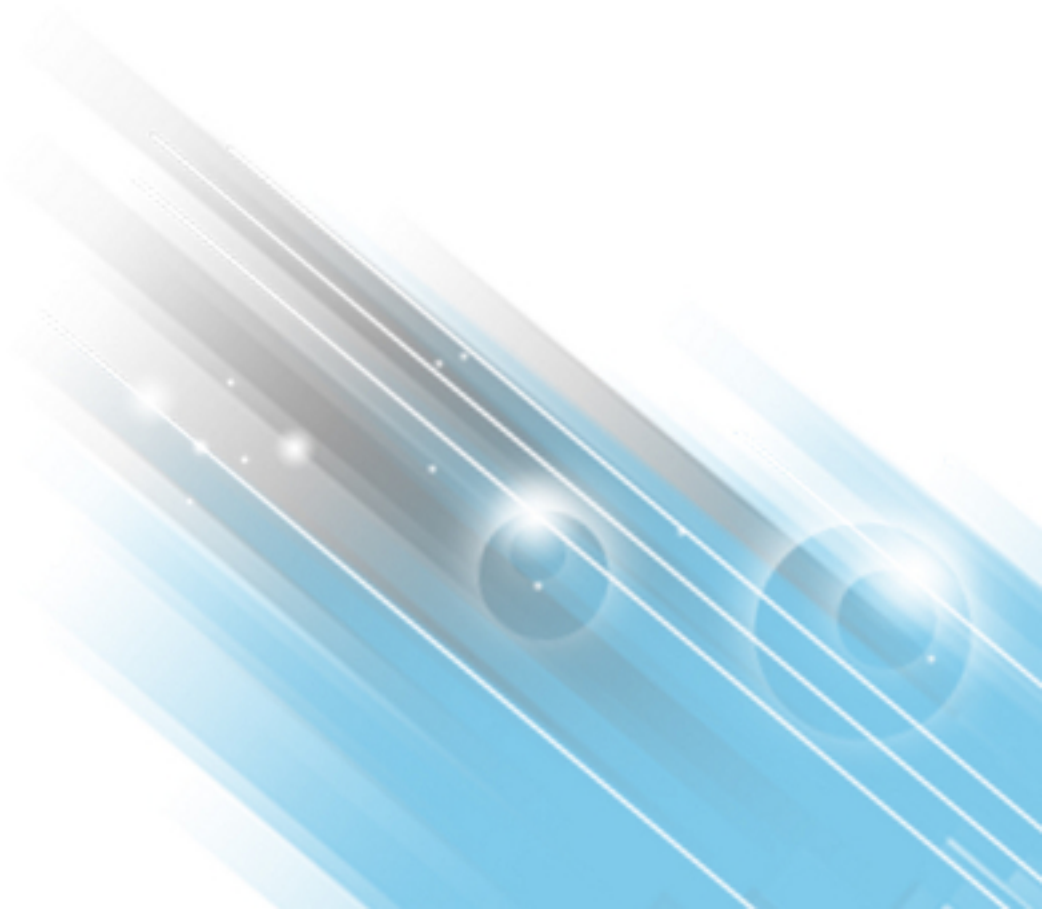
HP Universal CMDB 및 Configuration Manager

소프트웨어 버전: 10.20

강화 안내서



문서 릴리스 날짜: 2015년 1월
소프트웨어 릴리스 날짜: 2015년 1월

법적 고지 사항

보증

HP 제품 및 서비스에 대한 모든 보증 사항은 해당 제품 및 서비스와 함께 제공된 명시적 보증서에 규정되어 있습니다. 여기에 수록된 어떤 내용도 추가 보증을 구성하는 것으로 해석될 수 없습니다. HP는 여기에 수록된 기술적 또는 편집상의 오류나 누락에 대해 책임지지 않습니다.

여기에 수록된 정보는 통지 없이 변경될 수 있습니다.

제한된 권한 범례

기밀 컴퓨터 소프트웨어. 보유, 사용 또는 복사에 필요한 HP에서 제공한 유효한 라이센스. FAR 12.211 및 12.212에 의거하여 상용 컴퓨터 소프트웨어, 컴퓨터 소프트웨어 문서 및 상용 품목에 대한 기술 데이터는 벤더의 표준 상용 라이센스 하에서 미국 정부에 사용이 허가되었습니다.

저작권 고지

상표 고지 사항

Adobe ® 는 Adobe Systems Incorporated의 상표입니다.

Microsoft ® 및 Windows ® 는 Microsoft Corporation의 미국 등록 상표입니다.

Oracle 및 Java는 Oracle 및/또는 계열사의 등록 상표입니다.

UNIX ® 는 The Open Group의 등록 상표입니다.

Linux ® 는 미국 및 기타 국가에서 Linus Torvalds의 등록 상표입니다.

오픈 소스 및 타사 확인의 전체 목록을 보려면 HP Software Support Online 웹 사이트를 방문하여 HP Service Manager Open Source and Third Party License Agreements라는 제품 설명서를 검색합니다.

문서 업데이트

이 문서의 제목 페이지에는 다음 식별 정보가 포함됩니다.

- 소프트웨어 버전 번호 - 소프트웨어 버전을 나타냅니다.
- 문서 릴리스 날짜 - 문서가 업데이트될 때마다 변경됩니다.
- 소프트웨어 릴리스 날짜 - 이 소프트웨어 버전의 릴리스 날짜를 나타냅니다.

최근 업데이트를 확인하거나 문서의 최신 버전을 사용하고 있는지 확인하려면 다음 웹 사이트를 방문하십시오. **https://softwaresupport.hp.com**

이 사이트를 사용하려면 HP Passport 사용자로 등록하여 로그인해야 합니다. HP Passport ID를 등록하려면 다음 웹 사이트를 방문하십시오. **https://hpp12.passport.hp.com/hppcf/createuser.do**

또는 HP Software 지원 페이지의 맨 위에서 **Register** 링크를 클릭합니다.

적절한 제품 지원 서비스에 가입할 경우 업데이트 버전이나 새 버전도 제공됩니다. 자세한 내용은 HP 판매 담당자에게 문의하십시오.

지원

다음 위치에서 HP Software 지원 온라인 웹 사이트를 방문하십시오. **https://softwaresupport.hp.com**

이 웹 사이트에서는 연락처 정보를 비롯하여 HP 소프트웨어에서 제공하는 제품, 서비스 및 지원에 대한 자세한 내용을 확인할 수 있습니다.

온라인 지원을 통해 사용자가 스스로 문제를 해결할 수 있습니다. 또한 업무 관리에 필요한 대화식 기술 지원 도구에 신속하고 효율적으로 액세스할 수 있습니다. 소중한 지원 고객으로서 지원 웹 사이트를 통해 다음과 같은 혜택을 누릴 수 있습니다.

- 관심 있는 지식 문서를 검색할 수 있습니다.
- 지원 사례 및 개선 요청을 제출하고 추적할 수 있습니다.
- 소프트웨어 패치를 다운로드할 수 있습니다.
- 지원 계약을 관리할 수 있습니다.
- HP 고객지원센터 연락처를 조회할 수 있습니다.
- 사용 가능한 서비스에 대한 정보를 검토할 수 있습니다.
- 다른 소프트웨어 고객과의 토론에 참여할 수 있습니다.
- 소프트웨어 교육을 조사하고 등록할 수 있습니다.

대부분의 지원 영역을 이용하려면 HP Passport 사용자로 등록하여 로그인해야 합니다. 이 영역에서는 지원 계약이 필요할 수도 있습니다. HP Passport ID를 등록하려면 다음 위치로 이동하십시오.

https://hpp12.passport.hp.com/hppcf/createuser.do

액세스 수준에 대한 자세한 내용을 보려면 다음 위치로 이동하십시오.

https://softwaresupport.hp.com/web/softwaresupport/access-levels

HP Software Solutions Now에서는 HPSW 솔루션 및 통합 포털 웹 사이트에 액세스합니다. 이 사이트에서는 비즈니스 요구 사항에 맞는 HP 제품 솔루션을 탐색할 수 있고 HP 제품 간의 전체 통합 목록 및 ITIL 프로세스 목록을 제공합니다. 이 웹 사이트의 URL은 **http://h20230.www2.hp.com/sc/solutions/index.jsp**입니다.

목차

1장: 강화 소개

이 장의 내용:

강화 개요

이 섹션에서는 보안 HP Universal CMDB 응용 프로그램의 개념에 대해 소개하고 보안 구현에 필요한 계획 및 아키텍처에 대해 설명합니다. 다음 섹션의 강화 관련 내용을 설명하기 전에 이 섹션의 내용을 먼저 확인하는 것이 좋습니다.

HP Universal CMDB는 보안 아키텍처에 포함할 수 있도록 설계되어 있으므로, 보안 위협에 노출되더라도 해당 위협을 처리할 수 있습니다.

강화 지침에서는 보다 안전한(강화된) HP Universal CMDB를 구현하는 데 필요한 구성에 대해 다룹니다.

제공되는 강화 정보는 강화 절차를 시작하기 전에 강화 설정과 권장 사항을 알아 두어야 하는 HP Universal CMDB 관리자를 주 대상으로 합니다.

안전한 아키텍처를 구축하려면 HP Universal CMDB에서 리버스 프록시를 사용하는 것이 좋습니다. HP Universal CMDB에서 사용할 리버스 프록시를 구성하는 방법에 대한 자세한 내용은 "리버스 프록시 사용"(34페이지)을 참조하십시오.

이 문서에 설명된 아키텍처가 아닌 다른 유형의 보안 아키텍처를 HP Universal CMDB에서 사용해야 하는 경우에는 HP Software 지원에 문의하여 사용하기에 가장 적합한 아키텍처를 확인하십시오.

Data Flow Probe 강화에 대한 자세한 내용은 "Data Flow Probe 강화"(62페이지)를 참조하십시오.

참고:

- 강화 절차는 이 장에 설명된 지침만 구현하고 다른 문서에 설명된 기타 강화 단계는 수행하지 않는다는 가정에 따라 설명되었습니다.
- 이 강화 절차에서는 특정한 분산 아키텍처를 중심으로 설명하지만, 해당 아키텍처가 개별 사용자 조직의 요구에 가장 적합하다는 의미는 아닙니다.
- 이 문서에서는 다음 장에 포함된 절차를 HP Universal CMDB 전용 컴퓨터에서 수행한다고 가정합니다. HP Universal CMDB 외에 다른 목적으로 시스템을 사용하면 문제가 발생할 수 있습니다.
- 이 섹션에서 제공하는 강화 정보는 전산화된 시스템에서 보안 위험을 평가하는 지침으로 사용할 수 없습니다.

강화 준비

- 일반 네트워크에 대한 보안 위험/보안 상태를 평가하고, 네트워크에 HP Universal CMDB를 가장 효율적으로 통합하는 방법을 결정할 때 해당 평가 결과를 사용합니다.
- HP Universal CMDB 기술 프레임워크 및 HP Universal CMDB 보안 기능에 대해 명확하게 이해합니다.
- 모든 강화 지침을 검토합니다.
- 강화 절차를 시작하기 전에 HP Universal CMDB가 완전하게 작동하는지 확인합니다.
- 각 장에서 강화 절차 단계를 순서대로 수행합니다. 예를 들어 SSL을 지원하도록 HP Universal CMDB 서버를 구성하려는 경우 "SSL(Secure Sockets Layer) 통신 사용"(16페이지)의 내용을 확인한 후 모든 지침을 순서대로 따릅니다.
- HP Universal CMDB에서는 비밀번호를 비워 둔 기본 인증을 지원하지 않습니다. 기본 인증 연결 매개변수를 설정할 때 비밀번호를 비워 두지 마십시오.

팁: 강화 절차를 인쇄하여 강화를 구현할 때 확인합니다.

보안 아키텍처에 UCMDB 배포

HP Universal CMDB 서버를 안전하게 배포할 수 있는 권장 방법에는 다음 몇 가지가 있습니다.

- **방화벽을 사용한 DMZ 아키텍처**

  이 문서에 나와 있는 보안 아키텍처는 방화벽을 장치로 사용하는 일반 DMZ 아키텍처입니다. 이러한 아키텍처의 기본적인 개념은 HP Universal CMDB 클라이언트와 HP Universal CMDB 서버를 완전히 분리하여 클라이언트와 서버 간에 직접 액세스하지 못하도록 하는 것입니다.

- **보안 브라우저**

  Windows 환경의 Internet Explorer 및 Firefox는 Java 스크립트, 애플릿 및 쿠키를 안전하게 처리하도록 구성해야 합니다.

- **SSL 통신 프로토콜**

Secure Sockets Layer 프로토콜은 클라이언트와 서버 간의 연결을 보호합니다. SSL 연결이 필요한 URL은 HTTPS(보안 버전 Hypertext Transfer Protocol)를 사용합니다. 자세한 내용은 "SSL(Secure Sockets Layer) 통신 사용"(16페이지)을 참조하십시오.

- **리버스 프록시 아키텍처**

  HP Universal CMDB를 보다 안전하게 배포하는 권장 방법 중 하나는 리버스 프록시를 사용하는 것입니다. HP Universal CMDB는 보안 리버스 프록시 아키텍처를 완벽하게 지원합니다. 자세한 내용은 "리버스 프록시 사용"(34페이지)을 참조하십시오.

시스템 액세스

이 장의 내용:

Java JMX 액세스 강화

참고: 여기에 설명된 절차를 Data Flow Probe JMX에도 사용할 수 있습니다.

사용자 자격 증명을 제공할 때만 JMX RMI 포트에 액세스할 수 있도록 하려면 다음 절차를 수행합니다.

1. 서버의 **C:\hp\UCMDB\UCMDBServer\bin**에 있는 **wrapper.conf** 파일에서 다음을 설정합니다.

   wrapper.java.additional.16=-Dcom.sun.management.jmxremote.authenticate=true

   이 설정을 적용하려면 JMX에서 인증을 요청해야 합니다.

   - **Data Flow Probe JMX**의 경우 다음을 수행합니다.

     C:\hp\UCMDB\DataFlowProbe\bin에 있는 **WrapperGateway.conf** 및 **WrapperManager.conf** 파일에서 다음을 설정합니다.

     wrapper.java.additional.17=-Dcom.sun.management.jmxremote.authenticate=true

2. **jmxremote.password.template** 파일(위치: **C:\hp\UCMDB\UCMDBServer\bin\jre\lib\management**) 이름을 **jmxremote.password**로 바꿉니다.

   참고: Data Flow Probe JMX의 경우 이 파일이 **C:\hp\UCMDB\DataFlowProbe\bin\jre\lib\management**에 있습니다.

3. **jmxremote.password**에서 **monitorRole** 및 **controlRole** 역할에 비밀번호를 추가합니다.

예:

monitorRole QED

controlRole R&D

를 사용하면 **QED**라는 비밀번호가 **monitorRole**에 할당되고 **R&D**라는 비밀번호가 **controlRole**에 할당됩니다.

참고: jmxremote.password에는 비밀번호가 단순 텍스트로 포함되어 있으므로 소유자에게만 읽기 및 쓰기 권한을 지정해야 합니다. 파일 소유자는 UCMDB 서버를 실행하는 사용자와 같아야 합니다.

4. **jmxremote.access** 파일에서(위치: **C:\hp\UCMDB\UCMDBServer\bin\jre\lib\management**) **monitorRole** 및 **controlRole**에 액세스 권한을 할당합니다.

   예를 들면 다음과 같습니다.

   monitorRole readonly

   controlRole readwrite

   설정은 **monitorRole**에 읽기 전용 액세스를 허용하고 **controlRole**에 읽기/쓰기 액세스를 허용합니다.

   참고: Data Flow Probe JMX의 경우 이 파일이 **C:\hp\UCMDB\DataFlowProbe\bin\jre\lib\management**에 있습니다.

5. 다음과 같이 파일을 보호합니다.
   - **Windows에만 해당:** 명령줄에서 다음 명령을 실행하여 파일을 보호합니다.

     icacls jmxremote.password /grant Administrator:F

     icacls jmxremote.access /grant Administrator:R

     여기서 **<username>**은 두 파일의 속성에 표시되는 파일 소유자입니다. 이러한 파일의 속성을 열고 소유자가 올바르며 한 명만 있는지 확인합니다.

   - **Solaris 및 Linux 운영 체제의 경우:** 다음을 실행하여 비밀번호에 대한 파일 권한을 설정합니다.

     chmod 600 jmxremote.password

6. **서비스 팩 업그레이드, 서버 마이그레이션 및 재해 복구의 경우: jmxremote.access** 파일(위치 : **C:\hp\UCMDB\UCMDBServer\bin\jre\lib\management**)의 소유권을 업그레이드 또는 마이그레이션 설치를 실행하는 운영 체제 사용자로 변경합니다.

   참고:
   - Data Flow Probe JMX의 경우 이 파일이 **C:\hp\UCMDB\DataFlowProbe\bin\jre\lib\management**에 있습니다.
   - 제품을 제거하기 전에 로그인한 사용자가 편집할 수 있도록 **<UMCDB 설치 폴더>\bin\jre\lib\management\jmxremote.password**의 파일 권한을 편집합니다.

JMX 콘솔의 시스템 사용자 이름 또는 비밀번호 변경

JMX 콘솔은 시스템 사용자(다중 고객 환경의 교차 고객 사용자)를 사용합니다. 임의의 시스템 사용자 이름으로 JMX 콘솔에 로그인할 수 있습니다. 기본 이름과 비밀번호는 **sysadmin/sysadmin**입니다.

JMX 콘솔 또는 서버 관리 도구를 통해 비밀번호를 변경할 수 있습니다.

JMX 콘솔을 통해 기본 시스템 사용자 이름 또는 비밀번호를 변경하려면 다음을 수행합니다.

1. 웹 브라우저를 시작하고 주소창에 **http://localhost.<domain_name>:8080/jmx-console**을 입력합니다.
2. JMX 콘솔 인증 자격 증명을 입력합니다.
3. **UCMDB:service=Authorization Services**를 찾은 다음 링크를 클릭하여 작업 페이지를 엽니다.
4. **resetPassword** 작업을 찾습니다.
   - **userName** 필드에 **sysadmin**을 입력합니다.
   - **password** 필드에 새 비밀번호를 입력합니다.
5. **Invoke**를 클릭하여 변경 내용을 저장합니다.

서버 관리 도구를 통해 기본 시스템 사용자 이름 또는 비밀번호를 변경하려면 다음을 수행합니다.

1. **Windows: C:\hp\UCMDB\UCMDBServer\tools\server_management.bat** 파일을 실행합니다.

   Linux: /opt/hp/UCMDB/UCMDBServer/tools/ 폴더의 **server_management.sh** 파일을 실행합니다.
2. 인증 자격 증명 **sysadmin/sysadmin**을 사용하여 도구에 로그인합니다.
3. 사용자 링크를 클릭합니다.
4. 시스템 사용자를 선택하고 **로그인 사용자의 비밀번호 변경**을 클릭합니다.
5. 이전 비밀번호와 새 비밀번호를 입력하고 **확인**을 클릭합니다.

HP Universal CMDB 서버 서비스 사용자 변경

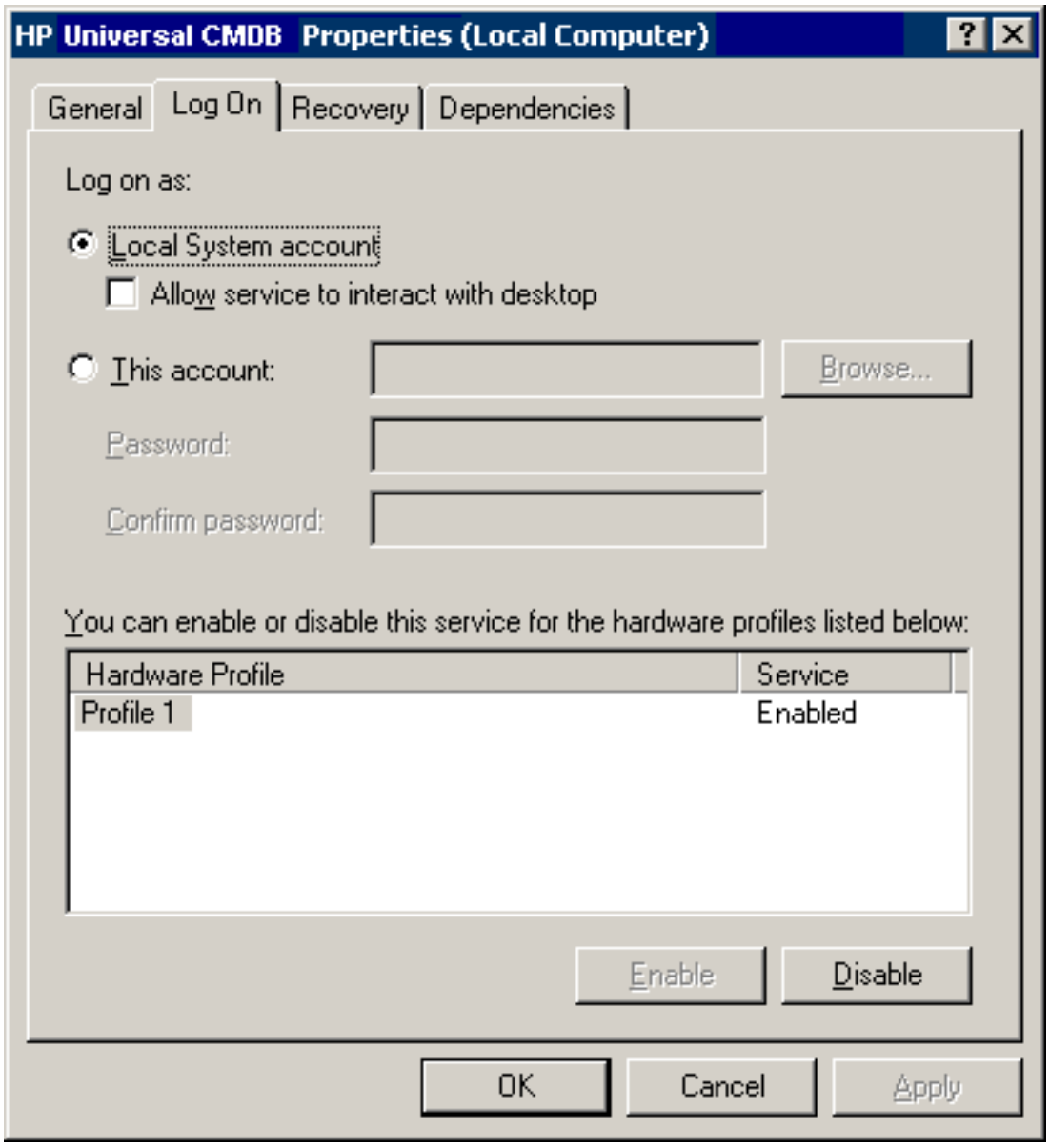
Windows 플랫폼에서 모든 HP Universal CMDB 서비스 및 프로세스를 실행하는 HP Universal CMDB 서비스는 서버 및 데이터베이스 구성 유틸리티를 실행하면 설치됩니다. 기본적으로 이 서비스는 로컬 시스템 사용자로 실행됩니다. 그러나 NTLM 인증을 사용하는 경우와 같이 다른 사용자가 서비스를 실행하도록 할당해야 하는 경우도 있습니다.

서비스를 실행하도록 할당하는 사용자에게는 다음 권한이 있어야 합니다.

- 데이터베이스 관리자가 정의한 충분한 데이터베이스 사용 권한
- 충분한 네트워크 권한
- 로컬 서버의 관리자 권한

서비스 사용자를 변경하려면 다음을 수행합니다.

1. 시작 메뉴(**시작 > 모든 프로그램 > HP UCMDB > HP Universal CMDB 서버 중지**)를 사용하거나 HP Universal CMDB 서버 서비스를 중지하여 HP Universal CMDB를 사용하지 않도록 설정합니다. 자세한 내용은 *HP Universal CMDB 관리 안내서*에서 UCMDB 서버 서비스를 시작하고 중지하는 방법을 설명하는 섹션을 참조하십시오.
2. Windows **서비스** 창에서 **UCMDB_Server**를 두 번 클릭합니다. **UCMDB_Server 속성(로컬 컴퓨터)** 대화 상자가 열립니다.
3. 로그온 탭을 클릭합니다.



4. **계정 지정**을 선택하고 컴퓨터의 유효한 사용자 목록에서 다른 사용자를 찾아 선택합니다.
5. 선택한 사용자의 Windows 비밀번호를 입력한 후 이 비밀번호를 확인합니다.
6. **적용**을 클릭하여 설정을 저장하고 **확인**을 클릭하여 대화 상자를 닫습니다.

7. 시작 메뉴(**시작 > 모든 프로그램 > HP UCMDB > HP Universal CMDB 서버 중지**)를 사용하거나 HP Universal CMDB 서버 서비스를 중지하여 HP Universal CMDB를 사용하지 않도록 설정합니다. 자세한 내용은 *HP Universal CMDB 관리 안내서*에서 UCMDB 서버 서비스를 시작하고 중지하는 방법을 설명하는 섹션을 참조하십시오.

Configuration Manager의 데이터베이스 비밀번호 암호화

CM 데이터베이스 비밀번호는 **<*Configuration_Manager_installation_directory*>\conf\database.properties** 파일에 저장됩니다. 비밀번호를 암호화하려면 기본 암호화 알고리즘이 FIPS 140-2의 표준을 준수해야 합니다.

암호화는 키를 통해 수행됩니다. 그리고 이 키 자체가 마스터 키라고 하는 다른 키를 사용하여 암호화됩니다. 두 키 모두 동일한 알고리즘을 사용하여 암호화됩니다. 암호화 프로세스에 사용되는 매개 변수에 대한 자세한 내용은 "Configuration Manager 데이터베이스 비밀번호 암호화에 사용되는 매개 변수"(13 페이지)를 참조하십시오.

주의: 암호화 알고리즘을 변경하면 이전에 암호화된 모든 비밀번호를 더 이상 사용할 수 없습니다.

데이터베이스 비밀번호의 암호화를 변경하려면 다음을 수행합니다.

1. **<*Configuration_Manager_installation_directory*>\conf\database.properties** 파일을 열고 다음 필드를 편집합니다.
   - **engineName.** 암호화 알고리즘의 이름을 입력합니다.
   - **keySize.** 선택한 알고리즘에 대한 마스터 키 크기를 입력합니다.
2. **generate-keys.bat** 스크립트를 실행하여 **<*Configuration_Manager_installation_directory*>\security\encrypt_repository** 파일을 만들고 암호화 키를 생성합니다.
3. **bin\encrypt-password.bat** 유틸리티를 실행하여 비밀번호를 암호화합니다. 사용할 수 있는 옵션을 확인하도록 **-h** 플래그를 설정합니다.
4. 비밀번호 암호화 유틸리티의 결과를 복사하여 결과로 나온 암호화를 **conf\database.properties** 파일에 붙여넣습니다.

Configuration Manager 데이터베이스 비밀번호 암호화에 사용되는 매개 변수

다음 표는 CM 데이터베이스 비밀번호 암호화에 사용하는 **encryption.properties** 파일에 포함된 매개 변수입니다. 데이터베이스 비밀번호 암호화에 대한 자세한 내용은 "Configuration Manager의 데이터베이스 비밀번호 암호화"(13페이지)를 참조하십시오.

매개 변수	설명
cryptoSource	암호화 알고리즘을 구현하는 인프라를 나타냅니다. 사용 가능한 옵션은 다음과 같습니다. • **lw.** Bouncy Castle lightweight 구현 사용(기본 옵션) • **jce.** Java Cryptography Enhancement(표준 Java 암호 기법 인프라)
storageType	키 저장소의 유형을 나타냅니다. 현재는 **이진 파일**만 지원됩니다.
binaryFileStorageName	마스터 키가 저장되는 파일의 위치를 나타냅니다.
cipherType	암호 유형입니다. 현재는 **symmetricBlockCipher**만 지원됩니다.
engineName	암호화 알고리즘의 이름입니다. 다음 옵션을 사용할 수 있습니다. • **AES.** American Encryption Standard. 이 암호화는 FIPS 140-2를 준수합니다 (기본 옵션). • **Blowfish** • **DES** • **3DES.**(FIPS 140-2 준수) • **Null**. 암호화 안 함
keySize	마스터 키의 크기입니다. 크기는 알고리즘에 의해 결정됩니다. • **AES.** 128, 192, 또는 256(기본 옵션은 256) • **Blowfish.** 0-400 • **DES.** 56 • **3DES.** 156
encodingMode	이진 암호화 결과의 ASCII 인코딩입니다. 다음 옵션을 사용할 수 있습니다. • **Base64**(기본 옵션) • **Base64Url** • **Hex**
algorithmModeName	알고리즘 모드입니다. 현재는 **CBC**만 지원됩니다.
algorithmPaddingName	사용된 패딩 알고리즘입니다.

매개 변수	설명
	다음 옵션을 사용할 수 있습니다. • **PKCS7Padding**(기본 옵션) • **PKCS5Padding**
jceProviderName	JCE 암호화 알고리즘의 이름입니다. **참고:** crytpSource가 jce일 때만 해당됩니다. lw의 경우 engineName이 사용됩니다.

2장: SSL(Secure Sockets Layer) 통신 사용

이 장의 내용:

자체 서명된 인증서를 사용하여 서버 컴퓨터에서 SSL 사용 - UCMDB

이 섹션에서는 SSL(Secure Sockets Layer) 채널을 사용하는 통신을 지원하도록 HP Universal CMDB를 구성하는 방법을 설명합니다.

1. **선행 조건**
   a. 다음 절차를 시작하기 전에 **C:\hp\UCMDB\UCMDBServer\conf\security\server.keystore**에 있는 이전 **server.keystore**를 제거합니다.
   b. HP Universal CMDB 키 저장소(JKS 유형)를 **C:\hp\UCMDB\UCMDBServer\conf\security** 폴더에 저장합니다.

2. **서버 키 저장소 생성**
   a. 자체 서명된 인증서와 일치하는 개인 키가 포함된 키 저장소(JKS 유형)를 만듭니다.
      - **C:\hp\UCMDB\UCMDBServer\bin\jre\bin**에서 다음 명령을 실행합니다.

        keytool -genkey -alias hpcert -keystore C:\hp\UCMDB\UCMDBServer\conf\security\server.keystore

        콘솔 대화 상자가 열립니다.
      - 키 저장소 비밀번호를 입력합니다. 비밀번호가 변경된 경우에는 **UCMDB:service=Security Services**에서 **changeKeystorePassword** JMX 작업을 실행합니다. 비밀번호가 변경되지 않은 경우에는 기본 **hppass** 비밀번호를 사용합니다.
      - **이름과 성은 무엇입니까?**라는 메시지가 표시되면 HP Universal CMDB 웹 서버 이름을 입력합니다. 다른 매개 변수는 조직에 맞게 입력합니다.
      - 키 비밀번호를 입력합니다. 키 비밀번호는 키 저장소 비밀번호와 같아야 합니다.

        server.keystore라는 JKS 키 저장소가 만들어집니다. 이 저장소에는 **hpcert**라는 서버 인증서가 있습니다.

   b. 자체 서명된 인증서를 파일로 내보냅니다.

   C:\hp\UCMDB\UCMDBServer\bin\jre\bin에서 다음 명령을 실행합니다.

   keytool -export -alias hpcert -keystore C:\hp\UCMDB\UCMDBServer\conf\security\server.keystore -storepass <사용자 암호> -file hpcert

3. **클라이언트의 신뢰 저장소에 인증서 저장**

   server.keystore를 생성하고 서버 인증서를 내보낸 다음, 이 자체 서명된 인증서를 사용하여 SSL을 통해 HP Universal CMDB와 통신해야 하는 모든 클라이언트에 대해 이 인증서를 클라이언트의 신뢰 저장소에 저장합니다.

   참고: server.keystore에는 서버 인증서가 하나만 있을 수 있습니다.

4. **HTTP 포트 8080을 사용하지 않도록 설정**

   자세한 내용은 "HTTP/HTTPS 포트를 사용하거나 사용하지 않도록 설정"(28페이지)을 참조하십시오.

   참고: HTTP 포트를 닫기 전에 HTTPS 통신이 작동하는지 확인합니다.

5. **서버 다시 시작**
6. **HP Universal CMDB 표시**

   UCMDB 서버가 안전한지 확인하려면 웹 브라우저에 다음 URL을 입력합니다. **https://<UCMDB 서버 이름 또는 IP 주소>:8443/ucmdb-ui**

자체 서명된 인증서를 사용하여 서버 시스템에서 SSL 활성화 - Configuration Manager

이 섹션에서는 SSL(Secure Sockets Layer) 채널을 사용하여 Configuration Manager가 인증 및 암호화를 지원하도록 구성하는 방법에 대해 설명합니다.

Configuration Manager는 응용 프로그램 서버로 Tomcat 7.0.19를 사용합니다.

1. **선행 조건** (처음으로 설치하면 관련 없음)

   다음 절차를 시작하기 전에, **<*Configuration_Manager_installation_directory*>\java\windows\x86_64\lib\security** 폴더 또는 **<*Configuration_Manager_installation_directory*>\java\linux\x86_64\lib\security** 폴더(해당되는 위치)에 이전 **tomcat.keystore** 파일이 있으면 제거합니다.

2. **서버 키 저장소 생성**

   자체 서명된 인증서와 일치하는 개인 키가 포함된 키 저장소(JKS 유형)를 만듭니다.

   - **<*Configuration_Manager_installation_directory*>\java\windows\x86_64\bin** 또는 **<*Configuration_Manager_installation_directory*>\java\linux\x86_64\bin**에서 다음 명령을 실행합니다.

     ```
     keytool -genkey -alias tomcat -keyalg RSA -keystore ..\lib\
     security\tomcat.keystore
     ```

     콘솔 대화 상자가 열립니다.

   - 키 저장소 비밀번호를 입력합니다. 비밀번호가 변경되었으면 파일에서 수동으로 변경합니다.

   - **이름과 성은 무엇입니까?**라는 메시지가 표시되면 Configuration Manager 웹 서버 이름을 입력합니다. 다른 매개 변수는 조직에 맞게 입력합니다.

   - 키 비밀번호를 입력합니다. 키 비밀번호는 키 저장소 비밀번호와 같아야 합니다.

     hpcert라는 서버 인증서를 사용하여 **tomcat.keystore**라는 JKS 키 저장소가 만들어집니다.

3. **클라이언트가 신뢰할 수 있는 저장소에 인증서 배치**

   컴퓨터의 Internet Explorer에서 클라이언트가 신뢰할 수 있는 저장소에 인증서를 추가합니다(**도구 > 인터넷 옵션 > 내용 > 인증서**). 이렇게 하지 않으면 Configuration Manager를 처음 사용하려고 시도할 때 이렇게 하라는 메시지가 나타납니다.

   > **제한**: **tomcat.keystore**에는 서버 인증서가 하나만 있을 수 있습니다.

4. **server.xml 파일 수정**

<*Configuration_Manager_installation_directory*>\servers\server-0\conf에 있는 **server.xml** 파일을 엽니다. 주석에서

Connector port="8143"

으로 시작하는 섹션을 찾습니다. 주석 문자를 제거하여 스크립트를 활성화하고 HTTPS 커넥터에 다음 특성을 추가합니다.

keystoreFile="<tomcat.keystore 파일 위치>"**(단계 2 참조)**
keystorePass="<비밀번호>"

다음 줄을 주석 처리합니다.

<Listener className="org.apache.catalina.core.AprLifecycleListener" SSLEngine="on" />

참고: HTTP 연결 포트를 차단하지 않아야 합니다. HTTP 통신을 차단하려는 경우 이 용도로 방화벽을 사용할 수 있습니다.

5. **서버 다시 시작**

   Configuration Manager 서버를 다시 시작합니다.

6. **서버 보안 확인**

   Configuration Manager 서버가 안전한지 확인하려면 웹 브라우저에 다음 URL을 입력합니다.
   https://<Configuration Manager 서버 이름 또는 IP 주소>:8143/cnc

7. Configuration Manger에서 **설정 > 응용 프로그램 관리> 메일 설정**으로 이동하고 **Configuration Manager 전체 URL**의 프로토콜과 포트를 위의 값에 따라서 변경합니다.

8. UCMDB에서 **인프라 설정 관리자 > 일반 설정**으로 이동하고 **Configuration Manager URL**의 프로토콜과 포트를 위의 값에 따라서 변경합니다.

팁: 연결 설정에 실패하면 다른 브라우저를 사용하여 시도하거나 브라우저를 최신 버전으로 업그레이드합니다.

인증 기관에서 발급한 인증서를 사용하여 서버 컴퓨터에서 SSL 사용 - UCMDB

CA(인증 기관)에서 발급한 인증서를 사용하려면 키 저장소가 Java 형식이어야 합니다. 다음 예는 Windows 컴퓨터에 대해 키 저장소의 형식을 지정하는 방법을 보여 줍니다.

1. **선행 조건**

   다음 절차를 시작하기 전에 **C:\hp\UCMDB\UCMDBServer\conf\security\server.keystore**에 있는 이전 **server.keystore**를 제거합니다.

2. **서버 키 저장소 생성**

   a. CA에서 서명한 인증서를 생성하여 Windows에 설치합니다.

   b. Microsoft 관리 콘솔(**mmc.exe**)을 사용하여 인증서를 ***.pfx** 파일(개인 키 포함)로 내보냅니다.

pfx 파일에 대한 비밀번호로 문자열을 입력합니다. 키 저장소 유형을 JAVA 키 저장소로 변환할 때 이 비밀번호를 입력하라는 메시지가 표시됩니다. 이제 **.pfx** 파일은 공개 인증서와 개인 키를 포함하며 비밀번호로 보호됩니다.

c. 만든 **.pfx** 파일을 다음 폴더에 복사합니다. **C:\hp\UCMDB\UCMDBServer\conf\security**

d. 명령 프롬프트를 열고 디렉터리를 **C:\hp\UCMDB\UCMDBServer\bin\jre\bin**으로 변경합니다.

다음 명령을 실행하여 키 저장소 유형을 **PKCS12**에서 **JAVA** 키 저장소로 변경합니다.

keytool -importkeystore -srckeystore c:\hp\UCMDB\UCMDBServer\conf\security\<pfx file name> -srcstoretype PKCS12 -destkeystore server.keystore

원본(**.pfx**) 키 저장소 비밀번호를 입력하라는 메시지가 표시됩니다. 이 비밀번호는 단계 b에서 **pfx** 파일을 만들 때 제공한 비밀번호입니다.)

e. 대상 키 저장소 비밀번호를 입력합니다. 이 비밀번호는 이전에 **changeKeystorePassword** JMX 메서드(보안 서비스)에서 정의한 비밀번호와 같아야 합니다. 비밀번호가 변경되지 않은 경우 기본 **hppass** 비밀번호를 사용합니다.

> **참고:** 원본 키 저장소 비밀번호는 대상 키 저장소 비밀번호와 같아야 합니다.

f. 인증서를 생성한 후에 HTTP 포트 8080을 사용하지 않도록 설정합니다. 자세한 내용은 "HTTP/HTTPS 포트를 사용하거나 사용하지 않도록 설정"(28페이지)을 참조하십시오.

g. **hppass** 이외의 비밀번호 또는 **.pfx** 파일에 사용했던 비밀번호를 사용한 경우에는 **changeKeystorePassword** JMX 메서드를 실행하여 키의 비밀번호가 같은지 확인합니다.

> **참고:** HTTP 포트를 닫기 전에 HTTPS 통신이 작동하는지 확인합니다.

3. **서버 다시 시작**
4. **서버 보안 확인**

UCMDB 서버가 안전한지 확인하려면 웹 브라우저에 다음 URL을 입력합니다. **https://<UCMDB 서버 이름 또는 IP 주소>:8443/ucmdb-ui**

> **주의: server.keystore**에는 서버 인증서가 하나만 있을 수 있습니다.

인증 기관에서 발급된 인증서를 사용하여 서버 시스템에서 SSL 활성화 - Configuration Manager

Configuration Manager의 경우 CA(인증 기관)에서 발급된 인증서를 사용하려면 키 저장소가 Java 형식이어야 합니다. 다음 예는 Windows 컴퓨터에 대해 키 저장소의 형식을 지정하는 방법을 보여 줍니다.

1. **선행 조건**

   다음 절차를 시작하기 전에, **<Configuration_Manager_installation_directory>\java\windows\x86_64\lib\security** 폴더 또는 **<*Configuration Manager 설치 디렉터리*>\java\linux\x86_64\lib\security** 폴더(해당되는 위치)에 이전 **tomcat.keystore** 파일이 있으면 제거합니다.

2. **서버 키 저장소 생성**

   a. CA에서 서명한 인증서를 생성하여 Windows에 설치합니다.

   b. Microsoft 관리 콘솔(**mmc.exe**)을 사용하여 인증서를 ***.pfx** 파일(개인 키 포함)로 내보냅니다.

   pfx 파일에 대한 비밀번호로 문자열을 입력합니다. 키 저장소 유형을 JAVA 키 저장소로 변환할 때 이 비밀번호를 입력하라는 메시지가 표시됩니다.
   이제 **.pfx** 파일은 공개 인증서와 개인 키를 포함하며 비밀번호로 보호됩니다.

   만든 **.pfx** 파일을 **<*Configuration_Manager_installation_directory*>\java\windows\x86_64\lib\security** 폴더에 복사합니다.

   c. 명령 프롬프트를 열고 디렉터리를 **<*Configuration_Manager_installation_directory*>\java\bin**으로 변경합니다.

   다음 명령을 실행하여 키 저장소 유형을 **PKCS12**에서 **JAVA** 키 저장소로 변경합니다.

   keytool -importkeystore -srckeystore <*Configuration_Manager_installation_directory*>\conf\security\<pfx file name> -srcstoretype PKCS12 -destkeystore tomcat.keystore

   원본(**.pfx**) 키 저장소 비밀번호를 입력하라는 메시지가 표시됩니다. 이 비밀번호는 단계 b에서 pfx 파일을 만들 때 지정한 비밀번호입니다.

3. **server.xml 파일 수정**

   <*Configuration_Manager_installation_directory*>\servers\server-0\conf에 있는 **server.xml** 파일을 엽니다. 주석에서

   Connector port="8143"

   으로 시작하는 섹션을 찾습니다. 주석 문자를 지우고 다음 두 줄을 추가하여 스크립트를 활성화합니다.

   keystoreFile="../../java/lib/security/tomcat.keystore"
   keystorePass="password" />

   다음 줄을 주석 처리합니다.

   <Listener className="org.apache.catalina.core.AprLifecycleListener" SSLEngine="on" />

   참고: HTTP 연결 포트를 차단하지 않아야 합니다. HTTP 통신을 차단하려는 경우 이 용도로 방화벽을 사용할 수 있습니다.

4. **서버 다시 시작**

   Configuration Manager 서버를 다시 시작합니다.

5. **서버 보안 확인**

   Configuration Manager 서버가 안전한지 확인하려면 웹 브라우저에 다음 URL을 입력합니다.
   https://<Configuration Manager 서버 이름 또는 IP 주소>:8143/cnc

6. Configuration Manger에서 **설정 > 응용 프로그램 관리> 메일 설정**으로 이동하고 **Configuration Manager 전체 URL**의 프로토콜과 포트를 위의 값에 따라서 변경합니다.
7. UCMDB에서 **인프라 설정 관리자 > 일반 설정**으로 이동하고 **Configuration Manager URL**의 프로토콜과 포트를 위의 값에 따라서 변경합니다.

제한: **tomcat.keystore**에는 서버 인증서가 하나만 있을 수 있습니다.

클라이언트 컴퓨터에서 SSL 사용 - UCMDB

HP Universal CMDB 웹 서버에서 사용하는 인증서가 알려진 CA(인증 기관)에서 발급한 인증서인 경우 웹 브라우저에서 추가 수행 없이 인증서의 유효성을 검사할 수 있습니다.

웹 브라우저에서 CA를 신뢰하지 않는 경우에는 전체 인증서 신뢰 경로를 가져오거나, HP Universal CMDB에서 사용하는 인증서를 명시적으로 브라우저의 신뢰 저장소로 가져와야 합니다.

다음 예는 자체 서명된 **hpcert** 인증서를 Internet Explorer에서 사용하도록 Windows 신뢰 저장소로 가져오는 방법을 보여 줍니다.

Windows 신뢰 저장소로 인증서를 가져오려면 다음을 수행합니다.

1. **hpcert** 인증서를 찾아 이름을 **hpcert.cer**로 바꿉니다.

   Windows Explorer에서 해당 파일이 보안 인증서임을 나타내는 아이콘이 표시됩니다.
2. **hpcert.cer**을 두 번 클릭하여 Internet Explorer 인증서 대화 상자를 엽니다.
3. 인증서 가져오기 마법사를 통해 인증서를 설치하여 신뢰를 사용하도록 설정하는 지침을 따릅니다.

참고: UCMDB 서버에서 발급한 인증서를 웹 브라우저로 가져오는 또 다른 방법은 UCMDB에 로그인한 다음 신뢰할 수 없는 인증서 경고가 표시될 때 인증서를 설치하는 것입니다.

클라이언트 인증서를 사용하여 SSL 활성화 - Configuration Manager

Configuration Manager 웹 서버에서 사용하는 인증서가 잘 알려진 CA(인증 기관)에서 발급된 것이면 추가 작업 없이 웹 브라우저로 인증서의 유효성을 검사할 수 있습니다.

서버 신뢰 저장소에서 신뢰하는 CA가 아닐 경우에는 CA 인증서를 서버 신뢰 저장소로 가져옵니다.

다음 예는 자체 서명 **hpcert** 인증서를 서버 신뢰 저장소(cacerts)로 가져오는 방법에 대해 설명합니다.

인증서를 서버 신뢰 저장소로 가져오려면 다음을 수행합니다.

1. 클라이언트 시스템에서 **hpcert** 인증서를 찾아 **hpcert.cer**로 이름을 바꿉니다.
2. **hpcert.cer**을 서버 컴퓨터의 **<*Configuration_Manager_installation_directory*>\java\windows\x86_64\bin** 폴더에 복사합니다.

3. 서버 시스템에서 다음 명령으로 keytool 유틸리티를 사용하여 CA 인증서를 신뢰 저장소(cacerts)로 가져옵니다.

   <*Configuration_Manager_installation_directory*>\java\bin\keytool.exe -import -alias hp -file hpcert.cer -keystore ..\lib\security\cacerts

4. **<*Configuration_Manager_installation_directory*>\servers\server-0\conf** 폴더에 있는 **server.xml** 파일을 다음과 같이 수정합니다.
   a. "server.xml 파일 수정"(21페이지)에서 설명한 대로 변경합니다.
   b. 변경 직후 HTTPS 커넥터에 다음 특성을 추가합니다.

      truststoreFile="../../java/lib/security/cacerts"
      truststorePass="changeit" />

   c. clientAuth="true"를 설정합니다.
5. "서버 보안 확인"(21페이지)에서 설명한 대로 서버 보안을 확인합니다.

클라이언트 SDK에서 SSL 사용

클라이언트 SDK와 서버 SDK 간에 HTTPS 전송을 사용할 수 있습니다.

1. 클라이언트 컴퓨터에 설치되어 있는 클라이언트 SDK가 포함된 제품에서 전송 설정을 찾아서 HTTP가 아닌 HTTPS로 구성되어 있는지 확인합니다.
2. CA 인증서/자체 서명된 공용 인증서를 클라이언트 컴퓨터에 다운로드한 다음 서버에 연결할 JRE의 **cacerts** 신뢰 저장소로 가져옵니다.

   다음 명령을 실행합니다.

   Keytool -import -alias <CA name> -trustcacerts -file <server public certificate path> -keystore <path to client jre trusted cacerts store (e.g. x:\program files\java\jre\lib\security\cacerts)>

SDK에 대해 상호 인증서 인증 사용

이 모드에서는 SSL을 사용하며, UCMDB를 통한 서버 인증과 UCMDB-API 클라이언트를 통한 클라이언트 인증이 모두 가능합니다. 서버와 UCMDB-API 클라이언트는 모두 인증을 위해 다른 엔터티로 인증서를 보냅니다.

참고: 상호 인증을 통해 SDK에서 SSL을 사용하도록 설정하는 다음 방법이 가장 안전한 방법이자 권장 통신 모드입니다.

1. UCMDB에서 UCMDB-API 클라이언트 커넥터를 강화합니다.
   a. 웹 브라우저를 시작하고 주소창에 **http://<UCMDB 컴퓨터 이름 또는 IP 주소>:8080/jmx-console**을 입력하여 UCMDB JMX 콘솔에 액세스합니다. 사용자 이름과 비밀번호(기본값: sysadmin/sysadmin)를 입력하여 로그인해야 할 수도 있습니다.

b. **UCMDB:service=Ports Management Services**를 찾은 다음 링크를 클릭하여 작업 페이지를 엽니다.

c. **PortsDetails** 작업을 찾아서 **Invoke**를 클릭합니다. HTTPS와 클라이언트 인증 포트 번호를 적어 둡니다. 기본값인 8444 포트를 사용하도록 설정해야 합니다.

d. 작업 페이지로 돌아옵니다.

e. ucmdb-api 커넥터를 상호 인증 모드에 매핑하려면 다음 매개 변수를 사용하여 **mapComponentToConnectors** 메서드를 호출합니다.

- **componentName**: ucmdb-api
- **isHTTPSWithClientAuth**: true
- 기타 모든 플래그: false

다음 메시지가 표시됩니다.

Operation succeeded. Component ucmdb-api is now mapped to: HTTPS_CLIENT_AUTH ports.

f. 작업 페이지로 돌아옵니다.

2. **ping** 구성 요소에 대해 1단계를 반복합니다.
3. UCMDB-API 클라이언트를 실행하는 JRE에 클라이언트 인증서가 포함된 키 저장소가 있는지 확인합니다.
4. 키 저장소에서 UCMDB-API 클라이언트 인증서를 내보냅니다.
5. 내보낸 UCMDB-API 클라이언트 인증서를 UCMDB 서버 신뢰 저장소로 가져옵니다.

a. UCMDB 컴퓨터에서 만들어진 UCMDB-API 클라이언트 인증서 파일을 UCMDB의 다음 디렉터리로 복사합니다.

C:\HP\UCMDB\UCMDBServer\conf\security

b. 다음 명령을 실행합니다.

```
C:\HP\UCMDB\UCMDBServer\bin\jre\bin\keytool.exe -import -v -keystore
C:\HP\UCMDB\UCMDBServer\conf\security\server.truststore -file <내보낸
UCMDB-api 클라이언트 인증서> - alias ucmdb-api
```

c. UCMDB 서버 신뢰 저장소 비밀번호(기본값: **hppass**)를 입력합니다.

d. **이 인증서를 신뢰하시겠습니까?**라는 메시지가 표시되면 **y**를 누르고 **Enter** 키를 누릅니다.

e. **인증서가 키 저장소에 추가되었습니다.**가 출력되는지 확인합니다.

6. 서버 키 저장소에서 UCMDB 서버 인증서를 내보냅니다.

a. UCMDB 컴퓨터에서 다음 명령을 실행합니다.

```
C:\HP\UCMDB\UCMDBServer\bin\jre\bin\keytool.exe -export -alias hpcert
-keystore
C:\HP\UCMDB\UCMDBServer\conf\security\server.keystore
-file C:\HP\UCMDB\conf\security\server.cert
```

b. UCMDB 서버 신뢰 저장소 비밀번호(기본값: **hppass**)를 입력합니다.

c. 다음 디렉터리에 인증서가 만들어졌는지 확인합니다.

C:\HP\UCMDB\UCMDBServer\conf\security\server.cert

7. 내보낸 UCMDB 인증서를 UCMDB-API 클라이언트 신뢰 저장소의 JRE로 가져옵니다.
8. API 클라이언트에서 사용되는 인증서의 CN(일반 이름) 필드에는 UCMDB에 있는 사용자의 이름이 포함되어야 합니다.

   이 사용자는 빈 비밀번호 및 SDK 액세스에 대한 모든 필수 권한을 가져야 합니다.

   기존 UCMDB 사용자에게 빈 비밀번호를 설정하려면 다음을 수행합니다.

   a. **JMX 콘솔 > UCMDB:service=URM Services > listResourceTypes**로 이동합니다.
   b. **Auth_USER**를 클릭합니다.
   c. 사용자를 클릭하고 XML이 로드될 때까지 기다립니다.
   d. XML에서 비밀번호를 **s39t30*tfoZXg30xd/nvJGL5is8=**로 바꿉니다.
   e. **리소스 저장**을 클릭합니다.
9. UCMDB 서버와 UCMDB-API 클라이언트를 다시 시작합니다.
10. UCMDB-API 클라이언트에서 UCMDB-API 서버로 연결하려면 다음 코드를 사용합니다.

```
UcmdbServiceProvider provider = UcmdbServiceFactory.getServiceProvider
("https", <SOME_HOST_NAME>, <HTTPS_WITH_CLIENT_AUTH_PORT_NUMBER
(default:8444>));
UcmdbService ucmdbService = provider.connect(provider.createCertificateCredentials
(<TheClientKeystore.
e.g: "c:\\client.keystore">, <KeystorePassword>), provider.createClientContext
(<ClientIdentification>));
```

UCMDB에서 CAC(스마트 카드/PKI 인증) 지원 구성

이 섹션에서는 UCMDB에 스마트 카드 인증 또는 PKI 인증(CAC) 지원을 구성하는 방법을 설명합니다.

참고: CAC 지원은 Internet Explorer 8, 9 또는 10을 사용할 때만 사용 가능합니다.

1. 루트 CA와 중간 자격 증명을 모두 UCMDB 서버 신뢰 저장소에 다음과 같이 가져옵니다.
   a. UCMDB 컴퓨터에서 인증서 파일을 UCMDB의 다음 디렉터리에 복사합니다.

   C:\HP\UCMDB\UCMDBServer\conf\security

   참고: 인증서가 Microsoft p7b 형식이면 PEM 형식으로 변환해야 할 수도 있습니다.

   b. 각 인증서에 대해 다음 명령을 실행합니다.

```
C:\HP\UCMDB\UCMDBServer\bin\jre\bin\keytool.exe -import -v -keystore
C:\HP\UCMDB\UCMDBServer\conf\security\server.truststore -file
<인증서> - alias <인증서 별칭>
```

c. UCMDB 서버 신뢰 저장소 비밀번호(기본값: **hppass**)를 입력합니다.

d. **이 인증서를 신뢰하시겠습니까?**라는 메시지가 표시되면 **y**를 누르고 **Enter** 키를 누릅니다.

e. **인증서가 키 저장소에 추가되었습니다.**가 출력되는지 확인합니다.

2. 웹 브라우저를 시작하고 서버 주소를 http://<UCMDB 서버 호스트 이름 또는 IP>:8080/jmx-console 과 같이 입력하여 JMX 콘솔을 엽니다.

   사용자 이름과 비밀번호를 입력하여 로그인해야 할 수도 있습니다.

3. UCMDB에서 **UCMDB:service=Ports Management Services**를 클릭하여 작업 페이지를 엽니다.

   - (선택 사항) **ComponentsConfigurations**를 클릭하고 다음을 수행합니다.
     - **HTTPSSetPort**를 **8444**로 설정하고 **Invoke**를 클릭합니다.
     - **Back to MBean**을 클릭합니다.
   - **mapComponentToConnectors**를 클릭하고 다음을 수행합니다.
     - mapComponentToConnectors 서비스에서 **componentName**을 **ucmdb-ui**로 설정합니다.
     - **isHTTPSWithClientAuth**만 **true**로 설정하고 **Invoke**를 클릭합니다.
     - **Back to MBean**을 클릭합니다.
     - mapComponentToConnectors 서비스에서 **componentName**을 **root**로 설정합니다.
     - **isHTTPSWithClientAuth**만 **true**로 설정하고 **Invoke**를 클릭합니다.

4. UCMDB에서 **UCMDB:service=Security Services**를 클릭하여 작업 페이지를 엽니다. **loginWithCAC** 서비스에서 다음을 수행합니다.

   - **loginWithCAC**를 **true**로 설정하고 **Invoke**를 클릭합니다.
   - **Back to MBean**을 클릭합니다.
   - (선택 사항) UCMDB에서 사용자 이름을 추출하는 데 사용할 인증서의 필드를 지정하려면 **usernameField**를 클릭하고 **Invoke**를 클릭합니다.

     참고: 필드를 지정하지 않는 경우 기본값인 PRINCIPAL_NAME_FROM_SAN_FIELD가 사용됩니다.

   - **Back to MBean**을 클릭합니다.
   - 인증서로부터 온라인 목록을 사용할 수 없는 경우에 사용할 오프라인 CRL(인증서 폐기 목록)의 경로를 설정하려면 **pathToCRL**을 클릭하고 **Invoke**를 클릭합니다.

     참고: 로컬 CRL을 사용하는 경우 UCMDB 서버에 대한 인터넷 연결이 작동하지 않으면 로컬 CRL이 사용됩니다. 다음과 같은 상황에서는 인증서가 폐기되지 않았더라도 인증서 유효성

검사가 실패합니다.

- CRL 경로가 설정되었지만 CRL 파일 자체가 누락된 경우
- CRL이 만료된 경우
- CRL에 잘못된 서명이 있는 경우

오프라인 CRL의 경로를 설정하지 않은 경우 UCMDB 서버에서 온라인 CRL에 액세스할 수 없을 때는 CRL 또는 OCSP URL이 포함된 모든 인증서가 거부됩니다. URL이 액세스되지 않으므로 폐기 검사에 실패합니다. UCMDB 서버의 인터넷 액세스를 가능하게 하려면 **wrapper.conf** 파일에서 다음 줄의 주석을 지우고 올바른 프록시 및 포트를 지정합니다.

```
#wrapper.java.additional.40=-Dhttp.proxyHost=<PROXY_ADDR>
#wrapper.java.additional.41=-Dhttp.proxyPort=<PORT>
#wrapper.java.additional.42=-Dhttps.proxyHost=<PROXY_ADDR>
#wrapper.java.additional.43=-Dhttps.proxyPort=<PORT>
```

- **Back to MBean**을 클릭합니다.
- (선택 사항) **onlyCACCerts**를 **true**로 설정하고 **Invoke**를 클릭합니다.

  실제 CAC 장치에서 오는 인증서만 허용하려면 이 작업을 **true**로 설정합니다.

이제 https://<UCMDB 서버 호스트 이름 또는 IP>.<domainname>:8444로 UCMDB에 로그인할 수 있어야 합니다.

5. LW-SSO 인증을 사용하도록 UCMDB를 구성하고 UCMDB 서버를 다시 시작합니다.

LW-SSO 인증에 대한 자세한 내용은 "LW-SSO를 사용하여 HP Universal CMDB에 로그인하도록 설정"(87 페이지)을 참조하십시오.

서버 키 저장소 비밀번호 변경

서버를 설치한 후에는 HTTPS 포트가 열리고 저장소는 취약한 비밀번호(기본값: **hppass**)를 사용하여 보호됩니다. SSL만 사용하려는 경우에는 비밀번호를 변경해야 합니다.

다음 절차에서는 **server.keystore** 비밀번호만 변경하는 방법을 설명합니다. 그러나 **server.truststore** 비밀번호를 변경할 때도 같은 절차를 수행해야 합니다.

참고: 이 절차의 모든 단계를 수행해야 합니다.

1. UCMDB 서버를 시작합니다.
2. JMX 콘솔에서 비밀번호 변경 작업을 실행합니다.
   a. 웹 브라우저를 시작하고 서버 주소를 **http://<UCMDB 서버 호스트 이름 또는 IP>:8080/jmx-console**과 같이 입력합니다.

      사용자 이름과 비밀번호를 입력하여 로그인해야 할 수도 있습니다.
   b. UCMDB에서 **UCMDB:service=Security Services**를 클릭하여 작업 페이지를 엽니다.

c. **changeKeystorePassword** 작업을 찾아 실행합니다.

이 필드는 비워 두어야 하며 길이는 6자 이상이어야 합니다. 비밀번호는 데이터베이스에서만 변경됩니다.

3. UCMDB 서버를 중지합니다.
4. 명령을 실행합니다.

C:\hp\UCMDB\UCMBServer\bin\jre\bin에서 다음 명령을 실행합니다.

a. 저장소 비밀번호를 변경합니다.

keytool -storepasswd -new <new_keystore_pass> -keystore C:\hp\UCMDB\UCMDBServer\conf\security\server.keystore -storepass <current_ keystore_pass>

b. 다음 명령은 키 저장소의 내부 키를 표시합니다. 첫 번째 매개 변수는 별칭입니다. 다음 명령에 사용할 수 있도록 이 매개 변수를 저장합니다.

keytool -list -keystore C:\hp\UCMDB\UCMDBServer\conf\security\server.keystore

c. 저장소가 비어 있지 않은 경우 키 비밀번호를 변경합니다.

keytool -keypasswd -alias <alias> -keypass <currentPass> -new <newPass> -keystore C:\hp\UCMDB\UCMDBServer\conf\security\server.keystore

d. 새 비밀번호를 입력합니다.

5. UCMDB 서버를 시작합니다.
6. 서버 신뢰 저장소에 대해 절차를 반복합니다.

HTTP/HTTPS 포트를 사용하거나 사용하지 않도록 설정

사용자 인터페이스 또는 JMX 콘솔 내에서 HTTP 및 HTTPS 포트를 사용하거나 사용하지 않도록 설정할 수 있습니다.

사용자 인터페이스 내에서 HTTP/HTTPS 포트를 사용하거나 사용하지 않도록 설정하려면 다음을 수행합니다.

1. HP Universal CMDB에 로그온합니다.
2. **관리> 인프라 설정**을 선택합니다.
3. **필터**(이름별) 상자에 **http** 또는 **https**를 입력하여 HTTP 설정을 표시합니다.
   - **HTTP(S) 연결 사용**. **True**: 포트를 사용하도록 설정합니다. **False**: 포트를 사용하지 않도록 설정합니다.
4. 서버를 다시 시작하여 변경 내용을 적용합니다.

주의: HTTPS 포트는 기본적으로 열리며, 이 포트를 닫으면 **Server_Management.bat**가 작동하지 않습니다.

JMX 콘솔에서 HTTP/HTTPS 포트를 사용하거나 사용하지 않도록 설정하려면 다음을 수행합니다.

1. 웹 브라우저를 시작하고 주소창에 http://localhost.<domain_name>:8080/jmx-console을 입력합니다.
2. JMX 콘솔 인증 자격 증명을 입력합니다. 기본 자격 증명은 다음과 같습니다.
   - 로그인 이름 = **sysadmin**
   - 비밀번호 = **sysadmin**
3. **UCMDB:service=Ports Management Services**를 찾은 다음 링크를 클릭하여 작업 페이지를 엽니다.
4. HTTP 포트를 사용하거나 사용하지 않도록 설정하려면 **HTTPSetEnable** 작업을 찾아 값을 설정합니다.
   - **True:** 포트를 사용하도록 설정합니다.
   - **False:** 포트를 사용하지 않도록 설정합니다.
5. HTTPS 포트를 사용하거나 사용하지 않도록 설정하려면 **HTTPSSetEnable** 작업을 찾아 값을 설정합니다.
   - **True:** 포트를 사용하도록 설정합니다.
   - **False:** 포트를 사용하지 않도록 설정합니다.
6. 클라이언트 인증을 사용하는 HTTPS 포트를 사용하거나 사용하지 않도록 설정하려면 **HTTPSClientAuthSetEnable** 작업을 찾아 값을 설정합니다.
   - **True:** 포트를 사용하도록 설정합니다.
   - **False:** 포트를 사용하지 않도록 설정합니다.

UCMDB 웹 구성 요소를 포트에 매핑

JMX 콘솔에서 사용 가능한 포트에 대한 각 UCMDB 구성 요소의 매핑을 구성할 수 있습니다.

현재 구성 요소 구성을 보려면 다음을 수행합니다.

1. 웹 브라우저를 시작하고 주소창에 **http://localhost.<domain_name>:8080/jmx-console**을 입력합니다.
2. JMX 콘솔 인증 자격 증명을 입력합니다. 기본 자격 증명은 다음과 같습니다.

   로그인 이름 = **sysadmin**

   비밀번호 = **sysadmin**
3. **UCMDB:service=Ports Management Services**를 찾은 다음 링크를 클릭하여 작업 페이지를 엽니다.
4. **ComponentsConfigurations** 메서드를 찾아서 **Invoke**를 클릭합니다.
5. 각 구성 요소에 대해 유효한 포트 및 현재 매핑된 포트가 표시됩니다.

구성 요소를 매핑하려면 다음을 수행합니다.

1. **UCMDB:service=Ports Management Services**를 찾은 다음 링크를 클릭하여 작업 페이지를 엽니다.
2. **mapComponentToConnectors** 메서드를 찾습니다.
3. 값 상자에 구성 요소 이름을 입력합니다. 선택한 항목에 해당하는 각 포트에 대해 **True** 또는 **False**를 선택합니다. **Invoke**를 클릭합니다. 선택한 구성 요소가 선택한 포트에 매핑됩니다. **serverComponentsNames** 메서드를 호출하면 구성 요소 이름을 찾을 수 있습니다.
4. 각 관련 구성 요소에 대해 이 프로세스를 반복합니다.

참고:

- 각 구성 요소는 하나 이상의 포트에 매핑되어야 합니다. 포트에 매핑되지 않은 구성 요소는 기본적으로 HTTP 포트에 매핑됩니다.
- 한 구성 요소를 HTTPS 포트와 클라이언트 인증을 사용하는 HTTPS 포트에 동시에 매핑하는 경우에는 클라이언트 인증 옵션만 매핑되고 나머지 옵션은 중복되는 것으로 간주됩니다.
- UCMDB UI 구성 요소에 대해 **isHTTPSWithClientAuth**를 **True**로 설정하면 루트 구성 요소에 대해서도 **True**로 설정해야 합니다.

각 포트에 할당된 값을 변경할 수도 있습니다.

포트의 값을 설정하려면 다음을 수행합니다.

1. **UCMDB:service=Ports Management Services**를 찾은 다음 링크를 클릭하여 작업 페이지를 엽니다.
2. HTTP 포트의 값을 설정하려면 **HTTPSetPort** 메서드를 찾은 다음 **값** 상자에 값을 입력하고 **Invoke**를 클릭합니다.
3. HTTPS 포트의 값을 설정하려면 **HTTPSSetPort** 메서드를 찾은 다음 **값** 상자에 값을 입력하고 **Invoke**를 클릭합니다.
4. 클라이언트 인증을 사용하는 HTTPS 포트의 값을 설정하려면 **HTTPSClientAuthSetPort** 메서드를 찾은 다음 **값** 상자에 값을 입력하고 **Invoke**를 클릭합니다.

SSL을 사용하여 UCMDB와 함께 작동하도록 Configuration Manager 구성

SSL(Secure Sockets Layer)을 사용하여 UCMDB와 함께 작동하도록 Configuration Manager를 구성할 수 있습니다. UCMDB에서 8443 포트의 SSL 커넥터가 기본적으로 활성화됩니다.

1. **<UCMDB 설치 디렉터리>\bin\jre\bin**으로 이동하여 다음 명령을 실행합니다.

   keytool -export -alias hpcert -keystore <*UCMDB_server_directory*>
   \conf\security\server.keystore -storepass hppass -file <certificatefile>
2. 인증서 파일을 로컬 Configuration Manager 컴퓨터의 임시 위치에 복사합니다.
3. Configuration Manager를 새로 설치하거나 기존 설치를 재구성합니다. 자세한 내용은 대화형 *HP Universal CMDB 배포 안내서*에서 해당 섹션을 참조하십시오.

UCMDB 구성 화면에서 프로토콜을 HTTPS로 설정하고 단계 2에서 복사한 인증서 파일을 선택합니다.

4. **hpcert.cer**을 서버 컴퓨터의 **<*Configuration_Manager_installation_directory*>\java\windows\x86_64\bin** 폴더에 복사합니다.
5. 서버 컴퓨터에서 다음 명령으로 keytool 유틸리티를 사용하여 인증서를 신뢰 저장소(cacerts)로 가져옵니다.

   <*Configuration_Manager_installation_directory*>\java\bin\keytool.exe -import -alias hp -file hpcert.cer -keystore <*Configuration_Manager_installation_directory*>\java\windows\x86_64\lib\security\cacerts
6. **hpcert.cer**을 서버 컴퓨터의 **<*Configuration_Manager_installation_directory*>\java\windows\x86_64\lib\security** 폴더에 복사합니다.
7. 자체 서명 인증서 및 일치하는 개인 키가 포함된 서버 키 저장소(JKS 유형)를 만듭니다. **<*Configuration_Manager_installation_directory*>\java\windows\x86_64\bin** 폴더에서 다음 명령을 실행합니다.

   keytool -genkey -alias tomcat -keyalg RSA -keystore <Configuration_Manager_installation_directory>\java\windows\x86_64\lib\security\tomcat.keystore

   a. 키 저장소 비밀번호를 입력합니다.
   b. 이름과 성은 무엇입니까?라는 메시지가 나타나면 Configuration Manager 웹 서버 이름을 입력하고 조직에 따라 다른 매개 변수를 입력합니다.
   c. 키 비밀번호를 입력합니다. 키 비밀번호는 키 저장소 비밀번호와 같아야 합니다. **hpcert**라는 서버 인증서를 사용하여 **tomcat.keystore**라는 JKS 키 저장소가 만들어집니다.
8. **server.xml** 파일을 다음과 같이 수정합니다.
   a. **<*Configuration_Manager_installation_directory*>\servers\server-0\conf** 폴더에 있는 server.xml file 파일을 엽니다. 주석에서

      Connector port="8143"

      내용으로 시작되는 섹션을 찾습니다. 주석 문자를 지우고 다음 줄을 추가하여 스크립트를 활성화합니다.

      keystoreFile="<Configuration_Manager_installation_directory>\java\windows\x86_64\lib\security\tomcat.keystore"
      keystorePass="password"
      truststoreFile="<Configuration_Manager_installation_directory>\java\windows\x86_64\lib\security\cacerts"
      truststorePass="changeit" />

   b. 다음 줄을 주석 처리합니다.

      <Listener className="org.apache.catalina.core.AprLifecycleListener" SSLEngine="on" />
9. 서버를 다시 시작합니다.

SSL을 사용하여 다른 제품(예: 로드 균형 조정)과 함께 작동하도록 Configuration Manager를 구성하려면, 다음 명령을 실행하여 제품의 보안 인증서를 Configuration Manager 신뢰 저장소(기본 jre 신뢰 저장소)로 가져옵니다.

<CM_JAVA_HOME>\bin\keytool -import -trustcacerts -alias <alias> -keystore
<CM_JAVA_HOME>\lib\security\cacerts -storepass changeit -file <certificatefile>

UCMDB KPI 어댑터에 SSL을 사용하도록 설정

SSL(Secure Sockets Layer)을 통해 UCMDB KPI 어댑터 정보를 전송하도록 구성할 수 있습니다.

1. Configuration Manager 인증서를 내보냅니다.

   <CM_JAVA_HOME>\bin\keytool -export -alias tomcat -keystore
   <CM_JAVA_HOME>\lib\security\tomcat.keystore -storepass
   <keystore pass> -file <인증서 파일 이름>

2. 다음과 같이 Configuration Manager에서 내보낸 인증서를 UCMDB 신뢰 저장소로 가져옵니다.

   <UCMDB server dir>\bin\jre\bin keytool -import -trustcacerts
   -alias tomcat -keystore <UCMDB server dir>\bin\jre\lib
   \security\cacerts -storepass changeit -file <인증서 파일>

3. 다음과 같이 Configuration Manager에서 내보낸 인증서를 프로브의 신뢰 저장소로 가져옵니다.

   a. 명령 프롬프트를 열고 명령을 실행합니다.

      <DataFlowProbe dir>\bin\jre\bin\keytool.exe -import -v -keystore
      <DataFlowProbe dir>\conf\security\hpprobeTrustStore.jks -file
      <certificatefile> -alias tomcat

   b. 키 저장소 비밀번호(logomania)를 입력합니다.

   c. **Trust this certificate?**라는 메시지가 표시되면 **y**를 누르고 **Enter** 키를 누릅니다.

      다음 메시지가 표시됩니다.

      Certificate was added to keystore.

      Data Flow Probe 강화에 대한 자세한 내용은 "Data Flow Probe 강화"(62페이지)를 참조하십시오.

4. UCMDB, Data Flow Probe 및 Configuration Manager를 다시 시작합니다.

UCMDB Browser에 대한 SSL 지원 구성

참고: 여기에 있는 지침은 UCMDB Browser 버전 1.95와 관련된 내용입니다. 나머지 UCMDB 제품군과는 별도로 업그레이드된 더 높은 버전의 UCMDB Browser가 있는 경우, 해당 버전의 *HP Universal CMDB Browser Installation and Configuration Guide*에서 SSL 지원 구성에 대한 섹션을 참조하십시오.

Tomcat에서 SSL 지원을 설치 및 구성하려면 다음을 수행합니다.

1. 다음 명령 중 하나를 실행하여 서버의 개인 키와 자체 서명된 인증서를 저장할 keystore 파일을 만듭니다.
   - Windows의 경우: **%JAVA_HOME%\bin\keytool -genkey -alias tomcat -keyalg RSA**
   - Unix의 경우: **$JAVA_HOME/bin/keytool -genkey -alias tomcat -keyalg RSA**

   두 명령 모두에 대해 비밀번호 값 **changeit**를 사용합니다(열리는 콘솔 대화 상자의 다른 모든 필드에는 임의의 값 사용 가능).
2. **$CATALINA_BASE/conf/server.xml**의 **SSL HTTP/1.1 Connector** 항목에서 주석을 제거합니다. 여기서 **$CATALINA_BASE**는 Tomcat을 설치한 디렉터리입니다.

   참고: server.xml이 SSL을 사용하도록 구성하는 방법에 대한 전체 설명은 Apache Tomcat 공식 사이트: http://tomcat.apache.org/tomcat-7.0-doc/ssl-howto.html을 참조하십시오.
3. Tomcat 서버를 다시 시작합니다.

UCMDB 서버 연결에 HTTPS 프로토콜을 사용하려면 다음을 수행합니다.

1. **ucmdb_browser_config.xml**에서 **<protocol>** 태그에 **https** 값을 할당하고 UCMDB 서버 HTTPS 포트 값(기본값 8443)을 **<port> 태그에 할당합니다.**
2. UCMDB 서버 공용 인증서를 UCMDB Browser 컴퓨터로 다운로드(UCMDB-Server에서 SSL을 사용하는 경우에는 UCMDB 관리자가 이 인증서를 제공)하고 다음 명령을 실행하여 서버에 연결할 JRE에 있는 **cacerts** 신뢰 저장소로 가져옵니다.

   "%JAVA_HOME%\bin\keytool" -import -alias ucmdb -trustcacerts -file <UCMDB-Server-certificate-file> -keystore "%JAVA_HOME%\jre\lib\security\cacerts"

   여기서 **<UCMDB-Server-certificate-file>**은 UCMDB 서버 공용 인증서 파일의 전체 경로입니다.
3. Tomcat 서버를 다시 시작합니다.

3장: 리버스 프록시 사용

이 섹션에서는 리버스 프록시가 보안에 미치는 영향과 HP Universal CMDB 및 Configuration Manager에서 리버스 프록시를 사용하기 위한 지침을 설명합니다. 이 장에서 다루는 문제는 리버스 프록시의 보안 관련 문제이고, 캐싱 및 로드 균형 조정 등의 기타 문제에 대해서는 다루지 않습니다.

이 장의 내용:

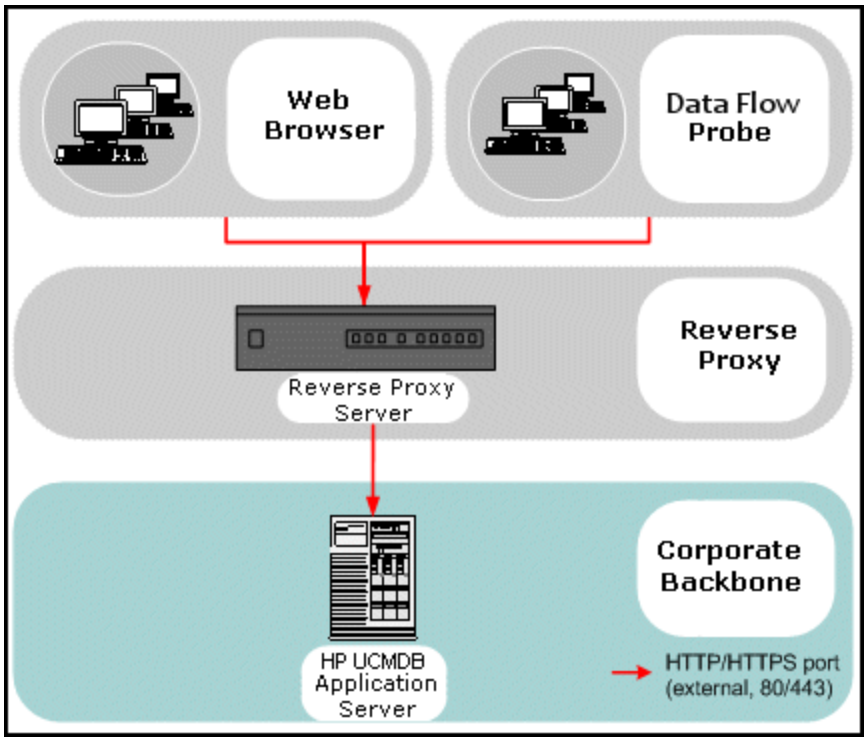
리버스 프록시 개요

리버스 프록시는 클라이언트 컴퓨터와 웹 서버 간에 배치된 중간 서버입니다. 클라이언트 컴퓨터에서 리버스 프록시는 클라이언트 컴퓨터의 HTTP 프로토콜 요청을 처리하는 표준 웹 서버로 표시됩니다.

클라이언트 컴퓨터는 웹 서버의 이름 대신 리버스 프록시의 이름을 사용하여 웹 컨텐츠에 대한 일반 요청을 보냅니다. 리버스 프록시는 웹 서버 중 하나로 요청을 보냅니다. 응답은 리버스 프록시를 통해 다시 클라이언트 컴퓨터로 전송되지만, 클라이언트 컴퓨터에는 웹 서버에서 보내는 것으로 표시됩니다.

서로 다른 URL을 사용하는 다중 리버스 프록시가 같은 UCMDB/CM 인스턴스를 나타낼 수도 있습니다. 또는 UCMDB/CM 서버에 서로 다른 루트 컨텍스트를 설정하여 리버스 프록시 서버 하나로 여러 UCMDB/CM 서버에 액세스할 수도 있습니다.

HP Universal CMDB 및 Configuration Manager는 DMZ 아키텍처에서 리버스 프록시를 지원합니다. 리버스 프록시는 Data Flow Probe와 웹 클라이언트 및 HP Universal CMDB/CM 서버 간의 HTTP 중재자입니다.



참고:

- 각 리버스 프록시 유형마다 서로 다른 구성 구문이 필요합니다. Apache 2.0.x 리버스 프록시 구성의 예는 "예: Apache 2.0.x 구성 "(37페이지)을 참조하십시오.
- 스케줄러를 사용하여 보고서에 대한 직접 링크를 만드는 경우에만 프런트 엔드 URL 설정 구성이 필요합니다.

리버스 프록시 서버 사용 시의 보안 문제

리버스 프록시 서버는 배스천 호스트 기능을 합니다. 프록시는 외부 클라이언트에서 직접 주소를 지정하는 유일한 컴퓨터로 구성되므로, 내부 네트워크의 다른 컴퓨터는 확인하기가 어렵습니다. 리버스 프록시를 사용하면 응용 프로그램 서버를 내부 네트워크에서 별도의 컴퓨터에 배치할 수 있습니다.

이 섹션에서는 연속 토폴로지 환경에서 DMZ 및 리버스 프록시를 사용하는 방법을 설명합니다.

다음은 이러한 환경에서 리버스 프록시를 사용하는 경우의 주요 보안 이점입니다.

- DMZ 프로토콜이 변환되지 않습니다. 들어오는 프로토콜과 나가는 프로토콜은 동일하며 머리글만 변경됩니다.
- HTTP를 통해서만 리버스 프록시에 액세스할 수 있습니다. 즉, 상태 저장 패킷 검사 방화벽이 통신을 보다 효율적으로 보호할 수 있습니다.
- 제한된 정적 리디렉션 요청 집합을 리버스 프록시에서 정의할 수 있습니다.
- 인증 방법, 암호화 등 대부분의 웹 서버 보안 기능을 리버스 프록시에서 사용할 수 있습니다.

- 리버스 프록시는 실제 서버의 IP 주소와 내부 네트워크의 아키텍처를 모두 차단합니다.
- 웹 서버에서 액세스할 수 있는 클라이언트는 리버스 프록시뿐입니다.
- 이 구성에서는 다른 솔루션과 달리 NAT 방화벽이 지원됩니다.
- 리버스 프록시의 경우 방화벽에서 최소한의 포트만 열려 있으면 됩니다.
- 리버스 프록시를 사용하는 경우 다른 배스천 솔루션에 비해 성능이 우수합니다.

리버스 프록시 구성

이 섹션에서는 리버스 프록시를 구성하는 방법을 설명합니다. UCMDB 버전 10.01부터 UCMDB에서 구성할 필요가 없습니다. 리버스 프록시 쪽에서 리버스 프록시 문서에 따라 구성 파일을 편집합니다. 예는 "예: Apache 2.0.x 구성 "(37페이지)을 참조하십시오.

UCMDB 버전 10.01 이전에 생성된 예약된 작업의 경우 다음과 같이 UCMDB에서도 구성 설정이 필요합니다.

인프라 설정을 사용하여 리버스 프록시 구성

다음 절차에서는 인프라 설정에 액세스하여 리버스 프록시를 구성하는 방법을 설명합니다. 이 구성은 스케줄러를 사용하여 보고서에 대한 직접 링크를 만드는 경우에만 필요합니다.

리버스 프록시를 구성하려면 다음을 수행합니다.

1. **관리 > 인프라 설정 > 일반 설정** 범주를 선택합니다.
2. **프런트 엔드 URL** 설정을 변경합니다. **https://my_proxy_server:443/**과 같이 주소를 입력합니다.

참고: 이 사항을 변경하고 나면 클라이언트를 통해 직접 HP Universal CMDB 서버에 액세스할 수 없습니다. 리버스 프록시 구성을 변경하려면 서버 컴퓨터에서 JMX 콘솔을 사용합니다. 자세한 내용은 아래의 "JMX 콘솔을 사용하여 리버스 프록시 구성"을 참조하십시오.

JMX 콘솔을 사용하여 리버스 프록시 구성

HP Universal CMDB 서버 컴퓨터에서 JMX 콘솔을 사용하여 리버스 프록시 구성을 변경할 수 있습니다. 이 구성은 스케줄러를 사용하여 보고서에 대한 직접 링크를 만드는 경우에만 필요합니다.

리버스 프록시 구성을 변경하려면 다음을 수행합니다.

1. HP Universal CMDB 서버 컴퓨터에서 웹 브라우저를 시작하고 다음 주소를 입력합니다.

   http://<컴퓨터 이름 또는 IP 주소>.<도메인 이름>:8080/jmx-console

   여기서 **<컴퓨터 이름 또는 IP 주소>**는 HP Universal CMDB가 설치되어 있는 컴퓨터입니다. 사용자 이름과 비밀번호를 입력하여 로그인해야 할 수도 있습니다.

2. **UCMDB-UI > UCMDB-UI:name=UI Server frontend settings** 링크를 클릭합니다.

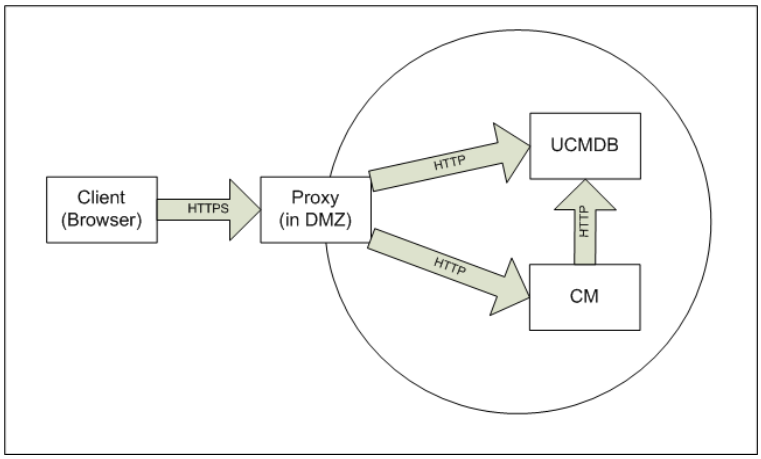
   setUseFrontendURLBySettings 필드에 서버 프록시 URL을 https://my_proxy_server:443/과 같이 입력합니다.

3. **Invoke**를 클릭합니다.
4. 이 설정의 값을 확인하려면 **showFrontendURLInSettings** 메서드를 사용합니다.

예: Apache 2.0.x 구성

이 섹션에서는 Data Flow Probe 및 응용 프로그램 사용자가 모두 HP Universal CMDB에 연결하는 경우 Apache 2.0.x 리버스 프록시 사용을 지원하는 구성 파일의 예입니다.

다음 다이어그램은 Configuration Manager와 UCMDB에 대한 리버스 프록시의 구성 프로세스를 나타냅니다.



참고:

- 이 예에서 HP Universal CMDB 컴퓨터의 DNS 이름과 포트는 UCMDB_server입니다.
- 이 예에서 HP Configuration Manger의 DNS 이름과 포트는 UCMDB_CM_server입니다.
- Apache 관리 방식을 알고 있는 사용자만 이 구성을 변경해야 합니다.

1. **<Apache 컴퓨터 루트 디렉터리>\Webserver\conf\httpd.conf** 파일을 엽니다.
2. 다음 모듈을 사용하도록 설정합니다.
   - **LoadModule proxy_module modules/mod_proxy.so**
   - **LoadModule proxy_http_module modules/mod_proxy_http.so**
3. 다음 줄을 **httpd.conf** 파일에 추가합니다.

```
ProxyRequests off
<Proxy *>
   Order deny,allow
   Deny from all
   Allow from all
</Proxy>

ProxyPass /mam http://UCMDB_server/mam
ProxyPassReverse /mam http://UCMDB_server/mam
```

```
ProxyPass /mam_images http://UCMDB_server/mam_images
ProxyPassReverse /mam_images http://UCMDB_server/mam_images
ProxyPass /mam-collectors http://UCMDB_server/mam-collectors
ProxyPassReverse /mam-collectors http://UCMDB_server/mam-collectors
ProxyPass /ucmdb http://UCMDB_server/ucmdb
ProxyPassReverse /ucmdb http://UCMDB_server/ucmdb
ProxyPass /site http://UCMDB_server/site
ProxyPassReverse /site http://UCMDB_server/site
ProxyPass /ucmdb-ui http://UCMDB_server/ucmdb-ui
ProxyPassReverse /ucmdb-ui http://UCMDB_server/ucmdb-ui
ProxyPass /status http://UCMDB_server/status
ProxyPassReverse /status http://UCMDB_server/status
ProxyPass /jmx-console http://UCMDB_server/jmx-console
ProxyPassReverse /jmx-console http://UCMDB_server/jmx-console
ProxyPass /axis2 http://UCMDB_server/axis2
ProxyPassReverse /axis2 http://UCMDB_server/axis2
ProxyPass /icons http://UCMDB_server/icons
ProxyPassReverse /icons http://UCMDB_server/icons
ProxyPass /ucmdb-api http://UCMDB_server/ucmdb-api
ProxyPassReverse /ucmdb-api http://UCMDB_server/ucmdb-api
ProxyPass /ucmdb-docs http://UCMDB_server/ucmdb-docs
ProxyPassReverse /ucmdb-docs http://UCMDB_server/ucmdb-docs
ProxyPass /ucmdb-api/8.0 http://UCMDB_server/ucmdb-api/8.0
ProxyPassReverse /ucmdb-api/8.0 http://UCMDB_server/ucmdb-api/8.0
ProxyPass /cm http://UCMDB_Server/cm
ProxyPassReverse /cm http://UCMDB_Server /cm
ProxyPass /cnc http://UCMDB_CM_server/cnc
ProxyPassReverse /cnc http://UCMDB_CM_server/cnc
ProxyPass /docs http://UCMDB_CM_server/docs
ProxyPassReverse /docs http://UCMDB_CM_server/docs
ProxyPass /ucmdb-browser http://UCMDB_CM_server/ucmdb-browser
ProxyPassReverse /ucmdb-browser http://UCMDB_CM_server/ucmdb-browser
```

4. **변경 내용을 저장합니다.**

상호 인증을 사용하여 리버스 프록시 또는 로드 균형 조정으로 Data Flow Probe 연결

상호 인증을 사용하여 리버스 프록시 또는 로드 균형 조정을 통해 Data Flow Probe를 연결하려면 다음 절차를 수행합니다. 이 절차는 다음 구성에 적용됩니다.

- **리버스 프록시 또는 로드 균형 조정에 필요하며 프로브에서 제공하는 클라이언트 인증서를 기반으로 하는, 프로브와 리버스 프록시 또는 로드 균형 조정 사이의 상호 SSL 인증**
- **리버스 프록시 또는 로드 균형 조정과 UCMDB 서버 사이의 일반 SSL 연결**

참고: 다음 지침에서는 **cKeyStoreFile** 키 저장소를 프로브 키 저장소로 사용합니다. 이 키 저장소는 Data Flow Probe 설치의 일부이며 자체 서명된 인증서를 포함하는 미리 정의된 클라이언트 키 저장소입니다. 자세한 내용은 "서버 및 Data Flow Probe 기본 키 저장소와 신뢰 저장소"(78페이지)를 참조하십시오.

새로 생성된 개인 키를 포함할 고유한 키 저장소를 새로 만드는 것이 좋습니다. 자세한 내용은 "Data Flow Probe용 키 저장소 만들기"(77페이지)를 참조하십시오.

인증 기관에서 인증서 얻기

CA 루트 인증서를 받아 다음 위치로 가져옵니다.

- Data Flow Probe 신뢰 저장소
- Data Flow Probe JVM cacerts
- UCMDB 서버 신뢰 저장소
- 리버스 프록시 신뢰 저장소

1. CA 루트 인증서를 Data Flow Probe 신뢰 저장소로 가져옵니다.
   a. CA 루트 인증서를 다음 디렉터리에 배치합니다. <Data Flow Probe 설치 디렉터리>\conf\security\<인증서 파일 이름>
   b. 다음 스크립트를 실행하여 CA 루트 인증서를 데이터 흐름 신뢰 저장소로 가져옵니다.

      <Data Flow Probe 설치 디렉터리>\bin\jre\bin\keytool.exe -import -trustcacerts -alias <사용자 별칭> -file C:\hp\UCMDB\DataFlowProbe\conf\security\<인증서 파일 이름> -keystore <Data Flow Probe 설치 디렉터리>\conf\security\MAMTrustStoreExp.jks

      기본 비밀번호는 **logomania**입니다.
2. 다음 스크립트를 실행하여 CA 루트 인증서를 Data Flow Probe JVM cacerts로 가져옵니다.

   <Data Flow Probe 설치 디렉터리>\bin\jre\bin\keytool.exe -import -trustcacerts -alias <사용자 별칭> -file <Data Flow Probe 설치 디렉터리>\conf\security\<인증서 파일 이름> -keystore <Data Flow Probe 설치 디렉터리>\bin\jre\lib\security\cacerts

   기본 비밀번호는 **changeit**입니다.
3. CA 루트 인증서를 UCMDB 신뢰 저장소로 가져옵니다.
   a. CA 루트 인증서를 다음 디렉터리에 배치합니다. <UCMDB 설치 디렉터리>\conf\security\<인증서 파일 이름>
   b. 다음 스크립트를 실행하여 CA 루트 인증서를 UCMDB 신뢰 저장소로 가져옵니다.

      <UCMDB 설치 디렉터리>\bin\jre\bin\keytool.exe -import -trustcacerts -alias <사용자 별칭> -file <UCMDB 설치 디렉터리>\conf\security\<인증서 파일 이름> -keystore <UCMDB 설치 디렉터리>\conf\security\sever.truststore

      기본 비밀번호는 **hppass**입니다.
4. CA 루트 인증서를 리버스 프록시 신뢰 저장소로 가져옵니다. 이 단계는 벤더에 대한 종속성이 큽니다.

인증서를 Java 키 저장소로 변환

CA(인증 기관)에서 PFX/PKCS12 형식으로 Data Flow Probe에 대한 클라이언트 인증서와 개인 키를 얻은 후 다음 스크립트를 실행하여 Java 키 저장소로 변환합니다.

<Data Flow Probe 설치 디렉터리>\bin\jre\bin\keytool.exe -importkeystore -srckeystore <PFX 키 저장소 전체 경로> -destkeystore <새 대상 키 저장소 전체 경로> -srcstoretype PKCS12

소스 및 대상 키 저장소 비밀번호를 입력하는 프롬프트가 표시됩니다.

소스 키 저장소 비밀번호에는 PFX 키 저장소를 내보낼 때 사용한 것과 같은 비밀번호를 사용합니다.

Data Flow Probe 키 저장소의 기본 대상 키 저장소 비밀번호는 **logomania**입니다.

참고: 기본 Data Flow Probe 키 저장소 비밀번호(logomania)와 다른 대상 키 저장소 비밀번호를 입력한 경우에는 **<Data Flow Probe 설치 디렉터리>\conf\ssl.properties** 파일에 암호화된 형식으로 새 비밀번호를 제공해야 합니다(javax.net.ssl.keyStorePassword). 자세한 내용은 "프로브 키 저장소 및 신뢰 저장소 비밀번호 암호화"(77페이지)를 참조하십시오.

다음 디렉터리에 새 키 저장소를 배치합니다. <Data Flow Probe 설치 디렉터리>\conf\security

주의: hpprobeKeyStore.jks 파일을 덮어쓰지 마십시오.

새로 만든 키 저장소를 사용하도록 SSL 속성 파일 변경

<Data Flow Probe 설치 디렉터리>\conf\ssl.properties 파일에서 클라이언트 인증서가 포함된 키 저장소를 **javax.net.ssl.keyStore**로 설정합니다.

키 저장소의 비밀번호가 기본 Data Flow Probe 키 저장소 비밀번호(logomania)가 아닌 경우에는 암호화한 후에 **javax.net.ssl.keyStorePassword**를 업데이트합니다. 비밀번호 암호화에 대한 자세한 내용은 "프로브 키 저장소 및 신뢰 저장소 비밀번호 암호화"(77페이지)를 참조하십시오.

Data Flow Probe 구성 검토

<Data Flow Probe 설치 디렉터리>\conf\DataFlowProbe.properties 파일을 다음과 같이 편집합니다.

appilog.agent.probe.protocol = HTTPS

serverName = <리버스 프록시 서버 주소>

serverPortHttps = <리버스 프록시가 요청을 UCMDB로 리디렉션하기 위해 수신하는 HTTPS 포트>

UCMDB가 SSL을 사용하여 작업하도록 구성

자세한 내용은 "SSL(Secure Sockets Layer) 통신 사용"(16페이지)을 참조하십시오.

이 절차에서 나머지 인증서를 만든 것과 같은 CA에서 UCMDB 서버 인증서를 만든 경우, 리버스 프록시 또는 로드 균형 조정에서 UCMDB 인증서를 신뢰합니다.

리버스 프록시를 통해 UCMDB에서 CAC 지원 구성

이 섹션에서는 리버스 프록시를 사용하여 UCMDB에 CAC(공통 액세스 카드) 지원을 구성하는 방법을 설명합니다.

1. 웹 브라우저를 시작하고 서버 주소를 http://<UCMDB 서버 호스트 이름 또는 IP>:8080/jmx-console과 같이 입력하여 JMX 콘솔을 엽니다.

   사용자 이름과 비밀번호를 입력하여 로그인해야 할 수도 있습니다.
2. UCMDB에서 **UCMDB:service=Ports Management Services**를 클릭하여 작업 페이지를 엽니다.
   - (선택 사항) **ComponentsConfigurations**를 클릭하고 다음을 수행합니다.
     - **HTTPSetPort**를 **8080**으로 설정하고 **Invoke**를 클릭합니다.
     - **Back to MBean**을 클릭합니다.
   - **mapComponentToConnectors**를 클릭하고 다음을 수행합니다.
     - mapComponentToConnectors 서비스에서 **componentName**을 **ucmdb-ui**로 설정합니다.
     - **isHTTP**만 **true**로 설정하고 **Invoke**를 클릭합니다.
     - **Back to MBean**을 클릭합니다.
     - mapComponentToConnectors 서비스에서 **componentName**을 **root**로 설정합니다.
     - **isHTTP**만 **true**로 설정하고 **Invoke**를 클릭합니다.
3. UCMDB에서 **UCMDB:service=Security Services**를 클릭하여 작업 페이지를 엽니다.
   - **loginWithCAC**를 **true**로 설정하고 **Invoke**를 클릭합니다.
   - **Back to MBean**을 클릭합니다.
   - **withReverseProxy**를 **true**로 설정하고 **Invoke**를 클릭합니다.

     이 설정은 UCMDB에서 사용할 사용자 이름과 인증에 사용할 인증서를 UCMDB 서버가 UCMDB_SSL_CLIENT_CERT 헤더에서 추출하도록 지시합니다.
   - **Back to MBean**을 클릭합니다.
   - (선택 사항) **onlyCACCerts**를 **true**로 설정하고 **Invoke**를 클릭합니다.

     실제 CAC 장치에서 오는 인증서만 허용하려면 이 작업을 **true**로 설정합니다.
   - (선택 사항) UCMDB에서 사용자 이름을 추출하는 데 사용할 인증서의 필드를 지정하려면 **usernameField**를 클릭하고 **호출**을 클릭합니다.

참고: 필드를 지정하지 않는 경우 기본값인 PRINCIPAL_NAME_FROM_SAN_FIELD가 사용됩니다.

4. UCMDB 서버를 다시 시작합니다.

예: Apache 2.4.4 구성

이 섹션에서는 Apache 2.4.4에 대한 샘플 구성 파일을 설명합니다.

참고: 이 예제에서는 Apache 서버가 **c:\Apache24**에 설치되었다고 가정합니다. 다른 폴더에 설치되어 있으면 올바른 위치를 지정하도록 모든 경우의 예를 변경해야 합니다.

이 예에 사용되는 상호 인증의 포트는 443입니다. **c:\Apache24\conf** 폴더에서 다음을 복사합니다.

- Apache 서버에서 사용되는 인증서(**server.crt**)
- Apache 서버의 개인 키(**server.key**)
- Apache 서버의 신뢰할 수 있는 CA(**ssl.crt**)
- 인증 폐기 목록(**ssl.crt**).

참고: 이러한 4개 파일은 모두 PEM 형식이어야 합니다.

c:\Apache24\conf\httpd.conf의 컨텐츠를 다음으로 바꿉니다(이에 따라 [APACHE_MACHINE_FQD] 변경).

```
ServerRoot "c:/Apache24"
Listen 80
LoadModule access_compat_module modules/mod_access_compat.so
LoadModule actions_module modules/mod_actions.so
LoadModule alias_module modules/mod_alias.so
LoadModule allowmethods_module modules/mod_allowmethods.so
LoadModule asis_module modules/mod_asis.so
LoadModule auth_basic_module modules/mod_auth_basic.so
LoadModule authn_core_module modules/mod_authn_core.so
LoadModule authn_file_module modules/mod_authn_file.so
LoadModule authz_core_module modules/mod_authz_core.so
LoadModule authz_groupfile_module modules/mod_authz_groupfile.so
LoadModule authz_host_module modules/mod_authz_host.so
LoadModule authz_user_module modules/mod_authz_user.so
LoadModule autoindex_module modules/mod_autoindex.so
LoadModule cgi_module modules/mod_cgi.so
LoadModule dir_module modules/mod_dir.so
LoadModule env_module modules/mod_env.so
LoadModule headers_module modules/mod_headers.so
LoadModule include_module modules/mod_include.so
LoadModule isapi_module modules/mod_isapi.so
LoadModule log_config_module modules/mod_log_config.so
LoadModule mime_module modules/mod_mime.so
```

```
LoadModule negotiation_module modules/mod_negotiation.so
LoadModule proxy_module modules/mod_proxy.so
LoadModule proxy_html_module modules/mod_proxy_html.so
LoadModule proxy_http_module modules/mod_proxy_http.so
LoadModule setenvif_module modules/mod_setenvif.so
LoadModule socache_shmcb_module modules/mod_socache_shmcb.so
LoadModule ssl_module modules/mod_ssl.so
LoadModule xml2enc_module modules/mod_xml2enc.so
<IfModule unixd_module>
User daemon
Group daemon
</IfModule>
ServerAdmin admin@example.com
ServerName [APACHE_MACHINE_FQD]:80
<Directory />
   AllowOverride none
   Require all denied
</Directory>
DocumentRoot "c:/Apache24/htdocs"
<Directory "c:/Apache24/htdocs">
   Options Indexes FollowSymLinks
   AllowOverride None
   Require all granted
</Directory>
<IfModule dir_module>
   DirectoryIndex index.html
</IfModule>
<Files ".ht*">
   Require all denied
</Files>
ErrorLog "logs/error.log"
LogLevel warn
<IfModule log_config_module>
   LogFormat "%h %l %u %t \"%r\" %>s %b \"%{Referer}i\" \"%{User-Agent}i\"" combined
   LogFormat "%h %l %u %t \"%r\" %>s %b" common
   <IfModule logio_module>
    LogFormat "%h %l %u %t \"%r\" %>s %b \"%{Referer}i\" \"%{User-Agent}i\" %I %O" combinedio
   </IfModule>
   CustomLog "logs/access.log" common
</IfModule>
<IfModule alias_module>
   ScriptAlias /cgi-bin/ "c:/Apache24/cgi-bin/"
</IfModule>
<IfModule cgid_module>
</IfModule>
<Directory "c:/Apache24/cgi-bin">
   AllowOverride None
   Options None
```

```
    Require all granted
</Directory>
<IfModule mime_module>
    TypesConfig conf/mime.types
    AddType application/x-compress .Z
    AddType application/x-gzip .gz .tgz
</IfModule>
<IfModule proxy_html_module>
Include conf/extra/proxy-html.conf
</IfModule>
Include conf/extra/httpd-ssl.conf
<IfModule ssl_module>
SSLRandomSeed startup builtin
SSLRandomSeed connect builtin
</IfModule>
```

또한 c:\Apache24\conf\extra\httpd-ssl.conf의 컨텐츠를 다음으로 바꿉니다(이에 따라 [APACHE_MACHINE_FQD], [UCMDB_SERVER_NAME] 및 [UCMDB_CM_SERVER_NAME] 변경).

```
Listen 443
SSLCipherSuite HIGH:MEDIUM:!aNULL:!MD5
SSLPassPhraseDialog  builtin
SSLSessionCache        "shmcb:c:/Apache24/logs/ssl_scache(512000)"
SSLSessionCacheTimeout  300
<VirtualHost _default_:443>
DocumentRoot "c:/Apache24/htdocs"
ServerName [APACHE_MACHINE_FQD]:443
ServerAdmin admin@example.com
ErrorLog "c:/Apache24/logs/error.log"
TransferLog "c:/Apache24/logs/access.log"
SSLEngine on
SSLCertificateFile "c:/Apache24/conf/server.crt"
SSLCertificateKeyFile "c:/Apache24/conf/server.key"
SSLCACertificateFile "c:/Apache24/conf/ssl.crt"
SSLCARevocationFile "c:/Apache24/conf/ssl.crl"
SSLCARevocationCheck leaf
SSLVerifyClient require
SSLVerifyDepth  10
SSLOptions +ExportCertData
<FilesMatch "\.(cgi|shtml|phtml|php)$">
    SSLOptions +StdEnvVars
</FilesMatch>
<Directory "c:/Apache24/cgi-bin">
    SSLOptions +StdEnvVars
</Directory>
BrowserMatch "MSIE [2-5]" \
        nokeepalive ssl-unclean-shutdown \
        downgrade-1.0 force-response-1.0
CustomLog "c:/Apache24/logs/ssl_request.log" \
        "%t %h %{SSL_PROTOCOL}x %{SSL_CIPHER}x \"%r\" %b"
```

```
RequestHeader set UCMDB_SSL_CLIENT_CERT %{SSL_CLIENT_CERT}e
ProxyRequests off
<Proxy *>
   Order deny,allow
   Deny from all
   Allow from all
</Proxy>
ProxyPass / http://[UCMDB_SERVER_NAME]:8080/
ProxyPassReverse / http://[UCMDB_SERVER_NAME]:8080/
ProxyPass /mam http://[UCMDB_SERVER_NAME]:8080/mam
ProxyPassReverse /mam http://[UCMDB_SERVER_NAME]:8080/mam
ProxyPass /mam_images http://[UCMDB_SERVER_NAME]:8080/mam_images
ProxyPassReverse /mam_images http://[UCMDB_SERVER_NAME]:8080/mam_images
ProxyPass /mam-collectors http://[UCMDB_SERVER_NAME]:8080/mam-collectors
ProxyPassReverse /mam-collectors http://[UCMDB_SERVER_NAME]:8080/mam-collectors
ProxyPass /ucmdb http://[UCMDB_SERVER_NAME]:8080/ucmdb
ProxyPassReverse /ucmdb http://[UCMDB_SERVER_NAME]:8080/ucmdb
ProxyPass /site http://[UCMDB_SERVER_NAME]:8080/site
ProxyPassReverse /site http://[UCMDB_SERVER_NAME]:8080/site
ProxyPass /ucmdb-ui http://[UCMDB_SERVER_NAME]:8080/ucmdb-ui
ProxyPassReverse /ucmdb-ui http://[UCMDB_SERVER_NAME]:8080/ucmdb-ui
ProxyPass /status http://[UCMDB_SERVER_NAME]:8080/status
ProxyPassReverse /status http://[UCMDB_SERVER_NAME]:8080/status
ProxyPass /jmx-console http://[UCMDB_SERVER_NAME]:8080/jmx-console
ProxyPassReverse /jmx-console http://[UCMDB_SERVER_NAME]:8080/jmx-console
ProxyPass /axis2 http://[UCMDB_SERVER_NAME]:8080/axis2
ProxyPassReverse /axis2 http://[UCMDB_SERVER_NAME]:8080/axis2
ProxyPass /icons http://[UCMDB_SERVER_NAME]:8080/icons
ProxyPassReverse /icons http://[UCMDB_SERVER_NAME]:8080/icons
ProxyPass /ucmdb-api http://[UCMDB_SERVER_NAME]:8080/ucmdb-api
ProxyPassReverse /ucmdb-api http://[UCMDB_SERVER_NAME]:8080/ucmdb-api
ProxyPass /ucmdb-docs http://[UCMDB_SERVER_NAME]:8080/ucmdb-docs
ProxyPassReverse /ucmdb-docs http://[UCMDB_SERVER_NAME]:8080/ucmdb-docs
ProxyPass /ucmdb-api/8.0 http://[UCMDB_SERVER_NAME]:8080/ucmdb-api/8.0
ProxyPassReverse /ucmdb-api/8.0 http://[UCMDB_SERVER_NAME]:8080/ucmdb-api/8.0
ProxyPass /cm http://[UCMDB_SERVER_NAME]:8080/cm
ProxyPassReverse /cm http://[UCMDB_SERVER_NAME]:8080/cm
ProxyPass /cnc http://[UCMDB_CM_SERVER_NAME]/cnc
ProxyPassReverse /cnc http://[UCMDB_CM_SERVER_NAME]/cnc
ProxyPass /docs http://[UCMDB_CM_SERVER_NAME]/docs
ProxyPassReverse /docs http://[UCMDB_CM_SERVER_NAME]/docs
ProxyPass /ucmdb-browser http://[UCMDB_CM_SERVER_NAME]/ucmdb-browser
ProxyPassReverse /ucmdb-browser http://[UCMDB_CM_SERVER_NAME]/ucmdb-browser
</VirtualHost>
```

이제 https://[APACHE_MACHINE_FQD]로 이동하여 리버스 프록시를 통해 UCMDB 서버에 액세스할 수 있습니다.

참고: Internet Explorer에서 가져온 유효한 인증서가 있어야 합니다. 유효한 인증서는 Apache에서

신뢰할 수 있는 CA의 CA가 서명한 인증서입니다(**ssl.crt** 파일에 있어야 함).

4장: 데이터 흐름 자격 증명 관리

이 장의 내용:

데이터 흐름 자격 증명 관리 개요

디스커버리 또는 실행 통합을 수행하려면 원격 시스템에 액세스하는 데 사용할 자격 증명을 설정해야 합니다. 자격 증명은 Data Flow Probe 설정 창에서 구성되어 UCMDB 서버에 저장됩니다. 자세한 내용은 *HP Universal CMDB 데이터 흐름 관리 안내서*에서 Data Flow Probe 설정에 대한 설명 섹션을 참조하십시오.

자격 증명 저장소는 Confidential Manager 구성 요소를 통해 관리됩니다. 자세한 내용은 "Confidential Manager"(94페이지)를 참조하십시오.

Data Flow Probe는 Confidential Manager 클라이언트를 사용하여 자격 증명에 액세스할 수 있습니다. Confidential Manager 클라이언트는 Data Flow Probe에 있으며 UCMDB 서버에 있는 Confidential Manager 서버와 통신합니다. Confidential Manager 클라이언트와 Confidential Manager 서버 간의 통신은 암호화되며, Confidential Manager 클라이언트가 Confidential Manager 서버에 연결할 때 Confidential Manager 클라이언트에서 인증을 해야 합니다.

Confidential Manager 서버의 Confidential Manager 클라이언트 인증은 LW-SSO 구성 요소를 기반으로 합니다. Confidential Manager 서버에 연결하기 전에 Confidential Manager 클라이언트는 먼저 LW-SSO 쿠키를 보냅니다. 그러면 Confidential Manager 서버에서 쿠키를 확인하며, 확인이 성공하면 Confidential Manager 클라이언트와의 통신이 시작됩니다. LW-SSO에 대한 자세한 내용은 "LW-SSO 설정 구성"(52페이지)을 참조하십시오.

Confidential Manager 클라이언트와 Confidential Manager 서버 간의 통신은 암호화됩니다. 암호화 구성 업데이트에 대한 자세한 내용은 "Confidential Manager 통신 암호화 구성 "(52페이지)을 참조하십시오.

주의: Confidential Manager 인증에는 UTC(컴퓨터에 정의된 범용 시간)가 사용됩니다. 인증에 성공하려면 Data Flow Probe와 UCMDB 서버에 있는 범용 시간이 일치해야 합니다. UTC는 시간대나 일광 절약 시간제에 대해 독립적이기 때문에 서버와 프로브가 서로 다른 시간대에 있을 수도 있습니다.

Confidential Manager 클라이언트는 자격 증명의 로컬 캐시를 유지합니다. Confidential Manager 클라이언트는 Confidential Manager 서버에서 모든 자격 증명을 다운로드하여 캐시에 저장하도록 구성됩니다. 자격 증명 변경 내용은 Confidential Manager 서버에서 지속적으로 자동 동기화됩니다. 미리 구성된 설정에 따라 캐시는 파일 시스템 또는 메모리 내 캐시일 수 있습니다. 또한 캐시는 암호화되며 외부에서 액세스할 수 없습니다. 캐시 설정 업데이트에 대한 자세한 내용은 "프로브에서 Confidential Manager 클라이언트의 캐시 모드 구성"(56페이지)을 참조하십시오. 캐시 암호화 업데이트에 대한 자세한 내용은 "프로브에서 Confidential Manager 클라이언트의 캐시 암호화 설정 구성"(57페이지)을 참조하십시오.

문제 해결에 대한 자세한 내용은 "Confidential Manager 클라이언트 로그 파일 메시지 수준 변경"(59페이지)을 참조하십시오.

UCMDB 서버 간에 자격 증명 정보를 복사할 수 있습니다. 자세한 내용은 "암호화된 형식으로 자격 증명과 범위 정보 내보내기 및 가져오기"(58페이지)를 참조하십시오.

참고: UCMDB 버전 9.01 이하의 프로브에서 자격 증명 저장소에 사용되었던 DSD (**DomainScopeDocument**)는 더 이상 자격 증명과 관련된 중요한 정보를 포함하지 않습니다. 이제 이 파일은 Probe 목록과 네트워크 범위 정보를 포함합니다. 또한 각 도메인의 자격 증명 항목 목록도

포함합니다. 여기서 각 항목에는 자격 증명 ID와 해당 자격 증명 항목에 대해 정의된 네트워크 범위만 포함됩니다.

이 섹션에는 다음 항목이 포함됩니다.

보안 관련 기본 가정 사항

이 문서는 다음과 같이 보안을 설정했다는 가정 하에 작성되었습니다.

UCMDB 시스템 관리자에 한해 기본적으로 localhost 액세스를 통해서만 액세스할 수 있도록 UCMDB 서버 및 프로브 JMX 콘솔의 보안을 설정했습니다.

별도의 모드에서 실행되는 Data Flow Probe

프로브 게이트웨이와 관리자가 별도의 프로세스로 실행되는 경우 Confidential Manager 클라이언트 구성 요소가 관리자 프로세스에 포함됩니다. 자격 증명 정보는 캐시되어 프로브 관리자만 사용할 수 있습니다. UCMDB 시스템에서 Confidential Manager 서버에 액세스하기 위해 Confidential Manager 클라이언트 요청이 게이트웨이 프로세스에서 처리된 다음 UCMDB 시스템으로 전달됩니다.

프로브를 별도의 모드에서 구성하는 경우 이 구성은 자동으로 적용됩니다.

지속적인 자격 증명 캐시 업데이트

Confidential Manager 클라이언트가 처음으로 Confidential Manager 서버에 연결하는 데 성공하면 모든 관련 자격 증명(프로브의 도메인에 구성되어 있는 모든 자격 증명)을 다운로드합니다. 첫 통신에 성공한 후에 Confidential Manager 클라이언트는 Confidential Manager 서버와 지속적으로 동기화 상태를 유지합니다. 1분 간격으로 차등 동기화가 수행되며, 동기화 중에는 Confidential Manager 서버와 Confidential Manager 클라이언트 간의 차이점만 동기화됩니다. UCMDB 서버 쪽에서 새 자격 증명이 추가되거나 기존 자격 증명이 업데이트 또는 삭제되는 등 자격 증명이 변경되는 경우 Confidential Manager 클라이언트는 UCMDB 서버에서 즉시 알림을 받으며 추가 동기화를 수행합니다.

구성 변경 내용으로 모든 프로브 동기화

성공적인 통신을 수행하려면 Confidential Manager 서버 인증 구성(LW-SSO 초기 문자열) 및 암호화 구성(Confidential Manager 통신 암호화)으로 Confidential Manager 클라이언트를 업데이트해야 합니다. 예

를 들어 서버에서 초기 문자열이 변경되면 프로브에서 새로운 초기 문자열을 인식해야 인증을 할 수 있습니다.

UCMDB 서버는 Confidential Manager 통신 암호화 구성 및 Confidential Manager 인증 구성의 변경 내용을 지속적으로 모니터링합니다. 모니터링은 15초 간격으로 수행합니다. 변경 내용이 있으면 업데이트된 구성이 프로브로 전송됩니다. 구성은 암호화된 형식으로 프로브에 전달되어 프로브 쪽의 보안 저장소에 저장됩니다. 보내는 구성은 대칭 암호화 키를 사용하여 암호화됩니다. 기본적으로 UCMDB 서버와 Data Flow Probe는 같은 기본 대칭 암호화 키를 사용하여 설치됩니다. 보안을 최적화하려면 자격 증명을 시스템에 추가하기 전에 이 키를 변경하는 것이 좋습니다. 자세한 내용은 "암호화 키 생성 또는 업데이트"(60 페이지)를 참조하십시오.

참고: 모니터링은 15초 간격으로 수행되므로, 프로브 쪽의 Confidential Manager 클라이언트가 15초 동안 최신 구성으로 업데이트되지 않을 수 있습니다.

UCMDB 서버와 Data Flow Probe 간의 인증 구성 및 Confidential Manager 통신 자동 동기화를 사용하지 않도록 설정하는 경우에는 UCMDB 서버 쪽에서 인증 구성 및 CM 통신을 업데이트할 때마다 새 구성을 사용하여 모든 프로브를 업데이트하는 작업도 수행해야 합니다. 자세한 내용은 "서버와 프로브 간의 암호화 설정 및 Confidential Manager 클라이언트 인증 자동 동기화를 사용하지 않도록 설정"(54페이지)을 참조하십시오.

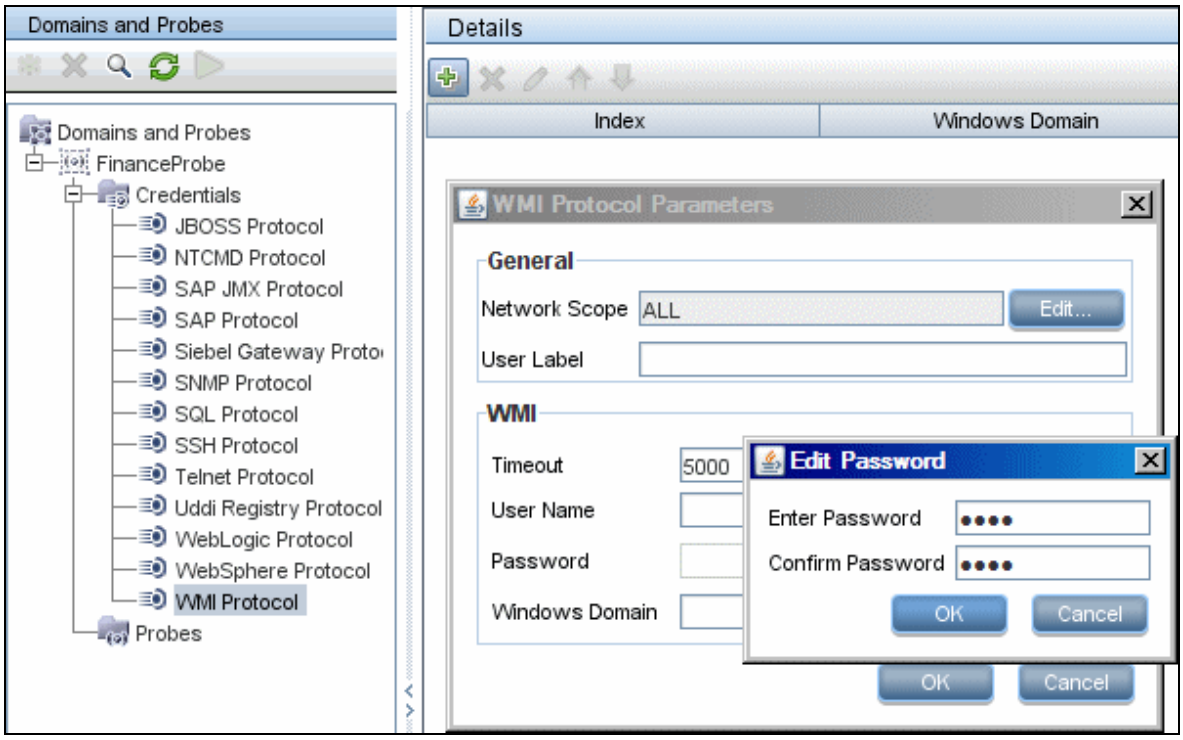
프로브의 보안 저장소

Confidential Manager 통신 및 인증 구성과 암호화 키 등의 중요한 정보는 모두 프로브 보안 저장소의 **C:\hp\UCMDB\DataFlowProbe\conf\security**에 있는 **secured_storage.bin** 파일에 저장됩니다. 이 보안 저장소는 암호화 프로세스에서 Windows 사용자 비밀번호를 사용하는 DPAPI를 통해 암호화됩니다. DPAPI는 Windows 시스템에서 인증서, 개인 키 등의 기밀 데이터를 보호하는 데 사용되는 표준 방법입니다. 프로브는 항상 같은 Windows 사용자로 실행해야 합니다. 그래야 비밀번호가 변경되어도 프로브에서 보안 저장소에 저장된 정보를 읽을 수 있습니다.

자격 증명 정보 보기

참고: 이 섹션에서는 데이터가 CMDB에서 HP Universal CMDB로 이동할 때 자격 증명 정보를 보는 방법에 대해 설명합니다.

비밀번호는 CMDB에서 응용 프로그램으로 전송되지 않습니다. 즉, HP Universal CMDB의 비밀번호 필드에는 내용에 관계없이 별표(*)가 표시됩니다.



자격 증명 업데이트

참고: 이 섹션에서는 데이터가 HP Universal CMDB에서 CMDB으로 이동할 때 자격 증명을 업데이트하는 방법을 설명합니다.

- 이 방향의 통신은 암호화되지 않으므로 https\SSL을 사용하여 UCMDB 서버에 연결하거나, 신뢰할 수 있는 네트워크를 통해 연결해야 합니다.

  통신은 암호화되지 않지만 네트워크에서 비밀번호는 일반 텍스트로 전송되지 않습니다. 즉, 비밀번호는 기본 키를 사용하여 암호화되므로 비밀번호 전송 시에 기밀을 효율적으로 유지하려면 SSL을 사용하는 것이 좋습니다.
- 특수 문자와 영어가 아닌 문자를 비밀번호로 사용할 수 있습니다.

Confidential Manager 클라이언트 인증 및 암호화 설정 구성

이 작업은 UCMDB 서버에 Confidential Manager 클라이언트 인증 및 암호화 설정을 구성하는 방법을 설명하며, 다음 단계가 포함되어 있습니다.

- "LW-SSO 설정 구성"(52페이지)
- "Confidential Manager 통신 암호화 구성 "(52페이지)

LW-SSO 설정 구성

이 절차에서는 UCMDB 서버에서 LW-SSO 초기 문자열을 변경하는 방법을 설명합니다. 이 변경 내용을 암호화된 문자열로 프로브에 자동으로 보냅니다. 단, UCMDB 서버가 자동 보내기를 수행하지 않도록 구성된 경우에는 변경 내용을 자동으로 보내지 않습니다. 자세한 내용은 "서버와 프로브 간의 암호화 설정 및 Confidential Manager 클라이언트 인증 자동 동기화를 사용하지 않도록 설정"(54페이지)을 참조하십시오.

1. UCMDB 서버에서 웹 브라우저를 시작하고 주소창에 **http://localhost:8080/jmx-console**을 입력합니다.
2. **UCMDB-UI:name=LW-SSO Configuration**을 클릭하여 JMX MBEAN 보기 페이지를 엽니다.
3. **setInitString** 메서드를 찾습니다.
4. 새 LW-SSO 초기 문자열을 입력합니다.
5. Invoke를 클릭합니다.

Confidential Manager 통신 암호화 구성

이 절차는 UCMDB 서버에서 Confidential Manager 통신 암호화 설정을 변경하는 방법을 설명합니다. 이러한 설정은 Confidential Manager 클라이언트와 Confidential Manager 서버 간의 통신이 암호화되는 방법을 지정합니다. 이 변경 내용을 암호화된 문자열로 프로브에 자동으로 보냅니다. 단, UCMDB 서버가 자동 보내기를 수행하지 않도록 구성된 경우에는 변경 내용을 자동으로 보내지 않습니다. 자세한 내용은 "서버와 프로브 간의 암호화 설정 및 Confidential Manager 클라이언트 인증 자동 동기화를 사용하지 않도록 설정"(54페이지)을 참조하십시오.

1. UCMDB 서버에서 웹 브라우저를 시작하고 주소창에 **http://localhost:8080/jmx-console**을 입력합니다.
2. **UCMDB:service=Security Services**를 클릭하여 JMX MBEAN 보기 페이지를 엽니다.
3. **CMGetConfiguration** 메서드를 클릭합니다.
4. **Invoke**를 클릭합니다.

현재 Confidential Manager 구성의 XML이 표시됩니다.

5. 표시된 XML의 내용을 복사합니다.
6. 다시 **Security Services** JMX MBean 보기 페이지로 돌아옵니다.
7. **CMSetConfiguration** 메서드를 클릭합니다.
8. 복사한 XML을 **Value** 필드에 붙여 넣습니다.
9. 관련된 전송 관련 설정을 업데이트하고 **Invoke**를 클릭합니다.

   예:

```
<transport>
	<encryptTransportMode>true</encryptTransportMode>
	<CMEncryptionDecryption>
		<encryptDecryptInitString>radiohead</encryptDecryptInitString>
		<cryptoSource>lw</cryptoSource>
		<lwJCEPBECompatibilityMode>true</lwJCEPBECompatibilityMode>
		<cipherType>symmetricBlockCipher</cipherType>
		<engineName>AES</engineName>
		<algorithmModeName>CBC</algorithmModeName>
		<algorithmPaddingName>PKCS7Padding</algorithmPaddingName>
		<keySize>256</keySize>
		<pbeCount>20</pbeCount>
		<pbeDigestAlgorithm>SHA1</pbeDigestAlgorithm>
		<encodingMode>Base64Url</encodingMode>
		<useMacWithCrypto>false</useMacWithCrypto>
		<macType>hmac</macType>
		<macKeySize>256</macKeySize>
		<macHashName>SHA256</macHashName>
	</CMEncryptionDecryption>
</transport>
```

업데이트할 수 있는 값에 대한 자세한 내용은 "Confidential Manager 암호화 설정"(60페이지)을 참조하십시오.

프로브에서 Confidential Manager 클라이언트 인증 및 암호화 설정 수동 구성

이 작업에는 다음 단계가 포함됩니다.

서버와 프로브 간의 암호화 설정 및 Confidential Manager 클라이언트 인증 자동 동기화를 사용하지 않도록 설정

기본적으로 UCMDB 서버는 Confidential Manager/LW-SSO 설정을 모든 프로브에 자동으로 보내도록 구성됩니다. 이 정보는 프로브에 암호화된 문자열로 전송되며, 프로브에서는 정보를 검색하는 즉시 암호를 해독합니다. Confidential Manager/LW-SSO 구성 파일을 모든 프로브에 자동으로 보내지 않도록 UCMDB 서버를 구성할 수 있습니다. 이 경우에는 새 Confidential Manager/LW-SSO 설정을 사용하여 모든 프로브를 수동으로 업데이트해야 합니다.

Confidential Manager/LW-SSO 설정 자동 동기화를 사용하지 않도록 설정하려면 다음을 수행합니다.

1. UCMDB에서 **관리 > 인프라 설정 관리자 > 일반 설정**을 클릭합니다.
2. **CM/LW-SSO 구성 및 초기 문자열과 프로브 자동 동기화 사용**을 선택합니다.
3. **값** 필드를 클릭하고 **True**를 **False**로 변경합니다.
4. **저장** 버튼을 클릭합니다.
5. UCMDB 서버를 다시 시작합니다.

프로브에서 Confidential Manager 클라이언트 인증 및 암호화 설정 구성

LW-SSO/Confidential Manager 구성 및 설정을 프로브에 자동으로 보내지 않도록 UCMDB 서버를 구성한 경우 이 절차를 수행합니다. 자세한 내용은 "서버와 프로브 간의 암호화 설정 및 Confidential Manager 클라이언트 인증 자동 동기화를 사용하지 않도록 설정"(54페이지)을 참조하십시오.

1. 프로브 컴퓨터에서 웹 브라우저를 시작하고 주소창에 **http://localhost:1977**을 입력합니다.

   참고: 프로브 관리자 및 프로브 게이트웨이가 각각 별도의 프로세스로 실행되는 경우에는 프로브 관리자를 실행하는 컴퓨터에서 주소를 **http://localhost:1978**과 같이 입력해야 합니다.

2. **type=CMClient**를 클릭하여 JMX MBEAN 보기 페이지를 엽니다.
3. **setLWSSOInitString** 메서드를 찾아서 UCMDB의 LW-SSO 구성에 대해 제공한 것과 같은 초기 문자열을 제공합니다.
4. **setLWSSOInitString** 버튼을 클릭합니다.

프로브에서 Confidential Manager 통신 암호화 구성

LW-SSO/Confidential Manager 구성 및 설정을 프로브에 자동으로 보내지 않도록 UCMDB 서버를 구성한 경우 이 절차를 수행합니다. 자세한 내용은 "서버와 프로브 간의 암호화 설정 및 Confidential Manager 클라이언트 인증 자동 동기화를 사용하지 않도록 설정"(54페이지)를 참조하십시오.

1. 프로브 컴퓨터에서 웹 브라우저를 시작하고 주소창에 **http://localhost:1977**을 입력합니다.

   참고: 프로브 관리자 및 프로브 게이트웨이가 각각 별도의 프로세스로 실행되는 경우에는 프로브 관리자를 실행하는 컴퓨터에서 주소를 **http://localhost:1978**과 같이 입력해야 합니다.

2. **type=CMClient**를 클릭하여 JMX MBEAN 보기 페이지를 엽니다.
3. 다음의 전송 관련 설정을 업데이트합니다.

   참고: UCMDB 서버에서 업데이트했던 설정을 업데이트해야 합니다. 이렇게 하려면 프로브에서 업데이트하는 메서드 중 일부에 여러 매개 변수가 필요할 수 있습니다. 현재 프로브 구성을 보려면 JMX MBEAN 보기 페이지에서 **displayTransportConfiguration**을 클릭합니다. 자세한 내용은 "Confidential Manager 통신 암호화 구성 "(52페이지)을 참조하십시오. 업데이트할 수 있는 값에 대한 자세한 내용은 "Confidential Manager 암호화 설정"(60페이지)을 참조하십시오.

   a. **setTransportInitString**은 **encryptDecryptInitString** 설정을 변경합니다.
   b. **setTransportEncryptionAlgorithm**은 다음 맵에 따라 프로브의 Confidential Manager 설정을 변경합니다.
      - **엔진 이름**은 <engineName> 항목을 참조합니다.
      - **키 크기**는 <keySize> 항목을 참조합니다.
      - **알고리즘 패딩 이름**은 <algorithmPaddingName> 항목을 참조합니다.
      - **PBE 개수**는 <pbeCount> 항목을 참조합니다.
      - **PBE 다이제스트 알고리즘**은 <pbeDigestAlgorithm> 항목을 참조합니다.
   c. **setTransportEncryptionLibrary**는 다음 맵에 따라 프로브의 Confidential Manager 설정을 변경합니다.
      - **암호화 라이브러리 이름**은 <cryptoSource> 항목을 참조합니다.
      - **이전의 경량 암호화 버전 지원**은 <lwJCEPBECompatibilityMode> 항목을 참조합니다.

d. **setTransportMacDetails**는 다음 맵에 따라 프로브의 Confidential Manager 설정을 변경합니다.
   - **MAC 사용(암호화 포함)**은 <useMacWithCrypto> 항목을 참조합니다.
   - **MAC 키 크기**는 <macKeySize> 항목을 참조합니다.

4. **reloadTransportConfiguration** 버튼을 클릭하여 프로브에 변경 내용을 적용합니다.

서로 다른 각 설정 및 가능한 값에 대한 자세한 내용은 "Confidential Manager 암호화 설정"(60페이지)을 참조하십시오.

Confidential Manager 클라이언트 캐시 구성

이 작업에는 다음 단계가 포함됩니다.

- "프로브에서 Confidential Manager 클라이언트의 캐시 모드 구성"(56페이지)
- "프로브에서 Confidential Manager 클라이언트의 캐시 암호화 설정 구성"(57페이지)

프로브에서 Confidential Manager 클라이언트의 캐시 모드 구성

Confidential Manager 클라이언트는 캐시에 자격 증명 정보를 저장하며, 서버에서 정보가 변경되면 저장된 정보를 업데이트합니다. 캐시는 파일 시스템이나 메모리 내에 저장할 수 있습니다.

- **파일 시스템에 저장하는 경우** 프로브가 다시 시작되어 서버에 연결할 수 없는 경우에도 자격 증명 정보는 계속 사용할 수 있습니다.
- **메모리 내에 저장하는 경우** 프로브가 다시 시작되면 캐시가 지워지고 모든 정보를 서버에서 다시 검색합니다. 서버를 사용할 수 없는 경우 프로브에 자격 증명 정보가 포함되지 않으므로 디스커버리 또는 통합을 실행할 수 없습니다.

이 설정을 변경하려면 다음을 수행합니다.

1. 텍스트 편집기에서 **DataFlowProbe.properties** 파일을 엽니다. 이 파일은 **c:\hp\UCMDB\DataFlowProbe\conf** 폴더에 있습니다.
2. 다음 특성을 찾습니다. **com.hp.ucmdb.discovery.common.security.storeCMData=true**
   - 파일 시스템에 정보를 저장하려면 기본값(**true**)을 그대로 둡니다.
   - 메모리 내에 정보를 저장하려면 **false**를 입력합니다.
3. **DataFlowProbe.properties** 파일을 저장합니다.
4. 프로브를 다시 시작합니다.

프로브에서 Confidential Manager 클라이언트의 캐시 암호화 설정 구성

이 절차에서는 Confidential Manager 클라이언트 파일 시스템 캐시 파일의 암호화 설정을 변경하는 방법을 설명합니다. Confidential Manager 클라이언트 파일 시스템 캐시의 암호화 설정을 변경하면 파일 시스템 캐시 파일이 다시 만들어집니다. 이 다시 만들기 프로세스에서는 프로브를 다시 시작해야 하며 UCMDB 서버와의 전체 동기화를 수행해야 합니다.

1. 프로브 컴퓨터에서 웹 브라우저를 시작하고 주소창에 **http://localhost:1977**을 입력합니다.

   참고: 프로브 관리자 및 프로브 게이트웨이가 각각 별도의 프로세스로 실행되는 경우에는 프로브 관리자를 실행하는 컴퓨터에서 주소를 **http://localhost:1978**과 같이 입력해야 합니다.

2. **type=CMClient**를 클릭하여 JMX MBEAN 보기 페이지를 엽니다.
3. 다음의 캐시 관련 설정을 업데이트합니다.

   참고: 프로브에서 업데이트하는 메서드 중 일부에 여러 매개 변수가 필요할 수 있습니다. 현재 프로브 구성을 보려면 JMX MBEAN 보기 페이지에서 **displayCacheConfiguration**을 클릭합니다.

   a. **setCacheInitString**은 파일 시스템 캐시 <encryptDecryptInitString> 설정을 변경합니다.
   b. **setCacheEncryptionAlgorithm**은 다음 맵에 따라 파일 시스템 캐시 설정을 변경합니다.
      - **엔진 이름**은 <engineName> 항목을 참조합니다.
      - **키 크기**는 <keySize> 항목을 참조합니다.
      - **알고리즘 패딩 이름**은 <algorithmPaddingName> 항목을 참조합니다.
      - **PBE 개수**는 <pbeCount> 항목을 참조합니다.
      - **PBE 다이제스트 알고리즘**은 <pbeDigestAlgorithm> 항목을 참조합니다.
   c. **setCacheEncryptionLibrary**는 다음 맵에 따라 캐시 파일 시스템 설정을 변경합니다.
      - **암호화 라이브러리 이름**은 <cryptoSource> 항목을 참조합니다.
      - **이전의 경량 암호화 버전 지원**은 <lwJCEPBECompatibilityMode> 항목을 참조합니다.
   d. **setCacheMacDetails**는 다음 맵에 따라 캐시 파일 시스템 설정을 변경합니다.
      - **MAC 사용(암호화 포함)**은 <useMacWithCrypto> 항목을 참조합니다.
      - **MAC 키 크기**는 <macKeySize> 항목을 참조합니다.
4. **reloadCacheConfiguration** 버튼을 클릭하여 프로브에 변경 내용을 적용합니다. 그러면 프로브가 다시 시작됩니다.

   참고: 이 작업 중에는 프로브에서 실행되고 있는 작업이 없어야 합니다.

서로 다른 각 설정 및 가능한 값에 대한 자세한 내용은 "Confidential Manager 암호화 설정"(60페이지)을 참조하십시오.

암호화된 형식으로 자격 증명과 범위 정보 내보내기 및 가져오기

UCMDB 서버 간에 자격 증명 정보를 복사하기 위해 자격 증명 및 네트워크 범위 정보를 암호화된 형식으로 내보내고 가져올 수 있습니다. 예를 들어 시스템 크래시 이후의 복구 또는 업그레이드 중에 이 작업을 수행할 수 있습니다.

- **자격 증명 정보를 내보내는 경우** 원하는 비밀번호를 입력해야 합니다. 그러면 정보가 해당 비밀번호를 사용하여 암호화됩니다.
- **자격 증명 정보를 가져오는 경우** DSD 파일을 내보낼 때 정의한 비밀번호를 사용해야 합니다.

참고: 내보낸 자격 증명 문서에는 내보낸 문서가 원래 있었던 시스템에서 정의한 범위 정보도 포함되어 있습니다. 자격 증명 문서를 가져올 때 범위 정보도 가져옵니다.

UCMDB 서버에서 자격 증명 정보를 내보내려면 다음을 수행합니다.

1. UCMDB 서버에서 웹 브라우저를 시작하고 주소창에 **http://localhost:8080/jmx-console**을 입력합니다. 사용자 이름과 비밀번호를 입력하여 로그인해야 할 수도 있습니다.
2. **UCMDB:service=DiscoveryManager**를 클릭하여 JMX MBEAN 보기 페이지를 엽니다.
3. **exportCredentialsAndRangesInformation** 작업을 찾습니다. 다음을 수행합니다.
   - 고객 ID(기본값: 1)를 입력합니다.
   - 내보낸 파일의 이름을 입력합니다.
   - 비밀번호를 입력합니다.
   - 내보낸 파일을 제공된 비밀번호로 암호화하려면 **isEncrypted=True**로 설정하고, 내보낸 파일을 암호화하지 않으려면 **isEncrypted=False**로 설정합니다. False로 설정하는 경우에는 비밀번호와 기타 중요한 정보를 내보내지 않습니다.
4. **Invoke**를 클릭하여 내보냅니다.

   내보내기 프로세스가 성공적으로 완료되면 파일이 **c:\hp\UCMDB\UCMDBServer\conf\discovery\<customer_dir>** 위치에 저장됩니다.

UCMDB 서버에서 자격 증명 정보를 가져오려면 다음을 수행합니다.

1. UCMDB 서버에서 웹 브라우저를 시작하고 주소창에 **http://localhost:8080/jmx-console**을 입력합니다.

   사용자 이름과 비밀번호를 입력하여 로그인해야 할 수도 있습니다.
2. **UCMDB:service=DiscoveryManager**를 클릭하여 JMX MBEAN 보기 페이지를 엽니다.
3. **importCredentialsAndRangesInformation** 작업을 찾습니다.
4. 고객 ID(기본값: 1)를 입력합니다.

5. 가져올 파일의 이름을 입력합니다. 이 파일은 **c:\hp\UCMDB\UCMDBServer\conf\discovery\<customer_dir>**에 있어야 합니다.
6. 비밀번호를 입력합니다. 이 비밀번호는 파일을 내보낼 때 사용한 비밀번호와 같아야 합니다.
7. **Invoke**를 클릭하여 자격 증명을 가져옵니다.

Confidential Manager 클라이언트 로그 파일 메시지 수준 변경

프로브는 Confidential Manager 서버와 Confidential Manager 클라이언트 간의 Confidential Manager 관련 통신에 대한 정보를 포함하는 두 개의 로그 파일을 제공합니다.

- "Confidential Manager 클라이언트 로그 파일"(59페이지)
- "LW-SSO 로그 파일"(60페이지)

Confidential Manager 클라이언트 로그 파일

security.cm.log 파일은 **c:\hp\UCMDB\DataFlowProbe\runtime\log** 폴더에 있습니다.

로그에는 Confidential Manager 서버와 Confidential Manager 클라이언트 간에 교환한 정보 메시지가 포함되어 있습니다. 기본적으로 이러한 메시지의 로그 수준은 INFO로 설정됩니다.

메시지의 로그 수준을 DEBUG 수준으로 변경하려면 다음을 수행합니다.

1. Data Flow Probe 관리자 서버에서 **c:\hp\UCMDB\DataFlowProbe\conf\log**로 이동합니다.
2. **security.properties** 파일을 텍스트 편집기에서 엽니다.
3. 다음 줄을

   loglevel.cm=INFO

   다음과 같이 변경합니다.

   loglevel.cm=DEBUG
4. 파일을 저장합니다.

LW-SSO 로그 파일

security.lwsso.log 파일은 **c:\hp\UCMDB\DataFlowProbe\runtime\log** 폴더에 있습니다.

로그에는 LW-SSO와 관련된 정보 메시지가 포함되어 있습니다. 기본적으로 이러한 메시지의 로그 수준은 INFO로 설정됩니다.

메시지의 로그 수준을 DEBUG 수준으로 변경하려면 다음을 수행합니다.

1. Data Flow Probe 관리자 서버에서 **c:\hp\UCMDB\DataFlowProbe\conf\log**로 이동합니다.
2. security.properties 파일을 텍스트 편집기에서 엽니다.
3. 다음 줄을

   loglevel.lwsso=INFO

   다음과 같이 변경합니다.

   loglevel.lwsso=DEBUG
4. 파일을 저장합니다.

암호화 키 생성 또는 업데이트

(missing or bad snippet)

Confidential Manager 암호화 설정

아래 표에는 다양한 JMX 메서드를 사용하여 변경할 수 있는 암호화 설정이 나와 있습니다. 이러한 암호화 설정은 Confidential Manager 클라이언트와 Confidential Manager 서버 간의 통신을 암호화할 때와 Confidential Manager 클라이언트의 캐시를 암호화할 때 사용됩니다.

Confidential Manager 설정 이름	프로브 Confidential Manager 설정 이름	설정 설명	사용 가능한 값	기본값
cryptoSource	암호화 라이브러리 이름	이 설정은 사용할 암호화 라이브러리를 결정합니다.	lw, jce, windowsDPAPI, lwJCECompatible	lw
lwJCEPBE 호환성 모드	이전의 경량 암호화 버전 지원	이 설정은 이전의 경량 암호화 지원 여부를 정의합니다.	true, false	true
engineName	엔진 이름	암호화 메커니즘 이름입니다.	AES, DES, 3DES, Blowfish	AES

Confidential Manager 설정 이름	프로브 Confidential Manager 설정 이름	설정 설명	사용 가능한 값	기본값
keySize	키 크기	암호화 키 길이(비트 단위)입니다.	AES - 128, 192, 256. DES - 64. 3DES - 192. Blowfish - 32-448 사이의 숫자	256
알고리즘 패딩 이름	알고리즘 패딩 이름	패딩 표준입니다.	PKCS7Padding, PKCS5Padding	PKCS7Padding
pbeCount	PBE 개수	비밀번호(초기 문자열)에서 키를 만들기 위해 해시를 실행할 횟수입니다.	임의의 양수	20
pbeDigest 알고리즘	PBE 다이제스트 알고리즘	해시 유형입니다.	SHA1, SHA256, MD5	SHA1
useMacWith 암호	MAC 사용(암호화 포함)	암호화를 포함하여 MAC를 사용할지 여부를 나타냅니다.	true, false	false
macKeySize	MAC 키 크기	MAC 알고리즘에 따라 달라집니다.	256	256

문제 해결 및 제한 사항

UCMDB 서버에서 기본 도메인 이름을 변경한 경우에는 먼저 Data Flow Probe가 실행되고 있지 않은지 확인해야 합니다. 기본 도메인 이름을 적용한 후에 Data Flow Probe 쪽에서 **DataFlowProbe\tools\clearProbeData.bat** 스크립트를 실행해야 합니다.

참고: clearProbeData.bat 스크립트를 실행하면 프로브가 시작된 후 프로브 쪽에서 디스커버리 주기가 시작됩니다.

5장: Data Flow Probe 강화

이 장의 내용:

PostgreSQL 데이터베이스 암호화된 비밀번호 수정

이 섹션에서는 PostgreSQL 데이터베이스 사용자의 암호화된 비밀번호를 수정하는 방법을 설명합니다.

1. 암호화된 형식의 비밀번호 만들기(AES, 192비트 키)
   a. Data Flow Probe JMX 콘솔에 액세스합니다. 웹 브라우저를 시작하고 주소창에 **http://<Data Flow Probe 컴퓨터 이름 또는 IP 주소>:1977**을 입력합니다. Data Flow Probe를 로컬로 실행하는 경우에는 **http://localhost:1977**을 입력합니다.

   사용자 이름과 비밀번호를 입력하여 로그인해야 할 수도 있습니다.

> **참고:** 사용자를 만들지 않은 경우 기본 사용자 이름 sysadmin 및 비밀번호 sysadmin을 사용하여 로그인합니다.

b. **Type=MainProbe** 서비스를 찾은 다음 링크를 클릭하여 작업 페이지를 엽니다.

c. **getEncryptedDBPassword** 작업을 찾습니다.

d. **DB Password** 필드에 암호화할 비밀번호를 입력합니다.

e. **getEncryptedDBPassword** 버튼을 클릭하여 작업을 호출합니다.

호출 결과로 다음과 같은 암호화된 비밀번호 문자열이 생성됩니다.

66,85,54,78,69,117,56,65,99,90,86,117,97,75,50,112,65,53,67,114,112,65,61,61

2. Data Flow Probe 중지

시작 > 모든 프로그램 > HP UCMDB > Data Flow Probe 중지

3. set_dbuser_password.cmd 스크립트 실행

이 스크립트는 **C:\hp\UCMDB\DataFlowProbe\tools\dbscripts\set_dbuser_password.cmd** 폴더에 있습니다.

새 비밀번호를 첫 번째 인수로 사용하고 PostgreSQL 루트 계정 비밀번호를 두 번째 인수로 사용하여 **set_dbuser_password.cmd** 스크립트를 실행합니다.

예를 들면 다음과 같습니다.

set_dbuser_password <my_password><root_password>

비밀번호는 암호화되지 않은 형식(일반 텍스트)으로 입력해야 합니다.

4. Data Flow Probe 구성 파일에서 비밀번호 업데이트

a. 비밀번호는 구성 파일에서 암호화되어 있어야 합니다. 암호화된 비밀번호 형식을 검색하려면 단계 1에 설명된 대로 **getEncryptedDBPassword** JMX 메서드를 사용합니다.

b. 암호화된 비밀번호를 **C:\hp\UCMDB\DataFlowProbe\conf\DataFlowProbe.properties** 파일의 다음 속성에 추가합니다.

- **appilog.agent.probe.jdbc.pwd**

  예를 들면 다음과 같습니다.

  appilog.agent.probe.jdbc.user = mamprobe
  appilog.agent.probe.jdbc.pwd =
  66,85,54,78,69,117,56,65,99,90,86,117,97,75,50,112,65,53,67,114,112,65,61,61

- **appilog.agent.local.jdbc.pwd**
- **appilog.agent.normalization.jdbc.pwd**

5. Data Flow Probe 시작

시작 > 모든 프로그램 > HP UCMDB > Data Flow Probe 시작

clearProbeData 스크립트: 사용

현재 비밀번호를 변경하지 않고 데이터베이스 사용자를 다시 만들려면 Windows의 경우 **clearProbeData.bat** 스크립트, Linux의 경우 **clearProbeData.sh** 스크립트를 실행합니다.

스크립트를 실행한 후에는 다음을 수행합니다.

- 다음 파일에서 오류를 검토합니다.
  C:\hp\UCMDB\DataFlowProbe\runtime\log\probe_setup.log(Windows의 경우),
  /opt/hp/UCMDB/DataFlowProbe/runtime/log/probe_setup.log(Linux의 경우)
- 데이터베이스 비밀번호가 포함되어 있으므로 해당 파일을 삭제합니다.

참고: HP Software 지원에서 요청한 경우가 아니라면 이 스크립트를 실행하지 마십시오.

JMX 콘솔의 암호화된 비밀번호 설정

이 섹션에서는 JMX 사용자의 비밀번호를 암호화하는 방법을 설명합니다. 암호화된 비밀번호는 DataFlowProbe.properties 파일에 저장됩니다. 사용자는 로그인해야 JMX 콘솔에 액세스할 수 있습니다.

1. 암호화된 형식의 비밀번호 만들기(AES, 192비트 키)
   a. Data Flow Probe JMX 콘솔에 액세스합니다. 웹 브라우저를 시작하고 주소창에 **http://<Data Flow Probe 컴퓨터 이름 또는 IP 주소>:1977**을 입력합니다. Data Flow Probe를 로컬로 실행하는 경우에는 **http://localhost:1977**을 입력합니다.

   사용자 이름과 비밀번호를 입력하여 로그인해야 할 수도 있습니다.

   참고: 사용자를 만들지 않은 경우 기본 사용자 이름 sysadmin 및 비밀번호 sysadmin을 사용하여 로그인합니다.

   b. **Type=MainProbe** 서비스를 찾은 다음 링크를 클릭하여 작업 페이지를 엽니다.
   c. **getEncryptedKeyPassword** 작업을 찾습니다.
   d. **Key Password** 필드에 암호화할 비밀번호를 입력합니다.
   e. **getEncryptedKeyPassword** 버튼을 클릭하여 작업을 호출합니다.

   호출 결과로 다음과 같은 암호화된 비밀번호 문자열이 생성됩니다.

   85,-9,-61,11,105,-93,-81,118

2. Data Flow Probe 중지

   시작 > 모든 프로그램 > HP UCMDB > Data Flow Probe 중지

3. 암호화된 비밀번호 추가

암호화된 비밀번호를 **C:\hp\UCMDB\DataFlowProbe\conf\DataFlowProbe.properties** 파일의 다음 속성에 추가합니다.

appilog.agent.Probe.JMX.BasicAuth.Pwd

예를 들면 다음과 같습니다.

appilog.agent.Probe.JMX.BasicAuth.User=sysadmin
appilog.agent.Probe.JMX.BasicAuth.Pwd=12,-35,-37,82,-2,20,57,-40,38,80,-111,-99,-64,-5,35,-122

참고: 인증을 사용하지 않도록 설정하려면 이러한 필드를 비워 둡니다. 그러면 사용자가 인증 정보를 입력하지 않고도 프로브의 JMX 콘솔 주 페이지를 열 수 있습니다.

4. Data Flow Probe 시작

   시작 > 모든 프로그램 > HP UCMDB > Data Flow Probe 시작

   웹 브라우저에서 결과를 테스트합니다.

UpLoadScanFile 비밀번호 설정

이 섹션에서는 오프사이트 스캔 저장에 사용되는 **UpLoadScanFile**의 비밀번호 설정 방법을 설명합니다. 암호화된 비밀번호는 **DataFlowProbe.properties** 파일에 저장됩니다. 사용자는 로그인해야 JMX 콘솔에 액세스할 수 있습니다.

1. **암호화된 형식의 비밀번호 만들기(AES, 192비트 키)**
   a. Data Flow Probe JMX 콘솔에 액세스합니다. 웹 브라우저를 시작하고 주소창에 **http://<Data Flow Probe 컴퓨터 이름 또는 IP 주소>:1977**을 입력합니다. Data Flow Probe를 로컬로 실행하는 경우에는 **http://localhost:1977**을 입력합니다.

      사용자 이름과 비밀번호를 입력하여 로그인해야 할 수도 있습니다.

      참고: 사용자를 만들지 않은 경우 기본 사용자 이름 sysadmin 및 비밀번호 sysadmin을 사용하여 로그인합니다.

   b. **Type=MainProbe** 서비스를 찾은 다음 링크를 클릭하여 작업 페이지를 엽니다.
   c. **getEncryptedKeyPassword** 작업을 찾습니다.
   d. **Key Password** 필드에 암호화할 비밀번호를 입력합니다.
   e. **getEncryptedKeyPassword** 버튼을 클릭하여 작업을 호출합니다.

      호출 결과로 다음과 같은 암호화된 비밀번호 문자열이 생성됩니다.

      85,-9,-61,11,105,-93,-81,118
2. **Data Flow Probe 중지**

   시작 > 모든 프로그램 > HP UCMDB > Data Flow Probe 중지
3. **암호화된 비밀번호 추가**

암호화된 비밀번호를 **C:\hp\UCMDB\DataFlowProbe\conf\DataFlowProbe.properties** 파일의 다음 속성에 추가합니다.

com.hp.ucmdb.discovery.Probe.JMX.UploadAuth.Pwd

예를 들면 다음과 같습니다.

com.hp.ucmdb.discovery.Probe.JMX.UploadAuth.User=UploadScanFile
com.hp.ucmdb.discovery.Probe.JMX.UploadAuth.Pwd=116,116,21,34,-59,77,-108,14,127,4,-89,101,-33,-31,116,53

4. **Data Flow Probe 시작**

   시작 > 모든 프로그램 > HP UCMDB > Data Flow Probe 시작

   웹 브라우저에서 결과를 테스트합니다.

PostgreSQL Server에 원격 액세스

이 섹션에서는 원격 컴퓨터에서의 PostgreSQL Data Flow Probe 계정 액세스를 허용/제한하는 방법을 설명합니다.

참고:

- 기본적으로는 액세스가 제한됩니다.
- 원격 컴퓨터에서는 PostgreSQL 루트 계정에 액세스할 수 없습니다.

PostgreSQL 액세스를 허용하려면 다음을 수행합니다.

- 명령 프롬프트 창에서 다음 스크립트를 실행합니다.

  C:\hp\UCMDB\DataFlowProbe\tools\dbscripts\enable_remote_user_access.cmd

PostgreSQL 액세스를 제한하려면 다음을 수행합니다.

- 명령 프롬프트 창에서 다음 스크립트를 실행합니다.

  C:\hp\UCMDB\DataFlowProbe\tools\dbscripts\remove_remote_user_access.cmd

UCMDB 서버와 Data Flow Probe 간에 SSL을 사용하도록 설정

Data Flow Probe와 UCMDB 서버 둘 모두 인증서를 사용하여 인증을 설정할 수 있습니다. 연결이 설정되기 전에 각 구성 요소에 대한 인증서를 보내서 인증을 합니다.

참고: Data Flow Probe에서 SSL을 사용하도록 설정하는 다음 방법이 가장 안전한 방법이자 권장 통신 모드입니다. 기본 인증 절차 대신 이 방법을 사용할 수 있습니다.

이 섹션에는 다음 항목이 포함됩니다.

개요

UCMDB에서는 UCMDB 서버와 Data Flow Probe 간에 다음 통신 모드를 지원합니다.

- **서버 인증.** 이 모드에서는 SSL을 사용하며, 프로브가 UCMDB 서버 인증서를 인증합니다. 자세한 내용은 "서버(단방향) 인증을 통해 SSL 사용"(67페이지)을 참조하십시오.
- **상호 인증.** 이 모드에서는 SSL을 사용하며, 프로브를 통한 서버 인증과 서버를 통한 클라이언트 인증이 모두 가능합니다. 자세한 내용은 "상호(양방향) 인증서 인증 사용"(70페이지)을 참조하십시오.
- **표준 HTTP.** SSL 통신이 사용되지 않습니다. 기본 모드이며, UCMDB의 Data Flow Probe 구성 요소에 인증서가 필요하지 않습니다. Data Flow Probe는 표준 HTTP 프로토콜을 통해 서버와 통신합니다.

참고: SSL 작업을 사용하는 경우에는 디스커버리에 인증서 체인을 사용할 수 없습니다. 따라서 인증서 체인을 사용하는 경우 Data Flow Probe가 UCMDB 서버와 통신하게 만들려면 자체 서명된 인증서를 생성해야 합니다.

키 저장소 및 신뢰 저장소

UCMDB 서버 및 Data Flow Probe에서는 키 저장소와 신뢰 저장소를 사용합니다.

- **키 저장소.** 키 항목(인증서와 그에 일치하는 개인 키)이 포함된 파일입니다.
- **신뢰 저장소.** 원격 호스트를 확인하는 데 사용되는 인증서가 포함된 파일입니다. 예를 들어 서버 인증을 사용하는 경우 Data Flow Probe의 신뢰 저장소에는 UCMDB 서버 인증서가 포함되어 있어야 합니다.

상호 인증 제한

C:\HP\UCMDB\DataFlowProbe\conf\security\ssl.properties에 정의된 Data Flow Probe 키 저장소에는 키 항목이 하나만 포함되어야 합니다.

서버(단방향) 인증을 통해 SSL 사용

SSL을 사용하며, 프로브가 서버의 인증서를 인증합니다.

이 작업에는 다음이 포함됩니다.

- "선행 조건"(68페이지)
- "UCMDB 서버 구성"(68페이지)

- "Data Flow Probe 구성"(69페이지)
- "컴퓨터 다시 시작"(69페이지)

선행 조건

1. UCMDB 및 Data Flow Probe가 모두 실행 중인지 확인합니다.

   참고: 프로브를 별도의 모드에서 설치하는 경우 이 지침은 프로브 게이트웨이에 적용됩니다.

2. UCMDB 또는 Data Flow Probe가 기본 폴더에 설치되어 있지 않은 경우에는 올바른 위치를 확인하고 위치에 맞게 명령을 변경합니다.

UCMDB 서버 구성

1. **UCMDB 인증서 내보내기**
   a. 명령 프롬프트를 열고 명령을 실행합니다.

      C:\HP\UCMDB\UCMDBServer\bin\jre\bin\keytool.exe -export -alias <키 저장소 별칭> -keystore <키 저장소 파일 경로> -file C:\HP\UCMDB\UCMDBServer\conf\security\server.cert

      여기서 각 항목은 다음과 같습니다.
      - **키 저장소 별칭**은 키 저장소에 지정한 이름입니다.
      - **키 저장소 파일 경로**는 keystore 파일 위치의 전체 경로입니다.

      예를 들어 기본 server.keystore의 경우 다음 명령을 사용합니다.

      C:\HP\UCMDB\UCMDBServer\bin\jre\bin\keytool.exe -export -alias hpcert -keystore C:\hpmdbmdbserver\conf\security\server.keystore -file C:\HP\UCMDB\UCMDBServer\conf\security\server.cert

   b. 키 저장소 비밀번호를 입력합니다. 예를 들어 기본 키 저장소 비밀번호는 **hppass**입니다.
   c. 다음 디렉터리에 인증서가 만들어졌는지 확인합니다. **C:\HP\UCMDB\UCMDBServer\conf\security\server.cert**
2. **UCMDB에서 Data Flow Probe 커넥터 강화**
   a. UCMDB JMX 콘솔에 액세스: 웹 브라우저에 다음 URL을 입력합니다. **http://<ucmdb machine name or IP address>:8080/jmx-console** 사용자 이름과 비밀번호를 입력하여 로그인해야 할 수도 있습니다.
   b. 서비스 선택: **Ports Management Services**
   c. **PortsDetails** 메서드를 호출하고 HTTPS 포트 번호를 확인합니다. (기본값: 8443) **Is Enabled** 열의 값이 **True**인지 확인합니다.
   d. **Ports Management Services**로 돌아갑니다.
   e. Data Flow Probe 커넥터를 서버 인증 모드에 매핑하려면 다음 매개 변수를 사용하여 **mapComponentToConnectors** 메서드를 호출합니다.
      - **componentName**: mam-collectors
      - **isHTTPS**: true

- **기타 모든 플래그**: false

다음 메시지가 표시됩니다.

Operation succeeded. Component mam-collectors is now mapped to: HTTPS ports.

f. **Ports Management Services**로 돌아갑니다.

g. Confidential Manager 커넥터를 서버 인증 모드에 매핑하려면 다음 매개 변수를 사용하여 **mapComponentToConnectors** 메서드를 호출합니다.

- **componentName**: cm
- **isHTTPS**: true
- **기타 모든 플래그**: false

다음 메시지가 표시됩니다.

Operation succeeded. Component cm is now mapped to: HTTPS ports.

3. **UCMDB 인증서를 각각의 프로브 컴퓨터에 복사**

UCMDB 서버 컴퓨터에 있는 인증서 파일 **C:\HP\UCMDB\UCMDBServer\conf\security\server.cert**를 각 Data Flow Probe 컴퓨터의 다음 폴더로 복사합니다.
C:\HP\UCMDB\DataFlowProbe\conf\security

Data Flow Probe 구성

참고: 각 Data Flow Probe 컴퓨터를 구성해야 합니다.

1. "UCMDB 인증서 내보내기"(68페이지)**에 만들어진 server.cert 파일을 프로브의 신뢰 저장소로 가져옵니다.**

a. 명령 프롬프트를 열고 명령을 실행합니다.

```
C:\HP\UCMDB\DataFlowProbe\bin\jre\bin\keytool.exe -import -v -keystore
C:\HP\UCMDB\DataFlowProbe\conf\security\hpprobeTrustStore.jks -file
C:\HP\UCMDB\DataFlowProbe\conf\security\server.cert -alias hpcert
```

b. 키 저장소 비밀번호(logomania)를 입력합니다.

c. **Trust this certificate?**라는 메시지가 표시되면 **y**를 누르고 **Enter** 키를 누릅니다.

다음 메시지가 표시됩니다.

Certificate was added to keystore.

2. **C:\HP\UCMDB\DataFlowProbe\conf\에 있는 DataFlowProbe.properties 파일을 엽니다.**

a. **appilog.agent.probe.protocol** 속성을 **HTTPS**로 업데이트합니다.

b. **serverPortHttps** 속성을 관련 포트 번호로 업데이트합니다. ("UCMDB 서버 구성"(68페이지)의 단계 2c에 사용된 포트 번호를 사용합니다.)

컴퓨터 다시 시작

UCMDB 서버와 프로브 컴퓨터를 모두 다시 시작합니다.

상호(양방향) 인증서 인증 사용

이 모드에서는 SSL을 사용하며, 프로브를 통한 서버 인증과 서버를 통한 클라이언트 인증이 모두 가능합니다. 서버와 프로브는 모두 인증을 위해 다른 엔터티로 인증서를 보냅니다.

참고: 인증서 체인을 통해 상호 인증서 인증을 사용하도록 설정할 수 있습니다. 인증서 체인을 생성하는 방법에 대한 자세한 내용은 "(선택 사항) UCMDB 인증서 체인 생성"(74페이지)을 참조하십시오.

이 작업에는 다음이 포함됩니다.

선행 조건

1. UCMDB 및 Data Flow Probe가 모두 실행 중인지 확인합니다.

   참고: 프로브를 별도의 모드에서 설치하는 경우 이 지침은 프로브 게이트웨이에 적용됩니다.

2. UCMDB 또는 Data Flow Probe가 기본 폴더에 설치되어 있지 않은 경우에는 올바른 위치를 확인하고 위치에 맞게 명령을 변경합니다.

초기 UCMDB 서버 구성

1. **UCMDB 인증서 내보내기**
   a. 명령 프롬프트를 열고 명령을 실행합니다.

      C:\HP\UCMDB\UCMDBServer\bin\jre\bin\keytool.exe -export -alias <키 저장소 별칭> -keystore <키 저장소 파일 경로> -file C:\HP\UCMDB\UCMDBServer\conf\security\server.cert

      여기서 각 항목은 다음과 같습니다.
      - **키 저장소 별칭**은 키 저장소에 지정한 이름입니다.
      - **키 저장소 파일 경로**는 keystore 파일 위치의 전체 경로입니다.

      예를 들어 기본 server.keystore의 경우 다음 명령을 사용합니다.

      C:\HP\UCMDB\UCMDBServer\bin\jre\bin\keytool.exe -export -alias hpcert -keystore C:\hpmdbmdbserver\conf\security\server.keystore -file C:\HP\UCMDB\UCMDBServer\conf\security\server.cert

   b. 키 저장소 비밀번호를 입력합니다. 예를 들어 기본 키 저장소 비밀번호는 **hppass**입니다.

c. 다음 디렉터리에 인증서가 만들어졌는지 확인합니다.
C:\HP\UCMDB\UCMDBServer\conf\security\server.cert

2. **UCMDB에서 Data Flow Probe 커넥터 강화**

a. UCMDB JMX 콘솔에 액세스: 웹 브라우저에 다음 URL을 입력합니다. **http://<ucmdb machine name or IP address>:8080/jmx-console** 사용자 이름과 비밀번호를 입력하여 로그인해야 할 수도 있습니다.

b. 서비스 선택: **Ports Management Services**

c. **PortsDetails** 메서드를 호출하고 클라이언트 인증에 사용되는 HTTPS 포트 번호를 확인합니다. (기본값: 8444) **Is Enabled** 열의 값이 **True**인지 확인합니다.

d. **Ports Management Services**로 돌아갑니다.

e. Data Flow Probe 커넥터를 상호 인증 모드에 매핑하려면 다음 매개 변수를 사용하여 **mapComponentToConnectors** 메서드를 호출합니다.

- **componentName**: mam-collectors
- **isHTTPSWithClientAuth**: true
- **기타 모든 플래그**: false

다음 메시지가 표시됩니다.

Operation succeeded. Component mam-collectors is now mapped to: HTTPS_CLIENT_AUTH ports.

f. **Ports Management Services**로 돌아갑니다.

g. Confidential Manager 커넥터를 상호 인증 모드에 매핑하려면 다음 매개 변수를 사용하여 **mapComponentToConnectors** 메서드를 호출합니다.

- **componentName**: cm
- **isHTTPSWithClientAuth**: true
- **기타 모든 플래그**: false

다음 메시지가 표시됩니다.

Operation succeeded. Component cm is now mapped to: HTTPS_CLIENT_AUTH ports.

3. **UCMDB 인증서를 각각의 프로브 컴퓨터에 복사**

UCMDB 서버 컴퓨터에 있는 인증서 파일 **C:\HP\UCMDB\UCMDBServer\conf\security\server.cert**를 각 Data Flow Probe 컴퓨터의 다음 폴더로 복사합니다.
C:\HP\UCMDB\DataFlowProbe\conf\security

Data Flow Probe 구성

참고: 각 Data Flow Probe 컴퓨터를 구성해야 합니다.

1. "UCMDB 인증서 내보내기"(70페이지)**에 만들어진 server.cert 파일을 프로브의 신뢰 저장소로 가져옵니다.**

a. 명령 프롬프트를 열고 명령을 실행합니다.

C:\HP\UCMDB\DataFlowProbe\bin\jre\bin\keytool.exe -import -v -keystore C:\HP\UCMDB\DataFlowProbe\conf\security\hpprobeTrustStore.jks -file C:\HP\UCMDB\DataFlowProbe\conf\security\server.cert -alias hpcert

b. 키 저장소 비밀번호(logomania)를 입력합니다.

c. **Trust this certificate?**라는 메시지가 표시되면 **y**를 누르고 **Enter** 키를 누릅니다.

다음 메시지가 표시됩니다.

Certificate was added to keystore.

2. **새 client.keystore 파일 만들기**

a. 명령 프롬프트를 열고 명령을 실행합니다.

c:\HP\UCMDB\DataFlowProbe\bin\jre\keytool -genkey -alias <ProbeName> -keyalg RSA -sigalg SHA256withRSA -keysize 2048 -keystore c:\HP\UCMDB\DataFlowProbe\conf\security\client.keystore

여기서 **ProbeName**은 Data Flow Probe의 고유한 별칭입니다.

참고: 이 별칭을 고유하게 만들려면 프로브를 정의할 때 프로브에 지정한 프로브 이름 식별자를 사용합니다.

b. 키 저장소 비밀번호를 6자 이상으로 입력하고 적어둡니다.

c. 비밀번호를 다시 입력하여 확인합니다.

d. 다음 각 질문에 답한 후 **Enter** 키를 누릅니다.

What is your first and last name?(이름과 성은 무엇입니까?) [Unknown]:

What is the name of your organizational unit?(조직 단위의 이름은 무엇입니까?) [Unknown]:

What is the name of your organization?(조직의 이름은 무엇입니까?)[Unknown]:

What is the name of your City or Locality?(구/군/시 이름은 무엇입니까?) [Unknown]:

What is the name of your State or Province?(시/도 이름은 무엇입니까?)[Unknown]:

What is the two-letter country code for this unit?(이 단위의 2글자 국가 코드는 무엇입니까?)[Unknown]:

e. **Is CN=Unknown, OU=Unknown, O=Unknown, L=Unknown, ST=Unknown, C=Unknown correct?**라는 질문이 표시되면 **yes**를 입력합니다.

f. 다음 질문에 답한 후 **Enter** 키를 누릅니다.

<probekey>의 키 비밀번호 입력(키 저장소 비밀번호와 같으면 RETURN):

g. 다음 폴더에 파일이 만들어졌는지 확인하고 파일 크기가 0보다 큰지 확인합니다.
C:\hp\UCMDB\DataFlowProbe\conf\security\client.keystore

3. **새 클라이언트 인증서 내보내기**

a. 명령 프롬프트를 열고 명령을 실행합니다.

C:\HP\UCMDB\DataFlowProbe\bin\jre\bin\keytool.exe -export -alias <ProbeName> -keystore C:\hp\UCMDB\DataFlowProbe\conf\security\client.keystore -file C:\hp\UCMDB\DataFlowProbe\conf\security\<ProbeName>.cert

b. 질문이 표시되면 키 저장소 비밀번호를 입력합니다 (위 단계 2b에서의 비밀번호).

다음 메시지가 표시됩니다.

Certificate stored in file <C:\hp\UCMDB\DataFlowProbe\conf\security\<ProbeName>.cert>

4. **C:\HP\UCMDB\DataFlowProbe\conf\에 있는 DataFlowProbe.properties 파일을 엽니다.**
   a. **appilog.agent.probe.protocol** 속성을 **HTTPS**로 업데이트합니다.
   b. **serverPortHttps** 속성을 관련 포트 번호로 업데이트합니다. ("초기 UCMDB 서버 구성"(70페이지)의 단계 2c에 사용된 포트 번호를 사용합니다.)
5. **다음 위치에서 ssl.properties 파일을 엽니다. C:\HP\UCMDB\DataFlowProbe\conf\security**
   a. **javax.net.ssl.keyStore** 속성을 **client.keystore**로 업데이트합니다.
   b. 위 단계 2b에서의 비밀번호 암호화:
      i. Data Flow Probe를 시작하거나, 이미 실행 중인지 확인합니다.
      ii. 프로브 JMX에 액세스합니다. **http://<probe_hostname>:1977**로 이동합니다.

      예를 들어 프로브를 로컬로 실행하는 경우 다음 위치로 이동합니다. **http://localhost:1977**
      iii. **type=MainProbe** 링크를 누릅니다.
      iv. 아래로 스크롤하여 **getEncryptedKeyPassword** 작업으로 이동합니다.
      v. **Key Password** 필드에 비밀번호를 입력합니다.
      vi. **getEncryptedKeyPassword** 버튼을 누릅니다.
   c. 암호화된 비밀번호를 복사한 후 붙여넣어 **javax.net.ssl.keyStorePassword** 속성을 업데이트합니다.

> **참고:** 번호는 쉼표로 구분됩니다. 예: -20,50,34,-40,-50

6. **프로브 인증서를 UCMDB 컴퓨터에 복사**

   C:\HP\UCMDB\DataFlowProbe\conf\security\client.cert 파일을 Data Flow Probe 컴퓨터에서 UCMDB 컴퓨터의 **C:\HP\UCMDB\UCMDBServer\conf\security\<ProbeName>.cert**로 복사합니다.

UCMDB 서버 추가 구성

1. **각 프로브 인증서를 UCMDB의 신뢰 저장소에 추가**

> **참고:** 각 프로브 인증서에 대해 다음 단계를 완료해야 합니다.

   a. 명령 프롬프트를 열고 명령을 실행합니다.

C:\HP\UCMDB\UCMDBServer\bin\jre\bin\keytool.exe -import -v -keystore C:\hp\UCMDB\UCMDBServer\conf\security\server.truststore -file C:\hp\UCMDB\UCMDBServer\conf\security\<ProbeName>.cert -alias <ProbeName>

b. 키 저장소 비밀번호를 입력합니다. 예를 들어 기본 키 저장소 비밀번호는 **hppass**입니다.

c. **Trust this certificate?**라는 메시지가 표시되면 **y**를 누르고 **Enter** 키를 누릅니다.

다음 메시지가 표시됩니다.

Certificate was added to keystore

컴퓨터 다시 시작

UCMDB 서버와 프로브 컴퓨터를 모두 다시 시작합니다.

(선택 사항) UCMDB 인증서 체인 생성

1. **키 저장소 생성**

다음 절차를 시작하기 전에 **C:\hp\UCMDB\UCMDBServer\conf\security\server.keystore** 디렉터리에 있는 이전 **server.keystore**를 제거합니다.

a. 명령 프롬프트를 열고 명령을 실행합니다.

C:\HP\UCMDB\UCMDBServer\bin\jre\bin\keytool -genkey -alias <키 저장소 별칭> -keyalg RSA -sigalg SHA256withRSA -keysize 2048 -keystore <키 저장소 파일 경로>

여기서 각 항목은 다음과 같습니다.

- **키 저장소 별칭**은 키 저장소에 지정한 이름입니다.
- **키 저장소 파일 경로**는 keystore 파일 위치의 전체 경로입니다.

b. 키 저장소 비밀번호를 6자 이상으로 입력하고 적어둡니다.

키 저장소 비밀번호를 입력합니다.

- 비밀번호가 변경된 경우에는 **UCMDB:service=Security Services**에서 **changeKeystorePassword** JMX 작업을 실행합니다.
- 비밀번호가 변경되지 않은 경우에는 기본 **hppass** 비밀번호를 사용합니다.

c. 비밀번호를 다시 입력하여 확인합니다.

d. 다음 각 질문에 답한 후 **Enter** 키를 누릅니다.

- **이름과 성은 무엇입니까?**

[알 수 없음]: [일반 이름(CN)]

CN을 올바르게 입력해야 합니다. CN은 FQDN(정규화된 도메인 이름)이어야 합니다. "sitename" 같은 단일 짧은 이름이나 IP 주소는 사용할 수 없습니다.

유효한 FQDN의 예는 다음과 같습니다.

www.sitename.com

sitename.com

sitename.hp.com

sitename.eds.com

- **조직 단위의 이름은 무엇입니까?**

  [알 수 없음]: [조직 단위(OU)]

  참고: 이 필드는 어떠한 형식으로도 회사 이름을 참조하면 안 됩니다(예: HP, Hewlett-Packard, Google 등). CSR 때문에 이 필드를 비워 둘 수 없으면(비워두는 것이 좋음) 일종의 부서를 참조해야 합니다(예: 온라인, 회계, 재무 등). 이 필드를 잘못 입력하면 등록에 실패할 수 있습니다.

- **조직의 이름은 무엇입니까?**

  [알 수 없음]: [조직(O)]

  조직 이름을 입력합니다(예: **Hewlett-Packard**).

- **구/군/시 이름은 무엇입니까?**

  [알 수 없음]: [구/군/시(L)]

  SSL 인증서가 있는 서버의 구/군/시를 입력합니다. 이 필드는 비워 둘 수 없습니다.

- **시/도 이름은 무엇입니까?**

  [알 수 없음]: [시/도(S)]

  SSL 인증서가 있는 서버의 시/도를 입력합니다. 시/도는 전체 이름(3자 이상)을 입력해야 하고 약어를 입력할 수 없습니다(예: **CO**가 아니라 **Colorado** 입력). 이 필드는 비워 둘 수 없습니다.

- **이 단위의 2글자 국가 코드는 무엇입니까?**

  [알 수 없음]: [국가(C)]

  SSL 인증서가 있는 서버의 국가를 입력합니다. 2글자 ISO 3166 국가 코드를 입력해야 합니다. 이 필드는 비워 둘 수 없습니다.

e. **CN=[XXX], OU=[XXXX], O=[XXXX], L=[XXXX], ST=[XXXX], C=[XXX]이(가) 정확합니까?**라는 질문이 표시되면 **y**를 입력합니다.

f. 다음 질문에 답한 후 **Enter** 키를 누릅니다.

<serverkey>의 키 비밀번호를 입력합니다(키 저장소 비밀번호와 같으면 RETURN).

새 비밀번호를 다시 입력합니다.

g. 다음 폴더에 파일이 만들어졌는지 확인하고 파일 크기가 0보다 큰지 확인합니다. **C:\hp\UCMDB\UCMDBServer\conf\security\server.keystore**.

2. **CSR을 생성합니다.**

   다음 명령을 실행하여 CSR을 생성합니다.

```
c:\HP\UCMDB\UCMDBServer\bin\jre\bin\keytool -certreq -alias server -file
c:\HP\UCMDB\UCMDBServer\conf\security\certreq.csr -keystore
c:\HP\UCMDB\UCMDBServer\conf\security\server.keystore -sigalg SHA256withRSA
```

3. **서버 개인 인증을 받습니다.**
   a. 먼저 CA 루트 인증서를 다운로드하고 신뢰할 수 있는 루트 인증 기관으로 설치합니다.
   b. 접미사를 **.cer** 또는 **.crt**로 수정합니다.
   c. 인증 파일을 다음 디렉터리에 넣습니다.
   C:\HP\UCMDB\UCMDBServer\conf\security\serverserver.cer.
4. **인증서 체인을 생성합니다.**
   a. 다음 명령을 사용하여 루트 인증서를 키 저장소로 가져옵니다.

   ```
   c:\HP\UCMDB\UCMDBServer\bin\jre\bin\keytool -import -v -trustcacerts -alias root -keystore
   c:\HP\UCMDB\UCMDBServer\conf\security\server.keystore  file
   c:\HP\UCMDB\UCMDBServer\conf\security\server.cer
   ```

   b. 다음 명령을 사용하여 서버 인증서를 키 저장소로 가져옵니다.

   ```
   c:\HP\UCMDB\UCMDBServer\bin\jre\bin\keytool -import -v -trustcacerts -alias server -keystore
   c:\HP\UCMDB\UCMDBServer\conf\security\server.keystore -file
   c:\HP\UCMDB\UCMDBServer\conf\security\server.cer
   ```

   참고: 별칭 이름은 키 저장소를 생성할 때 사용한 별칭 이름과 같아야 하고 가져오기 순서는 변경할 수 없습니다.

   c. 인증서 체인이 생성됩니다.
   다음 명령을 사용하여 키 저장소의 세부 정보를 확인합니다.

   ```
   C:\HP\UCMDB\UCMDBServer\bin\jre\bin\keytool.exe -list -v -keystore
   C:\hp\UCMDB\UCMDBServer\conf\security\server.keystore
   ```

참고: 프로브 인증서 체인을 생성하려면 위 단계를 반복합니다. 유일한 차이점은 별칭 이름을 "client"로 지정하고 **client.keystore** 및 **client.cer** 파일을 생성한다는 것입니다.

에이전트 또는 스캐너에 대한 aioptionrc 파일 권한을 변경하는 방법

/.discagent/aioptionrc 파일은 모든 사용자가 쓸 수 있고 기본 권한은 666으로 설정됩니다. 사용자가 **-home** 옵션(에이전트 설치 및 스캐너에 대해)을 사용하여 **aioptionrc** 파일 경로를 특정 고정 디렉터리로 설정하면 이 파일의 기본 권한으로 여러 사용자가 자동으로 인벤토리 작업을 실행(**sudo** 구성됨)하거나 스캐너를 수동으로 실행할 수 있습니다.

에이전트 또는 스캐너에 대한 **aioptionrc** 파일 권한을 변경하려면 다음을 수행합니다.

1. 스캔 후 스크립트 편집기를 엽니다.
   스캔 후 스크립트 편집기에 액세스하는 방법에 대한 자세한 내용은 *HP Universal CMDB 데이터 흐름 관리 안내서*에서 스캔 전/후 스크립트 편집기를 참조하십시오.

2. **chmod o-w ./aioptionrc** 명령을 스캐너의 UNIX 운영 체제용 스캔 후 스크립트에 추가합니다.

   스캔 후 스크립트를 편집하는 방법에 대한 자세한 내용은 *HP Universal CMDB 데이터 흐름 관리 안내서*에서 스캔 전/후 스크립트를 편집하는 방법을 참조하십시오.

3. 결과

   파일 권한이 변경되고 에이전트를 설치하거나 스캐너를 실행하는 첫 번째 사용자가 유일한 사용자가 됩니다.

   참고: 같은 그룹의 소유자와 사용자는 해당 파일에 대해 같은 쓰기 권한을 가집니다.

Data Flow Probe용 키 저장소 만들기

1. 프로브 컴퓨터에서 다음 명령을 실행합니다.

   c:\HP\UCMDB\DataFlowProbe\bin\jre\keytool -genkey -alias <ProbeName> -keyalg RSA -sigalg SHA256withRSA -keysize 2048 -keystore c:\HP\UCMDB\DataFlowProbe\conf\security\client.keystore

2. 새 키 저장소의 비밀번호를 입력합니다.
3. 정보를 입력하라는 메시지가 표시되면 정보를 입력합니다.
4. **CN=....C=...이(가) 맞습니까?**라는 메시지가 표시되면 **예**를 입력하고 **Enter** 키를 누릅니다.
5. **Enter** 키를 다시 눌러 키 저장소 비밀번호를 키 비밀번호로 사용하도록 합니다.
6. **client.keystore**가 **C:\HP\UCMDB\DataFlowProbe\conf\security** 디렉터리에 생성되었는지 확인합니다.

프로브 키 저장소 및 신뢰 저장소 비밀번호 암호화

프로브 키 저장소 및 신뢰 저장소 비밀번호는 암호화되어 **C:\HP\UCMDB\DataFlowProbe\conf\security\ssl.properties**에 저장됩니다. 이 절차에서는 비밀번호를 암호화하는 방법을 설명합니다.

1. Data Flow Probe를 시작하거나, 이미 실행 중인지 확인합니다.
2. Data Flow Probe JMX 콘솔에서 웹 브라우저를 시작하고 주소창에 http://<Data Flow Probe 컴퓨터 이름 또는 IP 주소>:1977을 입력합니다. Data Flow Probe를 로컬로 실행하는 경우에는 http://localhost:1977을 입력합니다.

   참고: 사용자 이름과 비밀번호를 입력하여 로그인해야 할 수도 있습니다. 사용자를 만들지 않은 경우 기본 사용자 이름 sysadmin 및 비밀번호 sysadmin을 사용하여 로그인합니다.

3. **Type=MainProbe** 서비스를 찾은 다음 링크를 클릭하여 작업 페이지를 엽니다.

4. **getEncryptedKeyPassword** 작업을 찾습니다.
5. **키 비밀번호** 필드에 키 저장소 또는 신뢰 저장소의 비밀번호를 입력하고 **getEncryptedKeyPassword**를 클릭하여 작업을 호출합니다.
6. 호출 결과로 다음과 같은 암호화된 비밀번호 문자열이 생성됩니다.

   66,85,54,78,69,117,56,65,99,90,86,117,97,75,50,112,65,53,67,114,112,65,61,61

7. 암호화된 비밀번호를 복사하여 **C:\HP\UCMDB\DataFlowProbe\conf\security\ssl.properties** 파일의 키 저장소 또는 신뢰 저장소의 해당 줄에 붙여 넣습니다.

서버 및 Data Flow Probe 기본 키 저장소와 신뢰 저장소

이 섹션에는 다음 항목이 포함됩니다.

- "UCMDB 서버"(78페이지)
- "Data Flow Probe"(78페이지)

UCMDB 서버

파일은 **C:\HP\UCMDB\UCMDBServer\conf\security** 디렉터리에 있습니다.

엔터티	파일 이름/용어	비밀번호/용어	별칭
서버 키 저장소	server.keystore (sKeyStoreFile)	hppass(sKeyStorePass)	hpcert
서버 신뢰 저장소	server.truststore (sTrustStoreFile)	hppass (sTrustStorePass)	hpcert(기본 신뢰 항목)
클라이언트 키 저장소	client.keystore (cKeyStoreFile)	clientpass (cKeyStorePass)	clientcert

Data Flow Probe

파일은 **C:\HP\UCMDB\DataFlowProbe\conf\security** 디렉터리에 있습니다.

엔터티	파일 이름/용어	비밀번호/용어	별칭
프로브 키 저장소	hpprobeKeyStore.jks (pKeyStoreFile)	logomania (pKeyStorePass)	hpprobe

엔터티	파일 이름/용어	비밀번호/용어	별칭
Data Flow Probe는 상호 인증 절차 중에 **cKeyStoreFile** 키 저장소를 기본 키 저장소로 사용합니다. 이 키 저장소는 UCMDB 설치의 일부분인 클라이언트 키 저장소입니다.			
프로브 신뢰 저장소	hpprobeTrustStore.jks (pTrustStoreFile)	logomania (pTrustStorePass)	hpprobe(기본 신뢰 항목)
cKeyStorePass 비밀번호는 **cKeyStoreFile**의 기본 비밀번호입니다.			

에이전트 또는 스캐너에 대한 aioptionrc 파일 권한을 변경하는 방법

/.discagent/aioptionrc 파일은 모든 사용자가 쓸 수 있고 기본 권한은 666으로 설정됩니다. 사용자가 **-home** 옵션(에이전트 설치 및 스캐너에 대해)을 사용하여 **aioptionrc** 파일 경로를 특정 고정 디렉터리로 설정하면 이 파일의 기본 권한으로 여러 사용자가 자동으로 인벤토리 작업을 실행(**sudo** 구성됨)하거나 스캐너를 수동으로 실행할 수 있습니다.

에이전트 또는 스캐너에 대한 **aioptionrc** 파일 권한을 변경하려면 다음을 수행합니다.

1. 스캔 후 스크립트 편집기를 엽니다.

   스캔 후 스크립트 편집기에 액세스하는 방법에 대한 자세한 내용은 *HP Universal CMDB 데이터 흐름 관리 안내서*에서 스캔 전/후 스크립트 편집기를 참조하십시오.
2. **chmod o-w ./aioptionrc** 명령을 스캐너의 UNIX 운영 체제용 스캔 후 스크립트에 추가합니다.

   스캔 후 스크립트를 편집하는 방법에 대한 자세한 내용은 *HP Universal CMDB 데이터 흐름 관리 안내서*에서 스캔 전/후 스크립트를 편집하는 방법을 참조하십시오.
3. 결과

   파일 권한이 변경되고 에이전트를 설치하거나 스캐너를 실행하는 첫 번째 사용자가 유일한 사용자가 됩니다.

   참고: 같은 그룹의 소유자와 사용자는 해당 파일에 대해 같은 쓰기 권한을 가집니다.

6장: LW-SSO(Lightweight Single Sign-On) 인증

이 장의 내용:

LW-SSO 인증 개요

LW-SSO는 한 번 로그인한 사용자에게 다시 로그인하라는 메시지를 표시하지 않고 다중 소프트웨어 시스템의 리소스에 액세스할 수 있는 권한을 주는 액세스 제어 방법입니다. 구성된 소프트웨어 시스템 그룹 내의 응용 프로그램은 인증을 신뢰하므로 응용 프로그램 간에 이동할 때 추가로 인증할 필요가 없습니다.

이 섹션의 정보는 LW-SSO 버전 2.2와 2.3에 적용됩니다.

- **LW-SSO 토큰 만료**

  LW-SSO 토큰의 만료 값은 응용 프로그램의 세션 유효성을 결정합니다. 따라서 토큰의 만료 값은 최소한 응용 프로그램 세션 만료 값과 같아야 합니다.

- **LW-SSO 토큰 만료의 권장 구성**

  LW-SSO를 사용하는 각 응용 프로그램은 토큰 만료를 구성해야 합니다. 권장되는 값은 60분입니다. 높은 보안 수준이 필요하지 않은 응용 프로그램의 경우에는 이 값을 300분으로 구성해도 됩니다.

- **GMT 시간**

  LW-SSO 통합에 포함된 모든 어플리케이션은 최대 시간 차이가 15분인 동일한 GMT 시간을 사용해야 합니다.

- **멀티 도메인 기능**

  다중 도메인 기능을 사용하려면 LW-SSO 통합에 참가하는 모든 응용 프로그램(다른 DNS 도메인에 있는 응용 프로그램과 통합해야 하는 경우)에서 trustedHosts 설정 또는 **protectedDomains** 설정을 구성해야 합니다. 또한 구성의 **lwsso** 요소에 올바른 도메인을 추가해야 합니다.

- **URL용 보안 토큰 가져오기 기능**

다른 응용 프로그램에서 **URL의 SecurityToken**으로 보낸 정보를 받으려면 호스트 응용 프로그램에서 구성의 **lwsso** 요소에 올바른 도메인을 구성해야 합니다.

LW-SSO 시스템 요구 사항

Application	버전	설명
Java	1.5 이상	
HTTP 서블릿 API	2.1 이상	
Internet Explorer	6.0 이상	브라우저에서 HTTP 세션 쿠키 및 HTTP 302 리디렉션 기능을 사용해야 합니다.
Firefox	2.0 이상	브라우저에서 HTTP 세션 쿠키 및 HTTP 302 리디렉션 기능을 사용해야 합니다.
JBoss 인증	JBoss 4.0.3 JBoss 4.3.0	
Tomcat 인증	독립 실행형 Tomcat 5.0.28 독립 실행형 Tomcat 5.5.20	
Acegi 인증	Acegi 0.9.0 Acegi 1.0.4	
웹 서비스 엔진	Axis 1 - 1.4 Axis 2 - 1.2 JAX-WS-RI 2.1.1	

LW-SSO 보안 경고

이 섹션에서는 LW-SSO 구성과 관련된 다음과 같은 보안 경고에 대해 설명합니다.

- **LW-SSO의 기밀 InitString 매개 변수.**LW-SSO에서는 대칭형 암호화를 사용하여 LW-SSO 토큰의 유효성을 검사하고 해당 토큰을 생성합니다. 구성 내의 **initString** 매개 변수는 보안 키의 초기화에 사용

됩니다. 한 응용 프로그램에서 토큰을 만들면 같은 initString 매개 변수를 사용하는 각 응용 프로그램에서 토큰의 유효성을 검사합니다.

주의:

- **initString** 매개 변수를 설정하지 않으면 LW-SSO를 사용할 수 없습니다.
- **initString** 매개 변수는 기밀 정보이므로 게시, 전송 및 지속성 면에서 기밀 정보로 취급해야 합니다.
- **initString** 매개 변수는 LW-SSO를 사용하여 서로 통합된 응용 프로그램 간에서만 공유되어야 합니다.
- **initString** 매개 변수는 12자 이상이어야 합니다.

- **필요한 경우에만 LW-SSO 사용.**LW-SSO는 특별히 필요한 경우가 아니면 사용하지 않아야 합니다.
- **인증 보안 수준**. 가장 취약한 인증 프레임워크를 사용하고 다른 통합 응용 프로그램이 신뢰한 LW-SSO 토큰을 발급하는 응용 프로그램은 모든 응용 프로그램에 대해 인증 보안 수준을 확인합니다.

  강력하고 안전한 인증 프레임워크를 사용하는 응용 프로그램만 LW-SSO 토큰을 발급하는 것이 좋습니다.
- **대칭 암호화의 의미.**LW-SSO는 LW-SSO 토큰을 발급하고 유효성을 검사하는 데 대칭형 암호 기법을 사용합니다. 따라서 LW-SSO를 사용하는 응용 프로그램은 동일한 initString 매개 변수를 공유하는 다른 모든 응용 프로그램에서 신뢰할 토큰을 발급할 수 있습니다. initString을 공유하는 응용 프로그램이 신뢰할 수 없는 위치에 있거나 이러한 위치에서 액세스 가능한 경우에는 보안이 위험해질 수 있습니다.
- **사용자 매핑(동기화).**LW-SSO 프레임워크는 통합 응용 프로그램 간의 사용자 매핑을 보장하지 않습니다. 따라서 통합 응용 프로그램에서 사용자 매핑을 모니터링해야 합니다. 모든 통합 응용 프로그램이 동일한 사용자 레지스트리(예: LDAP/AD)를 공유하는 것이 좋습니다.

  사용자를 매핑하지 않으면 보안 위반이 발생할 수 있고 응용 프로그램 동작에 부정적인 영향을 줄 수 있습니다. 예를 들어 같은 사용자 이름이 여러 응용 프로그램에서 실제로는 다른 사용자에게 할당될 수 있습니다.

  또한, 사용자가 한 응용 프로그램(AppA)에 로그온한 후에 컨테이너 또는 응용 프로그램 인증을 사용하는 두 번째 응용 프로그램(AppB)에 액세스하는 경우 사용자를 매핑하지 않으면 해당 사용자가 AppB에 수동으로 로그온하여 사용자 이름을 입력해야 합니다. 사용자가 AppA에 로그온할 때 사용된 것과 다른 사용자 이름을 입력하면 다음 동작이 발생할 수 있습니다. 사용자가 이후에 AppA 또는 AppB에서 세 번째 응용 프로그램(AppC)에 액세스하는 경우, 각각 AppA 또는 AppB에 로그온할 때 사용했던 사용자 이름을 사용하여 액세스하게 됩니다.
- **ID 관리자**. 인증 목적으로 사용하려면 Identity Manager의 보호되지 않는 모든 리소스는 LW-SSO 구성 파일의 **nonsecureURLs** 설정을 사용하여 구성되어야 합니다.
- **LW-SSO 데모 모드**
  - 데모 모드는 데모용으로만 사용해야 합니다.
  - 데모 모드는 비보안 네트워크에서만 사용해야 합니다.

- 프로덕션 시에는 데모 모드를 사용하지 않아야 합니다. 데모 모드와 프로덕션 모드를 함께 사용하면 안 됩니다.

문제 해결 및 제한 사항

이 섹션에서는 LW-SSO 인증 작업과 관련된 알려진 문제와 제한에 대해 설명합니다.

알려진 문제

이 섹션에서는 LW-SSO 인증의 알려진 문제에 대해 설명합니다.

- **보안 컨텍스트.**LW-SSO 보안 컨텍스트는 특성 이름별로 하나의 특성 값만 지원합니다.

  따라서 SAML2 토큰이 동일한 특성 이름에 대해 둘 이상의 값을 보내면 LW-SSO 프레임워크에서는 하나의 값만 허용됩니다.

  이와 유사하게, IdM 토큰이 동일한 특성 이름에 대해 둘 이상의 값을 보내도록 구성되어 있으면 LW-SSO 프레임워크에서는 하나의 값만 허용됩니다.
- **Internet Explorer 7을 사용 중인 경우 다중 도메인 로그아웃 기능.** 다음과 같은 상황에서는 다중 도메인 로그아웃 기능을 사용하지 못할 수 있습니다.
  - 사용하는 브라우저가 Internet Explorer 7이고, 응용 프로그램의 로그아웃 절차에서 HTTP 302 리디렉션 동사를 연속 4회 이상 호출하는 경우

  이 경우 Internet Explorer 7에서 HTTP 302 리디렉션 응답을 잘못 처리하여 **Internet Explorer에서 웹 페이지를 표시할 수 없습니다.**라는 오류 메시지가 대신 표시될 수 있습니다.

  이 문제를 해결하려면 가능한 경우 로그아웃 시퀀스에서 응용 프로그램 리디렉션 명령 횟수를 줄이는 것이 좋습니다.

제한

LW-SSO 인증을 사용할 때 다음 제한 사항에 유의하십시오.

- **응용 프로그램에 대한 클라이언트 액세스**

  도메인이 LW-SSO 구성에 정의된 경우:
  - 응용 프로그램 클라이언트가 로그인 URL에 http://myserver.**companydomain**.com/WebApp과 같은 FQDN(정규화된 도메인 이름)을 사용하여 응용 프로그램에 액세스해야 합니다.

    참고: FQDN 길이는 인프라 설정 관리자의 **최대 도메인 확장 길이** 설정 값보다 길 수 없습니다. 기본값은 8입니다.

- LW-SSO는 IP 주소(예: http://192.168.12.13/WebApp)를 사용하는 URL을 지원하지 않습니다.
- LW-SSO는 도메인이 없는 URL(예: http://myserver/WebApp)을 지원하지 않습니다.

LW-SSO 구성에 도메인이 정의되어 있지 않은 경우: 클라이언트가 로그인 URL에 FQDN을 사용하지 않아도 응용 프로그램에 액세스할 수 있습니다. 이 경우 단일 시스템에 대한 LW-SSO 세션 쿠키가 도메인 정보 없이 특별히 만들어집니다. 따라서 쿠키는 브라우저를 통해 다른 컴퓨터로 위임되지 않으며 같은 DNS 도메인에 있는 다른 컴퓨터로 전달되지 않습니다. 이는 LW-SSO가 동일한 도메인에서 작동하지 않음을 의미합니다.

- **LW-SSO 프레임워크 통합.**응용 프로그램은 사전에 LW-SSO 프레임워크 내에서 통합된 경우에만 LW-SSO 기능을 활용할 수 있습니다.
- **다중 도메인 지원**
  - 다중 도메인 기능은 HTTP 참조 페이지에 기반합니다. 따라서 LW-SSO는 응용 프로그램 간 링크를 지원하고 브라우저 창에 URL 입력은 지원하지 않습니다(두 응용 프로그램이 동일한 도메인에 있는 경우는 예외).
  - 도메인 간의 첫 번째 링크는 **HTTP POST**를 사용할 수 없습니다.

    다중 도메인 기능은 두 번째 응용 프로그램에 대한 첫 번째 **HTTP POST** 요청을 지원하지 않고 **HTTP GET** 요청만 지원됩니다. 예를 들어 응용 프로그램에 두 번째 응용 프로그램에 대한 HTTP 링크가 있는 경우 **HTTP GET** 요청은 지원되지만 **HTTP FORM** 요청은 지원되지 않습니다. 첫 번째 요청 뒤의 모든 요청은 **HTTP POST** 또는 **HTTP GET** 중 하나입니다.
  - LW-SSO 토큰 크기:

    한 도메인의 응용 프로그램에서 다른 도메인의 응용 프로그램으로 LW-SSO가 전송할 수 있는 정보의 크기는 그룹/역할/특성 15개로 제한됩니다(각 항목의 길이는 평균 15자로 간주함).
  - 다중 도메인 시나리오 중 보호되는(HTTPS)페이지에서 보호되지 않는(HTTP) 페이지로 연결하는 경우:

    보호되는(HTTPS)페이지에서 보호되지 않는(HTTP) 페이지로 연결할 때는 다중 도메인 기능이 작동하지 않습니다. 이것은 보호되는 리소스에서 보호되지 않는 리소스로 연결할 경우 참조 페이지 헤더를 보내지 않는 브라우저의 제한입니다. 예는 다음 페이지를 참조하십시오.
    http://support.microsoft.com/support/kb/articles/Q178/0/66.ASP
  - Internet Explorer에서 타사 쿠키의 동작:

    Microsoft Internet Explorer 6에는 "P3P(개인 정보 기본 설정용 플랫폼) 프로젝트"를 지원하는 모듈이 포함되어 있습니다. 즉, 타사 도메인에서 보내는 쿠키는 인터넷 보안 영역에서 기본적으로 차단됩니다. IE에서는 세션 쿠키도 타사 쿠키로 간주되어 차단되기 때문에 LW-SSO의 작동이 중지됩니다. 자세한 내용은 http://support.microsoft.com/kb/323752/ko-kr을 참조하십시오.

    이 문제를 해결하려면 시작한 응용 프로그램이나 *.mydomain.com과 같은 DNS 도메인 하위 집합을 컴퓨터의 인트라넷/신뢰할 수 있는 영역에 추가합니다. 이렇게 하려면 Microsoft Internet Explorer에서 **메뉴 > 도구 > 인터넷 옵션 > 보안 > 로컬 인트라넷 > 사이트 > 고급**을 선택합니다.

그러면 쿠키가 허용됩니다.

주의: LW-SSO 세션 쿠키는 차단된 타사 응용 프로그램에서 사용하는 유일한 쿠키입니다.

- **SAML2 토큰**
  - SAML2 토큰을 사용할 때는 로그아웃 기능이 지원되지 않습니다.

    따라서 SAML2 토큰을 사용하여 두 번째 응용 프로그램에 액세스하는 경우 첫 번째 응용 프로그램에서 로그아웃하는 사용자가 두 번째 응용 프로그램에서는 로그아웃되지 않습니다.
  - **응용 프로그램 세션 관리에는 SAML2 토큰 만료가 반영되지 않습니다.**

    따라서 SAML2 토큰을 사용하여 두 번째 응용 프로그램에 액세스하는 경우 각 응용 프로그램의 세션 관리는 독립적으로 처리됩니다.
- **JAAS 영역.**Tomcat의 JAAS 영역은 지원되지 않습니다.
- **Tomcat 디렉터리에 공백 사용.**Tomcat 디렉터리에서는 공백을 사용할 수 없습니다.

  Tomcat 설치 경로(폴더)에 공백이 있고(예: Program Files) LW-SSO 구성 파일이 **common\classes** Tomcat 폴더에 있는 경우 LW-SSO를 사용할 수 없습니다.
- **로드 균형 조정 구성.**스티키 세션을 사용하도록 LW-SSO와 함께 배포된 로드 밸런서를 구성해야 합니다.
- **데모 모드.** 데모 모드에서는 LW-SSO가 응용 프로그램 간의 링크를 지원하지만, HTTP 참조 페이지 헤더가 없기 때문에 브라우저 창에 URL을 입력할 수는 없습니다.

7장: HP Universal CMDB 로그인 인증

이 장의 내용:

인증 방법 설정

인증을 수행하려면 다음 작업을 수행할 수 있습니다.

- **내부 HP Universal CMDB 서비스에 대해**
- **LDAP(Lightweight Directory Access Protocol)를 통해 인증.** 내부 HP Universal CMDB 서비스를 사용하는 대신 전용 외부 LDAP 서버를 사용하여 인증 정보를 저장할 수 있습니다. LDAP 서버는 모든 HP Universal CMDB 서버와 같은 서브넷에 있어야 합니다.

  LDAP에 대한 자세한 내용은 *HP Universal CMDB 관리 안내서*의 LDAP 매핑에 관한 섹션을 참조하십시오.

  기본 인증 방법에서는 내부 HP Universal CMDB 서비스가 사용됩니다. 기본 방법을 사용하는 경우에는 시스템을 변경하지 않아도 됩니다.

  이러한 옵션은 웹 서비스를 통해 수행하는 로그인과 사용자 인터페이스를 통해 수행하는 로그인에 모두 적용됩니다.
- **LW-SSO를 통해 인증.** HP Universal CMDB에는 LW-SSO가 구성되어 있습니다. LWSSO를 사용하는 경우, HP Universal CMDB에 로그인하면 같은 도메인에서 실행되는 다른 구성된 응용 프로그램에도 로그인하지 않고 자동으로 액세스할 수 있습니다.

  LW-SSO 인증 지원을 사용하는 경우(기본적으로는 사용하지 않도록 설정됨) Single Sign-On 환경의 다른 응용 프로그램에서도 LW-SSO를 사용하고 같은 initString 매개 변수를 사용하는지 확인해야 합니다.

LW-SSO를 사용하여 HP Universal CMDB에 로그인하도록 설정

1. 웹 브라우저 주소창에 **http://<server_name>:8080/jmx-console**을 입력하여 JMX 콘솔에 액세스합니다. 여기서 **<server_name>**은 HP Universal CMDB가 설치된 컴퓨터 이름입니다.
2. **UCMDB-UI**에서 **name=LW-SSO Configuration**을 클릭하여 작업 페이지를 엽니다.
3. **setInitString** 메서드를 사용하여 초기 문자열을 설정합니다.
4. **setDomain** 메서드를 사용하여 UCMDB가 설치된 컴퓨터의 도메인 이름을 설정합니다.
5. 매개 변수를 **True**로 설정하여 **setEnabledForUI** 메서드를 호출합니다.
6. **선택 사항**. 다중 도메인 기능을 사용하려는 경우 **addTrustedDomains** 메서드를 선택하고 도메인 값을 입력한 후 **Invoke**를 클릭합니다.
7. **선택 사항입니다.** 리버스 프록시를 사용하여 작업하려는 경우 **updateReverseProxy** 메서드를 선택하고 **Is reverse proxy enabled** 매개 변수를 **True**로 설정한 다음 **Reverse proxy full server URL** 매개 변수에 URL을 입력하고 Invoke를 클릭합니다. UCMDB에 직접 액세스하는 방법과 리버스 프록시를 통해 액세스하는 방법을 모두 사용하려면 **setReverseProxyIPs** 메서드를 선택하고, 리버스 프록시 IP/매개 변수에 대한 IP 주소를 입력하고, **Invoke**를 클릭하는 추가 구성 설정을 수행합니다.
8. **선택 사항**. 외부 인증 지점을 사용하여 UCMDB에 액세스하려는 경우, **setValidationPointHandlerEnable** 메서드를 선택하고 **Is validation point handler enabled** 매개 변수를 **True**로 설정한 후 **Authentication point server** 매개 변수에서 인증 지점의 URL을 입력하고 **Invoke**를 클릭합니다.
9. 설정 메커니즘에 저장된 LW-SSO 구성을 확인하려면 **retrieveConfigurationFromSettings** 메서드를 호출합니다.
10. 실제로 로드된 LW-SSO 구성을 확인하려면 **retrieveConfiguration** 메서드를 호출합니다.

참고: 사용자 인터페이스를 통해 LW-SSO를 사용하도록 설정할 수는 없습니다.

SSL(Secure Sockets Layer) 프로토콜을 사용하여 보안 연결 설정

로그인 프로세스 중에는 HP Universal CMDB와 LDAP 서버 간에 기밀 정보가 전달되므로 컨텐츠에 특정 보안 수준을 적용할 수 있습니다. 이렇게 하려면 LDAP 서버에서 SSL 통신을 사용하도록 설정하고 HP Universal CMDB가 SSL을 사용하여 작동하도록 구성하면 됩니다.

HP Universal CMDB에서는 신뢰할 수 있는 CA(인증 기관)에서 발급한 인증서를 사용하는 SSL을 지원합니다.

Active Directory를 비롯한 대부분의 LDAP 서버는 SSL 기반 연결을 위해 보안 포트를 노출할 수 있습니다. Active Directory와 개인 CA를 사용 중인 경우에는 JRE에서 해당 CA를 신뢰할 수 있는 CA에 추가해야 할 수 있습니다.

SSL을 사용한 통신을 지원하도록 HP Universal CMDB 플랫폼을 구성하는 방법에 대한 자세한 내용은 "SSL(Secure Sockets Layer) 통신 사용"(16페이지)을 참조하십시오.

SSL 기반 연결을 위해 보안 포트를 노출하도록 CA를 신뢰할 수 있는 CA에 추가하려면 다음을 수행합니다.

1. 다음 단계를 수행하여 CA에서 인증서를 내보낸 다음 HP Universal CMDB에서 사용하는 JVM으로 가져옵니다.
   a. UCMDB 서버 컴퓨터에서 **UCMDBServer\bin\JRE\bin** 폴더에 액세스합니다.
   b. 다음 명령을 실행합니다.

      Keytool -import -file <your certificate file> -keystore C:\hp\UCMDB\UCMDBServer\bin\JRE\lib\security\cacerts

      예:

      Keytool -import -file c:\ca2ss_ie.cer -keystore C:\hp\UCMDB\UCMDBServer\bin\JRE\lib\security\cacerts

2. **관리 > 인프라 설정 > LDAP 일반** 범주를 선택합니다.

   참고: JMX 콘솔을 사용하여 이러한 설정을 구성할 수도 있습니다. 자세한 내용은 "JMX 콘솔을 사용하여 LDAP 인증 방법을 사용하도록 설정하고 정의하는 방법"(90페이지)을 참조하십시오.

3. **LDAP 서버 URL**을 찾은 후에 값을 다음 형식으로 입력합니다.

   ldaps://<ldapHost>[:<port>]/[<baseDN>][??scope]

   예:

   ldaps://my.ldap.server:389/ou=People,o=myOrg.com??sub

   ldaps와 같이 **s**를 붙여야 합니다.

4. **저장**을 클릭하여 새 값을 저장하거나, **기본값 복원**을 클릭하여 항목을 기본값(빈 URL)으로 바꿉니다.

JMX 콘솔을 사용하여 LDAP 연결 테스트

이 섹션에서는 JMX 콘솔을 사용하여 LDAP 인증 구성을 테스트하는 방법을 설명합니다.

1. 웹 브라우저를 시작하고 주소창에 **http://<server_name>:8080/jmx-console**을 입력합니다. 여기서 **<server_name>**은 HP Universal CMDB가 설치된 컴퓨터 이름입니다.

   사용자 이름과 비밀번호를 입력하여 로그인해야 할 수도 있습니다.

2. **UCMDB**에서 **UCMDB:service=LDAP Services**를 클릭하여 작업 페이지를 엽니다.
3. **testLDAPConnection**을 찾습니다.

4. **customer id** 매개 변수의 **Value** 상자에 고객 ID를 입력합니다.
5. **Invoke**를 클릭합니다.

   JMX MBEAN Operation Result 페이지에 LDAP 연결의 성공 여부가 표시됩니다. 연결이 성공한 경우에는 이 페이지에 LDAP 루트 그룹도 표시됩니다.

LDAP 인증 방법을 사용하도록 설정하고 정의하는 방법

HP Universal CMDB 시스템에 대해 LDAP 인증 방법을 사용하도록 설정하고 정의할 수 있습니다.

참고:

- JMX 콘솔을 사용하여 LDAP 인증 설정을 구성할 수도 있습니다. 자세한 내용은 "JMX 콘솔을 사용하여 LDAP 인증 방법을 사용하도록 설정하고 정의하는 방법"(90페이지)을 참조하십시오.
- LDAP 인증 설정의 예는 "LDAP 인증 설정 - 예"(91페이지)를 참조하십시오.

UCMDB 사용자 인터페이스에 LDAP 인증 방법을 설정하고 정의하려면 다음을 수행합니다.

1. **관리 > 인프라 설정 > LDAP 일반** 범주를 선택합니다.
2. **LDAP 서버 URL**을 선택하고 LDAP URL 값을 다음 형식으로 입력합니다.

   ```
   ldap://<ldapHost>[:<port>]/[<baseDN>][??scope]
   ```

   예:

   ```
   ldap://my.ldap.server:389/ou=People,o=myOrg.com??sub
   ```

3. **LDAP 그룹 정의** 범주를 선택하고 **그룹 기본 DN**을 찾은 후에 일반 그룹의 고유 이름을 입력합니다.
4. **루트 그룹 기본 DN**을 찾아 루트 그룹의 고유 이름을 입력합니다.
5. **LDAP 일반** 범주를 선택하고 **사용자 권한 동기화 사용**을 찾은 후에 해당 값이 **True**로 설정되어 있는지 확인합니다.
6. **LDAP 일반 인증** 범주를 선택하고 **검색 권한을 가지는 사용자의 비밀번호**를 찾은 후에 비밀번호를 입력합니다.
7. **클래스 및 특성에 대한 LDAP 옵션** 범주를 선택하고 **그룹 클래스 개체**를 찾은 다음 개체 클래스 이름을 채웁니다(Microsoft Active Directory의 경우 **group**, Oracle Directory Server의 경우 **groupOfUniqueNames**).
8. **그룹 구성원 특성**을 찾아 특성 이름을 채웁니다(Microsoft Active Directory의 경우 **member**, Oracle Directory Server의 경우 **uniqueMember**).
9. **사용자 개체 클래스**를 찾아 개체 클래스 이름을 채웁니다(Microsoft Active Directory의 경우 **user**, Oracle Directory Server의 경우 **inetOrgPerson**).

10. **UUID 특성**을 찾아 디렉터리 서버에 있는 사용자의 고유 식별 특성을 채웁니다. 디렉터리 서버에서 고유한 특성을 선택해야 합니다. 예를 들어 SunOne/Oracle Directory Server를 사용하는 경우 UID 특성은 고유하지 않습니다. 그런 경우에는 전자 메일 주소 특성이나 고유 이름을 사용합니다. 고유하지 않은 특성을 UCMDB에서 고유 식별 특성으로 사용하면 로그인 동작의 일관성이 손상될 수 있습니다.
11. 새 값을 저장합니다. 항목을 기본값으로 바꾸려면 **기본값 복원**을 클릭합니다.
12. **LDAP 일반**의 인프라 설정인 **LDAP 인증에 대/소문자 구분 적용**이 **True**이면 인증은 대/소문자를 구분합니다.

    주의: 이 인프라 설정 값이 변경된 경우에는 모든 외부 사용자를 UCMDB 관리자가 수동으로 삭제해야 합니다.

13. LDAP 사용자 그룹을 UCMDB 사용자 그룹에 매핑합니다. 자세한 내용은 "HP Universal CMDB 로그인 인증"(86페이지)을 참조하십시오.
14. LDAP 그룹에서 그룹 매핑이 없는 사용자에 대해 기본 권한 집합을 정의하려면 **LDAP 일반** 범주를 선택하고 **자동으로 할당된 사용자 그룹**을 찾은 다음 그룹 이름을 입력합니다.
15. **중요:** 최고 가용성 환경에서 LDAP를 구성 중인 경우 변경 내용이 적용되도록 클러스터를 다시 시작해야 합니다.

참고: 모든 LDAP 사용자의 이름, 성 및 전자 메일 주소가 로컬 저장소에 저장됩니다. LDAP 서버에 저장된 이러한 매개 변수 중에 로컬 저장소와 값이 다른 항목이 있는 경우에는 로그인할 때마다 LDAP 서버 값이 로컬 값을 덮어씁니다.

JMX 콘솔을 사용하여 LDAP 인증 방법을 사용하도록 설정하고 정의하는 방법

이 작업에서는 JMX 콘솔을 사용하여 LDAP 인증 설정을 구성하는 방법을 설명합니다.

참고:

- 최고 가용성 환경에서 Writer 서버의 JMX 콘솔로 로그인했는지 확인합니다.
- UCMDB의 LDAP 인증 설정을 구성할 수도 있습니다. 자세한 내용은 "LDAP 인증 방법을 사용하도록 설정하고 정의하는 방법"(89페이지)을 참조하십시오.
- LDAP 인증 설정의 예는 "LDAP 인증 설정 - 예"(91페이지)를 참조하십시오.

LDAP 인증 설정을 구성하려면 다음을 수행합니다.

1. 웹 브라우저를 시작하고 주소창에 **http://<server_name>:8080/jmx-console**을 입력합니다. 여기서 **<server_name>**은 HP Universal CMDB가 설치된 컴퓨터 이름입니다.

   사용자 이름과 비밀번호를 입력하여 로그인해야 할 수도 있습니다.
2. **UCMDB**에서 **UCMDB:service=LDAP Services**를 클릭하여 작업 페이지를 엽니다.

3. 현재 LDAP 인증 설정을 보려면 **getLDAPSettings** 메서드를 찾은 다음 **Invoke**를 클릭합니다. 그러면 표에 모든 LDAP 설정과 해당 값이 표시됩니다.
4. LDAP 인증 설정의 값을 변경하려면 **configureLDAP** 메서드를 찾은 다음 관련 설정의 값을 입력하고 **Invoke**를 클릭합니다. JMX MBEAN Operation Result 페이지에 LDAP 인증 설정 업데이트 성공 여부가 표시됩니다.

   참고: 설정에 대해 값을 입력하지 않으면 설정의 현재 값이 유지됩니다.

5. LDAP 설정을 구성한 후에는 LDAP 사용자 자격 증명을 확인할 수 있습니다.
   a. **verifyLDAPCredentials** 메서드를 찾은 다음
   b. 고객 ID, 사용자 이름 및 비밀번호를 입력합니다.
   c. **Invoke**를 클릭합니다.

      JMX MBEAN Operation Result 페이지에 사용자의 LDAP 인증 통과 여부가 표시됩니다.
6. **중요:** 최고 가용성 환경에서 LDAP를 구성 중인 경우 변경 내용이 적용되도록 클러스터를 다시 시작해야 합니다.

참고: 모든 LDAP 사용자의 이름, 성 및 전자 메일 주소가 로컬 저장소에 저장됩니다. LDAP 서버에 저장된 이러한 매개 변수 중에 로컬 저장소와 값이 다른 항목이 있는 경우에는 로그인할 때마다 LDAP 서버 값이 로컬 값을 덮어씁니다.

LDAP 인증 설정 - 예

다음 표에는 LDAP 인증 값 설정 예가 나와 있습니다.

설정	값
사용자 개체 클래스	user
LDAP 인증에 대/소문자 구분 적용	false
그룹 구성원 특성	member
DN(고유 이름) 확인	true
루트 그룹 필터	(objectCategory=group)
LDAP 연결 문자열	ldap://myldap.example.com:389/OU=Users,OU=Dept,OU=US,DC=example,DC=com??sub
LDAP 검색 사용자	CN=John Doe,OU=Users,OU=Dept,OU=US,DC=example,DC=com

설정	값
그룹 클래스 개체	group
상위 그룹을 찾는데 바텀업 알고리즘을 사용	true
UUID 특성	sAMAccountName
그룹 이름 특성	cn
그룹 기본 필터	(objectclass=group)
사용자 필터	(&(sAMAccountName=*)(objectclass=user))
검색 다시 시도 횟수	3
그룹 표시 이름 특성	cn
루트 그룹 범위	sub
사용자 표시 이름 특성	cn
그룹 검색 범위	sub
LDAP 인증 사용	false
LDAP 동기화 사용	true
루트 그룹	OU=Users,OU=Security Groups,DC=example,DC=com
그룹 기본	OU=AMRND,OU=Security Groups,DC=example,DC=com
기본 그룹	관리 그룹
그룹 설명 특성	description

분산 환경에서 현재 LW-SSO 구성 검색

BSM 배포와 같이 UCMDB가 분산 환경에 포함되어 있는 경우 처리 컴퓨터에서 현재 LW-SSO 구성을 검색하려면 다음 절차를 수행합니다.

현재 LW-SSO 구성을 검색하려면 다음을 수행합니다.

1. 웹 브라우저를 시작하고 주소창에 http://localhost.<domain_name>:8080/jmx-console을 입력합니다.

사용자 이름과 비밀번호를 입력하라는 메시지가 표시될 수 있습니다.

2. **UCMDB:service=Security Services**를 찾은 다음 링크를 클릭하여 작업 페이지를 엽니다.
3. **retrieveLWSSOConfiguration** 작업을 찾습니다.
4. **Invoke**를 클릭하여 구성을 검색합니다.

8장: Confidential Manager

이 장의 내용:

Confidential Manager 개요

Confidential Manager 프레임워크는 HP Universal CMDB 및 기타 HP소프트웨어 제품의 중요한 데이터를 관리하고 배포하는 문제를 해결해 줍니다.

Confidential Manager는 클라이언트와 서버라는 두 가지 기본 구성 요소로 구성됩니다. 이 두 구성 요소는 안전한 방식으로 데이터를 전송합니다.

- Confidential Manager 클라이언트는 응용 프로그램에서 중요한 데이터에 액세스하는 데 사용하는 라이브러리입니다.
- Confidential Manager 서버는 Confidential Manager 클라이언트 또는 타사 클라이언트로부터 요청을 받아 필요한 작업을 수행합니다. Confidential Manager 서버는 안전한 방식으로 데이터를 저장합니다.

Confidential Manager는 전송, 클라이언트 캐시, 지속성, 그리고 메모리 작업 시에 자격 증명을 암호화합니다. Confidential Manager는 공유 암호를 사용하여 Confidential Manager 클라이언트와 Confidential Manager 서버 간에 자격 증명을 전송할 때 대칭 암호화를 사용합니다. Confidential Manager는 구성에 따라 캐시, 지속성 및 전송을 암호화하기 위해 다양한 암호를 사용합니다.

Data Flow Probe에서 자격 증명 암호화를 관리하는 자세한 지침은 "데이터 흐름 자격 증명 관리"(47페이지)를 참조하십시오.

보안 고려 사항

- 보안 알고리즘에는 128, 192 및 256비트의 키 크기를 사용할 수 있습니다. 키 크기 값이 작을수록 알고리즘 실행 속도는 빨라지지만 보안성은 떨어집니다. 대부분의 경우에는 128비트 크기를 사용해도 충분히 안전합니다.

- 시스템 보안을 향상시키려면 MAC를 사용합니다. **useMacWithCrypto**를 **true**로 설정합니다. 자세한 내용은 "암호화 속성"(96페이지)을 참조하십시오.
- 강력한 고객 보안 공급자를 활용하려는 경우 JCE 모드를 사용할 수 있습니다.

HP Universal CMDB 서버 구성

HP Universal CMDB를 사용할 때는 다음과 같은 JMX 방법을 사용하여 암호화의 암호 및 암호화 속성을 구성해야 합니다.

1. HP Universal CMDB 서버 컴퓨터에서 웹 브라우저를 시작하고 서버 주소를 **http://<UCMDB Server Host Name or IP>:8080/jmx-console**과 같이 입력합니다.

   사용자 이름과 비밀번호를 입력하여 로그인해야 할 수도 있습니다.
2. UCMDB에서 **UCMDB:service=Security Services**를 클릭하여 작업 페이지를 엽니다.
3. 현재 구성을 검색하려면 **CMGetConfiguration** 작업을 찾습니다.

   Invoke를 클릭하여 Confidential Manager 서버 구성 XML 파일을 표시합니다.
4. 구성을 변경하려면 이전 단계에서 호출한 XML을 텍스트 편집기에 복사합니다. "암호화 속성"(96페이지)의 표를 참고하여 변경합니다.

   CMSetConfiguration 작업을 찾습니다. 업데이트된 구성을 **Value** 상자에 복사하고 **Invoke**를 클릭합니다. 새 구성이 UCMDB 서버에 기록됩니다.
5. Confidential Manager에 인증 및 복제할 사용자를 추가하려면 **CMAddUser** 작업을 찾습니다. 이 프로세스는 복제 프로세스에서도 유용합니다. 복제 시에 서버 슬레이브는 권한이 있는 사용자로 서버 마스터와 통신해야 합니다.
   - **username**. 사용자 이름입니다.
   - **customer.** 기본값은 ALL_CUSTOMERS입니다.
   - **resource.** 리소스 이름입니다. 기본값은 ROOT_FOLDER입니다.
   - **permission.** ALL_PERMISSIONS, CREATE, READ, UPDATE, DELETE 중에서 선택합니다. 기본값은 ALL_PERMISSIONS입니다.

   Invoke를 클릭합니다.
6. 필요한 경우 HP Universal CMDB를 다시 시작합니다.

   대부분의 경우에는 서버를 다시 시작할 필요가 없습니다. 다음 리소스 중 하나를 변경할 때는 서버를 다시 시작해야 할 수 있습니다.
   - 저장소 유형
   - 데이터베이스 테이블 이름 또는 열 이름
   - 데이터베이스 연결을 만든 사람

- 데이터베이스 연결 속성(URL, 사용자, 비밀번호, 드라이버 클래스 이름)
- 데이터베이스 유형

참고:

- UCMDB 서버와 해당 클라이언트의 전송 암호화 속성은 같아야 합니다. UCMDB 서버에서 이러한 속성을 변경하는 경우에는 모든 클라이언트에서 속성을 변경해야 합니다. Data Flow Probe의 경우에는 UCMDB 서버와 같은 프로세스에서 실행되므로 이와 같이 속성을 변경하지 않아도 됩니다. 즉, 전송 암호화 구성이 필요하지 않습니다.
- Confidential Manager 복제는 기본적으로 구성되지 않으며 필요한 경우 구성할 수 있습니다.
- Confidential Manager 복제가 사용하도록 설정된 경우 마스터의 전송 **initString** 또는 기타 암호화 속성을 변경하면 모든 슬레이브에서 이 변경 내용을 적용해야 합니다.

정의

저장소 암호화 속성. 서버에서 데이터를 저장하고 암호화하는 방법(데이터베이스 또는 파일에서 데이터를 암호화하거나 암호를 해독하는 데 사용할 암호화 속성), 자격 증명을 안전한 방식으로 저장하는 방법, 암호화를 처리하는 방법 및 따를 구성을 정의하는 구성입니다.

전송 암호화 속성. 전송 구성은 서버와 클라이언트에서 둘 사이의 전송을 암호화하는 방법, 사용할 구성, 자격 증명을 안전한 방식으로 전송하는 방법, 암호화를 처리하는 방법 및 따를 구성을 정의합니다. 서버와 클라이언트 둘 다에서 전송 암호화 및 암호 해독에 동일한 암호화 속성을 사용해야 합니다.

복제 및 복제 암호화 속성. Confidential Manager에서 안전하게 저장하는 데이터는 여러 서버 간에 안전하게 복제됩니다. 이러한 속성은 슬레이브 서버와 마스터 서버 간에 데이터를 전송하는 방법을 정의합니다.

참고:

- Confidential Manager 서버 구성이 저장되는 데이터베이스 테이블 이름은 **CM_CONFIGURATION**으로 지정됩니다.
- Confidential Manager 서버의 기본 구성 파일은 app-infra.jar에 있으며 이름은 **defaultCMServerConfig.xml**입니다.

암호화 속성

다음 표에는 암호화 속성에 대한 설명이 나와 있습니다. 이러한 매개 변수에 대한 자세한 내용은 "HP Universal CMDB 서버 구성"(95페이지)을 참조하십시오.

매개 변수	설명	권장 값
encryptTransportMode	전송되는 데이터를 암호화합니다. true false	true
encryptDecrypt InitString	암호화에 사용되는 비밀번호입니다.	9자 이상
cryptoSource	사용할 암호화 구현 라이브러리입니다. • lw • jce • windowsDPAPI • lwJCECompatible	lw
lwJCEPBE CompatibilityMode	이전의 경량 암호화 버전을 지원합니다. • true • false	true
cipherType	Confidential Manager에서 사용하는 암호화 유형입니다. Confidential Manager는 한 값만 지원합니다. **symmetricBlockCipher**	symmetric BlockCipher
engineName	• AES • Blowfish • DES • 3DES • Null(암호화 안 함)	AES
algorithmModeName	블록 암호화 알고리즘의 모드입니다. • CBC	CBC
algorithmPaddingName	패딩 표준입니다. • PKCS7Padding • PKCS5Padding	PKCS7Padding
keySize	알고리즘(**engineName**이 지원하는 항목)에 따라 달라집니다.	256
pbeCount	**encryptDecryptInitString**에서 키를 만들기 위해 해시를 실행할 횟수입니다.	1000

매개 변수	설명	권장 값
	임의의 양수입니다.	
pbeDigestAlgorithm	해시 유형입니다. • SHA1 • SHA256 • MD5	SHA256
encodingMode	암호화된 개체의 ASCII 표현입니다. • Base64 • Base64Url	Base64Url
useMacWithCrypto	암호화와 함께 MAC를 사용할지 여부를 정의합니다. • true • false	false
macType	MAC(메시지 인증 코드)의 유형입니다. • hmac	hmac
macKeySize SHA256	MAC 알고리즘에 따라 달라집니다.	256
macHashName	해시 MAC 알고리즘입니다. • SHA256	SHA256

9장: 최고 가용성 강화

이 장의 내용:

클러스터 인증

클러스터 인증을 사용하려면 다음을 수행합니다.

1. UCMDB에서 **관리 > 인프라 설정 관리자**로 이동합니다.
2. **Enable joining High Availability cluster authentication** 설정을 찾아 **true**로 설정합니다.
3. 단일 서버 인증 키 저장소(인증서 + 개인 및 공용 키)를 JKS 형식으로 제공합니다. 이 키 저장소는 모든 서버에 배치되며 최고 가용성 클러스터에 연결할 때 인증용으로 사용됩니다.

   키 저장소를 **<UCMDB 설치 폴더>\conf\security** 위치에 두고 **cluster.authentication.keystore**로 이름을 바꿉니다.

   참고: UCMDB는 미리 구성된 이 기본 키 저장소와 함께 제공됩니다. 이 키 저장소는 새로 설치되는 모든 UCMDB에 대해 동일하므로 안전하지 않습니다. 조인 요청을 안전하게 인증하려면 이 파일을 삭제하고 새로 만듭니다.

4. 다음과 같이 클러스터 인증 키 저장소를 생성합니다.
   a. C:\hp\UCMDB\UCMDBServer\bin\jre\bin에서 다음 명령을 실행합니다.

      ```
      keytool -genkey -alias hpcert  keystore <UCMDB 설치 폴더>\conf\security\cluster.authentication.keystore -keyalg RSA
      ```

      콘솔 대화 상자가 열리고 새 키 저장소 비밀번호를 묻는 메시지가 나타납니다.

   b. 기본 비밀번호는 **hppass**입니다. 다른 비밀번호를 사용하려면 JMX 메서드를 **UCMDB:service=High Availability Services: changeClusterAuthenticationKeystorePassword**와 같이 실행하여 서버를 업데이트합니다.

   c. 콘솔 대화 상자에서 **이름과 성은 무엇입니까?**라는 질문에 클러스터 이름을 입력합니다.

   d. 다른 매개 변수는 조직의 세부 정보에 맞게 입력합니다.

   e. 키 비밀번호를 입력합니다. 키 비밀번호는 키 저장소 비밀번호와 같아야 합니다.

      JKS 키 저장소는 **<UCMDB 설치 폴더>\conf\security\cluster.authentication.keystore**에 만들어집니다.

5. 클러스터의 모든 서버에 있는 이전 **<UCMDB 설치 폴더>\conf\security\cluster.authentication.keystore**를 새 키 저장소로 바꿉니다.
6. 클러스터의 모든 서버를 다시 시작합니다.

클러스터 메시지 암호화

클러스터 메시지 암호화를 사용하여 클러스터의 모든 메시지를 암호화합니다.

클러스터 메시지 암호화를 사용하려면 다음을 수행합니다.

1. UCMDB에서 **관리 > 인프라 설정 관리자**로 이동합니다.
2. **최고 가용성 클러스터 통신 암호화 사용** 설정을 찾아 **true**로 설정합니다.
3. 모든 서버의 대칭 암호화를 위한 보안 키를 제공합니다. 키는 **<UCMDB 설치 폴더>\conf\security\cluster.encryption.keystore** 위치의 JCEKS 유형의 키 저장소에 보관되어야 합니다.

   참고: UCMDB는 미리 구성된 이 기본 키 저장소와 함께 제공됩니다. 이 키 저장소는 새로 설치되는 모든 UCMDB에 대해 동일하므로 안전하지 않습니다. 클러스터 메시지를 안전하게 암호화하려면 이 파일을 삭제하고 다음 절차에 따라 새로 만드십시오.

4. **<UCMDB 설치 폴더>\bin\jre\bin**에서 다음 명령을 실행합니다.

   Keytool -genseckey -alias hpcert -keystore <UCMDB 설치 폴더>\conf\security\cluster.encryption.keystore -storetype JCEKS
5. 새 키 저장소 비밀번호를 묻는 메시지가 나타납니다. 기본 비밀번호는 "hppass"입니다. 다른 비밀번호를 사용하려면 다음 JMX 메서드를 실행하여 서버를 업데이트해야 합니다.

   UCMDB:service=High Availability Services: changeClusterEncryptionKeystorePassword
6. 클러스터의 모든 서버에 있는 이전 **<UCMDB 설치 폴더>\conf\security\cluster.encryption.keystore**를 새 키 저장소로 바꿉니다.
7. 서버를 다시 시작합니다.

문제 해결

서버를 시작할 때마다 클러스터에 성공적으로 연결되었는지 확인하는 테스트 메시지가 클러스터에 바로 전송됩니다. 연결에 문제가 있으면 메시지가 전송되지 않으며 전체 클러스터가 중단되는 것을 막기 위해 서버가 중지됩니다.

잘못된 클러스터 암호화 구성의 예를 들면 다음과 같습니다.

- 한 노드에서는 암호화를 사용하고 다른 노드에서는 사용하지 않도록 설정된 경우
- cluster.encryption.keystore가 잘못되거나 누락된 경우
- 키 저장소의 키가 잘못되거나 누락된 경우

구성 문제로 인해 서버가 중단되는 경우 오류 메시지는 다음과 같습니다.

2012-09-11 17:48:23,584 [Thread-14] FATAL - #### Server failed to connect properly to the cluster and its service is stopped! Please fix the problem and start it again ####

2012-09-11 17:48:23,586 [Thread-14] FATAL - Potential problems can be: wrong security configuration (wrong or missing cluster.encryption.keystore, wrong key, disabled encryption in a cluster with enabled encryption)

key.bin에서 키 변경

여러 서버를 사용하는 최고 가용성 환경에서는 **key.bin**의 **key**를 다음과 같이 변경합니다.

1. JMX에서 writer 컴퓨터로 이동합니다. 클러스터의 아무 컴퓨터나 선택하고 각 페이지 위쪽에서 **writer** 링크를 클릭하면 됩니다.
2. 콘솔의 UCMDB 섹션에서 **UCMDB:service=Discovery Manager**를 클릭합니다.
3. 다음 중 한 가지 방법으로 키를 변경합니다.
   - **changeEncryptionKey**를 클릭합니다(기존 암호화 키를 가져오게 됨).
   - **generateEncryptionKey**를 클릭합니다(임의의 암호화 키가 생성됨).
4. writer 컴퓨터에서 파일 시스템으로 이동하여 **C:\hp\UCMDB\UCMDBServer\conf\discovery\key.bin**에서 **key.bin**을 찾습니다.
5. writer 컴퓨터의 해당 위치에서 **key.bin**을 클러스터의 다른 각각의 컴퓨터의 **C:\hp\UCMDB\UCMDBServer\conf\discovery\customer_1** 폴더에 복사하고 대상 파일 이름을 바꿉니다(예: **key_new.bin**).
6. 다른 서버들(reader) 각각에 대해서는 다음을 수행합니다.
   a. reader를 writer로 전환하고(최고 가용성 JMX에서 수행 가능) 변경될 때까지 기다립니다.
   b. 현재 writer의 JMX에 연결하고 **UCMDB:service=Discovery Manager**를 클릭합니다.
   c. **changeEncryptionKey**를 클릭하여 호출하고 3단계에서 입력한 동일한 세부 정보를 사용합니다. **newKeyFileName**에 대해서는 5단계에서 할당한 새 이름을 사용합니다.
   d. **Key was created successfully** 메시지가 나타나는지 확인합니다.

문서 피드백 보내기

이 문서에 대한 의견이 있는 경우 전자 메일로 문서 팀에 의견을 보내 주십시오. 이 시스템에 전자 메일 클라이언트가 구성되어 있을 경우 위의 링크를 클릭하면 제목 줄에 다음 정보가 포함된 전자 메일 창이 열립니다.

강화 안내서에 대한 피드백(Universal CMDB 및 Configuration Manager 10.20)

전자 메일에 피드백을 추가하고 보내기를 클릭하기만 하면 됩니다.

전자 메일 클라이언트를 사용할 수 없으면 위의 정보를 웹 메일 클라이언트에서 새 메시지에 복사하고 피드백을 cms-doc@hp.com에 보내십시오.

피드백을 보내 주셔서 감사합니다!