第1回　OpenSSO&OpenAMコンソーシアム 技術セミナー

OpenAM入門

OSSTech

オープンソース・ソリューション・テクノロジ（株）
2010/11/18
小田切　耕司、岩片　靖、野村　健太郎

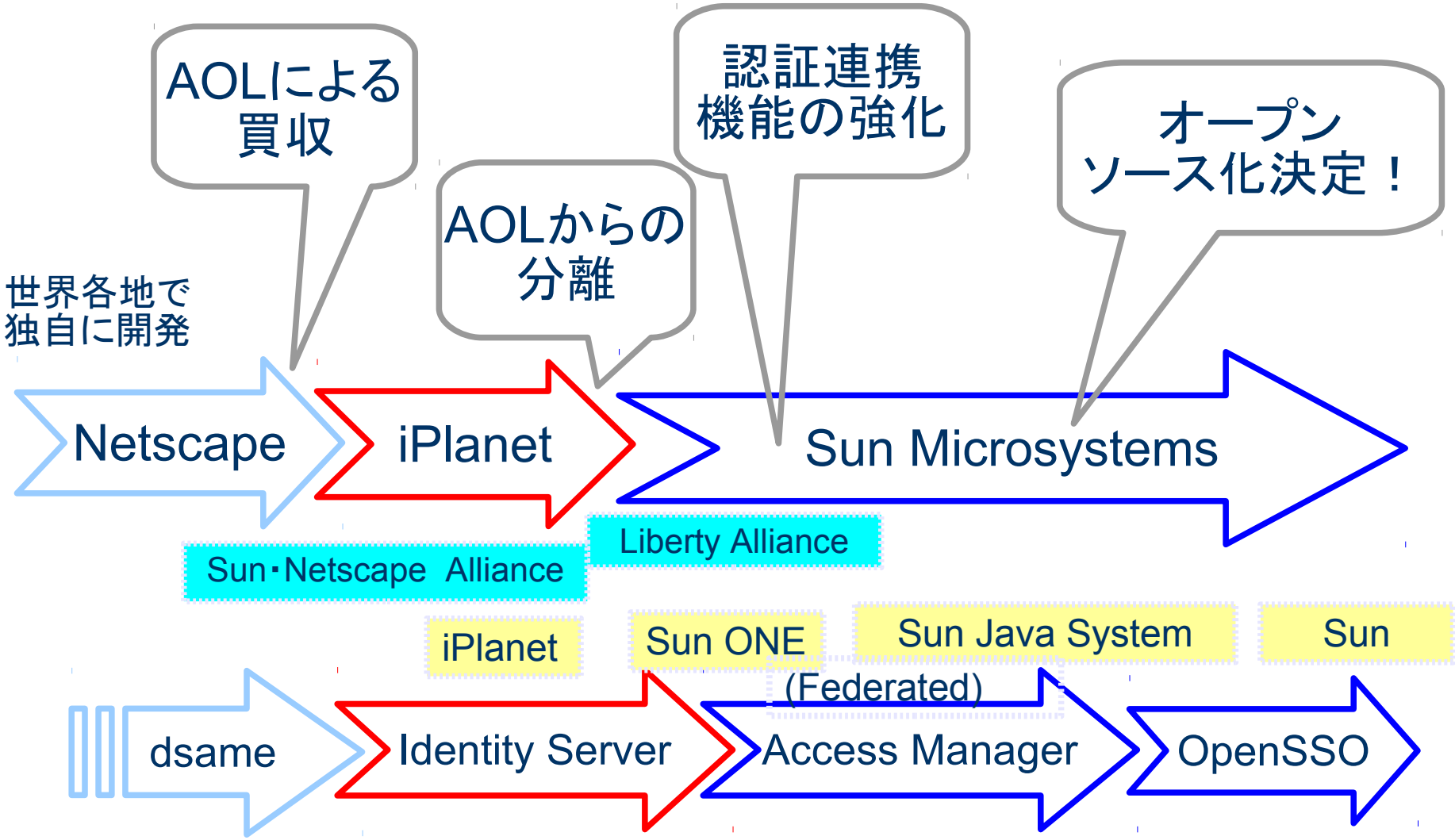
目次

- OpenSSO/OpenAMの歴史
- OpenAMになって変わったこと
- 基本的な機能
- デモ

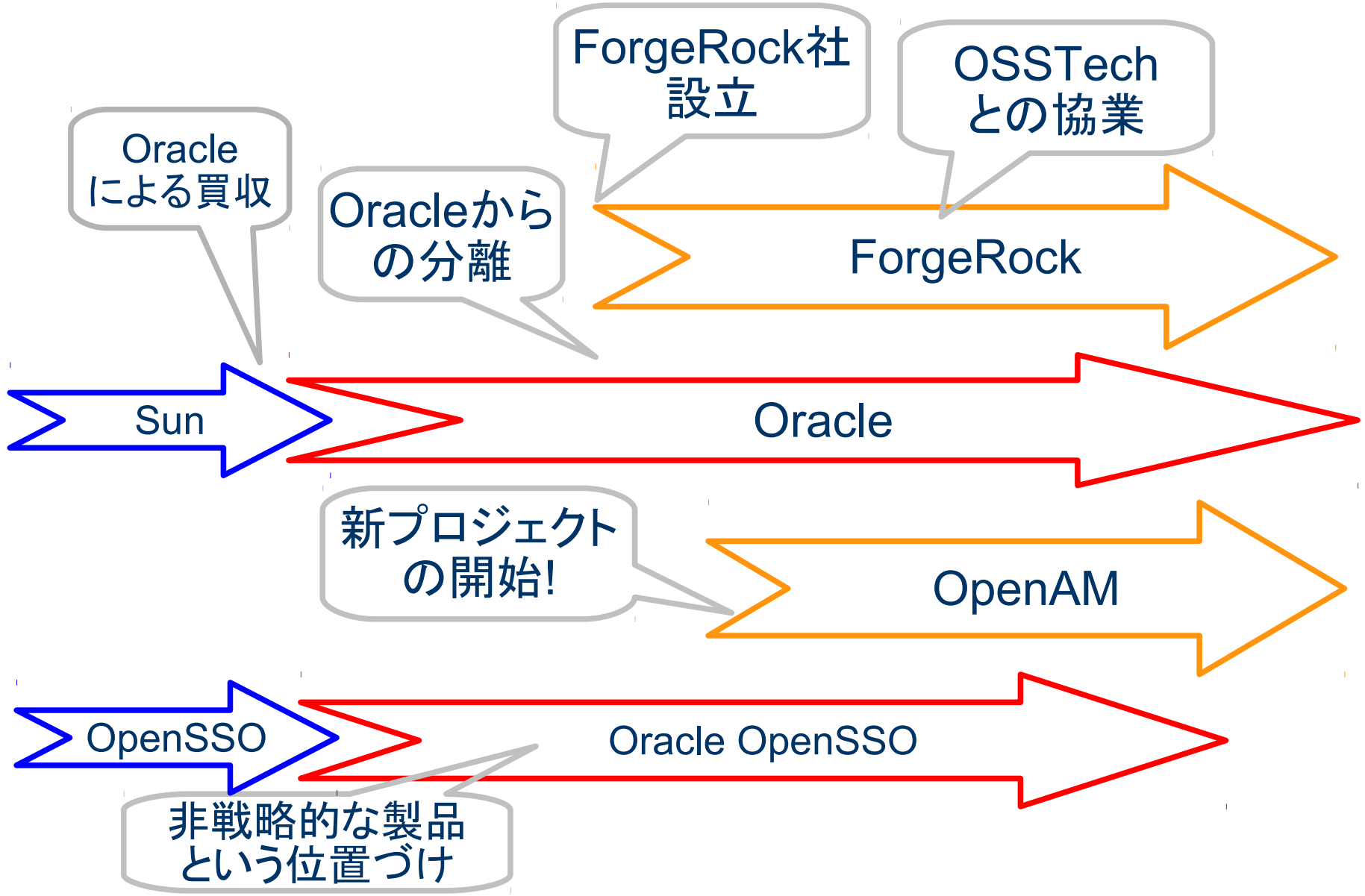
OpenSSO/OpenAM

の歴史

OpenSSO/OpenAMの歴史 (Part 1)



OpenSSO/OpenAMの歴史 (Part 2)



OpenAMになって

変わったこと

正常な進化形

- 作っている人が同じ
  - OpenSSOを担当していたエンジニアが中心になりForgerockを設立
- ベースにするソースコードが同じ
  - 最新のVer. 9.5では多量のバグフィックスを適用
- ユーザも同じ場合がほとんど
  - 既存ユーザからの移行促進（米国、ヨーロッパ）
  - 日本では多くが新規ユーザ

提供形態の変化（予想も含む）

- 他のOSSとの整合性強化
  - リポジトリとしてのOpenLDAP, OpenDS, MySQL
  - 動作プラットフォームとしての CentOS, Tomcat
- ベンダ独自のパッケージング
  - 弊社ではOpenLDAP用拡張スキーマを提供
- 得意分野と組合わせた統合ソリューション
  - 生体認証等の認証方式との組み合わせ
  - プロビジョニングシステムとの組み合わせ
  - 人事管理システムとの組み合わせ
  - 弊社ではUnicorn IDマネージャと組合わせてGoogle Appsとのシングルサインオン　ソリューションを提供

最新版における機能強化ポイント

- クラウド対応
  - Google Apps, SalesforceとのSAML連携を強化
  - GUIによる操作で連携設定が可能
- OpenDSの最新版を内蔵
  - Version 2.3 安定版
  - 標準ツールの添付
- 多量のバグフィックス
  - OpenSSO Expressで開発してきたユーザへの対応
  - OpenAMへの移行促進

OpenAMの基本機能（その１）

認証方式と多要素認証

多要素認証

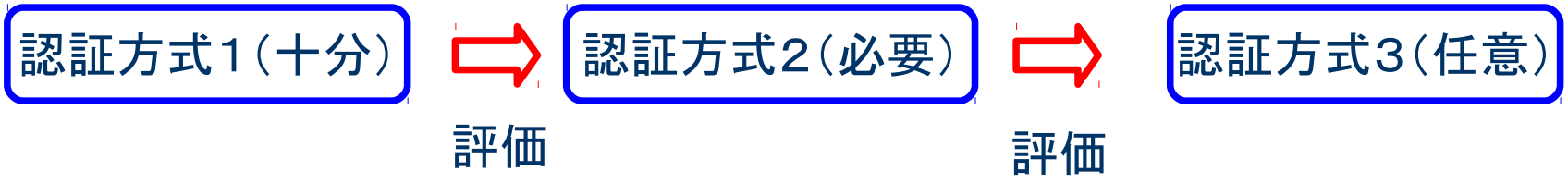
OSSTech

複数の認証方式を組合わせて認証を行うことにより
個々の認証方式の欠点を補完

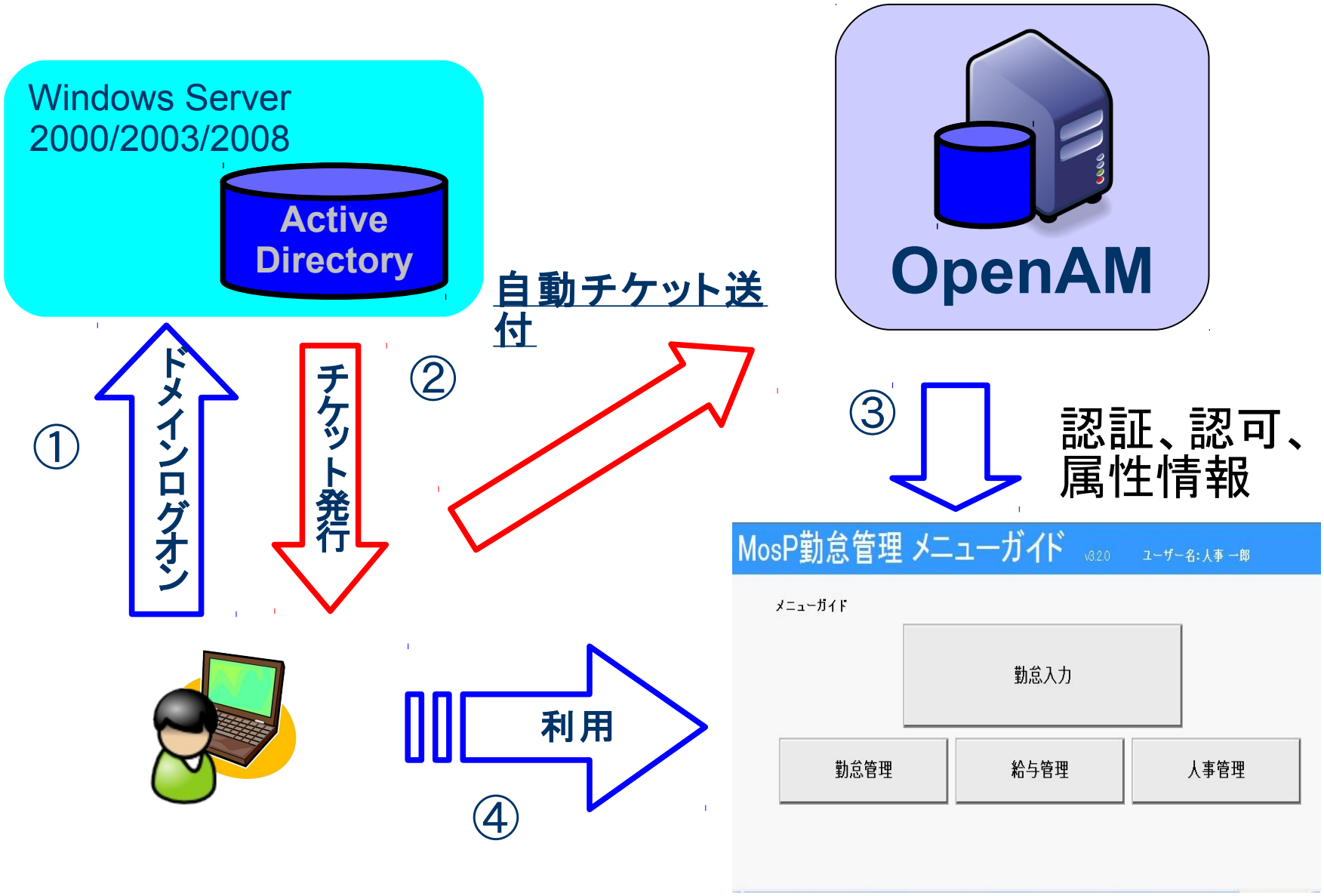
- 厳密なユーザ認証
  - 異なるタイプの認証方式を組合わせることが重要
- 使い勝手の向上
  - いつも同じ認証方式が使えるとは限らない
  - 状況により要求される認証の精度が異なる
- 認証方式間での連携
  - 組合わせて使うことを前提にしている認証方式もある

認証方式を組合わせる方法を指定する

- 認証方式にはそれぞれ適用条件を指定する
  - 十分：成功したらそこで終了
  - 必要：成功しても失敗しても次に継続
  - 必須：失敗したらそこで終了
  - 任意：認証結果には関係しない付随的な処理
- 認証成功時には認証方式に応じて認証レベルが設定される



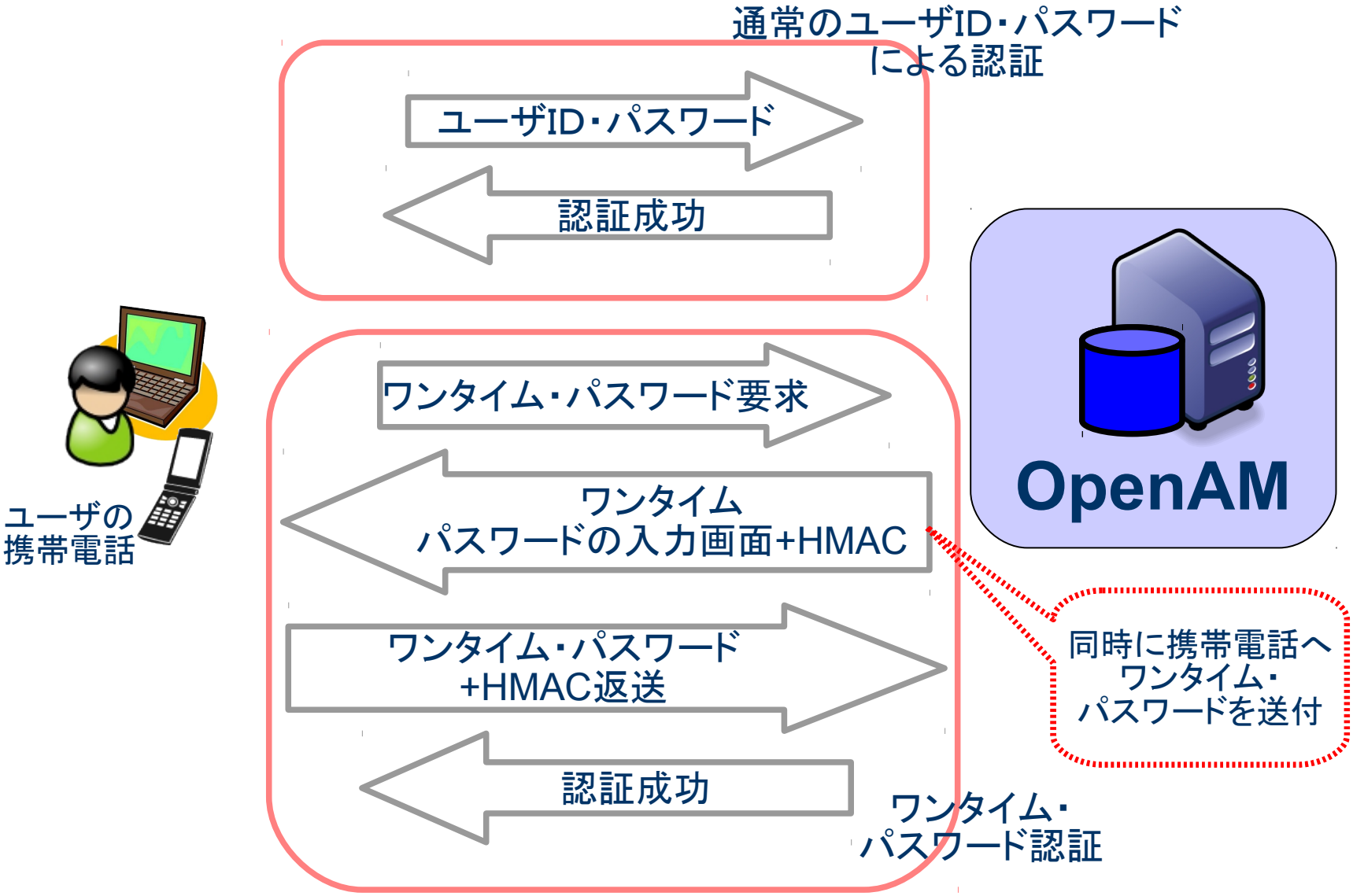
認証連鎖の例1. Windows Desktop SSO



Windowsドメインログオンするだけで Webアプリケーションにも SSOが可能になる便利な方式

- いつも、全てのユーザがドメインログオン可能であるとは限らない
  - リモート・アクセスの場合
  - 非常勤社員の場合
- 通常のユーザID・パスワードによる認証と組み合わせて以下のように認証連鎖構成する
  - Windows Desktop SSO: 十分
  - ユーザID・パスワードによる認証：必須

認証連鎖の例2. ワンタイムパスワード



認証連鎖の例2．ワンタイムパスワード

- 所持物認証と知識認証の組合わせによる厳密なユーザ認証が可能
- 携帯電話を使うことによる利点
  - 導入コストの低減
  - 所持品の軽減
- フィッシングへの対応
  - HMACを利用
  - 両方のパスワードが盗まれた場合は問題

認証連鎖の応用例

- Windows Desktop SSOによる認証は便利なのでぜひ使いたいが全てのユーザがドメインログオン可能とは限らない
- ワンタイム・パスワードは厳密な認証が出来る点は良いが、いつも携帯電話を開いてパスワードを確認するのは面倒だ

- 2つを組合わせることにより便利かつ厳密な認証を行うことが可能
  - Windows Desktop SSO: 十分
  - ユーザID・パスワードによる認証： 必須
  - ワンタイム・パスワードによる認証： 必須

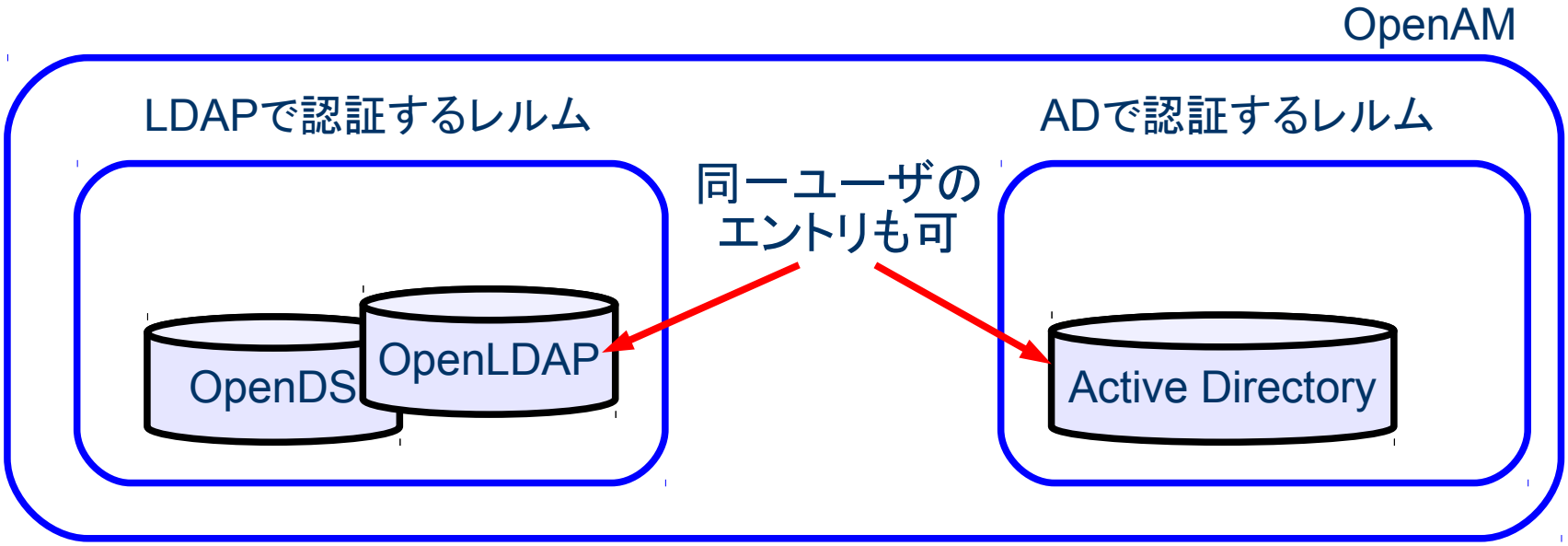
OpenAMの基本機能（その2）

レルムと委任による

ユーザ管理

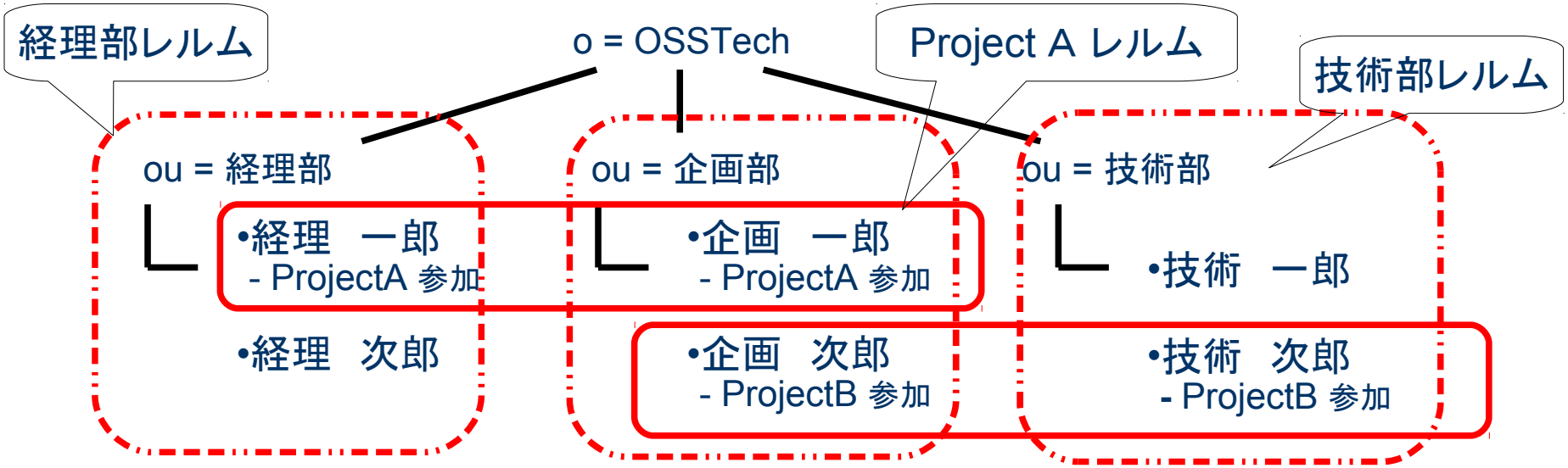
レルムの基本

- レルム：設定を管理するための単位
  - ユーザリポジトリ（OpenLDAP, OpenDS, AD, RDB...）
  - アクセス制御ポリシー
  - 認証方式
- ユーザは複数のレルムの所属することが可能
- ひとつのレルムに複数のリポジトリを設定可能
- レルム毎に管理者を置き管理を委任することが可能



レルム使用の具体例

- 社員は組織別に分けられてLDAPサーバに保存されている
- 社内Projectでは組織を横断してメンバーが参加する
- 管理は組織単位で行う他にProject単位でも行いたい
  - 組織単位のレルム：ベースDNを指定
  - プロジェクト単位のレルム：ユーザ属性をフィルタに指定
  - 管理を各レルムの管理者に委任



OpenAMの基本機能（その３）

多様なシングルサインオン方式

様々なシングルサインオン方式

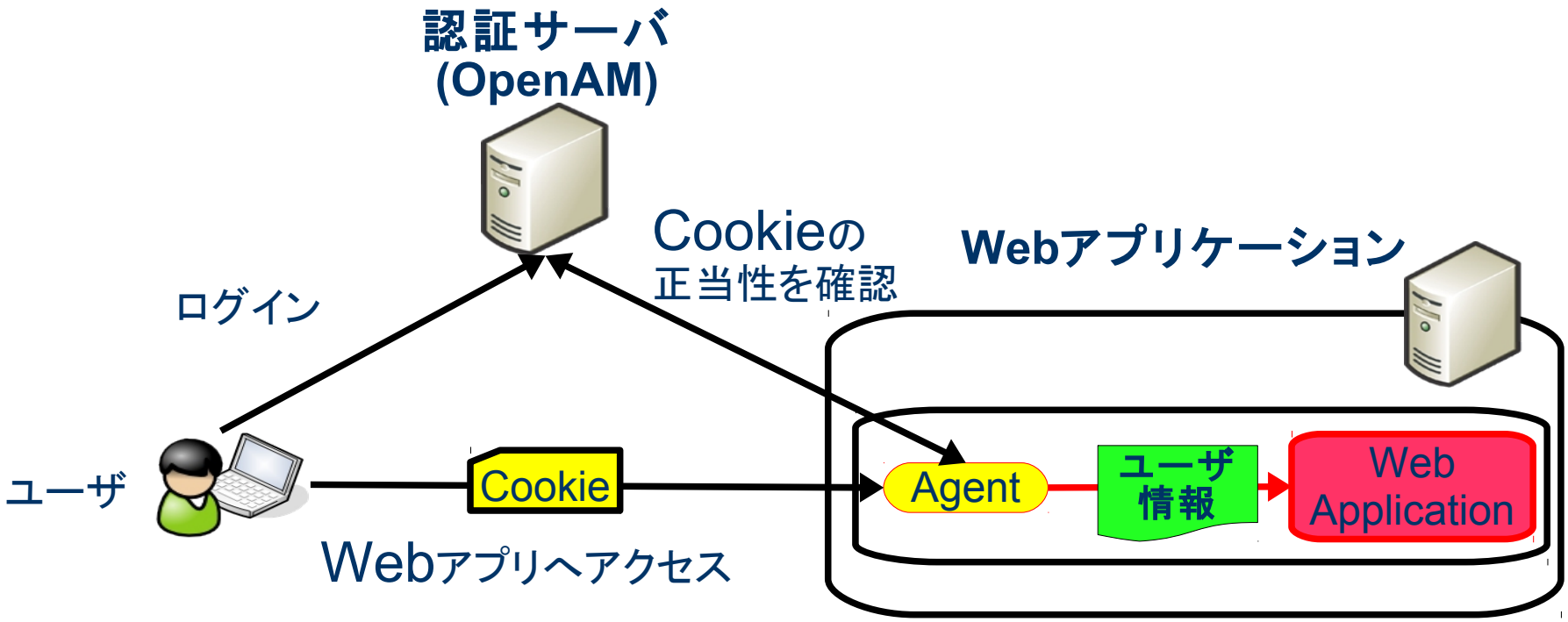
OSSTech

- エージェント方式
  - 保護対象のアプリが動作するサーバ上にアクセス制御用のモジュールを配置する方式
  - APIレベルでの細かな連携が可能
  - 保護対象のアプリやサーバのバージョンや設定変更に影響されやすい
- リバースプロキシ方式
  - リバースプロキシを使ってアクセス制御を行う方式
  - データの受け渡し方法がHTTPヘッダに限定
  - 保護対象のバージョンや設定変更の影響が少ない
  - 性能のボトルネックになる可能性も

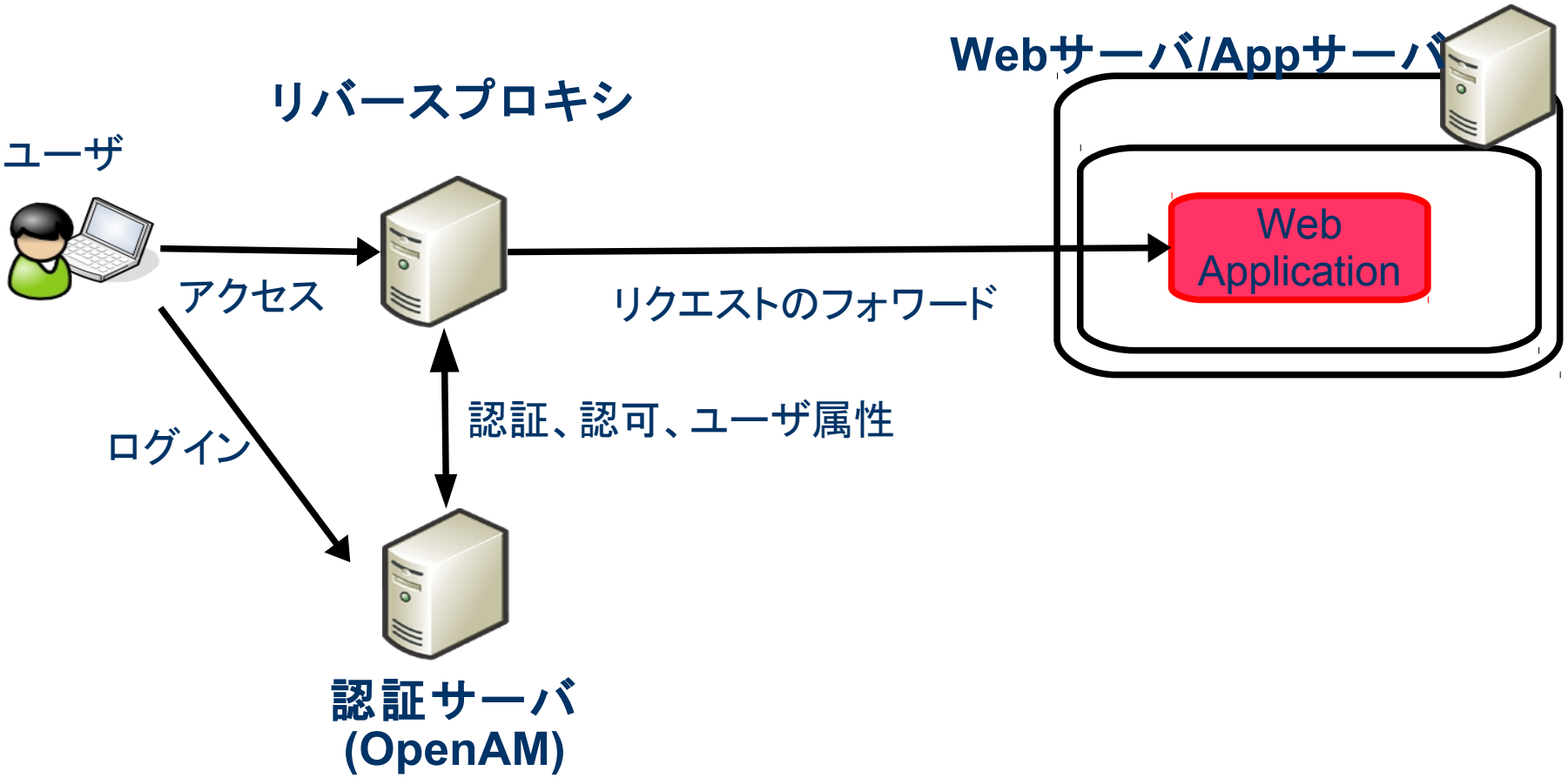
SAMLを使ったシングルサインオン

- SAML
  - Secure Assertion Markup Lauguage
  - 認証、認可、ユーザ属性情報などをXMLで送受信するためのフレームワーク
  - 標準化団体OASISにより策定
  - 通常はサイト間連携で使用

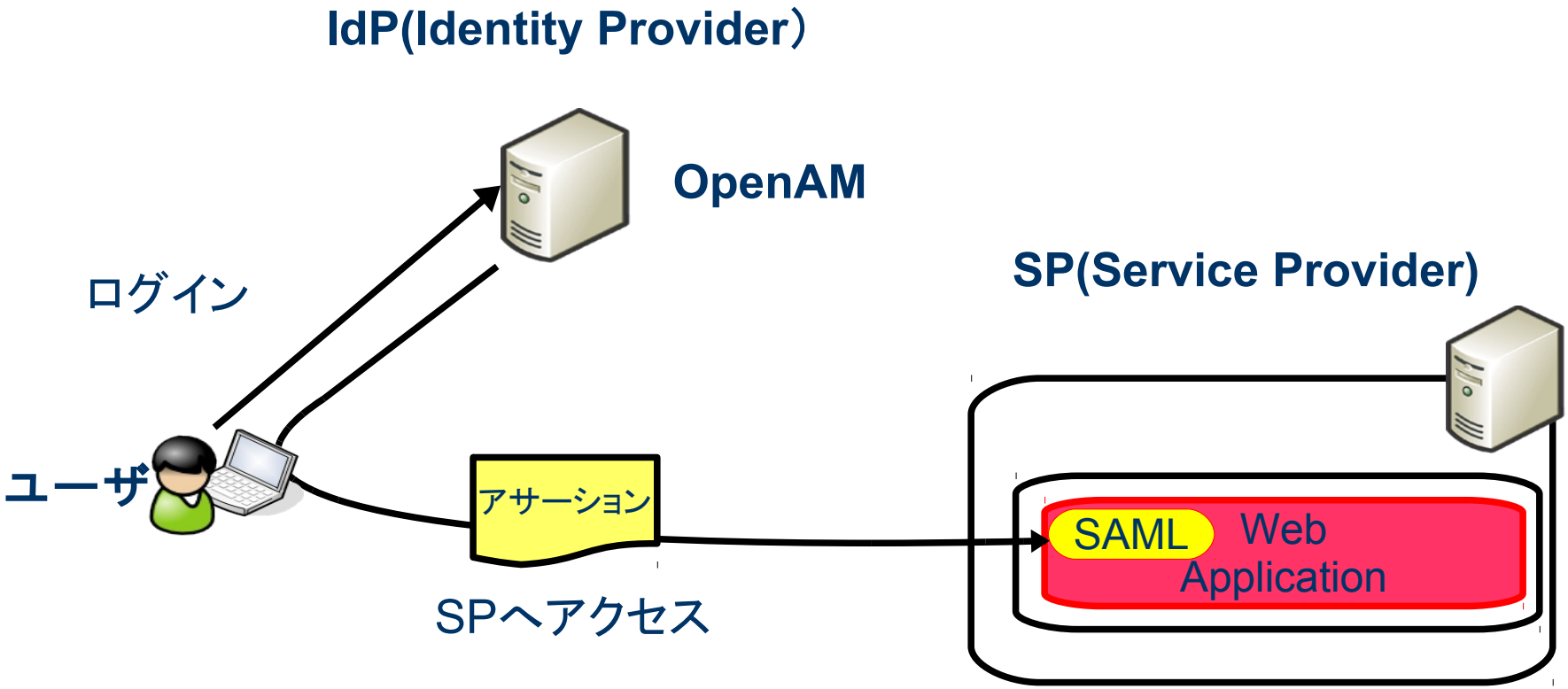
エージェント方式



リバースプロキシ方式



SAMLによるシングルサインオン



OpenAMの基本機能（その４）

アクセス制御ポリシー

アクセス制御ポリシー

誰が ＋ 何に対して ＋ どのような操作ができるか

- 所属組織、グループ
- ロール
- 認証方式（認証レベル）
- 個人
- アクセス方法

URLを正規表現で指定

POST & GET

…を定めたルールの集まり

本日のまとめ

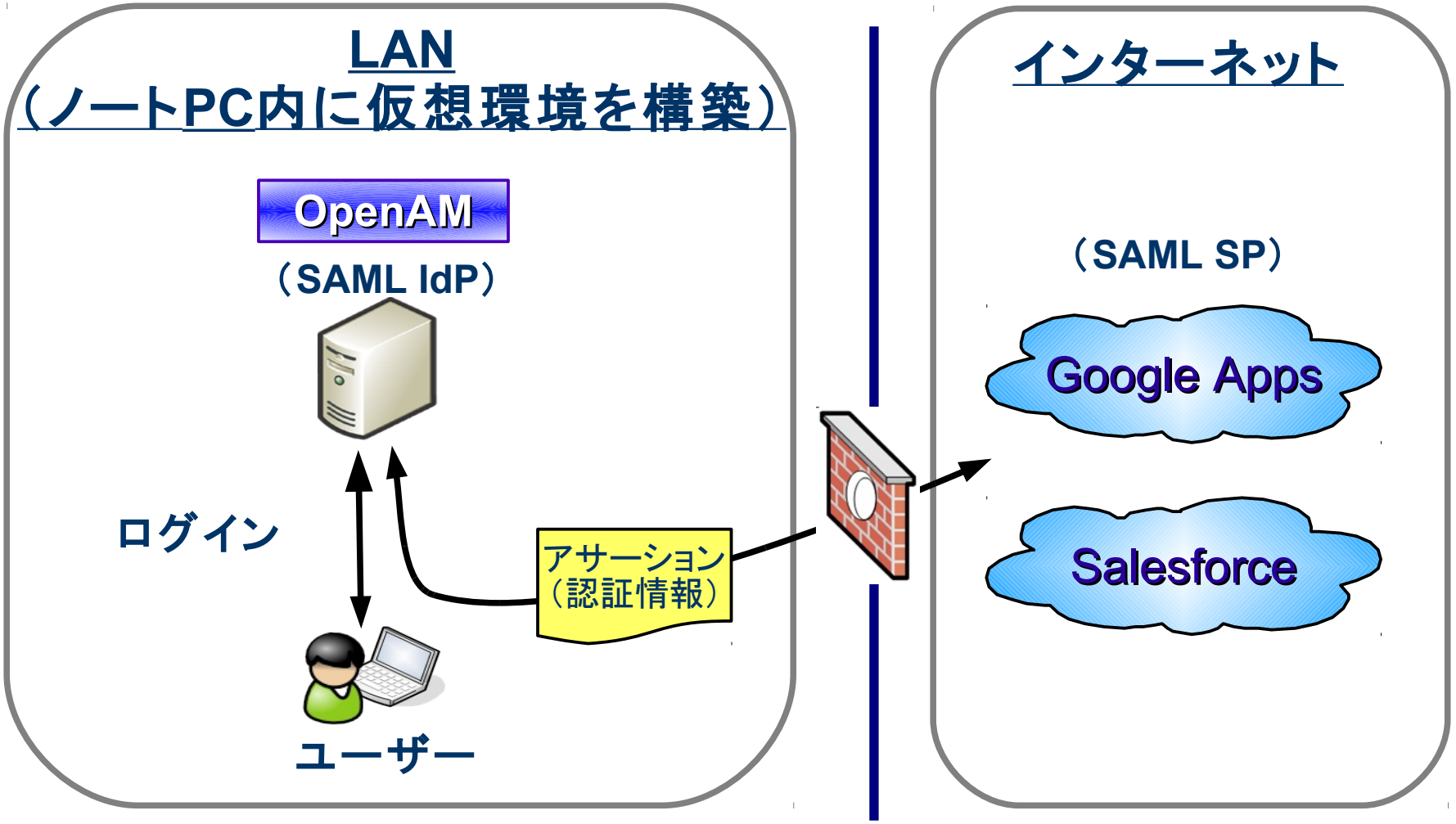
- OpenAMは長い期間をかけて着実に進化してきました
- OpenAMの最新版はクラウド対応と安定稼動を目標としています
- コミュニティ・ベースの開発になることによって、様々な提供形態が出現することが予想されます
- OpenAMはユーザ管理、シングルサインオン、アクセス制御に関して様々なオプションを提供しています
- OpenAMはGoogle Appsとの連携に使いたいというユーザから、本格的なクラウドサービスを構築したいというユーザまで幅広く対応可能です

OpenAM デモ

OpenAM デモ内容

- （1）SAMLによるシングルサインオン
  - Google Apps と Salesforce にシングルサインオンでアクセスする
- （2）認証連鎖
  - ワンタイムパスワード認証を追加して二要素認証を行なう

OpenAM デモ環境



※この図は SAML におけるHTTP Redirect Binding/HTTP POST Binding の場合の例です

参考情報

- SAML 仕様
  - http://www.oasis-open.org/specs/index.php#saml
- 弊社の OpenSSO 社内勉強会の資料
  - http://www.osstech.co.jp/techinfo/opensso