

# 설치 및 조정 설명서

## *Sun™ ONE Directory Server*

**버전 5.2**

816-6850-10  
2003년 6월

Copyright © 2003 Sun Microsystems, Inc., 4150 Network Circle, Santa Clara, California 95054, U.S.A. 모든 권리는 저작권자의 소유입니다.

미국 정부의 권리 - 상용 소프트웨어. 정부 사용자는 Sun Microsystems, Inc.의 표준 사용권 계약과 해당 FAR 규정 및 보충 규정을 준수해야 합니다. 본 설명서에는 타사에서 개발한 내용이 포함될 수 있습니다. 제품 중에는 캘리포니아 대학에서 허가한 Berkeley BSD 시스템에서 파생된 부분이 포함되어 있을 수 있습니다. UNIX는 미국 및 다른 국가에서 X/Open Company, Ltd를 통해 독점적으로 사용권이 부여되는 등록 상표입니다. Sun, Sun Microsystems, Sun 로고, Java, Solaris, SunTone, Sun[tm] ONE, The Network is the Computer, SunTone Certified 로고 및 Sun[tm] ONE 로고는 미국 및 다른 국가에서 Sun Microsystems, Inc.의 상표 또는 등록 상표입니다. SPARC 상표는 사용 허가를 받았으며 미국 및 다른 국가에서 SPARC International, Inc.의 상표 또는 등록 상표입니다. SPARC 상표를 사용하는 제품은 Sun Microsystems, Inc.에서 개발한 구조에 기반을 두고 있습니다. Mozilla, Netscape 및 Netscape Navigator는 미국 및 다른 국가에서 Netscape Communications Corporation의 상표 또는 등록 상표입니다. 이 서비스 설명서에서 다루는 제품과 수록된 정보는 미국 수출 관리법이 적용되며 다른 국가의 수출입법이 적용될 수 있습니다. 이 제품과 정보를 직간접적으로 핵무기, 미사일 또는 생화학 무기에 사용하거나 핵과 관련하여 해상에서 사용하는 것은 엄격하게 금지됩니다. 미국 수출 금지 국가 또는 금지된 개인과 특별히 지정된 국민 목록을 포함하여 미국 수출 금지 목록에 지정된 대상으로의 수출이나 재수출은 엄격하게 금지됩니다.

이 문서에서는 본문의 내용을 "있는 그대로" 제공하며, 법률을 위반하지 않는 범위 내에서 상품성, 특정 목적에 대한 적합성 또는 비침해에 대한 묵시적인 보증을 포함하여 모든 명시적 또는 묵시적 조건, 표현 및 보증을 배제합니다.



# 목차

<b>본 설명서 정보</b> .....	<b>9</b>
본 설명서의 목적 .....	9
필수 사항 .....	9
표기 규칙 .....	10
기본 경로 및 파일 이름 .....	10
Directory Server 도구 다운로드 .....	12
관련 자료 .....	12
<b>1부 설치</b> .....	<b>15</b>
<b>1장 Sun ONE Directory Server 설치</b> .....	<b>17</b>
시작 전 주의 사항 .....	17
디렉토리 배포 계획 .....	17
Directory Server 소프트웨어 다운로드 .....	19
설치 .....	19
설치할 소프트웨어 결정 .....	20
설치 방법 결정 .....	20
설치 정보 준비 .....	21
Solaris 시스템에 설치 .....	23
기타 UNIX 시스템에 설치 .....	34
Windows 시스템에 설치 .....	38
제거 .....	41
Solaris 시스템에서 제거 .....	42
기타 UNIX 시스템에서 제거 .....	44
Windows 시스템에서 제거 .....	45
문제 해결 .....	46

<b>2장 이전 버전에서 업그레이드</b> .....	<b>51</b>
업그레이드 시 주의 사항 .....	51
단일 서버 인스턴스 업그레이드 .....	51
여러 개의 복제된 서버 업그레이드 .....	53
업그레이드 지원 .....	54
단일 서버 업그레이드 .....	54
새 서버 설치 .....	54
(4.x를 5.2로 업그레이드하는 경우) 사용자 정의 스키마 처리 .....	55
기존 데이터 이전 .....	55
(4.x를 5.2로 업그레이드하는 경우에만 해당) 복제 계약 작성 .....	56
(선택 사항) 기존 포트 번호 다시 사용 .....	57
(4.x를 5.2로 업그레이드하는 경우) 복제된 서버 업그레이드 .....	57
새 마스터 준비 .....	57
소비자 업그레이드 .....	58
분기 업그레이드 .....	58
서버 추가 .....	59
4.x 업그레이드 시나리오 예 .....	60
(5.x를 5.2로 업그레이드하는 경우) 복제된 서버 업그레이드 .....	66
5.x 서버 업그레이드 .....	67
서버 추가 .....	67
5.x 업그레이드 시나리오 예 .....	67

**2부 조정** .....

**71**

**3장 유용한 조정 팁** .....

**73**

**4장 하드웨어 크기 조정** .....

**79**

최소 요구 사항 .....	79
사용 가능한 최소 메모리 .....	80
최소 로컬 디스크 공간 .....	80
최소 처리 능력 .....	81
최소 네트워크 용량 .....	81
물리적 메모리 크기 조정 .....	81
Directory Server 메모리 크기 조정 .....	82
운영 체제 메모리 크기 조정 .....	83
총 메모리 크기 조정 .....	84
메모리가 부족한 경우의 대처 방법 .....	84
디스크 하위 시스템 크기 조정 .....	85
디렉토리 접미사 크기 조정 .....	85
Directory Server의 디스크 사용 방법 .....	86
디스크에 파일 분산 .....	88

디스크 하위 시스템 대체 방안 .....	90
입출력 및 디스크 사용 모니터 .....	94
다중 프로세서 시스템 크기 조정 .....	94
네트워크 용량 크기 조정 .....	94
SSL 크기 조정 .....	94

<b>5장 운영 체제 조정 .....</b>	<b>97</b>
플랫폼 지원 확인 .....	97
시스템 패치 설치 .....	98
기본 보안 강화 .....	98
시스템 격리 .....	98
이중 부트 금지 .....	98
강력한 암호 .....	99
(Windows) 로컬 보안 정책 .....	99
(UNIX 플랫폼) 사용자 및 그룹 .....	99
불필요한 서비스 비활성화 .....	100
정확한 시간 유지 .....	100
시스템 장애 후 다시 시작 .....	100
권장되는 기본 조정 작업 생성 .....	101
시스템 설정 조정 .....	102
(Windows) 지연된 프로세서 호출(DPC) .....	102
파일 설명자 .....	103
(HP-UX) 큰 파일 지원 .....	103
(HP-UX) 스레드 보류 시간 제한 .....	103
(HP-UX) 프로세스 당 스레드 .....	103
TCP(Transmission Control Protocol) 설정 .....	104

<b>6장 캐시 크기 조정 .....</b>	<b>107</b>
캐시 종류 .....	107
데이터베이스 캐시 .....	108
항목 캐시 .....	109
가져오기 캐시 .....	110
파일 시스템 캐시 .....	111
총 캐시 크기 .....	111
검색 시 캐시 사용 방법 .....	112
업데이트 시 캐시 사용 방법 .....	115
접미사 초기화 시 캐시 사용 방법 .....	116
검색 최적화 .....	118
메모리에 모든 항목과 색인 저장 .....	118
충분한 메모리, 32비트 Directory Server .....	120
메모리 축소, 일부 파일 시스템 캐시 사용 .....	120
메모리 축소, 파일 시스템 캐시 축소 .....	121

업데이트 최적화 .....	121
캐시 초기화 및 모니터 .....	122
기타 최적화 .....	123

**7장 색인화 조정 ..... 125**

색인 .....	125
이점: 검색 시 색인의 역할 .....	126
손실: 업데이트 시 색인에 미치는 영향 .....	127
실제 색인 .....	128
동일 색인 .....	129
하위 문자열 색인 .....	130
찾아보기(가상 목록 보기) 색인 .....	131
근사 색인 .....	132
국가별 색인 .....	132
예제: 항목 색인화 .....	132
성능 향상을 위한 색인화 조정 .....	134
색인화된 검색만 허용 .....	134
색인 목록 길이 제한 .....	135
색인 단편화 문제 해결 .....	138

**8장 로깅 조정 ..... 139**

액세스 로깅 .....	140
감사 로깅 .....	142
오류 로깅 .....	143
다중 마스터 복제 변경 로깅 .....	145
역 변경 로깅 .....	146
트랜잭션 로깅 .....	147

**9장 기타 자원의 사용 관리 ..... 149**

클라이언트에서 사용할 수 있는 자원 제한 .....	149
사용할 수 있는 시스템 자원 사용 .....	152
액세스 제어 관리 .....	155
서버 플러그 인 구성 .....	156

**부록 A 설치된 제품 레이아웃 ..... 159**

ServerRoot 디렉토리 .....	159
서버 인스턴스 디렉토리 .....	162
내부용으로만 사용 .....	164

**부록 B Sun Crypto Accelerator 보드 사용 ..... 165**

시작 전 주의 사항 .....	165
------------------	-----

토큰 작성 .....	166
보드 바인드 생성 .....	167
인증서 가져오기 .....	168
SSL 구성 .....	168
<b>부록 C Directory Server용 Sun Cluster HA 설치 .....</b>	<b>171</b>
시작 전 주의 사항 .....	171
네트워크 자원 설정 .....	172
서버 설치 .....	174
활성 노드에 설치 .....	174
다른 노드에 설치 .....	175
데이터 서비스 패키지 설치 .....	175
서버 구성 .....	176
등록 및 구성 예제 .....	177
확장 등록정보 구성 .....	178
구성 가능한 등록정보 .....	178
장애 모니터 작동 방식 .....	179
HA 저장소 및 데이터 서비스 동기화 .....	180
추가 Directory Server 인스턴스 작성 .....	181
제거 .....	182
<b>색인 .....</b>	<b>185</b>





# 본 설명서 정보

Sun™ ONE Directory Server 5.2는 업계 표준인 LDAP(Lightweight Directory Access Protocol)를 기초로 하는 강력하고 확장성 있는 분산 디렉토리 서비스입니다. 또한 Sun ONE Directory Server 소프트웨어는 주문형 서비스 구현 및 배포에 관한 Sun의 표준 기반 소프트웨어 비전, 아키텍처, 플랫폼 및 전문 지식인 Sun ONE(Sun Open Net Environment)의 한 구성 요소입니다.

Sun ONE Directory Server는 중앙에서 관리되는 분산형 데이터 저장소를 구축할 수 있는 획기적인 소프트웨어입니다. 이러한 데이터 저장소는 주로 인트라넷에서 내부 업무용으로, 엑스트라넷을 통해 파트너와의 거래용으로 또는 공공 인터넷을 통해 고객 도달 범위를 확대하는 데 사용됩니다.

## 본 설명서의 목적

본 설명서는 생산 환경에서 Directory Server를 설치하는 방법에 대해 설명합니다. Directory Server가 생산 환경에서 뛰어난 성능을 발휘하도록 하기 위해서는 상당량의 구성 및 조정 작업이 필요합니다.

생산 환경에서 사용하지 않고 테스트용으로 Directory Server를 설치하는 경우에는 1장, "Sun ONE Directory Server 설치"만 읽어 보셔도 됩니다.

## 필수 사항

생산 환경에서의 사용을 위해 Directory Server를 설치하는 경우에는 먼저 배포 목적을 명확히 이해해야 합니다. 자세한 내용은 *Sun ONE Directory Server Deployment Guide*를 참조하십시오.

## 표기 규칙

이 절에서는 본 설명서에 사용된 표기 규칙에 대해 설명합니다.

고정 폭 글꼴 - 텍스트에 표시되는 속성과 개체 클래스 이름 등 리터럴 텍스트에 사용됩니다. 이 글꼴은 URL, 파일 이름, 예 등에도 사용됩니다.

*기울임꼴 글꼴* - 강조, 새 용어, 경로 이름의 자리 표시자 등 실제 값으로 대체해야 하는 텍스트에 사용됩니다.

보다 큼 기호(>)는 메뉴 또는 하위 메뉴 항목의 이름을 지정할 때 구분 기호로 사용됩니다. 예를 들어, "개체 > 새로 만들기 > 사용자"는 "개체" 메뉴의 "새로 만들기" 하위 메뉴에 있는 "사용자" 항목을 선택해야 한다는 의미입니다.

---

**주** 주, 주의, 팁은 중요한 조건이나 제한 사항을 강조합니다. 먼저 이 정보를 읽은 후에 계속 진행하십시오.

---

## 기본 경로 및 파일 이름

Sun ONE Directory Server 제품 설명서에서 예로 제시된 모든 경로 및 파일 이름은 다음 두 가지 형식 중 하나를 사용합니다.

- *ServerRoot* / ... - *ServerRoot*는 Sun ONE Directory Server 제품이 설치된 위치입니다. 이 경로에는 Directory Server, Sun ONE 관리 서버 및 명령줄 도구에서 공유하는 이진 파일이 저장되어 있습니다.

실제 *ServerRoot* 경로는 사용하는 플랫폼, 설치 및 구성에 따라 달라집니다. 기본 경로는 표 1에 표시된 것처럼 제품 플랫폼과 패키지에 따라 달라집니다.

- *ServerRoot* / *slapd-serverID* / ... - *serverID*는 설치 또는 구성 중에 정의한 Directory Server 인스턴스의 이름입니다. 이 경로에는 해당 인스턴스에 고유한 데이터베이스 및 구성 파일이 저장되어 있습니다.

---

**주** 본 설명서에서 지정된 경로는 UNIX의 슬래시 형식을 사용하며, 명령은 파일 확장명 없이 지정됩니다. Sun ONE Directory Server의 Windows 버전을 사용하는 경우 이와 동격인 역슬래시 형식을 사용하십시오. Windows 시스템에서 실행 파일은 일반적으로 .exe 또는 .bat 확장명을 사용합니다.

---

**표 1** *ServerRoot* 기본 경로

제품 버전	<i>ServerRoot</i> 경로
Solaris 패키지 <sup>1</sup>	<p><code>/var/mps/serverroot</code> - 구성 후에 이 디렉토리는 다음 위치에 대한 링크가 포함됩니다.</p> <ul style="list-style-type: none"> <li><code>/etc/ds/v5.2</code>(정적 구성 파일)</li> <li><code>/usr/admserv/mps/admin</code>(Sun ONE 관리 서버 이진 파일)</li> <li><code>/usr/admserv/mps/admin</code>(서버 콘솔 이진)</li> <li><code>/usr/ds/v5.2</code>(Directory Server 이진)</li> </ul>
Solaris 및 기타 UNIX 시스템에서의 압축된 아카이브 설치	<code>/var/Sun/mps</code>
Windows 시스템에서의 Zip 설치	<code>C:\Program Files\Sun\MPS</code>

1. Solaris Operating Environment에서 작업 중이고 Sun ONE Directory Server 소프트웨어의 어떤 버전이 설치되어 있는지 확실하지 않으면 `pkginfo` 명령을 사용하여 `SUNWdsvu`와 같은 주요 패키지가 있는지 확인하십시오. 예를 들어 `pkginfo | grep SUNWdsvu`와 같이 입력합니다.

Directory Server 인스턴스는 `ServerRoot/slapd-serverID/`에 위치해 있습니다. 여기서 `serverID`는 처음 인스턴스를 작성할 때 지정한 서버 식별자를 나타냅니다. 예를 들어, Directory Server 이름을 `dirserv`라고 지정하면 실제 경로는 표 2와 같이 표시됩니다. 다른 위치에 Directory Server 인스턴스를 작성한 경우 해당 위치에 따라 경로를 수정하십시오.

**표 2** `dirserv` 예제 인스턴스의 기본 위치

제품 버전	인스턴스 위치
Solaris 패키지	<code>/var/mps/serverroot/slapd-dirserv</code>
Solaris 및 기타 UNIX 시스템에서의 압축된 아카이브 설치	<code>/usr/Sun/mps/slapd-dirserv</code>
Windows 시스템에서의 Zip 설치	<code>C:\Program Files\Sun\MPS\slapd-dirserv</code>

## Directory Server 도구 다운로드

일부 지원 플랫폼에서는 Directory Server에 대한 고유 액세스 도구를 제공합니다. LDAP 디렉토리 서버용 추가 테스트 및 유지관리 도구를 원할 경우 Sun ONE Directory Server Resource Kit(DSRK)를 다운로드하십시오. 이 소프트웨어는 아래 URL에서 구할 수 있습니다.

<http://www.sun.com/software/download/>

DSRK 도구에 대한 설치 지침 및 참조 설명서는 *Sun ONE Directory Server Resource Kit Tools Reference*에서 제공합니다.

디렉토리 클라이언트 응용 프로그램을 개발하려면 위의 URL에서 iPlanet Directory SDK for C 및 iPlanet Directory SDK for Java도 다운로드할 수 있습니다.

이외에도 JNDI(Java Naming and Directory Interface) 기술을 통해 Java 응용 프로그램에서 LDAP 및 DSML v2를 사용하여 Directory Server에 액세스할 수 있습니다. JNDI에 대한 자세한 내용은 아래 URL을 참조하십시오.

<http://java.sun.com/products/jndi/>

JNDI Tutorial에는 JNDI 사용법에 대한 자세한 설명과 예가 수록되어 있으며 아래 URL에서 사용할 수 있습니다.

<http://java.sun.com/products/jndi/tutorial/>

## 관련 자료

Sun ONE Directory Server 제품 설명서는 다음과 같이 HTML 및 PDF로 제공되는 다양한 문서로 구성되어 있습니다.

- *Sun ONE Directory Server Getting Started Guide* - Directory Server 5.2의 다양한 주요 기능을 쉽게 확인할 수 있습니다.
- *Sun ONE Directory Server Deployment Guide* - 디렉토리 토폴로지, 데이터 구조, 보안 및 모니터링의 계획 방법을 설명하고 배포 과정에 대한 예를 소개합니다.
- *Sun ONE Directory Server 설치 및 조정 설명서* - 설치 및 업그레이드 절차를 설명하고 Directory Server 성능을 최적화하기 위한 팁을 제공합니다.
- *Sun ONE Directory Server 관리 설명서* - 콘솔과 명령줄을 사용하여 디렉토리 내용을 관리하고 Directory Server의 모든 기능을 구성하는 절차를 소개합니다.

- *Sun ONE Directory Server Reference Manual* - Directory Server 구성 매개 변수, 명령, 파일, 오류 메시지 및 스키마에 대해 자세히 설명합니다.
- *Sun ONE Directory Server Plug-In API Programming Guide* - Directory Server 플러그인 개발 방법을 보여줍니다.
- *Sun ONE Directory Server Plug-In API Reference* - Directory Server 플러그인 API의 데이터 구조와 기능에 대해 자세히 설명합니다.
- *Sun ONE Server Console Server Management Guide* - Sun ONE 관리 서버와 Java 기반의 콘솔을 사용하여 서버를 관리하는 방법에 대해 설명합니다.
- *Sun ONE Directory Server Resource Kit Tools Reference* - 많은 유용한 도구 등 Sun ONE Directory Server Resource Kit의 설치 및 기능에 대해 설명합니다.

아래 웹 사이트에서도 기타 유용한 정보를 확인할 수 있습니다.

- 온라인 제품 설명서: [http://docs.sun.com/coll/S1\\_DirectoryServer\\_52](http://docs.sun.com/coll/S1_DirectoryServer_52)
- Sun 소프트웨어: <http://www.sun.com/software/>
- Sun ONE 서비스: <http://www.sun.com/service/sunps/sunone/>
- Sun 지원 서비스: <http://www.sun.com/service/support/>
- 개발자용 Sun ONE: <http://sunonedev.sun.com/>
- 교육: <http://suned.sun.com/>

관련 자료

# 설치

1장, "Sun ONE Directory Server 설치"

2장, "이전 버전에서 업그레이드"

부록 A, "설치된 제품 레이아웃"

부록 B, "Sun Crypto Accelerator 보드 사용"

부록 C, "Directory Server용 Sun Cluster HA 설치"





# Sun ONE Directory Server 설치

이 장에서는 Sun ONE Directory Server 소프트웨어의 초기 설치 및 제거 과정을 안내하며, 다음 내용을 설명합니다.

- 시작 전 주의 사항
- 설치
- 제거
- 문제 해결

## 시작 전 주의 사항

생산 환경에 Directory Server를 설치하려면 디렉토리 서비스를 실행하는 데 필요한 최소한의 장비가 장착되어 있으며 시스템이 올바르게 구성되어 있어야 합니다. 그리고 최소한 *Sun ONE Directory Server Deployment Guide*에 설명된 개념에 대해서는 명확히 이해해야 합니다.

---

**주**                    최적의 성능을 얻으려면 본 설명서의 조정 및 구성 지침을 따르십시오.

---

## 디렉토리 배포 계획

기본 플랫폼 관련 작업은 해당 운영 체제의 설명서를 참조하여 다음 단계를 수행합니다.

1. 디렉토리 서비스 배포를 계획합니다.

자세한 내용은 *Sun ONE Directory Server Deployment Guide*를 참조하십시오.

2. 여러 디렉토리 설치의 서버 구성, 사용자 및 그룹을 중앙 집중식으로 관리할 수 있도록 배포하려면 구성 및 사용자 디렉토리 위치를 결정합니다.

*구성 디렉토리* 또는 구성 Directory Server(CDS)에는 Directory Server 자체의 구성 방법에 대한 정보가 저장됩니다. 일반적으로 이 디렉토리가 먼저 설치되고 이후의 모든 서버는 이 디렉토리에 등록되기 때문에 단일 구성 디렉토리에서 모든 서버를 중앙 집중식으로 관리할 수 있습니다.

*사용자 디렉토리*에는 디렉토리 서비스를 액세스하는 사용자 및 그룹 항목이 저장됩니다. 일반적으로 네트워크 도메인에는 하나의 사용자 디렉토리가 존재하며 이 디렉토리에 액세스하여 사용자 및 그룹 정보를 얻으므로 단일 사용자 디렉토리에서 모든 사용자 및 그룹을 중앙 집중식으로 관리할 수 있습니다.

소규모 배포의 경우 구성 디렉토리, 사용자 디렉토리 및 기타 디렉토리를 모두 디렉토리 인스턴스 하나에 설치할 수도 있습니다. 대규모 배포의 경우에는 구성 디렉토리와 사용자 디렉토리를 개별 서버에 설치하는 것이 좋습니다.

구성, 사용자 및 그룹 데이터의 적절한 위치에 대한 자세한 내용은 *Sun ONE Server Console Server Management Guide*를 참조하십시오.

3. 표 1-1에 요약된 것처럼 호스트 시스템이 지원되는 구조에서 지원되는 플랫폼을 실행하는지 확인합니다.

**표 1-1** 지원되는 플랫폼 및 구조

플랫폼	구조
Sun Solaris™ Operating Environment 9	SPARC® 프로세서, 32비트 및 64비트 모드  지원되는 x86 플랫폼
Sun Solaris Operating Environment 8	UltraSPARC 프로세서, 32비트 및 64비트 모드
Sun Linux 5.0	Sun LX50 서버
Hewlett Packard HP-UX 11i	PA-RISC 2.0 프로세서, 32비트 및 64비트 모드
IBM AIX 5.1	PowerPC 프로세서
Microsoft Windows 2000 Server, SP 3	Pentium II 이상 IA-32 프로세서
Microsoft Windows 2000 Advanced Server, SP 3	
Red Hat Linux 7.2	Pentium II 이상 IA-32 프로세서

4. 80페이지의 표 4-1에 간략히 요약된 것처럼 호스트 시스템이 최소 디스크 공간과 메모리 요구 사항을 충족시키는지 확인합니다.
5. 호스트 시스템에 대한 물리적 액세스를 제한합니다.
6. 호스트 시스템이 정적 IP 주소를 사용하는지 확인합니다.
7. Directory Server 인스턴스가 네트워크에 이름 지정 서비스를 제공하지 않거나 배포 계획에 Directory Server의 원격 관리가 포함되어 있으면 이름 지정 서비스와 호스트의 도메인 이름이 올바르게 구성되어 있는지 확인합니다.

## Directory Server 소프트웨어 다운로드

17페이지의 "디렉토리 배포 계획"에 요약된 절차를 완료한 후 다음 단계를 수행합니다.

1. 소프트웨어 압축을 풀 수 있도록 압축 해제 유틸리티가 설치되어 있는지 확인합니다.
2. 소프트웨어를 다운로드합니다. 본 설명서를 작성 중인 현 시점에서는 아래 URL에서 다운로드할 수 있습니다.

<http://www.sun.com/software/download/>

3. Directory Server를 설치할 디렉토리 이외의 디렉토리에 소프트웨어 압축을 풉니다.

## 설치

수행하는 Directory Server 설치 단계는 특정 배포 요구 사항에 따라 달라집니다. 이러한 특정 배포 요구 사항을 고려하여 해당 절의 설명에 따라 설치하십시오.

- 설치할 소프트웨어 결정
- 설치 방법 결정
- 설치 정보 준비
- Solaris 시스템에 설치
- 기타 UNIX 시스템에 설치
- Windows 시스템에 설치

## 설치할 소프트웨어 결정

설치할 소프트웨어를 결정하기 전에 확인해야 할 많은 대안이 있습니다. 다음과 같은 질문을 가정해 보십시오.

- 대규모 배포를 위한 큰 캐시 기능이 필요한가?

그렇다면 Directory Server를 64비트 프로세스로 실행할 수 있는 플랫폼을 사용하여 64비트 버전을 설치하는 것이 좋습니다.

Directory Server 배포의 규모가 비교적 작아서 데이터베이스 크기가 500MB 미만인 경우에는 64비트 버전을 지원하는 플랫폼에도 32비트 지원만 설치합니다.

- 그래픽 사용자 인터페이스를 통해 Directory Server를 관리할 계획인가?

그렇다면 Sun ONE 서버 콘솔과 Sun ONE Administration Server를 설치합니다.

명령줄 인터페이스를 통해서만 Directory Server를 관리할 계획이면 콘솔과 관리 서버를 설치하지 않도록 선택할 수 있습니다.

그래픽 사용자 인터페이스를 통해 시스템에서 원격 관리를 수행하려면 콘솔과 관리 서버만 설치하도록 선택할 수도 있습니다.

- Sun Cluster 소프트웨어에 Directory Server를 배포할 계획인가?

그렇다면 자세한 내용은 부록 C, "Directory Server용 Sun Cluster HA 설치"를 참조하십시오.

## 설치 방법 결정

또한, 배포에 가장 적합한 패키지 및 대화형으로 설치할지 여부를 결정하기 전에 확인해야 할 대안이 있습니다. 다음과 같은 질문을 가정해 보십시오.

- Solaris 시스템 관리 프로세스와의 긴밀한 통합을 원하는가? 같은 시스템에 설치된 여러 Sun ONE 서버에서 구성 요소를 공유하기를 원하는가?

그렇다면 Solaris 패키지를 사용하여 설치하는 것이 좋습니다.

- 먼저 슈퍼유저가 되지 않고 설치하기를 원하는가? 같은 시스템에 여러 개의 독립적인 Directory Server 이진 집합을 설치하기를 원하는가?  
그렇다면 Solaris 시스템에서도 압축된 아카이브를 사용하여 설치하는 것이 좋습니다.
- Directory Server를 평가하기 위해 신속한 설치를 원하는가? 이 버전의 Directory Server를 처음 설치하는 것인가?  
그렇다면 대화형으로 설치하는 것이 좋습니다.
- 설치 스크립트를 작성하기를 원하는가? 유사한 구성으로 많은 시스템을 설치하기를 원하는가?  
그렇다면 자동 설치 프로세스를 사용하는 것이 좋습니다.

## 설치 정보 준비

미리 정보를 준비하면 설치 프로세스를 신속하게 완료할 수 있습니다. 대화형 설치를 수행하기 전에 일반 설치에 대해 표 1-2에 요약된 것처럼 먼저 설치 정보 워크시트를 작성하는 것이 좋습니다.

**표 1-2** 일반 설치 중에 필요한 기본 정보

설명	예	대답
관리 도메인	example.com	
관리 서버 포트 번호	5201	
디렉토리 관리자 ID	admin	
디렉토리 관리자 암호	\$3kReT4wD	
디렉토리 관리자 DN <sup>1</sup> (디렉토리의 슈퍼유저)	cn=Directory Manager	
디렉토리 관리자 암호(최소 8자 이상)	#\$8Yk\$-%&	
Directory Server 포트 번호(1에서 65535까지) <sup>2</sup>	389 (기본 LDAP) 636 (기본 LDAP/SSL)	
정규화된 호스트 고유 이름(DN)	dirserv.example.com	
(선택 사항) 기존 구성 디렉토리를 사용하는 경우 구성 디렉토리 호스트, 포트, 바인드 ID 및 암호	config.example.com 389 admin \$3kReT4wD	

표 1-2 일반 설치 중에 필요한 기본 정보 (계속)

설명	예	대답
(선택 사항) 기존 사용자 디렉토리를 사용하는 경우 사용자 디렉토리 호스트, 포트, 바인드 DN, 암호 및 접미사	usergroup.example.com 389 cn=Directory Manager #\$8Yk\$-%& dc=example, dc=com	
서버 ID(마침표나 공백은 사용할 수 없음)	dirserv	
서버 접미사(디렉토리 내용을 보유할 하나 이상의 접미사가 있어야 함)	dc=example,dc=com	
<i>ServerRoot</i> (소프트웨어 설치 디렉토리). 자세한 내용은 10페이지의 "기본 경로 및 파일 이름"을 참조하십시오.  이전 버전을 덮어쓰는 방식으로 설치하지 마십시오.	/var/mps/serverroot  /var/Sun/mps  C:\Program Files\Sun\MPS	
Directory Server와 Sun ONE Web Server를 동일한 <i>ServerRoot</i> 에 설치하지 마십시오.		
(UNIX 플랫폼) 공백은 사용할 수 없음		
(UNIX 플랫폼) 서버 그룹 ID <sup>3</sup>  그룹 ID 번호가 아닌 이름을 사용합니다.	noaccess	
(UNIX 플랫폼) 서버 사용자 ID  사용자 ID 번호가 아닌 이름을 사용합니다.	diruser	
(Windows) Administrator 암호	시스템 관리자에게 문의하십시오.	
(선택 사항, 기타 플랫폼) 슈퍼유저 암호		

1. DN은 모두 UTF-8 인코딩으로 입력해야 합니다. RFC 2253을 참조하십시오. ISO-8859-1과 같은 이전 인코딩은 지원되지 않습니다.
2. 1024 미만의 포트 번호는 IANA(Internet Assigned Numbers Authority)에서 할당합니다. 1024 미만의 포트를 사용하려면 슈퍼유저로 설치하십시오.
3. 설치 절차에 설명된 것처럼 적절한 UNIX 사용자 및 그룹을 작성합니다.

디렉토리 관리자 및 디렉토리 관리자 계정에 대한 정보를 제공하는 경우, Directory Server 액세스 제어 메커니즘을 사용하여 디렉토리 관리자 액세스 권한은 관리할 수 있지만 디렉토리 관리자 계정에는 Directory Server 액세스 제어가 적용되지 않는다는 것을 상기하십시오.

자동 설치 구성 파일에도 유사한 정보가 포함되어 있습니다.

## Solaris 시스템에 설치

Directory Server 소프트웨어의 설치 방법은 사용할 패키지 및 대화형 설치의 사용 여부에 따라 결정됩니다. 해당 절의 지침에 따라 수행합니다.

- Solaris 패키지를 사용한 설치 준비
- Solaris 패키지를 사용한 대화형 설치
- Solaris 패키지를 사용한 자동 설치
- 압축된 아카이브를 사용한 설치 준비
- 압축된 아카이브를 사용한 대화형 설치
- 압축된 아카이브를 사용한 자동 설치
- 설치 프로세스 완료

Sun Cluster 시스템에 Directory Server를 설치하는 경우 부록 C, "Directory Server용 Sun Cluster HA 설치"의 지침에 따라 수행합니다.

### Solaris 패키지를 사용한 설치 준비

1. (선택 사항) Directory Server에 대한 사용자 및 그룹 계정을 작성합니다.

Directory Server는 설치 중에 지정한 사용자 및 그룹으로 실행됩니다. 권한 없이 디렉토리 및 시스템의 다른 자원을 액세스하지 못하도록 사용 권한을 설정합니다. 자세한 내용은 99페이지의 "(UNIX 플랫폼) 사용자 및 그룹"을 참조하십시오.

2. (선택 사항) `xhost(1)` 명령을 사용하여 디스플레이에 대한 액세스를 허용합니다.

DISPLAY 환경 변수를 적절하게 설정하고 디스플레이 액세스 권한이 있는 사용자로 설치를 수행하면 설치 프로그램에서 기본적으로 그래픽 사용자 인터페이스를 표시합니다.

그래픽 사용자 인터페이스를 표시할 수 없는 경우 설치 프로그램은 명령줄 모드로 설치를 시작합니다.

3. 영어 이외의 로케일을 사용하여 설치하려면 먼저 LANG 환경 변수를 C로 설정합니다.
4. 기본 시스템에 기본적으로 설치되는 모든 Solaris 패키지 외에 표 1-3에 열거된 필수 패키지가 설치되어 있는지 확인합니다.

**표 1-3** 필수 Solaris 패키지

패키지	설명	32비트 Directory Server에 필요	64비트 Directory Server에 필요
SUNWj3rt <sup>1</sup>	J2SDK 1.4 런타임 환경	예	예
SUNWzlib	Zip 압축 라이브러리	예	예
SUNWzlibx	Zip 압축 라이브러리(64비트)	아니오	예

1. Java Runtime Environment 버전 1.4.1 이상을 사용해야 합니다.

## Solaris 패키지를 사용한 대화형 설치

아래 절차의 단계를 수행합니다.

### Solaris 패키지 설치

Solaris 패키지는 pkgadd(1M) 유틸리티를 사용하여 설치합니다. 업그레이드를 수행하려는 경우와 같이 이미 설치되어 있는 패키지를 확인하려면 pkginfo(1) 명령을 사용하십시오. 여러 호스트에 패키지를 설치하는 경우 admin(4)에 설명된 설치용 기본 파일을 사용하여 기본 설치 작업을 정의할 수 있습니다. 어떤 경우라도 모든 패키지는 동일한 *basedir*을 공유해야 합니다.

소프트웨어 패키지 처리에 대한 자세한 내용은 Solaris Operating Environment 시스템 관리 설명서를 참조하십시오.

1. 표 1-4 또는 표 1-5에 열거된 전체 패키지 목록을 확인합니다.

**표 1-4** 제공된 Solaris 패키지(SPARC 플랫폼)

패키지	설명
SUNWasha	Sun Cluster용 Sun ONE Administration Server 구성 요소
SUNWasvc	Sun ONE 관리 콘솔
SUNWasvcp	Sun ONE Administration Server 콘솔 플러그인
SUNWasvr	Sun ONE Administration Server(Root)
SUNWasvu	Sun ONE Administration Server(사용자)
SUNWdsha	Sun Cluster용 Sun ONE Directory Server 구성 요소
SUNWdsvcp	Sun ONE Directory Server 콘솔 플러그인
SUNWdsvh	Sun ONE Directory Server 힙(Heap) 할당자(Solaris 8 시스템만)
SUNWdsvhx	Sun ONE Directory Server 힙(Heap) 할당자(64비트, Solaris 8 시스템만)



**표 1-4** 제공된 Solaris 패키지(SPARC 플랫폼) (계속)

패키지	설명
SUNWdsvpl	Sun ONE Directory Server PerLDAP 모듈
SUNWdsvr	Sun ONE Directory Server(Root)
SUNWdsvu	Sun ONE Directory Server(사용자)
SUNWdsvx	Sun ONE Directory Server(64비트)
SUNWicu	ICU(International Components for Unicode) 사용자 파일
SUNWicux	ICU(International Components for Unicode) 사용자 파일(64비트)
SUNWjss	Java용 네트워크 보안 서비스(JSS)
SUNWldk	LDAP C SDK
SUNWldkx	LDAP C SDK(64비트)
SUNWpr	Netscape Portable Runtime 인터페이스
SUNWprx	Netscape Portable Runtime 인터페이스(64비트)
SUNWsas1	SASL(Simple Authentication and Security Layer)
SUNWsas1x	SASL(Simple Authentication and Security Layer)(64비트)
SUNWt1s	네트워크 보안 서비스
SUNWt1sx	네트워크 보안 서비스(64비트)

**표 1-5** 제공된 Solaris 패키지(x86 플랫폼)

패키지	설명
SUNWasvc	Sun ONE 관리 콘솔
SUNWasvcp	Sun ONE Administration Server 콘솔 플러그인
SUNWasvr	Sun ONE Administration Server(Root)
SUNWasvu	Sun ONE Administration Server(사용자)
SUNWdsvcp	Sun ONE Directory Server 콘솔 플러그인
SUNWdsvpl	Sun ONE Directory Server PerLDAP 모듈
SUNWdsvr	Sun ONE Directory Server(Root)
SUNWdsvu	Sun ONE Directory Server(사용자)
SUNWicu	ICU(International Components for Unicode) 사용자 파일
SUNWjss	Java용 네트워크 보안 서비스(JSS)

**표 1-5** 제공된 Solaris 패키지(x86 플랫폼) (계속)

패키지	설명
SUNWldk	LDAP C SDK
SUNWpr	Netscape Portable Runtime 인터페이스
SUNWsas1	SASL(Simple Authentication and Security Layer)
SUNWtls	네트워크 보안 서비스

패키지를 설치할 때는 항상 /var과 같은 쓰기 가능한 *basedir*을 사용하는 것이 좋습니다. 단, 패키지 위치를 변경할 경우 SUNWasvr과 SUNWdsvr은 시작 및 종료 스크립트를 *basedir/etc*에 저장합니다.

2. 표 1-6에 설명된 힌트를 사용하여 설치할 패키지를 결정합니다.

**표 1-6** 설치할 패키지 종류

구성	설치할 패키지 목록 <sup>1</sup>
32비트 Directory Server, 관리 서버 및 콘솔	SUNWascv SUNWasvcvcp SUNWasvr SUNWasvu SUNWdsvcp SUNWdsvh SUNWdsvpl SUNWdsvr SUNWdsvu SUNWicu SUNWjss SUNWldk SUNWpr SUNWsas1 SUNWtls
32비트 Directory Server만(콘솔 없음)	SUNWasvu SUNWdsvh SUNWdsvpl SUNWdsvr SUNWdsvu SUNWicu SUNWjss SUNWldk SUNWpr SSUNWsas1 SUNWtls
64비트 Directory Server, 32비트 관리 서버 및 콘솔	SUNWascv SUNWasvcvcp SUNWasvr SUNWasvu SUNWdsvcp SUNWdsvh SUNWdsvhx SUNWdsvpl SUNWdsvr SUNWdsvu SUNWdsvx SUNWicu SUNWicux SUNWjss SUNWldk SUNWldkx SUNWpr SUNWprx SUNWsas1 SUNWsas1x SUNWtls SUNWtlx
64비트 Directory Server만(콘솔 없음)	SUNWasvu,SUNWdsvh SUNWdsvhx SUNWdsvpl SUNWdsvr SUNWdsvu SUNWdsvx SUNWicu SUNWicux SUNWjss SUNWldk SUNWldkx SUNWpr SUNWprx SUNWsas1 SUNWsas1x SUNWtls SUNWtlx
클러스터 노드	SUNWasha SUNWdsha 추가
Sun ONE 서버 콘솔 및 관리 서버만 (Directory Server 없음, 원격 관리만)	SUNWasvc SUNWasvcvcp SUNWasvr SUNWasvu SUNWdsvcp SUNWjss SUNWldk SUNWpr SUNWsas1 SUNWtls

1. SUNWdsvh(32비트) 및 SUNWdsvhx(64비트) 패키지는 Solaris 8 시스템에 설치된 Directory Server에만 필요합니다.

3. 설치할 패키지 중에서 아직 설치되지 않은 패키지가 있는지 확인합니다.

시스템에 설치되어 있는 패키지는 다시 설치하지 마십시오.

4. 슈퍼유저가 됩니다.

5. pkgadd(1M) 유틸리티를 사용하여 제품 패키지를 시스템으로 전송합니다.

SUNWicu 및 SUNwicux 패키지는 Directory Server를 설치한 시스템에서 실행하는 Solaris 버전에 따라 달라집니다.

또한 구성 요소 패키지인 SUNWpr, SUNWprx, SUNWsas1, SUNWsas1x, SUNWt1s 및 SUNWt1sx의 설치 및 패치에 대한 자세한 내용은 이후의 "필수 패치 설치" 절을 참조하십시오.

6. pkgadd를 종료한 후에 필수 제품 패키지가 모두 설치되었는지 확인합니다.

IPLT\* Solaris 패키지를 사용하여 설치한 iPlanet Directory Server 5.1을 업그레이드하는 경우에는 `5.1 /usr/sbin/directoryserver` 명령의 이름이

`/usr/sbin/directoryserver.51bak`으로 바꿉니다. 바뀐 이름의 명령을 사용하여 5.1 버전을 관리할 수도 있습니다.

### 필수 패치 설치

Directory Server는 최신 수정 프로그램을 포함하도록 업데이트되었으며 권장 시스템 패치에 있는 SUNWpr, SUNWprx, SUNWsas1, SUNWsas1x, SUNWt1s 및 SUNWt1sx 패키지를 사용합니다.

1. pkginfo(1)를 `-x` 옵션과 함께 사용하여 시스템에 설치되어 있는 패키지를 확인합니다. 특히 표 1-7에 열거된 적절한 패키지 버전이 시스템에 설치되어 있는지 확인합니다.

**표 1-7** 구성 요소에 적합한 버전 및 패치

시스템 버전 및 구조	SUNWpr(x) 버전	SUNWsas(x) 버전	SUNWt1s(x) 버전	패치
Solaris 9(SPARC 플랫폼)	4.1.2 이상	2.01 이상	3.3.2 이상	114049, 115342
Solaris 9(x86 플랫폼)	4.1.3 이상	2.01 이상	3.3.3 이상	114050, 115343
Solaris 8(SPARC 플랫폼)	4.1.2 이상	2.01 이상	3.3.2 이상	114045, 115328

2. showrev(1M)를 `-p` 옵션과 함께 사용하여 표 1-7에 열거된 적절한 패치가 플랫폼에 설치되어 있는지 확인합니다.

3. 표 1-8에 설명된 힌트를 사용하여 구성 요소 패치를 설치할지 여부를 결정합니다.

**표 1-8** 구성 요소 패치의 설치 여부

시스템 상태	수행 작업
패키지와 패치가 이미 설치되어 있습니다.	단계 4로 넘어갑니다.
패치를 제외하고 패키지만 설치되어 있습니다.	Directory Server와 함께 제공된 사용자 플랫폼용 패치를 설치합니다.
패키지가 설치되어 있지 않습니다.	Directory Server와 함께 제공된 패키지과 해당 패치를 설치합니다.

- 수퍼유저로 아래 명령을 실행합니다.

```
root# /usr/sbin/directoryserver idsktune -q > idsktune.out
```

idsktune은 시스템 변경 사항을 제안하기만 하고 시스템을 변경하지는 않습니다.

- 최소한 표시된 ERROR 조건을 모두 수정합니다.

ERROR 조건을 수정하지 않으면 제대로 설치되지 않을 수도 있습니다. idsktune 하위 명령은 릴리스 시 권장되었지만 시스템에 설치되지 않은 패치는 물론 시스템에 설치되지 않은 패키지 패치까지 모두 보고합니다.

패치는 <http://sunsolve.sun.com/>에서 다운로드할 수 있습니다.

자세한 내용은 5장, "운영 체제 조정"을 참조하십시오.

### Directory Server 구성

- 구성 프로그램을 시작합니다.

그래픽 사용자 인터페이스를 사용하려면 아래 명령을 실행합니다.

```
root# /usr/sbin/directoryserver configure
```

명령줄 인터페이스를 사용하려면 아래 명령을 실행합니다.

```
root# /usr/sbin/directoryserver configure -nodisplay
```

첫 설치 화면이 나타납니다.

- 21페이지의 "설치 정보 준비"에서 작성한 워크시트를 사용하여 각 화면에 표시된 지침에 따라 수행합니다.

## 관리 서버 구성

1. 구성 프로그램을 시작합니다.

그래픽 사용자 인터페이스를 사용하려면 아래 명령을 실행합니다.

```
root# /usr/sbin/mpsadmserver configure
```

명령줄 인터페이스를 사용하려면 아래 명령을 실행합니다.

```
root# /usr/sbin/mpsadmserver configure -nodisplay
```

첫 설치 화면이 나타납니다.

2. 21페이지의 "설치 정보 준비"에서 작성한 워크시트를 사용하여 각 화면에 표시된 지침에 따라 수행합니다.

33페이지의 "설치 프로세스 완료"로 넘어갑니다.

## Solaris 패키지를 사용한 자동 설치

아래 절차의 단계를 수행합니다.

### Solaris 패키지 설치

24페이지의 "Solaris 패키지 설치"에 설명된 지침에 따라 수행합니다.

### 필수 패치 설치

27페이지의 "필수 패치 설치"에 설명된 지침에 따라 수행합니다.

### 사양 파일 작성

전자동 설치를 수행하려면 각각 Directory Server와 관리 서버의 설치 사양이 포함된 두 파일을 먼저 작성해야 합니다. Directory Server의 설치 사양 파일 템플리트는 `/usr/ds/v5.2/setup/typical.ins`를 참조하십시오. 관리 서버의 경우 `/usr/sadm/mps/admin/v5.2/setup/admin/typicalInstall.ins`를 참조하십시오.

---

**주** 사양 파일에는 암호가 일반 텍스트로 포함되어 있을 수 있습니다. 적절한 파일 사용 권한을 할당하여 파일을 보호하십시오.

---

템플리트 파일 복사본을 수동으로 편집하거나 Directory Server 및 관리 서버 구성 프로그램을 사용한 대화형 구성을 통해 자동 설치 사양 파일을 작성할 수 있습니다.

Directory Server 및 관리 서버의 자동 설치 사양 파일을 대화형 방식으로 작성하려면 다음 단계를 수행합니다.

1. `-saveState` 옵션을 사용하여 Directory Server를 구성합니다.

```
root# /usr/sbin/directoryserver configure -saveState dirserv-file
dirserv-file 사양 파일이 작성됩니다.
```

2. `-saveState` 옵션을 사용하여 관리 서버를 구성합니다.

```
root# /usr/sbin/mpsadmserver configure -saveState admserve-file
admserve-file 사양 파일이 작성됩니다.
```

3. *dirserv-file* 및 *admserve-file* 사양 파일을 사용하여 다른 시스템에 설치하기 전에 두 파일을 조정합니다.

자동 설치 사양 파일에서 `FullMachineName`과 같은 일부 지시문은 기본 호스트 시스템에 따라 결정되며 일반적으로 생성할 수 없습니다.

자동 설치 사양 파일에는 설치 프로그램의 빌드 버전에 해당하는 체크섬 문자열이 포함되어 있습니다. 다른 빌드 또는 릴리스의 설치 프로그램에서 자동 설치 사양 파일을 다시 사용하려면 `[STATE_BEGIN` 및 `[STATE_DONE`으로 시작되는 줄에서 체크섬 문자열을 업데이트합니다. 업데이트된 체크섬은 Directory Server의 경우 `/usr/ds/v5.2/setup/typical.ins`에, 관리 서버의 경우 `/usr/sadm/mps/admin/v5.2/setup/admin/typicalInstall.ins`에 있습니다. 코드 예제 1-1에서는 샘플 체크섬을 보여줍니다.

#### 코드 예제 1-1          자동 설치 체크섬 줄

```
[STATE_BEGIN Sun ONE Directory Distribution a7cc64b2f71a0452899e1c3b853eceed72027b3b]
```

### 사양 파일을 사용한 설치

Directory Server와 관리 서버를 대화형 방식으로 구성하려면 다음 단계를 수행합니다.

1. 자동 설치 사양 파일의 변경 사항을 확인합니다.
2. 아래 명령을 실행하여 자동 모드로 Directory Server를 구성합니다.

```
root# /usr/sbin/directoryserver configure -f dirserv-file
여기서 dirserv-file은 자동 설치 구성 파일입니다.
```

3. 아래 명령을 실행하여 자동 모드로 관리 서버를 구성합니다.

```
root# /usr/sbin/mpsadmserver configure -f admserv-file
```

여기서 *admserv-file*은 자동 설치 구성 파일입니다.

33페이지의 "설치 프로세스 완료"로 넘어갑니다.

## 압축된 아카이브를 사용한 설치 준비

1. 19페이지의 "Directory Server 소프트웨어 다운로드"에 설명된 것처럼 압축을 푼 소프트웨어 디렉토리에서 `idsktune` 유틸리티를 실행합니다. `idsktune`은 적절한 패치가 설치되어 있는지 검사하여 시스템의 디렉토리 서비스 성능이 최적화되었는지 확인합니다.

수퍼유저로 아래 명령을 실행합니다.

```
root# ./idsktune -q > idsktune.out
```

제안된 시스템 설정을 수동으로 변경해야 합니다. `idsktune` 유틸리티는 시스템을 변경하지 않습니다.

2. 최소한 `idsktune`에서 표시한 `ERROR` 조건은 모두 수정합니다. `ERROR` 조건을 수정하지 않으면 제대로 설치되지 않을 수도 있습니다. `idsktune`은 릴리스 시 권장되었지만 시스템에 설치되지 않은 패치는 물론 시스템에 설치되지 않은 패키지 패치까지 모두 보고합니다.

패치는 <http://sunsolve.sun.com/>에서 다운로드할 수 있습니다.

자세한 내용은 5장, "운영 체제 조정"을 참조하십시오.

3. (선택 사항) `Directory Server`에 대한 사용자 및 그룹 계정을 작성합니다.

`Directory Server`는 설치 중에 지정한 사용자 및 그룹으로 실행됩니다. 권한 없이 디렉토리 및 시스템의 다른 자원을 액세스하지 못하도록 사용 권한을 설정합니다. 자세한 내용은 99페이지의 "(UNIX 플랫폼) 사용자 및 그룹"을 참조하십시오.

4. (선택 사항) 다른 사용자로 대화형 설치를 수행하는 경우 `xhost(1)` 명령을 사용하여 디스플레이에 대한 액세스를 허용합니다.

`DISPLAY` 환경 변수를 적절하게 설정하고 디스플레이 액세스 권한이 있는 사용자로 설치를 수행하면 설치 프로그램에서 기본적으로 그래픽 사용자 인터페이스를 표시합니다.

그래픽 사용자 인터페이스를 표시할 수 없는 경우 설치 프로그램은 명령줄 모드로 설치를 시작합니다.

5. 영어 이외의 로케일을 사용하여 설치하려면 먼저 `LANG` 환경 변수를 `C`로 설정합니다.

## 압축된 아카이브를 사용한 대화형 설치

1. 압축을 푼 소프트웨어 디렉토리에 있는 설치 프로그램을 시작합니다.

그래픽 사용자 인터페이스의 경우 아래 명령을 실행합니다.

```
root# ./setup
```

명령줄 인터페이스의 경우 아래 명령을 실행합니다.

```
root# ./setup -nodisplay
```

첫 설치 화면이 나타납니다.

2. 21페이지의 "설치 정보 준비"에서 작성한 워크시트를 사용하여 각 화면에 표시된 지침에 따라 수행합니다.

---

**주** 32비트 Directory Server를 설치하려면 구성 요소 선택 마법사 화면에서 "Sun ONE Directory Suite > Sun ONE Directory Server(64비트 지원)" 옆에 있는 확인란을 선택 취소합니다.

---

이 버전을 Directory Server의 이전 버전과 같은 디렉토리에 설치하지 마십시오. 같은 디렉토리 위치를 다시 사용해야 하는 경우 먼저 이전 버전을 제거해야 합니다. 자세한 내용은 2장, "이전 버전에서 업그레이드"를 참조하십시오.

## 압축된 아카이브를 사용한 자동 설치

아래 절차의 단계를 수행합니다.

### 사양 파일 작성

자동 설치를 수행하려면 먼저 설치 사양이 포함된 파일을 작성해야 합니다. 자동 설치 사양 파일 템플릿은 소프트웨어 압축을 푼 디렉토리 아래의 `setup_data/typical.ins`를 참조하십시오.

---

**주** 사양 파일에는 암호가 일반 텍스트로 포함되어 있을 수 있습니다. 적절한 파일 사용 권한을 할당하여 파일을 보호하십시오.

---

템플릿 파일 복사본을 수동으로 편집하거나 설치 프로그램을 사용한 대화형 구성을 통해 자동 설치 사양 파일을 작성할 수 있습니다.

1. 슈퍼유저가 됩니다.



2. `-saveState` 옵션을 사용하여 설치 프로그램을 시작합니다.

```
root# ./setup -saveState filename
```

*filename* 사양 파일이 작성됩니다.

3. 대화형 설치를 수행합니다.
4. *filename* 사양 파일을 사용하여 다른 시스템에 설치하기 전에 파일을 조정합니다.

자동 설치 사양 파일에서 `FullMachineName`과 같은 일부 지시문은 기본 호스트 시스템에 따라 결정되며 일반적으로 생성할 수 없습니다.

자동 설치 사양 파일에는 설치 프로그램의 빌드 버전에 해당하는 체크섬 문자열이 포함되어 있습니다. 다른 빌드 또는 릴리스의 설치 프로그램에서 자동 설치 사양 파일을 다시 사용하려면 `[STATE_BEGIN` 및 `[STATE_DONE`으로 시작되는 줄에서 체크섬 문자열을 업데이트합니다. 업데이트된 체크섬은 `typical.ins`에서 확인할 수 있습니다. 30페이지의 코드 예제 1-1에서는 샘플 체크섬을 보여줍니다.

### 사양 파일을 사용한 설치

1. 설치 사양 파일의 변경 사항을 확인합니다.
2. 자동 모드로 설치 프로그램을 시작합니다.

```
root# ./setup -noconsole -nodisplay -state filename
```

여기서 *filename*은 자동 설치 사양 파일입니다.

### 설치 프로세스 완료

1. `ServerRoot/alias` 파일에 대한 액세스 권한이 `ServerRoot` 아래에 설치한 서버 이외의 모든 사용자 액세스를 차단하도록 설정되어 있는지 확인합니다.
2. (선택 사항) 압축된 아카이브를 사용하여 설치한 경우 시스템 재부트 시 `Directory Server`를 시작하는 지원을 추가합니다. 이 지원은 Solaris 패키지 버전에 포함되어 있습니다.

자세한 내용은 Solaris 시스템 관리 설명서를 참조하십시오.

### 3. (선택 사항) 코어 파일 생성 기능을 활성화합니다.

수퍼유저로 Directory Server를 설치했지만 사용자 및 그룹 ID를 다른 계정의 사용자 및 그룹 ID로 설정한 경우에는 Directory Server가 크래시 중에 core 파일을 생성할 수 없습니다. core 파일을 위한 충분한 공간을 계획하고 Directory Server가 크래시 중에 이 파일을 생성하도록 설정하는 것이 좋습니다.

coreadm(1M) 명령을 사용하여 core 파일 생성 기능을 관리할 수 있으므로 아래 명령을 실행하여 Directory Server에서 core 파일을 생성하도록 설정할 수도 있습니다.

```
root# coreadm -e proc-setid
```

자세한 내용은 88페이지의 "(UNIX 플랫폼) 코어 파일"을 참조하십시오.

### 4. (선택 사항) Perl로 작성된 많은 명령줄 스크립트는 이제 대화형 방식으로 바인드 암호를 읽을 수 있습니다(-w - 옵션). 이 기능을 사용하려면 다음을 수행합니다.

- a. CPAN에서 별도로 사용할 수 있는 Term::ReadKey Perl 모듈을 설치합니다.
- b. 해당 줄의 주석 처리를 제거하여 Perl 스크립트에서 대화형 방식으로 바인드 암호를 읽을 수 있도록 각 스크립트를 편집합니다.

Term::ReadKey 모듈이 없어도 Perl 스크립트의 다른 모든 기능은 계속 사용할 수 있습니다.

Directory Server가 최소 구성으로 시작됩니다.

## 기타 UNIX 시스템에 설치

해당 절의 지침에 따라 수행합니다.

- 설치 준비
- 대화형 설치
- 자동 설치
- 설치 프로세스 완료

### 설치 준비

해당 절의 지침에 따라 수행합니다.

- 모든 UNIX 플랫폼에 대한 지침
- AIX 시스템에 대한 추가 지침
- HP-UX 시스템에 대한 추가 지침

## 모든 UNIX 플랫폼에 대한 지침

1. 압축을 푼 소프트웨어 디렉토리에 있는 `idsktune` 유틸리티를 실행합니다. `idsktune`은 적절한 패치가 설치되어 있는지 검사하여 시스템의 디렉토리 서비스 성능이 최적화되었는지 확인합니다.

수퍼유저로 아래 명령을 실행합니다.

```
root# ./idsktune -q > idsktune.out
```

제안된 시스템 설정을 수동으로 변경해야 합니다. `idsktune` 유틸리티는 시스템을 변경하지 않습니다.

2. 최소한 `idsktune`에서 표시한 **ERROR** 조건은 모두 수정합니다. **ERROR** 조건을 수정하지 않으면 제대로 설치되지 않을 수도 있습니다.

표 1-9에는 시스템에 설치되지 않은 공식 패치를 다운로드할 수 있는 사이트가 나와 있습니다.

**표 1-9** 플랫폼별 패치 다운로드 사이트

플랫폼	다운로드 사이트
Hewlett Packard HP-UX	<a href="http://www.hp.com/support/">http://www.hp.com/support/</a>
IBM AIX	<a href="http://www.ibm.com/support/">http://www.ibm.com/support/</a>
Red Hat Linux	<a href="http://www.redhat.com/">http://www.redhat.com/</a>

자세한 내용은 97페이지의 5장, "운영 체제 조정"을 참조하십시오.

3. (선택 사항) **Directory Server**에 대한 사용자 및 그룹 계정을 작성합니다.

**Directory Server**는 설치 중에 지정한 사용자 및 그룹으로 실행됩니다. 권한 없이 디렉토리 및 시스템의 다른 자원을 액세스하지 못하도록 사용 권한을 설정합니다. 자세한 내용은 99페이지의 "(UNIX 플랫폼) 사용자 및 그룹"을 참조하십시오.

4. (선택 사항) 다른 사용자로 대화형 설치를 수행하는 경우 `xhost(1)` 명령을 사용하여 디스플레이에 대한 액세스를 허용합니다.

`DISPLAY` 환경 변수를 적절하게 설정하고 디스플레이 액세스 권한이 있는 사용자로 설치를 수행하면 설치 프로그램에서 기본적으로 그래픽 사용자 인터페이스를 표시합니다.

그래픽 사용자 인터페이스를 표시할 수 없는 경우 설치 프로그램은 명령줄 모드로 설치를 시작합니다.

5. 영어 이외의 로케를 사용하여 설치하려면 먼저 LANG 환경 변수를 C로 설정합니다.

### **AIX 시스템에 대한 추가 지침**

- 콘솔을 사용하려면 x11.adt 패키지를 설치합니다.

이 패키지는 표준 번들의 일부는 아니지만 IBM에서 구할 수 있습니다.

### **HP-UX 시스템에 대한 추가 지침**

1. Directory Server에서 IPv6 인터페이스를 사용하지 않는 경우에도 IPv6 지원을 설치해야 합니다.
2. US 영어를 지원하지 않는 폰트가 있는 로케를 사용하여 원격으로 설치하려면 원격 세션에 대한 폰트 별명에 액세스할 수 있는지 확인합니다.

자세한 내용은 운영 체제 설명서를 참조하십시오.

## **대화형 설치**

1. 압축을 푼 소프트웨어 디렉토리에 있는 설치 프로그램을 시작합니다.

그래픽 사용자 인터페이스의 경우 아래 명령을 실행합니다.

```
root# ./setup
```

명령줄 인터페이스의 경우 아래 명령을 실행합니다.

```
root# ./setup -nodisplay
```

첫 설치 화면이 나타납니다.

2. 21페이지의 "설치 정보 준비"에서 작성한 워크시트를 사용하여 각 화면에 표시된 지침에 따라 수행합니다.

---

**주** 64비트 서버를 지원하는 플랫폼에 32비트 Directory Server를 설치하려면 구성 요소 선택 마법사 화면에서 "Sun ONE Directory Suite > Sun ONE Directory Server(64비트 지원)" 옆에 있는 확인란을 선택 취소합니다.

---

이 버전을 Directory Server의 이전 버전과 같은 디렉토리에 설치하지 마십시오. 같은 디렉토리 위치를 다시 사용해야 하는 경우 먼저 이전 버전을 제거해야 합니다. 자세한 내용은 2장, "이전 버전에서 업그레이드"를 참조하십시오.

38페이지의 "설치 프로세스 완료"로 넘어갑니다.

## 자동 설치

아래 절차의 단계를 수행합니다.

### 사양 파일 작성

자동 설치를 수행하려면 먼저 설치 사양이 포함된 파일을 작성해야 합니다. 자동 설치 사양 파일 템플리트는 소프트웨어 압축을 푼 디렉토리 아래의 `setup_data/typical.ins`를 참조하십시오.

---

**주** 사양 파일에는 암호가 일반 텍스트로 포함되어 있을 수 있습니다. 적절한 파일 사용 권한을 할당하여 파일을 보호하십시오.

---

템플리트 파일 복사본을 수동으로 편집하거나 설치 프로그램을 사용한 대화형 구성을 통해 자동 설치 사양 파일을 작성할 수 있습니다.

1. 슈퍼유저가 됩니다.
2. `-saveState` 옵션을 사용하여 설치 프로그램을 시작합니다.  

```
root# ./setup -saveState filename
```

`filename` 사양 파일이 작성됩니다.
3. 대화형 설치를 수행합니다.
4. `filename` 사양 파일을 사용하여 다른 시스템에 설치하기 전에 파일을 조정합니다.

자동 설치 사양 파일에서 `FullMachineName`과 같은 일부 지시문은 기본 호스트 시스템에 따라 결정되며 일반적으로 생성할 수 없습니다.

자동 설치 사양 파일에는 설치 프로그램의 빌드 버전에 해당하는 체크섬 문자열이 포함되어 있습니다. 다른 빌드 또는 릴리스의 설치 프로그램에서 자동 설치 사양 파일을 다시 사용하려면 `[STATE_BEGIN` 및 `[STATE_DONE`으로 시작되는 줄에서 체크섬 문자열을 업데이트합니다. 업데이트된 체크섬은 `typical.ins`에서 확인할 수 있습니다. 30페이지의 코드 예제 1-1에서는 샘플 체크섬을 보여줍니다.

### 사양 파일을 사용한 설치

1. 설치 사양 파일의 변경 사항을 확인합니다.
2. 자동 모드로 설치 프로그램을 시작합니다.

```
root# ./setup -noconsole -nodisplay -state filename
```

여기서 *filename*은 자동 설치 사양 파일입니다.

## 설치 프로세스 완료

1. *ServerRoot/alias* 파일에 대한 액세스 권한이 *ServerRoot* 아래에 설치한 서버 이외의 모든 사용자 액세스를 차단하도록 설정되어 있는지 확인합니다.

2. (선택 사항) 시스템 재부트 시 **Directory Server**를 시작하는 지원을 추가합니다.

자세한 내용은 운영 체제 설명서를 참조하십시오.

3. (선택 사항) 코어 파일 생성 기능을 활성화합니다.

수퍼유저로 **Directory Server**를 설치했지만 사용자 및 그룹 ID를 다른 계정의 사용자 및 그룹 ID로 설정한 경우에는 **Directory Server**가 크래시 중에 *core* 파일을 생성할 수 없습니다. *core* 파일을 위한 충분한 공간을 계획하고 **Directory Server**가 크래시 중에 이 파일을 생성하도록 설정하는 것이 좋습니다.

자세한 내용은 88페이지의 "(UNIX 플랫폼) 코어 파일"을 참조하십시오.

4. (선택 사항) Perl로 작성된 많은 명령줄 스크립트는 이제 대화형 방식으로 바인드 암호를 읽을 수 있습니다(-w - 옵션). 이 기능을 사용하려면 다음을 수행합니다.

- a. CPAN에서 별도로 사용할 수 있는 `Term::ReadKey` Perl 모듈을 설치합니다.

- a. 해당 줄의 주석 처리를 제거하여 Perl 스크립트에서 대화형 방식으로 바인드 암호를 읽을 수 있도록 각 스크립트를 편집합니다.

`Term::ReadKey` 모듈이 없어도 Perl 스크립트의 다른 모든 기능은 계속 사용할 수 있습니다.

**Directory Server**가 최소 구성으로 시작됩니다.

## Windows 시스템에 설치

해당 절의 지침에 따라 수행합니다.

- 설치 준비
- 대화형 설치
- 자동 설치
- 설치 프로세스 완료

## 설치 준비

1. Windows 2000을 설치하는 경우, 네트워크 보안 서비스에 대한 의존도를 줄이기 위해 컴퓨터를 기존 도메인이나 작업 그룹의 구성원이 아니라 독립 실행형 서버로 지정합니다.
2. 서비스 팩 3을 설치합니다.
3. 디스플레이 드라이버가 256개 이상의 색상을 지원하는지 확인합니다.
4. Administrator 권한이 있는 사용자로 로그인합니다.
5. TEMP 환경 변수를 임시 파일을 저장할 유효한 폴더로 설정합니다.

## 대화형 설치

1. 압축을 푼 소프트웨어가 들어 있는 폴더에서 `setup.exe`를 두 번 누릅니다.  
첫 설치 화면이 나타납니다.
2. 21페이지의 "설치 정보 준비"에서 작성한 워크시트를 사용하여 각 화면에 표시된 지침에 따라 수행합니다.  
  
이 버전을 Directory Server의 이전 버전과 같은 폴더에 설치하지 마십시오. 같은 폴더를 다시 사용해야 하는 경우 먼저 이전 버전을 제거해야 합니다. 자세한 내용은 2장, "이전 버전에서 업그레이드"를 참조하십시오.  
  
40페이지의 "설치 프로세스 완료"로 넘어갑니다.

## 자동 설치

아래 절차의 단계를 수행합니다.

### 사양 파일 작성

자동 설치를 수행하려면 먼저 설치 사양이 포함된 파일을 작성해야 합니다. 자동 설치 사양 파일 템플리트는 소프트웨어 압축을 푼 폴더에 있는 `setup_data\typical.ins`를 참조하십시오.

---

**주** 사양 파일에는 암호가 일반 텍스트로 포함되어 있을 수 있습니다. 적절한 파일 사용 권한을 할당하여 파일을 보호하십시오.

---

템플리트 파일 복사본을 수동으로 편집하거나 설치 프로그램을 사용한 대화형 구성을 통해 자동 설치 사양 파일을 작성할 수 있습니다.

1. Administrator 권한이 있는 사용자로 로그인합니다.
2. `-saveState` 옵션을 사용하여 설치 프로그램을 시작합니다.

제품의 압축을 푼 폴더에서 아래 명령을 실행합니다.

```
Prompt>setup -saveState filename
```

`filename` 사양 파일이 작성됩니다.

3. 대화형 설치를 수행합니다.
4. `filename` 사양 파일을 사용하여 다른 시스템에 설치하기 전에 파일을 조정합니다.

자동 설치 사양 파일에서 `FullMachineName`과 같은 일부 지시문은 기본 호스트 시스템에 따라 결정되며 일반적으로 생성할 수 없습니다.

자동 설치 사양 파일에는 설치 프로그램의 빌드 버전에 해당하는 체크섬 문자열이 포함되어 있습니다. 다른 빌드 또는 릴리스의 설치 프로그램에서 자동 설치 사양 파일을 다시 사용하려면 `[STATE_BEGIN` 및 `[STATE_DONE`으로 시작되는 줄에서 체크섬 문자열을 업데이트합니다. 업데이트된 체크섬은 `typical.ins`에서 확인할 수 있습니다. 30페이지의 코드 예제 1-1에서는 샘플 체크섬을 보여줍니다.

### 사양 파일을 사용한 설치

1. 설치 사양 파일의 변경 사항을 확인합니다.
2. 자동 모드로 설치 프로그램을 시작합니다.

제품의 압축을 푼 폴더에서 아래 명령을 실행합니다.

```
Prompt>setup -noconsole -nodisplay -state filename
```

여기서 `filename`은 자동 설치 사양 파일입니다.

### 설치 프로세스 완료

1. `ServerRoot\alias` 파일에 대한 액세스 권한이 `ServerRoot` 아래에 설치한 서버 이외의 모든 사용자 액세스를 차단하도록 설정되어 있는지 확인합니다.
2. 설치가 끝나면 관리 서버를 실행하는 사용자 및 그룹에게만 읽기-쓰기 액세스 권한이 있고 다른 모든 사용자에게는 액세스 권한이 없도록 다음 파일에 특수 액세스 권한을 수동으로 설정합니다.
  - `ServerRoot\admin-serv\config\adm.conf`
  - `ServerRoot\admin-serv\config\admpw`
  - `ServerRoot\admin-serv\config\magnus.conf`



- o `ServerRoot\admin-serv\config\obj.conf`
- o `ServerRoot\admin-serv\config\secmod.db`
- o `ServerRoot\admin-serv\config\server.xml`

파일에 특수 액세스 권한을 설정하는 방법은 Windows 도움말을 참조하십시오. 이렇게 액세스 권한을 설정하면 권한 없는 사용자는 관리 서버 구성 데이터를 수정할 수 없게 됩니다.

3. (선택 사항) Perl로 작성된 많은 명령줄 스크립트는 이제 대화형 방식으로 바인드 암호를 읽을 수 있습니다(`-w` 옵션). 이 기능을 사용하려면 다음을 수행합니다.
  - a. CPAN에서 별도로 사용할 수 있는 `Term::ReadKey` Perl 모듈을 설치합니다.
  - b. 해당 줄의 주석 처리를 제거하여 Perl 스크립트에서 대화형 방식으로 바인드 암호를 읽을 수 있도록 각 스크립트를 편집합니다.

`Term::ReadKey` 모듈이 없어도 Perl 스크립트의 다른 모든 기능은 계속 사용할 수 있습니다.

Directory Server가 최소 구성으로 시작됩니다.

## 제거

제거 프로그램은 소프트웨어와 관련 데이터를 컴퓨터에서 삭제하므로 Directory Server를 사용할 수 없게 되고 모든 설정과 데이터가 손실됩니다.

제거 프로그램은 서버 소프트웨어뿐만 아니라 시스템에 저장된 레지스트리 데이터까지 모두 삭제합니다. 제거 프로그램을 사용하기 전에 수동으로 파일을 삭제하면 레지스트리가 손상될 수 있습니다. 레지스트리 손상을 방지하려면 수동으로 제품의 파일을 삭제하기 전에 제거 프로그램을 사용하십시오.

---

**주** `o=NetscapeRoot` 접미사로 구성 정보가 들어 있는 구성 디렉토리를 제거하는 경우에는 경고가 표시되지 않습니다.

다른 디렉토리가 구성 정보를 확인하는 중앙의 구성 디렉토리를 제거하면 다른 디렉토리를 관리할 수 없게 됩니다.

---

해당 절의 지침에 따라 수행합니다.

- Solaris 시스템에서 제거

- 기타 UNIX 시스템에서 제거
- Windows 시스템에서 제거

## Solaris 시스템에서 제거

Directory Server 소프트웨어의 제거 방법은 설치 프로세스에 사용한 패키지 및 대화형 제거의 사용 여부에 따라 결정됩니다. 해당 절의 지침에 따라 수행합니다.

- Solaris 패키지를 사용한 설치 후에 대화형 제거
- 압축된 아카이브를 사용한 설치 후에 대화형 제거
- Solaris 패키지를 사용한 설치 후에 자동 제거
- 압축된 아카이브를 사용한 설치 후에 자동 제거

### Solaris 패키지를 사용한 설치 후에 대화형 제거

해당 절의 지침에 따라 수행합니다.

- Directory Server의 이전 버전 제거
- 관리 서버 구성 해제
- Directory Server 구성 해제
- 패키지 제거

#### *Directory Server의 이전 버전 제거*

- 중요 Solaris 시스템에서 Directory Server 5.1을 5.2로 업그레이드하는 중이며 5.1 버전이 IPLT\* Solaris 패키지에서 설치된 경우에는 아래 명령을 실행하여 5.1 버전을 제거합니다.

```
root# /usr/sbin/directoryserver.51bak uninstall
```

#### *관리 서버 구성 해제*

- 아래 명령을 실행하여 관리 서버 구성을 삭제합니다.

```
root# /usr/sbin/mpsadmserver unconfigure
```

첫 제거 화면이 나타납니다. 각 화면에 표시된 지침에 따라 수행합니다.

#### *Directory Server 구성 해제*

- 아래 명령을 실행하여 Directory Server 구성을 삭제합니다.

```
root# /usr/sbin/directoryserver unconfigure
```

첫 제거 화면이 나타납니다. 각 화면에 표시된 지침에 따라 수행합니다.

## 패키지 제거

- pkgrm(1M) 유틸리티를 사용하여 24페이지의 "Solaris 패키지를 사용한 대화형 설치"에서 설치한 패키지를 제거합니다.

## 압축된 아카이브를 사용한 설치 후에 대화형 제거

1. 아래 명령을 실행하여 *ServerRoot* 디렉토리에서 제거 프로그램을 시작합니다.

```
root# ./uninstall_dirserver
```

첫 제거 화면이 나타납니다.

2. 각 화면에 표시된 지침에 따라 수행합니다.

선택한 소프트웨어가 제거됩니다. 제거 프로그램에서 *ServerRoot* 디렉토리의 모든 파일을 제거할 수 없으면 메시지가 표시됩니다. 이런 경우 *ServerRoot*에 남아 있는 파일을 수동으로 제거할 수 있습니다.

## Solaris 패키지를 사용한 설치 후에 자동 제거

1. 제거 사양 파일(*ServerRoot/setup/uninstall.ins*)에 해당 관리자 식별자와 암호가 포함되도록 파일을 편집합니다.

### 코드 예제 1-2

샘플 제거 사양 파일

```
[STATE_BEGIN Sun ONE Directory Distribution checksum]

ConfigDirectoryAdminID = admin-user
ConfigDirectoryAdminPwd = admin-password

[STATE_DONE Sun ONE Directory Distribution checksum]
```

2. Solaris 시스템에서 Directory Server 5.1을 5.2로 업그레이드하는 중이며 5.1 버전이 IPLT\* Solaris 패키지에서 설치된 경우에는 아래 명령을 실행하여 5.1 버전을 제거합니다.

```
root# /usr/sbin/directoryserver.51bak uninstall -f 51-uninstaller-file
```

3. 아래의 unconfigure 하위 명령을 실행하여 관리 서버 구성을 삭제합니다.

```
root# /usr/sbin/mpsadmserver unconfigure -f ServerRoot/setup/uninstall.ins
```

4. 아래의 unconfigure 하위 명령을 실행하여 Directory Server 구성을 삭제합니다.

```
root# /usr/sbin/directoryserver unconfigure -f ServerRoot/setup/uninstall.ins
```

5. pkgrm(1M) 유틸리티를 사용하여 29페이지의 "Solaris 패키지를 사용한 자동 설치"에서 설치한 패키지를 제거합니다.

제거를 완료한 후 남아 있는 파일은 수동으로 제거할 수 있습니다.

### 압축된 아카이브를 사용한 설치 후에 자동 제거

1. 43페이지의 코드 예제 1-2에 표시된 것처럼 제거 사양 파일 (*ServerRoot/setup/uninstall.ins*)에 해당 관리자 식별자와 암호가 포함되도록 파일을 편집합니다.
2. 아래 명령을 실행하여 자동 모드로 제거 프로그램을 실행합니다.

```
root# cd ServerRoot
```

```
root# ./uninstall_dirserver -noconsole -nodisplay -state setup/uninstall.ins
```

제거를 완료한 후 남아 있는 파일은 수동으로 제거할 수 있습니다.

## 기타 UNIX 시스템에서 제거

해당 절의 지침에 따라 수행합니다.

### 대화형 제거

1. 아래 명령을 실행하여 *ServerRoot* 디렉토리에서 제거 프로그램을 시작합니다.

```
root# ./uninstall_dirserver
```

첫 제거 화면이 나타납니다.

2. 각 화면에 표시된 지침에 따라 수행합니다.

선택한 소프트웨어가 제거됩니다. 제거 프로그램에서 *ServerRoot* 디렉토리의 모든 파일을 제거할 수 없으면 메시지가 표시됩니다. 이런 경우 *ServerRoot*에 남아 있는 파일을 수동으로 제거할 수 있습니다.

## 자동 제거

1. 43페이지의 코드 예제 1-2에 표시된 것처럼 제거 사양 파일 (*ServerRoot/setup/uninstall.ins*)에 해당 관리자 식별자와 암호가 포함되도록 파일을 편집합니다.
2. 아래 명령을 실행하여 자동 모드로 제거 프로그램을 실행합니다.

```
root# cd ServerRoot
root# ./uninstall_dirserver -noconsole -nodisplay -state setup/uninstall.ins
```

제거를 완료한 후 남아 있는 파일은 수동으로 제거할 수 있습니다.

## Windows 시스템에서 제거

해당 절의 지침에 따라 수행합니다.

### 대화형 제거

1. 시작을 누른 다음 설정 > 제어판을 선택합니다.
2. 프로그램 추가/제거를 두 번 누릅니다.
3. 프로그램 추가/제거 창에서 Directory Server를 선택한 다음 제거를 누릅니다.
4. Sun ONE 제거 창에 표시된 지침에 따릅니다.

Directory Server를 업그레이드한 경우 사용자 정의 제거 모드를 사용하여 새 Directory Server 인스턴스에서 공유하는 .dll이 포함된 Basic System Libraries를 제거하지 않도록 선택합니다.

### 자동 제거

1. 43페이지의 코드 예제 1-2에 표시된 것처럼 제거 사양 파일 (*ServerRoot/setup/uninstall.ins*)에 해당 관리자 식별자와 암호가 포함되도록 파일을 편집합니다.
2. 아래 명령을 실행하여 자동 모드로 제거 프로그램을 실행합니다.

```
Prompt>cd ServerRoot
Prompt>uninstall_dirserver -noconsole -nodisplay -state setup\uninstall.ins
```

제거를 완료한 후 남아 있는 파일은 수동으로 제거할 수 있습니다.

제거 후에 Windows 시스템을 재부트해야 합니다.

# 문제 해결

**표 1-10** 일반적인 설치 문제 및 해결 방법

문제	해결 방법
라이브러리가 없다는 메시지가 표시됩니다.	idsktune을 실행하고 모든 권장 패치를 설치하여 최소한 ERROR 조건은 모두 수정하십시오.
설치 프로그램이 작동하지 않으며 제거할 수도 없습니다. 어떻게 해야 하나요?	<p><i>다른 제품에 영향을 주지 않는다면</i> 다음과 같은 제품 레지스트리 파일을 제거합니다.</p> <ul style="list-style-type: none"> <li>• 슈퍼유저로 설치하는 경우 Solaris 시스템의 <code>/var/sadm/install/productregistry</code></li> <li>• 기타 UNIX 시스템의 <code>/var/tmp/productregistry</code></li> <li>• Windows 시스템 폴더 아래의 <code>system32</code> 폴더에 있는 <code>productregistry</code>(예: Windows의 <code>C:\WINNT\system32\productregistry</code>)</li> </ul> <p>그런 다음, 소프트웨어를 다시 설치하기 전에 부분 설치된 파일을 수동으로 제거합니다.</p>
설치에 실패했으며 이유를 알 수 없습니다. 설치 로그가 있습니까?	<p>예. 로그 파일은 아래 위치에서 확인할 수 있습니다.</p> <ul style="list-style-type: none"> <li>• Solaris 시스템의 경우 <code>/var/sadm/install/logs</code>(슈퍼유저로 설치) 또는 <code>/var/tmp</code>(일반 사용자로 설치)</li> <li>• 기타 UNIX 시스템의 경우 <code>/var/tmp</code></li> <li>• Windows 시스템의 경우 <code>%TEMP%</code> 폴더</li> </ul>
클라이언트에서 서버를 찾을 수 없습니다.	<p><code>dirserv</code>와 같은 호스트 이름을 사용해 보십시오.</p> <p>그래도 문제가 계속되면 해당 서버가 사용하는 이름 서비스(예: DNS)에 표시되어 있는지 확인한 다음 <code>dirserv.example.com</code>과 같은 전체 도메인 이름을 사용합니다.</p> <p>위의 두 가지 방법이 모두 효과가 없으면 <code>192.168.0.30</code>과 같은 호스트 IP 주소를 사용해 보십시오.</p>
포트가 사용 중입니다.	<p>업그레이드 중인 경우라면 Directory Server를 업그레이드하기 전에 종료하지 않았기 때문입니다. 이전 서버를 종료한 다음 업그레이드된 서버를 수동으로 시작하십시오.</p> <p>그렇지 않으면 다른 서버가 해당 포트를 사용하고 있기 때문입니다. UNIX 시스템인 경우 <code>netstat(1M)</code> 유틸리티를 <code>-a</code> 옵션과 함께 사용하는 등 적절한 도구를 통해 사용 중인 포트를 검사하고 사용할 수 있는 포트를 확인합니다.</p>

**표 1-10** 일반적인 설치 문제 및 해결 방법 (계속)

문제	해결 방법
LDAP 인증 오류로 인해 설치에 실패했습니다.	<p>설치 중에 <code>dirserv.example.com</code> 대신 <code>dirserv.nisDomain.Example.COM</code>을 사용하는 등 전체 도메인 이름을 잘못 입력했을 수 있습니다.</p>
디렉토리 관리자 DN과 암호를 잊어버렸습니다.	<p>디렉토리 관리자 DN은 <code>ServerRoot/slapd-serverID/config/dse.ldif</code>에 <code>nsslapd-rootdn</code> 값으로 기록되어 있습니다.</p> <p>디렉토리 관리자 암호는 <code>dse.ldif</code>에 <code>nsslapd-rootpw</code> 값으로 기록되어 있습니다. 암호는 반드시 암호화하는 것이 좋지만 암호화되어 있지 않은 암호는 접두사로 <code>{SSHA}</code>와 같은 암호화 체계 식별자가 붙어 있지 않고 <code>dse.ldif</code>에 일반 텍스트로 표시됩니다.</p> <p>암호가 암호화되어 있으면 수동으로 문제를 해결해야 합니다.</p> <ol style="list-style-type: none"> <li>1. Directory Server를 중지합니다.</li> <li>2. 뒤에 공백을 추가하지 않도록 주의하여 <code>dse.ldif</code>의 <code>nsslapd-rootpw</code> 값을 변경합니다.</li> <li>3. <code>dse.ldif</code>를 저장하고 닫습니다.</li> <li>4. 서버를 다시 시작합니다.</li> <li>5. <code>nsslapd-rootpw</code>에 지정한 값을 사용하여 디렉토리 관리자로 로그인합니다.</li> <li>6. <i>Sun ONE Directory Server 관리 설명서</i>에 설명된 것처럼 디렉토리 관리자 암호에 대해 암호화 체계를 설정하고 다시 암호를 변경합니다.</li> </ol>

**표 1-10** 일반적인 설치 문제 및 해결 방법 (계속)

문제	해결 방법
<p>실수로 Directory Server의 32비트 버전을 설치했습니다.</p> <p>64비트 버전을 실행하려면 어떻게 해야 합니까?</p>	<ol style="list-style-type: none"> <li>1. <i>Sun ONE Directory Server 관리 설명서</i>에 설명된 것처럼 모든 접미사를 LDIF로 내보냅니다.</li> <li>2. 데이터베이스 파일을 모두 삭제합니다. 데이터베이스 파일은 가령 <code>cn=config, cn=ldbm database, cn=plugins, cn=config</code>에서 <code>nsslapd-directory</code> 값으로 지정된 경로에 있습니다.</li> <li>3. 아직 설치하지 않았으면 64비트 구성 요소를 설치합니다.</li> <li>4. <code>ServerRoot/bin/slapd/server/64/ns-slapd</code>를 실행할 수 있도록 설정합니다.</li> <li>5. 32비트 모드로 운영 체제를 실행 중이면 64비트 모드로 재부트합니다.</li> <li>6. 필요한 경우 32비트 모드에서 작동하도록 캐시 크기 설정을 변경합니다. 자세한 내용은 6장, "캐시 크기 조정"을 참조하십시오.</li> <li>7. <i>Sun ONE Directory Server 관리 설명서</i>에 설명된 것처럼 내보낸 LDIF를 사용하여 모든 접미사를 초기화합니다.</li> <li>8. 서버를 다시 시작합니다.</li> </ol>
<p>실수로 Directory Server의 64비트 버전을 설치했습니다.</p> <p>32비트 버전을 실행하려면 어떻게 해야 합니까?</p>	<ol style="list-style-type: none"> <li>1. <i>Sun ONE Directory Server 관리 설명서</i>에 설명된 것처럼 모든 접미사를 LDIF로 내보냅니다.</li> <li>2. 데이터베이스 파일을 모두 삭제합니다. 데이터베이스 파일은 가령 <code>cn=config, cn=ldbm database, cn=plugins, cn=config</code>에서 <code>nsslapd-directory</code> 값으로 지정된 경로에 있습니다.</li> <li>3. <code>ServerRoot/bin/slapd/server/64/ns-slapd</code>를 실행할 수 없도록 모드를 변경합니다.</li> <li>4. <i>Sun ONE Directory Server 관리 설명서</i>에 설명된 것처럼 내보낸 LDIF를 사용하여 모든 접미사를 초기화합니다.</li> <li>5. 서버를 다시 시작합니다.</li> </ol>



**표 1-10** 일반적인 설치 문제 및 해결 방법 (계속)

문제	해결 방법
설치 처리 스크립트를 작성한 다음 이 스크립트를 사용하여 설치를 시도하자 0이 아닌 73이 반환되었습니다.	설치 프로그램 반환 코드는 다음과 같습니다.
이 반환 코드는 어떤 의미입니까?	<ul style="list-style-type: none"> <li>0 - SUCCESS</li> <li>1 - WARNING_REBOOT_REQUIRED</li> <li>2 - WARNING_PLATFORM_SUPPORT_LIMITED</li> <li>3 - WARNING_RESOURCE_NOT_FOUND</li> <li>4 - WARNING_CANNOT_WRITE_LOG</li> <li>5 - WARNING_LOCALE_NOT_SUPPORTED</li> <li>50 - ERROR_FATAL</li> <li>51 - ERROR_ACCESS</li> <li>52 - ERROR_PLATFORM_NOT_SUPPORTED</li> <li>53 - ERROR_NO_WINDOWING_SYSTEM_AVAILABLE</li> <li>54 - ERROR_RESOURCE_NOT_FOUND</li> <li>55 - ERROR_TASK_FAILURE</li> <li>56 - ERROR_USER_EXIT</li> <li>57 - ERROR_CANNOT_UPGRADE</li> <li>58 - ERROR_NOTHING_TO_DO</li> <li>59 - ERROR_IN_SERIALIZATION</li> <li>60 - ERROR_ABNORMAL_EXIT</li> <li>61 - ERROR_INCOMPATIBLE_STATEFILE</li> <li>62 - ERROR_UNKNOWN_COMMANDLINE_OPTION</li> <li>70 - ERROR_NOT_INSTALLED</li> <li>71 - PARTIALLY_UNINSTALLED</li> <li>72 - FULLY_UNINSTALLED</li> <li>73 - INSTALLED</li> <li>74 - ERROR_FAILED</li> <li>75 - ERROR_STOPPED</li> <li>76 - ERROR_STOPPED_ON_ERROR</li> <li>77 - PARTIALLY_INSTALLED</li> </ul>
	즉, 73은 성공적으로 설치되었음을 나타냅니다.

문제 해결

## 이전 버전에서 업그레이드

이 장에서는 Netscape Directory Server 4.x 및 iPlanet Directory Server 5.x를 Sun ONE Directory Server 5.2로 업그레이드하는 방법에 대해 설명합니다.

---

**주** Innosoft Distributed Directory Server 4.5.1에서 업그레이드하는 방법은 포함되어 있지 않습니다.

---

이 장에서는 이전 서버의 디렉토리 데이터를 새 서버로 이전하는 방법을 중점적으로 설명합니다. 이전 서버에서 새 서버로 이전되는 구성 속성에 대한 자세한 내용은 *Sun ONE Directory Server Reference Manual*을 참조하십시오.

## 업그레이드 시 주의 사항

업그레이드하기 전에 먼저 12페이지의 "관련 자료"에 설명된 Sun ONE Directory Server 5.2의 새 기능을 살펴보십시오. 기존 디렉토리 서비스를 구현할 때 사용했던 설계 설정을 확인해 보는 것도 좋습니다.

### 단일 서버 인스턴스 업그레이드

서버 인스턴스를 업그레이드하려면 기존 서버를 그대로 두고 다른 *serverID*와 다른 관리 서버 및 Directory Server 포트 번호를 사용하여 다른 *ServerRoot*에 새 서버를 설치한 다음, 이전 서버를 중지하고 구성 및 디렉토리 데이터를 이전하여 클라이언트가 새 서버로 요청을 보내도록 설정해야 합니다.

---

**주** 기존 서버를 실행하는 호스트의 디스크 공간이 충분한지 확인합니다. 업그레이드 프로세스에는 최소한 이전 서버와 새 서버의 이전 및 데이터베이스를 모두 수용할 만한 로컬 디스크 공간이 있어야 하며, 기존 접미사의 항목이 모두 포함된 LDIF 파일을 저장할 추가 공간도 필요합니다. 필요한 로컬 디스크 공간은 아래 수식에 의해 계산된 값보다 약간 여유있게 잡는 것이 좋습니다.

$$2 * (\text{기존 서버 공간}) + (\text{LDIF 파일 저장 공간})$$

네트워크 드라이브를 통해 데이터를 이전할 수 없으므로 두 서버를 같은 호스트에 설정하여 업그레이드 프로세스를 수행해야 합니다.

---

Sun ONE Directory Server 5.2는 서버 인스턴스 데이터를 이전할 수 있도록 도와주는 스크립트를 제공합니다. 이전 스크립트는 다음 작업을 순서대로 수행합니다.

1. 기존 서버를 중지하고 현재 구성을 백업합니다.
2. 스키마 구성 파일을 확인하여 표준 스키마 구성 파일과 기존 서버에서 사용한 스키마 구성 파일 간의 차이점을 알려줍니다.  
  
(4.x를 5.2로 업그레이드하는 경우에만 해당) 기존 4.x 서버가 `ServerRoot/slaped-serverID/config`의 기본 위치에 설치되지 않은 사용자 정의 스키마를 사용하는 경우 디렉토리 데이터를 이전하기 전에 수동으로 구성을 조정해야 합니다.
3. 이전 서버에 저장된 각 접미사에 대한 데이터베이스를 작성합니다.  
  
(4.x를 5.2로 업그레이드하는 경우에만 해당) 4.x 서버는 한 데이터베이스에서 여러 개의 접미사를 지원했습니다. 이전 스크립트는 각 접미사에 대한 데이터베이스를 새 서버에 작성합니다.
4. 서버 및 데이터베이스 구성 매개 변수를 이전합니다.  
  
4.x 서버는 `slapd.conf`에 이러한 매개 변수를 저장하고 5.x 서버는 `dse.ldif`에 저장합니다.

---

**주** 이 스크립트는 `o=NetscapeRoot` 데이터를 이전하지 않습니다.

Sun ONE Messaging Server와 같이 이 접미사의 데이터를 사용하는 서버를 배포하는 경우에는 해당 서버와 함께 제공된 도구를 이용하거나 수동으로 `o=NetscapeRoot` 데이터를 이전해야 합니다.

---

(4.x를 5.2로 업그레이드하는 경우에만 해당) 이전 스크립트는 4.x 서버 매개 변수 중 일부만 이전하므로 4.x 속성 값을 수동으로 이전해야 하는 경우도 있습니다. 자세한 내용은 *Sun ONE Directory Server Reference Manual*의 현재 버전을 참조하십시오.

5. 사용자 정의된 스키마 개체를 이전합니다.
6. 색인을 이전합니다.
7. 표준 서버 플러그 인을 이전합니다.

사용자 정의 플러그 인은 수동으로 이전해야 하며, 최소한 모든 사용자 정의 플러그 인을 다시 컴파일해야 합니다. 플러그 인 API 변경 사항에 대한 자세한 목록은 *Sun ONE Directory Server Plug-In API Programming Guide*를 참조하십시오.

8. (5.x를 5.2로 업그레이드하는 경우에만 해당) 복제 계약을 이전합니다.

---

## 주

5.2 Directory Server에서 5.1 서버로 복제하기 전에 `cn=config`의 `nsslapd-schema-repl-useronly`를 `on`으로 설정하십시오. 그렇지 않으면 5.2 스키마가 5.1 서버로 푸시(push)되고 복제 개체로 인해 5.1 서버가 다시 시작되지 않습니다.

---

9. 인증서 데이터베이스와 SSL 매개 변수를 이전합니다.
10. (5.x를 5.2로 업그레이드하는 경우에만 해당) 데이터베이스 링크를 이전합니다.
11. (5.x를 5.2로 업그레이드하는 경우에만 해당) 복제 항목을 이전합니다.
12. SNMP 구성을 이전합니다.

이전 스크립트가 완료되면 클라이언트에서 새 서버로 요청을 보낼 수 있습니다.

## 여러 개의 복제된 서버 업그레이드

여러 개의 서버를 업그레이드하려면 각 서버를 개별적으로 업그레이드해야 합니다. 그러나 서버 업그레이드 순서는 기존 서버의 소프트웨어 버전과 복제 토폴로지에 따라 달라집니다.

5.x를 5.2로 업그레이드하는 경우 표준 프로세스는 상향식으로 진행됩니다. 먼저 고객을 이전한 다음 허브를 업그레이드하고 마지막으로 마스터를 업그레이드합니다. 특정 인스턴스에서 이러한 작업을 수행하는 방법은 67페이지의 "5.x 업그레이드 시나리오 예"를 참조하십시오.

4.x를 5.2로 업그레이드하는 경우 먼저 4.x 마스터를 업그레이드한 다음, 복제 측면에서 마스터와 가장 가까운 소비자부터 시작하여 마스터에서 복제되는 각 소비자 분기를 업그레이드합니다. 특정 인스턴스에서 이러한 작업을 수행하는 방법은 60페이지의 "4.x 업그레이드 시나리오 예"를 참조하십시오.

기존 환경에 여러 개의 복제된 서버가 포함되어 있으면 업그레이드를 시작하기 전에 이 장에 소개된 모든 관련 절을 주의깊게 읽어 보십시오. 불필요한 시스템 중단 시간을 줄이려면 완벽하게 업그레이드 계획을 세워야 합니다.

## 업그레이드 지원

중요한 디렉토리 서비스를 업그레이드하는 경우 Sun Professional Services의 도움을 받을 수 있습니다.

연락처 정보는 <http://www.sun.com/service/sunps/sunone/>을 참조하십시오.

## 단일 서버 업그레이드

이 절에서는 기존의 단일 서버를 단일 5.2 서버로 업그레이드하는 프로세스에 대해 설명합니다.

- 
- 주** 기존 4.x 서버가 사용자 정의 스키마를 사용하는 경우 데이터를 이전하기 전에 이전 스크립트가 사용자 정의 스키마를 찾을 수 있는지 확인합니다. 자세한 내용은 55페이지의 "(4.x를 5.2로 업그레이드하는 경우) 사용자 정의 스키마 처리"를 참조하십시오.
- 이전 스크립트가 사용자 정의 스키마를 인식하지 못하면 스키마가 이전되지 않으며, 새 서버로 데이터가 이전된 후 표준 스키마 파일이 적용됩니다. 사용자 정의 스키마를 따르는 항목에 표준 스키마를 적용하면 항목을 수정할 수 없게 되므로 업그레이드된 디렉토리가 읽기 전용으로 설정됩니다.
- 

## 새 서버 설치

1장, "Sun ONE Directory Server 설치"에 설명된 지침에 따라 기존 서버와 같은 호스트에 새 서버를 설치합니다.

- 
- 주** 새 서버를 설치하기 전에 기존 서버가 백업되어 있는지 확인합니다.
- 

새 서버가 설치되는 *ServerRoot*는 기존 서버의 위치와 달라야 합니다. 또한 새 서버는 다른 *serverID*로 식별되어야 합니다.

초기 설치에 제공했던 대부분의 구성 정보를 다시 사용하도록 선택할 수 있지만 기존 포트 번호는 다시 사용하지 마십시오. 기존 데이터를 이전한 후 새 서버의 포트 번호를 변경할 수 있습니다.

## (4.x를 5.2로 업그레이드하는 경우) 사용자 정의 스키마 처리

데이터 이전 스크립트는 표준 `slapd.user_oc.conf` 및 `slapd.user_at.conf` 파일에 있는 사용자 정의 스키마와 다른 파일에 있지만 `useroc` 및 `userat` 지시문을 사용하여 `slapd.conf` 파일에 추가된 사용자 정의 스키마만 인식합니다. 예를 들어, `slapd.at.conf` 또는 `slapd.oc.conf` 파일에 직접 추가된 사용자 정의 스키마는 이전 스크립트에서 인식하지 못합니다.

업그레이드를 수행하기 전에 다음 단계를 완료합니다.

1. `slapd.at.conf` 또는 `slapd.oc.conf` 파일과 새 서버의 `ServerRoot/bin/slapd/install/version4/`에 있는 표준 파일을 비교하여 사용자 정의 스키마 요소를 `slapd.user_oc.conf` 및 `slapd.user_at.conf` 파일에 추가합니다.  
  
사용자 정의 개체 클래스 간에 상속 관계가 있으면 스키마 구성 파일에서 상위 개체 클래스가 하위 개체 클래스보다 앞에 오게 하십시오.
2. `slapd.oc.conf` 파일의 표준 개체 클래스에 사용자 정의 속성을 추가한 경우 `slapd.user_oc.conf` 파일에 해당 속성을 포함하는 새 개체 클래스를 작성하고 기존 디렉토리에서 이 사용자 정의 속성을 사용하는 모든 항목에 새 개체 클래스를 추가합니다.
3. 다른 파일에 대한 `include` 문 근처에서 `useroc` 및 `userat` 지시문을 사용하여 기존 서버의 `slapd.conf` 파일에 `slapd.user_oc.conf` 및 `slapd.user_at.conf` 파일을 추가합니다.

이제 기존 서버에서 사용한 모든 사용자 정의 스키마는 `slapd.user_oc.conf` 또는 `slapd.user_at.conf` 파일에 있어야 하고 `slapd.conf` 파일에는 `useroc` 및 `userat` 지시문을 사용하여 두 파일이 추가되어 있어야 합니다.

## 기존 데이터 이전

사용자 정의 스키마를 처리한 후 다음 단계를 수행하여 기존 데이터를 새 서버로 이전합니다.

1. 오프라인 상태로 파일을 사용하여 새 Directory Server에서 복제를 초기화하려면 먼저 파일을 확보합니다.

Directory Server 데이터 내보내기에 대한 자세한 내용은 *Sun ONE Directory Server 관리 설명서*를 참조하십시오.

2. 새 Directory Server를 실행 중인지 확인합니다.
3. 이전 서버와 새 서버에서 데이터베이스 내보내기와 가져오기를 시작, 중지 및 실행할 수 있는 권한이 있는 사용자로 작업합니다.

예를 들어, 슈퍼유저가 되거나 Administrator로 로그인합니다.

4. 표 2-1에 표시된 것처럼 환경 변수를 설정합니다.

**표 2-1** 이전 환경 변수

변수	값
PATH	(UNIX) <i>ServerRoot</i> /bin/slapd/admin/bin:\$PATH (Windows) <i>ServerRoot</i> /bin/slapd/admin/bin;%PATH%
PERL5LIB	<i>ServerRoot</i> /bin/slapd/admin/bin

5. 아래 명령을 실행하여 새 서버 인스턴스의 이전 스크립트를 실행합니다.

```
# cd ServerRoot/bin/slapd/admin/bin
# perl migrateInstance5 -p port52 -D "cn=directory manager" -w password -o oldServ -n newServ
```

여기서 *oldServ*는 /usr/iplanet/servers/slapd-ldap 또는 /usr/iplanet/ds5/slapd-ldap와 같이 이전 서버 인스턴스에 대한 전체 경로를 나타내고 *newServ*는 /var/ds/v5.2/slapd-dirserv와 같이 새 서버 인스턴스에 대한 전체 경로를 나타냅니다.

스크립트가 실행되면서 출력이 생성됩니다. 이전이 완료된 후에 확인할 수 있도록 이 출력을 파일로 리디렉션할 수 있습니다.

기존 데이터를 새 서버로 이전한 후에 이전 서버를 삭제하십시오.

## (4.x를 5.2로 업그레이드하는 경우에만 해당) 복제 계약 작성

기존 4.x 서버가 복제에 관련되어 있는 경우 소프트웨어를 업그레이드하려면 데이터를 이전한 후 복제 계약을 다시 작성해야 합니다. 업그레이드 프로세스를 수행하기 전에 먼저 57페이지의 "(4.x를 5.2로 업그레이드하는 경우) 복제된 서버 업그레이드"를 읽어 보십시오.



5.2 서버 복제 구성에 대한 자세한 내용은 *Sun ONE Directory Server 관리 설명서*를 참조하십시오.

## (선택 사항) 기존 포트 번호 다시 사용

이전 서버의 데이터를 새 서버로 이전한 후 이전 서버를 삭제하고 새 서버가 이전 서버와 같은 포트에서 수신하도록 선택할 수 있습니다. 같은 포트를 사용하면 구성을 변경하지 않아도 클라이언트 응용 프로그램이 계속 작동할 수 있습니다.

서버 포트 변경에 대한 자세한 내용은 *Sun ONE Directory Server 관리 설명서*를 참조하십시오. 새 서버가 이전 포트에서 수신을 시작하기 전에 이전 서버를 중지해야 합니다.

# (4.x를 5.2로 업그레이드하는 경우) 복제된 서버 업그레이드

복제된 4.x 서버를 업그레이드하는 경우 먼저 새 마스터에 복제한 다음, 복제 토폴로지를 통해 분기별로 업그레이드합니다. 이 방법을 사용하면 서버 동기화 트래픽 양이 제한됩니다.

---

**주** 복제 구성 및 초기화에 대한 자세한 내용은 *Sun ONE Directory Server 관리 설명서*를 참조하십시오.

---

## 새 마스터 준비

업그레이드 중에 5.2 서버는 마스터로 구성되지만 4.x 토폴로지에서 레거시 소비자로 작동합니다. 업그레이드가 끝나면 4.x 소비자 기능이 비활성화되고 새 서버는 5.2 토폴로지에서 마스터로 작동합니다.

이 과정에서 새 마스터 서버를 수동으로 구성해야 하므로 기존 마스터와 다른 호스트에 새 마스터를 설치할 수 있습니다.

1. 1장, "Sun ONE Directory Server 설치"에 설명된 지침에 따라 새 서버를 설치합니다.
2. 새 서버에 4.x 마스터 구성을 수동으로 복제합니다.
3. 새 서버를 5.2 토폴로지의 마스터로 설정합니다.

자세한 내용은 *Sun ONE Directory Server 관리 설명서*를 참조하십시오.

4. 새 서버를 4.x 토폴로지의 4.x 마스터 레저시 소비자로 설정합니다.  
자세한 내용은 *Sun ONE Directory Server 관리 설명서*를 참조하십시오.
5. 4.x 마스터로부터 새 서버로의 복제를 초기화합니다.

이 프로세스는 *Netscape Directory Server Administration Guide*의 Chapter 13, "Managing Replication"에 설명되어 있습니다. "Manual Consumer Initialization" 절을 참조하십시오.

이제 소비자를 업그레이드할 수 있습니다.

## 소비자 업그레이드

아래 절차에는 소비자 업그레이드 방법이 요약되어 있습니다. 자세한 내용은 이후 절차를 참조하십시오.

1. 4.x 토폴로지의 모든 분기를 업그레이드합니다.
2. 5.2 토폴로지에 필요한 서버를 추가합니다.
3. 새 마스터에서 레저시 소비자 계약을 비활성화하여 새 토폴로지를 이전 토폴로지에서 분리합니다.

이 절차를 수행하면 업데이트 프로세스가 완료됩니다.

## 분기 업그레이드

기존 4.x 복제 토폴로지를 마스터가 루트 요소가 되는 하나의 트리로 가정해 보십시오. 여기서 분기는 복제 흐름이 루트 노드 공급자에서 시작되어 트리 중앙의 소비자를 통과하고 마지막으로 리프 노드 소비자 서버에 도착하는 트리에서 복제된 서버의 한 집합을 나타냅니다.

분기를 업그레이드하려면 하향식으로 분기의 이전 서버를 모두 새 서버로 교체합니다.

---

**주** 서버를 업그레이드하는 동안 분기의 모든 다운스트림 서버에 대한 복제 흐름은 중지됩니다. 업그레이드하는 동안에는 클라이언트 요청을 다른 분기로 리디렉션하십시오.

---

1. 54페이지의 "단일 서버 업그레이드"에 설명된 지침에 따라 분기의 최상위 서버를 업그레이드합니다.  
이렇게 하면 분기에 대한 복제 흐름이 끊겨 분기의 다운스트림 서버에 대한 복제 업데이트가 일시적으로 중지됩니다.
2. 5.2 분기의 새 서버에서 복제 계약을 구성하여 복제 토폴로지에서 새 마스터에 더 가까이 있는 5.2 서버의 업데이트를 받습니다.  
예를 들어, 새 분기의 최상위 서버가 5.2 마스터의 업데이트를 받도록 구성합니다.
3. 5.2 공급자로부터 5.2 서버로의 복제를 초기화합니다.  
네트워크 성능 및 업데이트와 디렉토리 데이터 용량 간의 비율에 따라 오프라인 초기화가 온라인 초기화보다 빠를 수 있습니다.
4. 모든 리프 소비자에 대해 단계를 완료할 때까지 분기를 따라 단계 1, 단계 2 및 단계 3을 반복적으로 수행합니다.

복제 계약 구성 및 복제 초기화에 대한 자세한 내용은 *Sun ONE Directory Server 관리 설명서*를 참조하십시오.

이제 분기에 대한 업데이트 프로세스가 완료되었습니다. 나머지 4.x 분기에 대해 이 절차를 반복하십시오.

## 서버 추가

4.x 토폴로지를 5.2 토폴로지로 업그레이드한 후 새 토폴로지에 필요한 마스터, 허브 및 소비자를 추가할 수 있습니다.

추가할 각 서버에 대해 다음 단계를 수행합니다.

1. 1장, "Sun ONE Directory Server 설치"에 설명된 지침에 따라 새 서버를 설치합니다.
2. 계획한 토폴로지에 맞게 새 서버의 복제 계약을 조정합니다.  
자세한 내용은 *Sun ONE Directory Server 관리 설명서*를 참조하십시오.
3. 새 서버에서 복제를 초기화합니다.  
자세한 내용은 *Sun ONE Directory Server 관리 설명서*를 참조하십시오.

## 4.x 업그레이드 시나리오 예

소비자 한 개가 있는 분기와 허브에서 소비자 두 개를 공급하는 분기로 복제되는 4.x 마스터의 업그레이드를 가정해 보십시오. 이 절에서는 새로운 다중 마스터 토폴로지로 업그레이드할 때 수행되는 단계를 보여줍니다.

그림 2-1은 업그레이드 전의 4.x 토폴로지입니다.

**그림 2-1**      기존의 4.x 토폴로지 예

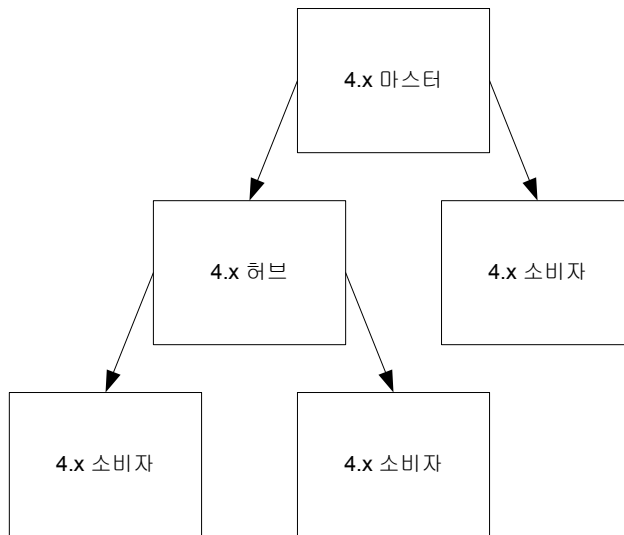


그림 2-2는 4.x 마스터의 레거시 소비자 기능을 하는 5.2 마스터가 추가된 상태를 보여줍니다.

**그림 2-2** 새 서버가 추가된 4.x 토폴로지 예

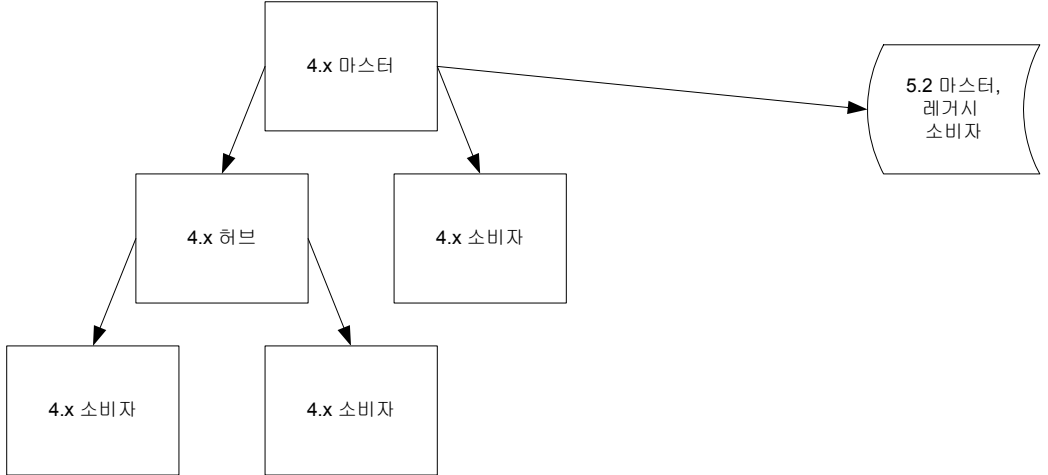


그림 2-3은 4.x 분기 교체의 첫 단계를 보여줍니다.

업그레이드하는 동안에는 전체 분기의 복제 업데이트 수신이 중지됩니다. 업데이트 수신은 업스트림 4.x 소비자가 업그레이드를 위해 중지된 때부터 4.x 소비자가 다시 시작될 때까지 중단됩니다.

지침에 설명된 것처럼 클라이언트에 최신 업데이트가 필요하면 클라이언트 요청을 다른 분기의 소비자로 보내도록 선택할 수 있습니다.

**그림 2-3** 업그레이드 중의 4.x 분기 예 - 1단계

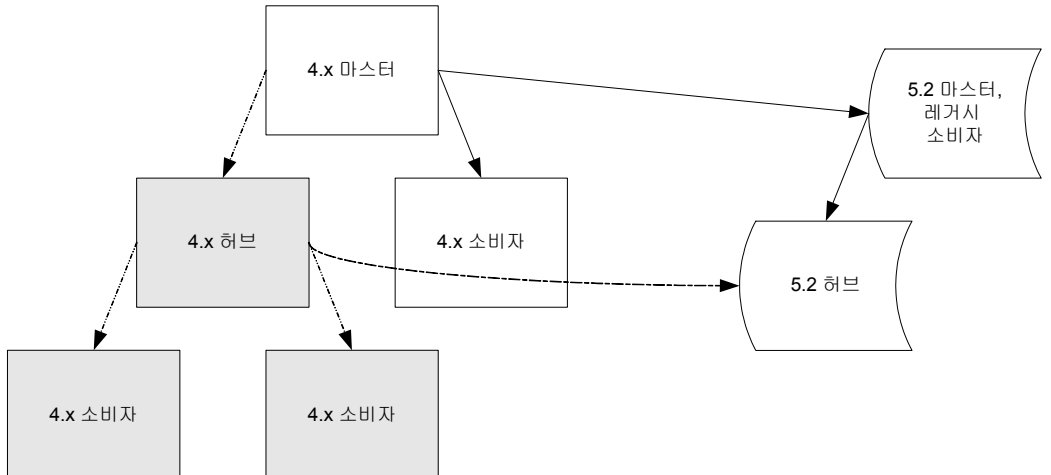


그림 2-4는 4.x 분기 교체의 다음 단계를 보여줍니다.

**그림 2-4** 업그레이드 중의 4.x 분기 예 - 2단계

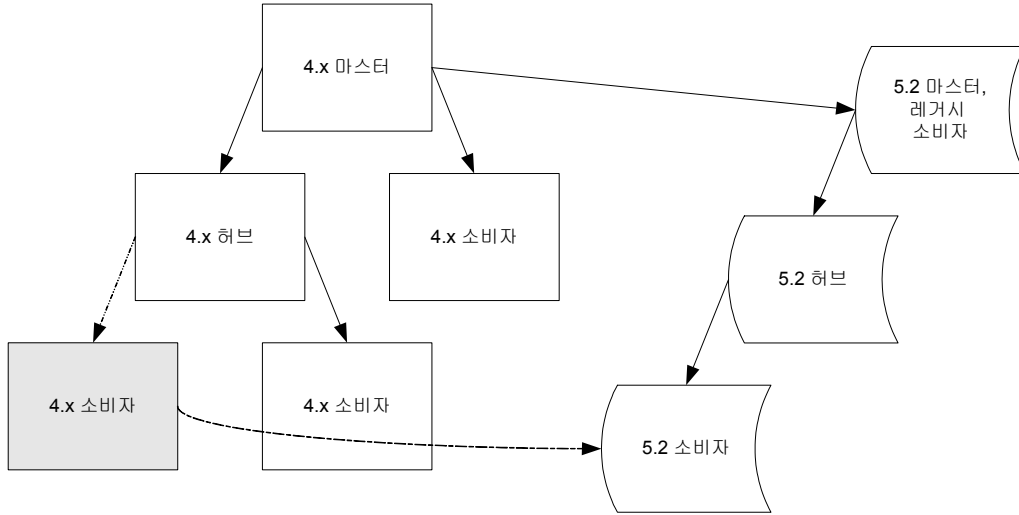


그림 2-5는 4.x 분기 교체의 다음 단계를 보여줍니다.

**그림 2-5** 업그레이드 중의 4.x 분기 예 - 3단계

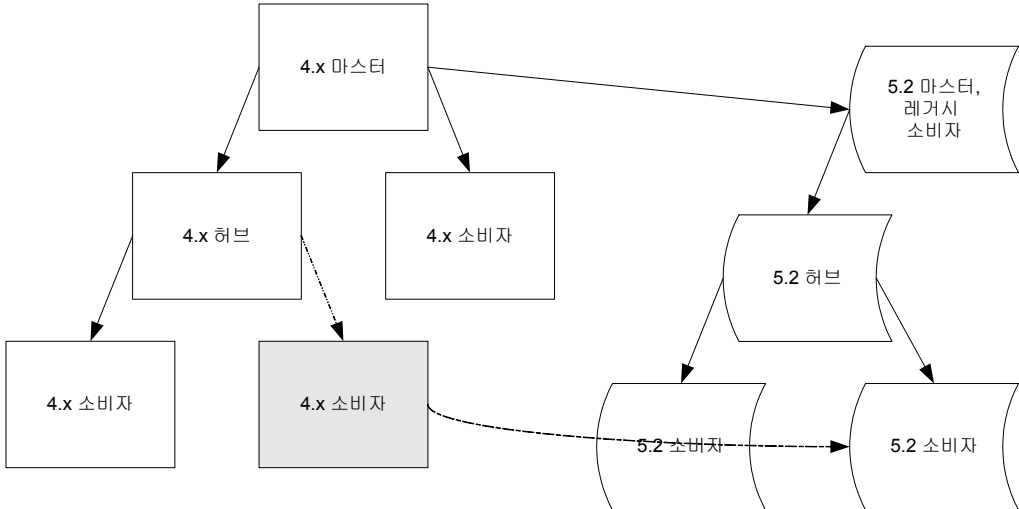


그림 2-6은 다른 4.x 분기로 교체된 상태를 보여줍니다.

그림 2-6 업그레이드 중의 4.x 분기 예 - 다음 분기

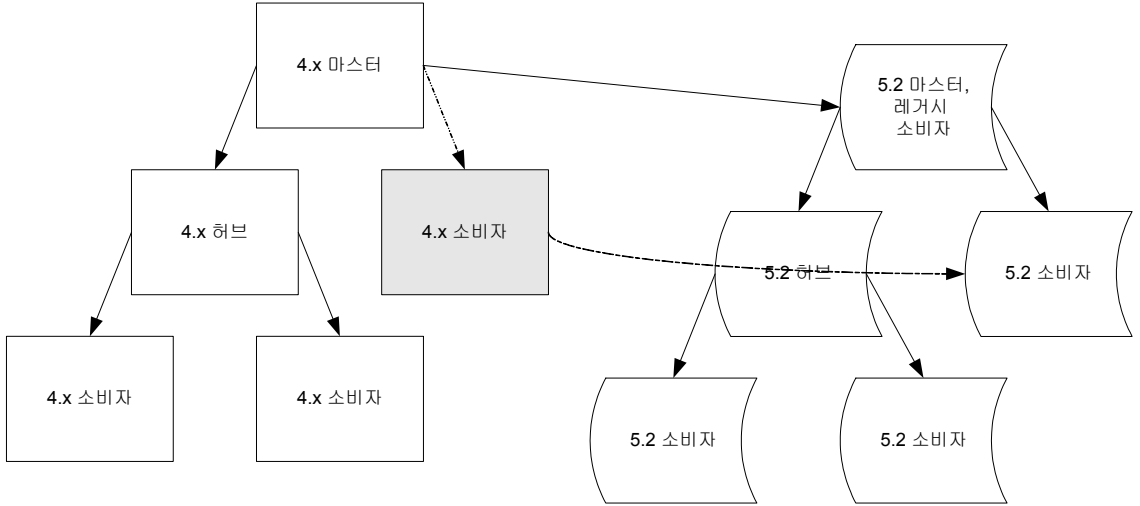


그림 2-7은 두 토폴로지를 나란히 보여줍니다.

그림 2-7 업그레이드 중의 4.x 및 5.2 토폴로지 예

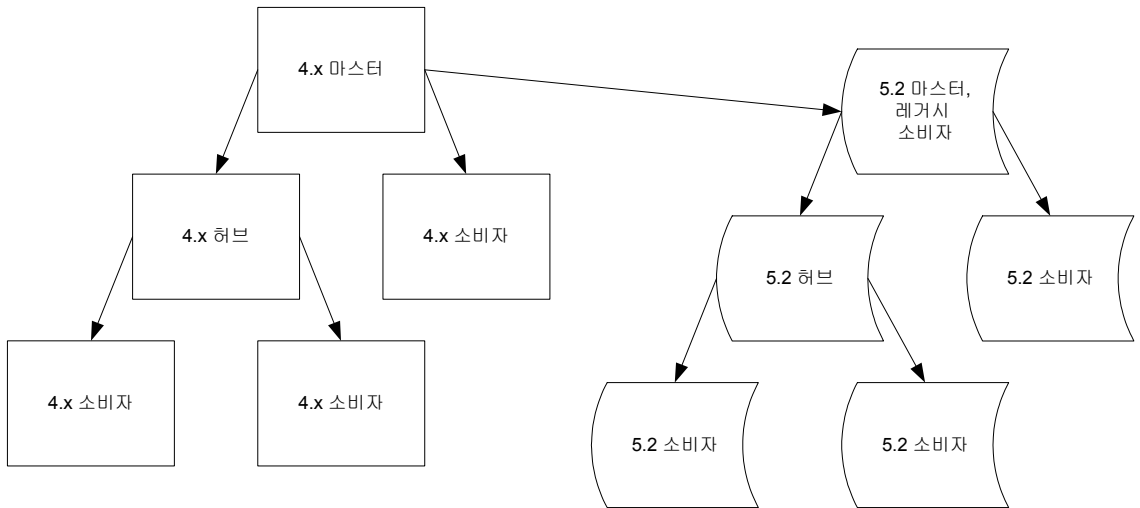
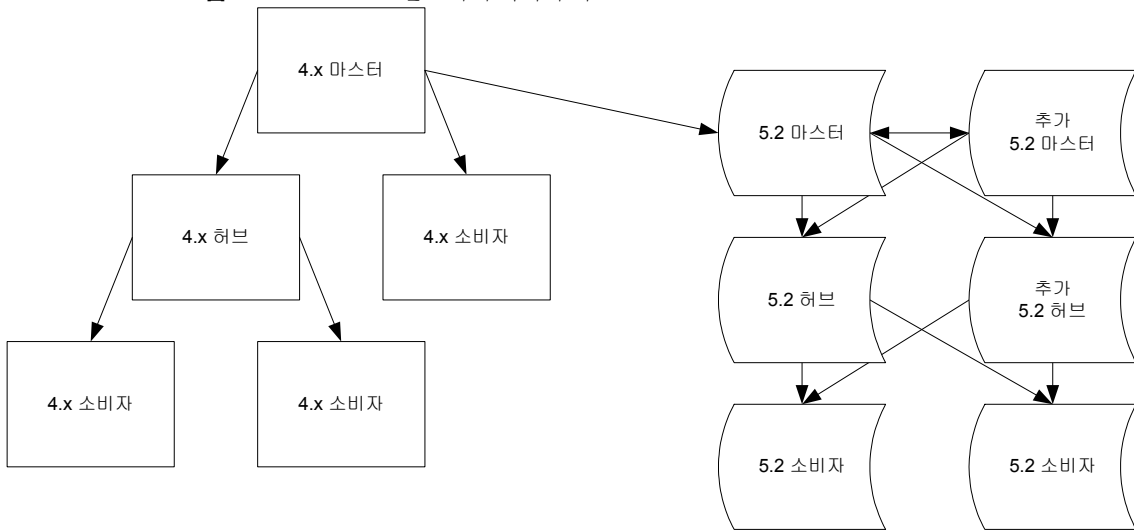


그림 2-8은 새 토폴로지에 마스터, 허브 및 새로운 복제 계약이 추가된 상태를 보여줍니다.

**그림 2-8** 5.2 토폴로지에 서버 추가

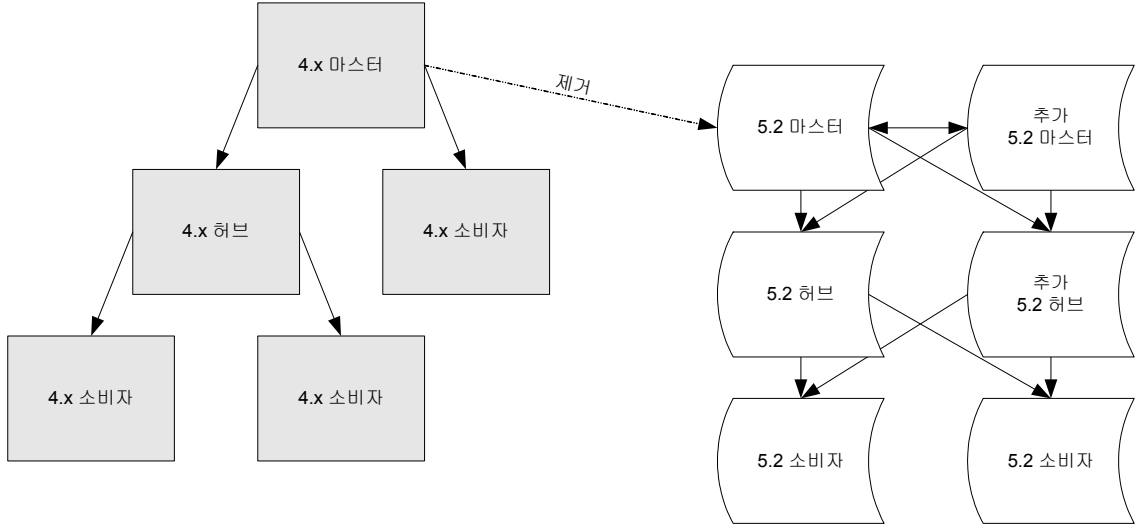


업그레이드 프로세스를 완료한 후 새 서버를 추가할 수도 있습니다.

그림 2-9는 이전 4.x 마스터로부터 새 5.2 마스터로의 복제 계약이 제거된 상태를 보여줍니다.



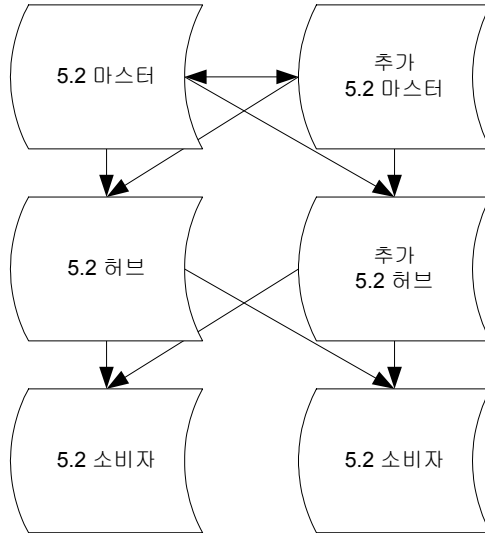
그림 2-9 복제 계약 제거



클라이언트 요청을 리디렉션하고 복제 계약을 제거한 후에는 4.x 서버를 비활성화할 수 있습니다.

그림 2-10은 완성된 5.2 토폴로지입니다.

그림 2-10 완성된 5.2 토폴로지



이제 클라이언트 요청이 5.2 토폴로지로 보내집니다.

## (5.x를 5.2로 업그레이드하는 경우) 복제된 서버 업그레이드

복제된 5.x 서버를 업그레이드하는 경우 일반적으로 소비자, 허브, 마스터 순으로 업그레이드 합니다. 이 상향식 방법에서는 복제 토폴로지의 전체 분기를 중단하지 않고 한 번에 한 개의 서버만 중단하므로 마스터와 소비자 간의 잠재적 사용자 정의 스키마 동기화 문제를 방지할 수 있습니다.

---

**주** 여기에 설명된 절차는 일반적인 5.x 토폴로지 업그레이드 방법을 사용합니다. 그러나 이 상향식 업그레이드 방법이 사용자의 특정 요구 사항에 적합하지 않으면 다른 방법을 계획하십시오.

---

## 5.x 서버 업그레이드

1. 54페이지의 "단일 서버 업그레이드"에 설명된 지침에 따라 기존 토폴로지의 각 소비자를 업그레이드합니다.
2. 같은 지침에 따라 기존 토폴로지의 각 허브를 업그레이드합니다.
3. 같은 지침에 따라 기존 토폴로지의 각 마스터를 업그레이드합니다.

## 서버 추가

5.x 토폴로지를 5.2 토폴로지로 업그레이드한 후 새 토폴로지에 필요한 마스터, 허브 및 소비자를 추가할 수 있습니다.

추가할 각 서버에 대해 다음 단계를 수행합니다.

1. 1장, "Sun ONE Directory Server 설치"에 설명된 지침에 따라 새 서버를 설치합니다.
2. 계획한 토폴로지에 맞게 새 서버의 복제 계약을 조정합니다.
3. 새 서버에서 복제를 초기화합니다.

복제 계약 구성 및 복제 초기화에 대한 자세한 내용은 *Sun ONE Directory Server 관리 설명서*를 참조하십시오.

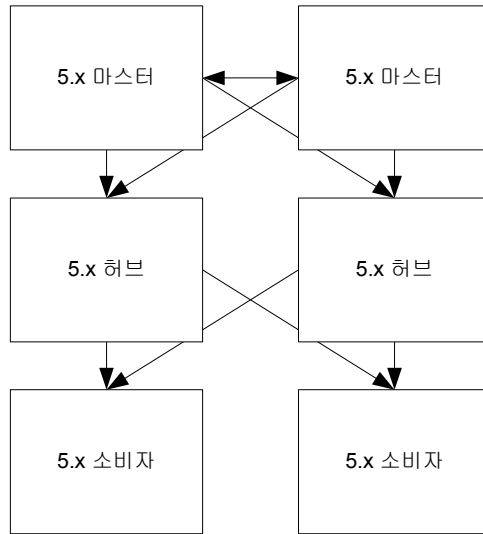
이 절차를 수행하면 업데이트 프로세스가 완료됩니다. 클라이언트가 업그레이드된 복제 토폴로지의 서버를 사용할 수 있습니다.

## 5.x 업그레이드 시나리오 예

두 소비자를 공급하는 두 개의 허브로 복제되는 5.x 이중 마스터의 업그레이드를 가정해 보십시오. 이 절에서는 5.2 서버를 사용하도록 토폴로지를 업그레이드할 때 수행되는 단계를 보여줍니다.

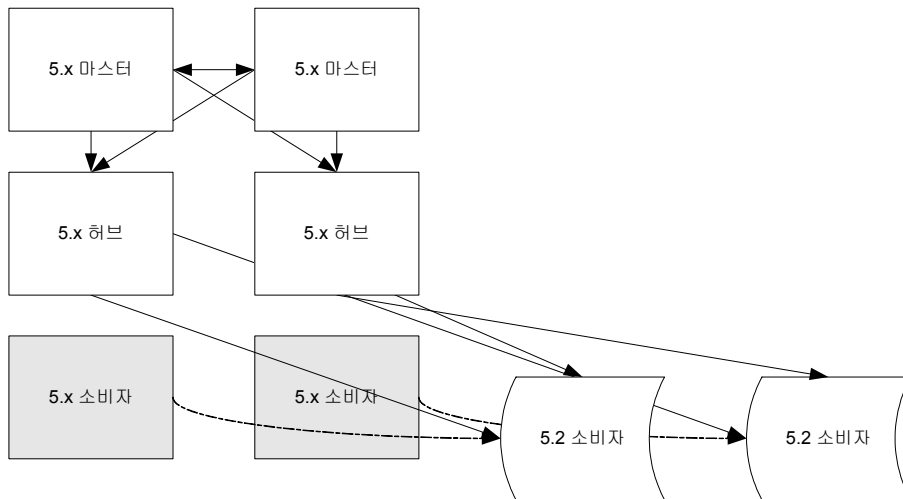
그림 2-11은 업그레이드 전의 5.x 토폴로지입니다.

**그림 2-11** 기존의 5.x 토폴로지 예



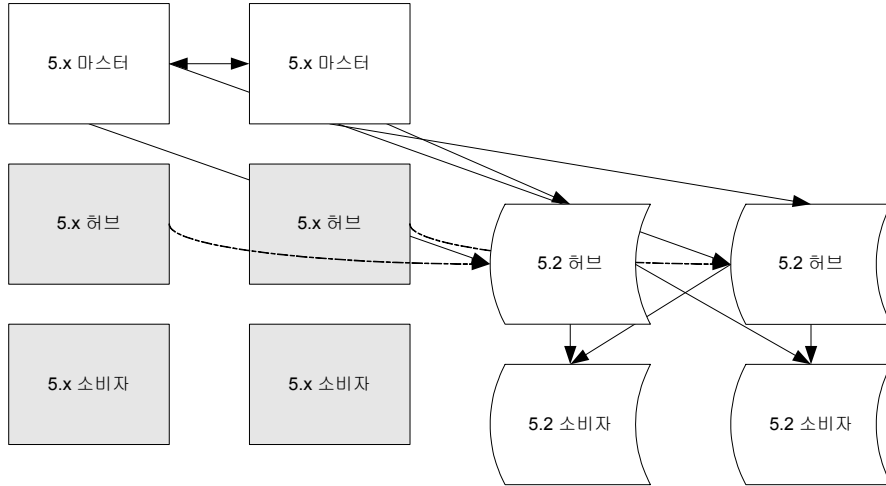
첫 단계에서는 소비자를 업그레이드합니다. 그림 2-12는 완성된 토폴로지입니다.

**그림 2-12** 5.x 소비자 업그레이드 단계 예



다음 단계에서는 허브를 업그레이드합니다. 그림 2-13은 완성된 토폴로지입니다.

**그림 2-13** 5.x 허브 업그레이드 단계 예



다음 단계에서는 마스터를 업그레이드합니다. 그림 2-14는 완성된 토폴로지입니다.

**그림 2-14** 5.x 마스터 업그레이드 예 - 3단계

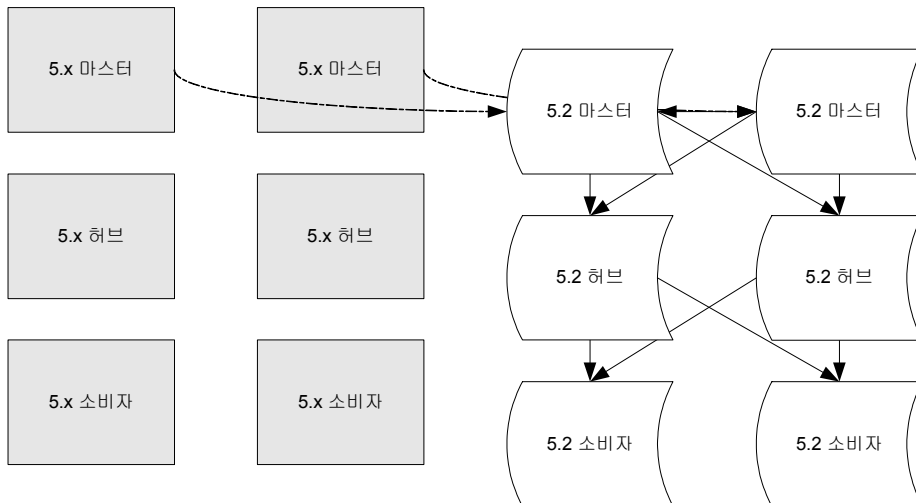
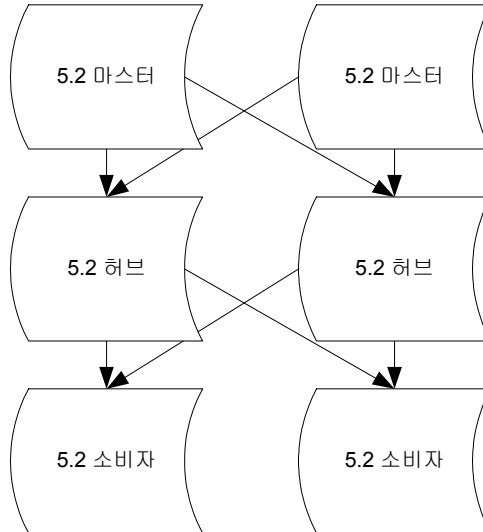


그림 2-15는 업그레이드 후의 5.2 토폴로지입니다. 이제 이전 토폴로지의 서버를 삭제할 수 있으며 5.2 토폴로지에 새 서버가 추가되었습니다.

**그림 2-15** 업그레이드 후의 5.2 토폴로지 예



이제 클라이언트 요청이 5.2 토폴로지로 보내집니다.

# 조정

3장, "유용한 조정 팁"

4장, "하드웨어 크기 조정"

5장, "운영 체제 조정"

6장, "캐시 크기 조정"

7장, "색인화 조정"

8장, "로깅 조정"

9장, "기타 자원의 사용 관리"





## 유용한 조정 팁

성능 조정이란 특정 배포 요구 사항을 반영하여 기본 구성을 수정하는 것을 의미합니다.

본 설명서에서는 개별 **Directory Server** 인스턴스의 조정 방법에 대해 설명하고, 복제 토폴로지를 포함한 전체 디렉토리 서비스 설계가 완전하며 설계 요구 사항을 충족시키기 위해 여기에 소개된 정보를 사용하여 **Directory Server** 인스턴스를 조정한다고 가정합니다. 전체 디렉토리 서비스 설계가 완성되지 않은 경우에는 *Sun ONE Directory Server Deployment Guide*의 제안 사항을 참조하십시오.

성능을 조정하려면 표 3-1에 설명된 것처럼 시간과 노력, 창의적인 사고가 필요합니다.

**표 3-1**      조정 프로세스

단계	설명
<p>목표 정의</p>	<p>배포 요구 사항에 따라 조정에 대한 구체적이고 측정 가능한 목표를 정의합니다. 다음과 같은 질문을 가정해 보십시오.</p> <ul style="list-style-type: none"> <li>• 어떤 응용 프로그램에서 Directory Server를 사용하는가?</li> <li>• 시스템이 Directory Server 전용인가? 다른 응용 프로그램을 실행하는가? 그렇다면 어떤 응용 프로그램인가?</li> <li>• 배포에 필요한 항목수는 몇 개인가? 이 항목의 크기는 어느 정도인가?</li> <li>• Directory Server에서 지원해야 하는 초 당 검색수는 몇 개인가? 어떤 검색 유형이 예상되는가?</li> <li>• Directory Server에서 지원해야 하는 초 당 업데이트수는 몇 개인가? 어떤 업데이트 유형이 예상되는가?</li> <li>• 예상되는 업데이트 및 검색 최대 속도는 어느 정도인가? 예상되는 평균 속도는 어느 정도인가?</li> <li>• 배포 중에 이 시스템에서 대량 가져오기 초기화를 반복 수행해야 하는가? 그렇다면 가져오기 빈도는 어느 정도인가? 한 번에 가져오는 항목 수는 몇 개인가? 어떤 유형의 항목을 가져오는가? 서버가 실행되고 있을 때 온라인으로 초기화를 수행해야 하는가?</li> </ul> <p>이 목록은 완전하지 않습니다. 가능한 모든 상황을 고려하여 완벽한 목록을 작성하십시오.</p>
<p>방법 선택</p>	<p>조정 최적화를 구현하는 방법과 이를 측정하고 분석하는 방법을 결정합니다.</p> <p>시스템의 하드웨어 구성을 변경할 수 있는가? 기존 하드웨어 사용이 제한되어 기본 운영 체제와 Directory Server만 조정할 수 있는가? 어떻게 다른 응용 프로그램을 시뮬레이트할 수 있는가? 테스트를 위해 데이터 샘플을 생성하려면 어떻게 해야 하는가? 결과를 측정하려면 어떻게 해야 하는가? 결과를 분석하려면 어떻게 해야 하는가?</p>
<p>테스트 수행</p>	<p>계획대로 테스트를 수행합니다. 대규모의 복잡한 배포 시에는 이 단계에 상당한 시간이 소요됩니다.</p>
<p>결과 확인</p>	<p>테스트 후 최적화가 프로세스 초기 과정에서 정의한 목표를 달성했는지 확인합니다.</p> <p>목표를 달성했다면 결과를 문서화합니다.</p> <p>목표를 달성하지 못했다면 조정 중인 Directory Server의 프로필을 작성하고 모니터링합니다.</p>

**표 3-1** 조정 프로세스 (계속)

단계	설명
프로필 작성 및 모니터	원하는 설정을 변경한 후 Directory Server의 동작 프로필을 작성하고 모니터합니다. 관련된 모든 동작의 측정값을 수집합니다.
그래프 작성 및 분석	프로필을 작성하고 모니터하는 동안 관찰된 동작을 그래프로 작성하고 분석합니다. 추가 테스트의 필요성을 입증하는 증거와 패턴을 찾으려고 노력합니다.  추가 데이터를 수집하기 위해 프로필 작성 및 모니터 단계로 다시 돌아가야 할 수도 있습니다.
설정 조정	측정값 분석에서 제시된 최적화를 추가로 적용합니다.  테스트 수행 단계로 돌아갑니다.
결과 문서화	적용된 최적화가 프로세스 초기 단계에서 정의한 목표를 달성하면 나중에 쉽게 재현할 수 있도록 문서화합니다.

이 장에는 대부분의 Directory Server 인스턴스를 조정할 때 권장되는 기본 작업이 수록되어 있습니다. 여기서 권장되는 작업은 일반적으로 타당하지만 특정 배포에 미치는 영향을 명확히 이해한 후에 적용해야 합니다. 이 장은 모든 상황에 적용되는 해결 방법이 아니라 일종의 검사 목록으로 사용하십시오.

**1. 캐시 크기를 조정합니다.**

서버의 물리적 메모리가 Directory Server에서 사용한 모든 캐시를 저장할 수 있다면 가장 바람직할 것입니다. 이 경우 디렉토리의 모든 항목을 저장할 수 있도록 항목 캐시 크기를 여유 있게 설정하고 모든 색인을 저장할 수 있도록 데이터베이스 캐시 크기를 설정합니다.

자세한 내용은 6장, "캐시 크기 조정"을 참조하십시오.

**2. 색인을 최적화합니다.**

- a. 불필요한 색인을 제거하고 예상되는 요청을 지원하는 색인을 추가합니다.

때로는 새 응용 프로그램의 요청을 지원하는 색인을 추가해야 할 수도 있습니다. Directory Server를 실행하는 동안 색인을 추가, 제거 및 수정할 수 있습니다. 단, 기존 데이터는 해당 시점부터 색인화됩니다.

자세한 내용은 126페이지의 "이점: 검색 시 색인의 역할" 및 127페이지의 "손실: 업데이트 시 색인에 미치는 영향"을 참조하십시오.

- b. 색인화된 검색만 허용합니다.

색인화되지 않은 검색을 허용하면 서버 성능이 크게 저하되고 상당한 서버 자원이 소모될 수 있습니다. 응용 프로그램이 수행할 수 있는 특정 검색을 지원하는 색인을 추가하고 서버에서 색인화되지 않은 검색을 거부하도록 설정합니다.

자세한 내용은 134페이지의 "색인화된 검색만 허용"을 참조하십시오.

- c. 색인 목록의 최대 길이를 조정합니다.

자세한 내용은 135페이지의 "색인 목록 길이 제한"을 참조하십시오.

### 3. 기본 운영 체제를 조정합니다.

자세한 내용은 5장, "운영 체제 조정"을 참조하십시오.

### 4. 작업 제한을 조정합니다.

작업 제한을 조정하면 Directory Server가 한 개의 작업에 과도한 자원을 소모하지 않도록 방지할 수 있습니다. 기능 향상이 필요한 클라이언트 응용 프로그램에 고유 바인드 DN을 할당한 다음 이 고유 바인드 DN에 자원 제한을 설정합니다.

자세한 내용은 9장, "기타 자원의 사용 관리"를 참조하십시오.

### 5. 불필요한 로깅을 사용하지 않습니다.

디스크 액세스는 메모리 액세스보다 훨씬 속도가 느리기 때문에 디스크의 로그 파일에 자주 쓰면 성능이 크게 저하될 수 있습니다. 필요하지 않을 때는 액세스, 오류 및 감사 로깅을 해제하여 디스크 쓰기를 차단하십시오. 최소한 서로 다른 컨트롤러를 사용하는 별도의 디스크에 로그 파일을 저장하여 로깅 영향을 줄여야 합니다.

자세한 내용은 8장, "로깅 조정"을 참조하십시오.

**6. 디스크 작업을 분산합니다.**

특히 다수의 업데이트를 지원하는 배포 시에는 **Directory Server**가 디스크 입출력을 과도하게 소모할 수 있습니다. 가능하면 서로 다른 컨트롤러를 사용하는 여러 디스크에 작업 로드를 분산합니다.

자세한 내용은 85페이지의 "디스크 하위 시스템 크기 조정"을 참조하십시오.



## 하드웨어 크기 조정

적절한 하드웨어 크기 조정은 디렉토리 서비스 계획 및 배포의 중요한 구성 요소입니다. 하드웨어 크기 조정에서 가장 중요한 요인은 사용할 수 있는 메모리 양과 로컬 디스크 공간 크기입니다.

---

**주**           최상의 결과를 내려면 생산 환경에서 사용되는 항목의 부분 집합으로 테스트 시스템을 설치하고 구성합니다. 이 테스트 시스템을 사용하여 생산 서버의 동작을 추정할 수 있습니다.

특정 시스템에 대해 최적화하는 경우 시스템 버스, 주변 장치 버스, 입출력 장치 및 지원되는 파일 시스템의 작동 방식을 알아야만 **Directory Server**를 지원하도록 조정할 때 입출력 하위 시스템 기능을 이용할 수 있습니다.

---

이 장에서는 **Directory Server** 인스턴스의 디스크 및 메모리 요구 사항을 추정하는 방법에 대해 설명하며 네트워크 및 SSL 액셀러레이터 하드웨어 요구 사항에 대해서도 간략하게 다룹니다.

### 최소 요구 사항

표 4-1에서는 생산 환경에서 소프트웨어를 설치 및 사용하는 데 필요한 최소 메모리 및 디스크 공간 요구 사항을 보여줍니다.

지정된 항목 수에 대한 실제 최소 요구 사항은 표 4-1과 다를 수도 있습니다. 여기에 제시된 요구 사항은 기본 구성에 따라 색인이 설정되어 있고 최소 수준으로 캐시가 조정된 비교적 소수의 항목을 기준으로 상정한 것입니다. 항목에 디지털 사진과 같은 큰 이진 속성 값이 포함되어 있거나 색인화 또는 캐시를 다른 방식으로 구성할 경우 최소 디스크 공간 및 메모리 추정 값도에 따라 상향 조정하십시오.

**표 4-1** 최소 디스크 공간 및 메모리 요구 사항

작업	사용 가능한 로컬 디스크 공간	사용 가능한 RAM
제품 압축 풀기	최소 125MB	-
제품 설치	최소 200MB	최소 256MB
10,000-250,000개 항목	최소 3GB 추가	최소 256MB 추가
250,000-1,000,000개 항목	최소 5GB 추가	최소 512MB 추가
1,000,000개 항목 이상	8GB 이상 추가	1GB 이상 추가

최소 디스크 공간 요구 사항에는 액세스 로그에 할당된 1GB가 포함되어 있습니다. 기본적으로 Directory Server는 각각 최대 100MB(cn=config의 nsslapd-accesslog-maxlogsize)의 메시지를 저장하는 액세스 로그 파일 10개(cn=config의 nsslapd-accesslog-maxlogspersdir)를 순환적으로 사용하도록 구성되어 있습니다. 오류 로그와 감사 로그 용량은 Directory Server 구성 방법에 따라 달라집니다. 로깅 구성에 대한 자세한 내용은 *Sun ONE Directory Server 관리 설명서*를 참조 하십시오.

## 사용 가능한 최소 메모리

최소 메모리 추정 값은 일반 배포 시 Directory Server 인스턴스에서 사용하는 메모리를 반영합니다. 시스템 및 다른 응용 프로그램에서 사용하는 메모리는 이 추정 값에 포함되어 있지 않습니다. 보다 정확하게 산정하기 위해서는 실제 메모리 사용량을 측정해야 합니다. 자세한 내용은 81페이지의 "물리적 메모리 크기 조정"을 참조하십시오.

일반적으로 사용 가능한 메모리 양이 많을수록 더 바람직합니다.

## 최소 로컬 디스크 공간

최소 로컬 디스크 공간 추정 값은 일반 배포 시 Directory Server 인스턴스에 필요한 공간을 반영합니다. 경험에 따르면 디렉토리 항목이 클 경우 디스크에 있는 해당 LDIF 크기의 최소 네 배에 해당하는 공간이 필요합니다. 자세한 내용은 85페이지의 "디스크 하위 시스템 크기 조정"을 참조하십시오.

네트워크 디스크에는 서버 또는 이 서버에서 액세스하는 데이터를 설치하지 *마십시오*. Sun ONE Directory Server 소프트웨어는 NFS, AFS 또는 SMB를 통해 네트워크에 연결된 저장소의 사용을 지원하지 않습니다. 설치 후에도 구성, 로그, 데이터베이스 및 색인 파일은 모두 로컬 저장소에 있어야 합니다.



Windows 시스템을 사용하는 경우 드라이브를 FAT 대신 NTFS로 포맷하십시오. FAT는 Directory Server에서 지원되지 않습니다. NTFS를 사용하면 파일과 디렉토리에 대해 액세스 제어를 설정할 수 있습니다.

## 최소 처리 능력

일반적으로 대규모 시스템은 여러 개의 고속 프로세서를 사용하여 동시 다중 검색, 광범위한 색인화, 복제 및 기타 기능에 필요한 처리 능력을 제공합니다. 자세한 내용은 94페이지의 "다중 프로세서 시스템 크기 조정"을 참조하십시오.

## 최소 네트워크 용량

테스트에 따르면 예상되는 최대 처리량에 따라 100Mbit 이더넷으로 서비스 공급자 성능도 제공할 수 있습니다. 아래 수식에 의해 이론적인 최대 처리량을 추정할 수 있습니다.

최대 처리량 = 반환된 최대 항목 수/초 x 평균 항목 크기

예를 들어, Directory Server에서 초 당 최대 5000개의 검색에 응답해야 하며 각 검색에 대해 1개의 항목을 반환하고 항목의 평균 크기가 2000바이트이면 이론적인 최대 처리량은 10MB 또는 80Mbit가 됩니다. 80Mbit는 100Mbit 이더넷 어댑터 한 개가 제공할 수 있는 처리량보다 많습니다. 실제 성능은 이와 다를 수 있습니다.

WAN상에서 다중 마스터 복제를 수행하는 경우 네트워크 연결이 최소 대기 시간과 제로에 가까운 패킷 손실을 보장하면서 충분한 처리량을 제공하는지 확인합니다.

자세한 내용은 94페이지의 "네트워크 용량 크기 조정"을 참조하십시오.

## 물리적 메모리 크기 조정

Directory Server는 데이터베이스 기술을 사용하여 정보를 저장합니다. 데이터베이스 기술을 이용하는 모든 응용 프로그램의 경우와 마찬가지로 Directory Server 성능을 최적화하려면 빠른 메모리가 필요합니다. 일반적으로 사용 가능한 메모리 양이 많을수록 더 많은 디렉토리 정보를 캐시에 저장하여 신속하게 액세스할 수 있습니다. 항상 전체 디렉토리 내용을 캐시하기에 충분한 메모리가 각 서버에 있다면 가장 바람직할 것입니다. Sun ONE Directory Server 5.2는 64비트 메모리 주소 지정을 지원하기 때문에 캐시 크기가 몇 GB로 제한되지 않으며, 이론적으로 64비트 구조에서 총 1.5테라바이트 이상의 캐시 크기를 처리할 수 있습니다.

---

**주** 생산 환경에 Directory Server를 배포하는 경우 캐시 크기를 이론적인 프로세스 제한보다 훨씬 낮게 구성하여 일반 시스템 작동에 필요한 자원을 남겨 두십시오.

---

Directory Server 실행에 필요한 메모리 크기를 계산하려면 특정 Directory Server 구성에 필요한 메모리와 Directory Server를 실행하는 기본 시스템에 필요한 메모리를 추정해야 합니다.

## Directory Server 메모리 크기 조정

특정 배포에 예상되는 구성 값이 있으면 Directory Server 인스턴스에 필요한 물리적 메모리를 추정할 수 있습니다. 표 4-2에는 이 절에 설명된 계산에서 사용하는 값이 요약되어 있습니다.

**표 4-2** Directory Server 메모리 크기 조정 값

값	설명 <sup>1</sup>
nsslapd-cachememsize	접미사의 항목 캐시 크기  항목 캐시에는 클라이언트 요청에 응답하여 바로 보낼 수 있는, 서식이 지정된 항목이 포함되어 있습니다. 한 개의 인스턴스에서 여러 개의 항목 캐시를 처리할 수 있습니다.
nsslapd-dbcachesize	데이터베이스 캐시 크기  데이터베이스 캐시에는 데이터베이스 요소와 서버에서 사용하는 색인이 저장됩니다.
nsslapd-import-cachesize	대량 가져오기용 데이터베이스 캐시 크기  가져오기 캐시는 항목을 가져올 때만 사용됩니다. 오프라인 가져오기 만수행하는 경우 가져오기 캐시에 추가 메모리를 할당하는 대신 항목 캐시나 데이터베이스 캐시에 할당된 메모리를 다시 사용할 수 있습니다.
nsslapd-maxconnections	관리되는 최대 연결 수
nsslapd-threadnumber	서버를 시작할 때 작성된 작업 스레드 수

1. 자세한 설명은 *Sun ONE Directory Server Reference Manual*을 참조하십시오.

대략적인 메모리 크기를 추정하려면 다음 단계를 수행합니다.

1. 서버 프로세스 `slapdBase`의 기본 크기를 추정합니다.

```
slapdBase = 75MB +(nsslapped-threadnumber x 0.5MB) +(nsslapped-maxconnections x 0.5KB)
```

2. 항목 캐시 크기의 합계인 `entryCacheSum`을 결정합니다.

```
entryCacheSum = Sum all entry caches(nsslapped-cachememsize)
```

3. 모든 캐시의 전체 크기인 `cacheSum`을 결정합니다.

```
cacheSum = entryCacheSum + nsslapped-dbcachesize + nsslapped-import-cachesize
```

4. Directory Server 프로세스의 전체 크기인 `cacheSum`을 결정합니다.

```
slapdSize = slapdBase + cacheSum
```

Solaris 시스템에서 `pmap(1)`와 같은 유틸리티를 사용하거나 Windows 작업 관리자 사용하여 Directory Server에서 사용한 물리적 메모리를 측정할 수 있습니다.

5. 들어오는 클라이언트 요청인 `slapdGrowth`를 처리하는 데 필요한 메모리를 추정합니다.

```
slapdGrowth = 20% x slapdSize
```

첫 번째 추정 값에서는 클라이언트 요청을 처리하는 데 20%의 오버헤드를 가정합니다. 실제 백분율은 특정 배포의 특징에 따라 달라질 수 있습니다. Directory Server를 생산 환경에 설치하기 전에 실제 테스트를 통해 이 백분율을 확인하십시오.

6. Directory Server의 전체 메모리 크기인 `slapdTotal`을 결정합니다.

```
slapdTotal = slapdSize + slapdGrowth
```

32비트 서버가 포함된 대규모 배포의 경우 `slapdTotal`이 실제 제한인 3.4GB는 물론 약 3.7GB의 이론적인 프로세스 제한까지 초과할 수 있습니다. 이 경우 6장, "캐시 크기 조정"에 설명된 것처럼 시스템 제한 내에서 작동하도록 캐시를 조정하거나 64비트 버전의 제품을 사용하도록 선택할 수 있습니다.

## 운영 체제 메모리 크기 조정

세부적인 시스템 구성에 따라 운영 체제 메모리 요구 사항이 크게 달라질 수 있으므로 기본 운영 체제를 실행하는 데 필요한 메모리는 실제 테스트를 통해 추정해야 합니다. 따라서 기본 운영 체제에 필요한 메모리를 추정하기 전에 5장, "운영 체제 조정"에 설명된 것처럼 배포용 샘플 시스템을 조정해 보십시오. 시스템을 조정 후 메모리 사용을 모니터링하여 초기 추정 값 `systemBase`에 도달하는지 확인합니다. Solaris 시스템에서 `sar(1)`과 같은 유틸리티를 사용하거나 Windows 작업 관리자 사용하여 메모리 사용을 측정할 수 있습니다.

- 
- 주**           최상의 성능을 내려면 Directory Server 서비스 전용으로 시스템을 실행하십시오.
- 다른 응용 프로그램이나 서비스를 실행해야 한다면 필요한 총 메모리 크기를 조정할 때 이들 응용 프로그램이나 서비스에서 사용하는 메모리도 모니터해야 합니다.
- 

일반적인 시스템 오버헤드와 정상적인 관리 용도를 위한 메모리도 할당하십시오. 첫 번째 추정 값인 `systemOverhead`는 수백 메가바이트 이상 또는 전체 물리적 메모리의 10% 중에서 더 큰 값이어야 합니다. `systemOverhead`에 충분한 공간을 할당하면 시스템이 작업 중에 페이지를 메모리로 스왑인 및 스왑아웃하는 것을 방지할 수 있습니다.

이렇게 해서 운영 체제에 필요한 총 메모리인 `systemTotal`은 다음과 같이 추정할 수 있습니다.

```
systemTotal = systemBase + systemOverhead
```

## 총 메모리 크기 조정

이전 절에서 계산된 `slapdTtotal` 및 `systemTotal` 추정 값을 사용하여 필요한 총 메모리인 `totalRAM`을 추정합니다.

```
totalRAM = slapdTtotal + systemTotal
```

`totalRAM`은 필요한 총 메모리의 추정 값입니다. 단, 시스템이 Directory Server 프로세스 전용이라고 가정하며 시스템에서 실행하는 다른 모든 응용 프로그램과 서비스의 예상 메모리 사용량을 포함합니다.

## 메모리가 부족한 경우의 대처 방법

대부분의 경우 Directory Server에서 사용하는 모든 데이터를 캐시하기에 충분한 메모리를 설치하는 것은 비용 효과적이지 않습니다.

하지만 최소한 Directory Server 실행으로 인해 지속적인 페이지 스와핑이 발생하지 않을 정도의 메모리는 서버에 설치되어 있어야 합니다. 지속적인 페이지 스와핑은 서버 성능을 크게 저하시킵니다. Solaris 및 기타 시스템에서는 `vmstat(1M)`와 같은 유틸리티를 사용하여 Directory Server 시작 및 항목 캐시 초기화 전후의 메모리 통계를 볼 수 있습니다.

MemTool과 같이 Solaris 시스템에 지원되지는 않지만 별도로 사용할 수 있는 유틸리티는 테스트 시스템에서 응용 프로그램을 실행할 때 메모리가 어떻게 사용 및 할당되는지 모니터링하는데 유용할 수 있습니다.

지속적인 페이지 스와핑이 발생하는 상황에서 시스템에 추가 메모리를 설치할 수 없는 경우에는 데이터베이스 캐시와 항목 캐시의 크기를 줄이십시오. 스왑 공간이 부족하면 Directory Server가 중단될 수 있습니다.

모든 디렉토리 데이터를 캐시하기에 충분한 물리적 메모리를 제공할 수 없을 때의 대체 방안에 대해서는 6장, "캐시 크기 조정"을 참조하십시오.

## 디스크 하위 시스템 크기 조정

디스크 사용 및 입출력 기능은 성능에 큰 영향을 줄 수 있습니다. 특히 다수의 수정 작업을 지원하는 배포 시 디스크 하위 시스템은 입출력 병목이 될 수 있습니다. 이 절에서는 Directory Server 인스턴스의 전체 디스크 용량을 추정하고 디스크 입출력 병목을 줄이기 위해 권장되는 작업에 대해 설명합니다.

디스크 입출력 병목을 줄이는 방법은 8장, "로깅 조정"을 참조하십시오.

## 디렉토리 접미사 크기 조정

접미사의 디스크 공간 요구 사항은 디렉토리 항목의 크기와 개수는 물론 디렉토리 구성과 특히 접미사 색인화 방법에 따라 달라집니다. 대규모 배포에 필요한 디스크 공간을 측정하려면 다음 단계를 수행합니다.

1. 10,000개 항목, 100,000개 항목, 1,000,000개 항목에서 각각 한 개씩 선정하여 배포용으로 예상되는 세 가지 샘플 항목 집합의 LDIF를 생성합니다.  
생성된 항목은 예상되는 항목 유형의 혼합(사용자, 그룹, 역할, 확장된 스키마 항목)은 물론, 특히 userCertificate 및 jpegPhoto와 같은 단일 대규모 속성 값이 예상되는 경우 개별 속성 값의 평균 크기도 반영해야 합니다.
2. 배포에 예상되는 방식으로 Directory Server 인스턴스를 구성합니다.  
특히 생산 디렉토리 와 같은 방식으로 데이터베이스를 색인화합니다. 나중에 색인을 추가하려면 색인 공간도 추가해야 합니다.
3. 각 항목 집합을 로드하고 각 집합에 사용된 디스크 공간을 기록합니다.
4. 결과를 그래프로 작성하여 배포에 예상되는 접미사 크기를 추정합니다.

5. 오류 및 편차를 보완하기 위해 여분의 디스크 공간을 추가합니다.

접미사 디스크 공간과 더불어 Directory Server의 디스크 사용 방법도 고려해야 합니다.

## Directory Server의 디스크 사용 방법

디렉토리 접미사는 Directory Server에서 디스크에 저장하는 항목 중 하나입니다. 디스크 사용에 영향을 주는 기타 많은 요소는 배포 후 Directory Server의 사용 방법에 따라 크게 달라질 수 있으므로 여기서는 일반적인 경우에 대해서만 설명합니다. 여기에 설명된 항목 구성에 대한 자세한 내용은 *Sun ONE Directory Server 관리 설명서*를 참조하십시오.

### Directory Server 이진

이 버전의 Directory Server를 설치하려면 약 200MB의 디스크 공간이 필요합니다. 이 추정 값은 제품 이진에 필요한 공간이며 데이터 공간은 포함되지 않습니다.

### 이벤트 로깅

로그 파일의 디스크 사용 추정 값은 Directory Server 작업 속도, 로깅 유형 및 수준, 로그 순환 전략에 따라 달라집니다.

로깅 요구 사항은 대부분 미리 예측하여 계획할 수 있습니다. Directory Server에서 로그, 특히 감사 로그에 쓰면 디스크 사용량이 로드 수준만큼 증가합니다. 로드가 큰 배포에 광범위한 로깅이 필요하다면 높은 로드를 수용하기 위한 여분의 디스크 공간을 계획하십시오. 지능적 로그 순환과 보관 시스템을 구축하고, 자주 로그를 순환시키며, 이전 파일을 테이프나 값싼 디스크 클러스터와 같은 저렴한 대규모 저장 매체로 자동 이전함으로써 로드가 큰 로깅을 사용하는 배포의 디스크 공간 요구 사항을 줄일 수 있습니다.

로깅 요구 사항을 쉽게 예측할 수 없는 경우도 있습니다. 예를 들어, 디버그 로깅은 일시적이긴 하지만 errors 로그 크기를 급격하게 증가시킬 수 있습니다. 로드가 큰 대규모 배포의 경우 일시적인 대규모 디버그 로깅을 위해 몇 기가바이트의 전용 디스크 공간을 설정하는 것이 좋습니다. 자세한 내용은 8장, "로깅 조정"을 참조하십시오.

## 트랜잭션 로그

트랜잭션 로그 용량은 최대 쓰기 로드에 따라 달라지며, 쓰기 로드가 일정할 때보다 갑자기 증가할 때 더 많은 공간이 사용됩니다. Directory Server에서 정기적으로 트랜잭션 로그를 지우기 때문에 트랜잭션 로그는 계속 증가하지 않습니다. 하지만 온라인 백업 중에는 트랜잭션 로그가 플러시되지 않습니다.

Directory Server는 일반적으로 영구적 트랜잭션을 사용하여 실행됩니다. 영구적 트랜잭션 기능을 사용하는 Directory Server는 각 수정 작업(add, delete, modify, modrdn)을 동기적 방식으로 트랜잭션 로그에 씁니다. 이 경우 해당 디스크를 사용 중이면 작업이 차단될 수 있으므로 잠재적 입출력 병목이 발생합니다.

업데이트 성능이 중요하면 트랜잭션 로그 전용의 빠른 쓰기 캐시를 제공하는 디스크 하위 시스템을 사용합니다. 자세한 내용은 8장, "로깅 조정"을 참조하십시오.

## 복제 변경 로그 데이터베이스

배포 시 복제가 발생하는 경우 Directory Server 공급자는 변경 로깅을 수행합니다. 변경 로그 크기는 수정 용량과 사용된 변경 로그 지우기 유형에 따라 달라집니다. 변경 로그 지우기 방법에 따라 용량을 계획하십시오. 로드가 큰 대규모 배포의 경우 수정량이 비정상적으로 높은 기간 동안의 변경 로그 증가를 처리하기 위해 몇 기가바이트의 디스크 공간을 설정하는 것이 좋습니다. 자세한 내용은 8장, "로깅 조정"을 참조하십시오.

## 접미사 초기화 및 LDIF 파일

대량 로드 또는 가져오기라고도 하는 접미사 초기화 동안 Directory Server에는 접미사 데이터베이스 파일과 접미사 초기화에 사용된 LDIF를 저장할 디스크 공간 및 초기화 프로세스 중에 사용된 중간 파일을 저장할 공간이 필요합니다. 접미사 초기화 중에 사용된 LDIF 파일과 중간 파일을 저장할 데이터베이스 파일이 위치해 있는 디렉토리에 여분의 임시 용량을 계획하십시오.

## 백업 및 LDIF 파일

일반적으로 백업을 수행하면 다량의 디스크 공간이 소모됩니다. 백업 크기는 관련된 데이터베이스 파일의 크기와 같습니다. 데이터베이스 파일 용량의 몇 배에 해당하는 공간을 할당하여 여러 백업을 수용하고 데이터베이스와 해당 백업이 별도의 디스크에서 유지관리되게 합니다. 오래된 백업을 저렴한 저장 매체로 이전하는 지능형 전략을 사용하십시오.

초기화 LDIF 파일은 백업 LDIF 파일과 다르기 때문에 배포 시 복제가 발생하는 경우 초기화 LDIF 파일을 저장할 추가 공간을 계획합니다.

## 디스크 기반이 아닌 메모리 기반의 파일 시스템

일부 시스템은 메모리 기반의 `tmpfs` 파일 시스템을 지원합니다. 예를 들어, Solaris에서 `/tmp`는 성능을 향상시키기 위해 메모리 기반의 파일 시스템으로 마운트됩니다. 시스템의 다른 응용 프로그램과 공유하는 `/tmp`에 캐시 파일이 있을 경우 시스템의 `/tmp` 공간이 소모되지 않게 하십시오. 메모리가 부족하면 메모리 기반의 파일 시스템에 저장된 Directory Server 파일이 스왑 분할 영역에 할당된 디스크 공간으로 페이지징될 수 있습니다.

일부 시스템은 RAM 디스크 및 메모리 기반의 다른 대체 파일 시스템을 지원합니다. 메모리 기반의 파일 시스템 작성 및 관리에 대한 자세한 내용은 운영 체제 설명서를 참조하십시오. 이러한 파일 시스템에 저장된 모든 내용은 휘발성이며, 시스템 재부트 후에 다시 메모리에 로드해야 합니다.

## (UNIX 플랫폼) 코어 파일

최소한 `core` 파일 한두 개를 저장할 공간은 남겨 두십시오. Directory Server에서 코어를 덤프해서는 안 되지만 시스템 중단 중에 생성된 `core` 파일을 사용하여 검사할 수 있으면 시스템 중단 후의 복구 및 문제 해결이 훨씬 간편해집니다. 생성된 `core` 파일은 `cn=config`의 `nsslapd-errorlog`에서 지정된 파일과 동일한 디렉토리에 저장되거나, 시작 중에 시스템이 중단된 경우 `ServerRoot/bin/slapd/server/` 아래에 저장됩니다.

## 관리 공간

시스템 및 Directory Server 관리를 비롯한 예상되는 시스템 사용 공간을 남겨 두십시오. 기본 Directory Server 설치, 로컬 인스턴스에 있는 경우의 구성 접미사, 구성 파일 등에 충분한 공간이 할당되어 있는지 확인합니다.

## 디스크에 파일 분산

자주 업데이트되는 Directory Server 데이터베이스와 로그 파일을 별도의 디스크 하위 시스템에 저장하면 여러 개의 디스크 스핀들과 컨트롤러에 입출력 트래픽을 분산하여 입출력 병목을 방지할 수 있습니다. 다음 각 항목에 대해 전용 디스크 하위 시스템을 제공하는 것이 좋습니다.



## 트랜잭션 로그

영구적 트랜잭션 기능을 사용하는 Directory Server는 각 수정 작업을 동기적 방식으로 트랜잭션 로그에 씁니다. 따라서 해당 디스크를 사용 중이면 작업이 차단됩니다. 트랜잭션 로그를 전용 디스크에 저장하면 쓰기 성능이 향상되며 Directory Server에서 처리할 수 있는 수정량이 증가합니다.

자세한 내용은 147페이지의 "트랜잭션 로깅"을 참조하십시오.

## 데이터베이스

다중 데이터베이스 지원에서는 각 데이터베이스를 자체 물리적 디스크에 저장할 수 있으므로 자체 디스크 하위 시스템에 있는 여러 데이터베이스에 Directory Server 로드를 분산시킬 수 있습니다. 데이터베이스 작업을 위한 입출력 경쟁을 방지하려면 각 데이터베이스 파일 집합을 별도의 디스크 하위 시스템에 저장하는 것이 좋습니다.

최상의 성능을 내려면 큰 입출력 버퍼가 있는 빠른 전용 디스크 하위 시스템에 데이터베이스 파일을 저장하십시오. Directory Server는 캐시에서 후보 항목을 찾지 못하면 디스크에서 데이터를 읽고 정기적으로 쓰기를 플러시합니다. 빠른 전용 디스크 하위 시스템을 이러한 작업에 사용하면 잠재적인 입출력 병목을 줄일 수 있습니다.

`cn=config,cn=ldbm database,cn=plugins,cn=config`의 `nsslapd-directory` 속성은 Directory Server에서 색인 파일을 비롯한 데이터베이스 파일을 저장하는 디스크 위치를 지정합니다. 기본적으로 이러한 파일은 `ServerRoot/slapd-ServerID/db/`에 저장됩니다.

데이터베이스 위치를 변경하면 Directory Server를 다시 시작해야 할 뿐만 아니라 데이터베이스를 완전히 다시 작성해야 합니다. 생산 서버에서 데이터베이스 위치를 변경하는 것은 쉬운 작업이 아니므로 서버를 생산 환경에 설치하기 전에 가장 중요한 데이터베이스를 식별하여 별도의 디스크에 저장하십시오.

## 로그 파일

Directory Server는 버퍼 로깅 기능을 사용하여 액세스, 오류 및 감사 로그를 제공합니다. 버퍼링 기능이 있긴 하지만 로그 파일에 쓰려면 디스크에 액세스해야 하기 때문에 다른 입출력 작업과 경쟁하게 될 수도 있습니다. 성능, 용량 및 관리 향상을 위해 별도의 디스크에 로그 파일을 저장하는 것이 좋습니다.

자세한 내용은 8장, "로깅 조정"을 참조하십시오.

## 메모리 기반 파일 시스템의 캐시 파일

tmpfs 파일 시스템에서 파일은 물리적 메모리가 모두 사용되었을 경우에만 디스크로 스왑됩니다. 물리적 메모리에 모든 캐시 파일을 저장하기에 충분한 메모리가 있으면 Solaris 플랫폼의 tmpfs 파일 시스템이나 다른 플랫폼의 기타 메모리 기반의 파일 시스템(예: RAM 디스크)에 이와 동등한 디스크 공간을 할당한 다음, Directory Server에서 이 파일 시스템에 캐시 파일을 저장하도록 nsslapd-db-home-directory 값을 설정하여 성능을 향상시킬 수 있습니다. 이 경우 시스템이 메모리에 매핑된 캐시 파일을 불필요하게 디스크로 플러시하는 것을 방지할 수 있습니다.

## 디스크 하위 시스템 대체 방안

*"Fast, cheap, safe: pick any two." - Sun Performance and Tuning, Cockroft and Pettit.*

### 빠르고 안전한 대안

성능과 작동 시간이 모두 중요한 배포를 구현하는 경우, 대형 디스크 어레이에 분산된 고속 버퍼 입출력을 제공하기 위해 비휘발성 메모리 캐시가 있는 하드웨어 기반의 RAID 컨트롤러를 사용해 보십시오. 다수의 스피들에 로드를 분산시키고 고속 연결을 통한 버퍼링 액세스를 제공함으로써 입출력을 최적화할 수 있으며, 고성능 RAID 스트라이핑 또는 패리티 블록을 통해 우수한 안정성이 제공됩니다.

Sun StorEdge™ 제품에서 제공하는 것과 같은 대규모의 비휘발성 입출력 버퍼와 고성능 디스크 하위 시스템은 Directory Server 성능과 작동 시간을 크게 향상시킬 수 있습니다.

특히 트랜잭션 로그 전용으로 빠른 쓰기 캐시 카드를 사용하면 쓰기 성능이 향상됩니다. 빠른 쓰기 캐시 카드는 디스크 컨트롤러의 영향을 받지 않는 비휘발성 메모리 캐시를 제공합니다.

### 빠르고 저렴한 대안

빠르고 저렴한 성능을 원할 경우 다수의 디스크에 적절한 용량을 분산하십시오. 회전 속도가 빠르고 탐색 시간이 짧은 디스크를 사용하는 것이 좋습니다. 최상의 결과를 내려면 각 분산 구성 요소에 전용 디스크를 할당합니다. 단일 장애 지점을 피하기 위해 다중 마스터 복제를 사용하는 것이 좋습니다.

### 저렴하고 안전한 대안

저렴하고 안전한 구성을 원할 경우 Solaris Volume Manager와 같은 소프트웨어 기반의 저렴한 RAID 컨트롤러를 사용해 보십시오.

## RAID 대체 방안

RAID는 Redundant Array of Inexpensive Disks(저렴한 디스크의 중복 어레이)의 약어입니다. 이름에서 알 수 있듯이 RAID는 주로 복구 작업에 사용됩니다. 어레이에 구성된 특정 디스크에 장애가 발생해도 해당 디스크의 데이터는 손실되지 않고 이 어레이에 포함된 한 개 이상의 다른 디스크에서 계속 사용할 수 있습니다. 복구 기능을 구현하기 위해 RAID는 다중 디스크 드라이브를 볼륨이라는 보다 큰 가상 디스크로 구성할 수 있는 추상화를 제공합니다. 이 경우 물리적 디스크를 연결하거나 미러링 또는 스트라이핑해야 합니다. 연결을 구현하려면 논리적으로 특정 디스크 블록이 다른 디스크 블록 뒤에 오게 합니다. 예를 들어, 디스크 1에는 0-99 블록이 있고 디스크 2에는 100-199 블록이 있습니다. 미러링을 구현하려면 특정 디스크 블록을 다른 디스크에 복사한 다음 두 디스크가 계속해서 동기화되게 합니다. 스트라이핑은 여러 물리적 디스크에 가상 디스크 블록을 분산하는 알고리즘을 사용합니다.

스트라이핑은 성능 향상을 위해 사용됩니다. 디스크가 병렬로 작업할 수 있기 때문에 기록되는 데이터가 스트라이프 볼륨에 있는 두 개 이상의 디스크로 보내져서 매우 신속하게 임의 쓰기를 처리할 수 있습니다. 임의 읽기의 경우도 마찬가지입니다. 대규모의 순차적 읽기 및 쓰기에서는 큰 효과가 없을 수도 있지만 순차적 입출력 성능은 확실히 향상시킬 수 있습니다. 예를 들어, 한 개의 응용 프로그램이 다수의 입출력 요청을 생성하여 단일 디스크 컨트롤러를 무력하게 만들 수 있습니다. 하지만 스트라이프 볼륨 형식의 모든 디스크에 각각 전용 컨트롤러가 있으면 이러한 무력화를 방지할 수 있으므로 성능이 향상됩니다.

RAID는 소프트웨어나 하드웨어 RAID 관리자 장치를 사용하여 구현할 수 있습니다. 두 방법의 장단점은 다음과 같습니다.

- 일반적으로 하드웨어에 구현되는 하드웨어 RAID가 성능이 더 뛰어나기 때문에 소프트웨어 RAID보다 처리 오버헤드의 발생 가능성이 낮습니다. 또한 하드웨어 RAID는 호스트 시스템에서 분리되므로 호스트 자원으로 다른 응용 프로그램을 실행할 수 있습니다.
- 일반적으로 하드웨어 RAID가 소프트웨어 RAID보다 가격이 높습니다.
- 소프트웨어 RAID는 하드웨어 RAID보다 유연성이 뛰어날 수 있습니다. 예를 들어 하드웨어 RAID 관리자는 대체로 단일 디스크 어레이나 예정된 어레이 집합과 연결되는 반면, 소프트웨어 RAID는 수에 관계 없이 디스크 어레이를 캡슐화하거나 원할 경우 어레이 내의 특정 디스크만 캡슐화할 수 있습니다.

다음 절에서는 수준이라는 RAID 구성에 대해 설명합니다. 가장 일반적인 RAID 수준인 0, 1, 1+0, 5에 대해서는 어느 정도 자세히 다루고 있지만 일반적이지 않은 수준에 대해서는 간단한 비교 설명만 제공합니다.

### **RAID 0, 스트라이프 볼륨**

스트라이핑은 여러 물리적 디스크에 데이터를 분산합니다. 논리적 디스크인 볼륨은 청크나 스트라이프로 나뉘어진 후에 라운드 로빈 방식으로 물리적 디스크에 분산됩니다. 스트라이프는 항상 한 개 이상의 디스크 블록 크기이며 모든 스트라이프 크기가 같습니다.

RAID 0은 중복을 제공하지 않는다는 점에서 이름과 모순됩니다. RAID 0 스트라이프에서 디스크 장애가 발생하면 전체 논리적 볼륨이 손실됩니다. 하지만 RAID 0은 모든 디스크가 데이터 전용이기 때문에 RAID 수준 중에서 가장 저렴합니다.

### **RAID 1, 미러 볼륨**

미러링은 중복을 제공하기 위해 사용됩니다. 미러링되고 있는 디스크 중 하나에 장애가 발생해도 데이터를 계속 사용할 수 있으며 계속 처리됩니다. 한 가지 단점은 각 물리적 디스크가 미러링되므로 물리적 디스크 공간의 절반이 미러링에 사용된다는 것입니다.

### **RAID 1+0**

RAID 10이라고도 하는 RAID 1+0은 가장 높은 수준의 성능과 복구 기능을 제공하기 때문에 구현 비용이 가장 높은 RAID 수준입니다. 장애가 발생한 모든 디스크에서 다른 미러를 형성하는 경우 최대 세 개의 디스크에 장애가 발생해도 계속 데이터를 사용할 수 있습니다. RAID 1+0은 각 세그먼트가 RAID 1인 스트라이프 어레이로 구현됩니다.

### **RAID 0+1**

RAID 0+1은 RAID 1+0보다 복구 기능이 다소 떨어지며 스트라이프가 작성된 다음 미러됩니다. 미러의 같은 쪽에 있는 디스크는 한 개 이상에 장애가 발생해도 계속 데이터를 사용할 수 있지만 미러의 다른 쪽에 있는 디스크에 장애가 발생하면 논리적 볼륨이 손실됩니다. RAID 1+0과의 미세한 차이점은 같은 쪽에 있는 디스크는 동시에 장애가 발생해도 데이터를 계속 사용할 수 있다는 것입니다. RAID 0+1은 각 세그먼트가 RAID 0인 미러 어레이로 구현됩니다.

### **RAID 5**

RAID 5는 미러링만큼 복구 기능이 뛰어나지는 않지만 단일 디스크 장애 후에도 데이터를 사용할 수 있다는 점에서 중복을 제공합니다. RAID 5는 다른 디스크의 해당 스트라이프 바이트에 대해 배타적 논리합 OR(XOR)을 수행하여 작성된 패리티 스트라이프를 사용하여 중복을 구현합니다. 특정 디스크에 장애가 발생하면 이 디스크의 데이터는 나머지 디스크의 해당 스트라이프에 있는 데이터와 패리티를 사용하여 다시 계산됩니다. 하지만 이러한 수정 계산을 수행할 경우 성능이 저하됩니다.

RAID 5 볼륨은 각각의 논리적 쓰기에 대해 네 번의 물리적 입출력 작업을 해야 하기 때문에 정상적으로 작동하는 RAID 5는 대체로 RAID 0, 1+0 및 0+1보다 성능이 낮습니다. 이전 데이터와 패리티를 읽고 두 번의 배타적 논리합 OR(XOR) 작업을 수행한 다음 새 데이터와 패리티를 씁니다. 읽기 작업은 별로 영향을 받지 않기 때문에 동등한 수의 디스크를 사용하는 표준 스트라이프보다 약간 성능이 떨어질 뿐입니다. 즉, RAID 5 볼륨의 스트라이프에는 실제로 디스크 수가 한 개 부족하며 이 공간은 패리티에 사용됩니다. RAID 5는 보다 많은 여유 디스크 공간을 데이터에 할당하기 때문에 RAID 5 볼륨이 일반적으로 RAID 1+0 및 0+1보다 저렴합니다.

성능 문제를 감안할 때 데이터가 읽기 전용이거나 볼륨에 대한 쓰기가 거의 없는 경우가 아니면 RAID 5는 일반적으로 권장되지 않습니다. 하지만 디스크 어레이에 쓰기 캐시와 빠른 배타적 논리합 OR 엔진이 있으면 이러한 성능 문제가 완화되므로 RAID 5는 일부 배포의 미러링에 적합한 저렴한 대안이 될 수 있습니다.

### **RAID 수준 2, 3, 4**

RAID 수준 2와 3은 비디오 스트리밍과 같은 대규모의 순차적 데이터 전송에 적합합니다. 두 수준은 한 번에 한 개의 입출력 작업만 처리할 수 있으므로 임의 액세스가 필요한 응용 프로그램에는 바람직하지 않습니다. RAID 2는 해밍 오류 정정 코드(ECC)를 사용하여 구현됩니다. 즉, ECC 데이터를 저장하는 데 세 개의 물리적 디스크 드라이브가 필요하기 때문에 RAID 5보다는 비싸지만 스트라이프에 네 개 이상의 디스크가 있을 경우 RAID 1+0보다 저렴합니다. RAID 3은 비트 패리티 방법을 사용하여 중복을 구현합니다. 패리티는 RAID 5와 같이 분산되지 않고 단일 전용 디스크에 기록됩니다.

RAID 수준 2 또는 3과 달리 RAID 4는 여러 디스크 드라이브를 동시에 액세스하는 독립적인 액세스 방법을 사용합니다. 또한, 단일 디스크에 패리티를 쓰는 것을 제외하면 RAID 5와 유사한 방식으로 패리티를 사용합니다. 따라서 다중 쓰기를 효과적으로 일련화하여 쓰기를 수행할 때마다 패리티 디스크를 액세스하기 때문에 패리티 디스크가 병목이 될 수 있습니다.

### **소프트웨어 볼륨 관리자**

Solaris™ Volume Manager와 같은 볼륨 관리자도 Directory Server 디스크 관리에 사용할 수 있습니다. 생산 환경에 배포할 때 Solaris Volume Manager는 다른 소프트웨어 볼륨 관리자보다 훨씬 뛰어난 성능을 보입니다.

## 입출력 및 디스크 사용 모니터

정상 작동 시 디스크 공간이 불필요하게 소모되어서는 안 됩니다. Solaris 및 기타 시스템에서는 `iostat(1M)`와 같은 유틸리티를 사용하여 잠재적인 입출력 병목을 확인할 수 있습니다. Windows 시스템에서 입출력 병목을 처리하는 방법은 Windows 도움말을 참조하십시오.

## 다중 프로세서 시스템 크기 조정

Directory Server 소프트웨어는 다중 프로세서로 확장할 수 있도록 최적화되어 있습니다. 일반적으로 프로세서를 추가하면 전체적인 검색, 색인 유지 관리 및 복제 성능이 향상됩니다.

하지만 특정 디렉토리 배포 시에는 프로세서를 추가해도 성능에 큰 효과가 없는 한계효용 체감점에 도달할 수 있습니다. 검색, 색인화 및 응용 프로그램에 대한 매우 엄격한 성능 요구 사항을 처리해야 하는 경우 로드 균형 조정과 디렉토리 프록시 기술을 솔루션의 일부로 사용해 보십시오.

## 네트워크 용량 크기 조정

Directory Server는 네트워크 사용량이 많은 응용 프로그램입니다. Directory Server 인스턴스의 네트워크 가용성을 향상시키려면 시스템에 두 개 이상의 네트워크 인터페이스를 설치하십시오. Directory Server는 한 프로세스를 처리할 때 다중 네트워크 인터페이스에서 수신하여 이러한 하드웨어 구성을 지원할 수 있습니다.

로드 균형을 조정하기 위해 같은 네트워크에 디렉토리 서버 클러스터를 구현하려면 네트워크 인프라가 생성된 추가 로드를 지원할 수 있는지 확인합니다. WAN 환경에서 복제를 위한 고속 업데이트를 지원하려면 실제 테스트를 통해 네트워크 품질과 대역폭이 복제 처리량 요구 사항을 충족시키는지 확인합니다.

## SSL 크기 조정

기본적으로 SSL(Secure Sockets Layer) 프로토콜 지원은 소프트웨어에서 구현됩니다. 소프트웨어 기반의 SSL 구현을 사용하면 Directory Server 성능이 크게 저하될 수 있습니다. SSL 모드로 디렉토리를 실행하려면 전체적인 성능 요구 사항을 충족시키기 위해 여러 개의 디렉토리 복제본이 필요합니다.

하드웨어 액셀러레이터 카드도 SSL 사용으로 인한 영향을 완전히 제거할 수는 없지만 소프트웨어 기반의 구현에 비해 성능이 훨씬 향상됩니다. Sun ONE Directory Server 5.2는 지원되는 Sun Crypto Accelerator 하드웨어와 같은 SSL 하드웨어 액셀러레이터의 사용을 지원합니다.

Sun Crypto Accelerator 보드는 SSL 키 계산이 병목인 경우에 유용할 수 있습니다. 하지만 그 외의 경우에는 이러한 하드웨어를 사용해도 성능이 향상되지 않습니다. Sun Crypto Accelerator 보드는 특히 연결을 협상하는 SSL 핸드셰이크 중에 키 계산을 가속화하지만 이후의 메시지 암호화 및 암호 해독은 가속화하지 않습니다. Directory Server 인스턴스에서 이러한 하드웨어를 사용하는 방법은 부록 B, "Sun Crypto Accelerator 보드 사용"을 참조하십시오.





## 운영 체제 조정

기본 시스템과 네트워크 설정은 고성능 디렉토리 서비스에 적합하지 않습니다. **Directory Server**의 성능 최적화를 위해 시스템을 조정하려면 최소한 최신 권장 패치가 시스템에 설치되어 있어야 하며 기본 보안 조치를 강화하고 일부 시스템 및 네트워크 설정을 변경해야 합니다. 이 장에서는 이러한 조정 작업에 대해 설명합니다.

제품과 함께 제공된 `idsktune` 유틸리티(Solaris 패키지 버전의 `/usr/sbin/directoryserver idsktune`)를 사용하면 기본 시스템 구성의 단점을 쉽게 진단할 수 있습니다. 이 유틸리티는 고성능 디렉토리 서비스 지원에 권장되는 시스템 조정 작업을 소개할 뿐 실제로 시스템을 조정하지는 않습니다. 권장되는 조정 작업은 해당 시스템 관리자가 구현해야 합니다.

### 플랫폼 지원 확인

18페이지의 표 1-1에는 이번 릴리스에서 지원되는 플랫폼 및 관련된 하드웨어 구조가 명시되어 있습니다. 업데이트된 지원 플랫폼 목록은 제품 릴리스 노트를 참조하십시오.

Windows 시스템을 설치하는 경우, 네트워크 보안 서비스에 대한 의존도를 줄이기 위해 컴퓨터를 기존 도메인이나 작업 그룹의 구성원이 아니라 독립 실행형 서버로 지정하십시오.

# 시스템 패치 설치

전체적인 시스템 보안을 유지하고 Sun ONE Directory Server 5.2를 적절하게 설치하여 작동하려면 권장되는 최신 시스템 패치, 서비스 팩 또는 수정 프로그램을 설치합니다. 표 5-1에는 필요한 패치를 다운로드할 수 있는 URL이 나와 있습니다.

**표 5-1** 플랫폼별 패치 다운로드 사이트

플랫폼	다운로드 사이트
Sun Solaris™ Operating Environment	<a href="http://sunsolve.sun.com/">http://sunsolve.sun.com/</a>
Hewlett Packard HP-UX	<a href="http://www.hp.com/support/">http://www.hp.com/support/</a>
IBM AIX	<a href="http://www.ibm.com/support/">http://www.ibm.com/support/</a>
Microsoft Windows	<a href="http://support.microsoft.com/">http://support.microsoft.com/</a>
Red Hat Linux	<a href="http://www.redhat.com/">http://www.redhat.com/</a>

## 기본 보안 강화

이 절의 권장되는 모든 작업을 수행한다고 해서 위험이 완전히 제거되는 것은 아닙니다. 이러한 작업은 몇 가지 명백한 보안 위험을 줄일 수 있도록 도와주는 간략한 검사 목록일 뿐입니다.

### 시스템 격리

가능하면 네트워크 방화벽을 사용하여 Directory Server를 실행하는 시스템을 공용 인터넷과 격리하십시오. IP 기반 공격을 차단해야 하는 Windows 플랫폼에서 Directory Server를 실행하는 경우 시스템 격리가 특히 중요합니다.

### 이중 부트 금지

Directory Server를 실행하는 시스템에서는 이중 부트를 구성하거나 다른 운영 체제를 실행하지 마십시오. 다른 시스템이 제한된 파일에 대한 액세스를 허용할 수 있습니다.

## 강력한 암호

문장 부호나 알파벳 이외의 문자가 포함된 8자 이상의 수퍼유저 또는 관리자 암호를 사용합니다. Windows 플랫폼에서 Directory Server를 실행하는 경우 강력한 암호의 사용이 특히 중요합니다.

보다 긴 운영 체제 암호를 사용하려면 시스템의 암호 처리 방법을 새로 구성해야 할 수도 있습니다. 자세한 내용은 운영 체제 설명서를 참조하십시오.

## (Windows) 로컬 보안 정책

잘못된 로그인 시도가 있을 경우 사용자를 차단하는 로컬 보안 정책을 Windows 서버에 구현합니다. 배포에 적합한 크기의 로그를 관리하는 이벤트 로깅을 활성화하여 구성하고 로그인 시도에 대한 감사 로깅도 활성화합니다. 추측하기 어렵도록 관리자 계정의 이름을 바꾸는 것이 좋습니다.

자세한 내용은 Windows 도움말을 참조하십시오.

## (UNIX 플랫폼) 사용자 및 그룹

보안상 수퍼유저 권한을 사용하여 Directory Server 또는 관리 서버를 실행하는 것은 바람직하지 않습니다. 예를 들어, 로그인 권한이 없는 사용자 및 그룹을 작성한 다음 이 사용자 및 그룹으로 서버를 설치하여 실행할 수 있습니다. 로컬 파일에 사용자 및 그룹을 추가하는 경우 /etc/passwd 항목은 다음과 같이 표시될 수 있습니다.

```
server:x:61001:Server User:/dev/null:/dev/null
```

해당 /etc/group 항목은 다음과 같이 표시될 수 있습니다.

```
servers::61001:
```

디버깅하기 쉽도록 Solaris 시스템에서는 coreadm(1M)과 같은 유틸리티를 사용하여 이 사용자 및 그룹 ID로 실행되는 프로세스가 코어를 덤프할 수 있도록 선택할 수 있습니다.

특정 배포 시 메시징 서버와 같은 다른 서버와 Directory Server 파일을 공유해야 하는 경우 같은 사용자 및 그룹으로 해당 서버를 실행하십시오.

수퍼유저로 관리 서버를 실행해야 하면 서비스를 사용하지 않을 때는 중지하는 것이 좋습니다.

## 불필요한 서비스 비활성화

최상의 성능을 내고 위험을 줄이려면 시스템을 Directory Server 전용으로 사용합니다. 다른 서비스, 특히 네트워크 서비스를 실행하면 서버 성능과 확장성이 저하되고 보안 위험이 증가할 수 있습니다.

최대한 많은 네트워크 서비스를 비활성화하십시오. Directory Server는 TCP/IP만 사용하므로 파일 공유와 기타 서비스를 필요로 하지 않습니다. IP 라우팅, 우편, NetBIOS, NFS, RAS, 웹 게시, Windows 네트워크 클라이언트 서비스 등을 비활성화합니다. 특히 Windows에서는 이벤트 로그, 플러그 앤 플레이, 개인 데이터 보호, 보안 계정 관리자, Sun ONE 관리 서버, Sun ONE Directory Server, 원격 프로시저 호출(RPC), SNMP를 제외한 모든 서비스를 중지하고 비활성화하십시오. telnet과 ftp도 비활성화하는 것이 좋습니다.

대부분의 네트워크 서비스와 마찬가지로 telnet과 ftp도 보안상 위험을 초래할 수 있습니다. 두 서비스는 네트워크를 통해 암호화되지 않은 사용자 암호를 전송한다는 점에서 특히 위험합니다. telnet과 ftp 대신 보안 셸(ssh)이나 보안 FTP(sftp)와 같은 클라이언트를 사용할 수 있습니다.

Directory Server 인스턴스가 네트워크에 이름 지정 서비스를 제공하지 않으면 시스템의 이름 지정 서비스를 사용하십시오. Sun ONE 서버 콘솔과 같은 원격 관리 도구는 IP 주소와 호스트 이름 간의 변환 등 일부 작업을 수행할 때 이름 지정 서비스를 이용합니다.

네트워크 서비스 비활성화에 대한 자세한 내용은 운영 체제 설명서를 참조하십시오.

## 정확한 시간 유지

시스템 간에 로그 파일의 날짜 및 타임스탬프를 보다 쉽게 복제하고 상관 관계를 설정하려면 시스템 간에 시스템 클럭을 동기화해야 합니다. 예를 들어, 특히 Windows 시스템에서 정확한 시스템 시간을 설정하려면 NTP(Network Time Protocol) 클라이언트를 사용합니다.

## 시스템 장애 후 다시 시작

가능하면 Sun ONE Directory Server 관리 설명서에 설명된 것처럼 Directory Server를 중지하십시오. Directory Server가 올바르게 종료되지 않고 시스템 종료 중에 갑자기 중지되면 데이터 베이스가 손상되어 시작 속도가 느려질 수 있습니다. 이 경우 데이터베이스를 복구할 시간이 필요합니다.

(Solaris 패키지) 설치 및 구성 프로세스의 일부로 해당 스크립트가 부트 시 소프트웨어를 다시 시작합니다.

(Windows) 시스템 장애 후 자동으로 소프트웨어를 다시 시작하도록 Windows를 구성합니다. 자세한 내용은 Windows 도움말을 참조하십시오.

기타 플랫폼에서 부트 시 서비스를 시작하는 방법은 해당 운영 체제 설명서를 참조하십시오.

## 권장되는 기본 조정 작업 생성

Windows 이외의 플랫폼에서 권장되는 기본 조정 작업을 생성하려면 Solaris 패키지인 경우 `/usr/sbin/directoryserver idsktune`에 있는 `idsktune` 유틸리티, 기타 버전인 경우 제품 이진이 포함된 디렉토리에 있는 `idsktune` 유틸리티를 사용합니다.

수퍼유저로 이 유틸리티를 실행하면 유틸리티에서 시스템 정보를 수집하고 권장되는 수정 작업에 대해 알림, 경고 및 오류를 표시합니다. 예를 들어, 이 유틸리티는 다음과 같은 사항을 확인합니다.

- 운영 체제와 커널 버전이 이번 릴리스에서 지원되는지 여부
- 사용 가능한 메모리 및 디스크 공간이 일반 용도의 최소 요구 사항을 충족시키는지 여부
- 시스템 자원 제한이 일반 용도의 최소 요구 사항을 충족시키는지 여부
- 필요한 패치 또는 서비스 팩이 설치되어 있는지 여부

---

**주**                    생산용으로 사용할 시스템에 Directory Server를 설치하는 경우 먼저 모든 ERROR 조건을 수정해야 합니다.

---

특정 배포의 요구 사항은 최소 요구 사항을 초과할 수 있으므로 `idsktune` 유틸리티에서 최소 시스템 요구 사항으로 확인된 것보다 많은 자원을 제공할 수 있습니다.

유틸리티에 관한 자세한 내용은 Sun ONE Directory Server Resource Kit 설명서를 참조하십시오. Sun ONE Directory Server Resource Kit는 12페이지의 "Directory Server 도구 다운로드"에 설명된 방법으로 구할 수 있습니다.

# 시스템 설정 조정

현재 시스템 설정을 읽고 변경 사항을 권장하는 `idsktune` 도구를 사용할 수 있습니다. 일반적으로 권장되는 작업을 구현하면 **Directory Server**를 실행하는 전용 시스템과 추가 응용 프로그램을 실행하는 시스템의 성능이 모두 최적화됩니다.

권장되는 특정 작업을 구현하기 전에 로컬 네트워크 조건과 기타 응용 프로그램을 고려합니다. 추가 네트워크 조정 팁은 운영 체제 설명서를 참조하십시오.

**표 5-2** 배포 전에 확인해야 하는 구성 파일

플랫폼	파일	설명
Solaris Operating Environment	<code>/etc/init.d/inetinit</code>	조정을 위해 <code>ndd</code> 문을 추가합니다.
	<code>/etc/system</code>	시스템 제한을 확인합니다.
	<code>/etc/vfstab</code>	파일이 로컬에 있는지 확인합니다.
HP-UX	<code>/etc/rc.config.d/nddconf</code>	조정을 위해 <code>ndd</code> 문을 추가합니다. 혹은 <code>sam(1M)</code> 을 사용합니다.
Red Hat Linux	<code>/etc/fstab</code>	파일이 로컬에 있는지 확인합니다.
	<code>/etc/security/limits.conf</code>	<code>nofile</code> 하드 제한( <b>hard limit</b> ) 지시문을 추가합니다.
	<code>/etc/sysctl.conf</code>	커널 매개 변수를 확인 및 설정합니다.
	<code>/proc/sys/fs/file-max</code>	파일 설명자 제한을 확인합니다.

## (Windows) 지연된 프로시저 호출(DPC)

다중 프로세서 시스템에서 들어오는 네트워크 트래픽을 처리하기 위해 지연된 인터럽트 요청을 처리하는 Windows 기본 DPC는 성능을 저하시킬 수 있습니다. 실제로 DPC가 되는 지연된 인터럽트는 한 프로세서에서 다른 프로세서로 다시 예약되어 상당한 오버헤드를 발생시킵니다. 이러한 DPC 오버헤드 문제를 방지하려면 아래 레지스트리 키의 `ProcessorAffinityMask` 값을 0으로 설정하십시오.

`HKEY_LOCAL_MACHINE\System\CurrentControlSet\Services\NDIS\Parameters`

## 파일 설명자

Directory Server는 파일 설명자를 사용하여 동시 클라이언트 연결을 처리합니다. 따라서 시스템이나 프로세스에서 사용할 수 있는 최대 파일 설명자 수가 작으면 동시 연결 수가 제한될 수 있습니다. 파일 설명자 수에 대해 권장되는 작업은 시스템에서 Directory Server가 처리할 수 있는 동시 연결 수와 관련이 있습니다.

Solaris 시스템에서 사용할 수 있는 파일 설명자 수는 `/usr/sbin/directoryserver` `idsktune`(패키지 버전) 또는 `idsktune`(패키지 없음)의 출력에 설명된 것처럼 `rlim_fd_max` 매개 변수를 통해 구성됩니다. 사용할 수 있는 파일 설명자 수를 수정하는 방법은 운영 체제 설명서를 참조하십시오.

시스템에서 사용할 수 있는 최대 파일 설명자 수를 수정한 후 Directory Server에서 이 파일 설명자를 사용하도록 구성하는 방법은 152페이지의 표 9-2를 참조하십시오.

## (HP-UX) 큰 파일 지원

일부 HP-UX 시스템에서는 기본적으로 큰 파일은 지원하지 않습니다. Directory Server를 설치하려는 파일 시스템에서 큰 파일을 지원하도록 설정하는 방법은 HP-UX 제품 설명서를 참조하십시오. `idsktune` 유틸리티의 출력에도 지침이 나와 있습니다.

## (HP-UX) 스레드 보류 시간 제한

`idsktune` 유틸리티에 표시된 것처럼 일부 HP-UX 시스템에서 최대 스레드 보류 시간 제한 수가 잘못 설정되었을 수 있습니다. 최대 스레드 보류 시간 제한 수를 증가시키는 방법은 HP-UX 설명서를 참조하십시오. `idsktune` 유틸리티의 출력에도 지침이 나와 있습니다.

## (HP-UX) 프로세스 당 스레드

`idsktune` 유틸리티에 표시된 것처럼 일부 HP-UX 시스템에서 프로세스 당 최대 스레드 수가 너무 낮게 설정되었을 수 있습니다. 프로세스 당 최대 스레드 수를 증가시키는 방법은 HP-UX 설명서를 참조하십시오. `idsktune` 유틸리티의 출력에도 지침이 나와 있습니다.

## TCP(Transmission Control Protocol) 설정

특정 네트워크 설정은 플랫폼에 따라 달라집니다. 일부 시스템에서는 TCP 설정을 수정하여 Directory Server 성능을 향상시킬 수 있습니다. 이 절에서는 TCP 설정에 관해 idsktune 유틸리티가 권장하는 작업의 추론 과정을 설명합니다.

### TIME-WAIT 상태로 닫힌 연결

일부 시스템에서는 종료된 TCP 연결을 커널 테이블에 보관하는 기간을 구성할 수 있습니다. 보관되어 있는 연결은 신속하게 다시 열 수 있습니다. 이 기간을 너무 길게 설정하면 시스템이 오랜 기간 동안 커널 테이블에서 다수의 연결을 추적하므로 Directory Server에서 사용할 수 있는 연결 수가 줄어듭니다. 이 매개 변수 값을 30초(30,000밀리초)로 설정하면 대부분의 배포 시 Directory Server에 대해 더 많은 동시 연결을 제공할 수 있습니다.

Solaris 시스템에서 이 기간은 `/usr/sbin/directoryserver idsktune`(패키지 버전) 또는 `idsktune`(패키지 없음)의 출력에 설명된 것처럼 `tcp_time_wait_interval` 매개 변수를 통해 구성됩니다.

### 허용되는 보류 연결

일부 시스템에서는 Directory Server와 같은 TCP Listener에서 허용되는 TCP 보류 연결 수를 구성할 수 있습니다. 이 개수를 너무 작게 설정하면 Directory Server에서 허용할 수 있는 보류 연결 수가 제한됩니다. 이 매개 변수 값을 1024 이상으로 설정하면 대부분의 배포 시 Directory Server가 더 많은 동시 연결 요청을 처리할 수 있습니다.

Solaris 시스템에서 허용되는 보류 연결 수는 `/usr/sbin/directoryserver idsktune`(패키지 버전) 또는 `idsktune`(패키지 없음)의 출력에 설명된 것처럼 `tcp_conn_req_max_q` 매개 변수를 통해 구성됩니다. `tcp_conn_req_max_q0`을 2048로 늘리는 것이 좋습니다.

### 지연 승인

일부 시스템에서는 직접 시스템에 연결되어 있지 않은 호스트에 대해 TCP 승인이 지연되는 기간을 구성할 수 있습니다. 직접 지연 시간을 구성하는 대신 152페이지의 표 9-2에 설명된 것처럼 `cn=config`의 `nsslapd-nagle`를 `off`로 설정합니다.



## 비활성 연결

일부 시스템에서는 연결 유지 패킷의 전송 간격을 구성할 수 있습니다. 이 설정은 비활성화되어 잠재적으로 연결이 끊어진 TCP 연결을 유지하는 시간을 결정할 수 있습니다. 이 시간을 너무 길게 설정하면 연결 유지 간격으로 인해 시스템이 이미 연결이 끊어진 클라이언트의 연결을 유지하느라 불필요한 자원을 사용할 수 있습니다. 이 매개 변수 값을 600초(600,000밀리초, 10분)로 설정하면 대부분의 배포 시 Directory Server에 대해 더 많은 동시 연결을 제공할 수 있습니다.

Solaris 시스템에서 이 기간은 `/usr/sbin/directoryserver idsktune`(패키지 버전) 또는 `idsktune`(패키지 없음)의 출력에 설명된 것처럼 `tcp_keepalive_interval` 매개 변수를 통해 구성됩니다.

## 들어오는 연결

일부 시스템에서는 승인을 보내지 않고 들어오는 연결을 기다리는 시간을 구성할 수 있습니다. 이 기간을 너무 길게 설정하면 연결 실패 검색이 지연될 수 있습니다. 빠르고 믿을 수 있는 네트워크상의 인트라넷 배포 시 이 매개 변수 값을 600초(600,000밀리초, 10분)로 설정하면 성능이 향상됩니다.

Solaris 시스템에서 이 기간은 `/usr/sbin/directoryserver idsktune`(패키지 버전) 또는 `idsktune`(패키지 없음)의 출력에 설명된 것처럼 `tcp_ip_abort_interval` 매개 변수를 통해 구성됩니다.

## 나가는 연결

일부 시스템에서는 나가는 연결이 만들어질 때까지 기다리는 시간을 구성할 수 있습니다. 이 기간을 너무 길게 설정하면 나가는 연결을 만들려는 복제본 등의 대상 서버가 신속하게 응답하지 않을 경우 오랜 지연이 발생할 수 있습니다. 빠르고 믿을 수 있는 네트워크상의 인트라넷 배포 시 이 매개 변수 값을 10초로 설정하면 성능이 향상됩니다.

Solaris 시스템에서 이 기간은 `/usr/sbin/directoryserver idsktune`(패키지 버전) 또는 `idsktune`(패키지 없음)의 출력에 설명된 것처럼 `tcp_ip_abort_cinterval` 매개 변수를 통해 구성됩니다.

## 재전송 시간 제한

일부 시스템에서는 패킷의 초기 재전송 간격을 구성할 수 있습니다. 이 설정은 승인되지 않은 패킷을 재전송하기 전의 대기 시간에 영향을 줍니다. 이 시간을 너무 길게 설정하면 손실된 패킷에 대한 클라이언트 대기 시간이 길어질 수 있습니다. 빠르고 믿을 수 있는 네트워크상의 인터넷 배포 시 이 매개 변수 값을 500밀리초로 설정하면 성능이 향상됩니다.

Solaris 시스템에서 이 기간은 `/usr/sbin/directoryserver idsktune`(패키지 버전) 또는 `idsktune`(패키지 없음)의 출력에 설명된 것처럼 `tcp_rexmit_interval_initial` 매개 변수를 통해 구성됩니다.

Windows는 Van Jacobson TCP 고속 재전송 및 복구 알고리즘을 구현하여 재전송 타이머의 만료를 기다릴 필요 없이 ACK를 수신하면 즉시 누락된 세그먼트를 재전송할 수 있습니다. Van Jacobson 알고리즘을 구현하려면 아래 레지스트리 키를 수정합니다.

`HKEY_LOCAL_MACHINE/SYSTEM/CurrentControlSet/Services/Tcpip/Parameters`

`REG_DWORD` 종류의 `TcpMaxDupAcks`를 추가하고 값을 ACK 개수로 설정합니다. 값의 범위는 1-3이고 기본값은 2입니다.

## 시퀀스 번호

일부 시스템에서는 초기 시퀀스 번호를 생성하는 방법을 구성할 수 있습니다. 엑스트라넷 및 인터넷 배포 시에는 RFC 1948에 따라 초기 시퀀스 번호를 생성하여 시퀀스 번호 공격을 방지할 수 있도록 이 매개 변수를 설정합니다.

Solaris 시스템에서 이 동작은 `/usr/sbin/directoryserver idsktune`(패키지 버전) 또는 `idsktune`(패키지 없음)의 출력에 설명된 것처럼 `tcp_strong_iss` 매개 변수를 통해 구성됩니다.

## 캐시 크기 조정

Directory Server는 클라이언트 요청에 보다 신속하게 응답할 수 있도록 디렉토리 정보를 메모리와 디스크에 캐시합니다. 캐싱을 적절히 조절하면 클라이언트 요청을 처리할 때 디스크 하위 시스템에 대한 액세스를 최소화할 수 있습니다.

---

**주** 캐시가 제대로 조정 및 작동되지 않으면 다른 조정 작업도 성능에 큰 도움이 되지 않습니다.

---

## 캐시 종류

Directory Server는 표 6-1에 설명된 것처럼 세 가지 종류의 캐시를 처리합니다.

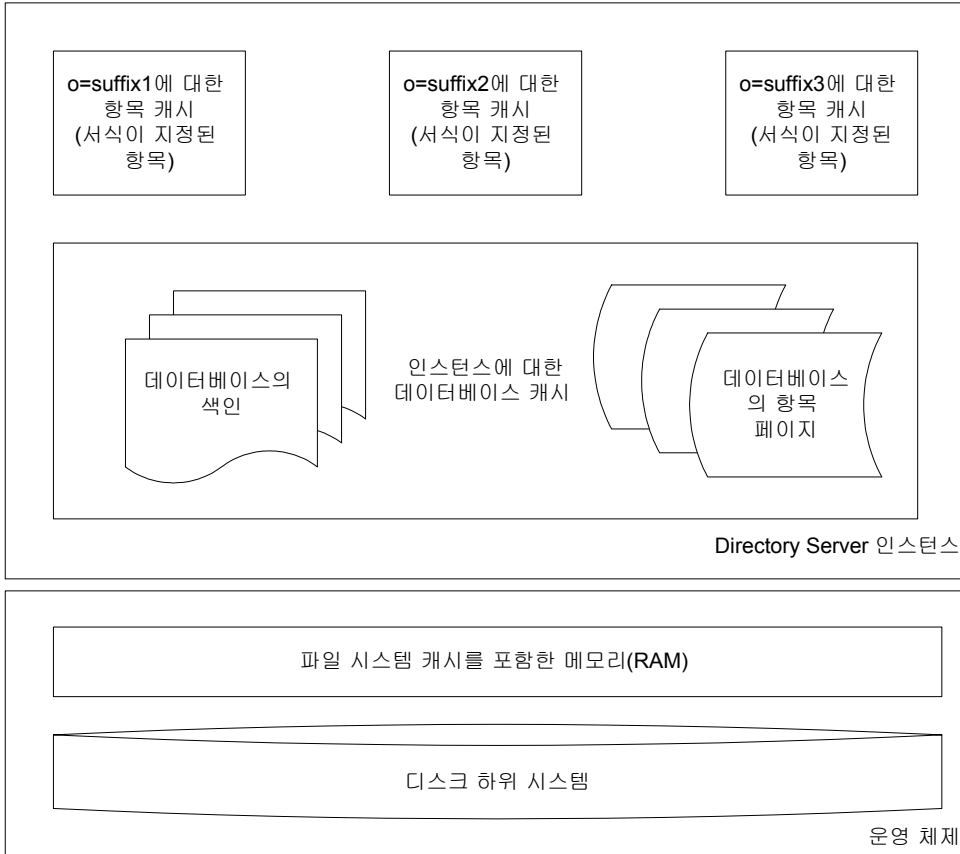
**표 6-1** 캐시

캐시 종류	설명
데이터베이스	각 Directory Server 인스턴스에는 색인과 항목을 데이터베이스 형식으로 저장하는 한 개의 데이터베이스 캐시가 있습니다.
항목	각 접미사에는 이전 작업 중에 데이터베이스에서 검색되어 클라이언트 응용 프로그램에 신속하게 전달할 수 있도록 서식이 지정된 항목을 저장하는 항목 캐시가 있습니다.
가져오기	각 Directory Server 인스턴스에는 데이터베이스 캐시와 구조적으로 유사하며 대량 로드 중에 사용되는 가져오기 캐시가 있습니다.

Directory Server는 기본 운영 체제에서 처리하는 파일 시스템과 디스크 하위 시스템의 입출력 버퍼도 사용합니다.

그림 6-1은 자체 항목 캐시가 있는 접미사 세 개를 처리하는 Directory Server 인스턴스의 캐시를 보여줍니다. 이 인스턴스는 8장, "로깅 조정"에 설명된 것처럼 트랜잭션 로그, 데이터베이스 및 기타 파일과 로그를 별도의 디스크 하위 시스템에 저장하여 중요한 디스크 작업을 처리하도록 구성되어 있습니다.

**그림 6-1** 특정 인스턴스의 항목 캐시와 데이터베이스 캐시



## 데이터베이스 캐시

각 Directory Server 인스턴스에는 한 개의 데이터베이스 캐시가 있습니다. 데이터베이스 캐시는 색인과 항목이 있는 데이터베이스의 페이지를 저장합니다. 각 페이지는 항목이 아니라 데이터베이스 일부가 포함된 메모리 조각입니다. 데이터베이스 캐시 크기 (nsslapped-dbcachesize)는 사용자가 지정합니다. 데이터베이스 캐시 공간은 서버를 시작할 때 할당되기 때문에 데이터베이스 캐시 크기의 변경 사항은 서버를 다시 시작한 후에 적용됩니다.

Directory Server는 최대 데이터베이스 캐시 크기를 유지하기 위해 데이터베이스 파일과 데이터베이스 캐시 간에 페이지를 이동합니다. 데이터베이스 캐시 자체를 관리하기 위한 메모리가 추가로 필요하므로 Directory Server에서 데이터베이스 캐시에 사용하는 실제 메모리 양은 지정한 크기보다 최대 25%까지 늘어날 수 있습니다.

대규모 데이터베이스 캐시를 사용하는 경우, Solaris 시스템에서는 pmap(1) 유틸리티와 같은 도구로 메모리 사용을 모니터하고 실제로 테스트함으로써 Directory Server의 메모리 사용량이 사용 가능한 물리적 메모리 크기를 초과하지 않도록 하십시오. 사용할 수 있는 물리적 메모리를 초과하면 시스템이 반복해서 페이지징을 시작하므로 성능이 크게 저하됩니다.

Directory Server에서 지원하는 UNIX 플랫폼의 ps(1) 유틸리티를 -p pid 및 -o format 옵션과 함께 사용하여 Directory Server와 같은 특정 프로세스에서 현재 사용하는 메모리를 볼 수도 있습니다(ns-slapd). Windows 시스템에서는 "작업 관리자"의 "프로세스" 탭 페이지에 프로세스 당 메모리 사용량이 표시됩니다(slapd.exe). 자세한 내용은 운영 체제 설명서를 참조하십시오.

실제로 32비트 서버에서는 데이터베이스 캐시 크기를 2GB 이하로 제한해야 합니다.

---

**주** Windows 및 AIX 플랫폼에서는 데이터베이스 캐시에 1GB(1,073,741,824 바이트) 이상 할당하지 마십시오.

---

nsslapd-dbcachesize 값의 올바른 범위에 대한 자세한 내용은 *Sun ONE Directory Server Reference Manual*을 참조하십시오.

## 항목 캐시

항목 캐시에는 클라이언트 응용 프로그램에 전달할 수 있도록 서식이 지정된 최근 액세스 항목이 저장됩니다. 접미사의 항목 캐시 크기(nsslapd-cachememsize)와 최대 항목 수(nsslapd-cachesize)는 사용자가 지정합니다. 항목 캐시는 필요에 따라 할당할 수 있습니다.

항목 캐시에 저장된 항목은 이미 서식이 지정되어 있으므로 Directory Server는 항목 캐시의 항목을 매우 효율적으로 반환할 수 있습니다. 데이터베이스 항목은 원시 바이트 문자열로 저장되며 클라이언트 응용 프로그램에 전달하기 전에 서식을 지정하고 항목 캐시에 저장해야 합니다.

항목 캐시 크기를 지정하는 경우 nsslapd-cachememsize는 Directory Server가 기본 메모리 할당 라이브러리에서 요청하는 메모리 양을 나타냅니다. 메모리 할당 라이브러리에서 이러한 요청을 처리하는 방법에 따라 실제 사용되는 메모리는 Directory Server가 항목 캐시에 최종적으로 사용할 수 있는 메모리 양보다 훨씬 클 수 있습니다.

Directory Server 프로세스에 사용되는 실제 메모리는 주로 사용된 메모리 할당 라이브러리와 캐시된 항목에 따라 결정됩니다. 작은 속성 값을 가진 다수의 항목이 큰 속성 값을 가진 소수의 항목보다 오버헤드가 큽니다.

실제로 32비트 서버에서는 항목 캐시 크기를 2GB 이하로 제한해야 합니다.

---

**주** AIX 플랫폼에서는 Directory Server가 `maxdata = 0x50000000`으로 작성되기 때문에 데이터베이스 캐시와 항목 캐시에 각각 1GB를 할당할 수 있습니다. `maxdata` 값을 변경해야 하면 Sun ONE 지원 담당자에게 문의하십시오.

---

`nsslapd-cachememsize` 및 `nsslapd-cachesize` 값의 올바른 범위에 대한 자세한 내용은 *Sun ONE Directory Server Reference Manual*을 참조하십시오.

## 가져오기 캐시

가져오기 캐시는 대량 로드 또는 가져오기라고도 하는 접미사 초기화 중에만 작성되어 사용됩니다. 배포 시 *offline* 접미사 초기화만 사용하는 경우 가져오기 캐시와 데이터베이스 캐시는 사용되지 않으므로 "총 캐시 크기"에 설명된 것처럼 총 캐시 크기를 계산할 때 이 두 항목을 추가할 필요가 없습니다. 가져오기 캐시 크기(`nsslapd-import-cachesize`)는 사용자가 지정합니다. 가져오기 캐시는 초기화에 할당된 다음 초기화 후에 해제되기 때문에 가져오기 캐시 크기를 변경하면 다음에 접미사를 재설정하여 초기화할 때 적용됩니다.

Directory Server는 데이터베이스 캐시와 같은 방식으로 가져오기 캐시를 처리합니다. 따라서 스와핑을 방지하려면 사용할 수 있는 물리적 메모리가 충분한지 확인해야 합니다.

실제로 32비트 서버에서는 가져오기 캐시 크기를 2GB 이하로 제한해야 합니다.

`nsslapd-import-cachesize` 값의 올바른 범위에 대한 자세한 내용은 *Sun ONE Directory Server Reference Manual*을 참조하십시오.

## 파일 시스템 캐시

운영 체제는 사용할 수 있는 메모리 중 Directory Server 캐시와 기타 응용 프로그램에서 사용하지 않는 메모리는 파일 시스템 캐시에 할당합니다. 이 캐시에는 디스크에서 최근에 읽은 데이터가 저장되므로 이후 요청은 디스크에서 다시 데이터를 읽을 필요 없이 캐시에서 복사된 데이터를 가져올 수 있습니다. 메모리 액세스가 디스크 액세스보다 훨씬 빠르기 때문에 파일 시스템 캐시에서 사용할 수 있는 물리적 메모리를 남겨 두면 성능이 향상됩니다.

파일 시스템 캐시에 대한 자세한 내용은 운영 체제 설명서를 참조하십시오.

## 총 캐시 크기

동시에 사용되는 모든 캐시의 합계는 사용할 수 있는 물리적 메모리의 전체 크기보다 파일 시스템 캐시에 할당된 메모리만큼 작아야 합니다. 따라서 실제로 32비트 서버에서는 총 캐시 크기를 2GB 이하로 제한해야 합니다. *사용되는 총 캐시는 지정된 크기보다 훨씬 늘어날 수 있습니다.* 캐시 크기 및 Directory Server 프로세스 크기가 사용 가능한 물리적 메모리를 초과하지 않도록 설정하는 방법은 108페이지의 "데이터베이스 캐시"를 참조하십시오.

---

**주** Windows 플랫폼에서 응용 프로그램이 사용할 수 있는 최대 주소 공간은 2GB입니다. 총 캐시 크기가 이 제한을 초과하면 Directory Server는 오류 메시지를 표시하고 종료됩니다.

---

Directory Server가 온라인 상태일 때 접미사를 초기화(대량 로드)하는 경우 데이터베이스 캐시, 항목 캐시 및 가져오기 캐시 크기의 합계는 사용 가능한 물리적 메모리의 전체 크기보다 작아야 합니다.

**표 6-2** 접미사 초기화(가져오기) 작업 및 캐시 사용 여부

캐시 종류	오프라인 가져오기	온라인 가져오기
데이터베이스	아니요	예
항목	예	예
가져오기	예	예

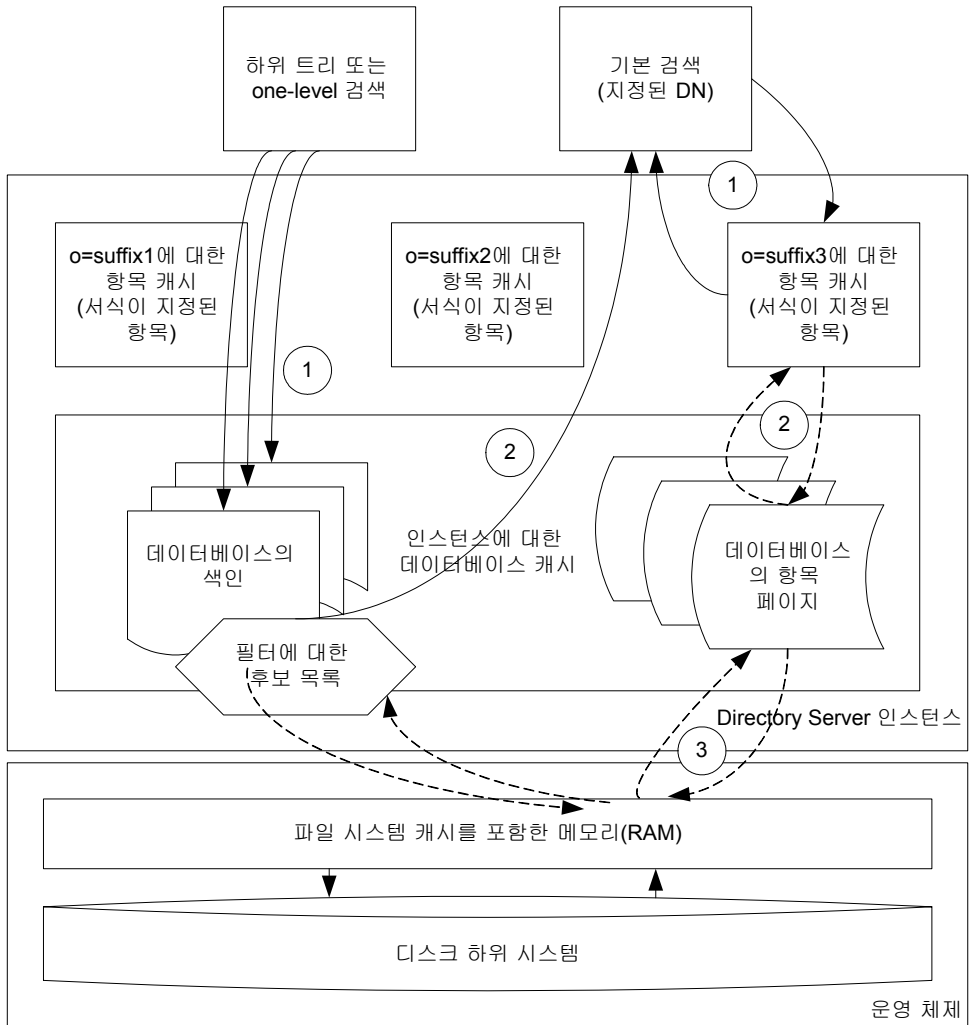
Directory Server가 중지된 상태에서 모든 접미사 초기화를 오프라인으로 수행하면 이러한 제한을 해결할 수 있습니다. 이 경우 가져오기 캐시와 데이터베이스 캐시가 함께 사용되지 않으므로 오프라인 접미사 초기화에서는 가져오기 캐시에, 온라인 사용에서는 데이터베이스 캐시에 같은 메모리를 할당할 수 있습니다. 하지만 이런 특수한 경우를 구현하려면 작업 시스템에서 다른 사용자가 온라인 대량 로드를 수행하지 않도록 해야 합니다. 이 경우에도 동시에 사용되는 캐시의 합계는 사용 가능한 물리적 메모리의 전체 크기보다 작아야 합니다.

## 검색 시 캐시 사용 방법

그림 6-2는 Directory Server에서 어떻게 기본 DN을 지정한 검색과 필터를 사용한 검색을 모두 처리하는지 보여줍니다. 각각의 선은 여러 수준의 메모리를 액세스하는 스레드를 나타내고 점선은 효과적인 조정을 통해 캐시 사용을 최소화하는 단계를 나타냅니다.



그림 6-2 검색 및 캐시



### 기본 검색 프로세스

그림에 표시된 것처럼 기본 검색(기본 DN을 지정한 검색)은 Directory Server에서 처리되는 가장 간단한 검색 유형입니다. 이 검색을 처리하기 위해 Directory Server는 다음과 같은 작업을 수행합니다.

1. 항목 캐시에서 지정된 기본 DN의 항목을 검색하려고 시도합니다.  
항목을 찾으면 Directory Server는 후보 항목이 검색에 제공된 필터와 일치하는지 확인합니다.  
항목이 일치하면 Directory Server는 서식이 지정된 캐시 항목을 신속하게 클라이언트 응용 프로그램으로 반환합니다.
2. 데이터베이스 캐시에서 항목을 검색하려고 시도합니다.  
항목을 찾으면 Directory Server는 이 항목을 접미사의 항목 캐시에 복사한 다음, 항목 캐시에서 찾은 것처럼 처리합니다.
3. 데이터베이스 자체에서 항목을 검색하려고 시도합니다.  
항목을 찾으면 Directory Server는 이 항목을 데이터베이스 캐시에 복사한 다음, 데이터베이스 캐시에서 찾은 것처럼 처리합니다.

## 하위 트리 및 one-level 검색 프로세스

역시 113페이지의 그림 6-2에 표시된 것처럼 하위 트리 또는 트리의 한 수준에서 검색하려면 항목 집합을 처리하는 추가 작업이 필요합니다. 이 검색을 처리하기 위해 Directory Server는 다음과 같은 작업을 수행합니다.

1. 데이터베이스 캐시의 색인을 사용하여 필터와 일치하는 후보 항목 집합을 구성하려고 시도합니다.  
적절한 색인이 없으면 데이터베이스 자체 항목에서 후보 항목 집합을 생성해야 합니다.
2. 다음과 같은 방법으로 각 후보 항목을 처리합니다.
  - a. 기본 검색을 수행하여 항목을 검색합니다.
  - b. 이 항목이 검색에 제공된 필터와 일치하는지 확인합니다.
  - c. 항목이 필터와 일치하면 클라이언트 응용 프로그램으로 반환합니다.

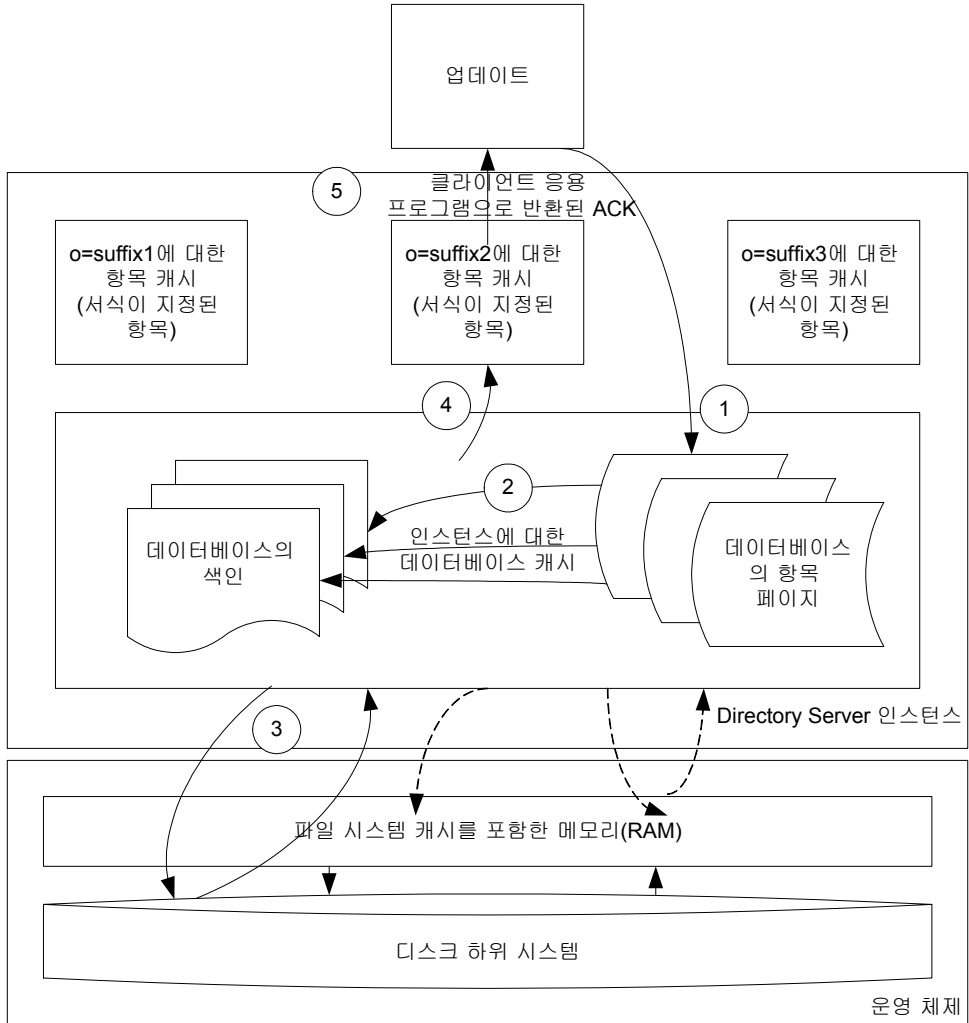
이렇게 하면 Directory Server에서 메모리에 후보 항목 집합을 구성할 필요가 없습니다.

Directory Server를 조정하기 전에 예상되는 검색 유형을 알고 있다면 가장 바람직하겠지만 실제 테스트를 통해 가정을 확인하는 것도 한 가지 대안이 될 수 있습니다.

# 업데이트 시 캐시 사용 방법

그림 6-3은 Directory Server에서 어떻게 업데이트를 처리하는지 보여줍니다. 각각의 선은 여러 수준의 메모리를 액세스하는 스레드를 나타내고 점선은 효과적인 조정을 통해 캐시 사용을 최소화하는 단계를 나타냅니다.

그림 6-3 업데이트 및 캐시



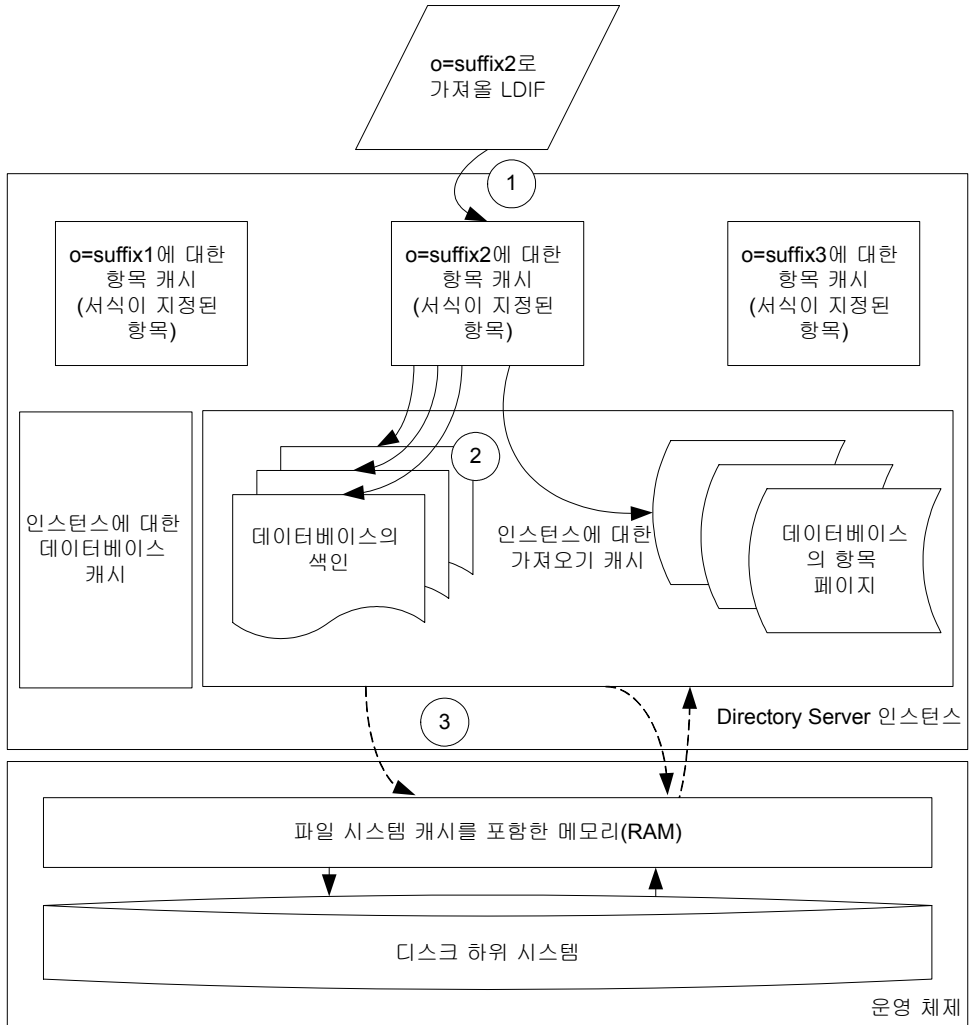
업데이트는 검색보다 많은 처리가 필요합니다. 업데이트를 처리하기 위해 Directory Server는 다음과 같은 작업을 수행합니다.

1. 기본 DN 검색을 수행하여 업데이트할 항목을 검색하거나 추가 작업인 경우 해당 항목이 확실히 없는지 확인합니다.
2. 데이터베이스 캐시를 변경하고 특히 업데이트로 영향을 받는 색인을 업데이트합니다.  
업데이트로 영향을 받는 데이터가 데이터베이스 캐시에 로드되어 있지 않으면 해당 데이터가 캐시에 로드되는 동안 이 단계에서 디스크 작업이 수행될 수 있습니다.
3. 변경 사항을 트랜잭션 로그에 쓰고 이 정보가 디스크로 플러시될 때까지 기다립니다.  
자세한 내용은 147페이지의 "트랜잭션 로깅"을 참조하십시오.
4. 업데이트된 항목의 서식을 지정하고 접미사의 항목 캐시에 복사합니다.
5. 업데이트 성공 확인 메시지를 클라이언트 응용 프로그램으로 반환합니다.

## 접미사 초기화 시 캐시 사용 방법

그림 6-4는 Directory Server에서 어떻게 접미사 초기화(대량 로드 가져오기)를 처리하는지 보여줍니다. 각각의 선은 여러 수준의 메모리를 액세스하는 스레드를 나타내고 점선은 효과적인 조정을 통해 캐시 사용을 최소화하는 단계를 나타냅니다.

그림 6-4 접미사 초기화(대량 로드) 및 캐시



접미사를 초기화하기 위해 Directory Server는 다음과 같은 작업을 수행합니다.

1. 버퍼로 사용되는 항목 캐시에 LDIF 항목을 공급하는 스레드를 시작합니다.
2. 영향을 받는 각 색인에 대한 스레드와 가져오기 캐시에 항목을 작성하는 스레드를 시작합니다. 이 스레드는 항목 캐시에 공급된 항목을 사용합니다.

3. 가져오기 캐시가 부족하면 데이터베이스 파일에서 읽고 씁니다.

Directory Server는 접미사 초기화 중에 로그 메시지를 쓸 수도 있지만 트랜잭션 로그에는 쓰지 않습니다.

Directory Server와 함께 제공되는 `ldif2db(/usr/sbin/directoryserver ldif2db)`와 같은 접미사 초기화 도구는 캐시 적중률과 가져오기 처리량에 관한 피드백을 제공합니다. 캐시 적중률과 가져오기 처리량이 모두 감소하면 가져오기 캐시가 너무 작은 것이므로 가져오기 캐시 크기를 증가시키는 것이 좋습니다.

## 검색 최적화

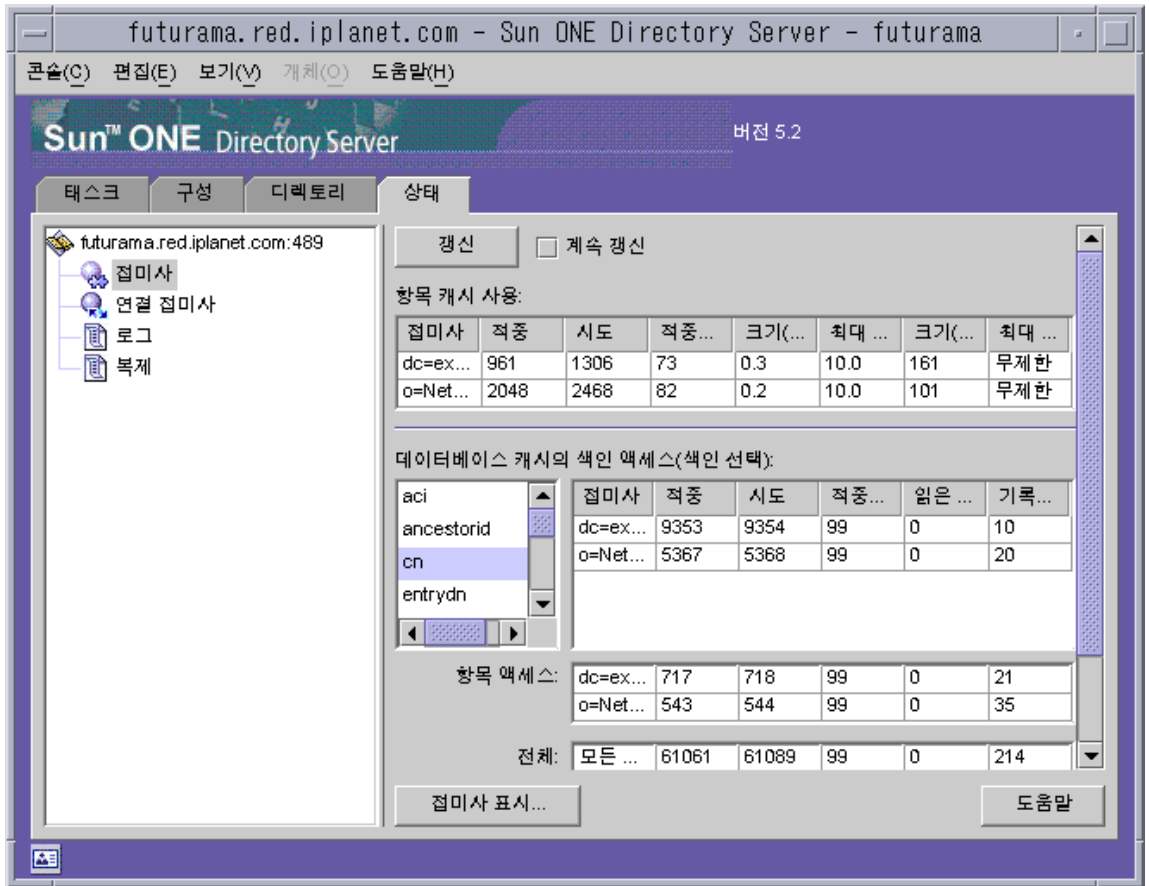
최상의 성능을 내려면 최대한 많은 디렉토리 데이터를 메모리에 캐시하십시오. 디렉토리가 디스크에서 정보를 읽지 않도록 차단하면 디스크 입출력 병목이 감소합니다. 디렉토리 트리 크기, 사용할 수 있는 메모리 양, 사용되는 하드웨어 등에 따라 여러 가지 방법으로 이 작업을 수행할 수 있습니다. 검색 성능을 최적화하기 위해 배포에 따라 항목 캐시와 데이터베이스 캐시에 할당하는 메모리 양을 늘리거나 줄일 수도 있습니다. 아니면 다른 서버의 Directory Server 소비자에 검색을 분산하도록 선택할 수 있습니다.

## 메모리에 모든 항목과 색인 저장

최적의 경우를 가정해 보십시오. 사용할 수 있는 물리적 메모리가 데이터베이스 캐시와 항목 캐시를 모두 저장하기에 충분합니다. 항목 캐시는 디렉토리에 있는 모든 항목을 저장할 수 있고 데이터베이스 캐시는 최소한 모든 색인을 저장할 수 있을 정도로 큼니다. 이 경우 캐시에서 모든 항목을 찾을 수 있으므로 Directory Server는 파일 시스템 캐시나 디스크에서 항목을 검색할 필요가 없습니다.

하지만 업데이트 후에도 데이터베이스 색인이 모두 데이터베이스 캐시에 저장될 수 있는지 확인해야 합니다. 데이터베이스 캐시에 색인을 저장할 공간이 부족하면 Directory Server는 검색 요청이 있을 때마다 디스크에서 색인을 읽어와야 하기 때문에 처리량이 크게 감소합니다. 그림 6-5와 같이 Directory Server 콘솔의 "상태" 탭에는 적중률과 기타 유용한 정보가 표시됩니다.

그림 6-5 Directory Server 콘솔에서 캐시 적중률 모니터



혹은 명령줄에서 아래 명령으로 검색을 실행하여 페이징 및 캐시 작업을 모니터할 수 있습니다.

```
$ ldapsearch -D admin -w password \
-b cn=monitor,cn=database_name,cn=ldbm database,cn=plugins,cn=config
```

데이터베이스 캐시에 .db3 파일의 데이터베이스 색인을 모두 저장하는 데 필요한 메모리 양을 대략적으로 추정하려면 아래 수식을 사용합니다. 사진과 같은 큰 이진 속성이 없는 일반 항목과 기본 색인 구성을 사용하는 경우 이 수식으로 상당히 정확한 결과를 얻을 수 있습니다.

$$nsslapped-dbcachesize = 1.2 \times \text{SUMall.db3 file (파일 크기)}$$

항목 캐시에 모든 항목을 저장하는 데 필요한 메모리 양과 항목 캐시의 슬롯 수를 대략적으로 추정하려면 아래 수식을 사용합니다. 사진과 같은 큰 이진 속성이 없는 일반 항목과 기본 색인 구성을 사용하는 경우 다시 이 수식으로 상당히 정확한 결과를 얻을 수 있습니다.

`nsslapd-cachesize = 4.5 x (LDIF의 항목 수)`

`nsslapd-cachememsize = 3.8 x (id2entry.db3 파일 크기)`

실제 테스트를 통해 추정 값을 확인하고 조정합니다. 특히 항목 캐시는 할당된 양보다 훨씬 많은 메모리를 사용할 수 있습니다.

## 충분한 메모리, 32비트 Directory Server

항목 캐시와 데이터베이스 캐시에 모든 데이터를 저장하기에 충분한 메모리가 있지만 64비트 Directory Server 프로세스를 지원하지 않는 시스템을 가정해 보십시오. 예를 들어, 하드웨어 제약 조건 때문에 Solaris 시스템에 배포할 수 없는 경우 32비트 프로세스의 메모리 제한을 고려하여 적절하게 캐시 크기를 지정하고 파일 시스템 캐시에 사용할 수 있는 메모리를 남겨 두는 것이 중요합니다.

---

**주**      파일 시스템 캐시는 시스템에서 실행되는 다른 프로세스, 특히 파일 기반의 작업과 공유됩니다. 따라서 Directory Server 전용이 아닌 시스템에서는 특히 다른 캐시보다 제어하기가 어렵습니다.

시스템에서 파일 시스템 캐시를 다른 프로세스에 재할당할 수도 있습니다.

---

## 메모리 축소, 일부 파일 시스템 캐시 사용

항목 캐시와 데이터베이스 캐시에 모든 데이터를 저장하기에는 불충분하지만 상당량의 가용 메모리가 있는 시스템을 가정해 보십시오. 이 경우 항목 캐시와 데이터베이스 캐시 크기의 합계가 사용할 수 있는 물리적 메모리를 초과함으로써 잦은 가상 메모리 페이지 발생하여 시스템이 거의 중단되는 상황이 발생하지 않도록 해야 합니다.

항목 캐시와 데이터베이스 캐시 크기를 500KB 정도로 낮게 설정하여 파일 시스템 캐시에서 사용할 수 있는 메모리를 남겨 두는 것이 좋습니다. 이렇게 하면 Directory Server는 디스크의 항목과 색인을 반복해서 읽을 필요 없이 파일 시스템 캐시에 저장된 데이터만으로 검색을 끝낼 수 있습니다.



혹은 어느 정도의 검색 패턴이 있을 경우 특정 배포의 검색이 대부분 디렉토리 항목 중 소규모의 동일 집합에 액세스하므로 이러한 검색용으로 항목과 색인을 캐시할 때 얻는 이점이 이례적인 검색 요청을 처리할 때의 손실을 상쇄한다는 가정 하에 항목 캐시와 데이터베이스 캐시를 더 크게 설정할 수도 있습니다. 실제 테스트를 통해 가정을 확인하고 조정합니다.

## 메모리 축소, 파일 시스템 캐시 축소

항목 캐시와 데이터베이스 캐시에 모든 데이터를 저장할 수 없을 뿐만 아니라 파일 시스템 캐시에 데이터를 캐시할 수 있는 메모리도 부족한 시스템을 가정해 보십시오. 이 경우 사용할 수 있는 메모리를 최대한 활용해야 합니다.

항목 캐시와 데이터베이스 캐시 크기를 가능한 한 작게 설정하여 파일 시스템 캐시에서 사용할 수 있는 메모리를 최대한 많이 남겨 두십시오. 파일 시스템 캐시에 메모리를 남겨 두면 최소한 항목이 3에서 4.5의 계수로 데이터베이스나 항목으로 확장되는 것을 방지할 수 있으므로 이론상 디스크 입출력 작업이 감소합니다. 특정 배포에 대해 실제 테스트를 수행하여 이 가정을 확인합니다.

## 업데이트 최적화

최상의 업데이트 성능을 내려면 먼저 발견되는 모든 트랜잭션 로그 병목을 제거합니다. 자세한 내용은 147페이지의 "트랜잭션 로깅"을 참조하십시오.

그런 다음, 메모리의 업데이트를 처리하기에 충분한 메모리를 데이터베이스 캐시에 할당하여 디스크 작업을 최소화합니다. **Directory Server** 콘솔에서 적중률을 읽어 데이터베이스 캐시의 효율성을 모니터링할 수 있습니다. 119페이지의 그림 6-5와 같이 **Directory Server** 콘솔의 "상태" 탭에는 접미사 적중률이 표시됩니다.

또한, 파일 시스템 캐시용으로 상당량의 가용 메모리를 남겨 두십시오. **Directory Server**를 일정 시간 동안 실행하면 디스크 읽기가 필요 없을 정도의 충분한 항목과 색인이 파일 시스템 캐시에 저장될 것입니다. 메모리의 데이터베이스 캐시가 크면 데이터가 자주 플러시되지 않으므로 메모리의 데이터베이스 캐시를 업데이트할 때 성능이 저하됩니다.

데이터를 디스크로 플러시하는 것도 병목이 될 수 있으므로 **Sun StorEdge™** 디스크 어레이와 같은 별도의 **RAID** 시스템에 데이터베이스를 저장하여 업데이트 성능을 향상시킬 수 있습니다. **Solaris** 시스템에서는 **iostat(1M)**와 같은 유틸리티를 사용하여 잠재적인 입출력 병목을 확인할 수도 있습니다. **Windows** 시스템에서 입출력 병목을 처리하는 방법은 **Windows** 도움말 말을 참조하십시오.

## 캐시 초기화 및 모니터

캐시 초기화란 이후의 Directory Server 동작이 램프 업(ramp up)이 아니라 정상적인 작동 성능을 반영하도록 캐시에 데이터를 채우는 것을 의미합니다. 잠재적인 최적화를 측정하고 분석하기 전에 캐시를 초기화하십시오.

ldapsearch 유틸리티를 사용하여 접미사의 항목 캐시를 초기화합니다. 예를 들어 아래 명령을 실행합니다.

```
$ ldapsearch -D directoryManager -w password -b suffix objectclass=* > /dev/null
```

검색을 수행하여 데이터베이스 캐시를 초기화하고 특히 캐시에 색인을 로드합니다.

(mail=\*)와 같은 필터를 사용하여 검색하면 실제 색인을 초기화할 수 있습니다. 다른 색인을 초기화하려면 색인화할 각 속성의 가능한 모든 값을 검색하는 필터 형식을 적용하여 Sun ONE Directory Server Resource Kit searchrate 유틸리티를 사용해 보십시오. 예를 들어 mail 속성에 대한 동일 검색의 성능을 확인하려면 각 전자 우편 주소 줄에 한 개의 전자 우편 주소가 있는 파일을 생성한 다음 searchrate 유틸리티를 사용하여 이 파일로 검색합니다. 예를 들어 아래 명령을 실행합니다.

```
$ searchrate -b suffix -f "(mail=%s)" -i mail.file -K -t 10
```

시간을 절약하려면 -K 옵션과 -t 옵션을 사용하는 것이 좋습니다. searchrate 유틸리티를 -K 옵션과 함께 사용하면 연결을 계속 열어두기 때문에 검색할 때마다 바인드되지 않고 한 번만 바인드됩니다. -t 옵션을 사용하면 사용할 스레드 수를 지정할 수 있습니다. searchrate 유틸리티에 대한 자세한 내용은 Sun ONE Directory Server Resource Kit 설명서를 참조하십시오. Sun ONE Directory Server Resource Kit는 12페이지의 "Directory Server 도구 다운로드"에 설명된 방법으로 구할 수 있습니다.

다른 캐시를 초기화한 후에 사용 가능한 파일 시스템 캐시를 초기화할 수 있습니다. 파일 시스템 캐시를 초기화할 경우 파일 시스템 캐시의 정보가 플러시되지 않도록 차단할 수는 없지만 램프 업(ramp up) 시간을 향상시킬 수 있습니다. UNIX 시스템에서 파일 시스템 캐시를 초기화하려면 슈퍼유저로 dd(1M) 명령을 사용할 수 있습니다. 데이터베이스 파일이 기본 위치에 저장되어 있는 Solaris 시스템에서는 아래 명령을 실행합니다.

```
# for db in ServerRoot/slapd-serverID/db/*/*.db3
> do
> dd if='pwd'/$db of=/dev/null bs=512k
> done
0+1 records in
0+1 records out
...
```

캐시를 초기화한 후 테스트를 수행하여 캐시 조정으로 원하는 결과를 달성했는지 여부를 모니터할 수 있습니다. 119페이지의 그림 6-5와 같이 **Directory Server** 콘솔의 "상태" 탭에서 "접미사" 노드를 선택하면 캐시 모니터 정보가 표시됩니다. 혹은 명령줄에서 아래 명령으로 검색을 실행하여 페이지징 및 캐시 작업을 모니터할 수 있습니다.

```
$ ldapsearch -D admin -w password \
-b cn=monitor,cn=database_name,cn=ldbm\ database,cn=plugins,cn=config
```

데이터베이스 캐시가 충분히 크고 캐시가 초기화되면 적중률(`dbcachehitratio`)은 높고, 읽는 페이지 수(`dbcachepagein`)와 기록되는 클린 페이지 수(`dbcacheroevict`)는 작아야 합니다. 여기서 "높음"과 "낮음"은 배포 제약 조건에 상대적인 의미로 이해해야 합니다.

접미사 항목 캐시가 충분히 크고 캐시가 초기화되면 적중률(`entrycachehitratio`)이 높아야 합니다. 항목 캐시 크기(`currententrycachesize`)는 최대 크기(`maxentrycachesize`)의 80% 이하여야 합니다. 마지막으로 항목 크기(`currententrycachecount`)는 접미사의 총 항목 수와 같거나 비슷해야 합니다.

## 기타 최적화

캐시 크기 조정은 검색, 업데이트 또는 대량 로드 속도를 향상시키는 방법 중 하나입니다. 캐시를 조정하면 캐시의 성능 병목이 시스템의 다른 부분으로 이동합니다. 자세한 내용은 본 설명서의 다른 장을 참조하십시오.

기타 최적화

# 색인화 조정

Directory Server에서 처리하는 항목 수가 늘면 검색에 사용되는 시간과 시스템 자원도 증가합니다. 색인은 검색 성능을 향상시키는 도구입니다. 이 장에서는 특정 배포에 특정 색인을 사용함으로써 발생하는 손실과 이점을 이해할 수 있도록 Directory Server 색인이 어떻게 작동하는지에 대해 설명합니다.

## 색인

색인은 조회 정보와 Directory Server 항목을 연결하며 파일 형식으로 Directory Server 데이터베이스와 함께 저장됩니다. 여기서 *데이터베이스*는 접미사의 물리적 표시입니다. 대부분의 배포 시 접미사와 데이터베이스는 1대1로 대응하지만 한 개의 접미사가 여러 데이터베이스에 분산될 수도 있습니다. Directory Server는 기본적으로 `ServerRoot/slappd-ServerID/db/`에 `nsslapd-directory`의 기본값으로 데이터베이스를 저장합니다. 여기에 저장된 각 데이터베이스 인스턴스는 색인화된 속성 당 한 개의 색인 파일을 갖습니다. 예를 들어, 데이터베이스의 CN 색인 파일인 `example`은 `dc=example,dc=com` 접미사 항목을 저장하며 `ServerRoot/slappd-ServerID/db/example/example_cn.db3`이라고 합니다.

색인화되는 항목은 클라이언트 응용 프로그램이 디렉토리 데이터에 액세스하는 방법에 따라 달라집니다. 표 7-1에는 표준 색인 유형이 간략하게 설명되어 있습니다.

**표 7-1** 표준 색인 유형

색인 유형	질문 형식
근사	속성 값이 <code>foobar</code> 와 비슷한 값을 가진 항목은?
찾아보기	이 가상 목록 보기 검색에 맞는 항목은?
동일	속성 값이 <code>foobar</code> 인 항목은?

**표 7-1** 표준 색인 유형 (계속)

색인 유형	질문 형식
국가별	이 국가별 로케일에 맞는 항목은?
있음	이 속성이 있는 항목은?
하위 문자열	속성 값이 *foo*에 일치하는 항목은?

CN과 같은 특정 속성의 색인 파일에는 여러 유형의 색인이 포함될 수 있습니다. 예를 들어 CN이 동일 및 하위 문자열 유형으로 example 데이터베이스에 색인화되어 있으면 example\_cn.db3에는 동일 색인과 하위 문자열 색인이 모두 포함됩니다.

다음 내용은 *Sun ONE Directory Server 관리 설명서*를 참조하십시오.

- 각 색인 유형에 대한 개요
- 색인 작성 및 삭제 방법
- Directory Server에서 작성하는 기본 색인 목록
- Directory Server에 필요한 시스템 색인 목록

대부분의 경우 기본 색인은 검색 성능을 향상시키며 메시징과 같은 다른 응용 프로그램을 일부 지원합니다. 성능 향상을 위해 특정 기본 색인을 비활성화하거나 삭제할 수도 있습니다. 시스템 색인은 Directory Server 작동에 중요한 역할을 하므로 삭제하거나 수정하지 마십시오.

## 이점: 검색 시 색인의 역할

색인은 검색 속도를 증가시킵니다. 색인에는 값 목록이 포함되어 있으며, 각각의 값은 해당되는 항목 식별자 목록과 연결되어 있습니다. Directory Server는 색인에 있는 항목 식별자 목록을 사용하여 신속하게 항목을 조회할 수 있습니다. 항목 목록을 관리하는 색인이 없으면 Directory Server는 검색에 일치하는 항목을 찾기 위해 접미사의 모든 항목을 확인해야 합니다.

검색 요청이 어떻게 처리되는지 이해하면 색인화된 검색이 색인화되지 않은 검색보다 훨씬 간단하게 처리되는 이유를 명확히 알 수 있습니다. Directory Server에서 각 검색 요청을 처리하는 과정은 다음과 같습니다.

1. 클라이언트 응용 프로그램이 Directory Server로 검색 요청을 보냅니다.

2. Directory Server에서 이 요청을 검사하여 검색 기준이 해당 디렉토리 서버에서 처리할 수 있는 접미사인지 확인합니다. 아닌 경우 클라이언트에 오류를 반환하며, 다른 Directory Server 인스턴스에 대한 참조를 반환할 수도 있습니다.

3. Directory Server에서 해당 검색에 적합한 색인을 관리하고 있는지 확인합니다.

이러한 색인이 있으면 Directory Server는 113페이지의 그림 6-2와 같이 각 색인에서 검색 요청과 일치할 가능성이 있는 후보 항목을 조사합니다.

*이러한 색인이 없으면 Directory Server는 데이터베이스에 있는 모든 항목에서 후보 항목 집합을 생성합니다.* 대규모 배포 시에는 검색에 따라 이 단계에 상당한 시간과 시스템 자원이 소요될 수 있습니다.

4. Directory Server는 각 후보 항목을 검사하여 검색 조건에 일치하는지 확인합니다. Directory Server에서 일치하는 항목을 찾으면 해당 항목은 클라이언트 응용 프로그램으로 반환됩니다.

Directory Server는 149페이지의 "클라이언트에서 사용할 수 있는 자원 제한"에 설명된 것처럼 `nsslapd-lookthroughlimit`, `nsslapd-sizelimit` 또는 `nsslapd-timelimit`와 같은 자원 제한에 도달하거나 모든 후보 항목이 검사될 때까지 계속해서 후보 항목을 검사합니다.

단계 3에서 알 수 있듯이 색인을 사용하면 Directory Server가 클라이언트의 검색 요청에 응답하여 수행하는 처리량을 크게 줄일 수 있습니다.

## 손실: 업데이트 시 색인에 미치는 영향

업데이트는 항목 자체는 물론 해당 항목을 참조하는 색인까지 변경합니다. 색인에 항목 참조가 많을수록 업데이트 중에 색인을 수정하기 위한 처리 손실이 증가합니다. 즉, Directory Server는 *클라이언트 응용 프로그램에 업데이트 확인 메시지를 보내기 전에* 115페이지의 그림 6-3과 같이 영향을 받는 모든 색인을 수정합니다.

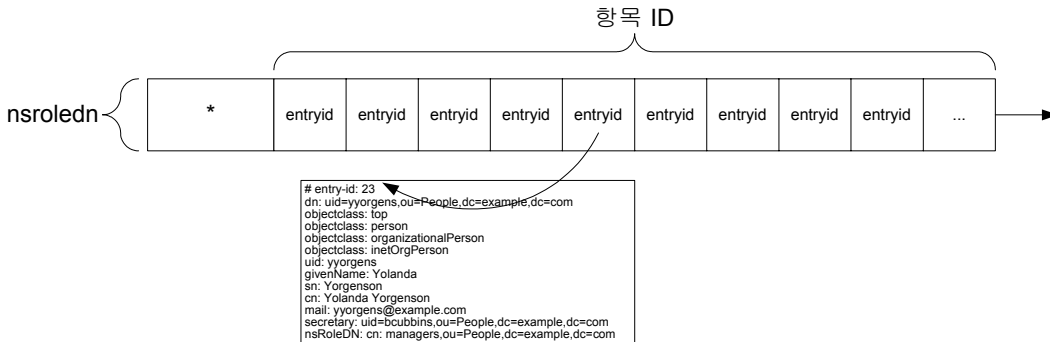
색인 유지관리를 위한 처리 손실 외에도 색인은 디스크 공간과 메모리 공간을 소모합니다. 118페이지의 "검색 최적화"에 설명된 것처럼 데이터베이스 캐시 크기를 검색에 맞게 최적화하는 경우 데이터베이스 캐시에 항목과 색인을 모두 저장할 수 있는 메모리를 제공할 수도 있습니다. 색인이 클수록 더 많은 공간이 필요하고 64비트 색인이 32비트 색인보다 많은 공간을 사용합니다.

일반적으로 Directory Server 인스턴스의 색인화 조정은 색인을 유지관리함으로써 검색 처리 속도를 증가시켜 업데이트 처리 및 공간 소모 비용을 상쇄할 수 있다는 전제 하에 이루어집니다. 따라서 유용한 색인을 유지관리하는 것이 바람직하며 클라이언트가 거의 검색하지 않는 속성의 색인을 유지관리하는 것은 비용 낭비라고 할 수 있습니다.

## 실제 색인

그림 7-1은 nsRoleDN 속성에 대한 실제 색인으로, 단순히 nsRoleDN 속성이 있는 모든 데이터베이스 항목이 포함되어 있을 뿐이지만 이 색인이 어떻게 속성 값과 별개로 작동하는지 보여줍니다. 각 속성 값은 \*와 일치 합니다.

그림 7-1 실제 색인의 예



그림과 같이 Directory Server는 `entryid` 속성 값을 사용하여 빠른 검색을 허용하는 항목 참조를 저장할 수 있습니다. Directory Server는 실제로 `dbinstance_id2entry.db3` 색인 파일을 사용하여 항목을 검색합니다. 여기서 `dbinstance`는 125페이지의 "색인"에 설명된 데이터베이스 식별자에 따라 달라집니다.

Directory Server는 있음 유형으로 색인화된 속성이 포함된 항목의 업데이트 요청을 받으면 클라이언트 응용 프로그램에 업데이트 확인 메시지를 반환하기 전에 색인에서 해당 항목을 제거할지 여부를 결정한 다음 필요한 항목을 모두 수정해야 합니다.

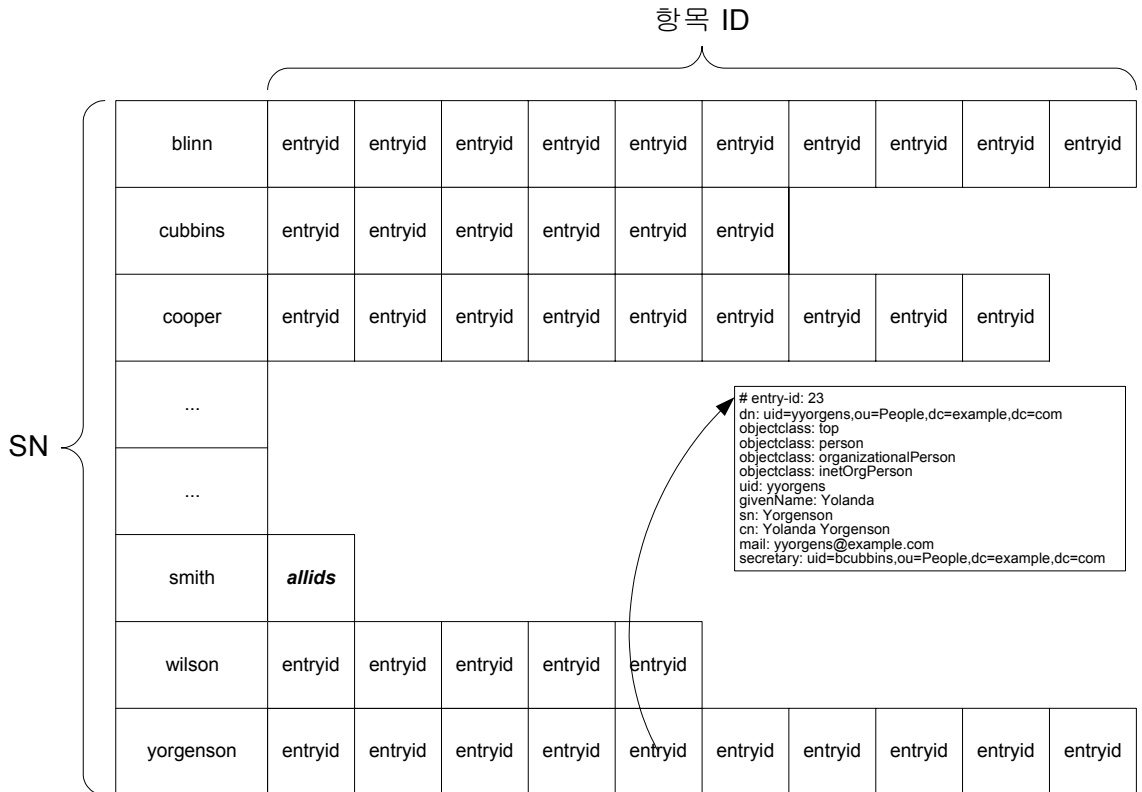
실제 색인에 다수의 항목이 유지관리되는 경우도 있지만 실제 색인으로 인한 손실은 일반적으로 다른 색인 유형보다 적습니다.



## 동일 색인

그림 7-2는 SN(성) 속성의 동일 색인으로, 이 색인이 SN 속성 값별로 해당 값을 가진 항목 목록을 유지관리하는 방법을 보여줍니다.

그림 7-2 동일 색인의 예



Directory Server는 동일 유형으로 색인화된 속성이 있는 항목의 업데이트 요청을 받으면 클라이언트 응용 프로그램에 업데이트 확인 메시지를 반환하기 전에 색인에서 해당 항목을 제거할지 여부와 색인에서 목록을 추가하거나 제거할지 여부를 결정한 다음 필요한 항목을 모두 수정해야 합니다.

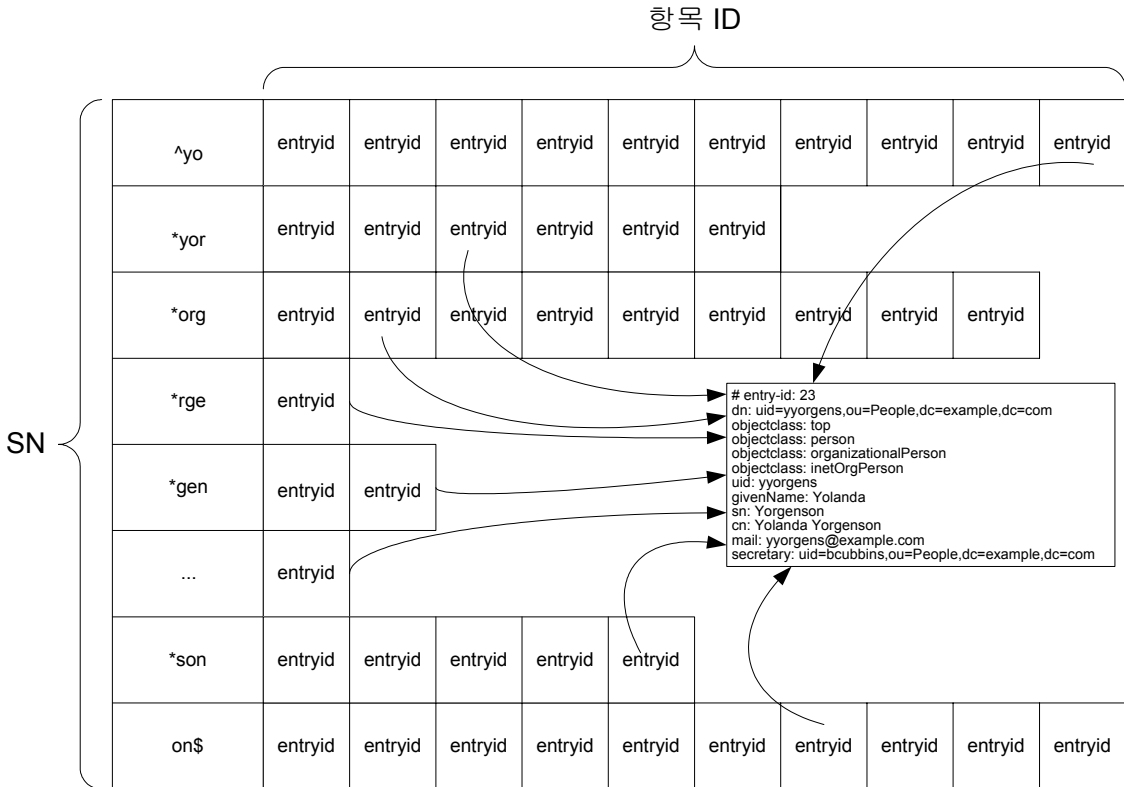
예를 들어, 동일 색인은 일반적으로 하위 문자열 색인보다 손실이 적지만 공간 사용 면에서 실제 색인보다 손실이 큽니다. 하지만 메시징 서버와 같은 일부 클라이언트 응용 프로그램은 최상의 검색 성능을 위해 동일 색인을 사용합니다. 사진이나 암호화된 암호와 같은 큰 이진 속성에 대해서는 동일 색인을 사용하지 마십시오.

## 하위 문자열 색인

그림 7-3은 SN(성) 속성의 하위 문자열 색인으로, 이 색인이 어떻게 속성 값별로 일련의 목록을 유지관리하는지 보여줍니다.

Directory Server는 색인에서 두 문자의 하위 문자열을 검색할 수 있도록 하위 문자열을 색인화합니다. 따라서 색인을 사용하면 (sn=\*ab\*) 검색 속도는 증가시킬 수 있지만 (sn=\*a\*) 검색 속도는 향상되지 않습니다.

**그림 7-3** 하위 문자열 색인의 예



Directory Server는 와일드카드 앞에 한 문자뿐인 이니셜 하위 문자열 검색을 허용하여 최적화를 향상시킵니다. 따라서 하위 문자열 색인이 있는 경우 (sn=a\*) 검색 속도도 증가시킬 수 있지만 (sn=\*a\*) 또는 (sn=\*a) 검색 속도는 향상되지 않습니다.

Directory Server는 자체 기본 규칙에 따라 하위 문자열 색인을 구축하므로 시스템 관리자도 이러한 하위 문자열을 구성할 수 없습니다.

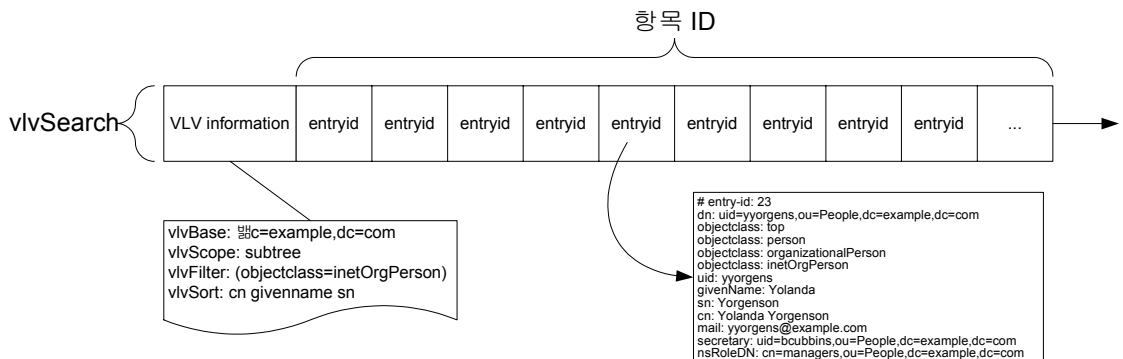
Directory Server는 하위 문자열 유형으로 색인화된 속성이 있는 항목의 업데이트 요청을 받으면 클라이언트 응용 프로그램에 업데이트 확인 메시지를 반환하기 전에 색인에서 해당 항목을 제거할지 여부와 이 항목을 수정할 때 색인에 미치는 영향을 결정하고 색인에서 항목 ID 또는 항목 ID 목록을 추가하거나 제거할지 여부를 결정한 다음 필요한 항목을 모두 수정해야 합니다. 업데이트 수는 속성 값의 문자열 길이에 따라 달라집니다.

하위 문자열 색인을 유지관리하면 일반적으로 상당한 손실이 발생합니다. 이러한 손실은 대체로 색인화된 문자열의 길이에서 비롯되므로 특히 description과 같이 긴 문자열 값을 갖는 속성에 대해서는 불필요한 하위 문자열 색인을 사용하지 마십시오. 사건과 같은 이진 속성은 하위 문자열 색인을 적용할 수 없습니다.

## 찾아보기(가상 목록 보기) 색인

그림 7-4는 가상 목록 보기용 찾아보기 색인으로, 가상 목록 보기 정보인 vlvBase, vlvScope, vlvFilter 및 vlvSort 속성 값에 의해 찾아보기 색인이 어떻게 영향을 받는지 보여줍니다. 이 색인 유형의 항목 ID는 vlvSort를 기준으로 정렬됩니다.

그림 7-4 찾아보기 색인의 예



Directory Server는 `vlvFilter` 값에 일치하는 항목의 업데이트 요청을 받으면 클라이언트 응용 프로그램에 업데이트 확인 메시지를 반환하기 전에 색인에서 해당 항목을 제거할지 여부와 목록에서 올바른 항목 위치를 결정한 다음 필요한 항목을 모두 수정해야 합니다.

## 근사 색인

Directory Server는 *Metaphone Phonetic* 알고리즘의 변형을 사용하여 근사 색인을 유지관리합니다. 이 알고리즘은 속성 문자열 값을 유사한 영어 발음으로 분석합니다. 검색 요청에 따라 비교할 값도 동일한 알고리즘을 사용하여 처리됩니다. 이 알고리즘은 음절을 기초로 하기 때문에 전화 번호와 같이 숫자가 포함된 속성에는 효과가 없습니다.

이 알고리즘은 각각의 속성 값 문자열에 대해 한 개의 대상 문자열을 생성하기 때문에 영어 문자열의 "유사 발음" 색인화는 동일 색인화와 손실이 비슷합니다.

## 국가별 색인

국가별 색인은 특정 로캘에 대한 일치 규칙을 사용하여 색인을 유지관리합니다. 따라서 이 색인은 하위 문자열 색인 및 동일 색인과 손실이 비슷합니다.

사용자 정의 일치 규칙 서버 플러그인을 사용하면 국가별 색인 및 기타 색인 유형에 대한 표준 지원을 확장할 수 있습니다. 사용자 정의 일치 규칙 플러그인에 대한 자세한 내용은 *Sun ONE Directory Server Plug-In API Programming Guide*를 참조하십시오.

## 예제: 항목 색인화

아래 코드 예제와 같이 `uid`에 대한 동일 검색, 일반 이름(`cn`)과 성(`sn`) 속성에 대한 동일, 하위 문자열 및 근사 검색, `mail` 속성에 대한 동일 검색, `telephoneNumber` 속성에 대한 동일 및 하위 문자열 검색, `description` 속성에 대한 하위 문자열 검색을 위해 색인화된 접미사에 사용자 항목을 추가하는 경우를 가정해 보십시오.

**코드 예 7-1**                      샘플 사용자 항목

```
dn: uid=yyorgens,ou=People,dc=example,dc=com
objectclass: top
objectclass: person
objectclass: organizationalPerson
objectclass: inetOrgPerson
uid: yyorgens
givenName: Yolanda
sn: Yorgenson
cn: Yolanda Yorgenson
mail: yolanda.yorgenson@example.com
telephoneNumber: 1-650-960-1300
description: Business Development Manager, Platinum Partners
```

Directory Server는 이 항목을 추가할 때 cn, sn, mail, telephoneNumber 및 description 속성의 색인을 수정해야 합니다. 표 7-2에는 예상되는 항목 수가 나와 있습니다.

**표 7-2**                      샘플 사용자 항목의 색인 업데이트

속성	근사	동일	하위 문자열 <sup>1</sup>	색인 업데이트 총수
uid		1		1
cn	1	1	17	19
sn	1	1	9	11
mail		1		1
telephoneNumber		1	11	12
description			47	47

1. 이 코드 예제의 description 문자열처럼 긴 문자열에 대해 하위 문자열 색인화를 사용하는 것은 대부분의 배포 시 바람직하지 않습니다.

description 문자열에 대한 하위 문자열 색인 업데이트 수(47)가 다른 모든 속성을 합친 업데이트 수(44)보다 많습니다. description 문자열을 추가로 수정하면 새 문자열에 따라 다시 최대 수 이상의 업데이트가 필요할 수 있습니다. 가능하면 description 값처럼 긴 문자열에는 하위 문자열 색인화를 적용하지 마십시오.

## 성능 향상을 위한 색인화 조정

대체로 성능 향상을 위한 색인화 조정은 빈번한 검색의 속도를 증가시키는 색인을 활성화하고 유지관리 손실이 크며 자주 사용되지 않는 색인은 비활성화하는 것을 의미합니다.

---

**주** 데이터베이스 백업에는 색인이 포함되므로 **Directory Server** 구성과 일치해야 합니다.

색인 구성 방법을 변경한 후 구성과 데이터를 모두 백업하십시오.

---

특정 응용 프로그램 전용의 복제본이 필요한 대규모 배포 시에는 **Directory Server** 인스턴스마다 다른 색인을 구성할 수도 있습니다. 예를 들어, 다음과 같은 요소로 구성된 토폴로지를 가짐을 보십시오.

- 쓰기만 처리하는 마스터
- 소비자에 대한 복제 로드를 처리하는 허브
- 메시징과 같은 특정 응용 프로그램 전용의 일부 소비자

이 경우 마스터는 검색을 처리하지 않으므로, 예를 들어 마스터에서 광범위한 하위 문자열 색인을 유지관리하지 않도록 선택할 수 있습니다. 거의 사용되지 않는 일부 색인을 확인하여 비활성화할 수도 있습니다.

허브는 본질적으로 관리 요청 이외의 클라이언트 요청은 받지 않으므로 이 경우 **Directory Server** 자체에 필요한 시스템 색인을 제외한 모든 색인을 비활성화할 수 있습니다.

개별 응용 프로그램 전용의 소비자에서는 이 응용 프로그램에서 사용하지 않는 모든 색인을 비활성화할 수 있습니다. 비활성화되는 색인은 특정 응용 프로그램에서 수행하는 검색에 따라 달라집니다.

## 색인화된 검색만 허용

**Directory Server**는 색인화되지 않은 검색을 요청하는 클라이언트에 **LDAP\_UNWILLING\_TO\_PERFORM**을 반환하여 손실이 큰 색인화되지 않은 검색을 방지할 수 있습니다.

특정 데이터베이스에 대해 색인화되지 않은 검색을 방지하려면 아래 명령을 실행하여 이 데이터베이스의 **nsslapd-require-index** 속성 값을 **on**으로 설정합니다.

```
$ ldapmodify -h host -p port -D "cn=directory manager" -w password
dn: cn=example,cn=lldb database, cn=plugins, cn=config
changetype: modify
replace: nsslapd-require-index
nsslapd-require-index: on
^D (^Z on Windows systems)
```

변경 사항은 즉시 적용되므로 Directory Server를 다시 시작할 필요가 없습니다.

## 색인 목록 길이 제한

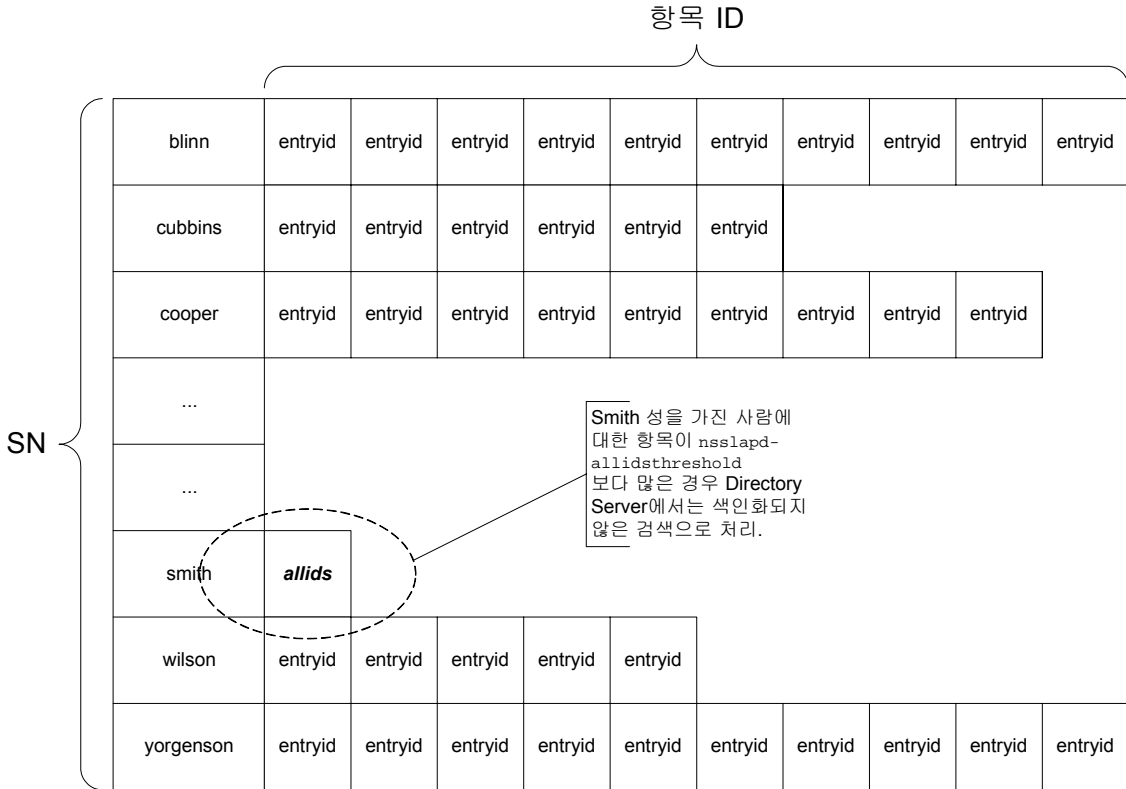
급성장하는 대규모 디렉토리 배포 시에는 색인화가 특정 색인 키에 대한 한계효용 체감점에 도달할 수 있습니다. 한계효용 체감점에 도달하면 특정 키와 관련된 목록이 너무 길어져 때때로 해당 키에서 후보 항목을 찾는 색인화되지 않은 검색을 수행하는 것보다 이 목록을 유지관리함으로써 발생하는 손실이 더 큼니다. 예를 들어, 성 속성에 대해 동일 유형으로 색인화된 큰 전화 번호부 응용 프로그램을 가정해 보십시오. 전화 번호부에 **Smiths**라는 이름이 너무 많아서 **Smiths**에 대해 색인을 유지관리하는 손실이 조회 이점보다 크다고 가정합니다. 이때 Directory Server는 **Smith** 성의 색인화를 중지해야 합니다. 하지만 다른 성의 색인화는 Directory Server에서 계속되어야 합니다.

Directory Server에는 이 작업을 처리하기 위한 메커니즘이 있습니다. 구성 속성을 임계값으로 설정하십시오. 특정 키에 대한 목록의 항목 수가 설정된 값에 도달하면 Directory Server는 해당 키에 대한 후보 항목을 찾기 위해 색인화되지 않은 검색을 수행하도록 지정하는 토큰으로 키 목록을 바꿉니다. 이 값은 검색 중에 확인되는 최대 후보 항목 수보다 약간 작으며 150페이지의 표 9-1에 설명된 것처럼 `nsslapd-lookthroughlimit`를 사용하여 설정됩니다.

이 메커니즘은 전역 임계값(`cn=config,cn=lldb database,cn=plugins,cn=config`의 `nsslapd-allidsthreshold`) 설정에 사용된 구성 속성의 이름을 따서 *모든ID 임계*라고 합니다. 현재 이 값은 Directory Server 인스턴스에서 전역으로 사용되며 각 색인에 대해 다르게 설정할 수 없습니다.

그림 7-5는 `nsslapd-allidsthreshold`보다 많은 수의 **Smiths** 성에 대한 색인화의 예를 보여줍니다.

**그림 7-5** 색인 키에 대한 모든 ID 임계에 도달



임계는 색인 테이블에 있는 한 개의 목록에만 영향을 주기 때문에 다른 키 목록은 영향을 받지 않습니다.

### 색인 목록 크기가 부적절한 경우의 증상

클라이언트에서 주로 색인화된 검색을 수행하고 6장, "캐시 크기 조정"에 설명된 것처럼 캐시 크기를 올바르게 조정했는데도 여전히 검색 성능이 느릴 경우 부적절한 임계값이 원인일 수 있습니다. 색인화된 검색의 성능이 느리면 먼저 캐시 크기가 올바르게 조정되었는지 확인하십시오. 그런 다음, access 로그를 검사하여 Directory Server가 모든 ID 임계에 자주 도달하는지 확인합니다.



access 로그의 RESULT 메시지 끝에 있는 notes=U 플래그는 Directory Server가 색인화되지 않은 검색을 수행했음을 나타냅니다. 동일한 연결 및 작업에 대한 이전 SRCH 메시지는 사용되는 검색 필터를 지정합니다. 아래 두 줄의 예제는 10000개 항목을 반환하는 색인화되지 않은 (cn=Smith) 검색을 추적합니다. 타임스탬프는 메시지에서 제거되었습니다.

```
conn=2 op=1 SRCH base="o=example.com" scope=0 filter="(cn=Smith)"
conn=2 op=1 RESULT err=0 tag=101 nentries=10000 notes=U
```

색인화해야 하는 이러한 검색 쌍이 많이 발견되면 임계를 증가시켜 검색 성능을 향상시킬 수도 있습니다.

## 색인 목록 임계 크기 변경

nsslapd-allidsthreshold에 적합한 값은 일반적으로 디렉토리에 있는 총 항목 수의 약 5% 범위입니다. 예를 들어, 기본값 4000은 80,000 이하의 항목을 처리하는 Directory Server 인스턴스에 적합합니다. 근시일 내에 디렉토리에 많은 수의 항목을 추가할 예정이거나 디렉토리가 상당히 확장될 것으로 예상되면 총수의 5%보다 훨씬 높은 값을 설정할 수 있습니다. 거의 쓰기만 지원하는 마스터와 많은 검색을 지원하는 소비자 복제본에 각각 다른 임계를 설정할 수도 있습니다. nsslapd-allidsthreshold 속성 값을 변경할 경우 모든 색인을 다시 작성해야 하므로 근시일 내에 LDIF를 사용하여 큰 디렉토리를 다시 초기화할 예정이면 초기화 직전에 값을 조정해도 됩니다. 어느 경우든지 명확한 이유가 없으면 대규모 배포에 대해서도 모든 ID 임계를 매우 높게(50,000 이상) 설정하지는 마십시오.

다음과 같이 모든 ID 임계를 변경합니다. 변경 중에는 Directory Server 인스턴스에서 서비스가 중단됩니다.

1. 해당 Directory Server 인스턴스를 중지합니다.
2. 모든 디렉토리 데이터베이스를 LDIF로 내보냅니다.  
자세한 내용은 *Sun ONE Directory Server 관리 설명서*를 참조하십시오.
3. `ServerRoot/slapd-ServerID/config/dse.ldif`의 nsslapd-allidsthreshold 속성 값을 신중하게 조정합니다.
4. LDIF를 사용하여 모든 디렉토리 데이터베이스를 다시 초기화합니다.  
자세한 내용은 *Sun ONE Directory Server 관리 설명서*를 참조하십시오.

5. 데이터베이스 캐시 크기가 이전의 모든 ID 임계값에 맞춰 조정되었으며 서버의 물리적 메모리가 충분하면 데이터베이스 캐시 크기를 임계 증가값의 25%만큼 늘리는 것이 좋습니다.

즉, 모든 ID 임계를 4000에서 6000으로 늘리는 경우 증가된 색인 목록 크기를 수용하기 위해 데이터베이스 캐시 크기를 약 12.5%만큼 늘릴 수 있습니다. 생산 서버에 변경 사항을 적용하기 전에 실제 테스트를 통해 최적 크기를 확인합니다. 데이터베이스 캐시 조정에 대한 자세한 내용은 6장, "캐시 크기 조정"을 참조하십시오.

6. Directory Server 인스턴스를 다시 시작합니다.

## 색인 단편화 문제 해결

대규모 색인과 높은 업데이트 비율을 지원하는 Directory Server 인스턴스에서 과도한 색인 키 단편화는 심각한 문제가 될 수 있습니다. 색인 키 단편화가 과도하면 데이터베이스 크기는 같아도 성능이 저하됩니다. 과도한 색인 키 단편화로 인해 서버 성능이 크게 저하된 경우 영향을 받은 색인을 다시 생성하여 단편화를 줄이는 것이 좋습니다.

색인 작성에 대한 자세한 내용은 *Sun ONE Directory Server 관리 설명서*를 참조하십시오.

# 로깅 조정

Directory Server는 표 8-1에 요약된 것처럼 여러 가지 로그 유형을 제공합니다. 이 장에서는 이러한 로그 유형을 처리하는 방법에 대해 설명합니다.

**표 8-1** Directory Server에서 사용하는 로그 유형

로그	유형	용도
액세스	플랫 파일	디렉토리 사용 패턴 평가, 구성 설정 확인, 액세스 문제 진단
감사	플랫 파일	보안 및 데이터 무결성에 대한 감사 기록 제공
변경 로그	데이터베이스	복제본 간의 동기화 허용
오류	플랫 파일	디렉토리 배포 디버깅
레트로 변경 로그	데이터베이스	이전 버전과의 호환성 허용
트랜잭션	데이터베이스	데이터베이스 무결성 유지

대규모 배포 시 로그에 쓰면 디스크 사용량이 많아져서 성능이 크게 저하될 수 있습니다. 대규모 시스템의 많은 로깅 작업으로 인한 잠재적 입출력 병목을 감안하면 자체 디스크 컨트롤러가 있는 별도의 물리적 디스크에 로그를 저장하는 것이 좋습니다.

# 액세스 로깅

액세스 로그에는 클라이언트 연결 및 수행된 작업에 대한 자세한 정보가 포함되어 있습니다. 액세스 문제를 진단하거나 서버 구성을 확인하고 서버 사용 패턴을 평가할 때 액세스 로그는 큰 도움이 됩니다. 하지만 기본 로깅 수준은 대부분의 배포 시 상당한 디스크 작업을 초래하므로 서버 성능이 저하될 수 있습니다.

액세스 로그는 유용한 문제 해결 정보를 제공하지만 입출력 병목이 될 수도 있습니다. 디렉토리가 배포된 후 오류나 성능 문제 없이 실행되면 액세스 로깅을 비활성화하는 것이 좋습니다. 작업 환경에서 액세스 로깅이 필요하다면 로깅 수준을 최소 필수 수준으로 설정하십시오. 또한, 자체 물리적 디스크나 큰 입출력 버퍼가 있는 빠른 디스크 하위 시스템에 액세스 로그를 저장하는 것이 좋습니다. 표 8-2에는 특정 속성에 대해 권장되는 추가 작업이 나와 있습니다.

**표 8-2** 액세스 로깅에 대해 권장되는 조정 작업

구성 속성(on dn: cn=config)	간략한 설명 및 권장되는 조정 작업
nsslapd-accesslog	<p>액세스 로그 파일의 경로와 파일 이름을 지정합니다.</p> <p>소규모 배포 시에는 액세스 로그와 감사 및 오류 로그가 디스크를 공유할 수 있습니다.</p> <p>대규모 배포 시에는 별도의 컨트롤러가 있는 자체 디스크나 디스크 하위 시스템에 액세스 로그를 저장하는 것이 좋습니다. 큰 입출력 버퍼가 있는 디스크를 선택합니다.</p>
nsslapd-accesslog-level	<p>사용되는 정보 로깅 수준을 지정합니다.</p> <p>높은 수준이 필요하지 않으면 속성 값을 0, 액세스 로깅 안 함으로 변경합니다(기본값은 256, 항목에 대한 액세스 로깅).</p>
nsslapd-accesslog-logbuffering	<p>액세스 로그의 버퍼링 여부를 지정합니다.</p> <p>표시되는 액세스 로그 메시지를 보기 위해 버퍼링을 비활성화해야 하는 경우가 아니면 속성 값을 on(기본값)으로 그대로 두십시오. 버퍼링을 비활성화하면 전체적인 성능이 저하될 수 있습니다.</p>

**표 8-2** 액세스 로깅에 대해 권장되는 조정 작업 (계속)

구성 속성(on dn: cn=config)	간략한 설명 및 권장되는 조정 작업
nsslapd-accesslog-logging-enabled	<p>액세스 로깅을 활성화하거나 비활성화합니다.</p> <p>최상의 성능을 내려면 속성 값을 off로 변경합니다(기본 값은 on).</p> <p>배포 시 액세스 로깅을 사용해야 하는 경우 nsslapd-accesslog-level 속성을 가장 낮은 값으로 설정하고 자체 디스크나 디스크 하위 시스템에 액세스 로그를 저장합니다. 자주(매일 또는 매주) 액세스 로그를 순환시키고 nsslapd-accesslog-logmaxdiskspace 및 nsslapd-accesslog-logminfreediskspace 속성을 사용하여 디스크 공간 사용을 관리하십시오.</p>
nsslapd-accesslog-logmaxdiskspace	<p>모든 액세스 로그(현재 로그와 순환된 로그)에서 사용할 수 있는 최대 디스크 공간을 지정합니다.</p> <p>액세스 로깅에 할당된 총 디스크 공간보다 작은 값을 설정하십시오.</p> <p>감사, 액세스 및 오류 로깅을 같은 디스크에 저장하는 경우 해당 디스크 공간이 세 가지 로깅을 저장하기에 충분해야 합니다.</p> <p>액세스 로그가 자체 디스크에 있으면 이 변수를 디스크 크기로 설정합니다.</p>
nsslapd-accesslog-logminfreediskspace	<p>이전의 로그가 제거되지 않은 상태에서 사용 가능한 최소 디스크 공간을 지정합니다.</p> <p>사용할 수 있는 디스크 공간이 이 속성에 지정된 값보다 작으면 이 속성의 설정에 맞게 충분한 디스크 공간이 확보될 때까지 가장 오래된 액세스 로그부터 삭제됩니다. 디스크가 가득 차서 액세스 로그를 쓸 수 없으면 서버가 종료됩니다.</p>

각 구성 속성에 대한 자세한 내용은 *Sun ONE Directory Server Reference Manual*을 참조하십시오.

Sun ONE Directory Server Resource Kit 설명서에는 액세스 로그에서 정보를 추출하는 방법이 설명되어 있습니다. 자세한 내용은 12페이지의 "Directory Server 도구 다운로드"를 참조하십시오.

# 감사 로깅

감사 로그에는 각 데이터베이스 및 서버 구성의 모든 변경 사항에 대한 자세한 정보가 포함되어 있습니다. 감사 로깅은 기본적으로 사용되지 않습니다.

수정 용량이 많은 배포 시 감사 로깅을 사용하면 전체적인 성능이 크게 저하됩니다. 배포에 필요한 경우가 아니면 감사 로깅은 비활성화된 상태로 두십시오. 감사 로깅이 필요한 대규모 또는 고용량 배포 시에는 별도의 컨트롤러가 있는 자체 디스크를 감사 로그에 할당하는 것이 좋습니다. 표 8-3에는 특정 속성에 대해 권장되는 추가 작업이 나와 있습니다.

**표 8-3** 감사 로깅에 대해 권장되는 조정 작업

구성 속성(on dn: cn=config)	간략한 설명 및 권장되는 조정 작업
nsslapd-auditlog	<p>감사 로그 파일의 경로와 파일 이름을 지정합니다.</p> <p>소규모 배포 시에는 감사 로그와 액세스 및 오류 로그가 디스크를 공유할 수 있습니다.</p> <p>대규모 배포 시에는 별도의 컨트롤러가 있는 자체 디스크에 감사 로그를 저장하는 것이 좋습니다. 큰 입출력 버퍼가 있는 디스크를 선택합니다.</p>
nsslapd-auditlog-logging-enabled	<p>감사 로깅을 활성화하거나 비활성화합니다.</p> <p>감사 로깅이 필요하지 않으면 속성 값을 <code>off</code>(기본값)로 그대로 두십시오.</p>
nsslapd-auditlog-logmaxdiskpace	<p>모든 감사 로그(현재 로그와 순환된 로그)에서 사용할 수 있는 최대 디스크 공간을 지정합니다.</p> <p>감사 로깅에 할당된 총 디스크 공간보다 작은 값을 설정하십시오.</p> <p>감사, 액세스 및 오류 로깅을 같은 디스크에 저장하는 경우 해당 디스크 공간이 세 가지 로깅을 저장하기에 충분해야 합니다.</p> <p>감사 로그가 자체 디스크에 있으면 이 변수를 디스크 크기로 설정합니다.</p>

**표 8-3** 감사 로깅에 대해 권장되는 조정 작업 (계속)

구성 속성(on dn: cn=config)	간략한 설명 및 권장되는 조정 작업
nsslapd-auditlog-logminfreediskspace	<p>이전의 로그가 제거되지 않은 상태에서 사용 가능한 최소 디스크 공간을 지정합니다.</p> <p>사용할 수 있는 디스크 공간이 이 속성에 지정된 값보다 작으면 이 속성의 설정에 맞게 충분한 디스크 공간이 확보될 때까지 가장 오래된 감사 로그부터 삭제됩니다. 디스크가 가득 차서 감사 로그를 쓸 수 없으면 서버가 종료됩니다.</p>

각 구성 속성에 대한 자세한 내용은 *Sun ONE Directory Server Reference Manual*을 참조하십시오.

## 오류 로깅

Directory Server 인스턴스의 오류 로그에는 정상적인 서버 작동 중에 표시되는 자세한 오류, 경고 및 정보 메시지가 포함되어 있습니다. 낮은 기본 로깅 수준에서는 비교적 적은 디스크 작업이 발생합니다.

하지만 디버깅 정보를 생성하기 위해 로그 수준을 높게 설정하면 Directory Server는 대량의 메시지를 디스크에 쓰기 시작합니다. 이러한 쓰기 로드로 인해 전체적인 성능이 크게 저하될 수 있습니다. 성능 저하를 방지하려면 동시에 모든 구성 요소의 로그 수준을 활성화하는 대신 구성요소별로 하나씩 로그 수준을 올리십시오.

오류 로그는 로그 버퍼링을 지원하지 않으므로 모든 메시지가 즉시 디스크로 플러시됩니다. 대규모 또는 고용량 배포 시에는 별도의 컨트롤러가 있는 자체 디스크를 오류 로그에 할당하는 것이 좋습니다. 오류 로그는 디버깅이 필요할 때 사용됩니다. 표 8-4에는 특정 속성에 대해 권장되는 작업이 나와 있습니다.

**표 8-4** 오류 로깅에 대해 권장되는 조정 작업

구성 속성(on dn: cn=config)	간략한 설명 및 권장되는 조정 작업
nsslapd-errorlog	<p>오류 로그 파일의 경로와 파일 이름을 지정합니다.</p> <p>소규모 배포 시에는 오류 로그와 액세스 및 감사 로그가 디스크를 공유할 수 있습니다.</p> <p>대규모 배포 시에는 별도의 컨트롤러가 있는 자체 디스크에 오류 로그를 저장하는 것이 좋습니다. 큰 입출력 버퍼가 있는 디스크를 선택합니다.</p>
nsslapd-errorlog-logging-enabled	<p>오류 로깅을 활성화하거나 비활성화합니다.</p> <p>속성 값을 on(기본값)으로 그대로 두십시오.</p>
nsslapd-errorlog-logmaxdiskspace	<p>모든 오류 로그(현재 로그와 순환된 로그)에서 사용할 수 있는 최대 디스크 공간을 지정합니다.</p> <p>오류 로깅에 할당된 총 디스크 공간보다 작은 값을 설정하십시오.</p> <p>감사, 액세스 및 오류 로깅을 같은 디스크에 저장하는 경우 해당 디스크 공간이 세 가지 로깅을 저장하기에 충분해야 합니다.</p> <p>오류 로그가 자체 디스크에 있으면 이 변수를 디스크 크기로 설정합니다.</p>
nsslapd-errorlog-logminfreediskspace	<p>이전의 로그가 제거되지 않은 상태에서 사용 가능한 최소 디스크 공간을 지정합니다.</p> <p>사용할 수 있는 디스크 공간이 이 속성에 지정된 값보다 작으면 이 속성의 설정에 맞게 충분한 디스크 공간이 확보될 때까지 가장 오래된 오류 로그부터 삭제됩니다. 디스크가 가득 차서 오류 로그를 쓸 수 없으면 서버가 종료됩니다.</p>
nsslapd-infolog-area	<p>해당 정보 메시지를 기록할 구성 요소를 지정합니다.</p> <p>구성 요소를 디버깅하는 경우가 아니면 속성 값을 0(기본값)으로 그대로 두십시오. 생산 서버에서는 한 번에 하나의 구성 요소에 대해서만 설정하는 것이 좋습니다.</p>



**표 8-4** 오류 로깅에 대해 권장되는 조정 작업 (계속)

구성 속성(on dn: cn=config)	간략한 설명 및 권장되는 조정 작업
nsslapd-infolog-level	사용되는 정보 로깅 수준을 지정합니다.  nsslapd-infolog-area 설정만으로 충분한 세부 정보가 생성되지 않는 구성 요소를 디버깅하는 경우가 아니라면 속성 값을 0(기본값)으로 그대로 두십시오.

각 구성 속성에 대한 자세한 내용은 *Sun ONE Directory Server Reference Manual*을 참조하십시오.

## 다중 마스터 복제 변경 로깅

Directory Server는 복제 변경 로그를 사용하여 복제본 간의 동기화를 허용합니다. 변경 로그에 대해서는 *Sun ONE Directory Server Deployment Guide*를, 구성 세부 정보에 대해서는 *Sun ONE Directory Server Reference Manual*을 참조하십시오. 표 8-5에는 특정 속성에 대해 권장되는 추가 작업이 나와 있습니다.

**표 8-5** 다중 마스터 변경 로깅에 대해 권장되는 조정 작업

구성 항목 DN 및 구성 속성	간략한 설명 및 권장되는 조정 작업
dn: cn=changelog5,cn=config nsslapd-cachememsize	변경 로그 데이터베이스의 캐시 크기를 지정합니다.  대규모 배포 시에는 기본값 10MB를 변경하는 것이 좋습니다.
dn: cn=changelog5,cn=config nsslapd-changelogdir	변경 로그 데이터베이스의 경로와 파일 이름을 지정합니다.  별도의 컨트롤러가 있는 자체 디스크나 디스크 하위 시스템에 변경 로그를 저장하는 것이 좋습니다. 큰 입출력 버퍼가 있으면 도움이 됩니다.
dn: cn=changelog5,cn=config nsslapd-changemaxage	변경 로그에 있는 항목의 최대 수명을 지정합니다.  속성 값 0(기본값, 제한 없음)을 복제된 서버가 완전히 동기화 되어 변경 로그를 지울 수 있을 때까지의 기간으로 변경합니다.

**표 8-5** 다중 마스터 변경 로깅에 대해 권장되는 조정 작업 (계속)

구성 항목 DN 및 구성 속성	간략한 설명 및 권장되는 조정 작업
dn: cn=changelog5,cn=config nsslapd-changemaxentries	변경 로그의 최대 항목 수를 지정합니다. 속성 값 0(기본값, 제한 없음)을 변경 로그가 지워지기 전에 복제된 서버가 완전히 동기화될 수 있는 개수로 변경합니다.
dn: cn=changelog5,cn=config nsslapd-cachesize	변경 로그 데이터베이스 캐시의 최대 항목 수를 지정합니다. 속성 값 -1(기본값, 제한 없음)을 항목이 플러시되지 않고 변경 로그에 보관되는 최대 항목 수로 변경합니다.

각 구성 속성에 대한 자세한 내용은 *Sun ONE Directory Server Reference Manual*을 참조하십시오.

## 역 변경 로깅

Directory Server는 공급자 서버의 변경 사항을 Directory Server 4.x 릴리스와 호환되고 LDAP를 통해 액세스할 수 있는 형식으로 기록할 수 있는 레트로 변경 로그 플러그인과 함께 제공됩니다. 레트로 변경 로그 플러그인은 기본적으로 사용되지 않으며 호환성을 위해 필요한 경우에만 사용해야 합니다. 자세한 내용은 *Sun ONE Directory Server Reference Manual*을 참조하십시오. 표 8-6에는 특정 속성에 대해 권장되는 추가 작업이 나와 있습니다.

**표 8-6** 역 변경 로깅에 대해 권장되는 조정 작업

구성 항목 DN 및 구성 속성	간략한 설명 및 권장되는 조정 작업
dn: cn=Retro Changelog Plugin,cn=plugins,cn=config nsslapd-changelogdir	레트로 변경 로그의 경로와 파일 이름을 지정합니다. 별도의 컨트롤러가 있는 자체 디스크나 디스크 하위 시스템에 레트로 변경 로그를 저장하는 것이 좋습니다. 큰 입출력 버퍼가 있으면 도움이 됩니다.

**표 8-6** 역 변경 로깅에 대해 권장되는 조정 작업 (계속)

구성 항목 DN 및 구성 속성	간략한 설명 및 권장되는 조정 작업
dn: cn=Retro Changelog Plugin,cn=plugins,cn=config nsslapd-changelogmaxage	레트로 변경 로그에 있는 항목의 최대 수명을 지정합니다.  속성 값 0(기본값, 제한 없음)을 레트로 변경 로그를 사용하는 클라이언트가 생성된 로그 항목을 처리하지 않고 그대로 둘 수 있는 기간으로 변경합니다.
dn: cn=Retro Changelog Plugin,cn=plugins,cn=config nsslapd-changelogmaxentries	레트로 변경 로그의 최대 항목 수를 지정합니다.  속성 값 0(기본값, 제한 없음)을 항목이 지워지지 않고 레트로 변경 로그에 보관되는 최대 항목 수로 변경합니다.

각 구성 속성에 대한 자세한 내용은 *Sun ONE Directory Server Reference Manual*을 참조하십시오.

## 트랜잭션 로깅

Directory Server는 트랜잭션 로깅을 통해 데이터베이스 무결성을 유지합니다. add, modify, delete 또는 modrdn과 같은 업데이트 작업을 받으면 Directory Server는 트랜잭션 로그에 이 작업에 대한 로그 메시지를 씁니다. 기본적으로 사용되는 영구 트랜잭션 로깅은 데이터 무결성을 보장하며, 이를 위해 업데이트 작업에 대한 결과 코드를 클라이언트 응용 프로그램에 반환하기 전에 각 업데이트 작업을 디스크에 있는 트랜잭션 로그로 완결합니다. 시스템에 장애가 발생하면 Directory Server는 트랜잭션 로그를 사용하여 데이터베이스를 복구합니다. 트랜잭션 로그는 비정상적으로 종료된 데이터베이스의 복구에 도움이 되므로 트랜잭션 로그와 디렉토리 데이터베이스를 별도의 디스크 하위 시스템에 저장하는 것이 좋습니다.

특히 영구성을 설정한 경우 트랜잭션 로깅은 디스크 용량을 많이 차지하기 때문에 업데이트 성능의 주된 병목이 될 수 있습니다. 트랜잭션 로그와 데이터베이스를 Sun StorEdge 디스크 어레이와 같은 별도의 RAID 시스템에 저장하면 시스템 장애 시 데이터 무결성 보호에 도움이 될 뿐만 아니라 업데이트 성능이 향상됩니다. 표 8-7에는 특정 속성에 대해 권장되는 추가 작업이 나와 있습니다.

**표 8-7** 트랜잭션 로깅에 대해 권장되는 조정 작업

구성 항목 DN 및 구성 속성	간략한 설명 및 권장되는 조정 작업
dn: cn=config,cn=ldbm database,cn=plugins,cn=config nsslapd-db-checkpoint-interval	Directory Server에서 트랜잭션 로그를 검사하는 빈도를 지정하고 전체 데이터베이스 시스템이 디스크와 동기화되게 하며 트랜잭션 로그를 정리합니다.  실제 테스트에 따라 데이터베이스 성능 최적화를 위해 다른 값을 설정해야 하는 경우가 아니면 속성 값을 60초(기본값)로 그대로 두십시오. 속성 값을 증가시키면 업데이트 성능은 향상되지만 비정상적인 종료 후의 복구 작업에 오랜 시간이 걸리기 때문에 트랜잭션 로그에서 더 많은 디스크 공간을 사용하게 됩니다.
dn: cn=config,cn=ldbm database,cn=plugins,cn=config nsslapd-db-durable-transaction	결과 코드를 클라이언트로 반환하기 전에 업데이트 작업을 디스크에 있는 트랜잭션 로그로 완결할지 여부를 지정합니다.  높은 수준의 데이터 무결성이 필요한 배포 시에는 속성 값을 on(기본값)으로 그대로 두십시오. 성능 향상을 위해 일부 배포 시에는 영구 트랜잭션 로깅을 비활성화할 수도 있습니다. 영구 트랜잭션 로깅을 사용하지 않을 경우 시스템에 장애가 발생하면 파일 시스템으로만 플러시되어 아직 디스크로 플러시되지 않은 로그 메시지는 손실될 수 있습니다. 따라서 영구 트랜잭션 로깅을 off로 설정하면 클라이언트가 업데이트 성공 결과 코드를 받은 후에도 업데이트를 복구할 수 없는 경우가 발생합니다.
dn: cn=config,cn=ldbm database,cn=plugins,cn=config nsslapd-db-logdirectory	트랜잭션 로그 파일의 경로와 파일 이름을 지정합니다.  별도의 컨트롤러가 있는 빠른 자체 디스크나 디스크 하위 시스템에 트랜잭션 로그를 저장하는 것이 좋습니다.

각 구성 속성에 대한 자세한 내용은 *Sun ONE Directory Server Reference Manual*을 참조하십시오.

## 기타 자원의 사용 관리

캐시 크기, 속성 값 색인화, 로그 관리의 최적화가 끝나면 Directory Server에서 클라이언트 응용 프로그램이 사용할 수 있는 자원을 제한하는 방법과 Directory Server의 시스템 자원 이용 방법을 조정할 수 있습니다. Directory Server 플러그 인으로 제공된 일부 기능을 다시 구성하거나 비활성화하는 것도 도움이 될 수 있습니다.

### 클라이언트에서 사용할 수 있는 자원 제한

기본 구성에서는 클라이언트 응용 프로그램이 실제 필요한 것보다 많은 Directory Server 자원을 사용할 수 있습니다. 이 때문에 클라이언트 응용 프로그램이 많은 연결을 연 다음 유휴 상태로 두고 사용하지 않거나 손실이 큰 색인화되지 않은 검색을 불필요하게 수행하거나 이전에 적합하지 않은 큰 속성 값을 디렉토리에 저장하는 등 실수나 고의로 자원을 남용하여 서버 성능이 저하될 수도 있습니다.

일부 배포 시에는 기본 구성을 수정하지 않는 것이 좋습니다. 이 절에 설명된 구성 속성 값을 변경하지 않는 배포 시에는 Sun ONE Directory Proxy Server 소프트웨어를 사용하여 외부적으로 제한을 설정하고 서비스 거부 공격을 차단하십시오.

일부 배포 시 Directory Server의 한 인스턴스는 메시징 서버와 같이 디렉토리 사용량이 많은 클라이언트 응용 프로그램 및 사용자 전자 우편 응용 프로그램과 같이 가끔 사용되는 디렉토리 클라이언트를 모두 지원해야 합니다. 이 경우 Sun ONE Directory Server 관리 설명서에 설명된 것처럼 바인드 DN 기반의 자원 제한을 사용하여 디렉토리 사용량이 많은 응용 프로그램의 제한을 높이는 것이 좋습니다.

표 9-1에는 모든 클라이언트 응용 프로그램에서 사용할 수 있는 자원 제한에 권장되는 설정이 나와 있습니다. 이 제한은 디렉토리 관리자 사용자에게는 적용되지 않으므로 클라이언트 응용 프로그램이 디렉토리 관리자 사용자로 연결하지 않도록 하십시오.

**표 9-1** 클라이언트에서 사용할 수 있는 자원 제한에 대해 권장되는 조정 작업

<b>구성 항목 DN 및 속성</b>	<b>간략한 설명 및 권장되는 조정 작업</b>
dn: cn=config nsslapd-idletimeout	<p>Directory Server에서 유휴 클라이언트 연결을 닫지 않고 기다리는 시간(초)을 설정합니다. 여기서 유휴상태란 연결은 열려 있지만 작업이 요청되지 않는 상태를 의미합니다. 기본적으로 시간 제한은 설정되지 않습니다.</p> <p>메시징 서버와 같은 일부 응용 프로그램은 트래픽이 작아서 유휴 상태로 남아 있지만 닫아서는 안 되는 연결 풀을 열 수 있습니다. 이 경우 응용 프로그램을 지원하는 복제본을 전달할 수 있다면 가장 바람직할 것입니다. 그럴 수 없으면 바인드 DN 기반의 제한을 사용합니다.</p> <p>어떤 경우든지 다른 응용 프로그램이 열려 두기를 원하는 연결을 닫지 않도록 충분히 높게, 하지만 불필요하게 연결을 유휴 상태로 두지 않도록 충분히 낮게 이 값을 설정하십시오. 테스트 최적화를 위한 시작점으로 120초(2분)를 사용하는 것이 좋습니다.</p>
dn: cn=config nsslapd-ioblocktimeout	<p>Directory Server가 정지된 클라이언트 연결을 닫지 않고 기다리는 시간(밀리초)을 설정합니다. 여기서 정지된상태란 서버가 클라이언트로 출력을 보내거나 클라이언트에서 입력을 읽지 못하도록 차단된 것을 의미합니다.</p> <p>특히 서비스 거부 공격에 노출된 Directory Server 인스턴스에서는 기본값인 1,800,000 밀리초(30분)보다 낮게 이 값을 설정하는 것이 좋습니다.</p>
dn: cn=config,cn=ldbm database,cn=plugins,cn=config nsslapd-lookthroughlimit	<p>검색 중에 확인되는 최대 후보 항목 수를 설정합니다.</p> <p>메시징 서버와 같은 일부 응용 프로그램은 전체 디렉토리를 검색해야 합니다. 이 경우 응용 프로그램을 지원하는 복제본을 전달할 수 있다면 가장 바람직할 것입니다. 그럴 수 없으면 바인드 DN 기반의 제한을 사용합니다.</p> <p>어떤 경우든지 기본값인 5000개 항목보다 낮게, nsslapd-sizelimit의 임계값보다는 높게 이 값을 설정하는 것이 좋습니다.</p>

**표 9-1** 클라이언트에서 사용할 수 있는 자원 제한에 대해 권장되는 조정 작업 (계속)

구성 항목 DN 및 속성	간략한 설명 및 권장되는 조정 작업
dn: cn=config nsslapd-maxbersize	들어오는 메시지의 최대 크기(바이트)를 설정합니다. Directory Server는 이 제한보다 큰 항목을 추가하는 요청을 거부합니다.  디렉토리 데이터의 최대 항목 크기를 정확하게 예측할 수 있으면 기본값인 2097152(2MB)를 예상되는 최대 디렉토리 항목 크기로 변경하십시오.
dn: cn=config nsslapd-maxthreadsperconn	클라이언트 연결 당 최대 스레드 수를 설정합니다.  메시징 서버와 같은 일부 응용 프로그램은 연결 풀을 연 다음 각 연결에서 많은 요청을 수행할 수 있습니다. 이 경우 응용 프로그램을 지원하는 복제본을 전달할 수 있다면 가장 바람직할 것입니다. 그럴 수 없으면 바인드 DN 기반의 제한을 사용합니다.  일부 응용 프로그램에서 연결 당 많은 요청을 수행할 것으로 예상되면 이 값을 증가시키되 기본값인 5보다 높게, 10보다는 낮게 유지하는 것이 좋습니다. 일반적으로 연결 당 10개 이상의 스레드를 지정하는 것은 바람직하지 않습니다.
dn: cn=config nsslapd-sizelimit	Directory Server에서 검색 요청에 응답하여 반환하는 최대 항목 수를 설정합니다.  메시징 서버와 같은 일부 응용 프로그램은 전체 디렉토리를 검색해야 합니다. 이 경우 응용 프로그램을 지원하는 복제본을 전달할 수 있다면 가장 바람직할 것입니다. 그럴 수 없으면 바인드 DN 기반의 제한을 사용합니다.  어떤 경우든지 기본값인 2000개 항목보다 낮게 이 값을 설정하는 것이 좋습니다.

**표 9-1** 클라이언트에서 사용할 수 있는 자원 제한에 대해 권장되는 조정 작업 (계속)

구성 항목 DN 및 속성	간략한 설명 및 권장되는 조정 작업
dn: cn=config nsslapd-timelimit	<p>Directory Server에서 검색 요청을 처리할 수 있는 최대 초 수를 설정합니다.</p> <p>메시징 서버와 같은 일부 응용 프로그램은 대량 검색을 수행해야 합니다. 이 경우 응용 프로그램을 지원하는 복제본을 전달할 수 있다면 가장 바람직할 것입니다. 그럴 수 없으면 바인드 DN 기반의 제한을 사용합니다.</p> <p>어떤 경우든지 이 값이 배포 요구 사항을 충족시킬 수 있도록 설정하되 최대한 낮게 유지하십시오. 대부분의 배포 시 기본값인 3600초(1시간)면 충분합니다. 테스트 최적화를 위한 시작점으로 600초(10분)를 사용하는 것이 좋습니다.</p>

각 구성 속성에 대한 자세한 내용은 *Sun ONE Directory Server Reference Manual*을 참조하십시오.

## 사용할 수 있는 시스템 자원 사용

배포 요구 사항에 따라 Directory Server 인스턴스가 시스템 및 네트워크 자원을 사용하는 방법, 액세스 제어 관리 방법 및 서버 플러그 인의 구성 방법을 조정할 수 있습니다. 표 9-2에는 시스템 자원에 권장되는 설정이 나와 있습니다.

**표 9-2** 시스템 자원 사용 구성에 권장되는 조정 작업

속성 (on dn: cn=config)	간략한 설명 및 권장되는 조정 작업
nsslapd-listenhost	<p>Directory Server에서 수신하는 IP 인터페이스의 호스트 이름을 설정하며 한 개의 값을 갖습니다.</p> <p>기본 동작은 모든 인터페이스에서 수신하는 것입니다. 중복 네트워크 인터페이스를 사용하는 대규모 배포에 맞게 기본 동작을 조정하여 가용성 및 처리량을 증가시킬 수 있습니다.</p> <p>다중 홈 시스템에 배포하거나 별도의 인터페이스를 통해 각 프로토콜을 지원하는 시스템에서 IPv4 또는 IPv6 트래픽만 수신할 때 이 값을 설정하는 것이 좋습니다. SSL을 사용하는 경우에는 <code>nsslapd-securelistenhost</code>를 설정하십시오.</p>



**표 9-2** 시스템 자원 사용 구성에 권장되는 조정 작업 (계속)

속성 (on dn: cn=config)	간략한 설명 및 권장되는 조정 작업
nsslapd-maxdescriptors	<p>Directory Server에서 사용하는 최대 파일 설명자 수를 설정합니다.</p> <p>Directory Server는 파일 설명자를 사용하여 클라이언트 연결을 처리하고 내부적으로 파일을 유지관리합니다. 사용할 수 있는 파일 설명자가 충분하지 않아 Directory Server에서 때때로 새 연결 수신을 중지한다는 오류 로그가 있으면 속성 값을 증가시켜 Directory Server에서 동시에 처리할 수 있는 클라이언트 연결 수를 늘릴 수 있습니다.</p> <p>103페이지의 "파일 설명자"에 설명된 것처럼 시스템에서 사용할 수 있는 파일 설명자 수를 증가시킨 경우 이에 따라 속성 값을 설정하십시오. 속성 값은 시스템에서 사용할 수 있는 최대 파일 설명자 수보다 작거나 같아야 합니다.</p>
nsslapd-nagle	<p>소켓 수준에서 TCP 패킷 전송을 지연할지 여부를 설정합니다.</p> <p>클라이언트 응용 프로그램에 결과를 보낼 때 프로토콜 수준의 지연을 방지하려면 속성 값을 off(기본값)로 그대로 두십시오.</p>

**표 9-2** 시스템 자원 사용 구성에 권장되는 조정 작업 (계속)

속성 (on dn: cn=config)	간략한 설명 및 권장되는 조정 작업
nsslapd-reservedescriptors	<p>Directory Server에서 색인화, 복제 및 기타 내부 처리를 관리하기 위해 유지관리하는 파일 설명자 수를 설정합니다. Directory Server는 클라이언트 연결을 처리할 때는 파일 설명자를 사용하지 않습니다.</p> <p>다음과 같은 조건에 모두 부합되면 기본값인 64보다 높게 이 값을 설정하는 것이 좋습니다.</p> <ul style="list-style-type: none"> <li>• Directory Server가 10개 이상의 소비자에 복제되거나 Directory Server에서 30개 이상의 색인 파일을 유지관리합니다.</li> <li>• Directory Server에서 다수의 클라이언트 연결을 처리합니다.</li> <li>• Directory Server에 클라이언트 연결과 관련되지 않은 작업을 위한 파일 설명자가 부족하다는 오류 로그 메시지가 있습니다.</li> </ul> <p>예약된 파일 설명자 수가 증가하면 클라이언트 연결 처리에 사용할 수 있는 파일 설명자 수는 감소합니다. 속성 값을 증가시키는 경우 시스템에서 사용할 수 있는 파일 설명자 수와 nsslapd-maxdescriptors 값도 증가시키는 것이 좋습니다.</p> <p>예약할 파일 설명자 개수의 첫 추정 값에서 이 속성을 변경하려면 아래 수식에 따라 nsslapd-reservedescriptors 값을 설정합니다.</p> $20 + 4 * (\text{데이터베이스 수}) + (\text{총 색인 수}) + (\text{nsoperationconnectionslimit 값}) * (\text{연결 백엔드 수}) + \text{ReplDescriptors} + \text{PTADescriptors} + \text{SSLDescriptors}$ <p>여기서 복제를 사용하는 경우 <i>ReplDescriptors</i> = 공급자 복제본 수 + 8, PTA(Pass Through Authentication) 플러그인을 사용하는 경우 <i>PTADescriptors</i> = 3(그렇지 않으면 0), SSL을 사용하는 경우 <i>SSLDescriptors</i> = 5(그렇지 않으면 0)가 됩니다.</p> <p>인스턴스가 접미사 당 한 개의 데이터베이스를 사용하도록 구성되어 있으면 데이터베이스 수는 인스턴스의 접미사 수와 같습니다. 실제 테스트를 통해 추정 값을 확인합니다.</p>
nsslapd-securelistenhost	<p>Directory Server에서 SSL 연결을 수신하는 IP 인터페이스의 호스트 이름을 설정하며 한 개의 값을 갖습니다.</p> <p>기본 동작은 모든 인터페이스에서 수신하는 것입니다.</p> <p>nsslapd-listenhost와 같은 방식으로 속성을 사용하십시오.</p>

**표 9-2** 시스템 자원 사용 구성에 권장되는 조정 작업 (계속)

속성 (on dn: cn=config)	간략한 설명 및 권장되는 조정 작업
nsslapd-threadnumber	<p>Directory Server에서 사용하는 스레드 수를 설정합니다.</p> <p>다음 중 부합되는 조건이 있으면 속성 값을 조정하는 것이 좋습니다.</p> <ul style="list-style-type: none"> <li>클라이언트 응용 프로그램에서 업데이트나 복잡한 검색과 같은 시간 소모적인 작업을 동시에 많이 수행합니다.</li> <li>Directory Server에서 다수의 동시 클라이언트 연결을 지원합니다.</li> <li>Directory Server에서 5,000,000개 이상의 항목을 처리합니다.</li> </ul> <p>다중 프로세서 시스템은 단일 프로세서 시스템보다 큰 스레드 풀을 유지할 수 있습니다. 속성 값을 최적화하는 경우 첫 추정 값으로 프로세서 수의 두 배나 20개 이상의 동시 업데이트 수를 사용하십시오. 표 9-1에 설명된 것처럼 클라이언트 연결 당 최대 스레드 수인 nsslapd-maxthreadsperconn도 조정하는 것이 좋습니다. 클라이언트 연결을 처리하는 최대 스레드 수는 시스템에서 사용할 수 있는 최대 파일 설명자 수를 초과할 수 없습니다. 어떤 경우에는 이 속성 값을 늘리는 것보다 줄이는 것이 도움이 됩니다.</p> <p>실제 테스트를 통해 추정 값을 확인합니다. 결과는 특정 배포 상황뿐만 아니라 기본 시스템에 따라 다를 수 있습니다.</p>

각 구성 속성에 대한 자세한 내용은 *Sun ONE Directory Server Reference Manual*을 참조하십시오.

## 액세스 제어 관리

Directory Server는 우수한 메모리 관리와 매크로 ACI 지원 등 ACI(Access Control Instructions)에 대해 향상된 성능과 확장성을 제공합니다. 하지만 Directory Server에서 복잡한 ACI를 평가하려면 상당한 시스템 자원이 사용되므로 복잡한 ACI를 많이 사용하면 성능이 저하될 수 있습니다.

매크로 ACI는 사용되는 ACI 수를 제한하는 데 도움이 됩니다. ACI 수를 제한하면 보다 간편하게 액세스 제어를 관리하고 시스템 로드를 줄일 수 있습니다. 매크로는 ACI의 DN 또는 DN의 일부를 나타내는 자리 표시자로, ACI 대상이나 ACI 바인드 규칙 또는 둘 모두에서 사용할 수 있습니다. Directory Server는 수신된 요청 처리에 필요한 자원과 일치하는 ACI 매크로를 확인합니다. 일치하는 매크로를 찾으면 Directory Server는 이 매크로를 실제 DN 값으로 바꿉니다. 그런 다음 Directory Server는 정상적으로 ACI를 평가합니다. ACI에 대한 자세한 내용은 *Sun ONE Directory Server 관리 설명서*를 참조하십시오.

테스트 결과에 따르면 Directory Server는 50,000개 이상의 ACI를 지원할 수 있습니다. 다양한 배포 시나리오에서 성능에 미치는 영향에 대해서는 현재 분석 중에 있습니다. 성능 저하를 방지하고 액세스 제어 관리를 간소화하려면 ACI 수를 최대한 작게 유지하십시오. 복잡한 ACI 환경이 필요한 배포 시에는 Sun ONE Directory Proxy Server를 사용하여 액세스 제어 기능을 제공하는 것이 좋습니다.

## 서버 플러그인 구성

Directory Server는 플러그인을 사용하여 액세스 제어, 복제, 구문 검사, 속성의 고유성과 같은 다양한 주요 기능을 구현합니다. 특정 배포 시에는 일부 플러그인을 다시 구성하는 것이 도움이 될 수도 있습니다. 표 9-3에는 표준 플러그인에 권장되는 설정이 나와 있습니다.

**표 9-3** 표준 플러그인에 권장되는 조정 작업

이름 및 DN	간략한 설명 및 권장되는 조정 작업
7비트 검사 플러그인 dn: cn=7-bit check,cn=plugins,cn=config	Directory Server에서 속성 값이 7비트 클린인지 검사할 수 있습니다. 즉, 7비트 인코딩에 맞는 문자만 해당 속성 값에 포함됩니다.  인프라가 일본어와 같은 광범위한 인코딩을 지원하도록 설계되었으면 이 플러그인을 비활성화할 수도 있습니다(기본값은 on).
레거시 복제 플러그인 dn: cn=Legacy Replication Plugin,cn=plugins,cn=config	Directory Server에서 4.x 공급자의 소비자 기능을 할 수 있습니다.  예를 들어, 업그레이드 중에 Directory Server를 4.x 공급자의 소비자 사용하지 않으려면 이 플러그인을 off로 설정합니다(4.x 복제 기능이 필요한 경우 기본값은 on).

**표 9-3** 표준 플러그 인에 권장되는 조정 작업 (계속)

이름 및 DN	간략한 설명 및 권장되는 조정 작업
<p>참조 무결성 플러그 인</p> <p>dn: cn=referential integrity postoperation,cn=plugins,cn=config</p>	<p>Directory Server에서 관련 항목 간의 관계를 유지관리할 수 있습니다. 예를 들어, 사용자 항목을 디렉토리에서 제거하거나 이름을 바꾸면 수동으로 작업하지 않아도 필요에 따라 해당 사용자가 소속된 그룹이 업데이트됩니다.</p> <p>모든 마스터에서 이 플러그 인을 활성화 및 구성합니다.</p> <p>플러그 인을 사용하는 경우 이 플러그 인에 구성된 모든 속성의 동일 색인을 작성합니다. 플러그 인은 업데이트할 항목을 검색할 때 이러한 색인을 사용합니다. 사용되는 속성에 동일 색인이 없으면 플러그 인은 손실이 큰 색인화되지 않은 검색을 수행해야 하므로 성능이 저하됩니다.</p> <p>플러그 인 활성화 및 구성에 대한 자세한 내용은 <i>Sun ONE Directory Server 관리 설명서</i>를 참조하십시오.</p>

각 구성 속성에 대한 자세한 내용은 *Sun ONE Directory Server Reference Manual*을 참조하십시오.

사용할 수 있는 시스템 자원 사용

## 설치된 제품 레이아웃

이 부록에는 일반 설치 후의 제품 소프트웨어 레이아웃이 요약되어 있습니다. 설치된 파일 중에서 여기에 명시되고 제품 설명서에 언급된 파일만 지원되는 공개 제품 인터페이스에 포함됩니다.

---

**주** 여기서 소개된 예는 Solaris Operating Environment용 제품 설치를 반영합니다. 다른 플랫폼에 설치하는 경우 파일 이름과 확장명이 다를 수 있습니다.

Solaris 패키지 버전의 제품을 설치한 후 pkgchk(1M) 유틸리티인 pkgchk -v *package-name*을 사용하여 특정 패키지의 전체 설치 경로 이름을 확인할 수도 있습니다.

---

Solaris Operating Environment와 같은 일부 플랫폼에서는 디렉토리 서비스 관리를 위한 통합 도구를 제공합니다. Sun ONE Directory Server도 이러한 도구를 제공합니다. 여기서 설명된 도구에 대한 자세한 내용은 *Sun ONE Directory Server 관리 설명서* 및 *Sun ONE Directory Server Reference Manual*을 참조하십시오.

## ServerRoot 디렉토리

ServerRoot 디렉토리에는 다양한 시스템 관리 유틸리티가 포함되어 있습니다. 사용하는 플랫폼, 구성 및 설치에서 ServerRoot 디렉토리에 해당하는 경로를 확인하려면 10페이지의 "기본 경로 및 파일 이름"을 참조하십시오.

**표 A-1** ServerRoot에 있는 유틸리티

유틸리티	설명
ServerRoot/restart-admin	관리 서버를 다시 시작합니다.

**표 A-1** ServerRoot에 있는 유틸리티 (계속)

유틸리티	설명
<i>ServerRoot</i> /start-admin	관리 서버를 시작합니다.
<i>ServerRoot</i> /startconsole	Sun ONE 서버 콘솔을 시작합니다.
<i>ServerRoot</i> /stop-admin	관리 서버를 중지합니다.
<i>ServerRoot</i> /uninstall	제품 소프트웨어를 제거합니다.

*ServerRoot*/bin 디렉토리에는 서버 인스턴스를 작성할 때 내부적으로 사용되는 제품 이진과 구성 템플릿이 포함되어 있습니다.

**표 A-2** ServerRoot/bin에 있는 파일

파일	설명
<i>ServerRoot</i> /bin/	내부용. 다음과 같은 경우는 제외됩니다.
<i>ServerRoot</i> /bin/admin/admconfig	관리 서버를 구성합니다.
<i>ServerRoot</i> /bin/https/bin/ns-httpd	Sun ONE 관리 서버
<i>ServerRoot</i> /bin/https/bin/uxwdog	관리 서버 워치독(watchdog)
<i>ServerRoot</i> /bin/slaped/server/ns-ldapagt	LDAP 기반의 SNMP 하위 에이전트
<i>ServerRoot</i> /bin/slaped/server/ns-slaped	Sun ONE Directory Server

*ServerRoot*/lib 디렉토리에는 플러그 인을 비롯한 제품 라이브러리가 포함되어 있습니다.

**표 A-3** ServerRoot/lib에 있는 라이브러리

라이브러리	설명
<i>ServerRoot</i> /lib/	내부용 플러그 인
<i>ServerRoot</i> /lib/libnspr4.so	NSPR, 버전 4.x

*ServerRoot*/manual 디렉토리에는 콘솔 온라인 도움말 파일이 포함되어 있습니다.

**표 A-4** ServerRoot/manual에 있는 온라인 도움말 파일

디렉토리	설명
<i>ServerRoot</i> /manual/	온라인 도움말 지원



*ServerRoot/plugins* 디렉토리에는 샘플 서버 플러그인, 플러그인 배포용 헤더 파일, SNMP 지원용 플러그인이 포함되어 있습니다.

**표 A-5** *ServerRoot/plugins*에 있는 플러그인 파일

디렉토리 또는 파일	설명
<i>ServerRoot/plugins/</i>	샘플, 헤더, SNMP 지원
<i>ServerRoot/plugins/slapd/slapi/examples/</i>	샘플 플러그인
<i>ServerRoot/plugins/slapd/slapi/include/</i>	플러그인 헤더 파일
<i>ServerRoot/plugins/snmp/magt/magt</i>	관리 에이전트를 구성합니다.
<i>ServerRoot/plugins/snmp/mibs/</i>	SNMP MIB
<i>ServerRoot/plugins/snmp/sagt/sagt</i>	SNMP 에이전트를 구성합니다.

*ServerRoot/shared/bin* 디렉토리에는 서버 관리 도구가 포함되어 있습니다.

**표 A-6** *ServerRoot/shared/bin*에 있는 도구 및 클라이언트

디렉토리 또는 파일	설명
<i>ServerRoot/shared/bin</i>	내부용. 다음과 같은 경우는 제외됩니다.
<i>ServerRoot/shared/bin/admin_ip.pl</i>	IP 주소를 변경합니다.
<i>ServerRoot/shared/bin/entrycmp</i>	복제할 항목을 비교합니다.
<i>ServerRoot/shared/bin/fildif</i>	필터링된 LDIF를 덤프합니다.
<i>ServerRoot/shared/bin/insync</i>	복제 동기화를 확인합니다.
<i>ServerRoot/shared/bin/ldapcompare</i>	속성 값을 비교합니다.
<i>ServerRoot/shared/bin/ldapdelete</i>	디렉토리 항목을 삭제합니다.
<i>ServerRoot/shared/bin/ldapmodify</i>	디렉토리 항목을 수정합니다.
<i>ServerRoot/shared/bin/ldapsearch</i>	디렉토리 항목을 찾습니다.
<i>ServerRoot/shared/bin/modutil</i>	PKCS #11 모듈을 관리합니다.
<i>ServerRoot/shared/bin/uconv</i>	ISO를 UTF-8로 변환합니다.
<i>ServerRoot/shared/bin/repldisc</i>	복제 토폴로지를 찾습니다.

*ServerRoot/shared/config* 디렉토리에는 인증서를 디렉토리 항목으로 매핑하기 위한 구성 파일이 포함되어 있습니다.

**표 A-7** *ServerRoot*/shared/config에 있는 인증서 매핑 구성 파일

디렉토리 또는 파일	설명
<i>ServerRoot</i> /shared/config	내부용. 다음과 같은 경우는 제외됩니다.
<i>ServerRoot</i> /shared/config/certmap.conf	인증서를 항목으로 매핑합니다.

*ServerRoot*/setup5 디렉토리에는 자동 설치 및 제거를 위한 샘플 템플릿이 포함되어 있습니다.

**표 A-8** *ServerRoot*/setup5에 있는 자동 설치 및 제거 템플릿

디렉토리 또는 파일	설명
<i>ServerRoot</i> /setup5	내부용. 다음과 같은 경우는 제외됩니다.
<i>ServerRoot</i> /setup5/typical.ins	자동 설치 템플릿 파일
<i>ServerRoot</i> /setup5/uninstall.ins	자동 제거 템플릿 파일

## 서버 인스턴스 디렉토리

*slapd-ServerID* 디렉토리에는 서버 인스턴스 *ServerID*에 해당하는 파일이 포함되어 있습니다. *ServerRoot*/*slapd-ServerID* 디렉토리에는 명령줄 관리를 위한 다양한 스크립트가 포함되어 있습니다.

**표 A-9** 서버 인스턴스 스크립트

스크립트	설명
<i>ServerRoot</i> / <i>slapd-ServerID</i> /	서버 인스턴스
<i>ServerRoot</i> / <i>slapd-ServerID</i> /bak2db	데이터베이스를 복원합니다(오프라인).
<i>ServerRoot</i> / <i>slapd-ServerID</i> /bak2db.pl	데이터베이스를 복원합니다(온라인).
<i>ServerRoot</i> / <i>slapd-ServerID</i> /db2bak	데이터베이스를 백업합니다(오프라인).
<i>ServerRoot</i> / <i>slapd-ServerID</i> /db2bak.pl	데이터베이스를 백업합니다(온라인).
<i>ServerRoot</i> / <i>slapd-ServerID</i> /db2index.pl	색인을 생성합니다(온라인).
<i>ServerRoot</i> / <i>slapd-ServerID</i> /db2ldif	데이터베이스를 LDIF로 덤프합니다(오프라인)

**표 A-9** 서버 인스턴스 스크립트 (계속)

스크립트	설명
<i>ServerRoot</i> / <i>slapd-ServerID</i> / <i>db2ldif.pl</i>	데이터베이스를 LDIF로 덤프합니다 (온라인)
<i>ServerRoot</i> / <i>slapd-ServerID</i> / <i>getpwenc</i>	암호화된 암호를 인쇄합니다.
<i>ServerRoot</i> / <i>slapd-ServerID</i> / <i>ldif2db</i>	LDIF를 가져옵니다(오프라인).
<i>ServerRoot</i> / <i>slapd-ServerID</i> / <i>ldif2db.pl</i>	LDIF를 가져옵니다(온라인).
<i>ServerRoot</i> / <i>slapd-ServerID</i> / <i>ldif2ldap</i>	LDAP를 통해 LDIF를 가져옵니다.
<i>ServerRoot</i> / <i>slapd-ServerID</i> / <i>monitor</i>	모니터 정보를 검색합니다.
<i>ServerRoot</i> / <i>slapd-ServerID</i> / <i>ns-accountstatus.pl</i>	계정 상태를 작성합니다.
<i>ServerRoot</i> / <i>slapd-ServerID</i> / <i>ns-activate.pl</i>	항목을 활성화합니다.
<i>ServerRoot</i> / <i>slapd-ServerID</i> / <i>ns-inactivate.pl</i>	항목을 비활성화합니다.
<i>ServerRoot</i> / <i>slapd-ServerID</i> / <i>restart-slapd</i>	Directory Server를 다시 시작합니다.
<i>ServerRoot</i> / <i>slapd-ServerID</i> / <i>restoreconfig</i>	관리 서버 구성을 복원합니다.
<i>ServerRoot</i> / <i>slapd-ServerID</i> / <i>saveconfig</i>	관리 서버 구성을 저장합니다.
<i>ServerRoot</i> / <i>slapd-ServerID</i> / <i>start-slapd</i>	Directory Server를 시작합니다.
<i>ServerRoot</i> / <i>slapd-ServerID</i> / <i>stop-slapd</i>	Directory Server를 중지합니다.
<i>ServerRoot</i> / <i>slapd-ServerID</i> / <i>suffix2instance</i>	접미사를 백엔드로 매핑합니다.
<i>ServerRoot</i> / <i>slapd-ServerID</i> / <i>vlvindex</i>	가상 목록 보기 색인을 작성합니다.

*ServerRoot* / *slapd-ServerID*의 하위 디렉토리에는 구성, 로그 및 백업 데이터가 포함되어 있습니다.

**표 A-10** 서버 인스턴스 하위 디렉토리

디렉토리	설명
<i>ServerRoot</i> / <i>slapd-ServerID</i> /	서버 인스턴스
<i>ServerRoot</i> / <i>slapd-ServerID</i> / <i>bak</i> /	디렉토리 데이터베이스 백업
<i>ServerRoot</i> / <i>slapd-ServerID</i> / <i>confbak</i> /	관리 서버 구성 백업

**표 A-10** 서버 인스턴스 하위 디렉토리 (계속)

디렉토리	설명
<i>ServerRoot</i> / slapd- <i>ServerID</i> / conf_bk /	Directory Server 구성 백업
<i>ServerRoot</i> / slapd- <i>ServerID</i> / config /	Directory Server 구성
<i>ServerRoot</i> / slapd- <i>ServerID</i> / config / schema /	디렉토리 스키마 구성
<i>ServerRoot</i> / slapd- <i>ServerID</i> / db /	디렉토리 데이터베이스
<i>ServerRoot</i> / slapd- <i>ServerID</i> / ldif /	샘플 LDIF 파일
<i>ServerRoot</i> / slapd- <i>ServerID</i> / locks /	런타임 프로세스 잠금
<i>ServerRoot</i> / slapd- <i>ServerID</i> / logs /	서버 로그 파일
<i>ServerRoot</i> / slapd- <i>ServerID</i> / tmp /	런타임 임시 파일

제공된 도구를 사용하여 서버 인스턴스를 관리합니다. 수동으로 디렉토리 내용을 수정하지 마십시오.

## 내부용으로만 사용

다음과 같은 내용은 Directory Server에서 내부용으로 사용됩니다. 이러한 내부 구성 요소는 지원되는 공개 인터페이스에 포함되지 않습니다.

- *ServerRoot* / adminacl /
- *ServerRoot* / admin-serv /
- *ServerRoot* / admserv
- *ServerRoot* / alias /
- *ServerRoot* / dist /
- *ServerRoot* / httpacl /
- *ServerRoot* / include /
- *ServerRoot* / install /
- *ServerRoot* / java /
- *ServerRoot* / userdb /

이 디렉토리나 해당 내용은 수정하지 마십시오.

# Sun Crypto Accelerator 보드 사용

이 부록에서는 Directory Server와 함께 Sun Crypto Accelerator 보드를 사용하여 인증서 기반의 인증과 SSL(Secure Sockets Layer) 프로토콜을 사용한 연결의 성능 향상 방법에 대해 설명합니다.

## 시작 전 주의 사항

표 B-1에는 Sun Crypto Accelerator 보드를 사용하여 SSL 연결 성능을 향상시키기 위해 먼저 완료해야 하는 항목이 나와 있습니다.

**표 B-1** 보드 사용을 위한 필수 사항

필수 사항	설명
보드 설치	호스트에 하드웨어, 드라이버, 패치 및 관리 유틸리티를 설치하는 경우 보드용으로 제공된 제품 설명서를 참조하십시오.
Directory Server 설치	자세한 내용은 1장, "Sun ONE Directory Server 설치"를 참조하십시오.
서버 인증서(PKCS#12 형식)	Directory Server에 대한 서버 인증서를 .p12 파일로 받습니다.
CA 인증서(PEM 형식)	인증 기관(CA)에 대한 CA 인증서를 PEM(Privacy Enhanced Mail) 형식의 파일로 받습니다.

SSL 프로토콜과 SSL 인증서에 대한 자세한 설명과 Sun ONE 서버 콘솔 관리를 지원하는 Sun ONE 서버와 함께 이 프로토콜을 사용하는 방법은 *Sun ONE Server Console Server Management Guide*를 참조하십시오.

## 토큰 작성

Directory Server는 토큰과 암호를 사용하여 가속 보드에 있는 해당 암호화 키 자료에 액세스합니다. 토큰은 *user@realm* 형식을 사용하며, 여기서 *user*는 가속 보드의 사용자로, 암호화 키 생성 자료의 소유자를 나타내고 *realm*은 가속 보드의 영역으로, 사용자 및 해당 키 생성 자료의 논리적 분할 영역입니다. 가속 보드 *user*는 시스템의 사용자 계정과 관련이 없으며 보드에만 적용됩니다. 사용자 및 영역에 대한 자세한 내용은 가속 보드 제품 설명서를 참조하십시오.

보드용으로 제공된 *secadm(1M)* 유틸리티를 사용하여 토큰 사용자와 영역을 작성할 수 있습니다. 가속 보드에서는 다수의 *slots*를 작성하여 여러 응용 프로그램에 대한 토큰을 관리할 수도 있습니다. 여기서는 성능을 위해 호스트를 Directory Server 전용으로 설정하여 기본값인 한 개의 슬롯을 사용한다고 가정합니다. 여러 소프트웨어 응용 프로그램에 대해 보드를 사용하는 방법은 가속 보드 제품 설명서를 참조하십시오.

기본 슬롯에 액세스할 토큰 사용자와 영역을 작성하려면 다음 단계를 수행합니다.

1. 아래 명령을 실행하여 *secadm* 유틸리티를 시작합니다.

```
$ CryptoPath/bin/secadm
```

기본 *CryptoPath*는 */opt/SUNWconn/crypto*입니다.

2. 토큰 영역을 작성합니다.

```
secadm> create realm=dsrealm
System Administrator Login Required
Login: super-user
Password:
Realm dsrealm created successfully.
```

3. 작성하려는 사용자가 포함될 영역을 설정합니다.

```
secadm> set realm=dsrealm
secadm{dsrealm}> su
System Administrator Login Required
Login: super-user
Password:
secadm{root@dsrealm}#
```

4. SSL이 구성된 Directory Server를 다시 시작할 때 사용되는 암호를 입력하여 기본 슬롯을 사용할 nobody 사용자를 작성합니다.

```
secadm{root@dsrealm}# create user=nobody
Initial password: password
Confirm password: password
User nobody created successfully.
secadm{root@dsrealm}# exit
```

이제 nobody@dsrealm 토큰의 사용자와 영역이 작성되었으며 Directory Server를 다시 시작할 때 사용되는 암호를 입력했습니다.

## 보드 바인드 생성

가속 보드의 바인드는 Directory Server가 보드에 바인드할 수 있도록 생성하는 외부 보안 모듈의 형식을 사용합니다. 다양한 SSL 알고리즘을 지원하는 Directory Server 인증서 데이터베이스와 외부 보안 모듈 간의 바인드를 생성하려면 다음 단계를 수행합니다.

1. 아래 명령을 실행하여 modutil을 사용하기 전에 먼저 LD\_LIBRARY\_PATH를 설정합니다.

```
$ set LD_LIBRARY_PATH=ServerRoot/lib ; export LD_LIBRARY_PATH
```

2. 보안 모듈 데이터베이스가 없으면 아래 명령을 실행하여 작성합니다.

```
$ cd ServerRoot/shared/bin
$ ./modutil -create -dbdir ../../alias -dbprefix "slapd-serverID"
```

3. 아래 명령을 실행하여 외부 보안 모듈을 보안 모듈 데이터베이스에 추가합니다.

```
$ ./modutil -add "Crypto Mod" -dbdir ../../alias -nocertdb \
-libfile CryptoPath/lib/libpkcs11.so \
-mechanisms "RSA:DSA:RC4:DES" -dbprefix "slapd-serverID"
```

기본 *CryptoPath*는 /opt/SUNWconn/crypto입니다.

4. 아래 명령을 실행하여 보안 모듈을 표시하고 모듈이 제대로 추가되었는지 확인합니다.

```
$ ./modutil -list -dbdir ../../alias -dbprefix "slapd-serverID"
```

단계 3에서 추가한 Crypto Mod 항목이 표시되어야 합니다.

5. 아래 명령을 실행하여 이 외부 보안 모듈을 RSA, DSA, RC4 및 DES의 기본값으로 설정합니다.

```
$ ./modutil -default "Crypto Mod" -dbdir ../../alias \
-mechanisms "RSA:DSA:RC4:DES" -dbprefix "slapd-serverID"
```

이렇게 하면 기본 보안 모듈이 제대로 변경됩니다.

이제 가속 보드에 대한 바인드를 생성했으며 인증서를 가져올 수 있습니다.

## 인증서 가져오기

SSL을 구성하기 전에 먼저 165페이지의 표 B-1에 설명된 형식으로 받은 서버 인증서와 CA 인증서를 가져와야 합니다. 인증서를 가져오려면 다음 단계를 수행합니다.

1. .p12 형식의 서버 인증서 파일을 가져옵니다.

```
$ cd ServerRoot/shared/bin
$ ./pk12util -i ServerCert.p12 -d ../../alias -P "slapd-serverID" \
-h "nobody@dsrealm"
Enter Password or Pin for "nobody@dsrealm": password
Enter Password for PKCS12 file: password
```

2. 아래 명령을 실행하여 CA 인증서를 가져옵니다.

```
$ ./certutil -A -n "Crypto CA Cert" -t CT -i CACert.txt \
-d ../../alias -P "slapd-serverID" -h "nobody@dsrealm"
```

3. 아래 명령을 실행하여 토큰과 관련된 인증서를 표시하고 제대로 인증서를 가져왔는지 확인합니다.

```
$ ./certutil -L -d ../../alias -P "slapd-serverID" \
-h "nobody@dsrealm"
```

단계 1 및 단계 2에서 추가한 인증서 항목이 표시되어야 합니다.

이제 인증서를 가져왔으며 Directory Server에서 SSL 연결을 수신하도록 구성할 수 있습니다.

## SSL 구성

작성한 토큰과 암호, 외부 보안 모듈과 Directory Server 인증서 데이터베이스 간에 생성한 바인드, 그리고 가져온 인증서를 사용하여 Directory Server가 보안 모드로 시작되도록 구성할 수 있습니다. SSL을 구성하고 Directory Server를 보안 모드로 다시 시작하려면 다음 단계를 수행합니다.



1. SSL 관련 Directory Server 구성 항목을 변경하기 위한 수정 작업이 요약된 `ssl.ldif` 파일을 작성합니다.

**코드 예제 B-1**보드를 사용하여 SSL을 활성화하기 위한 수정 작업(`ssl.ldif`)

```

dn: cn=RSA,cn=encryption,cn=config
changetype: add
objectclass: top
objectclass: nsEncryptionModule
cn: RSA
nsSSLToken: nobody@dsrealm
nsSSLPersonalitySSL: ServerCertNickname1
nsSSLActivation: on

dn: cn=encryption,cn=config
changetype: modify
replace: nsSSL3
nsSSL3: on
-
replace: nsSSLClientAuth
nsSSLClientAuth: allowed
-
replace: nsSSL3Ciphers
nsSSL3Ciphers: -rsa_null_md5,+rsa_rc4_128_md5,+rsa_rc4_40_md5,
+rsa_rc2_40_md5,+rsa_des_sha,+rsa_fips_des_sha,+rsa_3des_sha,
+rsa_fips_3des_sha,+fortezza,+fortezza_rc4_128_sha,
+fortezza_null,+tls_rsa_export1024_with_rc4_56_sha,
+tls_rsa_export1024_with_rc4_56_sha,
+tls_rsa_export1024_with_des_cbc_sha
-
replace: nsCertfile
nsCertfile: alias/slaped-serverID-cert7.db
-
replace: nsKeyFile
nsKeyFile: alias/slaped-serverID-key3.db

dn: cn=config
changetype: modify
replace: nsslapd-secureport
nsslapd-secureport: port
-
replace: nsslapd-security
nsslapd-security: on

```

1. 이 별칭은 Directory Server 인증서에 포함되어 있습니다.

여기서 `nsslapd-secureport` 값인 `port`는 보안 모드로 시작된 Directory Server에서 SSL 연결을 수신하는 포트입니다.

2. 아래 명령으로 수정 작업을 적용하여 Directory Server 구성을 변경합니다.

```
$ ldapmodify -p currPort -D "cn=directory manager" -w password -f ssl.ldif
```

여기서 *currPort*는 Directory Server에서 현재 클라이언트 요청을 수신하는 포트 번호입니다.

**3. Directory Server를 보안 모드로 다시 시작합니다.**

```
$ ServerRoot/slapd-serverID/restart-slapd
Enter PIN for nobody@dsrealm: password
```

여기서 *password*는 nobody@dsrealm 토큰을 작성할 때 입력한 nobody 사용자 암호입니다.

이제 Directory Server는 지정한 포트에서 SSL 트래픽을 수신합니다. 이 포트에서 SSL을 통해 Directory Server에 액세스하도록 Sun ONE 관리 서버와 클라이언트 응용 프로그램을 구성할 수 있습니다. 자세한 내용은 *Sun ONE Directory Server 관리 설명서*를 참조하십시오.

# Directory Server용 Sun Cluster HA 설치

이 부록에서는 Directory Server 데이터 서비스용 Sun Cluster HA 및 연결된 관리 서버 데이터 서비스를 설치하고 구성하는 방법에 대해 설명합니다. Sun Cluster 설치 지침 및 주요 개념에 대해서는 Sun Cluster 3.0 제품 설명서를 참조하십시오.

데이터 서비스는 페일오버 서비스로 구성해야 합니다.

## 시작 전 주의 사항

소프트웨어를 설치하고 구성하기 전에 이 절의 정보와 *Sun Cluster 3.0 Release Notes*의 워크시트를 검사 목록으로 사용합니다.

설치를 시작하기 전에 다음과 같은 질문을 가정해 보십시오.

- 같은 노드에서 여러 개의 Directory Server 인스턴스를 실행할 계획인가?

그렇다면 해당 네트워크 자원에 대한 `cn=config`의

`nsslapd-listenhost`(`dirserv.example.com`과 같은 논리적 호스트 이름)를 각 인스턴스의 IP 주소로 설정할 수 있습니다. Directory Server의 기본 동작은 모든 네트워크 인터페이스에서 수신하는 것입니다.

- Sun Cluster 구성에서 여러 데이터 서비스를 실행하는가?

순서에 상관없이 여러 개의 데이터 서비스를 설정할 수 있습니다. 단, DNS용 Sun Cluster HA는 Directory Server용 Sun Cluster HA보다 먼저 설정해야 합니다.

표 C-1에는 Directory Server용 Sun Cluster HA 설치 및 구성 프로세스가 요약되어 있습니다.

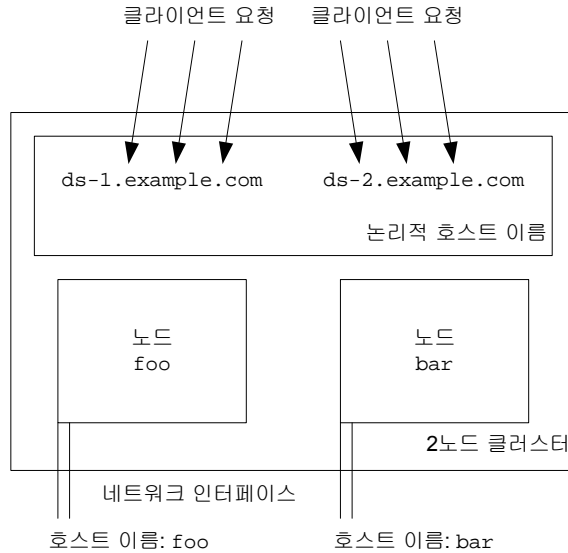
**표 C-1**      설치 및 구성 프로세스

작업	확인해야 할 사항
173페이지의 "네트워크 자원 설정"	<p>데이터 서비스의 마스터로 사용할 수 있는 클러스터 노드 이름</p> <p>Directory Server에 액세스하는 클라이언트가 사용할 논리적 호스트 이름(예: <code>ds1.example.com</code>, <code>ds2.example.com</code>)</p> <p>논리적 호스트 이름 설정 방법은 Sun Cluster 3.0 제품 설명서를 참조하십시오.</p>
175페이지의 "서버 설치"	<p>전역 파일 시스템에서 Directory Server를 설치하는 <i>ServerRoot</i> 위치(예: <code>/global/ds</code>)</p> <p>21페이지의 표 1-2에 요약된 설치 세부 정보</p>
176페이지의 "데이터 서비스 패키지 설치"	<p>SUNWdsha 패키지와 SUNWasha 패키지는 클러스터의 다른 데이터 서비스와 동일한 도구를 사용하여 Directory Server와 관리 서버를 관리할 수 있도록 데이터 서비스에 대한 관리 인터페이스를 제공합니다.</p>
176페이지의 "서버 구성"	<p>Directory Server 데이터 서비스의 자원 유형 이름인 <code>SUNW.dsldap</code>와 관리 서버 데이터 서비스인 <code>SUNW.mps</code></p> <p>데이터 서비스의 마스터로 사용할 수 있는 클러스터 노드 이름</p> <p>Directory Server와 관리 서버에 액세스하는 클라이언트가 사용하는 논리적 호스트 이름</p> <p>전역 파일 시스템에서 Directory Server를 설치하는 <i>ServerRoot</i> 위치</p> <p>Directory Server에서 클라이언트 요청을 수신하는 포트</p> <p>관리 서버에서 클라이언트 요청을 수신하는 포트</p> <p>173페이지의 "네트워크 자원 설정"에 정의된 자원 그룹 이름</p>
178페이지의 "확장 등록정보 구성"	<p>자세한 내용은 해당 절을 참조하십시오.</p>

# 네트워크 자원 설정

Sun Cluster 소프트웨어는 개별 네트워크 인터페이스의 노드 이름 및 호스트 이름과 다른 논리적 호스트 이름을 관리합니다. 그림 C-1은 2노드 클러스터에서 관리되는 논리적 호스트 이름이 어떻게 어느 노드와도 영구적으로 연결되지 않는지에 대해 보여줍니다.

**그림 C-1** 2노드 클러스터



Directory Server 데이터 서비스용 Sun Cluster HA를 설치하는 경우 Directory Server와 관리 서버가 논리적 호스트 이름 인터페이스에서 수신하도록 구성합니다. 이렇게 하면 두 서버가 클러스터의 특정 노드에 연결되지 않으므로 Sun Cluster 소프트웨어에서 페일오버를 관리할 수 있습니다. 그림 C-1에서 두 노드 이름은 foo와 bar로 지정되어 있습니다. 하지만 그림 C-1과 같이 설치 중에 사용하는 논리적 호스트 이름은 foo와 bar가 아니라 ds-1.example.com과 ds-2.example.com이 됩니다. 사용되는 논리적 호스트 이름은 전체 도메인 이름입니다.

이러한 주요 개념에 대한 자세한 내용과 논리적 호스트 이름 설정 방법은 Sun Cluster 3.0 제품 설명서를 참조하십시오.

논리적 호스트 이름을 설정한 후 다음 단계를 수행합니다.

1. 클러스터 노드에서 수퍼유저가 됩니다.

2. 사용하는 모든 네트워크 주소가 이름 서비스 데이터베이스에 추가되어 있는지 확인합니다.

이름 서비스 조회 중에 장애를 방지하려면 각 클러스터 노드의 `/etc/hosts` 파일에 전체 논리적 호스트 이름과 공유 IP 주소가 있는지도 확인합니다. 또한 다른 이름 서비스에 액세스하기 전에 먼저 각 클러스터 노드에 있는 `/etc/nsswitch.conf`의 이름 서비스 매핑에서 로컬 파일을 확인하도록 구성합니다.

3. 네트워크 및 응용 프로그램 자원을 저장할 페일오버 자원 그룹을 작성합니다. 예를 들어 아래 명령을 실행합니다.

```
# scrgadm -a -g resource-group [-h node-list]
```

여기서 `resource-group`은 해당 그룹의 이름을 지정합니다.

선택 사항인 `node-list`는 클러스터의 잠재적 마스터 노드를 식별하는 ID 또는 물리적 노드 이름의 쉼표로 구분된 목록입니다. 노드 이름 순서에 따라 페일오버 중에 주 노드로 사용되는 노드 순서가 결정됩니다. 클러스터의 모든 노드가 잠재적 마스터인 경우에는 `node-list`를 지정할 필요가 없습니다.

4. 아래 명령을 실행하여 논리적 호스트 이름 자원을 자원 그룹에 추가합니다.

```
# scrgadm -a -L -g resource-group -l logical-host-names [-n netif-list]
```

여기서 `logical-host-names`는 논리적 호스트 이름으로 사용되는 전체 도메인 이름의 쉼표로 구분된 목록입니다. Directory Server 인스턴스 당 한 개의 논리적 호스트 이름을 사용합니다.

선택 사항인 `netif-list`는 각 노드의 NAFO 그룹을 식별하는 쉼표로 구분된 목록입니다. 이 옵션을 지정하지 않으면 `scrgadm(1M)` 유틸리티는 단계 3에서 지정된 `node-list`의 각 노드에 설정된 논리적 호스트 이름이 사용하는 서브넷에서 네트워크 어댑터를 찾습니다.

5. 단계 4에서 논리적 호스트 이름으로 지정된 전체 도메인 이름이 이름 서비스 데이터베이스에 추가되어 있는지 확인합니다.

6. 아래 명령을 실행하여 자원 그룹을 활성화하고 온라인 상태로 만듭니다.

```
# scswitch -Z -g resource-group
```

자원 그룹이 온라인 상태이면 서버를 설치할 수 있습니다.

# 서버 설치

Directory Server용 Sun Cluster HA에서 Directory Server와 관리 서버는 모두 Sun Cluster의 제어를 받습니다. 따라서 설치 중에 물리적 노드의 전체 도메인 이름을 제공하는 것이 아니라 장애가 발생할 때 다른 노드로 복구될 수 있는 전체 논리적 호스트 이름을 서버에 제공합니다.

디렉토리 클라이언트 응용 프로그램에서 사용하는 논리적 호스트 이름에 대해 온라인 상태인 노드부터 설치를 수행하여 Directory Server 데이터 서비스의 마스터로 사용할 다른 모든 클러스터 노드에 대해 이 프로세스를 반복합니다.

## 활성 노드에 설치

디렉토리 클라이언트 응용 프로그램에서 사용하는 논리적 호스트 이름에 대해 온라인 상태인 클러스터 노드인 경우 다음을 수행합니다.

1. 24페이지의 "Solaris 패키지 설치"를 참조하여 Directory Server 및 관리 서버용 Solaris 패키지를 설치합니다.
2. Directory Server를 구성합니다. 자세한 내용은 28페이지의 "Directory Server 구성"을 참조하십시오.

이 단계를 완료할 때 다음을 수행합니다.

- o Directory Server 인스턴스를 전역클러스터 파일 시스템에 저장합니다.
- o 노드 이름이 아닌 논리적 호스트 이름을 사용합니다.

3. 29페이지의 "관리 서버 구성"을 참조하고 Directory Server 구성에 사용된 논리적 호스트 이름을 사용하여 관리 서버를 구성합니다.
4. Directory Server를 보안 모드 전용으로 사용하는 경우 `ServerRoot/slapd-serverID/keypass`라는 빈 파일을 작성하여 Directory Server 인스턴스가 보안 모드로 실행되고 있음을 클러스터에 표시합니다.

또한 자동으로 보안 모드의 인스턴스를 시작하기 위한 암호가 저장된 `ServerRoot/alias/slapd-serverID-pin.txt` 파일을 작성합니다. 이렇게 하면 클러스터에서 사용자 개입 없이 데이터 서비스를 다시 시작할 수 있습니다.

## 다른 노드에 설치

Directory Server 데이터 서비스의 마스터로 사용할 노드인 경우 다음을 수행합니다.

1. 24페이지의 "Solaris 패키지 설치"를 참조하여 Directory Server 및 관리 서버용 Solaris 패키지를 설치합니다.
2. 175페이지의 "활성 노드에 설치"에서와 동일한 설정을 사용하여 Directory Server를 구성합니다.
3. 175페이지의 "활성 노드에 설치"에서와 동일한 설정을 사용하여 관리 서버를 구성합니다.
4. 첫 노드의 `ServerRoot/alias/slaped-serverID-pin.txt`를 `ServerRoot/alias/`에 복사합니다.

---

**주** 전역 파일 시스템에 있는 파일은 제거하거나 위치를 바꾸지 마십시오.

---

## 데이터 서비스 패키지 설치

데이터 서비스 패키지인 SUNWdsha와 SUNWasha는 서버를 클러스터 내의 데이터 서비스로 관리하기 위한 관리 인터페이스를 제공합니다.

- Directory Server 데이터 서비스를 지원할 각 클러스터 노드에서 `pkgadd(1M)` 유틸리티를 사용하여 데이터 서비스 패키지를 설치합니다.

```
# pkgadd -d dirContainingPackages SUNWasha SUNWdsha
```

## 서버 구성

아래 단계는 Directory Server에서 사용하는 논리적 호스트 이름에 대해 온라인 상태인 클러스터 노드 *에서만* 사용합니다.

1. 슈퍼유저가 됩니다.
2. 아래 명령을 실행하여 Directory Server와 관리 서버를 중지합니다.

```
# /usr/sbin/directoryserver stop
# /usr/sbin/mpsadminserver stop
```

3. 아래 명령을 실행하여 두 데이터 서비스의 자원 유형을 등록합니다.

```
# scrgadm -a -t SUNW.dsldap -f /etc/ds/v5.2/cluster/SUNW.dsldap
# scrgadm -a -t SUNW.mps -f /etc/mps/admin/v5.2/cluster/SUNW.mps
```

여기서 `SUNW.dsldap`와 `SUNW.mps`는 데이터 서비스에 미리 정의된 자원 유형 이름입니다. `/etc/ds/v5.2/cluster/SUNW.dsldap` 및

`/etc/mps/admin/v5.2/cluster/SUNW.mps`는 데이터 서비스를 정의합니다.



4. 아래 명령을 실행하여 173페이지의 "네트워크 자원 설정"에서 작성한 페일오버 자원에 서버를 추가합니다.

```
# scrgadm -a -j resource-name-ds -g resource-group -t SUNW.dsldap \
-y Network_resources_used=logical-host-name \
-y Port_list=port-number/tcp \
-x Confdir_list=ServerRoot/slapd-serverID
```

```
# scrgadm -a -j resource-name-as -g resource-group -t SUNW.mps \
-y Network_resources_used=logical-host-name \
-y Port_list=port-number/tcp \
-x Confdir_list=ServerRoot
```

여기서 새로운 *resource-name-ds*를 제공하여 Directory Server 인스턴스를 식별하고 새로운 *resource-name-as*를 제공하여 관리 서버 인스턴스를 식별합니다.

*resource-group* 매개 변수는 173페이지의 "네트워크 자원 설정"에서 지정한 그룹의 이름입니다.

*logical-host-name*은 현재 Directory Server 인스턴스에 사용된 로컬 호스트 이름을 식별합니다.

*port-number*는 175페이지의 "서버 설치"에서 지정한 포트 번호로, 서버 인스턴스에서 클라이언트 요청을 수신하는 포트입니다. 각 명령의 *Port\_list* 매개 변수는 한 개의 항목만 받습니다.

*ServerRoot* 및 *ServerRoot/slapd-serverID*는 175페이지의 "서버 설치"에서 지정한 경로입니다. 각 명령의 *Confdir\_list* 매개 변수는 한 개의 항목만 받습니다.

5. 아래 명령을 실행하여 서버 자원과 모니터를 활성화합니다.

```
# scswitch -e -j resource-name-ds
# scswitch -e -j resource-name-as
```

여기서 *resource-name-ds*와 *resource-name-as*는 단계 4에서 서버를 식별하기 위해 제공한 이름입니다.

---

**주** 서버를 구성한 후에는 클러스터의 비활성 노드에서 db2bak, db2ldif, back2db, ldif2db 등의 백업 및 복원 명령을 실행하지 마십시오. 백업 및 복원 절차는 모두 활성 노드에서 수행해야 합니다.

---

6. 페일오버의 성능을 향상시키려면 181페이지의 "HA 저장소 및 데이터 서비스 동기화" 절에 설명된 단계를 수행합니다.

## 등록 및 구성 예제

코드 예제 C-1은 173페이지의 그림 C-1에 표시된 클러스터의 데이터 서비스를 등록하고 구성하는 방법을 보여줍니다.

### 코드 예제 C-1      데이터 서비스 등록 및 구성

```
(Create a failover resource group on the node that is online.)
# scrgadm -a -g ds-resource-group-1 -h foo,bar

(Add a logical hostname resource to the resource group.)
# scrgadm -a -L -g ds-resource-group-1 -l ds-1.example.com

(Bring the resource group online.)
# scswitch -Z -g ds-resource-group-1

(Install packages on each node in the cluster.)

(Stop the servers on the node that is online.)
# /usr/sbin/directoryserver stop
# /usr/sbin/mpsadminserver stop

(Register the SUNW.dsldap and SUNW.mps resource types.)
# scrgadm -a -t SUNW.dsldap -f /etc/ds/v5.2/cluster/SUNW.dsldap
# scrgadm -a -t SUNW.mps -f /etc/mps/admin/v5.2/cluster/SUNW.mps

(Create resources for the servers and add them to the resource group.)
# scrgadm -a -j ds-1 -g ds-resource-group-1 \
-t SUNW.dsldap -y Network_resources_used=ds-1.example.com \
-y Port_list=389/tcp \
-x Confdir_list=/global/ds/slapd-ds-1
# scrgadm -a -j as-1 -g ds-resource-group-1 \
-t SUNW.mps -y Network_resources_used=ds-1.example.com \
-y Port_list=5201/tcp \
-x Confdir_list=/global/ds

(Enable the application resources.)
# scswitch -e -j ds-1
# scswitch -e -j as-1
```

## 확장 등록정보 구성

확장 등록정보를 사용하여 클러스터 소프트웨어에서 응용 프로그램 소프트웨어를 처리하는 방법을 구성할 수 있습니다. 예를 들어, 클러스터 소프트웨어에서 데이터 서비스의 페일오버 시기를 확인하는 방법을 조정할 수 있습니다.

## 구성 가능한 등록정보

일반적으로 자원 확장 등록정보를 구성할 때는 Sun Management Center의 Cluster 모듈을 사용하거나 `scrgadm` 유틸리티를 사용합니다. `scrgadm` 유틸리티를 `-x parameter=value` 옵션과 함께 사용하여 표 C-2에 열거된 확장 등록정보를 변경할 수 있습니다.

**표 C-2** SUNW.dslldap 자원 확장 등록정보

등록정보	설명	기본값	범위
Monitor_retry_count	Monitor_retry_interval 값으로 지정된 시간 창 내에 PMF(Process Monitor Facility)가 장애 모니터를 다시 시작하는 횟수를 표시하는 정수 값입니다.	4회	-1 - 2,147,483,641회 -1은 재시도 횟수에 제한이 없음을 나타냅니다.
Monitor_retry_interval	장애 모니터의 실패 횟수가 카운트된 시간을 표시하는 정수 값입니다.  이 기간 동안 장애 모니터 실패 횟수가 Monitor_retry_count에 지정된 값을 초과하면 PMF에서 장애 모니터를 다시 시작할 수 없습니다.	2분	-1 - 2,147,483,641분 -1은 재시도 간격에 제한이 없음을 나타냅니다.
Probe_timeout	장애 모니터가 디렉토리 서버 인스턴스 검색에 사용하는 제한 시간 값(초)을 표시하는 정수 값입니다.	30초	0 - 2,147,483,641초

Sun Cluster 등록정보에 대한 자세한 내용은 Sun Cluster 3.0 제품 설명서를 참조하십시오.

## 장애 모니터 작동 방식

클러스터 소프트웨어는 장애 모니터를 사용하여 데이터 서비스의 상태를 확인합니다. 장애 모니터는 데이터 서비스를 검색한 다음, 이 검색 결과에 따라 서비스 상태가 양호한지 또는 다시 시작해야 하는지 결정합니다.

**표 C-3** 장애 모니터의 검색 해석 방법

Directory Server 실행 모드	사용된 검색	알고리즘
일반 모드	ldapsearch	<ol style="list-style-type: none"> <li>1. 검색을 시도합니다.</li> <li>2. 검색 결과, <ul style="list-style-type: none"> <li>• LDAP_SUCCESS가 표시되면 서비스는 양호한 상태입니다.</li> <li>• LDAP 오류가 발생하면 서비스를 다시 시작해야 합니다.</li> <li>• 시간 초과 이외의 문제가 발생하면 장애 모니터는 Monitor_retry_count 및 Monitor_retry_interval 값에 따라 다시 검색합니다.</li> <li>• Probe_timeout 기간이 초과되면 장애 모니터는 Monitor_retry_count 및 Monitor_retry_interval 값에 따라 다시 검색합니다.</li> </ul> </li> </ol> <p>시스템, 네트워크 또는 Directory Server 인스턴스의 로드 증가로 인해 시간 초과가 발생할 수 있습니다. 모니터되는 Directory Server 인스턴스의 수에 비해 Probe_timeout 값이 너무 낮게 설정된 경우에도 시간 초과가 나타납니다.</p>
보안 모드(SSL)	TCP 연결	<ol style="list-style-type: none"> <li>1. 연결을 시도합니다.</li> <li>2. 그 결과, <ul style="list-style-type: none"> <li>• 연결에 성공하면 서비스는 양호한 상태입니다.</li> <li>• 연결에 실패하면 서비스를 다시 시작해야 합니다.</li> <li>• Probe_timeout을 초과하면 서비스를 다시 시작해야 합니다.</li> </ul> </li> </ol>

장애 모니터는 176페이지의 "서버 구성"에서 지정한 IP 주소 및 포트 번호를 사용하여 검색 작업을 수행합니다. Directory Server가 두 포트에서 각각 SSL 트래픽과 일반 트래픽을 수신하도록 구성되어 있으면 장애 모니터는 TCP 연결을 사용하여 두 포트를 모두 검색하고 보안 모드 포트에 사용되는 장애 모니터링 알고리즘을 따릅니다.

## HA 저장소 및 데이터 서비스 동기화

SUNW.HAStorage 자원 유형은 HA 저장소와 데이터 서비스 간의 작업을 동기화하여 Directory Server와 같이 디스크 사용량이 많은 데이터 서비스의 페일오버 성능을 향상시킵니다.

Directory Server 데이터 서비스를 HA 저장소와 동기화하려면 데이터 서비스에서 사용하는 논리적 호스트 이름에 대해 온라인 상태인 노드에서 다음 단계를 수행합니다.

1. 아래 명령을 실행하여 HA 저장소 자원 유형을 등록합니다.

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

2. 아래 명령을 실행하여 저장소 자원이 동기화 상태를 유지하도록 구성합니다.

```
# scrgadm -a -j HAStorage-resource-name -g HAStorage-resource-group \  
-t SUNW.HAStorage -x ServicePaths=volume-mount-point \  
-x AffinityOn=True
```

여기서 *volume-mount-point*는 Directory Server에서 데이터를 저장하는 디스크 볼륨을 식별합니다.

3. 아래 명령을 실행하여 저장소 자원과 모니터를 활성화합니다.

```
# scswitch -e -j HAStorage-resource-name
```

4. 아래 명령을 실행하여 기존 Directory Server 자원에 대한 종속성을 추가합니다.

```
# scrgadm -c -j resource-name-ds \  
-y Resource_Dependencies=HAStorage-resource-name
```

배경 정보는 SUNW.HAStorage(5)를, 새 자원에 SUNW.HAStorage 자원 유형을 설정하는 방법은 Sun Cluster 3.0 제품 설명서를 참조하십시오.

## 추가 Directory Server 인스턴스 작성

다음 단계를 수행합니다.

1. Sun ONE 서버 콘솔을 사용하여 추가 Directory Server 인스턴스를 작성합니다.

자세한 내용은 *Sun ONE Server Console Server Management Guide*를 참조하십시오.

2. 아래 명령을 실행하여 데이터 서비스에서 사용하는 논리적 호스트 이름에 대해 온라인 상태인 노드에서 새 Directory Server 인스턴스를 중지합니다.

```
# /usr/sbin/directoryserver -server serverID stop
```

3. 아래 명령을 실행하여 173페이지의 "네트워크 자원 설정"에서 작성한 페일오버 자원에 Directory Server를 추가합니다.

```
# scrgadm -a -j resource-name-ds -g resource-group -t SUNW.dsldap \
-y Network_resources_used=logical-host-name \
-y Port_list=port-number/tcp \
-x Confdir_list=ServerRoot/slapd-serverID
```

여기서 새로운 *resource-name-ds*를 제공하여 Directory Server 인스턴스를 식별합니다.

*resource-group* 매개 변수는 173페이지의 "네트워크 자원 설정"에서 지정한 그룹의 이름입니다.

*logical-host-name*은 해당 인스턴스에 사용된 로컬 호스트 이름을 식별합니다.

*port-number*는 175페이지의 "서버 설치"에서 지정한 포트 번호로, 해당 인스턴스에서 클라이언트 요청을 수신하는 포트입니다. *Port\_list* 매개 변수는 한 개의 항목만 받습니다.

*ServerRoot* 및 *ServerRoot/slapd-serverID*는 175페이지의 "서버 설치"에서 지정한 경로입니다. *Confdir\_list* 매개 변수는 한 개의 항목만 받습니다.

4. 아래 명령을 실행하여 서버 자원과 모니터를 활성화합니다.

```
# scswitch -e -j resource-name-ds
```

여기서 *resource-name-ds*는 단계 3에서 Directory Server를 식별하기 위해 제공한 이름입니다.

## 제거

Directory Server용 Sun Cluster HA 및 연결된 관리 서버를 클러스터에서 제거하려면 다음 단계를 수행합니다.

1. 아래 명령을 실행하여 서버 인스턴스를 중지합니다.

```
# scswitch -n -j resource-name-ds
# scswitch -n -j resource-name-as
```

2. 아래 명령을 실행하여 자원을 제거합니다.

```
# scrgadm -r -j resource-name-ds
# scrgadm -r -j resource-name-as
```

3. 아래 명령을 실행하여 클러스터 데이터베이스에서 자원 유형을 제거합니다.

```
# scrgadm -r -t SUNW.dsldap  
# scrgadm -r -t SUNW.mps
```

4. 아래 명령을 실행하여 서버 구성을 삭제합니다.

```
# /usr/sbin/mpsadmserver unconfigure  
# /usr/sbin/directoryserver unconfigure
```

5. pkgrm(1M) 유틸리티를 사용하여 SUNWdsha와 SUNWasha 등의 설치된 패키지를 각 노드에서 제거합니다.

제거



## 가

가상 목록 보기 색인 131  
감사 로그 142  
구성 디렉토리 18  
근사 색인 132

## 다

다시 시작  
    디렉토리 서비스 100  
데이터 이전. 업그레이드 참조  
동일 색인 129  
디렉토리 관리자 21

## 라

레이아웃  
    구성 파일 162  
    도구 161  
    서버 인스턴스 파일 162-164  
    온라인 도움말 파일 160  
    유틸리티 159  
    자동 설치 템플릿 파일 162  
    제품 라이브러리 160  
    제품 이진 160  
    플러그인 파일 161

레트로 변경 로그 146  
로그  
    감사 142  
    레트로 변경 로그 146  
    복제 변경 로그 145  
    액세스 140  
    오류 143  
    유형 139  
    트랜잭션 147

## 마

문제 해결 46-49

## 바

보안 98-100  
    강력한 암호 99  
    방화벽 98  
    사용자 및 그룹 99  
    서비스 100  
    이중 부트 금지 98  
복제 변경 로그 145

## 사

사용자 디렉토리 18

색인

검색 시 사용 127, 134

국가별 132

근사 132

단편화 138

동일 129

손실 127-133

유형 125

이점 75, 126-127

있음 128

조정 134-138

찾아보기(VLV) 131

크기 제한 76, 135-138

파일 125

하위 문자열 130

32비트와 64비트 비교 127

설치 23-41

레지스트리 41

압축된 아카이브 31-38

자동 29-31, 32-33, 37-38, 39-40

클러스터 171-179

패키지 23-31

필수 사항 17-22, 23-28, 31, 34, 39

설치 위치 10-11

실재 색인 128

## 아

액세스 로그 140

액세스 제어 155-156

업그레이드

단일 서버 51-53, 54-57

데이터 이전 55

복제 계약 56

복제된 서버 53, 57-70

사용자 정의 4.x 스키마 55

지원 받기 54

포트 번호 57

필수 사항 51-54

오류 로그 143

이전. 업그레이드 참조

## 자

자동 설치 29-31, 32-33, 37-38, 39-40

템플릿 파일 162

제거 41-45

클러스터 182

조정

검색 크기 151

권장되는 작업 생성 101

로그 76, 140-148

색인 134-138

스레드 103, 151, 155

시간 제한 152

시스템 설정 102-106

시스템 자원 152-155

액세스 제어 155-156

휴식 연결 150

자원 제한 76, 149-152

차단된 연결 150

캐시 75, 107-123

큰 파일 103

팁 73-77

파일 설명자 103, 153, 154

플러그인 156-157

항목 크기 151

IP 인터페이스 152, 154

SSL 165-170

TCP 104-106, 153

지원되는 플랫폼 18

## 차

찾아보기 색인 131

## 카

### 캐시

- 가져오기 110
- 검색에서 사용 112-114
- 데이터베이스 108
- 모니터 119, 123
- 업데이트 시 사용 115-116
- 접미사 초기화 시 사용 116-118
- 초기화 122
- 총 크기 111
- 최적화 118-123
- 파일 시스템 111
- 항목 109

### 캐시 종류 107

### 코어 파일

- 생성 기능 활성화 34, 38
- 크기 조정 88

### 크기 조정

- 네트워크 용량 94
- 다중 프로세서 시스템 94
- 데이터베이스 파일 89
- 디스크 하위 시스템 85-94
- 로그 86, 89
- 백업 87
- 총 캐시 111
- 최소 요구 사항 79-81
- 코어 파일 88
- iostat 94
- LDIF 파일 87
- RAID 90-93
- RAM 81-85
- RAM 부족 84
- SSL 94

### 클러스터

- 구성 176
- 네트워크 자원 172
- 설치 174
- 자원 확장 178
- 필수 사항 171-172

## 타

트랜잭션 로그 147

## 파

### 패치

필수 27, 31, 35, 98

포트 번호 21, 57

### 플러그 인

- 레거시 복제 156
- 참조 무결성 157
- 7비트 검사 156

하드웨어 크기 조정. 크기 조정 참조

하위 문자열 색인 130

## C

coreadm 34, 99

currenttrycachecount 123

currenttrycachesize 123

## D

dbcachehitratio 123

dbcachepagein 123

dbcacheroevict 123

DPC 102

## E

entrycachehitratio 123

## I

idsktune 28, 31, 35, 97, 98, 101, 102

## M

maxentrycachesize 123

## N

nsslapd-accesslog 140  
nsslapd-accesslog-level 140, 141  
nsslapd-accesslog-logbuffering 140  
nsslapd-accesslog-logging-enabled 141  
nsslapd-accesslog-logmaxdiskspace 141  
nsslapd-accesslog-logminfreediskspace 141  
nsslapd-allidsthreshold 135, 137  
nsslapd-auditlog 142  
nsslapd-auditlog-logging-enabled 142  
nsslapd-auditlog-logmaxdiskspace 142  
nsslapd-auditlog-logminfreediskspace 143  
nsslapd-cachememsize 82, 109, 120, 145  
nsslapd-cachesize 109, 120, 146  
nsslapd-changelogdir 145, 146  
nsslapd-changelogmaxage 147  
nsslapd-changemaxage 145  
nsslapd-changemaxentries 146  
nsslapd-dbcachesize 82, 108, 119  
nsslapd-db-checkpoint-interval 148  
nsslapd-db-durable-transaction 148  
nsslapd-db-home-directory 90  
nsslapd-db-logdirectory 148  
nsslapd-directory 89, 125  
nsslapd-errorlog 88, 144  
nsslapd-errorlog-logging-enabled 144  
nsslapd-errorlog-logmaxdiskspace 144  
nsslapd-errorlog-logminfreediskspace 144  
nsslapd-idletimeout 150

nsslapd-import-cachesize 82, 110  
nsslapd-infolog-area 144  
nsslapd-infolog-level 145  
nsslapd-ioblocktimeout 150  
nsslapd-listenhost 152, 154  
nsslapd-lookthroughlimit 127, 135  
    조정  
        검색 크기 150  
nsslapd-maxbersize 151  
nsslapd-maxconnections 82  
nsslapd-maxdescriptors 153, 154  
nsslapd-maxthreadsperconn 151, 155  
nsslapd-nagle 153  
nsslapd-require-index 134  
nsslapd-reserveddescriptors 154  
nsslapd-schema-repl-useronly 53  
nsslapd-securelistenhost 152, 154  
nsslapd-sizelimit 127, 150, 151  
nsslapd-threadnumber 82, 155  
nsslapd-timelimit 127, 152  
NTP 100

## S

*ServerRoot*. 설치 위치 참조  
SSL  
    가속 165–170