



Sun Java™ Desktop System Configuration Manager, 릴리스 1 설치 설명서

Sun Microsystems, Inc.
4150 Network Circle
Santa Clara, CA 95054,
U.S.A. 650-960-1300

부품 번호: 817-5592-10

2004년 4월, 개정판 A

저작권 및 상표

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

Sun Microsystems, Inc.는 본 설명서에 설명된 기술 관련 지적 재산을 소유합니다. 특히 이 지적 재산권에는 <http://www.sun.com/patents>에 나열된 하나 이상의 미국 특허권이 포함될 수 있으며 하나 이상의 추가 특허권 또는 미국 및 다른 국가에서 특허 출원 중인 응용 프로그램이 제한 없이 포함될 수 있습니다.

본 설명서와 제품은 사용, 복제, 배포, 역컴파일을 제한하는 라이선스 규정에 따라 배포됩니다. Sun과 사용 허여자(있을 경우)의 사전 서면 승인 없이는 본 제품이나 설명서를 일체 복제할 수 없습니다.

글꼴 기술을 비롯한 타사 소프트웨어의 저작권 및 라이선스는 Sun 공급업체에게 있습니다.

본 제품은 독립 회사인 JPEG Group과 FreeType Project의 기술을 부분적으로 활용하고 있습니다.

Portions Copyright 2000 SuSE, Inc. Word for Word Copyright © 1996 Inso Corp. International CorrectSpell spelling correction system Copyright © 1995 by Lernout & Hauspie Speech Products N.V. 모든 권리는 저작권자의 소유입니다.

Sun, Sun Microsystems, Sun 로고, Java, Solaris, StarSuite, Butterfly 로고, Solaris 로고 및 StarSuite 로고는 미국 및 다른 국가에서 Sun Microsystems, Inc.의 상표 또는 등록 상표입니다.

UNIX는 미국 및 다른 국가에서 X/Open Company, Ltd.를 통해 독점적으로 사용권이 부여되는 등록 상표입니다. Screen Beans와 Screen Beans 클립아트 문자는 A Bit Better Corporation의 등록 상표입니다.

연방 정부 취득 상용 소프트웨어 정부 사용자는 표준 사용권 조항 및 규정을 준수해야 합니다.

설명서에 "있는 그대로" 제공되며 법률을 위반하지 않는 범위 내에서 상품성, 특정 목적에 대한 적합성 또는 비침해에 대한 묵시적인 보증을 포함하여 모든 명시적 또는 묵시적 조건, 표현 및 보증 배제합니다.

목차

1 소개.....	5
2 LDAP 서버.....	7
개념.....	7
설치.....	7
추가 고려 사항.....	12
3 Sun™ Web Console.....	13
시스템 요구 사항.....	13
Sun Web Console 설치.....	14
콘솔 실행.....	15
Sun Web Console 제거.....	15
Sun Web Console 포트 정보.....	15
4 Sun Java™ Desktop System Configuration Manager, 릴리스 1.....	17
Configuration Manager 설치.....	17
Configuration Manager 실행.....	18
Configuration Manager 제거.....	18
5 데스크탑 구성 요소.....	19
데이터 액세스/사용자 인증.....	19
Configuration Agent.....	20
GConf 어댑터.....	22
Mozilla 어댑터.....	22
StarSuite 어댑터.....	22
6 부록 A – Sun Web Console.....	23
알려진 문제점.....	23
설치 스크립트 사용.....	23
Sun Web Console 패키지.....	24
7 부록 B – Configuration Manager.....	25
Configuration Manager 패키지.....	25
8 부록 C.....	27
Configuration Manager 와 함께 OpenLDAP 서버 사용.....	27
Configuration Manager 와 함께 Active Directory Server 사용.....	28

소개

Sun Java™ Desktop System Configuration Manager, 릴리스 1은 Sun Java™ Desktop System이 실행되는 데스크탑 호스트를 중앙 집중식으로 구성하기 위한 제품입니다. 이 제품을 사용하면 조직 구조의 다양한 요소에 설정을 지정할 수 있으므로 관리자가 사용자 또는 호스트 그룹을 편리하게 관리할 수 있습니다. 제품의 주요 구성 요소는 다음과 같습니다.

- 관리할 사용자 및 호스트의 조직 구조가 포함되어 있으며 구성 데이터가 보관되는 LDAP 서버
- 관리자가 구성 데이터를 정의하여 조직 구조의 각 요소에 지정할 수 있도록 도와주는 웹 기반 관리 도구
- 클라이언트 호스트에 설치되어 현재 로그인한 사용자 대신 구성 데이터를 검색한 후 Java Desktop System을 구성하는 다양한 응용 프로그램에 제공하는 데스크탑 구성 요소

관리 도구는 Sun Web Console에서 실행되는 웹 기반 응용 프로그램입니다. 관리자는 이 도구를 사용하여 LDAP 서버의 조직 구조를 탐색하고 각 요소에 정책을 지정할 수 있습니다. 정책은 관리 도구에서 처리할 설정을 정의하는 정책 템플릿에 따라 표시 및 편집됩니다.

데스크탑 구성 요소는 사용자 대신 LDAP 서버로부터 구성 데이터를 검색하여 다수의 구성 시스템 어댑터에 제공하는 Sun Java™ Desktop System Configuration Agent를 중심으로 구성됩니다. 구성 시스템 어댑터는 정책 설정을 사용하여 로컬 구성(응용 프로그램에서 제공하는 기본 설정과 사용자 설정)을 보완합니다. 현재 지원되는 구성 시스템에는 Gnome 데스크탑이나 Evolution과 같은 Gnome 응용 프로그램의 구성을 처리하는 GConf, Mozilla™ Preferences, StarRegistry(StarSuite 구성 시스템) 등이 있습니다.

LDAP 서버

개념

Java Desktop System Configuration Manager 프레임워크에서 구성 데이터는 LDAP 리포지토리의 항목으로 회사 조직 구조의 각 요소에 해당하는 엔티티와 연결됩니다.

인식되는 엔티티는 다음과 같습니다.

- 조직: 일반적으로 전체 계층 구조의 조직(사업부, 그룹, 팀) 또는 지역(대륙, 국가, 사이트) 단위를 나타냅니다.
- 사용자: 전체 계층 구조의 리프 노드를 나타내며 이름에서 알 수 있듯이 대체로 사용자가 됩니다.
- 도메인: 네트워크 조직의 논리적 구조 단위를 나타냅니다.
- 호스트: 역시 전체 계층 구조의 리프 노드를 나타내지만 네트워크상의 개별 컴퓨터를 가리킵니다.
- 역할: 일반적으로 기능(관리자, 사이트 관리) 면에서 구별되며 특정 사용자 집합에 적용되는 등록 정보를 나타냅니다.

조직 및 사용자 엔티티는 사용자 트리를 정의하는 데 사용되고 도메인 및 호스트 엔티티는 호스트 트리를 정의합니다. 두 트리는 독립적이지만 프레임워크에서 유사한 방식으로 처리됩니다.

조직 및 도메인 엔티티와 다른 항목 간의 관계는 리포지토리 내의 물리적 항목 위치에 의해 정의됩니다. 즉, 트리에서 두 엔티티 아래에 위치하는 항목은 모두 조직 및 도메인 엔티티에 포함될 수 있습니다. 역할과 사용자 또는 호스트 간의 관계는 사용자 및 호스트 항목의 속성에 의해 정의됩니다.

엔티티에 연결된 구성 데이터는 프레임워크에서 관리하는 특수 항목에 저장됩니다. 특수 항목은 각 항목에 연결된 서비스 이름과 서비스 컨테이너로 식별됩니다.

설치

기존 LDAP 서버를 Configuration Manager와 함께 사용하려면 다음 작업을 수행해야 합니다.

- Configuration Manager에서 구성 데이터를 저장하는 데 사용하는 사용자 정의 개체 클래스와 속성을 지원하도록 서버 스키마를 확장합니다.
- 리포지토리에 있는 항목과 Configuration Manager에서 지원하는 엔티티의 매핑 정보를 사용자 정의 및 저장합니다.

배포 도구

기존 LDAP 서버를 Configuration Manager와 함께 사용하려면 설치 CD에 있는 다음 배포 도구를 설치해야 합니다.

- 88apoc-registry.ldif - 구성 데이터를 저장하는 데 필요한 개체 클래스와 속성을 소개하는 스키마 파일
- OrganizationMapping - LDAP 항목과 Configuration Manager 엔티티 간의 매핑을 설명하는 기본 등록 정보 파일
- UserProfileMapping - LDAP 사용자 항목 속성과 Configuration Manager 사용자 프로필 속성 간의 매핑을 설명하는 기본 등록 정보 파일
- createServiceTree - LDAP 리포지토리에 매핑 파일을 저장하는 스크립트
- deployApoc: - LDAP 서버의 스키마를 확장하고 LDAP 리포지토리에 매핑 파일을 저장하는 스크립트

스키마 확장

구성 데이터는 해당 항목에 연결된 항목 트리에 저장됩니다. 각 트리에서 사용되는 개체 클래스와 속성을 LDAP 서버에 저장하려면 먼저 개체 클래스와 클래스를 LDAP 서버 스키마에 추가해야 합니다. 예를 들어, 제공된 스키마 확장 파일은 LDIF 형식을 사용하여 개체와 클래스를 Sun Java System Directory Server에 추가합니다. 다른 LDAP 서버에 개체와 클래스를 추가하려면 해당 서버에서 인식하는 형식을 사용해야 합니다.

조직 매핑

LDAP 항목과 Configuration Manager 엔티티 간의 매핑을 정의하려면 조직 매핑 파일을 편집하여 LDAP 리포지토리의 레이아웃에 맞는 값을 각 키에 입력해야 합니다.

사용자 엔티티는 모든 엔티티가 사용하는 개체 클래스와 전체 리포지토리에서 고유한 값을 갖는 속성으로 식별됩니다. 관리 응용 프로그램에서의 사용자 표시에 영향을 주는 표시 이름 형식을 제공할 수 있으며, 조직 내의 사용자 항목이 컨테이너 항목을 사용할 경우 컨테이너 항목을 선택적으로 정의할 수도 있습니다. 키 이름과 해당 기본값은 다음과 같습니다.

```
# Object class that all user entries use
User/ObjectClass=inetorgperson

# Attribute whose value in user entries is unique within the repository
User/UniqueIdAttribute=uid

# Optional container in organization entries of the user entries, remove
line if not used
User/Container=ou=People

# Display name format within the management application
User/DisplayNameFormat=sn, givenname
```

역할 엔티티는 사용할 수 있는 개체 클래스 목록과 해당 이름 지정 속성으로 식별됩니다. 각 목록에서 항목은 <item1>,<item2>,...,<itemN> 형식으로 정렬되어야 합니다. 즉, 동일한 수의 항목이 목록에 있어야 하고 n번째 개체 클래스에는 n번째 이름 지정 속성을 사용해야 합니다. 두 키는 역할

과 사용자 간의 관계는 물론 역할과 호스트 간의 관계도 정의합니다. `VirtualMemberAttribute` 키는 사용자 또는 호스트 항목에서 쿼리할 수 있는 값을 가진 속성을 지정하고 항목이 속해 있는 역할의 전체 DN을 포함해야 합니다. `MemberAttribute` 키는 사용자 또는 호스트 항목의 속성을 검색 필터에 지정하고 사용자 또는 호스트가 속해 있는 역할의 전체 DN을 포함해야 합니다.

`VirtualMemberAttribute` 키에는 서비스 클래스 가상 속성을 사용할 수 있지만 `MemberAttribute` 키는 필터에 사용할 수 있는 물리적 속성이어야 합니다. 키 이름과 해당 기본값은 다음과 같습니다.

```
# List of object classes for roles
Role/ObjectClass=nsRoleDefinition
# Aligned list of corresponding naming attributes
Role/NamingAttribute=cn
# Physical attribute (usable in a filter) containing the DN's of the roles of
a user/host
Role/MemberAttribute=nsRoleDN
# Attribute whose query on a user or host return the DN's of the roles it
belongs to
Role/VirtualMemberAttribute=nsRole
```

조직 엔티티는 정렬된 개체 클래스 목록과 해당 이름 지정 속성 목록을 사용하여 역할과 유사한 방식으로 식별됩니다. 키 이름과 해당 기본값은 다음과 같습니다.

```
# List of object classes for organizations
Organization/ObjectClass=organization
# Aligned list of corresponding naming attributes
Organization/NamingAttribute=o
```

도메인 엔티티는 조직 엔티티와 유사한 방식으로 식별됩니다. 키 이름과 해당 기본값은 다음과 같습니다.

```
# List of object classes for domains
Domain/ObjectClass=ipNetwork
# Aligned list of corresponding naming attributes
Domain/NamingAttribute=cn
```

호스트 엔티티는 사용자 엔티티와 유사한 방식으로 식별됩니다. 키 이름과 기본값은 다음과 같습니다.

```
# Object class that all host entries use
Host/ObjectClass=ipHost
# Attribute whose value in host entries is unique within the repository
Host/UniqueIdAttribute=cn
# Optional container in domain entries of the host entries, remove line if
not used
Host/Container=ou=Hosts
```

사용자 프로필 매핑

LDAP 사용자 항목 속성과 Configuration Manager 사용자 항목 속성 간의 매핑을 정의하려면 사용자 프로필 매핑 파일을 편집해야 합니다. 각 키는 Configuration Manager의 개별 사용자 속성을 나타내며 조직 매핑으로 식별되는 사용자 항목 속성의 이름 값으로 지정할 수 있습니다.

User/DisplayNameFormat 설정에 사용되는 속성은 사용자 프로필 매핑에서 지정해야 합니다. 키 이름과 해당 기본값은 다음과 같습니다.

```
# inetOrgPerson.givenName
org.openoffice.UserProfile/Data/givenname = givenname
# person.sn
org.openoffice.UserProfile/Data/sn = sn
# inetOrgPerson.initials
org.openoffice.UserProfile/Data/initials = initials
# organizationalPerson.street
org.openoffice.UserProfile/Data/street =
street,postalAddress,streetAddress
# organizationalPerson.l (city)
org.openoffice.UserProfile/Data/l = l
# organizationalPerson.st (state)
org.openoffice.UserProfile/Data/st = st
# organizationalPerson.postalCode
org.openoffice.UserProfile/Data/postalcode = postalcode
# country.c (country)
org.openoffice.UserProfile/Data/c =
# organizationalPerson.o (company)
org.openoffice.UserProfile/Data/o = o,organizationName
# deprecated -- no LDAP corollary
org.openoffice.UserProfile/Data/position =
# organizationalPerson.title
org.openoffice.UserProfile/Data/title = title
# inetOrgPerson.homePhone
org.openoffice.UserProfile/Data/homephone = homephone
# organizationalPerson.telephoneNumber
org.openoffice.UserProfile/Data/telephonenumber = telephonenumber
# organizationalPerson.facsimileTelephoneNumber
org.openoffice.UserProfile/Data/facsimiletelephonenumber =
facsimiletelephonenumber,officeFax
# inetOrgPerson.mail
org.openoffice.UserProfile/Data/mail = mail
```

배포

LDAP 리포지토리의 상태를 반영하여 매핑 파일의 사용자 정의가 끝나면 파일을 배포할 수 있습니다. 필요한 개체 클래스와 속성이 LDAP 서버 스키마에 이미 포함되어 있으면 createServiceTree 스크립트를 직접 실행할 수 있지만, 그렇지 않은 경우에는 deployApoc 스크립트를 실행해야 합니다.

deployApoc 스크립트는 Sun Java System Directory Server에서 사용하도록 작성되었으며 제공된 스키마 확장 파일을 올바른 디렉토리에 복사하고 LDAP 서버를 순환시킨 다음 createServiceTree 스크립트를 호출합니다. 스키마 리포지토리에 있는 파일을 복사하고 서버를 다시 시작할 수 있는 권한을 가진 사용자가 다음 명령을 실행하여 이 스크립트를 호출해야 합니다.

```
./deployApoc <directory server directory>
```

<directory server directory> 매개 변수는 Directory Server 설치의 slapd-<server name> 하위 디렉토리 경로여야 합니다.

설치 시 기본 디렉토리를 사용하고 서버 이름을 myserver.mydomain으로 지정한 경우 이 디렉토리는 /var/Sun/mps/slapd-myserver.mydomain이 됩니다.

직접 호출했던 deployApoc 스크립트에서 호출했던 관계 없이 createServiceTree 스크립트는 사용자에게 LDAP 서버 위치(호스트 이름, 포트 번호, 기본 DN)와 관리 권한이 있는 사용자의 정의(전체 DN, 암호)를 표시합니다. 그런 다음 이 스크립트는 LDAP 서버에 부트스트랩 서비스 트리를 만들어 매핑 파일을 저장합니다. 누구든지 다음 명령을 실행하여 스크립트를 실행할 수 있습니다.

```
./createServiceTree
```

이 스크립트는 사용자에게 다음 정보를 확인합니다.

- 호스트 이름(기본값: localhost): LDAP 서버의 호스트 이름
- 포트 번호(기본값: 389): LDAP 서버의 포트 번호
- 기본 DN: LDAP 리포지토리의 기본 DN
- 사용자 DN(기본값: cn=Directory Manager): 기본 DN에 새 항목을 만들 수 있는 권한을 가진 사용자의 전체 DN
- 암호: 해당 사용자 암호

다음 DN을 가진 항목이 만들어집니다.

```
ou=ApocRegistry,ou=default,ou=OrganizationConfig,ou=1.0,ou=ApocService,ou=services,<baseDN>
```

이 항목에는 두 매핑 파일의 내용이 채워집니다.

위에서 설명했듯이 deployApoc 스크립트는 설치 디렉토리, 레이아웃 및 스키마 확장 절차가 Sun Java System Directory Server와 거의 일치하는 LDAP 서버를 가정하여 작업을 수행합니다. 다른 디렉토리를 사용하려면 createServiceTree 스크립트를 실행하기 전에 스키마를 수동으로 확장해야 합니다.

OpenLDAP 및 ActiveDirectory 사용에 대한 자세한 내용은 부록 C를 참조하십시오.

이렇게 만들어진 트리는 엔티티에 연결된 구성 데이터를 보관하는 트리와 일치하며 Sun Java System Identity Server에서 서비스 관리에 사용되는 트리 구조와 연결됩니다.

추가 고려 사항

Configuration Manager 프레임워크에서는 데스크탑에서 지정된 사용자 또는 호스트 식별자에 연결된 전체 DN을 식별하기 위해 읽기 및 검색 권한이 있는 LDAP 서버 연결을 구성할 수 있어야 합니다. 따라서 익명 연결을 허용하도록 리포지토리를 구성하거나 읽기 및 검색 액세스 권한이 있는 특수 사용자를 만들어야 합니다.

관리 응용 프로그램은 엔티티로 매핑되는 항목 아래에 해당 엔티티의 구성 데이터를 보관할 서비스 트리를 만듭니다. 이 때문에 관리 목적으로 사용되는 사용자 항목에는 관리하는 항목에 하위 항목을 만들 수 있는 권한이 있어야 합니다.

프레임워크에서 데스크탑 클라이언트 사용자는 익명과 GSSAPI의 두 가지 방법으로 인증될 수 있습니다. 익명 방법을 사용할 경우 데스크탑 클라이언트가 LDAP 서버에서 데이터를 검색할 때 자격 증명을 제공하지 않으므로 전체 리포지토리에서 읽기 및 검색에 대한 익명 액세스를 허용해야 합니다. GSSAPI 방법(커버로스 인증)을 사용하려면 "Sun Java System Directory Server Administration Guide"의 "Implementing Security"에 설명된 것처럼 LDAP 서버를 구성해야 합니다.

Sun™ Web Console

Sun Web Console은 Sun Microsystems의 공통 웹 기반 시스템 관리 솔루션으로 설계되었습니다. 사용자가 이 콘솔을 통해 액세스하는 시스템 관리 응용 프로그램은 모두 동일한 사용자 인터페이스를 제공합니다.

다양한 이유로 인해 콘솔은 웹 모델을 기반으로 설계되었지만, 주된 이유는 시스템 관리자가 웹 브라우저를 사용하여 시스템 관리 응용 프로그램에 액세스할 수 있도록 하기 위한 것입니다.

Sun Web Console에서 제공하는 기능은 다음과 같습니다.

- 공통 인증 및 권한 부여
- 공통 로깅
- 동일한 HTTPS 기반 포트를 통해 모든 시스템 관리 응용 프로그램에 액세스할 수 있는 단일 진입점
- 공통된 모양 및 색감

콘솔의 가장 큰 이점은 관리자가 한 번만 로그인하면 콘솔 내에서 모든 응용 프로그램을 사용할 수 있다는 것입니다.

시스템 요구 사항

Sun Web Console은 다양한 클라이언트 및 서버 운영 체제와 여러 브라우저를 지원합니다.

클라이언트

- Solaris™ 8 이상에서 실행되는 Netscape™ 4.7x, 6.2x 및 7.x
- Windows 98, 98 SE, ME, 2000 및 XP에서 실행되는 Netscape 4.7x, 6.2x 및 7.x
- Windows 98, 98 SE, ME, 2000 및 XP에서 실행되는 Internet Explorer 5.x 및 6.x
- Linux 및 Solaris에서 실행되는 Mozilla

서버

- Solaris 8 이상
- Red Hat 8 이상, Red Hat Enterprise Linux 2.1
- SuSE Linux 2.1 이상
- J2SE™ 버전 1.4.1_03 이상

- 서버에서 J2SE 1.4.1 이전 버전이 검색되면 설치 프로그램에서 Java Desktop System Management Tools CD에 있는 J2SE 버전을 사용하여 설치를 업그레이드하라는 메시지를 표시합니다.
- Tomcat: 4.0.3 이상
Tomcat은 Java Desktop System Management Tools CD에 포함되어 있습니다.

Sun Web Console 설치

Sun Web Console을 설치하기 전에 먼저 본 설명서의 부록 A에서 패키지 요약과 알려진 문제점을 읽어 보십시오.

Java Desktop System Management Tools CD에서는 Solaris SPARC®(버전 8 이상) 및 Linux 운영 체제용 Sun Web Console의 설치 바이너리를 사용할 수 있습니다.

Sun Web Console을 설치하려면 다음 작업을 수행합니다.

1. Java Desktop System Management Tools CD에서 콘솔이 설치되는 운영 체제의 Sun Web Console 디렉토리로 변경합니다.

Linux 시스템에 설치하려면 `/linux/swc`로 변경하고 Solaris SPARC에 설치하려면 `/solsparc/swc`로 변경합니다.

2. `./setup`을 입력합니다.

기본적으로 Sun Web Console은 설치 로그 파일을 만들지 않습니다. "logfile"이라는 설치 로그를 만들려면 `./setup | tee logfile`을 입력합니다.

주: `setup`을 실행하면 웹 콘솔 설치 및 구성 작업이 대부분 자동으로 수행됩니다. Sun Web Console의 `setup` 응용 프로그램에 대한 자세한 내용은 부록 A를 참조하십시오.

3. Sun Web Console을 현지화하려면 각 언어에 대해 두 개의 패키지를 추가로 설치해야 합니다. 아래 표에서 해당 언어의 패키지 이름을 확인하고 다음 중 하나를 수행합니다.
 - Solaris를 사용하는 경우 `pkgadd -d path/pkgname.pkg pkgname`을 입력합니다. 여기서 `pkgname`은 추가할 언어 패키지의 이름입니다.
 - Linux를 사용하는 경우 `rpm -i path/pkgname<...>.rpm`을 입력합니다. 여기서 `pkgname`은 추가할 언어 패키지의 이름입니다.

패키지 이름	설명
SUNWcmcon, SUNWcmctg	중국어 간체 Sun(TM) Web Console 2.0
SUNWdmcon, SUNWdmctg	독일어 Sun(TM) Web Console 2.0
SUNWemcon, SUNWemctg	스페인어 Sun(TM) Web Console 2.0
SUNWfmcon, SUNWfmctg	프랑스어 Sun(TM) Web Console 2.0
SUNWhmcon, SUNWhmctg	중국어 번체 Sun(TM) Web Console 2.0
SUNWimcon, SUNWimctg	이탈리아어 Sun(TM) Web Console 2.0
SUNWjmcon, SUNWjmctg	일본어 Sun(TM) Web Console 2.0
SUNWkmcon, SUNWkmctg	한국어 Sun(TM) Web Console 2.0

패키지 이름	설명
SUNWsmcon, SUNWsmctg	스웨덴어 Sun(TM) Web Console 2.0

콘솔 실행

일반적으로 Sun Web Console 서버를 중지했다가 다시 시작하기만 하면 새 응용 프로그램을 등록할 수 있습니다.

처음 Sun Web Console을 시작하는 경우 먼저 Configuration Manager 설치가 완료되었는지 확인합니다.

- Sun Web Console을 시작하려면 `smcwebserver start`를 입력합니다.
- Sun Web Console을 중지하려면 `smcwebserver stop`을 입력합니다.
- Sun Web Console에 액세스하려면 브라우저에 다음 URL을 입력합니다.
`https://<hostname>.<domainname>:6789`

Sun Web Console은 기본적으로 Unix 기반 인증과 역할 기반 액세스 제어(RBAC)를 지원합니다. 하지만 LDAP 인증과 같은 다른 인증 메커니즘을 구성할 수도 있습니다.

주: 기본 세션 시간 제한은 15분입니다. `smreg` 명령을 사용하여 시간 제한을 구성할 수 있습니다. 예를 들어, 시간 제한을 5분으로 설정하려면 `smreg add -p -c session.timeout.value=5`를 입력합니다.

Sun Web Console 명령에 대한 자세한 내용은 `smcwebserver` 및 `smreg` 설명서 페이지를 참조하십시오.

Sun Web Console 제거

Sun Web Console을 제거하려면 `/usr/lib/webconsole/setup -u`를 실행합니다.

주: `/usr/lib/webconsole` 디렉토리나 관련 하위 디렉토리에서는 이 명령을 실행하지 마십시오. 그렇지 않으면 `pkgrm`이 실패합니다.

Sun Web Console 포트 정보

Configuration Manager는 다음 Sun Web Console 포트를 사용합니다.

- 8005: 서비스 종료
- 6789: https 액세스

이 두 포트는 `/etc/opt/webconsole/server.xml`에서 변경할 수 있습니다. 포트를 변경한 다음 `/usr/sbin/smcwebserver restart`를 사용하여 Sun Web Console을 다시 시작합니다.

Sun Java™ Desktop System Configuration Manager, 릴리스 1

Configuration Manager는 Sun Web Console에서 실행되는 관리 도구를 제공합니다. 관리자는 웹 기반 사용자 인터페이스를 통해 조직 계층 구조를 이동하면서 데스크탑 응용 프로그램에 대한 정책을 정의할 수 있습니다. 계층 구조에서 조직, 역할, 사용자, 도메인, 호스트 등 각 항목에 대해 정책을 정의할 수 있습니다. Configuration Manager는 다양한 구성 템플리트를 사용하여 Gnome, Mozilla, StarSuite 및 Evolution과 같은 데스크탑 응용 프로그램별 설정을 표시합니다.

Configuration Manager 설치

Configuration Manager를 설치하려면 Sun Web Console이 이미 설치되어 작동하고 있어야 합니다.

1. Java Desktop System Management Tools CD에서 해당 Configuration Manager 디렉토리로 변경합니다.

Linux 시스템에 설치하려면 `/linux/apoc`로 변경하고 Solaris SPARC에 설치하려면 `/solsparc/apoc`로 변경합니다.

2. `./setup`을 입력합니다.
3. LDAP 서버의 호스트 이름을 입력합니다.
기본 이름은 `localhost`입니다.
4. LDAP 서버의 포트 번호를 입력합니다(기본값: 389).
5. LDAP 리포지토리의 기본 DN을 입력합니다.
6. 사용자 엔티티를 식별하는 데 사용되는 개체 클래스 이름을 입력합니다. 기본 개체 클래스는 `inetorgperson`입니다.

자세한 내용은 LDAP 서버 장의 조직 매핑 절을 참조하십시오.

7. 전체 LDAP 리포지토리에서 고유한 속성 이름을 입력합니다. 기본 속성은 `uid`입니다.

자세한 내용은 LDAP 서버 장의 조직 매핑 절을 참조하십시오.

8. LDAP 서버에서 쿼리를 수행해야 하는 사용자의 전체 DN을 입력합니다.

읽기 및 검색 액세스 권한을 가진 임의의 전체 DN을 사용합니다. 익명 액세스를 허용하려면 이 필드를 비워 두십시오.

9. LDAP 액세스 권한이 지정된 사용자 암호를 입력합니다.

LDAP 서버에 대해 익명 액세스를 설정한 경우에는 이 단계를 무시합니다.

LDAP을 통해 사용자를 인증할 수 있도록 설치 도중 다른 로그인 모듈이 Sun Web Console에 추가됩니다.

설치가 끝나면 Sun Web Console이 자동으로 다시 시작되어 Configuration Manager에 액세스할 수 있습니다.

주: 언제든지 `/usr/share/webconsole/apoc/configure` 스크립트를 사용하여 이전의 Configuration Manager 설정을 수정할 수 있습니다. 예를 들어, Configuration Manager를 다시 설치하지 않아도 이 스크립트를 사용하여 다른 LDAP 서버로 변경할 수 있습니다.

Configuration Manager 실행

1. Configuration Manager에 액세스하려면 브라우저에 다음 URL을 입력합니다.

`https://<hostname>.<domainname>:6789`

2. 프롬프트에 기존 LDAP 사용자의 사용자 이름(uid)과 암호를 입력합니다.

Sun Web Console이 열립니다.

3. 콘솔 창에서 **Sun Java™ Desktop System Configuration Manager**, 릴리스 1을 클릭합니다.

주: Sun Web Console 시작 페이지를 건너뛰고 바로 Configuration Manager로 이동하려면 브라우저에 다음 URL을 입력합니다.

`https://<hostname>.<domainname>:6789/apoc`

Configuration Manager 제거

Sun Web Console에서 Configuration Manager를 제거하려면 Java Desktop System Management Tools CD에서 해당 Configuration Manager 디렉토리로 변경한 다음 `./setup -u`를 실행합니다.

주: Configuration Manager를 제거하면 LDAP 로그인 모듈도 Sun Web Console에서 제거됩니다.

데스크탑 구성 요소

Configuration Manager에서 구성 데이터에 액세스하려면 데스크탑 클라이언트에 Sun Java™ Desktop System Configuration Agent가 설치되어 있어야 합니다. Configuration Agent는 원격 구성 데이터 리포지토리 및 어댑터와 통신하고 데이터를 특정 구성 시스템에 통합합니다. 현재 지원되는 구성 시스템은 GConf, Mozilla Preferences 및 StarSuite Registry입니다.

이러한 구성 요소는 모두 Java Desktop System의 일부로 제공되고 설치됩니다.

데이터 액세스/사용자 인증

Configuration Agent는 데스크탑 사용자의 로그인 ID에 따라 LDAP 서버로부터 정보를 검색합니다. 조직 매핑 파일의 User/UniqueIdAttribute 설정은 로그인 ID를 LDAP 서버의 사용자 엔티티로 매핑합니다. Configuration Agent는 호스트 이름이나 IP 주소와 같은 호스트 관련 정보도 검색합니다. 이 정보는 조직 매핑 파일의 Host/UniqueIdAttribute 설정을 통해 LDAP 서버의 호스트 엔티티로 매핑됩니다.

LDAP 서버에 액세스하는 데는 익명과 GSSAPI의 두 가지 방법이 있습니다. 익명 액세스를 사용하는 경우 데스크탑에서 별도의 작업을 수행할 필요가 없습니다. GSSAPI 방법을 사용하려면 데스크탑에서 커버로스 자격 증명을 획득해야 합니다. 커버로스 자격 증명 획득과 사용자 로그인을 통합하려면 Java Desktop System 호스트에서 pam_krb5 모듈을 설치 및 구성해야 합니다.

Java Desktop System CD의 /usr/share/doc/packages/pam_krb5/README.SuSE 디렉토리에서 pam 모듈에 대한 예제 구성을 확인할 수 있습니다. gdm을 사용하여 커버로스와 사용자 로그인을 통합할 수도 있습니다. 예를 들어, 다음 /etc/pam.d/gdm 파일을 사용합니다.

```

#%PAM-1.0
auth    required    pam_unix2.so  nullok #set_secrcp
auth    optional    pam_krb5.so  use_first_pass missing_keytab_ok
ccache=SAFE putenv_direct
account required    pam_unix2.so
password required    pam_unix2.so  #strict=false
session required    pam_unix2.so  # trace or none
session required    pam_devperm.so
session optional    pam_console.so

```

Configuration Agent

Configuration Agent는 apoc 패키지에 포함되어 있습니다. 해당 RPM을 설치하면 이 API에 필요한 파일이 설치되어 inetd에 등록됩니다. RPM은 수동이나 Java Desktop System 설치를 통해 설치할 수 있습니다.

부트스트랩 정보

원격 구성 데이터에 액세스하려면 Configuration Agent에 LDAP 서버 위치를 제공해야 합니다. YaST2 구성 도구인 autoYaST를 사용하거나 /opt/apoc/lib 디렉토리의 policymgr.properties 등록 정보 파일을 수동으로 편집하여 이 위치를 추가할 수 있습니다. YaST2를 사용하는 경우 Network/Advanced 섹션에 이 데이터를 추가할 수 있습니다.

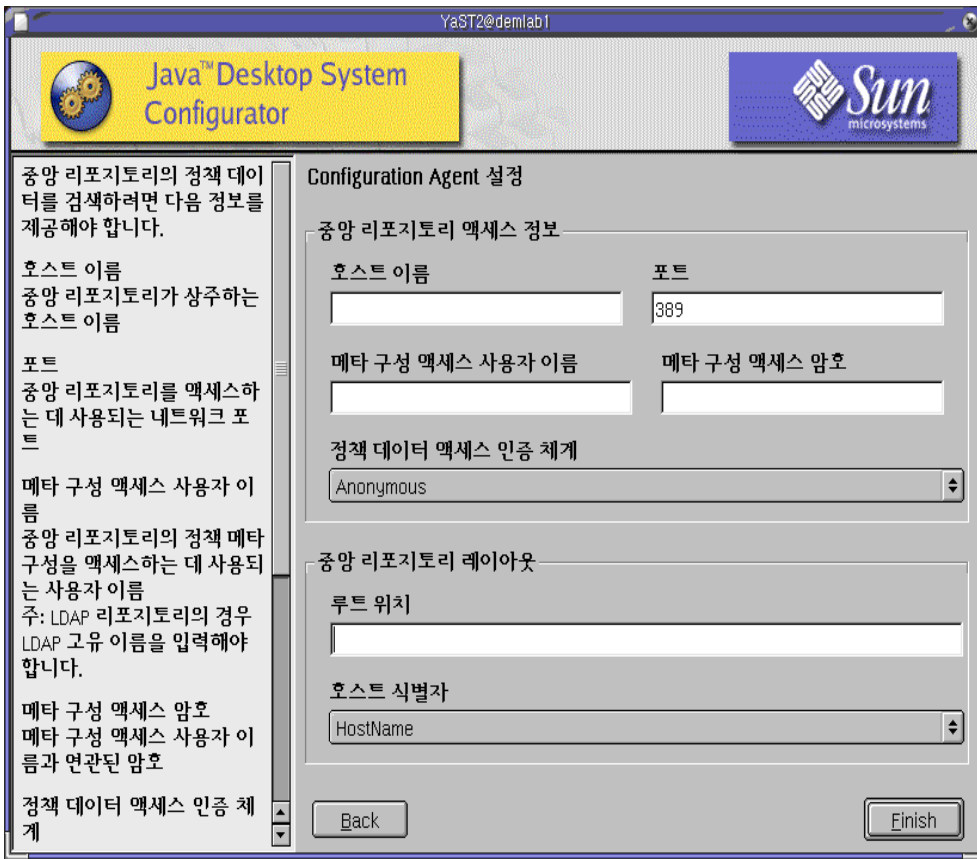


그림 1 YaST의 Java Desktop System Configuration Agent

Configuration Agent를 실행하려면 다음 정보가 필요합니다.

- 호스트 이름(Server): LDAP 서버의 호스트 이름
- 포트(Port): LDAP 서버의 포트 번호
- 메타데이터 액세스 사용자 이름(AuthDn): 리포지토리에 대한 읽기 및 검색 액세스 권한을 가진 사용자의 전체 DN

주: 디렉토리에서 익명 액세스를 허용하려면 이 설정을 비워둘 수 있습니다.

- 메타데이터 액세스 암호>Password): 등록된 LDAP 사용자 암호

주: 디렉토리에서 익명 액세스를 허용하려면 이 설정을 비워둘 수 있습니다.

- 정책 데이터 액세스 인증 체계(AuthType): LDAP 서버에서 사용자를 인증하는 방법에 따라 익명 또는 GSSAPI 방법을 사용할 수 있습니다.
- 루트 위치(BaseDn): LDAP 리포지토리의 기본 DN
- 호스트 식별자(HostIdentifier): HostName 또는 IPAddress가 지정될 수 있으며 호스트를 식별하는 데 사용되는 LDAP 속성 내용과 일치하도록 설정해야 합니다. 이 속성은 매핑 파일에서 Host/UniqueIdAttribute로 정의됩니다.
- 연결 시간 제한(Connect Timeout): LDAP 서버에 대한 연결을 시도한 후 시간 초과가 발생하기까지 경과되는 시간(초)을 나타냅니다. 기본값은 1초입니다.

주: 부트스트랩 및 작동 설정을 변경할 때마다 Configuration Agent를 다시 시작해야 합니다.

데스크탑에서 Configuration Agent를 다시 시작하려면 관련된 클라이언트 응용 프로그램이 실행되고 있지 않은지 확인하고 루트로 로그인하여 `/opt/apoc/bin/apocd restart` 명령을 입력합니다.

작동 설정

로컬 또는 원격으로 Configuration Agent의 작동 설정을 구성할 수 있습니다. 로컬에서 설정을 구성하려면 `/opt/apoc/lib` 디렉토리의 `apocd.properties` 파일을 편집해야 합니다. 원격으로 설정을 구성하려면 Configuration Manager에서 Configuration Agent 정책을 사용합니다. 등록 정보 파일에서 구성할 수 있는 설정은 다음과 같습니다.

- **DaemonPort:** Configuration Agent가 데스크탑 클라이언트에서 보내는 통신을 수신하는 포트
- **MaxClientThreads:** 동시에 처리할 수 있는 최대 클라이언트 요청 수
- **MaxClientConnections:** 최대 클라이언트 연결 수
- **MaxRequestSize:** 최대 클라이언트 요청 크기
- **DaemonChangeDetectionInterval:** 이 구성 설정 목록에 대한 변경 감지 간격(분)
- **ChangeDetectionInterval:** 클라이언트 구성 데이터에 대한 변경 감지 간격(분)
- **GarbageCollectionInterval:** 로컬 구성 데이터베이스의 가비지 컬렉션 간격(분)
- **TimeToLive:** 오프라인 상태가 아닌 구성 데이터가 로컬 데이터베이스에 보관되는 시간(분)
- **LogLevel:** 에이전트 로그 파일에 기록되는 세부 정보 수준

DaemonPort 설정은 로컬에서만 수정할 수 있으며 에이전트를 다시 시작해야만 변경 사항이 적용됩니다. 다른 설정은 에이전트 구성에 대한 다음 변경 감지 시 모두 적용됩니다. LogLevel에 지정된 로깅 수준 값은 Java Logger 수준과 일치해야 합니다. 로깅 수준을 가장 높은 것부터 순서대로 나열하면 심각, 경고, 정보, 구성, 자세히, 더 자세히, 가장 자세히 순입니다.

구성 데이터 변경 사항 전파

작동 설정 절에 설명된 ChangeDetectionInterval 설정을 사용하여 원하는 대로 원격 구성 데이터 변경 사항을 클라이언트측 응용 프로그램으로 전파할 수 있습니다. 이 설정에 지정되는 값은 원격 변경 사항이 클라이언트 응용 프로그램에서 적용될 때까지의 최대 지연 시간(분)입니다.

ChangeDetectionInterval에 작은 값을 지정하면 Configuration Agent와 LDAP 서버 작업이 증가하므로 설정 값을 변경할 때는 주의해야 합니다. 예를 들어, 초기 배포 단계에서 클라이언트 응용 프로그램에 대한 원격 구성의 영향을 쉽게 테스트하기 위해 이 값을 1분으로 설정했다가 테스트가 끝나면 다시 이 설정을 원래 값으로 되돌릴 수 있습니다.

Configuration Agent 포트 정보

Configuration Agent는 다음 두 포트를 사용합니다.

1. 데몬 포트(기본: 38900): 클라이언트 응용 프로그램과 통신하는 데몬에서 사용하는 포트
2. 데몬 관리 포트(기본: 38901): 데몬과 통신할 때 데몬 컨트롤러 프로그램인 apocdct1에서 사용하는 포트

데몬 포트 변경:

데몬 포트를 변경하려면 데몬의 apocd.properties 파일에 있는 DaemonPort 속성과 /etc/services 및 /etc/inetd.conf에 있는 apocd 항목을 수정해야 합니다. 그런 다음 데몬을 다시 시작하고 inetd를 다시 로드합니다.

데몬 관리 포트 변경:

데몬 관리 포트를 변경하려면 데몬의 apocd.properties 파일에 있는 DaemonAdminPort 속성을 수정해야 합니다. 그런 다음 데몬을 다시 시작합니다.

GConf 어댑터

GConf 어댑터는 apoc-adapter-gconf 패키지에 포함되어 있습니다. 해당 RPM에서 어댑터를 설치하면 /etc/gconf/2/path의 GConf 데이터 원본 경로가 업데이트되어 Configuration Manager 원본을 포함합니다. 이전 경로는 /etc/gconf/2/path.apocBackup에 백업됩니다. 이전 경로가 사용자 정의 데이터 원본을 참조하는 경우 경로를 업데이트할 때 기본 경로를 새로 설치된 Manager 경로로 변경해야 합니다. 어댑터는 다음 두 가지 데이터 원본을 제공합니다.

- "apoc:readonly": 보호되지 않은 정책 설정에 대한 액세스를 제공합니다. 이 데이터 원본은 사용자 설정 뒤, 로컬 기본값 앞에 삽입합니다.
- "apoc:readonly:mandatory@": 보호된 정책 설정에 대한 액세스를 제공합니다. 이 데이터 원본은 로컬 필수 설정 뒤, 사용자 설정 앞에 삽입합니다.

Mozilla 어댑터

Mozilla 어댑터는 mozilla-apoc-integration 패키지에 포함되어 있습니다. 해당 RPM에서 어댑터를 설치하면 필요한 파일이 기존 Mozilla 설치에 추가되어 자동으로 등록됩니다.

StarSuite 어댑터

StarSuite 어댑터는 표준 StarSuite 설치에 포함되어 있으며 특별한 수정 없이 정책 구성 데이터에 액세스할 수 있도록 지원합니다.

부록 A – Sun Web Console

알려진 문제점

보안

사용자가 인식하지 못하는 동안 특정 사용자 작업에 의해 세션이 활성 상태로 방치될 수 있습니다. 예를 들어, 사용자가 브라우저 창을 닫아도 Sun Web Console에서 자동으로 로그아웃되지 않습니다. 응용 프로그램 창을 닫기 전에 Sun Web Console을 사용하여 명시적으로 세션에서 로그아웃해야 합니다.

설치 스크립트 사용

개요: `setup [-h] | [-n] | [-d <ver>,<arch>[,client1,client2,...]] [-u [-f]]`

-h = 사용법을 인쇄합니다.

-n = 설치 후 서버를 시작하지 않습니다.

-u = Sun Web Console을 제거합니다.

-f = 설치 응용 프로그램과 함께 설치된 Tomcat 및 Java 1.4 패키지를 제거합니다. 이 매개 변수는 -u 매개 변수와 함께 사용해야 합니다.

사용 가능한 설치 매개 변수에 대한 자세한 내용을 보려면 `setup -h`를 실행합니다.

Sun Web Console을 제거하려면 `/usr/lib/webconsole/setup -u`를 실행합니다.

주: `/usr/lib/webconsole` 디렉토리나 관련 하위 디렉토리에서는 이 명령을 실행하지 마십시오. 그렇지 않으면 `pkgrm`이 실패합니다.

Sun Web Console 패키지

Solaris 패키지

패키지 이름	설명
SUNWmctag	Sun Web Console UI 태그 라이브러리
SUNWmcon	Sun Web Console
SUNWmcos	Sun Web Console을 위한 공통 Solaris 서비스
SUNWmcosx	Sun Web Console을 위한 Solaris 릴리스별 서비스
SUNWmconr	Sun Web Console 루트
SUNWjato	Sun One Application Framework 런타임
SUNWtcatu	Tomcat

Linux RPM

패키지 이름	설명
SUNWmctag	Sun Web Console UI 태그 라이브러리
SUNWmcon	Sun Web Console
SUNWmcos	Sun Web Console을 위한 공통 Linux 서비스
SUNWmcosx	Sun Web Console을 위한 Linux 릴리스별 서비스
SUNWmconr	Sun Web Console 루트
SUNWjato	Sun One Application Framework 런타임
SUNWtcatu	Tomcat

부록 B – Configuration Manager

Configuration Manager 패키지

Solaris 패키지

패키지 이름	설명
SUNWapm	Configuration Manager
SUNWapmca	Configuration Agent 템플릿
SUNWapmev	Evolution 템플릿
SUNWapmgo	Gnome 템플릿
SUNWapmmo	Mozilla 템플릿
SUNWapmso	StarSuite 템플릿

Linux RPM

패키지 이름	설명
apoc-manager	Configuration Manager
apoc-agent-templates	Configuration Agent 템플릿
apoc-evolution-templates	Evolution 템플릿
apoc-gnome-templates	Gnome 템플릿
apoc-mozilla-templates	Mozilla 템플릿
apoc-starsuite-templates	StarSuite 템플릿

부록 C

Configuration Manager와 함께 OpenLDAP 서버 사용

OpenLDAP 서버를 Configuration Manager 데이터의 리포지토리로 사용하려면 서버의 스키마가 구성 데이터를 저장하는 데 사용하는 개체 클래스 및 속성 기능으로 확장되어야 합니다. apoc.schema라는 사용자 정의 스키마 파일은 Java Desktop System Management Tools CD에서 제공되는 Configuration Manager 배포 도구의 openldap 하위 디렉토리에 있습니다.

이 파일을 OpenLDAP 구성 디렉토리(/etc/openldap)의 schema 하위 디렉토리에 복사해야 하며 해당 디렉토리에 있는 slapd.conf 파일에 포함시켜 OpenLDAP 스키마에 추가해야 합니다. 이 작업은 해당 파일에서 나타나는 스키마 시퀀스 맨 끝에 /etc/openldap/schema/apoc.schema라는 행을 삽입하여 수행할 수 있습니다. OpenLDAP 서버의 스키마 확장에 대한 자세한 내용은 서버의 설명서를 참조하십시오.

OpenLDAP 데이터베이스가 구성 데이터를 저장할 수 있도록 하려면 Configuration Manager와 함께 제공된 배포 도구를 사용해야 합니다. 이전 설치 단계에서 이미 확장된 스키마의 경우 createServiceTree 스크립트만 실행하면 됩니다. 스크립트는 어떤 사용자라도 반드시 ./createServiceTree 명령을 사용하여 배포 도구 디렉토리에서 시작해야 합니다. 스크립트는 이 문서의 배포 도구 절에서 설명한 대로 사용자에게 OpenLDAP 데이터베이스에 대한 정보를 표시합니다. OpenLDAP에서 일반적인 개체 클래스 및 속성 기능을 사용하는 기본 매핑 파일은 배포 도구의 openldap 하위 디렉토리에서 제공됩니다. 파일 이름은 OrganisationalMapping이며 createServiceTree를 시작하기 전에 주 배포 도구 디렉토리에서 동일한 이름을 사용하여 파일에 복사하여 배포할 수 있습니다.

Configuration Manager 에이전트는 데이터를 요구하나 비밀번호는 요구하지 않는 사용자 DN을 제공하여 OpenLDAP 서버에 익명으로 연결을 시도합니다. 이 익명 인증 모드는 OpenLDAP 서버의 일부 릴리스에서는 기본으로 비활성화되어 있을 수 있습니다. 이런 경우 반드시 OpenLDAP 구성 디렉토리(/etc/openldap)에 있는 slapd.conf 파일에서 정의된 공용 서버 매개 변수에 bind_anon_cred를 나타내는 행을 추가하여 활성화해야 합니다. 해당 매핑에 대한 자세한 내용은 서버의 설명서를 참조하십시오.

Configuration Manager와 함께 Active Directory Server 사용

Active Directory 서버를 Configuration Manager 데이터의 리포지토리로 사용하려면 서버의 스키마가 구성 데이터를 저장하는 데 사용하는 개체 클래스 및 속성 기능으로 확장되어야 합니다. apoc-ad.1df라는 스키마 확장 파일은 Management Tools CD에서 제공되는 Configuration Manager 배포 도구의 ad 하위 디렉토리에 있습니다. 자세한 내용은 배포 도구 절을 참조하십시오.

다음 단계를 통해 apoc-ad.1df 파일을 Active Directory 스키마로 가져와야 합니다.

1. 스키마 확장을 활성화합니다. 해당 작업을 수행하는 방법에 대한 자세한 내용은 Active Directory 설명서를 참조하십시오.

2. 명령 프롬프트에서 다음을 실행합니다.

```
ldifde -i -c "DC=Sun,DC=COM" <Base DN> -f apoc-ad-registry.1df.
```

주: <Base DN>을 Active Directory 기본 DN으로 바꿉니다.

Active Directory 서버가 구성 데이터를 저장하도록 준비하려면 반드시 배치 도구를 사용해야 합니다. 이전 설치 단계에서 이미 확장된 스키마의 경우 createServiceTree script 스크립트만 실행하면 됩니다. 스크립트는 어떤 사용자라도 반드시 ./createServiceTree 명령을 사용하여 배포 도구 디렉토리에서 시작해야 합니다. 스크립트는 사용자에게 Active Directory 데이터베이스에 대한 정보를 표시합니다. Active Directory에서 일반적인 개체 클래스 및 속성 기능을 사용하는 기본 매핑 파일은 배포 도구 디렉토리의 ad 하위 디렉토리에서 제공됩니다. 파일 이름은 OrganisationalMapping이며 createServiceTree를 시작하기 전에 주 배포 도구 디렉토리에서 동일한 이름을 사용하여 파일에 복사하여 배포할 수 있습니다.

이러한 점에서 Active Directory 서버를 Configuration Manager와 함께 사용할 수 있습니다.

Configuration Manager를 설치할 때 사용자에게 트리 읽기 권한과 함께 전체 DN 및 비밀번호를 제공합니다. 따라서 사용자가 다른 목적으로 Active Directory를 사용할 수 없습니다. 해당 사용자를 설정하는 방법에 대한 자세한 내용은 Active Directory 설명서를 참조하십시오.

또한 Configuration Manager를 실행 중인 컴퓨터가 Active Directory의 도메인 이름을 알고 있어야 합니다. 이 작업은 Active Directory 서버의 IP 주소를 도메인 이름에 매핑하는 행을 해당 컴퓨터의/etc/hosts file 파일에 추가하여 수행할 수 있습니다.

JDS (Java Desktop System) 호스트에서 구성 데이터를 검색하려면 해당 호스트에서 Active Directory의 도메인 이름을 알고 있어야 합니다. 익명으로 또는 GSSAPI를 사용하는 두 가지 방식으로 JDS 사용자를 인증할 수 있습니다.

- 익명 연결을 사용하여 인증하려면 Active Directory 서버가 모든 사람에게 읽기 권한을 부여하도록 구성되어야 합니다. 해당 작업을 수행하는 방법에 대한 자세한 내용은 Active Directory 설명서를 참조하십시오.
- GSSAPI를 사용하여 인증하려면 커beros 매개 변수를 지정하는 /etc/krb5.conf 파일이 Active Directory 영역을 정의하고 Active Directory 서버가 KDC (Key Distribution Center)로 나타나도록 수정되어야 합니다. 또한 기본 암호화 유형으로 Active Directory가 지원하는 DES 유형, 즉 des-cbc-crc 및 des-cbc-md5를 지정해야 합니다. 해당 작업을 수행하는 방법에 대한 자세한 내용은 커beros 설명서를 참조하십시오. 구성 데이터에 액세스하기 전에 Java Desktop System에 로그인한 사용자에 대한 유효한 자격 증명을 획득해야 합니다. 이 작업은 수동으로 kinit 명령을 실행하고 Active Directory에서 정의된 사용자 비밀번호를 제공하여 수행할 수 있습니다. 다른 스키마는 로그인 시 자동으로 이러한 자격 증명을 생성할 수 있습니다. 자세한 내용은 Java Desktop System 설명서를 참조하십시오.