OpenAM最新情報

野村総合研究所
生産革新ソリューション開発二部
OSS推進グループ
和田　広之

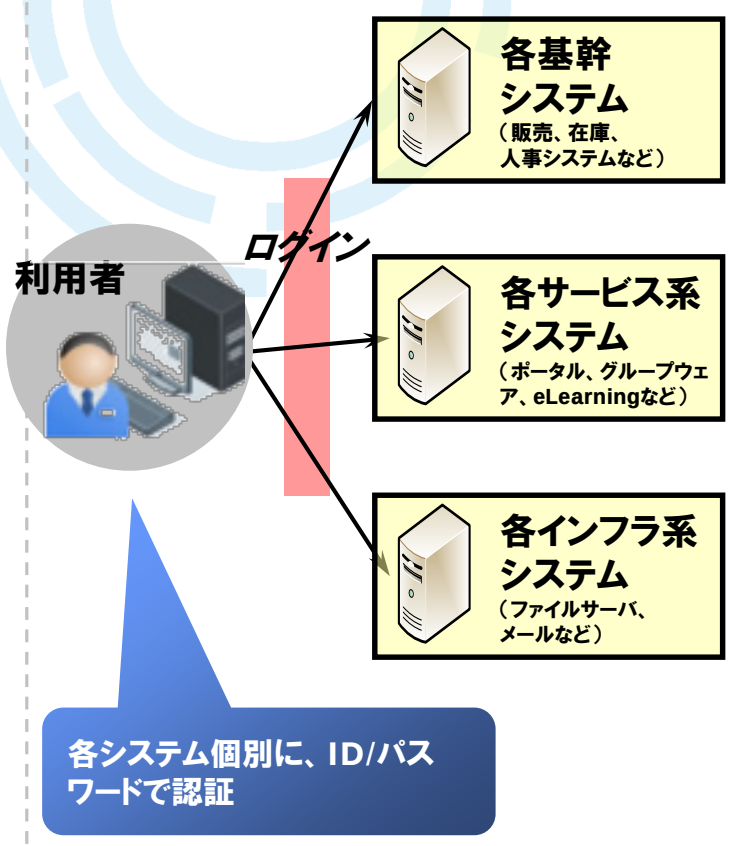
自己紹介

- **所属部署**
  - **生産革新ソリューション開発二部** OSS**推進グループ**
  - OSS**を使ったシステム構築から運用までワンストップでサポート**
  - **対象**OSS**は**50**種類以上**
- **私の担当**
  - **各種**OSS**の技術的サポート**
  - OpenAM、OpenIDM**の導入支援**
  - OpenAM、OpenIDM**の機能拡張、バグ修正も実施**
    - OpenStandia**チームから挙げたバグ・改善要望チケット数**: **270件以上**

はじめに

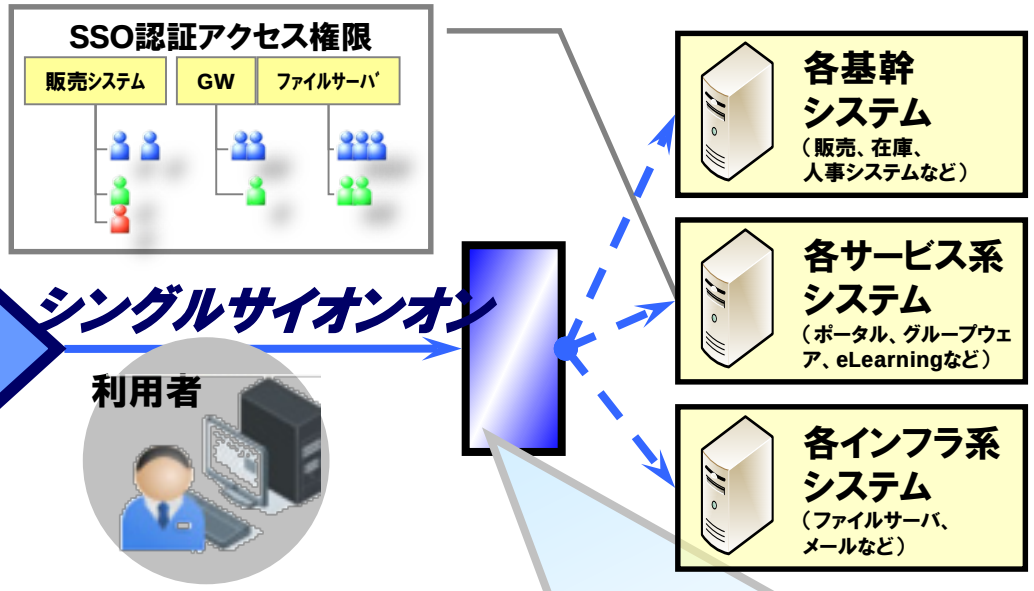
シングルサインオン(SSO)について

As-Is(現状運用)



各システム個別に、ID/パスワードで認証

To-Be(SSO導入後)



- アクセス権限付与に基づく厳密なアクセス制御
- セキュリティポリシに基づいた厳密な認証インタフェース
- システムへのアクセスの認証の統合化による利便性向上
- アクセスログの一括取得
- 多様化する認証方式への対応
  - ✓ 対象システム ： Web系システム、C/S系システム、OS…
  - ✓ 認証連携インタフェース ： ID/PWD、生体認証、PKIなど

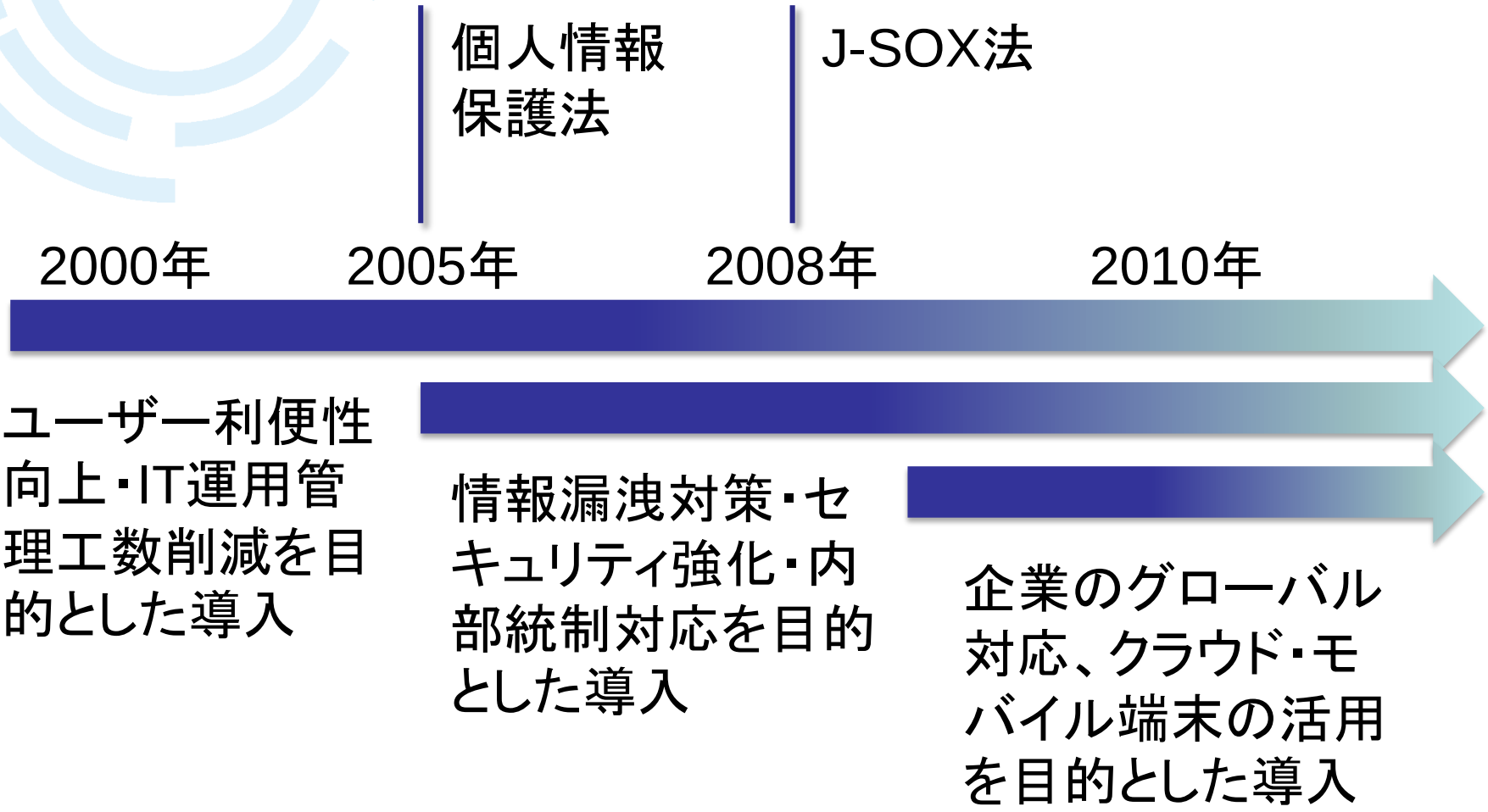
SSOを実現するソフトウェア=OpenAM

オープンソースまるごと OpenStandia™ Open Source Technology

NRI 未来創発 Dream up the future.

シングルサインオン・ID管理が求められる時代の環境変化

時代とともにSSO/IDMの導入目的も変化



シングルサインオン・ID管理が求められる時代の環境変化

近年さまざまな環境変化により、企業内システム利用の在り方、及びそれに基づくID管理の在り方、認証の仕組みが見直されてきている

社内環境の変化

- システム、ユーザアカウント、権限の複雑化
- 内部統制・コンプライアンス・個人情報保護の強化
- 採用形態の複雑化（グローバル人材、アウトソース、出向等）

IT環境の変化

- クラウド時代の到来による「所有」から「利用」への流れ
- 社内システムのSaaS利用
- モバイル端末、スマートフォン、タブレットの利用拡大
- 今後はIoTの活用

事業環境の変化

- グローバル化
- M＆A、企業合併によるグループ企業の統廃合
- 新規サービス事業の開始

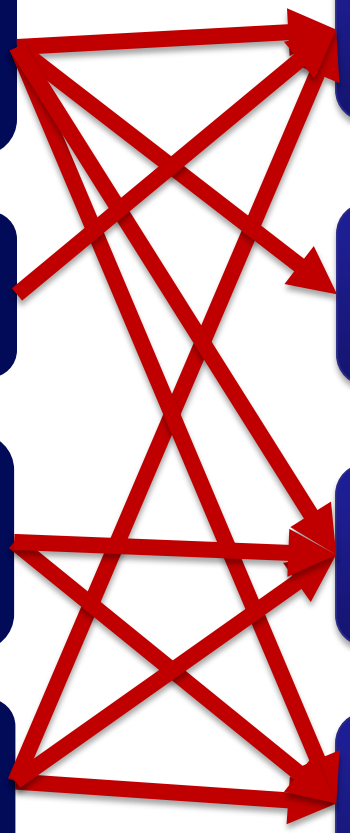
今後、認証基盤に求められる要件

環境変化

- モバイル端末、タブレットなどの利用拡大
- グローバル対応
- SaaS・クラウドの活用
- IoTの活用

求められる要件

- スケーラビリティ
- 高度認証
- 認可処理
- 標準プロトコルによる相互接続性



OpenAM

- 既に国内でも多数の導入実績があり、基本機能は枯れている
- 環境変化に合わせて継続的にアップデート
- IoT時代を見据えた機能の追加・最新の標準仕様にも対応
  - 開発元のForgeRock社は、標準仕様の策定にも深く関わっている

最新版で追加された新機能について紹介

OpenAMの最新情報

OpenAMの最新情報

- 13.0.0が1月26日にリリース !!
- リリースノート
  - https://backstage.forgerock.com/#!/docs/openam/13/release-notes
  - 非公式ですが、弊社OpenStandiaチームのメンバーによる日本語訳があります。
    - ✓https://t246osslab.wordpress.com/2016/01/30/openam-13-0-0がリリースされました/
- 過去バージョンで非推奨となっていたいくつかの機能が削除されているので注意
  - https://backstage.forgerock.com/#!/docs/openam/13/release-notes#removed-functionality
  - /identity/attributes などのREST APIが削除に

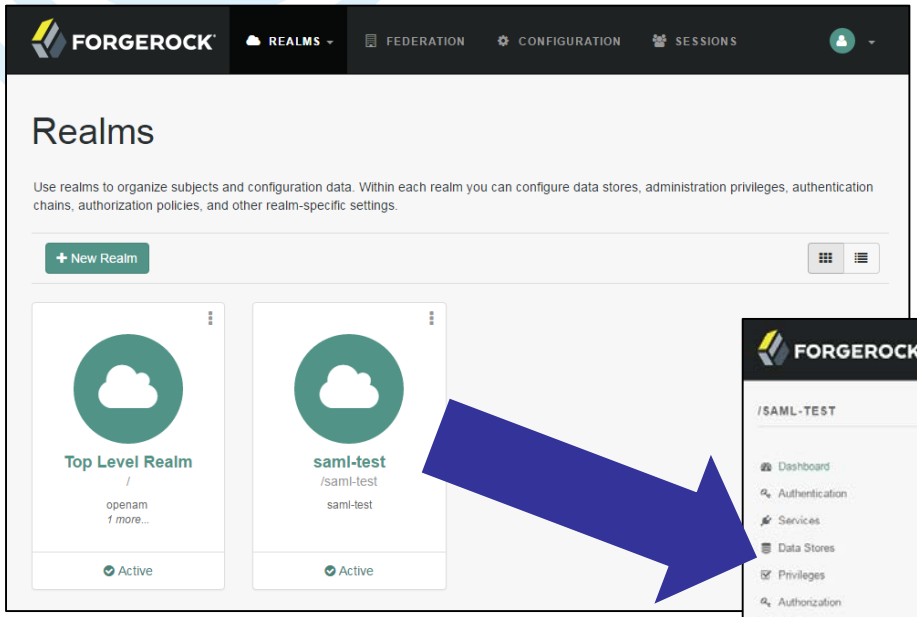
OpenAMの最新情報

本日紹介するアップデート内容

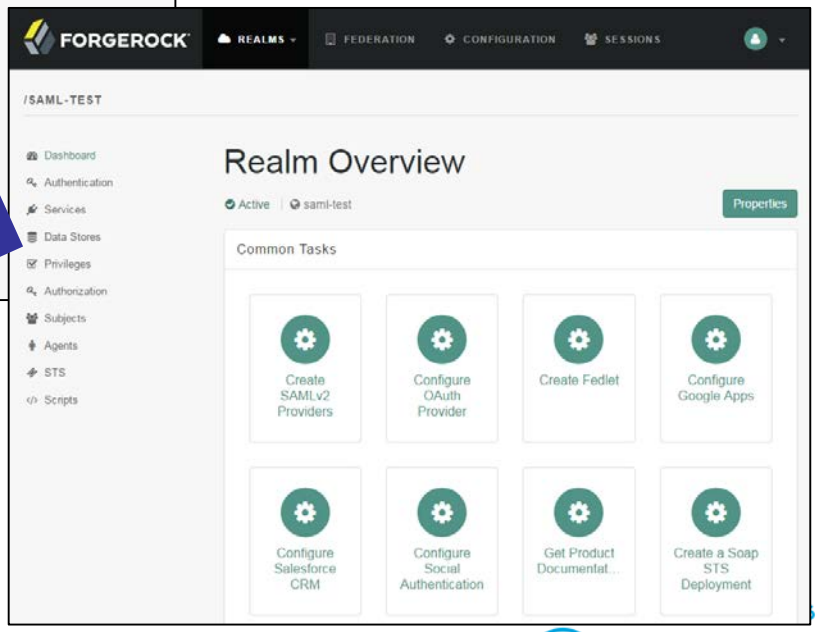
1. UI
   - ✓管理コンソールの刷新
2. スケーラビリティ
   - ✓Stateless Session 機能の追加
3. 認証
   - ✓モバイルデバイスを使った2段階認証機能の強化
4. 認可
   - ✓標準プロトコル UMA(User-Managed Access) に対応
5. 開発者向け
   - ✓スクリプティングサービス

管理コンソールの刷新

- デザインが今風のおしゃれな感じに
- レルムを中心としたUIに



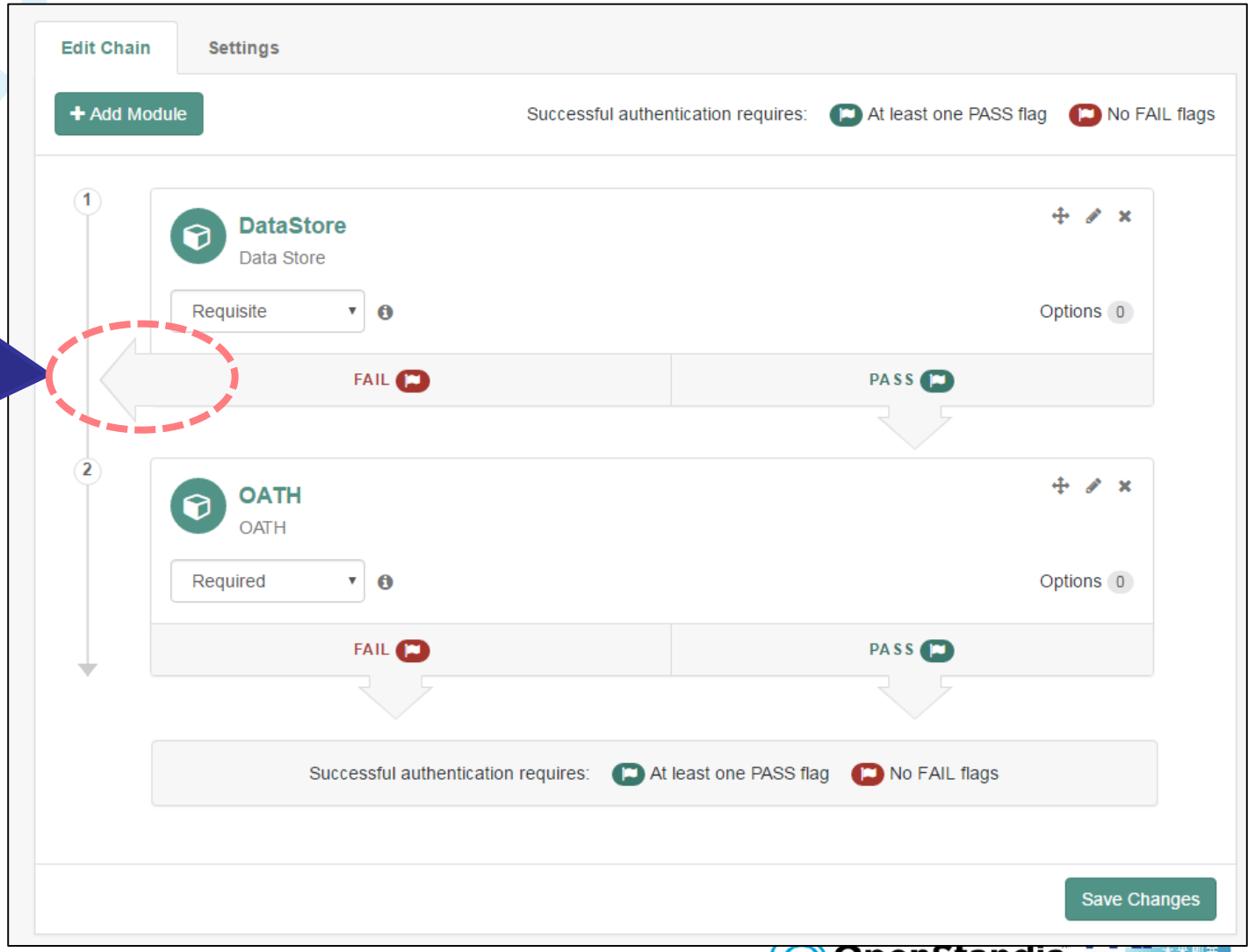
最初にレルムを選択し、そのレルム内の設定を行うUIに



管理コンソールの刷新

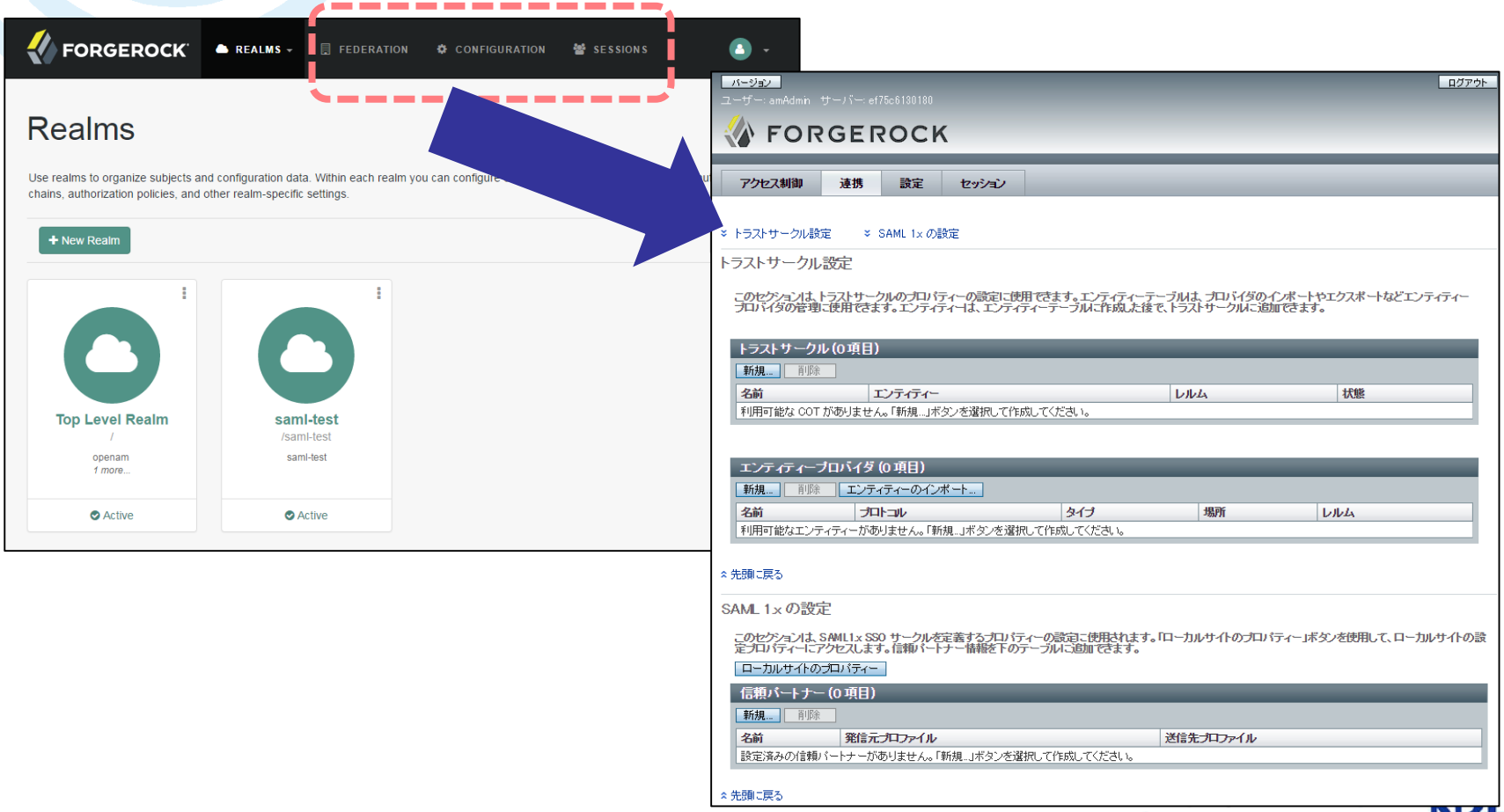
認証連鎖の設定はグラフィカルに

失敗すると即認証エラーとなること視覚的に表示



管理コンソールの刷新

- ただし、まだレガシーUIの箇所も・・・
  - 今後のさらなる改善に期待



Stateless Session 機能の追加

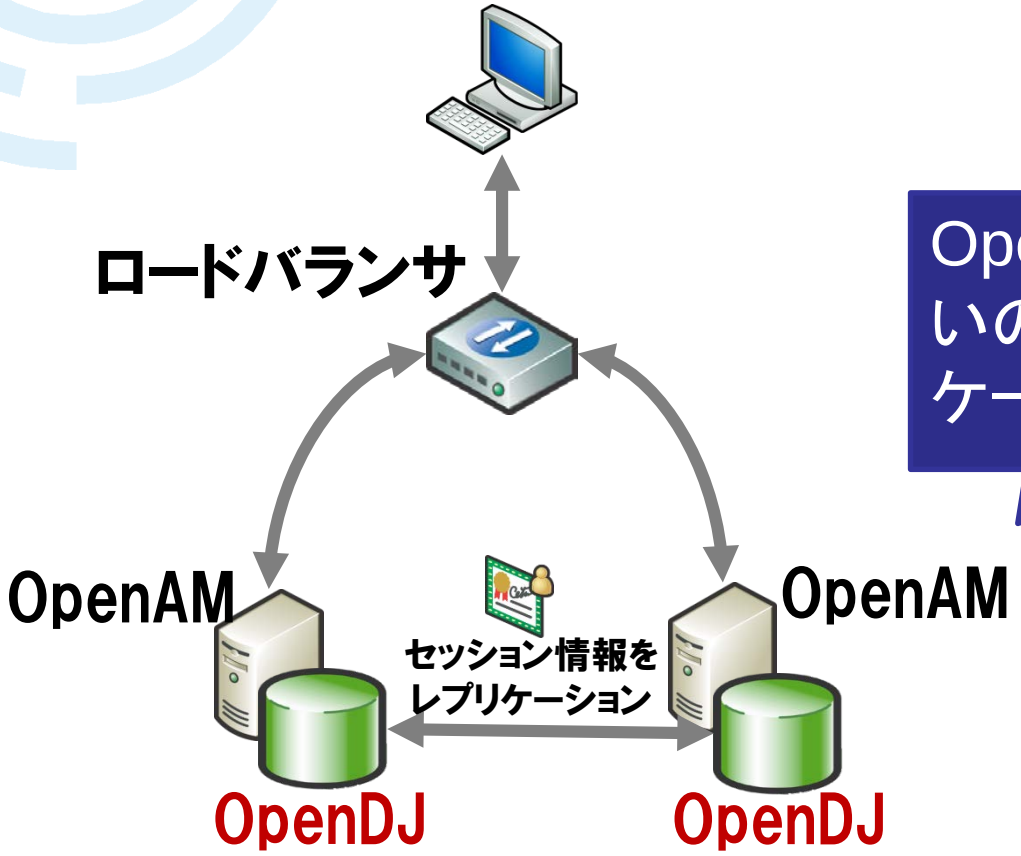
- **セッションタイプにStatelessが選択可能に**
  - **従来は、Stateful モードのみ**
- **Cookieの中にセッション情報を保存し、OpenAMサーバのメモリ上には持たない**
  - **スケールアウトが必要な大規模構成で有効**
  - **セッション情報はJWT(JSON Web Token)で格納されており、デジタル署名(改ざん防止)、暗号化も可能**

CookieにJWT形式でセッション情報を保存

出所: http://openam.forgerock.org/doc/bootstrap/admin-guide/index.html#session-state-cookies

従来のセッション管理方式

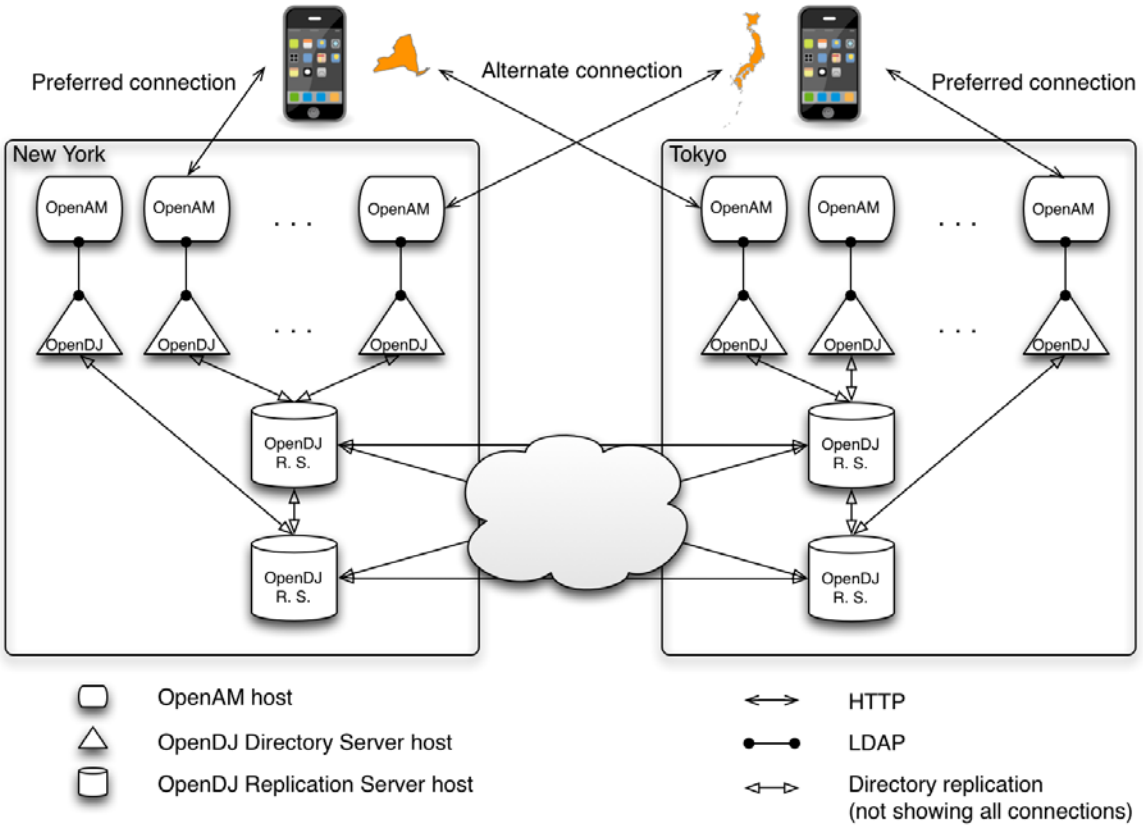
- OpenAMが内蔵している組み込みLDAP(OpenDJ)にセッション情報を保存してレプリケーション



OpenAMが2～4台くらいの規模であればスケールする

従来のセッション管理方式

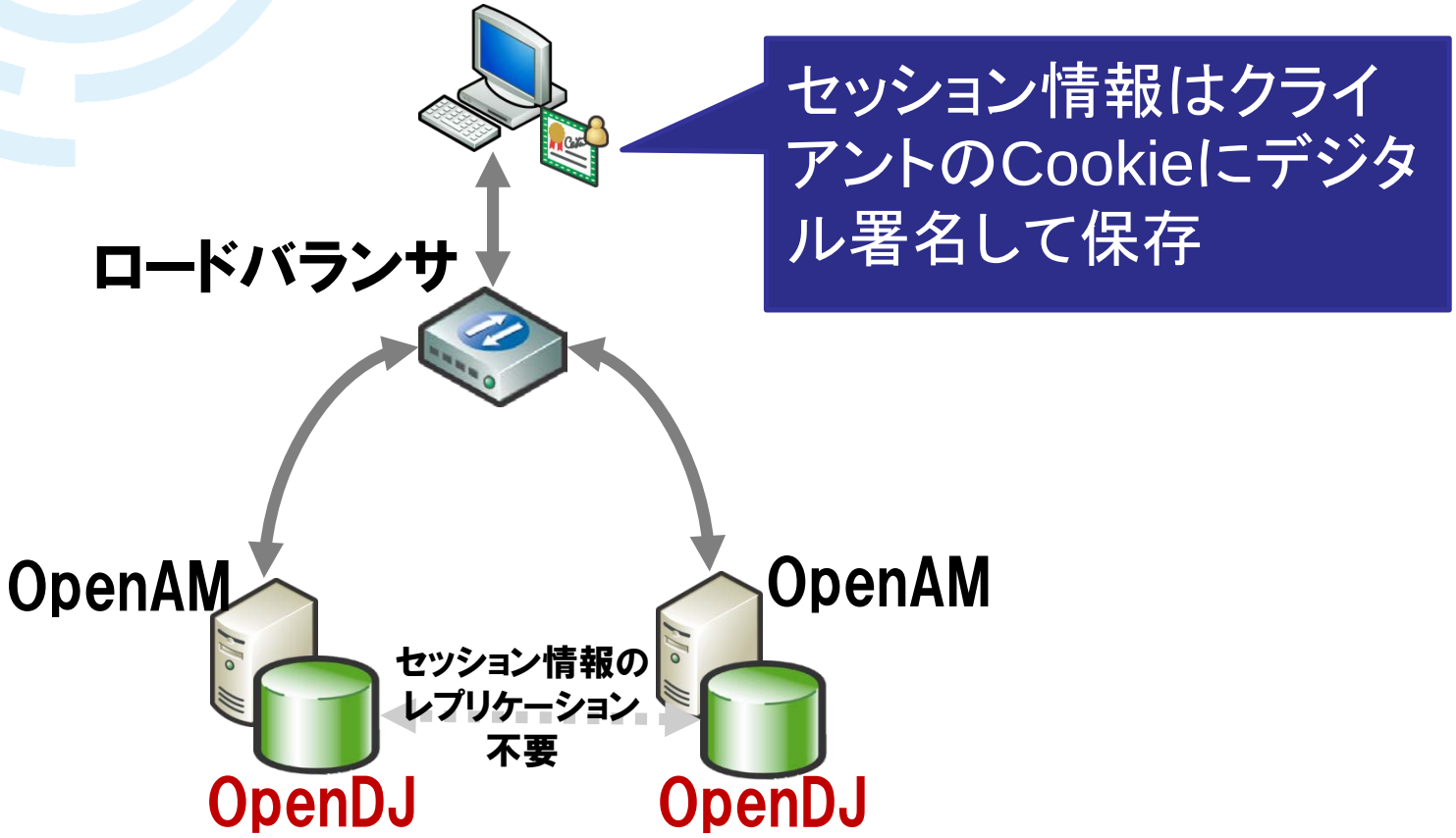
- 大規模になると…
  - レプリケーション処理の増大によりスケールしなくなる
  - レプリケーション接続数を減らすために、レプリケーション用のOpenDJを別で立てるといった対応が必要



出所: https://backstage.forgerock.com/#!/docs/openam/12.0.0/install-guide/chap-session-failover

新しいStateless Sessionの方式

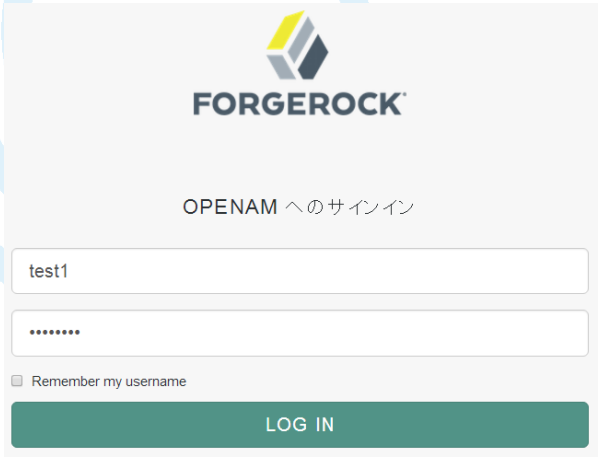
- サーバ側でセッション情報を一切共有しないため、大規模構成でも容易にスケールする

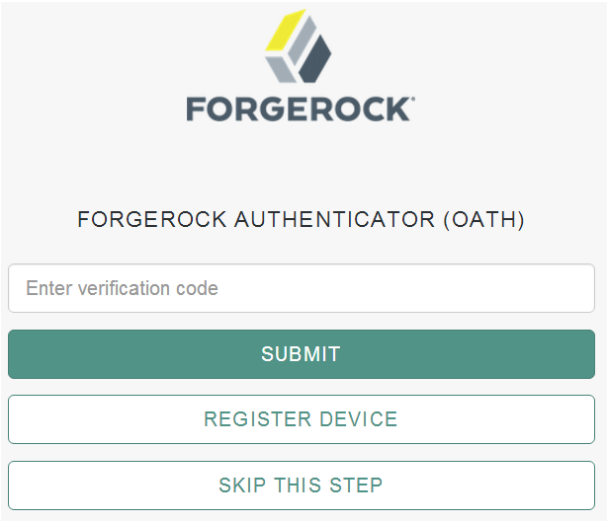


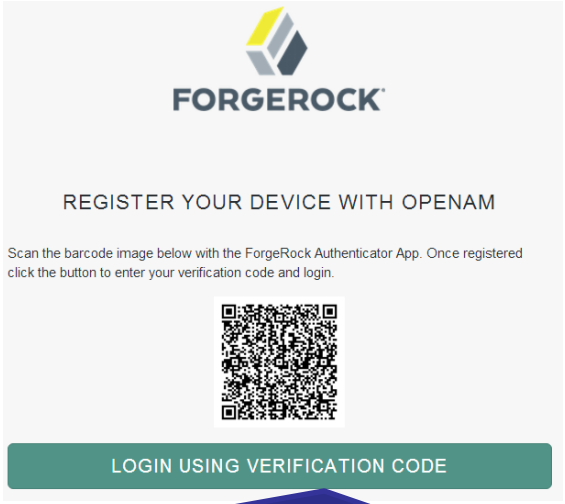
認証機能の強化

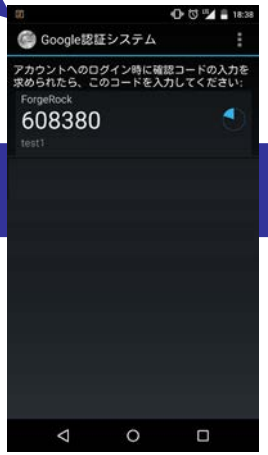
- OpenAMの2段階認証
  - 通常のユーザID/パスワード認証に加え、ワンタイムパスワード(OTP)を利用した2段階認証
  - OTPの方式は標準仕様のOATHに準拠
    - ✓HOTP、TOTPに対応
    - ✓Android/iOSアプリのGoogle AuthenticatorなどをOTP発行機として利用可能
- OpenAM13での強化ポイント
  - OPT発行のデバイス登録・管理のUIが標準で追加

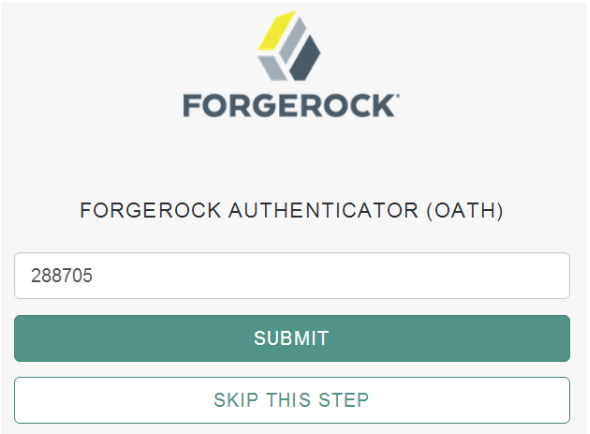
利用イメージ











デバイス登録画面を標準で用意

認可機能の強化

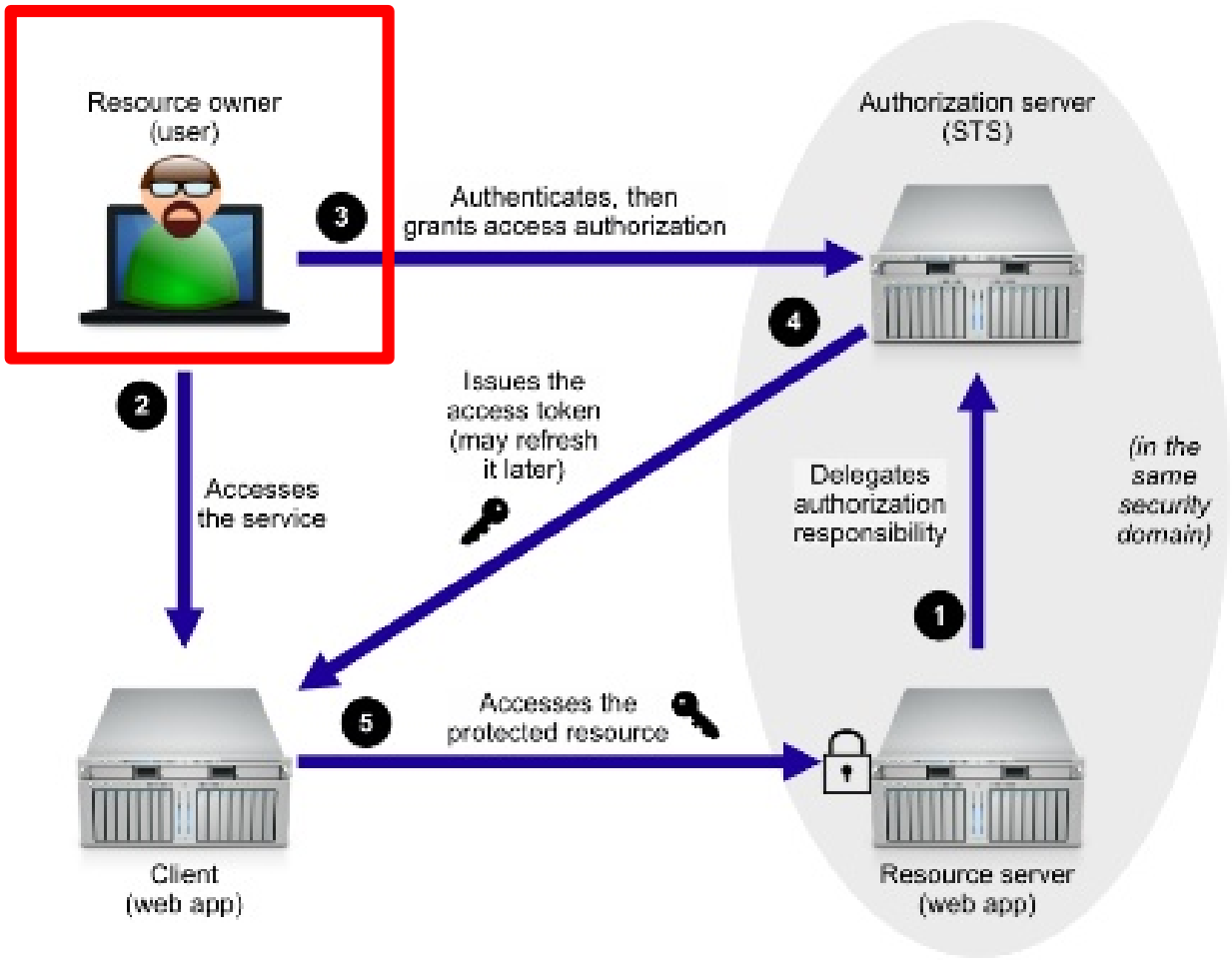
- UMA Authorization Server **機能の追加**
  - UMA**とは、**Kantara Initiative**のワーキンググループで仕様策定された** Web**ベースのアクセス管理プロトコル**
  - **現在**Web**で一般的に使われる認可プロトコルの**OAuth2**を拡張したもの**
    - ✓Facebook**の例**

出所: http://blog.unfindable.net/archives/1891

  - OAuth2**との大きな違い**
    - ✓OAuth2**はユーザーが外部のアプリケーションから自分のデータにアクセスするユースケースを想定**
    - ✓UMA**は自分のデータを他人が見れるように権限を設定するといった、他人とデータを共有するユースケースも想定**

OAuth2の場合

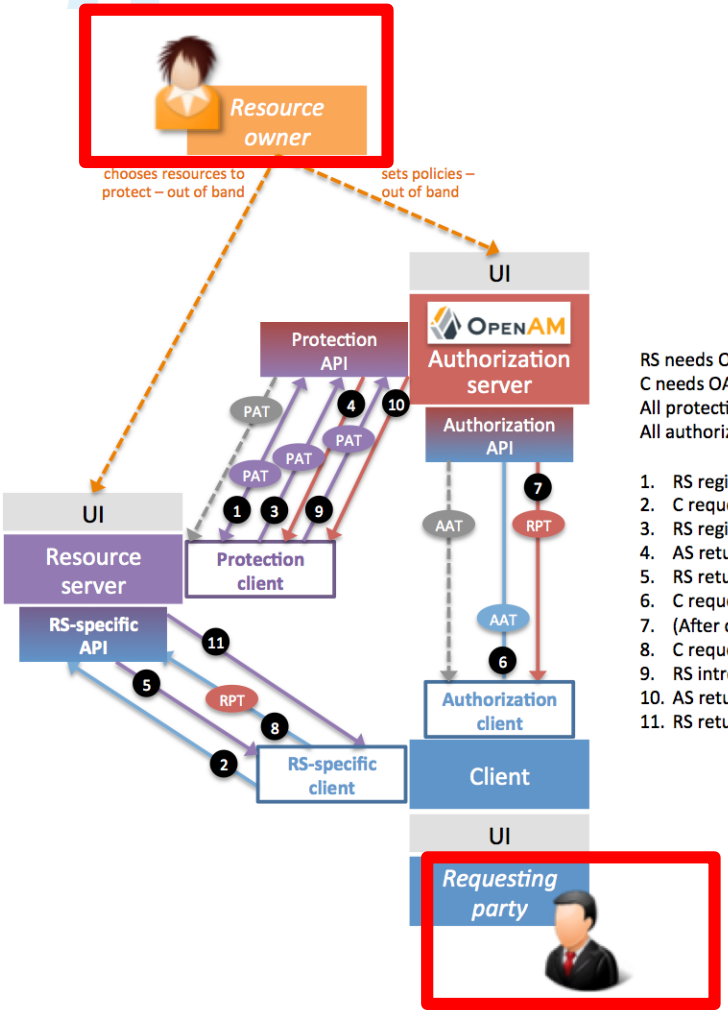
- リソースオーナー ＝ 利用者



出所: http://image.slidesharecdn.com/uma-for-ace-150114044632-conversion-gate02/95/uma-for-ace-10-638.jpg?cb=1421210822

UMAの場合

リソースオーナー ≠ 利用者 (でも良い)



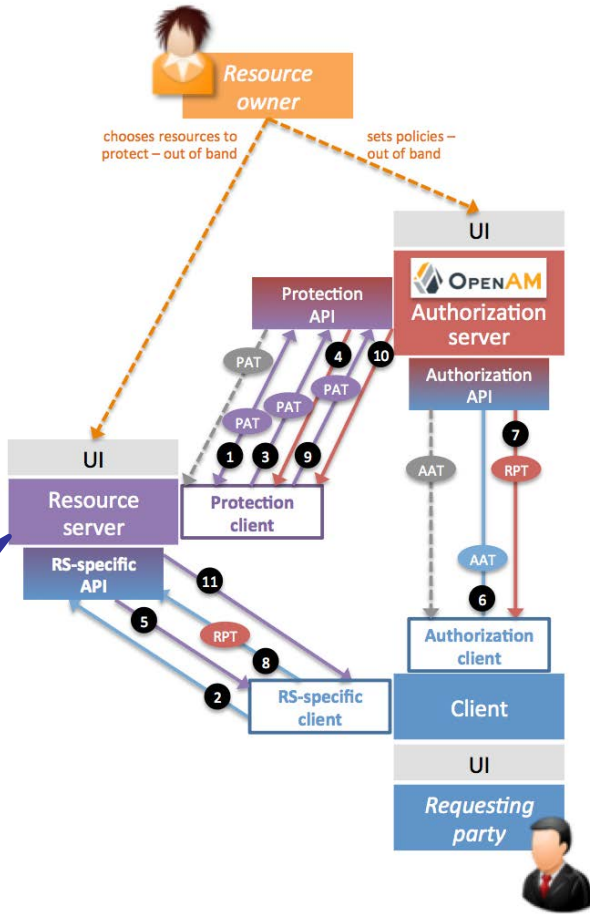
RS needs OAuth client credentials at AS to get PAT
C needs OAuth client credentials at AS to get AAT
All protection API calls must carry PAT
All authorization API calls must carry AAT

1. RS registers resource sets and scopes (ongoing – CRUD API calls)
2. C requests resource (provisioned out of band; must be unique to RO)
3. RS registers permission (resource set and scope) for attempted access
4. AS returns permission ticket
5. RS returns error 403 with as_uri and permission ticket
6. C requests authz data, providing permission ticket
7. (After claims-gathering flows not shown) AS gives RPT and authz data
8. C requests resource with RPT
9. RS introspects RPT at AS (if using default "bearer" RPT profile)
10. AS returns token status
11. RS returns 20x

出所: https://forgerock.org/app/uploads/2014/10/uma-info-highres.png

UMAの概要

- UMAの処理は大きく3つに分かれる
  - 1: リソースの登録
    - ✓リソースオーナ/リソースサーバ/認可サーバでのやりとり
  - 2: 認可処理
    - ✓リソースサーバ/認可サーバ/クライアント/Requesting Partyでのやりとり
  - 3: リソースアクセス
    - ✓Requesting Party/リソースサーバ/認可サーバでのやりとり

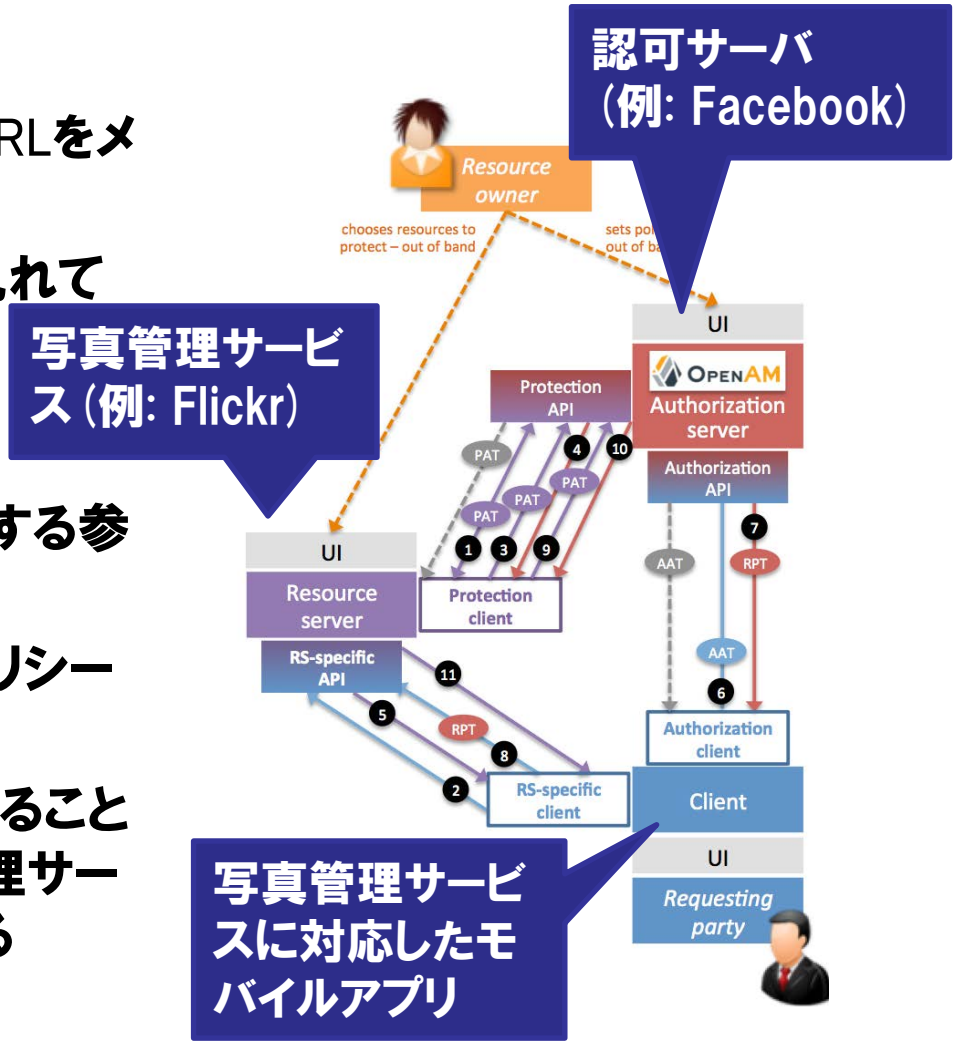


最終的には、リソースサーバにて、クライアントから渡されたトークン（RPT）から権限情報を取りアクセス制御する

```
{
    "active": true,
    "exp": 1447051542,
    "iss": "http://openam.example.org:8080/openam/oauth2",
    "token_type": "requesting_party_token",
    "permissions": [
        {
            "resource_set_id": "592e20b8-8f43-48a0-90cf-eae722ced36b0",
            "scopes": [
                    "http://photoz.example.com/dev/scopes/view",
                    "http://photoz.example.com/dev/scopes/all"
            ]
        }
    ]
}
```

UMAのユースケース (B2C)

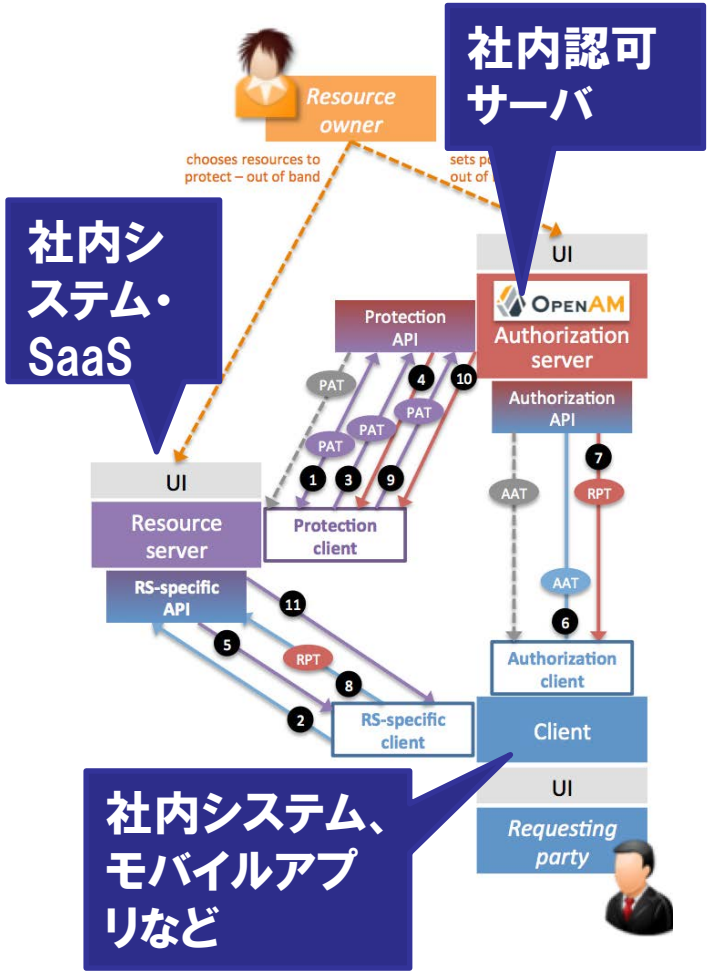
- 写真を知り合いに共有、といった個人データの共有
  - よくあるのは、自動生成されたURLをメール等で送信して共有するなど
  - これだと、URLが他人に渡ると見れてしまう
- UMAを使うと…
  - ユーザーは共有したい写真に対する参照権限を認可サーバに登録
  - 認可サーバにて、参照権限のポリシーを設定(公開範囲を友人、など)
  - 友人は認可サーバにて許可を得ることで、クライアントアプリは写真管理サービスからデータを取得可能になる



出所: https://forgerock.org/app/uploads/2014/10/uma-info-highres.png

UMAのユースケース (B2E)

- 企業内のリソースに対するアクセス制御に利用
  - 従来型の認証基盤だと、リバプロ/エージェントの導入が必要
  - URLベースでの制御しかできない
  - モバイルアプリや外部サービス等、リバプロ/エージェントが導入できない環境では使えない
- UMAを使うと…
  - リソースオーナーは企業であり、リソースサーバ(社内システム・SaaSなど)に対するアクセス権限を認可サーバに設定する
  - 認可サーバにて権限のポリシーを設定(○○部は見れる、部長以上は承認できる、など)
  - 社員は認可サーバにて許可を得ることでアクセス可能になる



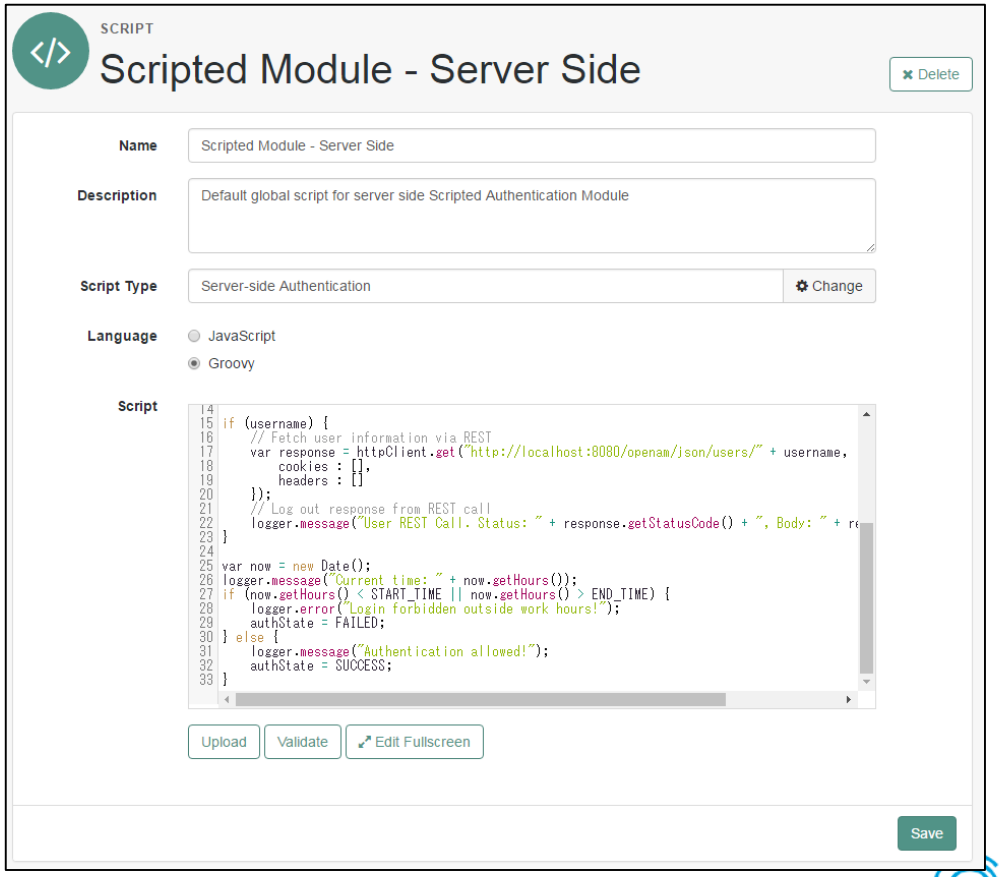
出所:
https://forgerock.org/app/uploads/2014/10/uma-info-highres.png

スクリプティングサービス

- **従来**
  - Java言語で認証モジュールなどのカスタマイズが可能
- **OpenAM13での強化ポイント**
  - 下記をスクリプト(JavaScriptまたはGroovy)にてカスタマイズ可能に
    - ✓認証ロジック
    - ✓認可ポリシー条件
    - ✓OpenID Connectクレーム

例）認証ロジックのカスタマイズ

- OpenAM管理コンソールよりスクリプトを定義
- AM9時～PM5時の間しか認証できないようにするサンプルソースが付いている



まとめ

まとめ

- 環境変化により、企業内の認証基盤に求められる要件も変化
  - 近年のキーワードとしては、SaaS/グローバル/モバイル/IoT
- オープンソース認証基盤であるOpenAMは変化にいち早く対応

- **OpenStandiaは、「攻めのIT」を支援します。**
- **オープンソースのことなら、なんでもご相談ください！**

オープンソースまるごと

OpenStandia™

Open Source Technology

お問い合わせは、NRI OpenStandiaチームへ

ossc@nri.co.jp

http://openstandia.jp/

未来創発

Dream up the future.