情報処理技術セミナー「Shibboleth環境の構築」

概要説明

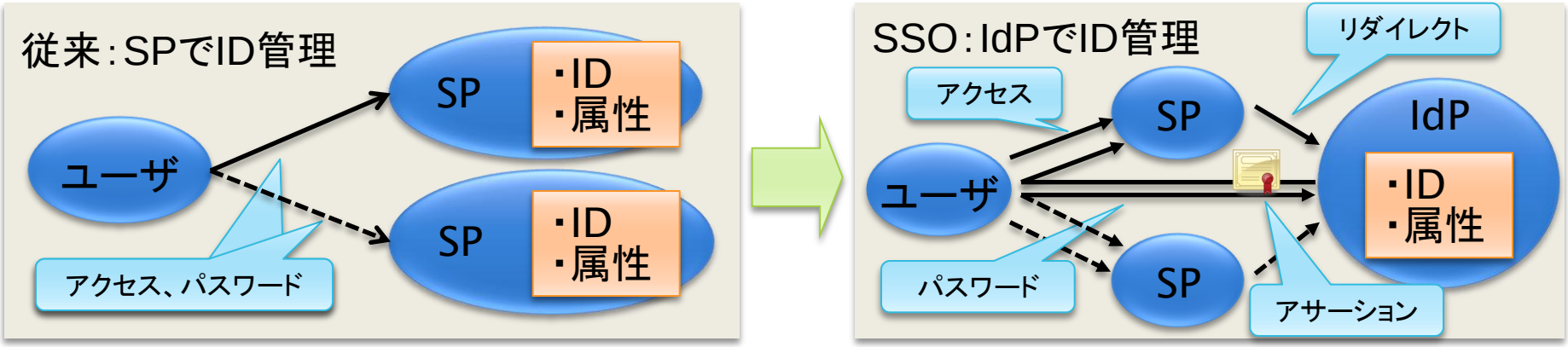
セミナーの目的と内容

- 目的
  - 学術認証フェデレーションへの参加に必要な
    Shibbolethに基づくIdPとSPの構築、運用の基本を理解する

- 内容
  - 1日目：“IdP”の構築実習
    - jdk、tomcat、Shibbolethのインストール
    - 設定、接続テスト
  - 2日目：“SP”の構築実習
    - Apache、Shibbolethのインストール（shibd, mod_shib）
    - 設定、接続テスト

学術認証フェデレーションとは

- 学術認証フェデレーションとは
  - 定められた規程（ポリシー）を信頼しあうことで，相互に認証連携を実現し，学術リソースを利用・提供する機関や組織から構成された連合体のこと
  - 機関（IdP）がIDと属性を管理し，サービス提供者（SP）がそれを利用して認可
- プライバシ保護を考慮したシングルサインオン（SSO）技術
  - ユーザのユニークネスを保証しつつ個人情報は出さない
  - SPは必要な情報のみをIdPに要求
  - ユーザは各SPに対する各属性の公開を制御可能



Shibboleth（シボレス）

- 米国EDUCAUSE／Internet2にて2000年に発足したプロジェクト
  - http://shibboleth.internet2.edu/
- SAML、eduPerson等の標準仕様を利用した，認可のための属性交換を行う標準仕様とミドルウェア （オープンソースソフトウェア）
- 米国、欧州でShibbolethによるFederationが運用、拡大
- バージョン1.3系と2.0系が広く利用されている（プロトコルが少し異なる）
  - 最新はShibboleth V2.1

cf.

- 欧州（特に北欧）では，simpleSAMLphpも利用されて
  - ノルウェーUNINETT
    - http://rnd.feide.no/simplesamlphp
  - 日本語化プロジェクト
    - http://sourceforge.jp/projects/ssp-japan/

フェデレーション構築に必要なサーバ

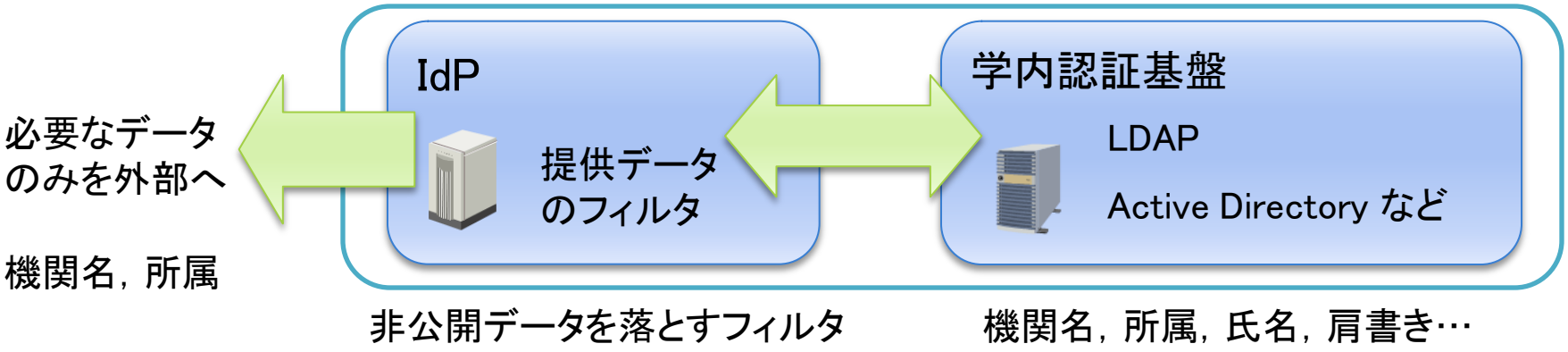
- IdP（Identity Provider）　大学（サービス利用者側）が用意
  - フェデレーション内に構成員の情報を提供するサーバ
  - フェデレーションに参加する大学等が構築

- SP（Service Provider）　大学他（サービス提供側）が用意
  - 認証を受けた人に対してサービスを行うサーバ
  - 電子ジャーナル，データベース，E-ラーニング等
    Webベースのシステムであれば何でも可

- DS（Discovery Service）　フェデレーションが用意
  - SPへのアクセスの際にIdPを検索するシステム
  - フェデレーションが運用
  - ここに名前がのることにより「フェデレーションに参加」
  - WAYF (Where Are You From) サービスとも呼ばれる(Shib 1.x)

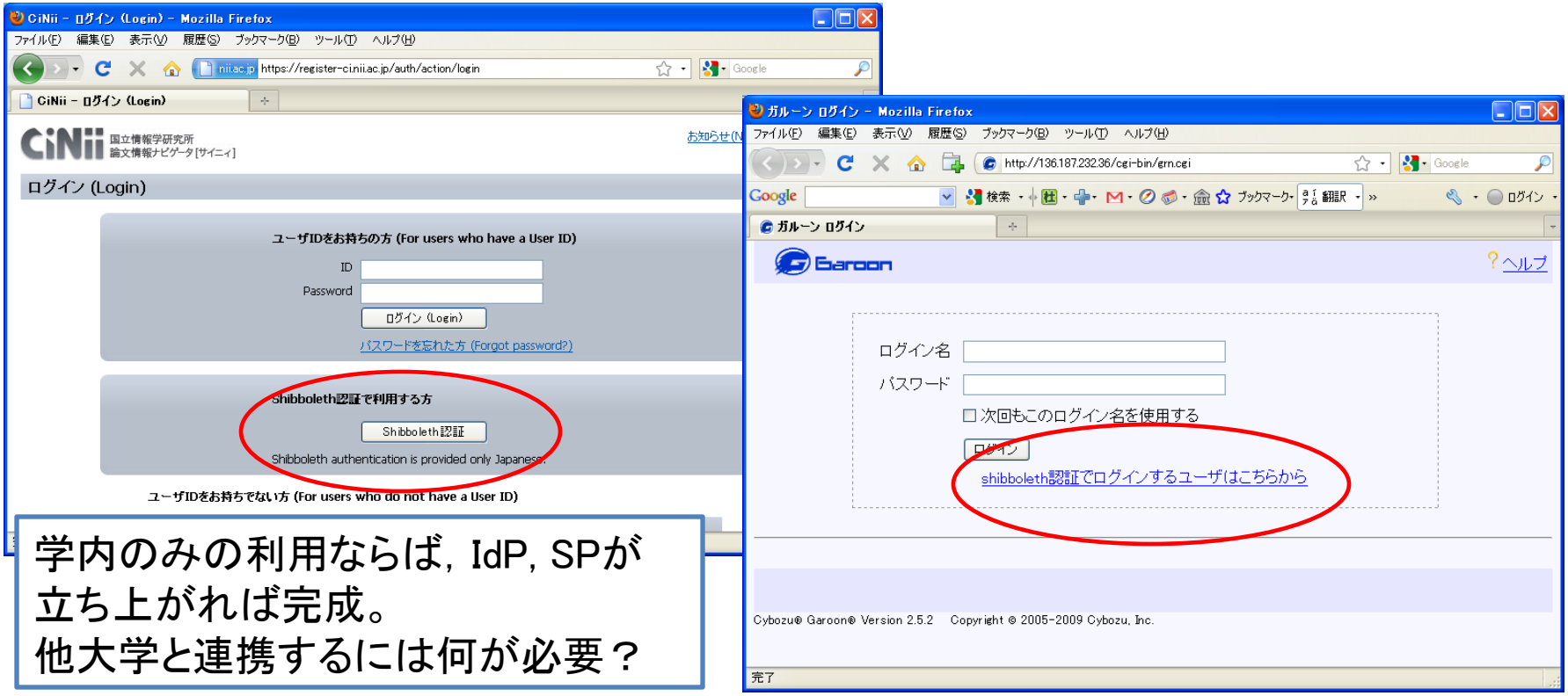
IdP (Identity Provider)とは

- フェデレーション内に情報を提供するサーバであり，大学等が構築
- IdP自身は情報を持たない
- 情報はLDAPやActive Directory等，既存の認証基盤を参照
- IdPは単なるフィルタであり，学内認証基盤から特定のデータのみを抽出して提供する
- 公開できるデータの制御が可能である
  - このため，Shibbolethはしばしば個人情報保護に優れていると言われるが，サーバ自体がハッキングに強固という意味ではない。
  - 慎重な操作が必要なのは，LDAPやActive Directoryと同じ



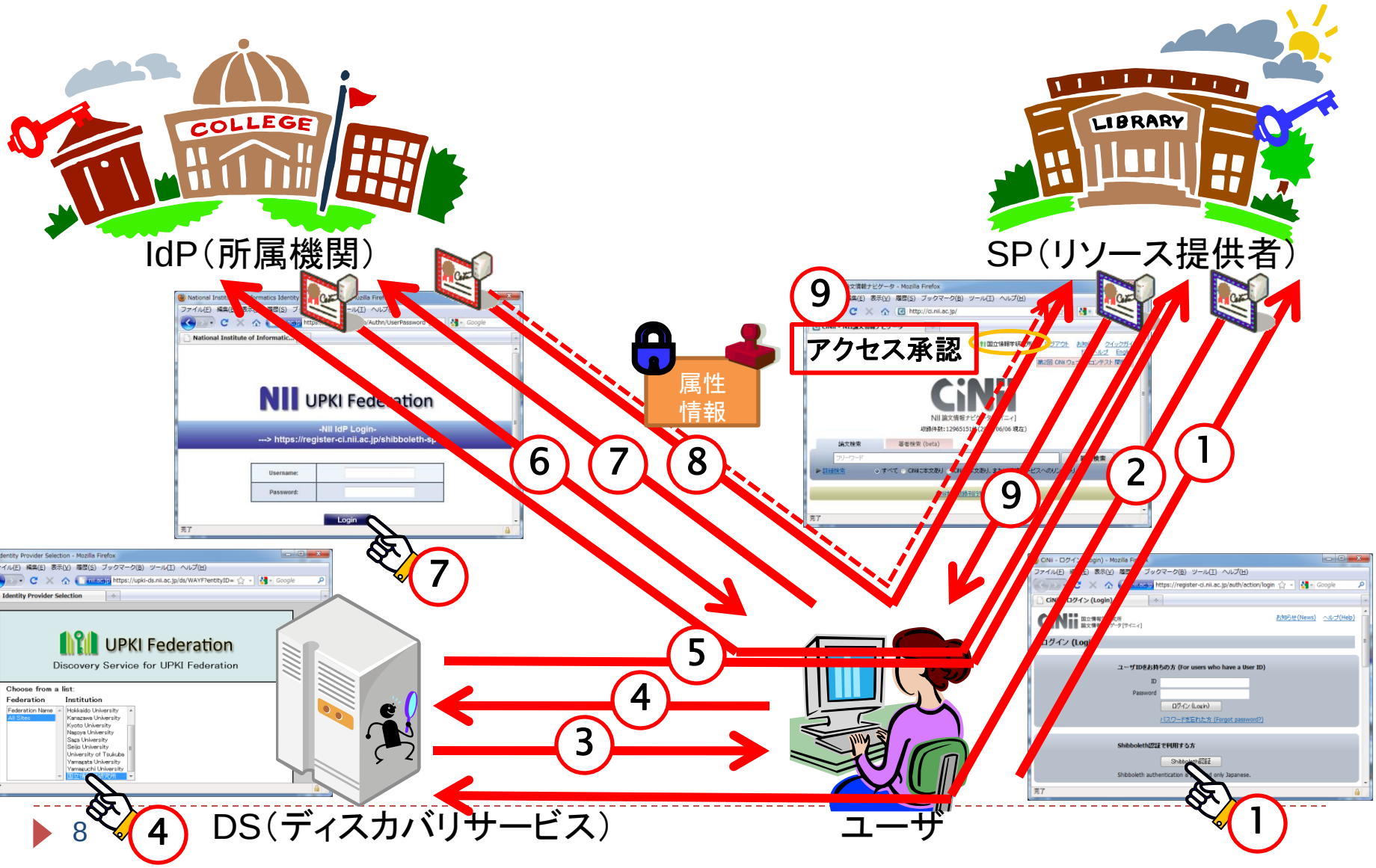
SP (Service Provider)とは

- サービスを提供するWebサーバのこと
- “シボレスログイン”等のボタンがあればShibbolethで利用可能なSPである
- 電子ジャーナルに限らず，いろいろなサービスをShibboleth化することが可能（例：無線LAN認証，サイボウズ）

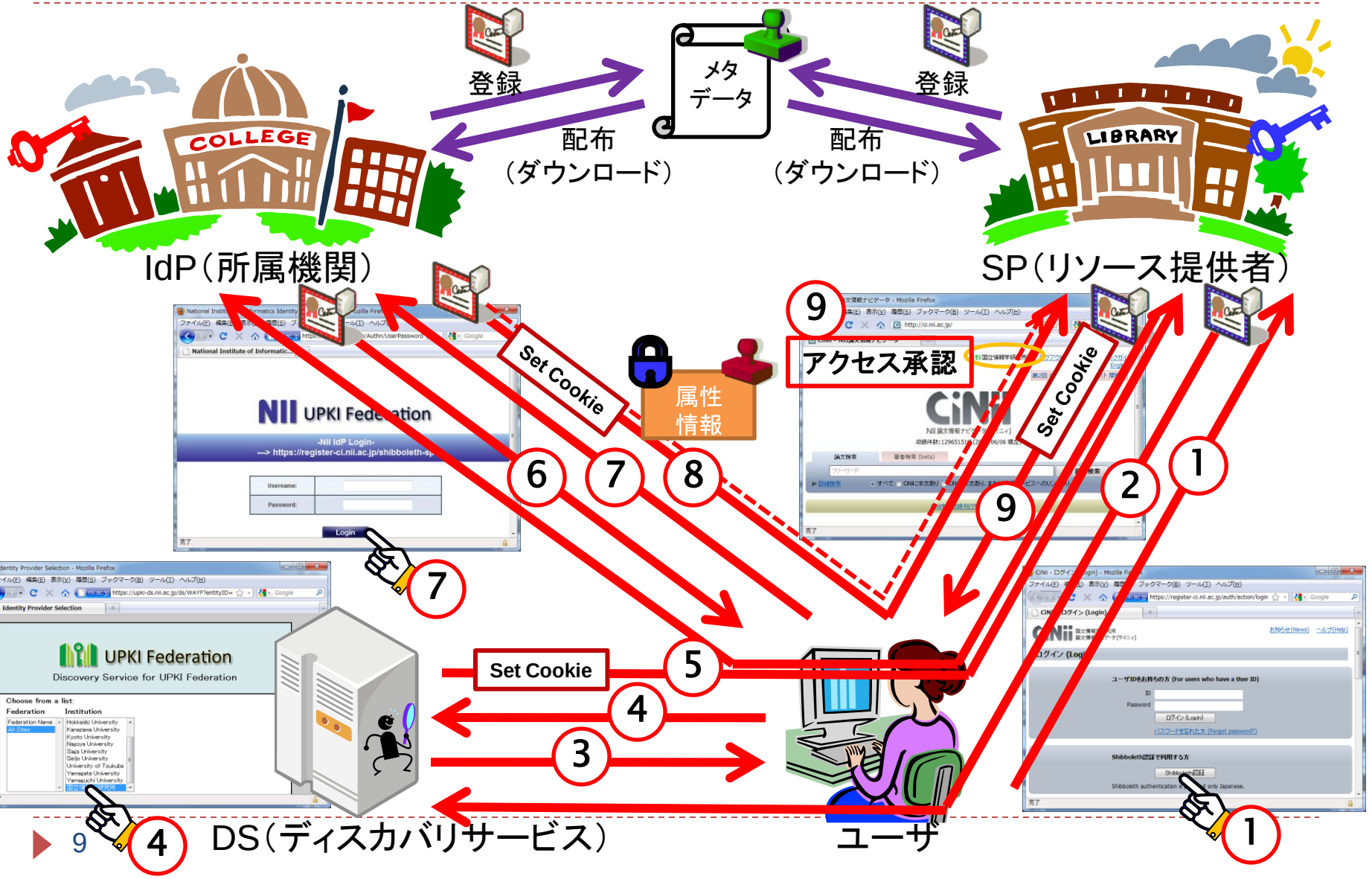


学内のみの利用ならば，IdP，SPが立ち上がれば完成。
他大学と連携するには何が必要？

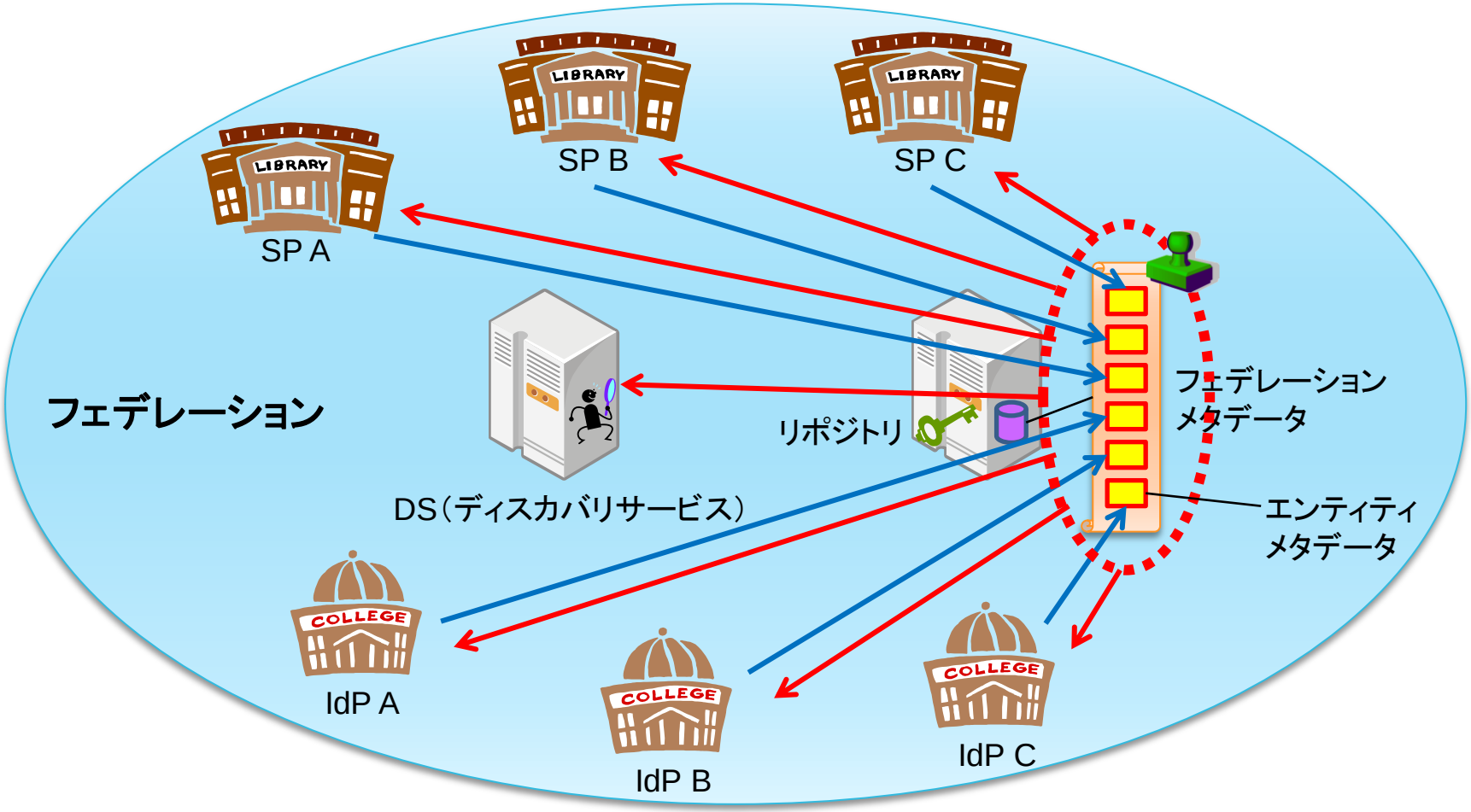
Shibbolethの基本動作



Cookieによる処理の記憶



メタデータを用いた信頼の構築

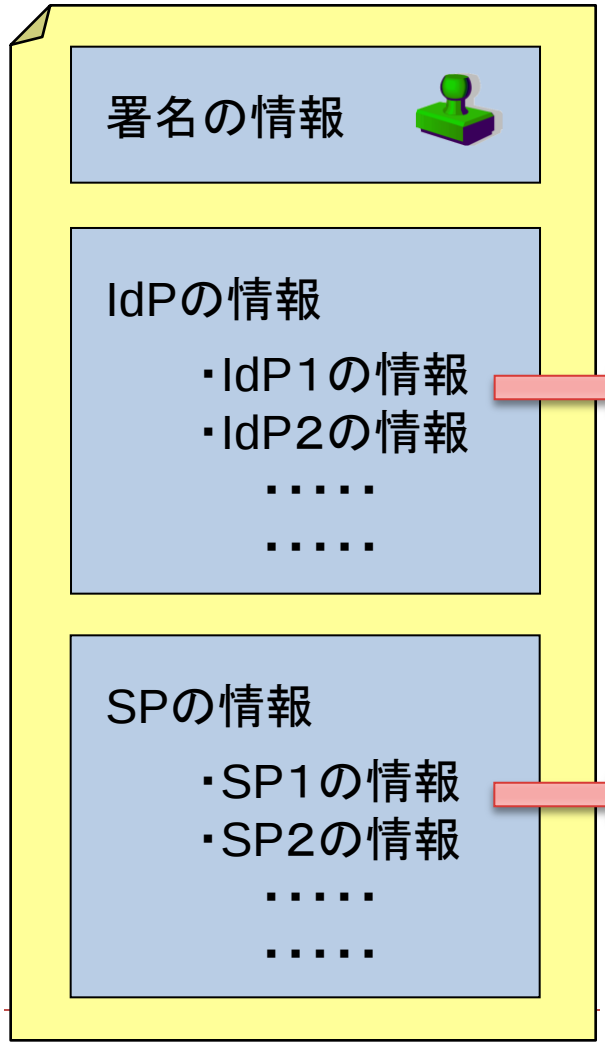


自動ダウンロードするフェデレーションメタデータの信頼性は、フェデレーションの証明書 で担保（事前に入手・検証し、事前にIdP/SPにインストール）

（検証は、別チャンネルで入手したfingerprintとの比較等による）

メタデータ(XML形式)の構成

フェデレーションメタデータ ≒ relying party（信頼関係）



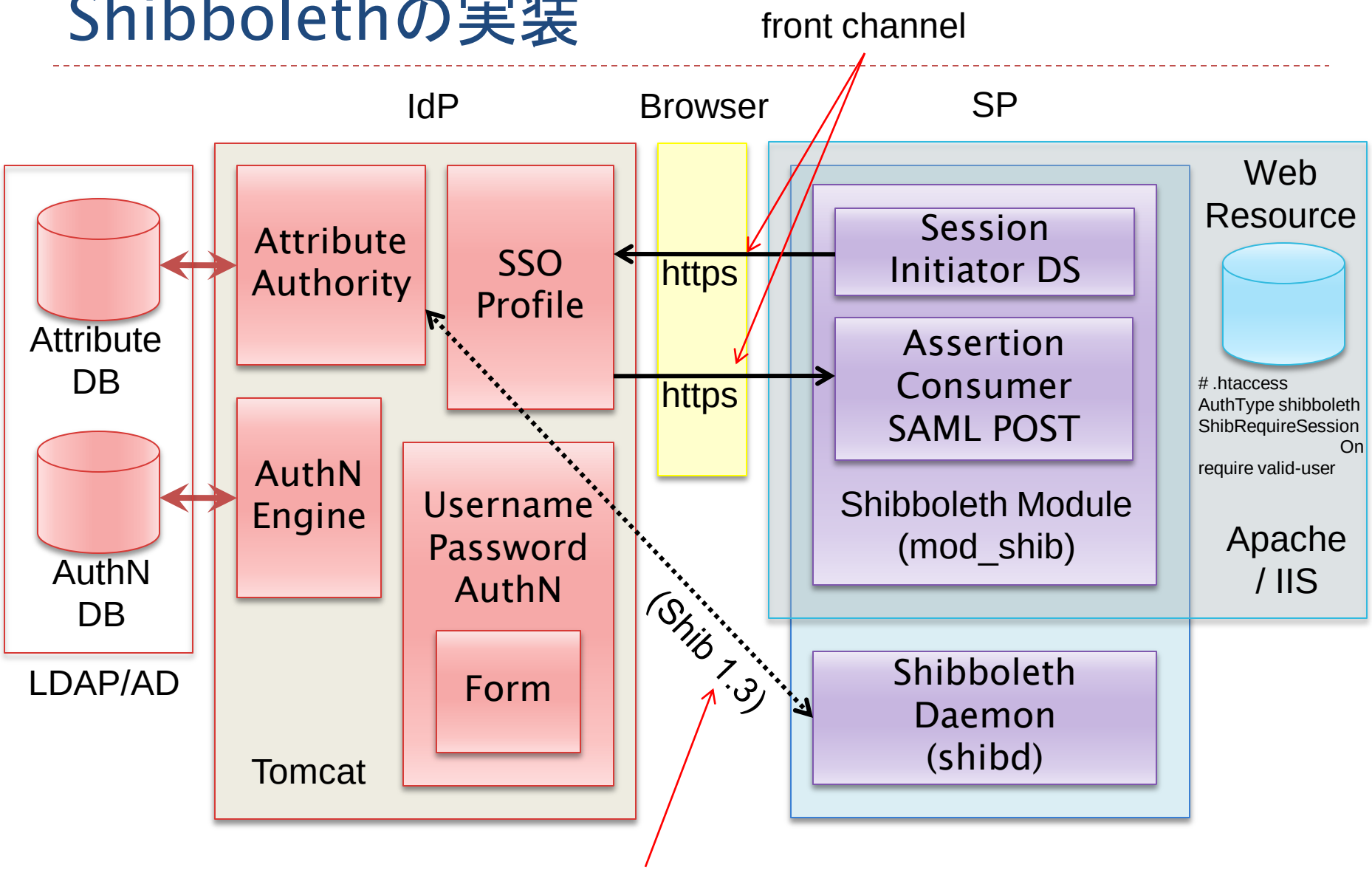
エンティティメタデータ（IdP）

・IdP1のID＝entityID
・利用する証明書
・利用可能なプロトコル
・組織情報
・・・・・

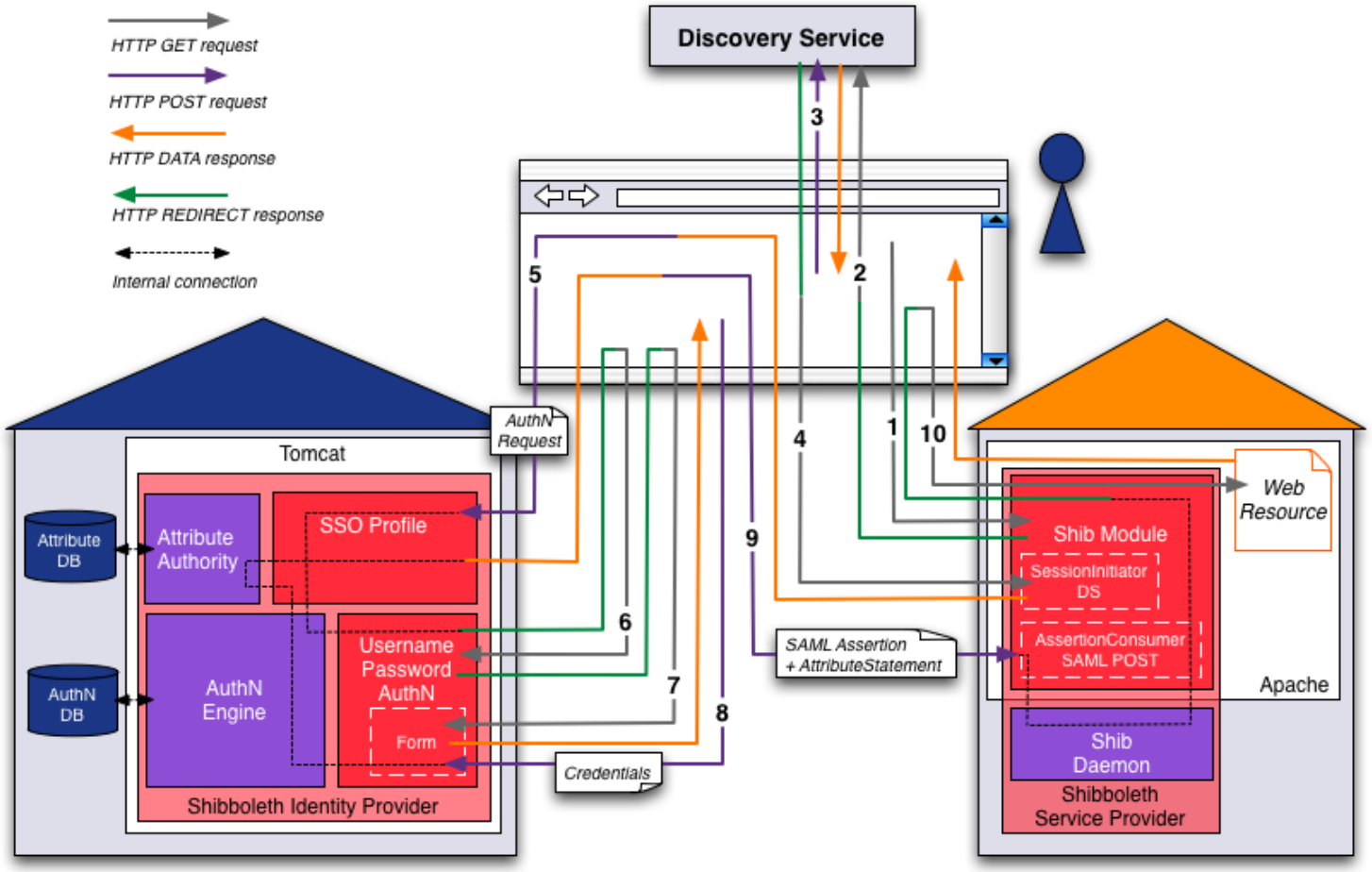
エンティティメタデータ(SP)

・SP1のID＝entityID
・利用する証明書
・利用可能なプロトコル
・組織情報
・・・・・

Shibbolethの実装

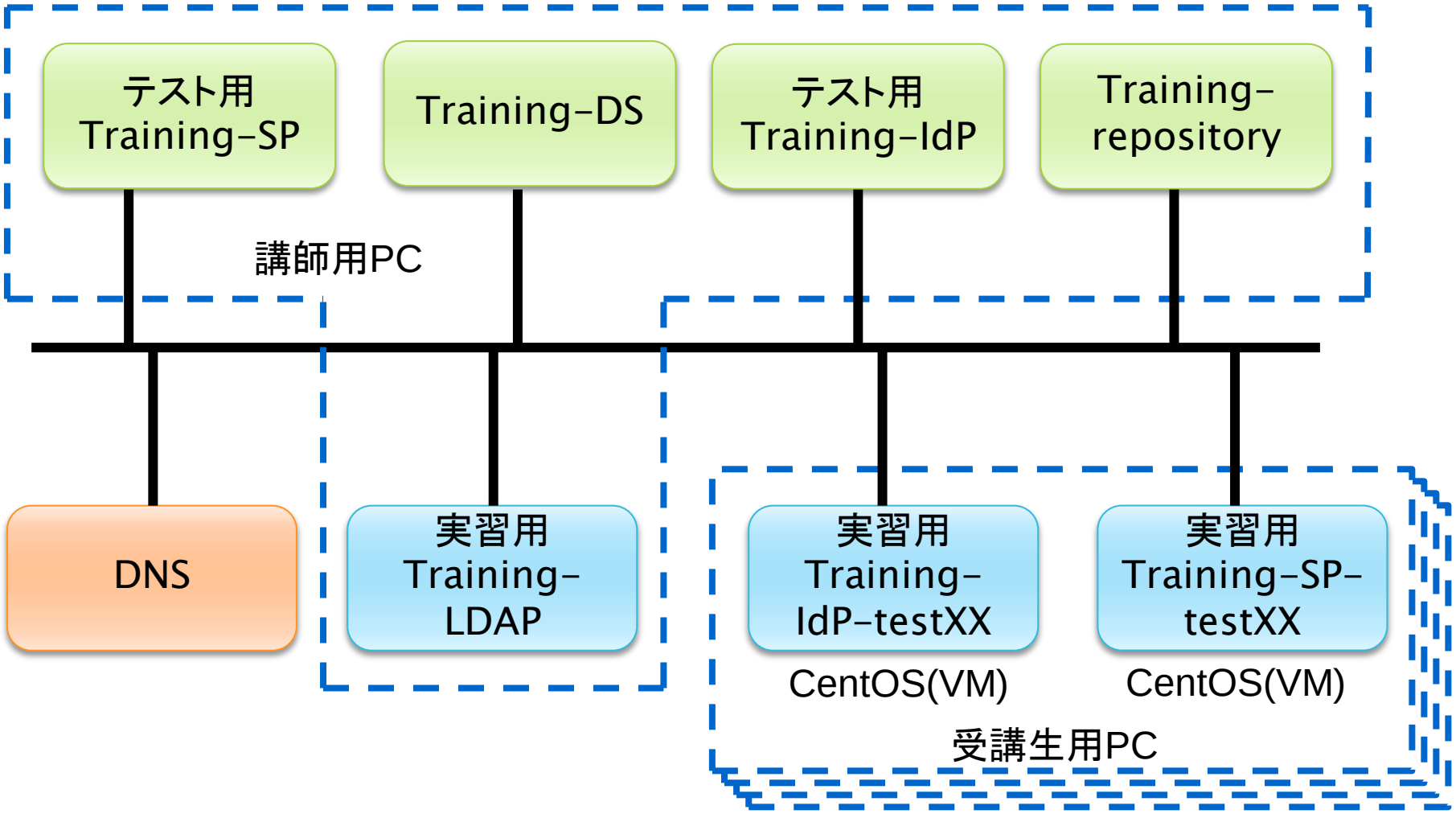


Shibboleth動作の詳細 (Shibboleth 2.x)



http://www.switch.ch/aai/demo/ より

実習環境


テスト用
Training-SP
Training-DS
テスト用
Training-IdP
Training-
repository
講師用PC
DNS
実習用
Training-
LDAP
実習用
Training-
IdP-testXX
実習用
Training-SP-
testXX
CentOS(VM)
CentOS(VM)
受講生用PC

すでに準備されているもの

- テスト用IdP、SP
  - 他機関が用意したものと想定
- Training-DS
  - フェデレーションが用意するもの
- DNSサーバ
  - IdP、SPのサーバ名を登録
- NTPサーバ
  - Shibbolethの動作には時間同期が重要
- 鍵ペア、サーバ証明書 
  - 実際には、鍵の生成と証明書発行申請が必要
- LDAPサーバ
  - 別途用意されていると想定

各種情報

1. 学術認証フェデレーションに関するWebサイト

   UPKIイニシアティブ 「学術認証フェデレーション」
   https ://upki-portal.nii.ac.jp/docs/fed

2. ポリシー、申請書

   UPKIイニシアティブ 「学術認証フェデレーション」ー「参加」
   https ://upki-portal.nii.ac.jp/docs/fed/join

3. IdP、SP構築ガイド

   UPKIイニシアティブ 「学術認証フェデレーション」ー「技術ガイド」
   https ://upki-portal.nii.ac.jp/docs/fed/technical

4. IdP構築用VMWareServerイメージ

   UPKIイニシアティブ 「学術認証フェデレーション」ー「技術ガイド」ー「IdP構築関連ファイル」
   https ://upki-portal.nii.ac.jp/docs/fed/technical/idp/files

5. テンプレート（メタデータ、IdP属性管理）

   学術認証フェデレーションのリポジトリ
   http ://upki-repo.nii.ac.jp/Template/index.html

6. 情報交換メーリングリスト（アーカイブ）

   UPKIイニシアティブ 「学術認証フェデレーション」ー「情報交換ML」
   https ://upki-portal.nii.ac.jp/docs/fed/ml

追加課題1（IdPでの属性制御）

- 属性情報を全くSPに送出されないように設定を変更し、動作確認する
- eduPersonTargetedID、eduPersonPrincipalNameのみが送出されるように設定を変更し、動作確認する
- あるSPに対してeduPersonTargetedID のみが送出されるように設定を変更し、動作確認する
- あるユーザについて、eduPersonEntitlementの値としてadminを送出するように設定し、動作確認する
- eduPersonEntitlementの値について、あるSPに対して必要な値のみ通過させるように設定し、動作確認する

追加課題2（SPでの属性制御）

- 受信した全ての属性がフィルタされるように設定を変更し、動作確認する
- IdPからeduPersonEntitlementについて複数の値を送出するようにし、SPで一方の値のみを通過させるように設定を変更し、動作確認する
- IdPでtrainingTestAttributeという新たな属性を送出し、SPでその属性を受信するように設定を変更し、動作確認する

追加課題3（SPでのアクセス制御）

- シングルサインオン（SSO）の動作を確認
- eduPersonAffiliationがstaff の場合にだけアクセスを許可するように設定し、動作確認する
- eduPersonEntitlementにtestが含まれる場合にだけアクセスを許可するように設定し、動作確認

- LazySession の設定を行い動作確認する
- ForceAuthentication を指定した場合の SSO の動作を確認する
- PassiveAuthentication を設定した場合の動作を確認する