為Cisco Identity Service(IdS)安裝和配置Shibboleth Identity Provider(IdP)以啟用SSO

目錄

簡介

本檔案介紹OpenAM身份提供程式(IdP)上啟用單一登入(SSO)的配置。

Cisco IdS部署模式

產品	部署
UCCX	共住者
PCCE	與CUIC（思科統一情報中心）和LD（即時資料）共存
UCCE	與CUIC和LD共駐以進行2k部署。 獨立式，適用於4k和12k部署。

必要條件

需求

思科建議您瞭解以下主題：

- Cisco Unified Contact Center Express(UCCX)版本11.6或Cisco Unified Contact Center Enterprise版本11.6或Packaged Contact Center Enterprise(PCCE)版本11.6（如果適用）。

  附註：本文檔引用有關思科身份識別服務(IdS)和身份提供方(IdP)的配置。文檔在螢幕截圖和示例中引用UCCX，但是配置與思科身份識別服務(UCCX/UCCE/PCCE)和IdP相似。

採用元件

本文件所述內容不限於特定軟體和硬體版本。

本文中的資訊是根據特定實驗室環境內的裝置所建立。文中使用到的所有裝置皆從已清除（預設）的組態來啟動。如果您的網路運作中，請確保您瞭解任何指令可能造成的影響。

安裝

Shibboeth是一個開放原始碼專案，它提供單點登入功能，允許站點以隱私保護的方式對受保護線上資源的單獨訪問做出明智的授權決策。它支援安全斷言標籤語言(SAML2)。 IdS是SAML2客戶端，應該支援Shibboeth，只需對IdS進行極少更改或不更改。在11.6中，IdS適用於使用Shibboleth IdP。

> **附註**：本文檔引用Shibboleth 3.3.0版作為SSO資格認證的一部分

系統要求

元件	**詳細資料**
Shibboeth版本	v3.3.0
下載位置	http://shibboleth.net/downloads/identity-provider/
安裝平台	Ubuntu 14.0.4 java版本"1.8.0_121"
輕量型目錄存取通訊協定(LDAP)版本	Active Directory 2.0
Shibboleth Webserver	Apache Tomcat/8.5.12

請參閱Shiboleth安裝維基

https://wiki.shibboleth.net/confluence/display/IDP30/Installation

設定

與LDAP伺服器整合

要將LDAP伺服器與shibboleth整合，需要在**$shibboleth_home**/conf/ldap.properties 中更新欄位，其中**$shibboleth_home**（預設為/opt/shibboleth-idp）是指安裝shibboleth時使用的安裝目錄。

欄位	預期值
idp.authn.LDAP.trustCertificates	用於載入信任錨點的資源，通常是${idp.home}/credentials中的本地檔案 其中idp.home是在setenv.sh中匯出為JAVA_OPTS的環境變數
idp.authn.LDAP.trustStore	用於載入包含信任錨點的Java金鑰庫的資源，通常是位於%{idp.home}/
idp.authn.LDAP.returnAttributes	需要返回的LDAPAattributes的逗號分隔清單。如果要返回所有屬性，請
idp.authn.LDAP.baseDN	需要執行LDAP搜尋的基本DN
idp.authn.LDAP.subtreeSearch	是否遞迴搜尋
idp.authn.LDAP.userFilter	LDAP搜尋篩選器
idp.authn.LDAP.bindDN	執行搜尋時要繫結的DN
idp.authn.LDAP.bindDNCredential	執行搜尋時要繫結的密碼
idp.authn.LDAP.dn格式	用於生成要進行身份驗證的使用者DN的格式字串

idp.authn.LDAP.authenticator	控制如何對LDAP進行身份驗證的工作流程
idp.authn.LDAP.ldapURL	LDAP目錄的連線URI

如需詳細資訊，請參閱：

https://wiki.shibboleth.net/confluence/display/IDP30/LDAPAuthnConfiguration

示例配置檔案

```
#
#idp.authn.LDAP.responseTimeout = PT3S
## SSLjvmTrustcertificateTrustkeyStoreTrust
#idp.authn.LDAP.sslConfig = certificateTrust
##certificateTrust
idp.authn.LDAP.trustCertificates =
%{idp.home}/credentials/ldap-server.crt
##keyStoreTrusttruststore
idp.authn.LDAP.trustStore =
%{idp.home}/credentials/ldap-server.truststore
##
#idp.authn.LDAP.returnAttributes =
userPrincipalName sAMAccountName
idp.authn.LDAP.returnAttributes = *
## DN##
#DNanonSearchAuthenticatorbindSearchAuthenticator
# AD:CN=UsersDC=exampleDC=org
idp.authn.LDAP.baseDN = CN=usersDC=ciscoDC=com
idp.authn.LDAP.subtreeSearch = true
*idp.authn.LDAP.userFilter
=(sAMAccountName={user})*
#
# AD:idp.authn.LDAP.bindDN=adminuser@domain.com
idp.authn.LDAP.bindDN =@cisco.com
idp.authn.LDAP.bindDNCredential =@123
#DNdirectAuthenticatoradAuthenticator
# ADidp.authn.LDAP.dnFormat=%s@domain.com
#idp.authn.LDAP.dnFormat =
%s@adfsserver.cisco.com
# LDAPattribute-resolver.xml
# V2
idp.attribute.resolver.LDAP.ldapURL =
%{idp.authn.LDAP.ldapURL}
idp.attribute.resolver.LDAP.connectTimeout =
%{idp.authn.LDAP.connectTimeout:PT3S}
idp.attribute.resolver.LDAP.responseTimeout =
%{idp.authn.LDAP.responseTimeout:PT3S}
idp.attribute.resolver.LDAP.baseDN =
%{idp.authn.LDAP.baseDN:undefined}
idp.attribute.resolver.LDAP.bindDN =
%{idp.authn.LDAP.bindDN:undefined}
idp.attribute.resolver.LDAP.bindDNCredential =
%{idp.authn.LDAP.bindDNCredential:undefined}
idp.attribute.resolver.LDAP.useStartTLS =
%{idp.authn.LDAP.useStartTLS:true}
idp.attribute.resolver.LDAP.trustCertificates =
%{idp.authn.LDAP.trustCertificates:undefined}
idp.attribute.resolver.LDAP.searchFilter
=(sAMAccountName=$resolutionContext.principal)
```

允許來自所有客戶端的請求

為確保來自所有客戶端的請求都能到達，需要在「$shibboleth_home/conf/access-control.xml」中進行更改

```
<entry key="AccessByIPAddress">
<bean id="AccessByIPAddress" parent="shibboleth.IPRangeAccessControl"
p:allowedRanges="#{ {'127.0.0.1/32','0.0.0.0/0', '::1/128', '10.78.93.103/32'} }" />
</entry>
```

將'0.0.0.0/0'新增到允許的範圍。這允許來自任何ip範圍的請求。

配置Shibboeth以與IdS整合

IdS中的安全雜湊演演算法(SHA1)和加密配置

要將Id配置為預設的SHA1，請開啟「**$shibboleth_home/conf/idp.properties**」並設定：

idp.signing.config = shibboeth.SigningConfiguration.SHA1

此配置也可更改：

idp.encryption.optional = true

如果將其設定為true，則當啟用時，找不到要使用的加密金鑰不會導致請求失敗。此h有助於進行「機會式」加密，即儘可能進行加密（在對等體的後設資料中發現要加密的相容金鑰），但會跳過加密。

將uid和user_principal配置為SAML響應

在**"$shibboleth_home/conf/attribute-resolver.xml"中新增AttributeDefinition以將sAMAccountName和userPrincipalName對映到SAML響應中的to uid和user_principal。**

此外，新增帶有<DataConnector>標籤的LDAP聯結器設定。

註意：需要使用值「sAMAccountName userPrincipalName」**指定**ReturnAttributes。

註：如果與Active Directory(AD)整合，則必須使用LDAPProperty。

在"$shibboleth_home/conf/attribute-filter.xml"**中合併更改**

更改 「**$shibboleth_home/conf/saml-nameid.xml」以包含**

IdP後設資料

IdP後設資料在資料夾「$shibboleth_**home/metadata」中可**用。 idp-metadata.xml檔案可以通過應用程式程式設計介面(API)上載到IdS

PUT https://<idshost>:<idsport>/ids/v1/config/idpmetadata

其中**disport**不可配置實體，且值為**"8553"**

> **警告**：Shibboeth後設資料**可**包含2個簽名證書、常規簽名證書和backchannel。導航到**"$shibboleth_home/credentials"**中的**idp-backchannel.crt**檔案以標識後台證書**。**如果後台證書在後設資料中可用，則應在上載到IdS之前從後設資料xml中刪除後台證書。這是因為IdS使用的Fedlet 12.0庫僅支援後設資料中的一個證書。如果有多個簽名證書可用，則Fedlet使用第一個可用證書。

配置後設資料提供程式

我們需要使用$shibboleth_home/metadata-providers.xml中的條目配置後設資料**提供程式。**

```
<MetadataProvider id="smart-86"  xsi:type="FilesystemMetadataProvider"
metadataFile="/opt/shibboleth-idp/SP/sp.xml"/>
```

其中**「id」屬性**可以是任何唯一名稱。

此條目表示已使用給定ID註冊後設資料提供程式，並且後設資料在指定的檔案/opt/shibboleth-idp/SP/sp.xml中可用。

必須將IdS的Service Provider(SP)後設資料複製到條目中指定的後設資料檔案。

> **注意**：可以通過**GET https://<idshost>:<idsport>/ids/v1/config/spmetadata 檢索ID的SP後設資料，其中idsport不是可配置實體，其值為「8553」。**

SSO的進一步配置

本文檔從SSO的IdP方面描述了要與思科身份服務整合的配置。有關詳細資訊，請參閱各個產品配置指南：

- UCCX
- UCCE
- PCCE