪽 창에 표시되며 탭으로 구성되어 있습니다 .

Administration Server 에 액세스하면 처음 표시되는 페이지는 Servers 입니다 . 이 페이지의 버튼을 사용하여 Sun ONE Web Server 를 관리 , 추가 , 제거 및 이전할 수 있습니다 . Administration Server 에는 다음과 같은 관리 수준 작업용 탭이 있습니다 .

- Servers
- Preferences
- Global Settings
- Users and Groups
- Security
- Cluster Mgmt (Cluster Management)

참고 서버를 구성하는 데 필요한 CGI 프로그램을 실행하려면 브라우저가 쿠키를 사용하도록 설정해야 합니다 .

이들 관리 수준 작업을 포함하여 Administration Server 에 대한 더 자세한 내용은 "Sun ONE Web Server 관리 " 페이지 45 를 참조하십시오 .

Server Manager

Server Manager 는 모든 Sun ONE Web Server 의 개별 인스턴스를 구성하는데 사용하는 Java 폼이 들어 있는 웹 기반 서버입니다 .

Sun ONE Web Server 용 Server Manager 로 액세스하려면 다음과 같이 합니다 .

1. Sun ONE Web Server 를 설치하고 시작합니다 .

   Administration Server 에 Servers 페이지가 표시됩니다 .

2. Manage Servers 영역에서 원하는 서버를 선택하고 Manage 를 누릅니다 .

   Sun ONE Web Server 에 Server Manager Preferences 페이지가 표시됩니다 .

참고 참고로 서버를 구성하는 데 필요한 CGI 프로그램을 실행하려면 브라우저가 쿠키를 사용하도록 설정해야 합니다 .

Preferences 페이지의 링크를 사용하여 스레드 풀 설정 등의 옵션을 관리하고 웹 서버를 사용 또는 사용하지 않도록 설정합니다 .

또한 Server Manager 에는 추가의 Sun ONE Web Server 관리 작업을 할 수 있는 다음의 탭이 있습니다 .

- Security
- Logs
- Monitor
- Virtual Server Class
- Java

자세한 내용은 온라인 도움말의 Server Manager 를 참조하십시오 .

Class Manager

Class Manager 는 가상 Sun ONE Web Server 를 구성하는데 사용하는 Java 폼이 들어 있는 웹 기반 인터페이스입니다 . 가상 서버용 사용자 인터페이스는 Server Manager 와 Class Manager 의 두 부분으로 구성됩니다 . Class Manager 에는 단일 클래스 또는 단일 가상 서버에 영향을 미치는 설정이 포함됩니다 . Class Manager 에 있는 클래스용 서비스를 설정하는 것뿐 아니라 가상 서버 (클래스의 구성원) 를 추가하고 개별 가상 서버용 설정을 구성할 수 있습니다 .

Sun ONE Web Server 용 Class Manager 로 액세스하려면 다음과 같이 합니다 .

1. Server Manager 에서 Virtual Server Class 탭을 누릅니다 .

   Server Manager 에 Manage a Class of Virtual Server 페이지가 표시됩니다 .

2. 드롭 다운 목록에서 가상 서버 클래스를 선택하고 Manage 를 누릅니다 .

   Sun ONE Web Server 에 Class Manager 의 Select a Virtual Server 페이지가 표시됩니다 .

   또한 간단히 화면의 상단 오른쪽 구석에 있는 Class Manager 링크를 눌러 Class Manager 에 액세스할 수 있습니다 .

Class Manager 에는 Sun ONE Web Server 가상 서버를 관리할 수 있는 다음의 탭이 제공됩니다 .

- Virtual Servers
- Programs
- Content Management
- Styles

자세한 내용은 온라인 도움말의 Class Manager 를 참조하십시오 .

Virtual Server Manager

Virtual Server Manager 에 액세스하려면 Class Manager 의 Virtual Servers 탭으로 이동한 후 , Manager Virtual Servers 페이지에 있는 목록에서 가상 서버를 선택하고 Manage 를 누르거나 트리 보기의 아래에 있는 가상 서버로의 링크를 누릅니다 .

Virtual Server Manager 에 제공되는 페이지를 사용하여 상태와 설정을 확인하고 , Java 웹 응용 프로그램 상태를 ON 으로 설정하고 , 선택한 가상 서버용 보고서를 생성할 수 있습니다 .

Virtual Server Manager 에는 Sun ONE Web Server 가상 서버를 관리할 수 있는 다음의 탭이 제공됩니다 .

- Preferences
- Logs
- Web Applications
- WebDAV
- Search

Resource Picker 사용

대부분의 Server Manager 및 Class Manager 페이지는 Sun ONE Web Server 전체 또는 클래스 전체를 구성합니다 . 그러나 일부 페이지는 전체 서버 (또는 클래스) 또는 서버 (또는 클래스) 가 유지하는 파일 및 디렉토리를 구성합니다 . 이 페이지의 상단에는 Resource Picker 가 있습니다 .

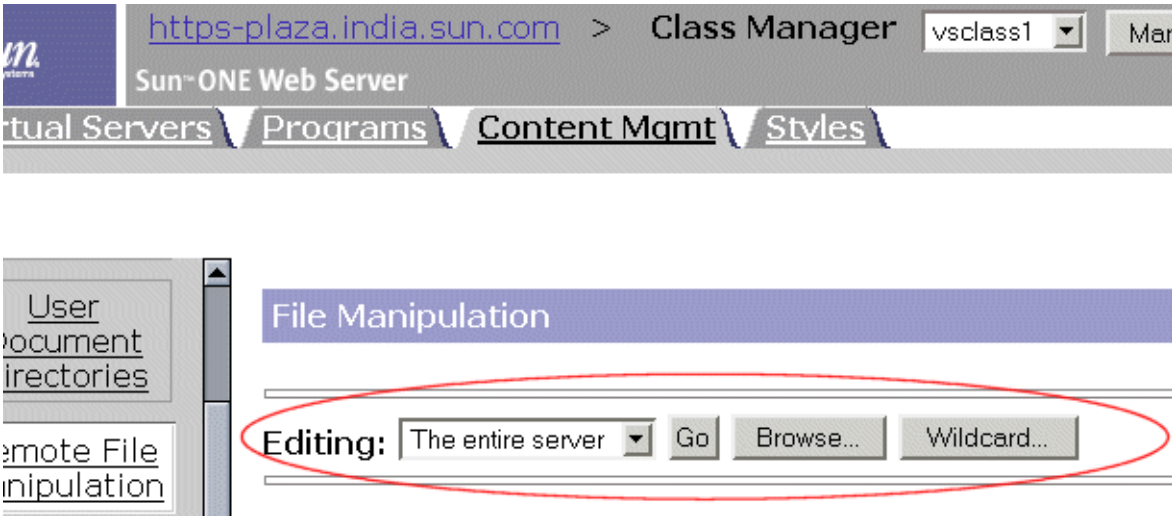


그림 1-1) Resource Picker

Resource Picker 는 Server Manager 의 Log Preferences 페이지와 Class Manager 의 Content Management 탭에서 액세스할 수 있는 대부분의 페이지 등 , 다양한 페이지에 표시됩니다 .

Resource Picker 를 사용하려면 구성용 드롭 다운 목록에서 해당 리소스를 선택합니다 . 기본 문서를 직접 찾으려면 Browse 를 누르고 , 특정 확장자가 있는 파일을 구성하려면 Wildcard 를 누릅니다 .

Resource Picker 에서 와일드카드 사용

서버 구성의 많은 부분에서 와일드카드 패턴을 사용하여 구성할 항목을 여러 개 표시할 수 있습니다 . 액세스 제어용 와일드카드는 이 부분에서 설명한 것과 다를 수 있으므로 유의하십시오 .

와일드카드 패턴에는 특수 문자가 사용됩니다 . 특별한 의미 없이 이들 문자를 사용하려면 문자 앞에 역슬래시 (\) 를 추가하십시오 .

와일드카드 패턴은 파일 이름뿐 아니라 디렉토리 경로에도 적용됩니다 . 따라서 특정 디렉토리에 있는 파일에만 적용될 수 있습니다 . 예를 들어 , /tmp/ 디렉토리에 파일을 추가하려면 tmp/*.html 과 같은 와일드카드 패턴을 지정할 수 있습니다 . 모든 디렉토리에 있는 index.html 파일을 추가하려면 */index.html 과 같은 패턴을 사용합니다 .

표 1) Resource Picker 와일드카드 패턴

패턴	용도
*	0 이상의 문자 일치 .
?	임의 문자가 오직 한 번만 나타나는 경우 일치 .
|	OR 표현식 이 연산자에서 사용된 하위 문자열에는 * 또는 $ 등의 특수 문자가 포함될 수 있습니다 . 예를 들어 , 하위 문자열을 괄호 안에 넣을 수 있으나 (a|b|c) 괄호는 중복하여 사용할 수 없습니다 .
$	문자열의 끝 일치 . OR 표현식에서 유용합니다 .
[abc]	a, b 또는 c 문자가 한 번 발생하는 경우 일치 . 이 표현식에서 특수 문자로 처리되어야 하는 문자는] 이며 다른 문자는 특수 문자가 아닙니다 .
[a-z]	a 에서 z 사이의 문자가 한 번 발생하는 경우 일치 .
[^az]	a 또는 z 를 제외한 임의의 문자 일치 .
*~	이 표현식 뒤에는 다른 표현식이 이어지며 두 번째 표현식의 모든 패턴 일치를 제거합니다 .

표 2) Resource Picker 와일드카드 예제

패턴	용도
*.sun.com	.sun.com 으로 끝나는 모든 문자열 일치 .
(products|docs).sun.com	products.sun.com 또는 docs.sun.com 검색 .
198.93.9[23].???	198.93.92 또는 198.93.93 으로 시작하며 끝에 세 개의 문자가 이어지는 숫자 문자열 검색 .
.	마침표를 포함하는 임의의 문자열 검색 .
~sun-	sun- 으로 시작하는 문자열 외의 임의의 문자열 검색 .
*.sun.com~docs.sun.com	단일 호스트 docs.sun.com 을 제외하고 sun.com 도메인에 있는 임의의 호스트 검색 .

표 2) Resource Picker 와일드카드 예제

패턴	용도
`*.sun.com~(products|docs|software).sun.com`	호스트 `products.sun.com`, `docs.sun.com` 및 `software.sun.com` 을 제외하고 `sun.com` 도메인에 있는 임의의 호스트 검색 .
`*.com~*.sun.com`	하위 도메인 `sun.com` 의 호스트를 제외하고 com 도메인에 있는 임의의 호스트 검색 .

2 장

Sun ONE Web Server 관리

이 장에서는 Sun ONE Web Server Administration Server 를 이용하여 Sun ONE Web Server 6.1 을 관리하는 방법에 대하여 설명합니다 . Administration Server 를 이용하면 서버 관리 , 서버 추가 및 제거 , 이전 릴리스에서의 서버 이전 등의 작업을 할 수 있습니다 .

이 장의 내용 :

Administration Server 시작

이 부분에서는 UNIX/Linux 및 Windows 버전용 Administration Server 에 액세스하는 방법에 대하여 설명합니다 .

UNIX/Linux 플랫폼

UNIX 또는 Linux 플랫폼에서 Administration Server 에 액세스하려면 다음과 같이 합니다 .

1. `server_root/https-admserv/` 디렉토리로 이동합니다 . (예 : `/usr/s1ws61/servers/https-admserv/`)

2. `./start` 를 입력합니다 .

   이 명령에 따라 Administration Server 가 시작되며 설치시 지정한 포트 번호가 사용됩니다 .

Windows 플랫폼

Sun ONE Web Server 설치 프로그램은 Windows 플랫폼용으로 여러 개의 아이콘이 있는 프로그램 그룹을 만듭니다 . 이 프로그램 그룹에는 다음의 아이콘이 포함됩니다 .

- Release Notes
- Start Web Server Administration Server
- Uninstall Web Server
- Administer Web Server

참고로 Administration Server 는 서비스 애플릿으로 실행되므로 제어판에서 이 서비스를 직접 시작할 수도 있습니다 .

Windows 플랫폼에서 Administration Server 에 액세스하려면 다음과 같이 합니다 .

1. "Start Web Server Administration Server" 아이콘을 두 번 누르거나 브라우저에서 다음 URL 을 입력하여 Administration Server 를 시작합니다 .

   `http://`*`hostname.domain-name:administration_port`*

   Sun ONE Web Server 에 사용자 이름과 암호를 묻는 프롬프트 창이 표시됩니다 .

2. 설치시 지정한 관리자 이름과 암호를 입력합니다 .

   Sun ONE Web Server 에 Administration Server 페이지가 표시됩니다 .

자세한 내용은 온라인 도움말의 Administration Server 페이지를 참조하십시오 .

참고 서버를 구성하는 데 필요한 CGI 프로그램을 실행하려면 브라우저가 쿠키를 사용하도록 설정해야 합니다 .

또한 Netscape Navigator 등의 클라이언트에 액세스할 수 있는 경우 원격 위치에서 Administration Server 에 액세스할 수 있습니다 . Administrator Server 는 브라우저를 통하여 액세스하므로 네트워크를 통하여 서버에 연결할 수 있는 모든 컴퓨터에서 액세스할 수 있습니다 .

복수 서버 실행

시스템에서 여러 개의 웹 서버를 실행하는 방법은 두 가지입니다 .

- 가상 서버 사용
- 복수 서버 인스턴스 설치

가상 서버

가상 서버를 사용하면 서버를 하나만 설치한 경우에도 여러 회사나 개인에게 도메인 이름 , IP 주소 및 일련의 서버 관리 기능을 제공할 수 있습니다 . 사용자의 경우 하드웨어와 기본적인 웹 서버 유지 보수를 제공해야 하지만 , 거의 자신의 전용 웹 서버를 가지고 있는 것과 같습니다 .

가상 서버용 설정은 `server.xml` 파일에 저장되어 있으며 , 이 파일은 *server_root/server_id*`/config` 디렉토리에 있습니다 . 가상 서버를 사용하기 위하여 이 파일을 편집할 필요는 없으나 , 이 파일에 대하여 더 자세히 알고 싶은 경우에는 Sun ONE Web Server 6.1 *Administrator's Configuration File Reference* 를 참조하십시오 .

가상 서버에 대한 자세한 내용은 제 13 장 , " 가상 서버 사용 " 을 참조하십시오 .

복수 서버 인스턴스 설치

Sun ONE Web Server 의 이전 버전에서는 가상 서버에 고유한 구성 정보가 없었습니다 . 서버에 개별적인 구성 정보가 포함되도록 하는 유일한 방법은 신규 서버 인스턴스를 만드는 것뿐이었습니다 . 그러나 Sun ONE Web Server 6.1 을 사용하면 가상 서버에 별도의 구성 정보가 포함될 수 있으므로 복수 서버 인스턴스가 더 이상 필요하지 않습니다 . 복수 서버가 여전히 지원되기는 하지만 복수 서버를 위한 좋은 방법은 아닙니다 .

복수 웹 서버를 설치하려고 선택한 경우 Administration Server 를 이용하여 다음의 작업을 수행합니다 .

- Windows 에 별도의 인스턴스로 서버 사본을 여러 개 설치하고 , 각각에 서로 다른 IP 주소를 지정합니다 .
- 모두 동일한 IP 주소를 사용하지만 포트 번호는 서로 다른 서버의 집합을 구성합니다 .

시스템이 복수 IP 주소를 청취하도록 구성된 경우에는 설치한 각 서버에 대하여 시스템이 호스팅하는 IP 주소를 한 개 입력합니다 .

시스템이 복수 IP 주소를 호스팅하도록 구성하기 전에 서버를 설치한 경우에는 시스템이 서로 다른 IP 주소에 응답하도록 구성합니다 . 그런 후 , 하드웨어 가상 서버를 설치하거나 , Server Manager 를 사용하여 서버의 바인드 주소를 변경하고 각 IP 주소에 대하여 별도의 서버 인스턴스를 설치합니다 .

다른 서버 인스턴스를 추가하려면 다음과 같이 합니다 .

1. Administration Server 에 액세스하고 Servers 탭을 선택합니다 .
2. Add Server 링크를 누릅니다 .
3. 지정된 필드에 원하는 정보를 입력합니다 .

   참고로 서버 ID 는 숫자로 시작할 수 없으며 인스턴스 이름에는 오직 Latin-1 문자만 사용해야 합니다 .
4. OK 를 누릅니다 .

자세한 내용은 온라인 도움말의 Add Server 페이지를 참조하십시오 .

서버 제거

시스템에서 Administration Server 를 이용하여 서버를 제거할 수 있습니다 . 이 작업은 복구할 수 없으므로 서버를 제거하기 전에 해당 서버가 필요 없는지 확인하십시오 .

참고 일부 Windows 서버에는 서버와 해당 관련 관리 서버를 제거할 수 있는 제거 프로그램이 있습니다 . 자세한 내용은 제품 설명서를 확인하십시오 .

컴퓨터에서 서버를 제거하려면 다음과 같이 합니다 .

1. Administration Server 에 액세스하고 Servers 탭을 선택합니다 .
2. Remove Server 를 누릅니다 .
3. 제거하려는 서버를 선택하고 Yes 를 클릭합니다 .
4. OK 를 누릅니다 .

Administration Server 는 서버의 구성 파일 , Server Manager 폼 및 다음 디렉토리 (하위 디렉토리 포함) 를 모두 삭제합니다 .

server_root/https-*server-id*

자세한 내용은 온라인 도움말의 Remove Server 페이지를 참조하십시오 .

이전 버전에서 서버 이전

4.1 또는 6.0 버전의 Sun ONE Web Server 에서 6.1 버전으로 이전할 수 있습니다 . 4.1 또는 6.0 서버는 보존되며 신규 6.1 서버는 동일한 설정을 사용합니다 .

설정을 이전하기 전에 4.1 또는 6.0 서버의 실행을 중단해야 합니다 . 설정을 이전하기 전에 컴퓨터에 호환되는 버전의 웹 브라우저가 설치되어 있는지 확인하십시오 .

이전 버전에서 Sun ONE Web Server 6.1 로 이전하는 방법에 대한 자세한 설명은 *설치 및 이전 설명서*를 참조하십시오 .

자세한 내용은 온라인 도움말의 Migrate Server 페이지를 참조하십시오 .

제 2 부

Administration Server 사용

3 장

사용자 및 그룹 관리

이 장에서는 Sun ONE Web Server 에 액세스할 수 있는 사용자 및 그룹을 추가 , 삭제 및 편집하는 방법에 대하여 설명합니다 .

이 장의 내용 :

사용자 및 그룹에 대한 정보 액세스

Administration Server 에서 사용자 계정 , 그룹 목록 , 액세스 권한 , 조직 단위 및 기타 사용자 및 그룹 특정 정보에 대한 응용 프로그램 데이터에 액세스할 수 있습니다 .

사용자 및 그룹 정보는 보통 파일 형식이나 Sun ONE Directory Server 와 같이 LDAP(Lightweight Directory Access Protocol) 을 지원하는 디렉토리 서버 안에 저장됩니다 . LDAP 는 개방형 디렉토리 액세스 프로토콜로 TCP/IP 에서 실행되며 전세계적 규모의 수 백만 항목을 수용하도록 확장될 수 있습니다 .

Sun ONE Web Server 는 로컬 LDAP 를 지원하지 않으므로 반드시 사용자와 그룹을 추가하기 전에 디렉토리 서버를 설치해야 합니다 .

Directory Service 설명

Sun ONE Directory Server 와 같은 디렉토리 서버를 사용하면 단일 소스에서 모든 사용자 정보를 관리할 수 있습니다 . 또한 사용자가 복수의 쉽게 액세스할 수 있는 네트워크 위치에서 디렉토리 정보를 검색할 수 있도록 디렉토리 서버를 구성할 수 있습니다 .

Sun ONE Web Server 6.1 의 경우 세 가지 디렉토리 서비스 유형을 구성하여 사용자 및 그룹을 인증 / 권한 부여할 수 있습니다 . 다른 디렉토리 서비스가 구성되지 않은 경우 디렉토리 서비스를 새로 만들면 , 이는 유형에 상관 없이 `default` 값으로 설정됩니다 .

디렉토리 서비스를 만들면 *`server-root`*`/userdb/dbswitch.conf` 파일이 디렉토리 서비스 세부 사항을 포함하여 업데이트됩니다 .

디렉토리 서비스 유형

Sun ONE Web Server 6.1 이 지원하는 세 가지 디렉토리 서비스는 다음과 같습니다 .

- **LDAP.** 사용자 및 그룹 정보를 LDAP 기반 디렉토리 서버에 저장합니다 .

  LDAP 서비스가 기본 서비스인 경우 `dbswitch.conf` 파일이 아래의 예와 같이 업데이트됩니다 .

  ```
  directory default
  ldap://draco.india.sun.com:589/dc%3Dindia%2Cdc%3Dsun%2Cdc%3Dcom

  default:binddn cn=Directory Manager

  default:encoded bindpw YWRtaW5hZG1pbg==
  ```

  LDAP 서비스가 기본 서비스가 아닌 경우 `dbswitch.conf` 파일이 아래의 예와 같이 업데이트됩니다 .

  ```
  directory ldap
  ldap://draco.india.sun.com:589/dc%3Dindia%2Cdc%3Dsun%2Cdc%3Dcom

  ldap:binddn cn=Directory Manager

  ldap:encoded bindpw YWRtaW5hZG1pbg==
  ```

- **Key File.** 키파일은 해시 형색의 사용자 비밀번호와 사용자가 속한 그룹 목록이 포함된 텍스트파일입니다. 키파일에 저장된 사용자 및 그룹은 오직 `file` 영역에 의하여만 인증 및 권한 부여에 사용되며 , 시스템 사용자 및 그룹과는 어떤 관계도 없습니다 . `file` 영역에 대한 자세한 내용은 " 파일 영역 " 을 참조하십시오 .

  키파일 기반 데이터베이스를 만들면 `dbswitch.conf` 파일이 다음 예와 같이 업데이트됩니다 .

  ```
  directory keyfile file

  keyfile:syntax keyfile

  keyfile:keyfile D:\draco\keyfile\keyfiledb
  ```

- **Digest File.** 암호화된 사용자 이름 및 비밀번호에 기반한 사용자 및 그룹 정보가 저장됩니다 .

  키파일 기반 데이터베이스를 만들면 `dbswitch.conf` 파일이 다음 예와 같이 업데이트됩니다 .

  ```
  directory digest file

  digest:syntax digest

  digest:digestfile D:\draco\digest\digestdb
  ```

참고 분산 관리를 설정하려는 경우 기본 디렉토리 서비스는 반드시 LDAP 기반 디렉토리 서비스여야 합니다 .

디렉토리 서비스 구성

디렉토리 서비스 기본 설정을 구성하려면 다음과 같이 합니다 .

1. Administration Server 에 액세스하고 Global Settings 탭을 선택합니다 .
2. Configure Directory Service 링크를 누릅니다 .
3. Create New Service of Type 드롭 다운 목록에서 만들려는 디렉토리 서비스 유형을 선택합니다 .

4. New 를 누릅니다 .

   이제 선택한 디렉토리 서비스 유형에 해당하는 페이지에서 디렉토리 서비스 정보를 구성할 수 있습니다 .

참고 다른 디렉토리 서비스가 구성되지 않은 경우 , 새로 작성된 디렉토리 서비스가 유형에 상관 없이 기본값으로 설정됩니다 .

5. Save Changes 를 눌러 변경 사항을 저장합니다 .

디렉토리 서비스를 만들고 구성했으면 각 가상 서버에 디렉토리 서비스를 지정할 수 있습니다 . 디렉토리 서비스에 연결된 권한과 허가는 이후 서버가 액세스 제어 규칙을 평가 및 집행하는 데 사용됩니다 . 더 자세한 내용은 " 가상 서버에 대한 디렉토리 서비스 선택 " 를 참조하십시오 .

DN(Distinguished Name) 이해

사용자 , 그룹 및 조직 단위를 만들거나 수정하려면 Administration Server 의 Users and Groups 탭을 사용합니다 . 사용자는 회사 고용인 등의 LDAP 데이터베이스에 있는 개인입니다 . 그룹은 공통 속성을 공유하는 둘 이상의 사용자입니다 . 조직 단위는 회사 내의 하위 부서로 `organizationalUnit` 개체 클래스를 사용합니다 . 사용자 , 구룹 및 조직 단위는 이 장의 나중에 자세히 설명합니다 .

기업의 각 사용자와 그룹은 DN(Distinguished Name) 속성으로 구분됩니다 . DN 속성은 연결된 사용자 , 그룹 또는 개체에 대한 구분 정보가 있는 문자열입니다 . 사용자 또는 그룹 디렉토리 항목을 변경하는 경우 항상 DN 을 사용합니다 . 예를 들어 디렉토리 항목을 작성 / 수정 , 액세스 제어 설정 , 전자우편이나 게시 등의 응용 프로그램용 사용자 계정 설정 등의 작업을 할 때 항상 DN 정보를 지정해야 합니다 . Sun ONE Web Server Administration Console 의 사용자 및 그룹 인터페이스를 사용하면 DN 을 만들거나 수정하는데 도움이 됩니다 .

다음의 예는 전형적인 Sun Microsystems 의 고용인용 DN 입니다 .

```
uid=doe,e=doe@sun.com,cn=John Doe,o=Sun Microsystems Inc.,c=US
```

이 예에서 각 등호 (=) 앞의 약자는 다음의 의미입니다 .

- `uid`: 사용자 ID
- `e`: email address

- cn: 사용자의 공통 이름
- o: 조직
- c: 국가

DN 에는 다양한 이름 - 값 쌍이 있을 수 있습니다 . 이는 LDAP 를 지원하는 디렉토리의 인증서 개체 및 항목을 확인하는데 사용됩니다 .

LDIF 사용

현재 디렉토리가 없으나 기존 디렉토리에 새 하위 트리를 추가하려는 경우 Directory Server 의 Administration Server LDIF 가져오기 기능을 사용할 수 있습니다 . 이 기능은 LDIF 가 포함된 파일을 받아서 LDIF 항목에서 디렉토리를 구축하거나 새 하위 트리를 만듭니다 . 또한 Directory Server 의 LDIF 내보내기 기능을 사용하여 현재 디렉토리를 LDIF 로 내보낼 수 있습니다 . 이 기능은 디렉토리에 대한 LDIF 형식 파일을 만듭니다 . ldapmodify 명령어와 적절한 LDIF 업데이트문을 사용하여 항목을 추가하거나 편집합니다 .

LDIF 를 사용하여 데이터베이스에 항목을 추가하려면 우선 `LDIF` 파일에 항목을 정의한 후 , Directory Server 에서 `LDIF` 파일을 가져옵니다 .

사용자 생성

사용자 항목 만들거나 수정하려면 Administration Server 의 Users and Groups 탭을 사용합니다 . 사용자 항목에는 데이터베이스이 개인 또는 개체에 대한 정보가 포함됩니다 .

사용자를 만들 때 반드시 사용자가 리소스에 무단 액세스할 수 없도록 하여 서버 보안을 보호해야 합니다 . Sun ONE Web Server 6.1 에서는 보안을 향상시킬 수 있는 다양한 옵션이 제공됩니다 .

- J2EE/Servlet 기반 영역 인증을 사용하여 사용자에게 인증 및 권한을 부여하는 방법에 대한 자세한 내용은 " 영역 기반 보안 " 페이지 87 을 참조하십시오 .
- ACL(Access Control List) 기반 인증 및 권한 부여 기법 사용에 대한 자세한 내용은 " 액세스 제어 작동 원리 " 페이지 186 를 참조하십시오 .
- Java 기반 보안 모델과 ACL 기반 보안 모델의 사이를 연결하는 Native Realm 기능의 사용에 대한 자세한 내용은 " 원시 영역 구성 " 페이지 93 을 참조하십시오 .

이 부분에서는 다음 항목에 대해 설명합니다 .

LDAP 기반 인증 데이터베이스에 신규 사용자 생성

LDAP 기반 디렉토리 서비스에 사용자 항목을 추가하는 경우 배후의 LDAP 기반 디렉토리 서버의 서비스가 사용자를 인증하고 권한을 부여하는 데 사용됩니다 . 이 부분에서는 LDAP 기반 인증 데이터베이스를 사용하는 경우 고려해야 할 지침과 Administration Server 를 통하여 사용자를 추가하는 방법에 대하여 설명합니다 .

LDAP 기반 사용자 항목 생성에 대한 지침

관리자 폼을 사용하여 LDAP 기반 디렉토리 서비스에 새 사용자 항목을 만드는 경우 다음의 지침을 고려하십시오 .

- 이름과 성을 입력하는 경우 폼에서 자동으로 사용자의 전체 이름과 사용자 ID 를 입력합니다 . 사용자 ID 는 사용자 이름의 첫 자와 사용자 성을 조합하여 만듭니다 . 예를 들어 사용자의 이름이 Billie Holiday 인 경우 사용자 ID 는 자동으로 bholiday 가 됩니다 . 원하는 경우 이 사용자 ID 는 원하는 ID 로 바꿀 수 있습니다 .
- 사용자 ID 는 반드시 고유해야 합니다 . Administration Server 는 검색 기반 (기본 DN) 에서 시작하여 전체 디렉토리에서 해당 사용자 ID 가 사용되는지 검색하여 해당 사용자 ID 가 고유한지 확인합니다 . 그러나 사용자를 만들 때 Directory Server ldapmodify 명령줄 유틸리티 (사용 가능한 경우) 를 사용하는 경우 고유한 사용자 ID 가 보장되지 않습니다 . 디렉토리의 사용자 ID 가 중복되는 경우 관련 사용자는 디렉토리에 대하여 인증되지 않습니다 .
- 참고로 기본 DN 은 디렉토리 검색이 기본으로 수행될 위치의 고유 이름과 디렉토리 트리에서 Sun ONE Web Administration Server 의 항목이 위치할 고유 이름을 지정합니다 . "DN" 은 디렉토리 서버에 있는 항목 이름을 문자열로 나타낸 것입니다 .
- 최소한 새 사용자 항목을 만들 때 반드시 다음의 사용자 정보를 지정해야 합니다 .
  - 성

- 전체 이름
- 사용자 ID

- 조직 단위가 디렉토리로 정의된 경우 Add New User To 목록을 사용하여 신규 사용자를 추가할 위치를 지정할 수 있습니다 . 기본 위치는 디렉토리의 기본 DN(또는 루트 지점) 입니다 .

참고 국제 정보용 사용자 편집 텍스트 필드는 Administration Server 의 국제 정보용 사용자 편집 텍스트 필드는 Sun ONE Web Server Administration Console 과 다릅니다 . Sun ONE Web Server Administration Console 의 경우 태그되지 않은 cn 필드에 더하여 기본 언어 cn 필드가 있으나 Administration Server 에는 없습니다 .

신규 사용자 항목 생성 방법

사용자 항목을 만들려면 "LDAP 기반 사용자 항목 생성에 대한 지침 " 페이지 58 에 간단히 설명한 지침을 읽고 다음과 같이 합니다 .

1. Administration Server 에 액세스하고 Users & Groups 탭을 선택합니다 .
2. New User 링크를 누릅니다 .
3. Select Directory Service 드롭 다운 목록에서 LDAP Directory Service를 선택하고 Select 를 누릅니다 .
4. 표시되는 페이지에 필요한 정보를 입력합니다 .

   더 자세한 내용은 디렉토리 서버 사용자 항목를 참조하십시오 .
5. OK 를 누릅니다 .

자세한 내용은 온라인 도움말의 New User 페이지를 참조하십시오 .

디렉토리 서버 사용자 항목

디렉토리 관리자는 다음의 항목 참고에 유의할 필요가 있습니다 .

- 사용자 항목은 `inetOrgPerson`, `OrganizationalPerson` 및 `person` 개체 클래스를 사용합니다 .
- 기본적으로 사용자용 고유 이름의 형식은 다음과 같습니다 .

  ```
  cn=full name, ou=organization, ...,o=base organization, c=country
  ```

  예를 들어 Billie Holiday 의 사용자 항목이 조직 단위 Marketing 안에 만들어졌으며 디렉토리의 기본 DN 이 o=Ace Industry, c=US 인 경우 이 사용자의 DN 은 다음과 같습니다 .

```
cn=Billie Holiday, ou=Marketing, o=Ace Industry, c=US
```

그러나 참고로 이 형식은 uid 기반 고유 이름으로 변경할 수 있습니다 .

- 사용자 폼 필드의 값은 다음의 LDAP 속성으로 저장됩니다 . (참고로 " 사용자 " 및 "group" 이 아닌 정보가 저장되는 경우 완전한 Directory Server 라이센스가 필요합니다 .)

표 3-1) LDAP 속성

사용자 필드	해당 LDAP 속성
이름	`givenName`
성	`sn`
전체 이름	`cn`
사용자 ID	`uid`
비밀번호	`userPassword`
전자우편 주소	`mail`

또한 사용자 항목을 편집하는 경우 다음 필드를 사용할 수 있습니다 .

표 3-2) 사용자 항목 LDAP 속성

사용자 필드	해당 LDAP 속성
제목	`title`
전화	`telephoneNumber`

- 기본 언어 외의 다른 언어 문자로 사용자 이름을 더 정확하게 표현할 수 있는 경우도 있습니다 . 기본 언어가 영어인 경우에도 사용자용 기본 언어를 선택하여 이름이 선택한 언어의 문자로 표시되도록 할 수 있습니다 . 사용자의 기본 언어를 설정하는 방법에 대한 자세한 내용은 온라인 도움말의 Manage Users 페이지를 참조하십시오 .

파일 기반 인증 데이터베이스에 신규 사용자 생성

Sun ONE Web Server 6.1 에서는 처음으로 사용자 정보를 텍스트 형식의 보통 파일에 저장하는 원시 인증 데이터베이스를 지원합니다 . 파일 기반 인증 데이터베이스는 다음 형식의 파일과 호환됩니다 .

- keyfile 스타일 파일
- digest 스타일 파일
- .htaccess 스타일 파일

새 사용자 항목 생성

파일 기반 인증 데이터베이스에 사용자 항목을 만들려면 다음과 같이 합니다 .

1. Administration Server 에 액세스하고 Users & Groups 탭을 선택합니다 .
2. New User 링크를 누릅니다 .
3. Select Directory Service 드롭 다운 목록에서 파일 기반 디렉토리 서비스 ID 를 선택하고 Select 를 누릅니다 .
4. 다음 정보를 입력합니다 .
   - **User ID.** (필수) 사용자용으로 고유한 사용자 이름을 지정합니다 .
   - **Password.** 사용자의 암호를 지정합니다 .
   - **Password (again).** 암호 필드에 입력한 암호를 확인합니다 .
   - **Groups.** 사용자가 속한 그룹의 목록을 쉼표로 분리하여 지정합니다 .
5. Create User 를 누릅니다 .

다이제스트 기반 인증 데이터베이스에 신규 사용자 생성

사용자 및 그룹 정보를 암호화된 형식으로 저장하는 다이제스트 기반 인증 데이터베이스에 사용자 항목을 만들려면 다음과 같이 합니다 .

1. Administration Server 에 액세스하고 Users & Groups 탭을 선택합니다 .
2. New User 링크를 누릅니다 .

3. Select Directory Service 드롭 다운 목록에서 다이제스트 기반 디렉토리 서비스 ID 를 선택하고 Select 를 누릅니다 .
4. 다음 정보를 입력합니다 .
   - **User ID.** (필수) 사용자용으로 고유한 사용자 이름을 지정합니다 .
   - **Realm.** 이 사용자를 인증할 영역을 지정합니다 .
   - **Password.** 사용자의 암호를 지정합니다 .
   - **Password (again).** 암호 필드에 입력한 암호를 확인합니다 .
   - **Groups.** 사용자가 속한 그룹의 목록을 쉼표로 분리하여 지정합니다 .
5. OK 를 누릅니다 .

사용자 관리

사용자 속성은 Administration Server Manage Users 폼에서 편집합니다 . 이 폼에서 사용자 항목을 검색 , 변경 , 이름 변경 및 삭제할 수 있으며 사용자 라이센스를 관리하고 , 때로 제품에 국한된 정보를 변경할 수도 있습니다 .

모두는 아니지만 일부 Sun ONE 서버에는 이 부분에 사용자가 제품에 국한된 정보를 관리할 수 있는 추가 폼이 있습니다 . 예를 들어 Administration Server 아래에 메시징 서버가 설치된 경우 메시징 서버에 대한 정보를 편집할 수 있는 폼이 추가됩니다 . 추가 관리 기능에 대한 자세한 내용은 서버 설명서를 참조하십시오 .

이 부분에서는 다음 항목에 대해 설명합니다 .

- 사용자 정보 검색
- 사용자 정보 편집
- 사용자 비밀번호 관리
- 사용자 라이센스 관리
- 사용자 이름 변경
- 사용자 제거

사용자 정보 검색

사용자 항목을 편집하려면 반드시 관련 정보가 표시되어야 합니다 . 특정 사용자 정보를 찾으려면 다음과 같이 합니다 .

1. Administration Server 에 액세스하고 Users & Groups 탭을 선택합니다 .
2. Manage Users 링크를 누릅니다 .
3. Find User 필드에서 편집하려는 항목에 대한 일부 기술적인 값을 입력합니다 . 검색 필드에 다음의 항목을 입력할 수 있습니다 .
   - 이름. 이름의 전체 또는 부분을 입력합니다. 검색 문자열과 일치되는 모든 항목이 검색됩니다 . 해당 항목을 찾을 수 없는 경우 검색 문자열을 포함하는 모든 항목이 검색됩니다 . 해당 항목을 찾을 수 없는 경우 검색 문자열과 소리가 유사한 모든 항목이 검색됩니다 .
   - 사용자 ID.
   - 전화번호 . 번호를 부분적으로 입력하면 검색 번호로 끝나는 전화번호를 포함하는 모든 항목이 검색됩니다 .
   - 전자우편 주소 . @ 을 포함하는 모든 검색 문자열은 전자우편 주소인 것으로 가정합니다 . 정확한 일치가 검색되지 않는 경우 검색은 검색 문자열로 시작하는 모든 전자우편 주소를 찾습니다 .
   - 디렉토리에 있는 모든 항목을 보려면 별표 (*) 를 사용합니다 . 필드에 아무런 값을 입력하지 않아도 동일한 효과를 얻을 수 있습니다 .
   - 임의 LDAP 검색 필터 . 등호 (=) 가 포함된 문자열은 검색 필터로 간주됩니다 .

   다른 방법으로 Find all users whose 필드의 드롭 다운 메뉴를 사용하여 검색 범위를 좁힐 수 있습니다 .
4. Look within 필드에서 검색하려는 항목의 상위 조직 단위를 선택합니다 .

   기본값은 디렉토리의 루트 지점 (또는 최상위 항목) 입니다 .
5. Format 필드에서 On-Screen 또는 Printer 를 선택합니다 .
6. Find 를 누릅니다 .

   선택한 조직 단위의 모든 사용자가 표시됩니다 .
7. 결과 테이블에서 편집하려는 항목의 이름을 누릅니다 .

   사용자 편집 폼이 표시됩니다 .
8. 표시된 필드를 원하는 대로 변경하고 Save Changes 를 누릅니다 .

   변경 사항은 즉시 적용됩니다 .

사용자 정의 검색 쿼리 작성

"Find all users whose" 필드를 사용하면 사용자 정의 검색 필터를 만들 수 있습니다. 이 필드를 사용하면 "Find User" 필드 검색의 결과를 더욱 세밀하게 할 수 있습니다.

Find all users whose 필드에는 다음의 검색 분류가 제공됩니다.

- 왼쪽의 드롭 다운 목록에서 검색 기준으로 사용할 속성을 지정할 수 있습니다.

사용 가능한 검색 속성 옵션은 다음 테이블과 같습니다.

표 3-3) 검색 속성 옵션

옵션 이름	설명
full name	각 항목의 전체 이름이 일치하도록 검색합니다.
last name	각 항목의 성이 일치하도록 검색합니다.
user id	각 항목의 사용자 ID 가 일치하도록 검색합니다.
phone number	각 항목의 전화번호가 일치하도록 검색합니다.
email address	각 항목의 전자우편 주소가 일치하도록 검색합니다.
unit name	각 항목의 이름이 일치하도록 검색합니다.
description	각 조직 단위 항목의 설명이 일치되는지 검색합니다.

- 가운데 드롭 다운 목록에서 수행할 검색 유형을 선택합니다.

사용 가능한 검색 유형 옵션은 다음 테이블과 같습니다.

표 3-4) 검색 유형 옵션

옵션 이름	설명
`contains`	하위 문자열 검색이 수행되도록 합니다. 지정된 검색 문자열을 포함하는 속성값이 있는 항목이 검색됩니다. 예를 들어 사용자 이름에 "Dylan" 라는 단어가 포함된 것을 알고 있는 경우 이 옵션을 검색 문자열 "Dylan" 과 함께 사용하여 해당 사용자 항목을 찾을 수 있습니다.
`is`	정확히 일치되는 항목을 검색합니다. 이 옵션은 일치 검색입니다. 사용자 속성의 값을 정확히 아는 경우 이 옵션을 사용합니다. 예를 들어 사용자 이름의 정확한 철자를 아는 경우 이 옵션을 사용합니다.

표 3-4) 검색 유형 옵션

옵션 이름	설명
`isn't`	검색 문자열과 정확히 일치하지 않는 속성값의 모든 항목을 검색합니다 . 즉 , 디렉토리에서 이름이 "John Smith" 가 아닌 사용자를 모두 찾으려는 경우 이 옵션을 사용합니다 . 그러나 이 옵션을 사용하면 지나치게 많은 수의 항목이 검색될 수 있습니다 .
`sounds like`	근사치 또는 발음에 의한 검색이 수행됩니다 . 속성의 값은 알지만 철자법이 확실하지 않은 경우 이 옵션을 사용합니다 . 예를 들어 사용자의 이름 철자가 "Sarret," "Sarette" 또는 "Sarett" 인지 확실하지 않은 경우 이 옵션을 사용합니다 .
`starts with`	하위 문자열 검색이 수행되도록 합니다 . 속성값이 지정된 검색 문자열로 시작하는 모든 항목을 검색합니다 . 예를 들어 , 사용자 이름이 "Miles" 로 시작되지만 나머지 이름은 알지 못하는 경우 이 옵션을 사용합니다 .
`ends with`	하위 문자열 검색이 수행되도록 합니다 . 속성값이 지정된 검색 문자열로 끝나는 모든 항목을 검색합니다 . 예를 들어 사용자 이름이 "Dimaggio" 으로 끝나지만 나머지 이름은 알지 못하는 경우 이 옵션을 사용합니다 .

- 오른쪽 텍스트 필드에 검색 문자열을 입력합니다 .

Look Within 디렉토리에 포함된 모든 사용자 항목을 표시하려면 별표 (*) 를 입력하거나 이 필드를 공란으로 남겨둡니다 .

사용자 정보 편집

사용자 항목을 변경하려면 다음과 같이 합니다 .

1. Administration Server 에 액세스하고 Users & Groups 탭을 선택합니다 .
2. "사용자 정보 검색 ,"(페이지 63) 에서 설명한 것과 같이 사용자 항목을 표시합니다 .
3. 변경하려는 속성에 해당하는 필드를 편집합니다 .

자세한 내용은 온라인 도움말의 Edit Users 페이지를 참조하십시오 .

참고 사용자 폼을 편집하여 표시되지 않은 속성 값을 변경해야 하는 경우도 있습니다 . 이 경우 사용할 수 있으면 Directory Server `ldapmodify` 명령줄 유틸리티를 사용하십시오 .

또한 이 폼에서 사용자의 이름 , 성 및 전체 이름을 변경할 수 있으나 항목의 이름 전체 (항목의 고유 이름 포함) 를 변경하려면 Rename User 폼을 사용해야 합니다 . 항목의 이름을 변경하는 방법은 페이지 67 의 " 사용자 이름 변경 ," 을 참조하십시오 .

사용자 비밀번호 관리

사용자 항목용으로 설정한 암호는 다양한 서버가 사용자를 인증할 때 사용됩니다 .

사용자의 비밀번호를 변경하거나 만들려면 다음과 같이 합니다 .

1. Administration Server 에 액세스하고 Users & Groups 탭을 선택합니다 .
2. " 사용자 정보 검색 ,"(페이지 63) 에서 설명한 것과 같이 사용자 항목을 표시합니다 .
3. 원하는 사항을 변경한 다음 OK 를 누릅니다 .

자세한 내용은 온라인 도움말의 Manage Users 페이지를 참조하십시오 .

참고 운영 시스템의 Administration Server 사용자를 루트에서 다른 사용자로 변경하여 여러 사용자 (그룹에 속한 사용자) 가 구성 파일을 편집 / 관리하도록 할 수 있습니다 . 그러나 UNIX/Linux 플랫폼의 경우 설치자는 그룹에게 구성 파일에 대한 "rw" 권한을 부여할 수 있는 반면 Windows 플랫폼의 경우 사용자는 반드시 "Administrators" 그룹에 속해야 합니다 .

또한 Disable Password 버튼을 눌러 사용자의 비밀번호를 사용하지 않도록 설정할 수 있습니다 . 이렇게 하면 사용자가 사용자의 디렉토리 항목을 삭제하지 않고 서버에 로그인 할 수 없도록 방지합니다 . 사용자가 다시 액세스할 수 있도록 하려면 Password Management Form 을 사용하여 새 비밀번호를 입력합니다 .

사용자 라이센스 관리

Administration Server 에서 사용자에게 라이센스가 부여된 Sun ONE 서버 제품을 추적할 수 있습니다 .

사용자가 사용할 수 있는 라이센스를 관리하려면 다음과 같이 합니다 .

1. Administration Server 에 액세스하고 Users & Groups 탭을 선택합니다 .

2. "사용자 정보 검색,"(페이지 63) 에서 설명한 것과 같이 사용자 항목을 표시합니다 .
3. User Edit 폼의 상단에 있는 Licenses 링크를 누릅니다 .
4. 원하는 사항을 변경한 다음 OK 를 누릅니다 .

자세한 내용은 온라인 도움말의 Manage Users 페이지를 참조하십시오 .

사용자 이름 변경

이름 변경 기능은 오직 사용자의 이름만 변경하며 기타 필드는 그대로 유지됩니다 . 또한 사용자의 이전 이름은 여전히 유지되므로 이전 이름으로 검색해도 새 항목을 찾을 수 있습니다 .

사용자 항목의 이름을 변경하는 경우 오직 사용자의 이름만 변경하게 되며 , 이름 변경 기능을 사용하여 항목을 하나의 조직 단위에서 다른 단위로 옮길 수는 없습니다 . 예를 들어 Marketing 과 Accounting 의 두 조직 단위가 있으며 "Billie Holiday" 는 Marketing 조직 단위에 속한 것으로 가정합니다 . 항목의 이름을 Billie Holiday 에서 Doc Holiday 로 변경할 수는 있으나 Marketing 조직 단위에 있는 Billie Holiday 가 Accounting 조직 단위에 있는 Billie Holiday 가 되도록 변경할 수는 없습니다 .

사용자 항목의 이름을 변경하려면 다음과 같이 합니다 .

1. Administration Server 에 액세스하고 Users & Groups 탭을 선택합니다 .
2. "사용자 정보 검색,"(페이지 63) 에서 설명한 것과 같이 사용자 항목을 표시합니다 .

   참고로 공통 이름 기반 DN 을 사용하는 경우 사용자의 전체 이름을 지정하십시오 . uid 기반 고유 이름을 사용하는 경우에는 항목용으로 사용하려는 신규 uid 값을 입력합니다 .
3. Rename User 버튼을 누릅니다 .
4. Given Name, Surname, Full Name 또는 UID 필드를 항목의 새로운 고유 이름에 맞게 적절히 변경합니다 .
5. 항목의 이름을 변경할 때 Administration Server가 더 이상 이전의 전체 이름 또는 uid 값을 보관하지 않도록 지정하려면 keepOldValueWhenRenaming 매개 변수를 false 로 설정합니다 . 이 매개 변수는 다음의 파일에서 찾을 수 있습니다 .

   server_root/admin-serv/config/dsgw-orgperson.conf

자세한 내용은 온라인 도움말의 Manage Users 페이지를 참조하십시오 .

사용자 제거

사용자 항목을 삭제하려면 다음과 같이 합니다 .

1. Administration Server 에 액세스하고 Users & Groups 탭을 선택합니다 .
2. "사용자 정보 검색 ,"(페이지 63) 에서 설명한 것과 같이 사용자 항목을 표시합니다 .
3. Delete User 를 누릅니다 .

자세한 내용은 온라인 도움말의 Manage Users 페이지를 참조하십시오 .

그룹 생성

그룹은 LDAP 데이터베이스에 있는 일련의 개체를 기술하는 개체입니다 . Sun ONE Web Server 그룹은 공통 속성을 공유하는 사용자로 구성됩니다 . 예를 들어 일련의 개체는 회사의 마케팅 부서에서 일하는 다수의 고용인일 수 있습니다 . 이들 고용인은 Marketing 이라는 이름의 그룹에 속할 수 있습니다 .

그룹의 구성원은 정적인 방법과 동적인 방법으로 정의할 수 있습니다 . 정적 그룹은 구성원 개체를 명시적으로 열거합니다 . 정적 그룹은 CN 이며 `uniqueMembers`, `memberURL` 및 `memberCertDescriptions` 를 포함합니다 . 정적 그룹의 경우 구성원은 `CN=<Gropuname>` 속성을 제외한 공통 속성을 공유하지 않습니다 .

동적 그룹을 사용하면 LDAP URL 을 사용하여 그룹 구성원에만 적용되는 일련의 규칙을 정의할 수 있습니다 . Dynamic Group 의 경우 구성원은 공통 속성 또는 `memberURL` 필터에 정의된 일련의 속성을 공유합니다 . 예를 들어 Sales 의 모든 고용인을 포함하는 그룹이 필요하며 이들이 이미 LDAP 데이터베이스의

"`ou=Sales,o=Airius.com`" 에 있는 경우 다음의 `memberurl` 로 동적 그룹을 정의합니다 .

```
ldap:///ou=Sales,o=sun??sub?(uid=*)
```

결과적으로 그룹에는 "`ou=Sales,o=sun`" 지점 아래의 트리에 있는 `uid` 속성이 포함되며 , 따라서 모든 Sales 구성원이 포함됩니다 .

정적 및 동적 그룹의 경우 `memberCertDescription` 을 사용하면 구성원이 인증서에 있는 공통 속성을 공유할 수 있습니다 . 참고로 이는 ACL 이 SSL 메소드를 사용하는 경우에만 작동합니다 .

새 그룹을 만들었으면 그룹에 사용자 또는 구성원을 추가할 수 있습니다 .

이 부분에서는 다음 항목에 대해 설명합니다 .

정적 그룹

Administration Server 을 사용하면 DN 에서 동일한 그룹 속성을 지정하여 원하는 수의 사용자에 대한 정적 그룹을 만들 수 있습니다 . 정적 그룹은 사용자를 추가하거나 제거하지 않는 한 변경되지 않습니다 .

정적 그룹 생성을 위한 지침

Administration Server 폼을 사용하여 새 정적 그룹을 만드는 경우 다음의 지침을 고려하십시오 .

- 정적 그룹에는 다른 정적 또는 동적 그룹이 포함될 수 있습니다 .
- 또한 선택적으로 새 그룹에 대한 설명을 추가할 수 있습니다 .
- 조직 단위가 디렉토리용으로 정의된 경우 Add New Group To 목록을 사용하여 신규 그룹을 추가할 위치를 지정할 수 있습니다 . 기본 위치는 디렉토리의 루트 지점 또는 최상위 항목입니다 .
- 원하는 정보를 모두 입력했으면 Create Group 을 눌러 그룹을 추가하고 바로 New Group 폼으로 되돌아갑니다 . 다른 방법으로 Create and Edit Group 을 눌러 그룹을 추가한 후 , 방금 추가한 그룹의 Edit Group 폼으로 계속합니다 . 그룹 편집에 대한 더 자세한 내용은 페이지 75 의 " 그룹 속성 편집 ," 을 참조하십시오 .

정적 그룹 생성

정적 그룹 항목을 만들려면 다음과 같이 합니다 .

1. Administration Server 에 액세스하고 Users & Groups 탭을 선택합니다 .
2. New Group 링크를 누릅니다 .
3. 필요한 정보를 입력하고 OK 를 누릅니다 .

자세한 내용은 온라인 도움말의 New Group 페이지를 참조하십시오 .

동적 그룹

동적 그룹에는 `groupOfURLs` 의 `objectclass` 가 부여되며 임의의 `memberURL` 속성이 포함됩니다 . 이 각각은 일련의 개체를 기술하는 LDAP URL 입니다 .

Sun ONE Web Server 에서는 그룹 사용자가 자동으로 임의의 속성에 기반하도록 하거나 일치하는 DN 이 있는 특정 그룹에 ACL 을 적용하려는 경우 동적 그룹을 만들 수 있습니다 . 예를 들어 `department=marketing` 속성이 있는 DN 이 자동으로 포함되도록 그룹을 만들 수 있습니다 . department=marketing 검색 필터를 적용하면 `department=marketing` 속성이 있는 모든 DN 을 포함하는 그룹이 검색됩니다 . 그 후 , 이 필터에 기반하여 검색 결과에서 동적 그룹을 정의할 수 있습니다 . 따라서 결과의 동적 그룹에 대한 ACL 을 정의할 수 있습니다 .

이 부분에서는 다음 항목에 대해 설명합니다 .

Sun ONE Web Server 의 동적 그룹 구현 원리

Sun ONE Web Server 는 `objectclass=groupOfURLs` 의 LDAP 서버 스키마에 동적 그룹을 구현합니다 . `groupOfURLs` 클래스에는 복수 `memberURL` 속성이 있을 수 있으며 , 이 각각은 디렉토리의 개체 세트를 나열하는 LDAP URL 로 구성됩니다 . 그룹의 구성원은 이 세트의 조합이 됩니다 . 예를 들어 다음 그룹은 오직 하나의 구성원 URL 만 포함합니다 .

```
ldap:///o=mcom.com??sub?(department=marketing)
```

이 예는 부서가 "marketing" 인 `o=mcom.com` 아래의 모든 개체로 구성되는 세트입니다 . LDAP URL 은 검색 기반 DN, 범위 및 필터 등을 포함하지만 호스트 이름과 포트는 포함하지 않습니다 . 따라서 동일한 LDAP 서버에 있는 개체만 참조할 수 있습니다 . 범위는 모두 지원됩니다 .

DN 은 자동으로 포함되므로 직접 개인을 그룹에 추가할 필요가 없습니다 . ACL 검증을 위하여 그룹 조회가 필요할 때마다 Sun ONE Web Server 가 LDAP 서버 검색을 수행하므로 그룹은 동적으로 변경됩니다 . ACL 파일에서 사용된 사용자 및 그룹 이름은 LDAP 데이터베이스에 있는 개체의 cn 속성에 대응됩니다 .

참고 Sun ONE Web Server 는 ACL 용 그룹 이름으로 `cn(commonName)` 속성을 사용합니다 .

ACL 에서 LDAP 데이터베이스로의 매핑은 `dbswitch.conf` 구성 파일 (실제 LDAP 데이터베이스 URL 로 ACL 데이터베이스 이름과 연결) 및 ACL 파일 (ACL 용으로 사용할 데이터베이스 정의) 모두에 정의됩니다 . 예를 들어 "staff" 라는 이름의 그룹의 구성원에게 기본 액세스 권한을 부여하려는 경우 ACL 코드는 개체 클래스가 `groupOf<anything>` 이며 CN 이 `"staff"` 로 설정된 개체를 조회합니다 . 개체는 구성원 DN 을 직접 나열 (정적 그룹용 `groupOfUniqueNames` 와 동일) 하거나 또는 LDAP URL 을 지정 (예 : `groupOfURLs`) 하여 그룹의 구성원을 정의합니다 .

정적 및 동적 그룹 가능

그룹 개체에는 `objectclass=groupOfUniqueMembers` 와 `objectclass=groupOfURL` 이 모두 있을 수 있으므로 `uniqueMember` 와 `memberURL` 속성이 모두 유효합니다 . 그룹의 구성원은 동적 및 정적 구성원의 조합입니다 .

서버 성능에 미치는 동적 그룹의 영향

동적 그룹을 사용하는 경우 서버 성능에 영향을 미칠 수 있습니다 . 그룹 구성원을 시험하며 DN 이 정적 그룹의 구성원이 아닌 경우 Sun ONE Web Server 는 데이터베이스의 baseDN 에 있는 모든 동적 그룹을 확인합니다 . Sun ONE Web Server 는 이 작업을 위하여 해당 baseDN 과 사용자의 DN 에 대한 범위를 확인하여 각 `memberURL` 이 일치하는지 확인하고 , 그런 후 , 사용자 DN 을 baseDN 으로 사용하고 `memberURL`의 필터를 사용하여 기본 검색을 수행합니다. 이 절차로 인하여 많은 수의 개별 검색이 누적될 수 있습니다 .

동적 그룹 생성을 위한 지침

Administration Server 폼을 사용하여 새 동적 그룹을 만드는 경우 다음의 지침을 고려하십시오 .

- 동적 그룹에는 다른 그룹이 포함될 수 없습니다 .
- 다음 형식으로 그룹의 LDAP URL 을 입력합니다 (호스트 및 포트 정보는 무시되므로 생략).

  ```
  ldap:///<basedn>?<attributes>?<scope>?<(filter)>
  ```

  필요한 매개 변수는 다음 표에 설명한 것과 같습니다 .

표 3-5) 동적 그룹 : 필요한 매개 변수

매개 변수 이름	설명
<base_dn>	검색 기반용 DN(Distinguished Name) 또는 LDAP 디렉토리에서 검색이 수행되는 지점 . 이 매개 변수는 때로 o=mcom.com 등의 디렉토리의 접미사 또는 루트로 설정됩니다 .
<attributes>	검색이 반환할 수 있는 속성 목록 . 하나 이상을 지정하려면 속성 사이를 쉼표로 분리 (예 : "cn,mail,telephoneNumber") 합니다 . 속성이 지정되지 않으면 모든 속성이 반환됩니다 . 참고로 동적 그룹 구성원 확인의 경우 이 매개 변수는 무시됩니다 .
<scope>	검색의 범위로 다음 중 한 가지 값을 가집니다 . • **base** 는 해당 URL 에 지정된 고유 이름 (<base_dn>) 에 대한 정보를 검색합니다 . • **one** 은 해당 URL 에 지정된 고유 이름 (<base_dn>) 보다 한 수준 아래의 항목에 대한 정보를 검색합니다 . 기본 항목은 이 범위에 포함되지 않습니다 . • **sub** 은 해당 URL 에 지정된 고유 이름 (<base_dn>) 아래의 모든 수준에 있는 항목에 대한 정보를 검색합니다 . 기본 항목은 이 범위에 포함되지 않습니다 . 이 매개변수는 필수입니다 .
<(filter)>	검색의 지정된 범위 안에 있는 항목에 적용되는 검색 필터 . Administration Server 폼을 사용하는 경우 반드시 이 속성을 지정해야 합니다 . 괄호는 반드시 필수입니다 . 이 매개변수는 필수입니다 .

참고로 <attributes>, <scope> 및 < (filter) > 매개 변수는 URL 에서의 위치에 따라 구분됩니다 . 속성을 지정하지 않으려는 경우에도 해당 필드에 물음표를 넣어 구분해야 합니다 .

- 또한 선택적으로 새 그룹에 대한 설명을 추가할 수 있습니다 .
- 조직 단위가 디렉토리용으로 정의된 경우 Add New Group To 목록을 사용하여 신규 그룹을 추가할 위치를 지정할 수 있습니다 . 기본 위치는 디렉토리의 루트 지점 또는 최상위 항목입니다 .

- 원하는 정보를 모두 입력했으면 Create Group 을 눌러 그룹을 추가하고 바로 New Group 폼으로 되돌아갑니다 . 다른 방법으로 Create and Edit Group 을 눌러 그룹을 추가한 후 , 방금 추가한 그룹의 Edit Group 폼으로 계속합니다 . 그룹 편집에 대한 더 자세한 내용은 " 그룹 속성 편집 " 페이지 75 을 참조하십시오 .

동적 그룹 생성

디렉토리에서 동적 그룹 항목을 만들려면 다음과 같이 합니다 .

1. Administration Server 에 액세스하고 Users & Groups 탭을 선택합니다 .
2. New Group 링크를 누릅니다 .
3. Type of Group 드롭 다운 목록에서 Dynamic Group 을 선택합니다 .
4. 필요한 정보를 입력하고 OK 를 누릅니다 .

자세한 내용은 온라인 도움말의 New Group 페이지를 참조하십시오 .

그룹 관리

Administration Server 의 Manage Group 폼에서 그룹을 편집하고 그룹 구성원을 관리할 수 있습니다 . 이 단원에서는 다음 항목에 대해 설명합니다 .

- 그룹 항목 찾기
- 그룹 속성 편집
- 그룹 구성원 추가
- 그룹 구성원 목록에 그룹 추가
- 그룹 구성원 목록에서 항목 제거
- 소유자 관리
- 추가 참조 관리
- 그룹 제거
- 그룹 이름 변경

그룹 항목 찾기

그룹 항목을 편집하려면 반드시 해당 항목을 찾아 표시해야 합니다 .

그룹 항목을 찾으려면 다음과 같이 합니다 .

1. Administration Server 에 액세스하고 Users & Groups 탭을 선택합니다 .
2. Manage Groups 링크를 누릅니다 .
3. Find Group 필드에 찾으려는 그룹 이름을 입력합니다 .

   검색 필드에 다음을 입력할 수 있습니다 .

   - 이름. 이름의 전체 또는 부분을 입력합니다. 검색 문자열과 일치되는 모든 항목이 검색됩니다 . 해당 항목을 찾을 수 없는 경우 검색 문자열을 포함하는 모든 항목이 검색됩니다 . 해당 항목을 찾을 수 없는 경우 검색 문자열과 소리가 유사한 모든 항목이 검색됩니다 .
   - 디렉토리에 있는 그룹을 모두 보려면 별표 (*) 를 사용합니다 . 필드에 아무런 값을 입력하지 않아도 동일한 효과를 얻을 수 있습니다 .
   - 임의 LDAP 검색 필터 . 등호 (=) 가 포함된 문자열은 검색 필터로 간주됩니다 .

   다른 방법으로 "Find all groups whose" 필드의 드롭 다운 메뉴를 사용하여 검색 범위를 좁힐 수 있습니다 .
4. Look within 필드에서 검색하려는 항목의 상위 조직 단위를 선택합니다 .

   기본값은 디렉토리의 루트 지점 또는 최상위 항목입니다 .
5. Format 필드에서 On-Screen 또는 Printer 를 선택합니다 .
6. Find 를 누릅니다 .

   검색 조건과 일치하는 그룹이 모두 표시됩니다 .
7. 결과 테이블에서 편집하려는 항목의 이름을 누릅니다 .

"Find all groups whose" 필드

"Find all groups whose" 필드를 사용하면 사용자 정의 검색 필터를 만들 수 있습니다 . 이 필드를 사용하면 Find Groups 필드에서 검색되는 결과를 더욱 정밀하게 할 수 있습니다 .

Look Within 디렉토리에 포함된 모든 그룹 항목을 표시하려면 별표 (*) 를 입력하거나 이 필드를 공란으로 남겨둡니다 .

사용자 정의 검색 필터를 만드는 방법에 대한 내용은 " 사용자 정의 검색 쿼리 작성 ," 를 (페이지 64) 참조하십시오 .

그룹 속성 편집

그룹 항목을 편집하려면 다음과 같이 합니다 .

1. Administration Server 에 액세스하고 Users & Groups 탭을 선택합니다 .
2. Manage Groups 링크를 누릅니다 .
3. 편집하려는 그룹을 찾은 후 원하는 변경 사항을 입력합니다 .

   특정 항목을 찾는 방법에 대한 자세한 내용은 " 그룹 항목 찾기 ," (페이지 73) 의 간단한 개념 설명을 참조하십시오 .

참고 운영 시스템의 Administration Server 사용자를 루트에서 다른 사용자로 변경하여 여러 사용자 (그룹에 속한 사용자) 가 구성 파일을 편집 / 관리하도록 할 수 있습니다. 그러나 UNIX/Linux 플랫폼의 경우 설치자는 그룹에게 구성 파일에 대한 "rw" 권한을 부여할 수 있는 반면 Windows 플랫폼의 경우 사용자는 반드시 "Administrators" 그룹에 속해야 합니다 .

그룹 속성 편집에 대한 자세한 내용은 온라인 도움말의 Manage Groups 페이지를 참조하십시오 .

참고 그룹 편집 폼을 편집하여 표시되지 않은 속성 값을 변경해야 하는 경우도 있습니다 . 이 경우 사용할 수 있으면 Directory Server `ldapmodify` 명령줄 유틸리티를 사용하십시오 .

그룹 구성원 추가

그룹에 구성원을 추가하려면 다음과 같이 합니다 .

1. Administration Server 에 액세스하고 Users & Groups 탭을 선택합니다 .
2. Manage Groups 링크를 누릅니다 .
3. " 그룹 항목 찾기 ," (페이지 73) 에 설명한 것과 같이 관리하려는 그룹을 찾은 후 Group Members 아래의 Edit 버튼을 누릅니다 .

   Sun ONE Web Server 에 항목을 검색할 수 있는 새 폼이 표시됩니다 . 목록에 사용자 항목을 추가하려면 Find 드롭 다운 목록에 Users 가 표시되어 있는지 확인합니다 . 그룹에 그룹 항목을 추가하려는 경우에는 Group 이 표시되어 있어야 합니다 .
4. 가장 오른쪽 텍스트 필드에 검색 문자열을 입력합니다. 다음 옵션을 입력합니다.

- 이름. 이름의 전체 또는 부분을 입력합니다. 검색 문자열과 일치되는 이름의 항목이 검색됩니다. 해당 항목을 찾을 수 없는 경우 검색 문자열을 포함하는 모든 항목이 검색됩니다 . 해당 항목을 찾을 수 없는 경우 검색 문자열과 소리가 유사한 모든 항목이 검색됩니다 .
- 사용자 항목을 검색하는 경우 사용자 ID.
- 전화번호 . 번호를 부분적으로 입력하면 검색 번호로 끝나는 전화번호를 포함하는 모든 항목이 검색됩니다 .
- 전자우편 주소 . @ 을 포함하는 모든 검색 문자열은 전자우편 주소인 것으로 가정합니다 . 정확한 일치가 검색되지 않는 경우 검색은 검색 문자열로 시작하는 모든 전자우편 주소를 찾습니다 .
- 현재 디렉토리에 있는 모든 항목이나 그룹을 보려면 이 텍스트 필드에 별표 (*) 를 입력하거나 빈 칸으로 남겨놓습니다 .
- 임의 LDAP 검색 필터. 등호(=)가 포함된 문자열은 검색 필터로 간주됩니다.

5. Find and Add 를 눌러 모든 일치 항목을 찾아 이를 그룹에 추가합니다 .

   검색 결과에 그룹에 추가하지 않으려는 항목이 포함된 경우 Remove from list? 열의 선택란을 선택합니다 . 또한 제거하려는 항목과 일치하는 검색 필터를 만든 후 Find and Remove 를 누르면 됩니다 .

6. 그룹 구성원 목록이 완료되었으면 Save Changes 를 누릅니다 .

   현재 표시된 항목이 그룹의 구성원이 됩니다 .

그룹 구성원 추가에 대한 자세한 내용은 온라인 도움말의 Edit Members 페이지를 참조하십시오 .

그룹 구성원 목록에 그룹 추가

그룹의 구성원 목록에 그룹 (개별 구성원 아님) 을 추가할 수 있습니다 . 이렇게 하면 포함된 그룹에 속한 사용자는 모두 대상 그룹의 구성원이 됩니다 . 예를 들어 Neil Armstrong 이 Engineering Managers 그룹의 구성원이며 Engineering Managers 그룹을 Engineering Personnel 그룹의 구성원으로 추가하면 Neil Armstrong 또한 Engineering Personnel 그룹의 구성원이 됩니다 .

그룹을 다른 그룹의 구성원 목록에 추가하려면 그룹이 사용자 항목인 것처럼 추가합니다 . 자세한 내용은 " 그룹 구성원 추가 ," (페이지 75) 를 참조하십시오 .

그룹 구성원 목록에서 항목 제거

그룹 구성원 목록에서 항목을 제거하려면 다음과 같이 합니다 .

1. Administration Server 에 액세스하고 Users & Groups 탭을 선택합니다 .
2. Manage Group 링크를 클릭하고 " 그룹 항목 찾기 ,"(페이지 73) 의 설명과 같이 관리하려는 그룹을 찾은 후 Group Members 아래의 Edit 버튼을 누릅니다 .
3. 목록에서 제거하려는 각 구성원에 대하여 Remove from list? 열 아래의 해당 선택란을 선택합니다 .

   다른 방법으로 제거하려는 항목을 검색하는 필터를 만든 후 Find and Remove 버튼을 누릅니다 . 검색 필터를 만드는 방법에 대한 자세한 내용은 " 그룹 구성원 추가 ," (페이지 75) 를 참조하십시오 .
4. Save Changes 를 누릅니다 . 그룹 구성원 목록에서 해당 항목이 삭제됩니다 .

소유자 관리

그룹 구성원 목록을 관리하는 것과 마찬가지로 그룹의 소유자를 관리할 수 있습니다. 더 자세한 내용은 아래의 표에 명시한 부분을 참조하십시오 .

표 3-6) 추가 정보

원하는 작업	해당 부분
그룹에 소유자 추가	" 그룹 구성원 추가 ,"(페이지 75)
소유자 목록에 그룹 추가	" 그룹 구성원 목록에 그룹 추가 ,"(페이지 76)
소유자 목록에서 항목 제거	" 그룹 구성원 목록에서 항목 제거 ,"(페이지 77)

추가 참조 관리

" 추가 참조 " 는 현재 그룹과 관련될 수 있는 다른 디렉토리 항목을 참조합니다 . 여기에서 현재 그룹과 관련된 사용자 및 기타 그룹의 항목을 쉽게 찾을 수 있습니다 .

그룹 구성원 목록을 관리하는 것과 마찬가지로 추가 참조를 관리할 수 있습니다 . 더 자세한 내용은 아래의 표에 명시한 부분을 참조하십시오 .

표 3-7) 추가 정보

원하는 작업	해당 부분
추가 참조에 사용자 추가	" 그룹 구성원 추가 ,"(페이지 75)
추가 참조에 그룹 추가	" 그룹 구성원 목록에 그룹 추가 ,"(페이지 76)
추가 참조에서 항목 제거	" 그룹 구성원 목록에서 항목 제거 ,"(페이지 77)

그룹 제거

그룹을 삭제하려면 다음과 같이 합니다 .

1. Administration Server 에 액세스하고 Users & Groups 탭을 선택합니다 .
2. Manage Group 링크를 클릭하고 "그룹 항목 찾기," (페이지 73)의 설명과 같이 관리하려는 그룹을 찾은 후 Delete Group 버튼을 누릅니다 .

참고 Administration Server 는 제거하는 그룹의 개별 구성원을 제거하지 않으며 단지 제거되는 그룹 항목만 삭제합니다 .

그룹 이름 변경

그룹의 이름을 변경하려면 다음과 같이 합니다 .

1. Administration Server 에 액세스하고 Users & Groups 탭을 선택합니다 .
2. Manage Group 링크를 클릭하고 "그룹 항목 찾기," (페이지 73)의 설명과 같이 관리하려는 그룹을 찾습니다 .
3. Rename Group 버튼을 누르고 표시되는 대화 상자에 새 그룹 이름을 입력합니다.

그룹 항목의 이름을 변경하는 경우 오직 그룹의 이름만 변경하게 되며 , Rename Group 기능을 사용하여 항목을 하나의 조직 단위에서 다른 단위로 옮길 수는 없습니다 . 예를 들어 회사에 다음과 같은 조직이 있는 것으로 가정합니다 .

- Marketing 및 Product Management 를 위한 조직 단위
- Marketing 조직 단위 아래의 Online Sales 그룹

이 예에서 그룹의 이름을 Online Sales 에서 Internet Investments 로 변경할 수 있으나 Marketing 조직 단위 아래에 있는 Online Sales 가 Product Management 조직 단위 아래의 Online Sales 로 되도록 항목의 이름을 변경할 수는 없습니다 .

조직 단위 생성

조직 단위에는 그룹 구성원이 포함될 수 있으며 보통 사업 단위 , 부서 또는 기타 명확히 구분되는 사업 그룹을 나타냅니다 . DN 은 하나 이상의 조직 단위에 존재할 수 있습니다 .

조직 단위를 만들려면 다음과 같이 합니다 .

1. Administration Server 에 액세스하고 Users & Groups 탭을 선택합니다 .
2. New Organizational Unit 링크를 누르고 필요한 정보를 입력합니다 .

자세한 내용은 온라인 도움말의 New Group 페이지를 참조하십시오 .

디렉토리 관리자는 다음의 참고에 유의할 필요가 있습니다 .

- 새 조직 단위는 `organizationalUnit` 개체 클래스를 사용하여 만들어집니다 .
- 새 조직 단위용 고유 이름의 형식은 다음과 같습니다 .

```
ou=new organization, ou=parent organization, ...,o=base
organization, c=country
```

예를 들어 Accounting 이라는 이름의 새 조직을 West Coast 라는 이름의 조직 단위 내에 만들며 Base DN 이 o=Ace Industry, c=US 인 경우 , 새 조직 단위의 DN 은 다음과 같습니다 .

```
ou=Accounting, ou=West Coast, o=Ace Industry, c=US
```

조직 단위 관리

Organizational Unit Edit 폼에서 조직 단위를 편집하고 관리할 수 있습니다 . 이 단원에서는 다음 작업에 대해 설명합니다 .

- 조직 단위 찾기
- 조직 단위 속성 편집
- 조직 단위 이름 변경
- 조직 단위 삭제

조직 단위 찾기

조직 단위를 찾으려면 다음과 같이 합니다 .

1. Administration Server 에 액세스하고 Users & Groups 탭을 선택합니다 .
2. Manage Organizational Units 링크를 누릅니다 .
3. Find organizational unit 필드에 찾으려는 단위의 이름을 입력합니다 . 검색 필드에 다음의 항목을 입력할 수 있습니다 .
   - 이름. 이름의 전체 또는 부분을 입력합니다. 검색 문자열과 일치되는 모든 항목이 검색됩니다 . 해당 항목을 찾을 수 없는 경우 검색 문자열을 포함하는 모든 항목이 검색됩니다 . 해당 항목을 찾을 수 없는 경우 검색 문자열과 소리가 유사한 모든 항목이 검색됩니다 .
   - 디렉토리에 있는 그룹을 모두 보려면 별표 (*) 를 사용합니다 . 필드에 아무런 값을 입력하지 않아도 동일한 결과를 얻을 수 있습니다 .
   - 임의 LDAP 검색 필터 . 등호 (=) 가 포함된 문자열은 검색 필터로 간주됩니다 .

   다른 방법으로 Find all units whose 필드의 드롭 다운 메뉴를 사용하여 검색 범위를 좁힐 수 있습니다 .
4. Look within 필드에서 검색하려는 항목의 상위 조직 단위를 선택합니다 .

   기본값은 디렉토리의 루트 지점입니다 .
5. Format 필드에서 On-Screen 또는 Printer 를 선택합니다 .
6. Find 를 누릅니다 .

   검색 조건과 일치하는 조직 단위가 모두 표시됩니다 .
7. 결과 표에서 찾으려는 조직 단위의 이름을 누릅니다 .

"Find all units whose" 필드

Find all units whose 필드를 사용하면 사용자 정의 검색 필터를 만들 수 있습니다 . 이 필드를 사용하면 Find Organizational Unit 필드에서 검색되는 결과를 더욱 정밀하게 할 수 있습니다 .

Look Within 디렉토리에 포함된 모든 그룹 항목을 표시하려면 별표 (*) 를 입력하거나 이 필드를 공란으로 남겨둡니다 .

사용자 정의 검색 필터를 만드는 방법에 대한 내용은 " 사용자 정의 검색 쿼리 작성 ," 를 (페이지 64) 참조하십시오 .

조직 단위 속성 편집

조직 단위 항목을 변경하려면 Administration Server 에 액세스하고 다음과 같이 합니다 .

1. "조직 단위 찾기,"(페이지 80)에 설명한 대로 편집하려는 조직 단위를 찾습니다.

   조직 단위 편집 폼이 표시됩니다 .

2. 표시된 필드를 원하는 대로 변경하고 Save Changes 를 누릅니다 .

   변경 사항은 즉시 적용됩니다 .

참고 조직 단위 편집 폼을 편집하여 표시되지 않은 속성 값을 변경해야 하는 경우도 있습니다 . 이 경우 사용할 수 있으면 Directory Server `ldapmodify` 명령줄 유틸리티를 사용하십시오 .

조직 단위 이름 변경

조직 단위 항목의 이름을 변경하려면 Administration Server 에 액세스하고 다음과 같이 합니다 .

1. 이름을 변경하려는 조직 아래의 디렉토리에 항목이 있으면 안 됩니다 .
2. "조직 단위 찾기,"(페이지 80)에 설명한 대로 편집하려는 조직 단위를 찾습니다.
3. Rename 버튼을 누릅니다 .
4. 표시되는 대화 상자에 새 조직 단위 이름을 입력합니다 .

참고 조직 단위 항목의 이름을 변경하는 경우 오직 조직 단위의 이름만 변경하게 되며 , 이름 변경 기능을 사용하여 항목을 하나의 조직 단위에서 다른 단위로 옮길 수는 없습니다 . 자세한 내용은 " 조직 단위 이름 변경 ," (페이지 81) 를 참조하십시오 .

조직 단위 삭제

조직 단위 항목을 삭제하려면 Administration Server 에 액세스하고 다음과 같이 합니다 .

1. 이름을 변경하려는 조직 아래의 디렉토리에 항목이 있으면 안 됩니다 .
2. "조직 단위 찾기,"(페이지 80)에 설명한 대로 삭제하려는 조직 단위를 찾습니다.

3. Delete 버튼을 누릅니다 .
4. 표시되는 확인 대화 상자에서 OK 를 누릅니다 .
   조직 단위는 즉시 삭제됩니다 .

4 장

웹 컨테이너 및 웹 응용 프로그램용 J2EE 기반 보안

이 장에서는 Sun ONE Web Server 6.1 웹 컨테이너 및 웹 응용 프로그램용 J2EE 기반 보안의 기본 기능에 대하여 설명합니다 . 우선 Web Server 가 지원하는 두 가지 기본 인증 및 권한 모델인 ACL(Access Control List) 기반 보안 모델과 J2EE/ 서브릿 기반 보안 모델에 대하여 설명합니다 . 또한 Sun ONE Web Server 6.1 의 새로운 기능에 대하여 설명합니다 . 이 기능을 사용하면 두 보안 시스템의 이익을 활용할 수 있는 Java 응용 프로그램을 구현할 수 있습니다 .

이 장의 나머지 부분에서 J2EE/ 서브릿 구성 문제에 대하여 설명하며 , 또한 관련 보안 사항은 다음 장에서 설명합니다 .

- 인증서 및 공용 키 암호화 : 제 6 장 , " 인증서 및 키 사용 ".
- ACL 기반 보안 : 제 9 장 , " 서버 액세스 제어 ".

이 장의 내용 :

Sun ONE Web Server 보안 설명

인증 , 권한 , 액세스 제어 등 다양한 보안 서비스 및 메커니즘을 통하여 Web Server 에 상주하는 리소스를 보호할 수 있습니다 .

인증은 신분을 확인하는 과정입니다 . 권한은 개인 또는 단체에게 제한된 리소스에 대한 액세스를 허용하는 것으로 액세스 제어 메커니즘이 이러한 제한을 집행합니다 . 인증과 권한은 다양한 보안 모델과 서비스로 집행할 수 있습니다 .

Sun ONE Web Server 6.1 은 HTTP 엔진이 제공하는 ACL 기반 보안 모델과 웹 컨테이너가 제공하는 J2EE Servlet 버전 2.3 표준 등 두 가지 보안 모델을 지원합니다 .

이 두 모델은 Sun ONE Web Server 6.1 프로세스와 같은 시간 동안 지속됩니다 . 각 모델은 클라이언트 인증 및 권한 보안 서비스를 모두 지원합니다 .

Sun ONE Web Server 웹 컨테이너는 JAAS(Java Authentication and Authorization Service) 기반 영역 메커니즘을 통하여 클라이언트 인증을 제공하며 J2EE 역할 기반 메커니즘을 통하여 권한을 제공합니다 . 원시 영역은 Sun ONE Web Server 6.1 이 제공하는 영영 중 하나입니다 . 이 영역은 두 가지 보안 모델을 잇는 다리 역할을 합니다 .

Sun ONE Web Server 6.1 은 선언적 보안과 프로그램적 보안을 모두 지원합니다 .

Sun ONE Web Server 6.1 은 J2EE 플랫폼의 기능을 활용하여 응용 프로그램 구성요소를 개발 및 어셈블링한 존재와 운영 환경에서 응용 프로그램을 구성한 존재 사이의 선언적 계약을 정의합니다 . 응용 프로그램 보안의 맥락에서 응용 프로그램 제공업체는 보안 요구 사항이 응용 프로그램 구성시 만족될 수 있는 방식으로 응용 프로그램의 보안 요구 사항을 선언해야 합니다 . 응용 프로그램에서 사용되는 보안적 보안 메커니즘은 *구현 기술자 (deployment descriptor)* 라는 문서에 선언적 구문으로 명시됩니다 . 이에 따라 응용 프로그램 구현자는 컨테이너 특정 도구를 채택하여 구현 기술자에 있는 응용 프로그램 요구 사항을 J2EE 컨테이너에 의하여 구현된 보안 메커니즘으로 매핑합니다 . Sun ONE Web Server 6.1 의 웹 응용 프로그램용 구현 기술자 파일은 `web.xml` 및 `sun-web.xml` 파일에 있습니다 .

프로그램적 보안은 보안 인식 응용 프로그램이 만든 보안 결정을 말합니다 . 프로그램적 보안은 응용 프로그램의 보안 모델을 명시하는 데 선언적 보안만으로는 부족한 경우에 유용합니다 . 예를 들어 응용 프로그램의 권한 결정은 하루의 시간 , 호출의 매개 변수 또는 웹 구성요소의 내부 상태 등에 따라 결정될 수 있습니다 . 다른 응용 프로그램의 경우에는 데이터베이스에 저장된 사용자 정보에 따라 액세스를 제한할 수 있습니다 .

이 장의 나머지에서는 Sun ONE Web Server 6.1 에서 지원되는 인증 및 권한의 주요 개념에 대하여 살펴봅니다 .

- AC 기반 액세스 제어 : ACL 기반 액세스 컨트롤 개요 .
- J2EE 기반 액세스 제어 : J2EE/ 서브릿 기반 액세스 컨트롤 개요 .
- 기본 영역 지원 : 원시 영역 .
- 프로그램적 보안 : 프로그램적 보안 사용 .

ACL 기반 액세스 컨트롤 개요

ACL 기반 액세스 제어는 제 9 장 , " 서버 액세스 제어 " 에서 자세히 설명합니다 . 다음에서는 주요 개념의 개요를 간단히 설명합니다 .

Sun ONE Web Server 6.1 은 로컬에 저장된 ACL(access control list) 을 사용하여 인증 및 권한을 지원합니다 . 이 ACL 에는 리소스에 대하여 사용자에게 부여할 액세스 권한이 기술되어 있습니다 . 예를 들어 , ACL 에 있는 항목에 따라 John 이라는 사용자에게 특정 폴더 `mics` 에 대한 읽기 권한을 부여할 수 있습니다 .

```
acl "path=/export/user/990628.1/docs/misc/";
  authenticate (user,group) {
     database = "default";
     method = "basic";
  };
  deny (all) (user=”anyone”;
  allow (read) (user = "John");
```

Sun ONE Web Server 의 핵심 ACL 은 기본 , SSL 및 다이제스트의 세가지 인증 유형을 지원합니다 .

기본 인증은 보통 텍스트로 전달된 사용자 이름 및 암호 목록에 따라 달라집니다 . SSL 방법을 사용하려면 브라우저에 사용자 인증서가 있어야 하며 , 여기에는 사용자의 공용키와 이름 , 전자우편 등의 기타 사용자 정보가 있습니다 . 다이제스트 인증은 암호화 기법을 사용하여 사용자의 신분 증명을 암호화합니다 .

ACL 기반 액세스 모델의 기본 기능은 아래와 같습니다 .

- ACL 기반 인증 및 권한은 다음 구성 파일을 사용합니다 .
  - *server-install*`/httpacl/*.acl` 파일
  - *server-install*`/userdb/dbswitch.conf`

- *server-install*/*server-instance*/config/server.xml

- 인증서 데이터베이스는 auth-db 모듈에 의하여 제공되며 , 이 모듈은 dbswitch.conf 파일에서 구성됩니다 .
- 인증 및 권한 부여는 ACL 이 구성된 경우 *server-install*/httpacl/*.acl 에 설정된 액세스 제어 규칙에 따라 수행됩니다 . 적용되는 인증 규칙은 해당 요청을 처리하는 (server.xml 의 적절한 VS 항목의 구성 대로) 가상 서버에 해당하는 ACL 파일에 정의됩니다 . Sun ONE Web Server 6.1 *Administrator's Configuration File Reference* 의 ACLFILE 요소와 VS 요소의 aclids 등록정보를 참조하십시오 . 보통 이들 파일은 /httpacl/ 디렉토리에 있으나 server.xml 구성을 변경하는 경우에는 다른 위치에 있을 수도 있습니다 .

또한 , Sun ONE Web Server 6.1 SSL 엔진은 외부 암호 하드웨어를 지원하여 SLL 처리 부하를 줄이고 최적의 조작 방지 키 저장소를 제공합니다 .

액세스 제어와 외부 암호화 하드웨어에 대한 자세한 내용은 제 9 장 , " 서버 액세스 제어 " 를 참조하십시오 .

J2EE/ 서브릿 기반 액세스 컨트롤 개요

J2EE/ 서브릿 기반 액세스 제어는 Sun ONE Web Server 6.1 *Programmer's Guide to Web Applications* 에서 자세히 설명합니다 . 다음에서는 주요 개념의 개요를 간단히 설명합니다 .

Sun ONE Web Server 6.1 은 ACL 기반 인증을 제공하는 것 외에 J2EE 1.3 표준에 정의된 보안 모델을 활용하여 안전한 Java 웹 응용 프로그램을 개발하고 배치하는데 도움이 되는 여러 기능을 제공합니다 .

보통 J2EE 기반 웹 응용 프로그램은 다음으로 구성되며 이들의 일부분 또는 전체에 대한 액세스를 제한할 수 있습니다 .

- 서브릿
- JSP(JavaServer Pages) 구성요소
- HTML 문서
- 이미지 파일이나 압축 보관 등의 기타 리소스

J2EE/ 서브릿 기반 액세스 제어 인프라에는 보안 영역이 사용되어야 합니다 . 사용자가 웹 서버를 통하여 응용 프로그램의 액세스 보호 부분에 액세스하려는 경우 웹 컨테이너는 사용자에게 신분 정보를 요구합니다 . 그 후 , 이 특정 응용 프로그램용 보안 서비스에서 현재 사용 중인 영역에 대한 검증으로 정보를 전달합니다 .

J2EE/ 서브릿 기반 액세스 모델의 기본 기능은 아래와 같습니다 .

- J2EE/ 서브릿 기반 인증은 다음 구성 파일을 사용합니다 .
  - 웹 응용 프로그램 개발 기술 파일 `web.xml` 및 `sun-web.xml`
  - *server-install*`/`*server-instance*`/config/server.xml`
- 인증은 `xerver.xml` 파일의 `AUTHREALM` 항목을 통하여 구성된 Java 보안 영역에 의하여 수행됩니다 .
- 액세스 제어 규칙이 개발 기술 파일 `web.xml` 에 설정된 경우에는 이 규칙에 따라 인증이 수행됩니다 .

다음은 보안 영역의 개념에 대하여 간단히 설명합니다 . J2EE 보안 모델과 영역 기반 인증에 대한 더 자세한 내용은 *Sun ONE Web Server 6.1 Programmer's Guide to Web Applications* 를 참조하십시오 .

영역 기반 보안

J2EE 기반 보안 모델은 사용자를 확인하고 인증하는 보안 영역을 제공합니다 . 사용자 정보는 배후의 보안 영역에서 가져옵니다 . 영역 기반 보안은 두 가지 측면으로 구성됩니다 .

- 영역 기반 사용자 인증 . 배후의 영역을 통하여 사용자를 검증합니다 .
- 역할 기반 인증 . 사용자에게 역할을 지정하며 , 따라서 리소스에 대한 액세스를 부여 또는 제한합니다 .

영역 기반 사용자 인증

인증 과정은 보안 도메인이라고 하는 배후의 영역을 통하여 사용자를 검증합니다 . 영역은 일련의 사용자 , 그룹 매핑 (선택) 및 인증 요청을 검증할 수 있는 인증 로직으로 구성됩니다 . 구성된 영역에 의하여 인증 요청이 검증되고 보안 컨텍스트가 설정되면 이는 `urn-as` 조건에 의하여 재설정되지 않는 한 모든 후속 인증 결정에 적용됩니다 .

서버 인스턴스의 구성된 영역 수에는 제한이 없습니다 . 구성 정보는 `server.xml` 파일의 `AUTHREALM` 요소에 있습니다 .

Sun ONE Web Server 에서 인증 서비스는 JAAS 에 구축되며 , 이는 플러그 인 가능한 보안 도메인을 제공합니다 . Sun ONE Web Server 6.1 의 Java 인증 영역은 Sun ONE Application 7.0 영역과 호환됩니다 .

Sun ONE Web Server 6.1 이 제공하는 영역 :

- LDAP 영역
- 파일 영역
- Solaris 영역
- 인증서 영역
- 사용자 정의 영역
- 원시 영역

LDAP 영역

`ldap` 영역을 사용하여 사용자 보안 정보용으로 LDAP 데이터베이스를 사용할 수 있습니다 . LDAP 디렉토리 서비스는 고유한 ID 를 가진 속성의 컬렉션입니다 . `ldap` 영역은 프로덕션 시스템을 구현하는데 이상적입니다 .

`ldap` 영역에 대하여 사용자를 인증하려면 반드시 LDAP 디렉토리에 원하는 사용자를 만들어야 합니다 . 이 작업은 Administration Server 의 Users & Group 탭이나 LDAP 디렉토리 제품의 관리 콘솔에서 할 수 있습니다 . 더 자세한 내용은 "LDAP 기반 인증 데이터베이스에 신규 사용자 생성 " 페이지 58 를 참조하십시오 .

파일 영역

파일영역은 Sun ONE Web Server 를 처음 설치할 때 기본으로 설정되는 영역입니다 . 이 영역은 설정이 쉽고 간단하여 개발자에게 매우 편리합니다 .

파일 영역은 텍스트 파일에 저장된 사용자 데이터를 기준으로 사용자를 인증합니다 . 파일 영역은 다음의 인증 데이터베이스를 지원합니다 .

- keyfile 스타일 데이터베이스
- htaccess 스타일 데이터베이스
- digest 스타일 데이터베이스

다양한 파일 기반 인증 데이터베이스에 대한 자세한 내용은 <add> 를 참조하십시오 .

파일 영역에서 사용되는 사용자 정보는 비어있는 상태로 시작하므로 파일 영역을 사용하기 전에 반드시 사용자를 추가해야 합니다 . 이 작업에 대한 자세한 설명은 " 파일 기반 인증 데이터베이스에 신규 사용자 생성 " 페이지 61 를 참조하십시오 .

Solaris 영역

`solaris` 영역을 사용하면 Solaris 사용자 이름 + 암호 데이터를 사용하여 인증할 수 있습니다 . 이 영역은 Solaris 9 에서만 지원됩니다 . 이 영역은 Solaris 9 Operating Environment 에 있는 사용자 데이터를 사용하므로 별도 데이터베이스를 설정하는 추가 작업을 하지 않아도 됩니다 .

인증서 영역

인증서 영역은 SSL 인증을 지원합니다 . 인증서 영역은 Sun ONE Web Server 의 보안 컨텍스트에 사용자 ID 를 설정하며 클라이언트 인증서에 있는 사용자 데이터를 입력합니다 . 그런 후 , J2EE 컨테이너는 각 사용자의 인증서에 있는 사용자 DN 을 기준으로 인증 프로세스를 처리합니다 . 인증서 영역은 X.509 인증서를 통한 SSL 또는 TLS 클라이언트 인증으로 사용자를 인증합니다 .

서버 및 클라이언트 인증서 설정에 대한 자세한 내용은 제 6 장 , " 인증서 및 키 사용 " 을 참조하십시오 .

사용자 정의 영역

Oracle 등의 기타 데이터베이스용 영역을 구축하면 플러그 인 가능한 JAAS 로그인 모듈과 영역 구현을 사용하여 필요에 맞는 영역을 만들 수 있습니다 . 참고로 클라이언트측 JAAS 로그인 모듈은 Sun ONE Web Server 에서 사용하는 데 적합하지 않습니다 .

Sun ONE web Server 6.1 의 예제 영역을 템플릿으로 참조하십시오 .

원시 영역

원시 영역은 특별 영역으로 핵심 ACL 기반 인증 모델과 J2EE/ 서브릿 인증 모델 사이의 다리 역할을 합니다 . Java 웹 응용 프로그램용 원시 영역을 사용하여 ACL 하위 시스템이 인증을 수행할 수 있도록 할뿐 아니라 Java 웹 응용 프로그램에서 이를 사용할 수 있도록 합니다 .

인증 작업이 시작되면 원시 영역이 이 인증을 해당 코어 인증 하위 시스템으로 넘깁니다 . 사용자가 볼 때 이는 예를 들어 LDAP 영역이 인증을 구성된 LDAP 로 넘기는 것과 같습니다 . 원시 영역이 그룹 구성원 조회를 처리하는 경우 또한 코어 인증 하위 시스템으로 이 작업을 넘깁니다 . Java 웹 모듈 및 개발자의 관점에서 원시 영역은 웹 모듈과 함께 사용할 수 있는 기타 Java 영역과 다르지 않습니다 .

원시 영역이 인증을 코어에 위임하므로 몇 가지 추가 구성이 필요합니다 . 더 자세한 내용은 원시 영역 구성를 참조하십시오 .

Sun ONE Web Server 6.1 *Programmer's Guide to Web Applications* 에서 J2EE 보안 영역 및 보안 영역 구성에 사용할 수 있는 구성 매개 변수에 대하여 자세히 설명합니다 .

역할 기반 인증

Java Servlet 2.3 표준은 다양한 J2EE 응용 프로그램 리소스에 대한 액세스를 제한하는 액세스 제어 규칙 설정 방법을 정의합니다 .

역할을 제한된 영역으로 매핑

J2EE 액세스 제어는 역할을 기준으로 합니다 . 특정 HTML 페이지 , 서브릿 , JSP 등에 대한 액세스를 제한하려면 반드시 다음을 정의해야 합니다 .

- 웹 모듈 기술자 (`web.xml`) 의 목록에 따른 제한 영역
- 각 제한 영역에 대한 액세스를 허용할 역할 (`web.xml`)
- 사용자 및 그룹을 역할에 매핑하여 특정 사용자가 제한된 영역에 액세스할 수 있는지의 여부를 결정 (`sun-web.xml`)

사용자는 여러 가지 역할을 가질 수 있으며 사용자에게 최소한 하나의 역할이 지정되어 있음이 확인되면 해당 영역에 대한 액세스가 허용됩니다 .

`webapps/security` 디렉토리에 있는 예제에는 Sun ONE Web Server 6.1 의 다양한 액세스 제한이 템플릿으로 구성되어 있습니다 . Servlet 역할 기반 보안에 대한 자세한 내용은 Servelet 2.3 표준을 참조하십시오 .

역할별 액세스 제어 정의

J2EE 응용 프로그램 역할은 요약으로 특정 응용 프로그램에 적용됩니다 . 응용 프로그램을 권한 있는 사용자에게만 액세스가 제한된 실제 환경에서 실행하려면 반드시 사용자 이름을 `sun-web.xml` 기술자에 있는 역할과 매핑해야 합니다 . 다음과 같은 방법을 사용합니다 .

책임 매핑 - 사용자 이름을 또는 여러 개의 이름을 `sun-web.xml` 의 역할로 직접 매핑합니다 . 이 방법은 시험용으로 편리하지만 각 역할의 사용자 수 제한을 넘어 확장될 수 없습니다 .

그룹 매핑 - 사용자 이름을 또는 여러 개의 이름을 하나 이상의 그룹을 통하여 `sun-web.xml` 의 역할로 직접 매핑합니다 . (예를 들어 그룹 이름은 engineers, managers, staff 등이 될 수 있습니다 .) 그룹에 속한 인증된 사용자는 응용 프로그램 역할에 지정됩니다 . 사용 중인 영역 구현 (또는 참고인 데이터베이스) 는 해당 그룹에 속한 사용자를 결정해야 한다는 점에 유의하십시오 .

책임자 (사용자) 가 서브릿이나 JSP 등의 특정 웹 리소스를 요청하면 웹 컨테이너는 구현 기술자 파일에서 리소스에 연결된 보안 제한 또는 권한을 확인하여 책임자에게 액세스 권한이 있는지 결정합니다 .

역할 매핑 항목은 모듈 기술자에 있는 사용자 또는 그룹으로 역할을 매핑합니다 . 예 :

```
<sun-web-app>
<security-role-mapping>
<role-name>manager</role-name>
<principal-name>jsmith</principal-name>
<group-name>divmanagers</group-name>
</security-role-mapping>
</sun-web-app>
```

구현 기술자 파일에 대한 자세한 내용은 Sun ONE Web Server 6.1 *Programmer's Guide to Web Applications* 를 참조하십시오 .

영역 구성 방법

다음 중 한가지 방법으로 영역을 구성할 수 있습니다 .

- 관리 인터페이스 사용
- server.xml 파일 편집

관리 인터페이스 사용

Administration 인터페이스를 사용하여 영역을 구성하려면 다음과 같이 합니다 .

1. Administration Server 인터페이스에서 관리하려는 서버 인스턴스에 액세스한 후 Java 탭을 누릅니다 .
2. Security Realms 링크를 누릅니다 .

   기본적으로 다음의 영역이 제공됩니다 .

   - file
   - native
   - ldap
3. 영역을 추가하려면 New 버튼을 누릅니다 . 영역을 삭제하려면 영역 옆의 선택란을 선택하고 확인을 누릅니다 . 영역을 편집하려면 영역 이름을 누릅니다 .
4. 영역을 추가하거나 편집하는 경우 영역의 이름 , 클래스 이름 , 등록 정보 및 사용자를 입력 (파일 영역만 해당) 한 후 확인 버튼을 누릅니다 .
5. OK 를 누릅니다 .

server.xml 파일 편집

기본 영역은 배경에서 `server.xml` 파일의 `SECURITY` 요소에 설정됩니다 . `SECURITY` 구성은 다음과 같습니다 .

```
<SECURITY defaultrealm="file" anonymousrole="ANYONE"
      audit="false">
   <AUTHREALM name="file"
         classname="com.iplanet.ias.security.auth.realm.file.FileRe
alm">
      <property name="file" value="instance_dir/config/keyfile"/>
      <property name="jaas-context" value="fileRealm"/>
   </AUTHREALM>
   ...
</SECURITY>
```

`defaultrealm` 속성은 서버가 기본으로 사용하는 영역을 가리킵니다 . 기본 영역은 자체의 `web.xml` 에서 유효한 영역을 제공하지 않는 모든 웹 응용 프로그램에 의하여 사용됩니다 . 반드시 `AUTHREALM` 이름으로 구성된 영역 중 하나를 가리켜야 합니다 . 기본은 파일 영역입니다 .

`audit` 플래그에 따라 감사 정보를 기록할 것인지 결정됩니다. 플래그가 `true`로 설정되면 서버가 모든 인증 및 권한 이벤트에 대한 감사 메시지를 기록합니다 .

영역 구성을 변경하는 경우 반드시 서버를 다시 시작하여 변경 사항이 적용되도록 합니다 .

server.xml 파일에 대한 자세한 내용은 Sun ONE Web Server 6.1 *Administrator's Configuration File Reference* 를 참조하십시오 .

원시 영역 구성

다른 영역과 마찬가지로 server.xml 의 SECURITY 요소에 있는 AUTHREALM 요소를 사용하여 원시 영역을 구성할 수 있습니다 . 예 :

```
<AUTHREALM name="native"
classname="com.sun.enterprise.security.auth.realm.webcore.NativeRea
lm">
   <PROPERTY name="auth-db" value="mykeyfile" />
   <PROPERTY name="jaas-context" value="nativeRealm"/>
</AUTHREALM>
```

auth-db 등록 정보는 원시 영역이 모든 인증 요청을 위임할 코어 응용 프로그램 데이터베이스를 가리키며 , 이 예의 경우 "mykeyfile" 이라는 이름의 인증 데이터베이스입니다 . 이 등록 정보는 선택 사항입니다 . 지정되지 않는 경우 코어 인증 엔진은 기본 auth-db 를 사용하여 원시 영역에서 수신되는 모든 요청을 처리합니다 . 대부분의 영역과 마찬가지로 jaas-context 등록 정보는 사용되는 JAAS 로그인 컨텍스트 (login.conf 에 정의) 를 가리킵니다 .

원시 영역에는 다른 구성이 필요하지 않습니다 . 그러나 요청이 코어 인증 데이터베이스로 위임되므로 특정 인증 데이터베이스 또한 반드시 적절하게 구성되어야 합니다 . 이 부분의 나머지에서는 코어 인증 데이터베이스의 구성 예를 살펴봅니다 .

코어 (Native) 인증 데이터베이스를 구성하려면 server.xml 의 VS 요소에 반드시 auth-db 이름을 데이터베이스 이름으로 매핑하는 USERDB 요소가 포함되어야 합니다 . 예 :

```
<VS id="https-plaza.com"  ....
....
   <USERDB id="mykeyfile" database="myalt"/>
....
</VS>
```

참고로 auth-db 등록정보가 지정되지 않은 경우 (이 경우 "default" 를 사용) USERDB 항목의 id="default" 를 임의의 데이터베이스 이름으로 매핑할 수 있습니다 . 매핑이 없는 경우에는 default 로 매핑됩니다 .

다음 , *install-root*/userdb/dbswitch.conf 파일에 반드시 myalt 데이터베이스용 구성이 있어야 합니다 . 다음 예에서는 myalt 를 파일 기반 인증 데이터베이스로 정의합니다 .

```
directory myalt file

myalt:syntax keyfile

myalt:keyfile /local/ws61/https-plaza.com/config/keyfile
```

위의 구성은 원시 영역에만 국한되지 않습니다 . 원시 영역은 유효한 인증 디렉토리 구성을 대상 인증 데이터베이스로 사용할 수 있습니다 . 따라서 원시 영역은 원시 LDAP 인증 데이터베이스 또는 심지어 사용자 정의 원시 인증 데이터베이스로 위임될 수 있습니다 .

참고 Sun ONE Web Server 6.1 의 경우 웹 응용 프로그램에는 LDAP 를 인증 엔진으로 사용하기 위한 두 가지 서로 다른 메커니즘이 있습니다 .

- Java LDAP 영역 사용
- 원시 LDAP 인증 데이터베이스로 위임되도록 구성된 Java 원시 영역 사용 .

기본 영역 지정

기본 영역은 자체의 web.xml 구현 기술자 파일에 유효한 대체 영역이 지정되지 않은 모든 웹 응용 프로그램용 인증 이벤트를 처리하는 데 사용됩니다 . 서버 인스턴스용으로 사용할 인증 영역을 지정하려면 다음과 같이 합니다 .

1. Server Manager 에 액세스하고 Java 탭을 선택합니다 .
2. Java Security 링크를 누릅니다 .
3. 다음 정보를 설정합니다 .
   - **Default Realm**. 이 서버 인스턴스용으로 사용 중인 인증 영역 (AUTHREALM 이름 속성) 을 지정합니다 .
   - **Anonymous Role** (선택). 기본 또는 익명 역할의 이름으로 사용됩니다 .
   - **Audit Enabled** (선택). true 인 경우 감사 정보를 제공할 추가 로깅이 수행됩니다 . 감사 정보의 내용은 다음과 같습니다 .
     - 인증 성공 및 실패 이벤트 .
     - 서브릿 액세스 허가 및 거부

- **Log Level(** 선택). 오류 로그에 기록될 메시지의 유형을 제어합니다 .

4. OK 를 누릅니다 .

프로그램적 보안 사용

영역이 제공하는 컨테이너 관리 인증에 더하여 Sun ONE Web Server 6.1 은 또한 프로그램적 로그인 인터페이스를 통하여 액세스된 관리 인증을 지원합니다 . 이 인터페이스는 영역 인프라에 맞지 않는 사용자 정의 인증 모델을 지원합니다 . 또한 J2EE 응용 프로그램은 프로그램적 로그인을 사용하여 자체용 인증 컨텍스트를 직접 설정할 수 있습니다 . 그러나 이러한 방법은 응용 프로그램의 이식성과 관리 용이성을 떨어뜨리므로 권장하지 않습니다 .

응용 프로그램용 프로그램적 로그인 메커니즘을 시작하라면 `ProgrammaticLoginPermission` 허용이 필요합니다 . 이 권한은 표준 J2EE 메커니즘이 아니므로 구현된 응용 프로그램의 기본 권한으로 부여될 수는 없습니다 .

Sun ONE Web Server 6.1 은 Security Manager 를 지원합니다 . Security Manager 는 서버를 처음 설치할 때 기본적으로 사용하지 않도록 설정됩니다 . 서버 인스턴스에서 Java Security Manager 를 사용하도록 설정한 경우 프로그램적 로그인을 사용하는 모든 웹 응용 프로그램에 이 권한을 부여해야 합니다 .

해당 응용 프로그램에 필요한 권한을 허용하려면 `server.policy` 파일을 편집해야 합니다 .

`server.xml` 파일의 표준 Java 정책 항목을 지정하여 정책 지원을 사용하도록 설정할 수 있습니다 .

```
<JVMOPTIONS>-Djava.security.manager</JVMOPTIONS>
```

```
<JVMOPTIONS>-Djava.security.policy=install-root/https-servername/config/server.policy</JVMOPTIONS>
```

`server.policy` 파일에 대한 자세한 내용은 Sun ONE Web Server 6.1 *Programmer's Guide to Web Applications* 를 참조하십시오 .

J2EE/ 서브릿 인증 모델의 사용 시기 결정

이 부분은 J2EE/ 서브릿 기반 인증 모델을 사용하도록 결정해야 하는 주변 조건을 이해하는데 도움이 되도록 만들어졌습니다 .

J2EE/ 서브릿 인증 모델 사용 :

- 보통 대부분의 신규 J2EE/ 서브릿 기반 응용 프로그램에 사용합니다 .
- 변경할 계획이 없는 기존 `.war` 파일에 사용합니다 .
- 현재 또는 미래에 J2EE/ 서브릿 호환성이 중요한 웹 응용 프로그램을 만들 때 사용합니다 .
- ACL 이 폼 기반 인증을 지원하지 않으므로 폼 기반 인증을 사용하려 할 때 사용합니다 .

ACL 기반 인프라를 사용하는 경우라도 여전히 원시 영역 Java 영역을 사용하여 사용자 신분을 입력함으로써 서브릿용으로 사용할 수 있도록 할 수 있습니다 .

5 장

관리 기본 설정

Preferences and Global Settings 탭의 페이지를 사용하여 Administration Server 를 구성할 수 있습니다 . 참고로 서버를 구성하는 데 필요한 CGI 프로그램을 실행하려면 브라우저가 쿠키를 사용하도록 설정해야 합니다 .

이 장의 내용 :

Administration Server 종료

서버는 설치된 후 지속적으로 실행되어 HTTP 요청을 청취하고 허용합니다 . 예를 들어 , JDK(Java Development Kit) 또는 Directory Server 를 설치하거나 청취 소켓 설정을 변경하는 등의 경우 서버를 종료하고 재시작해야 할 수 있습니다 .

다음 중 한 가지 방법으로 서버를 정지시킬 수 있습니다 .

- Administration Server 에 액세스하고 Preference 탭을 선택한 후 , Shut Down 링크를 선택하고 "Shut down the administration server!" 버튼을 누릅니다 .

자세한 내용은 온라인 도움말의 Shut Down 페이지를 참조하십시오 .

- 제어판의 서비스 창을 사용합니다 (Windows).
- `stop` 을 사용하면 서버가 완전히 종료되며 서버가 다시 시작할 때까지 서비스가 중단됩니다 .

서버를 종료하면 서버가 종료 과정을 완료하고 상태를 "Off" 로 변경하는 데 약간의 시간이 걸릴 수 있습니다 .

청취 소켓 설정 편집

서버가 요청을 처리하려면 청취 소켓을 통하여 요청을 접수한 후 요청을 올바른 가상 서버로 보내야 합니다 . Sun ONE Web Server 를 설치하는 경우 청취 소켓 한 개 (ls1) 가 자동으로 만들어집니다 . 이 청취 소켓은 IP 주소 0.0.0.0 과 설치 도중 HTTP 서버 포트 번호로 지정한 포트 번호 (기본 값은 8888) 를 사용합니다 . 기본 청취 소켓은 삭제할 수 없습니다 .

Administration Server 의 Listen Sockets Table 을 사용하여 서버의 청취 소켓 설정을 편집할 수 있습니다 . 이 테이블에 액세스하려면 다음과 같이 합니다 .

1. Administration Server 에 액세스하고 Preferences 탭을 누릅니다 .
2. Edit Listen Sockets 링크를 누릅니다 .
3. 원하는 사항을 변경한 다음 OK 를 누릅니다 .

더 자세한 내용은 제 13 장 , " 가상 서버 사용 " 와 온라인 도움말의 Edit Listen Sockets 페이지를 참조하십시오 .

사용자 계정 변경 (UNIX/Linux)

Server Settings 페이지에서 UNIX 및 Linux 컴퓨터에 있는 웹 서버용 사용자 계정을 변경할 수 있습니다 . 서버의 모든 프로세스는 이 사용자로 실행됩니다 .

1024 보다 큰 포트 번호를 사용하며 `root` 사용자로 실행하지 않는 경우 서버 사용자를 지정할 필요는 없습니다 . (이 경우 서버를 시작할 때 `root` 로 로그할 필요가 없습니다 .) 여기에 사용자 계정을 지정하지 않으면 서버는 사용자가 시작할 때 사용한 사용자 계정으로 실행됩니다 . 서버를 시작할 때 올바른 사용자 계정을 사용해야 합니다 .

참고 시스템에서 신규 사용자를 만드는 방법을 모르는 경우에는 시스템 관리자에게 문의하거나 시스템 설명서를 참조하십시오 .

서버를 루트로 시작하는 경우라도 서버를 항상 루트로 실행하면 안 됩니다 . 서버의 시스템 리소스 액세스가 제한되어야 하며 권한이 부여되지 않은 사용자로 실행되어야 합니다 . 서버 사용자로 입력한 사용자 이름은 이미 정상적인 UNIX/Linux 사용자 계정으로 존재해야 합니다 . 서버가 시작되면 이 사용자로 실행됩니다 .

신규 사용자 계정을 만들지 않으려면 nobody 사용자 또는 동일한 호스트에서 실행되는 다른 HTTP 서버가 사용하는 계정을 선택할 수 있습니다 . 그러나 시스템에 따라 nobody 사용자가 파일을 소유할 수는 있으나 프로그램은 실행할 수 없는 경우도 있습니다 .

Server Settings 페이지에 액세스하려면 다음과 같이 합니다 .

1. Administration Server 에 액세스하고 Preferences 탭을 선택합니다 .
2. Server Settings 링크를 누릅니다 .
3. 원하는 사항을 변경한 다음 OK 를 누릅니다 .

수퍼유저 설정 변경

Administration Server 용 수퍼유저 액세스를 구성할 수 있습니다 . 이 설정은 오직 수퍼유저 계정에만 영향을 미칩니다 . 즉 , Administration Server 가 분산 관리를 사용하는 경우 허용하는 관리자에 대한 추가 액세스 제어를 설정해야 합니다 .

주의 사용자 및 그룹 관리에 Sun ONE Directory Server 를 사용하는 경우 , 수퍼유저 사용자 이름이나 암호를 변경하기 *전에* 해당 디렉토리에서 수퍼유저 항목을 업데이트해야 합니다 . 디렉토리를 먼저 업데이트하지 않는 경우 Administration Server 의 Users & Groups 폼에 액세스할 수 없게 됩니다 . 이 문제를 해결하려면 디렉토리에 대한 액세스 권한이 없는 관리자 계정으로 Administration Server 에 액세스하거나 Sun ONE Directory Server 의 콘솔 또는 구성 파일을 사용하여 디렉토리를 업데이트해야 합니다 .

Administration Server 용 수퍼유저 설정을 변경하려면 다음과 같이 합니다 .

1. Administration Server 에 액세스하고 Preferences 탭을 선택합니다 .
2. Superuser Access Control 링크를 누릅니다 .

3. 원하는 사항을 변경한 다음 OK 를 누릅니다 .

참고 운영 시스템의 Administration Server 사용자를 루트에서 다른 사용자로 변경하여 여러 사용자 (그룹에 속한 사용자) 가 구성 파일을 편집 / 관리하도록 할 수 있습니다. 그러나 UNIX/Linux 플랫폼의 경우 설치자는 그룹에게 구성 파일에 대한 "rw"(읽기 / 쓰기) 권한을 부여할 수 있는 반면 Windows 플랫폼의 경우 사용자는 반드시 "Administrators" 그룹에 속해야 합니다 .

수퍼유저의 사용자 이름과 암호는 *server_root*/https-admserv/config/admpw 파일에 저장됩니다 . 사용자 이름을 잊은 경우에는 이 파일에서 실제 이름을 확인할 수 있으나 암호는 암호화되어 읽을 수 없습니다 . 파일의 형식은 username:password 입니다 . 암호를 잊은 경우에는 admpw 파일을 편집하여 간단히 암호화된 암호를 삭제합니다 . 그런 후 Server Manager 폼으로 이동하여 새 암호를 지정합니다 .

주의 admpw 파일을 편집할 수 있으므로 서버 컴퓨터를 안전한 위치에 보관하고 파일 시스템으로의 액세스를 제한하는 것이 중요합니다 .

- UNIX/Linux 시스템의 경우 파일의 소유권을 변경하여 루트나 Administration Server 데몬을 실행하는 시스템 사용자만 쓸 수 있도록 합니다 .
- Windows 시스템의 경우 파일의 소유권을 Administration Server 가 사용하는 사용자 계정으로 제한합니다 .

복수 관리자 허용

여러 관리자가 분산 관리를 통하여 서버의 특정한 부분을 변경할 수 있습니다 .

참고 분산 관리를 사용하려면 반드시 기본 Directory Service가 LDAP 기반 디렉토리 서비스여야 합니다 .

분산 관리의 경우 두 가지 수준의 사용자가 있습니다 .

- **수퍼유저**는 server_root/https-admserv/config/admpw 파일의 목록에 있는 사용자입니다 . 이는 설치시 지정한 관리자 이름 (또한 암호) 입니다 . 이 사용자는 Users & Group 폼을 제외한 Administration Server 의 모든 폼에 액세스할 수 있습니다 . Users & Group 폼의 경우 수퍼유저가 Sun ONE Directory Server 등의 LDAP 서버에 유효한 계정이 있어야 합니다 .

- **administrators** 는 Administration Server 를 포함하여 특정 서버의 Server Manager 폼으로 직접 이동할 수 있습니다 . 표시되는 폼은 administrator 용으로 설정된 액세스 제어 규칙 (보통 수퍼유저가 설정) 에 따라 다릅니다 . Administrators 는 제한된 관리 작업을 수행하며 사용자 추가나 액세스 제어 변경 등 다른 사용자에게 영향을 미치는 사항을 변경할 수 있습니다 .

액세스 제어에 대한 자세한 내용은 제 9 장 , " 서버 액세스 제어 " 의 " 액세스 제어 설명 ," 을 (페이지 175) 참조하십시오 .

참고 분산 관리를 사용하기 전에 반드시 Directory Server 를 설치해야 합니다 . 더 자세한 내용은 Sun ONE Web Server *설치 및 이전 설명서*와 Sun ONE Directory Server *관리자 설명서*를 참조하십시오 .

관리 서버를 사용하려면 다음과 같이 합니다 .

1. Directory Server 가 설치되었는지 확인합니다 .
2. Administration Server 로 액세스합니다 .
3. Directory Server 를 설치한 후 , administration 그룹을 만들지 않았으면 이 그룹을 만들어야 합니다 .

   그룹을 만들려면 다음과 같이 합니다 .

   a. Users & Groups 탭을 선택합니다 .
   b. New Group 링크를 누릅니다 .
   c. LDAP 디렉토리에 "administrators" 그룹을 만들고 Administration Server 또는 해당 서버 루트에 설치된 서버의 구성 권한을 부여하려는 사용자의 이름을 추가합니다 . "administrators" 그룹의 모든 사용자는 Administration Server 전체에 액세스할 수 있으나 액세스 제어를 사용하여 구성할 수 있는 서버 및 양식을 제한할 수 있습니다 .

주의 액세스 제어 목록을 만들면 분산 관리자 그룹이 목록에 추가됩니다 . "administrators" 그룹의 이름을 변경하는 경우 반드시 직접 액세스 제어를 편집하여 제어가 참조하는 그룹을 변경해야 합니다 .

4. Preferences 탭을 선택합니다 .
5. Distributed Admin 링크를 누릅니다 .
6. 원하는 사항을 변경한 다음 OK 를 누릅니다 .

자세한 내용은 온라인 도움말의 Distributed Administration 페이지를 참조하십시오 .

로그 파일 옵션 지정

Administration Server 로그 파일은 발생한 오류의 유형을 포함하여 서버에 대한 데이터와 서버 액세스에 대한 정보를 기록합니다 . 이 로그를 확인하여 서버 작동을 모니터하고 발생한 오류의 유형이나 특정 파일에 액세스한 시간 등의 데이터를 제공함으로써 문제를 해결할 수 있습니다 .

Log Preferences 페이지를 사용하여 Administration Server 로그에 기록되는 데이터의 유형과 형식을 지정할 수 있습니다 . 예를 들어 Administration Server 에 액세스하는 모든 클라이언트의 데이터를 선택하거나 로그에서 특정 클라이언트를 생략할 수 있습니다 . 또한 Common Logfile Format 을 선택할 수 있습니다 . 이는 서버에 대한 고정된 양의 정보를 제공합니다 . 또는 요구 사항에 맞추어 로그 파일 형식을 사용자 정의할 수 있습니다 .

Preferences 탭을 선택하고 Logging Options 링크를 눌러 Administration Server Log Preferences 페이지에 액세스합니다 .

자세한 내용은 온라인 도움말의 Logging Options 페이지와 제 10 장 , " 로그 파일 사용 " 을 참조하십시오 .

로그 파일 확인

Administration Server 로그 파일은 서버 루트 디렉토리의 `admin/logs` 에 있습니다 . 예를 들어 Windows 의 경우 로그 파일의 경로는 `c:\Sun\server6\https-admserv\logs` 와 유사할 것입니다 . Sun ONE Web Server 콘솔을 통하거나 텍스트 편집기를 사용하여 오류 로그와 액세스 로그를 모두 확인할 수 있습니다 .

액세스 로그 파일

액세스 로그에는 서버로 오고 가는 요청에 대한 정보가 기록됩니다 .

액세스 로그 파일을 보려면 다음과 같이 합니다 .

1. Administration Server 에 액세스하고 Preferences 탭을 선택합니다 .
2. View Access Log 링크를 누르고 OK 를 누릅니다 .

자세한 내용은 온라인 도움말의 View Error Log 페이지와 제 10 장 , " 로그 파일 사용 " 을 참조하십시오 .

오류 로그 파일

오류 로그에는 로그 파일이 생성된 후 서버에 발생한 모든 오류 목록이 기록됩니다 . 또한 서버가 시작된 시간과 서버에 로그를 시도했으나 실패한 사용자 등의 서버에 대한 정보 메시지가 포함됩니다 .

오류 로그 파일을 보려면 다음과 같이 합니다 .

1. Administration Server 에 액세스하고 Preferences 탭을 선택합니다 .
2. View Error Log 링크를 누르고 OK 를 누릅니다 .

자세한 내용은 온라인 도움말의 View Access Log 페이지와 제 10 장 , " 로그 파일 사용 " 을 참조하십시오 .

로그 파일 보관

로그 파일이 자동 보관되도록 설정할 수 있습니다 . 특정 시간 또는 특정 시간의 경과 후 Sun ONE Web Server 는 액세스 로그를 순환시킵니다 . Sun ONE Web Server 는 이전 로그 파일을 저장하고 파일이 저장된 일자 및 시간이 포함된 이름을 파일에 지정합니다 .

예를 들어 , 파일이 매시간 순환되도록 설정하면 Sun ONE Web Server 는 파일을 "`access.199907152400`" 이라는 이름으로 저장합니다 . 여기에서 " 이름 | 년 | 월 | 일 |24 시간 형식 시간 " 은 단일 문자열로 합쳐집니다 . 액세스 로그 보관 파일의 정확한 형식은 설정한 로그 순환 유형에 따라 달라집니다 .

액세스 로그 순환은 서버가 시작할 때 초기화됩니다 . 순환을 사용하는 경우 Sun ONE Web Server 는 시간 스탬프 액세스 로그 파일을 만들고 서버가 시작할 때 순환이 시작됩니다 .

순환이 시작되면 Sun ONE Web Server 는 액세스 로그 파일애 기록해야 할 요청이 있는 경우 새로운 시간 스탬프 액세스 로그 파일을 만들며, 또한 이 작업은 미리 설정된 " 다음 순환 시간 " 이 경과하면 수행됩니다 .

스케줄된 제어 기준 로그 순환 사용 (UNIX/Linux)

Sun ONE Web Server 의 다양한 기능이 자동으로 수행되고 일정한 시간에 시작되도록 구성할 수 있습니다 . `schedulerd` 컨트롤 데몬은 컴퓨터 시계를 확인하고 일정한 시간에 프로세스를 생성합니다 . (이 설정은 `schedulerd` 파일에 저장됩니다 .)

이 schedulerd 컨트롤 데몬은 Sun ONE Web Server 용 cron 작업을 제어하며 Administration Server 에서 활성화 또는 비활성화할 수 있습니다 . cron 프로세스가 수행하는 작업은 다양한 서버에 따라 다릅니다 . (참고로 Windows 플랫폼의 경우 스케줄 작업은 개별 서버에서 수행됩니다 .)

schedulerd 컨트롤 데몬으로 제어할 수 있는 작업에는 컬렉션 유지 보수 스케줄 및 로그 파일 보관 등을 포함하여 여러 가지입니다 . 스케줄된 작업용 설정을 변경하면 schedulerd 컨트롤 데몬을 재시작해야 합니다 .

schedulerd 컨트롤 데몬을 재시작 , 시작 또는 정지하려면 다음과 같이 합니다 .

1. Administration Server 에 액세스하고 Global Settings 탭을 선택합니다 .
2. Cron Control 링크를 누릅니다 .
3. Start, Stop 또는 Restart 를 눌러 schedulerd 컨트롤을 변경합니다 .

참고로 schedulerd 에 작업을 추가하면 항상 데몬을 재시작해야 합니다 .

Directory Service 구성

LDAP(Lightweight Directory Access Protocal) 이라고 하는 개방형 시스템 서버 프로토콜을 사용하여 단일 Directory Server 에 있는 사용자 이름 및 암호 등의 정보를 저장하고 관리할 수 있습니다 . 또한 사용자가 쉽게 액세스 할 수 있는 복수 네트워크 위치에서 디렉토리 정보를 검색할 수 있도록 서버를 구성할 수 있습니다 .

디렉토리 서비스 기본 설정을 구성하려면 다음과 같이 합니다 .

1. Administration Server 에 액세스하고 Global Settings 탭을 선택합니다 .
2. Configure Directory Service 링크를 누릅니다 .
3. 원하는 사항을 변경한 다음 OK 를 누릅니다 .

자세한 내용은 온라인 도움말의 Configure Directory Service 페이지를 참조하십시오 .

서버 액세스 제한

전체 서버 또는 서버의 일부분 (디렉토리 , 파일 , 파일 유형 등) 에 대한 액세스를 제어할 수 있습니다 . 서버는 입중계 요청을 평가할 때 ACE(Access-control Entries) 라고 하는 규칙 계층에 따라 액세스를 결정하며 , 그런 후 일치되는 항목을 사용하여 요청의 허가 여부를 결정합니다 . 각 ACE 는 서버가 계층의 다음 ACE 로 계속할 것인

지의 여부를 지정합니다 . ACE 의 컬렉션은 액세스 제어목록 (ACL) 이라고 합니다 . 서버로 요청이 들어오면 서버는 *vsclass*.obj.conf(여기에서 *vsclass* 는 가상 서버 클래스 이름) 에서 액세스 제어 목록 (ACL) 에 대한 참조를 찾으며 , 이는 다시 액세스를 결정하는데 사용됩니다 . 기본으로 서버에는 하나의 액세스 제어 목록 (ACL) 파일이 있으며 여기에는 여러 개의 ACL 이 있습니다 .

Administration Server 를 통하여 모든 서버용으로 액세스 제어를 전역적으로 설정하거나 Server Manager 를 통하여 특정 서버 인스턴스 내의 리소스용으로 설정할 수 있습니다 . 리소스용으로 액세스 제어를 설정하는 방법은 페이지 188 에 있는 제 9 장 , " 서버 액세스 제어 " 의 " 액세스 제어 설정 ," 를 참조하십시오 .

참고 서버 액세스를 제한하기 전에 반드시 분산 서버를 사용하도록 설정해야 합니다 .

Sun ONE Web Server 에 대한 액세스를 제한하려면 다음과 같이 합니다 .

1. Administration Server 에 액세스하고 Global Settings 탭을 선택합니다 .
2. Restrict Access 링크를 누릅니다 .
3. 원하는 서버를 선택하고 Create ACL 을 누릅니다 .

   Administration Server 에 지정한 서버용 액세스 제어 규칙이 표시됩니다 .
4. 원하는 액세스 제어를 변경한 다음 OK 를 누릅니다 . 자세한 내용은 온라인 도움말의 Restrict Access 페이지를 참조하십시오 .

6 장

인증서 및 키 사용

이 장에서는 인증서 및 키 인증을 사용하여 Sun ONE Web Server 6.1 의 보안을 강화하는 방법에 대하여 설명합니다 . 또한 다양한 보안 기능을 사용하여 데이터를 보호하고 침입자 액세스를 거부하며 원하는 사용자의 액세스만 허용하는 방법에 대하여 설명합니다 . Sun ONE Web Server 6.1 에는 모든 Sun ONE 서버의 보안 아키텍처가 포함되어 있으며 , 이 아키텍처는 최대의 상호 운영성과 일관성을 위하여 업계 표준 및 공용 프로토콜을 기반으로 구축되었습니다 .

이 장을 읽기 전에 공용 키 암호화에 대한 기본 개념을 알고 있어야 합니다 . 이 개념에는 암호화 및 복호화 , 공용 및 개인 키 , 전자 인증서 , 암호화 프로토콜 등이 포함됩니다 . 자세한 내용은 *SSL 개요*를 참조하십시오 .

웹 서버를 보안하는 과정은 다음 부분에서 자세히 설명합니다 .

인증서 기반 인증

인증은 신분을 확인하는 과정입니다 . 네트워크 상호작용이라는 맥락에서 인증은 한 쪽이 다른 쪽의 신분을 명확히 확인하는 것입니다 . 인증서는 인증을 지원하는 방법 중 한 가지입니다 .

인증용 인증서 사용

인증서는 개인 , 회사 또는 기타 단체의 이름을 명시하는 디지털 데이터로 구성되며 인증서에 포함된 공용 키가 해당 단체의 소유인지 검사합니다 . 클라이언트와 서버는 모두 인증서를 가질 수 있습니다 .

인증서는 인증 기관 (또는 CA) 이 발행하고 전자적으로 서명합니다 . CA 는 인터넷에서 인증서를 판매하는 회사일 수도 있으며 회사의 인트라넷 또는 익스트라넷용으로 인증서를 발행하는 부서일 수도 있습니다 . 다른 사람의 신분을 확인하는데 충분히 신뢰할 수 있는 CA 를 선택합니다 .

인증서에 의하여 확인되는 공용 키와 단체의 이름에 더하여 인증서에는 또한 만기일 , 인증서를 발행한 CA 의 이름 및 발행 CA 의 " 전자 서명 " 이 포함됩니다 . 인증서의 내용과 형식에 대한 자세한 내용은 *SSL 개요를 참조하십시오* .

참고 서버 인증서는 반드시 암호화 기능을 사용하기 전에 설치되어야 합니다 .

서버 인증

서버 인증이란 클라이언트가 서버의 신분을 확인하는 것을 말합니다 . 즉 , 특정 네트워크 주소에서 서버에 대한 책임이 있는 단체의 신분을 확인하는 것입니다 .

클라이언트 인증

클라이언트 인증이란 서버가 클라이언트의 신분을 확인하는 것으로 , 즉 , 클라이언트 소프트웨어를 사용하는 사람의 신분을 확인하는 것입니다 . 개인이 여러 개의 신분증을 가질 수 있는 것처럼 클라이언트에는 여러 개의 인증서가 있을 수 있습니다 .

가상 서버 인증서

각 가상 서버마다 다른 인증서 데이터베이스를 부여할 수 있습니다 . 각 가상 서버 데이터베이스에는 여러 개의 인증서가 있을 수 있습니다 . 가상 서버에는 또한 각 인스턴스마다 서로 다른 인증서가 있을 수 있습니다 .

신뢰 데이터베이스 생성

서버 인증서를 요청하기 전에 반드시 신뢰할 수 있는 데이터베이스를 만들어야 합니다 . Sun ONE Web Server 에서는 Administration Server 와 각 서버 인스턴스에 자체의 신뢰 데이터베이스를 부여할 수 있습니다 . 신뢰 데이터베이스는 오직 로컬 컴퓨터에 만들어야 합니다 .

신뢰 데이터베이스를 만들 때 키쌍 파일용으로 사용할 암호를 지정합니다 . 또한 암호화된 통신을 사용하여 서버를 시작할 때에도 이 암호가 필요합니다 . 암호를 변경하는 경우 고려해야 할 지침은 " 암호 또는 PIN 변경 " 페이지 148 를 참조하십시오 .

신뢰 데이터베이스에서 공용 및 개인 키를 만들고 저장할 수 있습니다 . 이는 키쌍 파일이라고 합니다 . 키쌍 파일은 SSL 암호화에 사용됩니다 . 키쌍 파일은 서버 인증서를 요청하고 설치할 때 사용합니다 . 인증서가 설치되면 신뢰 데이터베이스에 저장됩니다 . 키쌍 파일은 다음 디렉토리에 암호화되어 저장됩니다 .

server_root/alias/<serverid-hostname>-key3.db.

Administration Server 에는 오직 하나의 신뢰 데이터베이스만 있습니다 . 각 서버 인스턴스에는 자체의 신뢰 데이터베이스를 부여할 수 있습니다 . 가상 서버는 해당 서버 인스턴스용으로 만들어진 신뢰 데이터베이스를 사용합니다 .

신뢰 데이터베이스 생성

신뢰 데이터베이스를 만들려면 다음과 같이 합니다 .

1. Administration Server 또는 Server Manager 에 액세스하고 Security 탭을 선택합니다 .

   Server Manager 의 경우 반드시 드롭 다운 목록에서 서버 인스턴스를 먼저 선택해야 합니다 .

2. Create Database 링크를 누릅니다 .
3. 데이터베이스용 비밀번호를 입력합니다 .
4. 반복 .
5. OK 를 누릅니다 .
6. Server Manager 의 경우 Apply 를 누른 다음 Restart 를 눌러 변경 사항이 적용되도록 합니다 .

password.conf 사용

기본적으로 시작하기 전에 관리자에게 키 데이터베이스 암호를 입력하라는 프롬프트가 웹 서버에 표시됩니다 . 무인 작업으로 웹 서버를 재시작하려는 경우에는 암호를 `password.conf` 파일에 저장해 놓아야 합니다 . 시스템이 적절히 보호되어 이 파일과 암호 데이터베이스가 조작되지 않을 경우에만 이 기능을 사용하십시오 .

보통 UNIX SSL 을 사용하는 서버의 경우 시작하기 전에 암호가 필요하므로 `etc/rc.local` 또는 `etc/inittab` 파일을 사용할 수 없습니다 . 암호를 일반 텍스트 파일에 저장하여 SSL 을 사용하는 서버를 자동으로 시작할 수는 있으나 , 이는 권장하지 않습니다 . 서버의 `password.conf` 파일은 루트 또는 서버를 설치한 사용자의 소유여야 하며 , 오직 소유자만 이 파일을 읽고 쓸 수 있어야 합니다 .

UNIX 의 경우 SSL 을 사용하는 서버의 암호를 `password.conf` 파일에 남겨두면 보안상의 위험이 커집니다 . 파일에 액세스할 수 있는 사용자는 모두 SSL 을 사용하는 서버의 암호를 알 수 있습니다 . SSL 을 사용하는 서버의 암호를 `password.conf` 파일에 보관하기 전에 보안의 위험에 대하여 고려해야 합니다 .

Windows 에서 NTSF 파일 시스템을 사용하는 경우 , `password.conf` 파일을 사용하지 않는 경우라도 이 파일이 들어 있는 디렉토리에 대한 액세스를 제한하여 보호해야 합니다 . 디렉토리에는 관리 서버 사용자 및 웹 서버 사용자용 읽기 / 쓰기 권한이 있어야 합니다 . 디렉토리를 보호하면 다른 사람이 잘못된 `password.conf` 파일을 만들 수 없도록 방지합니다 . FAT 파일 시스템의 경우 액세스를 제한하는 경우에도 디렉토리를 보호할 수 없습니다 .

SSL 사용 서버 자동 시작

보안의 위험을 걱정하지 않는 경우 다음과 같이 SSL 을 사용하는 서버를 자동으로 시작할 수 있습니다 .

1. SSL 이 사용되는지 확인합니다 .
2. 서버 인스턴스의 `config` 하위 디렉토리에 `password.conf` 파일을 새로 만듭니다 .
   - 서버와 함께 제공되는 내부 PKCS#11 소프트웨어 암호화 모듈을 사용하는 경우에는 다음 정보를 입력합니다 .

     `internal:`*`your_password`*
   - 다른 PKCS#11 모듈(하드웨어 암호화 또는 하드웨어 가속용)을 사용하는 경우에는 해당 PKCS#11 모듈의 이름과 암호를 지정합니다 . 예 :

     `nFast:`*`your_password`*

3. 서버를 종료한 후 다시 시작하여 변경 사항이 적용되도록 합니다 .

`password.conf` 파일을 만든 후라도 웹 서버를 시작할 때에는 항상 암호를 입력하라는 프롬프트가 표시됩니다 .

VeriSign 인증서 요청 및 설치

VeriSign 은 Sun ONE Web Server 에서 자주 사용되는 인증 기관입니다 . VeriSign 의 VICE 프로토콜을 사용하면 인증서 요청 프로세스를 간단히 할 수 있습니다 . VeriSign 에는 인증서를 사용자의 서버로 직접 회신할 수 있다는 장점이 있습니다 .

서버용 인증서 신뢰 데이터베이스를 만든 후 인증서를 요청하고 이를 인증기관 (CA) 에 제출할 수 있습니다 . 회사에 내부 CA 가 있는 경우에는 해당 CA 로 인증서를 요청합니다 . 상용 CA 로부터 인증서를 구매할 계획인 경우에는 CA 를 선택하고 필요한 정보의 형식이 있는지 문의합니다 . Request a Certificate 페이지에 사용 가능한 인증 기관 목록과 해당 사이트로의 링크가 있습니다 . CA 에 필요한 것에 대한 더 자세한 내용은 Request a Certificate 에 있는 Server Administrator 또는 Server Manager Security 페이지에 있는 인증 기관 목록을 참조하십시오 .

Administration Server 에는 오직 하나의 서버 인증서만 부여할 수 있습니다 . 각 서버 인스턴스에는 자체의 서버 인증서를 부여할 수 있습니다 . 각 가상 서버에 대하여 서버 인스턴스 인증서를 선택할 수 있습니다 .

VeriSign 인증서 요청

VeriSign 인증서를 요청하려면 다음과 같이 합니다 .

1. Administration Server 또는 Server Manager 에 액세스하고 Security 탭을 선택합니다 .

   Server Manager 의 경우 반드시 드롭 다운 목록에서 서버 인스턴스를 먼저 선택해야 합니다 .

2. Request VeriSign Certificate 링크를 누릅니다 .
3. 필요한 과정을 확인합니다 .
4. OK 를 누릅니다 .
5. VeriSign 절차를 따라 합니다 .

VeriSign 인증서 설치

VeriSign 인증서를 요청하고 이에 대한 승인을 받으면 3 일 이내에 Install VeriSign Certificate 페이지의 드롭 다운 목록에 인증서가 표시됩니다 . VeriSign 인증서를 설치하려면 다음과 같이 합니다 .

1. Administration Server 또는 Server Manager 에 액세스하고 Security 탭을 선택합니다 .

   Server Manager 의 경우 반드시 드롭 다운 목록에서 서버 인스턴스를 먼저 선택해야 합니다 .

2. Install VeriSign Certificate 링크를 누릅니다 .
3. 외부 암호화 모듈을 사용하지 않는 경우 , 암호화 모듈용 드롭 다운 목록에서 internal(software) 를 선택합니다 .
4. 키 쌍 파일 암호 또는 PIN 을 입력합니다 .
5. 드롭 다운 목록에서 Transaction ID to Retrieve 를 선택합니다 .

   보통 마지막 항목을 선택합니다 .

6. OK 를 누릅니다 .
7. Server Manager 의 경우 Apply 를 누른 다음 Restart 를 눌러 변경 사항이 적용되도록 합니다 .

기타 서버 인증서 요청 및 설치

VeriSign 외에도 다른 인증 기관에 인증서를 요청하여 설치할 수 있습니다 . Request a Certificate 의 Server Administrator 와 Server Manager Security 페이지에 사용 가능한 CA 목록이 있습니다 . 회사나 조직에서 자체의 내부 인증서를 제공할 수도 있습니다 . 여기에서는 다른 종류의 서버 인증서를 요청하고 설치하는 방법에 대하여 설명합니다 .

필수 CA 정보

요청 프로세스를 시작하기 전에 CA 가 요구하는 정보가 무엇인지 알아야 합니다 . 상용 CA 또는 내부 CA 중 어느 곳에 서버 인증서를 요청할 것인가에 상관 없이 다음 정보를 제공해야 합니다 .

- **Common Name** 에는 DNS 조회에서 사용되는 유효한 호스트 이름을 지정합니다 (예를 들어 *www.sun.com*). 이는 브라우저가 사이트에 연결할 때 사용하는 URL 내의 호스트 이름입니다 . 이들 이름 두 개가 일치하지 않으면 클라이언트에게 인증서 이름이 사이트 이름과 일치하지 않는다는 통지가 보내지며 인증서의 신빙성이 의심 받게 됩니다 . CA 에 따라 요구 사항이 다를 수 있으므로 반드시 해당 CA 에 확인해야 합니다 .

  또한 내부 CA 에 인증서를 요청하는 경우에는 이 필드에 와일드카드와 정상적인 표현식을 입력할 수 있습니다 . 공급업체의 경우에는 대부분 공통 이름에 와일드카드나 정상 표현식이 있으면 인증서 요청을 승인하지 않습니다 .

- **Email Address** 는 업무용 전자우편 주소입니다 . 이 주소는 CA 와의 의견교환을 위하여 사용됩니다 .
- **Organization** 에는 회사 , 교육 기관 , 협력관계 등의 공식적 , 법적 이름을 지정합니다 . 대부분의 CA 는 정보에 대하여 법적 서류 (사업자 등록 등) 로 확인할 것을 요구합니다 .
- **Organizational Unit** 에는 회사 내의 조직에 대한 설명을 입력하는 선택 필드입니다 . 또한 비공식적인 회사 이름을 (*주식회사*, *법인* 등을 제외하고) 알리는 데 사용할 수 있습니다 .
- **Locality** 는 선택 필드로 조직의 소재 시 / 도 또는 국가를 입력합니다 .
- **State or Province** 는 보통 필수 항목이 아니지만 CA 에 따라 요구하는 경우도 있습니다 . 대부분의 CA 는 약자를 허용하지 않으나 , 해당 CA 에 확인하는 것이 좋습니다 .
- **Country** 는 필수 필드로 국가 이름의 두 자리 약자를 지정합니다 . (ISO 형식) 미국의 국가 코드는 US 입니다 .

이 모든 정보는 DN(Distinguished Name) 이라고 하는 일련의 속성 - 값 쌍으로 조합되어 인증서의 개체를 고유하게 구분합니다 .

상용 CA 에서 인증서를 구매하는 경우에는 반드시 CA 에 연락하여 인증서를 발행하기 위하여 필요한 추가 정보가 있는지 확인해야 합니다 . 대부분의 CA 는 신분에 대한 증명을 요구합니다 . 예를 들어 , 회사 이름에 대한 확인 , 회사가 서버를 관리하도록 지정한 사용자 등을 확인하며 사용자가 제공하는 정보를 사용할 법적 권한이 있는지 확인할 것입니다 .

일부 상용 CA 는 완벽한 신분을 제공하는 조직이나 개인에게 더 자세하고 정확한 인증서를 제공합니다 . 예를 들어 , 사용자가 www.sun.com 컴퓨터에 대한 관리의 권한이 사용자에게 있는지 확인하지 않았으나 , 3 년간 경영해온 회사로 유의할 고객 소송이 없었다는 사실을 표시하는 인증서를 구매할 수 있습니다 .

기타 서버 인증서 요청

인증서를 요청하려면 다음과 같이 합니다 .

1. Administration Server 또는 Server Manager 에 액세스하고 Security 탭을 선택합니다 .

   Server Manager 의 경우 반드시 드롭 다운 목록에서 서버 인스턴스를 먼저 선택해야 합니다 .

2. Request a Certificate 링크를 누릅니다 .

3. 신규 인증서인지 또는 인증서 갱신인지 선택합니다 .

   인증서는 대부분 6 개월이나 1 년 등 , 일정 시간이 경과하면 무효화됩니다 . CA 에 따라 자동으로 갱신을 송신하는 경우도 있습니다 .

4. 인증서용 요청을 제출하는 방법을 지정하려면 다음과 같이 합니다 .

   - CA 에 전자우편 메시지로 요청서를 송신해야 하는 경우에는 CA Email 을 선택하고 CA 의 전자우편 주소를 입력합니다 . CA 의 목록을 보려면 List of available certificate authorities 를 누릅니다 .
   - Netscape Certificate Server를 사용하는 내부 CA에 인증서를 요청하려면 CA URL 을 누르고 Certificate Server 용 URL 을 입력합니다 . 이 URL 은 인증서 요청을 처리하는 인증서 서버의 프로그램을 가리키는 URL 이어야 합니다 . URL 예 : `https://CA.mozilla.com:444/cms`.

5. 드롭 다운 목록에서 인증서를 요청할 때 사용할 키쌍 파일용 암호화 모듈을 선택합니다 .

6. 키쌍 파일용 암호를 입력합니다 .

   내부 모듈이 아닌 다른 암호화 모듈을 선택하지 않은 한 , 이 암호는 신뢰 데이터를 만들 때 지정한 암호입니다 . 서버는 암호를 사용하여 개인 키를 구하고 CA 로 전송되는 메시지를 암호화합니다 . 그런 후 , 서버는 *공용 키*와 암호화된 메시지를 모두 CA 로 전송합니다 . CA 는 공용 키를 사용하여 메시지를 해독합니다 .

7. ID 정보를 입력합니다 .

   이 정보의 형식은 CA 에 따라 다릅니다 . 이 필드에 대한 일반적인 설명은 Request a Certificate 에 있는 Server Administrator 또는 Server Manager Security 페이지에 있는 인증 기관 목록을 참조하십시오 . 참고로 이 정보의 대부분은 인증서 갱신의 경우에는 필요하지 않습니다 .

8. 입력한 사항이 정확한지 다시 한 번 확인합니다 .

   정보가 정확할 수록 인증서가 더욱 빨리 승인될 수 있습니다 . 요청이 서버 인증을 위한 것인 경우에는 요청을 제출하기 전에 양식 정보를 확인하라는 프롬프트가 표시될 것입니다 .

9. OK 를 누릅니다 .

10. Server Manager 의 경우 Apply 를 누른 다음 Restart 를 눌러 변경 사항이 적용되도록 합니다 .

서버가 정보를 포함하는 인증서 요청을 생성합니다 . 요청에는 개인 키를 사용하여 만든 전자 서명이 포함됩니다 . CA 는 전자 서명을 사용하여 요청이 서버 컴퓨터에서 CA 로 라우팅되는 동안 조작되지 않았는지 검사합니다 . 드물지만 요청이 조작된 경우에는 보통 CA 가 전화를 통하여 사용자에게 문의합니다 .

요청을 전자우편으로 보내는 경우 서버가 요청이 포함된 전자우편 메시지를 작성하고 메시지를 CA 로 전송합니다 . 이 후 , 보통 인증서는 전자우편을 통하여 회신됩니다 . 대신 인증 서버의 URL 을 지정하는 경우에는 서버가 URL 을 사용하여 요청을 Certificate Server 에 제출합니다 . CA 에 따라 회신은 전자우편을 통하거나 다른 방법을 통하여 수신됩니다 .

CA 가 인증서 발행에 동의하는 경우 해당 사실을 통지합니다 . 대부분의 경우 CA 는 전자우편을 통하여 인증서를 전송합니다 . 회사에서 인증 서버를 사용하는 경우 인증서 서버의 폼을 사용하여 인증서를 검색할 수 있습니다 .

참고 상용 CA 로 인증서를 요청하는 모든 사람에게 인증서가 발행되는 것은 아닙니다 . 많은 CA 가 인증서를 발행하기 전에 신분 증명을 요구합니다 . 또한 승인에는 하루에서 두 달까지 걸릴 수 있습니다 . 사용자는 CA 에 필요한 정보를 모두 신속히 제공할 책임이 있습니다 .

인증서를 받은 후 , 이를 설치할 수 있습니다 . 그 동안에는 SSL 없이 서버를 계속 사용할 수 있습니다 .

기타 서버 인증서 설치

CA 에서 인증서를 수신하면 인증서는 공용 키로 암호화되므로 오직 귀사만 이를 해독할 수 있습니다 . 오직 정확한 신뢰 데이터베이스용 암호를 입력해야만 인증서를 해독하고 설치할 수 있습니다 .

인증서에는 세 가지 유형이 있습니다 .

- 클라이언트에게 제시할 자체 서버의 인증서
- 인증서 체인에서 사용할 CA 의 자체 인증서
- 신뢰된 CA 의 인증서

인증서 체인이란 연속적인 인증 기관이 서명한 일련의 계층적 인증서를 말합니다 . CA 인증서는 인증기관 (CA) 을 확인하고 해당 기관이 발행한 인증서에 서명하는 데 사용됩니다 . 이 CA 인증서는 다시 상위 CA 의 CA 인증서에 의하여 서명되는 과정을 되풀이하여 루트 CA 의 서명까지 이어집니다 .

참고 CA 가 자동으로 인증서를 보내지 않는 경우에는 요청해야 합니다 . 많은 CA 가 귀사의 인증서가 있는 전자우편에 자체의 인증서를 포함하며 , 서버는 이 두 인증서를 동시에 설치합니다 .

CA 에서 인증서를 수신하면 인증서는 공용 키로 암호화되므로 오직 귀사만 이를 해독할 수 있습니다 . 인증서를 설치하면 서버는 지정한 키 쌍 파일 암호를 사용하여 이를 해독합니다 . 전자우편을 서버에 액세스할 수 있는 다른 위치에 저장하거나 전자우편의 텍스트를 복사한 후 Install Certificate 폼의 해당 위치에 붙여 넣을 수 있도록 합니다 .

인증서 설치

인증서를 설치하려면 다음과 같이 합니다 .

1. Administration Server 또는 Server Manager 에 액세스하고 Security 탭을 선택합니다 .

   Server Manager 의 경우 반드시 드롭 다운 목록에서 서버 인스턴스를 먼저 선택해야 합니다 .

2. Install Certificate 링크를 누릅니다 .
3. 설치하는 인증서의 유형을 선택합니다 .
   - This Server 는 오직 사용하는 서버에만 연결된 단일 인증서입니다 .
   - Server Certificate Chain 은 인증서 체인을 포함하는 CA 의 인증서용입니다 .
   - Trusted Certificate Authority(CA)는 클라이언트 인증을 위하여 신뢰된 CA로 승인하려는 CA 의 인증서용입니다 .
4. 드롭 다운 목록에서 Cryptographic Module 을 선택합니다 .
5. 키쌍 파일 암호를 입력합니다 .

6. 인증서가 이 서버 인스턴스용으로만 사용될 경우에는 인증서 이름 필드를 입력하지 않습니다 . 그렇지 않은 경우에는 다음과 같이 합니다 .
   - Multiple certificates will be used for virtual servers

     서버 인스턴스에 고유한 인증서 이름을 입력합니다 .
   - Cryptographic modules other than internal are used

     단일 암호화 모듈에 있는 모든 서버 인스턴스에 대하여 고유한 인증서 이름을 입력합니다 .

   이름을 입력하면 Manage Certificates 목록에 표시되므로 설명적인 이름을 입력하십시오 . 예를 들어 "United States Portal Service CA" 는 CA 의 이름이며 "VeriSign Class 2 Primary CA" 는 CA 와 인증서 유형을 모두 나타냅니다 . 인증서 이름을 입력하지 않으면 기본 값이 적용됩니다 .
7. 다음 중 한 가지를 선택합니다 .
   - 이 파일의 메시지 및 저장된 전자우편의 전체 경로 입력
   - 메시지 텍스트 (헤더 포함) 및 전자우편 텍스트 붙여 넣기 .

     텍스트를 복사하여 붙여 넣은 경우 반드시 헤더의 "Begin Certificate" 및 "End Certificate" 를 시작 및 끝 하이픈과 함께 포함해야 합니다 .
8. OK 를 누릅니다 .
9. 다음 중 한 가지를 선택합니다 .
   - 신규 인증서를 설치하는 경우 Add Certificate.
   - 인증서 갱신을 설치하는 경우 Replace Certificate.
10. Server Manager 의 경우 Apply 를 누른 다음 Restart 를 눌러 변경 사항이 적용되도록 합니다 .

인증서는 서버의 인증서 데이터베이스에 저장됩니다 . 파일 이름은 < 별칭 >-cert8.db 입니다 . 예 :

```
https-serverid-hostname-cert8.db
```

업그레이드시 인증서 이전

iPlanet Web Server 4.1 이나 6.0 에서 이전하는 경우 신뢰 및 인증서 데이터베이스를 포함하여 사용 중인 파일이 자동으로 업데이트됩니다 .

키쌍 파일 및 인증서는 서버의 보안 기능을 사용하는 경우에만 이전됩니다 . 또한 Administration Server 페이지와 Server Management 페이지에 있는 Security 탭을 사용하여 키와 인증서를 이전할 수 있습니다 .

이전 버전의 경우 인증서와 키쌍 파일은 여러 서버 인스턴스가 사용할 수 있는 별칭에 의하여 참조되었습니다 . Administration Server 가 모든 별칭과 해당 구성 인증서를 관리했습니다 . Sun ONE Web Server 6.1 의 경우 Administration Server 와 각 서버 인스턴스에는 자체의 인증서 및 키쌍 파일이 있으며 , 이는 별칭이 아닌 신뢰 인증서라고 합니다 .

Administration Server 의 경우에는 그 자체에서 , 서버 인스턴스의 경우에는 Server Manager 에서 서버 인증서와 모든 포함된 인증 기관을 포함하여 신뢰 데이터베이스와 해당 구성 인증서를 관리합니다 . 이제 인증서와 키쌍 데이터베이스 파일은 이를 사용하는 서버 인스턴스의 이름을 따라 이름이 지정됩니다 . 이전 버전에서 여러 서버 인스턴스가 동일한 별칭을 공유한 경우 , 이전된 인증서와 키쌍 파일의 이름은 새로운 서버 인스턴스용으로 변경됩니다 .

서버 인스턴스와 연결된 신뢰 데이터베이스 전체가 이전됩니다 . 이전 데이터베이스에 있는 모든 인증기관 목록이 Sun ONE Web Server 6.1 데이터베이스로 이전됩니다 . CA 가 중복되는 경우 유효 기간 동안 이전 CA 를 사용합니다 . 중복되는 CA 를 삭제하면 안 됩니다 .

내장 루트 인증서 모듈 사용 .

동적으로 로드할 수 있는 루트 인증서 모듈이 Sun ONE Web Server 6.1 에 포함되어 있으며 , 여기에는 VeriSign 을 비롯하여 많은 CA 용 루트 인증서가 있습니다 . 루트 인증서 모듈을 사용하면 이전보다 훨씬 쉽게 루트 인증서를 신규 버전으로 업그레이드할 수 있습니다 . 이전에는 오래된 루트 인증서를 한 번에 하나씩 삭제하고 새 인증서를 한 번에 하나씩 설치해야 했습니다 . 이제 잘 알려진 CA 인증서를 설치하는 경우 , 간단히 Sun ONE Web Server 의 신규 버전이나 Service Pack 이 발표될 때 루프 인증서 모듈 파일을 신규 버전으로 업데이트하면 됩니다 .

루트 인증서는 PKCS#11 암호화 모듈로 구현되므로 여기에 포함된 루트 인증서는 삭제할 수 없으며 , 해당 인증서를 관리하는 경우 삭제 옵션은 사용할 수 없게 됩니다 . 서버 인스턴스에서 루트 인증서를 제거하려면 서버의 `alias` 파일에서 다음을 삭제하여 루트 인증서 모듈을 사용하지 않도록 설정해야 합니다 .

- `libnssckbi.so` (대부분의 UNIX 플랫폼에 적용)
- `libnssckbi.sl` (HP-UX)

- `nssckbi.dll` (Windows)

이 후 , 루트 인증서 모듈을 복구하려면 `bin/https/lib` (UNIX 및 HP) 또는 `bin\https\bin`(Windows)에서 확장 기능을 별칭 하위 디렉토리로 복사하면 됩니다.

루트 인증서의 신뢰 정보를 수정할 수 있습니다 . 신뢰 정보는 루트 인증서 모듈 그 자체가 아니라 편집하는 서버 인스턴스용 인증서 데이터베이스에 기록되어 있습니다 .

인증서 관리

서버에 설치된 다양한 인증서의 신뢰 설정을 확인 , 삭제 또는 편집할 수 있습니다 . 여기에는 귀사 자체의 인증서와 CA 의 인증서가 포함됩니다 .

인증서 목록을 관리하려면 다음과 같이 합니다 .

1. Administration Server 또는 Server Manager 에 액세스하고 Security 탭을 선택합니다 .

   Server Manager 의 경우 반드시 드롭 다운 목록에서 서버 인스턴스를 먼저 선택해야 합니다 .

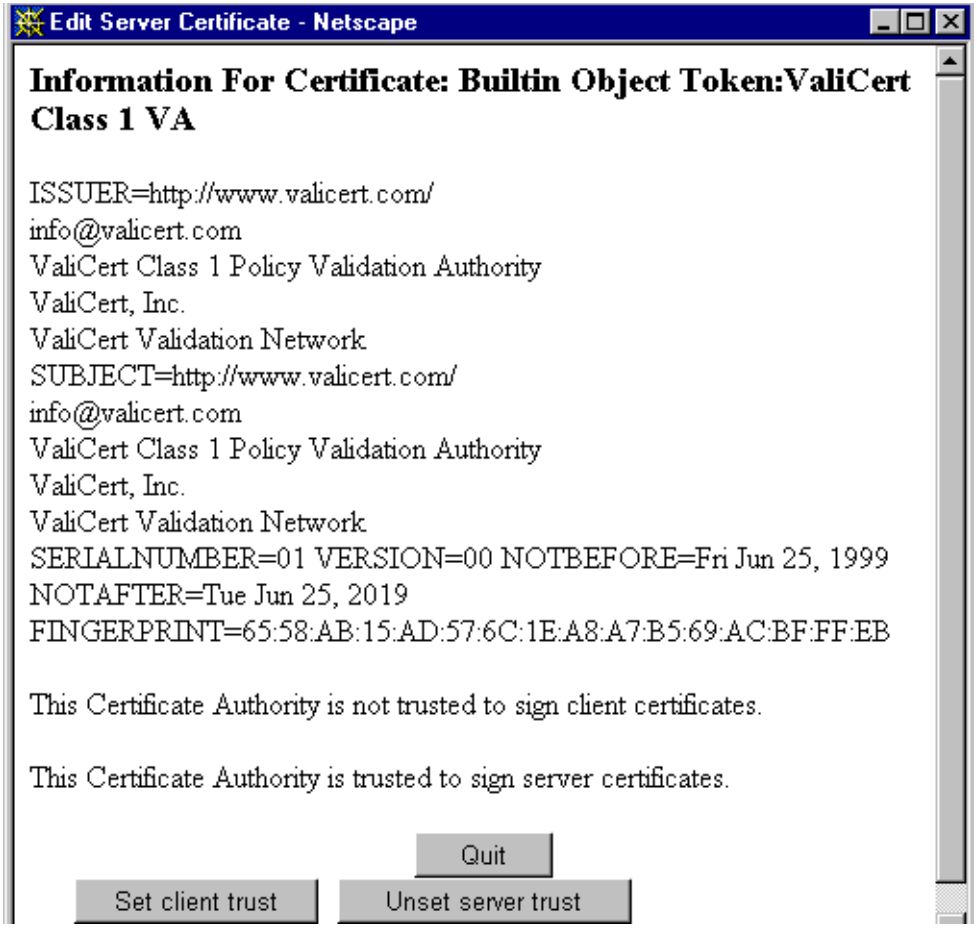
2. Manage Certificates 링크를 누릅니다 .

   - 내부 암호화 모듈을 사용하는 기본 구성용 인증서를 관리하는 경우에는 설치된 모든 인증서의 목록이 해당 유형 및 유효 기간과 함께 표시됩니다 . 모든 인증서는 *`server_root`*`/alias` 에 저장됩니다 .
   - 하드웨어 가속기 등의 외부 암호화 모듈을 사용하는 경우에는 우선 해당 모듈용 암호를 입력하고 OK 를 눌러야 합니다 . 인증서 목록이 해당 모듈에 있는 인증서를 포함하여 업데이트됩니다 .

3. 관리하려는 Certificate Name 을 누릅니다 .

   해당 유형의 인증서에 대한 관리 옵션이 있는 Edit Server Certificate 페이지가 표시됩니다 . 오직 CA 인증서의 경우에만 클라이언트 신뢰를 설정 또는 해제할 수 있습니다 . 외부 암호화 모듈에 따라 인증서를 삭제할 수 없는 경우도 있습니다 .

   Edit Server Certificate



4. Edit Server Certificate 창에서 다음 옵션을 선택할 수 있습니다 .
   - Delete Certificate 또는 Quit - 내부 인증서용
   - Set client trust, Unset server trust 또는 Quit - CA 인증서용
5. OK 를 누릅니다 .

6. Server Manager 의 경우 Apply 를 누른 다음 Restart 를 눌러 변경 사항이 적용되도록 합니다 .

인증서 정보에는 소유자와 발행자가 표시됩니다 .

신뢰 설정을 이용하여 클라이언트 신뢰를 설정하거나 서버 신뢰를 해제할 수 있습니다 . LDAP 서버 인증서의 경우 서버가 반드시 신뢰되어야 합니다 .

CRL 및 KRL 설치 / 관리

인증서 철회 목록 (CRL) 과 변조된 키 목록 (CKL) 은 클라이언트나 서버 사용자가 더 이상 신뢰하면 안 되는 인증서를 표시합니다 . 예를 들어 인증서의 유효기간이 끝나기 전에 사용자가 사무실을 이전하거나 퇴사하는 등 인증서의 데이터가 변경되면 인증서는 취소되며 CRL 에 해당 데이터가 표시됩니다 . 키가 조작 또는 변형되는 경우 해당 키와 데이터가 CKL 에 표시됩니다 . CRL 과 CKL 은 모두 CA 에 의하여 만들어지고 주기적으로 업데이트됩니다 .

CRL 또는 CKL 설치

CA 에서 CRL 또는 CKL 을 받으려면 다음과 같이 합니다 .

1. CRL 이나 CKL 을 다운로드할 CA 의 URL 을 확인합니다 .
2. 브라우저에 URL 을 입력하여 사이트로 이동합니다 .
3. CA 의 설명에 따라 CRL 또는 CKL 을 로컬 디렉토리로 다운로드합니다 .
4. Administration Server 또는 Server Manager 에 액세스하고 Security 탭을 선택합니다 .

   Server Manager 의 경우 반드시 드롭 다운 목록에서 서버 인스턴스를 먼저 선택해야 합니다 .
5. Install CRL/CKL 링크를 누릅니다 .
6. 다음 중 한 가지를 선택합니다 .
   - Certificate Revocation List
   - Compromised Key List
7. 해당 파일의 전체 경로 이름을 입력합니다 .
8. OK 를 누릅니다 .

- Certificate Revocation List 를 선택하면 CRL 정보 목록이 있는 Add Certificate Revocation List 페이지가 표시됩니다 .
- Compromised Key List 를 선택하면 CKL 정보가 있는 Add Compromised Key List 페이지가 표시됩니다 .

참고 데이터베이스에 이미 CRL 또는 CKL 목록이 있는 경우에는 Replace Certificate Revocation List 또는 Replace Compromised Key List 페이지가 표시됩니다 .

9. Add 를 누릅니다 .
10. OK 를 누릅니다 .
11. Server Manager 의 경우 Apply 를 누른 다음 Restart 를 눌러 변경 사항이 적용되도록 합니다 .

CRL 및 CKL 관리

CRL 과 CKL 을 관리하려면 다음과 같이 합니다 .

1. Administration Server 또는 Server Manager 에 액세스하고 Security 탭을 선택합니다 .

   Server Manager 의 경우 반드시 드롭 다운 목록에서 서버 인스턴스를 먼저 선택해야 합니다 .
2. Manage CRL/CKLs 링크를 누릅니다 .

   Manage Certificate Revocation List/Compromised Key List 페이지가 나타나며 설치된 서버의 CRL 과 CKL 목록이 유효 기간과 함께 표시됩니다 .
3. Server CRL 또는 Server CKL 목록에서 Certificate Name 을 선택합니다 .
4. 옵션 :
   - Delete CRL
   - Delete CKL
5. Server Manager 의 경우 Apply 를 누른 다음 Restart 를 눌러 변경 사항이 적용되도록 합니다 .

보안 기본 설정

인증서를 만든 후 , 서버의 보안 작업을 시작할 수 있습니다 . Sun ONE Web Server 에는 다양한 보안 요소가 제공됩니다 .

암호화는 정보를 변환하여 의도된 수신자 외에 아무도 알아볼 수 없도록 하는 프로세스입니다 . 해독화는 암호화된 정보를 변환하여 다시 알아볼 수 있도록 하는 프로세스입니다 . Sun ONE Web Server 6.1 은 SSL 및 TLS 암호화 프로토콜을 지원합니다 .

암호**는** 암호화 알고리즘 (수학적 함수) 으로 암호화 또는 해독화에 사용됩니다 . SSL 및 TLS 프로토콜에는 다양한 암호 제품군이 포함됩니다 . 보안의 안전성과 강도는 암호마다 다릅니다 . 일반적으로 암호가 사용하는 비트의 수가 많을 수록 데이터를 해독하는 것이 어렵습니다 .

양방향 암호화 프로세스에서 양쪽에는 반드시 동일한 암호가 있어야 합니다 . 다양한 암호를 사용할 수 있으므로 서버를 가장 공통적으로 사용되는 암호용으로 설정해야 합니다 .

보안 연결에서 클라이언트와 서버는 양쪽이 통신에 사용할 수 있는 가장 강력한 암호화를 사용하도록 동의합니다 . 암호는 SSL2, SSL3 및 TLS 프로토콜 중 선택할 수 있습니다 .

참고 SSL 버전 2.0 이후 보안과 성능이 향상되었으므로 SSL3 를 사용할 수 있는 클라이언트가 아닌 경우에는 SSL2 를 사용하면 안 됩니다 . 클라이언트 인증서가 SSL 2 암호와 작동되도록 보장되지 않습니다 .

암호화 프로세스 그 자체로는 서버의 비밀 정보를 보안하는 데 충분하지 않습니다 . 실제의 암호화 결과를 얻거나 이전에 암호화된 정보를 해독하려면 암호화 암호와 함께 키가 사용되어야 합니다 . 암호화 프로세스에는 이 결과를 위하여 공용 키와 개인 키의 두 가지 키가 사용됩니다 . 공용 키로 암호화된 정보는 오직 연결된 개인 키로만 해독할 수 있습니다 . 공용 키는 인증서의 일부로 만들어지며 오직 연결된 개인 키만 보호됩니다 .

다양한 암호 제품군에 대한 설명과 키 및 인증서에 대한 자세한 내용은 *SSL 개요*를 참조하십시오 .

서버가 사용할 수 있는 암호를 지정하려면 목록에서 해당 목록을 선택합니다 . 특정 암호를 사용하면 안 되는 충분한 이유가 있지 않는 한 , 모두 선택해야 합니다 . 그러나 최적 암호화에 미치지 않는 암호를 사용하지는 않을 것입니다 .

주의 "No Encryption, only MD5 message authentication" 을 선택하면 안 됩니다 . 클라이언트 측에 사용 가능한 다른 암호가 없는 경우 서버는 설정을 기본값으로 되돌리며 암호화가 수행되지 않습니다 .

SSL 및 TLS 프로토콜

Sun ONE Web Server 6.1 은 암호화 통신용으로 SSL(Secure Sockets Layer) 및 TLS(Transport Layer Security) 프로토콜을 지원합니다 . SSL 과 TLS 는 응용 프로그램 종속적인 고수준 프로토콜로 응용 프로그램에 투명하게 배치될 수 있습니다 .

SSL 과 TLS 프로토콜은 서버와 클라이언트가 서로를 인증하고 , 인증서를 전송하며 세션 키를 설정하는 등의 작업에 사용되는 다양한 암호를 지원합니다 . 클라이언트와 서버는 지원하는 프로토콜 , 암호화 정도에 대한 회사 정책 , 암호화된 소프트웨어의 수출에 대한 정부 규제 등 , 다양한 요인에 따라 지원하는 암호 제품군이 달라집니다 . 다른 기능 중 SSL 과 TLS 핸드셰이크 프로토콜에 따라 서버와 클라이언트가 통신에 사용할 암호 제품군을 선택하는 방식이 결정됩니다 .

LDAP 와의 통신에 SSL 사용

Administration Server 는 SSL 을 사용하여 LDAP 와 통신하도록 해야 합니다 . Administration Server 에서 SSL 을 사용하도록 설정하려면 다음과 같이 합니다 .

1. Administration Server 에 액세스하고 Global Settings 탭을 선택합니다 .
2. Configure Directory Service 링크를 누릅니다 .
3. Yes to use Secure Sockets Layer(SSL) for connections 를 선택합니다 .
4. Save Changes 를 누릅니다 .
5. OK 를 눌러 포트를 SSL 을 통한 LDAP 용 기본 포트로 변경합니다 .

청취 소켓용 보안 사용 설정

다음과 같이 서버의 청취 소켓을 보안할 수 있습니다 .

- 보안 기능 사용
- 청취 소켓용 서버 인증서 선택

- 암호 선택

보안 기능 사용

청취 소켓용으로 다른 보안 설정을 구성하기 전에 반드시 보안을 사용하도록 설정해야 합니다. 보안은 새 청취 소켓을 만들거나 기존 청취 소켓을 편집할 때 사용 설정할 수 있습니다 .

청취 소켓을 만들 때 보안 기능 사용 설정

새 청취 소켓을 만들 때 보안 기능을 사용하도록 설정하려면 다음과 같이 합니다 .

1. Server Manager 에 액세스하고 드롭 다운 목록에서 청취 소켓을 만들 서버 인스턴스를 선택합니다 .
2. Preference 탭이 표시되지 않았으면 선택하여 표시합니다 .
3. Edit Listen Sockets 링크를 선택합니다 .

   Edit Listen Sockets 페이지가 표시됩니다 .
4. New 버튼을 누릅니다 .

   Add Listen Socket 페이지가 표시됩니다 .
5. 필요한 정보를 입력하고 기본 가상 서버를 선택합니다 .
6. 보안을 사용하려면 Security 드롭 다운 목록에서 Enabled 를 선택합니다 .
7. OK 를 누릅니다 .
8. Apply 를 누른 후 Restart 를 눌러 변경 사항을 적용합니다 .

참고 청취 소켓을 만든 후 보안 설정을 구성하려면 Edit Listen Sockets 링크를 사용해야 합니다 .

청취 소켓을 편집할 때 보안 기능 사용 설정

또한 Administration Server 나 Server Manager 에서 청취 소켓을 편집할 때 보안을 사용하도록 설정할 수 있습니다 . 청취 소켓을 편집할 때 보안 기능을 사용하도록 설정하려면 다음과 같이 합니다 .

1. Administration Server 또는 Server Manager 에 액세스하고 Security 탭을 선택합니다 .

   Server Manager 의 경우 반드시 드롭 다운 목록에서 서버 인스턴스를 먼저 선택해야 합니다 .
2. Preference 탭이 표시되지 않았으면 선택하여 표시합니다 .

3. Edit Listen Sockets 링크를 선택합니다 .

   Edit Listen Sockets 페이지가 표시됩니다 .

4. 청취 소켓을 편집하려면 편집하려는 청취 소켓의 Listen Socket ID 를 누릅니다 .

   Edit Listen Socket 페이지가 표시됩니다 .

5. 해당 청취 소켓의 보안을 사용하려면 Security 드롭 다운 목록에서 Enabled 를 선택합니다 .

6. OK 를 누릅니다 .

7. Server Manager 의 경우 Apply 를 누른 다음 Restart 를 눌러 변경 사항이 적용되도록 합니다 .

청취 소켓용 서버 인증서 선택

Administration Server 나 Server Manager 에서 청취 소켓을 구성하여 요청 및 설치한 서버 인증서를 사용할 수 있습니다 .

참고 반드시 설치된 인증서가 한 개 이상 있어야 합니다 .

청취 소켓이 사용할 서버 인증서를 선택하려면 다음과 같이 합니다 .

1. Administration Server 또는 Server Manager 에 액세스하고 Preferences 탭을 선택합니다 .

   Server Manager 의 경우 반드시 드롭 다운 목록에서 서버 인스턴스를 먼저 선택해야 합니다 .

2. Edit Listen Sockets 링크를 선택합니다 .

   Edit Listen Sockets 페이지가 표시됩니다 .

3. 청취 소켓을 편집하려면 편집하려는 청취 소켓의 Listen Socket ID 를 누릅니다 .

   Edit Listen Socket 페이지가 표시됩니다 .

4. 해당 청취 소켓의 보안을 사용하려면 Security 드롭 다운 목록에서 Enabled 를 선택합니다 .

   .

참고 외부 모듈이 설치된 경우에는 계속하기 전에 Manage Server Certificates 페이지에 외부 모듈의 암호를 요구하는 메시지가 표시됩니다 .

5. Server Certificate 드롭 다운 목록에서 해당 소켓용 서버 인증서를 선택합니다 .
   이 목록에는 설치된 모든 내부 및 외부 인증서가 표시됩니다 .

참고 설치된 서버 인증서가 없는 경우 Server Certificate Name 드롭 다운 목록 대신 경고 메시지가 표시됩니다 .

6. OK 를 누릅니다 .
7. Server Manager 의 경우 Apply 를 누른 다음 Restart 를 눌러 변경 사항이 적용되도록 합니다 .

암호 선택

웹 서버의 안전을 보호하려면 SSL 을 사용하도록 설정해야 합니다 . SSL 2.0, SSL 3.0 및 TLS 암호화 프로토콜을 선택할 수 있으며 다양한 암호 제품군을 선택할 수 있습니다 . SSL 과 TLS 는 Administration Server 용 청취 소켓에서 사용하도록 설정할 수 있습니다 . Server Manager 용 청취 소켓에서 SSL 및 TLS 를 사용하면 해당 청취 소켓에 연결된 모든 가상 서버용 보안 기본 설정이 설정됩니다 .

보안되지 않은 가상 서버를 원하는 경우 , 해당 서버는 모두 보안을 사용하지 않도록 설정된 동일한 청취 소켓에 구성되어야 합니다 .

기본 설정의 경우 가장 많이 사용되는 암호를 허용합니다 . 특정 암호를 사용하면 안 되는 충분한 이유가 있지 않는 한 , 모두 허용해야 합니다 . 특정 암호에 대한 자세한 내용은 *SSL 개요*를 참조하십시오 .

참고 반드시 설치된 인증서가 한 개 이상 있어야 합니다 .

`tlsrollback` 매개 변수의 기본 및 권장 설정은 `true` 입니다 . 이렇게 구성하면 서버가 중간개입자 (man-in-the-middle) 버전 롤백 공격 시도를 감지할 수 있습니다 . TLS 표준을 잘못 구현한 일부 클라이언트와의 상호 운용성을 위하여 이 설정을 `false` 로 해야 하는 경우도 있습니다 .

`tlsrollback` 을 `false` 로 설정하면 연결이 버전 롤백 공격에 취약해진다는 점에 유의하십시오 . 버전 롤백 공격은 제 3 자가 클라이언트와 서버로 하여금 SSLv2 등의 덜 안전한 이전 프로토콜을 사용하도록 하는 메커니즘입니다 . SSLv2 에는 알려진 결함이 있으므로 버전 롤백 공격을 감지하지 못하는 경우 제 3 자가 암호화된 연결을 가로채어 해독할 수 있습니다 .

SSL 및 TLS 를 사용하려면 다음과 같이 합니다 .

1. Administration Server 또는 Server Manager 에 액세스하고 Preferences 탭을 선택합니다 .

   Server Manager 의 경우 반드시 드롭 다운 목록에서 서버 인스턴스를 먼저 선택해야 합니다 .

2. Edit Listen Sockets 링크를 누릅니다 .

   Edit Listen Sockets 페이지가 표시됩니다 . 보안 청취 소켓에 대하여 Edit Listen Socket 페이지에 사용 가능한 암호 설정이 표시됩니다 .

참고 청취 소켓에서 Security 가 사용되지 않는 경우에는 SSL 및 TLS 정보가 표시되지 않습니다. 암호를 사용하려면 선택한 청취 소켓에서 보안이 사용되도록 설정해야 합니다 . 더 자세한 내용은 " 청취 소켓용 보안 사용 설정 " 를 참조하십시오 .

3. 필요한 암호화 설정에 해당하는 선택란을 선택합니다 .

참고 Netscape Navigator 6.0 의 경우 TLS 와 SSL 3 를 모두 선택합니다 . TLS Rollback 에 대하여 TLS 를 선택하고 SSL 3 와 SSL2 가 모두 사용 안 함으로 설정되었는지 확인합니다 .

4. OK 를 누릅니다 .
5. Server Manager 의 경우 Apply 를 누른 다음 Restart 를 눌러 변경 사항이 적용되도록 합니다 .

참고 청취 소켓용 보안을 사용하도록 설정하고 변경 사항을 적용하면 `magnus.conf` 파일이 자동으로 수정되어 보안 기능이 표시되며, 해당 청취 소켓에 연결된 모든 가상 서버에 기본 보안 매개 변수가 지정됩니다 .

서버에서 SSL 을 사용 설정하면 URL 은 `http` 가 아닌 `https` 를 사용합니다 . SSL 을 사용하는 서버의 문서를 가리키는 URL 의 형식은 다음과 같습니다 .

`https://`*`servername`*`.[`*`domain`*`.[`*`dom`*`]]:[`*`port#`*`]`

예 : `https://admin. admin.sun.com:443`.

기본 보안 http 포트 번호 (443) 을 사용하는 경우 URL 에 포트 번호를 입력하지 않아도 됩니다 .

전역적 보안 구성

SSL 을 사용하는 서버를 설치하면 `magnus.conf` 파일 (서버의 기본 구성 파일) 에 전역 보안 매개 변수용 지시문 항목이 만들어집니다 . 가상 서버 보안 설정이 작동하려면 Security 가 반드시 "on" 으로 설정되어야 합니다 . 가상 서버용 SSL 등록 정보는 각 서버별로 `server.xml` 파일의 `SSLPARAMS` 요소에 있습니다 .

SSL 구성 파일 지시문의 값을 설정하려면 다음과 같이 합니다 .

1. Server Manager 에 액세스하고 드롭 다운 목록에서 해당 가상 서버의 서버 인스턴스를 선택합니다 .
2. 구성하려는 청취 소켓에서 보안을 사용하는지 확인하십시오 . 보안을 사용하도록 설정하려면 다음과 같이 합니다 .
   a. Edit Listen Sockets 링크를 누릅니다 .
   b. 보안을 사용하도록 설정하려는 청취 소켓에 해당하는 Listen Socket ID 를 누릅니다 .

      이렇게 하면 Edit Listen Socket 페이지로 이동합니다 .
   c. Security 드롭 다운 목록에서 Enabled 를 선택합니다 .
   d. OK 를 누릅니다 .
3. Magnus Editor 링크를 누릅니다 .
4. 드롭 다운 목록에서 SSL Settings 를 선택하고 Manage 를 누릅니다 .
5. 다음 항목의 값을 입력합니다 .
   - SSLSessionTimeout
   - SSLCacheEntries
   - SSL3SessionTimeout
6. OK 를 누릅니다 .
7. Apply 를 누른 후 Restart 를 눌러 변경 사항을 적용합니다 .

SSL Configuration File Directives 의 설명은 다음과 같습니다 .

SSLSessionTimeout

`SSLSessionTimeout` 은 SSL2 세션 캐시를 제어합니다 .

구문

`SSLSessionTimeout` 초

초는 캐시된 SSL 세션의 유효 시간을 초 단위로 입력합니다 . 기본값은 100 입니다 . `SSLSessionTimeout` 지시문이 지정되면 초 단위 값은 자동으로 5 에서 100 초로 제한됩니다 .

SSLCacheEntries

캐시할 수 있는 SSL 세션의 수를 지정합니다 .

SSL3SessionTimeout

SSL3SessionTimeout 지시문은 SSL3 및 TLS 세션 캐싱을 제어합니다 .

구문

`SSL3SessionTimeout` 초

초는 캐시된 SSL3 세션의 유효 시간을 초 단위로 입력합니다 . 기본값은 86400(24 시간) 입니다 . `SSL3SessionTimeout` 지지문이 지정되면 초 단위 값은 자동으로 5 에서 86400 초로 제한됩니다 .

외부 암호화 모듈 사용

Sun ONE Web Server 6.1 은 스마트 카드나 토큰 링 등의 외부 암호화 모듈을 사용하는 다음의 방법을 지원합니다 .

- PKCS#11
- FIPS-140

FIPS-140 암호화 표준을 사용하기 전에 PKCS#11 모듈을 추가해야 합니다 .

PKCS#11 모듈 설치

Sun ONE Web Server 는 PKCS(Public Key Cryptography Standard)#11 을 지원합니다 . 이는 SSL 과 PKCS#11 모듈 사이의 통신용으로 사용되는 인터페이스를 정의합니다 . PKCS#11 모듈은 SSL 하드웨어 가속기에 대한 표준 기반 연결용으로 사용됩니다 . 외부 하드웨어 가속시용으로 가져온 인증서와 키는 `secmod.db` 에 저장되며 , 이 파일은 PKCS#11 이 설치될 때 생성됩니다 .

modutil 을 사용하여 PKCS#11 모듈 설치

`modutil` 도구를 사용하여 PKCS#11 모듈을 `.jar` 파일 또는 개체 파일 형태로 설치할 수 있습니다 .

`modutil` 을 사용하여 PKCS#11 모듈을 설치하려면 다음과 같이 합니다 .

1. Administration Server 를 포함하여 모든 서버가 종료되어야 합니다 .
2. 데이터베이스가 있는 `server_root/alias` 디렉토리로 이동합니다 .
3. `server_root/bin/https/admin/bin` 을 PATH 에 추가합니다 .
4. `server_root/bin/https/admin/bin` 에서 `modutil` 을 찾습니다 .
5. 환경을 설정합니다 . 예 :
   - UNIX: `setenv`

     `LD_LIBRARY_PATH server_root/bin/https/lib:${LD_LIBRARY_PATH}`
   - IBM-AIX: `LIBPATH`
   - HP-UX: `SHLIB_PATH`
   - Windows 의 경우 이를 `PATH` 에 추가합니다 .

     `LD_LIBRARY_PATH server_root/bin/https/bin`

     아래 목록의 컴퓨터용 PATH 를 찾을 수 있습니다 . `server_root/https-admin/start`.
6. 다음 명령을 입력합니다 . `modutil`.

   옵션 목록이 표시됩니다 .
7. 필요한 조치를 수행합니다 .

   예를 들어 PCKS#11 모듈을 UNIX 에 추가하려면 다음과 같이 입력합니다 .

   `modutil -add` (the name of PCKS#11 file) `-libfile` (your `libfile` for PCKS#11) `-nocertdb -dbdir .` (사용하는 db 디렉토리)

pk12util 사용

`pk12util` 을 사용하면 내부 데이터베이스에서 인증서와 키를 내보내고 이를 내부 또는 외부 PKCS#11 모듈로 가져올 수 있습니다 . 언제라도 인증서와 키를 내부 데이터베이스로 내보낼 수 있으나 , 외부 토큰의 경우 대부분 인증서와 키를 내보낼 수 없습니다 . 기본적으로 `pk12util` 은 `cert8.db` 와 `kyes3.db` 라는 이름의 인증서 및 키 데이터베이스를 사용합니다 .

pk12util 을 사용하여 내보내기

내부 데이터베이스에서 인증서와 키를 내보내려면 다음과 같이 합니다 .

1. 데이터베이스가 있는 `server_root/alias` 디렉토리로 이동합니다 .
2. `server_root/bin/https/admin/bin` 을 PATH 에 추가합니다 .
3. `server_root/bin/https/admin/bin` 에서 `pk12util` 을 찾습니다 .
4. 환경을 설정합니다 . 예 :
   - UNIX: `setenv`

     `LD_LIBRARY_PATH/server_root/bin/https/lib:${LD_LIBRARY_PATH}`
   - IBM-AIX: `LIBPATH`
   - HP-UX: `SHLIB_PATH`
   - Windows 의 경우 이를 `PATH` 에 추가합니다 .

     `LD_LIBRARY_PATH server_root/bin/https/bin`

     아래 목록의 컴퓨터용 PATH 를 찾을 수 있습니다 .
     `server_root/https-admin/start`.
5. 다음 명령을 입력합니다 . `pk12util`.

   옵션 목록이 표시됩니다 .
6. 필요한 조치를 수행합니다 .

   예를 들어 UNIX 의 경우 다음과 같이 입력합니다 .

   ```
   pk12util -o certpk12 -n Server-Cert [-d /server/alias] [-P
   https-test-host]
   ```
7. 데이터베이스 암호를 입력합니다 .
8. `pkcs12` 암호를 입력합니다 .

pk12util 을 사용하여 가져오기

내부 또는 외부 PKCS#11 모듈로 인증서와 키를 가져오려면 다음과 같이 합니다 .

1. 데이터베이스가 있는 `server_root/alias` 디렉토리로 이동합니다 .
2. `server_root/bin/https/admin/bin` 을 PATH 에 추가합니다 .
3. `server_root/bin/https/admin/bin` 에서 `pk12util` 을 찾습니다 .
4. 환경을 설정합니다 . 예 :

- UNIX: `setenv`

  `LD_LIBRARY_PATH/server_root/bin/https/lib:${LD_LIBRARY_PATH}`
- IBM-AIX: `LIBPATH`
- HP-UX: `SHLIB_PATH`
- Windows 의 경우 이를 `PATH` 에 추가합니다 .

  `LD_LIBRARY_PATH server_root/bin/https/bin`

  아래 목록의 컴퓨터용 PATH 를 찾을 수 있습니다 .
  `server_root/https-admin/start`.

5. 다음 명령을 입력합니다 . `pk12util`.

   옵션 목록이 표시됩니다 .
6. 필요한 조치를 수행합니다 .

   예를 들어 UNIX 의 경우 다음과 같이 입력합니다 .

   ```
   pk12util -i pk12_sunspot [-d certdir][-h ?Cipher?[-P
   https-jones.redplanet.com-jones-]
   ```

   -P 는 반드시 -h 뒤에 있어야 하며 마지막 인수여야 합니다 .

   대문자와 인용 부호 사이의 공백을 포함하여 토큰 이름을 정확히 입력합니다 .
7. 데이터베이스 암호를 입력합니다 .
8. `pkcs12` 암호를 입력합니다 . 외부 인증서를 사용하여 서버 시작

서버용 인증서를 외부 PKCS#11 모듈 (하드웨어 가속기 등) 에 설치한 경우 , `server.xml`을 편집하거나 아래에 설명한 것과 같이 인증서 이름을 지정하지 않으면 서버가 인증서를 사용하여 시작할 수 없습니다 .

서버는 항상 "Server-Cert" 라는 이름의 인증서를 사용하여 시작하려고 시도합니다 . 그러나 외부 PKCS#11 모듈에 있는 인증서에는 해당 ID 에 모듈의 토큰 이름 중 하나가 포함됩니다 . 예를 들어 외부 스마트카드 판독기에 설치된 서버 인증서가 "smartcard0" 인 경우 , 이름은 "smartcard0:Server-Cert" 가 됩니다 .

외부 모듈에 설치된 인증서를 사용하여 서버를 시작하려면 서버가 실행되는 청취 소켓용 인증서 이름을 지정해야 합니다 .

청취 소켓용 인증서 이름 선택

청취 소켓용 인증서 이름을 선택하려면 다음과 같이 합니다 .

참고 청취 소켓에서 Security 가 사용되지 않는 경우에는 인증서 정보가 표시되지 않습니다 . 청취 소켓용 인증서 이름을 선택하려면 반드시 우선 청취 소켓에서 보안을 사용하도록 설정해야 합니다 . 더 자세한 내용은 " 청취 소켓용 보안 사용 설정 " 를 참조하십시오 .

1. Administration Server 또는 Server Manager 에 액세스하고 Preferences 탭을 선택합니다 .

   Server Manager 의 경우 반드시 드롭 다운 목록에서 서버 인스턴스를 먼저 선택해야 합니다 .

2. Preference 탭이 선택되지 않았으면 선택합니다 .
3. Edit Listen Sockets 링크를 누릅니다 .

   Edit Listen Sockets 페이지가 표시됩니다 .

4. 인증서와 연결하려는 청취 소켓에 해당하는 Listen Socket ID 링크를 누릅니다 .

   Edit Listen Socket 페이지가 표시됩니다 .

5. Server Certificate 드롭 다운 목록에서 해당 소켓용 서버 인증서를 선택합니다 .

   이 목록에는 설치된 모든 내부 및 외부 인증서가 표시됩니다 .

참고 설치된 서버 인증서가 없는 경우 Server Certificate Name 드롭 다운 목록 대신 경고가 표시됩니다 .

6. OK 를 누릅니다 .
7. Server Manager 의 경우 Apply 를 누른 다음 Restart 를 눌러 변경 사항이 적용되도록 합니다 .

또한 `server.xml` 파일을 직접 편집하여 서버가 해당 인증서를 사용하여 시작하도록 할 수 있습니다 . SSLPARAMS 의 `servercertnickname` 을 다음으로 변경합니다 .

```
$TOKENNAME:Server-Cert
```

$TOKENNAME 용으로 사용할 값을 찾으려면 서버의 Security 탭으로 이동하여 Manage Certificate 링크를 선택합니다 . Server-Cert 가 저장된 외부 모듈로 로그인하면 해당 인증서가 $TOKENNAME:$NICKNAME 폼의 목록에 표시됩니다 .

참고 신뢰 데이터베이스를 만들지 않은 경우에는 외부 PKCS#11 모듈에서 인증서를 요청하거나 설치하면 자동으로 만들어집니다 . 만들어진 기본 데이터베이스에는 암호가 없으며 액세스할 수 없습니다 . 외부 모듈은 작동하지만 서버 인증서를 요청하거나 설치할 수는 없습니다 . 기본 데이터베이스가 암호 없이 만들어진 경우 Create Database 의 Security 탭에서 암호를 설정합니다 .

FIPS-140 표준

PKCS#11 API 를 사용하면 암호화 작업을 수행하는 소프트웨어 또는 하드웨어 모듈과 통신할 수 있습니다 . PKCS#11 이 서버에 설치되면 Sun ONE Web Server 가 FIPS(Federal Information Processing Standards)-140 표준을 따르도록 구성할 수 있습니다 . 이 라이브러리는 오직 SSL 버전 3.0 에만 포함되어 있습니다 .

FIPS-140 을 사용하려면 다음과 같이 합니다 .

1. FIPS-140 의 설명을 따라 플러그인을 설치합니다 .
2. Administration Server 또는 Server Manager 에 액세스하고 Preferences 탭을 선택합니다 .

   Server Manager 의 경우 반드시 드롭 다운 목록에서 서버 인스턴스를 먼저 선택해야 합니다 .
3. Edit Listen Sockets 링크를 누릅니다 .

   Edit Listen Sockets 페이지가 표시됩니다 . 보안 청취 소켓에 대하여 Edit Listen Socket 페이지에 사용 가능한 보안 설정이 표시됩니다 .

참고 FIPS-140 을 사용하려면 선택한 청취 소켓에서 보안이 사용되도록 설정해야 합니다 . 더 자세한 내용은 " 청취 소켓용 보안 사용 설정 " 을 참조하십시오 .

4. Enable 이 선택되어 있지 않으면 SSL Version 3 드롭 다운 목록에서 선택합니다 .
5. 적절한 FIPS-140 암호 제품군을 선택합니다 .
   - 56 비트 암호화와 SHA 메시지 인증이 포함된 (FIPS)DES.
   - 168 비트 암호화와 SHA 메시지 인증이 포함된 (FIPS)Triple DES

6. OK 를 누릅니다 .
7. Server Manager 의 경우 Apply 를 누른 다음 Restart 를 눌러 변경 사항이 적용되도록 합니다 .

클라이언트 보안 요구 사항 설정

서버 보안 단계를 모두 수행 한 후, 클라이언트에 대한 추가 보안 요구 사항을 설정할 수 있습니다 .

클라이언트 인증 필수화

Administration Server 용 청취 소켓을 사용하도록 설정하고 각 서버 인스턴스가 클라이언트 인증을 요청하도록 할 수 있습니다 . 클라이언트 인증을 사용하면 서버가 쿼리에 대한 응답을 보내기 전에 클라이언트에 인증서를 요구합니다 .

Sun ONE Web Server 는 클라이언트 인증서에 있는 CA 와 클라이언트 인증서 서명용으로 신뢰된 CA 를 비교하여 클라이언트 인증서를 인증합니다 . 클라이언트 인증서 서명용 CA 의 목록은 Administrator Server 의 Security 에 있는 Manage Certificates 페이지에서 확인할 수 있습니다 . CA 의 유형은 네 가지입니다 .

- Untrusted CA (비교되지 않음)
- Trusted Server CA (비교되지 않음)
- Trusted Client CA (비교됨)
- Trusted Client/Server CA (비교됨)

웹 서버가 신뢰된 CA 의 인증서를 보유하지 않은 클라이언트를 거부하도록 구성할 수 있습니다 . CA 를 승인 또는 거부하려면 반드시 CA 용 클라이언트 신뢰를 설정해야 합니다 . 더 자세한 내용은 " 인증서 관리 " 페이지 119 를 참조하십시오 .

Sun ONE Web Server 는 인증서의 유효 기간이 만료된 경우 오류를 기록하고 인증서를 거부하며 클라이언트에게 메시지를 반송합니다 . 또한 Administration Servers Manage Certificates 페이지에서 만기된 인증서를 확인할 수 있습니다 .

서버가 인증서 클라이언트에서 정보를 수집하여 이를 LDAP 디렉토리에 있는 사용자 항목과 비교하도록 구성할 수 있습니다 . 이렇게 하면 클라이언트의 인증서가 유효하며 LDAP 디렉토리에 항목이 보관되도록 합니다 . 또한 클라이언트 인증서가 LDAP 디렉토리의 항목 중 하나와 일치되도록 합니다 . 이에 대한 방법은 " 클라이언트 인증서를 LDAP 로 매핑 " 페이지 138 을 참조하십시오 .

클라이언트 인증서를 액세스 제어와 조합할 수 있으므로 신뢰된 CA 의 요구 사항 이외에 인증서에 연결된 사용자는 반드시 액세스 제어 규칙 (ACL) 과 일치되어야 합니다 . 더 자세한 내용은 " 액세스 제어 파일 사용 " 페이지 184 를 참조하십시오 .

또한 클라이언트 인증서의 정보를 처리할 수 있습니다 . 자세한 내용은 Sun ONE Web Server 6.1 *NSAPI Programmer's Guide* 를 참조하십시오 .

클라이언트 인증 요구

클라이언트 인증을 요구하려면 다음과 같이 합니다 .

1. Administration Server 또는 Server Manager 에 액세스하고 Preferences 탭을 선택합니다 .

   Server Manager 의 경우 반드시 드롭 다운 목록에서 서버 인스턴스를 먼저 선택해야 합니다 .

2. Edit Listen Sockets 링크를 누릅니다 .

   Edit Listen Sockets 페이지가 표시됩니다 .

3. 클라이언트 인증을 요구할 청취 소켓에 해당하는 Listen Socket ID 링크를 누릅니다 .

   Edit Listen Socket 페이지가 표시됩니다 .

4. 해당 청취 소켓용 클라이언트 인증을 요구하려면 Client Authentication 드롭 다운 목록에서 Required 를 선택합니다 .

5. OK 를 누릅니다 .

6. Server Manager 의 경우 Apply 를 누른 다음 Restart 를 눌러 변경 사항이 적용되도록 합니다 .

참고 현재 각 웹 서버 인스턴스에는 하나의 인증서 신뢰 데이터베이스만 존재합니다 . 해당 서버 인스턴스에 의하여 실행되는 모든 보안 가상 서버는 동일한 목록의 신뢰 인증서 CA 를 공유합니다 . 두 개의 가상 서버에 서로 다른 인증 CA 가 필요한 경우에는 해당 가상 서버는 별도의 신뢰 데이터베이스가 있는 서로 다른 서버 인스턴스에서 실행되어야 합니다 .

클라이언트 인증서를 LDAP 로 매핑

이 부분에서는 Sun ONE Web Server 가 클라이언트 인증서를 LDAP 디렉토리의 항목과 매핑하는 데 사용하는 프로세스에 대하여 설명합니다 .

서버에 클라이언트의 요청이 수신되면 이를 처리하기 전에 클라이언트의 인증서를 요구합니다 . 클라이언트에 따라 서버에 요청과 함께 클라이언트를 전송하는 경우도 있습니다 .

참고 또한 , 클라이언트 인증서를 LDAP 로 매핑하려면 필요한 ACL 을 설정해야 합니다 . 자세한 내용은 제 9 장 , " 서버 액세스 제어 " 를 참조하십시오 .

서버는 CA 를 Administration Server 에 있는 신뢰 CA 의 목록과 비교하려고 합니다 . 일치 항목이 없으면 Sun ONE Web Server 가 연결을 종료합니다 . 일치 항목이 있으면 서버가 요청 처리를 계속합니다 .

신뢰 CA 에서 인증서를 확인한 후 , 서버는 다음과 같이 인증서를 LDAP 항목과 매핑합니다 .

- 클라이언트 인증서에 있는 발행자와 대상 DN 을 LDAP 디렉토리의 분기점과 매핑합니다 .
- LDAP 디렉토리에 클라이언트 인증서의 대상 (최종 사용자) 에 대한 정보와 일치하는 항목이 있는지 검색합니다 .
- (선택) 클라이언트 인증서를 DN 에 해당하는 LDAP 항목 중 하나와 확인합니다 .

서버는 `certmap.conf` 라는 인증서 매핑 파일을 사용하여 LDAP 검색 방법을 결정합니다 . 서버는 매핑 파일에 따라 클라이언트 인증서에서 가져올 값 (최종 사용자의 이름 , 전자우편 주소 등) 을 결정합니다 . 서버는 이들 값을 사용하여 LDAP 디렉토리에서 사용자 항목을 검색하지만 , 우선 서버가 LDAP 디렉토리에서 검색을 시작할 위치를 결정해야 합니다 . 서버는 또한 인증서 매핑 파일에서 시작 위치를 알 수 있습니다 .

서버가 검색을 시작할 위치와 검색할 항목을 결정하면 (제 1 단계) LDAP 디렉토리에서 검색을 수행합니다 (제 2 단계). 일치 항목이 없거나 일치 항목이 여러 개인 경우 매핑이 인증서 확인으로 설정되지 *않고* 검색은 실패합니다 . 예상되는 검색 결과의 전체 목록은 아래의 표 6-1 을 참조하십시오 . 참고로 ACL 에서 예상되는 작동을 지정할 수 있습니다 . 예를 들어 , Sun ONE Web Server 가 오직 인증서 일치에 실패하는 경우에만 승인하도록 지정할 수 있습니다 . ACL 기본 설정에 대한 자세한 내용은 " 액세스 제어 파일 사용 " 페이지 184 을 참조하십시오 .

표 6-1) LDAP 검색 결과

LDAP 검색 결과	인증서 검증 ON	인증서 검증 OFF
검색된 항목 없음	인증 실패	인증 실패
정확히 한 개 항목 일치	인증 실패	인증 성공
여러 항목 일치	인증 실패	인증 실패

서버가 LDAP 디렉토리에서 일치 항목과 인증서를 찾으면 해당 정보를 사용하여 트랜잭션을 처리할 수 있습니다 . 예를 들어 서버에 따라 인증서 -LDAP 매핑을 사용하여 서버에 대한 액세스를 결정합니다 .

certmap.conf 파일 사용

인증서 매핑에 따라 서버가 LDAP 디렉토리에서 사용자 항목을 찾는 방법이 결정됩니다 . `certmap.conf` 를 사용하여 이름으로 명시된 인증서를 LDAP 항목과 일치시키는 방법을 구성할 수 있습니다 . 이 파이를 편집하고 항목을 추가하여 LDAP 디렉토리의 조직을 검색하고 사용자에게 부여할 인증서 목록을 표시할 수 있습니다 . 사용자는 사용자 ID, 전자우편 또는 `subjectDN` 에서 사용되는 다른 값을 기준으로 인증될 수 있습니다 . 특히 , 매핑 파일에는 다음의 정보가 정의됩니다 .

- 서버가 검색을 시작하는 LDAP 트리 안의 위치
- LDAP 디렉토리에서 항목을 검색할 때 서버가 검색 범주로 사용할 인증서 속성
- 서버가 추가의 검증 과정을 수행할 것인지의 여부

인증서 매핑 파일은 다음에 있습니다 .

server_root`/userdb/certmap.conf`

파일에는 하나 이상의 이름 매핑이 있으며 , 각각의 매핑은 서로 다른 CA 에 적용됩니다 . 매핑의 구문은 다음과 같습니다 .

```
certmap <name> <issuerDN>

<name>:<property> [<value>]
```

첫 번째 줄은 항목의 이름과 CA 인증서에 있는 고유 이름을 구성하는 속성을 지정합니다 . name 은 임의로 원하는 이름을 정의할 수 있습니다 . 그러나 `issuerDN` 은 반드시 클라이언트 인증서를 발행한 CA 의 발행자 DN 과 *정확히* 일치해야 합니다 . 예를 들어 아래의 `issuerDN` 줄의 차이는 단지 속성을 구분하는 공백이지만 서버는 이 두 항목을 서로 다른 것으로 처리합니다 .

```
certmap sun1 ou=Sun Certificate Authority,o=Sun, c=US
certmap sun2 ou=Sun Certificate Authority,o=Sun, c=US
```

팁 Sun ONE Directory Server 를 사용하며 `issuerDN` 을 검색하는데 문제가 발생하는 경우에는 Directory Server 오류 로그에 유용한 정보가 있는지 확인하십시오 .

이름 매핑의 두 번째 및 이후 줄은 등록 정보를 값과 매핑합니다 . `certmap.conf` 파일에는 여섯 개의 등록 정보가 있습니다 . (인증서 API 를 사용하여 등록 정보를 사용자 정의할 수 있습니다 .)

- `DNComps` 는 쉼표로 분리된 속성 목록으로 , LDAP 디렉토리에서 사용자 정보 (즉 , 클라이언트 인증서의 소유자) 와 일치하는 항목 검색을 시작할 위치를 결정하는데 사용합니다 . 서버는 클라이언트 인증서에서 이들 속성 값을 수집하고 값을 사용하여 LDAP DN 을 구성합니다 . 그런 후 LDAP 디렉토리에서 서버가 검색을 시작할 위치를 결정합니다 . 예를 들어 , `DNComps` 가 DN 의 `o` 와 `c` 속성을 사용하도록 설정하면 서버는 LDAP 디렉토리에서 속성이 `o=<org>, c=<country>` 인 항목부터 검색을 시작합니다. 여기에서 `<org>` 와 `<country>` 는 인증서의 DN 에 있는 값으로 대체됩니다 .

  다음 상황에 유의하십시오 .

  - 매핑에 `DNComps` 항목이 없는 경우에는 서버는 `CmapLdapAttr` 설정을 사용하거나 클라이언트 인증서에 있는 전체 대상 DN(즉 , 최종 사용자의 정보) 을 사용합니다 .
  - `DNComps` 항목은 있으나 값이 없는 경우 , 서버는 전체 LDAP 트리에서 필터와 일치하는 항목을 검색합니다 .
- `FilterComps` 는 쉼표로 분리된 속성 목록으로 클라이언트 인증서에 있는 사용자의 DN 에서 정보를 수집하여 필터를 만드는 데 사용합니다 . 서버는 이들 속성용 값을 사용하여 LDAP 디렉토리에서 항목을 비교하는데 사용할 검색 범주를 구성합니다 . LDAP 에서 인증서에서 수집한 사용자의 정보와 일치하는 항목이 하나 이상 검색되는 경우 검색은 성공적이며 서버는 선택적으로 검증을 수행합니다 .

예를 들어 `FilterComps` 가 전자우편과 사용자 ID 속성을 사용하도록 설정하는 경우 (`FilterComps=e, uid`), 서버는 디렉토리에서 전자우편과 사용자 ID 값이 클라이언트 인증서에서 수집한 사용자 정보와 일치하는 항목을 검색합니다 . 전자우편 주소와 사용자 ID 는 보통 디렉토리에서 고유한 항목이므로 좋은 필터입니다 . LDAP 데이터베이스에서 오직 하나의 항목만 검색하려면 필터가 구체적이어야 합니다 .

x509v3 인증서 속성 목록은 다음 표를 참조하십시오 .

표 6-2) x509v3 인증서용 속성

속성	설명
`c`	국가
`o`	단체
`cn`	공통 이름
`l`	장 소
`st`	주
`ou`	조직 단위
`uid`	UNIX/Linux 사용자 ID
`email`	전자우편 주소

필터용 속성 이름은 LDAP 디렉토리가 아닌 인증서의 속성 이름이어야 합니다 . 예를 들어 일부 인증서에는 사용자의 전자우편 주소용 속성으로 `e` 가 있는 반면 LDAP 에서 이 속성의 이름은 `mail` 입니다 .

- 서버는 `verifycert` 의 설정에 따라 클라이언트의 인증서를 LDAP 디렉토리에서 검색된 인증서와 비교할 것인지 결정합니다 . 값은 ON 이거나 OFF 입니다 . 이 등록 정보는 LDAP 디렉토리에 인증서가 있는 경우에만 사용해야 합니다 . 기기능은 최종 사용자의 인증서가 유효하며 취소되지 않았는지 확인하는 데 유용합니다 .
- `CmapLdapAttr` 은 LDAP 디렉토리에 있는 속성 이름으로 사용자에게 속한 모든 인증서의 대상 DN 을 포함합니다 . 이 등록 정보의 기본값은 `certSubjectDN` 입니다 . 이 등록 정보는 표준 LDAP 속성이 아니므로 이 등록 정보를 사용하려면 반드시 LDAP 스키마를 확장해야 합니다 . 자세한 내용은 *SSL 개요*를 참조하십시오 .

certmap.conf 에 이 등록 정보가 있으면 서버는 전체 LDAP 디렉토리에서 속성이 대상의 전체 DN(인증서에서 가져온 DN) 과 일치하는 항목을 검색합니다 . 해당 항목이 검색되지 않는 경우 서버는 DNComps 와 FilterComps 매핑을 사용하여 다시 검색합니다 .

인증서를 LDAP 항목과 일치시키는 이 접근 방법은 DNComps 와 FilterComps 를 사용하여 항목을 일치시키는 것이 어려울 때 유용합니다 .

- Library 는 값이 공유 라이브러리 또는 DLL 에 대한 경로 이름인 등록 정보입니다 . 이 등록 정보는 인증서 API 를 사용하여 자체의 등록 정보를 만든 경우에만 사용합니다 . 더 자세한 내용은 *NSAPI Programmer's Guide* 를 참조하십시오 .
- InitFn 은 값이 사용자 정의 라이브러리에 있는 init 함수의 이름인 등록 정보입니다 . 이 등록 정보는 인증서 API 를 사용하여 자체의 등록 정보를 만든 경우에만 사용합니다 .

이 등록 정보에 대한 자세한 내용은 " 매핑 예제 " 페이지 142 에서 설명한 예제를 참조하십시오 .

사용자 정의 등록 정보 생성

클라이언트 인증서 API 를 사용하여 자체의 등록 정보를 만들 수 있습니다 . 클라이언트 인증서 API 프로그래밍과 사용 방법은 *NSAPI Programmer's Guide* 를 참조하십시오 .

사용자 정의 매핑이 있는 경우 매핑은 다음과 같이 참조합니다 .

```
<name>:library <path_to_shared_library>
<name>:InitFn <name_of_init_function>
```

예 :

```
certmap default1 o=Sun Microsystems, c=US
default1:library /usr/sun/userdb/plugin.so
default1:InitFn plugin_init_fn
default1:DNComps  ou o c
default1:FilterComps l
default1:verifycert on
```

매핑 예제

certmap.conf 파일에는 최소한 한 개 이상의 항목이 있어야 합니다. certmap.conf 파일을 사용하는 다양한 방법은 다음 예제에 보이는 것과 같습니다 .

예제 1

이 예제의 certmap.conf 파일에는 오직 한 개의 " 기본 " 매핑만 있습니다 .

```
certmap default default
default:DNComps ou, o, c
default:FilterComps e, uid
default:verifycert on
```

이 예제를 사용하면 서버는 `ou=<orgunit>, o=<org>, c=<country>` 항목을 포함하는 LDAP 분기점에서 검색을 시작하며 , 여기에서 `<>` 의 텍스트는 클라이언트 인증서에 있는 대상 DN 의 값으로 대체됩니다 .

그 후 , 서버는 인증서에 있는 전자우편 주소와 사용자 ID 값을 사용하여 LDAP 디렉토리에 일치하는 항목이 있는지 검색합니다 . 항목이 검색되면 서버는 클라이언트가 전송한 인증서와 디렉토리에 있는 인증서를 비교하여 인증서를 검증합니다 .

예제 2

다음 예제 파일에는 기본값과 US 우편 서비스용의 두 가지 매핑이 있습니다 .

```
certmap default default
default:DNComps
default:FilterComps e, uid

certmap usps ou=United States Postal Service, o=usps, c=US
usps:DNComps ou,o,c
usps:FilterComps e
usps:verifycert on
```

서버에 미국 우편 서비스가 아닌 다른 인증서가 수신되면 기본 매핑을 사용합니다 . 이 경우 LDAP 트리의 상단에서 시작하여 클라이언트의 전자우편 및 사용자 ID 와 일치하는 항목을 검색합니다 . 미국 우편 서비스의 인증서인 경우 서버는 조직 단위를 포함하는 LDAP 분기에서 검색을 시작하며 일치하는 전자우편 주소를 검색합니다 . 또한 인증서가 USPS 의 것이면 서버가 인증서를 검증하지만 기타 인증서는 검증되지 않습니다 .

주의 인증서의 발행자 DN(즉 , CA 정보) 은 반드시 매핑의 첫 번째 줄 목록에 있는 발행자 DN 과 동일해야 합니다 . 앞의 예제에서 `o=United States Postal Service,c=US`인 발행자 DN의 인증서는 `o`와 `c` 속성 사이에 공백이 없으므로 일치되지 않습니다 .

예제 3

다음 예제에서는 `CmapLdapAttr` 등록 정보를 사용하여 LDAP 데이터베이스에서 `certSubjectDN` 이라고 하는 속성을 검색합니다 . 이 속성의 값은 클라이언트 인증서에서 가져온 대상 DN 전체와 정확히 일치합니다 .

```
certmap myco ou=My Company Inc, o=myco, c=US
myco:CmapLdapAttr certSubjectDN
myco:DNComps  o, c
myco:FilterComps mail, uid
myco:verifycert on
```

클라이언트 인증서 대상이 다음인 경우 ,

```
uid=Walt Whitman, o=LeavesOfGrass Inc, c=US
```

서버는 우선 다음 정보를 포함한 항목을 검색합니다 .

```
certSubjectDN=uid=Walt Whitman, o=LeavesOfGrass Inc, c=US
```

일치하는 항목이 하나 이상인 경우에는 서버가 항목을 검증합니다 . 검색된 항목이 없는 경우 서버는 `DNComps` 와 `FilterComps` 를 사용하여 일치하는 항목을 검색합니다 . 이 예제에서 서버는 `o=LeavesOfGrass Inc, c=US` 아래의 모든 항목에서 `uid=Walt Whitman` 을 검색합니다 .

참고 이 예제에서는 LDAP 디렉토리에 `certSubjectDN` 속성이 있는 항목이 포함된 것으로 가정합니다 .

고급 보안 설정

Stronger Ciphers 옵션에서는 액세스용으로 168, 128 또는 56 비트 비밀 키를 선택하거나 제한을 설정하지 않을 수 있습니다 . 제한에 맞지 않는 경우 서비스될 파일을 지정할 수 있습니다 . 파일을 지정하지 않으면 Sun ONE Web Server 는 "Forbidden" 상태를 반환합니다 .

선택한 액세스용 키 크기가 Security Preferences 의 현재 암호 설정과 맞지 않는 경우 Sun ONE Web Server 에 더 큰 비밀 키로 암호를 사용해야 한다는 경고 대화 상자가 표시됩니다 .

현재 키 크기 제한은 Service `fn=key-toosmall` 이 아닌 `obj.conf` 의 NSAPI `PathCheck` 지시문을 기준으로 구현됩니다 . 지시문 :

```
PathCheck fn="ssl-check" [secret-keysize=<nbits>]
[bong-file=<filename>]
```

여기에서 `<nbits>` 는 비밀 키에 필요한 최소 비트 수이며 `<filename>` 은 제한을 만족하지 않는 경우 서비스될 파일의 이름 (URL 아님) 입니다 .

SSL 을 사용하지 않거나 `secret-keysize` 매개 변수가 지정되지 않은 경우 `PathCheck` 은 `REQ_NOACTION` 을 반환합니다 . 현재 세션의 비밀 키 크기가 지정된 `secret-keysize` 보다 작은 경우 , `bong-file` 이 지정되지 않으면 이 함수는 `REQ_ABORTED` 를 `PROTOCOL_FORBIDDEN` 상태와 함께 반환합니다 . 그렇지 않은 경우 `REQ_PROCEED` 를 반환하며 "path" 변수가 `bong-file <filename>` 으로 설정됩니다 . 또한 키 크기 제한을 만족하지 않은 경우 현재 세션용 SSL 세션 캐시가 무효화되어 다음 번 클라이언트가 서버로 연결하면 전체 SSL 핸드셰이크가 발생합니다 .

참고 Stronger Ciphers 폼을 사용하면 `PathCheck fn=ssl-check` 을 추가할 때 개체에서 검색되는 Service `fn=key-toosmall` 지시문이 제거됩니다 .

고급 암호를 설정하려면 다음과 같이 합니다 .

1. Server Manager에 액세스하고 드롭 다운 목록에서 서버 인스턴스를 선택합니다.
2. Virtual Server Class 탭을 누릅니다 .
3. 드롭 다운 목록에서 클래스 폼을 선택하고 Manage 를 누릅니다 .

   Class Manager 페이지가 표시됩니다 .
4. Content Mgmt 탭을 선택합니다 .
5. Stronger Ciphers 를 선택합니다 .
6. 다음 편집 옵션 중 한 가지를 선택합니다 .
   - 드롭 다운 목록
   - Browse 누름
   - Wildcard 누름
7. 비밀 키 크기 제한을 선택합니다 .
   - 168 비트 이상
   - 128 비트 이상
   - 56 비트 이상
   - 제한 없음
8. 액세스를 거부할 메시지의 파일 위치를 입력합니다 .
9. OK 를 누릅니다 .

10. Apply 를 누릅니다 .

11. hard start/restart 또는 dynamically apply 를 선택합니다 .

자세한 내용은 *SSL 개요*를 참조하십시오 .

추가 보안 고려 사항

누군가 암호를 해독하려는 시도 외에도 다른 보안 위험이 있습니다 . 네트워크는 서버와 서버의 정보에 액세스하려는 다양한 방법을 사용하는 외부 및 내부 해커의 위험에 직면해 있습니다 .

따라서 서버의 암호화를 사용하는 것 이외에 추가의 보안 조치를 취해야 합니다 . 예를 들어 서버 컴퓨터를 안전한 곳에 위치시키거나 신뢰되지 않은 개인이 서버에 프로그램을 올리지 못하도록 해야 합니다 .

다음에서는 서버를 더욱 안전하게 만드는데 가장 중요한 사항들에 대하여 설명할 것입니다 .

- 실제 액세스 제한
- 관리 액세스 제한
- 강력한 암호 선택
- 암호 또는 PIN 변경
- 서버에서 기타 응용 프로그램 제한
- 클라이언트가 SSL 파일을 캐시하지 못하도록 방지
- 포트 제한
- 서버의 한계 파악
- 서버 보호를 위한 추가 변경 적용

실제 액세스 제한

이 간단한 보안 수단이 종종 잊혀지고 있습니다 . 서버 컴퓨터를 잠금 장치가 있는 곳에 위치시켜 권한 있는 사람만 들어갈 수 있도록 합니다 . 이렇게 하면 서버 컴퓨터 자체를 해킹할 수 없도록 방지합니다 .

또한 있는 경우 컴퓨터의 관리 (루트) 암호를 보호하십시오 .

관리 액세스 제한

원격 구성을 사용하는 경우 오직 몇몇의 사용자와 컴퓨터에만 관리를 허용하도록 액세스 제어를 설정해야 합니다 . Administrator Server 가 LDAP 사용자에게 LDAP 서버나 로컬 디렉토리 정보에 대한 액세스를 부여하도록 하는 경우 , SSL 을 사용하는 Administration Server 가 마스터 서버의 역할을 하고 사용자가 기타 Administration Server 에 액세스할 수 있도록 두 대의 Administration Server 를 유지 및 클러스터 관리 사용을 고려하십시오 .

클러스터에 대한 자세한 내용은 " 클러스터 설명 " 페이지 153 를 참조하십시오 .

또한 Administration Server 용 암호화를 사용해야 합니다 . 관리용 SSL 연결을 사용하지 않는 경우에는 보안되지 않은 네트워크를 통하여 원격 서버 관리를 수행할 때 조심해야 합니다 . 누구라도 관리 암호를 가로채어 서버를 재구성할 수 있습니다 .

강력한 암호 선택

서버에는 관리 암호 , 개인 키 암호 , 데이터베이스 암호 등 여러 가지 암호를 사용합니다 . 관리 암호는 누구라도 컴퓨터에 있는 모든 서버를 구성할 수 있으므로 가장 중요한 암호입니다 . 개인 키 암호는 다음으로 중요한 암호입니다 . 개인 키와 개인 키 암호가 있으면 사용자의 것으로 보이는 가짜 서버를 만들거나 서버로 오고 가는 통신을 가로채고 변경할 수 있습니다 .

좋은 암호는 자신은 기억할 수 있지만 다른 사람은 추측할 수 없는 암호입니다 . 예를 들어 *Mci12!mo* 를 "My Child is 12 months old!" 라고 기억할 수 있을 것입니다 . 자녀의 이름이나 생일은 나쁜 암호입니다 .

해독하기 어려운 암호

강력한 암호를 만드는 데 도움이 되는 간단한 지침이 몇 가지 있습니다 .

하나의 암호에 다음의 규칙을 모두 적용해야 하는 것은 아니지만 더 많은 규칙을 적용할 수록 암호가 더욱 알아내기 어려워질 것입니다 .

- 암호의 길이는 6-14 문자여야 합니다 . (Mac 암호의 길이는 8 자 이상일 수 없습니다 .)
- *, " 또는 공백 등 " 부적절한 " 문자를 사용하면 안 됩니다 .
- 사전의 단어를 사용하면 안 됩니다 . (모든 언어)
- E 를 3 으로 , L 을 1 로 하는 등의 일반적인 문자 대체를 사용하지 마십시오 .

- 다음 종류의 문자를 가능한 한 많이 포함시킵니다 .
  - 대문자
  - 소문자
  - 숫자
  - 기호

암호 또는 PIN 변경

주기적으로 신뢰 데이터베이스 / 키쌍 파일 암호 또는 PIN 을 변경하는 것이 좋습니다 . Administration Server 에서 SSL 을 사용하는 경우 서버를 시작할 때 이 암호가 필요합니다 . 주기적으로 암호를 변경하면 서버 보호를 한 단계 높일 수 있습니다 .

이 암호는 오직 로컬 컴퓨터에서 변경해야 합니다 . 암호를 변경하는 경우 고려해야 할 지침은 " 해독하기 어려운 암호 " 페이지 147 를 참조하십시오 .

비밀번호 변경

Administration Server 또는 서버 인스턴스용 신뢰 데이터베이스 / 키쌍 파일 암호를 변경하려면 다음과 같이 합니다 .

1. Administration Server 또는 Server Manager 에 액세스합니다 .

   Server Manager 의 경우 반드시 드롭 다운 목록에서 서버 인스턴스를 먼저 선택해야 합니다 .

2. Change Password 링크를 선택합니다 .
3. 드롭 다운 목록에서 암호를 변경하려는 보안 토큰을 선택합니다 .

   기본으로 이는 내부 키 데이터베이스용 " 내부 " 암호입니다 . PKCS#11 모듈이 설치된 경우 모든 토큰 목록이 표시됩니다 . Change Password 링크를 누릅니다 .
4. 현재 암호를 입력합니다 .
5. 새 암호를 입력합니다 .
6. 새 암호를 다시 입력합니다 .
7. OK 를 누릅니다 .
8. Server Manager 의 경우 Apply 를 누른 다음 Restart 를 눌러 변경 사항이 적용되도록 합니다 .

키쌍 파일이 보호되는지 확인하십시오 . Administration Server 는 키 쌍 파일을 *server_root*/alias 디렉토리에 저장합니다 . 오직 컴퓨터에 설치된 Sun ONE 서버만이 해당 파일과 디렉토리를 읽을 수 있도록 하는 경우를 생각할 수 있습니다 .

또한 파일이 백업 테이프에 저장되어 있는지 또는 기타 다른 사람이 가로챌 가능성이 있는지 확인하는 것이 중요합니다 . 이 경우 백업을 서버와 마찬가지로 완벽하게 보호해야 합니다 .

서버에서 기타 응용 프로그램 제한

서버와 동일한 컴퓨터에서 실행되는 모든 응용 프로그램을 신중하게 고려해야 합니다 . 서버에서 실행되는 다른 프로그램의 취약점을 활용하여 서버의 보안을 우회하는 것이 가능합니다 . 필요하지 않은 모든 프로그램과 서비스를 종료합니다 . 예를 들어 UNIX sendmail 데몬은 안전하게 구성하는 것이 어려우며 서버 컴퓨터에서 해로운 프로그램을 실행하도록 프로그램될 수 있습니다 .

UNIX 및 Linux

inittab 및 rc 스크립트에서 시작하는 프로세스를 신중하게 선택합니다 . 서버 컴퓨터에서 telnet 또는 rlogin 을 실행하면 안 됩니다 . 또한 서버 컴퓨터에 rdist 가 있으면 안 됩니다 . (이는 파일을 배포할 수 있으나 또한 서버 컴퓨터의 파일을 업데이트하도록 사용될 수 있습니다 .)

Windows

다른 컴퓨터와 공유할 드라이브 및 디렉토리를 신중히 고려합니다 . 또한 계정이나 Guest 권한을 부여할 사용자를 고려합니다 .

마찬가지로 서버에 들어갈 프로그램과 다른 사람이 서버에 설치할 프로그램을 신중히 고려해야 합니다 . 다른 사람의 프로그램에는 보안 취약점이 있을 수 있습니다 . 또한 특히 보안을 손상시키도록 디자인된 악의적 프로그램을 업로드할 수도 있습니다 . 서버에서 프로그램을 허용하기 전에 신중하게 프로그램을 평가해야 합니다 .

클라이언트가 SSL 파일을 캐시하지 못하도록 방지

HTML 에 있는 파일의 <HEAD> 부분에 다음 줄을 추가하여 클라이언트가 미리 암호화된 파일을 캐시할 수 없도록 방지할 수 있습니다 .

```
<meta http-equiv="pragma" content="no-cache">
```

포트 제한

컴퓨터에서 사용되지 않는 포트는 모두 비활성화합니다 . 라우터 또는 방화벽 구성을 사용하여 절대적으로 최소 포트 세트 이외의 것으로 향하는 입중계 연결을 방지합니다 . 이렇게 하면 실제로 이미 제한된 영역에 위치한 서버의 컴퓨터를 사용할 때에만 컴퓨터에서 셸을 사용할 수 있게 됩니다 .

서버의 한계 파악

서버는 서버와 클라이언트 사이에 안전한 연결을 제공합니다 . 클라이언트가 일단 정보를 보유하면 정보의 보안을 제어할 수 없으며 서버 컴퓨터 자체와 해당 디렉토리 및 파일에 대한 액세스는 제어할 수 없습니다 .

이러한 제한을 알고 있으면 피해야 할 상황을 이해하는 데 도움이 됩니다 . 예를 들어 SSL 연결을 통하여 신용 카드 번호를 구할 수 있으나 이 번호가 서버 컴퓨터의 보안 파일에 저장되어 있을까요 ? SSL 연결이 종료된 후 이 번호는 어떻게 될까요 ? 클라이언트가 SSL 을 통하여 송신하는 모든 정보를 보안할 책임이 있습니다 .

서버 보호를 위한 추가 변경 적용

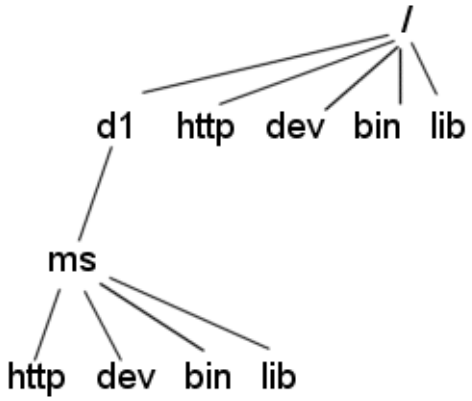
보호된 서버와 보호되지 않은 서버를 모두 사용하려면 보호되지 않은 서버를 보호된 서버와 다른 컴퓨터에서 운용해야 합니다 . 자원의 한계로 인하여 보호되지 않은 서버를 보호된 서버와 동일한 컴퓨터에서 실행해야 하는 경우에는 다음과 같이 합니다 .

- 적절한 포트 번호를 지정합니다 . 보호된 서버와 보호되지 않은 서버에 반드시 서로 다른 포트 번호가 지정되도록 해야 합니다 . 등록된 기본 포트 번호 :
  - 443 - 보안된 서버용
  - 80 - 보안되지 않은 서버용
- UNIX 또는 Linux 의 경우 문서 루트 디렉토리에 대하여 `chroot` 기능을 사용합니다 . 보호되지 않은 서버는 `chroot` 를 사용하여 재지정한 문서 루트를 참조하게 됩니다 .

chroot 를 사용하면 서버를 특정 디렉토리로 제한하는 제 2 의 루트 디렉토리를 만들 수 있습니다 . 이 기능은 보호되지 않은 서버에 대한 안전 장치로 사용할 수 있다 . 예를 들어 루트 디렉토리가 /dl/ms 로 만들 수 있습니다 . 그런 후 , 웹 서버가 루트 디렉토리에 액세스하면 실제로 dl/ms 로 액세스하게 됩니다 . 예를 들어 /dev 에 액세스하려고 시도하면 /dl/ms/dev 에 액세스하게 됩니다 . 이렇게 하면 실제 루트 디렉토리의 모든 파일에 대한 액세스를 허용하지 않고 UNIX/Linux 시스템에서 웹 서버를 실행할 수 있습니다 .

그러나 chroot 를 사용하는 경우 , 아래의 그림에 보이는 것과 같이 대체 루트 디렉토리에 Sun ONE Web Server 가 요구하는 전체 디렉토리 구조를 만들어야 합니다 .

chroot 디렉토리 구조 예제



가상 서버 클래스용 chroot 지정

다음과 같이 가상 서버 클래스용 chroot 디렉토리를 지정할 수 있습니다 .

1. Server Manager에 액세스하고 드롭 다운 목록에서 서버 인스턴스를 선택합니다.
2. Virtual Server Class 탭을 선택합니다 .
3. Edit Classes 링크를 누릅니다 .
4. chroot 를 지정하려는 클래스용 Option 이 Edit 로 설정되었는지 확인합니다 .
5. 해당 클래스의 Advanced 버튼을 누릅니다 .

   Virtual Servers CGI Settings 페이지가 표시됩니다 .
6. Chroot 필드에 전체 경로를 입력합니다 .
7. OK 를 누릅니다 .

8. Apply 를 누릅니다 .
9. Load Configuration Files 를 선택하여 동적으로 적용합니다 .

가상 서버용 chroot 지정

다음과 같이 가상 서버용 `chroot` 디렉토리를 지정할 수 있습니다 .

1. Server Manager에 액세스하고 드롭 다운 목록에서 서버 인스턴스를 선택합니다.
2. Virtual Server Class 탭을 선택합니다 .
3. Server 의 Tree View 에서 `chroot` 디렉토리를 지정하려는 가상 서버용 링크를 누릅니다 .
4. Settings 탭을 선택합니다 .

   Settings 페이지가 표시됩니다 .
5. Chroot Directory 옆의 Set to 필드에 전체 경로를 입력합니다 .
6. OK 를 누릅니다 .
7. Apply 를 누릅니다 .
8. Load Configuration Files 를 선택하여 동적으로 적용합니다 .

또한 Class Manager Virtual Servers 탭과 CGI Settings 링크를 사용하여 가상 서버용 `chroot` 디렉토리를 지정할 수 있습니다 .

가상 서버용으로 `chroot` 디렉토리를 설정하는 방법에 대한 자세한 내용은 Sun ONE Web Server 6.1 *Programmer's Guide* 를 참조하십시오 .

7 장

서버 클러스터 관리

이 장에서는 Sun ONE Web 서버 클러스터링의 개념을 설명하고 이를 사용하여 서버 사이에서 구성을 공유하는 방법에 대하여 설명합니다 .

이 장의 내용 :

클러스터 설명

클러스터는 단일 Administration Server 에 의하여 관리될 수 있는 Sun ONE Web Server 의 그룹입니다 . 각 클러스터에는 반드시 관리 서버로 지정된 서버가 하나 있어야 합니다 . 클러스터가 여러 개인 경우 하나의 " 마스터 " Administration Server 에서 모든 클러스터를 관리할 수 있습니다 . 마스터 관리 서버는 모든 클러스터의 정보를 검색하고 각 클러스터에 설치된 Sun ONE Web Server 를 관리할 수 있는 인터페이스를 제공합니다 .

서버를 클러스터로 조직하여 할 수 있는 몇 가지 작업은 다음과 같습니다 .

- 모든 Sun ONE Web Server 를 관리할 수 있는 중앙의 위치

- 서버 사이에서 하나 이상의 구성 파일 공유
- 하나의 " 마스터 " Administration Server 에서 모든 서버를 시작 및 종료
- 선택한 서버용 액세스 및 오류 로그 확인

Sun ONE Web Server 를 클러스터링하면 모든 클러스터를 관리할 수 있는 마스터 Administration Server 를 지정할 수 있습니다 .

참고 개별 서버는 네트워크의 어느 컴퓨터에나 설치할 수 있으나 , " 마스터 " 로 지정한 Administration Server에는 클러스터된 모든 서버의 정보가 포함되며 반드시 각 클러스터의 개별 Administration Server 에 액세스할 수 있어야 합니다 .

서버 클러스터 사용에 대한 지침

클러스터를 구성하는 경우 모든 클러스터에 대한 정보가 있는 마스터 Administration Server 는 각 클러스터의 Administration Server 와 통신합니다 . 각 클러스터용 관리 서버에는 마스터 Administration Server 에 부여되는 것과 동일한 관리 사용자 이름 및 암호가 부여되어야 합니다 .

클러스터를 만들기 전에 클러스터에 포함시킬 모든 서버가 설치되어 있어야 합니다 . 예를 들어 , 세 개의 클러스터를 만들고 각 클러스터마다 다섯 대의 Sun ONE Web Server 를 넣으려면 다음과 같이 합니다 .

1. 마스터 Administration Server 와 동일한 관리자 사용자 이름 및 암호를 사용하여 실행할 컴퓨터에 모든 서버를 설치합니다 .
2. 각 클러스터에서 Sun ONE Web Server 중 하나를 Administration Server 로 구성합니다 .
3. 단일 클러스터의 관리 서버 중 하나를 모든 클러스터에 대한 마스터 Administration Server로 구성합니다 . 어느 서버를 마스터 관리 서버로 지정하는가는 상관 없습니다 .

주의 클러스터는 오직 같은 종류여야 합니다 . 클러스터의 서버는 모두 UNIX 이거나 Windows 여야 합니다 . UNIX 와 Windows 서버를 동일한 클러스터에서 혼합하면 서버가 중지되거나 손상될 수 있습니다 .

다음 목록은 서버 그룹을 클러스터로 구성하는 데 대한 몇 가지 지침입니다 .

- 클러스터를 만들기 전에 특정 클러스터에 포함할 모든 서버를 설치합니다 .

- 클러스터의 모든 서버는 반드시 버전 6.1 Sun ONE Web Server 여야 합니다 .
- 각 클러스터의 Administration Server 에 마스터 관리 서버와 동일한 사용자 ID 및 암호가 부여되어야 합니다 . 분산 관리를 사용하여 각 Administration Server 에 있는 여러 관리자를 설정할 수 있습니다 .
- 네트워크의 임의 컴퓨터에 서버를 설치합니다 . 단 , 클러스터의 컴퓨터는 모두 Windows 이거나 UNIX 여야 합니다 .
- 임의의 클러스터 특정 Administration Server 를 마스터 관리 서버로 지정할 수 있습니다 .
- 마스터 Administration Server 는 반드시 각 클러스터 특정 Administration Server 에 액세스할 수 있어야 합니다 . 마스터 Administration Server 는 모든 설치된 Sun ONE Web Server 에 대한 정보를 검색합니다 .
- 모든 Administration Server 는 반드시 Sun ONE Web Server 버전 6.0 또는 6.1 이어야 하며 동일한 프로토콜 , HTTP 또는 HTTPS 를 사용해야 합니다 . 오직 Sun ONE Web Server 6.0 또는 6.1 만이 클러스터로의 추가를 지원합니다 .
- 클러스터에 있는 Administration Server 중 하나의 프로토콜을 변경한 경우 모든 Administration Server 의 프로토콜을 변경해야 합니다 . 클러스터의 개별 서버를 변경하려면 Modify Server 인터페이스를 사용합니다 .

클러스터 설정

Sun ONE Web Server 클러스터를 설정하려면 다음과 같이 합니다 .

1. 클러스터에 포함하려는 컴퓨터에 Sun ONE Web Server 를 설치합니다 .

   클러스터용 Administration Server 에 마스터 서버가 인증용으로 사용하는 것과 동일한 사용자 이름 및 암호가 있어야 합니다 . 기본 사용자 이름 및 암호를 사용하거나 분산 관리를 설정하면 동일한 사용자 이름 및 암호를 부여할 수 있습니다 .

2. 마스터 Administration Server 가 포함될 서버를 설치하고 사용자 이름과 암호가 제 1 단계에서 설정한 것과 동일한지 확인합니다 .
3. 클러스터 목록에 서버를 추가합니다 .
4. 클러스터 폼에서 Server Manager 폼에 액세스하거나 클러스터에 있는 한 서버의 구성 파일을 다른 서버로 복사하여 원격 서버를 관리합니다 .

참고 원격 서버용 구성을 변경한 후 , 원격 서버를 재시작합니다 .

클러스터에 서버 추가

클러스터에 서버를 설치하는 경우 해당 Administration Server 와 포트 번호를 지정합니다 . 이 Administration Server 에 하나 이상의 서버에 대한 정보가 있는 경우 해당 서버 모두가 클러스터에 추가됩니다 . 나중에 개별 서버를 제거할 수 있습니다 .

참고 원격 Administration Server 에 클러스터에 대한 정보가 있는 경우 원격 클러스터에 있는 서버는 추가되지 않습니다 . 마스터 Administration Server 는 원격 컴퓨터에 실제로 설치된 서버만 추가합니다 .

클러스터에 원격 서버를 추가하려면 다음과 같이 합니다 .

1. 마스터 Administration Server 가 작동 중인지 확인합니다 .
2. Administration Server 에 액세스하고 Cluster Mgmt 탭을 선택합니다 .
3. Add Server 링크를 누릅니다 .
4. 원격 Administration Server 가 사용하는 프로토콜을 선택합니다 .
   - 정상 Administration Server 의 경우 `http`
   - 보안 Administration Server 의 경우 `https`
5. Admin Server Hostname 필드에 원격 서버의 `magnus.conf` 파일에 표시되는 것과 동일하게 유효한 도메인 이름을 입력합니다 .

   예 : `plaza.sun.com`
6. 원격 Administration Server 의 포트 번호를 입력합니다 .
7. OK 를 누릅니다 .

   마스터 Administration Server 는 이제 원격 서버와의 연결을 시도합니다 . 이 작업에는 수 분 정도 소요됩니다 . 서버가 클러스터에 추가되었음을 확인하는 메시지가 수신됩니다 .
8. OK 를 누릅니다 .

참고 동일한 ID 를 사용하는 서로 다른 시스템에 두 대 이상의 서버가 있는 경우 각 컴퓨터의 서버 ID 와 호스트 이름이 표시됩니다 . 서버 ID 와 호스트 이름이 모두 동일하면 포트 번호 또한 표시됩니다 .

참고 클러스터 제어를 사용하도록 설정하면 클러스터의 마스터가 클러스터의 각 슬레이브용 디렉토리 `https-`*server-instance*`/config/cluster/`*server-name*`/https-`*server-name*`/` 에 여러 개의 파일을 만듭니다 . 이 파일의 구성은 변경할 수 없습니다 .

서버 정보 수정

Modify Server 옵션은 슬레이브 서버에서 슬레이브 관리 포트 번호가 변경된 경우에만 이를 업데이트하는데 사용합니다 . 클러스터에 있는 원격 Administration Server 의 포트 번호를 변경하는 경우 또한 클러스터에 저장된 Administration Server 의 정보도 수정해야 합니다 . 슬레이브 관리 서버에 기타 사항을 변경하면 서버를 삭제한 후 , 변경 사항이 적용된 후 해당 서버를 클러스터에 다시 추가해야 합니다 .

마스터 클러스터 데이터베이스를 수정하는 경우에도 해당 파일을 Cluster Control 을 통하여 전송하지 않는 한 원격 관리 서버는 영향을 받지 않습니다 .

클러스터에 있는 서버에 대한 정보를 수정하려면 다음과 같이 합니다 .

1. 마스터 Administration Server 로 이동하고 Cluster Mgmt 탭을 선택합니다 .
2. Modify Server 링크를 누릅니다 .

   모든 서버 목록이 고유 서버 ID 별로 표시됩니다 .
3. 다음의 방법으로 변경할 서버를 선택합니다 .
   - 특정 서버 선택
   - Select All 누름

   Reset 을 눌러 모든 선택을 취소합니다 .
4. 새로운 포트 번호를 입력합니다 .
5. OK 를 누릅니다 .

클러스터에서 서버 제거

클러스터에서 서버를 제거하려면 다음과 같이 합니다 .

1. 마스터 Administration Server 로 이동하고 Cluster Mgmt 탭을 선택합니다 .
2. Remove Server 링크를 누릅니다 .

3. 다음의 방법으로 변경할 원격 서버를 선택합니다 .
   - 특정 서버 선택
   - Select All 누름

   Reset Selection 을 눌러 모든 선택을 취소합니다 .
4. OK 를 누릅니다 .

클러스터에서 서버가 제거되었음을 확인하는 메시지가 표시됩니다 . 해당 클러스터를 통하여 제거된 서버에 더 이상 액세스할 수 없으며 , 오직 해당 서버의 Administration Server 를 통하여 액세스할 수 있습니다 .

서버 클러스터 제어

Sun ONE Web Server 6.1 에서는 다음과 같이 클러스터에 있는 원격 서버를 제어할 수 있습니다 .

- 서버 시작 및 종료
- 액세스 및 오류 로그 확인
- 구성 파일 전송

주의 클러스터는 반드시 같은 종류여야 합니다 . 클러스터의 서버는 모두 UNIX 이거나 Windows 여야 합니다 . 서로 다른 플랫폼으로 구성 파일을 전송하면 서버가 중단되거나 손상됩니다 .

클러스터 내의 서버를 제어하려면 다음과 같이 합니다 .

1. 마스터 Administration Server 용 Server Manager 로 이동하고 Cluster Mgmt 탭을 선택합니다 .
2. Cluster Control 링크를 누릅니다 .
3. 다음의 방법으로 제어할 서버를 선택합니다 .
   - 특정 서버 선택
   - Select All 을 눌러 클러스터의 모든 서버 선택

   Reset Selection 을 눌러 모든 선택을 취소합니다 .
4. 드롭 다운 목록에서 원격 서버에 대한 Start 또는 Stop 을 선택합니다 .

5. 드롭 다운 목록에서 View Access 또는 View Error 로그 레코드를 선택하고 표시할 줄 수를 입력합니다 .
6. 구성 파일을 전송하려면 다음과 같이 합니다 .
   a. 드롭 다운 목록에서 전송할 구성 파일을 선택합니다 .
   b. 드롭 다운 목록에서 파일을 전송할 원본 서버를 선택합니다 .
   c. Transfer 를 누릅니다 .

변수 추가

변수는 클러스터의 서버를 다른 값으로 구성해야 할 때 사용합니다 . 이들 값은 서로 다른 포트 번호를 사용하는 슬레이브 지정용 마크로이거나 서로 다른 `shlib` 경로를 정의하는 플러그 인일 수 있습니다 .

변수를 추가하면 오직 마스터 클러스터 데이터베이스에만 영향을 미칩니다 . Cluster Control 을 통하여 해당 파일을 전송하지 않는 한 원격 관리 서버는 영향을 받지 않습니다 . 변수가 정의되면 Administration Server 는 더 이상 독립적으로 실행될 수 없습니다 .

클러스터에 원격 서버용 변수를 추가하려면 다음과 같이 합니다 .

1. 마스터 Administration Server 에서 Cluster Mgmt 탭을 선택합니다 .
2. Add Variables 링크를 누릅니다 .
3. 변수를 추가하려는 특정 서버를 선택합니다 .
4. Name 필드에 추가하는 변수의 유형을 입력합니다 .

   예 : "Port".
5. Value 필드에 추가하는 값을 입력합니다 .

   예 : "Port" 를 Name 필드에 입력했으면 값은 포트 번호가 될 것입니다 .
6. OK 를 누릅니다 .

   서버 변수가 추가되었다는 확인 메시지가 표시됩니다 .
7. OK 를 누릅니다 .

변수는 또한 반드시 슬레이브로 전송하는 서버의 구성 파일에도 추가되어야 합니다 . 예를 들어 , 변수 `port` 를 전송하는 경우 변수는 아래와 같이 `server.xml` 등의 서버 구성 파일에 선언되어야 합니다 .

```
<SERVER legacyls="ls1" qosactive="no" qosmetricsinterval="30"
qosrecomputeinterval="100">

...

<LS id="ls1" ip="0.0.0.0" port="$port" security="off"
acceptorthreads="1" blocking="no">

...

</SERVER>
```

구성 파일의 각 슬레이브마다 서로 다른 값으로 변수를 설정할 수 있습니다 . 일단 변수가 추가되면 Add Variable 페이지의 드롭 다운 Option 목록에서 편집하거나 삭제할 수 있습니다 .

제 3 부

구성 , 모니터링 및 성능 조정

8 장

서버 기본설정 구성

이 장에서는 Sun ONE Web Server 용 서버 기본 설정을 구성하는 방법에 대하여 설명합니다 .

이 장의 내용 :

서버 시작 및 정지

서버는 설치된 후 지속적으로 실행되어 HTTP 요청을 청취하고 허용합니다 .

서버의 상태는 Server On/Off 페이지에 표시됩니다 . 다음 중 한 가지 방법으로 서버를 시작 및 정지시킬 수 있습니다 .

- Server On/Off 페이지에서 Server On 또는 Server Off 를 누릅니다 .
- 제어판의 서비스 창을 사용합니다 (Windows).

- start 를 사용합니다 . 이 스크립트를 `init` 와 함께 사용하려면 반드시 start 명령 `http:2:respawn:`*`server_root`*`/`*`type-identifier`*`/start -start -i` 를 `/etc/inittab` 에 포함시켜야 합니다 . (UNIX/Linux)
- `stop` 을 사용하면 서버가 완전히 종료되며 서버가 다시 시작할 때까지 서비스가 중단됩니다 . `etc/inittab` 파일이 자동으로 재시작하도록 설정하려면 ("`respawn`" 사용) 반드시 서버를 종료하기 전에 `etc/inittab` 에 있는 웹 서버 관련 줄을 제거해야 합니다 . 그렇지 않은 경우 서버가 자동으로 재시작합니다 . (UNIX/Linux)

서버를 종료하면 서버가 종료 과정을 완료하고 상태를 "Off" 로 변경하는 데 약간의 시간이 걸릴 수 있습니다 .

컴퓨터가 중단되거나 오프라인이 되는 경우에는 서버가 중단되며 서비스하는 모든 요청을 잃게 됩니다 .

참고 서버에 보안 모듈이 설치된 경우 서버를 시작 또는 중지하기 전에 적절한 비밀번호를 입력해야 합니다 .

참고 UNIX 의 경우 Sun ONE Web Server 설치에 따라 운영 체제에서 기본적으로 허용하는 것보다 많은 메모리 및 파일 기술자가 필요한 경우가 있습니다 . 서버를 시작할 수 없는 경우에는 `ulimit` 명령을 사용하여 운영 체제가 적용한 리소스 한계를 확인합니다 . 운영 체제의 `ulimit` man 페이지에 더 자세한 정보가 제공될 것입니다 .

종료 시간 제한 설정

서버가 정지되면 새 연결을 받지 않습니다 . 또한 기존 연결이 완료되도록 대기합니다 . 제한 시간까지 서버가 대기하는 시간은 `magnus.conf` 파일에서 구성할 수 있으며 , 이 파일은 *`server_root`*`/https-`*`server_name`*`/config/` 에 있습니다 . 기본값은 30 초입니다 . 이 값을 변경하려면 `magnus.conf` 에 다음과 같은 줄을 추가합니다 .

```
TerminateTimeout seconds
```

여기에서 `seconds` 는 서버가 시간 제한 동안 대기하는 초 단위 시간입니다 .

이 값을 구성하면 서버가 연결이 완료될 때까지 더 긴 시간 동안 대기하는 장점이 있습니다 . 그러나 때로 서버에는 응답하지 않는 클라이언트의 연결이 있으므로 종료 시간 제한을 크게 하면 서버가 종료되는 시간이 더 오래 걸릴 수 있습니다 .

서버 재시작 (UNIX/Linux)

다음 중 한 가지 방법으로 서버를 재시작할 수 있습니다 .

- `inittab` 파일에서 자동으로 재시작합니다 .

  참고로 System V 에서 유도되지 않은 UNIX/Linux 버전 (SunOS 4.1.3 등) 을 사용하는 경우 `inittab` 파일을 사용할 수 없습니다 .

- 컴퓨터가 재부팅할 때 `/etc/rc2.d` 에 있는 데몬으로 자동 재시작합니다 .
- 직접 재시작합니다 .

설치 스크립트는 `/etc/rc.local` 또는 `/etc/inittab` 파일을 편집할 수 없으므로 반드시 텍스트 편집기에서 이 파일을 편집해야 합니다 . 이 파일의 편집 방법을 모르는 경우에는 시스템 관리자에게 문의하거나 시스템 설명서를 참조하십시오 .

보통 SSL 을 사용하는 서버의 경우 시작하기 전에 비밀번호가 필요하므로 이 파일을 사용하여 시작할 수 없습니다 . 암호를 일반 텍스트 파일에 저장하여 SSL 을 사용하는 서버를 자동으로 시작할 수는 있으나 , 이는 권장하지 않습니다 .

주의 SSL 을 사용하는 서버의 비밀 번호를 서버 시작 스크립트의 보통 텍스트에 보관하면 보안의 위험이 커집니다 . 파일에 액세스할 수 있는 사용자는 모두 SSL 을 사용하는 서버의 암호를 알 수 있습니다 . SSL 을 사용하는 서버의 비밀 번호를 보통 텍스트에 보관하기 전에 보안의 위험에 대하여 고려해야 합니다 .

서버의 시작 스크립트 , 키쌍 파일 및 키 비밀번호는 루트가 소유해야 하며 (또는 루트가 아닌 서버가 서버를 설치한 경우 해당 사용자 계정) 오직 소유자만이 이에 대한 읽기 및 쓰기 액세스 권한이 부여되어야 합니다 .

SSL 사용 서버 자동 시작

보안의 위험을 걱정하지 않는 경우 다음과 같이 SSL 을 사용하는 서버를 자동으로 시작할 수 있습니다 .

1. 텍스트 편집기를 사용하여 시작 파일을 엽니다 . 이 파일은 *server_root*`/https-`*server_id* 에 있습니다 .

2. 스크립트에서 `-start` 줄을 찾은 후 다음을 삽입합니다 .

   `echo "`*password*`"|`

   여기에서 *password* 는 선택한 SSL 비밀번호입니다 .

   예를 들어 SSL 비밀번호가 `netscape` 인 경우 이 줄은 다음과 같을 것입니다 .

   ```
   -start)
   echo "netscape"|./$PRODUCT_BIN -d $PRODUCT_SUBDIR/config $@
   ```

Inittab 을 사용하여 재시작 (UNIX/Linux)

`inittab` 을 사용하여 서버를 재시작하려면 다음 줄을 `/etc/inittab` 파일에 추가합니다 .

```
http:23:respawn:server_root/type-identifier/start -start -i
```

여기에서 *`server_root`* 는 서버를 설치한 디렉토리이며 *`type-identifier`* 는 서버의 디렉토리입니다 .

`-i` 옵션을 사용하면 서버가 자체를 배경 프로세스로 전환할 수 없도록 방지합니다 .

서버를 정지하기 전에 반드시 이 줄을 제거해야 합니다 .

시스템 RC 스크립트를 사용하여 재시작 (UNIX/Linux)

`/etc/rc.local` 또는 시스템의 해당 파일을 사용하는 경우 `/etc/rc.local` 에 다음을 추가합니다 .

```
server_root/type-identifier/start
```

`server_root` 를 서버를 설치한 디렉토리로 대체합니다 .

서버를 직접 재시작 (UNIX/Linux)

명령줄에서 서버를 재시작하려면 서버가 1024 보다 낮은 번호의 포트에서 실행되는 경우 root 로 로그인하고 , 그렇지 않은 경우 root 또는 서버의 사용자 계정으로 로그인합니다 . 명령줄 프롬프트에서 다음 줄을 입력하고 Enter 를 누릅니다 .

```
server_root/type-identifier/start
```

여기에서 *`server_root`* 는 서버를 설치한 디렉토리입니다 .

줄의 마지막에 선택 매개 변수인 -i 를 사용할 수 있습니다 . -i 옵션을 사용하면 서버가 inittab 모드로 실행되므로 서버 프로세스가 중단 또는 정지되는 경우 inittab 이 서버를 자동으로 재시작합니다 . 또한 이 옵션을 사용하면 서버가 자체를 배경 프로세스로 전환할 수 없도록 방지합니다 .

참고 서버가 이미 실행 중인 경우 start 명령은 실패합니다 . 반드시 서버를 우선 정지시킨 후 start 명령을 사용해야 합니다 . 또한 서버가 시작되지 않는 경우에는 서버를 재시작하기 전에 프로세스를 종료해야 합니다 .

서버를 직접 정지 (UNIX/Linux)

etc/inittab 파일을 사용하여 서버를 재시작하는 경우 반드시 /etc/inittab 에서 서버를 시작하는 줄을 제거하고 서버를 정지하기 전에 kill -1 1 을 입력해야 합니다 . 그렇지 않으면 서버가 정지된 후 자동으로 재시작하게 됩니다 .

서버를 직접 정지하려면 root 또는 서버의 사용자 계정 (이 계정으로 서버를 시작한 경우) 을 사용하여 로그인 한 후 , 명령 줄에서 다음을 입력합니다 .

```
server_root/type-identifier/stop
```

서버 재시작 (Windows)

다음과 같이 서버를 재시작할 수 있습니다 .

- Service Control Panel 을 사용하여 서버를 재시작합니다 .
- Services Control Panel 을 사용하여 컴퓨터를 재시작할 때 항상 운영 체제가 서버 또는 관리 서버를 재시작하도록 구성합니다 .

Windows 의 경우 다음과 같이 합니다 .

1. 제어판에서 서비스 아이콘을 두 번 누릅니다 .
2. 서비스 목록에서 서버용 서비스를 선택합니다 .
3. 자동을 선택하여 컴퓨터가 시작하거나 재부팅할 때 항상 서버를 시작하도록 합니다 .
4. OK 를 누릅니다 .

참고 또한 서비스 대화 상자를 사용하여 서버가 사용하는 계정을 변경할 수 있습니다 . 서버가 사용하는 계정을 변경하는 방법은 " 사용자 계정 변경 (UNIX/Linux)" 페이지 98 을 참조하십시오 .

기본적으로 시작하기 전에 관리자에게 키 데이터베이스 암호를 입력하라는 프롬프트가 웹 서버에 표시됩니다 . 무인 작업으로 웹 서버를 재시작하려는 경우에는 암호를 `password.conf` 파일에 저장해 놓아야 합니다 . 시스템이 적절히 보호되어 이 파일과 암호 데이터베이스가 조작되지 않을 경우에만 이 기능을 사용하십시오 .

자동 재시작 유틸리티 사용 (Windows)

서버가 중단되는 경우 서버는 서버 모니터 유틸리티에 의하여 재시작됩니다 . 디버깅 도구가 설치된 시스템의 경우 서버가 중지되면 디버깅 정보가 있는 대화 상자가 표시됩니다 . 시간 제한 값을 매우 길게 설정하여 자동 시작 기능을 사용하지 않도록 하면 디버깅 서버 플러그인 API 프로그램 (예를 들어 SSAPI 프로그램) 에 도움이 됩니다 . 또한 레지스트리 편집기를 사용하여 디버깅 대화 상자가 표시되지 않도록 할 수 있습니다 .

시간 간격 변경 (Windows)

시작과 서버가 자동으로 재시작되는 시간 사이의 간격을 변경하려면 다음과 같이 합니다 .

1. 레지스트리 편집기를 시작합니다 .
2. 서버의 키를 선택합니다 . (레지스트리 편집기 창의 왼쪽 , `HKEY_LOCAL_MACHINE\SOFTWARE\Netscape\Enterprise\6.0` 에 위치)
3. 편집 메뉴에서 값 추가를 선택합니다 . 키 추가 대화 상자가 표시됩니다 .
4. 값 이름에서 `MortalityTimeSecs` 를 입력합니다 .
5. 데이터 유형 드롭 다운 목록에서 REG_DWORD 를 선택합니다 .
6. OK 를 누릅니다 . DWORD 편집기 대화 상자가 표시됩니다 .
7. 시작과 서버가 자동으로 재시작하는 시간 사이에 경과하는 시간 간격을 입력(초 단위) 합니다 .

   간격은 이진수 , 10 진수 또는 16 진수 형식으로 입력할 수 있습니다 .
8. 앞의 단계에서 입력하는 값의 숫자 형식(이진수, 10진수 또는 16진수)을 누릅니다 .
9. OK 를 누릅니다 .

   레지스트리 편집기의 오른쪽 창에 `MortalityTimeSecs` 가 16 진수로 표시됩니다 .

디버깅 대화 상자 사용 중지 (Windows)

시스템 디버깅 설정을 수정한 응용 프로그램 (컴파일러 등) 을 설치하고 서버가 중지된 경우 시스템이 생성한 응용 프로그램 오류 대화 상자를 확인할 수 있습니다 . 서버는 확인을 누를 때까지 재시작되지 않습니다 .

서버가 중지되는 경우 표시되는 디버깅 대화 상자를 사용하지 않으려면 다음과 같이 합니다 .

1. 레지스트리 편집기를 시작합니다 .
2. AeDebug 키를 선택합니다 . (레지스트리 편집기 창의 왼쪽 , `HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion` 에 위치)
3. 창의 오른쪽에서 자동 값을 두 번 누릅니다 .

   문자열 편집기 편집기 대화 상자가 표시됩니다 .
4. 값을 1 로 변경합니다 .

성능을 위한 서버 조정

스레드 한계는 `magnus.conf` 파일을 편집하거나 Server Manager 를 통하여 조정할 수 있습니다 .

`magnus.conf` 파일을 편집하는 경우 `RqThrottleMinPerSocket` 은 최소 값이며 `RqThrottle` 은 최대 값입니다 .

최소 한계는 서버가 `WaitingThreads` 상태에 유지할 목표 스레드의 수입니다 . 이 숫자는 단지 목표치입니다. 이 상태의 실제 스레드 수는 이 값보다 약간 많거나 적을 수 있습니다 . 기본값은 48 입니다 . 최대 스레드는 동시에 실행할 수 있는 사용 중인 스레드의 최대 수에 대한 고정된 한계로 성능의 병목이 될 수 있습니다 . 기본값은 128 입니다 .

Server Manager 를 사용하는 경우 다음과 같이 합니다 .

1. Preferences 탭으로 이동합니다 .
2. Performance Tuning 링크를 누릅니다 .
3. Maximum simultaneous requests 필드에 원하는 값을 입력합니다 .

`RqThrottleMinPerSocket` 및 `RqThrottle` 매개 변수에 대한 자세한 내용은 Sun ONE Web Server 6.1 *Administrator's Configuration File Reference* 를 참조하십시오 .

이 설정 및 기타 내용의 성능 내용에 대한 자세한 내용은 Sun ONE Web Server 6.1 *Performance Tuning, Sizing, and Scaling Guide* 를 참조하십시오 .

magnus.conf 파일 편집

Sun ONE Web Server 가 시작되면 *server_root*/*server_id*/`config` 디렉토리의 `magnus.conf` 파일을 찾아 서버의 작동과 구성에 영향을 미치는 전역 변수 세트를 설정합니다 . Sun ONE Web Server 는 `magnus.conf` 에 정의된 모든 지시문을 실행합니다 . Server Manager 의 Magnus Editor 를 사용하여 `magnus.conf` 의 설정을 편집할 수 있습니다 .

`magnus.conf` 파일에 대한 자세한 설명과 텍스트 편집기를 사용하여 파일을 편집하는 방법은 Sun ONE Web Server 6.1 *Administrator's Configuration File Reference* 와 Sun ONE Web Server 6.1 *NSAPI Programmer's Guide* 를 참조하십시오 .

Magnus Editor 에 액세스하려면 다음과 같이 합니다 .

1. Server Manager 에 액세스하고 Preferences 탭을 선택합니다 .
2. Magnus Editor 링크를 누릅니다 .
3. 드롭 다운 목록에서 설정을 선택하고 Manage 를 누릅니다 .

   Server Manager 에 지정한 설정에 대한 편집기가 표시됩니다 .
4. 원하는 설정을 변경한 다음 OK 를 누릅니다 .

Settings 페이지에 대한 자세한 내용은 온라인 도움말의 Magnus Editor 페이지를 참조하십시오 .

청취 소켓 추가 및 편집

서버가 요청을 처리하려면 청취 소켓을 통하여 요청을 접수한 후 요청을 올바른 가상 서버로 보내야 합니다 . Sun ONE Web Server 를 설치하는 경우 청취 소켓 한 개 (ls1) 가 자동으로 만들어집니다 . 이 청취 소켓은 IP 주소 0.0.0.0 과 설치 도중 HTTP 서버 포트 번호로 지정한 포트 번호 (기본 값은 80) 를 사용합니다 . 기본 청취 소켓은 삭제할 수 없습니다 .

Server Manager 의 Listen Sockets Table 을 사용하여 서버의 청취 소켓 설정을 편집할 수 있습니다 . 이 테이블에 액세스하려면 다음과 같이 합니다 .

1. Server Manager 에 액세스하고 Preferences 탭을 누릅니다 .
2. Edit Listen Sockets 링크를 누릅니다 .
3. 원하는 사항을 변경한 다음 OK 를 누릅니다 .

MIME 유형 선택

Mime Types 페이지에서 서버의 MIME 파일을 편집할 수 있습니다 .

MIME(Multi-purpose Internet Mail Extension) 유형에 따라 전자우편 시스템이 지원하는 멀티미디어 파일의 유형이 달라집니다 . MIME 유형은 또한 예를 들어 CGI 프로그램에 지정할 파일 유형과 같이 특정 서버 파일 유형에 해당하는 파일 확장자를 지정합니다 .

각 가상 서버용으로 별도의 MIME 유형을 만들 필요는 없습니다 . 대신 , 필요한 만큼의 MIME 유형 파일을 만들고 이를 가상 서버에 연결합니다 . 서버에는 하나의 MIME 유형 파일인 `mime.types` 가 기본으로 존재하며 , 이는 삭제할 수 없습니다 . 이 파일은 절대 경로일 수 있습니다 .

MIME Types 페이지에 액세스하려면 다음과 같이 합니다 .

1. Server Manager 에 액세스하고 Preferences 탭을 누릅니다 .
2. MIME Types 링크를 누릅니다 .
3. 원하는 사항을 변경한 다음 OK 를 누릅니다 .

자세한 내용은 온라인 도움말의 Mime Settings 페이지와 제 13 장 , " 가상 서버 사용 " 을 참조하십시오 .

액세스 제한

Server Manager 의 Restrict Access 페이지를 사용하여 전체 서버 또는 서버의 일부분 (디렉토리 , 파일 , 파일 유형 등) 에 대한 액세스를 제어할 수 있습니다 . 서버는 입중계 요청을 평가할 때 ACE(Access-control Entries) 라고 하는 규칙 계층에 따라 액세스를 결정하며 , 그런 후 일치되는 항목을 사용하여 요청의 허가 여부를 결정합니다 . 각 ACE 는 서버가 계층의 다음 ACE 로 계속할 것인지의 여부를 지정합니다 . ACD

의 컬렉션은 ACL(Access-control List) 라고 합니다 . 서버로 요청이 들어오면 서버는 *vsclass*`.obj.conf`(여기에서 *vsclass* 는 가상 서버 클래스 이름) 에서 ACL 에 대한 참조를 찾으며 , 이는 다시 액세스를 결정하는데 사용됩니다 . 기본으로 서버에는 하나의 ACL 파일이 있으며 여기에는 여러 개의 ACL 이 있습니다 .

Administration Server 를 통하여 모든 서버용으로 액세스 제어를 전역적으로 설정하거나 Server Manager 를 통하여 특정 서버 인스턴스 내의 리소스용으로 설정할 수 있습니다 . 리소스용으로 액세스 제어를 설정하는 방법은 을 참조하십시오 .

" 액세스 제어 설정 " 페이지 188(제 9 장 , " 서버 액세스 제어 ").

참고	서버 액세스를 제한하기 전에 반드시 분산 서버를 사용하도록 설정해야 합니다 .

Sun ONE Web Server 에 대한 액세스를 제한하려면 다음과 같이 합니다 .

1. Server Manager 에 액세스하고 Preferences 탭을 선택합니다 .
2. Restrict Access 링크를 누릅니다 .

자세한 내용은 제 9 장 , " 서버 액세스 제어 " 와 온라인 도움말의 Restrict Access 페이지를 참조하십시오 .

구성 설정 복구

구성 복원 페이지에서 구성 파일의 백업 사본을 확인하고 특정 일자에 저장된 구성 데이터로 되돌릴 수 있습니다 .

참고	Windows 의 경우 이 페이지에서 오직 구성 파일에 직접 적용한 변경 사항만 복원됩니다 . 설치 동안 만들어진 백업 버전으로는 되돌리면 안 됩니다 . 이 버전은 완전하지 않을 수 있습니다 .

자세한 내용은 온라인 도움말의 Restore Configuration 페이지를 참조하십시오 .

파일 캐시 구성

Sun ONE Web Server 는 파일 캐시를 사용하여 정적 정보를 더욱 빨리 서비스합니다 . 이전 버전의 서버에서는 요청을 파일 캐시로 라우팅하는 가속기 캐시가 있었으나 , 가속기 캐시는 더 이상 사용되지 않습니다 . 파일 캐시에는 파일에 대한 내용과 정적 파일 컨텐츠가 포함됩니다 . 또한 파일 캐시는 서버가 파싱한 HTML 의 처리를 빠르게 하는데 사용되는 정보를 캐시합니다 .

기본적으로 파일 캐시는 사용하도록 설정됩니다 . 파일 캐시 설정은 `nsfc.conf` 라는 파일에 들어 있습니다 . Server Manager 를 사용하여 파일 캐시 설정을 변경할 수 있습니다 .

더 자세한 내용은 `http://docs.sun.com/` 의 *Performance Tuning and Sizing Guide* 를 참조하십시오 .

스레드 풀 추가 및 사용

스레드 풀을 사용하여 일정한 수의 스레드를 특정 서비스에 할당할 수 있습니다 .

스레드 풀의 다른 용도는 스레드를 사용할 수 없는 플러그인을 실행하는 것입니다 . 최대 스레드 수를 1 로 설정하여 풀을 지정하면 지정된 서비스 기능에 오직 하나의 요청만 허용됩니다 .

스레드 풀을 추가하면 지정한 정보에 스레드의 최소 및 최대 수 , 스택 크기 및 큐 크기가 포함됩니다 .

더 자세한 내용은 `http://docs.sun.com/` 의 *Performance Tuning and Sizing Guide* 를 참조하십시오 .

원시 스레드 풀 및 일반 스레드 풀 (Windows)

Windows 의 경우 원시 스레드 풀 (`NativePool`) 과 추가의 일반 스레드 풀 등 두 가지 스레드 풀을 사용할 수 있습니다 .

원시 스레드 풀을 편집하려면 Server Manager 의 Native Thread Pool 페이지에 액세스합니다 .

원하는 용도만큼 원하는 수의 일반 스레드 풀을 만들 수 있습니다 . 일반 스레드 풀을 만들려면 Server Manager 의 Generic Thread Pools 페이지에 액세스합니다 .

스레드 풀 (UNIX/Linux)

UNIX/Linux 의 스레드는 항상 운영 체제가 스케줄하므로 (사용자 스케줄과 반대) UNIX/Linux 사용자는 `NativePool` 을 사용할 필요가 없으며 이 설정을 편집하는 Server Manager 페이지가 없습니다. 그러나 UNIX/Linux 사용자는 계속 스레드 풀을 만들 수 있습니다 . 스레드 풀을 만들려면 Server Manager 의 Thread Pools 페이지에 액세스합니다 .

스레드 풀 편집

일단 스레드 풀을 추가하면 Server Manager 를 통하여 스레드 풀 설정 값 (최소 스레드 , 최대 스레드 등) 을 편집할 수 있습니다 .

또한 *vsclass*`.obj.conf` 에 있는 스레드 풀 설정을 편집할 수 있는데 , 여기에서 *vsclass* 는 가상 서버 클래스 이름입니다 .

vsclass`.obj.conf` 에 표시되는 스레드 풀은 다음과 같습니다 .

```
Init fn="thread-pool-init" name=name_of_the_pool MaxThreads=n
MinThreads=n QueueSize=n StackSize=n
```

폴을 변경하려면 다음 매개 변수를 사용합니다 . `MinThreads`, `MaxThreads`, `QueueSize` 및 `StackSize`.

Windows 사용자는 Server Manager 를 사용하여 언제라도 원시 풀용 설정을 편집할 수 있습니다 .

스레드 풀 사용

스레드 풀을 설정한 후 스레드 풀을 특정 서비스용으로 지정하여 사용합니다 .

스레드 풀을 구성하려면 Server Manager 탭으로 이동하고 Thread Pool 을 선택합니다 . 스레드 풀이 구성되었으면 지정한 특정 서비스용으로 사용할 수 있는 스레드 풀이 Thread Pool 목록에 표시됩니다 .

또한 *vsclass*`.obj.conf` 에 있는 `load-modules` 함수의 `pool` 매개 변수를 사용하여 스레드 풀을 지정할 수 있습니다 . 여기에서 *vsclass* 는 가상 서버 클래스 이름입니다 .

`pool=` *"name_of_pool"*

또한 임의의 NSAPI 함수에서 `pool` 매개 변수를 사용하여 오직 해당 NSAPI 기능만 지정한 풀에서 실행되도록 할 수 있습니다 .

9 장

서버 액세스 제어

이 장에서는 Administration Server 와 웹 사이트의 파일 또는 디렉토리에 대한 액세스를 제어하는 다양한 방법에 대하여 설명합니다 . 예를 들어 Administration Server 의 경우 컴퓨터에 설치된 모든 서버를 모두 제어할 수 있는 사람과 하나 이상의 서버를 부분적으로 제어할 수 있는 사람을 지정할 수 있습니다 . Administration Server 에 대한 액세스 제어를 사용하기 전에 반드시 분산 관리를 사용하도록 설정하고 LDAP 데이터베이스에 관리 그룹을 설정해야 합니다 . 이 장에서는 이미 분산 관리를 구성했으며 LDAP 데이터베이스에 사용자 및 그룹을 정의한 것으로 가정합니다 .

또한 제 4 장 , " 웹 컨테이너 및 웹 응용 프로그램용 J2EE 기반 보안 " 와 제 6 장 , " 인증서 및 키 사용 " 에 설명한 것과 같이 웹 서버가 보안되었는지 확인해야 합니다 .

이 장의 내용 :

액세스 제어 설명

액세스 제어를 사용하여 다음을 결정할 수 있습니다 .

- Sun ONE Web Administration Server 에 액세스할 수 있는 사용자
- 액세스할 수 있는 프로그램
- 웹 사이트의 파일 또는 디렉토리에 액세스할 수 있는 사용자

서버의 전체 또는 일부 , 또는 웹 사이트의 파일 또는 디렉토리에 대한 액세스를 제어할 수 있습니다 . ACE(Access Control Entry) 라는 규칙의 계층을 만들어 액세스를 허용하거나 거부할 수 있습니다 . 각 ACE 는 서버가 계층의 다음 ACE 를 확인할 것인지의 여부를 지정합니다 . 만드는 ACE 의 컬렉션은 ACL(Access-control List) 이라고 합니다 .

기본으로 서버에는 하나의 ACL 파일이 있으며 여기에는 여러 개의 ACL 이 있습니다 . 입중계 요청용 가상 서버가 결정된 후 , Sun ONE Web Server 는 해당 가상 서버용으로 구성된 ACL 이 있는지 확인합니다 . 현재 요청에 대하여 적용되는 ACL 이 있는 경우 Sun ONE Web Server 는 ACE 를 평가하여 액세스를 허용할 것인지 또는 거부할 것인지 결정합니다 .

다음을 기준으로 액세스를 허용 또는 거부합니다 .

- 요청하는 사용자 (사용자 - 그룹)
- 요청의 출처 (호스트 IP)
- 요청이 발생한 시간 (예 : 하루 중 시간)
- 사용되는 연결의 종류 (SSL)

사용자 - 그룹용 액세스 제어 설정

웹 서버에 대한 액세스를 특정 사용자 또는 그룹으로 제한할 수 있습니다 . 사용자 - 그룹 액세스 제어를 사용하려면 사용자가 해당 서버에 액세스하기 전에 사용자 이름과 비밀번호를 입력해야 합니다 . 서버는 클라이언트 인증서에 있는 정보 , 또는 클라이언트 인증서 자체를 디렉토리 서버 항목과 비교합니다 .

Administration Server 는 오직 기본 인증만 사용합니다 . Administration Server 에 클라이언트 인증이 필요하도록 하려면 반드시 `obj.conf` 의 ACL 파일을 직접 편집하여 방법을 SSL 로 변경해야 합니다 .

사용자 - 그룹 인증은 서버용으로 구성된 디렉토리 서비스가 수행합니다 . 이 부분에 대한 더 자세한 내용은 디렉토리 서비스 구성를 참조하십시오 . 디렉토리 서비스가 액세스 제어를 구현하는데 사용하는 정보는 다음 중 한 가지 소스에서 구합니다 .

- 내부 보통 파일 유형 데이터베이스

- 외부 LDAP 데이터베이스

서버가 LDAP 기반 디렉토리 서비스를 사용하는 경우 서버 인스턴스용으로 다음 유형의 사용자 - 그룹 인증 메서드를 지원합니다 .

- 기본값
- Basic
- SSL
- Digest
- 기타

서버가 내부 파일 기반 디렉토리 서비스를 사용하는 경우 서버 인스턴스용으로 다음 유형의 사용자 - 그룹 인증 메서드를 지원합니다 .

- 기본값
- Basic
- Digest

사용자 - 그룹 인증의 경우 사용자가 Administration Server 에 액세스하거나 웹 사이트의 파일 및 디렉토리에 액세스하기 전에 자신의 신분을 증명해야 합니다 . 인증 과정에서 사용자는 사용자 이름과 비밀번호를 입력하거나 클라이언트 인증서 또는 다이제스트 인증 플러그인을 사용하여 자신의 신분을 증명합니다 . 클라이언트 인증서를 사용하려면 암호화가 필요합니다 . 암호화 및 클라이언트 인증서 사용에 대한 자세한 내용은 제 4 장 , " 웹 컨테이너 및 웹 응용 프로그램용 J2EE 기반 보안 " 을 참조하십시오 .

Default 인증

Default 인증은 가장 많이 사용되는 방법입니다 . Default 설정은 `obj.conf` 에 지정한 기본 방법을 사용하거나 , `obj.conf` 에 설정이 없는 경우에는 "Basic" 을 사용합니다 . Default 를 선택하는 경우 ACL 규칙은 ACL 파일에 메소드를 지정하지 않습니다 . Default 를 선택하면 `obj.conf` 파일에서 한 줄만 편집하면 모든 ACL 에 대한 메서드를 쉽게 변경할 수 있습니다 .

Basic 인증

Basic 인증의 경우 사용자가 웹 서버 또는 웹 사이트에 액세스하기 위한 사용자 이름과 암호를 입력해야 합니다 . 이 설정이 기본값입니다 . 반드시 사용자 및 그룹의 목록을 만들고 이를 Sun ONE Directory Server 등의 LDAP 데이터베이스 또는 파일에 저장해야 합니다 . 반드시 웹 서버와 다른 루트 디렉토리에 설치된 디렉토리 서버 또는 원격 컴퓨터에 설치된 디렉토리 서버를 사용해야 합니다 .

Administration Server 또는 웹 사이트에서 사용자 - 그룹 인증이 있는 리소스에 액세스하려는 경우 웹 브라우저에 사용자 이름과 비밀번호를 입력하라는 대화 상자가 표시됩니다 . 서버에서 암호화 기능이 사용되는지의 여부에 따라 이 정보는 암호화 또는 암호화되지 않은 형태로 서버에 입력됩니다 .

참고 SSL 암호화가 없는 Basic 인증을 사용하는 경우 사용자 이름과 비밀번호가 암호화되지 않은 텍스트로 네트워크에 전송됩니다 . 이 네트워크 패킷은 포착될 수 있으며 사용자 이름과 암호가 도용될 수 있습니다 . Basic 인증은 SSL 암호화나 Host-IP 인증 , 또는 이 둘 모두와 함께 사용할 때 가장 효과적입니다 . Digest 인증을 사용하면 이 문제를 피할 수 있습니다 .

사용자가 서버에 자신의 신분을 인증하면 다음 대화상자가 표시됩니다 .

사용자 이름 및 비밀번호 프롬프트 예

확인을 누르면 다음이 표시됩니다 .

- Sun ONE Web Administration Server 로 인증된 경우 Server Administration 페이지
- 웹 사이트로 로그인된 경우 요청한 파일 또는 디렉토리 목록
- 사용자 이름이나 비밀번호가 잘못된 경우 액세스를 거부하는 메시지

Access Denied Response 페이지에서 허가되지 않은 사용자에게 표시되는 액세스 거부 메시지를 사용자 정의할 수 있습니다 .

SSL 인증

서버가 보안 인증서가 있는 사용자의 신분을 확인하는 방법은 두 가지입니다 .

- 클라이언트 인증서의 정보를 신분 증명으로 사용
- LDAP 디렉토리에 게시된 클라이언트 인증서 확인 (추가)

서버가 클라이언트 인증용으로 인증서 정보를 사용하도록 설정하면 서버는 다음의 작업을 합니다 .

- 우선 인증서가 신뢰된 CA 에서 발행된 것인지 확인합니다 . 그렇지 않은 경우 인증이 실패하며 트랜잭션이 종료됩니다 . 클라이언트 인증을 사용하는 방법은 " 클라이언트 인증 필수화 " 페이지 136 을 참조하십시오 .
- 인증서가 신뢰된 인증기관 (CA) 에서 발행된 경우 `certmap.conf` 파일을 사용하여 인증서를 사용자 항목과 매핑합니다 . 인증서 매핑 파일의 설정 방법은 "certmap.conf 파일 사용 " 페이지 139 을 참조하십시오 .
- 인증서가 올바로 매핑된 경우 해당 사용자에 대하여 설정된 ACL 규칙을 확인합니다 . 인증서가 올바로 매핑된 경우라도 ACL 규칙에 따라 사용자 액세스를 거부할 수 있습니다 .

특정 리소스에 대한 액세스를 제어하는 용도로 필요한 클라이언트 인증은 서버에 대한 모든 연결에 대하여 클라이언트 인증을 요구하는 것과 다릅니다 . 모든 연결에 대하여 서버가 클라이언트 인증을 요구하도록 설정한 경우 클라이언트는 단지 신뢰된 CA 가 발행한 유효한 인증서만 제시하면 됩니다 . 서버의 액세스 제어가 사용자 및 그룹 인증용 SSL 방법을 사용하도록 설정하는 경우 클라이언트는 다음의 작업을 해야 합니다 .

- 신뢰된 CA 가 발행한 유효한 인증서를 제시합니다 .
- 인증서는 반드시 LDAP 의 유효한 사용자와 매핑되어야 합니다 .
- 액세스 제어 목록이 반드시 적절히 평가해야 합니다 .

액세스 제어와 함께 클라이언트 인증을 요구하는 경우 웹 서버용 SSL 암호를 사용하도록 설정해야 합니다 . SSL 을 사용하도록 하는 방법은 제 6 장 , " 인증서 및 키 사용 " 을 참조하십시오 .

SSL 인증이 요구되는 리소스에 성공적으로 액세스하려면 반드시 웹 서버가 신뢰하는 CA 로부터 클라이언트 인증서를 발행되어야 합니다 . 서버의 `certmap.conf` 파일이 브라우저에 있는 클라이언트 인증서를 디렉토리 서버에 있는 클라이언트 인증서와 비교하도록 구성된 경우에는 클라이언트 인증서가 디렉토리 서버 내에 게시되어야 합니다 . 그러나 `certmap.conf` 파일은 인증서의 선택된 정보만 디렉토리 서버 항

목과 비교되도록 구성할 수 있습니다 . 예를 들어 브라우저 인증서에 있는 사용자 ID 와 전자우편 주소만 디렉토리 서버 항목과 비교하도록 `certmap.conf` 파일을 구성할 수 있습니다 . `certmap.conf` 와 인증서 매핑에 대한 자세한 내용은 제 6 장 , " 인증서 및 키 사용 " 을 참조하십시오 .

참고 오직 SSL 인증 방법의 경우에는 인증서가 LDAP 디렉토리에 대하여 확인되므로 , 이 방법의 경우에만 `certmap.conf` 파일을 변경해야 합니다 . 서버로의 모든 연결에 대하여 클라이언트 인증이 요구되는 경우에는 이 파일을 변경할 필요가 없습니다 . 클라이언트 인증서를 사용하도록 선택한 경우 `magnus.conf` 의 `AcceptTimeout` 지시문 값을 올려야 합니다 .

Digest 인증

Sun ONE Web Server 6.1 이 LDAP 기반 또는 파일 기반 디렉토리 서버스를 사용하여 Digest 인증을 수행하도록 구성할 수 있습니다 .

Digest 인증을 사용하면 사용자 이름과 비밀번호를 보통 텍스트로 보내지 않고 사용자 이름 및 비밀번호를 기반으로 인증할 수 있습니다 . 브라우저는 MD5 알고리즘을 이용하여 Web Server 가 제공하는 사용자의 비밀번호 및 일부 정보를 사용하는 다이제스트 값을 만듭니다 .

서버가 LDAP 기반 디렉토리 서비스를 사용하여 digest 인증을 수행하는 경우 이 digest 값은 또한 Digest Authentication 플러그인을 사용하는 서버에서 컴퓨팅되며 클라이언트가 제공하는 digest 값과 비교됩니다 . 다이제스트 값이 일치하면 사용자가 인증됩니다 . 이렇게 하려면 디렉토리 서버가 보통 텍스트의 사용자 비밀번호에 액세스해야 합니다 . Sun ONE Directory Server 에는 역변환 가능한 비밀번호 플러그인이 있으며 , 이는 데이터를 암호화된 형태로 저장하여 나중에 원래의 형태로 해독할 수 있는 대칭 암호화 알고리즘을 사용합니다 . 오직 Directory Server 만이 데이터의 키를 보유합니다 .

LDAP 기반 인증의 경우 Sun ONE Web Server 6.1 에 포함된 역전환 가능 비밀번호 플러그인과 disgestauth 특정 플러그인을 사용하도록 설정해야 합니다 . 서버가 다이제스트 인증을 처리하도록 구성하려면 `dbswitch.conf` 에 있는 데이터베이스의 `digestauth` 속성을 설정해야 합니다 .

서버는 그림 에 보이는 것과 같이 지정된 ACL 방법에 기반하여 LDAP 데이터베이스에 대한 인증을 시도합니다 . ACL 방법을 지정하지 않으면 , 서버는 인증이 요구되는 경우 digest 또는 basic 을 사용하며 인증이 요구되지 않는 경우 basic 을 사용합니다 . 이것이 가장 많이 사용되는 방법입니다 .

표 9-1) Digest 인증 질문 생성

ACL 방법	인증 데이터베이스가 지원하는 다이제스트 인증	인증 데이터베이스가 지원하지 않는 다이제스트 인증
"default"	digest 및 basic	basic
지정된 사항 없음		
"basic"	basic	basic
"digest"	digest	ERROR

`method=digest` 로 설정된 ACL 을 처리하는 경우 서버는 다음과 같이 인증을 시도합니다 .

- Authorization 요청 헤더 확인 . 없는 경우 Digest 질문을 포함하는 401 응답이 생성되며 프로세스는 정지합니다 .
- Authorization 유형 확인 . Authentication 유형이 Digest 인 경우 :
  - nonce 확인. 유효하지 않은 경우 이 서버가 새 nonce를 생성하며, 401 응답이 생성되며 프로세스가 정지됩니다 . 오래 된 경우 `stale=truth` 로 설정된 401 응답이 생성되며 프로세스가 정지됩니다 .

    `magnus.conf` 파일에 있는 `DigestStalTimeout` 매개 변수의 값을 변경하여 nonce 가 새로운 상태를 유지하는 시간을 구성할 수 있습니다 . 이 파일의 위치는 *`server_root`*`/https-`*`server_name`*`/config/` 입니다 . 이 값을 설정하려면 `magnus.conf` 에 다음과 같은 줄을 추가합니다 .

    `DigestStaleTimeout` *`seconds`*

    여기에서 *seconds* 는 nonce 가 새로운 상태를 유지하는 초 단위 시간입니다 . 지정된 시간이 경과하면 nonce 가 만기되며 사용자에 대한 새로운 인증이 요구됩니다 .
  - 영역 확인. 일치되지 않는 경우 401 응답이 생성되며 프로세스가 정지됩니다.
  - 인증이 LDAP 기반인 경우 LDAP 디렉토리에 사용자가 있는지 확인하며 인증이 파일 기반인 경우 파일 데이터베이스에 사용자가 있는지 확인합니다 . 찾을 수 없는 경우 401 응답이 생성되며 프로세스가 정지됩니다 .
  - 디렉토리 서버 또는 파일 데이터베이스에서 요청 - 다이제스트 값을 가져오고 클라이언트의 요청 -digest 와 일치하는지 확인 . 일치하지 않는 경우 401 응답이 생성되며 프로세스가 정지됩니다 .

- Authorization-Info 헤더를 만들고 이를 서버 헤더에 삽입 .

Digest Authentication 플러그인 설치

LDAP 기반 디렉토리 서비스를 사용하는 digest 인증의 경우 digest 인증 플러그인을 설치해야 합니다 . 이 플러그인은 서버 측의 digest 값을 계산하고 이를 클라이언트가 제공한 digest 값과 비교합니다 . 다이제스트 값이 일치하면 사용자가 인증됩니다 .

파일 기반 인증 데이터베이스를 사용하는 경우 digest 인증 플러그인을 설치할 필요는 없습니다 .

UNIX 에 Digest Authentication 플러그인 설치

Digest Authentication 플러그인은 다음의 모두에 있는 공유 라이브러리로 구성됩니다 .

- `libdigest-plugin.lib`
- `libdigest-plugin.ldif`

UNIX 에 Digest Authentication 플러그인을 설치하려면 다음과 같이 합니다 .

1. 이 공유 라이브러리가 Sun ONE Directory Server 를 설치한 동일한 서버 컴퓨터에 있는지 확인합니다 .
2. Directory Manager 비밀번호가 올바른지 확인합니다 .
3. `libdigest-plugin.ldif` 파일의 `/path/to` 에 대한 모든 참조를 다이제스트 플러그인 공유 라이브러리를 설치한 위치로 변경합니다 .
4. 플러그인을 설치하려면 다음 명령을 입력합니다 .

   ```
   % ldapmodify -D "cn=Directory Manager" -w password -a <
   libdigest-plugin.ldif
   ```

Windows 에 Digest Authentication 플러그인 설치

Sun ONE Directory Server 가 Digest 플러그인과 함께 적절히 시작되려면 여러 개의 `.dll` 파일을 Sun ONE Web Server 설치 위치에서 Sun ONE Directory Server 컴퓨터로 복사해야 합니다 .

Windows 에 Digest Authentication 플러그인을 설치하려면 다음과 같이 합니다 .

1. 다음의 Sun ONE Web Server 설치 위치에 있는 공유 라이브러리에 액세스합니다 .

   [*server_root*]\bin\https\bin

2. 다음 파일을 복사합니다 .

- nsldap32v50.dll
- libspnr4.dll
- libplds4.dll

3. 복사한 파일을 다음 중 한 곳에 붙여 넣습니다 .
   - \Winnt\system32
   - Sun ONE Directory Server 설치 디렉토리 : [*server_root*]\bin\sldap\server

DES 알고리즘 사용을 위한 Sun ONE Directory Server 설정

다이제스트 비밀번호가 저장된 위치의 속성을 암호화하려면 DES 알고리즘이 필요합니다 .

DES 알고리즘을 사용하도록 Sun ONE Directory Server 를 설정하려면 다음과 같이 합니다 .

1. Sun ONE Directory Server Console 을 시작합니다 .
2. iDS 5.0 인스턴스를 엽니다 .
3. Configuration 탭을 선택합니다 .
4. 플러그인 옆의 + 기호를 누릅니다 .
5. DES 플러그인을 선택합니다 .
6. Add 를 선택하여 새 속성을 추가합니다 .
7. iplanetReversiblePassword 를 입력합니다 .
8. Save 를 누릅니다 .
9. Sun ONE Directory Server 인스턴스를 재시작합니다 .

참고 사용자용 iplanetReversiblePassword 속성의 다이제스트 인증 비밀번호를 설정하려면 항목에 반드시 iplanetReversiblePasswordobject 개체가 포함되어야 합니다 .

Other 인증

액세스 컨트롤 API 를 사용하여 사용자 정의 메소드를 만들 수 있습니다 .

Host-IP 용 액세스 제어 설정

Administration Server 또는 웹 사이트의 파일 및 디렉토리를 특정 컴퓨터를 이용하는 사용자만 사용할 수 있도록 설정하여 이에 대한 액세스를 제한할 수 있습니다 . 허용 또는 거부하려는 컴퓨터용 호스트 이름 또는 IP 주소를 지정합니다 . 여러 대의 컴퓨터 또는 전체 네트워크를 지정하려면 와일드카드 패턴을 사용합니다 . Host-IP 인증을 사용하는 파일 또는 디렉토리 액세스는 사용자가 알 수 없게 진행됩니다 . 사용자는 사용자 이름이나 비밀번호를 입력하지 않고 즉시 파일과 디엑토리에 액세스할 수 있습니다 .

여러 사람이 특정 컴퓨터를 사용할 수 있으므로 Host-IP 인증은 사용자 - 그룹 인증과 함께 사용할 때 더욱 효과적입니다 . 두 가지 인증 방법이 모두 사용되는 경우 액세스할 때 사용자 이름과 비밀번호가 필요합니다 .

Host-IP 인증의 경우 서버에서 DNS 를 구성할 필요가 없습니다 . Host-IP 인증을 선택한 경우 반드시 DNS 가 네트워크에서 실행되어야 하며 서버가 이를 사용하도록 구성되어야 합니다 . 서버의 DNS 는 Server Manager 의 Preferences 탭에 있는 Performance Tuning 페이지에서 사용하도록 설정할 수 있습니다 .

DNS 를 사용하도록 설정하면 서버가 DNS 조회를 수행해야 하므로 Sun ONE Web Server 의 성능이 낮아집니다 . DNS 조회가 서버 성능에 미치는 영향을 낮추려면 모든 요청의 IP 주소를 변환하는 대신 오직 액세스 제어 및 CGI 용 IP 주소만 변환합니다 . 이렇게 하려면 `obj.conf` 파일에 있는 `"AddLog fn="flex-log" name="access"` 의 `iponly=1` 로 설정합니다 .

```
AddLog fn="flex-log" name="access" iponly=1
```

액세스 제어 파일 사용

Administration Server 또는 웹 사이트의 파일이나 디렉토리에서 액세스 제어를 사용하는 경우 해당 설정은 확장자가 `.acl` 인 파일에 저장됩니다 . 액세스 제어 파일은 *install_dir*`/httpacl` 에 저장되며 *install_dir* 은 서버가 설치된 위치입니다 . 예를 들어 서버를 `/user/Sun/Servers` 에 설치한 경우 Administration Server 와 서버에 구성된 각 서버 인스턴스용 ACL 파일의 위치는 `/usr/Sun/Servers/httpacl/` 입니다 .

기본 ACL 파일 이름은 `generated-htpps-`*server-id*`.acl` 이며 , 임시 작동 파일의 이름은 `genwork-https-`*server-id*`.acl` 입니다 . Sun ONE Administration Server 를 사용하여 액세스를 구성하는 경우 이 두 파일이 만들어집니다 . 그러나 제한을 더욱 복잡하게 하려면 여러 개의 파일을 만들고 `server.xml` 파일에서 이들 파일을 참조합니다 . 또한 하루 중 시간 또는 요일을 기준으로 서버에 대한 액세스를 제한하는 등 , 파일을 편집할 때에만 사용할 수 있는 몇 가지 기능이 있습니다 .

또한 직접 `.acl` 파일을 만들고 편집하여 API 를 사용하는 액세스 제어를 사용자 정의할 수 있습니다 . 액세스 제어 API 를 사용하는 데 대한 자세한 내용은 *Programmer's Guide* 를 참조하십시오 .

액세스 제어 파일과 구문에 대한 자세한 정보는 부록 C, "ACL 파일 구문 " 을 참조하십시오 .

ACL 사용자 캐시 구성

기본적으로 Sun ONE Web Server 는 사용자 및 그룹 인증 결과를 ACL 사용자 캐시에 캐시합니다 . `magnus.conf` 파일의 `ACLCacheLifetime` 지시문을 사용하여 ACL 사용자 캐시의 유효 시간을 조정할 수 있습니다 . 캐시에 있는 항목이 참조될 때마다 시간이 계산되고 `ACLCacheLifetime` 과 비교됩니다 . 항목의 시간이 `ACLCacheLifetime`과 같거나 크면 해당 항목은 사용되지 않습니다. 기본값은 120 초입니다 . 값을 0 으로 설정하면 캐시가 Off 로 설정됩니다 . 이 값에 큰 값을 사용하면 LDAP 항목을 변경할 때마다 Sun ONE Web Server 를 다시 시작해야 합니다 . 예를 들어 이 값을 120 초로 설정하는 경우 최대 2 분까지 Sun ONE Web Server 가 LDAP 디렉토리와 동기화되지 않을 수 있습니다 . LDAP 디렉토리가 자주 변경되지 않는 경우에만 큰 값을 사용하십시오 .

`ACLUserCacheSize` 의 `magnus.conf` 매개변수를 사용하여 캐시에 유지할 항목의 최대 수를 구성할 수 있습니다 . 이 매개 변수의 기본값은 200 입니다 . 새 항목은 목록의 앞에 추가되며 목록의 끝에 있는 항목은 재활용되어 캐시가 최대 크기에 도달하면 새로운 항목이 됩니다 .

또한 `magnus.conf` 매개변수 `ACLGroupCacheSize` 를 사용하여 각 사용자 항목마다 캐시될 수 있는 그룹 구성원의 최대 수를 설정할 수 있습니다 . 이 매개 변수의 기본값은 4 입니다 . 유감스럽게도 그룹에 있는 사용자가 구성원이 아닌 경우 캐시되지 않으며 , 요청마다 여러 LDAP 디렉토리 액세스가 발생하게 됩니다 .

ACL 파일 지시문에 대한 더 자세한 내용은 Sun ONE Web Server 6.1 *NSAPI Programmer's Guide* 를 참조하십시오 .

액세스 제어 작동 원리

서버에 페이지에 대한 요청이 수신되면 서버는 ACL 파일에 있는 규칙을 사용하여 액세스를 허용할지 결정합니다 . 규칙은 요청을 보내는 컴퓨터의 호스트 이름 또는 IP 주소를 참조할 수 있습니다 . 또한 LDAP 디렉토리에 저장된 사용자 및 그룹을 참조할 수 있습니다 .

예를 들어 ACL 파일에 Administration Server(`admin-serv`) 용 기본 항목이 두 개이며 , 이에 더하여 admin-reduced 그룹에 있는 사용자가 Administration Server 의 Preferences 탭에 액세스하도록 허용하는 경우를 들 수 있습니다 .

```
version 3.0;
# The following "es-internal" rules protect files such
# as icons and images related to Sun ONE Web Server.
# These "es-internal" rules should not be modified.
  acl "es-internal";
  allow (read, list, execute,info) user = "anyone";
  deny (write, delete) user = "anyone";
# The following "default" rules apply to the entire document
# directory of Sun ONE Web Server. In this example, the rules
# are set up so that "all" users in the directory server are
# allowed to read, execute, list, and get information.
# The "all" users are not allowed to write to or delete any files.
# All clients that accesses the document directory of the web
# server will be required to submit a username and password
# since this example is using the "basic" method of
# authentication. A client must be in the directory server
# to gain access to this default directory since "anyone"
# not in the directory server is denied, and "all" in the
# directory server are allowed.
  acl "default";
  authenticate (user,group) {
     database = "default";
     method = "basic";
  };
  deny (all)
  (user = "anyone");
  allow (read,execute,list,info)
  (user = "all");
# The following rules deny access to the directory "web"
# to everyone not in the directory server and deny everyone
# in the directory server who is not in GroupB.
# Only the users in GroupB are allowed read, execute, list,
# and info permissions. GroupA can not gain access to the
# directory "web" even though (in the ACL rule below) they
# can access the directory "my_stuff". Furthermore, members
# of GroupB can not write or delete files.
  acl "path=/export/user/990628.1/docs/my_stuff/web/";
  authenticate (user,group) {
     database = "default";
     method = "basic";
```

```
  };
  deny (all)
  (user = "anyone");

  allow (read,execute,list,info)
  (group = "GroupB");

# The following rule denies everyone not in the directory
# server and denies everyone in the directory server except
# user with the ID of "SpecificMemberOfGroupB". The ACL rule
# in this setting also has a requirement that the user
# connect from a specific IP address. The IP address setting
# in the rule is optional; it has been added to for extra
# security. Also, this ACl rule has a Customized prompt
# of "Presentation Owner". This Customized prompt appears
# in the username and password dialog box in the client's
# browser.

  acl
"path=/export/user/990628.1/docs/my_stuff/web/presentation.html"
;
  authenticate (user,group) {
     database = "default";
     method = "basic";
     prompt = "Presentation Owner";
  };
  deny (all)
  (user = "anyone" or group = "my_group");
  allow (all)
  (user = "SpecificMemberOfGroupB") and
  (ip = "208.12.54.76");

# The following ACL rule denies everyone not in the directory
# server and everyone in the directory server except for
# GroupA and GroupB access to the directory ?y_stuff
  acl "path=/export/user/990628.1/docs/my_stuff/";
  authenticate (user,group) {
     database = "default";
     method = "basic";
  };
  deny (all)
  (user = "anyone");
  allow (read,execute,list,info)
  (group = "GroupA,GroupB");
```

예를 들어 사용자가 다음 URL 을 요청하는 경우 ,
`http://`*`server_name`*`/my_stuff/web/presentation.html`

Sun ONE Web Server 는 우선 전체 서버에 대한 액세스 제어를 확인합니다 . 전체 서버용 ACL 이 계속으로 설정된 경우 서버는 `my_stuff` 디렉토리용 ACL 을 확인합니다 . ACL 이 존재하면 서버는 ACL 에 있는 ACE 를 확인한 후 , 다음 디렉토리로 이동합니다 . 이 프로세스는 액세스를 거부하는 ACL 이 발견되거나 요청된 URL 에 대한 마지막 URL(이 경우에는 `presentation.html` 파일) 에 도달할 때까지 계속됩니다 .

Server Manager 를 이용하여 이 예제의 액세스 제어를 설정하려면 파일 전용 또는 파일로 유도되는 각 리소스용 ACL 을 만들 수 있습니다 . 즉 , 전체 서버용 1 개 , `my_stuff` 디렉토리용 1 개 , `my_stuff/web` 디렉토리용 1 개 및 해당 파일용 1 개를 만들 수 있습니다 .

참고 일치되는 ACL 이 하나 이상인 경우 서버는 일치되는 마지막 ACL 문을 사용합니다 . 일치되는 ACL 이 `uri` 이므로 `default` ACL 은 무시됩니다 .

액세스 제어 설정

여기에서는 웹 사이트의 파일 또는 디렉토리에 대한 액세스를 제한하는 프로세스에 대하여 설명합니다 . 모든 서버에 대한 전역 액세스 제어 규칙을 만들 수 있으며 , 또한 특정 서버에 대한 개별 규칙을 만들 수 있습니다 . 예를 들어 인력관리 부서에서는 모든 인증된 사용자가 자신의 연봉 데이터를 볼 수 있으나 오직 인력관리 부서의 연봉을 담당하는 직원만 데이터를 액세스할 수 있도록 제한하는 ACL 을 만들 수 있습니다 .

Administration Server 를 통하여 모든 서버에 대한 액세스를 전역적으로 제어할 수 있습니다 . 각 옵션에 대한 설명은 다음 부분 Access Control 옵션 선택에서 자세히 설명합니다 .

참고 전역 액세스 제어를 만들기 전에 반드시 분산 관리를 구성하고 사용해야 합니다 .

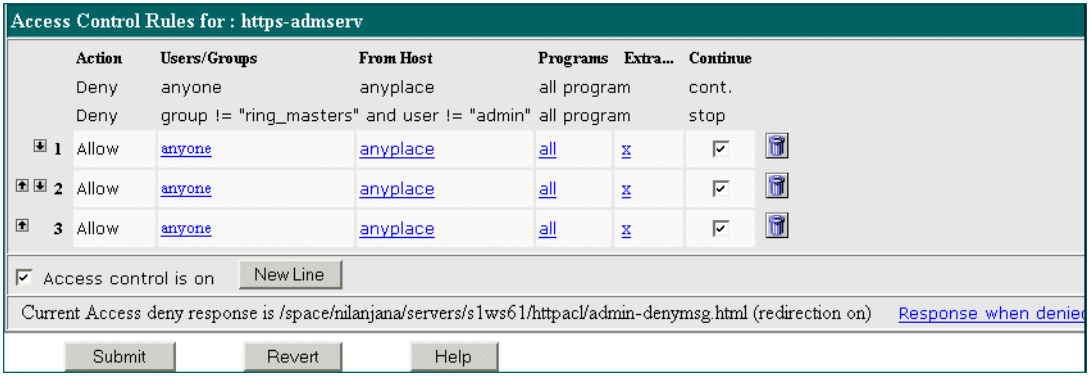
전역적 액세스 제어 설정

모든 서버에 대한 액세스 제어를 전역적으로 만들고 편집하려면 다음과 같이 합니다 .

1. Administration Server 에 액세스하고 Global Settings 탭을 선택합니다 .
2. Restrict Access 링크를 누릅니다 .
3. 드롭 다운 목록에서 administration server(https-admserv) 를 선택합니다 .
4. Create ACL 을 누르고 Go 버튼을 누릅니다 .

   uri=/https-admserv/ 용 Access Control Rules 페이지가 표시됩니다 .

   Access Control Rules 페이지



   Administration Server 에는 편집할 수 없는 기본 액세스 제어 규칙이 두 줄 있습니다 .

5. Access 제어가 아직 선택되지 않았으면 ON 으로 선택합니다 .
6. 표의 하단에 기본 ACL 규칙을 추가하려면 New Line 버튼을 누릅니다 .

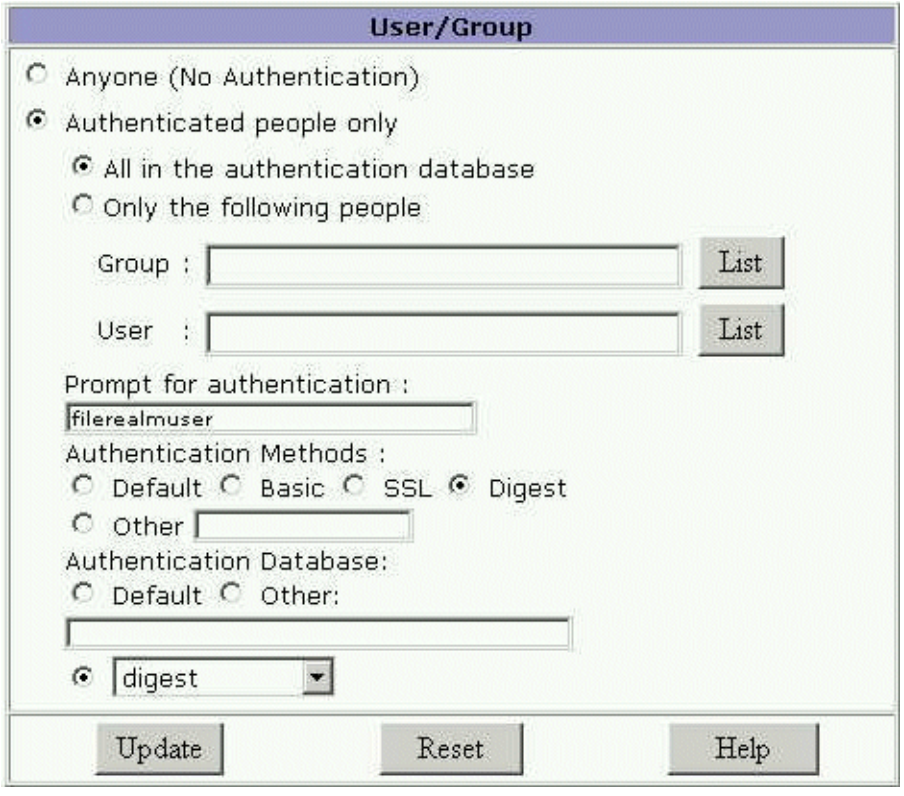
   액세스 제어 제한이 앞에 오도록 액세스 제어 제한을 스왑하려면 위쪽 화살표 그림을 누릅니다 .

   액세스 제어 제한이 뒤에 오도록 액세스 제어 제한을 스왑하려면 아래쪽 화살표 그림을 누릅니다 .

7. Users/Groups 열에서 임의의 항목을 누릅니다 .

   User/Group 페이지가 아래 창에 표시됩니다 .

   User/Group 페이지



8. 액세스를 허용할 사용자 및 그룹을 선택하고 Update 를 누릅니다 .

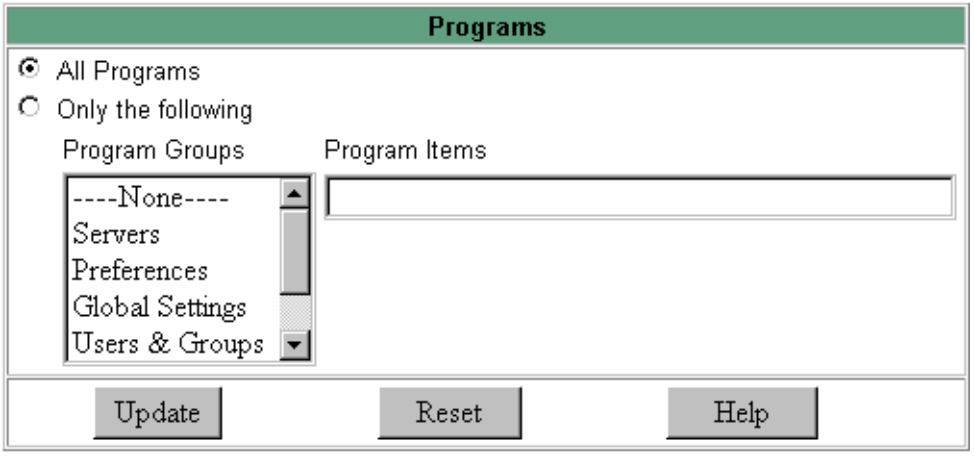
   List for Group and User 를 누르면 선택할 수 있는 목록이 표시됩니다 .

9. From Host 열에서 아무 곳이나 누릅니다 .

10. 액세스가 허용된 Host Name 및 IP Address 를 입력한 후 Update 를 누릅니다 .

11. Programs 열의 All Programs 를 누릅니다 .

    Programs



12. 액세스를 허용할 Program Groups를 선택하거나 Program Items 필드에 특정 파일 이름을 입력합니다 .

13. (선택) 사용자 정의 ACL 표현식을 추가하려면 Extra 열 아래의 x 를 누릅니다 .

14. Continue 열이 아직 선택되지 않았으면 이 열의 선택란을 선택합니다 .

    서버는 사용자의 액세스가 허용되었는지 결정하기 전에 다음 줄을 확인합니다 . 여러 줄을 만드는 경우에는 가장 일반적인 제한에서 가장 국부적인 제한으로 진행합니다 .

15. (선택) 거부된 경우 사용자를 다른 URL 또는 URI 로 재설정하려면 Response 를 누릅니다 .

16. 절대 URL 또는 상대 URI 에 대한 경로를 입력하고 Update 를 누릅니다 .

17. Submit 를 눌러 새 액세스 제어 규칙을 ACL 파일에 저장합니다 .

참고 Revert 를 누르면 지금 만든 모든 설정이 제거됩니다 .

서버 인스턴스용 액세스 제어 설정

Server Manager 를 사용하여 특정 서버 인스턴스용 액세스 제어를 만들거나 편집 또는 삭제할 수 있습니다 .

참고 삭제하는 경우 ACL 파일에서 ACL 규칙을 모두 삭제하면 안 됩니다. 서버를 시작하려면 ACL 규칙을 한 개 이상 포함하는 ACL 파일이 적어도 하나 이상 있어야 합니다 . ACL 규칙을 모두 삭제하고 서버를 재시작하면 구문 오류가 발생합니다 .

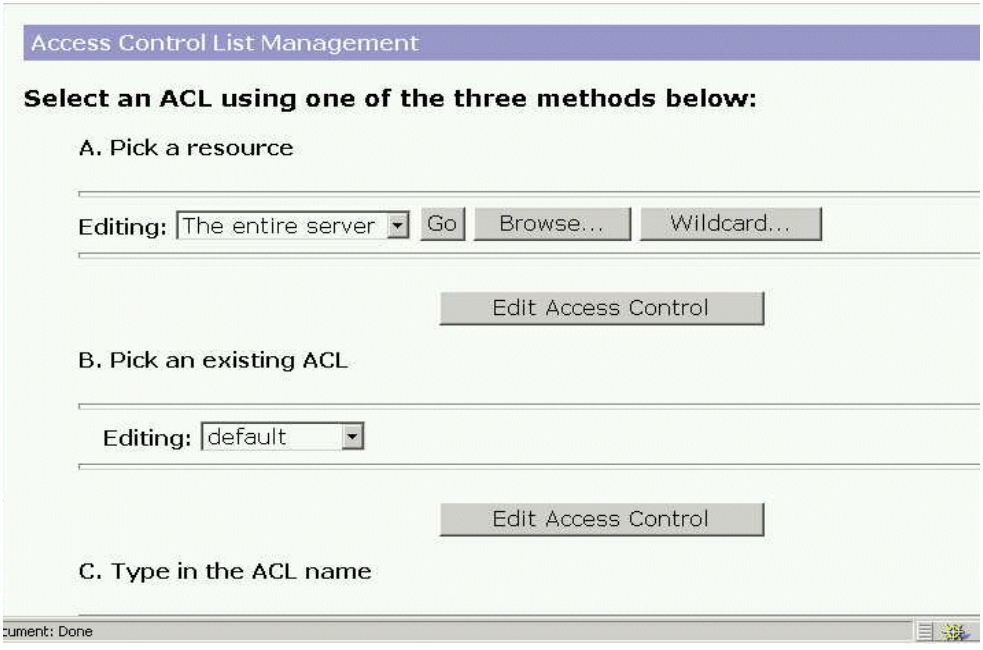
서버 인스턴스용 액세스 제어를 만들려면 다음과 같이 합니다 .

1. Server Manager 에 액세스하고 ACL 을 만들거나 편집하려는 서버 인스턴스를 선택합니다 .
2. Server Manager 에서 Preferences 탭을 선택합니다 .
3. Restrict Access 링크를 누릅니다 .
4. Option 열 아래에서 다음 중 한 가지를 선택합니다 .
   - ACL 파일 위치 추가 및 입력합니다 .
   - 드롭다운 목록에서 ACL 파일 편집 및 선택합니다 .

- 드롭다운 목록에서 ACL 파일 삭제 및 선택합니다 .

Access Control List Management 페이지에 제공되는 옵션은 세 가지입니다 .

Access Control List Management 페이지



5. 다음 중 한 가지를 선택합니다 .
   - 파일 또는 디렉토리용 와일드카드 패턴(`*.html` 등)을 지정하는 리소스를 선택하거나 , 제한할 디렉토리 또는 파일 이름을 선택하거나 , 파일 또는 디렉토리를 찾습니다 .
   - 사용하도록 설정한 모든 ACL 목록에서 기존 ACL 을 선택합니다 . 사용하도록 설정하지 않은 기존 ACL 은 목록에 표시되지 않습니다 .

- **Enter the ACL name** 을 사용하여 이름이 지정된 ACL 을 만들 수 있습니다. 이 옵션은 ACL 파일에 익숙한 경우에만 사용하십시오 . 이름이 지정된 ACL 을 리소스에 적용하려는 경우에는 `obj.conf` 를 직접 편집해야 합니다 .

사용할 수 있는 리소스 와일드카드는 표 9-2 에 설명된 것과 같습니다 .

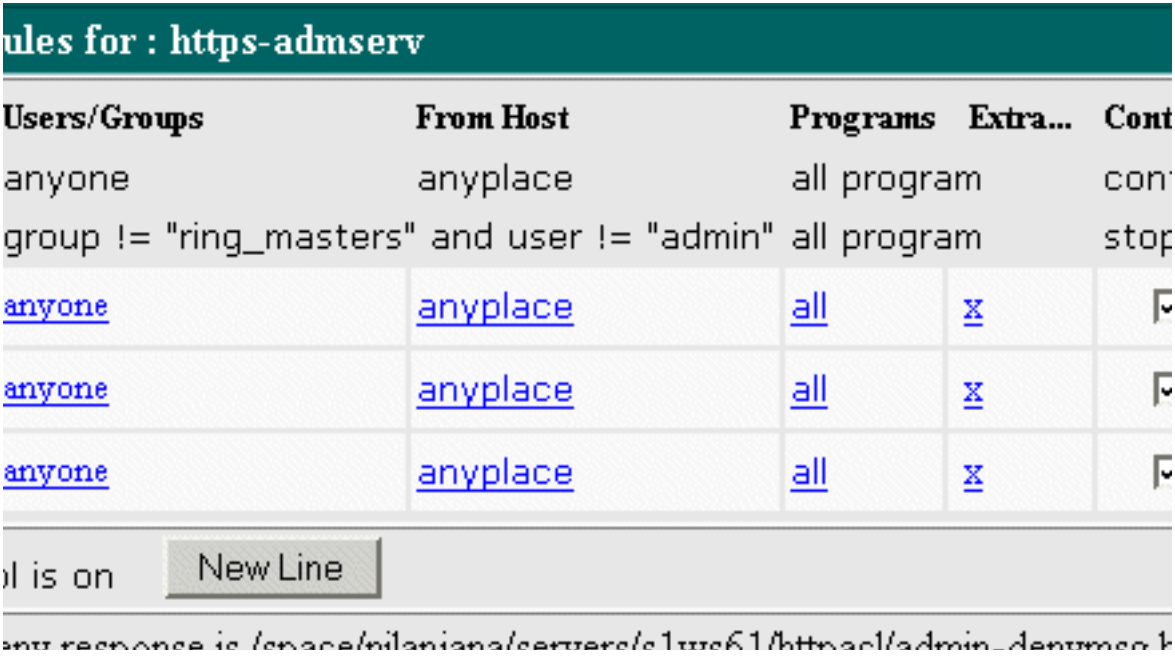
표 9-2) 서버 리소스 와일드카드

리소스 와일드카드	의미
default	설치시 만들어진 이름이 지정된 ACL 로 쓰기 액세스를 제한하므로 오직 LDAP 디렉토리에 있는 사용자만 문서를 게시할 수 있습니다 .
Entire Server	전체 웹 사이트에 대한 액세스를 결정하는 규칙 세트로 실행하는 모든 가상 서버가 포함됩니다 . 가상 서버에 대한 액세스를 제한하려면 해당 문서 루트의 경로를 지정합니다 .
`/usr/sun/server4/docs/cgi-bin/*`	`cgi-bin` 디렉토리의 모든 파일 및 디렉토리에 대한 액세스를 제어합니다 . 반드시 절대 경로를 지정해야 합니다 . Windows 의 경우 경로에 반드시 드라이브 문자가 포함되어야 합니다 .
`uri="sales"`	문서 루트의 `sales` 디렉토리에 대한 액세스를 제어합니다 . URI 를 지정하려면 이름이 지정된 ACL 을 만듭니다 .

6. Edit Access Control 을 누릅니다 .

   Access Control Rules: (서버 인스턴스) 가 표시됩니다 .

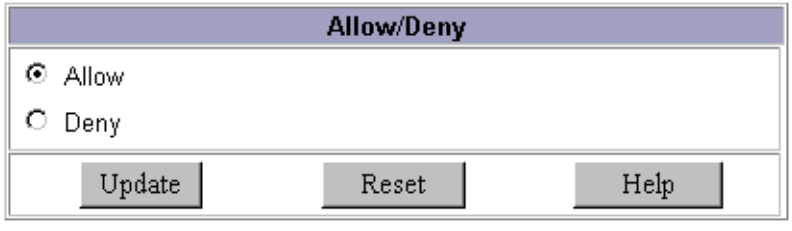
   Access Control Rules 페이지



7. Access 제어가 아직 선택되지 않았으면 ON 으로 선택합니다 .
8. 이 서버 인스턴스용 ACL을 만들거나 편집하려면 Action 열의 Deny를 누릅니다.

   Allow/Deny 페이지가 아래의 창에 표시됩니다 .

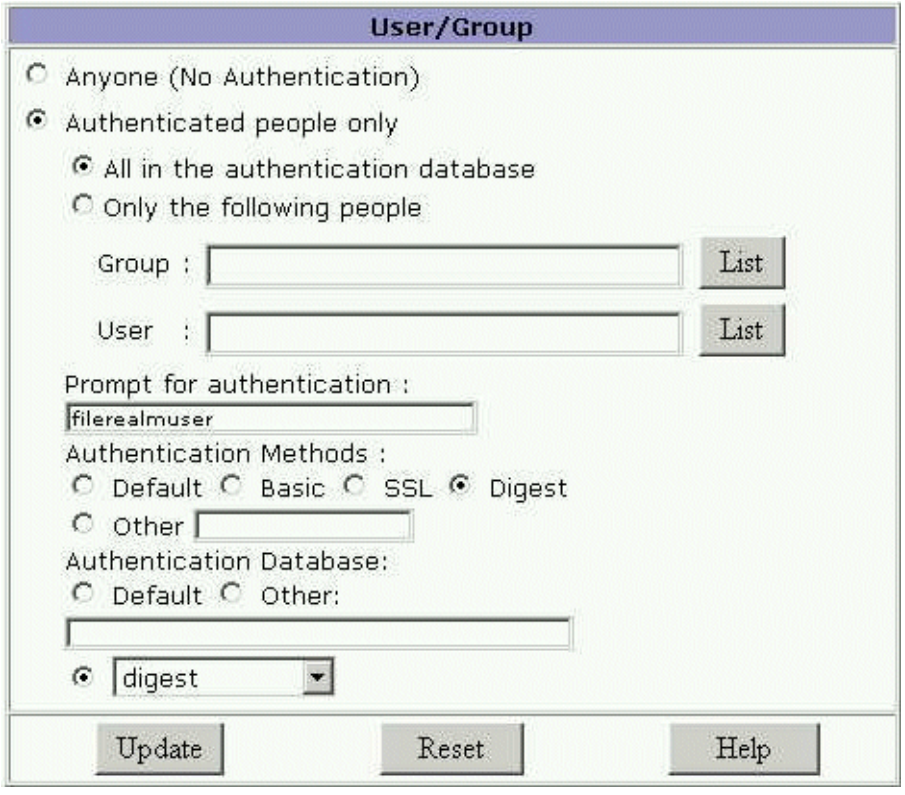
   Allow/Deny 페이지



9. Allow 가 아직 기본값으로 선택되지 않았으면 선택하고 Update 를 누릅니다 .
10. Users/Groups 열에서 임의의 항목을 누릅니다 .

    User/Group 페이지가 아래 창에 표시됩니다 .

    User/Group 페이지



11. 액세스를 허용할 사용자 및 그룹을 선택하고 Update 를 누릅니다 .

    List for Group and User 를 누르면 선택할 수 있는 목록이 표시됩니다 .
12. From Host 열에서 아무 곳이나 누릅니다 .
13. 액세스가 허용된 Host Name 및 IP Address 를 입력한 후 Update 를 누릅니다 .

14. Rights 열의 항목을 모두 누릅니다 .

    Access Rights 페이지

15. 다음 중 한 가지를 선택하고 Update 를 누릅니다 .
    - All Access Rights
    - Only the following rights - 사용자에게 적절한 모든 권한을 선택
16. (선택) 사용자 정의 ACL 표현식을 추가하려면 Extra 열 아래의 x 를 누릅니다 .
17. Continue 열이 아직 선택되지 않았으면 이 열의 선택란을 선택합니다 .

    서버는 사용자의 액세스가 허용되었는지 결정하기 전에 다음 줄을 확인합니다 . 여러 줄을 만드는 경우에는 가장 일반적인 제한에서 가장 국부적인 제한으로 진행합니다 .
18. (선택) 거부된 경우 사용자를 다른 URL 또는 URI 로 재설정하려면 Response 를 누릅니다 .
19. 절대 URL 또는 상대 URI 에 대한 경로를 입력하고 Update 를 누릅니다 .
20. Submit 를 눌러 새 액세스 제어 규칙을 ACL 파일에 저장합니다 .

참고 Revert 를 누르면 지금 만든 모든 설정이 제거됩니다 .

21. 액세스 제어를 설정하려는 각 서버 인스턴스마다 위의 단계를 반복합니다 .

22. 작업이 완료되면 Apply 를 누릅니다 .

23. hard start/restart 또는 dynamically apply 를 선택합니다 .

ACL 설정은 가상 서버 단위로 사용 설정할 수 있습니다 . 이에 대한 방법은 " 가상 서버용 액세스 제어 목록 편집 " 페이지 223 을 참조하십시오 .

Access Control 옵션 선택

다음에서는 액세스 제어를 설정할 때 선택할 수 있는 다양한 옵션에 대하여 설명합니다 . Administration Server 의 경우 첫 두 줄은 기본으로 설정되며 편집할 수 없습니다 .

작동 설정

요청이 액세스 제어 규칙과 일치할 때 서버의 작동을 지정할 수 있습니다 .

- **Allow**: 사용자 또는 시스템이 요청된 리소스에 액세스할 수 있습니다 .
- **Deny**: 사용자 또는 시스템이 리소스에 액세스할 수 없습니다 .

서버는 ACE(access control expression) 목록 전체를 확인하여 액세스 권한을 판단합니다 . 예를 들어 첫 번째 ACE 는 보통 모든 사용자를 거부합니다 . 첫 번째 ACE 가 "continue" 로 설정된 경우 서버는 목록의 두 번째 ACE 를 확인하며 , 일치되는 경우 다음 ACE 를 사용합니다 . continue 가 선택되지 *않은* 경우 리소스에 대한 모든 사용자의 액세스가 거부될 것입니다 . 서버는 일치되지 않는 ACE 가 발견되거나 일치되지만 continue 가 설정되지 않은 ACE 를 발견할 때까지 계속합니다 . 마지막으로 일치되는 ACE 에 따라 액세스의 허용 또는 거부가 결정됩니다 .

사용자 및 그룹 지정

사용자 및 그룹 인증을 사용하면 사용자가 액세스 제어 규칙에 지정된 리소스에 액세스하기 전에 사용자 이름 및 비밀번호를 입력하라는 프롬프트가 표시됩니다 .

Sun ONE Web Server 는 Sun ONE Directory Server 등의 LDAP 서버 또는 내부 파일 기반 인증 데이터베이스에 저장된 사용자 및 그룹 목록을 확인합니다 .

데이터베이스에 있는 모든 사용자의 액세스를 허용 또는 거부할 수 있으며 , 와일드카드 패턴을 사용하여 특정 사용자를 허용 또는 거부할 수 있습니다 . 또한 사용자 및 그룹 목록에서 허용 또는 거부할 사용자를 선택할 수 있습니다 .

- **Anyone(No Authentication)** 은 기본값으로 모든 사용자가 사용자 이름 및 비밀번호를 입력하지 않고 리소스에 액세스할 수 있습니다 . 그러나 호스트 이름 또는 IP 주소 등의 기타 설정에 따라 액세스를 거부할 수 있습니다 . Administration Server 의 경우 분산 관리로 지정한 관리 그룹의 모든 사용자가 페이지에 액세스할 수 있습니다 .
- **Authenticated people only**
  - **All in the authentication database** 는 데이터베이스에 항목이 있는 임의 사용자를 일치시킵니다 .
  - **Only the following people** 의 경우 일치할 사용자 및 그룹을 지정할 수 있습니다 . 사용자 또는 사용자 그룹의 목록을 만들 수 있으며 각 항목을 쉼표로 분리하거나 , 와일드카드 패턴을 사용할 수 있습니다 . 또는 데이터베이스에 저장된 사용자 및 그룹 목록에서 선택할 수 있습니다 . **Group** 의 경우 지정한 그룹의 모든 사용자를 검색합니다 . **User** 는 지정한 개별 사용자를 검색합니다 . Administration Server 의 경우 사용자는 반드시 분산된 관리용으로 지정한 관리 그룹에 속해야 합니다 .
- **Prompt for authentication** 을 사용하면 인증 대화 상자에 표시되는 메시지 텍스트를 입력할 수 있습니다 . 이 텍스트를 사용하여 사용자가 입력해야 할 것을 설명할 수 있습니다 . 운영 체제에 따라 사용자는 프롬프트의 첫 40 자 정도만 보게 될 수 있습니다 . Netscape Navigator 및 Netscape Communicator 는 사용자 이름과 암호를 캐시하고 이를 프롬프트 텍스트에 연결합니다 . 사용자가 동일한 프롬프트를 가지는 서버의 파일 및 디렉토리에 액세스하는 경우에는 사용자 이름과 비밀번호를 다시 입력하지 않아도 됩니다 . 특정 파일 및 디렉토리에 대하여 사용자가 인증하기를 원하는 경우 간단히 해당 리소스에 대한 ACL 용 프롬프트를 변경하면 됩니다 .
- **Authentication Methods** 에는 서버가 클라이언트에서 인증 정보를 가져올 때 사용하는 메서드를 지정합니다 . Administration Server 의 경우 오직 Basic 인증 방법만 제공됩니다 .
  - **Default**는 `obj.conf`에 지정한 기본 방법을 사용하거나, `obj.conf`에 설정이 없는 경우에는 "Basic" 을 사용합니다 . Default 를 선택하는 경우 ACL 규칙은 ACL 파일에 메소드를 지정하지 않습니다 . Default 를 선택하면 `obj.conf` 파일에서 한 줄만 편집하면 모든 ACL 에 대한 메서드를 쉽게 변경할 수 있습니다 .
  - **Basic**은 HTTP 메소드를 사용하여 클라이언트에서 인증 정보를 가져옵니다. 서버용으로 암호화를 사용하는 경우 오직 사용자 이름과 암호만 암호화됩니다 .

- **SSL** 은 클라이언트 인증서를 사용하여 사용자를 인증합니다 . 이 방법을 사용하려면 반드시 서버용에 SSL 을 사용해야 합니다 . 암호화를 사용하는 경우 Basic 과 SSL 방법을 조합할 수 있습니다 .
- **Digest** 는 사용자 이름과 비밀번호를 보통의 텍스트로 송신하지 않고 브라우저가 사용자 이름과 비밀번호를 기준으로 인증할 수 있는 방법을 제공하는 인증 메커니즘을 사용합니다 . 브라우저는 MD5 알고리즘을 이용하여 Web Server 가 제공하는 사용자의 비밀번호 및 일부 정보를 사용하는 다이제스트 값을 만듭니다 . 다이제스트 값은 또한 Digest Authentication 플러그인을 사용하는 서버 측에서도 계산되며 이 값은 클라이언트가 제공하는 다이제스트 값과 비교됩니다 .
- **Other** 는 액세스 컨트롤 API 를 사용하여 만든 사용자 정의 메소드를 사용합니다 .

- **Authentication Database** 에서 서버가 사용자를 인증하는 데 사용하는 데이터베이스를 선택할 수 있습니다 . 이 옵션은 오직 Server Manager 를 통하여만 사용할 수 있습니다 . Default 를 선택하는 경우 서버는 기본으로 구성된 디렉토리 서비스에서 사용자 및 그룹을 찾습니다 . 개별 ACL 를 구성하여 서로 다른 데이터베이스에 사용하려는 경우 Other 를 선택하고 드롭 다운 목록에서 데이터베이스를 선택합니다 . 기본이 아닌 데이터베이스와 LDAP 디렉토리는 이미 *server_root*`/userdb/dbswitch.conf` 에 지정되어 있어야 합니다 . Oracle 또는 Informix 등의 사용자 정의 데이터베이스용 액세스 제어 API 를 사용하는 경우 Other 를 선택하고 데이터베이스 이름을 입력합니다 .

송신 호스트 지정

요청을 보내는 컴퓨터를 기준으로 Administration Server 또는 웹 사이트에 대한 액세스를 제한할 수 있습니다 .

- **Anyplace** 는 모든 사용자 및 시스템의 엑세스를 허용
- **Only from** 은 특정 Host Name 또는 IP Address 로 액세스 제한

Only from 옵션을 선택하면 Host Names 또는 IP Address 필드에 와일드카드 패턴 또는 쉼표로 분리된 목록을 입력합니다 . 호스트 이름을 기준으로 제한하는 것이 IP 주소를 기준으로 하는 것 보다 유연성이 많습니다 . 사용자의 IP 주소가 변경되는 경우 목록을 업데이트할 필요가 없습니다 . 그러나 IP 주소를 기준으로 제한하는 것이 더욱 안전한데 , 연결된 클라이언트에 대한 DNS 조회가 실패하는 경우 호스트 이름 제한은 사용할 수 없습니다 .

컴퓨터의 호스트 이름 또는 IP 주소를 검색하는 와일드카드 패턴에는 오직 * 와일드카드만 사용할 수 있습니다 . 예를 들어 , 특정 도메인에 있는 모든 컴퓨터를 허용 또는 거부하려면 *.sun.com 과 같이 해당 도메인의 모든 호스트에 일치하는 와일드카드 패턴을 입력합니다 . Administration Server 에 액세스하는 수퍼유저용으로 다른 호스트 이름 및 IP 주소를 설정할 수 있습니다 .

호스트 이름의 경우 * 는 반드시 이름의 구성요소 전체로 대체되어야 합니다 . 즉 , *.sun.com은 사용 가능하지만 *users.sun.com은 사용할 수 없습니다. 호스트 이름에 * 이 있는 경우 반드시 가장 왼쪽 문자이어야 합니다 . 예를 들어 *.sun.com 은 사용 가능하지만 users.*.com 은 사용할 수 없습니다 .

IP 주소의 경우 * 은 반드시 주소의 전체 바이트로 대체되어야 합니다 . 예를 들어 198.95.251.*는 사용 가능하지만 198.95.251.3*는 사용할 수 없습니다. IP 주소에 *가 있는 경우 반드시 가장 오른쪽 문자이어야 합니다. 예를 들어 198.*는 사용 가능하지만 198.*.251.30 은 사용할 수 없습니다 .

프로그램에 대한 액세스 제한

프로그램에 대한 액세스는 오직 Administration Server 에 의하여 제한될 수 있습니다 . 프로그램에 대한 액세스를 제한하면 오직 지정된 사용자만 Server Manager 페이지를 볼 수 있으며 페이지가 해당 서버용으로 구성되었는지 판단할 수 있습니다 . 예를 들어 일부 관리자에게 Administration Server 의 Users & Groups 부분을 구성할 수는 있으나 Global Settings 에는 액세스할 수 없도록 설정할 수 있습니다 .

서로 다른 사용자가 서로 다른 기능 영역에 액세스하도록 구성할 수 있습니다 . 사용자가 몇 가지 선택된 기능 영역에 액세스하도록 설정되고 해당 사용자가 로그인하면 , 오직 해당 사용자에게 액세스를 허용한 기능 영역의 Administration Server 만 볼 수 있습니다 .

- **All Programs** 는 모든 프로그램에 대한 액세스를 허용 또는 거부합니다 . 기본적으로 관리자는 서버의 모든 프로그램에 액세스할 수 있습니다 .
- **Only the following Program Groups** 에서 사용자가 액세스할 수 있는 프로그램을 지정할 수 있습니다 . 드롭 다운 목록에서 프로그램을 선택합니다 . Control 키를 누른 채 그룹을 누르면 프로그램 그룹을 여러 개 선택할 수 있습니다 . 다음 프로그램 그룹에 대한 액세스를 제한할 수 있습니다 .
  - None (기본값)
  - Servers
  - Preferences

- Global Settings
- Users & Groups
- Security
- Cluster Mgmt

예를 들어 Preferences 및 Global Settings 등의 Program Groups 목록은 Administration Server 의 탭이며 해당 페이지에 대한 액세스를 나타냅니다 . 관리자가 Administration Server 에 액세스하면 서버는 사용자 이름 , 호스트 및 IP 를 사용하여 표시할 수 있는 페이지를 결정합니다 .

- **Program Items** 에서는 Program Item 필드에 페이지 이름을 입력하여 프로그램에 있는 특정 페이지에 대한 액세스를 제어할 수 있습니다 .

액세스 권한 설정

액세스 권한은 오직 Server Manager 가 서버 인스턴스에 대하여 설정합니다 . 액세스 권한은 웹 사이트의 파일 및 디렉토리에 대한 액세스를 제한합니다 . 액세스 권한의 허용 또는 거부에 더하여 부분적인 액세스 권한을 허용 또는 거부하는 규칙을 지정할 수 있습니다 . 예를 들어 사용자에게 파일에 대한 읽기 전용 액세스 권한을 부여하여 정보를 볼 수 있으나 파일을 변경할 수 없도록 합니다 .

- **All Access Rights** 는 기본값으로 모든 권한을 허용 또는 거부합니다 .
- Only the following rights 에서는 허용 또는 거부할 권한을 조합하여 선택할 수 있습니다 .
  - **Read**는 HTTP 메소드 GET, HEAD, POST 및 INDEX를 포함하여 파일을 볼 수 있도록 허용합니다 .
  - **Write** 는 HTTP 메소드 PUT, DELETE, MKDIR, RMDIR 및 MOVE 를 포함하여 파일을 변경하거나 삭제할 수 있도록 허용합니다 . 파일을 삭제하려면 사용자에게 반드시 쓰기 및 삭제 권한이 있어야 합니다 .
  - **Execute**는 사용자가 CGI 프로그램, Java 애플릿 및 에이전트 등의 서버측 애플리케이션을 실행할 수 있도록 허용합니다 .
  - **Delete** 는 쓰기 권한을 가진 사용자가 파일 또는 디렉토리를 삭제할 수 있도록 허용합니다 .
  - **List** 는 `index.html` 파일을 포함하지 않은 디렉토리의 파일 목록에 액세스할 수 있도록 허용합니다 .
  - **Info**는 사용자가 `http_head` 등의 URI에 대한 정보를 받을 수 있도록 허용합니다 .

사용자 정의 표현식 작성

ACL 용 사용자 정의 표현식을 입력할 수 있습니다 . 오직 ACL 파일의 구문과 구조에 익숙한 경우에만 이 옵션을 선택하십시오 . ACL 파일을 편집하거나 사용자 정의 표현식을 만들 때에만 사용할 수 있는 몇 가지 기능이 있습니다 . 예를 들어 하루 중 시간 , 요일 또는 이 둘 모두를 기준으로 서버에 대한 액세스를 제한할 수 있습니다 .

하루 중 시간 및 요일을 기준으로 액세스를 제한하는 사용자 정의 표현식의 예는 다음과 같습니다 . 이 예에서는 LDAP 디렉토리에 두 개의 그룹이 있는 것으로 가정하며 , "regular" 그룹은 월요일에서 금요일까지 오전 8:00 에서 오후 5:00 사이에 액세스할 수 있습니다 . "critical" 그룹은 항상 액세스할 수 있습니다 .

```
allow (read)
{
   (group=regular and dayofweek=?on,tue,wed,thu,fri?;
   (group=regular and (timeofday>=0800 and timeofday<=1700));
   (group=critical)
}
```

유효한 구문과 ACL 파일에 대한 자세한 내용은 부록 C, "ACL 파일 구문 " 과 "obj.conf 내의 ACL 파일 참조 " 페이지 464 를 참조하십시오 .

엑세스 제어 사용 중지

"Access control is on" 이라는 옵션을 선택하지 않으면 ACL 의 기록을 삭제할 것인지 묻는 프롬프트가 표시됩니다 . 확인을 누르면 서버는 ACL 파일에서 해당 리소스용 ACL 항목을 삭제합니다 .

ACL 을 사용하지 않도록 설정하는 경우 `generated-https-server-id.acl` 파일의 각 ACL 줄 앞에 # 기호를 삽입하여 이를 주석으로 만듭니다 .

Administration Server 에서 특정 서버 인스턴스에 대한 액세스 제어를 만들어 사용하며 기타 서버에 대하여는 사용하지 않도록 (기본값) 할 수 있습니다 . 예를 들어 Administration Server 의 Server Manager 페이지에서 모든 액세스를 거부할 수 있습니다 . 기타 서버는 기본적으로 분산 관리를 사용하며 액세스 제어는 사용하지 않으므로 관리자는 다른 서버에 액세스하여 구성할 수 있으나 Administration Server 는 구성할 수 없습니다 .

참고 이 액세스 제어는 분산 관리용으로 설정된 관리자 그룹에 속한 사용자에 추가되는 기능입니다 . Administration Server 는 우선 해당 사용자 (수퍼유저 아님) 가 관리자 그룹에 있는지 확인한 후 , 액세스 제어 규칙을 평가합니다 .

액세스가 거부된 경우의 응답

Sun ONE Web Server 에는 액세스가 거부된 경우 "FORBIDDEN: Your client is not allowed access to the restricted object" 이라는 기본 메시지가 제공됩니다 . 액세스가 거부된 경우 다른 응답 메시지를 선택할 수 있습니다 . 또한 각 액세스 제어 개체마다 서로 다른 메시지를 만들 수 있습니다 .

특정 ACL 용으로 송신하는 메시지를 변경하려면 다음과 같이 합니다 .

1. ACL 페이지의 Response when denied 링크를 누릅니다 .
2. 아래 창에서 Respond with the following file 을 선택합니다 .
3. 절대 URL 또는 상대 URI 에 대한 경로를 입력하고 Update 를 누릅니다 .

   반드시 제설정된 URL 또는 URI 에 대한 액세스 권한이 있어야 합니다 .
4. Update 를 누릅니다 .
5. 위 창에 있는 Submit 을 눌러 액세스 제어 규칙을 제출합니다 .

서버의 영역에 대한 액세스 제한

여기에서는 웹 서버 및 해당 컨텐츠에 대한 액세스를 제한하는 데 일반적인 제한에 대하여 설명합니다 . 각 절차에 대하여 필요한 작업을 단계별로 설명했으나 , " 서버 인스턴스용 액세스 제어 설정 " 페이지 192 에 설명한 모든 단계를 완료해야 합니다 .

여기에서는 다음 항목에 대해 설명합니다 .

- 전체 서버에 대한 액세스 제한
- 디렉토리 (경로) 에 대한 액세스 제한
- URI(경로) 에 대한 액세스 제한
- 파일 유형에 대한 액세스 제한하루 중 시간을 기준으로 액세스 제한
- 보안을 기준으로 액세스 제한

전체 서버에 대한 액세스 제한

하위 도메인의 컴퓨터에서 서버에 액세스하도록 호출된 그룹의 사용자에게 액세스를 허용할 하는 경우가 있습니다 . 예를 들어 회사 부서의 서버의 경우 사용자가 오직 네트워크의 특정 도메인의 컴퓨터에서 액세스하도록 할 수 있습니다 .

서버 인스턴스에 대한 액세스 제어를 설정하는 방법을 사용하여 다음과 같이 설정할 수 있습니다 .

1. Server Manager 를 사용하여 서버 인스턴스를 선택합니다 .
2. Preferences 탭을 선택합니다 .
3. Restrict Access 링크를 누릅니다 .
4. 편집할 ACL 파일을 선택합니다 .
5. 전체 서버 리소스를 선택하고 Edit Access Control 을 누릅니다 .
6. 모두의 액세스를 거부할 새 규칙을 추가합니다 .
7. 특정 그룹의 액세스를 허용하는 다른 규칙을 새로 추가합니다 .
8. 허용할 컴퓨터의 호스트 이름용 와일드카드 패턴을 입력합니다 .

   예 : `*.employee.sun.com`
9. Continue 의 선택을 취소합니다 .
10. 변경 사항을 Submit 및 Apply 합니다 .

디렉토리 (경로) 에 대한 액세스 제한

사용자가 디렉토리 , 또는 그룹의 소유자가 제어하는 해당 하위 디렉토리 및 파일에서 응용 프로그램을 읽거나 실행하도록 허용할 수 있습니다 . 예를 들어 프로젝트 관리자는 프로젝트 팀이 검토할 수 있도록 상태 정보를 업데이트할 수 있습니다 .

서버의 디렉토리에 대한 액세스를 제한하려면 서버 인스턴스에 대한 액세스 제어 설정 방법을 이용합니다 .

1. Server Manager 를 사용하여 서버 인스턴스를 선택합니다 .
2. Preferences 탭을 선택합니다 .
3. Restrict Access 링크를 누릅니다 .
4. 편집할 ACL 파일을 선택합니다 .
5. Pick a Resource 부분을 찾아 제한하려는 디렉토리를 선택합니다 .

   서버의 문서 루트에 있는 디렉토리가 표시됩니다 . 선택하면 Editing 드롭 다운 목록에 해당 디렉토리의 절대 경로가 표시됩니다 .

참고 서버 루트의 모든 파일을 보려면 Options 를 누르고 List files as well as directories 를 선택합니다 .

6. Edit Access Control 을 누릅니다 .
7. 새 규칙을 만들고 기본 값을 유지하여 기타 위치로부터의 사용자 액세스를 거부합니다 .
8. 특정 그룹의 사용자에게 오직 읽기 및 실행 권한만 허용하는 새 규칙을 만듭니다 .
9. 특정 사용자에게 모든 권한을 허용하는 세 번째 줄을 만듭니다 .
10. 두 번째 및 세 번째 줄의 Continue 의 선택을 해제하고 Update 를 누릅니다 .
11. 변경 사항을 Submit 및 Apply 합니다 .

파일 또는 디렉토리의 절대 경로가 docroot 디렉토리에 만들어집니다 . ACL 파일의 항목은 다음과 같이 표시됩니다 .

```
acl "path=d:\sun\suitespot\docroot1\sales/";
```

URI(경로) 에 대한 액세스 제한

URI 를 사용하여 웹 서버에 있는 단일 사용자의 컨텐츠에 대한 액세스를 제어할 수 있습니다 . URI 는 서버의 문서 루트 디렉토리에 상대적인 경로 및 파일입니다 . 서버의 컨텐츠를 모두 (예를 들어 디스크 공간) 자주 옮기거나 이름을 변경하는 경우 URI 를 사용하면 해당 컨텐츠를 쉽게 관리할 수 있습니다 . 또한 추가의 문서 루트가 있는 경우 액세스 제어를 처리하는 좋은 방법입니다 .

URI 에 대한 액세스를 제한하려면 서버 인스턴스에 대한 액세스 제어 설정 방법을 이용합니다 .

1. Server Manager 를 사용하여 서버 인스턴스를 선택합니다 .
2. Preferences 탭을 선택합니다 .
3. Restrict Access 링크를 누릅니다 .
4. ACL 이름 부분의 Type 에 제한하려는 URI 를 입력합니다 .

   예 : `uri=/my_directory`.
5. Edit Access Control 을 누릅니다 .
6. 모든 사용자에게 읽기 액세스를 허용하는 새 규칙을 만듭니다 .

7. 디렉토리의 사용자에게 액세스를 허용하는 새 규칙을 추가로 만듭니다 .
8. 첫 번째 및 두 번째 줄 모두의 Continue 를 선택 해제 합니다 .
9. 변경 사항에 대하여 Submit 및 Apply 를 누릅니다 .

문서 루트에 상대적인 URI 용 경로가 만들어집니다 . ACL 파일의 항목은 다음과 같이 표시됩니다 . `acl "uri=/my_directory";`

파일 유형에 대한 액세스 제한

서버 또는 웹 사이트의 파일 유형에 대한 액세스를 제한할 수 있습니다 . 예를 들어 오직 지정된 사용자만 서버에서 실행되는 프로그램을 만들 수 있도록 허용할 수 있습니다 . 모든 사람이 프로그램을 실행할 수 있으나 오직 그룹의 지정된 사용자만 프로그램을 만들거나 삭제할 수 있습니다 .

파일 유형에 대한 액세스를 제한하려면 서버 인스턴스에 대한 액세스 제어 설정 방법을 이용합니다 .

1. Server Manager 를 사용하여 서버 인스턴스를 선택합니다 .
2. Preferences 탭을 선택합니다 .
3. Restrict Access 링크를 누릅니다 .
4. Pick a resource 부분에서 Wildcard 를 누르고 와일드카드 패턴을 입력합니다 .

   예 : `*.cgi.`
5. Edit Access Control 을 누릅니다 .
6. 모든 사용자에게 읽기 액세스를 허용하는 새 규칙을 만듭니다 .
7. 오직 지정된 그룹에게 쓰기 및 삭제 액세스를 허용하는 다른 규칙을 만듭니다 .
8. 변경 사항을 Submit 및 Apply 합니다 .

파일 유형 제한의 경우 두 줄의 continue 선택란을 모두 선택합니다 . 파일에 대한 요청이 수신되면 서버는 우선 해당 하일 유형에 대한 ACL 을 확인합니다 .

Patchcheck 기능이 `obj.conf` 에 만들어지며 , 여기에는 파일 또는 디렉토리용 와일드카드 패턴이 포함될 수 있습니다 . ACL 파일의 항목은 다음과 같이 표시됩니다 . `acl "*.cgi";`

하루 중 시간을 기준으로 액세스 제한

특정 서버에 대하여 또는 지정된 시간 또는 일자 동안 쓰기 및 삭제 액세스를 제한할 수 있습니다 . 이 기능을 사용하여 다른 사용자가 파일에 액세스하는 업무 시간 동안 해당 문서가 게시되지 않도록 방지할 수 있습니다 .

하루 중 시간을 기준으로 액세스를 제한하려면 서버 인스턴스에 대한 액세스 제어 설정 방법을 이용합니다 .

1. Server Manager 를 사용하여 서버 인스턴스를 선택합니다 .
2. Preferences 탭을 선택합니다 .
3. Restrict Access 링크를 누릅니다 .
4. Pick a Resource의 드롭 다운 목록에서 전체 서버를 선택하고 Edit Access Control 을 누릅니다 .
5. 모든 사용자에게 읽기 및 실행 권한을 허용하는 새 규칙을 만듭니다 .

   이렇게 하면 사용자가 파일이나 디렉토리를 추가 , 업데이트 또는 삭제하려 할 때 이 규칙이 적용되지 않으며 서버는 일치되는 다른 규칙을 검색합니다 .
6. 모든 사용자의 쓰기 및 삭제 권한을 거부하는 다른 규칙을 만듭니다 .
7. X 링크를 눌러 사용자 정의 표현식을 만듭니다 .
8. 허용할 주 중 요일과 하루 중 시간을 입력합니다 .

   예 :

   ```
   user = "anyone" and
   dayofweek = "sat,sun" or
   (timeofday >= 1800 and
   timeofday <= 600)
   ```

   사용자 정의 표현식을 만들면 Users/Groups 및 From Host 필드에 "Unrecognized expressions" 메시지가 표시될 것입니다 .
9. 변경 사항을 Submit 및 Apply 합니다 .

사용자 정의 표현식에 오류가 있는 경우 오류 메시지가 생성됩니다 . 오류를 수정하고 다시 제출하십시오 .

보안을 기준으로 액세스 제한

Sun ONE Web Server 6.1 의 경우 동일한 서버 인스턴스에 대하여 SSL 청취 소켓과 SSL 이 아닌 청취 소켓을 구성할 수 있습니다 . 보안을 기준으로 액세스를 제한하면 오직 보안 채널을 통하여 전송되어야 하는 리소스를 보호할 수 잇습니다 .

보안을 기준으로 액세스를 제한하려면 서버 인스턴스에 대한 액세스 제어 설정 방법을 이용합니다 .

1. Server Manager 를 사용하여 서버 인스턴스를 선택합니다 .
2. Preferences 탭을 선택합니다 .
3. Restrict Access 링크를 누릅니다 .
4. Pick a Resource의 드롭 다운 목록에서 전체 서버를 선택하고 Edit Access Control 을 누릅니다 .
5. 모든 사용자에게 읽기 및 실행 권한을 허용하는 새 규칙을 만듭니다 .

   이렇게 하면 사용자가 파일이나 디렉토리를 추가 , 업데이트 또는 삭제하려 할 때 이 규칙이 적용되지 않으며 서버는 일치되는 다른 규칙을 검색합니다 .
6. 모든 사용자의 쓰기 및 삭제 권한을 거부하는 다른 규칙을 만듭니다 .
7. X 링크를 눌러 사용자 정의 표현식을 만듭니다 .
8. `ssl="on"` 을 입력합니다 .

   예 :

   ```
   user = "anyone" and ssl="on"
   ```

9. 변경 사항을 Submit 및 Apply 합니다 .

사용자 정의 표현식에 오류가 있는 경우 오류 메시지가 생성됩니다 . 오류를 수정하고 다시 제출하십시오 .

분산 관리로 액세스 제어 보안

이 부분에는 분산 관리를 사용하도록 설정한 후 Sun ONE Web Server 6.1 의 액세스 제어를 보안하기 위하여 수행해야 하는 추가 작업 목록이 있습니다 .

- 리소스에 대한 액세스 보안

리소스에 대한 액세스 보안

`generated.https-`*`server-id`*`.acl` 파일의 `https-`*`server-id`* 개체 태그에 표시되는 PathCheck 지시문의 순서에 따라 리소스에 대한 원하지 않는 액세스가 허용될 수 있습니다 . 이를 방지하려면 아래에 보이는 것과 같이 `<`*`server-root`*`>/generated.https-`*`server-id`*`.acl` 파일을 편집하여 액세스 제어가 필요한 프로그램 그룹을 쉼표로 분리하여 지정합니다 .

시작 라인 위치 :

```
allow (all)
```

```
user=<username> and program=<program group, program group...>;
```

다음 줄 추가 :

```
deny absolute (all)
```

```
user=<username> and program!=<program group, program group...>;
```

서버 인스턴스에 대한 액세스 보안

Sun ONE Web Server 6.1 이 서버 인스턴스에 대한 액세스를 제어하도록 구성하려면 `<`*`server-root`*`>/httpacl/*.https-admserv.acl` 파일을 편집하여 액세스 제어 권한을 부여하려는 사용자를 지정합니다 . 예 :

```
acl "https-<instance>";
```

```
authenticate (user,group) {
```

```
database = "default";
```

```
method = "basic";
```

```
};
```

```
deny absolute (all) user != "UserA";
```

IP 기반 액세스 제어 사용

`ip` 속성을 참조하는 액세스 제어 항목이 ACL 파일 (`gen*.https-admserv.acl`) 에 관련된 Administration Server 안에 있는 경우 아래의 단계 (1) 과 단계 (2) 를 완료합니다 .

ip 속성을 참조하는 액세스 제어 항목이 서버 인스턴스에 관련된 ACL 파일 안에 있는 경우 해당 ACL 에 대하여 단계 (1) 만 완료합니다 .

1. 아래에 보이는 것과 같이 <*server-root*>/httpacl/gen*.https-admserv.acl 파일을 편집하여 user 및 group 에 추가로 인증 목록에 ip 를 추가합니다 .

```
acl "https-admserv";

authenticate (user,group,ip) {

database = "default";

method = "basic";

};
```

2. 다음 액세스 제어 항목을 추가합니다 .

```
deny absolute (all) ip !="ip_for_which_access_is_allowed";
```

예 :

```
acl "https-admserv";

authenticate (user,group,ip) {

database = "default";

method = "basic";

};

deny absolute (all) ip !="205.217.243.119";
```

동적 액세스 제어 파일 작업

서버 컨텐트가 전적으로 한 사람에 의하여 관리되는 경우는 거의 없습니다 . Sun ONE Web Server 에 대한 액세스를 허용하지 않고 최종 사용자가 구성 옵션의 일부에 액세스하여 필요한 사항을 구성할 수 있도록 해야 할 경우가 있습니다 . 구성 옵션의 일부는 동적 구성 파일에 저장됩니다 .

여기에서는 다음 항목에 대해 설명합니다 .

- .htaccess 파일 사용
- 지원되는 .htaccess 지시문
- .htaccess 보안 고려사항

.htaccess 파일 사용

Sun ONE Web Server 는 .htaccess 동적 구성 파일을 지원합니다 . .htaccess 파일은 사용자 인터페이스를 이용하거나 구성 파일을 직접 변경하여 사용하도록 설정합니다 . .htaccess 를 지원하는 파일은 *server_root*/plugins/htaccess 디렉토리에 있습니다. 이 파일에는 .htaccess 파일을 사용할 수 있는 플러그인과 .nsconfig 파일을 .htaccess 파일로 변환하는 스크립트가 포함됩니다 .

.htaccess 파일은 서버의 표준 액세스 제어와 조합하여 사용할 수 있습니다 . PathCheck 지시문의 순서에 상관 없이 표준 액세스 제어는 모든 .htaccess 액세스 제어에 우선하여 적용됩니다 . 사용자 - 그룹 인증이 "Basic" 인 경우에는 사용자 인증에 표준과 .htaccess 액세스 제어를 모두 요구하면 안 됩니다 . 표준 서버 액세스 제어를 통하여 SSL 클라이언트 인증을 사용하는 동시에 .htaccess 파일을 통하여 HTTP "Basic" 인증을 요구할 수 있습니다 .

이 부분에서는 다음 항목에 대해 설명합니다 .

- 사용자 인터페이스에서 .htaccess 사용 설정
- magnus.conf 에서 .htaccess 사용 설정
- 기존 .nsconfig 파일을 .htaccess 파일로 변환
- htaccess-register 사용
- .htaccess 파일 예제

사용자 인터페이스에서 .htaccess 사용 설정

Sun ONE Web Server 가 .htaccess 를 사용하도록 구성하려면 다음과 같이 합니다 .

1. Server Manager 에 액세스하고 .htaccess 를 사용으로 설정할 서버 인스턴스를 선택합니다 .
2. 화면 상단의 Class Manager 링크를 누릅니다 .
3. Content Mgmt 탭을 선택합니다 .
4. .htaccess Configuration 링크를 누릅니다 .
5. 다음의 방법으로 편집하려는 서버를 선택합니다 .
   - 전체 서버를 선택하거나 드롭 다운 목록에서 특정 서버 선택
   - Browse 를 눌러 편집하려는 디렉토리 및 파일 선택
   - Wildcard 를 눌러 편집하려는 와일드카드 패턴 선택

6. Yes 를 선택하여 .htaccess 를 사용합니다 .
7. `.htaccess` 구성이 추가될 파일 이름을 입력합니다 .
8. OK 를 누릅니다 .
9. 작업이 완료되면 Apply 를 누릅니다 .
10. hard start/restart 또는 dynamically apply 를 선택합니다 .

magnus.conf 에서 .htaccess 사용 설정

서버가 `.htaccess` 를 사용하도록 직접 설정하려면 우선 서버의 `magnus.conf` 파일을 수정하여 플러그인을 로드 , 초기화 및 사용하도록 해야 합니다 .

1. *server_root*`/https-`*server_name*`/config` 파일의 `magnus.conf` 를 엽니다 .
2. 다른 `Init` 지시문 뒤에 다음 줄을 추가합니다 .
   - UNIX/Linux:

     ```
     Init fn="load-modules" funcs="htaccess-init,htaccess-find"
     shlib="server_root/plugins/htaccess/htaccess.so?NativeThread="n
     o"
     Init fn="htaccess-init"
     ```

   - Windows:

     ```
     Init fn="load-modules"
     funcs="htaccess-init,htaccess-find,htaccess-register"
     shlib="server_root/plugins/htaccess/htaccess.dll"
     NativeThread="no"
     Init fn="htaccess-init"
     ```

   - HP:

     ```
     Initfn="load-modules"
     funcs="htaccess-init,htaccess-find,htaccess-register"
     shlib="<server_root>/plugins/htaccess/htaccess.sl"
     NativeThread="no"

     Init fn="htaccess-init"
     ```

3. (선택) 마지막 줄을 다음과 같이 수정합니다 .

   ```
   Init fn="htaccess-init"[groups-with-users=yes]
   ```

4. File/Save 를 누릅니다 .
5. `obj.conf` 를 엽니다 .

6. 개체의 마지막 지시문으로 PathCheck 지시문을 추가합니다 .

   a. 가상 서버가 관리하는 모든 디렉토리를 처리하는 .htaccess 를 사용하려면 object.conf 파일의 기본 개체에 PathCheck 지시문을 추가합니다 .

   ```
   <Object name="default">

   ...

   PathCheck fn="htaccess-find"

   </Object>
   ```

   .htaccess 처리는 개체의 마지막 PathCheck 지시문이어야 합니다 .

   b. 특정 서버 디렉토리용 .htaccess 파일 처리를 사용하려면 magnus.conf 에 있는 해당 정의에 PathCheck 지시문을 삽입합니다 .

7. .htaccess 파일의 이름을 .htaccess가 아닌 다른 이름으로 지정하려면 반드시 다음 형식을 사용하여 PathCheck 지시문에 파일 이름을 지정해야 합니다 .

   ```
   PathCheck fn="htaccess-find" filename="filename"
   ```

참고 다음에 Administration Server를 사용하면 수동 편집이 적용되었다는 경고가 표시됩니다 . Apply 를 눌러 변경 사항을 적용합니다 .

이 후 서버에 액세스하면 지정된 디렉터리의 .htaccess 액세스 제어의 대상이 됩니다 . 예를 들어 .htaccess 파일에 대한 쓰기 액세스를 제한하려면 이에 대한 구성 스타일을 만들고 해당 구성 스타일에 액세스 제어를 적용합니다 . 자세한 내용은 제 17 장 , " 구성 스타일 적용 " 을 참조하십시오 .

기존 .nsconfig 파일을 .htaccess 파일로 변환

Sun ONE Web Server 6.1 에는 기존 .nsconfig 파일을 .htaccess 파일로 변환할 수 있는 htconvert 플러그인이 있습니다 . nsconfig 파일은 더 이상 지원되지 않습니다 . 이제까지 .nsconfig 파일을 사용한 경우에는 이를 .htaccess 파일로 변환해야 합니다 .

htconvert를 시작하면 server.xml 파일에서 pfx2dir과 document-root 지시문을 검색합니다 . 검색된 각 .nsconfig 파일은 .htaccess 파일로 변환됩니다 . 구성에 따라 여러 개의 obj.conf 파일이 변환될 수 있습니다 .

참고 기존 .htaccess 파일이 있는 경우 htconvert 는 htaccess.new 파일을 만들고 경고가 표시됩니다 . .htaccess 와 .htaccess.new 파일이 모두 있는 경우 새 파일의 이름은 htaccess.new.new 가 됩니다 . .new 는 반복적으로 추가됩니다 .

htconvert 플러그인은 현재 restrictAccess 및 RequireAuth 지시문과 <Files> 래퍼 함수만 지원합니다 . <Files*> 가 아닌 <Files> 이 있는 경우 스크립트는 경고를 표시하며 해당 디렉토리의 모든 파일에 액세스 제어가 적용된 것처럼 작동합니다 .

파일을 변환하려면 명령 프롬프트에서 시스템의 Perl 경로 , 플러그인 스크립트의 경로 , 및 serer.xml 파일의 경로를 입력합니다 . 예 :

```
server_root\install\perl server_root/plugins/htaccess/htconvert
server_root/https-server_name/config/server.xml
```

모든 .nsconfig 파일은 .htaccess 파일로 변환되지만 삭제되지는 않습니다 .

groups-with-users 옵션을 사용하면 그룹의 사용자가 많은 경우 쉽게 처리할 수 있습니다 . 그룹의 사용자가 많은 경우 다음과 같이 합니다 .

1. 사용자 파일 형식을 다음과 같이 사용자가 속한 그룹 목록으로 수정합니다 .

   ```
   username:password:group1,group2,group3,...groupn
   ```

2. AuthGroupFile 지시문이 AuthUserFile 과 동일한 파일을 가리키도록 수정합니다 .

다른 방법으로 다음과 같이 할 수 있습니다 .

1. AuthGroupFile 지시문 전체를 제거합니다 .
2. 다음을 magnus.conf 파일의 Init fn=htaccess-init 줄에 추가합니다 .

   ```
   groups-with-users="yes"
   ```

htaccess-register 사용

`htaccess-register` 은 새로운 기능으로 사용자가 자신의 인증 방법을 만들 수 있습니다 . Apache 와 마찬가지로 외부 인증 모듈을 만들고 이를 `htaccess-register` 를 통하여 `.htaccess` 모듈에 삽입할 수 있습니다 .
server_root`/plugins/nsapi/htaccess` 에 두 가지 예제 모델이 있습니다 .

외부 모듈을 사용하여 하나 이상의 새 지시문을 만들 수 있습니다 . 예를 들어 인증용 사용자 데이터베이스를 지정할 수 있습니다 . 지시문은 `<Limit>` 또는 `<LimitExcept>` 태그에 표시되지 않을 수도 있습니다 .

.htaccess 파일 예제

다음은 `.htaccess` 파일의 예입니다 .

```
<Limit GET POST>
order deny,allow
deny from all
allow from all
</Limit>
<Limit PUT DELETE>
order deny,allow
deny from all
</Limit>
AuthName mxyzptlk.kawaii.com
AuthUserFile /server_root/mxyz-docs/service.pwd
AuthGroupFile /server_root/mxyz-docs/service.grp
```

지원되는 .htaccess 지시문

이 릴리스에서는 다음의 `.htaccess` 지시문을 지원합니다 .

allow

구문

다음의 호스트를 허용합니다 .

- 호스트가 all 이거나 모든 클라이언트 호스트의 액세스를 허용
- 호스트가 all 이거나 DNS 호스트 이름의 일부분인 경우
- 호스트가 full 이거나 IP 주소의 일부분인 경우

보통 <Limit> 또는 <LimitExcept> 범위에 포함시키지만 반드시 그렇게 해야 하는 것은 아닙니다 .

효과
지정된 호스트로의 액세스를 허용합니다 . 보통 <Limit> 범위 안에 표시됩니다 .

deny

구문
다음 위치의 호스트를 거부합니다 .

- 호스트가 all 이거나 모든 클라이언트 호스트의 액세스를 거부
- 호스트가 all 이거나 DNS 호스트 이름의 일부분인 경우
- 호스트가 full 이거나 IP 주소의 일부분인 경우

보통 <Limit> 또는 <LimitExcept> 범위에 포함시키지만 반드시 그렇게 해야 하는 것은 아닙니다 .

효과
지정된 호스트로의 액세스를 거부합니다 . 보통 <Limit> 범위 안에 표시됩니다 .

AuthGroupFile

구문
AuthGroupFile filename으로 여기에서 filename은 groupname:user user 형식의 그룹 정의를 포함하는 파일의 이름입니다 .

<Limit> 또는 <LimitExcept> 범위 내에 표시되면 안 됩니다 .

효과
해당 이름의 그룹 파일이 필요한 group 지시문에서 참조된 그룹 정의용으로 사용되도록 지정합니다 . 참고로 , AuthGroupFile 지지문에 지정된 filename 이 AuthUserFile 지시문에 지정된 filename 과 동일한 경우 파일에 다음 형식의 사용자 및 그룹이 있는 것으로 가정합니다 .

```
username:DES-encrypted-password:comma-separated-list-of-groups
```

AuthUserFile

구문

`AuthUserFile` filename. 여기에서 ,

- filename 은 다음 형식의 사용자 정의가 포함된 파일의 이름입니다 . `username:password`
- username 은 사용자 로그인 이름이며 password 는 DES 암호화 비밀번호입니다 .

`<Limit>` 또는 `<LimitExcept>` 범위 내에 표시되면 안 됩니다 .

효과

해당 이름의 파일이 필요 user 또는 필요한 valid-user 지시문에서 참조되는 사용자 이름용으로 사용되도록 지정합니다 .

`obj.conf` 의 `Init fn=htaccess-init` 지시문 내의 `groups-with-users=yes` 를 사용하거나 `AuthGroupFile` 지시문을 지정하면 파일의 형식이 다음과 같은 것으로 가정합니다 .

```
username:DES-encrypted-password:comma-separated-list-of-groups
```

AuthName

구문

`AuthName` 인증 영역 . 여기에서 인증 영역은 임의의 사용자 인증용 요청과 연결되는 인증 영역을 구분하는 문자열입니다 .

`<Limit>` 또는 `<LimitExcept>` 범위 내에 표시되면 안 됩니다 .

효과

인증 영역 문자열은 보통 클라이언트 측의 사용자 이름 및 비밀번호를 묻는 프롬프트에 표시됩니다 . 해당 클라이언트에서 사용자 이름 및 비밀번호를 캐시하는 데 영향을 미칠 수 있습니다 .

AuthType

구문

`AuthType` Basic. `<Limit>` 또는 `<LimitExcept>` 범위 내에 표시되면 안 됩니다 .

효과
사용자 인증을 HTTP Basic Authentication 으로 지정하며 , 현재 유일하게 지원되는 방법입니다 .

<Limit>

구문

```
<Limit method method ...>

allow, deny, order, or require directives

</Limit>
```

여기에서 method 는 GET, POST 또는 PUT 등의 HTTP 메소드입니다 . 여기에는 서버가 알 수 있는 모든 메소드를 사용할 수 있습니다 .

효과
지정된 HTTP 메소드를 사용하는 요청에만 포함된 지시문을 적용합니다 .

<LimitExcept>

구문

```
<LimitExcept method method ...>

allow, deny, order, or require directives

</LimitExcept>
```

여기에서 method 는 GET, POST 또는 PUT 등의 HTTP 메소드입니다 . 여기에는 서버가 알 수 있는 모든 메소드를 사용할 수 있습니다 .

효과
지정된 HTTP 메소드와 일치하지 않는 요청 유형에만 포함된 지시문을 적용합니다 .

order

구문
Order 순서 . 여기에서 순서는 다음 중 한 가지입니다 .

- allow, deny
- deny, allow
- mutual-failure

보통 <Limit> 또는 <LimitExcept> 범위에 포함시키지만 반드시 그렇게 해야 하는 것은 아닙니다 .

효과

- deny, allow, 모든 deny 지시문을 평가한 후 allow 지시문 평가
- allow, deny, 모든 allow 지시문을 평가한 후 deny 지시문 평가
- Mutual-failure 는 순서에 상관 없이 allow 와 deny 지시문 목록 모두에 존재하는 호스트의 액세스를 거부

require

구문

- require group groupname groupname
- require user username username
- require valid-user

보통 <Limit> 또는 <LimitExcept> 범위에 포함시키지만 반드시 그렇게 해야 하는 것은 아닙니다 .

효과

- require group 의 경우 인증된 사용자가 지정된 그룹의 구성원이어야 합니다 .
- require user 의 경우 인증된 사용자가 지정된 사용자 중 하나이어야 합니다 .
- require valid-user 의 경우 인증된 사용자이어야 합니다 .

.htaccess 보안 고려사항

기본적으로 HTTP PUT 용 서버 지원은 사용하지 않습니다 . Class Manager 에 있는 Content Mgmt 의 Remote File Manipulation 페이지를 사용하여 HTTP PUT 을 사용하도록 설정할 수 있습니다 . .htaccess 파일을 포함하는 디렉토리에 PUT 액세스를 허용하는 경우 해당 파일이 변경될 수 있으므로 매우 신중해야 합니다 . 액세스를 제한하여 디렉토리의 모든 파일에 대하여 PUT 액세스를 금지할 수 있습니다 . " 디렉토리 (경로) 에 대한 액세스 제한 " 페이지 205 참조 .

가상 서버용 액세스 제어

Sun ONE Web Server 6.1 의 액세스 제어 정보는 각 가상 서버 ACL 파일과 문서 디렉토리의 `.htaccess` 파일에서 찾을 수 있습니다 . `.htaccess` 시스템은 iPlanet Web Server 4.x 부터 변경되지 않았습니다 .

`server.xml` 파일에는 하나 이상의 ACLFILE 태그가 있을 수 있으며, 이 태그는 특정 표준 Sun ONE Web Server 6.x ACL 파일에 연결된 ID 를 정의합니다 . 예 :

```
<ACLFILE id="standard" file="standard.acl">
```

가상 서버에서 액세스 제어를 사용하려면 반드시 해당 "aclids" 등록 정보에 있는 하나 이상의 ACL 파일 ID 에 대한 참조를 만들어야 합니다 . 예 :

```
<VS aclids="standard">
```

이 구성을 사용하면 여러 대의 가상 서버가 동일한 ACL 파일을 공유할 수 있습니다 . 가상 서버용으로 사용자 - 그룹 인증을 요구하도록 하려면 하나 이상의 USERDB 태그를 해당 정의에 추가해야 합니다 . USERDB 태그는 ACL 파일의 데이터베이스 이름과 `dbswitch.conf` 파일에 있는 실제 데이터베이스 사이를 연결합니다 .

다음 예제에서는 "database" 속성이 없는 ACL 을 `dbswitch.conf` 의 "default" 데이터베이스로 매핑합니다 .

```
<VS>
   <USERDB id="default" database="default"/>
</VS>
```

가상 서버에서 데이터베이스 액세스

`dbswitch.conf` 파일에 전역적으로 사용자 인증 데이터베이스를 정의할 수 있습니다 . 오직 서버가 시작할 때에만 이 파일을 읽습니다 .

`dbswitch.conf` 에 있는 LDAP URL 의 `baseDN` 은 데이터베이스로의 모든 액세스에 대한 전역 루트를 정의합니다 . 따라서 역방향 호환성이 유지됩니다 . 신규 설치의 경우 대부분 `baseDN` 은 비어있습니다 .

dcsuffix는 dbswitch.conf에 있는 LDAP 데이터베이스용의 새로운 속성으로 Sun ONE LDAP 스키마에 따라 DC 트리의 루트를 정의합니다 . 이는 LDAP URL 의 baseDN 에 상대적입니다 . dcsuffix 속성이 있는 경우 LDAP 데이터베이스는 Sun ONE LDAP 스키마와 호환되며 일부 운영의 작동이 변경됩니다 . Sun ONE LDAP 스키마에 대한 자세한 내용과 예제는 Sun ONE Web Server 6.1 *Administrator's Configuration Reference* 의 제 2 장에 있는 "The Sun ONE LDAP Schema" 를 참조하십시오 .

모든 가상 서버에 대하여 디렉토리 중 하나를 가리키는 USERDB 블록을 하나 이상 정의할 수 있으며 추가 정보를 정의할 수 있습니다 . USERDB 블록 ID 는 ACL 의 데이터베이스 매개 변수에서 참조할 수 있습니다 . 가상 서버에 USERDB 블록이 없는 경우 사용자 또는 그룹 기반 ACL 은 실패합니다 .

USERDB 태그는 ACL 의 데이터베이스 속성과 dbswitch.conf 사이에 추가의 간접 레이어를 정의합니다 . 이 간접 레이어는 서버 관리자가 가상 서버 관리자에 의하여 액세스되는 데이터베이스를 완전히 제어할 수 있도록 필요한 보호를 추가합니다 .

USERDB 에 대한 자세한 내용은 Sun ONE Web Server 6.1 *Administrtor's Configuration Reference* 의 제 2 장에 있는 "User Database Selection" 을 참조하십시오 .

사용자 인터페이스에서 LDAP 데이터베이스 지정

dbswitch.conf 에 하나 이상의 사용자 인증 데이터베이스를 정의한 후 , Class Manager 를 사용하여 각 가상 서버가 인증용으로 사용할 데이터베이스를 구성할 수 있습니다 . 또한 Class Manager 를 사용하여 dbswitch.conf 에서 새로 만들어진 데이터베이스 정의를 추가하여 가상 서버가 인증에 사용하도록 할 수 있습니다 .

가상 서버가 사용할 LDAP 데이터베이스를 지정하려면 다음과 같이 합니다 .

1. Server Manager 에 액세스하고 Virtual Server Class 탭을 선택합니다 .
2. Server 의 Tree View 에 있는 LDAP 데이터베이스 목록에서 데이터베이스를 지정할 가상 서버 클래스 링크를 누릅니다 .
3. Virtual Servers 탭이 표시되지 않았으면 선택하여 표시합니다 .
4. ACL Settings 링크를 누릅니다 .

   ACL Settings for Virtual Servers 페이지가 표시됩니다 .
5. Option 열의 드롭 다운 목록에 Edit 이 표시되어 있지 않으면 지금 선택합니다 .
6. 편집하는 가상 서버의 Database 열에 있는 드롭 다운 목록에서 데이터베이스 구성을 선택합니다 .

7. OK 를 누릅니다 .
8. Edit ACL Files 창을 닫습니다 .
9. Apply 를 누릅니다 .
10. Dynamically Apply 를 선택합니다 .

가상 서버용 액세스 제어 목록 편집

가상 서버용 ACL 은 가상 서버가 상주하는 서버 인스턴스용으로 만들어집니다 . 가상 서버 ACL 설정은 해당 서버 인스턴스용으로 만들어진 ACL 의 기본값이 됩니다 . 그러나 각 가상 서버의 액세스 제어는 Class Manager 에서 변경할 수 있습니다 . 또한 이 방법을 사용하여 가상 서버에 새로 만든 ACL 파일을 추가할 수 있습니다 .

가상 서버용 ACL 설정을 변경하려면 다음과 같이 합니다 .

1. Server Manager 에 액세스하고 Virtual Server Class 탭을 선택합니다 .
2. Server 의 Tree View 에 있는 LDAP 데이터베이스 목록에서 데이터베이스를 지정할 가상 서버 클래스 링크를 누릅니다 .
3. Virtual Servers 탭이 표시되지 않았으면 선택하여 표시합니다 .
4. ACL Settings 링크를 누릅니다 .
5. 변경하려는 각 가상 서버에 대하여 Option 필드에 있는 드롭 다운 목록에서 Edit 또는 Delete 를 선택합니다 .
6. 사용 가능한 ACL 파일을 표시하려면 ACL File 필드에서 Edit 링크를 누릅니다 .
7. 해당 가상 서버에 추가 또는 삭제할 ACL 파일을 하나 이상 선택합니다 .

   가상 서버의 문서 루트는 여러 개일 수 있으므로 ACL 파일 또한 여러 개가 있을 수 있습니다 .
8. 드롭 다운 목록에서 ACL 목록에 연결할 데이터베이스를 선택합니다 .
9. (선택) BaseDN 을 입력합니다 .
10. 변경을 완료했으면 OK 를 누릅니다 .
11. Apply 를 누릅니다 .
12. Dynamically Apply 를 선택합니다 .

파일 기반 인증용 ACL 생성

Sun ONE Web Server 6.1 에서는 파일 기반 인증 데이터베이스를 사용할 수 있으며 , 이 파일은 보통 파일로 텍스트 형식으로 사용자 및 그룹 정보를 저장합니다 . ACL 프레임워크는 파일 인증 데이터베이스와 함께 작동하도록 디자인되었습니다 .

참고 Sun ONE Web Server 6.1 은 동적 보통 파일을 지원하지 않습니다 . 보통 파일 데이터베이스는 서버가 시작할 때 로드됩니다 . 파일이 변경되는 경우 오직 서버가 재시작 되어야 적용됩니다 .

ACL 항목은 `database` 키워드를 사용하여 사용자 데이터베이스를 참조할 수 있습니다 . 예 :

```
acl "default";
    authenticate (user) {
...
    database="myfile";
...
};
```

`myfile` 데이터베이스는 `server.xml` 에 있는 `VS` 의 `USERDB` 요소에서 참조될 수 있으며 , 이 경우 `server-root/userdb/dbswitch.conf` 파일과 연결됩니다 . 예 :

```
<VS>
...
    <USERDB id="myfile" database="myfiledb">
...
</VS>
```

`server-root/userdb/dbswitch.conf` 파일에는 파일 auth-db 와 해당 구성을 정의하는 항목이 있습니다 . 예 :

```
directory myfiledb file
myfiledb:syntax keyfile
myfiledb:keyfile /path/to/config/keyfile
```

아래의 표

표 9-3) 파일 인증 데이터베이스가 지원하는 매개 변수

구문 .	[선택] 값은 keyfile, digest 또는 htaccess 입니다 . 지정하지 않는 경우 기본값은 keyfile 입니다 .
keyfile	[syntax=keyfile 인 경우 필요] 사용자 데이터가 있는 파일 경로 .
digestfile	[syntax=digest 인 경우 필요] 다이제스트 인증용 사용자 데이터가 있는 파일 경로 .
groupfile	[syntax=htaccess 인 경우 필요] AuthGroupFile 의 경로 .
userfile	[syntax=htaccess 인 경우 필요] AuthUserFile 의 경로 .

주의

파일 인증 데이터베이스 파일 (htaccess, digestfile 또는 keyfile) 의 최대 줄 수는 255 입니다 .

이 한계를 초과하는 경우 서버는 시작할 수 없으며 로그 파일에 오류가 기록됩니다 .

참고

파일 기반 인증 데이터베이스를 사용하는 ACL을 설정하기 전에 다음의 전제 조건을 만족하는지 확인하십시오 .

- 파일 기반 인증 디렉토리 서비스가 이미 구성되어야 합니다 . 구성 방법에 대한 자세한 내용은 " 디렉토리 서비스 구성 " 페이지 55 를 참조하십시오 .
- ACL 이 설정될 가상 서버가 필요한 파일 기반 인증 데이터베이스 유형 (keyfile, htaccess 또는 digestauth) 을 사용하도록 구성되어야 합니다 . 구성되지 않은 경우 기본으로 구성된 디렉토리 서비스에 대하여 ACL 제한이 구성됩니다 .

파일 인증을 기반으로 디렉토리 서비스용 ACL 생성

파일 인증을 기반으로 디렉토리 서비스용 ACL 항목을 만들려면 다음과 같이 합니다 .

1. Server Manager 에 액세스하고 ACL 을 만들거나 편집하려는 서버 인스턴스를 선택합니다 .

2. Server Manager 에서 Preferences 탭을 선택합니다 .
3. Restrict Access 링크를 누릅니다 .
4. Option 열의 드롭 다운 목록에서 ACL 파일을 선택하고 Edit ACL 을 클릭합니다 .
5. 상단 창의 Access Control Rules 페이지에서 편집하려는 ACL 의 Users/Groups 링크를 누릅니다 .
6. 하단 창의 User/Group 페이지의 Authentication 데이터베이스 드롭 다운 목록에서 keyfile 을 선택합니다 .
7. Update 를 누릅니다 .

   아래의 예제와 같이 키파일 기반 파일 인증 데이터베이스에 대한 ACL 을 설정하는 경우 `dbswitch.conf` 파일이 ACL 항목을 포함하여 업데이트됩니다 .

```
version 3.0;
  acl "default";
  authenticate (user) {
  prompt = "Sun One Web Server 6.1";
  database = "mykeyfile";
  method = "basic";
  };
deny (all) user = "anyone";
allow (all) user = "all";
```

htaccess 인증을 기반으로 디렉토리 서비스용 ACL 생성

Sun ONE Web Server 에서 .htaccess 기반 보통 파일 인증을 지원합니다 . 이제까지 .htaccess 인증을 사용한 경우에는 기존 데이터파일을 변경 없이 파일 인증 데이터파일로 이전할 수 있습니다 . .htaccess 파일 사용에 언급한 것과 같이 .htaccess 사용자 및 그룹 데이터는 단일 파일 또는 두 개의 별도 파일 (사용자 데이터용 한 개 및 그룹 데이터용 한 개) 에 저장됩니다 . 파일 인증 데이터베이스는 기존 형식 두 가지를 모두 지원합니다 .

htaccess 인증을 기반으로 디렉토리 서비스용 ACL 항목을 만들려면 다음과 같이 합니다 .

1. Server Manager에 액세스하고 ACL을 만들거나 편집하려는 서버 인스턴스를 선택합니다.
2. Server Manager 에서 Preferences 탭을 선택합니다.
3. Restrict Access 링크를 누릅니다.
4. Option 열의 드롭 다운 목록에서 ACL 파일을 선택하고 Edit ACL을 클릭합니다.
5. 상단 창의 Access Control Rules 페이지에서 편집하려는 ACL 의 Users/Groups 링크를 누릅니다.
6. 하단 창의 User/Group 페이지의 Authentication 데이터베이스 드롭 다운 목록에서 htaccess 를 선택합니다.
7. Update 를 누릅니다.

   htaccess 기반 파일 인증 데이터베이스에 대한 ACL 을 설정하는 경우 `dbswitch.conf` 파일이 아래의 예제와 같은 ACL 항목을 포함하여 업데이트됩니다.

```
version 3.0;
acl "default";
  authenticate (user) {
  prompt = "Sun One Web Server 6.1";
  database = "myhtaccessfile";
  method = "basic";
  };
deny (all) user = "anyone";
allow (all) user = "all";
```

기존 .htaccess 정보를 파일 인증 데이터베이스로 이전

기존 .htaccess 정보를 Sun ONE Web Server 6.1 의 파일 인증 데이터베이스로 이전하려면 다음과 같이 합니다.

- .htaccesss userfile 데이터베이스를 *server-root*/*server-instance*/config/userfile 로 복사합니다.
- .htaccesss group 데이터베이스를 *server-root*/*server-instance*/config/gropufile 로 복사합니다.

사용자 파일 형식 :

```
#user:password
```

그룹 파일 형식 :

```
#group1:user1 user2
#group2:user3 user4
```

참고 구성원 이름은 공백으로 분리합니다 .

userfile 과 groupfile 의 이름이 동일한 경우 파일이 조합되며 조합된 파일의 각 줄은 아래에 보이는 것과 같은 구문을 따릅니다 .

```
#user:password:group1,group2
```

참고 열은 콜론 (:) 으로 분리합니다 .

htaccess 데이터베이스 예제

예제 1

```
#sample userfile  (user/password "j2ee/j2eepwd"  user/password
"user1/user1pwd" )
j2ee:9hmjfRwNxvJLU
user1:wvQirF86BsjSk
```

예제 2

```
#sample group file
staff:j2ee  user1
eng:j2ee
```

예제 3

```
#sample user/group file (username "j2ee", user password "j2eepwd")
j2ee:9hmjfRwNxvJLU:staff,eng
```

다이제스트 인증을 기반으로 디렉토리 서비스용 ACL 생성

파일 인증 데이터베이스는 또한 각 암호 기반 RFC 2617.A 해시 다이제스트 인증을 사용하기 적합한 파일 형식을 지원하며 , 영역은 저장됩니다 . 보통 텍스트 비밀번호는 보관되지 않습니다 .

다이제스트 인증을 기반으로 디렉토리 서비스용 ACL 항목을 만들려면 다음과 같이 합니다 .

1. Server Manager 에 액세스하고 ACL 을 만들거나 편집하려는 서버 인스턴스를 선택합니다 .
2. Server Manager 에서 Preferences 탭을 선택합니다 .
3. Restrict Access 링크를 누릅니다 .
4. Option 열의 드롭 다운 목록에서 ACL 파일을 선택하고 Edit ACL 을 클릭합니다 .
5. 상단 창의 Access Control Rules 페이지에서 편집하려는 ACL 의 Users/Groups 링크를 누릅니다 .
6. 하단 창의 User/Group 페이지의 Authentication 데이터베이스 드롭 다운 목록에서 digest 를 선택합니다 .
7. Update 를 누릅니다 .

   digestauth 기반 파일 인증 데이터베이스에 대한 ACL 을 설정하는 경우 `dbswitch.conf` 파일이 아래의 예제와 같은 ACL 항목을 포함하여 업데이트됩니다 .

```
version 3.0;
  acl "default";
  authenticate (user) {
  prompt = "filerealm";
  database = "mydigestfile";
  method = "digest";
  };
deny (all) user = "anyone";
allow (all) user = "all";
```

10 장

로그 파일 사용

다양한 방법으로 서버의 작동을 모니터할 수 있습니다 . 이 장에서는 로그 파일을 기록하고 확인하여 서버를 모니터하는 방법에 대하여 설명합니다 . 내장 성능 모니터 서비스 , 서비스 품질 기능 또는 SNMP 사용에 대한 내용은 모니터 서비스를 참조하십시오 .

이 장의 내용 :

로그 파일 설명

서버 로그 파일은 서버의 작동을 기록합니다 . 이 로그를 서버 모니터와 문제 해결에 사용할 수 있습니다 . 서버에 발생한 모든 오류 목록은 서버 루트 디렉토리의 `https-`*`server_name`*`/logs/errors`에 있는 오류 로그 파일에 있습니다. 액세스 로그는 서버 루트 디렉토리의 `https-`*`server_name`*`/logs/access` 에 있으며 , 서버로의 요청과 서버로부터의 응답에 대한 정보가 기록됩니다 . Sun ONE Web Server `access` 로그 파일에 기록된 정보를 구성할 수 있습니다 . 서버 통계를 생성하려면 로그 분석기를 사용합니다 . 서버 오류 및 액세스 로그를 백업하려면 해당 파일을 보관합니다 .

참고	운영 체제의 한계로 인하여 Linux 의 경우 Sun ONE Web Server 는 2GB 를 초과하는 로그 파일을 사용할 수 없습니다 . 최대 파일 크기가 초과되면 기록이 중단됩니다 .

UNIX 및 Windows 플랫폼에서의 로깅

여기에서는 로그 파일이 만들어지는 방법에 대하여 설명합니다 . 또한 다음 항목에 대하여 설명합니다 .

- 기본 오류 로깅
- syslog 을 사용하여 로깅
- Windows eventlog 을 사용하여 로깅

기본 오류 로깅

UNIX 및 Windows 플랫폼 모두의 경우 Administration Server 의 로그는 관리 서버 `https-admserve/logs` 디렉토리에서 수집됩니다 . 서버 인스턴스로부터의 로그는 `https-`*`server_name`*`/logs/` 디렉토리에 수집됩니다 .

전체 서버용 기본 로그 수준을 설정할 수 있습니다 . stdout 및 stderr 을 서버의 이벤트 로그로 재지정할 수 있으며 로그 출력을 운영 시스템의 시스템 로그로 지정할 수 있습니다 . 또한 stdout 및 stderr 내용을 서버의 이벤트 로그로 지정할 수 있습니다 . 기본적으로 로그 메시지는 지정된 서버 로그 파일뿐 아니라 stderr 로 또한 전송됩니다 .

사용할 수 있는 다른 기능은 가상 서버 ID 를 로그 메시지와 함께 기록할 수 있는 기능입니다 . 이 기능은 여러 개의 가상 서버가 동일한 로그 파일에 메시지를 기록하는 경우 유용합니다 . 메시지를 시스템 로그에 기록하도록 선택할 할 수 있습니다 . 이렇게 하면 로깅이 오류 로그 파일에 수행되지 않습니다 . 그 대신 UNIX 에서의 syslog 로깅 서비스 또는 Windows 플랫폼에서의 시스템 로깅 서비스가 로그를 만들고 관리할 수 있습니다 .

또한 `server.xml` 속성을 사용하여 이 파일의 내용을 제어할 수 있습니다 . `server.xml` 파일에 대한 자세한 내용은 Sun ONE Web Server 6.1 *Administrator's Configuration File Reference* 를 참조하십시오 .

syslog 을 사용하여 로깅

중앙집중식 로깅이 필요한 안정된 운영 환경의 경우 syslog 를 사용하는 것이 더 좋습니다 . 진단 및 디버깅용으로 로그 출력이 자주 필요한 환경의 경우 개별 서버 인스턴스 또는 가상 서버 로그가 더 관리하기 쉽습니다 .

참고

- 하나의 파일에 기록되는 모든 서버 인스턴스 및 관리 서버용 데이터는 읽고 디버깅하기 어려운 것이 될 수 있습니다 . 오직 문제 없이 실행되는 응용 프로그램에 대하여 syslog 마스터 로그 파일을 사용하는 것이 좋습니다 .
- 기록된 메시지는 Solaris 데몬 응용 프로그램으로부터의 모든 기타 로그와 혼합됩니다 .

syslogd 및 시스템 로드 데몬과 함께 syslog 로그 파일을 사용하면 `syslog.conf` 파일을 다음과 같이 구성할 수 있습니다 .

- 적절한 시스템 로그로 메시지 기록
- 시스템 콘솔에 메시지 표시
- 기록된 메시지를 전달하여 모든 사용자 목록을 표시하거나 네트워크를 통하여 다른 호스트의 다른 syslog 로 기록된 메시지 전달

syslog 으로의 로깅은 Sun ONE Web Server 로부터의 로그를 의하며 기타 데몬 응용 프로그램이 동일한 파일에 수집되므로 , 기록된 메시지는 다음 정보를 포함하여 특정 서버 또는 가상 서버 인슨턴스로부터의 Sun ONE Web Server 특정 메시지를 구분합니다 .

- Unique message ID
- Timestamp

- Instancename
- Program name (webservd 또는 webserv-wdog)
- Process ID (webserv 프로세스의 PID)
- Thread ID (선택)
- Server ID

LOG 요소는 `server.xml` 파일에서 관리 서버와 서버 인스턴스 모두에 대하여 구성할 수 있습니다 .

UNIX 운영 체제에서 사용되는 syslog 로깅 메커니즘에 대한 자세한 내용은 터미널 프롬프트에서 다음의 man 명령을 사용하십시오 .

```
man syslog
man syslogd
man syslog.conf
```

Windows eventlog 을 사용하여 로깅

Windows 운영 체제에서 사용하는 이벤트 로그 메커니즘에 대한 내용은 Windows 도움말에서 Event Logging 키워드로 검색하십시오 .

로그 수준

Sun ONE Web Server 의 로그 수준과 메시지는 중요도의 순서에 따라 다음 표에 정의된 것과 같습니다 .

표 10-1) 로그 수준

로그 수준	설명
`finest`	메시지는 디버그 메시지의 다변화 수준을 표시합니다 . `finest` 의 경우 다변화가 최대입니다 .
`finer`	
`fine`	
`info`	원래 정보를 제공하는 메시지이며 , 보통 서버 구성 또는 서버 상태에 관련된 메시지입니다 . 즉각적인 조치가 필요한 오류를 표시하는 메시지는 아닙니다 .

표 10-1) 로그 수준

로그 수준	설명
warning	경고를 표시하는 메시지입니다 . 이 메지시에는 예외가 포함될 수 있습니다 .
failure	정상의 응용 프로그램 실행을 방해할 수 있는 중요한 이상을 표시하는 메시지입니다 .
config	다양한 정적 구성 정보에 관련된 메시지로 특정 구성에 관련된 문제를 해결하는데 도움이 됩니다 .
security	보안 문제를 표시하는 메시지입니다 .
catastrophe	중요한 오류를 표시하는 메시지입니다 .

가상 서버 및 로깅 설명

Sun ONE Web Server 에는 가상 서버 인스턴스가 있을 수 있습니다 . Sun ONE Web Server 인스턴스에 있는 각 가상 서버는 자체의 ID 가 있으며 자체의 로그 파일이 있을 수 있습니다 . 각 가상 서버에 별도의 로그 파일을 사용하면 특정 트랜잭션 및 리소스에 대한 서버 작동을 추적하는 데 도움이 됩니다 .

또한 여러 가상 서버에서 기록된 메시지를 단일 서버 로그 파일로 보낼 수 있습니다 . 이렇게 하는 경우 `server.xml` 파일의 `LOG` 요소에 있는 `logvsid` 를 사용하도록 설정해야 합니다 . 이렇게 하면 사용자가 서로 다른 가상 서버에서 전송되는 로그 메시지를 구분할 수 있습니다 .

```
<SERVER>

  ...

  <LOG file="/export//https-iws-files2.red.iplanet.com/logs/errors"
loglevel="finest" logtoconsole="true" usesyslog="false"
createconsole="false" logstderr="true" logstdout="true"
logvsid="true"/>

</SERVER>
```

이 예에서 `<Log logvsid="true">` 에 의하여 모든 로그 메시지에 가상 서버 ID 가 포함됩니다 . 따라서 서로 다른 가상 서버에서 들어오는 메시지를 구분할 수 있습니다 . `VS` 요소에 `errorlog` 요소가 없는 경우 모든 가상 서버가 단일 파일에 메시지를 기록하게 됩니다 .

응용 프로그램 및 서버 로그 출력 재지정

개발자에게 있어 Web 응용 프로그램 구성요소와 J2EE 응용 프로그램을 시험하는 동안 응용 프로그램 로그와 서버 로그를 즉시 사용할 수 있도록 하는 것이 중요합니다 . Windows 플랫폼의 경우 개발자는 바탕화면의 명령 프롬프트 창에 로그 메시지를 표시하는 경우가 많습니다 . UNIX 플랫폼의 경우 많은 개발자는 서버 인스턴스가 시작된 터미널 창의 stderr 로 로그 메시지 문자열이 표시되도록 하거나 `tail -f` 명령을 사용하여 로그 파일에 기록된 로그 메시지를 표시합니다 .

`server.xml` 파일에 기록된 메시지를 로그 파일이나 터미널 창 등으로 보내는 stdout 및 stderr 용으로 설정 가능한 속성이 있습니다 . stdout 및 stderr 의 사용에 대한 내용은 Sun ONE Web Server 6.1 *Administrator's Configuration File Reference* 를 참조하십시오 .

로그 파일 보관

액세스 및 오류 로그 파일이 자동 보관되도록 설정할 수 있습니다 . 특정 시간이나 지정된 시간이 경과하면 로그가 교체됩니다 . Sun ONE Web Server 는 이전 로그 파일을 저장하고 파일이 저장된 일자 및 시간이 포함된 이름을 파일에 지정합니다 .

예를 들어 , 파일이 매시간 교체되도록 설정하면 Sun ONE Web Server 는 파일을 "*`access.200307152400`*" 이라는 이름으로 저장합니다 . 여기에서 로그 파일 이름 , 년 , 월 , 일 및 24 시간 형식 시간은 단일 문자열로 합쳐집니다 . 로그 보관 파일의 정확한 형식은 설정한 로그 교체 유형에 따라 달라집니다 .

Sun ONE Web Server 는 파일 아카이브용으로 두 가지의 로그 교체 유형을 제공합니다 . 바로 내부 데몬 로그 교체 및 Cron 기반 로그 교체입니다 .

내부 데몬 로그 교체

이러한 형태의 로그 교체는 HTTP 데몬에서 수행되며 오직 시작할 때 구성될 수 있습니다 . 내부 데몬 로그 교체를 사용하면 서버가 서버 재시작 없이 내부적으로 로그를 교체할 수 있습니다 . 이 방법으로 교체한 로그는 다음의 형식으로 저장됩니다 .

`access.<YYYY><MM><DD><HHMM>`

`error. <YYYY><MM><DD><HHMM>`

로그 파일을 교체하고 새 로그 파일을 시작할 기준으로 사용할 시간을 지정할 수 있습니다 . 예를 들어 , 교체 시작 시간이 12:00 a.m. 이고 교체 간격이 1440 분 (하루) 이면 , 현재 시간에 상관 없이 변경 사항을 저장 및 적용할 때 새 로그 파일이 만들어집니다 . 로그 파일은 매일 오전 12:00 에 교체되며 액세스 로그 파일은 12:00am 으로 스탬프되고 `access.200307152400` 으로 저장됩니다 . 마찬가지로 간격을 240 분 (4 시간) 으로 설정하고 간격이 오전 12:00 에 시작하면 액세스 로그 파일에는 오전 12:00 에서 오전 4:00 까지 , 오전 4:00 에서 오전 8:00 까지 등의 순서로 정보를 수집됩니다 .

로그 교체를 사용하는 경우 로그 교체는 서버가 시작할 때 시작됩니다 . 첫 로그 파일은 현재 시간부터 다음 교체 시간까지 정보를 수집합니다 . 앞의 예에서 시작 시간을 오전 12:00 으로 , 교체 간격을 240 분으로 설정하며 현재 시간이 오전 6:00 이라면 , 교체의 첫 번째 로그 파일에는 오전 6:00 에서 오전 8:00 까지 수집된 정보가 포함되며 다음 로그 파일에는 오전 8:00 에서 오후 12:00(정오) 까지의 정보가 포함됩니다 .

스케줄 기반 로그 교체

이 유형의 로그 교체는 *server_root*`/https-admserv/config/` 디렉토리의 `scheduler.conf` 파일에 저장된 시간에 따라 수행됩니다 . 이 방법을 사용하면 로그 파일을 즉시 보관하거나 특정 일자의 특정 시간에 서버가 로그 파일을 보관하도록 할 수 있습니다 . 서버의 스케줄러 구성은 *server_root*`/https-admserv/config/` 디렉토리의 `schedulerd.conf` 파일에 저장됩니다 . 스케줄 기반 방법으로 교체된 로그는 다음의 형식으로 저장됩니다 .

`<`*original_filename*`>.<YYYY><MM><DD><HHMM>`

예를 들어 `access` 가 오후 4:30 에 교체되면 `access.200307151630` 이 됩니다 .

로그 교체는 서버가 시작할 때 초기화됩니다 . 순환을 사용하는 경우 Sun ONE Web Server 는 시간 스탬프 액세스 로그 파일을 만들고 서버가 시작할 때 순환이 시작됩니다 .

교체가 시작되면 Sun ONE Web Server 는 액세스 로그 파일에 기록해야 할 요청이 있는 경우 , 새로운 시간 스탬프 로그 파일을 만들며 , 또한 이 작업은 미리 설정된 " 다음 교체 시간 " 이 경과하면 수행됩니다 .

참고 로그 분석기를 실행하기 전에 서버 로그를 보관해야 합니다 .

로그 파일을 보관하고 내부 데몬 방법 또는 스케줄 기반 방법을 사용할 것인지 지정하려면 Server Manager 의 Archive Log Files 페이지를 사용합니다 .

액세스 로그 기본 설정

설치할 때 access 라는 이름의 액세스 로그 파일이 해당 서버용으로 만들어집니다 . 액세스를 기록할 것인지의 여부 , 기록에 사용할 형식 및 리소스에 액세스할 때 서버가 클라이언트의 도메인 이름을 조회할 것인지의 여부를 지정하여 모든 리소스에 대한 액세스 로깅을 사용자 정의할 수 있습니다 .

로그 파일 형식 문자열에 %vsid% 를 추가하려면 다음과 같이 합니다 .

1. Server Manager 에 액세스하고 Logs 탭을 선택합니다 .
2. Access Log Preferences 링크를 누릅니다 .
3. Log File: 입력란에 새 로그 파일 위치 및 파일 이름을 입력합니다 .
4. Only Log: 선택 버튼을 누릅니다 .
5. Virtual Server ID 선택란을 누릅니다 . 다른 방법으로 custom Format: 선택 버튼을 누르고 "%vsid% 문자열을 추가합니다 .

참고 사용자 정의 문자열 %vsid% 를 추가하는 경우 반드시 새 액세스 로그 파일을 사용해야 합니다 .

기존 로그 파일의 형식을 변경하는 경우 우선 기존 로그 파일을 삭제 / 이름 변경하거나 다른 파일 이름을 사용해야 합니다 .

서버 액세스 로그는 Common LogFile Format, 유연한 로그 형식 또는 사용자 정의 형식을 사용할 수 있습니다 . Common LogFile Format 은 흔히 지원되는 형식으로 서버에 대한 고정된 양의 정보를 제공합니다 . 유연한 로그 형식을 사용하면 로그할 내용을 선택 (Sun ONE Web Server 에서) 할 수 있습니다 . 사용자 지정 형식의 경우 로그할 사항을 조정하는 매개 변수 블록을 사용합니다 . 사용자 정의할 수 있는 형식 매개 변수의 목록은 *NSAPI Programmer's Guide* 를 참조하십시오 .

리소스용 액세스 로그가 일단 만들어지면 해당 로그를 보관하거나 해당 리소스용으로 새 액세스로그 파일을 만들지 않는 한 , 이 로그를 변경할 수 없습니다 .

Server Manager 의 Access Log Preferences 페이지를 사용하거나 `obj.conf` 파일에서 다음 지지문을 직접 구성하여 로깅 기본 설정을 지정할 수 있습니다 . `magnus.conf` 에서 서버는 `flex-init` 함수를 호출하여 유연한 로깅 시스템을 초기화하며 `flex-log` 함수를 호출하여 요청에 대한 데이터를 유연한 로그 형식으로 기록합니다 . 요청을 공통 로그 파일 형식으로 기록하려면 서버가 `init-clf` 를 호출하여 `obj.conf` 에서 사용되는 Common Log 하위 시스템을 초기화하고 `common-log` 을 호출하여 요청에 대한 데이터를 공통 로그 형식 (대부분의 HTTP 서버에서 사용) 으로 기록합니다 .

NSAPI 로깅 함수 및 포함된 지시문과 매개 변수에 대한 자세한 내용은 *NSAPI Programmer's Guide* 를 참조하십시오 .

용이한 쿠키 로깅

Sun ONE Web Server 에서는 flexlog 기능을 사용하여 특정 쿠키를 쉽게 기록할 수 있습니다 . 구성 파일 `obj.conf` 에서 `flex-log` 하위 시스템을 초기화하는 줄에 "`Req->headers.cookie.`*`cookie_name`*" 을 추가합니다 . 이렇게 하면 쿠키 변수가 요청의 헤더에 있는 경우 쿠키 변수 *`cookie_name`* 의 값을 기록하며 , 쿠키 변수가 없는 경우에는 "`-`" 을 기록합니다 .

오류 로깅 옵션 설정

Sun ONE Web Server 6.1 에서 서버의 오류 로그에 기록될 정보를 구성할 수 있습니다 .

Administration Server 인스턴스의 경우

1. Administration Server 에 액세스합니다 .
2. Preferences 탭을 선택합니다 .
3. Access Log Options 링크를 누릅니다 .
4. 필요한 정보를 입력합니다 .
5. 확인을 누른 후 Apply 를 눌러 변경 사항을 저장하고 적용합니다 .

Server Instance 의 경우

1. 해당 서버 인스턴스에 액세스합니다 .

2. Logs 탭을 선택합니다 .
3. Error Log Preferences 링크를 누릅니다 .
4. 필요한 정보를 입력합니다 .
5. 확인을 누른 후 Apply 를 눌러 변경 사항을 저장하고 적용합니다 .

LOG 요소 구성

server.xml 파일에서 구성할 수 있는 LOG 요소용 속성은 다음 표와 같습니다 .

표 2) LOG 속성

속성	기본값	설명
file	errors	기본 서버에서 보내는 메시지를 저장할 파일을 지정합니다 . errorlog 속성이 VS 요소에 명시적으로 지정되지 않는 한 구성된 다른 서버의 메시지도 여기로 보내집니다 .
loglevel	info	다른 요소가 오류 로그에 기록한 메시지의 기본 유형을 제어합니다 . 최고에서 최저까지 허용되는 값은 다음과 같습니다 . finest, finer, fine, info, warning, failure, config, security, 및 catastrophe.
logvsid	false	(선택) true 인 경우 가상 서버 로그에 가상 서버 ID 가 표시됩니다 . 이는 여러 VS 요소가 동일한 로그 파일을 공유할 경우 유용합니다 . 참고로 Sun ONE Web Server 6.1 에서 logvsid 요소는 magnus.conf 파일에 구성될 수 없습니다 .
logstdout	true	(선택) true 인 경우 stdout 출력을 오류 로그로 보냅니다 . 적절한 값은 on, off, yes, no, 1, 0, true, false 입니다 .
logstderr	true	(선택) true 인 경우 stderr 출력을 오류 로그로 보냅니다 . 적절한 값은 on, off, yes, no, 1, 0, true, false 입니다 .
logtoconsole	true	(선택 , UNIX 전용) true 인 경우 로그 메시지를 콘솔로 보냅니다 .

표 2) LOG 속성

속성	기본값	설명
createconsole	false	(선택 , Windows 전용) true 인 경우 stderr 출력용 Windows 콘솔을 만듭니다. 적절한 값은 on, off, yes, no, 1, 0, true, false 입니다 .
usesyslog	false	(선택) true 인 경우 UNIX syslog 서비스 또는 Windows Event Logging 을 사용하여 로그를 생성하고 관리합니다 . 적절한 값은 on, off, yes, no, 1, 0, true, false 입니다 .

액세스 로그 파일 확인

서버의 사용 중인 로그 파일과 보관된 로그 파일을 볼 수 있습니다 .

Administration Server 에서 Administration Server 의 액세스 로그를 보려면 Preferences 탭을 선택한 후 , View Access Log 페이지를 선택합니다 .

Server Manager 에서 서버 인스턴스용 액세스 로그를 보려면 Logs 탭을 선택한 후 , View Access Log 페이지를 선택합니다 .

Class Manager 에서 개별 가상 서버의 액세스 로그를 보려면 선택된 Manage Virtual Servers 페이지에서 관리하려는 가상 서버를 선택한 후 Virtual Server Manager 페이지의 Access Log 아래의 해당 링크를 누릅니다 . 표시할 항목의 수를 지정하거나 선택한 조건에 맞는 항목을 표시할 수 있습니다 .

다음은 Common Logfile Format 의 액세스 로그 예입니다 . (형식은 Log Preferences 창에서 지정합니다 . 자세한 내용은 " 액세스 로그 기본 설정 " 페이지 238 을 참조하십시오 .

```
wiley.a.com - - [16/Feb/2001:21:18:26 -0800] ?ET / HTTP/1.0?200 751
wiley.a.com - - [17/Feb/2001:1:04:38 -0800] ?ET /docs/grafx/icon.gif HTTP/1.0?204 342
wiley.a.com - - [20/Feb/2001:4:36:53 -0800] ?ET /help HTTP/1.0?401 571
arrow.a.com - john [29/Mar/2001:4:36:53 -0800] ?ET /help HTTP/1.0?401 571
```

표 10-3 은 이 예제 액세스 로그 마지막 줄에 대한 설명입니다 .

표 10-3) 예제 액세스 로그 파일의 마지막 줄 필드

액세스 로그 필드	예
클라이언트의 호스트 이름 또는 IP 주소	arrow.a.com. (이 경우 DNS 조회용 서버 설정이 사용으로 되어 있으므로 호스트 이름이 표시됩니다. DNS 조회가 사용 안 함으로 설정되면 클라이언트의 IP 주소가 표시됩니다 .)
RFC 931 정보	- (RFC 931 ID 는 구현되지 않음)
사용자 이름	john (클라이언트가 인증용으로 입력한 사용자 이름)
요청 일자 / 시간	29/Mar/1999:4:36:53 -0800
요청	GET /help
프로토콜	HTTP/1.0
상태 코드	401
전송된 바이트	571

다음은 유연한 로깅 형식의 액세스 로그 예입니다 . (형식은 Log Preferences 페이지에서 지정합니다 . 자세한 내용은 " 액세스 로그 기본 설정 " 페이지 238 을 참조하십시오 .

```
wiley.a.com - - [25/Mar/2001:12:55:26 -0800] "GET /index.htm HTTP/1.0" "GET" "/?-" "HTTP/   1.0" 304 0 -
Mozilla/2.0 (WinNT; I)
wiley.a.com - - [25/Mar/2001:12:55:26 -0800] "GET / HTTP/1.0" "GET" "/?-" "HTTP/1.0" 304 0    -
Mozilla/2.0 (WinNT; I)
wiley.a.com - - [25/Mar/2001:12:55:26 -0800] "GET / HTTP/1.0" "GET" "/?-" "HTTP/1.0" 304 0    -
Mozilla/2.0 (X11; I; IRIX 5.3 IP22)
```

오류 로그 파일 확인

오류 로그 파일에는 로그 파일이 만들어진 후부터 서버에 발생한 오류가 기록되며 , 서버의 시작 시간 등 서버에 대한 정보 메시지가 들어 있습니다 . 성공하지 못한 사용자 인증 또한 오류 로그에 기록됩니다 . 오류 로그를 사용하여 끊어진 URL 경로나 누락된 파일을 찾을 수 있습니다 .

Administration Server 의 오류 로그 파일을 보려면 Administration Server 에서 Preferences 탭을 선택한 후 , View Error Log 페이지를 선택합니다 .

서버 인스턴스의 오류 로그 파일을 보려면 Server Manager 에서 Logs 탭을 선택한 후 , View Error Log 페이지를 선택합니다 .

개별 가상 서버의 오류 로그를 보려면 Class Manager 에서 선택된 Manage Virtual Servers 페이지에서 관리하려는 가상 서버를 선택한 후 Virtual Server Manager 페이지에 있는 Error Log 아래의 해당 링크를 누릅니다 . 표시할 항목의 수를 지정하거나 선택한 조건에 맞는 항목을 표시할 수 있습니다 .

다음은 오류 로그 항목의 두 가지 예입니다 . 첫 번째 예는 서버가 성공적으로 시작되었다는 정보 메시지를 표시하며 , 두 번째 예는 클라이언트 `wiley.a.com` 이 파일 `report.html` 을 요청했으나 파일이 서버의 기본 문서 디렉토리에 존재하지 않음을 표시합니다 .

```
[[22/Jan/2001:14:31:41] info (39700): successful server startup [22/Jan/2001:14:31:41] info (39700):
SunONE-WebServer/6.1 BB1-01/22/2001 01:45
[22/Jan/2001:14:31:42] warning (13751): for host wiley.a.com trying to GET /report.html, send-file reports:
can't find /usr1/irenem/ES60-0424/docs/report.html (File not found)
```

로그 분석기 실행

server-root`/extras/log_anly` 디렉토리에는 Server Manager 사용자 인터페이스를 통하여 실행할 수 있는 로그 분석 도구가 있습니다 . 이 로그 분석기는 오직 공통 로그 형식의 파일만 분석합니다 . log_anly 디렉토리내의 HTML 문서 도구의 매개 변수를 설명합니다 . *server-root*`/extras/flex_anlg` 디렉토리에는 유연한 로그 파일 형식용 명령줄 로그 분석기가 있습니다 . 그러나 Server Manager 는 공통 또는 유연한 로그 파일 형식 중 어느 것을 선택하는지에 상관 없이 기본적으로 유연한 로그 파일 보고 도구를 사용하도록 설정합니다 .

로그 분석기를 사용하여 작동 요약 , 가장 많이 액세스된 URL, 하루 중 서버에 대한 액세스가 가장 많은 시간 등의 기본 서버에 대한 통계를 생성합니다 . 또한 Sun ONE Web Server 또는 명령줄에서 로그 분석기를 실행할 수 있습니다 . 로그 분석기는 기본 서버가 아닌 가상 서버용 통계는 생성할 수 없습니다 . 그러나 통계는 " 액세스 로그 파일 확인 " 페이지 241 에 설명한 것과 같이 각 서버에 대한 통계를 볼 수 있습니다 .

`flexanlg` 명령줄 유틸리티를 실행하기 전에 반드시 라이브러리 경로를 설정해야 합니다 . 다양한 플랫폼용 설정은 다음과 같습니다 .

Solaris 및 Linux:

```
LD_LIBRARY_PATH=server_root/bin/https/lib:$LD_LIBRARY_PATH
```

AIX:

```
LIBPATH=server_root/bin/https/lib:$LIBPATH
```

HP-UX:

```
SHLIB_PATH=server_root/bin/https/lib:$SHLIB_PATH
```

Windows:

```
path=server_root\bin\https\bin;%path%
```

참고 로그 분석기를 실행하기 전에 서버 로그를 보관해야 합니다 . 서버 로그 보관에 대한 내용은 " 로그 파일 보관 " 페이지 236 를 참조하십시오 .

Server Manager 에서 로그 분석기를 실행하려면 다음과 같이 합니다 .

1. Server Manager 에서 Logs 탭을 누릅니다 .
2. Generate Report 를 누릅니다 .
3. 필드에 값을 입력합니다 .
4. OK 를 누릅니다 .

   새 창에 보고서가 표시됩니다 .

자세한 내용은 온라인 도움말의 Generate Report 페이지를 참조하십시오 .

명령줄에서 액세스 로그 파일을 분석하려면 `flexanlg` 도구를 실행합니다 . 이 도구는 *server-install*`/extras/flex_anlg` 디렉토리에 있습니다 .

`flexanlg` 를 실행하려면 명령 프롬프트에서 다음 명령과 옵션을 입력합니다 .

```
flexanlg [ -P ] [-n name] [-x] [-r] [-p order] [-i file]* [ -m
metafile ]* [ o file][ c opts] [-t opts] [-l opts]
```

구문은 다음의 설명과 같습니다 .

```
flexanlg -h.):
-P: 프록시 로그 형식                          기본값 : no
-n servername: 서버의 이름
-x : HTML 출력                              기본값 : no
-r : IP 주소를 호스트 이름으로 변환                 기본값 : no
-p [c,t,l]: 출력 순서 ( 계수 , 시간 상태 , 목록 )  기본값 : ctl
-i filename: 입력 로그 파일                    기본값 : none
-o filename: 출력 로그 파일                     기본값 : stdout
-m filename: 메타 파일                       기본값 : none
-c [h,n,r,f,e,u,o,k,c,z]: 이들 항목 계수          기본값 : hnreuokc
   h: 전체 히트
   n: 304 상태 코드 수정 안 됨 ( 로컬 사본 사용 )
   r: 302 상태 코드 찾음 ( 재지정 )
   f: 404 상태 코드 없음 ( 문서 없음 )
   e: 500 서버 오류 상태 코드 ( 구성오류 )
   u: 총 고유 URL 의 수
   o: 총 고유 호스트 수
   k: 전송된 KB
   c: 캐시가 저장한 KB
   z: 항목 계수 안 함
-t [sx,mx,hx, xx,z]: 일반 상태 검색 - 기본값 : s5m5h24x10
   s(number): 로그의 최상위 ( 숫자 ) 초 검색
   m(number): 로그의 최상위 ( 숫자 ) 분 검색
   h(number): 로그의 최상위 ( 숫자 ) 시간 검색
   u(number): 로그의 최상위 ( 숫자 ) 사용자 검색
   a(number): 로그의 최상위 ( 숫자 ) 사용자 에이전트 검색
   r(number): 로그의 최상위 ( 숫자 ) 참조자 검색
   x(number): 보조 키워드용 최상위 ( 숫자 ) 검색
   z: 일반 상태 찾지 않음
-l [cx,hx]: 다음 목록 만듬 - 기본값 : c+3h5
   c(x,+x): 가장 많이 액세스된 URL
         (x: x 항목 목록만 표시 )
         (+x: x 번 이상 액세스된 목록만 표시 )
   h(x,+x): 서버에 가장 많이 액세스한 호스트 ( 또는 IP 주소 )
         (x: x 항목 목록만 표시 )
         (+x: x 번 이상 액세스된 목록만 표시 )
   z: 목록 만들지 않음
```

이벤트 보기 (Windows)

서버 오류 로그에 오류를 기록 (" 오류 로그 파일 확인 " 페이지 242 참조) 하는 것 외에 Sun ONE Web Server 는 Event Viewer 에 심각한 시스템 오류를 기록합니다 . Event Viewer 를 사용하여 시스템의 이벤트를 모니터할 수 있습니다 . Event Viewer 를 사용하여 기능적 구성 문제로 인한 오류를 볼 수 있습니다. 이 오류는 오류 로그가 열리기 전에 발생할 수 있습니다 .

Event Viewer 를 사용하려면 다음과 같이 합니다 .

1. 시작 메뉴에서 모든 프로그램을 선택한 후 관리 도구를 선택합니다 . 관리 도구 프로그램 그룹에서 Event Viewer 를 선택합니다 .
2. Log 메뉴에서 Application 을 선택합니다 .

   Event Viewer 에 Application 리그가 표시됩니다 . Sun ONE Web Server 의 오류에는 `https-`*`serverid`* 또는 `WebServer6.1` 의 소스 레이블이 포함됩니다 .
3. View 메뉴에서 Find 를 선택하여 로그에서 이들 레이블 중 한 가지를 검색합니다 . 로그 항목을 업데이트하려면 Veiw 메뉴에서 Refresh 를 선택합니다 .

   Event Viewer 에 대한 자세한 내용은 시스템 설명서를 참조하십시오 .

11 장

모니터 서비스

이 장에서는 내정 모니터 도구 , 서비스 품질 기능 및 SNMP(Simple Network Management Protocol) 등을 포함하여 서버를 모니터하는 방법에 대항 설명합니다 .

SNMP 를 Sun ONE 관리 정보 베이스 (MIB) 및 HP OpenView 등의 네트워크 소프트웨어와 함께 사용하여 네트워크에서 기타 장치를 모니터하는 것과 마찬가지로 서버를 실시간으로 모니터할 수 있습니다 .

참고 Windows 의 경우 Sun ONE Web Server 6.1 을 설치하기 전에 컴퓨터에 Windows SNMP 구성요소가 이미 설치되었는지 확인해야 합니다 .

통계 기능 또는 SNMP 를 사용하여 실시간으로 서버의 상태를 확인할 수 있습니다 . UNIX 또는 Linux 를 사용하는 경우 SNMP 를 사용하려면 반드시 Sun ONE 서버를 SNMP 용으로 구성해야 합니다 . 이 장에서는 UNIX 또는 Linux 에서 Sun ONE 서버와 함께 SNMP 를 사용하는 데 필요한 정보를 제공합니다 .

이 장에서는 다음 항목에 대해 설명합니다 .

통계를 사용하여 서버 모니터

통계 기능을 사용하여 서버의 현재 작동을 모니터할 수 있습니다 . 통계에는 서버가 처리하는 요청의 수와 해당 요청을 처리하는 상태 등이 표시됩니다 . 개별 가상 서버용 일부 통계와 전체 서버 인스턴스에 대한 기타 통계도 확인할 수 있습니다 . 대화형 서버 모니터 보고서에 서버가 많은 수의 요청을 처리하는 것으로 표시되는 경우 요청을 수용하도록 서버 구성 또는 시스템의 네트워크 커넬을 조정할 수 있습니다 . 자세한 내용은 온라인 Sun ONE Web Server 6.1 *Performance Tuning, Sizing, and Scaling Guide* 를 참조하십시오 .

통계를 사용하도록 설정하면 다음 영역에 대한 통계를 볼 수 있습니다 .

- 연결
- DNS
- KeepAlive
- 캐시
- 가상 서버

대화형 서버 모니터 보고서의 다양한 서버 통계에 대한 전체적인 설명은 온라인 도움말의 Monitor Current Activity 페이지를 참조하십시오 .

주의 통계 / 프로필 작성을 사용하는 경우 서버의 모든 사용자가 통계 정보를 사용할 수 있습니다. 자세한 내용은 Sun ONE Web Server 6.1 *NSAPI Programmer's Guide* 의 stats-xml 설명을 참조하십시오 .

통계 사용 설정

통계를 사용하려면 다음과 같이 합니다 .

1. Server Manager 에서 Monitor 탭을 누릅니다 .

2. Monitor Current Activity 를 누릅니다 .
3. Yes 를 눌러 통계를 사용하도록 설정합니다 .
4. OK 를 누릅니다 .
5. Apply 를 눌러 변경 사항을 적용합니다 . 서버를 재시작할 필요는 없습니다 .

통계를 사용하도록 설정하는 데 대한 내용은 온라인 도움말을 참조하십시오 .

통계 사용

통계를 사용하도록 설정하면 서버 인스턴스와 가상 서버가 실행되는 상태에 대한 다양한 정보를 얻을 수 있습니다 . 통계는 기능적 영역으로 나누어집니다 .

통계에 액세스하려면 다음과 같이 합니다 .

1. Server Manager 에서 Monitor 탭을 누릅니다 .
2. Monitor Current Activity 를 누릅니다 .
3. 드롭 다운 목록에서 폴 간격을 선택합니다 .

   폴 간격은 표시되는 통계 정보를 업데이트하는 초 단위 간격입니다 .
4. 드롭 다운 목록에서 표시하려는 통계의 종류를 선택합니다 .
5. Submit 을 누릅니다 .

   서버 인스턴스가 실행 중이며 통계 / 프로필 작성을 사용하는 경우 선택한 종류의 통계를 표시하는 페이지가 나타납니다 . 이 페이지는 폴 간격에서 선택한 값에 따라 5-15 초마다 업데이트됩니다 .

통계에 표시되는 데이터를 사용하여 서버를 조정할 수 있습니다 . 자세한 내용은 온라인 Sun ONE Web Server 6.1 *Performance Tuning, Sizing, and Scaling Guide* 를 참조하십시오 .

서비스 품질 사용

Quality of Service 는 서버 인스턴스 가상 서버 클래스 또는 가상 서버에 대해 설정한 성능 한계를 가리킵니다 . 예를 들어 ISP 인 경우 가상 서버에 대하여 허용하는 대역폭의 정도에 따라 해당 가상 서버에 서로 다른 요금을 부과할 수 있습니다 . 대역폭의 양과 연결의 수 등 두 가지 영역을 제한할 수 있습니다 .

Monitor 탭에서 전체 서버 또는 Server Manager 의 가상 서버 클래스용 설정을 사용할 수 있습니다 . 그러나 개별 가상 서버용으로 서버 또는 클래스 수준 설정을 무시할 수 있습니다 . 개별 서버용 서비스 품질 한계 설정에 대한 자세한 내용은 " 가상 서버 서비스 품질 구성 " 페이지 332 을 참조하십시오 .

재계산 간격 및 메트릭 간격 등 두 가지 설정이 트래픽 계수 방법과 대역폭을 다시 계산하는 빈도를 결정합니다 . 재계산은 대역폭을 계산하는 빈도 (1/000 초) 입니다 . 메트릭 간격은 데이터가 트래픽 계산에 사용되는 시간입니다 .

이 부분에서는 다음 항목에 대해 설명합니다 .

- 서비스 품질 예제
- 서비스 품질 설정
- obj.conf 의 필요한 변경 사항
- 서비스 품질의 알려진 한계

서비스 품질 예제

다음의 예는 서비스 품질 정보가 수집되고 계산되는 방법입니다 .

서버의 메트릭 간격은 30 초 입니다 .

시간 0 초에 서버가 시작합니다 .

시간 1 초에 HTTP 연결이 서버에 대하여 5000 바이트의 트래픽을 만듭니다 .

이 후 더 이상 연결이 만들어지지 않습니다 . 30 초에 지난 30 초 동안의 총 트래픽은 5000 바이트입니다 .

32 초에 1 초에 만들어진 트래픽 예제는 메트릭 간격 30 초를 경과하였으므로 삭제됩니다 . 이제 지난 30 초 동안의 총 트래픽은 0 입니다 .

재계산 간격도 비슷하게 작동합니다 . 서버의 재계산 간격은 100ms 입니다 .

동일한 예제에서 대역폭은 매 100ms 마다 다시 계산됩니다 . 계산은 트래픽의 양뿐 아니라 메트릭 간격에 따라 수행됩니다 .

시간 0 초에 대역폭이 처음 계산됩니다 . 총 트래픽은 0 이며 , 30 초의 메트릭 간격으로 나누는 경우 대역폭은 0 입니다 .

1 초에 대역폭은 10 번 째 시간 (1000ms/100ms) 으로 계산됩니다 . 총 트래픽은 5000 바이트이며 , 이는 30 초로 나누어집니다 . 대역폭은 5000/30 = 166 바이트 / 초입니다 .

시간 30 초에 대역폭이 300 번째 시간으로 계산됩니다 . 총 트래픽은 5000 바이트이며 , 이는 30 초로 나누어집니다 . 대역폭은 5000/30 = 166 바이트 / 초입니다 .

시간 32 초에 대역폭이 320 번째 시간으로 계산됩니다 . 이제 트래픽은 0 이 되며 (트래픽을 생성한 연결의 시간 경과), 30 으로 나누면 대역폭은 0 바이트 / 초가 됩니다 .

서비스 품질 설정

서버 인스턴스 또는 가상 서버 클래스용 서비스 품질 설정을 구성하려면 사용자 인터페이스를 통한 설정을 구성해야 합니다 . 서비스 품질 설정을 실제로 집행하려면 반드시 obj.conf 파일의 SAF(Server Application Function) 을 설정해야 합니다 .

서비스 품질을 구성하려면 다음과 같이 합니다 .

1. Server Manager 에서 Monitor 탭을 누릅니다 .
2. Quality of Service 를 누릅니다 .

   서비스 품질용 일반 설정 목록과 전체로서의 서버 인스턴스 및 각 가상 서버 클래스 목록이 있는 페이지가 표시됩니다 .
3. 전체에 대하여 서비스 품질을 사용하려면 Enable 을 누릅니다 .

   기본적으로 서비스 품질을 사용하도록 설정됩니다 . 서비스 품질을 사용하면 서버의 오버헤드가 약간 증가합니다 .
4. Recompute Interval 을 선택합니다 .

   재계산 간격은 모든 서버 , 클래스 및 가상 서버용 대역폭에 대한 각 계산의 간격을 1/1000 초 단위로 입력합니다 . 기본값은 100 입니다 .

5. Metric Interval 을 선택합니다 .

   메트릭 간격은 트래픽이 측정되는 간격을 초 단위로 지정합니다 . 기본값은 30 초입니다 . 이 시간 동안 측정되는 모든 대역폭은 바이트 / 초로 평균됩니다 .

   사이트에 큰 파일 전송이 많은 경우 이 필드에 큰 값 (수 분 이상) 을 사용하십시오 . 큰 파일을 전송하려면 짧은 메트릭 기간 동안 모든 허용된 대역폭을 사용할 수 있으므로 최대 대역폭 설정을 사용하는 경우 연결이 거부될 수 있습니다 . 대역폭은 메트릭 간격으로 평균되므로 간격이 길면 큰 파일로 인하여 발생하는 첨두치를 완화시킬 수 있습니다 .

   대역폭 한계가 사용 가능한 대역폭 보다 많이 낮은 경우 (예를 들어 백본으로의 1GB/ 초 연결에서 1MB/ 초로 대역폭을 제한하는 경우), 메트릭 간격을 짧게 해야 합니다 .

   또한 큰 정적 파일을 전송하며 대역폭 한계가 사용 가능한 대역폭 보다 많이 낮은 경우 문제를 해결하려면 반대의 해결책이 필요하므로 , 어떤 상황을 조정할 것인지 결정해야 합니다 .

6. 서버 인스턴스 및 가상 서버 클래스용 서비스 품질을 사용하도록 설정합니다 .

   화면의 아래 부분에는 서버 인스턴스 및 가상 서버의 목록이 표시됩니다 . 서비스 품질을 사용하려는 항목 옆의 작동으로 Enable 을 선택합니다 .

7. 최대 대역폭을 바이트 / 초로 설정합니다 .

8. 최대 대역폭 설정을 집행할 것인지 선택합니다 .

   최대 대역폭을 집행하도록 선택하면 , 서버의 대역폭 한계를 초과하는 경우 추가의 연결은 거부됩니다 .

   최대 대역폭을 집행하지 않으면 최대값을 초과하는 경우 서버가 오류 로그에 메시지를 기록합니다 .

9. 허용할 최대 연결 수를 선택합니다 .

   이는 동시에 처리되는 요청의 수 입니다 .

10. 최대 연결 수 설정을 집행할 것인지 선택합니다 .

    최대 연결 수를 집행하도록 선택하면 , 서버의 한계를 초과하는 경우 추가의 연결은 거부됩니다 .

11. 최대 연결 수를 집행하지 않으면 최대값을 초과하는 경우 서버가 오류 로그에 메시지를 기록합니다 .

12. OK 를 누릅니다 .

obj.conf 의 필요한 변경 사항

서비스 품질을 사용하려면 `obj.conf` 파일에 지시문을 추가하여 AuthTrans `qos-handler` 및 Error `qos-error`등의 두 가지 SAF(Server Application Function)을 시작해야 합니다 .

적절히 작동하려면 `qos-handler` AuthTrans 지시문이 기본 개체에 구성된 `AuthTrans` 중 첫 번째이어야 합니다 . 서비스 품질 처리기의 역할은 가상 서버 , 가상 서버 클래스 및 전역 서버용 현재 통계를 검사하고 오류를 반환하여 한계를 집행하는 것입니다 .

Sun ONE Web Server 에는 `qos-handler` 라고 하는 예제 서비스 품질 처리기 SAF 가 포함되어 있습니다 . 이 SAF 는 한계를 초과하는 경우 이를 기록하며 503 "Server busy" 를 서버로 반환하여 NSAPI 가 이를 처리하도록 합니다 .

Sun ONE Web Server 에는 또한 `qos-error` 이라는 예제 오류 SAF 가 포함되어 있으며 , 이는 503 오류가 발생한 한계를 표시하는 오류 페이지와 한계가 적용된 통계 값을 반환합니다 . 예제 코드를 변경하여 다른 오류 정보를 제공할 수 있습니다 .

이들 예제는 *server_root*`/plugins/nsapi/examples/qos.c` 에서 사용할 수 있습니다 . 이 예제를 사용하거나 SAF 를 새로 만들 수 있습니다 .

SAF 에 대한 설명과 사용 방법은 Sun ONE Web Server 6.1 *NSAPI Programmer's Guide* 를 참조하십시오 .

서비스 품질의 알려진 한계

서비스 품질 기능을 사용하는 경우 다음의 한계에 유의하십시오 .

- 성능으로 인하여 연결 또는 대역폭 통계를 서버 프로세스 전체에서 공유할 수 없습니다 . 즉 , `MaxProc` 의 설정을 사용할 수 없습니다 . 따라서 모든 한계는 서버 프로세스에 개별적으로 적용되며 , 모든 프로세스에 대하여 누적되지 않습니다 . `MaxProcs` 및 복수 프로세스에 대한 자세한 내용은 온라인 Sun ONE Web Server 6.1 *Performance Tuning, Sizing, and Scaling Guide* 를 참조하십시오 .
- 서비스 품질 기능은 오직 응용 프로그램 수준의 HTTP 대역폭만을 측정합니다 . 다양한 이유로 HTTP 대역폭은 실제 TCP 네트워크 대역폭과 다를 수 있습니다 .
  - SSL 을 사용하는 경우 핸드셰이크와 클라이언트 인증서 교환이 트래픽에 추가되지만 , 이는 측정되지 않습니다 .

- 한 방향 또는 양방향으로 청크 부호화를 사용하는 경우 청크화 레이어가 청크 헤더를 제거하지만 트래픽에서는 계산되지 않습니다 . 기타 헤더 및 프로토콜 항목은 계산됩니다 .

- 서비스 품질 기능은 PR_TransmitFile 호출의 트래픽을 정확히 측정할 수 없습니다 . PR_Send()/net_write 또는 PR_Recv()/net_read 등의 기본 I/O 동작의 경우 , 하나의 시스템 호출에서 전송된 바이트 수는 보통 버퍼의 크기이며 I/O 호출은 빠르게 반송되므로 전송된 데이터는 대역폭 관리자에 의하여 즉시 계산됩니다 . 이는 동적 컨텐츠 응용 프로그램의 순간적인 대역폭을 측정하는 경우 좋습니다 . 그러나 PR_TransmitFile 에서 전송된 데이터의 양은 전송의 끝에서 알 수 있으므로 전송이 완료되기 전에는 측정되지 않습니다 .

  PR_TransmitFile 이 작은 경우 서비스 품질 기능이 적절히 작동할 것입니다 . 그러나 전화 접속 사용자가 큰 파일을 다운로드 하는 경우와 같이 PR_TransmitFile 이 큰 경우 전송된 전체 데이터 양은 전송이 완료될 때 계산됩니다 . 다음 재계산 간격이 시작된 후 대역폭 관리자가 대역폭을 재계산하면 최근의 큰 PR_TransmitFile 로 인하여 계산된 대역폭이 상당히 커지게 됩니다 . 이 경우 지속 시간이 경과하여 대역폭 관리자가 파일 전송 작업을 " 무효화 " 시키는 다음 메트릭 간격까지 (따라서 대역폭이 낮아질 때까지) 서버는 모든 요청을 거부할 수 있습니다 . 사이트에서 아주 긴 정적 파일 다운로드가 많은 경우 메트릭 간격을 기본값인 30 초보다 크게 설정해야 합니다 .

- 계산된 대역폭은 즉시 계산되지 않고 정해진 간격과 일정한 기간 후에 다시 계산되므로 항상 근사치입니다 . 예를 들어 메트릭 간격이 기본값인 30 초이며 서버가 29 초 동안 유휴 상태인 경우 30 초에서 클라이언트는 1 초 동안 30 배의 대역폭을 사용할 가능성이 있습니다 .

- 서비스 품질 대역폭 통계는 서버가 동적으로 구성되면 항상 잃게 됩니다 . 또한 서비스 품질 한계는 이전의 사용하지 않는 구성의 연결이 있는 스레드에는 집행되지 않습니다 . 이는 대역폭 관리자 스레드가 오직 사용 중인 구성의 대역폭 통계만 계산하기 때문입니다 . 클라이언트가 오래 동안 소켓을 닫지 않고 사용 상태를 유지하여 서버가 이를 무효화시키지 않는 경우 이는 서버가 동적으로 재구성된 후에도 서비스 품질의 대상이 되지 않을 수 있습니다 .

- 동시 연결은 계산 세밀도는 가상 서버의 경우 가상 서버 클래스와 전역 서버 인스턴스와는 다른 정도로 계산됩니다 . 개별 가상 서버용 연결 카운터는 요청이 파싱되고 해당 가상 서버로 라우팅되면 즉시 자동적으로 증가됩니다 . 또한 해당 요청용 응답 처리가 완료되면 자동으로 기록됩니다 . 따라서 가상 서버 연결 통계는 항상 모든 인스턴스에 대하여 정확하게 됩니다 .

  그러나 가상 서버 클래스 및 전역 서버 인스턴스에 대한 연결 통계는 즉각적으로 업데이트되지 않습니다 . 이 통계는 매 재계산 간격마다 대역폭 관리자에 의하여 업데이트됩니다 . 가상 서버 클래스용 연결 계수는 해당 클래스의 모든 가상 서버 연결의 합이며 , 가상 서버 인스턴스 연결 계수는 모든 가상 서버 클래스 연결의 합입니다 .

이 값의 계산 방식으로 인하여 가상 서버의 연결 수는 항상 정확하며 (또한 연결 수에 대한 제한을 집행하는 경우 한계 이상의 연결이 발생할 수 없음), 가상 서버 클래스 및 서버 인스턴스 값은 오직 매 간격마다 계산되므로 정확하지 않습니다 .

SNMP 기초

SNMP 는 네트워크 작동에 대한 데이터를 교환하는 데 사용하는 프로토콜입니다 . SNMP 를 사용하면 데이터가 관리된 장치와 네트워크 관리 스테이션 (NMS) 사이에서 전송됩니다 . 관리된 장치는 호스트 , 라우터 , 웹 서버 및 네트워크의 기타 서버 등 NSMP 에서 실행되는 모든 것입니다 . NMS 는 네트워크를 원격으로 관리하는 컴퓨터입니다 . 보통 NMS 소프트웨어는 수집된 데이터를 표시하는 그래프를 제공하거나 해당 데이터를 사용하여 서버가 특정 임계치 내에서 작동하는지 확인합니다 .

NMS 는 보통 하나 이상의 네트워크 관리 응용 프로그램이 설치된 고기능 워크스테이션입니다 . HP OpenView 등의 네트워크 관리 응용 프로그램은 웹 서버 등의 관리된 장치에 대한 정보를 그래픽을 표시합니다 . 예를 들어 기업에서 작동 중이거나 정지된 서버를 표시할 수 있으며 수신된 오류 메시지의 수와 유형을 표시할 수 있습니다 . Sun ONE 서버에서 SNMP 를 사용하는 경우 이 정보는 하위 에이전트와 마스터 에이전트 등 두 종류의 에이전트를 통하여 NMS 와 서버 사이에 전송됩니다 .

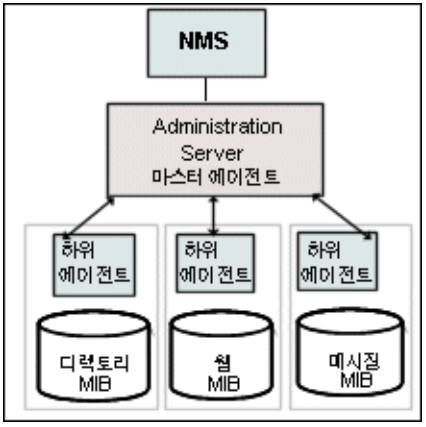
하위 에이전트는 서버에 대한 정보를 수집하며 해당 정보를 서버의 마스터 에이전트로 전달합니다 . Administration Server 를 제외한 모든 Sun ONE 서버에는 하위 에이전트가 있습니다 .

참고 SNMP 구성을 변경한 경우에는 반드시 Apply 버튼을 눌러 SNMP 하위 에이전트가 다시 시작되도록 해야 합니다 .

마스터 에이전트는 NMS 와 통신합니다 . 마스터 에이전트는 Administration Server 에 설치됩니다 .

호스트 컴퓨터에 여러 개의 하위 에이전트가 있을 수 있으나 마스터 에이전트는 하나만 있어야 합니다 . 예를 들어 동일한 호스트에 Directory Server, Sun ONE Web Server 및 Messaging Server 가 설치된 경우 아래에 보이는 것과 같이 각 서버의 하위 에이전트가 동일한 마스터 에이전트와 통신합니다 .

NMS(Network Management Station) 과 SNMP 에이전트

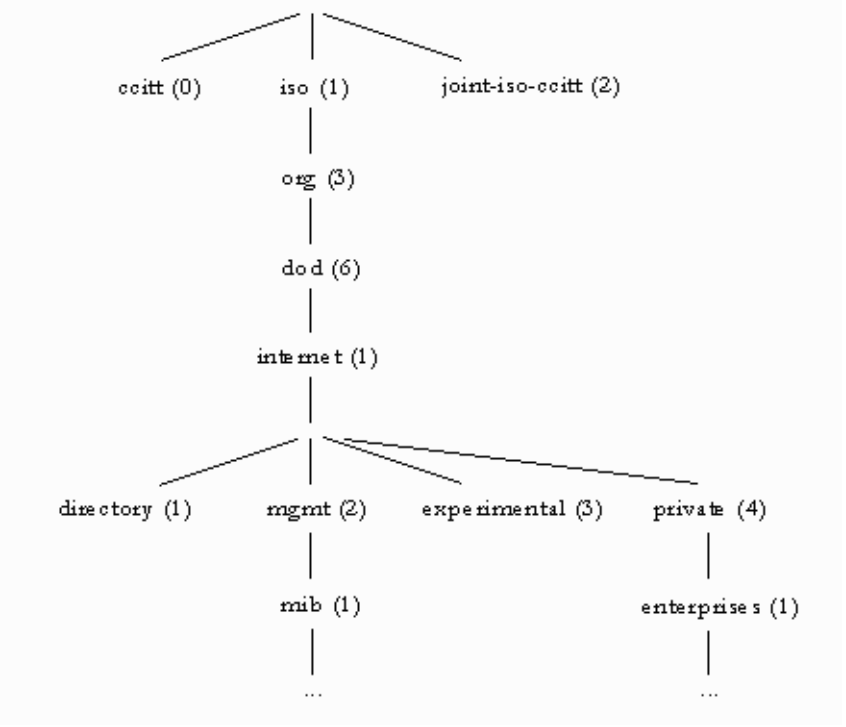


Sun ONE Web Server MIB

Sun ONE Web Server 에는 네트워크 관리에 관련된 변수가 저장됩니다 . 마스터 에이전트가 액세스할 수 있는 변수는 관리된 개체라고 합니다 . 이 개체는 MIB(Management Information Base) 라고 하는 트리 형식의 구조로 정의됩니다 . MIB 는 서버의 네트워크 구성 , 상태 및 통계에 대한 액세스를 제공합니다 . SNMP 를 사용하면 NMS(Network Management Station) 에서 이 정보를 볼 수 있습니다 .

서버의 MIB 에는 해당 특정 서버용 네트워크 관리에 관련된 변수 정의가 있습니다 . MIB 트리의 최상위 수준은 아래 그림에 보이는 것과 같습니다 .

MIB 트리의 최상위 수준



MIB 트리의 최상위 수준에는 인터넷 개체 ID 가 표시되며 , 여기에는 directory(1), mbmt(2), experimental(3) 및 private(4) 의 네 가지 하위 트리가 있습니다 . private(4) 하위 트리에는 enterprise(1) 노드가 있습니다 . enterprises(1) 의 각 하위 트리는 개별 엔터프라이즈에 지정되며 , 해당 엔터프라이즈는 자체의 MIB 확장자를 등록한 조직입니다 . 따라서 엔터프라이즈는 자체의 하위 트리에 제품 특정 하위 트리를 만들 수 있습니다 . 회사가 만든 MIB 는 enterprises(1) 노드 아래에 위치합니다 . Sun ONE MIB 는 enterprises(1) 노드 아래에 위치합니다 .

각 Sun ONE 서버 하위 에이전트는 SNMP 통신에서 사용할 MIB 를 제공합니다 . 서버는 이들 변수가 포함된 메시지 또는 트랩 을 송신하여 NMS(Network Management Station) 에 중요한 이벤트를 보고합니다 . NMS 는 또한 서버의 MIB 에서 데이터를 쿼리하거나 MIB 의 변수를 원격으로 변경할 수 있습니다 .

각 Sun ONE 서버에는 자체의 MIB(Management Information Base) 가 있습니다 . 모든 Sun ONE MIB 는 다음에 위치합니다 .

server_root/plugins/snmp

Sun ONE Web Server 의 MIB 는 webserv61.mib 라는 이름의 파일입니다 . 이 MIB 에는 Sun ONE Web Server 용 네트워크 관리에 관련된 다양한 변수의 정의가 포함됩니다 .

Sun ONE Web Server 6.1 MIB 에는 http 60 (iws60 OBJECT IDENTIFIER ::= {http 60}) 의 개체 식별자가 있으며 위치는 *server_root*/plugins/snmp 디렉토리입니다 .

Sun ONE Web Server MIB 를 사용하여 실시간으로 서버에 대한 관리 정보를 볼 수 있으며 서버를 모니터할 수 있습니다 . webserv61.mib 에 저장된 관리된 개체의 목록과 설명은 표에 보이는 것과 같습니다 .

표 11-1) webserv61.mib 관리된 개체 및 설명

관리된 개체	설명
iwsInstanceTable	Sun ONE Web Server 인스턴스 .
iwsInstanceEntry	Sun ONE Web Server 인스턴스
iwsInstanceIndex	서버 인스턴스 색인
iwsInstanceId	서버 인스턴스 식별자
iwsInstanceVersion	SunONE-WebServer/6.1 BB1-01/24/2001 17:15 (SunOS DOMESTIC) 등의 문자열
iwsInstanceDescription	서버 인스턴스의 설명
iwsInstanceOrganization	서버 인스턴스를 담당하는 조직
iwsInstanceContact	서버 인스턴스를 담당하는 개인의 연락처 정보
iwsInstanceLocation	서버의 위치
iwsInstanceStatus	서버 인스턴스의 상태
iwsInstanceUptime	서버가 실행된 시간
iwsInstanceDeathCount	서버 인스턴스 프로세스가 정지된 회수
iwsInstanceRequests	서버 인스턴스가 처리한 요청의 수
iwsInstanceInOctets	서버 인스턴스에 수신된 8 진수의 수 정보를 사용할 수 없는 경우 0 으로 표시 .
iwsInstanceOutOctets	서버 인스턴스가 전송한 8 진수의 수 정보를 사용할 수 없는 경우 0 으로 표시 .

표 11-1) webserv61.mib 관리된 개체 및 설명 *(계속)*

관리된 개체	설명
iwsInstanceCount2xx	서버 인스턴스가 발행한 200 수준 (성공) 응답의 수
iwsInstanceCount3xx	서버 인스턴스가 발행한 300 수준 (재지정) 응답의 수
iwsInstanceCount4xx	서버 인스턴스가 발행한 400 수준 (클라이언트 오류) 응답의 수
iwsInstanceCount5xx	서버 인스턴스가 발행한 500 수준 (서버 오류) 응답의 수
iwsInstanceCountOther	서버 인스턴스가 발행한 기타 (2xx, 3xx, 4xx 또는 5xx 제외) 응답의 수
iwsInstanceCount200	서버 인스턴스가 발행한 200 (요청 이행) 응답의 수
iwsInstanceCount302	서버 인스턴스가 발행한 302 (임시 이동) 응답의 수
iwsInstanceCount304	서버 인스턴스가 발행한 304 (수정 안 됨) 응답의 수
iwsInstanceCount400	서버 인스턴스가 발행한 400 (잘못된 요청) 응답의 수
iwsInstanceCount401	서버 인스턴스가 발행한 401 (권한 없음) 응답의 수
iwsInstanceCount403	서버 인스턴스가 발행한 403 (금지됨) 응답의 수
iwsInstanceCount404	서버 인스턴스가 발행한 404 (찾을 수 없음) 응답의 수
iwsInstanceCount503	발행된 503 (사용 불가) 응답의 수
iwsVsTable	Sun ONE Web Server 가상 서버
iwsVsEntry	Sun ONE Web Server 가상 서버 .
iwsVsIndex	가상 서버 색인
iwsVsId	가상 서버 식별자
iwsVsRequests	가상 서버가 처리한 요청의 수
iwsVsInOctets	가상 서버에 수신된 8 진수의 수
iwsVsOutOctets	가상 서버가 전송한 8 진수의 수

표 11-1) webserv61.mib 관리된 개체 및 설명 *(계속)*

관리된 개체	설명
iwsVsCount2xx	가상 서버가 발행한 200 수준 (성공) 응답의 수
iwsVsCount3xx	가상 서버가 발행한 300 수준 (재지정) 응답의 수
iwsVsCount4xx	가상 서버가 발행한 400 수준 (클라이언트 오류) 응답의 수
iwsVsCount5xx	가상 서버가 발행한 500 수준 (서버 오류) 응답의 수
iwsVsCountOther	가상 서버가 발행한 기타 (2xx, 3xx, 4xx 또는 5xx 제외) 응답의 수
iwsVsCount200	가상 서버가 발행한 200 (요청 이행) 응답의 수
iwsVsCount302	가상 서버가 발행한 302 (임시 이동) 응답의 수
iwsVsCount304	가상 서버가 발행한 304 (수정 안 됨) 응답의 수
iwsVsCount400	가상 서버가 발행한 400 (잘못된 요청) 응답의 수
iwsVsCount401	가상 서버가 발행한 401 (권한 없음) 응답의 수
iwsVsCount403	가상 서버가 발행한 403 (금지됨) 응답의 수
iwsVsCount404	가상 서버가 발행한 404 (찾을 수 없음) 응답의 수
iwsVsCount503	발행된 503 (사용 불가) 응답의 수
iwsProcessTable	Sun ONE Web Server 프로세스
iwsProcessEntry	Sun ONE Web Server 프로세스
iwsProcessIndex	프로세스 색인
iwsProcessId	운영 체제 프로세스 식별자
iwsProcessThreadCount	요청 처리 스레드의 수
iwsProcessThreadIdle	현재 유휴 상태인 요청 처리 스레드의 수
iwsProcessConnectionQueueCount	현재 연결 큐에 있는 연결의 수
iwsProcessConnectionQueuePeak	동시에 큐에 저장된 연결의 최대 수

표 11-1) `webserv61.mib` 관리된 개체 및 설명 *(계속)*

관리된 개체	설명
iwsProcessConnectionQueueMax	연결 큐에 허용된 최대 연결 수
iwsProcessConnectionQueueTotal	승인된 연결의 수
iwsProcessConnectionQueueOverflows	연결 큐 초과로 인하여 거부된 연결의 수
iwsProcessKeepaliveCount	현재 keepalive 큐에 있는 연결의 수
iwsProcessKeepaliveMax	keepalive 큐에 허용된 최대 연결 수
iwsProcessSizeResident	프로세스 상주 크기 (kb)
iwsProcessSizeVirtual	프로세스 크기 (kb)
iwsProcessFractionSystemMemoryUsage	시스템 메모리의 프로세스 메모리 조각
iwsListenTable	Sun ONE Web Server 청취 소켓
iwsListenEntry	Sun ONE Web Server 청취 소켓
iwsListenIndex	청취 소켓 색인
iwsListenId	청취 소켓 식별자
iwsListenAddress	소켓이 청취하는 위치의 주소
iwsListenPort	소켓이 청취하는 위치의 포트
iwsListenSecurity	암호화 지원
iwsThreadPoolTable	Sun ONE Web Server 스레드 풀
iwsThreadPoolEntry	Sun ONE Web Server 스레드 풀
iwsThreadPoolIndex	스레드 풀 색인
iwsThreadPoolID	스레드 풀 식별자
iwsThreadPoolCount	큐된 요청의 수
iwsThreadPoolPeak	동시에 큐에 저장된 요청의 최대 수
iwsThreadPoolMax	큐에 허용된 최대 요청 수
iwsInstanceStatusChange	`iwsInstanceStatusChange` 트랩은 `iwsInstanceStatus` 가 변경되었음을 표시합니다 .
iwsInstanceLoad1MinuteAverage	1 분간의 시스템 로드 평균
iwsInstanceLoad5MinuteAverage	5 분간의 시스템 로드 평균
iwsInstanceLoad15MinuteAverage	15 분간의 시스템 로드 평균
iwsInstanceNetworkInOctets	1 초에 네트워크에서 전송된 8 진수의 수

표 11-1) `webserv61.mib` 관리된 개체 및 설명 *(계속)*

관리된 개체	설명
iwsInstanceNetworkOutOctets	1 초에 네트워크에서 수신된 8 진수의 수
iwsCpuIndex	CPU 색인
iwsCpuId	CPU ID
iwsCpuIdleTime	CPU 유휴 시간
iwsCpuUserTime	CPU 사용자 시간
iwsCpuKernelTime	CPU 커넬 시간

SNMP 설정

일반적으로 SNMP 를 사용하려면 반드시 시스템에 마스터 에이전트와 최소한 하나의 하위 에이전트가 설치되고 실행되어야 합니다 . 하위 에이전트를 사용하기 전에 마스터 에이전트를 설치해야 합니다 .

SNMP 를 설정하는 방법은 시스템에 따라 다릅니다 . 상황에 따른 절차의 개요는 표 8.1 에 보이는 것과 같습니다 . 실제의 절차는 이 장의 뒤에서 자세히 설명합니다 .

시작하기 전에 두 가지를 확인해야 합니다 .

- 시스템에 이미 SNMP 에이전트 (운영 시스템의 원시 에이전트) 가 실행되고 있는가 ?
- 있는 경우 SNMP 에이전트가 SMUX 통신을 지원하는가 ? (AIX 플랫폼을 사용하는 경우에는 시스템이 SMUX 를 지원합니다 .)

이 정보를 확인하는 방법은 시스템 설명서를 참조하십시오 .

참고 Administration Server 에서 SNMP 설정을 변경하거나 , 새 서버를 설치하거나 , 기존 서버를 삭제한 후에는 반드시 다음과 같이 해야 합니다 .

- (Windows) Windows SNMP 서비스를 재시작하거나 컴퓨터를 재부팅합니다 .
- (UNIX) Administration Server 를 사용하여 SNMP 마스터 에이전트를 재시작합니다 .

표 2) SNMP 마스터 에이전트 및 하위 에이전트 사용 설정을 위한 절차 개요 .

시스템의 전제 조건	수행할 작업 . 이 작업은 다음 부분에서 자세히 설명합니다 .
• 현재 실행되는 원시 에이전트 없음	1. 마스터 에이전트를 시작합니다 . 2. 서버에 설치된 각 서버용 하위 에이전트를 사용하도록 설정합니다 .
• 현재 원시 에이전트 실행 • SMUX 없음 • 원시 에이전트를 사용하여 계속할 필요 없음	1. Administration Server 용 마스터 에이전트를 설치할 때 원시 에이전트를 중지합니다 . 2. 마스터 에이전트를 시작합니다 . 3. 서버에 설치된 각 서버용 하위 에이전트를 사용하도록 설정합니다 .
• 현재 원시 에이전트 실행 • SMUX 없음 • 원시 에이전트를 사용하여 계속	1. 프록시 SNMP 에이전트를 설치합니다 . 2. 마스터 에이전트를 시작합니다 . 3. 해당 프록시 SNMP 에이전트를 시작합니다 . 4. 마스터 에이전트 포트 번호가 아닌 포트 번호를 사용하여 원시 에이전트를 재시작합니다 . 5. 서버에 설치된 각 서버용 하위 에이전트를 사용하도록 설정합니다 .
• 현재 원시 에이전트 실행 • SMUX 지원	1. SNMP 원시 에이전트를 재구성합니다 . 2. 서버에 설치된 각 서버용 하위 에이전트를 사용하도록 설정합니다 .

프록시 SNMP 에이전트 사용 (UNIX/Linux)

이미 원시 에이전트가 실행되며 이를 Sun ONE Web Server 마스터 에이전트와 함께 계속 사용하려는 경우 프록시 SNMP 에이전트를 사용해야 합니다 . 시작하기 전에 원시 마스터 에이전트가 중단되었는지 확인합니다 . (자세한 정보는 시스템 설명서를 참조하십시오 .)

참고 프록시 에이전트를 사용하려면 이를 설치한 후 시작해야 합니다 . 또한 Sun ONE Web Server 마스터 에이전트가 실행되는 포트 번호가 아닌 다른 포트 번호를 사용하여 원시 SNMP 마스터 에이전트를 재시작해야 합니다 .

이 부분에서는 다음 항목에 대해 설명합니다 .

- 프록시 SNMP 에이전트 설치
- 프록시 SNMP 에이전트 시작
- 원시 SNMP 데몬 재시작

프록시 SNMP 에이전트 설치

SNMP 에이전트가 시스템에서 실행되며 원시 SNMP 데몬을 계속 사용하려면 다음과 같이 합니다 .

1. SNMP 마스터 에이전트를 설치합니다 . "SNMP 마스터 에이전트 설치 " 페이지 266 참조 .
2. 프록시 SNMP 를 설치 및 시작하고 원시 SNMP 데몬을 재시작합니다 . " 프록시 SNMP 에이전트 사용 (UNIX/Linux)" 페이지 263 참조 .
3. SNMP 마스터 에이전트를 시작합니다. "SNMP 마스터 에이전트 사용 설정 및 시작 " 페이지 267 참조 .
4. 하위 에이전트를 사용하도록 설정합니다 . " 하위 에이전트 사용 설정 " 페이지 272 참조 .

SNMP 프록시 에이전트를 설치하려면 서버 루트 디렉토리의 `plugins/snmp/sgat` 에 있는 `CONFIG` 파일 (다른 이름을 지정할 수 있음) 을 편집합니다 . 또한 MIB 트리와 프록시 SNMP 에이전트가 전달할 트랩을 포함해야 합니다 .

`CONFIG` 파일의 예는 다음과 같습니다 .

```
AGENT AT PORT 1161 WITH COMMUNITY public
SUBTREES 1.3.6.1.2.1.1,
        1.3.6.1.2.1.2,
        1.3.6.1.2.1.3,
        1.3.6.1.2.1.4,
        1.3.6.1.2.1.5,
        1.3.6.1.2.1.6,
        1.3.6.1.2.1.7,
        1.3.6.1.2.1.8
FORWARD ALL TRAPS;
```

프록시 SNMP 에이전트 시작

프록시 SNMP 에이전트를 시작하려면 명령 프롬프트에서 다음을 입력합니다 .

```
# sagt -c CONFIG&
```

원시 SNMP 데몬 재시작

프록시 SNMP 에이전트를 시작한 후 , `CONFIG` 파일에서 지정한 포트에서 원시 SNMP 데몬을 재시작해야 합니다 . 원시 SNMP 에이전트를 재시작하려면 명령 프롬프트에서 다음을 입력합니다 .

```
# snmpd -P port_number
```

여기에서 *port_number* 는 `CONFIG` 파일에 지정된 포트 번호입니다 . 예를 들어 Solais 플랫폼의 경우 앞에서 언급한 예제의 `CONFIG` 파일의 포트를 사용하는 경우 다음을 입력합니다 .

```
# snmpd -P 1161
```

SNMP 원시 에이전트 재구성

SNMP 데몬이 AIX 에서 실행되는 경우 SMUX 가 지원됩니다 . 따라서 마스터 에이전트를 설치할 필요는 없습니다 . 그러나 AIX SNMP 데몬 구성을 변경해야 합니다 .

AIX 는 여러 구성 파일을 사용하여 통신을 검사합니다 . 이 중 한 가지인 `snmpd.conf` 를 변경하여 SNMP 데몬이 SMUX 하위 에이전트에서 들어오는 메시지를 받도록 해야 합니다 . 더 자세한 내용은 `snmpd.conf` 용 온라인 설명서 페이지를 참조하십시오 . 각 하위 에이전트를 정의하는 줄을 추가해야 합니다 .

예를 들어 다음 줄은 `snmpd.conf` 에 추가할 수 있습니다 .

```
smux 1.3.6.1.4.1.1.1450.1 “” IP_address net_mask
```

IP_address 는 하위 에이전트가 실행되는 호스트의 IP 주소이며 *net_mask* 는 이 호스트의 네트워크 마스크입니다 .

참고 루프백 주소 127.0.0.1 을 사용하면 안 되며 , 실제의 IP 주소를 사용해야 합니다 .

SNMP 마스터 에이전트 설치

SNMP 마스터 에이전트를 구성하려면 반드시 Administration Server 인스턴스를 `root` 사용자로 설치해야 합니다. 그러나 `root` 가 아닌 사용자라도, 웹 서버 인스턴스에서 SNMP 하위 에이전트가 마스터 에이전트와 함께 작동하도록 구성하여 MIB 찾아 보기 등의 기본적인 SNMP 작업을 수행할 수 있습니다 .

Server Manager 를 사용하여 마스터 SNMP 에이전트를 설치하려면 다음과 같이 합니다 .

1. root 로 로그인합니다 .
2. 포트 161 에 SNMP 데몬 (`snmpd`) 이 실행되는지 확인합니다 .

   실행되는 SNMP 데몬이 없으면 단계 4 로 계속합니다 .

   SNMP 데몬이 실행 중이면 데몬을 시작하는 방법과 지원하는 MIB 트리를 확인합니다 .
3. SNMP 데몬이 실행 중이면 해당 프로세스를 종료합니다 .
4. Server Manager 에서 Global Settings 탭의 SNMP Master Agent Trap 페이지를 선택합니다 . Manager Entries 페이지가 표시됩니다 .
5. 네트워크 관리 소프트웨어를 실행하는 시스템의 이름을 입력합니다 .
6. 네트워크 관리 시스템이 트랩을 청취할 포트 번호를 입력합니다. (주로 사용하는 포트는 162 입니다 .) 트랩에 대한 더 자세한 내용은 " 트랩 대상 구성 " 페이지 271 을 참조하십시오 .
7. 트랩에서 사용할 커뮤니티 문자열을 입력합니다 . 커뮤니티 문자열에 대한 자세한 내용은 " 커뮤니티 문자열 구성 " 페이지 271 을 참조하십시오 .
8. OK 를 누릅니다 .
9. Server Manager 에서 Global Settings 탭의 SNMP Master Agent Community 페이지를 선택합니다 . Community Strings 페이지가 표시됩니다 .
10. 마스터 에이전트용 커뮤니티 문자열을 입력합니다 .
11. 커뮤니티용 작업을 선택합니다 .
12. OK 를 누릅니다 .

SNMP 마스터 에이전트 사용 설정 및 시작

마스터 에이전트 작업은 `CONFIG` 라는 에이전트 구성 파일에 정의됩니다 . Server Manager 를 사용하여 `CONFIG` 파일을 편집하거나 , 파일을 직접 편집할 수 있습니다 . SNMP 하위 에이전트를 사용 설정하기 전에 반드시 마스터 SNMP 에이전트를 설치해야 합니다 .

마스터 에이전트를 시작할 때 "System Error: Could not bind to port" 와 유사한 바인드 오류가 발생하면 `ps -ef | grep snmp` 를 사용하여 `magt` 가 실행되고 있는지 확인합니다 . 실행되는 경우 `kill -9` *`pid`* 명령을 사용하여 해당 프로세스를 종료합니다 . SNMP 용 CGI 가 다시 작동을 시작할 것입니다 .

이 부분에서는 다음 항목에 대해 설명합니다 .

- 다른 포트에서 마스터 에이전트 시작
- SNMP 마스터 에이전트 직접 구성
- 마스터 에이전트 CONFIG 파일 편집
- sysContact 및 sysLocation 변수 정의
- SNMP 마스터 에이전트 구성
- SNMP 마스터 에이전트 시작

다른 포트에서 마스터 에이전트 시작

Administration Interface 는 161 이 아닌 다른 포트에서 SNMP 에이전트를 시작하지 않을 것입니다 . 그러나 다음과 같이 다른 포트에서 직접 마스터 에이전트를 시작할 수 있습니다 .

1. /*server_root*/`plugins/snmp/magt/CONFIG` 를 편집하여 원하는 포트를 지정합니다 .
2. 다음과 같은 시작 스크립트를 실행합니다 .

   ```
   cd /server_root/https-admserv

   ./start -shell /server_root/plugins/snmp/magt/magt

   /server_root/plugins/snmp/magt/CONFIG

   /server_root/plugins/snmp/magt/INIT
   ```

마스터 에이전트가 원하는 포트에서 시작될 것입니다 . 그러나 사용자 인터페이스는 해당 에이전트가 실행되는 것을 감지할 수 있습니다 .

SNMP 마스터 에이전트 직접 구성

마스터 SNMP 에이전트를 직접 구성하려면 다음과 같이 합니다 .

1. 수퍼유저로 로그인합니다 .
2. 포트 161 에 SNMP 데몬 (snmpd) 이 실행되는지 확인합니다 .

   SNMP 데몬이 실행 중이면 데몬을 시작하는 방법과 지원하는 MIB 를 확인합니다 . 해당 프로세스를 종료합니다 .
3. 서버 루트 디렉토리의 `plugins/snmp/magt` 에 있는 `CONFIG` 파일을 편집합니다 .
4. (선택) `CONFIG` 파일에 `sysContact` 및 `sysLocation` 변수를 정의합니다 .

마스터 에이전트 CONFIG 파일 편집

`CONFIG` 파일은 마스터 에이전트가 작동하는 커뮤니티와 관리자를 정의합니다 . 관리자 값은 유효한 시스템 이름이나 IP 주소이어야 합니다 .

기본 `CONFIG` 파일의 예는 다음과 같습니다 .

```
COMMUNITY          public
                   ALLOW ALL OPERATIONS

MANAGER            manager_station_name
                   SEND ALL TRAPS TO PORT 162
                    WITH COMMUNITY public
```

sysContact 및 sysLocation 변수 정의

`CONFIG` 파일을 편집하여 `sysContact` 및 `sysLocation` MIB-II 변수를 지정하는 `sysContact` 및 `sysLocation`의 초기 값을 추가할 수 있습니다. 이 예의 `sysContact` 및 `sysLocation` 문자열은 인용 부호 안에 넣습니다 . 공백 , 줄바꿈 , 탭 등을 포함하는 문자열은 인용부호 안에 넣어야 합니다 . 또한 16 진수 표기법으로 값을 지정할 수 있습니다 .

sysContract 및 sysLocation 변수가 정의된 CONFIG 파일의 예는 다음과 같습니다.

```
COMMUNITY          public
                   ALLOW ALL OPERATIONS

MANAGER            nms2
                   SEND ALL TRAPS TO PORT 162
                   WITH COMMUNITY public

INITIAL            sysLocation "Server room
501 East Middlefield Road
Mountain View, CA 94043
USA_

INITIAL            sysContact "John Doe
email: jdoe@netscape.com"
```

SNMP 하위 에이전트 구성

SNMP 하위 에이전트를 구성하여 서버를 모니터할 수 있습니다.

SNMP 하위 에이전트를 구성하려면 다음과 같이 합니다.

1. Administration Server 에서 서버 인스턴스를 선택하고 Manage 를 누릅니다.
2. Monitor 탭을 선택합니다.
3. SNMP Subagent Configuration 을 선택합니다.
4. (UNIX 전용) Master Host 필드에 서버의 이름과 도메인을 입력합니다.
5. 운영 체제 정보를 포함하여 서버의 설명을 입력합니다.
6. 서버를 담당하는 조직을 입력합니다.
7. Location 필드에 서버의 절대 경로를 입력합니다.
8. Contact 필드에 서버를 담당하는 담당자의 이름과 연락처 정보를 입력합니다.
9. Enable the SNMP Statistics Collection 에 On 을 선택합니다.
10. OK 를 누릅니다.
11. Apply 를 누릅니다.
12. 서버를 재시작하여 변경을 적용하려면 Apply Changes 를 선택합니다.

SNMP 마스터 에이전트 시작

SNMP 마스터 에이전트를 설치하면 에이전트를 직접 시작하거나 Administration Server 를 이용하여 시작할 수 있습니다 .

SNMP 마스터 에이전트 직접 시작

마스터 에이전트를 직접 시작하려면 명령 프롬프트에서 다음을 입력합니다 .

```
# magt CONFIG INIT&
```

INIT 파일은 MIB-II 시스템 그룹으로부터의 정보를 포함하는 비휘발성 파일로 여기에는 시스템 위치와 연락처 정보가 있습니다 . INIT 파일이 없는 경우 마스터 에이전트를 처음 시작하면 파일이 만들어집니다 . CONFIG 파일에 잘못된 관리자 이름이 있는 경우 마스터 에이전트가 시작할 수 없습니다 .

비표준 포트에서 마스터 에이전트를 시작하려면 다음 중 한 가지 방법을 사용합니다 .

방법 1:CONFIG 파일에서 마스터 에이전트가 관리자로부터의 SNMP 요청을 청취할 각 인터페이스에 대한 전송 매핑을 지정합니다 . 전송 매핑을 사용하면 마스터 에이전트가 표준 포트뿐 아니라 비표준 포트의 연결을 수락합니다 . 마스터 에이전트는 또한 비표준 포트의 SNMP 트래픽을 수락합니다 . 대상 시스템의 한계에 의하여 정해진 동시 SNMP 의 최대 수에 따라 각 프로세스에 대한 개방 소켓 또는 파일 기술자의 수가 제한됩니다 . 전송 매핑 항목의 예는 다음과 같습니다 .

```
TRANSPORT          extraordinary   SNMP
                   OVER UDP SOCKET
                    AT PORT 11161
```

CONFIG 파일을 직접 편집한 후 , 명령 프롬프트에서 다음을 입력하여 마스터 에이전트를 직접 시작해야 합니다 .

```
# magt CONFIG INIT&
```

방법 2:/etc/services 파일을 편집하여 마스터 에이전트가 표준 포트뿐 아니라 비표준 포트의 연결을 수락하도록 합니다 .

Administration Server 를 사용하여 SNMP 마스터 에이전트 시작

Administration Server 를 사용하여 SNMP 마스터 에이전트를 시작하려면 다음과 같이 합니다 .

1. Administration Server 에 로그인합니다 .

2. Server Manager 에서 Global Settings 탭의 SNMP Master Agent Control 페이지를 선택합니다 . SNMP Master Agent Control 페이지가 표시됩니다 .
3. Start 를 누릅니다 .

   또한 SNMP Master Agent Control 페이지에서 SNMP 마스터 에이전트를 정지하고 재시작할 수 있습니다 .

SNMP 마스터 에이전트 구성

마스터 에이전트를 사용 설정하고 호스트 컴퓨터의 하위 에이전트를 사용 설정하면 호스트의 Administration Server 를 구성해야 합니다 . 여기에는 커뮤니티 문자열과 트랩 대상을 지정해야 합니다 .

커뮤니티 문자열 구성

커뮤니티 문자열은 SNMP 가 권한 부여에 사용하는 텍스트 문자열입니다 . 따라서 네트워크 관리 스테이션은 에이전트에 보내는 각 메시지에 커뮤니티 문자열을 함께 보냅니다 . 그런 후 에이전트는 네트워크 관리 스테이션이 정보에 대하여 인증되었는지 확인합니다 . 커뮤니티 문자열이 SNMP 패킷과 보내질 때에는 감추어지지 않으며 , 문자열은 ASCII 텍스트로 보내집니다 .

Server Manager 의 Community Strings 페이지에서 SNMP 마스터 에이전트용 커뮤니티 문자열을 구성할 수 있습니다 . 또한 특정 커뮤니티가 수행할 수 있는 SNMP 관련 작업을 정의합니다 . 또한 Server Manager 에서 이미 구성한 커뮤니티를 확인 , 편집 및 제거할 수 있습니다 .

트랩 대상 구성

SNMP 트랩은 SNMP 에이전트가 네트워크 관리 스테이션으로 송신하는 메시지입니다 . 예를 들어 , SNMP 에이전트는 인터페이스의 상태가 가동 (up) 에서 정지 (down) 로 변경될 때 트랩을 송신합니다 . SNMP 에이전트는 반드시 네트워크 관리 스테이션의 주소를 알고 있어야 트랩을 보낼 위치를 알 수 있습니다 . Sun ONE Web Server 에서 SNMP 마스터 에이전트용 트랩 대상을 구성할 수 있습니다 . 또한 이미 구성한 트랩 대상을 확인 , 편집 및 제거할 수 있습니다 . Sun ONE Web Server 를 사용하여 트랩 대상을 구성하는 경우 실제로는 `CONFIG` 파일을 편집하는 것입니다 .

하위 에이전트 사용 설정

Administration Server 와 함께 제공되는 마스터 에이전트를 설치한 후에는 반드시 에이전트를 시작하기 전에 서버 인스턴스용 하위 에이전트를 사용하도록 설정해야 합니다 . 마스터 에이전트 설치에 대한 자세한 내용은 "SNMP 마스터 에이전트 설치 " 페이지 266 를 참조하십시오 . Server Manager 를 사용하여 하위 에이전트를 사용하도록 설정할 수 있습니다 .

UNIX/Linux 플랫폼에서 SNMP 기능을 정지하려면 반드시 우선 하위 에이전트를 정지시킨 후 , 마스터 에이전트를 정지시킵니다 . 마스터 에이전트를 먼저 정지시키는 경우 하위 에이전트를 정지시킬 수 없게 될 수 있습니다 . 이러한 경우 마스터 에이전트를 다시 시작하고 하위 에이전트를 정지한 후 , 마스터 에이전트를 정지시킵니다 .

SNMP 하위 에이전트를 사용하도록 설정하려면 Server Manger 의 SNMP Subagent Configuration 페이지를 사용하고 SNMP Subagent Control 페이지에서 하위 에이전트를 시작합니다 . 자세한 내용은 온라인 도움말의 해당 부분을 참조하십시오 .

하위 에이전트를 사용하도록 설정한 후에는 SNMP Subagent Control 페이지나 Windows 의 Services Control Panel 에서 에이전트를 정지 또는 재시작할 수 있습니다 .

참고 SNMP 구성을 변경한 경우에는 반드시 Apply 버튼을 눌러 SNMP 하위 에이전트가 다시 시작되도록 해야 합니다 .

SNMP 메시지 이해

GET 과 SET 는 SNMP 에 의하여 정의되는 두 가지 유형의 메시지입니다 . GET 과 SET 메시지는 NMS(Network Management Station) 이 마스터 에이전트로 보내는 메시지입니다 . 이 중 한가지 또는 둘 모두를 Administration Server 에서 사용할 수 있습니다 .

SNMP 는 PDU(protocol data unit) 의 형태로 네트워크 정보를 교환합니다 . 이 유닛에는 웹 서버 등의 관리된 장치에 저장된 변수에 대한 정보가 들어 있습니다 . 이들 변수는 관리된 개체라고도 하며 , 필요한 경우 NMS 로 보고하는 값과 제목이 포함됩니다 . 서버가 NMS 로 보내는 프로토콜 데이터 유닛은 " 트랩 " 이라고도 합니다 . GET, SET 및 " 트랩 " 메시지를 사용하는 방법은 다음 예에서 설명합니다 .

NMS 가 시작한 통신 NMS 는 서버에서의 정보를 요청하거나 서버의 MIB 에 저장된 변수의 값을 변경합니다 . 예 :

1. NMS 는 Administration Server 마스터 에이전트에 메시지를 보냅니다 . 메시지는 데이터에 대한 요청 (GET 메시지) 일 수 있으며 MIB 의 변수를 설정하는 지시문 (SET 메시지) 일 수 있습니다 .
2. 마스터 에이전트는 메시지를 적절한 하위 에이전트로 전달합니다 .
3. 하위 에이전트는 데이터를 수신하거나 MIB 의 변수를 변경합니다 .
4. 하위 에이전트는 데이터 또는 상태를 마스터 에이전트에 보고하고 , 마스터 에이전트는 해당 메시지 (GET 메시지) 를 다시 NMS 로 전달합니다 .
5. NMS 는 네트워크 관리 응용 프로그램을 통하여 데이터를 텍스트 또는 그래픽으로 표시합니다 .

서버가 시작한 통신 . 중요한 이벤트가 발생하면 서버 하위 에이전트는 메시지 또는 " 트랩 " 을 NMS 로 보냅니다 . 예 :

1. 하위 에이전트는 마스터 에이전트에게 서버가 정지되었음을 알립니다 .
2. 마스터 에이전트는 메시지 또는 " 트랩 " 를 보내어 NMS 에 해당 이벤트를 보고합니다 .
3. NMS 는 네트워크 관리 응용 프로그램을 통하여 정보를 텍스트 또는 그래픽으로 표시합니다 .

12 장

이름 지정 구성과 리소스

구성요소 기반 Java™ 2 Platform, Enterprise Edition(J2EE™) 기술은 엔터프라이즈 개발 및 구현을 단순화하는 웹 서비스용 인프라를 제공합니다 .

이 장에서는 Sun ONE Web Server 에서 제공되는 J2EE 리소스에 대하여 설명하고 이들 리소스를 만들고 관리하는 방법에 대하여 살펴봅니다 .

Java 보안 및 영역 기반 인증에 대한 설명은 제 4 장 , " 웹 컨테이너 및 웹 응용 프로그램용 J2EE 기반 보안 " 을 참조하십시오 .

이 장에서는 다음 항목에 대해 설명합니다 .

- Java 사용 설정 및 해제
- JVM 설정 구성
- J2EE 이름 지정 서비스 및 리소스 설명
- JNDI(Java Naming and Directory Interface) 설명
- Java 기반 리소스 생성
- Java 기반 리소스 수정
- Java 기반 리소스 삭제

Java 사용 설정 및 해제

Java 는 전역적 , 즉 Sun ONE Web Server 인스턴스마다 사용 설정하거나 특정 가상 서버 클래스마다 사용 설정할 수 있습니다 . 기본적으로 Java 는 Sun ONE Web Server 에서 사용 설정되며 다음 줄이 `magnus.conf` 파일에 추가됩니다 .

Init fn=” load-modules” shlib=” <server-root>/bin/https/lib/libj2eeplugin.so”

또한 특정 가상 서버용으로 Java 를 사용할 수 있습니다 . Java 를 설정하면 서버가 해당 가상 서버 클래스용 `obj.conf` 파일을 필요한 J2EE 지시문을 포함하여 업데이트합니다 .

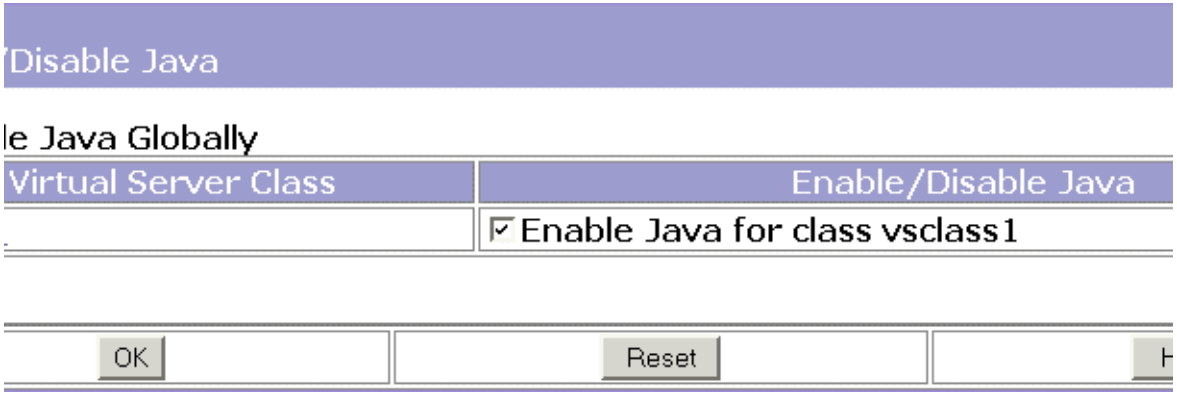
`obj.conf` 및 `magnus.conf` 파일에 대한 자세한 내용은 Sun ONE Web Server 6.1 *Administrator's Configuration File Reference* 와 Sun One Web Server 6.1 *NSAPI Programmer's Guide* 를 참조하십시오 .

예를 들어 전체 서버 또는 특정 가상 서버 클래스가 오직 정적 컨텐츠만 제공하는 등의 일부 경우 Java 를 전역적으로 또는 해당 클래스에 대하여 사용하지 않도록 하는 것이 좋을 때가 있습니다 .

Java 사용을 설정 또는 해제하려면 다음과 같이 합니다 .

1. Server Manager 에 액세스하고 Java 탭을 선택합니다 .
2. Enable/Disable Servelets/JSP 를 누릅니다 .

   Enable/Disable Servlets/JSP 인터페이스



3. Java 를 전역적으로 사용 설정 또는 해제하려면 Enable/Disable Java Globally 를 선택 또는 해제합니다 .

   또는

   특정 가상 서버에 대하여 Java 를 사용 설정 또는 해제하려면 해당 가상 서버 클래스의 Enable/Disable Java 선택란을 선택 또는 해제합니다 .
4. OK 를 누릅니다 .

JVM 설정 구성

Sun ONE Web Server 6.1 은 이전 버전과 달리 더 이상 독립형 JRE(Java Runtime Environment) 를 지원하지 않습니다 . 대신 서버를 사용하려면 JDK 1.4.1 이상이 필요합니다 . 서버를 설치할 때 기본 JDK 옵션을 선택하면 JDK(Java Development Kit) 버전 1.4.1_03 이 *<server-root>*`/bin/https/jdk` 디렉토리에 설치됩니다 .

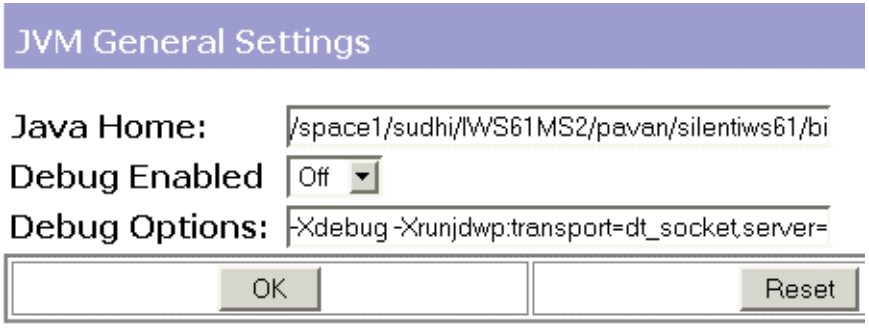
서버 인스턴스용 JVM(Java Virtual Machine) 설정을 구성할 수 있습니다 . 이 설정에는 Java 홈의 위치 , 컴파일러 옵션 , 디버깅 옵션 및 프로파일러 정보 등이 포함됩니다 . 이 설정을 구성하는 이유 중 하나는 성능을 향상시키기 위한 것입니다 . 성능에 대한 자세한 내용은 Sun ONE Web Server 6.1 *Performance Tuning, Sizing, and Scaling Guide* 를 참조하십시오 .

일반 설정 구성

JDK 의 위치를 변경하고 디버깅 옵션을 지정하려면 다음과 같이 합니다 .

1. Server Manager 에 액세스하고 Java 탭을 선택합니다 .
2. JVM General 을 누릅니다 .

   JVM General 인터페이스



3. Java Home 을 설정합니다 .

   Java Home 은 JDK(Java Developer's Kit) 가 설치된 위치의 경로입니다 . Sun ONE Web Server 는 Sun JDK 1.4.1_03 을 지원합니다 .

4. 디버깅을 사용할 것인지의 여부와 디버깅 옵션을 선택합니다 .

   디버깅 옵션 목록은 다음에서 사용할 수 있습니다 .
   `http://java.sun.com/products/jpda/doc/conninv.html#Invocation`

5. OK 를 누릅니다 .

경로 설정 구성

어떤 이유로 인하여 JVM 경로 설정을 구성해야 하는 경우가 있습니다 . 예를 들어 XML Parser 클래스 등의 시스템 클래스를 대체하기 위하여 시스템의 classpath 용 접미사를 선택하거나 프로덕션 환경에서의 환경 변수 부작용을 방지하기 위하여 환경 classpath 를 무시할 수 있습니다 .

Administration 인터페이스에서 JVM 의 경로 설정을 구성하려면 다음과 같이 합니다 .

1. Server Manager 에 액세스하고 Java 탭을 선택합니다 .
2. JVM Path Settings 를 누릅니다 .
3. 시스템의 classpath 에 대한 접미사를 선택합니다 .
4. 환경 클래스 경로를 무시할지 여부를 선택합니다 .

   classpath 를 무시하지 않으면 `CLASSPATH` 환경 변수가 읽혀서 Sun ONE Web Server classpath 에 추가됩니다 . `CLASSPATH` 환경 변수는 `classpathsuffix` 뒤인 제일 끝에 추가됩니다 .

   개발 환경을 위해서는 클래스 경로를 사용해야 합니다 . 생성 환경을 위해서는 환경 변수 부작용을 방지하기 위해 이 클래스 경로를 무시해야 합니다 .
5. 원시 라이브러리 경로 접두사 및 접미사를 설정합니다 .

   원시 라이브러리 경로는 원시 공유 라이브러리에 대한 Web Server 설치 상대 경로 , 표준 JRE 원시 라이브러리 경로 , 셸 환경 설정 (UNIX 의 경우 `LD_LIBRARY_PATH`) 및 `profiler` 요소에 지정된 모든 경로의 자동 구성된 연쇄입니다 . 이것이 합성되면 서버 구성에 명시적으로 나타나지 않습니다 .
6. OK 를 누릅니다 .

JVM 옵션 구성

Administration 인터페이스에서 JVM 명령줄 옵션을 설정하려면 다음과 같이 합니다 .

1. Server Manager 에 액세스하고 Java 탭을 선택합니다 .

2. JVM Options 를 누르고 필요한 사항을 변경합니다 .

   특정 JVM 옵션에 대한 내용은 다음을 참조하십시오 .

   `http://java.sun.com/docs/hotspot/VMOptions.html`

3. OK 를 누릅니다 .

JVM 프로파일러 구성

프로파일러를 사용하여 Sun ONE Web Server 에서 원격 프로파일링을 수행하면 서버측 성능의 병목 현상을 찾을 수 있습니다 .

Administration 인터페이스에서 JVM Profiler 를 구성하려면 다음과 같이 합니다 .

1. Server Manager 에 액세스하고 Java 탭을 선택합니다 .
2. JVM Profiler 를 누릅니다 .
3. classpath, 원시 라이브러리 경로 및 프로파일러를 사용할 것인지의 여부를 지정합니다 .
4. 해당 프로파일러의 JVM 옵션을 추가 , 삭제 또는 편집한 후 확인을 누릅니다 .

프로파일러에 대한 자세한 내용은 Sun ONE Web Server 6.1 *Programmer's Guide* 를 참조하십시오 .

J2EE 이름 지정 서비스 및 리소스 설명

웹 응용 프로그램은 리소스 관리자 , 데이터 소스 (SQL 데이터소스 등), 전자우편 세션 및 URL 연결 팩토리 등 다양한 리소스에 액세스할 수 있습니다 . J2EE 플랫폼은 JNDI(Java Naming and Directory Interface) 서비스를 통하여 응용 프로그램이 이러한 리소스를 사용할 수 있도록 합니다 .

Sun ONE Web Server 에서는 다음의 J2EE 리소스를 만들고 관리할 수 있습니다 .

- JDBC 데이터소스
- JDBC 연결 풀
- Java 전자우편 세션
- 사용자 정의 리소스
- 외부 JNDI 리소스

JDBC 데이터소스

JDBC Datasource 는 sun ONE Web Server 를 이용하여 만들고 관리할 수 있는 J2EE 리소스입니다 .

JDBC API 는 관련 데이터베이스 시스템과의 연결을 위한 API 입니다 . JDBC API 는 두 부분으로 이루어집니다 .

- 응용 프로그램 구성요소가 데이터베이스에 액세스하는 데 사용하는 응용 프로그램 수준 인터페이스 .
- JDBC 드라이버를 F2EE 플랫폼에 연결하는 서비스 제공업체 인터페이스 .

JDBC Datasource 개체는 Java 프로그래밍 언어에서 데이터 소스를 구현하는 것입니다 . 간단히 말하면 데이터 소스는 데이터를 저장하기 쉽게 하는 것입니다 . 이는 대규모 기업용의 복잡한 데이터베이스처럼 고차원적인 것일 수 있으며 행과 열로 이루어진 파일과 같이 간단한 것일 수도 있습니다 . JDBC Datasource 는 Sun ONE Web Server 를 통하여 만들고 관리할 수 있는 J2EE 리소스입니다 .

JDBC API 는 일련의 Java 용 클래스를 제공하며 , 여기에는 다양한 관계형 데이터베이스에 대한 액세스가 동일하도록 하는 표준 SQL 데이터베이스 액세스 인터페이스가 포함됩니다 .

JDBC 를 사용하면 SQL 문을 거의 모든 데이터베이스 관리 시스템 (DBMS) 으로 보낼 수 있습니다 . 또한 관계형 및 개체형 DBMS 모두에 대한 인터페이스로 사용됩니다 .

사용자 정의 리소스를 만드는 방법은 JDBC 리소스 생성을 참조하십시오 .

JDBC 연결 풀

JDBC 연결 풀은 데이터베이스에 연결되는 JDBC 그룹의 이름입니다 . 이 연결은 Sun ONE Web Server 를 시작할 때 풀에서 연결에 대한 요청을 처음 수행하면 만들어집니다 .

JDBC 연결 풀은 연결 풀을 만드는 데 사용하는 등록 정보를 정의합니다 . 각 연결 풀은 JDBC 드라이버를 사용하여 서버가 시작할 때 실제 데이터베이스로의 연결을 설정합니다 .

JDBC 기반 응용 프로그램이나 리소스는 풀에서 연결을 가져와 사용하며, 연결이 더 이상 필요하지 않으면 연결을 종료하고 연결 풀로 되돌립니다. 둘 이상의 JDBC 리소스가 동일한 풀 정의를 가리키는 경우에는 런타임에서 연결의 동일한 풀을 사용하게 됩니다.

새 JDBC 연결 풀을 만드는 방법은 새 JDBC 연결 풀 생성을 참조하십시오.

Java 전자우편 세션

JMS 대상 (destination) 은 Sun ONE Web Server 를 통하여 만들고 관리할 수 있는 J2EE 리소스입니다.

많은 인터넷 응용 프로그램에는 전자우편 통지를 보낼 수 있는 기능이 필요하므로 J2EE 플랫폼에 JavaMail API 와 응용 프로그램 구성요소가 인터넷 전자우편을 보낼 수 있는 JavaMail 서비스 제공자가 함께 포함되어 있습니다. JavaMail API 는 두 부분으로 이루어집니다.

- 응용 프로그램 구성요소가 전자우편을 송신하는 데 사용하는 응용 프로그램 수준 인터페이스.
- J2EE API 수준에서 사용되는 서비스 제공자 인터페이스.

Java Mail Session 은 Sun ONE Web Server 를 통하여 만들고 관리할 수 있는 J2EE 리소스입니다.

참고 Sun ONE Web Server 에는 Java Mail Sessions 를 만들 수 있는 Administration Server 인터페이스가 없습니다. 이 작업을 하려면 명령줄 인터페이스를 사용합니다. 명령줄 유틸리티를 사용하여 전자우편 리소스를 만드는 방법은 전자우편 리소스 생성을 참조하십시오.

사용자 정의 리소스

사용자 정의 리소스는 로컬 JNDI 저장소에 액세스합니다. 사용자 정의 서버 전체 리소스 개체 팩토리를 지정하는 방법은 `server.xml` 에 정의된 `customresource` 요소에 제공됩니다. 이러한 개체 팩토리는 `javax.naming.spi.ObjectFactory` 인터페이스를 구현합니다. 이 요소는 JNDI 이름 (기타 Sun ONE Web Server 리소스 등의 `jndiname` 하위 요소) 이 서버 전체 이름 공간, 해당 유형, 리소스 팩토리 클래스의 이름 및 이를 인스턴스화하는 표준 등록 정보 세트에서 사용되도록 연결합니다.

리소스 참조의 환경 참조가 `server.xml` 의 `customresource` 및 `externaljndiresource` 태그를 사용하여 정의한 서버 전체 리소스 구성에 연결되어야 합니다 . 응용 프로그램 구성요소의 동적 재구현이 JNDI 이름 지정 환경의 경우 문제가 됩니다 . Sun ONE Web Server 는 모든 응용 프로그램 특정 참조를 해제하고 모든 새 참조를 새로 설치된 응용 프로그램의 이름 지정 컨텍스트로 다시 바인드합니다 .

사용자 정의 리소스를 만드는 방법은 사용자 정의 리소스 생성을 참조하십시오 .

외부 JNDI 리소스

Sun ONE Web Server 에서 실행되는 응용 프로그램이 외부 JNDI 저장소에 저장된 리소스에 액세스해야 하는 경우가 있습니다 . 예를 들어 일반 Java 개체는 Java 스키마에 따라 LDAP 서버에 저장될 수 있습니다 . 사용자 정의 리소스를 사용하면 로컬 JNDI 저장소에 액세스할 수 있으나 , 외부 JNDI 저장소에 액세스하려면 반드시 외부 JNDI 리소스를 사용해야 합니다 . 외부 JNDI 팩토리는 반드시 `javax.naming.spi.InitialContextFactory` 를 구현해야 합니다 .

외부 JNDI 리소스를 만드는 방법은 외부 JNDI 리소스 생성을 참조하십시오 .

JNDI(Java Naming and Directory Interface) 설명

여기에서는 서로 다른 이름 지정 및 디렉토리 서비스에 액세스하는 데 사용하는 응용 프로그램 프로그래밍 인터페이스 (API) 인 JNDI(Java Naming and Directory Interface) 에 대하여 설명합니다 . J2EE 구성요소는 JNDI 조회 메소드를 시작하여 개체의 위치를 찾습니다 .

여기에서는 다음 항목에 대해 설명합니다 .

- J2EE 이름 지정 서비스
- 이름 지정 참조 및 바인드 정보
- J2EE 표준 구현 기술자 내의 이름 지정 참조
- JNDI 연결 팩토리

J2EE 이름 지정 서비스

JNDI 이름은 사용자에게 익숙한 개체 이름입니다 . 이 이름은 J2EE 서버가 제공하는 이름 지정 및 디렉토리 서비스에 의하여 해당 개체와 바인드됩니다 . J2EE 구성요소는 JNDI API 를 통하여 이 서비스에 액세스하므로 사용자는 보통 JNDI 이름으로 사용자에게 익숙한 개체 이름을 참조합니다 . 예를 들어 Oracle 데이터베이스의 JNDI 이름은 `jdbc/Oracle` 로 지정할 수 있습니다 . 이 데이터베이스가 시작되면 Sun ONE Web Server 는 구성 파일에서 해당 정보를 읽고 JNDI 데이터베이스 이름을 자동으로 이름 공간에 추가합니다 .

응용 프로그램 구성요소의 이름 지정 환경은 구현 및 어셈블리 과정에서 응용 프로그램 구성요소의 비즈니스 로직을 사용자 정의 할 수 있는 메커니즘입니다 . 응용 프로그램 구성요소의 환경을 사용하면 응용 프로그램 구성요소의 소스 코드에 액세스하거나 이를 변경하지 않고 응용 프로그램 구성요소를 사용자 정의할 수 있습니다 .

J2EE 컨테이너는 Web 응용 프로그램 구성요소의 환경을 구현하고 이를 JNDI 이름 지정 컨텍스트로 응용 프로그램 구성요소 인터페이스에 제공합니다 . 응용 프로그램 구성요소의 환경은 다음과 같이 사용됩니다 .

- Web 응용 프로그램 구성요소의 비즈니스 메소드는 JNDI 인터페이스를 사용하여 환경에 액세스합니다 . 응용 프로그램 구성요소 제공자는 응용 프로그램 구성요소가 런타임에 해당 환경에서 제공될 것으로 기대하는 모든 환경 요소를 구현 기술자에 선언합니다 .
- 컨테이너는 응용 프로그램 구성요소 환경을 저장하는 JNDI 이름 지정 컨텍스트를 구현한 것입니다 . 컨테이너는 또한 구현자가 각 응용 프로그램 구성요소의 환경을 만들고 관리할 수 있는 도구를 제공합니다 .
- 구현자는 컨테이너가 제공하는 도구를 사용하여 응용 프로그램 구성요소의 구현 기술자에 선언된 환경 항목을 초기화합니다 . 구현자는 환경 항목의 값을 설정 및 수정할 수 있습니다 .
- 컨테이너는 런타임에서 응용 프로그램 구성요소 인스턴스가 사용할 수 있는 환경 이름 지정 컨텍스트를 만듭니다 . 응용 프로그램 구성요소의 인스턴스는 JNDI 인터페이스를 사용하여 환경 항목의 값을 구합니다 .

각 응용 프로그램 구성요소는 자체의 환경 항목 세트를 정의합니다 . 동일한 컨테이너에 있는 응용 프로그램 구성요소의 모든 인스턴스는 동일한 환경 항목을 공유합니다 . 응용 프로그램 구성요소 인스턴스는 런타임에 환경을 수정할 수 없습니다 .

이름 지정 참조 및 바인드 정보

리소스 참조는 구현 기술자에 있는 요소로 구성요소의 리소스용으로 코드화된 이름을 구분합니다 . 더 정확히 말하면 해당 리소스를 위한 코드화된 이름 참조 연결 팩토리입니다 . 다음 부분에서 제공된 예제에서 리소스 참조 이름은 `jdbc/SavingsAccountDB` 입니다 .

리소스의 JNDI 이름과 리소스 참조의 이름은 동일하지 않습니다 . 이러한 이름 지정 방법을 사용하려면 구현 전에 이름 두 개를 매핑해야 하며 , 또한 리소스에서 구성 요소를 분리해야 합니다 . 이 분리로 인하여 이후에 구성요소가 다른 리소스에 액세스해야 하는 경우 코드 내의 이름을 변경할 필요는 없습니다 . 이러한 유연성으로 인하여 미리 존재하는 구성요소에서 쉽게 J2EE 응용 프로그램을 어샘블할 수 있습니다 .

Sun ONE Web Server 가 사용하는 J2EE 리소스에 대한 권장 JNDI 조회와 해당 연결 참조는 다음 표와 같습니다 .

표 1) JNDI 조회 및 해당 연결 참조

JNDI 조회 이름 .	연결된 참조
`java:comp/env`	응용 프로그램 환경 항목
`java:comp/env/jdbc`	JDBC DataSource 리소스
`java:comp/env/mail`	JavaMail 세션 연결 팩토리
`java:comp/env/url`	URL 연결 팩토리

J2EE 표준 구현 기술자 내의 이름 지정 참조

이름 지정 참조는 응용 프로그램이 지정된 이름 지정 컨텍스트에서 개체를 조회할 때 사용하는 문자열입니다 . 각 Web 응용 프로그램에는 이름 지정 컨텍스트가 있으며 참조는 표준 구성요소 구현 기술자 내에 구성됩니다 . 여기에서는 Sun ONE Web Server 에서 사용되는 표준 구현 기술자 기능에 대하여 설명합니다 . 여기에서는 다음 항목에 대해 설명합니다 .

- 응용 프로그램 환경 항목
- 리소스 참조
- 리소스 환경 참조

응용 프로그램 환경 항목

환경 항목은 <env-entry> 를 사용하여 정의하며 J2EE Web 응용 프로그램에 구현 시간 매개 변수를 지정하는 방법을 제공합니다 . 참고로 서브릿 컨텍스트 초기화 매개 변수는 <context-param> 을 사용하여 정의할 수 있으나 , <env-entry> 는 응용 프로그램 구현자가 이름 , 유형 및 값을 구체적으로 지정하여 응용 프로그램 매게 변수를 구성할 수 있으므로 더 많이 사용됩니다 .

J2EE 표준 개발 기술자에 지정된 <env-entry> 의 구문은 다음 예제에 보이는 것과 같습니다 .

<env-entry>

<description> Send pincode by mail </description>

<env-entry-name> mailPincode </env-entry-name>

<env-entry-value> false </env-entry-value>

<env-entry-type> java.lang.Boolean </env-entry-type>

</env-entry>

<env-entry-type> 태그는 해당 엔트리용의 유효한 클래스 이름을 지정합니다 . 서브릿 또는 JSP 에서 JNDI 를 사용하여 <env-entry> 를 조회하는 코드는 다음과 같습니다 .

```
Context initContext = new InitialContext();
Boolean mailPincode = (Boolean)
initContext.lookup("java:comp/env/mailPincode");
// one could use relative names into the sub-context
Context envContext = initContext.lookup("java:comp/env");
Boolean mailPincode = (Boolean)
envContext.lookup("mailPincode");
```

리소스 참조

팩토리는 필요할 때 다른 개체를 만드는 개체입니다 . 리소스 팩토리는 데이터베이스 연결이나 메시지 서비스 연결 등의 리소스 개체를 만듭니다 . 이들 개체는 표준 구현 기술자에 있는 <resource-ref> 를 사용하여 구성합니다 .

팩토리의 사용 예는 다음과 같습니다 .

예

javax.sql.DataSource 유형의 개체를 만드는 JDBC 연결 팩토리로의 참조 선언 :

<resource-ref>

<description> Primary database </description>

<res-ref-name> jdbc/primaryDB </res-ref-name>

<res-type> javax.sql.DataSource </res-type>

<res-auth> Container </res-auth>

</resource-ref>

`<res-type>` 은 해당 리소스 팩토리의 유효한 클래스 이름입니다 . `<res-auth>` 변수는 `Container` 또는 `Application` 에 값으로 지정될 수 있습니다 .

`Container` 가 지정되는 경우 웹 컨테이너가 리소스 팩토리를 JNDI 조회 레지스트에 바인드하기 전에 인증을 처리합니다 . `Application` 이 지정되는 경우 서브릿은 반드시 인증을 프로그램적으로 처리해야 합니다 . 서로 다른 리소스 팩토리가 다음과 같이 리소스 유형을 기술하는 별도의 하위 컨텍스트에서 조회됩니다 .

- `JDBC Javax.sql.DataSource` 팩토리용 `jdbc/`
- JavaMail `javax.mail.Session` 팩토리용 `mail/`
- `java.net.URL` 팩토리용 `url/`

인증을 처리하는 컨테이너가 있는 응용 프로그램 구성요소에서 JDBC 연결을 가져오는 코드는 다음과 같습니다 .

InitialContext initContext = new InitialContext();

DataSource source =

(DataSource) initContext.lookup("java:comp/env/jdbc/primaryDB");

Connection conn = source.getConnection();

참고로 , 이들 리소스 참조가 작동하려면 런타임에 `res-ref-name` 이 반드시 유효한 리소스 팩토리로 매핑되어야 합니다 .

리소스 환경 참조

리소스 환경 참조는 JNDI 조회를 통하여 리소스에 연결되어 관리된 개체에 액세스할 수 있는 방법을 제공합니다 . 표준 구현 기술자에 정의된 `<resource-env-ref>` 는 응용 프로그램이 리소스 요구 사항을 선언하도록 합니다 .

`<resource-env-ref>` 와 `<resource-ref>` 요소의 가장 큰 차이점은 특정 리소스 인증 요구 사항이 없다는 점으로 이들 요소는 리소스 팩토리 기술자의 보조가 필요합니다 .

예

```
<resource-env-ref>
   <description> My Topic </description>
   <res-env-ref-name> jdbc/MyTopic </res-ref-name>
   <res-env-ref-type> javax.jdbc.Topic </res-type>
</resource-env-ref>
```

JMS Topic 개체에 액세스하는 코드는 다음과 같습니다 .

```
InitialContext initContext = new InitialContext();
javax.jms.Topic myTopic = (javax.jdbc.Topic)
initContext.lookup("java:comp/env/jdbc/MyTopic");
```

초기 이름 지정 컨텍스트

Sun ONE Web Server 에서의 이름 지정 지원은 주로 J2EE 1.3 을 기반으로 몇 가지 기능이 추가되었습니다 . 응용 프로그램 구성요소가 `InitialContext` 를 통하여 초기 컨텍스트를 만들면 , Sun ONE Web Server 는 Web 응용 프로그램의 이름 지정 환경에 대한 핸들 역할을 하는 개체를 반환합니다 . 이 개체는 다시 `java:comp/env` 이름 공간용 하위 컨텍스트를 제공합니다 . 각 Web 응용 프로그램은 자체의 이름 공간이 있습니다 . 즉 , `java:comp/env` 이름 공간은 각 Web 응용 프로그램마다 있으며 하나의 Web 응용 프로그램 이름 공간에 바인드된 개체는 다른 Web 응용 프로그램에 바인드된 개체와 충돌하지 않습니다 .

JNDI 연결 팩토리

J2EE 웹 응용 프로그램의 경우 `web.xml` 파일에 있는 구현 기술자는 참조를 응용 프로그램 환경 엔트리 또는 리소스 관리자 (SQL Data Source 등) 연결 팩토리로 정의하는 용도의 자리 표시자입니다 . 응용 프로그램은 J2EE 컨테이너에서 제공되는 `JNDI InitialNamingContext` 를 사용하여 이러한 참조를 조회합니다 . 이렇게 하면 간단히 구현 기술자를 변경하여 , 즉 응용 프로그램의 소스 코드를 액세스하거나 변경하지 않고 해당 응용 프로그램을 다른 Web Server 환경으로 이식할 수 있습니다 .

연결 팩토리는 연결 개체를 만드는 개체로 , J2EE 구성요소가 리소스에 액세스할 수 있도록 합니다 . 데이터베이스용 연결 팩토리는 `javax.sql.DataSource object` 로 이는 `java.sql.Connection` 개체를 만듭니다 .

Sun ONE Web Server 에서는 다음 리소스 및 리소스 팩토리에 액세스할 수 있는 수단을 구성할 수 있습니다 .

- JDBC 연결 팩토리
- JavaMail 세션 연결 팩토리
- 일반 , 사용자 정의 리소스 개체 팩토리
- LDAP 등의 외부 리소스 저장소용 지원이 소

모든 Sun ONE Web Server 리소스 팩토리는 `server.xml` 의 `<resource></resources>` 태그 내에 지정되며 `jndiname` 속성을 사용하여 지정 (한 JNDI 이름을 가집니다 (jndiname 이 없는 `jdbconnectionpool` 은 제외). 이 속성은 서버 전체의 이름 공간에서 팩토리를 등록하는 데 사용합니다 . 구현자는 `sun-web.xml`의 `resource-ref`를 사용하여 사용자 정의 응용 프로그램 특정 리소스 참조 이름 (`resource-ref` 또는 `resource-env-ref` 요소에 선언) 을 이들 서버 전체의 리소스 팩토리로 매핑할 수 있습니다 . 이렇게 하면 주어진 응용 프로그램에 사용할 JDBC 리소스 (또한 기타 리소스 팩토리) 에 관련된 구현 시간을 결정할 수 있습니다 .

사용자 정의 리소스는 로컬 JNDI 저장소로 액세스하며 외부 리소스는 외부 JNDI 저장소로 액세스합니다 . 이 두 유형의 리소스 모두에는 사용자 지정 팩토리 클래스 요소 , JNDI 이름 속성 등등이 필요합니다 .

여기에서는 다양한 J2EE 리소스를 만드는 방법과 해당 리소스에 액세스하는 방법에 대하여 설명합니다 .

- Java 기반 리소스 생성
- Java 기반 리소스 수정

Java 기반 리소스 생성

여기에서는 Administration 인터페이스를 사용하여 다양한 J2EE 기반 리소스를 만드는 방법에 대하여 설명합니다 .

- 새 JDBC 연결 풀 생성
- JDBC 리소스 생성
- 사용자 정의 리소스 생성
- 외부 JNDI 리소스 생성

새 JDBC 연결 풀 생성

다음과 같이 새 JDBC 연결 풀을 만들 수 있습니다 .

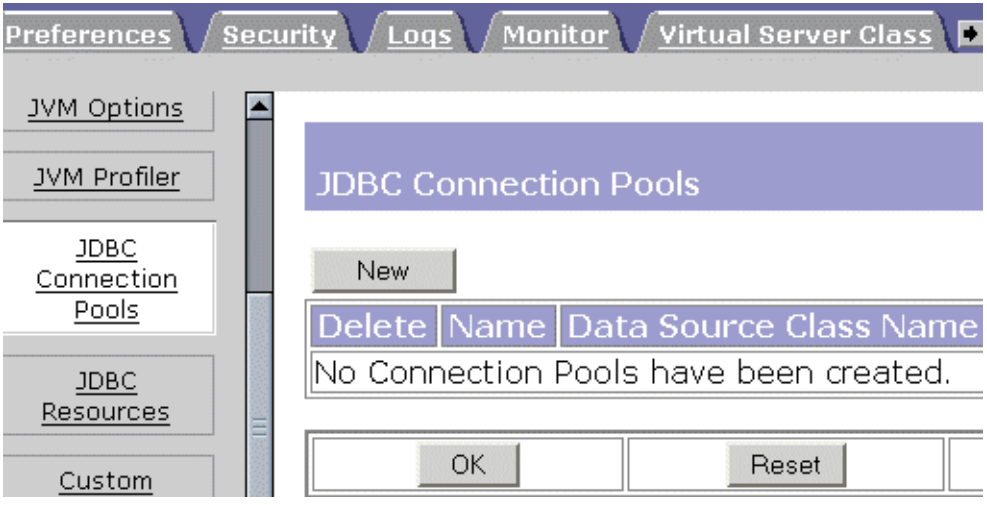
- Administration 인터페이스 사용
- 명령줄 인터페이스 사용

Administration 인터페이스 사용

Administration 인터페이스를 사용하여 새 JDBC 연결 풀을 만들려면 다음과 같이 합니다 .

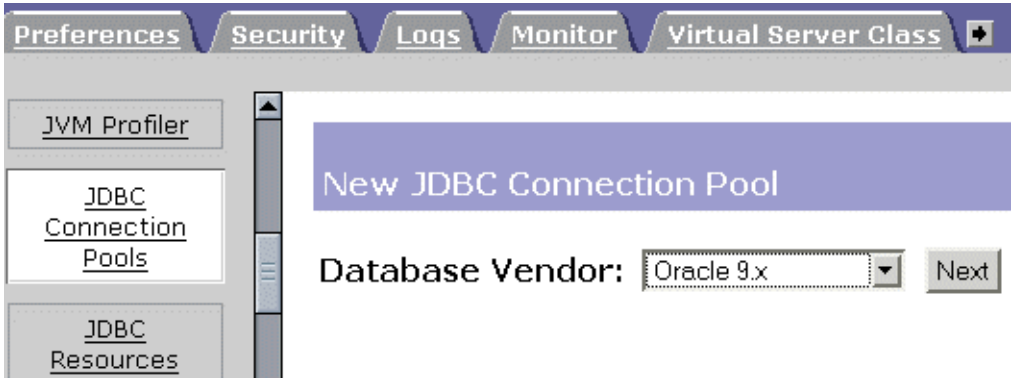
1. Server Manager 에 액세스하고 Java 탭을 선택합니다 .
2. JDBC Connection Pools 를 누릅니다 .
3. New 를 누릅니다 .

   JDBC Connection Pools Interface 페이지



4. Database Vendor 드롭 다운 목록에서 연결하려는 데이터베이스의 유형을 선택합니다 . 목록에 사용하는 DBMS 가 없는 경우 Other 를 선택합니다 .

   New JDBC Connection Pools Interface 페이지



5. Next 를 누릅니다 .

   Add New JDBC Connection Pool 페이지가 표시됩니다 .

6. 새 연결 풀의 등록 정보를 지정하고 확인을 누릅니다 .

   아래의 목록은 반드시 지정해야 하는 연결 풀 등록정보입니다 .

   General

   - **Pool Name**. 새 연결 풀의 이름을 입력합니다 .
   - **DataSource Classname**. 데이터 소스를 구현하는 클래스 이름으로 공급자가 지정합니다 . New JDBC Connection Pool 페이지의 Database Vendor 목록에서 Other 를 선택하는 경우 반드시 사용하려는 데이터 소스의 공급자가 지정한 클래스 이름을 입력해야 합니다 . 참고로 이 클래스는 반드시 `javax.sql.DataSource` 를 구현해야 합니다 .

   Properties

   표준 및 사유 JDBC 연결 풀 등록 정보를 지정합니다 . 이 등록 정보 중 많은 부분은 선택입니다 . 기본적으로 표준 등록 정보의 이름은 모두 입력되어 있습니다 . 어떤 표준 및 공급자 지정 등록 정보가 필요한지 결정하려면 데이터베이스 공급자의 설명서를 참조하십시오 .

   Pool Settings

- **Steady Pool Size**. 풀이 유지해야 하는 최소 연결 수를 지정합니다. 요청하는 스레드에 연결이 부여되면 연결은 풀에서 제거되므로 현재 풀 크기가 작아집니다 . 고정된 풀 크기는 또한 서버가 시작할 때 풀에 추가되는 연결의 수를 나타냅니다 .
- **Max Pool Size**. 임의 시간에 풀에서 허용할 수 있는 연결의 최대 수를 지정합니다 .
- **Pool Resize Quantity**. 풀이 고정된 풀 크기까지 줄어들면 풀의 크기가 배치 작업으로 조정됩니다 . 이 값에 따라 배치의 크기가 결정됩니다 . 이 값을 너무 크게 하면 연결 재사용이 지연되며 , 너무 작게 하면 효율성이 떨어집니다 . 참고로 풀의 용량은 한 번에 하나의 연결 단위로 증가되므로 이 필드는 풀 용량의 증가에 영향을 미치지 않습니다 .
- **Idle Timeout (secs)**. 풀에서 연결이 유휴 상태를 유지할 수 있는 최대 시간을 초 단위로 지정합니다 . 이 시간이 경과하면 풀 구현에 따라 해당 연결을 끊을 수 있습니다 .
- **Max Wait Time (milli secs)**. 호출자에게 연결 제한 시간이 지정될 때까지 대기하는 시간을 지정합니다 . 기본 대기 시간은 `long` 으로 , 호출자는 오랜 시간 동안 대기할 수 있습니다 . 이 값을 `0` 으로 설정하면 호출자는 연결이 사용 가능해질 때까지 차단됩니다 .

Connection Validation

- **Connection Validation Required**. 이 필드를 선택하면 연결이 응용 프로그램으로 전달되기 전에 확인 과정을 거칩니다 . 따라서 네트워크 고장 또는 데이터베이스 서버 장애로 인하여 데이터베이스를 사용할 수 없는 경우 서버가 자동으로 데이터베이스 연결을 재설정할 수 있습니다 . 연결 확인에는 추가의 오버헤드가 필요하므로 성능이 약간 저하될 수 있습니다 .
- **Validation Method**. 웹 서버가 데이터베이스 확인에 사용할 방법을 지정합니다 . 다음의 값을 선택합니다 .
  - **auto-commit**. 이 모드의 경우 쿼리문이 개별 트랜잭션으로 실행 및 커밋됩니다 . `auto-commit` 을 사용하지 않도록 설정하면 쿼리문은 커밋 또는 롤백 메커니즘에 의하여 종료될 수 있는 트랜잭션으로 그룹화됩니다 .
  - **meta-data**. 이 모드의 경우 연결의 데이터베이스가 해당 테이블, 저장된 프로시저 등에 대한 메타 정보를 제공할 수 있습니다 . meta-data 개체의 각 인스턴스에는 이에 연결된 특정 쿼리가 있습니다 . meta-data 개체는 해당 쿼리를 실행하고 결과를 캐시합니다 .
  - **table**. 이 방법의 경우 웹 서버가 사용자 지정 테이블에 대한 쿼리를 수행해야 합니다 .

- **Table Name**. Validation Method 드롭 다운 목록에서 유효성 검사 옵션 table 을 선택하는 경우 여기에 테이블 이름을 지정합니다 .
- **Fail All Connections**. 풀에 있는 모든 연결을 차단한 후 다시 설정하여 단일 연결이 실패했는지 확인하도록 할 것인지 지정합니다 . 선택하지 않는 경우 연결은 사용되는 경우에만 개별적으로 재설정됩니다 .

Transaction Isolation

트랜잭션이 사용하는 분리 수준에 따라 다른 사용자의 트랜잭션에 의한 변경에 대한 응용 프로그램의 민감도가 달라지며 , 따라서 이러한 변경에 대한 보호를 위하여 트랜잭션이 잠금을 유지해야 하는 시간이 결정됩니다 .

- **Transaction Isolation**. 이 연결용 트랜잭션 분리 수준을 선택할 수 있습니다 . 다음의 값을 선택합니다 .
  - **read-uncommitted**. dirty read 라도고 하며 , 이 옵션을 사용하면 데이터의 커밋 여부에 상관 없이 트랜잭션이 현재 데이터 페이지에 있는 모든 데이터를 읽을 수 있습니다 .
  - **read-committed**. 데이터의 공유 잠금을 다른 트랜잭션이 변경했지만 아직 커밋하지 않은 데이터를 읽을 수 없도록 하는 방식으로 적용합니다 . 커밋되지 않은 데이터는 읽을 수 없으므로 `read-committed` 분리로 실행되는 트랜잭션이 데이터를 다시 쿼리하면, 해당 데이터가 변경되거나 원래 쿼리의 범주에 해당하는 추가 데이터가 표시될 수 있습니다 .
  - **repeatable-read**. 쿼리에서 사용되는 모든 데이터에 잠금이 적용되도록 합니다 . 트랜잭션을 커밋하거나 롤백하지 않는 한 해당 트랜잭션이 사용하는 데이터를 다른 사용자가 수정할 수 없습니다 .
  - **serializable**. 잠금은 데이터 전체에 적용되므로 쿼리가 다시 수행되는 경우 , 첫 번째와 두 번째 쿼리 사이의 시간 동안 데이터가 변경되거나 추가 데이터 열이 표시되지 않습니다 .
- **Guarantee Isolation Level**. 풀에서 가져온 모든 연결에 일정한 수준의 분리가 적용되도록 합니다 . 예를 들어 해당 연결의 분리 수준이 지난 번 사용할 때 프로그램적으로 (예를 들어 `con.setTransactionIsolation`) 변경되는 경우 , 이 메커니즘은 지정된 분리 수준으로 복구됩니다 .

명령줄 인터페이스 사용

명령줄 인터페이스를 사용하여 새 JDBC 연결 풀을 만드는 방법은 부록 A, " 명령줄 유틸리티 " 의 JDBC 연결 풀 생성을 참조하십시오 .

JDBC 리소스 생성

JDBC 리소스는 또한 데이터 소스라고도 하며 `getConnection()` 을 사용하여 데이터베이스로의 연결을 만들 수 있습니다 . 다음 중 한 가지 방법으로 JDBC 리소스를 만듭니다 .

- 관리 인터페이스 사용
- 명령줄 인터페이스 사용

관리 인터페이스 사용

Administration 인터페이스를 사용하여 JDBC 리소스를 만들려면 다음과 같이 합니다 .

1. Server Manager 에 액세스하고 Java 탭을 선택합니다 .
2. JDBC Resources 를 누릅니다 .
3. New 버튼을 누릅니다 .
4. 다음 정보를 입력합니다 .
   - **JNDI Name**(필수). 응용 프로그램 구성 요소가 JDBC 리소스에 액세스할 때 반드시 사용해야 하는 JNDI 이름을 입력합니다 .
   - **Pool Name**(필수). 이 JDBC 리소스가 사용하는 연결 풀의 이름 (또는 ID) 을 목록에서 선택합니다 . 더 자세한 내용은 새 JDBC 연결 풀 생성을 참조하십시오 .
5. JDBC 리소스를 사용하려면 Data Source Enabled 드롭 다운 목록에서 On 을 선택합니다 .

   JDBC 리소스를 사용하지 않도록 설정하면 어떤 응용 프로그램 구성요소도 이에 연결할 수 없으나 , 구성은 서버 인스턴스에 유지됩니다 .
6. OK 를 누릅니다 .
7. Apply Changes 를 누릅니다 .

명령줄 인터페이스 사용

명령줄 인터페이스를 사용하여 새 JDBC 리소스를 만드는 방법은 부록 A, " 명령줄 유틸리티 " 의 JDBC 리소스 생성을 참조하십시오 .

사용자 정의 리소스 생성

다음과 같이 사용자 정의 리소스를 만들 수 있습니다 .

- 관리 인터페이스 사용
- 명령줄 인터페이스 사용

관리 인터페이스 사용

1. Server Manager 에 액세스하고 Java 탭을 선택합니다 .
2. Custom Resources 를 누릅니다 .
3. New 버튼을 누릅니다 .
4. 다음 정보를 입력합니다 .
    - **JNDI Name**(필수). 응용 프로그램 구성 요소가 사용자 정의 리소스에 액세스할 때 반드시 사용해야 하는 JNDI 이름을 입력합니다 .
    - **Resource Type**(필수). 사용자 정의 리소스의 인증된 유형을 입력합니다 .
    - **Factory Class**(필수). 사용자가 작성한 팩토리 클래스의 인증된 이름을 입력합니다 . 이에 따라 `javax.naming.spi.ObjectFactory` 가 구현됩니다 .
    - **Custom Resource Enabled**(선택). 런타임에 사용자 정의 리소스를 사용하도록 하려면 On 을 선택합니다 .
5. OK 를 누릅니다 .
6. Apply Changes 를 누릅니다 .

명령줄 인터페이스 사용

명령줄 인터페이스를 사용하여 새 사용자 정의 리소스를 만드는 방법은 부록 A, " 명령줄 유틸리티 " 의 사용자 지정 리소스 생성을 참조하십시오 .

외부 JNDI 리소스 생성

다음과 같이 외부 리소스를 만들 수 있습니다 .

- 관리 인터페이스 사용
- 명령줄 인터페이스 사용

관리 인터페이스 사용

1. Server Manager 에 액세스하고 Java 탭을 선택합니다 .
2. External JNDI Resources 를 누릅니다 .
3. New 버튼을 누릅니다 .
4. 다음 정보를 입력합니다 .
   - **JNDI Name**(필수). 응용 프로그램 구성 요소가 사용자 정의 리소스에 액세스할 때 반드시 사용해야 하는 JNDI 이름을 입력합니다 .
   - **Resource Type**(필수). 사용자 정의 리소스의 인증된 유형을 입력합니다 .
   - **Factory Class**(필수). 사용자가 작성한 팩토리 클래스의 인증된 이름을 입력합니다 . 이에 따라 `javax.naming.spi.ObjectFactory` 가 구현됩니다 .
   - **JNDI Lookup**(필수). 외부 저장소에서 조회할 JNDI 값을 입력합니다 . 예를 들어 , 외부 저장소로 연결하여 메일 클래스를 시험할 외부 리소스를 만드는 경우 JNDI 조회는 `cn=testmail` 이어야 합니다 .
   - **External Resource Enabled**(선택). 런타임에 외부 리소스를 사용하도록 하려면 `ON` 을 선택합니다 .
5. OK 를 누릅니다 .
6. Apply Changes 를 누릅니다 .

명령줄 인터페이스 사용

명령 줄을 인터페이스를 사용하여 새 사용자 정의 리소스를 만드는 방법은 부록 A, " 명령줄 유틸리티 " 의 외부 JNDI 리소스 생성을 참조하십시오 .

Java 기반 리소스 수정

여기에서는 Administration 인터페이스를 사용하여 Java 기반 리소스의 등록 정보를 수정하는 방법에 대하여 설명합니다 .

JDBC 연결 풀 수정

JDBC 연결 풀의 등록 정보를 수정하려면 다음과 같이 합니다 .

1. Server Manager 에 액세스하고 Java 탭을 선택합니다 .
2. JDBC Connection Pools 링크를 누릅니다 .
3. 편집하려는 JDBC 연결 풀에 해당하는 링크를 누릅니다 .
4. 필요한 설정을 변경합니다 .
5. OK 를 누릅니다 .

JDBC 리소스 수정

JDBC 리소스의 등록 정보를 수정하려면 다음과 같이 합니다 .

1. Server Manager 에 액세스하고 Java 탭을 선택합니다 .
2. JDBC Resources 링크를 누릅니다 .
3. 편집하려는 JDBC 리소스에 해당하는 링크를 누릅니다 .
4. 필요한 설정을 변경합니다 .
5. OK 를 누릅니다 .

사용자 정의 리소스 수정

사용자 정의 리소스의 등록 정보를 수정하려면 다음과 같이 합니다 .

1. Server Manager 에 액세스하고 Java 탭을 선택합니다 .
2. Custom Resources 링크를 누릅니다 .
3. 편집하려는 사용자 정의 리소스에 해당하는 링크를 누릅니다 .
4. 필요한 설정을 변경합니다 .
5. OK 를 누릅니다 .

외부 JNDI 리소스 수정

외부 JNDI 리소스의 등록 정보를 수정하려면 다음과 같이 합니다 .

1. Server Manager 에 액세스하고 Java 탭을 선택합니다 .
2. External JNDI Resources 링크를 누릅니다 .
3. 편집하려는 외부 JNDI 리소스에 해당하는 링크를 누릅니다 .
4. 필요한 설정을 변경합니다 .
5. OK 를 누릅니다 .

Java 기반 리소스 삭제

여기에서는 Administration 인터페이스를 사용하여 J2EE 기반 리소스를 삭제하는 방법에 대하여 설명합니다 .

- JDBC 연결 풀 삭제
- JDBC 리소스 삭제
- JDBC 리소스 삭제
- JDBC 리소스 삭제

JDBC 연결 풀 삭제

다음과 같은 방법으로 JDBC 리소스를 삭제할 수 있습니다 .

- Administration Server 사용
- 명령줄 유틸리티 사용

Administration Server 사용

Administration Server 를 사용하여 JDBC 연결 풀을 삭제하려면 다음과 같이 합니다 .

1. Server Manager 에 액세스하고 Java 탭을 선택합니다 .
2. JDBC Connection Pools 링크를 누릅니다 .
3. 삭제하려는 JDBC 연결 풀에 해당하는 선택란을 선택합니다 .
4. OK 를 누릅니다 .

명령줄 유틸리티 사용

사용 가능한 명령줄 옵션의 구문에 대한 내용은 명령줄 유틸리티를 참조하십시오 .

JDBC 리소스 삭제

다음과 같은 방법으로 JDBC 리소스를 삭제할 수 있습니다 .

- Administration Server 사용
- 명령줄 유틸리티 사용

Administration Server 사용

Administration Server 를 사용하여 JDBC 리소스를 삭제하려면 다음과 같이 합니다 .

1. Server Manager 에 액세스하고 Java 탭을 선택합니다 .
2. JDBC Resources 링크를 누릅니다 .
3. 삭제하려는 JDBC 리소스에 해당하는 선택란을 선택합니다 .
4. OK 를 누릅니다 .

명령줄 유틸리티 사용

사용 가능한 명령줄 옵션의 구문에 대한 내용은 명령줄 유틸리티를 참조하십시오 .

사용자 정의 리소스 삭제

다음과 같은 방법으로 사용자 정의 리소스를 삭제할 수 있습니다 .

- Administration Server 사용
- 명령줄 유틸리티 사용

Administration Server 사용

Administration Server 를 사용하여 사용자 정의 리소스를 삭제하려면 다음과 같이 합니다 .

1. Server Manager 에 액세스하고 Java 탭을 선택합니다 .
2. Custom Resources 링크를 누릅니다 .
3. 삭제하려는 사용자 정의 리소스에 해당하는 선택란을 선택합니다 .
4. OK 를 누릅니다 .

명령줄 유틸리티 사용

사용 가능한 명령줄 옵션의 구문에 대한 내용은 명령줄 유틸리티를 참조하십시오 .

외부 JNDI 리소스 삭제

다음과 같은 방법으로 외부 JDBC 리소스를 삭제할 수 있습니다 .

- Administration Server 사용
- 명령줄 유틸리티 사용

Administration Server 사용

Administration Server 를 사용하여 외부 JDBC 리소스를 삭제하려면 다음과 같이 합니다 .

1. Server Manager 에 액세스하고 Java 탭을 선택합니다 .
2. External JNDI Resources 링크를 누릅니다 .
3. 삭제하려는 외부 JNDI 리소스에 해당하는 선택란을 선택합니다 .
4. OK 를 누릅니다 .

명령줄 유틸리티 사용

사용 가능한 명령줄 옵션의 구문에 대한 내용은 명령줄 유틸리티를 참조하십시오 .

제 4 부

가상 서버 및 서비스 관리

13 장

가상 서버 사용

이 장에서는 Sun ONE Web Server 를 사용하여 가상 서버를 설정하고 관리하는 방법에 대하여 설명합니다 .

이 장의 내용 :

가상 서버 개요

가상 서버를 사용할 때 단일하게 설치된 서버로 회사 또는 개별 도메인 이름, IP 주소 및 일부 서버 모니터 기능을 제공할 수 있습니다 . 사용자의 경우 하드웨어와 기본적인 웹 서버 유지 보수를 제공해야 하지만 , 거의 자신의 전용 웹 서버를 가지고 있는 것과 같습니다 .

참고 가상 서버를 사용하지 않을 경우 Class Manager 의 항목을 사용하여 웹 서버 인스턴스에 대한 내용 , 프로그램 및 기타 기능을 구성합니다 . 웹 서버를 설치한 경우 인스턴스용 기본 가상 서버가 만들어집니다 . 가상 서버 사용자 인터페이스를 사용하여 이 기본 가상 서버의 내용과 서비스를 관리합니다 .

가상 서버를 설정하려면 다음 사항을 설정해야 합니다 .

- 가상 서버 클래스
- 청취 소켓
- 가상 서버

가상 서버용 설정은 `server.xml` 파일에 저장되어 있으며 , 이 파일은 *server_root/server_ID*/`config` 디렉토리에 있습니다. 가상 서버를 사용하기 위하여 이 파일을 편집할 필요는 없으나 , 이 파일에 대하여 더 자세히 알고 싶은 경우에는 Sun ONE Web Server 6.1 *Administrator's Configuration File Reference* 를 참조하십시오 .

이 부분에서는 다음 항목에 대해 설명합니다 .

- 복수 서버 인스턴스
- 가상 서버 클래스
- 청취 소켓
- 가상 서버
- 요청 처리용 가상 서버 선택
- 문서 루트
- 로그 파일
- 이전 릴리스에서 가상 서버 이전

복수 서버 인스턴스

Sun ONE Web Server 의 지난 릴리스에서는 가상 서버의 고유한 구성 정보에 유연성이 없었습니다 . 사용자는 서버가 별도의 구성 정보를 갖도록 하는 직접적 수단을 확보하고자 별도의 서버 인스턴스를 만드는 경우가 많았습니다 . Sun ONE Web Server 6.0 버전은 각 가상 서버 클래스에 대해 별도의 구성 정보를 도입했습니다 . 복수 서버 인스턴스가 여전히 지원되지만 여러 서버가 별도의 구성 정보를 갖도록 하는 것이 목적이라면 가상 서버가 더 나은 선택입니다 .

가상 서버 클래스

가상 서버는 클래스로 그룹화됩니다 . 클래스를 사용하여 동시에 유사한 가상 서버를 구성할 수 있으므로 각각을 별도로 구성할 필요가 없습니다 . 한 클래스의 모든 가상 서버가 동일한 기본 구성 정보를 공유한다고 해도 변수를 설정하고 가상 서버별로 구성을 변경할 수 있습니다 . 가상 서버가 구성 정보를 공유하지 않게 하려면 가상 서버 클래스별로 단일한 가상 서버를 만들 수 있습니다 . 그러나 , 가상 서버가 유사한 등록정보를 공유하면 클래스로 그룹화하여 함께 구성할 수 있습니다 .

예를 들어 , 인터넷 서비스 제공자 (ISP) 를 대상으로 하고 고객에게마다 다른 가격으로 다른 호스팅 수준을 제공하려 한다면 고객에게 여러 가지 가상 서버 클래스를 설정할 수 있습니다 . 하나의 가상 서버 클래스에 대해 Java 서브릿 및 JSP 를 사용함으로 설정하고 그보다 저렴한 가상 서버 클래스에 대해서는 Java 서브릿 및 JSP 를 사용안함으로 설정할 수 있습니다 .

가상 서버 클래스에 이름을 지정하고 문서 루트를 설정하여 가상 서버 클래스를 만들 수 있습니다 . 여기에서 클래스에 속하는 모든 가상 서버는 기본적으로 문서 루트를 갖습니다 . 클래스 내의 각 가상 서버가 클래스의 문서 루트 내부에 별도의 문서 루트를 갖도록 `$id` 변수를 사용할 수 있습니다 . 더 자세한 내용은 " 문서 루트 " 페이지 309 를 참조하십시오 .

가상 서버의 클래스를 만든 후 해당 클래스와 서비스를 연결합니다 . 가상 서버 클래스에 대해 다음 서비스 유형을 사용 설정 또는 구성할 수 있습니다 .

- 프로그램 , 프로그램으로 서버 확장 참조
- 내용 관리 , 내용 관리 참조
- 구성 스타일 , 구성 스타일 적용 참조

obj.conf 파일

클래스의 모든 가상 서버는 가상 서버 클래스에 대한 정보를 저장하는 `obj.conf` 파일을 공유합니다 . 해당 정보 중 일부는 변수로 저장되어 개별 가상 서버가 신속하게 특정 변수 값을 교체할 수 있습니다 .

`obj.conf` 및 변수에 대한 더 자세한 내용은 *NSAPI Programmer's Guide 를 참조하십시오* . 사용자 인터페이스의 변수 사용에 대한 더 자세한 내용은 " 변수 사용 " 페이지 313 를 참조하십시오 .

클래스의 가상 서버

클래스에 속하는 가상 서버는 해당 클래스의 구성원이라고 합니다 . 일부 가상 서버 설정은 클래스의 모든 가상 서버에 대해 구성되며 일부는 개별적으로 구성됩니다 . 이러한 설정은 Class Manager 의 Virtual Servers 탭에서 구성됩니다 . 더 자세한 내용은 제 14 장 , " 가상 서버 만들기 및 구성 " 을 참조하십시오 .

기본 클래스

Sun ONE Web Server 를 설치할 경우 설치자는 자동으로 `defaultclass` 라고 하는 단일 클래스를 만듭니다 . 여기에는 기본적으로 서버 인스턴스에 대해 하나의 가상 서버 구성원이 들어있습니다 . 기본 클래스에 추가 가상 서버를 추가할 수 있지만 클래스에서 기본 가상 서버를 삭제할 수는 없습니다 . 또한 기본 클래스를 삭제할 수 없습니다 .

청취 소켓

서버와 클라이언트 사이의 연결은 청취 소켓에서 이루어진다 . 만들어진 각 청취 소켓에는 IP 주소 , 포트 번호 , 서버 이름 및 기본 가상 서버가 있습니다 . 청취 소켓이 시스템에 주어진 포트의 모든 구성된 IP 주소에서 청취하도록 하려면 IP 주소에 대해 0.0.0.0, any, ANY 또는 INADDR_ANY 를 사용합니다 .

Sun ONE Web Server 를 설치하는 경우 청취 소켓 한 개 (ls1) 가 자동으로 만들어집니다 . 이 청취 소켓은 IP 주소 0.0.0.0 과 설치 도중 HTTP 서버 포트 번호로 지정한 포트 번호 (기본 값은 80) 를 사용합니다 . 기본 청취 소켓은 삭제할 수 없습니다 . 가상 서버를 사용하지 않으면 이 한 개 청취 소켓으로 충분합니다 . 그러나 가상 서버를 사용할 경우에는 가상 서버에 대해 여러 청취 소켓을 만들 수 있습니다 .

청취 소켓은 IP 주소와 포트 번호의 조합이므로 IP 주소는 같고 포트 번호는 다르거나 IP 주소는 다르고 포트 번호는 같은 여러 청취 소켓을 가질 수 있습니다 . 예를 들어 , 1.1.1.1:81 및 1.1.1.1:82 를 가질 경우 시스템이 이들 주소에 모두 응답하도록 구성되어 있기만 하면 추가적으로 1.1.1.1:81 과 1.2.3.4:81 을 가질 수 있습니다 .

또한 , 청취 소켓에서 여러 개의 승인자 스레드 (때로는 승인 스레드라고도 함) 을 지정합니다 . 승인자 스레드는 연결을 대기하는 스레드입니다 . 이 스레드가 연결을 승인하고 대기열에 배치되면 이후에 작업자 스레드에 의해 선택됩니다 . 이상적으로는 새로운 요청이 들어올 때 항상 사용 가능하도록 승인 스레드를 충분히 갖고 있으면 좋지만 시스템에 지나치게 부담이 되지 않도록 조정하는 것이 좋습니다 . 기본값은 1 입니다 . 시스템의 CPU 당 한 개 승인 스레드가 있는 것이 바람직한 원칙입니다 . 성능 저하 현상이 있으면 이 값을 조정할 수 있습니다 .

가상 서버

가상 서버를 만들려면 먼저 어떤 클래스에 속했는지 결정해야 합니다 . 다음으로 어떤 종류의 가상 서버를 원하는지 결정해야 합니다 . 가상 서버를 만들려면 가상 서버 ID, 하나 또는 하나 이상의 URL 호스트만 지정하면 됩니다 .

이 부분에서는 다음 항목에 대해 설명합니다 .

가상 서버의 유형

Sun ONE Web Server 릴리스 버전 6.0 이전에는 두 가지 종류의 가상 서버 , 즉 하드웨어와 소프트웨어가 있었습니다 . 하드웨어 가상 서버에는 고유한 IP 주소가 연결되어 있었습니다 . 소프트웨어 가상 서버에는 고유한 IP 주소가 없는 대신 고유한 URL 호스트가 있었습니다 .

Sun ONE Web Server 6.0 및 Sun ONE Web Server 6.1 에서는 이러한 개념이 더 이상 해당되지 않습니다 . 모든 가상 서버에는 URL 호스트가 지정되어 있습니다 . 그러나 , 가상 서버에 청취 소켓을 기준으로 하는 IP 주소와 연결되어 있을 수도 있습니다 .

새로운 요청이 들어오면 서버는 IP 주소 또는 Host 헤더의 값을 기준으로 어떤 가상 서버를 전송할지 결정합니다 . 먼저 IP 주소를 검토합니다 . 더 자세한 내용은 " 요청 처리용 가상 서버 선택 " 페이지 308 을 참조하십시오 .

IP 주소 기반 가상 서버

단일한 컴퓨터에서 여러 IP 주소를 갖기 위해서는 운영 체제를 통해 매핑하거나 추가 카드를 제공해야 합니다 . 운영 체제를 통해 여러 IP 주소를 설정하려면 네트워크 제어판 (Windows) 또는 `ifconfig` 유틸리티 (UNIX/Linux) 를 사용합니다 . 참고로 , `ifconfig` 를 사용하는 방법은 플랫폼마다 다릅니다 . 더 자세한 내용은 운영 체제 설명서를 참조하십시오 .

일반적으로 특정한 IP 주소에서 청취하는 청취 소켓을 만들어 IP 주소 기반 가상 서버를 만듭니다 . 청취 소켓의 기본 가상 서버는 IP 주소 기반 가상 서버입니다 . 가상 서버 구현 방법에 대한 더 자세한 내용은 " 가상 서버 구현 " 페이지 321 를 참조하십시오 .

URL 호스트 기반 가상 서버

가상 서버에 고유한 URL 을 부여하여 URL 호스트 기반 가상 서버를 설정할 수 있습니다 . 호스트 요청 헤더의 내용이 서버에 올바른 가상 서버를 지시합니다 .

예를 들어 , 고객 , aaa, bbb, ccc 에 대해 각 고객이 개별 도메인 이름을 갖도록 가상 서버를 설정하려면 먼저 각 고객의 URL, `www.aaa.com`, `www.bbb.com`, `www.ccc.com` 을 인지하도록 DNS 를 구성하고 사용하는 청취 소켓의 IP 주소로 확인합니다 . 그 다음 각 가상 서버의 URL 호스트를 올바른 설정 (예를 들어 , `www.aaa.com`) 으로 설정합니다 .

URL 호스트 기반 가상 서버는 호스트 요청 헤더를 사용하여 사용자에게 올바른 페이지를 지시하기 때문에 모든 클라이언트 소프트웨어가 그러한 가상 서버와 더불어 작동하는 것은 아닙니다 . HTTP Host 헤더를 지원하지 않는 클라이언트 소프트웨어는 작동하지 않습니다 . 이러한 클라이언트는 청취 소켓에 대해 기본 가상 서버를 받습니다 .

기본 가상 서버

URL 호스트 기반 가상 서버가 호스트 요청 헤더를 사용하여 선택됩니다 . 최종 사용자의 브라우저가 Host 헤더를 전송하지 않거나 서버가 지정된 Host 헤더를 찾을 수 없으면 기본 가상 서버가 요청을 서비스합니다 .

기본 가상 서버는 청취 소켓에 의해 설정됩니다 . 청취 소켓을 만들 때 기본 가상 서버를 지정합니다 . 항상 기본 가상 서버를 변경할 수 있습니다 .

요청 처리용 가상 서버 선택

서버가 요청을 처리하려면 청취 소켓을 통하여 요청을 접수한 후 요청을 올바른 가상 서버로 보내야 합니다 .

그 다음 가상 서버가 다음과 같이 선택됩니다 .

- 청취 소켓이 기본 가상 서버에 대해서만 구성된 경우 가상 서버가 선택됩니다 .
- 청취 소켓에 하나 이상의 가상 서버가 구성되어 있으면 요청 `Host` 헤더가 가상 서버의 URL 호스트에 일치됩니다 . `Host` 헤더가 없거나 일치하는 URL 호스트가 없으면 연결 그룹용 기본 가상 서버가 선택됩니다 .

가상 서버가 SSL 청취 소켓에 대해 구성되었고 해당 URL 호스트가 서버 시작시 인증서의 개체 패턴에 대해 확인되었으나 일치하지 않을 경우 경고가 발행되고 오류 로그가 작성됩니다 .

가상 서버가 결정된 후 서버는 가상 서버가 속한 가상 서버 클래스에 대해 `obj.conf` 파일을 실행합니다 . 서버가 `obj.conf` 에서 실행될 명령을 결정하는 방법에 대한 더 자세한 내용은 Sun ONE Web Server 6.1 *NSAPI Programmer's Guide* 를 참조하십시오 .

문서 루트

기본 문서 디렉토리 또는 문서 루트는 원격 클라이언트가 사용할 수 있게 되는 모든 가상 서버 파일을 담은 중앙 디렉토리입니다 .

문서 루트 디렉토리는 가상 서버의 파일에 대한 액세스를 제한하는 손쉬운 방법을 제공합니다 . 또한 URL 에 지정된 경로가 기본 문서 디렉토리에 대해 상대적이기 때문에 URL 을 변경하지 않고 새로운 디렉토리 (아마도 다른 디스크에 있는) 로 문서를 이동하는 일 또한 쉬워집니다 .

예를 들어 , 문서 디렉토리가 `C:\sun\servers\docs` 일 경우 `http://www.sun.com/products/info.html` 과 같은 요청이 서버에 `C:\sun\servers\docs\products\info.html` 의 파일을 찾으라고 지시합니다 . 문서 루트를 변경하면 (즉 , 모든 파일 및 하위 디렉토리를 이동하면) 모든 URL 을 새로운 디렉토리에 매핑하거나 클라이언트에게 새 디렉토리를 찾으라고 지시하는 대신 가상 서버가 사용하는 문서 루트를 변경하기만 하면 됩니다 .

Sun ONE Web Server 를 설치한 경우 웹 서버 인스턴스에 대해 문서 루트를 지정합니다 . 그러면 그것이 기본 클래스에 대한 문서 루트가 됩니다 . 클래스 수준에서 해당 디렉토리를 변경하거나 개별 가상 서버 수준에서 그것을 무시할 수 있습니다 .

클래스를 추가할 때도 문서 디렉토리를 지정할 필요가 없습니다 . 해당 디렉토리를 절대 경로입니다 . 그러나 단지 절대 경로만 입력하면 클래스에 속하는 모든 갓아 서버의 문서 루트의 기본값이 동일한 디렉토리가 됩니다 . 문서 루트 절대 경로의 끝에 변수 $id 를 포함시키면 모든 가상 서버가 기본적으로 *class_doc_root*/*virtual_server_ID* 의 기본 문서 루트를 갖습니다 . 예를 들어 , 클래스의 문서 디렉토리는 `/sun/servers/docs/$id` 이고 클래스에 속한 가상 서버 `vs1` 의 주 문서 디렉토리는 `/sun/servers/docs/vs1` 입니다 .

변수에 대한 더 자세한 내용은 " 변수 사용 " 페이지 313 을 참조하십시오 .

개별 가상 서버 수준에서 클래스의 기본 문서 디렉토리를 무시할 수도 있습니다 .

로그 파일

새로운 가상 서버를 만들면 기본적으로 로그 파일은 서버 인스턴스와 동일한 로그 파일입니다 . 대부분의 경우 각 개별 가상 서버가 자체 로그 파일을 가지려 합니다 . 이러한 설정을 하려면 각 가상 서버의 로그 경로를 변경할 수 있습니다 .

더 자세한 내용은 " 가상 서버 로그 설정 구성 " 페이지 333 을 참조하십시오 .

이전 릴리스에서 가상 서버 이전

iPlanet Web Server 4.1 버전의 가상 서버를 사용한다면 이전 도구를 사용하여 현재 릴리스로 이전할 수 있습니다 . 더 자세한 내용은 *설치 및 이전 설명서*를 참조하십시오 .

가상 서버와 함께 Sun ONE Web Server 기능 사용

Sun ONE Web Server 에는 SSL 및 액세스 제어와 같이 가상 서버와 함께 사용할 수 있는 많은 기능이 있습니다 . 이러한 기능 중 대다수가 모든 서버 , 서버 인스턴스 , 가상 서버 클래스 또는 개별 가상 서버에 대한 구성에 관한 것입니다 . 다음 부분에서 기능에 대해 설명하고 더 자세한 내용을 찾을 수 있는 위치 정보를 제공합니다 .

이 부분에서는 다음 항목에 대해 설명합니다 .

- 가상 서버와 함께 SSL 사용
- 가상 서버와 함께 액세스 제어 사용
- 가상 서버와 더불어 CGI 사용
- 가상 서버와 함께 구성 스타일 사용

가상 서버와 함께 SSL 사용

가상 서버에서 SSL 을 사용하려면 대부분 IP 주소 기반 가상 서버를 사용합니다 . 통상적인 포트는 443 입니다 . Sun ONE Web Server 는 요청을 전송할 URL 호스트를 결정하기 전에 요청을 읽어야 하기 때문에 URL 호스트 기반 가상 서버의 SSL 을 사용하기가 어렵습니다 서버가 요청을 읽으면 보안 정보가 교환되는 초기 핸드셰이크가 이미 발생합니다 .

유일한 예외는 URL 호스트 기반 가상 서버가 모두 동일한 서버 인증서를 비롯한 동일한 SSL 구성을 가지며 "wildcard certificates" 를 사용하는 경우입니다 . 더 자세한 내용은 제 6 장 , " 인증서 및 키 사용 " 을 참조하십시오 .

가상 서버와 함께 SSL 을 구현하는 한 가지 방법은 두 개의 청취 소켓을 갖는 것입니다 . 하나는 SSL 을 사용하여 포트 443 에서 청취하는 것이고 하나는 SSL 을 사용하지 않는 것입니다 . 사용자는 일반적으로 SSL 을 사용하지 않는 청취 소켓을 통해 가상 서버에 액세스합니다 . 보안 트랜잭션이 발생해야 할 경우 사용자는 웹 페이지에서 버튼을 한 번 눌러 보안 트랜잭션을 시작할 수 있습니다 . 그 후에 요청이 보안 청취 소켓을 통과합니다 .

SSL 트랜잭션은 SSL 을 사용하지 않는 트랜잭션보다 훨씬 느리기 때문에 이 디자인은 SSL 트랜잭션을 필요한 경우로만 제한합니다 . 보다 빠른 , SSL 을 사용하지 않는 연결은 나머지 시간에 사용됩니다 .

Sun ONE Web Server 및 가상 서버로 보안을 설정 및 사용하는 더 자세한 내용은 제 6 장 , " 인증서 및 키 사용 " 을 참조하십시오 . 가상 서버와 함께 예제 SSL 구성의 그림은 " 예제 2 보안 서버 " 페이지 323 을 참조하십시오 .

가상 서버와 함께 액세스 제어 사용

가상 서버와 더불어 가상 서버 하나당 액세스 제어를 설정할 수 있습니다 . 각 가상 서버가 LDAP 데이터베이스를 사용하는 사용자 및 그룹 인증을 가질 수 있도록 구성할 수도 있습니다 . 더 자세한 내용은 " 가상 서버용 액세스 제어 " 페이지 221 을 참조하십시오 .

가상 서버와 더불어 CGI 사용

가상 서버에서 CGI 를 사용할 수 있습니다 . 액세스 및 보안상의 이유로 구성할 수 있는 설정이 많이 있습니다 .

CGI 설정 및 사용에 대한 더 자세한 내용은 "CGI 프로그램 설치 " 페이지 349 를 참조하십시오 .

가상 서버와 함께 구성 스타일 사용

구성 스타일을 사용하면 다양한 가상 서버가 유지보수하는 특정한 파일 또는 디렉토리에 일련의 옵션을 쉽게 적용할 수 있습니다 . 구성 스타일 사용에 대한 더 자세한 내용은 구성 스타일 적용을 참조하십시오 .

가상 서버 사용자 인터페이스 사용

가상 서버를 만들고 편집하려면 사용자 인터페이스 또는 명령줄 유틸리티를 사용할 수 있습니다 .

가상 서버 관리용 사용자 인터페이스는 다음 세 부분으로 이루어집니다 .

- Server Manager 에는 서버에 전체적으로 (또는 모든 가상 서버에) 영향을 미치는 설정이 포함됩니다 .
- Class Manager 에는 단일 클래스 또는 클래스 내부의 가상 서버에 영향을 미치는 설정이 포함됩니다 .
- Virtual Server Manager 에는 개별 가상 서버의 설정이 포함됩니다 .

또한 , 개별 가상 서버를 가진 최종 사용자의 사용자 인터페이스가 사용 가능합니다 . 더 자세한 내용은 " 사용자가 개별 가상 서버를 모니터하도록 허용 " 페이지 317 를 참조하십시오 .

이 부분에서는 다음 항목에 대해 설명합니다 .

- Class Manager
- Virtual Server Manager
- 변수 사용
- 동적 재구성

Class Manager

Class Manager 에 액세스하려면 다음 단계를 따르십시오 .

1. Server Manager 에서 Virtual Server Class 탭을 누릅니다 .
2. Manage Classes 를 누릅니다 .
3. 클래스를 선택하고 Manage 를 누릅니다 .

서버의 트리 보기에서 클래스 이름을 누르거나 Server Manager 의 오른쪽 상단 모서리의 Class Manager 버튼 링크를 누릅니다 .

Virtual Server Manager

Server Manager 에 액세스하려면 다음과 같이 합니다 .

1. Class Manager 에서 Virtual Server 탭을 누릅니다 .
2. Manage Virtual Servers 를 누릅니다 .
3. 가상 서버를 선택하고 Manage 를 누릅니다 .

서버의 트리 보기에서 가상 서버 이름을 누를 수도 있습니다 .

명령줄 유틸리티 , `HttpServerAdmin` 을 사용하여 사용자 인터페이스을 사용하여 할 수 있는 동일한 가상 서버 작업을 수행할 수 있습니다 . 명령줄 유틸리티 `HttpServerAdmin` 에 대한 더 자세한 내용은 "HttpServerAdmin (가상 서버 관리)" 페이지 433 을 참조하십시오 .

변수 사용

각각의 값을 개별적으로 정의할 필요 없이 변수를 사용하여 클래스에 대해 가상 서버 특정 값을 부여할 수 있습니다 . 변수는 `obj.conf` 파일에서 정의됩니다 . 자체적으로 변수를 정의하지만 사용자 인터페이스는 그것을 인지하지 않습니다 . 사용자 인터페이스에서 가장 유용한 변수는 $id 입니다 . 이것은 가상 서버의 ID 를 나타냅니다 . 이 변수를 입력할 때마다 서버가 개별 가상 서버 ID 값을 교체합니다 .

때대로 마주치게 되는 기타 변수로는 $accesslog(각 가상 서버의 액세스 로그에 대한 경로) 와 $docroot(각 가상 서버의 문서 루트에 대한 경로) 가 있으나 $id 가 필드에 입력해야 하는 유일한 변수입니다 .

변수에 대한 더 자세한 내용은 Sun ONE Web Server 6.1 *NSAPI Programmer's Guide* 를 참조하십시오 .

동적 재구성

동적 재구성을 사용하여 변경 사항을 적용하기 위해 웹 서버를 중지하고 재시작할 필요 없이 라이브 웹 서버에 구성 변경을 할 수 있습니다 . 서버를 재시작하지 않고 `server.xml` 과 연결된 파일의 모든 설정과 속성을 동적으로 변경할 수 있습니다 . 따라서 가상 서버 사용자 인터페이스 내부에서 변경한 모든 사항이 서버를 재시작하지 않고도 적용될 수 있습니다 . 재구성 스크립트 또는 사용자 인터페이스를 사용하여 변경 후 서버를 동적으로 재구성할 수 있습니다 .

UNIX 플랫폼에서 동적 재구성 스크립트는 각 인스턴스의 디렉토리에 있는 '`reconfig`' 라는 셸 스크립트입니다 . 이 스크립트에는 명령줄 인수가 없습니다 . 서버 인스턴스 디렉토리에서 '`reconfig`' 를 입력하기만 하면 재구성 스크립트를 실행할 수 있습니다 .

Windows 에서 동적 재구성 스크립트는 각 인스턴스의 디렉토리에 있는 '`reconfig.bat` 라는 배치 파일입니다 . 명령줄 인수는 없습니다 . 서버 인스턴스의 디렉토리에서 '`reconfig`' 또는 '`reconfig.bat`' 를 입력하기만 하면 재구성 스크립트를 실행할 수 있습니다 .

이 스크립트는 실행되면 사용자 인터페이스와 유사하게 서버의 동적 재구성을 시작하며 재구성과 관련된 서버 메시지를 표시합니다 .

동적 재구성 화면에 액세스하려면 Server Manager, Class Manager 및 Virtual Server Manager 페이지의 오른쪽 상단 모서리에 있는 Apply 링크를 누른 다음 Apply Changes 페이지의 Load Configuration Files 버튼을 누릅니다 . 새로운 구성을 설치할 때 오류가 발생하면 이전 구성이 복원됩니다 .

가상 서버 설정

가상 서버를 설정하려면 다음 단계를 따르십시오 .

1. 청취 소켓을 삭제합니다 .
2. 가상 서버의 클래스를 만듭니다 .
3. 클래스에 대해 서비스를 구성합니다 .
4. 가상 서버 클래스에서 가상 서버를 만듭니다 .
5. 가상 서버를 구성합니다 .

청취 소켓을 만들 때는 기본 가상 서버 필드에 기존 가상 서버를 입력해야 합니다 . 서버를 설치할 때 만든 가상 서버를 사용한 다음 원할 경우 추가 가상 서버를 만든 후 다시 돌아가서 해당 가상 서버를 변경할 수 있습니다 .

청취 소켓 만들기

청취 소켓을 만들려면 다음 단계를 따르십시오 .

1. Server Manager 에서 Preferences 탭을 누릅니다 .
2. Add Listen Socket 을 누릅니다 .
3. 필드에 값을 입력합니다 .

   청취 소켓은 포트 번호 및 IP 주소의 고유한 조합이어야 합니다. IPV4 또는 IPV6 주소를 사용할 수 있습니다 . IP 주소 기반 가상 서버에 대해 청취 소켓을 만들려면 IP 주소는 0.0.0.0, ANY, any 또는 INADDR_ANY 여야 합니다 . 해당 포트의 모든 IP 주소에서 청취한다는 의미입니다 .

   이 청취 소켓에 대해 보안 (SSL) 을 사용함으로 설정할 수도 있습니다 .

   Server Name 필드는 서버가 클라이언트에 전송하는 URL 의 호스트 이름을 지정합니다 . 이에 따라 서버가 자동으로 생성하는 URL 이 달라지지만 , 서버에 저장된 디렉토리 및 파일용 URL 에는 영향을 미치지 않습니다 . 서버에서 별칭을 사용하는 경우 이 이름은 별칭이어야 합니다 .
4. OK 를 누릅니다 .

가상 서버 클래스 만들기

가상 서버 클래스를 만들려면 다음 단계를 따르십시오 .

1. Server Manager 에서 Virtual Server Class 탭을 누릅니다 .
2. Add 를 누릅니다 .
3. 클래스에 이름을 지정합니다 .
4. 클래스에 대해 문서 루트를 삽입합니다 .

   디렉토리가 이미 존재해야 합니다 . 이 클래스의 모든 가상 서버는 다른 방법으로 지정하지 않는 한 이 절대 경로로 문서 루트를 갖습니다 . 경로의 마지막 부분으로 `/$id` 를 사용하면 가상 서버 ID 에 대해 이름 지정된 문서 루트 폴더가 클래스의 문서 루트 경로 내부에 자동으로 만들어집니다 .

5. OK 를 누릅니다 .

가상 서버의 클래스를 만들었으면 클래스와 관련된 서비스를 선택합니다 . 더 자세한 내용은 내용 관리를 참조하십시오 .

가상 서버 편집 및 삭제

가상 서버 클래스의 설정을 편집하려면 다음 단계를 따르십시오 .

1. Server Manager 에서 Virtual Server Class 탭을 누릅니다 .
2. Edit Classes 를 누릅니다 .
3. 원하는 클래스 옆에 있는 드롭다운 목록에서 Edit 또는 Delete 를 선택합니다 .

   참고로 기본 클래스는 삭제할 수 없습니다 .
4. Document Root 필드를 사용하여 클래스의 기본 문서 루트에 대한 절대 경로로 변경합니다 .

   이 클래스의 가상 서버의 문서 루트는 기본적으로 이 디렉토리 내에서 만들어집니다 .
5. 이 가상 서버 클래스가 허용 언어 헤더 파싱을 사용하게 하려면 Accept Language 필드에 On 을 입력합니다 .

   기본값은 Off 입니다 .
6. 클래스와 연결된 CGI 기본값을 변경하려면 Advanced 를 누릅니다 .

   CGI 기본값이 있는 창이 나타납니다 . 필드를 편집하고 OK 를 눌러 Edit a Class 창으로 돌아갑니다 . Reset 버튼이 변경 사항을 복원합니다 .
7. OK 를 누릅니다 . 클래스가 변경되거나 삭제됩니다 .

가상 서버 클래스와 연결된 서비스 지정

가상 서버의 클래스를 다른 클래스와 구별하는 일부 특징은 해당 가상 서버 클래스에 대해 사용함으로 설정된 서비스입니다 . 예를 들어 , 하나의 가상 서버 클래스는 다른 클래스와 달리 CGI 를 사용할 수 있습니다 . 서비스 설정에 대한 더 자세한 내용은 내용 관리를 참조하십시오 .

가상 서버 만들기

가상 서버 클래스를 설정하면 가상 서버를 만들 수 있습니다 . 가상 서버가 특정한 가상 서버 클래스의 구성원이면 Class Manager 에서 가상 서버를 만듭니다 .

더 자세한 내용은 " 가상 서버 만들기 " 페이지 329 를 참조하십시오 .

가상 서버와 연결된 설정 지정

가상 서버 수준에서 일부 클래스 설정을 무시하고 추가 설정을 구성할 수도 있습니다 . Class Manager 에서 이러한 설정을 구성합니다 .

더 자세한 내용은 " 가상 서버 만들기 " 페이지 329 를 참조하십시오 .

사용자가 개별 가상 서버를 모니터하도록 허용

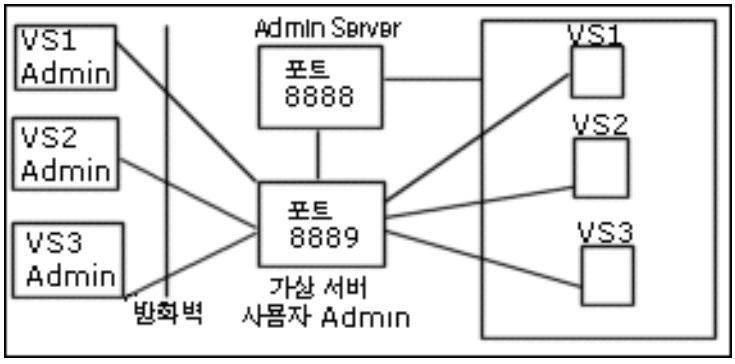
사용자가 자신의 가상 서버 설정을 확인하고 자신의 액세스 및 오류 로그를 보도록 하는 , 개별 가상 서버 관리자를 위한 특별한 사용자 인터페이스가 있습니다 . 예를 들어 , 3 개의 다른 부서에 대해 3 개의 다른 가상 서버가 있는 인트라넷이 있으면 각 부서는 각 부서의 설정과 로그 파일을 개별적으로 볼 수 있습니다 .

보안상의 이유로 이 관리 사용자 인터페이스는 관리 서버 포트 또는 웹 서버 인스턴스 포트와 별도의 포트에 있습니다 .

이 사용자 인터페이스는 관리 서버 내부의 가상 서버에서 실행됩니다 . 이 가상 서버는 기본적으로 설정되며 useradmin 이라고 합니다 . 사람들이 관리 서버 포트에 액세스하지 않고도 가상 서버 관리 사용자 인터페이스에 액세스할 수 있도록 관리 서버가 실행되는 청취 소켓과 별도로 관리 서버의 청취 소켓을 설정해야 합니다 .

다음 그림 , 은 개별 가상 서버의 관리자가 자신의 가상 서버에 대한 정보에 액세스하기 위해 useradmin 가상 서버에 액세스하는 모습을 보여줍니다 .

가상 서버 관리자의 사용자 인터페이스 구성



가상 서버를 켜고 Administration Server 의 `/config/server.xml` 파일의 특정한 설정을 편집하면 사용자는 다음 URL 을 통해 서버를 관리할 수 있습니다

server_name:*port*/`user-app`/*server_instance*/*virtual_server_ID*

예 :

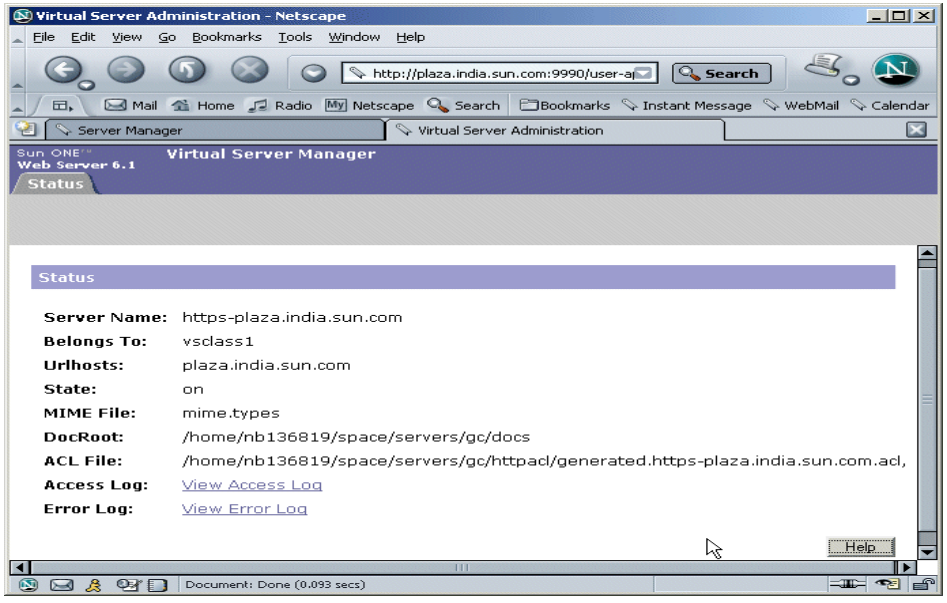
```
sun:9999/user-app/sun/vs2
```

서버 인스턴스는 서버 인스턴스 이름의 "`https`" 부분을 포함하지 않습니다 .

가상 서버 ID 를 결정하려면 서버 인스턴스의 `server.xml` 파일을 확인합니다 .

다음 그림은 최종 사용자가 보는 사용자 인터페이스입니다 .

가상 서버 관리 사용자 인터페이스



Sun ONE Web Server 6.1 을 설치한 후 *server_root*/https-admserv/config/server.xml 파일이 다음을 만드는 특정한 주석이 달린 항목을 포함한다는 것을 확인할 수 있습니다 .

- useradmin 이라는 가상 서버용 기본 청취 소켓 .
- 가상 서버용 가상 서버 클래스 .

useradmin 을 설정하려면 이러한 항목에서 주석을 제거하기만 하면 됩니다 .

서버가 이 기능을 사용하도록 구성하려면 다음 단계를 따르십시오 .

1. 관리 서버가 사용하는 포트와 별도의 포트를 실행하는 새로운 청취 소켓을 만듭니다 .

   예를 들어 , 관리 서버가 포트 8888 에서 실행되면 이 새로운 청취 소켓은 다른 포트 번호를 가져야 합니다 . 다른 청취 소켓을 사용하면 관리 서버를 보호할 수 있습니다 .

   보안상의 이유로 사용자 인터페이스를 통해 이 청취 소켓을 추가할 수 없습니다 . 대신 , 관리 서버의 server.xml 파일에서 추가할 수 있습니다 .

2. *server_root*/https-admserv/config/server.xml 에 있는 관리 서버의 server.xml 파일을 엽니다 .

3. LS, VSCLASS 및 VS 요소의 기본값이 들어있는 주석 달린 행에서 주석을 제거합니다 . 예 :

```
<!--
<LS id="ls2" port="9999" servername="plaza"
defaultvs="useradmin"/>
-->
<!--
<VSCLASS id="userclass" objectfile="userclass.obj.conf">
   <VS id="useradmin" connections="ls2" mime="mime1"
aclids="acl1" urlhosts="plaza">
      <PROPERTY name="docroot" value="/export1/wsinst/docs"/>
      <USERDB id="default"/>
      <WEBAPP uri="/user-app"
path="/export1/wsinst/bin/https/webapps/user-app"/>
   </VS>
</VSCLASS>
-->
```

   그러면 보안상의 이유로 별도의 포트에서 만들어진 useradmin 이 사용함으로 설정됩니다 .

4. 변경 사항을 `server.xml` 에 저장합니다 .
5. Administration Server 를 재시작하여 변경 사항을 적용합니다 .
6. 모든 서버 인스턴스의 모든 가상 서버의 대하여 이제 다음 URL을 사용하여 관리 URL 에 액세스할 수 있어야 합니다 .

   server_name:*port*/user-app/*server_instance/virtual_server_ID*

   예 :

   ```
   plaza:9999/user-app/plaza/https-plaza
   ```

액세스 제어

권한 없는 사용자로부터 가상 서버를 보호하려면 ACL 을 설정할 수 있습니다 . 각 가상 서버의 URL 은 고유하기 때문에 올바른 관리자만이 가상 서버의 설정에 액세스할 수 있도록 액세스를 설정할 수 있습니다 .

더 자세한 내용은 제 9 장 , " 서버 액세스 제어 " 를 참조하십시오 .

로그 파일

각 가상 서버에는 자체 로그 파일이 있을 수 있습니다 . 기본적으로 모든 가상 서버는 서버 인스턴스의 로그 파일을 공유합니다 . 사용자가 로그 파일을 볼 수 있도록 할 경우 대부분의 경우 각 가상 서버가 자체 액세스 및 오류 로그를 갖도록 로그 파일 설정을 변경해야 합니다 .

더 자세한 내용은 " 가상 서버 로그 설정 구성 " 페이지 333 을 참조하십시오 .

가상 서버 구현

Sun ONE Web Server 의 가상 서버 아키텍처는 매우 유연합니다 . 서버 인스턴스에는 보안 및 비보안을 모두 포함하는 많은 청취 소켓이 있을 수 있습니다 . IP 주소 기반과 URL 호스트 기반 가상 서버를 모두 가질 수 있습니다 .

또한 , 유사한 설정을 가진 가상 서버를 여러 개의 가상 서버 클래스로 그룹화할 수 있습니다 . 가상 서버 클래스의 모든 가상 서버는 `obj.conf` 의 동일한 요청 처리 명령을 공유합니다 .

모든 가상 서버는 자체 ACL 목록 , 자체 `mime.types` 파일 , 자체 Java Web Applications 세트를 가질 수 있습니다 (그러나 반드시 그렇지는 않습니다).

이 디자인은 다양한 응용 프로그램에 대해 서버를 구성하는 최대한의 유연성을 제공합니다 . 다음 예는 Sun ONE Web Server 에 대해 사용 가능한 구성의 일부에 대해 설명합니다 .

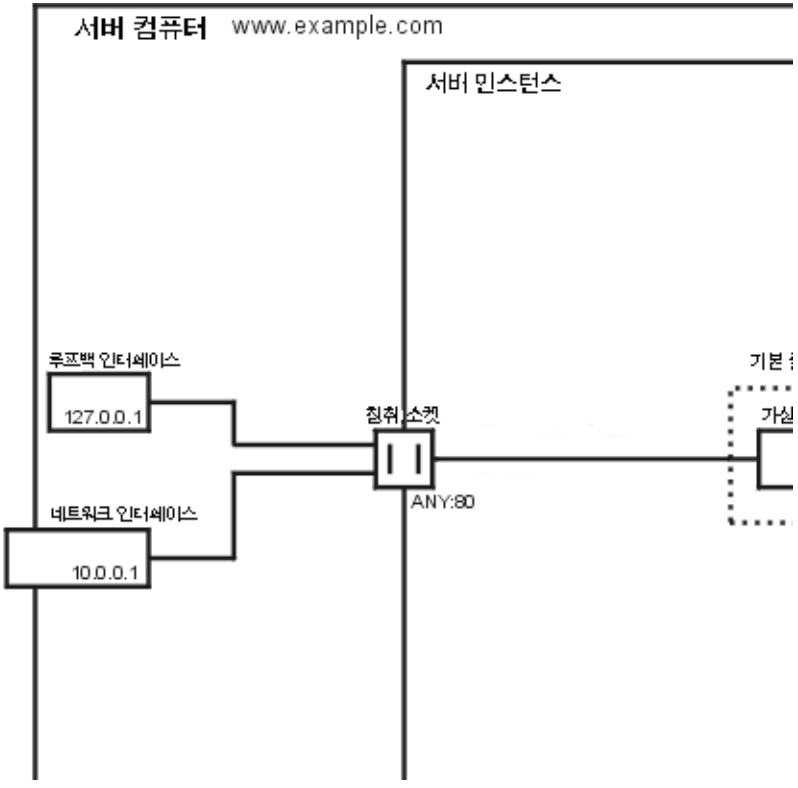
예 1: 기본 구성

Sun ONE Web Server 를 새로 설치한 후에는 하나의 서버 인스턴스를 갖게 됩니다 . 이 서버 인스턴스는 컴퓨터가 구성된 IP 주소의 포트 80(또는 설치시 선택한 것) 에서 청취하는 단 하나의 청취 소켓을 갖습니다 .

로컬 네트워크의 일부 메커니즘은 컴퓨터가 구성된 각 주소에 대하여 이름 대 주소 매핑을 설정합니다 . 다음 예에서 , 컴퓨터에는 두 개의 네트워크 인터페이스 , 즉 주소 127.0.0.1 의 루프백 인터페이스 (네트워크 카드가 없이도 존재하는 인터페이스) 와 주소 10.0.0.1 의 이더넷 인터페이스가 있습니다 .

이름 example.com 은 DNS 를 통해 10.0.0.1 에 매핑됩니다 . 청취 소켓은 시스템이 구성되는 주소의 포트 80 에서 청취하도록 구성됩니다 ("ANY:80" 또는 "0.0.0.0:80").

기본 구성



이 구성에서 다음에 대한 연결이 서버에 도달하고 가상 서버 VS1 에 의해 서비스됩니다 .

- `http://127.0.0.1/` (initiated on `example.com`)
- `http://localhost/` (initiated on `example.com`)
- `http://example.com/`
- `http://10.0.0.1/`

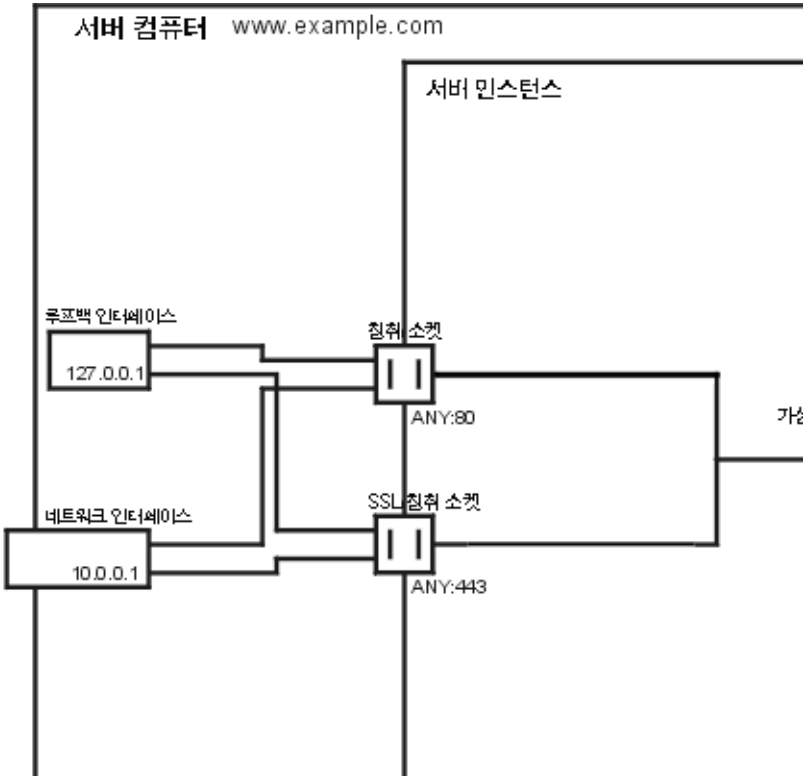
전통적인 웹 서버 사용을 위해서는 이 구성을 사용합니다 . 추가 가상 서버나 청취 소켓을 추가할 필요가 없습니다 . 설정을 defaultclass(VS1 은 defaultclass 의 구성원임) 및 VS1 자체로 변경하여 서버의 설정을 구성합니다 .

예제 2 보안 서버

기본 구성에서 SSL 을 사용하려면 청취 소켓을 보안 모드로 변경하기만 하면 됩니다 . 이것은 이전 Sun ONE Web Server. 버전에서 보안을 설정하는 방법과 유사합니다 .

ANY:443 으로 구성된 새 보안 청취 소켓을 추가하고 VS1 을 새 청취 소켓으로 연결할 수도 있습니다 . 이제 가상 서버는 SSL 을 사용하는 청취 소켓과 사용하지 않는 청취 소켓을 갖게 됩니다 . 이제 서버는 SSL 을 포함하거나 포함하지 않은 채 동일한 내용을 서비스합니다 . 즉 , `http://example.com/` 및 `https://example.com/` 이 동일한 내용을 제공하게 됩니다 .

보안 서버



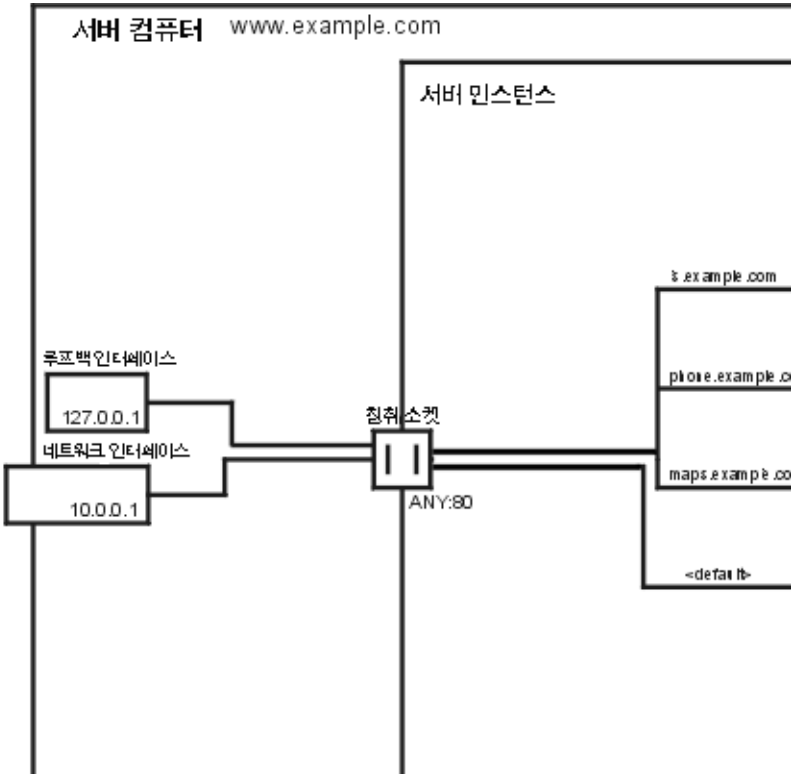
참고로 SSL 매개변수는 청취 소켓에 연결되어 있습니다 . 따라서 특정한 청취 소켓으로 구성된 모든 가상 서버에 대해 한 세트의 SSL 매개변수만 있을수 있습니다 .

예제 3: 인트라넷 호스트

보다 복잡한 Sun ONE Web Server 구성은 서버가 인트라넷 구현을 위해 몇 개 가상 서버를 호스트하는 구성입니다 . 예를 들어 , 3 개의 내부 사이트가 있고 해당 사이트에서 직원들은 사용자의 전화 번호를 보고 캠퍼스 지도를 보고 Information Services 부서에 대한 요청 상태를 추적할 수 있습니다 . 이전에 (이 예에서) 이들 사이트는 이름 `phone.example.com`, `maps.example.com` 및 `is.example.com` 이 매핑된 3 개의 다른 컴퓨터에서 호스트되었습니다 .

하드웨어 및 관리 오버헤드를 최소화하기 위해 컴퓨터 `example.com` 에 있는 한 개 웹 서버로 3 개 사이트 모두를 통합하려 합니다 . 이 작업을 두 가지 방법으로 할 수 있습니다 . 즉 , URL 호스트 기반 가상 서버를 사용하거나 별도의 청취 소켓을 사용하는 것입니다 . 두 가지 방법 모두 뚜렷한 장단점이 있습니다 .

URL 호스트 기반 가상 서버를 사용하는 인트라넷 호스팅

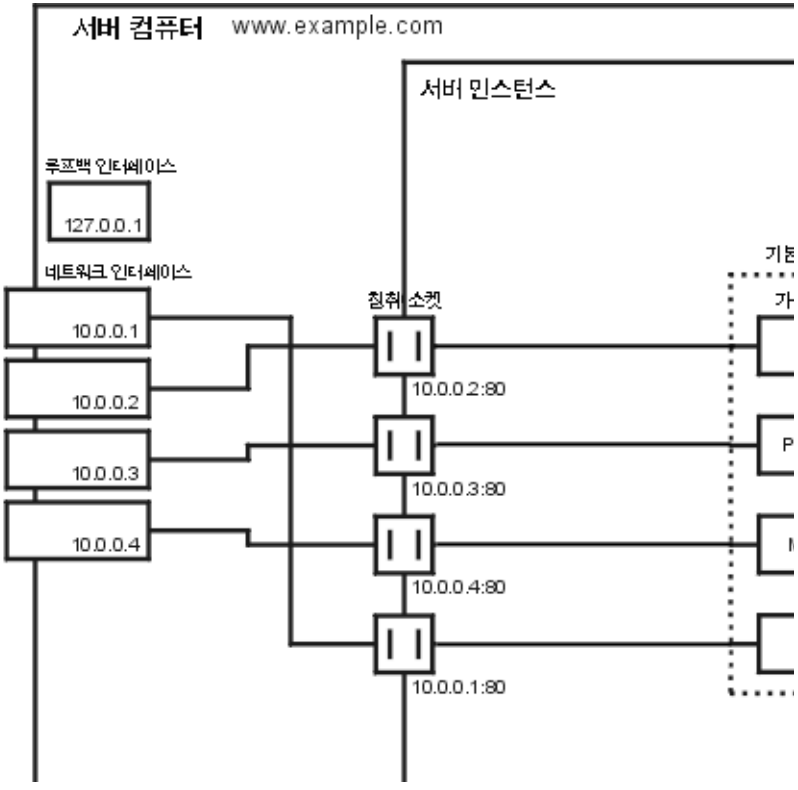


URL 호스트 기반 가상 서버가 설정하기 용이하지만 다음과 같은 단점이 있습니다 .

- 이 구성의 SSL 지원을 위해서는 와일드카드 인증서를 사용하는 비표준 설정이 필요합니다 . 더 자세한 내용은 제 4 장 , " 웹 컨테이너 및 웹 응용 프로그램용 J2EE 기반 보안 " 을 참조하십시오 .
- URL 호스트 기반 가상 서버는 기존 HTTP 클라이언트와 함께 작동하지 않습니다 .

주소당 한 개 청취 소켓을 가지는 IP 주소 기반 구성도 설정할 수 있습니다 .

별도의 청취 소켓을 사용하는 인트라넷 호스팅



IP 주소 기반 가상 서버의 장점은 다음과 같습니다 .

- HTTP/1.1 Host 헤더를 지원하지 않는 이전 클라이언트와 함께 작동합니다 .
- SSL 지원 제공이 간단합니다 .

다음과 같은 단점이 있습니다 .

- 호스트 컴퓨터의 구성 변경이 필요합니다 (실제 또는 가상 네트워크 인터페이스의 구성).

- 수많은 가상 서버 구성으로 확장되지 않습니다 .

두 가지 구성을 위해서는 3 개 이름에 대한 이름 대 주소 매핑 설정이 필요합니다 . IP 주소 기반 구성에서는 각 이름이 다른 주소에 매핑됩니다 . 호스트 컴퓨터가 이러한 주소의 연결을 수신하도록 설정되어야 합니다 . URL 호스트 기반 구성에서 모든 이름은 컴퓨터가 원래 가졌던 동일한 주소에 매핑될 수 있습니다 .

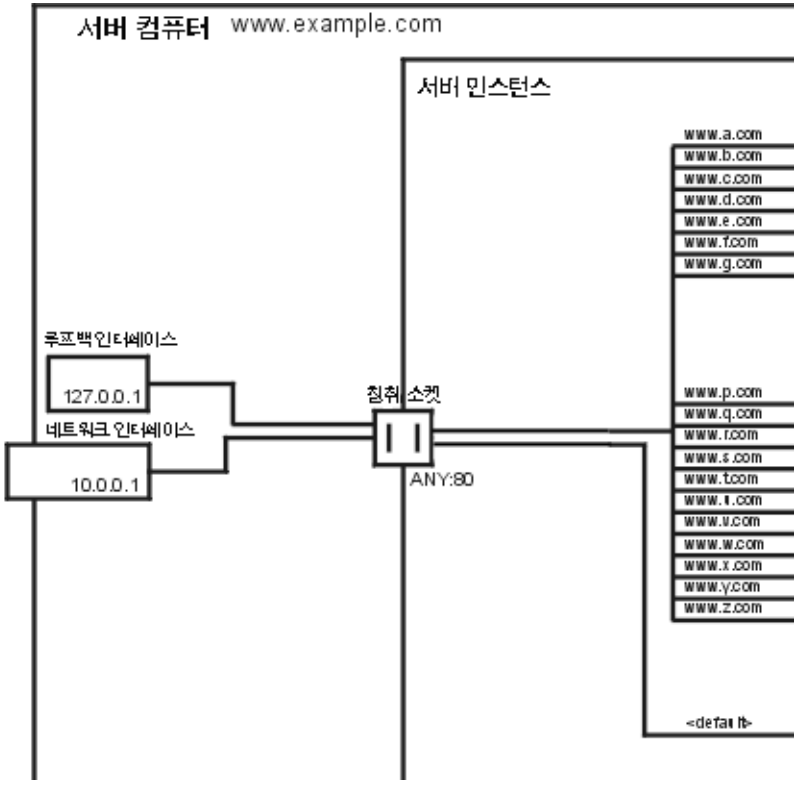
여러 청취 소켓으로 구성하면 서버가 요청이 들어오는 주소를 발견할 필요가 없으므로 최소한의 성능 이득을 제공합니다 . 그러나 여러 청취 소켓을 사용하면 추가 승인자 스레드 때문에 추가 오버헤드 (메모리 및 스케줄링) 가 생기기도 합니다 .

예제 4: 대량 호스팅

대량 호스팅은 트래픽 수준이 낮은 많은 가상 서버를 사용하도록 설정하는 구성입니다 . 예를 들어 , 트래픽 수준이 낮은 많은 개인용 홈 페이지를 호스트하는 ISP 가 이 범주에 해당합니다 .

가상 서버는 제공 서비스 수준에 따라 대개 URL 호스트 기반이고 여러 가상 서버 클래스 중 하나입니다 . 예를 들어 , 정적 내용만 허용하는 클래스와 정적 내용에 CGI 도 허용하는 클래스를 가질 수 있습니다 .

대량 호스팅



서버를 설치할 때 설치된 가상 서버 VS1 이 `defaultclass` 에 여전히 존재함을 유의하십시오 .

14 장

가상 서버 만들기 및 구성

가상 서버의 클래스에는 가상 서버 (클래스의 구성원) 가 연결되어 있습니다 . 가상 서버 수준에서 클래스 수준 설정의 일부를 무시할 수 있습니다 . 이 장에서는 각각의 가상 서버를 만들고 구성하는 방법에 대하여 설명합니다 . 가상 서버 구성에 대한 더 자세한 내용은 내용 관리를 참조하십시오 . 가상 서버의 개요는 가상 서버 사용을 참조하십시오 .

이 장의 내용 :

가상 서버 만들기

가상 서버를 사용하면 서버를 하나만 설치한 경우에도 여러 회사나 개인에게 도메인 이름 , IP 주소 및 일련의 서버 관리 기능을 제공할 수 있습니다 . 가상 서버에 대한 소개와 Sun ONE Web Server 에서 가상 서버를 설정하는 방법은 가상 서버 사용을 참조하십시오 .

가상 서버를 만들려면 다음 단계를 따르십시오 .

1. Class Manager 에서 Virtual Servers 탭을 누릅니다 .
2. Add Virtual Server 를 누릅니다 .
3. 가상 서버의 고유 이름을 선택합니다 .

4. 가상 서버의 URL 호스트를 선택합니다 .

   공백으로 구분된 URL 호스트를 하나 이상 입력할 수 있습니다 .

5. OK 를 누릅니다 .

이러한 설정이 가상 서버를 만드는 데 필요한 전부입니다 . 그러나 이 탭의 다른 페이지를 사용하여 추가 가상 서버 설정을 구성할 수 있습니다 .

가상 서버 설정 편집

가상 서버를 설정한 다음에는 편집할 수 있습니다 . 변경은 Class Manager 또는 Virtual Server Manager 의 두 가지 방법으로 수행할 수 있습니다 .

Class Manager 에서 페이지는 변경하려는 설정의 유형별로 구성됩니다 . 예를 들어 , 클래스의 하나 이상의 가상 서버의 Quality of Service 설정을 변경하려면 Quality of Service 페이지로 이동합니다 .

Virtual Server Manager 에서는 페이지가 하나의 가상 서버와만 관련이 되므로 해당 가상 서버의 설정을 모두 보고 변경할 수 있습니다 .

Class Manager 를 사용하는 편집

다음 Class Manager 페이지를 사용하여 가상 서버 설정을 편집합니다 .

가상 서버 설정 편집

가상 서버의 일반 설정을 편집하려면 Edit Virtual Servers 페이지를 사용합니다 . 이 페이지에 액세스하려면 다음 단계를 따르십시오 .

1. Class Manager 에서 Virtual Servers 탭을 누릅니다 .
2. Edit Virtual Servers 를 누릅니다 .
3. 가상 서버를 편집하려면 편집하려는 가상 서버 옆에 있는 From 드롭다운 목록을 누르고 Edit 또는 Delete 를 선택합니다 .

   기본 가상 서버는 편집할 수 있을 뿐 삭제할 수 없습니다 .

4. State 를 On, Off, Disabled 로 설정합니다 .

   상태를 Disabled 로 설정하면 서버를 다시 켤 수 있지만 서버의 최종 사용자는 그렇게 할 수 없습니다 .

   상태는 가상 서버의 상태로 서버 인스턴스의 On/Off 상태와는 상관 없습니다 . 이 페이지에서 가상 서버의 상태가 On 으로 표시되는 경우 해당 가상 서버는 서버 인스턴스 또한 On 인 경우에만 요청을 받아들일 수 있습니다 .

   또한 기본 서버 인스턴스용 기본 가상 서버의 경우에도 마찬가지입니다 . 서버 인스턴스를 Off 로 전환하면 가상 서버는 여전히 On 으로 설정될 수 있으나 연결을 수락할 수는 없습니다 .

   서버 인스턴스용 기본 가상 서버는 Off 또는 사용 안 함으로 설정할 수 없습니다 .

5. Urlhosts 열 아래 표시된 것과 다른 경우 사용하려는 URL Hosts 를 입력합니다 .

   공백으로 구분된 URL 호스트를 하나 이상 입력할 수 있습니다 .

6. 가상 서버 편집을 완료하면 OK 를 누릅니다 .

MIME 설정 구성

개별 가상 서버에 대해 MIME 유형 파일을 설정할 수 있습니다 . MIME 유형 파일은 파일 유형에 대한 파일 확장자의 매핑을 포함합니다 . 예를 들어 , MIME 유형 파일에서 `.cgi` 로 끝나는 모든 파일을 CGI 파일로 지정할 수 있습니다 .

각 가상 서버 또는 가상 서버 클래스용으로 별도의 MIME 유형을 만들 필요는 없습니다 . 대신 , 필요한 만큼의 MIME 유형 파일을 만들고 이를 가상 서버에 연결합니다 . 서버에는 하나의 MIME 유형 파일인 `mime.types` 가 기본으로 존재합니다 . 새로운 MIME 유형 파일을 만들거나 MIME 유형 파일의 정의를 편집하려면 "MIME 유형 선택 " 페이지 171 를 참조하십시오 .

특정 가상 서버의 MIME 유형 파일을 설정하려면 다음 단계를 따르십시오 .

1. Class Manager 에서 Virtual Servers 탭을 누릅니다 .
2. MIME Settings 를 누릅니다 .
3. 가상 서버 옆에 있는 드롭다운 목록에서 MIME 유형 파일을 선택합니다 .
4. OK 를 누릅니다 .

가상 서버 ACL 설정 구성

ACL 을 사용하여 가상 서버에 대한 액세스를 제어할 수 있습니다 . 각 가상 서버에는 LDAP 데이터베이스에 다른 기반 DN 이 있습니다 . 따라서 각 가상 서버는 Sun ONE Web Server 가 사용하는 단일한 LDAP 데이터베이스에 자체 항목을 가질 수 있습니다 .

더 자세한 내용은 " 가상 서버용 액세스 제어 " 페이지 221 을 참조하십시오 .

가상 서버 보안 구성

가상 서버가 보안 청취 소켓에 바인드된 경우 가상 서버에 대해 보안을 설정할 수 있습니다 .

보안에 대한 더 자세한 내용은 제 4 장 , " 웹 컨테이너 및 웹 응용 프로그램용 J2EE 기반 보안 " 을 참조하십시오 .

가상 서버 서비스 품질 구성

서비스 품질은 가상 서버에 대해 설정하는 성능 한계를 말합니다 . 예를 들어 , ISP 는 허용된 대역폭 양에 따라 가상 서버에 다른 금액을 부과할 수 있습니다 .

Status 탭의 Server Manager 에서 전체 서버 또는 가상 서버의 클래스에 대해 이러한 설정을 사용하도록 할 수 있습니다 . 그러나 개별 가상 서버용으로 서버 또는 클래스 수준 설정을 무시할 수 있습니다 .

가상 서버에 대해 서비스 품질을 사용하도록 하기 전에 먼저 전체 서버에 대해 사용을 설정하고 일부 기본값도 설정해야 합니다 . " 서비스 품질 사용 " 페이지 249 를 참조하십시오 .

가상 서버에 대해 서비스 품질을 구성하려면 다음 단계를 따르십시오 .

1. Class Manager 에서 Virtual Servers 탭을 누릅니다 .
2. Quality of Service 를 누릅니다 .

   클래스의 모든 가상 서버와 해당하는 서비스 품질 설정을 나열하는 페이지가 나타납니다 .

3. 가상 서버의 서비스 품질을 사용하도록 하려면 드롭다운 목록에서 Enable 을 선택합니다 .

   기본적으로 서비스 품질은 사용하지 않습니다 . 서비스 품질을 사용하면 서버의 오버헤드가 약간 증가합니다 .

4. 가상 서버의 최대 대역폭 한도를 바이트 / 초로 지정합니다 .

5. 최대 대역폭 설정을 집행할 것인지 선택합니다 .

   최대 대역폭을 집행하도록 선택하면 , 서버의 대역폭 한계를 초과하는 경우 추가의 연결은 거부됩니다 .

   최대 대역폭을 집행하지 않으면 최대값을 초과하는 경우 서버가 오류 로그에 메시지를 기록합니다 .

6. 가상 서버에 대해 허용된 최대 연결 수를 선택합니다 .

   이는 동시에 처리되는 요청의 수 입니다 .

7. 최대 연결 수 설정을 집행할 것인지 선택합니다 .

   최대 연결 수를 집행하도록 선택하면 , 서버의 한계를 초과하는 경우 추가의 연결은 거부됩니다 .

   최대 연결 수를 집행하지 않으면 최대값을 초과하는 경우 서버가 오류 로그에 메시지를 기록합니다 .

8. OK 를 누릅니다 .

서비스 품질 기능의 제한에 대한 더 자세한 내용은 " 서비스 품질 사용 " 페이지 249 를 참조하십시오 .

가상 서버 로그 설정 구성

가상 서버 액세스 및 오류 로그의 위치를 기본값에서 변경하려면 다음 단계를 따르십시오 .

1. Class Manager 에서 Virtual Servers 탭을 누릅니다 .

2. Logging Settings 를 누릅니다 .

   클래스의 모든 가상 서버와 해당 오류 로그의 위치를 나열하는 페이지가 나타납니다 .

3. 오류 및 액세스 로그의 절대 경로를 입력합니다 . 경로가 이미 존재해야 합니다 .

   기본적으로 모든 가상 서버의 액세스 및 오류 메시지는 서버 인스턴스의 액세스 및 오류 로그에 기록됩니다 . 가상 서버가 별도의 로그 파일을 갖도록 하려면 여기서 설정합니다 .

4. 경로를 기본값으로 변경하려면 Default 를 누릅니다 .
5. OK 를 누릅니다 .

특정 가상 서버에 대한 로그를 보려면 다음 단계를 따르십시오 .

1. Virtual Server Manager 에서 Logs 탭을 선택합니다 .
2. View Access Log or View Error Log 를 누릅니다 .
3. 표시할 항목의 수와 표시 기준을 선택합니다 .

   예를 들어 , 로그에 모든 가상 서버의 항목이 들어 있으면 특정 가상 서버의 항목만 표시하도록 선택할 수 있습니다 .

4. OK 를 누릅니다 .

가상 서버에 대한 로깅 사용

가상 서버 로깅을 사용하도록 설정하려면 다음 단계를 따르십시오 .

1. 서버 인스턴스에 대한 Server Manager 의 Logs 탭으로 이동하여 Log Preferences 를 선택합니다 .
2. Log File 필드에 경로와 파일 이름을 입력하여 새로운 액세스 로그를 만듭니다 .

   다음과 같이 변경하여 `magnus.conf` 에 새로운 액세스 로그를 직접 만들 수도 있습니다 .

   `Init fn=init access="$accesslog"` 에서 `Init fn=init access="newaccesslog"` 로

3. Format 아래에서 Only Log 를 선택하고 Virtual Server Id 를 확인합니다 .

   사용자 정의 형식의 경우 Custom Format 을 선택하고 줄 끝에 `%vsid%` 를 추가합니다 .

   `%vsid%` 는 여러 가상 서버를 사용할 경우 유용합니다 . 이 항목은 액세스 로그에 vsid 를 기록합니다 .

   또한 `magnus.conf` 파일의 `Init fn` 끝에 `%vsid%` 를 직접 추가할 수도 있습니다 .

4. OK 를 누릅니다 .
5. Apply 를 누릅니다 .
6. Apply Changes 를 눌러 변경 사항을 적용합니다 .

가상 서버 Java 웹 응용 프로그램 설정 구성

웹 응용 프로그램은 Java 서블릿 , JSP, HTML 페이지 , 클래스 및 기타 리소스의 컬렉션입니다 . 모든 리소스는 디렉토리에 저장되며 해당 디렉토리에 대한 모든 요청이 응용 프로그램을 실행합니다 . Virtual Server Manager 의 Web Applications 탭 아래 페이지를 사용하여 특정 가상 서버의 웹 응용 프로그램을 편집합니다 .

웹 응용 프로그램 `sun-web.xml` 과 웹 응용 프로그램용 배치 기술 파일에 대한 더 자세한 내용은 , Sun ONE Web Server 6.1 *Administrator? Configuration File Reference* 를 참조하십시오 .

Virtual Server Manager 를 통한 편집

Virtual Server Manager 에는 Preferences, Logs, Web Applications, WebDAV 등 4 개 탭이 포함됩니다 .

Preferences 탭은 다음에 대한 페이지를 포함합니다 .

- 상태
- 설정

Status 페이지는 일부 설정을 나열하고 가상 서버의 액세스 및 오류 로그에 대한 링크를 제공합니다 .

Settings 페이지에는 다음의 가상 서버용 설정이 있습니다 .

- 상태 (On 또는 Off)
- 문서 루트
- 액세스 및 오류 로그 디렉토리
- 디렉토리 서비스
- ACL 파일
- MIME 유형 파일

- CGI 설정

단일한 가상 서버를 편집할 경우에는 Virtual Server Manager 를 사용하여 한 페이지에서 모든 설정을 변경하는 것이 편리합니다 .

Logs 탭의 페이지는 단 하나이며 여기에서 선택한 가상 서버용 보고서를 만들 수 있습니다 .

웹 응용 프로그램 파일 배치 및 편집에 대한 더 자세한 내용은 제 15 장 , " 프로그램으로 서버 확장 " 을 참조하십시오 .

WebDAV 탭에서 가상 서버의 WebDAV 컬렉션을 만들고 편집할 수 있습니다 . WebDAV 컬렉션은 WebDAV 작업에 대해 사용 설정된 리소스 또는 리소스 집합입니다 . WebDAV 를 사용하여 웹에서 공동으로 문서를 작성할 수 있습니다 . WebDAV 를 통해 WebDAV 사용 리소스에 대하여 세밀한 수준으로 잠금을 적용하여 웹에서 공동 내용 작성 동안 겹쳐쓰기 충돌을 효과적으로 방지할 수 있습니다 .

WebDAV 탭에 포함된 페이지는 다음과 같습니다 .

- Add Collection 페이지
- Edit DAV Collection 페이지
- Lock Management 페이지

Add Collection 페이지를 사용하여 WebDAV 컬렉션을 만들 수 있습니다 .

Edit DAV Collection 페이지를 사용하여 WebDAV 사용 컬렉션을 구성할 수 있습니다 .

Lock Management 페이지를 사용하여 두드러진 잠금과 서버의 WebDAV 가능 리소스에 관한 기타 잠금 관련 정보을 볼 수 있습니다 .

더 자세한 내용은 제 19 장 , "WebDAV 를 사용하는 웹 게시 " 를 참조하십시오 .

가상 서버에 대한 보고서 만들기

이제 Virtual Server Manager 를 사용하여 단일 가상 서버에 대한 보고서를 만들 수 있습니다 . 보고서를 만들려면 아래의 설명과 같이 우선 가상 서버가 사용할 새 액세스 로그를 만들어야 하며 , 새 액세스 로그를 가상 서버 설정에 추가해야 합니다 .

가상 서버에 대한 보고서를 생성하려면 다음 단계를 따르십시오 .

1. 서버 인스턴스에 대한 Server Manager 의 Logs 탭으로 이동하여 Log Preferences 를 선택합니다 .

2. Log File 필드에 경로와 파일 이름을 입력하여 새로운 액세스 로그를 만듭니다 .

   다음과 같이 변경하여 magnus.conf 의 새로운 액세스 로그를 직접 만들 수도 있습니다 .

   `Init fn=init access="$accesslog"` 에서 `Init fn=init access="newaccesslog"` 로

3. Format 아래에서 Only Log 를 선택하고 Virtual Server Id 를 확인합니다 .

   사용자 정의 형식의 경우 Custom Format 을 선택하고 줄 끝에 `%vsid%` 를 추가합니다 .

   `%vsid%` 는 여러 가상 서버를 사용할 경우 유용합니다 . 이 항목은 액세스 로그에 vsid 를 기록합니다 .

   또한 `magnus.conf` 파일의 `Init fn` 끝에 `%vsid%` 를 직접 추가할 수도 있습니다 .

4. OK 를 누릅니다 .
5. Apply 를 누릅니다 .
6. Apply Changes 를 눌러 변경 사항을 적용합니다 .
7. 보고서를 생성하려는 가상 서버를 선택하고 Virtual Server Manager > Manage Classes 로 이동하고 > 트리 보기에서 Virtual server 를 선택합니다 .
8. Preferences 탭으로 이동하고 Settings 를 선택합니다 .

   Access Log 필드에서 액세스 로그를 새로 생성한 로그로 변경합니다 .

9. OK 를 누릅니다 .
10. Apply 를 누릅니다 .
11. Apply Changes 를 눌러 변경 사항을 적용합니다 .
12. Logs 탭을 선택합니다 .

    Generate Reports 페이지가 나타납니다 .

    이 페이지는 가상 서버를 만들고 `LogVSid` 가 On 이 아닌 경우에는 나타나지 않습니다 . 가상 서버에 대한 자세한 내용은 가상 서버에 대한 로깅 사용을 참조하십시오 .

13. (선택) 원할 경우 설정을 변경합니다 .
14. 보고서를 생성하려면 OK 를 누릅니다 .

가상 서버에 대한 디렉토리 서비스 선택

특정 가상 서버에 대해 특정한 디렉토리 서비스를 지정할 수 있습니다 . 그렇게 할 경우 선택한 디렉토리 서비스는 `server.xml` 파일의 해당하는 `VS`(가상 서버) 의 `USERDB` 요소 아래 로깅되빈다 . 디렉토리 서비스에 연결된 권한과 허가는 이후 서버가 액세스 제어 규칙을 평가 및 집행하는 데 사용됩니다 .

디렉토리 서비스를 가상 서버에 추가하려면 다음 단계를 수행합니다 .

1. Virtual Server Manager 에서 Settings 탭을 선택합니다 .

   가상 서버 설정의 목록이 표시됩니다 .

2. Directory Services 옆에 있는 Edit 링크를 선택합니다 .

   Pick Directory Services for Virtual Server 페이지가 새 창에서 시작됩니다 .

3. 디렉토리 서비스를 선택하고 OK 를 누릅니다 .
4. 변경 사항을 저장하고 적용합니다 .

참고 특정 가상 서버에 대해 선택한 디렉토리 서비스는 다른 가상 서버 사이에서 공유되지 않습니다 . 반면 액세스 제어 파일은 가상 서버 사이에 걸쳐 공유됩니다 .

가상 서버 삭제

가상 서버를 삭제하려면 다음 단계를 따르십시오 .

1. Class Manager 에서 Virtual Servers 탭을 누릅니다 .
2. Edit Virtual Servers 를 누릅니다 .
3. 원하는 서버 옆에 있는 드롭다운 목록에서 Delete 를 선택합니다 .

   서버를 설치할 때 만들어진 기본 가상 서버는 삭제할 수 없습니다 .

4. OK 를 누릅니다 .

   가상 서버가 삭제됩니다 .

15 장

프로그램으로 서버 확장

이 장에서는 클라이언트의 요청에 따라 HTML 페이지를 동적으로 생성하는 프로그램을 Sun ONE Web Server 에 설치하는 방법에 대하여 설명합니다 . 이러한 프로그램은 *서버측 응용 프로그램 (server-side application)* 이라고 합니다 . (*클라이언트측 응용 프로그램*은 클라이언트로 다운로드되어 클라이언트 컴퓨터에서 실행됩니다 .)

이 장의 내용 :

서버측 프로그램의 개요

Java 서블릿 및 CGI 프로그램은 강점과 용도가 다릅니다 . 다음 목록이 이러한 서버측 프로그램 사이의 차이점을 보여줍니다 .

- Java 서블릿은 네트워크 응용 프로그램을 작성하기 위한 완전 구비 프로그래밍 언어인 Java 로 작성됩니다 .
- CGI(Common Gateway Interface) 프로그램은 C, Perl 또는 기타 프로그래밍 언어로 작성할 수 있습니다 . 모든 CGI 프로그램은 클라이언트와 서버 사이에서 정보를 전달하는 표준적인 방법을 가지고 있습니다 .

서버에서 실행되는 서버측 응용 프로그램의 유형

Sun ONE Web Server 는 동적으로 내용을 생성하기 위해 다음과 같은 유형의 서버측 응용 프로그램을 실행할 수 있습니다 .

- Java 서블릿
- CGI 프로그램

Sun ONE Web Server 는 서버 자체의 행동을 확장하거나 수정하는 프로그램을 실행할 수도 있습니다 . 이러한 프로그램은 플러그인이라고 하며 Netscape Server Application Programming Interface(NSAPI) 로 작성됩니다 . 플러그인 프로그램의 작성과 설치에 대한 더 자세한 내용은 Sun ONE Web Server 6.1 *NSAPI Programmer's Guide* 를 참조하십시오 .

서버측 응용 프로그램이 서버에 설치되는 방법

각 유형의 프로그램은 서버에 다른 방법으로 설치됩니다 . 다음 목록이 해당 절차를 요약합니다 .

- Java 서블릿의 경우 웹 응용 프로그램을 작성 및 구현할 수 있습니다 . 더 자세한 내용은 " 서버가 서브릿을 실행하기 위해 필요한 사항 " 페이지 342 를 참조하십시오 .
- CGI 프로그램의 경우 특정한 파일 이름 확장자를 가진 모든 파일이나 CGI 프로그램으러 지정된 디렉토리의 모든 파일 또는 두 가지 모두를 인식하도록 서버를 구성합니다 . 더 자세한 내용은 "CGI 프로그램 설치 " 페이지 349, "Windows CGI 프로그램 설치 " 페이지 354, 및 "Windows 용 셸 CGI 프로그램 설치 " 페이지 356 을 참조하십시오 .

이러한 설치 절차는 다음 부분에서 설명합니다 .

Java 서브릿 및 JavaServer Pages(JSP)

이 부분은 Sun ONE Web Server 에서 Java 서브릿 및 JavaServer Pages 를 설치하는 방법에 대해 설명합니다 .

다음 주제에 대해 설명합니다 .

서브릿 및 JavaServer 페이지의 개요

Sun ONE Web Server 6.1 은 서브릿 및 JSP 가 웹 응용 프로그램에 포함되도록 하는 Servlet 2.3 API 표준을 지원합니다 .

서브릿 , JavaServer Page, HTML 문서 및 기타 웹 리소스의 집합으로 여기에는 이미지 파일 , 압축된 자료 및 기타 데이터가 포함될 수 있습니다 . 웹 응용 프로그램은 저장 파일로 패키지화 되거나 (WAR 파일) 또는 개방형 디렉토리 구조로 존재할 수 있습니다 .

참고 Servlet API 버전 2.3 은 버전 2.1 과 완전 역방향 호환되므로 모든 기존 서브릿은 수정이나 재컴파일 없이 계속 작업할 수 있습니다 .

서브릿을 개발하려면 Sun Microsystems 의 Java Servlet API 를 사용합니다 . Java Servlet API 사용에 대한 더 자세한 내용은 Sun Microsystems 에서 제공하는 설명서를 참조하십시오 .

`http://java.sun.com/products/servlet/index.jsp`

JSP 는 웹 브라우저에서 볼 수 있는 HTML 페이지와 아주 유사한 페이지입니다 . 그러나 HTML 태그 외에 JSP 태그 및 Java 코드와 혼합된 지시문 세트를 포함할 수 있어 페이지의 동적 내용을 통합하도록 웹 페이지 디자이너의 능력을 확장합니다 . 이러한 추가 기능은 등록정보 값 표시 및 단순한 조건문 사용 등과 같은 기능을 제공합니다 . Sun ONE Web Server 6.1 은 JavaServer Pages (JSP) 1.2 API 표준을 지원합니다 .

참고 응용 프로그램이 요청하는 URI 의 대소문자 (예를 들어 , `/foo.JSP`) 가 파일 시스템 경로의 대소문자와 일치하도록 하십시오 (예를 들어 , `C:\Program Files\WebServer\docs\foo.jsp`). Sun ONE Web Server 6.1 Java 웹 컨테이너가 현재 대소문자를 구분하는 일치를 수행하기 때문에 필요합니다 .

JSP 만들기에 대한 더 자세한 내용은 Sun Microsystem 의 JavaServer Pages 웹 사이트를 참조하십시오 .

```
http://java.sun.com/products/jsp/index.jsp
```

Sun ONE Web Server 와 함께 서브릿 및 JSP 를 개발하는 데 대한 더 자세한 내용은 Sun ONE Web Server 6.1 *Programmer's Guide to Web Applications 을 참조하십시오* .

서버가 서브릿을 실행하기 위해 필요한 사항

Sun ONE Web Server 는 Java Development Kit (JDK) 버전 1.4.1_03 을 포함합니다 . Web Server 이전 버전에서 , Java 는 서버측에서 구성되었으나 6.1 릴리스에서는 Web Server 의 인스턴스당 Java 를 구성할 수 있습니다 .

Sun ONE Web Server 6.1 과 함께 번들 제공되는 JDK 를 사용하거나 각자 선택한 JDK 를 사용할 수 있습니다 . 이 경우에는 JDK 에 대한 경로를 지정해야 합니다 . 이 작업에 대한 자세한 설명은 "JVM 설정 구성 " 페이지 277 을 참조하십시오 .

기본적으로 , Java 는 Sun ONE Web Server 를 설치할 때 사용하지 않도록 설정됩니다 . 서브릿을 사용하도록 설정하려면 먼저 Java 를 설정해야 합니다 .

Java 설정 방법에 대한 더 자세한 내용은 "Java 사용 설정 및 해제 " 페이지 275 를 참조하십시오 .

웹 응용 프로그램 구현

다음 부분은 `wdeploy` 명령줄 유틸리티를 사용하여 또는 사용자 인터페이스를 통해 수동으로 웹 응용 프로그램을 구현 , 편집 및 삭제하는 방법에 대해 설명합니다 .

server.xml 파일 사용

웹 응용 프로그램은 구현되면 기본적으로 사용하도록 설정됩니다 . 구현된 웹 응용 프로그램을 수동으로 사용하지 않도록 설정하려면 `server.xml` 파일을 다음과 같이 수동으로 수정해야 합니다 .

```
<VS>
<WEBAPP uri="/mywebapp" path="/webappdir" enabled = "false" >
</WEBAPP>

...

</VS>
```

잘못해서 동일한 설명을 가진 하나 이상의 웹 응용 프로그램을 구현 또는 편집하면 그 중 하나는 사용하지 않도록 설정되고 서버는 `enabled = “false”` 를 무시하며 `enabled = “true”` 의 기본 설명으로 계속합니다 .

`server.xml` 파일에 대한 자세한 내용은 Sun ONE Web Server 6.1 *Programmer's Guide to Web Applications* 를 참조하십시오 .

두 가지 방법으로 웹 응용 프로그램을 구현 및 편집할 수 있습니다 .

- Administration Server Interface 사용
- 명령줄 인터페이스 사용

Administration Server Interface 사용

Sun ONE Web Server 6.1 을 사용하면 특화된 가상 서버용 웹 응용 프로그램을 구현 , 편집 , 삭제 및 사용 여부 설정 등의 작업을 할 수 있습니다 .

웹 응용 프로그램 구현

Virtual Server Manager 의 Web Applications 탭 아래에서 Deploy Web Applications 페이지를 선택하여 Deploy Web Applications 페이지에 액세스할 수 있습니다 .

웹 응용 프로그램을 구현하려면 다음 단계를 따르십시오 .

1. WAR File On 드롭다운 목록에서 Local Machine 또는 Server Machine 을 선택합니다 .

   WAR 파일을 서버로 업로드하는 경우에는 Local Machine 을 선택합니다 . WAR 파일이 이미 서버 컴퓨터에 있는 경우에는 Server Machine 을 선택합니다 .

2. 제공된 필드의 웹 응용 프로그램이 포함된 `WAR` 파일의 로컬 또는 서버 컴퓨터 경로를 입력합니다 .

   서버 컴퓨터의 경우 WAR 파일의 절대 경로를 입력합니다 .

   로컬 컴퓨터에서 사용 가능 경로를 찾아볼 수 있습니다 . Browse 를 누르면 File Upload 창이 표시되며 , 여기에서 서버로 업로드할 WAR 파일을 선택할 수 있습니다 .

3. 제공된 필드에 가상 서버에 있는 웹 응용 프로그램용 URI 를 입력합니다 .

4. WAR 파일의 내용을 추출할 서버 컴퓨터의 디렉토리 절대 경로를 입력합니다 . 디렉토리가 없는 경우 새로 만들어집니다 .

5. OK 를 누릅니다 .

6. Apply 를 누릅니다 .

7. 웹 응용 프로그램을 구현할 Dynamic Reconfiguration 을 선택합니다 .

웹 응용 프로그램 편집

이미 구현된 웹 응용 프로그램을 편집 , 삭제 , 사용 안 함 또는 사용함으로 설정할 수 있습니다 . Virtual Server Manager 의 Web Applications 탭 아래에서 Edit Web Applications 를 선택하여 Edit Web Applications 페이지에 액세스합니다 .

이미 구현된 웹 응용 프로그램을 편집 , 삭제 , 사용 안 함 또는 사용함으로 설정하려면 다음 단계를 따르십시오 .

1. 편집할 웹 응용 프로그램 옆에 있는 Action 열의 드롭다운 목록에서 수행하려는 작업을 선택합니다 . 옵션 :
   - Edit 를 선택하여 웹 응용 프로그램을 액세스할 수 있는 URI 를 변경합니다 .
   - Delete 를 선택하여 웹 응용 프로그램 파일에서 웹 응용 프로그램 항목을 삭제하고 웹 응용 프로그램이 구현된 디렉토리를 삭제합니다 .
   - Disable 을 선택하면 URI 에서 웹 응용 프로그램에 액세스할 수 없으나 웹 응용 프로그램이 삭제되지는 않습니다 .
   - Enable 을 선택하면 이전에 사용 안 함으로 설정된 웹 응용 프로그램을 다시 사용할 수 있습니다 .

주의 웹 응용 프로그램을 삭제하면 응용 프로그램이 구현된 디렉토리도 삭제됩니다 .

2. (선택) 웹 응용 프로그램을 편집할 경우 URI 필드에 새 URI 를 입력합니다 .
3. OK 를 누릅니다 .
4. Apply 를 누릅니다 .
5. 웹 응용 프로그램을 구현할 Dynamic Reconfiguration 을 선택합니다 .

명령줄 인터페이스 사용

웹 응용 프로그램을 수동 구현하기 전에 *server_root*`/bin/https/httpsadmin/bin` 디렉토리가 경로에 있고 `IWS_SERVER_HOME` 환경 변수가 *server_root* 디렉토리에 대해 설정되도록 해야 합니다 .

가상 서버 응용 프로그램을 구현하려면 :

명령줄에서 `wdeploy` 유틸리티를 사용하여 WAR 파일을 가상 서버 웹 응용 프로그램 환경으로 구현할 수 있습니다

```
wdeploy deploy -u <uri_path> -i <instance> -v <vs_id> [ [-V
<verboseLevel>]| [-q] ] [-n] [-d <directory>] <war_file>
```

가상 서버 웹 응용 프로그램을 삭제하려면 :

```
wdeploy delete -u <uri_path> -i <instance> -v <vs_id> [ [-V
<verboseLevel>] | [-q] ] [-n] hard|soft
```

가상 서버에 대한 웹 응용 프로그램 URI 및 디렉토리를 나열하려면 :

```
wdeploy list -i <instance> -v <vs_id> [ [-V <verboseLevel>] | [ -q] ]
```

명령줄 매개변수는 다음과 같은 의미가 있습니다 .

`uri_path`	웹 응용 프로그램에 대한 URI 접두사 .
`instance`	서버 인스턴스 이름 .
`vs_id`	가상 서버 ID.
`directory`	(선택) 응용 프로그램이 구현되거나 응용 프로램이 삭제되는 디렉토리 . 디렉토리가 구현에 대해 지정되지 않으면 응용 프로그램은 문서 루트 디렉토리에 구현됩니다 .
`hard | soft`	디렉토리 및 `server.xml` 항목이 삭제되는지 (`hard`) 아니면 `server.xml` 항목만 삭제되는지 (`soft`) 지정합니다 .
`war_file`	WAR 파일 이름
verboseLevel	콘솔에 로그 메시지를 표시하는 자세한 수준 . 값은 0 ~ 4 범위입니다 . 기본값은 1 입니다 . Sun ONE web Server 6.1 에서는 `server.xml` 의 `LOG` 요소의 `loglevel` 속성이 이 요소 대신 사용된다는 점을 유의하십시오 .
`-q`	(안정) 자세한 수준을 0 으로 설정합니다 . 이것은 설정 `-V 0` 과 동등합니다 .
`-n`	`wdeploy` 가 웹 서버에 대한 재구성 명령을 자동 전송하지 못하게 합니다 . 더 자세한 내용은 wdeploy 명령에서 -n 사용을 참조하십시오 .

주의 웹 응용 프로그램을 구현하고 *디렉토리*를 지정하지 않으면 응용 프로그램이 문서 루트 디렉토리에 구현됩니다 . 그런 다음 `hard` 매개변수를 사용하여 응용 프로그램을 삭제하면 문서 루트 디렉토리가 삭제됩니다 .

`wdeploy deploy` 명령을 실행하면 세 가지 일이 발생합니다 .

- 주어진 *uri_path* 및 *directory* 를 가진 웹 응용 프로그램에 `server.xml` 파일이 추가됩니다 .
- WAR 파일은 대상 디렉토리에서 추출됩니다
- 서버는 동적으로 새 웹 응용 프로그램을 로드하도록 구성됩니다 .

예 :

```
wdeploy deploy -u /hello -i server.sun.com -v acme.com
-d /s1ws61/https-server.sun.com/acme.com/web-apps/hello
/s1ws61/plugins/servlets/examples/web-apps/HelloWorld/HelloWorld.wa
r
```

이 유틸리티의 결과는 다음 `server.xml` 항목과 같습니다 .

```
<VS>
   <WEBAPP uri="/hello"
   dir="/s1ws61/https-server.sun.com/acme.com/webapps/hello"/>
</VS>
```

`/s1ws61/https-server.sun.com/acme.com/web-apps/hello` 디렉토리는 다음 내용을 갖습니다 .

```
colors
index.jsp
META-INF
WEB-INF/
   web.xml
   /classes/
      HelloWorldServlet.class
      HelloWorldServlet.java
      SnoopServlet.class
      SnoopServlet.java
```

wdeploy 명령에서 -n 사용

Sun ONE Web Server 6.1 에서는 웹 응용 프로그램을 구현 또는 삭제한 후 `wdeploy` 가 서버를 동적으로 구성하여 구현 또는 서버가 삭제된 웹 응용 프로그램을 로드 또는 언로드하도록 합니다 . 이전에 다음 중 하나를 수행하여 변경 사항이 적용되도록 서버를 명시적으로 재구성해야 했습니다

- reconfig 스크립트 사용
- 서버 재시작
- Administration User Interface 에서 Apply 누르기

이제 wdeploy 명령 성공이 새 웹 응용 프로그램에 대한 요청을 서비스하도록 또는 삭제된 웹 응용 프로그램에 대한 요청 서비스를 중지하도록 자동 설정됩니다 .

-n 옵션은 wdeploy 가 웹 서버에 대한 재구성 명령을 자동 전송하지 못하게 합니다 . 다중 웹 응용 프로그램을 구현 또는 구현 취소할 경우 (예를 들어 , 스크립트에서) 명령의 -n 옵션을 사용합니다 . 그러면 마지막 웹 응용 프로그램이 구현된 후 한 번만 서버를 재구성합니다 .

구현된 웹 응용 프로그램 액세스

응용 프로그램을 구현한 후 다음과 같이 브라우저에서 액세스할 수 있습니다 .

http://*vs_urlhost*[:*vs_port*]/*uri_path*/[*index_page*]

URL 의 일부는 다음과 같은 의미를 갖습니다 .

vs_urlhost	가상 서버의 urlhosts 값 중 하나 .
vs_port	(선택) 가상 서버가 기본값이 아닌 포트를 사용할 경우에만 필요합니다 .
uri_path	응용 프로그램 구현에 사용한 것과 동일합니다 . 이것은 컨텍스트 경로이기도 합니다 .
index_page	(선택) 최종 사용자가 먼저 액세스하려하는 응용 프로그램의 페이지 .

예 :

```
http://acme.com:80/hello/index.jsp
```

또는

```
http://acme.com/hello/
```

값 반환

wdeploy 옵션은 다음 종료 값을 반환합니다 .

- 0. wdeploy 옵션이 성공적으로 실행되었음을 나타냅니다 .
- 1. 구성 파일의 잘못된 명령줄 인수 또는 잘못된 내용으로 인해 wdeploy 옵션 실행 도중 오류가 발생했음을 나타냅니다 .
- 2. 오류가 운영 체제 설정 때문임을 나타냅니다 . 지정된 디렉토리가 종료되지 않거나 파일 권한이 설정되지 않습니다 .

웹 응용 프로그램에서 서브릿 및 JSP 구현

응용 프로그램 외부에서 4.*x* 서브릿 및 JSP 를 구현할 수 있지만 기본 가상 서버에서만 가능합니다. 자세한 내용은 Sun ONE Web Server 6.1 *Programmer's Guide to Web Applications* 를 참조하십시오 .

JVM 설정 구성

Server Manager 의 Java 탭에서 Java Virtual Machine(JVM) 의 속성을 구성할 수 있습니다 .

이러한 옵션에 대한 자세한 내용은 Sun ONE Web Server 6.1 *Programmer's Guide to Web Applications* 를 참조하십시오 .

Deleting Version Files

Server Manager 의 Java 탭의 Delete Version Files 페이지를 사용하여 JavaServer Pages 클래스 캐시 및 세션 데이터 캐시에 대한 버전 번호를 포함하는 파일을 삭제할 수 있습니다 . 이 페이지에는 다음 필드가 있습니다 .

Clear Session Data

서버가 MmapSessionManager 세션 관리자를 사용할 경우 지속적인 세션 정보를 저장하는 SessionData 디렉토리를 삭제합니다 .

Delete JSP ClassCache Files

JavaServer Pages(JSP) 에 대한 정보를 캐시하는 ClassCache 디렉토리를 삭제합니다 . 이 디렉토리의 기본 위치는 다음과 같습니다 .

```
server_root/https-server_id/ClassCache/virtual_server_id/webapp_uri
/
```

서버가 JSP 페이지를 서비스할 경우 JSP 와 연관된 .java 및 .class 파일을 만들고 ClassCache 디렉토리 아래의 JSP 클래스 캐시에 저장합니다 .

서버는 JavaServer Pages(JSP) 및 서브릿에 대한 캐시 정보에 대해 두 개의 디렉토리를 사용합니다 .

- `ClassCache`

  서버는 다음 디렉토리를 사용하여 JavaServer Pages(JSP) 의 캐시 정보에 대해 다음 디렉토리를 사용합니다 .

server_root/https-*server_id*/ClassCache/*virtual_server_id*/*webapp_uri*/

서버가 JSP 페이지를 서비스할 경우 서버는 JSP 와 연관된 .java 및 .class 파일을 만들고 ClassCache 디렉토리 아래의 JSP 클래스 캐시에 저장합니다 .

- SessionData

  서버가 MMappedSessionManager 세션 관리자를 사용할 경우 SessionData 디렉토리에 지속적 세션 정보를 저장합니다 .

각 캐시에는 서버가 캐시의 디렉토리 및 파일 구조를 결정하는 데 사용하는 버전 번호가 담긴 version 파일이 있습니다 . 버전 파일만 삭제하면 캐시를 비울 수 있습니다 .

서버가 시작하고 버전 파일을 찾지 못하면 해당 캐시의 디렉토리 구조를 삭제하고 버전 파일을 다시 만듭니다 . 다음에 서버가 JSP 페이지를 서비스할 때 JSP 클래스 캐시를 다시 만듭니다 . 다음에 MMappedSessionManager 세션 관리자를 사용하는 동안 JSP 페이지 또는 서브릿을 서비스할 때 세션 데이터 캐시를 다시 만듭니다 .

서버의 장래 업그레이드가 캐시에 대해 다른 형식을 사용하면 서버는 버전 파일의 번호를 확인하고 버전 번호가 올바르지 않을 경우 캐시를 비웁니다 .

CGI 프로그램 설치

이 부분은 CGI 프로그램을 설치하는 방법에 대해 다룹니다 . 다음 주제에 대해 설명합니다 .

- CGI 의 개요
- CGI 디렉토리 지정
- CGI 를 파일 유형으로 지정
- 실행 파일 다운로드

또한 다음 부분은 Windows 특정 CGI 프로그램을 설치하는 방법에 대해 설명합니다 .

- Windows CGI 프로그램 설치
- Windows 용 셸 CGI 프로그램 설치

CGI 의 개요

Common Gateway Interface(CGI) 프로그램은 어떤 프로그래밍 언어로도 정의할 수 있습니다 . UNIX/Linux machine 시스템에서 Bourne 셸 또는 Perl 스크립트로 작성된 CGI 프로그램을 찾을 수 있습니다 .

참고 UNIX/Linux 하에서는 추가 `CGIStub` 프로세스가 실행되고 서버는 이것을 CGI 실행을 돕는 데 사용합니다 . 이러한 프로세스는 CGI 에 대한 최초 액세스 동안에만 만들어집니다 . 프로세스의 숫자는 서버의 CGI 로드에 따라 다릅니다 . 이러한 CGIStub 프로세스를 종료하지 마십시오 . 서버가 중지하면 사라집니다 .

Windows 컴퓨터에서 C++ 또는 배치 파일로 작성된 CGI 프로그램을 찾을 수 있습니다 . Windows 의 경우 Visual Basic 과 같은 Windows 기반 프로그래밍 언어로 작성된 CGI 프로그램은 다른 메커니즘을 사용하여 서버와 작동합니다 . 이들을 Windows CGI 프로그램이라고 합니다 . Windows CGI 에 대한 더 자세한 내용은 "Windows CGI 프로그램 설치 " 페이지 354 를 참조하십시오 .

참고 명령줄 유틸리티를 실행하려면 `Path` 변수가 *server_root*`/bin/https/bin` 을 포함하도록 수동으로 설정해야 합니다 .

프로그래밍 언어와 관계없이 모든 CGI 프로그램은 동일한 방식으로 데이터를 받아들이고 반환합니다 . CGI 프로그램 작성에 대한 더 자세한 내용은 다음 정보원을 참조하십시오 .

- Sun ONE Web Server 6.1 *Programmer's Guide*
- *Common Gateway Interface*:

  `http://hoohoo.ncsa.uiuc.edu/cgi/overview.html`

- 온라인 설명서 웹 사이트에서 구할 수 있는 CGI 관련 아티클 :

  `http://docs.sun.com/`

서버 시스템에 CGI 프로그램을 저장하는 방법은 두 가지입니다 .

- CGI 프로그램만 포함하는 디렉토리를 지정합니다 . 모든 파일은 파일 확장자에 관계없이 프로그램을 실행됩니다 .

- CGI 프로그램이 모든 특정 파일 유형임을 지정합니다 . 즉 , 모두 파일 확장자 `.cgi`, `.exe` 또는 `.bat` 를 사용합니다 . 프로그램은 모든 디렉토리에 또는 문서 루트 디렉토리 아래 있을 수 있습니다 .

원할 경우 동시에 두 가지 옵션을 모두 사용함으로 설정할 수 있습니다 .

두 가지 구현에는 장점이 있습니다 . 특정한 사용자 집합만 CGI 프로그램을 추가하도록 하려면 지정된 디렉토리에 CGI 프로그램을 유지하고 해당 디렉토리에 대한 액세스를 제한합니다 . HTML 파일을 추가할 수 있는 모든 사람이 CGI 프로그램을 추가할 수 있도록 하려면 파일 유형 방법을 사용합니다 . 사용자는 자신의 HTML 파일과 동일한 디렉토리에 CGI 파일을 유지할 수 있습니다 .

디렉토리 옵션을 선택하면 서버는 해당 디렉토리의 모든 파일을 CGI 프로그램으로 해석합니다 . 동일한 토큰으로 파일 유형 옵션을 선택하면 서버는 파일 확장자 `.cgi`, `.exe` 또는 `.bat` 를 가진 모든 파일을 CGI 프로그램으로 처리합니다 . 파일이 CGI 프로그램이 아닌 확장자 중 하나를 가지면 사용자가 액세스를 시도할 때 오류가 발생합니다 .

참고 기본적으로 CGI 프로그램의 파일 확장자는 `.cgi`, `.exe` 및 `.bat` 입니다 그러나 MIME 유형 파일을 수정하여 CGI 프로그램을 나타내는 확장자를 변경할 수 있습니다 . Server Preferences 탭을 선택하고 MIME Types 링크를 눌러 이 작업을 수행할 수 있습니다 .

CGI 디렉토리 지정

가상 섭의 클래스에 대해 CGI 전용 디렉토리를 지정하려면 다음 단계를 따르십시오 .

1. Class Manager 에서 Programs 탭을 선택합니다 .

   CGI Directory 창이 나타납니다 .

2. URL Prefix 필드에서 이 디렉토리에 대해 사용할 URL 접두사를 입력합니다 . 즉, 입력한 텍스트가 URL 에 있는 CGI 프로그램용 디렉토리로 표시됩니다 .

   예를 들어 , URL 접두사로 `cgi-bin` 을 입력하는 경우 해당 CGI 프로그램에 대한 모든 URL 의 구조는 다음과 같습니다 .

   `http://`*yourserver.domain.com*`/cgi-bin/`*program-name*

참고 지정하는 URL 접두사는 CGI 디렉토리 필드에서 지정하는 실제 CGI 디렉토리와 다를 수 있습니다 .

3. CGI Directory 텍스트 필드에서 디렉토리 위치를 절대 경로로 입력합니다 . 이 디렉토리가 반드시 문서 루트 아래에 있어야 하는 것은 아닙니다 . 다음 단계에서 URL 접두사를 지정해야 하기 때문입니다 .
4. OK 를 누릅니다 .
5. 변경 사항을 저장 및 적용합니다 .

기존 CGI 디렉토리를 제거하려면 CGI Directory 양식에서 해당 디렉토리의 Remove 버튼을 누릅니다 . 기존 디렉토리의 URL 접두사 또는 CGI 디렉토리를 변경하려면 해당 디렉토리의 Edit 버튼을 누릅니다 .

CGI 프로그램을 지정한 디렉토리로 복사합니다 . 해당 디렉토리의 모든 파일이 CGI 파일로 처리되므로 HTML 파일은 CGI 디렉토리에 넣지 않도록 합니다 .

각 소프트웨어 가상 서버에 대해 고유한 CGI 속성 구성

단일한 가상 서버에 대해 CGI 속성을 지정하려면 다음 단계를 따르십시오 .

1. Class Manager 에서 Manager Virtual Servers 버튼을 누릅니다 .
2. Virtual Server Manager 에서 Settings 링크를 선택합니다 .
3. CGI User 텍스트 필드에 CGI 프로글매을 실행할 사용자의 이름을 입력합니다 .
4. CGI Grouptext 필드에 CGI 프로그램을 실행할 그룹 이름을 입력합니다 .
5. CGI Directory 텍스트 필드에서 `chroot` 뒤에 그러나 실행이 시작하기 전에 `chdir` 에 대한 디렉토리를 입력합니다 .
6. (UNIX 만 해당) CGI Nice 텍스트 필드에 서버에 대한 상대적인 CGI 프로그램의 우선 순위를 결정하는 증가분을 입력합니다 . 보통 서버는 값이 0 인 nice 로 실행되며 nice 의 증가분은 0(CGI 프로그램이 서버와 동일한 우선 순위로 실행) 에서 19(CGI 프로그램이 서버보다 매우 낮은 우선 순위로 실행) 사이입니다 . -1 의 증가분을 지정하여 CGI 프로그램의 운선 순위를 서버보다 높게 설정할 수 있으나 , 권장되지 않습니다 .
7. Chroot Directory 텍스트 필드에 실행이 시작하기 전에 chroot에 대한 디렉토리를 입력합니다 .
8. OK 를 누릅니다 .
9. 변경 사항을 저장 및 적용합니다 .

CGI 를 파일 유형으로 지정

CGI 프로그램을 파일 유형으로 지정하려면 다음 단계를 따르십시오 .

1. Class Manager 에서 Programs 탭을 선택합니다 .
2. CGI File Type 페이지를 누릅니다 .

   CGI as a File Type 창이 나타납니다 .
3. Editing picker 에서 이 변경 사항을 적용할 리소스를 선택합니다 .
4. Activate CGI as a File Type 아래에서 Yes 라디오 버튼을 누릅니다 .
5. OK 를 누릅니다 .
6. 변경 사항을 저장 및 적용합니다 .

CGI 파일은 파일 확장자 `.bat`, `.exe` 또는 `.cgi` 를 가져야 합니다 . 이러한 확장자를 가진 CGI 가 아닌 파일을 서버가 CGI 파일로 처리하면 오류가 발생합니다 .

실행 파일 다운로드

`.exe` 를 CGI 파일 유형으로 사용할 경우 `.exe` 파일을 실행 파일로 다운로드할 수 없습니다 .

이 문제에 대한 한 가지 해결책은 사용자가 다운로드할 수 있도록 하려는 실행 파일을 압축하여 확장자가 `.exe` 가 안 되게 하는 것입니다 . 이 해결책은 다운로드 시간을 단축시키는 추가 장점도 있습니다 .

또 다른 가능한 해결책은 `magnus-internal/cgi` 유형에서 `.exe` 를 파일 확장자로 제거하여 그것을 `application/octet-stream` 유형 (일반 다운로드 가능 파일에 대한 MIME 유형) 대신 추가하는 것입니다 . Server Manager 에서 Server Preferences 탭을 누르고 MIME Types 링크를 선택하여 할 수 있습니다 . 그러나 이 방법의 단점은 변경을 한 후에 `.exe` 파일을 CGI 프로그램으로 사용할 수 없다는 것입니다 .

또 다른 해결책은 디렉토리의 모든 파일이 자동으로 다운로드되는 다운로드 디렉토리를 설정하는 `obj.conf` 파일을 편집하는 것입니다 . 서버의 나머지는 영향을 받지 않습니다 . 더 자세한 내용은 다음을 참조하십시오 .

`http://developer.netscape.com/docs/manuals/enterprise/admnunix/programs.htm`

Windows CGI 프로그램 설치

이 부분은 Windows CGI 프로그램을 설치하는 방법을 다룹니다 . 다음 주제가 이 부분에 포함됩니다 .

Windows CGI 프로그램의 개요

Windows CGI 프로그램은 상당수가 다른 CGI 프로그램으로 처리됩니다 . Windows CGI 프로그램만 포함하는 디렉토리를 지정하거나 Windows CGI 프로그램이 동일한 파일 확장자를 갖도록 지정합니다 . 다른 CGI 프로그램과 같이 원할 경우 동시에 두 가지 방법을 모두 사용할 수 있습니다 . 예를 들어 , 모든 Windows CGI 프로그램에 대한 디렉토리를 만들고 Windows CGI 파일 확장자를 지정할 수 있습니다 .

Windows CGI 프로그램이 일반 CGI 프로그램과 같이 행동하더라도 서버는 실제 프로그램을 조금 다르게 처리합니다 . 따라서 Windows CGI 프로그램에 대해 다른 디렉토리를 지정해야 합니다 . Windows CGI 파일 유형을 사용함으로 설정하면 파일 확장자 `.wcg` 를 사용합니다 .

Sun ONE Web Servers 는 Windows CGI 1.3a 비공식 표준을 지원하며 다음과 같은 차이점이 있습니다 .

- 다음 키워드가 보안 방법을 지원하는 [CGI] 부분에 추가되었습니다 .
  - HTTPS 이 값은 트랜잭션이 SSL 을 통해 시행되는지 여부에 따라 On 또는 Off 입니다 .
  - HTTPS 키 크기 : HTTPS 가 On 이면 이 값은 암호화에 사용되는 세션 키의 비트 수를 보고합니다 .
  - HTTPS 비밀 키 크기 : HTTPS 가 On 이면 이 값은 서버의 전용 키를 생성하는 데 사용되는 비트 수를 보고합니다 .
- [CGI] 부분의 키워드 Document Root 는 서버에 단일한 문서 루트만 있는 것이 아니기 때문에 예상되는 문서 루트를 가리키지 않을 수 있습니다 . 이 변수에서 반환되는 디렉토리는 Windows CGI 프로그램용 루트 디렉토리입니다 .
- [CGI] 부분의 키워드 Server Admin 은 지원되지 않습니다 .
- [CGI] 부분의 키워드 Authentication Realm 은 지원되지 않습니다 .

- 복수 부분 / 양식 데이터 부호화로 전송되는 양식은 지원되지 않습니다 .

Windows CGI **디렉토리 지정**

Windows CGI 전용 디렉토리를 지정하려면 :

1. Class Manager 에서 Programs 탭을 선택합니다 .
2. WinCGI Directory 링크를 누릅니다 .

   WinCGI Directory 창이 나타납니다 .
3. URL Prefix 텍스트필드에 이 디렉토리에 대해 사용하려는 URL 접두사를 입력합니다 .

   즉 , 입력한 텍스트가 URL 에 있는 Windows CGI 프로그램용 디렉토리로 표시됩니다 . 예를 들어 , URL 접두사로 `wcgi-programs` 을 입력하는 경우 해당 Windows CGI 프로그램에 대한 모든 URL 의 구조는 다음과 같습니다 .

   http://*yourserver*.domain.com/wcgi-programs/program-name

참고 지정하는 URL 접두사는 단계 5 에서 지정하는 실제 실제 Windows CGI 디렉토리와 다를 수 있습니다 .

4. 스크립트 추적을 사용함으로 설정할지 선택합니다 .

   "Enable Script Tracing?" 아래에서 Yes 또는 No 라디오 버튼을 누릅니다 .

   CGI 매개변수가 파일을 통하여 서버에서 Windows CGI 프로그램으로 전달되며 , 이 경우 서버는 Windows CGI 프로그램의 실행이 종료된 후 파일을 삭제합니다 . 스크립트 추적을 사용하는 경우 이 파일은 `/temp` 디렉토리 또는 환경 변수 TMP 및 TEMP 가 가리키는 위치에 보관됩니다 . 또한 스크립트 추적을 사용하면 Windows 프로그램이 시작하는 모든 창이 표시됩니다 .
5. WinCGI Directory 필드에서 디렉토리의 위치를 절대 경로로 입력합니다 .

   이 디렉토리가 반드시 문서 루트 아래에 있어야 하는 것은 아닙니다 . 이것이 URL 접두사를 단계 3 에서 지정해야 하는 이유입니다 .
6. OK 를 누릅니다 .
7. 변경 사항을 저장 및 적용합니다 .

기존 Windows CGI 디렉토리를 제거하려면 Windows CGI Directory 양식에서 해당 디렉토리의 Remove 버튼을 누릅니다 . 기존 디렉토리의 URL 접두사 또는 Windows CGI 디렉토리를 변경하려면 해당 디렉토리의 Edit 버튼을 누릅니다 .

지정한 디렉토리로 Windows CGI 프로그램을 복사합니다 . 이러한 디렉토리의 모든 파일은 Windows CGI 파일로 처리된다는 점을 기억하십시오 .

Windows CGI 를 파일 유형으로 지정

Windows CGI 파일에 대해 파일 확장자를 지정하려면 다음 단계를 수행하십시오 .

1. Server Manager 에서 Server Preferences 탭을 누릅니다 .
2. MIME Types 링크를 누릅니다 .

   Global MIME Types 창이 나타납니다 . Global MIME 유형에 대한 더 자세한 내용은 "MIME 유형 선택 " 페이지 171 를 참조하십시오 .
3. 다음 설정으로 MIME 유형을 추가합니다 .
   - 유형 : `type:`
   - 컨텐트 유형 : `magnus-internal/wincgi`.
   - 파일 접미사 : 서버가 Windows CGI 와 연결하게 하려는 파일 접미사를 입력합니다 . CGI, WinCGI 및 셸 CGI 파일 유형을 사용하는 경우 반드시 각 유형의 CGI 에 다른 접미사를 지정해야 합니다 . 예를 들어 , CGI 프로그램과 셸 CGI 프로그램에 모두 `.exe` 접미사를 사용하면 안 됩니다 . 그래야 할 경우 접미사가 고유하도록 페이지의 다른 MIME 유형 필드를 편집하면 됩니다 .
4. New Type 버튼을 누릅니다 .
5. 변경 사항을 저장 및 적용합니다 .

Windows 용 셸 CGI 프로그램 설치

이 부분은 Windows 용 Shell CGI 프로그램을 설치하는 방법에 대해 설명합니다 . 다음 주제가 이 부분에 포함됩니다 .

Windows 용 CGI 프로그램의 개요

셸 CGI 는 Windows 에서 설정된 파일 연결을 사용하여 CGI 응용 프로그램을 실행하도록 하는 서버 구성입니다 .

예를 들어 , 서버가 `hello.pl` 이라고 하는 셸 CGI 파일에 대한 요청을 받으면 서버는 Windows 파일 연결을 통해 `.pl` 확장자와 연결된 프로그램을 사용하여 파일을 실행합니다 . `.pl` 확장자가 프로그램 `C:\bin\perl.exe` 와 연결되면 서버는 `hello.pl` 파일을 다음과 같이 실행하려 시도합니다 .

```
c:\bin\perl.exe hello.pl
```

셸 CGI 를 구성하는 가장 쉬운 방법은 오직 셸 CGI 파일만 포함하는 디렉토리를 서버의 문서 루트 아래에 만드는 것입니다 . 그러나 Sun ONE Web Server 에서 MIME 유형을 편집하여 특정한 파일 확장자를 셸 CGI 와 연결하도록 서버를 구성할 수도 있습니다 .

참고 Windows 파일 확장자 설정에 대한 더 자세한 내용은 Windows 설명서를 참조하십시오 .

셸 CGI 디렉토리 지정 (Windows)

셸 CGI 파일용 디렉토리를 만들려면 다음 단계를 수행하십시오 .

1. 컴퓨터에서 셸 디렉토리를 만듭니다 . 이 디렉토리가 반드시 문서 루트 디렉토리의 하위 디렉토리일 필요는 없습니다 .
2. Server Manager 에서 Class Manager 링크를 선택합니다 .
3. 다음으로 Class Manager 를 선택합니다 .

   셸 CGI Directory 링크가 강조되고 CGI 창이 나타납니다 .
4. URL Prefix 필드에서 셸 CGI 디렉토리와 연결하려는 URL 접두사를 입력합니다.

   예를 들어 , 모든 셸 CGI 파일을 디렉토리 `C:\docs\programs\cgi\shell-cgi` 에 저장한다고 가정합니다 . 그러나 사용자에게 해당 디렉토리가 `http://www.`*yourserver*`.com/shell/` 로 보이도록 하려 합니다 . 이 경우 `shell` 을 URL 접두사로 입력합니다 .
5. Shell CGI Directory 필드에서 만든 디렉토리에 대한 절대 경로를 입력합니다 .

주의 서버에 반드시 이 디렉토리에 대한 읽기 및 실행 권한이 있어야 합니다 . Windows 의 경우 서버를 실행하는 사용자 계정 (예를 들어 `LocalSystem`) 에 셸 CGI 디렉토리의 프로그램을 읽고 실행할 수 있는 권한이 있어야 합니다 .

6. 또한 셸 CGI 디렉토리의 모든 파일에 Windows 에서 설정한 파일 연결이 있는지 확인하십시오 . 파일 확장자 연결이 없는 파일을 실행하면 서버에 오류가 발생합니다 .

셸 CGI 를 파일 유형으로 지정 (Windows)

Sun ONE Web Server 의 MIME Types 창을 사용하여 파일 확장자를 셸 CGI 기능과 연결할 수 있습니다 . Windows 에서 연결을 만드는 것과는 다릅니다 .

서버에서 파일 확장자를 셸 CGI 기능과 연결하려면 예를 들어, `.pl` 확장자로 파일에 대한 연결을 만들 수 있습니다 . 서버에 해당 확장자가 있는 파일이 요청되면 Windows 에서 해당 파일 확장자와 연결된 실행 파일을 호출함으로써 해당 파일을 셸 CGI 파일로 처리합니다 .

파일 확장자를 셸 CGI 파일과 연결하려면 다음 단계를 수행하십시오 .

1. 컴퓨터에서 셸 디렉토리를 만듭니다 . 이 디렉토리가 반드시 문서 루트 디렉토리의 하위 디렉토리일 필요는 없습니다 .
2. Server Manager 에서 Server Preferences 를 선택합니다 .
3. MIME Types 링크를 누릅니다 .

   Global MIME Types 창이 나타납니다 . Global MIME 유형에 대한 더 자세한 내용은 "MIME 유형 선택 " 페이지 171 를 참조하십시오 .
4. 다음 설정으로 새로운 MIME 유형을 추가합니다 .
   - 유형 : `type:`
   - 컨텐트 유형 : `magnus-internal/shellcgi`.
   - 파일 접미사: 서버가 셸 CGI 와 연결하게 하려는 파일 접미사를 입력합니다. CGI, WinCGI 및 셸 CGI 파일 유형을 사용하는 경우 반드시 각 유형의 CGI 에 다른 접미사를 지정해야 합니다 . 예를 들어 , CGI 프로그램과 셸 CGI 프로그램에 모두 `.exe` 접미사를 사용하면 안 됩니다 . 그래야 할 경우 접미사가 고유하도록 페이지의 다른 MIME 유형 필드를 편집하면 됩니다 .
5. New Type 버튼을 누릅니다 .

6. 변경 사항을 저장 및 적용합니다 .

쿼리 처리기 사용

참고 쿼리 처리기의 사용은 진부한 방법입니다 . Sun ONE Web Server 및 Netscape Navigator 클라이언트가 이것을 지원하기는 하지만 사용되는 경우는 드뭅니다 . 사람들이 HTML 페이지의 양식을 사용하여 쿼리를 제출하는 경우가 훨씬 더 일반적입니다 .

기본 쿼리 처리기 CGI 프로그램을 지정할 수 있습니다 . 쿼리 처리기는 HTML 파일의 ISINDEX 태그를 통하여 보내진 텍스트를 처리합니다 .

ISINDEX 는 HTML 페이지에서 텍스트 필드를 만드는 형식 텍스트 필드와 유사하며 , 여기에서 입력을 받을 수 있습니다 . 그러나 페이지 텍스트 필드의 정보와 달리 ISINDEX 상자의 정보는 사용자가 Return 을 누를 때 바로 제출됩니다 . 기본 쿼리 처리기를 지정하면 서버가 어느 프로그램에 입력을 전달할 지 알 수 있습니다 . ISINDEX 태그에 대한 자세한 설명은 HTML 참조 설명서를 참조하십시오 .

쿼리 처리기를 만들려면 다음과 같이 합니다 .

1. Class Manager 에서 Programs 탭을 선택합니다 .
2. Query Handler 링크를 누릅니다 .

   Query Handler 창이 나타납니다 .
3. Editing Picker 를 사용하여 기본 쿼리 처리기로 설정하려는 리소스를 선택합니다 .

   디렉토리를 선택하면 서버가 해당 디렉토리 또는 디렉토리의 파일용 URL 을 수신하는 경우에만 지정한 쿼리 처리기가 실행됩니다 .
4. Default Query Handler 필드에서 선택한 리소스에 대해 기본적으로 사용하려는 CGI 프로그램에 대한 전체 경로를 입력합니다 .
5. OK 를 누릅니다 .
6. 변경 사항을 저장 및 적용합니다 .

16 장

내용 관리

이 장은 가상 서버의 클래스와 가상 서버의 내용을 구성 및 관리하는 방법에 대해 설명합니다 .

이 장의 내용 :

주 문서 디렉토리 설정

주 문서 디렉토리 (문서 루트라고도 함) 는 원격 클라이언트가 사용할 수 있도록 할 모든 파일을 저장하는 중앙 디렉토리입니다 .

클래스를 추가하면 절대 경로로 문서 디렉토리를 지정합니다 . 해당 경로의 일부로 변수를 사용하지 않으면 클래스의 모든 가상 서버의 문서 루트는 기본적으로 같은 디렉토리가 됩니다 . 그런 다음 Class Manager 에서 개별적으로 디렉토리를 변경할 수 있습니다 .

또 다른 방법은 클래스에 경로를 설정할 때 변수를 사용하는 것입니다 . 예를 들어 , 클래스의 모든 가상 서버에 대해 가상 서버 id 로 이름 지정된 디렉토리를 작성하기 위해 `$id` 변수를 사용할 수 있습니다 . 클래스의 문서 루트를 *class_doc_root*/$id 로 설정할 수 있습니다 . 이 경로를 사용하면 클래스의 문서 디렉토리가 `/sun/servers/docs/$id` 인 경우 클래스에 속한 가상 서버 `vs1` 의 주 문서 디렉토리는 `/sun/servers/docs/vs1` 입니다 .

문서 디렉토리 및 서버 인스턴스 , 클래스 , 가상 서버 수준에서 문서 디렉토리를 사용하는 방법에 대한 자세한 내용은 " 문서 루트 " 페이지 309 를 참조하십시오 .

다른 경로 또는 변수를 사용하기 위해 주 문서 디렉토리를 변경하려면 다음 단계를 따르십시오 .

1. Class Manager 에서 Content Management 탭을 누릅니다 .
2. Primary Document Directory 를 누릅니다 .
3. 가상 서버 옆에 절대 디렉토리 경로나 변수 또는 경로 및 변수 조합을 입력합니다 .

   문서 루트 절대 경로의 끝에 변수 $id 를 포함시키면 모든 가상 서버가 기본적으로 *class_doc_root*/*virtual_server_ID* 의 주 문서 루트를 갖습니다 예를 들어 , 클래스의 문서 디렉토리는 `/sun/servers/docs/$id` 이고 클래스에 속한 가상 서버 `vs1` 의 주 문서 디렉토리는 `/sun/servers/docs/vs1` 입니다 .

   변수에 대한 더 자세한 내용은 " 변수 사용 " 페이지 313 를 참조하십시오 .
4. OK 를 누릅니다 .

자세한 내용은 Primary Document Directory 페이지에 대한 온라인 도움말을 참조하십시오 .

참고 각 가상 서버는 보통 자체의 주 문서 디렉토리가 있습니다 .

추가 문서 디렉토리 설정

대부분 , 가상 또는 서버 인스턴스의 문서는 주 문서 디렉토리에 있습니다 . 그러나 때로는 문서 루트 외부의 디렉토리에서 문서를 서비스하고자 할 수 있습니다 . 이 경우 추가 문서 디렉토리를 설정함으로써 가능합니다 . 문서 루트 외부의 문서 디렉토리에서 서비스함으로써 누군가에게 주 문서 루트에 대한 액세스를 부여하지 않고 문서 그룹을 관리하도록 할 수 있습니다 .

변수를 사용하지 않고 추가 문서 디렉토리를 설정하면 해당 디렉토리는 클래스 수준에서 설정되고 클래스에 있는 모든 가상 서버에 의해 사용됩니다 .

클래스에 있는 개별 가상 서버용 추가 문서 디렉토리를 설정하려면 변수를 사용하여 해당 URL 접두사가 매핑하는 디렉토리가 각 가상 서버마다 다르게 합니다 .

추가 문서 디렉토리를 추가하려면 다음 단계를 따르십시오 .

1. Class Manager 에서 Content Management 탭을 누릅니다 .
2. Additional Document Directories 를 누릅니다 .
3. 매핑할 URL 접두사를 선택합니다 .

   클라이언트가 문서를 원할 때 서버에 이 URL 을 보냅니다 .
4. 해당 URL 을 매핑할 디렉토리를 지정합니다 .
5. 원할 경우 기존 구성 스타일을 사용하여 이 디렉토리를 구성할 방법을 지정합니다 .
6. OK 를 누릅니다 .

자세한 내용은 User Document Directories 페이지에 대한 온라인 도움말을 참조하십시오 .

기본적으로 서버 인스턴스는 몇 개의 추가 문서 디렉토리를 갖습니다 . 해당 디렉토리에는 다음 접두사가 있습니다

- `/manual`
- `/servlet`

사용자가 해당 디렉토리에 기록할 수 없도록 디렉토리에 대한 액세스를 제한해야 합니다 . 예제 ACL 은 다음과 같습니다 .

```
deny (all) anyone;
allow (rxli) all;
allow (wd) privileged_user;
```

공용 정보 디렉토리 사용자 정의 (UNIX/Linux)

때로는 사용자가 자신의 웹 페이지를 유지하려 합니다 . 서버의 모든 사용자가 관리자 간섭 없이 홈페이지와 기타 문서를 만들 수 있도록 하는 공용 정보 디렉토리를 구성할 수 있습니다 .

이 옵션은 전체 클래스에 대하여만 설정할 수 있습니다 . 가상 서버 단위로 이를 사용자 정의할 수 있는 방법은 없습니다 .

이 시스템으로 클라이언트는 서버가 공용 정보 디렉토리로 인지하는 특정 URL 로 서버에 액세스할 수 있습니다 . 예를 들어 , 접두사 ~ 와 디렉토리 `public_html` 을 선택한다고 가정합니다 . `http://www.sun.com/~jdoe/aboutjane.html` 에 대한 요청이 들어오면 서버는 `~jdoe` 가 사용자의 공용 정보 디렉토리를 가리킨다는 사실을 인지합니다 . 서버는 시스템의 사용자 데이터베이스에서 `jdoe` 를 조회하고 Jane 의 홈 디렉토리를 발견합니다 . 그런 다음 `~/jdoe/public_html/aboutjane.html` 을 찾습니다 .

서버가 공용 디렉토리를 사용하도록 구성하려면 다음 단계를 따르십시오 .

1. Class Manager 에서 Content Management 탭을 누릅니다 .
2. User Document Directories 를 누릅니다 .
3. 사용자 URL 접두사를 선택합니다 .

   사용자 접두사는 ~ 입니다 . 왜냐하면 ~ 문자가 사용자의 홈 디렉토리에 액세스하기 위한 표준 UNIX/Linux 접두사이기 때문입니다 .
4. 서버가 HTML 파일을 찾는 사용자의 홈 디렉토리의 하위 디렉토리를 선택합니다 .

   일반적인 디렉토리는 `public_html` 입니다 .
5. 암호 파일을 지정합니다 .

   서버는 시스템에 있는 사용자를 목록화하는 파일을 어디서 찾을지 알아야 합니다 . 서버는 이 파일을 사용하여 유효한 사용자 이름을 결정하고 그의 홈 디렉토리를 찾습니다 . 시스템 암호 파일을 이 목적으로 사용하면 서버는 표준 라이브러리 호출을 사용하여 사용자를 조회합니다 . 또는 , 다른 사용자 파일을 작성하여 사용자를 찾을 수 있습니다 . 해당 사용자 파일은 절대 경로로 지정할 수 있습니다 .

   파일의 각 줄은 다음 구조를 가져야 합니다 (필요하지 않은 `/etc/passwd` 파일의 요소는 * 로 표시됨).

   ```
   username:*:*:groupid:*:homedir:*
   ```

6. 시작시 암호 데이터베이스를 로드할지 선택합니다 .

   더 자세한 내용은 " 시작시 전체 암호 파일 로드 " 페이지 365 를 참조하십시오 .

7. 구성 스타일을 적용할지 선택합니다 .

8. OK 를 누릅니다 .

자세한 내용은 User Document Directories 페이지에 대한 온라인 도움말을 참조하십시오 .

사용자에게 별도의 디렉토리를 부여하는 또 다른 방법은 모든 사용자가 수정할 수 있는 중앙 디렉토리에 매핑되는 URL 을 작성하는 것입니다 .

내용 게시 제한

어떤 상황에서는 시스템 관리자가 사용자 문서 디렉토리를 통해 내용을 게시할 수 있는 사용자 계정을 제한하고자 할 수 있습니다 . 사용자의 게시를 제한하려면 `/etc/passwd file` 의 사용자의 홈 디렉토리 경로 끝에 슬래시를 추가합니다 .

```
jdoe::1234:1234:John Doe:/home/jdoe:/bin/sh
```

becomes:

```
jdoe::1234:1234:John Doe:/home/jdoe/:/bin/sh
```

이 수정을 하면 Sun ONE Web Server 는 이 사용자의 디렉토리에서 페이지를 서비스하지 않습니다 . 해당 URI 를 요청하는 브라우저는 "404 File Not Found" 오류를 수신하고 404 오류는 웹 서버 액세스 로그에 기록됩니다 . 오류 로그에 기록되는 오류는 없습니다 .

나중에 이 사용자로 하여금 내용을 게시하도록 허용하기로 하면 `/etc/passwd` 항목에서 끝에 오는 슬래시를 제거한 다음 웹 서버를 재시작합니다 .

시작시 전체 암호 파일 로드

시작시 전체 암호 파일을 로드하는 옵션도 있습니다 . 이 옵션을 선택하면 서버는 시작시 암호 파일을 메모리에 로드하여 사용자 조회 속도를 훨씬 더 빠르게 만듭니다 . 암호 파일이 매우 대용량이면 이 옵션이 훨씬 더 많은 메모리를 사용할 수 있습니다 .

구성 스타일 사용

서버가 공용 정보 디렉토리에서 디렉토리로의 액세스를 제어하도록 하는 구성 스타일을 적용할 수 있습니다 . 이렇게 하면 사용자가 공용으로 만들고 싶지 않은 정보에 심볼 링크를 작성하지 못하게 됩니다 . 구성 파일에 대한 더 자세한 내용은 제 17 장 , " 구성 스타일 적용 " 을 참조하십시오 .

원격 파일 조작 사용

원격 파일 조작을 사용하면 클라이언트가 서버의 파일 업로드 , 파일 삭제 , 디렉토리 생성 , 디렉토리 제거 , 디렉토리 내용 목록 표시 및 파일 이름 변경 등의 작업을 할 수 있도록 합니다 . 디렉토리 *server_root*/https-*serve-id*/config 의 파일 obj.conf 에는 원격 파일 조작을 사용할 경우 활성화되는 명령이 들어 있습니다 . 이러한 명령을 활성화하여 원격 브라우저가 서버의 문서를 변경하도록 합니다 . 권한 없는 간섭을 방지하기 위해 이러한 리소스에 대해 쓰기 액세스를 제한하는 액세스 제어를 사용해야 합니다 .

참고로 원격 파일 조작 사용은 Microsoft Frontpage 와 같은 내용 관리 시스템 사용에 아무런 영향이 없어야 합니다 .

UNIX/Linux: 파일에 대해 올바른 권한을 가져야 합니다 . 그렇지 않으면 이 기능이 제대로 작동하지 않습니다 . 즉 , 문서 루트 사용자가 서버 사용자와 동일해야 합니다 .

원격 파일 조작을 사용하려면 다음 단계를 따르십시오 .

1. Class Manager 에서 Content Management 탭을 누릅니다 .
2. Remote File Manipulation 을 누릅니다 .
3. 원격 파일 조작을 활성화할 것을 선택합니다 .
4. OK 를 누릅니다 .

자세한 내용은 Remote File Manipulation 페이지의 온라인 도움말을 참조하십시오 .

문서 기본설정 구성

Document Preferences 페이지를 사용하여 문서 기본설정을 설정합니다 . 이 부분은 다음 주제에 대해 다룹니다 .

- 문서 기본설정 설정
- 색인 파일 이름 입력
- 디렉토리 색인화 선택
- 서버 홈 페이지 지정
- 기본 MIME 유형 지정

이러한 설정은 모두 개별 가상 서버가 아닌 클래스에 대해 구성됩니다 .

문서 기본설정 설정

문서 기본설정을 설정하려면 다음 단계를 따르십시오 .

1. Class Manager 에서 Content Management 탭을 누릅니다 .
2. Document Preferences 를 누릅니다 .
3. 해당하는 필드 값을 선택합니다 . 이에 대해서는 다음 부분에서 다룹니다 .
4. OK 를 누릅니다 .

설정할 수 있는 기본설정은 다음 부분에서 보다 충분히 다룹니다 . 추가 정보는 Document Preferences 페이지에 대한 온라인 도움말을 참조하십시오 .

색인 파일 이름 입력

문서 이름이 URL 에서 지정되지 않으면 서버는 색인 파일을 자동으로 표시합니다 . 기본 색인 파일은 `index.html` 과 `home.html` 입니다 . 색인 파일이 하나 이상 지정되면 서버는 이름이 발견될 때까지 이름이 이 필드에서 나타나는 순서로 찾습니다 . 예를 들어 , 색인 파일 이름이 `index.html` 및 `home.html` 이고 서버가 `index.html` 을 찾았지만 발견하지 못하면 `home.html` 을 찾습니다 .

디렉토리 색인화 선택

문서 디렉토리는 몇 개의 하위 디렉토리를 갖습니다 . 예를 들어 , `products` 라는 디렉토리 , `people` 이라는 디렉토리 등이 있을 수 있습니다 . 대개 클라이언트가 이 디렉토리의 개요 (또는 색인) 에 액세스하는 것이 유용합니다 .

서버는 index.html 또는 home.html 이라는 , 디렉토리 내용의 개요를 작성하고 유지하는 색인 파일용 디렉토리를 찾음으로써 디렉토리를 색인화합니다 . 더 자세한 내용은 다음 부분 , " 색인 파일 이름 입력 " 페이지 367 을 참조하십시오 . 원하는 파일에 이 기본 이름을 부여하여 디렉토리용 색인 파일로 지정할 수 있으며 , 따라서 CGI 를 사용하는 경우 CGI 프로그램을 색인으로 사용할 수 있습니다 .

색인 파일이 없으면 서버는 문서 루트의 모든 파일을 목록화하는 색인 파일을 생성합니다 .

주의 서버가 방화벽 외부에 있는 경우에는 디렉토리 색인화를 사용하지 않도록 하여 디렉토리 구조와 파일 이름에 액세스할 수 없도록 하십시오 .

서버 홈 페이지 지정

최종 사용자가 처음으로 서버에 액세스할 경우 가장 먼저 보게 되는 파일을 보통 홈 페이지라고 합니다 . 일반적으로 , 이 파일은 서버와 다른 문서로의 링크에 대한 일반 정보를 담고 있습니다 .

기본적으로 서버는 Document Preferences 페이지의 색인 파일 이름 필드에 지정된 색인 파일을 찾고 이를 홈 페이지용으로 사용합니다 . 그러나 홈 페이지용으로 사용할 파일을 지정할 수도 있습니다 .

기본 MIME 유형 지정

문서가 클라이언트에게 전송되면 서버는 문서의 유형을 식별하는 부분을 포함시켜 클라이언트가 문서를 바로 제시할 수 있도록 합니다 . 그러나 , 때로는 문서의 확장자가 서버에 대해 정의되지 않아서 문서의 적절한 유형을 결정할 수 없습니다 . 그런 경우에는 기본값이 전송됩니다 .

기본값은 보통 text/plain 이나 , 서버에 가장 일반적으로 저장되는 유형의 파일을 설정해야 합니다 . 일반적인 MIME 유형은 다음과 같습니다 .

- text/plain
- text/richtext
- image/jpeg
- application/x-tar
- text/html
- image/tiff
- image/gif
- application/postscript

- `application/x-gzip`
- `audio/basic`

URL 전달 구성

URL 전달을 통해 문서 요청을 다른 서버로 재지정할 수 있습니다 . URL 전달 또는 재지정은 서버가 사용자에게 (예를 들어 파일을 다른 디렉토리 또는 서버로 옮겼으므로) URL 이 변경되었음을 알리는 방법입니다 . 또한 재지정을 사용하여 서버에서 문서를 요청하는 사용자를 사용자가 모르는 상태에서 다른 서버의 문서로 보낼 수 있습니다 .

예를 들어 , `http://www.sun.com/info/movies` 를 접두사 `film.sun.com` 으로 전달하면 URL `http://www.sun.com/info/movies` 가 `http://film.sun.com/info/movies` 로 재지정됩니다 .

변수를 사용하여 디렉토리를 새 디렉토리에 매핑할 수 있습니다 . 예를 들어 , `/new` 를 `/$docroot/new` 로 매핑할 수 있습니다 . 매핑은 가상 서버의 문서 루트로 이동합니다 .

변수에 대한 더 자세한 내용은 " 변수 사용 " 페이지 313 를 참조하십시오 .

때로는 한 하위 디렉토리의 모든 문서에 대한 요청을 특정 URL 로 재지정할 경우가 있습니다 . 예를 들어 , 어떤 디렉토리에 너무 많은 트래픽이 발생하거나 해당 문서가 어떤 이유이든 더 이상 서비스되지 않으므로 이 디렉토리를 삭제해야 하는 경우 문서를 더 이상 사용할 수 없음을 표시하는 문서로 해당 요청을 보낼 수 있습니다. 예를 들어 , `/info/movies` 의 접두사는 `http://www.sun.com/explain.html` 로 재지정될 수 있습니다 .

URL 전달을 구성하려면 다음 단계를 따르십시오 .

1. Class Manager 에서 Content Management 탭을 누릅니다 .
2. URL Forwarding 을 누릅니다 .
3. 재지정하려는 URL 접두사를 입력하고 그것을 다른 접두사나 정적 URL 로 재지정할 것인지 입력합니다 .
4. OK 를 누릅니다 .

더 자세한 내용은 URL Forwarding 페이지에 대한 온라인 도움말을 참조하십시오 .

오류 응답 사용자 정의

사용자 정의 오류 응답은 가상 서버에서 오류가 수신된 경우 클라이언트에게 자세한 메시지를 보냅니다 . 전송할 파일이나 실행할 CGI 프로그램을 지정할 수 있습니다 .

예를 들어 , 서버가 특정 디렉토리에 대한 오류를 수신한 경우 작동하는 방식을 변경할 수 있습니다 . 클라이언트가 액세스 제어로 보호된 서버의 일부에 연결하려 하면 계정을 얻는 방법에 대한 정보를 담은 오류 파일을 반환할 수 있습니다 .

사용자 정의 오류 응답을 사용하도록 설정하기 전에 오류에 대한 응답으로 전송할 HTML 파일이나 실행할 CGI 프로그램을 작성해야 합니다 . 이렇게 한 뒤 Class Manager 에서 응답을 사용하도록 설정합니다 .

사용자 정의된 오류 응답을 사용하려면 다음 단계를 따르십시오 .

1. Class Manager 에서 Content Management 탭을 누릅니다 .
2. Error Responses 를 누릅니다 .
3. 리소스 선택자에서 Entire Server 를 선택하여 변경 사항을 전체 클래스에 적용하거나 특정 가상 서버용 문서 루트 또는 특정 디렉토리나 특정 가상 서버 내부를 탐색합니다 .
4. 변경하려는 각 오류 코드에 대한 오류 응답을 담은 파일이나 CGI 의 절대 경로를 지정합니다 .
5. OK 를 누릅니다 .

더 자세한 내용은 Error Responses 페이지에 대한 온라인 도움말을 참조하십시오 .

문자 세트 변경

문서의 문자 세트는 문서가 작성된 언어에 의해 일부 결정됩니다 . 리소스를 선택하고 해당 리소스의 문자 세트를 입력하여 문서에 대한 클라이언트의 기본 문자 세트 설정 , 문서 세트 또는 디렉토리보다 우선할 수 있습니다 .

Netscape Navigator 는 문자 세트를 변경할 HTTP 에서 MIME 유형 `charset` 매개변수를 사용할 수 있습니다 . 서버가 응답에서 이 매개변수를 포함하면 Netscape Navigator 는 그에 따라 문자 세트를 변경합니다 . 예를 들면 다음과 같습니다 .

- `Content-Type: text/html;charset=iso-8859-1`
- `Content-Type: text/html;charset=iso-2022-jp`

Netscape Navigator 에 의해 인지되는 다음 charset 이름은 RFC 1700 에서 지정됩니다 (x- 로 시작되는 이름 제외).

- us-ascii
- iso-8859-1
- iso-2022-jp
- x-sjis
- x-euc-jp
- x-mac-roman

또 , 다음 별칭은 us-ascii 에 대해 인지됩니다 .

- ansi_x3.4-1968
- iso-ir-6
- ansi_x3.4-1986
- iso_646.irv:1991
- ascii
- iso646-us
- us
- ibm367
- cp367

다음 별칭은 iso_8859-1 에 대해 인지됩니다 .

- latin1
- iso_8859-1
- iso_8859-1:1987
- iso-ir-100
- ibm819
- cp819

문자 세트를 변경하려면 다음 단계를 따르십시오 .

1. Class Manager 에서 Content Management 탭을 누릅니다 .
2. International Characters 를 누릅니다 .
3. 리소스 선택자에서 Entire Server 를 선택하여 변경 사항을 전체 클래스에 적용하거나 특정 가상 서버용 문서 루트 또는 특정 디렉토리나 특정 가상 서버 내부를 탐색합니다 .

4. 서버 전체 또는 일부에 대한 문자 세트를 설정합니다 .

   이 필드를 공백으로 두면 문자 세트는 NONE 으로 설정됩니다 .

5. OK 를 누릅니다 .

더 자세한 내용은 International Characters 페이지에 대한 온라인 도움말을 참조하십시오 .

문서 꼬리말 설정

서버의 특정 부분의 모든 문서에 대해 마지막 수정된 시간을 포함할 수 있는 문서 꼬리말을 지정할 수 있습니다 . 이 꼬리말은 CGI 스크립트의 출력이나 파싱된 HTML(.shtml) 파일을 제외한 모든 파일에 사용됩니다 . CGI 스크립트 출력이나 파싱된 HTML 파일에 문서 꼬리말을 표시해야 하는 경우 , 꼬리말 텍스트를 별도의 파일에 입력하고 코드 라인을 추가하거나 , 다른 서버 측에서 이 파일을 페이지의 출력에 포함하도록 합니다 .

문서 꼬리말을 설정하려면 다음 단계에 따르십시오 .

1. Class Manager 에서 Content Management 탭을 누릅니다 .
2. Document Footer 를 누릅니다 .
3. 리소스 선택자에서 Entire Server 를 선택하여 변경 사항을 전체 클래스에 적용하거나 특정 가상 서버용 문서 루트 또는 특정 디렉토리나 특정 가상 서버 내부를 탐색합니다 .

   디렉토리를 선택하면 서버가 해당 디렉토리 또는 디렉토리의 파일용 URL 을 수신하는 경우에만 문서 꼬리말이 적용됩니다 .

4. 꼬리말에 포함시킬 파일 유형을 지정합니다 .
5. 날짜 형식을 지정합니다 .
6. 꼬리말에 표시할 텍스트를 입력합니다 .

   문서 꼬리말에 대한 최대 문자 수는 765 입니다 . 문서가 마지막으로 수정된 날짜를 포함시키려면 문자열 `:LASTMOD:` 를 입력합니다 .

7. OK 를 누릅니다 .

더 자세한 내용은 Document Footer 페이지에 대한 온라인 도움말을 참조하십시오 .

htaccess 사용

htaccess 에 대한 자세한 내용은 ".htaccess 파일 사용 " 페이지 212 을 참조하십시오 .

심볼 링크 (UNIX/Linux) 제한

서버의 파일 시스템 링크의 사용을 제한할 수 있습니다 . 파일 시스템 링크는 다른 디렉토리 또는 파일 시스템에 저장된 파일을 참조합니다 . 참조를 통하여 파일이 현재 디렉토리에 있는 것처럼 원격에서 액세스할 수 있습니다 . 다음 두 가지 유형의 파일 시스템 링크가 있습니다 .

- **하드 링크 -** 동일한 데이터 블록 집합을 가리키는 두 개의 파일 이름으로 , 원래 파일과 링크가 동일합니다 . 따라서 하드 링크는 다른 파일 시스템에 적용할 수 없습니다 .
- **심볼 (소프트) 링크 -** 심볼 링크는 데이터가 있는 원래 파일과 원래 파일을 가리키는 다른 파일로 구성됩니다 . 심볼 링크는 하드 링크보다 유연성이 있습니다 . 심볼 링크는 다른 파일 시스템에 걸쳐 사용하고 디렉토리에 링크할 수 있습니다 .

하드 및 심볼 링크에 대한 자세한 내용은 UNIX/Linux 시스템 설명서를 참조하십시오 .

파일 시스템 링크는 기본 문서 디렉토리 외부의 문서에 대해 포인터를 작성할 수 있는 쉬운 방법으로 누구나 이러한 링크를 작성할 수 있습니다 . 이 때문에 사람들이 민감한 파일 (예를 들어 , 기밀 문서 또는 시스템 암호 파일) 에 대해 포인터를 작성하는 문제를 염려할 수 있습니다 .

심볼 링크를 제한하려면 다음 단계를 따르십시오 .

1. Class Manager 에서 Content Management 탭을 누릅니다 .
2. Symbolic Links 를 누릅니다 .
3. 리소스 선택자에서 Entire Server 를 선택하여 변경 사항을 전체 클래스에 적용하거나 특정 가상 서버용 문서 루트 또는 특정 디렉토리나 특정 가상 서버 내부를 탐색합니다 .
4. 소프트 및 / 또는 하드 링크 및 시작할 디렉토리를 사용할지 선택합니다 .
5. OK 를 누릅니다 .

더 자세한 내용은 Symbolic Link 페이지에 대한 온라인 도움말을 참조하십시오 .

서버 파싱 HTML 설정

HTML 은 보통 아무런 서버 작업 없이 디스크에 있는 그대로 클라이언트에게 보내집니다 . 그러나 서버는 문서를 보내기 전에 HTML 파일에 있는 특수한 명령을 검색할 수 있습니다 . (즉 , HTML 을 파싱할 수 있습니다 .) 서버가 파일을 파싱하고 파일에 요청에 대한 정보나 파일을 삽입하도록 하려면 우선 HTML 파싱을 사용하도록 설정해야 합니다 .

HTML 을 파싱하려면 다음 단계를 따르십시오 .

1. Class Manager 에서 Content Management 탭을 누릅니다 .
2. Parse HTML 을 누릅니다 .
3. 서버가 HTML 을 파싱할 리소스를 지정합니다 .

   리소스 선택자에서 Entire Server 를 선택하여 변경 사항을 전체 클래스에 적용하거나 특정 가상 서버용 문서 루트 또는 특정 디렉토리나 특정 가상 서버 내부를 탐색합니다 .

   디렉토리를 선택하면 서버가 해당 디렉토리 또는 디렉토리의 파일용 URL 을 수신하는 경우에만 서버가 HTML 을 파싱합니다 .
4. 서버 파싱 HTML 을 사용할지 선택합니다 .

   HTML 파일은 사용하지만 exec 태그는 사용하지 않거나 HTML 파일과 exec 태그를 사용하여 HTML 파일이 서버의 다른 프로그램을 실행하도록 할 수 있습니다 .
5. 파싱할 파일을 선택합니다 .

   shtml 확장자를 가진 파일만 파싱할지 또는 성능이 저하되더라도 모든 HTML 파일을 파싱할지 선택할 수 있습니다 . UNIX/Linux 를 사용하는 경우 , 신뢰성이 떨어지더라도 실행 권한이 설정된 UNIX/Linux 파일을 파싱할 것을 선택할 수 있습니다 .
6. OK 를 누릅니다 .

파싱된 HTML 을 수신하도록 서버를 설정하는 더 자세한 내용은 Parse HTML 페이지에 대한 온라인 도움말을 참조하십시오 .

서버 파싱 HTML 사용에 대한 더 자세한 내용은 Sun ONE Web Server 6.1 *Programmer's Guide* 를 참조하십시오 .

캐시 제어 지시문 설정

캐시 제어 지시문은 프록시 서버가 어떤 정보를 캐시할 것인지 제어하는 Sun ONE Web Server 의 방법 중 한 가지입니다 . 캐시 제어 지시문을 사용하면 프록시의 기본 캐시 작업을 변경하여 중요한 정보가 캐시되거나 이후 검색되지 않도록 보호할 수 있습니다 . 이 지시문을 사용하려면 프록시 서버가 반드시 HTTP 1.1 을 사용해야 합니다 .

HTTP 1.1 에 대한 자세한 내용은 다음 웹 페이지의 하이퍼텍스트 전송 프로토콜 -- HTTP/1.1 표준 (RFC 2068) 을 참조하십시오 .

http://www.ietf.org/

캐시 제어 지시문을 설정하려면 다음 단계를 따르십시오 .

1. Class Manager 에서 Content Management 탭을 누릅니다 .
2. Cache Control Directives 를 누릅니다 .
3. 필드에 값을 입력합니다 . 응답 지시문으로 유효한 값은 다음과 같습니다 .
   - **Public.** 임의의 캐시로 응답을 캐시할 수 있습니다 . 기본값입니다 .
   - **Private.** 오직 개인용 (비공유) 캐시로만 응답을 캐시할 수 있습니다 .
   - **No Cache.** 응답을 캐시하지 않습니다 .
   - **No Store.** 캐시가 요청이나 응답을 영구적 저장 장소에 저장할 수 없습니다 .
   - **Must Revalidate.** 원래 서버에서 반드시 캐시 항목을 재확인합니다 .
   - **Maximum Age (sec).** 클라이언트는 이 지속시간보다 오래된 응답을 허용하지 않습니다 .
4. OK 를 누릅니다 .

더 자세한 내용은 Cache Control Directives 페이지에 대한 온라인 도움말을 참조하십시오 .

고급 암호화 사용

고급 암호화 설정에 대한 더 자세한 내용은 " 고급 보안 설정 " 페이지 144 를 참조하십시오 .

내용 압축으로 서버 구성

Sun ONE Web Server 6.1 은 HTTP 내용 압축을 지원합니다 . 내용 압축을 사용하면 클라이언트로 전송 속도가 빨라지고 하드웨어 비용을 증가시키지 않고 더 큰 용량의 내용을 서비스할 수 있습니다 . 내용 압축은 내용 다운로드 시간을 떨어뜨리지만 전화 접속 및 높은 수준의 트래픽 연결 사용자는 더 많은 혜택을 누릴 수 있습니다 .

내용 압축을 사용하여 웹 서버는 압축된 데이터를 전송하고 브라우저에게 전송 중에 데이터를 압축 해제할 것을 지시하여 전송된 데이터 양을 감소시키고 페이지 표시 속도를 높입니다 .

서버를 두 가지 방법으로 구성하여 압축된 데이터를 처리할 수 있습니다 .

- 서버를 미리 압축된 내용을 서비스하도록 구성
- 필요시 내용 압축으로 서버 구성

서버의 압축 처리 기능 향상에 대한 더 자세한 내용은 obj.conf 의 압축 관련 변경 사항를 참조하십시오 .

서버를 미리 압축된 내용을 서비스하도록 구성

지정된 디렉토리에 미리 압축된 파일 버전을 생성하고 저장하도록 Sun ONE Web Server 를 구성할 수 있습니다 . 서버를 구성하면 `Accept-encoding: gzip` 헤더만 수신된 경우에만 미리 압축된 내용을 서비스하도록 구성된 디렉토리의 파일에 대한 모든 요청이 그러한 파일이 존재하는 해당 디렉토리에서 상응하는 압축된 파일에 대한 요청으로 재지정됩니다 . 예를 들어 , 웹 서버가 `myfile.html` 이나 `myfile.html` 및 `myfile.html.gz` 둘 다에 대한 요청을 수신하면 적절한 `Accept-encoding` 헤더를 가진 그러한 요청은 압축된 파일을 수신합니다 .

미리 압축된 내용을 서비스하도록 서버를 구성하려면 다음 단계를 수행하십시오 .

1. Class Manager 에서 Content Management 탭을 누릅니다 .
2. Serve Precompressed Content 를 누릅니다 .
3. 다음 정보를 입력합니다 .

- **Editing.** 드롭다운 목록에서 미리 압축된 내용을 서비스할 리소스를 선택합니다 . 디렉토리를 선택하면 서버가 해당 디렉토리 또는 디렉토리의 파일용 URL 을 수신하는 경우에만 서버가 미리 압축된 내용을 서비스합니다 .

  Browse 버튼을 눌러 기본 문서 디렉토리를 찾아보거나 Wildcard 버튼을 눌러 와일드카드 패턴을 지정합니다 . 와일드카드 패턴을 사용하는 자세한 방법은 Resource Picker 에서 와일드카드 사용를 참조하십시오 .

- **Activate Serving Precompressed Content?** 서버가 선택된 리소스에 대하여 미리 압축된 내용을 서비스하도록 지시합니다 .

- **Check Age.** 압축된 버전이 압축되지 않은 버전보다 오래되었는지 확인하도록 지정합니다 . yes 또는 no 를 선택할 수 있습니다 .

  yes로 설정하는 경우 압축된 버전이 압축되지 않은 버전보다 오래된 경우 서비스되지 않습니다 .

  no 로 설정하는 경우 압축된 버전이 압축되지 않은 버전보다 오래된 경우라도 항상 선택됩니다 .

  기본값은 yes 로 설정됩니다 .

- **Vary Header.** `Vary: Accept-encoding` 헤더를 삽입할지 지정합니다 . yes 또는 no 를 선택합니다 .

  yes 로 설정하면 파일의 압축된 버전이 선택될 경우 항상 `Vary: Accept-encoding` 헤더가 삽입됩니다 .

  no 로 설정되면 `Vary: Accept-encoding` 헤더가 삽입되지 않습니다 .

  기본값은 yes 로 설정됩니다 .

4. OK 를 누릅니다 .

필요시 내용 압축으로 서버 구성

전송 중에 전송 데이터를 압축하도록 Sun ONE Web Server 6.1 를 구성할 수도 있습니다 . 동적으로 생성된 HTML 페이지는 사용자가 요청할 때까지는 존재하지 않습니다 . 이것은 전자 상거래 웹 응용 프로그램과 데이터베이스 기반 사이트에 특히 유용합니다 .

필요시 내용을 압축하도록 서버를 구성하려면 다음 단계를 수행하십시오 .

1. Class Manager 에서 Content Management 탭을 누릅니다 .
2. Compress Content on Demand 를 누릅니다 .
3. 다음 정보를 입력합니다 .

- **Editing.** 드롭다운 목록에서 필요시 동적으로 압축된 내용을 서비스할 리소스를 선택합니다 . 디렉토리를 선택하면 서버가 해당 디렉토리 또는 디렉토리의 파일용 URL 을 수신하는 경우에만 서버가 압축된 내용을 서비스합니다 .

  Browse 버튼을 눌러 기본 문서 디렉토리를 찾아보거나 Wildcard 버튼을 눌러 와일드카드 패턴을 지정합니다 . 와일드카드 패턴을 사용하는 자세한 방법은 Resource Picker 에서 와일드카드 사용를 참조하십시오 .

- **Activate Compress Content on Demand?** 서버가 선택된 리소스에 대해 미리 압축된 내용을 서비스해야 하는지 선택합니다 .

- **Vary Header.** `Vary: Accept-encoding` 헤더를 삽입할지 지정합니다 . yes 또는 no 를 선택합니다 .

  `yes` 로 설정하면 파일의 압축된 버전이 선택될 경우 항상 `Vary: Accept-encoding` 헤더가 삽입됩니다 .

  `no` 로 설정되면 `Vary: Accept-encoding` 헤더가 삽입되지 않습니다 .

  기본값은 `yes` 로 설정됩니다 .

- **Fragment Size.** 압축 라이브러리(zlib)가 한 번에 압축할 양을 제어하는데 사용하는 메모리 조각의 크기를 바이트 단위로 지정합니다 . 기본값은 `8096` 입니다 .

- **Compression Level.** 압축의 수준을 지정합니다 . 1 ~ 9 사이 값을 선택합니다 . 값 1 은 속도가 최고이고 9 는 압축율이 최고입니다 . 기본값은 6 으로 속도와 압축율이 조화된 값입니다 .

4. OK 를 누릅니다 .

obj.conf 의 압축 관련 변경 사항

서버에서 압축이 사용되면 입력 항목에 `obj.conf` 파일이 추가됩니다 . 다음은 예제 입력 항목입니다 .

```
Output fn=?nsert-filter?filter=?ttp-compression?type=?ext/*_
```

특정 유형의 문서로만 압축을 제한하거나 압축된 내용을 제대로 처리하지 못하는 브라우저를 제외하려면 `obj.conf` 파일을 편집해야 합니다 . 이 작업을 완수하기 위해 해야 할 바에 대한 더 자세한 내용은 Sun ONE Web Server 6.1 *NSAPI Programmer's Guide* 를 참조하십시오 .

17 장

구성 스타일 적용

구성 스타일을 사용하면 다양한 가상 서버가 유지보수하는 특정한 파일 또는 디렉토리에 일련의 옵션을 쉽게 적용할 수 있습니다 . 예를 들어 , 액세스 로깅을 설정하는 구성 스타일을 만들 수 있습니다 . 구성 스타일을 로깅하려는 파일 및 디렉토리에 적용하면 가상 서버의 모든 파일 및 디렉토리에 대한 액세스 로깅을 개별적으로 구성할 필요가 없습니다 .

이 장의 내용 :

구성 스타일 만들기

구성 스타일을 만들려면 다음 단계를 따르십시오 .

1. Class Manager 에 액세스합니다 .
2. Styles 탭을 선택합니다 .
3. New Style 링크를 누릅니다 .
4. 구성 스타일에 부여할 이름을 입력합니다 . OK 를 누릅니다 .

   Sun ONE Web Server 가 Edit a Style 페이지를 표시됩니다 .
5. 드롭다운 목록에서 편집할 구성 스타일을 선택하고 Edit this Style 을 누릅니다 .

6. 사용 가능 링크 목록에서 스타일에 대해 구성하려는 카테고리를 누릅니다 .

   표의 목록에 표시된 정보를 구성할 수 있습니다 .

7. 나타나는 양식을 입력하고 OK 를 누릅니다 .

8. 4, 5 단계를 반복하여 구성 스타일에 다른 구성 변경을 합니다 . OK 를 누릅니다 .

편집할 스타일을 선택하면 Resource Picker 가 다른 리소스 대신 구성 스타일을 나열합니다 . 스타일 편집을 완료한 후 OK, Save, Apply 를 누릅니다 . Resource Picker 가 스타일 모드를 종료합니다 . Resource Picker 에서 Exit styles mode 를 선택하여 스타일 모드를 종료할 수도 있습니다 . Resource Picker 에 대한 더 자세한 내용은 제 1 장 , "Sun ONE Web Server 소개 " 의 "Resource Picker 사용 " 페이지 41 를 참조하십시오 .

표 17-1) 구성 스타일 카테고리

카테고리	설명
CGI 파일 유형	CGI 를 파일 유형으로 사용할 수 있습니다 . CGI 에 대한 더 자세한 내용은 제 15 장 , " 프로그램으로 서버 확장 " 의 페이지 349 의 "CGI 프로그램 설치 ," 를 참조하십시오 .
문자 세트	리소스용 문자 세트를 변경할 수 있습니다 . 문자 세트에 대한 더 자세한 내용은 제 16 장 , " 내용 관리 " 의 페이지 370 의 " 문자 세트 변경 ," 을 참조하십시오 .
기본 쿼리 처리기	서버 리소스용 기본 쿼리 처리기를 설정할 수 있습니다 . 쿼리 처리에 대한 더 자세한 내용은 제 15 장 , " 프로그램으로 서버 확장 " 의 페이지 359 의 " 쿼리 처리기 사용 ," 를 참조하십시오 .
문서 꼬리말	서버 리소스에 문서 꼬리말을 추가할 수 있습니다 . 더 자세한 내용은 제 16 장 , " 내용 관리 " 의 " 문서 꼬리말 설정 " 페이지 372 를 참조하십시오 .
.htaccess 구성	사용자에게 Server Manager 에 대한 액세스 권한 없이 구성 옵션의 일부분을 제공할 수 있습니다 . 액세스 제어에 대한 더 자세한 내용은 제 9 장 , " 서버 액세스 제어 " 를 참조하십시오 .
고급 보안 필요	키 크기 제한을 지정하거나 특정 파일에 대한 액세스를 거부할 수 있습니다 .
오류 응답	서버에서 오류가 발생하는 경우 클라이언트에게 표시하는 오류 응답을 사용자 정의할 수 있습니다 .
로그 기본설정	액세스 로그용 기본설정을 설정할 수 있습니다 . 로그 기본설정에 대한 더 자세한 내용은 제 10 장 , " 로그 파일 사용 " 의 페이지 238 의 " 액세스 로그 기본 설정 ," 을 참조하십시오 .

표 17-1) 구성 스타일 카테고리 *(계속)*

카테고리	설명
원격 파일 조작	원격 브라우저가 서버의 문서를 변경할 수 있도록 파일 조작 명령어를 사용할 수 있습니다 . 더 자세한 내용은 제 16 장 , " 내용 관리 " 의 " 원격 파일 조작 사용 " 페이지 366 를 참조하십시오 .
서버가 파싱한 HTML	서버가 클라이언트에게 파일을 보내기 전에 해당 파일을 파싱할 것인지 지정할 수 있습니다 . 자세한 내용은 Sun ONE Web Server 6.1 *Programmer's Guide* 를 참조하십시오 .
미리 압축된 내용 서비스	서버가 파일의 미리 압축된 버전을 보내게 할 것인지 지정할 수 있습니다 . 더 자세한 내용은 제 16 장 , " 내용 관리 " 의 " 서버를 미리 압축된 내용을 서비스하도록 구성 " 페이지 376 를 참조하십시오 .
필요시 내용 압축	서버가 클라이언트에게 파일을 보내기 전에 해당 파일을 동적으로 압축할 것인지 지정할 수 있습니다 . 더 자세한 내용은 제 16 장 , " 내용 관리 " 의 " 필요시 내용 압축으로 서버 구성 " 페이지 377 를 참조하십시오 .
심볼 링크 (UNIX/Linux)	서버의 파일시스템 링크 사용을 제한할 수 있습니다 . 더 자세한 내용은 제 16 장 , " 내용 관리 " 의 " 심볼 링크 (UNIX/Linux) 제한 " 페이지 373 를 참조하십시오 .

자세한 내용은 온라인 도움말의 New Style page 를 참조하십시오 .

구성 스타일 지정

구성 스타일을 만들었으면 가상 서버의 파일이나 디렉토리에 지정할 수 있습니다 . 개별 파일 및 디렉토리를 지정하거나 와일드카드 패턴 (예를 들어 , `*.gif`) 을 지정할 수 있습니다 .

구성 스타일을 지정하려면 다음과 단계를 따르십시오 .

1. Class Manager 에 액세스합니다 .
2. Styles 탭을 선택합니다 .
3. Assign Style 링크를 누릅니다 .
4. 이 구성 스타일을 적용할 URL 의 접두사를 입력합니다 .

   문서 루트 내부 디렉토리를 선택하면 문서 루트 뒤의 경로만 입력합니다 . 디렉토리 뒤에 `/*` 를 입력하면 구성 스타일이 디렉토리의 모든 내용에 적용됩니다 .

5. 적용하려는 구성 스타일을 선택합니다 .

   리소스에 이전에 적용된 구성 스타일을 제거하려면 None 구성 스타일을 적용합니다 . OK 를 누릅니다 .

자세한 내용은 온라인 도움말의 Assign a Style 페이지를 참조하십시오 .

구성 스타일 지정 목록

구성 스타일을 만들고 파일 또는 디렉토리에 적용한 후에 구성 스타일의 목록과 그것을 적용한 위치를 얻을 수 있습니다 .

구성 스타일 지정을 나열하려면 다음 단계를 따르십시오 .

1. Class Manager 에 액세스합니다 .
2. Styles 탭을 선택합니다 .
3. List Assignments 링크를 누릅니다 .

   Sun ONE Web Server 가 List Assignments 페이지를 표시하여 서버 리소스에 적용한 구성 스타일을 나타냅니다 .

4. 구성 스타일 지정을 편집하려면 구성 스타일 이름 옆의 Edit 링크를 누릅니다 .

자세한 내용은 온라인 도움말의 List Assignments 페이지를 참조하십시오 .

구성 스타일 편집

구성 스타일을 편집하려면 다음 단계를 따르십시오 .

1. Class Manager 에 액세스합니다 .
2. Styles 탭을 선택합니다 .
3. Edit Style 링크를 누릅니다 .
4. 편집하려는 구성 스타일을 선택하고 "Edit this style" 버튼을 누릅니다 .
5. 사용 가능 링크 목록에서 스타일에 대해 구성하려는 카테고리를 누릅니다 .

   이러한 카테고리에 대한 자세한 내용은 " 구성 스타일 만들기 " 페이지 379 부분을 참조하십시오 .

6. 나타나는 양식을 입력한 다음 OK 를 누릅니다 .

7. 4, 5 단계를 반복하여 구성 스타일에 다른 변경을 합니다 . OK 를 누릅니다 .

편집할 스타일을 선택하면 Resource Picker 가 다른 리소스 대신 구성 스타일을 나열합니다 . 스타일 편집을 완료한 후 OK, Save, Apply 를 누릅니다 . Resource Picker 가 스타일 모드를 종료합니다 . Resource Picker 에서 Exit styles mode 를 선택하여 스타일 모드를 종료할 수도 있습니다 . Resource Picker 에 대한 더 자세한 내용은 제 1 장 , "Sun ONE Web Server 소개 " 의 "Resource Picker 사용 " 페이지 41 를 참조하십시오 .

자세한 내용은 온라인 도움말의 Edit Style 페이지를 참조하십시오 .

구성 스타일 제거

구성 스타일을 제거하기 전에 구성 스타일이 적용된 지정을 제거합니다 . 구성 스타일을 제거하기 전에 이 작업을 수행하지 않으면 가상 서버 클래스의 `obj.conf` 파일을 직접 편집하여 파일에서 구성 스타일을 검색한 다음 그것을 `None` 으로 교체해야 합니다 . 이 검색 및 교체를 수행하지 않으면 적용된 구성 스타일이 삭제된 파일 또는 디렉토리에 누군가 액세스할 경우 서버 구성 오류 메시지가 나타납니다 .

구성 스타일을 제거하려면 다음 단계를 따르십시오 .

1. Class Manager 에 액세스합니다 .
2. Styles 탭을 선택합니다 .
3. List Assignments 링크를 누릅니다 .
4. 제거하려는 Edit Style Assignment 를 선택합니다 .
5. Remove this Assignment 를 누릅니다 .

자세한 내용은 온라인 도움말의 Remove Style 페이지를 참조하십시오 .

18 장

검색 사용

Sun ONE Web Server 6.1 은 사용자가 서버의 문서를 검색하고 웹 페이지에 결과를 표시하도록 하는 검색 기능을 포함합니다 . 서버 관리자는 사용자가 검색할 문서 (컬렉션) 의 색인을 만들고 검색 인터페이스를 사용자 정의하여 사용자의 요구에 맞출 수 있습니다 .

이 장의 내용 :

Search 정보

검색 기능은 Sun ONE Web Server 의 설치 동안 다른 웹 구성 요소로 설치됩니다 . 검색은 Sun ONE Web Server 6.0 에서와 마찬가지로 서버 인스턴스 수준이 아닌 가상 서버 수준에서 구성 및 관리됩니다 .

Virtual Server Manager 의 Search 탭은 각 가상 서버에 대한 검색을 구성하는 데 사용됩니다 . 이 탭에서 할 수 있는 작업은 다음과 같습니다 .

- 검색 기능을 사용함 및 사용 안함으로 설정
- 검색 컬렉션 만들기 , 수정 , 삭제 및 다시 색인
- 검색 컬렉션에 대해 스케줄된 유지보수 작업 만들기 , 수정 및 제거

관리 인터페이스에서 얻은 정보는 *<server-root>*`/config/server.xml` 파일에 저장되고 이 파일에서 해당 정보는 VS 요소 내부에서 매핑됩니다 .

서버 관리자는 검색 쿼리 및 검색 결과 페이지를 사용자 정의할 수 있습니다 . 여기에는 페이지에 기업 로고로 다시 브랜드 표시 또는 검색 결과가 나타나는 방법 변경이 포함될 수 있습니다 . 이전 릴리스에서는 패턴 파일 사용으로 이 작업을 수행했습니다 . 패턴 파일은 Sun ONE Web Server 6.1 에서 지원되지 않습니다 . 대신 , 사용자 정의는 이제 제품과 함께 포함되는 JSP 태그 라이브러리 세트를 사용하여 수행됩니다 . 이러한 라이브러리는 패턴 파일이 제공하는 것과 유사한 기능을 제공합니다 . 검색 인터페이스 사용자 정의에 대한 더 자세한 내용은 검색 페이지 사용자 정의를 참조하십시오 .

이전 릴리스에 있었던 , 검색에 대한 전역적 "on" 또는 "off" 기능은 없습니다 . 대신 , 기본 검색 웹 응용 프로그램이 제공되고 특정 가상 서버에서 사용함 또는 사용 안함으로 설정됩니다 . 이 검색 응용 프로그램은 컬렉션 쿼리 및 결과 보기에 사용되는 기본 웹 페이지를 제공합니다 . 검색 응용 프로그램에는 검색 태그 라이브러리를 사용하여 사용자 정의 검색 인터페이스를 구축하는 방법을 나타내는 예제 JSP 가 포함됩니다 .

주의 Sun ONE Web Server 6.0 과 달리 버전 6.1 은 검색 결과에 대한 액세스 확인을 제공하지 않습니다 . 잠재적인 보안 모델 및 영역의 수로 인해 검색 응용 프로그램 내부에서 보안 확인을 수행하고 결과를 필터링하는 것이 불가능합니다 . 적절한 보안 메커니즘을 갖추어 컨텐트를 보호하는 일은 서버 관리자의 책임입니다 .

Sun ONE Web Server 6.1 은 복수 문서 검색을 지원합니다 . 여러 가지 형식 (예를 들어 , HTML, ASCII, 및 PDF) 을 가진 문서가 색인되고 검색됩니다 .

참고 Sun ONE Web Server 6.1 은 Linux 플랫폼에서는 여러 문서 형식에 대한 검색을 지원하지 않습니다 .

이전 릴리스에서 사용된 검색 엔진은 Sun ONE Web Server 6.1 에서 새로운 검색 엔진으로 교체되었습니다 . 따라서 이전 릴리스의 웹 서버를 Sun ONE Web Server 6.1 로 이전하려면 기존 검색 컬렉션과 인덱스는 이전되지 않습니다 .

가상 서버에 대한 검색 응용 프로그램 사용 설정

검색은 Sun ONE Web Server 에 포함된 검색 응용 프로그램을 사용함으로 설정하여 가상 서버에 대해 사용 설정됩니다 . 관리 인터페이스는 검색을 사용함으로 설정하는 데 사용됩니다 .

참고 Java 웹 컨테이너는 검색이 실행되려면 반드시 사용함으로 설정되어야 합니다 .

Java 가 구성하려는 가상 서버를 포함하는 가상 서버 클래스에 대해 사용함으로 설정되도록 한 후 다음 단계를 수행하여 검색을 설정합니다 .

1. 검색을 사용함으로 설정할 가상 서버를 선택하고 Manage 버튼을 누릅니다 .
2. Search 탭을 누른 다음 Search Configuration 링크를 누릅니다 .
3. 다음 정보를 입력합니다 .
   - **Max Hits.** 검색 쿼리에서 검색되는 최대 결과를 지정합니다 .
   - **URI.** 사용자 정의 검색 응용 프로그램을 사용하려면 URI 를 입력합니다 . 기본 검색 응용 프로그램을 사용하려면 여기에 값을 지정할 필요가 없습니다 .
   - **Path.** 사용자 정의 검색 응용 프로그램을 사용하려면 경로를 입력합니다 . 기본 검색 응용 프로그램을 사용하는 경우 여기에 값을 입력할 필요가 없습니다 .
   - **Enabled.** 이 상자를 선택하여 기본 검색 응용 프로그램을 사용하도록 할 수 있습니다 .
4. OK 를 누릅니다 .

가상 서버에 대해 검색 응용 프로그램을 사용하지 않도록 설정

Sun ONE Web Server 와 함께 포함된 검색 응용 프로그램을 사용하지 않도록 하여 가상 서버에 대해 검색을 사용하지 않도록 설정합니다 . 관리 인터페이스는 검색을 사용하지 않도록 설정하는 데 사용합니다 .

가상 서버에 대해 검색을 사용하지 않도록 하려면 다음 단계를 따르십시오 .

1. 검색을 사용하지 않도록 할 가상 서버를 선택하고 Manage 버튼을 누릅니다 .
2. Search 탭을 누른 다음 Search Configuration 링크를 누릅니다 .
3. Enabled 확인란을 선택 해제합니다 .
4. OK 를 누릅니다 .

Search 컬렉션 정보

검색을 하려면 사용자를 검색할 검색 가능 데이터의 데이터베이스가 필요합니다 . 서버 관리자가 이 데이터베이스를 만듭니다 . 이것은 컬렉션이라고 하며 서버의 문서에 대한 정보를 색인하고 저장합니다 . 서버 관리자가 서버 문서의 전부 또는 일부를 색인하면 제목 , 작성일 , 저자와 같은 정보를 검색할 수 있습니다 .

컬렉션에 대한 다음 사항을 주의하십시오 .

- 컬렉션은 관리되고 있는 가상 서버에 특정적입니다 .
- 가상 서버에서 보이는 문서만 관리 인터페이스에 제시되고 색인 가능합니다 .
- 서버에 존재할 수 있는 컬렉션 수에는 제한이 없습니다 .
- 단일 컬렉션은 파일 시스템의 상위 디렉토리 아래 위치하는 파일만 포함할 수 있습니다
- 여러 가지 형식의 문서 (예를 들어 , HTML, ASCII 및 PDF) 를 색인 및 검색할 수 있습니다 .
- 검색 컬렉션의 문서는 문자 부호화에 있어서 특정적이지 않습니다 . 즉 , 검색 컬렉션은 여러 부호화에 연결될 수 있습니다 .
- 컬렉션에 대한 정보는 `server.xml` 의 VS 요소에 저장됩니다 .

이 부분에서는 다음 항목에 대해 설명합니다 .

컬렉션 만들기

컬렉션은 관리 인터페이스에서 만들고 관리합니다 . 색인활 문서를 지정하여 새로운 컬렉션을 만듭니다 .

새로운 컬렉션을 만들려면 다음과 같이 합니다 .

1. 컬렉션을 만들려는 가상 서버를 선택하고 Manage 버튼을 누릅니다 .
2. Search 탭을 선택한 다음 Create Collection 링크를 누릅니다 .
3. 다음 정보를 입력합니다 .
    - **Directory to Index.** 드롭다운 목록에서 문서가 컬렉션에 의해 색인될 디렉토리를 선택합니다 . 이 가상 서버에서 보이는 디렉토리만 나열됩니다 .

      디렉토리의 내용을 보려면 View 를 누릅니다 . 선택한 디렉토리가 하위 디렉토리일 경우 이들은 "*View directory_name*" 페이지에 나열됩니다 . 색인할 디렉토리를 선택하려면 index 를 누릅니다 . 디렉토리를 보려면 해당 폴더를 누릅니다 .

      색인 가능 디렉토리의 목록에 디렉토리를 추가하려면 먼저 추가 문서 디렉토리를 만들어야 합니다 . 더 자세한 내용은 추가 문서 디렉토리 설정을 참조하십시오 .
    - **Collection Name.** 컬렉션의 이름을 입력합니다 .
    - **Display Name.** (선택) 이것은 검색 쿼리 페이지의 컬렉션 이름으로 나타납니다 . 표시 이름을 지정하지 않으면 컬렉션 이름이 표시 이름으로 사용됩니다 .
    - **Description.** (선택) 새 컬렉션을 설명하는 텍스트를 입력합니다 .

- **Include Subdirectories?** No 를 선택하면 선택한 디렉토리의 하위 디렉토리 내부 문서는 색인되지 않습니다 . 기본값은 Yes 입니다 .
- **Pattern.** 와일드카드를 선택하여 색인할 파일을 선택합니다 . 와일드카드에 대한 더 자세한 내용은 Resource Picker 에서 와일드카드 사용를 참조하십시오 .

주의	와일드카드 패턴을 적절하게 사용하여 특정한 파일만 색인되도록 합니다 . 예를 들어 *.* 을 지정하면 실행 파일과 perl 스크립트도 색인될 수 있습니다 .

- **Default Encoding.** 문서가 색인될 문자 부호화를 지정합니다 . 기본값은 "ISO-8859-1" 입니다 . 색인 엔진은 내장 메타 태그에서 HTML 문서의 부호화를 결정하려 합니다 . 이것이 지정되지 않으면 기본 부호화가 사용됩니다 .

  컬렉션의 문서는 단일 언어 / 부호화에 제한되지 않습니다 . 문서를 추가할 때마다 단일 부호화만 지정됩니다 . 그러나 다음에 컬렉션에 문서를 추가할 때 다른 기본 부호화를 선택할 수 있습니다 .

4. OK 를 누릅니다 .

그러면 다음 위치에 지정된 이름으로 새로운 컬렉션이 만들어집니다 .

<*instance-root*>/collections/<*vs-id*>/<*collection-name*>

또한 server.xml 파일에 SEARCHCOLLECTION 항목도 만들어집니다 .

컬렉션 구성

컬렉션이 만들어진 후 설정의 일부를 수정할 수 있습니다 . 이 설정은 server.xml 파일에 저장됩니다 . 컬렉션을 다시 구성하면 server.xml 파일이 업데이트되어 변경사항을 반영합니다 .

컬렉션 설정에 대해 불필요한 변경은 하지 말아야 합니다 .

기존 컬렉션을 편집하려면 다음 작업을 수행합니다 .

1. 구성하려는 컬렉션을 포함하는 가상 서버를 선택하고 Manage 버튼을 누릅니다 .
2. Search 탭을 누른 다음 Configure Collection 링크를 누릅니다 .
3. Collection 드롭다운 목록에서 구성하려는 컬렉션을 선택하고 Go 를 누릅니다 .
4. 선택한 컬렉션에 대하여 다음의 정보를 편집할 수 있습니다 .

- **Display name.** (선택) 이것은 검색 쿼리 페이지의 새 컬렉션 이름으로 나타납니다 .
- **Description.** (선택) 컬렉션의 텍스트 설명을 편집합니다 .
- **Document URI.** 검색 컬렉션에 대한 문서의 URI 를 편집합니다 .

참고 Additional Document Directories 페이지에서 문서 루트에 대한 URI 매핑을 변경하지 않았을 경우에는 Document URI 를 변경하지 마십시오 . 더 자세한 내용은 추가 문서 디렉토리 설정를 참조하십시오 .

- **Enabled.** Yes를 선택하여 사용하도록 설정합니다. No를 선택하면 컬렉션이 검색 쿼리 페이지에 나타나지 않습니다 .

5. OK 를 누릅니다 .

이것은 컬렉션을 재구성하고 `server.xml` 파일의 해당하는 `SEARCHCOLLECTION` 항목을 수정합니다 .

컬렉션 업데이트

컬렉션이 만들어진 후 파일을 추가 또는 제거할 수 있습니다 . 문서는 컬렉션을 만드는 동안 지정된 디렉토리 아래에만 추가될 수 있습니다 . 문서를 제거하는 경우 파일의 항목과 해당 메타데이터만 컬렉션에서 제거됩니다 . 실제 파일 자체는 파일 시스템에서 제거되지 않습니다 .

컬렉션을 업데이트하려면 다음 단계를 수행합니다 .

1. 업데이트하려는 컬렉션을 포함하는 가상 서버를 선택하고 Manage 버튼을 누릅니다 .
2. Search 탭을 선택한 다음 Update Collection 링크를 누릅니다 .
3. Collection 드롭다운 목록에서 업데이트하려는 컬렉션을 선택합니다 .
4. Docs
5. 선택한 컬렉션에 대하여 다음의 정보를 업데이트할 수 있습니다 .
   - **Include subdirectories?** No 를 선택하면 선택한 디렉토리의 하위 디렉토리 내부 문서는 색인되지 않습니다 . 기본값은 Yes 입니다 .

참고 **Include Subdirectories?** 는 문서 추가시에만 유효합니다 .

- **Pattern.** 와일드카드를 지정하여 색인할 파일을 선택하고 컬렉션에서 제거합니다 . 와일드카드에 대한 더 자세한 내용은 Resource Picker 에서 와일드카드 사용를 참조하십시오 .

주의 문서를 추가하는 동안 와일드카드 패턴을 적절하게 사용하여 특정한 파일만 색인되도록 합니다 . 예를 들어 *.* 을 지정하면 실행 파일과 perl 스크립트도 색인될 수 있습니다 .

- **Default Encoding.** 문서가 색인될 문자 부호화를 지정합니다 . 기본값은 "ISO-8859-1" 입니다 . 색인 엔진은 내장 메타 태그에서 HTML 문서의 부호화를 결정하려 합니다 . 이것이 지정되지 않으면 기본 부호화가 사용됩니다 .

  컬렉션의 문서는 단일 언어 / 부호화에 제한되지 않습니다 . 문서를 추가할 때마다 단일 부호화만 지정됩니다 . 그러나 다음에 컬렉션에 문서를 추가할 때 다른 기본 부호화를 선택할 수 있습니다 .

6. Add Documents 를 눌러 색인에 문서를 추가하거나 Remove Documents 를 눌러 해당하는 색인 항목을 제거합니다 .

참고 문서는 컬렉션을 만들 때 지정한 디렉토리에 있을 경우에만 추가할 수 있습니다 .

컬렉션 제거

컬렉션은 만들어진 후에 제거할 수 있습니다 . 컬렉셔이 삭제되면 검색 쿼리 페이지에서 사용자에게 더 이상 보이지 않고 컬렉션에 연결된 모든 구성 및 색인 파일은 삭제됩니다 . 컬렉션을 구성하는 실제 문서는 파일 시스템에서 삭제되지 않고 컬렉션의 색인 항목만 삭제됩니다 .

컬렉션을 제거하려면 다음 단계를 수행합니다 .

1. 제거하려는 컬렉션을 포함하는 가상 서버를 선택하고 Manage 버튼을 누릅니다 .
2. Search 탭을 선택한 다음 Maintain Collection 링크를 누릅니다 .
3. Collection 드롭다운 목록에서 제거하려는 컬렉션을 선택합니다 .
4. Remove Collection 버튼을 누릅니다 .

참고 컬렉션이 제거되면 컬렉션에 대해 스케줄된 유지보수도 제거됩니다. 스케줄된 유지보수에 대한 더 자세한 내용은 스케줄된 컬렉션 유지보수 추가를 참조하십시오.

참고 컬렉션을 제거하는 데 로컬 파일 관리자를 사용하지 마십시오. 그렇게 하면 해당하는 구성 파일이 업데이트되지 않습니다.

컬렉션 유지보수

주기적으로 컬렉션을 유지보수할 수 있습니다. 컬렉션을 자주 색인하고 업데이트하지 않는 한 이러한 작업이 필요하지 않을 수 있습니다. 다음 작업을 할 수 있습니다.

- 컬렉션 다시 색인
- 컬렉션 업데이트

컬렉션 다시 색인

컬렉션을 만든 후 다시 색인할 수 있습니다. 컬렉션을 만든 후 문서를 수정하면 컬렉션이 다시 색인됩니다. 컬렉션을 다시 색인하면 컬렉션의 새 내용은 색인되지 않지만 컬렉션의 기존 내용은 업데이트됩니다. 색인 항목이 서버 파일 시스템에 더 이상 존재하지 않는 문서에 대해 존재하면 해당 항목은 삭제됩니다.

컬렉션을 다시 색인하려면 다음 단계를 수행합니다.

1. 다시 색인하려는 컬렉션을 포함하는 가상 서버를 선택하고 Manage 버튼을 누릅니다.
2. Search 탭을 선택한 다음 Maintain Collection 링크를 누릅니다.
3. Collection 드롭다운 목록에서 다시 색인하려는 컬렉션을 선택합니다.
4. Reindex 버튼을 누릅니다.

스케줄된 컬렉션 유지보수 추가

컬렉션에 대해 정기적으로 수행되도록 유지보수 작업을 스케줄할 수 있습니다 . 스케줄할 수 있는 작업은 다시 색인과 업데이트입니다 . 관리 인터페이스는 특정 컬렉션에 대한 작업을 스케줄하는 데 사용됩니다 . 다음 작업을 지정할 수 있습니다 .

- 수행할 작업 (재색인 또는 업데이트)
- 작업을 수행할 시간
- 작업을 수행할 날짜

컬렉션의 정기적인 유지보수를 추가하려면 다음 단계를 수행하십시오 .

1. 유지보수를 스케줄하려는 컬렉션을 선택하고 Add Scheduled Maintenance 링크를 누릅니다 .
2. 다음 정보를 입력합니다 .

   ❍ **Task.** 자동화하려는 작업을 선택합니다 . 선택 사항은 다시 색인과 업데이트입니다 .

   Update 를 선택하면 다음 정보를 입력해야 합니다

   - **Recurse Subdirectories?** No 를 선택하면 선택한 디렉토리의 하위 디렉토리 내부 문서는 색인되지 않습니다 . 기본값은 Yes 입니다 .

   ❍ **Pattern.** 와일드카드를 선택하여 색인할 파일을 선택합니다 . 와일드카드에 대한 더 자세한 내용은 Resource Picker 에서 와일드카드 사용을 참조하십시오 .

주의 와일드카드 패턴을 적절하게 사용하여 특정한 파일만 색인되도록 합니다 . 예를 들어 *.* 을 지정하면 실행 파일과 perl 스크립트도 색인될 수 있습니다 .

   ❍ **Default Encoding.** 문서가 색인될 문자 부호화를 지정합니다 . 기본값은 "ISO-8859-1" 입니다 . 색인 엔진은 내장 메타 태그에서 HTML 문서의 부호화를 결정하려 합니다 . 이것이 지정되지 않으면 기본 부호화가 사용됩니다 .

   컬렉션의 문서는 단일 언어 / 부호화에 제한되지 않습니다 . 문서를 추가할 때마다 단일 부호화만 지정됩니다 . 그러나 다음에 컬렉션에 문서를 추가할 때 다른 기본 부호화를 선택할 수 있습니다 .

   ❍ **Scheduled Time.** (필수) HH:MM 형식으로 스케줄된 유지보수를 실행하려는 시간을 지정합니다 . 예를 들어 , 컬렉션의 문서가 수정되었을 가능성이 높은 날의 끝에 스케줄된 유지보수를 실행할 수 있습니다 .

- **Day(s) of week.** (필수) 하나 이상의 확인란을 선택하여 스케줄된 유지보수가 실행될 날짜를 지정합니다 .

3. OK 를 누릅니다 .

참고 UNIX/Linux 사용자는 스케줄된 유지보수를 추가한 후 변경 사항을 적용하기 위해 크론 제어 프로세스를 다시 시작해야 합니다 .

스케줄된 컬렉션 유지보수 편집

요구 사항이 변하면 컬렉션의 스케줄된 유지보수의 등록정보를 변경할 수 있습니다 . 예를 들어 , 사이트가 업데이트될 가능성이 가장 높은 시간을 염두에 두고 유지보수를 다시 스케줄하기로 할 수 있습니다 .

컬렉션에 대한 스케줄된 유지보수를 변경하려면 다음 단계를 수행하십시오 .

1. Collection 드롭다운 목록에서 유지보수를 다시 스케줄하려는 컬렉션을 선택합니다 .
2. 재구성하려는 작업을 선택하고 필요 정보를 입력합니다 . 자세한 내용은 온라인 도움말의 Edit Scheduled Collection 페이지를 참조하십시오 .
3. OK 를 누릅니다 .

참고 컬렉션이 제거되면 컬렉션에 대해 스케줄된 유지보수도 제거됩니다 .

참고 UNIX/Linux 사용자는 스케줄된 유지보수를 재구성한 후 변경 사항을 적용하기 위해 크론 제어 프로세스를 다시 시작해야 합니다 .

스케줄된 컬렉션 유지보수 제거

더 이상 필요하지 않을 경우 컬렉션의 스케줄된 유지보수를 취소할 수 있습니다 .

스케줄된 유지보수를 취소하려면 다음 단계를 수행하십시오 .

1. Collection 드롭다운 목록에서 유지보수를 제거하려는 컬렉션을 선택합니다 .
2. 스케줄된 유지보수를 제거하려는 작업 , 즉 Reindex 또는 Update 중에서 선택합니다 . 작업이 스케줄되면 자세한 내용이 표시됩니다 .

3. Update 작업의 경우 제거하려는 작업 옆에 있는 Delete 확인란을 선택합니다 .
4. OK 를 누릅니다 .

참고 UNIX/Linux 사용자는 스케줄된 유지보수를 제거한 후 변경 사항을 적용하기 위해 크론 제어 프로세스를 다시 시작해야 합니다 .

검색 수행

사용자는 주로 검색 컬렉션에서 데이터 질문을 하는 것과 문서 목록을 반환받는 것에 관심을 가집니다 . Sun ONE Web Server 와 함께 설치된 검색 웹 응용 프로그램은 기본 검색 쿼리와 검색 결과 페이지를 제공합니다 . 이러한 페이지는 있는 그대로 사용하거나 검색 페이지 사용자 정의에 설명된 것과 같이 jsp 태그 세트를 사용하여 사용자 정의할 수 있습니다 .

사용자는 서버 관리자가 만든 컬렉션에 대해 검색합니다 . 사용자는 다음과 같이 할 수 있습니다 .

- 검색할 키워드 세트 및 선택적 쿼리 연산자를 입력합니다 .
- 가상 서버에 보이는 컬렉션만 검색합니다 .
- 단일 컬렉션에 대해 또는 가상 서버에 보이는 컬렉션 세트에 걸쳐 검색합니다 .

서버 관리자는 가상 서버에 대한 검색 쿼리 페이지에 액세스하는 데 필요한 URL 을 사용자에게 제공해야 합니다 .

주의 Sun ONE Web Server 6.0 과 달리 버전 6.1 은 검색 결과에 대한 액세스 확인을 제공하지 않습니다 . 잠재적인 보안 모델 및 영역의 수로 인해 검색 응용 프로그램 내부에서 보안 확인을 수행하고 결과를 필터링하는 것이 불가능합니다 . 적절한 보안 메커니즘을 갖추어 컨텐트를 보호하는 일은 서버 관리자의 책임입니다 .

Settings 페이지

최종 사용자가 검색 기능에 액세스하는 데 사용할 수 있는 기본 URL 은 다음과 같습니다 .

`http://<`*server-instance*`>:`*port number*`/search`

예 :

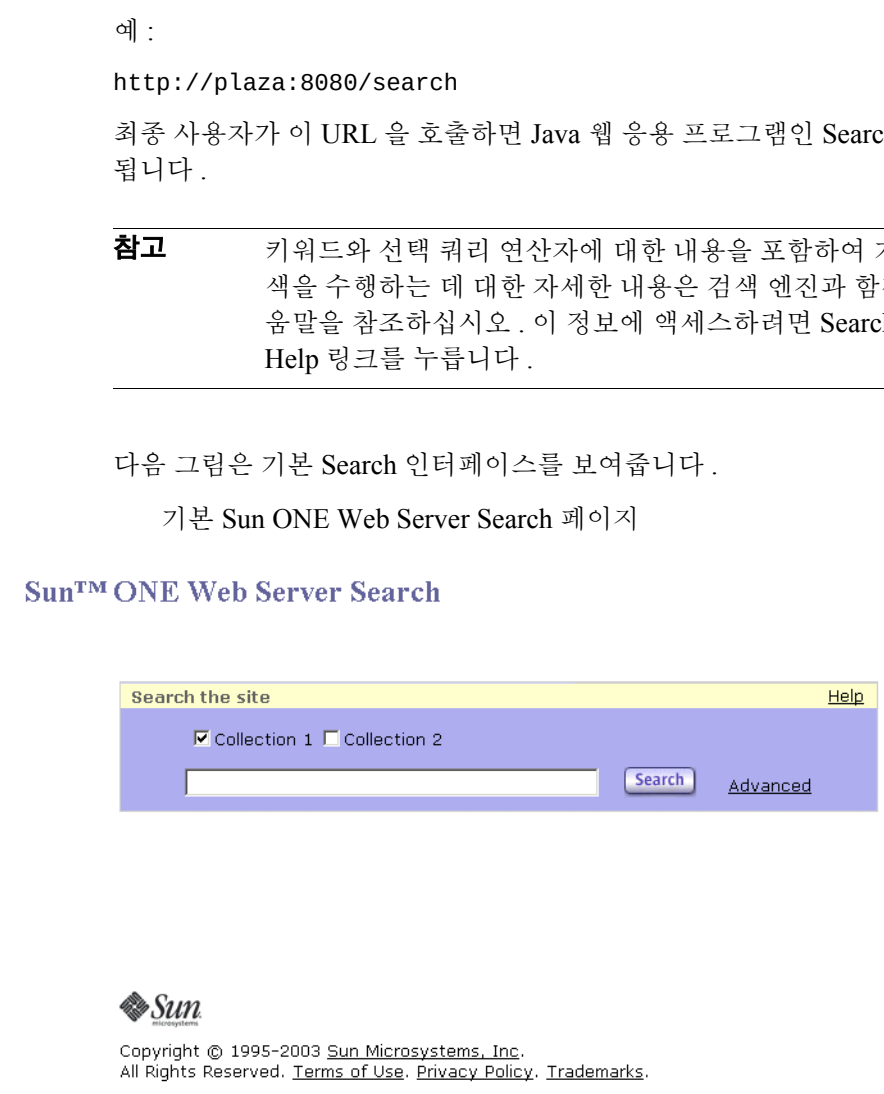
```
http://plaza:8080/search
```

최종 사용자가 이 URL 을 호출하면 Java 웹 응용 프로그램인 Search 페이지가 시작됩니다 .

참고 키워드와 선택 쿼리 연산자에 대한 내용을 포함하여 기본 및 고급 검색을 수행하는 데 대한 자세한 내용은 검색 엔진과 함께 제공되는 도움말을 참조하십시오 . 이 정보에 액세스하려면 Search 페이지에서 Help 링크를 누릅니다 .

다음 그림은 기본 Search 인터페이스를 보여줍니다 .

기본 Sun ONE Web Server Search 페이지



"Customizing Search Pages" 에 설명된 것과 같은 JSP 태그 세트를 사용하여 이 페이지를 사용자 정의할 수 있습니다 .

쿼리 만들기

검색 쿼리 페이지는 컬렉션에 대한 검색에 사용됩니다 . 사용자는 키워드 세트 및 선택적 쿼리 연산자를 입력한 다음 브라우저에 표시된 웹 페이지로 결과를 받습니다 . 결과 페이지는 검색 기준에 일치하는 서버 문서에 대한 링크를 포함합니다 .

참고 서버 관리자는 "Customizing Search Pages" 에 설명된 것과 같이 이 검색 쿼리 페이지를 사용자 정의할 수 있습니다 .

쿼리를 만들려면 다음과 같이 합니다 .

1. 브라우저의 Location 표시줄에 다음과 같은 형식으로 해당 URL 을 입력하여 Search 웹 응용 프로그램에 액세스합니다 .

   `http://<`*server-instance*`>:`*port number*`/search`

2. 나타나는 검색 쿼리 페이지에서 "Search in" 필드에 검색하려는 컬렉션을 나타내는 확인란을 선택합니다 .
3. 쿼리를 설명하는 단어를 몇 개 입력하고 관련 웹 페이지 목록을 보기 위해 'enter' 키를 누릅니다 (또는 Search 버튼을 누름).

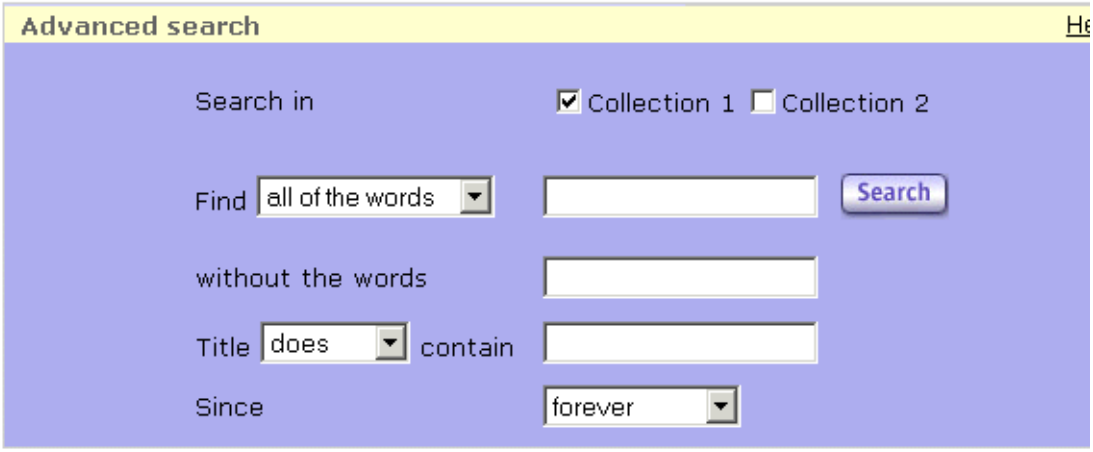
보다 미세 조정된 검색을 위해 다음 부분에 설명된 Advanced Search 페이지에 제공된 검색 매개변수를 사용할 수 있습니다 .

고급 검색

사용자는 키워드를 미세 조정하는 연산자를 추가하여 검색 정확도를 높일 수 있습니다 . 이러한 옵션은 Advanced Search 페이지에서 선택할 수 있습니다 .

다음 그림은 Advanced Search 페이지를 보여줍니다 .

Advanced Search 페이지



고급 검색 쿼리를 만들려면 다음 단계를 수행하십시오 .

1. 브라우저의 Location 표시줄에 다음과 같은 형식으로 해당 URL 을 입력하여 Search 웹 응용 프로그램에 액세스합니다 .

   `http://<`*server-instance*`>:`*port number*`/search`

2. Advanced 링크를 누릅니다 .
3. 다음 정보 중 일부 또는 전부를 입력합니다
   - **Search in.** 검색하려는 컬렉션을 선택합니다 .
   - **Find.** 4 개 옵션이 지원됩니다 .
     - **All of the words.** Find 에 지정된 모든 키워드를 포함하는 페이지를 찾습니다 .
     - **Any of the words.** Find 에 지정된 키워드 중 일부를 포함하는 페이지를 찾습니다 .
     - **The exact phrase.** Find 에 사용된 문구에 정확하게 일치하는 페이지를 찾습니다 .
     - **Passage search.** 검색된 페이지에서 키워드 또는 단어를 포함하는 부분을 강조합니다 .

- **Without the words.** 검색이 지정된 단어를 포함하는 웹 페이지를 제외합니다 .
- **Title "does/does not" contain.** 지정된 키워드를 포함하는 제목을 가진 페이지로 검색을 제한합니다 .
- **Since.** 선택한 기간에 색인된 웹 페이지로 검색 작업을 제한합니다 .

검색 결과 보기

검색 결과는 검색 기준과 일치하는 서버의 문서에 대한 HTML 하이퍼링크를 포함하는 웹 페이지로 사용자의 브라우저에 표시됩니다 . 각 페이지는 기본적으로 10 개 레코드 (히트 수) 를 표시하며 관련도를 기준으로 내림차순으로 정렬됩니다 . 각 레코드는 파일 이름 , 크기 , 작성일 등과 같은 정보를 나열합니다 . 일치하는 단어도 강조됩니다 .

참고	서버 관리자는 검색 페이지 사용자 정의에 표시된 것과 같이 이 결과 검색 페이지를 사용자 정의할 수 있습니다

검색 페이지 사용자 정의

Sun ONE Web Server 는 기본 검색 쿼리 및 검색 결과 페이지를 제공하는 기본 검색 응용 프로그램을 포함합니다 . 이러한 웹 페이지는 있는 그대로 사용하거나 특정 요구를 충족하도록 사용자 정의할 수 있습니다 . 그러한 사용자 정의는 다른 로고로 웹 페이지의 브랜드를 다시 표시하는 것과 같이 간단하거나 검색 결과를 표시하는 순서를 변경하는 것과 같이 복잡할 수 있습니다 .

패턴 파일은 Sun ONE Web Server 6.0 에서와 같이 검색 인터페이스를 사용자 정의하는 데 더 이상 사용되지 않습니다 . 대신 , 사용자 정의는 이제 Sun ONE Web Server 6.1 과 함께 포함되는 JSP 태그 라이브러리 세트를 사용하여 수행됩니다 . 기본 검색 응용 프로그램은 검색 태그 라이브러리를 사용하여 사용자 정의 검색 인터페이스를 구축하는 방법을 보여주는 예제 JSP 를 제공합니다 . `/bin/https/webapps/search`에 있는 기본 검색 응용 프로그램을 사용자 정의 검색 태그 사용을 보여주는 예제 응용 프로그램으로 살펴볼 수 있습니다 .

기본 검색 인터페이스는 4 가지 기본 구성요소 , 즉 헤더 , 꼬리말 , 쿼리 형식 및 결과로 구성됩니다 .

이러한 기본 요소는 태그의 속성 값을 변경하기만 하면 쉽게 사용자 정의할 수 있습니다 . 더 자세한 사용자 정의는 태그 라이브러리를 사용하여 완수할 수 있습니다 .

이 부분에서는 다음 항목에 대해 설명합니다 .

Search 인터페이스 구성요소

Search 인터페이스는 다음 구성요소로 구성됩니다 .

Header

헤더는 로고 , 제목 및 짧은 설명을 포함합니다 .

Footer

꼬리말은 저작권 정보를 포함합니다 .

Form

쿼리 형식은 검색 컬렉션 , 쿼리 입력 상자를 나타내는 확인란 세트 , submit 및 Help 버튼을 포함합니다 .

Results

결과는 기본적으로 페이지당 10 개 레코드로 나열됩니다 . 각 레코드에 대해 제목 , 구절 , 크기 , 기준일 , URL 과 같은 정보가 표시됩니다 . 구절은 강조된 일치 단어가 있는 페이지의 짧은 부분입니다 .

Search Query 페이지 사용자 정의

쿼리 형식은 검색 컬렉션 , 쿼리 입력 상자 및 submit 버튼에 대한 확인란 목록을 포함합니다 . 이 형식은 <collElem>, <queryBox> 및 <submitButton> 태그와 함께 <s1ws:form> 태그를 사용하여 작성됩니다 .

```
<s1ws:form>
   <s1ws:collElem />
   <s1ws:queryBox /> <s1ws:submitButton />
</s1ws:form>
```

쿼리 형식은 페이지 , 중간 , 사이드바 등 어느 곳에든 위치할 수 있습니다 . 또한 컬렉션 선택 상자 , 쿼리 문자열 입력 상자 , Submit 버튼이 수직으로 나열된 크로스바 또는 컬렉션이 확인란으로 나타나고 쿼리 입력 상자 및 Submit 버튼이 그 아래 위치하는 블록와 같은 다른 형식으로도 표시될 수 있습니다 .

다음 예는 <searchForm> 태그 세트을 사용하여 다른 형식으로 쿼리 형식을 작성하는 방법을 보여줍니다 .

수직 바

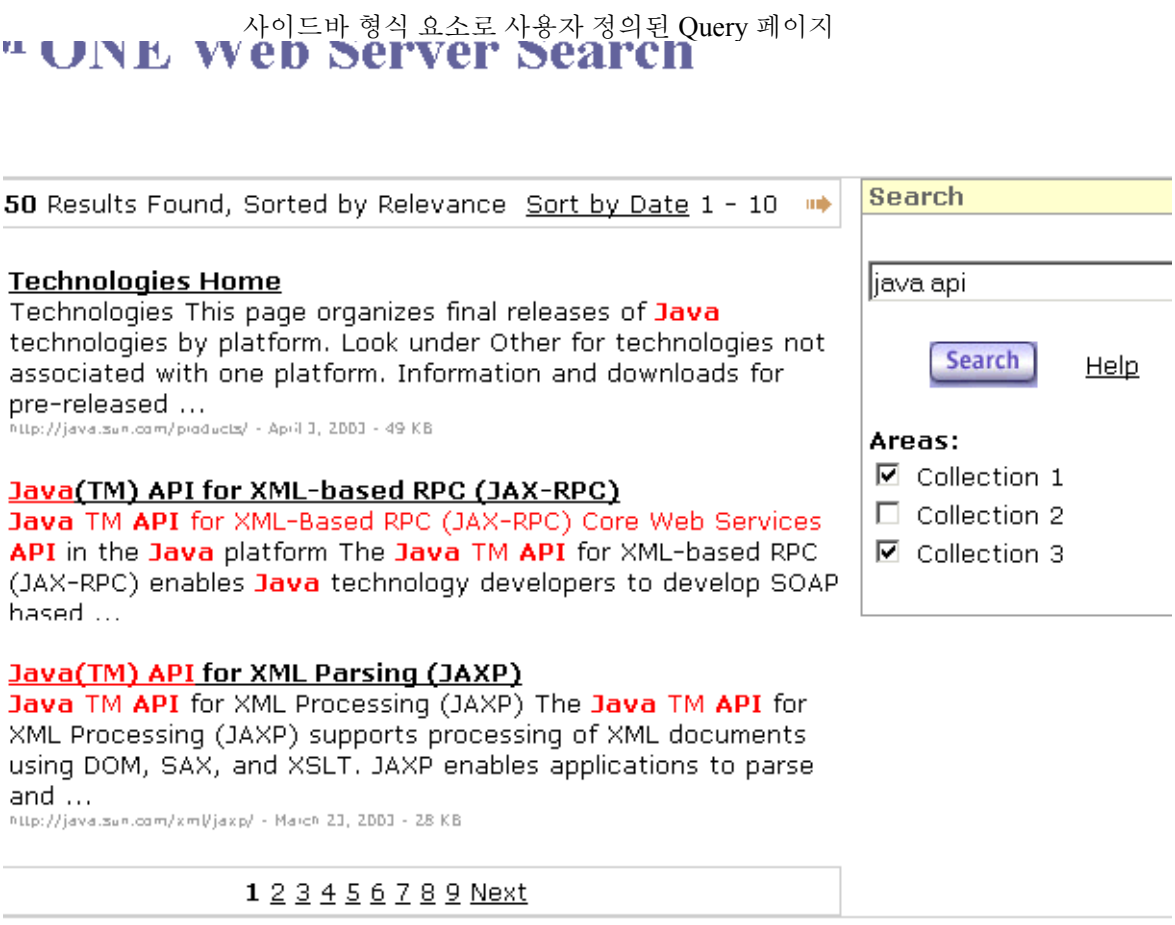
아래 예제 코드는 모든 컬렉션의 선택 상자 , 쿼리 입력 상자 및 submission 버튼이 모두 한 행으로 표시되는 형식을 만듭니다 .

```
<s1ws:form>
   <table cellspacing="0" cellpadding="3" border="0">
   <tr class="navBar">
      <td class="navBar"><s1ws:collElem type=° select° /></td>
      <td class="navBar">
         <s1ws:querybox size="30" />
         <s1ws:submitButton class="navBar" style="padding:0px;
margin:0px; width:50px" />
      </td>
   </tr>
   </table>
</s1ws:form>
```

사이드바 블록

형식 요소가 사이드바에 정렬되고 사이드바의 다른 항목과 동일한 형식을 사용하는 "Search" 라는 제목을 가진 형식 블록을 만들 수 있습니다 . 그러한 정렬의 효과는 다음 그림에 표시된 것과 같습니다 .

사이드바 형식 요소로 사용자 정의된 Query 페이지



아래 제공된 예제 코드에서 형식 본문에는 한 개 열로 정렬된 3 개 확인란이 사용 가능 검색 컬렉션을 나열하고 있습니다 . 쿼리 입력 상자와 Submit 상자는 아래 위치합니다 .

```
<s1ws:searchForm>
   <table>
<!--... other sidebar items ... -->
   <tr class="Title"><td>Search</td></tr>
   <tr class="Body">
      <td>
      <table cellspacing="0" cellpadding="3" border="0">
      <tr class="formBlock">
         <td class="formBlock"> <s1ws:collElem type="checkbox"
cols="1" values="1,0,1,0" /> </td>
      </tr>
      <tr class="formBlock">
         <td class="formBlock"> <s1ws:querybox size="15"
maxlength="50" /> </td>
      </tr>
      <tr class="formBlock">
         <td class="formBlock"> <s1ws:submitButton class="navBar"
style="padding:0px; margin:0px; width:50px" /> </td>
      </tr>
      </table>
      </td>
   </tr>
   </table>
</s1ws:searchForm>
```

Search Results **페이지 사용자 정의**

검색 결과는 다음과 같이 생성됩니다 .

- `<formAction>` 태그는 모든 형식 요소에서 값을 검색하여 기본 확인을 수행합니다 .
- `<search>` 태그 , `<resultIteration>` 태그 및 기타 태그는 `<formAction>` 태그 내부에 나타나고 모든 형식 요소의 값에 액세스합니다 .

- <search> 태그는 <formAction> 에서 쿼리 문자열 및 컬렉션으로 검색을 실행하고 검색 결과를 pageContext 에 저장합니다 .
- 그러면 <resultIteration> 태그가 검색을 수행하고 결과 세트를 통해 반복합니다 .

태그의 속성 값을 변경하기만 하면 Search Results 페이지를 사용자 정의할 수 있습니다 .

다음 예제 코드는 제목 표시줄로 시작한 다음 레코드 수를 지정된 대로 표시하고 마지막으로 검색 표시줄을 표시합니다 . 제목 표시줄은 반환된 총 레코드 범위 , 예를 들어 1 - 10 과 함께 검색에 사용된 쿼리 문자열을 포함합니다 . 각 레코드의 경우 레코드 부분은 파일에 대한 링크 , 키워드가 강조된 최고 3 개 구절 , 작성일 , 문서 크기를 가진 제목을 표시합니다 .

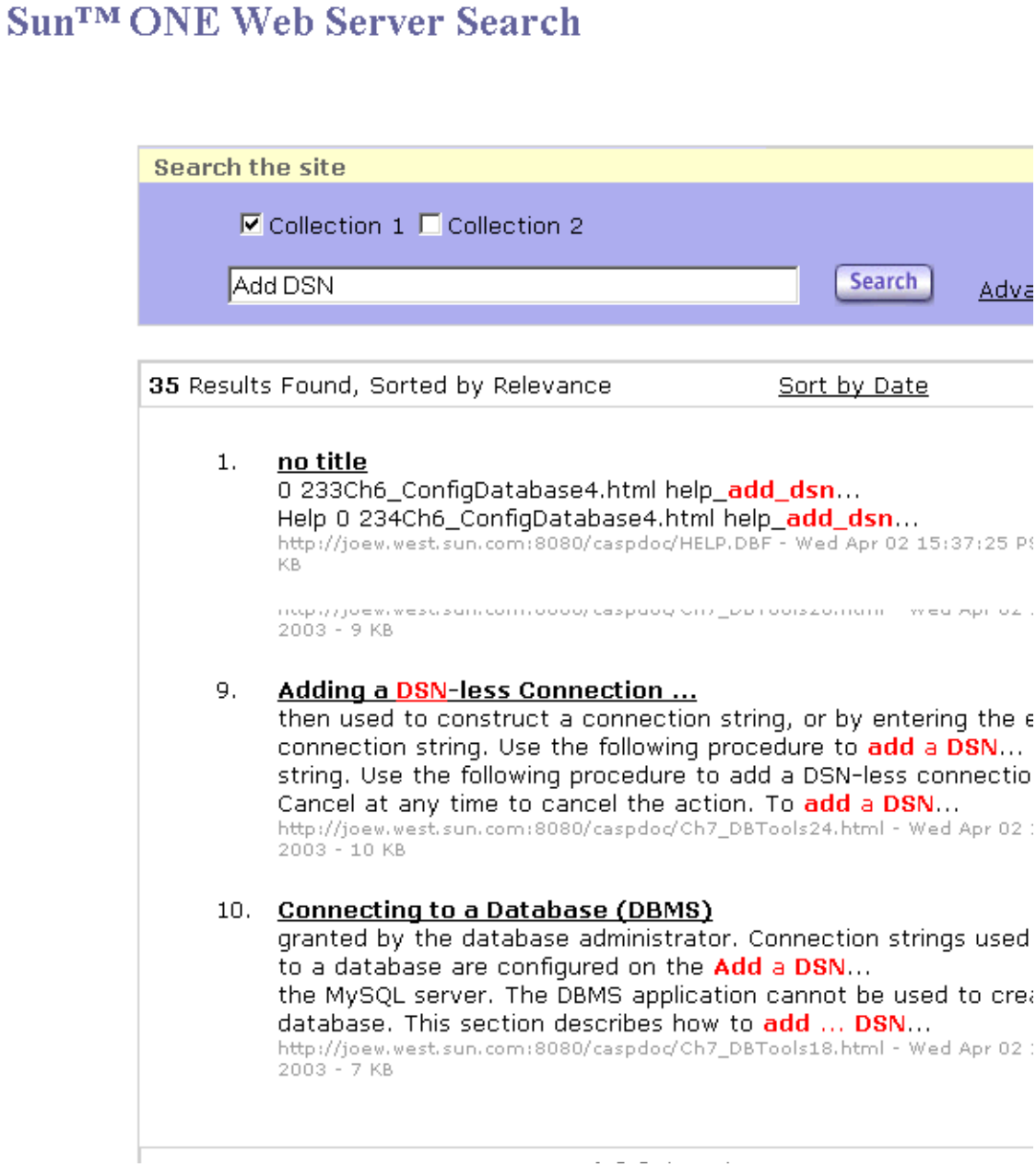
부분 끝에 있는 검색 표시줄은 이전 및 다음 페이지에 대한 링크와 현재 페이지 앞과 뒤의 8 개 추가 페이지에 대한 직접 링크를 제공합니다 .

```
<s1ws:formAction />

<s1ws:formSubmission success="true" >

   <s1ws:search scope="page" />

   <!--search results-->

   (...html omitted...)

       <s1ws:resultStat formId="test" type="total" /></b> Results
Found, Sorted by Relevance</span></td><td>

       <span class="body"><a href="/search/search.jsp?">Sort by
Date</a></span></td>

       <td align="right"><span class="body">

       <s1ws:resultNav formId="test" type="previous" caption="<img
border=0 src=\"images/arrow-left.gif\" alt=\"Previous\">" />

        <s1ws:resultStat formId="test" type="range" />

        <s1ws:resultNav formId="test" type="next" caption="<img
border=0 src=\"images/arrow-right.gif\" alt=\"Next\">" />

         <!img alt="Next" src="images/arrow-right.gif"
border="0" WIDTH="13" HEIGHT="9">

          (...html omitted...)

       <table border=0>

       <s1ws:resultIteration formId="test" start="1" results="15">
```

```
            <tr class=body>
                <td valign=top>
                <s1ws:item property='number' />.  
                </td>
                <td>
                    <b><a href="<s1ws:item property='url'
/>"><s1ws:item property='title' /></a></b>
                    <br>
                    <s1ws:item property='passages' />
                    <font color="#999999" size="-2">
                    <s1ws:item property='url' /> -
                    <s1ws:item property='date' /> -
                    <s1ws:item property='size' /> KB
                    </font><br><br>
                </td>
            </tr>
        </s1ws:resultIteration>
        </table>
        (...html omitted...)
        <s1ws:resultNav formId="test" type="previous" />
        <s1ws:resultNav formId="test" type="full" offset="8" />
        <s1ws:resultNav formId="test" type="next" />
    (...html omitted...)
</s1ws:formSubmission>
```

다음 그림은 사용자 정의된 검색 결과 페이지를 표시합니다 .

사용자 정의된 Search Results 페이지



기본 검색 결과 인터페이스는 태그를 조작하고 HTML 을 수정하여 쉽게 사용자 정의할 수 있습니다 . 예를 들어 , 검색 표시줄은 검색 결과 앞에 복사되어 위치할 수 있습니다 . 사용자는 검색 레코드의 모든 속성을 표시하거나 표시하지 않도록 선택할 수 있습니다 .

형식과 더불어 사용되는 것 외에 <search>, <resultIterate> 및 관련 태그는 나열된 특정 주제에 사용될 수 있습니다 . 다음 예제 코드는 사이트의 Java Web Services 의 상위 10 개 아티클을 나열합니다 .

```
<s1ws:search collection="Articles" query="Java Web Services" />
<table cellspacing="0" cellpadding="3" border="0">
  <tr class="Title"><td>Java Web Services</td></tr>
</table>
<table cellspacing="0" cellpadding="3" border="0">
<s1ws:resultIteration>
<tr>
<td><a href="<s1ws:item property='URL' />"> <s1ws:item
property='Title'/></a></td>
</tr>
</s1ws:resultIteration>
</table>
```

별도의 페이지로 Form 및 Results 사용자 정의

Form 및 Results 페이지를 분리해야 할 경우 <form> 태그 세트를 사용하여 Form 페이지를 , <formAction> 태그 세트를 사용하여 Results 페이지를 만들어야 합니다 .

순조로운 페이지 흐름을 위해 Form 페이지는 Results 페이지에 추가되어야 합니다 .

태그 규약

다음 태그 규약을 주의하십시오 .

- 태그 클래스는 패키지 com.sun.web.search.taglibs 에 속합니다 .

- 모든 pageContext 속성은 접두사 com.sun.web 을 갖습니다 . 검색 결과의 속성은 예를 들어 , com.sun.web.searchresults.*form_id* 입니다 . *form_id* 는 형식의 이름입니다 .
- 태그 라이브러리는 접두사 s1ws 로 참조됩니다 . 태그 이름 및 해당 속성은 첫 문자가 대문자 표시된 각 내부 단어와 함께 대소문자 혼합됩니다 . 예를 들어 , pageContext 와 같습니다 .

태그 표준

Sun ONE Web Server 는 검색 인터페이스에서 검색 쿼리 및 검색 결과 페이지를 사용자 정의하는 데 사용할 수 있는 JSP 태그 세트를 포함합니다 .

검색 페이지를 사용자 정의하는 데 사용할 수 있는 완전한 목록의 JSP 태그에 대한 더 자세한 내용은 Sun ONE Web Server 6.1 *Programmer's Guide to Web Applications* 를 참조하십시오 .

19 장

WebDAV 를 사용하는 웹 게시

Sun ONE Web Server 6.1 은 웹 기반 공동 작업에서 표준으로 부상하고 있는 WebDAV, 즉 Web-based Distributed Authoring and Versioning 을 지원합니다 . WebDAV 는 클라이언트가 원격 웹 컨텐트 저작 작업을 수행하도록 하는 HTTP/1.1 프로토콜의 확장입니다 .

이 장에서는 Sun ONE Web Server 6.1 에서 WebDAV 를 사용하는 방법에 대해 설명합니다 . 다음 부분을 포함합니다 .

WebDAV 정보

WebDAV 는 HTTP/1.1 프로토콜의 확장으로서 HTTP 및 XML 뿐만 아니라 텍스트 , 그래픽 , 스프레드시트 및 모든 기타 형식을 포함하는 모든 유형의 웹 리소스에 대해 저작 지원을 제공하는 새로운 HTML 메소드 및 헤더를 추가합니다 .

WebDAV 로 실행할 수 있는 작업에는 다음과 같은 것들이 있습니다 .

- **등록정보 (메타 데이터) 조작 .** WebDAV 메소드 PROPFIND 및 PROPPATCH 를 사용하여 저작자 및 작성 날짜와 같은 웹 페이지에 대한 정보를 만들고 제거 , 쿼리할 수 있습니다 .
- **컬렉션 및 리소스 관리 .** WebDAV 메소드 GET, PUT, DELETE 및 MKCOL 을 사용하여 문서 세트를 만들고 계층적 구성원 목록 (파일 시스템의 디렉토리 목록과 유사) 을 검색
- **잠금 .** WebDAV 를 사용하여 한 사람 이상이 동시에 한 문서에서 작업하지 못하도록 할 수 있습니다 . WebDAV 메소드 LOCK 및 UNLOCK 을 사용하여 exclusive 또는 shared 잠금을 사용함으로써 " 업데이트 유실 "(변경 사항 겹쳐쓰기) 문제를 방지할 수 있습니다 .
- **이름 공간 작업 .** WebDAV 를 통해 WebDAV 메소드 COPY 및 MOVE 를 사용하여 웹 리소스를 복사 및 이동하도록 서버에게 지시할 수 있습니다

Sun ONE Web Server 6.1 의 WebDAV 지원은 다음 기능을 제공합니다 .

- RFC2518 과 호환성 및 RFC2518 클라이언트와 상호운용성
- 게시를 위한 보안 및 액세스 제어
- 파일 시스템 기반 WebDAV 컬렉션 및 리소스에서 효율적인 게시 작업

일반 WebDAV 용어

이 부분은 WebDAV 와 작업하면서 만나게 되는 일반적인 용어를 개략적으로 제시합니다 .

URI. URI(Uniform Resource Identifier)는 URL 약자를 사용하여 추가 보안 레이어를 제공하는 파일 ID 입니다 . URL 의 첫 번째 부분은 URL 매핑으로 대체되므로 사용자는 파일의 실제 경로를 알 수 없게 된다 .

소스 URI. 소스 URI 라는 용어는 액세스할 수 있는 리소스의 소스를 가리킵니다 . 소스 URI 의 개념을 이해하려면 다음 예를 보십시오 .

JSP 페이지 , `foo.jsp` 는 URI `/docs/date.jsp` 에 위치합니다 . 이 페이지에는 HTML 마크업 및 Java 코드가 들어 있고 이것을 실행하면 클라이언트 브라우저에 오늘 데이터를 인쇄합니다 . 서버가 페이지를 서비스하기 전에 클라이언트로부터 `foo.jsp` 에 대한 GET 요청을 수신하면 서버는 Java 코드를 실행합니다 . 클라이언트가 수신한 것은 서버에 상주하는 `foo.jsp` 가 아니라 현재 데이터를 표시하는 동적 생성 페이지입니다 .

소스 URI, 즉 /publish/docs 를 만들고 foo.jsp 가 포함된 /docs 디렉토리로 매핑하면 /publish/docs/foo.jsp 에 대한 요청은 /docs/foo.jsp JSP 페이지의 소스 코드에 대한 요청이 됩니다 . 이 경우 , 서버는 Java 코드를 실행하지 않고 페이지를 서비스합니다 . 클라이언트는 처리되지 않은 페이지를 디스크에 저장된 그대로 수신합니다 .

소스 URI 에 대한 요청은 따라서 리소스 소스에 대한 요청입니다 .

컬렉션 . WebDAV 컬렉션은 WebDAV 작업에 대해 사용 설정된 리소스 또는 리소스 집합입니다 . 컬렉션은 URI 세트와 WebDAV 사용 가능한 구성원 리소스를 식별하는 용어인 구성원 URI 를 포함합니다 .

구성원 URI. 컬렉션 내부의 URI 집합의 구성원인 URI.

내부 구성원 URI. 컬렉션의 URI 와 직접 연관된 구성원 URI. 예를 들어 , URL http://info.sun.com/resources/info 를 가진 리소스가 WebDAV 사용 가능이고 URL http://info.sun.com/resources/ 를 가진 리소스 역시 WebDAV 사용 가능일 경우 URL http://info.sun.com/resources/ 를 가진 리소스는 컬렉션이고 http://info.sun.com/resources/info 를 내부 구성원으로 포함합니다 .

등록정보 . 리소스에 대한 설명적 정보를 포함하는 이름 / 값 쌍 . 등록정보는 효율적인 리소스 찾기와 관리에 사용됩니다 . 예를 들어 , ' 작성일 ' 등록정보를 사용하여 리소스가 작성된 날짜를 기준으로 모든 리소스의 색인화를 할 수 있고 ' 작성자 ' 등록정보를 기준으로 작성자 이름별로 색인화할 수 있습니다 .

라이브 등록정보 . 서버가 집행하는 등록정보 . 예를 들어 , 라이브 getcontentlength 등록정보가 GET 요청이 반환하는 엔티티의 길이를 값으로 가지며 이것은 서버가 자동으로 계산합니다 . 라이브 등록정보는 다음을 포함합니다 .

- 등록정보 값은 읽기 전용이며 서버가 유지합니다 .
- 등록정보의 값은 클라이언트가 유지하지만 서버는 제출된 값의 구문 확인을 수행합니다 .

데드 등록정보 . 서버가 집행하지 않는 등록정보 . 서버는 데드 등록정보의 값만 기록합니다 . 클라이언트가 일관성 유지를 책임집니다 .

Sun ONE Web Server 6.1 은 다음 라이브 등록정보를 지원합니다 .

- creationdate
- displayname
- getcontentlanguage
- getcontentlength

- `getcontenttype`
- `gettag`
- `getlastmodified`
- `lockdiscovery`
- `resourcetype`
- `supportedlock`
- `executable`

참고 Sun ONE web Server 는 클라이언트가 리소스와 관련된 파일 권한을 변경할 수 있도록 하는 라이브 등록정보 `executable` 을 지원합니다 .

`executable` 라이브 등록정보에 대한 PROPPATCH 요청의 예 :

```
PROPPATCH /test/index.html HTTP/1.1

Host: sun

Content-type: text/xml

Content-length: XXXX

<?xml version="1.0"?>

<A:propertyupdate xmlns:A="DAV:"
xmlns:B="http://apache.org/dav/props/">

<A:set>

<A:prop>

<B:executable>T</B:executable>

</A:prop>

</A:set>

</A:propertyupdate>
```

잠금 . 리소스를 잠그는 기능은 다른 사용자가 편집하는 동안에는 사용자가 리소스를 수정할 수 없도록 하는 메커니즘을 제공합니다 . 잠금은 겹쳐쓰기 충돌을 방지하고 " 업데이트 유실 " 문제를 해결합니다 .

Sun ONE Web Server 는 두 가지 잠금 유형을 지원합니다 . shared 와 exclusive 가 그것입니다 .

새 HTTP 헤더 . WebDAV 가 HTTP/1.1 프로토콜을 확장하여 작업합니다 . 이것은 클라이언트가 WebDAV 리소스에 대한 요청을 통신할 수 있는 새 HTTP 헤더를 정의합니다 . 이러한 헤더는 다음과 같습니다 .

- `Destination:`
- `Lock-Token:`
- `Timeout:`
- `DAV:`
- `If:`
- `Depth:`
- `Overwrite:`

새 HTTP 메소드 . WebDAV 는 WebDAV 사용 가능 서버에게 요청을 처리하는 방법을 지시하는 몇 가지 새 HTTP 메소드를 도입합니다 . 이러한 방법은 WebDAV 트랜잭션을 수행하는 데 `GET`, `PUT` 및 `DELETE` 와 같은 기존 HTTP 메소드에 추가로 사용됩니다 . 새 HTTP 메소드에 대해 다음에서 간략하게 설명합니다 .

- `COPY`. 리소스 복사에 사용됩니다 . 컬렉션 복사는 `Depth:` 헤더를 사용하고 `Destination:` 헤더는 대상을 지정합니다 . COPY 메소드는 적절한 경우 `Overwrite:` 헤더도 사용합니다 .
- `MOVE`. 리소스 이동에 사용됩니다 . 컬렉션 이동은 `Depth:` 헤더를 사용하고 `Destination:` 헤더는 대상을 지정합니다 . MOVE 메소드는 적절한 경우 `Overwrite:` 헤더도 사용합니다 .
- `MKCOL`. 새 컬렉션 만들기에 사용됩니다 . 이 메소드는 PUT 메소드 오버로드를 피하기 위해 사용됩니다 .
- `PROPPATCH`. 단일 리소스에서 등록정보를 설정 , 변경 , 삭제하는 데 사용됩니다 .
- `PROPFIND`. 하나 이상의 리소스에 속하는 하나 이상의 등록정보를 가져오는 데 사용됩니다 . 클라이언트가 서버에 컬렉션에 대한 PROPFIND 요청을 제출하면 요청에는 Depth: `0`, `1` 또는 `infinity` 의 값을 가진 Depth: 헤더를 포함할 수 있습니다 .
  - `0`. 지정된 URI 의 컬렉션의 등록정보를 가져올 것을 지정합니다 .
  - `1`. 지정된 URI 바로 아래 있는 컬렉션과 리소스의 등록정보를 가져올 것을 지정합니다 .
  - `infinity`. 컬렉션과 그것이 포함하는 모든 구성원 URI 의 등록정보를 가져올 것을 지정합니다 . infinite 깊이를 가진 요청은 전체 컬렉션을 통과하므로 서버에 큰 부담을 부과할 수 있습니다 .

- LOCK. 리소스에 대한 잠금을 추가합니다 . `Lock-Token:` 헤더를 사용합니다 .
- UNLOCK. 리소스에서 잠금을 제거합니다 . `Lock-Token:` 헤더를 사용합니다 .

WebDAV 사용

완전한 WebDAV 트랜잭션은 Sun ONE Web Server 6.1 과 같이 WebDAV 리소스에 대한 요청을 서비스할 수 있는 WebDAV 사용 가능 서버와 Adobe® GoLive® 또는 Macromedia® DreamWeaver® 같이 WebDAV 사용 가능 웹 게시 요청을 지원하는 WebDAV 클라이언트를 포함합니다 .

서버측에서 , WebDAV 요청을 서비스할 수 있도록 Sun ONE Web Server 6.1 사용 가능을 설정 및 구성해야 합니다 .

WebDAV 를 사용하도록 Sun ONE Web Server 6.1 을 구성하려면 다음 단계가 필요합니다 .

- WebDAV 사용 설정
- WebDAV 컬렉션 만들기
- WebDAV 구성
- WebDAV 에 대한 액세스 제어 사용

WebDAV 사용 설정

Sun ONE Web Server 6.1 을 설치하면 기본적으로 WebDAV 는 사용하지 않도록 설정됩니다 .

컬렉션 수준에서 WebDAV 를 사용 설정하려면 서버 인스턴스 수준과 가상 서버 클래스 수준에서도 WebDAV 를 사용 설정해야 합니다 .

참고 컬렉션에 지정된 속성은 가상 서버 수준에서 설정된 속성 값에 우선합니다 .

WebDAV 를 사용 설정할 수 있는 다른 수준은 다음 부분에서 설명합니다 .

- 서버 인스턴스에 대해 WebDAV 사용 설정
- 가상 서버 클래스에 대해 WebDAV 사용 설정

- 컬렉션에 대한 WebDAV 사용 설정

서버 인스턴스에 대해 WebDAV 사용 설정

Administration Server 를 사용하여 전체 서버에 대해 WebDAV 를 사용 가능하게 만들 수 있습니다 . 그렇게 할 경우 다음 지시문이 WebDAV 플러그인을 로드하는 `magnus.conf` 파일에 추가됩니다 .

```
Init fn="load-modules" shlib="/s1ws6.1/lib/libdavplugin.so"
funcs="init-dav,ntrans-dav,pcheck-dav,service-dav"
shlib_flags="(global|now)"
Init fn="init-dav" LateInit=yes
```

`init-dav` Init 기능은 WebDAV 하위 시스템을 초기화 및 등록합니다 .

WebDAV 를 전역적으로 사용하도록 설정하려면 다음 작업을 수행합니다 .

1. WebDAV 를 사용하도록 설정하려는 서버의 Server Manager 에 액세스합니다 .
2. Preferences 탭의 Enable/Disable WebDAV 링크를 누릅니다 .
3. Enable WebDAV Globally 확인란을 표시합니다 .

인스턴스에 대해 WebDAV 사용 설정

4. Apply 를 누릅니다 .
5. Apply Changes 버튼을 눌러 서버를 재새작합니다

   or

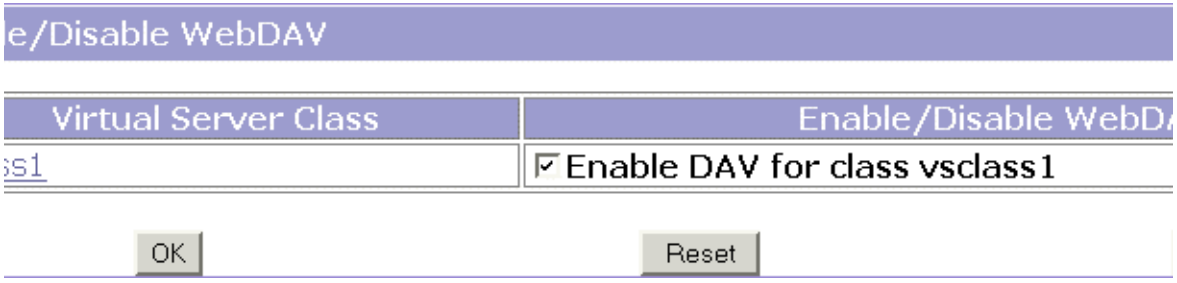
   Load Configuration Files 를 눌러 동적으로 변경 사항을 적용합니다 .

가상 서버 클래스에 대해 WebDAV 사용 설정

특정 가상 서버 클래스에 대해 WebDAV 를 사용 설정하려면 :

1. 가상 서버 클래스 목록이 표시됩니다 .
2. Content Mgmt 탭을 누릅니다 .
3. Enable/Disable WebDAV 링크를 누릅니다 .

   가상 서버 클래스에 대한 WebDAV 사용 설정 .



4. DAV 체크박스에 체크합니다 .
5. OK 를 누릅니다 .

가상 서버 클래스에 대해 WebDAV 를 사용 설정하려면 관련 `obj.conf` 파일이 다음 항목으로 업데이트됩니다 .

```
<Object name="default">
...
Service fn="service-dav"
method="(OPTIONS|PUT|DELETE|COPY|MOVE|PROPFIND|PROPPATCH|LOCK|UN
LOCK|MKCOL)"
Error fn="error-j2ee"
...
</Object>
...
<Object name="dav">
PathCheck fn="check-acl" acl="dav-src"
Service fn="service-dav"
method="(GET|HEAD|POST|PUT|DELETE|COPY|MOVE|PROPFIND|PROPPATCH|L
OCK|UNLOCK|MKCOL)"
</Object>
```

컬렉션에 대한 WebDAV 사용 설정

가상 서버에 하나 이상의 WebDAV 컬렉션을 추가했으면 언제든지 사용하지 않음 및 사용함으로 설정할 것을 선택할 수 있습니다 . 그렇게 하는 방법에 대한 더 자세한 내용은 "WebDAV 컬렉션 편집 " 페이지 421 을 참조하십시오 .

WebDAV 컬렉션 만들기

WebDAV 컬렉션은 WebDAV 작업에 대해 사용 설정된 리소스 또는 리소스 집합입니다 . 이 작업은 웹 게시 , 공동 저작 , 이름 공간 관리 , 메타 데이터 관리를 포함합니다 .

WebDAV 컬렉션을 가상 서버에 추가하려면 다음 작업을 수행합니다 .

1. WebDAV가 서버 인스턴스와 가상 서버 클래스에 대해 사용 설정되도록 합니다. 더 자세한 내용은 " 서버 인스턴스에 대해 WebDAV 사용 설정 " 페이지 417 및 " 가상 서버 클래스에 대해 WebDAV 사용 설정 " 페이지 418 을 참조하십시오 .
2. 관리하려는 가상 서버에 액세스하고 WebDAV 탭을 누릅니다 .
3. Add DAV Collection 페이지에서 다음 정보를 입력합니다 .
   - **URI** (필수). 내용이 액세스되는 URI 입니다 .

- **Source URI** (선택). 리소스가 액세스되는 URI 입니다 .

참고 CGI 또는 SHTML 과 같은 동적 내용을 게시하려면 소스 URI 가 구성되어 있어야 합니다 .

용어 소스 URI 에 대한 설명은 일반 WebDAV 용어를 참조하십시오 .

- **Lock Database** (선택). 잠금 데이터베이스가 유지보수되는 디렉토리입니다 . 기본값은 *server-instance*/lock-db/vs-id 입니다 .
- **Min Lock Timeout** (선택). 초 단위의 잠금 최소 사용 시간입니다. 기본값은 0 입니다 . 더 자세한 내용은 최소 잠금 시간초과를 참조하십시오 .
- **Limit XML Request Body** (선택). 요청 본문의 XML 내용의 최소 크기입니다 . Denial of Service (DOS) 공격 가능성을 방지하기 위해 크기를 제한합니다 .
- **Maximum Property Depth** (선택). PROPFIND 요청의 깊이입니다 .
  - 0 은 지정된 리소스에만 적용됩니다 .
  - 1 은 지정된 리소스와 이에 포함된 다음 수준의 리소스에 적용됩니다 .
  - infinity 는 지정된 리소스와 이에 포함된 모든 리소스에 적용됩니다 .

  기본으로 값은 0 으로 설정됩니다 .
- **Enabled** (선택) 컬렉션에 대해 WebDAV 기능을 사용하도록 설정합니다 .

4. OK 를 누릅니다 .

참고
- Administration server를 사용하여 컬렉션을 추가할 경우에는 서버가 파일 시스템의 컬렉션에 대한 디렉토리를 자동으로 만들지 않습니다 . 컬렉션에 해당하는 디렉토리가 파일 시스템에 만들어졌는지 확인하는 일은 관리자의 책임입니다 .
- UNIX 시스템에서 Web Server 를 root (수퍼유저) 로 설치했고 서버를 다른 사용자로 실행할 경우에는 서버를 실행하는 사용자가 만들어진 WebDAV 컬렉션에 해당하는 디렉토리에 대한 읽기 쓰기 권한이 있도록 하십시오 .

WebDAV 컬렉션 편집

기존 DAV 컬렉션의 속성을 편집할 수 있습니다 . 예를 들어 , 컬렉션에서 액세스 제어를 구성합니다 .

기존 WebDAV 컬렉션을 편집하려면 다음 작업을 수행합니다 .

1. 컬렉션이 존재하는 가상 서버에 액세스하여 WebDAV 탭을 누릅니다 .
2. Edit DAV Collections 페이지에서 다음 정보를 수정합니다 .
   - **Delete**. 컬렉션을 제거할 수 있습니다 .
   - **URI**. 내용이 액세스되는 URI 를 표시합니다 .
   - **Enabled**. WebDAV 가 사용 가능인지 (`true`) 아닌지 (`false`) 나타냅니다 .
   - **Edit Collection**. 다음을 구성하려면 이 버튼을 누릅니다 .
     - **URI** (필수). 내용이 액세스되는 URI 입니다 .
     - **Source URI** (선택). 리소스가 액세스되는 URI 입니다 .
     - **Lock Database** (선택). 잠금 데이터베이스가 유지보수되는 디렉토리입니다 .
     - **Min Lock Timeout** (선택). 초 단위의 잠금 최소 사용 시간입니다 . 더 자세한 내용은 최소 잠금 시간초과를 참조하십시오 .

참고 `minlocktimeout` 의 값이 -1 이면 무한대 잠금을 나타냅니다 .

     - **Limit XML Request Body** (선택). 요청 본문의 XML 내용의 최소 크기입니다 .
     - **Maximum Property Depth** (선택). `PROPFIND` 요청의 깊이입니다 .

       - `0` 은 지정된 리소스에만 적용됩니다 .

       - `1` 은 지정된 리소스와 이에 포함된 다음 수준의 리소스에 적용됩니다 .

       - `infinity` 는 지정된 리소스와 이에 포함된 모든 리소스에 적용됩니다 .

       기본으로 값은 0 으로 설정됩니다 .
     - **Enabled** (선택) 컬렉션에 대해 WebDAV 기능을 사용하도록 설정합니다 .

- **Edit ACL** 이 컬렉션 또는 URI 의 액세스 제어 제한을 설정하려면 이 버튼을 누릅니다 .

WebDAV 구성

몇 가지 이유로 인해 WebDAV 을 구성할 수 있습니다 . 예를 들어 , 서버 성능을 조정하고 보안 위험을 제거하고 또는 충돌 없는 원격 저작을 제거하기 위한 것일 수 있습니다 .

각각의 구성 요구 사항에 맞추기 위해 WebDAV 리소스에서 서버가 잠금을 유지하는 시간 , 컬렉션의 PROPFIND 요청 깊이 , 요청 본문에 허용된 XML 내용의 최소 크기 등을 변경할 수 있습니다 .

기본 WebDAV 속성은 가상 서버 아래 모든 컬렉션에 대한 가상 서버 수준에서 구성될 수 있습니다 . 구성된 값은 `server.xml` 파일의 `DAV` 요소에 해당합니다 .

WebDAV 속성은 컬렉션 수준에서 구성되어 컬렉션에 대해 구성된 가상 서버 수준 속성에 우선할 수 있습니다 . 컬렉션 수준에서 구성된 속성 값은 `server.xml` 파일의 `DAVCOLLECTION` 요소에 해당합니다 .

- 가상 서버 수준에서 WebDAV 구성
- URI 수준에서 WebDAV 구성

가상 서버 수준에서 WebDAV 구성

가상 서버에 대해 WebDAV 기능을 구성하려면 `DAV` 개체의 속성을 편집해야 합니다 . Administration Server 를 사용하거나 수동으로 `server.xml` 파일을 편집하여 할 수 있습니다 .

다음 표는 구성할 수 있는 `DAV` 개체의 속성에 대해 설명합니다 .

표 1) `DAV` 개체의 속성

속성	설명
`enabled`	WebDAV 기능이 이 가상 서버에 대해 사용 가능한지 지정합니다 . 이것은 선택적 속성입니다 . 기본값은 `true` 입니다 . 가능한 값은 `true` 및 `false` 입니다 .

표 1) DAV 개체의 속성

속성	설명
`lockdb`	잠금 데이터베이스가 유지보수되는 디렉토리를 지정합니다 . 이것은 선택적 속성입니다 .
`minlocktimeout`	잠금의 최소 사용 시간을 초 단위로 지정합니다 . 이 값은 잠금이 자동으로 제거되기 전에 요소가 잠기는 시간 길이를 나타냅니다 . 더 자세한 내용은 최소 잠금 시간초과를 참조하십시오 . 이것은 선택적 속성입니다 .
`maxxmlrequestbodysize`	요청 본문의 XML 내용의 최소 크기를 지정합니다 . 이것은 선택적 속성입니다 . 기본값은 8K 입니다 . Denial of service(DOS) 공격 가능성을 방지하기 위해 크기를 제한합니다 .
`maxpropdepth`	PROPFIND 요청의 깊이를 지정합니다 . 이것은 선택적 매개변수입니다 . 기본값은 `0` 입니다 . 이 매개변수의 크기를 제한하여 과도한 메모리 소모를 방지합니다 .

URI 수준에서 WebDAV 구성

URI 수준에서 WebDAV 기능을 구성하려면 `server.xml` 파일의 `DAVCOLLECTION` 개체를 편집해야 합니다 .

다음 표는 구성할 수 있는 `DAVCOLLECTION` 개체의 속성을 설명합니다 .

표 2) `DAVCOLLECTION` 개체의 속성

속성	설명
`enabled`	DAV 기능이 이 컬렉션에 대해 사용 가능한지 지정합니다 . 이것은 선택적 속성입니다 . 가능한 값은 `true` 및 `false` 입니다 . 기본값은 `true` 입니다 .
`uri`	내용이 액세스되는 URI 를 지정합니다 . 이것은 필수 속성입니다 .

표 2) DAVCOLLECTION 개체의 속성

속성	설명
sourceuri	리소스가 액세스되는 URI 를 지정합니다 . 더 자세한 내용은 일반 WebDAV 용어 및 WebDAV 사용 가능 서버에서 소스 URI 및 Translate:f 헤더 사용을 참조하십시오 . 이것은 선택적 속성입니다 . sourceuri가 지정되지 않으면 기본적으로 컬렉션의 모든 동적 컨텐트의 소스에 대한 액세스가 거부됩니다 . uri 및 sourceuri 모두에 대해 동일한 URI 를 지정할 수 있습니다 . 이 경우 서버는 항상 동적 내용의 소스를 반환합니다 . 게시를 위한 별도의 보안 가상 서버를 사용할 경우 이렇게 하는 것이 유용할 수 있습니다 .
lockdb	잠금 데이터베이스가 유지보수되는 디렉토리를 지정합니다 . 이것은 선택적 속성입니다 .
minlocktimeout	잠금의 최소 사용 시간을 초 단위로 지정합니다 . 이 값은 잠금이 자동으로 제거되기 전에 요소가 잠기는 시간 길이를 나타냅니다 . 더 자세한 내용은 최소 잠금 시간초과를 참조하십시오 . 이것은 선택적 속성입니다 .
maxxmlrequestbodysize	요청 본문의 XML 내용의 최소 크기를 지정합니다 . 이것은 선택적 속성입니다 . Denial of Service (DOS) 공격 가능성을 방지하기 위해 크기를 제한합니다 .
maxpropdepth	PROPFIND 요청의 깊이를 지정하여 컬렉션의 구성원 리소스를 나열합니다 . 이것은 선택적 매개변수입니다 . 이 매개변수의 크기를 제한하여 과도한 메모리 소모를 방지합니다 .

WebDAV 사용 가능 서버에서 소스 URI 및 Translate:f 헤더 사용

WebDAV 메소드는 리소스나 컬렉션의 소스에서 작동합니다 . GET 및 PUT 과 같은 HTTP 메소드는 WebDAV 프로토콜에 의해 오버로드되므로 이러한 방법을 사용한 요청은 리소스 소스에 대한 요청이거나 리소스 내용 (출력) 에 대한 요청일 수 있습니다 .

Microsoft 및 기타 많은 WebDAV 공급자가 요청을 가진 `Translate:f` 헤더를 전송함으로써 요청이 소스에 대한 것임을 서버에게 알리는 방법으로 이 문제를 해결하였습니다 . 대중적인 WebDAV 클라이언트 Microsoft WebFolders 와 상호운용 가능하기 위해 Sun ONE Web Server 6.1 은 `Translate:f` 헤더를 리소스 소스에 대한 요청으로 인지합니다 . `Translate:f` 헤더를 보내지 않는 클라이언트를 수용하기 위해 Sun ONE Web Server 6.1 은 소스 URI 를 정의합니다 . 용어 소스 URI 의 상세한 설명은 일반 WebDAV 용어를 참조하십시오 .

WebDAV 사용 가능 컬레션의 경우 URI 에 대한 요청은 요청의 내용 (출력) 을 검색하고 소스 URI 에 대한 요청은 리소스의 소스를 검색합니다 . `Translate:f` 헤더를 가진 URI 에 대한 요청은 소스 URI 에 대한 요청으로 처리됩니다 .

기본적으로 리소스의 소스에 대한 모든 액세스는 서버 인스턴스 특정 ACI 파일에 다음 선언이 있는 `dav-src` ACL 에 의해 거부됩니다 .

```
deny (all) user = "anyone";
```

사용자는 소스 URI 에 액세스 권한을 추가함으로써 사용자에게 소스에 대한 액세스를 허용할 수 있습니다 . URI 특정 ACL 추가에 대한 더 자세한 내용은 WebDAV 에 대한 액세스 제어 사용을 참조하십시오 .

리소스 잠금 및 잠금 해제

Sun ONE Web Server 는 서버 관리자가 리스소를 잠궈서 해당 리소스에 대한 액세스를 직렬화할 수 있도록 합니다 . 잠금을 사용하여 특정 리소스에 액세스하는 사용자는 다른 사용자가 동일한 리소스를 수정하지 않는다는 점을 안심할 수 있습니다 . 이런 식으로 " 업데이트 유실 " 문제가 해결되어 여러 사용자가 서버의 리소스를 공유합니다 . 서버가 유지보수하는 잠금 데이터베이스는 발행된 잠금 토큰을 추적하고 클라이언트에 의해 사용됩니다 .

Sun ONE Web Server 6.1 은 항상 모든 리소스에 대해 고유하도록 고안된 `opaquelocktoken` URI 스킴을 지원합니다 . 이것은 ISO-11578 에서 설명되는 Universal Unique Identifier (UUID) 메커니즘을 사용합니다 .

Sun ONE Web Server 6.1 은 다음 두 가지 유형의 잠금 메커니즘을 인지합니다 .

- Exclusive 잠금
- Shared 잠금

Exclusive 잠금

exclusive 잠금은 단일 사용자에게만 리소스에 대한 액세스를 부여하는 잠금입니다 . 다른 사용자는 리소스의 exclusive 잠금이 제거된 후에만 동일한 리소스에 액세스할 수 있습니다 .

Exclusive 잠금은 때로는 리소스 잠금에 대해 너무 경직되고 비용이 많이 드는 메커니즘입니다 . 예를 들어 , 프로그램 장애 또는 잠금 소유자가 리소스 잠금 해제를 잊어버린 경우 exclusive 잠금을 제거하기 위해 잠금 시간초과 또는 관리자의 개입이 필요할 수 있습니다 .

Shared 잠금

shared 잠금을 사용하여 여러 사용자가 리소스에 대한 잠금을 받을 수 있습니다 . 그러므로 적절한 액세스 권한이 있는 사용자는 누구나 잠금을 얻을 수 있습니다 .

shared 잠금을 사용하면 잠금 소유자는 다른 통신 채널을 사용하여 작업을 조정할 수 있습니다 . shared 잠금을 하는 의도는 공동 작업자가 어떤 다른 사람이 리소스에서 작업하는지 알게 하기 위한 것입니다 .

잠금 관리

Sun ONE Web Server 6.1 은 모든 두드러진 잠금 , 그 유형 , 보유 리소스 , 잠금 지속 시간 등을 볼 수 있도록 하는 잠금 관리 기능을 제공합니다 .

잠금 관리를 사용하려면 다음을 수행하십시오 .

1. WebDAV 사용 가능 가상 서버에 액세스합니다 .
2. WebDAV 탭을 누릅니다 .

3. Lock Management 링크를 누릅니다 .
4. 잠금 데이터베이스와 두드러진 잠금 및 기타 정보를 보려는 WebDAV 사용 가능 URI 를 선택합니다 .
5. List Lock Info 를 누릅니다 .

최소 잠금 시간초과

`server.xml` 파일의 `DAV` 또는 `DAVCOLLECTION` 개체의 `minlocktimeout` 속성의 값을 구성하여 잠금을 제어할 수 있습니다 . `minlocktimeout` 속성은 잠금의 최소 사용 시간을 초 단위로 지정합니다 . 이 값은 잠금이 자동으로 제거되기 전에 요소가 잠기는 시간 길이를 나타냅니다 .

이것은 선택적 속성입니다 . 값이 -1 로 설정되면 잠금은 만료되지 않습니다 . 값을 `0` 으로 설정하면 컬렉션의 모든 요청이 요청에 지정된 Timeout 헤더로 잠깁니다 .

Timeout 헤더가 지정되지 않으면 리소스는 시간초과 무한대로 잠금됩니다 . 요청의 Timeout 헤더가 값 `Infinite` 로 설정되는 경우에도 리소스는 무한대 시간초과로 잠깁니다 .

WebDAV 리소스에 대한 요청의 Timeout 헤더 값이 `server.xml` 에 지정된 `minlocktimeout` 값과 같거나 크면 리소스는 요청에 지정된 시간 동안 잠깁니다 .

그러나 리소스의 Timeout 헤더 값이 `server.xml` 에 지정된 `minlocktimeout` 값보다 작으면 리소스는 `server.xml` 에 지정된 `minlocktimeout` 값으로 잠깁니다 .

다음 표는 Sun ONE Web Server 가 잠금 요청을 처리하는 방법을 보여줍니다 .

표 3) Sun ONE Web Server 가 잠금 요청을 처리하는 방법

요청의 Timeout 헤더 값이 다음으로 설정된 경우	**리소스는 :**
`Infinite`	시간초과가 -1(무한대) 로 잠김
None	시간초과가 -1(무한대) 로 잠김
`Second-`*xxx*	• *xxx* 값으로 잠김 , *xxx* 가 `server.xml` 에 설정된 `minlocktimeout` 값과 같거나 클 경우 또는 • `server.xml`, 에 지정된 `minlocktimeout` 값으로 잠김 , *xxx* 가 `server.xml` 에 설정된 `minlocktimeout` 값보다 작을 경우

잠금 요청의 예

이 예는 시간초과가 500 초인 리소스 `/col1/myfile.html` 의 exclusive 쓰기 잠금에 대한 요청을 예시합니다 .

```
LOCK /col1/myfile.html HTTP/1.1
Host: sun
Content-Type: text/xml; charset="utf-8"
Content-Length: 259
Timeout: Second-500
<?xml version="1.0" encoding="utf-8" ?>
<d:lockinfo xmlns:d="DAV:">
    <d:locktype><d:write/></d:locktype>
    <d:lockscope><d:exclusive/></d:lockscope>
    <d:owner>
    <d:href>http://info.sun.com/resources/info.html</d:href>
    </d:owner>
</d:lockinfo>
```

WebDAV 에 대한 액세스 제어 사용

누가 WebDAV 사용 문서 및 디렉토리에 액세스하고 다른 사용자 또는 사용자 그룹이 해당 파일에 대해 수행할 수 있는 작업은 무엇인지 제어할 수 있습니다 . 또한 파일이나 폴더에 대한 액세스를 완전 금지하거나 특정 권한이 있는 사용자에게 액세스를 제한할 수 있습니다 .

서버를 지배하는 기본 액세스 제어 (ACL) 가 사용자의 요구에 맞게 제한적이거나 유연하지 않을 경우 , Restrict Access 기능 (Server Preferences 를 선택하고 Restrict Access 링크를 누름) 을 사용하여 WebDAV 사용 리소스에 대한 액세스를 제한하는 데 보다 적합한 ACL 을 만들 수 있습니다 .

WebDAV 요청은 각각 AuthTrans 및 PathCheck NSAPI 단계에 의해 각각 인증 및 허가됩니다 . 다음 예에서 액세스 제어 규칙은 사용자 이름 "joe" 를 제외한 모두에게 컬렉션 `/catalog` 에 대한 쓰기를 거부하고 액세스를 제거하도록 정의됩니다 .

```
acl "uri=/catalog/*";

deny(all)

user="anyone";

allow (read,list,execute,info)
```

```
user = "all";
allow(write,delete)
user="joe";
```

더 자세한 내용은 WebDAV 컬렉션 편집를 참조하십시오 .

WebDAV 사용 리소스의 액세스 제한

WebDAV 컬렉션에 대한 액세스 제한은 원시 ACL 파일을 사용하여 지정됩니다 . 모든 WebDAV 메소드는 WebDAV 사용 리소스에 대한 특정 액세스 권한을 필요로 합니다 . 예를 들어 , WebDAV 사용 파일이 동시 사용자에 의해 공유되려면 동시 제어를 위한 리소스를 잠그거나 잠금 해제하기 위해 리소스에 대한 쓰기 권한이 필요합니다 .

다음 테이블은 WebDAV 메소드에 필요한 권한을 요약합니다 .

표 19-4) WebDAV 에 필요한 권한

DAV 메소드	**필요한 액세스 권한**
DELETE	삭제
PROPFIND	읽기
PROPPATCH	쓰기
LOCK/UNLOCK	쓰기
MKCOL	쓰기
COPY(*src*,*dst)*	*src* - 읽기
	dst - 쓰기
MOVE(*src*,*dst*)	*src* - 삭제
	dst - 쓰기
GET on request-uri	읽기
GET on request-uri	읽기
Translate:f	
PUT on request-uri	쓰기
PUT on request-uri	쓰기
Translate:f	

보안 고려사항

WebDAV 을 사용할 경우 다음 보안 고려사항을 염두에 두십시오 .

- WebDAV 사용 서버 프로세스가 제어해야 하는 파일 시스템에 대한 읽기 / 쓰기 권한을 가지도록 합니다 .
- 보안상의 이유로 다른 청취 소켓 , 즉 액세스가 제한되고 SSL 을 사용하여 전송된 데이터를 암호화하는 청취 소켓에서 WebDAV 사용 가상 서버를 구성하고자 할 수 있습니다 . SSL 사용에 대한 더 자세한 내용은 인증서 및 키 사용를 참조하십시오 .
- 요청 본문의 XML 내용의 크기를 제한하여 Denial of Service (DOS) 공격을 방지합니다 . 기본적으로 크기는 8K 로 제한됩니다 .
- Basic 인증은 cleartext 를 사용하여 인증 세부 내용을 전송하기 때문에 연결이 안전하지 않는 한 WebDAV 클라이언트를 인증하기 위해 Basic 인증이 아닌 Digest 를 사용합니다 .
- PROPFIND 요청은 서버 내용에 대한 원하지 않은 액세스가 발생할 잠재적 위험이 있기 때문에 WebDAV 사용 리소스를 보안하기 위해 액세스 제어 기술을 사용합니다 .
- WebDAV 는 소스 URI 기능을 통해 잠재적으로 스크립트 리소스와 같은 민감한 정보를 담은 URI 를 노출시킬 수 있습니다 . 스크립트의 원격 저작을 허용할 위험이 있음을 인식하고 권한 있는 사용자로만 소스 리소스에 대한 읽기 쓰기 액세스를 제한해야 합니다 .
- PROPFIND 요청의 깊이를 제한하여 과도한 메모리 소모를 방지합니다 . 기본으로 깊이는 0 으로 제한됩니다 .

제 5 부

부록

부록 A

명령줄 유틸리티

이 부록에서는 HttpServerAdmin 명령줄 유틸리티를 이용하는 방법에 대하여 설명합니다 .

HttpServerAdmin (가상 서버 관리)

HttpServerAdmin 은 Server Manager 와 Class Manager 에 있는 가상 서버 사용자 인터페이스와 동일한 관리 기능을 수행하는 명령줄 유틸리티입니다 . 명령줄 인터페이스를 사용하여 가상 서버를 설정하려는 경우에는 HttpServerAdmin 을 사용합니다 .

참고 HttpServerAdmin 명령줄 유틸리티를 사용하려면 시스템에 대한 수퍼유저 권한이 있어야 합니다 .

HttpServer Admin 명령줄 유틸리티의 위치는 *server_root*/bin/https/httpadmin/bin 디렉토리입니다 .

HttpServerAdmin 을 실행하기 전에 환경의 서버 루트 디렉토리에 대하여 IWS_SERVER_HOME 환경 변수를 설정해야 합니다 .

예를 들어 UNIX/Linux 시스템의 경우 ,

```
setenv IWS_SERVER_HOME /usr/sun/servers
```

Windows 시스템의 경우 ,

1. 제어판에서 시스템을 선택합니다 .
2. 환경 탭을 누릅니다 .
3. 변수 필드에 IWS_SERVER_HOME 을 입력하고 값 필드에 서버 루트의 경로를 입력합니다 .

4. 설정을 누릅니다 .
5. OK 를 누릅니다 .

참고 모든 명령을 수행할 수 있도록 하려면 가상 서버 정보가 저장된 `server.xml` 파일에 대한 쓰기 권한이 있어야 합니다 .

HttpServerAdmin 구문

`HttpServerAdmin` 구문은 다음과 같습니다 .

`HttpServerAdmin` *command_name command_options* `-d` *server_root* `-sinst` *http_instance*

명령 매개 변수에 대한 온라인 설명을 보려면 다음 명령을 입력합니다 .

```
./HttpServerAdmin -h
```

command_name 매개 변수에는 네 가지 값을 사용할 수 있습니다 .

- control
- create
- delete
- list

각 명령줄에는 자체의 명령 옵션 세트가 있습니다 . 자세한 내용은 이 장의 뒤에 있는 각 명령에 대한 설명을 참조하십시오 .

명령 매개 변수의 값에 상관 없이 표에 있는 매개 변수는 모든 `HttpServerAdmin` 명령 사용에 적용할 수 있습니다 .

표 A-1) HttpServerAdmin 매개 변수

매개 변수	값
`-d` *server_root*	(필수). 이 매개 변수는 서버 루트에 대한 경로 (서버가 설치된 위치) 를 지정합니다 .
`-sinst` *http_instance*	(필수). 이 매개 변수는 `HttpServeAdmin` 이 영향을 미치는 인스턴스를 지정합니다 .

control 명령

클래스와 가상 서버를 시작 , 정지 및 사용 하지 않도록 설정하려면 `control` 명령을 사용합니다 . 가상 서버를 지정하지 않으면 클래스의 모든 가상 서버를 시작 , 정지 또는 사용하지 않도록 설정합니다 .

옵션

`control` 명령과 표에 있는 옵션을 사용하여 클래스 및 가상 서버를 제어할 수 있습니다 .

표 A-2) control 명령 옵션

옵션	값
-start	지정된 서버를 시작하거나 , 서버를 지정하지 않는 경우 클래스의 모든 서버를 시작합니다 .
-stop	지정된 서버를 정지하거나 , 서버를 지정하지 않는 경우 클래스의 모든 서버를 정지합니다 .
-disable	지정된 서버를 사용하지 않도록 설정하거나 , 서버를 지정하지 않는 경우 클래스의 모든 서버를 사용하지 않도록 설정합니다 .

구문

`HttpServerAdmin control -cl` *classname*, `-`*control_option* `[-id` *virtual_server*`] -d` *server_root* `-sinst` *http_instance*

매개 변수

이들 매개 변수를 명령 옵션과 함께 사용하여 가상 서버를 제어합니다 .

표 A-3) control 명령 매개 변수

매개 변수	값
-cl *classname*	가상 서버 클래스를 지정합니다 .
-id *virtual_server*	(선택) 제어하는 가상 서버 ID 를 지정합니다 .

예

```
HttpServerAdmin control -cl myclass -start -id myvirtualserver -d
/usr/sun/servers -sinst https-sun.com

HttpServerAdmin control -cl myclass -stop -id myvirtualserver -d
/usr/sun/servers -sinst https-sun.com

HttpServerAdmin control -cl myclass -disable -id myvirtualserver
-d /usr/sun/servers -sinst https-sun.com
```

create 명령

가상 서버 클래스 , 가상 서버 및 청취 소켓을 만들려면 create 명령을 사용합니다 .

옵션

create 명령과 표에 있는 옵션을 사용하여 클래스, 청취 소켓, 가상 서버 및 리소스를 만들 수 있습니다 .

표 A-4) create 명령 옵션

옵션	값
-c	가상 서버 클래스를 만듭니다 .
-l	청취 소켓을 만듭니다 .
-v	가상 서버를 만듭니다 .
-r	리소스를 만듭니다 .

각 옵션에는 모두 다음에서 설명하는 차제의 매개 변수가 있습니다 .

가상 서버 클래스 생성

create 명령의 이 옵션을 사용하여 가상 서버 클래스를 만듭니다 .

구문

HttpServerAdmin create -c -cl *classname* -docroot *document_root* [-obj *obj.conf_file*] [-acptlang *accept_language*] -d *server_root* -sinst *http_instance*

매개 변수

표에 있는 매개 변수를 `create -c` 명령 옵션과 함께 사용하여 클래스를 만듭니다 .

표 A-5) 가상 서버 클래스 생성 매개 변수

매개 변수	값
-cl *classname*	만들려는 클래스의 이름을 지정합니다 .
-docroot *document_root*	클래스의 문서 루트 . 여기에는 반드시 절대 경로를 사용해야 합니다 .
-obj *obj.conf_file*	(선택) 해당 클래스용 obj.conf 파일 . 이 매개 변수를 지정하지 않으면 서버는 obj.conf 파일을 *classname*.obj.conf 롤 만듭니다 . 클래스의 obj.conf 파일을 다른 이름으로 만들려면 여기에 해당 이름을 지정합니다 .
-acptlang *accept_language*	(선택) 이 매개 변수를 지정하지 않으면 기본 값으로 acptlang 가 사용되지 않습니다 .

예

```
HttpServerAdmin create -c -cl myclass1 -docroot /docs -d
/export/sun/servers -sinst https-sun.com
```

청취 소켓 생성

create 명령에 이 옵션을 사용하여 청취 소켓을 만듭니다 .

구문

`HttpServerAdmin create -l -ip` *ip_address* `-port` *port_number* `-sname` *server_name* `-id` *default_virtual_server* `[-sec security] [-acct` *number_of_accept_threads*`] -d` *server_root* `-sinst http_instance`

매개 변수

표에 있는 매개 변수를 `create -l` 명령 옵션과 함께 사용하여 청취 소켓을 만듭니다 .

표 A-6) 청취 소켓 생성 매개 변수

매개 변수	값
-ip *ip_address*	청취 소켓의 IP 주소 .
-port *port_number*	청취 소켓의 포트 번호 .
-sname *server_name*	청취 소켓에 연결할 서버 이름 .
-id *default_virtual_server*	기본 가상 서버의 ID. 가상 서버를 사용하여 청취 소켓을 만들기 전에 이 가상 서버가 반드시 존재해야 합니다 .
-acct *number_of_accept_threads*	(옵션) 청취 소켓용 승인 스레드의 수 .
-sec *on*	(선택) 지정하는 경우 on 을 사용하여 청취 소켓용 보안을 사용 설정합니다 . 지정하지 않는 경우 보안을 사용하지 않습니다 .

예

```
HttpServerAdmin create -l -id ls3 -ip 0.0.0.0 -port 1333 -sname
austen -defaultvs vs2 -sec on -acct 4 -d /export/carey/server6
-sinst https-austen.com
```

가상 서버 생성

create 명령의 이 옵션을 사용하여 가상 서버를 만듭니다 .

참고로 선택 매개 변수의 일부에 대한 값을 포함하지 않으면 기본 값이 사용됩니다 . 해당 기본 값은 서버를 만든 후 언제라도 변경할 수 있습니다 .

구문

`HttpServerAdmin create -v -id` *virtual_server* `-cl` *classname* `-urlh` *urlhosts* `[-state` *state*`][-docroot` *document_root*`] [-mime` *mime_types_file*`] [-aclid` *acl_ID*`] -d` *server_root* `-sinst` *http_instance*

매개 변수

표에 있는 매개 변수를 `create -v` 명령 옵션과 함께 사용하여 가상 서버를 만듭니다 .

표 A-7) 청취 소켓 생성 매개 변수

매개 변수	값
-id *virtual_server*	만들려는 가상 서버의 ID.
-cl *classname*	가상 서버가 속할 클래스 .
-urlh *URL_hosts*	가상 서버의 URL 호스트 . 공백으로 구분된 URL 호스트를 하나 여러 개 지정할 수 있습니다 .
-state *state*	(선택) 유효한 값은 on, off 또는 disable 입니다 .
-docroot *document_root*	(선택) 가상 서버용 문서 루트를 지정하려는 경우 이 매개 변수를 사용합니다 . 반드시 절대 경로 이름을 사용해야 합니다 .
-mime *mime_types_file*	(선택) 가상 서버용 MIME 유형의 이름 .
-aclid *acl_ID*	(선택) `server.xml` 파일에서 사용하는 ACL 파일 ID <ACLID>.

예

```
HttpServerAdmin create -v -id vs3 -cl class1 -urlh annh -d
/export/sun/server6 -sinst https-sun.com

HttpServerAdmin create -v -id vs4 -cl class1 -urlh annh,annh2
-state off -mime mime.types -d /export/sun/server6 -sinst
https-sun.com
```

JDBC 연결 풀 생성

명령줄 인터페이스를 사용하여 JDBC 연결 풀을 만들려면 `create -r` 명령을 사용합니다 .

구문

`HttpServerAdmin -create -r -jdbcconnectionpool -poolname` *jdbcpoolname* `-classname` *classname* [`-steadypoolsize` *steadypoolsize*] [`-maxpoolsize` *maxpoolsize*] [`-poolresizequantity` *poolresizequantity*] [`-idletimeout` *idletimeout*] [`-maxwaittime` *maxwaittime*] [`-connectionvalidation` *true*/*false*] [`-connectionvalidationmethod` *connectionvalidationmethod*] [`-validationtablename` *validationtablename*] [`-failall` *true*/*false*] [`-desc` *description*] [[`-property` *propertyname*=*value*],...]

옵션

create -r 명령 옵션으로 연결 풀을 만드는데 필요한 모든 옵션은 다음 표에서 설명하는 것과 같습니다 .

표 A-8) 연결 풀 생성 매개 변수

매개 변수	값
poolname *jdbcpoolname*	JDBC 연결 풀용 풀 이름 .
classname *classname*	데이터 소스를 구현하는 클래스 이름으로 공급자가 지정합니다 .
steadypoolsize *steadypoolsize*	풀에서 반드시 유지되어야 하는 연결의 최소 수 .
maxpoolsize *maxpoolsize*	풀에서 허용하는 최대 연결 수 .
poolresizequantity *poolresizequantity*	`steadpoolsize` 값에 가까워지는 경우 풀의 크기가 조정되는 배치 크기 .
idletimeout *idletimeout*	풀에서 연결이 유휴 상태를 유지할 수 있는 최대 시간을 초 단위로 지정합니다 .
maxwaittime *maxwaittime*	호출자에게 연결 제한 시간이 지정될 때까지 대기하는 시간을 지정합니다 .
`connectionvalidation` *true/false*	연결이 응용 프로그램에 대하여 파싱되기 전에 해당 연결 풀의 유효성을 검사할 것인지 지정합니다 .
connectionvalidationmethod *connectionvalidationmethod*	데이터베이스 연결의 유효성을 검사할 메소드 . 유효한 값은 `auto-commit`, `meta-data` 및 `table` 입니다 .
validationtablename *validationtablename*	`connectionvalidationmethod` 가 `table` 로 설정된 경우 테이블의 이름 .
failall *true*/*false*	풀에 있는 모든 연결을 차단한 후 다시 설정하여 단일 연결이 실패했는지 확인하도록 할 것인지 지정합니다 .
desc *description*	풀에 대한 설명
property *propertyname=value*	표준 및 사유 JDBC 연결 풀 등록 정보를 지정하는 이름 - 값 쌍 .

예

```
HttpServerAdmin create -r -jdbcconnectionpool -poolname testpool
-classname "oracle.jdbc.pool.OracleDataSource" -property
"URL=jdbc:oracle:thin:@dbhost:1521:ORCL,user=scott,password=tige
r" -d /opt/Sun/S1WS6.1 -sinst testinstance
```

JDBC 리소스 생성

명령줄 인터페이스를 사용하여 JDBC 리소스를 만들려면 `create -r` 명령을 사용합니다 .

구문

`HttpServerAdmin -create -r -jdbc -jndiname` *jndiname* `-poolname` *poolname* [`-desc` *description*] [`-enabled` *true/false*]

옵션

`create -r` 명령 옵션으로 새 JDBC 리소스를 만드는데 필요한 모든 옵션은 다음 표에서 설명하는 것과 같습니다 .

표 A-9) JFBC 리소스 생성 매개 변수

매개 변수	값
jndiname *jndiname*	리소스의 JNDI 이름 .
poolname *poolname*	JDBC 연결 풀용 풀 이름 .
desc *description*	풀에 대한 설명
enabled *true/false*	리소스를 사용할 것인지 지정합니다 .
	JDBC 리소스를 사용하지 않도록 설정하면 어떤 응용 프로그램 구성요소도 이에 연결할 수 없으나 , 구성은 서버 인스턴스에 유지됩니다 .

예

```
HttpServerAdmin create -r -jdbc -jndiname "jdbc/testjdbcresource"
-poolname testpool -d /opt/Sun/S1WS6.1 -sinst testinstance
```

사용자 지정 리소스 생성

명령줄 인터페이스를 사용하여 새 사용자 지정 리소스를 만들려면 `create -r` 명령을 사용합니다 .

구문

`HttpServerAdmin -create -r -custom -jndiname` *jndiname* `-resourcetype` *resourcetype* -factoryclass *factoryclassname* [`-enabled` *true/false*] [`-desc` *description*] [[`-property` *propertyname=value*],...]

옵션

`create -r` 명령 옵션으로 새 JDBC 리소스를 만드는데 필요한 모든 옵션은 다음 표에서 설명하는 것과 같습니다 .

표 A-10) 사용자 정의 리소스 생성 매개 변수

매개 변수	값
jndiname *jndiname*	리소스의 JNDI 이름 .
resourcetype *resourcetype*	리소스 유형 .
factoryclassname *factoryclassname*	개체 팩토리의 클래스이름 .
enabled *true/false*	리소스를 사용할 것인지 지정합니다 .
desc *description*	풀에 대한 설명
property *propertyname=value*	사용자 정의 리소스의 등록 정보를 지정하는 이름 - 값 쌍 .

예

```
HttpServerAdmin create -r -custom -jndiname "testcustomresource"
-resourcetype "java.lang.String" -factoryclass
"com.mycom.test.StringFactory" -d /opt/Sun/S1WS6.1 -sinst
testinstance
```

외부 JNDI 리소스 생성

명령줄 인터페이스를 사용하여 외부 JNDI 리소스를 만들려면 `create -r` 명령을 사용합니다 .

구문

`HttpServerAdmin -create -r -external -jndiname` *jndiname* `-jndilookupname` *jndilookupname* `-restype` *restype* -factoryclass *factoryclass* [`-enabled` *true/false*] [`-desc` *description*] [[`-property` *propertyname=value*],...]

옵션

`create -r` 명령 옵션으로 새 외부 JNDI 리소스를 만드는데 필요한 모든 옵션은 다음 표에서 설명하는 것과 같습니다 .

표 A-11) 외부 JNDI 리소스 생성 매개 변수

매개 변수	값
jndiname *jndiname*	리소스의 JNDI 이름 .
jndilookupname *jndilookupname*	리소스용 JNDI 조회 이름 .
restype *restype*	리소스 유형 .
factoryclass *factoryclass*	개체 팩토리의 클래스이름 .
enabled *true/false*	리소스를 사용할 것인지 지정합니다 .
desc *description*	풀에 대한 설명
property *propertyname=value*	사용자 정의 리소스의 등록 정보를 지정하는 이름 - 값 쌍 .

예

```
HttpServerAdmin create -r -external -jndiname
"testexternalresource" -jndilookupname "rmiconverter" -restype
"samples.rmi.simple.ejb.ConverterHome" -factoryclass
"com.sun.jndi.cosnaming.CNCtxFactory" -property
"java.naming.provider.url=iiop://localhost:3700" -d
/opt/Sun/S1WS6.1 -sinst testinstance
```

전자우편 리소스 생성

명령줄 인터페이스를 사용하여 새 전자우편 리소스를 만들려면 `create -r` 명령을 사용합니다 .

구문

`HttpServerAdmin -create -r -mail -jndiname` *jndiname* `-host` *host* `-user` *user* `-from` *from* `[-storeprotocol` *storeprotocol*`] [-storeprotocolclass` *storeprotocolclass*`] [-transportprotocol` *transportprotocol*`] [-transportprotocolclass` *transportprotocolclass*`] [-enabled` *true/false*`] [-desc` *description*`] [[-property` *propertyname*=*value*`]...]`

옵션

`create -r` 명령 옵션으로 새 전자우편 리소스를 만드는데 필요한 모든 옵션은 다음 표에서 설명하는 것과 같습니다 .

표 A-12) 전자우편 리소스 생성 매개 변수

매개 변수	값
`jndiname` *`jndiname`*	리소스의 JNDI 이름 .
`host` *`host`*	전자우편 서버 호스트 이름 .
`user` *`user`*	전자우편 서버 사용자 이름 .
`from` *`from`*	전자 우편 서버가 메시지 송신자를 표시할 때 사용하는 전자우편 주소 .
`storeprotocol` *`storeprotocol`*	저장 프로토콜 서비스를 지정하며 , 이 서비스는 전자 우편에 연결하고 , 메시지를 검색하며 메시지를 폴더에 저장합니다 . 값의 예는 `imap` 및 `pop3` 입니다 .
`storeprotocolclass` *`storeprotocolclass`*	저장소용 서비스 제공업체 구현 클래스를 지정합니다 . 이 클래스의 위치 : • `http://java.sun.com/products/javamail/` • `http://java.sun.com/products/javabeans/glasgow/jaf.html`
`transportprotocol` *`transportprotocol`*	메시지를 전송하는 전송 프로토콜 서비스를 지정합니다 .

표 A-12) 전자우편 리소스 생성 매개 변수

매개 변수	값
`transportprotocolclass` *`transportprotocolclass`*	전송용 서비스 제공업체 구현 클래스를 지정합니다 . 이 클래스의 위치 : • `http://java.sun.com/products/javamail/` • `http://java.sun.com/products/javabeans/glassgow/jaf.html`
`enabled` *`true`*/`false`	리소스가 런타임에 사용될 것인지 결정합니다 . 적절한 값은 `on`, `off`, `yes`, `no`, `1`, `0`, `true`, `false` 입니다 .
`desc` *`description`*	리소스에 대한 설명 .
`property` *`propertyname=value`*	사용자 정의 리소스의 등록 정보를 지정하는 이름 - 값 쌍 .

예

```
HttpServerAdmin create -r -mail -jndiname "localmail" -host
localhost -user mailid -from mailid@mailhost -d /opt/Sun/S1WS6.1
-sinst testinstance
```

delete 명령

가상 서버 클래스 , 가상 서버 및 청취 소켓을 삭제하려면 delete 명령을 사용합니다 .

옵션

표에 있는 옵션을 `delete` 명령과 함께 사용하여 클래스를 삭제합니다 .

표 A-13) delete 명령 옵션

옵션	값
-c	지정된 가상 서버 클래스를 삭제합니다 .
-l	지정된 청취 소켓 ID 를 삭제합니다 .
-v	지정된 가상 서버를 삭제합니다 .
-r	지정된 리소스를 삭제합니다 .

클래스 삭제

delete 명령의 이 옵션을 사용하여 가상 서버 클래스를 삭제합니다 .

구문

```
HttpServerAdmin delete -c -cl classname -d server_root -sinst http_instance
```

매개 변수

표에 있는 매개 변수를 `delete` 명령과 함께 사용하여 클래스를 삭제합니다 .

표 A-14) 클래스 삭제 매개 변수

매개 변수	값
-c l *class*	삭제하려는 클래스 이름 .

예

```
HttpServerAdmin delete -c -cl class1 -d /export/sun/server6
-sinst https-sun.com
```

청취 소켓 삭제

delete 명령에 이 옵션을 사용하여 청취 소켓을 삭제합니다 .

구문

```
HttpServerAdmin delete -l -id listen_socket -d server_root -sinst http_instance
```

매개 변수

표에 있는 매개 변수를 `delete` 명령과 함께 사용하여 클래스를 삭제합니다 .

표 A-15) 클래스 삭제 매개 변수

매개 변수	값
-id *listen_socket*	삭제하려는 청취 소켓의 ID.

예

```
HttpServerAdmin delete -l -id ls3 -d /export/sun/server6 -sinst
https-sun.com
```

가상 서버 삭제

delete 명령의 이 옵션을 사용하여 가상 서버를 삭제합니다 .

구문

`HttpServerAdmin delete -v -id` *virtual_server* `-cl` *classname* `-d` *server_root* `-sinst` *http_instance*

매개 변수

표에 있는 매개 변수를 `delete` 명령과 함께 사용하여 가상 서버를 삭제합니다 .

표 A-16) 가상 서버 삭제 매개 변수

매개 변수	값
-id *virtual_server*	삭제하려는 가상 서버 ID.
-cl *class*	가상 서버가 속한 클래스 .

예

```
HttpServerAdmin delete -v -id vs3 -cl class1 -d
/export/sun/server6 -sinst https-sun.com
```

JDBC 연결 풀 삭제

delete 명령에 이 옵션을 사용하여 연결 풀을 삭제합니다 .

구문

`HttpServerAdmin delete -r` *jdbcconnectionpoolname*

매개 변수

표에 있는 매개 변수를 `delete` 명령과 함께 사용하여 연결 풀을 삭제합니다 .

표 A-17) 연결 풀 삭제 매개 변수

매개 변수	값
connectionpoolname	삭제하려는 연결 풀의 이름 .

예

```
HttpServerAdmin delete -r connpool
```

JNDI 자원 삭제

delete 명령에 이 옵션을 사용하여 JNDI 리소스를 삭제합니다 .

구문

`HttpServerAdmin delete -r` *jndiname*

매개 변수

표에 있는 매개 변수를 `delete` 명령과 함께 사용하여 JNDI 리소스를 삭제합니다 .

표 A-18) JNDI 리소스 삭제 매개 변수

매개 변수	값
jndiname	삭제하려는 리소스의 JNDI 이름

예

```
HttpServerAdmin delete -r testresource
```

list 명령

가상 서버 클래스 , 가상 서버 , 청취 소켓 및 리소스의 목록을 표시하려면 `list` 명령을 사용합니다 .

구문

```
HttpServerAdmin list -command_option -d server_root -sinst http_instance
```

옵션

표 A-19) list 명령 옵션

옵션	값
-c	모든 가상 서버 클래스 목록을 표시합니다 .
-l	모든 청취 소켓 목록을 표시합니다 .
-v	모든 가상 서버 목록을 표시합니다 .
-r	지정된 리소스의 목록을 표시합니다 .

예

```
HttpServerAdmin list -c -d /export/sun/server6 -sinst
https-sun.com

HttpServerAdmin list -l -d /export/sun/server6 -sinst
https-sun.com
```

명령 창에 나타나는 정보의 목록을 표시합니다 .

부록 B

Hypertext Transfer Protocol

부록에서는 몇 가지 하이퍼 텍스트 전송 프로토콜 (HTTP) 기본에 대하여 소개합니다 . HTTP 에 대한 자세한 내용은 다음의 Internet Engineering Task Force(IETF) 홈 페이지를 참조하십시오 .

```
http://www.ietf.org/home.html
```

이 부록에서는 다음 단원에 대해 설명합니다 .

하이퍼 텍스트 전송 프로토콜 (HTTP) 설명

하이퍼 텍스트 전송 프로토콜 (HTTP) 은 웹 브라우저와 웹 서버가 ISO Latin1 영문자를 사용하여 서로 ' 통신 ' 할 수 있도록 하는 프로토콜 (네트워크에서 정보를 교환하는 방법을 기술하는 규칙) 입니다 . ISO Latin1 영문자는 유럽 언어용 확장자가 있는 ASCII 입니다 .

HTTP 는 요청 / 응답 모델에 기반합니다 . 클라이언트는 서버에 연결하고 서버로 요청을 보냅니다 . 요청에는 요청 방법 , URI 및 프로토콜 버전이 포함됩니다 . 그런 후 클라이언트는 몇 가지 헤더 정보를 보냅니다 . 서버의 응답에는 프로토콜 버전 , 상태 코드의 반환과 함께 서버 정보가 있는 응답 , 그리고 요청된 데이터가 포함됩니다 . 그런 후 연결이 종료됩니다 .

iPlanet Web Server 4.x 는 HTTP 1.1 을 지원합니다 . 이전 버전의 서버는 HTTP 1.0 을 지원합니다 . 서버는 조건적으로 제안된 HTTP 1.1 표준과 호환되며 이는 IESG(Internet Engineering Steering Group) 및 IETF(Internet Engineering Task Force) HTTP 작업 그룹에서 승인한 것과 같습니다 . 조건적 호환의 범주에 대한 자세한 내용은 IETF 웹사이트에 있는 Hypertext Transfer Protocol-HTTP/1.1 specification(RFC 2068) 을 참조하십시오 .

요청

클라이언트에서 서버로 전송되는 요청에는 다음의 정보가 있습니다 .

- 요청 방법
- 요청 헤더
- 요청 데이터

요청 메소드

클라이언트는 다양한 방법으로 정보를 요청할 수 있습니다 . 보통 사용되는 방법은 다음과 같습니다 .

- GET- 지정된 문서를 요청
- HEAD- 문서의 헤더 정보만 요청
- POST- 서버가 클라이언트에서 GCI 프로그램용 폼 입력 등의 일부 데이터를 수락하도록 요청
- PUT- 서버 문서의 내용을 클라이언트의 데이터로 대체

요청 헤더

클라이언트는 헤더 필드를 서버로 보낼 수 있습니다 . 대부분은 선택입니다 . 몇 가지 흔히 사용되는 요청 헤더는 표에 보이는 것과 같습니다 .

표 B-1) 일반적인 요청 헤더

요청 헤더	설명
Accept	클라이언트가 수락할 수 있는 파일 유형

표 B-1) 일반적인 요청 헤더 *(계속)*

요청 헤더	설명
Authorization	서버에 대하여 클라이언트 자신을 인증하는 경우 사용되며 사용자 이름과 암호 등의 정보 포함
User-agent	클라이언트 소프트웨어의 이름 및 버전 .
Referer	링크에서 사용자가 클릭한 문서의 URL.
Host	요청된 리소스의 인터넷 호스트 및 포트 번호 .

요청 데이터

클라이언트가 POST 또는 PUT 요청을 하는 경우 요청 헤더와 빈 줄 뒤에 데이터를 보낼 수 있습니다 . 클라이언트가 GET 또는 HEAD 요청을 보내는 경우 전송되는 데이터는 없으며 클라이언트는 서버의 응답을 기다립니다 .

응답

서버의 응답에는 다음이 포함됩니다 .

- 상태 코드
- 응답 헤더
- 응답 데이터

상태 코드

클라이언트가 요청을 보내면 서버가 되돌리는 항목 중 하나가 상태 코드로 이는 세 자리 숫자 코드입니다 . 상태 코드에는 네 가지 범주가 있습니다 .

- 100 - 199 범위의 상태 코드는 예비적 응답을 표시합니다 .
- 200 - 299 범위의 상태 코드는 성공적 트랜잭션을 표시합니다 .
- 300 - 399 범위의 상태 코드는 요청된 문서가 이동되어 URL 을 가져올 수 없을 때 반송됩니다 .
- 400 - 499 사이의 상태 코드는 클라이언트에 오류가 있음을 표시합니다 .

- 500 이상 범위의 상태 코드는 서버가 요청을 수행할 수 없거나 오류가 발생했음을 표시합니다 .

표에는 몇 가지 일반적인 상태 코드가 있습니다 .

표 B-2) 일반적인 HTTP 상태 코드

상태 코드	의미
200	OK; 전송 성공 . 오류가 아닙니다 .
302	검색 . 새 URL 로 리디렉션합니다 . 원본 URL 이 이동되었습니다 . 오류가 아니며 , 대부분의 브라우저에서 새 페이지가 표시됩니다 .
304	로컬 사본 사용 . 브라우저의 캐시에 이미 페이지가 있으며 해당 페이지를 다시 요청하는 경우 브라우저에 따라 (Netscape Navigator 등) 브라우저의 캐시된 사본에 있는 " 최종 수정 (last-modified)" 시간 스탬프를 웹 서버로 전달합니다 . 서버에 있는 사본이 브라우저의 사본보다 최신이 아닌 경우 서버는 페이지를 되돌리지 않고 304 코드를 되돌려 불필요한 네트워크 트래픽을 줄입니다 . 오류가 아닙니다 .
401	인증 안 됨 . 사용자가 문서를 요청했으나 유효한 사용자 이름 또는 암호를 제공하지 않았습니다 .
403	금지됨 . 이 URL 은 액세스할 수 없습니다 .
404	없음 . 요청된 문서가 서버에 없습니다 . 이 코드는 서버가 인증 되지 않은 사람에게 해당 문서가 없는 것으로 알리도록 설정된 경우에도 전송됩니다 .
500	서버 오류 . 서버 관련 오류가 발생했습니다 . 서버 관리자는 서버의 오류 로그에서 어떤 일이 발생했는지 확인합니다 .

응답 헤더

응답 헤더에는 서버에 대한 정보와 문서에 대한 정보가 포함됩니다 . 일반적인 응답 헤더는 표에 보이는 것과 같습니다 .

표 B-3) 일반적인 응답 헤더

응답 헤더	설명
Server	웹 서버의 이름 및 버전 .
Date	현재 날짜 (그리니치 표준시).
Last-modified	문서가 마지막으로 수정된 날짜 .

표 B-3) 일반적인 응답 헤더

응답 헤더	설명
Expires	문서가 만기되는 날짜 .
Content-length	이어지는 데이터의 길이 (바이트).
Content-type	이어지는 데이터의 MIME 유형 .
WWW-authenticate	인증 동안 사용되며 클라이언트 소프트웨어에 인증을 위하여 필요한 내용 (사용자 이름 및 암호 등) 을 알리는 정보가 포함됩니다 .

응답 데이터

서버가 마지막 헤더 필드 뒤에 빈 줄을 전송합니다 . 그런 후 문서 데이터를 전송합니다 .

부록 C

ACL 파일 구문

이 부록에서는 ACL(Access-Control List) 과 해당 구문을 설명합니다 . ACL 파일은 텍스트 파일로 웹 서버에 저장된 리소스에 액세스할 수 있는 사용자를 정의한 목록이 있습니다 . 기본적으로 웹 서버에는 서버에 액세스할 수 있는 모든 목록이 포함된 ACL 파일이 하나 있습니다 . 그러나 여러 개의 ACL 파일을 만들고 `obj.conf` 파일에서 이를 참조할 수 있습니다 .

액세스 제어 API 를 사용하여 액세스 제어를 사용자 정의하려는 경우 ACL 파일의 구문과 함수에 대하여 알아야 합니다 . 예를 들어 Oracle 이나 Informix 데이터베이스 등의 다른 데이터베이스와의 인터페이스로 액세스 제어 API 를 사용할 수 있습니다 . API 에 대한 자세한 내용은 다음 사이트의 Sun ONE 설명서를 참조하십시오 .

```
http://docs.sun.com/
```

이 부록에서는 다음 단원에 대해 설명합니다 .

- ACL 파일 구문
- obj.conf 내의 ACL 파일 참조

ACL 파일 구문

모든 ACL 파일은 특정 형식과 구문을 따라야 합니다 . ACL 파일은 하나 이상의 ACL 이 포함된 텍스트 파일입니다 . 모든 ACL 파일은 사용하는 버전 번호로 시작해야 합니다 . 버전 줄은 하나 뿐이며 그 앞에 원하는 만큼의 주석을 삽입할 수 있습니다 . Sun ONE Web Server 6.1 은 버전 3.0 을 사용합니다 . 예 :

```
version 3.0;
```

주석은 줄 앞에 # 기호를 삽입하여 포함합니다 .

파일의 각 ACL 은 해당 유형을 정의하는 정의문으로 시작합니다 . ACL 에 사용할 수 있는 유형은 세 가지입니다 .

- **Path ACL** 은 영향을 미치는 리소스에 대한 절대 경로를 지정합니다 .
- **URI(Uniform Resource Indicator) ACL** 은 서버의 문서 루트에 상대적인 디렉토리 또는 파일을 지정합니다 .
- **Named ACL** 은 `obj.conf` 파일의 리소스에서 참조되는 이름을 지정합니다 . 서버에는 "default" 이름의 리소스가 함께 제공되어 모든 사용자에게 읽기 액세스를 허용하며 LDAP 디렉토리의 사용자에게 쓰기 액세스를 허용합니다 . Sun ONE Web Server 창에서 이름이 지정된 ACL 을 만들 수 있다 하더라도 반드시 이름이 지정된 ACL 을 `obj.conf` 파일의 리소스와 직접 참조해야 합니다 .

Path 와 URI ACL 의 항목 끝에는 와일드카드가 포함될 수 있습니다 . 예 : `/a/b/*`. 항목의 끝이 아닌 다른 곳에 있는 와일드카드는 적용되지 않습니다 .

유형 줄은 `acl` 로 시작하며 유영 정보는 인용부호 안에 포함되고 , 그 뒤에 세미콜론을 넣습니다 . 모든 ACL 의 유형 정보는 서로 다른 ACL 파일이라 할지라도 고유한 이름이어야 합니다 . 다양한 ACL 유형에 대한 예는 다음 줄에 보이는 것과 같습니다 .

```
acl "path=C:/sun/Servers/docs/mydocs/";
acl "default";
acl "uri=/mydocs/";
```

ACL 의 유형을 정의한 후 , ACL 과 함께 사용할 메소드를 정의하는 줄 (인증문) 과 액세스를 허용 또는 거부할 컴퓨터 또는 사용자를 정의하는 줄 (인증문) 을 하나 이상 추가합니다 . 다음에서는 이러한 줄의 구문에 대해 설명합니다 .

이 부분에서는 다음 항목에 대해 설명합니다 .

- 인증 방법
- 권한 부여문
- 기본 ACL 파일

인증 방법

ACL 은 선택적으로 ACL 을 처리할 때 서버가 반드시 사용해야 하는 인증 방법을 지정할 수 있습니다 . 세 가지 방법이 있습니다 .

- Basic (기본값)
- Digest

- SSL

Basic 및 digest 의 경우 사용자가 리소스에 액세스하기 전에 사용자 이름과 비밀 번호를 입력해야 합니다 .

SSL 의 경우 사용자에게 클라이언트 인증서가 있어야 합니다 . 서버의 암호화가 사용되어야 하며 사용자의 인증서 발행자가 인증될 신뢰된 CA 의 목록에 있어야 합니다 .

기본적으로 서버는 방법이 지정되지 않은 ACL 에 대하여 Basic 방법을 사용합니다 . 서버의 인증 데이터베이스가 반드시 사용자가 송신한 다이제스트 인증을 처리할 수 있어야 합니다 .

각 인증 줄은 반드시 서버가 인증할 속성 (사용자 , 그룹 또는 이 둘 모두) 을 지정해야 합니다 . 다음 인증문은 ACL 유형 줄 다음 표시되는 것으로 사용자를 데이터베이스 또는 디렉토리의 개별 사용자와 일치시키는 기본 인증을 지정합니다 .

```
authenticate (user) {
     method = "basic";
};
```

다음 예에서는 SSL 을 사용자 및 그룹용 인증 방법으로 사용합니다 .

```
authenticate (user, group) {
     method = "ssl";
};
```

다음 예에서는 사용자 이름이 `sales` 로 시작하는 모든 사용자를 허용합니다 .

```
authenticate (user)

allow (all)

   user = sales*
```

마지막 줄은 `group=sales` 로 변경하면 그룹 속성이 인증되지 않으므로 ACL 이 실패하게 됩니다 .

권한 부여문

각 ACL 항목에는 하나 이상의 권한 부여문이 있습니다 . 권한 부여문은 서버 리소스에 대한 액세스를 허용 또는 거부할 사용자를 지정합니다 . 권한 부여문을 작성하는 경우 다음 구문을 사용합니다 .

```
allow|deny [absolute] (right[,right...]) attribute expression;
```

각 줄은 allow 또는 deny 로 시작합니다 . 보통 첫 번째 규칙에는 모든 사용자의 액세스를 거부한 후 , 이후의 규칙에 사용자 , 그룹 또는 컴퓨터의 액세스를 구체적으로 지정하는 것이 좋습니다 . 이는 규칙의 계층 때문입니다 . 즉 , `/my_stuff` 라는 디렉토리에 대하여 모든 사용자의 액세스를 허용한 후 , `/my_stuff/personal` 이라는 하위 디렉토리에 대하여 일부 사용자에게만 액세스를 허용하는 경우 하위 디렉토리에 대한 액세스 제어가 제대로 적용되지 않습니다 . 이는 `/my_stuff/` 디렉토리에 액세스가 허용된 모든 사용자가 `/my_stuff/personal` 디렉토리에 액세스할 수 있기 때문입니다 . 이러한 경우를 예방하려면 모든 사용자의 액세스를 거부한 후 일부 사용자에게 액세스를 허용하는 하위 디렉토리용 규칙을 만듭니다 .

그러나 기본 ACL 이 모든 사용자의 액세스를 거부하도록 설정하는 경우 다른 ACL 규칙에 "deny all" 규칙이 필요하지 않은 경우가 있습니다 .

다음 줄은 모든 사용자의 액세스를 거부합니다 .

```
deny (all)
    user = "anyone";
```

이 부분에서는 다음 항목에 대해 설명합니다 .

- 권한 부여문의 계층
- 속성 표현식
- 표현식용 연산자

권한 부여문의 계층

ACL 에는 리소스에 따른 계층이 있습니다 . 예를 들어 서버가 문서 (URI) `/my_stuff/web/presentation.html` 에 대한 요청을 받는 경우 서버는 이 URL 에 적용되는 ACL 목록을 만듭니다 . 서버는 우선 "check-acl" 문에 있는 ACL 목록을 자체의 `obj.conf` 파일에 추가합니다 . 그런 후 , 서버는 일치되는 URI 와 PATH ACL 을 추가합니다 .

서버는 이 목록은 동일한 순서로 처리합니다 . "absolute" ACL 문이 있지 않는 한 모든 줄은 순서 대로 평가됩니다 . "absolute allow" 또는 "absolute deny" 문이 "true" 인 경우 서버는 처리를 중단하고 모든 결과를 승인합니다 .

일치되는 ACL 이 하나 이상인 경우 서버는 일치되는 마지막 줄을 사용합니다 . 그러나 absolute 문을 사용하는 경우 서버는 다른 일치에 대한 조회를 중단하고 absolute 문이 포함된 ACL 을 사용합니다 . 동일한 리소스에 대하여 absolute 문이 둘인 경우 서버는 파일의 첫 번째를 사용하고 일치되는 다른 리소스에 대한 조회를 중단합니다 .

```
version 3.0;
acl "default";
authenticate (user,group) {
     prompt="Web Server";
};
allow (read,execute,list,info)
     user = "anyone";
allow (write,delete)
     user = "all";
acl "uri=/my_stuff/web/presentation.html";
deny (all)
     user = "anyone";
allow (all)
     user = "joe";
```

속성 표현식

속성 표현식은 사용자 이름 , 그룹 이름 , 호스트 이름 또는 IP 주소를 기준으로 허용 또는 거부할 사용자를 정의합니다 . 서로 다른 사용자 또는 컴퓨터의 액세스를 허용하는 예는 다음과 같습니다 .

- `user = "anyone"`
- `user = "smith*"`
- `group = "sales"`
- `dns = "*.sun.com"`
- `dns = "*.sun.com,*.mozilla.com"`
- `ip = "198.*"`
- `ciphers = "rc4`
- `ssl = "on"`

또한 timeofday 속성을 사용하여 하루 중 시간 (서버의 로컬 시간) 에 따라 서버로의 액세스를 제한할 수 있습니다 . 예를 들어 timeofday 속성을 사용하여 특정 사용자가 정해진 시간에만 액세스하도록 제한할 수 있습니다 .

참고 시간을 지정하려면 24 시간 형식을 사용합니다 . 예를 들어 오전 4:00 의 경우에는 0400, 오후 10:30 인 경우에는 2230 을 사용합니다 .

guest 라는 이름의 사용자 그룹이 오전 8:00 에서 오후 4:59 까지 액세스하도록 제한하려면 다음 예제와 같이 합니다 .

```
allow (read)

   (group="guests") and
   (timeofday<0800 or timeofday=1700);
```

또한 주중 요일에 따라 액세스를 제한할 수 있습니다 . Sun, Mon, Tue, Wed, Thu, Fri, Sat 등의 세자리 약자를 사용하여 요일을 지정합니다 .

다음 줄은 premium 그룹의 사용자에게 항상 액세스를 허용합니다 . discount 그룹의 사용자는 주말의 모든 시간과 주 중 오전 8 시 - 오후 4:49 를 제외한 모든 시간에 액세스할 수 있습니다 .

```
allow (read) (group="discount" and dayofweek="Sat,Sun") or
(group="discount" and (dayofweek="mon,tue,wed,thu,fri" and
(timeofday<0800 or timeofday=1700)))
or
(group="premium");
```

표현식용 연산자

속성 표현식에 다양한 연산자를 사용할 수 있습니다 . 괄호는 연산자의 순서를 변경할 때 사용합니다 . user, group, dns, 및 ip 인 경우 다음의 연산자를 사용할 수 있습니다 .

- and
- or
- not
- =(등호)
- !=(부등호)

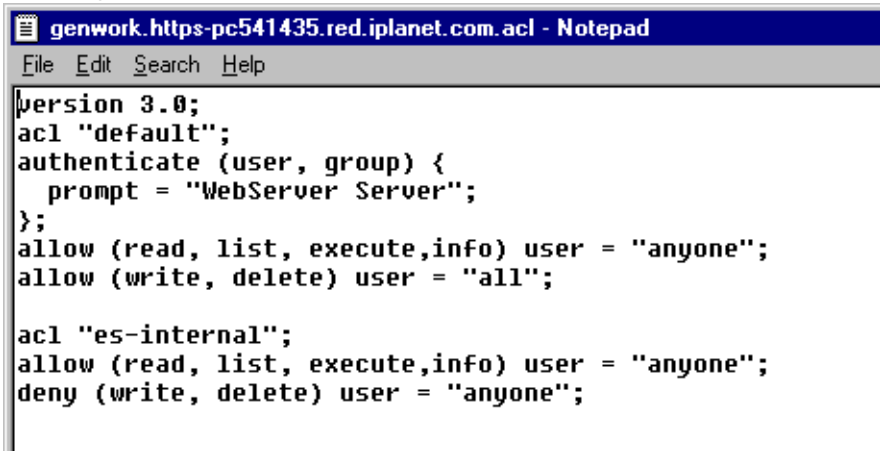
timeofday 및 dayofweek 의 경우 다음을 사용할 수 있습니다 .

- greater than
- < less than
- = 이상
- <= 이하

기본 ACL 파일

설치 후 *server_root*/httpacl/generated.https-*serverid*.acl 파일에 서버용 기본 설정이 제공됩니다 . 사용자 인터페이스에서 설정을 만들 때까지 서버는 작업 파일 genwork.https-*serverid*.acl 을 사용합니다 . ACL 을 파일을 편집할 때 genwork 파일을 변경할 수 있으며 , 그런 후 Sun ONE Web Server 를 사용하여 변경 사항을 저장 및 적용할 수 있습니다 .

genwork 파일



일반 구문 항목

입력 문자열에는 다음 문자를 포함할 수 있습니다 .

- a 에서 z 까지의 문자
- 0 에서 9 까지의 숫자
- 마침표 및 밑줄

다른 문자를 사용하는 경우 문자를 인용부호 (") 안에 넣어야 합니다 .

단일문은 한 줄에 위치해야 하며 세미콜론으로 끝을 표시합니다 . 복수 줄은 대괄호 ([]) 안에 넣습니다 . 항목 목록은 반드시 쉼표로 분리해야 하며 인용부호 (") 안에 넣어야 합니다 .

obj.conf 내의 ACL 파일 참조

이름이 지정된 ACL 이나 별도의 ACL 이 있는 경우 `obj.conf` 파일에서 이를 참조할 수 있습니다 . 이 작업은 `check-acl` 함수를 사용하는 `PathCheck` 지시문에서 수행합니다 . 이 줄의 구문은 다음과 같습니다 .

```
PathCheck fn="check-acl" acl="aclname"
```

`aclname` 은 ACL 파일에 표시되는 ACL 의 고유한 이름입니다 .

예를 들어 ACL 이름 지정 `testacl` 을 사용하여 디렉토리에 대한 액세스를 제한하려면 다음 줄을 `obj.conf` 파일에 추가합니다 .

```
<Object ppath="/usr/ns-home/docs/test/*"
PathCheck fn="check-acl" acl="testacl"
</Object
```

앞의 예에서 첫 번째 줄은 액세스를 제한하려는 서버 리소스를 표시하는 개체입니다 . 두 번째 줄은 `PathCheck` 지시문으로 `check-acl` 함수를 사용하여 이름 지정 ACL(`testacl`) 을 지시문이 나타나는 개체에 바인드합니다 . `testacl` ACL 은 `magnus.conf` 에서 참조하는 모든 ACL 파일에 존재할 수 있습니다 .

부록 D

국제화 및 현지화용 지원

Sun ONE WEB Server 6.1 의 국제화 및 현지화된 버전은 다국어 및 복수 인코딩을 지원합니다 .

주요 기능은 이 부록에서 설명합니다 .

복수 바이트 데이터 입력

Server Manager 또는 Administration Server 페이지에서 복수 바이트 데이터를 입력하려면 다음의 사항에 주의해야 합니다 .

파일 또는 디렉토리 이름

파일 또는 디렉토리 이름이 URL 에 표시되는 경우 8 비트 또는 복수 바이트 문자를 포함할 수 없습니다 .

LDAP 사용자 및 그룹

전자 우편 주소의 경우 RFC 1700(`ftp://ds.internic.net/rfc/rfc1700.txt`) 에서 허용한 문자만 사용합니다 . 사용자 ID 와 암호 정보는 반드시 ASCII 로 저장해야 합니다 .

문자가 사용자 및 그룹용의 올바른 형식으로 입력되도록 하려면 8 비트 또는 복수 바이트 데이터를 입력할 때 UTF-8 형식을 사용할 수 있는 클라이언트 (Netscape Communicator) 를 사용합니다 .

복수 문자 인코딩 지원

Sun ONE Web Server 6.1 에서는 다음 기능에 대한 복수 문자 인코딩을 지원합니다 .

- WebDAV
- 검색

WebDAV

Sun ONE Web Server 6.1 은 PROPPPATCH 및 PROPFIND 메소드에서의 복수 바이트 속성 설정 및 검색을 지원합니다 . 요청의 인코딩 형식은 어느 것이라도 상관 없으나 서버에서의 응답은 항상 UTF-8 입니다 .

검색

Sun ONE Web Server 6.1 에서는 전체 텍스트 색인화와 Java VM 지원에 내재된 모든 문자 인코딩에서 문서를 검색하는 Java 기반 검색 엔진을 지원합니다 . 문서의 기본 인코딩은 검색 컬렉션을 만들 때 지정할 수 있습니다 . HTML 문서의 경우 인덱서 (indexer) 가 HTML 메타태그에서 인코딩을 추출하려 하며 , 추출하지 못하는 경우 기본 인코딩을 사용하도록 되돌아갑니다 .

검색 인터페이스는 JSP 태그 라이브러리를 사용하며 원하는 언어 및 인코딩으로 사용자 정의하거나 현지화할 수 없습니다 . 태그 라이브러리의 목록은 Sun ONE Web Server 6.1 *Programmer's Guide to Web Applications* 에 있습니다 . 더 자세한 내용은 "Search Query 페이지 사용자 정의 " 페이지 402 를 참조하십시오 .

언어 기본 설정

서버 기본 설정의 Magnus Editor 를 사용하여 모든 사용자 오류 메시지에 사용되는 서버용 Default Language 를 설정할 수 있습니다 . 현지화된 Sun ONE Web Server 6.1 에서는 일곱 가지의 언어를 사용할 수 있습니다 .

- en (영어)
- fr (프랑스어)
- de (독일어)
- ja (일본어)
- ko (한국어)
- zh (중국어 간체)
- zh_TW (중국어 번체)

현지화된 버전의 Sun ONE Web Server 6.1 에 있는 사용자 검색 인터페이스는 완전히 현지화되었습니다 .

참고 현지화되지 않은 버전의 웹 서버에서는 이 설정이 적용되지 않습니다 .

현지화된 컨텐트를 서비스하도록 서버 구성

사용자는 브라우저가 Accept-language 헤더를 보내도록 브라우저를 구성할 수 있으며 , 이 헤더에는 액세스하는 컨텐트용 언어 기본 설정이 기술됩니다 . Administration Server 의 Edit Classes 메뉴에 있는 VS 클래스용 `acceptlanguage` 설정을 사용하도록 설정하여 서버가 Accept-language 헤더에 따라 컨텐트를 서비스하도록 구성할 수 있습니다 . 이렇게 하면 사용자 오류 메시지 또한 Accept-language 헤더에 따라 전송됩니다 .

예를 들어 `acceptlanguage` 가 `ON` 으로 설정되고 클라이언트가 값이 `fr-CH, de` 인 Accept-language 헤더를 보내 다음 URL 을 요청할 수 있습니다 .

```
http://www.someplace.com/somepage.html
```

서버는 다음 순서로 파일을 검색합니다 .

1. `Accept-language` 목록 `fr-CH, de`.

   ```
   http://www.someplace.com/fr_ch/somepage.html
   http://www.someplace.com/somepage_fr_ch.html
   http://www.someplace.com/de/somepage.html
   http://www.someplace.com/somepage_de.html
   ```

2. 국가 코드가 없는 언어 코드 (`fr-CH` 의 경우 `fr`):

   `http://www.someplace.com/fr/somepage.html`

   `http://www.someplace.com/somepage_fr.html`

3. `En` 등의 `magnus.conf` 파일에 정의된 `DefaultLanguage`.

   `http://www.someplace.com/en/somepage.html`

   `http://www.someplace.com/somepage_en.html`

4. 위의 내용이 검색되지 않는 경우 서버는 다음을 시도합니다 .

   `http://www.someplace.com/somepage.html`

참고 유의할 점은 국가 코드가 CH 및 TW 등인 현지화된 파일의 이름을 지정하는 경우 소문자로 변환되며 대시 (-) 는 밑줄 (_) 로 변환됩니다 .

주의 `acceptlanguage` 설정을 사용하는 경우 위에 설명한 알고리즘대로 서버가 Accept-language 에 지정된 모든 언어의 컨텐트를 확인하므로 성능의 저하가 발생할 수 있습니다 .

용어

ACE (Access Control Entries) 웹 서버가 입중계 액세스 요청을 평가하는 데 사용하는 규칙의 계층 구조 .

ACL (Access Control List) ACE 를 모아 놓은 것 . ACL 은 서버에 대한 액세스 권한이 있는 사용자를 정의하는 메커니즘이다 . 특정 파일이나 디렉토리 , 또는 하나 이상의 사용자와 그룹에 액세스를 허용 또는 거부하는 ACL 규칙을 정의할 수 있다 .

admpw Enterprise Administrator Server 수퍼유저용 사용자 이름 및 암호 파일 .

에이전트 라우터 , 호스트 또는 X 터미널 등 네트워크 장치에서 네트워크 관리 소프트웨어를 실행하는 소프트웨어 . 지능형 에이전트 참조 .

인증 클라이언트가 서버에 대한 ID 를 확인할 수 있도록 하는 과정 . 기본 또는 출하시 인증의 경우 사용자가 웹 서버 또는 웹 사이트에 액세스하기 위한 사용자 이름과 암호를 입력해야 한다 . LDAP 데이터베이스에 사용자 및 그룹 목록이 있어야 한다 . 다이제스트 및 SSL 인증 참조 .

서버 전체 또는 서버의 특정 파일 및 디렉토리에 액세스할 권한을 부여한다 . 인증은 호스트 이름과 IP 주소를 포함한 범주로 제한할 수 있다 .

브라우저 클라이언트 참조 .

캐시 로컬에 저장된 원본 데이터의 사본 . 캐시된 데이터가 다시 요청되는 경우 원격 서버에서 다시 검색할 필요가 없다 .

인증서 제 3 자가 발행하며 통신하는 양 쪽이 이미 신뢰하는 양도 불가 , 위조 불가 디지털 파일 .

인증기관 (CA) 내부 또는 제 3 자 조직으로 암호화된 트랜잭션용으로 사용되는 디지털 파일을 발행한다 .

인증서 철회 목록 (CRL - Certificate Revocation List) CA 가 제공하는 모든 철회된 인증서에 대한 CA 목록 .

변조된 키 목록 (CKL - Compromised Key List) 키를 조작한 사용자에 대한 주요 정보 목록 . CA 또한 이 목록을 제공한다 .

CGI CGI (Common Gateway Interface). 외부 프로그램이 HTTP 서버와 통신하는 인터페이스 . CGI 사용을 위하여 작성된 프로그램을 CGI 프로그램 또는 CGI 스크립트라 한다 . CGI 프로그램은 폼을 처리하거나 보통의 경우 서버가 처리하거나 파싱하지 않는 출력을 파싱한다 .

chroot 서버를 특정 디렉토리로 제한하기 위하여 만드는 추가의 루트 디렉토리 . 이 기능은 보호되지 않은 서버에 대한 안전 장치로 사용할 수 있다 .

암호 암호는 부호화 알고리즘 (수학적 함수) 으로 암호화 또는 복호화에 사용된다 .

ciphertext 오직 수신자만이 해독할 수 있도록 암호화에 의하여 위장된 정보 .

클라이언트 WWW(World Wide Web) 자료를 요청 및 확인하는 데 사용하는 Netscape Navigator 등의 소프트웨어 . 브라우저 프로그램이라고도 한다 .

클라이언트 인증 클라이언트 인증 .

클러스터 " 마스터 " 및 관리 서버에 추가되고 이에 의하여 제어되는 원격 " 슬레이브 " 관리 서버의 그룹 . 클러스터의 모든 서버는 동일한 플랫폼에 있어야 하며 동일한 사용자 ID 와 암호가 있어야 한다 .

컬렉션 단어 목록과 파일 속성 등 , 문서에 대한 정보가 포함된 데이터베이스 . 검색 기능은 컬렉션을 사용하여 지정된 검색 범주와 일치하는 문서를 검색한다 .

공통 로그파일 형식 서버가 액세스 로그에 정보를 입력하는 용도로 사용하는 형식 . 형식은 Sun ONE Web Server 를 포함하여 모든 주요 서버에 동일하다 .

DHCP 동적 호스트 구성 프로토콜 (Dynamic Host Configuration Protocol). 시스템이 네트워크의 개별 컴퓨터에 동적으로 IP 주소를 지정하도록 하는 IPSP(Internet Proposed Standard Protocol).

데몬 (UNIX) 특정 시스템 작업을 담당하는 이면의 프로세스 .

다이제스트 인증 . 사용자가 명확한 텍스트로 사용자 이름과 암호를 송신하지 않고 인증할 수 있는 인증 방법 . 브라우저는 MD5 알고리즘을 사용하여 다이제스트 값을 만든다 . 서버는 Digest Authentication 플러그 인을 사용하여 클라이언트가 제공한 다이제스트 값을 비교한다 .

DNS 도메인 이름 시스템 . 네트워크의 컴퓨터가 IP 주소 (198.93.93.10 등) 를 호스트 이름 (`www.sun.com` 등) 과 연결하는데 사용하는 체계 . 컴퓨터는 보통 DNS 서버에서 이 변환된 정보를 받거나 자체 시스템에 보관된 테이블에서 이 정보를 검색한다 .

DNS 별칭 특히 NDS CNAME 레코드 등 , DNS 가 다른 호스트에 대한 포인터를 이미 알고 있는 호스트 이름 . 컴퓨터에는 항상 하나의 실제 이름이 있으나 별칭은 하나 이상일 수 있다 . 예를 들어 , `www.` *yourdomain.domain* 과 같은 별칭은 `realthing.`*yourdomain.domain* 과 같은 실제 컴퓨터를 가리킬 수 있으며 , 이는 서버가 현재 존재하는 도메인이다 .

document root 서버에 액세스하는 사용자에게 제시할 파일 , 이미지 및 데이터 등이 들어있는 서버 컴퓨터의 디렉토리 .

drop word stop word 참조 .

암호화 대상의 수신자를 제외한 누구도 해독하거나 읽을 수 없도록 정보를 변환하는 프로세스 .

Administration Server 모든 Sun ONE Web Server 를 구성하는데 사용하는 폼이 들어 있는 웹 기반 서버 .

expires 헤더 반환된 문서의 유효 시간으로 원격 서버가 지정 .

익스트라넷 회사의 인트라넷을 인터넷으로 확장한 것으로 고객 , 공급업체 및 원격 작업자가 데이터에 액세스할 수 있다 .

파일 확장자 파일 이름의 마지막 부분으로 보통 파일의 유형을 정의한다 . 예를 들어 , `index.html` 이라는 파일 이름의 경우 파일 확장자는 `html` 이다 .

파일 유형 파일의 형식 . 예를 들어 , 그래픽 파일의 형식은 텍스트 형식과 다르다 . 파일 유형은 주로 파일 확장자 (`.gif` 또는 `.html` 등) 에 의하여 구분된다 .

방화벽 보통 하드웨어와 소프트웨어 모두를 사용하는 네트워크 구성으로 조직 안에서 네트워크된 컴퓨터를 외부 액세스로부터 보호 . 방화벽은 보통 실제의 빌딩 또는 조직 사이트 내에서 네트워크의 전자우편 및 데이터 파일 등의 정보를 보호하는데 사용된다 .

유연한 로그 형식 서버가 액세스 로그에 정보를 입력하는 용도로 사용하는 형식 .

FORTEZZA 미국 정부가 중요하지만 비밀은 아닌 정보를 관리하는데 사용하는 암호화 시스템 .

FTP 파일 전송 프로토콜 (File Transfer Protocol). 인터넷 프로토콜로 파일이 하나의 컴퓨터에서 네트워크의 다른 컴퓨터로 전송될 수 있도록 한다 .

GIF Graphics Interchange Format. 원래 CompuServe가 개발한 교차 플랫폼 이미지 형식 . GIF 파일은 기타 그래픽 파일 형식 (BMP, TIFF) 보다 훨씬 크기가 작다 . GIF 는 가장 많이 사용되는 교환 형식이다 . GIF 이미지는 UNIX, Microsoft Windows 및 Apple Macintosh 시스템에서 바로 표시할 수 있다 .

하드 리스타트 (hard restart) 프로세스 또는 서비스를 종료시키고 다시 시작하는 과정 . 소프트 리스타트 (soft restart) 참조 .

홈페이지 서버에 존재하는 문서로 서버의 내용에 대한 카탈로그 또는 진입 지점의 역할을 한다 . 이 문서의 위치는 서버의 구성 파일에 정의된다 .

호스트 이름 *machine.domain.dom*의 형식으로 표시되는 컴퓨터의 이름으로 이는 IP 주소로 변환된다 . 예를 들어 , `www.sun.com` 은 `com` 도메인의 `sun` 하위 도메인에 있는 `www` 컴퓨터를 나타낸다 .

HTML Hypertext Markup Language. WWW(World Wide Web) 에 있는 문서에 사용되는 서식 언어 . HTML 파일은 서식 코드가 있는 보통의 텍스트 파일로 Netscape Navigator 등의 브라우저는 서식 코드에 따라 텍스트 표시 , 그래픽 및 양식 항목의 위치 및 다른 페이지로 연결되는 링크 등을 표시한다 .

HTTP HyperText Transfer Protocol. HTTP 서버와 클라이언트 사이에서 정보를 교환하는 메소드 .

HTTP-NG 차세대 HTTP.

HTTPD HTP 데몬 또는 서비스의 약자로 HTTP 프로토콜을 사용하여 정보를 서비스하는 프로그램 . Sun ONE Web Server 는 때로 HTTPD 라고도 한다 .

HTTPS HTTP 의 보안 버전으로 SSL(Secure Sockets Layer) 을 사용하여 구현한다 .

imagemap 이미지의 영역을 활성화시키는 프로세스로 사용자가 마우스를 사용하여 서로 다른 영역을 클릭하면 해당 정보로 이동하거나 정보를 구할 수 있다 . imagemap 은 또한 "imagemap" 이라는 CGI 프로그램을 말하기도 하는데 , 이 경우 다른 HTPD 구현에서 imagemap 기능성을 처리하는데 사용된다 .

inittab (UNIX) UNIX 파일 목록 프로그램으로 어떤 이유이든 정지된 경우 재시작해야 한다 . 이는 프로그램이 지속적으로 실행되도록 한다 . 이 파일의 위치로 인하여 때로 `/etc/inittab` 라고도 한다 . UNIX 시스템에서는 이 파일을 사용할 수 없다 .

지능형 에이전트 사용자를 대신하여 다양한 요청 (HTTP, NNTP, SMTP 및 FTP 요청) 을 수행하는 서버 내의 개체 . 어떤 면에서 지능형 에이전트는 서버에 대하여 서버가 이행하는 요청을 하는 클라이언트의 역할을 한다 .

IP 주소 인터넷 프로토콜 주소 . 마침표로 분리된 일련의 번호로 인터넷에 있는 컴퓨터의 실제 위치를 지정한다 . (예 : 198.93.93.10)

ISDN 종합 정보 통신망 (Integrated Services Digital Network).

ISINDEX 클라이언트에서 검색이 시작되도록 하는 HTML 태그. 문서는 네트워크 네비게이터의 기능을 이용하여 검색 문자열을 받아들이고 이를 서버로 보내어 검색 가능한 색인에 액세스한다 . 이 때 양식은 사용하지 않는다 . <ISINDEX> 를 사용하려면 반드시 쿼리 처리기를 만들어야 한다 .

ISMAP ISMAP 은 HTML 문서에서 사용하는 IMG SRC 의 확장자로 서버에게 해당 이름의 이미지가 imagemap 임을 알려준다 .

ISP 인터넷 서비스 제공자 (Internet Service Provider). 인터넷 연결을 제공하는 단체 .

Java Sun Microsystems 가 개발한 개체 지향형 프로그램 언어로 애플릿이라고 하는 실시간 대화형 프로그램을 만들 때 사용한다 .

JavaScript 클라이언트 및 서버 인터넷 응용 프로그램 개발용의 소형 개체 지향형 스크립트 언어 .

JavaServer Pages 인스턴스화 , 초기화 , 제거 , 기타 구성요소로부터의 액세스 및 구성 관리 등을 포함하여 모든 JavaServer 페이지 메타기능을 사용할 수 있도록 하는 확장 기능 . JSP 는 재사용 가능한 Java 응용 프로그램으로 웹 브라우저가 아닌 웹 서버에서 실행된다 .

Java Servlets 인스턴스화 , 초기화 , 제거 , 기타 구성요소로부터의 액세스 및 구성 관리 등을 포함하여 모든 Java 서브릿 메타기능을 사용할 수 있도록 하는 확장 기능 . Java 서브릿은 재사용 가능한 Java 응용 프로그램으로 웹 브라우저가 아닌 웹 서버에서 실행된다 .

최종 수정된 헤더 서버에서 HTTP 응답으로 반환된 문서 파일의 최종 수정 시간 .

LDAP 데이터베이스 인증용으로 사용자 및 그룹 목록이 저장된 데이터베이스 .

청취 소켓 포트 번호와 IP 주소의 조합 . 서버와 클라이언트 사이의 연결은 청취 소켓에서 이루어진다 .

magnus.conf 기본 Web Server 구성 파일 . 이 파일에는 전역 서버 구성 정보 (포트 , 보안 등) 가 포함된다 . 이 파일은 초기화 동안 서버를 구성하는 변수의 값을 설정한다 . Enterprise Server 는 이 파일을 읽어 시작시 변수 설정을 실행한다 . 서버는 다시 시작할 때까지 이 파일을 다시 읽지 않으므로 이 파일이 변경되는 경우 매번 서버를 다시 시작해야 한다 .

MD5 RSA Data Security 가 개발한 메시지 다이제스트 알고리즘 . MD5 는 고유성과 고도의 이식성을 갖춘 짧은 다이제스트 데이터를 만들 수 있다 . 동일한 메시지 다이제스트 전자우편을 수학적으로 만들어 내는 것은 매우 어려운 작업이다 .

MD5 서명 MD5 알고리즘으로 만든 메시지 다이제스트 .

MIB 관리 정보 베이스 (Management Information Base)

MIME Multi-Purpose Internet Mail Extensions. 멀티미디어 전자우편 및 메시징용으로 새로이 등장하는 표준 .

mime.types MIME(Multi-purpose Internet Mail Extension) 유형 구성 파일 . 이 파일은 파일 확장자와 MIME 유형을 매핑하며 , 이에 따라 서버가 요청되는 컨텐츠의 유형을 결정할 수 있다 . 예를 들어 , 리소스에 대한 요청의 확장자가 `html` 인 경우 클라이언트가 HTML 파일을 요청하는 것이며 , 요청의 확장자가 `.gif` 인 경우에는 클라이언트가 GIF 형식의 이미지 파일을 요청하는 것이다 .

modutil 외부 암호화 또는 하드웨어 가속 장치용으로 PKCS#11 을 설치할 때 필요한 소프트웨어 유틸리티 .

MTA 메시지 전송 에이전트 (Message Transfer Agent). 서버에서 에이전트 서비스를 사용하려면 반드시 서버의 MTA Host 를 정의해야 한다 .

NIS (UNIX) 네트워크 정보 서비스(Network Information Service). UNIX 컴퓨터가 컴퓨터 네트워크 전체에서 컴퓨터 , 사용자 , 파일 시스템 및 네트워크 매개 변수 등에 대한 특정 정보를 수집 , 대조 및 공유하는데 사용하는 프로그램 및 데이터 파일 시스템 .

네트워크 관리 스테이션 (NMS) 사용자가 원격으로 네트워크를 관리할 때 사용하는 컴퓨터 . 관리 대상의 장치는 호스트 , 라우터 및 Sun ONE 서버 등의 SNMP 를 실행하는 모든 것이다 . NMS 는 보통 하나 이상의 네트워크 관리 응용 프로그램이 설치된 고기능 워크스테이션이다 .

NNTP 뉴스그룹용 네트워크 뉴스 전송 프로토콜 (Network News Transfer Protocol). 서버에서 에이전트 서비스를 사용하려면 반드시 뉴스 서버 호스트를 정의해야 한다 .

NSAPI 서버 플러그 인 API 참조 .

obj.conf 서버의 개체 구성 파일 . 이 파일에는 추가의 초기화 정보 , 서버 사용자 정의용 설정 및 서버가 클라이언트 (브라우저 등) 의 요청을 처리할 때 사용하는 지시문 등이 포함된다 . Sun ONE Web Server 는 클라이언트 요청을 처리할 때마다 이 파일을 읽는다 .

암호 파일 (UNIX) UNIX 컴퓨터에 있는 파일로 UNIX 사용자 로그인 이름 , 암호 및 사용자 ID 번호를 저장한다 . 또한 위치에 따라 `etc/passwd` 라고도 한다 .

pk12util 내부 컴퓨터에서 인증서와 키 데이터베이스를 내보내고 이를 외부의 PKCS#11 모듈로 가져오기 위하여 필요한 소프트웨어 유틸리티 .

기본 문서 디렉토리 document root 참조 .

프로토콜 네트워크에 있는 장치가 정보를 교환하는 방법을 기술한 일련의 규칙 .

개인 키 공용 키 암호화에서 사용되는 해독 키 .

공용 키 공용 키 암호화에서 사용되는 암호화 키 .

공용 정보 디렉토리 (UNIX) UNIX 사용자의 홈 디렉토리에서 문서 루트 디렉터리가 아닌 다른 위치의 디렉토리 , 또는 사용자가 제어하는 디렉토리 .

서비스 품질 서버 인스턴스 , 가상 서버 클래스 또는 가상 서버용으로 설정한 성능 한계 .

RAM Random access memory. 컴퓨터에 있는 실제의 반도체 메모리 .

rc.2.d (UNIX) UNIX 에 있는 파일로 컴퓨터가 시작할 때 실행되는 프로그램을 기술한다 . 또한 위치에 따라 `etc/rc.2.d` 라고도 한다 .

재지정 특정 URL 로 액세스하는 클라이언트가 동일한 서버 또는 다른 서버의 다른 위치로 보내지는 시스템 . 이 시스템은 리소스가 이동되었으나 클라이언트가 새 위치를 사용할 때 이를 알 수 없도록 하는 경우에 유용하다 . 또한 디렉터리에 액세스할 때 끝에 슬래시가 없는 경우 관련 링크의 무결성을 유지하는 데에도 사용된다 .

리소스 서버가 액세스하여 이를 요청하는 클라이언트에 전송하는 모든 문서 (URL), 디렉토리 또는 프로그램 .

RFC Request For Comments. 보통 인터넷 커뮤니티로 제출되는 프로시저 또는 표준 문서 . 표준을 수용하기 전에 기술에 대한 의견을 보낼 수 있다 .

root (UNIX) UNIX 컴퓨터에서 가장 권한이 많은 사용자. 루트 사용자는 컴퓨터의 모든 파일에 액세스할 수 있는 완전한 권한을 가진다 .

서버 데몬 일단 실행되면 클라이언트의 요청을 청취하고 수용하는 프로세스 .

서버 플러그 인 API Sun ONE 서버의 핵심 기능을 확장 또는 사용자 정의하고 HTTP 서버와 백엔드 응용 프로그램 사이의 인터페이스를 구축하는 용도의 확장성 및 효율성을 갖춘 메커니즘을 제공하는 확장 기능 . NSAPI 라도고 함 .

서버 루트 . 서버 프로그램 , 구성 , 유지 보수 및 정보 파일 전용으로 할당된 서버 컴퓨터 내의 디렉토리 .

SOCKS 내부에서 외부로의 직접 연결이 방화벽 소프트웨어 또는 하드웨어 (라우터 구성 등) 에 의하여 금지된 경우 방화벽 내부에서 외부로 연결을 설정하는 방화벽 소프트웨어 .

소프트 리스타트 서버를 재시작하는 한 가지 방법으로 서버는 내부적으로 재시작하며 해당 구성 파일을 다시 읽는다 . 소프트 리스타트는 `HUP` 신호 (signal number one) 프로세스를 보낸다 . 하드 스타트와는 달리 프로세스 그 자체는 종료되지 않는다 .

SSL Secure Sockets Layer. HTTP 의 보안 버전인 HTTPS 를 구현할 때 사용되는 양쪽 (클라이언트와 서버) 사이에 안전한 연결을 설정하는 소프트웨어 라이브러리 .

SSL 인증 클라이언트 인증서에 있는 정보를 ID 의 증거로 사용하거나 LDAP 디렉토리에 게시된 클라이언트 인증서를 확인하여 해당 보안 인증서가 있는 사용자의 ID 를 확인한다 .

stop word 검색을 진행하지 않도록 검색 기능에 정의된 단어 . 보통 a, an, and 등의 단어가 포함된다 . 또한 drop word 라고도 한다 .

strftime 날짜와 시간을 문자열로 변환하는 함수 . 서버가 접미부를 추가할 때 사용된다 . `strftime` 에는 날짜 및 시간용 특수 형식 언어가 있으므로 서버는 접미부에서 이를 사용하여 파일의 최종 수정 날짜를 표시한다 .

Sun ONE Web Server Administration Console 이전에는 Netscape Console 이라고 했으며 Java 응용 프로그램으로 서버 관리자에게 기업 네트워크의 임의 지점에서 중앙집중식으로 모든 Sun ONE 서버를 관리할 수 있는 그래픽 인터페이스를 제공한다 . Sun ONE Web Server Administration Console 이 설치된 위치에서 기업 네트워크에 있는 서버 중 액세스 권한이 부여된 모든 Sun ONE 서버를 확인하고 액세스할 수 있다 .

수퍼유저 (UNIX) UNIX 컴퓨터에서 가장 권한이 많은 사용자 (루트라고도 함). 수퍼유저는 컴퓨터의 모든 파일에 액세스할 수 있는 완전한 권한을 가진다 .

Sym-links (UNIX) 심볼 링크의 약자로 UNIX 운영 체제가 사용하는 재지정 유형 중 한 가지 . Sym-link 를 사용하여 파일 시스템의 일정 부분에서 파일 시스템의 다른 부분에 있는 기존 파일 또는 디렉터리로 향하는 포인터를 만들 수 있다 .

TCP/IP Transmission Control Protocol/Internet Protocol. 인터넷 및 기업 네트워크용 기본 네트워크 프로토콜 .

telnet 네트워크에 있는 두 대의 컴퓨터가 서로 연결되고 원격 로그인용 터미널 에뮬레이션을 지원하는 프로토콜 .

시간 초과 서버가 고착된 것으로 보이는 서비스 루틴에 대한 시도를 포기하도록 지정된 시간 .

TLS Secure Sockets Layer. HTTP 의 보안 버전인 HTTPS 를 구현할 때 사용되는 양쪽 (클라이언트와 서버) 사이에 안전한 연결을 설정하는 소프트웨어 라이브러리 .

top (UNIX) 일부 UNIX 시스템에 있는 프로그램으로 시스템 리소스의 현재 사용 상태를 표시 .

최상위 도메인 권한 호스트 이름 분류에서 최상위 분류로 보통 도메인의 조직 형태 (예 : `com` 은 회사 , `edu` 는 교육 기관 등) 또는 원래의 국가 (예 : `.us` 는 미국 , `.jp` 는 일본 , `.au` 는 호주 , `.fi` 는 핀란드 등) 를 표시한다 .

uid (UNIX) UNIX 시스템의 각 사용자에 연결된 고유 번호 .

URI Uniform Resource Identifier. 단축 URL 을 사용함으로써 추가의 보안을 제공하는 파일 ID. URL 의 첫 번째 부분은 URL 매핑으로 대체되므로 사용자는 파일의 실제 경로를 알 수 없게 된다 . URL 매핑 참조 .

URL Uniform Resource Locator. 서버와 클라이언트가 문서를 요청할 때 사용하는 주소 지정 체계 . URL 은 또한 위치라고도 함 . URL 의 형식은 *protocol://machine:port/document* 이다 .

예제 URL: `http://www.sun.com/index.html`.

URL 데이터베이스 수정 소프트웨어 장애 , 시스템 고장 , 디스크 손상 또는 파일 시스템의 사용 공간 부족 등으로 손상이 발생한 경우 URL 데이터베이스를 수리하고 업데이트하는 프로세스 .

URL 매핑 문서 디렉토리의 실제 경로를 사용자 정의 별칭으로 매핑하여 디렉토리에 있는 파일이 파일의 실제 경로 이름이 아닌 디렉토리의 별칭으로 액세스되도록 하는 프로세스 . 따라서 파일을 `usr/sun/servers/docs/index.html` 로 지정하는 것이 아니라 `/myDocs/index.html` 로 지정할 수 있다 . 사용자가 서버 파일의 실제 위치를 알 필요가 없으므로 추가의 보안 기능을 제공한다 .

가상 서버 클래스 obj.conf 파일의 동일한 기본 구성을 공유하는 가상 서버의 집합 .

가상 서버 설치된 하나의 서버에서 여러 개의 도메인 이름 , IP 주소 및 서버 모니터 기능을 설정하는 한 가지 방법 .

웹 응용 프로그램 서브릿 , JSP(JavaServer Page), HTML 문서 및 기타 웹 리소스의 집합으로 여기에는 이미지 파일 , 압축된 자료 및 기타 데이터가 포함될 수 있다 . 웹 응용 프로그램은 저장 파일로 패키지화 되거나 (WAR 파일) 또는 개방형 디렉토리 구조로 존재할 수 있다 .

WAR(Web Application Archive) 완전한 웹 응용 프로그램을 압축된 형태로 포함하는 보관 파일 . Sun ONE Web Server 는 WAR 파일 내의 응용 프로그램에는 액세스할 수 없다 . Sun ONE Web Server 가 서비스하도록 하려면 반드시 해당 웹 응용 프로그램의 압축을 해제해야 한다 (`wdeploy` 유틸리티 사용).

Windows CGI (Windows) Visual Basic 등의 Windows 기반 프로그램 언어로 작성된 CGI 프로그램 .

색인

기호

숫자

A

B

C

D

E

F

G

H

I

J

K

L

M

N

O

P

Q

R

S

T

U

V

W

X

Z