Red Hat Enterprise Linux 8

IdM에서 인증서 관리

인증서 발행, 인증서 기반 인증 구성, Red Hat Enterprise Linux 8의 ID 관리에서 인증서 유효성 제어

Last Updated: 2022-08-21

Red Hat Enterprise Linux 8 IdM에서 인증서 관리

인증서 발행, 인증서 기반 인증 구성, Red Hat Enterprise Linux 8의 ID 관리에서 인증서 유효성 제어

Enter your first name here. Enter your surname here.
Enter your organisation's name here. Enter your organisational division here.
Enter your email address here.

법적 공지

초록

이 문서에서는 IdM 인증 기관에서 발급한 인증서 관리, 인증에 인증서를 사용하도록 사용자 계정 구성, Red Hat Enterprise Linux 8의 ID 관리에서 인증서 유지 관리에 대해 다룹니다.

차례

보다 포괄적 수용을 위한 오픈 소스 용어 교체

Red Hat은 코드, 문서, 웹 속성에서 문제가 있는 용어를 교체하기 위해 최선을 다하고 있습니다. 먼저 마스터(master), 슬레이브(slave), 블랙리스트(blacklist), 화이트리스트(whitelist) 등 네 가지 용어를 교체하고 있습니다. 이러한 변경 작업은 작업 범위가 크므로 향후 여러 릴리스에 걸쳐 점차 구현할 예정입니다. 자세한 내용은 CTO Chris Wright의 메시지 를 참조하십시오.

ID 관리에서 계획된 용어 교체는 다음과 같습니다.

- ***차단 목록*** *블랙리스트*대체
- ***허용 목록*** 대체 *허용 목록*
- ***슬레이브*** 교체
- *마스터* 라는 단어는 컨텍스트에 따라 보다 정확한 언어로 교체되고 있습니다.
  - ***IdM 서버가*** *IdM 마스터*교체
  - ***CA 갱신 마스터를 대체하는CA 갱신서버***
  - ***CRL 게시자 서버는*** *CRL 마스터*교체
  - ***다중*** *마스터*교체

RED HAT 문서에 관한 피드백 제공

문서 개선을 위한 의견을 보내 주십시오. Red Hat이 어떻게 이를 개선할 수 있는지 알려 주십시오.

- 특정 문구에 대한 간단한 의견 작성 방법은 다음과 같습니다.
  1. 문서가 *Multi-page HTML* 형식으로 표시되는지 확인합니다. 또한 문서 오른쪽 상단에 **피드백** 버튼이 있는지 확인합니다.
  2. 마우스 커서를 사용하여 주석 처리하려는 텍스트 부분을 강조 표시합니다.
  3. 강조 표시된 텍스트 아래에 표시되는 **피드백 추가** 팝업을 클릭합니다.
  4. 표시된 지침을 따릅니다.
- Bugzilla를 통해 피드백을 제출하려면 새 티켓을 생성합니다.
  1. Bugzilla 웹 사이트로 이동하십시오.
  2. 구성 요소로 **문서**를 사용합니다.
  3. **설명** 필드에 문서 개선을 위한 제안 사항을 기입하십시오. 관련된 문서의 해당 부분 링크를 알려주십시오.
  4. **버그 제출**을 클릭합니다.

1장. IDENTITY MANAGEMENT의 공개 키 인증서

이 장에서는 IdM(Identity Management)의 사용자, 호스트 및 서비스를 인증하는 데 사용되는 X.509 공개 키 인증서에 대해 설명합니다. 인증 외에도 X.509 인증서는 디지털 서명 및 암호화를 통해 개인 정보 보호, 무결성 및 비결합을 제공할 수 있습니다.

인증서에는 다음 정보가 포함되어 있습니다.

- 인증서가 인증하는 주체입니다.
- 인증서에 서명한 CA입니다.
- 인증서의 유효성의 시작 및 종료 날짜입니다.
- 인증서의 유효한 사용입니다.
- 주체의 공개 키입니다.

공개 키로 암호화된 메시지는 해당 개인 키로만 암호를 해독할 수 있습니다. 포함된 인증서 및 공개 키는 공개적으로 사용할 수 있지만 사용자, 호스트 또는 서비스는 개인 키 시크릿을 유지해야 합니다.

1.1. IDM의 인증 기관

인증 기관은 신뢰 계층 구조로 작동합니다. 내부 CA(인증 기관)가 있는 IdM 환경에서 모든 IdM 호스트, 사용자 및 서비스 신뢰 인증서는 CA에서 서명한 인증서를 신뢰합니다. 이 루트 CA 외에도 IdM은 루트 CA에 인증서를 서명하는 기능을 부여하는 하위 CA를 지원합니다. 이러한 하위 CA가 서명할 수 있는 인증서는 특정 종류의 인증서(예: VPN 인증서)입니다. 마지막으로 IdM은 외부 CA 사용을 지원합니다. 아래 표에는 IdM에서 개별 CA 유형 사용의 세부 사항이 나와 있습니다.

표 1.1. IdM에서 통합 및 외부 CA 사용 비교

CA 이름	설명	사용	유용한 링크
ipa CA	Dogtag 업스트림 프로젝트를 기반으로 하는 통합 CA	통합 CA는 사용자, 호스트 및 서비스에 대한 인증서를 생성, 취소 및 발행할 수 있습니다.	ipa CA를 사용하여 새 사용자 인증서를 요청하고 클라이언트로 내보내기
IdM 하위 CA	**ipa** CA에 종속된 통합 CA	IdM 하위 보안은 **ipa** CA에 인증서에 서명하는 기능이 부여된 CA입니다. 일반적으로 이러한 인증서는 특정 종류의 (예: VPN 인증서)입니다.	인증서의 하위 집합만 신뢰하도록 애플리케이션 제한
외부 CA	외부 CA는 통합된 IdM CA 또는 해당 하위 CA 이외의 CA입니다.	IdM 툴을 사용하여 이러한 CA에서 발급한 인증서를 사용자, 서비스 또는 호스트에 추가하고 제거할 수 있습니다.	IdM 사용자, 호스트 및 서비스용 외부 서명된 인증서 관리

인증서 관점에서는 자체 서명된 IdM CA에서 서명하고 외부적으로 서명하는 것 사이에 차이가 없습니다.

CA의 역할에는 다음과 같은 용도가 포함됩니다.

- 이는 디지털 인증서를 발급합니다.

- 인증서에 서명하면 인증서에서 이름이 지정된 주체가 공개 키를 소유하고 있음을 증명합니다. 주체는 사용자, 호스트 또는 서비스일 수 있습니다.
- 인증서를 해지할 수 있으며 인증서 해지 목록(CRL) 및 OCSP(Online Certificate Status Protocol)를 통해 해지 상태를 제공할 수 있습니다.

추가 리소스

- CA 서비스 계획을 참조하십시오.

1.2. 인증서 및 KERBEROS 비교

인증서는 Kerberos 티켓에서 수행하는 것과 유사한 기능을 수행합니다. Kerberos는 티켓에 따라 작동하는 컴퓨터 네트워크 인증 프로토콜로, 보안되지 않은 네트워크를 통해 통신하는 노드가 안전한 방식으로 자신의 ID를 서로 증명할 수 있도록 합니다. 다음 표에서는 Kerberos 및 X.509 인증서의 비교를 보여줍니다.

표 1.2. 인증서 및 Kerberos 비교

특성	Kerberos	X.509
인증	있음	있음
개인 정보 보호	선택 사항	있음
무결성	선택 사항	있음
암호화 유형 관련	symmetrical	symmetrical
기본 유효성 검사	짧은 (1일)	long(2년)

기본적으로 Identity Management의 Kerberos는 통신 당사자의 ID만 보장합니다.

1.3. 인증서를 사용하여 IDM의 사용자를 인증하는 프록시 및 제한 사항

IdM에서 사용자를 인증하기 위해 인증서를 사용할 때의 이점은 다음과 같습니다.

- 스마트 카드에서 개인 키를 보호하는 PIN은 일반적으로 덜 복잡하고 일반 암호보다 쉽게 기억할 수 있습니다.
- 장치에 따라 스마트 카드에 저장된 개인 키를 내보낼 수 없습니다. 이는 추가 보안을 제공합니다.
- 스마트 카드는 logout automatic을 수행할 수 있습니다. 리더에서 스마트 카드를 제거할 때 사용자를 로그아웃하도록 IdM을 구성할 수 있습니다.
- 개인 키를 훔치려면 스마트 카드에 대한 실제 물리적 액세스 권한이 필요하므로 해킹 공격으로부터 스마트 카드를 안전하게 보호할 수 있습니다.
- 스마트 카드 인증은 2 단계 인증의 예입니다. 즉, 카드 및 사용자가 알고있는 것 (POP)이 모두 필요합니다.
- 스마트 카드는 이메일 암호화와 같은 다른 용도로 사용할 수 있는 키를 제공하기 때문에 암호보다 유연합니다.

- IdM 클라이언트인 공유 시스템에서 스마트 카드를 사용하면 일반적으로 시스템 관리자에게 추가 구성 문제가 발생하지 않습니다. 실제로 스마트 카드 인증은 공유 시스템에 이상적인 옵션입니다.

IdM의 사용자를 인증하기 위해 인증서를 사용하는 단점은 다음과 같습니다.

- 사용자는 스마트 카드 또는 인증서를 가져오고 효과적으로 잠글 수 있습니다.
- Pinstyping을 여러 번 수행하면 카드가 잠길 수 있습니다.
- 일반적으로 일종의 보안 담당자 또는 승인자의 요청과 승인 사이에 중간 단계가 있습니다. IdM에서 보안 담당자 또는 관리자는 **ipa cert-request** 명령을 실행해야 합니다.
- 스마트 카드 및 독자는 벤더와 드라이버별 경향이 있습니다: 많은 독자가 다른 카드에 사용될 수 있지만 특정 공급 업체의 스마트 카드는 다른 공급 업체의 독자 또는 설계되지 않은 리더의 유형에서 작동하지 않을 수 있습니다.
- 인증서 및 스마트 카드에는 관리자를 위한 맞춤형 학습 곡선이 있습니다.

2장. 통합된 IDM CA를 사용하여 사용자, 호스트 및 서비스의 인증서 관리

이 장에서는 통합 CA, **ipa** CA 및 해당 하위 CA를 사용하여 IdM(Identity Management)에서 인증서를 관리하는 방법을 설명합니다.

이 장에서는 다음 섹션이 포함되어 있습니다.

- IdM 웹 UI를 사용하여 사용자, 호스트 또는 서비스에 대한 새 인증서 요청 .
- IdM CLI를 사용하여 IdM CA에서 사용자, 호스트 또는 서비스에 대한 새 인증서 요청:
  - certutil을 사용하여 IdM CA에서 사용자, 호스트 또는 서비스에 대한 새 인증서 요청
    - **certutil** 유틸리티를 사용하여 IdM CA에서 새 사용자 인증서를 요청하고 IdM 클라이언트로 내보내는 구체적인 예는 새 사용자 인증서를 요청하여 클라이언트로 내보내는 것을 참조하십시오.
  - openssl을 사용하여 IdM CA에서 사용자, 호스트 또는 서비스에 대한 새 인증서 요청

certmonger 유틸리티를 사용하여 IdM CA에서 서비스에 대한 새 인증서를 요청할 수도 있습니다. 자세한 내용은 certmonger를 사용하여 IdM CA에서 서비스에 대한 새 인증서 요청을 참조하십시오.

사전 요구 사항

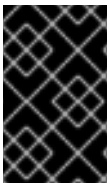
- IdM 배포에는 통합 CA가 포함되어 있습니다.
  - IdM에서 CA 서비스를 계획하는 방법에 대한 자세한 내용은 CA 서비스 계획을 참조하십시오.
  - 통합 DNS 및 통합 CA를 루트 CA로 사용하여 IdM 서버를 설치하는 방법에 대한 자세한 내용은 IdM 서버 설치를 참조하십시오. 통합 DNS와 함께 루트 CA로 통합 CA
  - 통합 DNS 및 외부 CA를 루트 CA로 사용하여 IdM 서버를 설치하는 방법에 대한 자세한 내용은 IdM 서버 설치를 참조하십시오. 루트 CA로 외부 CA가 있는 통합 DNS 사용
  - 통합 DNS 및 통합 CA를 루트 CA로 사용하여 IdM 서버를 설치하는 방법에 대한 자세한 내용은 IdM 서버 설치를 참조하십시오. 통합 DNS가 없으면 루트 CA 로 통합 CA가 포함됩니다.
  - [선택 사항] IdM 배포는 인증서로 인증하는 사용자를 지원합니다.
    - IdM 클라이언트 파일 시스템에 저장된 인증서로 사용자 인증을 지원하도록 IdM 배포를 구성하는 방법에 대한 자세한 내용은 IdM 클라이언트의 데스크탑에 저장된 인증서로 인증 구성을 참조하십시오.
    - IdM 클라이언트에 삽입된 스마트 카드에 저장된 인증서로 사용자 인증을 지원하도록 IdM 배포를 구성하는 방법에 대한 자세한 내용은 스마트 카드 인증에 대한 ID 관리 구성을 참조하십시오.
    - Active Directory 인증서 시스템에서 발행한 스마트 카드로 사용자 인증을 지원하도록 IdM 배포를 구성하는 방법에 대한 자세한 내용은 IdM의 스마트 카드 인증에 대해 ADCS에서 발급한 인증서 구성을 참조하십시오.

2.1. IDM 웹 UI를 사용하여 사용자, 호스트 또는 서비스에 대한 새 인증서 요청

이 섹션에서는 IdM(Identity Management) 웹 UI를 사용하여 통합 IdM 인증 기관(CA)에서 IdM 엔터티에 대한 새 인증서를 요청하는 방법에 대해 설명합니다. **ipa** CA 또는 모든 하위 CA입니다.

IdM 엔티티는 다음과 같습니다.

- 사용자
- 호스트
- 서비스



중요

일반적으로 서비스는 개인 키가 저장된 전용 서비스 노드에서 실행됩니다. 서비스의 개인 키를 IdM 서버에 복사하는 것은 안전하지 않은 것으로 간주됩니다. 따라서 서비스에 대한 인증서를 요청할 때 서비스 노드에 인증서 서명 요청(CSR)을 생성합니다.

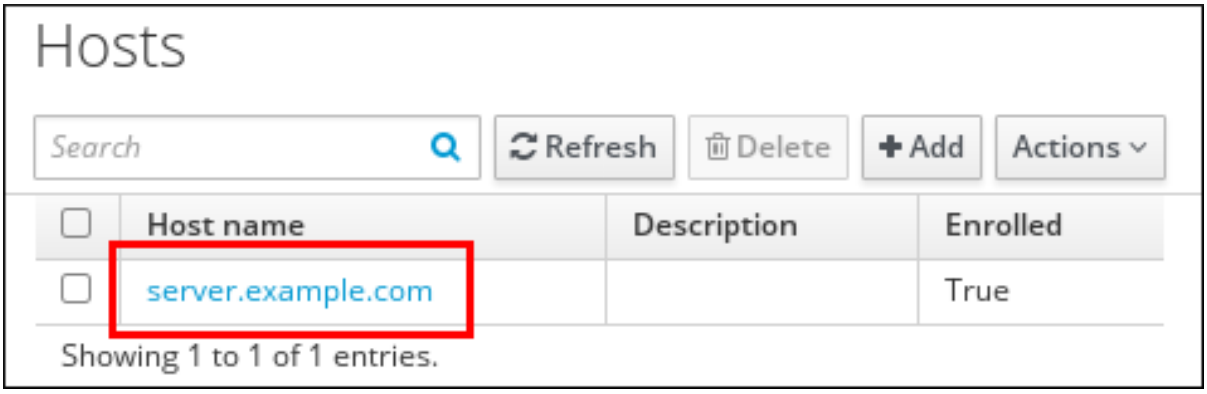
사전 요구 사항

- IdM 배포에는 통합 CA가 포함되어 있습니다.
- IdM 관리자로 IdM 웹 UI에 로그인되어 있습니다.

절차

1. **Identity** (ID) 탭에서 **Users**,**Hosts** (호스트) 또는 **Services** (서비스) 하위 탭을 선택합니다.
2. 사용자, 호스트 또는 서비스의 이름을 클릭하여 구성 페이지를 엽니다.

   그림 2.1. 호스트 목록

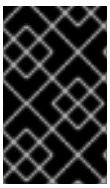


3. **Actions → New Certificate** 을 클릭합니다.
4. 선택 사항: 발행 CA 및 프로필 ID를 선택합니다.
5. 화면에서 **certutil** 명령줄(CLI) 유틸리티 사용 지침을 따릅니다.
6. **Issue** 를 클릭합니다.

2.2. CERTUTIL을 사용하여 IDM CA에서 사용자, 호스트 또는 서비스에 대한 새 인증서 요청

certutil 유틸리티를 사용하여 표준 IdM 상황에서 IdM(Identity Management) 사용자, 호스트 또는 서비스에 대한 인증서를 요청할 수 있습니다. 호스트 또는 서비스 Kerberos 별칭에서 인증서를 사용할 수 있도록 하려면 openssl 유틸리티를 사용하여 인증서를 요청합니다.

이 섹션에서는 **certutil** 을 사용하여 **ipa**, IdM 인증 기관(CA)에서 IdM 사용자, 호스트 또는 서비스에 대한 인증서를 요청하는 방법을 설명합니다.



중요

일반적으로 서비스는 개인 키가 저장된 전용 서비스 노드에서 실행됩니다. 서비스의 개인 키를 IdM 서버에 복사하는 것은 안전하지 않은 것으로 간주됩니다. 따라서 서비스에 대한 인증서를 요청할 때 서비스 노드에 인증서 서명 요청(CSR)을 생성합니다.

사전 요구 사항

- IdM 배포에는 통합 CA가 포함되어 있습니다.
- IdM 관리자로서 IdM(명령줄 인터페이스)에 로그인되어 있습니다.

절차

1. 인증서 데이터베이스의 임시 디렉터리를 생성합니다.

```
# mkdir ~/certdb/
```

2. 임시 인증서 데이터베이스를 새로 생성합니다. 예를 들면 다음과 같습니다.

```
# certutil -N -d ~/certdb/
```

3. CSR을 생성하고 출력을 파일로 리디렉션합니다. 예를 들어 4096비트 인증서에 대한 CSR을 생성하고 제목을 *CN=server.example.com,O=EXAMPLE.COM*:

```
# certutil -R -d ~/certdb/ -a -g 4096 -s "CN=server.example.com,O=EXAMPLE.COM" -8 server.example.com > certificate_request.csr
```

4. IdM 서버에서 실행 중인 CA에 인증서 요청 파일을 제출합니다. 새로 발급된 인증서와 연결할 Kerberos 주체를 지정합니다.

```
# ipa cert-request certificate_request.csr --principal=host/server.example.com
```

IdM의 **ipa cert-request** 명령은 다음 기본값을 사용합니다.

- **caIPAserviceCert** 인증서 프로필
  사용자 정의 프로필을 선택하려면 **--profile-id** 옵션을 사용합니다.
- 통합 IdM 루트 CA, **ipa**
  하위 서비스를 선택하려면 **--ca** 옵션을 사용합니다.

추가 리소스

- **ipa cert-request --help** 명령의 출력을 참조하십시오.
- ID 관리에서 인증서 프로파일 생성 및 관리를 참조하십시오.

2.3. OPENSSL을 사용하여 IDM CA에서 사용자, 호스트 또는 서비스에 대한 새 인증서 요청

호스트 또는 서비스의 Kerberos 별칭이 인증서를 사용할 수 있도록 하려면 **openssl** 유틸리티를 사용하여 IdM(Identity Management) 호스트 또는 서비스에 대한 인증서를 요청할 수 있습니다. 표준 상황에서는 certutil 유틸리티를 사용하여 새 인증서를 요청하는 것이 좋습니다.

이 섹션에서는 **openssl** 을 사용하여 IdM 호스트 또는 IdM 인증 기관인 **ipa** 로부터 인증서를 요청하는 방법을 설명합니다.

중요

일반적으로 서비스는 개인 키가 저장된 전용 서비스 노드에서 실행됩니다. 서비스의 개인 키를 IdM 서버에 복사하는 것은 안전하지 않은 것으로 간주됩니다. 따라서 서비스에 대한 인증서를 요청할 때 서비스 노드에 인증서 서명 요청(CSR)을 생성합니다.

사전 요구 사항

- IdM 배포에는 통합 CA가 포함되어 있습니다.
- IdM 관리자로서 IdM(명령줄 인터페이스)에 로그인되어 있습니다.

절차

1. Kerberos 주체 *test/server.example.com* 에 대한 별칭을 하나 이상 생성합니다. 예를 들어 *test1/server.example.com* 및 *test2/server.example.com*.
2. CSR에서 dnsName(*server.example.com*) 및 otherName(*test2/server.example.com*)의 subjectAltName을 추가합니다. 이렇게 하려면 UPN otherName 및 subjectAltName을 지정하는 다음 행을 포함하도록 **openssl.conf** 파일을 구성합니다.

```
otherName=1.3.6.1.4.1.311.20.2.3;UTF8:test2/server.example.com@EXAMPLE.COM
DNS.1 = server.example.com
```

3. **openssl** 을 사용하여 인증서 요청 생성 :

```
openssl req -new -newkey rsa:2048 -keyout test2service.key -sha256 -nodes -out
certificate_request.csr -config openssl.conf
```

4. IdM 서버에서 실행 중인 CA에 인증서 요청 파일을 제출합니다. 새로 발급된 인증서와 연결할 Kerberos 주체를 지정합니다.

```
# ipa cert-request certificate_request.csr --principal=host/server.example.com
```

IdM의 **ipa cert-request** 명령은 다음 기본값을 사용합니다.

- **caIPAserviceCert** 인증서 프로필
  사용자 정의 프로필을 선택하려면 **--profile-id** 옵션을 사용합니다.
- 통합 IdM 루트 CA, **ipa**
  하위 서비스를 선택하려면 **--ca** 옵션을 사용합니다.

추가 리소스

- **ipa cert-request --help** 명령의 출력을 참조하십시오.
- ID 관리에서 인증서 프로파일 생성 및 관리를 참조하십시오.

2.4. 추가 리소스

- 통합 IdM CA가 있는 인증서 해지를 참조하십시오.
- 통합 IdM CA가 있는 인증서 복원을 참조하십시오.
- 인증서의 하위 집합만 신뢰하도록 애플리케이션 제한을 참조하십시오.

3장. IDM 사용자, 호스트 및 서비스용 외부 서명된 인증서 관리

이 장에서는 IdM(Identity Management) 명령줄 인터페이스(CLI)와 IdM 웹 UI를 사용하여 외부 인증 기관(CA)에서 발급한 사용자, 호스트 또는 서비스 인증서를 추가하거나 제거하는 방법을 설명합니다.

3.1. IDM CLI를 사용하여 외부 CA에서 발급한 인증서 추가, IDM 사용자, 호스트 또는 서비스

IdM(Identity Management) 관리자는 IdM(Identity Management) CLI를 사용하여 IdM 사용자, 호스트 또는 서비스 계정에 외부 서명된 인증서를 추가할 수 있습니다.

사전 요구 사항

- 관리 사용자의 티켓 개발 티켓이 있습니다.

절차

- IdM 사용자에게 인증서를 추가하려면 다음을 입력합니다.

```
$ ipa user-add-cert user --certificate=MIQTPrajQAwg...
```

명령을 사용하려면 다음 정보를 지정해야 합니다.

  - 사용자의 이름
  - Base64로 인코딩된 DER 인증서

참고

인증서 내용을 명령줄에 복사하고 붙여넣는 대신 인증서를 DER 형식으로 변환한 다음 Base64로 다시 코딩할 수 있습니다. 예를 들어 **user_cert.pem** 인증서를 **사용자** 에 추가하려면 다음을 입력합니다.

```
$ ipa user-add-cert user --certificate="$(openssl x509 -outform der -in
user_cert.pem | base64 -w 0)"
```

옵션을 추가하지 않고 **ipa user-add-cert** 명령을 실행할 수 있습니다.

IdM 호스트에 인증서를 추가하려면 다음을 입력합니다.

- **ipa host-add-cert**

IdM 서비스에 인증서를 추가하려면 다음을 입력합니다.

- **ipa service-add-cert**

추가 리소스

- 통합 IdM CA를 사용하여 사용자, 호스트 및 서비스의 인증서 관리

3.2. IDM 웹 UI를 사용하여 IDM 사용자, 호스트 또는 서비스에 외부 CA에서 발급한 인증서 추가

IdM(Identity Management) 관리자는 IdM(Identity Management) 웹 UI를 사용하여 IdM 사용자, 호스트 또는 서비스 계정에 외부 서명된 인증서를 추가할 수 있습니다.

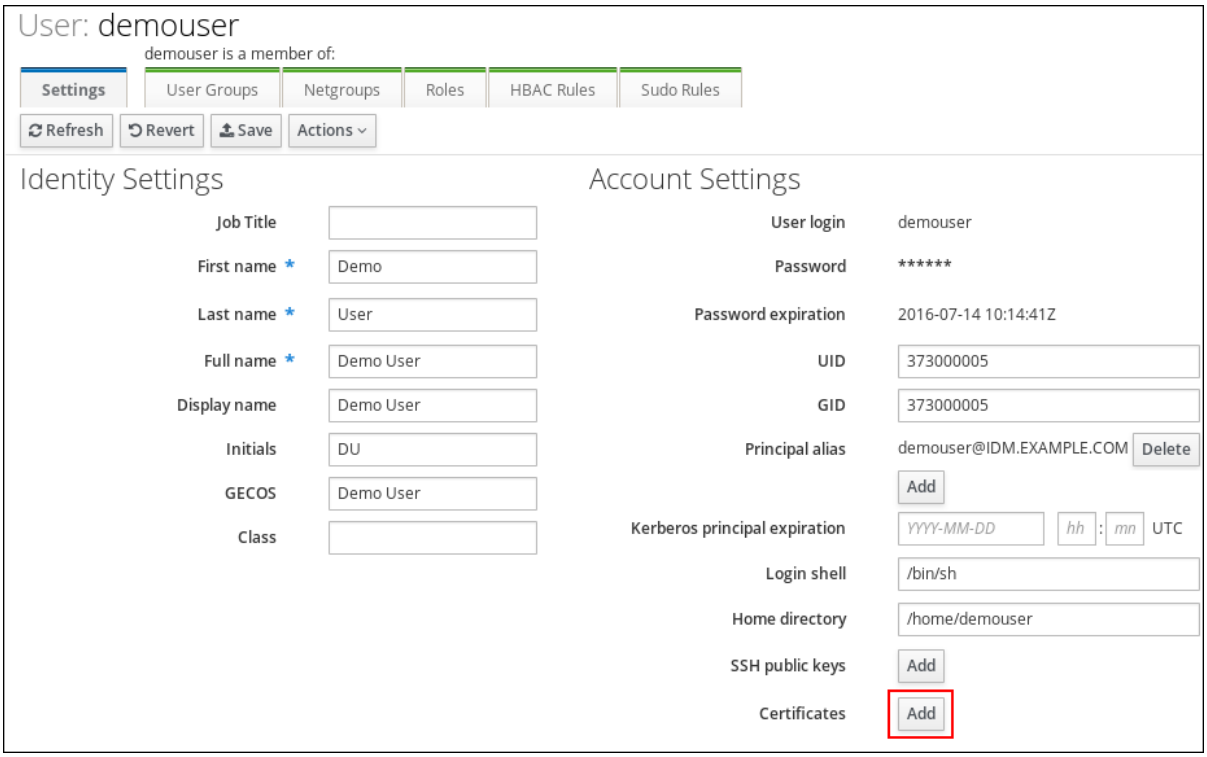
사전 요구 사항

- IdM(Identity Management) 웹 UI에 관리자로 로그인되어 있습니다.

절차

1. **Identity** 탭을 열고 **사용자,호스트** 또는 **서비스** 하위 탭을 선택합니다.
2. 사용자, 호스트 또는 서비스의 이름을 클릭하여 구성 페이지를 엽니다.
3. **Certificates** (인증서) 항목 옆에 있는 **Add** (추가)를 클릭합니다.

그림 3.1. 사용자 계정에 인증서 추가



4. Base64 또는 PEM으로 인코딩된 형식으로 인증서를 붙여넣고 텍스트 필드에 인증서를 붙여넣고 **추가를** 클릭합니다.
5. **저장을** 클릭하여 변경 사항을 저장합니다.

3.3. IDM CLI를 사용하여 IDM 사용자, 호스트 또는 서비스 계정에서 외부 CA에서 발급한 인증서 제거

IdM(Identity Management) 관리자는 IdM(Identity Management) CLI를 사용하여 IdM 사용자, 호스트 또는 서비스의 계정에서 외부 서명된 인증서를 제거할 수 있습니다.

사전 요구 사항

- 관리 사용자의 티켓 개발 티켓이 있습니다.

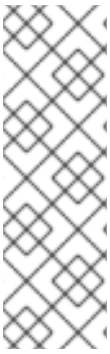
절차

- IdM 사용자의 인증서를 제거하려면 다음을 입력합니다.

```
$ ipa user-remove-cert user --certificate=MIQTPrajQAwg...
```

명령을 사용하려면 다음 정보를 지정해야 합니다.

  - 사용자의 이름
  - Base64로 인코딩된 DER 인증서



참고

인증서 내용을 명령줄에 복사하고 붙여넣는 대신 인증서를 DER 형식으로 변환한 다음 Base64로 다시 코딩할 수 있습니다. 예를 들어 사용자 에서 **user_cert.pem** 인증서를 제거하려면 다음을 입력합니다.

```
$ ipa user-remove-cert user --certificate="$(openssl x509 -outform der -in
user_cert.pem | base64 -w 0)"
```

옵션을 추가하지 않고 ipa user-remove-cert 명령을 대화형으로 실행할 수 있습니다.

IdM 호스트에서 인증서를 제거하려면 다음을 입력합니다.

- **ipa host-remove-cert**

IdM 서비스에서 인증서를 제거하려면 다음을 입력합니다.

- **ipa service-remove-cert**

추가 리소스

- **통합 IdM CA를 사용하여 사용자, 호스트 및 서비스의 인증서 관리**

3.4. IDM 웹 UI를 사용하여 IDM 사용자, 호스트 또는 서비스 계정에서 외부 CA에서 발급한 인증서 제거

IdM(Identity Management) 관리자는 IdM(Identity Management) 웹 UI를 사용하여 IdM 사용자, 호스트 또는 서비스의 계정에서 외부 서명된 인증서를 제거할 수 있습니다.

사전 요구 사항

- IdM(Identity Management) 웹 UI에 관리자로 로그인되어 있습니다.

절차

1. Identity 탭을 열고 사용자,호스트 또는 서비스 하위 탭을 선택합니다.
2. 사용자, 호스트 또는 서비스의 이름을 클릭하여 구성 페이지를 엽니다.
3. 삭제할 인증서 옆에 있는 작업을 클릭하고 삭제를 선택합니다.
4. 저장을 클릭하여 변경 사항을 저장합니다.

3.5. 추가 리소스

[role="_additional-resources"]

- Ansible 플레이북을 사용하여 IdM 서비스 항목에 외부 서명 인증서가 있는지 확인

4장. IDM으로 인증서 형식 변환

이 사용자 스토리는 IdM 시스템 관리자로서 특정 IdM 명령이 있는 인증서 형식을 올바르게 사용하고 있는지 확인하는 방법을 설명합니다. 예를 들어 다음과 같은 경우 유용합니다.

- 외부 인증서를 사용자 프로필에 로드하고 있습니다. 자세한 내용은 IdM 사용자 계정에 로드할 외부 인증서 변환을 참조하십시오.
- 스마트 카드 인증에 대해 IdM 서버를 구성하거나 스마트 카드 인증을 위해 IdM 클라이언트를 구성할 때 외부 CA 인증서를 사용하고 있어 사용자가 외부 인증 기관에서 발급한 인증서와 함께 스마트 카드를 사용하여 IdM을 인증할 수 있습니다.
- NSS 데이터베이스의 인증서를 인증서와 개인 키를 모두 포함하는 pkcs #12 형식으로 내보내고 있습니다. 자세한 내용은 NSS 데이터베이스에서 PKCS #12 파일로 인증서 및 개인 키 내보내기를 참조하십시오.

4.1. IDM의 인증서 형식 및 인코딩

IdM의 스마트 카드 인증을 포함한 인증서 인증은 사용자가 사용자의 IdM 프로필에 저장된 인증서 또는 인증서 데이터와 함께 제공하는 인증서를 비교하여 진행됩니다.

시스템 구성

IdM 프로필에 저장된 데이터는 해당 개인 키가 아닌 인증서일 뿐입니다. 인증 중에는 사용자가 해당 개인 키를 소유하고 있음을 표시해야합니다. 사용자는 인증서와 개인 키를 둘 다 포함하는 PKCS #12 파일을 제공하거나 인증서가 포함된 파일과 개인 키를 포함하는 다른 파일 중 하나를 제공합니다.

따라서 사용자 프로필에 인증서를 로드하는 등의 프로세스는 개인 키가 포함되지 않은 인증서 파일만 허용합니다.

마찬가지로 시스템 관리자가 외부 CA 인증서를 제공하면 개인 키가 없는 인증서인 공개 데이터만 제공합니다. IdM 서버 또는 스마트 카드 인증에 대한 IdM 클라이언트를 구성하기 위한 ipa-advise 유틸리티에는 입력 파일에 외부 CA의 인증서가 포함되지만 개인 키는 포함되지 않습니다.

인증서 인코딩

두 가지 일반적인 인증서 인코딩이 있습니다. 개인 정보 보호 규정 (PEM) 및 Distinguished octets 규칙 (DER). base64 형식은 PEM 형식과 거의 동일하지만 ---- BEGINCERTIFICATE--------END

CERTIFICATE---------- header 및 footer는 포함되어 있지 않습니다.

DER 를 사용하여 인코딩된 인증서는 바이너리 X509 디지털 인증서 파일입니다. 바이너리 파일로, 인증서는 사람이 읽을 수 없습니다. DER 파일에는 .der 파일 이름 확장자를 사용하는 경우가 있지만 .crt 및 .cer 파일 이름 확장자가 있는 파일에 DER 인증서도 포함되어 있는 경우가 있습니다. 키가 포함된 DER 파일의 이름은 .key . key일 수 있습니다.

PEM Base64를 사용하여 인코딩된 인증서는 사람이 읽을 수 있는 파일입니다. 파일에는 ASCII(Base64)가 "-----BEGIN" 행이 접두사로 붙은 데이터가 포함됩니다. PEM 파일에는 .pem 파일 이름 확장자를 사용하는 경우가 있지만 .crt 및 .cer 파일 이름 확장자가 있는 파일에도 PEM 인증서가 포함되어 있는 경우가 있습니다. 키가 포함된 PEM 파일의 이름은 .key . key일 수 있습니다.

ipa 명령과 다른 ipa 명령에는 허용되는 인증서 유형과 관련하여 다른 제한 사항이 있습니다. 예를 들어 ipa user-add-cert 명령은 base64 형식으로 인코딩된 인증서만 허용하지만 ipa-server-certinstall 은 PEM, DER, PKCS #7, PKCS #8 및 PKCS #12 인증서를 수락합니다.

표 4.1. 인증서 인코딩

인코딩 형식	사용자가 읽을 수 있는	일반적인 파일 이름 확장	인코딩 형식을 수락하는 샘플 IdM 명령
PEM/base64	있음	.pem, .crt, .cer	ipa user-add-cert, ipa-server-certinstall, ...
DER	없음	.der, .crt, .cer	ipa-server-certinstall, ...

IdM의 인증서 관련 명령 및 형식에는 명령이 허용하는 인증서 형식과 함께 추가 ipa 명령이 나열됩니다.

사용자 인증

웹 UI를 사용하여 IdM에 액세스하는 경우 사용자는 브라우저의 데이터베이스에 모두 저장되어 있는 개인 키의 인증서가 해당 개인 키임을 증명합니다.

CLI를 사용하여 IdM에 액세스하는 경우 사용자는 다음 방법 중 하나로 인증서에 해당하는 개인 키의 소유임을 증명합니다.

- 사용자는 kinit -X 명령의 X509_user_identity 매개변수 값으로, 인증서와 키가 모두 포함된 스마트 카드 모듈에 연결된 스마트 카드 모듈의 경로로 추가합니다.

```
$ kinit -X X509_user_identity='PKCS11:opensc-pkcs11.so' idm_user
```

- 사용자는 **kinit -X** 명령의 **X509_user_identity** 매개변수 값으로 두 개의 파일을 추가하고 다른 하나는 인증서를 포함하는 개인 키입니다.

```
$ kinit -X X509_user_identity='FILE:`/path/to/cert.pem,/path/to/cert.key`' idm_user
```

유용한 인증서 명령

주체 및 발급자와 같은 인증서 데이터를 보려면 다음을 수행합니다.

```
$ openssl x509 -noout -text -in ca.pem
```

두 개의 인증서가 다른 행을 비교하려면 다음을 수행합니다.

```
$ diff cert1.crt cert2.crt
```

두 개의 인증서가 두 열에 표시된 출력과 다른 행을 비교하려면 다음을 수행합니다.

```
$ diff cert1.crt cert2.crt -y
```

4.2. 외부 인증서를 IDM 사용자 계정으로 로드하도록 변환

이 섹션에서는 사용자 항목에 추가하기 전에 외부 인증서가 올바르게 인코딩되고 포맷되었는지 확인하는 방법에 대해 설명합니다.

4.2.1. 사전 요구 사항

- **Active Directory** 인증 기관에서 인증서를 발급하고 **PEM** 인코딩을 사용하는 경우 **PEM** 파일이 **UNIX** 형식으로 변환되었는지 확인합니다. 파일을 변환하려면 **eponymous** 패키지에서 제공하는 **dos2unix** 유틸리티를 사용하십시오.

4.2.2. IdM CLI에서 외부 인증서를 변환하고 IdM 사용자 계정에 로드

IdM CLI 는 첫 번째 행과 마지막 줄(**-----BEGINassRTIFICATE-----**) 및 **----ENDCERTIFICATE---------**) 이 제거된 **PEM** 인증서만 허용합니다.

다음 절차에 따라 외부 인증서를 **PEM** 형식으로 변환하고 **IdM CLI**를 사용하여 **IdM** 사용자 계정에 추가합니다.

절차

1. 인증서를 **PEM** 형식으로 변환합니다.

- 인증서 형식이 **DER** 인 경우:

```
$ openssl x509 -in cert.crt -inform der -outform pem -out cert.pem
```

- 파일이 **PKCS #12** 형식인 경우 일반 파일 이름 확장자가 **.pfx** 및 **.p12** 이고 인증서, 개인 키 및 기타 데이터를 포함하는 경우 **openssl pkcs12** 유틸리티를 사용하여 인증서를 추출합니다. 메시지가 표시되면 파일에 저장된 개인 키를 보호하는 암호를 입력합니다.

```
$ openssl pkcs12 -in cert_and_key.p12 -clcerts -nokeys -out cert.pem
Enter Import Password:
```

2. 관리자의 자격 증명을 가져옵니다.

```
$ kinit admin
```

3. 다음 방법 중 하나를 통해 **IdM CLI** 를 사용하여 사용자 계정에 인증서를 추가합니다.

- **ipa user-add-cert** 명령에 문자열을 추가하기 전에 **sed** 유틸리티를 사용하여 **PEM** 파일의 첫 번째 및 마지막 줄(**----BEGINCEIFICATE-----** 및 **------**)을 제거합니다.

```
$ ipa user-add-cert some_user --certificate="$(sed -e '/BEGIN
CERTIFICATE/d;/END CERTIFICATE/d' cert.pem)"
```

- 첫 번째 및 마지막 줄(**-----BEGINassRTIFICATE------**) 없이 인증서 파일의 콘텐츠를 복사하여 붙여 넣습니다. **ipa user-add-cert** 명령에 다음 명령을 실행합니다.

```
$ ipa user-add-cert some_user --
certificate=MIIDlzCCAn+gAwIBAgIBATANBgkqhki...
```

참고

인증서가 포함된 PEM 파일을 ipa user-add-cert 명령에 직접 전달할 수 없습니다. 첫 번째 행과 마지막 줄(-----BEGIN CERTIFICATE----------)을 먼저 제거하지 않고도 ipa user-add-cert 명령에 직접 PEM 파일을 전달할 수 없습니다.

```
$ ipa user-add-cert some_user --cert=some_user_cert.pem
```

이 명령을 실행하면 "ipa: ERROR: Base64 디코딩에 실패했습니다. 오류 메시지가 잘못되었습니다.

4. 시스템에 의해 인증서가 수락되었는지 확인하려면 다음을 수행합니다.

```
[idm_user@r8server]$ ipa user-show some_user
```

4.2.3. IdM 사용자 계정으로 로드하기 위해 IdM 웹 UI의 외부 인증서 변환

외부 인증서를 PEM 형식으로 변환한 후 IdM 웹 UI의 IdM 사용자 계정에 추가하려면 다음 절차를 따르십시오.

절차

1. CLI 를 사용하여 인증서를 PEM 형식으로 변환합니다.

- 인증서 형식이 DER 인 경우:

```
$ openssl x509 -in cert.crt -inform der -outform pem -out cert.pem
```

- 파일이 PKCS #12 형식인 경우 일반 파일 이름 확장자가 .pfx 및 .p12 이고 인증서, 개인 키 및 기타 데이터를 포함하는 경우 openssl pkcs12 유틸리티를 사용하여 인증서를 추출합니다. 메시지가 표시되면 파일에 저장된 개인 키를 보호하는 암호를 입력합니다.

```
$ openssl pkcs12 -in cert_and_key.p12 -clcerts -nokeys -out cert.pem
Enter Import Password:
```

2. 편집기에서 인증서를 열고 콘텐츠를 복사합니다. "-----BEGINCERTIFICATE-----" 및 "---ENDCERTIFICATE----" 헤더와 피터 라인을 포함할 수 있지만, 사용자는 IdM 웹 UI에서 PEM 및

base64 형식을 모두 수락할 수 있습니다.

3. IdM 웹 UI에서 보안 담당자로 로그인합니다.

4. Identity → Users → some_user 으로 이동합니다.

5. 인증서 옆에 있는 추가 를 클릭합니다.

6. 인증서의 PEM 형식의 콘텐츠를 열리는 창에 붙여넣습니다.

7. 추가를 클릭합니다.

시스템에서 인증서를 승인한 경우 사용자 프로필의 인증서 중 나열되는 내용을 확인할 수 있습니다.

4.3. 브라우저에 인증서 로드 준비

브라우저로 사용자 인증서를 가져오기 전에 인증서와 해당 개인 키가 PKCS #12 형식이어야 합니다. 추가 준비 작업이 필요한 두 가지 일반적인 상황이 있습니다.

- 인증서는 NSS 데이터베이스에 있습니다. 이 상황에서 진행하는 방법에 대한 자세한 내용은 NSS 데이터베이스에서 PKCS #12 파일로 인증서 및 개인 키 내보내기를 참조하십시오.

- 인증서와 개인 키는 두 개의 개별 PEM 파일에 있습니다. 이 상황에서 진행하는 방법에 대한 자세한 내용은 인증서 및 개인 키 PEM 파일을 PKCS #12 파일로 결합을 참조하십시오.

나중에 PKCS #12 형식의 PKCS #12 형식의 CA 인증서와 사용자 인증서를 모두 브라우저로 가져오려면 브라우저 구성의 절차에 따라 인증서 인증 및 인증서를 ID 관리 사용자로 사용하는 ID 관리 웹 UI를 인증 하십시오.

4.3.1. NSS 데이터베이스에서 인증서 및 개인 키 내보내기를 PKCS #12 파일로 내보내기

절차

1.

pk12util 명령을 사용하여 NSS 데이터베이스에서 PKCS12 형식으로 인증서를 내보냅니다. 예를 들어 ~/certdb 디렉터리에 저장된 NSS 데이터베이스에서 some_user nickname을 사용하여 인증서를 ~/some_user.p12 파일로 내보내려면 다음을 실행합니다.

```
$ pk12util -d ~/certdb -o ~/some_user.p12 -n some_user
Enter Password or Pin for "NSS Certificate DB":
Enter password for PKCS12 file:
Re-enter password:
pk12util: PKCS12 EXPORT SUCCESSFUL
```

2. .p12 파일에 적절한 권한을 설정합니다.

```
# chmod 600 ~/some_user.p12
```

PKCS #12 파일에는 개인 키가 포함되어 있으므로 다른 사용자가 파일을 사용하지 못하도록 보호해야 합니다. 그렇지 않으면 사용자를 가장할 수 있습니다.

4.3.2. 인증서와 개인 키 PEM 파일을 PKCS #12 파일로 결합

이 섹션에서는 인증서와 별도의 PEM 파일에 저장된 해당 키를 PKCS #12 파일로 결합하는 방법을 설명합니다.

절차

- certfile.cer 에 저장된 인증서와 certfile.key 에 저장된 키를 인증서와 키가 모두 포함된 certfile.p12 파일로 결합하려면 다음을 실행합니다.

```
$ openssl pkcs12 -export -in certfile.cer -inkey certfile.key -out certfile.p12
```

4.4. IDM의 인증서 관련 명령 및 형식

표 IdM 인증서 명령 및 형식은 IdM의 인증서 관련 명령을 허용 가능한 형식으로 표시합니다.

표 4.2. IdM 인증서 명령 및 형식

명령	허용 가능한 형식	참고
ipa user-add-cert some_user --certificate	base64 PEM 인증서	

명령	허용 가능한 형식	참고
ipa-server-certinstall	PEM 및 DER 인증서, PKCS#7 인증서 체인, PKCS#8 및 원시 개인 키; PKCS#12 인증서 및 개인 키	
ipa-cacert-manage install	DER; PEM; PKCS#7	
ipa-cacert-manage renew --external-cert-file	PEM 및 DER 인증서; PKCS#7 인증서 체인	
ipa-ca-install --external-cert-file	PEM 및 DER 인증서; PKCS#7 인증서 체인	
ipa cert-show <cert serial> --certificate-out /path/to/file.pem	해당 없음	**<cert_serial>** 일련 번호가 있는 인증서를 사용하여 PEM 인코딩 **file.pem** 파일을 생성합니다.
ipa cert-show <cert serial> --certificate-out /path/to/file.pem	해당 없음	**<cert_serial>** 일련 번호가 있는 인증서를 사용하여 PEM 인코딩 **file.pem** 파일을 생성합니다. **--chain** 옵션을 사용하는 경우 PEM 파일에 인증서 체인을 포함한 인증서가 포함되어 있습니다.
ipa cert-request --certificate-out=FILE /path/to/req.csr	해당 없음	새 인증서를 사용하여 PEM 형식으로 **req.csr** 파일을 생성합니다.
ipa cert-request --certificate-out=FILE /path/to/req.csr	해당 없음	새 인증서를 사용하여 PEM 형식으로 **req.csr** 파일을 생성합니다. **--chain** 옵션을 사용하는 경우 PEM 파일에 인증서 체인을 포함한 인증서가 포함되어 있습니다.

5장. IDENTITY MANAGEMENT에서 인증서 프로필 생성 및 관리

인증서 프로필은 인증서에 서명하여 CSR(인증서 서명 요청)이 허용되는지, 인증서 및 확장 기능이 인증서에 있는지 확인하는 경우 CA(인증 기관)에서 사용합니다. 인증서 프로필은 특정 유형의 인증서를 발급하는 것과 관련이 있습니다. 인증서 프로필과 ACL(액세스 제어 목록)을 결합하면 사용자 정의 인증서 프로필에 대한 액세스를 정의하고 제어할 수 있습니다.

인증서 프로필을 생성하는 방법을 설명할 때 프로시저는 S/journal 인증서를 예제로 사용합니다. 일부 이메일 프로그램은 S/octets(Secure Multipurpose Internet mail Extension) 프로토콜을 사용하여 디지털 서명되고 암호화된 이메일을 지원합니다. S/mtls를 사용하여 이메일 메시지에 서명하거나 암호화하려면 메시지 발신자가 S/octets 인증서를 보유해야 합니다.

- 인증서 프로필이란?
- 인증서 프로필 생성
- CA 액세스 제어 목록이란?
- 인증서 프로필에 대한 액세스를 제어하기 위한 CA ACL 정의
- 인증서 프로필 및 CA ACL을 사용하여 인증서 발급
- 인증서 프로필 수정
- 인증서 프로필 구성 매개변수

5.1. 인증서 프로필이란 무엇입니까?

인증서 프로필을 사용하여 다음과 같은 인증서를 발행하기 위한 제약 조건 및 인증서 내용을 확인할 수 있습니다.

- 인증서 서명 요청을 처리하는 데 사용할 서명 알고리즘입니다.

- 인증서의 기본 유효성입니다.
- 인증서를 취소하는 데 사용할 수 있는 해지 이유.
- 주체의 공통 이름이 주체 대체 이름 필드에 복사되는 경우.
- 인증서에 제공해야 하는 기능 및 확장입니다.

단일 인증서 프로필은 특정 유형의 인증서를 발급하는 것과 관련이 있습니다. IdM에서 사용자, 서비스 및 호스트에 대한 다양한 인증서 프로필을 정의할 수 있습니다. IdM에는 기본적으로 다음 인증서 프로필이 포함되어 있습니다.

- caIPAserviceCert
- IECUserRoles
- mtlss_PKINIT_Certs (내부에서 사용됨)

또한 특정 용도로 인증서를 발급할 수 있는 사용자 정의 프로필을 생성하고 가져올 수 있습니다. 예를 들어 특정 프로필의 사용을 하나의 사용자 또는 하나의 그룹으로만 제한하여 다른 사용자와 그룹이 해당 프로필을 사용하여 인증에 대한 인증서를 발급하지 못하도록 할 수 있습니다. 사용자 정의 인증서 프로필을 생성하려면 ipa certprofile 명령을 사용합니다.

추가 리소스

- ipa help certprofile 명령을 참조하십시오.

5.2. 인증서 프로필 생성

이 절차에서는 S/octets 인증서를 요청하는 프로필 구성 파일을 생성하여 명령줄을 통해 인증서 프로필을 생성하는 방법을 설명합니다.

절차

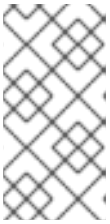
1. 기존 기본 프로필을 복사하여 사용자 정의 프로필을 생성합니다.

```
$ ipa certprofile-show --out smime.cfg caIPAserviceCert
-----------------------------------------------
Profile configuration stored in file 'smime.cfg'
-----------------------------------------------
  Profile ID: caIPAserviceCert
  Profile description: Standard profile for network services
  Store issued certificates: TRUE
```

2. 텍스트 편집기에서 새로 생성된 프로필 구성 파일을 엽니다.

```
$ vi  smime.cfg
```

3. 프로필 ID 를 프로필 사용량을 반영하는 이름으로 변경합니다(예: smime).



참고

새로 생성된 프로필을 가져오는 경우, profileId 필드는 명령줄에 지정된 ID와 일치해야 합니다.

4. 확장 키 사용량 구성을 업데이트합니다. 기본 확장 키 사용량 확장 프로그램은 TLS 서버 및 클라이언트 인증을 위한 것입니다. 예를 들어 S/octets의 경우 이메일 보호를 위해 확장 키 사용을 구성해야 합니다.

```
policyset.serverCertSet.7.default.params.exKeyUsageOIDs=1.3.6.1.5.5.7.3.4
```

5. 새 프로필을 가져옵니다.

```
$ ipa certprofile-import smime --file smime.cfg \
  --desc "S/MIME certificates" --store TRUE

-----------------------
Imported profile "smime"
-----------------------
  Profile ID: smime
  Profile description: S/MIME certificates
  Store issued certificates: TRUE
```

검증 단계

- 새 인증서 프로필을 가져왔는지 확인합니다.

```
$ ipa certprofile-find

-----------------
4 profiles matched
-----------------
  Profile ID: caIPAserviceCert
  Profile description: Standard profile for network services
  Store issued certificates: TRUE

  Profile ID: IECUserRoles
  Profile description: User profile that includes IECUserRoles extension from request
  Store issued certificates: TRUE

  Profile ID: KDCs_PKINIT_Certs
  Profile description: Profile for PKINIT support by KDCs
  Store issued certificates: TRUE

  Profile ID: smime
  Profile description: S/MIME certificates
  Store issued certificates: TRUE
----------------------------
Number of entries returned 4
----------------------------
```

추가 리소스

- **ipa help certprofile** 을 참조하십시오.
- RFC 5280, 섹션 4.2.1.12 를 참조하십시오.

5.3. CA 액세스 제어 목록이란 무엇입니까?

CA ACL(인증 기관 액세스 제어 목록) 규칙은 보안 주체의 인증서를 발급하는 데 사용할 수 있는 프로필을 정의합니다. 다음과 같이 **CA ACL**을 사용하여 이 작업을 수행할 수 있습니다.

- 특정 프로필이 있는 인증서를 발급할 수 있는 사용자, 호스트 또는 서비스를 결정합니다.
- 인증서를 발급할 수 있는 **IdM** 인증 기관 또는 하위 **CA** 확인

예를 들어, **CA ACL**을 사용하면 런던에 위치한 사무실에서 런던 사무실 관련 **IdM** 사용자 그룹의 멤버

인 사용자에게만 사용할 수 있는 프로필 사용을 제한할 수 있습니다.

CA ACL 규칙 관리를 위한 **ipa caacl** 유틸리티를 사용하면 권한 있는 사용자가 지정된 **CA ACL**을 추가, 표시, 수정 또는 삭제할 수 있습니다.

추가 리소스

- **ipa help caacl** 을 참조하십시오.

5.4. 인증서 프로필에 대한 액세스를 제어하기 위한 CA ACL 정의

이 절차에서는 **caacl** 유틸리티를 사용하여 그룹 내 사용자가 사용자 정의 인증서 프로필에 액세스할 수 있도록 **CA ACL**(액세스 제어 목록) 규칙을 정의하는 방법을 설명합니다. 이 경우 절차에서는 **S/octets** 사용자 그룹 및 **CA ACL**을 생성하여 해당 그룹의 사용자가 **smime** 인증서 프로필에 액세스할 수 있도록 하는 방법을 설명합니다.

사전 요구 사항

- **IdM** 관리자의 자격 증명을 취득했는지 확인합니다.

절차

1. 인증서 프로필의 사용자에 대한 새 그룹을 생성합니다.

```
$ ipa group-add smime_users_group
--------------------------------
Added group "smime users group"
--------------------------------
  Group name: smime_users_group
  GID: 75400001
```

2. **smime_user_group** 그룹에 추가할 새 사용자를 만듭니다.

```
$ ipa user-add smime_user
First name: smime
Last name: user
----------------------
Added user "smime_user"
----------------------
  User login: smime_user
  First name: smime
```

```
  Last name: user
  Full name: smime user
  Display name: smime user
  Initials: TU
  Home directory: /home/smime_user
  GECOS: smime user
  Login shell: /bin/sh
  Principal name: smime_user@IDM.EXAMPLE.COM
  Principal alias: smime_user@IDM.EXAMPLE.COM
  Email address: smime_user@idm.example.com
  UID: 1505000004
  GID: 1505000004
  Password: False
  Member of groups: ipausers
  Kerberos keys available: False
```

3. **smime_user** 를 **smime_users_group** 그룹에 추가합니다.

```
$ ipa group-add-member smime_users_group --users=smime_user
  Group name: smime_users_group
  GID: 1505000003
  Member users: smime_user
-------------------------
Number of members added 1
-------------------------
```

4. 그룹의 사용자가 인증서 프로필에 액세스할 수 있도록 **CA ACL**을 생성합니다.

```
$ ipa caacl-add smime_acl
-----------------------
Added CA ACL "smime_acl"
-----------------------
  ACL name: smime_acl
  Enabled: TRUE
```

5. 사용자 그룹을 **CA ACL**에 추가합니다.

```
$ ipa caacl-add-user smime_acl --group smime_users_group
  ACL name: smime_acl
  Enabled: TRUE
  User Groups: smime_users_group
-------------------------
Number of members added 1
-------------------------
```

6. **CA ACL**에 인증서 프로필을 추가합니다.

```
$ ipa caacl-add-profile smime_acl --certprofile smime
  ACL name: smime_acl
  Enabled: TRUE
  Profiles: smime
  User Groups: smime_users_group
------------------------
Number of members added 1
------------------------
```

검증 단계

- 생성한 CA ACL의 세부 정보를 확인합니다.

```
$ ipa caacl-show smime_acl
  ACL name: smime_acl
  Enabled: TRUE
  Profiles: smime
  User Groups: smime_users_group
...
```

추가 리소스

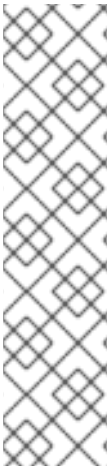
- ipa man 페이지를 참조하십시오.
- ipa help caacl 을 참조하십시오.

5.5. 인증서 프로필 및 CA ACL을 사용하여 인증서 발급

CA ACL(인증 기관 액세스 제어 목록)에서 허용하는 경우 인증서 프로필을 사용하여 인증서를 요청할 수 있습니다. 이 절차에서는 CA ACL을 통해 액세스 권한이 부여된 사용자 정의 인증서 프로필을 사용하여 사용자의 S/octets 인증서를 요청하는 방법을 설명합니다.

사전 요구 사항

- 인증서 프로필이 생성되었습니다.
- 사용자가 필수 인증서 프로필을 사용하여 인증서를 요청할 수 있는 CA ACL이 생성되었습니다.



참고

cert-request 명령을 수행하는 사용자가 CA ACL 검사를 바이패스할 수 있습니다.

- admin 사용자입니다.
- CA ACL 권한을 무시하는 요청 인증서가 있습니다.

절차

1. 사용자에 대한 인증서 요청을 생성합니다. 예를 들어 OpenSSL을 사용합니다.

```
$ openssl req -new -newkey rsa:2048 -days 365 -nodes -keyout private.key -out
cert.csr -subj '/CN=smime_user'
```

2. IdM CA에서 사용자에 대한 새 인증서를 요청합니다.

```
$ ipa cert-request cert.csr --principal=smime_user --profile-id=smime
```

선택적으로 루트 CA 대신 하위 CA에서 인증서를 요청하기 위해 명령에 --ca *sub-CA_name* 옵션을 전달합니다.

검증 단계

- 새로 발급된 인증서가 사용자에게 할당되었는지 확인합니다.

```
$ ipa user-show user
 User login: user
 ...
 Certificate: MIICfzCCAWcCAQA...
 ...
```

추가 리소스

- ipa(a) 도움말 페이지를 참조하십시오.
- ipa help user-show 명령을 참조하십시오.

- **ipa help cert-request** 명령을 참조하십시오.

- **openssl(1ssl)** 도움말 페이지를 참조하십시오.

5.6. 인증서 프로필 수정

이 절차에서는 **ipa certprofile-mod** 명령을 사용하여 명령행을 통해 인증서 프로파일을 직접 수정하는 방법을 설명합니다.

절차

1. 수정하려는 인증서 프로필의 인증서 프로필 **ID**를 확인합니다. 현재 **IdM**에 저장된 모든 인증서 프로필을 표시하려면 다음을 수행하십시오.

```
# ipa certprofile-find

-----------------
4 profiles matched
-----------------
  Profile ID: caIPAserviceCert
  Profile description: Standard profile for network services
  Store issued certificates: TRUE

  Profile ID: IECUserRoles
  ...

  Profile ID: smime
  Profile description: S/MIME certificates
  Store issued certificates: TRUE
-------------------------
Number of entries returned
-------------------------
```

2. 인증서 프로필 설명을 수정합니다. 예를 들어 기존 프로필을 사용하여 **S/hiera** 인증서에 대한 사용자 정의 인증서 프로필을 생성한 경우 새 사용법에 따라 설명을 변경합니다.

```
# ipa certprofile-mod smime --desc "New certificate profile description"
-----------------------------------
Modified Certificate Profile "smime"
-----------------------------------
   Profile ID: smime
   Profile description: New certificate profile description
   Store issued certificates: TRUE
```

3. 텍스트 편집기에서 고객 인증서 프로필 파일을 열고 요구 사항에 맞게 수정합니다.

```
# vi smime.cfg
```

인증서 프로필 구성 파일에서 구성할 수 있는 옵션에 대한 자세한 내용은 인증서 프로필 구성 매개 변수를 참조하십시오.

4. 기존 인증서 프로필 구성 파일을 업데이트합니다.

```
# ipa certprofile-mod _profile_ID_ --file=smime.cfg
```

검증 단계

- 인증서 프로필이 업데이트되었는지 확인합니다.

```
$ ipa certprofile-show smime
  Profile ID: smime
  Profile description: New certificate profile description
  Store issued certificates: TRUE
```

추가 리소스

- ipa(a) 도움말 페이지를 참조하십시오.
- ipa help certprofile-mod 를 참조하십시오.

5.7. 인증서 프로필 구성 매개변수

인증서 프로필 구성 매개변수는 CA 프로필 디렉터리의 *profile_name*.cfg 파일에 저장됩니다. /var/lib/pki/pki-tomcat/ca/profiles/ca. 프로필의 모든 매개변수(기본값, 입력, 출력 및 제약 조건)는 단일 정책 세트 내에서 구성됩니다. 인증서 프로필에 설정된 정책에는 name policyset.*policyName.policyNumber* 가 있습니다. 예를 들어 정책 set serverCertSet 은 다음과 같습니다.

```
policyset.list=serverCertSet
policyset.serverCertSet.list=1,2,3,4,5,6,7,8
policyset.serverCertSet.1.constraint.class_id=subjectNameConstraintImpl
policyset.serverCertSet.1.constraint.name=Subject Name Constraint
policyset.serverCertSet.1.constraint.params.pattern=CN=[^,]+,.+
policyset.serverCertSet.1.constraint.params.accept=true
```

```
policyset.serverCertSet.1.default.class_id=subjectNameDefaultImpl
policyset.serverCertSet.1.default.name=Subject Name Default
policyset.serverCertSet.1.default.params.name=CN=$request.req_subject_name.cn$, OU=pki-ipa, O=IPA
policyset.serverCertSet.2.constraint.class_id=validityConstraintImpl
policyset.serverCertSet.2.constraint.name=Validity Constraint
policyset.serverCertSet.2.constraint.params.range=740
policyset.serverCertSet.2.constraint.params.notBeforeCheck=false
policyset.serverCertSet.2.constraint.params.notAfterCheck=false
policyset.serverCertSet.2.default.class_id=validityDefaultImpl
policyset.serverCertSet.2.default.name=Validity Default
policyset.serverCertSet.2.default.params.range=731
policyset.serverCertSet.2.default.params.startTime=0
```

각 정책 세트에는 평가해야 하는 순서대로 정책 ID 번호별로 인증서 프로필에 대해 구성된 정책 목록이 포함되어 있습니다. 서버는 수신하는 각 요청에 대해 각 정책 세트를 평가합니다. 단일 인증서 요청이 수신되면 하나의 집합이 평가되고 프로필의 다른 세트가 무시됩니다. 이중 키 쌍이 발행되면 첫 번째 정책 세트가 첫 번째 인증서 요청에 대해 평가되고 두 번째 세트는 두 번째 인증서 요청에 대해 평가됩니다. 듀얼 키 쌍을 발행할 때 단일 인증서 또는 두 개 이상의 세트를 발행할 때 두 개 이상의 정책 세트가 필요하지 않습니다.

표 5.1. 인증서 프로필 구성 파일 매개변수

매개변수	설명
desc	인증서 프로필에 대한 무료 텍스트 설명(end-entities) 페이지에 표시됩니다. 예를 들어 **desc= 이 인증서 프로필은 에이전트 인증에 서버 인증서를 등록하는** 것입니다.
enable	엔드 포인트 페이지를 통해 액세스할 수 있도록 프로필을 활성화합니다. 예를 들면 **enable=true** 입니다.
auth.instance_id	인증서 요청을 인증하는 데 사용할 인증 관리자 플러그인을 설정합니다. 자동 등록의 경우 인증에 성공하면 CA에서 즉시 인증서를 발행합니다. 인증이 실패하거나 인증 플러그인이 지정되지 않은 경우 요청이 에이전트에서 수동으로 승인하도록 큐에 큐에 추가됩니다. 예를 들면 **auth.instance_id=AgentCertAuth** 입니다.

매개변수	설명
authz.acl	권한 부여 제약 조건을 지정합니다. 이 기능은 ACL(그룹 평가 액세스 제어 목록)을 설정하는 데 주로 사용됩니다. 예를 들어 **caCMCUserCert** 매개변수를 사용하려면 CMC 요청의 서명자가 Certificate Manager 에이전트 그룹에 속해야 합니다. **authz.acl=group="Certificate Manager 에이전트** 디렉터리 기반 사용자 인증서 업데이트에서 이 옵션은 원래 요청자 및 현재 인증 된 사용자가 동일한지 확인하는 데 사용됩니다. 권한 부여를 평가하기 전에 엔터티에서 인증(바인딩 또는, 기본적으로 시스템에 로그인)해야 합니다.
name	인증서 프로필의 이름입니다. 예를 들어 **name=Agent-Authenticated Server Certificate Enrollment**. 이 이름은 최종 사용자 등록 또는 업데이트 페이지에 표시됩니다.
input.list	이름별로 인증서 프로필에 허용된 입력을 나열합니다. 예를 들면 **input.list=i1,i2** 가 있습니다.
input.input_id.class_id	입력 ID로 입력의 java 클래스 이름을 나타냅니다(input.list에 나열된 입력의 이름). 예를 들어 **input.i1.class_id=certReqInputImpl**.
output.list	인증서 프로필에 사용 가능한 출력 형식을 이름으로 나열합니다. 예를 들면 **output.list=o1** 이 있습니다.
output.output_id.class_id	output.list에 이름이 지정된 출력 형식의 java 클래스 이름을 지정합니다. 예를 들어 **output.o1.class_id=certOutputImpl**.
policyset.list	구성된 인증서 프로필 규칙을 나열합니다. 이중 인증서의 경우 하나의 규칙 세트가 서명 키에 적용되며 다른 규칙 세트가 암호화 키에 적용됩니다. 단일 인증서는 하나의 인증서 프로필 규칙 세트만 사용합니다. 예를 들어 **policyset.list=serverCertSet** 입니다.
policyset.policyset_id.list	평가해야 하는 순서에 따라 인증서 프로필에 대해 구성된 정책 프로필 내의 정책을 정책 ID 번호로 나열합니다. 예를 들어 **policyset.serverCertSet.list=1,2,3,4,5,6,7,8** 입니다.

매개변수	설명
policyset.policyset_id.policy_number.constraint.class_id	프로필 규칙에 구성된 기본값에 대해 제약 조건 플러그인 세트의 java 클래스 이름을 나타냅니다. 예를 들어 policyset.serverCertSet.1.constraint.class_id=subjectNameConstraintImpl입니다.
policyset.policyset_id.policy_number.constraint.name	제약 조건의 사용자 정의 이름을 제공합니다. 예를 들어 policyset.serverCertSet.1.constraint.name=Subject Name Constraint입니다.
policyset.policyset_id.policy_number.constraint.params.attribute	제약 조건에 대해 허용되는 속성에 대한 값을 지정합니다. 사용 가능한 속성은 제약 조건 유형에 따라 달라집니다. 예를 들어 policyset.serverCertSet.1.octets.params.octets=CN=.*입니다.
policyset.policyset_id.policy_number.default.class_id	프로필 규칙에 기본 세트에 대한 java 클래스 이름을 제공합니다. For example, policyset.serverCertSet.1.default.class_id=userSubjectNameDefaultImpl
policyset.policyset_id.policy_number.default.name	기본값의 사용자 정의 이름을 지정합니다. 예를 들어 policyset.serverCertSet.1.default.name=Subject Name Default
policyset.policyset_id.policy_number.default.params.attribute	기본값에 대해 허용된 속성의 값을 지정합니다. 가능한 속성은 기본값 유형에 따라 다릅니다. For example, policyset.serverCertSet.1.default.params.name=CN=(Name)$request.requestor_name$.

6장. IDM에서 인증서의 유효성 관리

IdM(Identity Management)에서는 향후 발급하려는 기존 인증서와 인증서의 유효성을 모두 관리할 수 있지만 방법은 다릅니다.

6.1. IDM CA에서 발급한 기존 인증서의 유효성 관리

IdM에서 인증서 만료 날짜를 확인하는 다음 방법을 사용할 수 있습니다.

- IdM WebUI의 만료 날짜 보기.
- CLI에서 만료 날짜 보기.

다음과 같은 방법으로 IdM CA에서 발급한 기존 인증서의 유효성을 관리할 수 있습니다.

- 원래 CSR(인증서 서명 요청) 또는 개인 키에서 생성된 새 CSR을 사용하여 새 인증서를 요청하여 인증서를 갱신합니다. 다음 유틸리티를 사용하여 새 인증서를 요청할 수 있습니다.

  certmonger

  certmonger 를 사용하여 서비스 인증서를 요청할 수 있습니다. 인증서가 만료되기 전에 certmonger 는 인증서를 자동으로 갱신하여 서비스 인증서의 유효성을 계속 확인합니다. 자세한 내용은 certmonger를 사용하여 서비스에 대한 IdM 인증서 가져오기를 참조하십시오.

  certutil

  certutil 을 사용하여 사용자, 호스트 및 서비스 인증서를 갱신할 수 있습니다. 사용자 인증서 요청에 대한 자세한 내용은 새 사용자 인증서 요청 및 클라이언트로 내보내기를 참조하십시오.

  OpenSSL

  openssl 을 사용하여 사용자, 호스트 및 서비스 인증서를 갱신할 수 있습니다.

- 인증서를 취소합니다. 자세한 내용은 다음을 참조하십시오.
  - IdM WebUI를 사용하여 통합 IdM CA로 인증서 해지

- IdM CLI를 사용하여 통합 IdM CA로 인증서 해지

- 인증서가 일시적으로 해지된 경우 복원하십시오. 자세한 내용은 다음을 참조하십시오.
  - IdM WebUI를 사용하여 통합 IdM CA로 인증서 복원
  - IdM CLI를 사용하여 통합 IdM CA로 인증서 복원.

6.2. IDM CA에서 발급한 향후 인증서의 유효성 관리

IdM CA에서 발급한 향후 인증서의 유효성을 관리하려면 인증서 프로필을 수정, 가져오기 또는 생성합니다. 자세한 내용은 Identity Management에서 인증서 프로필 생성 및 관리를 참조하십시오.

6.3. IDM WEBUI에서 인증서 만료일 보기

IdM WebUI를 사용하여 IdM CA에서 발행한 모든 인증서의 만료 날짜를 확인할 수 있습니다.

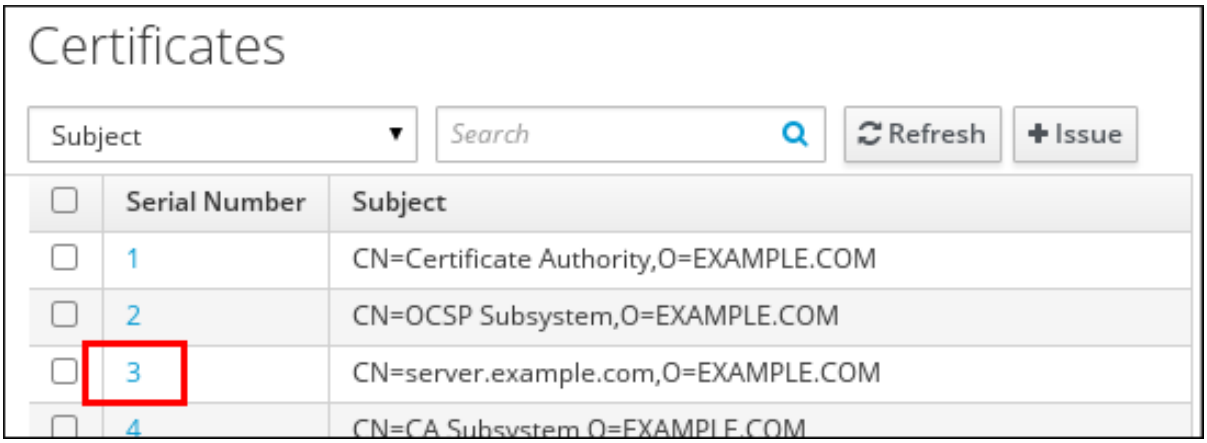
사전 요구 사항

- 관리자의 자격 증명을 확보했는지 확인합니다.

절차

1. 인증 메뉴에서 인증서 > 인증서를 클릭합니다.
2. 인증서의 일련 번호를 클릭하여 인증서 정보 페이지를 엽니다.

그림 6.1. 인증서 목록



3. 인증서 정보 페이지에서 **Expires On** 정보를 찾습니다.

6.4. CLI에서 인증서 만료일 보기

CLI(명령줄 인터페이스)를 사용하여 인증서 만료 날짜를 확인할 수 있습니다.

절차

- **openssl** 유틸리티를 사용하여 사람이 읽을 수 있는 형식으로 파일을 엽니다.

```
$ openssl x509 -noout -text -in ca.pem
Certificate:
    Data:
        Version: 3 (0x2)
        Serial Number: 1 (0x1)
        Signature Algorithm: sha256WithRSAEncryption
        Issuer: O = IDM.EXAMPLE.COM, CN = Certificate Authority
        Validity
            Not Before: Oct 30 19:39:14 2017 GMT
            Not After : Oct 30 19:39:14 2037 GMT
```

6.5. 통합된 IDM CA를 사용하여 인증서 해지

6.5.1. 인증서 해지 이유

해지된 인증서는 유효하지 않으며 인증에 사용할 수 없습니다. 6을 제외한 모든 취소는 영구적입니다. 인증서 보유.

기본 해지 이유는 0: 지정되지 않은 것입니다.

표 6.1. 해지 이유

ID	이유	설명
0	지정되지 않음	
1	키 완료	인증서를 발급한 키는 더 이상 신뢰할 수 없습니다. 가능한 원인: 토큰 손실, 부적절하게 액세스 가능한 파일.
2	ca Compromised	인증서를 발급한 CA는 더 이상 신뢰되지 않습니다.
3	영향을 받는 기능 변경	가능한 원인은 다음과 같습니다. * 한 사람이 퇴사했거나 다른 부서로 이동했습니다. * 호스트 또는 서비스가 폐지되고 있습니다.
4	superseded	최신 인증서가 현재 인증서를 대체했습니다.
5	작업 종료	호스트 또는 서비스가 해제되고 있습니다.
6	인증서 보유	인증서가 일시적으로 취소되었습니다. 나중에 인증서를 복원할 수 있습니다.
8	CRL에서 제거	인증서 해지 목록(CRL)에 포함되어 있지 않습니다.
9	권한 삭제	사용자, 호스트 또는 서비스가 더 이상 인증서를 사용할 수 없습니다.
10	Property Authority (AA) Compromise	AA 인증서는 더 이상 신뢰되지 않습니다.

6.5.2. IdM WebUI를 사용하여 통합 IdM CA로 인증서 해지

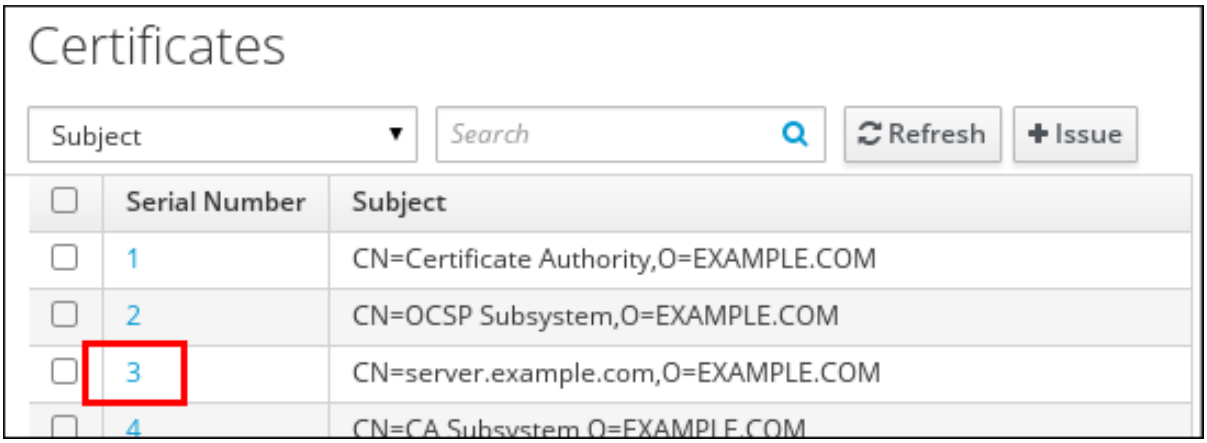
인증서의 개인 키를 손실했음을 알고 있는 경우 인증서를 해지하여 악용을 방지해야 합니다. IdM WebUI를 사용하여 IdM CA에서 발급한 인증서를 취소하려면 이 절차를 완료합니다.

절차

1. **인증 > 인증서 > 인증서 를 클릭합니다.**

2. **인증서의 일련 번호를 클릭하여 인증서 정보 페이지를 엽니다.**

그림 6.2. 인증서 목록



3. 인증서 정보 페이지에서 **Actions** → **Revoke Certificate** 을 클릭합니다.

4. 취소 이유를 선택하고 **Revoke** 를 클릭합니다. 자세한 내용은 인증서 취소 이유를 참조하십시오.

6.5.3. IdM CLI를 사용하여 통합 IdM CA로 인증서 해지

인증서의 개인 키를 손실했음을 알고 있는 경우 인증서를 해지하여 악용을 방지해야 합니다. **IdM CLI** 를 사용하여 **IdM CA**에서 발급한 인증서를 취소하려면 이 절차를 완료합니다.

절차

- **ipa cert-revoke** 명령을 사용하여 다음을 지정합니다.
  - 인증서 일련 번호
  - 취소 사유의 **ID** 번호; 자세한 내용은 인증서 취소 사유 를 참조하십시오.

예를 들어 이유 **1**로 인해 일련 번호가 **1032** 인 인증서를 취소하려면 다음을 수행합니다. 키 완료 .를 입력합니다.

```
$ ipa cert-revoke 1032 --revocation-reason=1
```

새 인증서를 요청하는 방법에 대한 자세한 내용은 다음 설명서를 참조하십시오.

- 새 사용자 인증서를 요청하고 클라이언트로 내보낼 수 있습니다.
- certmonger를 사용하여 서비스에 대한 IdM 인증서 가져오기.

6.6. 통합된 IDM CA로 인증서 복원

6 이유로 인해 인증서를 취소한 경우: 인증서 보유, 인증서의 개인 키가 손상되지 않은 경우 다시 복원할 수 있습니다. 인증서를 복원하려면 다음 절차 중 하나를 사용합니다.

- IdM WebUI를 사용하여 통합 IdM CA로 인증서를 복원
- IdM CLI를 사용하여 통합 IdM CA로 인증서를 복원합니다.

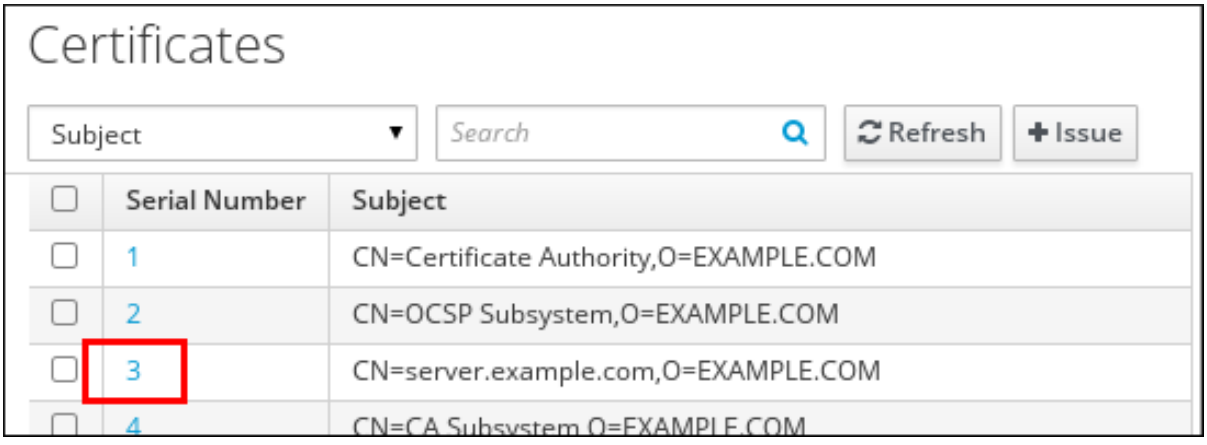
6.6.1. IdM WebUI를 사용하여 통합 IdM CA로 인증서 복원

IdM WebUI를 사용하여 이유 6로 인해 해지된 IdM 인증서를 복원하려면 이 절차를 완료합니다. 인증서 보유.

절차

1. 인증 메뉴에서 인증서 > 인증서를 클릭합니다.
2. 인증서의 일련 번호를 클릭하여 인증서 정보 페이지를 엽니다.

그림 6.3. 인증서 목록



3. 인증서 정보 페이지에서 인증서복원을 클릭합니다.

6.6.2. IdM CLI를 사용하여 통합 IdM CA로 인증서 복원

IdM CLI를 사용하여 이유 6로 인해 해지된 IdM 인증서를 복원하려면 이 절차를 완료합니다. 인증서 보유.

절차

- ipa cert-remove-hold 명령을 사용하고 인증서 일련 번호를 지정합니다. 예를 들면 다음과 같습니다.

```
$ ipa cert-remove-hold 1032
```

7장. 스마트 카드 인증에 대한 ID 관리 구성

스마트 카드를 기반으로 하는 인증은 암호 대신 사용됩니다. 스마트 카드에 사용자 자격 증명을 개인 키 및 인증서 형식으로 저장할 수 있으며 특수 소프트웨어 및 하드웨어는 이를 액세스하는 데 사용됩니다. 스마트 카드를 리더 또는 USB 포트에 배치하고 비밀번호를 제공하는 대신 스마트 카드의 PIN 코드를 제공합니다.

IdM(Identity Management)은 다음을 사용하여 스마트 카드 인증을 지원합니다.

- IdM 인증 기관에서 발급한 사용자 인증서
- 외부 인증 기관에서 발급한 사용자 인증서

이 사용자 스토리는 두 가지 인증서 모두에 대해 IdM에서 스마트 카드 인증을 설정하는 방법을 보여줍니다. 사용자 스토리에서 smartcard_ca.pem CA 인증서는 신뢰할 수 있는 외부 인증 기관의 인증서가 포함된 파일입니다.

사용자 스토리에는 다음 모듈이 포함되어 있습니다.

- 스마트 카드 인증을 위한 IdM 서버 구성
- 스마트 카드 인증을 위한 IdM 클라이언트 구성
- IdM 웹 UI의 사용자 항목에 인증서 추가
- IdM CLI의 사용자 항목에 인증서 추가
- 스마트 카드를 관리하고 사용하기 위한 도구 설치
- 스마트 카드에 인증서 저장

- 스마트 카드를 사용하여 IdM에 로그인
- 스마트 카드 인증을 사용하여 GDM 액세스 구성
- 스마트 카드 인증을 사용하여 su 액세스 구성

7.1. 스마트 카드 인증을 위한 IDM 서버 구성

EXAMPLE.ORG 도메인의 인증서 기관에서 인증서가 발급한 사용자에 대해 스마트 카드 인증을 활성화하려면, LDAP 고유 이름(DN)이 CN=Certificate Authority,DC=EXAMPLE,DC=ORG 이므로 IdM 서버를 구성하는 스크립트로 실행할 수 있도록 기관의 인증서를 가져와야 합니다. 예를 들어 인증 기관에서 인증서를 발급한 웹 페이지에서 인증서를 다운로드할 수 있습니다. 자세한 내용은 인증서 인증을 사용하도록 브라우저 구성 의 1 - 4a 단계를 참조하십시오.

IdM 인증 기관에서 인증서를 발급한 IdM 사용자의 스마트 카드 인증을 활성화하려면 IdM CA가 실행되고 있는 IdM 서버의 /etc/ipa/ca.crt 파일에서 CA 인증서를 받으십시오.

이 섹션에서는 스마트 카드 인증을 위해 IdM 서버를 구성하는 방법을 설명합니다. 먼저 PEM 형식의 CA 인증서를 사용하여 파일을 가져온 다음 빌드된 ipa-advise 스크립트를 실행합니다. 마지막으로 시스템 구성을 다시 로드합니다.

사전 요구 사항

- IdM 서버에 대한 루트 액세스 권한이 있습니다.
- 루트 CA 인증서 및 하위 CA 인증서가 있습니다.

절차

1. 구성을 수행할 디렉터리를 생성합니다.

```
[root@server]# mkdir ~/SmartCard/
```

2. 디렉터리로 이동합니다.

```
[root@server]# cd ~/SmartCard/
```

3. **PEM** 형식의 파일에 저장된 관련 **CA** 인증서를 가져옵니다. **CA** 인증서가 **DER**와 같은 다른 형식의 파일에 저장된 경우 **PEM** 형식으로 변환합니다. **IdM** 인증 기관 인증서는 **/etc/ipa/ca.crt** 파일에 있습니다.

   DER 파일을 **PEM** 파일로 변환합니다.

```
# openssl x509 -in <filename>.der -inform DER -out <filename>.pem -outform PEM
```

4. 편의를 위해 구성을 수행하려는 디렉터리에 인증서를 복사합니다.

```
[root@server SmartCard]# cp /etc/ipa/ca.crt ~/SmartCard/
[root@server SmartCard]# cp /tmp/smartcard_ca.pem ~/SmartCard/
```

5. 선택적으로 외부 인증 기관의 인증서를 사용하는 경우 **openssl x509** 유틸리티를 사용하여 **Issuer** 및 **Subject** 값이 올바른지 확인하려면 **PEM** 형식의 파일 내용을 확인합니다.

```
[root@server SmartCard]# openssl x509 -noout -text -in smartcard_ca.pem | more
```

6. 관리자 권한으로 내장 **ipa-advise** 유틸리티를 사용하여 구성 스크립트를 생성합니다.

```
[root@server SmartCard]# kinit admin
[root@server SmartCard]# ipa-advise config-server-for-smart-card-auth > config-server-for-smart-card-auth.sh
```

   config-server-for-smart-card-auth.sh 스크립트는 다음 작업을 수행합니다.

   - **IdM Apache HTTP** 서버를 구성합니다.
   - **KDC (Key Distribution Center)**에서 **Kerberos (PKINIT)**에서 초기 인증의 공개 키 암호화를 활성화합니다.
   - 스마트 카드 권한 부여 요청을 수락하도록 **IdM** 웹 **UI**를 구성합니다.

7. 스크립트를 실행하고 루트 CA 및 하위 CA 인증서가 포함된 PEM 파일을 인수로 추가합니다.

```
[root@server SmartCard]# chmod +x config-server-for-smart-card-auth.sh
[root@server SmartCard]# ./config-server-for-smart-card-auth.sh smartcard_ca.pem
ca.crt
Ticket cache:KEYRING:persistent:0:0
Default principal: admin@IDM.EXAMPLE.COM
[...]
Systemwide CA database updated.
The ipa-certupdate command was successful
```

참고

하위 CA 인증서보다 먼저 루트 CA의 인증서를 인수로 추가하고 CA 또는 하위 CA 인증서가 만료되지 않았는지 확인합니다.

8. 선택적으로 사용자 인증서를 발급한 인증 기관이 OCSP 검사를 IdM 웹 UI에 대한 인증을 비활성화해야 할 수 있습니다.

a. /etc/httpd/conf.d/ssl.conf 파일에서 SSLOCSPEnable 매개변수를 off 로 설정합니다.

```
SSLOCSPEnable off
```

b. 변경 사항을 즉시 적용하려면 Apache 데몬(httpd)을 다시 시작합니다.

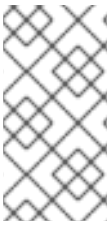
```
[root@server SmartCard]# systemctl restart httpd
```

주의

IdM CA에서 발급한 사용자 인증서만 사용하는 경우 OCSP 검사를 비활성화하지 마십시오. OCSP 응답자는 IdM의 일부입니다.

OCSP 검사를 계속 활성화하는 방법에 대한 지침은 Apache mod_ssl 구성 옵션에서 SSLOCSPDefaultResponder 지시문을 참조하십시오.

이제 서버가 스마트 카드 인증용으로 구성되어 있습니다.



참고

전체 토폴로지에서 스마트 카드 인증을 활성화하려면 각 IdM 서버에서 절차를 실행합니다.

7.2. 스마트 카드 인증을 위한 IDM 클라이언트 구성

이 섹션에서는 스마트 카드 인증을 위해 IdM 클라이언트를 구성하는 방법을 설명합니다. 절차는 인증에 스마트 카드를 사용하는 동안 연결하려는 각 IdM 시스템, 클라이언트 또는 서버에서 실행해야 합니다. 예를 들어 호스트 A에서 호스트 B로 ssh 연결을 활성화하려면 호스트 B에서 스크립트를 실행해야 합니다.

관리자는 다음을 사용하여 스마트 카드 인증을 사용하도록 이 절차를 실행하십시오.

- ssh 프로토콜

  자세한 내용은 스마트 카드 인증을 사용하여 SSH 액세스 구성을 참조하십시오.

- 콘솔 로그인
- Gnome Display Manager (GDM)
- su 명령

IdM 웹 UI로 인증하는 데는 이 절차가 필요하지 않습니다. IdM 웹 UI에 인증하려면 두 개의 호스트가 필요합니다. 이 호스트 중 어느 것도 IdM 클라이언트여야 합니다.

- 시스템 - 브라우저를 실행 중인 IdM 도메인 외부일 수 있습니다.
- httpd 가 실행 중인 IdM 서버

다음 절차에서는 IdM 서버가 아닌 IdM 클라이언트에서 스마트 카드 인증을 구성하는 것으로 가정합니다. 따라서 구성 스크립트를 생성하는 IdM 서버와 스크립트를 실행하는 IdM 클라이언트라는 두 개의 컴퓨터가 필요합니다.

사전 요구 사항

- 스마트 카드 인증을 위해 IdM 서버 구성에 설명된 대로 IdM 서버가 스마트 카드 인증을 위해 구성되어 있습니다.
- IdM 서버와 IdM 클라이언트에 대한 루트 액세스 권한이 있습니다.
- 루트 CA 인증서 및 모든 하위 CA 인증서에 액세스할 수 있습니다.
- 원격 사용자가 성공적으로 로그인할 수 있도록 --mkhomedir 옵션으로 IdM 클라이언트를 설치했습니다. 홈 디렉터리를 생성하지 않으면 기본 로그인 위치는 root입니다.

절차

1. IdM 서버에서 관리자 권한을 사용하여 ipa-advise 를 사용하여 구성 스크립트를 생성합니다.

```
[root@server SmartCard]# kinit admin
[root@server SmartCard]# ipa-advise config-client-for-smart-card-auth > config-client-for-smart-card-auth.sh
```

config-client-for-smart-card-auth.sh 스크립트는 다음 작업을 수행합니다.

- 스마트 카드 데몬을 구성합니다.
- 전체 시스템 신뢰 저장소를 설정합니다.
- 데스크탑에 스마트 카드 로그인을 허용하도록 SSSD(System Security Services Daemon)를 구성합니다.

2. IdM 서버에서 IdM 클라이언트 시스템에서 선택한 디렉터리에 스크립트를 복사합니다.

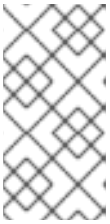
```
[root@server SmartCard]# scp config-client-for-smart-card-auth.sh
root@client.idm.example.com:/root/SmartCard/
Password:
config-client-for-smart-card-auth.sh       100%   2419       3.5MB/s   00:00
```

3. IdM 서버에서 이전 단계에서 사용한 것처럼 PEM 형식의 CA 인증서 파일을 IdM 클라이언트 시스템의 동일한 디렉터리에 복사합니다.

```
[root@server SmartCard]# scp {smartcard_ca.pem,ca.crt}
root@client.idm.example.com:/root/SmartCard/
Password:
smartcard_ca.pem              100%   1237     9.6KB/s   00:00
ca.crt                        100%   2514    19.6KB/s   00:00
```

4. 클라이언트 시스템에서 스크립트를 실행하고 CA 인증서가 포함된 PEM 파일을 인수로 추가합니다.

```
[root@client SmartCard]# kinit admin
[root@client SmartCard]# chmod +x config-client-for-smart-card-auth.sh
[root@client SmartCard]# ./config-client-for-smart-card-auth.sh smartcard_ca.pem
ca.crt
Ticket cache:KEYRING:persistent:0:0
Default principal: admin@IDM.EXAMPLE.COM
[...]
Systemwide CA database updated.
The ipa-certupdate command was successful
```



참고

하위 CA 인증서보다 먼저 루트 CA의 인증서를 인수로 추가하고 CA 또는 하위 CA 인증서가 만료되지 않았는지 확인합니다.

이제 클라이언트가 스마트 카드 인증용으로 구성되어 있습니다.

7.3. IDM 웹 UI의 사용자 항목에 인증서 추가

이 절차에서는 IdM 웹 UI의 사용자 항목에 외부 인증서를 추가하는 방법을 설명합니다.

전체 인증서를 업로드하는 대신 IdM의 사용자 항목에 인증서 매핑 데이터를 업로드할 수도 있습니다. 전체 인증서 또는 인증서 매핑 데이터를 포함하는 사용자 항목을 해당 인증서 매핑 규칙과 함께 사용하여 시스템 관리자를 위한 스마트 카드 인증을 쉽게 구성할 수 있습니다. 자세한 내용은 참조하십시오.

스마트 카드에서 인증을 구성하기 위한 인증서 매핑 규칙 입니다.

참고

IdM 인증 기관에서 사용자 인증서를 발급한 경우 인증서가 사용자 항목에 이미 저장되어 있으며 이 섹션을 건너뛸 수 있습니다.

사전 요구 사항

- 사용 중인 사용자 항목에 추가할 인증서가 있습니다.

절차

1. 다른 사용자에게 인증서를 추가하려면 IdM 웹 UI에 관리자로 로그인합니다. 자체 프로필에 인증서를 추가하려면 관리자의 자격 증명이 필요하지 않습니다.

2. 사용자 → 활성 사용자 → sc_user 로 이동합니다.

3. 인증서 옵션을 찾아 추가 를 클릭합니다.

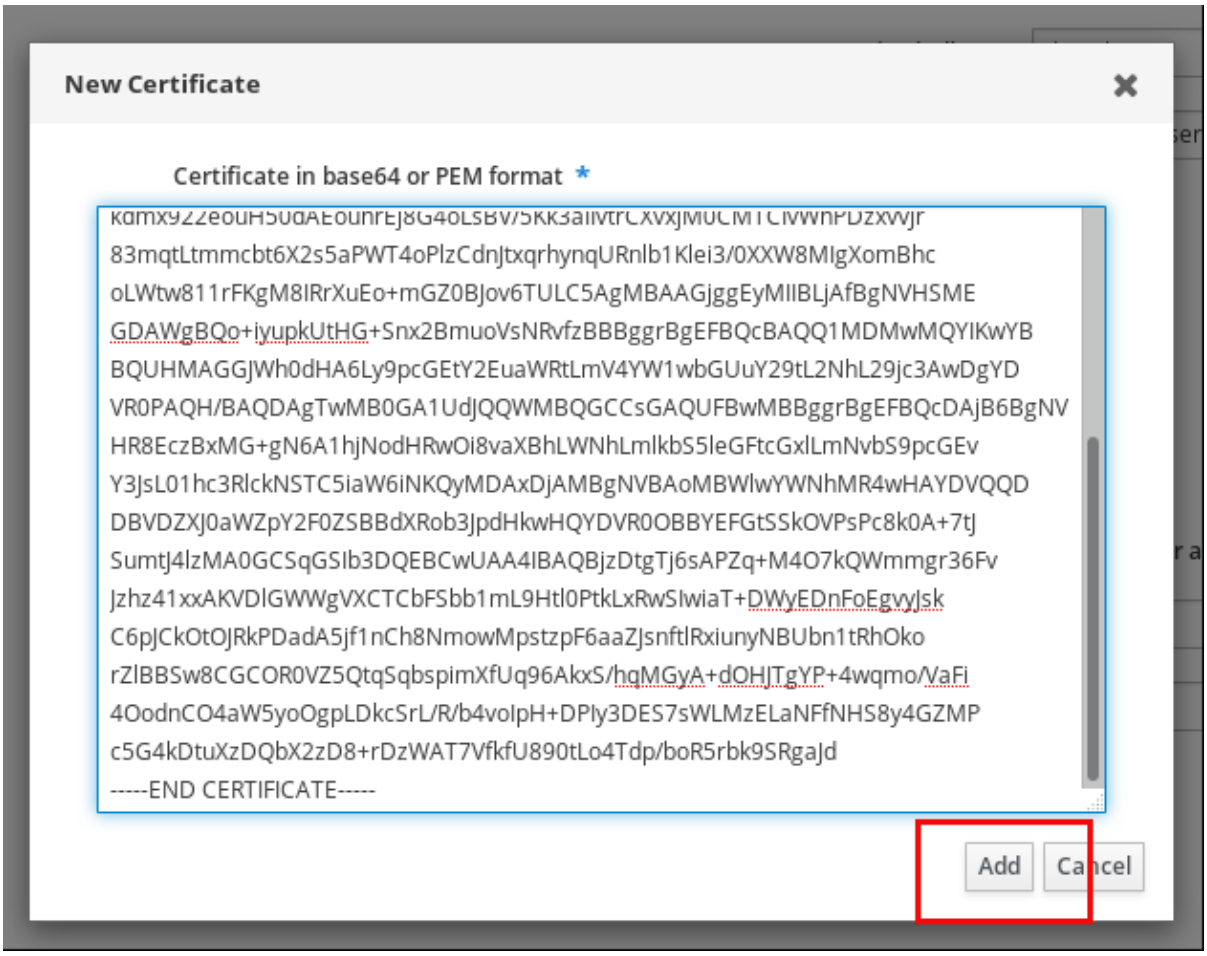
4. 명령줄 인터페이스의 cat 유틸리티 또는 텍스트 편집기를 사용하여 PEM 형식으로 인증서를 표시합니다.

```
[user@client SmartCard]$ cat testuser.crt
```

5. CLI에서 인증서를 복사하여 웹 UI에서 연 창에 붙여넣습니다.

6. 추가를 클릭합니다.

그림 7.1. IdM 웹 UI에 새 인증서 추가



이제 **sc_user** 항목에 외부 인증서가 포함되어 있습니다.

7.4. IDM CLI의 사용자 항목에 인증서 추가

이 절차에서는 **IdM CLI**의 사용자 항목에 외부 인증서를 추가하는 방법을 설명합니다.

전체 인증서를 업로드하는 대신 **IdM**의 사용자 항목에 인증서 매핑 데이터를 업로드할 수도 있습니다. 전체 인증서 또는 인증서 매핑 데이터를 포함하는 사용자 항목을 해당 인증서 매핑 규칙과 함께 사용하여 시스템 관리자를 위한 스마트 카드 인증을 쉽게 구성할 수 있습니다. 자세한 내용은 참조하십시오.

스마트 카드에서 인증을 구성하기 위한 인증서 매핑 규칙 입니다.

참고

IdM 인증 기관에서 사용자 인증서를 발급한 경우 인증서가 사용자 항목에 이미 저장되어 있으며 이 섹션을 건너뛸 수 있습니다.

사전 요구 사항

- 사용 중인 사용자 항목에 추가할 인증서가 있습니다.

절차

1. 다른 사용자에게 인증서를 추가하려면 IdM CLI에 관리자로 로그인합니다.

```
[user@client SmartCard]$ kinit admin
```

자체 프로필에 인증서를 추가하려면 관리자의 자격 증명이 필요하지 않습니다.

```
[user@client SmartCard]$ kinit sc_user
```

2. header 및 footer가 제거된 인증서가 포함된 환경 변수를 생성하고 단일 행에 연결합니다. ipa user-add-cert 명령에서 예상되는 형식입니다.

```
[user@client SmartCard]$ export CERT=`openssl x509 -outform der -in testuser.crt |
base64 -w0 -`
```

testuser.crt 파일의 인증서는 PEM 형식이어야 합니다.

3. ipa user-add-cert 명령을 사용하여 sc_user 프로필에 인증서를 추가합니다.

```
[user@client SmartCard]$ ipa user-add-cert sc_user --certificate=$CERT
```

이제 sc_user 항목에 외부 인증서가 포함되어 있습니다.

7.5. 스마트 카드를 관리하고 사용하기 위한 도구 설치

스마트 카드를 구성하려면 인증서를 생성하고 스마트 카드에 저장할 수 있는 도구가 필요합니다.

다음 작업을 수행해야 합니다.

- 인증서를 관리하는 데 도움이 되는 **gnutls-utils** 패키지를 설치합니다.
- 스마트 카드로 작동하도록 라이브러리 및 유틸리티 세트를 제공하는 **opensc** 패키지를 설치합니다.
- 스마트 카드 리더와 통신하는 **pcscd** 서비스를 시작합니다.

절차

1. **opensc** 및 **gnutls-utils** 패키지를 설치합니다.

```
# dnf -y install opensc gnutls-utils
```

2. **pcscd** 서비스를 시작합니다.

```
# systemctl start pcscd
```

pcscd 서비스가 실행 중인지 확인합니다.

7.6. 스마트 카드에 인증서 저장

이 섹션에서는 구성하는 데 도움이 되는 **pkcs15-init** 도구를 사용한 스마트 카드 구성에 대해 설명합니다.

- 스마트 카드 삭제
- 새로운 **PINs** 설정 및 **PUK** (선택 사항적인 **Pin Unblocking Keys**)
- 스마트 카드에 새 슬롯 만들기
- 슬롯에 인증서, 개인 키 및 공개 키 저장

- 스마트 카드 설정 잠금 (일부 스마트 카드에는 이러한 유형의 종료가 필요합니다)

사전 요구 사항

- **pkcs15-init** 도구를 포함하는 **opensc** 패키지가 설치되어 있습니다.

  자세한 내용은 스마트 카드를 관리하고 사용하기 위한 툴 설치를 참조하십시오.

- 카드가 리더에 삽입되어 컴퓨터에 연결되어 있습니다.

- 스마트 카드에 저장할 개인 키, 공개 키 및 인증서가 있습니다. 이 절차에서는 **testuser.key,testuserpublic.key, testuser.crt** 는 개인 키, 공개 키 및 인증서에 사용되는 이름입니다.

- 현재 스마트 카드 사용자 **PIN** 및 보안 사무실 **(SO-PIN)**

절차

1. 스마트 카드를 지우고 **PIN**으로 인증합니다.

```
$ pkcs15-init --erase-card --use-default-transport-keys
Using reader with a card: Reader name
PIN [Security Officer PIN] required.
Please enter PIN [Security Officer PIN]:
```

   카드가 삭제되었습니다.

2. 스마트 카드를 초기화하고, 사용자 **PIN** 및 **PUK**, 보안 지사 및 **PUK**를 설정합니다.

```
$ pkcs15-init --create-pkcs15 --use-default-transport-keys \
   --pin 963214 --puk 321478 --so-pin 65498714 --so-puk 784123
Using reader with a card: Reader name
```

   PCks15-init 툴은 스마트 카드에 새 슬롯을 생성합니다.

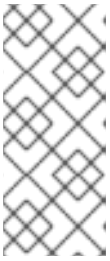
3. 슬롯의 라벨 및 인증 ID를 설정합니다.

```
$ pkcs15-init --store-pin --label testuser \
   --auth-id 01 --so-pin 65498714 --pin 963214 --puk 321478
Using reader with a card: Reader name
```

레이블은 사람이 읽을 수 있는 값(이 경우 testuser)으로 설정됩니다. auth-id 는 두 개의 16 진수 값이어야 합니다. 이 경우 01 로 설정됩니다.

4. 개인 키를 스마트 카드의 새 슬롯에 저장하고 레이블을 지정합니다.

```
$ pkcs15-init --store-private-key testuser.key --label testuser_key \
   --auth-id 01 --id 01 --pin 963214
Using reader with a card: Reader name
```



참고

--id 에 지정하는 값은 개인 키 및 인증서를 저장할 때 동일해야 합니다. --id 의 값을 지정하지 않으면 툴에 의해 더 복잡한 값을 계산하므로 자체 값을 더 쉽게 정의할 수 있습니다.

5. 스마트 카드의 새 슬롯에 인증서를 저장하고 레이블을 지정합니다.

```
$ pkcs15-init --store-certificate testuser.crt --label testuser_crt \
   --auth-id 01 --id 01 --format pem --pin 963214
Using reader with a card: Reader name
```

6. (선택 사항) 스마트 카드의 새 슬롯에 공개 키를 저장하고 레이블을 지정합니다.

```
$ pkcs15-init --store-public-key testuserpublic.key
   --label testuserpublic_key --auth-id 01 --id 01 --pin 963214
Using reader with a card: Reader name
```

참고

공개 키가 개인 키 및/또는 인증서에 해당하는 경우 해당 개인 키 및/또는 인증서와 동일한 ID를 지정해야 합니다.

7.

(선택 사항) 일부 스마트 카드는 설정을 잠그서 카드를 종료해야 합니다.

```
$ pkcs15-init -F
```

이 단계에서 스마트 카드에는 새로 생성된 슬롯의 인증서, 개인 키 및 공개 키가 포함되어 있습니다. 또한 사용자 **PIN** 및 **PUK** 및 **Security Officer PIN** 및 **PUK**를 생성했습니다.

7.7. 스마트 카드를 사용하여 IDM에 로그인

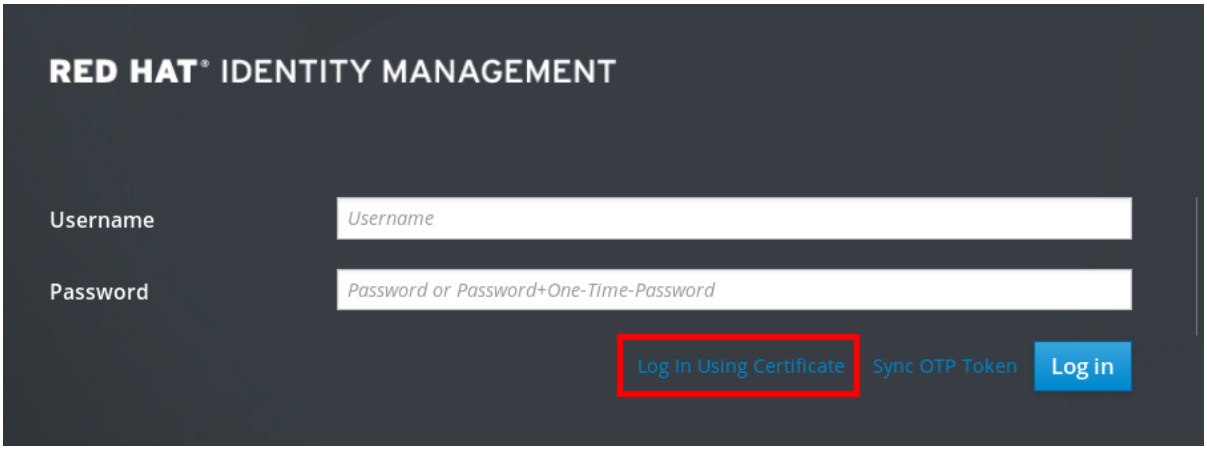
이 섹션에서는 IdM 웹 UI에 로그인하는 데 스마트 카드 사용에 대한 정보를 제공합니다.

사전 요구 사항

- 웹 브라우저가 스마트 카드 인증을 사용하도록 구성되어 있습니다.
- 스마트 카드 인증을 위해 IdM 서버가 구성되어 있습니다.
- 스마트 카드에 설치된 인증서는 IdM 서버로 알려져 있습니다.
- 스마트 카드를 잠금 해제하려면 Pin이 필요합니다.
- 스마트 카드가 리더에 연결되었습니다.

절차

1. 브라우저에서 IdM 웹 UI를 엽니다.
2. **Log In Using Certificate** 를 클릭합니다.



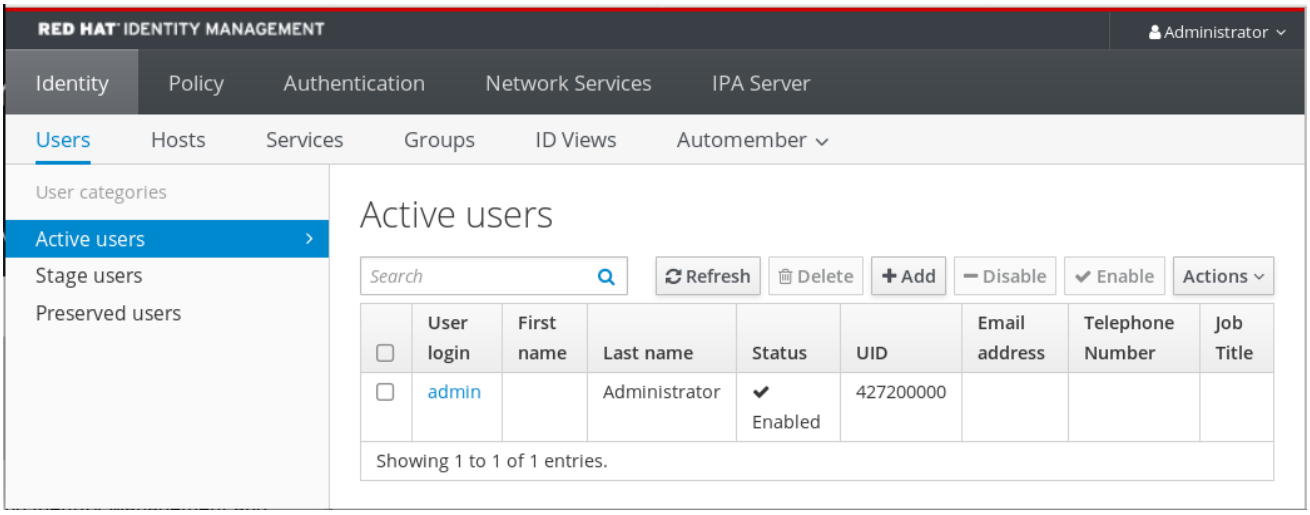
3. 암호 필요 대화 상자가 열리면 스마트 카드의 잠금을 해제하고 확인 버튼을 클릭합니다.

사용자 ID 요청 대화 상자가 열립니다.

스마트 카드에 두 개 이상의 인증서가 포함된 경우 드롭다운 목록에서 인증에 사용할 인증서를 선택하여 ID로 표시할 인증서 선택.

4. OK 버튼을 클릭합니다.

이제 IdM 웹 UI에 로그인했습니다.



7.8. 스마트 카드 인증을 사용하여 GDM 액세스 구성

Gnome Desktop Manager (GDM)에는 인증이 필요합니다. 비밀번호는 사용할 수 있지만 인증에 스마트 카드를 사용할 수도 있습니다.

이 섹션에서는 GDM에 액세스하기 위한 스마트 카드 인증에 대해 설명합니다.

스마트 카드 인증을 사용할 때의 장점은 사용자 계정이 Identity Management 도메인의 일부인 경우 티켓 생성 티켓(TGT)도 받을 수 있다는 것입니다.

사전 요구 사항

- 스마트 카드에는 인증서 및 개인 키가 포함되어 있습니다.
- 사용자 계정은 IdM 도메인의 멤버입니다.
- 스마트 카드의 인증서는 다음을 통해 사용자 항목에 매핑됩니다.
  - 특정 사용자 항목에 인증서를 할당합니다. 자세한 내용은 IdM 웹 UI의 사용자 항목에 인증서 추가 또는 IdM CLI의 사용자 항목에 인증서 추가를 참조하십시오.
  - 계정에 적용되는 인증서 매핑 데이터입니다. 자세한 내용은 스마트 카드에 대한 인증을 구성하기 위한 인증서 매핑 규칙을 참조하십시오.

절차

1. 리더에 스마트 카드를 삽입합니다.
2. 스마트 카드 Pin을 입력합니다.
3. Sign In 을 클릭합니다.

RHEL 시스템에 성공적으로 로그인되어 있으며 IdM 서버에서 제공하는 TGT가 있습니다.

검증 단계

검증 단계

- 터미널 창에서 **klist** 를 입력하고 결과를 확인합니다.

```
$ klist
Ticket cache: KEYRING:persistent:1358900015:krb_cache_TObtNMd
Default principal: example.user@REDHAT.COM

Valid starting       Expires              Service principal
04/20/2020 13:58:24  04/20/2020 23:58:24  krbtgt/EXAMPLE.COM@EXAMPLE.COM
 renew until 04/27/2020 08:58:15
```

7.9. 스마트 카드 인증을 사용하여 SU 액세스 구성

다른 사용자로 변경하려면 인증이 필요합니다. 암호 또는 인증서를 사용할 수 있습니다. 이 섹션에서는 **su** 명령과 함께 스마트 카드를 사용하는 방법을 설명합니다. 즉, **su** 명령을 입력한 후 스마트 카드 **PIN**을 입력하라는 메시지가 표시됩니다.

사전 요구 사항

- 스마트 카드에는 인증서 및 개인 키가 포함되어 있습니다.
- 카드가 리더에 삽입되어 컴퓨터에 연결되어 있습니다.

절차

- 터미널 창에서 **su** 명령을 사용하여 다른 사용자로 변경합니다.

```
$ su - example.user
PIN for smart_card
```

구성이 성공하면 스마트 카드 **PIN**을 입력하라는 메시지가 표시됩니다.

8장. IDM의 스마트 카드 인증에 대해 ADCS에서 발급한 인증서 구성

이 시나리오에서는 다음 상황을 설명합니다.

- 배포는 IdM(Identity Management)과 AD(Active Directory) 간 신뢰를 기반으로 합니다.
- 계정이 AD에 저장된 사용자에 대해 스마트 카드 인증을 허용하려고 합니다.
- 인증서는 ADCS(Active Directory Certificate Services)에 생성되고 저장됩니다.

구성은 다음 단계에서 수행됩니다.

- Active Directory에서 IdM 서버 및 클라이언트로 CA 및 사용자 인증서 복사
- ADCS 인증서를 사용하여 스마트 카드 인증을 위해 IdM 서버 및 클라이언트 구성
- 인증서 및 개인 키를 스마트 카드에 저장할 수 있도록 Pdatabind (PKCS#12) 파일을 변환
- sssd.conf 파일에서 타임아웃 구성
- 스마트 카드 인증을 위한 인증서 매핑 규칙 생성

사전 요구 사항

- IdM(Identity Management) 및 AD(Active Directory) 신뢰가 설치되어 있습니다.

  자세한 내용은 IdM과 AD 간 신뢰 설치를 참조하십시오.

- ADCS(Active Directory Certificate Services)가 설치되고 사용자를 위한 인증서가 생성됩니다.

8.1. 스마트 카드 인증

스마트 카드는 카드에 저장된 인증서를 사용하여 개인 인증을 제공할 수 있는 물리적 장치입니다. 개인 인증이란 사용자 암호와 동일한 방식으로 스마트 카드를 사용할 수 있음을 의미합니다.

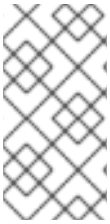
스마트 카드에 사용자 자격 증명을 개인 키와 인증서 형식으로 저장할 수 있으며 특수 소프트웨어 및 하드웨어는 이를 액세스하는 데 사용됩니다. 스마트 카드를 리더 또는 USB 소켓에 배치하고 비밀번호를 제공하는 대신 스마트 카드의 PIN 코드를 제공합니다.

스마트 카드 인증이 특정 IdM 클라이언트에서 작동하도록 하는 방법을 구성할 수 있습니다.

- 사용자는 사용자 이름 및 비밀번호 또는 스마트 카드로 인증할 수 있습니다.
- 사용자는 스마트 카드로 인증할 수 있으며 암호는 허용되지 않습니다.
- 사용자는 제거 시 기능 잠금으로 로그 아웃에 스마트 카드를 사용할 수 있으며 암호는 허용되지 않습니다.

IdM(Identity Management)은 다음을 사용하여 스마트 카드 인증을 지원합니다.

- IdM 인증 기관에서 발급한 사용자 인증서입니다. 자세한 내용은 스마트 카드 인증에 대한 ID 관리 구성을 참조하십시오.
- ADCS 인증 기관에서 발급한 사용자 인증서입니다. 자세한 내용은 IdM에서 스마트 카드 인증에 대해 ADCS에서 발급한 인증서 구성을 참조하십시오.
- RHEL 시스템에서 생성된 로컬 인증 기관에서 발급한 사용자 인증서입니다. 자세한 내용은 스마트 카드로 로컬 인증서 구성 및 가져오기를 참조하십시오.
- 외부 인증 기관에서 발급한 사용자 인증서입니다.



참고

스마트 카드 인증을 사용하려면 하드웨어 요구 사항을 참조하십시오. RHEL8의 스마트 카드 지원.

8.2. 신뢰 구성 및 인증서 사용에 필요한 WINDOWS SERVER 설정

이 섹션에서는 Windows Server에서 구성해야 하는 내용을 요약합니다.

- ADCS(Active Directory 인증서 서비스)가 설치됨
- 인증 기관 생성
- [선택 사항] 인증 기관 웹 등록을 사용하는 경우 인터넷 정보 서비스 (IIS)를 구성해야 합니다.

인증서를 내보냅니다.

- 키의 2048 bit 이상이 있어야 합니다.
- 개인 키 포함
- 다음 형식의 인증서가 필요합니다. Personal Information Exchange — PKCS #12(.PFX)
  - 인증서 개인 정보 보호 활성화

8.3. SFTP를 사용하여 ACTIVE DIRECTORY에서 인증서 복사

스마트 카드 인증 기능을 사용하려면 다음 인증서 파일을 복사해야 합니다.

- IdM 서버의 CER 형식의 루트 CA 인증서: adcs-winserver-ca.cer.

- 개인 키가 있는 사용자 인증서: IdM 클라이언트의 aduser1.pfx 입니다.

참고

이 절차에서는 SSH 액세스가 허용될 수 있습니다. SSH를 사용할 수 없는 경우 사용자는 AD Server에서 IdM 서버 및 클라이언트로 파일을 복사해야 합니다.

절차

1. IdM 서버에서 연결하고 adcs-winserver-ca.cer 루트 인증서를 IdM 서버에 복사합니다.

```
root@idmserver ~]# sftp Administrator@winserver.ad.example.com
Administrator@winserver.ad.example.com's password:
Connected to Administrator@winserver.ad.example.com.
sftp> cd <Path to certificates>
sftp> ls
adcs-winserver-ca.cer    aduser1.pfx
sftp>
sftp> get adcs-winserver-ca.cer
Fetching <Path to certificates>/adcs-winserver-ca.cer to adcs-winserver-ca.cer
<Path to certificates>/adcs-winserver-ca.cer                100%  1254    15KB/s 00:00
sftp quit
```

2. IdM 클라이언트에서 연결하고 aduser1.pfx 사용자 인증서를 클라이언트에 복사합니다.

```
[root@client1 ~]# sftp Administrator@winserver.ad.example.com
Administrator@winserver.ad.example.com's password:
Connected to Administrator@winserver.ad.example.com.
sftp> cd /<Path to certificates>
sftp> get aduser1.pfx
Fetching <Path to certificates>/aduser1.pfx to aduser1.pfx
<Path to certificates>/aduser1.pfx                100%  1254    15KB/s 00:00
sftp quit
```

이제 CA 인증서가 IdM 서버에 저장되고 사용자 인증서가 클라이언트 시스템에 저장됩니다.

8.4. ADCS 인증서를 사용하여 스마트 카드 인증을 위해 IDM 서버 및 클라이언트 구성

IdM 환경에서 스마트 카드 인증을 사용할 수 있도록 IdM(Identity Management) 서버와 클라이언트를 구성해야 합니다. IdM에는 필요한 모든 변경을 수행하는 ipa-advise 스크립트가 포함되어 있습니다.

- 필수 패키지 설치
- IdM 서버 및 클라이언트를 설정합니다.
- CA 인증서를 예상 위치에 복사

IdM 서버에서 ipa-advise 를 실행할 수 있습니다.

이 절차에서는 다음을 설명합니다.

- IdM 서버에서 다음을 수행합니다. 스마트 카드 인증을 위해 IdM 서버를 구성하기 위해 ipa-advise 스크립트를 준비합니다.
- IdM 서버에서 다음을 수행합니다. 스마트 카드 인증을 위해 IdM 클라이언트를 구성하기 위해 ipa-advise 스크립트를 준비합니다.
- IdM 서버에서 다음을 수행합니다. AD 인증서를 사용하여 IdM 서버에 ipa-advise 서버 스크립트 적용.
- 클라이언트 스크립트를 IdM 클라이언트 시스템으로 이동합니다.
- IdM 클라이언트의 경우: AD 인증서를 사용하여 IdM 클라이언트에 ipa-advise 클라이언트 스크립트 적용.

사전 요구 사항

- 인증서가 IdM 서버에 복사되었습니다.
- Kerberos 티켓을 받으십시오.
- 관리 권한이 있는 사용자로 로그인합니다.

절차

1. IdM 서버에서 ipa-advise 스크립트를 사용하여 클라이언트를 구성합니다.

```
[root@idmserver ~]# ipa-advise config-client-for-smart-card-auth > sc_client.sh
```

2. IdM 서버에서 ipa-advise 스크립트를 사용하여 서버를 구성합니다.

```
[root@idmserver ~]# ipa-advise config-server-for-smart-card-auth > sc_server.sh
```

3. IdM 서버에서 스크립트를 실행합니다.

```
[root@idmserver ~]# sh -x sc_server.sh adcs-winserver-ca.cer
```

- IdM Apache HTTP 서버를 구성합니다.
- KDC (Key Distribution Center)에서 Kerberos (PKINIT)에서 초기 인증의 공개 키 암호화를 활성화합니다.
- 스마트 카드 권한 부여 요청을 수락하도록 IdM 웹 UI를 구성합니다.

4. sc_client.sh 스크립트를 클라이언트 시스템에 복사합니다.

```
[root@idmserver ~]# scp sc_client.sh root@client1.idm.example.com:/root
Password:
sc_client.sh                  100%  2857   1.6MB/s   00:00
```

5. Windows 인증서를 클라이언트 시스템에 복사합니다.

```
[root@idmserver ~]# scp adcs-winserver-ca.cer root@client1.idm.example.com:/root
Password:
adcs-winserver-ca.cer                  100%  1254   952.0KB/s   00:00
```

6. 클라이언트 시스템에서 클라이언트 스크립트를 실행합니다.

```
[root@idmclient1 ~]# sh -x sc_client.sh adcs-winserver-ca.cer
```

CA 인증서는 IdM 서버 및 클라이언트 시스템에 올바른 형식으로 설치되며 다음 단계는 사용자 인증서를 스마트 카드 자체에 복사하는 것입니다.

8.5. PNETWORKPOLICY 파일 변환

PNetworkPolicy (PKCS#12) 파일을 스마트 카드에 저장하기 전에 다음을 수행해야합니다.

- 파일을 PEM 형식으로 변환
- 개인 키와 인증서를 두 개의 다른 파일에 추출

사전 요구 사항

- Pjournal은 IdM 클라이언트 시스템에 복사됩니다.

절차

1. IdM 클라이언트에서 PEM 형식으로 다음을 수행합니다.

```
[root@idmclient1 ~]# openssl pkcs12 -in aduser1.pfx -out aduser1_cert_only.pem -
clcerts -nodes
Enter Import Password:
```

2. 키를 별도의 파일에 추출합니다.

```
[root@idmclient1 ~]# openssl pkcs12 -in adduser1.pfx -nocerts -out adduser1.pem >
aduser1.key
```

3. 공용 인증서를 별도의 파일에 추출합니다.

```
[root@idmclient1 ~]# openssl pkcs12 -in adduser1.pfx -clcerts -nokeys -out
aduser1_cert_only.pem > aduser1.crt
```

이 시점에서 aduser1.key 및 aduser1.crt 를 스마트 카드에 저장할 수 있습니다.

8.6. 스마트 카드를 관리하고 사용하기 위한 도구 설치

스마트 카드를 구성하려면 인증서를 생성하고 스마트 카드에 저장할 수 있는 도구가 필요합니다.

다음 작업을 수행해야 합니다.

- 인증서를 관리하는 데 도움이 되는 **gnutls-utils** 패키지를 설치합니다.
- 스마트 카드로 작동하도록 라이브러리 및 유틸리티 세트를 제공하는 **opensc** 패키지를 설치합니다.
- 스마트 카드 리더와 통신하는 **pcscd** 서비스를 시작합니다.

절차

1. **opensc** 및 **gnutls-utils** 패키지를 설치합니다.

```
# dnf -y install opensc gnutls-utils
```

2. **pcscd** 서비스를 시작합니다.

```
# systemctl start pcscd
```

pcscd 서비스가 실행 중인지 확인합니다.

8.7. 스마트 카드에 인증서 저장

이 섹션에서는 구성하는 데 도움이 되는 **pkcs15-init** 도구를 사용한 스마트 카드 구성에 대해 설명합니다.

- 스마트 카드 삭제

- 새로운 PINs 설정 및 PUK (선택 사항적인 Pin Unblocking Keys)
- 스마트 카드에 새 슬롯 만들기
- 슬롯에 인증서, 개인 키 및 공개 키 저장
- 스마트 카드 설정 잠금 (일부 스마트 카드에는 이러한 유형의 종료가 필요합니다)

사전 요구 사항

- pkcs15-init 도구를 포함하는 opensc 패키지가 설치되어 있습니다.

  자세한 내용은 스마트 카드를 관리하고 사용하기 위한 툴 설치를 참조하십시오.
- 카드가 리더에 삽입되어 컴퓨터에 연결되어 있습니다.
- 스마트 카드에 저장할 개인 키, 공개 키 및 인증서가 있습니다. 이 절차에서는 testuser.key,testuserpublic.key, testuser.crt 는 개인 키, 공개 키 및 인증서에 사용되는 이름입니다.
- 현재 스마트 카드 사용자 PIN 및 보안 사무실 (SO-PIN)

절차

1. 스마트 카드를 지우고 PIN으로 인증합니다.

```
$ pkcs15-init --erase-card --use-default-transport-keys
Using reader with a card: Reader name
PIN [Security Officer PIN] required.
Please enter PIN [Security Officer PIN]:
```

   카드가 삭제되었습니다.

2. 스마트 카드를 초기화하고, 사용자 **PIN** 및 **PUK**, 보안 지사 및 **PUK**를 설정합니다.

```
$ pkcs15-init --create-pkcs15 --use-default-transport-keys \
   --pin 963214 --puk 321478 --so-pin 65498714 --so-puk 784123
Using reader with a card: Reader name
```

PCks15-init 툴은 스마트 카드에 새 슬롯을 생성합니다.

3. 슬롯의 라벨 및 인증 **ID**를 설정합니다.

```
$ pkcs15-init --store-pin --label testuser \
   --auth-id 01 --so-pin 65498714 --pin 963214 --puk 321478
Using reader with a card: Reader name
```

레이블은 사람이 읽을 수 있는 값(이 경우 **testuser**)으로 설정됩니다. **auth-id** 는 두 개의 **16** 진수 값이어야 합니다. 이 경우 **01** 로 설정됩니다.

4. 개인 키를 스마트 카드의 새 슬롯에 저장하고 레이블을 지정합니다.

```
$ pkcs15-init --store-private-key testuser.key --label testuser_key \
   --auth-id 01 --id 01 --pin 963214
Using reader with a card: Reader name
```

참고

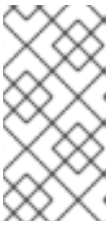
--id 에 지정하는 값은 개인 키 및 인증서를 저장할 때 동일해야 합니다. **--id** 의 값을 지정하지 않으면 툴에 의해 더 복잡한 값을 계산하므로 자체 값을 더 쉽게 정의할 수 있습니다.

5. 스마트 카드의 새 슬롯에 인증서를 저장하고 레이블을 지정합니다.

```
$ pkcs15-init --store-certificate testuser.crt --label testuser_crt \
   --auth-id 01 --id 01 --format pem --pin 963214
Using reader with a card: Reader name
```

6. (선택 사항) 스마트 카드의 새 슬롯에 공개 키를 저장하고 레이블을 지정합니다.

```
$ pkcs15-init --store-public-key testuserpublic.key
   --label testuserpublic_key --auth-id 01 --id 01 --pin 963214
Using reader with a card: Reader name
```



참고

공개 키가 개인 키 및/또는 인증서에 해당하는 경우 해당 개인 키 및/또는 인증서와 동일한 ID를 지정해야 합니다.

7. (선택 사항) 일부 스마트 카드는 설정을 잠그서 카드를 종료해야 합니다.

```
$ pkcs15-init -F
```

이 단계에서 스마트 카드에는 새로 생성된 슬롯의 인증서, 개인 키 및 공개 키가 포함되어 있습니다. 또한 사용자 PIN 및 PUK 및 Security Officer PIN 및 PUK를 생성했습니다.

8.8. SSSD.CONF에서 타임아웃 구성

스마트 카드 인증서의 인증은 SSSD에서 사용하는 기본 시간 초과보다 오래 걸릴 수 있습니다. 시간 초과 만료는 다음과 같이 인해 발생할 수 있습니다.

- 느린 리더
- 전달은 물리적 장치를 가상 환경으로 구성
- 스마트 카드에 저장된 인증서가 너무 많습니다.
- OCSP (Online Certificate Status Protocol) 응답기 (Online Certificate Status Protocol)가 인증서를 확인하는 데 사용되는 경우 속도 저하

이 경우 sssd.conf 파일에서 다음 시간 초과를 연장할 수 있습니다(예: 60초).

- p11_child_timeout

- krb5_auth_timeout

사전 요구 사항

- root로 로그인해야 합니다.

절차

1. sssd.conf 파일을 엽니다.

```
[root@idmclient1 ~]# vim /etc/sssd/sssd.conf
```

2. p11_hiera_timeout 값을 변경합니다.

```
[pam]
p11_child_timeout = 60
```

3. RHEA 5_auth_timeout:의 값을 변경합니다.

```
[domain/IDM.EXAMPLE.COM]
krb5_auth_timeout = 60
```

4. 설정을 저장합니다.

이제 시간 초과로 인증에 실패하기 전에 스마트 카드와의 상호 작용이 1분(60초) 동안 실행될 수 있습니다.

8.9. 스마트 카드 인증을 위한 인증서 매핑 규칙 생성

AD(Active Directory)에 계정이 있고 IdM(Identity Management)에 계정이 있는 사용자에 대해 하나의 인증서를 사용하려는 경우 IdM 서버에서 인증서 매핑 규칙을 생성할 수 있습니다.

이러한 규칙을 만든 후 사용자는 두 도메인에서 스마트 카드로 인증할 수 있습니다.

인증서 매핑 규칙에 대한 자세한 내용은 스마트 카드에서 인증 구성을 위한 인증서 매핑 규칙을 참조하

십시오.

9장. ID 관리에서 인증서 매핑 규칙 구성

9.1. 스마트 카드에서 인증을 구성하기 위한 인증서 매핑 규칙

인증서 매핑 규칙은 **IdM(Identity Management)** 관리자가 특정 사용자의 인증서에 액세스할 수 없는 경우 시나리오에서 인증서를 사용하여 인증할 수 있는 편리한 방법입니다. 이러한 액세스 부족은 일반적으로 외부 인증 기관에서 인증서를 발급했기 때문에 발생합니다. 특수 사용 사례는 **IdM** 도메인이 신뢰 관계에 있는 **AD(Active Directory)**의 인증서 시스템에서 발급한 인증서로 표시됩니다.

IdM 환경이 스마트 카드를 사용하는 많은 사용자가 사용하는 경우 인증서 매핑 규칙도 편리합니다. 이 경우 전체 인증서를 추가하는 것이 복잡할 수 있습니다. 제목과 발행자는 대부분의 시나리오에서 예측할 수 있으므로 전체 인증서보다 시간을 더 쉽게 추가할 수 있습니다. 시스템 관리자는 인증서 매핑 규칙을 생성하고 특정 사용자에게 인증서를 발행하기 전에 인증서 매핑 데이터를 사용자 항목에 추가할 수 있습니다. 인증서가 발급되면 전체 인증서가 사용자 항목에 아직 업로드되지 않은 경우에도 인증서를 사용하여 로그인할 수 있습니다.

또한 인증서를 정기적으로 갱신해야 하므로 인증서 매핑 규칙이 관리 오버헤드를 줄입니다. 사용자의 인증서가 갱신되면 관리자는 사용자 항목을 업데이트할 필요가 없습니다. 예를 들어 매핑이 **Subject** 및 **Issuer** 값을 기반으로 하고 새 인증서가 이전 인증서와 동일한 주체 및 발행자 값을 기반으로 하는 경우 매핑이 계속 적용됩니다. 반대로 전체 인증서가 사용된 경우 관리자는 새 인증서를 사용자 항목에 업로드하여 이전 인증서를 교체해야 합니다.

인증서 매핑을 설정하려면 다음을 수행합니다.

1. 관리자는 인증서 매핑 데이터(일반적으로 발급자 및 주체) 또는 전체 인증서를 사용자 계정으로 로드해야 합니다.

2. 관리자가 사용자를 위해 **IdM**에 성공적으로 로그인할 수 있도록 인증서 매핑 규칙을 생성해야 합니다.

    a. 계정에 인증서 매핑 데이터 항목이 포함된 경우

    b. 인증서 매핑 데이터 항목이 인증서 정보와 일치하는 경우

매핑 규칙을 구성하는 개별 구성 요소와 해당 규칙을 가져와서 사용하는 방법에 대한 자세한 내용은 **IdM**의 **ID** 매핑 규칙 구성 요소 및 일치하는 규칙에 사용할 인증서에서 발행자 가져오기를 참조하십시오.

이후 최종 사용자가 인증서를 제공하면 파일 시스템 이나 스마트 카드에 저장된 인증이 성공합니다.

9.1.1. Active Directory 도메인을 사용하여 신뢰를 위한 인증서 매핑 규칙

이 섹션에서는 IdM 배포가 AD(Active Directory) 도메인과의 신뢰 관계에 있는 경우 가능한 다양한 인증서 매핑 사용 사례에 대해 간단히 설명합니다.

인증서 매핑 규칙은 신뢰할 수 있는 AD 인증서 시스템에서 발행한 스마트 카드 인증서가 있는 사용자에 대해 IdM 리소스에 대한 액세스를 활성화하는 편리한 방법입니다. AD 구성에 따라 다음 시나리오가 가능합니다.

- AD에서 인증서를 발행하지만 사용자와 인증서가 IdM에 저장된 경우 매핑과 인증 요청 전체 처리가 IdM 측에서 수행됩니다. 이 시나리오 구성에 대한 자세한 내용은 IdM에 저장된 사용자의 인증서 매핑구성을 참조하십시오.

- 사용자가 AD에 저장되면 인증 요청 처리가 AD에서 수행됩니다. 여기에는 세 가지 다른 케이스가 있습니다:

  - AD 사용자 항목에는 전체 인증서가 포함되어 있습니다. 이 시나리오에서 IdM을 구성하는 방법에 대한 자세한 내용은 AD 사용자 항목에 전체 인증서가 포함된 사용자에 대한 인증서 매핑 구성을 참조하십시오.

  - AD는 사용자 인증서를 사용자 계정에 매핑하도록 구성되어 있습니다. 이 경우 AD 사용자 항목에는 전체 인증서가 포함되어 있지 않지만 altSecurityIdentities 라는 속성이 포함되어 있습니다. 이 시나리오에서 IdM을 구성하는 방법에 대한 자세한 내용은 사용자 계정에 사용자 인증서를 매핑하도록 AD가 구성된 경우 인증서 매핑 구성을 참조하십시오.

  - AD 사용자 항목에는 전체 인증서와 매핑 데이터가 포함되어 있지 않습니다. 이 경우 유일한 해결책은 ipa idoverrideuser-add 명령을 사용하여 IdM의 AD 사용자 ID 재정의에 전체 인증서를 추가하는 것입니다. 자세한 내용은 AD 사용자 항목에 인증서 또는 매핑 데이터가 없는 경우 인증서 매핑 구성을 참조하십시오.

9.1.2. IdM의 ID 매핑 규칙의 구성 요소

이 섹션에서는 IdM의 *ID 매핑 규칙*과 구성 방법에 대해 설명합니다. 각 구성 요소에는 재정의할 수 있는 기본값이 있습니다. 웹 UI 또는 CLI에서 구성 요소를 정의할 수 있습니다. CLI에서 ipa certmaprule-add 명령을 사용하여 ID 매핑 규칙이 생성됩니다.

매핑 규칙

매핑 규칙 구성 요소는 하나 이상의 사용자 계정과 인증서를 연결(또는 *매핑*)합니다. 규칙은 인증서를 의도된 사용자 계정과 연결하는 **LDAP** 검색 필터를 정의합니다.

다른 **CA**(인증 기관)에서 발급한 인증서는 속성이 다를 수 있으며 다른 도메인에서 사용할 수 있습니다. 따라서 **IdM**은 무조건 매핑 규칙을 적용하지 않고 적절한 인증서에만 적용됩니다. 적절한 인증서는 *일치하는 규칙을* 사용하여 정의합니다.

매핑 규칙 옵션을 비워 두면 **DER** 인코딩 바이너리 파일로 **userCertificate** 속성에서 인증서가 검색됩니다.

--maprule 옵션을 사용하여 **CLI**에서 매핑 규칙을 정의합니다.

일치 규칙

일치하는 규칙 구성 요소는 매핑 규칙을 적용할 인증서를 선택합니다. 기본 일치 규칙은 디지털 서명 키 사용 및 **clientAuth** 확장 키 사용과의 인증서와 일치합니다.

--matchrule 옵션을 사용하여 **CLI**에서 일치하는 규칙을 정의합니다.

도메인 목록

도메인 목록은 **ID** 매핑 규칙을 처리할 때 **IdM**에서 사용자를 검색할 **ID** 도메인을 지정합니다. 옵션을 지정하지 않은 경우 **IdM**은 **IdM** 클라이언트가 속한 로컬 도메인에서 사용자만 검색합니다.

--domain 옵션을 사용하여 **CLI**에서 도메인을 정의합니다.

우선 순위

인증서에 여러 규칙이 적용되면 우선 순위가 가장 높은 규칙이 우선합니다. 다른 모든 규칙은 무시됩니다.

- 숫자 값이 낮으면 **ID** 매핑 규칙의 우선 순위가 높습니다. 예를 들어 우선순위 **1**이 있는 규칙은 우선순위 **2**가 있는 규칙보다 우선 순위가 높습니다.
- 규칙의 우선순위 값이 정의되지 않은 경우 우선 순위가 가장 낮습니다.

--priority 옵션을 사용하여 CLI에서 매핑 규칙 우선 순위를 정의합니다.

인증서 매핑 규칙 예

CLI를 사용하여 해당 인증서의 주체가 IdM의 사용자 계정의 certmapdata 항목과 일치하는 한 EXAMPLE.ORG 조직의 스마트 카드 CA 에서 발급한 인증서에 대한 인증을 허용하는 simple_rule 이라는 인증서 매핑 규칙을 정의합니다.

```
# ipa certmaprule-add simple_rule --matchrule '<ISSUER>CN=Smart Card
CA,O=EXAMPLE.ORG' --maprule '(ipacertmapdata=X509:<I>{issuer_dn!nss_x500}<S>
{subject_dn!nss_x500})'
```

9.1.3. 일치하는 규칙에 사용할 인증서에서 발급자 가져오기

이 절차에서는 인증서 매핑 규칙의 일치하는 규칙에 복사하여 붙여넣을 수 있도록 인증서에서 발급자 정보를 가져오는 방법을 설명합니다. 일치하는 규칙에 필요한 발행자 형식을 가져오려면 openssl x509 유틸리티를 사용합니다.

사전 요구 사항

- .pem 또는 .crt 형식의 사용자 인증서가 있습니다.

절차

1. 인증서에서 사용자 정보를 가져옵니다. 다음을 사용하여 openssl x509 인증서 디스플레이 및 서명 유틸리티를 사용합니다.

   - 인코딩된 버전의 요청 출력을 방지하는 -noout 옵션
   - 발행자 이름을 출력하는 -issuer 옵션
   - 인증서를 읽을 입력 파일 이름을 지정하는 -in 옵션
   - 가장 구체적인 상대 고유 이름(RDN)으로 출력을 표시하는 RFC2253 값과 함께 -nameopt 옵션

   입력 파일에 Identity Management 인증서가 포함된 경우 명령의 출력은 조직 정보를

사용하여 **Issuer**가 정의되었음을 보여줍니다.

```
# openssl x509 -noout -issuer -in idm_user.crt -nameopt RFC2253
issuer=CN=Certificate Authority,O=REALM.EXAMPLE.COM
```

입력 파일에 **Active Directory** 인증서가 포함된 경우 명령의 출력은 도메인 구성 요소 정보를 사용하여 **Issuer**가 정의되었음을 보여줍니다.

```
# openssl x509 -noout -issuer -in ad_user.crt -nameopt RFC2253
issuer=CN=AD-WIN2012R2-CA,DC=AD,DC=EXAMPLE,DC=COM
```

2.

선택적으로 인증서 발행자가 **ad.example.com** 도메인의 추출된 **AD-WIN2012R2-CA** 여야 하며 인증서의 주체가 **IdM**의 사용자 계정에 있는 **certmapdata** 항목과 일치하도록 지정하는 일치 규칙에 따라 **CLI**에서 새 매핑 규칙을 생성하려면 다음을 수행합니다.

```
# ipa certmaprule-add simple_rule --matchrule '<ISSUER>CN=AD-WIN2012R2-
CA,DC=AD,DC=EXAMPLE,DC=COM' --maprule '(ipacertmapdata=X509:<I>
{issuer_dn!nss_x500}<S>{subject_dn!nss_x500})'
```

9.1.4. 추가 리소스

- **sss-certmap(5)** 매뉴얼 페이지를 참조하십시오.

9.2. IDM에 저장된 사용자의 인증서 매핑 구성

이 사용자 스토리는 **IdM**에 인증서 인증이 저장되는 사용자가 **IdM**에 저장된 경우 **IdM**에서 인증서 매핑을 활성화하기 위해 시스템 관리자가 수행해야 하는 단계를 설명합니다. 이 섹션에서는 다음에 대해 설명합니다.

- 매핑 규칙에 지정된 조건과 일치하는 인증서가 있고 인증서 매핑 데이터 항목이 **IdM**에 인증될 수 있도록 인증서 매핑 규칙을 설정하는 방법은 인증서 매핑 규칙을 설정하는 방법입니다.

- 모두 인증서 매핑 데이터 항목에 지정된 값이 포함되어 있는 한 사용자가 여러 인증서를 사용하여 인증할 수 있도록 인증서 매핑 데이터를 **IdM** 사용자 항목에 입력하는 방법.

사전 요구 사항

- 사용자 계정은 **IdM**에 있습니다.

- 관리자는 사용자 항목에 추가할 전체 인증서 또는 인증서 매핑 데이터를 보유합니다.

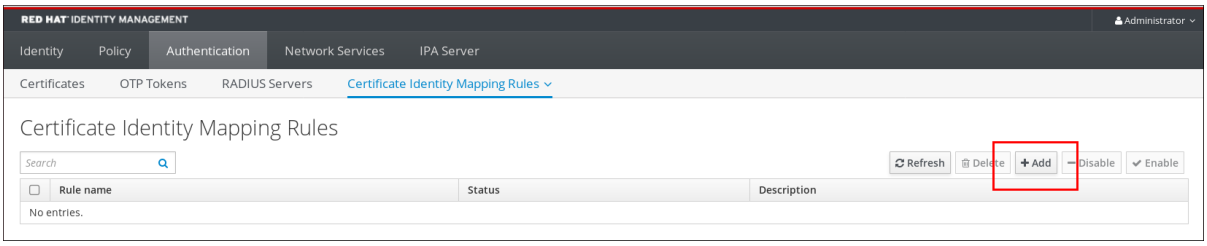
9.2.1. IdM 웹 UI에서 인증서 매핑 규칙 추가

1. 관리자로 IdM 웹 UI에 로그인합니다.

2. 인증 → 인증서 매핑 규칙 → 인증서 ID 매핑 규칙으로 이동합니다.

3. 추가를 클릭합니다.

그림 9.1. IdM 웹 UI에서 새 인증서 매핑 규칙 추가



4. 규칙 이름을 입력합니다.

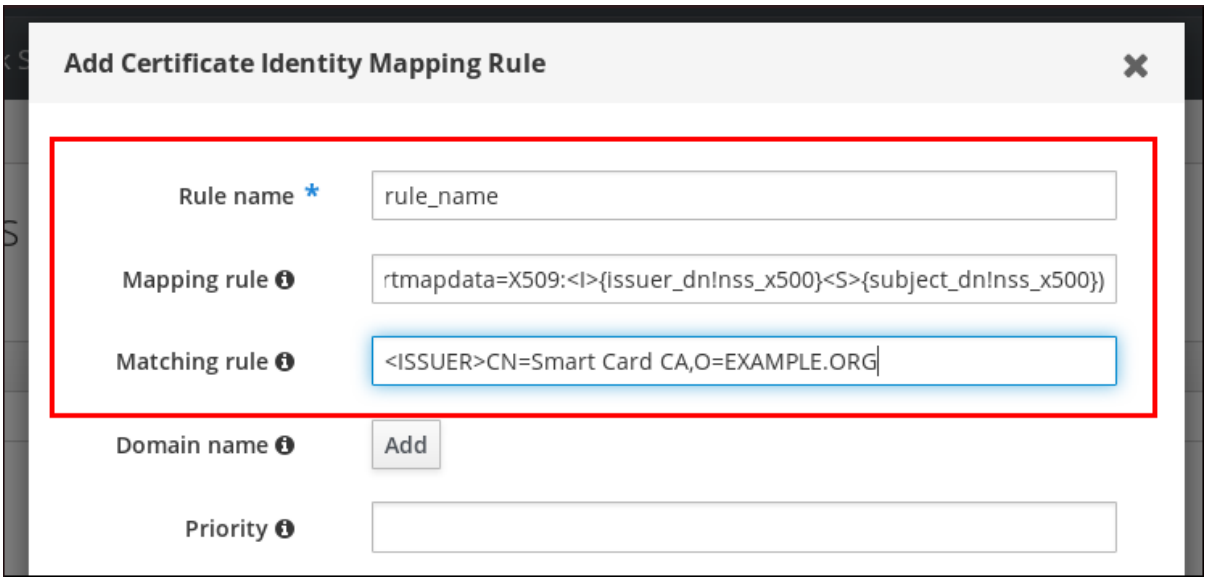
5. 매핑 규칙을 입력합니다. 예를 들어 IdM이 제공된 인증서의 Issuer 및 Subject 항목을 검색하고 제공된 인증서의 두 항목에 있는 정보에 대해 인증하거나 사용하지 않도록 결정하십시오.

```
(ipacertmapdata=X509:<I>{issuer_dn!nss_x500}<S>{subject_dn!nss_x500})
```

6. 일치 규칙을 입력합니다. 예를 들어 EXAMPLE.ORG 조직의 스마트 카드 CA 에서 발급한 인증서만 허용하도록 하려면 IdM에 사용자를 인증합니다.

```
<ISSUER>CN=Smart Card CA,O=EXAMPLE.ORG
```

그림 9.2. IdM 웹 UI에서 인증서 매핑 규칙의 세부 정보 입력



7. 대화 상자 하단에 있는 추가 를 클릭하여 규칙을 추가하고 상자를 닫습니다.

8. **SSSD(System Security Services Daemon)**는 주기적으로 인증서 매핑 규칙을 다시 읽습니다. 새로 생성된 규칙이 즉시 로드되도록 하려면 **SSSD**를 다시 시작합니다.

```
# systemctl restart sssd
```

이제 **IdM** 사용자 항목의 인증서 매핑 데이터를 사용하여 스마트 카드 인증서에서 찾은 매핑 규칙에 지정된 데이터 유형을 비교하는 인증서 매핑 규칙 세트가 있습니다. 일치 항목을 찾으면 일치하는 사용자를 인증합니다.

9.2.2. IdM CLI에서 인증서 매핑 규칙 추가

1. 관리자의 자격 증명을 가져옵니다.

```
# kinit admin
```

2. 매핑 규칙을 입력하고 매핑 규칙을 기반으로 하는 일치 규칙을 입력합니다. 예를 들어 **IdM**이 제공되는 인증서의 **Issuer** 및 **Subject** 항목을 검색하고 제공된 인증서의 두 항목에 있는 정보에 기반하여 **EXAMPLE.ORG** 조직의 **Smart Card CA** 에서 발행한 인증서만 인식하도록 합니다.

```
# ipa certmaprule-add rule_name --matchrule '<ISSUER>CN=Smart Card
CA,O=EXAMPLE.ORG' --maprule '(ipacertmapdata=X509:<I>{issuer_dn!nss_x500}<S>
{subject_dn!nss_x500})'
----------------------------------------------------
Added Certificate Identity Mapping Rule "rule_name"
```

```
----------------------------------------------------
  Rule name: rule_name
  Mapping rule: (ipacertmapdata=X509:<I>{issuer_dn!nss_x500}<S>
{subject_dn!nss_x500})
  Matching rule: <ISSUER>CN=Smart Card CA,O=EXAMPLE.ORG
  Enabled: TRUE
```

3. SSSD(System Security Services Daemon)는 주기적으로 인증서 매핑 규칙을 다시 읽습니다. 새로 생성된 규칙이 즉시 로드되도록 하려면 SSSD를 다시 시작합니다.

```
# systemctl restart sssd
```

이제 IdM 사용자 항목의 인증서 매핑 데이터를 사용하여 스마트 카드 인증서에서 찾은 매핑 규칙에 지정된 데이터 유형을 비교하는 인증서 매핑 규칙 세트가 있습니다. 일치 항목을 찾으면 일치하는 사용자를 인증합니다.

9.2.3. IdM 웹 UI의 사용자 항목에 인증서 매핑 데이터 추가

1. 관리자로 IdM 웹 UI에 로그인합니다.

2. 사용자 → 활성 사용자 → idm_user 로 이동합니다.

3. 인증서 매핑 데이터 옵션을 찾아 추가 를 클릭합니다.

4. idm_user 의 인증서가 있는 경우:

   a. 명령줄 인터페이스에서 cat 유틸리티 또는 텍스트 편집기를 사용하여 인증서를 표시합니다.

```
[root@server ~]# cat idm_user_certificate.pem
-----BEGIN CERTIFICATE-----
MIIFFTCCA/2gAwIBAgIBEjANBgkqhkiG9w0BAQsFADA6MRgwFgYDVQQKDA9JRE
0u
RVhBTVBMRS5DT00xHjAcBgNVBAMMFUNlcnRpZmljYXRlIEF1dGhvcml0eTAeFw0
x
ODA5MDIxODE1MzlaFw0yMDA5MDIxODE1MzlaMCwxGDAWBgNVBAoMD0lETS5F
WEFN
[...output truncated...]
```

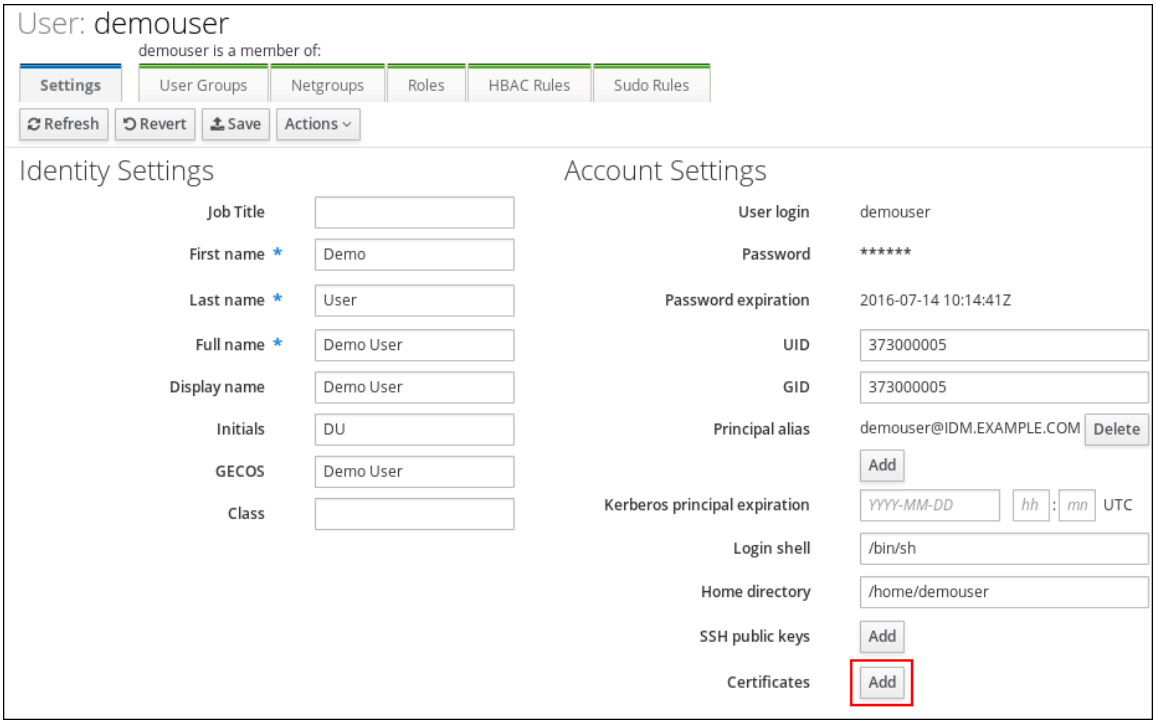
b.

인증서를 복사합니다.

c.

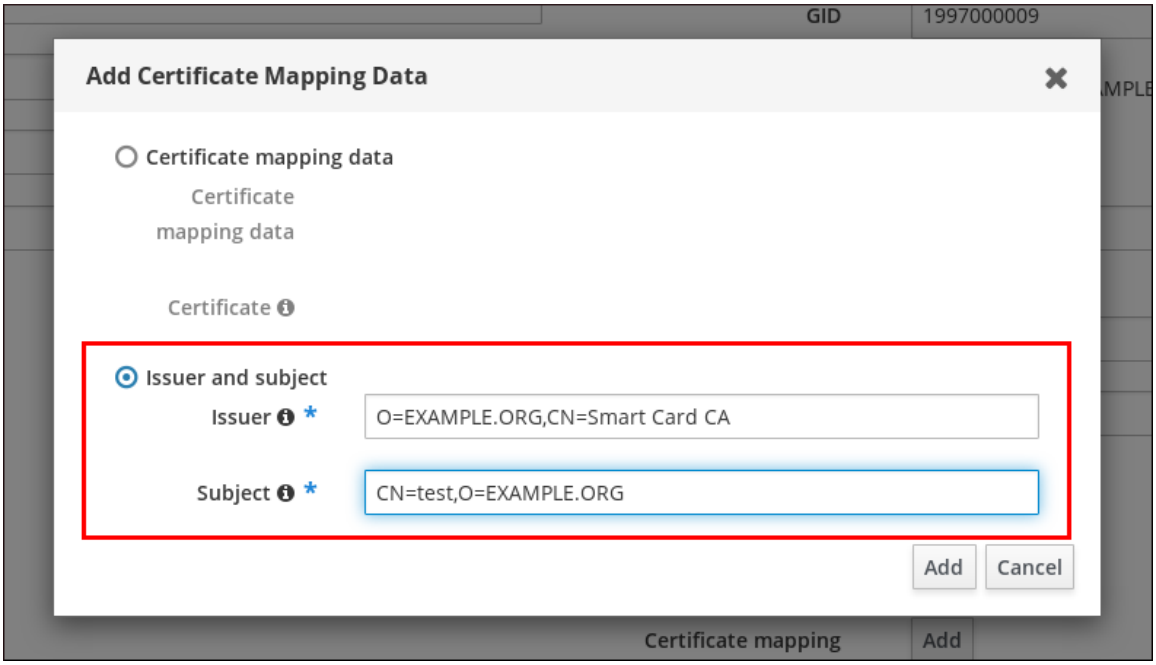
IdM 웹 UI에서 인증서 옆에 있는 추가 를 클릭하고 인증서가 열리는 창에 인증서를 붙여 넣습니다.

그림 9.3. 사용자 인증서 매핑 데이터 추가: 인증서



또는 **idm_user** 에 대한 인증서가 없지만 **Issuer** 및 인증서 주체 를 알고 있는 경우 **Issuer** 및 **subject** 의 라디오 버튼을 확인하고 해당 상자에 값을 입력합니다.

그림 9.4. 사용자의 인증서 매핑 데이터 추가: 발급자 및 주체



5. 추가를 클릭합니다.

6. 선택적으로 **.pem** 형식의 전체 인증서에 액세스할 수 있는 경우 사용자 및 인증서가 연결되어 있는지 확인합니다.

a. **sss_cache** 유틸리티를 사용하여 **SSSD** 캐시의 **idm_user** 레코드를 무효화하고 **idm_user** 정보를 강제로 다시 로드합니다.

```
# sss_cache -u idm_user
```

b. **IdM** 사용자의 인증서가 포함된 파일 이름과 **ipa certmap-match** 명령을 실행합니다.

```
# ipa certmap-match idm_user_cert.pem
--------------
1 user matched
--------------
 Domain: IDM.EXAMPLE.COM
 User logins: idm_user
----------------------------
Number of entries returned 1
----------------------------
```

출력에 이제 **idm_user** 에 인증서 매핑 데이터가 추가되고 해당 매핑 규칙이 있는지 확인합니다. 즉, 정의된 인증서 매핑 데이터와 일치하는 모든 인증서를 사용하여 **idm_user** 로 인증할 수 있습니다.

9.2.4. IdM CLI의 사용자 항목에 인증서 매핑 데이터 추가

1. 관리자의 자격 증명을 가져옵니다.

```
# kinit admin
```

2. 사용 시 **idm_user** 의 인증서가 있는 경우 **ipa user-add-cert** 명령을 사용하여 사용자 계정에 인증서를 추가합니다.

```
# CERT=`cat idm_user_cert.pem | tail -n +2| head -n -1 | tr -d '\r\n'\`
# ipa user-add-certmapdata idm_user --certificate $CERT
```

또는 **idm_user** 의 인증서가 폐기되어 있지 않지만 **Issuer** 및 **idm_user** 인증서의 주체 를 알

고 있는 경우:

```
# ipa user-add-certmapdata idm_user --subject "O=EXAMPLE.ORG,CN=test" --issuer
"CN=Smart Card CA,O=EXAMPLE.ORG"
-----------------------------------------
Added certificate mappings to user "idm_user"
-----------------------------------------
  User login: idm_user
  Certificate mapping data: X509:<I>O=EXAMPLE.ORG,CN=Smart Card
CA<S>CN=test,O=EXAMPLE.ORG
```

3. 선택적으로 **.pem** 형식의 전체 인증서에 액세스할 수 있는 경우 사용자 및 인증서가 연결되어 있는지 확인합니다.

   a. **sss_cache** 유틸리티를 사용하여 **SSSD** 캐시의 **idm_user** 레코드를 무효화하고 **idm_user** 정보를 강제로 다시 로드합니다.

   ```
   # sss_cache -u idm_user
   ```

   b. **IdM** 사용자의 인증서가 포함된 파일 이름과 **ipa certmap-match** 명령을 실행합니다.

   ```
   # ipa certmap-match idm_user_cert.pem
   --------------
   1 user matched
   --------------
    Domain: IDM.EXAMPLE.COM
    User logins: idm_user
   ----------------------------
   Number of entries returned 1
   ----------------------------
   ```

   출력에 이제 **idm_user** 에 인증서 매핑 데이터가 추가되고 해당 매핑 규칙이 있는지 확인합니다. 즉, 정의된 인증서 매핑 데이터와 일치하는 모든 인증서를 사용하여 **idm_user** 로 인증할 수 있습니다.

/include::modules/identity-management/proc_add-certmapdata-to-user.adoc[leveloffset=+1]

9.3. AD 사용자 항목에 전체 인증서가 포함된 사용자에 대한 인증서 매핑 구성

이 사용자 스토리에서는 **IdM** 배포가 **AD(Active Directory)**와 함께 신뢰할 수 있는 경우 **IdM**에서 인증서 매핑을 활성화하는 데 필요한 단계를 설명합니다. 이 사용자는 **AD**에 저장되고 **AD**의 사용자 항목에 전

체 인증서가 포함되어 있습니다.

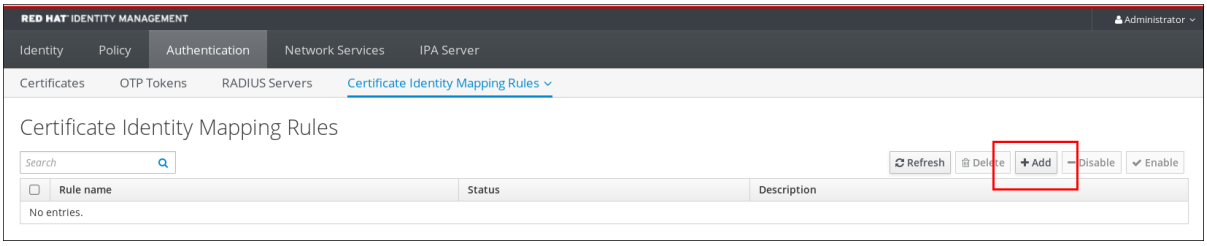
사전 요구 사항

- IdM에는 사용자에게 계정이 없습니다.
- 사용자에게 인증서가 포함된 AD의 계정이 있습니다.
- IdM 관리자는 IdM 인증서 매핑 규칙을 기반으로 하는 데이터에 액세스할 수 있습니다.

9.3.1. IdM 웹 UI에서 인증서 매핑 규칙 추가

1. 관리자로 IdM 웹 UI에 로그인합니다.
2. 인증 → 인증서 매핑 규칙 → 인증서 ID 매핑 규칙으로 이동합니다.
3. 추가를 클릭합니다.

그림 9.5. IdM 웹 UI에서 새 인증서 매핑 규칙 추가



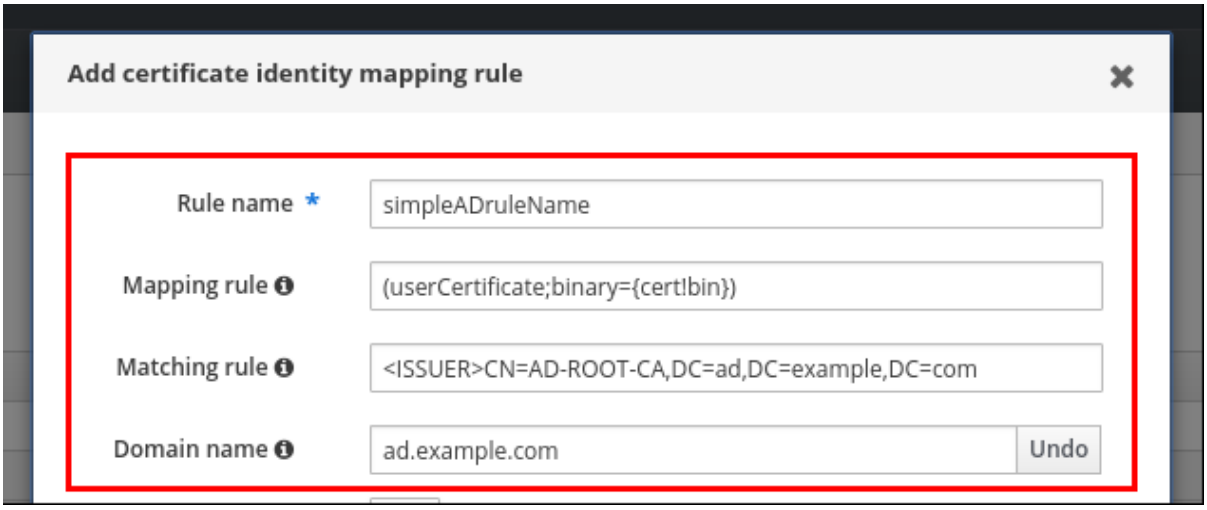
4. 규칙 이름을 입력합니다.
5. 매핑 규칙을 입력합니다. 인증을 위해 IdM에 제공되는 전체 인증서를 AD에서 사용할 수 있는 것과 비교하려면 다음을 수행합니다.

```
(userCertificate;binary={cert!bin})
```

6. 일치 규칙을 입력합니다. 예를 들어 AD.EXAMPLE.COM 도메인의 AD-ROOT-CA 에서 발급한 인증서만 허용하려면 다음을 수행합니다.

```
<ISSUER>CN=AD-ROOT-CA,DC=ad,DC=example,DC=com
```

그림 9.6. AD에 저장된 인증서가 있는 사용자의 인증서 매핑 규칙



7. 추가를 클릭합니다.

8. **SSSD(System Security Services Daemon)**는 주기적으로 인증서 매핑 규칙을 다시 읽습니다. 새로 생성된 규칙이 즉시 로드되도록 하려면 **CLI**에서 **SSSD**를 다시 시작합니다.

```
# systemctl restart sssd
```

9.3.2. IdM CLI에서 인증서 매핑 규칙 추가

1. 관리자의 자격 증명을 가져옵니다.

```
# kinit admin
```

2. 매핑 규칙을 입력하고 매핑 규칙을 기반으로 하는 일치 규칙을 입력합니다. **AD**에서 사용할 수 있는 것과 비교하여 인증을 위해 제공되는 전체 인증서를 보유하려면 **AD.EXAMPLE.COM** 도메인의 **AD-ROOT-CA** 에서 발급한 인증서만 허용합니다.

```
# ipa certmaprule-add simpleADrule --matchrule '<ISSUER>CN=AD-ROOT-
CA,DC=ad,DC=example,DC=com' --maprule '(userCertificate;binary={cert!bin})' --
domain ad.example.com
----------------------------------------------------
Added Certificate Identity Mapping Rule "simpleADrule"
----------------------------------------------------
  Rule name: simpleADrule
  Mapping rule: (userCertificate;binary={cert!bin})
  Matching rule: <ISSUER>CN=AD-ROOT-CA,DC=ad,DC=example,DC=com
  Domain name: ad.example.com
  Enabled: TRUE
```

3. **SSSD(System Security Services Daemon)**는 주기적으로 인증서 매핑 규칙을 다시 읽습니다. 새로 생성된 규칙이 즉시 로드되도록 하려면 **SSSD**를 다시 시작합니다.

```
# systemctl restart sssd
```

9.4. 사용자 인증서를 사용자 계정에 매핑하도록 AD가 구성된 경우 인증서 매핑 구성

이 사용자 스토리에서는 **IdM** 배포가 **AD(Active Directory)**와 함께 신뢰할 수 있는 경우 **IdM**에서 인증서 매핑을 활성화하는 데 필요한 단계를 설명하고 **AD**의 사용자 항목은 **AD**에 저장되고 **AD**의 사용자 항목에 인증서 매핑 데이터가 포함되어 있습니다.

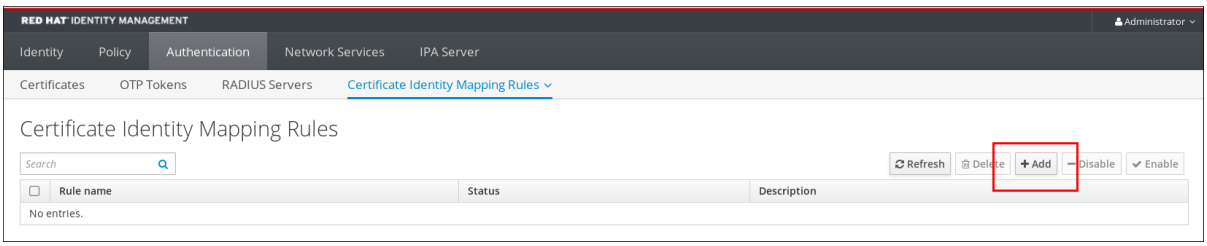
사전 요구 사항

- **IdM**에는 사용자에게 계정이 없습니다.
- 사용자에게 **AD**에는 **altSecurityIdentities** 속성이 포함된 계정이 있으며, **IdM certmapdata** 속성과 동등한 **AD**가 있습니다.
- **IdM** 관리자는 **IdM** 인증서 매핑 규칙을 기반으로 하는 데이터에 액세스할 수 있습니다.

9.4.1. IdM 웹 UI에서 인증서 매핑 규칙 추가

1. 관리자로 **IdM** 웹 **UI**에 로그인합니다.
2. 인증 → 인증서 매핑 규칙 → 인증서 **ID** 매핑 규칙으로 이동합니다.
3. 추가를 클릭합니다.

그림 9.7. IdM 웹 UI에서 새 인증서 매핑 규칙 추가



4. 규칙 이름을 입력합니다.

5. 매핑 규칙을 입력합니다. 예를 들어 **AD DC**가 제공된 인증서의 **Issuer** 및 **Subject** 항목을 검색하고 제공된 인증서의 두 항목에 있는 정보에 대해 인증하거나 그렇지 않은 경우 다음을 수행합니다.

```
(altSecurityIdentities=X509:<I>{issuer_dn!ad_x500}<S>{subject_dn!ad_x500})
```

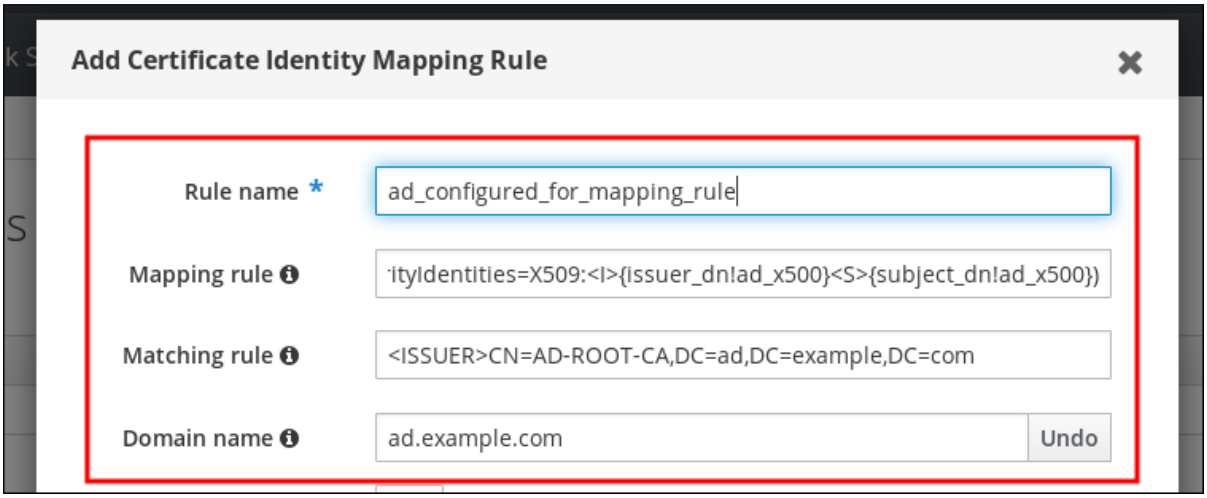
6. 일치 규칙을 입력합니다. 예를 들어 **AD.EXAMPLE.COM** 도메인의 **AD-ROOT-CA** 에서 발급한 인증서만 허용하도록 하여 **IdM**에 사용자를 인증합니다.

```
<ISSUER>CN=AD-ROOT-CA,DC=ad,DC=example,DC=com
```

7. 도메인을 입력합니다.

```
ad.example.com
```

그림 9.8. 매핑을 위해 AD가 구성된 경우 인증서 매핑 규칙



8. 추가를 클릭합니다.

9. **SSSD(System Security Services Daemon)**는 주기적으로 인증서 매핑 규칙을 다시 읽습니다. 새로 생성된 규칙이 즉시 로드되도록 하려면 **CLI**에서 **SSSD**를 다시 시작합니다.

```
# systemctl restart sssd
```

9.4.2. IdM CLI에서 인증서 매핑 규칙 추가

1. 관리자의 자격 증명을 가져옵니다.

```
# kinit admin
```

2. 매핑 규칙을 입력하고 매핑 규칙을 기반으로 하는 일치 규칙을 입력합니다. 예를 들어 AD가 제공된 인증서의 Issuer 및 Subject 항목을 검색하고 AD.EXAMPLE.COM 도메인의 AD-ROOT-CA 에 의해 발급된 인증서만 허용하려면 다음을 수행합니다.

```
# ipa certmaprule-add ad_configured_for_mapping_rule --matchrule
'<ISSUER>CN=AD-ROOT-CA,DC=ad,DC=example,DC=com' --maprule
'(altSecurityIdentities=X509:<I>{issuer_dn!ad_x500}<S>{subject_dn!ad_x500})' --
domain=ad.example.com
----------------------------------------------------
Added Certificate Identity Mapping Rule "ad_configured_for_mapping_rule"
----------------------------------------------------
  Rule name: ad_configured_for_mapping_rule
  Mapping rule: (altSecurityIdentities=X509:<I>{issuer_dn!ad_x500}<S>
{subject_dn!ad_x500})
  Matching rule: <ISSUER>CN=AD-ROOT-CA,DC=ad,DC=example,DC=com
  Domain name: ad.example.com
  Enabled: TRUE
```

3. SSSD(System Security Services Daemon)는 주기적으로 인증서 매핑 규칙을 다시 읽습니다. 새로 생성된 규칙이 즉시 로드되도록 하려면 SSSD를 다시 시작합니다.

```
# systemctl restart sssd
```

9.4.3. AD 측에서 인증서 매핑 데이터 확인

altSecurityIdentities 속성은 IdM의 certmapdata 사용자 특성과 동등한 Active Directory(AD)입니다. 사용자 인증서를 사용자 계정에 매핑하도록 신뢰할 수 있는 AD 도메인이 구성된 시나리오에서 IdM에서 인증서 매핑을 구성하는 경우 IdM 시스템 관리자가 AD의 사용자 항목에 altSecurityIdentities 특성이 올바르게 설정되어 있는지 확인해야 합니다.

AD에 AD에 저장된 사용자에 대한 올바른 정보가 포함되어 있는지 확인하려면 ldapsearch 명령을 사용하십시오.

- 예를 들어 아래 명령을 입력하여 다음 조건이 적용되는 adserver.ad.example.com 서버를 확인합니다.
  - altSecurityIdentities 속성은 ad_user 의 사용자 항목에 설정되어 있습니다.

- matchrule에서는 다음 조건이 적용되도록 지정합니다.
  - ad_user 가 AD에 인증하는 데 사용하는 인증서는 ad.example.com 도메인의 AD-ROOT-CA 에 의해 발행되었습니다.
  - 제목은 <S>DC=com,DC=example,DC=ad,CN=Users,CN=ad_user 입니다.

```
$ ldapsearch -o ldif-wrap=no -LLL -h adserver.ad.example.com \
-p 389 -D cn=Administrator,cn=users,dc=ad,dc=example,dc=com \
-W -b cn=users,dc=ad,dc=example,dc=com "(cn=ad_user)" \
altSecurityIdentities
Enter LDAP Password:
dn: CN=ad_user,CN=Users,DC=ad,DC=example,DC=com
altSecurityIdentities: X509:<I>DC=com,DC=example,DC=ad,CN=AD-ROOT-
CA<S>DC=com,DC=example,DC=ad,CN=Users,CN=ad_user
```

9.5. AD 사용자 항목에 인증서 또는 매핑 데이터가 없는 경우 인증서 매핑 구성

이 사용자 스토리에서는 IdM 배포가 Active Directory(AD)를 사용하여 신뢰할 수 있는 경우 IdM에서 인증서 매핑을 활성화하는 데 필요한 단계를 설명하고, AD에 사용자가 저장되는 경우 전체 인증서나 인증서 매핑 데이터가 포함되어 있지 않습니다.

사전 요구 사항

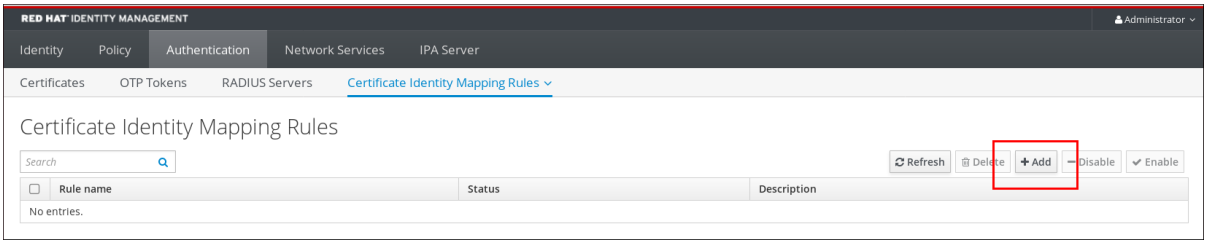
- IdM에는 사용자에게 계정이 없습니다.
- AD에는 전체 인증서나 altSecurityIdentities 특성을 포함하지 않는 계정이 있으며, IdM certmapdata 속성과 동등한 AD가 있습니다.
- IdM 관리자에게는 IdM에서 AD 사용자의 사용자 ID 재정의에 추가할 전체 AD 사용자 인증서가 있습니다.

9.5.1. IdM 웹 UI에서 인증서 매핑 규칙 추가

1. 관리자로 IdM 웹 UI에 로그인합니다.
2. 인증 → 인증서 매핑 규칙 → 인증서 ID 매핑 규칙으로 이동합니다.

3. 추가를 클릭합니다.

그림 9.9. IdM 웹 UI에서 새 인증서 매핑 규칙 추가



4. 규칙 이름을 입력합니다.

5. 매핑 규칙을 입력합니다. IdM 사용자 항목의 사용자 ID 덮어쓰기 항목에 저장된 인증서와 비교하여 인증을 위해 IdM에 제공되는 전체 인증서를 보유하려면 다음을 수행합니다.

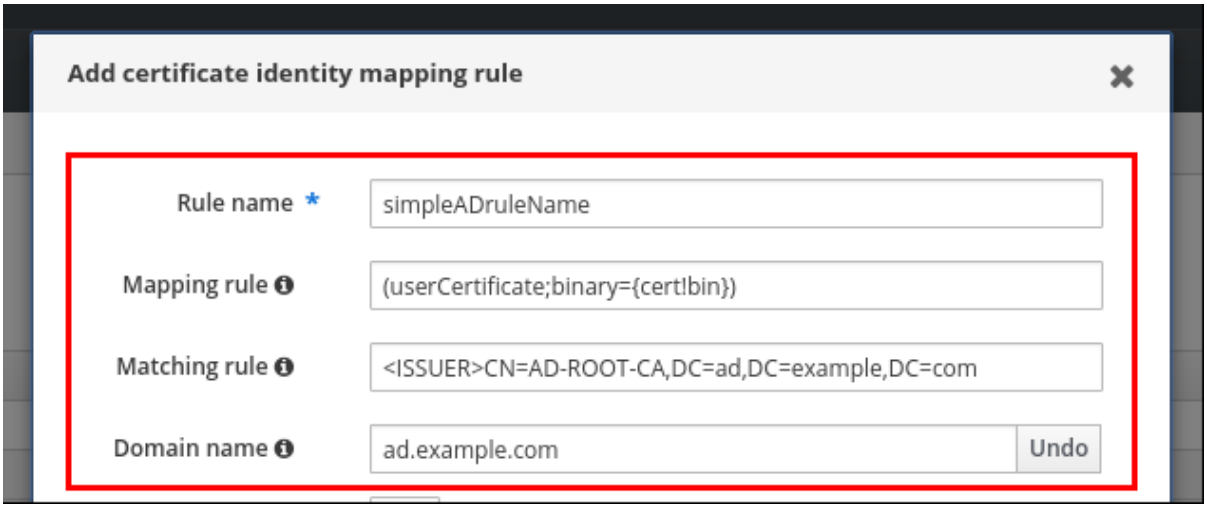
```
(userCertificate;binary={cert!bin})
```

6. 일치 규칙을 입력합니다. 예를 들어 AD.EXAMPLE.COM 도메인의 AD-ROOT-CA 에서 발급한 인증서만 허용하려면 다음을 수행합니다.

```
<ISSUER>CN=AD-ROOT-CA,DC=ad,DC=example,DC=com
```

7. 도메인 이름을 입력합니다. 예를 들어 ad.example.com 도메인에서 사용자를 검색하려면 다음을 수행합니다.

그림 9.10. 인증서가 없거나 AD에 저장된 데이터를 매핑하는 사용자의 인증서 매핑 규칙



8. 추가를 클릭합니다.

9. SSSD(System Security Services Daemon)는 주기적으로 인증서 매핑 규칙을 다시 읽습니다. 새로 생성된 규칙이 즉시 로드되도록 하려면 CLI에서 SSSD를 다시 시작합니다.

```
# systemctl restart sssd
```

9.5.2. IdM CLI에서 인증서 매핑 규칙 추가

1. 관리자의 자격 증명을 가져옵니다.

```
# kinit admin
```

2. 매핑 규칙을 입력하고 매핑 규칙을 기반으로 하는 일치 규칙을 입력합니다. IdM의 AD 사용자 항목에 저장된 사용자 ID 재정의 항목에 비해 인증을 위해 제공되는 전체 인증서를 보유하려면 AD.EXAMPLE.COM 도메인의 AD-ROOT-CA 에서 발급한 인증서만 허용합니다.

```
# ipa certmaprule-add simpleADrule --matchrule '<ISSUER>CN=AD-ROOT-
CA,DC=ad,DC=example,DC=com' --maprule '(userCertificate;binary={cert!bin})' --
domain ad.example.com
----------------------------------------------------
Added Certificate Identity Mapping Rule "simpleADrule"
----------------------------------------------------
  Rule name: simpleADrule
  Mapping rule: (userCertificate;binary={cert!bin})
  Matching rule: <ISSUER>CN=AD-ROOT-CA,DC=ad,DC=example,DC=com
  Domain name: ad.example.com
  Enabled: TRUE
```

3. SSSD(System Security Services Daemon)는 주기적으로 인증서 매핑 규칙을 다시 읽습니다. 새로 생성된 규칙이 즉시 로드되도록 하려면 SSSD를 다시 시작합니다.

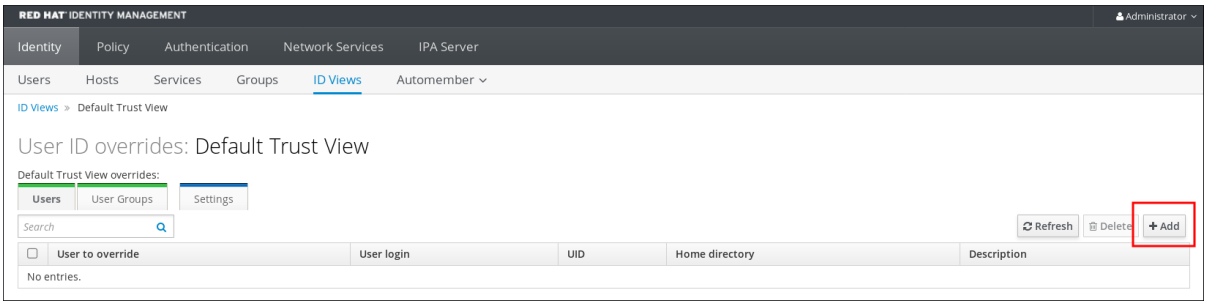
```
# systemctl restart sssd
```

9.5.3. IdM 웹 UI에서 AD 사용자 ID 재정의에 인증서 추가

1. Identity → ID Views → Default Trust View 로 이동합니다.

2. 추가를 클릭합니다.

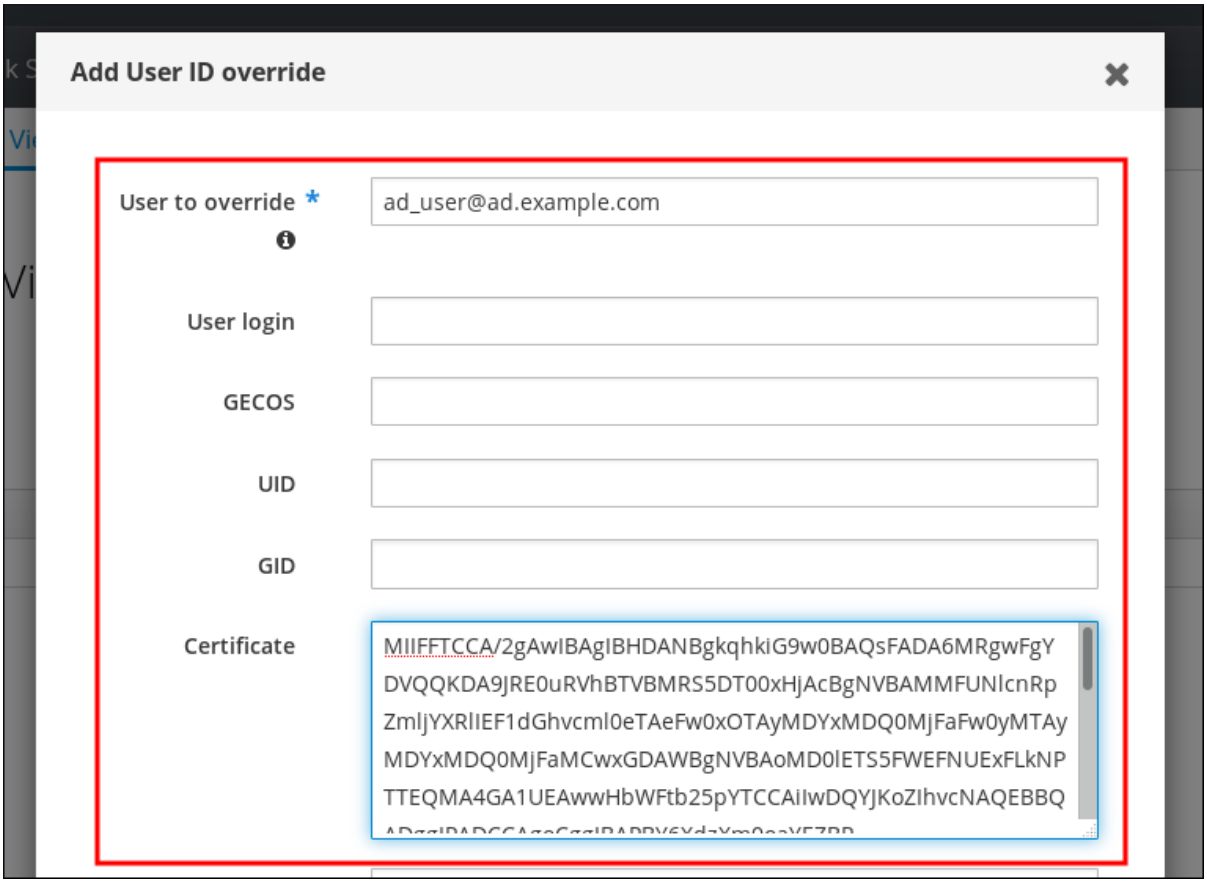
그림 9.11. IdM 웹 UI에 새 사용자 ID 덮어쓰기 추가



3. User to override 필드에 ad_user@ad.example.com 을 입력합니다.

4. ad_user 인증서를 복사하여 인증서 필드에 붙여넣습니다.

그림 9.12. AD 사용자의 사용자 ID 덮어쓰기 구성



5. 추가를 클릭합니다.

검증 단계

- 사용자 및 인증서가 연결되었는지 확인합니다.
  -

sss_cache 유틸리티를 사용하여 SSSD 캐시에서 ad_user@ad.example.com 레코드를 무효화하고 ad_user@ad.example.com 정보를 다시 로드하도록 합니다.

```
# sss_cache -u ad_user@ad.example.com
```

- AD 사용자의 인증서가 포함된 파일 이름으로 ipa certmap-match 명령을 실행합니다.

```
# ipa certmap-match ad_user_cert.pem
--------------
1 user matched
--------------
 Domain: AD.EXAMPLE.COM
 User logins: ad_user@ad.example.com
----------------------------
Number of entries returned 1
----------------------------
```

출력은 ad_user@ad.example.com 에 인증서 매핑 데이터가 추가되었음을 확인하고 AD 사용자 항목에 인증서 또는 매핑 데이터가 없는 경우 인증서 매핑 규칙 추가에 정의된 해당 매핑 규칙이 있는지 확인합니다. 즉, 정의된 인증서 매핑 데이터와 일치하는 모든 인증서를 사용하여 ad_user@ad.example.com 로 인증할 수 있습니다.

9.5.4. IdM CLI에서 AD 사용자 ID 재정의에 인증서 추가

추가 리소스

1. 관리자의 자격 증명을 가져옵니다.

```
# kinit admin
```

2. CERT 라는 새 변수에 인증서 Blob을 저장합니다.

```
# CERT=`cat ad_user_cert.pem | tail -n +2| head -n -1 | tr -d '\r\n'\`
```

3. ipa idoverrideuser-add-cert 명령을 사용하여 사용자 계정에 ad_user@ad.example.com 의 인증서를 추가합니다.

```
# ipa idoverrideuser-add-cert ad_user@ad.example.com --certificate $CERT
```

검증 단계

- 사용자 및 인증서가 연결되었는지 확인합니다.
  - **sss_cache** 유틸리티를 사용하여 **SSSD** 캐시에서 **ad_user@ad.example.com** 레코드를 무효화하고 **ad_user@ad.example.com** 정보를 다시 로드하도록 합니다.

```
# sss_cache -u ad_user@ad.example.com
```

  - **AD** 사용자의 인증서가 포함된 파일 이름으로 **ipa certmap-match** 명령을 실행합니다.

```
# ipa certmap-match ad_user_cert.pem
--------------
1 user matched
--------------
 Domain: AD.EXAMPLE.COM
 User logins: ad_user@ad.example.com
----------------------------
Number of entries returned 1
----------------------------
```

출력은 **ad_user@ad.example.com** 에 인증서 매핑 데이터가 추가되었음을 확인하고 **AD** 사용자 항목에 인증서 또는 매핑 데이터가 없는 경우 인증서 매핑 규칙 추가에 정의된 해당 매핑 규칙이 있는지 확인합니다. 즉, 정의된 인증서 매핑 데이터와 일치하는 모든 인증서를 사용하여 **ad_user@ad.example.com** 로 인증할 수 있습니다.

9.6. 여러 ID 매핑 규칙을 하나로 결합

추가 리소스

여러 **ID** 매핑 규칙을 하나의 결합된 규칙으로 결합하려면 **|** (또는) 문자를 사용하여 개별 매핑 규칙 앞에 배치하고 () 대괄호를 사용하여 구분합니다. 예를 들면 다음과 같습니다.

인증서 매핑 필터 예 **1**

```
$ ipa certmaprule-add ad_cert_for_ipa_and_ad_users \ --maprule='(|(ipacertmapdata=X509:<I>
{issuer_dn!nss_x500}<S>{subject_dn!nss_x500})(altSecurityIdentities=X509:<I>
{issuer_dn!ad_x500}<S>{subject_dn!ad_x500}))' \ --matchrule='<ISSUER>CN=AD-ROOT-
CA,DC=ad,DC=example,DC=com' \ --domain=ad.example.com
```

위 예에서 **--maprule** 옵션의 필터 정의에는 다음과 같은 기준이 포함됩니다.

- **ipacertmapdata=X509:<I>{issuer_dn!nss_x500}<S>{subject_dn!nss_x500}** 는 스마트 카드 인증서의 제목 및 발급자를 **IdM** 사용자 계정의 **ipacertmapdata** 속성 값으로 연결하는 필터입니다. https://access.redhat.com/documentation/en-us/red_hat_enterprise_linux/8/html/configuring_and_managing_identity_management/conf-certmap-idm_configuring-and-managing-idm#proc-add-maprule_conf-certmap-for-users-in-idm

- **altSecurityIdentities=X509:<I>{issuer_dn!ad_x500}<S>{subject_dn!ad_x500}** 는 스마트 카드 인증서의 주체 및 발급자를 **AD** 사용자 계정의 **altSecurityIdentities** 속성 값에 연결하는 필터입니다.

- **--domain=ad.example.com** 옵션을 추가하면 지정된 인증서에 매핑된 사용자가 로컬 **idm.example.com** 도메인에서 검색될 뿐만 아니라 **ad.example.com** 도메인에서도 검색됩니다.

--maprule 옵션의 필터 정의에서는 여러 조건을 지정할 수 있도록 논리 연산자 **|** (또는)를 허용합니다. 이 경우 규칙은 조건 중 하나 이상을 충족하는 모든 사용자 계정을 매핑합니다.

인증서 매핑 필터 예 **2**

```
$ ipa certmaprule-add ipa_cert_for_ad_users \
  --maprule='(|(userCertificate;binary={cert!bin})(ipacertmapdata=X509:<I>
{issuer_dn!nss_x500}<S>{subject_dn!nss_x500})(altSecurityIdentities=X509:<I>
{issuer_dn!ad_x500}<S>{subject_dn!ad_x500}))' \
  --matchrule='<ISSUER>CN=Certificate Authority,O=REALM.EXAMPLE.COM' \
  --domain=idm.example.com --domain=ad.example.com
```

위 예에서 **--maprule** 옵션의 필터 정의에는 다음과 같은 기준이 포함됩니다.

- **userCertificate;binary={cert!bin}** 은 전체 인증서가 포함된 사용자 항목을 반환하는 필터입니다. **AD** 사용자의 경우 **AD** 사용자 항목에 인증서가 없거나 데이터를 매핑하지 않은 경우 이러한 유형의 필터를 만드는 것은 인증서 매핑 규칙 추가 에서 자세히 설명되어 있습니다.

- **ipacertmapdata=X509:<I>{issuer_dn!nss_x500}<S>{subject_dn!nss_x500}** 는 스마트 카

드 인증서의 제목 및 발급자를 IdM 사용자 계정의 ipacertmapdata 속성 값으로 연결하는 필터입니다. https://access.redhat.com/documentation/en-us/red_hat_enterprise_linux/8/html/configuring_and_managing_identity_management/conf-certmap-idm_configuring-and-managing-idm#proc-add-maprule_conf-certmap-for-users-in-idm

- altSecurityIdentities=X509:<I>{issuer_dn!ad_x500}<S>{subject_dn!ad_x500} 는 스마트 카드 인증서의 주체 및 발급자를 AD 사용자 계정의 altSecurityIdentities 속성 값에 연결하는 필터입니다.

--maprule 옵션의 필터 정의에서는 여러 조건을 지정할 수 있도록 논리 연산자 | (또는)를 허용합니다. 이 경우 규칙은 조건 중 하나 이상을 충족하는 모든 사용자 계정을 매핑합니다.

10장. IDM 클라이언트의 데스크탑에 저장된 인증서로 인증 구성

IdM(Identity Management)을 구성하면 IdM 시스템 관리자는 사용자가 인증 기관(CA)이 사용자에게 발행한 인증서를 사용하여 IdM 웹 UI 및 CLI(명령줄 인터페이스)에 인증할 수 있습니다.

웹 브라우저는 IdM 도메인에 포함되지 않은 시스템에서 실행할 수 있습니다.

이 사용자 스토리는 IdM 클라이언트의 데스크탑에 저장된 인증서를 사용하여 Identity Management 웹 UI 및 CLI에 대한 로그인을 효과적으로 구성하고 테스트하는 방법에 대한 지침을 제공합니다. 이 사용자 정보를 사용하여,

- 인증서를 사용하여 인증하려는 사용자가 이미 인증서를 사용하여 인증하려는 경우 새 사용자 인증서 요청 및 클라이언트로 내보내기 를 건너뛸 수 있습니다.
- IdM CA 에서 사용자 인증서를 발급한 경우 인증서와 사용자가 함께 연결되어 있는지 건너뛸 수 있습니다.

참고

Identity Management 사용자만 인증서를 사용하여 웹 UI에 로그인할 수 있습니다. Active Directory 사용자는 사용자 이름 및 암호를 사용하여 로그인할 수 있습니다.

10.1. 웹 UI에서 인증서 인증을 위한 ID 관리 서버 구성

IdM(Identity Management) 관리자는 사용자가 인증서를 사용하여 IdM 환경에 인증할 수 있도록 허용할 수 있습니다.

절차

ID 관리 관리자로 다음을 수행합니다.

1. Identity Management 서버에서 관리자 권한을 얻고 쉘 스크립트를 생성하여 서버를 구성합니다.
   a. ipa-advise config-server-for-smart-card-auth 명령을 실행하고 출력을 파일에 저장합니다(예: server_certificate_script.sh):

```
# kinit admin
# ipa-advise config-server-for-smart-card-auth > server_certificate_script.sh
```

b. **CronJob** 유틸리티를 사용하여 파일에 실행 권한을 추가합니다.

```
# chmod +x server_certificate_script.sh
```

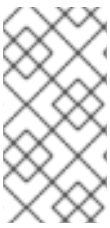
2. **Identity Management** 도메인의 모든 서버에서 **server_certificate_script.sh** 스크립트를 실행합니다.

a. **IdM** 인증 기관 인증서의 경로를 사용하여 **IdM CA**가 다음을 위해 인증서 인증을 활성화하려는 사용자의 인증서를 발급한 유일한 인증 기관인 경우 입력으로 **/etc/ipa/ca.crt.crt**를 입력합니다.

```
# ./server_certificate_script.sh /etc/ipa/ca.crt
```

b. 다른 외부 **CA**가 에 대한 인증서 인증을 활성화하려는 사용자의 인증서에 서명한 경우 관련 **CA** 인증서를 입력으로 전달되는 경로를 사용합니다.

```
# ./server_certificate_script.sh /tmp/ca1.pem /tmp/ca2.pem
```



참고

전체 토폴로지에서 활성화된 사용자에 대한 인증서 인증이 있어야 하는 경우 나중에 시스템에 추가하는 각 새 복제본에서 스크립트를 실행해야 합니다.

10.2. 새 사용자 인증서를 요청하고 클라이언트로 내보내기

IdM(Identity Management) 관리자는 **IdM** 환경에서 사용자에 대한 인증서를 생성하고 사용자에 대한 인증서 인증을 활성화하려는 **IdM** 클라이언트에 내보낼 수 있습니다.

참고

인증서를 사용하여 인증하려는 사용자가 이미 인증서가 있는 경우 이 섹션을 건너뛸 수 있습니다.

절차

1. 선택적으로 새 디렉터리(예: ~/certdb/)를 만들고 임시 인증서 데이터베이스로 만듭니다. 메시지가 표시되면 NSS 인증서 DB 암호를 생성하여 후속 단계에서 생성할 인증서의 키를 암호화하십시오.

```
# mkdir ~/certdb/
# certutil -N -d ~/certdb/
Enter a password which will be used to encrypt your keys.
The password should be at least 8 characters long,
and should contain at least one non-alphabetic character.

Enter new password:
Re-enter password:
```

2. CSR(인증서 서명 요청)을 생성하고 출력을 파일로 리디렉션합니다. 예를 들어 IDM.EXAMPLE.COM 영역에서 idm_user 사용자에 대한 4096 비트 인증서에 대한 certificate_request.csr 이라는 이름으로 CSR을 생성하려면 인증서 개인 키의 nickname을 idm_user 로 설정하고 주체를 CN=id_user,OID_user,=OIDM.AMPLE :COM :COM :COM 영역에서 설정할 수 있습니다.

```
# certutil -R -d ~/certdb/ -a -g 4096 -n idm_user -s
"CN=idm_user,O=IDM.EXAMPLE.COM" > certificate_request.csr
```

3. 메시지가 표시되면 certutil 을 사용하여 임시 데이터베이스를 생성할 때 입력한 것과 동일한 암호를 입력합니다. 그런 다음 중지하도록 지시할 때까지 randlomly를 계속 입력합니다.

```
Enter Password or Pin for "NSS Certificate DB":

A random seed must be generated that will be used in the
creation of your key.  One of the easiest ways to create a
random seed is to use the timing of keystrokes on a keyboard.

To begin, type keys on the keyboard until this progress meter
is full.  DO NOT USE THE AUTOREPEAT FUNCTION ON YOUR KEYBOARD!


Continue typing until the progress meter is full:
```

4. 인증서 요청 파일을 서버에 제출합니다. 새로 발행된 인증서, 인증서를 저장할 출력 파일, 인증서 프로필을 선택적으로 연결할 Kerberos 주체를 지정합니다. 예를 들어 IECUserRoles 프로필의 인증서를 가져오려면 idm_user@IDM.EXAMPLE.COM 주체에 대해 사용자 역할 확장자가 추가된 프로파일, ~/idm_user.pem 파일에 저장합니다.

```
# ipa cert-request certificate_request.csr --principal=idm_user@IDM.EXAMPLE.COM --
profile-id=IECUserRoles --certificate-out=~/idm_user.pem
```

5.

NSS 데이터베이스에 인증서를 추가합니다. 인증서가 NSS 데이터베이스의 개인 키와 일치하도록 이전에 CSR을 생성할 때 사용한 것과 동일한 닉네임을 설정하려면 -n 옵션을 사용합니다. -t 옵션은 신뢰 수준을 설정합니다. 자세한 내용은 certutil(1) 매뉴얼 페이지를 참조하십시오. i 옵션은 입력 인증서 파일을 지정합니다. 예를 들어, NSS 데이터베이스에 추가하려면 ~/certdb/ 데이터베이스의 ~/ idm_user.pem 파일에 저장된 idm_user nickname이 있는 인증서를 사용합니다.

```
# certutil -A -d ~/certdb/ -n idm_user -t "P,," -i ~/idm_user.pem
```

6. NSS 데이터베이스의 키가 별 이름으로 표시되지 않는지 확인합니다. 예를 들어 ~/certdb/ 데이터베이스에 저장된 인증서가 분리되지 않았는지 확인하려면 다음을 수행합니다.

```
# certutil -K -d ~/certdb/
< 0> rsa      5ad14d41463b87a095b1896cf0068ccc467df395   NSS Certificate
DB:idm_user
```

7. pk12util 명령을 사용하여 NSS 데이터베이스에서 PKCS12 형식으로 인증서를 내보냅니다. 예를 들어 /root/certdb NSS 데이터베이스에서 idm_user nickname이 있는 인증서를 ~/idm_user.p12 파일로 내보내려면 다음을 실행합니다.

```
# pk12util -d ~/certdb -o ~/idm_user.p12 -n idm_user
Enter Password or Pin for "NSS Certificate DB":
Enter password for PKCS12 file:
Re-enter password:
pk12util: PKCS12 EXPORT SUCCESSFUL
```

8. idm_user 의 인증서 인증을 활성화할 호스트로 인증서를 전송합니다.

```
# scp ~/idm_user.p12 idm_user@client.idm.example.com:/home/idm_user/
```

9. 인증서가 전송된 호스트에서 .pkcs12 파일이 보안상의 이유로 'other' 그룹에 액세스할 수 없는 디렉터리로 설정합니다.

```
# chmod o-rwx /home/idm_user/
```

10. 보안상의 이유로 서버에서 임시 NSS 데이터베이스와 .pkcs12 파일을 제거하십시오.

```
# rm ~/certdb/
# rm ~/idm_user.p12
```

10.3. 인증서와 사용자가 연결되어 있는지 확인

참고

IdM CA에서 사용자 인증서를 발급한 경우 이 섹션을 건너뛸 수 있습니다.

인증서 인증이 작동하려면 IdM(Identity Management)에 인증하는 데 인증서를 사용할 사용자에 대한 인증서가 연결되어 있는지 확인해야 합니다.

- ID 관리 환경에 포함되지 않은 인증 기관에서 인증서를 제공하는 경우 사용자 계정 연결 프로세스에 설명된 절차에 따라 사용자와 인증서를 연결합니다.
- ID 관리 CA에서 인증서를 제공하면 인증서가 사용자 항목에 자동으로 추가되고 인증서를 사용자 계정에 연결할 필요가 없습니다. IdM에 새 인증서를 생성하는 방법에 대한 자세한 내용은 새 사용자 인증서 요청 및 클라이언트로 내보내기를 참조하십시오.

10.4. 인증서 인증을 사용하도록 브라우저 구성

WebUI를 사용하여 IdM(Identity Management)에 로그인할 때 인증서로 인증하려면 사용자 및 관련 CA(인증 기관) 인증서를 Mozilla Firefox 또는 Google Chrome 브라우저로 가져와야 합니다. 브라우저가 실행 중인 호스트 자체는 IdM 도메인의 일부가 될 필요가 없습니다.

IdM은 WebUI에 연결하기 위해 다음 브라우저를 지원합니다.

- Mozilla Firefox 38 이상
- Google Chrome 46 이상

다음 절차는 Mozilla Firefox 57.0.1 브라우저를 구성하는 방법을 보여줍니다.

사전 요구 사항

- PKCS#12 형식으로 사용 중인 브라우저로 가져오려는 사용자 인증서가 있습니다.

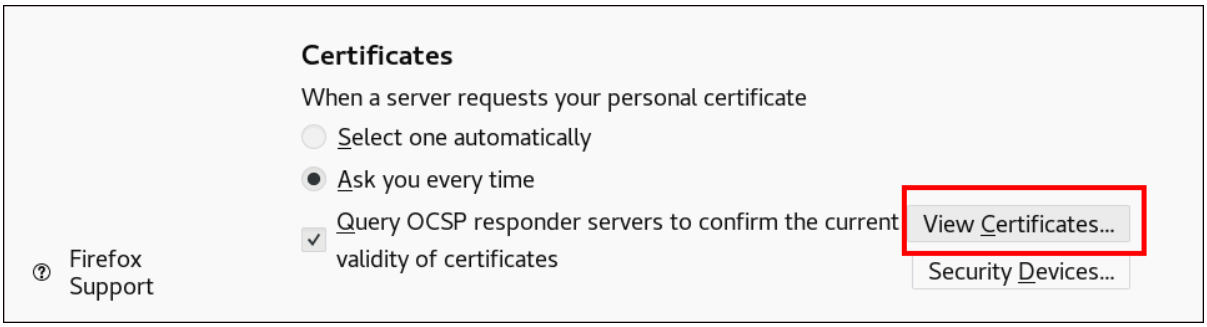
절차

1. **Firefox**를 연 다음 기본 설정 → 개인 정보 보호 및 보안 으로 이동합니다.

그림 10.1. 환경 설정의 개인 정보 보호 및 보안 섹션

2. 인증서 보기를 클릭합니다.

그림 10.2. 개인 정보 보호 및 보안에서 인증서 보기



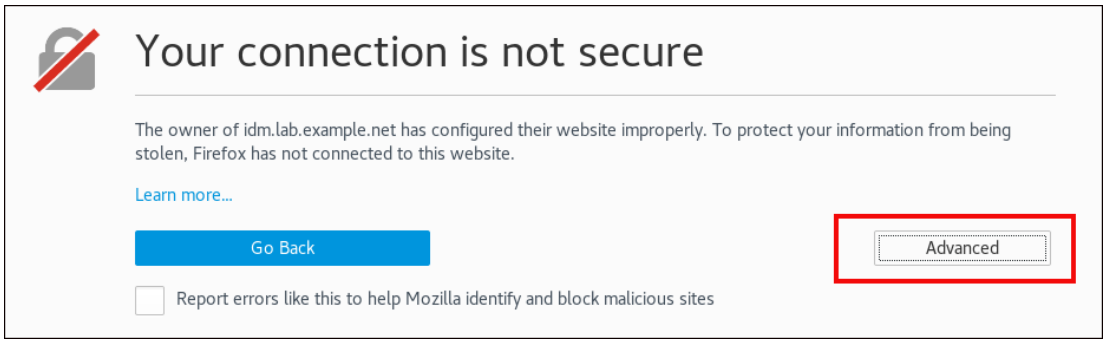
3. 사용자 인증서 탭에서 가져오기 를 클릭합니다. **PKCS12** 형식으로 사용자의 인증서를 찾아 연 다음 **OK** 및 **OK** 를 클릭합니다.

4. **ID** 관리 인증 기관에서 **Firefox**에서 신뢰할 수 있는 기관으로 인식했는지 확인합니다.

   a. **IdM CA** 인증서를 로컬에 저장합니다.

   - **Firefox** 주소 표시줄에 **IdM** 서버 이름을 작성하여 **IdM** 웹 **UI**로 이동합니다. 보안 연결 경고 페이지에서 고급 을 클릭합니다.

그림 10.3. 비보안 연결

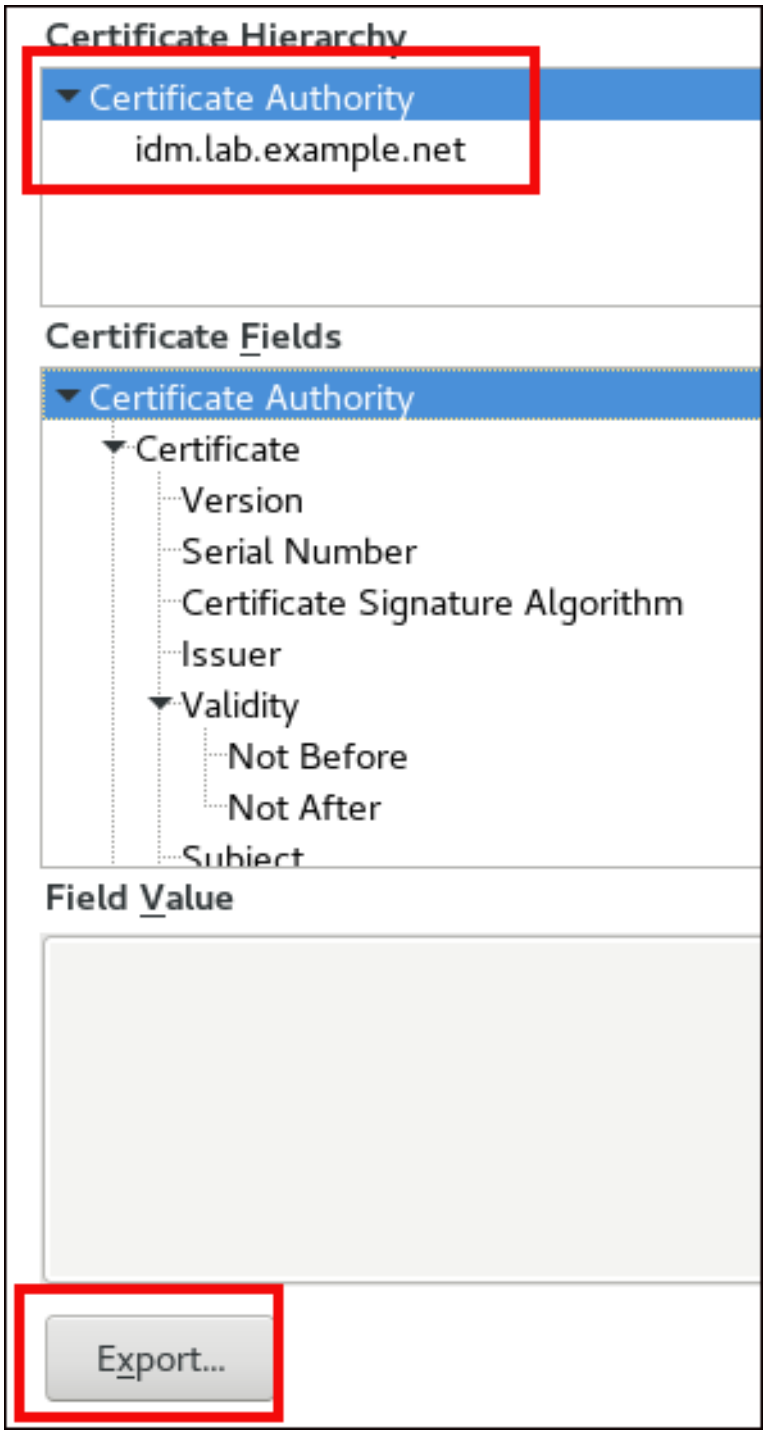


- **Exception**을 추가합니다. 보기를 클릭합니다.

그림 10.4. 인증서 세부 정보 보기

- **Details** (세부 정보) 탭에서 **Certificate Authority** 필드를 강조 표시합니다.

그림 10.5. CA 인증서 내보내기



- Export 를 클릭합니다. 예를 들어 CA 인증서를 CertificateAuthority.crt 파일로 저장한 다음 닫기 를 클릭하고 취소 를 클릭합니다.

b. IdM CA 인증서를 신뢰할 수 있는 인증 기관 인증서로 Firefox로 가져옵니다.

- Firefox를 열고 기본 설정으로 이동하여 개인 정보 보호 및 보안을 클릭합니다.

그림 10.6. 환경 설정의 개인 정보 보호 및 보안 섹션

- 인증서 보기를 클릭합니다.

그림 10.7. 개인 정보 보호 및 보안에서 인증서 보기

- **Authorities** (권한) 탭에서 가져오기 를 클릭합니다. **CertificateAuthority.crt** 파일에서 이전 단계에서 저장한 **CA** 인증서를 찾아 엽니다. 인증서를 신뢰하여 웹 사이트를 확인한 다음 확인 및 확인 을 클릭합니다.

5. **ID** 관리 사용자로 인증서를 사용하여 **ID** 관리 웹 **UI**를 계속 인증하십시오.

10.5. 인증서를 ID 관리 사용자로 사용하여 ID 관리 웹 UI에 인증

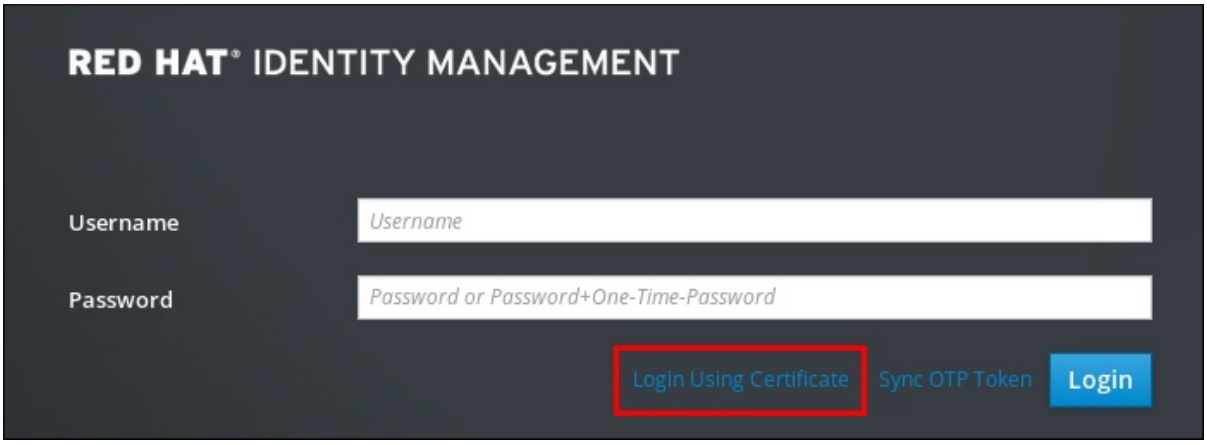
이 절차에서는 **Identity Management** 클라이언트의 데스크탑에 저장된 인증서를 사용하여 **IdM(Identity Management)** 웹 **UI**에 대한 사용자로 인증하는 방법을 설명합니다.

절차

1. 브라우저에서 의 **Identity Management** 웹 **UI**로 이동합니다(예: **https://server.idm.example.com/ipa/ui**).

2. 인증서를 사용하여 로그인을 클릭합니다.

. Identity Management 웹 UI에서인증서를 사용하여 로그인



3. 사용자의 인증서가 이미 선택되어 있어야 합니다. **Remember this decision** (이 결정에 기억)를 선택 취소한 다음 확인 을 클릭합니다.

이제 인증서에 해당하는 사용자로 인증됩니다.

추가 리소스

- 스마트 카드 인증을 위한 ID 관리 구성을 참조하십시오.

10.6. 인증서를 사용하여 CLI에 인증할 수 있도록 IDM 클라이언트 구성

IdM 클라이언트의 CLI(명령줄 인터페이스)에서 IdM 사용자에 대한 인증서 인증을 수행하려면 IdM 사용자 인증서 및 개인 키를 IdM 클라이언트로 가져옵니다. 사용자 인증서 생성 및 전송에 대한 자세한 내용은 새 사용자 인증서 요청 및 클라이언트로 내보내기를 참조하십시오.

절차

- IdM 클라이언트에 로그인하고 사용자의 인증서와 개인 키가 포함된 .p12 파일이 준비되었는지 확인합니다. TGT(Kerberos 티켓 부여)를 가져와서 캐시하려면 X509_username:/path/to/file.p12 속성과 함께 -X 옵션을 사용하여 사용자 주체와 함께 kinit 명령을 실행하여 사용자의 X509 ID 정보를 찾을 위치를 지정합니다. 예를 들어 ~/ idm_user.p12 파일에 저장된 사용자의 ID 정보를 사용하여 idm_user에 대한 TGT를 가져오려면 다음을 수행합니다.

```
$ kinit -X X509_idm_user='PKCS12:~/idm_user.p12' idm_user
```

참고

이 명령은 .pem 파일 형식도 지원합니다. kinit -X X509_username='FILE:/path/to/cert.pem,/path/to/key' user_principal

11장. IDM CA 갱신 서버 사용

11.1. IDM CA 갱신 서버 설명

CA(인증 기관)를 사용하는 IdM(Identity Management) 배포에서 CA 갱신 서버는 IdM 시스템 인증서를 유지 관리하고 갱신합니다. 강력한 IdM 배포를 보장합니다.

IdM 시스템 인증서는 다음과 같습니다.

- IdM CA 인증서
- OCSP 서명 인증서
- IdM CA 하위 시스템 인증서
- IdM CA 감사 서명 인증서
- IdM 갱신 에이전트 (RA) 인증서
- KRA 전송 및 스토리지 인증서

시스템 인증서를 특성화하여 모든 CA 복제본에서 키를 공유하는 것은 무엇입니까. 반면 IdM 서비스 인증서(예: LDAP,HTTP 및 PKINIT 인증서)에는 서로 다른 IdM CA 서버에서 키 쌍과 제목 이름이 다릅니다.

IdM 토폴로지에서 기본적으로 첫 번째 IdM CA 서버는 CA 갱신 서버입니다.

참고

업스트림 문서에서 IdM CA는 Dogtag 라고 합니다.

CA 갱신 서버의 역할

IdM CA,IdM CA 하위 시스템, IdM RA 인증서는 IdM 배포에 중요합니다. 각 인증서는 /etc/pki/pki-tomcat/ 디렉터리의 NSS 데이터베이스에 저장되고 LDAP 데이터베이스 항목으로도 저장됩니다. LDAP에 저장된 인증서는 NSS 데이터베이스에 저장된 인증서와 일치해야 합니다. 일치하지 않는 경우 IdM 프레임워크와 IdM CA와 IdM CA 간 인증 오류가 발생합니다.

모든 IdM CA 복제본은 모든 시스템 인증서에 대한 요청을 추적합니다. 통합 CA를 사용한 IdM 배포에 CA 갱신 서버가 없는 경우 각 IdM CA 서버에서 독립적으로 시스템 인증서 갱신을 요청합니다. 이로 인해 서로 다른 CA 복제본이 다양한 시스템 인증서 및 인증 오류가 발생합니다.

갱신 서버로 하나의 CA 복제본을 추가하면 필요에 따라 시스템 인증서를 정확하게 갱신할 수 있으므로 인증 오류가 발생하지 않습니다.

CA 복제본에서 certmonger 서비스의 역할

모든 IdM CA 복제본에서 실행되는 certmonger 서비스는 dogtag-ipa-ca-renew-agent 업데이트 도우미를 사용하여 IdM 시스템 인증서를 추적합니다. 갱신 도우미 프로그램은 CA 갱신 서버 구성을 읽습니다. CA 갱신 서버가 아닌 각 CA 복제본에서 갱신 도우미는 ca_renewal LDAP 항목에서 최신 시스템 인증서를 검색합니다. 정확히 certmonger 갱신 시도가 발생할 경우 CA 갱신 서버가 실제로 인증서를 갱신하기 전에 시스템 인증서를 업데이트하려고 하는 경우가 있음에 따라ts tag-ipa-ca-renew-agent 도우미가 시스템 인증서를 업데이트하려고 할 수 있습니다. 이 경우 이전 인증서가 즉시 CA 복제본의 certmonger 서비스로 반환됩니다. certmonger 서비스는 데이터베이스에 이미 저장된 인증서와 동일한 인증서이며 CA 갱신 서버에서 업데이트된 인증서를 검색할 수 있을 때까지 개별 시도 사이에 특정 지연으로 인증서를 업데이트하려고 합니다.

IdM CA 갱신 서버의 올바른 기능

CA가 포함된 IdM 배포는 IdM CA와 함께 설치되었거나 IdM CA 서버가 나중에 설치된 IdM 배포입니다. 포함된 CA를 사용하는 IdM 배포에는 항상 갱신 서버로 구성된 정확히 하나의 CA 복제본이 있어야 합니다. 갱신 서버는 온라인이고 완전히 작동해야 하며 다른 서버와 함께 제대로 복제해야 합니다.

ipa server-del,ipa-replica-manage del,ipa-csreplica-manage del 또는 ipa-server-install --uninstall 명령을 사용하여 현재 CA 갱신 서버가 삭제되면 다른 CA 복제본이 CA 갱신 서버로 자동으로 할당됩니다. 이 정책은 갱신 서버 구성이 유효한 상태로 유지됩니다.

이 정책은 다음 상황을 다루지 않습니다.

- 오프라인 갱신 서버

  연장된 기간 동안 갱신 서버가 오프라인인 경우 갱신 기간이 누락될 수 있습니다. 이 경우 모든 비갱신 CA 서버는 인증서가 만료될 때까지 현재 시스템 인증서를 계속 다시 설치합니다. 이 경

우 하나의 만료된 인증서로 인해 다른 인증서에 대한 갱신 오류가 발생할 수 있기 때문에 IdM 배포가 중단됩니다.

- 복제 문제

갱신 서버와 기타 CA 복제본 간에 복제 문제가 있는 경우 갱신에 성공할 수 있지만 다른 CA 복제본은 만료되기 전에 업데이트된 인증서를 검색하지 못할 수 있습니다.

이러한 상황을 방지하려면 복제 계약이 올바르게 작동하는지 확인하십시오. 자세한 내용은 RHEL 7 *Linux 도메인 ID, 인증 및 정책 가이드의 일반 또는 특정 복제 문제 해결 지침을 참조하십시오*.

11.2. IDM CA 갱신 서버 변경 및 재설정

CA(인증 기관) 갱신 서버가 해제되면 IdM(Identity Management)은 IdM CA 서버 목록에서 새 CA 갱신 서버를 자동으로 선택합니다. 시스템 관리자가 선택 항목에 영향을 미칠 수 없습니다.

새 IdM CA 갱신 서버를 선택하려면 시스템 관리자가 교체를 수동으로 수행해야 합니다. 현재 갱신 서버 해제 프로세스를 시작하기 전에 새 CA 갱신 서버를 선택합니다.

현재 CA 갱신 서버 구성이 유효하지 않은 경우 IdM CA 갱신 서버를 재설정합니다.

CA 갱신 서버를 변경하거나 재설정하려면 다음 절차를 완료합니다.

사전 요구 사항

- IdM 관리자 인증 정보가 있습니다.

절차

1. IdM 관리자 인증 정보를 가져옵니다.

```
~]$ kinit admin
Password for admin@IDM.EXAMPLE.COM:
```

2. 경우에 따라 배포에서 새 CA 갱신 서버가 되는 데 필요한 CA 역할이 있는 IdM 서버를 확인할

수 있습니다.

```
~]$ ipa server-role-find --role 'CA server'
---------------------
2 server roles matched
---------------------
  Server name: server.idm.example.com
  Role name: CA server
  Role status: enabled

  Server name: replica.idm.example.com
  Role name: CA server
  Role status: enabled
----------------------------
Number of entries returned 2
----------------------------
```

배포에 두 개의 **CA** 서버가 있습니다.

3. 선택적으로 현재 **CA** 갱신 서버인 **CA** 서버를 확인하려면 다음을 입력합니다.

```
~]$ ipa config-show | grep 'CA renewal'
  IPA CA renewal master: server.idm.example.com
```

현재 갱신 서버는 **server.idm.example.com** 입니다.

4. 갱신 서버 구성을 변경하려면 **--ca-renewal-master-server** 옵션과 함께 **ipa config-mod** 유틸리티를 사용하십시오.

```
~]$ ipa config-mod --ca-renewal-master-server replica.idm.example.com | grep 'CA
renewal'
  IPA CA renewal master: replica.idm.example.com
```

중요

다음을 사용하여 새 **CA** 갱신 서버로 전환할 수도 있습니다.

- **ipa-cacert-manage --renew** 명령. 이 명령은 **CA** 인증서를 갱신 *하고* 새 **CA** 갱신 서버를 실행하는 **CA** 서버를 만듭니다.
- **ipa-cert-fix** 명령. 이 명령은 만료된 인증서가 실패할 때 배포를 복구합니다. 또한 새 **CA** 갱신 서버로 명령을 실행하는 **CA** 서버를 만듭니다.

자세한 내용은 **IdM**이 오프라인인 경우 만료된 시스템 인증서 업데이트를 참조하십시오.

11.3. IDM의 외부에서 자체 서명된 CA로 전환

외부 서명에서 **IdM(Identity Management)** 인증 기관(**CA**)의 자체 서명 인증서로 전환하려면 다음 절차를 완료합니다. 자체 서명된 **CA**를 사용하면 **CA** 인증서의 갱신이 자동으로 관리됩니다. 시스템 관리자가 외부 기관에 인증서 서명 요청(**CSR**)을 제출할 필요가 없습니다.

외부 서명에서 자체 서명된 **CA**로 전환하면 **CA** 인증서만 교체됩니다. 이전 **CA**에서 서명한 인증서는 계속 유효하며 계속 사용됩니다. 예를 들어, 자체 서명된 **CA**로 이동한 후에도 **LDAP** 인증서의 인증서 체인은 변경되지 않은 상태로 유지됩니다.

```
external_CA certificate > IdM CA certificate > LDAP certificate
```

사전 요구 사항

- **IdM CA** 갱신 서버와 모든 **IdM** 클라이언트 및 서버에 대한 루트 액세스 권한이 있습니다.

절차

1. **IdM CA** 갱신 서버에서 **CA** 인증서를 자체 서명으로 갱신합니다.

```
# ipa-cacert-manage renew --self-signed
Renewing CA certificate, please wait
CA certificate successfully renewed
The ipa-cacert-manage command was successful
```

2.

나머지 IdM 서버 및 클라이언트에 root 로 SSH 연결을 수행합니다. 예를 들면 다음과 같습니다.

```
# ssh root@idmclient01.idm.example.com
```

3. IdM 클라이언트에서 서버의 인증서로 로컬 IdM 인증서 데이터베이스를 업데이트합니다.

```
[idmclient01 ~]# ipa-certupdate
Systemwide CA database updated.
Systemwide CA database updated.
The ipa-certupdate command was successful
```

4. 경우에 따라 업데이트가 성공적으로 수행되었으며 새 CA 인증서가 /etc/ipa/ca.crt 파일에 추가되었는지 확인하려면 다음을 실행합니다.

```
[idmclient01 ~]$ openssl crl2pkcs7 -nocrl -certfile /etc/ipa/ca.crt | openssl pkcs7 -
print_certs -text -noout
[...]
Certificate:
   Data:
      Version: 3 (0x2)
      Serial Number: 39 (0x27)
      Signature Algorithm: sha256WithRSAEncryption
      Issuer: O=IDM.EXAMPLE.COM, CN=Certificate Authority
      Validity
         Not Before: Jul  1 16:32:45 2019 GMT
         Not After : Jul  1 16:32:45 2039 GMT
      Subject: O=IDM.EXAMPLE.COM, CN=Certificate Authority
[...]
```

이전 CA 인증서를 사용하여 새 CA 인증서가 나열되므로 출력에 업데이트가 성공한 것으로 표시됩니다.

11.4. 외부 서명된 인증서로 IDM CA 갱신 서버 갱신

이 섹션에서는 외부 CA를 사용하여 CSR(Identity Management) 인증 기관(CA) 인증서를 갱신하는 방법을 설명합니다. 이 구성에서 IdM CA 서버는 외부 CA의 하위 CA입니다. 외부 CA는 AD CS(Active Directory 인증서 서버)일 수 있지만 필요하지 않습니다.

외부 인증 기관이 AD CS인 경우 CSR에서 IdM CA 인증서에 원하는 템플릿을 지정할 수 있습니다. 인증서 템플릿은 인증서 요청이 수신될 때 CA에서 사용하는 정책 및 규칙을 정의합니다. AD의 인증서 템플릿은 IdM의 인증서 프로필에 해당합니다.

특정 AD CS 템플릿을 해당 Object Identifier(OID)로 정의할 수 있습니다. OIDS는 분산 애플리케이션의 데이터 요소, 구문 및 기타 부분을 고유하게 식별하기 위해 다양한 발행 기관에서 발행한 고유한 숫자 값입니다.

또는 특정 AD CS 템플릿을 이름으로 정의할 수도 있습니다. 예를 들어 AD CS에 IdM CA에서 제출한 CSR에 사용되는 기본 프로필의 이름은 하위 CA입니다.

CSR에서 OID 또는 이름을 지정하여 프로필을 정의하려면 external-ca-profile 옵션을 사용합니다. 자세한 내용은 ipa-cacert-manage man 페이지를 참조하십시오.

미리 준비된 인증서 템플릿을 사용하는 것 외에도 AD CS에 사용자 정의 인증서 템플릿을 생성하여 CSR에서 사용할 수도 있습니다.

사전 요구 사항

- IdM CA 갱신 서버에 대한 루트 액세스 권한이 있습니다.

절차

현재 CA 인증서가 자체 서명되었는지 또는 외부 서명되었는지 여부에 관계없이 외부 서명으로 IdM CA의 인증서를 갱신하려면 다음 절차를 완료합니다.

1. 외부 CA에 제출할 CSR을 생성합니다.

   - 외부 CA가 AD CS인 경우 --external-ca-type=ms-cs 옵션을 사용합니다. 기본 subCA 템플릿과 다른 템플릿을 원하는 경우 --external-ca-profile 옵션을 사용하여 지정합니다.

     ```
     ~]# ipa-cacert-manage renew --external-ca --external-ca-type=ms-cs [--external-ca-profile=PROFILE]
     Exporting CA certificate signing request, please wait
     The next step is to get /var/lib/ipa/ca.csr signed by your CA and re-run ipa-cacert-manage as:
     ipa-cacert-manage renew --external-cert-file=/path/to/signed_certificate --external-cert-file=/path/to/external_ca_certificate
     The ipa-cacert-manage command was successful
     ```

   - 외부 CA가 AD CS가 아닌 경우:

```
~]# ipa-cacert-manage renew --external-ca
Exporting CA certificate signing request, please wait
The next step is to get /var/lib/ipa/ca.csr signed by your CA and re-run ipa-cacert-manage as:
ipa-cacert-manage renew --external-cert-file=/path/to/signed_certificate --external-cert-file=/path/to/external_ca_certificate
The ipa-cacert-manage command was successful
```

출력은 **CSR**이 생성되고 **/var/lib/ipa/ca.csr** 파일에 저장되었음을 보여줍니다.

2. **/var/lib/ipa/ca.csr** 에 있는 **CSR**을 외부 **CA**에 제출합니다. 프로세스는 외부 **CA**로 사용할 서비스에 따라 다릅니다.

3. 다음과 같은 기본 **64** 인코딩 **Blob**에서 발행 **CA**의 발급 인증서 및 **CA** 인증서 체인을 검색합니다.

- 외부 **CA**가 **AD CS**가 아닌 경우 **PEM** 파일입니다.
- 외부 **CA**가 **AD CS**인 경우 **Base_64** 인증서입니다.

프로세스는 모든 인증서 서비스에 따라 다릅니다. 일반적으로 웹 페이지 또는 알림 이메일의 다운로드 링크를 통해 관리자는 필요한 모든 인증서를 다운로드할 수 있습니다.

외부 **CA**가 **AD CS**이고 **Microsoft Windows** 인증 기관 관리 창을 통해 알려진 템플릿으로 **CSR**을 제출한 경우 **AD CS**에서 인증서를 즉시 발행하고 인증서 저장 대화 상자가 **AD CS** 웹 인터페이스에 표시되어 발급된 인증서를 저장할 위치를 묻는 메시지가 표시됩니다.

4. **ipa-cacert-manage renew** 명령을 다시 실행하여 전체 인증서 체인을 제공하는 데 필요한 모든 **CA** 인증서 파일을 추가합니다. **--external-cert-file** 옵션을 여러 번 사용하여 필요한 파일을 여러 번 지정합니다.

```
~]# ipa-cacert-manage renew --external-cert-file=/path/to/signed_certificate --external-cert-file=/path/to/external_ca_certificate_1 --external-cert-file=/path/to/external_ca_certificate_2
```

5. 모든 **IdM** 서버 및 클라이언트에서 서버의 인증서로 로컬 **IdM** 인증서 데이터베이스를 업데이트합니다.

```
[client ~]$ ipa-certupdate
```

```
Systemwide CA database updated.
Systemwide CA database updated.
The ipa-certupdate command was successful
```

6. 경우에 따라 업데이트가 성공적으로 수행되었으며 새 **CA** 인증서가 **/etc/ipa/ca.crt** 파일에 추가되었는지 확인하려면 다음을 실행합니다.

```
[client ~]$ openssl crl2pkcs7 -nocrl -certfile /etc/ipa/ca.crt | openssl pkcs7 -print_certs
-text -noout
[...]
Certificate:
   Data:
      Version: 3 (0x2)
      Serial Number: 39 (0x27)
      Signature Algorithm: sha256WithRSAEncryption
      Issuer: O=IDM.EXAMPLE.COM, CN=Certificate Authority
      Validity
         Not Before: Jul  1 16:32:45 2019 GMT
         Not After : Jul  1 16:32:45 2039 GMT
      Subject: O=IDM.EXAMPLE.COM, CN=Certificate Authority
[...]
```

이전 **CA** 인증서를 사용하여 새 **CA** 인증서가 나열되므로 출력에 업데이트가 성공한 것으로 표시됩니다.

12장. IDM이 오프라인 상태일 때 만료된 시스템 인증서 갱신

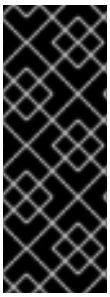
시스템 인증서가 만료된 경우 IdM(Identity Management)이 시작되지 않습니다. IdM은 ipa-cert-fix 툴을 사용하여 이 경우에도 시스템 인증서 갱신을 지원합니다.

사전 요구 사항

- IdM은 Red Hat Enterprise Linux 8.1 이상에만 설치됩니다.
- 호스트에 ipactl start --ignore-service-failures 명령을 입력하여 LDAP 서비스가 실행 중인지 확인합니다.

12.1. CA 갱신 서버에서 만료된 시스템 인증서 갱신

이 섹션에서는 만료된 IdM 인증서에 ipa-cert-fix 툴을 적용하는 방법을 설명합니다.



중요

CA 갱신 서버가 아닌 CA(Certificate Authority) 호스트에서 ipa-cert-fix 툴을 실행하고 유틸리티가 공유 인증서를 갱신하는 경우 해당 호스트는 도메인의 새 CA 갱신 서버가 자동으로 됩니다. 불일치를 방지하려면 도메인에는 항상 하나의 CA 갱신 서버가 있어야 합니다.

사전 요구 사항

- 관리 권한이 있는 서버에 로그인합니다.

절차

1. ipa-cert-fix 도구를 시작하여 시스템을 분석하고 갱신이 필요한 만료된 인증서를 나열합니다.

```
# ipa-cert-fix
...
The following certificates will be renewed:

Dogtag sslserver certificate:
  Subject: CN=ca1.example.com,O=EXAMPLE.COM 201905222205
  Serial:  13
```

```
  Expires: 2019-05-12 05:55:47
...
Enter "yes" to proceed:
```

2. yes 를 입력하여 갱신 프로세스를 시작합니다.

```
Enter "yes" to proceed: yes
Proceeding.
Renewed Dogtag sslserver certificate:
  Subject: CN=ca1.example.com,O=EXAMPLE.COM 201905222205
  Serial:  268369925
  Expires: 2021-08-14 02:19:33
...

Becoming renewal master.
The ipa-cert-fix command was successful
```

ipa-cert-fix 가 만료된 모든 인증서를 갱신하기 전까지 최대 1분이 걸릴 수 있습니다.

3. 필요한 경우 모든 서비스가 현재 실행 중인지 확인합니다.

```
# ipactl status
Directory Service: RUNNING
krb5kdc Service: RUNNING
kadmin Service: RUNNING
httpd Service: RUNNING
ipa-custodia Service: RUNNING
pki-tomcatd Service: RUNNING
ipa-otpd Service: RUNNING
ipa: INFO: The ipactl command was successful
```

이 시점에서 인증서가 업데이트되었으며 서비스가 실행 중입니다. 다음 단계는 IdM 도메인의 다른 서버를 확인하는 것입니다.

참고

여러 **CA** 서버에서 인증서를 복구해야 하는 경우**:**

1. **LDAP** 복제가 토폴로지 전체에서 작동하는지 확인한 후 먼저 위의 절차에 따라 하나의 **CA** 서버에서 **ipa-cert-fix** 를 실행합니다.

2. 다른 **CA** 서버에서 **ipa-cert-fix** 를 실행하기 전에 **getcert-resubmit** (기타 **CA** 서버에서)을 통해 공유 인증서의 인증서 갱신을 트리거하여 공유 인증서의 불필요한 갱신을 방지합니다.

12.2. 갱신 후 IDM 도메인의 다른 IDM 서버 확인

ipa-cert-fix 도구를 사용하여 **CA** 갱신 서버의 인증서를 갱신한 후 다음을 수행해야 합니다.

- 도메인에서 다른 **IdM(Identity Management)** 서버를 다시 시작합니다.
- **certmonger**가 인증서를 갱신했는지 확인합니다.
- 만료된 시스템 인증서가 있는 다른 **CA**(인증 기관) 복제본이 있는 경우 **ipa-cert-fix** 툴을 사용하여 해당 인증서를 갱신합니다.

사전 요구 사항

- 관리 권한이 있는 서버에 로그인합니다.

절차

1. **--force** 매개변수를 사용하여 **IdM**을 다시 시작하십시오.

```
# ipactl restart --force
```

--force 매개변수를 사용하면 **ipactl** 유틸리티에서 개별 서비스 시작 실패를 무시합니다. 예를 들어 서버가 만료된 인증서가 있는 **CA**인 경우 **pki-tomcat** 서비스가 시작되지 않습니다. 이는 **--force** 매개 변수를 사용하기 때문에 예상되고 무시됩니다.

2. 다시 시작한 후 **certmonger** 서비스가 인증서를 갱신하는지 확인합니다(인증서 상태가 **MONITORING**이라고 함).

```
# getcert list | egrep '^Request|status:|subject:'
Request ID '20190522120745':
        status: MONITORING
        subject: CN=IPA RA,O=EXAMPLE.COM 201905222205
Request ID '20190522120834':
        status: MONITORING
        subject: CN=Certificate Authority,O=EXAMPLE.COM 201905222205
...
```

certmonger 가 복제본의 공유 인증서를 갱신하기 전에 다소 시간이 걸릴 수 있습니다.

3. 서버가 **CA**인 경우 이전 명령은 **pki-tomcat** 서비스에서 사용하는 인증서에 대해 **CA_UNREACHABLE** 을 보고합니다.

```
Request ID '20190522120835':
        status: CA_UNREACHABLE
        subject: CN=ca2.example.com,O=EXAMPLE.COM 201905222205
...
```

4. 이 인증서를 업데이트하려면 **ipa-cert-fix** 유틸리티를 사용하십시오.

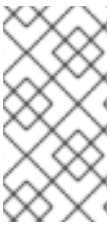
```
# ipa-cert-fix
Dogtag sslserver certificate:
  Subject: CN=ca2.example.com,O=EXAMPLE.COM
  Serial:  3
  Expires: 2019-05-11 12:07:11

Enter "yes" to proceed: yes
Proceeding.
Renewed Dogtag sslserver certificate:
  Subject: CN=ca2.example.com,O=EXAMPLE.COM 201905222205
  Serial:  15
  Expires: 2019-08-14 04:25:05

The ipa-cert-fix command was successful
```

이제 모든 **IdM** 인증서가 올바르게 갱신되고 작동합니다.

12.3. 웹 서버 및 LDAP 서버 인증서 교체

IdM(Identity Management) 시스템 관리자는 IdM 서버에서 실행되는 웹(또는 httpd) 및 LDAP(또는 디렉터리) 서비스의 인증서를 수동으로 교체할 수 있습니다. 예를 들어 certmonger 유틸리티가 인증서를 자동으로 갱신하도록 구성되지 않았거나 인증서가 외부 인증 기관(CA)에 의해 서명된 경우 이 작업이 필요할 수 있습니다.

이 예제에서는 server.idm.example.com IdM 서버에서 실행되는 서비스의 인증서를 설치합니다. 외부 CA에서 인증서를 가져옵니다.



참고

HTTP 및 LDAP 서비스 인증서에는 다른 IdM 서버에 다른 키 쌍과 제목 이름이 있으므로 각 IdM 서버에서 인증서를 개별적으로 갱신해야 합니다.

사전 요구 사항

- IdM 서버에 대한 루트 액세스 권한이 있습니다.
- Directory Manager 암호를 알고 있습니다.
- 외부 CA의 CA 인증서 체인인 *ca_certificate_chain_file.crt* 를 저장하는 파일에 액세스할 수 있습니다.

절차

1. IdM에 대한 추가 CA 인증서로 *ca_certificate_chain_file.crt* 에 포함된 인증서를 설치합니다.

```
# ipa-cacert-manage install
```

2. *ca_certicate_chain_file.crt* 의 인증서로 로컬 IdM 인증서 데이터베이스를 업데이트합니다.

```
# ipa-certupdate
```

3. OpenSSL 유틸리티를 사용하여 개인 키와 CSR(인증서 서명 요청)을 생성합니다.

```
$ openssl req -new -newkey rsa:2048 -days 365 -nodes -keyout new.key -out new.csr -
addext "subjectAltName = DNS:ipa-ca.idm.example.test" -subj
'/CN=server.idm.example.com,O=IDM.EXAMPLE.COM'
```

CSR을 외부 CA에 제출합니다. 프로세스는 외부 CA로 사용할 서비스에 따라 다릅니다. CA에서 인증서에 서명한 후 인증서를 IdM 서버로 가져옵니다.

4. IdM 서버에서 Apache 웹 서버의 이전 개인 키 및 인증서를 새 키 및 새로 서명된 인증서로 바꿉니다.

```
# ipa-server-certinstall -w --pin=password new.key new.crt
```

위 명령에서 다음을 수행합니다.

- w 옵션 은 웹 서버에 인증서를 설치하도록 지정합니다.
- pin 옵션 은 개인 키를 보호하는 암호를 지정합니다.

5. 메시지가 표시되면 Directory Manager 암호를 입력합니다.

6. LDAP 서버의 이전 개인 키 및 인증서를 새 키 및 새로 서명된 인증서로 바꿉니다.

```
# ipa-server-certinstall -d --pin=password new.key new.cert
```

위 명령에서 다음을 수행합니다.

- d 옵션은 인증서를 LDAP 서버에 설치하도록 지정합니다.
- pin 옵션 은 개인 키를 보호하는 암호를 지정합니다.

7. 메시지가 표시되면 Directory Manager 암호를 입력합니다.

8. httpd 서비스를 다시 시작합니다.

```
# systemctl restart httpd.service
```

9. **Directory** 서비스를 다시 시작하십시오.

```
# systemctl restart dirsrv@IDM.EXAMPLE.COM.service
```

추가 리소스

- IdM에서 작동하도록 인증서 형식 변환
- **ipa-server-certinstall(1)** 설명서 페이지

13장. IDM CA 서버에서 CRL 생성

IdM 배포에서 포함된 CA(인증 기관)를 사용하는 경우 하나의 IdM(Identity Management) 서버에서 다른 서버로 CRL(Certificate Revocation List) 생성을 이동해야 할 수 있습니다. 예를 들어 서버를 다른 시스템으로 마이그레이션하려는 경우 필요할 수 있습니다.

CRL을 생성하도록 하나의 서버만 구성합니다. CRL 게시자 역할을 수행하는 IdM 서버는 일반적으로 CA 갱신 서버 역할을 수행하는 서버와 동일하지만 필수는 아닙니다. CRL 게시자 서버를 해제하기 전에 CRL 게시자 서버 역할을 수행하도록 다른 서버를 선택하고 구성합니다.

이 장에서는 다음을 설명합니다.

- IdM 서버에서 CRL 생성을 중지합니다.
- IdM 복제본에서 CRL 생성 시작.

13.1. IDM 서버에서 CRL 생성 중지

IdM CRL 게시자 서버에서 CRL(인증서 취소 목록) 생성을 중지하려면 ipa-crlgen-manage 명령을 사용합니다. 생성을 비활성화하기 전에 서버가 CRL을 실제로 생성하는지 확인합니다. 그런 다음 비활성화할 수 있습니다.

사전 요구 사항

- IdM(Identity Management) 서버가 RHEL 8.1 시스템에 설치되어 있습니다.
- root로 로그인해야 합니다.

절차

1. 서버가 CRL을 생성하고 있는지 확인합니다.

```
[root@server ~]# ipa-crlgen-manage status
CRL generation: enabled
Last CRL update: 2019-10-31 12:00:00
Last CRL Number: 6
The ipa-crlgen-manage command was successful
```

2. 서버에서 **CRL** 생성을 중지합니다.

```
[root@server ~]# ipa-crlgen-manage disable
Stopping pki-tomcatd
Editing /var/lib/pki/pki-tomcat/conf/ca/CS.cfg
Starting pki-tomcatd
Editing /etc/httpd/conf.d/ipa-pki-proxy.conf
Restarting httpd
CRL generation disabled on the local host. Please make sure to configure CRL
generation on another master with ipa-crlgen-manage enable.
The ipa-crlgen-manage command was successful
```

3. 서버가 **CRL** 생성을 중지했는지 확인합니다.

```
[root@server ~]# ipa-crlgen-manage status
```

CRL 생성을 중지했습니다. 다음 단계는 **IdM** 복제본에서 **CRL** 생성을 활성화하는 것입니다.

13.2. IDM 복제본 서버에서 CRL 생성 시작

ipa-crlgen-manage 명령을 사용하여 **IdM CA** 서버에서 **CRL(**인증서 취소 목록**)**을 생성할 수 있습니다.

사전 요구 사항

- **IdM(Identity Management)** 서버가 **RHEL 8.1** 시스템에 설치되어 있습니다.
- **RHEL** 시스템은 **IdM** 인증 기관 서버여야 합니다.
- **root**로 로그인해야 합니다.

절차

1. **CRL** 생성을 시작합니다.

```
[root@replica1 ~]# ipa-crlgen-manage enable
Stopping pki-tomcatd
Editing /var/lib/pki/pki-tomcat/conf/ca/CS.cfg
```

```
Starting pki-tomcatd
Editing /etc/httpd/conf.d/ipa-pki-proxy.conf
Restarting httpd
Forcing CRL update
CRL generation enabled on the local host. Please make sure to have only a single CRL
generation master.
The ipa-crlgen-manage command was successful
```

2. **CRL이 생성되었는지 확인합니다.**

```
[root@replica1 ~]# ipa-crlgen-manage status
CRL generation: enabled
Last CRL update: 2019-10-31 12:10:00
Last CRL Number: 7
The ipa-crlgen-manage command was successful
```

14장. CA 갱신 서버 및 CRL 게시자 역할을 수행하는 서버 해제

하나의 서버가 CA(인증 기관) 갱신 서버 역할과 CRL(Certificate Revocation List) 게시자 역할을 모두 수행해야 할 수 있습니다. 이 서버를 오프라인 상태로 전환하거나 해제해야 하는 경우 이러한 역할을 수행하도록 다른 CA 서버를 선택하고 구성하십시오.

이 예에서는 CA 갱신 서버 및 CRL 게시자 역할을 이행하는 호스트 server.idm.example.com.com을 해제해야 합니다. 이 절차에서는 CA 갱신 서버 및 CRL 게시자 역할을 호스트 replica.idm.example.com 으로 전송하고 IdM 환경에서 server.idm.example.com 을 제거합니다.

참고

CA 갱신 서버와 CRL 게시자 역할을 모두 수행하도록 동일한 서버를 구성할 필요가 없습니다.

사전 요구 사항

- IdM 관리자 인증 정보가 있습니다.
- 해제 중인 서버의 root 암호가 있습니다.
- IdM 환경에 두 개 이상의 CA 복제본이 있습니다.

절차

1. IdM 관리자 인증 정보를 가져옵니다.

```
[user@server ~]$ kinit admin
Password for admin@IDM.EXAMPLE.COM:
```

2. *(선택 사항)* CA 갱신 서버 및 CRL 게시자 역할을 수행하는 서버가 확실하지 않은 경우:

   a. 현재 CA 갱신 서버를 표시합니다. IdM 서버에서 다음 명령을 실행할 수 있습니다.

```
[user@server ~]$ ipa config-show | grep 'CA renewal'
  IPA CA renewal master: server.idm.example.com
```

b.

호스트가 현재 **CRL** 게시자인지 테스트합니다.

```
[user@server ~]$ ipa-crlgen-manage status
CRL generation: enabled
Last CRL update: 2019-10-31 12:00:00
Last CRL Number: 6
The ipa-crlgen-manage command was successful
```

CRL을 생성하지 않는 **CA** 서버는 **CRL** 생성**: disabled** 를 표시합니다.

```
[user@replica ~]$ ipa-crlgen-manage status
CRL generation: disabled
The ipa-crlgen-manage command was successful
```

CRL 게시자 서버를 찾을 때까지 **CA** 서버에서 이 명령을 계속 입력합니다.

c.

이러한 역할을 수행하기 위해 승격할 수 있는 다른 모든 **CA** 서버를 표시합니다. 이 환경에는 두 개의 **CA** 서버가 있습니다.

```
[user@server ~]$ ipa server-role-find --role 'CA server'
----------------------
2 server roles matched
----------------------
  Server name: server.idm.example.com
  Role name: CA server
  Role status: enabled
  Server name: replica.idm.example.com
  Role name: CA server
  Role status: enabled
----------------------------
Number of entries returned 2
----------------------------
```

3.

replica.idm.example.com 을 **CA** 갱신 서버로 설정합니다.

```
[user@server ~]$ ipa config-mod --ca-renewal-master-server replica.idm.example.com
```

4.

server.idm.example.com 에서 다음을 수행합니다.

a.

인증서 업데이트 작업을 비활성화합니다.

```
[root@server ~]# pki-server ca-config-set ca.certStatusUpdateInterval 0
```

b. IdM 서비스를 다시 시작하십시오.

```
[user@server ~]$ ipactl restart
```

5. replica.idm.example.com 에서 다음을 수행합니다.

a. 인증서 업데이트 프로그램을 활성화합니다.

```
[root@server ~]# pki-server ca-config-unset ca.certStatusUpdateInterval
```

b. IdM 서비스를 다시 시작하십시오.

```
[user@replica ~]$ ipactl restart
```

6. server.idm.example.com 에서 CRL 생성을 중지합니다.

```
[user@server ~]$ ipa-crlgen-manage disable
Stopping pki-tomcatd
Editing /var/lib/pki/pki-tomcat/conf/ca/CS.cfg
Starting pki-tomcatd
Editing /etc/httpd/conf.d/ipa-pki-proxy.conf
Restarting httpd
CRL generation disabled on the local host. Please make sure to configure CRL
generation on another master with ipa-crlgen-manage enable.
The ipa-crlgen-manage command was successful
```

7. replica.idm.example.com 에서 CRL 생성을 시작합니다.

```
[user@replica ~]$ ipa-crlgen-manage enable
Stopping pki-tomcatd
Editing /var/lib/pki/pki-tomcat/conf/ca/CS.cfg
Starting pki-tomcatd
Editing /etc/httpd/conf.d/ipa-pki-proxy.conf
Restarting httpd
Forcing CRL update
CRL generation enabled on the local host. Please make sure to have only a single CRL
generation master.
The ipa-crlgen-manage command was successful
```

8. server.idm.example.com 에서 IdM 서비스를 중지합니다.

```
[user@server ~]$ ipactl stop
```

9. replica.idm.example.com 에서 IdM 환경에서 server.idm.example.com 을 삭제합니다.

```
[user@replica ~]$ ipa server-del server.idm.example.com
```

10. server.idm.example.com 에서 ipa-server-install --uninstall 명령을 root 계정으로 사용합니다.

```
[root@server ~]# ipa-server-install --uninstall
...
Are you sure you want to continue with the uninstall procedure? [no]: yes
```

검증 단계

- 현재 CA 갱신 서버를 표시합니다.

```
[user@replica ~]$ ipa config-show | grep 'CA renewal'
  IPA CA renewal master: replica.idm.example.com
```

- replica.idm.example.com 호스트에서 CRL을 생성하고 있는지 확인합니다.

```
[user@replica ~]$ ipa-crlgen-manage status
CRL generation: enabled
Last CRL update: 2019-10-31 12:10:00
Last CRL Number: 7
The ipa-crlgen-manage command was successful
```

추가 리소스

- IdM CA 갱신 서버 변경 및 재설정
- IdM CA 서버에서 CRL 생성
- IdM 복제본 설치 제거

15장. CERTMONGER를 사용하여 서비스에 대한 IDM 인증서 가져오기

15.1. CERTMONGER 개요

통합된 IdM 인증 기관(CA)과 함께 IdM 관리(IdM)를 설치할 때 certmonger 서비스를 사용하여 시스템 및 서비스 인증서를 추적 및 갱신합니다. 인증서가 만료 날짜에 도달하면 certmonger 는 갱신 프로세스를 다음과 같이 관리합니다.

- 원래 요청에 제공된 옵션을 사용하여 CSR(인증서 서명 요청)을 다시 생성합니다.
- IdM API cert-request 명령을 사용하여 CSR을 IdM CA에 제출.
- IdM CA에서 인증서 받기.
- 원래 요청으로 지정한 경우 사전 저장 명령 실행
- 갱신 요청에 지정된 위치에 새 인증서 설치(NSS 데이터베이스 또는 파일에 있음).
- 원래 요청으로 지정한 경우 post-save 명령 실행 예를 들어 post-save 명령은 certmonger 에 관련 서비스를 다시 시작하도록 지시하여 서비스가 새 인증서를 선택하도록 지시할 수 있습니다.

인증서 certmonger 트랙의 유형

인증서는 시스템 및 서비스 인증서로 나눌 수 있습니다.

서비스 인증서(예: HTTP,LDAP 및 PKINIT)와 달리 다른 서버에서 키 쌍과 제목 이름이 있는 경우 IdM 시스템 인증서 및 해당 키는 모든 CA 복제본에서 공유합니다. IdM 시스템 인증서는 다음과 같습니다.

- IdM CA 인증서
- OCSP 서명 인증서

- IdM CA 하위 시스템 인증서
- IdM CA 감사 서명 인증서
- IdM 갱신 에이전트 (RA) 인증서
- KRA 전송 및 스토리지 인증서

certmonger 서비스는 통합된 CA를 사용하여 IdM 환경을 설치하는 동안 요청된 IdM 시스템 및 서비스 인증서를 추적합니다. 또한 certmonger 는 IdM 호스트에서 실행되는 다른 서비스에 대해 시스템 관리자가 수동으로 요청한 인증서를 추적합니다. certmonger는 외부 CA 인증서 또는 사용자 인증서를 추적하지 않습니다.

certmonger 구성 요소

certmonger 서비스는 다음 두 가지 주요 구성 요소로 구성됩니다.

- 인증서 목록을 추적하고 갱신 명령 시작 엔진인 certmonger 데몬
- 시스템 관리자가 명령을 certmonger 데몬에 적극적으로 보낼 수 있는 CLI(명령줄 인터페이스)용 getcert 유틸리티입니다.

특히 시스템 관리자는 getcert 유틸리티를 사용하여 다음을 수행할 수 있습니다.

- 새 인증서 요청
- certmonger 가 추적하는 인증서 목록 보기
- 인증서 추적 또는 시작
- 인증서 갱신

15.2. CERTMONGER를 사용하여 서비스에 대한 IDM 인증서 가져오기

IdM(Identity Management) 클라이언트에서 실행되는 브라우저와 웹 서비스 간의 통신이 안전하고 암호화되었는지 확인하려면 TLS 인증서를 사용하십시오. IdM CA(인증 기관)에서 웹 서비스의 TLS 인증서를 가져옵니다.

이 섹션에서는 certmonger 를 사용하여 IdM 클라이언트에서 실행되는 서비스 (HTTP/my_company.idm.example.com@IDM.EXAMPLE.COM)에 대한 IdM 인증서를 가져오는 방법을 설명합니다.

certmonger 를 사용하여 인증서를 자동으로 요청하면 certmonger 가 인증서를 갱신할 때 인증서를 관리하고 갱신할 수 있습니다.

certmonger 가 서비스 인증서를 요청할 때 발생하는 상황을 시각적으로 표현하려면 서비스 인증서를 요청하는 certmonger의 통신 흐름을 참조하십시오.

사전 요구 사항

- 웹 서버는 IdM 클라이언트로 등록됩니다.
- 프로시저를 실행하는 IdM 클라이언트에 대한 루트 액세스 권한이 있습니다.
- 인증서를 요청하는 서비스는 IdM에 사전 존재할 필요가 없습니다.

절차

1. HTTP 서비스가 실행 중인 my_company.idm.example.com IdM 클라이언트에서 HTTP/my_company.idm.example.com@IDM.EXAMPLE.COM principal에 해당하는 서비스의 인증서를 요청하고 이를 지정합니다.
   - 인증서는 로컬 /etc/pki/tls/certs/httpd.pem 파일에 저장됩니다.
   - 개인 키는 로컬 /etc/pki/tls/private/httpd.key 파일에 저장해야 합니다.
   - SubjectAltName 에 대한 extensionRequest가 my_company.idm.example.com 의

DNS 이름으로 서명 요청에 추가됩니다.

```
# ipa-getcert request -K HTTP/my_company.idm.example.com -k
/etc/pki/tls/private/httpd.key -f /etc/pki/tls/certs/httpd.pem -g 2048 -D
my_company.idm.example.com -C "systemctl restart httpd"
New signing request "20190604065735" added.
```

위 명령에서 다음을 수행합니다.

- ipa-getcert request 명령은 IdM CA에서 인증서를 가져올 수 있도록 지정합니다. ipa-getcert request 명령은 getcert request -c IPA 를 위한 바로 가기입니다.
- g 옵션은 아직 없는 경우 생성할 키 크기를 지정합니다.
- -D 옵션은 요청에 추가할 SubjectAltName DNS 값을 지정합니다.
- c 옵션은 인증서를 가져온 후 certmonger 에 httpd 서비스를 다시 시작하도록 지시합니다.
- 특정 프로필로 인증서를 발급하도록 지정하려면 -T 옵션을 사용합니다.
- 지정된 CA에서 이름이 지정된 발행자를 사용하여 인증서를 요청하려면 -X ISSUER 옵션을 사용합니다.

참고

RHEL 8은 RHEL 7에서 사용되는 Apache와 다른 SSL 모듈을 사용합니다. SSL 모듈은 NSS가 아닌 OpenSSL에 의존합니다. 이러한 이유로 RHEL 8에서는 NSS 데이터베이스를 사용하여 HTTPS 인증서와 개인 키를 저장할 수 없습니다.

2. 필요한 경우 요청 상태를 확인하려면 다음을 수행합니다.

```
# ipa-getcert list -f /etc/pki/tls/certs/httpd.pem
Number of certificates and requests being tracked: 3.
Request ID '20190604065735':
```

```
    status: MONITORING
    stuck: no
    key pair storage: type=FILE,location='/etc/pki/tls/private/httpd.key'
    certificate: type=FILE,location='/etc/pki/tls/certs/httpd.crt'
    CA: IPA
[...]
```

출력에 요청이 **MONITORING** 상태임을 보여줍니다. 즉, 인증서를 가져왔음이 표시됩니다. 키 쌍과 인증서의 위치는 요청된 위치입니다.

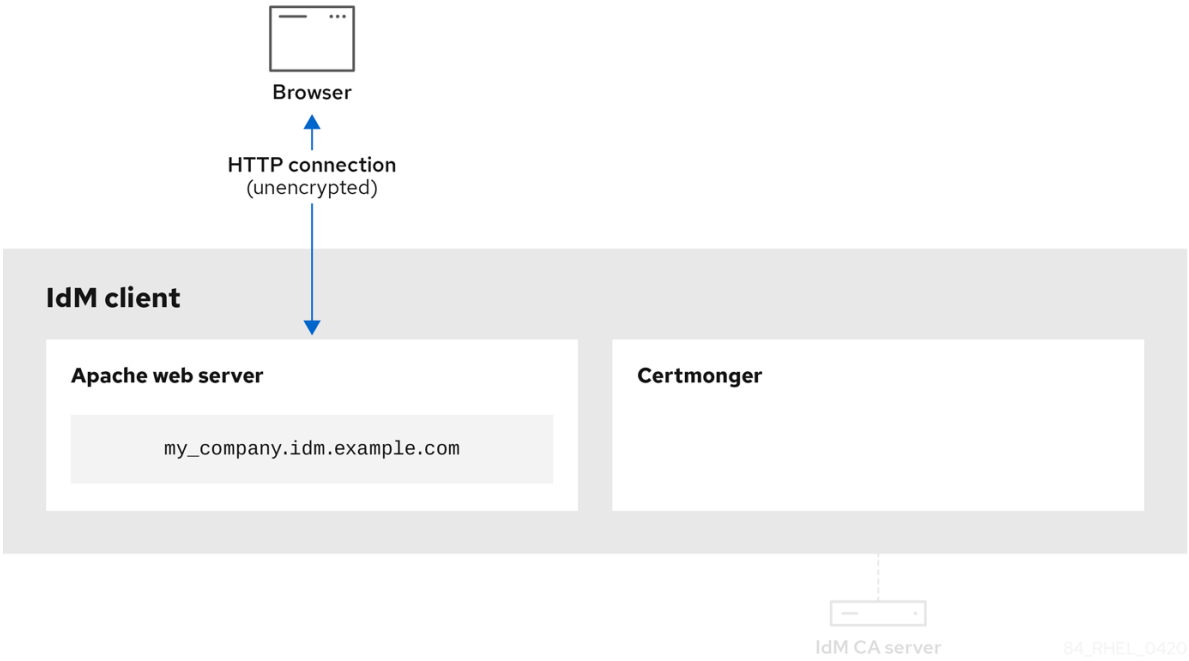
15.3. 서비스 인증서를 요청하는 CERTMONGER의 통신 흐름

이 섹션의 다이어그램은 **certmonger** 가 **IdM(Identity Management)** 인증 기관**(CA)** 서버에서 서비스 인증서를 요청할 때 발생하는 단계의 단계를 보여줍니다. 시퀀스는 다음 다이어그램으로 구성됩니다.

- 암호화되지 않은 통신
- **certmonger** 서비스 인증서 요청
- **IdM CA**가 서비스 인증서를 발급합니다.
- **certmonger** 서비스 인증서 적용
- **certmonger** 이전 인증서가 만료될 때 새 인증서를 요청

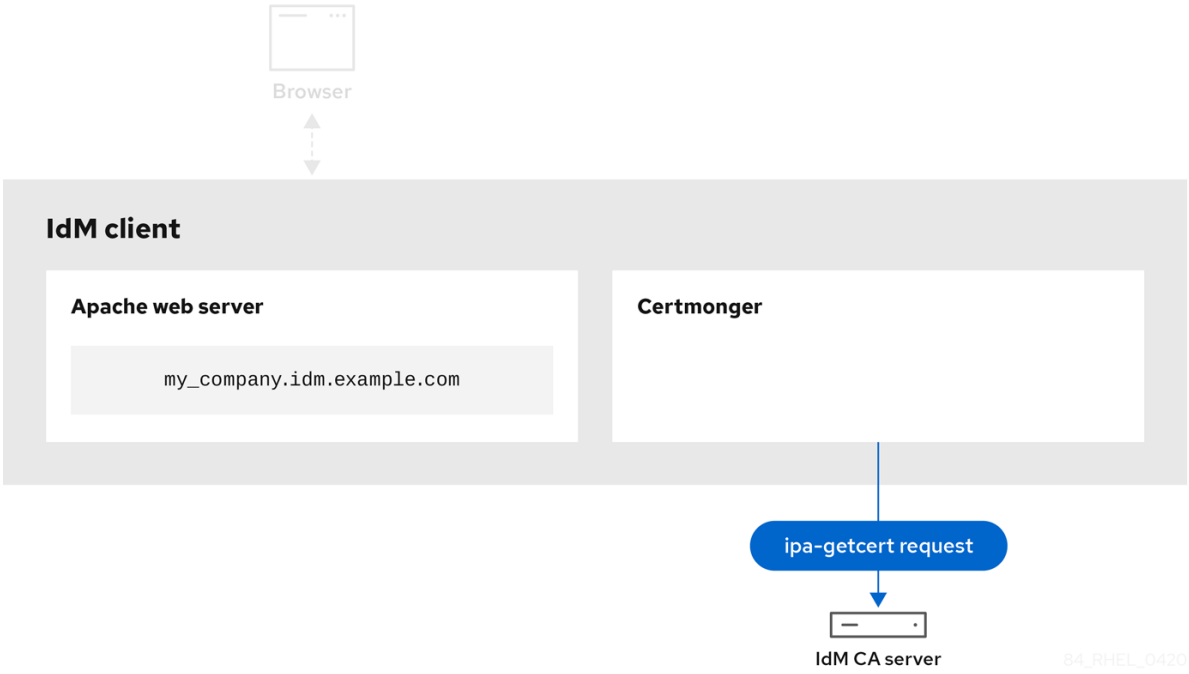
암호화되지 않은 통신은 초기 상황을 보여줍니다. **HTTPS** 인증서가 없으면 웹 서버와 브라우저 간의 통신이 암호화되지 않습니다.

그림 15.1. 암호화되지 않은 통신



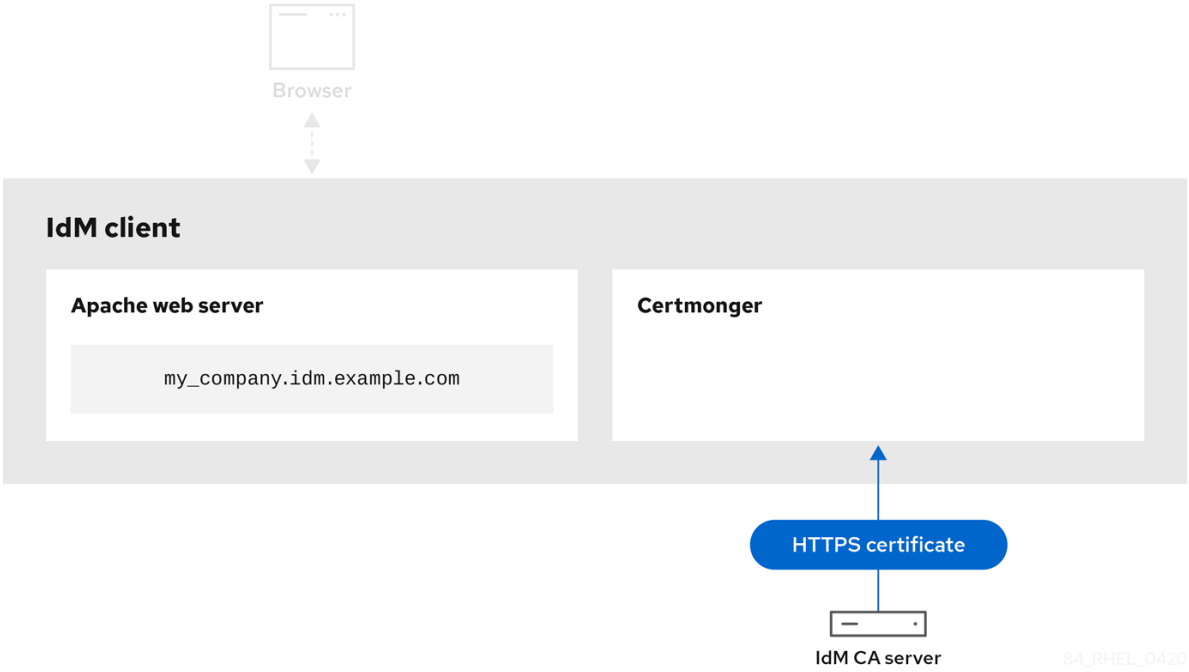
certmonger에서 서비스 인증서를 요청하는 것은 certmonger 를 사용하여 Apache 웹 서버에 대한 HTTPS 인증서를 수동으로 요청하는 시스템 관리자에게 표시됩니다. 웹 서버 인증서를 요청할 때 certmonger는 CA와 직접 통신하지 않습니다. IdM을 통해 프록시합니다.

그림 15.2. 서비스 인증서를 요청하는 certmonger



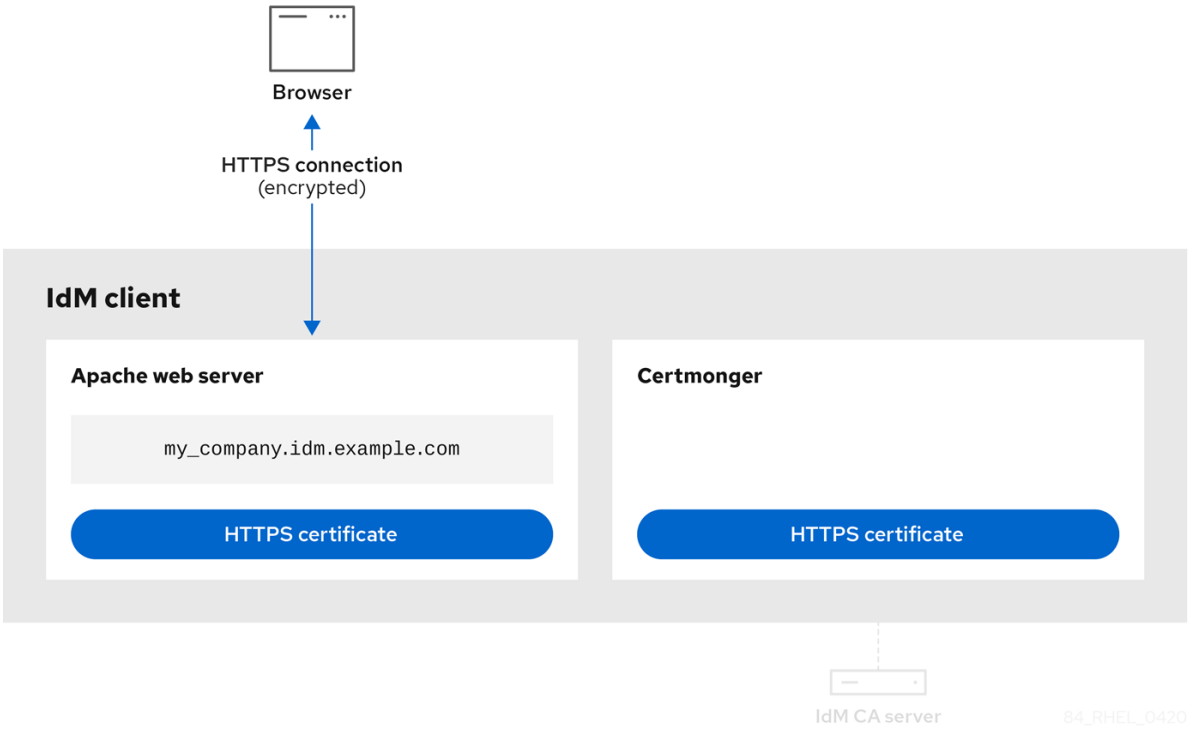
서비스 인증서를 발행하는 IdM CA 에서 웹 서버의 HTTPS 인증서를 발행하는 IdM CA가 표시됩니다.

그림 15.3. 서비스 인증서 발행 IdM CA



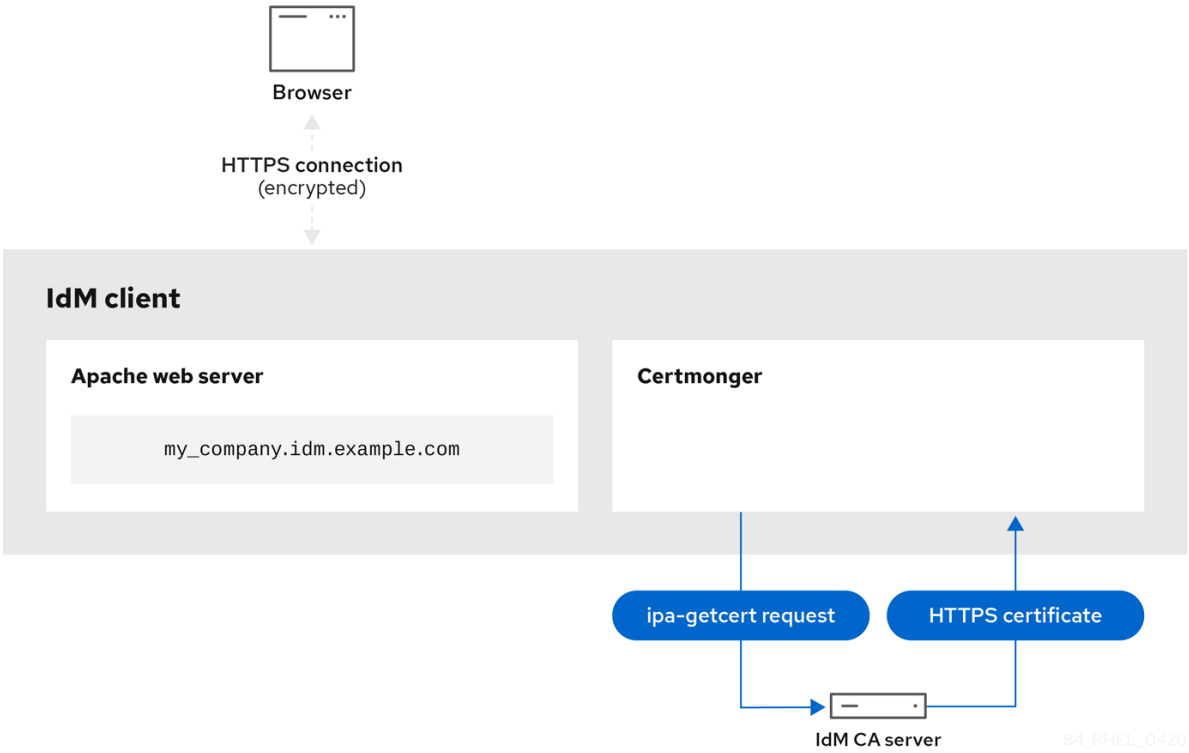
certmonger 에서 서비스 인증서를 적용하는 경우 IdM 클라이언트의 적절한 위치에 HTTPS 인증서를 배치하고 이를 수행하라는 지침이 나타나면 httpd 서비스를 다시 시작합니다. 이후 Apache 서버는 HTTPS 인증서를 사용하여 자체와 브라우저 간의 트래픽을 암호화합니다.

그림 15.4. 서비스 인증서 적용 certmonger



certmonger에서 이전 만료일이 가까워지면 certmonger 에서 인증서가 만료되기 전에 IdM CA에서 서비스 인증서 갱신을 자동으로 요청합니다. IdM CA는 새 인증서를 발행합니다.

그림 15.5. 이전 인증서가 만료될 때 새 인증서를 요청합니다.



15.4. CERTMONGER에서 추적하는 인증서 요청의 세부 정보 보기

certmonger 서비스는 인증서 요청을 모니터링합니다. 인증서에 대한 요청이 성공적으로 서명되면 인증서가 생성됩니다. **certmonger** 는 결과 인증서를 포함한 인증서 요청을 관리합니다. 이 섹션에서는 **certmonger** 가 관리하는 특정 인증서 요청의 세부 정보를 보는 방법을 설명합니다.

절차

- 인증서 요청을 지정하는 방법을 알고 있는 경우 해당 특정 인증서 요청의 세부 정보만 나열합니다. 예를 들어 다음을 지정할 수 있습니다.
  - 요청 ID
  - 인증서의 위치
  - 인증서 이름

예를 들어 요청 ID가 20190408143846인 인증서의 세부 정보를 보려면 -v 옵션을 사용하여 인증서에 대한 요청이 실패한 경우 오류의 모든 세부 정보를 볼 수 있습니다.

```
# getcert list -i 20190408143846 -v
Number of certificates and requests being tracked: 16.
Request ID '20190408143846':
 status: MONITORING
 stuck: no
 key pair storage: type=NSSDB,location='/etc/dirsrv/slapd-IDM-EXAMPLE-COM',nickname='Server-Cert',token='NSS Certificate DB',pinfile='/etc/dirsrv/slapd-IDM-EXAMPLE-COM/pwdfile.txt'
 certificate: type=NSSDB,location='/etc/dirsrv/slapd-IDM-EXAMPLE-COM',nickname='Server-Cert',token='NSS Certificate DB'
 CA: IPA
 issuer: CN=Certificate Authority,O=IDM.EXAMPLE.COM
 subject: CN=r8server.idm.example.com,O=IDM.EXAMPLE.COM
 expires: 2021-04-08 16:38:47 CEST
 dns: r8server.idm.example.com
 principal name: ldap/server.idm.example.com@IDM.EXAMPLE.COM
 key usage: digitalSignature,nonRepudiation,keyEncipherment,dataEncipherment
 eku: id-kp-serverAuth,id-kp-clientAuth
 pre-save command:
 post-save command: /usr/libexec/ipa/certmonger/restart_dirsrv IDM-EXAMPLE-COM
 track: yes
 auto-renew: yes
```

출력에는 인증서에 대한 여러 정보가 표시됩니다. 예를 들면 다음과 같습니다.

- 인증서 위치(위의 예에서)는 /etc/dirsrv/slapd-IDM-EXAMPLE-COM 디렉토리에 있는 NSS 데이터베이스입니다.
- 위의 예에서 인증서 닉네임은 Server-Cert입니다.
- 핀을 저장하는 파일; 위의 예에서, /etc/dirsrv/slapd-IDM-EXAMPLE-COM/pwdfile.txt
- 인증서를 갱신하는 데 사용되는 CA(인증 기관)입니다. 위의 예에서는 IPA CA입니다.
- 만료일(위의 예에서는 2021-04-08 16:38:47 CEST)입니다.
- 인증서의 상태는 위 예제에서 MONITORING 상태는 인증서가 유효하고 추적되고 있음

을 나타냅니다.

- post-save 명령(위의 예에서는 LDAP 서비스를 다시 시작함)

- 인증서 요청을 지정하는 방법을 모르는 경우 certmonger 가 모니터링하거나 가져오기를 시도하는 모든 인증서의 세부 정보를 나열합니다.

```
# getcert list
```

추가 리소스

- getcert list man 페이지를 참조하십시오.

15.5. 인증서 추적 시작 및 중지

이 섹션에서는 getcert stop-tracking 및 getcert start-tracking 명령을 사용하여 인증서를 모니터링하는 방법을 설명합니다. 두 명령은 certmonger 서비스에서 제공합니다. 인증서 추적을 활성화하면 IdM(Identity Management) 인증 기관(CA)에서 다른 IdM 클라이언트의 시스템으로 발급한 인증서를 가져온 경우 특히 유용합니다. 인증서 추적을 활성화하는 것도 다음 프로비저닝 시나리오의 최종 단계가 될 수 있습니다.

1. IdM 서버에서 아직 존재하지 않는 시스템의 인증서를 생성합니다.
2. 새 시스템을 생성합니다.
3. 새 시스템을 IdM 클라이언트로 등록합니다.
4. 의 IdM 서버에서 IdM 클라이언트로 인증서와 키를 가져옵니다.
5. 인증서가 만료될 때 갱신되는지 확인하기 위해 certmonger 를 사용하여 인증서 추적을 시작합니다.

절차

- 요청 ID가 20190408143846인 인증서 모니터링을 비활성화하려면 다음을 수행합니다.

```
# getcert stop-tracking -i 20190408143846
```

자세한 내용은 getcert stop-tracking man 페이지를 참조하십시오.

- /tmp/some_cert.crt 파일에 저장된 인증서 모니터링을 활성화하려면 개인 키가 /tmp/some_key.key 파일에 저장됩니다.

```
# getcert start-tracking -c IPA -f /tmp/some_cert.crt -k /tmp/some_key.key
```

certmonger는 인증서를 발급한 CA 유형을 자동으로 식별할 수 없습니다. 따라서 IdM CA에서 인증서를 발급한 경우 IPA 값과 함께 -c 옵션을 getcert start-tracking 명령에 추가합니다. c 옵션을 추가하려면 생략하면 certmonger 가 NEED_CA 상태를 입력합니다.

자세한 내용은 getcert start-tracking man 페이지를 참조하십시오.

참고

두 명령은 인증서를 조작하지 않습니다. 예를 들어, getcert stop-tracking 은 인증서를 삭제하거나 NSS 데이터베이스 또는 파일 시스템에서 제거하지 않지만 모니터링된 인증서 목록에서 인증서를 제거합니다. 마찬가지로 getcert start-tracking 은 모니터링된 인증서 목록에 인증서만 추가합니다.

15.6. 수동으로 인증서 갱신

인증서가 만료일이 되면 certmonger 데몬은 인증 기관(CA) 도우미를 사용하여 갱신 명령을 자동으로 발행하고, 갱신된 인증서를 가져온 후 이전 인증서를 새 인증서로 교체합니다.

getcert resubmit 명령을 사용하여 사전에 인증서를 수동으로 갱신할 수도 있습니다. 이렇게 하면 인증서가 포함된 정보를 업데이트할 수 있습니다(예: SAN(Subject Alternative Name)).

이 섹션에서는 인증서를 수동으로 갱신하는 방법을 설명합니다.

절차

- 요청 ID가 20190408143846인 인증서를 갱신하려면 다음을 수행합니다.

```
# getcert resubmit -i 20190408143846
```

특정 인증서에 대한 요청 ID를 가져오려면 getcert list 명령을 사용합니다. 자세한 내용은 getcert list man 페이지를 참조하십시오.

15.7. CA 복제본에서 CERTMONGER가 IDM 인증서 추적을 재개하도록 설정

다음 절차에서는 certmonger 가 인증서 추적이 중단된 후 통합된 인증 기관을 사용하여 IdM(Identity Management) 시스템 인증서 추적을 재개하도록 하는 방법을 보여줍니다. 시스템 인증서를 갱신하거나 복제 토폴로지가 제대로 작동하지 않는 동안 IdM 호스트가 IdM에서 해제되어 발생했기 때문일 수 있습니다. 이 절차에서는 certmonger 가 IdM 서비스 인증서의 추적을 재개하도록 하는 방법(HTTP,LDAP 및 PKINIT 인증서)도 보여줍니다.

사전 요구 사항

- 시스템 인증서 추적을 재개하려는 호스트는 IdM CA 갱신 서버가 아닌 IdM 인증 기관(CA)이기도 합니다.

절차

1. 하위 시스템 CA 인증서에 대한 Pin을 가져옵니다.

```
# grep 'internal=' /var/lib/pki/pki-tomcat/conf/password.conf
```

2. 하위 시스템 CA 인증서에 추적을 추가하고 아래 명령의 [internal PIN] 을 이전 단계에서 얻은 PIN으로 교체합니다.

```
# getcert start-tracking -d /etc/pki/pki-tomcat/alias -n "caSigningCert cert-pki-ca" -c
'dogtag-ipa-ca-renew-agent' -P [internal PIN] -B
/usr/libexec/ipa/certmonger/stop_pkicad -C '/usr/libexec/ipa/certmonger/renew_ca_cert
-T caCACert "caSigningCert cert-pki-ca"'

# getcert start-tracking -d /etc/pki/pki-tomcat/alias -n "auditSigningCert cert-pki-ca" -c
'dogtag-ipa-ca-renew-agent' -P [internal PIN] -B
/usr/libexec/ipa/certmonger/stop_pkicad -C '/usr/libexec/ipa/certmonger/renew_ca_cert
-T caSignedLogCert "auditSigningCert cert-pki-ca"'

# getcert start-tracking -d /etc/pki/pki-tomcat/alias -n "ocspSigningCert cert-pki-ca" -c
'dogtag-ipa-ca-renew-agent' -P [internal PIN] -B
/usr/libexec/ipa/certmonger/stop_pkicad -C '/usr/libexec/ipa/certmonger/renew_ca_cert
```

```
-T caOCSPCert "ocspSigningCert cert-pki-ca"'

# getcert start-tracking -d /etc/pki/pki-tomcat/alias -n "subsystemCert cert-pki-ca" -c
'dogtag-ipa-ca-renew-agent' -P [internal PIN] -B
/usr/libexec/ipa/certmonger/stop_pkicad -C '/usr/libexec/ipa/certmonger/renew_ca_cert
-T caSubsystemCert "subsystemCert cert-pki-ca"'

# getcert start-tracking -d /etc/pki/pki-tomcat/alias -n "Server-Cert cert-pki-ca" -c
'dogtag-ipa-ca-renew-agent' -P [internal PIN] -B
/usr/libexec/ipa/certmonger/stop_pkicad -C '/usr/libexec/ipa/certmonger/renew_ca_cert
-T caServerCert "Server-Cert cert-pki-ca"'
```

3. 나머지 IdM 인증서, HTTP,LDAP,IPA 갱신 에이전트 및 PKINIT 인증서에 대한 추적을 추가합니다.

```
# getcert start-tracking -f /var/lib/ipa/certs/httpd.crt -k /var/lib/ipa/private/httpd.key -p
/var/lib/ipa/passwds/idm.example.com-443-RSA -c IPA -C
/usr/libexec/ipa/certmonger/restart_httpd -T calPAserviceCert

# getcert start-tracking -d /etc/dirsrv/slapd-IDM-EXAMPLE-COM -n "Server-Cert" -c IPA
-p /etc/dirsrv/slapd-IDM-EXAMPLE-COM/pwdfile.txt -C
'/usr/libexec/ipa/certmonger/restart_dirsrv -T calPAserviceCert "IDM-EXAMPLE-COM"'

# getcert start-tracking -f /var/lib/ipa/ra-agent.pem -k /var/lib/ipa/ra-agent.key -c
dogtag-ipa-ca-renew-agent -B /usr/libexec/ipa/certmonger/renew_ra_cert_pre -C
/usr/libexec/ipa/certmonger/renew_ra_cert -T caSubsystemCert

# getcert start-tracking -f /var/kerberos/krb5kdc/kdc.crt -k
/var/kerberos/krb5kdc/kdc.key -c dogtag-ipa-ca-renew-agent -B
/usr/libexec/ipa/certmonger/renew_ra_cert_pre -C
/usr/libexec/ipa/certmonger/renew_kdc_cert -T KDCs_PKINIT_Certs
```

4. certmonger 를 다시 시작합니다.

```
# systemctl restart certmonger
```

5. certmonger 가 시작된 후 1분 정도 기다린 후 새 인증서의 상태를 확인합니다.

```
# getcert list
```

추가 리소스

- IdM 시스템 인증서가 모두 만료된 경우 이 KCS(기술 센터 지원) 솔루션을 참조하여 CA 갱신 서버 및 CRL 게시자 서버이기도 IdM CA 서버에서 IdM 시스템 인증서를 수동으로 갱신합니다. 그런 다음 이 KCS 솔루션에 설명된 절차에 따라 토폴로지의 다른 모든 CA 서버에서 IdM 시스템 인증서를 수동으로 갱신합니다.

16장. RHEL 시스템 역할을 사용하여 인증서 요청

인증서 문제 및 업데이트 시스템 역할을 통해 Red Hat Ansible Core를 사용하여 인증서를 발행하고 관리할 수 있습니다.

이 장에서는 다음 항목에 대해 설명합니다.

- 인증서 시스템 역할
- 인증서 시스템 역할을 사용하여 새 자체 서명 인증서 요청
- 인증서 시스템 역할을 사용하여 IdM CA에서 새 인증서 요청

16.1. 인증서 문제 및 업데이트 시스템 역할

인증서 문제 및 업데이트 시스템 역할을 사용하여 Ansible Core를 사용하여 TLS 및 SSL 인증서를 발급 및 갱신할 수 있습니다.

이 역할은 certmonger 를 인증서 공급자로 사용하며 현재 자체 서명된 인증서 발행 및 갱신 및 IdM 통합 인증 기관(CA)을 지원합니다.

인증서 문제 및 업데이트 시스템 역할에서 Ansible 플레이북에서 다음 변수를 사용할 수 있습니다.

certificate_wait

작업에서 인증서가 발급될 때까지 기다려야 하는지를 지정합니다.

certificate_requests

발행할 각 인증서와 해당 매개변수를 나타냅니다.

추가 리소스

- /usr/share/ansible/roles/rhel-system-roles.certificate/README.md 파일을 참조하십시오.

- RHEL 시스템 역할 시작하기를 참조하십시오.

16.2. 인증서 문제 및 업데이트 시스템 역할을 사용하여 자체 서명된 새 인증서 요청

인증서 문제 및 업데이트 시스템 역할을 통해 Ansible Core를 사용하여 자체 서명된 인증서를 발행할 수 있습니다.

이 프로세스에서는 certmonger 공급자를 사용하고 getcert 명령을 통해 인증서를 요청합니다.

참고

기본적으로 certmonger 는 만료되기 전에 인증서를 자동으로 업데이트하려고 합니다. Ansible 플레이북의 auto_renew 매개변수를 no 로 설정하여 이 설정을 비활성화할 수 있습니다.

사전 요구 사항

- Ansible Core 패키지는 제어 시스템에 설치됩니다.
- 플레이북을 실행할 시스템에 rhel-system-roles 패키지가 설치되어 있습니다.

RHEL 시스템 역할에 대한 자세한 내용과 이를 적용하는 방법에 대한 자세한 내용은 RHEL 시스템 역할 시작하기를 참조하십시오.

절차

1. *선택 사항:* 인벤토리 파일(예: inventory.file)을 생성합니다.

```
$ touch inventory.file
```

2. 인벤토리 파일을 열고 인증서를 요청할 호스트를 정의합니다. 예를 들면 다음과 같습니다.

```
[webserver]
server.idm.example.com
```

3. 플레이북 파일을 만듭니다(예: request-certificate.yml):

- 웹 서버와 같이 인증서를 요청할 호스트를 포함하도록 호스트를 설정합니다.
- 다음을 포함하도록 certificate_requests 변수를 설정합니다.
  - name 매개변수를 mycert 와 같은 인증서의 원하는 이름으로 설정합니다.
  - dns 매개 변수를 인증서에 포함할 도메인(예: *.example.com)으로 설정합니다.
  - ca 매개 변수를 자체 서명 으로 설정합니다.
- 역할에 rhel-system- roles.certificate 역할을 설정합니다.

이 예제는 플레이북 파일입니다.

```
---
- hosts: webserver

  vars:
    certificate_requests:
      - name: mycert
        dns: "*.example.com"
        ca: self-sign

  roles:
    - rhel-system-roles.certificate
```

4. 파일을 저장합니다.

5. 플레이북을 실행합니다.

```
$ ansible-playbook -i inventory.file request-certificate.yml
```

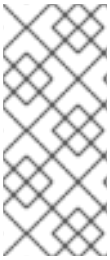
추가 리소스

- /usr/share/ansible/roles/rhel-system-roles.certificate/README.md 파일을 참조하십시오.
- ansible-playbook(1) 매뉴얼 페이지를 참조하십시오.

16.3. 인증서 문제 및 업데이트 시스템 역할을 사용하여 IDM CA에서 새 인증서 요청

인증서 문제 및 업데이트 시스템 역할을 사용하면 통합 인증 기관(CA)과 함께 IdM 서버를 사용하는 동안 anible-core 를 사용하여 인증서를 발행할 수 있습니다. 따라서 IdM을 CA로 사용할 때 여러 시스템의 인증서 신뢰 체인을 효율적이고 일관되게 관리할 수 있습니다.

이 프로세스에서는 certmonger 공급자를 사용하고 getcert 명령을 통해 인증서를 요청합니다.



참고

기본적으로 certmonger 는 만료되기 전에 인증서를 자동으로 업데이트하려고 합니다. Ansible 플레이북의 auto_renew 매개변수를 no 로 설정하여 이 설정을 비활성화할 수 있습니다.

사전 요구 사항

- Ansible Core 패키지는 제어 시스템에 설치됩니다.
- 플레이북을 실행할 시스템에 rhel-system-roles 패키지가 설치되어 있습니다.

RHEL 시스템 역할에 대한 자세한 내용과 이를 적용하는 방법에 대한 자세한 내용은 RHEL 시스템 역할 시작하기를 참조하십시오.

절차

1. *선택 사항:* 인벤토리 파일(예: inventory.file)을 생성합니다.

```
$ touch inventory.file
```

2. 인벤토리 파일을 열고 인증서를 요청할 호스트를 정의합니다. 예를 들면 다음과 같습니다.

```
[webserver]
server.idm.example.com
```

3. 플레이북 파일을 만듭니다(예: **request-certificate.yml**):

- 웹 서버와 같이 인증서를 요청할 호스트를 포함하도록 호스트를 설정합니다.
- 다음을 포함하도록 **certificate_requests** 변수를 설정합니다.
  - **name** 매개변수를 **mycert** 와 같은 인증서의 원하는 이름으로 설정합니다.
  - **dns** 매개변수를 인증서에 포함할 도메인(예: **www.example.com**)으로 설정합니다.
  - **principal** 매개 변수를 설정하여 **HTTP/www.example.com@EXAMPLE.COM** 와 같은 **Kerberos** 주체를 지정합니다.
  - **ca** 매개변수를 **ipa** 로 설정합니다.
- 역할에 **rhel-system- roles.certificate** 역할을 설정합니다.

이 예제는 플레이북 파일입니다.

```
---
- hosts: webserver
  vars:
    certificate_requests:
      - name: mycert
        dns: www.example.com
        principal: HTTP/www.example.com@EXAMPLE.COM
        ca: ipa

  roles:
    - rhel-system-roles.certificate
```

4. 파일을 저장합니다.

5. 플레이북을 실행합니다.

```
$ ansible-playbook -i inventory.file request-certificate.yml
```

추가 리소스

- /usr/share/ansible/roles/rhel-system-roles.certificate/README.md 파일을 참조하십시오.
- ansible-playbook(1) 매뉴얼 페이지를 참조하십시오.

16.4. 인증서 발행 및 업데이트 시스템 역할을 사용하여 인증서 발행 전 또는 이후에 실행할 명령 지정

인증서 시스템 문제 및 업데이트 역할을 사용하면 Ansible Core를 사용하여 인증서가 발행되거나 갱신된 후 이전과 후에 명령을 실행할 수 있습니다.

다음 예에서 관리자는 www.example.com 의 자체 서명 인증서 전에 httpd 서비스를 중지한 후 나중에 다시 시작합니다.

참고

기본적으로 certmonger 는 만료되기 전에 인증서를 자동으로 업데이트하려고 합니다. Ansible 플레이북의 auto_renew 매개변수를 no 로 설정하여 이 설정을 비활성화할 수 있습니다.

사전 요구 사항

- Ansible Core 패키지는 제어 시스템에 설치됩니다.
- 플레이북을 실행할 시스템에 rhel-system-roles 패키지가 설치되어 있습니다.

RHEL 시스템 역할에 대한 자세한 내용과 이를 적용하는 방법에 대한 자세한 내용은 RHEL 시스템 역할 시작하기를 참조하십시오.

절차

1. *선택 사항:* 인벤토리 파일(예: **inventory.file**)을 생성합니다.

```
$ touch inventory.file
```

2. 인벤토리 파일을 열고 인증서를 요청할 호스트를 정의합니다. 예를 들면 다음과 같습니다.

```
[webserver]
server.idm.example.com
```

3. 플레이북 파일을 만듭니다(예: **request-certificate.yml**):

- 웹 서버와 같이 인증서를 요청할 호스트를 포함하도록 호스트를 설정합니다.
- 다음을 포함하도록 **certificate_requests** 변수를 설정합니다.
  - **name** 매개변수를 **mycert** 와 같은 인증서의 원하는 이름으로 설정합니다.
  - **dns** 매개변수를 인증서에 포함할 도메인(예: **www.example.com**)으로 설정합니다.
  - **self-sign** 과 같은 인증서를 발급하는 데 사용할 **ca** 매개변수를 **CA**로 설정합니다.
  - 이 인증서를 발행하거나 갱신하기 전에 실행할 명령에 **run_before** 매개 변수를 **systemctl stop httpd.service** 와 같이 설정합니다.
  - 이 인증서가 발행되거나 갱신된 후 실행할 명령에 **run_after** 매개 변수를 다음과 같이 설정합니다(예: **systemctl start httpd.service**).
- 역할에 **rhel-system- roles.certificate** 역할을 설정합니다.

이 예제는 플레이북 파일입니다.

```
---
```

```
- hosts: webserver
  vars:
    certificate_requests:
      - name: mycert
        dns: www.example.com
        ca: self-sign
        run_before: systemctl stop httpd.service
        run_after: systemctl start httpd.service

  roles:
    - rhel-system-roles.certificate
```

4. 파일을 저장합니다.

5. 플레이북을 실행합니다.

```
$ ansible-playbook -i inventory.file request-certificate.yml
```

추가 리소스

- **/usr/share/ansible/roles/rhel-system-roles.certificate/README.md** 파일을 참조하십시오.
- **ansible-playbook(1)** 매뉴얼 페이지를 참조하십시오.

17장. 인증서의 하위 집합만 신뢰하도록 애플리케이션 제한

IdM(Identity Management) 설치가 통합 CA(Certificate System) CA(인증 기관)로 구성된 경우 간단한 하위 시스템을 생성할 수 있습니다. 생성하는 모든 하위 CA는 인증서 시스템의 기본 CA인 ipa CA로 하위 설정됩니다.

이 컨텍스트의 *간단한 하위* 보안 *은 하위 CA에서 특정 용도로 인증서를 발행하는* 것을 의미합니다. 예를 들어, 경량 하위 CA를 사용하면 가상 사설 네트워크(VPN) 게이트웨이 및 웹 브라우저와 같은 서비스를 구성하여 *하위 CA A* 에서 발급한 인증서만 허용할 수 있습니다. 하위 CA *B에서 발급한 인증서만 허용하도록 기타 서비스를 구성하면 하위* CA *A*, 기본 CA(기본 CA, 즉 ipa CA)에서 발급한 인증서가 수락되지 않도록 하고, 둘 사이의 중간 하위 CA입니다.

하위 CA의 중간 인증서를 취소하면 이 하위 CA에서 발급한 모든 인증서가 올바르게 구성된 클라이언트에 의해 자동으로 유효하지 않은 것으로 간주됩니다. 루트 CA, ipa 또는 다른 하위 CA에서 직접 발급한 다른 모든 인증서는 유효한 상태로 유지됩니다.

이 섹션에서는 Apache 웹 서버의 예제를 사용하여 인증서의 하위 집합만 신뢰하도록 애플리케이션을 제한하는 방법을 설명합니다. webclient-ca IdM 하위 CA에서 발급한 인증서를 사용하도록 IdM 클라이언트에서 실행 중인 웹 서버를 제한하고, 사용자가 webclient -ca IdM 하위 시스템에서 발행한 사용자 인증서를 사용하여 웹 서버를 인증하도록 하려면 이 섹션을 완료합니다.

취해야 하는 단계는 다음과 같습니다.

1. IdM 하위 CA 생성
2. IdM WebUI에서 하위 CA 인증서 다운로드
3. 사용자, 서비스 및 CA의 올바른 조합 및 사용된 인증서 프로필을 지정하는 CA ACL 생성
4. IdM 하위 CA의 IdM 클라이언트에서 실행되는 웹 서비스에 대한 인증서를 요청합니다.
5. 단일 인스턴스 Apache HTTP Server 설정
6. Apache HTTP 서버에 TLS 암호화 추가

7. Apache HTTP Server에서 지원되는 TLS 프로토콜 버전 설정

8. Apache HTTP Server에서 지원되는 암호 설정

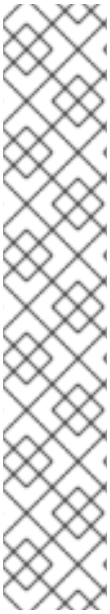
9. 웹 서버에서 TLS 클라이언트 인증서 인증 구성

10. IdM 하위 CA에서 사용자에 대한 인증서를 요청하고 클라이언트로 내보냅니다.

11. 브라우저로 사용자 인증서를 가져오고 하위 CA 인증서를 신뢰하도록 브라우저를 구성합니다.

17.1. 경량 하위 서비스 관리

이 섹션에서는 경량의 하위 인증 기관(sub-CA)을 관리하는 방법을 설명합니다. 생성하는 모든 하위 CA는 인증서 시스템의 기본 CA인 ipa CA로 하위 설정됩니다. 하위 서비스를 비활성화하고 삭제할 수도 있습니다.



참고

- 하위 CA를 삭제하면 해당 하위 서비스에 대한 해지 검사가 더 이상 작동하지 않습니다. 향후 만료 시간이 아닌 해당 하위 CA에서 발급한 인증서가 더 이상 없는 경우에만 하위 CA를 삭제합니다.
- 해당 하위 CA에서 발행한 인증서가 아직 만료되지 않은 동안에는 하위 CA만 비활성화해야 합니다. 하위 CA에서 발급한 모든 인증서가 만료된 경우 해당 하위 CA를 삭제할 수 있습니다.
- IdM CA를 비활성화하거나 삭제할 수 없습니다.

하위 서비스 관리에 대한 자세한 내용은 다음을 참조하십시오.

- IdM WebUI에서 하위 CA 생성

- IdM WebUI에서 하위 CA 삭제
- IdM CLI에서 하위 CA 생성
- IdM CLI에서 하위 CA 비활성화
- IdM CLI에서 하위 CA 삭제

17.1.1. IdM 웹 UI에서 하위 CA 생성

이 절차에서는 **IdM WebUI**를 사용하여 **webserver-ca** 및 **webclient-ca** 라는 새 하위 시스템을 생성하는 방법을 설명합니다.

사전 요구 사항

- 관리자의 자격 증명을 확보했는지 확인합니다.

절차

1. 인증 메뉴에서 인증서 를 클릭합니다.
2. 인증 기관을 선택하고 추가 를 클릭합니다.
3. **webserver-ca** 하위 **CA**의 이름을 입력합니다. 주체 **DN** (예: **CN=WEBSERVER,O=IDM.EXAMPLE.COM**)을 주체 **DN** 필드에 입력합니다. 주체 **DN**은 **IdM CA** 인프라에서 고유해야 합니다.
4. **webclient-ca** 하위 **CA**의 이름을 입력합니다. 주체 **DN** 필드에 **CN=WEBCLIENT,O=IDM.EXAMPLE.COM** 을 입력합니다.
5. 명령행 인터페이스에서 **ipa-certupdate** 명령을 실행하여 **webserver-ca** 및 **webclient-ca** 하위 인증서에 대한 **certmonger** 추적 요청을 생성합니다.

```
[root@ipaserver ~]# ipa-certupdate
```

-

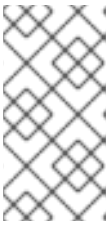
중요

하위 인증서를 생성한 후 **ipa-certupdate** 명령을 실행하는 것을 잊어 버린 경우, 하위 인증서가 만료되지 않은 경우에도 하위 **CA**에서 발행한 엔드-엔티 인증서가 유효하지 않은 것으로 간주됩니다.

검증

- 새 하위 **CA**의 서명 인증서가 **IdM** 데이터베이스에 추가되었는지 확인합니다.

```
[root@ipaserver ~]# certutil -d /etc/pki/pki-tomcat/alias/ -L

Certificate Nickname                    Trust Attributes
                                        SSL,S/MIME,JAR/XPI

caSigningCert cert-pki-ca               CTu,Cu,Cu
Server-Cert cert-pki-ca                 u,u,u
auditSigningCert cert-pki-ca            u,u,Pu
caSigningCert cert-pki-ca ba83f324-5e50-4114-b109-acca05d6f1dc u,u,u
ocspSigningCert cert-pki-ca             u,u,u
subsystemCert cert-pki-ca               u,u,u
```



참고

새 하위 **CA** 인증서는 인증서 시스템 인스턴스가 설치된 모든 복제본으로 자동 전송됩니다.

17.1.2. IdM 웹 UI에서 하위 CA 삭제

이 절차에서는 **IdM WebUI**에서 간단한 하위 **CA**를 삭제하는 방법을 설명합니다.

참고

- 하위 CA를 삭제하면 해당 하위 서비스에 대한 해지 검사가 더 이상 작동하지 않습니다. 향후 만료 시간이 아닌 해당 하위 CA에서 발급한 인증서가 더 이상 없는 경우에만 하위 CA를 삭제합니다.
- 해당 하위 CA에서 발행한 인증서가 아직 만료되지 않은 동안에는 하위 CA만 비활성화해야 합니다. 하위 CA에서 발급한 모든 인증서가 만료된 경우 해당 하위 CA를 삭제할 수 있습니다.
- IdM CA를 비활성화하거나 삭제할 수 없습니다.

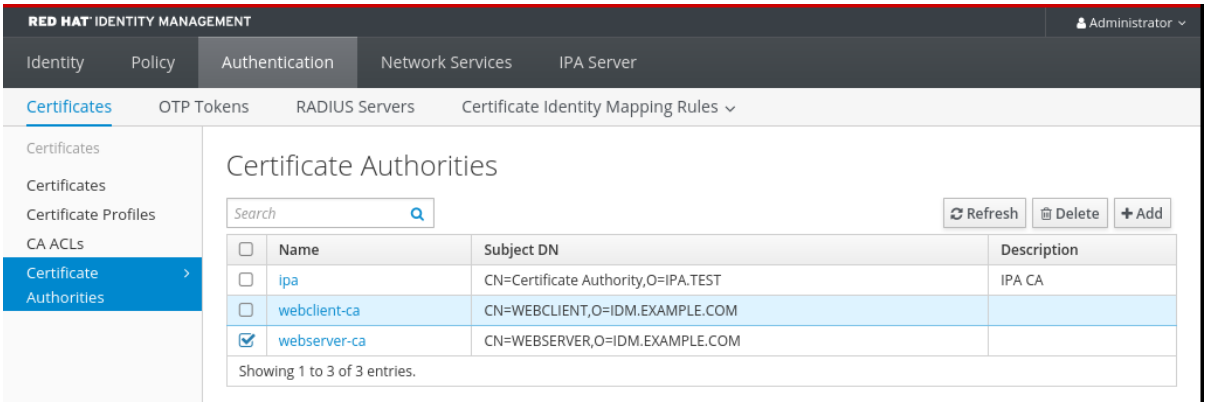
사전 요구 사항

- 관리자의 자격 증명을 확보했는지 확인합니다.
- IdM CLI에서 하위 CA를 비활성화했습니다. IdM CLI에서 하위 서비스 비활성화 단원을참조하십시오.

절차

1. IdM WebUI에서 인증 탭을 열고 인증서 하위 탭을 선택합니다.
2. 인증 기관을 선택합니다.
3. 제거할 하위 요소를 선택하고 삭제를 클릭합니다.

그림 17.1. IdM 웹 UI에서 하위 CA 삭제



4. 삭제를 클릭하여 확인합니다.

하위 서비스는 인증 기관 목록에서 제거됩니다.

17.1.3. IdM CLI에서 하위 서비스 생성

이 절차에서는 IdM CLI를 사용하여 webserver-ca 및 webclient-ca 라는 새 하위 시스템을 생성하는 방법을 설명합니다.

사전 요구 사항

- 관리자의 자격 증명을 획득했는지 확인합니다.
- CA 서버인 IdM 서버에 로그인했는지 확인합니다.

절차

1. ipa ca-add 명령을 입력하고 webserver-ca 하위 CA의 이름과 해당 Subject Distinguished Name(DN)을 지정합니다.

```
[root@ipaserver ~]# ipa ca-add webserver-ca --
subject="CN=WEBSERVER,O=IDM.EXAMPLE.COM"
------------------
Created CA "webserver-ca"
------------------
  Name: webserver-ca
  Authority ID: ba83f324-5e50-4114-b109-acca05d6f1dc
  Subject DN: CN=WEBSERVER,O=IDM.EXAMPLE.COM
  Issuer DN: CN=Certificate Authority,O=IDM.EXAMPLE.COM
```

이름

CA의 이름입니다.

권한 ID

CA의 개별 ID가 자동으로 생성됩니다.

제목 dn

DN(subject Distinguished Name)입니다. 주체 DN은 IdM CA 인프라에서 고유해야

합니다.

발행자 DN

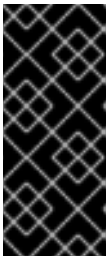
하위 CA 인증서를 발급한 상위 CA입니다. 모든 하위 서비스는 IdM 루트 CA의 하위로 생성됩니다.

2. 웹 클라이언트에 인증서를 발행할 webclient-ca 하위 CA를 생성합니다.

```
[root@ipaserver ~]# ipa ca-add webclient-ca --
subject="CN=WEBCLIENT,O=IDM.EXAMPLE.COM"
------------------
Created CA "webclient-ca"
------------------
  Name: webclient-ca
  Authority ID: 8a479f3a-0454-4a4d-8ade-fd3b5a54ab2e
  Subject DN: CN=WEBCLIENT,O=IDM.EXAMPLE.COM
  Issuer DN: CN=Certificate Authority,O=IDM.EXAMPLE.COM
```

3. ipa-certupdate 명령을 실행하여 webserver-ca 및 webclient-ca 하위 CA 인증서에 대한 certmonger 추적 요청을 생성합니다.

```
[root@ipaserver ~]# ipa-certupdate
```



중요

하위 CA 및 하위 CA 인증서를 생성한 후 ipa-certupdate 명령을 실행하는 것을 잊어 버린 경우, 엔드-센티 인증서가 만료되지 않은 경우에도 해당 하위 계정에서 발급한 엔드-엔티 인증서가 유효하지 않은 것으로 간주됩니다.

검증 단계

- 새 하위 CA의 서명 인증서가 IdM 데이터베이스에 추가되었는지 확인합니다.

```
[root@ipaserver ~]# certutil -d /etc/pki/pki-tomcat/alias/ -L

Certificate Nickname                    Trust Attributes
                                        SSL,S/MIME,JAR/XPI

caSigningCert cert-pki-ca               CTu,Cu,Cu
Server-Cert cert-pki-ca                 u,u,u
auditSigningCert cert-pki-ca            u,u,Pu
caSigningCert cert-pki-ca ba83f324-5e50-4114-b109-acca05d6f1dc u,u,u
ocspSigningCert cert-pki-ca             u,u,u
subsystemCert cert-pki-ca               u,u,u
```

-

참고

새 하위 CA 인증서는 인증서 시스템 인스턴스가 설치된 모든 복제본으로 자동 전송됩니다.

17.1.4. IdM CLI에서 하위 서비스 비활성화

이 절차에서는 IdM CLI에서 하위 CA를 비활성화하는 방법을 설명합니다. 하위 CA에서 발급한 만료되지 않은 인증서가 여전히 있는 경우 삭제할 수는 없지만 비활성화할 수 있습니다. 서브-CA를 삭제하면 해당 하위 서비스에 대한 해지 검사가 더 이상 작동하지 않습니다.

사전 요구 사항

- 관리자의 자격 증명을 확보했는지 확인합니다.

절차

1. ipa ca-find 명령을 실행하여 삭제 중인 하위 CA의 이름을 확인합니다.

```
[root@ipaserver ~]# ipa ca-find
-------------
3 CAs matched
-------------
  Name: ipa
  Description: IPA CA
  Authority ID: 5195deaf-3b61-4aab-b608-317aff38497c
  Subject DN: CN=Certificate Authority,O=IPA.TEST
  Issuer DN: CN=Certificate Authority,O=IPA.TEST

  Name: webclient-ca
  Authority ID: 605a472c-9c6e-425e-b959-f1955209b092
  Subject DN: CN=WEBCLIENT,O=IDM.EXAMPLE.COM
  Issuer DN: CN=Certificate Authority,O=IPA.TEST

 Name: webserver-ca
  Authority ID: 02d537f9-c178-4433-98ea-53aa92126fc3
  Subject DN: CN=WEBSERVER,O=IDM.EXAMPLE.COM
  Issuer DN: CN=Certificate Authority,O=IPA.TEST
----------------------------
Number of entries returned 3
----------------------------
```

2. ipa ca-disable 명령을 실행하여 하위 CA(이 예에서는 webserver-ca)를 비활성화합니다.

-

```
ipa ca-disable webserver-ca
-------------------------
Disabled CA "webserver-ca"
-------------------------
```

17.1.5. IdM CLI에서 하위 CA 삭제

다음 절차에서는 **IdM CLI**에서 간단한 하위 **CA**를 삭제하는 방법을 설명합니다.

참고

- 하위 **CA**를 삭제하면 해당 하위 서비스에 대한 해지 검사가 더 이상 작동하지 않습니다. 향후 만료 시간이 아닌 해당 하위 **CA**에서 발급한 인증서가 더 이상 없는 경우에만 하위 **CA**를 삭제합니다.
- 해당 하위 **CA**에서 발행한 인증서가 아직 만료되지 않은 동안에는 하위 **CA**만 비활성화해야 합니다. 하위 **CA**에서 발급한 모든 인증서가 만료된 경우 해당 하위 **CA**를 삭제할 수 있습니다.
- **IdM CA**를 비활성화하거나 삭제할 수 없습니다.

사전 요구 사항

- 관리자의 자격 증명을 확보했는지 확인합니다.

절차

1. 하위 **CA** 및 **CA** 목록을 표시하려면 **ipa ca-find** 명령을 실행합니다.

```
# ipa ca-find
-------------
3 CAs matched
-------------
  Name: ipa
  Description: IPA CA
  Authority ID: 5195deaf-3b61-4aab-b608-317aff38497c
  Subject DN: CN=Certificate Authority,O=IPA.TEST
  Issuer DN: CN=Certificate Authority,O=IPA.TEST

  Name: webclient-ca
  Authority ID: 605a472c-9c6e-425e-b959-f1955209b092
  Subject DN: CN=WEBCLIENT,O=IDM.EXAMPLE.COM
```

```
  Issuer DN: CN=Certificate Authority,O=IPA.TEST

 Name: webserver-ca
  Authority ID: 02d537f9-c178-4433-98ea-53aa92126fc3
  Subject DN: CN=WEBSERVER,O=IDM.EXAMPLE.COM
  Issuer DN: CN=Certificate Authority,O=IPA.TEST
----------------------------
Number of entries returned 3
----------------------------
```

2. **ipa ca-disable** 명령을 실행하여 하위 **CA(**이 예에서는 **webserver-ca)**를 비활성화합니다**.**

```
# ipa ca-disable webserver-ca
-------------------------
Disabled CA "webserver-ca"
-------------------------
```

3. 이 예에서 **sub-CA**를 삭제합니다**. webserver-ca:**

```
# ipa ca-del webserver-ca
------------------------
Deleted CA "webserver-ca"
------------------------
```

검증

- **ipa ca-find** 를 실행하여 **CA** 및 하위 **CA** 목록을 표시합니다**. webserver-ca** 는 더 이상 목록에 없습니다**.**

```
# ipa ca-find
-------------
2 CAs matched
-------------
  Name: ipa
  Description: IPA CA
  Authority ID: 5195deaf-3b61-4aab-b608-317aff38497c
  Subject DN: CN=Certificate Authority,O=IPA.TEST
  Issuer DN: CN=Certificate Authority,O=IPA.TEST

  Name: webclient-ca
  Authority ID: 605a472c-9c6e-425e-b959-f1955209b092
  Subject DN: CN=WEBCLIENT,O=IDM.EXAMPLE.COM
  Issuer DN: CN=Certificate Authority,O=IPA.TEST
----------------------------
Number of entries returned 2
----------------------------
```

17.2. IDM WEBUI에서 하위 CA 인증서 다운로드

사전 요구 사항

- IdM 관리자의 자격 증명을 취득했는지 확인합니다.

절차

1. 인증 메뉴에서 인증서 > 인증서를 클릭합니다.

그림 17.2. 인증서 목록의 하위 CA 인증서

2. 하위 CA 인증서의 일련 번호를 클릭하여 인증서 정보 페이지를 엽니다.

3. 인증서 정보 페이지에서 작업 > 다운로드를 클릭합니다.

4. CLI에서 하위 CA 인증서를 /etc/pki/tls/private/ 디렉터리로 이동합니다.

```
# mv path/to/the/downloaded/certificate /etc/pki/tls/private/sub-ca.crt
```

17.3. 웹 서버 및 클라이언트 인증을 위한 CA ACL 생성

CA ACL(인증 기관 액세스 제어 목록) 규칙은 사용자, 서비스 또는 호스트를 발급하는 데 사용할 수 있는 프로필을 정의합니다. CA ACL을 사용하면 프로필, 주체 및 그룹을 연결하면 보안 주체 또는 그룹이 특정 프로필을 사용하여 인증서를 요청할 수 있습니다.

예를 들어, CA ACL을 사용하여, 관리자는 런던에 위치한 사무실에서 런던 관련 그룹의 멤버인 사용자에게만 사용할 수 있는 프로필 사용을 제한할 수 있습니다.

17.3.1. IdM CLI에서 CA ACL 보기

IdM 배포에서 사용 가능한 CA ACL(인증 기관 액세스 제어 목록) 목록과 특정 CA ACL의 세부 정보를 보려면 이 섹션을 완료합니다.

절차

1. IdM 환경의 모든 CA ACL을 보려면 ipa caacl-find 명령을 입력합니다.

```
$ ipa caacl-find
----------------
1 CA ACL matched
----------------
  ACL name: hosts_services_calPAserviceCert
  Enabled: TRUE
```

2. CA ACL의 세부 정보를 보려면 ipa caacl-show 명령을 입력하고 CA ACL 이름을 지정합니다. 예를 들어 hosts_services_calPAserviceCert CA ACL의 세부 정보를 보려면 다음을 입력합니다.

```
$ ipa caacl-show hosts_services_calPAserviceCert
  ACL name: hosts_services_calPAserviceCert
  Enabled: TRUE
  Host category: all
  Service category: all
  CAs: ipa
  Profiles: calPAserviceCert
  Users: admin
```

17.3.2. webserver-ca에서 발급한 인증서를 사용하여 웹 클라이언트에 인증하는 웹 서버의 CA ACL 생성

이 섹션에서는 시스템 관리자가 HTTP/my_company.idm.example.com@IDM.EXAMPLE.COM 서비스에 대한 인증서를 요청할 때 webserver-ca 하위 CA 및 calPAserviceCert 프로필을 사용해야 하는 CA ACL을 생성하는 방법에 대해 설명합니다. 사용자가 다른 하위 CA 또는 다른 프로필의 인증서를 요청하면 요청이 실패합니다. 유일한 예외는 활성화된 또 다른 일치하는 CA ACL이 있는 경우입니다. 사용 가능한 CA ACL을 보려면 IdM CLI에서 CA ACL 보기를 참조하십시오.

사전 요구 사항

- HTTP/my_company.idm.example.com@IDM.EXAMPLE.COM 서비스가 IdM의 일부인지 확인합니다.
- IdM 관리자의 자격 증명을 취득했는지 확인합니다.

절차

1. ipa caacl 명령을 사용하여 CA ACL을 생성하고 해당 이름을 지정합니다.

```
$ ipa caacl-add TLS_web_server_authentication
```

```
----------------------------------------
Added CA ACL "TLS_web_server_authentication"
----------------------------------------
  ACL name: TLS_web_server_authentication
  Enabled: TRUE
```

2. **ipa caacl-mod** 명령을 사용하여 **CA ACL**을 수정하여 **CA ACL**에 대한 설명을 지정합니다.

```
$ ipa caacl-mod TLS_web_server_authentication --desc="CAACL for web servers
authenticating to web clients using certificates issued by webserver-ca"
----------------------------------------
Modified CA ACL "TLS_web_server_authentication"
----------------------------------------
  ACL name: TLS_web_server_authentication
  Description: CAACL for web servers authenticating to web clients using certificates
issued by webserver-ca
  Enabled: TRUE
```

3. **webserver-ca** 하위 **CA**를 **CA ACL**에 추가합니다.

```
$ ipa caacl-add-ca TLS_web_server_authentication --ca=webserver-ca
  ACL name: TLS_web_server_authentication
  Description: CAACL for web servers authenticating to web clients using certificates
issued by webserver-ca
  Enabled: TRUE
  CAs: webserver-ca
------------------------
Number of members added 1
------------------------
```

4. **ipa caacl-add-service** 를 사용하여 보안 주체가 인증서를 요청할 수 있는 서비스를 지정합니다.

```
$ ipa caacl-add-service TLS_web_server_authentication --
service=HTTP/my_company.idm.example.com@IDM.EXAMPLE.COM
  ACL name: TLS_web_server_authentication
  Description: CAACL for web servers authenticating to web clients using certificates
issued by webserver-ca
  Enabled: TRUE
  CAs: webserver-ca
  Services: HTTP/my_company.idm.example.com@IDM.EXAMPLE.COM
------------------------
Number of members added 1
------------------------
```

5. **ipa caacl-add-profile** 명령을 사용하여 요청된 인증서에 대한 인증서 프로필을 지정합니다.

```
$ ipa caacl-add-profile TLS_web_server_authentication --
certprofiles=caIPAserviceCert
  ACL name: TLS_web_server_authentication
  Description: CAACL for web servers authenticating to web clients using certificates
issued by webserver-ca
  Enabled: TRUE
  CAs: webserver-ca
  Profiles: caIPAserviceCert
  Services: HTTP/my_company.idm.example.com@IDM.EXAMPLE.COM
-------------------------
Number of members added 1
-------------------------
```

새로 생성된 **CA ACL**을 즉시 사용할 수 있습니다. 기본적으로 생성 후 활성화됩니다.

참고

CA ACL은 특정 주체 또는 그룹의 요청에 허용되는 **CA** 및 프로필 조합을 지정하는 것입니다. **CA ACL**은 인증서 검증 또는 신뢰에 영향을 미치지 않습니다. 발급된 인증서 사용 방법에는 영향을 미치지 않습니다.

17.3.3. webclient-ca에서 발급한 인증서를 사용하여 웹 서버에 인증하는 사용자 웹 브라우저에 대한 CA ACL 생성

이 섹션에서는 시스템 관리자가 인증서를 요청할 때 **webclient-ca** 하위 **CA** 및 **IECUserRoles** 프로필을 사용해야 하는 **CA ACL**을 생성하는 방법에 대해 설명합니다. 사용자가 다른 하위 **CA** 또는 다른 프로필의 인증서를 요청하면 요청이 실패합니다. 유일한 예외는 활성화된 또 다른 일치하는 **CA ACL**이 있는 경우입니다. 사용 가능한 **CA ACL**을 보려면 IdM CLI에서 CA ACL 보기를 참조하십시오.

사전 요구 사항

- **IdM** 관리자의 자격 증명을 취득했는지 확인합니다.

절차

1. **ipa caacl** 명령을 사용하여 **CA ACL**을 생성하고 해당 이름을 지정합니다.

```
$ ipa caacl-add TLS_web_client_authentication
----------------------------------------
Added CA ACL "TLS_web_client_authentication"
----------------------------------------
  ACL name: TLS_web_client_authentication
  Enabled: TRUE
```

2. **ipa caacl-mod** 명령을 사용하여 **CA ACL**을 수정하여 **CA ACL**에 대한 설명을 지정합니다.

```
$ ipa caacl-mod TLS_web_client_authentication --desc="CAACL for user web
browsers authenticating to web servers using certificates issued by webclient-ca"
-----------------------------------------------
Modified CA ACL "TLS_web_client_authentication"
-----------------------------------------------
  ACL name: TLS_web_client_authentication
  Description: CAACL for user web browsers authenticating to web servers using
certificates issued by webclient-ca
  Enabled: TRUE
```

3. **webclient-ca** 하위 **CA**를 **CA ACL**에 추가합니다.

```
$ ipa caacl-add-ca TLS_web_client_authentication --ca=webclient-ca
  ACL name: TLS_web_client_authentication
  Description: CAACL for user web browsers authenticating to web servers using
certificates issued by webclient-ca
  Enabled: TRUE
  CAs: webclient-ca
------------------------
Number of members added 1
------------------------
```

4. **ipa caacl-add-profile** 명령을 사용하여 요청된 인증서에 대한 인증서 프로필을 지정합니다.

```
$ ipa caacl-add-profile TLS_web_client_authentication --certprofiles=IECUserRoles
  ACL name: TLS_web_client_authentication
  Description: CAACL for user web browsers authenticating to web servers using
certificates issued by webclient-ca
  Enabled: TRUE
  CAs: webclient-ca
  Profiles: IECUserRoles
------------------------
Number of members added 1
------------------------
```

5. **ipa caacl-mod** 명령을 사용하여 **CA ACL**을 수정하여 모든 **IdM** 사용자에게 **CA ACL**을 적용하도록 지정합니다.

```
$ ipa caacl-mod TLS_web_client_authentication --usercat=all
-----------------------------------------------
Modified CA ACL "TLS_web_client_authentication"
-----------------------------------------------
  ACL name: TLS_web_client_authentication
  Description: CAACL for user web browsers authenticating to web servers using
certificates issued by webclient-ca
```

```
Enabled: TRUE
User category: all
CAs: webclient-ca
Profiles: IECUserRoles
```

새로 생성된 CA ACL을 즉시 사용할 수 있습니다. 기본적으로 생성 후 활성화됩니다.

참고

CA ACL은 특정 주체 또는 그룹의 요청에 허용되는 CA 및 프로필 조합을 지정하는 것입니다. CA ACL은 인증서 검증 또는 신뢰에 영향을 미치지 않습니다. 발급된 인증서 사용 방법에는 영향을 미치지 않습니다.

17.4. CERTMONGER를 사용하여 서비스에 대한 IDM 인증서 가져오기

IdM 클라이언트에서 실행되는 브라우저와 웹 서비스 간의 통신을 보호 및 암호화하려면 TLS 인증서를 사용하십시오. webserver-ca 하위 계정에서 발급한 인증서를 신뢰하도록 웹 브라우저에 제한하지만 다른 IdM 하위 CA는 없는 경우 webserver-ca 하위 CA에서 웹 서비스에 대한 TLS 인증서를 받으십시오.

이 섹션에서는 certmonger 를 사용하여 IdM 클라이언트에서 실행되는 서비스 (HTTP/my_company.idm.example.com@IDM.EXAMPLE.COM)에 대한 IdM 인증서를 가져오는 방법을 설명합니다.

certmonger 를 사용하여 인증서를 자동으로 요청하면 certmonger 가 인증서를 갱신할 때 인증서를 관리하고 갱신할 수 있습니다.

certmonger 가 서비스 인증서를 요청할 때 발생하는 상황을 시각적으로 표현하려면 서비스 인증서를 요청하는 certmonger의 통신 흐름을 참조하십시오.

사전 요구 사항

- 웹 서버는 IdM 클라이언트로 등록됩니다.
- 프로시저를 실행하는 IdM 클라이언트에 대한 루트 액세스 권한이 있습니다.
- 인증서를 요청하는 서비스는 IdM에 사전 존재할 필요가 없습니다.

절차

1. HTTP 서비스가 실행 중인 my_company.idm.example.com IdM 클라이언트에서 HTTP/my_company.idm.example.com@IDM.EXAMPLE.COM principal에 해당하는 서비스의 인증서를 요청하고 이를 지정합니다.

- 인증서는 로컬 /etc/pki/tls/certs/httpd.pem 파일에 저장됩니다.
- 개인 키는 로컬 /etc/pki/tls/private/httpd.key 파일에 저장해야 합니다.
- webserver-ca 하위 시스템은 발급 인증 기관입니다.
- SubjectAltName 에 대한 extensionRequest가 my_company.idm.example.com 의 DNS 이름으로 서명 요청에 추가됩니다.

```
# ipa-getcert request -K HTTP/my_company.idm.example.com -k
/etc/pki/tls/private/httpd.key -f /etc/pki/tls/certs/httpd.pem -g 2048 -D
my_company.idm.example.com -X webserver-ca -C "systemctl restart httpd"
New signing request "20190604065735" added.
```

위 명령에서 다음을 수행합니다.

- ipa-getcert request 명령은 IdM CA에서 인증서를 가져올 수 있도록 지정합니다. ipa-getcert request 명령은 getcert request -c IPA 를 위한 바로 가기입니다.
- g 옵션은 아직 없는 경우 생성할 키 크기를 지정합니다.
- -D 옵션은 요청에 추가할 SubjectAltName DNS 값을 지정합니다.
- X 옵션 은 인증서 발행자가 ipa 가 아닌 webserver-ca 여야 함을 지정합니다.
- c 옵션은 인증서를 가져온 후 certmonger 에 httpd 서비스를 다시 시작하도록 지시합니다.

- 특정 프로필로 인증서를 발급하도록 지정하려면 **-T** 옵션을 사용합니다.

참고

RHEL 8은 **RHEL 7**에서 사용되는 **Apache**와 다른 **SSL** 모듈을 사용합니다. **SSL** 모듈은 **NSS**가 아닌 **OpenSSL**에 의존합니다. 이러한 이유로 **RHEL 8**에서는 **NSS** 데이터베이스를 사용하여 **HTTPS** 인증서와 개인 키를 저장할 수 없습니다.

2. 필요한 경우 요청 상태를 확인하려면 다음을 수행합니다.

```
# ipa-getcert list -f /etc/pki/tls/certs/httpd.pem
Number of certificates and requests being tracked: 3.
Request ID '20190604065735':
   status: MONITORING
   stuck: no
   key pair storage: type=FILE,location='/etc/pki/tls/private/httpd.key'
   certificate: type=FILE,location='/etc/pki/tls/certs/httpd.crt'
   CA: IPA
   issuer: CN=WEBSERVER,O=IDM.EXAMPLE.COM

[...]
```

출력에 요청이 **MONITORING** 상태임을 보여줍니다. 즉, 인증서를 가져왔음이 표시됩니다. 키 쌍과 인증서의 위치는 요청된 위치입니다.

17.5. 서비스 인증서를 요청하는 CERTMONGER의 통신 흐름

이 섹션의 다이어그램은 **certmonger** 가 **IdM(Identity Management)** 인증 기관**(CA)** 서버에서 서비스 인증서를 요청할 때 발생하는 단계의 단계를 보여줍니다. 시퀀스는 다음 다이어그램으로 구성됩니다.

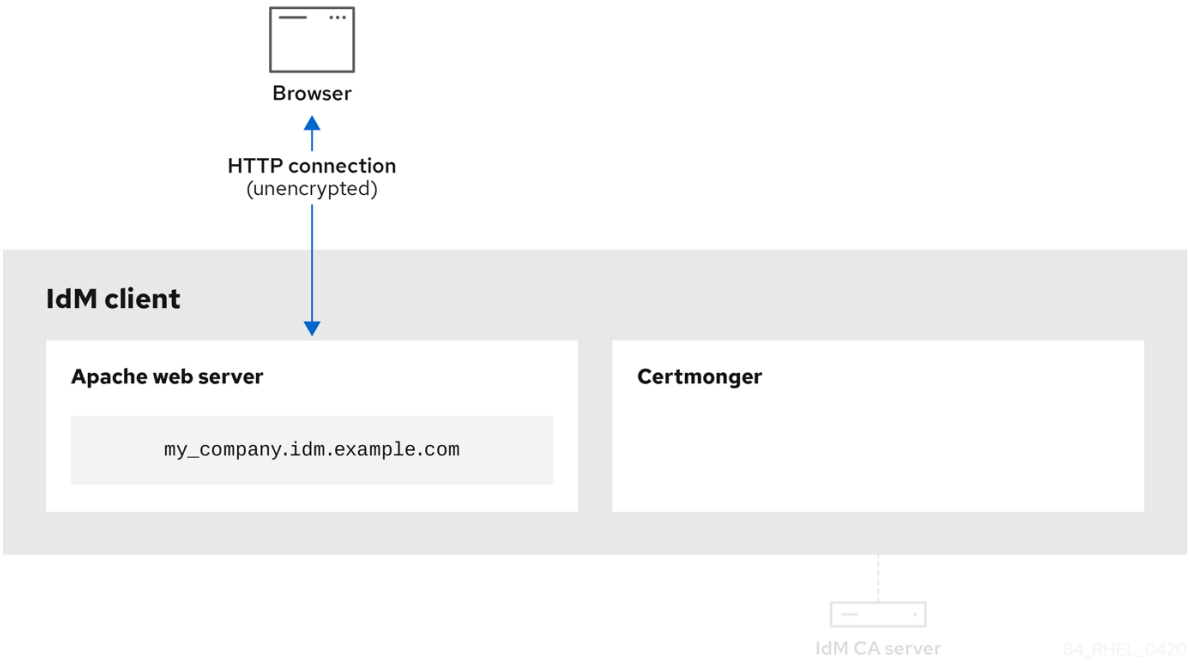
- 암호화되지 않은 통신
- **certmonger** 서비스 인증서 요청
- **IdM CA**가 서비스 인증서를 발급합니다.

- certmonger 서비스 인증서 적용
- certmonger 이전 인증서가 만료될 때 새 인증서를 요청

다이어그램에서 **webserver-ca** 하위 **CA**는 일반 **IdM CA** 서버로 표시됩니다.

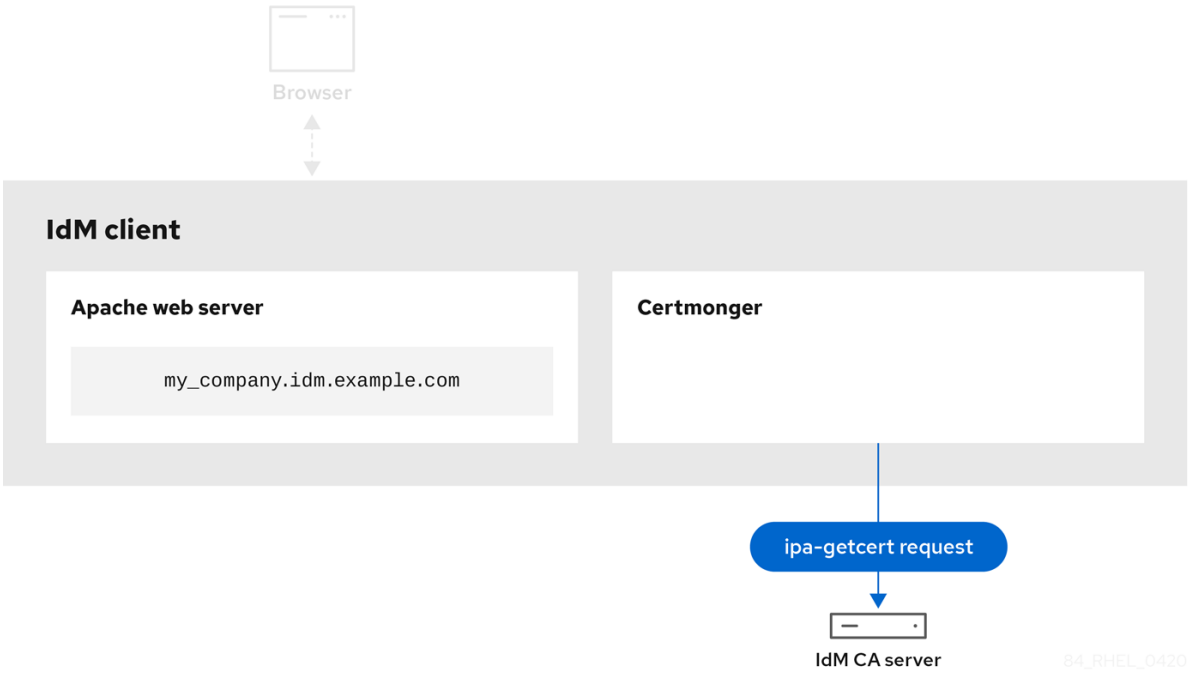
암호화되지 않은 통신은 초기 상황을 보여줍니다. **HTTPS** 인증서가 없으면 웹 서버와 브라우저 간의 통신이 암호화되지 않습니다.

그림 **17.3.** 암호화되지 않은 통신



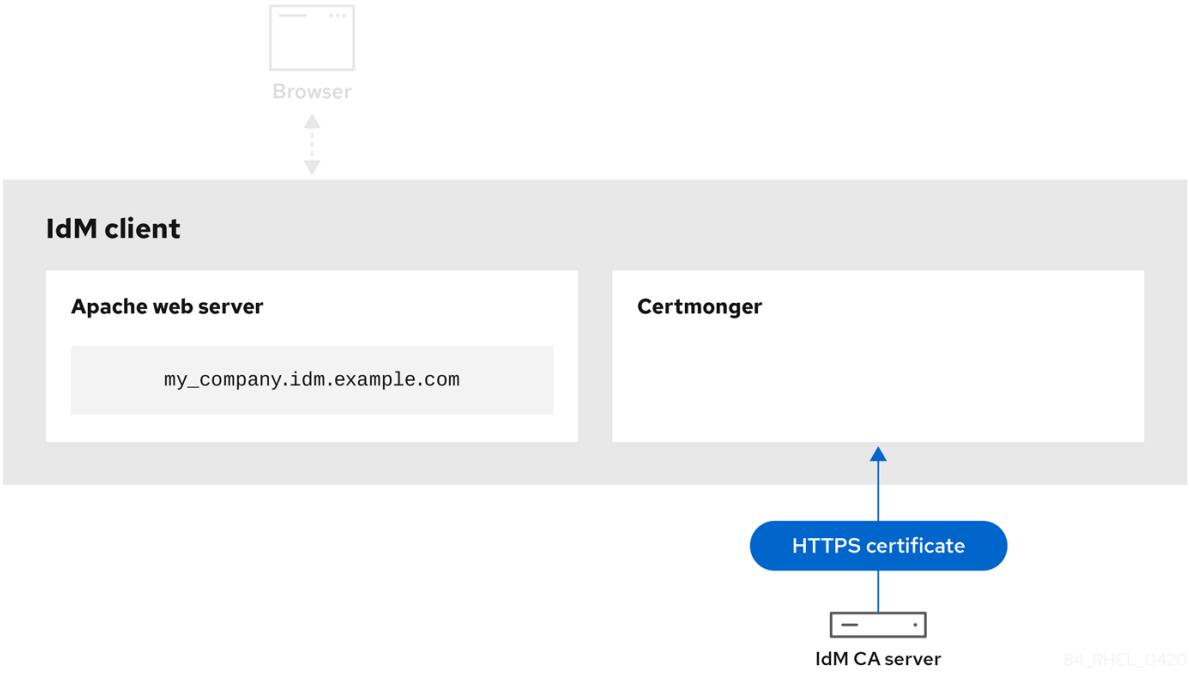
certmonger에서 서비스 인증서를 요청하는 것은 **certmonger** 를 사용하여 **Apache** 웹 서버에 대한 **HTTPS** 인증서를 수동으로 요청하는 시스템 관리자에게 표시됩니다. 웹 서버 인증서를 요청할 때 **certmonger**는 **CA**와 직접 통신하지 않습니다. **IdM**을 통해 프록시합니다.

그림 17.4. 서비스 인증서를 요청하는 certmonger



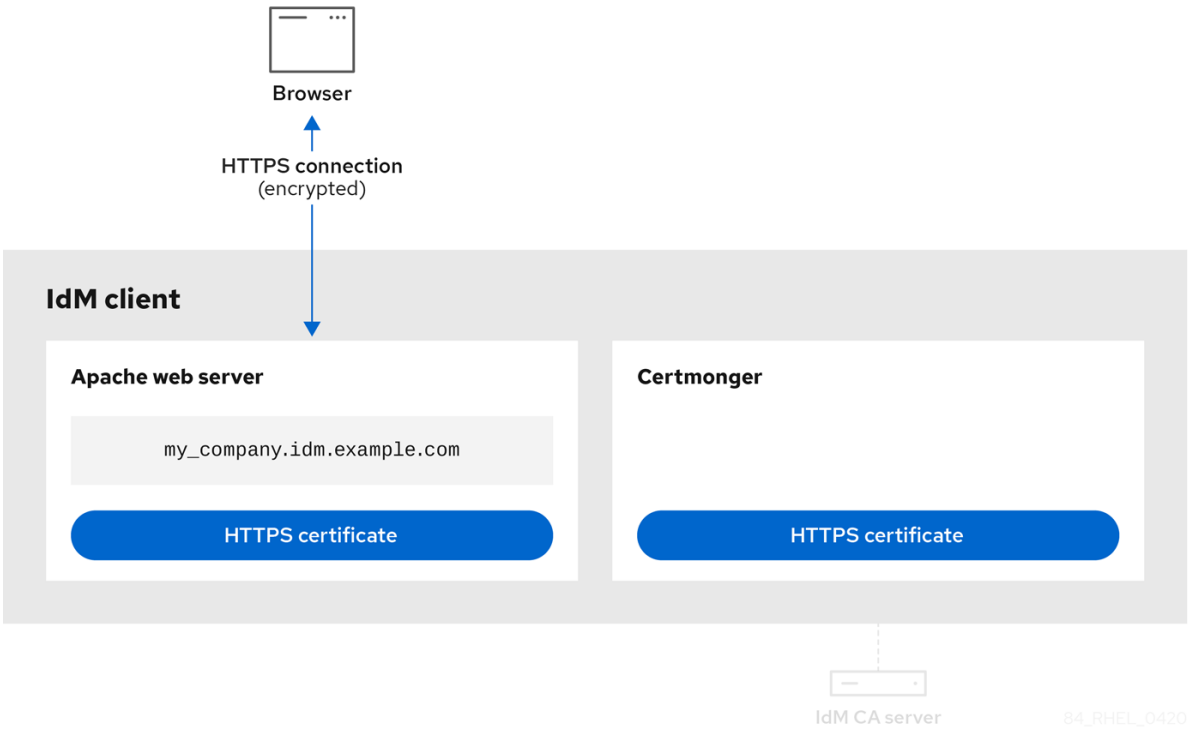
서비스 인증서를 발행하는 IdM CA 에서 웹 서버의 HTTPS 인증서를 발행하는 IdM CA가 표시됩니다.

그림 17.5. 서비스 인증서 발행 IdM CA



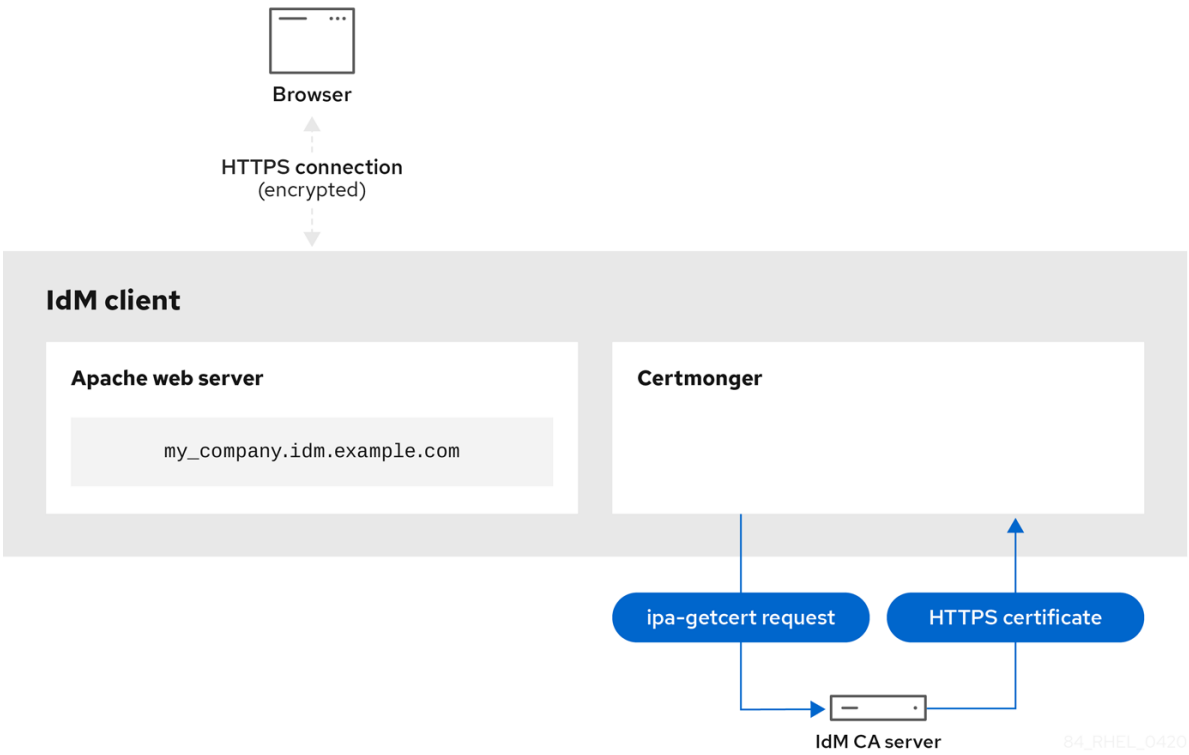
certmonger 에서 서비스 인증서를 적용하는 경우 IdM 클라이언트의 적절한 위치에 HTTPS 인증서를 배치하고 이를 수행하라는 지침이 나타나면 httpd 서비스를 다시 시작합니다. 이후 Apache 서버는 HTTPS 인증서를 사용하여 자체와 브라우저 간의 트래픽을 암호화합니다.

그림 17.6. 서비스 인증서 적용 certmonger



certmonger에서 이전 만료일이 가까워지면 certmonger 에서 인증서가 만료되기 전에 IdM CA에서 서비스 인증서 갱신을 자동으로 요청합니다. IdM CA는 새 인증서를 발행합니다.

그림 17.7. 이전 인증서가 만료될 때 새 인증서를 요청합니다.



17.6. 단일 인스턴스 APACHE HTTP SERVER 설정

이 섹션에서는 정적 **HTML** 콘텐츠를 제공하기 위해 단일 인스턴스 **Apache HTTP Server**를 설정하는 방법을 설명합니다.

웹 서버가 서버와 연결된 모든 도메인에 동일한 콘텐츠를 제공해야 하는 경우 이 섹션의 절차를 따릅니다. 도메인마다 다른 콘텐츠를 제공하려면 이름 기반 가상 호스트를 설정합니다. 자세한 내용은 Apache 이름 기반 가상 호스트 구성을 참조하십시오.

절차

1. **httpd** 패키지를 설치합니다.

```
# yum install httpd
```

2. 로컬 방화벽에서 **TCP** 포트 **80**을 엽니다.

```
# firewall-cmd --permanent --add-port=80/tcp
# firewall-cmd --reload
```

3. **httpd** 서비스를 활성화하고 시작합니다.

```
# systemctl enable --now httpd
```

4. 선택 사항: **HTML** 파일을 **/var/www/html/** 디렉터리에 추가합니다.

참고

콘텐츠를 **/var/www/html/**에 추가할 때 기본적으로 **httpd**가 실행되는 사용자가 파일 및 디렉터리를 읽을 수 있어야 합니다. 콘텐츠 소유자는 **root** 사용자 및 **root** 사용자 그룹 또는 다른 사용자 또는 관리자가 선택한 그룹일 수 있습니다. 콘텐츠 소유자가 **root** 사용자 및 **root** 사용자 그룹인 경우 다른 사용자가 파일을 읽을 수 있어야 합니다. 모든 파일 및 디렉터리에 대한 **SELinux** 컨텍스트는 기본적으로 **/var/www** 디렉터리 내의 모든 콘텐츠에 적용되는 **httpd_sys_content_t**이어야 합니다.

검증 단계

- 웹 브라우저에 **http://my_company.idm.example.com/** 또는 **http://*server_IP*/** 에 연결합니다.

/var/www/html/ 디렉터리가 비어 있거나 **index.html** 또는 **index.htm** 파일이 포함되어 있지 않은 경우 **Apache**에서 **Red Hat Enterprise Linux Test Page**를 표시합니다. **/var/www/html/** 에 다른 이름이 있는 **HTML** 파일이 포함된 경우 **http://*server_IP*/*example.html***또는 **http://my_company.idm.example.com/*example.html*** 과 같은 해당 파일에 **URL**을 입력하여 로드할 수 있습니다.

추가 리소스

- **Apache** 설명서를 참조하십시오. **Apache HTTP Server** 수동 설치를 참조하십시오.
- **httpd.service(8)** 매뉴얼 페이지를 참조하십시오.

17.7. APACHE HTTP SERVER에 TLS 암호화 추가

이 섹션에서는 **idm.example.com** 도메인의 **my_company.idm.example.com Apache HTTP Server**에서 **TLS** 암호화를 활성화하는 방법을 설명합니다.

사전 요구 사항

- my_company.idm.example.com Apache HTTP Server가 설치되고 실행됩니다.
- webserver-ca 하위 CA에서 TLS 인증서를 가져와 17.4절. “certmonger를 사용하여 서비스에 대한 IdM 인증서 가져오기” 에 설명된 대로 /etc/pki/tls/certs/httpd.pem 파일에 저장했습니다. 다른 경로를 사용하는 경우 절차의 해당 단계를 조정합니다.
- 해당 개인 키는 /etc/pki/tls/private/httpd.key 파일에 저장됩니다. 다른 경로를 사용하는 경우 절차의 해당 단계를 조정합니다.
- webserver-ca CA 인증서는 /etc/pki/tls/private/sub-ca.crt 파일에 저장됩니다. 다른 경로를 사용하는 경우 절차의 해당 단계를 조정합니다.
- 클라이언트와 my_company.idm.example.com 웹 서버는 서버의 호스트 이름을 웹 서버의 IP 주소로 확인합니다.

절차

1. mod_ssl 패키지를 설치합니다.

```
# yum install mod_ssl
```

2. /etc/httpd/conf.d/ssl.conf 파일을 편집하고 다음 설정을 <VirtualHost _default_:443> 지시문에 추가합니다.

   a. 서버 이름을 설정합니다.

```
ServerName my_company.idm.example.com
```

중요

서버 이름은 인증서의 Common Name 필드에 설정된 항목과 일치해야 합니다.

b.

선택 사항: 인증서에 **SAN(Subject Alt Names)** 필드에 추가 호스트 이름이 포함된 경우 이러한 호스트 이름에도 **TLS** 암호화를 제공하도록 **mod_ssl**을 구성할 수 있습니다. 이를 구성하려면 해당 이름으로 **ServerAliases** 매개변수를 추가합니다.

```
ServerAlias www.my_company.idm.example.com
server.my_company.idm.example.com
```

c.

개인 키, 서버 인증서 및 **CA** 인증서에 대한 경로를 설정합니다.

```
SSLCertificateKeyFile "/etc/pki/tls/private/httpd.key"
SSLCertificateFile "/etc/pki/tls/certs/httpd.pem"
SSLCACertificateFile "/etc/pki/tls/certs/ca.crt"
```

3.

보안상의 이유로 **root** 사용자만 개인 키 파일에 액세스할 수 있도록 구성합니다.

```
# chown root:root /etc/pki/tls/private/httpd.key
# chmod 600 //etc/pki/tls/private/httpd.key
```

주의

권한이 없는 사용자가 개인 키에 액세스한 경우 인증서를 취소하고 새 개인 키를 만들고 새 인증서를 요청합니다. 그렇지 않으면 **TLS** 연결이 더 이상 안전하지 않습니다.

4.

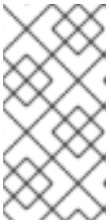
로컬 방화벽에서 **443** 포트를 엽니다.

```
# firewall-cmd --permanent --add-port=443/tcp
# firewall-cmd --reload
```

5.

httpd 서비스를 다시 시작합니다.

```
# systemctl restart httpd
```



참고

개인 키 파일을 암호로 보호한 경우 **httpd** 서비스가 시작될 때마다 이 암호를 입력해야 합니다.

- 브라우저를 사용하고 **https://my_company.idm.example.com** 에 연결합니다.

추가 리소스

- **Apache** 설명서의 **SSL/TLS** 암호화 설명서를 참조하십시오.
- **Apache HTTP Server** 수동 설치를 참조하십시오.
- **RHEL 8**에서 **TLS**의 보안 고려 사항

17.8. APACHE HTTP SERVER에서 지원되는 TLS 프로토콜 버전 설정

기본적으로 **RHEL**의 **Apache HTTP Server**는 최신 브라우저와 호환되는 안전한 기본값을 정의하는 시스템 전체 암호화 정책을 사용합니다. 예를 들어 **DEFAULT** 정책은 **apache**에서 **TLSv1.2** 및 **TLSv1.3** 프로토콜 버전만 사용하도록 정의합니다.

이 섹션에서는 **my_company.idm.example.com Apache HTTP Server**에서 지원하는 **TLS** 프로토콜 버전을 수동으로 구성하는 방법을 설명합니다. 환경에서 특정 **TLS** 프로토콜 버전만 활성화해야 하는 경우 절차를 따르십시오. 예를 들면 다음과 같습니다.

- 환경에 해당 클라이언트가 클라이언트가 취약한 **TLS1 (TLSv1.0)** 또는 **TLS1.1** 프로토콜을 사용할 수도 있어야 하는 경우
- **Apache**가 **TLSv1.2** 또는 **TLSv1.3** 프로토콜만 지원하도록 구성하려는 경우

사전 요구 사항

- **Apache HTTP** 서버에 **TLS** 암호화 추가에 설명된 대로 **my_company.idm.example.com** 서버에서 **TLS** 암호화가 활성화되어 있습니다.

절차

1. /etc/httpd/conf/httpd.conf 파일을 편집하고 다음 설정을 TLS 프로토콜 버전을 설정하려는 <VirtualHost> 지시문에 추가합니다. 예를 들어 TLSv1.3 프로토콜만 활성화하려면 다음을 수행합니다.

```
SSLProtocol -All TLSv1.3
```

2. httpd 서비스를 다시 시작합니다.

```
# systemctl restart httpd
```

검증 단계

1. 다음 명령을 사용하여 서버가 TLSv1.3을 지원하는지 확인합니다:

```
# openssl s_client -connect example.com:443 -tls1_3
```

2. 다음 명령을 사용하여 서버가 TLSv1.2를 지원하지 않는지 확인합니다.

```
# openssl s_client -connect example.com:443 -tls1_2
```

서버가 프로토콜을 지원하지 않으면 명령에서 오류를 반환합니다.

```
140111600609088:error:1409442E:SSL routines:ssl3_read_bytes:tlsv1 alert protocol
version:ssl/record/rec_layer_s3.c:1543:SSL alert number 70
```

3. 선택 사항: 다른 TLS 프로토콜 버전에 대해 명령을 반복합니다.

추가 리소스

- update-crypto-policies(8) 매뉴얼 페이지를 참조하십시오.
- 시스템 전체 암호화 정책 사용을 참조하십시오.
- SSLProtocol 매개변수에 대한 자세한 내용은 Apache 설명서의 mod_ssl 설명서를 참조하십시오.

- Apache HTTP Server 수동 설치를 참조하십시오.

17.9. APACHE HTTP SERVER에서 지원되는 암호 설정

기본적으로 Apache HTTP 서버는 최근 브라우저와 호환되는 안전한 기본값을 정의하는 시스템 전체 암호화 정책을 사용합니다. 시스템 전체에서 허용하는 암호 목록은 /etc/crypto-policies/back-ends/openssl.config 파일을 참조하십시오.

이 섹션에서는 my_company.idm.example.com Apache HTTP 서버가 지원하는 암호를 수동으로 구성하는 방법을 설명합니다. 환경에 특정 암호가 필요한 경우 절차를 따르십시오.

사전 요구 사항

- Apache HTTP 서버에 TLS 암호화 추가에 설명된 대로 my_company.idm.example.com 서버에서 TLS 암호화가 활성화되어 있습니다.

절차

1. /etc/httpd/conf/httpd.conf 파일을 편집하고 SSLCipherSuite 매개변수를 TLS 암호를 설정하려는 <VirtualHost> 지시문에 추가합니다.

```
SSLCipherSuite
"EECDH+AESGCM:EDH+AESGCM:AES256+EECDH:AES256+EDH:!SHA1:!SHA256"
```

이 예제에서는 EECDH+AESGCM,EDH+AESGCM,AES256+EECDH 및 AES256+EDH 암호만 활성화하며 SHA1 및 SHA256 메시지 인증 코드(MAC)를 사용하는 모든 암호를 비활성화합니다.

2. httpd 서비스를 다시 시작합니다.

```
# systemctl restart httpd
```

검증 단계

1. Apache HTTP Server에서 지원하는 암호 목록을 표시하려면 다음을 수행합니다.

a. nmap 패키지를 설치합니다.

```
# yum install nmap
```

b. nmap 유틸리티를 사용하여 지원되는 암호를 표시합니다.

```
# nmap --script ssl-enum-ciphers -p 443 example.com
...
PORT    STATE SERVICE
443/tcp open  https
| ssl-enum-ciphers:
|   TLSv1.2:
|     ciphers:
|       TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384 (ecdh_x25519) - A
|       TLS_DHE_RSA_WITH_AES_256_GCM_SHA384 (dh 2048) - A
|       TLS_ECDHE_RSA_WITH_CHACHA20_POLY1305_SHA256 (ecdh_x25519) - A
...
```

추가 리소스

- update-crypto-policies(8) 매뉴얼 페이지를 참조하십시오.
- 시스템 전체 암호화 정책 사용을 참조하십시오.
- SSLCipherSuite 매개변수에 대한 자세한 내용은 Apache 설명서의 mod_ssl 설명서를 참조하십시오.
- Apache HTTP Server 수동 설치를 참조하십시오.

17.10. TLS 클라이언트 인증서 인증 구성

클라이언트 인증서 인증을 통해 관리자는 인증서를 사용하여 인증하는 사용자만 my_company.idm.example.com 웹 서버의 리소스에 액세스할 수 있도록 허용할 수 있습니다. 이 섹션에서는 /var/www/html/Example/ 디렉터리에 대한 클라이언트 인증서 인증을 구성하는 방법을 설명합니다.

중요

my_company.idm.example.com Apache 서버에서 TLS 1.3 프로토콜을 사용하는 경우 특정 클라이언트에 추가 구성이 필요합니다. 예를 들어 Firefox에서 about:config 메뉴의 security.tls.enable_post_handshake_auth 매개 변수를 true로 설정합니다. 자세한 내용은 Red Hat Enterprise Linux 8의 전송 계층 보안 버전 1.3을 참조하십시오.

사전 요구 사항

- Apache HTTP 서버에 TLS 암호화 추가에 설명된 대로 my_company.idm.example.com 서버에서 TLS 암호화가 활성화되어 있습니다.

절차

1. /etc/httpd/conf/httpd.conf 파일을 편집하고 다음 설정을 클라이언트 인증을 구성하려는 <VirtualHost> 지시문에 추가합니다.

```
<Directory "/var/www/html/Example/">
  SSLVerifyClient require
</Directory>
```

SSLverifyClient require를 설정해야 클라이언트가 /var/www/html/Example/ 디렉터리의 콘텐츠에 액세스하기 전에 서버가 클라이언트 인증서의 유효성을 검사해야 합니다.

2. httpd 서비스를 다시 시작합니다.

```
# systemctl restart httpd
```

검증 단계

1. curl 유틸리티를 사용하여 클라이언트 인증 없이 https://my_company.idm.example.com/Example/ URL에 액세스합니다.

```
$ curl https://my_company.idm.example.com/Example/
curl: (56) OpenSSL SSL_read: error:1409445C:SSL routines:ssl3_read_bytes:tlsv13 **alert
certificate required**, errno 0
```

이 오류는 my_company.idm.example.com 웹 서버에 클라이언트 인증서 인증이 필요함을 나타냅니다.

2.

클라이언트 개인 키 및 인증서와 **CA** 인증서를 **curl**에 전달하여 클라이언트 인증으로 동일한 **URL**에 액세스합니다.

```
$ curl --cacert ca.crt --key client.key --cert client.crt
https://my_company.idm.example.com/Example/
```

요청이 성공하면 **curl**은 **/var/www/html/Example/** 디렉터리에 저장된 **index.html** 파일을 표시합니다.

추가 리소스

- **Apache** 설명서의 **mod_ssl** 구성 **How-To** 설명서를 참조하십시오.
- **Apache HTTP Server** 수동 설치를 참조하십시오.

17.11. 새 사용자 인증서를 요청하고 클라이언트로 내보내기

IdM(Identity Management) 관리자는 **IdM** 클라이언트에서 실행 중인 웹 서버를 구성하여 웹 브라우저를 사용하여 특정 **IdM** 하위 **CA**에서 발급한 인증서로 인증하는 사용자를 요청할 수 있습니다. 특정 **IdM** 하위 **CA**의 사용자 인증서를 요청하고 인증서 및 해당 개인 키를 사용자가 웹 브라우저를 사용하여 웹 서버에 액세스하려는 호스트로 내보내려면 이 섹션을 완료합니다. 인증서와 개인 키를 브라우저로 가져옵니다.

절차

1. 선택적으로 새 디렉터리(예: **~/certdb/**)를 만들고 임시 인증서 데이터베이스로 만듭니다. 메시지가 표시되면 **NSS** 인증서 **DB** 암호를 생성하여 후속 단계에서 생성할 인증서의 키를 암호화하십시오.

```
# mkdir ~/certdb/
# certutil -N -d ~/certdb/
Enter a password which will be used to encrypt your keys.
The password should be at least 8 characters long,
and should contain at least one non-alphabetic character.

Enter new password:
Re-enter password:
```

2. **CSR**(인증서 서명 요청)을 생성하고 출력을 파일로 리디렉션합니다. 예를 들어 **IDM.EXAMPLE.COM** 영역에서 **idm_user** 사용자에 대한 **4096** 비트 인증서에 대한 **certificate_request.csr** 이라는 이름으로 **CSR**을 생성하려면 인증서 개인 키의 **nickname**을

idm_user 로 설정하고 주체를 CN=id_user,OID_user,=OIDM.AMPLE :COM :COM :COM 영역에서 설정할 수 있습니다.

```
# certutil -R -d ~/certdb/ -a -g 4096 -n idm_user -s
"CN=idm_user,O=IDM.EXAMPLE.COM" > certificate_request.csr
```

3.

메시지가 표시되면 certutil 을 사용하여 임시 데이터베이스를 생성할 때 입력한 것과 동일한 암호를 입력합니다. 그런 다음 중지하도록 지시할 때까지 randlomly를 계속 입력합니다.

```
Enter Password or Pin for "NSS Certificate DB":

A random seed must be generated that will be used in the
creation of your key.  One of the easiest ways to create a
random seed is to use the timing of keystrokes on a keyboard.

To begin, type keys on the keyboard until this progress meter
is full.  DO NOT USE THE AUTOREPEAT FUNCTION ON YOUR KEYBOARD!


Continue typing until the progress meter is full:
```

4.

인증서 요청 파일을 서버에 제출합니다. 새로 발행된 인증서, 인증서를 저장할 출력 파일, 인증서 프로필을 선택적으로 연결할 Kerberos 주체를 지정합니다. 인증서를 발행할 IdM 하위 CA 를 지정합니다. 예를 들어, IECUserRoles 프로필의 인증서를 가져오려면 webclient-ca 에서 idm_user@IDM.EXAMPLE.COM 주체에 대한 사용자 역할 확장자가 추가된 프로파일, webclient- user.pem 파일의 인증서를 저장합니다.

```
# ipa cert-request certificate_request.csr --principal=idm_user@IDM.EXAMPLE.COM --
profile-id=IECUserRoles --ca=webclient-ca --certificate-out=~/idm_user.pem
```

5.

NSS 데이터베이스에 인증서를 추가합니다. 인증서가 NSS 데이터베이스의 개인 키와 일치하도록 이전에 CSR을 생성할 때 사용한 것과 동일한 닉네임을 설정하려면 -n 옵션을 사용합니다. -t 옵션은 신뢰 수준을 설정합니다. 자세한 내용은 certutil(1) 매뉴얼 페이지를 참조하십시오. i 옵션은 입력 인증서 파일을 지정합니다. 예를 들어, NSS 데이터베이스에 추가하려면 ~/certdb/ 데이터베이스의 ~/ idm_user.pem 파일에 저장된 idm_user nickname이 있는 인증서를 사용합니다.

```
# certutil -A -d ~/certdb/ -n idm_user -t "P,," -i ~/idm_user.pem
```

6.

NSS 데이터베이스의 키가 별 이름으로 표시되지 않는지 확인합니다. 예를 들어 ~/certdb/ 데이터베이스에 저장된 인증서가 분리되지 않았는지 확인하려면 다음을 수행합니다.

```
# certutil -K -d ~/certdb/
< 0> rsa      5ad14d41463b87a095b1896cf0068ccc467df395   NSS Certificate
DB:idm_user
```

7. pk12util 명령을 사용하여 NSS 데이터베이스에서 PKCS12 형식으로 인증서를 내보냅니다. 예를 들어 /root/certdb NSS 데이터베이스에서 idm_user nickname이 있는 인증서를 ~/idm_user.p12 파일로 내보내려면 다음을 실행합니다.

```
# pk12util -d ~/certdb -o ~/idm_user.p12 -n idm_user
Enter Password or Pin for "NSS Certificate DB":
Enter password for PKCS12 file:
Re-enter password:
pk12util: PKCS12 EXPORT SUCCESSFUL
```

8. idm_user 의 인증서 인증을 활성화할 호스트로 인증서를 전송합니다.

```
# scp ~/idm_user.p12 idm_user@client.idm.example.com:/home/idm_user/
```

9. 인증서가 전송된 호스트에서 .pkcs12 파일이 보안상의 이유로 'other' 그룹에 액세스할 수 없는 디렉터리로 설정합니다.

```
# chmod o-rwx /home/idm_user/
```

10. 보안상의 이유로 서버에서 임시 NSS 데이터베이스와 .pkcs12 파일을 제거하십시오.

```
# rm ~/certdb/
# rm ~/idm_user.p12
```

17.12. 인증서 인증을 사용하도록 브라우저 구성

WebUI를 사용하여 IdM(Identity Management)에 로그인할 때 인증서로 인증하려면 사용자 및 관련 CA(인증 기관) 인증서를 Mozilla Firefox 또는 Google Chrome 브라우저로 가져와야 합니다. 브라우저가 실행 중인 호스트 자체는 IdM 도메인의 일부가 될 필요가 없습니다.

IdM은 WebUI에 연결하기 위해 다음 브라우저를 지원합니다.

- Mozilla Firefox 38 이상
- Google Chrome 46 이상

다음 절차는 **Mozilla Firefox 57.0.1** 브라우저를 구성하는 방법을 보여줍니다.

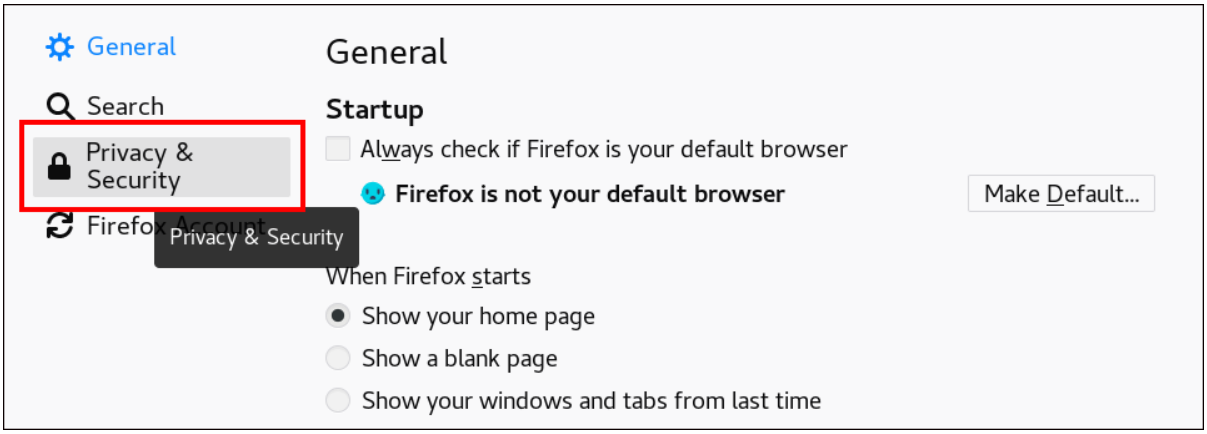
사전 요구 사항

- **PKCS#12** 형식으로 사용 중인 브라우저로 가져오려는 사용자 인증서가 있습니다.
- 하위 **CA** 인증서를 다운로드하여 **PEM** 형식으로 처리했습니다.

절차

1. **Firefox**를 연 다음 기본 설정 → 개인 정보 보호 및 보안 으로 이동합니다.

그림 17.8. 환경 설정의 개인 정보 보호 및 보안 섹션



2. 인증서 보기를 클릭합니다.

그림 17.9. 개인 정보 보호 및 보안에서 인증서 보기

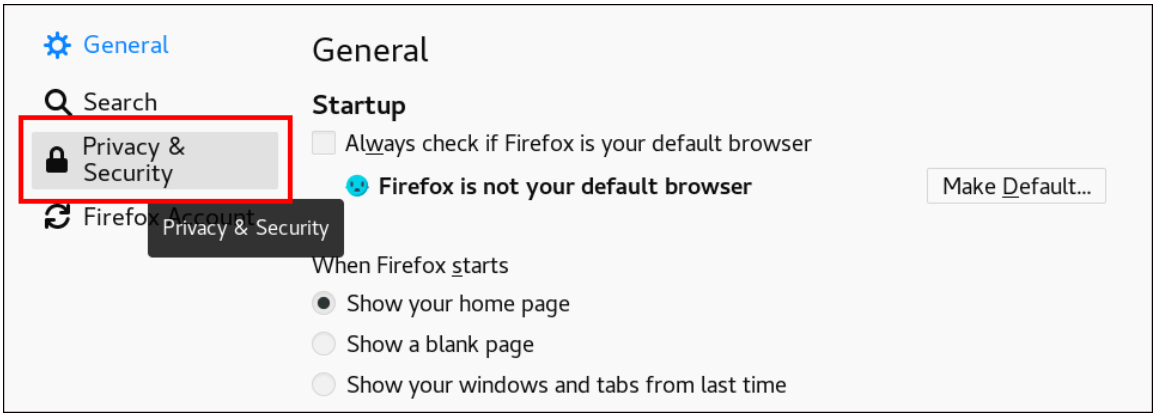
3. 사용자 인증서 탭에서 가져오기 를 클릭합니다. **PKCS12** 형식으로 사용자의 인증서를 찾아 연 다음 **OK** 및 **OK** 를 클릭합니다.

4.

IdM 하위 CA가 Firefox에서 신뢰할 수 있는 기관으로 인식되도록 하려면 신뢰할 수 있는 인증 기관 인증서로 IdM WebUI에서 하위 CA 인증서를 다운로드할 때 저장한 IdM 하위 CA 인증서를 가져옵니다.

a. Firefox를 열고 기본 설정으로 이동하여 개인 정보 보호 및 보안을 클릭합니다.

그림 17.10. 환경 설정의 개인 정보 보호 및 보안 섹션



b. 인증서 보기를 클릭합니다.

그림 17.11. 개인 정보 보호 및 보안에서 인증서 보기

c. Authorities (권한) 탭에서 가져오기 를 클릭합니다. 하위 CA 인증서를 찾아 엽니다. 인증서를 신뢰하여 웹 사이트를 확인한 다음 확인 및 확인 을 클릭합니다.

18장. 특정 관련 인증서 그룹을 빠르게 무효화

시스템 관리자는 관련 인증서의 특정 그룹을 신속하게 무효화할 수 있도록 하려면 다음을 수행하십시오.

- 특정 IdM(Lightweight Identity Management) 하위 CA에서 발행한 인증서만 신뢰하도록 애플리케이션을 설계합니다. 이후에는 이러한 인증서를 발급한 IdM(Identity Management) 하위 CA의 인증서만 취소하여 이러한 모든 인증서를 무효화할 수 있습니다. IdM에서 간단한 하위 서비스를 생성하고 사용하는 방법에 대한 자세한 내용은 관련 인증서의 특정 그룹이나 신속하게 무효화 를 참조하십시오.

- to-be-revoked IdM 하위 계정에서 발급한 모든 인증서가 즉시 무효화되도록 하려면 IdM OCSP 응답자를 사용하도록 이러한 인증서에 의존하는 애플리케이션을 구성합니다. 예를 들어, OCSP 응답자를 사용하도록 Firefox 브라우저를 구성하려면 Query OCSP 응답기 서버가 Firefox preferences에서 현재 유효성 이 있는지 확인하십시오.

  IdM에서 CRL(인증서 해지 목록)은 4시간마다 업데이트됩니다. IdM 하위 CA에서 발행한 모든 인증서를 무효화하려면 IdM 하위 CA 인증서를 취소 하십시오. 또한 관련 CA ACL을 비활성화하고 IdM 하위 시스템을 비활성화하는 것이 좋습니다. 서브-CA를 비활성화하면 하위 CA가 새 인증서를 발행하지 못하지만, 하위 CA의 서명 키가 유지되므로 이전에 발행한 인증서에 대해 OCSP(Online Certificate Status Protocol) 응답을 생성할 수 있습니다.

중요

사용자 환경에서 OCSP를 사용하는 경우 하위 CA를 삭제하지 마십시오. 하위 CA를 삭제하면 하위 CA의 서명 키가 삭제되어 해당 하위 CA에서 발급한 인증서의 OCSP 응답이 발생하지 않습니다.

하위 CA를 삭제하는 유일한 시나리오는 동일한 주체 고유 이름(DN)을 사용하여 새 하위 시스템을 생성하지만 새 서명 키를 사용하여 새 하위 서비스를 생성해야 하는 경우입니다.

18.1. IDM CLI에서 CA ACL 비활성화

IdM 서비스 또는 IdM 서비스 그룹을 삭제하려면 기존 해당 CA ACL을 비활성화하는 것이 좋습니다.

이 섹션을 완료하고, IdM 클라이언트에서 실행 중인 웹 서버를 제한하여 webserver-ca IdM 하위 시스템에서 발급하도록 인증서를 요청하고, IdM 사용자가 웹 클라이언트-ca IdM 하위 시스템에서 발행하도록 사용자 인증서를 제한하도록 TLS_web_client_authentication CA ACL을 비활성화하는 TLS_web_server_authentication CA ACL을 비활성화합니다.

절차

1. 선택적으로 IdM 환경의 모든 CA ACL을 보려면 ipa caacl-find 명령을 입력합니다.

```
$ ipa caacl-find
----------------
3 CA ACLs matched
----------------
  ACL name: hosts_services_calPAserviceCert
  Enabled: TRUE

  ACL name: TLS_web_server_authentication
  Enabled: TRUE

  ACL name: TLS_web_client_authentication
  Enabled: TRUE
```

2. 선택적으로 CA ACL의 세부 정보를 보려면 ipa caacl-show 명령을 입력하고 CA ACL 이름을 지정합니다.

```
$ ipa caacl-show TLS_web_server_authentication
  ACL name: TLS_web_server_authentication
  Description: CAACL for web servers authenticating to web clients using certificates
issued by webserver-ca
  Enabled: TRUE
  CAs: webserver-ca
  Profiles: calPAserviceCert
  Services: HTTP/rhel8server.idm.example.com@IDM.EXAMPLE.COM
```

3. CA ACL을 비활성화하려면 ipa caacl-disable 명령을 입력하고 CA ACL 이름을 지정합니다.

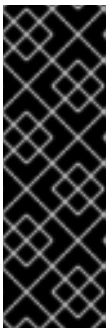
- TLS_web_server_authentication CA ACL을 비활성화하려면 다음을 입력합니다.

```
$ ipa caacl-disable TLS_web_server_authentication
-----------------------------------------------
Disabled CA ACL "TLS_web_server_authentication"
-----------------------------------------------
```

- TLS_web_client_authentication CA ACL을 비활성화하려면 다음을 입력합니다.

```
$ ipa caacl-disable TLS_web_client_authentication
-----------------------------------------------
Disabled CA ACL "TLS_web_client_authentication"
-----------------------------------------------
```

이제 유일하게 활성화된 **CA ACL**은 **hosts_services_caIPAserviceCert CA ACL**입니다.



중요

hosts_services_caIPAserviceCert CA ACL을 비활성화하면 주의해야 합니다. **ca IPAserviceCert** 프로필로 **ipa CA** 사용을 부여하는 다른 **CA ACL**을 제공하지 않고 **host_services_ca IPAserviceCert**를 비활성화하면 **IdM HTTP** 및 **LDAP** 인증서의 인증서 갱신이 실패합니다. **IdM HTTP** 및 **LDAP** 인증서가 만료되면 결국 **IdM** 시스템 오류가 발생합니다.

18.2. IDM 하위 서비스 비활성화

해당 하위 **CA**에서 발급한 모든 인증서를 무효화하기 위해 **IdM** 하위 **CA**의 **CA** 인증서를 취소한 후 더 이상 필요하지 않은 경우 **IdM** 하위 **CA**를 비활성화하는 것이 좋습니다. 나중에 하위 서비스를 다시 활성화할 수 있습니다.

서브-**CA**를 비활성화하면 하위 **CA**가 새 인증서를 발행하지 못하지만, 하위 **CA**의 서명 키가 유지되므로 이전에 발행한 인증서에 대해 **OCSP(Online Certificate Status Protocol)** 응답을 생성할 수 있습니다.

사전 요구 사항

- **IdM** 관리자로 로그인되어 있습니다.

절차

- **ipa ca-disable** 명령을 입력하고 하위 서비스의 이름을 지정합니다.

```
$ ipa ca-disable webserver-CA
--------------------
Disabled CA "webserver-CA"
--------------------
```

19장. IDM 상태 점검을 사용하여 인증서 확인

이 섹션에서는 IdM(Identity Management)의 Healthcheck 툴을 이해하고 사용하여 certmonger가 유지 관리하는 IPA 인증서의 문제를 식별하는 데 도움이 됩니다.

자세한 내용은 IdM의 상태 점검을 참조하십시오.

사전 요구 사항

- Healthcheck 툴은 RHEL 8.1 이상에서만 사용할 수 있습니다.

19.1. IDM 인증서 상태 점검 테스트

Healthcheck 툴에는 IdM(Identity Management)에서 certmonger가 유지 관리하는 인증서 상태를 확인하기 위한 여러 테스트가 포함되어 있습니다. certmonger에 대한 자세한 내용은 certmonger 를 사용하여 서비스에 대한 IdM 인증서 받기를 참조하십시오.

이 테스트 제품군은 만료, 검증, 신뢰 및 기타 문제를 확인합니다. 동일한 기본 문제에 대해 여러 오류가 발생할 수 있습니다.

모든 인증서 테스트를 보려면 --list-sources 옵션을 사용하여 ipa-healthcheck 를 실행합니다.

```
# ipa-healthcheck --list-sources
```

ipahealthcheck.ipa.certs 소스에서 모든 테스트를 찾을 수 있습니다.

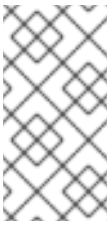
IPACertmongerExpirationCheck

이 테스트는 certmonger 의 만료를 확인합니다.

오류가 보고되면 인증서가 만료됩니다.

경고가 표시되면 인증서가 곧 만료됩니다. 기본적으로 이 테스트는 인증서 만료일로부터 28일 이내에 적용됩니다.

/etc/ipahealthcheck/ipahealthcheck.conf 파일에서 일 수를 구성할 수 있습니다. 파일을 연 후 default 섹션에 있는 cert_expiration_days 옵션을 변경합니다.



참고

certmonger는 인증서 만료를 로드하고 유지 관리합니다. 이 검사에서는 디스크상의 인증서의 유효성을 검사하지 않습니다.

IPACertfileExpirationCheck

이 테스트에서는 인증서 파일 또는 NSS 데이터베이스를 열 수 없는지 확인합니다. 이 테스트도 만료되었는지 확인합니다. 따라서 오류 또는 경고 출력에서 msg 특성을 주의 깊게 읽으십시오. 메시지는 문제를 지정합니다.

참고

이 테스트에서는 디스크상의 인증서를 확인합니다. 인증서가 누락된 경우 읽을 수 없으므로 별도의 오류도 발생할 수 있습니다.

IPACertNSSTrust

이 테스트는 NSS 데이터베이스에 저장된 인증서의 신뢰를 비교합니다. NSS 데이터베이스에서 예상되는 추적 인증서의 경우 신뢰는 예상 값과 비교되고 일치하지 않는 경우 오류가 발생합니다.

IPANSSChainValidation

이 테스트는 NSS 인증서의 인증서 체인을 검증합니다. 테스트 실행: certutil -V -u V -e -d [dbdir] -n [nickname]

IPAOpenSSLChainValidation

이 테스트에서는 OpenSSL 인증서의 인증서 체인을 검증합니다. NSSChain 검증과 비교하여 볼 수 있는 OpenSSL 명령은 다음과 같습니다.

```
openssl verify -verbose -show_chain -CAfile /etc/ipa/ca.crt [cert file]
```

IPARAAgent

이 테스트는 디스크의 인증서를 uid=ipara,ou=People,o=ipaca 의 LDAP의 동등한 레코드와 비교합니다.

IPACertRevocation

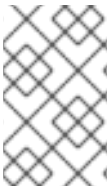
이 테스트에서는 **certmonger**를 사용하여 인증서가 취소되지 않았는지 확인합니다. 따라서 이 테스트는 **certmonger**가 관리하는 인증서와 관련된 문제를 찾을 수 있습니다.

IPACertmongerCA

이 테스트에서는 **certmonger CA**(인증 기관) 구성을 확인합니다. **IdM**은 **CA** 없이 인증서를 발행할 수 없습니다.

certmonger는 일련의 **CA** 도우미를 유지 관리합니다. **IdM**에는 호스트 또는 서비스 인증서에 대한 호스트 또는 사용자 주체로 인증하는 **IdM**을 통해 인증서를 발급하는 **IPA**라는 **CA**가 있습니다.

또한 **CA** 하위 시스템 인증서를 갱신하는 개**tag-ipa-ca-renew-agent** 및 **dogtag-ipa-ca-renew-agent-reuse** 도 있습니다.



참고

문제를 확인하려고 할 때 모든 **IdM** 서버에서 이러한 테스트를 실행합니다.

19.2. HEALTHCHECK 도구를 사용하여 인증서 검사

이 섹션에서는 **Healthcheck** 도구를 사용하여 **IdM(Identity Management)** 인증서 상태 점검의 독립 실행형 수동 테스트에 대해 설명합니다.

상태 점검 툴에는 많은 테스트가 포함되어 있으므로 다음을 사용하여 결과를 단축할 수 있습니다.

- 모든 성공적인 테스트 제외: **--failures-only**
- 인증서 테스트 포함: **--source=ipahealthcheck.ipa.certs**

사전 요구 사항

- 상태 점검 테스트는 **root** 사용자로 수행해야 합니다.

절차

- 인증서와 관련된 경고, 오류 및 심각한 문제를 사용하여 상태 점검을 실행하려면 다음을 입력

합니다.

```
# ipa-healthcheck --source=ipahealthcheck.ipa.certs --failures-only
```

테스트에 성공하면 빈 괄호가 표시됩니다.

```
[]
```

실패한 테스트에서는 다음 출력을 보여줍니다.

```
{
  "source": "ipahealthcheck.ipa.certs",
  "check": "IPACertfileExpirationCheck",
  "result": "ERROR",
  "kw": {
    "key": 1234,
    "dbdir": "/path/to/nssdb",
    "error": [error],
    "msg": "Unable to open NSS database '/path/to/nssdb': [error]"
  }
}
```

이 IPACertfileExpirationCheck 테스트는 NSS 데이터베이스를 여는 데 실패했습니다.

추가 리소스

- man ipa-healthcheck 을 참조하십시오.

20장. IDM 상태 점검을 사용하여 시스템 인증서 확인

이 섹션에서는 시스템 인증서 문제를 식별하는 IdM(Identity Management)의 Healthcheck 툴에 대해 설명합니다.

자세한 내용은 IdM의 Healthcheck를 참조하십시오. .Prerequisites

- Healthcheck 툴은 RHEL 8.1 이상에서만 사용할 수 있습니다.

20.1. 시스템 인증서 상태 점검 테스트

Healthcheck 툴에는 시스템(DogTag) 인증서를 확인하기 위한 여러 테스트가 포함되어 있습니다.

모든 테스트를 보려면 --list-sources 옵션을 사용하여 ipa-healthcheck 를 실행합니다.

```
# ipa-healthcheck --list-sources
```

ipahealthcheck.dogtag.ca 소스에서 모든 테스트를 찾을 수 있습니다.

DogtagCertsConfigCheck

이 테스트는 NSS 데이터베이스의 CA(Certificate Authority) 인증서를 CS.cfg 에 저장된 동일한 값과 비교합니다. 일치하지 않는 경우 CA가 시작되지 않습니다.

특히, 다음을 확인합니다.

- auditSigningCert cert-pki-ca against ca.audit_signing.cert
- ocspSigningCert cert-pki-ca against ca.ocsp_signing.cert
- caSigningCert cert-pki-ca against ca.signing.cert
- subsystemCert cert-pki-ca against ca.subsystem.cert

- ca.sslserver.cert에 대한 server-Cert cert-pki-ca

키 복구 기관 (KRA)이 설치된 경우:

- transportCert cert-pki-kra against ca.connector.KRA.transportCert

DogtagCertsConnectivityCheck

이 테스트는 연결을 확인합니다. 이 테스트는 다음을 확인하는 ipa cert-show 1 명령과 동일합니다.

- Apache의 PKI 프록시 구성
- IdM에서 CA를 찾을 수 있음
- RA 에이전트 클라이언트 인증서
- 요청에 대한 CA 응답의 수정

인증서는 cert-show 를 실행하고 CA에서 예상되는 결과(인증서 또는 찾을 수 없음)를 반환하는지 확인하려고 하므로 직렬 #1이 있는 인증서를 확인합니다.

참고

문제를 찾고자 할 때 모든 IdM 서버에서 이러한 테스트를 실행합니다.

20.2. HEALTHCHECK를 사용하여 시스템 인증서 검사

이 섹션에서는 Healthcheck 도구를 사용하여 IdM(Identity Management) 인증서에 대한 독립형 수동 테스트에 대해 설명합니다.

Healthcheck 툴에는 많은 테스트가 포함되어 있으므로 DogTag 테스트만 추가하여 결과를 좁힐 수 있

습니다. **--source=ipahealthcheck.dogtag.ca**

절차

- **DogTag** 인증서로 제한적으로 **Healthcheck**를 실행하려면 다음을 입력합니다.

```
# ipa-healthcheck --source=ipahealthcheck.dogtag.ca
```

성공적인 테스트의 예는 다음과 같습니다.

```
{
  "source: ipahealthcheck.dogtag.ca",
  "check: DogtagCertsConfigCheck",
  "result: SUCCESS",
  "uuid: 9b366200-9ec8-4bd9-bb5e-9a280c803a9c",
  "when: 20191008135826Z",
  "duration: 0.252280",
  "kw:" {
    "key": "Server-Cert cert-pki-ca",
    "configfile":  "/var/lib/pki/pki-tomcat/conf/ca/CS.cfg"
    }
}
```

실패한 테스트의 예는 다음과 같습니다.

```
{
  "source: ipahealthcheck.dogtag.ca",
  "check: DogtagCertsConfigCheck",
  "result: CRITICAL",
  "uuid: 59d66200-1447-4b3b-be01-89810c803a98",
  "when: 20191008135912Z",
  "duration: 0.002022",
  "kw:" {
    "exception": "NSDB /etc/pki/pki-tomcat/alias not initialized",
    }
}
```

추가 리소스

- **man ipa-healthcheck** 을 참조하십시오.