CA SiteMinder Federation Standalone

Federation Standalone 안내서

r12.52 SP1

CA Technologies 제품 참조

이 문서는 다음 CA Technologies 제품을 참조합니다 :

- SiteMinder

CA 에 문의

기술 지원팀에 문의

온라인 기술 지원 및 지사 목록, 기본 서비스 시간, 전화 번호에 대해서는 http://www.ca.com/worldwide 에서 기술 지원팀에 문의하십시오.

설명서 변경 사항

SiteMinder 의 이전 릴리스에서 발견된 문제점으로 인해 다음과 같은 내용이 12.52 설명서에서 업데이트되었습니다.

- 서명 확인 오류 해결 (페이지 463) - 서명 확인 실패 시 XML 서명 확인 공격을 무산시키는 방법에 대한 정보가 추가되었습니다. CQ: 168095(STAR 문제: 21321479-1)를 해결합니다.
- 페더레이션 작업을 모니터링하기 위한 로그 (페이지 429) - server.log 파일에서의 log4j 사용을 반영하여 로깅 관련 정보가 업데이트되었습니다. 또한 모든 로깅 정보가 이 섹션에서 통합되었습니다. CQ 171956(STAR 문제 21454409-1) 및 CQ 165412(Star 문제: 21257428-1)를 해결합니다.
- 기존 파일에서 키/인증서 쌍 가져오기 (페이지 144) - "CA 로 사용" 옵션의 동작을 설명하는 내용이 추가되었습니다. CQ 173083 을 해결합니다.
- FEDSESSION 쿠키 시간 만료 설정 (페이지 217) - 유휴 시간 만료 및 최대 시간 만료 설정이 수정되었습니다. CQ 173107 을 해결합니다.
- 기존 구성 백업 (페이지 456) - XPSExport 명령 옵션 **-xa** 가 더 이상 사용되지 않습니다. 이 명령은 **-xe -xp** 명령으로 대체되었습니다. CQ 173659 및 STAR 문제 21480783-2 를 해결합니다.
- 위임된 인증을 위한 쿼리 문자열 방법 (페이지 269) - 프로덕션 환경에서만 쿼리 문자열 방법을 사용하도록 지시하는 참고가 추가되었습니다. CQ 165470 및 CQ 165473 을 해결합니다.
- SSL 을 통해 LDAP 사용자 디렉터리에 연결하는 방법 (페이지 90) - cert7.db 파일에 대한 참조가 cert8.db 파일로 업데이트되었습니다. CQ 172315(STAR 문제: 21454358-01)를 해결합니다.
- 부하 분산 (페이지 389) 및 장애 조치 지원 (페이지 375) - 그래픽이 업데이트되었고 이러한 기능에 대한 프로세스와 설명이 분명해졌습니다. CQ 145146(STAR 문제: 20533073-1)을 해결합니다.

목차

제 1 장: 소개

이 섹션은 다음 항목을 포함하고 있습니다.

제품 및 구성 개요

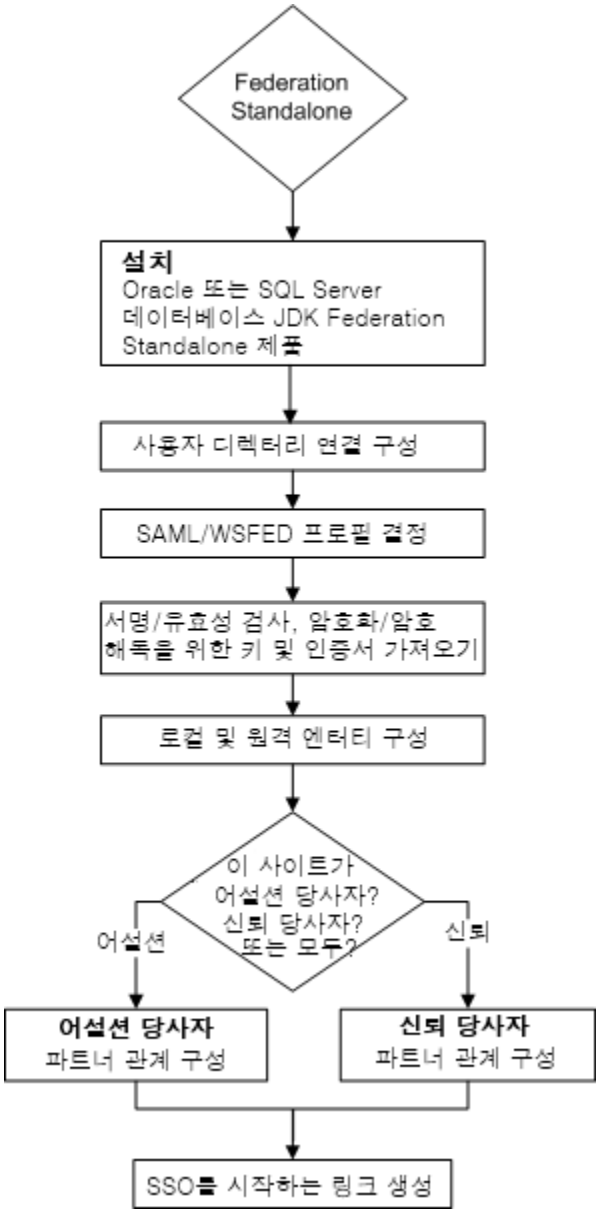
엔터프라이즈 응용 프로그램 및 서비스는 보안뿐만 아니라 도메인 간 원활한 서비스 액세스도 중요하게 되었습니다. CA SiteMinder?Federation Standalone 을 사용하면 아이덴티티 정보의 융통성과 이동성이 높아지므로 신뢰할 수 있는 비즈니스 파트너의 네트워크 전반에서 안전한 싱글 사인온 및 싱글 로그아웃이 가능합니다. 독립 실행형인 이 제품은 대상 시스템에 페더레이션 소프트웨어가 필요 없습니다.

CA SiteMinder?Federation Standalone 은 다음과 같은 기능을 지원합니다.

- SAML 1.1
- SAML 2.0
- WS-protocol 지원
- FIPS 140-2 호환 암호화 지원

- 이 제품은 독립 실행형 엔터티로 배포하거나, 페더레이션 및 액세스 제어를 위해 SiteMinder Web Access Manager 와 함께 배포할 수 있습니다.
- 사이트는 어설션 당사자이자 신뢰 당사자로 기능할 수 있습니다.
- 암호화된 쿠키 또는 헤더로 대상 응용 프로그램에 아이덴티티 데이터 전달

다음 흐름 다이어그램은 제품에서 페더레이션을 구성하는 일반 프로세스를 보여 줍니다.



제품 구성 요소

CA SiteMinder?Federation Standalone 에는 다음과 같은 구성 요소가 포함되어 있습니다.

- 안전한 프록시 엔진

  백엔드 서버에 트래픽을 전달합니다. 이 엔진은 웹 서버, 서블릿 엔진, 프록시 서버 및 페더레이션 웹 서비스 기능을 사용합니다.

  보안 프록시 엔진의 구성 요소는 다음과 같습니다.

  - Apache 웹 서버

    들어오는 요청에 대한 HTTP 트래픽을 처리하는 HTTP 수신기 역할을 하며 적절히 구성될 경우 HTTPS 트래픽을 처리할 수 있습니다.

  - Tomcat 서버

    Administrative UI 작동에 필요한 서블릿 컨테이너를 제공합니다. Apache 웹 서버는 mod_jk 라는 Tomcat 커넥터를 통해 Tomcat 서버와 통신합니다.

- 페더레이션 서버

  사용자 디렉터리 연결, 인증 기능 및 세션 저장소 기능을 사용하도록 설정합니다.

- 확장 가능한 정책 저장소

  모든 CA SiteMinder® Federation Standalone 데이터 개체를 저장합니다.

- 웹 기반 사용자 인터페이스

  페더레이션 엔터티와 파트너 관계, 개인 키와 인증서 및 다양한 서버 설정의 구성을 관리합니다.

CA SiteMinder® Federation Standalone 에서 제공하는 FIPS 140-2 지원 기능

CA SiteMinder?Federation Standalone 은 인증된 FIPS(Federal Information Processing Standard) 140–2 호환 암호화 라이브러리를 사용합니다. 이러한 라이브러리는 환경에서 중요한 데이터를 암호화하는 데 FIPS 호환 AES(Advanced Encryption Standard) 알고리즘만 사용하는 경우에 FIPS 작동 모드를 제공합니다.

다음 FIPS 작동 모드 중 하나에서 제품을 설치할 수 있습니다.

FIPS_COMPAT

FIPS_COMPAT(호환성) 모드는 설치 중에 사용되는 기본 FIPS 모드입니다. FIPS_COMPAT 모드에서 시스템은 현재의 비 FIPS 알고리즘뿐 아니라 지원되는 FIPS 호환 알고리즘도 함께 지원합니다.

FIPS_COMPAT 모드는 제품의 이전 버전과도 호환됩니다. 이 호환성은 12.0 SP1 이전 버전을 사용하는 환경이 최신 버전과 상호 운용될 수 있도록 해 줍니다. IPS_COMPAT 모드는 현재 구현되어 있는 제품의 보안 수준에 만족하는 클라이언트에게도 적합합니다.

조직에서 FIPS 를 사용할 필요가 없는 경우에는 FIPS_COMPAT 모드로 제품을 설치하십시오. 추가 구성이 필요하지 않습니다.

FIPS_ONLY

FIPS_ONLY 모드의 환경에서는 FIPS 호환 알고리즘만 사용하여 중요한 데이터가 암호화됩니다.

FIPS 호환 알고리즘만 사용할 새로운 설치 환경에서는 제품을 FIPS_ONLY 모드로 설치하십시오.

이 안내서의 부록 (페이지 473)에는 서로 다른 FIPS 모드로 작동할 때 시스템에서 사용하는 특정 암호화 및 암호 해독 알고리즘이 나열되어 있습니다.

중요! FIPS_ONLY 모드로 작동하는 r12.52 SP1 설치는 제품에 노출된 이전 버전의 모든 API 를 포함하여 이전 버전의 제품과 상호 작용하거나 이전 버전의 제품과 호환되지 않습니다. 전체 FIPS_ONLY 모드에 필요한 지원을 받으려면 이러한 모든 소프트웨어를 해당하는 SDK 의 r12.52 SP1 버전과 다시 연결하십시오.

프로그래머리스 페더레이션

프로그래머리스 페더레이션은 보안 인증, 사용자 명확성, 조사 및 SAML 어설션의 수정을 가능하게 하는 HTTP 기반 접근법입니다. 프로그래머리스 페더레이션의 장점은 응용 프로그램이 언어별 SDK 또는 다른 바인딩을 사용할 필요 없이 이러한 태스크를 수행할 수 있다는 점입니다.

프로그래머리스 페더레이션은 HTTP/HTTPS 요청 및 응답을 사용합니다. 이러한 요청 및 응답에는 REST(Representational State Transfer) 시스템 아키텍처를 구현한 웹 서비스를 사용하는 URL 및 HTML 기반 프로토콜을 통해 액세스할 수 있습니다.

모든 응용 프로그램이 HTTP 요청을 보내고, HTTP 응답을 읽고, XML 을 구문 분석하여 CA SiteMinder?Federation Standalone 프로그래머리스 기능의 장점을 활용할 수 있습니다.

프로그래머리스 페더레이션의 핵심 부분은 데이터 교환에 보안을 적용하는 기능입니다. CA SiteMinder?Federation Standalone 은 데이터 보안을 위해 개방 형식 쿠키를 사용합니다. 개방 형식 쿠키는 강력한 암호화 알고리즘을 지원하는 잘 정의된 쿠키 형식입니다. 암호화된 쿠키는 CA SiteMinder?Federation Standalone 과 로컬 또는 원격 응용 프로그램 사이의 요청에 대한 응답을 보호하며, Perl 또는 Ruby 와 같이 개방 형식 쿠키가 사용하는 것과 동일한 암호화 및 암호 해독 알고리즘을 지원하는 모든 프로그래밍 언어로 작성될 수 있습니다.

CA SiteMinder?Federation Standalone SDK 도 개방 형식 쿠키를 지원하여 여러 응용 프로그램을 혼합하여 사용할 수 있습니다.

다음의 CA SiteMinder?Federation Standalone 기능은 프로그래머리스 페더레이션 모델을 구현합니다.

위임된 인증

위임된 인증을 통해 CA SiteMinder® Federation Standalone 은 타사 WAM(웹 액세스 관리) 시스템을 사용하여 보호된 페더레이션 리소스를 요청하는 모든 사용자의 인증을 수행할 수 있습니다. 타사 WAM 은 인증을 수행한 다음 페더레이션된 사용자 아이덴티티를 CA SiteMinder® Federation Standalone 에 전송합니다.

위임된 인증을 위한 통신은 HTTP/HTTPS 요청과 응답을 통해 처리됩니다.

신뢰 당사자의 프로비저닝

프로비저닝은 데이터 및 응용 프로그램에 액세스하기 위해 필요한 계정 권한 및 액세스 권한을 가진 클라이언트 계정을 생성하는 프로세스입니다. CA SiteMinder® Federation Standalone 프로비저닝은 사용자를 위한 새 계정을 설정하거나 SAML 어설션에서 전송된 정보를 기존 사용자 계정에 입력할 수 있습니다.

원격 프로비저닝은 프로비저닝 방법 중 하나입니다. 원격 프로비저닝은 독립적 프로비저닝 응용 프로그램을 사용하여 사용자 레코드를 설정합니다. CA SiteMinder® Federation Standalone 은 어설션 데이터를 전달하기 위해 데이터가 포함된 암호화된 쿠키를 생성합니다. 이 쿠키는 사용자 계정 생성을 담당하는 원격 프로비저닝 응용 프로그램으로 전송됩니다.

프로비저닝을 위한 통신은 HTTP/HTTPS 요청과 응답을 통해 처리됩니다.

대상 사용자

이 안내서에서는 독자가 다음 개념을 이해하고 있는 것으로 가정합니다.

- 기본 SAML 기초 사항
- SAML 바인딩 POST 및 아티팩트
- SSO(싱글 사인온), SLO(싱글 로그아웃) 및 ECP(향상된 클라이언트 또는 프록시)와 같은 SAML 프로필
- PKI(공개 키 인프라) 기초 사항
- SSL(Secure Socket Layer) 통신 기본

이 안내서에서 사용하는 용어

이 안내서에서는 다음과 같은 용어를 사용합니다.

어설션 당사자

신뢰 당사자가 사용할 어설션을 생성하는 SAML 기관입니다. 어설션 당사자는 사용자에 대한 아이덴티티 정보를 생성, 유지 및 관리하고 다른 신뢰 당사자에게 사용자 인증을 제공합니다. SAML 1.1 의 경우 어설션 당사자를 "생산자"라고 합니다. SAML 2.0 및 WS-페더레이션의 경우 어설션 당사자를 "아이덴티티 공급자"라고 합니다.

어설션 소비자 서비스(SAML 1.1 및 2.0)

포함된 SAML 응답이 있는 HTTP 양식이나 SAML 아티팩트를 받고 해당 SAML 어설션을 확보하는 서비스 공급자 구성 요소입니다. 어설션 소비자 서비스는 페더레이션 세션 쿠키를 발행하며 사용자가 SiteMinder 와 통합되어 있는 경우에는 SiteMinder 세션 쿠키를 발행합니다.

어설션 검색 서비스(SAML 1.1)

HTTP 아티팩트 바인딩을 사용하여 SAML 1.1 인증을 처리하는 생산자 측 서비스입니다. 이 서비스는 생산자에 저장되어 있는 어설션을 가져옵니다.

아티팩트 레졸루션 서비스(SAML 2.0)

HTTP 아티팩트 바인딩을 사용하여 SAML 2.0 인증을 수행하는 아이덴티티 공급자 측 서비스입니다. 이 서비스는 아이덴티티 공급자에 저장되어 있는 어설션을 가져옵니다.

AuthnRequest 서비스(SAML 2.0)

서비스 공급자가 도메인 간 싱글 사인온에 대한 AuthnRequest 메시지를 생성할 수 있도록 하는 서비스입니다. 이 메시지에는 CA SiteMinder® Federation Standalone 이 브라우저를 아이덴티티 공급자의 싱글 사인온 서비스로 리디렉션할 수 있도록 해 주는 정보가 들어 있습니다. AuthnRequest 서비스는 POST 및 아티팩트 바인딩을 사용한 싱글 사인온에 사용됩니다.

참고: 이 서비스에서 발행하는 AuthnRequest 메시지의 형식은 OASIS SAML(Security Assertion Markup Language) V2.0 의 프로필에 지정됩니다.

위임된 인증

타사 웹 액세스 관리 시스템을 사용하여 사용자를 인증한 후 다시 CA SiteMinder® Federation Standalone 으로 리디렉션하여 페더레이션 프로세스를 진행하는 기능입니다.

레거시 쿠키

(기존의 FEDPROFILE 쿠키) 사용자 아이덴티티 정보가 포함된 쿠키입니다. 이 쿠키는 FIPS 와 호환되지 않는 PBE 암호화 알고리즘만 지원합니다.

어설션 당사자 측에서는 Java SDK 가 레거시 쿠키를 생성하고 CA SiteMinder® Federation Standalone 이 이 쿠키를 읽습니다. 신뢰 당사자 측에서는 CA SiteMinder® Federation Standalone 이 Java 기반 최종 사용자 응용 프로그램에 사용하도록 레거시 쿠키를 생성하고, 응용 프로그램이 Java SDK 를 사용하여 쿠키를 읽습니다.

개방 형식 쿠키

사용자 아이덴티티 정보가 포함된 쿠키입니다. 개방 형식 쿠키는 쿠키를 생성하는 방식에 따라 FIPS 또는 비 FIPS 호환 알고리즘을 사용하여 암호화할 수 있습니다. CA SiteMinder® Federation Standalone SDK 를 사용하여 개방 형식 쿠키를 생성하거나 UTF-8 인코딩을 지원하는 프로그래밍 언어를 사용하여 수동으로 생성할 수 있습니다.

FIPS 암호화 개방 형식 쿠키가 필요한 경우 쿠키를 생성하고 읽으려면 CA SiteMinder® Federation Standalone SDK 를 사용하십시오. CA SiteMinder® Federation Standalone Java SDK 는 FIPS 호환(AES) 알고리즘 또는 비 FIPS(PBE) 알고리즘을 사용하여 쿠키를 암호화할 수 있습니다. CA SiteMinder® Federation Standalone .NET SDK 는 FIPS 호환 알고리즘만 사용하여 쿠키를 암호화할 수 있습니다.

신뢰 당사자

SAML 기관의 정보를 사용하여 서비스에 대한 액세스를 제공하는 SAML 엔터티입니다. 신뢰 당사자는 어설션 당사자로부터 받은 어설션을 사용하여 사용자를 인증합니다. SAML 1.1 의 경우 신뢰 당사자를 "소비자"라고 합니다. SAML 2.0 의 경우 신뢰 당사자를 "서비스 공급자"라고 합니다.

중요! 이 안내서에서 *신뢰 당사자*라는 용어는 소비자 또는 서비스 공급자를 의미합니다.

싱글 로그아웃 서비스(SAML 2.0)

사용자는 이 서비스를 통해 싱글 로그아웃 이벤트와 동시에 페더레이션의 모든 응용 프로그램에서 로그아웃할 수 있습니다. 아이덴티티 공급자 또는 서비스 공급자는 싱글 로그아웃을 시작할 수 있습니다.

싱글 사인온 서비스(SAML 1.1 및 SAML 2.0)

SAML 1.1 의 경우 생산자는 SSO 서비스를 통해 페더레이션된 리소스에 대해 생산자가 시작한 요청을 처리할 수 있습니다.

SAML 2.0 의 경우 아이덴티티 공급자는 SSO 서비스를 통해 페더레이션된 리소스에 대해 IdP 또는 SP 가 시작한 요청을 처리할 수 있습니다.

생산자/IdP 는 소비자/SP 에서 필요한 정보를 수집하여 어설션을 생성하고 이를 다시 소비자/SP 로 전달합니다. 그러면 소비자/SP 는 이 어설션을 인증을 위해 사용합니다.

UEL(Unified Expression Language)

UEL(Unified Expression Language)은 주로 Java 웹 응용 프로그램에 사용되는 특수한 Java 식 구문입니다. 웹 페이지에 식을 포함하는 용도로 UEL 을 사용할 수 있습니다. CA SiteMinder® Federation Standalone 의 경우 UEL 은 어설션 특성과 신뢰 당사자의 응용 프로그램 특성 간에 매핑을 정의하는 데 사용해야 하는 언어입니다.

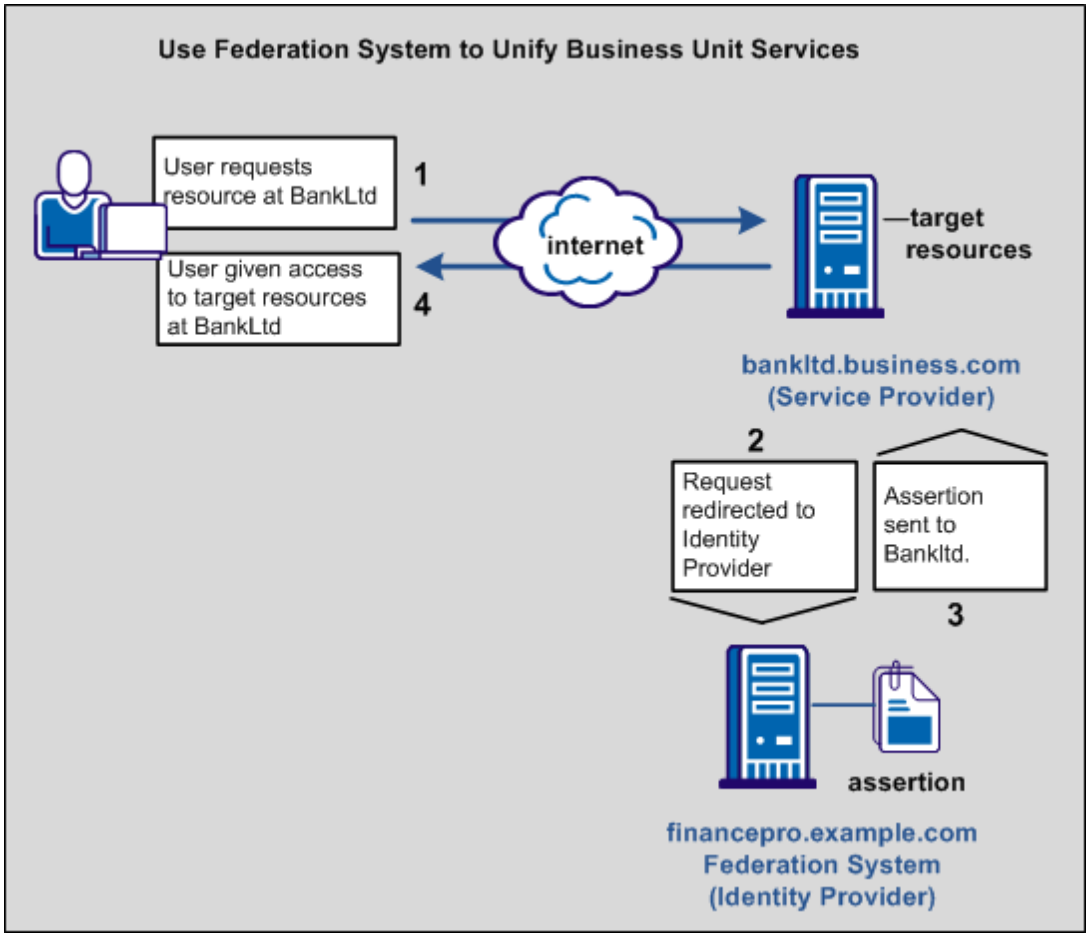
엔터프라이즈의 페더레이션

샘플 비즈니스 사례는 페더레이션 시스템으로 일반 비즈니스 문제를 해결할 수 있는 방법을 가장 잘 보여 줍니다.

이 비즈니스 사례에서 Financepro 는 최근에 고객에게 프라이빗 뱅킹 서비스를 제공하기 위해 BankLtd 라는 금융 회사를 인수한 재무 계획 회사입니다. 이 두 회사는 서로 다른 정보 인프라를 보유하고 있지만 고객에게 한 회사처럼 보이고자 합니다. 이 문제를 해결하기 위해 두 회사는 페더레이션된 파트너 관계를 설정했습니다.

페더레이션된 파트너 관계를 설정함으로써 두 회사는 싱글 사인온을 사용하여 원활한 고객 환경을 제공할 수 있습니다. 고객은 지속적인 인증 챌린지 없이 Financepro 와 BankLtd 간에 이동할 수 있습니다. 또한 고객 ID 와 고객 정보가 공유되므로 사용자 환경을 추가로 사용자 지정하고 각 파트너의 재무 상품을 상호 홍보할 수 있습니다.

다음 그림에서는 Financepro 와 BankLtd 간의 페더레이션된 파트너 관계를 보여 줍니다. 통신 흐름은 SAML 2.0 서비스 공급자에서 시작되는 싱글 사인온을 기반으로 합니다.



그림에서는 다음과 같은 정보 흐름을 설명합니다.

1. 사용자가 BankLtd 에서 페더레이션된 리소스에 액세스하려고 합니다.
2. 사용자가 인증을 위해 Financepro 로 리디렉션되고 어설션이 생성됩니다.
3. 어설션이 BankLtd 로 다시 전달됩니다.
4. SAML HTTP-아티팩트 또는 HTTP-POST 에 기반한 싱글 사인온이 발생합니다. 사용자가 대상 리소스에 대한 액세스 권한을 얻습니다.

이 파트너 관계가 올바로 작동하도록 하려면 CA SiteMinder?Federation Standalone 을 사용하여 관계를 구현하기 전에 파트너 관계의 작동 방식을 결정하십시오.

고려해야 할 문제는 다음과 같습니다.

- 파트너 관계에서 사용자를 식별하는 방법
- 어설션에서 전송되는 특성 및 전송 목적
- 사용할 페더레이션 바인딩(SAML 또는 WS-페더레이션)

결정한 사항은 비즈니스 파트너 관계를 구성하는 데 도움이 됩니다.

파트너 관계에서의 사용자 식별

비즈니스 파트너는 해당 사용자 저장소에서 자체적인 방법으로 사용자 아이덴티티를 정의합니다. 사용자를 식별하는 방법에 따라 파트너 간에 사용자를 매핑할 수 있는 방법이 결정됩니다.

다음 시나리오를 고려하십시오.

- 사용자 ID 가 각 사이트의 사용자 저장소에서 동일합니다.

  계정 연결이 사용자 식별 방법입니다.

- 사용자 ID 가 각 사이트의 사용자 저장소에서 고유합니다.

  아이덴티티 매핑이 사용자 식별 방법입니다. FinancePro 에서는 고객이 JohnDoe 로 식별되고 BankLtd 에서는 해당 고객이 DoeJ 로 식별됩니다. 파트너가 아이덴티티 매핑에 사용할 사용자 특성 프로필에 동의해야 합니다.

- 사용자 ID 가 신뢰 당사자에 존재하지 않습니다.

  계정 프로비저닝이 사용자 식별 방법입니다. 계정을 프로비저닝하는 경우 사용자에 대한 계정을 생성하거나 단순히 SAML 어설션에 포함된 정보로 기존 사용자 계정을 채워야 할 수 있습니다.

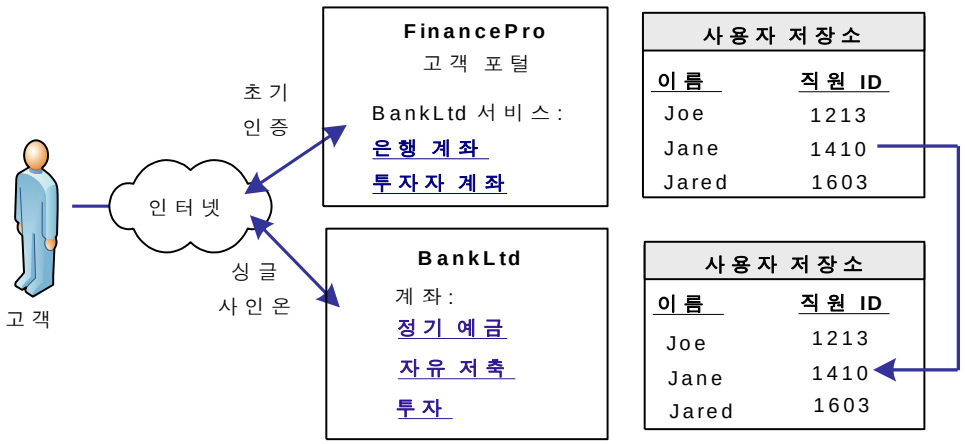
결정한 사용자 식별 방법에 따라 어설션에 포함된 사용자 아이덴티티로 전송되는 정보가 결정됩니다.

페더레이션된 아이덴티티를 설정하기 위한 계정 연결

FinancePro 의 고객이 BankLtd 의 리소스에 액세스하면 NameID 가 어설션에 항상 포함됩니다. 이 식별자를 통해 BankLtd 는 고객을 확인하고 해당 고객에 대해 허용할 액세스 수준을 결정할 수 있습니다.

NameID 는 각 파트너의 사용자 저장소가 동일한 ID 를 사용하여 동일한 방식으로 사용자를 식별할 때 페더레이션된 아이덴티티를 설정할 수 있습니다.

다음 그림에서는 동일한 직원 ID 가 있는 각 사이트의 사용자 저장소를 보여 줍니다.



CA SiteMinder?Federation Standalone 을 통해 파트너 관계 구성 프로세스의 일부로 계정 연결을 구성할 수 있습니다. NameID 형식과 이름 ID 유형을 지정합니다. 그러면 이름 ID 유형에 따라 이름을 정의하는 값의 유형이 결정됩니다. 특정 이름 ID 유형을 사용자 디렉터리의 정적, 사용자 또는 DN 특성과 연결합니다. CA SiteMinder?Federation Standalone 이 어설션에 포함하는 NameID 는 정의하는 구성을 따릅니다.

신뢰 당사자가 어설션을 받으면 BankLtd 에서 사용자 명확성 프로세스가 발생합니다. 이 프로세스를 통해 어설션의 NameID 값이 해당 사용자 저장소의 기록에 연결됩니다.

페더레이션된 아이덴티티를 설정하기 위한 ID 매핑

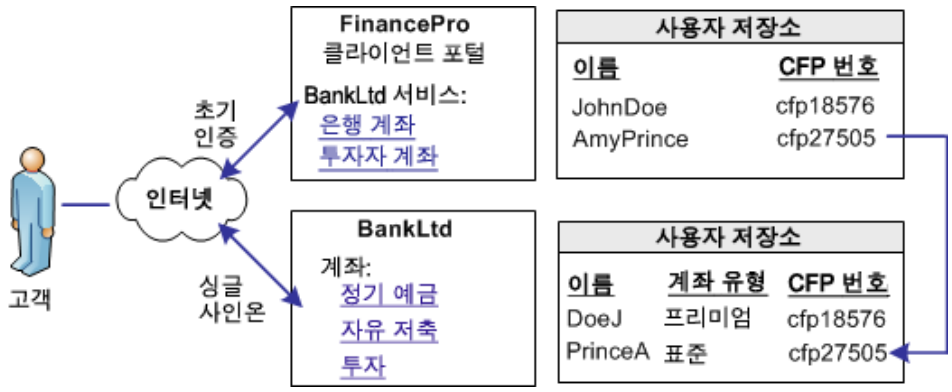
Financepro 에 있는 투자자가 인증을 받은 다음 BankLtd 의 정보에 액세스하기 위해 링크를 선택합니다. 이 투자자는 사인온할 필요 없이 BankLtd 웹 사이트의 계정 영역에 직접 연결됩니다.

BankLtd 가 Financepro 의 모든 고객에 대한 사용자 아이덴티티를 유지 관리하지만 이 아이덴티티는 FinancePro 의 아이덴티티와 다릅니다. 예를 들어 FinancePro 에서는 JohnDoe 가 고객입니다. BankLtd 에서는 해당 고객이 DoeJ 로 식별됩니다. 그럼에도 불구하고 BankLtd 는 회사 웹 사이트의 중요한 부분에 대한 액세스를 제어해야 합니다. 페더레이션된 아이덴티티를 설정하기 위해 파트너가 둘 중 어느 사이트에서나 단일 고객의 해당 아이덴티티에 매핑되는 특성에 동의합니다.

파트너가 대역 외 정보 교환 중에 사용할 특성에 동의합니다. 즉, 이 동의는 채널을 통한 메시지 통신의 일부가 아닙니다. 이 예에서 파트너가 동의하는 특성은 인증된 재무 계획자 라이선스 번호(각 사용자 저장소에서 CFPNum 이라고 함)입니다.

고객이 BankLtd 에서 페더레이션된 리소스에 액세스하려고 하면 요청이 싱글 사인온 프로세스를 트리거합니다. FinancePro 에 생성되는 어설션에는 CFPNum 특성이 포함됩니다. BankLtd 가 어설션을 받으면 해당 사이트에 있는 응용 프로그램이 사용자 명확성 프로세스를 수행해야 합니다. 이 프로세스는 특성을 사용하여 요청에 사용되는 프로필 아이덴티티를 결정합니다.

다음 그림에서는 동일한 사용자가 각 파트너에서 다르게 식별되는 방법을 보여 줍니다.



SiteMinder Federation 을 통해 파트너 관계 구성 프로세스의 일부로 아이덴티티 매핑을 구성할 수 있습니다. NameID 및 특성 구성의 경우 CFPID 라는 특성을 정의합니다. 이 특성을 사용자 특성 CFPNum, 즉 각 파트너의 사용자 저장소에 있는 특성의 이름과 연결하십시오.

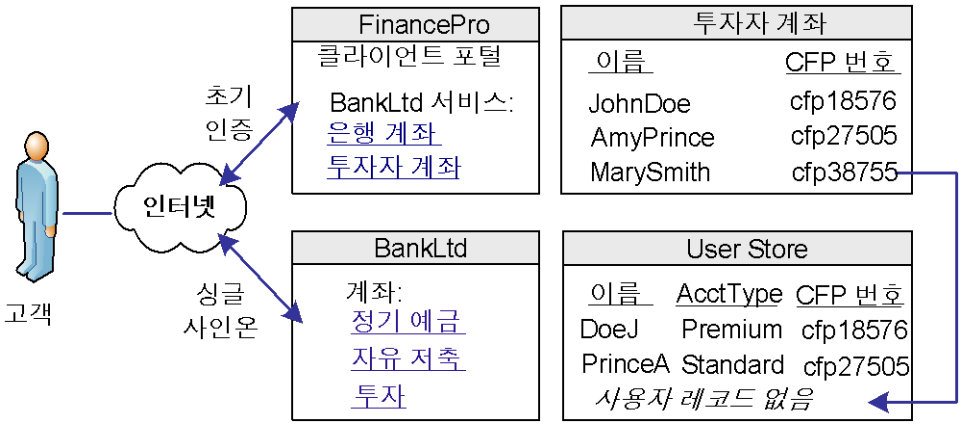
SiteMinder Federation 이 특성을 어설션에 포함합니다. BankLtd 가 어설션을 받으면 사용자 명확성 프로세스가 어설션에 포함된 특성을 해당 사용자 저장소의 적절한 기록에 연결합니다.

페더레이션된 아이덴티티를 설정하기 위한 사용자 프로비저닝

Financepro 에 투자하고 있는 Mary Smith 가 인증을 받은 다음 BankLtd 의 정보에 액세스하기 위해 링크를 클릭합니다. 처음에는 BankLtd 가 Mary Smith 의 사용자 계정을 찾을 수 없습니다. BankLtd 는 새 고객을 허용하면서 해당 웹 사이트의 중요한 부분을 보호하고자 합니다.

BankLtd 가 Mary Smith 의 새 페더레이션된 아이덴티티를 설정하기 위한 프로비저닝을 구현하도록 CA SiteMinder?Federation Standalone 을 구성했습니다. CA SiteMinder?Federation Standalone 이 Mary Smith 를 BankLtd 의 프로비저닝 서버로 리디렉션합니다. 그러면 프로비저닝 응용 프로그램이 CA SiteMinder?Federation Standalone 의 아이덴티티 정보를 사용하여 사용자 저장소에 사용자 계정을 생성합니다.

다음 그림에서는 FinancePro 와 BankLtd 의 사용자 저장소를 보여 줍니다.



CA SiteMinder?Federation Standalone 을 통해 신뢰 당사자에 있는 파트너 관계 구성의 일부로 프로비저닝을 구성할 수 있습니다. 이 예에서 원격 프로비저닝을 선택하고 어설션 데이터를 BankLtd 프로비저닝 서버에 전달하는 방법을 결정합니다. 이 구성을 사용하면 사용자 저장소에 사용자 항목을 동적으로 생성할 수 있습니다.

응용 프로그램을 사용자 지정하기 위한 특성

CA SiteMinder?Federation Standalone 에서는 다음 두 가지 방법으로 특성을 사용하여 대상 응용 프로그램을 사용자 지정할 수 있습니다.

어설션 당사자 측에서 어설션에 특성 추가

응용 프로그램을 사용자 지정하려는 목적으로 사용자 저장소 기록의 특성을 어설션에 포함하여 사용자를 식별할 수 있습니다.

서블릿, 웹 응용 프로그램 및 기타 사용자 지정 응용 프로그램이 특성을 사용하여 사용자 지정된 콘텐츠를 표시하거나 다른 사용자 지정 기능을 사용하거나 사용하지 않도록 설정할 수 있습니다. 웹 응용 프로그램과 함께 사용되는 경우 특성은 대상 사이트의 사용자 활동을 제한하여 세부적인 액세스 제어를 구현할 수 있습니다. 예를 들어 Account Balance 라는 특성 변수를 보내고 BankLtd 에 있는 사용자의 유보 계정 잔액을 반영하도록 설정합니다.

특성은 이름/값 쌍의 형식을 사용합니다. 어설션을 받는 경우 신뢰 당사자는 특성 값을 응용 프로그램에 제공합니다.

신뢰 당사자 측에서 특성 매핑

신뢰 당사자는 대상 응용 프로그램에 전달되고 있는 응용 프로그램 특성 집합에 매핑할 수 있는 어설션 특성 집합을 받습니다.

예를 들어 FinancePro 에는 어설션 특성 CellNo=5555555555 가 포함되어 있습니다. BankLtd 에서 이 특성 이름은 응용 프로그램 특성 Mobile=5555555555 로 변환됩니다. 특성 이름은 변환되지만 값은 동일하게 유지됩니다.

여러 어설션 특성을 단일 응용 프로그램 특성으로 변환할 수도 있습니다. 예를 들어 FinancePro 는 Acct=Savings 및 Type=Retirement 특성이 포함되어 있고 BankLtd 에서 FundType= Retirement Savings 로 변환되는 수신 어설션을 보냅니다.

추가 정보:

파트너 관계 생성 및 활성화 (페이지 169)

싱글 사인온에 대한 페더레이션 프로필

파트너 관계에 대한 프로필은 각 파트너가 지원할 수 있는 바인딩에 따라 결정됩니다.

새 페더레이션의 경우 어느 파트너에 대해서도 필요한 레거시 요구 사항이 없습니다. 따라서 싱글 사인온에 사용하도록 권장되는 프로필은 SAML 2.0 POST 프로필입니다. SAML 2.0 POST 프로필은 어설션 데이터를 안전하게 전송하며 SAML 아티팩트 프로필에 비해 구성 프로세스가 간단합니다. 하지만 두 파트너의 동의에 SAML 아티팩트가 필요한 경우 이 바인딩도 구현할 수 있습니다.

파트너 관계 모델

CA SiteMinder?Federation Standalone 파트너 관계 모델을 사용하면 Financepro 와 BankLtd 사이에 페더레이션을 설정하여 각 회사의 사이트 간에 이동하는 번거로움을 줄이고 두 회사가 한 회사인 것처럼 보이게 할 수 있습니다.

Administrative UI 는 파트너 관계 생성과 싱글 사인온을 수행하기 위한 파트너 관계의 각 파트너 식별에 중점을 둡니다.

이러한 단계는 다음과 같습니다.

1. 파트너 관계 구성 - 파트너 관계를 명명하고 파트너 관계를 구성하는 엔터티 두 개를 식별합니다.
2. 페더레이션 사용자/사용자 ID 설정 - 어설션 당사자가 생성하는 어설션과 신뢰 당사자가 수행하는 인증의 대상이 되는 사용자를 지정합니다.
3. NameID 및 특성 - 페더레이션된 아이덴티티를 설정하는 방법을 결정합니다. 어설션의 콘텐츠를 식별 및 사용자 지정하기 위한 특성을 추가할 수 있습니다.

   NameID 및 특성을 사용하면 적절한 정보가 신뢰 당사자의 응용 프로그램에 제공되도록 할 수 있습니다. 이 단계에서 계정 연결 및 아이덴티티 매핑을 구성합니다.
4. SSO - 신뢰 당사자 측에서 어설션을 소비하는 서비스 위치를 포함하여 싱글 사인온(아티팩트 또는 POST 바인딩)을 정의합니다. SAML 2.0 의 경우 SLO(싱글 로그아웃), ECP(Enhanced Client or Proxy) 프로필, 아이덴티티 공급자 검색 프로필 등의 추가 기능을 구성할 수 있습니다.

5. 서명 및 암호화 - 어설션, 인증 요청 및 SAML 2.0 의 경우 싱글 로그아웃 요청과 응답을 안전하게 교환하기 위한 서명 및 암호화 옵션을 정의합니다.
6. 응용 프로그램 통합 - 대상 응용 프로그램으로의 리디렉션을 구성할 수 있으며 사용자 레코드 프로비저닝을 설정하고 신뢰 당사자 측 특성 매핑을 정의할 수 있습니다. 실패한 사용자 인증에 대한 리디렉션을 설정할 수도 있습니다.

제 2 장: Administrative UI

이 섹션은 다음 항목을 포함하고 있습니다.

Administrative UI 개요

구성은 Administrative UI 를 통해 관리됩니다. Administrative UI 는 관리자가 제품의 모든 시스템 관리 기능에 액세스할 수 있는 웹 응용 프로그램입니다.

관리자란 페더레이션된 솔루션을 관리할 수 있는 권한과 책임을 가진 사용자입니다. 여러 명의 관리자가 시스템 관리를 담당할 수도 있습니다. 자세한 내용은 여러 명의 관리자를 구성하는 절차를 참조하십시오.

Administrative UI 는 구성 탭, 하위 범주, 목록 및 태스크 단추로 구성됩니다.

UI 에서 이동할 때 다음 사항에 유의하십시오.

- 기본 구성 범주는 다음과 같은 탭으로 구성됩니다.
  - 페더레이션
  - 인증서 및 키
  - 사용자 디렉터리
  - 인프라

  개체를 구성할 때 이러한 탭 중 하나로 먼저 이동해야 합니다.
- 각각의 기본 구성 탭에 있는 하위 범주는 페더레이션 설정의 특정 측면을 구성할 때 선택할 수 있습니다.
- 대부분의 하위 범주에 개체 목록이 표시됩니다. 이러한 목록에는 기존 개체에 액세스할 수 있는 상황에 맞는 링크와 개체를 생성하거나 수정할 수 있는 태스크 단추가 포함되어 있습니다.

개체 관리

Administrative UI 에서 개체를 생성, 확인, 수정 및 삭제할 수 있습니다. 각 태스크를 수행하는 구체적인 방법은 개체마다 다르지만 전반적인 방법은 유사합니다. 예를 들어 페더레이션 엔터티를 삭제하는 절차는 사용자 디렉터리 연결을 삭제하는 절차와 유사합니다.

각 하위 범주 내의 목록에서는 개체를 조작할 수 있습니다. 다음 중 하나를 수행할 수 있습니다.

- 새 개체를 생성합니다.
- 개체 목록에서 기존 개체를 선택하여 수정합니다.
- "작업" 단추를 사용하여 개체에 태스크를 수행합니다.

새 개체 생성

구성 탭 중 하나에서 하위 범주를 선택하면 개체 목록이 표시됩니다. 개체 목록에서 개체를 생성할 수 있습니다.

예를 들어 새로운 파트너 관계를 구성하려면 "페더레이션" 탭을 선택하여 "페더레이션 파트너 관계 보기" 창을 표시하십시오. "페더레이션 파트너 관계" 목록에서 "파트너 관계 만들기"를 클릭합니다.

개체를 생성하는 단추를 클릭하면 과정을 안내할 적절한 대화 상자나 구성 마법사가 표시됩니다.

개체 목록

UI 에서 구성 탭이나 하위 범주를 보면 관련된 개체 목록도 표시됩니다. 목록에서 원하는 개체의 링크를 클릭하면 해당 개체에 대한 세부 정보를 볼 수 있습니다.

예를 들어 "페더레이션" 탭에서 "엔터티"를 선택하면 "페더레이션 엔터티 보기" 페이지에 페더레이션 엔터티 목록이 표시됩니다.

개체 목록의 작업 단추

"작업" 단추는 개체 목록에 포함된 모든 개체의 왼쪽에 표시되며 단추를 클릭하면 개체에 수행할 수 있는 태스크 메뉴가 표시됩니다.

개체에 따라 "작업" 단추에 표시되는 태스크가 다릅니다.

개체 목록 필터링

특정 개체 유형에 대해 구성된 항목의 목록이 매우 긴 경우에는 목록을 읽기 쉽게 하기 위해 표시되는 항목을 필터링할 수 있습니다. 예를 들어 활성 상태인 구성된 모든 파트너 관계를 검색하면 "페더레이션 파트너 관계" 목록에 해당 정보만 표시됩니다.

검색 필터를 지정하려면

1. 기본 구성 탭을 클릭합니다.

   "필터" 그룹 상자와 "목록" 그룹 상자가 표시됩니다.

2. **필터** *개체* 그룹 상자에서 다음 지침에 따라 검색을 구성합니다.

   a. "검색 대상" 필드에서 검색할 항목을 선택합니다.

      예: 인증서의 경우 "별칭"을 피연산자로 선택할 수 있습니다. 페더레이션 엔터티의 경우 "엔터티 유형"을 피연산자로 선택할 수 있습니다.

   b. 가운데 필드의 풀다운 메뉴에서 연산자를 선택합니다.

      예: =, 다음으로 시작, 다음 포함, 다음으로 끝남, 현재 또는 이전("만료 날짜"에 대해 선택할 수 있는 유일한 옵션)

c. 다음 작업 중 하나를 수행하십시오.

- "만료 날짜"를 제외한 피연산자의 경우 마지막 필드에 문자열(따옴표 사용 안 함)을 입력합니다. 이 문자열은 검색 필터 값입니다.
- "만료 날짜"의 경우 필드 오른쪽의 달력 아이콘을 선택하고 날짜를 선택합니다. 날짜를 수동으로 입력할 수 있지만 달력에서 날짜를 선택하면 올바른 날짜 형식인지 확인할 수 있습니다.

참고:

- 전체 목록을 검색하려면 세 번째 필드를 비워 두십시오. <ANY> 또는 별표(*)를 입력할 수도 있습니다.
- 별표는 포함된 와일드카드 문자로 사용할 수 없습니다. 예를 들어 별표를 단독으로 입력할 수는 있지만 **partner*** 를 값으로 입력할 수 없습니다.

3. "실행"을 클릭하여 검색을 시작합니다.

페이지 화면

UI 의 특정 개체에 대한 목록을 표시하면 기본적으로 레코드가 10 개만 표시됩니다. 목록의 오른쪽 아래에 있는 보다 큼 기호(>)를 선택하면 다음 페이지로 이동합니다. 2 배 보다 큼 기호(>>)를 선택하면 목록의 맨 끝으로 이동합니다.

참고: 한 페이지에 표시되는 기본 레코드 수(10 개)는 변경할 수 없습니다.

목록 오른쪽 위에 있는 "모두 표시" 링크를 선택하면 창 하나에 모든 항목을 표시할 수 있습니다.

개체 구성을 위한 마법사

UI 에는 여러 개체에 사용되는 구성 마법사가 제공됩니다. 엔터티 또는 파트너 관계를 생성하거나 편집할 때, 인증서를 가져올 때 또는 메타데이터 파일을 가져올 때 적절한 마법사가 나타납니다.

UI 마법사는 특정 개체를 구성하는 단계를 안내합니다. 특정 단계에 나오는 필수 설정을 지정하지 않으면 누락된 정보를 입력하라는 메시지가 대화 상자 위쪽에 표시됩니다. 모든 필수 필드를 완료하기 전에는 다음 단계로 이동할 수 없습니다.

Administrative UI 에 로그인합니다.

관리자는 Administrative UI 를 통해 페더레이션 엔터티를 구성합니다. 여러 명의 관리자를 구성하고 각 관리자에게 서로 다른 Administrative UI 액세스 권한 수준을 할당할 수 있습니다. 자세한 내용은 여러 관리자 구성의 지침을 참조하십시오.

다음 단계를 수행하십시오.

1. 브라우저에서 Java 스크립트를 사용하도록 설정합니다. JavaScript 는 Administrative UI 를 여는 데 필요합니다.
2. 사용하는 플랫폼에 대한 지침을 따르십시오.

   Windows

   "시작", "모든 프로그램", "CA", "Federation Standalone", "CA SiteMinder® Federation Standalone 관리 UI"를 차례로 선택합니다.

   UNIX

   웹 브라우저를 열고 URL 로 http://*fed_server:ui_port*/ca/federation/adminui 를 입력합니다.

   fed_server:ui_port

   UI 의 포트를 포함하여 CA SiteMinder® Federation Standalone 이 설치되어 있는 서버의 정규화된 도메인 이름을 지정합니다. 기본 포트는 8888 입니다.

   예:

   http://fed1.example.com:8888/ca/federation/adminui

   로그인 창이 나타납니다.

3. 사용자 이름과 암호를 입력하고 "Log in"(로그인)을 클릭합니다.

   중요! 기본 관리자의 사용자 이름은 항상 **admin** 입니다. 이 이름은 변경할 수 없습니다. 기본 관리자 암호는 설치 시 설정됩니다.

   Administrative UI 가 시작됩니다.

Active Directory 를 사용할 경우의 UI 로그인 암호 조건

Active Directory 를 관리 인증을 위한 사용자 저장소로 구성할 경우 Administrative UI 에 로그인할 때 다음과 같은 암호 조건이 적용됩니다.

- "사용 안 함" 상태인 사용자는 Administrative UI 에 로그인할 수 없습니다. 이 경우 "Error: Invalid user name or password."(오류: 잘못된 사용자 이름 또는 암호)라는 메시지가 표시됩니다.
- 사용자가 만료된 경우 Administrative UI 에 로그인할 수 없습니다. 이 경우 "Error: Invalid user name or password."(오류: 잘못된 사용자 이름 또는 암호)라는 메시지가 표시됩니다.
- 사용자에 대해 "다음에 로그인할 때 암호 변경" 특성이 설정되어 있으면 사용자가 Administrative UI 에 로그인할 수 있습니다. Administrative UI 에서는 암호를 관리하지 않습니다.

제 3 장: 간단한 파트너 관계를 사용하여 시작

이 섹션은 다음 항목을 포함하고 있습니다.

기본 SAML 2.0 파트너 관계

CA SiteMinder?Federation Standalone 을 시작하는 한 가지 방법은 파트너 관계를 구성하는 것입니다. 이 장에서는 기본 SAML 2.0 페더레이션 파트너 관계인 SAML 2.0 POST 프로필을 사용하여 싱글 사인온을 설정하는 방법을 설명합니다. 기본 구성으로 시작하여 최소 개수의 단계를 완료하면 제품의 작동 방식을 확인할 수 있습니다.

참고: 이 파트너 관계는 SAML 2.0 에 중점을 두지만 전체 프로세스는 SAML 1.1 과 동일합니다. 파트너 관계의 각 단계에서 구성 설정은 SAML 프로토콜에 따라 다를 수 있습니다.

또한 이 장에서는 실제 프로덕션 환경을 반영하여 디지털 서명 및 싱글 로그아웃과 같은 추가 기능의 구성도 설명합니다. 구성에 아티팩트 바인딩을 추가할 수도 있습니다.

이 장에서 사용하는 샘플 네트워크에는 파트너 관계인 두 사이트 모두에 CA SiteMinder?Federation Standalone 이 설치되어 있는 것으로 가정합니다. 하지만 한 사이트에는 CA SiteMinder?Federation Standalone 을 설치하고 다른 사이트에는 다른 SAML 호환 제품을 설치하여 파트너 관계를 설정할 수도 있습니다.

두 사이트 모두에 CA SiteMinder?Federation Standalone 이 있으면 파트너 관계를 구성하는 관점을 이해해야 합니다. 완전한 파트너 관계를 구성하려면 각 사이트의 통신 방향마다 하나씩, 각 사이트에서 *파트너 관계 정의*를 정의하는 것부터 시작하십시오. 예를 들어 로컬 사이트가 아이덴티티 공급자(IdP)인 경우 로컬 IdP-원격 SP 파트너 관계를 구성하십시오. 이 구성은 하나의 파트너 관계 정의입니다. 파트너 관계 구성을 완료하려면 로컬 SP 에서 역방향의 로컬 SP-원격 IdP 파트너 관계를 구성하십시오.

파트너 관계 정의는 항상 로컬과 원격 엔터티를 구분합니다. 로컬 엔터티는 CA SiteMinder?Federation Standalone 을 구성하는 사이트의 엔터티입니다. 이 엔터티는 CA SiteMinder?Federation Standalone 이 설치되어 있는 시스템과 같을 필요는 없지만 같은 도메인에 속해 있어야 합니다. 원격 엔터티는 CA SiteMinder?Federation Standalone 을 구성하는 도메인과 다른 도메인에 있는 파트너의 엔터티입니다.

다음 프로세스는 CA SiteMinder?Federation Standalone 이 두 사이트 모두에 있을 때 기본 CA SiteMinder?Federation Standalone 파트너 관계를 생성하는 단계를 보여 줍니다.

1. 사용자 디렉터리 연결을 설정합니다.
2. 로컬 및 원격 엔터티를 생성합니다.
3. IdP 사이트에서 로컬 IdP-SP 파트너 관계 정의를 구성합니다.
4. SP 사이트에서 로컬 SP-IdP 파트너 관계 정의를 구성합니다.
5. 파트너 관계를 활성화합니다.
6. 파트너 관계를 테스트합니다.

샘플 페더레이션 네트워크

처음 생성하는 파트너 관계는 다음의 샘플 네트워크와 같습니다.

비즈니스 파트너

- 이름이 IdP1 인 아이덴티티 공급자
- 이름이 SP1 인 서비스 공급자

SAML 프로필 및 기능

- POST 프로필을 사용하는 SAML 2.0
- 싱글 사인온
- 서명 처리 없음
- FIPS_COMPAT 모드

CA SiteMinder® Federation Standalone 배포 모드

Standalone - SiteMinder 커넥터 없음

IdP 의 SSO 서비스 URL

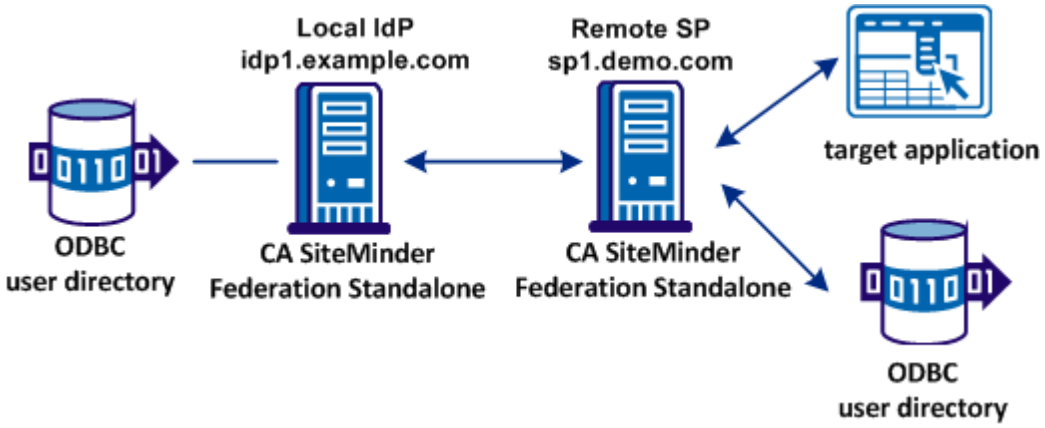
http://idp1.example.com:9090/affwebservices/public/saml2sso

SP 의 어설션 소비자 서비스 URL

http://sp1.demo.com:9091/affwebservices/public/saml2assertionconsumer

참고: 이 샘플 네트워크를 구현하려면 CA SiteMinder® Federation Standalone 이 설치된 시스템이 두 개 필요합니다.

다음 그림에서는 샘플 파트너 관계를 보여 줍니다.



IdP 파트너 구성

다음 구성 프로세스는 IdP1 의 관리자 관점에서 작성된 것입니다. 따라서 IdP1 은 로컬 IdP 입니다.

다음은 IdP 파트너를 설정하는 프로세스입니다.

1. Administrative UI 에 로그온합니다.
2. 사용자 디렉터리 연결을 설정합니다.
3. IdP 및 SP 엔터티를 식별합니다.
4. "파트너 관계 SAML2 IdP->SP 만들기"를 클릭합니다.
5. 파트너 관계 마법사를 따라 최소한의 필수 설정을 구성합니다.

사용자 디렉터리 연결 설정

파트너 관계를 설정하려면 먼저 사용자 디렉터리에 대한 연결을 정의해야 합니다.

다음에 제공되는 절차는 제품과 함께 설치되는 기본 데이터 원본을 사용하여 ODBC 사용자 디렉터리에 연결하는 방법을 보여 줍니다.

중요! CA FedManager 데이터 원본은 페더레이션 시스템 정책이 저장되는 위치입니다. 이 예에서는 이 데이터 원본을 사용자 디렉터리로 사용하지만 프로덕션 환경에서는 다른 데이터 원본을 사용하십시오.

이 데이터 원본을 사용하려면

- 데이터 원본의 샘플 사용자에 대해 스키마를 설정합니다.
- 디렉터리에 대한 연결을 설정합니다.

데이터 원본의 샘플 사용자 설정

ODBC 스키마 및 샘플 데이터를 가져와서 데이터 저장소의 샘플 사용자를 설정할 수 있습니다.

제품에는 CA FedManager 데이터 원본에 샘플 사용자를 저장하기 위한 스키마와 데이터를 생성하는 스크립트 파일이 포함되어 있습니다. CA SiteMinder?Federation Standalone 을 설치할 때 지정한 SQL Server 또는 Oracle 데이터베이스에 이 데이터를 저장할 수 있습니다.

다음 단계를 수행하십시오.

1. 다음 디렉터리로 이동합니다.

   Windows(기본 위치): *federation_install_dir*\siteminder\db\SQL

   UNIX: *federation_install_dir*/siteminder/db/sql

2. 데이터베이스에 샘플 사용자를 채우는 데 필요한 스키마 파일을 가져옵니다. 데이터베이스에 맞는 도구를 사용하여 가져오기를 수행합니다.

   다음 파일을 가져옵니다.

   - smsampleusers_sqlserver.sql

     SQL 서버 데이터베이스의 샘플 사용자에 대한 스키마를 생성하고 데이터베이스를 샘플 사용자로 채웁니다.

   - smsampleusers_oracle.sql

     Oracle 데이터베이스의 샘플 사용자에 대한 스키마를 생성하고 데이터베이스를 샘플 사용자로 채웁니다.

   예를 들어 스크립트를 살펴보면 이름이 GeorgeC 이고 암호가 siteminder 인 샘플 사용자가 있습니다.

3. 스키마 가져오기가 완료되면 디렉터리에 연결합니다.

ODBC 디렉터리에 연결

ODBC 사용자 디렉터리를 채울 올바른 스키마를 가져온 후 사용자 디렉터리에 대한 연결을 설정하십시오.

다음 단계를 수행하십시오.

1. 웹 브라우저를 열고 다음 URL 을 입력하여 Administrative UI 에 로그인합니다.

   http://idp1.example.com:8888/ca/federation/adminui

   CA SiteMinder® Federation Standalone 은 서버 이름 idp1.example.com 으로 설치되어 있습니다. 브라우저에서 이 호스트 이름을 CA SiteMinder® Federation Standalone 이 설치되어 있는 IP 주소에 매핑합니다.

   참고: Administrative UI 를 열려면 브라우저에 JavaScript 가 설정되어 있는지 확인하십시오.

2. Administrative UI 에서 "사용자 디렉터리" 탭을 선택합니다.

   "사용자 디렉터리 보기" 대화 상자가 표시됩니다.

3. "ODBC 에 연결"을 클릭합니다.

   "ODBC 에 연결" 대화 상자가 열립니다.

4. "ODBC 사용자 디렉터리 구성" 그룹 섹션에서 다음의 필수 필드에 값을 지정합니다.

   디렉터리 이름

   FedSQL

   데이터 원본

   CA FedManager 데이터 원본

5. "연결 자격 증명" 그룹 섹션에서 다음 필드를 완성합니다.

   연결할 자격 증명 필요

   확인란을 선택합니다.

   사용자 이름

   데이터베이스에 액세스할 때 사용하는 이름을 입력합니다.

암호

데이터베이스에 액세스할 때 사용하는 암호를 입력합니다.

암호 확인

데이터베이스 암호를 다시 입력합니다.

6. "디렉터리 필드" 그룹 섹션에서 다음 필드를 완성합니다.

유니버설 ID 열

"유니버설 ID"로 사용되는 ODBC 디렉터리 특성의 이름을 입력합니다. 이 값은 사용자의 아이덴티티를 유지 관리하기 위해 CA SiteMinder® Federation Standalone 과 통신하는 다른 응용 프로그램에 전달될 수 있습니다. SiteMinder 커넥터를 사용하고 있으면 이 필드는 필수 항목입니다.

7. "저장"을 클릭합니다.

"사용자 디렉터리 보기" 대화 상자로 돌아갑니다.

8. "작업", "연결 테스트"를 선택하여 CA SiteMinder® Federation Standalone 이 사용자 디렉터리에 연결할 수 있도록 합니다.

연결이 성공했는지 나타내는 메시지가 표시됩니다.

계속해서 IdP 와 SP 엔터티를 구성합니다.

파트너 관계 엔터티 구성

사용자 디렉터리 연결을 설정했으면 파트너 관계의 로컬 측과 원격 측을 식별해야 합니다. Administrative UI 에서는 각 파트너를 엔터티라고 합니다.

다음 절차에서는 로컬 및 원격 엔터티에 제공할 값을 보여 줍니다. 그러나 실제 네트워크 구성에서는 어설션 당사자와 신뢰 당사자 양쪽 모두에서 각각 로컬 엔터티를 생성하고, 로컬 엔터티를 메타데이터 파일로 내보낸 다음 해당 메타데이터 파일을 서로 교환하여 각각에서 원격 엔터티를 정의할 수 있습니다.

다음 단계를 수행하십시오.

1. "페더레이션" 탭에서 "엔터티"를 선택합니다.
2. "엔터티 만들기"를 클릭합니다.
3. 엔터티 마법사의 첫 번째 단계에서 다음 사항을 선택하고 "다음"을 클릭합니다.

   엔터티 위치

   로컬

   새 엔터티 유형

   SAML2 IDP

4. 마법사의 두 번째 단계에서 필드에 다음과 같이 데이터를 입력하고 "다음"을 클릭합니다.

   엔터티 ID

   idp1

   이 값으로 파트너는 엔터티를 식별합니다.

   엔터티 이름

   idp1

   이 값은 CA SiteMinder® Federation Standalone 데이터베이스에서 내부적으로 엔터티 개체를 식별합니다. 파트너는 이 값을 인식하지 않습니다.

   기준 URL

   http://idp1.example.com:9090

   다른 설정은 그대로 둡니다.

   참고: "엔터티 이름"이 "엔터티 ID"와 동일한 값일 수 있지만, 이런 경우에는 값을 사이트의 다른 엔터티와 공유하면 안 됩니다.

5. 마지막 단계에서 설정을 검토하고 "마침"을 클릭합니다.

"페더레이션 엔터티 보기" 창으로 돌아갑니다. 이제 원격 파트너를 구성합니다.

원격 SP 엔터티를 생성하려면

1. "페더레이션 엔터티 보기" 창에서 시작합니다.
2. "페더레이션 엔터티 목록"에서 "엔터티 만들기"를 클릭합니다.

   "엔터티 만들기" 대화 상자가 표시됩니다.

3. 엔터티 마법사의 첫 번째 단계에서 다음 사항을 선택하고 "다음"을 클릭합니다.

   엔터티 위치

   원격

   새 엔터티 유형

   SAML2 SP

4. 마법사의 두 번째 단계에서 필드에 다음과 같이 데이터를 입력하고 "다음"을 클릭합니다.

   엔터티 ID

   sp1

   이 값으로 파트너는 엔터티를 식별합니다.

   엔터티 이름

   sp1

   이 값은 CA SiteMinder® Federation Standalone 데이터베이스에서 내부적으로 엔터티 개체를 식별합니다. 파트너는 이 값을 인식하지 않습니다.

   "어설션 소비자 서비스 URL" 그룹 상자

   인덱스

   0

   바인딩

   HTTP-POST

   URL

   http://sp1.demo.com:9091/affwebservices/public/saml2assertionconsumer

   기본값

   항목 행에 대해 이 열의 확인란을 선택합니다.

   다른 설정은 그대로 둡니다.

5. 마지막 단계에서 설정을 검토하고 "마침"을 클릭합니다.

원격 SP 엔터티가 구성됩니다.

로컬 엔터티 및 원격 엔터티가 구성되었으면 이제 파트너 관계를 생성할 수 있습니다.

IdP-SP 파트너 관계 생성

파트너 관계 엔터티를 생성한 후에는 파트너 관계 마법사에 따라 IdP -> SP 파트너 관계를 구성하십시오. 먼저 파트너 관계의 이름과 기타 기본 정보를 지정해야 합니다.

다음 단계를 수행하십시오.

1. "페더레이션" 탭을 선택합니다.
2. "파트너 관계 SAML2 IdP -> SP 만들기"를 클릭합니다.

   이 옵션을 선택하는 것은 현재 로컬 IdP 라는 것을 의미합니다.

   파트너 관계 마법사의 첫 번째 단계가 나타납니다.
3. 필드에 다음 값을 입력합니다.

   파트너 관계 이름

   TestPartnership

   로컬 IDP ID

   idp1

   (풀다운 목록에서 선택)

   원격 SP ID

   sp1

   (풀다운 목록에서 선택)

   기준 URL

   http://idp1.example.com:9090

   이 값은 기본적으로 제공됩니다.

   차이 시간(초)

   기본값을 적용합니다.
4. ODBC 디렉터리(FedSQL)를 "사용 가능한 디렉터리" 상자에서 "선택한 디렉터리" 상자로 이동합니다.
5. "다음"을 클릭하여 "페더레이션 사용자" 단계로 이동합니다.

어설션 생성을 위한 페더레이션 사용자 지정

"페더레이션 사용자" 대화 상자에서 IdP 가 어설션을 생성할 사용자를 선택합니다.

다음 단계를 수행하십시오.

1. 기본값을 적용합니다.
2. "다음"을 눌러 계속합니다.

기본값을 적용하면 SiteMinder 가 사용자 디렉터리의 모든 사용자에 대해 어설션을 생성할 수 있도록 지정하게 됩니다.

어설션에 이름 ID 추가

"어설션 구성" 단계에서는 NameID 의 형식 및 값과 사용자를 식별하는 특성을 지정할 수 있습니다. 이러한 특성은 어설션에 포함됩니다.

참고: NameID 는 항상 어설션에 포함됩니다.

이 구성에서는 이름 ID 만 지정하십시오. 다른 특성을 추가하지 마십시오.

다음 단계를 수행하십시오.

1. "어설션 구성" 단계에서 다음 필드에 대한 값을 입력합니다.

   이름 ID 형식

   지정되지 않음

   이름 ID 유형

   정적

   값

   GeorgeC
2. "다음"을 클릭하여 SSO(싱글 사인온)를 설정합니다.

싱글 사인온 설정

파트너 간에 싱글 사인온을 설정하려면 SSO 설정을 구성하십시오.

다음 단계를 수행하십시오.

1. 파트너 관계 마법사의 SSO 및 SLO 단계부터 시작합니다.
2. "로컬 인증 유형" 필드와 "인증 클래스" 필드의 기본값(기본)을 그대로 사용합니다.
3. "SSO 바인딩" 필드에 대해 "HTTP-POST"를 선택합니다.
4. 원격 SP 엔터티가 이미 생성된 경우 "어설션 소비자 URL" 값이 자동으로 채워집니다.
5. "다음"을 클릭하여 "서명 및 암호화" 단계로 이동합니다.

서명 처리 사용 안 함

이 간단한 파트너 관계에서는 서명 처리가 사용되지 않도록 지정하십시오. 하지만 프로덕션 환경에서는 아이덴티티 공급자가 어설션에 서명해야 합니다.

다음 단계를 수행하십시오.

1. "서명 및 암호화" 단계에서 "서명 처리 사용 안 함"을 선택합니다.
2. "다음"을 클릭하여 다음 단계로 이동합니다.

IdP-SP 파트너 관계 설정 확인

페더레이션 파트너 관계의 한쪽에 대한 파트너 관계 정의를 완료했습니다. 이제 설정을 확인하십시오.

다음 단계를 수행하십시오.

1. "확인" 대화 상자에서 파트너 관계에 대한 설정을 검토합니다.
2. 설정을 수정하려면 원하는 섹션에서 "수정"을 클릭합니다.
3. 원하는 대로 구성되었으면 "마침"을 클릭합니다.

파트너 관계의 IdP 측이 완료되었습니다. IdP 시스템이 아닌 다른 시스템에서 파트너 관계의 SP 측을 정의하십시오.

SP 파트너 구성

다음 구성 프로세스는 SP(이 예에서는 SP1)의 관리자 관점에서 작성된 것입니다. 따라서 SP1 은 로컬 SP 입니다.

다음은 SP 파트너를 설정하는 프로세스입니다.

1. Administrative UI 에 로그온합니다.
2. 사용자 디렉터리 연결을 설정합니다.
3. IdP 및 SP 엔터티를 식별합니다.
4. "파트너 관계 SAML2 SP->IdP 만들기"를 클릭합니다.
5. 파트너 관계 마법사를 따라 최소한의 필수 설정을 구성합니다.

사용자 디렉터리 연결 설정

파트너 관계를 설정하려면 먼저 사용자 디렉터리에 대한 연결을 정의해야 합니다.

다음에 제공되는 절차는 CA SiteMinder?Federation Standalone 과 함께 설치되는 기본 데이터 원본을 사용하여 ODBC 사용자 디렉터리에 연결하는 방법을 보여 줍니다.

중요! CA FedManager 데이터 원본은 CA SiteMinder?Federation Standalone 정책이 저장되는 위치입니다. 이 예에서는 이 데이터 원본을 사용자 디렉터리로 사용하지만 프로덕션 환경에서는 다른 데이터 원본을 사용하십시오.

이 데이터 원본을 사용하려면

- 데이터 원본의 샘플 사용자에 대해 스키마를 설정합니다.
- 디렉터리에 대한 연결을 설정합니다.

데이터 원본의 샘플 사용자 설정

ODBC 스키마 및 샘플 데이터를 가져와서 데이터 저장소의 샘플 사용자를 설정할 수 있습니다.

제품에는 CA FedManager 데이터 원본에 샘플 사용자를 저장하기 위한 스키마와 데이터를 생성하는 스크립트 파일이 포함되어 있습니다. CA SiteMinder?Federation Standalone 을 설치할 때 지정한 SQL Server 또는 Oracle 데이터베이스에 이 데이터를 저장할 수 있습니다.

다음 단계를 수행하십시오.

1. 다음 디렉터리로 이동합니다.

   Windows(기본 위치): *federation_install_dir*\siteminder\db\SQL

   UNIX: *federation_install_dir*/siteminder/db/sql

2. 데이터베이스에 샘플 사용자를 채우는 데 필요한 스키마 파일을 가져옵니다. 데이터베이스에 맞는 도구를 사용하여 가져오기를 수행합니다.

   다음 파일을 가져옵니다.

   - smsampleusers_sqlserver.sql

     SQL 서버 데이터베이스의 샘플 사용자에 대한 스키마를 생성하고 데이터베이스를 샘플 사용자로 채웁니다.

   - smsampleusers_oracle.sql

     Oracle 데이터베이스의 샘플 사용자에 대한 스키마를 생성하고 데이터베이스를 샘플 사용자로 채웁니다.

   예를 들어 스크립트를 살펴보면 이름이 GeorgeC 이고 암호가 siteminder 인 샘플 사용자가 있습니다.

3. 스키마 가져오기가 완료되면 디렉터리에 연결합니다.

ODBC 디렉터리에 연결

ODBC 사용자 디렉터리를 채울 올바른 스키마를 가져온 후 사용자 디렉터리에 대한 연결을 설정하십시오.

다음 단계를 수행하십시오.

1. 웹 브라우저를 열고 다음 URL 을 입력하여 Administrative UI 에 로그인합니다.

   http://idp1.example.com:8888/ca/federation/adminui

   CA SiteMinder® Federation Standalone 은 서버 이름 idp1.example.com 으로 설치되어 있습니다. 브라우저에서 이 호스트 이름을 CA SiteMinder® Federation Standalone 이 설치되어 있는 IP 주소에 매핑합니다.

   참고: Administrative UI 를 열려면 브라우저에 JavaScript 가 설정되어 있는지 확인하십시오.

2. Administrative UI 에서 "사용자 디렉터리" 탭을 선택합니다.

   "사용자 디렉터리 보기" 대화 상자가 표시됩니다.

3. "ODBC 에 연결"을 클릭합니다.

   "ODBC 에 연결" 대화 상자가 열립니다.

4. "ODBC 사용자 디렉터리 구성" 그룹 섹션에서 다음의 필수 필드에 값을 지정합니다.

   디렉터리 이름

   FedSQL

   데이터 원본

   CA FedManager 데이터 원본

5. "연결 자격 증명" 그룹 섹션에서 다음 필드를 완성합니다.

   연결할 자격 증명 필요

   확인란을 선택합니다.

   사용자 이름

   데이터베이스에 액세스할 때 사용하는 이름을 입력합니다.

암호

데이터베이스에 액세스할 때 사용하는 암호를 입력합니다.

암호 확인

데이터베이스 암호를 다시 입력합니다.

6. "디렉터리 필드" 그룹 섹션에서 다음 필드를 완성합니다.

   유니버설 ID 열

   "유니버설 ID"로 사용되는 ODBC 디렉터리 특성의 이름을 입력합니다. 이 값은 사용자의 아이덴티티를 유지 관리하기 위해 CA SiteMinder® Federation Standalone 과 통신하는 다른 응용 프로그램에 전달될 수 있습니다. SiteMinder 커넥터를 사용하고 있으면 이 필드는 필수 항목입니다.

7. "저장"을 클릭합니다.

   "사용자 디렉터리 보기" 대화 상자로 돌아갑니다.

8. "작업", "연결 테스트"를 선택하여 CA SiteMinder® Federation Standalone 이 사용자 디렉터리에 연결할 수 있도록 합니다.

   연결이 성공했는지 나타내는 메시지가 표시됩니다.

계속해서 IdP 와 SP 엔터티를 구성합니다.

파트너 관계 엔터티 식별

사용자 디렉터리 연결을 설정했으면 파트너 관계의 로컬 측과 원격 측을 식별해야 합니다. Administrative UI 에서는 각 파트너를 엔터티라고 합니다.

다음 절차에서는 로컬 및 원격 엔터티에 제공할 값을 보여 줍니다. 그러나 실제 네트워크 구성에서는 어설션 당사자와 신뢰 당사자 양쪽 모두에서 각각 로컬 엔터티를 생성하고, 로컬 엔터티를 메타데이터 파일로 내보낸 다음 해당 메타데이터 파일을 서로 교환하여 각각에서 원격 엔터티를 정의할 수 있습니다.

다음 단계를 수행하십시오.

1. "페더레이션" 탭에서 "엔터티"를 선택합니다.
2. "엔터티 만들기"를 클릭합니다.

   "엔터티 만들기" 대화 상자가 표시됩니다.

3. 엔터티 마법사의 첫 번째 단계에서 다음 사항을 선택하고 "다음"을 클릭합니다.

   엔터티 위치

   로컬

   새 엔터티 유형

   SAML2 SP

4. 두 번째 단계에서 다음과 같이 필드를 완성하고 "다음"을 클릭합니다.

   엔터티 ID

   sp1

   이 값으로 파트너는 엔터티를 식별합니다.

   엔터티 이름

   sp1

   이 값은 CA SiteMinder® Federation Standalone 데이터베이스에서 내부적으로 엔터티 개체를 식별합니다. 파트너는 이 값을 인식하지 않습니다.

   기준 URL

   http://sp1.demo.com:9091

   참고: 엔터티 ID 와 이름은 아이덴티티 공급자에서 원격 SP 엔터티에 대해 지정한 것과 동일해야 합니다.

5. 설정을 검토하고 "마침"을 클릭합니다.

"페더레이션 엔터티 보기" 창으로 돌아갑니다. 이제 원격 파트너를 구성합니다.

원격 IdP 를 생성하려면

1. "페더레이션 파트너 관계 보기" 창에서 시작합니다.
2. "페더레이션 엔터티 목록"에서 "엔터티 만들기"를 클릭합니다.

   "엔터티 만들기" 대화 상자가 표시됩니다.

3. 엔터티 마법사의 첫 번째 단계에서 다음 사항을 선택하고 "다음"을 클릭합니다.

   엔터티 위치

   원격

   새 엔터티 유형

   SAML2 IDP

4. 마법사의 두 번째 단계에서 필드에 다음과 같이 데이터를 입력합니다.

   엔터티 ID

   idp1

   이 값으로 파트너는 엔터티를 식별합니다.

   엔터티 이름

   idp1

   이 값은 CA SiteMinder® Federation Standalone 데이터베이스에서 내부적으로 엔터티 개체를 식별합니다. 파트너는 이 값을 인식하지 않습니다.

   참고: 엔터티 ID 와 이름은 아이덴티티 공급자 측과 동일해야 합니다.

   "SSO 서비스 URL" 그룹 상자

   바인딩

   HTTP-리디렉션

   URL

   http://idp1.example.com:9090/affwebservices/public/saml2sso

5. 설정을 검토하고 "마침"을 클릭합니다.

로컬 엔터티 및 원격 엔터티가 구성된 후에는 파트너 관계를 생성할 수 있습니다.

SP-IdP 파트너 관계 생성

파트너 관계 엔터티를 생성한 후 파트너 관계 마법사에 따라 SP -> IdP 파트너 관계의 필수 구성 요소를 구성하십시오.

다음 단계를 수행하십시오.

1. "페더레이션" 탭을 선택합니다.

2. "파트너 관계 SAML2 SP->IdP 만들기"를 클릭합니다.

   파트너 관계 마법사의 첫 번째 단계가 나타납니다.

3. 필드에 다음 값을 입력합니다.

   파트너 관계 이름

   DemoPartnership

   로컬 SP ID

   sp1

   원격 IDP ID

   idp1

   차이 시간(초)

   기본값을 적용합니다.

4. ODBC 디렉터리(FedSQL)를 "사용 가능한 디렉터리" 상자에서 "선택한 디렉터리" 상자로 이동합니다.
5. "다음"을 클릭하여 "사용자 ID" 단계로 이동합니다.

사용자 ID 특성 지정

어설션에서 사용자를 식별하는 데 사용할 특성을 지정합니다. 이 아이덴티티 특성 값은 사용자 명확성 프로세스, 즉 SP 의 사용자 디렉터리에서 사용자 레코드를 찾는 프로세스에 사용됩니다.

다음 단계를 수행하십시오.

1. "사용자 ID" 단계로 이동합니다.
2. "어설션에서 아이덴티티 특성 선택" 그룹 상자에서 기본값인 "이름 ID 사용"을 그대로 사용합니다.

3. "사용자 디렉터리에 아이덴티티 특성 매핑" 그룹 상자에 다음을 입력합니다.

   ODBC 검색 사양

   Name=%s

   이 항목은 CA SiteMinder® Federation Standalone 에서 변수(%s)를 어설션의 이름 ID 특성 값으로 대체하고 샘플 사용자 데이터베이스의 이름 열과 일치시키도록 지시합니다. 일치하는 항목이 발견되면 사용자의 명확성이 확인되어 대상 리소스에 대한 액세스가 허용됩니다.

4. "다음"을 클릭하여 싱글 사인온을 구성합니다.

싱글 사인온 구성

파트너 간에 싱글 사인온을 설정하려면 SSO 설정을 구성하십시오.

다음 단계를 수행하십시오.

1. SSO 및 SLO 단계에서 시작합니다.
2. "SSO 바인딩" 필드에 대해 "HTTP-POST"를 선택합니다.
3. "대상" 필드의 SP 에서 대상 리소스를 지정합니다.

   이 샘플 파트너 관계에서 이 대상은 http://spapp.demo.com:80/spsample/welcome.html 입니다.
4. "리디렉션 모드" 필드에서 "데이터 없음"을 선택합니다.
5. 원격 IdP 가 이미 생성된 경우 "SSO 서비스 URL" 값이 자동으로 채워집니다.
6. "다음"을 클릭하여 "서명 및 암호화" 단계로 이동합니다.

서명 처리 사용 안 함

이 간단한 파트너 관계에서는 서명 처리가 사용되지 않도록 지정하십시오. 하지만 프로덕션 환경에서는 아이덴티티 공급자가 어설션에 서명해야 합니다.

다음 단계를 수행하십시오.

1. "서명 및 암호화" 단계에서 "서명 처리 사용 안 함"을 선택합니다.
2. "다음"을 클릭하여 다음 단계로 이동합니다.

SP 파트너 설정 확인

페더레이션 파트너 관계의 로컬 SP 측에 대한 파트너 관계를 완료했습니다.

다음 단계를 수행하십시오.

1. "확인" 대화 상자에서 SP 파트너에 대한 설정을 검토합니다.
2. 설정을 수정하려면 해당 섹션에서 "수정"을 클릭합니다.
3. 원하는 대로 구성되었으면 "마침"을 클릭합니다.

파트너 관계의 SP 측이 구성되었습니다.

파트너 관계 활성화

파트너 관계의 양측이 각각 정의되었으므로 이제 파트너 관계를 활성화할 수 있습니다.

CA SiteMinder?Federation Standalone 은 파트너 관계의 양측 모두에 설치되므로 IdP 와 SP 에서 파트너 관계를 활성화해야 합니다.

다음 단계를 수행하십시오.

1. "페더레이션" 탭에서 "파트너 관계"를 선택합니다.

   "페더레이션 파트너 관계 보기" 창이 나타납니다.
2. "페더레이션 파트너 관계" 목록에서 활성화할 항목을 찾습니다. "상태" 열의 값이 "정의됨"인지 확인합니다. 상태가 "미완성"이면 파트너 관계를 편집하여 모든 필수 설정이 구성되었는지 확인해야 합니다.
3. 활성화할 파트너 관계 항목 옆의 "작업", "활성화"를 선택합니다.

   "활성화 확인" 대화 상자가 표시됩니다.
4. "활성화 확인" 대화 상자에서 "예"를 클릭합니다.

   파트너 관계가 활성화되고 "상태" 열의 값이 "활성"이 됩니다.

파트너 관계 테스트(POST 프로필)

파트너 관계를 구성한 후 두 파트너 간에 싱글 사인온을 테스트합니다.

테스트 작업에는 다음이 포함됩니다.

- 싱글 사인온을 시작할 웹 페이지 생성
- 요청된 페더레이션 리소스 역할을 할 대상 웹 페이지 생성
- 단일 사인 온 테스트

기본 파트너 관계를 테스트한 후 샘플 구성을 추가로 변경할 수 있습니다.

싱글 사인온을 시작할 웹 페이지 생성

테스트를 위하여 싱글 사인온을 시작하는 링크가 있는 HTML 페이지를 직접 생성하십시오. IdP 또는 SP 에서 싱글 사인온을 시작할 수 있습니다. 이 예에서는 SP 에서 시작되는 싱글 사인온을 보여 줍니다.

다음 단계를 수행하십시오.

1. SP 사이트에서 샘플 HTML 페이지를 생성합니다. 다음과 같이 SP 에서 AuthnRequest 서비스에 하드 코딩된 링크를 포함합니다.

   ```
   <a href="http://sp1.demo.com:9091/affwebservices/public/
   saml2authnrequest?ProviderID=idp1.example.com>
   Link to Test POST Single Sign-on</a>
   ```

   이 링크는 AuthnRequest 서비스에 사용자를 지정된 아이덴티티 공급자로 리디렉션하여 인증 컨텍스트를 검색하도록 지시합니다.

2. 웹 페이지를 testsso.html 이라는 이름으로 저장합니다.
3. testsso.html 을 웹 서버 문서 루트 디렉터리의 /spsample 하위 폴더 아래에 복사합니다.

   이 샘플 네트워크에서는 대상 웹 서버가 http://spapp.demo:80 입니다.

대상 리소스 생성

싱글 사인온을 테스트하기 위한 마지막 단계는 대상 리소스를 생성하는 것입니다.

다음 단계를 수행하십시오.

1. SP 사이트에 샘플 HTML 페이지를 만들고 다음과 같은 메시지를 포함합니다.

   Welcome to SP1

   Single Sign-on is successful

2. 웹 페이지를 welcome.html 이라는 이름으로 저장합니다.
3. welcome.html 을 웹 서버 문서 루트 디렉터리의 /spsample 하위 폴더 아래에 복사합니다.

   이 샘플 네트워크에서는 대상 웹 서버가 http://spapp.demo.com:80 입니다.

POST 싱글 사인온 테스트

샘플 웹 페이지를 설정한 후에는 싱글 사인온을 테스트하고 파트너 관계 구성이 완료되었는지 확인하십시오.

다음 단계를 수행하십시오.

1. 파트너 관계의 양측이 모두 Administrative UI 에서 활성화되었는지 확인합니다.
2. 브라우저를 엽니다.

3. 싱글 사인온을 트리거하는 링크가 포함된 웹 페이지의 URL 을 입력합니다. 이 예에서는 다음 URL 을 입력합니다.

   http://spapp.demo.com:80/spsample/testsso.html

   참고: 이 샘플 네트워크에서는 CA SiteMinder® Federation Standalone 을 독립 실행형 모드로 배포했기 때문에 대상 웹 서버는 CA SiteMinder® Federation Standalone 이 설치된 서버가 아닌 다른 서버입니다.

   URL 을 입력하면 "Link to Test POST Single Sign-on"(테스트 POST 싱글 사인온 링크)이라는 링크가 있는 페이지가 표시됩니다.

4. **Link to Test POST Single Sign-on(테스트 POST 싱글 사인온 링크)**을 클릭합니다.

   싱글 사인온이 시작됩니다. SP 의 AuthnRequest 서비스에서 아이덴티티 공급자의 싱글 사인온 서비스로 사용자가 리디렉션됩니다.

아이덴티티 공급자는 사용자를 인증하고 세션을 설정한 후 사용자를 서비스 공급자의 대상 리소스인 welcome.html 에 다시 연결합니다. SP 에 생성한 샘플 시작 페이지가 표시되면 싱글 사인온이 성공했음을 알 수 있습니다.

서명 처리가 사용되도록 설정

SAML 2.0 POST 싱글 사인온에는 디지털 서명된 어설션이 필요합니다. 서명 및 확인 태스크를 위해 개인 키/인증서 쌍이 사용됩니다.

트랜잭션이나 런타임 작업 전에 IdP1 에서 관리자가 인증서 데이터가 포함된 파일을 SP1 에 보냅니다. 이 파일에는 IdP1 에서 어설션에 서명하는 데 사용되는 개인 키 관련 인증서(공개 키)가 포함되어 있습니다. SP1 의 관리자는 인증서를 인증서 데이터 저장소에 추가합니다.

싱글 사인온 트랜잭션이 발생하면 IdP1 은 개인 키를 사용하여 어설션에 서명합니다. SP1 은 어설션을 수신하고 인증서 데이터 저장소의 인증서를 사용하여 어설션 서명을 확인합니다.

다음 절차는 각 사이트에서 서명을 설정하는 방법을 설명합니다.

IdP 에서 서명 처리 구성

POST 싱글 사인온의 경우 Idp1 이 어설션에 서명해야 합니다. 이 경우 인증서 데이터 저장소에 있는 개인 키를 사용하여 어설션에 서명합니다.

참고: 이 예에서는 키와 인증서를 가져올 수 있는 파일이 있거나, 서명 및 확인 태스크에 사용할 수 있는 개인 키와 인증서가 이미 있다고 가정합니다.

다음 단계를 수행하십시오.

1. UI 에서 "페더레이션" 탭을 클릭하고 "파트너 관계"를 선택합니다.

   "페더레이션 파트너 관계 보기" 창이 나타납니다.

2. IdP ->SP 파트너 관계인 TestPartnership 에 대한 항목 옆의 "작업", "비활성화"를 선택합니다.

   파트너 관계를 편집하기 전에 비활성화합니다.

3. TestPartnership 에 대한 항목 옆의 "작업", "수정"을 클릭합니다.

   "파트너 관계" 마법사의 첫 번째 단계에 해당하는 대화 상자가 열립니다.

4. 파트너 관계 마법사에서 "서명 및 암호화" 단계를 클릭합니다.

5. "서명" 그룹 상자에서 다음을 수행합니다.

   a. "서명 처리 사용 안 함"의 선택을 취소합니다.

   b. "서명 개인 키 별칭" 필드 옆에서 "가져오기"를 클릭합니다.

      "인증서/개인 키 가져오기" 창이 열립니다.

6. 다음과 같이 가져오기 마법사를 완료합니다.

   a. 개인 키/인증서 쌍을 가져올 파일을 선택합니다.

   b. 파일이 pkcs#12 파일인 경우 파일을 암호화할 암호를 지정합니다.

   c. 가져올 파일에서 인증서 항목을 선택하고 "별칭"의 값을 cert1 과 같이 입력합니다.

   d. 선택을 확인하고 "마침"을 클릭합니다.

   "페더레이션 파트너 관계 보기" 창으로 돌아갑니다.

7. 파트너 관계 항목에 대해 "작업", "수정"을 선택합니다.

8. "서명 및 암호화" 단계로 이동합니다. 이제 가져온 키/인증서를 대화 상자의 "서명 개인 키 별칭" 드롭다운 목록에서 사용할 수 있습니다.

9. cert1 의 별칭을 선택하고 "다음"을 클릭합니다.
10. "확인" 대화 상자에서 설정을 검토하고 "마침"을 클릭합니다.

    "페더레이션 파트너 관계 보기" 창으로 돌아갑니다.
11. "페더레이션 파트너 관계" 목록에서 TestPartnership 항목 옆의 "작업", "활성화"를 선택하여 파트너 관계를 다시 활성화합니다.
12. 운영 환경에 맞게 페더레이션 서비스를 다시 시작합니다.

    - **Windows**

      다음과 같이 중지 및 시작 바로 가기를 사용합니다. 로컬 관리자가 아닌 네트워크 사용자로 로그인한 경우에는 바로 가기를 마우스 오른쪽 단추로 클릭하고 "관리자 권한으로 실행"을 선택하십시오.

      a. "시작", "모든 프로그램", "CA", "Federation Standalone", "서비스 중지"

      b. "시작", "모든 프로그램", "CA", "Federation Standalone", "서비스 시작"

    - **UNIX**

      a. 명령 창을 엽니다.

      b. 다음 스크립트를 실행합니다.

      *federation_install_dir/*fedmanager.sh stop

      federation_install_dir/fedmanager.sh start

      참고: 서비스를 중지했다가 루트 사용자로 시작하지 마십시오.

    페더레이션 서비스를 다시 시작하면 서명에 대한 변경 내용이 시스템에 적용됩니다.

이제 IdP 에서 서명 처리가 구성되었습니다.

SP 에서 서명 처리 구성

SP1 에서 어설션의 서명을 확인해야 합니다. 트랜잭션 전에 SP1 은 IdP1 로부터 인증서(공개 키)를 수신해야 합니다. 이 인증서는 IdP1 이 어설션에 서명하는 데 사용하는 개인 키와 관련된 인증서입니다.

이 인증서를 SP1 인증서 데이터 저장소로 가져와야 합니다.

다음 단계를 수행하십시오.

1. Administrative UI 에서 "페더레이션" 탭을 클릭하고 "파트너 관계"를 선택합니다.

   "페더레이션 파트너 관계 보기" 창이 나타납니다.

2. DemoPartnership 에 대한 항목 옆의 "작업", "비활성화"를 선택합니다.

   파트너 관계를 편집하기 전에 비활성화합니다.

3. DemoPartnership 에 대한 항목 옆의 "작업", "수정"을 클릭합니다.

   "파트너 관계" 마법사의 첫 번째 단계에 해당하는 대화 상자가 열립니다.

4. 파트너 관계 마법사에서 "서명 및 암호화" 단계를 클릭합니다.
5. "서명" 그룹 상자에서 다음을 수행합니다.
   a. "서명 처리 사용 안 함"의 선택을 취소합니다.
   b. "확인 인증서 별칭" 필드 옆의 "가져오기"를 클릭합니다.

      "인증서/개인 키 가져오기" 창이 열립니다.

6. 다음과 같이 가져오기 마법사를 완료합니다.
   a. 인증서를 가져올 파일을 선택합니다.
   b. 가져올 파일에서 인증서 항목을 선택하고 "별칭"의 값을 cert1 과 같이 입력합니다.
   c. 선택을 확인하고 "마침"을 클릭합니다.

   "페더레이션 파트너 관계 보기" 창으로 돌아갑니다.

7. 파트너 관계 항목에 대해 "작업", "수정"을 선택합니다.
8. "서명 및 암호화" 단계로 이동합니다. 이제 가져온 키/인증서를 대화 상자의 "서명 개인 키 별칭" 드롭다운 목록에서 사용할 수 있습니다.
9. 인증서의 별칭인 cert1 을 선택하고 "다음"을 클릭합니다.

10. "확인" 대화 상자에서 설정을 검토하고 "마침"을 클릭합니다.

    "페더레이션 파트너 관계 보기" 창으로 돌아갑니다.

11. "페더레이션 파트너 관계" 목록에서 DemoPartnership 항목 옆의 "작업", "활성화"를 선택하여 파트너 관계를 다시 활성화합니다.

12. 운영 환경에 맞게 페더레이션 서비스를 다시 시작합니다.

    - **Windows**

      다음과 같이 중지 및 시작 바로 가기를 사용합니다. 로컬 관리자가 아닌 네트워크 사용자로 로그인한 경우에는 바로 가기를 마우스 오른쪽 단추로 클릭하고 "관리자 권한으로 실행"을 선택하십시오.

      a. "시작", "모든 프로그램", "CA", "Federation Standalone", "서비스 중지"

      b. "시작", "모든 프로그램", "CA", "Federation Standalone", "서비스 시작"

    - **UNIX**

      a. 명령 창을 엽니다.

      b. 다음 스크립트를 실행합니다.

      federation_install_dir/fedmanager.sh stop

      federation_install_dir/fedmanager.sh start

      참고: 서비스를 중지했다가 루트 사용자로 시작하지 마십시오.

    페더레이션 서비스를 다시 시작하면 서명 변경 내용이 시스템에 적용됩니다.

이제 SP 에서 서명 확인이 구성되었습니다.

싱글 로그아웃 추가

SLO(싱글 로그아웃) 프로토콜을 통해 로그아웃을 시작한 브라우저에 대한 모든 사용자의 세션이 동시에 종료됩니다. 싱글 로그아웃을 구성하면 권한 없는 사용자가 서비스 공급자의 리소스에 액세스할 수 있도록 열려 있는 세션이 남지 않게 됩니다.

IdP 에서 싱글 로그아웃 구성

Idp1 에서 싱글 로그아웃을 구성합니다.

다음 단계를 수행하십시오.

1. Administrative UI 에 로그인합니다.
2. "페더레이션", "파트너 관계"를 클릭합니다.

   "페더레이션 파트너 관계 보기" 창이 나타납니다.
3. TestPartnership 에 대한 항목 옆의 "작업", "비활성화"를 선택합니다.

   편집하기 전에 비활성화해야 합니다.
4. TestPartnership 에 대한 항목 옆의 "작업", "수정"을 클릭합니다.

   파트너 관계의 첫 번째 단계에 해당하는 대화 상자가 열립니다.
5. "SSO 및 SLO" 단계를 클릭합니다.
6. "SLO" 섹션의 "SLO 바인딩" 값으로 "HTTP-리디렉션"을 선택하여 싱글 로그아웃을 사용하도록 설정합니다.
7. "SLO 서비스 URL" 테이블에서 "행 추가"를 클릭하고 다음 필드를 완성합니다.

   SLO 위치 URL

   http://sp1.demo.com:9091/affwebservices/public/saml2slo

   이 링크는 싱글 로그아웃 요청이 원격 SP 로 전송되었음을 나타냅니다.

   SLO 확인 URL

   http://idp1.example.com:9090/idpsample/SLOConfirm.html

   이 링크는 싱글 로그아웃을 시작한 사이트(이 경우 IdP1)의 확인 페이지입니다. 싱글 로그아웃이 성공적으로 완료되면 사용자가 이 페이지로 리디렉션됩니다.
8. "선택" 열의 옵션 단추를 클릭하여 방금 구성한 행을 선택합니다.
9. 마법사에서 "확인" 단계를 클릭하고 구성을 검토합니다.

10. "마침"을 클릭합니다.

    "페더레이션 파트너 관계 보기" 창으로 돌아갑니다.

11. "페더레이션 파트너 관계" 목록에서 TestPartnership 항목 옆의 "작업", "활성화"를 선택하여 파트너 관계를 다시 활성화합니다.

이제 IdP1 에서 싱글 로그아웃이 구성에 추가되었습니다.

SP 에서 싱글 로그아웃 구성

SP1 에서 싱글 로그아웃을 구성합니다.

다음 단계를 수행하십시오.

1. Administrative UI 에 로그인합니다.
2. "페더레이션", "파트너 관계"를 클릭합니다.
3. "Demo Partnership"(데모 파트너 관계)에 대한 항목 옆의 "작업", "비활성화"를 선택합니다.

   파트너 관계를 편집하려면 먼저 비활성화해야 합니다.

4. DemoPartnership 에 대한 항목 옆의 "작업", "수정"을 클릭합니다.

   "파트너 관계" 마법사의 첫 번째 단계에 해당하는 대화 상자가 열립니다.

5. "SSO 및 SLO" 단계를 클릭합니다.
6. "SLO" 그룹 상자에서 "SLO 바인딩" 값으로 "HTTP-리디렉션"을 선택하여 싱글 로그아웃을 사용하도록 설정합니다.
7. 사용할 수 있는 행이 없으면 "SLO 서비스 URL" 테이블에서 "행 추가"를 클릭하여 다음 필드를 완성합니다.

   SLO 위치 URL

   http://idp1.example.com:9090/affwebservices/public/saml2slo

   이 링크는 싱글 로그아웃 요청이 전송되는 위치입니다.

   SLO 확인 URL

   http://sp1.demo.com:9091/spsample/SLOConfirm.html

   이 URL 은 로그아웃을 시작한 사이트의 싱글 로그아웃 확인 페이지입니다.

8. "선택" 열의 라디오 단추를 클릭하여 방금 구성한 행을 선택합니다.

9. 마법사에서 "확인" 단계를 클릭하고 구성을 검토합니다.
10. "마침"을 클릭합니다.

    "페더레이션 파트너 관계 보기" 창으로 돌아갑니다.
11. "페더레이션 파트너 관계" 목록에서 DemoPartnership 항목 옆의 "작업", "활성화"를 선택하여 파트너 관계를 다시 활성화합니다.

이제 싱글 로그아웃이 SP 에서 구성되었습니다.

싱글 로그아웃 테스트

싱글 로그아웃을 구성한 후에는 이를 테스트하십시오. 이 테스트의 경우 싱글 로그아웃이 SP1 에서 시작됩니다.

SP 에서 싱글 로그아웃을 시작하려면 싱글 로그아웃을 시작하고 확인할 두 개의 웹 페이지가 있어야 합니다.

- welcome.html 을 사용하여 브라우저를 IdP1 의 싱글 로그아웃 서비스로 리디렉션하는 링크를 이 페이지에 추가하십시오. 이 링크의 구문은 다음과 같습니다.

  ```
  <a href="http://idp1.example.com:9090/affwebservices/public/
  saml2slo>Log Me Out</a>
  ```
- 다음과 같은 로그아웃 확인 메시지가 있는 SLOConfirm.html 이라는 이름의 확인 페이지를 생성하십시오.

  ```
  <p>성공적으로 로그아웃했습니다</p>
  ```

두 페이지를 모두 웹 서버 루트 디렉터리의 하위 폴더 /spsample 아래에 복사하십시오.

참고: SLO 를 테스트할 수 있도록 SSO 트랜잭션을 완료하십시오.

다음 단계를 수행하십시오.

1. 파트너 관계의 양측이 모두 Administrative UI 에서 활성화되었는지 확인합니다.
2. 이전에 설명한 지침에 따라 싱글 사인온을 구성하고 테스트합니다.

   싱글 사인온에 성공하면 브라우저에 시작 페이지가 표시됩니다.
3. 브라우저를 열어 두고 시작 페이지에서 **Log Me Out** 링크를 클릭합니다.

   성공하면 다음 메시지가 표시되는 확인 페이지로 리디렉션됩니다.

   성공적으로 로그아웃했습니다.

SSO 에 대한 아티팩트 프로필 설정

기본 파트너 관계는 싱글 사인온에 대한 HTTP-POST 바인딩으로 시작합니다. 하지만 파트너 관계에 SAML 2.0 아티팩트 프로필을 사용할 수 있습니다.

HTTP-아티팩트 바인딩을 구성하는 절차는 마법사의 SSO 및 SLO 단계까지는 POST 바인딩의 절차와 같습니다.

IdP 에서 아티팩트 SSO 구성

이 절차는 SSO 에 대한 HTTP-아티팩트 프로필을 구성하는 방법을 보여 줍니다.

다음 단계를 수행하십시오.

1. Administrative UI 에서 "페더레이션" 탭을 클릭하고 "파트너 관계"를 선택합니다.
2. TestPartnership 에 대한 항목 옆의 "작업", "비활성화"를 선택합니다.

   편집하기 전에 비활성화해야 합니다.
3. TestPartnership 에 대한 항목 옆의 "작업", "수정"을 클릭합니다.

   "파트너 관계" 마법사의 첫 번째 단계에 해당하는 대화 상자가 열립니다.
4. "SSO 및 SLO" 단계를 클릭합니다.
5. "인증" 그룹 상자의 기존 설정을 유지합니다.

6. "SSO" 그룹 상자에서 다음을 수행합니다.
   a. "SSO 바인딩" 필드에서 "HTTP-아티팩트"를 선택합니다.
   b. "어설션 소비자 서비스 URL" 테이블의 바인딩을 "HTTP-아티팩트"로 변경합니다. URL 은 POST 프로필에 사용했던 것과 동일하게 유지해도 됩니다.
7. "백 채널" 그룹 상자에서 다음을 선택합니다.

   인증 방법

   인증 없음
8. "SLO" 및 "IDP 검색" 그룹 상자로 건너뜁니다.
9. "확인" 단계를 클릭하여 구성을 검토합니다.
10. "마침"을 클릭하여 구성을 완료합니다.

이제 Idp1 에서 아티팩트 바인딩이 구성되었습니다.

SP 에서 아티팩트 SSO 구성

이 절차는 SSO 에 대한 HTTP-아티팩트 프로필을 구성하는 방법을 설명합니다.

다음 단계를 수행하십시오.

1. Administrative UI 에서 "페더레이션" 탭을 클릭하고 "파트너 관계"를 선택합니다.

   "페더레이션 파트너 관계 보기" 창이 나타납니다.
2. "Demo Partnership"(데모 파트너 관계)에 대한 항목 옆의 "작업", "비활성화"를 선택합니다.

   파트너 관계를 편집하기 전에 비활성화합니다.
3. DemoPartnership 에 대한 항목 옆의 "작업", "수정"을 클릭합니다.

   "파트너 관계" 마법사의 첫 번째 단계에 해당하는 대화 상자가 열립니다.
4. "SSO 및 SLO" 단계를 클릭합니다.

5. "SSO" 그룹 상자에서 다음 태스크를 수행합니다.
   a. "SSO 바인딩" 필드에서 "HTTP-아티팩트"를 선택합니다.
   b. "리디렉션 모드" 필드에서 "데이터 없음"을 선택합니다. URL 은 POST 프로필에 사용했던 것과 동일하게 유지해도 됩니다.
   c. "SSO 서비스 URL" 설정은 변경하지 않고 그대로 둡니다.
6. "SOAP 아티팩트 레졸루션 URL" 그룹 상자에서 "행 추가"를 클릭하고, 백 채널에 인증이 필요 없음을 나타내기 위해 다음 URL 을 입력합니다.

   http://idp1.example.com:9090/affwebservices/saml2artifactresolutionnoauth

   테이블의 "선택" 열에서 라디오 단추를 클릭하여 이 항목을 반드시 선택해야 합니다.
7. "백 채널" 그룹 상자에서 다음 옵션을 선택합니다.

   인증 방법

   인증 없음
8. "SLO" 및 "상태 리디렉션 URL" 그룹 상자는 건너뜁니다.
9. "확인" 단계를 클릭하여 구성을 검토합니다.
10. "마침"을 클릭하여 구성을 완료합니다.

이제 SP1 에 아티팩트 바인딩이 구성되었습니다.

파트너 관계 테스트(아티팩트 SSO)

파트너 관계의 양쪽이 작동하면 두 파트너 간에 싱글 사인온을 테스트합니다.

IdP1 이 요청을 받으면 아티팩트를 생성합니다. 그런 다음 아티팩트가 SP1 에 전송됩니다.

SP1 이 아티팩트를 받으면 요청을 다시 IdP1 에 리디렉션합니다. IdP 가 어설션을 검색하여 SP1 에 반환합니다.

싱글 사인온을 시작할 웹 페이지 생성(아티팩트)

테스트를 위하여 싱글 사인온을 시작하는 링크가 있는 HTML 페이지를 직접 생성하십시오. IdP 또는 SP 에서 싱글 사인온을 시작할 수 있습니다. 이 예에서는 SP 에서 시작되는 싱글 사인온을 보여 줍니다.

다음 단계를 수행하십시오.

1. SP 사이트에서 샘플 HTML 페이지를 생성하고 다음과 같이 SP 의 AuthnRequest 서비스에 대한 하드 코딩된 링크를 포함합니다.

   Link for ARTIFACT Single Sign-on

   이 링크는 AuthnRequest 서비스에 사용자를 지정된 아이덴티티 공급자로 리디렉션하여 사용자 인증 컨텍스트를 검색하도록 지시합니다.

2. 웹 페이지를 testartifact.html 이라는 이름으로 저장합니다.
3. testartifact.html 을 웹 서버 문서 루트 디렉터리의 /spsample 하위 폴더 아래에 복사합니다.

   이 샘플 네트워크에서는 대상 웹 서버가 http://spapp.demo:80 입니다.

대상 리소스 생성

싱글 사인온을 테스트하기 위한 마지막 단계는 대상 리소스를 생성하는 것입니다.

다음 단계를 수행하십시오.

1. SP 사이트에 샘플 HTML 페이지를 만들고 다음과 같은 메시지를 포함합니다.

   Welcome to SP1

   Single Sign-on is successful

2. 웹 페이지를 welcome.html 이라는 이름으로 저장합니다.
3. welcome.html 을 웹 서버 문서 루트 디렉터리의 /spsample 하위 폴더 아래에 복사합니다.

   이 샘플 네트워크에서는 대상 웹 서버가 http://spapp.demo.com:80 입니다.

아티팩트 싱글 사인온 테스트

샘플 웹 페이지를 설정한 후에는 싱글 사인온을 테스트하고 파트너 관계 구성이 성공적인지 확인하십시오.

다음 단계를 수행하십시오.

1. 파트너 관계의 양측이 활성화되었는지 확인합니다.
2. 브라우저를 엽니다.
3. 다음과 같이 싱글 사인온을 트리거하는 웹 페이지에 대한 URL 을 입력합니다.

   http://spapp.demo.com:80/spsample/testartifact.html

   참고: 이 샘플 네트워크에서는 CA SiteMinder® Federation Standalone 을 독립 실행형 모드로 배포했기 때문에 대상 웹 서버는 CA SiteMinder® Federation Standalone 이 설치된 서버가 아닌 다른 서버입니다.

   URL 을 입력하면 "Link to Test ARTIFACT Single Sign-on"(테스트 아티팩트 싱글 사인온 링크)이라는 링크가 있는 페이지가 표시됩니다.
4. **Link to Test ARTIFACT Single Sign-on**(테스트 아티팩트 싱글 사인온 링크)을 클릭하면 싱글 사인온이 시작됩니다.

   SP 의 AuthnRequest 서비스에서 아이덴티티 공급자의 싱글 사인온 서비스로 사용자가 리디렉션됩니다.

아이덴티티 공급자는 세션을 설정한 후에 사용자를 다시 서비스 공급자의 대상 리소스(welcome.html)로 리디렉션합니다. SP 에서 생성한 시작 샘플 페이지가 표시되어 싱글 사인온이 성공적으로 이루어졌음을 알 수 있습니다.

간단한 파트너 관계 이상의 구성 절차

이 장에서 설명한 간단한 파트너 관계는 CA SiteMinder?Federation Standalone 을 사용하여 페더레이션된 파트너 관계를 구성하는 방법을 간략하게 보여 줍니다.

안내서의 나머지 장에서는 CA SiteMinder?Federation Standalone 을 사용하여 수행 가능한 모든 태스크에 대한 자세한 절차를 설명합니다. 자세한 구성 지침에 대해서는 이 절차와 Administrative UI 의 도움말을 참조하십시오.

추가 정보:

페더레이션 엔터티 구성 (페이지 127)
파트너 관계 생성 및 활성화 (페이지 169)
인증을 위한 사용자 디렉터리 연결 (페이지 87)

제 4 장: 사용자 세션, 어설션 및 만료 데이터 저장

이 섹션은 다음 항목을 포함하고 있습니다.

페더레이션 기능에 세션 저장소가 필요함

세션 저장소에는 다음 페더레이션 기능에 대한 데이터가 저장됩니다.

- HTTP-아티팩트 싱글 사인온(SAML 1.x 또는 2.x)

  SAML 어설션 및 연결된 아티팩트가 어설션 당사자 측에서 생성됩니다. 아티팩트가 생성된 어설션을 식별합니다. 어설션 당사자는 신뢰 당사자에 아티팩트를 반환합니다. 신뢰 당사자는 아티팩트를 사용하여 어설션 당사자가 세션 저장소에 저장한 어설션을 검색합니다.

  이 프로세스가 작동하려면 영구 세션이 필요합니다.

  참고: SAML POST 프로필은 세션 저장소에 어설션을 저장하지 않습니다.

- HTTP-POST 단일 사용 정책(SAML 2.0 및 WS-페더레이션)

  단일 사용 정책 기능은 어설션이 신뢰 당사자 측에서 두 번째 세션을 설정하는 데 재사용되지 않도록 합니다. 신뢰 당사자는 자체 세션 저장소에 만료 데이터라고 하는 어설션에 대한 시간 기반 데이터를 저장합니다. 만료 데이터는 어설션이 한 번만 사용되었는지 확인합니다.

  신뢰 당사자에 세션 저장소가 필요하지만 영구 세션은 필요하지 않습니다.

- 싱글 로그아웃(SAML 2.0)

  싱글 로그아웃이 사용되도록 설정된 경우 어느 파트너든지 사용자 세션에 대한 정보를 저장할 수 있습니다. 세션 정보는 세션 저장소에 보관됩니다. 싱글 로그아웃 요청이 완료되면 사용자의 세션 정보가 제거되어 세션이 무효화됩니다.

  아이덴티티 공급자와 서비스 공급자에 영구 세션이 필요합니다.

- 사인아웃(WS-페더레이션)

  사인아웃이 사용되도록 설정된 경우 사용자 컨텍스트 정보가 세션 저장소에 저장됩니다. 이 정보를 통해 정책 서버가 사인아웃 요청을 생성할 수 있습니다. 사인아웃 요청이 완료되면 사용자의 세션 정보가 제거되면서 사용자 세션이 무효화됩니다.

  아이덴티티 공급자와 리소스 파트너에 영구 세션이 필요합니다.

- 인증 세션 변수 유지(모든 프로필)

  신뢰 당사자 측에서 페더레이션을 구성할 때 "인증 세션 변수 유지" 옵션을 선택할 수 있습니다. 이 옵션은 정책 서버에 인증 컨텍스트 데이터를 세션 저장소에 세션 변수로 저장하도록 지시합니다. 정책 서버는 이러한 변수에 액세스하여 인증 결정에 사용할 수 있습니다.

- 어설션 특성 유지(모든 프로필)

  신뢰 당사자 측에서 "특성 유지"를 리디렉션 모드로 선택할 수 있습니다. 리디렉션 모드는 사용자를 대상 응용 프로그램으로 리디렉션하는 방법을 결정합니다. 이 모드에서는 정책 서버가 어설션 특성을 세션 저장소에 저장하여 이 특성을 HTTP 헤더 변수로 제공할 수 있도록 합니다.

- 인증 요청 POST 바인딩(SAML 2.0)

  IdP 가 HTTP-POST 바인딩을 사용하여 전달된 인증 요청을 처리할 수 있기 위해서는 IdP 가 세션 저장소에 요청을 저장해야 합니다.

세션 저장소가 이러한 유형의 사용자 세션, 어설션, 만료 데이터를 저장할 수 있도록 합니다.

세션 저장소가 사용되도록 설정

싱글 사인온, 싱글 로그아웃을 위해 SAML 아티팩트를 사용하고 정책의 단일 사용을 가능하게 할 때는 데이터를 저장하는 세션 저장소가 사용되도록 설정하십시오.

정책 서버 관리 콘솔에서 세션 저장소가 사용되도록 설정하십시오.

다음 단계를 수행하십시오.

1. 정책 서버 관리 콘솔에 로그인합니다.
2. "데이터" 탭을 선택합니다.

3. "데이터베이스" 필드의 드롭다운 목록에서 "세션 저장소"를 선택합니다.
4. "저장소" 필드의 드롭다운 목록에서 사용 가능한 저장소 유형을 선택합니다.
5. "세션 저장소 사용" 확인란을 선택합니다.

   하나 이상의 영역에서 영구 세션을 사용하려면 세션 서버가 사용되도록 설정하십시오. 사용되도록 설정된 경우 세션 서버는 정책 서버 성능에 영향을 줍니다.

   참고: "정책 저장소 데이터베이스 사용" 옵션은 사용되지 않도록 설정됩니다. 성능상의 이유로, 세션 서버와 정책 저장소를 동일한 데이터베이스에서 실행할 수 없습니다.
6. 선택한 저장소 유형에 적절한 데이터 원본 정보를 지정합니다.
7. "확인"을 클릭하여 설정을 저장하고 콘솔을 종료합니다.
8. 정책 서버를 중지했다가 다시 시작합니다.

공유 세션 저장소가 필요한 환경

다음 기능을 사용하려면 SAML 어설션과 사용자 세션 정보를 저장할 공유 세션 저장소가 필요합니다.

클러스터된 정책 서버 환경에서 이러한 기능을 구현하려면 환경을 다음과 같이 설정하십시오.

- HTTP-POST 단일 사용 정책을 *제외한* 모든 기능에 대해 영구 세션의 로그인 영역을 구성하십시오.

  영구 세션은 영역 구성의 일부입니다.
- HTTP-아티팩트 싱글 사인온의 경우 클러스터의 모든 정책 서버에서 생산자/아이덴티티 공급자 사이트의 세션 저장소를 공유하십시오.

  세션 저장소를 공유하면 모든 정책 서버가 각각 어설션에 대한 요청을 받을 때 어설션에 액세스할 수 있게 됩니다.

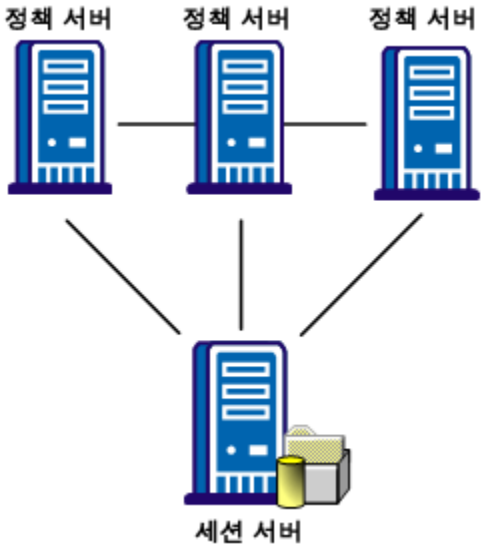
- SAML 2.0 싱글 로그아웃 및 WS-페더레이션 사인아웃의 경우 클러스터의 모든 정책 서버에서 어설션 당사자 및 신뢰 당사자의 세션 저장소를 공유합니다.

  세션 저장소를 공유하면 모든 정책 서버가 각각 세션 로그아웃에 대한 요청을 받을 때 사용자 세션 데이터에 액세스할 수 있게 됩니다.

- HTTP-POST 및 WS-페더레이션 단일 사용 정책 기능의 경우 클러스터에 있는 모든 정책 서버에서 신뢰 당사자의 세션 저장소를 공유합니다.

어설션을 생성 또는 소비하거나 영구 SMSESSION 쿠키를 처리하는 모든 정책 서버는 공용 세션 저장소에 연결할 수 있어야 합니다. 예를 들어 사용자가 example.com 에 로그인하고 해당 도메인에 대한 영구 세션 쿠키를 얻는다고 가정합니다. 이 경우 example.com 에 대한 요청을 처리하는 모든 정책 서버는 세션이 여전히 유효한지 확인할 수 있어야 합니다.

다음 그림에서는 세션 저장소 하나와 통신하는 정책 서버 클러스터를 보여 줍니다.



세션 저장소를 공유하려면 다음 방법 중 하나를 사용하십시오.

- 모든 정책 서버가 세션 저장소 하나를 가리키도록 지정

  정책 서버 관리 콘솔에서 지정된 세션 저장소가 사용되도록 정책 서버를 구성합니다.

- 세션 저장소를 여러 세션 저장소에 복제

  데이터베이스 복제에 대한 지침은 해당 데이터베이스 설명서를 참조하십시오.

제 5 장: 인증을 위한 사용자 디렉터리 연결

이 섹션은 다음 항목을 포함하고 있습니다.

사용자 디렉터리 관리 개요

디렉터리 연결은 CA SiteMinder?Federation Standalone 이 사용자 아이덴티티에 대한 컨텍스트를 설정하는 방식을 확인합니다. 시스템은 이러한 연결을 사용하여 사용자 아이덴티티를 확인하고 사용자 저장소에 포함된 사용자 특성을 검색합니다.

어설션 당사자는 사용자 디렉터리에 대해 각 사용자를 인증하여 어설션을 생성할 수 있는 사용자를 결정합니다. 신뢰 당사자 측에서는 인증 중에 사용자의 어설션이 제공될 때 신뢰 당사자가 사용자 디렉터리에서 사용자 레코드를 찾습니다.

Administrative UI 의 "사용자 디렉터리" 탭을 통해 기존 사용자 디렉터리에 대한 연결을 구성할 수 있습니다. 기존 사용자 디렉터리에 대한 연결만 설정하는 것입니다. 새로운 사용자 디렉터리를 구성하는 것이 아닙니다.

둘 이상의 디렉터리에 대한 연결을 구성할 수 있으며 디렉터리의 유형(LDAP 또는 ODBC)이 같을 필요는 없습니다.

중요! SiteMinder 커넥터를 사용하는 경우에는 SiteMinder 가 가리키는 것과 동일한 디렉터리에 연결하도록 사용자 디렉터리를 구성해야 하며 SiteMinder 가 해당 디렉터리에 사용하는 것과 동일한 이름으로 구성해야 합니다.

LDAP 디렉터리 연결

LDAP 디렉터리에 대한 연결을 설정하여 CA SiteMinder?Federation Standalone 에서 인증용 사용자 저장소로 사용할 수 있습니다.

다음 단계를 수행하십시오.

1. "사용자 디렉터리" 탭을 클릭합니다.
2. "사용자 디렉터리 목록" 섹션에서 "LDAP 에 연결"을 클릭합니다.
3. 각 섹션에서 설정을 구성합니다. 빨간색 점으로 표시된 매개 변수는 필수 매개 변수입니다.

   참고: "도움말"을 클릭하면 필드, 컨트롤 및 해당되는 각 요구 사항에 대한 설명을 볼 수 있습니다.
4. "장애 조치"/"부하 분산" 기능을 설정하려면 해당 항목을 클릭합니다.
5. "연결 테스트"를 클릭하여 디렉터리 연결이 유효한지 확인합니다.

   "콘텐츠 보기"를 클릭하여 사용자 디렉터리의 콘텐츠를 볼 수 있습니다.

   참고: "콘텐츠 보기" 단추는 "검색 루트", "사용자 DN 조회 시작", "사용자 DN 조회 끝", "유니버설 ID 특성" 값이 설정된 경우에만 표시됩니다.
6. "저장"을 클릭합니다.

   설정이 올바르면 "사용자 디렉터리 보기" 대화 상자로 리디렉션됩니다.

   LDAP 디렉터리 연결이 구성되었습니다.

LDAP 사용자 디렉터리의 부하 분산 및 장애 조치

CA SiteMinder?Federation Standalone 은 장애 조치 및 부하 분산을 위해 LDAP 사용자 디렉터리 요청을 여러 LDAP 서버에 배포할 수 있습니다.

부하 분산을 위해 시스템은 지정된 LDAP 서버로 요청을 균등하게 분산합니다. 부하 분산 기능을 장애 조치와 함께 사용하면 LDAP 사용자 디렉터리 정보에 보다 빠르고 효율적으로 액세스할 수 있습니다.

장애 조치를 위해 시스템은 LDAP 서버 하나에서 더 이상 응답이 없을 때까지 해당 LDAP 서버에서 모든 요청을 처리합니다. 기본 서버가 응답하지 않을 경우 시스템은 장애 조치가 구성된 다음 서버로 요청을 라우팅합니다. 이 프로세스는 여러 서버에서 반복될 수 있습니다. 기본 서버가 요청을 다시 처리할 수 있는 상태가 되면 요청이 원래 서버로 돌아갑니다.

다음 단계를 수행하십시오.

1. UI 에서 "사용자 디렉터리" 탭을 선택합니다.
2. 다음 작업 중 하나를 수행하십시오.
   - "LDAP 에 연결"을 선택하여 LDAP 디렉터리 연결을 생성합니다.
   - 편집하려는 기존 LDAP 항목 옆의 "작업", "수정"을 선택합니다.

   "사용자 디렉터리" 대화 상자가 열립니다.
3. 대화 상자의 "LDAP 사용자 디렉터리 구성" 섹션에서 "부하 분산 및/또는 장애 조치 구성"을 클릭합니다.

   "LDAP 서버 부하 분산 및 장애 조치" 테이블이 표시됩니다.
4. 첫 번째 "장애 조치 노드" 필드에 *ip_address*:*port* 형식으로 IP 주소와 포트 번호를 입력합니다. 장애 조치에 사용할 다른 디렉터리 서버의 주소도 나머지 필드에 추가합니다.

   참고: 장애 조치용 서버를 추가할 경우 장애 조치 디렉터리와 기본 디렉터리는 동일한 유형의 통신(SSL 또는 비 SSL)을 사용해야 합니다. 두 디렉터리는 동일한 포트 번호를 공유합니다.

   테이블에 항목이 하나만 있으면 장애 조치만 지원됩니다.
5. 부하 분산에 사용할 다른 그룹을 구성하려면 "행 추가"를 클릭하고 이전 단계와 마찬가지로 필드를 완성합니다.

   같은 서버를 부하 분산에 여러 번 추가할 수 있으며, 이렇게 하면 시스템 하나가 더 많은 요청을 처리합니다. 예를 들어 그룹에 서버 1 과 서버 2 라는 두 개의 서버가 있다고 가정합니다. 서버 1 은 고성능 서버이고 서버 2 는 그보다 성능이 낮습니다. 이 경우 서버 1 을 부하 분산 목록에 두 번 추가하면 서버 2 가 요청을 하나 처리할 때마다 서버 1 은 요청을 두 개 처리하게 됩니다.

예: 부하 분산 및 장애 조치

이 예에서는 SiteMinder 환경에 A 와 B 라는 두 개의 사용자 디렉터리가 있고 두 디렉터리가 다음 요구 사항을 충족해야 한다고 가정합니다.

- 사용자 디렉터리 A 는 사용자 디렉터리 B 로 장애 조치되고 B 와 부하를 분산해야 합니다.
- 사용자 디렉터리 B 는 사용자 디렉터리 A 로 장애 조치되고 사용자 디렉터리 A 와 부하를 분산해야 합니다.

이 구성에는 부하 분산 그룹 두 개가 필요합니다.

1. 첫 번째 부하 분산 그룹과 첫 번째 장애 조치 노드의 사용자 디렉터리 B 의 주소를 지정합니다.
2. "행 추가"를 클릭하여 부하 분산 그룹을 추가합니다.
3. 사용자 디렉터리 B 를 새 부하 분산 그룹의 첫 번째 서버로 나열합니다.
4. 사용자 디렉터리 A 를 부하 분산 그룹의 두 번째 서버로 나열합니다.

결과적으로 서로 부하를 분산하고 장애 조치 "A B"와 "B A"에 사용되는 서버가 하나씩 포함된 두 개의 부하 분산 그룹이 생성됩니다. 두 디렉터리를 모두 사용할 수 있는 경우 각 그룹의 첫 번째 디렉터리인 A 와 B 간에 부하가 분산됩니다. 사용자 디렉터리 A 를 사용할 수 없게 되면 사용자 디렉터리 B 로 장애 조치되어, 사용자 디렉터리 A 를 사용할 수 있게 될 때까지 사용자 디렉터리 B 에서 모든 요청이 처리됩니다.

SSL 을 통해 LDAP 사용자 디렉터리에 연결하는 방법

SSL 을 통해 LDAP 사용자 디렉터리에 연결하려면 인증서 데이터베이스 파일을 사용하도록 시스템을 구성해야 합니다.

아래의 섹션에 나오는 지침에 따라 SSL 을 통한 연결을 구성하십시오.

참고: CA 디렉터리에서는 이 방법으로 SSL 을 구성할 수 없습니다.

SSL 을 통한 LDAP 연결을 구성하기 전 수행할 작업

SSL 을 통한 LDAP 사용자 디렉터리 연결을 구성하기 전에 다음 사항을 검토하십시오.

- 디렉터리 서버가 SSL 이 사용되도록 설정되었는지 확인합니다.
- 데이터베이스 파일이 Netscape 데이터베이스 버전 파일 형식(cert8.db)인지 확인하십시오. 정책 서버는 Mozilla LDAP SDK 를 사용하여 LDAP 디렉터리와 통신합니다.

  중요! cert8.db 데이터베이스 파일에 인증서를 설치할 때 Microsoft Internet Explorer 를 사용하지 마십시오.

- (Active Directory) 다음 사항을 고려합니다.
  - AD 네임스페이스를 사용하여 사용자 디렉터리 연결이 구성된 경우에는 이후 항목에 설명되어 있는 SSL 프로세스가 적용되지 않습니다. AD 네임스페이스는 SSL 연결 설정 시 네이티브 Windows 인증서 리포지토리를 사용합니다. SSL 을 통해 통신하도록 AD 네임스페이스를 구성할 경우 다음과 같이 하십시오.
    - 사용자 디렉터리 연결에 대해 보안 연결이 구성되었는지 확인합니다.

Active Directory 인스턴스를 호스트하는 컴퓨터에서 루트 CA 인증서와 서버 인증서가 서비스 인증서 저장소에 추가되었는지 확인합니다.

인증서 데이터베이스 파일 만들기

인증서 데이터베이스 파일을 만들려면 정책 서버에 포함된 Mozilla Network Security Services(NSS) certutil 응용 프로그램을 사용하십시오.

참고: 다음 절차에서는 태스크를 완료하는 데 필요한 옵션 및 인수에 대해 자세히 설명합니다. NSS 유틸리티 옵션 및 인수의 전체 목록은 NSS project page (NSS 프로젝트 페이지)에서 Mozilla 설명서를 참조하십시오.

중요! Windows Server 2008 에서 SiteMinder 유틸리티 또는 실행 파일을 실행하기 전에 관리자 권한을 사용하여 명령줄 창을 여십시오. 사용하는 계정에 관리자 권한이 있는 경우에도 이 방식으로 명령줄 창을 여십시오.

다음 단계를 수행하십시오.

1. 명령 프롬프트에서 설치 bin 디렉터리로 이동합니다.

   예: C:\Program Files\CA\SiteMinder\bin

   참고: Windows 에는 네이티브 certutil 유틸리티가 있습니다. 정책 서버의 bin 디렉터리에서 작업 중인지 확인하십시오. 그렇지 않을 경우 실수로 Windows certutil 유틸리티를 실행할 수 있습니다.

2. 다음 명령을 입력합니다.

   `certutil -N -d certificate_database_directory`

   -N

   cert8.db, key3.db 및 secmod.db 인증서 데이터베이스 파일을 생성합니다.

   -d *certificate_database_directory*

   certutil 도구가 인증서 데이터베이스 파일을 생성할 디렉터리를 지정합니다.

   참고: 파일 경로에 공백이 있으면 경로를 따옴표로 묶으십시오.

   데이터베이스 키 암호화에 사용할 암호를 묻는 메시지가 표시됩니다.

3. 암호를 입력하고 확인합니다.

   필요한 다음 인증서 데이터베이스 파일이 생성됩니다.

   - cert8.db
   - key3.db
   - secmod.db

예: 인증서 데이터베이스 파일 만들기

```
certutil -N -d C:\certdatabase
```

인증서 데이터베이스에 루트 인증 기관 추가

루트 인증 기관(CA)을 추가하려면 정책 서버에 있는 Mozilla Network Security Services(NSS) certutil 응용 프로그램을 사용하십시오.

참고: 다음 절차에서는 태스크를 완료하는 데 필요한 옵션 및 인수에 대해 자세히 설명합니다. NSS 유틸리티 옵션 및 인수의 전체 목록은 NSS project page (NSS 프로젝트 페이지)에서 Mozilla 설명서를 참조하십시오.

중요! Windows Server 2008 에서 SiteMinder 유틸리티 또는 실행 파일을 실행하기 전에 관리자 권한을 사용하여 명령줄 창을 여십시오. 사용하는 계정에 관리자 권한이 있는 경우에도 이 방식으로 명령줄 창을 여십시오.

다음 단계를 수행하십시오.

1. 명령 프롬프트에서 정책 서버 설치 bin 디렉터리로 이동합니다.

   예: C:\Program Files\CA\SiteMinder\bin

   참고: Windows 에는 네이티브 certutil 유틸리티가 있습니다. NSS 유틸리티의 bin 디렉터리에서 작업 중인지 확인하십시오. 그렇지 않을 경우 실수로 Windows certutil 유틸리티를 실행할 수 있습니다.

2. 다음 명령을 실행합니다.

   ```
   certutil -A -n alias -t trust_arguments -i root_CA_path -d
   certificate_database_directory
   ```

   -A

   인증서 데이터베이스에 인증서를 추가합니다.

   -n *alias*

   인증서의 별칭을 지정합니다.

   참고: 별칭에 공백이 있는 경우 별칭을 따옴표로 묶으십시오.

-t *trust_arguments*

인증서에 적용할 트러스트 특성을 지정합니다. 세 개의 사용 가능한 트러스트 범주는 다음 순서로 표시됩니다: "SSL, 전자 메일, 개체 서명". 각 범주 위치에서 다음 특성 인수를 0 개 이상 사용할 수 있습니다.

p

유효한 피어입니다.

P

트러스트된 피어입니다. 이 인수는 p 를 내포합니다.

c

유효한 CA 입니다.

T

클라이언트 인증서를 발급하도록 트러스트된 CA 입니다. 이 인수는 c 를 내포합니다.

C

서버 인증서를 발급하도록 트러스트된 CA 입니다(SSL 만 해당). 이 인수는 c 를 내포합니다.

중요! 이 인수는 SSL 트러스트 범주에 필요합니다.

u

인증 또는 서명에 인증서를 사용할 수 있습니다.

-i *root_CA_path*

루트 CA 파일의 경로를 지정합니다. 이 경로는 인증서 이름을 포함합니다. 인증서의 유효한 확장명에는 cert, .cer, .pem 이 포함됩니다.

참고: 파일 경로에 공백이 있으면 경로를 따옴표로 묶으십시오.

-d *certificate_database_directory*

인증서 데이터베이스가 포함된 디렉터리의 경로를 지정합니다.

참고: 파일 경로에 공백이 있으면 경로를 따옴표로 묶으십시오.

예: 인증서 데이터베이스에 루트 CA 추가

```
certutil -A -n "My Root CA"  -t "C,," -i C:\certificates\cacert.cer -d C:\certdatabase
```

인증서 데이터베이스에 서버 인증서 추가

SSL 을 통한 통신을 사용하려면 인증서에 서버 인증서를 추가하십시오. 정책 서버에 있는 Mozilla Network Security Services(NSS) certutil 응용 프로그램을 사용하십시오.

참고: 다음 절차에서는 태스크를 완료하는 데 필요한 옵션 및 인수에 대해 자세히 설명합니다. NSS 유틸리티 옵션 및 인수의 전체 목록은 NSS project page (NSS 프로젝트 페이지)에서 Mozilla 설명서를 참조하십시오.

중요! Windows Server 2008 에서 SiteMinder 유틸리티 또는 실행 파일을 실행하기 전에 관리자 권한을 사용하여 명령줄 창을 여십시오. 사용하는 계정에 관리자 권한이 있는 경우에도 이 방식으로 명령줄 창을 여십시오.

다음 단계를 수행하십시오.

1. 명령 프롬프트에서 정책 서버 설치 bin 디렉터리로 이동합니다.

   예: C:\Program Files\CA\SiteMinder\bin

   참고: Windows 에는 네이티브 certutil 유틸리티가 있습니다. NSS 유틸리티의 bin 디렉터리에서 작업 중인지 확인하십시오. 그렇지 않을 경우 실수로 Windows certutil 유틸리티를 실행할 수 있습니다.

2. 다음 명령을 실행합니다.

   `certutil -A -n alias -t trust_arguments -i server_certificate_path -d certificate_database_directory`

   -A

   인증서 데이터베이스에 인증서를 추가합니다.

   -n *alias*

   인증서의 별칭을 지정합니다.

   참고: 별칭에 공백이 있는 경우 별칭을 따옴표로 묶으십시오.

-t *trust_arguments*

트러스트 인수를 지정합니다. 각 인증서에 대한 세 개의 사용 가능한 트러스트 범주는 다음 순서로 표시됩니다: "SSL, 전자 메일, 개체 서명". 각 범주 위치에서 다음 특성 인수를 0 개 이상 사용할 수 있습니다.

p

유효한 피어입니다.

P

트러스트된 피어입니다. 이 인수는 p 를 내포합니다.

중요! 이 인수는 SSL 트러스트 범주에 필요합니다.

-i *server_certificate_path*

서버 인증서의 경로를 지정합니다. 이 경로는 인증서 이름을 포함합니다. 인증서의 유효한 확장명에는 cert, .cer, .pem 이 포함됩니다.

참고: 파일 경로에 공백이 있으면 경로를 따옴표로 묶으십시오.

-d *certificate_database_directory*

인증서 데이터베이스가 포함된 디렉터리의 경로를 지정합니다.

참고: 파일 경로에 공백이 있으면 경로를 따옴표로 묶으십시오.

NSS 에서 인증서 데이터베이스에 서버 인증서를 추가합니다.

예: 인증서 데이터베이스에 서버 인증서 추가

```
certutil -A -n "My Server Certificate" -t "P,," -i C:\certificates\servercert.cer -d
C:\certdatabase
```

데이터베이스에 인증서가 있는지 확인

인증서가 인증서 데이터베이스에 있는지 확인하려면 Mozilla Network Security Services(NSS) certutil 응용 프로그램을 사용하십시오. 정책 서버에는 이 도구가 포함되어 있습니다.

참고: 다음 절차에서는 태스크를 완료하는 데 필요한 옵션 및 인수에 대해 자세히 설명합니다. NSS 유틸리티 옵션 및 인수의 전체 목록은 NSS project page (NSS 프로젝트 페이지)에서 Mozilla 설명서를 참조하십시오.

중요! Windows Server 2008 에서 SiteMinder 유틸리티 또는 실행 파일을 실행하기 전에 관리자 권한을 사용하여 명령줄 창을 여십시오. 사용하는 계정에 관리자 권한이 있는 경우에도 이 방식으로 명령줄 창을 여십시오.

다음 단계를 수행하십시오.

1. 명령 프롬프트에서 정책 서버 설치 bin 디렉터리로 이동합니다.

   예: C:\Program Files\CA\SiteMinder\bin

   참고: Windows 에는 네이티브 certutil 유틸리티가 있습니다. NSS 유틸리티의 bin 디렉터리에서 작업 중인지 확인하십시오. 그렇지 않을 경우 실수로 Windows certutil 유틸리티를 실행할 수 있습니다.

2. 다음 명령을 실행합니다.

   `certutil -L -d certificate_database_directory`

   -L

   인증서 데이터베이스에 있는 모든 인증서의 목록을 표시합니다.

   -d *certificate_database_directory*

   인증서 데이터베이스가 포함된 디렉터리의 경로를 지정합니다.

   참고: 파일 경로에 공백이 있으면 경로를 따옴표로 묶으십시오.

   이 명령은 루트 CA 별칭, 서버 인증서 별칭, 그리고 인증서를 인증서 데이터베이스에 추가할 때 지정한 트러스트 특성을 표시합니다.

예: 인증서 데이터베이스의 인증서 목록 표시

```
certutil -L -d C:\certdatabase
```

LDAP 사용자 디렉터리 연결에 SSL 설정

시스템에서 올바른 인증서 데이터베이스를 가리키도록 한 후 LDAP 사용자 디렉터리에 대해 SSL 보안 연결을 설정하십시오. SSL 은 정책 서버와 사용자 디렉터리 사이의 통신 보안을 강화합니다.

참고: 다음 절차에서는 LDAP 연결이 제대로 작동하고 있다고 가정합니다.

다음 단계를 수행하십시오.

1. Administrative UI 에 로그인합니다.
2. "사용자 디렉터리" 탭을 선택합니다.

   "사용자 디렉터리 목록"이 나타납니다.

3. SSL 을 사용할 LDAP 항목 옆의 "작업", "수정"을 클릭합니다.
4. "LDAP 사용자 디렉터리 구성" 섹션의 "서버" 필드에 SSL 연결에 필요한 올바른 서버 및 포트 값이 지정되었는지 확인합니다. 일반적으로 SSL 에는 SSL 연결을 사용하지 않을 때와 다른 포트가 사용됩니다.
5. "연결 자격 증명" 섹션에서 "보안된 연결" 확인란을 선택합니다.
6. "저장"을 클릭합니다.

   "사용자 디렉터리" 대화 상자로 돌아갑니다.
7. "사용자 디렉터리" 목록에서 SSL 을 사용하는 LDAP 항목 옆의 "작업", "연결 테스트"를 선택합니다.

   대화 상자 위쪽에 표시되는 메시지가 SSL 이 올바르게 구성되었음을 알리거나 오류를 보고합니다.

사용자 디렉터리 연결이 SSL 을 통해 통신하도록 구성됩니다.

인증서 데이터베이스에 대한 연결 설정

SSL 을 통해 LDAP 사용자 디렉터리에 연결하려면 시스템이 올바른 인증서 데이터베이스를 가리켜야 합니다. 이 데이터베이스에는 cert8.db 파일과 key3.db 파일이 들어 있어야 합니다.

제품에 포함되어 있는 XPSConfig 도구를 사용하면 LdapObjCertDbPath 설정을 사용하여 인증서 데이터베이스의 경로를 지정할 수 있습니다.

다음 단계를 수행하십시오.

1. 명령 창을 엽니다.
2. *federation_install_dir* 로 이동합니다.
3. XPSConfig 를 입력합니다. UNIX 플랫폼에서는 이 명령의 대/소문자를 구분합니다.
4. SM 를 입력합니다.
5. LdapObjCertDbPath 설정에 해당하는 숫자를 입력합니다.
6. C 를 입력하여 값을 변경합니다.
7. "새 값 입력" 프롬프트에 인증서 데이터베이스의 경로를 지정합니다.

   예:

   ```
   C:\Program Files\CA\Federation Standalone\ldaps\certdb
   ```

8. XPSConfig 가 종료될 때까지 Q 를 계속 입력합니다.

   새 값이 저장되었습니다.

이제 올바른 인증서 데이터베이스가 사용됩니다.

LDAP 디렉터리에 대한 SSL 연결 확인

SSL 연결을 확인하여 사용자 디렉터리 연결의 보안이 유지되는지 확인하십시오.

다음 단계를 수행하십시오.

1. Administrative UI 에 로그인합니다.
2. "사용자 디렉터리"를 선택합니다.

   "사용자 디렉터리" 화면이 표시됩니다. 테이블에 기존 사용자 디렉터리 연결의 이름이 나열됩니다.
3. 테스트할 사용자 디렉터리 이름 옆의 "작업", "수정"을 선택합니다.

   디렉터리 설정이 표시됩니다.
4. "콘텐츠 보기"를 클릭합니다.

   SSL 이 올바르게 구성되어 있으면 "디렉터리 콘텐츠" 화면이 나타나고 해당 사용자 디렉터리의 콘텐츠가 나열됩니다.

LDAP 사용자 디렉터리에 대한 SSL 연결 문제 해결

다음 목록은 SSL 을 사용하여 LDAP 사용자 디렉터리에 연결할 때 문제가 발생할 경우의 해결 방법을 보여 줍니다.

- 보안 연결 없이 사용자 디렉터리에 연결할 수 있는지 확인합니다.
- 사용 중인 LDAP 서버에 대해 SSL 이 설정되어 있는지 확인합니다.
- CA SiteMinder® Federation Standalone 호스트에서 LDAP 서버의 SSL 포트에 연결할 수 있는지 확인합니다.
- 시스템이 인증서 데이터베이스 파일이 들어 있는 디렉터리를 가리키는지 확인합니다.
- 인증서 데이터베이스 디렉터리에 cert8.db 및 key3.db 파일이 들어 있는지 확인합니다.
- LDAP 서버 구성(포트 포함), 연결 자격 증명 및 검색 루트가 Administrative UI 에 올바르게 구성되어 있는지 확인합니다.

ODBC 디렉터리 연결

기존 ODBC 사용자 저장소(SQL 또는 Oracle)를 대상으로 디렉터리 연결을 구성하여 CA SiteMinder?Federation Standalone 에서 인증에 사용할 수 있습니다.

참고: Solaris 에 있는 ODBC 데이터 원본에 연결하려면 데이터 원본에 대해 유선 프로토콜 드라이버를 구성하십시오. 자세한 내용은 유선 프로토콜 드라이버 지침을 참조하십시오.

다음 단계를 수행하십시오.

1. "사용자 디렉터리" 탭을 클릭합니다.
2. "사용자 디렉터리 목록" 섹션에서 "ODBC 에 연결"을 클릭합니다.

3. 이 대화 상자의 설정을 구성합니다. 빨간색 점으로 표시된 매개 변수는 필수 매개 변수입니다.

   참고: "도움말"을 클릭하면 필드, 컨트롤 및 해당되는 각 요구 사항에 대한 설명을 볼 수 있습니다.

4. 중복 모드를 위해 추가적인 ODBC 디렉터리를 설정하려면 "장애 조치"를 클릭합니다.

5. "연결 테스트"를 클릭하여 연결 유효성을 검사합니다.

   "콘텐츠 보기"를 클릭하여 사용자 디렉터리의 콘텐츠를 볼 수 있습니다.

   참고: "콘텐츠 보기" 단추는 "유니버설 ID 열" 값이 설정된 경우에만 표시됩니다.

6. "저장"을 클릭합니다.

   설정이 올바르면 "사용자 디렉터리 보기" 대화 상자로 리디렉션됩니다.

   ODBC 디렉터리 연결이 구성되었습니다.

추가 정보:

Solaris 에서 ODBC 데이터 원본 구성 요구 사항 (페이지 103)
Oracle 유선 프로토콜 드라이버 구성 (페이지 103)
SQL Server 유선 프로토콜 드라이버 구성 (페이지 104)

ODBC 디렉터리 장애 조치 구성

CA SiteMinder?Federation Standalone 은 장애 조치를 위해 ODBC 사용자 디렉터리 요청을 여러 데이터 원본 서버에 분산시킬 수 있습니다.

참고: ODBC 사용자 디렉터리에 대한 부하 분산은 CA SiteMinder® Federation Standalone 에서 지원되지 않습니다.

장애 조치의 경우 CA SiteMinder?Federation Standalone 은 저장소가 상주하는 서버가 더 이상 응답이 없을 때까지 ODBC 디렉터리 하나를 사용하여 요청을 처리합니다. 기본 디렉터리가 응답이 없으면 CA SiteMinder?Federation Standalone 은 장애 조치용으로 구성된 다음 저장소로 요청을 라우팅합니다. 이 프로세스는 여러 서버에서 반복될 수 있습니다. 기본 서버가 요청을 다시 처리할 수 있는 상태가 되면 CA SiteMinder?Federation Standalone 이 원래 서버로 요청을 다시 라우팅합니다.

ODBC 장애 조치를 구성하려면

1. UI 에서 "사용자 디렉터리" 탭을 선택합니다.
2. 다음 작업 중 하나를 수행하십시오.
   - "ODBC 에 연결"을 선택하여 ODBC 사용자 디렉터리 연결을 생성합니다.
   - 편집하려는 기존 ODBC 항목 옆의 "작업", "수정"을 선택합니다.

   "사용자 디렉터리" 대화 상자가 열립니다.
3. 대화 상자의 "ODBC 사용자 디렉터리 구성" 섹션에서 "장애 조치 구성"을 클릭합니다.

   "ODBC 데이터 원본 장애 조치" 테이블이 표시됩니다.
4. 첫 번째 "장애 조치 노드" 필드에 데이터 원본 이름을 입력합니다. 장애 조치에 사용할 다른 데이터 원본의 이름을 나머지 필드에 추가합니다.

   참고: 장애 조치용 서버를 추가할 경우 장애 조치 디렉터리와 기본 디렉터리는 동일한 유형의 통신(SSL 또는 비 SSL)을 사용해야 합니다. 두 디렉터리는 동일한 포트 번호를 공유합니다.

   테이블에 항목이 하나만 있으면 CA SiteMinder® Federation Standalone 은 장애 조치만 지원합니다.

예: ODBC 장애 조치

이 예에서는 SiteMinder 환경에 A 와 B 라는 두 개의 사용자 디렉터리가 있고 두 디렉터리가 다음 요구 사항을 충족해야 한다고 가정합니다.

- 사용자 디렉터리 A 는 사용자 디렉터리 B 로 장애 조치되어야 합니다.
- 사용자 디렉터리 B 는 사용자 디렉터리 A 로 장애 조치되어야 합니다.

이 구성에는 두 개의 장애 조치 노드, 즉 사용자 디렉터리 A 의 데이터 원본 이름과 사용자 디렉터리 B 의 데이터 원본 이름이 필요합니다.

Solaris 에서 ODBC 데이터 원본 구성 요구 사항

UNIX 시스템의 ODBC 데이터 원본을 사용자 디렉터리로 사용하는 경우 system_odbc.ini 파일에서 데이터 원본을 구성하십시오.

system_odbc.ini 파일은 *federation_install_dir*/siteminder/db 폴더에 위치하며, 사용 가능한 모든 데이터 원본의 이름을 포함합니다. 그뿐만 아니라 이 파일에는 이러한 데이터 원본과 관련된 특성도 들어 있습니다. 첫 번째 특성은 CA SiteMinder?Federation Standalone 에 할당된 ODBC 드라이버입니다. 나머지 특성은 드라이버마다 다릅니다.

새로운 데이터 원본을 구성하기 위해 이 파일을 업데이트할 경우 해당 데이터 원본을 설명하는 새로운 섹션을 추가해야 합니다. 이때 [CA FedManager 데이터 원본]이라는 섹션 바로 다음에 SQL Server 또는 Oracle 드라이버 항목을 추가합니다. 원래 텍스트는 수정하면 안 됩니다.

Oracle 유선 프로토콜 드라이버 구성

Oracle 유선 프로토콜 드라이버를 구성하여 CA SiteMinder?Federation Standalone 이 데이터 원본에 연결하는 데 사용할 설정을 지정할 수 있습니다.

Oracle 유선 프로토콜 드라이버를 구성하려면

1. *federation_install_dir*/siteminder/db 디렉터리로 이동합니다.
2. 텍스트 편집기에서 system_odbc.ini 파일을 엽니다.
3. [CA FedManager 데이터 원본] 섹션을 선택하고 현재 위치 바로 아래쪽에 해당 섹션을 복사합니다.
4. 템플릿으로 생성한 이 복사본에서 대괄호 안의 제목을 데이터 원본에 적합한 이름으로 변경합니다.

5. LogonID, Password, HostName 및 Service Name 항목의 값을 변경합니다.

   Oracle 데이터 원본의 수정된 내용은 다음과 같이 표시되어야 합니다.

```
Driver=federation_install_dir/siteminder/odbc/lib/NSora23.so
Description=DataDirect 5.3 Oracle Wire Protocol
LogonID=uid
Password=pwd
HostName=servername
PortNumber=1521
ServiceName=servicename
CatalogOptions=0
ProcedureRetResults=0
EnableDescribeParam=0
EnableStaticCursorsForLongData=0
ApplicationUsingThreads=1
```

6. 파일을 저장합니다.

   Oracle 유선 프로토콜 드라이버가 구성됩니다.

중요! 이 파일에서 특히 [CA FedManager 데이터 원본] 섹션 아래의 내용을 포함하여 다른 항목은 수정하지 마십시오.

SQL Server 유선 프로토콜 드라이버 구성

SQL 유선 프로토콜 드라이버를 구성하여 CA SiteMinder?Federation Standalone 이 데이터베이스에 연결하는 데 사용할 설정을 지정할 수 있습니다.

SQL Server 유선 프로토콜 드라이버를 구성하려면

1. *federation_install_dir*/siteminder/db 디렉터리로 이동합니다.
2. 텍스트 편집기에서 system_odbc.ini 파일을 엽니다.
3. [CA FedManager 데이터 원본] 섹션을 선택하고 현재 위치 바로 아래쪽에 해당 섹션을 복사합니다.
4. 템플릿으로 생성한 이 복사본에서 대괄호 안의 제목을 데이터 원본에 적합한 이름으로 변경합니다.

5. SQL Server 데이터 원본에 대한 수정된 텍스트가 다음과 같이 표시되도록 값을 변경하고 항목을 새로 추가합니다.

```
Driver=federation_install_dir/siteminder/odbc/lib/NSmass23.so
Description=DataDirect 5.0 SQL Server Wire Protocol
Database=database_instance
Address=host_IP_address, port_number (default: 1433)
QuotedId=No
AnsiNPW=No
```

6. 파일을 저장합니다.

   유선 프로토콜 드라이버가 구성되었습니다.

중요! 이 파일에서 특히 [CA FedManager 데이터 원본] 섹션 아래의 내용을 포함하여 다른 설정은 수정하지 마십시오.

"디렉터리" 목록에서 사용자 디렉터리 연결을 테스트합니다.

사용자 디렉터리에 대한 연결을 테스트할 수 있습니다.

사용자 디렉터리 연결을 테스트하려면

1. "사용자 디렉터리" 탭을 클릭합니다.

   "사용자 디렉터리 보기" 목록이 표시됩니다.

2. 목록에서 테스트할 항목 옆에 있는 "작업" 드롭다운 메뉴에서 "연결 테스트"를 선택합니다.

대화 상자 위쪽에 연결 확인 메시지 또는 오류 메시지가 표시됩니다.

참고: "사용자 디렉터리 만들기" 또는 "사용자 디렉터리 수정" 대화 상자의 "연결 자격 증명" 섹션에서 "연결 테스트"를 클릭하여 연결을 테스트할 수도 있습니다.

디렉터리 간에 동일한 사용자 정보의 공통 보기 생성

디렉터리 연결은 CA SiteMinder?Federation Standalone 이 사용자 아이덴티티에 대한 컨텍스트를 설정하는 방식을 확인합니다. 어설션 당사자는 사용자 디렉터리에 대해 각 사용자를 인증하여 어설션을 생성할 수 있는 사용자를 결정합니다.

페더레이션 환경 내의 여러 사용자 디렉터리에는 일반적으로 동일한 유형의 사용자 정보가 저장되지만 디렉터리마다 정보를 식별하는 데 사용하는 사용자 특성 이름 및 기본 스키마가 서로 다릅니다. 따라서 CA SiteMinder?Federation Standalone 에서도 동일한 사용자 정보에 대해 서로 다른 보기를 수신합니다. 예를 들어 LDAP 디렉터리에서는 **uid** 라는 특성을 사용하여 사용자 이름을 나타내는 반면 ODBC 디렉터리에서는 동일한 정보에 대해 **name** 특성을 사용할 수 있습니다.

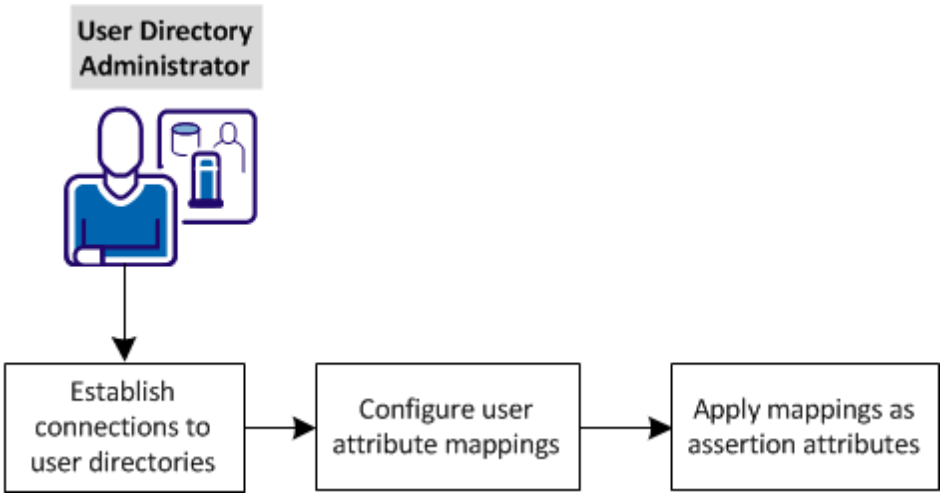
사용자 특성 매핑의 목적은 유니버설 스키마를 정의하여 동일한 정보의 공통 보기를 생성하는 것입니다. 유니버설 스키마를 통해 여러 사용자 디렉터리에서 사용자 정보를 확인할 수 있습니다. 이를 통해 시스템에서 디렉터리 유형에 관계없이 사용자 특성을 참조할 수 있기 때문에 여러 사용자 디렉터리를 사용할 경우에 필요한 구성 개체 수가 크게 줄어듭니다.

각 사용자 특성 매핑은 해당 사용자 특성 매핑이 정의된 사용자 디렉터리에 대해 적용됩니다.

사용자 디렉터리로의 연결을 구성한 후 하나의 일반 이름으로 서로 다른 사용자 디렉터리에 있는 동일한 정보를 참조할 수 있습니다.

유니버설 스키마를 생성하는 기능을 *사용자 특성 매핑*이라고 합니다. 이 기능은 Administrative UI 의 사용자 디렉터리 구성 내에서 구성하십시오.

다음 그림에서는 어설션 당사자 측에서 사용자 특성 매핑을 구성하는 프로세스를 보여 줍니다.



사용자 특성 매핑을 위해 어설션 당사자 측에서 다음 태스크를 완료하십시오.

1. 사용자 디렉터리에 대한 연결을 설정합니다 (페이지 107).
2. 사용자 특성 매핑을 구성합니다 (페이지 108).
3. 매핑을 어설션 특성으로 적용합니다 (페이지 124).

사용자 디렉터리에 대한 연결 설정

사용자 특성 매핑을 설정하려면 먼저 사용자 레코드가 저장되어 있는 사용자 디렉터리에 대한 연결을 설정해야 합니다.

제품이 연결할 수 있는 두 가지 유형의 디렉터리는 LDAP 와 ODBC 입니다.

다음 단계를 수행하십시오.

1. "사용자 디렉터리" 탭을 클릭합니다.
2. "LDAP 에 연결" 또는 "ODBC 에 연결"을 클릭합니다.
3. 각 섹션에서 설정을 구성합니다. 빨간색 점으로 표시된 매개 변수는 필수 매개 변수입니다.

   참고: "도움말"을 클릭하면 필드, 컨트롤 및 해당되는 각 요구 사항에 대한 설명을 볼 수 있습니다.
4. "장애 조치" 또는 "부하 분산" 기능을 설정하려면 해당 항목을 클릭합니다.
5. "연결 테스트"를 클릭하여 연결이 올바른지 확인합니다.

   "콘텐츠 보기"를 클릭하여 사용자 디렉터리의 콘텐츠를 볼 수 있습니다.

   참고:

   - LDAP 디렉터리 연결의 경우 "콘텐츠 보기" 단추는 "검색 루트", "사용자 DN 조회 시작", "사용자 DN 조회 끝", "유니버설 ID 특성" 값이 설정된 경우에만 표시됩니다.
   - ODBC 디렉터리 연결의 경우 "콘텐츠 보기" 단추는 "유니버설 ID 열" 값이 설정된 경우에만 표시됩니다.
6. "저장"을 클릭합니다.

   설정이 올바르면 "사용자 디렉터리 보기" 대화 상자로 리디렉션됩니다.

   디렉터리 연결이 구성되었습니다.

사용자 특성 매핑 구성

다음의 매핑 유형을 하나 이상 사용하여 특성 매핑을 정의하십시오.

- 별칭
- 그룹 이름
- 마스크
- 상수
- 식

다음 표에는 매핑 정의에 입력할 수 있는 데이터 유형이 나열되어 있습니다. 배포 환경의 사용자 디렉터리 각각에 대해 개별 매핑을 정의하십시오.

매핑 유형	일반 이름 매핑 대상	데이터 형식	액세스
별칭	디렉터리 내의 사용자 특성 이름	문자열, 숫자, 부울	읽기/쓰기
그룹 이름	사용자가 특정 그룹에 속해 있는지 여부를 나타내는 특성	부울	읽기/쓰기
마스크	비트 패턴을 저장하는 사용자 특성	부울	읽기/쓰기
상수	디렉터리의 모든 사용자에게 동일하거나 *일정한* 값	문자열, 숫자, 부울	읽기
식	식 구문 정보 전체를 보려면 *SiteMinder 정책 서버 구성 안내서*에서 특성 및 식 참조 부록을 참조하십시오.이 안내서는 SiteMinder 북셸프*의 일부입니다.*	문자열, 숫자, 부울	읽기

각 매핑 유형의 구성 절차는 기본적으로 같습니다. 각 매핑 유형별 사용 사례에서 구현 예를 참조하십시오.

다음 단계를 수행하십시오.

1. Administrative UI 에서 "사용자 디렉터리" 탭으로 이동합니다.
2. "사용자 디렉터리 목록"에서 *Connect to*(연결) 옵션 중 하나를 선택합니다.
3. 사용자 디렉터리 연결이 구성되어 있는지 확인하거나 새로 구성합니다.

4. "Directory Mapping Attribute"(디렉터리 매핑 특성) 섹션으로 스크롤하여 "매핑 만들기"를 선택합니다.
5. "일반" 필드를 완료합니다.

   이름

   이 매핑의 일반 이름을 지정합니다. 일반 이름은 사용자 특성 이름 규칙을 동일하게 준수해야 합니다.

   설명

   특성 매핑에 대한 설명을 입력합니다.
6. "속성" 필드를 완료합니다.

   매핑 유형

   구성할 매핑 유형을 선택합니다.

   정의

   적절한 구문을 사용하여 매핑 정의를 입력합니다. 앞에 나온 표를 참조하십시오.
7. (선택 사항) "사용 안 함"을 선택하여 특성 매핑이 사용되지 않도록 설정합니다.
8. "저장"을 클릭합니다.

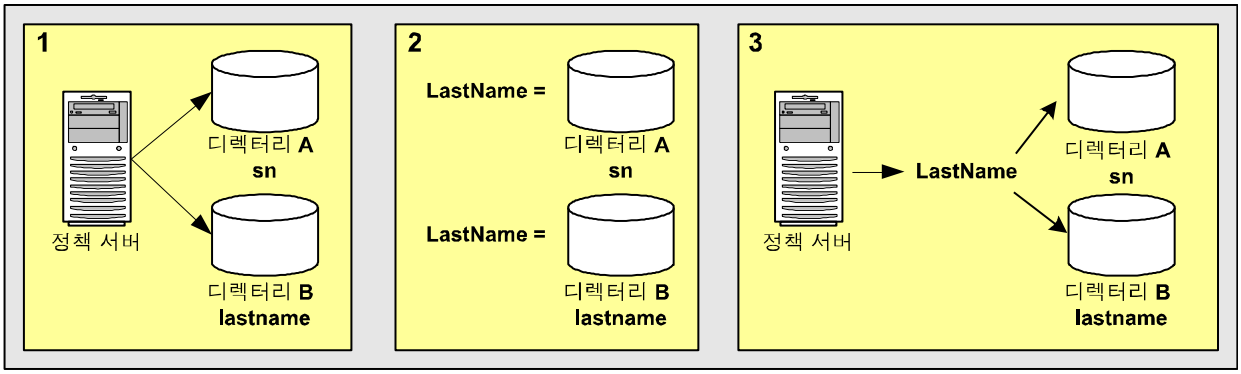
새 특성 매핑이 제출된 후 "특성 매핑 목록" 표의 목록에 추가됩니다.

별칭 특성 사용 사례

이 사용 사례에서는 사용자의 성을 식별하지만 각각 기본 스키마가 서로 다른 두 LDAP 사용자 디렉터리를 보여 줍니다.

참고: 다양한 특성 매핑 유형을 사용하여 서로 다른 디렉터리 유형 간에 동일한 사용자 특성을 식별하는 방법을 자세히 보여 주는 고급 사용자 특성 매핑 예를 검토하십시오.

다음 그림에서는 별칭 특성 매핑을 두 개 사용하여 동일한 사용자 정보의 공통 보기를 생성하는 방법을 보여 줍니다.



1. 두 사용자 디렉터리에서 사용자의 성은 서로 다르게 식별됩니다.
   - 디렉터리 A 는 sn 을 사용하여 사용자의 성을 식별합니다.
   - 디렉터리 B 는 lastname 을 사용하여 사용자의 성을 식별합니다.

   따라서 동일한 사용자 정보가 서로 다른 방식으로 표시됩니다.
2. LastName 은 기본 디렉터리 스키마에 매핑된 일반 이름 또는 *별칭*입니다.
   - 디렉터리 A 에서는 LastName 이 sn 에 매핑됩니다.
   - 디렉터리 B 에서는 LastName 이 lastname 에 매핑됩니다.

LastName 를 사용하면 동일한 사용자 정보의 공통 보기가 생성됩니다. 성(이름)을 사용하는 어설션 특성 또는 NameID 특성을 정의할 때 LastName 을 사용하십시오. 디렉터리의 작동 방식은 동일하기 때문에 시스템에서 디렉터리별 스키마는 고려할 필요가 없습니다.

추가 정보:

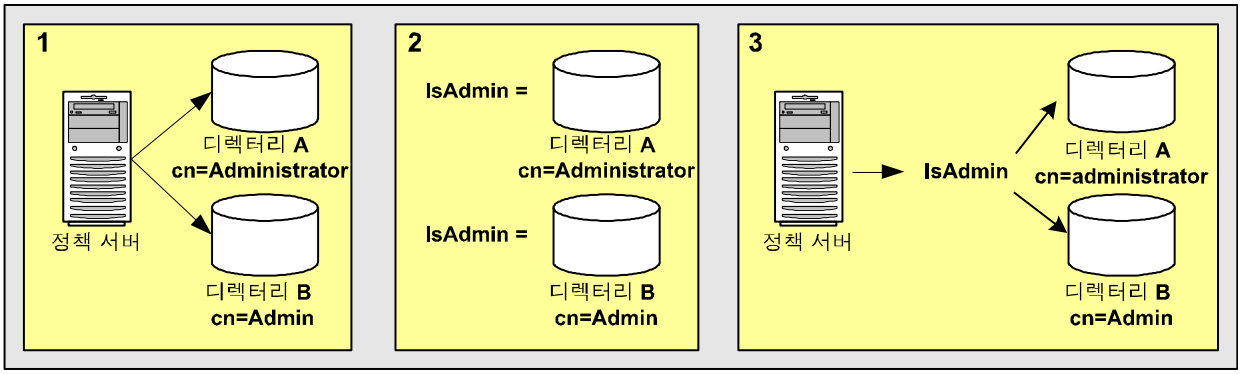
고급 사용자 특성 매핑 예 (페이지 118)

그룹 이름 사용 사례

이 사용 사례에서는 관리자 그룹에 속한 사용자를 식별하는 데 서로 다른 기본 스키마를 사용하는 두 개의 LDAP 사용자 디렉터리를 보여 줍니다.

참고: 다양한 특성 매핑 유형을 사용하여 서로 다른 디렉터리 유형 간에 동일한 사용자 특성을 식별하는 방법을 자세히 보여 주는 고급 사용자 특성 매핑 예를 검토하십시오.

다음 그림에서는 그룹 이름 특성 매핑을 두 개 사용하여 동일한 사용자 정보의 공통 보기를 생성하는 방법을 보여 줍니다.



1. 두 사용자 디렉터리는 관리자 그룹의 구성원을 서로 다르게 식별합니다.
   - 디렉터리 A 는 관리자 그룹의 구성원을 cn=Administrators,ou=groups,o=acme.com 으로 식별합니다.
   - 디렉터리 B 는 관리자 그룹의 구성원을 cn=Admin,ou=groups,o=acme.com 으로 식별합니다.

   따라서 동일한 사용자 정보가 서로 다른 방식으로 표시됩니다.
2. IsAdmin 은 기본 디렉터리 스키마에 매핑된 일반 이름입니다.
   - 디렉터리 A 에서는 IsAdmin 이 cn=Administrators,ou=groups,o=acme.com 에 매핑됩니다.
   - 디렉터리 B 에서는 IsAdmin 이 cn=Admin,ou=group,o=acme.com 에 매핑됩니다.

따라서 IsAdmin 을 사용하면 관리자 그룹의 공통 보기가 생성됩니다. 관리자 그룹에 적용되는 어설션 특성 또는 NameID 특성을 정의할 때 IsAdmin 을 참조할 수 있습니다. 디렉터리의 작동 방식은 동일하기 때문에 시스템에서 디렉터리별 스키마는 고려할 필요가 없습니다.

추가 정보:

고급 사용자 특성 매핑 예 (페이지 118)

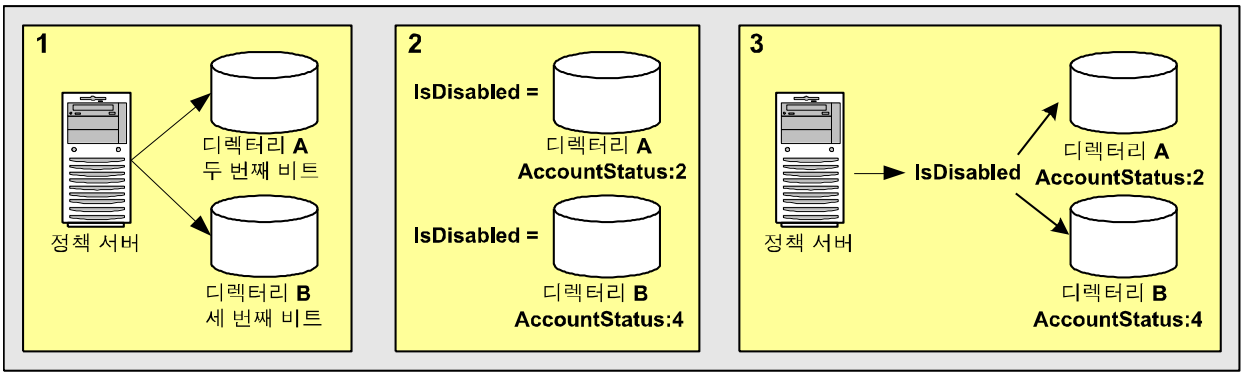
마스크 사용 사례

일부 디렉터리 구현에서는 특성의 개별 비트를 사용하여 계정 상태와 같은 특성 관련 정보를 제공합니다. 특성에 비트 마스크를 적용할 수 있습니다.

이 사용 사례에서는 비활성화된 사용자 계정을 식별하는 두 개의 Active Directory 사용자 저장소를 보여 줍니다. 각 계정에는 서로 다른 기본 스키마가 있습니다.

참고: 다양한 특성 매핑 유형을 사용하여 서로 다른 디렉터리 유형 간에 동일한 사용자 특성을 식별하는 방법을 자세히 보여 주는 고급 사용자 특성 매핑 예를 검토하십시오.

다음 그림에서는 마스크 특성 매핑을 두 개 사용하여 동일한 사용자 정보의 공통 보기를 생성하는 방법을 보여 줍니다.



1. 두 사용자 디렉터리에는 AccountStatus 라는 사용자 특성이 포함되어 있습니다. AccountStatus 는 사용자 정보를 각 비트가 플래그인 비트 패턴으로 저장합니다.
   - 디렉터리 A 에서는 두 번째 비트가 비활성화된 계정에 플래그를 설정합니다. 두 번째 비트가 1 인 경우 해당 계정은 비활성화된 것입니다.
   - 디렉터리 B 에서는 세 번째 비트가 비활성화된 계정에 플래그를 설정합니다. 세 번째 비트가 1 인 경우 해당 계정은 비활성화된 것입니다.

   따라서 동일한 사용자 정보가 서로 다른 방식으로 표시됩니다.

2. IsDisabled 는 기본 디렉터리 스키마에 매핑된 일반 이름입니다. 두 디렉터리 모두에서 IsDisabled 는 AccountStatus 에 매핑됩니다.
   - 디렉터리 A 에서는 비트 마스크 2(10 진수)가 AccountStatus 의 두 번째 비트가 설정되어 있고 계정이 비활성화되었는지 여부를 확인합니다.
   - 디렉터리 B 에서는 비트 마스크 4(10 진수)가 AccountStatus 의 세 번째 비트가 설정되어 있고 계정이 비활성화되었는지 여부를 확인합니다.

IsDisabled 를 사용하면 비활성화된 사용자 계정의 공통 보기가 생성됩니다. 사용자의 계정 상태가 필요한 어설션 특성 또는 NameID 특성을 정의할 때 IsDisabled 를 참조할 수 있습니다. 디렉터리의 작동 방식은 동일하기 때문에 시스템에서 디렉터리별 스키마는 고려할 필요가 없습니다.

추가 정보:

고급 사용자 특성 매핑 예 (페이지 118)

마스크 특성 매핑의 비트 마스크

비트 마스크 특성 매핑은 사용자 특성의 다른 비트 값을 마스크하여 하나 이상의 비트 값을 테스트합니다.

마스크 특성 매핑은 다음과 같이 정의됩니다.

```
user_attribute_name:bit_mask
```

예를 들어 사용자 특성 이름이 AccountStatus 라고 가정합니다. AccountStatus 특성은 다음 세 플래그의 상태를 비트 패턴으로 저장합니다.

비트 패턴	플래그
00?	계정 비활성화 여부
0?0	암호 만료 여부
?00	골드 구성원 여부

비트가 1 인 경우 플래그는 TRUE 입니다. 다음 표에서는 결과를 보여 줍니다.

비트 패턴	계정 상태
000 (0)	어떤 플래그도 TRUE 가 아님
001 (1)	계정 비활성화됨
010 (2)	암호 만료됨
100 (4)	골드 구성원
011 (3)	암호 만료됨, 계정 비활성화됨
101 (5)	골드 구성원, 계정 비활성화됨
110 (6)	골드 구성원, 암호 만료됨
111 (7)	골드 구성원, 암호 만료됨, 계정 비활성화됨

참고: 해당하는 10 진수 값은 괄호 안에 표시되어 있습니다.

사용자가 골드 구성원인지 여부만 테스트하려 한다고 가정합니다. 이 비트를 테스트하려면 비트 마스크로 골드 구성원에 해당하는 비트 패턴, 즉 100(이진수)을 선택하고 해당 값을 4(10 진수)로 지정하십시오. 결과 마스크 특성 매핑은 다음과 같이 정의됩니다.

```
AccountStatus:4
```

그러면 AccountStatus 에 대한 bitwise AND 연산이 비트 마스크에 대해 수행되고 결과가 비트 마스크와 같은지 여부가 테스트됩니다. 결과가 같으면 테스트된 비트의 값이 1 이고 플래그가 TRUE 임을 의미합니다. 다음 표에서는 결과를 보여 줍니다.

계정 상태	비트 마스크	bitwise AND 결과	골드 구성원 여부
000 (0)	100 (4)	000 (0)	FALSE
001 (1)	100 (4)	000 (0)	FALSE
010 (2)	100 (4)	000 (0)	FALSE
011 (3)	100 (4)	000 (0)	FALSE
100 (4)	100 (4)	100 (4)	TRUE
101 (5)	100 (4)	100 (4)	TRUE
110 (6)	100 (4)	100 (4)	TRUE

계정 상태	비트 마스크	bitwise AND 결과	골드 구성원 여부
111 (7)	100 (4)	100 (4)	TRUE

참고: 해당하는 10 진수 값은 괄호 안에 표시되어 있습니다.

비트 마스크를 사용하여 비트 집합 또는 여러 비트의 값을 한 번에 테스트할 수도 있습니다. 계정이 비활성화되어 있고 암호가 만료되었는지를 확인하려 한다고 가정합니다. 이러한 비트를 테스트하려면 비트 마스크를 011(이진수) 또는 3(10 진수)으로 지정하십시오. 결과 마스크 특성 매핑은 다음과 같이 정의됩니다.

```
AccountStatus:3
```

그러면 AccountStatus 에 대한 bitwise AND 연산이 비트 마스크에 대해 수행되고 결과가 비트 마스크와 같은지 여부가 테스트됩니다. 결과가 같으면 테스트된 두 비트의 값이 모두 1 이고 두 플래그가 모두 TRUE 임을 의미합니다. 다음 표에서는 결과를 보여 줍니다.

계정 상태	비트 마스크	bitwise AND 결과	두 플래그가 모두 설정되었는지 여부
000 (0)	011 (3)	000 (0)	FALSE
001 (1)	011 (3)	001 (1)	FALSE
010 (2)	011 (3)	010 (2)	FALSE
011 (3)	011 (3)	011 (3)	TRUE
100 (4)	011 (3)	000 (0)	FALSE
101 (5)	011 (3)	001 (1)	FALSE
110 (6)	011 (3)	010 (2)	FALSE
111 (7)	011 (3)	011 (3)	TRUE

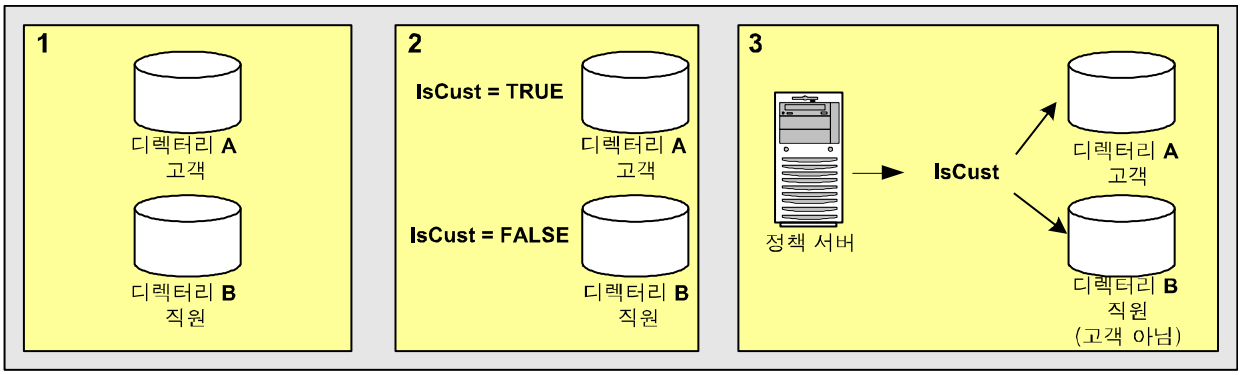
참고: 해당하는 10 진수 값은 괄호 안에 표시되어 있습니다.

상수 사용 사례

이 사용 사례에서는 한 사용자 디렉터리에는 고객만 저장되고 다른 사용자 디렉터리에는 직원만 저장되는 시나리오를 보여 줍니다.

참고: 다양한 특성 매핑 유형을 사용하여 서로 다른 디렉터리 유형 간에 동일한 사용자 특성을 식별하는 방법을 자세히 보여 주는 고급 사용자 특성 매핑 예를 검토하십시오.

다음 그림에서는 두 개의 상수 특성 매핑이 서로 다른 사용자 디렉터리에 대해 각기 다른 값을 나타내는 방식을 자세히 보여 줍니다.



1. 디렉터리 A 에는 고객만 저장됩니다. 디렉터리 B 에는 직원만 저장됩니다.
2. IsCust 는 서로 다른 디렉터리의 각기 다른 값에 매핑된 일반 이름입니다.
   - 디렉터리 A 에서는 IsCust 가 TRUE 에 매핑됩니다.
   - 디렉터리 B 에서는 IsCust 가 FALSE 에 매핑됩니다.
3. 어설션 특성 또는 NameID 특성을 정의할 때 IsCust 를 참조합니다. 일반 이름을 사용하면 시스템이 사용자가 저장된 특정 디렉터리에 관계없이 사용자가 고객인지 여부를 확인할 수 있습니다. 이 매핑은 디렉터리 A 의 모든 사용자가 고객인 반면 디렉터리 B 의 모든 사용자는 고객이 아님을 나타냅니다.

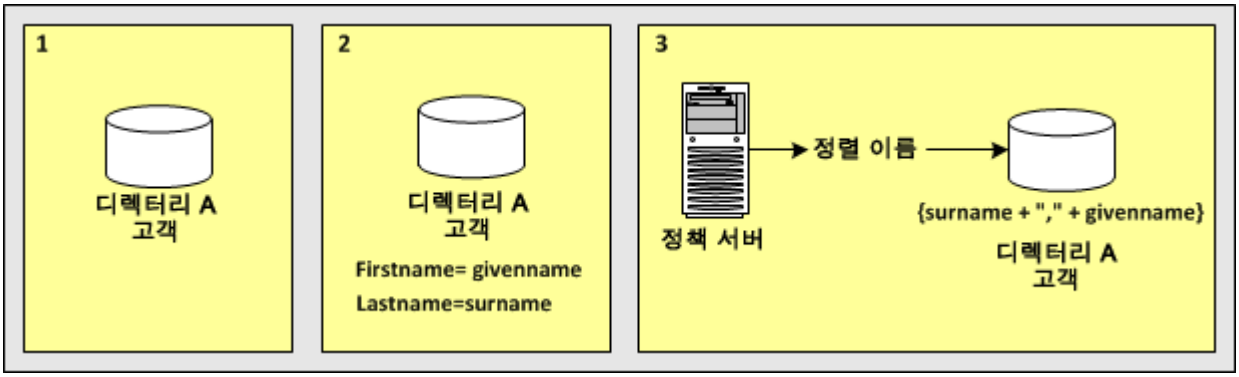
추가 정보:

고급 사용자 특성 매핑 예 (페이지 118)

식 사용 사례

이 사용 사례에서는 식 특성 매핑을 사용하여 한 디렉터리의 여러 사용자 특성에 대한 참조를 단순화하는 방법을 보여 줍니다. 보호된 리소스에는 각 사용자의 *정렬 이름*(성, 이름)이 필요합니다. 사용자 디렉터리는 이 특성을 고유하게 참조하지 않습니다. 대신 이 디렉터리에는 각 사용자의 성이 surname 으로, 각 사용자의 이름이 givenname 으로 저장됩니다.

다음 그림에서는 식 특성 매핑을 사용하여 동일한 사용자 정보의 공통 보기를 생성하는 방법을 자세히 설명합니다.



단일 사용자 디렉터리에서는 일반 이름이 디렉터리의 사용자 특성 이름을 사용하여 정렬 이름을 생성하는 식에 매핑됩니다.

- 디렉터리 A 에는 모든 사용자 레코드가 포함되어 있습니다.
- 매핑의 이름은 **SortName** 입니다.
- SortName 을 정의하는 식은 다음과 같습니다.

  ```
  {surname + "," + givenname}
  ```

  참고: 이 식은 SiteMinder 식 구문 규칙을 따라야 합니다. 구문 정보 전체를 보려면 SiteMinder 북셀프의 *SiteMinder 정책 서버 구성 안내서*에서 특성 및 식 참조 부록을 참조하십시오.
- SortName 은 surname 및 givenname 특성을 포함하는 식에 매핑된 일반 이름입니다.

디렉터리별 스키마에 관계없이 사용자의 정렬 이름이 필요한 어설션 특성 또는 NameID 특성을 정의할 때 SortName 을 참조하십시오.

추가 정보:

고급 사용자 특성 매핑 예 (페이지 118)

고급 사용자 특성 매핑 예

다음 예에서는 보다 복잡한 사용자 특성 매핑 구성을 보여 줍니다.

이 예의 배포 대상은 서로 다른 유형의 사용자 디렉터리 두 개를 사용하는 소매 의류업체입니다.

디렉터리 A

직원 전용의 내부 LDAP 사용자 디렉터리입니다.

디렉터리 B

고객 전용의 ODBC 사용자 디렉터리입니다.

각 사용자 특성 매핑은 해당 사용자 특성 매핑이 정의된 사용자 디렉터리에 대해 적용됩니다.

다음 표에서는 디렉터리 A 와 디렉터리 B 에서 동일한 사용자 정보가 식별되는 방식을 자세히 설명합니다. 함께 제공되는 사용 사례에서는 서로 다른 특성 매핑을 사용하여 동일한 사용자 정보의 공통 보기를 정의하는 방법에 대해 설명합니다. 공통 보기는 디렉터리의 작동 방식이 동일하게 만드는 유니버설 스키마의 역할을 합니다.

특성 설명	디렉터리 A 특성(LDAP)	디렉터리 B 특성(ODBC)
각 사용자의 이름	givenname	u_first_name
각 사용자의 성	surname	u_last_name
각 사용자의 정렬 이름(성, 이름)	사용자 디렉터리에는 사용자 특성이 고유하게 저장되지 않습니다.	sort_name
고객인 사용자	group:cn=customer,ou=groups,o=acme.com	사용자가 항상 고객입니다.
사용자 계정의 상태	AccountStatus 특성(플래그 집합). 두 번째 비트는 비활성화된 계정입니다.	u_disabled

별칭 매핑 유형을 사용하여 이름 특성 매핑

별칭 특성 매핑을 두 개 사용하여 디렉터리 A 와 디렉터리 B 의 이름 사용자 특성을 나타낼 수 있습니다.

배포

디렉터리 A 는 givenname 을 사용하여 사용자의 이름을 식별합니다.
디렉터리 B 는 u_first_name 을 사용하여 사용자의 이름을 식별합니다.

해결책

1. 디렉터리 A 에 대한 별칭 특성 매핑을 생성합니다.

   이름

   FirstName

   매핑 유형

   별칭

   정의

   givenname

2. 디렉터리 B 에 대한 별칭 특성 매핑을 생성합니다.

   이름

   FirstName

   매핑 유형

   별칭

   정의

   u_first_name

디렉터리 A 의 사용자를 참조할 때는 FirstName 이 givenname 에 매핑됩니다.
디렉터리 B 의 사용자를 참조할 때는 FirstName 이 u_first_name 에 매핑됩니다.

별칭 매핑 유형을 사용하여 성 특성 매핑

별칭 특성 매핑을 두 개 사용하여 디렉터리 A 와 디렉터리 B 의 성 사용자 특성을 나타낼 수 있습니다.

배포

사용자 디렉터리 A 는 surname 을 사용하여 사용자의 성을 식별합니다. 디렉터리 B 는 u_last_name 을 사용하여 사용자의 성을 식별합니다.

해결책

1. 디렉터리 A 에 대한 별칭 특성 매핑을 생성합니다.

   이름

   LastName

   매핑 유형

   별칭

   정의

   surname

2. 디렉터리 B 에 대한 별칭 특성 매핑을 생성합니다.

   이름

   LastName

   매핑 유형

   별칭

   정의

   u_last_name

디렉터리 A 의 사용자를 참조할 때는 공통 보기에서 사용자의 성이 surname 으로 식별되는지 확인합니다. 디렉터리 B 의 사용자를 참조할 때는 공통 보기에서 사용자의 성이 u_last_name 으로 식별되는지 확인합니다.

식 및 별칭 매핑 유형을 사용하여 정렬 이름 특성 매핑

디렉터리 A 와 디렉터리 B 에서 사용자의 정렬 이름을 나타내려면 식 특성 매핑과 별칭 특성 매핑을 사용하십시오.

배포

- 디렉터리 A 는 각 사용자의 정렬 이름을 고유하게 식별하지 않습니다. 디렉터리 A 에는 각 사용자에 대해 이름이 givenname 으로, 성이 surname 으로 저장됩니다.
- 디렉터리 B 는 sort_name 을 사용하여 정렬 이름을 식별합니다.

해결책

1. 디렉터리 A 에 대한 식 특성 매핑을 생성합니다.

   이름

   SortName

   매핑 유형

   식

   정의

   ```
   (surname + "," + givenname)
   ```

   참고: 식은 식 구문 규칙을 따라야 합니다.

2. 디렉터리 B 에 대한 별칭 특성 매핑을 생성합니다.

   이름

   SortName

   매핑 유형

   별칭

   정의

   sort_name

디렉터리 A 의 사용자를 참조할 때는 정렬 이름이 지정된 식을 기반으로 계산됩니다. 디렉터리 B 의 사용자를 참조할 때는 정렬 이름이 sort_name 특성으로 표시됩니다.

그룹 및 상수 매핑 유형을 사용하여 고객 매핑

디렉터리 A 와 디렉터리 B 의 고객을 식별하려면 그룹 및 상수 특성 매핑을 사용하십시오.

배포

- 디렉터리 A 에는 직원이 저장됩니다. 회사의 직원은 고객일 수도 있으므로 디렉터리 A 는 고객을 다음 그룹에 속해 있는 직원으로 식별합니다.

  `cn=Customers,ou=Groups,o=acme.com`

- 디렉터리 B 에는 고객만 저장됩니다. 디렉터리 B 에는 고객을 식별하는 사용자 특성이 없습니다. 디렉터리 B 에 사용자를 저장한다는 것은 사용자가 곧 고객임을 의미합니다.

해결책

1. 디렉터리 A 에 대한 그룹 특성 매핑을 생성합니다.

   이름

   IsCustomer

   매핑 유형

   Group

   정의

   `cn=Customers,ou=Groups,o=acme.com`

2. 디렉터리 B 에 대한 상수 특성 매핑을 생성합니다.

   이름

   IsCustomer

   매핑 유형

   상수

   정의

   TRUE

디렉터리 A 를 참조할 때는 cn=Customers,ou=Groups,o=acme.com 에 속한 사용자가 고객으로 간주됩니다. 디렉터리 B 를 참조할 때는 모든 사용자가 고객입니다.

마스크 및 식 매핑 유형을 사용하여 계정 상태 매핑

디렉터리 A 와 디렉터리 B 에서 비활성화된 사용자 계정을 식별하려면 마스크 특성 매핑과 식 특성 매핑을 사용하십시오.

배포

- 디렉터리 A 는 플래그 집합인 AccountStatus 라는 사용자 특성을 사용하여 비활성화된 계정을 식별합니다. 두 번째 비트는 비활성화된 계정을 나타냅니다.
- 디렉터리 B 는 u_disabled 라는 사용자 특성을 사용하여 비활성화된 계정을 식별합니다. u_disabled 가 "y"이면 계정이 비활성화된 것입니다. u_disabled 가 "n"이면 계정이 활성 상태입니다.

해결책

1. 디렉터리 A 에 대한 마스크 특성 매핑을 생성합니다.

   이름

   IsDisabled

   매핑 유형

   마스크

   정의

   AccountStatus:2

   이 정의는 AccountStatus 에 비트 패턴이 저장되며 비트 마스크가 2(10 진수)임을 나타냅니다.

2. 디렉터리 B 에 대한 식 특성 매핑을 생성합니다.

   이름

   IsDisabled

   매핑 유형

   식

   정의

   ```
   (u_disabled = "y")
   ```

   u_disabled 는 부울 식입니다.

디렉터리 A 를 참조할 때는 비트 패턴이 사용자 비활성화 여부를 확인합니다. 디렉터리 B 를 참조할 때는 식이 사용자 비활성화 여부를 확인합니다.

어설션 특성에 매핑 적용

사용자 디렉터리의 사용자 특성 매핑을 정의한 후에는 이 사용자 특성 매핑을 어설션 당사자와 신뢰 당사자 간 파트너 관계의 어설션 구성에 추가하십시오. 이 매핑은 디렉터리 유형마다 다른 특성에 관계없이 어설션 당사자가 올바른 특성을 어설션에 포함할 수 있도록 도와줍니다.

"이름 ID" 유형은 어설션 구성에서 사용자 특성으로 사용될 수 있습니다.

다음 단계를 수행하십시오.

1. Administrative UI 에 로그온합니다.
2. "페더레이션", "파트너 관계"를 클릭합니다.
3. 로컬 어설션 당사자 파트너 관계에 대해 "작업", "수정"을 선택합니다.
4. "어설션 구성" 탭으로 이동합니다.
5. "어설션 특성" 섹션에서 "행 추가"를 클릭합니다.
6. 사용자 매핑의 데이터를 다음과 같이 필드에 입력합니다.

   어설션 특성

   어설션 특성의 이름/값 쌍에 원하는 이름을 지정합니다.

   형식

   특성 이름을 해석하는 방법을 나타내는 형식을 선택합니다.

   유형

   사용자 특성

   항상 사용자 특성 유형을 이 필드의 값으로 선택합니다.

   값

   "사용자 디렉터리" 대화 상자의 사용자 매핑 섹션에 있는 "이름" 필드의 값을 입력합니다.

   예: 매핑에 할당한 이름이 FullName 인 경우 이 필드에 FullName 을 입력합니다.

7. (선택 사항) "이름 ID" 유형이 사용자 특성이 될 수 있으므로 "이름 ID" 항목의 "값" 필드를 어설션 특성 항목의 "값" 필드와 일치시킵니다. 이렇게 하면 어설션에서 사용자를 식별하는 어설션 특성과 "이름 ID"에 동일한 사용자 특성이 사용됩니다.

8. 모든 어설션 특성에 대해 위 단계의 절차를 반복합니다.
9. "확인" 단계로 이동하고 "마침"을 클릭하여 변경 내용을 저장합니다.

제 6 장: 페더레이션 엔터티 구성

이 섹션은 다음 항목을 포함하고 있습니다.

엔터티를 생성하는 방법

페더레이션 파트너 관계의 각 파트너는 *페더레이션 엔터티*로 간주됩니다. 파트너 관계를 설정하기 전에 로컬 파트너를 나타내는 로컬 엔터티와 원격 파트너를 나타내는 원격 엔터티를 정의하십시오.

페더레이션 엔터티를 구성하는 두 가지 방법은 다음과 같습니다.

- 메타데이터를 사용하지 않고 엔터티를 생성합니다 (페이지 127).
- 메타데이터를 가져와서 엔터티를 생성합니다.

메타데이터를 사용하지 않고 엔터티 만들기

메타데이터 없이 엔터티를 생성하려면 다음 프로세스를 사용하십시오.

1. 엔터티 유형을 지정합니다.
2. 엔터티 유형에 대한 구체적 사항을 구성합니다.
3. 엔터티 구성을 확인합니다.

엔터티 유형 선택

엔터티를 구성하는 첫 번째 단계는 엔터티 유형을 설정하고 엔터티 역할을 결정하는 것입니다.

다음 단계를 수행하십시오.

1. Administrative UI 에 로그인합니다.
2. "페더레이션" 탭에서 "엔터티"를 선택합니다.

3. "엔터티 만들기"를 클릭합니다.

   참고: "도움말"을 클릭하면 필드, 컨트롤 및 해당되는 각 요구 사항에 대한 설명을 볼 수 있습니다.

4. 엔터티 위치에 대해 다음 옵션 중 하나를 선택합니다.

   로컬

   사이트에 대하여 로컬인 엔터티를 생성하는 것임을 나타냅니다.

   원격

   원격 사이트에서 파트너를 나타내는 엔터티를 구성하는 것임을 나타냅니다.

5. "새 엔터티 유형" 필드에서 특정 엔터티 유형을 지정합니다. 드롭다운 목록에 모든 옵션이 표시됩니다.

6. "다음"을 클릭하여 엔터티의 세부 사항을 구성합니다.

상세한 로컬 엔터티 구성

엔터티 유형을 지정한 후에는 엔터티의 상세 정보를 구성하십시오. 로컬 엔터티의 경우 다음 정보를 정의하십시오.

- 엔터티에 대한 ID 정보
- 서명 및 암호화 옵션
- NameID 및 특성 정보

다음 개념을 확인하십시오.

엔터티 ID 및 엔터티 이름 설정

엔터티 ID 가 원격 파트너를 나타내는 경우 이 값은 고유해야 합니다. 엔터티 ID 가 로컬 파트너를 나타내는 경우에는 동일한 시스템에서 재사용될 수 있습니다.

"엔터티 이름"은 시스템 데이터베이스의 엔터티 개체를 식별합니다. 엔터티 이름은 고유한 값이어야 합니다. 이 값은 내부용으로만 사용되고 원격 파트너는 이 값을 알지 못합니다.

참고: "엔터티 이름"은 "엔터티 ID"와 동일한 값일 수 있지만 값을 공유할 수는 없습니다.

서명 및 암호화 기능

서명 및 암호화 기능을 사용하려면 데이터베이스에 해당 키/인증서 항목이 있어야 합니다. 해당 키/인증서 항목이 없을 경우 "가져오기"를 클릭하여 로컬 시스템의 파일에서 개인 키/인증서 쌍을 가져오십시오. 트러스트된 인증서를 가져올 수도 있습니다.

참고: SAML 2.0 POST 프로필을 사용하는 경우 서명 어설션이 필요합니다.

어설션 특성 구성

어설션을 생성할 때 구체적인 어설션 특성을 포함하도록 어설션 당사자를 구성할 수 있습니다. 이러한 특성은 엔터티 수준에서 정의하는 것이 좋습니다. 이 엔터티는 파트너 관계를 위한 템플릿 역할을 하므로 엔터티에 대해 정의하는 모든 어설션 특성이 해당 파트너 관계에 전파됩니다. 엔터티 수준에서 어설션 특성을 정의하면 여러 파트너 관계에서 엔터티를 사용할 수 있는 이점이 있습니다.

파트너 관계에 대한 어설션 특성을 추가하거나 제거하려면 엔터티 수준이 아닌 파트너 관계 수준에서 이러한 수정을 하십시오.

다음 단계를 수행하십시오.

1. "엔터티 구성" 단계부터 시작합니다.
2. 구성 중인 로컬 엔터티 유형과 관련된 기능 및 서비스에 대한 모든 필수 필드에 데이터를 입력합니다.

   참고: "도움말"을 클릭하면 필드, 컨트롤 및 해당되는 각 요구 사항에 대한 설명을 볼 수 있습니다.
3. "다음"을 클릭합니다.

   "확인" 대화 상자가 표시됩니다.

상세한 원격 엔터티 구성

엔터티 유형을 지정한 후에는 엔터티의 상세 정보를 구성하십시오. 원격 엔터티 유형의 경우 다음 옵션을 정의하십시오.

- 엔터티에 대한 ID 정보
- 서명 및 암호화 옵션
- NameID 및 특성 정보

다음 개념을 확인하십시오.

엔터티 ID 및 엔터티 이름 설정

엔터티 ID 가 원격 파트너를 나타내는 경우 이 값은 고유해야 합니다. 엔터티 ID 가 로컬 파트너를 나타내는 경우에는 동일한 시스템에서 재사용될 수 있습니다.

"엔터티 이름"은 시스템 데이터베이스의 엔터티 개체를 식별합니다. 엔터티 이름은 고유한 값이어야 합니다. 이 값은 내부용으로만 사용되고 원격 파트너는 이 값을 알지 못합니다.

참고: "엔터티 이름"은 "엔터티 ID"와 동일한 값일 수 있지만 값을 공유할 수는 없습니다.

서명 및 암호화 기능

서명 및 암호화 기능을 사용하려면 데이터베이스에 해당 키/인증서 항목이 있어야 합니다. 해당 키/인증서 항목이 없을 경우 "가져오기"를 클릭하여 로컬 시스템의 파일에서 개인 키/인증서 쌍을 가져오십시오. 트러스트된 인증서를 가져올 수도 있습니다.

참고: SAML 2.0 POST 프로필을 사용하는 경우 서명 어설션이 필요합니다.

어설션 특성 구성

어설션을 생성할 때 구체적인 어설션 특성을 포함하도록 어설션 당사자를 구성할 수 있습니다. 이러한 특성은 엔터티 수준에서 정의하는 것이 좋습니다. 이 엔터티는 파트너 관계를 위한 템플릿 역할을 하므로 엔터티에 대해 정의하는 모든 어설션 특성이 해당 파트너 관계에 전파됩니다. 엔터티 수준에서 어설션 특성을 정의하면 여러 파트너 관계에서 엔터티를 사용할 수 있는 이점이 있습니다.

파트너 관계에 대한 어설션 특성을 추가하거나 제거하려면 엔터티 수준이 아닌 파트너 관계 수준에서 이러한 수정을 하십시오.

다음 단계를 수행하십시오.

1. "엔터티 구성" 단계부터 시작합니다.
2. 구성 중인 원격 엔터티 유형과 관련된 기능 및 서비스에 대한 모든 필수 필드에 데이터를 입력합니다.

   참고: "도움말"을 클릭하면 필드, 컨트롤 및 해당되는 각 요구 사항에 대한 설명을 볼 수 있습니다.
3. "다음"을 클릭합니다.

"확인" 대화 상자가 표시됩니다.

엔터티 구성 확인

엔터티 구성을 저장하기 전에 검토하십시오.

다음 단계를 수행하십시오.

1. 엔터티 대화 상자에서 설정을 검토합니다.
2. "뒤로"를 클릭하여 이 대화 상자의 모든 설정을 수정합니다.
3. 원하는 대로 구성되었으면 "마침"을 클릭합니다.

이제 새 엔터티가 구성되었습니다.

파트너 관계에서 엔터티 구성 변경

단일 파트너 관계 구성의 컨텍스트 내에서 원격 엔터티에 대한 엔터티 ID 값을 변경할 수 있습니다. 하지만 파트너 관계 수준에서 엔터티 ID 를 변경하더라도 파트너 관계가 다른 엔터티로 연결되지 않으며 원본 엔터티가 업데이트되지도 않습니다. 엔터티 수정은 엔터티에서 파트너 관계로의 단방향 전파입니다. 파트너 관계 수준에서 엔터티 ID 에 대한 변경 내용은 원본 엔터티로 전파되지 않습니다.

참고: 지정하는 엔터티 ID 는 원격 파트너가 사용하는 ID 와 일치해야 합니다.

엔터티 구성을 템플릿이라고 생각하십시오. 파트너 관계는 엔터티 템플릿을 기반으로 생성되므로 파트너 관계를 변경하더라도 원본 엔터티 템플릿은 변경되지 않습니다.

파트너 관계 내의 엔터티에 대한 자세한 내용은 파트너 관계의 엔터티 편집 (페이지 173)을 참조하십시오.

메타데이터를 가져와서 엔터티를 생성하는 방법

메타데이터 파일에서 데이터를 가져와서 페더레이션 엔터티를 생성할 수 있습니다. SAML 메타데이터를 가져오면 파트너 관계를 생성하는 데 필요한 구성 단계가 줄어듭니다.

메타데이터는 다음과 같은 방법으로 이용할 수 있습니다.

- 원격 파트너에서 데이터를 가져와서 새 원격 엔터티를 생성합니다.
- 원격 파트너에서 데이터를 가져와서 기존 원격 엔터티를 업데이트합니다.
- 로컬 엔터티에서 데이터를 가져와서 새 로컬 엔터티를 생성합니다.

  이 옵션은 다른 페더레이션 제품에서 CA SiteMinder® Federation Standalone 으로 마이그레이션할 때 유용합니다.

참고: 기존 파트너 관계 및 로컬 엔터티를 업데이트하거나 복원하기 위해 메타데이터를 가져오는 것이 지원되지 않습니다. 기존 로컬 엔터티를 업데이트하려면 엔터티를 편집하여 변경해야 할 설정을 수정하십시오. 메타데이터는 *새* 로컬 엔터티를 생성하는 용도로만 가져올 수 있습니다.

메타데이터 기반 엔터티를 생성하는 프로세스는 다음과 같습니다.

1. 새 엔터티 구성의 기반이 될 메타데이터 파일을 선택합니다.
2. 메타데이터 파일에서 엔터티 항목을 선택합니다. 파일에는 여러 개의 엔터티가 포함될 수 있지만 파일당 엔터티가 하나인 것이 좋습니다.
3. (선택 사항) 엔터티를 구성하려면 가져올 메타데이터 파일에 포함된 인증서를 선택합니다.

   이러한 인증서는 서명, 확인 또는 싱글 로그아웃 같은 다양한 페더레이션 기능에 필요합니다.
4. 엔터티 구성을 확인합니다.

이 단계에 대한 상세 정보는 다음 섹션에서 설명합니다.

메타데이터 파일 선택

메타데이터에 기반하여 페더레이션 엔터티를 생성하는 첫 번째 단계는 메타데이터 파일을 선택하는 것입니다.

다음 단계를 수행하십시오.

1. Administrative UI 에 로그인합니다.
2. "페더레이션" 탭에서 "엔터티"를 선택합니다.
3. "메타데이터 가져오기"를 클릭합니다.

   참고: "도움말"을 클릭하면 필드, 컨트롤 및 해당되는 각 요구 사항에 대한 설명을 볼 수 있습니다.
4. 엔터티 만들기에 사용할 메타데이터 파일을 찾습니다.

5. 새 로컬 또는 원격 엔터티를 생성할지 아니면 기존의 원격 엔터티를 업데이트할지 선택합니다.

   참고: 기존 파트너 관계 및 로컬 엔터티를 업데이트하기 위해 메타데이터를 가져오는 것이 지원되지 않습니다. 새 로컬 엔터티만 생성할 수 있습니다. 기존 로컬 엔터티를 업데이트하려면 엔터티를 편집하여 변경할 설정을 수정하십시오. 기존 원격 엔터티를 업데이트하거나 새 원격 엔터티를 생성할 수 있습니다.

6. "다음"을 클릭하여 파일에서 엔터티를 선택합니다.

만료된 항목이 있는 메타데이터 파일을 선택하면 UI 에서 표시하는 다음 대화 상자에 만료된 항목이 나열된 섹션이 포함됩니다. 이 만료된 항목은 선택할 수 없으며 참조용으로만 표시됩니다. 메타데이터 파일의 모든 엔터티가 만료되면 엔터티가 표시되지 않습니다. 이 경우 새 문서를 업로드해야 합니다.

가져올 엔터티 선택

이 절차에서는 엔터티를 생성할 메타데이터 파일을 이미 선택한 것으로 가정합니다. 파일에서 엔터티를 선택하십시오.

다음 단계를 수행하십시오.

1. "파일에 정의된 엔터티 선택" 대화 상자에서 새 항목의 이름을 지정합니다.

   엔터티를 생성하기 위해 로컬 가져오기를 수행하는 경우에는 파트너 관계 이름을 정의합니다.

2. 옵션 버튼을 클릭하여 엔터티를 선택합니다.

3. "다음"을 클릭합니다.

   원격 엔터티의 메타데이터를 가져오는 경우 문서에 인증서 데이터가 포함되어 있으면 "인증서 가져오기" 대화 상자가 표시됩니다.

   가져온 메타데이터 파일에 인증서 항목이 포함된 경우 이러한 항목을 가져올 수 있습니다.

인증서 가져오기

서명된 어설션을 확인하려면 메타데이터에 인증서가 포함된 경우 인증서를 가져오십시오. 메타데이터가 인증서가 포함되지 않은 경우 이 단계를 건너뛰고 확인 단계로 이동하십시오.

다음 단계를 수행하십시오.

1. "인증서 가져오기" 단계에서 가져올 메타데이터 파일의 인증서 항목을 선택합니다.

   올바르지 않은 항목이 있는 인증서 파일을 선택하면 다음 대화 상자에 만료된 항목이 나열된 섹션이 포함됩니다. 이러한 만료된 항목은 선택할 수 없습니다. 이러한 항목은 참조용으로만 표시됩니다. 파일의 모든 항목이 올바르지 않은 경우에는 가져오기 마법사가 인증서 선택 단계를 건너뜁니다.

   선택한 각 항목에 고유한 별칭을 지정합니다.

2. "다음"을 클릭합니다.

   항목 테이블을 보여 주는 "확인" 대화 상자가 표시됩니다.

   메타데이터 파일에서 인증서가 동일한 두 항목을 선택할 수 있습니다. SAML 1.1 및 WS-페더레이션 메타데이터의 경우 SAML 1.1 은 데이터를 암호화하지 않으므로 모든 항목의 인증서 용도가 "서명"으로 표시됩니다.

   SAML 2.0 의 경우 각 항목의 용도가 인증서마다 다른 용도로(예를 들어 하나는 서명용, 다른 하나는 암호용) 표시될 수 있습니다. 확인 단계로 이동하면 창에 단일 인증서 항목이 있는 테이블이 표시됩니다. 인증서 용도는 "서명 및 암호화"로 나열됩니다. 이 항목은 이전에 선택한 두 항목의 조합입니다. 이 항목은 선택한 인증서 항목에 대해 지정된 첫 번째 별칭도 사용합니다.

   이러한 상황은 두 용도 모두에 대해 동일한 인증서가 메타데이터 파일에 나열된 경우에만 발생합니다. 파일에 별도의 인증서 두 개가 포함된 경우 확인 단계에서 테이블에 두 항목이 모두 표시됩니다.

예를 들어 메타데이터 파일에서 두 항목을 선택했는데 두 항목이 동일한 인증서라는 사실을 모르는 경우가 있습니다. 첫 번째 용도는 "서명"이며 이를 별칭 **cert1** 에 할당합니다.두 번째 용도는 "암호화"이며 이를 별칭 **cert2** 에 할당합니다. 가져오기를 확인하면 "선택된 인증서 데이터"라는 제목의 테이블에 다음과 유사한 항목이 표시됩니다.

별칭	발급된 대상	사용
cert1	Jane Doe	서명 및 암호화

메타데이터 파일에 용도가 지정되지 않은 경우 용도가 기본값 "서명 및 암호화"로 설정됩니다.

3. "다음"을 클릭하여 구성을 마칩니다.

엔터티 구성 확인

엔터티 구성을 저장하기 전에 검토하십시오.

다음 단계를 수행하십시오.

1. 엔터티 대화 상자에서 설정을 검토합니다.
2. "뒤로"를 클릭하여 이 대화 상자의 모든 설정을 수정합니다.
3. 원하는 대로 구성되었으면 "마침"을 클릭합니다.

이제 새 엔터티가 구성되었습니다.

제 7 장: 키 및 인증서 관리

이 섹션은 다음 항목을 포함하고 있습니다.

인증서 및 개인 키 사용

어설션에 보안을 적용하고 어설션 내의 데이터를 암호화하는 것은 파트너 관계 구성의 중요한 부분입니다. 페더레이션 환경에서 키/인증서 쌍과 독립 실행형 인증서는 다음과 같은 다양한 기능을 합니다.

- 어설션 서명/확인(세 가지 프로필 모두)
- 인증 요청 서명/확인(SAML 2.0 만 해당)
- 싱글 로그아웃 요청 및 응답 서명/확인(SAML 2.0)
- HTTP-아티팩트 SSO 에 대한 백 채널 요청 및 응답 서명(SAML 1.1 및 2.0)
- 전체 어설션 또는 어설션의 일부 암호화/암호 해독(SAML 2.0)
- 백 채널을 통한 아티팩트 싱글 사인온의 클라이언트 자격 증명(SAML 1.1 및 2.0)

*정책 서버 구성 안내서*에는 키 및 인증서 관리에 대한 개요 정보와 지침이 포함되어 있습니다.

SSL 서버 인증서를 사용하여 다음 태스크를 수행할 수 있습니다.

- SSL 연결을 통한 페더레이션 트래픽 관리
- 백 채널을 통한 아티팩트 싱글 사인온의 보안 통신

SiteMinder 웹 에이전트가 설치된 웹 서버에 대해 SSL 이 사용되도록 설정하기 위한 지침을 참조하십시오.

참고: SSL 이 사용되도록 설정하는 경우 "기준 URL" 매개 변수를 포함한 모든 서비스의 모든 URL 이 영향을 받습니다. 즉, 모든 서비스 URL 이 https://로 시작해야 합니다.

SAML 2.0 서명 알고리즘

SAML 2.0 의 경우 서명 태스크에 대한 서명 알고리즘을 선택할 수 있습니다. 알고리즘을 선택할 수 있으므로 다음과 같은 사용 사례가 지원됩니다.

- IdP 가 RSAwithSHA1 을 사용하는 어설션, 응답 및 SLO-SOAP 메시지 또는 RSAwithSHA256 알고리즘에 서명하는 IdP-->SP 파트너 관계
- SP 가 RSAwithSHA1 을 사용하는 인증 요청과 SLO-SOAP 메시지 또는 RSAwithSHA256 알고리즘에 서명하는 SP-->IdP 파트너 관계

서명 확인은 서명된 문서에서 사용 중인 알고리즘을 자동으로 감지하고 이를 확인합니다. 따라서 서명 확인을 위한 구성은 필요하지 않습니다.

인증서 데이터 저장소 콘텐츠를 참조하기 위한 별칭

인증서 데이터 저장소의 각 키/인증서 쌍, 클라이언트 인증서 및 트러스트된 인증서마다 각각 고유한 별칭이 있어야 합니다. 별칭은 인증서 저장소의 모든 개인 키/인증서 쌍 또는 단일 인증서를 참조하는 데 사용됩니다.
인증서 데이터 저장소에는 여러 키/인증서 쌍과 단일 인증서가 저장됩니다. 페더레이션 환경에는 여러 파트너가 있습니다. 파트너가 여러 개인 경우 각 파트너마다 다른 쌍을 사용할 수 있습니다.

서명 별칭이 서명 어설션에 대해 구성된 경우 어설션 생성기는 해당 별칭과 연결된 키를 사용하여 어설션에 서명합니다. 구성된 서명이 없으면 어설션 생성기는 다음 별칭의 키를 사용하여 어설션에 서명합니다.

```
defaultenterpriseprivatekey
```

어설션 생성기가 기본 엔터프라이즈 개인 키를 찾지 못한 경우에는 저장소의 첫 번째 개인 키를 사용하여 어설션에 서명합니다.

중요! 여러 개의 키를 저장하려는 경우에는 추가하는 첫 번째 키를 다음 별칭을 사용하여 정의한 다음 이후 키를 추가하십시오.

```
defaultenterpriseprivatekey
```

지정된 정책 서버가 응답에 서명하거나, 응답에 서명하고 이를 확인합니다. 서명 및 유효성 검사를 위한 키와 인증서를 동일한 인증서 데이터 저장소에 추가하십시오.

다음과 같은 유형의 키/인증서 쌍과 단일 인증서가 인증서 데이터 저장소에 저장됩니다.

기능	개인 키/인증서 쌍	인증서 (공개 키)	CA 인증서	클라이언트 인증서
어설션, 인증 요청, SLO 요청 및 응답에 서명	X			
서명된 어설션, 인증 요청 및 SLO 요청/응답 확인		X		
어설션, 이름 ID 및 특성을 암호화 (SAML 2.0 전용)		X		
어설션, 이름 ID 및 특성을 암호 해독 (SAML 2.0)	X			
아티팩트 백 채널의 클라이언트 인증에 대한 자격 증명 역할 수행				X
다른 인증서 및 인증서 해지 목록 확인			X	
SSL 연결을 사용하여 웹 서비스 변수 확인			X	

서명 및 확인 작업

시스템은 서명 및 확인 태스크에 개인 키/인증서 쌍을 사용합니다. 개인 키/인증서 쌍은 수행 중인 트랜잭션에 따라 어설션, 어설션 응답 또는 인증 요청에 서명합니다. 모든 서명 트랜잭션 전에 어설션에 서명하는 파트너는 개인 키/인증서 쌍과 연결된 인증서(공개 키)를 파트너에 전송합니다. 이 교환은 대역외 통신의 일부로 수행됩니다. 파트너는 인증서를 사용하여 서명을 확인합니다.

트랜잭션이 발생할 때 기본적으로 어설션 당사자가 어설션에 인증서를 포함합니다. 하지만 확인 중에 파트너는 파트너가 해당 사이트에 저장하는 인증서를 사용하여 서명의 유효성을 검사합니다.

SAML 2.0 싱글 로그아웃의 경우 로그아웃을 시작하는 측이 요청에 서명하고 요청을 받는 측이 서명을 확인합니다. 이와 반대로 받는 측은 SLO 응답에 서명하고 이니시에이터가 응답을 확인합니다.

암호화 및 암호 해독 작업

SAML 2.0 의 경우 전체 어설션, NameID 또는 기타 특성을 암호화하도록 CA SiteMinder?Federation Standalone 을 구성할 수 있습니다. 암호화가 사용되도록 설정하면 어설션 당사자는 신뢰 당사자가 데이터를 암호화하기 위해 보내는 인증서(공개 키)를 사용합니다. 트랜잭션 이전에 신뢰 당사자는 대역외 교환에서 인증서를 어설션 당사자에게 보냅니다. 신뢰 당사자는 개인 키/인증서 쌍을 사용하여 데이터를 암호 해독합니다.

참고: SAML 1.1 및 WS-페더레이션은 어설션 데이터의 암호화를 지원하지 않습니다.

SSL 연결에 대한 인증서

아티팩트 백 채널에 SSL 을 사용하여 SSL 연결을 보호하고 백 채널 통신의 보안을 유지할 수 있습니다.

SSL 연결을 설정하려면 신뢰 당사자는 CA 인증서를 서명된 SSL 서버 인증서와 연결해야 합니다. SSL 서버 인증서는 SSL 연결을 보호하고, CA 인증서는 SSL 서버 인증서를 신뢰할 수 있는지 확인합니다.

아티팩트 백 채널에 보안을 적용하기 위한 인증서

아티팩트 바인딩을 사용하여 싱글 사인온을 구축하려면 신뢰 당사자는 어설션에 대한 요청을 어설션 당사자의 CA SiteMinder?Federation Standalone 에 전송합니다. 어설션 요청은 어셜션 검색 서비스(SAML 1.1) 또는 아티팩트 레졸루션 서비스(SAML 2.0)로 보내집니다. 검색 서비스는 신뢰 당사자가 제공한 아티팩트를 사용하여 어설션을 검색합니다. CA SiteMinder?Federation Standalone 은 응답을 다시 백 채널을 통해 신뢰 당사자로 보냅니다. 백 채널은 어설션 수행 측과 신뢰 당사자 사이의 보안 연결입니다. 반면에 웹 브라우저 통신은 프런트 채널을 통해 수행됩니다.

다음 인증 방법 중 하나를 사용하여 백 채널과 검색 서비스를 권한 없는 액세스로부터 보호하십시오.

- 기본
- SSL 을 통한 기본 인증
- X.509 클라이언트 인증서

  X.509 클라이언트 인증서를 인증 방법으로 사용하는 경우 신뢰 당사자는 클라이언트 인증서를 자격 증명으로 제공해야 합니다. 이 자격 증명을 통해 신뢰 당사자는 어설션을 검색하는 어설션 당사자의 서비스에 액세스할 수 있습니다.

인증 방법을 선택할 때는 다음 사항을 고려하십시오.

- 백 채널에 SSL 연결을 사용하는 방법을 고려하십시오. 트러스트된 CA 가 서명한 SSL 서버 인증서로 SSL 연결에 보안을 적용하십시오.

  공통 루트 및 중간 CA 인증서의 기본 집합이 인증서 데이터 저장소와 함께 제공됩니다. CA 가 서명한 다른 서버 인증서를 사용하려면 CA 인증서를 저장소에 트러스트된 CA 인증서로 가져옵니다.

  페더레이션은 백 채널 요청을 처리할 때 SSL-클라이언트를 사용합니다. 어설션 당사자의 웹 서버는 다음 암호화를 통해 SSL 버전 TLSV1_1 및 TSLV1_2 를 사용하도록 구성될 수 있습니다.

  - RSA_With_AES_128_CBC_SHA256
  - RSA_With_AES_256_CBC_SHA256

  이러한 암호화는 FIPS 모드와 비 FIPS 모드에서 모두 지원됩니다. SHA256 사용 여부는 SP 서버 측에서 결정됩니다. 페더레이션에는 알고리즘 선택을 위한 구성이 없습니다. 따라서 관리자는 어설션 당사자의 서버가 적절하게 구성되어 있는지 확인해야 합니다.
- 연결을 설정하기 위해 X.509 클라이언트 인증서가 필요한 경우 신뢰 당사자에 키/인증서 쌍이 있어야 하며, 그렇지 않으면 클라이언트 인증서 인증이 실패합니다. 클라이언트 인증서가 어설션 당사자의 인증서 데이터 저장소에 있는지 확인하십시오. 신뢰 당사자가 어설션에 대한 요청을 보낼 때는 클라이언트 인증서가 검색 서비스에 액세스하기 위한 신뢰 당사자 자격 증명의 역할을 합니다.

페더레이션된 트랜잭션에 사용할 키/인증서 쌍 가져오기

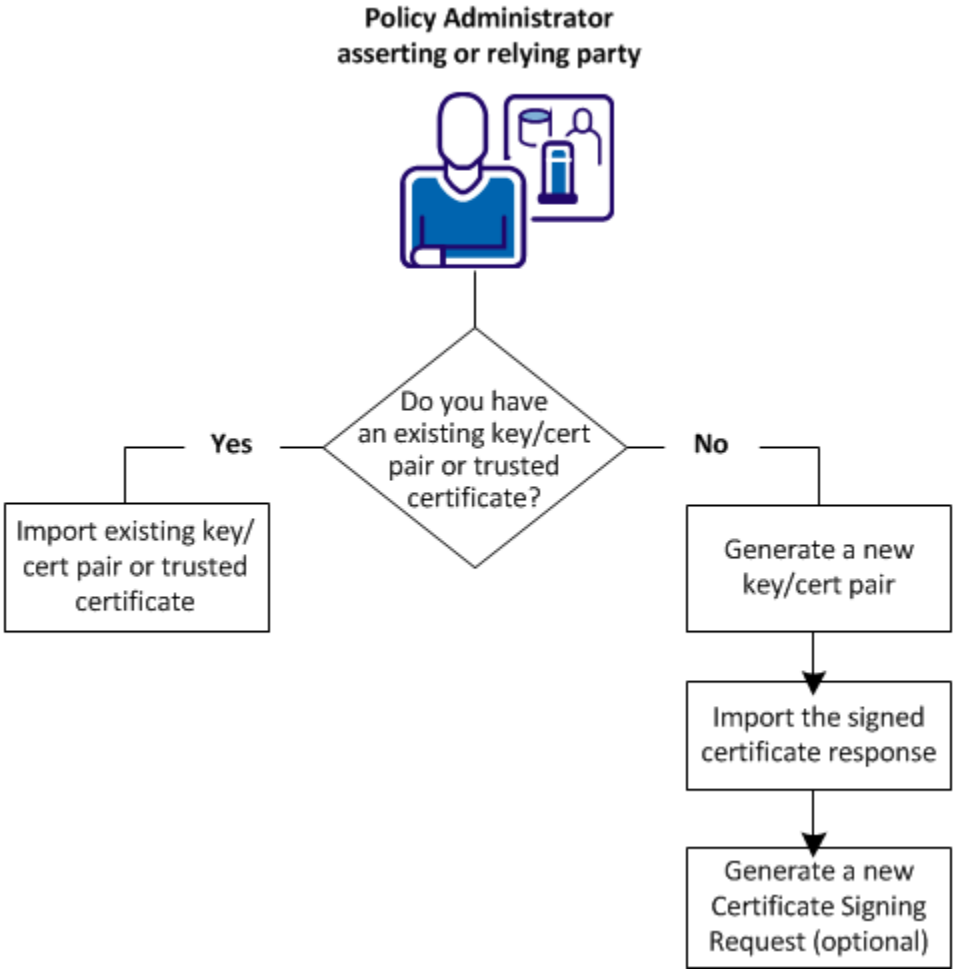
CA SiteMinder?Federation Standalone 은 여러 가지 기능에 키/인증서 쌍 및 트러스트된 인증서를 사용합니다. CA SiteMinder?Federation Standalone 이 키 및 인증서를 사용하는 태스크를 수행하려면 이러한 항목이 인증서 데이터 저장소에 있어야 합니다.

인증서 데이터 저장소에 키/인증서 쌍이 없는 경우에는 다음 두 가지 옵션을 사용할 수 있습니다.

- 기존 파일(.p12 또는 .pfx)에서 키/인증서 쌍을 가져옵니다.
- 키/인증서 쌍을 생성합니다.

  새 키/인증서 쌍을 생성하려면 트러스트된 인증 기관에 인증서를 요청한 다음 기관에서 반환한 서명된 인증서 응답을 가져옵니다.

다음 그림에서는 키/인증서 쌍 또는 트러스트된 인증서를 가져오는 각 방법의 단계를 보여 줍니다.



기존 파일에서 키/인증서 쌍 가져오기

인증서 데이터 저장소에 키/인증서 쌍이 없는 경우 기존 .p12 또는 .pfx 파일에서 가져옵니다.

CA SiteMinder?Federation Standalone 은 가져온 인증서를 트러스트된 인증서로 취급합니다. 자체 서명된 인증서는 예외입니다.

- 시스템이 V3 자체 서명 인증서를 CA 인증서로 식별하는 경우 인증서는 CA 인증서로 취급됩니다. 이 동작은 인증서/개인 키 대화 상자에서 가져오기를 시작하더라도 수행됩니다.

- CA SiteMinder® Federation Standalone 은 다음과 같은 경우 인증서를 트러스트된 인증서로 취급합니다.
  - CA SiteMinder® Federation Standalone 이 V3 자체 서명 인증서를 CA 로 식별하지 않는 경우
  - 인증서가 V1 자체 서명 인증서가 아닌 경우

다음 단계를 수행하십시오.

1. Administrative UI 에 로그인합니다.
2. "인증서 및 키" 탭에서 "인증서 및 개인 키"를 선택합니다.

   "인증서 및 개인 키 보기" 대화 상자가 열립니다.
3. "새로 가져오기"를 클릭하고 마법사의 단계를 따릅니다.

   참고: "도움말"을 클릭하면 필드, 컨트롤 및 해당되는 요구 사항에 대한 설명을 볼 수 있습니다.

   마법사를 진행할 때 다음 사항을 참조하십시오.
   - 키와 인증서가 들어 있는 단일 파일을 가져오거나, 키 파일과 인증서 파일을 개별적으로 가져올 수 있습니다. 사용하는 파일에 해당하는 옵션 단추를 선택하십시오.
   - 자체 서명된 인증서를 인증 기관 인증서로서 가져오려면 "Use as CA"(CA 로 사용) 옵션 단추를 "예"로 설정하십시오. 인증서는 CA 인증서로서 가져오기되며 파트너 관계를 구성할 때(예: 서명 또는 암호화를 위해 구성할 때)는 사용할 수 없습니다.

     그렇지 않은 경우, 기본값인 "아니요" 설정을 선택하여 파트너 관계를 구성할 때 사용 가능한 트러스트된 인증서로서 인증서를 가져오십시오.
   - DER(바이너리) 형식의 트러스트된 인증서 파일의 경우 파일에는 하나 이상의 인증서가 포함될 수 있습니다. PEM(base 64) 형식의 트러스트된 인증서 파일의 경우 CA SiteMinder® Federation Standalone 은 파일당 하나의 인증서가 있는 것으로 가정합니다.

     DER 또는 PEM 형식인 파일의 표준 확장명은 *.crt 또는 *.cer 입니다.
   - .p12 파일을 사용할 때는 암호를 입력해야 합니다. CA SiteMinder® Federation Standalone 에서는 p12 또는 .pfx 파일을 키/인증서 쌍이 포함된 파일로 처리합니다.
   - 인증서 데이터 저장소에 추가하려는 각 항목에 대해 해당 항목과 연결하려는 별칭을 입력하십시오. 여러 항목을 선택하는 경우 각 항목마다 고유한 별칭이 필요합니다.

4. "확인" 단계에서 정보를 검토하고 "마침"을 클릭합니다.

키/인증서 쌍을 인증서 데이터 저장소로 가져왔습니다.

키 및 인증서 쌍을 생성하는 방법

인증서 데이터 저장소에 키/인증서 쌍이 없는 경우에는 새 키/인증서 쌍을 생성할 수 있습니다.

다음 단계를 수행하십시오.

1. 인증서 요청을 생성하고 요청을 트러스트된 인증 기관에 보냅니다.
2. 기관으로부터 서명된 인증서 응답을 가져옵니다.

인증서 요청 생성

인증서 데이터 저장소에 키/인증서 쌍이 없는 경우에는 트러스트된 인증 기관에 새로 요청하십시오. CA 에서 서명된 인증서 응답을 반환하면 이를 인증서 데이터 저장소로 가져오십시오.

인증서 요청을 생성하면 CA SiteMinder?Federation Standalone 은 개인 키 및 자체 서명 인증서 쌍을 생성합니다. CA SiteMinder?Federation Standalone 은 이 쌍을 인증서 데이터 저장소에 저장합니다. 생성된 요청을 사용하여 인증 기관에 연락하고 생성된 요청의 내용을 양식에 붙여 넣어 CA 인증서 요청 양식을 기입하십시오.

CA 에서는 일반적으로 PKCS #7 형식으로 서명된 인증서 응답을 생성합니다. 서명된 인증서 응답을 인증서 데이터 저장소로 가져올 수 있습니다. 서명된 인증서 응답을 가져온 후에 별칭이 동일한 기존 자체 서명 인증서가 대체됩니다.

다음 단계를 수행하십시오.

1. Administrative UI 에 로그인합니다.
2. "인증서 및 키" 탭에서 "인증서 및 개인 키"를 선택합니다.

   "인증서 및 개인 키 보기" 대화 상자가 열립니다.

3. "인증서 요청"을 클릭합니다.

   "인증서 요청" 대화 상자가 열립니다.

4. 필수 필드를 기입합니다.

   참고: "도움말"을 클릭하면 필드, 컨트롤 및 해당되는 각 요구 사항에 대한 설명을 볼 수 있습니다.

5. "저장"을 클릭합니다.

PKCS #10 사양을 따르는 파일이 생성됩니다.

브라우저에서 인증서 요청이 포함된 파일을 저장할지 아니면 열지를 묻는 메시지가 표시됩니다. 이 파일을 저장하지 않더라도(또는 파일을 열고 텍스트를 추출하더라도) CA SiteMinder?Federation Standalone 은 개인 키와 자체 서명 인증서 쌍을 생성합니다. "CSR 생성" 기능을 사용하여 개인 키에 대한 새 요청 파일을 가져오려면 새 인증서 서명 요청을 생성하십시오.

서명된 인증서 응답 가져오기

인증서 요청을 작성하고 이를 인증 기관에 전송하면 인증 기관에서 서명된 인증서 응답을 발급합니다.

서명된 인증서를 인증 데이터 저장소로 가져와서 동일한 별칭을 가진 기존의 자체 서명된 인증서 항목을 이 서명된 인증서로 바꾸십시오.

다음 단계를 수행하십시오.

1. "인증서 및 키" 탭에서 "인증서 및 개인 키"를 선택합니다.

   "인증서 및 개인 키 보기" 대화 상자가 열립니다.

2. 별칭이 동일한 자체 서명된 항목을 검색합니다.

3. 자체 서명된 인증서가 들어 있는 항목 옆의 "작업", "인증서 업데이트"를 선택합니다.

   인증서 및 키 가져오기를 위한 마법사가 표시됩니다.

   참고: "도움말"을 클릭하면 필드, 컨트롤 및 해당되는 각 요구 사항에 대한 설명을 볼 수 있습니다.

4. 원하는 파일을 찾아 이동합니다. 사용 가능한 파일은 다음과 같습니다.
   - 서명된 인증서와 해당 인증서 체인이 포함된 .p7 또는 .p7b 파일
   - 인증서 체인 없이 서명된 인증서가 있는 .cer 또는 .crt 파일(base64 PEM 파일)

5. 적절한 항목을 선택합니다.
6. "확인" 단계에서 인증서 정보를 검토하고 "마침"을 클릭합니다.

서명된 인증서를 인증서 데이터 저장소로 가져오게 되고 자체 서명된 인증서가 대체됩니다.

새 인증서 서명 요청 생성

CSR(인증서 서명 요청)은 아이덴티티 인증서를 신청하기 위해 인증 기관에 보내는 메시지입니다. CA SiteMinder?Federation Standalone 에서 키/인증서 쌍을 생성해야만 CSR 을 생성할 수 있습니다. 그러면 인증서가 CSR 에 저장됩니다.

기존 개인 키에 대한 새 요청을 생성하는 이유는 다음과 같습니다.

- 개인 키/자체 서명 인증서 쌍에 대해 CA SiteMinder® Federation Standalone 에서 생성된 원래 요청이 더 이상 없는 경우
- 인증서가 만료되어 새 인증서가 필요한 경우. 이 경우에는 인증 기관에 제출하기 위한 새로운 CSR 사본이 필요합니다.

자체 서명되거나 CA 서명된 개인 키/인증서 쌍에 대한 새 CSR 를 생성할 수 있습니다. 개인 키는 항상 기존 개인 키를 수정하지 않고 동일한 CSR 를 생성합니다.

다음 단계를 수행하십시오.

1. "인증서 및 키" 탭에서 "인증서 및 개인 키"를 선택합니다.

   "인증서 및 개인 키 목록"이 표시됩니다.
2. 새 CSR 을 원하는 개인 키 항목에 대해 "작업", "CSR 생성"을 선택합니다.

   PKCS #10 사양을 따르는 파일이 생성되고 CA SiteMinder® Federation Standalone 에 CSR 를 저장하라는 메시지가 표시됩니다.
3. "저장"을 클릭합니다.
4. (선택 사항) CA 서명된 인증서가 필요한 경우 인증 기관에 연락하고 요청 제출에 필요한 인증 기관의 절차를 따르십시오. 이전 단계에서 저장한 PKCS#10 파일을 요청에 사용합니다.

인증서 요청 프로세스를 완료하면 인증 기관에서 서명된 인증서 응답을 발급합니다. 이 인증서를 인증서 데이터 저장소로 가져옵니다. 그러면 CA SiteMinder?Federation Standalone 은 별칭이 같은 기존 인증서 항목을 새로 가져온 인증서로 바꿉니다.

CRL 을 사용하여 인증서의 유효성을 확인하는 방법

CRL(인증서 해지 목록)은 인증 기관에서 해당 구독자에게 발급됩니다. 이 목록에는 무효화되거나 해지된 인증서의 일련 번호가 포함되어 있습니다. 서버에 액세스 요청이 수신되면 서버는 CRL 을 기반으로 액세스를 허용하거나 거부합니다.

CA SiteMinder?Federation Standalone 은 인증서 기능에 CRL 을 활용할 수 있습니다. CA SiteMinder?Federation Standalone 이 CRL 을 사용하려면 인증서 데이터 저장소가 현재 CRL 에 연결되어야 합니다. CA SiteMinder?Federation Standalone 이 해지된 파트너 인증서를 사용하려고 하면 오류 메시지가 표시됩니다. 레거시 페더레이션의 경우 이 오류 메시지는 SAML 어설션에 포함됩니다. 이 메시지는 인증이 실패했음을 나타냅니다.

CA SiteMinder?Federation Standalone 은 다음과 같은 CRL 기능을 지원합니다.

- 파일 기반 CRL 또는 LDAP CRL

  CA SiteMinder® Federation Standalone 은 CRL 을 인증서 데이터 저장소에 저장합니다. 파일 기반 CRL 은 Base64 또는 바이너리 형식으로 인코딩되어야 합니다. LDAP CRL 은 바이너리 형식으로 인코딩되어야 합니다. 또한 LDAP CRL 은 다음 특성 중 하나에 CRL 데이터를 포함해야 합니다.

  - certificateRevocationList;binary
  - authorityRevocationList;binary

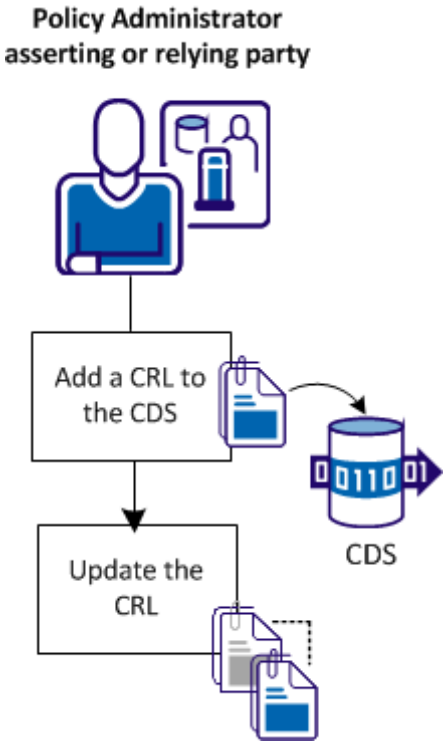
  인증 기관에서는 LDAP CRL 을 게시할 때 RFC4522 및 RFC4523 에 따라 CRL 데이터를 바이너리 형식으로 반환해야 합니다. 그렇지 않으면 CA SiteMinder® Federation Standalone 이 CRL 데이터를 사용할 수 없습니다.

- 파일 기반 CRL 의 PEM 및 DER 인코딩 형식
- LDAP CRL 의 DER 인코딩 형식

CA SiteMinder?Federation Standalone 은 CRL 을 기준으로 SSL 서버 인증서의 유효성을 검사하지 않습니다. CA SiteMinder?Federation Standalone 이 설치된 웹 서버가 SSL 서버 인증서를 관리합니다.

시스템의 각 루트 CA 마다 CRL 이 필요하지는 않습니다. 루트 CA 에 대한 CRL 이 없는 경우 CA SiteMinder?Federation Standalone 은 해당 CA 가 서명한 모든 인증서가 트러스트된 인증서인 것으로 간주합니다.

다음 그림에서는 CRL 관리 절차를 보여 줍니다.



CRL 구성 단계는 다음과 같습니다.

1. CDS 에 CRL 을 추가합니다. (페이지 151)
2. CRL 을 업데이트합니다 (페이지 152).

CDS 에 CRL 추가

인증서를 확인할 수 있는 CRL 을 사용하여 페더레이션 관련 PKI 기능에 유효한 인증서만 사용하도록 보장할 수 있습니다.

중요! CA SiteMinder® Federation Standalone 에서는 certificateRevocationList;binary LDAP 특성을 사용하여 바이너리 전송 인코딩에 LDAP CRL 을 명시적으로 요청합니다. 이는 이 특성에 CRL 데이터를 저장해야 함을 의미합니다. CA(인증 기관)가 LDAP 프로토콜을 사용하여 CRL 을 게시할 경우 RFC4522 및 RFC4523 에 따라 CRL 데이터를 바이너리 형식으로 반환해야 합니다.

CA SiteMinder?Federation Standalone 에서 CRL 을 사용하려면 CRL 위치를 지정하십시오.

다음 단계를 수행하십시오.

1. "인증서 및 키" 탭으로 이동합니다.
2. "해지 목록(CRL)"을 선택합니다.

   사용 가능한 CRL 위치 목록이 표시됩니다.
3. "추가"를 클릭합니다.

   "Add Certificate Revocation List"(인증서 해지 목록 추가)가 표시됩니다.

   참고: "도움말"을 클릭하면 필드, 컨트롤 및 해당되는 요구 사항에 대한 설명을 볼 수 있습니다.
4. CRL 발급자의 별칭과 인증서 해지 목록의 위치(URL)를 지정합니다.

   위치는 파일 CRL 의 파일 경로와 LDAP CRL 의 LDAP 검색 경로여야 합니다.
5. "저장"을 클릭합니다.

CRL 이 인증서 데이터 저장소에 추가되었습니다.

CRL 업데이트

CRL 을 업데이트하여 사용 중인 인증서 데이터가 최신 상태인지 확인하십시오.

다음 단계를 수행하십시오.

1. Administrative UI 에 로그인합니다.
2. "인증서 및 키" 탭을 선택합니다.
3. "CDS 설정"을 선택합니다.

   "인증서 설정" 대화 상자가 표시됩니다.
4. 다음 단계 중 하나를 완료합니다.
   - 저장된 CRL 파일에 NextUpdate 값이 포함되어 있지 않으면 "CRL 업데이트 간격"을 설정하여 다음 CRL 을 발급할 빈도를 지정합니다.
   - 업데이트 프로그램이 업데이트를 확인하는 빈도를 변경하려면 "CRL 업데이트 프로그램 대기 모드 간격"을 수정합니다.

   참고: "도움말"을 클릭하면 필드, 컨트롤 및 해당되는 각 요구 사항에 대한 설명을 볼 수 있습니다.
5. "CRL 업데이트 프로그램" 섹션의 "CRL 업데이트 프로그램 상태" 필드에서 "사용"을 선택합니다.
6. "저장"을 클릭합니다.

인증서 캐시 새로 고침 및 유예 기간 관리

인증서 유효성 검사(CRL 또는 OCSP)를 관리하기 위해 완료할 수 있는 다른 두 가지 태스크가 있습니다.

- 인증서 캐시 새로 고침을 수정하여 성능을 향상시킵니다.

  인증서 캐시 새로 고침 간격은 인증서 데이터 저장소가 정책 저장소의 인증서 데이터를 업데이트하는 빈도를 가리킵니다. SiteMinder 성능 향상을 위해 인증서 데이터는 메모리에 캐시됩니다. 데이터가 최신 상태가 되도록 메모리의 정보를 새로 고치십시오.

- 기본 해지 유예 기간 수정

  기본 해지 유예 기간은 인증서가 해지되는 시점부터 인증서가 무효화되는 시점까지의 지연 시간입니다. 유예 기간 동안 시스템에서는 해지된 인증서를 무효화되기 전까지 사용할 수 있습니다. 인증서가 무효화된 후에는 더 이상 활성화 상태가 아니며 CA SiteMinder® Federation Standalone 이 이를 사용할 수 없습니다.

  이러한 구성 요소를 추가할 때 CRL 또는 OCSP 응답자 유예 기간 값을 지정하지 않으면 CA SiteMinder® Federation Standalone 에 기본 유예 기간이 사용됩니다. CRL 또는 OCSP 에 대한 개별 유예 기간 설정이 이 기본 유예 기간 값보다 높은 우선 순위를 가집니다.

다음 단계를 수행하십시오.

1. Administrative UI 에 로그인합니다.
2. "인증서 및 키" 탭을 선택합니다.
3. "CDS 설정"을 선택합니다.

   "인증서 설정" 대화 상자가 표시됩니다.
4. 다음 설정을 수정할 수 있습니다.
   - 인증서 캐시 새로 고침 간격
   - 해지 유예 기간
   - LDAP 액세스 시간 만료

   참고: "도움말"을 클릭하면 필드, 컨트롤 및 해당되는 각 요구 사항에 대한 설명을 볼 수 있습니다.
5. "저장"을 클릭합니다.

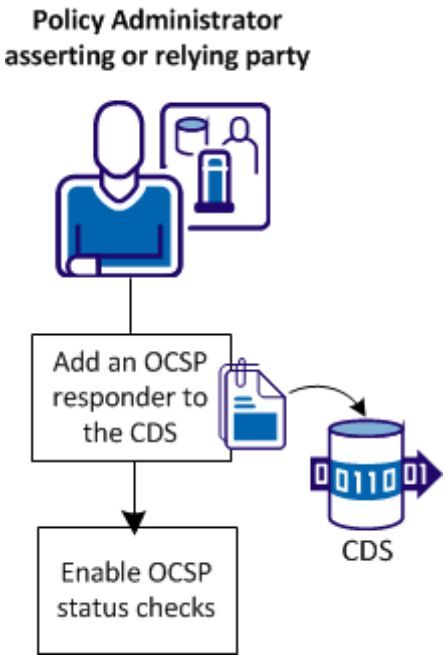
OCSP 를 사용하여 인증서 유효성을 확인하는 방법

일부 페더레이션 태스크를 수행하려면 인증서 데이터 저장소에 있는 인증서의 유효성을 검사해야 합니다. 이 태스크에는 HTTP-아티팩트 백 채널 보호, SAML 메시지 확인 및 SAML 메시지 암호화가 포함됩니다.

인증서 데이터 저장소는 인증서의 유효성을 검사하기 위해 OCSP 서비스를 사용할 수 있습니다. OCSP 에서는 CA(인증 기관)에서 제공하는 HTTP 서비스를 사용하여 필요 시 인증서 해지 상태를 제공합니다.

기본적으로 CA SiteMinder?Federation Standalone 은 인증서 데이터 저장소에서 인증서의 해지 상태를 확인하지 않습니다. OCSP 응답자를 통해 해지 상태를 확인하려면 Administrative UI 를 통해 OCSP 를 사용하도록 설정하십시오. 사용되도록 설정된 OCSP 서비스는 구성된 OCSP 응답자에 대한 해지 상태를 5 분마다 확인합니다. 이 기본 주기는 구성할 수 있습니다.

다음 그림에서는 OCSP 구성 단계를 보여 줍니다.



구성 프로세스는 다음과 같습니다.

1. CDS 에 OCSP 응답자를 추가합니다. (페이지 155)
2. OCSP 상태 확인을 사용하도록 설정합니다. (페이지 156)

OCSP 사전 요구 사항

인증서 유효성 검사에 OCSP 를 사용하려면 다음 구성 요소를 설정하십시오.

- OCSP 응답자를 설정하십시오.

- OCSP 응답자의 트러스트된 인증서를 인증서 데이터 저장소에 저장하십시오. 응답자 인증서는 CA SiteMinder® Federation Standalone 에 반환된 OCSP 응답의 서명에 대한 유효성을 검사합니다. 이 인증서는 하나의 트러스트된 확인 인증서 또는 여러 인증서의 모음입니다.

  이러한 인증서는 OCSP 트랜잭션과는 별도의 통신을 통해 CA 에서 가져올 수 있습니다.

  CA SiteMinder® Federation Standalone 은 SHA-1 및 SHA-2 알고리즘 제품군(SHA224, SHA256, SHA384, SHA512)을 사용하여 서명된 OCSP 응답을 지원합니다.

  OCSP 응답자에는 서명 유효성 검사 인증서와 응답이 포함될 수 있습니다. 그런 다음 CA SiteMinder® Federation Standalone 은 인증서 데이터 저장소에 있는 트러스트된 인증서를 기준으로 인증서와 응답 서명의 유효성을 검사합니다.

  응답에 서명 확인 인증서가 없으면 CA SiteMinder® Federation Standalone 은 인증서 데이터 저장소에 있는 인증서 또는 인증서 모음을 사용하여 서명을 확인합니다.

  OCSP 는 Administrative UI 에서 구성하며 이때 인증서 또는 인증서 모음의 위치를 지정해야 합니다.

- 사용자 인증서를 발급한 CA 인증서를 인증서 데이터 저장소에 저장하십시오. 이 CA 인증서는 사용자 인증서를 확인합니다.
- (선택 사항) CA SiteMinder® Federation Standalone 에서 OCSP 요청에 서명하는 데 사용하는 개인 키/인증서 쌍을 인증서 데이터 저장소에 저장하십시오.

CDS 에 OCSP 응답자 추가

CA SiteMinder?Federation Standalone 과 상호 작용하는 응답자 각각에 대해 OCSP 응답자 레코드를 인증서 데이터 저장소에 추가하십시오.

다음 단계를 수행하십시오.

1. Administrative UI 에 로그인합니다.
2. "인증서 및 키" 탭으로 이동합니다.
3. "OCSP 구성" 옵션을 선택합니다.

   "OCSP 구성 목록"이 표시됩니다.

4. "추가"를 클릭합니다.
5. 필드를 완성하여 "OCSP" 응답자 구성을 추가합니다.

   참고: "도움말"을 클릭하면 필드, 컨트롤 및 해당되는 각 요구 사항에 대한 설명을 볼 수 있습니다.
6. "저장"을 클릭합니다.
7. 구성하려는 OCSP 응답자 각각에 대해 이 프로세스를 반복합니다.

인증서 데이터 저장소에 OCSP 응답자 레코드가 추가되었습니다.

OCSP 상태 확인 사용

CA SiteMinder?Federation Standalone 과 상호 작용하는 응답자 각각에 대해 OCSP 응답자 레코드를 인증서 데이터 저장소에 추가하십시오.

다음 단계를 수행하십시오.

1. Administrative UI 에 로그인합니다.
2. "인증서 및 키" 탭으로 이동합니다.
3. "CDS 설정" 옵션을 선택합니다.

   "CDS 구성 목록"이 표시됩니다.
4. "OCSP 업데이트 프로그램" 섹션의 "OCSP 업데이트 프로그램 상태" 필드에서 "사용"을 선택합니다.
5. "저장"을 클릭합니다.

OCSP 상태 확인 기능이 설정되었습니다.

인증서 캐시 새로 고침 및 유예 기간 관리

인증서 유효성 검사(CRL 또는 OCSP)를 관리하기 위해 완료할 수 있는 다른 두 가지 태스크가 있습니다.

- 인증서 캐시 새로 고침을 수정하여 성능을 향상시킵니다.

  인증서 캐시 새로 고침 간격은 인증서 데이터 저장소가 정책 저장소의 인증서 데이터를 업데이트하는 빈도를 가리킵니다. SiteMinder 성능 향상을 위해 인증서 데이터는 메모리에 캐시됩니다. 데이터가 최신 상태가 되도록 메모리의 정보를 새로 고치십시오.

- 기본 해지 유예 기간 수정

  기본 해지 유예 기간은 인증서가 해지되는 시점부터 인증서가 무효화되는 시점까지의 지연 시간입니다. 유예 기간 동안 시스템에서는 해지된 인증서를 무효화되기 전까지 사용할 수 있습니다. 인증서가 무효화된 후에는 더 이상 활성화 상태가 아니며 CA SiteMinder® Federation Standalone 이 이를 사용할 수 없습니다.

  이러한 구성 요소를 추가할 때 CRL 또는 OCSP 응답자 유예 기간 값을 지정하지 않으면 CA SiteMinder® Federation Standalone 에 기본 유예 기간이 사용됩니다. CRL 또는 OCSP 에 대한 개별 유예 기간 설정이 이 기본 유예 기간 값보다 높은 우선 순위를 가집니다.

다음 단계를 수행하십시오.

1. Administrative UI 에 로그인합니다.
2. "인증서 및 키" 탭을 선택합니다.
3. "CDS 설정"을 선택합니다.

   "인증서 설정" 대화 상자가 표시됩니다.
4. 다음 설정을 수정할 수 있습니다.
   - 인증서 캐시 새로 고침 간격
   - 해지 유예 기간
   - LDAP 액세스 시간 만료

   참고: "도움말"을 클릭하면 필드, 컨트롤 및 해당되는 각 요구 사항에 대한 설명을 볼 수 있습니다.
5. "저장"을 클릭합니다.

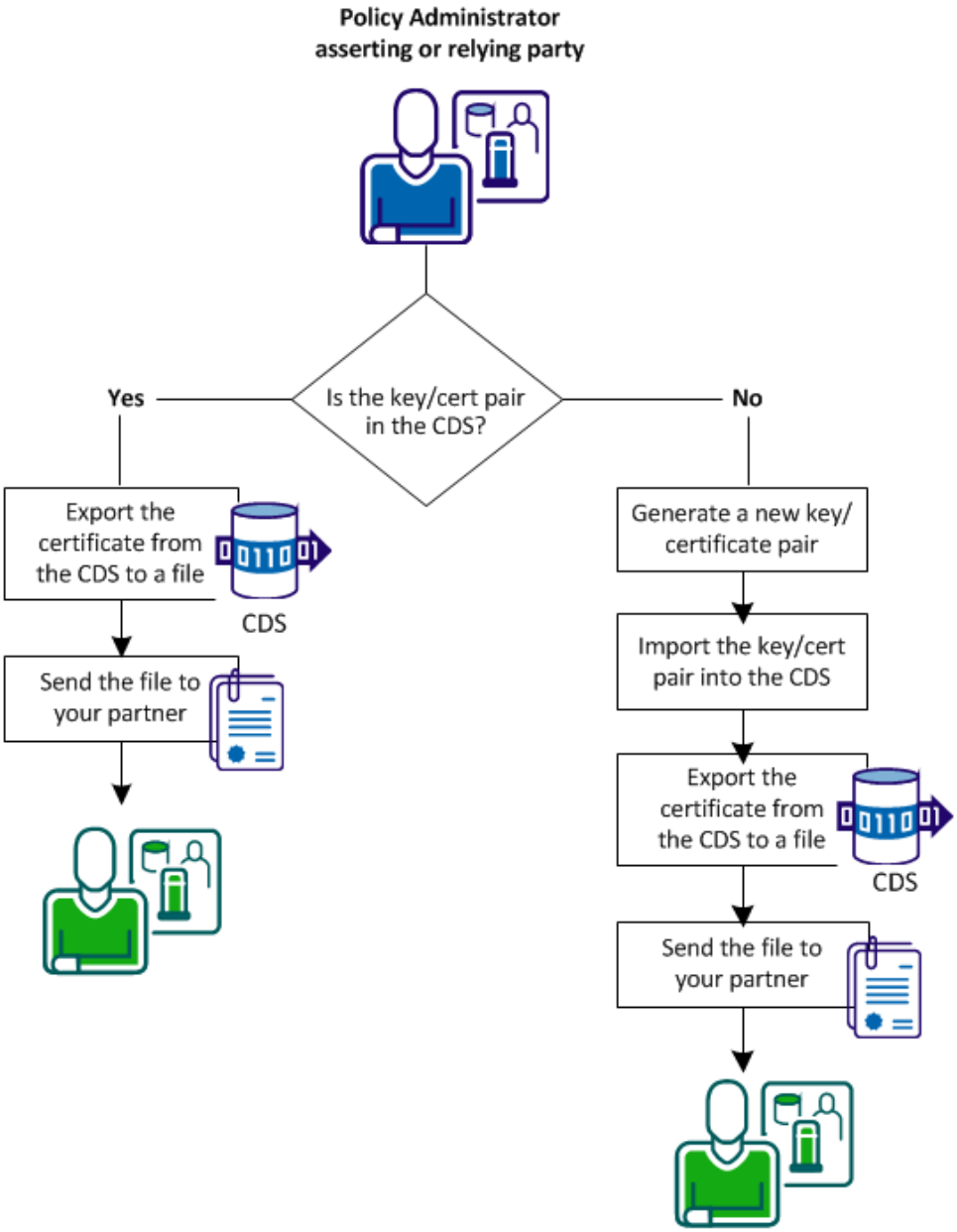
파트너에 인증서를 전송하는 방법

메시지에 서명하는 파트너는 상대 파트너가 메시지를 확인할 수 있도록 관련된 인증서(공개 키)를 전송해야 합니다.

메시지를 암호화하는 파트너는 메시지 암호를 해독할 대상 파트너로부터 인증서(개인 키)를 받아야 합니다.

필요한 인증서 파일을 파트너에게 보내는 절차는 키/인증서 쌍이 CDS 에 이미 있는지 여부에 따라 다릅니다.

다음 그림에서는 인증서 파일을 공유하는 단계를 보여 줍니다.



다음 단계를 수행하십시오.

1. 새로운 키/인증서 쌍을 생성합니다.
2. 키/인증서 쌍을 CDS 에 가져옵니다 (페이지 160).
3. CDS 에서 인증서를 파일로 내보냅니다 (페이지 163).
4. 인증서 파일을 파트너에게 보냅니다. (페이지 163)

UI 또는 타사 도구를 사용하여 새로운 키/인증서 쌍 생성

인증서 데이터 저장소에 키/인증서 쌍이 없는 경우에는 트러스트된 인증 기관에 새로 요청하십시오. CA 에서 서명된 인증서 응답을 반환하면 이를 인증서 데이터 저장소로 가져오십시오.

Administrative UI 또는 타사 도구를 사용하여 인증서 요청을 생성하십시오.

Administrative UI 를 사용하여 인증서 요청을 생성하면 CA SiteMinder?Federation Standalone 에서 개인 키 및 자체 서명 인증서 쌍이 생성됩니다. CA SiteMinder?Federation Standalone 은 이 쌍을 인증서 데이터 저장소에 저장합니다. 생성된 요청을 사용하여 인증 기관에 연락하고 생성된 요청의 내용을 양식에 붙여 넣어 CA 인증서 요청 양식을 기입하십시오.

CA 에서는 일반적으로 PKCS #7 형식으로 서명된 인증서 응답을 생성합니다. 서명된 인증서 응답을 인증서 데이터 저장소로 가져올 수 있습니다. 서명된 인증서 응답을 가져온 후에 별칭이 동일한 기존 자체 서명 인증서가 대체됩니다.

다음 단계를 수행하십시오.

1. Administrative UI 에 로그인합니다.
2. "인증서 및 키" 탭에서 "인증서 및 개인 키"를 선택합니다.
3. "인증서 요청"을 클릭합니다.
4. 필수 필드를 기입합니다.

   참고: "도움말"을 클릭하면 필드, 컨트롤 및 해당되는 각 요구 사항에 대한 설명을 볼 수 있습니다.
5. "저장"을 클릭합니다.

PKCS #10 사양을 따르는 파일이 생성됩니다.

브라우저에서 인증서 요청이 포함된 파일을 저장할지 아니면 열지를 묻는 메시지가 표시됩니다. 이 파일을 저장하지 않더라도(또는 파일을 열고 텍스트를 추출하더라도) CA SiteMinder?Federation Standalone 은 개인 키와 자체 서명 인증서 쌍을 생성합니다. "CSR 생성" 기능을 사용하여 개인 키에 대한 새 요청 파일을 가져오려면 새 인증서 서명 요청을 생성하십시오.

CDS 로 키/인증서 쌍 가져오기

키/인증서 쌍을 가져오는 절차는 상황에 따라 다릅니다. 아래에서 적절한 절차를 참조하십시오.

- 기존 파일에서 키/인증서 쌍을 가져옵니다.

  시스템에 로컬 파일이 있습니다.

- 서명된 인증서 응답을 가져옵니다 (페이지 147).

  Administrative UI 에서 키/인증서 쌍을 생성한 경우, 인증 기관에서 보낸 서명된 응답에서 인증서를 가져오십시오.

기존 파일에서 키/인증서 쌍 가져오기

인증서 데이터 저장소에 키/인증서 쌍이 없는 경우 기존 .p12 또는 .pfx 파일에서 가져옵니다.

CA SiteMinder?Federation Standalone 은 가져온 인증서를 트러스트된 인증서로 취급합니다. 자체 서명된 인증서는 예외입니다.

- 시스템이 V3 자체 서명 인증서를 CA 인증서로 식별하는 경우 인증서는 CA 인증서로 취급됩니다. 이 동작은 인증서/개인 키 대화 상자에서 가져오기를 시작하더라도 수행됩니다.
- CA SiteMinder® Federation Standalone 은 다음과 같은 경우 인증서를 트러스트된 인증서로 취급합니다.
  - CA SiteMinder® Federation Standalone 이 V3 자체 서명 인증서를 CA 로 식별하지 않는 경우
  - 인증서가 V1 자체 서명 인증서가 아닌 경우

다음 단계를 수행하십시오.

1. Administrative UI 에 로그인합니다.
2. "인증서 및 키" 탭에서 "인증서 및 개인 키"를 선택합니다.

   "인증서 및 개인 키 보기" 대화 상자가 열립니다.

3. "새로 가져오기"를 클릭하고 마법사의 단계를 따릅니다.

   참고: "도움말"을 클릭하면 필드, 컨트롤 및 해당되는 요구 사항에 대한 설명을 볼 수 있습니다.

   마법사를 진행할 때 다음 사항을 참조하십시오.

   - 키와 인증서가 들어 있는 단일 파일을 가져오거나, 키 파일과 인증서 파일을 개별적으로 가져올 수 있습니다. 사용하는 파일에 해당하는 옵션 단추를 선택하십시오.
   - 자체 서명된 인증서를 인증 기관 인증서로서 가져오려면 "Use as CA"(CA 로 사용) 옵션 단추를 "예"로 설정하십시오. 인증서는 CA 인증서로서 가져오기되며 파트너 관계를 구성할 때(예: 서명 또는 암호화를 위해 구성할 때)는 사용할 수 없습니다.

     그렇지 않은 경우, 기본값인 "아니요" 설정을 선택하여 파트너 관계를 구성할 때 사용 가능한 트러스트된 인증서로서 인증서를 가져오십시오.
   - DER(바이너리) 형식의 트러스트된 인증서 파일의 경우 파일에는 하나 이상의 인증서가 포함될 수 있습니다. PEM(base 64) 형식의 트러스트된 인증서 파일의 경우 CA SiteMinder® Federation Standalone 은 파일당 하나의 인증서가 있는 것으로 가정합니다.

     DER 또는 PEM 형식인 파일의 표준 확장명은 *.crt 또는 *.cer 입니다.
   - .p12 파일을 사용할 때는 암호를 입력해야 합니다. CA SiteMinder® Federation Standalone 에서는 p12 또는 .pfx 파일을 키/인증서 쌍이 포함된 파일로 처리합니다.
   - 인증서 데이터 저장소에 추가하려는 각 항목에 대해 해당 항목과 연결하려는 별칭을 입력하십시오. 여러 항목을 선택하는 경우 각 항목마다 고유한 별칭이 필요합니다.

4. "확인" 단계에서 정보를 검토하고 "마침"을 클릭합니다.

키/인증서 쌍을 인증서 데이터 저장소로 가져왔습니다.

서명된 인증서 응답 가져오기

인증서 요청을 작성하고 이를 인증 기관에 전송하면 인증 기관에서 서명된 인증서 응답을 발급합니다.

서명된 인증서를 인증 데이터 저장소로 가져와서 동일한 별칭을 가진 기존의 자체 서명된 인증서 항목을 이 서명된 인증서로 바꾸십시오.

다음 단계를 수행하십시오.

1. "인증서 및 키" 탭에서 "인증서 및 개인 키"를 선택합니다.

   "인증서 및 개인 키 보기" 대화 상자가 열립니다.

2. 별칭이 동일한 자체 서명된 항목을 검색합니다.
3. 자체 서명된 인증서가 들어 있는 항목 옆의 "작업", "인증서 업데이트"를 선택합니다.

   인증서 및 키 가져오기를 위한 마법사가 표시됩니다.

   참고: "도움말"을 클릭하면 필드, 컨트롤 및 해당되는 각 요구 사항에 대한 설명을 볼 수 있습니다.

4. 원하는 파일을 찾아 이동합니다. 사용 가능한 파일은 다음과 같습니다.
   - 서명된 인증서와 해당 인증서 체인이 포함된 .p7 또는 .p7b 파일
   - 인증서 체인 없이 서명된 인증서가 있는 .cer 또는 .crt 파일(base64 PEM 파일)
5. 적절한 항목을 선택합니다.
6. "확인" 단계에서 인증서 정보를 검토하고 "마침"을 클릭합니다.

서명된 인증서를 인증서 데이터 저장소로 가져오게 되고 자체 서명된 인증서가 대체됩니다.

Administrative UI 를 사용하여 CDS 에서 인증서 내보내기

개인 키/인증서 쌍을 파일로 내보낸 후 인증서 파일(공개 키)을 페더레이션 파트너에게 보낼 수 있습니다. 파트너는 관련된 개인 키를 사용하여 생성된 어설션 응답의 서명을 확인하거나, 관련된 개인 키를 사용하여 암호를 해독할 응답을 암호화하는 데 이 인증서를 사용할 수 있습니다.

중요! 백업 중 개인 키를 내보낼 경우 다른 사람과 공유하지 마십시오.

다음 단계를 수행하십시오.

1. Administrative UI 에 로그인합니다.
2. "인증서 및 키" 탭에서 "인증서 및 개인 키"를 선택합니다.

   "인증서 및 개인 키 보기" 창이 나타납니다.
3. "인증서 및 개인 키 목록"에서 내보낼 항목에 대해 "작업", "내보내기"를 선택합니다.

   "키 저장소 항목 내보내기" 대화 상자가 나타납니다.
4. 내보낸 데이터에서 생성하고자 하는 파일의 형식을 선택합니다.

   참고: "도움말"을 클릭하면 필드, 컨트롤 및 해당되는 각 요구 사항에 대한 설명을 볼 수 있습니다.
5. 파일 형식을 선택합니다.
6. "Export"(내보내기)를 클릭합니다.

   로컬 시스템에서 파일을 열거나 저장하라는 메시지가 표시됩니다.

   CA SiteMinder® Federation Standalone 은 키 또는 인증서를 나타내는 인코딩된 파일 콘텐츠를 생성합니다.
7. 파일을 파트너에게 보냅니다 (페이지 163).

파트너에게 인증서 파일 보내기

인증서가 포함된 인코딩 파일을 내보낸 후에는 이 파일을 페더레이션 파트너에게 보내십시오. 파트너는 페더레이션 메시지를 확인하거나 암호화하기 위해 이 인증서를 가져와야 합니다.

인증서 데이터 저장소에서 인증서 업데이트

키/인증서 쌍과 독립 실행형 인증서를 다음과 같은 방법으로 업데이트할 수 있습니다.

- 기존 인증서를 삭제하고 새 트러스트된 인증서를 가져와서 만료되는 트러스트된 인증서를 업데이트합니다. 새 인증서는 인증서 데이터 저장소의 만료 인증서와 일치해야 합니다.
- 서명되고 트러스트된 인증서 또는 PKCS7 서명된 응답을 가져와서 인증서를 업데이트합니다. 새 인증서는 인증서 데이터 저장소의 만료 인증서와 일치해야 합니다.
- 인증서를 PKCS#12 파일의 인증서로 업데이트합니다. 새 개인 키 및 인증서 쌍은 인증서 데이터 저장소의 만료되는 키/인증서 쌍과 일치해야 합니다.

새 인증서가 유효해야만 CA SiteMinder?Federation Standalone 이 이를 사용하여 만료되는 인증서를 업데이트할 수 있습니다. 인증서를 가져오면 즉시 인증서가 업데이트되고 사용 가능하게 됩니다. 유효 간격으로 확인한 결과 새 인증서가 유효하지 않으면 CA SiteMinder?Federation Standalone 은 새 인증서를 사용할 수 없습니다.

트러스트된 인증서만 가져오려면 PEM 또는 DER 인코딩을 사용하는 인증서 파일을 사용하십시오. 이러한 파일 유형의 표준 확장명은 *.crt 또는 *.cer 입니다. 파일이 .p12 또는 .pfx 로 끝나면 키/인증서 쌍이 포함된 인증서 데이터 저장소 파일로 처리됩니다. 마지막으로 파일이 .p7 또는 .p7b 로 끝나면 서명된 응답 파일로 처리됩니다. 다른 파일은 모두 인증서 파일로 취급되며 CA SiteMinder?Federation Standalone 은 여기에서 인증서 로드를 시도합니다.

참고: 페더레이션 환경의 인증서를 업데이트하는 경우 만료되는 인증서를 사용하는 페더레이션 개체는 업데이트할 필요가 없습니다.

인증 기관(CA) 인증서 사용

페더레이션 시스템은 인증 기관 인증서를 사용하여 다음 항목을 확인합니다.

- SAML HTTP-아티팩트 백 채널에 보안을 적용하는 SSL 연결에 대한 SSL 서버 인증서가 트러스트된 인증서인지 여부
- HTTP-아티팩트 싱글 사인온의 경우 SSL 연결로 백 채널에 보안을 적용하십시오. 페더레이션 시스템의 포함된 웹 서버는 인증 기관의 인증서를 확인하여 SSL 연결이 트러스트된 인증서로 보안 적용되는지 확인할 수 있습니다. 이 인증서는 인증서 데이터 저장소에 저장되어야 합니다.
- 인증서 해지 목록이 유효한지 여부.

  CRL 은 인증 기관에서 가져옵니다. CRL 의 유효성을 검사하려면 해당 CA 의 인증서가 필요합니다. CRL 은 런타임에 사용되기 위해 데이터 저장소에 저장됩니다.

이를 위해 공통 루트 및 중간 CA 인증서의 기본 집합이 제품과 함께 제공됩니다.

CA 인증서 가져오기

공통 루트 및 중간 CA 인증서의 집합이 제품에 포함되어 있습니다. 인증서 데이터 저장소에 없는 CA 인증서를 사용하려면 해당 CA 인증서를 가져오십시오.

가져오는 모든 인증서는 CA 인증서로 취급됩니다. 자체 서명된 인증서는 예외입니다.

- 시스템이 V3 자체 서명 인증서를 비 CA 인증서로 식별하는 경우 인증서는 트러스트된 인증서로 취급됩니다. 이 동작은 "CA 인증서 가져오기" 대화 상자에서 가져오기를 시작한 경우에도 적용됩니다.
- 시스템이 V1 자체 서명된 인증서를 식별하는 경우 인증서는 CA 인증서로 취급됩니다.

참고: 루트 CA 인증서를 가져오는 경우, 트러스트 체인의 일부인 모든 루트 CA 인증서를 가져오십시오.

CA 인증서를 가져오려면

1. Administrative UI 에 로그인합니다.
2. "인증서 및 키", "인증 기관"을 선택합니다.

   "인증 기관 목록"이 표시됩니다.
3. "새로 가져오기"를 클릭합니다.

   "CA 인증서 가져오기" 대화 상자가 표시됩니다.

   참고: "도움말"을 클릭하면 필드, 컨트롤 및 해당되는 각 요구 사항에 대한 설명을 볼 수 있습니다.
4. 마법사의 단계에 따라 새 항목을 가져옵니다.
5. "확인" 단계에서 인증서 정보를 검토하고 "마침"을 클릭합니다.

CA 인증서를 인증서 데이터 저장소로 가져왔습니다. 변경 내용은 가져오기가 완료된 직후 적용됩니다.

중요! 시스템에서 사용 중인 인증서에 대한 트러스트 체인의 일부인 CA 인증서는 삭제할 수 없습니다. 사용 중인 CA 인증서를 삭제하려고 하면 인증서를 삭제할 수 없다는 오류 메시지가 표시됩니다.

백 채널 통신을 위한 인증서 서명 확인 문제 해결

증상

HTTP-아티팩트를 싱글 사인온 프로필로 사용 중입니다. 어설션 당사자는 SSL 백 채널을 통해 신뢰 당사자와 통신합니다. 신뢰 당사자는 SSL 백 채널을 통해 통신하기 위해 어설션 당사자에 있는 서버 인증서의 서명을 확인해야 합니다.

서버 인증서의 서명을 확인하지 못할 경우 다음과 같은 오류가 기록됩니다. `[Dispatcher object thrown unknown exception while processing the request message. Message: Certificate not verified..](요청 메시지를 처리하는 동안 디스패처 개체가 알 수 없는 예외를 throw 했습니다. 메시지: 인증서가 확인되지 않았습니다.)`

해결 방법

신뢰 당사자가 루트 CA 인증서를 인증서 데이터 저장소에 가져와야 합니다. 이 인증서는 어설션 당사자 측에서 서버 인증서 서명을 확인하는 데 필요합니다. 인증서 서명을 확인하려면 서버 인증서에 서명한 루트 CA 를 가져오십시오.

확인을 위해 가져오는 CA 인증서에 대해 다음과 같은 정보를 확인하십시오.

- 루트 CA 인증서의 발급자 및 주체 DN 이 동일합니까? 다른 경우, 인증서는 중간 루트 CA 입니다. 트러스트된 체인에 포함된 모든 루트 CA 인증서를 가져오십시오.
- 어설션 당사자 서버 인증서의 발급자와 가져온 루트 CA 의 발급자 및 주체가 일치하는지 확인합니다.

제 8 장: 파트너 관계 생성 및 활성화

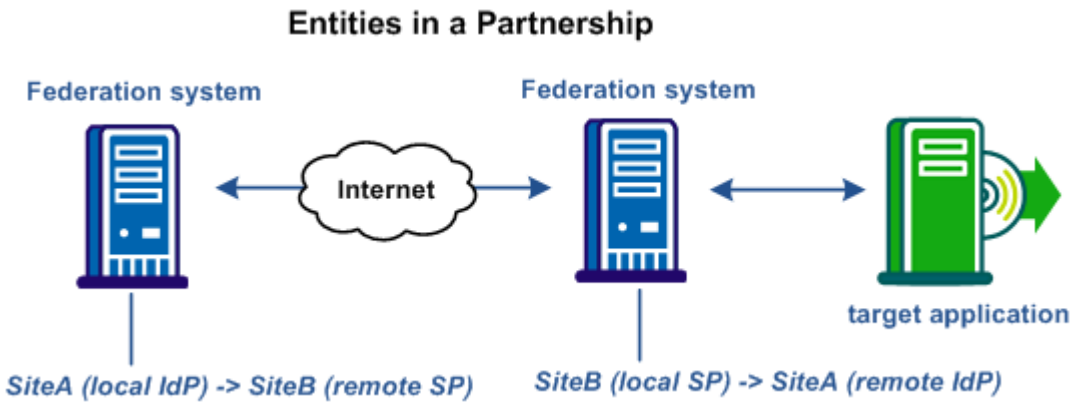
이 섹션은 다음 항목을 포함하고 있습니다.

파트너 관계 생성

CA SiteMinder?Federation Standalone 의 기본 목적은 두 조직이 사용자 아이덴티티 정보를 공유하고 SSO(싱글 사인온)를 사용할 수 있도록 조직 간에 파트너 관계를 설정하는 것입니다. 파트너 관계는 서로 다른 사이트에 있는 두 개의 엔터티로(로컬과 원격에 각각 하나씩) 구성됩니다. 각 엔터티는 어설션 당사자, 어설션 또는 신뢰 당사자를 생산하는 측, 어설션을 소비하는 측의 역할을 수행할 수 있습니다.

CA SiteMinder?Federation Standalone 이 두 사이트에 모두 설치된 경우 각 사이트마다 파트너 관계를 정의해야 합니다. 한 사이트의 각 로컬 어설션 당사자-신뢰 당사자 파트너 관계마다, 파트너 사이트에 이에 대응하는 로컬 신뢰 당사자-어설션 당사자 파트너 관계가 있어야 합니다. 두 개의 정의가 하나의 파트너 관계를 정의합니다.

다음 그림에서 SiteA 는 로컬 SAML 2.0 IdP 로 정의되었으며 SiteB 를 원격 SAML 2.0 SP 로 지정합니다. SiteB 는 로컬 SAML 2.0 SP 로 구성되었으며 SiteA 는 SiteB 의 원격 SAML 2.0 IdP 입니다.



참고: 어설션 당사자는 둘 이상의 신뢰 당사자와 파트너 관계를 가질 수 있으며 신뢰 당사자는 둘 이상의 어설션 당사자와 파트너 관계를 설정할 수 있습니다.

페더레이션 파트너 관계를 생성하는 작업은 다음의 두 단계로 이루어집니다.

1. 파트너 관계 유형을 지정합니다.
2. 다음과 같은 파트너 관계 상세 정보를 구성합니다.
    a. 파트너 관계 이름과 관련 엔터티
    b. 페더레이션 사용자(로컬 어설션 당사자에만 해당)
    c. 이름 ID 형식 및 기타 어설션 특성(로컬 어설션 당사자에만 해당)
    d. 사용자 ID(로컬 신뢰 당사자에만 해당)
    e. SSO(싱글 사인온)
    f. SLO(싱글 로그아웃) – SAML 2.0 에만 해당
    g. 서명
    h. 암호화 – SAML 2.0 에만 해당

파트너 관계 정의

페더레이션 파트너 관계 정의는 어떤 페더레이션 역할이 로컬이고 어떤 페더레이션 역할이 원격인지를 지정합니다.

다음 단계를 수행하십시오.

1. Administrative UI 에 로그인합니다.
2. "페더레이션" 탭에서 "파트너 관계"를 선택합니다.
3. "페더레이션 파트너 관계" 목록 섹션에서 "파트너 관계 만들기"를 클릭합니다.

   참고: "도움말"을 클릭하면 필드, 컨트롤 및 해당되는 각 요구 사항에 대한 설명을 볼 수 있습니다.
4. 다음 옵션 중 하나를 선택하십시오.
   - SAML2 IDP->SP(아이덴티티 공급자가 로컬)
   - SAML2 SP->IDP(서비스 공급자가 로컬)
   - SAML1.1 생산자->소비자(생산자가 로컬)
   - SAML1.1 소비자->생산자(소비자가 로컬)
   - WSFED IP->RP(아이덴티티 공급자가 로컬)
   - WSFED RP->IP(리소스 파트너가 로컬)

"페더레이션 파트너 관계 만들기" 대화 상자에 파트너 관계 구성의 첫 번째 단계가 표시됩니다.

파트너 관계 식별 및 구성

마법사의 "파트너 관계 구성" 단계에서 파트너 관계의 이름을 지정하고 로컬 및 원격 엔터티를 지정하여 파트너 관계를 식별하십시오.

참고: "도움말"을 클릭하면 필드, 컨트롤 및 해당되는 각 요구 사항에 대한 설명을 볼 수 있습니다.

다음 단계를 수행하십시오.

1. 파트너 관계의 이름을 입력합니다. 이름에는 영숫자 문자, 밑줄, 하이픈 및 마침표를 사용할 수 있습니다. 공백은 사용할 수 없습니다.
2. (선택 사항) 설명을 입력합니다.

3. 엔터티를 이미 구성한 경우 로컬 목록에서 로컬 엔터티를 선택합니다. 그렇지 않은 경우에는 "로컬 엔터티 만들기"를 클릭합니다.
4. 엔터티를 이미 구성한 경우 원격 목록에서 원격 엔터티를 선택합니다. 그렇지 않은 경우에는 "원격 엔터티 만들기"를 클릭합니다.

   참고: 나중에 메타데이터를 가져와서 원격 엔터티를 만들 계획인 경우 이 단계를 미룰 수 있습니다.
5. (선택 사항) "차이 시간"을 초 단위로 입력합니다.

   차이 시간은 로컬 시스템의 시스템 시간과 원격 시스템의 시스템 시간 사이의 차이입니다. 일반적으로 시스템 시계가 부정확할 때 이런 상태가 발생합니다. 현재 시간에서 초를 빼서 차이 시간을 결정하십시오.

   시스템에서는 차이 시간 및 SSO 유효 기간을 사용하여 어설션의 유효 기간을 결정합니다.
6. "사용 가능한 디렉터리" 목록에서 하나 이상의 사용자 디렉터리를 선택하고 이를 "선택된 디렉터리" 목록으로 이동합니다.

   사용자 디렉터리를 하나만 구성한 경우에는 이 디렉터리가 자동으로 "선택된 디렉터리" 목록으로 들어갑니다.

   중요! 사용자 디렉터리로 ODBC 데이터베이스를 사용하려면 SQL 쿼리 체계 및 올바른 SQL 쿼리를 정의하십시오. 사용자 디렉터리로 선택하려면 다음 단계를 수행해야 합니다.
7. "다음"을 클릭하여 파트너 관계 마법사를 계속합니다. 마법사의 단계를 따라 파트너 관계의 여러 기능(일부는 필수, 일부는 선택적 기능)을 구성할 수 있습니다. 이러한 기능의 구성 정보는 이 안내서의 이후 단원에서 설명됩니다.

참고: 파트너 관계를 편집하는 경우에는 이 필드 옆의 "업데이트 가져오기"를 클릭하여 엔터티 정보를 업데이트할 수 있습니다. 엔터티 구성의 최신 정보가 파트너 관계에 전파됩니다. 하지만 엔터티 정보를 파트너 관계에서 직접 편집하는 경우에는 변경 내용이 다시 개별 엔터티 구성으로 전파되지 않습니다.

파트너 관계의 엔터티 편집

로컬 및 원격 엔터티 필드 옆의 "업데이트 가져오기"를 클릭하여 엔터티에 대한 정보를 업데이트할 수 있습니다. "업데이트 가져오기"를 선택하면 시스템은 엔터티에서 최신 정보를 가져올지 묻습니다.

확인 후에는 편집하는 파트너 관계가 최신 엔터티 정보로 새로 고쳐집니다. 파트너 관계 마법사를 완료하면 변경 내용이 저장됩니다. 업데이트를 확인하지 않으면 파트너 관계 구성이 동일하게 유지됩니다.

"엔터티 이름"은 정책 저장소의 엔터티 개체를 식별합니다. 제품에서는 이 값을 사용하여 엔터티를 내부적으로 구분하기 때문에 "엔터티 이름"은 고유 식별자여야 합니다. 이 값은 외부에서 사용되지 않으며 원격 파트너는 이 값을 알지 못합니다.

엔터티 ID 가 원격 파트너를 나타내는 경우 이 값은 고유해야 합니다. 엔터티 ID 가 로컬 파트너를 나타내는 경우에는 동일한 시스템에서 재사용될 수 있습니다.

참고: "엔터티 이름"의 값은 "엔터티 ID"와 동일할 수 있지만 값을 다른 엔터티와 공유하지 마십시오.

엔터티는 페더레이션 파트너 관계의 핵심 구성 요소입니다. 엔터티를 변경하면 파트너 관계가 크게 변경됩니다. 따라서 Administrative UI 에서는 파트너 관계가 설정된 후 엔터티를 바꿀 수 없습니다. 엔터티를 바꾸려면 파트너 관계를 생성하십시오.

엔터티 ID 는 엔터티를 고유하게 식별하지 않기 때문에 파트너 관계 구성 내에서 약간의 유연성을 제공하기 위해 엔터티 ID 를 변경할 수 있습니다. 파트너 관계 수준에서 엔터티 ID 를 변경하더라도 파트너 관계가 다른 엔터티로 연결되지 않습니다. 파트너 관계의 원래 엔터티는 변경되지 않습니다. 엔터티 수정은 엔터티에서 파트너 관계로의 단방향 전파입니다. 파트너 관계의 엔터티 ID 변경 내용은 원본 엔터티로 다시 전파되지 않습니다.

엔터티 구성을 템플릿이라고 생각하십시오. 파트너 관계는 엔터티 템플릿을 기반으로 생성되므로 파트너 관계를 변경하더라도 원본 엔터티 템플릿은 변경되지 않습니다.

파트너 관계 확인

파트너 관계 구성을 저장하기 전에 검토하십시오.

다음 단계를 수행하십시오.

1. 파트너 관계 마법사의 "확인" 단계에서 설정을 검토합니다.
2. 각 그룹 상자에서 "수정"을 클릭하여 설정을 변경합니다.
3. 원하는 대로 구성되었으면 "마침"을 클릭합니다.

파트너 관계 구성이 완료되었습니다.

파트너 관계 활성화

파트너 관계에 필요한 모든 설정을 구성한 후에는 파트너 관계를 사용할 수 있도록 활성화하십시오. 동일한 프로세스를 사용하여 파트너 관계를 비활성화할 수도 있습니다.

다음 단계를 수행하십시오.

1. "페더레이션" 탭에서 "파트너 관계"를 선택합니다.
2. "작업" 메뉴에서 원하는 파트너 관계 옆의 "활성화" 또는 "비활성화"를 선택합니다.

   참고: "활성화"는 "정의됨" 또는 "비활성" 상태의 파트너 관계에 대해서만 사용할 수 있으며, "비활성화"는 "활성" 상태의 파트너 관계에 대해서만 사용할 수 있습니다.
3. "예"를 클릭하여 선택을 확인합니다.

   파트너 관계의 상태가 설정되고 표시가 새로 고쳐집니다.

중요! 파트너 관계를 수정하기 전에는 비활성화하십시오.

파트너 관계 내보내기

메타데이터를 원격 엔터티를 생성하고 파트너 관계를 형성하는 기반으로 사용할 수 있습니다. 메타데이터를 사용하면 엔터티의 많은 측면이 이미 메타데이터 파일에 정의되어 있으므로 파트너 관계를 더 효율적으로 구성할 수 있습니다. 그런 다음 파일을 가져와서 새로운 파트너 관계 또는 원격 엔터티를 생성할 수 있습니다.

파트너 관계를 완료해야만 내보낼 수 있는 것은 아닙니다. 파트너 관계의 일부만 구성한 후 내보낼 수 있습니다.

Administrative UI 에서는 기존 파트너 관계 항목의 메타데이터를 내보낼 수 있습니다.

참고: Administrative UI 에서는 기존 로컬 어설션 또는 신뢰 엔터티의 메타데이터를 내보낼 수 있습니다. SAML 1.1 데이터를 내보낼 때 결과 메타데이터 파일에 사용되는 용어는 SAML 2.0 용어입니다. 이러한 규칙은 SAML 사양의 일부입니다. SAML 1.1 데이터를 가져올 때 용어는 SAML 1.1 용어를 사용하여 올바르게 가져오게 됩니다.

파트너 관계에서 내보낼 때는 선택된 파트너 관계가 내보내기의 기반으로 사용됩니다. 새 파트너 관계 이름은 정의할 수 없습니다. 선택한 파트너 관계의 이름이 사용됩니다.

다음 단계를 수행하십시오.

1. "페더레이션" 탭에서 "파트너 관계"를 선택합니다.
2. 목록에서 적절한 항목 옆의 "작업" 풀다운 메뉴를 클릭하고 "메타데이터 내보내기"를 선택합니다.
3. 대화 상자의 필드를 입력합니다.

   "활성" 상태의 파트너 관계를 내보내는 경우에는 대부분의 필드가 읽기 전용이며 "유효 기간" 필드와 별칭 드롭다운 목록만 편집할 수 있습니다.

   참고: "도움말"을 클릭하면 필드, 컨트롤 및 해당되는 각 요구 사항에 대한 설명을 볼 수 있습니다.
4. "내보내기"를 클릭하여 완료합니다.

5. 메타데이터 파일을 열지 아니면 저장할지 묻는 대화 상자가 표시됩니다. 파일을 열어서 볼 수 있습니다.
6. 데이터를 로컬 시스템의 XML 파일에 저장합니다.

메타데이터가 지정된 XML 파일로 내보내졌습니다.

제 9 장: 파트너 관계에 대한 페더레이션된 사용자 식별

이 섹션은 다음 항목을 포함하고 있습니다.

어설션 당사자 측에서의 페더레이션 사용자 구성

"페더레이션 사용자" 대화 상자는 로컬 엔터티가 어설션 당사자인 경우 파트너 관계 마법사의 두 번째 단계입니다. 이 단계에서는 원격 사이트의 대상 리소스에 대한 액세스 권한을 부여할 사용자를 지정할 수 있습니다. 또한 CA SiteMinder?Federation Standalone 을 SiteMinder 에 통합하기 위한 SiteMinder 커넥터를 사용하도록 설정할 수도 있습니다.

SiteMinder 커넥터는 배포되어 있는 SiteMinder 시스템을 CA SiteMinder?Federation Standalone 과 통합하는 데 사용되는 소프트웨어 구성 요소입니다. CA SiteMinder?Federation Standalone 이 어설션 당사자 측에 있으면 SiteMinder 커넥터가 SiteMinder 세션에서 CA SiteMinder?Federation Standalone 세션을 생성할 수 있습니다. SiteMinder 세션을 설정하기 위해 SiteMinder 가 사용자를 먼저 인증한 후 사용자가 어설션 당사자를 방문합니다.

SiteMinder 커넥터를 파트너 관계별로 사용할 수 있지만 파트너 관계 전체에는 하나의 전역 커넥터 구성만 적용됩니다. "배포 설정"에서 확인란을 선택하고 구성이 정의되어 있는 경우에만 커넥터를 사용할 수 있습니다. "배포 설정"은 UI 의 "인프라" 탭에서 액세스할 수 있습니다. 커넥터를 전역으로 사용하도록 설정하면 CA SiteMinder?Federation Standalone 이 파트너 관계 구성을 평가하여 커넥터가 사용되는지 여부를 확인합니다. 파트너 관계에는 전역 커넥터 구성이 사용됩니다.

파트너 관계에서 커넥터를 사용하지 않으려면 파트너 관계 수준에서 확인란의 선택을 취소하십시오. 전역 수준에서 커넥터를 사용하지 않으려면 "배포 설정"에서 해당 설정을 해제하십시오.

중요! 전역 수준에서 커넥터를 사용하지 않도록 설정하면 CA SiteMinder?Federation Standalone 의 파트너 관계 수준에서 확인란이 무시됩니다.

페더레이션 사용자 구성

페더레이션 사용자는 보호된 페더레이션 리소스에 액세스할 수 있는 사용자입니다.

다음 단계를 수행하십시오.

참고: "도움말"을 클릭하면 필드, 컨트롤 및 해당되는 각 요구 사항에 대한 설명을 볼 수 있습니다.

1. "페더레이션된 사용자" 그룹 상자의 표에 있는 "디렉터리" 열의 목록에서 사용자 디렉터리를 선택합니다.

   풀다운 목록은 이전 대화 상자에서 지정한 디렉터리 수에 따라 하나 이상의 디렉터리 목록으로 구성됩니다.

2. "사용자 클래스" 열에서 사용자 클래스를 선택합니다. 이 항목은 인증할 수 있는 개별 사용자 또는 사용자 그룹의 범주를 지정합니다. 이 필드의 옵션은 사용자 디렉터리(LDAP 또는 ODBC)의 유형에 따라 다릅니다. 각 사용자 클래스의 설명 및 예제는 "사용자 클래스" 표를 참조하십시오.

3. 사용자 이름/필터 기준 열에 이름 또는 필터를 입력합니다. 이 열의 값을 사용하여 시스템은 페더레이션된 사용자를 인증하는 사용자 또는 사용자 그룹을 찾을 수 있습니다. 이 항목은 "사용자 클래스" 열에 대해 선택하는 값에 따라 다릅니다. 이름 및 필터의 예제는 이 절차의 끝에 나오는 표를 참조하십시오.

4. (선택 사항) 항목에 대해 "제외"를 선택하여 사용자 클래스를 제외하도록 지정할 수 있습니다. 기본값은 디렉터리의 모든 사용자를 포함하는 것입니다.

   참고: 제외 조건과 포함 조건이 충돌할 경우 항상 제외 조건이 포함 조건보다 우선합니다.

5. (선택 사항) 동일한 디렉터리 또는 다른 사용자 디렉터리에 대해 다른 사용자 클래스를 지정하려면 "행 추가"를 클릭합니다.

6. (선택 사항) SiteMinder 커넥터 설정을 구성합니다.

   a. CA SiteMinder® Federation Standalone 을 기존 SiteMinder 배포에 통합하는 경우 확인란을 선택하여 SiteMinder 커넥터를 사용하도록 설정합니다.

   b. (선택 사항) CA SiteMinder® Federation Standalone 또는 SiteMinder 에서 유니버설 ID 를 사용하여 사용자 레코드를 가져오도록 "UserDN 및 디렉터리 이름 비교 적용"의 선택을 취소합니다. "유니버설 ID"를 사용하면 사용자 디렉터리가 물리적으로 다르고 유형이 다를 수 있습니다. "유니버설 ID"만으로도 가져온 사용자 레코드가 올바른 레코드인지 확인할 수 있습니다.

      참고: "유니버설 ID"를 사용할 경우 각 사용자의 유니버설 ID 가 고유해야 합니다. 유니버설 ID 가 고유하지 않으면 사용자 레코드에 액세스하는 시스템이 잘못된 레코드를 가져올 수 있습니다.

      확인란을 선택된 상태(기본값)로 두면 CA SiteMinder® Federation Standalone 과 SiteMinder 가 물리적으로 같은 디렉터리를 사용해야 합니다. 사용자가 저장소 조회를 수행하려면 두 디렉터리의 이름이 같아야 합니다. 사용자를 인증하는 엔터티는 사용자가 제공하는 정보를 사용자 레코드의 UserDN 및 "디렉터리 이름"과 비교합니다.

   사용자 선택이 완료되었습니다.

7. "다음"을 클릭합니다.

   "어설션 구성" 대화 상자가 나타납니다.

사용자 클래스 항목의 예제

LDAP 예제

항목을 지정할 때 LDAP 필터 구문을 사용하십시오.

사용자 클래스	유효한 항목
사용자	사용자의 고유 이름입니다. 예: uid=user1,ou=People,dc=example,dc=com
Group	목록에서 선택된 그룹입니다. 예: ou=Sales,dc=example,dc=com
조직 단위	목록에서 선택된 조직 단위입니다. 예: ou=People,dc=example,dc=com
사용자 속성 필터링	LDAP 필터입니다. 현재 사용자부터 검색이 시작됩니다. **예 1:** mail=user@example.com **예 2:** (|(mail=*@.example.com)(memberOf=cn=Employees,ou=Groups,dc=example,dc=com))
그룹 속성 필터링	LDAP 필터입니다. 현재 사용자가 필터와 일치하는 그룹 중 하나의 구성원인 경우 현재 사용자가 권한 부여됩니다. SiteMinder 레지스트리에 구성된 대로 그룹의 objectclass 가 필터와 결합됩니다. **예 1:** 비즈니스 범주가 "CA Support"인 그룹의 구성원인 사용자를 권한 부여하려면 businessCategory=CA Support 를 입력하십시오. **예 2:** 설명에 "Administrator"가 포함되어 있고 비즈니스 범주가 "Administration"인 그룹의 구성원인 사용자를 권한 부여하려면 (|(description=*Administrator*)(businessCategory=Administration))를 입력하십시오. **참고:** 그룹의 일부 특성은 검색 조건으로 사용되지 않습니다.

사용자 클래스	유효한 항목
OU 속성 필터링	LDAP 필터입니다. 현재 사용자가 필터와 일치하는 조직 단위에 속하는 경우 현재 사용자가 권한 부여됩니다. SiteMinder 레지스트리에 구성된 대로 조직 단위의 objectclass 가 필터와 결합됩니다. **예 1:** 우편 주소가 "12345"인 조직 단위 내 사용자를 권한 부여하려면 postalCode=12345 를 입력하십시오. **예 2:** 기본 설정 배송 방법이 "phone"으로 끝나고 지역이 "London"인 조직 단위의 사용자를 권한 부여하려면 (|(preferredDeliveryMethod=*phone)(l=London))를 입력하십시오.
모두 필터링	LDAP 필터입니다. 현재 사용자가 필터와 일치하는 경우 현재 사용자가 권한 부여됩니다. **예 1:** 부서가 "CA Support"인 사용자를 권한 부여하려면 department=CA Support 를 입력하십시오. **예 2:** "Administrators" 그룹의 구성원이고 부서 번호가 "123" 또는 "789"인 사용자를 권한 부여하려면 (&(memberof=cn=Administrators,ou=Groups,dc=example,dc=com)(|(departmentNumber=123)(departmentNumber=789)))를 입력하십시오.

ODBC 예제

쿼리를 지정할 때 SQL 구문을 사용하십시오.

사용자 클래스	유효한 항목
사용자	사용자에 대한 "이름" 열의 값입니다. 현재 사용자가 항목과 일치하는 경우 현재 사용자가 권한 부여됩니다. 예: user1

사용자 클래스	유효한 항목
Group	사용자 그룹의 "이름" 열의 값입니다. 현재 사용자가 쿼리와 일치하는 그룹의 구성원인 경우 현재 사용자가 권한 부여됩니다. 예: Administrators
쿼리	SQL SELECT 문입니다. 현재 사용자가 쿼리와 일치하는 경우 현재 사용자가 권한 부여됩니다. **예 1:** user1 의 userid: 입력: SELECT * FROM SmUser 결과 쿼리: SELECT * FROM SmUser WHERE Name = 'user1' **예 2:** user1 의 userid: 입력: SELECT * FROM SmUser WHERE Status LIKE 'Active%' 결과 쿼리: SELECT * FROM SmUser WHERE Status LIKE 'Active%' AND Name = 'user1' **예 3:** user1 의 userid: 입력: SELECT * FROM SmUser WHERE Location IN ('London', 'Paris') 결과 쿼리: SELECT * FROM SmUser WHERE Location IN ('London', 'Paris') AND Name = 'user1'

신뢰 당사자 측에서의 사용자 식별

신뢰 당사자 측에서 파트너는 로컬 사용자 디렉터리의 사용자를 찾을 수 있어야 합니다. 사용자 디렉터리에서 사용자를 찾는 과정이 바로 명확성 프로세스입니다. "사용자 ID" 대화 상자에서 사용자 명확성을 위한 아이덴티티 특성을 구성하십시오.

CA SiteMinder?Federation Standalone 은 명확성 프로세스를 위해 다음 방법 중 하나를 사용할 수 있습니다.

- 어설션에서 "이름 ID" 값을 추출합니다.
- 어설션의 특정한 특성 값을 사용합니다.
- Xpath 쿼리를 통해 가져온 값을 사용합니다.

  Xpath 쿼리는 어설션에서 "이름 ID" 이외의 특성을 찾아서 추출합니다.

어설션에서 어떤 특성이 추출되었는지 확인한 후 이 특성을 검색 사양에 추가합니다. 그러면 CA SiteMinder?Federation Standalone 이 이 특성을 사용하여 사용자 저장소에서 사용자를 찾습니다. 명확성 프로세스에 성공하면 CA SiteMinder?Federation Standalone 이 사용자에 대한 세션을 생성합니다.

SAML 2.0 의 경우 어설션 당사자가 사용자 식별자를 생성할 수 있도록 허용하는 AllowCreate 기능 (페이지 186)을 구성할 수도 있습니다.

싱글 사인온은 어설션 당사자에게 인증 요청(AuthnRequest)을 보내는 신뢰 당사자가 시작할 수 있습니다. 이 요청에서 신뢰 당사자는 어설션 당사자가 어설션에 특정 사용자 특성을 포함하도록 요청할 수 있습니다. 그러나 필요한 특성 값이 어설션 당사자의 사용자 레코드에 없을 수도 있습니다.

신뢰 당사자가 보내는 인증 요청에 "허용/만들기" 특성이 포함되어 있고, 새 식별자를 생성하도록 어설션 당사자가 구성된 경우, 어설션 당사자는 고유한 값을 NameID 로 생성합니다. 이 값은 어설션에 추가되어 신뢰 당사자에게 다시 전송됩니다.

"사용자 ID" 대화 상자에서는 SiteMinder 커넥터를 사용하도록 설정할 수도 있습니다.

SiteMinder 커넥터는 CA SiteMinder?Federation Standalone 에 포함되어 있는 소프트웨어 구성 요소로, 배포되어 있는 SiteMinder 시스템을 CA SiteMinder?Federation Standalone 에 통합할 수 있게 합니다. 신뢰 당사자 측에서 SiteMinder 와 CA SiteMinder?Federation Standalone 을 통합할 경우 SiteMinder 는 사용자가 SiteMinder 에서 보호되는 리소스를 요청할 때 CA SiteMinder?Federation Standalone 에서 인증된 사용자에게 인증을 다시 요청하지 않습니다. 이 경우 CA SiteMinder?Federation Standalone 에서 인증된 사용자에 대해 정책 서버에 있는 사용자 지정 SiteMinder 인증 스키마와 커넥터를 사용하여 SiteMinder 세션을 생성할 수 있기 때문에 인증을 다시 요청하지 않습니다.

SiteMinder 커넥터를 파트너 관계별로 사용할 수 있지만 파트너 관계 전체에는 하나의 전역 SiteMinder 커넥터 구성만 적용됩니다. "배포 설정"에서 확인란을 선택하고 구성이 정의되어 있는 경우에만 커넥터를 사용할 수 있습니다. "배포 설정"은 UI 의 "인프라" 탭에서 액세스할 수 있습니다. 커넥터를 전역으로 사용하도록 설정하면 CA SiteMinder?Federation Standalone 이 파트너 관계 구성을 평가하여 커넥터가 사용되는지 여부를 확인합니다. 파트너 관계에는 전역 커넥터 구성이 사용됩니다.

파트너 관계에서 커넥터를 사용하지 않으려면 파트너 관계 수준에서 확인란의 선택을 취소하십시오. 전역 수준에서 커넥터를 사용하지 않으려면 "배포 설정"에서 해당 설정을 해제하십시오.

중요! 전역 수준에서 커넥터를 사용하지 않도록 설정하면 CA SiteMinder?Federation Standalone 의 파트너 관계 수준에서 확인란이 무시됩니다.

신뢰 당사자 측에서 사용자 ID 구성

신뢰 당사자가 로컬 사용자 디렉터리에서 사용자를 찾을 수 있도록 사용자 ID 를 구성해야 합니다.

다음 단계를 수행하십시오.

참고: "도움말"을 클릭하면 필드, 컨트롤 및 해당되는 각 요구 사항에 대한 설명을 볼 수 있습니다.

1. 다음 특성 중 하나를 선택합니다.
   - 이름 ID
   - 이전에 채워진 드롭다운 목록의 특성

     원격 어설션 특성이 특성을 포함한 메타데이터에 기반하여 생성된 경우 목록이 채워집니다.
   - 사용자가 입력하는 특성

     이 옵션은 메타데이터를 사용할 수 없고 원격 어설션 엔터티에 특성이 포함되지 않은 경우에 사용될 가능성이 많습니다.
   - Xpath 경로

2. (선택 사항 - SAML 2.0 만) "IDP 가 사용자 식별자를 만들도록 허용"을 선택합니다.

   이 특성은 어설션 당사자 측에서 이 기능을 사용할 수 있는 경우 어설션 당사자가 NameID 에 대한 새 값을 생성하도록 지시합니다. 어설션 당사자 측에서 구성하는 이름 ID 형식은 영구 식별자여야 합니다. NameID 의 새 값은 어설션 당사자가 신뢰 당사자에게 돌려 보내는 어설션에 포함됩니다.

3. LDAP 또는 ODBC 검색 사양을 지정합니다. 두 디렉터리가 모두 있는 경우 각각의 검색 사양을 구성합니다.

   LDAP 예제

   ou=%s,o-ca

   ODBC 예제

   name=%s

   ODBC 검색 지정 필드에서 검색 문자열의 %s 를 대체하는 사용자 저장소의 값에 등호(=)를 포함할 수 있습니다. 값에 등호가 포함되어 있으면 항목 맨 앞에 **user=** 값을 추가하십시오. 예를 들어 사용자 저장소에서 ElectronicMail 의 값이 CN=catechnologies 인 경우 ODBC 검색 지정 필드에 **user=ElectronicMail=%s** 를 입력하십시오. user=를 추가하면 정책 엔진이 문자열을 올바르게 해석할 수 있습니다.

4. (선택 사항) SiteMinder 커넥터 설정을 구성합니다.

   a. CA SiteMinder® Federation Standalone 을 기존 SiteMinder 배포에 통합하는 경우 확인란을 선택하여 SiteMinder 커넥터를 사용하도록 설정합니다.

b. (선택 사항) CA SiteMinder® Federation Standalone 또는 SiteMinder 에서 유니버설 ID 를 사용하여 사용자 레코드를 가져오도록 "UserDN 및 디렉터리 이름 비교 적용"의 선택을 취소합니다. "유니버설 ID"를 사용하면 사용자 디렉터리가 물리적으로 다르고 유형이 다를 수 있습니다. "유니버설 ID"만으로도 가져온 사용자 레코드가 올바른 레코드인지 확인할 수 있습니다.

참고: "유니버설 ID"를 사용할 경우 각 사용자의 유니버설 ID 가 고유해야 합니다. 유니버설 ID 가 고유하지 않으면 사용자 레코드에 액세스하는 시스템이 잘못된 레코드를 가져올 수 있습니다.

확인란을 선택된 상태(기본값)로 두면 CA SiteMinder® Federation Standalone 과 SiteMinder 가 물리적으로 같은 디렉터리를 사용해야 합니다. 사용자가 저장소 조회를 수행하려면 두 디렉터리의 이름이 같아야 합니다. 사용자를 인증하는 엔터티는 사용자가 제공하는 정보를 사용자 레코드의 UserDN 및 "디렉터리 이름"과 비교합니다.

5. "다음"을 클릭하여 파트너 관계 구성을 계속합니다.

사용자 식별을 위한 AllowCreate 사용(SAML 2.0)

SAML 2.0 AllowCreate 기능은 SP 에서 "사용자 ID" 구성의 선택적 설정입니다. 인증 요청에 AllowCreate 특성을 포함하면 아이덴티티 공급자가 SP 에 대한 사용자 식별자를 생성할 수 있게 됩니다.

SP 는 아이덴티티 공급자에 인증 요청을 전송하여 싱글 사인온을 시작할 수 있습니다. 요청의 일환으로 서비스 공급자는 true 로 설정된 AllowCreate 특성을 포함할 수 있습니다. 서비스 공급자는 사용자의 아이덴티티를 가져오려고 합니다. AuthnRequest 를 받으면 아이덴티티 공급자가 어설션을 생성합니다. 아이덴티티 공급자가 적절한 사용자 레코드에서 이름 ID 역할을 하는 어설션 특성을 검색합니다. 아이덴티티 공급자가 NameID 특성에 대한 값을 찾을 수 없으면 NameID 에 대한 고유한 영구 식별자를 생성합니다. 아이덴티티 공급자에서 "허용/만들기" 기능을 활성화하여 식별자를 생성하도록 합니다. 아이덴티티 공급자는 고유한 식별자가 있는 어설션을 SP 로 반환합니다.

AllowCreate 쿼리 매개 변수가 사용되도록 설정하여 AllowCreate 특성의 값을 대체할 수 있습니다. 쿼리 매개 변수를 사용하면 파트너 관계를 비활성화하고, 편집하여 다시 활성화하지 않아도 구성된 AllowCreate 설정을 무시할 수 있습니다. 쿼리 매개 변수는 기능을 더욱 유연하게 구현할 수 있게 만듭니다.

제 10 장: 어설션 당사자에서의 어설션 구성

이 섹션은 다음 항목을 포함하고 있습니다.

어설션 구성

파트너 관계 마법사의 "어설션 구성" 단계에서는 다음 설정의 구성을 정의합니다.

이름 ID

필수 어설션 특성인 "이름 ID" 특성은 사용자를 고유한 방식으로 식별합니다. "이름 ID 형식"은 페더레이션된 파트너가 지원하는 식별자 유형을 나타냅니다. "이름 ID 유형"은 이름 ID 형식과 연결된 사용자 프로필 특성을 지정합니다. 사용자 프로필 특성은 사용자 저장소 또는 세션 저장소에서 가져옵니다.

어설션 특성

서블릿, 웹 응용 프로그램 또는 기타 사용자 지정 응용 프로그램에서는 특성을 사용하여 사용자 지정 콘텐츠를 표시하거나 다른 사용자 지정 기능이 사용되도록 설정할 수 있습니다. 특성은 웹 응용 프로그램에서 사용되는 경우 사용자가 신뢰 당사자 측에서 수행하는 작업을 제한할 수 있습니다. 예를 들어 "Authorized Amount"(권한 부여된 금액)이라는 특성 변수는 사용자가 신뢰 당사자 측에서 소비할 수 있는 최대 금액(달러)으로 설정됩니다.

특성은 <AttributeStatement> 요소나 <EncryptedAttribute> 요소에서 지정됩니다. 특성은 이름/값 쌍의 형식을 사용합니다. 특성을 HTTP 헤더나 HTTP 쿠키로 제공할 수도 있습니다.

참고: 특성 명령문은 어설션에 필요하지 않습니다.

특성 명령문 하나에 여러 유형의 특성을 구성할 수 있습니다. 특성 유형은 다음과 같습니다.

- 사용자 특성
- DN 특성
- 정적 데이터
- 세션 특성

  세션 특성은 어설션이 세션 저장소에 있는 경우에만 어설션에 사용할 수 있습니다.

어설션 특성을 변환하는 식을 구성할 수도 있습니다. 이 기능을 클레임 변환이라고 합니다.

어설션을 받는 경우 신뢰 당사자는 특성 값을 응용 프로그램에 제공합니다.

어설션 생성기 플러그인

일반적으로 특성은 사용자 디렉터리 레코드에서 가져오지만 어설션에는 외부 데이터베이스 또는 응용 프로그램 콘텐츠와 같은 다른 출처의 특성이 포함될 수 있습니다. 다양한 출처에서 특성을 가져오는 어설션 생성기 플러그인을 작성할 수 있습니다. 어설션 생성기 플러그인은 어설션 생성기 플러그인 인터페이스에 따라 작성하는 사용자 지정 코드입니다.

플러그인 작성에 대한 자세한 내용은 *Programming Guide for the Federation Java SDK*(Federation Java SDK 프로그래밍 안내서)를 참조하십시오.

어설션 옵션 구성

어설션 당사자에서 어설션 옵션을 구성하십시오.

다음 단계를 수행하십시오.

1. 파트너 관계 마법사의 "어설션 구성" 단계로 이동합니다.
2. "이름 ID" 섹션에서 설정을 구성합니다.

   신뢰 당사자는 이러한 값을 사용하여 어설션에서 이름 ID 값을 해석합니다.

   선택한 "이름 ID 유형" 옵션에 따라 올바른 값으로 입력을 완료합니다.

   정적 특성

   "값" 필드에 임의 상수 문자열을 입력하십시오.

   사용자 특성

   "값" 필드에 올바른 사용자 저장소 특성을 입력하십시오. 예: 메일

   세션 특성

   "값" 필드에 올바른 세션 저장소 특성을 입력하십시오.

   DN 특성(LDAP 에만 해당)

   "값" 필드에 올바른 LDAP 사용자 디렉터리 특성을 입력하십시오. 또한, DN 사양 필드에 올바른 DN 을 입력하십시오. 예를 들어 DN 특성은 cn=JaneDoe 이고 사양은 ou=Engineering,o=example.com 입니다.

   참고: "도움말"을 클릭하면 필드, 컨트롤 및 해당되는 각 요구 사항에 대한 설명을 볼 수 있습니다.

3. (선택 사항 - SAML 2.0 만) 어설션 당사자가 "이름 ID"의 값을 생성할 수 있도록 "사용자 식별자의 생성 허용"을 선택합니다. 이 기능이 작동하려면 신뢰 당사자의 AuthnRequest 가 AllowCreate 특성을 포함해야 합니다.

   참고: 이 옵션을 선택하는 경우 "이름 ID 형식"의 값이 "영구 식별자"여야 합니다.

4. (선택 사항) "어설션 특성" 테이블에서 "행 추가"를 클릭하여 어설션에 대한 특성을 하나 이상 지정합니다. 원하는 경우 특성을 암호화할 수 있습니다.

   테이블 작성에 대한 도움이 필요하면 몇 가지 어설션 특성 예를 확인하십시오. 특성 테이블의 각 열에 대한 자세한 내용을 보려면 "도움말"을 클릭하십시오.

   참고: LDAP 사용자 저장소 특성의 경우 어설션에 다중값 사용자 특성을 추가할 수 있습니다. 다중값 사용자 특성을 지정하는 방법은 "도움말"에 설명되어 있습니다.

5. (선택 사항) CA SiteMinder® Federation Standalone Java SDK 를 사용하여 어설션 생성기 플러그인을 작성한 경우 "어설션 생성기 플러그인" 섹션에서 필드에 데이터를 입력합니다.

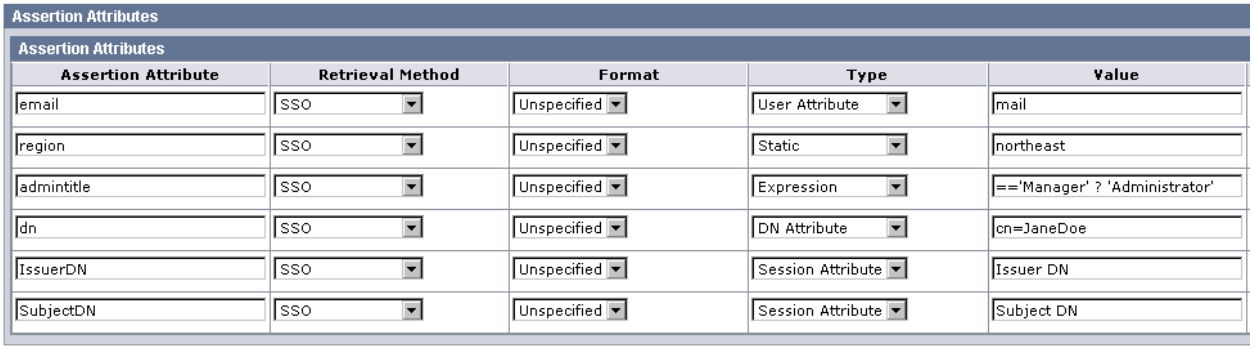
   플러그인을 작성하려면 *Programming Guide for the Federation Java SDK*(Federation Java SDK 프로그래밍 안내서)를 참조하십시오.

6. "다음"을 클릭하여 파트너 관계 구성을 계속합니다.

어설션 특성 구성 예

다음 그림에서는 어설션 특성 항목의 몇 가지 예를 보여 줍니다. 이 화면은 SAML 2.0 파트너 관계에 해당합니다. SAML 1.1 화면은 이와 비슷하지만 "검색 방법" 및 "형식" 열이 없고, 대신 "네임스페이스" 열이 있습니다.

참고: DN 특성 예에는 항목이 ou=Engineering,o=example.com 인 "DN 사양" 열이 포함됩니다. 이 열은 이 그림에 표시되어 있지 않습니다.



세션 특성을 어설션에 추가하는 방법

정책 서버는 사용자가 인증된 후 동적 사용자 정보를 유지하기 위해 세션 저장소를 사용합니다. 예를 들면 인증 컨텍스트 정보, SAML 특성, 사용자를 인증하는 타사 IdP, OAuth 인증의 클레임과 같은 정보가 저장됩니다. 정책 서버는 이 정보를 사용하여 사용자 토큰을 생성하거나 정책을 결정할 수 있습니다.

페더레이션된 싱글 사인온의 경우 정책 서버에서 세션 저장소의 특성을 어설션에 추가하여 요청된 응용 프로그램을 사용자 지정할 수 있습니다.

세션 특성은 다음과 같은 배포의 경우에 저장됩니다.

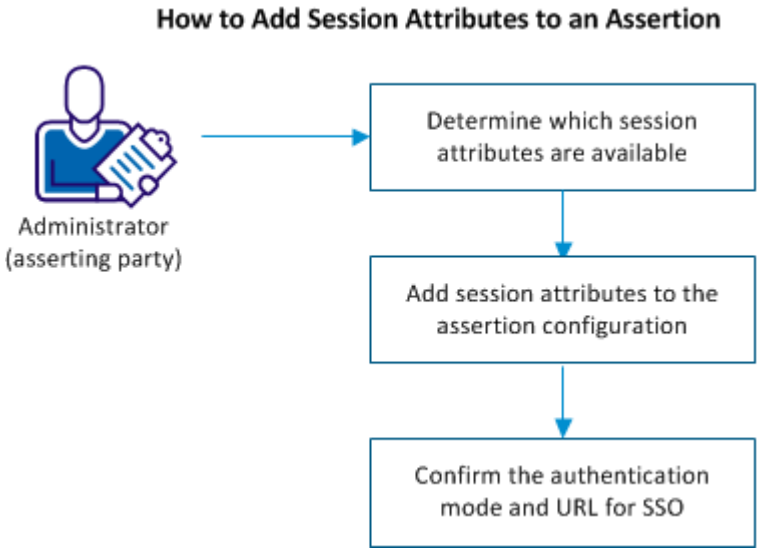
- 위임되지 않은 인증 배포

  로컬 시스템 또는 외부 타사에서 사용자를 인증하지만 시스템에서는 이를 로컬 인증으로 간주합니다. 로컬 인증 배포의 경우 싱글 사인온 구성에서 인증 모드가 로컬로 설정되어야 합니다. 또한 액세스 정책으로 인증 URL 을 보호해야 합니다. 액세스 정책의 인증 체계는 세션 특성을 유지하도록 구성됩니다.

- 위임된 인증 배포

  외부 타사에서 사용자를 인증할 수 있습니다. 타사 파트너는 세션 저장소에 저장되는 사용자 정보를 반환합니다.

다음 그림에서는 세션 특성을 구성한 후 어설션에 추가하는 데 필요한 단계를 보여 줍니다.



세션 특성을 지원하려면 다음 단계를 완료하십시오.

1. 사용할 수 있는 세션 특성을 확인합니다. (페이지 192)
2. 세션 특성을 어설션 구성에 추가합니다. (페이지 192)
3. SSO 에 대한 인증 모드 및 URL 을 확인합니다.

사용할 수 있는 세션 특성 확인

페더레이션 관리자는 파트너 관계에 사용되는 세션 특성을 식별해야 합니다. 사용 가능한 특성을 쉽게 파악할 수 있도록 데이터베이스나 사용자 디렉터리 같은 인증 원본을 사용하십시오.

세션 특성을 어설션 구성에 추가

세션 특성을 어설션 구성에 추가합니다. 구성은 IdP-SP 파트너 관계 같은 어설션 당사자 측에서 수행됩니다.

다음 단계를 수행하십시오.

1. Administrative UI 에 로그인합니다.
2. 파트너 관계 마법사의 "어설션 구성" 단계로 이동합니다.
3. "어설션 특성" 섹션에서 "행 추가"를 클릭합니다.

4. 세션 특성을 구성하려면 테이블 설정을 완료합니다. 예를 들면 다음과 같습니다.

   어설션 특성

   IssuerID

   검색 방법

   SSO

   형식

   지정되지 않음

   유형

   세션 특성

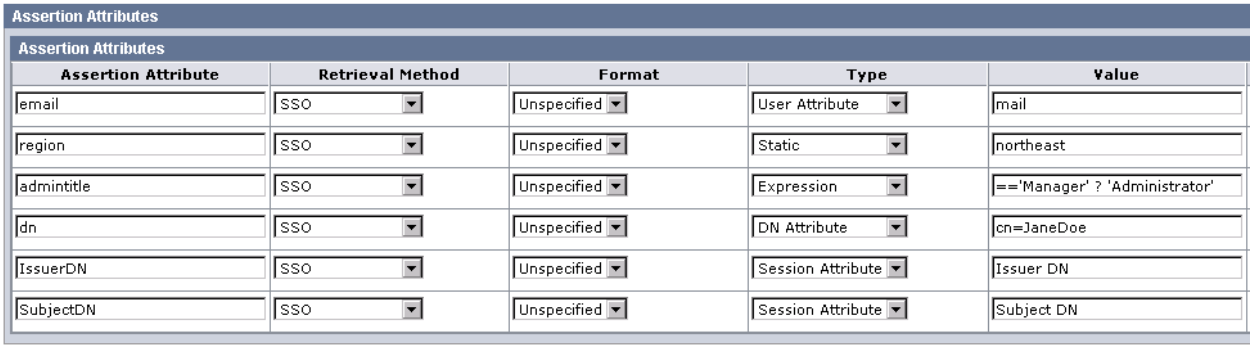
   값

   IssuerID

   특성 테이블에 대한 자세한 내용을 보려면 "도움말"을 클릭하십시오.
5. 필요한 항목 수에 맞춰 행을 추가합니다.
6. (선택 사항) 특성을 암호화하려면 "암호화"를 선택합니다.
7. "다음"을 클릭하여 "SSO 및 SLO" 단계로 이동합니다.

Administrative UI 의 세션 특성 예

다음 그림의 마지막 두 개 항목은 세션 특성 항목의 예를 보여 줍니다. 이 화면은 SAML 2.0 파트너 관계에 해당합니다. SAML 1.1 화면은 이와 비슷하지만 "검색 방법" 및 "형식" 열이 없고, 대신 "네임스페이스" 열이 있습니다.



SSO 에 대한 인증 모드 및 URL 확인

파트너 관계의 인증 모드 및 인증 URL 이 올바르게 설정되어 있는지 확인하십시오.

참고: 이 절차에서는 필요한 다른 SSO 설정이 구성되어 있다고 가정합니다.

다음 단계를 수행하십시오.

1. 파트너 관계 마법사의 "SSO 및 SLO" 단계로 이동합니다.
2. "인증" 섹션에서 다음 필드의 설정을 확인합니다.

   인증 모드

   로컬

   인증 URL

   이 URL 은 redirect.jsp

   파일(예: http://*myserver*.idpA.com/siteminderagent/redirectjsp/redirect.jsp)을 가리킵니다.

   myserver

   웹 에이전트 옵션 팩 또는 SPS 페더레이션 게이트웨이가 포함된 웹 서버를 식별합니다. redirect.jsp 파일은 어설션 당사자에 설치된 웹 에이전트 옵션 팩 또는 SPS 페더레이션 게이트웨이에 포함되어 있습니다.
3. "확인" 단계로 이동하고 "마침"을 클릭합니다.

어설션 당사자에서 클레임 변환을 구성하는 방법

클레임 변환 기능은 페더레이션된 싱글 사인온 트랜잭션 동안 클레임을 조작합니다. 클레임은 특성이라고도 하며, 특성을 사용자 지정하고 파트너의 사용자 환경을 개선하는 데 유용합니다.

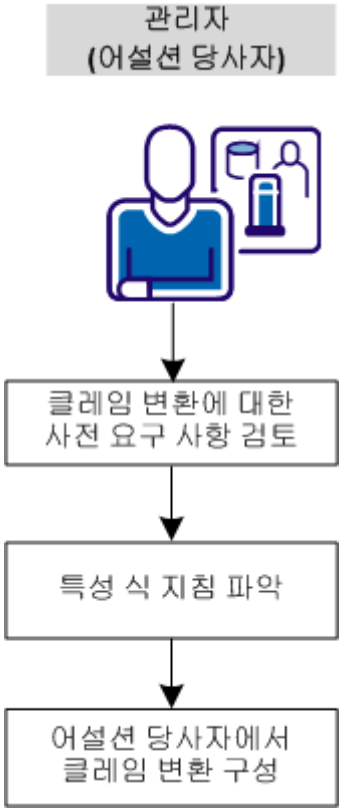
어설션 특성을 수정하면 신뢰 당사자는 대상 응용 프로그램에서 사용할 수 있도록 사용자 정보를 조정할 수 있게 됩니다. 예를 들어 클레임 변환을 통해 서로 다른 도메인에 있는 서로 다른 파트너에서의 역할을 연결할 수 있습니다. 한 도메인에서 사용자가 엔지니어링 관리자이고 EngineerAdmins 라는 그룹에 속하지만, 신뢰 당사자는 동일한 역할을 DevelAdmins 로 식별할 수 있습니다. 이 경우 어설션 당사자는 어설션을 발급하기 전에 역할 특성을 변경합니다. 그러면 해당 사용자가 신뢰 당사자 응용 프로그램에서 인식될 수 있는 DevelAdmins 역할로 식별됩니다.

클레임 변환은 로컬 어설션 당사자에서 어설션 생성 프로세스 도중에 발생합니다. 이 기능은 파트너 관계별로 구성해야 합니다. 어설션을 로컬 당사자가 생성하든 원격 당사자가 생성하든 관계없이 어설션을 수정할 수 있습니다. 클레임은 파트너 관계에 대해 구성된 식을 기반으로 변환됩니다. 식에서는 사용자 저장소와 SiteMinder 세션 저장소에서 가져온 사용자 정보를 사용합니다.

소프트웨어에서는 어설션 특성에 대해 다음과 같은 세 가지 수정 작업을 수행할 수 있습니다.

- **변환:** 어설션 특성의 값을 다른 값으로 변경합니다.
- **추가:** 어설션 특성이 아직 없는 경우 추가합니다.
- **삭제:** 조건에 따라 어설션 특성을 삭제합니다.

다음 그림에서는 구성 단계를 보여 줍니다.



클레임 변환을 설정하려면 다음 단계를 수행하십시오.

1. 클레임 변환을 위한 사전 요구 사항을 검토합니다. (페이지 196)
2. 특성 식 지침을 확인합니다 (페이지 197).
3. 어설션 당사자에서 클레임 변환을 구성합니다 (페이지 198).

클레임 변환에 대한 사전 요구 사항

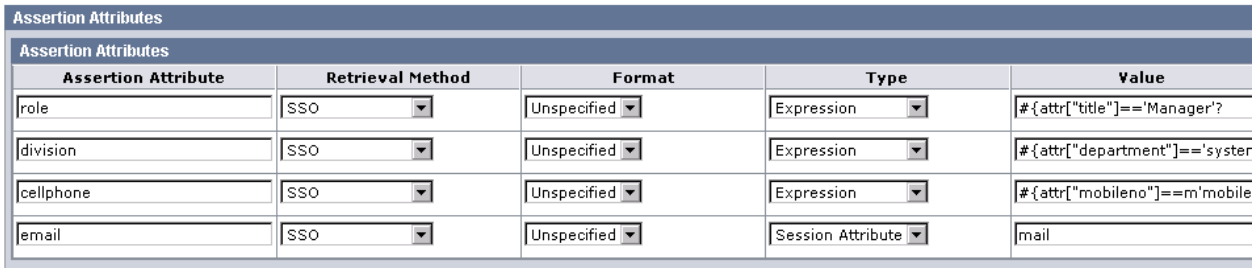
클레임 변환을 구성하기 전에 다음 사전 요구 사항을 고려하십시오.

- 사용할 수 있는 사용자 저장소 및 세션 저장소 특성을 잘 알고 있어야 합니다.
- 신뢰 당사자가 어설션에서 검색할 것으로 예상되는 특성을 확인해야 합니다.
- UEL(Unified Expression Language)의 오픈 소스 버전인 JUEL(Java Unified Expression Language)을 잘 알고 있어야 합니다.

특성 식 지침 확인

식은 소프트웨어에 어설션 특성의 조작 방법을 알려 주는 규칙입니다. 식이 전달하는 지침을 통해 소프트웨어에서 어설션 특성을 수정, 추가 또는 삭제하게 됩니다. JUEL(Java Unified Expression Language)을 사용하여 식을 구성하십시오. JUEL 식 계산기는 구성된 식을 검사하고 결과 어설션 특성을 생성합니다.

Administrative UI 의 "어설션 특성" 테이블에서 식을 정의하십시오. 이 테이블에 액세스하려면 파트너 관계 마법사의 "어설션 구성" 단계로 이동하십시오. 이 테이블은 다음 그림에 표시되어 있습니다.



어설션 특성 테이블의 "Value"(값) 열에 식을 입력하십시오. 식의 모든 특성은 사용자 저장소 또는 세션 저장소 특성입니다.

일반적으로 식은 조건에 따라 작동합니다. 조건이 충족되면 지정된 클레임 수정이 수행됩니다. 예를 들어 수신 어설션에 "role" 특성이 포함된 경우 "role" 어설션 특성을 수정하는 식은 다음과 같습니다.

#{attr["title"] == 'manager' ? 'administrator' : attr["title"]}

표현식의 첫 번째 부분 **#{attr["title"] == 'manager'**는 소프트웨어로 하여금 로그인한 사용자의 직책이 "manager"인지 확인하도록 합니다. 조회는 사용자 디렉터리에서 수행됩니다. 이 조건이 충족되면 표현식의 두 번째 부분 **? 'administrator' :**가 role 어설션 특성에 값 "administrator"를 할당합니다. 조건이 충족되지 않을 경우 식의 마지막 부분 **attr["title"]}**은 사용자 특성 "title"의 값이 "manager"로 유지되도록 합니다. 어설션 특성 "role"에는 이 "manager" 값이 할당됩니다.

참고: 식에서 attr["title"] 구문 대신 앞의 예에 나온 'administrator'와 같이 정적 값을 사용할 수 있습니다.

이 예에서는 어설션에 "role" 특성이 이미 있다고 가정합니다. 따라서 이 식은 기존 특성을 변환하는 식입니다. "role"이 어설션에 포함되지 않은 경우 소프트웨어는 role 특성을 어설션에 추가합니다.

식 구문

다음과 같은 올바른 구문을 사용하여 식을 구성하십시오.

- 사용자 저장소 특성은 **attr**["*attribute_name*"] 문자열로 나타냅니다.
- 세션 저장소 특성은 **session_attr**["*attribute_name*"] 문자열로 나타냅니다.
- 클레임을 삭제하려면 'DELETE' 인수를 사용합니다.

attr 및 **session_attr** 접두사에는 소문자 텍스트를 사용합니다. 특성 이름은 대/소문자를 구분하지 않습니다.

또한 다음과 같은 조건부 JUEL 연산자에 대해 잘 알고 있어야 합니다.

연산자	의미
conditional value ? value1 : value2	conditional value 가 value1 또는 value2 가 됩니다.
!=	같지 않음
==	같음

중요! 식의 특성은 사용자 디렉터리나 세션 저장소에서 사용할 수 있는 특성이어야 합니다. 특성이 올바르지 않으면 시스템에서는 단순히 해당 특성에 빈 값을 포함합니다. 어설션 생성은 오류 없이 실행됩니다.

더 많은 식의 예를 보려면 어설션 당사자 측에서 클레임 변환 구성 단원을 읽어 보십시오.

어설션 당사자 측에서 클레임 변환 구성

파트너 관계 수준에서 식을 정의하십시오. 이러한 식의 결과로 어설션에서 특성이 수정, 추가 또는 삭제됩니다. 규칙이 정의된 후에는 어설션이 수정되어 신뢰 당사자로 보내집니다. 클레임 변환을 구성하지 않으면 어설션 특성이 신뢰 당사자에게 "그대로" 전달됩니다.

다음 단계를 수행하십시오.

1. Administrative UI 에 로그인합니다.
2. "페더레이션", "파트너 관계"를 클릭합니다.

3. 수정할 파트너 관계를 선택합니다. 적합한 파트너 관계로는 다음이 포함됩니다.
   - 로컬 생산자-원격 소비자
   - 로컬 IdP-원격 SP
   - 로컬 IP-원격 RP
4. 파트너 관계 마법사의 "어설션 구성" 단계로 이동합니다.

   "어설션 특성" 섹션에서 "행 추가"를 클릭합니다.
5. 추가한 행에서 다음 필드에 특히 주의해야 합니다. 각 필드에 대한 자세한 설명을 보려면 "도움말"을 클릭합니다.

   어설션 특성

   어설션 특성을 입력합니다. 이 열의 모든 값은 어설션 특성입니다. 어설션의 기존 특성은 어설션에 유지되지만 해당 값은 구성된 식에 따라 새로 설정됩니다. DELETE 식을 구성한 경우에만 어설션에서 특성이 제거됩니다.

   검색 방법

   기본값 SSO 를 유지합니다.

   형식

   어설션에 추가되는 특성의 형식을 지정합니다. 형식 옵션은 엔터티에 대한 SAML 프로필에 따라 다릅니다.

   유형

   식

   클레임 변환에는 항상 이 값을 사용합니다.

   값

   어설션 특성을 수정할 방법을 반영하는 식을 입력합니다.

   클레임 식 구성에 대한 지침과 다음 예를 검토하십시오.
   - 어설션의 클레임 변환 (페이지 200)
   - 어설션에 클레임 추가 (페이지 202)
   - 어설션에서 클레임 삭제 (페이지 204)

6. (SAML 2.0 및 토큰 유형이 SAML 2.0 인 WSFED 의 경우 선택 사항) 어설션 특성을 암호화하려면 "암호화"를 선택합니다. 어설션 당사자는 파트너 관계 구성에 지정된 인증서를 사용하여 어설션을 암호화합니다.

   신뢰 당사자는 인증서와 연결된 개인 키를 사용하여 어설션 특성의 암호를 해독합니다.

7. 구성하려는 어설션 특성에 필요한 만큼 행을 추가합니다.

클레임 변환이 파트너 관계에 구성된 항목을 기준으로 구현됩니다.

어설션의 클레임 변환

클레임을 변환하면 어설션 특성 값이 다른 값으로 변경됩니다.

참고: 아래 예에서는 "어설션 특성", "유형" 및 "값"에 대한 항목만 보여 줍니다.

변환 예 1

다음 예에서는 어설션에 "title" 특성이 이미 있다고 가정합니다. 표에는 사용자 저장소의 사용자 특성이 나와 있습니다.

사용자 디렉터리 특성	특성 값
role	admin
admintitle	SeniorAdmin
supertitle	SuperUser

다음 구성을 사용하여 기존 title 특성의 값을 변환할 수 있습니다.

어설션 특성

title

유형

식

값

#{attr["role"] == 'admin' ? attr["admintitle"] : attr["supertitle"]}

결과: 이 식은 "role" 사용자 특성이 "admin"으로 설정되어 있는지 여부를 조건으로 합니다. 이 조건이 충족될 경우 어설션 특성 "title"이 "admintitle" 특성 값 SeniorAdmin 으로 설정됩니다. role 특성이 "admin"이 아닌 다른 값으로 설정되어 있으면 "title" 특성은 "supertitle" 특성 값인 SuperUser 가 됩니다.

변환 예 2

다음 예에서는 어설션에 ContactNo 특성이 이미 있다고 가정합니다.

사용자 디렉터리 특성	특성 값
homephone	555-3344
mobile	555-8888

다음 구성을 사용하여 기존 title 특성의 값을 변환할 수 있습니다.

어설션 특성

ContactNo

유형

식

값

#{attr["homephone"] == '555-3344' ? attr["mobile"] : attr["homephone"]}

결과: 이 식은 로그인한 사용자의 "homephone" 사용자 특성이 555-3344 로 설정되어 있는지 여부를 조건으로 합니다. 이 조건이 충족될 경우 어설션 특성은 "mobile" 특성 값인 555-8888 로 설정됩니다. 조건이 충족되지 않으면 "homephone" 값이 변경되지 않습니다.

참고: 세션 특성을 사용하는 식을 구성하려면 attr["*attribute_name*"]을 session_attr["*attribute_name*"]로 바꾸십시오. 예:

#{session_attr["att1"] == 'admin' ? session_attr["attr2"] : attr["attr3"]}

어설션에 클레임 추가

아직 없는 어설션 특성을 추가할 수 있습니다.

추가 예 1

다음 예에서는 어설션에 "title" 특성이 *없다고* 가정합니다.

사용자 디렉터리 특성	특성 값
role	admin
admintitle	director
supertitle	executive

다음 구성으로 어설션에 title 특성을 추가할 수 있습니다.

어설션 특성

title

유형

식

값

#{attr["role"] == 'admin' ? attr["admintitle"] : attr["supertitle"]}

결과: 이 식은 로그인한 사용자의 "role" 특성이 admin 으로 설정되어 있는지 여부를 조건으로 합니다. 이 조건이 충족될 경우 어설션 특성 "title"이 어설션에 추가되고 "admintitle" 특성 값인 "director" 값으로 설정됩니다. role 특성이 "admin"이 아닌 다른 값으로 설정되어 있으면 어설션 특성 "title"이 추가되지만 해당 값은 "supertitle" 특성 값인 "executive"가 됩니다.

추가 예 2

다음 예에서는 어설션에 "smtitle" 특성이 *없다고* 가정합니다.

사용자 디렉터리 특성	특성 값
title	관리자

어설션 특성

smtitle

유형

식

값

#{attr["title"] == 'manager' ? 'federation administrator' : attr["title"]}

결과: 로그인한 사용자의 title 특성이 "manager"인 경우 "smtitle"이 어설션에 추가되고 해당 값은 "federation administrator"로 설정됩니다. 물음표 뒤에는 attr["attribute_name"] 구문을 사용하는 대신 정적 값을 입력할 수 있습니다. 이 예에서는 federation administrator 가 정적 값에 해당합니다.

참고: 세션 특성을 사용하는 식을 구성하려면 attr["*attribute_name*"]을 session_attr["*attribute_name*"]로 바꾸십시오. 예:

#{session_attr["att1"] == 'admin' ? session_attr["attr2"] : attr["attr3"]}

어설션에서 클레임 삭제

어설션 특성을 삭제할 수 있습니다.

삭제 예 1

두 개의 항목을 구성하여 어설션 특성 admintitle 및 supertitle 을 삭제할 수 있습니다.

사용자 디렉터리 특성	특성 값
role	admin 또는 superuser
title	administrator
su	superuser

어설션 특성

admintitle

유형

식

값

#{attr["role"] == 'superuser' ? 'DELETE' : attr["title"]}

결과: 이 식 문자열은 "role" 사용자 특성을 조건으로 합니다. 로그인한 사용자의 role 특성이 superuser 면 어설션 특성 "admintitle"이 삭제되고, role 특성이 superuser 가 아니면 title 어설션 특성 값이 title 사용자 디렉터리 특성의 값인 administrator 로 설정됩니다.

어설션 특성

supertitle

유형

식

값

#{attr[“role”] == ‘admin’ ? 'DELETE' : attr["su"]}

결과: 이 식 문자열은 "role" 사용자 특성을 조건으로 합니다. 로그인한 사용자의 role 특성이 "admin"이면 어설션 특성 "supertitle"이 삭제되고, role 특성이 "admin"이 아니면 supertitle 어설션 특성 값이 su 사용자 디렉터리 특성의 값인 superuser 로 설정됩니다.

삭제 예 2

다음 예에서는 식 하나로 추가와 삭제를 결합하는 경우를 보여 줍니다.

사용자 디렉터리 특성	특성 값
title	관리자

어설션 특성

ManagerName

유형

식

값

#{attr["title"] != 'Manager' ? attr["manager"] : 'DELETE'}

결과: 로그인한 사용자의 사용자 특성 title 이 "manager"가 *아니면* ManagerName 특성이 어설션에 추가됩니다. 하지만 로그인한 사용자의 title 이 manager 이면 ManagerName 특성은 어설션에 포함된 것으로 가정하여 삭제됩니다.

참고: 세션 특성을 사용하는 식을 구성하려면 attr["*attribute_name*"]을 session_attr["*attribute_name*"]로 바꾸십시오. 예:

#{session_attr["att1"] == 'admin' ? session_attr["attr2"] : attr["attr3"]}

어설션 콘텐츠 사용자 지정

어설션 생성기 플러그인을 사용하여 어설션 콘텐츠를 수정할 수 있습니다. 플러그인을 통해 사용자와 파트너 및 공급업체 간의 비즈니스 계약을 기반으로 어설션 콘텐츠를 사용자 지정할 수 있습니다. 플러그인은 각 파트너에 대해 하나씩 허용됩니다.

어설션 생성기 플러그인 구성은 여러 단계로 이루어집니다.

1. 아직 설치하지 않은 경우 CA SiteMinder® Federation Standalone SDK 를 설치합니다.
2. CA SiteMinder® Federation Standalone SDK 의 일부인 AssertionGeneratorPlugin.java 인터페이스를 구현합니다.
3. 어설션 생성기 플러그인 구현 클래스를 배포합니다.
4. Administrative UI 에서 어설션 생성기 플러그인의 매개 변수를 구성합니다.

AssertionGeneratorPlugin 인터페이스 구현

사용자 지정 어설션 생성기 플러그인을 생성할 때의 첫 번째 단계는 AssertionGeneratorPlugin 인터페이스를 구현하는 것입니다. 다음 요구 사항이 구현 클래스에 적용됩니다.

- 구현에서는 매개 변수가 포함되지 않은 공개 기본 생성자 메서드를 제공해야 합니다.
- 구현은 상태 비저장이어야 합니다. 따라서 여러 스레드가 단일 플러그인 클래스를 사용할 수 있어야 합니다.
- 구현에 customizeAssertion 메서드 호출이 포함되어야 합니다. 요구 사항에 따라 이러한 메서드의 기존 구현을 덮어쓸 수 있습니다. 샘플 프로그램을 참조하십시오.
- customizeAssertion 메서드에 전달되는 매개 변수 문자열의 구문 요구 사항과 사용에 대한 책임은 사용자 지정 개체에 있습니다.

참고: *federation_sdk_home*\sample\com\ca\federation\sdk\plugin\sample 폴더에 두 개의 샘플 구현 클래스가 포함되어 있습니다.

어설션 생성기 플러그인 배포

AssertionGeneratoPlugin 인터페이스에 대한 구현 클래스를 코드로 지정한 다음에는 해당 구현 클래스를 컴파일하고 CA SiteMinder?Federation Standalone 이 실행 파일을 찾을 수 있는지 확인하십시오.

다음 단계를 수행하십시오.

1. 다음 방법 중 하나로 어설션 플러그인 코드를 컴파일합니다.
   - 샘플 플러그인을 사용하는 경우 플랫폼의 빌드 스크립트를 사용하여 플러그인을 컴파일합니다. 빌드 스크립트는 *federation_sdk_home*\sample 디렉터리에 설치됩니다. 빌드 스크립트는 다음과 같습니다.

     Windows: build_plugin.bat

     UNIX: build_plugin.sh

     컴파일된 샘플 플러그인 fedpluginsample.jar 은 *federation_sdk_home*\jar 디렉터리에 있습니다.
   - 플러그인을 직접 작성하는 경우에는 플러그인을 컴파일할 때 smapi.jar 를 포함하십시오.
2. JVMOptions.txt 파일에서 플러그인의 클래스 경로를 포함하도록 -Djava.class.path 값을 수정합니다. *federation_install_dir*\siteminder\config 디렉터리에서 JVMOptions.txt 파일을 찾습니다.

   플러그인 jar 를 원하는 디렉터리에 저장하고 JVMOptions.txt 파일에서 해당 위치를 가리키도록 할 수 있습니다. 샘플 플러그인을 사용하려면 fedpluginsample.jar 를 가리키도록 클래스 경로를 수정해야 하지만 smapi.jar 의 클래스 경로는 수정하지 마십시오.

   참고: Apache Xerces 또는 Xalan 을 플러그인에 사용하려면 제품과 함께 설치된 Xerces 또는 Xalan 바이너리 파일을 사용하십시오. 바이너리는 페더레이션 SDK 로는 설치되지 않습니다. 이 파일은 호환성을 위해 필요합니다.
3. CA SiteMinder® Federation Standalone 서비스를 다시 시작합니다.

   서비스를 다시 시작하면 CA SiteMinder® Federation Standalone 이 최신 버전의 어설션 생성기 플러그인을 사용합니다.

어설션 생성기 플러그인이 사용되도록 설정

어설션 생성기 플러그인을 작성하고 컴파일한 후 Administrative UI 에서 설정을 구성하여 플러그인이 사용되도록 설정합니다. UI 매개 변수는 CA SiteMinder?Federation Standalone 에게 플러그인을 찾을 수 있는 위치를 알려 줍니다.

플러그인을 배포할 때까지 플러그인 설정을 구성하지 마십시오.

다음 단계를 수행하십시오.

1. Administrative UI 에 로그온합니다.
2. 수정하려는 파트너 관계에 대한 파트너 관계 마법사의 "어설션 구성" 단계로 이동합니다.
3. 다음 어설션 생성기 플러그인 설정의 값을 입력합니다.

   플러그인 클래스

   플러그인의 Java 클래스 이름을 지정합니다. 이름을 입력합니다. 이 플러그인은 런타임에 호출됩니다.

   예: com.mycompany.assertiongenerator.AssertionSample

   플러그인 클래스는 어설션을 구문 분석 및 수정한 다음 최종 처리를 위해 결과를 CA SiteMinder® Federation Standalone 으로 반환할 수 있습니다. 각 신뢰 당사자의 어설션 생성기 플러그인을 지정합니다. SDK 에 컴파일된 샘플 플러그인이 포함되어 있습니다. 컴파일된 샘플 어설션 플러그인은 *federation_sdk_home*/jar 디렉터리에서 볼 수 있습니다.

   참고:
   federation_sdk_home\sample\com\ca\federation\sdk\plugin\sample 디렉터리에서 CA SiteMinder® Federation Standalone 샘플 플러그인의 소스 코드도 볼 수 있습니다.

   플러그인 매개 변수

   (선택 사항) CA SiteMinder® Federation Standalone 이 런타임에 플러그인에 매개 변수로 전달하는 문자열을 지정합니다. 이 문자열에는 어떤 값이든 포함될 수 있으며 따라야 하는 특정 구문이 없습니다.

   플러그인은 수신하는 매개 변수를 해석합니다. 예를 들어 매개 변수는 특성 이름일 수도 있고 플러그인에 작업을 수행하도록 지시하는 정수가 문자열에 포함될 수도 있습니다.

참조 정보(메서드 서명, 매개 변수, 반환 값, 데이터 형식)와 UserContext 클래스 및 APIContext 클래스에 대한 생성자는 *Javadoc 참조서*에서 확인할 수 있습니다. Javadoc 의 AssertionGeneratorPlugin 인터페이스를 참조하십시오.

제 11 장: 어설션 처리 사용자 지정(신뢰 당사자)

메시지 소비자 플러그인은 메시지 소비자 확장 API 를 구현하는 Java 프로그램입니다. 이 플러그인을 통해 어설션 거부, 상태 코드 반환 등의 어설션 처리를 위한 사용자 고유의 비즈니스 논리를 구현할 수 있습니다. 이 추가 처리는 어설션의 표준 처리와 함께 작동합니다.

인증이 진행되는 동안 시스템은 먼저 사용자를 로컬 사용자 저장소에 매핑하여 어설션을 처리하려고 합니다. CA SiteMinder?Federation Standalone 이 사용자를 찾을 수 없는 경우 메시지 소비자 플러그인의 postDisambiguateUser 메서드를 호출합니다.

플러그인이 사용자를 찾은 경우 인증의 두 번째 단계로 진행합니다. 플러그인이 사용자를 로컬 사용자 저장소에 매핑할 수 없는 경우에는 UserNotFound 오류가 반환됩니다. 플러그인이 선택적으로 리디렉션 URL 기능을 사용할 수 있습니다. 소비자 플러그인이 없는 경우 리디렉션 URL 은 SAML 인증 체계가 생성하는 오류를 기반으로 합니다.

두 번째 인증 단계에서 시스템은 플러그인이 구성된 경우 메시지 소비자 플러그인의 postAuthenticateUser 메서드를 호출합니다. 메서드가 성공하는 경우 CA SiteMinder?Federation Standalone 은 사용자를 요청된 리소스로 리디렉션합니다. 메서드가 실패하는 경우 사용자를 실패 페이지에 보내도록 플러그인을 구성할 수 있습니다. 실패 페이지는 인증 체계 구성으로 지정할 수 있는 리디렉션 URL 중 하나일 수 있습니다.

참조 정보(메서드 서명, 매개 변수, 반환 값, 데이터 형식)와 UserContext 클래스에 대한 생성자는 *Java SDK 프로그래밍 참조서*에 나와 있습니다. MessageConsumerPlugin 인터페이스를 참조하십시오.

플러그인을 구성하려면

1. CA SiteMinder® Federation Standalone SDK 를 설치합니다.
2. SDK 의 일부인 MessageconsumerPlugin.java 인터페이스를 구현합니다.
3. 메시지 소비자 플러그인 구현 클래스를 배포합니다.
4. Administrative UI 에서 메시지 소비자 플러그인이 사용되도록 설정합니다.

어설션 처리 사용자 지정(신뢰 당사자)

메시지 소비자 플러그인은 메시지 소비자 확장 API 를 구현하는 Java 프로그램입니다. 이 플러그인을 통해 어설션 거부, 상태 코드 반환 등의 어설션 처리를 위한 사용자 고유의 비즈니스 논리를 구현할 수 있습니다. 이 추가 처리는 어설션의 표준 처리와 함께 작동합니다.

인증이 진행되는 동안 시스템은 먼저 사용자를 로컬 사용자 저장소에 매핑하여 어설션을 처리하려고 합니다. CA SiteMinder?Federation Standalone 이 사용자를 찾을 수 없는 경우 메시지 소비자 플러그인의 postDisambiguateUser 메서드를 호출합니다.

플러그인이 사용자를 찾은 경우 인증의 두 번째 단계로 진행합니다. 플러그인이 사용자를 로컬 사용자 저장소에 매핑할 수 없는 경우에는 UserNotFound 오류가 반환됩니다. 플러그인이 선택적으로 리디렉션 URL 기능을 사용할 수 있습니다. 소비자 플러그인이 없는 경우 리디렉션 URL 은 SAML 인증 체계가 생성하는 오류를 기반으로 합니다.

두 번째 인증 단계에서 시스템은 플러그인이 구성된 경우 메시지 소비자 플러그인의 postAuthenticateUser 메서드를 호출합니다. 메서드가 성공하는 경우 CA SiteMinder?Federation Standalone 은 사용자를 요청된 리소스로 리디렉션합니다. 메서드가 실패하는 경우 사용자를 실패 페이지에 보내도록 플러그인을 구성할 수 있습니다. 실패 페이지는 인증 체계 구성으로 지정할 수 있는 리디렉션 URL 중 하나일 수 있습니다.

참조 정보(메서드 서명, 매개 변수, 반환 값, 데이터 형식)와 UserContext 클래스에 대한 생성자는 *Java SDK 프로그래밍 참조서*에 나와 있습니다. MessageConsumerPlugin 인터페이스를 참조하십시오.

플러그인을 구성하려면

1. CA SiteMinder® Federation Standalone SDK 를 설치합니다.
2. SDK 의 일부인 MessageconsumerPlugin.java 인터페이스를 구현합니다.
3. 메시지 소비자 플러그인 구현 클래스를 배포합니다.
4. Administrative UI 에서 메시지 소비자 플러그인이 사용되도록 설정합니다.

MessageConsumerPlugin 인터페이스 구현

MessageConsumerPlugin.java 인터페이스를 구현하여 사용자 지정 메시지 소비자 플러그인을 생성하십시오. 다음 절차에는 구현 클래스에 대한 최소 요구 사항이 나열되어 있습니다.

다음 단계를 수행하십시오.

1. 매개 변수가 포함되지 않은 공개 기본 생성자 메서드를 제공합니다.
2. 상태 비저장 구현이 되도록 코드를 제공합니다. 여러 스레드가 단일 플러그인 클래스를 사용할 수 있어야 합니다.
3. 인터페이스에서 요구 사항을 충족할 메서드를 구현합니다.

   MessageConsumerPlugin 에는 다음 네 가지 메서드가 포함됩니다.

   init()

   플러그인에 필요한 시작 절차를 수행합니다. SiteMinder 는 플러그인이 로드될 때 각 플러그인 인스턴스에 대해 한 번씩 이 메서드를 호출합니다.

   release()

   플러그인에 필요한 런다운 절차를 모두 수행합니다. SiteMinder 는 SiteMinder 가 종료될 때 각 플러그인 인스턴스에 대해 한 번씩 이 메서드를 호출합니다.

   postDisambiguateUser()

   인증 체계가 사용자 명확성 처리를 수행할 수 없을 때 해당 처리를 제공합니다. 또는 이 메서드가 새 페더레이션 사용자에 대한 데이터를 사용자 저장소에 추가할 수 있습니다. 이 메서드는 암호 해독된 어설션을 수신합니다. 암호 해독된 어설션은 플러그인에 전달된 속성 맵의 "_DecryptedAssertion" 키 아래에 추가됩니다.

   postAuthenticateUser()

   정책 서버 처리 성공 여부와 관계없이 어설션 처리 결과를 확인하기 위한 추가 코드를 제공합니다.

제품은 다음과 같은 메시지 소비자 플러그인 클래스 샘플을 제공합니다.

- MessageConsumerPluginSample.java
- MessageConsumerSAML20.java

샘플의 기본 위치는 다음과 같습니다.

Windows

C:\Program Files\Federation Standalone\sdk\java\sample

패키지 이름은 com\ca\federation\sdk\plugin\sample 입니다.

UNIX

/FederationStandalone/sdk/java/sample

패키지 이름은 com/ca/federation/sdk/plugin/sample 입니다.

UI 에서 메시지 소비자 플러그인이 사용되도록 설정

메시지 소비자 플러그인을 작성하고 컴파일한 후 Administrative UI 에서 설정을 구성하여 플러그인이 사용되도록 설정하십시오. UI 설정은 CA SiteMinder?Federation Standalone 에게 플러그인을 찾을 수 있는 위치를 알려 줍니다.

플러그인을 배포할 때까지 플러그인 설정을 구성하지 마십시오.

다음 단계를 수행하십시오.

1. Administrative UI 에 로그온합니다.

   수정할 소비자-생산자 또는 SP-IdP 파트너 관계를 선택합니다.
2. 파트너 관계 마법사의 "사용자 ID" 단계로 이동합니다.
3. "메시지 소비자 플러그인" 섹션의 다음 필드에 데이터를 입력합니다.

   플러그인 클래스

   플러그인에 대한 Java 클래스 이름을 지정합니다. 예를 들어 SDK 에 포함된 샘플 클래스는 다음과 같습니다.

   com.ca.messageconsumerplugin.MessageConsumerPluginSample

   플러그인 매개 변수

   "전체 Java 클래스 이름" 필드에서 지정한 플러그인에 전달되는 매개 변수 문자열을 지정합니다.

4. 운영 환경에 맞게 페더레이션 서비스를 다시 시작합니다.

   - **Windows**

     다음과 같이 중지 및 시작 바로 가기를 사용합니다. 로컬 관리자가 아닌 네트워크 사용자로 로그인한 경우에는 바로 가기를 마우스 오른쪽 단추로 클릭하고 "관리자 권한으로 실행"을 선택하십시오.

     a. "시작", "모든 프로그램", "CA", "Federation Standalone", "서비스 중지"

     b. "시작", "모든 프로그램", "CA", "Federation Standalone", "서비스 시작"

   - **UNIX**

     a. 명령 창을 엽니다.

     b. 다음 스크립트를 실행합니다.

     *federation_install_dir/*fedmanager.sh stop

     federation_install_dir/fedmanager.sh start

     참고: 서비스를 중지했다가 루트 사용자로 시작하지 마십시오.

메시지 소비자 플러그인 배포

MessageConsumerPlugin 인터페이스에 대한 구현 클래스를 코드로 지정한 다음에는 해당 구현 클래스를 컴파일하고 CA SiteMinder?Federation Standalone 이 실행 파일을 찾을 수 있는지 확인하십시오.

다음 단계를 수행하십시오.

1. MessageConsumerPlugin Java 파일을 컴파일합니다. 이 파일을 컴파일하려면 제품과 함께 설치되는 다음 종속 라이브러리가 필요합니다.

   federation_install_dir\siteminder\bin\jars\SmJavaApi.jar

   federation_install_dir 은 CA SiteMinder® Federation Standalone 이 설치된 디렉터리입니다.

2. 폴더나 jar 파일에서 플러그인 클래스를 사용할 수 있는 경우 JVMOptions.txt 파일에서 -Djava.class.path 값을 수정합니다. 이 단계를 수행하면 수정된 클래스 경로를 사용하여 플러그인 클래스를 로드할 수 있습니다.

   federation_install_dir\siteminder\config 디렉터리에서 JVMOptions.txt 파일을 찾습니다.

   참고: 기존 xerces.jar, xalan.jar 또는 SmJavaApi.jar 의 클래스 경로를 수정하지 마십시오.

3. 시스템을 다시 시작하여 최신 버전의 MessageConsumerPlugin 을 선택합니다. 이 단계는 플러그인 Java 파일이 다시 컴파일될 때마다 필요합니다.

4. 플러그인이 사용되도록 설정합니다.

제 12 장: 싱글 사인온 구성

이 섹션은 다음 항목을 포함하고 있습니다.

싱글 사인온 구성(어설션 당사자)

어설션 당사자 측에서 싱글 사인온을 구성할 때 어설션 당사자가 신뢰 당사자에게 어설션을 전달하는 방법을 지정합니다.

브라우저에 하나의 싱글 사인온 세션만 유지됩니다. 세션 정보는 FEDSESSION 쿠키에 저장됩니다. 같은 브라우저에서 다른 파트너 관계에 액세스할 경우, 같은 브라우저 세션 동안 이전에 액세스한 파트너 관계와 기본 사용자 디렉터리가 동일한 경우가 아니면 FEDSESSION 쿠키가 유효하지 않습니다.

FEDSESSION 쿠키에는 다음과 같은 시간 만료 설정이 사용됩니다.

- 유휴 시간 만료: 3600 초(1 시간)
- 최대 시간 만료: 7200 초(2 시간)

이러한 시간 만료 설정은 UI 에서 변경할 수 없습니다.

다음 단계를 수행하십시오.

1. 파트너 관계 마법사의 적절한 단계에서 시작합니다.

   SAML 1.1

   싱글 사인온

   SAML 2.0

   SSO 및 SLO

   참고: "도움말"을 클릭하면 필드, 컨트롤 및 해당되는 각 요구 사항에 대한 설명을 볼 수 있습니다.

2. "인증" 그룹 상자에서 "인증 모드" 옵션을 선택합니다.

   인증 모드

   "로컬" 또는 "위임됨"을 선택합니다.

   - 페더레이션 시스템이 사용자 인증을 처리하는 경우 "로컬"을 클릭합니다.
   - 타사 WAM(웹 액세스 관리) 시스템에서 사용자 인증을 처리하는 경우에는 "위임됨"을 클릭합니다.

3. 선택한 인증 모드에 해당하는 "인증 유형"을 선택합니다. 로컬 인증을 사용하는지 또는 위임된 인증을 사용하는지에 따라 옵션이 달라집니다.

   로컬 인증 유형(로컬 모드에만 해당)

   "기본" 또는 "양식 기반" 선택

   일본어 또는 프랑스어 사용자를 위해 로컬라이즈된 CA SiteMinder® Federation Standalone 을 사용하는 경우에는 양식 기반 인증 체계를 선택하십시오. 기본 인증은 로컬라이즈된 사용자에 대해 지원되지 않습니다.

   양식 인증을 위해 일본어 및 프랑스어로 된 샘플 로그인 양식이 제공됩니다. 이 양식은 *federation_install_dir*/secure-proxy/proxy-engine/examples 디렉터리의 formsja(일본어) 및 formsfr(프랑스어) 폴더에 있습니다.

   로컬라이즈된 양식을 사용하려면

   a. *federation_install_dir*/secure-proxy/proxy-engine/examples 로 이동합니다.

   b. 양식 폴더의 백업 사본을 만듭니다.

   c. 해당하는 언어의 폴더 이름(일본어의 경우 formsja, 프랑스어의 경우 formsfr)을 **forms** 로 변경합니다.

위임된 인증 유형

"레거시 쿠키", "쿼리 문자열", "개방 형식 쿠키"를 선택합니다.

참고: 위임된 인증에 사용할 수 있는 FIPS 호환 옵션은 개방 형식 쿠키뿐입니다.

4. "위임된 인증"을 사용하는 경우, 선택한 위임된 인증 유형의 필수 매개변수를 구성하십시오.

레거시 쿠키

타사 WAM 에서 쿠키에 사용자 아이덴티티 정보를 전달하는 경우 "위임된 인증 URL"을 구성하십시오. 이 URL 은 사용자가 CA SiteMinder® Federation Standalone 에 먼저 액세스한 경우에 요청을 WAM 시스템으로 리디렉션합니다. 사용자가 WAM 시스템에 먼저 액세스한 경우에는 이 URL 이 적용되지 않습니다.

쿼리 문자열

타사 WAM 에서 쿼리 문자열에 사용자 아이덴티티 정보를 전달하는 경우 다음 설정을 구성하십시오.

- 위임된 인증 URL

  이 URL 은 사용자가 CA SiteMinder® Federation Standalone 에 먼저 액세스한 경우에 요청을 WAM 시스템으로 리디렉션합니다. 사용자가 WAM 시스템에 먼저 액세스한 경우에는 이 URL 이 적용되지 않습니다.

- 해시 암호
- 해시 암호 확인

개방 형식 쿠키

타사 WAM 에서 FIPS 암호화된 쿠키에 사용자 아이덴티티 정보를 전달하는 경우 "위임된 인증 URL"을 구성하십시오. 위임된 인증에 사용할 수 있는 FIPS 호환 옵션은 개방 형식 쿠키뿐입니다. 이 URL 은 사용자가 CA SiteMinder® Federation Standalone 에 먼저 액세스한 경우에 요청을 WAM 시스템으로 리디렉션합니다. 사용자가 WAM 시스템에 먼저 액세스한 경우에는 이 URL 이 적용되지 않습니다.

참고: "위임된 인증 유형"으로 "레거시 쿠키" 또는 "개방 형식 쿠키"를 선택한 경우에는 필요한 전역 쿠키 설정을 구성하십시오. "인프라", "배포 설정"으로 이동하여 배포 설정을 찾으십시오.

5. 사용하려는 사용자 인증 방법에 해당하는 URI 를 입력하여 "인증 클래스" 필드의 값을 지정합니다. 이 URI 는 어설션의 AuthnContextClassRef 요소에 배치되어 사용자 인증 방법을 설명합니다.

   지침:

   - 사용자가 로컬로 인증될 경우 "암호"에 기본 URI 를 적용합니다.
   - 사용자가 원격 타사에서 인증될 경우 이 인증 방법을 반영하도록 필드를 편집합니다.

6. SSO 그룹 상자의 필수 필드의 값을 지정하여 싱글 사인온 작동 방법을 구성합니다.

   다음 지침에 유의하십시오.

   - 아티팩트 바인딩을 선택하는 경우에는 아티팩트 인코딩("URL" 또는 "양식")을 선택합니다. 인코딩에 따라 아티팩트가 신뢰 당사자로 반환되는 방식이 정의됩니다. "URL" 옵션을 선택하는 경우 아티팩트가 URL 의 쿼리 매개 변수로 반환됩니다. "양식"을 선택하는 경우 아티팩트가 양식 데이터로 게시됩니다.
   - SAML 2.0 의 경우 두 가지 바인딩 모두 선택할 수 있으며, 로컬 엔터티에 따라 바인딩이 시도되는 순서가 결정됩니다.

     참고: 아티팩트 바인딩의 경우 어설션이 보안 백 채널을 통해 전송됩니다. 따라서 "백 채널" 그룹 상자에서 설정을 구성하십시오.

   - SSO 바인딩을 선택하는 경우 일치하는 바인딩을 사용하는 어설션 소비자 서비스를 하나 이상 구성하십시오. "향상된 클라이언트 및 프록시 프로필"을 선택하는 경우에는 PAOS 바인딩을 사용하는 어설션 소비자 서비스가 필요합니다.
   - "SSO 유효 기간" 및 "차이 시간"에 따라 어설션이 유효한 시기가 결정됩니다. 이 설정이 함께 작동하는 방식에 대해서는 어설션 유효 기간 (페이지 222)을 참조하십시오.

7. 어설션 소비자 서비스에 대한 URL 을 지정합니다. 이 서비스는 수신된 어설션을 처리하는 신뢰 당사자의 서비스입니다.

원격 신뢰 당사자를 생성하거나 가져올 때 정의된 모든 값이 입력됩니다.

이로써 어설션 당사자의 SSO 구성이 완료되었습니다.

추가 정보:

싱글 사인온에 대한 어설션 유효 기간 (페이지 222)
위임된 인증 (페이지 267)

HTTP-POST SSO 에 사용할 자동 POST 양식 사용자 지정

사용자 환경을 개선할 수 있도록 SAML 응답에서 신뢰 당사자로 전송되는 자동 POST 양식을 사용자 지정할 수 있습니다.

사용자 지정된 양식을 사용하려면 마법사의 "SSO 및 SLO" 단계에서 "SSO" 섹션의 "사용자 지정 Post 양식" 필드에 양식 이름을 입력하십시오. 시스템은 사용자가 지정한 양식을 응답에 사용합니다. 제품에는 defaultpostform.html 이라는 양식이 포함되어 있습니다.

참고: 양식 이름만 입력하고 양식 경로는 입력하지 마십시오.

실제 페이지는 *federation_install_dir*\customization 디렉터리에 있어야 하며, 여기서 *federation_install_dir* 은 제품이 설치된 위치입니다.

파트너 관계 페더레이션을 사용하는 인증 옵션

독립 실행형 파트너 관계 페더레이션에서는 페더레이션된 싱글 사인온의 인증 모드를 선택할 수 있습니다. 이 인증 모드는 어설션 당사자 측에서 싱글 사인온을 구성할 때 선택합니다.

- **로컬 인증 모드**

  로컬 인증은 로컬 페더레이션 시스템에서 이루어집니다. 로컬 인증의 경우 인증 체계로 기본 인증이나 양식 인증을 선택할 수 있습니다. 이 두 옵션은 로컬로 사용할 수 있는 유일한 방법입니다.

- **위임된 인증 모드**

  위임된 인증에서는 타사 WAM(웹 액세스 관리) 시스템에 인증 태스크를 전달합니다. 타사에서 사용자를 인증하는 데 사용되는 방법은 해당 타사가 지원하는 인증 체계에 따라 달라집니다. 타사 WAM 은 사용자가 인증되면 사용자 인증을 원래 요청했던 엔터티에 페더레이션된 사용자 아이덴티티를 돌려 보냅니다.

싱글 사인온 구성(신뢰 당사자)

신뢰 당사자 측에서 싱글 사인온을 구성하려면 신뢰 당사자가 지원하는 SAML 바인딩 및 신뢰 당사자가 싱글 사인온 통신을 처리하는 방법과 관련된 사항을 지정해야 합니다.

신뢰 당사자 측에서 CA SiteMinder?Federation Standalone 은 파트너 관계에 설정된 차이 시간을 사용하여 수신된 어설션이 유효한지 확인합니다. CA SiteMinder?Federation Standalone 이 구성된 차이 시간을 사용하는 방법을 이해하려면 어설션 유효 기간 (페이지 222)을 참조하십시오.

다음 단계를 수행하십시오.

1. 파트너 관계 마법사의 적절한 단계에서 시작합니다.

   SAML 1.1

   싱글 사인온

   SAML 2.0

   SSO 및 SLO

2. 사용하는 프로필에 대해 "SSO" 그룹 상자에서 설정을 구성합니다.

   SAML 2.0 의 경우 "아티팩트"와 "POST"를 모두 선택할 수 있으며 로컬 엔터티에 따라 바인딩이 시도되는 순서가 결정됩니다.

   참고: "도움말"을 클릭하면 필드, 컨트롤 및 해당되는 각 요구 사항에 대한 설명을 볼 수 있습니다.

3. "HTTP-아티팩트"를 선택하면 나가는 백 채널에 대한 인증 방법도 구성합니다.

이로써 신뢰 당사자의 SSO 구성이 완료되었습니다.

싱글 사인온에 대한 어설션 유효 기간

싱글 사인온의 경우 차이 시간 및 SSO 유효 기간 값에 따라 CA SiteMinder?Federation Standalone 이 어설션의 총 유효 기간을 계산하는 방법이 결정됩니다. CA SiteMinder?Federation Standalone 은 어설션의 생성 및 소비에 차이 시간을 적용합니다. 어설션 문서에서 NotBefore 및 NotOnOrAfter 값은 유효 간격의 시작과 끝을 나타냅니다.

어설션 당사자 측에서 CA SiteMinder?Federation Standalone 은 어설션 유효 기간을 설정합니다. CA SiteMinder?Federation Standalone 은 어설션이 생성될 때 시스템 시간을 가져와서 유효 간격의 시작을 결정합니다. CA SiteMinder?Federation Standalone 은 이 시간부터 어설션에 IssueInstant 값을 설정합니다. 그런 다음 CA SiteMinder?Federation Standalone 은 IssueInstant 값에서 차이 시간 값을 뺍니다. 그 결과로 얻은 시간은 NotBefore 값이 됩니다.

NotBefore = IssueInstant - 차이 시간

유효 간격의 끝을 결정하기 위해 CA SiteMinder?Federation Standalone 은 유효 기간 값과 차이 시간을 IssueInstant 값에 더합니다. 그 결과로 얻은 시간은 NotOnOrAfter 값이 됩니다.

NotOnOrAfter = 유효 기간 + 차이 시간 + IssueInstant

시간은 GMT 를 기준으로 합니다.

예를 들어 어설션 당사자 측에서 어설션이 1:00 GMT 에 생성된다고 가정합니다. 차이 시간은 30 초이고 유효 기간은 60 초이며 어설션 유효 간격은 12:59:30 GMT 에서 1:01:30 GMT 사이입니다. 이 간격은 어설션이 생성된 시간보다 30 초 전에 시작되고 90 초 후에 끝납니다.

신뢰 당사자 측에서 CA SiteMinder?Federation Standalone 은 어설션 당사자 측에서와 동일한 계산을 수행하여 수신된 어설션이 유효한지 확인합니다.

CA SiteMinder® Federation Standalone 이 파트너 관계의 양쪽 모두에 있는 경우의 어설션 유효 기간 계산

CA SiteMinder?Federation Standalone 이 파트너 관계의 양쪽 모두에 있는 경우, 어설션은 SSO 유효 기간에 차이 시간의 두 배를 합한 기간 동안 유효합니다. 공식은 다음과 같습니다.

어설션 유효 기간 = 2 x 차이 시간(어설션 당사자) + SSO 유효 기간 + 2 x 차이 시간(신뢰 당사자)

공식의 첫 부분(2 x 차이 시간 + SSO 유효 기간)은 어설션 당사자의 유효 기간 시작 및 끝을 나타냅니다. 공식의 두 번째 부분(2 x 차이 시간)은 신뢰 당사자에 있는 시스템 클록의 차이 시간을 나타냅니다. 2 를 곱하는 이유는 유효 기간의 NotBefore 및 NotOnOrAfter 끝을 고려하기 때문입니다.

참고: CA SiteMinder® Federation Standalone 의 경우 SSO 유효 기간은 어설션 당사자 측에서만 설정됩니다.

예

어설션 당사자

어설션 당사자 측에서의 값은 다음과 같습니다.

IssueInstant = 5:00PM

SSO 유효 기간 = 60 초

차이 시간 = 60 초

NotBefore = 4:59PM

NotOnOrAfter = 5:02PM

신뢰 당사자

신뢰 당사자는 어설션에서 받은 NotBefore 및 NotOnOrAfter 값을 가져와서 해당 값에 차이 시간을 적용하여 새 NotBefore 및 NotOnOrAfter 값을 계산합니다.

차이 시간 = 180 초(3 분)

NotBefore = 4:56PM

NotOnOrAfter = 5:05PM

이러한 값을 기반으로 총 어설션 유효 기간에 대한 계산은 다음과 같습니다.

120 초(2 x 60) + 60 초 + 360 초(2 x 180) = 540 초(9 분)

서비스 공급자에서의 세션 유효 기간

서비스 공급자에서 인증 세션의 기간을 관리할 수 있습니다. SessionNotOnOrAfter 특성은 IdP 가 어설션의 <AuthnStatement>에 포함할 수 있는 선택적 특성입니다. 어설션 유효 기간의 구성은 IdP 에서 수행됩니다.

참고: SessionNotOnOrAfter 매개 변수는 어설션의 유효 기간을 결정하는 NotOnOrAfter 매개 변수와는 다릅니다.

타사 SP 는 SessionNotOnOrAfter 의 값을 사용하여 자체 시간 만료 값을 설정할 수 있으므로 너무 짧은 세션을 방지하는 데 도움이 됩니다. 사용자 세션이 무효화되면 사용자는 아이덴티티 공급자에서 다시 인증해야 합니다.

중요! SiteMinder 가 SP 로 작동하고 있는 경우 SessionNotOnOrAfter 값이 무시됩니다. 대신에 SiteMinder SP 는 대상 리소스를 보호하는 SAML 인증 체계에 해당하는 영역 시간 만료를 바탕으로 세션 시간 만료를 설정합니다.

다음 단계를 수행하십시오.

1. Administrative UI 에 로그인합니다.
2. 수정할 IdP->SP 파트너 관계를 선택합니다.
3. "SSO 및 SLO" 단계로 이동합니다.
4. SSO 섹션에서 "SP 세션 유효 기간"에 대한 옵션을 선택합니다 사용자 지정 옵션을 선택하면 여러 개의 옵션을 선택할 수 있습니다.

   필드 설명을 보려면 "도움말"을 클릭하십시오.
5. 변경을 완료하고 "마침"을 클릭한 후 "확인" 단계를 선택합니다.

HTTP 오류에 대한 상태 리디렉션(SAML 2.0 IdP)

아이덴티티 공급자의 경우 HTTP 500, 400 또는 405 오류 발생 시 SiteMinder 가 사용자를 리디렉션하는 방법을 구성할 수 있습니다. 예를 들어 요청의 URL 이 잘못된 대상을 가리키기 때문에 403 오류가 발생할 수 있습니다. 이러한 오류가 발생하면 추가 처리를 위해 지정된 URL 로 사용자가 리디렉션됩니다.

다음과 같이 리디렉션 옵션을 선택하십시오.

1. "SSO 및 SLO" 대화 상자의 "상태 리디렉션 URL" 섹션으로 이동합니다.
2. "상태 리디렉션 URL" 섹션에서 리디렉션을 수행하는 오류 조건에 대한 확인란을 선택합니다.
3. SiteMinder 가 사용자를 리디렉션하는 대상 URL 을 입력합니다.
4. 각 URL 에 대해 리디렉션 방법으로 "302 데이터 없음" 또는 "HTTP Post"를 선택합니다.

리디렉션 처리가 구성되었습니다.

싱글 사인온을 시작할 수 있는 SAML 2.0 엔터티

SAML 2.0 파트너 관계의 경우 싱글 사인온을 시작할 수 있는 항목을 IdP 또는 SP 또는 둘 다로 결정할 수 있습니다. 파트너 관계의 각 측에서 허용되는 트랜잭션을 구성할 수 있습니다.

트랜잭션 초기화의 제한이 사용자 인증 컨텍스트 정보 교환과 같은 다른 싱글 사인온 기능에 어떤 영향을 미치는지 고려하십시오.

다음 단계를 수행하십시오.

1. Administrative UI 에 로그인합니다.
2. 편집할 SAML 2.0 파트너 관계를 선택합니다.
3. 파트너 관계 마법사의 "SSO 및 SLO" 단계로 이동합니다.
4. "트랜잭션 허용됨" 필드의 풀다운 메뉴에서 옵션을 선택합니다.
5. 마법사의 "확인" 단계로 건너뛰고 변경 내용을 저장합니다.

아티팩트 SSO 에 대한 백 채널 인증

아티팩트 싱글 사인온을 위해서는 신뢰 당사자가 어설션을 검색하기 위해 어설션 당사자에 아티팩트를 보내야 합니다. 어설션 당사자는 아티팩트를 사용하여 올바른 어설션을 검색하고 백 채널을 통해 어설션을 신뢰 당사자에 반환합니다.

엔터티가 백 채널에 액세스하려면 인증이 필요하도록 설정할 수 있습니다. SSL 은 필수가 아니지만 SSL 을 사용하여 백 채널에 보안을 적용할 수도 있습니다.

SSL 을 사용하여 백 채널에 보안을 적용하는 절차는 다음과 같습니다.

1. SSL 이 사용되도록 설정합니다.

   기본 인증에는 SSL 이 필요하지 않지만 SSL 을 통한 기본 인증을 사용할 수 있습니다. 클라이언트 인증서 인증에는 SSL 이 필요합니다.

2. SAML 2.0 통신 교환에 대한 들어오는 백 채널 또는 나가는 백 채널을 구성합니다. 구성하는 방향은 로컬 엔터티의 역할에 따라 다릅니다.

   별도 채널의 구성은 SAML 2.0 에만 지원됩니다. SAML 1.1 아티팩트 싱글 사인온에 대한 백 채널 구성은 각 파트너 관계에 대해 단일 구성을 사용합니다. SiteMinder 는 자동으로 올바른 방향(로컬 생산자에 대해서는 들어오는 방향, 로컬 소비자에 대해서는 나가는 방향)을 사용합니다.

   구성 중인 엔터티에 기반하여 SAML 2.0 싱글 사인온에 대해 구성할 방향을 선택합니다.

   - 로컬 어설션 당사자는 들어오는 채널을 사용합니다.
   - 로컬 신뢰 당사자는 나가는 채널을 사용합니다.

   참고: 들어오고 나가는 백 채널을 구성할 수 있지만 한 채널은 하나의 구성만 가질 수 있습니다. 동일한 채널을 사용하는 두 서비스는 동일한 백 채널 구성을 사용합니다. 예를 들어 로컬 어설션 당사자의 수신 채널이 HTTP-아티팩트 SSO 와 SOAP 기반 SLO 를 지원할 경우 이 두 서비스는 동일한 백 채널 구성을 사용해야 합니다.

3. 신뢰 당사자가 보호된 백 채널을 통해 액세스를 얻기 위한 인증 유형을 선택합니다. 인증 방법은 채널별로(나가는 채널 또는 들어오는 채널) 적용됩니다.

   백 채널 인증에 대한 옵션은 다음과 같습니다.

   - 기본
   - 클라이언트 인증서
   - 인증 없음

   이러한 옵션은 Administrative UI 도움말에 자세히 설명되어 있습니다.

   중요! 들어오는 백 채널에 대한 인증 방법은 파트너 관계에서 다른 측의 나가는 백 채널에 대한 인증 방법과 일치해야 합니다. 인증 방법의 선택에 대한 동의는 대역 외 통신에서 처리됩니다.

HTTP-아티팩트 백 채널 구성

어설션 당사자가 신뢰 당사자에게 어설션을 보낼 때 사용되는 HTTP-아티팩트 백 채널을 보호하십시오.

다음 제한을 고려하십시오.

ServletExec 를 실행 중인 다음 웹 서버에는 클라이언트 인증서 인증을 사용할 수 없습니다.

- SiteMinder 생산자/아이덴티티 공급자의 IIS 웹 서버 - IIS 의 제한 때문
- SiteMinder 생산자/아이덴티티 공급자의 SunOne/Sun Java Server 웹 서버 - ServletExec 에 설명된 제한 때문

다음 단계를 수행하십시오.

1. 파트너 관계 마법사의 "싱글 사인온" 또는 "SSO 및 SLO" 단계의 "백 채널" 섹션에서 시작합니다.
2. SSO 섹션에서 HTTP-Artifact 를 선택합니다.

   "인증 방법" 필드가 활성화됩니다.
3. 들어오는 백 채널, 나가는 백 채널 또는 둘 다에 대한 인증 방법의 유형을 선택합니다.

   필드 설명을 보려면 "도움말"을 클릭하십시오.

   - 클라이언트 인증서 인증 체계를 선택하는 경우 인증서 데이터 저장소에 개인 키/인증서 쌍을 추가합니다. 개인 키/인증서 쌍은 인증 기관에서 발급합니다.

     중요! 인증서 제목의 CN 은 생산자에서 구성된 생산자 대 소비자 파트너 관계의 파트너 관계 이름과 동일해야 합니다.

     인증서 추가에 대한 지침은 "정책 서버 구성 안내서"를 참조하십시오. 키/인증서 쌍이 데이터 저장소에 이미 있는 경우 이 단계를 건너뛰십시오.
   - "인증 없음"을 인증 방법으로 선택하는 경우에는 추가적인 단계가 필요 없습니다.
4. 선택하는 인증 방법에 따라 구성해야 할 몇 개의 추가적인 필드가 표시됩니다.

모든 필수 필드에 값을 입력하면 백 채널 구성이 완료됩니다. 보안을 더욱 강화하기 위해 연결의 각 측에서 SSL 이 사용되도록 설정할 수 있습니다.

SAML 2.0 특성 쿼리 지원이 사용되도록 설정하는 방법

SiteMinder IdP 는 SAML 2.0 어설션 쿼리/요청 프로필을 지원하며 특성 쿼리에 응답할 수 있습니다. 또한 IdP 는 어설션이나 메타데이터에 없는 특성에 대한 쿼리를 수락하여 프로필 기능을 확장합니다. IdP 는 특성 쿼리를 받으면 먼저 사용자 디렉터리에서 특성을 찾습니다. 해당 특성이 없을 경우 정책 서버가 세션 저장소를 확인합니다. 세션 저장소는 외부 아이덴티티 공급자의 특성, 고급 인증 체계에서 수집된 특성 및 다른 원본의 특성을 포함할 수 있습니다.

참고: SiteMinder IdP 만 쿼리 프로필을 지원합니다. 특성 요청자로서 SiteMinder SP 는 프록시된 특성 쿼리 기능 (페이지 233)에 대해서만 지원됩니다.

IdP 에는 SP 가 메타데이터에서 요청할 수 있는 모든 사용자 특성이 있습니다. SP 는 다음과 같은 두 가지 방법으로 이러한 특성을 획득할 수 있습니다.

- 어설션에 전송된 특성 집합을 추출합니다.

  아이덴티티 공급자 어설션 구성은 포함되는 특성 집합을 결정합니다. 모든 특성의 하위 집합을 정의하는 경우 가장 필요한 특성만 포함하도록 특성의 수를 제한하여 처리 오버헤드를 줄일 수 있습니다.
- IdP 메타데이터를 가져옵니다.

메타데이터의 특성에 추가하여, SP 는 어설션 또는 메타데이터에 없는 특성을 요구할 수 있습니다. 다른 특성을 가져오려면 SP 에서 IdP 에 특성 쿼리를 보내야 합니다.

쿼리 요청 프로필은 다음 두 개 엔터티를 갖습니다.

- SAML 특성 기관
- SAML 특성 요청자

SiteMinder IdP 는 특성 기관으로만 기능할 수 있습니다. SiteMinder SP 는 특성 요청자가 될 수 없습니다.

다음 그림은 특성 기관에 대한 구성 단계를 보여 줍니다.

다음 단계를 완료합니다.

- IdP-SP 파트너 관계를 구성 또는 수정합니다 (페이지 232).
- 아이덴티티 공급자에 있는 경우 SAML 2.0 특성 기관을 구성합니다 (페이지 232).

SiteMinder 가 파트너 관계의 양쪽에 모두 있는 경우 어설션 쿼리/응답 프로필을 사용할 수 없습니다.

특성 쿼리 지원을 위한 파트너 관계 구성

IdP 가 특성 쿼리에 응답하려면 IdP-SP 파트너 관계가 있어야 합니다. 파트너 관계를 만들거나 기존 파트너 관계를 수정할 수 있습니다.

파트너 관계를 만들기 위한 단계는 다음을 포함합니다.

1. SAML 2.0 IdP 및 SP 엔터티를 만듭니다 (페이지 127).
2. 파트너 관계에 대한 사용자 디렉터리로의 연결을 구성합니다 (페이지 87).
3. SAML 2.0 IdP-SP 파트너 관계를 만듭니다 (페이지 169).
4. SAML 2.0 특성 기관을 구성합니다 (페이지 232).

SAML 2.0 특성 기관 구성

IdP 가 특성 기관으로 사용되도록 구성할 수 있습니다.

다음 단계를 수행하십시오.

1. Administrative UI 에 로그인합니다.
2. "페더레이션", "파트너 관계"를 클릭합니다.
3. 수정할 IdP-SP 파트너 관계를 선택하거나 새 파트너 관계를 생성합니다.
4. 파트너 관계 마법사의 "SSO 및 SLO" 단계로 이동합니다.
5. 대화 상자의 "특성 서비스" 섹션에서 "사용"을 선택합니다.
6. "유효 기간"에 시간(초)을 입력합니다.

   참고: "도움말"을 클릭하면 필드, 컨트롤 및 해당되는 각 요구 사항에 대한 설명을 볼 수 있습니다.
7. (선택 사항) 특성 쿼리에 서명이 필요한지 여부와 특성 어설션 및 응답에 대한 서명 요구 사항을 지정합니다.
8. "사용자 조회" 섹션에서 적절한 사용자 디렉터리 네임스페이스에 대한 검색 사양을 입력합니다. 특성 기관은 이 검색 사양을 사용하여 사용자를 명확히 구분합니다.

   LDAP 사용자 디렉터리의 예는 uid=%s 입니다. 하나 이상의 검색 사양이 필요합니다.

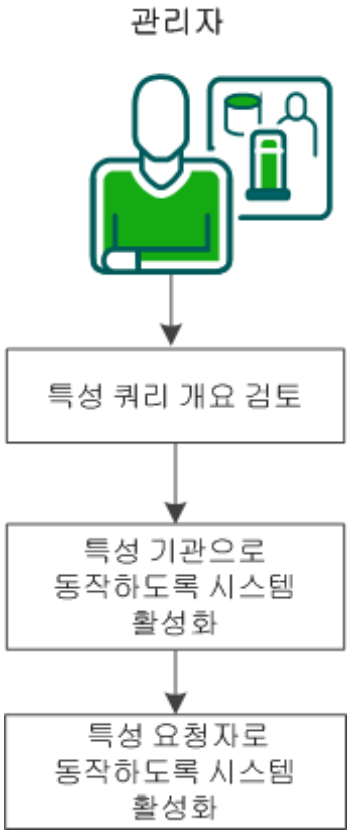
9. (선택 사항) "백 채널" 섹션에서 "보호 유형"으로 "파트너 관계"를 지정합니다. 그런 다음 인증 방법을 선택합니다. 백 채널에 대한 자세한 내용을 보려면 "도움말"을 클릭하십시오.
10. 파트너 관계를 저장한 후 활성화합니다.

이제 아이덴티티 공급자를 특성 기관으로 사용할 수 있습니다. 이 기관은 이제 타사 SP 의 특성 쿼리에 응답합니다.

타사 원본에서 사용자 특성 값을 가져오는 방법

경우에 따라 SAML 2.0 페더레이션된 환경의 서비스 공급자는 어설션에 제공되지 않는 사용자 관련 정보를 필요로 합니다. 서비스 공급자는 미리 결정된 사용자 특성의 값을 요청할 수 있습니다. 아이덴티티 공급자에 이러한 값이 없는 경우 타사에 값을 요청할 수 있습니다. SiteMinder 환경에서는 이 기능을 프록시 특성 쿼리라고 합니다.

다음 다이어그램에서는 프록시 특성 쿼리 기능이 사용되도록 설정하는 프로세스를 보여 줍니다.



프록시 특성 쿼리가 사용되도록 설정하려면 다음 태스크를 완료하십시오.

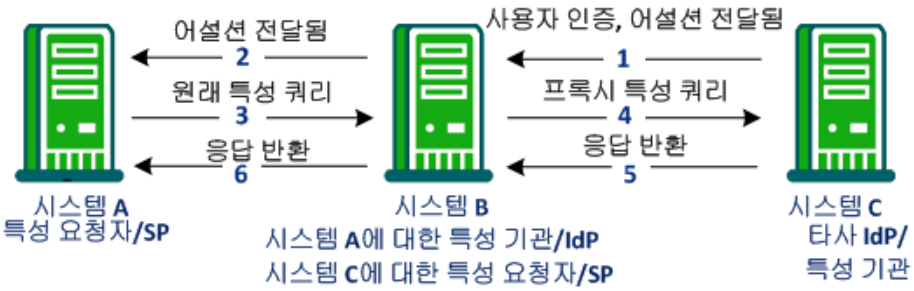
1. 프록시 특성 쿼리 개요를 검토합니다. (페이지 234)
2. 시스템이 특성 기관 역할을 하도록 설정합니다 (페이지 235).
3. 시스템이 특성 요청자 역할을 하도록 설정합니다 (페이지 236).

프록시 특성 쿼리 개요

프록시 특성 쿼리 기능은 SAML 2.0 어설션 쿼리/요청 프로필을 기반으로 하며 사용자 특성 검색을 확장합니다. 특성 기관은 먼저 사용자 디렉터리 및 세션 저장소에서 특성을 검색합니다. 특성을 찾을 수 없고 사용자가 처음에 타사 IdP 에서 인증된 경우 타사 IdP 에 요청을 전달할 수 있습니다.

프록시 특성 쿼리를 구현하기 위해 단일 SiteMinder 시스템은 두 원격 시스템 간의 릴레이 지점으로 사용됩니다. 두 원격 시스템 간에 요청을 릴레이하기 위해 단일 시스템에서 두 가지 역할을 수행합니다. 시스템은 먼저 원래 특성 요청자에 대한 특성 기관으로 사용됩니다. 또한 타사 IdP 에 대한 특성 요청자로 사용됩니다. 특성 요청자로 사용되는 시스템은 특성 쿼리를 원래 IdP 로 프록시합니다.

다음 그림에서는 단일 시스템이 프록시 쿼리를 처리하는 방식을 보여 줍니다.



다음 단계에서는 프록시 특성 쿼리의 흐름을 설명합니다.

1. 처음에 사용자는 타사 IdP 인 시스템 C 에서 인증됩니다. 시스템 C 는 어설션을 생성하여 시스템 B 에 전달합니다.
2. 시스템 B 는 어설션을 시스템 A 로 보내 시스템 A, B, C 사이에서 최초 싱글 사인온 트랜잭션을 완료합니다. 이 싱글 사인온 트랜잭션은 프록시 특성 쿼리를 처리하는 데 필요합니다.

3. 시스템 A 가 어설션을 받은 후에 시스템은 어설션에 없는 다른 특성이 필요함을 파악합니다. 특성 요청자로서 시스템 A 는 특성 기관/IdP, 시스템 B 로 특성 쿼리를 보냅니다.
4. 시스템 B 는 시스템 A 가 사용자 디렉터리 또는 세션 저장소에 없는 특성이 필요함을 파악합니다. 특성을 가져오기 위해 시스템 B 는 새 쿼리 요청을 생성합니다. 새 쿼리를 사용자가 원래 인증된 타사 IdP 인 시스템 C 로 보냅니다. 이 새 쿼리는 프록시 쿼리입니다.
5. 시스템 C 가 특성이 포함된 응답을 시스템 B 에 반환합니다. 시스템 B 가 이러한 특성을 세션 저장소에 저장합니다.
6. 특성 기관 역할을 수행하는 시스템 B 가 특성이 포함된 응답을 시스템 A 에 반환합니다.

중요! 시스템 A 의 구성된 특성 이름 및 이름 형식(지정되지 않음, uri, 기본)은 시스템 C 의 이러한 특성 이름과 일치해야 합니다. 이 정보는 트랜잭션이 발생하기 전에 전달됩니다.

시스템이 특성 기관 역할을 하도록 설정(IdP->SP)

프록시 쿼리 트랜잭션을 구현하려면 동일한 SiteMinder 시스템에서 다음과 같은 두 가지 파트너 관계를 구성하십시오.

- IdP-SP 파트너 관계
- SP-IdP 파트너 관계

SiteMinder 가 특성 기관의 역할을 하기 위해서는 기존 IdP-SP 파트너 관계를 수정하거나 새 파트너 관계를 생성해야 합니다. 이 파트너 관계에서 SiteMinder 는 로컬 IdP/특성 기관이 되고 원격 파트너는 SP/특성 요청자가 됩니다.

참고: 이 시스템은 또한 SP-IdP 파트너 관계에서 특성 요청자로서의 역할을 합니다.

다음 단계를 수행하십시오.

1. Administrative UI 에 로그인합니다.
2. "페더레이션", "파트너 관계"를 클릭합니다.
3. 수정할 IdP-SP 파트너 관계를 선택하거나 새 파트너 관계를 생성합니다.
4. 파트너 관계 마법사의 "SSO 및 SLO" 단계로 이동합니다.

5. 대화 상자의 "특성 서비스" 섹션에서 "사용"을 선택합니다.
6. "유효 기간"에 시간(초)을 입력합니다.

   참고: "도움말"을 클릭하면 필드, 컨트롤 및 해당되는 각 요구 사항에 대한 설명을 볼 수 있습니다.
7. (선택 사항) 특성 쿼리에 서명이 필요한지 여부와 특성 어설션 및 응답에 대한 서명 요구 사항을 지정합니다.
8. "프록시 쿼리 사용"을 선택합니다.
9. "사용자 조회" 섹션에서 적절한 사용자 디렉터리 네임스페이스에 대한 검색 사양을 입력합니다. 특성 기관은 이 검색 사양을 사용하여 사용자를 명확히 구분합니다.

   LDAP 사용자 디렉터리의 예는 uid=%s 입니다. 하나 이상의 검색 사양이 필요합니다.
10. (선택 사항) "백 채널" 섹션에서 "보호 유형"으로 "파트너 관계"를 지정합니다. 그런 다음 인증 방법을 선택합니다. 백 채널에 대한 자세한 내용을 보려면 "도움말"을 클릭하십시오.
11. 파트너 관계를 저장한 후 활성화합니다.

이제 시스템이 원래 특성 요청자에 대해 특성 기관의 역할을 할 수 있습니다.

시스템이 특성 요청자 역할을 하도록 설정(SP->IdP)

프록시 쿼리 트랜잭션을 구현하려면 동일한 SiteMinder 시스템에서 다음과 같은 두 가지 파트너 관계를 구성하십시오.

- IdP-SP 파트너 관계
- SP-IdP 파트너 관계

참고: 파트너 관계 페더레이션은 프록시 특성 쿼리 기능에 대해서만 특성 요청자로서 SP 를 지원합니다.

SiteMinder 가 특성 요청자의 역할을 하기 위해서는 기존 SP-IdP 파트너 관계를 수정하거나 새 파트너 관계를 생성해야 합니다. 이 파트너 관계에서 SiteMinder 는 로컬 SP/특성 요청자가 되고 원격 타사는 원격 IdP/특성 기관이 됩니다.

참고: 이 시스템은 또한 IdP-SP 파트너 관계에서 특성 기관으로서의 역할도 합니다.

다음 단계를 수행하십시오.

1. Administrative UI 에 로그인합니다.
2. "페더레이션", "파트너 관계"를 클릭합니다.
3. 수정할 SP-IdP 파트너 관계를 선택하거나 새 파트너 관계를 생성합니다.
4. 파트너 관계 마법사의 "SSO 및 SLO" 단계로 이동합니다.
5. "특성 요청자 서비스" 섹션에서 "사용" 및 "프록시 쿼리 사용"을 선택합니다.
6. "특성 서비스" 섹션에서 원격 IdP 의 URL 을 지정합니다.
7. 이름 ID 의 형식, 유형 및 값을 지정합니다.

   참고: "도움말"을 클릭하면 필드, 컨트롤 및 해당되는 각 요구 사항에 대한 설명을 볼 수 있습니다.
8. (선택 사항) 백 채널의 인증 유형을 선택합니다. 백 채널에 대한 자세한 내용을 보려면 "도움말"을 클릭하십시오.
9. 파트너 관계를 저장한 후 활성화합니다.

이제 서비스 공급자를 특성 요청자로 사용할 수 있습니다.

어설션 전송을 위한 사용자 동의를 얻는 방법

페더레이션된 파트너 관계는 두 당사자 사이의 신뢰에 기반합니다. 이러한 신뢰 관계에는 사용자 권한이 있어야만 신뢰 당사자에게 아이덴티티 정보를 전달할 수 있다는 계약 조건이 포함될 수 있습니다. 또한 요청된 서비스를 위해 자신의 아이덴티티 정보를 교환할지 여부를 사용자가 제어할 수 있게 하면 신뢰 관계를 유지하는 데 도움이 됩니다.

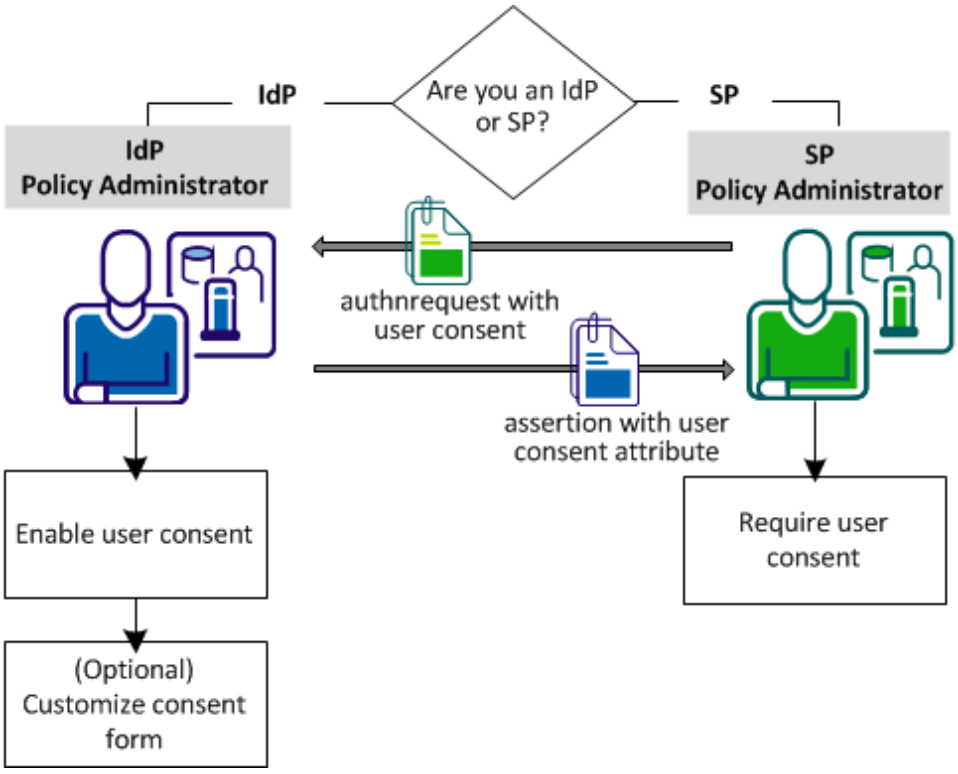
아이덴티티 공급자 역할을 하는 페더레이션 시스템에서는 SAML 2.0 사용자 동의 기능을 지원합니다. 아이덴티티 공급자 사이트에서 사용자 동의를 사용하려면 아이덴티티 공급자가 파트너에게 어설션을 전송하기 전에 사용자에게 권한 부여를 요청해야 합니다. 아이덴티티 공급자에서 사용자 동의가 사용되도록 설정하면 아이덴티티 공급자가 사용자에게 동의 여부를 묻습니다. 아이덴티티 공급자는 어설션에서 동의 값을 전달합니다.

동의 유효 기간은 5 분입니다. 아이덴티티 공급자가 사용자를 동의 페이지로 리디렉션하면 사용자는 5 분 이내에 동의를 해야 합니다. 동의하면 아이덴티티 공급자로 다시 리디렉션됩니다. 그러면 아이덴티티 공급자가 어설션을 생성하여 이를 서비스 공급자로 전송합니다. 이 태스크는 5 분 내에 완료되어야 합니다. 아이덴티티 공급자가 어설션을 생성하기 전에 시간이 만료되면 사용자 아이덴티티를 전달하지 않습니다.

동의는 단일 어설션에만 적용됩니다. 아이덴티티 공급자는 어설션을 생성한 후에 동의가 부여된 모든 레코드를 삭제합니다. 5 분의 유효 기간이 만료되기 전에 동일한 사용자가 아이덴티티 공급자로 돌아갈 수 있지만 아이덴티티 공급자는 여전히 사용자에게 동의할지를 묻습니다.

참고: 유효 기간은 구성할 수 없습니다.

다음 그림에서는 각 파트너에서의 구성 태스크를 보여 줍니다.



IdP 에서 수행하는 구성 태스크는 다음과 같습니다.

1. IdP 에서 사용자 동의 기능을 사용합니다 (페이지 239).
2. 사용자 동의 양식을 사용자 지정합니다(선택 사항) (페이지 240).

SP 에서 수행하는 구성 태스크는 다음과 같습니다.

1. SP 에서 사용자 동의를 요청합니다. (페이지 241)

사용자 동의 예

다음 사용 사례는 사용자 동의를 보여 줍니다.

User1 이 오후 2 시에 MyWorkPlace.com 에 로그인하고 인증합니다. MyWorkPlace 가 아이덴티티 공급자의 역할을 합니다. 오후 2 시 3 분에 사용자가 직원용 여행 특별 상품을 운영하는 파트너 회사의 링크를 선택합니다. User1 은 ExampleTravel.com 으로 보내지기 전에 동의 여부를 묻는 양식으로 리디렉션됩니다. User1 은 동의 양식을 기입하기 전에 전화 통화를 합니다. 현재 시간은 오후 2 시 10 분입니다. 이 경우 유효 기간이 만료되었기 때문에 MyWorkPlace 는 어설션을 생성하지 않습니다.

User1 이 오후 2 시 5 분까지 신속하게 동의하고 아이덴티티 공급자로 다시 리디렉션되면 아이덴티티 공급자는 어설션을 생성합니다. 동의와 어설션 사이에 2 분만 경과되었으므로 유효 기간이 여전히 활성 상태입니다.

IdP 에서 사용자 동의 설정

사용자 동의를 구성하려면 다음과 같이 해야 합니다.

- Administrative UI 에서 사용자 동의를 설정합니다.
- 사용자 동의 양식의 이름을 제공합니다.

  아이덴티티 공급자는 사용자의 동의를 구하기 위한 사용자 양식을 전송합니다.

Administrative UI 를 사용하여 아이덴티티 공급자에서 사용자 동의를 구성합니다. UI 를 통해 이 기능을 구성하면 어설션 응답에 다음 URI 만 사용됩니다.

```
urn:oasis:names:tc:SAML:2.0:consent:obtained
```

CA SiteMinder?Federation Standalone Java 또는 .NET SDK 를 사용하여 이 기능을 설정할 수도 있습니다. SDK 는 위임된 인증을 수행하는 타사로부터 수신하는 모든 사용자 동의 값을 전달합니다.

사용자 동의는 서비스 공급자에서도 구성할 수 있습니다. 서비스 공급자는 아이덴티티 공급자에게 어설션 응답에 사용자 동의 값을 전달하도록 요구할 수 있습니다.

1. Administrative UI 에 로그인합니다.
2. "페더레이션", "파트너 관계 페더레이션", "파트너 관계"로 이동합니다.
3. 수정할 IdP->SP 파트너 관계를 선택합니다.
4. 파트너 관계 마법사의 "SSO 및 SLO" 단계로 이동합니다.
5. SSO 섹션에서 다음을 수행합니다.
   a. "사용자 동의 사용" 확인란을 선택합니다.
   b. "사용자 동의 Post 양식" 필드에서 사용자 지정 양식의 이름을 지정합니다.

   참고: "사용자 동의 서비스 URL"은 기본적으로 지정됩니다. 이 값은 변경할 수 없습니다.
6. 구성이 완료되면 확인 단계로 이동하고 "마침"을 클릭합니다.

사용자 동의 양식 사용자 지정(선택 사항)

제품에는 ca_defaultconsentform.html 이라는 *페더레이션 동의* 양식이 함께 제공됩니다. 아이덴티티 공급자는 어설션을 보낼 수 있는 권한을 받기 위해 대상 사용자에게 사용자 지정 양식을 보냅니다. 기본 동의 양식은 다음 위치에 있습니다.

Windows: %FEDROOT%\customization

UNIX: $FEDROOT/customization

FEDROOT 는 시스템 환경 변수입니다.

기본 동의 양식을 사용하는 대신 사용자 지정 양식을 작성할 수 있습니다.

다음 단계를 수행하십시오.

1. 사용자 지정 HTML 양식을 생성합니다. 양식을 수정하고 다음 설정에 대한 값을 바꿉니다.

   $$userconsent_spid$$

   파트너 관계에서 구성된 SP ID 를 나타냅니다.

   $$userconsent_idpid$$

   파트너 관계에서 구성된 IDP ID 를 나타냅니다.
2. 양식을 사용자 지정 디렉터리에 넣습니다.
3. "사용자 동의 Post 양식" 양식의 위치를 Administrative UI 에 지정합니다.

SP 에서 사용자 동의 요구

SP 는 IdP 에서 반환되는 어설션 응답에 사용자 동의 특성을 포함하도록 요구할 수 있습니다. 인증 요청에 이 특성을 포함하려면 Administrative UI 에서 해당 설정을 사용하도록 설정하십시오.

다음 단계를 수행하십시오.

1. Administrative UI 에 로그인합니다.
2. 적절한 SP->IdP 파트너 관계를 수정합니다.
3. 파트너 관계 마법사의 "SSO 및 SLO" 단계로 이동합니다.
4. 대화 상자의 "SSO" 섹션에서 "사용자 동의 필요" 설정을 선택합니다.

   참고: "도움말"을 클릭하면 필드, 컨트롤 및 해당되는 각 요구 사항에 대한 설명을 볼 수 있습니다.
5. 다른 변경 내용이 없으면 "확인" 단계를 선택하고 "마침"을 클릭하여 변경 내용을 저장합니다.

이제 IdP 에 보내는 인증 요청에 사용자 동의 특성이 포함됩니다.

ECP(향상된 클라이언트 또는 프록시) 프로필 개요(SAML 2.0)

ECP(향상된 클라이언트 또는 프록시) 프로필은 싱글 사인온을 위한 응용 프로그램입니다. 향상된 클라이언트는 ECP 기능을 지원하는 브라우저 또는 일부 다른 사용자 에이전트입니다. 향상된 프록시는 무선 장치용 무선 액세스 프로토콜 프록시와 같은 HTTP 프록시입니다.

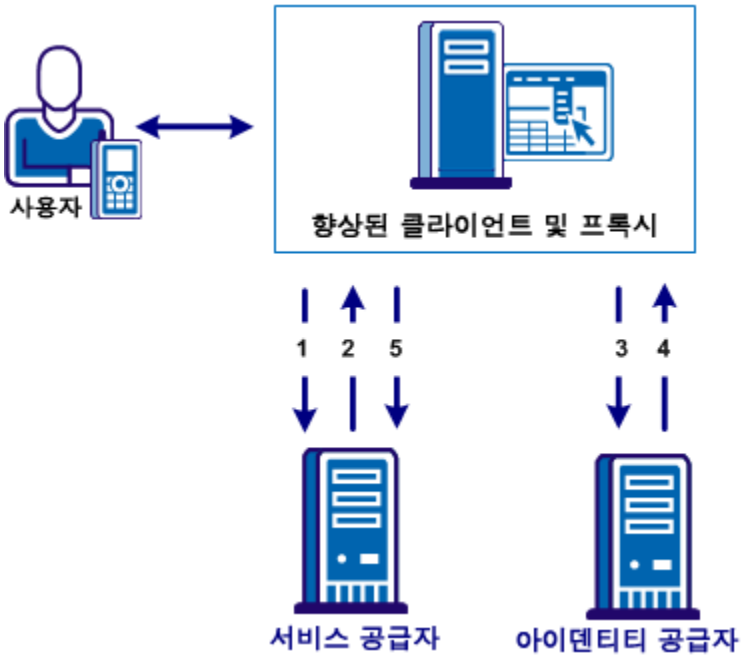
ECP 프로필은 아이덴티티 공급자와 서비스 공급자가 직접 통신할 수 없을 때 싱글 사인온을 가능하게 합니다. ECP 는 서비스 공급자와 아이덴티티 공급자 사이에서 중재자 역할을 합니다.

중재자 역할을 하는 것 외에도 ECP 프로필은 다음과 같은 경우에 유용합니다.

- 이 프로필이 필요한 향상된 클라이언트 또는 프록시에 서비스를 제공할 것으로 예상되는 서비스 공급자의 경우
- 기능이 제한된 모바일 장치 앞의 WAP(무선 액세스 프로토콜) 게이트웨이와 같은 프록시 서버가 사용되고 있는 경우

ECP 응용 프로그램은 사용자가 직접 얻거나 개발해야 합니다. CA SiteMinder?Federation Standalone 은 SAML 요구 사항에 맞는 ECP 응용 프로그램에 대한 ECP 요청 및 응답만 처리합니다.

다음 그림에서는 ECP 프로필의 흐름을 보여 줍니다.



ECP 통신에서 사용자는 휴대폰 등에서 응용 프로그램에 대한 액세스를 요청합니다. 응용 프로그램은 서비스 공급자에 있으며 사용자에 대한 아이덴티티 정보는 아이덴티티 공급자에 있습니다. 서비스 공급자와 아이덴티티 공급자는 직접 통신하지 않습니다.

이 호출의 흐름은 다음과 같습니다.

1. ECP 응용 프로그램이 리버스 SOAP(PAOS) 요청을 서비스 공급자에 전달합니다. 서비스 공급자는 아이덴티티 공급자에 직접 액세스할 수 없습니다.

   아이덴티티 공급자와 달리 ECP 엔터티에는 항상 직접 액세스할 수 있습니다.
2. 서비스 공급자는 ECP 응용 프로그램에 AuthnRequest 를 되보냅니다.
3. ECP 응용 프로그램은 AuthnRequest 를 처리 및 수정하여 아이덴티티 공급자에 보냅니다.
4. 아이덴티티 공급자는 요청을 처리한 후 ECP 응용 프로그램에 SOAP 응답을 반환합니다. 이 응답에는 어설션이 포함됩니다.
5. ECP 응용 프로그램은 서명된 PAOS 응답을 서비스 공급자에 되보냅니다.

싱글 사인온이 진행되고 사용자가 응용 프로그램에 대한 액세스 권한을 얻습니다.

아이덴티티 공급자에서 ECP 구성

ECP 를 구성하려면 아이덴티티 공급자와 서비스 공급자에서 해당 기능을 사용하도록 설정하십시오. 다음 절차는 CA SiteMinder?Federation Standalone 아이덴티티 공급자에 해당됩니다.

다음 단계를 수행하십시오.

1. Administrative UI 에 로그인합니다.
2. 수정할 로컬 아이덴티티 공급자 파트너 관계를 선택합니다.
3. 파트너 관계 마법사의 "SSO 및 SLO" 단계로 이동합니다.

4. "SSO" 섹션에서 "향상된 클라이언트 또는 프록시 프로필 사용" 확인란을 선택합니다.
5. "확인" 단계로 이동하고 "마침"을 클릭하여 변경 내용을 저장합니다.

이제 아이덴티티 공급자가 ECP 호출을 처리할 수 있습니다.

참고: 단일 서비스 공급자 개체가 싱글 사인온 요청에 대한 아티팩트, POST, SOAP 및 PAOS 바인딩을 처리할 수 있습니다. SOAP 및 PAOS 는 ECP 프로필에 대한 바인딩입니다. 아이덴티티 공급자와 서비스 공급자는 요청의 매개 변수를 기준으로 사용되는 바인딩을 확인합니다.

서비스 공급자에서 ECP 구성

ECP 를 구성하려면 아이덴티티 공급자와 서비스 공급자에서 해당 기능을 사용하도록 설정해야 합니다. 다음 절차는 서비스 공급자에 해당됩니다.

다음 단계를 수행하십시오.

1. 서비스 공급자에서 보호된 리소스에 대한 요청을 AuthnRequest 서비스에 전달합니다. URL 의 예는 다음과 같습니다.

   https://*host*:*port*/affwebservices/public/saml2authnrequest
2. Administrative UI 에 로그인합니다.
3. 관련된 로컬 서비스 공급자 파트너 관계를 수정합니다.
4. 파트너 관계 마법사의 "SSO 및 SLO" 단계로 이동합니다.
5. "SSO" 섹션에서 "향상된 클라이언트 또는 프록시 프로필 사용" 확인란을 선택합니다.
6. "확인" 단계로 이동하고 "마침"을 클릭하여 변경 내용을 저장합니다.

이제 서비스 공급자가 ECP 호출을 처리할 수 있습니다.

참고: 단일 서비스 공급자 개체가 싱글 사인온 요청에 대한 아티팩트, POST, SOAP 및 PAOS 바인딩을 처리할 수 있습니다. SOAP 및 PAOS 는 ECP 프로필에 대한 바인딩입니다. 아이덴티티 공급자와 서비스 공급자는 요청의 매개 변수를 기준으로 사용되는 바인딩을 확인합니다.

IDP 검색 프로필(SAML 2.0)

IPD(아이덴티티 공급자 검색) 프로필이 제공하는 공통 검색 서비스를 사용하면 서비스 공급자가 인증을 위해 고유 IdP 를 선택할 수 있습니다. 네트워크에 있는 모든 사이트가 아이덴티티 공급자 검색 서비스와 상호 작용하도록 파트너 간의 사전 비즈니스 계약이 설정되어 있습니다.

이 프로필은 어설션을 제공하는 파트너가 둘 이상 있는 페더레이션된 네트워크에 유용합니다. 서비스 공급자는 자신이 특정 사용자에 대한 인증 요청을 보내는 아이덴티티 공급자를 결정할 수 있습니다.

IdP 검색 프로필은 두 페더레이션된 파트너에 공통적인 쿠키 도메인을 사용하여 구현됩니다. 약정된 도메인의 쿠키에는 사용자가 방문한 IdP 목록이 포함되어 있습니다.

아이덴티티 공급자의 IDP 검색 구성

"SSO 및 SLO" 대화 상자의 "IDP 검색 섹션"에서 IDP 검색 프로필을 구성하십시오.

참고: "도움말"을 클릭하면 필드, 컨트롤 및 해당되는 각 요구 사항에 대한 설명을 볼 수 있습니다.

아이덴티티 공급자 검색 프로필이 사용되도록 설정하려면

1. "IDP 검색 사용" 확인란을 선택합니다.
2. "서비스 URL" 필드의 값을 "아이덴티티 공급자 검색 프로필" 서블릿으로 설정합니다. CA SiteMinder® Federation Standalone 의 경우 이 URL 은 다음과 같습니다.

   http://*host:port*/affwebservices/public/saml2ipd

   host

   "일반 도메인" 필드에 지정하는 일반 도메인을 나타냅니다.

   port

   CA SiteMinder® Federation Standalone 을 설치할 때 지정한 Apache HTTP 또는 HTTPS 포트를 지정합니다.

   URL 은 https 로 시작할 수도 있습니다.

3. "일반 도메인" 필드에 쿠키 도메인을 지정합니다.
4. (선택 사항) 브라우저에 영구 쿠키를 유지하려면 "영구 쿠키 사용" 확인란을 선택합니다.

IdP 에서 IdP 검색이 활성화됩니다.

서비스 공급자의 IDP 검색 구성

IDP 검색 프로필의 경우 SP(서비스 공급자)는 인증 요청을 보낼 IdP(아이덴티티 공급자)를 확인해야 합니다. SP 가 인증하려는 사용자는 이전에 아이덴티티 공급자를 방문하여 인증을 받은 상태여야 합니다.

일반 도메인 쿠키를 검색하려면 SP 가 사용자를 자체의 IdP 검색 서비스로 리디렉션해야 합니다. 쿠키에는 사용자가 이미 방문한 아이덴티티 공급자의 목록이 포함됩니다. 쿠키는 이 목록에서 올바른 IdP 를 선택한 다음 이 IdP 에 AuthnRequest 를 전송합니다.

IDP 검색 프로세스는 다음과 같습니다.

1. 브라우저가 SP 의 사이트 선택 페이지를 요청합니다.

   이 사이트 선택 페이지는 IDP 검색 서비스 URL 을 인식하고 있습니다.
2. 사이트 선택 페이지는 사용자를 IDP 검색 서비스 URL 로 리디렉션하고 일반 도메인 쿠키를 가져오고자 한다고 표시합니다.
3. IDP 검색 서비스는 일반 도메인 쿠키를 가져오고, 해당 도메인에서 쿠키를 읽고, 사용자를 다시 사이트 선택 페이지로 리디렉션합니다. 검색 서비스는 일반 도메인 쿠키를 쿼리 매개 변수로 제공합니다.
4. SP 는 사이트 선택 페이지에 사용자가 이전에 인증한 IdP URL 을 입력합니다.
5. 사용자가 IdP 를 선택하여 사용자 인증을 수행합니다.

SP 에서 IdP 검색을 구성하려면

1. SP 의 IdP 검색 서비스에서 일반 도메인 쿠키를 요청하는 사이트 선택 페이지를 생성합니다.

   CA SiteMinder® Federation Standalone 은 SP 가 IdP 검색을 구현하는 데 사용할 수 있는 IdPDiscovery.jsp 라는 샘플 사이트 선택 페이지와 함께 제공됩니다. 이 페이지는 다음 디렉터리에서 찾을 수 있습니다.

   federation_install_dir/secure-proxy/Tomcat/webapps/affwebservices/public

   첫 번째 링크는 브라우저를 한 도메인에서 일반 도메인의 IdPDiscovery 서비스로 리디렉션하고 **_saml_idp** 라는 일반 도메인 쿠키를 검색합니다. SP 의 IdP 검색 서비스는 요청을 수신하면 일반 도메인 쿠키를 가져와서 이를 쿼리 매개 변수로 추가합니다. 그런 다음 IDP 검색 서비스가 사용자를 일반 도메인의 IdPDiscovery.jsp 사이트 선택 페이지로 다시 리디렉션합니다. 기본적으로 IdPDiscovery.jsp 페이지에는 공용 쿠키에서 추출하는 IdP 에 대한 ID 목록만 표시됩니다. 이 목록은 정적이고, 연결된 IdP 와의 통신을 시작하는 목록과 연결된 HTML 링크가 없습니다.

2. SP 사이트에 대한 샘플 페이지에서 다음 링크를 편집합니다. 링크의 첫 번째 부분에서는 saml2idp 쿠키가 있는 일반 도메인을 지정합니다. 링크의 두 번째 부분에서는 IdPDiscovery.jsp 가 있는 일반 도메인을 지정합니다.

   예를 들면 다음과 같습니다.

   ```
   <a href="http://myspsystem.commondomain.com/affwebservices/public
   /saml2ipd/?IPDTarget=/http://myspsystem.spdomain.com/affwebservices
   /public/IdpDiscovery.jsp&SAMLRequest=getIPDCookie">
   Retrieve idp discovery cookie from IPD Service</a>
   ```

   사용자가 대상 사이트 선택 페이지가 있는 일반 도메인으로 다시 리디렉션되면 이제 해당 사용자가 공용 쿠키를 갖게 됩니다.

3. (선택 사항) 각 IdP 에 대해 HTML 링크를 표시하도록 IdPDiscovery.jsp 사이트 선택 페이지를 편집합니다. 각 링크는 싱글 사인온을 시작하도록 IdP 에 대한 AuthNRequest 를 트리거합니다. 기본적으로 IdPDiscovery.jsp 페이지에는 공용 쿠키에서 추출하는 IdP 에 대한 ID 목록만 표시됩니다.

4. 편집된 사이트 선택 페이지를 사용하여 IdP 검색을 테스트합니다.

IdP 검색이 제대로 작동하는 경우 사이트 선택 페이지에는 선택할 IdP 목록이 표시됩니다.

공격으로부터 IdP 검색 대상 보안

CA SiteMinder?Federation Standalone 아이덴티티 공급자 검색 서비스가 일반 도메인 쿠키에 대한 요청을 받는 경우 요청에는 IPDTarget 이라는 쿼리 매개 변수가 포함되어 있습니다. 이 쿼리 매개 변수는 검색 서비스가 요청을 처리한 후 리디렉션해야 하는 URL 을 나열합니다.

IdP 의 경우 IPDTarget 은 SAML 2.0 싱글 사인온 서비스입니다. SP 의 경우 대상은 일반 도메인 쿠키를 사용하려고 요청하는 응용 프로그램입니다.

보안 공격으로부터 IPDTarget 쿼리 매개 변수를 보호하는 것이 좋습니다. 권한 없는 사용자가 이 쿼리 매개 변수에 임의의 URL 을 넣어 악의적인 사이트로 리디렉션되도록 할 수 있습니다.

공격으로부터 쿼리 매개 변수를 보호하려면 "에이전트 구성 개체" 설정 **ValidFedTargetDomain** 을 구성하십시오. ValidFedTargetDomain 매개 변수는 페더레이션 환경에 대해 유효한 도메인을 모두 나열합니다.

참고: ValidFedTargetDomain 설정은 웹 에이전트가 사용하는 ValidTargetDomain 설정과 유사하지만 이 설정은 구체적으로 페더레이션에 대해 정의됩니다.

IPD 서비스는 IPDTarget 쿼리 매개 변수를 검사할 때 쿼리 매개 변수가 지정하는 URL 의 도메인을 확인합니다. 그런 후 IPD 서비스는 이 도메인을 ValidFedTargetDomain 매개 변수에 지정된 도메인 목록과 비교합니다. URL 도메인이 ValidFedTargetDomain 에 구성된 도메인 중 하나와 일치하면 IPD 서비스가 사용자를 지정된 URL 로 리디렉션합니다.

일치하는 도메인이 없으면 IPD 서비스가 사용자 요청을 거부하고 브라우저를 통해 "403 사용 권한 없음" 오류가 수신됩니다. 또한 FWS 추적 로그와 affwebservices 로그에서 오류가 보고됩니다. 이러한 메시지는 IPDTarget 의 도메인이 유효한 페더레이션 대상 도메인으로 정의되지 않았음을 나타냅니다.

ValidFedTargetDomain 설정을 구성하지 않는 경우 유효성 검사가 수행되지 않고 사용자가 대상 URL 로 리디렉션됩니다.

SAML 2.0 HTTP-POST 바인딩 구성

싱글 사인온 및 싱글 로그아웃 요청에 대해 요청 및 응답 교환 방법으로 SAML 2.0 HTTP-POST 바인딩을 사용할 수 있습니다. 이 바인딩은 SAML 프로토콜을 표준 메시징 형식 및 통신 프로토콜로 매핑합니다.

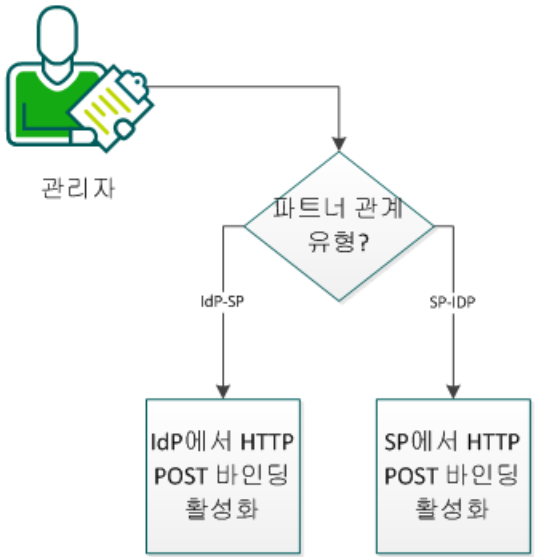
참고: 인증 요청 바인딩은 SSO 바인딩과 다릅니다. SSO 바인딩은 특정 사용 방식을 처리하기 위해 어설션, 프로토콜, 바인딩이 상호 작용하는 방법을 지정하는 프로필을 결정합니다.

이 절차를 수행하려면 페더레이션 환경에 대해 잘 알고 있어야 하며, 다음과 같은 파트너 관계 중 하나 이상을 생성되어 활성화되어 있어야 합니다.

- IdP-SP
- SP-IdP

다음 그림에서는 SAML 2.0 HTTP POST 바인딩을 사용하도록 설정하는 방법을 설명합니다.

SAML 2.0 HTTP POST 바인딩 구성 방법



다음 단계를 수행하십시오.

1. 파트너 관계 유형에 적합한 태스크를 수행하십시오.
   - IdP 에서 HTTP POST 바인딩을 활성화합니다 (페이지 250).
   - SP 에서 HTTP POST 바인딩을 활성화합니다 (페이지 251).

IdP 에서 HTTP POST 바인딩 활성화

IdP 에서 HTTP-POST 바인딩을 활성화할 수 있습니다.

중요! 인증 요청 바인딩을 구성하기 전에 세션 저장소를 활성화하십시오. IdP 가 HTTP-POST 바인딩을 사용하여 전달된 인증 요청을 처리할 수 있기 위해서는 IdP 가 세션 저장소에 요청을 저장해야 합니다.

세션 저장소가 사용되도록 설정

다음 단계를 수행하십시오.

1. 정책 서버 관리 콘솔을 열고 "데이터" 탭을 선택합니다.
2. 다음 필드를 설정합니다.

   데이터베이스

   세션 저장소

   저장소

   저장소 리포지토리를 선택합니다.

   세션 저장소 사용

   이 확인란을 선택하십시오.
3. 데이터 원본 정보를 입력합니다.
4. "확인"을 클릭하여 변경 내용을 저장합니다.

Administrative UI 에서 바인딩을 구성합니다.

다음 단계를 수행하십시오.

1. Administrative UI 를 엽니다.
2. 수정할 파트너 관계가 활성화되어 있는 경우 비활성화합니다.
3. "수정"을 클릭하여 파트너 관계 마법사를 엽니다.
4. "SSO 및 SLO" 단계로 이동합니다.
5. SSO 섹션에서 인증 요청 바인딩에 대해 "HTTP-POST"를 선택합니다.

   참고: 인증 요청에 대해 HTTP-리디렉션 및 HTTP-POST 바인딩을 모두 선택할 수 있습니다.
6. (선택 사항) SLO 섹션에서 "HTTP-POST" 확인란을 선택합니다.

   참고: 여러 SLO 바인딩을 선택할 수 있습니다.

7. SLO 바인딩과 일치하는 바인딩을 사용하여 SLO 서비스 URL 을 지정합니다. HTTP-리디렉션 및 HTTP-POST 바인딩을 선택한 경우 각 SLO 바인딩마다 하나씩 두 개의 SLO 서비스를 만드십시오.
8. 필요한 경우 다른 파트너 관계 정보를 입력합니다.
9. 확인 단계에서 "마침"을 클릭합니다.

SSO HTTP-POST 바인딩이 이제 활성화되었습니다.

SP 에서 HTTP POST 바인딩 활성화

SP 에서 인증 및 SLO 요청에 대한 HTTP-POST 바인딩을 활성화할 수 있습니다.

다음 단계를 수행하십시오.

1. Administrative UI 를 엽니다.
2. 수정할 파트너 관계가 활성화되어 있는 경우 비활성화합니다.
3. "수정"을 클릭하여 파트너 관계 마법사를 엽니다.
4. 파트너 관계 마법사의 "SSO 및 SLO" 탭으로 이동합니다.
5. SSO 섹션에서 인증 요청 바인딩에 대해 "HTTP-POST"를 선택합니다.

   참고: 인증 요청에 대해 HTTP-리디렉션 및 HTTP-POST 바인딩을 모두 선택할 수 있습니다.
6. 인증 요청 바인딩과 일치하는 바인딩을 사용하여 원격 SLO 서비스 URL 을 지정합니다. 예를 들어, HTTP-리디렉션 및 HTTP-POST 바인딩을 선택한 경우 각 바인딩마다 하나씩 두 개의 SLO 서비스 URL 을 만드십시오.
7. (선택 사항) SLO 섹션에서 "HTTP-POST" 확인란을 선택합니다.

   참고: 여러 SLO 바인딩을 선택할 수 있습니다.
8. SLO 바인딩과 일치하는 바인딩을 사용하여 SLO 서비스 URL 을 지정합니다. 예를 들어, HTTP-리디렉션 및 HTTP-POST SLO 바인딩을 선택한 경우 각 바인딩마다 하나씩 두 개의 SLO 서비스 URL 을 만드십시오.
9. 필요한 경우 다른 파트너 관계 정보를 입력합니다.
10. 확인 단계에서 "마침"을 클릭합니다.

SSO HTTP-POST 바인딩이 활성화되었습니다.

제 13 장: 소셜 사인온 구성

사용자가 페더레이션 시스템 자격 증명이 아닌 자신의 소셜 네트워킹 자격 증명을 사용하여 페더레이션된 리소스에 사인온할 수 있도록 CA SiteMinder?Federation Standalone(페더레이션 시스템)을 구성할 수 있습니다.

소셜 사인온 기능은 다음과 같은 기능으로 구성됩니다.

- 사용자가 자신의 OAuth 권한 부여 서버 자격 증명을 사용하여 페더레이션된 리소스에 사인온할 수 있도록 Facebook 같은 OAuth 권한 부여 서버를 사용한 사용자의 인증
- 사용자가 SAML 2.0 또는 Facebook 같은 여러 아이덴티티 공급자를 선택할 수 있는 자격 증명 선택기 페이지의 구성 사용자는 인증을 위한 아이덴티티 공급자를 선택하여 페더레이션된 리소스에 사인온할 수 있습니다.

이 기능은 서로 독립적이며 기능 중 하나 또는 모두를 구현하도록 페더레이션 시스템을 구성할 수 있습니다.

OAuth 권한 부여 서버를 사용한 사용자 인증

OAuth 권한 부여 서버를 사용하여 사용자를 인증하려면 페더레이션 시스템과 OAuth 권한 부여 서버 사이에 싱글 사인온을 구성하십시오.

페더레이션 시스템은 다음과 같은 OAuth 권한 부여 서버를 기본적으로 지원합니다.

OAuth 1.0a

- Twitter

OAuth 2.0

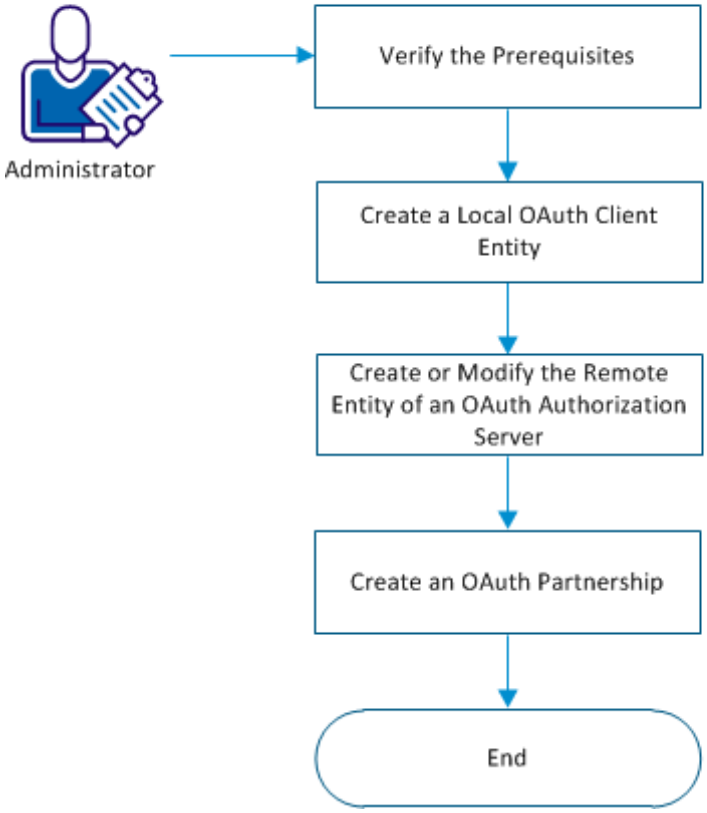
- Facebook
- Google
- LinkedIn
- Windows Live

다음 프로세스는 페더레이션 시스템이 페더레이션된 리소스에 대한 사용자 액세스 요청을 처리하는 방법을 설명합니다.

1. 페더레이션 시스템은 사용자 요청을 사용자 요청에 지정된 OAuth 권한 부여 서버에 리디렉션합니다.
2. OAuth 권한 부여 서버는 사용자를 인증하고 사용자에 대한 클레임을 포함한 인증 응답을 페더레이션 시스템에 전달합니다.
3. 페더레이션 시스템은 인증 응답을 확인하고, 인증 프로세스를 완료하고, 페더레이션된 시스템에 대한 사용자 액세스를 허가합니다.

다음 순서도는 OAuth 권한 부여 서버를 사용하여 사용자를 인증하는 방법을 설명합니다.

Authenticate Users Using an OAuth Authorization Server



다음 단계를 수행하십시오.

1. 사전 요구 사항을 확인합니다 (페이지 255).
2. 로컬 OAuth 클라이언트 엔터티를 만듭니다 (페이지 256).
3. (선택 사항) OAuth 권한 부여 서버의 원격 엔터티를 생성 또는 수정합니다 (페이지 256).
4. 싱글 사인온에 대한 OAuth 파트너 관계를 만듭니다 (페이지 258).

사전 요구 사항 확인

페더레이션 시스템과 OAuth 권한 부여 서버 사이에 싱글 사인온을 구성하기 위해 파트너 관계를 구성하기 전에 다음 단계를 수행하십시오.

- 페더레이션 시스템에서 SSL 을 활성화하십시오.
- 페더레이션 시스템이 기본적으로 지원하는 OAuth 권한 부여 서버를 사용하려면 파트너 관계를 호출하기 전에 다음 단계를 수행하십시오.
  - 독립 실행형 배포의 경우 OAuth 권한 부여 서버의 기본 CA 인증서를 가져왔는지 확인하십시오.
  - 통합 배포의 경우 smkeytool 을 사용하여 OAuth 권한 부여 서버의 기본 CA 인증서를 가져오십시오.
- 페더레이션 시스템이 기본적으로 지원하지 않는 OAuth 권한 부여 서버를 사용하려면 파트너 관계를 시작하기 전에 OAuth 권한 부여 서버의 SSL CA 인증서를 획득하고 가져오십시오.

로컬 OAuth 클라이언트 엔터티 만들기

페더레이션 시스템과 OAuth 권한 부여 서버 사이에 파트너 관계에 대한 로컬 OAuth 클라이언트 엔터티를 만듭니다.

다음 단계를 수행하십시오.

1. "페더레이션", "엔터티"로 이동하여 "엔터티 만들기"를 클릭합니다.
2. "엔터티 위치"에서 "로컬"을 선택합니다.
3. "새 엔터티 유형"에서 "OAuth 클라이언트"를 선택합니다.
4. OAuth 버전을 선택하고 "다음"을 클릭합니다.
5. 필수 값을 입력하고 "다음"을 클릭합니다.
6. 입력한 값을 확인하고 "마침"을 클릭합니다.

   리디렉션 URL 이 구성됩니다. 이 URL 을 사용하여 OAuth 트랜잭션을 시작하십시오.

권한 부여 서버의 원격 엔터티 생성 또는 수정

시스템은 기본적으로 제공되는 다음과 같은 OAuth 권한 부여 서버 각각에 대한 원격 엔터티를 제공합니다.

OAuth 1.0a

- Twitter

OAuth 2.0

- Facebook
- Google
- LinkedIn
- Windows Live

각 원격 엔터티의 값은 엔터티의 알려진 값을 사용하여 미리 구성되어 있습니다. 사용하는 페더레이션 환경에 맞게 이 값을 수정하거나 OAuth 권한 부여 서버에 대한 원격 엔터티를 만들 수 있습니다.

다음 단계를 수행하십시오.

1. 다음 작업 중 *하나*를 수행합니다.

   새 원격 엔터티를 만듭니다.

   a. "페더레이션", "엔터티", "엔터티 만들기"로 이동합니다.

   b. "엔터티 위치"로 "원격"을 선택한 다음 "새 엔터티 유형"으로 "OAuth 권한 부여 서버"를 선택합니다.

   c. "다음"을 클릭합니다.

   d. 값을 입력하고 "다음"을 클릭합니다.

   원격 엔터티의 미리 채워진 값을 수정합니다.

   a. "페더레이션", "엔터티"로 이동하여 수정할 엔터티를 검색합니다.

   b. 엔터티의 "작업" 옵션을 클릭한 다음 "수정"을 클릭합니다.

   c. "다음"을 클릭하여 "엔터티 구성" 탭으로 이동합니다.

   d. 값을 수정하고 "다음"을 클릭합니다.

2. 변경 내용을 확인하고 "마침"을 클릭합니다.

싱글 사인온에 대한 OAuth 파트너 관계 만들기

페더레이션 시스템이 권한 부여 서버로부터 사용자 정보를 가져올 수 있도록 하려면 OAuth 권한 부여 서버(어설션 당사자)와 페더레이션 시스템(신뢰 당사자) 사이에 OAuth 파트너 관계를 만드십시오.

다음 단계를 수행하십시오.

1. "페더레이션", "파트너 관계"로 이동한 다음 "파트너 관계 만들기"를 클릭합니다.
2. "OAuth 클라이언트 - 권한 부여 서버" 파트너 관계 유형을 선택합니다.
3. 파트너 관계 정보를 구성합니다.
4. 값을 확인하고 "마침"을 클릭합니다.

사용자가 OAuth 권한 부여 서버 자격 증명을 사용하여 페더레이션된 리소스에 싱글 사인온할 수 있도록 OAuth 파트너 관계가 구성됩니다.

페더레이션 시스템이 다음과 같은 형식의 사용자 요청을 받으면 이 요청을 파트너 관계 구성에 따라 처리됩니다.

https://*baseURL_of_the_partnership*/affwebservices/public/oauthtokenconsumer?AuthzServerID=*authorization_server_id*

또는

https://*baseURL_of_the_partnership*/affwebservices/public/oauthtokenconsumer/*disambiguation_id*?AuthzServerID=<*authorization_server_id*>

페더레이션 시스템이 소셜 사인온 기능을 구현하도록 구성됩니다.

OAuth 인증 체계 설정을 OAuth 파트너 관계로 마이그레이션

현재 환경에서 OAuth 공급자를 통해 사용자를 인증하도록 OAuth 인증 체계를 구성한 경우 인증 체계 설정을 페더레이션 파트너 관계로 마이그레이션할 수 있습니다.

다음 단계를 수행하십시오.

1. 다음 단계 중 하나를 수행합니다.
   - OAuth 인증 체계와 OAuth 파트너 관계를 동시에 사용하려는 경우 응용 프로그램을 OAuth 권한 부여 서버에 등록하고 다음 형식의 새 리디렉션 URL 을 기존 OAuth 인증 체계 리디렉션 URL 에 추가합니다.

     `https://server:port/affwebservices/public/oauthtokenconsumer`
   - OAuth 인증 체계 대신 OAuth 파트너 관계를 사용하려는 경우 OAuth 권한 부여 서버에서 기존 리디렉션 URL 을 다음 형식의 해당 파트너 관계 리디렉션 URL 로 업데이트합니다.

     `https://server:port/affwebservices/public/oauthtokenconsumer`

     참고: 인증 체계 리디렉션 URL 을 파트너 관계 리디렉션 URL 로 업데이트한 후에는 OAuth 인증 체계가 작동하지 않습니다.
2. OAuth 클라이언트와 OAuth 권한 부여 서버 간의 파트너 관계를 생성합니다.
3. OAuth 파트너 관계를 시작할 때 다음 URL 을 사용해야 함을 응용 프로그램 사용자에게 알립니다.

   `https://server:port/affwebservices/public/oauthtokenconsumer?AuthzServerID=AuthorizationServerID`

자격 증명 선택기 페이지 구성

사용자가 Facebook 또는 Twitter 등의 아이덴티티 공급자를 선택하여 인증할 수 있도록 파트너 관계를 구성할 수 있습니다. CA SiteMinder for Secure Proxy Server 에 설치된 자격 증명 처리 서비스를 사용하면 사용자 인증을 위해 선택할 수 있는 여러 아이덴티티 공급자가 수록된 자격 증명 선택기 페이지를 표시하도록 파트너 관계를 구성할 수 있습니다.

자격 증명 선택기 페이지를 구성하려면 다음 파트너 관계를 만드십시오.

1. 페더레이션 시스템과 아이덴티티 공급자 사이에서 싱글 사인온을 구성하기 위한 파트너 관계. 아이덴티티 공급자는 어설션 당사자로서 기능하고 페더레이션 시스템은 신뢰 당사자로서 기능합니다.
2. 페더레이션 시스템과 페더레이션된 리소스가 있는 엔터프라이즈 사이의 파트너 관계. 페더레이션 시스템은 어설션 당사자로서 기능하고 엔터프라이즈는 신뢰 당사자로서 기능합니다.

다음 프로세스는 페더레이션 시스템이 사용자 요청을 처리하는 방법을 설명합니다.

1. 엔터프라이즈(신뢰 당사자)는 사용자 요청을 페더레이션 시스템(어설션 당사자)으로 리디렉션합니다.
2. 페더레이션 시스템(어설션 당사자)은 파트너 관계가 자격 증명 선택기 페이지를 표시하도록 구성되었는지 확인합니다. 구성된 경우 사용자 인증을 위해 선택할 수 있는 여러 아이덴티티 공급자가 있는 자격 증명 선택기 페이지가 표시됩니다.
3. 사용자가 페더레이션 시스템에 등록된 경우 다음 단계가 수행됩니다. 사용자가 등록되지 않은 경우 다음 단계로 건너뜁니다.
   a. 사용자가 아이덴티티 공급자를 선택하고 이 아이덴티티 공급자에 사인온합니다.
   b. 아이덴티티 공급자가 액세스 토큰을 생성하고 사용자를 페더레이션 시스템(신뢰 당사자)으로 리디렉션합니다.
   c. 페더레이션 시스템(신뢰 당사자)이 액세스 토큰을 확인하고 사용자 저장소의 사용자를 식별하려고 시도합니다.
   d. 페더레이션 시스템(신뢰 당사자)이 세션을 생성하고 사용자를 페더레이션 시스템(어설션 당사자)으로 리디렉션합니다.
   e. 페더레이션 시스템(어설션 당사자)이 어설션을 생성하고 사용자를 엔터프라이즈(신뢰 당사자)로 리디렉션합니다.
   f. 엔터프라이즈(신뢰 당사자)가 어설션을 확인하고 페더레이션된 리소스에 대한 사용자 액세스를 허가합니다.
4. 사용자가 페더레이션 시스템에 등록되지 않은 경우 다음 단계가 수행됩니다.
   a. 사용자가 "등록" 링크를 클릭합니다.
   b. 페더레이션 시스템이 프로비저닝 서버와 파트너 관계가 구성된 아이덴티티 공급자의 목록을 표시합니다.

c. 사용자가 아이덴티티 공급자를 선택하고 이 아이덴티티 공급자에 사인온합니다.

d. 아이덴티티 공급자가 액세스 토큰을 생성하고 사용자를 페더레이션 시스템(신뢰 당사자)으로 리디렉션합니다.

e. 페더레이션 시스템(신뢰 당사자)이 액세스 토큰을 확인하고 사용자 저장소의 사용자를 식별하려고 시도합니다.

f. 페더레이션 시스템(신뢰 당사자)은 사용자를 파트너 관계에서 구성된 프로비저닝 서버로 사용자를 리디렉션합니다.

g. 프로비저닝 서버가 사용자를 만들어 페더레이션 시스템(신뢰 당사자)으로 리디렉션합니다.

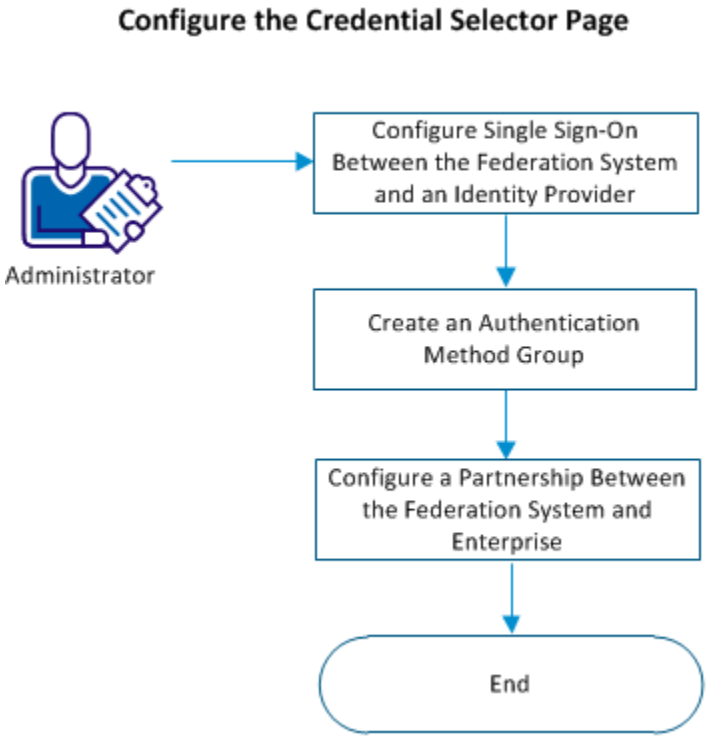
h. 페더레이션 시스템(신뢰 당사자)이 세션을 생성하고 사용자를 페더레이션 시스템(어설션 당사자)으로 리디렉션합니다.

i. 페더레이션 시스템(어설션 당사자)이 어설션을 생성하고 사용자를 엔터프라이즈(신뢰 당사자)로 리디렉션합니다.

j. 엔터프라이즈(신뢰 당사자)가 어설션을 확인하고 페더레이션된 리소스에 대한 사용자 액세스를 허가합니다.

사용자 요청이 처리됩니다.

다음 순서도는 자격 증명 선택기 페이지를 구성하는 방법을 설명합니다.



다음 단계를 수행하십시오.

1. 페더레이션 시스템과 아이덴티티 공급자 사이에서 싱글 사인온을 구성합니다. (페이지 263)
2. 인증 방법 그룹을 만듭니다 (페이지 263).
3. 페더레이션 시스템과 엔터프라이즈 사이에 파트너 관계를 구성합니다 (페이지 264).

페더레이션 시스템과 아이덴티티 공급자 사이에서 싱글 사인온 구성

자격 증명 선택기 페이지에 표시할 각 아이덴티티 공급자에 대해 아이덴티티 공급자와 페더레이션 시스템 사이에 싱글 사인온을 구성하기 위해 파트너 관계를 만드십시오. 아이덴티티 공급자는 어설션 당사자로서 기능하고 페더레이션 시스템은 신뢰 당사자로서 기능합니다.

인증에 사용할 아이덴티티 공급자는 다음 인증 프로토콜에 기반해야 합니다.

- SAML 1.1
- SAML 2.0
- WS-페더레이션
- OAuth

페더레이션 시스템이 아이덴티티 공급자로서 기능하도록 하려면 어설션 당사자와 신뢰 당사자 모두로 기능하는 시스템과 파트너 관계를 만드십시오.

다음 단계를 수행하십시오.

1. "페더레이션", "파트너 관계"로 이동합니다.
2. 자격 증명 선택기 페이지에 표시할 각 아이덴티티 공급자에 대해 파트너 관계를 만듭니다.

인증 방법 그룹 만들기

인증 방법 그룹은 자격 증명 선택기 페이지에 표시되어야 하는 아이덴티티 공급자의 목록을 정의합니다. 자격 증명 선택기 페이지에 표시할 SAML 또는 Facebook 같은 각 아이덴티티 공급자는 인증 방법 그룹의 일부여야 합니다. 인증 방법 그룹을 만들 때는 모든 파트너 관계 목록에서 어설션 당사자로 기능하는 아이덴티티 공급자를 선택할 수 있습니다.

다음 단계를 수행하십시오.

1. "인프라", "인증", "인증 방법 그룹"으로 이동합니다.
2. "인증 방법 그룹 만들기"를 클릭합니다.
3. 인증을 위해 선택할 수 있도록 표시할 아이덴티티 공급자의 파트너 관계를 추가하고 필수 값을 입력합니다.
4. 변경 내용을 저장합니다.

페더레이션 시스템과 엔터프라이즈 사이에 파트너 관계 구성

사용자가 페더레이션된 리소스에 액세스를 시도할 때 자격 증명 선택기 페이지를 표시하려면 페더레이션 시스템과 엔터프라이즈 사이에 파트너 관계를 구성하십시오. 페더레이션 시스템은 어설션 당사자로서 기능하고 엔터프라이즈는 신뢰 당사자로서 기능합니다. 파트너 관계를 만들거나 기존 파트너 관계를 수정할 수 있습니다.

파트너 관계는 다음 인증 프로토콜 중 *하나*에 기반해야 합니다.

- SAML 1.1
- SAML 2.0
- WS-페더레이션

다음 단계를 수행하십시오.

1. "페더레이션", "파트너 관계"로 이동합니다.
2. 각 단계에 값을 입력합니다.
3. "싱글 사인온" 또는 "SSO 및 SLO" 또는 "싱글 사인온 및 사인아웃" 단계에서 다음 단계를 수행합니다.
   a. "자격 증명 선택기"로 "인증 모드"를 선택합니다.
   b. "인증 기준 URL"을 정의합니다.
   c. "인증 방법 그룹"을 선택합니다.
4. "대상 응용 프로그램 구성" 단계에서 다음 필드를 선택합니다.
   - SAML 1.1: 대상
   - SAML 2.0 및 WS-페더레이션: 릴레이 상태가 대상 무시
5. 변경 내용을 저장합니다.

사용자가 페더레이션된 리소스에 액세스를 시도하면 자격 증명 선택기를 표시하도록 파트너 관계가 구성되었습니다.

페더레이션 시스템이 소셜 사인온 기능을 구현하도록 구성됩니다.

자격 증명 선택기 페이지에서 머리글 및 바닥글 사용자 지정

엔터프라이즈 요건에 맞게 자격 증명 선택기 페이지에 표시되는 머리글과 바닥글을 사용자 지정할 수 있습니다.

다음 단계를 수행하십시오.

1. 페더레이션 시스템에서 다음 위치로 이동합니다.

   <*install_path*>\CA\Federation Standalone\secure-proxy\Tomcat\webapps\chs\jsps

2. header.jsp 파일의 복사본을 만든 다음 새 파일의 이름을 header-custom.jsp 로 지정합니다.
3. footer.jsp 파일의 복사본을 만든 다음 새 파일의 이름을 footer-custom.jsp 로 지정합니다.

   참고: header-custom.jsp 및 footer-custom.jsp 파일이 있으면 머리글 및 바닥글 표시에 이 파일을 사용하도록 페더레이션 시스템이 구성됩니다.
4. 이 파일을 수정하여 자격 증명 선택기 페이지에 표시되어야 하는 머리글과 바닥글을 사용자 지정합니다.
5. 변경 내용을 저장합니다.
6. CA SiteMinder for Secure Proxy Server 를 다시 시작합니다.

   파트너 관계가 활성화되었을 때 사용자 지정된 머리글 및 바닥글이 자격 증명 선택기 페이지에 표시됩니다.

제 14 장: 위임된 인증

위임된 인증 개요

페더레이션 파트너 관계에 대해 싱글 사인온을 구성할 때 내려야 할 결정 중 하나는 사용자 인증 방법입니다.

CA SiteMinder?Federation Standalone 에서는 다음 두 가지 인증 방법을 제공합니다.

- 로컬 인증
- 위임된 인증

CA SiteMinder?Federation Standalone 에서는 로컬 인증을 수행할 수 있지만 사용할 수 있는 인증 체계는 기본 및 HTML 양식 인증 체계뿐입니다.

위임된 인증을 통해 CA SiteMinder?Federation Standalone 은 타사 WAM(웹 액세스 관리) 시스템을 사용하여 보호된 페더레이션 리소스를 요청하는 모든 사용자의 인증을 수행할 수 있습니다. 타사 WAM 시스템은 인증을 수행한 다음 페더레이션된 사용자 아이덴티티를 CA SiteMinder?Federation Standalone 에 전달합니다. CA SiteMinder?Federation Standalone 은 사용자 아이덴티티 정보를 수신한 후 자체 사용자 디렉터리에서 사용자를 찾고 신뢰 당사자와 페더레이션 프로세스를 시작합니다.

위임된 인증 요청은 어설션 당사자 측에서 수행되며 타사 WAM 시스템 또는 CA SiteMinder?Federation Standalone 에서 시작될 수 있습니다. 인증 요청은 신뢰 당사자 측에서 시작될 수 있지만 이는 위임된 인증으로 간주되지 않습니다.

인증은 다음과 같이 시작될 수 있습니다.

어설션 당사자 측에서 CA SiteMinder® Federation Standalone 에 의해 시작되는 인증

CA SiteMinder® Federation Standalone 은 어설션 당사자 측에서 인증 요청을 시작할 수 있습니다. 요청이 CA SiteMinder® Federation Standalone 으로 전송되면 이 요청은 위임된 인증 요청으로 인식됩니다. 그러면 CA SiteMinder® Federation Standalone 은 사용자를 타사 WAM 시스템으로 리디렉션합니다.

어설션 당사자 측에서 WAM 시스템에 대한 직접 로그인으로 시작된 인증

사용자가 어설션 당사자 측에서 WAM 시스템에 로그인하면 인증 요청이 시작됩니다. WAM 시스템이 사용자를 성공적으로 인증하면 아이덴티티 정보가 CA SiteMinder® Federation Standalone 으로 전달됩니다.

신뢰 당사자 측에서 시작된 인증

신뢰 당사자는 인증 요청을 시작할 수 있지만 이 시나리오는 위임된 인증으로 간주되지 않습니다. 위임된 인증은 어설션 당사자 측에서만 수행됩니다.

페더레이션된 리소스에 대한 요청은 신뢰 당사자로 직접 전송되며 신뢰 당사자는 AuthnRequest 를 어설션 당사자의 CA SiteMinder® Federation Standalone 으로 전송합니다. CA SiteMinder® Federation Standalone 은 이를 위임된 인증 요청으로 인식하고 사용자를 어설션 당사자의 타사 WAM 시스템으로 리디렉션합니다. 사용자는 WAM 시스템에 로그인하고 이 시스템이 인증 요청을 시작합니다. WAM 시스템이 사용자를 성공적으로 인증하면 아이덴티티 정보가 CA SiteMinder® Federation Standalone 으로 전달됩니다.

타사 WAM 시스템은 인증 요청을 받은 후 사용자 아이덴티티를 CA SiteMinder?Federation Standalone 으로 전달합니다. WAM 시스템이 사용자 아이덴티티를 전달하는 데 사용하는 방법은 위임된 인증 방법이 쿠키 기반인지 쿼리 문자열 기반인지에 따라 달라집니다.

타사 WAM 이 사용자 아이덴티티를 전달하는 방법

타사 WAM 시스템은 다음 두 방법 중 하나를 사용하여 페더레이션된 사용자 아이덴티티를 CA SiteMinder?Federation Standalone 으로 전달할 수 있습니다.

- 레거시 쿠키 또는 개방 형식 쿠키를 사용합니다.

  데이터의 보안을 위해 개방 형식 쿠키를 암호화할 수 있습니다.

- 브라우저를 CA SiteMinder® Federation Standalone 으로 보내는 리디렉션 URL 에 추가되는 쿼리 문자열을 사용합니다.

  쿼리 문자열은 일반 텍스트로 전송되며 FIPS 호환 파트너 관계가 생성되지 않습니다.

  중요! 프로덕션 환경에서는 쿼리 문자열 방법을 사용하지 마십시오. 쿼리 문자열 리디렉션 방법은 테스트 환경에서 개념 증명용으로만 사용해야 합니다.

타사 WAM 시스템이 선택하는 방법은 사용자 아이덴티티를 CA SiteMinder?Federation Standalone 으로 전달하기 위해 시스템에서 설정하려는 구성에 따라 달라집니다.

사용자 아이덴티티를 전달하는 방법은 다음 단원에서 자세히 설명합니다.

사용자 아이덴티티를 전달하기 위한 쿠키 방법

CA SiteMinder?Federation Standalone 에서는 사용자 아이덴티티를 전달하는 데 레거시 또는 개방 형식 쿠키를 사용할 수 있습니다. 쿠키에는 사용자 로그인 ID 가 값의 하나로 포함됩니다.

참고: CA SiteMinder?Federation Standalone Agent for Windows Authentication 에 사용할 위임된 인증을 구성할 경우 에이전트에서는 개방 형식 쿠키를 사용해야 합니다. 그러나 SiteMinder 커넥터도 구성되어 있는 경우에는 위임된 인증에 대해 개방 형식 쿠키 옵션을 사용할 수 없습니다. CA SiteMinder?Federation Standalone Windows Agent 와 SiteMinder 커넥터는 하나의 배포 환경에서 공존할 수 없습니다.

인증은 WAM 시스템 또는 CA SiteMinder?Federation Standalone 에서 시작될 수 있습니다. CA SiteMinder?Federation Standalone 에서 인증이 시작되는 경우, 사용자가 WAM 시스템으로 리디렉션되며 WAM 시스템에서 시작되는 경우와 동일하게 인증 프로세스가 진행됩니다.

위임된 인증 프로세스는 다음과 같습니다.

1. 인증 요청이 타사 WAM 시스템에 수신됩니다.
2. 사용자가 인증됩니다.
3. 타사 WAM 시스템은 다음 두 방법 중 하나로 쿠키를 가져옵니다.
   - WAM 시스템은 CA SiteMinder® Federation Standalone SDK 를 사용하여 레거시 쿠키 또는 개방 형식 쿠키를 생성합니다. SDK 가 쿠키를 생성하여 WAM 시스템에 대한 요청으로 다시 전송합니다.

     참고: FIPS 암호화 개방 형식 쿠키를 생성하려면 CA SiteMinder® Federation Standalone SDK 를 사용하십시오.

     타사 WAM 응용 프로그램은 쿠키를 생성하기 위해 사용하는 SDK 와 동일한 언어를 사용해야 합니다. CA SiteMinder® Federation Standalone Java SDK 를 사용하는 경우 타사 WAM 응용 프로그램은 Java 로 작성되어야 합니다. .NET SDK 를 사용하는 경우 타사 WAM 응용 프로그램이 .NET 을 지원해야 합니다.
   - WAM 시스템은 수동으로 생성한 개방 형식 쿠키를 사용합니다.

     CA SiteMinder® Federation Standalone SDK 를 사용하지 않고 개방 형식 쿠키를 생성할 수 있습니다. 개방 형식 쿠키를 수동으로 생성하려면 UTF-8 인코딩을 지원하는 프로그래밍 언어와 CA SiteMinder® Federation Standalone 에서 암호 기반 암호화를 위해 사용하는 다음의 PBE 암호화 알고리즘을 사용하십시오.
     - PBE/SHA1/AES/CBC/PKCS12PBE-1000-128
     - PBE/SHA1/AES/CBC/PKCS12PBE-1000-192
     - PBE/SHA1/AES/CBC/PKCS12PBE-1000-256
     - PBE/SHA256/AES/CBC/PKCS12PBE-1000-128
     - PBE/SHA256/AES/CBC/PKCS12PBE-1000-192
     - PBE/SHA256/AES/CBC/PKCS12PBE-1000-256
     - PBE/SHA1/3DES_EDE/CBC/PKCS12PBE-1000-3
     - PBE/SHA256/3DES_EDE/CBC/PKCS12PBE-1000-3

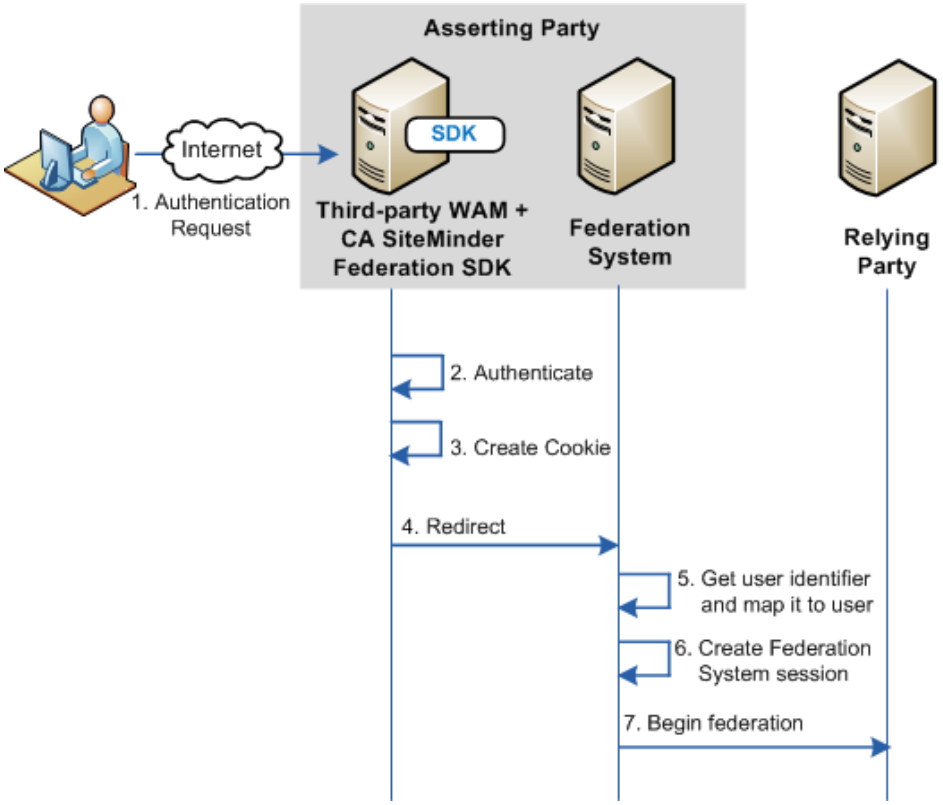
     이때 사용자의 브라우저에 개방 형식 쿠키가 설정되도록 해야 합니다.

     완전한 쿠키를 작성하려면 개방 형식 쿠키의 콘텐츠 (페이지 469)에 대한 자세한 정보를 참조하십시오.

   참고: WAM 시스템과 CA SiteMinder® Federation Standalone 은 동일한 쿠키 도메인에 있어야 합니다.

4. WAM 시스템이 브라우저를 CA SiteMinder® Federation Standalone 으로 리디렉션합니다.
5. CA SiteMinder® Federation Standalone 은 쿠키에서 로그인 ID 를 추출한 다음 자체 사용자 디렉터리에서 사용자를 찾습니다.
6. CA SiteMinder® Federation Standalone 에서 CA SiteMinder® Federation Standalone 세션을 생성합니다.
7. 세션이 생성되면 신뢰 당사자와의 페더레이션 통신이 진행됩니다.

다음 그림은 인증이 타사 WAM 에서 시작되었을 때의 쿠키 방법을 보여 줍니다.



중요! 레거시 쿠키 또는 SDK 를 통해 생성한 개방 형식 쿠키를 사용하려면 타사에서 CA SiteMinder® Federation Standalone SDK 를 설치해야 합니다. SDK 는 CA SiteMinder® Federation Standalone 과는 별도로 설치되는 구성 요소입니다. 설치 키트에는 위임된 인증에 SDK 를 사용하는 방법을 설명하는 문서가 포함되어 있습니다.

사용자 아이덴티티를 전달하기 위한 쿼리 문자열 방법

타사 WAM 시스템은 사용자를 WAM 시스템에서 CA SiteMinder?Federation Standalone 으로 보내는 리디렉션 URL 에 쿼리 문자열을 추가하여 사용자 아이덴티티를 CA SiteMinder?Federation Standalone 에 전달할 수 있습니다. 이 방법이 작동하려면 타사 WAM 시스템이 페더레이션된 사용자가 인증된 후에 이러한 사용자를 CA SiteMinder?Federation Standalone 으로 리디렉션하는 URL 을 구성해야 합니다.

중요! 프로덕션 환경에서는 쿼리 문자열 방법을 사용하지 마십시오. 쿼리 문자열 리디렉션 방법은 테스트 환경에서 개념 증명용으로만 사용해야 합니다.

참고:

- 쿼리 문자열 방법은 FIPS 호환 파트너 관계를 생성하지 않습니다.
- 인증은 CA SiteMinder® Federation Standalone 또는 신뢰 당사자 측에서 시작될 수도 있습니다.

인증이 WAM 시스템에서 시작되는 경우 쿼리 문자열을 사용하는 위임된 인증의 트랜잭션 흐름은 다음과 같습니다.

1. 타사 WAM 시스템이 인증 요청을 수신합니다.
2. 사용자가 인증됩니다.
3. 타사 WAM 시스템은 리디렉션 URL 을 구성하고 로그인 ID 와 해시된 로그인 ID 값을 LoginID=*LoginID*&LoginIDHash=*hashed_LoginID* 의 형식으로 쿼리 문자열에 추가합니다.

   중요! LoginID 및 LoginIDHash 매개 변수는 대/소문자를 구분합니다. 이 매개 변수를 예제에 나오는 그대로 리디렉션 URL 에 포함하십시오.

   CA SiteMinder® Federation Standalone 은 해싱 메커니즘을 통해 사용자 ID 가 변경되지 않고 수신되었음을 확인할 수 있습니다.

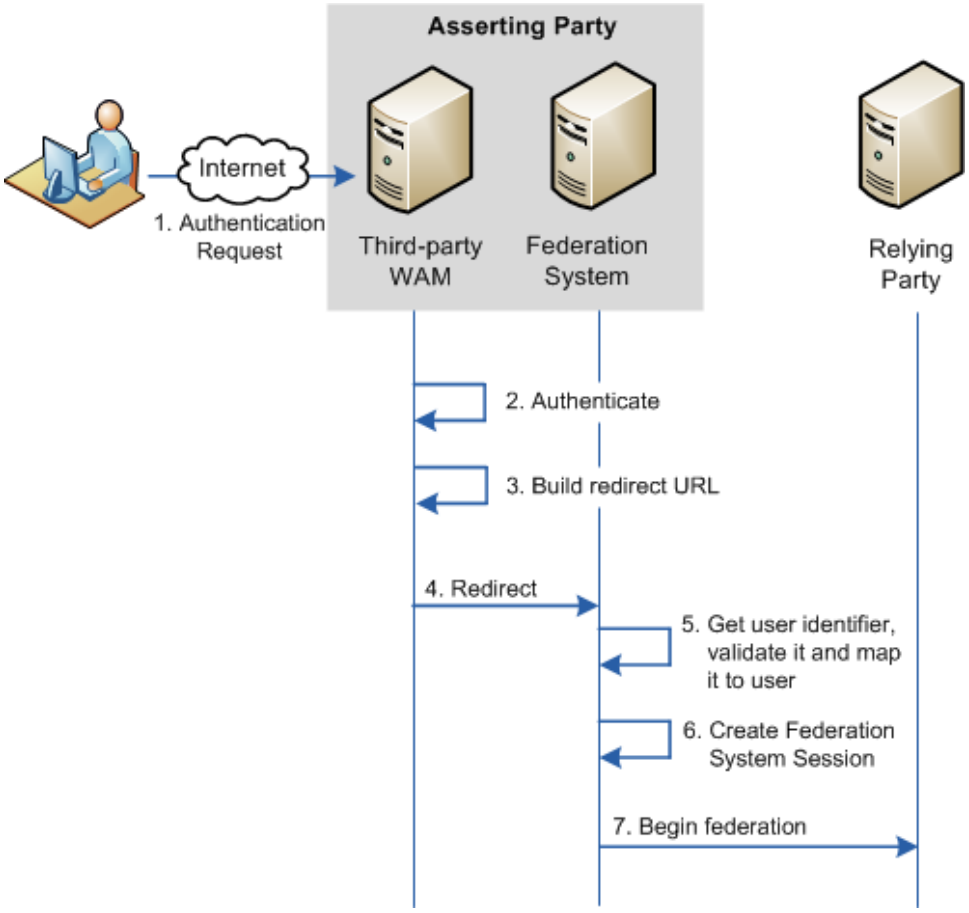
   리디렉션 URL 의 예

   http://idp1.example.com:9090/affwebservices/public/saml2sso?SPID=FmSP&ProtocolBinding=urn:oasis:names:tc:SAML:2.0:bindings:HTTP-POST&LoginID=jdoe&LoginIDHash=454d3bd5cb839168eeffcf060ae0b9c28ed6eec0

4. WAM 시스템이 브라우저를 CA SiteMinder® Federation Standalone 으로 리디렉션합니다.

5. CA SiteMinder® Federation Standalone 은 URL 에서 로그인 ID 와 해시된 로그인 ID 를 추출하고 해시된 값을 사용하여 식별자의 유효성을 검사한 다음 자체 사용자 디렉터리에서 사용자를 찾습니다.
6. CA SiteMinder® Federation Standalone 이 사용자 세션을 생성합니다.
7. 세션이 생성되면 신뢰 당사자와의 페더레이션 통신이 진행됩니다.

다음 그림은 인증이 어설션 당사자 측에서 시작될 때의 쿼리 문자열 방법을 보여 줍니다.



위임된 인증 구성

위임된 인증은 인증된 사용자 아이덴티티를 기반으로 어설션이 생성되는 어설션 당사자 측에서 구성됩니다.

위임된 인증을 구성하려면

1. 타사 WAM 이 사용자 아이덴티티를 전달하기 위해 사용하는 방법(쿠키 또는 쿼리 문자열)을 확인합니다.

   참고: 쿼리 문자열은 FIPS 호환 파트너 관계를 생성하지 않습니다.

2. 파트너 관계 마법사의 적절한 단계로 이동하여 위임된 인증을 설정합니다.

중요! SDK 로 생성한 개방 형식 쿠키를 사용하려면 타사에서 CA SiteMinder® Federation Standalone SDK 를 설치해야 합니다. SDK 는 별도로 설치되는 구성 요소입니다. 설치 키트에는 위임된 인증에 SDK 를 사용하는 방법을 설명하는 문서가 포함되어 있습니다.

추가 정보

배포 설정 (페이지 398)

쿠키 위임된 인증 샘플 설정

다음 샘플 구성은 SAML 2.0 IdP > SP 파트너 관계 관점에서의 샘플입니다. 위임된 인증 설정은 파트너 관계 마법사의 SSO 및 SLO 단계에 있습니다.

이 샘플 구성은 SAML 2.0 구성을 반영합니다. 아이덴티티 공급자는 http://idp1.xyz.com 이며 타사 WAM 시스템은 http://wamservice.xyz.com 입니다.

쿠키 위임된 인증을 구성하려면

1. 파트너 관계를 생성하거나 기존 파트너 관계를 편집합니다.

   참고: 파트너 관계를 편집하려면 먼저 비활성화하십시오.

2. 파트너 관계 마법사의 "SSO 및 SLO" 단계로 이동합니다.

3. "인증" 섹션에서 다음과 같이 필드를 설정합니다.

   인증 모드

   위임됨

   위임된 인증 유형

   개방 형식 쿠키

   웹 액세스 관리 응용 프로그램에 사용하기 위한 것입니다. CA SiteMinder® Federation Standalone SDK 를 사용하여 Java 또는 .NET 응용 프로그램을 생성할 수 있습니다. 또는 개방 형식 쿠키를 수동으로 만든 경우 다른 언어로 작성된 응용 프로그램을 사용할 수 있습니다.

   FIPS 140-2 암호화가 필요한 경우 CA SiteMinder® Federation Standalone Java 또는 .NET SDK 를 사용하여 개방 형식 쿠키를 만드십시오.

   위임된 인증 URL

   http://wamservice.xyz.com

   사용자를 인증하고 CA SiteMinder® Federation Standalone SDK 를 사용하여 쿠키를 만드는 타사 WAM 시스템의 URL 입니다.

   인증 클래스

   타사에서 사용되는 인증 방법을 입력합니다. 예를 들면 다음과 같습니다.

   urn:oasis:names:tc:SAML:2.0:ac:classes:Kerberos

4. 모든 개방 형식 쿠키 설정을 타사 WAM 시스템으로 전달합니다.

   SiteMinder 는 쿠키를 생성할 때 이 값을 사용합니다.

5. 파트너 구성을 계속합니다.

쿼리 문자열 위임된 인증 샘플 설정

다음 샘플 구성은 SAML 2.0 IdP > SP 파트너 관계 관점에서의 샘플입니다. 위임된 인증 설정은 파트너 관계 마법사의 SSO 및 SLO 단계에 있습니다.

참고: 쿼리 문자열 방법은 FIPS 호환 파트너 관계를 생성하지 않습니다.

이 샘플 구성은 SAML 2.0 구성을 반영합니다. 아이덴티티 공급자는 http://idp1.xyz.com 이며 타사 WAM 시스템은 http://wamservice.xyz.com 입니다.

중요! 프로덕션 환경에서는 쿼리 문자열 방법을 사용하지 마십시오. 쿼리 문자열 리디렉션 방법은 테스트 환경에서 개념 증명용으로만 사용해야 합니다.

쿼리 문자열 위임된 인증을 구성하려면

1. 파트너 관계를 생성하거나 기존 파트너 관계를 편집합니다.

   참고: 파트너 관계를 편집하려면 먼저 비활성화하십시오.

2. 파트너 관계 마법사의 적절한 단계로 이동합니다.
3. "인증" 섹션에서 다음과 같이 필드를 설정합니다.

   인증 모드

   위임됨

   위임된 인증 유형

   쿼리 문자열

   위임된 인증 URL

   http://wamservice.xyz.com

   사용자를 인증하고 쿼리 매개 변수가 포함된 SiteMinder 로의 리디렉션 URL 을 구성하는 타사 WAM 시스템의 URL 입니다.

   해시 암호

   FederatedAuth1

   타사 WAM 시스템은 이 암호를 사용하여 로그인 ID 를 해시합니다.

   해시 암호 확인

   FederatedAuth1

   인증 클래스

   타사에서 사용되는 인증 방법을 입력합니다. 예를 들면 다음과 같습니다.

   urn:oasis:names:tc:SAML:2.0:ac:classes:Kerberos

4. 파트너 구성을 계속합니다.

쿠키 위임된 인증에 대한 타사 WAM 구성

위임된 인증에 성공하려면 타사 WAM 이 다음과 같이 페더레이션된 응용 프로그램을 조정해야 합니다.

- 인증된 사용자 로그인 ID 를 쿠키를 통해 통신하려면 타사 WAM 시스템이 쿠키를 생성해야 합니다.
  - Java 응용 프로그램의 경우 WAM 은 CA SiteMinder® Federation Standalone Java SDK 를 사용하여 레거시 쿠키 또는 개방 형식 쿠키를 생성할 수 있습니다.
  - .NET 응용 프로그램의 경우 WAM 은 CA SiteMinder® Federation Standalone .NET SDK 를 사용하여 개방 형식 쿠키를 생성할 수 있습니다.
  - Java 및 .NET 이외의 언어인 경우 WAM 은 개방 형식 쿠키를 수동으로 생성할 수 있습니다.

  필요한 클래스 및 방법의 구현에 대한 자세한 내용은 *CA SiteMinder® Federation Standalone Java SDK 안내서* 또는 *CA SiteMinder® Federation Standalone .NET SDK 안내서* 를 참조하십시오.각 안내서는 SDK 와 함께 설치됩니다. 개방 형식 쿠키를 수동으로 생성하는 경우 쿠키의 필수 콘텐츠 (페이지 469)에 대한 자세한 내용을 검토하십시오.
- 타사에서는 CA SiteMinder® Federation Standalone 어설션 당사자에 구성되어 있는 Administrative UI 설정인 "전역 쿠키 영역" 및 "암호화 암호" 매개 변수의 값을 알고 있어야 합니다.
  - 전역 쿠키 영역
  - 암호화 암호
  - 개방 형식 쿠키 이름
  - 개방 형식 쿠키 암호화 변환

  이러한 값은 쿠키를 생성할 때 사용됩니다.

- 타사 WAM 시스템은 사용자를 다시 CA SiteMinder® Federation Standalone 으로 보내는 리디렉션 URL 을 생성해야 합니다. 이 URL 은 사용자를 다시 CA SiteMinder® Federation Standalone 싱글 사인온 서비스로 보내야 합니다. 따라서 CA SiteMinder® Federation Standalone 관리자는 대역 외 통신을 통해 싱글 사인온 서비스를 타사로 전달해야 합니다.

  중요! 타사 WAM 시스템은 CA SiteMinder® Federation Standalone 에서 인증 요청을 받은 후 인증 요청의 일부로 받은 모든 기존 쿼리 문자열을 캡처하고 다시 전송해야 합니다. 들어오는 요청의 쿼리 문자열 내에 CA SiteMinder® Federation Standalone 요청 정보가 있을 수 있으며 요청은 변경 없이 전달되어야 합니다.

  참고: 쿠키를 전달하려면 타사 WAM 시스템은 어설션 당사자의 CA SiteMinder® Federation Standalone 과 동일한 쿠키 도메인에 있어야 합니다.

쿼리 문자열 인증에 대한 타사 WAM 구성

타사 WAM 시스템과 어설션 당사자의 CA SiteMinder?Federation Standalone 은 쿼리 문자열을 통해 로그인 ID 를 전달합니다. WAM 시스템은 리디렉션 URL 에서 쿼리 문자열에 다음 두 개의 특성을 추가해야 합니다.

LoginID

타사 WAM 시스템에서 사용자를 식별하는 데 사용되는 값을 지정합니다.

LoginIDHash

LoginID 의 해시입니다.

LoginIDHash 값을 생성할 때는 LoginID 가 해시 암호에 추가된 다음 전체 값에 SHA-1 해싱 알고리즘이 적용됩니다. 해시 암호는 어설션 당사자의 CA SiteMinder® Federation Standalone 구성에서 지정됩니다.

CA SiteMinder® Federation Standalone 은 쿼리 문자열에서 자격 증명을 가져올 때 이 값을 결합하고 해시합니다. 해시가 동일하면 CA SiteMinder® Federation Standalone 은 로그인 ID 를 유효한 것으로 간주하고 페더레이션 요청을 계속 진행합니다.

중요! LoginID 및 LoginIDHash 매개 변수는 대/소문자를 구분합니다.

타사 WAM 시스템은 페더레이션된 응용 프로그램을 구성하여 사용자를 다시 CA SiteMinder?Federation Standalone 싱글 사인온 서비스로 보내는 리디렉션 URL 을 구성해야 합니다. 따라서 CA SiteMinder?Federation Standalone 관리자는 대역 외 통신을 통해 싱글 사인온 서비스를 타사로 전달해야 합니다.

중요! 타사 WAM 시스템은 CA SiteMinder® Federation Standalone 에서 인증 요청을 받은 후 해당 인증 요청의 일부로 받은 모든 기존 쿼리 문자열을 캡처하고 다시 전송해야 합니다. 들어오는 요청의 쿼리 문자열 내에 CA SiteMinder® Federation Standalone 요청 정보가 있는 경우 이를 변경하지 않고 전달해야 합니다.

쿼리 문자열의 구문은 다음과 같습니다.

?*existing_query_string*&LoginID=*LoginID*&LoginIDHash=*hashed_LoginID*

예

```
https://johndoe3227.b.com/affwebservices/public/saml2sso?SPID=sp1&
LoginID=user1&LoginIDHash=de164152ed6e8e9a7f760e47d135ecf0c98a
3e4e&ProtocolBinding=urn:oasis:names:tc:SAML:2.0:bindings:HTTP-Artifact
```

제 15 장: 싱글 사인온을 시작하기 위한 URL

싱글 사인온을 시작하는 서블릿에 대한 링크

페더레이션된 콘텐츠의 사이트를 설계할 경우 해당 사이트에는 싱글 사인온을 트리거하는 특정 링크가 있는 페이지가 포함됩니다. 이러한 링크는 싱글 사인온 서비스 또는 AuthnRequest 서비스에 대한 서블릿의 URL 입니다.

싱글 사인온을 시작하려면 사용자는 어설션 당사자 또는 신뢰 당사자 측에서 시작할 수 있습니다. 각 사이트에서 적절한 링크를 구성하여 싱글 사인온 작업을 시작합니다.

생산자에서 시작되는 SSO(SAML 1.1)

생산자에서 사용자를 소비자 사이트에 연결하는 링크가 포함된 페이지를 생성하십시오. 각 링크는 사이트 간 전송 URL 을 나타냅니다. 사용자는 사이트 간 전송 URL 을 방문해야 합니다. URL 은 사용자가 소비자 사이트로 리디렉션되기 전에 생산자 측 웹 에이전트에 요청을 보냅니다.

SAML 아티팩트 및 POST 프로필의 경우 사이트 간 전송 URL 의 구문은 다음과 같습니다.

```
http://producer_host:port/affwebservices/public/intersitetransfer?
CONSUMERID=consumer_entity_ID&TARGET=http://consumer_site/target_url
```

이 사이트 간 전송 URL 의 변수 및 쿼리 매개 변수는 다음과 같습니다.

producer_host:port

사용자가 인증되는 서버 및 포트 번호를 지정합니다.

CONSUMERID

(필수) 소비자를 식별합니다. 생산자 측에서 생산자-소비자 파트너 관계에는 이름이 있고 원격 소비자 엔터티에는 ID 가 있습니다. CONSUMERID 는 원격 소비자의 엔터티 ID 입니다. 엔터티 ID 는 대/소문자를 구분합니다. "관리 UI"에 표시되는 그대로 입력하십시오.

CONSUMERID 대신 매개 변수 NAME 을 사용할 수 있지만 둘 다 사용할 수는 없습니다.

NAME 을 사용할 때는 생산자에서 정의된 생산자-소비자 파트너 관계의 이름을 지정하십시오.

consumer_entity_ID

사용자가 생산자 사이트에서 방문하려는 소비자 사이트를 나타냅니다. 엔터티 ID 는 대/소문자를 구분합니다. "관리 UI"에 표시되는 그대로 입력하십시오.

TARGET

(선택 사항) 소비자에서 요청된 대상 리소스를 나타냅니다.

TARGET 매개 변수는 선택 사항입니다. 대상을 정의해야 하지만 사이트 간 전송 URL 이 아니라 소비자 측 파트너 관계에서 정의할 수 있습니다. 대상은 파트너 관계 마법사의 "응용 프로그램 통합" 단계에서 정의됩니다. URL 또는 파트너 관계에서 대상을 정의하십시오.

consumer_site

소비자 측에서 서버를 지정합니다.

target_url

소비자 측의 대상 응용 프로그램을 지정합니다.

참고: SAML 아티팩트 바인딩에 대한 쿼리 매개 변수는 HTTP-인코딩을 사용해야 합니다.

아티팩트 및 POST 프로필에 대한 사이트 간 전송 URL 의 예는 다음과 같습니다.

```
http://www.smartway.com/affwebservices/public/intersitetransfer?
CONSUMERID=ahealthco&TARGET=http://www.ahealthco.com:85/
smartway/index.jsp
```

IdP 에서 시작되는 SSO(SAML 2.0 아티팩트 또는 POST)

사용자가 서비스 공급자로 가기 전에 CA SiteMinder?Federation Standalone 아이덴티티 공급자를 방문할 경우에는 아이덴티티 공급자에서 원치 않는 응답이 시작되어야 합니다. 원치 않는 응답을 시작하려면 CA SiteMinder?Federation Standalone 이 허용하는 HTTP Get 요청을 생성하는 하드 코딩된 링크를 생성하십시오. 이 HTTP Get 요청에는 서비스 공급자 ID 를 제공하는 쿼리 매개 변수가 포함되어야 합니다. 아이덴티티 공급자는 SAML 어설션 응답을 생성해야 합니다. 사용자는 이 링크를 클릭하여 원치 않는 응답을 시작합니다.

참고: 이 정보는 아티팩트 또는 POST 바인딩에 적용됩니다.

원치 않는 응답에서 아티팩트 또는 POST 프로필이 사용되도록 지정하려는 경우 원치 않는 응답 링크의 구문은 다음과 같습니다.

http://*idp_server:port*/affwebservices/public/saml2sso?SPID=*SP_ID*&ProtocolBinding=*URI_for_binding*&RelayState=*target_URL*

idp_server:port

CA SiteMinder® Federation Standalone 을 호스트하는 웹 서버 및 포트를 식별합니다.

SP_ID

파트너 관계에서 정의된 서비스 공급자의 엔터티 ID 를 지정합니다.

URI_for_binding

ProtocolBinding 요소에 대한 POST 또는 아티팩트 바인딩의 URI 를 식별합니다. 이 URI 는 SAML 2.0 사양에 의해 정의됩니다.

- SAML 2.0 사양으로 지정된 아티팩트 바인딩 URI 는 다음과 같습니다.

  urn:oasis:names:tc:SAML:2.0:bindings:HTTP-Artifact

- SAML 2.0 사양으로 지정된 POST 바인딩 URI 는 다음과 같습니다.

  urn:oasis:names:tc:SAML:2.0:bindings:HTTP-POST

  HTTP-POST 싱글 사인온에 대해서는 이 매개 변수를 설정할 필요가 없습니다.

참고: 요청이 작동하려면 파트너 관계에 바인딩을 사용할 수 있어야 합니다.

target_URL

서비스 공급자에서 페더레이션 리소스 대상의 URL 을 지정합니다.

다음에 주의하십시오.

- 링크에 ProtocolBinding 쿼리를 포함하지 않은 경우 서비스 공급자 속성에 구성된 바인딩 하나를 사용하십시오.
- 아티팩트 및 POST 가 서비스 공급자 속성에서 사용하도록 설정된 경우에는 POST 가 기본값입니다. 따라서 아티팩트 바인딩만 사용하려면 링크에 ProtocolBinding 쿼리 매개 변수를 포함하십시오.

중요! 어설션 소비자 서비스에 대해 인덱싱된 끝점 지원을 구성하면 ProtocolBinding 쿼리 매개 변수의 값이 어설션 소비자 서비스에 대한 바인딩을 무시합니다.

IdP 에서 사용되는 원치 않는 응답 쿼리 매개 변수

IdP 에서 싱글 사인온을 시작하는 원치 않는 응답에는 다음 쿼리 매개 변수가 포함될 수 있습니다.

SPID

(필수) 아이덴티티 공급자가 원치 않는 응답을 보내는 서비스 공급자의 ID 를 지정합니다. 엔터티 ID 는 대/소문자를 구분합니다. "관리 UI"에 표시되는 그대로 입력하십시오.

ProtocolBinding

원치 않는 응답의 ProtocolBinding 요소를 지정합니다. 이 요소는 서비스 공급자에게 어설션 응답을 보내기 위한 프로토콜을 지정합니다. 서비스 공급자가 지정된 프로토콜 바인딩을 지원하도록 구성되어 있지 않으면 요청이 실패합니다.

RelayState

서비스 공급자에서 대상 리소스의 URL 을 나타냅니다. 이 쿼리 매개 변수를 포함하면 IdP 에 사용자를 서비스 공급자의 적절한 리소스로 리디렉션하도록 지시하는 것입니다. 이 쿼리 매개 변수는 싱글 사인온을 구성할 때 대상 URL 을 지정하는 대신 사용할 수 있습니다.

ProtocolBinding 쿼리 매개 변수의 필수 사용

ProtocolBinding 매개 변수는 서비스 공급자 속성에서 아티팩트 및 POST 바인딩이 사용되도록 설정한 *경우에만* 필요합니다. 또한 사용자는 아티팩트 바인딩만 사용하려고 합니다.

- 아티팩트 바인딩에 대한 URI 는 다음과 같습니다.

  urn:oasis:names:tc:SAML:2.0:bindings:HTTP-Artifact

- POST 바인딩에 대한 URI 는 다음과 같습니다.

  urn:oasis:names:tc:SAML:2.0:bindings:HTTP-POST

  HTTP-POST 싱글 사인온에 대해서는 이 매개 변수를 설정할 필요가 없습니다.

참고: 쿼리 매개 변수 HTTP 코딩은 필요하지 않습니다.

ProtocolBinding 쿼리 매개 변수의 선택적 사용

ProtocolBinding 쿼리 매개 변수를 사용하지 *않는* 경우 다음 정보가 적용됩니다.

- 서비스 공급자에 대해 하나의 바인딩만 사용되도록 지정되고 원치 않는 응답에 ProtocolBinding 이 지정되지 않은 경우에는 사용되도록 지정된 바인딩이 사용됩니다.
- 서비스 공급자에 대해 두 바인딩이 모두 사용되도록 지정되었고 원치 않는 응답에 ProtocolBinding 이 지정되지 않은 경우에는 POST 바인딩이 기본값입니다.

예: ProtocolBinding 이 없는 원치 않는 응답

링크가 사용자를 싱글 사인온 서비스로 리디렉션합니다. 이 링크에는 SPID 쿼리 매개 변수가 지정하는 서비스 공급자 아이덴티티가 포함되어 있습니다. ProtocolBinding 쿼리 매개 변수는 없습니다. 사용자가 이 하드 코드된 링크를 클릭하면 싱글 사인온 서비스로 리디렉션됩니다.

http://fedsrv.fedsite.com:82/affwebservices/public/saml2sso?
SPID=http%3A%2F%2Ffedsrv.acme.com%2Fsmidp2for90

예: ProtocolBinding 이 있는 원치 않는 응답

링크가 사용자를 싱글 사인온 서비스로 리디렉션합니다. 이 링크에는 SPID 쿼리 매개 변수가 지정하는 서비스 공급자 아이덴티티가 포함되어 있으며 아티팩트 바인딩이 사용됩니다. 사용자가 이 하드 코드된 링크를 클릭하면 로컬 싱글 사인온 서비스로 리디렉션됩니다.

http://idp-ca:82/affwebservices/public/saml2sso?SPID=
http%3A%2F%2Ffedsrv.acme.com%2Fsmidp2for90&
ProtocolBinding=urn:oasis:names:tc:SAML:2.0:bindings:HTTP-Artifact

IdP 에서 ForceAuthn 및 IsPassive 처리

서비스 공급자가 싱글 사인온을 시작하는 경우 해당 서비스 공급자는 ForceAuthn 또는 IsPassive 쿼리 매개 변수를 AuthnRequest 메시지에 포함할 수 있습니다.

참고: CA SiteMinder® Federation Standalone 아이덴티티 공급자는 IsPassive 쿼리 매개 변수를 지원하지 않습니다. 그러나 타사 서비스 공급자가 보내는 AuthnRequest 메시지에는 IsPassive 매개 변수가 포함될 수 있습니다.

서비스 공급자가 AuthnRequest 에 ForceAuthn 또는 IsPassive 를 포함하면 CA SiteMinder?Federation Standalone 아이덴티티 공급자는 이 쿼리 매개 변수를 다음과 같이 처리합니다.

ForceAuthn 처리

서비스 공급자가 AuthnRequest 메시지에 ForceAuthn=True 를 포함하면 CA SiteMinder?Federation Standalone 아이덴티티 공급자는 세션이 있는 경우라도 사용자에게 자격 증명을 요청합니다.

IsPassive 처리

서비스 공급자가 IsPassive 를 AuthnRequest 에 포함하는 경우 아이덴티티 공급자가 이를 준수할 수 없으면 다음 SAML 응답 중 하나가 서비스 공급자에게 다시 전송됩니다.

- AuthnRequest 메시지에 IsPassive=True 가 있고 CA SiteMinder® Federation Standalone 세션이 없는 경우에는 세션이 필요하기 때문에 CA SiteMinder® Federation Standalone 아이덴티티 공급자가 오류 메시지가 포함된 SAML 응답을 반환합니다.
- AuthnRequest 메시지에 IsPassive=True 가 있고 CA SiteMinder® Federation Standalone 세션이 있는 경우에는 CA SiteMinder® Federation Standalone 아이덴티티 공급자가 어설션을 반환합니다.
- IsPassive 및 ForceAuthn 이 AuthnRequest 메시지에 있고 둘 다 True 로 설정된 경우, 이는 잘못된 요청이기 때문에 CA SiteMinder® Federation Standalone 아이덴티티 공급자가 오류를 반환합니다. IsPassive 와 ForceAuthn 은 동시에 사용할 수 없습니다.

SP 에서 시작되는 SSO(SAML 2.0)

SP 에서 시작되는 SSO 의 경우에는 서비스 공급자의 AuthnRequest 서비스에 대한 하드 코딩된 링크가 포함된 HTML 페이지가 서비스 공급자에 있어야 합니다. 링크는 사용자를 인증될 아이덴티티 공급자로 리디렉션하고 AuthnRequest 자체에 무엇이 포함되어 있는지 확인합니다.

이 정보는 아티팩트 또는 POST 바인딩에 적용됩니다.

사용자가 선택하는 하드 코딩된 링크에는 AuthnRequest 서비스에 대한 HTTP GET 요청에서 사용되는 특정 쿼리 매개 변수가 포함되어야 합니다.

참고: 이러한 하드 코드된 링크가 포함된 페이지는 보호되지 않은 영역에 있어야 합니다.

트랜잭션에 대해 아티팩트 또는 프로필 바인딩이 사용되도록 지정하기 위한 링크 구문은 다음과 같습니다.

http://*sp_server:port*/affwebservices/public/saml2authnrequest?
ProviderID=*IdP_ID*&ProtocolBinding=*URI_of_binding*&
RelayState=*target_URL*

sp_server:port

CA SiteMinder® Federation Standalone 을 호스트하는 서비스 공급자의 서버 및 포트를 지정합니다.

IdP_ID

아이덴티티 공급자에게 할당된 아이덴티티를 지정합니다. 엔터티 ID 는 대/소문자를 구분합니다. "관리 UI"에 표시되는 그대로 입력하십시오.

URI_of_binding

ProtocolBinding 요소에 대한 POST 또는 아티팩트 바인딩의 URI 를 식별합니다. 이 URI 는 SAML 2.0 사양에 의해 정의됩니다.

- 아티팩트 바인딩에 대한 URI 는 다음과 같습니다.

  urn:oasis:names:tc:SAML:2.0:bindings:HTTP-Artifact

- POST 바인딩에 대한 URI 는 다음과 같습니다.

  urn:oasis:names:tc:SAML:2.0:bindings:HTTP-POST

  HTTP-POST 싱글 사인온에 대해서는 이 매개 변수를 설정할 필요가 없습니다.

또한 요청이 작동하게 하려면 파트너 관계에 대한 바인딩이 사용되도록 설정하십시오.

target_URL

서비스 공급자에서 페더레이션 대상의 URL 을 지정합니다.

다음 정보에 주의하십시오.

- AuthnRequest 링크에 ProtocolBinding 쿼리 매개 변수를 포함하지 않는 경우 기본 바인딩은 파트너 관계에 대해 정의된 바인딩입니다. 두 바인딩이 모두 파트너 관계에 정의되어 있는 경우에는 AuthnRequest 에서 바인딩이 전달되지 않습니다. 따라서 아이덴티티 공급자의 기본 바인딩이 사용됩니다.
- 아티팩트 및 POST 바인딩이 파트너 관계에 대해 사용되도록 설정되어 있지만 아티팩트 바인딩만 사용하려는 경우에는 링크에 ProtocolBinding 쿼리를 포함하십시오.

SP 에서 사용하는 AuthnRequest 쿼리 매개 변수

CA SiteMinder?Federation Standalone SP 가 AuthnRequest 서비스에 대한 링크에 사용할 수 있는 쿼리 매개 변수는 다음과 같습니다.

ProviderID(필수)

AuthnRequest 서비스가 AuthnRequest 메시지를 보내는 아이덴티티 공급자의 엔터티 ID 입니다.

ProtocolBinding

AuthnRequest 메시지의 ProtocolBinding 요소를 지정합니다. 이 요소는 아이덴티티 공급자의 SAML 응답을 반환하는 데 사용되는 프로토콜을 지정합니다. 지정된 아이덴티티 공급자가 지정된 프로토콜 바인딩을 지원하도록 구성되어 있지 않으면 요청이 실패합니다.

AuthnRequest 에서 이 매개 변수를 사용하는 경우에는 AssertionConsumerServiceIndex 매개 변수를 포함할 수 없습니다. 두 매개 변수는 동시에 사용할 수 없습니다.

ForceAuthn

아이덴티티 공급자에게 기존 보안 컨텍스트를 사용하는 대신 사용자를 직접 인증해야 한다고 지시합니다. 아이덴티티 공급자가 CA SiteMinder® Federation Standalone 을 사용하면 이 쿼리 매개 변수를 사용하고, 타사 페더레이션 소프트웨어를 사용하면 이 쿼리 매개 변수를 사용하지 마십시오.

- SP 가 AuthnRequest 메시지에 ForceAuthn=True 를 설정하고 특정 사용자에 대한 세션이 존재하는 경우에는 아이덴티티 공급자가 사용자에게 인증을 요청합니다. 사용자가 성공적으로 인증되면 IdP 가 기존 세션의 아이덴티티 정보를 어설션에 넣어 보내고 인증을 위해 생성된 세션을 삭제합니다.
- SP 가 AuthnRequest 메시지에 ForceAuthn=True 를 설정한 경우 세션이 없으면 IdP 가 사용자에게 인증을 요청합니다. 사용자가 성공적으로 인증되면 세션이 설정됩니다.

예

http://sp1.demo.com:81/affwebservices/public/saml2authnrequest?ProviderID=idp1.example.com&ForceAuthn=yes

IsPassive

아이덴티티 공급자가 사용자에게 자격 증명을 요구하지 않고, 또는 어떤 방식으로든 사용자와 상호 작용하지 않고 사용자를 로그인하도록 지시합니다. SiteMinder 아이덴티티 공급자는 사용자에게 세션이 없으면 이 쿼리 매개 변수를 인정하지 않습니다. 사용자에게 세션이 없으면 아이덴티티 공급자는 오류를 반환합니다.

AssertionConsumerServiceIndex

어설션 소비자 서비스로 작동하는 끝점의 인덱스를 지정합니다. 인덱스는 아이덴티티 공급자에게 어설션 응답을 보낼 위치를 알려 줍니다.

AuthnRequest 에서 이 매개 변수를 사용하는 경우에는 ProtocolBinding 매개 변수를 포함하지 마십시오. 이 둘은 함께 사용할 수 없습니다. 어설션 소비자 서비스에는 자체 프로토콜 바인딩이 있으며 이는 ProtocolBinding 매개 변수와 충돌할 수 있습니다.

RelayState

서비스 공급자에서 대상 리소스의 URL 을 나타냅니다. 이 쿼리 매개 변수를 포함하면 서비스 공급자에게 사용자를 보낼 대상을 알려 줍니다. 그렇지 않으면 파트너 관계에 대해 정의된 기본 대상이 사용됩니다.

ProtocolBinding 쿼리 매개 변수의 필수 사용

아티팩트 및 POST 바인딩이 파트너 관계에 대해 사용되도록 설정되어 있고 사용자가 아티팩트 바인딩만 사용하고자 하는 경우에는 ProtocolBinding 매개 변수가 필요합니다.

- SAML 2.0 사양으로 지정된 아티팩트 바인딩 URI 는 다음과 같습니다.

  urn:oasis:names:tc:SAML:2.0:bindings:HTTP-Artifact

- SAML 2.0 사양으로 지정된 POST 바인딩 URI 는 다음과 같습니다.

  urn:oasis:names:tc:SAML:2.0:bindings:HTTP-POST

  HTTP-POST 싱글 사인온에 대해서는 이 매개 변수를 설정할 필요가 없습니다.

ProtocolBinding 의 선택적 사용

ProtocolBinding 쿼리 매개 변수를 사용하지 *않는* 경우 다음 정보가 적용됩니다.

- 파트너 관계에 대해 하나의 바인딩만 사용되도록 설정되어 있고 ProtocolBinding 쿼리 매개 변수가 지정되지 않은 경우에는 파트너 관계에 사용되도록 설정된 바인딩이 사용됩니다.
- 두 바인딩이 모두 사용되도록 설정된 경우 ProtocolBinding 쿼리 매개 변수를 지정하지 않으면 POST 바인딩이 기본값으로 사용됩니다.

참고: 쿼리 매개 변수를 HTTP-인코딩할 필요는 없습니다.

예: ProtocolBinding 쿼리 매개 변수가 없는 AuthnRequest 링크

이 샘플 링크는 AuthnRequest 서비스로 이동합니다. 이 링크는 ProviderID 쿼리 매개 변수에 있는 아이덴티티 공급자를 지정합니다.

http://ca.sp.com:90/affwebservices/public/saml2authnrequest?
ProviderID=http%3A%2F%2Ffedsrv.acme.com%2Fsmidp2for90

사용자가 서비스 공급자에서 링크를 클릭하면 CA SiteMinder?Federation Standalone 이 AuthnRequest 메시지에 대한 요청을 전달합니다.

예: ProtocolBinding 쿼리 매개 변수가 있는 AuthnRequest 링크

http://ca.sp.com:90/affwebservices/public/saml2authnrequest?
ProviderID=http%3A%2F%2Ffedsrv.acme.com%2Fsmidp2for90&
ProtocolBinding=urn:oasis:names:tc:SAML:2.0:bindings:HTTP-Artifact

사용자가 서비스 공급자에서 링크를 클릭하면 CA SiteMinder?Federation Standalone 이 AuthnRequest 메시지에 대한 요청을 전달합니다.

IP 에서 시작되는 싱글 사인온(WSFED)

사용자가 RP(리소스 파트너)로 이동하기 전에 IP(아이덴티티 공급자)를 방문할 수 있습니다. 사용자가 먼저 아이덴티티 공급자로 이동하면 링크가 HTTP Get 요청을 생성해야 합니다. 하드 코딩된 링크는 IP 의 피동 요청자 서비스를 가리킵니다. 요청에는 RP 공급자 ID 와 선택적으로 기타 매개 변수가 포함됩니다.

링크 구문은 다음과 같습니다.

https://i*p_server:port*/affwebservices/public/wsfedsso?wa=wsignin1.0&wtrealm=*rp_id*

ip_server:port

아이덴티티 파트너에 있는 시스템의 서버 및 포트 번호를 지정합니다. 시스템은 페더레이션 네트워크에 설치된 구성 요소에 따라 웹 에이전트 옵션 팩이나 SPS 페더레이션 게이트웨이를 호스트하고 있습니다.

rp_id

RP 의 ID 입니다. 엔터티 ID 는 대/소문자를 구분합니다. "관리 UI"에 표시되는 그대로 입력하십시오.

RP 에서 시작되는 싱글 사인온(WSFED)

사용자가 RP 에서 싱글 사인온을 시작하는 경우 일반적으로 사용자는 목록에서 IP 를 선택합니다. 사이트 선택 페이지는 보호되지 않은 영역에 있습니다.

사이트 선택 페이지의 링크는 IP 의 피동 요청자 서비스를 가리킵니다. 링크가 선택된 후 RP 는 어설션을 가져오기 위해 사용자를 IP 로 리디렉션합니다.

제 16 장: 사용자 세션에서 로그아웃

싱글 로그아웃(SAML 2.0)

SLO(싱글 로그아웃)가 수행되면 로그아웃을 시작한 브라우저에 대한 모든 사용자의 세션이 동시에 종료됩니다. 모든 세션을 닫으면 인증되지 않은 사용자가 SP 의 리소스에 액세스하지 못하게 됩니다.

싱글 로그아웃 바인딩에 따라 싱글 로그아웃 메시지와 함께 전송되는 내용 및 수신되는 각 메시지를 처리하는 방법이 결정됩니다.

싱글 로그아웃 작업에는 다음 두 개의 바인딩을 사용할 수 있습니다.

HTTP-리디렉션

HTTP-리디렉션 바인딩은 브라우저를 사용하여 각 로그아웃 트랜잭션을 수행합니다. 싱글 로그아웃 메시지는 항상 GET 요청입니다. 모든 요청 및 응답에 브라우저가 관여합니다. 브라우저가 관여한다는 것은 HTTP-리디렉션 바인딩은 SOAP 바인딩과 달리 브라우저 세션 데이터를 제공한다는 것을 의미합니다.

HTTP-리디렉션 바인딩의 단점은 메시지의 데이터가 쿼리 문자열에서 전송할 수 있는 데이터로 제한된다는 점입니다. 또한 HTTP-리디렉션 바인딩은 익명 프로세스이므로 시간 만료가 발생할 가능성이 별로 없습니다. 하지만 리디렉션에 실패하면 이로 인해 전체 싱글 로그아웃 체인이 중지됩니다.

SOAP

SOAP 바인딩은 POST 요청을 사용하여 싱글 로그아웃 트랜잭션을 수행합니다. POST 요청을 통해 HTTP-리디렉션 바인딩보다 많은 데이터를 전송할 수 있습니다. 또한 SOAP 를 통해 더 다양한 암호화 방법 및 다른 기능을 사용할 수 있습니다.

SOAP 는 동기식 프로세스입니다. IdP 는 더 많은 제어권을 가지며 하나의 SP 에서 발생하는 문제로 프로세스 전체가 방해되는 상황을 방지할 수 있습니다. SOAP 통신은 백 채널을 통해 수행됩니다. 한 번의 로그아웃 실패로 IdP 가 나머지 SP 에서 로그아웃 시도를 중지할 필요는 없습니다.

SOAP 는 백 채널 연결을 사용하므로 초기 싱글 로그아웃 호출 및 응답 이후에는 브라우저가 개입되지 않습니다. SOAP 바인딩은 로그아웃 프로세스의 일부로 원격 엔터티에서 쿠키를 정리하지 않습니다. 쿠키는 로컬 엔터티에서만 정리됩니다. 쿠키를 삭제해야 하는 경우 HTTP-리디렉션 바인딩을 사용하십시오.

HTTP-리디렉션 및 SOAP 를 사용하여 네트워크에서 싱글 로그아웃 관리

네트워크에는 HTTP-리디렉션 바인딩을 지원하는 사이트와 SOAP 바인딩을 지원하는 사이트가 있을 수 있습니다. IdP 는 여러 바인딩을 관리해야 하지만 SP 는 하나의 로그아웃 요청만 보내거나 받습니다.

다음 단원에서는 혼합 바인딩 환경을 처리하기 위한 구성 지침을 제공합니다.

CA SiteMinder® Federation Standalone 이 IdP 에 있을 때의 SLO 구성

CA SiteMinder® Federation Standalone 이 IdP 에 있을 때는 파트너 관계에 HTTP 리디렉션 기반 SLO 서비스 URL 과 SOAP 기반 SLO 서비스 URL 이 포함되도록 구성하십시오.

IdP 에서 CA SiteMinder® Federation Standalone 은 세션의 각 SP 에 대한 구성을 검사하고 SOAP 를 사용하는 모든 로그아웃을 먼저 처리합니다. SOAP 를 지원하지 않는 SP 에 대한 HTTP-리디렉션 로그아웃이 그 다음에 처리됩니다.

CA SiteMinder® Federation Standalone 이 SP 에 있을 때의 SLO 구성

CA SiteMinder® Federation Standalone 이 SP 에 있고 SP 가 싱글 로그아웃을 시작하는 경우에는 HTTP-리디렉션 바인딩으로 로그아웃을 시작하는 것이 좋습니다. 사용자 세션에 대한 다른 SP 는 SOAP 를 지원하지 않을 수 있습니다.

HTTP-리디렉션은 브라우저 세션을 사용하여 모든 리디렉션을 처리합니다. 이러한 이유로 HTTP-리디렉션은 HTTP 리디렉션만 지원하는 SP 의 로그아웃을 위해 IdP 에 있어야 하는 필수 데이터를 전송합니다. 시작하는 SP 가 HTTP-리디렉션으로 프로세스를 시작하는 경우 IdP 는 이를 지원하는 모든 SP 에 SOAP 를 사용할 수 있습니다. 나머지 SP 에 대해서는 HTTP-리디렉션 바인딩으로 전환하십시오.

SOAP 바인딩을 사용하여 싱글 로그아웃을 시작하는 경우에는 브라우저 세션 데이터가 존재하지 않습니다.

SP 에서 시작되는 로그아웃이 HTTP-리디렉션이 사용되도록 하려면 SP 의 로컬 서블릿을 가리키는 HTTP-리디렉션 링크를 페이지나 응용 프로그램에 포함하십시오. CA SiteMinder® Federation Standalone 의 경우 이 링크는 다음과 같습니다.

http://*sp_host:port*/affwebservices/public/saml2slo.

이 포함된 링크로 인해 CA SiteMinder® Federation Standalone 이 IdP 의 SLO 서비스로 보내는 SAML <LogoutRequest> 메시지를 생성합니다. 사용자가 로그아웃하면 먼저 SP 에서 로그아웃이 수행된 다음 로그아웃 요청이 IdP 로 전송됩니다. 그러면 IdP 는 사용자 세션에 관여한 다른 모든 SP 에 대해 로그아웃 프로세스를 완료합니다.

SLO 요청 유효 기간에 대한 차이 시간 이해

로그아웃 요청의 유효 기간을 계산할 때는 두 개의 값이 관련됩니다. 이 값은 IssueInstant 값과 NotOnOrAfter 값입니다. SLO 응답에서 싱글 로그아웃 요청은 NotOnOrAfter 값에 도달할 때까지 유효합니다. 싱글 로그아웃 요청이 생성될 때 CA SiteMinder?Federation Standalone 의 시스템 시간이 사용됩니다. 그 결과로 얻은 시간은 요청 메시지에 설정되는 IssueInstant 가 됩니다. 로그아웃 요청이 만료되는 시점을 확인하기 위해 CA SiteMinder?Federation Standalone 은 현재 시스템 시간을 가져와서 여기에 "차이 시간"과 "SLO 유효 기간"을 더합니다. 그 결과로 얻은 시간은 NotOnOrAfter 값이 됩니다.

참고: 시간은 GMT 를 기준으로 합니다.

예를 들어 어설션 당사자 측에서 로그아웃 요청이 1:00 GMT 에 생성된다고 가정합니다. 차이 시간은 30 초이고 SLO 유효 기간은 60 초입니다. 따라서 요청은 1:00 GMT 에서 1:01:30 GMT 까지 유효합니다. IssueInstant 값은 1:00 GMT 이고 싱글 로그아웃 요청 메시지는 90 초 뒤에 더 이상 유효하지 않습니다.

싱글 로그아웃 구성

싱글 로그아웃을 구성할 때 다음 사항에 유의하십시오.

- 파트너가 HTTP-리디렉션을 사용하여 SAML <LogoutRequest> 메시지를 수신하는 경우 보내는 당사자에 대한 응답은 HTTP-리디렉션을 사용해야 합니다.
- 파트너가 SOAP 를 사용하여 SAML <LogoutRequest> 메시지를 수신하는 경우 보내는 당사자에 대한 응답은 SOAP 를 통해 전송되어야 합니다.
- 파트너가 지원하지 않는 바인딩을 통해 SLO 를 수신하는 경우에는 싱글 로그아웃이 실패합니다.
- 싱글 로그아웃 사용자 세션에 HTTP-리디렉션 및 SOAP 를 사용하는 파트너가 포함된 경우에는 두 바인딩을 모두 지원하도록 CA SiteMinder® Federation Standalone 을 구성하십시오. IdP 는 로그아웃을 진행할 때 SOAP 를 사용하여 모든 SP 를 로그아웃한 다음 HTTP-리디렉션 바인딩을 사용하여 모든 SP 를 로그아웃합니다.
- CA SiteMinder® Federation Standalone SP 가 싱글 로그아웃을 시작하는 경우에는 SP 가 SOAP 를 지원하더라도 HTTP-리디렉션 바인딩부터 시작하는 것이 좋습니다.

  SOAP 및 HTTP-리디렉션을 지원하는 환경에서 싱글 로그아웃을 관리하기 위한 구성 지침 (페이지 296)을 참조하십시오.

파트너 관계의 양쪽에서 싱글 로그아웃을 구성하려면

참고: SLO 구성 설정은 IdP 와 SP 에서 동일합니다.

1. 파트너 관계 마법사의 SSO 및 SLO 단계부터 시작합니다.
2. SLO 섹션에서 SLO 바인딩을 하나 또는 둘 다 선택합니다.

   SLO 바인딩은 싱글 로그아웃을 가능하게 하며 로컬 엔터티에서 사용 중인 바인딩을 나타냅니다. 또한 SLO 바인딩은 로컬 엔터티가 싱글 로그아웃 요청을 수신할 때 수락하는 바인딩도 나타냅니다.

   SOAP 를 선택하면 SOAP 메시지에서 이름 ID 를 암호화할 수 있습니다. 이 옵션의 설정은 파트너 관계 마법사의 "서명 및 암호화" 단계에 나옵니다.

SOAP 를 바인딩으로 선택하면 백 채널에 대한 수신 및 송신 구성이 활성화됩니다. SLO 요청 및 응답이 백 채널을 통해 전송됩니다. 각 로컬 파트너는 원격 파트너에게 인증을 요청하여 백 채널에 보안을 적용할 수 있습니다.

SLO 에 대한 백 채널 설정에 대한 더 많은 내용을 확인할 수 있습니다.

3. 추가 SLO 설정을 구성합니다.
   - SLO 확인 URL
   - 유효 기간
   - 릴레이 상태가 SLO 확인 URL 무시

   참고: "도움말"을 클릭하면 필드, 컨트롤 및 해당되는 각 요구 사항에 대한 설명을 볼 수 있습니다.
4. SLO 서비스 URL 에 대한 테이블을 완성합니다. 최소한 하나의 항목이 있어야 합니다.

   SLO 서비스 URL 은 싱글 로그아웃을 시작한 다음 SAML <LogoutRequest> 메시지를 생성하도록 CA SiteMinder® Federation Standalone 을 트리거합니다. 또한 SLO 서비스 URL 은 로그아웃 요청 메시지를 전송할 대상을 CA SiteMinder® Federation Standalone 에 알립니다.

   지원되는 각 SLO 바인딩에 대해 다음과 같이 SLO 서비스 URL 을 지정합니다.
   - HTTP-Redirect 사용 - HTTP-Redirect 를 바인딩으로 사용하는 하나의 URL 을 선택합니다.
   - SOAP 사용 - SOAP 를 바인딩으로 사용하는 하나의 URL 을 선택합니다.
   - 리디렉션 및 SOAP 사용 - 두 개의 URL(HTTP-리디렉션으로 설정된 URL 한 개와 SOAP 로 설정된 URL 한 개)을 선택합니다.

   참고: "응답 위치 URL" 필드는 선택 사항입니다.

   테이블에 항목을 추가하려면 "행 추가"를 클릭하십시오. 선택된 원격 엔터티에 대해 정의된 값이 이미 테이블에 입력되어 있습니다.

이러한 단계를 완료하면 싱글 로그아웃이 구성됩니다.

싱글 로그아웃에 대한 백 채널 구성

SOAP 바인딩을 사용한 싱글 로그아웃은 로그아웃 요청 및 응답을 백 채널을 통해 전송합니다. 엔터티가 백 채널에 액세스하려면 인증이 필요하도록 설정할 수 있습니다. SSL 은 필수가 아니지만 SSL 을 사용하여 백 채널에 보안을 적용할 수도 있습니다.

SSL 을 사용하여 백 채널에 보안을 적용하는 절차는 다음과 같습니다.

- SSL 이 사용되도록 설정합니다.

  기본 인증에는 SSL 이 필요하지 않지만 SSL 을 통한 기본 인증을 사용할 수 있습니다. 클라이언트 인증서 인증에는 SSL 이 필요합니다.

- 싱글 로그아웃 교환에 대해 들어오는 백 채널과 나가는 백 채널을 구성합니다. 로컬 엔터티는 나가는 채널을 통해 메시지를 보내고 들어오는 채널을 통해 메시지를 받을 수 있어야 합니다.

  참고: 들어오고 나가는 백 채널을 구성할 수 있지만 한 채널은 하나의 구성만 가질 수 있습니다. 동일한 채널을 사용하는 두 서비스는 동일한 백 채널 구성을 사용합니다. 예를 들어 로컬 어설션 당사자의 수신 채널이 HTTP-아티팩트 SSO 와 SOAP 기반 SLO 를 지원할 경우 이 두 서비스는 동일한 백 채널 구성을 사용해야 합니다.

- 원격 엔터티가 보호된 백 채널을 통해 액세스를 얻는 데 필요한 인증 유형을 선택합니다. 인증 방법은 채널별로(나가는 채널 또는 들어오는 채널) 적용됩니다.

  백 채널 인증에 대한 옵션은 다음과 같습니다.

  기본

  기본 인증 체계로 백 채널을 보호합니다.

  참고: 백 채널 연결에 대해 SSL 이 사용되도록 설정하는 경우에도 기본 인증을 선택할 수 있습니다.

  클라이언트 인증서

  X.509 클라이언트 인증서를 사용한 SSL 이 어설션 당사자 백 채널을 보호합니다.

  "클라이언트 인증서"를 인증 방법으로 선택하는 경우 모든 끝점 URL 이 SSL 통신을 사용해야 합니다. 즉, URL 이 **https://**로 시작해야 합니다. 끝점 URL 은 서버에서 싱글 사인온, 싱글 로그아웃, 어설션 소비자 서비스, 아티팩트 레졸루션 서비스(SAML 2.0), 어설션 검색 서비스(SAML 1.x) 등의 다양한 SAML 서비스를 찾습니다.

인증 없음

신뢰 당사자가 자격 증명을 제공할 필요가 없습니다. 백 채널에 보안이 적용되지 않습니다. 이 옵션을 사용할 때도 SSL 을 활성화할 수 있습니다. 백 채널 트래픽은 암호화되지만 당사자 간에 자격 증명이 교환되지 않습니다.

"인증 없음" 옵션은 테스트 용도로만 사용하고 CA SiteMinder® Federation Standalone 이 SSL 사용 장애 조치를 사용하도록 구성되고 프록시 서버 뒤에 있는 경우를 제외하고는 프로덕션 용도로 사용하지 마십시오. 이 경우, 백 채널을 보호하기 위해 클라이언트 인증서 인증이 사용되는 경우에는 서버 인증서를 갖고 있는 프록시 서버가 인증을 처리합니다. 이 경우 모든 IdP->SP 파트너 관계에서 인증 유형으로 "인증 없음"을 사용할 수 있습니다.

중요! 들어오는 백 채널에 대해 선택한 인증 방법은 파트너 관계에서 다른 측의 나가는 백 채널에 대한 인증 방법과 일치해야 합니다. 인증 방법의 선택에 대한 동의는 대역 외에서 처리됩니다.

싱글 로그아웃에 대한 백 채널에 보안을 적용하려면

1. 파트너 관계 마법사의 SSO 및 SLO 단계에 있는 "백 채널" 그룹 상자에서 시작합니다.
2. "SLO" 그룹 상자에서 "SOAP"를 선택합니다. "인증 방법" 필드가 활성화됩니다.
3. 들어오는 백 채널 및 나가는 백 채널에 대한 인증 방법의 유형을 선택합니다. "기본" 및 "클라이언트 인증서" 방법에 대해 구성할 추가 필드가 표시됩니다.

   참고: "도움말"을 클릭하면 필드, 컨트롤 및 해당되는 각 요구 사항에 대한 설명을 볼 수 있습니다.

   "인증 없음"을 인증 방법으로 선택하는 경우에는 추가적인 단계가 필요 없습니다.
4. 선택하는 인증 방법에 따라 구성해야 할 몇 개의 추가적인 필드가 표시됩니다.

   참고: "도움말"을 클릭하면 필드, 컨트롤 및 해당되는 각 요구 사항에 대한 설명을 볼 수 있습니다.

모든 필수 필드에 값을 입력하면 백 채널 구성이 완료됩니다.

추가 정보:

Apache 웹 서버 및 UI 에 대한 SSL 관리 (페이지 413)

사인아웃 개요(WS-페더레이션)

사인아웃은 사인아웃을 시작한 브라우저에 대해 모든 사용자 세션을 동시에 종료하는 것입니다. 모든 사용자 세션을 닫으면 권한 없는 사용자가 리소스 파트너의 리소스에 액세스하지 못하게 됩니다.

사인아웃이 반드시 사용자의 모든 세션을 종료하는 것은 아닙니다. 예를 들어 브라우저를 두 개 열어 둔 사용자에게는 두 개의 독립적 세션이 있을 수 있습니다. 이 경우 사인아웃을 시작하는 브라우저에 대한 세션만 해당 세션에 대한 모든 페더레이션된 사이트에서 종료됩니다. 다른 브라우저의 세션은 여전히 활성 상태입니다.

정책 서버는 signoutconfirmurl.jsp 를 사용하여 사인아웃을 수행합니다. 이 페이지는 아이덴티티 공급자 시스템에 있습니다. 아이덴티티 공급자 파트너는 사용자 대신 사인아웃 요청을 시작합니다. JSP 는 사용자가 특정 브라우저 세션 중에 사인온한 각 사이트에 사인아웃 요청을 보냅니다. 그러면 사용자가 사인아웃됩니다.

사용자는 아이덴티티 공급자에서만 사인아웃 요청을 시작할 수 있습니다. 적절한 서블릿을 가리키는 링크를 클릭하면 요청이 트리거됩니다. 아이덴티티 공급자 사이트에서 사인아웃 확인 페이지는 보호되지 않는 리소스여야 합니다.

참고: 정책 서버는 사인아웃에 대해 WS-페더레이션 피동 요청 프로필만 지원합니다.

WSFED 사인아웃이 사용되도록 설정

사인아웃을 구성하려면 다음 요구 사항을 충족해야 합니다.

- 아이덴티티 공급자에서 사인아웃이 사용되도록 설정하려면 정책 서버 관리 콘솔을 사용하여 세션 저장소가 사용되도록 설정합니다.

  세션 저장소에 대한 자세한 내용은 *정책 서버 관리 안내서*를 참조하십시오.

- 싱글 사인온은 유효한 SiteMinder 영구 세션이 필요합니다. 이 세션은 싱글 사인온 중에 구성됩니다. 리소스 파트너에서 인증 URL 을 포함하여 보호된 리소스가 포함된 영역에 대해 영구 세션을 구성하십시오.

  영역에 대한 자세한 내용은 *정책 서버 구성 안내서*를 참조하십시오.

다음 단계를 수행하십시오.

1. Administrative UI 에 로그인합니다.
2. 수정할 WS-Federation 파트너 관계를 선택합니다.
3. 파트너 관계 마법사의 "싱글 사인온 및 사인아웃" 단계로 이동합니다.
4. "사인아웃" 섹션에서 다음 필드를 설정합니다.
   - 사인아웃 사용
   - 사인아웃 확인 URL(IP 만 해당)
   - Sign-Out URL

   각 URL 의 입력은 https:// 또는 http://로 시작해야 합니다.
5. "확인" 단계로 이동하고 "마침"을 클릭하여 변경 내용을 저장합니다.

사인아웃이 구성되었습니다.

SP 에서 로컬 로그아웃(SAML 2.0)

SP 역할을 하는 SiteMinder 는 독립 실행형 응용 프로그램에 대한 로컬 로그아웃을 지원합니다. 로컬 로그아웃의 경우 사용자가 로컬 SP 측 응용 프로그램에서 로그아웃할 수 있습니다. SP 의 세션이 제거되지만 IdP 또는 다른 SP 와의 통신이 연관되지 않습니다. IdP 및 다른 SP 의 세션은 활성 상태로 유지됩니다.

SP 의 응용 프로그램에 로그아웃 링크를 포함하면 SP 는 로컬 싱글 로그아웃 서비스에 로그아웃 요청을 전송합니다. SP 는 요청을 수신하면 사용자를 로그아웃시킵니다. SP 의 응용 프로그램은 로그아웃에 성공했다는 확인 메시지를 보내는 역할을 담당합니다.

SiteMinder 는 **localLogout** 이라는 쿼리 매개 변수를 사용하여 로컬 로그아웃을 제공합니다. 이 매개 변수를 사용하기 위해 응용 프로그램에 다음과 같은 내용의 페이지가 있을 수 있습니다.

```
demoapp 을 사용하여 등록을 완료했습니다.
세션을 안전하게 종료하려면 LOGOUT(로그아웃)을 선택하십시오.
```

다음 샘플 문자열은 LOGOUT(로그아웃) 버튼에 대한 링크를 나타냅니다.

```
<http://sp1server.demo.com:8080/affwebservices/public/saml2slo?Loc
alLogout=true
```

제 17 장: 인증 컨텍스트 처리(SAML 2.0)

*인증 컨텍스트*는 아이덴티티 공급자에서 사용자가 인증되는 방법을 나타냅니다. 아이덴티티 공급자는 서비스 공급자의 요청에 따라, 또는 아이덴티티 공급자의 구성에 따라 싱글 사인온 어설션에 인증 컨텍스트를 포함합니다. 서비스 공급자는 리소스에 대한 액세스를 허용하기 전에 어설션에 신뢰 수준을 설정하기 위해 인증 프로세스에 대한 정보를 요청할 수 있습니다.

인증 컨텍스트 요청

인증 컨텍스트를 요청하려면 SiteMinder 서비스 공급자가 아이덴티티 공급자에 대한 인증 요청에 <RequestedAuthnContext> 요소를 포함시켜야 합니다. 서비스 공급자는 SP->IdP 파트너 관계의 구성 설정에 따라 요청에 이 요청을 포함시킵니다.

인증 컨텍스트 가져오기

SiteMinder 아이덴티티 공급자는 다음 두 가지 방법 중 *하나*를 사용하여 인증 컨텍스트를 가져옵니다.

- IdP->SP 파트너 관계 구성에서 정적 AuthnContext URI 를 지정합니다.

  페더레이션된 파트너가 AuthnContext 요청을 지원하지 않는 SiteMinder 서비스 공급자인 경우 Administrative UI 에 URI 를 직접 입력하십시오.

- AuthnContext URI 는 구성된 인증 컨텍스트 템플릿을 사용하여 동적으로 결정됩니다.

  정책 서버는 인증 컨텍스트 URI 를 정책 서버에서 정의된 인증 수준에 매핑합니다. 이 인증 수준은 연결된 사용자 세션에 대한 인증 컨텍스트의 강도를 나타냅니다. 이 수준에서는 아이덴티티 공급자의 사용자 세션에서 인증 컨텍스트를 추출할 수 있습니다.

  아이덴티티 공급자는 요청을 수신하면 <RequestedAuthnContext> 요소의 값을 인증 컨텍스트와 비교합니다. 비교는 서비스 공급자의 요청에 있는 비교 값을 기반으로 합니다. 비교가 성공적인 경우 아이덴티티 공급자는 서비스 공급자에 반환하는 어설션에 인증 컨텍스트를 포함합니다. 서비스 공급자에서 유효성 검사가 구성된 경우 서비스 공급자는 들어오는 인증 컨텍스트를 요청한 값과 비교하여 유효성을 검사합니다.

IdP 에서 시작되는 SSO 에 대한 인증 컨텍스트 처리

싱글 사인온이 IdP 에서 시작되는 경우 인증 컨텍스트 처리의 단계는 다음과 같습니다.

1. 사용자 요청이 IdP 에서 싱글 사인온을 트리거합니다.
2. 사용자가 인증되고 사용자 세션이 생성됩니다. 인증 체계를 사용하여 구성된 보호 수준이 세션과 연결됩니다.
3. IdP 의 인증 컨텍스트 구성에 따라 다음과 같은 상황 중 *하나*가 발생할 수 있습니다.
   - 자동 검색이 수행됩니다. IdP-SP 파트너 관계에 대해 SiteMinder 커넥터를 사용하도록 구성한 경우에만 해당됩니다.

     구성된 인증 컨텍스트 템플릿에 기반하여 AuthnContext 클래스가 세션의 보호 수준에 매핑됩니다.
   - 미리 정의된 인증 클래스가 사용됩니다.

     지정한 하드 코딩된 URI 가 어설션에 추가됩니다.
4. IdP 가 어설션을 생성하고 인증 컨텍스트를 여기에 추가합니다. 그런 다음 어설션이 SP 에 전송됩니다.
5. SP 에서 어설션의 인증 컨텍스트 클래스와 SP 의 구성된 인증 클래스 간에 또 다른 비교가 이루어집니다. 이 비교가 성공하면 인증 트랜잭션이 완료됩니다.

SP 에서 시작되는 SSO 에 대한 인증 컨텍스트 처리

싱글 사인온이 SP 에서 시작되는 경우 인증 컨텍스트 처리의 단계는 다음과 같습니다.

1. SP 가 <RequestedAuthnContext> 요소 및 비교 연산자와 함께 인증 요청을 보냅니다. SP-> IdP 파트너 관계 구성의 설정에 따라 요소가 포함됩니다.
2. IdP 가 요청을 받으면 사용자를 인증하고 사용자 세션이 생성됩니다. 인증 체계의 보호 수준이 세션과 연결됩니다.

3. IdP 의 인증 컨텍스트 구성에 따라 다음과 같은 상황 중 *하나*가 발생할 수 있습니다.
   - 자동 검색이 수행됩니다.

     구성된 인증 컨텍스트 템플릿에 기반하여 AuthnContext 클래스가 세션의 보호 수준에 매핑됩니다.
   - 미리 정의된 인증 클래스가 사용됩니다.

     지정한 하드 코딩된 URI 가 어설션에 추가됩니다.
4. IdP 가 AuthnContext 를 사용자 세션의 인증 클래스와 비교합니다. 이 비교는 요청과 함께 전송된 비교 연산자를 기반으로 합니다. 각 비교 연산자가 처리에 미치는 영향의 예를 보려면 이 절차 뒤에 나오는 표를 참조하십시오.

   SP 가 요청에 인증 컨텍스트 URI 를 여러 개 포함하는 경우 각각의 클래스가 일대일로 순차적으로 세션의 컨텍스트와 비교됩니다. 첫 번째 비교가 성공할 경우 IdP 가 세션 인증 컨텍스트를 어설션에 추가합니다.
5. 비교가 성공하면 SP 에 전송되는 어설션에 인증 컨텍스트가 추가됩니다.

   비교가 실패하면 트랜잭션이 종료되고 "noauthncontext" 상태 응답이 반환됩니다.
6. SP 에서 어설션의 인증 컨텍스트와 SP 의 구성된 인증 클래스 간에 두 번째 비교가 이루어집니다. 이 비교가 성공하면 인증 트랜잭션이 완료됩니다.

다음 표에서는 인증 컨텍스트 요청에서 전송된 비교 특성을 기반으로 인증 컨텍스트가 처리되는 방식의 예를 보여 줍니다.

SP 에서 요청된 인증 컨텍스트	비교 특성 값	IdP 에서 구성된 인증 컨텍스트	상태 응답
Password	exact	InternetProtocol	NoAuthnContext
Password	minimum	InternetProtocol	NoAuthnContext
Password	maximum	InternetProtocol	NoAuthnContext
InternetProtocol	exact	InternetProtocol	Success
InternetProtocol	minimum	InternetProtocol	Success

SP 에서 요청된 인증 컨텍스트	비교 특성 값	IdP 에서 구성된 인증 컨텍스트	상태 응답
InternetProtocol	maximum	InternetProtocol	Success
InternetProtocol	maximum	Password	NoAuthnContext
InternetProtocol	maximum	Password	Success

인증 컨텍스트 템플릿 구성

인증 컨텍스트 템플릿은 파트너가 지원하는 특정 SAML 2.0 AuthnContext URI 를 정의합니다. URI 각각은 특정 컨텍스트 클래스를 나타냅니다. 파트너 관계 단위로 템플릿을 선택할 수 있으며 여러 파트너 관계가 하나의 템플릿을 사용할 수 있습니다.

템플릿은 일반적인 기능 이외에 각 파트너에서 다음과 같은 고유한 기능을 수행합니다.

IdP 에서

IdP 에는 다음과 같은 경우에만 템플릿이 필요합니다.

- SiteMinder 커넥터가 사용하도록 설정된 경우
- IdP 가 SP 요청에서 인증 컨텍스트를 자동으로 검색하는 경우

템플릿은 URI 를 사용자 세션과 관련된 보호 수준에 매핑합니다. 보호 수준은 1 부터 1000 까지 정책 서버에서 인증 체계의 강도를 나타내며 1000 이 가장 강력한 강도입니다. 관리자는 사용자를 인증하고 사용자 세션을 설정하는 인증 체계를 구성할 때 보호 수준을 할당합니다.

참고: 보호 수준은 SiteMinder 커넥터에서만 사용할 수 있습니다.

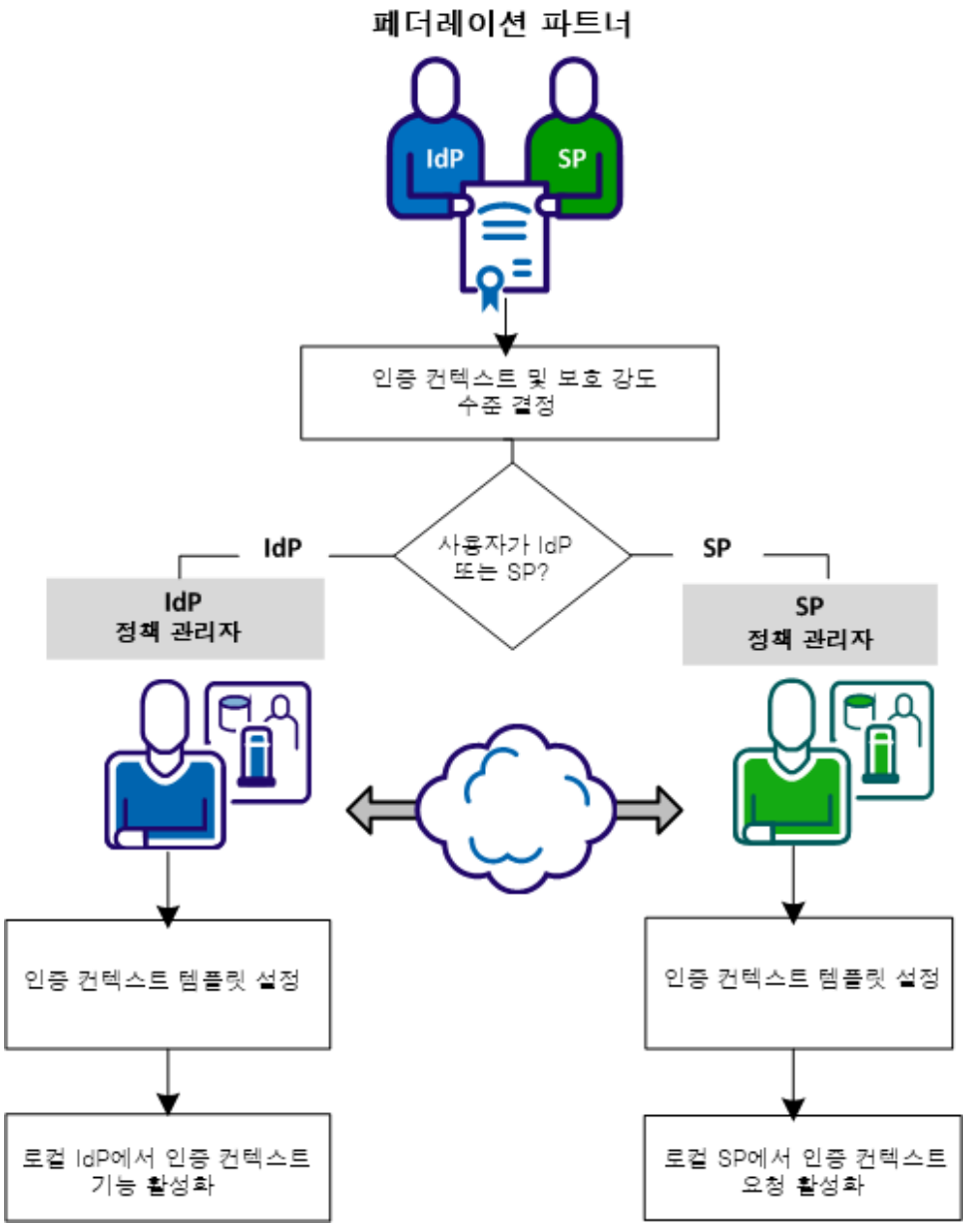
SP 에서

인증 요청을 통해 전달되는 인증 컨텍스트를 생성하려면 SP 의 인증 컨텍스트 템플릿이 필요합니다. SP 는 요청을 생성한 후 IdP 에 전송합니다. 수신한 어설션이 요청된 인증 컨텍스트를 충족하는지 SP 에서 유효성을 검사하는 데도 템플릿이 필요합니다.

구성을 진행하기 전에 최소한 다음과 같은 지식이 있는지 확인하십시오.

- 인증 컨텍스트 처리와 관련한 SAML 2.0 표준
- 페더레이션 구성 개체
- 관리 UI 에 액세스하여 사용하는 방법

다음 그림에서는 각 파트너의 구성 프로세스를 보여 줍니다. SiteMinder 페더레이션은 각 사이트에 설치되어 있지 않아도 됩니다.



인증 컨텍스트 처리를 구성하려면 다음 단계를 완료하십시오.

1. 인증 컨텍스트와 강도 수준을 결정합니다. (페이지 311)
2. 인증 컨텍스트 템플릿을 설정합니다 (페이지 311).
3. 사이트에서 다음 태스크를 완료합니다.
   - 로컬 IdP 파트너 관계에서 인증 컨텍스트 기능을 활성화합니다 (페이지 314).
   - 로컬 SP 파트너 관계에서 인증 컨텍스트 요청이 사용되도록 설정합니다 (페이지 317).

파트너와 상의하여 인증 컨텍스트 및 강도 수준 결정

요청된 리소스에 대한 액세스를 허용하기 전에 SP 가 특정 인증 컨텍스트 및 강도 수준을 요구할 수 있습니다.. SP 에서 리소스의 민감도에 따라 SP 는 IdP 로부터 받는 어설션을 신뢰해야 합니다.

IdP 와 SP 의 관리자들은 지원되는 인증 컨텍스트와 각 인증 컨텍스트 URI 의 상대적인 강도에 대한 지침을 마련해야 합니다. IdP 에서 URI 의 순서 및 관련된 상대적 강도 수준은 IdP 가 SP 에 응답하는 방식에 영향을 줍니다.

예를 들어, SP 는 X.509 인증서에 대한 인증 컨텍스트 및 정확한 비교 값을 요구합니다. IdP 는 적절한 강도 수준에서 요청하는 사용자를 인증하고 인증 컨텍스트의 평가 중에 비교 값을 충족시켜야 합니다.

인증 컨텍스트 템플릿 설정

인증 컨텍스트 처리를 구현하는 데 필요한 인증 컨텍스트 템플릿을 설정하십시오. 이 절차는 아이덴티티 공급자와 서비스 공급자에서 동일합니다.

다음 단계를 수행하십시오.

1. Administrative UI 에 로그인합니다.
2. "페더레이션" 탭에서 "AuthnContext 템플릿"을 선택합니다.

   "인증 컨텍스트 템플릿 보기" 창이 열립니다.
3. "템플릿 만들기"를 선택합니다.

   첫 번째 단계의 템플릿 마법사가 열립니다.

4. 템플릿의 이름을 입력합니다.
5. 다음 작업 중 하나를 수행하십시오.
   - URI 를 수동으로 입력하고 "URI 추가"를 클릭합니다.
   - "기본 URI 로드"를 클릭하고 사전 정의된 목록에서 URI 를 선택합니다. URI 를 "사용 가능한 URI"에서 "선택한 URI" 목록으로 이동합니다.
6. 선택한 URI 를 강도 수준별로 정렬합니다. 강도 수준은 내림차순이며, 가장 강한 URI 가 맨 위쪽에 있고 가장 약한 URI 가 맨 아래쪽에 있습니다.
7. "다음"을 클릭합니다.
8. (선택 사항) 같은 강도 수준으로 지정해야 하는 URI 를 연속으로 배치하여 그룹화합니다. "Change Grouping"(그룹화 변경) 화살표를 사용하여 URI 를 그룹 내부로 또는 외부로 이동합니다.
9. 다음 작업은 SiteMinder 커넥터 배포에만 해당합니다.
   a. "보호 수준 사용"을 클릭합니다.
   b. 보호 수준을 인증 체계에서 URI 로 매핑합니다. 보호 수준은 1 부터 1000 까지 인증 체계의 강도를 나타내며 1000 이 가장 강력한 강도입니다. 개별 URI 는 고유한 보호 수준을 가질 수 있지만 URI 를 그룹화하면 해당 그룹 내의 URI 가 동일한 강도 수준을 공유합니다.

      보호 수준을 할당할 때 다음과 같은 내용을 고려하십시오.
      - 보호 수준은 내림차순으로 할당하십시오. 가장 강력한 컨텍스트를 맨 위에 나열하고 가장 약한 컨텍스트를 맨 아래에 나열하십시오.
      - 최대 보호 수준을 수정할 수 있으며 그렇게 하면 Administrative UI 에서 최소값을 계산합니다. Administrative UI 는 수준 범위에 빈 간격이 없는지 확인하여 각 보호 수준마다 연결된 URI 가 있도록 합니다.

   보호 수준 할당 (페이지 313)에 대한 자세한 내용을 참조하십시오.
10. "다음"을 클릭하여 마법사의 마지막 단계로 이동합니다.
11. "마침"을 선택하여 구성을 완료합니다.

템플릿이 완료되었습니다.

컨텍스트 템플릿에 대한 보호 수준 할당

SiteMinder 커넥터를 위임된 인증에 사용하는 페더레이션 배포 환경에서는 각 인증 URI 에 보호 수준을 연결해야 합니다. 보호 수준은 인증 강도의 신뢰 수준을 나타냅니다. 각 보호 수준은 URI 강도 수준에 매핑됩니다. 할당하는 보호 수준은 SiteMinder 인증 체계의 보호 수준을 반영해야 합니다.

참고: SiteMinder 커넥터를 사용하는 배포 환경에서는 보호 수준이 커넥터 인증 체계에 지정된 수준을 무시합니다.

Administrative UI 에서 보호 수준을 할당할 때 범위를 지정하십시오. 목록의 각 URI 에 대해 최대 수준을 지정하십시오. 최소 보호 수준은 목록에 나오는 URI 의 최대 수준을 기반으로 자동으로 계산됩니다. 이 범위에는 구성된 SiteMinder 인증 체계가 포함되어야 합니다. 예를 들어 SiteMinder 에서 보호 수준 20 의 X.509 인증 체계가 구성된 경우, CA SiteMinder?Federation Standalone 에 지정된 범위에 20 이 포함되어야 합니다.

보호 수준 예

SiteMinder 인증 체계	보호 수준
urn:oasis:names:tc:SAML:2.0:ac:classes:X509	20
urn:oasis:names:tc:SAML:2.0:ac:classes:MobileTwoFactorContract	15
urn:oasis:names:tc:SAML:2.0:ac:classes:InternetProtocol	10
urn:oasis:names:tc:SAML:2.0:ac:classes:Password	5

각 보호 수준은 URI 강도 수준에 매핑됩니다. 이 표에서는 원래 URI 목록을 보여 줍니다.

URI	보호 수준 최대	URI 강도
urn:oasis:names:tc:SAML:2.0:ac:classes:X509	1000	4
urn:oasis:names:tc:SAML:2.0:ac:classes:MobileTwoFactorContract	15	3
urn:oasis:names:tc:SAML:2.0:ac:classes:InternetProtocol	10	2
urn:oasis:names:tc:SAML:2.0:ac:classes:Password	5	1

이러한 범위는 SiteMinder 인증 체계의 보호 수준을 포함합니다. 예를 들면 다음과 같습니다.

- X509 체계는 16 에서 1000 까지의 보호 수준 포함
- MobileTwoFactorContract 는 11 에서 15 까지의 보호 수준 포함
- Internet Protocol 은 6 에서 10 까지의 보호 수준 포함
- Password 는 1 에서 5 까지의 보호 수준 포함

몇 개의 URI 를 그룹화하면 그룹화를 통해 서로 다른 보호 수준의 URI 가 동일한 URI 강도를 갖게 됩니다. 아래의 수정된 표에서는 그룹을 보여 줍니다.

URI	보호 수준 최대	URI 강도
urn:oasis:names:tc:SAML:2.0:ac:classes:X509	1000	3
urn:oasis:names:tc:SAML:2.0:ac:classes:MobileTwoFactorContract	800	3
urn:oasis:names:tc:SAML:2.0:ac:classes:InternetProtocol	700	2
urn:oasis:names:tc:SAML:2.0:ac:classes:Password	200	1

강도 수준의 범위는 목록에 있는 총 그룹 수를 반영합니다. 예를 들어 세 개의 그룹이 있는 경우 강도 수준의 범위는 1 부터 총 그룹 수인 3 까지입니다.

로컬 IdP 파트너 관계에서 인증 컨텍스트 기능 활성화

CA SiteMinder?Federation Standalone IdP 는 다음 두 가지 방법으로 어설션을 위한 인증 컨텍스트를 가져올 수 있습니다.

- 미리 정의된 인증 클래스를 사용합니다.

  인증 클래스에 대한 URI 를 지정하고 SP 의 컨텍스트 요청을 무시합니다. 하드 코딩된 항목은 IdP 에서 시작되는 싱글 사인온에 대한 기본 인증 컨텍스트의 역할을 할 수 있습니다.

- 인증 클래스를 자동으로 검색합니다. 이 기능은 SiteMinder 커넥터가 설정된 경우에만 사용할 수 있습니다.

  시스템은 인증 컨텍스트 템플릿을 사용하여 인증 컨텍스트를 자동으로 검색합니다.

  IdP 는 SP 의 인증 요청에 <RequestedAuthnContext> 요소가 포함되지 않은 경우에도 템플릿을 사용합니다. 요소가 있으면 IdP 에 의한 추가적인 평가가 트리거되며 IdP 가 어설션에 넣을 수 있는 항목의 선택 범위가 제한됩니다.

  인증 컨텍스트 처리의 흐름에 대한 자세한 정보를 참조할 수 있습니다.

인증 컨텍스트를 가져오는 방법을 구성할 수 있습니다.

다음 단계를 수행하십시오.

1. IdP->SP 파트너 관계 마법사의 SSO 및 SLO 단계로 이동합니다.
2. "인증" 섹션에서 인증 컨텍스트를 가져오는 방법을 지정합니다.
   - 로컬 인증에는 미리 정의된 인증 클래스를 사용해야 합니다.
   - SiteMinder 커넥터를 사용한 위임된 인증의 경우에는 미리 정의된 인증 클래스를 선택하거나 인증 컨텍스트 템플릿을 사용하여 클래스를 자동으로 검색할 수 있습니다.
3. 이전 단계에서 선택한 방법의 단계를 수행합니다.
   - 미리 정의된 클래스를 어설션에 포함하려면 "인증 클래스" 풀다운 메뉴에서 URI 를 선택합니다.
   - 세션 컨텍스트와 템플릿에서 클래스를 포함하려면 "인증 컨텍스트 템플릿" 필드에서 템플릿을 선택하거나 "템플릿 만들기"를 클릭합니다.

     참고: 이 옵션은 SiteMinder 커넥터가 설정된 경우에만 사용할 수 있습니다.
4. (선택 사항) 인증 컨텍스트를 가져오는 방법에 따라 "RequestedAuthnContext 무시" 확인란을 선택할 수도 있습니다.

다음 표에서는 "AuthnContext 구성" 및 "RequestedAuthnContext 무시" 설정이 함께 작동하는 방식을 보여 줍니다.

AuthnContext 구성	RequestedAuthnContext 무시	SP 에서 AuthnContext 요청	결과
미리 정의된 클래스	선택됨	예	IdP 는 <RequestedAuthnContext>를 무시하고 어설션에 정의된 값을 사용합니다.
미리 정의된 클래스	선택됨	아니요	IdP 는 기본적으로 어설션에 정의된 값을 반환합니다.
미리 정의된 클래스	선택되지 않음	예	IdP 가 인증 컨텍스트 요청을 처리하도록 구성되지 않았기 때문에 트랜잭션이 실패합니다. IdP 가 SP 에 오류 메시지를 반환합니다.
미리 정의된 클래스	선택되지 않음	아니요	IdP 는 기본적으로 어설션에 정의된 클래스 값을 반환합니다.
클래스 자동 감지	선택됨	예	IdP 는 인증 체계의 보호 수준을 인증 컨텍스트 템플릿과 비교하고 일치하는 인증 URI 를 어설션에 반환합니다. IdP 는 SP 요청의 값을 무시합니다.
클래스 자동 감지	선택됨	아니요	IdP 는 인증 체계의 보호 수준을 인증 컨텍스트 템플릿과 비교하고 일치하는 인증 URI 를 어설션에 반환합니다. IdP 는 SP 요청의 값을 무시합니다.
클래스 자동 감지	선택되지 않음	예	IdP 는 보호 수준을 SP 가 보내는 인증 컨텍스트 클래스와 비교합니다. IdP 는 인증 컨텍스트 템플릿을 사용하여 IdP 가 어설션에 넣는 인증 URI 를 결정합니다.

AuthnContext 구성	RequestedAuthnContext 무시	SP 에서 AuthnContext 요청	결과
클래스 자동 감지	선택되지 않음	아니요	IdP 는 인증 체계의 보호 수준을 인증 컨텍스트 템플릿과 비교하고 일치하는 인증 URI 를 어설션에 반환합니다.

로컬 SP 파트너 관계에서 인증 컨텍스트 요청이 사용되도록 설정

인증 컨텍스트는 어설션 인증 문의 일부이며 이는 사용자가 IdP 에서 인증되는 방법을 나타냅니다. SP 는 리소스에 대한 액세스 권한을 부여하기 전에 어설션에 신뢰 수준을 설정하기 위해 인증 프로세스에 대한 정보를 요청할 수 있습니다.

인증 컨텍스트 URI 는 <AuthnContext> 요소 안의 <AuthnContextClassRef> 요소의 값입니다. 각 URI 는 SP 가 IdP 에서 어설션으로 반환하기를 원하는 컨텍스트 클래스를 식별합니다.

SP 의 인증 컨텍스트 템플릿은 다음 정보를 정의합니다.

- SP 가 IdP 에서 수신하고자 하는 URI. 나가는 요청의 경우 템플릿의 URI 는 요청된 리소스에 대한 액세스를 허용하기 전에 SP 가 허용하는 인증 컨텍스트를 나타냅니다.
- 요청의 URI 를 IdP 에 정의된 URI 에 비교하는 방법
- SP 가 URI 를 사용하는 방법. SP 는 나가는 인증 요청에 URI 를 포함할 수 있습니다. 또한 SP 는 들어오는 어설션 응답에서 URI 의 유효성을 검사할 수도 있습니다. URI 사용을 두 기능 모두에 대해 구성할 수 있습니다.

파트너 관계 단위로 템플릿을 선택할 수 있으며 여러 파트너 관계가 하나의 템플릿을 사용할 수 있습니다.

인증 컨텍스트 요청이 사용되도록 설정하거나 SP 파트너 관계를 구성할 때 인증 컨텍스트 템플릿을 구성하십시오.

다음 단계를 수행하십시오.

1. Administrative UI 에 로그인합니다.
2. 편집할 SP->IdP 파트너 관계를 선택합니다.
3. 파트너 관계 마법사의 "AuthnContext 구성" 단계로 이동합니다.

   구성 대화 상자가 열립니다.
4. "인증 컨텍스트 처리 사용" 확인란을 선택합니다.
5. 대화 상자의 필드를 입력합니다. "도움말"을 클릭하면 필드, 컨트롤 및 해당되는 각 요구 사항에 대한 설명을 볼 수 있습니다.

   다음 정보에 주의하십시오.

   - 인증 컨텍스트 템플릿이 없는 경우 템플릿 만들기를 선택합니다.
   - "비교" 필드에서는 SP 인증 요청의 URI 를 아이덴티티 공급자에 구성된 URI 와 비교하는 방법을 보여 줍니다.

     "도움말"에 각 비교 연산자에 대한 자세한 설명이 있습니다.
   - "사용 가능한 URI" 목록에서 URI 를 선택하면 사용 가능한 URI 는 선택된 템플릿에 대해 구성된 URI 를 반영합니다. 미리 정의된 템플릿이 없는 경우 "템플릿 만들기"를 클릭하여 구성합니다.

인증 컨텍스트 요청은 아이덴티티 공급자로 전송되는 인증 요청에 포함됩니다.

제 18 장: 페더레이션 메시지 서명 및 암호화

어설션에 보안을 적용하고 어설션 내의 데이터를 암호화하는 것은 파트너 관계 구성의 중요한 부분입니다. 서명 단계(SAML 1.1 및 WS-페더레이션)와 서명 및 암호화 단계(SAML 2.0)에서는 어설션의 서명 및 암호화를 구성할 수 있습니다.

SAML 2.0 의 경우 서명 태스크에 대한 서명 알고리즘을 선택할 수 있습니다. 알고리즘을 선택할 수 있으므로 다음과 같은 사용 사례가 지원됩니다.

- IdP 가 RSAwithSHA1 을 사용하는 어설션, 응답 및 SLO-SOAP 메시지 또는 RSAwithSHA256 알고리즘에 서명하는 IdP-->SP 파트너 관계
- SP 가 RSAwithSHA1 을 사용하는 인증 요청과 SLO-SOAP 메시지 또는 RSAwithSHA256 알고리즘에 서명하는 SP-->IdP 파트너 관계

서명 확인은 서명된 문서에서 사용 중인 알고리즘을 자동으로 감지하고 이를 확인합니다. 따라서 서명 확인을 위한 구성은 필요하지 않습니다.

이 섹션은 다음 항목을 포함하고 있습니다.

SAML 1.1 생산자 및 WSFED IP 에서 서명 구성

"서명" 단계에서는 정책 서버가 개인 키와 인증서를 사용하여 SAML 어설션 또는 WS-페더레이션 토큰 응답에 서명하는 방식을 정의할 수 있습니다. SAML 1.1 의 경우 어설션 응답 대신 어설션에만 서명하도록 선택할 수 있습니다.

SAML 1.1 과 WS-페더레이션은 암호화를 지원하지 않습니다.

인증서 데이터 저장소에 여러 개의 개인 키와 인증서가 있을 수 있습니다. 여러 개의 페더레이션된 파트너가 있는 경우 각 파트너마다 다른 키 쌍을 사용할 수 있습니다.

참고: 시스템이 FIPS_COMPAT 또는 FIPS_MIGRATE 모드에서 작동하는 경우 풀다운 목록에서 모든 인증서와 키 항목을 사용할 수 있습니다. 시스템이 FIPS 전용 모드에서 작동하는 경우에는 FIPS 승인 인증서 및 키 항목만 사용할 수 있습니다.

다음 단계를 수행하십시오.

1. Administrative UI 에 로그인합니다.
2. 수정하려는 어설션 당사자-신뢰 당사자 파트너 관계를 선택합니다.
3. 파트너 관계 마법사의 "서명" 단계로 이동합니다.
4. "서명" 섹션에 있는 "서명 개인 키 별칭" 필드의 풀다운 목록에서 별칭을 선택합니다.

   인증서 데이터 저장소에 개인 키가 없는 경우 "가져오기"를 클릭하여 키를 가져옵니다. 또는 "생성"을 클릭하여 인증서 요청을 생성합니다.

   이 필드에 데이터를 입력하면 어설션 당사자가 어설션 및 응답에 서명하기 위해 사용하는 개인 키가 지정됩니다.
5. (SAML 1.1 만 해당) "아티팩트" 및 "Post" 서명 옵션에서는 서명을 원하는 특정 구성 요소(어설션, 응답)를 선택합니다.

테스트 환경에서 SiteMinder 를 사용 중인 경우 테스트를 단순화하기 위해 서명 처리를 중지할 수 있습니다. "서명 처리 사용 안 함" 확인란을 클릭하십시오.

서명 구성이 완료되었습니다.

SAML 1.1 소비자 및 WSFED RP 에서의 서명 확인

"서명" 단계에서는 정책 서버가 개인 키와 인증서를 사용하여 SAML 어설션 또는 WS-페더레이션 토큰 응답을 확인하는 방식을 정의할 수 있습니다. SAML 1.1 의 경우 어설션만 확인하도록 선택할 수 있습니다.

SAML 1.1 과 WS-페더레이션은 암호화를 지원하지 않습니다.

인증서 데이터 저장소에 여러 개의 개인 키와 인증서가 있을 수 있습니다. 여러 개의 페더레이션된 파트너가 있는 경우 각 파트너마다 다른 키 쌍을 사용할 수 있습니다.

참고: 시스템이 FIPS_COMPAT 또는 FIPS_MIGRATE 모드에서 작동하는 경우 풀다운 목록에서 모든 인증서와 키 항목을 사용할 수 있습니다. 시스템이 FIPS 전용 모드에서 작동하는 경우에는 FIPS 승인 인증서 및 키 항목만 사용할 수 있습니다.

다음 단계를 수행하십시오.

1. Administrative UI 에 로그인합니다.
2. 수정하려는 신뢰 당사자-어설션 당사자 파트너 관계를 선택합니다.
3. 파트너 관계 마법사의 "서명" 단계로 이동합니다.
4. "확인 인증서 별칭" 필드에서 인증서 데이터 저장소의 별칭을 선택합니다.

   이 필드에 데이터를 입력하면 서명된 어설션, 응답 또는 둘 다를 확인하는 인증서가 지정됩니다. 인증서 데이터 저장소에 인증서가 없는 경우 "가져오기"를 클릭하여 가져옵니다. 또는 "생성"을 클릭하여 인증서 요청을 생성합니다.

참고: 테스트 환경에서 제품을 사용 중인 경우 테스트를 단순화하기 위해 서명 처리를 중지할 수 있습니다. "서명 처리 사용 안 함" 확인란을 클릭하십시오.

서명 구성이 완료되었습니다.

SAML 2.0 IdP 에서의 서명 구성

파트너 관계 마법사의 "서명 및 암호화" 단계에서는 제품에서 다음의 서명 기능에 개인 키 및 인증서가 사용되는 방식을 정의할 수 있습니다.

- SAML 어설션, 어설션 응답 및 인증 요청을 서명하고 확인합니다.

  SAML 2.0 POST 바인딩의 경우 어설션에 서명해야 합니다.
- 싱글 로그아웃 응답 및 요청에 서명합니다(HTTP-리디렉션 및 SOAP 바인딩).

인증서 데이터 저장소에 여러 개의 개인 키와 인증서가 있을 수 있습니다. 여러 개의 페더레이션된 파트너가 있는 경우 각 파트너마다 다른 키 쌍을 사용할 수 있습니다.

참고: 시스템이 FIPS_COMPAT 또는 FIPS_MIGRATE 모드에서 작동하는 경우 풀다운 목록에서 모든 인증서와 키 항목을 사용할 수 있습니다. 시스템이 FIPS 전용 모드에서 작동하는 경우에는 FIPS 승인 인증서 및 키 항목만 사용할 수 있습니다.

서명 옵션을 구성하려면

1. 파트너 관계 마법사에서 "서명 및 암호화" 단계를 선택합니다.
2. "서명" 섹션에서 "서명 개인 키 별칭" 필드에 대한 별칭을 선택합니다. 사용 가능한 개인 키가 없는 경우 "가져오기"를 클릭하여 가져오십시오. 또는 "생성"을 클릭하여 인증서 요청을 생성하십시오.

   이 필드에 데이터를 입력하면 어설션 당사자가 어설션, 싱글 로그아웃 요청 및 응답에 서명하기 위해 사용하는 개인 키가 지정됩니다.

   참고: 필드의 설명을 보려면 "도움말"을 클릭하십시오.
3. "서명 알고리즘" 필드에서 디지털 서명에 대한 해시 알고리즘을 선택합니다. IdP 는 지정된 알고리즘을 사용하여 어설션, 응답 및 SLO-SOAP 메시지에 서명합니다.

   응용 프로그램에 가장 적합한 알고리즘을 선택하십시오.

   RSAwithSHA256 이 RSAwithSHA1 보다 결과 암호화 해시 값에 사용되는 비트 수가 많으므로 더 안전합니다.

   선택하는 알고리즘은 시스템의 모든 서명 기능에 사용됩니다.
4. 인증서 데이터 저장소에서 또는 "확인 인증서 별칭" 필드에서 별칭을 선택합니다.

   이 필드에 데이터를 입력하면 서명된 인증 요청이나 싱글 로그아웃 요청 또는 응답을 확인하는 인증서가 지정됩니다. 데이터베이스에 인증서가 없는 경우 "가져오기"를 클릭하여 가져오십시오.
5. (선택 사항) 어설션이나 응답 또는 둘 다에 대한 아티팩트 및 POST 서명 옵션을 지정합니다.
6. (선택 사항) 싱글 로그아웃을 사용할 때 로그아웃 요청, 로그아웃 응답 또는 둘 다에 대한 SLO SOAP 서명 옵션을 지정합니다.
7. (선택 사항) "서명된 인증 요청 필요"에 대한 확인란을 선택합니다. 이 확인란은 어설션 당사자가 신뢰 당사자의 서명된 요청만 수락하는 것을 확인합니다.

파트너 관계를 활성화하여 구성 변경 내용을 모두 적용하고 파트너 관계를 사용 가능한 상태로 만듭니다. 서비스를 다시 시작하는 것만으로는 충분하지 않습니다.

테스트 환경에서 제품을 사용 중인 경우 테스트를 단순화하기 위해 서명 처리를 중지할 수 있습니다. "서명 처리 사용 안 함" 확인란을 클릭하십시오.

중요! SAML 2.0 프로덕션 환경에서는 서명 처리가 사용되도록 설정하십시오.

SAML 2.0 IdP 에서의 암호화 구성

파트너 관계 마법사의 "서명 및 암호화" 단계에서는 정책 서버가 다음 태스크를 수행하기 위해 개인 키 및 인증서를 사용하는 방법을 정의할 수 있습니다.

- SAML 어설션, 어설션 응답 및 인증 요청을 서명하고 확인합니다.

  SAML 2.0 POST 바인딩의 경우 어설션에 서명해야 합니다.
- 싱글 로그아웃 응답 및 요청에 서명합니다(HTTP-리디렉션 및 SOAP 바인딩).
- 전체 어설션, 이름 ID 및 특성을 암호화 및 암호 해독합니다.

인증서 데이터 저장소에 여러 개의 개인 키와 인증서가 있을 수 있습니다. 여러 개의 페더레이션된 파트너가 있는 경우 각 파트너마다 다른 키 쌍을 사용할 수 있습니다.

암호화 옵션을 구성하려면

1. "암호화" 섹션에서 다음 확인란 중 하나 또는 둘 다를 선택하여 암호화할 어설션 데이터를 지정합니다.
   - 이름 ID 암호화
   - 어설션 암호화
2. "암호화 인증서 별칭"에서 인증서 데이터 저장소의 인증서 별칭을 선택합니다.

   이 인증서는 어설션 데이터를 암호화합니다. 사용할 수 있는 인증서가 없는 경우에는 "가져오기"를 클릭하여 가져옵니다.

3. "암호화 블록 알고리즘" 및 "암호화 키 알고리즘" 필드에 대한 값을 선택합니다.

   다음의 블록/키 알고리즘 조합에서 인증에 필요한 최소 키 크기는 1024 비트입니다.

   - 암호화 블록 알고리즘: 3DES

     암호화 키 알고리즘: RSA-OEAP

   - 암호화 블록 알고리즘: AES-256

     암호화 키 알고리즘: RSA-OEAP

     참고: AES-256 비트 암호화 블록 알고리즘을 사용하려면 JCE(Java Cryptography Extension) Unlimited Strength Jurisdiction Policy Files 를 설치하십시오. 이 파일은 http://java.sun.com/javase/downloads/index.jsp 에서 다운로드할 수 있습니다.

암호화 구성이 완료되었습니다.

SAML 2.0 SP 에서의 서명 구성

파트너 관계 마법사의 "서명 및 암호화" 단계에서는 정책 서버가 다음 태스크를 수행하기 위해 개인 키 및 인증서를 사용하는 방법을 정의할 수 있습니다.

- SAML 어설션 서명 및 어설션 응답을 확인하고 인증 요청에 서명합니다.

  참고: SAML 2.0 POST 바인딩의 경우 IdP 가 어설션에 서명해야 합니다.

- 싱글 로그아웃 응답 및 요청에 서명합니다(HTTP-리디렉션 및 SOAP 바인딩).

인증서 데이터 저장소에 여러 개의 개인 키와 인증서가 있을 수 있습니다. 여러 개의 페더레이션된 파트너가 있는 경우 각 파트너마다 다른 키 쌍을 사용할 수 있습니다.

참고: 시스템이 FIPS_COMPAT 또는 FIPS_MIGRATE 모드에서 작동하는 경우 풀다운 목록에서 모든 인증서와 키 항목을 사용할 수 있습니다. 시스템이 FIPS 전용 모드에서 작동하는 경우에는 FIPS 승인 인증서 및 키 항목만 사용할 수 있습니다.

서명 옵션을 구성하려면

1. 파트너 관계 마법사에서 "서명 및 암호화" 단계를 선택하여 시작합니다.
2. "서명" 섹션의 "서명 개인 키 별칭" 필드에서 인증서 데이터 저장소의 별칭을 선택합니다. 데이터베이스에 개인 키가 없는 경우 "가져오기"를 클릭하여 가져옵니다. 또는 "생성"을 클릭하여 키 쌍을 만들고 인증서 요청을 생성합니다.

   이 필드에 데이터를 입력하면 신뢰 당사자가 인증 요청과 싱글 로그아웃 요청 및 응답에 서명하는 데 사용하는 개인 키가 지정됩니다.

   참고: "도움말"을 클릭하면 필드, 컨트롤 및 해당되는 각 요구 사항에 대한 설명을 볼 수 있습니다.
3. "서명 알고리즘" 필드에서 디지털 서명에 대한 해시 알고리즘을 선택합니다. SP 는 지정된 알고리즘을 사용하여 인증 요청 및 SLO-SOAP 메시지에 서명합니다.

   응용 프로그램에 가장 적합한 알고리즘을 선택하십시오.

   RSAwithSHA256 이 RSAwithSHA1 보다 결과 암호화 해시 값에 사용되는 비트 수가 많으므로 더 안전합니다.

   SiteMinder 는 선택하는 알고리즘을 모든 서명 기능에 사용합니다.
4. "확인 인증서 별칭" 필드에서 인증서 데이터 저장소의 별칭을 선택합니다.

   이 필드에 데이터를 입력하면 신뢰 당사자가 서명된 어설션이나 싱글 로그아웃 요청 및 응답을 확인하는 데 사용하는 인증서가 지정됩니다. 데이터베이스에 인증서가 없는 경우 "가져오기"를 클릭하여 가져오십시오.
5. (선택 사항) SP 가 모든 인증 요청에 서명하도록 하려면 "서명 인증 요청"을 선택합니다. 원격 어설션 당사자가 인증 요청에 서명을 요구하는 경우 이 옵션을 선택합니다.

   파트너 관계를 활성화하여 구성 변경 내용을 모두 적용하고 파트너 관계를 사용 가능한 상태로 만듭니다. 서비스를 다시 시작하는 것만으로는 충분하지 않습니다.

테스트 환경에서 SiteMinder 를 사용 중인 경우 테스트를 단순화하기 위해 서명 처리를 중지할 수 있습니다. 기능을 사용하지 않으려면 "서명 처리 사용 안 함" 확인란을 클릭하십시오.

중요! SAML 2.0 프로덕션 환경에서는 서명 처리가 사용되도록 설정하십시오.

SAML 2.0 SP 에서의 암호화 구성

"서명 및 암호화" 단계에서는 어설션, 이름 ID, 특성의 암호화 및 암호 해독을 포함하여 SP 가 개인 키 및 인증서를 사용하는 방법을 구성할 수 있습니다.

인증서 데이터 저장소에 여러 개의 개인 키와 인증서가 있을 수 있습니다. 여러 개의 페더레이션된 파트너가 있는 경우 각 파트너마다 다른 키 쌍을 사용할 수 있습니다.

참고: 시스템이 FIPS_COMPAT 또는 FIPS_MIGRATE 모드에서 작동하는 경우 풀다운 목록에서 모든 인증서와 키 항목을 사용할 수 있습니다. 시스템이 FIPS 전용 모드에서 작동하는 경우에는 FIPS 승인 인증서 및 키 항목만 사용할 수 있습니다.

암호화 옵션을 구성하려면

1. "암호화" 섹션에서 다음 확인란 중 하나 또는 둘 다를 선택하여 어설션에서 올바른 데이터가 암호화되도록 합니다.
   - 암호화된 이름 ID 필요
   - 암호화된 어설션 필요

   참고: AES-256 비트 암호화 블록 알고리즘을 사용하려면 Sun JCE(Java Cryptography Extension) Unlimited Strength Jurisdiction Policy Files 를 설치하십시오. 이 파일은 http://java.sun.com/javase/downloads/index.jsp 에서 다운로드할 수 있습니다.
2. "암호 해독 개인 키 별칭"에 대해 인증서 데이터 저장소의 별칭을 선택합니다.

   이 개인 키는 암호화된 어설션 데이터를 암호 해독합니다. 사용할 수 있는 인증서가 없는 경우 "가져오기"를 클릭하여 가져오거나 "생성"을 클릭하여 키 쌍을 만들고 인증서 요청을 생성합니다.

암호화 구성이 완료되었습니다.

제 19 장: 서비스 공급자 측 세션 기간 관리

이 섹션은 다음 항목을 포함하고 있습니다.

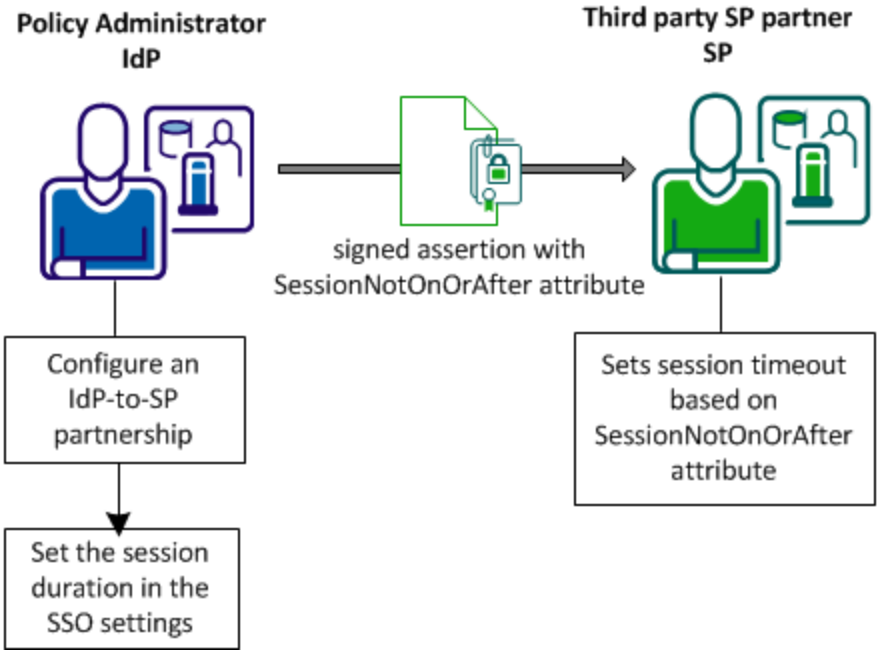
서비스 공급자에서 인증 세션 기간을 관리하는 방법

서비스 공급자에서 인증 세션의 기간을 관리할 수 있습니다. SessionNotOnOrAfter 특성은 IdP 가 어설션의 <AuthnStatement>에 포함할 수 있는 선택적 특성입니다.

참고: SessionNotOnOrAfter 매개 변수는 어설션의 유효 기간을 결정하는 NotOnOrAfter 매개 변수와는 다릅니다.

세션 기간을 결정하는 이유는 SP 에서 세션이 너무 짧은 경우에 사용자가 다시 인증을 받아야 하는 번거로움을 줄이기 위해서입니다. 타사 SP 는 SessionNotOnOrAfter 의 값을 사용하여 자체 시간 만료 값을 설정할 수 있으므로 너무 짧은 세션을 방지하는 데 도움이 됩니다. 사용자 세션이 무효화되면 사용자는 아이덴티티 공급자에서 다시 인증해야 합니다. 사용자에게 편리한 환경을 제공하려면 SP 에서 세션을 적절하게 관리해야 합니다.

다음 그림에서는 IdP 에서 수행해야 하는 구성 단계와 그 결과로 타사 SP 가 수행하는 작업을 보여 줍니다.



어설션에 세션 기간 특성 포함

세션 기간의 구성은 IdP 에서 수행됩니다. SP 에 전송된 어설션에는 SP 사이트의 시간 만료 값을 설정하는 데 사용되는 세션 특성이 포함됩니다.

중요! CA SiteMinder® Federation Standalone 이 SP 로 작동하고 있는 경우 SessionNotOnOrAfter 값이 무시됩니다. 대신 SP 는 대상 리소스를 보호하는 SAML 인증 체계에 해당하는 영역 시간 만료를 바탕으로 세션 시간 만료를 설정합니다.

다음 단계를 수행하십시오.

1. Administrative UI 에 로그인합니다.
2. 수정할 IdP->SP 파트너 관계를 선택합니다.
3. "SSO 및 SLO" 단계로 이동합니다.

4. "SSO" 섹션에서 "권장되는 SP 세션 기간"에 대한 옵션을 선택합니다. 사용자 지정 옵션을 선택하면 다음 옵션 중 하나를 선택할 수 있습니다.
   - 특성 생략
   - 특성을 IdP 세션 시간 만료 값으로 설정
   - 기간 직접 지정

   참고: "도움말"을 클릭하면 필드, 컨트롤 및 해당되는 각 요구 사항에 대한 설명을 볼 수 있습니다.
5. 변경을 완료하고 "마침"을 클릭한 후 "확인" 단계를 선택합니다.

구성에 기반하여 세션 특성이 어설션에 포함되어 SP 로 전송됩니다.

제 20 장: SiteMinder 와 CA SiteMinder® Federation Standalone 통합

이 섹션은 다음 항목을 포함하고 있습니다.

CA SiteMinder® Federation Standalone 과 SiteMinder 의 통합 방법

CA SiteMinder?Federation Standalone 에 포함된 소프트웨어 구성 요소인 SiteMinder 커넥터를 사용하면 이미 배포되어 있는 SiteMinder 시스템을 CA SiteMinder?Federation Standalone 과 통합할 수 있습니다. 이 커넥터는 페더레이션과 배포된 웹 액세스 관리 제품 사이에 다음과 같은 상호 작용을 지원합니다.

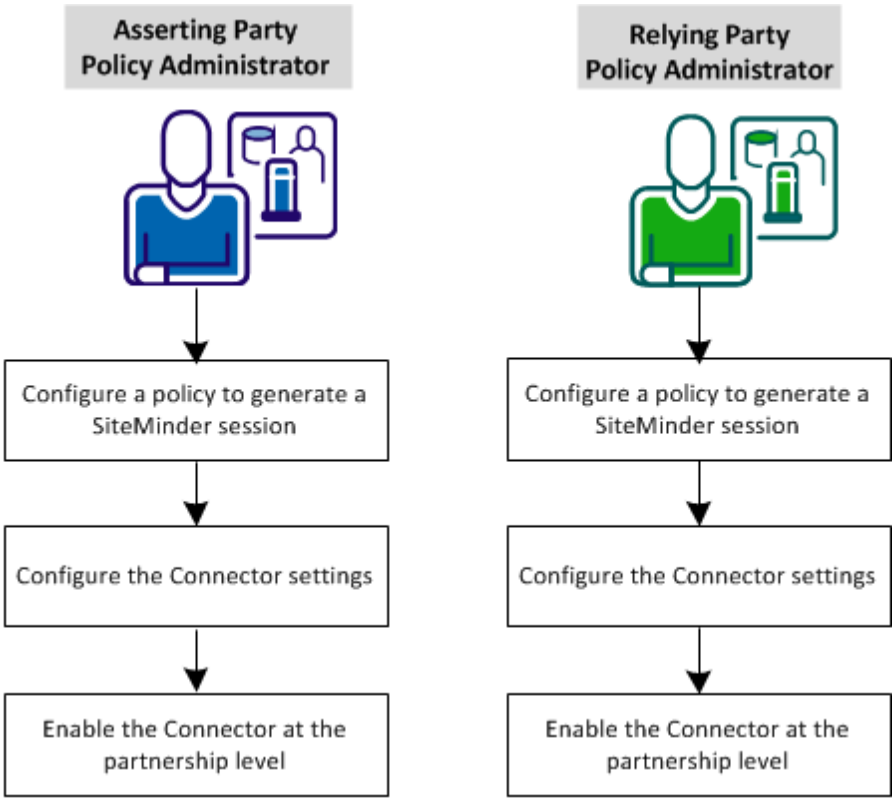
- 어설션을 생성하는 아이덴티티를 설정합니다.

  어설션 당사자 측에서 사용자가 CA SiteMinder® Federation Standalone 에 액세스하지만 세션이 아직 없습니다. 커넥터는 SiteMinder 와 통신하여 SiteMinder 세션을 설정합니다. 커넥터는 SiteMinder 세션의 내용에 따라 페더레이션 세션을 생성하고, 이 세션 정보를 사용하여 사용자에 대해 SAML 어설션이 생성됩니다.

- SiteMinder 가 권한 부여 권한을 결정하기 위한 아이덴티티를 설정합니다.

  신뢰 당사자 측에서는 사용자가 CA SiteMinder® Federation Standalone 에서 인증되고 페더레이션 세션이 생성됩니다. 커넥터는 페더레이션 세션과 사용자 이름을 SiteMinder 에 전달하고, 그 결과로 페더레이션 세션에서 SiteMinder 세션이 생성됩니다. 이제 사용자가 확인되었기 때문에 시스템에서 사용자에게 자격 증명을 다시 요청하지 않습니다. SiteMinder 는 신뢰 당사자 측에서 요청된 리소스에 대한 권한 부여 권한을 결정합니다.

다음 그림에서는 커넥터를 사용하여 통합할 경우의 구성 프로세스를 보여 줍니다.



다음 구성 단계를 완료하십시오.

1. SiteMinder 세션을 생성하는 정책을 구성합니다.
2. 커넥터 설정을 구성합니다.
3. 파트너 관계 수준에서 커넥터를 사용하도록 설정합니다.

SiteMinder 커넥터를 통해 SiteMinder 와 통합

SiteMinder 커넥터는 다음과 같은 통합을 지원합니다.

- 어설션을 생성하는 아이덴티티를 설정합니다.

  어설션 당사자 측에서 사용자가 CA SiteMinder® Federation Standalone 에 액세스하지만 세션이 아직 없습니다. 커넥터는 SiteMinder 와 통신하여 SiteMinder 세션을 설정합니다. 세션 정보를 사용하면 페더레이션 세션이 만들어지고 해당 사용자의 SAML 어설션이 생성됩니다. 사용자는 이 어설션을 사용하여 신뢰 당사자에 있는 보호된 페더레이션 리소스에 액세스할 수 있습니다.

- 어설션에서 권한 부여 권한을 결정합니다.

  신뢰 당사자 측에서는 사용자가 CA SiteMinder® Federation Standalone 에서 인증되고 페더레이션 세션이 생성됩니다. 커넥터는 페더레이션 세션과 사용자 이름을 SiteMinder 에 전달하고, 그 결과로 SiteMinder 세션이 생성됩니다. 이 세션이 설정되면 이러한 사용자는 보호된 리소스에 액세스할 때 다시 인증을 요청 받지 않습니다. 이제 사용자가 확인되었으며 신뢰 당사자 측에서 사용자의 액세스 권한을 결정할 수 있습니다.

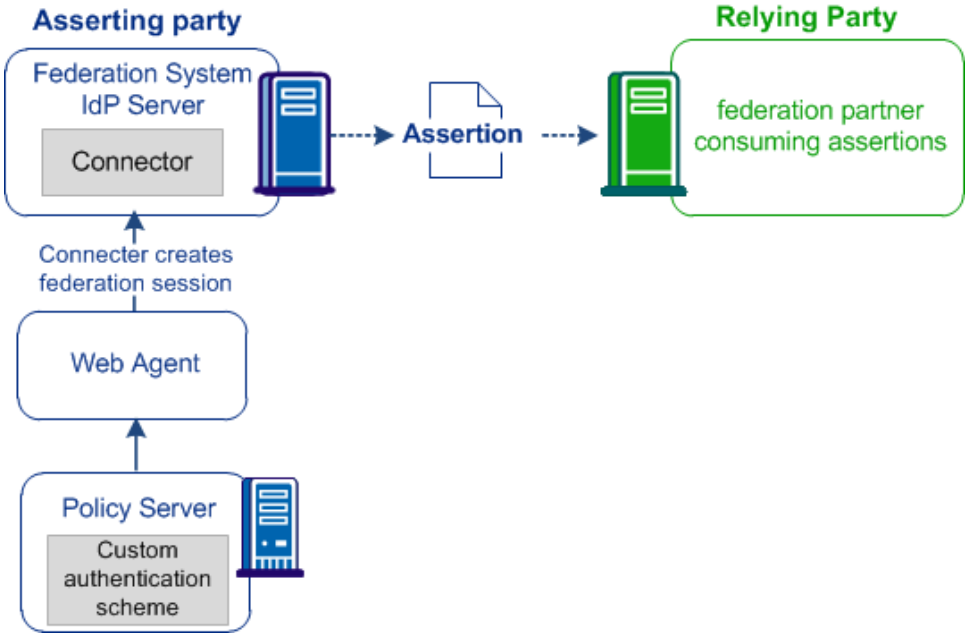
FEDSESSION 쿠키에는 다음과 같은 시간 만료 설정이 사용됩니다.

- 유휴 시간 만료: 600 초(10 분)
- 최대 시간 만료: 900 초(15 분)

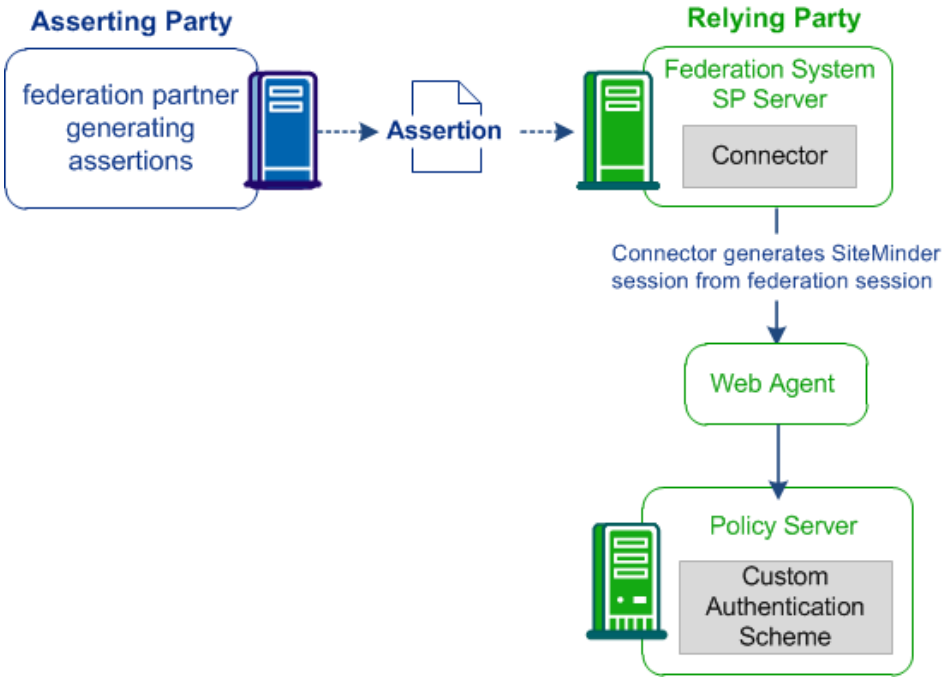
이러한 시간 만료 설정은 UI 에서 변경할 수 없습니다.

다음 그림에서 볼 수 있듯이 커넥터를 사용하려면 SiteMinder 환경과 CA SiteMinder?Federation Standalone 환경에서 다음과 같이 구성해야 합니다.

다음 그림에서는 어설션 당사자의 커넥터를 보여 줍니다.



다음 그림에서는 신뢰 당사자의 커넥터를 보여 줍니다.



각 사이트에 세션을 생성하는 정책 구성

SiteMinder 커넥터를 통해 CA SiteMinder?Federation Standalone 은 기존 정책 서버와 작동할 수 있습니다. 첫 번째 단계는 정책을 구성하는 것입니다. 어설션 당사자 측에서 정책이 페더레이션 세션을 생성합니다. 신뢰 당사자 측에서 정책이 SiteMinder 세션을 생성합니다. 이 정책은 다른 모든 정책과 마찬가지로 동작하지만 리소스를 보호하는 것이 아니라 세션을 트리거하는 것이 주요 목표입니다.

참고: 어설션 당사자와 신뢰 당사자 측에서 정책을 구성하십시오.

이 정책에는 일반적인 정책 개체를 구성할 뿐 아니라 사용자 지정 SiteMinder 커넥터 인증 체계도 적용해야 합니다. 이 정책은 커넥터 설정과 관련됩니다.

정책 서버 개체를 구성하려면 *정책 서버 구성 안내서*를 참조하십시오.

중요! 커넥터를 구성하기 전에 정책 서버에서 다음 단계를 완료하십시오.

다음 단계를 수행하십시오.

1. 페더레이션 시스템에 smauthconnectors.zip 아카이브의 압축을 풉니다. 이 아카이브는 페더레이션 제품 키트에 들어 있습니다.
2. SiteMinder 운영 환경에 맞는 사용자 지정 인증 체계 라이브러리를 선택합니다.
   - **Windows:** smauthsmconnector.dll
   - **Solaris/Linux:** libsmauthsmconnector.so

   참고: UNIX 플랫폼에서는 이름의 대/소문자를 구분합니다.
3. 라이브러리를 SiteMinder 시스템의 해당 정책 서버 디렉터리에 복사합니다.
   - **Windows:** *policy_server_home*/siteminder/bin
   - **Solaris/Linux:** *policy_server_home*/siteminder/lib
4. SiteMinder Administrative UI 에 로그온합니다.
5. 페더레이션 시스템을 나타내는 웹 에이전트를 생성합니다. 예를 들어 이름을 "페더레이션 에이전트"로 지정합니다.

   중요! 4.x 에이전트 지원 옵션은 선택하지 마십시오.
6. 에이전트 구성을 지정하는 에이전트 구성 개체를 생성하고 DefaultAgentName 설정의 값을 지정합니다. 개체에는 이 설정만으로 충분합니다.
7. 호스트 구성 개체를 생성합니다.

   호스트 구성 개체는 트러스트된 호스트와 정책 서버 사이의 연결을 정의합니다. 호스트 구성 개체는 페더레이션 시스템과 정책 서버를 통합하기 위해 페더레이션 시스템이 연결할 수 있는 정책 서버를 정의합니다.

   페더레이션 시스템이 기존 호스트 구성 개체에 지정된 하나 이상의 정책 서버에 연결하려는 경우 해당 개체를 사용합니다. 그렇지 않은 경우에는 페더레이션 시스템과 정책 서버 사이의 연결을 위해 개체를 생성합니다.

8. 다음과 같은 값을 사용하여 사용자 지정 커넥터 인증 체계를 생성합니다.

   라이브러리

   smauthsmconnector

   이 값은 대/소문자를 구분합니다.

   암호

   alphanumeric string

   이 필드의 값은 Administrative UI 에서 "커넥터" 설정의 "공유 암호" 값과 일치해야 합니다.

9. 페더레이션 제품에 대한 정책 도메인을 생성합니다. 이 도메인은 SiteMinder 세션을 생성하기 위해 정책에 추가하는 필수 영역과 리소스를 포함해야 합니다.

10. 페더레이션 시스템 및 정책 서버에 사용되는 사용자 디렉터리를 구성한 도메인에 추가합니다.

11. 다음 값을 사용하여 영역을 생성합니다.

    에이전트

    이전 단계에서 지정한 웹 에이전트를 지정합니다.

    리소스 필터

    더미 디렉터리(예: /federation/)를 지정합니다. 이 디렉터리는 웹 서버에 없어도 됩니다.

    인증 체계

    앞서 생성한 사용자 지정 인증 체계에 지정한 이름을 입력합니다.

12. 다음 값을 사용하여 규칙을 생성합니다.

    리소스

    *

    작업

    웹 에이전트 - Get 및 Post

13. 다음 설정을 사용하여 정책을 생성합니다.

    사용자

    페더레이션 시스템과 SiteMinder 가 공유하는 사용자 디렉터리의 사용자를 지정합니다.

    규칙

    커넥터에 대해 생성한 규칙을 추가합니다.

CA SiteMinder?Federation Standalone 과 통신할 때 SiteMinder 세션을 생성하는 정책이 만들어졌습니다.

커넥터 설정 구성

커넥터가 SiteMinder 와 상호 작용하기 위해서는 CA SiteMinder?Federation Standalone Administrative UI 에서 커넥터 설정을 구성해야 합니다. 커넥터를 사용하는 모든 파트너 관계는 하나의 구성을 사용하며 하나의 SiteMinder 환경에 연결합니다.

다음 단계를 수행하십시오.

1. Administrative UI 에 로그인합니다.
2. "인프라" 탭으로 이동합니다.
3. "배포 설정"을 선택합니다.

   "배포 설정 구성" 대화 상자가 열립니다.
4. "SiteMinder 커넥터 설정" 섹션의 모든 필드에 값을 지정합니다. 다음 고려 사항에 주의하십시오.
   - 특정 파트너 관계에 대해 커넥터를 사용하거나 사용하지 않으려면 파트너 관계 수준에서 설정해야 합니다.
   - 커넥터를 전역으로 사용하거나 사용하지 않으려면 배포 설정에 있는 확인란을 사용합니다.

   중요! 전역 수준에서 커넥터를 사용하지 않도록 설정하면 CA SiteMinder® Federation Standalone 의 파트너 관계 수준에서 확인란이 무시됩니다.

   참고: "도움말"을 클릭하면 필드, 컨트롤 및 해당되는 각 요구 사항에 대한 설명을 볼 수 있습니다.

5. "호스트 등록"을 선택하고 SiteMinder 정책 서버의 레거시 관리자 자격 증명을 지정합니다. 레거시 관리자만 호스트 등록을 수행할 수 있습니다.

   이 단계에서는 CA SiteMinder® Federation Standalone 이 SiteMinder 정책 서버에 에이전트로 등록됩니다.

   참고: 정책 서버를 두 개 이상 지정하여 호스트 등록 프로세스에서 장애 조치를 지원하도록 구성할 수 있습니다. 기본 정책 서버에서 등록이 실패할 경우 등록 프로세스가 성공적으로 완료될 때까지 지정된 다음 정책 서버에서 등록 프로세스가 시도됩니다.

6. "저장"을 클릭합니다.

   중요! 호스트 등록을 완료한 후 "SiteMinder 커넥터 설정" 섹션에서 "저장"을 선택하십시오.

7. 운영 환경에 맞게 페더레이션 서비스를 다시 시작합니다.

   - **Windows**

     다음과 같이 중지 및 시작 바로 가기를 사용합니다. 로컬 관리자가 아닌 네트워크 사용자로 로그인한 경우에는 바로 가기를 마우스 오른쪽 단추로 클릭하고 "관리자 권한으로 실행"을 선택하십시오.

     a. "시작", "모든 프로그램", "CA", "Federation Standalone", "서비스 중지"

     b. "시작", "모든 프로그램", "CA", "Federation Standalone", "서비스 시작"

   - **UNIX**

     a. 명령 창을 엽니다.

     b. 다음 스크립트를 실행합니다.

     federation_install_dir/fedmanager.sh stop

     federation_install_dir/fedmanager.sh start

     참고: 서비스를 중지했다가 루트 사용자로 시작하지 마십시오.

SiteMinder 커넥터 구성이 완료되었습니다.

파트너 관계 수준에서 커넥터 사용

커넥터를 사용하기 전에 다음 사항을 확인하십시오.

- SiteMinder 정책 관리자가 페더레이션된 통신에 맞게 정책을 구성했는지 여부
- CA SiteMinder® Federation Standalone 에서 커넥터 관련 설정을 구성했는지 여부

SiteMinder 가 배포된 위치에서 파트너 관계에 커넥터를 사용하도록 설정하십시오.

- SiteMinder 가 어설션 당사자 측에 있는 경우, IdP-SP 또는 생산자-소비자 파트너 관계에 대해 커넥터를 사용하도록 설정합니다.
- SiteMinder 가 신뢰 당사자 측에 있는 경우 SP-IdP 또는 소비자-생산자 파트너 관계에 대해 커넥터를 사용하도록 설정합니다.

기존 파트너 관계를 수정하는지 아니면 새로운 파트너 관계를 구성하는지에 관계없이 표준 파트너 관계 구성 단계가 적용되며, 특별한 구성 절차는 없습니다. 그러나 다음 지침에 따라 신뢰 당사자에 대상 리소스를 지정하십시오.

- CA SiteMinder® Federation Standalone 이 독립 실행형 모드로 배포된 경우에는 SiteMinder 웹 에이전트가 보호하는 웹 서버에 대상 리소스가 상주합니다.
- CA SiteMinder® Federation Standalone 이 프록시 모드로 배포된 경우에는 모든 프록시 요청이 SiteMinder 로 돌아가기 때문에 CA SiteMinder® Federation Standalone 서버의 URL 이 대상 리소스입니다.

다음 단계를 수행하십시오.

1. Administrative UI 에 로그인합니다.
2. 파트너 관계를 "페더레이션된 파트너 관계" 목록에서 선택하거나 새로 생성합니다.

   "파트너 관계" 대화 상자가 열립니다.
3. 마법사의 다음 단계 중 하나로 이동합니다.
   a. 신뢰 당사자 측에서 파트너 관계 마법사의 "사용자 ID" 단계로 이동합니다.
   b. 어설션 당사자 측에서 파트너 관계 마법사의 "페더레이션 사용자" 단계로 이동합니다.

4. "SiteMinder 커넥터 사용" 확인란을 선택합니다.

   그러면 구성 필드가 활성화됩니다.

5. (선택 사항) "UserDN 및 디렉터리 이름 비교 적용" 확인란을 선택합니다. 이 확인란을 선택하면 CA SiteMinder® Federation Standalone 의 사용자 디렉터리와 SiteMinder 의 디렉터리 사이에 UserDN 및 UserDirectory 이름 항목에 대한 비교가 수행됩니다.

   이 확인란을 선택할 경우에는 CA SiteMinder® Federation Standalone 및 SiteMinder 배포의 사용자 디렉터리가 물리적으로 같은 디렉터리여야 합니다. 사용자가 저장소 조회를 수행하려면 두 디렉터리의 이름이 같아야 합니다. 이 확인란을 선택하지 않으면 사용자 레코드를 찾는 데 유니버설 ID 특성이 사용됩니다. 유니버설 ID 를 사용할 경우에는 디렉터리가 서로 달라도 됩니다. 유니버설 ID 를 사용하면 각 사용자의 유니버설 ID 가 고유해야 합니다. 유니버설 ID 가 고유하지 않으면 사용자 레코드에 액세스하는 시스템이 잘못된 레코드를 가져올 수 있습니다.

6. 변경 내용을 저장합니다.

커넥터를 사용하지 않도록 설정할 때는 파트너 관계 수준에서 설정하거나 "배포 설정"에서 전역으로 설정할 수 있습니다.

제 21 장: 페더레이션 환경 보호

이 섹션은 다음 항목을 포함하고 있습니다.

페더레이션된 통신 보호

어설션 암호화 및 파트너 사이트 간에 SSL 연결 사용 등과 같이 페더레이션된 파트너 간의 트랜잭션에 보안을 적용하는 데 도움이 되는 몇 가지 메커니즘이 있습니다.

CA SiteMinder?Federation Standalone 을 사용하여 페더레이션 환경을 설정하는 경우 환경을 보호하기 위한 몇 가지 권장 사항은 다음과 같습니다.

- 한 번만 사용할 어설션을 생성합니다.
- 페더레이션 환경의 연결 보안을 유지합니다.
- 교차 사이트 스크립팅을 방지합니다.

이어지는 단원에서 이러한 내용을 설명합니다.

어설션의 일회 사용 적용

유효 기간이 지난 어설션을 재사용하면 오래된 아이덴티티 정보에 기반한 인증 결정이 내려집니다. 이러한 재사용을 방지하기 위해 CA SiteMinder?Federation Standalone 은 SAML 1.x 및 2.0 사양에 따라 일회용 어설션을 생성할 수 있습니다. 어설션 재사용으로 인한 문제를 방지하기 위해, 어설션에는 신뢰 당사자에게 이후 트랜잭션을 위해 어설션을 보존하지 않도록 지정하는 요소가 포함됩니다.

CA SiteMinder?Federation Standalone 이 어설션 당사자(생산자/IdP)로 작동하는 경우 어설션의 일회 사용을 구성할 수 있습니다. SAML 1.x 생산자의 경우 **DoNotCache 조건 설정** 설정을 선택할 수 있습니다. SAML 2.0 IdP 의 경우 **OneTimeUse 조건 설정** 설정을 선택할 수 있습니다. 이러한 구성 설정을 둘 다 사용하면 CA SiteMinder?Federation Standalone 이 일회 사용 조건을 나타내는 적절한 요소를 어설션에 삽입할 수 있습니다.

참고: 어설션의 일회 사용과 SAML 1.x 및 2.0 HTTP-POST 싱글 사인온에 대한 단일 사용 정책을 혼동하지 마십시오. CA SiteMinder?Federation Standalone 은 신뢰 당사자로 작동할 때 단일 사용 정책을 사용하며 이는 POST 트랜잭션 전용입니다. 일회 사용 기능은 HTTP-아티팩트 및 HTTP-POST 용입니다.

페더레이션 환경의 연결 보안

페더레이션된 파트너 간에 전송되거나 파트너와 응용 프로그램 간에 전송되는 아이덴티티 정보는 통신이 보안 연결을 통해 수행될 때 최적으로 보호됩니다.

신뢰 당사자와 대상 응용 프로그램 간의 연결 보안

신뢰 당사자 측에서 클라이언트 사이트의 대상 응용 프로그램으로 데이터를 안전하게 전송하는 것이 중요합니다. 보안 연결을 통신 채널로 사용하면 보안 공격에 대한 환경 취약점이 줄어듭니다.

예를 들어 신뢰 당사자가 추출하여 클라이언트 응용 프로그램에 보내는 특성이 어설션에 포함될 수 있습니다. 신뢰 당사자는 HTTP 헤더 변수나 쿠키를 사용하여 이러한 특성을 응용 프로그램에 전달할 수 있습니다. 헤더나 쿠키에 저장된 특성이 클라이언트 측에서 덮어쓰여질 수 있으므로 악의적인 사용자가 다른 사용자를 가장할 수 있습니다. SSL 연결을 사용하면 환경이 이러한 종류의 보안 위반으로부터 보호됩니다.

Administrative UI 의 "배포 설정"에서 "보안 쿠키 사용" 확인란을 설정하여 이와 같은 취약성으로부터 보호할 수 있습니다. "보안 쿠키 사용" 설정을 지정하면 CA SiteMinder?Federation Standalone 이 "secure" 플래그가 표시된 쿠키를 생성합니다. 이 플래그는 CA SiteMinder?Federation Standalone 이 SSL 통신 채널을 통해서만 쿠키를 전송함을 나타냅니다.

CA SiteMinder?Federation Standalone 어설션 당사자의 초기 인증 보안

CA SiteMinder?Federation Standalone 어설션 당사자의 초기 사용자 인증에는 잠재적인 취약점이 있습니다. 사용자가 어설션 당사자 측에서 사용자 세션을 설정하기 위해 맨 처음 인증할 때 세션 ID 쿠키가 브라우저에 기록됩니다. 쿠키가 비 SSL 연결을 통해 전송되면 공격자가 쿠키를 획득하고 중요한 사용자 정보를 빼내어 사용자를 가장하거나 아이덴티티를 도용할 수 있습니다.

Administrative UI 의 "배포 설정"에서 "보안 쿠키 사용" 확인란을 설정하여 이와 같은 취약성으로부터 보호할 수 있습니다. "보안 쿠키 사용" 설정을 지정하면 CA SiteMinder?Federation Standalone 이 "secure" 플래그가 표시된 쿠키를 생성합니다. 이 플래그는 브라우저가 SSL 연결을 통해서만 쿠키를 전달하여 보안이 강화됨을 나타냅니다. 일반적으로 모든 URL 에 대해 SSL 연결을 설정하는 것이 좋습니다.

페더레이션된 네트워크를 교차 사이트 스트립팅으로부터 보호

응용 프로그램이 일반적으로 post 데이터 또는 URL 의 쿼리 매개 변수 데이터와 같이 브라우저에 표시될 때 스크립트 공격이 발생할 수 있는 입력 텍스트를 필터링 없이 브라우저에 표시할 경우 CSS(교차 사이트 스크립팅) 공격이 발생할 수 있습니다. 이러한 문자가 표시되면 원치 않는 스크립트가 브라우저에서 실행될 수 있습니다.

CA SiteMinder?Federation Standalone 은 페더레이션 기능에 사용할 여러 JSP 를 제공합니다. 이러한 JSP 는 요청에 있는 문자를 확인하여 출력 스트림의 안전하지 않은 정보가 브라우저에 표시되지 않도록 합니다.

CA SiteMinder?Federation Standalone 이 요청을 받으면 다음 JSP 가 디코딩된 값을 검사하여 교차 사이트 스크립팅 문자가 있는지 확인합니다.

- idpdiscovery.jsp

  신뢰 당사자 측에서 아이덴티티 공급자 검색에 사용합니다.

- linkaccount.jsp

  신뢰 당사자 측에서 동적 계정 연결에 사용합니다.

- sample_application.jsp

  IDP 측에서 싱글 사인온을 시작하는 데 사용합니다. 이 샘플 응용 프로그램을 사용하여 사용자를 먼저 SSO 서비스에 연결한 다음 사용자 지정 웹 응용 프로그램에 연결할 수 있습니다. 일반적으로 사용자 고유의 응용 프로그램을 사용합니다.

- signoutconfirmurl.jsp

  계정 파트너에서 WS-페더레이션 사인아웃에 사용합니다.

- unsolicited_application.jsp

  사용자가 SSO 서비스에 먼저 전송되지 않고 웹 응용 프로그램에 직접 전송될 때 IdP 에서 시작되는 싱글 사인온에 사용됩니다.

해당 페이지에서는 요청을 검사하여 다음 문자가 있는지 확인합니다.

문자	설명
<	왼쪽 꺾쇠 괄호
>	오른쪽 꺾쇠 괄호
‘	작은따옴표
"	큰따옴표
%	백분율 기호
;	세미콜론
(	여는(왼쪽) 괄호
)	닫는(오른쪽) 괄호
&	앰퍼샌드
+	더하기 기호

CA SiteMinder?Federation Standalone 에서 제공하는 JSP 각각에는 검사할 문자를 정의하는 변수가 포함되어 있습니다. 문자 집합을 확장하려면 이러한 JSP 를 수정하십시오.

제 22 장: 신뢰 당사자에서의 응용 프로그램 통합

이 섹션은 다음 항목을 포함하고 있습니다.

신뢰 당사자와 응용 프로그램의 상호 작용

파트너 관계 마법사의 "응용 프로그램 통합" 단계는 신뢰 당사자 측에서만 적용됩니다. 이 단계에서는 사용자 신원을 확인하고 사용자를 대상 응용 프로그램으로 연결하기 위한 페더레이션된 작업의 다양한 측면을 정의할 수 있습니다.

"응용 프로그램 통합" 단계에서 구성할 수 있는 기능은 다음과 같습니다.

- 사용자를 대상 응용 프로그램으로 리디렉션
- 어설션 특성을 응용 프로그램 특성에 매핑(SAML 만 해당)
- 사용자 아이덴티티 프로비저닝
- 인증 실패 시 사용자 리디렉션

사용자를 대상 응용 프로그램으로 리디렉션

"응용 프로그램 통합" 단계의 "대상 응용 프로그램" 그룹 상자에서는 사용자를 CA SiteMinder?Federation Standalone 에서 대상 응용 프로그램으로 리디렉션하는 방법을 정의할 수 있습니다. 몇 가지 방법 중에서 선택할 수 있습니다. 선택하는 리디렉션 방법은 사용자와 함께 대상 응용 프로그램에 전달하고자 하는 데이터의 유형에 따라 달라집니다.

다음 단계를 수행하십시오.

1. 파트너 관계 마법사의 "응용 프로그램 통합" 단계로 이동합니다.
2. "리디렉션 모드" 필드에서 리디렉션 방법을 선택합니다.
   - "쿠키 데이터"를 선택하면 "URL 인코드 특성 쿠키 데이터" 확인란을 선택하여 쿠키의 특성 데이터를 URL 인코딩할 수 있습니다.
   - "개방 형식 쿠키" 또는 "개방 형식 쿠키 게시" 옵션을 선택하는 경우 추가 필수 설정 및 옵션 설정을 구성하십시오. 개방 형식 쿠키와는 달리, 개방 형식 쿠키 게시는 HTTP-POST 요청 형식으로 데이터를 보냅니다.

     신뢰 당사자가 여러 특성 값이 있는 어설션을 받는 경우 페더레이션 시스템은 모든 값을 대상 응용 프로그램에 쿠키로 전달합니다.
   - FIPS 호환 알고리즘 중 하나(AES 알고리즘)를 선택하는 경우 CA SiteMinder® Federation Standalone SDK 를 사용하여 개방 형식 쿠키를 사용해야 합니다. .NET SDK 를 사용하는 경우에는 AES128/CBC/PKCS5Padding 암호화 알고리즘을 사용해야 합니다.
   - CA SiteMinder® Federation Standalone 을 프록시 모드로 구성하고 "HTTP 헤더"를 리디렉션 모드로 선택하면 CA SiteMinder® Federation Standalone 은 여러 특성 값을 쉼표로 구분하여 단일 헤더에 전달할 수 있습니다.

   참고: "도움말"을 클릭하면 필드, 컨트롤 및 해당되는 각 요구 사항에 대한 설명을 볼 수 있습니다.

3. "대상" 필드에 대상 응용 프로그램의 URL 을 입력합니다.

   CA SiteMinder® Federation Standalone 이 프록시 모드에서 작동하는 경우 프록시가 모든 페더레이션 요청을 로컬로 처리하기 때문에 프록시 호스트의 URL 을 입력하십시오. 프록시 호스트는 CA SiteMinder® Federation Standalone 앞에 있는 모든 시스템이 될 수 있습니다. 인터넷에서 직접 CA SiteMinder® Federation Standalone 에 액세스하는 경우에는 CA SiteMinder® Federation Standalone 자체가 프록시 호스트가 될 수도 있습니다. 프록시 모드에서 작동할 때는 대상으로 지정하는 URL 이 CA SiteMinder® Federation Standalone 을 통과해야 합니다. 예를 들어, CA SiteMinder® Federation Standalone 의 기준 URL 이 fed.demo.com:5555 이고 백엔드 서버 리소스가 mytarget/target.jsp 인 경우 "대상" 필드의 값은 http://fed.demo.com:5555/mytarget/target.jsp 입니다.

   참고: 프록시 호스트 뒤에 상주하는 백엔드 서버는 CA SiteMinder® Federation Standalone 구성 마법사를 실행할 때 지정합니다. 필요할 경우 구성 마법사를 다시 실행하여 백엔드 서버 항목을 수정할 수 있습니다.

   SAML 2.0 의 경우, 싱글 사인온을 트리거하기 위한 "릴레이 상태" 쿼리 매개 변수 값을 URL 에 포함하여 이 필드를 덮어쓰면 이 필드를 비워둘 수 있습니다. 이렇게 설정하려면 "릴레이 상태가 대상 무시" 확인란을 선택하십시오.

대상으로의 리디렉션이 설정되었습니다.

HTTP 헤더를 사용하여 어설션 데이터 전달(SAML 만 해당)

SAML 엔터티의 경우 정책 서버는 HTTP 헤더를 사용하여 어설션에서 백엔드 응용 프로그램으로 아이덴티티 특성을 전달할 수 있습니다. 백엔드 응용 프로그램은 싱글 사인온의 대상 응용 프로그램 또는 사용자 프로비저닝 응용 프로그램일 수 있습니다. 이 헤더는 암호화된 쿠키에 포함되어 전달됩니다.

헤더의 이름은 어설션 특성과 동일합니다. 예를 들어 어설션 특성이 "address"인 경우 응용 프로그램은 "ADDRESS"라는 HTTP 헤더를 찾습니다.

어설션 특성은 대/소문자를 구분하지만 HTTP 헤더는 그렇지 않습니다. 정책 서버는 대/소문자만 다른 동일한 특성을 전달한 다음 이를 HTTP 헤더에 매핑할 수 없습니다. 예를 들어 "address"와 "Address"가 동시에 헤더로 전달될 수 없습니다. 일반적으로 대/소문자나 형식만 다르고 이름이 동일한 특성을 사용하지 마십시오.

다음의 추가적인 값이 헤더로 전달됩니다.

- NAMEID
- FORMAT
- AUTHNCONTEXT

HTTP 헤더 보호

인증되지 않은 사용자가 어설션 특성의 이름을 알고 있는 경우 이 사용자는 이 이름을 브라우저에서 헤더로 설정할 수 있습니다. 헤더가 설정되면 악의적인 사용자가 대상 응용 프로그램에 대한 액세스 권한을 획득할 수 있습니다. 대상 응용 프로그램은 예기치 않은 헤더 값을 발견하고 SiteMinder 의 어설션 소비 없이 리소스에 대한 액세스를 허용합니다.

FedHeaderPrefix 에 대한 값을 설정하면 다음과 같은 시나리오가 방지됩니다.

1. 인증되지 않은 사용자가 HTTP 헤더의 이름을 알아냅니다. 이 헤더 이름에는 접두사가 포함되어 있습니다.
2. 악의적인 사용자가 헤더를 포함하여 들어오는 요청을 정책 서버에 전송합니다.
3. 정책 서버는 접두사를 포함한 헤더의 출처가 들어오는 요청이고 이 헤더가 내부에서 생성되지 않은 것으로 인식하므로 이를 제거합니다.
4. 시스템에서는 내부에서 생성한 정상 헤더를 백엔드 응용 프로그램에 전달하기 전에 각 헤더에 지정된 접두사를 추가합니다. 그런 다음 이 헤더가 응용 프로그램에 전달됩니다.

어설션 데이터를 전달하도록 HTTP 헤더 구성(SAML 만 해당)

SiteMinder 는 HTTP 헤더를 사용하여 어설션 데이터를 전달할 수 있습니다.

다음 단계를 수행하십시오.

1. 페더레이션 트래픽을 처리하는 신뢰 당사자 시스템에 SiteMinder 웹 에이전트가 설치되어 있는지 확인합니다.

2. (선택 사항이지만 권장됨) HTTP 헤더에 대한 접두사로서 임의 문자를 입력합니다.

   SiteMinder 는 이 접두사를 모든 HTTP 헤더에 추가합니다. 접두사를 설정하면 SiteMinder 가 어설션을 소비하기 전에 인증되지 않은 사용자가 HTTP 헤더를 조작하지 못하게 됩니다. 따라서 정상 헤더만 대상 응용 프로그램으로 전달됩니다. HTTP 헤더 보호 (페이지 349)에 대한 자세한 내용을 참조하십시오.

   HTTP 헤더에 접두사를 추가하려면 다음을 수행하십시오.

   a. Administrative UI 에 로그인합니다.

   b. "인프라", "배포 설정"을 클릭합니다.

   c. "HTTP 헤더 접두사"에 대한 문자열을 지정합니다.

      참고: 이 옵션은 프록시 배포 모드의 경우에만 사용할 수 있습니다.

   d. "저장"을 클릭합니다.

3. 파트너 관계 마법사의 "응용 프로그램 통합" 단계에서 다음 태스크 중 *하나*를 수행합니다.

   - "HTTP 헤더"를 대상 응용 프로그램에 대한 "리디렉션 모드"로 선택합니다.
   - "HTTP 헤더"를 사용자 프로비저닝에 대한 "전송 옵션"으로 선택합니다.

이제 HTTP 헤더가 특성 데이터를 전달하도록 구성되었습니다.

어설션 특성을 응용 프로그램 특성에 매핑(SAML 만 해당)

신뢰 당사자 측에서 CA SiteMinder?Federation Standalone 은 어설션 특성 집합을 나가는 응용 프로그램 특성 집합에 매핑할 수 있게 해 줍니다. 그런 다음 CA SiteMinder?Federation Standalone 은 응용 프로그램 특성을 대상 응용 프로그램에 전송합니다. 특성 매핑을 사용하면 대상 응용 프로그램을 수정할 필요 없이 사용자에게 사용자 지정된 환경을 제공할 수 있습니다. 특성은 파트너 관계 단위로 매핑되므로 신뢰 당사자 측 응용 프로그램을 여러 어설션 당사자에 사용할 수 있습니다.

CA SiteMinder?Federation Standalone 은 다음 유형의 매핑을 수행할 수 있습니다.

- 어설션 특성 이름을 응용 프로그램 특성 이름으로 변환합니다.

  예

  들어오는 어설션 특성이 Region=US 일 수 있습니다. 특성은 나가는 응용 프로그램 특성 ServiceLocation=US 로 변환될 수 있습니다.

- 개별적인 특성과 해당 값을 단일 특성으로 전환합니다.

  예

  어설션에 Name=Bob 및 LastName=Smith 의 두 특성이 포함되어 있습니다. 이 두 특성을 FullName =Bob Smith 로 변환할 수 있습니다.

응용 프로그램 특성 정의 테이블 사용

"응용 프로그램 통합" 대화 상자의 "응용 프로그램 특성 정의" 테이블에서 특성 매핑 규칙을 정의할 수 있습니다.

"응용 프로그램 특성" 및 "어설션 특성" 열은 원격 생산자 또는 IdP 엔터티에 대해 지정된 어설션 특성을 기준으로 채워집니다. 이러한 특성은 이 로컬 신뢰 당사자 측에서 구성하십시오. "어설션 특성" 열에 대해서는 어설션 특성 이름이 입력됩니다. 이에 해당하는 UEL(Unified Expression Language) 문자열이 "어설션 특성" 열에 입력됩니다.

신뢰 당사자의 관리자 또는 응용 프로그램 통합자는 특성 매핑을 구성하기 위해 다음 정보를 알아야 합니다.

- 대상 응용 프로그램 특성의 이름
- 어설션의 특성 이름
- 어설션 특성과 대상 응용 프로그램 특성 간의 매핑 관계. 매핑 관계를 안다는 것은 사용 가능한 어설션 특성을 필요한 응용 프로그램 특성으로 전환하는 방법을 안다는 것을 의미합니다.

특성 매핑을 설정하기 전에 필요한 당사자 측에서 응용 프로그램 및 어설션 특성의 이름을 수집하십시오.

응용 프로그램 특성은 대상 응용 프로그램이 사용하는 특성을 반영해야 하므로, 응용 프로그램에 맞게 기본값을 수정해야 합니다. 응용 프로그램 관리자와의 대역 외 통신을 통해 응용 프로그램 특성을 가져올 수 있습니다.

식 작성기를 사용하여 매핑 규칙 작성

UI 에서는 매핑 규칙을 작성하는 데 유용한 식 작성기를 제공합니다. 식 작성기에 액세스하려면 "어설션 특성" 필드 오른쪽의 슬라이더 단추(<<)를 선택하십시오. 슬라이더 단추를 선택하면 빈 필드와 풀다운 화살표가 표시됩니다. 화살표를 선택하면 매핑 구성에 사용할 수 있는 어설션 특성과 특수 문자의 목록이 표시됩니다. 식 작성기를 숨기려면 슬라이더 단추(>>)를 클릭하십시오.

식 작성기의 "어설션 특성" 목록은 이 로컬 신뢰 당사자 측에서 구성하는 원격 생산자 또는 IdP 엔터티에 대해 지정한 어설션 특성을 기준으로 미리 채워집니다. 특성이 어설션에 있다는 점을 알고 있으면 항목을 수동으로 지정할 수 있습니다. 식 작성기 메뉴의 옵션만 사용할 필요는 없습니다.

"특수 문자" 목록에는 매핑 규칙을 작성하는 데 사용할 수 있는 쉼표와 백분율 기호 등의 문자가 포함되어 있습니다. 목록에서 문자를 선택하거나 문자를 수동으로 입력할 수 있습니다.

중요! 이 테이블에 어설션 특성을 입력할 때 어설션 특성은 원격 어설션 당사자 측에서 지정된 어설션 특성을 기준으로 대/소문자가 구분됩니다. 대/소문자가 일치해야 합니다. CA SiteMinder® Federation Standalone 이 파트너 관계의 양쪽 모두에 있는 경우 특성은 원격 IdP 파트너 관계 마법사의 NameID 및 특성 단계에서 지정됩니다. 파트너와의 대역 외 통신에서 또는 메타데이터 가져오기를 통해 어설션 특성을 가져오십시오.

매핑 규칙이 정의되면 CA SiteMinder?Federation Standalone 은 데이터를 레거시 쿠키, 개방 형식 쿠키 또는 HTTP 헤더에 넣어 응용 프로그램에 전송합니다. "응용 프로그램 통합" 대화 상자의 "대상 응용 프로그램" 섹션에서 전송 방법을 지정하십시오.

매핑 수정 및 삭제

언제든지 "응용 프로그램 특성 정의" 테이블에서 특성 매핑을 변경하거나 제거할 수 있습니다.

매핑을 수정하려면

1. 커서를 수정할 행의 필드에 놓은 다음 새 텍스트를 입력합니다. 식 작성기를 사용하여 현재 식의 끝에 값을 더 추가할 수도 있습니다.
2. "다음"을 클릭하고 마법사의 마지막 단계로 진행하여 변경 내용을 저장합니다.

매핑을 삭제하려면

1. 제거할 항목의 "삭제" 열에서 휴지통을 클릭합니다.
2. "다음"을 클릭하고 마법사의 마지막 단계로 진행하여 변경 내용을 저장합니다.

적절한 구문을 사용하여 특성 매핑 규칙 작성

특성 매핑은 어설션 특성을 응용 프로그램 특성으로 전환하는 매핑 규칙을 사용합니다. 특성 매핑을 사용하면 CA SiteMinder?Federation Standalone 이 기본 매핑 규칙을 생성합니다. 규칙은 원격 생산자 또는 IdP 엔터티에 대해 지정된 어설션 특성을 기반으로 합니다. 이러한 구성 태스크는 모두 로컬 신뢰 당사자 측에서 수행됩니다. 특성 매핑을 사용하지 않으면 어설션 특성은 대상 응용 프로그램에 "있는 그대로" 전달됩니다.

CA SiteMinder?Federation Standalone 은 JSP 및 JSF 와 비슷한 UEL(Unified Expression Language) 구문을 매핑에 사용합니다. 각 어설션 특성이 hashmap 에 들어가고 **attr** 키워드가 할당됩니다. UEL 식 계산기가 매핑 규칙 목록을 순환하여 어설션 특성의 hashmap 에 이를 적용합니다. 그런 후 식 계산기는 결과 응용 프로그램 특성이 포함된 다른 hashmap 을 생성합니다. 나가는 응용 프로그램 특성의 hashmap 은 쿠키 콘텐츠 또는 헤더 변수로 변환되어 대상 응용 프로그램으로 전송됩니다.

UEL 에 대한 자세한 내용은 Sun Developer Network http://developers.sun.com/를 참조하십시오.

식을 작성하려면 CA SiteMinder?Federation Standalone 에서 식에 사용되는 구문을 이해하는 것이 중요합니다.

단일 특성 표현

단일 어설션 특성을 표현하려면 다음 구문을 사용하십시오.

#{attr["*attribute_name*"]}

예: #{attr["Name"]}는 "이름" 어설션 특성의 값을 나타냅니다.

복합 특성 표현

값 식을 연결하여 복합 값을 구성할 수 있습니다(선택적 구분 기호 사용). 복합 어설션 특성을 표현하려면 다음 구문을 사용하십시오.

#{attr["*first_attribute*"]}*optional_character* #{attr["*second_attribute*"]}

매핑 예

다음은 매핑 규칙의 예입니다. 이러한 예는 다음 형식으로 표시됩니다.

application_attribute=assertion_attributes_expression

이름 예

구문

ID = #{attr["Name"]}

샘플 결과

BobSmith

단순 연결 예

구문

FullName = #{attr["FirstName"]},#{attr["LastName"]}

샘플 결과

Bob,Smith

구문

FullName = #{attr["LastName"]},#{attr["FirstName"]}

샘플 결과

Smith,Bob

공백은 특수 문자로 간주됩니다. 식에서 특성 사이에 공백을 넣으려면 공백을 입력하십시오. 예를 들면 다음과 같습니다.

구문

FullName = #{attr["LastName"]}, #{attr["FirstName"]}

샘플 결과

Smith, Bob

날짜 예

구문

Date = #{attr["month"]}/#{attr["dateOfMonth"]}/#{attr["year"]}

샘플 결과

01/05/2010

구문

Date = #{attr["monthSymbol"]} #{attr["dateOfMonth"]}, #{attr["year"]}

샘플 결과

January 5, 2010

통화 예

구문

Price = #{attr["amount"]}#{attr["currency"]}

샘플 결과

2.50EUR

전자 메일 주소 예

구문

EmailAddress = #{attr["userName"]}@#{attr["domainName"]}

샘플 결과

JaneDoe@company.com

구문

AcmeEmailAddress = #{attr["AcmeIDKey"]}@acme.com

샘플 결과

bsmith@acme.com

신뢰 당사자 측에서 특성 매핑 구성

CA SiteMinder?Federation Standalone 이 어설션 특성에 적용할 수 있는 매핑 규칙 집합을 정의하십시오. CA SiteMinder?Federation Standalone 에서는 특정한 어설션 특성 또는 몇 가지 특성의 조합을 매핑할 수 있습니다. 매핑의 결과는 단일 응용 프로그램 특성이거나 여러 특성일 수 있습니다.

특성 매핑을 구성하려면

1. 파트너 관계 마법사의 "응용 프로그램 통합" 단계로 이동합니다.
2. "응용 프로그램 특성에 매핑" 섹션에서 "특성 매핑 사용" 확인란을 선택합니다.

   "응용 프로그램 특성 정의" 테이블이 표시됩니다.
3. 기존 응용 프로그램 특성을 수정하거나 테이블에서 새로 정의합니다. 모든 응용 프로그램 특성이 대상 응용 프로그램으로 전송됩니다.

   "어설션 특성" 열의 값 구문은 UEL(Unified Expression Language)을 준수해야 합니다.

   슬라이더 단추(<<)를 사용하여 식 작성기를 열고 사용 가능한 옵션을 표시합니다. 목록의 항목을 특성 값에 추가하려면 어설션 또는 특수 문자를 선택하고 "추가"를 클릭합니다.

   참고: "응용 프로그램 특성 테이블"에서 "쿠키 데이터" 및 특수 문자를 지정한 경우에는 "URL 인코드 특성 쿠키 데이터" 옵션을 선택하십시오. 확인란은 대화 상자의 "대상 응용 프로그램" 섹션에 있습니다. 특수 문자는 드롭다운 목록에서 추가하거나 직접 입력할 수 있습니다. 또한 대상 응용 프로그램은 수신되는 응용 프로그램 특성의 이름과 값을 URL 디코딩해야 합니다.
4. (선택 사항) 기본 매핑으로 부족한 경우 원하는 수의 행을 추가합니다.

   기본적으로 원격 생산자 또는 IdP 엔터티에서 정의된 모든 어설션 특성이 기본 매핑으로 테이블에 포함됩니다. 원래의 어설션 특성은 변경되지 않습니다. 이 매핑은 수정할 수 있습니다.
5. 응용 프로그램 특성이 대상 응용 프로그램으로 전송되는 방법을 구성합니다. "응용 프로그램 통합" 대화 상자의 "대상 응용 프로그램" 섹션에서 방법을 구성합니다.

특성 매핑 구성이 완료되었습니다.

추가 정보:

적절한 구문을 사용하여 특성 매핑 규칙 작성 (페이지 354)

신뢰 당사자 측에서 사용자 아이덴티티의 동적 프로비저닝

페더레이션된 네트워크에서는 신뢰 당사자가 여러 어설션 당사자 측에서 페더레이션된 사용자의 계정을 설정하는 경우가 많습니다. 동적 프로비저닝은 데이터 및 응용 프로그램에 액세스하기 위해 필요한 계정 권한 및 액세스 권한을 가진 클라이언트 계정을 생성하는 프로세스를 CA SiteMinder?Federation Standalone 에서 수행할 수 있도록 지원합니다.

CA SiteMinder?Federation Standalone 에서는 다음과 같은 두 가지 방법으로 프로비저닝을 지원합니다.

- 로컬 계정 연결(SAML 2.0 에만 해당)
- 원격 프로비저닝

각 단계는 다음 섹션에서 설명합니다.

프로비저닝을 위한 로컬 계정 연결

로컬 계정을 연결하여 프로비저닝을 구현하는 방법은 SAML 2.0 배포 환경에서만 사용할 수 있습니다. IdP 에 있는 사용자 계정을 SP 에 있는 계정에 연결하여 프로비저닝이 이루어집니다.

로컬 계정 연결 프로세스는 다음과 같습니다.

1. 사용자가 SP 에 있는 페더레이션된 대상 리소스에 대한 액세스를 요청합니다.
2. SP 는 이름이 AllowCreate 이고 값이 true 로 설정된 특성이 포함된 AuthnRequest 를 생성합니다. SP 는 AuthnRequest 를 IdP 로 전송하여 사용자의 아이덴티티를 가져옵니다.
3. AuthnRequest 를 받으면 IdP 가 어설션을 생성합니다. 어설션 생성 중에 IdP 는 어설션에서 이름 ID 로 사용하도록 구성된 특성에 해당하는 사용자 레코드를 검색합니다.
4. NameID 로 사용되는 특성의 값을 찾지 못할 경우, IdP 는 영구 식별자를 생성합니다(식별자를 생성하는 기능을 사용 중인 경우에만 해당).

영구 식별자는 무작위로 생성된 ID 입니다. IdP 는 이 식별자를 NameID 특성의 값으로 사용하고 어설션에 넣습니다. 그런 다음 IdP 는 어설션을 SP 에 다시 보냅니다.

참고: 양쪽 사이트 모두에 "허용/만들기" 기능이 구성되어 있어야 합니다. 식별자를 생성하도록 IdP 가 구성되어 있지 않으면 AuthnRequest 메시지에 "허용/만들기" 특성이 포함되었는지에 관계없이 어설션 생성이 실패합니다.

5. SP 에 있는 CA SiteMinder® Federation Standalone 이 IdP 에서 받은 어설션을 처리하지만 NameID 값이 존재하지 않기 때문에 SP 에서 사용자 레코드를 검색할 수 없습니다. 그 결과 인증이 실패합니다.
6. 인증 시도가 실패하면 모든 어설션 및 어설션 소비자 서비스가 전달한 기타 데이터가 들어 있는 linkaccount.jsp 페이지로의 리디렉션이 트리거됩니다.
7. 사용자는 로컬 자격 증명을 사용하여 인증되어야만 linkaccount.jsp 페이지에 액세스할 수 있습니다. 로그인이 성공하면 사용자가 식별됩니다. CA SiteMinder® Federation Standalone 은 로컬 사용자 디렉터리에서 적절한 사용자 레코드를 참조하여 해당 사용자의 세션을 생성합니다. 이 시점에서 사용자는 원래 요청되었던 페더레이션된 리소스에는 아직 액세스하지 못합니다.
8. linkaccount.jsp 는 어설션 및 다른 모든 데이터를 어설션 소비자 서비스에 다시 전달합니다. CA SiteMinder® Federation Standalone 이 어설션을 사용하여 사용자 인증을 다시 시도합니다. 이제는 사용자를 식별하는 세션이 있기 때문에 CA SiteMinder® Federation Standalone 은 어설션에 포함된 영구 식별자를 사용하여 로컬 사용자 디렉터리의 적절한 사용자 레코드를 채웁니다. CA SiteMinder® Federation Standalone 은 이 용도로만 구성할 수 있는 특성에 영구 식별자를 저장합니다. 이제 IdP 에 있는 계정이 SP 에 있는 계정과 연결되었습니다. 인증이 성공합니다.
9. 마지막으로 CA SiteMinder® Federation Standalone 이 사용자를 요청된 리소스로 리디렉션합니다.

참고: 로컬 계정 연결 기능을 사용하려면 **IDP 가 사용자 식별자를 만들도록 허용** 기능을 사용해야 하지만 AllowCreate 기능은 로컬 계정 연결에만 사용되는 것은 아닙니다. 이 기능은 로컬 계정 연결을 구현하는 경우가 아니더라도 선택할 수 있습니다.

로컬 계정 연결 구성(SAML 2.0)

로컬 계정 연결 방법으로 프로비저닝을 구현하려면 아이덴티티 공급자와 서비스 공급자에서 구성 단계를 수행해야 합니다.

아이덴티티 공급자에서 로컬 계정 연결을 구성하려면

1. 파트너 관계 마법사를 열어 파트너 관계 마법사의 "어설션 구성" 단계로 이동합니다.
2. "이름 ID" 그룹 상자의 필수 필드를 구성합니다.

   이러한 필드에서 어설션의 NameID 에 사용되는 특성을 결정합니다.

   참고: "도움말"을 클릭하면 필드, 컨트롤 및 해당되는 각 요구 사항에 대한 설명을 볼 수 있습니다.
3. **사용자 식별자의 생성 허용** 확인란을 선택합니다.
4. 파트너 관계 마법사에서 "확인" 단계를 선택하고 "마침"을 클릭하여 변경 내용을 저장합니다.

아이덴티티 공급자에서 구성 단계가 완료되었습니다.

서비스 공급자에서 로컬 계정 연결을 구성하려면

1. 파트너 관계 마법사를 열어 "사용자 ID" 단계로 이동합니다.
2. "어설션에서 아이덴티티 특성 선택" 그룹 상자에서 다음을 수행합니다.
   - 어설션에서 식별에 사용되는 특성으로 "NameID"를 선택합니다.
   - "IDP 가 사용자 식별자를 만들도록 허용"을 선택합니다.
3. "검색 사양" 필드에 값을 입력합니다.

   "검색 사양" 값은 CA SiteMinder® Federation Standalone 이 사용자를 조회하고 IdP 에서 받은 영구 식별자를 저장하는 데 사용되는 특성입니다. 예를 들어 buyerID 에 NameID 값을 저장해야 할 경우 문자열을 buyerID=%s 와 같이 설정하십시오.
4. "응용 프로그램 통합" 단계로 이동합니다.
5. 대화 상자의 "사용자 프로비저닝" 섹션에 있는 "프로비저닝 유형" 필드에서 "로컬 계정 연결"을 선택합니다.

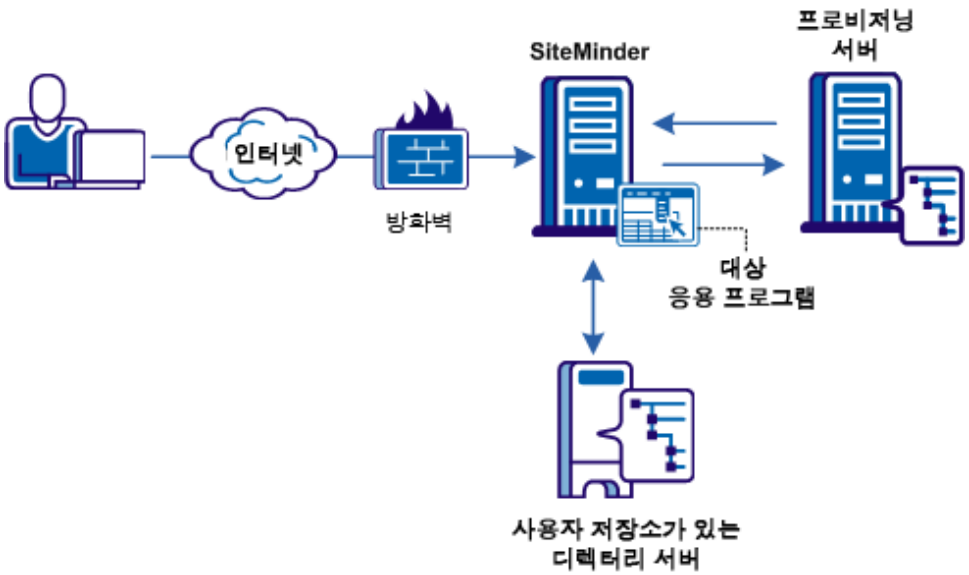
   이 옵션을 선택하면 "사용자 찾지 못함 URL"이 POST 방식의 linkaccount.jsp 페이지로 자동 구성됩니다. 이 URL 은 첫 번째 인증 시도가 실패했을 때 CA SiteMinder® Federation Standalone 이 사용자를 리디렉션하는 위치입니다.

6. (선택 사항) 실패한 인증 시도 후 사용자가 리디렉션될 때 사용자 지정 사용자 환경을 제공하도록 linkaccount.jsp 파일을 사용자 지정합니다. 이 파일에서 accountlinking 및 samlresponse 매개 변수를 어설션 소비자 서비스에 다시 포스트해야 합니다. accountlinking 매개 변수는 yes 로 설정해야 합니다. 이 페이지는 *federation_install_dir*/secure-proxy/Tomcat/webapps/affwebservices/public 에 있습니다.
7. 파트너 관계 마법사에서 "확인" 단계를 선택하고 "마침"을 클릭하여 변경 내용을 저장합니다.

원격 프로비저닝

원격 프로비저닝은 타사 프로비저닝 응용 프로그램을 사용하여 새 사용자 계정을 만듭니다. 그런 다음 응용 프로그램이 필요한 정보를 CA SiteMinder?Federation Standalone 시스템에 다시 전달합니다. 페더레이션 시스템은 해당 데이터를 사용하여 사용자 자격 증명을 만듭니다.

다음 그림에서는 원격 프로비저닝 설정을 구성하는 방법을 보여 줍니다.



상위 수준 프로비저닝 프로세스는 다음과 같습니다.

1. 신뢰 당사자의 정책 서버는 어설션과 함께 리소스에 대한 요청을 수신합니다. 하지만 사용자 디렉터리에서 사용자를 찾을 수 없습니다.
2. 프로비저닝이 사용되는 경우 정책 서버는 어설션 데이터가 포함된 활성 응답을 처리하고 어설션 데이터가 포함된 쿠키를 생성합니다. 또한 상태를 유지하는 쿠키가 생성되어 프로비저닝 요청이 수행되었음을 나타냅니다.
3. 브라우저는 개방 형식 쿠키 또는 헤더와 함께 프로비저닝 응용 프로그램으로 리디렉션됩니다.
4. 일반적으로 프로비저닝 응용 프로그램은 사용자에게 로그인하라는 메시지를 표시합니다. 사용자가 로그인하면 응용 프로그램은 쿠키나 헤더를 읽습니다. 응용 프로그램은 어설션 데이터와 로그인 자격 증명을 사용하여 사용자 계정을 설정합니다.

   프로비저닝 응용 프로그램은 CA SiteMinder® Federation Standalone Java 또는 .NET SDK 를 사용하여 개방 형식 쿠키를 소비할 수 있습니다.
5. 계정이 프로비저닝되면 브라우저는 사용자를 신뢰 당사자의 어설션 소비자 서비스로 다시 리디렉션합니다. 프로비저닝에 대한 상태 정보를 유지 관리하는 쿠키가 검사되어 사용자가 프로비저닝되었음이 확인됩니다. 자격 증명이 생성되어 인증 체계로 전달됩니다.

   참고: 프로비저닝 응용 프로그램은 신뢰 당사자에 있는 어설션 소비자 서비스의 URI 를 알아야 합니다. 예를 들어 신뢰 당사자 SiteMinder 의 SAML 2.0 URI 는 https://*sp_server:port*/affwebservices/public/saml2assertionconsumer 입니다.
6. 정책 서버는 사용자 명확성 확인 과정을 두 번째 시도합니다. 프로비저닝에 성공하면 사용자가 인증되고 쿠키 또는 헤더가 대상 응용 프로그램으로 전송됩니다.

   대상 응용 프로그램에 대해 선택한 리디렉션 모드에 따라 대상 응용 프로그램으로의 데이터 전달 방법이 결정됩니다.
7. 사용자가 대상 리소스로 리디렉션됩니다.

프로비저닝 응용 프로그램으로 어설션 데이터 전송

원격 프로비저닝을 수행하기 위해 페더레이션 시스템은 브라우저를 어설션 데이터와 함께 프로비저닝 응용 프로그램으로 리디렉션합니다.

페더레이션 시스템은 다음 방법 중 하나를 사용하여 어설션 데이터를 전달할 수 있습니다.

레거시 쿠키

페더레이션 시스템에서 생성한 레거시 쿠키에 SAML 어설션 정보를 전달합니다. 쿠키에는 어설션 데이터에 기반한 로그인 ID가 포함됩니다. 레거시 쿠키를 사용하는 경우에는 프로비저닝 응용 프로그램이 레거시 쿠키를 읽을 수 있도록 프로비저닝 응용 프로그램과 함께 CA SiteMinder® Federation Standalone Java SDK 를 시스템에 설치해야 합니다.

참고: 레거시 쿠키를 사용하는 경우에는 페더레이션 시스템과 원격 프로비저닝 시스템이 동일한 도메인에 있어야 합니다.

개방 형식 쿠키

SAML 어설션 정보를 개방 형식 쿠키로 전송합니다. 쿠키에는 어설션 데이터에 기반한 로그인 ID 가 포함됩니다.

참고: 개방 형식 쿠키를 사용하는 경우에는 페더레이션 시스템과 원격 프로비저닝 시스템이 동일한 도메인에 있어야 합니다.

쿠키는 다음 두 가지 방법 중 하나로 만들 수 있습니다.

- CA SiteMinder® Federation Standalone SDK 에서 쿠키를 생성합니다.

  FIPS 알고리즘 중 하나(AES 알고리즘)를 선택하는 경우 CA SiteMinder® Federation Standalone SDK 를 사용하여 쿠키를 생성해야 합니다. .NET SDK 를 사용하려는 경우에는 AES128/CBC/PKCS5Padding 암호화 알고리즘을 사용해야 합니다. 프로비저닝 응용 프로그램이 .NET 을 사용하는 경우 CA SiteMinder® Federation Standalone .NET SDK 를 프로비저닝 서버에 설치하여 개방 형식 쿠키를 읽는 데 사용할 수 있습니다.

  프로비저닝 응용 프로그램은 쿠키를 만들기 위해 사용 중인 SDK 와 동일한 언어를 사용해야 합니다. CA SiteMinder® Federation Standalone Java SDK 를 사용하는 경우 응용 프로그램은 Java 로 작성되어야 합니다. .NET SDK 를 사용하는 경우 응용 프로그램이 .NET 을 지원해야 합니다.

- 개방 형식 쿠키를 수동으로 생성합니다.

  CA SiteMinder® Federation Standalone SDK 를 사용하지 않고 개방 형식 쿠키를 생성하려면 원하는 프로그래밍 언어로 쿠키를 생성하십시오. 개방 형식 쿠키의 내용 (페이지 469)에 대한 상세 정보를 참조하십시오.

  쿠키를 작성하는 데 사용하는 언어는 UTF-8 인코딩 및 CA SiteMinder® Federation Standalone 이 암호 기반 암호화에 사용하는 다음과 같은 PBE 암호화 알고리즘을 지원해야 합니다.

  - PBE/SHA1/AES/CBC/PKCS12PBE-1000-128
  - PBE/SHA1/AES/CBC/PKCS12PBE-1000-192
  - PBE/SHA1/AES/CBC/PKCS12PBE-1000-256
  - PBE/SHA256/AES/CBC/PKCS12PBE-1000-128
  - PBE/SHA256/AES/CBC/PKCS12PBE-1000-192
  - PBE/SHA256/AES/CBC/PKCS12PBE-1000-256
  - PBE/SHA1/3DES_EDE/CBC/PKCS12PBE-1000-3
  - PBE/SHA256/3DES_EDE/CBC/PKCS12PBE-1000-3

  쿠키를 암호화하는 데 FIPS 호환(AES) 알고리즘을 선택하는 경우 SDK 가 없으면 프로비저닝 응용 프로그램이 개방 형식 쿠키를 읽지 못합니다.

  또한 사용자의 브라우저에 개방 형식 쿠키가 설정되도록 해야 합니다.

  참고: CA SiteMinder® Federation Standalone 을 FIPS 전용 모드로 설치한 경우에는 개방 형식 쿠키만 사용할 수 있습니다.

개방 형식 쿠키 게시

개방 형식 쿠키 게시는 개방 형식 쿠키와 유사하지만 HTTP-POST 요청 형식으로 데이터를 보냅니다. 쿠키 데이터 제한으로 인해 데이터가 손실될 것을 우려하는 경우 이 옵션을 사용하십시오.

HTTP 헤더

프록시 모드를 사용하는 경우 이 정보를 HTTP 헤더로 전달할 수도 있습니다. HTTP 헤더를 사용하는 경우 CA SiteMinder® Federation Standalone 시스템 및 원격 프로비저닝 시스템이 서로 다른 도메인에 있을 수 있습니다.

전송 옵션은 파트너 관계 마법사의 "응용 프로그램 통합" 단계에서 구성할 수 있습니다.

사용자가 프로비저닝 응용 프로그램으로 리디렉션된 후에는 CA SiteMinder?Federation Standalone 에 더 이상 프로세스 제어권이 없습니다. 사용자 계정 프로비저닝에 많은 시간이 소요될 경우 프로비저닝 응용 프로그램은 현재 프로비저닝이 진행 중이라는 메시지를 사용자에게 전송하는 것과 같은 방법으로 문제를 처리하는 역할을 수행합니다. 이 정보를 통해 사용자는 사용자 계정을 사용할 수 있을 때까지 로그인을 시도하지 않아야 한다는 점을 알 수 있습니다.

원격 프로비저닝 구성

원격 프로비저닝을 구성하려면 어설션 데이터에 대한 전송 옵션을 결정하고 프로비저닝 서버의 URL 을 제공해야 합니다.

원격 프로비저닝 구성 이외에도 "IDP 가 사용자 식별자를 만들도록 허용" 옵션을 선택할 수 있습니다. 이 옵션을 선택하면 사용자에 대한 식별자가 없는 경우 IdP 가 영구 식별자를 생성할 수 있습니다. 이 "허용/만들기" 기능은 로컬 방법에서는 필수지만 로컬 계정 연결을 사용한 프로비저닝에만 사용되는 것은 아닙니다.

다른 특성과 함께 원격 프로비저닝 서버로 보낼 사용자 식별자를 IdP 가 생성하도록 하려면 "허용/만들기" 기능과 함께 원격 프로비저닝을 사용하도록 설정할 수 있습니다. 생성된 식별자를 사용하는 방법은 원격 프로비저닝 서버의 응용 프로그램에서 결정합니다. 응용 프로그램은 로컬 계정 연결을 수행할 수 있지만 CA SiteMinder?Federation Standalone 로컬 계정 연결은 수행할 수 없습니다.

원격 프로비저닝을 구성하려면

1. 파트너 관계 마법사의 "응용 프로그램 통합" 단계에서 시작합니다.
2. "사용자 프로비저닝" 그룹 상자에서 프로비저닝 유형을 선택합니다.

   참고: "도움말"을 클릭하면 필드, 컨트롤 및 해당되는 각 요구 사항에 대한 설명을 볼 수 있습니다.
3. 프로비저닝 유형으로 "원격"을 선택한 경우에는 다음과 같은 추가적인 필드에 데이터를 입력합니다.
   - 전송 옵션
   - 프로비저닝 서버 URL

4. 개방 형식 쿠키를 전송 옵션으로 지정하면 "개방 형식 쿠키" 그룹 상자에서 추가적인 설정을 지정해야 합니다.

   이러한 설정에는 쿠키 이름, 쿠키를 암호화하는 알고리즘 및 암호화 암호가 포함됩니다. 필요한 경우 HMAC 기능을 사용하여 쿠키의 무결성을 확인할 수도 있습니다.

5. 마법사에서 "확인" 단계를 선택하고 "마침"을 클릭하여 변경 내용을 저장합니다.

원격 프로비저닝 구성을 완료했습니다.

추가 정보:

프로비저닝 응용 프로그램으로 어설션 데이터 전송 (페이지 363)

리디렉션 URL 을 사용하여 실패한 인증 처리(신뢰 당사자)

어설션 기반 인증은 어설션을 소비하는 사이트에서 실패할 수 있습니다. 인증이 실패하는 경우 추가 처리를 위해 사용자를 다른 응용 프로그램(URL)으로 리디렉션하도록 CA SiteMinder?Federation Standalone 을 구성할 수 있습니다. 예를 들어 사용자 명확성이 실패하는 경우 SAML 어설션에 포함된 정보를 기반으로 사용자 계정을 생성할 수 있는 프로비저닝 시스템으로 사용자를 리디렉션하도록 CA SiteMinder?Federation Standalone 을 구성할 수 있습니다.

리디렉션 URL 설정은 선택 사항이며 신뢰 당사자 측에서만 구성할 수 있습니다.

리디렉션 URL 을 구성하려면

1. 파트너 관계 마법사의 "응용 프로그램 통합" 단계에서 시작합니다.
2. 대화 상자의 "상태 리디렉션 URL" 섹션에 사용자를 리디렉션할 오류 조건에 해당하는 설정만 구성합니다. "상태 리디렉션 URL" 그룹 상자의 설정은 다음과 같습니다.
   - 사용자를 찾을 수 없음
   - 잘못된 SSO 메시지
   - 수락되지 않은 사용자 자격 증명(SSO 메시지)

   참고: "도움말"을 클릭하면 필드, 컨트롤 및 해당되는 각 요구 사항에 대한 설명을 볼 수 있습니다.
3. 구성하는 각 리디렉션 옵션에 대해 CA SiteMinder® Federation Standalone 이 사용자를 리디렉션하는 방법을 지정합니다. 옵션은 다음과 같습니다.

   302 데이터 없음(기본값)

   HTTP 302 리디렉션을 사용하여 데이터 없이 사용자를 리디렉션합니다.

   HTTP Post

   HTTP Post 프로토콜을 사용하여 사용자를 리디렉션합니다.

리디렉션 URL 의 구성이 완료되었습니다.

제 23 장: 파트너 관계 구성에 유용한 메타데이터 내보내기

이 섹션은 다음 항목을 포함하고 있습니다.

메타데이터 내보내기 개요

로컬 엔터티는 원격 엔터티가 손쉽게 엔터티를 생성하고 파트너 관계를 구성할 수 있도록 메타데이터를 생성합니다. 파트너 관계의 많은 요소가 메타데이터 파일에서 정의되므로 메타데이터를 사용하면 파트너 관계를 보다 효율적으로 구성할 수 있습니다. 원격 파트너는 메타데이터를 가져와서 메타데이터 문서의 정보를 기반으로 한 파트너 관계나 원격 엔터티를 생성할 수 있습니다.

기존의 로컬 어설션 엔터티 또는 신뢰 엔터티에서 메타데이터를 내보낼 수 있습니다.

Administrative UI 에는 다음과 같이 메타데이터를 내보내기 위한 몇 가지 옵션이 있습니다.

- 로컬 엔터티에서 내보내기
- 로컬 파트너 관계에서 내보내기
- 로컬 WSFED 파트너 관계에 대한 메타데이터 교환

메타데이터를 보낼 때 파일을 사용하든 메타데이터 교환 프로필을 사용하든 관계없이 메타데이터를 가져오는 최종 목적은 동일합니다.

참고: SAML 1.1 의 경우 메타데이터 파일의 용어는 SAML 2.0 용어입니다. 이 규칙은 SAML 사양을 따릅니다. SAML 1.1 데이터를 가져올 때 용어는 SAML 1.1 용어를 사용하여 올바르게 가져오게 됩니다.

엔터티 수준 메타데이터 교환

로컬 엔터티에서 데이터를 내보낼 수 있습니다. 엔터티 수준에서 메타데이터를 내보낼 때는 내보내는 데이터에 파트너 관계 이름을 제공하십시오. 이 수준에서의 내보내기는 기본 파트너 관계 데이터를 정의합니다.

다음 단계를 수행하십시오.

1. Administrative UI 에 로그인합니다.
2. "페더레이션", "엔터티"를 클릭합니다.
3. 목록에서 로컬 항목 옆의 "작업" 풀다운 메뉴를 클릭하고 "메타데이터 내보내기"를 선택합니다.

   "메타데이터 내보내기" 대화 상자가 열립니다.
4. 새 파트너 관계 이름을 지정합니다. 내보내기를 통해 생성된 메타데이터 파일에는 기본 파트너 관계를 설정하기 위한 정보가 포함되어 있습니다.
5. 대화 상자의 나머지 필드를 채웁니다. 대화 상자의 "메타데이터 내보내기 옵션" 섹션에서 설정을 입력하십시오.

   참고: "도움말"을 클릭하면 필드, 컨트롤 및 해당되는 각 요구 사항에 대한 설명을 볼 수 있습니다.
6. "Export"(내보내기)를 클릭합니다.
7. 메타데이터 파일을 열지 아니면 저장할지 묻는 대화 상자가 표시됩니다.

   파일을 열기만 하여 표시합니다.
8. 데이터를 로컬 시스템의 XML 파일에 저장합니다.

메타데이터가 지정된 XML 파일로 내보내졌습니다. 이 파일을 모든 파트너로 전송할 수 있습니다.

파트너 관계 수준 메타데이터 교환

로컬 파트너 관계에서 데이터를 내보낼 수 있습니다. 이 수준에서의 내보내기는 기본 파트너 관계 데이터를 정의합니다.

다음 단계를 수행하십시오.

1. Administrative UI 에 로그인합니다.
2. "페더레이션", "파트너 관계"를 클릭합니다.
3. 목록의 파트너 관계 옆에 있는 "작업" 풀다운 메뉴를 선택합니다.
4. "메타데이터 내보내기"를 선택합니다.

   "메타데이터 내보내기" 대화 상자가 열립니다.
5. 정보를 검토합니다. 내보내기를 통해 생성된 메타데이터 파일에는 기본 파트너 관계를 설정하기 위한 정보가 포함되어 있습니다.
6. "메타데이터 내보내기 옵션" 섹션에서 메타데이터 문서의 서명 및 유효성 검사를 위한 설정을 완료합니다.

   참고: "도움말"을 클릭하면 필드, 컨트롤 및 해당되는 각 요구 사항에 대한 설명을 볼 수 있습니다.
7. "Export"(내보내기)를 클릭합니다.
8. 메타데이터 파일을 열지 아니면 저장할지 묻는 대화 상자가 표시됩니다.

   파일을 열기만 하여 표시합니다.
9. 데이터를 로컬 시스템의 XML 파일에 저장합니다.

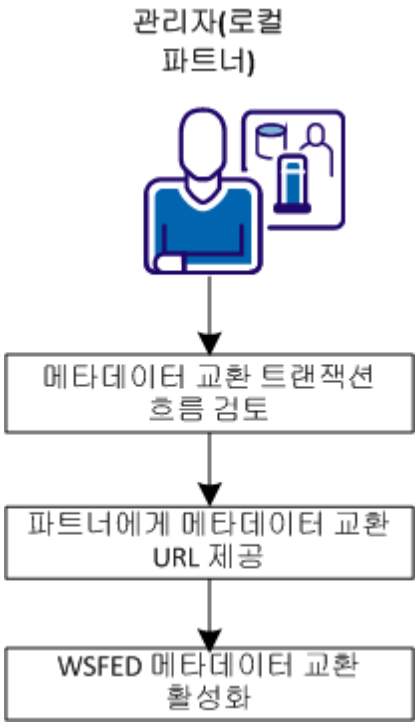
메타데이터가 지정된 XML 파일로 내보내졌습니다. 이 파일을 모든 파트너로 전송할 수 있습니다.

WS-페더레이션 메타데이터 교환이 사용되도록 설정하는 방법

정책 서버는 WS-페더레이션 파트너 관계에 대해 웹 서비스 메타데이터 교환 프로필을 지원합니다. 이 웹 서비스를 사용하면 SiteMinder 로컬 파트너가 원격 파트너의 메타데이터 요청에 응답할 수 있습니다. 교환은 HTTP 요청 및 응답의 형태로 이루어집니다.

HTTP 프로토콜을 사용하면 원격 엔터티가 프로그래밍 방식으로 페더레이션을 구성할 수 있습니다. 응용 프로그램에서 URL 을 사용하여 필요한 정보를 수집할 수 있습니다.

다음 그림에서는 메타데이터 교환을 위한 구성 단계를 보여 줍니다.



메타데이터 교환을 위한 다음 구성을 완료하십시오.

1. 메타데이터 교환 트랜잭션 흐름을 검토합니다. (페이지 373)
2. 메타데이터 교환 URL 을 파트너에게 제공합니다. (페이지 373)
3. WSFED 메타데이터 교환이 사용되도록 설정합니다 (페이지 374).

메타데이터 교환 트랜잭션 흐름

메타데이터 교환 트랜잭션의 프로세스 흐름은 다음과 같습니다.

1. 로컬 파트너가 제공한 메타데이터 교환 URL 로 원격 파트너가 요청을 보냅니다.
2. 로컬 파트너가 HTTP 응답에서 원격 파트너로 메타데이터를 다시 보냅니다. 정책 서버가 응답에 서명하여 메타데이터를 보호합니다. 원격 파트너가 응답을 확인하는 데 사용할 수 있는 인증서는 응답에 포함되어 있습니다.

   정책 서버는 요청이 있을 때 메타데이터 문서를 생성합니다. 이 문서는 로컬 파트너에 저장되지 않습니다.
3. 원격 파트너가 응답의 서명을 확인합니다. 서명이 유효하면 원격 파트너는 메타데이터 문서를 구문 분석하고 해당 정보를 사용하여 엔터티와 파트너 관계를 설정합니다.

파트너에 메타데이터 교환 URL 제공

메타데이터 트랜잭션이 발생하기 전에 원격 파트너에 메타데이터 교환 요청을 위한 URL 을 제공하십시오. 페더레이션된 파트너는 다음 URL 로 요청을 보내야 합니다.

https://*server:port*/affwebservices/public/FederationMetadata/*partnership_name*

server:port

메타데이터 교환 서비스를 호스트하는 시스템의 이름입니다.

partnership_name

구성된 파트너 관계의 이름입니다.

WSFED 메타데이터 교환이 사용되도록 설정

로컬 WS-페더레이션 파트너에서 메타데이터 교환 기능이 사용되도록 설정하십시오.

다음 단계를 수행하십시오.

1. Administrative UI 에 로그인합니다.
2. 수정할 WSFED 파트너 관계를 선택합니다.
3. 파트너 관계 마법사의 "파트너 관계 구성" 단계에서 "메타데이터 교환 사용" 확인란을 선택합니다.
4. "확인" 단계로 이동하고 "마침"을 클릭합니다.
5. 기본 "파트너 관계 페더레이션" 탭으로 돌아갑니다("페더레이션", "파트너 관계 페더레이션).
6. 왼쪽 창에서 "메타데이터 교환 구성"을 선택합니다.

   "메타데이터 교환 구성" 화면이 표시됩니다.
7. 응답에 서명할 값을 제공합니다.
8. "저장"을 클릭합니다.

파트너 관계에 대한 메타데이터 교환이 구성되었습니다.

제 24 장: 페더레이션 시스템의 장애 조치 지원

이 섹션은 다음 항목을 포함하고 있습니다.

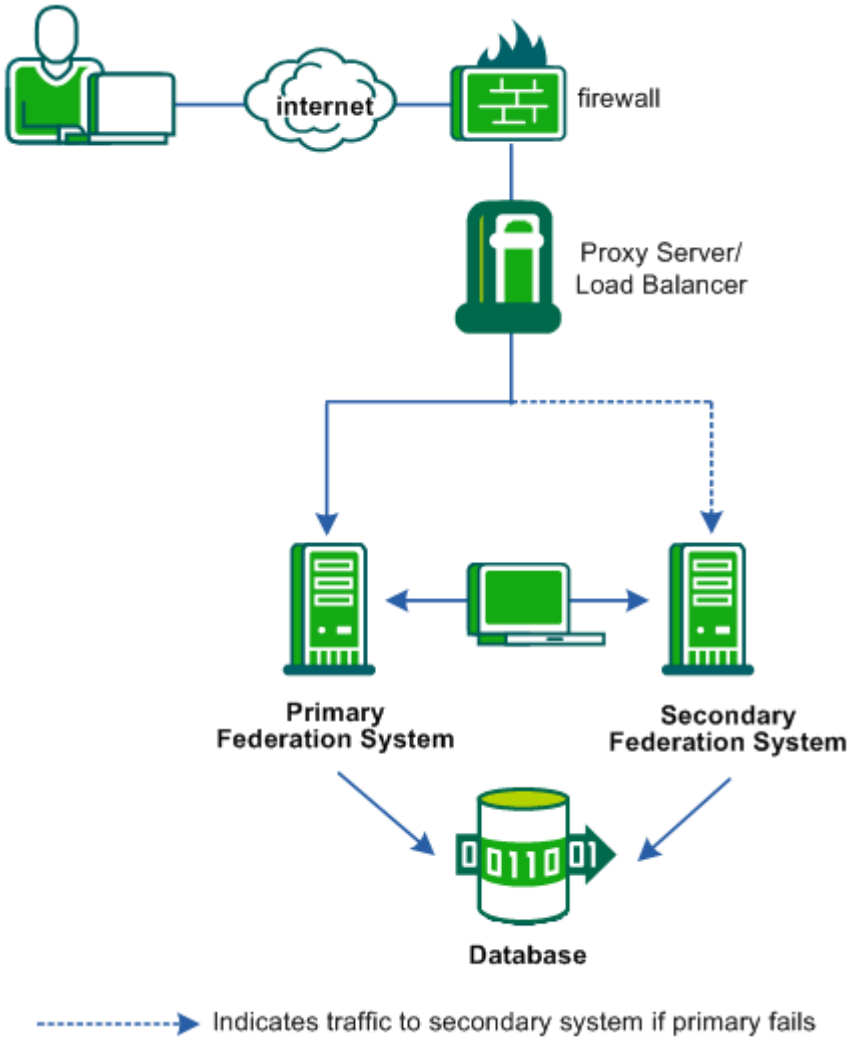
장애 조치 소개

장애 조치 기능은 페더레이션된 네트워크에서 CA SiteMinder® Federation Standalone 이 단일 장애 지점이 되지 않도록 도와줍니다. 장애 조치는 기본 및 보조 CA SiteMinder® Federation Standalone 시스템을 구성하여 네트워크에 중복 구조를 구축합니다. 기본 CA SiteMinder® Federation Standalone 시스템에서 장애가 발생할 경우 백업 시스템이 필요한 페더레이션 통신을 수행할 수 있습니다.

어설션 당사자 또는 신뢰 당사자 역할을 하는 CA SiteMinder?Federation Standalone 에 대해 장애 조치를 구성할 수 있습니다.

참고: SiteMinder 커넥터를 사용하도록 설정한 경우에는 커넥터 등록 프로세스에 대해 장애 조치 기능을 사용할 수 있습니다. 세부 지침은 Configure the SiteMinder Connector(SiteMinder 커넥터 구성) 섹션에 설명되어 있습니다.

다음 그림에서는 장애 조치가 구현된 CA SiteMinder?Federation Standalone 배포를 보여 줍니다. 기본 시스템에서 장애가 발생하면 트랜잭션이 보조 시스템으로 리디렉션됩니다.



위의 그림에서 볼 수 있듯이 CA SiteMinder?Federation Standalone 은 동일한 데이터베이스를 사용하는 두 개의 시스템에 설치됩니다.

장애 조치 구성 방법

장애 조치 구성은 다음과 같은 태스크로 이루어집니다.

- 적어도 두 개의 시스템에 CA SiteMinder® Federation Standalone 을 설치합니다.
- 두 페더레이션 시스템에서 구성 마법사를 실행합니다.
- 페더레이션 시스템의 장애 조치를 관리할 프록시 서버 또는 부하 분산 장치를 설정합니다.

각 페더레이션 시스템을 구성한 후에 프록시 서버 또는 부하 분산 장치에서 장애 조치를 구성하는 것이 좋습니다.

중요! 페더레이션 서비스에 SSL 을 사용하려면 SSL 사용 장애 조치 환경 (페이지 380)에 대한 지침을 따르십시오.

각 페더레이션 시스템에 장애 조치 설정

페더레이션 배포 환경에서 장애 조치를 사용하려면 기본 및 보조 CA SiteMinder?Federation Standalone 시스템이 설치 및 구성되어 있어야 합니다.

SSL 을 사용하는 장애 조치 환경의 경우 장애 조치 환경에 SSL 사용 (페이지 380)에 대한 지침을 참조하십시오.

중요! Solaris 플랫폼의 경우 Solaris 영역을 물리적 시스템으로 처리하십시오. 각 영역에 개별 CA SiteMinder® Federation Standalone 인스턴스를 설치 및 구성하십시오. 영역마다 호스트 ID 가 다르기 때문에 CA SiteMinder® Federation Standalone 에서는 단일 인스턴스에 대해 영역 간의 장애 조치를 지원하지 않습니다.

다음 단계를 수행하십시오.

1. 각 시스템에 제품을 설치하고 각각의 페더레이션 관리자 암호를 동일하게 지정합니다.

   참고: 제품은 독립 실행형 모드 또는 프록시 모드로 실행할 수 있지만 기본 서버와 보조 서버가 동일한 모드를 사용해야 합니다.

2. 두 시스템에 대해 동일한 데이터베이스 정보를 사용하여 각 시스템에서 페더레이션 시스템 구성 마법사를 실행합니다.
3. Administrative UI 에 로그인합니다.

4. "인프라" 탭에서 "시스템 설정"을 선택합니다.
5. 페더레이션된 네트워크에 있는 프록시 서버 또는 부하 분산 장치의 호스트와 포트를 포함하도록 "전역 기준 URL"을 변경합니다. 이 URL 을 설정하면 모든 파트너 관계의 모든 엔터티에 대한 기본 URL 을 올바르게 유지할 수 있습니다.

   CA SiteMinder® Federation Standalone 에서 가상 호스트나 도메인을 둘 이상 사용하는 경우에는 모든 항목을 포함하도록 server.conf 파일을 수정하십시오.

   server.conf 파일을 수정하려면

   a. *federation_install_dir*/secure-proxy/proxy-engine/conf 로 이동합니다.

   b. 편집기에서 server.conf 파일을 엽니다.

   c. # Default Virtual Host 섹션으로 이동합니다.

   d. 다음과 같이 정규화된 호스트 이름을 사용하여 **hostnames** 설정에 기본 URL 을 추가합니다.

      <VirtualHost name="default">

      hostnames="defaultbaseurl.example.com:80, newbaseurl.example.com:80"

      </VirtualHost>

      참고: hostnames 설정에 *host_name:port* 항목을 여러 개 지정하려면 각 항목을 쉼표로 구분하십시오.

      예:

      <VirtualHost name="default"

      hostnames=lb5.example.com:80

      </VirtualHost>

CA SiteMinder?Federation Standalone 시스템 둘 모두 같은 데이터베이스를 가리킵니다. 이제 기본 시스템에서 보조 시스템으로 장애 조치되도록 프록시 서버나 부하 분산 장치를 설정할 수 있습니다.

장애 조치에 사용할 프록시 서버 또는 부하 분산 장치 설정

프록시 서버나 부하 분산 장치가 CA SiteMinder?Federation Standalone 으로 장애 조치되도록 구성할 수 있습니다.

참고: 프록시 서버 또는 부하 분산 장치의 관리자는 배포 환경에서 시스템의 장애 조치를 설정하는 방법을 알고 있어야 합니다.

다음 단계를 수행하십시오.

1. CA SiteMinder® Federation Standalone 시스템 하나를 기본 호스트로 지정하고 다른 하나를 보조 호스트로 지정합니다.

   시스템에 대한 부하 분산은 구성하지 마십시오.

2. CA SiteMinder® Federation Standalone 배포 시 사용할 프록시 서버 또는 부하 분산 장치를 구성하고, 다음과 같은 URL 을 CA SiteMinder® Federation Standalone 시스템에 전달합니다.

   - /affwebservices/*
   - /siteminderagent/*

   이러한 URL 은 프록시 서버나 부하 분산 장치가 CA SiteMinder® Federation Standalone 시스템 간의 트래픽 부하를 분산할 수 있게 합니다.

프록시 서버 또는 부하 분산 장치가 구성되었습니다.

SSL 사용 장애 조치 구성 방법

부하 분산 장치 또는 프록시 서버 뒤에 페더레이션 시스템이 상주하는지 여부에 관계없이 장애 조치 환경에서 SSL 을 사용하도록 설정할 수 있습니다. 이와 같은 유형의 설정을 사용하려면 특별한 구성 지침을 따라야 합니다.

SSL 사용 장애 조치를 구성하려면 다음과 같은 태스크를 수행해야 합니다.

- 적어도 두 개의 시스템에 CA SiteMinder® Federation Standalone 을 설치합니다.
- 두 페더레이션 시스템에서 구성 마법사를 실행합니다.
- 포함된 Apache 웹 서버에서 SSL 을 사용하도록 설정합니다(페더레이션 시스템이 부하 분산 장치 뒤에 상주하는 경우에만 해당).
- SSL 구성을 기본 시스템에서 보조 시스템으로 마이그레이션합니다(<페더레이션 시스템이 부하 분산 장치 뒤에 상주하는 경우에만 해당).
- 페더레이션 시스템의 장애 조치를 관리할 프록시 서버 또는 부하 분산 장치를 설정합니다.

각 페더레이션 시스템을 구성한 후에 프록시 서버 또는 부하 분산 장치에서 장애 조치를 구성하는 것이 좋습니다.

부하 분산 장치 뒤에서 SSL 을 사용하는 장애 조치 구성

TCP 부하 분산 장치 뒤에 상주하도록 시스템을 구성할 수 있습니다. 부하 분산 장치가 요청을 시스템에 전달하면 시스템에서는 서버 측 SSL 처리를 진행합니다.

다음 단계를 수행하십시오.

1. 각 시스템에 제품을 설치하고 각각의 페더레이션 관리자 암호를 동일하게 지정합니다.

   참고: 제품은 독립 실행형 모드 또는 프록시 모드로 실행할 수 있지만 기본 서버와 보조 서버가 동일한 모드를 사용해야 합니다.

2. 구성 마법사를 실행하고 양쪽 시스템 모두에 동일한 데이터베이스 연결 정보를 사용합니다.
3. 구성 마법사에서 Apache 구성 정보를 요청합니다. 기본 및 보조 페더레이션 시스템의 "서버 이름" 설정에 동일한 가상 호스트 이름을 지정합니다. 두 시스템 모두 같은 가상 호스트 이름을 사용해야 합니다.

제품에서 둘 이상의 가상 호스트나 도메인을 사용하는 경우 프록시 엔진의 server.conf 파일을 수정하십시오. server.conf 파일에는 모든 호스트 이름과 도메인이 포함되어 있어야 합니다. Default VirtualHost 의 hostnames 필드에 이름을 추가하십시오.

server.conf 를 편집하려면

a. 다음 디렉터리로 이동합니다.

Windows: *federation_install_dir*\secure-proxy\proxy-engine\conf

UNIX: *federation_install_dir*/secure-proxy/proxy-engine/conf

b. 편집기에서 server.conf 파일을 엽니다.

c. # Default Virtual Host 섹션으로 이동하여 다음과 같이 정규화된 URL 을 사용하여 hostnames 설정에 이름을 추가합니다.

```
<VirtualHost name="default">
hostnames="virtualhost1.example.com, virtualhost2.example.com"
</VirtualHost>
```

참고: URL 여러 개를 쉼표로 구분하여 hostnames 설정에 지정할 수 있습니다.

4. Administrative UI 에 로그인합니다.
5. "인프라", "시스템 설정"을 클릭합니다.
6. 페더레이션된 네트워크에 있는 프록시 서버 또는 부하 분산 장치의 호스트와 포트를 포함하도록 "전역 기준 URL"을 변경합니다. 이 URL 을 설정하면 모든 파트너 관계의 모든 엔터티에 대한 기본 URL 을 올바르게 유지할 수 있습니다.

server.conf 파일을 수정하려면

a. *federation_install_dir*/secure-proxy/proxy-engine/conf 로 이동합니다.

b. 편집기에서 server.conf 파일을 엽니다.

c. # Default Virtual Host 섹션으로 이동합니다.

d. 다음과 같이 정규화된 호스트 이름을 사용하여 **hostnames** 설정에 기본 URL 을 추가합니다.

```
<VirtualHost name="default">
hostnames="defaultbaseurl.example.com:80,
newbaseurl.example.com:80"
</VirtualHost>
```

참고: hostnames 설정에 *host_name:port* 항목을 여러 개 지정하려면 각 항목을 쉼표로 구분하십시오.

7. 기본 페더레이션 시스템에서 포함된 Apache 웹 서버에 SSL 을 사용하도록 설정합니다.
8. 장애 조치 배포의 보조 시스템에 Apache SSL 구성을 마이그레이션합니다.
9. 부하 분산 장치에서 페더레이션 시스템에 매핑되는 여러 IP 주소를 동일한 호스트 이름에 대해 구성합니다.

보조 시스템에 SSL 설정 마이그레이션

기본 CA SiteMinder?Federation Standalone 시스템에 Apache SSL 을 구성한 후에는 부하 분산 장치 뒤에 상주하는 보조 시스템에 이 설정을 마이그레이션할 수 있습니다.

참고: CA SiteMinder® Federation Standalone 이 프록시 서버 뒤에 상주하는 경우에는 이 절차가 적용되지 않습니다.

다음과 같은 조건을 충족하는지 확인하십시오.

- 각 CA SiteMinder® Federation Standalone 시스템에서 동일한 인증서를 사용합니다.
- 각 CA SiteMinder® Federation Standalone 시스템이 동일한 호스트 이름을 사용하여 구성되어야 합니다.
- 부하 분산 장치를 통해 CA SiteMinder® Federation Standalone 에 액세스합니다.
- 모든 시스템이 같은 플랫폼(Windows/Solaris/Linux)을 사용해야 합니다.

SSL 구성을 보조 시스템에 복사하려면

1. 기본 CA SiteMinder® Federation Standalone 시스템에서 Apache SSL 을 사용하도록 설정합니다. Apache SSL 을 사용하면 다음과 같은 구성 요소를 사용할 수 있습니다.

   - SSL 서버 인증서

     federation_install_dir/secure-proxy/SSL/certs/server.crt

   - CA 번들

     federation_install_dir/secure-proxy/SSL/certs/ca-bundle.cert

   - SSL 서버 키

     federation_install_dir/secure-proxy/SSL/keys/server.key

   - 인증서 요청 파일

     federation_install_dir/secure-proxy/SSL/keys/fedmgrsslcertrequest.pem

   - SSL 속성 파일

     federation_install_dir/config/fedmanager.properties

2. SSL 서버 인증서에 서명한 CA 인증서를 보조 시스템에 가져옵니다. Administrative UI 를 사용하여 인증서를 가져옵니다.

   이 인증서는 기본 시스템에서 SSL 구성을 진행하는 동안이나 그 전에 가져와야 합니다. 기본 시스템에서 이 인증서에 사용한 것과 동일한 별칭을 사용하는 것이 좋습니다.

3. 1 단계에 나열된 각 파일을 보조 시스템의 동일한 위치에 복사합니다. 해당 폴더가 이미 있어야 합니다.

   다음에 주의하십시오.

   - 보조 시스템에 ca-bundle.cert 의 복사본이 이미 있어야 합니다. 이 복사본은 백업하거나 삭제해야 합니다. 기본 시스템에서 가져온 새 복사본에는 보조 시스템에 필요한 추가 데이터가 포함되어 있습니다.
   - 인증서 요청 파일(fedmgrsslcertrequest.pem)은 보조 시스템에서 Administrative UI 를 사용하여 해당 파일을 가져오고자 하는 경우에만 복사하십시오. 그렇지 않으면 파일을 복사하지 않아도 됩니다.

- SSL 속성 파일에는 적어도 다음의 두 가지 속성이 포함되어 있어야 합니다.
  - fedmgr.ssl.enabled - Y 로 설정
  - fedmgr.ssl.ca.alias - SSL 서버 인증서 요청에 서명한 CA 의 별칭으로 설정
  - 보조 시스템에 이 인증서를 가져올 때 다른 별칭을 사용한 경우, 실제로 사용한 별칭 값으로 이 속성을 업데이트하십시오.

이제 구성이 마이그레이션되어 보조 시스템에서 SSL 을 활성화할 수 있습니다.

보조 장애 조치 시스템에서 SSL 활성화

보조 시스템에 Apache SSL 구성을 마이그레이션한 후 SSL 을 사용할 수 있습니다.

보조 시스템에서 SSL 을 활성화하려면(Windows)

1. 보조 시스템에서 명령 프로그램 창을 엽니다.
2. *federation_install_dir*/secure-proxy/httpd/bin 폴더로 이동합니다.
3. 다음 명령을 실행합니다.

   configssl.bat -enable
4. 다음의 바로 가기를 사용하여 CA SiteMinder® Federation Standalone 서비스를 중지했다가 다시 시작합니다.

   로컬 관리자가 아닌 네트워크 사용자로 로그인한 경우에는 바로 가기를 마우스 오른쪽 단추로 클릭하고 "관리자 권한으로 실행"을 선택하십시오.
   - "시작", "모든 프로그램", "CA", "Federation Standalone", "서비스 중지"
   - "시작", "모든 프로그램", "CA", "Federation Standalone", "서비스 시작"

보조 시스템에서 SSL 을 활성화하려면(UNIX)

1. *federation_install_dir* 로 이동합니다.
2. 다음 명령을 실행하여 CA SiteMinder® Federation Standalone 서비스를 종료합니다.

   ./fedmanager.sh stop
3. 다음 명령을 실행하여 CA SiteMinder® Federation Standalone 서비스를 SSL 사용 모드로 다시 시작합니다.

   ./fedmanager.sh startssl
4. 메시지가 표시되면 SSL 모드에서 Apache 웹 서버를 시작할 수 있도록 Administrative UI 암호를 입력합니다.

프록시 서버 뒤에 SSL 사용 장애 조치 구성

CA SiteMinder?Federation Standalone 이 프록시 서버 뒤에 상주하는 경우에는 프록시 서버가 SSL 프로세스를 처리합니다. 대부분의 프록시 서버가 SSL 요청의 처리를 다른 시스템에 위임할 수 없기 때문에 CA SiteMinder?Federation Standalone 에서 SSL 을 처리할 수 없습니다. 따라서 프록시 서버에서 서버 인증서, 서버 인증서에 서명한 CA 인증서 및 모든 원격 클라이언트 인증서를 구성해야 합니다.

프록시 서버 뒤에 상주하는 CA SiteMinder?Federation Standalone 시스템에는 별도의 SSL 구성이 필요하지 않습니다.

장애 조치에 사용할 프록시 서버 또는 부하 분산 장치 설정

프록시 서버나 부하 분산 장치가 CA SiteMinder?Federation Standalone 으로 장애 조치되도록 구성할 수 있습니다.

참고: 프록시 서버 또는 부하 분산 장치의 관리자는 배포 환경에서 시스템의 장애 조치를 설정하는 방법을 알고 있어야 합니다.

다음 단계를 수행하십시오.

1. CA SiteMinder® Federation Standalone 시스템 하나를 기본 호스트로 지정하고 다른 하나를 보조 호스트로 지정합니다.

   시스템에 대한 부하 분산은 구성하지 마십시오.

2. CA SiteMinder® Federation Standalone 배포 시 사용할 프록시 서버 또는 부하 분산 장치를 구성하고, 다음과 같은 URL 을 CA SiteMinder® Federation Standalone 시스템에 전달합니다.

   - /affwebservices/*
   - /siteminderagent/*

   이러한 URL 은 프록시 서버나 부하 분산 장치가 CA SiteMinder® Federation Standalone 시스템 간의 트래픽 부하를 분산할 수 있게 합니다.

프록시 서버 또는 부하 분산 장치가 구성되었습니다.

각 시스템에 동일한 구성 유지

구성을 변경한 경우에는 항상 해당 변경 내용을 기본 CA SiteMinder?Federation Standalone 시스템에서 관리한 후 보조 시스템에 구성을 내보내야 합니다.

구성 변경과 관련하여 다음과 같은 내용을 참조하십시오.

구성 변경 후 지연

기본 시스템 UI 를 사용하여 변경한 내용이 보조 시스템에 항상 즉각적으로 적용되지는 않습니다. 데이터베이스에 저장되는 UI 관리 데이터를 보조 시스템에서 사용할 수 있는 이유는 기본 시스템과 보조 시스템이 동일한 데이터베이스를 공유하기 때문입니다. 그러나 보조 시스템의 정책 엔진에 변경 내용이 적용되는 데 어느 정도의 시간이 필요할 수 있습니다.

다시 시작해야 적용되는 일부 구성 변경

일부 구성 변경 내용은 시스템을 다시 시작해야 적용됩니다. 다시 시작해야 하는 기본 시스템 구성을 변경하면 보조 시스템 또한 다시 시작해야 합니다.

동일한 기본 시스템에서 관리 수행

항상 동일한 기본 시스템에서 관리를 수행해야 합니다. 그러기 위해 보조 시스템에서는 UI 관리 기능이 사용되지 않도록 설정할 수 있습니다.

다음 단계를 수행하십시오.

1. Administrative UI 에 로그온합니다.
2. "인프라", "시스템 설정"을 선택합니다.

   "시스템 설정 구성" 대화 상자가 나타납니다.
3. "UI 설정" 그룹 상자에서 "관리 사용 안 함"을 클릭합니다.

   작업을 확인하는 메시지가 나타납니다.
4. "예"를 클릭하여 UI 관리가 사용되지 않도록 설정합니다.

   UI 의 다른 부분이 모두 사용 불가능해지면서 "관리 사용 안 함" 대화 상자가 표시됩니다. 이 대화 상자에서 관리를 다시 사용하도록 설정할 수 있습니다.

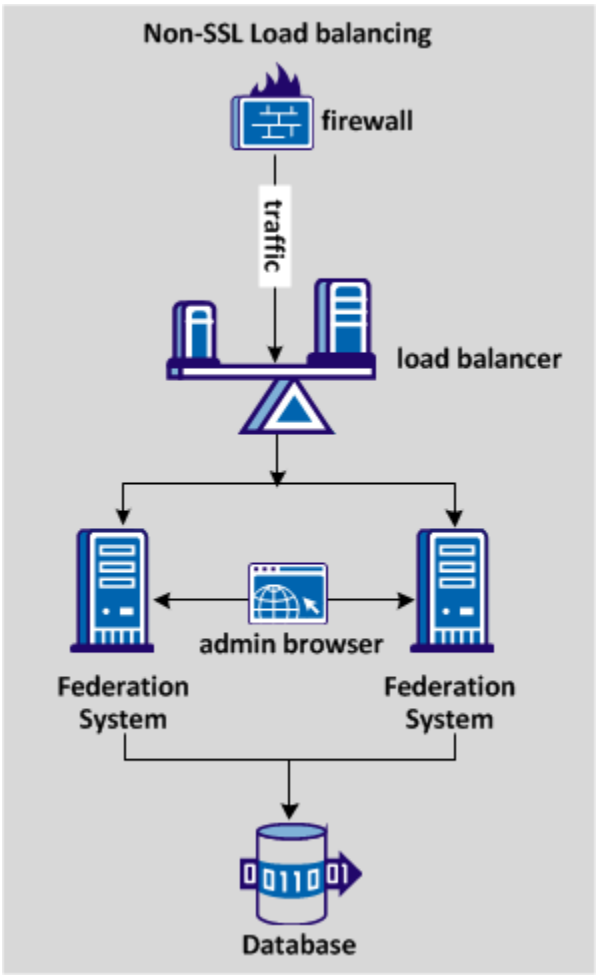
제 25 장: 페더레이션 시스템의 부하 분산 지원

이 섹션은 다음 항목을 포함하고 있습니다.

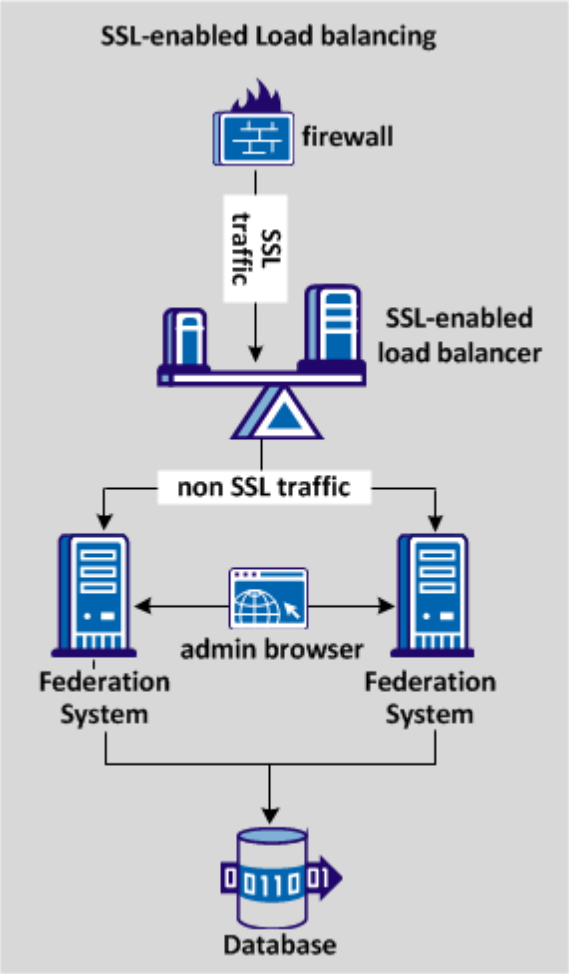
부하 분산 구성 방법

부하 분산은 단일 장치에 부하가 집중되지 않도록 통신 작업을 네트워크에 균일하게 분산시킵니다. CA SiteMinder?Federation Standalone 은 독립 실행형 끝점으로 사용되도록 개발되었기 때문에 부하 분산이 기본적으로 지원되지 않습니다. 그러나 CA SiteMinder?Federation Standalone 이 사용되는 네트워크에서 간단한 부하 분산 배포를 사용할 수 있습니다. SSL 사용 부하 분산을 사용할지 여부는 선택 사항이지만 중요한 데이터를 전송하는 경우에는 이 기능을 사용하는 것이 좋습니다.

다음 그림에서는 SSL 을 사용하지 않는 부하 분산 배포를 보여 줍니다.



다음 그림에서는 동일한 데이터베이스 저장소를 사용하는 두 시스템 사이에 트래픽을 분산시키는 SSL 사용 부하 분산 배포를 보여 줍니다.



부하 분산 트래픽에 대해 다음의 구성 단계를 완료하십시오.

1. 부하 분산 장치를 구성합니다.
2. 부하 분산 장치와 함께 사용되도록 둘 이상의 CA SiteMinder® Federation Standalone 시스템을 설정합니다.
3. (선택 사항) 부하 분산 장치가 SSL 을 사용하는 경우, SSL 리디렉션을 처리하도록 CA SiteMinder® Federation Standalone 을 구성합니다.

부하 분산 장치 구성

CA SiteMinder?Federation Standalone 을 사용하는 네트워크에서 부하 분산 장치가 작동하도록 구성할 수 있습니다. 부하 분산 장치 호스트와 포트를 사용하면 CA SiteMinder?Federation Standalone 시스템에 상주하는 리소스를 요청하는 사용자를 리디렉션할 수 있습니다. 부하 분산 장치 호스트와 포트의 사용은 CA SiteMinder?Federation Standalone 시스템의 모든 리소스에 적용됩니다.

참고: 이 절차에서는 부하 분산 장치 관리자가 배포 시 시스템을 설정하는 방법을 알고 있다고 가정합니다.

다음 단계를 수행하십시오.

1. 페더레이션 배포의 IP 주소와 호스트 이름을 매핑하도록 부하 분산 장치를 구성합니다.
2. CA SiteMinder® Federation Standalone 시스템 전체에 다음 URL 이 전달되도록 배포 환경에서 부하 분산 장치를 구성합니다.
   - /affwebservices/*
   - /siteminderagent/*
   - /forms/*

   이러한 URL 은 부하 분산 장치가 페더레이션 시스템 간에 트래픽을 분산시키는 데 사용됩니다.
3. (선택 사항) SSL 트래픽을 처리하도록 부하 분산 장치를 구성합니다.

   SSL 을 사용하도록 부하 분산 장치를 설정하면 모든 페더레이션 트래픽이 SSL 을 통해 부하 분산 장치에 수신됩니다. 그러나 부하 분산 장치는 비 SSL(HTTP) 연결을 통해 CA SiteMinder® Federation Standalone 시스템에 트래픽을 전송합니다.

이제 CA SiteMinder?Federation Standalone 시스템에서 부하 분산 장치를 사용하도록 구성을 완료했습니다.

부하 분산 장치와 함께 작동하도록 페더레이션 시스템 설정

페더레이션 배포 환경에서 부하 분산을 사용하려면 둘 이상의 CA SiteMinder?Federation Standalone 시스템을 설정해야 합니다.

참고: 이 절차에서는 모든 시스템이 r12.52 SP1 버전이라고 가정합니다.

다음 단계를 수행하십시오.

1. 각 시스템에 제품을 설치하고 각각의 페더레이션 관리자 암호를 동일하게 지정합니다.

   참고: 제품을 독립 실행형 모드로 실행하는지 또는 프록시 모드로 실행하는지에 관계없이 서버가 동일한 모드를 사용해야 합니다.

2. 시스템 하나에서 구성 마법사를 실행합니다.
3. Administrative UI 에 로그인합니다.
4. "인프라", "시스템 설정"으로 이동합니다.
5. "서버 설정" 섹션에서 네트워크에 있는 부하 분산 장치의 호스트와 포트를 포함하도록 "전역 기준 URL"을 변경합니다. 모든 파트너 관계 엔터티의 기본 URL 이 올바르게 이 URL 을 설정합니다.
6. 다음 태스크를 완료하여 페더레이션 파트너 관계를 설정합니다.
   a. 인증서와 개인 키를 가져옵니다.
   b. 사용자 디렉터리 연결을 설정합니다 (페이지 87).
   c. 로컬 엔터티를 구성합니다 (페이지 127).
   d. 원격 엔터티를 지정합니다 (페이지 127).
   e. 로컬 엔터티와 원격 엔터티 사이에 파트너 관계를 구성합니다.
   f. 원격 파트너에서 페더레이션이 작동하는지 확인합니다.
7. 첫 번째 시스템에서 입력한 부하 분산 장치의 가상 호스트 이름을 동일하게 사용하여 보조 시스템에서 구성 마법사를 실행합니다.

   페더레이션 시스템 각각에서 동일한 가상 호스트 이름을 사용해야 합니다. 가상 호스트 이름은 구성 마법사를 실행할 때 Apache 구성에서 "서버 이름"으로 지정한 호스트입니다.

제품이 둘 이상의 가상 호스트 또는 도메인을 사용하는 경우에는 server.conf 파일을 수정하여 이러한 추가적인 항목도 포함해야 합니다.

server.conf 파일을 수정하려면

a. *federation_install_dir*/secure-proxy/proxy-engine/conf 로 이동합니다.

b. 편집기에서 server.conf 파일을 엽니다.

c. # Default Virtual Host 섹션으로 이동합니다.

d. 다음과 같이 정규화된 호스트 이름을 사용하여 **hostnames** 설정에 기본 URL 을 추가합니다.

```
<VirtualHost name="default">
hostnames="defaultbaseurl.example.com:80,
newbaseurl.example.com:80"
</VirtualHost>
```

참고: hostnames 설정에 *host_name:port* 항목을 여러 개 지정하려면 각 항목을 쉼표로 구분하십시오.

예:

```
<VirtualHost name="default"
hostnames=lb5.example.com:80
</VirtualHost>
```

8. 포함된 Apache 및 Tomcat 웹 서버에 저장되어 있는 SSL 키와 인증서를 마이그레이션합니다.

- 이 태스크는 SSL 마이그레이션 절차 (페이지 422)에 따라 완료합니다. SSL 데이터를 마이그레이션하면 새로운 키나 인증서를 구입하지 않아도 됩니다.
- 새로운 키/인증서 요청을 생성한 다음 인증서 서명을 받습니다. 가져온 구성 파일에는 SSL 인증서가 포함되어 있지 않습니다.

참고: 시스템 하나에서 인증서 구성을 변경할 경우 해당 변경 내용을 다른 모든 시스템에 복제하십시오. 구성 변경 내용은 UI 의 "인증서 및 키" 페이지에서 지정하십시오. 변경 내용으로는 인증서, 키 또는 CRL 데이터의 추가 또는 제거가 포함됩니다.

9. 파트너 관계가 구성되어 있지 않은 다른 시스템의 Administrative UI 에 로그인합니다.
10. "인프라", "시스템 설정"으로 이동합니다. "UI 설정" 섹션에서 "관리 사용 안 함"을 클릭합니다.

    부하 분산 장치를 통하지 않고 로컬로 Administrative UI 에 액세스합니다. 다른 시스템이 현재 실행 중이면 시스템 하나에서만 관리를 사용하도록 설정합니다. 언제든지 관리 시스템을 사용할 수 없는 경우 다른 시스템에 로그인하여 관리를 다시 사용하도록 설정합니다.

이제 모든 페더레이션 시스템이 동일한 데이터 저장소를 가리키도록 설정되었으므로 구성된 부하 분산 장치가 해당 시스템 사이에 트래픽을 분산시킬 수 있습니다.

SSL 부하 분산 장치로의 리디렉션 구성(선택 사항)

부하 분산 장치에서 SSL 을 사용할 경우, SSL 연결을 통해 트래픽을 리디렉션하도록 시스템을 구성하는 것이 좋습니다. 트래픽을 리디렉션하려면 페더레이션 시스템 각각에서 다음 파일 두 개를 수정하십시오.

- LocalConfig.conf
- httpd.conf

참고: 트래픽을 리디렉션하는 모든 페더레이션 시스템에서 이러한 파일을 수정하십시오.

다음 단계를 수행하십시오.

1. *federation_install_dir*/secure-proxy/proxy-engine/conf/defaultagent 로 이동합니다.
2. WebAgent.conf 파일을 편집기에서 엽니다. 파일에서 **localconfigfile** 로 시작되는 줄의 주석을 제거하고 파일을 저장합니다.
3. LocalConfig.conf 파일을 편집기에서 엽니다.
4. LocalConfig.conf 파일에 다음 설정을 추가한 후 파일을 저장합니다.

   ```
   HttpsPorts="443"
   ```

   부하 분산 장치가 수신 대기하는 포트를 지정합니다.

   ```
   GetPortFromHeaders="YES"
   ```

5. *federation_install_dir*\secure-proxy\httpd\conf 로 이동합니다.

6. 편집기에서 httpd.conf 파일을 엽니다.
7. SeverName 설정을 찾아 부하 분산 장치 *hostname:port* 를 지정합니다. 여기에 페더레이션 시스템 서버 호스트 이름을 입력하지 마십시오.

   예:

   ```
   ServerName lb5.example.com:443
   ```

8. ServerName 설정 다음에 UseCanonicalName 설정을 추가하고 값을 On 으로 지정합니다. 예:

   ```
   UseCanonicalName on
   ```

이렇게 하면 페더레이션 시스템이 SSL 연결을 통해 트래픽을 리디렉션합니다.

제 26 장: 페더레이션 시스템 관리

이 섹션은 다음 항목을 포함하고 있습니다.

서버 상태 모니터링

"서버 상태" 대화 상자에는 서버 상태를 확인하는 데 유용한 스냅샷 정보가 표시됩니다. 시스템 성능을 향상시키거나 설치가 예상대로 진행되었는지 확인하는 것을 예로 들 수 있습니다.

참고: "도움말"을 클릭하면 필드, 컨트롤 및 해당되는 각 요구 사항에 대한 설명을 볼 수 있습니다.

서버 설정을 보려면

1. Administrative UI 에 로그인합니다.
2. "인프라", "서버 상태"를 차례로 선택합니다.
3. 상태 페이지에 표시되는 정보를 검토합니다.

언제든지 "새로 고침"을 클릭하면 업데이트된 서버 정보를 볼 수 있습니다.

시스템 설정 수정

"시스템 설정 구성" 대화 상자에서는 시스템 성능에 영향을 줄 수 있는 활성 서버 스레드 수와 허용되는 활성 서버 연결 수를 지정할 수 있습니다. 또한 로컬 호스트의 UI 관리 기능을 사용하지 않도록 설정하거나 다시 사용하도록 설정할 수도 있습니다.

다음 단계를 수행하십시오.

1. Administrative UI 를 시작합니다.
2. "인프라" 탭에서 "시스템 설정"을 선택합니다.

3. 필요에 따라 설정을 수정합니다.

   참고: "도움말"을 클릭하면 필드, 컨트롤 및 해당되는 각 요구 사항에 대한 설명을 볼 수 있습니다.

4. "저장"을 클릭합니다.
5. CA SiteMinder® Federation Standalone 을 다시 시작합니다.

   변경 사항은 서버를 다시 시작해야만 적용되지만 예외적으로 "관리 사용 안 함" 기능은 즉시 적용됩니다.

배포 설정

배포 설정을 사용하면 다음을 수행할 수 있습니다.

- FIPS 모드 설정을 봅니다.
- CA SiteMinder® Federation Standalone 배포 모드(독립 실행형 또는 프록시)를 봅니다.
- 올바른 페더레이션 도메인을 지정합니다.
- SiteMinder 커넥터를 사용하는 상태에서 CA SiteMinder® Federation Standalone 이 작동하는 경우 SiteMinder 커넥터를 구성합니다.
- CA SiteMinder® Federation Standalone 세션 쿠키 이름을 수정합니다.

배포 모드 및 FIPS 설정

"배포 설정"에는 CA SiteMinder?Federation Standalone 을 설치하고 구성할 때 선택한 배포 모드와 FIPS 모드의 상태가 표시됩니다. 그뿐만 아니라 올바른 페더레이션 도메인을 지정하고 프록시 모드에서 HTTP 헤더를 보호하는 접두사를 설정할 수도 있습니다.

각 설정을 수정하는 프로세스는 서로 다릅니다.

올바른 페더레이션 도메인 및 HTTP 헤더 접두사

UI 에서 "올바른 페더레이션 도메인" 및 "HTTP 헤더 접두사" 항목을 수정하십시오. 이러한 설정을 변경하는 것은 선택 사항입니다.

다음 단계를 수행하십시오.

1. 필요한 경우 필드에 값을 입력합니다.
2. 섹션 오른쪽에 있는 "저장"을 클릭합니다.

FIPS 모드 변경

FIPS 모드를 변경하려면 설치 마법사를 다시 실행하여 새 설정을 선택하십시오.

중요! FIPS 모드를 변경할 때마다 CA SiteMinder® Federation Standalone 을 다시 시작하십시오.

배포 모드 변경

배포 모드를 변경하려면 구성 마법사를 다시 실행하여 모드를 변경하십시오.

추가 정보:

암호화 및 암호 해독 알고리즘 (페이지 473)

신뢰 당사자의 프록시 모드 배포에 대한 HTTP 헤더 보호

신뢰 당사자의 프록시 모드 배포에서 CA SiteMinder?Federation Standalone 은 HTTP 헤더를 사용하여 아이덴티티 특성을 SAML 어설션에서 백엔드 응용 프로그램으로 전달합니다. 대개의 경우 헤더는 안전합니다. 하지만 권한 없는 사용자가 어설션 특성 이름을 알면 이 이름을 브라우저에서 헤더로 설정하여 대상 응용 프로그램에 액세스할 수 있습니다. 대상 응용 프로그램은 예상된 헤더 값을 발견하고 CA SiteMinder?Federation Standalone 의 어설션 소비 없이 리소스에 대한 액세스를 허용합니다.

"HTTP 헤더 접두사" 설정에 대한 값을 지정하면 다음 시나리오를 방지할 수 있습니다.

1. 인증되지 않은 사용자가 HTTP 헤더의 이름을 알아냅니다. 이 헤더 이름에는 접두사가 포함되어 있습니다.
2. 악의적인 사용자가 헤더를 포함하여 들어오는 요청을 CA SiteMinder® Federation Standalone 에 전송합니다.
3. CA SiteMinder® Federation Standalone 은 접두사를 포함한 헤더의 출처가 들어오는 요청이며 내부적으로 생성되지 않은 것으로 인식하므로 이를 제거합니다.
4. CA SiteMinder® Federation Standalone 은 자체의 정상 헤더를 대상 응용 프로그램에 전달하기 전에 각 헤더에 지정된 접두사를 추가하고 헤더를 대상 응용 프로그램에 전달합니다.

HTTP 헤더 접두사를 설정하려면

1. "인프라", "배포 설정"으로 이동합니다.
2. "HTTP 헤더 접두사" 필드에 유효한 문자열을 접두사로 입력합니다.

   CA SiteMinder® Federation Standalone 을 설치할 때 프록시 모드를 사용하도록 설정한 경우에만 이 필드가 표시됩니다.
3. 변경 내용을 저장합니다.

SiteMinder 커넥터 설정

SiteMinder 커넥터는 CA SiteMinder?Federation Standalone 을 SiteMinder 환경에 통합되어 페더레이션된 통신을 할 수 있게 도와줍니다.

어설션 당사자 측에서 SiteMinder 커넥터는 위임된 인증을 위한 타사 WAM 으로 SiteMinder 와 함께 작동할 수 있습니다. 신뢰 당사자 측에서 SiteMinder 는 대상 리소스가 상주하고 있는 서버를 보호할 수 있습니다. SiteMinder 가 액세스 제어를 수행하는 경우 SiteMinder 커넥터는 SiteMinder 가 대상 리소스에 대한 액세스를 부여할 수 있도록 정책 서버에 연결하여 SiteMinder 세션을 설정합니다.

CA SiteMinder?Federation Standalone 을 SiteMinder 와 함께 사용하려면 Administrative UI 에서 SiteMinder 커넥터 설정을 구성하십시오.

SiteMinder 커넥터를 사용하는 모든 파트너 관계는 하나의 구성을 사용하며 SiteMinder 환경 하나에만 연결됩니다. 커넥터 구성은 Administrative UI 의 "배포 설정"에서 정의하십시오. 특정 파트너 관계에 대해 커넥터를 사용하려면 커넥터를 해당 파트너 관계 수준에서 설정하십시오. 커넥터를 사용하지 않으려면 "배포 설정"에서 사용하지 않도록 전역적으로 설정하거나 파트너 관계 수준에서 사용하지 않도록 설정하십시오.

중요! 전역 수준에서 커넥터를 사용하지 않도록 설정하면 CA SiteMinder® Federation Standalone 의 파트너 관계 수준에서 확인란이 무시됩니다.

다음 단계를 수행하십시오.

1. Administrative UI 에 로그인합니다.
2. "페더레이션된 파트너 관계 목록"에서 파트너 관계를 선택합니다.

   "파트너 관계" 대화 상자가 열립니다.
3. 다음 작업 중 하나를 수행하십시오.
   a. 신뢰 당사자 측에서 파트너 관계 마법사의 "사용자 ID" 단계로 이동합니다.
   b. 어설션 당사자 측에서 파트너 관계 마법사의 "페더레이션 사용자" 단계로 이동합니다.
4. "SiteMinder 커넥터 사용" 확인란을 선택합니다.

   그러면 구성 필드가 활성화됩니다.
5. (선택 사항) "UserDN 및 디렉터리 이름 비교 적용" 확인란을 선택합니다. 이 확인란을 선택하면 CA SiteMinder® Federation Standalone 의 사용자 디렉터리와 SiteMinder 의 디렉터리 사이에 UserDN 및 UserDirectory 이름 항목에 대한 비교가 수행됩니다.

   이 확인란을 선택할 경우에는 CA SiteMinder® Federation Standalone 배포와 SiteMinder 배포의 사용자 디렉터리가 물리적으로 같은 디렉터리여야 합니다. 사용자가 저장소 조회를 수행하려면 두 디렉터리의 이름이 같아야 합니다. 이 확인란의 선택을 취소하면 CA SiteMinder® Federation Standalone 에서 유니버설 ID 를 사용하여 사용자 레코드를 찾기 때문에 두 디렉터리가 서로 달라도 됩니다. 유니버설 ID 를 사용하면 각 사용자의 유니버설 ID 가 고유해야 합니다. 유니버설 ID 가 고유하지 않으면 사용자 레코드에 액세스하는 시스템이 잘못된 레코드를 가져올 수 있습니다.

6. 변경 내용을 저장합니다.
7. "인프라" 탭으로 이동합니다.
8. "인프라" 탭에서 "배포 설정"을 선택합니다.

   "배포 설정 구성" 대화 상자가 열립니다.
9. "SiteMinder 커넥터 설정" 섹션의 모든 필드에 값을 지정합니다.

   참고: "도움말"을 클릭하면 필드, 컨트롤 및 해당되는 각 요구 사항에 대한 설명을 볼 수 있습니다.
10. "호스트 등록"을 선택하고 SiteMinder 정책 서버의 관리자 자격 증명을 입력합니다.

    이 단계에서는 CA SiteMinder® Federation Standalone 이 SiteMinder 정책 서버에 에이전트로 등록됩니다.

    참고: 정책 서버를 두 개 이상 지정하여 호스트 등록 프로세스에서 장애 조치를 지원하도록 구성할 수 있습니다. 기본 정책 서버에서 등록에 실패할 경우 등록 프로세스가 성공적으로 완료될 때까지 지정된 다음 정책 서버로 CA SiteMinder® Federation Standalone 이 이동합니다.
11. 대화 상자의 "SiteMinder 커넥터 설정" 섹션에서 "저장"을 선택합니다.

    호스트를 등록한 후에는 "SiteMinder 커넥터 설정" 섹션에서 "저장"을 선택해야 합니다.
12. 운영 환경에 맞게 페더레이션 서비스를 다시 시작합니다.

    - **Windows**

      다음과 같이 중지 및 시작 바로 가기를 사용합니다. 로컬 관리자가 아닌 네트워크 사용자로 로그인한 경우에는 바로 가기를 마우스 오른쪽 단추로 클릭하고 "관리자 권한으로 실행"을 선택하십시오.

      a. "시작", "모든 프로그램", "CA", "Federation Standalone", "서비스 중지"

      b. "시작", "모든 프로그램", "CA", "Federation Standalone", "서비스 시작"

- **UNIX**

  a. 명령 창을 엽니다.

  b. 다음 스크립트를 실행합니다.

  federation_install_dir/fedmanager.sh stop

  federation_install_dir/fedmanager.sh start

  참고: 서비스를 중지했다가 루트 사용자로 시작하지 마십시오.

SiteMinder 커넥터 구성이 완료되었습니다.

세션 및 아이덴티티 쿠키의 쿠키 설정

CA SiteMinder?Federation Standalone 은 싱글 사인온 보안 영역을 지원합니다. 싱글 사인온 보안 영역은 같은 쿠키 도메인 내에 있는 응용 프로그램 그룹 간에 구성 가능한 트러스트 관계를 제공합니다.

하나의 영역 내에서는 싱글 사인온이 적용되지만, 사용자가 다른 영역에 액세스할 경우에는 영역 간에 정의된 트러스트 관계에 따라 재인증이 요청될 수 있습니다. 트러스트 관계에 포함된 보안 영역은 해당 그룹의 영역에서 유효한 세션을 가진 사용자에게 재인증을 요청하지 않습니다.

보안 영역 가맹은 쿠키 이름에 반영됩니다. CA SiteMinder?Federation Standalone 의 경우 기본 세션 및 아이덴티티 쿠키의 이름은 각각 FEDSESSION 과 FEDPROFILE 입니다.

페더레이션 파트너도 고유한 세션 또는 아이덴티티 쿠키를 사용하는 응용 프로그램을 갖고 있을 수 있습니다. 경우에 따라 파트너 쿠키의 이름과 CA SiteMinder?Federation Standalone 의 쿠키 이름이 충돌할 수 있습니다. 예를 들어 SiteMinder 사이트와 통신하는 경우, SiteMinder 도 고유한 세션 및 아이덴티티 쿠키를 생성하기 때문에 FEDSESSION 및 FEDPROFILE 이라는 이름의 쿠키가 존재할 수 있습니다. 이런 경우, CA SiteMinder?Federation Standalone 의 전역 쿠키 영역 접두사를 변경하여 쿠키 이름을 변경할 수 있습니다.

참고: CA SiteMinder® Federation Standalone SDK 를 사용하는 응용 프로그램이 있는 경우 "전역 쿠키 영역" 및 "암호화 암호" 설정에 구성하는 값이 해당 SDK 에서 사용하는 값과 일치해야 합니다. 조직 내의 관련 당사자들에게 이러한 설정 값을 반드시 공유하십시오. 어설션 당사자의 경우 SDK 및 웹 액세스 관리 시스템에 이러한 값이 필요합니다. 신뢰 당사자의 경우 CA SiteMinder® Federation Standalone 및 응용 프로그램을 호스트하는 대상 시스템에 이러한 값이 필요합니다.

자세한 내용은 *CA SiteMinder® Federation Standalone Java SDK 안내서* 또는 *.NET SDK 안내서*를 참조하십시오.

이 그룹 상자의 나머지 쿠키 매개 변수는 개방 형식 쿠키 설정입니다. 개방 형식 쿠키 설정은 개방 형식 쿠키 방식의 위임된 인증에만 사용되며 파트너 관계 기반이 아니라 전역 수준으로 적용됩니다.

참고: 신뢰 당사자 측에서는 이 쿠키 데이터를 전역 수준이 아니라 파트너 관계 수준에서 구성합니다.

쿠키 설정을 변경하려면

1. Administrative UI 에 로그인합니다.
2. "인프라" 탭에서 "배포 설정"을 선택합니다.

   "배포 설정 구성" 대화 상자가 나타납니다.
3. (선택 사항) 필요에 따라 "쿠키 설정" 섹션의 모든 설정을 수정합니다.

   참고: "도움말"을 클릭하면 필드, 컨트롤 및 해당되는 각 요구 사항에 대한 설명을 볼 수 있습니다.
4. 섹션 오른쪽에 있는 "저장"을 클릭합니다.

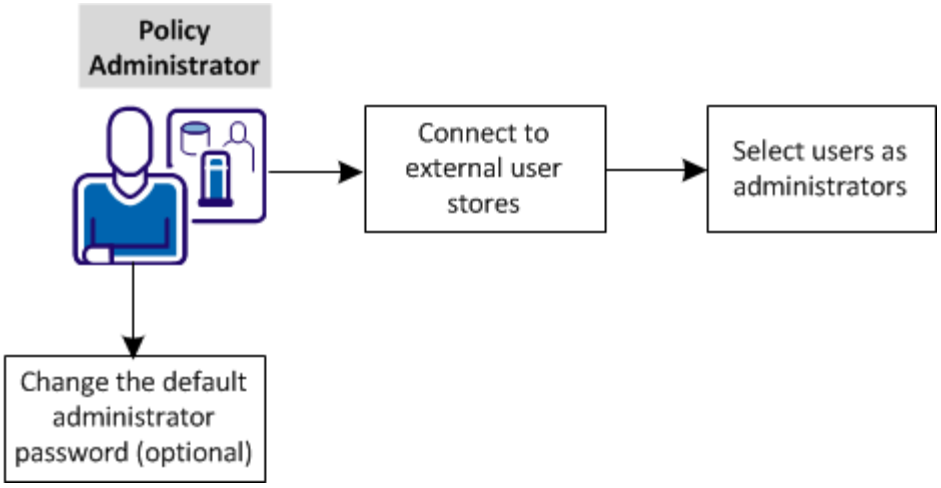
페더레이션 시스템 관리자를 구성하는 방법

회사 내에서 페더레이션 관리의 서로 다른 측면을 여러 명의 관리자가 담당할 수 있습니다. CA SiteMinder?Federation Standalone 관리 업무를 조직 내의 여러 사람에게 할당하여 의무와 책임을 분담할 수 있습니다.

CA SiteMinder?Federation Standalone 관리를 담당하는 기본 관리자 계정이 항상 있습니다. 필요에 따라 새로운 관리자를 추가한 후에는 기본 관리자 계정을 사용하지 않도록 설정하십시오.

새로운 관리 사용자는 Administrative UI 를 통해 생성하고 유지 관리하십시오.

다음 그림에서는 관리자 구성을 위한 구성 태스크를 보여 줍니다.



다음 태스크를 완료하십시오.

1. 외부 사용자 디렉터리에 연결합니다 (페이지 405).
2. 사용자를 관리자로 선택합니다 (페이지 407).
3. 기본 관리자 암호를 변경합니다(선택 사항) (페이지 408).

외부 사용자 저장소에 연결

LDAP 및 ODBC 외부 사용자 저장소에 대한 연결을 생성할 수 있습니다. 이 단계는 여러 명의 관리자를 구성하기 전에 반드시 수행해야 합니다.

페더레이션 시스템에는 LDAP 와 ODBC 두 가지 유형의 디렉터리가 지원됩니다.

다음 단계를 수행하십시오.

1. "사용자 디렉터리" 탭을 클릭합니다.
2. "LDAP 에 연결" 또는 "ODBC 에 연결"을 클릭합니다.

   "작업", "수정"을 선택하면 기존 디렉터리 연결의 구성을 확인할 수 있습니다.

3. 각 섹션에서 필수 설정을 모두 구성합니다. 빨간색 점은 필수 매개 변수를 나타냅니다.

   참고: "도움말"을 클릭하면 필드, 컨트롤 및 해당되는 각 요구 사항에 대한 설명을 볼 수 있습니다.

4. "유니버설 ID 특성"(LDAP) 또는 "유니버설 ID 열"(ODBC)에 값을 입력합니다. 이 값은 관리자를 여러 명 구성할 경우 필수 항목입니다.

   유니버설 ID 값은 디렉터리 내의 개별 사용자를 식별할 수 있는 고유한 값이어야 합니다. 예를 들어 사용자마다 uid 가 있으므로 uid 를 LDAP 디렉터리의 유니버설 ID 로 입력할 수 있습니다. 직책과 같이 여러 사용자가 공통으로 가질 수 있는 특성은 유니버설 ID 로 적합하지 않습니다.

5. LDAP 디렉터리의 경우 "사용자 DN 조회 시작" 필드와 "사용자 DN 조회 끝"의 값을 지정합니다. 예를 들면 다음과 같습니다.

   사용자 DN 조회 시작

   (uid=

   사용자 DN 조회 끝

)

6. "연결 테스트"를 클릭하여 연결이 올바른지 확인합니다.

   "콘텐츠 보기"를 클릭하여 사용자 디렉터리의 콘텐츠를 볼 수 있습니다.

   참고:

   - LDAP 디렉터리 연결의 경우 "콘텐츠 보기" 단추는 "검색 루트", "사용자 DN 조회 시작", "사용자 DN 조회 끝", "유니버설 ID 특성" 값이 설정된 경우에만 표시됩니다.
   - ODBC 디렉터리 연결의 경우 "콘텐츠 보기" 단추는 "유니버설 ID 열" 값이 설정된 경우에만 표시됩니다.

7. "저장"을 클릭합니다.

   설정이 올바르면 "사용자 디렉터리 보기" 대화 상자로 리디렉션됩니다.

   디렉터리 연결이 구성되었습니다.

관리자로 사용자 선택

외부 사용자 저장소에 연결된 후에는 관리자 역할을 할 사용자를 선택해야 합니다.

다음 단계를 수행하십시오.

1. Administrative UI 에 로그온합니다.
2. "인프라", "관리자"로 이동합니다.
3. "관리 인증 구성"을 선택합니다.
4. 구성 마법사에 따라 다음 태스크를 완료합니다.
   - 외부 사용자 저장소를 선택합니다.
   - 하나 이상의 사용자를 관리자로 선택합니다.
   - 각 관리자의 액세스 권한 유형을 결정합니다. 옵션은 다음과 같습니다.
     - 슈퍼 사용자
     - 페더레이션 관리자
     - 읽기 전용 사용자

     참고: 각 권한에 대한 설명을 보려면 "도움말"을 클릭하십시오.
   - 기본 관리자를 사용할지 여부를 결정합니다. 중앙 관리자 계정을 사용하지 않으려면 기본 관리자를 사용하지 않도록 설정합니다. 기본 관리자가 없으면 감사할 수 있는 개별 관리자에게 의존해야 합니다.
5. Administrative UI 에서 로그아웃한 후 변경 내용이 적용될 때까지 몇 분 동안 기다립니다.
6. 새 관리자의 자격 증명을 사용하여 Administrative UI 에 다시 로그인합니다.
7. "관리자" 페이지로 돌아가서 표시되는 관리자 목록을 살펴봅니다.
8. (선택 사항) "작업" 메뉴에서 항목을 수정하거나 봅니다.

   관리자의 권한을 변경하고 관리자를 사용하거나 사용하지 않도록 설정할 수 있습니다.

이제 여러 명의 관리자에게 페더레이션 관리 태스크를 나누어 할당할 수 있습니다.

기본 관리자 암호 변경(선택 사항)

보안상의 이유로 기본 관리자의 Administrative UI 액세스 암호를 변경하십시오. 이 태스크는 선택 사항입니다.

두 가지 방법으로 관리자 암호를 변경할 수 있습니다.

- UI 에서 암호를 수정합니다.
- 명령줄에서 암호를 수정합니다.

  관리자 사용자 계정이 잠겨 있거나 관리자를 사용할 수 없는 경우에는 명령줄에서 암호를 재설정하십시오.

UI 에서 기본 관리자 암호 변경

Administrative UI 에서 관리자 암호를 변경할 수 있습니다.

다음 단계를 수행하십시오.

1. "인프라" 탭에서 "암호"를 선택합니다.

   "관리자 암호 변경" 대화 상자가 나타납니다.
2. 이전 암호와 새 암호 필드를 완료합니다.
3. "제출"을 클릭합니다.
4. 시스템을 다시 시작합니다.

기본 관리자 암호가 변경되어 활성화됩니다.

명령줄에서 기본 관리자 암호 변경

관리자 사용자 계정이 잠겨 있거나 사용 불가능한 경우에는 XPSConfig 유틸리티를 사용하여 명령줄에서 관리자 암호를 변경해야 합니다.

다음 단계를 수행하십시오.

1. 페더레이션 시스템에서 명령 창을 엽니다.
2. XPSConfig 를 입력합니다.

   UNIX 플랫폼의 경우 여기에 나온 대로 유틸리티 이름을 입력합니다. 이름은 대/소문자를 구분합니다.

   "Products Menu"(제품 메뉴)가 표시됩니다.
3. FED 를 입력하여 페더레이션 제품을 선택합니다.

4. 1 을 입력합니다.

   옵션 1 이 암호에 해당하는 옵션입니다.

5. C 를 입력하여 암호의 값을 변경합니다.
6. 프롬프트에서 새 값을 입력합니다.
7. Q 를 입력하여 저장한 후 종료합니다.

새 암호가 활성화됩니다.

관리자 세션 관리

관리 세션은 한 번에 하나만 활성화할 수 있습니다. 하나의 관리 세션은 관리자가 새 세션을 설정하려고 할 때 페더레이션 개체의 동시 편집을 방지합니다. 관리 세션이 설정된 후에 새로운 로그인 시도가 있으면 경고 메시지가 표시됩니다.

이 경고 메시지는 세션이 이미 있다는 사실을 관리자에게 알려 줍니다. 관리자가 계속해서 동일한 자격 증명을 사용하여 로그인을 진행하면 기존 세션이 무효화되고 저장하지 않은 데이터가 모두 손실됩니다. 첫 번째 세션이 무효화되면 첫 번째 세션의 관리자가 로그아웃됩니다. 관리자가 구성 작업을 시도하면 시스템에서 해당 관리자는 로그인 대화 상자로 리디렉션됩니다.

다음 섹션에서는 이미 설정되어 있는 관리자 세션과 동일한 자격 증명을 사용하여 특정 관리자가 로그인을 시도할 경우의 결과에 대해 설명합니다.

관리 세션 상호 작용

다음과 같은 시나리오에서는 관리 세션 충돌이 발생합니다.

동일한 자격 증명으로 로그인 시도

관리자가 CA SiteMinder® Federation Standalone 에 로그인한 후 다른 관리자 또는 동일한 관리자가 다른 브라우저 세션에서 첫 번째 로그인과 같은 자격 증명을 사용하여 로그인을 시도합니다.

CA SiteMinder® Federation Standalone 에서 경고 대화 상자를 표시하지만 두 번째 사용자가 계속해서 로그인을 시도하고 CA SiteMinder® Federation Standalone 에서 첫 번째 세션을 무효화합니다. 첫 번째 세션의 관리자가 개체를 수정하려고 하면 CA SiteMinder® Federation Standalone 에서는 새 세션 때문에 기존 세션이 무효화되었다는 경고를 표시하고 첫 번째 관리자는 로그아웃됩니다.

두 번째 사용자는 필요에 따라 로그인하지 않을 수 있습니다.

브라우저 세션 종료 후 관리자가 로그아웃하지 않음

관리자가 CA SiteMinder® Federation Standalone 에 로그인한 후 로그아웃하지 않고 브라우저 세션을 닫거나, 브라우저 세션이 예기치 않게 종료되어 관리자가 직접 로그아웃하지 못한 상태입니다. 이 경우 다른 관리자가 다른 브라우저 세션에서 첫 번째 관리자와 같은 자격 증명을 사용하여 로그인합니다.

CA SiteMinder® Federation Standalone 에서 경고 대화 상자를 표시하지만 두 번째 사용자가 계속해서 로그인을 시도하고, 첫 번째 세션은 무효화됩니다.

첫 번째 세션의 관리자가 브라우저 세션을 다시 시작하여 개체를 수정하려고 시도하면 새 세션 때문에 기존 세션이 무효화되었다는 경고가 표시됩니다. 이 경우 첫 번째 세션은 브라우저가 닫힐 때 무효화되었습니다.

참고: 브라우저 세션이 예기치 않게 종료된 경우 사용자가 다시 시작할 수 있게 허용하는 기능은 일부 브라우저에서만 지원됩니다. 이러한 브라우저에서는 CA SiteMinder® Federation Standalone 이 기존 세션이 무효화되었다는 경고를 표시하지 않습니다.

관리 사용 안 함

관리자가 CA SiteMinder® Federation Standalone 에 로그인한 후 "시스템 설정"에서 관리 기능을 사용하지 않도록 설정합니다. 다른 관리자가 첫 번째 관리자와 같은 자격 증명을 사용하여 로그인을 시도합니다. CA SiteMinder® Federation Standalone 에서 경고 대화 상자를 표시하지만 두 번째 사용자가 계속해서 로그인을 시도합니다. CA SiteMinder® Federation Standalone 은 첫 번째 세션을 무효화합니다.

로그아웃하거나 브라우저를 닫지 않은 첫 번째 관리자가 관리 기능을 다시 사용하도록 설정하려고 합니다. 이 경우 CA SiteMinder® Federation Standalone 에서는 세션이 유효하지 않다는 메시지를 첫 번째 관리자에게 표시하고 해당 관리자를 로그아웃 처리합니다.

UI 관리 사용 안 함

"UI 설정" 그룹 상자에서는 로컬 호스트에서 CA SiteMinder?Federation Standalone 관리 기능을 사용하지 않도록 설정했다가 다시 사용하도록 설정할 수 있습니다.

UI 관리를 사용하지 않도록 설정하는 기능은 장애 조치를 지원하기 위해 CA SiteMinder?Federation Standalone 시스템 두 개가 설정된 경우에 유용합니다. CA SiteMinder?Federation Standalone 관리는 기본 CA SiteMinder?Federation Standalone 시스템에서만 수행할 수 있습니다. 이 경우 구성을 보조 CA SiteMinder?Federation Standalone 시스템에 내보낼 수 있습니다.

UI 를 사용하지 않도록 설정하는 기능은 기본 시스템에서만 관리가 수행되도록 보조 시스템에서 관리 기능을 사용하지 않도록 설정할 때 사용하십시오.

다음 단계를 수행하십시오.

1. Administrative UI 에 로그인합니다.
2. "인프라", "시스템 설정"으로 이동합니다.
3. "UI 설정" 그룹 상자에서 "관리 사용 안 함"을 클릭합니다.

   관리를 사용하지 않도록 설정하면 모든 관리 작업을 사용할 수 없습니다.

중요! 변경 내용은 작업을 확인하는 즉시 적용됩니다.

관리를 사용하지 않도록 설정하면 "관리 사용 안 함" 대화 상자가 표시되고 UI 의 다른 모든 부분을 사용할 수 없게 됩니다. 이후에 로그인을 시도하면 경고 메시지와 함께 관리를 다시 사용하도록 설정하는 단추가 표시됩니다.

관리를 다시 사용하도록 설정하려면

"관리 사용 안 함" 대화 상자에서 "관리 사용"을 클릭합니다.

그러면 UI 의 모든 부분이 다시 활성화됩니다.

제 27 장: 페더레이션 시스템의 SSL 관리

이 섹션은 다음 항목을 포함하고 있습니다.

Apache 웹 서버 및 UI 에 대한 SSL 관리

SSL 은 다음과 같은 용도로 사용할 수 있습니다.

- SSL 연결을 통한 페더레이션 트래픽 처리
- 백 채널을 통한 HTTP 아티팩트 싱글 사인온의 보안 통신
- Administrative UI 에 대한 보안 액세스

포함된 Apache 웹 서버는 페더레이션 시스템에서 SSL 페더레이션 트래픽을 처리하고 HTTP 아티팩트 싱글 사인온을 위한 백 채널의 보안을 보장할 수 있게 도와줍니다. 포함된 Tomcat 웹 서버를 사용하면 UI 에 안전하게 액세스할 수 있습니다.

Apache 및 Tomcat 웹 서버에 대해 SSL 을 사용하려면 다음 프로세스를 완료하십시오.

1. 서버 인증서를 위한 인증서 요청을 생성합니다.
2. CA(인증 기관)에서 발급한 인증서를 가져옵니다.
3. Administrative UI 에서 SSL 을 활성화합니다. "인프라", "SSL 구성"에서 설정을 찾습니다.

참고: FIPS 마이그레이션 또는 FIPS 전용 모드에서 작동하는 CA SiteMinder® Federation Standalone 설치에는 인증서에 사용할 수 있는 FIPS 호환 암호화 키 알고리즘이 있습니다.

Apache 웹 서버 및 UI 에 대해 SSL 을 사용하는 방법

포함된 Apache 웹 서버와 Administrative UI 에 대해 SSL 을 사용하도록 설정하는 절차는 동일합니다.

- 다음과 같은 작업을 수행하려는 경우 포함된 Apache 웹 서버에 SSL 을 사용하십시오.
  - SSL 연결을 통해 페더레이션 트래픽 관리
  - 백 채널을 통한 아티팩트 싱글 사인온의 보안 통신

  SSL 포트 번호는 구성 마법사를 실행하는 과정에서 지정합니다.
- UI 에 대한 연결 보안을 유지하기 위해 Administrative UI 에 대해 SSL 을 사용하도록 설정하십시오.

  SSL 을 사용하면 CA SiteMinder® Federation Standalone 이 서버 인증서에 사용할 FIPS 호환 개인 키를 생성합니다.

참고: SSL 이 사용되도록 설정하는 경우 "기준 URL" 매개 변수를 포함한 모든 서비스의 모든 URL 이 영향을 받습니다. 즉, 모든 서비스 URL 이 https://로 시작해야 합니다.

SSL 통신을 사용하려면

1. 서버 인증서를 요청합니다.
2. 서버 인증서에 서명하는 CA 인증서를 지정합니다.
3. 서명된 인증서를 시스템에 업로드합니다.

   인증서가 성공적으로 로드되면 CA SiteMinder® Federation Standalone 이 SSL 연결을 활성화합니다.

이러한 필수 단계 이외에 다음을 수행할 수 있습니다.

- 인증서 서명 요청을 가져옵니다.
- SSL 을 사용하지 않도록 설정합니다.
- 시스템에서 SSL 구성을 삭제합니다.

SSL 서버 인증서 요청

SSL 연결을 설정하는 첫 번째 단계는 서버 인증서 요청을 완료하는 것입니다. 완료된 요청을 트러스트된 CA(인증 기관)에 보내면 CA 에서 서명된 서버 인증서를 반환합니다.

중요! SSL 서버 인증서를 요청하십시오.

다음 단계를 수행하십시오.

1. Administrative UI 에서 "인프라", "SSL 구성"을 선택합니다.

   "SSL 구성" 대화 상자가 열립니다. "SSL 구성 상태" 필드에 **서버 인증서 요청되지 않음** 상태로 표시됩니다.

2. "요청"을 클릭하여 인증서 요청을 생성합니다.

3. "인증서 요청" 대화 상자의 필드를 완성한 후 "저장"을 클릭합니다.

   일부 필드에는 필수 값이 이미 할당되어 있습니다. "요청자 이름" 필드에는 권장되는 기본값이 있지만 이 값은 변경할 수 있습니다. "요청자 이름" 값은 CA SiteMinder® Federation Standalone 이 배포되어 있는 서버와 관련된 정규화된 도메인 이름이어야 합니다.

   참고: "도움말"을 클릭하면 필드, 컨트롤 및 해당되는 각 요구 사항에 대한 설명을 볼 수 있습니다.

인증서 요청이 생성되면 CA SiteMinder?Federation Standalone 이 개인 키를 생성합니다. 이 개인 키는 내부 파일 위치에 저장됩니다.

요청이 생성된 후 인증서에 서명할 지정된 CA 에 서버 인증서 요청을 보내십시오.

생성된 인증서 요청에 따라 인증 기관에서 인증서를 발급합니다. 인증서의 유효 기간은 다음 값 중 하나와 같습니다.

- 인증 기관 기본값
- 요청자와 인증 기관 사이의 비즈니스 계약에 기반한 값

서명된 서버 인증서 업로드

인증서 요청을 완료했는데 "SSL 구성 상태" 필드에 **서버 인증서가 요청되었지만 서명되지 않음**, 즉 인증서 요청에 서명해야 한다는 내용이 표시됩니다. CA SiteMinder?Federation Standalone 에서는 base-64 로 인코딩된 PEM 인증서 또는 전체 PKCS #7 인증서/체인 응답을 지원합니다.

서명된 인증서를 CA 에서 받은 후에는 해당 인증서를 저장소 위치에 업로드해야 합니다.

참고: "도움말"을 클릭하면 필드, 컨트롤 및 해당되는 각 요구 사항에 대한 설명을 볼 수 있습니다.

서명된 서버 인증서를 업로드하려면

1. 요청을 시작한 것과 동일한 SSL 구성에서 작업을 시작합니다.
2. "서명된 인증서 응답" 필드에서 서명된 인증서 응답을 선택합니다. 파일을 찾으려면 "찾아보기"를 클릭하십시오.

   참고: SSL 에서는 키와 인증서를 한 쌍만 지원하기 때문에 SSL 기능에는 키/인증서 쌍이 하나만 필요합니다.
3. SSL 인증서에 서명한 CA 를 "CA 인증서" 필드의 풀다운 메뉴에서 확인합니다.

   CA 인증서가 키 저장소에 없으면 SSL 인증서 요청에 서명하는 데 사용된 CA 인증서의 복사본을 가져옵니다. "가져오기"를 클릭한 후 가져오기 단계를 완료하여 인증서를 가져옵니다.
4. "적용"을 클릭하여 서버 인증서를 CA SiteMinder® Federation Standalone 에 업로드합니다.

   확인 메시지가 표시되고, 인증서 업데이트를 나타내도록 SSL 구성이 변경됩니다.
5. 운영 환경에 맞게 페더레이션 서비스를 다시 시작합니다.

   - **Windows**

     다음과 같이 중지 및 시작 바로 가기를 사용합니다. 로컬 관리자가 아닌 네트워크 사용자로 로그인한 경우에는 바로 가기를 마우스 오른쪽 단추로 클릭하고 "관리자 권한으로 실행"을 선택하십시오.

     a. "시작", "모든 프로그램", "CA", "Federation Standalone", "서비스 중지"

     b. "시작", "모든 프로그램", "CA", "Federation Standalone", "서비스 시작"

- **UNIX**

  a. 명령 창을 엽니다.

  b. 다음 스크립트를 실행합니다.

  federation_install_dir/fedmanager.sh stop

  federation_install_dir/fedmanager.sh startssl

  참고: 서비스를 중지했다가 루트 사용자로 시작하지 마십시오.

시스템에 서버 인증서를 업로드하면 CA SiteMinder?Federation Standalone 에서 인증서를 업데이트하고 SSL 을 활성화합니다. 인증서가 성공적으로 업로드되면 "SSL 구성 상태"가 **SSL 활성화됨**으로 표시됩니다. 그리고 구성 그룹 상자의 단추가 "비활성화"로 바뀝니다.

UI 에는 업로드한 인증서가 FIPS 승인을 받았는지 여부도 표시됩니다.

SSL 비활성화

SSL 이 더 이상 필요하지 않을 경우 SSL 구성을 비활성화할 수 있습니다. 예를 들어 백 채널 인증이 더 이상 필요하지 않거나, UI 에 SSL 연결을 더 이상 사용하지 않으려면 SSL 을 비활성화할 수 있습니다.

참고: SSL 이 사용되도록 설정한 상태로 Windows 시스템을 다시 구성하려면 시스템을 다시 구성하기 전에 SSL 구성을 비활성화하십시오. 재구성이 완료되면 SSL 을 다시 활성화하십시오.

다음 단계를 수행하십시오.

1. "SSL 구성" 대화 상자에서 시작합니다.
2. 포함된 웹 서버 또는 관리 UI 섹션에서 "비활성화"를 클릭합니다.

   SSL 을 비활성화할지 묻는 확인 메시지가 표시됩니다.
3. "예"를 클릭하여 비활성화를 완료합니다.
4. Administrative UI 의 경우에만 tomcat.keystore 파일을 수동으로 삭제합니다. 이 파일은 다음 디렉터리에 있습니다.

   federation_install_dir/secure-proxy/SSL/keys

   Administrative UI 에서 SSL 을 비활성화해도 해당 키 저장소 파일은 삭제되지 않습니다. 어떤 이유로든 UI SSL 인증서를 변경하면 인증서가 업데이트되지 않기 때문에 CA SiteMinder® Federation Standalone 이 잘못된 인증서를 사용하게 됩니다. Tomcat 키 저장소를 삭제하면 SSL 인증서에 대한 모든 업데이트 사항이 반영됩니다.
5. 운영 환경에 맞게 페더레이션 서비스를 다시 시작합니다.

   - **Windows**

     다음과 같이 중지 및 시작 바로 가기를 사용합니다. 로컬 관리자가 아닌 네트워크 사용자로 로그인한 경우에는 바로 가기를 마우스 오른쪽 단추로 클릭하고 "관리자 권한으로 실행"을 선택하십시오.

     a. "시작", "모든 프로그램", "CA", "Federation Standalone", "서비스 중지"

     b. "시작", "모든 프로그램", "CA", "Federation Standalone", "서비스 시작"

   - **UNIX**

     a. 명령 창을 엽니다.

     b. 다음 스크립트를 실행합니다.

     federation_install_dir/fedmanager.sh stop

     federation_install_dir/fedmanager.sh start

     참고: 서비스를 중지했다가 루트 사용자로 시작하지 마십시오.

이제 SSL 연결은 더 이상 활성 상태가 아니며 "SSL 구성 상태" 설정이 **서버 인증서가 CA 서명됨, SSL 준비됨**으로 바뀝니다. 인증서와 키 파일은 SSL 을 다시 사용할 수 있도록 보관됩니다.

SSL 다시 활성화

어떠한 이유로든 SSL 을 비활성화하는 경우 다시 활성화하십시오. SSL 을 사용하면 CA SiteMinder?Federation Standalone 이 서버 인증서에 사용할 FIPS 호환 개인 키를 생성합니다.

참고: 상태 설정이 **서버 인증서가 CA 서명됨, SSL 준비됨**으로 표시되면 SSL 연결을 활성화하십시오.

다음 단계를 수행하십시오.

1. "SSL 구성" 대화 상자에서 시작합니다.
2. "포함된 웹 서버 SSL 구성" 그룹 상자에서 "활성화"를 클릭합니다.

   "SSL 구성 상태" 설정이 **SSL 활성화됨**으로 변경되고 대화 상자에 확인 메시지가 표시됩니다.
3. 운영 환경에 맞게 페더레이션 서비스를 다시 시작합니다.
   - **Windows**

     다음과 같이 중지 및 시작 바로 가기를 사용합니다. 로컬 관리자가 아닌 네트워크 사용자로 로그인한 경우에는 바로 가기를 마우스 오른쪽 단추로 클릭하고 "관리자 권한으로 실행"을 선택하십시오.

     a. "시작", "모든 프로그램", "CA", "Federation Standalone", "서비스 중지"

     b. "시작", "모든 프로그램", "CA", "Federation Standalone", "서비스 시작"
   - **UNIX**

     a. 명령 창을 엽니다.

     b. 다음 스크립트를 실행합니다.

     federation_install_dir/fedmanager.sh stop

     federation_install_dir/fedmanager.sh startssl

     참고: 서비스를 중지했다가 루트 사용자로 시작하지 마십시오.

이제 SSL 가 사용되도록 설정되었습니다. 인증서가 만료되기 전까지는 SSL 구성을 수정할 필요가 없습니다.

SSL 에 사용할 인증서 서명 요청 대체 또는 다시 제출

Apache 서버 또는 UI 에서 사용 중인 개인 키/인증서 쌍과 관련된 인증서 서명 요청의 복사본을 가져올 수 있습니다. 인증서 서명 요청 복사본을 가져오는 기능은 요청 파일을 삭제했거나 저장하지 않은 경우에 유용합니다. 그뿐만 아니라 요청 복사본은 나중에 서명된 인증서가 만료되기 이전에 요청을 다시 제출할 때도 유용합니다.

"가져오기" 기능을 사용하면 인증서 서명 요청 복사본을 가져올 수 있습니다.

참고: "가져오기" 옵션은 인증서를 요청했고 "다시 시작" 단추를 사용하여 SSL 구성을 삭제하지 않은 경우에만 사용할 수 있습니다.

인증서 서명 요청을 가져오려면

1. UI 에서 "인프라", "SSL 구성"을 선택합니다.

   "SSL 구성" 대화 상자가 열립니다.

2. "가져오기"를 클릭합니다.

   파일 다운로드 대화 상자에 파일을 열거나 저장할지 묻는 메시지가 표시됩니다.

3. 파일을 저장합니다.

서명 요청 가져오기가 완료되고 "SSL 구성" 대화 상자가 다시 표시됩니다.

포함된 Apache 서버와 UI 에서 SSL 제거

- 다음과 같은 경우 포함된 Apache 웹 서버에서 SSL 을 제거하십시오.
  - 아티팩트 싱글 사인온을 위해 백 채널 연결이 더 이상 필요하지 않은 경우
  - SSL 을 더 이상 사용하지 않으려는 경우
- UI 에 SSL 연결을 더 이상 사용하지 않으려면 UI 연결에서 SSL 을 제거하십시오.

"다시 시작" 기능을 통해 기존 SSL 구성을 비활성화하고 SSL 구성과 관련된 모든 파일을 삭제할 수 있습니다. 이 기능을 사용하면 개인 키와 서버 인증서 및 원래 서버 요청 파일이 삭제됩니다.

SSL 을 사용하지 않고 관련 파일을 제거하려면

1. Administrative UI 에 로그인합니다.
2. "인프라", "SSL 구성"을 선택합니다.

   "SSL 구성" 대화 상자가 열립니다.
3. SSL 이 필요하지 않은 기능의 그룹 상자에서 "다시 시작"을 클릭합니다.

   다시 시작을 확인하는 메시지가 표시됩니다.
4. "예"를 클릭합니다.

   그러면 SSL 구성이 시스템에서 제거됩니다.
5. 운영 환경에 맞게 페더레이션 서비스를 다시 시작합니다.

   - **Windows**

     다음과 같이 중지 및 시작 바로 가기를 사용합니다. 로컬 관리자가 아닌 네트워크 사용자로 로그인한 경우에는 바로 가기를 마우스 오른쪽 단추로 클릭하고 "관리자 권한으로 실행"을 선택하십시오.

     a. "시작", "모든 프로그램", "CA", "Federation Standalone", "서비스 중지"

     b. "시작", "모든 프로그램", "CA", "Federation Standalone", "서비스 시작"

   - **UNIX**

     a. 명령 창을 엽니다.

     b. 다음 스크립트를 실행합니다.

     federation_install_dir/fedmanager.sh stop

     federation_install_dir/fedmanager.sh start

     참고: 서비스를 중지했다가 루트 사용자로 시작하지 마십시오.

   서비스를 다시 시작하면 Apache 웹 서버와 UI 가 SSL 을 사용하지 않는 작업 모드로 전환됩니다. 이후에 CA SiteMinder® Federation Standalone 으로 들어오는 HTTPS 요청이 실패합니다. SSL 이 제거되면 모든 서비스 URL 이 HTTP 로 시작되어야 합니다.

SSL 키 및 인증서를 마이그레이션하는 방법

CA SiteMinder?Federation Standalone r12.52 SP1 의 경우 포함된 Apache 및 Tomcat 서버에 대한 SSL 키와 인증서 파일이 암호화됩니다. 12.0 및 12.0 SP1 릴리스에서는 이러한 파일이 암호화되지 않습니다. 암호화된 파일에 대한 키/인증서 쌍을 새로 구매하는 것을 방지하려면 기존 키 또는 인증서 파일을 r12.0/r12.0 SP1 에서 r12.52 SP1 로 마이그레이션하십시오. 이러한 파일을 마이그레이션하지 않고 백업 용도로 내보낼 수도 있습니다.

중요! r12.1 이전 페더레이션 시스템에서는 포함된 Tomcat 서버가 자체 서명된 인증서를 사용합니다. r12.52 SP1 로의 마이그레이션에는 이 자체 서명된 인증서를 사용할 수 없습니다. 서명된 인증서를 구매하고 Tomcat SSL 구성을 서명된 인증서로 업그레이드하십시오.

Apache 의 경우 r12.0 부터 SSL 연결을 위한 파일을 마이그레이션할 수 있습니다. Tomcat 의 경우 릴리스 12.0 에서는 자체 서명된 인증서로 Tomcat 키 저장소를 보호하기 때문에 r12.1 이상의 파일만 마이그레이션할 수 있습니다. r12.1 부터 페더레이션 시스템은 인증 기관이 인증서를 서명해야 합니다.

SSL 키 및 인증서 파일은 다음과 같은 경우에 유용합니다.

- 기존 시스템을 업그레이드하는 대신 새 시스템에 있는 다른 버전의 CA SiteMinder® Federation Standalone 으로 이동하는 경우. SSL 키 또는 인증서를 기존 시스템에서 새 시스템으로 마이그레이션합니다.
- SSL 키와 인증서를 클러스터의 한 시스템에서 다른 시스템으로 마이그레이션하는 경우. 마이그레이션을 수행하면 키와 인증서를 재사용할 수 있습니다. 예를 들어 부하 분산 장치가 SSL 요청을 클러스터의 페더레이션 시스템으로 전달하는 경우 각 시스템이 동일한 키와 인증서를 사용해야 합니다. 따라서 키와 인증서를 한 시스템에서 다른 시스템으로 마이그레이션합니다.

참고: 페더레이션 12.0 시스템을 r12.52 SP1 로 업그레이드하면 설치 관리자는 자동으로 Apache 및 Tomcat SSL 키와 인증서 파일을 암호화된 파일로 업그레이드합니다. 마이그레이션에는 이 자동화가 적용되지 않습니다.

인증서 및 개인 키 파일은 다음과 같습니다.

Apache

- server.key 파일에 개인 키가 포함되어 있습니다.
- server.cert 파일에 서버 인증서가 포함되어 있습니다.

Tomcat

- r12.0 의 경우 tomcat.keystore 파일에 자체 서명된 인증서가 포함되어 있습니다. r12.1x 의 경우 tomcat.keystore 파일에 CA 서명된 인증서 및 개인 키 쌍이 포함되어 있습니다.

이러한 파일을 마이그레이션하거나 내보내려면 migratessl 이라는 CA SiteMinder?Federation Standalone SSL 유틸리티를 사용하십시오. 마이그레이션 유틸리티는 Windows 시스템의 경우 배치 파일로, UNIX 시스템의 경우 셸 스크립트의 형태로 제품에 포함되어 있습니다. 이 유틸리티는 federation_install_dir/bin 폴더에 설치되어 있습니다.

SSL 파일을 마이그레이션하는 프로세스는 다음과 같습니다.

1. 키 및 인증서 파일을 기존 r12 페더레이션 시스템에서 r12.52 SP1 페더레이션 시스템의 원하는 위치로 복사합니다.
2. 키와 인증서 파일을 복사한 위치로 migratessl 도구를 복사합니다.
3. 서명된 인증서를 마이그레이션하는 경우 SSL 인증서에 서명한 인증 기관 인증서를 내보냅니다. 마이그레이션을 계속하기 전에 먼저 CA 인증서를 가져옵니다.

r12 시스템에서 키 및 인증서 파일 복사

SSL 마이그레이션 도구를 사용하려면 먼저 마이그레이션하거나 내보낼 원본 CA SiteMinder?Federation Standalone 시스템의 키 및 인증서 파일을 수집한 후 복사하십시오.

SSL 키 및 인증서 파일을 복사하려면

1. 기존 CA SiteMinder® Federation Standalone 시스템에서 파일을 찾습니다.

   Apache SSL 키 및 인증서 파일은 다음 위치에 있습니다.

   - *federation_install_dir*/secure-proxy/SSL/keys/server.key
   - *federation_install_dir*/secure-proxy/SSL/certs/server.crt

Tomcat SSL 키 저장소 파일의 위치는 다음과 같습니다.

- *federation_install_dir*/secure-proxy/SSL/keys/tomcat.keystore

2. 키 및 인증서 파일을 새 CA SiteMinder® Federation Standalone 컴퓨터의 원하는 위치로 복사합니다.

SSL 마이그레이션 도구를 키/인증서 파일과 동일한 폴더에 복사

SSL 마이그레이션 도구에는 CA SiteMinder?Federation Standalone 12.1 SP3 과 함께 배포되는 소프트웨어가 필요합니다. CA SiteMinder?Federation Standalone 12.1 SP3 제품이 설치된 컴퓨터에서 도구를 실행하십시오. 도구는 마이그레이션될 파일을 복사한 폴더와 동일한 폴더에 있어야 합니다.

SSL 유틸리티 도구를 복사하려면

1. r12.52 SP1 시스템에서 *federation_install_dir*/bin 으로 이동합니다.
2. migratessl 파일(.bat 또는 .sh)을 키 및 인증서 파일을 복사한 r12.52 SP1 시스템의 위치로 복사합니다.

SSL 키 및 인증서 마이그레이션 또는 내보내기

migratessl 유틸리티를 실행하여 SSL 키 또는 인증서 파일 마이그레이션을 완료하십시오.

다음 단계를 수행하십시오.

1. 마이그레이션하려는 SSL 인증서에 원래 서명한 인증 기관 인증서를 가져옵니다.
   a. 마이그레이션하려는 원래 시스템에서 Administrative UI 를 사용하여 CA 인증서를 내보냅니다.
   b. 마이그레이션하는 새 시스템에서 Administrative UI 를 사용하여 CA 인증서를 가져옵니다.
2. 기존 키 또는 인증서 파일을 복사한 새 시스템에서 명령 창을 엽니다.
3. 구성 요소를 복사한 폴더로 이동합니다.

4. migratessl 명령을 필요한 명령 인수와 함께 지정합니다. 모든 옵션을 보려면 마이그레이션 도구 명령 인수 목록을 참조하십시오.

   예

   - Apache SSL 연결에 사용할 SSL server.key 를 마이그레이션하려면 다음을 입력하십시오.

   ```
   migratessl.bat -op migrate -keytype Apache
   -sourcefile server.key -certfile server.crt
   -sourcever 12.0 -sourceos Windows -oldpwd admin1
   -newpwd admin2 -issueralias trustedca
   ```

   - Tomcat SSL 연결에 사용되는 키/인증서 파일을 마이그레이션하려면 다음을 입력하십시오.

   ```
   migratessl.sh -op migrate -keytype Tomcat
   -sourcefile tomcat.keystore -sourcever 12.1
   -sourceos UNIX -issueralias trustedca
   -oldpwd admin1 -newpwd admin2
   ```

   - Tomcat SSL 연결에 사용되는 키/인증서 파일을 내보내려면 다음을 입력하십시오.

   ```
   migratessl.sh -op export -keytype Tomcat
   -sourcefile tomcat.keystore -sourcever 12.1
   -sourceos UNIX  -dest ca/federationmgr/secure-proxy/
   SSL/keys/ -oldpwd admin1 -newpwd admin2
   ```

전체 구성 마이그레이션의 일부로 SSL 키와 인증서를 마이그레이션하는 경우에는 파트너 관계를 다시 활성화하여 마이그레이션 프로세스를 완료하십시오.

SSL 마이그레이션 도구 명령 인수

명령줄에서 migratessl 도구가 호출됩니다. 명령을 입력할 때 다음 사항에 유의하십시오.

- 각 명령 인수에 값을 하나씩만 입력합니다(Help 플래그 제외).
- 디렉터리 경로처럼 공백이 있는 값은 큰따옴표로 묶습니다.

명령 인수	의미
-op	마이그레이션 또는 내보내기 기본값: Migrate -certfile 인수를 지정한 경우 도구는 Apache 용으로 내보낼 때 server.key 파일 및 server.crt 파일을 내보냅니다. Tomcat 의 경우 도구는 PKCS#12 키/인증서 파일인 tomcat.p12 파일을 내보냅니다.
-keytype	Apache 또는 Tomcat 기본값: Apache
-sourcefile	SSL 키가 포함된 파일(Apache) 또는 키와 인증서가 포함된 키 저장소(Tomcat)의 이름입니다.
-certfile	Apache SSL 서버 인증서가 포함된 파일의 이름입니다(Apache 에만 해당).
-sourcever	키 또는 인증서를 가져온 원본 CA SiteMinder® Federation Standalone 버전입니다(예: 12.0, 12.1). 기본값: 12.0
-sourceos	키를 가져온 원본 환경의 운영 체제입니다(예: Windows 또는 UNIX). **참고:** Linux 는 r12.1 SP3 부터 지원되었기 때문에 Linux 옵션은 없습니다. 기본값: 도구를 실행 중인 컴퓨터의 OS 입니다.
-dest	출력 파일에 대한 폴더의 경로입니다. 마이그레이션에서 이 옵션은 무시됩니다. 내보내기 기본값: 현재 폴더 **중요!** 대상 폴더를 지정하지 않으면 마이그레이션할 파일을 덮어쓰게 됩니다.

-issueralias	마이그레이션 중인 인증서에 서명한 CA 인증서의 별칭입니다. CA 인증서를 이 별칭을 사용하여 대상 CA SiteMinder® Federation Standalone 시스템으로 가져옵니다. 마이그레이션에만 사용되며 내보내기의 경우에는 무시됩니다.
-oldpwd	키가 있는 원본 시스템의 CA SiteMinder® Federation Standalone 관리 암호입니다.
-newpwd	키를 이동할 대상 시스템의 CA SiteMinder® Federation Standalone 관리 암호입니다.
-h	사용 지침을 표시합니다.
-help	사용 지침을 표시합니다.
-?	사용 지침을 표시합니다.

제 28 장: 페더레이션 작업을 모니터링하기 위한 로그

이 섹션은 다음 항목을 포함하고 있습니다.

페더레이션 로깅 개요

페더레이션 작업의 문제를 해결하기 위해 로깅을 사용하십시오. 로그는 사용자 및 CA 지원 팀을 위한 중요한 진단 정보를 제공합니다.

여러 가지 로그가 페더레이션 작업에 대한 정보를 제공합니다. 기본적으로 시스템은 다음과 같은 로그를 지원합니다.

- FWS(페더레이션 웹 서비스) 응용 프로그램 로그는 다음과 같습니다.

  affwebservices.log - 이 로그 파일에는 FWS(페더레이션 웹 서비스) 응용 프로그램에 대한 메시지가 들어 있습니다. 이 파일의 기본 경로는 *federation_install_dir*\logs\fws 입니다.

  FWSTrace.log - 이 추적 로그에는 FWS 런타임 작업에 대한 정보가 들어 있습니다.

- 페더레이션 제품에 사용되는 정책 서버에 대한 서버 로그는 다음과 같습니다.

  smtracedefault.log - 이 추적 로그는 서버 런타임 작업을 추적합니다. 이 추적 로그의 기본 위치는 *federation_install_dir*\logs\server 디렉터리입니다.

  참고: 추적을 사용하면 로그 파일의 크기가 커질 수 있습니다.

  smps.log - 이 로그 파일에는 서버에 대한 정보 및 추적 메시지가 들어 있습니다. 이 로그 파일은 *federation_install_dir*\logs\server 디렉터리에 있습니다.

- Administrative UI 작업 로그

  server.log - 이 로그 파일에는 Administrative UI 및 포함된 SPS 서버에 대한 메시지가 들어 있습니다. 이 로그 파일은 *federation_install_dir*\logs\ui 디렉터리에 있습니다.

또한 페더레이션 데이터 저장소 개체 추적 로그인 **XPSConfig_***date_time_stamp*.log 를 사용하도록 설정할 수 있습니다. 이 추적 로그는 데이터 저장소의 페더레이션 개체에 대한 추적 작업을 모니터링합니다.

검사점 로그 메시지

FWSTrace.log 및 smtracedefault.log 에는 트랜잭션 중 발생하는 사항을 나타내는 검사점 로그 메시지가 있습니다. 예:

```
[07/30/2013][11:34:44][4260][5824][1181adbb-993f775c-33ba08f3-76b5
2f3b-3d2280cd-4ae][SSO.java][processRequest][Reading SAML 2.0 SP
Configuration [CHECKPOINT = SSOSAML2_SPCONFREAD_REQ]
```

트랜잭션 중 발생하는 일부 프로세스를 추적하기 위해 이러한 검사점 메시지를 검색할 수 있습니다.

검사점 메시지 외에도 트랜잭션을 추적하는 데 사용할 수 있는 트랜잭션 ID (페이지 452)도 있습니다. 트랜잭션이 실패하면 검사점 메시지 및 트랜잭션 ID 는 특정 문제를 파악하는 데 도움을 줄 수 있습니다.

FWS(페더레이션 웹 서비스) 로깅

다음 로그를 사용하도록 설정하여 FWS 응용 프로그램 런타임 작업을 모니터링할 수 있습니다.

- 정보 로깅(affwebserv.log)
- 추적 로깅(FWSTrace.log)

롤오버 빈도 및 로그 크기 같은 로그 동작을 관리하려면 LoggerConfig.properties 파일에서 설정을 수정하십시오.

참고: LoggerConfig.properties 파일과 server.log 파일을 구성하는 logger.properties 파일을 혼동하지 마십시오. 이름은 비슷하지만 두 파일은 서로 다릅니다.

다음 단계를 수행하십시오.

1. 다음 디렉터리로 이동합니다.
   federation_install_dir\secure-proxy\Tomcat\webapps\affwebservices\WEB-INF\classes\

   참고: UNIX 운영 환경에서는 경로에 슬래시(/)를 사용하십시오.
2. 텍스트 편집기에서 LoggerConfig.properties 파일을 엽니다.
3. (선택 사항) 로그 설정을 수정합니다. LoggerConfig.properties 파일의 각 설정에 대한 설명과 옵션을 검토합니다. 설정은 다음과 같습니다.

   LoggingOn

   정보 로깅을 사용하거나 사용하지 않도록 설정합니다.

   LogFileName

   기본값: *federation_install_dir*\\logs\\fws\\affwebserv.log

   기본 파일 이름은 affwebserv.log 입니다. 이 이름은 변경할 수 있습니다.

   LogLocalTime

   LogRollover

   LogSize

   LogCount

4. (선택 사항) FWS 메시지 로깅에 대한 추적 설정을 수정합니다. LoggerConfig.properties 파일의 각 설정에 대한 설명과 옵션을 검토합니다.

   TracingOn

   FWSTrace.log 파일에 대해 FWS 추적 로깅을 사용하거나 사용하지 않도록 설정합니다.

   EnableDNSLookUp

   TraceFileName

   기본 출력 파일 이름은 FWSTrace.log 입니다. 이 이름은 변경할 수 있습니다.

   TraceConfigFile

   추적 구성 파일을 나타냅니다. 구성 파일은 시스템이 모니터링하고 메시지를 로깅하는 구성 요소와 하위 구성 요소를 결정합니다.

   TraceRollover

   TraceSize

   TraceCount

   TraceFormat

   TraceDelim

5. 파일을 저장한 후 닫습니다.
6. 운영 환경에 맞게 페더레이션 서비스를 다시 시작합니다.

   - **Windows**

     다음과 같이 중지 및 시작 바로 가기를 사용합니다. 로컬 관리자가 아닌 네트워크 사용자로 로그인한 경우에는 바로 가기를 마우스 오른쪽 단추로 클릭하고 "관리자 권한으로 실행"을 선택하십시오.

     a. "시작", "모든 프로그램", "CA", "Federation Standalone", "서비스 중지"

     b. "시작", "모든 프로그램", "CA", "Federation Standalone", "서비스 시작"

- **UNIX**

  a. 명령 창을 엽니다.

  b. 다음 스크립트를 실행합니다.

  federation_install_dir/fedmanager.sh stop

  federation_install_dir/fedmanager.sh start

  참고: 서비스를 중지했다가 루트 사용자로 시작하지 마십시오.

서버 추적 로깅

서버 추적 로그(smtracedefault.log 파일)는 페더레이션 서버 런타임 작업을 추적합니다. 이 추적 로그의 기본 위치는 *federation_install_dir*\logs\server 디렉터리입니다.

참고: 추적을 사용하면 로그 파일의 크기가 커질 수 있습니다.

서버 측 추적 로깅을 설정하려면 다음 두 가지 태스크를 완료해야 합니다.

1. 서버 추적 로그 구성 파일을 설정합니다. (페이지 434) 이 구성 파일은 모니터링되어 smtracedefault.log 파일에 기록되는 구성 요소를 정의합니다. 기본 파일인 smtracedefault.txt 를 사용하거나 제공된 다른 템플릿 중 하나를 사용할 수 있습니다.
2. 서버 추적 로그 파일 smtracedefault.log 의 동작을 구성합니다 (페이지 435). 로그 출력 파일의 위치, 로그 구성 파일의 위치, 로그 출력 파일의 형식 및 로그 롤오버 빈도를 지정합니다.

서버 추적 로그 구성 파일 설정

로그 구성 파일을 설정할 수 있습니다. 로그 구성 파일은 모니터링되어 smtracedefault.log 파일에 기록되는 구성 요소를 정의합니다. 다음 파일 중 하나를 페더레이션에 사용할 수 있습니다.

- smtracedefault.txt(기본값)
- samlidp_trace.template(어설션 당사자의 작업)
- samlsp_trace.template(신뢰 당사자의 작업)

효율성을 높이기 위해 이 템플릿 중 하나를 사용하십시오. LogTraceConfig 매개 변수에 템플릿의 이름을 입력하십시오. 이 매개 변수는 XPSConfig 명령을 사용하고 SM 옵션을 선택하여 액세스합니다.

참고: 미리 구성된 템플릿은 *federation_install_dir*\siteminder\config\profiler_templates 에 있습니다.

템플릿 대신 기본 파일을 사용하고 이 파일에 모든 페더레이션 구성 요소를 수동으로 추가할 수 있습니다.

다음 단계를 수행하십시오.

1. *federation_install_dir*\siteminder\config\smtracedefault.txt 로 이동합니다.
2. 템플릿 파일을 백업합니다.
3. 편집기에서 smtracedefault.txt 파일을 엽니다.
4. 다음 텍스트를 복사하고 파일로 붙여 넣어 파일을 편집합니다. 기존 텍스트를 덮어씁니다.

```
components: Server/Policy_Server_General, IsProtected/Resource_Protection,
Login_Logout/Authentication, Login_Logout/Policy_Evaluation,
Login_Logout/Active_Expression, Login_Logout/Session_Management,
IsAuthorized/Policy_Evaluation, JavaAPI,
Fed_Server/Assertion_Generator, Fed_Server/Auth_Scheme,
Fed_Server/Configuration
data: Date, Time, Tid, TransactionID, SrcFile, Function, Domain,
Resource, Action, User, SessionID, Data, AuthReason, Message
```

5. 파일을 저장합니다.

서버 추적 로그 파일의 동작 구성

제품에 포함된 XPSConfig 도구를 사용하여 서버 측 런타임 작업에 대해 추적을 사용하도록 설정할 수 있습니다. XPSConfig 는 제품 매개 변수를 보고 해당 설정을 편집하는 데 사용할 수 있는 대화형 명령줄 유틸리티입니다.

다음 단계를 수행하십시오.

1. 제품이 설치된 시스템에서 명령 창을 엽니다.
2. XPSConfig 를 입력합니다.

   명령은 여기 표시된 그대로 입력하십시오. 명령은 대/소문자를 구분합니다.

   "Products Menu"(제품 메뉴)가 표시됩니다.
3. SM 를 입력합니다.

   "Parameters Menu"(매개 변수 메뉴)가 표시됩니다. 수정할 수 있는 각 매개 변수에는 번호가 연결되어 있습니다.
4. 수정하려는 매개 변수와 관련된 숫자를 입력합니다.
5. c 를 입력하여 값을 변경합니다.
6. q 를 입력하여 매개 변수 목록으로 돌아갑니다.
7. 서버 추적을 위해 다음 설정 값을 지정합니다.

   LogTrace

   추적이 사용되도록 설정합니다. 기본적으로 추적은 사용되지 않도록 설정되며 두 큰따옴표 사이의 공백으로 나타냅니다(" "). 설정을 비워 두지 마십시오.

   LogTraceConfig

   이 값은 추적 구성 파일을 가리킵니다. 기본값은 없습니다. *federation_install_dir*\siteminder\config*template_name* 을 입력합니다.기본 템플릿은 smtracedefault.txt 입니다. 페더레이션을 위해 특별히 마련된 다른 추적 템플릿이 있습니다.

   예:

   federation_install_dir\siteminder\config\samlidp_trace.template

   federation_install_dir\siteminder\config\samlsp_trace.template

LogTraceConsole

메시지를 콘솔 창에 표시할지 여부를 나타냅니다. 기본적으로 로그가 콘솔에 표시되지 않습니다.

LogTraceFormat

정보가 로그에 표시되는 방식을 결정합니다. 기본값은 sm 입니다. LogTraceDelimiter 설정과 함께 사용하여 구분 기호로 사용될 문자를 지정합니다.

LogTraceMode

추적 모드를 지정합니다. 기본값은 0 입니다.

LogTraceDelimiter

로그 출력 파일에서 구분 기호로 사용될 문자를 나타냅니다. 기본값은 없습니다.

LogTraceOutput

로그 출력 파일의 위치를 지정합니다. 기본값은 *federation_install_dir*\logs\server 입니다.

8. 롤오버 설정을 수정하여 로깅 및 추적 파일을 롤오버하는 빈도를 구성하십시오. 수정하려는 매개 변수와 관련된 숫자를 입력합니다.

참고: 로그 롤오버 설정에 대한 모든 변경 내용은 smtracedefault.log 파일과 smps.log 파일에 적용됩니다.

롤오버 매개 변수는 다음과 같습니다.

LogFilesToKeep

유지할 정책 서버 오류 로그의 수를 나타냅니다. 오래된 파일이 삭제됩니다.

LogRolloverDays

롤오버 발생 주기를 일 단위로 나타냅니다. 롤오버가 발생하는 일 간격에 해당하는 숫자를 입력하십시오.

LogRolloverInterval

롤오버가 시간 단위로 발생하는지 나타냅니다. 이 값을 설정하면 LogRolloverDays 가 무시됩니다.

LogRolloverOnStart(기본적으로 설정됨)

서비스를 시작할 때 로그 파일이 롤오버되는지 나타냅니다.

LogRolloverSize

롤오버되는 로그 파일의 크기를 나타냅니다. 시스템에서 크기 제한에 도달하면 다음 롤오버 간격이 되기 전이라도 로그 파일이 롤오버됩니다.

LogRolloverTime

하루 중 롤오버를 수행할 시간을 나타냅니다. 이 설정은 LogRolloverDays 매개 변수와 함께 사용됩니다. "시간:분" 형식으로 값을 입력하십시오(24 시간 기준).

예: "22:00"

9. 매개 변수 구성을 마쳤으면 q 를 계속 입력하여 XPSConfig 를 종료합니다.

XPSConfig 에서의 변경 내용은 XPSConfig 도구를 종료하기 전까지는 인식되지 않습니다. 표시된 대로 일부 변경 내용은 시스템 서비스를 다시 시작해야 적용됩니다.

server.log 파일 설정

server.log 파일은 제품의 Administrative UI 작업을 검사하는 데 유용합니다. 이 로그 파일에는 포함된 SPS 서버에 대한 메시지가 들어 있습니다. 이 로그 파일은 *federation_install_home*/logs/ui 디렉터리에 있습니다.

logger.properties 파일 및 **log4j.properties** 파일은 server.log 파일에 기록되는 내용을 결정하는 로그 설정을 포함하고 있습니다. 이러한 파일의 설정은 시스템이 런타임에 읽는 이름/값 쌍이나 지시문의 그룹입니다.

logger.properties 파일

logger.properties 파일은 *federation_install_dir*/secure-proxy/Tomcat/properties 디렉터리에 있습니다. 이 파일의 내용은 다음과 같은 섹션으로 나뉘어 있습니다.

- SvrConsoleAppender 설정
- SvrFileAppender 설정
- Server.conf 설정
- 로그 롤오버 설정

이 파일에 포함된 지시문은 name=value 형식을 따릅니다. # 기호로 시작하는 행은 주석이므로 시스템이 구성 설정을 로드할 때 이 행은 읽지 않습니다.

참고: Windows 시스템의 경로 이름에는 이중 백슬래시(\\)가 사용됩니다.

Log4j.properties

lo4j.properties 파일은 *federation_install_dir*/secure-proxy/Tomcat/webapps/fedui/WEB-INF/classes 디렉터리에 있습니다. 이 파일은 Administrative UI 작업에 대해 기록되는 로그 수준을 결정합니다.

로그 파일을 수정하는 절차는 동일합니다. 이 파일은 시스템을 다시 시작하지 않고 수정할 수 있습니다.

다음 단계를 수행하십시오.

1. 텍스트 편집기에서 파일을 엽니다.
2. 지시문을 필요한 대로 편집합니다.
3. 파일을 저장합니다.

로그 설정이 변경되었습니다.

로그 설정

Server.conf 설정

logger.properties 파일의 Server.conf 설정은 로깅을 사용하거나 사용하지 않도록 설정하고 로깅 수준을 지정하며 로그 메시지의 출력 형식을 설정합니다. 이 섹션에서는 다음과 같은 형식으로 항목을 수정해야 합니다.

`log4j.rootCategory=<log_level>,<output_format>`

log_level

메시지의 로그 수준을 지정합니다. 다음 값은 우선 순위가 낮은 것부터 높은 순서로 나열됩니다.

OFF, FATAL, ERROR, WARN, INFO, DEBUG, ALL

로깅을 사용하지 않도록 설정하려면 로그 수준을 OFF 로 설정하십시오. 이 값이 기타 다른 값으로 설정되면 로깅이 활성화됩니다.

기본값: INFO

output_format

로그 메시지가 콘솔이나 파일 또는 둘 다에 표시되는지 여부를 지정합니다.

기본값: SvrFileAppender

예: 로그 수준을 INFO 로 설정하고 로그 메시지를 콘솔과 파일에 표시하려면 다음 항목을 사용하십시오.

`log4j.rootCategory=INFO,SvrConsoleAppender,SvrFileAppender`

SvrConsoleAppender 설정

SvrConsoleAppender 설정 섹션은 콘솔에 대한 로깅 이벤트를 제어합니다. 이 섹션에서는 다음과 같은 형식으로 항목을 수정해야 합니다.

`log4j.appender.SvrConsoleAppender.layout.ConversionPattern=<log_message_format>`

log_message_format

콘솔에 대한 로그 메시지 출력의 형식을 지정합니다. 이 제품은 모든 log4j 날짜 패턴 문자열을 지원합니다.

기본값: [%d{dd/MMM/yyyy:HH:mm:ss-SSS}] [%p] - %m%n

SvrFileAppender 설정

SvrFileAppender 설정 섹션은 파일에 대한 로깅 이벤트를 제어합니다. 이 섹션은 파일에 기록되는 로그 메시지의 로그 롤오버 빈도 및 형식을 정의합니다. 이 섹션에서는 다음과 같은 형식으로 항목을 수정해야 합니다.

```
log4j.appender.SvrFileAppender.File=<log_file_path>
log4j.appender.SvrFileAppender.Append=true
log4j.appender.SvrFileAppender.layout.ConversionPattern=<log_messa
ge_format>
```

log_file_path

로그 파일의 이름과 경로를 지정합니다.

기본 이름: server.log

기본 경로: *install_dir_home*/secure-proxy/proxy-engine/logs/ui/server.log

true|false

로그 메시지를 기존 파일에 추가하도록 할지 여부를 시스템에 지시합니다. 이 값을 true 로 설정하면 시스템이 새 로그 메시지를 기존 로그 파일에 추가합니다. 이 값을 false 로 설정하면 시스템이 기존 로그 파일을 롤오버하고 새 로그 파일을 생성합니다.

기본값: true

log_message_format

시스템이 server.log 파일에 기록하는 로그 메시지의 형식을 지정합니다. 이 제품은 모든 log4j 날짜 패턴 문자열을 지원합니다.

기본값: [%d{dd/MMM/yyyy:HH:mm:ss-SSS}] [%p] - %m%n

사용되는 로그 롤링의 유형

로그 롤링 섹션은 기존 로그 파일이 롤오버되고 새 로그가 생성되는 경우를 결정합니다. 파일 크기 또는 파일 날짜를 기준으로 로그 롤오버를 사용하도록 설정합니다.

이 섹션에서는 다음과 같은 형식으로 항목을 수정해야 합니다.

```
log4j.appender.SvrFileAppender.MaxFileSize=1MB
log4j.appender.SvrFileAppender.MaxBackupIndex=10
#log4j.appender.SvrFileAppender.DatePattern='.'yyyy-MM-dd
```

MaxFileSize

로그 파일의 최대 크기를 지정합니다. 이 크기를 초과하면 시스템이 로그 파일을 생성해야 합니다.

기본값: 1 MB

MaxBackupIndex

시스템이 생성하는 최대 로그 파일 수를 지정합니다. 로그 파일의 수가 MaxBackupIndex 값을 초과하면 시스템이 가장 오래된 로그 파일을 삭제하고 새 파일을 생성합니다.

기본값: 10

DatePattern

시스템이 로그 파일을 생성해야 하는 날짜를 지정합니다.

기본값: yyyy-MM-dd

새 로그 파일은 <*log_file_name*>.<*date_format*> 이름으로 생성됩니다.

log_file_name

로그 파일의 이름을 지정합니다.

기본값: server.log

date_format

로그 파일이 생성된 날짜를 지정합니다. 이 파일은 모든 log4j 날짜 패턴 문자열을 지원합니다.

기본값: yyyy-MM-dd

server.log 의 log4j.properties 파일

log4j.properties 파일은 시스템이 server.log 파일에 기록하는 추가적인 Administrative UI 로깅을 제어합니다. 이 파일은 *federation_install_dir*\secure-proxy\Tomcat\webapps\fedui\WEB-INF\classes 디렉터리에 있습니다.

다음 항목을 수정할 수 있습니다.
`log4j.appender.UIConsoleAppender.layout.ConversionPattern=<log_message_format>`

log_message_format

콘솔에 대한 로그 메시지의 출력 형식을 지정합니다. 이 제품은 모든 log4j 날짜 패턴 문자열을 지원합니다.

기본값: [%p] %c - %m%n

`log4j.rootCategory=<log_level>,<output_format>`

log_level

메시지의 로그 수준을 지정합니다. 다음 값은 우선 순위가 낮은 것부터 높은 순서로 나열됩니다.

OFF, FATAL, ERROR, WARN, INFO, DEBUG, ALL

로깅을 사용하지 않도록 설정하려면 로그 수준을 OFF 로 설정하십시오. 이 값이 기타 다른 값으로 설정되면 로깅이 활성화됩니다.

기본값: INFO

output_format

로그 메시지가 콘솔이나 파일 또는 둘 다에 출력되는지 여부를 지정합니다.

기본값: UIConsoleAppender

예: 로그 수준을 INFO 로 설정하고 로그 메시지를 콘솔과 파일에 표시하려면 다음 항목을 사용하십시오.
`log4j.rootCategory=INFO,UIConsoleAppender,UIFileAppender`

또한 두 가지 DEBUG 항목의 주석 처리를 제거할 수 있습니다.

페더레이션 데이터 개체 추적 로깅

페더레이션 데이터 저장소 개체를 모니터링하려면 XPS 추적을 사용하도록 설정하십시오. 이러한 모니터링 작업은 smtracedefault.log 에 기록됩니다. smtracedefault.log 는 *federation_install_dir*\logs\server 디렉터리에 있습니다.

다음 단계를 수행하십시오.

1. 명령 창을 엽니다.
2. XPSConfig 를 입력합니다.

   명령은 여기 표시된 그대로 입력하십시오. 명령은 대/소문자를 구분합니다.

   "Products Menu"(제품 메뉴)가 표시됩니다.
3. xTrace 옵션으로 X 를 입력합니다.

   "Tracer Menu"(추적 프로그램 메뉴)가 표시됩니다.
4. **fed** 옵션과 연결된 번호를 입력합니다. fed 와 관련된 모든 옵션이 선택되고 "x"로 표시됩니다.
5. U 를 입력하여 선택 사항을 저장합니다. 그러면 추적 프로그램 메뉴가 업데이트됩니다.
6. XPSConfig 가 종료될 때까지 q 를 입력합니다.
7. 변경 내용이 적용되도록 페더레이션 서비스를 다시 시작합니다.
8. 운영 환경에 맞게 페더레이션 서비스를 다시 시작합니다.
   - **Windows**

     다음과 같이 중지 및 시작 바로 가기를 사용합니다. 로컬 관리자가 아닌 네트워크 사용자로 로그인한 경우에는 바로 가기를 마우스 오른쪽 단추로 클릭하고 "관리자 권한으로 실행"을 선택하십시오.

     a. "시작", "모든 프로그램", "CA", "Federation Standalone", "서비스 중지"

     b. "시작", "모든 프로그램", "CA", "Federation Standalone", "서비스 시작"

- **UNIX**

  a. 명령 창을 엽니다.

  b. 다음 스크립트를 실행합니다.

  federation_install_dir/fedmanager.sh stop

  federation_install_dir/fedmanager.sh start

참고: 서비스를 중지했다가 루트 사용자로 시작하지 마십시오.

감사 로깅

CA SiteMinder?Federation Standalone 에서는 smaccess.log 라는 감사 로그가 *federation_install_dir*/logs/server 디렉터리에 자동으로 생성됩니다. 이 로그는 XPSConfig 명령을 사용하여 인증 이벤트나 권한 부여 이벤트 또는 둘 모두에 대한 로깅을 사용하도록 설정하기 전까지 비어 있습니다.

참고: UNIX 플랫폼에서는 XPSConfig 의 대/소문자를 구분합니다.

감사 로깅을 사용하도록 설정하려면

1. 명령 창을 엽니다.
2. 명령 프롬프트에 XPSConfig 를 입력합니다.

   "Product Menu"(제품 메뉴)가 표시됩니다.
3. SM 를 입력합니다.

   매개 변수 및 해당 매개 변수의 현재 값이 목록으로 표시됩니다.
4. (선택 사항) **f** 를 입력하여 설정 목록을 필터링합니다.

   "필터 입력" 프롬프트에서 **report** 를 입력하여 감사 로그와 관련된 모든 설정을 찾습니다.
5. 사용하도록 설정할 감사 로깅 유형과 관련된 번호를 입력합니다.

   ReportAuth

   인증 이벤트에 대한 로그 설정을 지정합니다.

   ReportAz

   권한 부여 이벤트에 대한 로그 설정을 지정합니다.
6. c 를 입력하여 값을 변경합니다. 기본값은 0 이며 이벤트가 로깅되지 않음을 나타냅니다.

7. 프롬프트에서 다음 값 중 하나를 입력합니다.

   1 = 모든 이벤트 로깅

   2 = 거부 이벤트만 로깅

8. "Product Menu"(제품 메뉴)로 돌아갈 때까지 q 를 입력합니다.

   감사 로깅을 사용할 수 있습니다.

참고: 언제든지 이 절차를 반복하여 감사 로그 설정을 업데이트할 수 있습니다.

감사 로그 이름 및 위치 설정(선택 사항)

감사 로그의 기본 이름은 smaccess.log 이고 기본 위치는 *federation_install_dir*/logs/server 입니다. 이러한 값은 변경할 수 있습니다.

다음 단계를 수행하십시오.

1. 명령 창을 엽니다.
2. 명령줄 프롬프트에 XPSConfig 를 입력합니다.

   참고: UNIX 플랫폼에서는 XPSConfig 의 대/소문자를 구분합니다.

   "Product Menu"(제품 메뉴)가 표시됩니다.

3. SM 을 입력합니다.

   매개 변수 및 해당 값이 목록으로 표시됩니다.

4. (선택 사항) **f** 를 입력하여 설정 목록을 필터링합니다.

   "필터 입력" 프롬프트에서 **text** 를 입력하여 감사 로그 텍스트 파일 이름과 관련된 설정을 찾습니다.

5. ReportTextFile 설정과 관련된 번호를 입력합니다.

   현재 값이 표시됩니다.

6. c 를 입력하여 파일 이름을 변경합니다.
7. 올바른 경로와 새 파일 이름을 입력합니다.
8. 시스템 명령 프롬프트로 돌아갈 때까지 q 를 입력합니다.

새 파일 이름과 위치가 저장되었습니다.

감사 로깅에 ODBC 데이터베이스 사용(선택 사항)

기본 텍스트 파일 대신 ODBC 데이터베이스를 사용하여 감사 데이터를 기록할 수 있습니다.

다음 단계를 수행하십시오.

1. 감사 로그 저장소 유형을 ODBC 로 변경합니다.
2. ODBC 데이터 원본을 구성합니다. 다음 지침 중 하나를 참조하십시오.
   - Windows 에서 SQL Server 데이터 원본 만들기 (페이지 448)
   - UNIX 시스템에서 SQL Server 데이터 원본 만들기 (페이지 449)
   - Windows 에서 Oracle 데이터 원본 만들기 (페이지 450)
   - UNIX 시스템에서 Oracle 데이터 원본 만들기 (페이지 451)

감사 로그 저장소 유형 변경

감사 로그는 기본적으로 텍스트 형식입니다. 감사 데이터를 ODBC 데이터베이스에 저장하려면 로그의 저장소 유형을 변경하십시오.

중요! 감사 로그 저장소 유형을 TEXT 에서 ODBC 로 변경하면 다시 되돌릴 수 없습니다.

다음 단계를 수행하십시오.

1. 명령 창을 엽니다.
2. 명령줄 프롬프트에 XPSConfig 를 입력합니다.

   참고: UNIX 플랫폼에서는 XPSConfig 의 대/소문자를 구분합니다.

   "Product Menu"(제품 메뉴)가 표시됩니다.
3. SM 을 입력합니다.

   매개 변수 및 해당 매개 변수의 현재 값이 목록으로 표시됩니다.
4. (선택 사항) **f** 를 입력하여 설정 목록을 필터링합니다.

   "필터 입력" 프롬프트에서 **store** 를 입력하여 감사 로그 저장소 유형과 관련된 모든 설정을 찾습니다.
5. LogStoreNamespace 설정에 해당하는 숫자를 입력합니다.

   현재 값이 표시됩니다.
6. 저장소 유형을 변경하려면 c 를 입력합니다.

7. 프롬프트에 **ODBC:**를 입력합니다.

   참고: 입력할 때 콜론을 포함하십시오.

8. q 를 두 번 입력하여 매개 변수 목록으로 돌아갑니다.

9. 이러한 추가 설정을 구성합니다. 수정하려는 각 설정의 번호를 입력합니다.

   참고: 설정 목록을 필터링하려면 **f** 를 입력하십시오. "필터 입력" 프롬프트에서 **Db** 를 입력하면 감사 로그 데이터베이스와 관련된 모든 설정을 찾을 수 있습니다.

   DbLogAdminName

   감사 로그의 데이터 원본 사용자 이름을 지정합니다.

   제한: 문자열입니다. 이 값은 LogStoreNamespace 를 ODBC 로 설정한 경우에만 적용됩니다.

   DbLogAdminPassword

   감사 로그의 데이터 원본 사용자 암호를 지정합니다.

   제한: 문자열입니다. 이 값은 LogStoreNamespace 를 ODBC 로 설정한 경우에만 적용됩니다.

   DbLogDataSource

   감사 로그의 데이터 원본 이름을 지정합니다.

   제한: 문자열입니다. 이 값은 LogStoreNamespace 를 ODBC 로 설정한 경우에만 적용됩니다.

   DbLogMaxConnections

   감사 로그를 위해 지원되는 데이터 원본과의 최대 연결 수를 지정합니다.

   기본값: 15

   제한: 정수여야 합니다. 이 값은 LogStoreNamespace 를 ODBC 로 설정한 경우에만 적용됩니다.

DbLogUseDefault

감사 로그에 정책 저장소와 같은 ODBC 데이터 원본을 사용할지 여부를 지정합니다.

기본값: FALSE

제한: TRUE 또는 FALSE 입니다. 이 값은 LogStoreNamespace 를 ODBC 로 설정한 경우에만 적용됩니다.

10. 시스템 명령 프롬프트로 돌아갈 때까지 q 를 입력합니다.
11. ODBC 데이터베이스를 사용하여 감사 데이터를 기록하려면 데이터 원본을 설정합니다 (페이지 446).

Windows 에서 SQL Server 데이터 원본 만들기

ODBC 를 사용하려면 MS SQL Server 유선 프로토콜에 대해 데이터 원본을 구성해야 합니다.

Windows 에서 데이터 원본을 생성하려면

1. 다음 작업 중 하나를 수행하십시오.
   - 지원되는 32 비트 Windows 운영 체제를 사용 중인 경우 "시작"을 클릭하고 "프로그램", "관리 도구", "ODBC 데이터 원본"을 차례로 선택합니다.
   - 지원되는 64 비트 Windows 운영 체제를 사용 중인 경우 다음을 수행합니다.
     a. *install_home*\Windows\SysWOW64 로 이동합니다.
     b. odbcad32.exe 를 두 번 클릭합니다.

   "ODBC 데이터 원본 관리자"가 표시됩니다.
2. "시스템 DSN" 탭을 클릭합니다.

   시스템 데이터 원본 설정이 표시됩니다.
3. "추가"를 클릭합니다.

   "새 데이터 원본 만들기" 대화 상자가 표시됩니다.
4. "SiteMinder SQL Server Wire Protocol"을 선택하고 "마침"을 클릭합니다.

   "ODBC SQL Server Wire Protocol Driver 설정" 대화 상자가 표시됩니다.

5. "데이터 원본 이름" 필드에 데이터 원본 이름을 입력합니다.

   예: CA SiteMinder® Federation Standalone 데이터 원본

   참고: 데이터 원본 이름을 적어 두십시오. 이 정보는 데이터베이스를 정책 저장소로 구성할 때 필요합니다.

6. "서버" 필드에 MS SQL Server 호스트 시스템의 이름을 입력합니다.
7. "데이터베이스 이름" 필드에 데이터베이스 이름을 입력합니다.
8. "테스트"를 클릭합니다.

   연결 설정이 테스트되고 연결에 성공했음을 나타내는 메시지가 표시됩니다.

9. "확인"을 클릭합니다.

   SQL Server 데이터 원본이 구성되어 "시스템 데이터 원본" 목록에 표시됩니다.

UNIX 시스템에서 SQL Server 데이터 원본 만들기

CA SiteMinder?Federation Standalone ODBC 데이터 원본은 system_odbc.ini 파일을 사용하여 구성합니다. 이 파일은 *federation_install_dir*/siteminder/db 에 있는 sqlserverwire.ini 를 system_odbc.ini 로 이름을 바꿔 생성할 수 있습니다. 이 system_odbc.ini 파일에는 사용할 수 있는 ODBC 데이터 원본의 이름뿐 아니라 이러한 데이터 원본과 연관된 특성도 모두 들어 있습니다. 각 사이트에 맞게 이 파일을 사용자 지정해야 합니다. SiteMinder 에 대한 추가 ODBC 사용자 디렉터리를 정의하는 등 이 파일에 데이터 원본을 더 추가할 수도 있습니다.

system_odbc.ini 파일의 첫 번째 섹션인 [ODBC Data Sources]에는 현재 사용할 수 있는 모든 데이터 원본의 목록이 들어 있습니다. "=" 앞의 이름은 해당 파일에서 각 개별 데이터 원본을 설명하는 이후 섹션을 나타냅니다. "=" 뒷부분은 주석 필드입니다.

참고: 데이터 원본 항목의 첫 행인 [SiteMinder Data Source]를 수정할 경우 변경한 값을 적어 두십시오. 이 값은 ODBC 데이터베이스를 정책 저장소로 구성할 때 필요합니다.

system_odbc.ini 파일에는 각 데이터 원본의 특성을 설명하는 섹션이 있습니다. 첫 번째 특성은 해당 데이터 원본이 SiteMinder 에서 사용될 때 로드되는 ODBC 드라이버입니다. 나머지 특성은 드라이버마다 다릅니다.

MS SQL Server 데이터 원본을 추가하려면 파일의 [ODBC Data Sources] 섹션에 새 데이터 원본 이름을 추가한 후 데이터 원본과 동일한 이름을 사용하여 데이터 원본을 설명하는 섹션을 추가해야 합니다. 새 서비스 이름을 생성하거나 다른 드라이버를 사용하려는 경우에는 system_odbc.ini 파일을 변경해야 합니다. Oracle 또는 SQL 드라이버에 대한 항목은 [SiteMinder Data Source] 아래에 추가해야 합니다.

또한 MS SQL Server 데이터 원본을 구성하려면 먼저 *federation_install_dir*/siteminder/db 디렉터리에 system_odbc.ini 파일을 생성해야 합니다. 이를 위해서는 *federation_install_dir*/siteminder/db 에 있는 sqlserverwire.ini 를 system_odbc.ini 로 이름을 바꿔야 합니다.

Windows 에서 Oracle 데이터 원본 만들기

Oracle 데이터베이스용 ODBC 데이터 원본을 만듭니다.

다음 단계를 수행하십시오.

1. 다음 작업 중 하나를 수행하십시오.
   - 지원되는 32 비트 Windows 운영 체제를 사용 중인 경우 "시작"을 클릭하고 "프로그램", "관리 도구", "ODBC 데이터 원본"을 차례로 선택합니다.
   - 지원되는 64 비트 Windows 운영 체제를 사용 중인 경우 다음을 수행합니다.
     a. *install_home*\Windows\SysWOW64 로 이동합니다.
     b. odbcad32.exe 를 두 번 클릭합니다.

   "ODBC 데이터 원본 관리자"가 표시됩니다.
2. "시스템 DSN" 탭을 클릭하고 "추가"를 클릭합니다.

   "새 데이터 원본 만들기" 대화 상자가 표시됩니다.
3. "SiteMinder Oracle Wire Protocol"을 선택하고 "마침"을 클릭합니다.

   "ODBC Oracle Wire Protocol Driver 설정" 대화 상자가 표시됩니다. "일반" 탭이 선택되어 있습니다.
4. "데이터 원본 이름" 필드에 데이터 원본을 식별할 수 있는 이름을 입력합니다.

   참고: 이 이름을 기록해 두십시오. 정책 서버를 데이터베이스에 연결할 때 이 데이터 원본 이름이 필요합니다.

5. "호스트 이름" 필드에 Oracle 데이터베이스가 설치된 컴퓨터의 컴퓨터 이름을 입력합니다.
6. "포트 번호" 필드에 Oracle 데이터베이스가 컴퓨터에서 수신 대기하는 포트 번호를 입력합니다.
7. "SID" 필드에 연결할 Oracle 인스턴스의 이름을 입력합니다.

   참고: 서비스 이름은 tnsnames.ora 파일에 지정되어 있습니다. SID 는 데이터베이스 인스턴스의 시스템 식별자입니다. tnsnames.ora 파일에는 서비스 이름과 Oracle 이 Oracle 인스턴스를 식별하고 해당 인스턴스에 연결하는 데 사용하는 상세 정보가 들어 있습니다.

   예: tnsnames.ora 파일에 Oracle 인스턴스에 대한 다음 항목이 들어 있으면 SID 필드에 instance1 을 입력합니다.

   ```
   instance1 =
       (Description=
       (Address = (PROTOCOL = TCP)(Host = myhost)(Port=1521))
       (Connect_DATA_ = (SID = SIDofinstance1))
       )
   ```

8. "연결 테스트"를 클릭합니다.

   연결 설정이 테스트되고 연결에 성공했음을 나타내는 메시지가 표시됩니다.

9. "확인"을 클릭합니다.

   유선 프로토콜 드라이버에 대해 Oracle 데이터 원본이 구성됩니다.

UNIX 시스템에서 Oracle 데이터 원본 만들기

SiteMinder ODBC 데이터 원본은 system_odbc.ini 파일을 사용하여 구성합니다. 이 파일은 *federation_install_dir*/siteminder/db 에 있는 oraclewire.ini 를 system_odbc.ini 로 이름을 바꿔 생성할 수 있습니다. 이 system_odbc.ini 파일에는 사용할 수 있는 ODBC 데이터 원본의 이름뿐 아니라 이러한 데이터 원본과 연관된 특성도 모두 들어 있습니다. 각 사이트에 맞게 이 파일을 사용자 지정해야 합니다. SiteMinder 에 대한 추가 ODBC 사용자 디렉터리를 정의하는 등 이 파일에 데이터 원본을 더 추가할 수도 있습니다.

system_odbc.ini 파일의 첫 번째 섹션인 [ODBC Data Sources]에는 현재 사용할 수 있는 모든 데이터 원본의 목록이 들어 있습니다. "=" 앞의 이름은 해당 파일에서 각 개별 데이터 원본을 설명하는 이후 섹션을 나타냅니다. "=" 뒷부분은 주석 필드입니다.

참고: 데이터 원본 항목의 첫 행인 [SiteMinder Data Source]를 수정할 경우 변경한 값을 적어 두십시오. 이 값은 ODBC 데이터베이스를 정책 저장소로 구성할 때 필요합니다.

system_odbc.ini 파일에는 각 데이터 원본의 특성을 설명하는 섹션이 있습니다. 첫 번째 특성은 해당 데이터 원본이 SiteMinder 에서 사용될 때 로드되는 ODBC 드라이버입니다. 나머지 특성은 드라이버마다 다릅니다.

Oracle 데이터 원본을 추가하려면 이 파일의 [ODBC Data Sources] 섹션에 새 데이터 원본 이름을 추가한 후 데이터 원본과 동일한 이름을 사용하여 데이터 원본을 설명하는 섹션을 추가해야 합니다. 새 서비스 이름을 생성하거나 다른 드라이버를 사용하려는 경우에는 system_odbc.ini 파일을 변경해야 합니다. [SiteMinder Data Source] 아래에 SQL Server 또는 Oracle 드라이버에 대한 항목을 추가해야 합니다.

또한 Oracle 데이터 원본을 구성하려면 먼저 *federation_install_dir*/siteminder/db 디렉터리에 system_odbc.ini 파일을 생성해야 합니다. 이를 위해서는 *federation_install_dir*/siteminder/db 에 있는 oraclewire.ini 를 system_odbc.ini 로 이름을 바꿔야 합니다.

페더레이션 문제 해결에 도움이 되는 트랜잭션 ID

한 파일 내에 수많은 트랜잭션이 기록되어 있는 경우 페더레이션 트랜잭션의 문제를 해결하는 것이 쉽지 않습니다. 트랜잭션 로그에서 단일 트랜잭션을 추적하려면 SAML 트랜잭션 ID 를 사용하십시오. 페더레이션 호출이 발생하면 FWS 응용 프로그램이 먼저 SAML 트랜잭션 ID 를 생성합니다. SAML 트랜잭션 ID 는 한 번만 생성됩니다. 이 고유 SAML 트랜잭션 ID 는 여러 트랜잭션 ID 로 매핑될 수 있습니다.

예를 들어 SAML 2.0 POST 트랜잭션용 fwstrace.log 에서 다음 메시지가 나타날 수 있습니다. 두 트랜잭션 ID 의 매핑을 보여 주는 굵게 표시된 행을 참고하십시오.

```
[08/01/2013][17:33:54][2292][1884][1c2d7650-b006e46a-ed071f41-bbbe
de33-fe78e2dd-38d][SSO.java][processAuthentication][SAMLTransactio
nID 2aaf90ec-fdef4897-0ef49d91-63d4031d-f508a3e9-12 maps to
TransactionID: 1c2d7650-b006e46a-ed071f41-bbbede33-fe78e2dd-38d.]
```

CA SiteMinder?Federation Standalone 시스템은 어설션 당사자로 기능하는 경우에만 새 SAMLTransactionID 를 생성합니다. 이러한 특정 활동은 다음의 경우 발생합니다.

- 페더레이션 웹 서비스가 세션을 구성하기 위해 브라우저를 인증 URL 로 리디렉션하는 경우
- 다음 HTTP-아티팩트 싱글 사인온 트랜잭션의 경우:
  - 어설션 당사자가 신뢰 당사자에 아티팩트를 반환하는 경우
  - 어설션 당사자가 아티팩트를 확인하는 경우
- 사용자가 아이덴티티 검색 프로필 URL 로 리디렉션되는 경우
- 어설션 당사자에서 싱글 로그아웃 중

신뢰 당사자에는 요청 ID 가 있으며, 이 ID 는 로그 파일을 통해 쉽게 추적될 수 있습니다. 요청 ID 는 신뢰 당사자에서 SAMLTransactionID 를 생성하기 위해 CA SiteMinder?Federation Standalone 시스템에서 필요하지 않습니다.

각 고유 SAML 트랜잭션 ID 에는 여러 트랜잭션 ID 가 있을 수 있습니다. 새 HTTP 트랜잭션이 발생하면 새 트랜잭션 ID 가 생성됩니다. 이 트랜잭션 ID 는 단일 SAML 트랜잭션 ID 로 매핑됩니다. 예를 들어, 추적 로그에서 다음과 같은 항목을 볼 수 있습니다.

```
SamlTransactionID ["xyz"] maps to TransationID["123"]
["123"] HTTP operation
["123"] HTTP operation
```

A new transaction ID "456" is generated:

```
SamlTransactionID["xyz"] Maps to Transactionid["456"]
["456"] <some operation>
["456"] <some operation>
```

트랜잭션 ID 는 fwstrace.log 및 smtracedefault.log 에 배치됩니다. 즉, 단일 트랜잭션에 대해 동일한 트랜잭션 ID 집합이 이들 로그에 각각 기록됩니다. 이들 로그에서 ID 를 추적하면 트랜잭션을 추적할 수 있습니다. 오류가 발생한 경우 ID 를 사용하여 오류가 발생한 해당 트랜잭션에서 어느 이벤트가 실패했는지 확인할 수 있습니다.

로그에서 단일 트랜잭션을 추적하는 방법

트랜잭션을 모니터링하려면 FWSTrace.log 또는 smtracedefault.log 에서 트랜잭션 ID 의 두 가지 유형을 추적할 수 있습니다. 오류가 발생하는 경우 ID 를 확인하면 오류 지점을 파악하는 데 도움이 될 수 있습니다.

로그에서 트랜잭션을 추적하려면 다음 방법 중 하나 이상을 사용하십시오.

- 텍스트 편집기에서 추적 파일을 열고 문자열 **SAMLTransactionID**(공백 없음) 또는 특정 SAMLTransactionID 를 검색하십시오. 로그에 있는 이 항목의 모음을 통해 전체 엔드-투-엔드 트랜잭션을 볼 수 있습니다. 트랜잭션이 진행된 정도를 파악할 수 있습니다.
- 로그 파일에서 트랜잭션 ID 를 추적하십시오. 트랜잭션 ID 는 HTTP 트랜잭션을 나타냅니다. 하나의 SAML 트랜잭션 ID 에 여러 트랜잭션 ID 가 연결될 수 있습니다. 실패한 트랜잭션은 브라우저에 트랜잭션 ID 를 표시합니다. FWSTrace.log 및 smtracedefault 로그에서 검사점 오류 메시지를 검색하려면 표시된 트랜잭션 ID 를 사용하십시오.
- 파일을 검색하는 도구를 사용하여 로그 파일을 구문 분석하십시오. UNIX 및 Windows 플랫폼에서는 grep 명령과 같은 도구를 사용할 수 있습니다. grep 명령은 원시 데이터 스트림을 한 줄씩 표시하므로 크기가 큰 텍스트 파일을 텍스트 편집기에 로드할 필요가 없습니다.

  예:

```
[usr@rhel632 etc]# more fwstrace.log| grep checkpoint
[CHECKPOINT = SSOSAML2_SPCONFFROMPS_REQ]]
[CHECKPOINT = SSOSAML2_SPCONFREAD_REQ]]
[CHECKPOINT = SSOSAML2_SPCONFFROMCACHE_REQ]]
[CHECKPOINT = SSOSAML2_SESSIONCOOKIEVALIDATE_REQ]]
```

제 29 장: 페더레이션 시스템 구성 복원

이 섹션은 다음 항목을 포함하고 있습니다.

시스템을 이전 구성으로 복원하는 방법

이전에 백업한 구성으로 되돌리는 방법으로 시스템 구성을 복원할 수 있습니다. 현재 구성에서 문제가 발생하는 경우 이전 구성으로 되돌리는 것이 유용할 수 있습니다.

이전 구성으로 되돌리는 프로세스는 다음과 같습니다.

1. 복원하려는 시스템의 기존 CA SiteMinder® Federation Standalone 구성을 백업합니다.
2. 백업 구성을 생성한 시점의 설정과 동일한 설정을 사용하여 이 시스템에서 구성 마법사를 실행합니다.

   구성 마법사를 실행할 때 설정은 동일하게 유지해야 합니다.

   다음 설정은 원래 구성과 일치해야 합니다.

   - **배포 설정**

     새 시스템에 대해 동일한 배포 모드(프록시 또는 독립 실행형)를 선택하십시오.

   - **포트 번호**

     백업한 시스템에서 사용하는 것과 동일한 포트를 지정하십시오.

- **가상 호스트 이름**

  시스템을 처음 구성할 때 가상 호스트를 사용한 경우 동일한 가상 호스트 이름을 사용하십시오. 또한 호스트 파일에 시스템에 대한 적절한 항목을 추가하십시오.

- **SiteMinder 커넥터**

  시스템에서 SiteMinder 커넥터를 사용한 경우 SiteMinder 커넥터를 다시 선택하십시오.

3. 백업한 구성을 시스템에 가져옵니다.

다음 섹션에서는 이 프로세스에 대해 자세히 설명합니다.

기존 구성 백업

구성을 백업하면 페더레이션 시스템을 복구하거나 마이그레이션하는 데 유용합니다.

참고: 이 절차는 r12.52 SP1 버전 이상에 적용됩니다.

구성을 백업하려면 구성 데이터를 내보내십시오. 제품에 포함되어 있는 XPSExport 도구를 사용하면 구성 데이터를 XML 파일로 내보낼 수 있습니다.

중요! 내보내기를 진행하는 중에는 페더레이션 트랜잭션이 제대로 처리되지 않습니다.

다음 단계를 수행하십시오.

1. 명령 창을 엽니다.
2. 다음 명령을 입력하여 구성을 내보냅니다.

   XPSExport *export_file_name* -xe -xp -passphrase *passphrase*

   export_file_name

   내보내기를 통해 생성되는 출력 파일의 이름을 지정합니다. XPSExport 에서 생성하는 출력은 XML 형식이기 때문에 파일 이름도 **.xml** 확장명으로 끝나야 합니다.

   passphrase

   중요한 데이터를 암호 해독하는 데 필요한 암호를 지정합니다. 암호는 8 자 이상이고 숫자, 대문자, 소문자를 각각 하나 이상 포함해야 합니다. 암호에 공백이 포함되어 있는 경우 암호를 따옴표로 묶어야 합니다.

   참고: 암호를 직접 입력하지 않으려면 명령에서 제외하십시오. 그러면 XPSExport 도구가 화면에 표시되지 않게 암호와 암호 확인을 묻는 메시지를 표시합니다.

내보내기를 수행하면 암호화된 구성 데이터가 포함된 XML 파일이 생성됩니다. 구성을 복원하려면 이 파일을 사용하십시오.

백업된 구성으로 되돌리기

기존 CA SiteMinder?Federation Standalone 구성에 문제가 있는 경우 동일 시스템의 백업된 이전 구성으로 되돌릴 수 있습니다.

구성을 복원하려는 경우 제품에 포함되어 있는 XPSImport 도구를 사용하여 XML 파일을 가져오십시오.

중요! 명시되어 있는 대로 가져오기 단계를 수행해야 합니다. 절차가 완료될 때까지 Administrative UI 의 "인증서 및 키" 탭에 액세스하지 마십시오.

다음 단계를 수행하십시오.

1. 페더레이션 데이터를 사용할 새 데이터베이스 인스턴스를 설정합니다.

   중요! 이 단계에서 기존 데이터베이스를 사용하지 마십시오. 기존 데이터베이스를 사용하면 가져오기가 실패합니다.

2. "구성 마법사"를 실행하고 해당 메시지가 표시되면 새 데이터베이스 인스턴스를 지정합니다.

   원래 구성에 사용했던 것과 동일한 설정을 새로 구성하는 데이터베이스에도 사용합니다. 다음과 같은 설정이 포함됩니다.

   - 배포 모드
   - 포트 번호
   - 가상 호스트 이름
   - SiteMinder 커넥터

3. 플랫폼에 따라 페더레이션 서비스를 중지합니다.

   Windows

   중지 바로 가기를 사용합니다. 로컬 관리자가 아닌 네트워크 사용자로 로그인한 경우에는 바로 가기를 마우스 오른쪽 단추로 클릭하고 "관리자 권한으로 실행"을 선택하십시오.

   "시작", "모든 프로그램", "CA", "Federation Standalone", "서비스 중지"를 차례로 선택합니다.

   UNIX

   a. 명령 창을 엽니다.

   b. 다음 스크립트를 실행합니다.

      federation_install_dir/fedmanager.sh stop

      참고: 서비스를 중지했다가 루트 사용자로 시작하지 마십시오.

4. XPSImport 명령을 사용하며 모든 구성 데이터를 복원합니다.

   XPSImport *export_file_name* -passphrase *passphrase*

   export_file_name

   원래 구성을 내보낼 때 생성된 XML 파일의 이름을 지정합니다. 파일 이름은 **.xml** 확장명으로 끝나야 합니다.

   passphrase

   중요한 데이터의 암호를 해독하는 데 필요한 암호를 지정합니다. 이 암호는 8 자 이상이고 숫자, 대문자, 소문자를 각각 하나 이상 포함해야 합니다. 암호에 공백이 포함되어 있는 경우 암호를 따옴표로 묶어야 합니다.

5. 구성 마법사를 다시 실행합니다.

   원래 구성에 사용했던 것과 동일한 설정을 새로 구성하는 데이터베이스에도 사용합니다. 다음과 같은 설정이 포함됩니다.

   - 배포 모드
   - 포트 번호
   - 가상 호스트 이름
   - SiteMinder 커넥터

6. (선택 사항) 원래 구성에서 SiteMinder 커넥터를 사용한 경우 다음을 수행하여 커넥터를 다시 설정합니다.

   a. Administrative UI 에 로그인합니다.

   b. "인프라" 탭을 클릭하고 "배포 설정"을 선택합니다.

   c. 원래 구성에 사용했던 것과 동일한 값을 사용하여 커넥터 설정을 다시 구성합니다.

   d. "호스트 등록"을 클릭하여 정책 서버에 페더레이션 시스템을 등록합니다.

이제 구성이 원래 상태로 복원되었습니다.

제 30 장: 문제 해결

이 섹션은 다음 항목을 포함하고 있습니다.

시스템 성능 문제 해결

다음은 시스템 성능 문제 해결에 대한 설명입니다.

부하가 높은 경우의 세션 저장소 시간 만료 구성

부하가 높은 경우에는 만료되거나 유휴 시간이 만료된 세션을 제거하는 등의 세션 저장소 유지 관리 태스크에 필요한 장기 실행 쿼리가 시간 만료될 수 있습니다. MaintenanceQueryTimeout 레지스트리 설정 값을 늘려 세션 저장소 유지 관리 태스크에 대한 만료 시간(기본값 60 초)을 조정하십시오. 유지 관리 스레드가 작업을 성공적으로 완료할 수 있도록 값을 늘리십시오.

MaintenanceQueryTimeout 레지스트리 설정은 다음 레지스트리 위치에서 찾을 수 있습니다.

```
HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\Netegrity\SiteMinder\CurrentVersion\
SessionServer
```

프록시 엔진 중단 및 요청 처리 중지

증상

요청 처리를 며칠 간 계속한 후 CA SiteMinder® Federation Standalone 의 기본 프록시 엔진이 중지됩니다.

해결 방법

프록시 엔진의 server.conf 에서 Apache 웹 서버(HTTP 수신기)와 프록시 엔진(Tomcat 서블릿 엔진) 사이의 연결에 대한 성능 조정 매개 변수를 수정하십시오.

수정하는 매개 변수는 Apache JServ 프로토콜(AJP)을 사용하여 Apache 웹 서버와 Tomcat 사이의 통신이 가능하도록 Tomcat 커넥터 역할을 하는 구성 요소 mod_jk 에 사용됩니다.

server.conf 파일을 수정하려면

1. 다음 디렉터리로 이동합니다.

   federation_install_dir/secure-proxy/proxy-engine/conf

2. 편집기에서 server.conf 파일을 엽니다.
3. 다음 매개 변수를 수정합니다.

   worker.ajp13.reply_timeout

   프록시 엔진에서 받은 패킷 두 개 사이에 허용되는 간격의 최대 시간(밀리초)을 지정합니다. 이 시간이 경과하면 Apache 서버(HTTP 수신기)와 프록시 엔진 사이의 연결이 끊어집니다. 값이 0 이면 프록시 엔진은 응답을 수신할 때까지 무기한 기다립니다.

   프록시 엔진의 응답을 무기한 기다리지 않도록 하려면 이 값을 0 보다 큰 값으로 설정해야 합니다.

   기본값: 0

   worker.ajp13.retries

   통신 오류가 발생한 경우에 mod_jk 구성 요소에서 프록시 엔진에 연결 요청을 보내는 최대 횟수를 나타냅니다. 지정한 횟수만큼 요청을 해도 프록시 엔진에서 응답이 없으면 연결이 끊어집니다.

   연결 요청을 더 많이 시도하려면 이 값을 높게 설정하십시오.

   기본값: 2

4. server.conf 파일을 저장합니다.

서명 확인 오류 해결

악의적인 사용자가 서명을 무효화하지 않고 문서의 내용을 변경하여 XML 서명 래핑 공격을 실행할 수 있습니다. 기본적으로 정책 서버와 웹 에이전트 옵션 팩에 대한 소프트웨어 제어는 서명 래핑 공격을 방어하도록 설정되어 있습니다. 그러나 타사 제품이 XML 사양을 준수하지 않는 방식으로 XML 문서를 발생할 수도 있습니다. 따라서 기본 서명 검사 시 서명 확인 오류가 발생할 수 있습니다.

서명 확인 오류는 다음과 같은 이유로 발생합니다.

- XML 문서에 중복 ID 요소가 있고 서명이 이 중복 ID 를 참조하는 경우. 중복 ID 특성은 허용되지 않습니다.
- XML 서명이 예상되는 상위 요소를 참조하지 않고, 서명 래핑 취약점이 로깅된 경우

페더레이션 트랜잭션에 실패할 경우 smtracedefault.log 파일과 fwstrace.log 파일에서 서명 확인 오류가 있는지 검사하십시오. 이러한 오류는 수신된 XML 문서가 XML 표준을 준수하지 않음을 나타낼 수 있습니다. 이 문제를 해결하기 위해 서명 래핑 공격에 대한 정책 서버 및 웹 에이전트의 기본 보호 기능을 사용하지 않도록 설정할 수 있습니다.

중요! 서명 취약점에 대한 보호 기능을 사용하지 않도록 설정할 경우 이러한 공격으로부터 보호할 다른 방법을 결정하십시오.

XML 서명 래핑 검사를 사용하지 않도록 설정하려면

1. xsw.properties 파일로 이동합니다. 이 파일은 정책 서버와 웹 에이전트에서 서로 다른 위치에 있습니다.
   - 정책 서버 smtracedefault.log 파일의 오류 메시지를 보려면 *siteminder_home*/config/properties 로 이동합니다.
   - 웹 에이전트 fwstrace.log 의 오류 메시지를 보려면

     *web_agent_option_pack_ home/*affwebservices/web-INF/classes 로 이동합니다.

     참고: 웹 에이전트 옵션 팩이 웹 에이전트와 동일한 시스템에 설치되어 있는 경우 이 파일은 *web_agent_home* 디렉터리에 있습니다.

2. xsw.properties 의 다음 설정을 true 로 변경합니다.
   - DisableXSWCheck=true(정책 서버 설정만 해당)
   - DisableUniqueIDCheck=true(정책 서버 및 웹 에이전트 옵션 팩 설정)

     참고: DisableUniqueIDCheck 설정의 값은 정책 서버와 웹 에이전트 옵션 팩에 대해 동일해야 합니다.
3. 파일을 저장합니다.

같은 브라우저 세션을 사용하는 SSO 트랜잭션 두 개에서 장애 발생

증상

사용자가 같은 브라우저 세션에서 싱글 사인온 트랜잭션을 두 번 시도합니다. 두 트랜잭션은 같은 어설션 당사자 측에서 서로 다른 신뢰 당사자에게 보내는 것입니다. 첫 번째 트랜잭션은 성공하지만 두 번째 트랜잭션은 어설션 당사자 측에서 권한 부여 오류가 발생합니다. 장애가 발생하는 이유는 두 파트너 관계가 서로 다른 어설션 당사자 사용자 디렉터리를 사용하도록 구성되었기 때문입니다.

CA SiteMinder® Federation Standalone 이 어설션 당사자에 싱글 사인온 트랜잭션을 시작하면 브라우저에 세션 쿠키가 사용됩니다. 이 세션 쿠키에는 사용자 ID 와 어설션 당사자 사용자 디렉터리에 대한 정보가 들어 있습니다. 브라우저에는 CA SiteMinder® Federation Standalone 세션 쿠키가 한 번에 하나만 있을 수 있습니다.

사용자가 첫 번째 트랜잭션과 같은 브라우저 세션에서 두 번째 트랜잭션을 시도할 경우 첫 번째 트랜잭션의 세션 쿠키가 브라우저에 남아 있습니다. 그러나 이 세션 쿠키에는 두 번째 파트너 관계에 대한 올바른 정보가 들어 있지 않기 때문에 권한 부여 작업이 실패합니다.

해결 방법

각 파트너 관계의 어설션 당사자 사용자 디렉터리가 동일한 경우에만 브라우저 세션 하나에 서로 다른 싱글 사인온 트랜잭션을 사용하십시오.

각 파트너 관계에 구성된 어설션 당사자 사용자 디렉터리가 서로 다른 경우에는 첫 번째 브라우저 세션을 닫은 후 새 브라우저 세션을 시작하여 두 번째 트랜잭션을 시도하십시오.

보안 프록시 엔진 로그를 검사하여 시스템 문제 해결

파트너 관계 기반 CA SiteMinder?Federation Standalone 에는 트래픽을 백엔드 서버에 전달하는 보안 프록시 엔진이 포함되어 있습니다. 보안 프록시 엔진의 구성 요소는 다음과 같습니다.

- Apache 웹 서버

  들어오는 요청에 대한 HTTP 트래픽을 처리하는 HTTP 수신기 역할을 하며 적절히 구성될 경우 HTTPS 트래픽을 처리할 수 있습니다.

- Tomcat 서버

  Administrative UI 작동에 필요한 서블릿 컨테이너를 제공합니다. Apache 웹 서버는 mod_jk 라는 Tomcat 커넥터를 통해 Tomcat 서버와 통신합니다.

CA SiteMinder?Federation Standalone 환경에서 문제를 해결하기 위해 이러한 구성 요소와 관련된 로그 파일을 CA 지원 팀에 제공할 수 있습니다.

CA SiteMinder?Federation Standalone 문제 해결에 유용한 두 가지 Apache 로그는 다음과 같습니다.

mod_jk.log

mod_jk.log 는 제품에서 기본적으로 사용하도록 설정됩니다. 페더레이션 서버와 처음으로 연결된 후 정보가 이 파일에 로깅되기 시작됩니다.

이 로그 파일을 수정하려면

1. *federation_install_dir*\secure-proxy\httpd\conf 로 이동합니다.
2. httpd.conf 파일을 엽니다.
3. 파일에서 다음의 줄을 설정하여 아래와 같은 설정을 반영합니다.

   ```
   JkLogFile "path_to_mod_jk_log"
   JkLogLevel debug
   JkRequestLogFormat "%w %V %T %m %h %p %U %s"
   ```

   참고: "logs/mod_jk.log" 경로는 JkLogFile 항목의 기본 위치입니다. 기본값을 사용하거나 원하는 위치를 이 경로에 설정하십시오.

mod_jk.log 가 사용되지 않도록 설정하려면 파일에서 이러한 행을 주석 처리하거나 제거합니다.

httpclient.log

디버깅 용도로만 httpclient.log 가 사용되도록 설정할 수 있습니다. httpclient.log 파일은 *federation_install_dir*\secure-proxy\proxy-engine\logs 에 있습니다.

이 로그 파일을 수정하려면

1. *federation_install_dir*\secure-proxy\proxy-engine\conf 로 이동합니다.
2. server.conf 파일을 엽니다.
3. 다음 행을 변경합니다.

   ```
   httpclientlog="yes"
   ```

httpclient.log 파일의 위치와 로그 수준을 수정하려면 httpclientlogging.properties 파일을 편집합니다. 이 파일을 *federation_install_dir*\secure-proxy\Tomcat\properties 디렉터리에서 찾습니다.

제 31 장: 개방 형식 쿠키 정보

페더레이션 개방 형식 쿠키를 사용하면 응용 프로그램은 SiteMinder 에 사용자 특성을 어설션하고 SiteMinder 가 캡슐화하는 사용자 특성을 사용할 수 있습니다. 개방 형식 쿠키의 일반적인 특성은 다음과 같습니다.

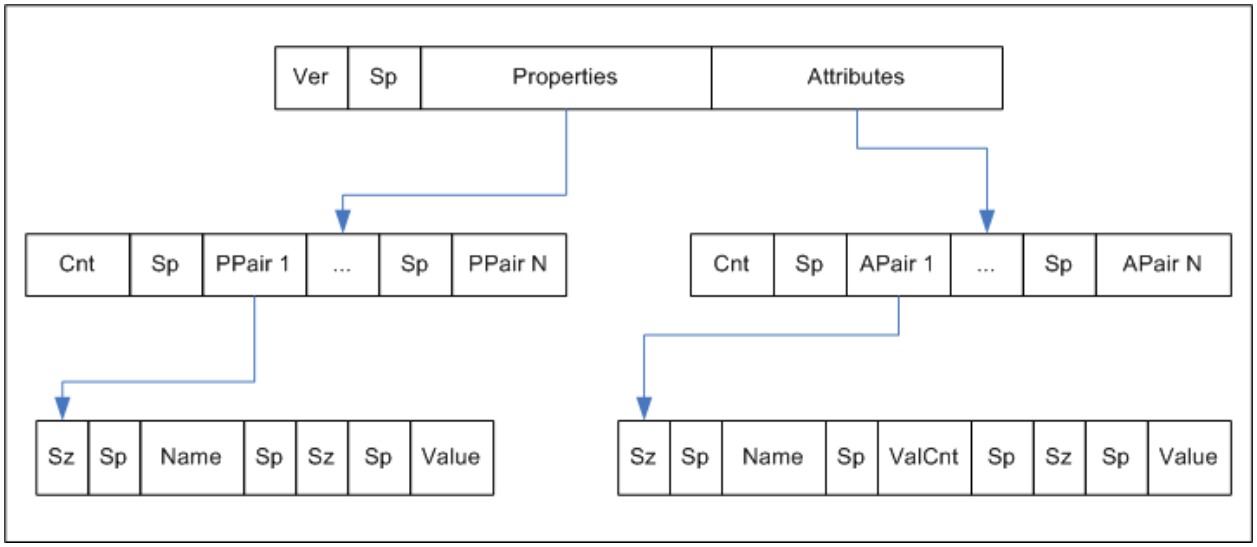
- 모든 프로그래밍 언어로 작성된 응용 프로그램이 쿠키를 사용할 수 있습니다.
- 쿠키 내용은 국제 문자 집합을 지원하는 UTF-8 바이트 문자열로 구성됩니다.
- 이름/값 쌍의 앞에 각 이름/값 쌍의 UTF-8 바이트 결합 크기가 나옵니다.
- 읽기 쉽도록 공백 문자가 추가됩니다.
- 쿠키는 구문 분석이 간단하며 쉽게 확장할 수 있습니다.

중요! 쿠키에 '='와 같은 안전하지 않은 문자가 포함될 경우 값을 큰따옴표로 묶으십시오. 사용자 인터페이스 또는 SDK 를 통해 이 옵션을 지정할 수 있습니다.

개방 형식 쿠키에는 다음의 속성 정보가 포함됩니다.

- 쿠키 버전
- 이름 ID
- 이름 ID 형식
- 세션 ID
- AuthnContext
- UserDN(사용자 ID 와 동일)

다음 다이어그램에서는 개방 형식을 보여 줍니다.



키:

- Ver - 쿠키 형식 버전. CA SiteMinder® Federation Standalone r12.1 의 경우 이 값은 1 입니다.
- Sp - ASCII 공백 문자. 가독성 향상의 목적으로만 사용됩니다.
- Properties - 프린서펄에 대한 정보입니다.
- Attributes - 어설션의 SAML 특성
- Cnt - 뒤에 나오는 이름 값 쌍의 수이며 ASCII 로 표현됩니다.
- Sz - 뒤에 나오는 이름 또는 값의 길이
- ValCnt - 뒤에 나오는 특성 값의 수입니다. CA SiteMinder® Federation Standalone r12.1 의 경우 특성에 대한 여러 개의 값이 지원됩니다. 이 값을 1 로 설정합니다.

이 형식의 BNF(Backus-Naur Form)는 다음과 같습니다(0*는 0 이상, 1*은 최소 1 을 의미).

- DIGIT = ASCII 숫자(0~9)
- CHAR = UTF-8 문자
- Sp = ASCII 공백(문자 32)

- Token = 1*CHAR
- Cookie = 버전 Sp 속성 특성
- Version = 1*DIGIT
- Cnt = 1*DIGIT
- Properties = Cnt 1*PPair
- Attributes = Cnt 0*APair
- ValCnt = 1*DIGIT
- PPair = Sz Sp 이름 Sp Sz Sp 값
- APair = Sz Sp 이름 Sp Sz Sp 값
- Sz = 1*DIGIT
- Name = 토큰

Value = 토큰

개방 형식 쿠키의 내용

페더레이션 개방 형식 쿠키를 사용하면 응용 프로그램은 CA SiteMinder?Federation Standalone 에 사용자 특성을 어설션하고 CA SiteMinder?Federation Standalone 이 캡슐화하는 사용자 특성을 사용할 수 있습니다. 개방 형식 쿠키의 일반적인 특성은 다음과 같습니다.

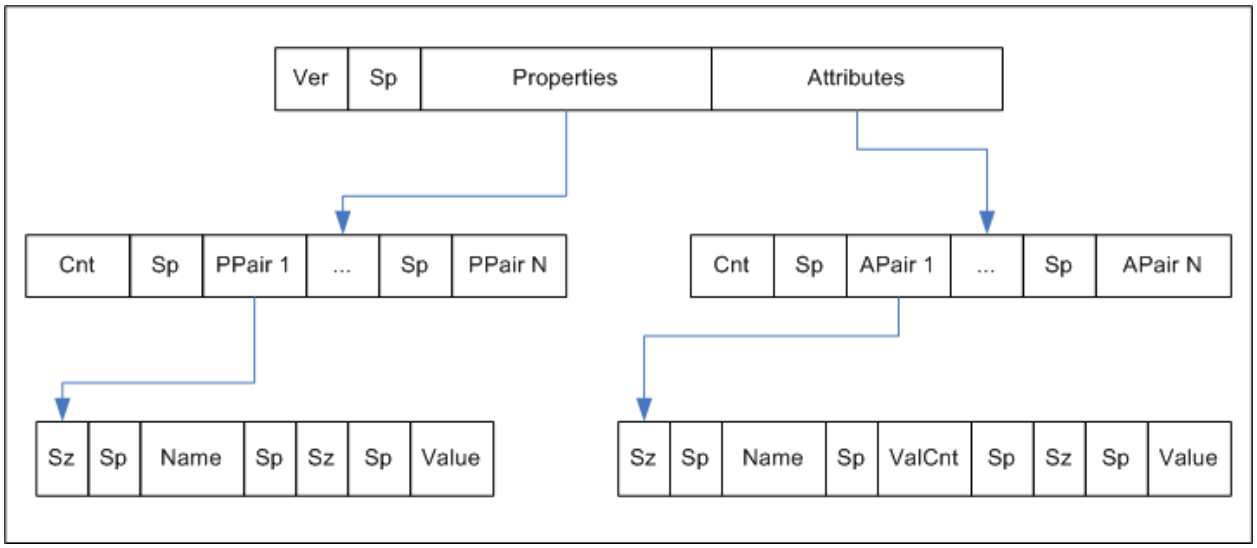
- 모든 프로그래밍 언어로 작성된 응용 프로그램이 쿠키를 사용할 수 있습니다.
- 쿠키 내용은 국제 문자 집합을 지원하는 UTF-8 바이트 문자열로 구성됩니다.
- 이름/값 쌍의 앞에 각 이름/값 쌍의 UTF-8 바이트 결합 크기가 나옵니다.
- 읽기 쉽도록 공백 문자가 추가됩니다.
- 쿠키는 구문 분석이 간단하며 쉽게 확장할 수 있습니다.

중요! 쿠키에 '='와 같은 안전하지 않은 문자가 포함될 경우 값을 큰따옴표로 묶으십시오. 사용자 인터페이스 또는 SDK 를 통해 이 옵션을 지정할 수 있습니다.

개방 형식 쿠키에는 다음의 속성 정보가 포함됩니다.

- 쿠키 버전
- 이름 ID
- 이름 ID 형식
- 세션 ID
- AuthnContext
- UserDN(사용자 ID 와 동일)
- UserConsent
- 로그인 ID
- ExpiresON(만료 시간)

다음 다이어그램에서는 개방 형식을 보여 줍니다.



키:

- Ver - 쿠키 형식 버전. 이 값은 1 입니다.
- Sp - ASCII 공백 문자. 가독성 향상의 목적으로만 사용됩니다.
- Properties - 프린서펄에 대한 정보
- Attributes - 어설션의 SAML 특성
- Cnt - 뒤에 나오는 이름 값 쌍의 수. ASCII 로 표현됩니다.

- Sz - 뒤에 나오는 이름 또는 값의 길이
- ValCnt - 특성 값의 수

이 형식의 BNF(Backus-Naur Form)는 다음과 같습니다(0*는 0 이상, 1*은 최소 1 을 의미).

- DIGIT = ASCII 숫자(0~9)
- CHAR = UTF-8 문자
- Sp = ASCII 공백(문자 32)
- Token = 1*CHAR
- Cookie = 버전 Sp 속성 특성
- Version = 1*DIGIT
- Cnt = 1*DIGIT
- Properties = Cnt 1*PPair
- Attributes = Cnt 0*APair
- ValCnt = 1*DIGIT
- PPair = Sz Sp 이름 Sp Sz Sp 값
- APair = Sz Sp 이름 Sp Sz Sp 값
- Sz = 1*DIGIT
- Name = 토큰

Value = 토큰

부록 A: 암호화 및 암호 해독 알고리즘

이 섹션은 다음 항목을 포함하고 있습니다.

개방 형식 쿠키 암호화 알고리즘

개방 형식 쿠키는 암호 기반 암호화에 대해 다음 옵션을 지원합니다.

FIPS_Compat 및 FIPS_Migration 모드

PBE/SHA1/AES/CBC/PKCS12PBE-1000-128

PBE/SHA1/AES/CBC/PKCS12PBE-1000-192

PBE/SHA1/AES/CBC/PKCS12PBE-1000-256

PBE/SHA256/AES/CBC/PKCS12PBE-1000-128

PBE/SHA256/AES/CBC/PKCS12PBE-1000-192

PBE/SHA256/AES/CBC/PKCS12PBE-1000-256

PBE/SHA1/3DES_EDE/CBC/PKCS12PBE-1000-3

PBE/SHA256/3DES_EDE/CBC/PKCS12PBE-1000-3

FIPS_Only 모드

AES128/CBC/PKCS5Padding

AES192/CBC/PKCS5Padding

AES256/CBC/PKCS5Padding

3DES_EDE/CBC/PKCS5Padding

디지털 서명 및 개인 키 알고리즘

CA SiteMinder?Federation Standalone 은 파트너 관계 서명 옵션에 다음 알고리즘을 사용합니다.

암호화 키 알고리즘

RSA-V15, RSA-OEAP

암호화 블록 알고리즘

3DES, AES-128, AES-256

CA SiteMinder?Federation Standalone 은 개인 키 생성에 다음 알고리즘(인증서/키)을 사용합니다.

키 알고리즘

RSA

서명 알고리즘

MD5withRSA, SHA1withRSA, SHA256withRSA 및 SHA512withRSA

백 채널 통신 알고리즘

HTTP-아티팩트 싱글 사인온 및 SAML 2.0 싱글 로그아웃에 사용되는 백 채널 통신에 대해 CA SiteMinder?Federation Standalone 은 FipsMode 에 따라 다음의 암호화를 지원합니다.

FIPS_Compat 및 FIPS_Migration 모드 - RC4 및 AES

RSA_With_RC4_SHA

RSA_With_RC4_MD5

RSA_With_AES_128_CBC_SHA

RSA_With_AES_256_CBC_SHA

FIPS_Only Mode - AES 만

RSA_With_AES_128_CBC_SHA

RSA_With_AES_256_CBC_SHA

백엔드 통신 알고리즘(SPS 서버)

백엔드 통신(SPS-백엔드 서버)에 대해서는 설정된 FipsMode 에 따라 다음과 같은 암호화가 지원됩니다. 이러한 암호화는 <fedroot>\secure-proxy\proxy-engine\conf\server.conf 에 정의됩니다.

FIPS_Compat 및 FIPS_Migration 모드

ciphers="-RSA_With_Null_SHA,+RSA_With_Null_MD5,-RSA_With_RC4_SHA,+RSA_With_RC4_MD5,+RSA_With_RC2_CBC_MD5,+RSA_With_DES_CBC_SHA,+RSA_With_DES_CBC_MD5,+RSA_With_3DES_EDE_CBC_MD5,+RSA_Export_With_RC4_40_MD5,-RSA_Export_With_DES_40_CBC_SHA,+RSA_Export_With_RC2_40_CBC_MD5,-DH_RSA_With_DES_CBC_SHA,-DH_RSA_With_3DES_EDE_CBC_SHA,-DH_RSA_Export_With_DES_40_CBC_SHA,-DH_DSS_With_DES_CBC_SHA,-DH_DSS_Export_With_DES_40_CBC_SHA,-DH_Anon_With_RC4_MD5,-DH_Anon_With_DES_CBC_SHA,-DH_Anon_With_3DES_EDE_CBC_SHA,-DH_Anon_Export_With_DES_40_CBC_SHA,-DH_Anon_Export_With_RC4_40_MD5,-DHE_RSA_With_DES_CBC_SHA,-DHE_RSA_Export_With_DES_40_CBC_SHA,-DHE_DSS_With_DES_CBC_SHA,-DHE_DSS_Export_With_DES_40_CBC_SHA,-Null_With_Null_Null"

FIPS_ONLY 모드

fipsciphers="+DHE_DSS_With_AES_256_CBC_SHA, +DHE_RSA_With_AES_256_CBC_SHA, +RSA_With_AES_256_CBC_SHA, +DH_DSS_With_AES_256_CBC_SHA, +DH_RSA_With_AES_256_CBC_SHA, +DHE_DSS_With_AES_128_CBC_SHA, +DHE_RSA_With_AES_128_CBC_SHA, +RSA_With_AES_128_CBC_SHA, +DH_DSS_With_AES_128_CBC_SHA, +DH_RSA_With_AES_128_CBC_SHA, +DHE_DSS_With_3DES_EDE_CBC_SHA, +DHE_RSA_With_3DES_EDE

Java SDK 암호화 알고리즘

CA SiteMinder?Federation Standalone Java SDK 는 다음의 암호화 알고리즘을 지원합니다.

암호 없음

"AES/CBC/PKCS5Padding"

암호 사용

"PBE/SHA1/AES/CBC/PKCS12PBE-5-128"

페더레이션 시스템 암호화 알고리즘

FMCrypto 암호화/암호 해독 알고리즘

AES_128

내부 키 암호화 알고리즘

CA SiteMinder?Federation Standalone 에는 사용하는 FIPS 작동 모드에 따라 다음과 같은 내부 키 암호화/암호 해독 알고리즘이 사용됩니다.

FIPS_MIGRATE 및 FIPS_ONLY 모드

AES_128

FIPS_COMPAT 모드

RC2

Apache 웹 서버 및 Administrative UI 의 SSL 키 알고리즘

CA SiteMinder?Federation Standalone 에서는 포함된 Apache 웹 서버 SSL 통신에 다음과 같은 알고리즘을 사용합니다.

Apache SSL 키 생성

SHA1withRSA

키 암호화

DES-EDE3-CBC

CA SiteMinder?Federation Standalone 에서는 Administrative UI 에 대한 SSL 통신에 다음 알고리즘을 사용합니다.

SSL 키 암호 암호화

aes-128-cbc