Shibboleth IdPの構築

山本哲寛[*]，永井謙芝[**]，髙井昌彰[*]

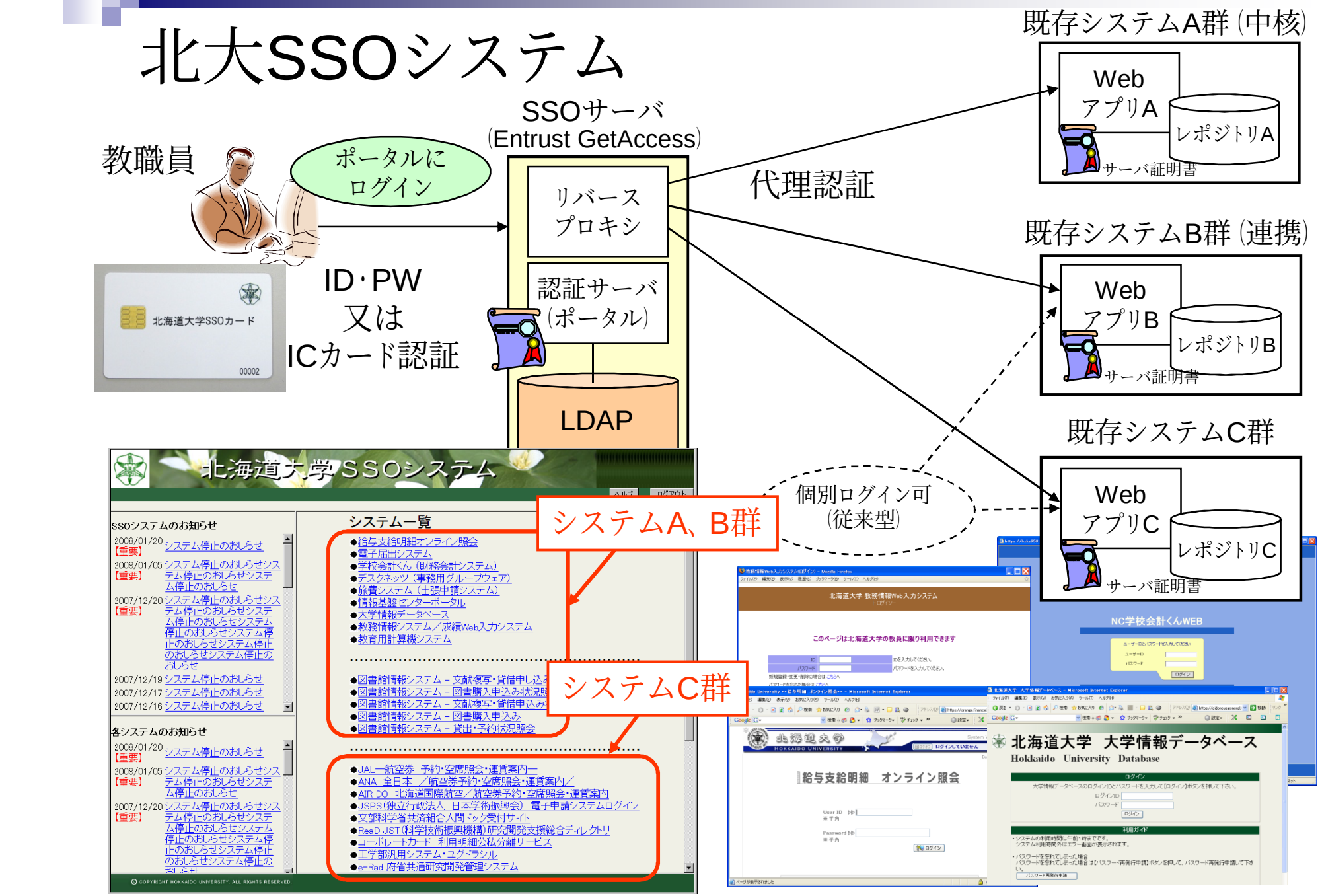
*北大情報基盤センター

**北大企画部情報基盤課ネットワークチーム

北海道大学におけるShibboleth実証実験

- IdPの構築
  - 廉価な機器による実装
  - ID/Password認証連携の実証試験
  - PKI認証連携の実証試験
- プライベート認証局の利用
  - 専用のプライベートCAを新設し、IdPサーバ証明書を発行
  - クライアント証明書は既設のプライベートCAから発行
- 既設の無線LAN認証テストベッドの有効活用
  - 新規に構築したサーバは1台（IdP01+IdP02+CA）
  - eduroam用クライアント証明書を流用（新規発行なし）
  - 既設LDAPサーバを利用

北大SSOシステム


既存システムA群（中核）
Web
アプリA
レポジトリA
サーバ証明書
SSOサーバ
(Entrust GetAccess)
教職員
ポータルに
ログイン
リバース
プロキシ
代理認証
認証サーバ
（ポータル）
ID・PW
又は
ICカード認証
北海道大学SSOカード
00002
LDAP
既存システムB群（連携）
Web
アプリB
レポジトリB
サーバ証明書
既存システムC群
Web
アプリC
レポジトリC
サーバ証明書
個別ログイン可
（従来型）
システムA、B群
システムC群
北海道大学SSOシステム
SSOシステムのお知らせ
各システムのお知らせ
システム一覧
●給与支給明細オンライン照会
●電子届出システム
●学校会計くん（財務会計システム）
●デスクネッツ（事務用グループウェア）
●旅費システム（出張申請システム）
●情報基盤センターポータル
●大学情報データベース
●教務情報システム／成績Web入力システム
●教育用計算機システム
●図書館情報システム－図書購入申込
●図書館情報システム－貸出・予約状況照会
●JAL一航空券 予約・空席照会・運賃案内一
●ANA 全日本 ／航空券予約・空席照会・運賃案内／
●AIR DO 北海道国際航空／航空券予約・空席照会・運賃案内
●JSPS（独立行政法人 日本学術振興会） 電子申請システムログイン
●文部科学省共済組合人間ドック受付サイト
●ReaD JST（科学技術振興機構）研究開発支援総合ディレクトリ
●コーポレートカード 利用明細公私分離サービス
●工学部汎用システム・ユグドラシル
●e-Rad 府省共通研究開発管理システム
北海道大学 教務情報Web入力システム
このページは北海道大学の教員に限り利用できます
NC学校会計くんWEB
給与支給明細 オンライン照会
北海道大学 大学情報データベース
Hokkaido University Database

Shibboleth実証実験システム

北大SSOと
連携（予定）

LDAP

IdP01, IdP02

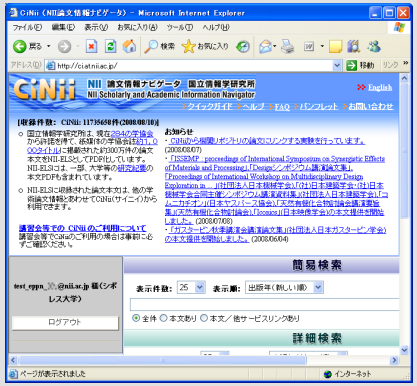
Shibboleth専用
プライベート認証局

サーバ証明書

HP DL380 / Vmware

認証

他機関のSP



SINET3

HINES
backbone

無線LAN用VLAN

Eduroam対応AP

L3SW

無線AP

AP-5100AW

無線AP

LDAPサーバ

RADIUSサーバ

プライベート認証局

無線LAN認証ローミングテストベッド

Shibboleth環境に接続

- ID/Password認証
  または
- PKI認証

附属図書館利用者に対し
テストIDを配布（未定）

クライアント
証明書発行

eduroam利用も可能

eduroam

クライアント証明書

SSOカードと
連携（予定）

実証実験に使用した証明書

	Shibboleth専用プライベートCA発行		既設プライベートCA発行	
	CA証明書	サーバ証明書	CA証明書	クライアント証明書
Issuer	C=JP ST=Hokkaido L=Sapporo O=Hokkaido University OU=Information Initiative Center CN=Hokkaido University Private CA for Shibboleth	C=JP ST=Hokkaido L=Sapporo O=Hokkaido University OU=Information Initiative Center CN=Hokkaido University Private CA for Shibboleth	C=JP ST=Hokkaido L=Sapporo O=Hokkaido University CN=Hokkaido University Test CA	C=JP ST=Hokkaido L=Sapporo O=Hokkaido University CN=Hokkaido University Test CA
Subject	C=JP ST=Hokkaido L=Sapporo O=Hokkaido University OU=Information Initiative Center CN=Hokkaido University Private CA for Shibboleth	C=JP O=Hokkaido University OU=Information Initiative Center L=Academe CN=idp01.iic.hokudai.ac.jp	Subject: C=JP ST=Hokkaido L=Sapporo O=Hokkaido University CN=Hokkaido University Test CA	C=JP O=Pentio OU=network OU=iic OU=hokudai OU=Certificate by PentioPKI PrivateCA CN=et-yamamoto emailAddress=et-yamamoto@iic.hokudai.ac.jp

IdP構築時に困ったこと

- 初期構築時
  - CiNiiにはログインできても、ploneにログインできない
    - 「eduPersonPrincipalName」が渡っていなかった

原因：　attribute-filter.xml内のAttributeRuleエレメント：attributeID属性と
attribute-resolver.xml内のAttributeDefinitionのid属性の値について
大文字・小文字の区別を間違えて記述していた

- 証明書認証移行時
  - 既設LDAPサーバをできるだけいじくりたくない

～「eduPersonPrincipalName」を追加したくない～

解決策：
LDAPに格納されていた情報「sn」を、「eduPersonPrincipalName」に格納

Attribute-resolver.xmlの変更点

変更前（395行目あたり）

```
<resolver:AttributeDefinition id="principalName" xsi:type="Scoped"
    xmlns="urn:mace:shibboleth:2.0:resolver:ad"
      scope="iic.hokudai.ac.jp"
    sourceAttributeID="eduPersonPrincipalName">
      <resolver:Dependency ref="remoteUser" />
```

変更後（395行目あたり）

```
<resolver:AttributeDefinition id="principalName" xsi:type="Scoped"
    xmlns="urn:mace:shibboleth:2.0:resolver:ad"
      scope="iic.hokudai.ac.jp" sourceAttributeID="sn">
      <resolver:Dependency ref="remoteUser" />
```

IdP構築のマニュアルにない注意点

●JDKインストール後作業としてjdk6をインストールした後で証明書を取り込まなくてはならない。しかし実際には、下記の作業をしないと、発行した証明書はうまく取り込めない。（jdkで利用する証明書は、適当にjdk付属のkeytoolで作成しても、動作はしてしまう）

●jdkのKeystoreへの証明書･秘密鍵の取り込み方法

（1）PKCS#12ファイルの作成

```
# openssl pkcs12 -export
        -inkey「サーバ証明書用CSRを作成した際に利用した秘密鍵ファイル」
        -in「サーバ証明書ファイル」-certfile「ＣＡ証明書ファイル」
        -out「PKCS#12形式ファイル」
```

（2）PKCS#12ファイルの取り込み

```
# keytool -list -keystore「上記で作成したPKCS#12形式ファイル」
        -storetype pkcs12
```

（PKCS#12ファイルの中身を確認できます）

IdP構築のマニュアルにない注意点（続き）

```
# keytool -importkeystore -srckeystore「上記で作成したPKCS#12形式ファイル」
        -srcstoretype pkcs12 -destkeystore /usr/java/tomcat/conf/keystore
        -deststoretype jks
```

（PKCS#12ファイルをjdkのkeystoreに取り込みます。
この例ではkeystoreファイルは/usr/java/tomcat/conf/keystoreとしています）

```
# keytool -list -keystore /usr/java/tomcat/conf/keystore
```

（jdkのkeystoreに取り込めたかどうかを確認できます）

●参考URL

http://mage.oops.jp/pyuki/wiki.cgi?keytool
http://mage.oops.jp/pyuki/wiki.cgi?pcks8Tojks

●その他、PrivateCAからサーバ証明書を発行した場合、NIIのCA証明書とPrivateCAのCA証明書の両方を/opt/shibboleth-idp/credentials配下に置かないと動作しない。

まとめと課題

- ID/Password認証連携及びPKI認証連携に成功
  - ID/Password認証連携（H20.9.5）、PKI認証連携（H20.9.26）
- Shibboleth（IdP）自体の導入は、さほど難しくない
  - NII提供のマニュアル＋MailによるQ&Aでほぼ解決
- 特に高性能なサーバを必要としない
  - 接続試験程度であれば廉価なNote PCでも十分に機能
  - 現在は、次年度の実運用を目指し、HP DL380にシステム移行済み（H21.3）

- 主に附属図書館利用者に対しテストIDを配布（未定）
  - 利用者に使用感などのアンケート実施を計画
- 他機関との時刻同期の問題
  - 時刻同期不備のため、NII提供のploneに接続できなくなることが多々あった
- 実運用に向けた方針の検討
  - 既存システム（LDAP）との安全な連携方法の検討と実装
    - 大学職員ID管理とShibboleth連携におけるID管理
      （既設LDAP内の情報とeduPersonPrincipalNameの関係等）