ownCloudとOpenAMで作るセキュアオンラインストレージ

株式会社 オージス総研
テミストラクトソリューション部
プロフェッショナルサービス第一チーム
千野 修平

Who am I ?

千野 修平（せんのしゅうへい）

(OpenAM,OpenIDM etc...)

開発リーダーやってます

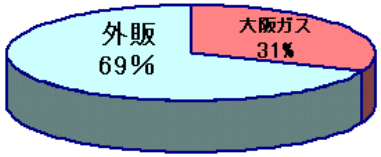
＜技＞ 認証、ID管理まわり
AWSまわり
Node.jsまわり

＜人＞ ４月から東京で
はたらいてます

オージス総研です。

株式会社オージス総研

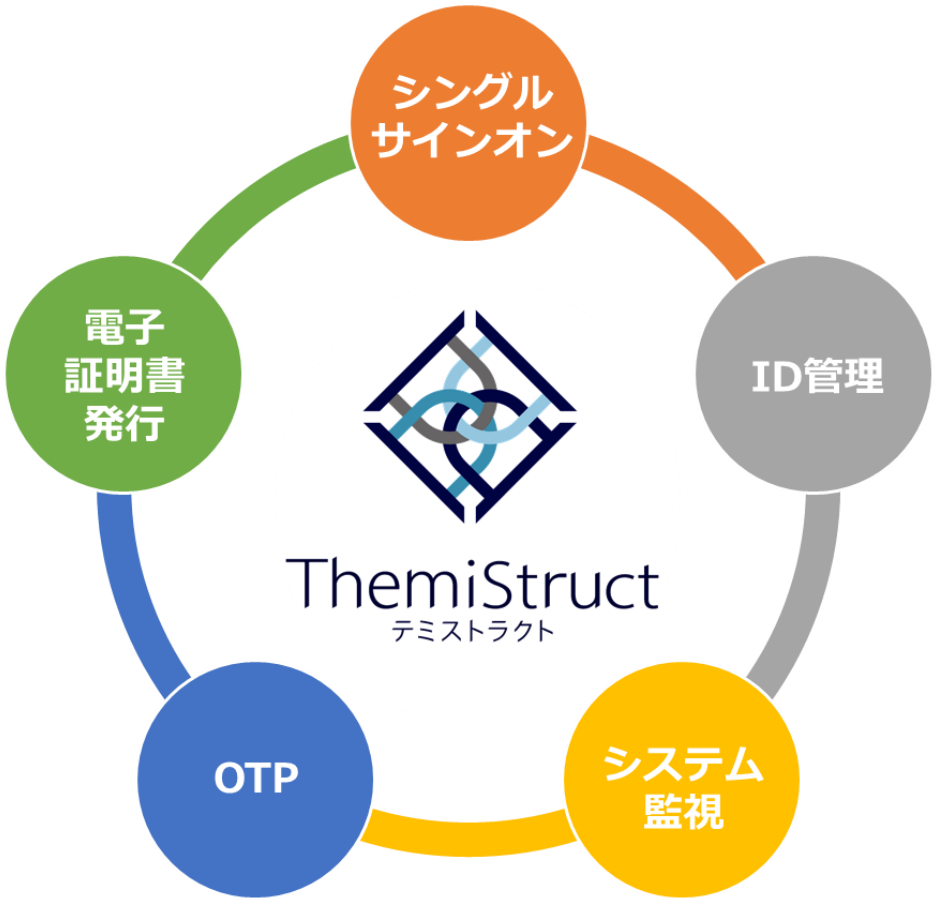
- 代表者：　代表取締役社長　西岡　信也
- 設　立：　1983年6月29日
- 資本金：　4億円　（大阪ガス株式会社100%出資）
- 事業内容：　システム開発、プラットフォームサービス、
  コンピュータ機器・ソフトウェアの販売、
  コンサルティング、研修・トレーニング
- 主な事業所
  - 本　社：　大阪府 大阪市西区千代崎3-南2-37 ICCビル
  - 東京本社：　東京都 港区港南2-15-1 品川インターシティA棟
  - 名古屋オフィス：　愛知県 名古屋市中区錦1-17-13 名興ビル
- 売上実績：　567億円（連結）　298億円（単体）　(2013年度)
- 従業員数：　3,104名（連結）　1,283名（単体）
- 関連会社：　さくら情報システム（株）、（株）宇部情報システム、（株）システムアンサー、
  OGIS International, Inc. 、上海欧計斯軟件有限公司（中国）
- オージス総研グループ 売上構成比　（連結）



大阪ガス GROUP

取得許可認定

ThemiStructとは？



ThemiStruct(テミストラクト)は、オージス総研が持つOSS活用のノウハウと導入実績に裏打ちされた統合認証ソリューションです。

テクニカルサポート

- 利用方法などの問合せへの回答
- 障害時の調査、回避策や代替案の提示、復旧の技術支援

プロフェッショナルサービス

- 要件実現方法の相談、回答
- 概念実証、技術検証の支援
- 自社で実施する開発、構築の技術支援

システムインテグレーションサービス

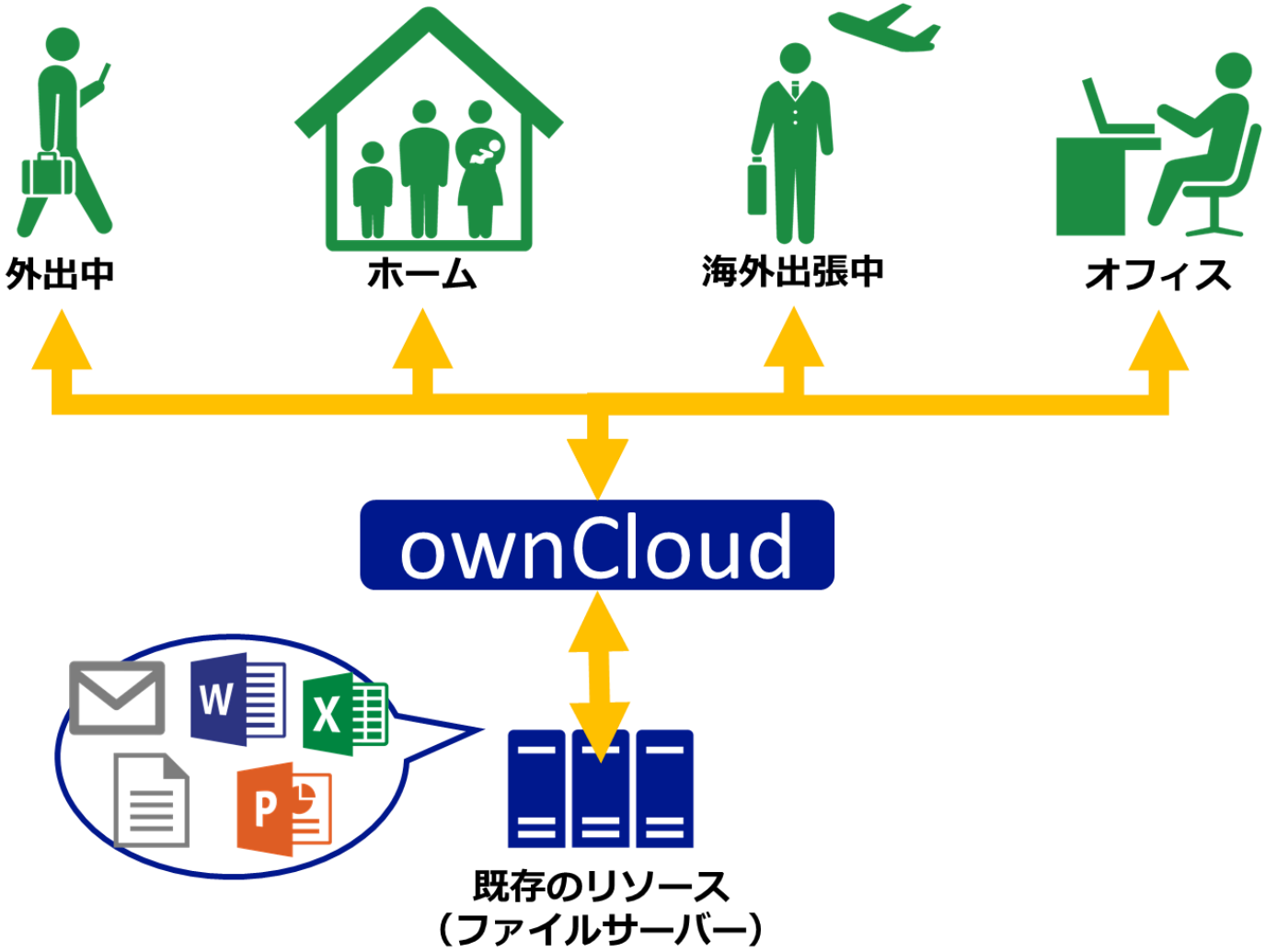
- お客様の要望に応じたシステムの実現を責任を持ってお引受け

ownCloud

OpenAM

おさらい：ownCloud活用のシナリオ

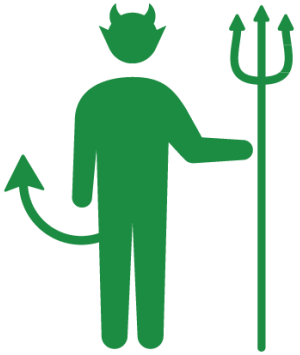
企業内のリソースとインターネットの境界に配置し、活用する。この構成を取ることで、できるようになる事は多い



「できるようになること」は増えるが・・・

- 同時に「やらなければならないこと」も増える

悪意を持って、情報を取得しようとする人たちへの対応



Authentication

認証を強くする

① 多段階で認証する
② 一時的な認証情報を使う
③ リスクをチェック
④ コールバックで本人確認をする

and so on ...

具体的にどうやるの？

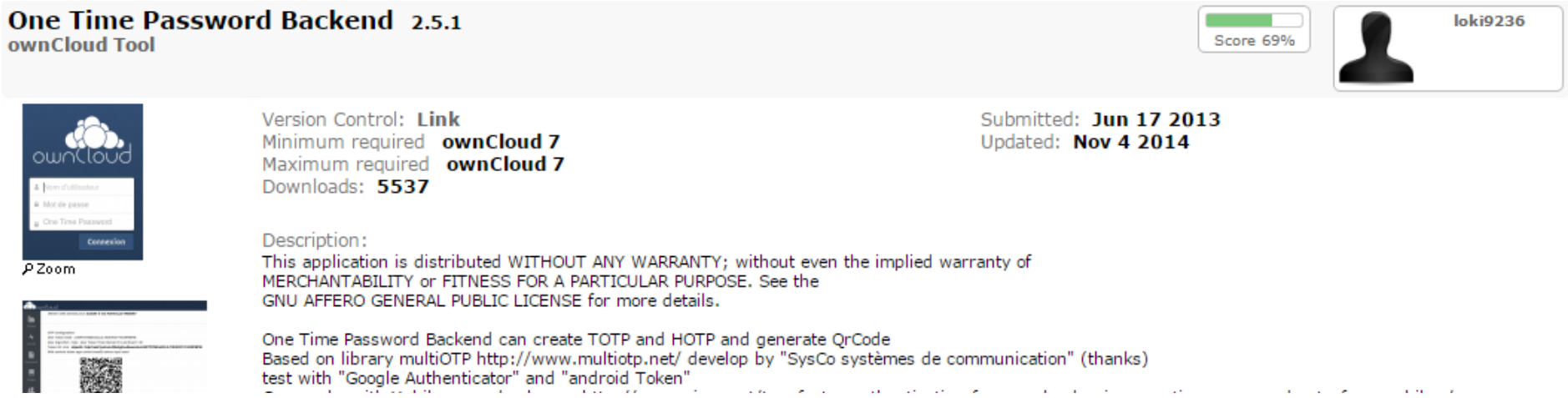
1. ownCloudのプラグインで対応

2. ownCloud Enterprise版で対応

3. OpenAMと連携して対応

ownCloudのプラグインで対応

- OTP認証対応プラグインありました（未検証）
  - 単純に、強化したいだけならこういうのでOK
  - 但し、アップデートが8.x系になってから行われていないので、動作するか不明
  - シードの管理、配布について、考えないといけない
  - 全員に一律のポリシーが適用される



▲ ソフトウェアトークン型のワンタイムパスワード認証プラグイン

ownCloud Enterprise版で対応

- セキュリティの機能は無いのか？
  - 「File Firewall」がある
  - だが、認証を強化する機能ではない
  - SAML/Shibboleth認証への対応で、他の認証基盤と連携する機能がある
  - （けど、エンプラ版…）

Enterprise　Editionにて提供される機能の代表例

No	機能	概要
1	File Firewall	接続種別、モバイルの種別、IP、Webクライアント等に応じて参照・アップロード可能なファイルの指定が可能です。
2	SAML/Shibboleth Authentication	Security Assertion Markup Language/Shibboleth認証への対応が可能です。
3	Home Directory Mounts	各ユーザーのOS上のhomeディレクトリをownCloudから参照することが可能です。既存のストレージをシームレスに拡張可能です。

(出典：http://owncloud.jp/solutions/enterprise)

3. OpenAMと連携して対応

OpenAMとは？

- 認証、認可、フェデレーション等の機能を備えた、WebアプリやクラウドサービスへのSSOを実現するOSS
- 旧Sun Microsystems社が開発したSun Java System Access Managerという商用製品がベース
- SunがOpenSSOとして、OSS化した後、OpenAMへ派生して開発が継続されている
- 日本を含む全世界で多くの導入実績があり、大規模な環境での稼働も実証されている

（出典） http://www.ogis-ri.co.jp/pickup/themistruct/note/note_sso02.html

できるっ！多要素認証！OpenAMで！

◘ その１：ワンタイムパスワード認証

できるっ！多要素認証！OpenAMで！

ロ その２：リスクベース認証

認証失敗回数は許容範囲か？
ソースIPは社外か？社内か？
過去のアクセスしてきた**ソースIP**と一致するか？
然るべき**Cookie**や**Header**がささっているか？
前回アクセスしてから**どれだけ経ったか**？
アクセスしてきた人の**所属**とか**役職**は何か？

OpenAMで取得できる情報を使って、
リスクの有／無を判断
次にどんな認証をユーザーに課すかコントロール

できるっ！多要素認証！OpenAMで！

◘ その３：デバイスID認証

ID/パスワード認証

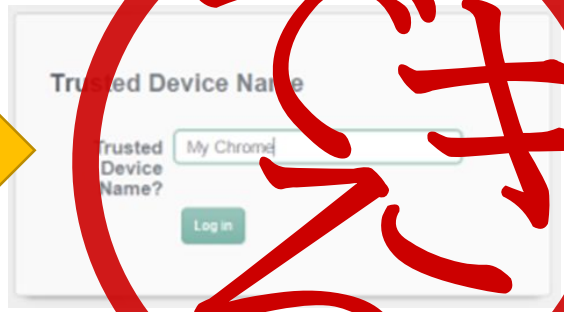
登録済み端末
ならシンプルに

ログイン成功

アクセスしているブラウザの
・種別・フォント
・画面サイズ
などを判別

OTPを要求

端末を登録

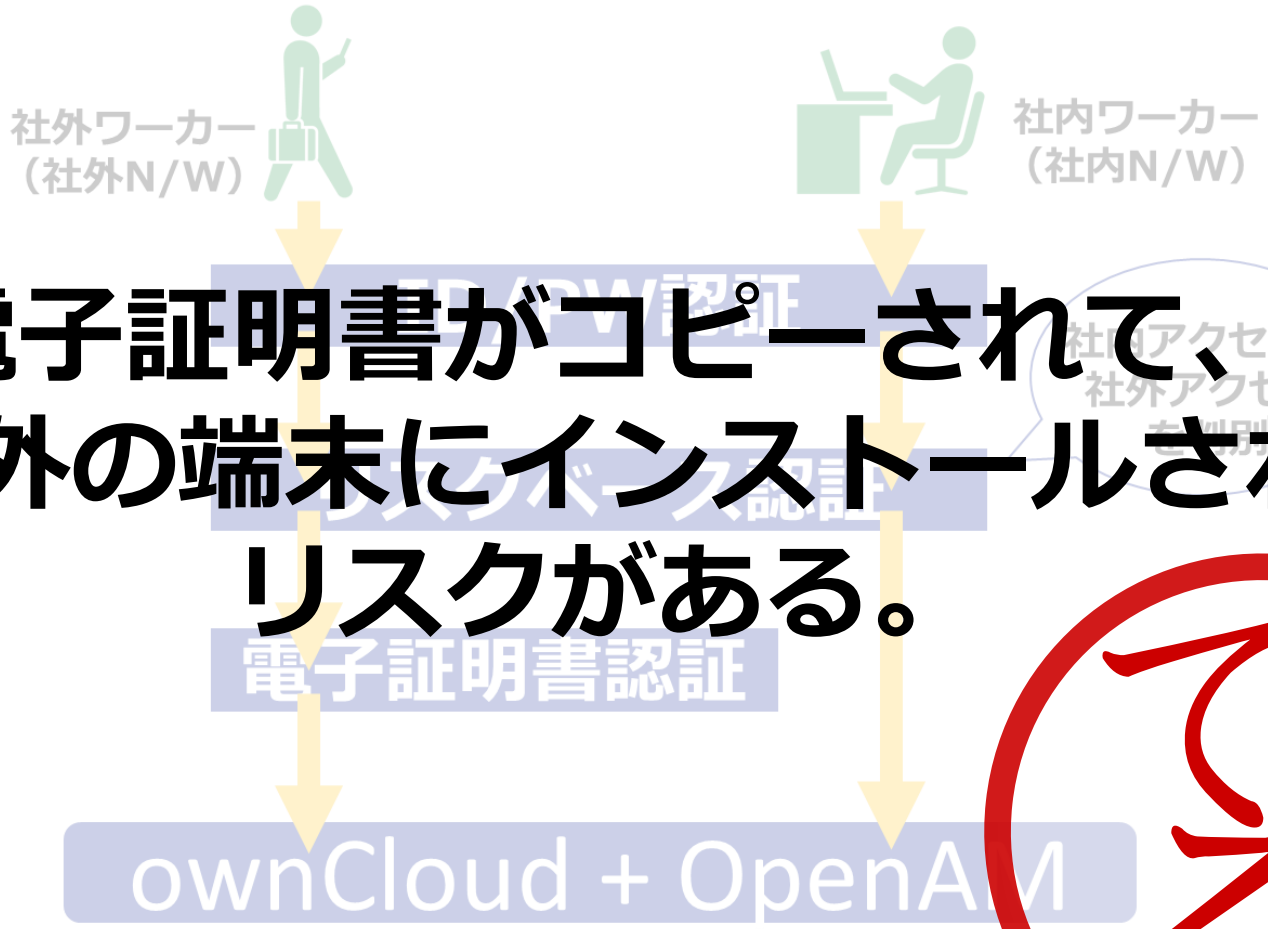


端末を登録

できる

できるっ！多要素認証！OpenAMで！

□ その４：電子証明書認証

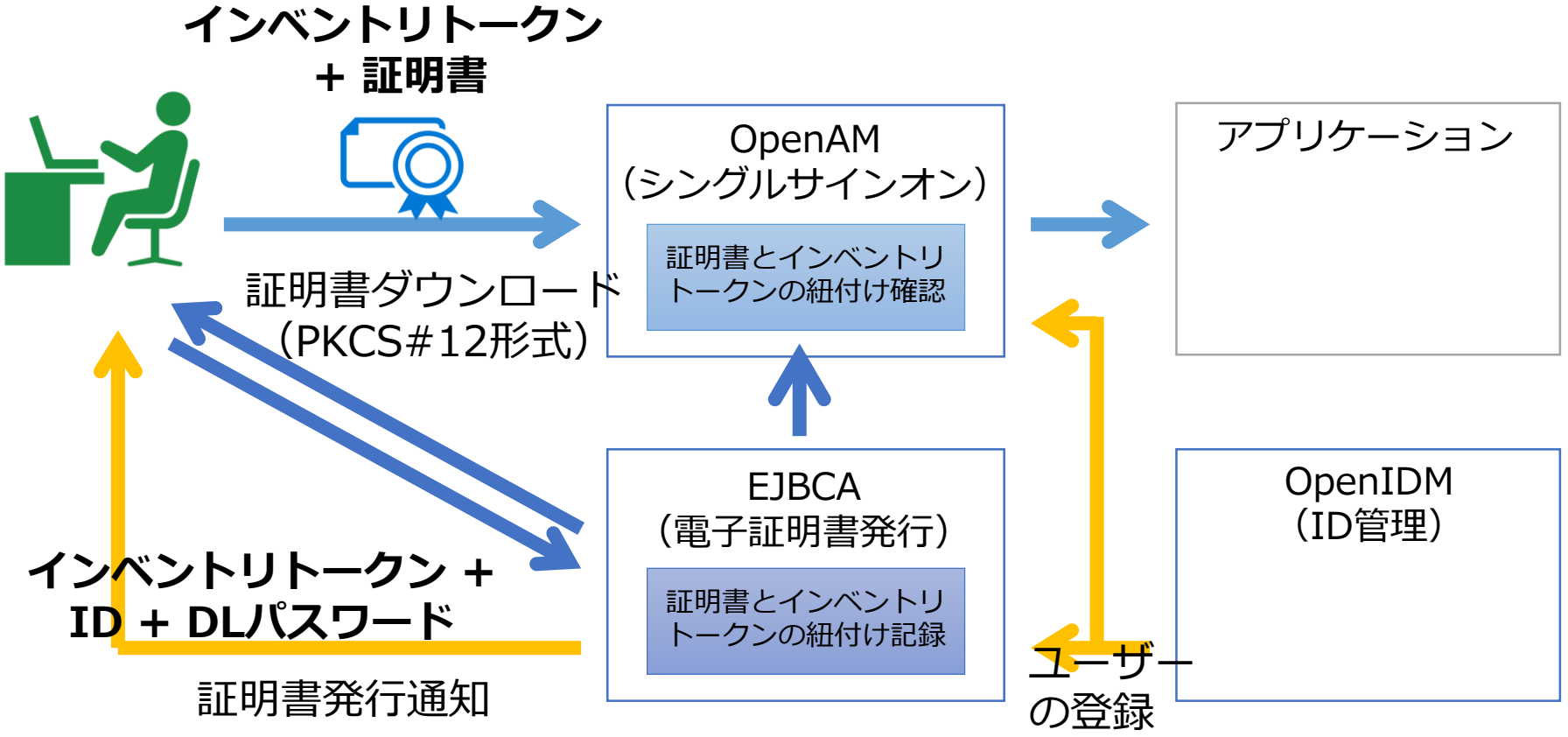


電子証明書がコピーされて、
想定外の端末にインストールされる
リスクがある。

できる

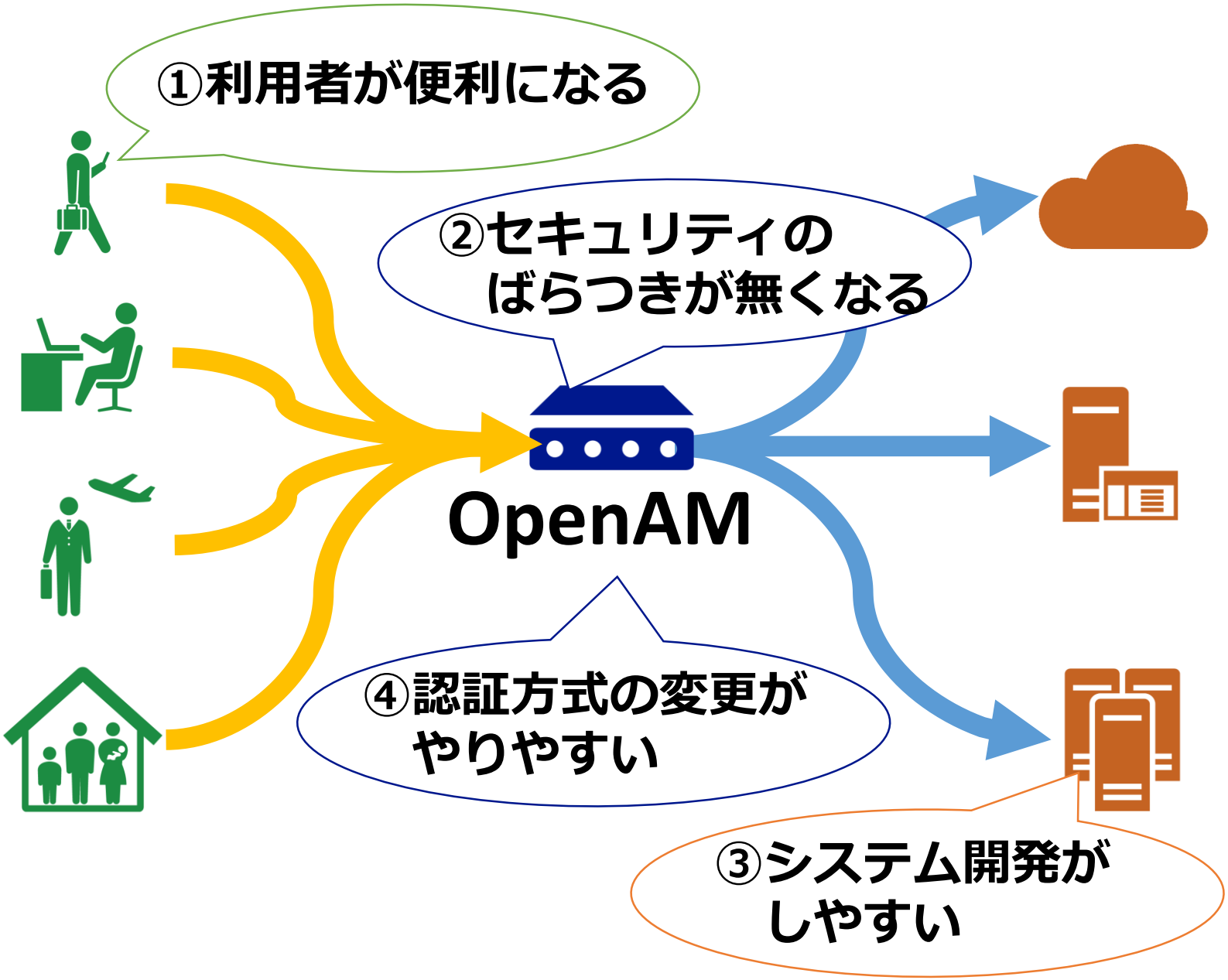
ちょっと宣伝：足らずはカスタマイズできる！

インベントリ認証
発行した証明書と証明書を導入した端末が
ちゃんと紐付いていることを保証する




認証強化
だけじゃない

わざわざ認証基盤（OpenAM）をたてる理由



① 利用者が便利になる

ID・パスワードの統合

利用者が複数のID・パスワードを覚える必要がなく、予期せぬ認証失敗や忘れを予防し負担を軽減できる。

シングルサインオン

アプリケーション毎にログイン認証をすることなく1度の認証でアプリケーションへアクセスできる。

結果

利用者の作業効率の向上、IT活用の促進に繋がる

② セキュリティレベルのばらつきが無くなる

パスワード運用の徹底

パスワード変更・パスワード忘れ対応・パスワードポリシーなど、パスワードを取り巻く運用を一元的に実施できる。

アクセスポリシーの統合

アプリケーションの認証レベル、アクセス制御を一元的に管理でき、セキュリティ強度の均一化ができる。

結果

システム、システム開発者に依存したばらつき、
ユーザーに依存したばらつきが解消される

③ システム開発がしやすい

認証画面の統合

認証経路を集約することでセキュリティ対策を集中的に実行できる。

認証セッションの統合

認証セッションの管理を一元化でき、セッションタイムアウト時間を統合できる。

結果

アプリケーション毎に認証機能を開発する必要がなくなる

④ 認証方式の変更がやりやすい

人・デバイス・ネットワークの認証強化

「知っている情報（ID/パスワードなど）」、「持っている情報（OTPや電子証明書）」や「アクセス元」の組み合わせでの認証ができる

リスク評価での認証

IPや認証の履歴、アクセス時間・曜日など、アクセス傾向による要素に基いて、ユーザーに課す認証の種類を変更することができる。

結果

**ID/パスワードを使った認証から多要素認証まで、
あらゆる認証方式を柔軟に組み合わせて適用可能**

むずかしそう…

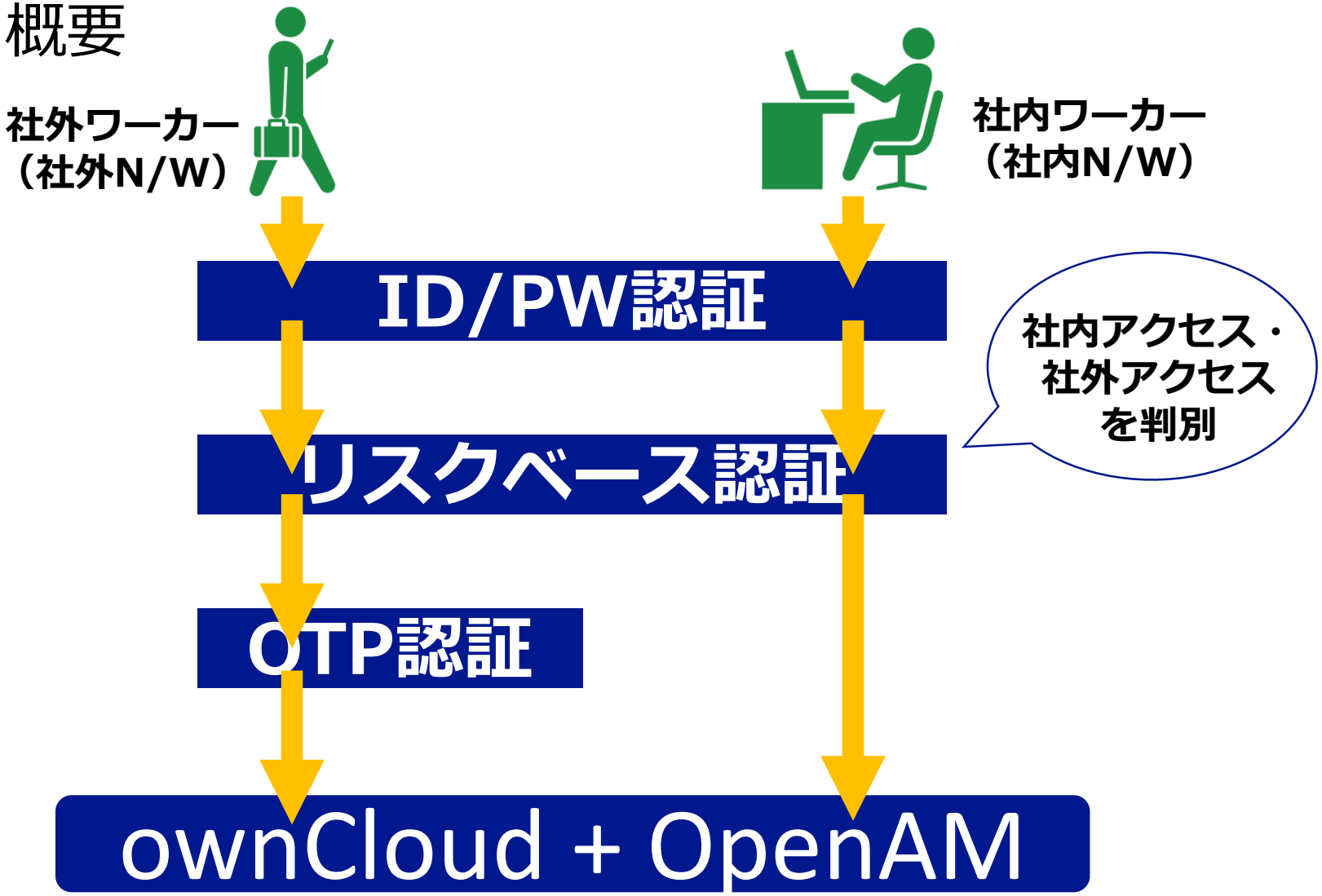
That sounds difficult.

Multi-Factor Authentication

1. Something you know
2. Something you have
3. Something you are

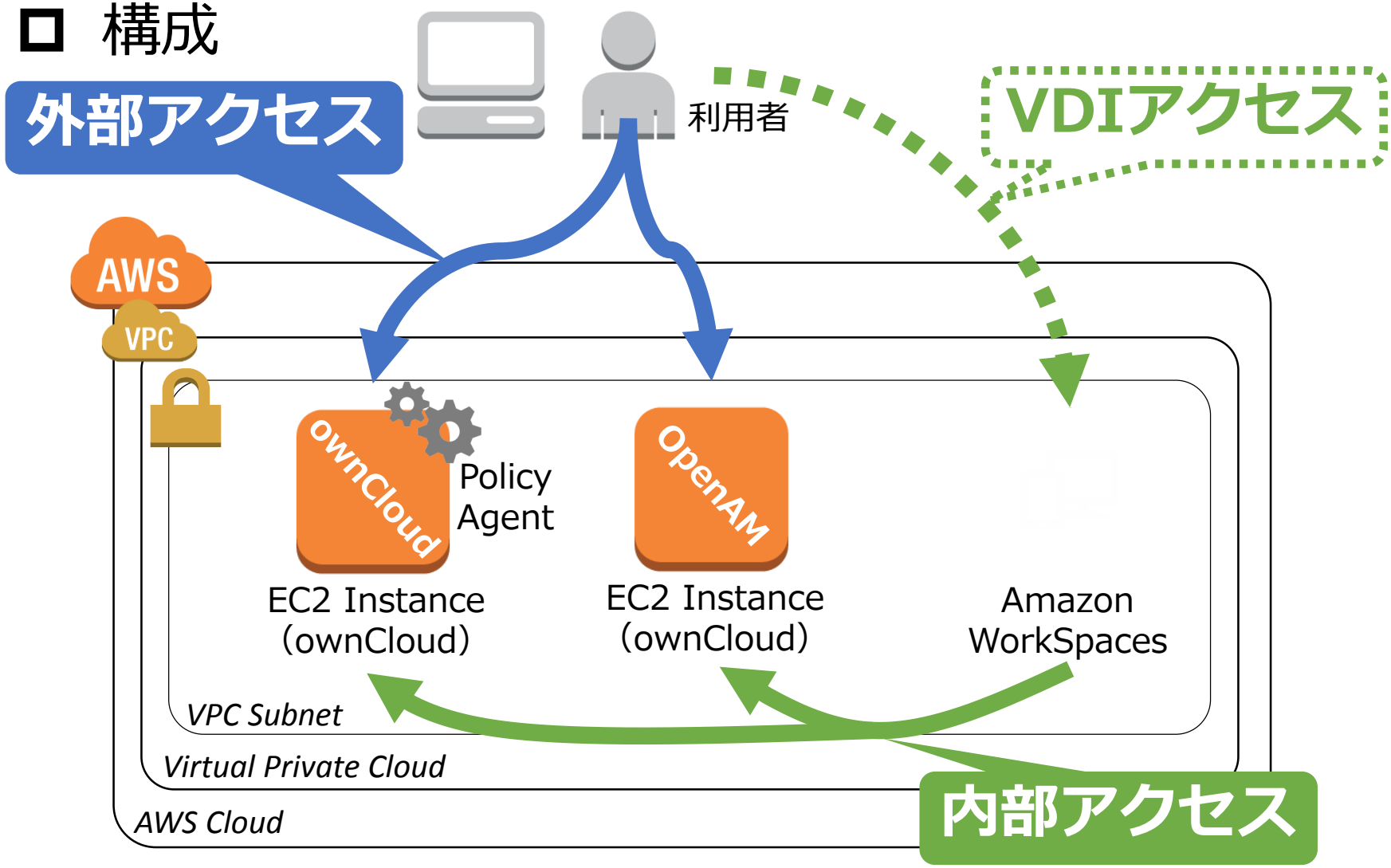
本日のデモ - 概要

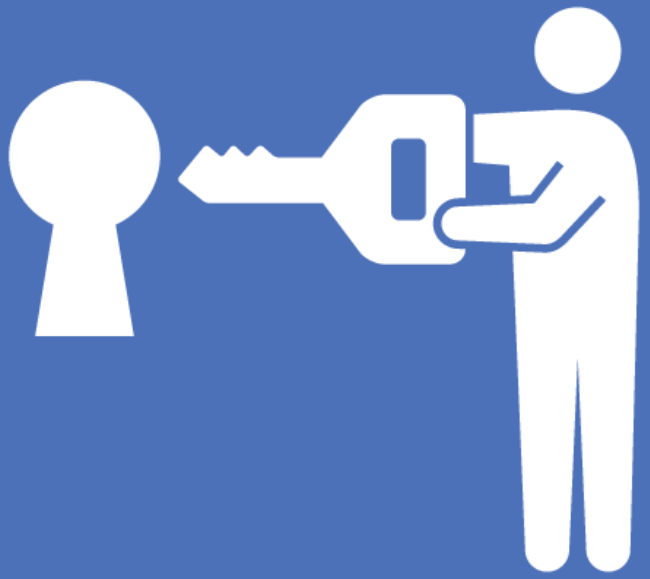
□ 概要



本日のデモ - 構成

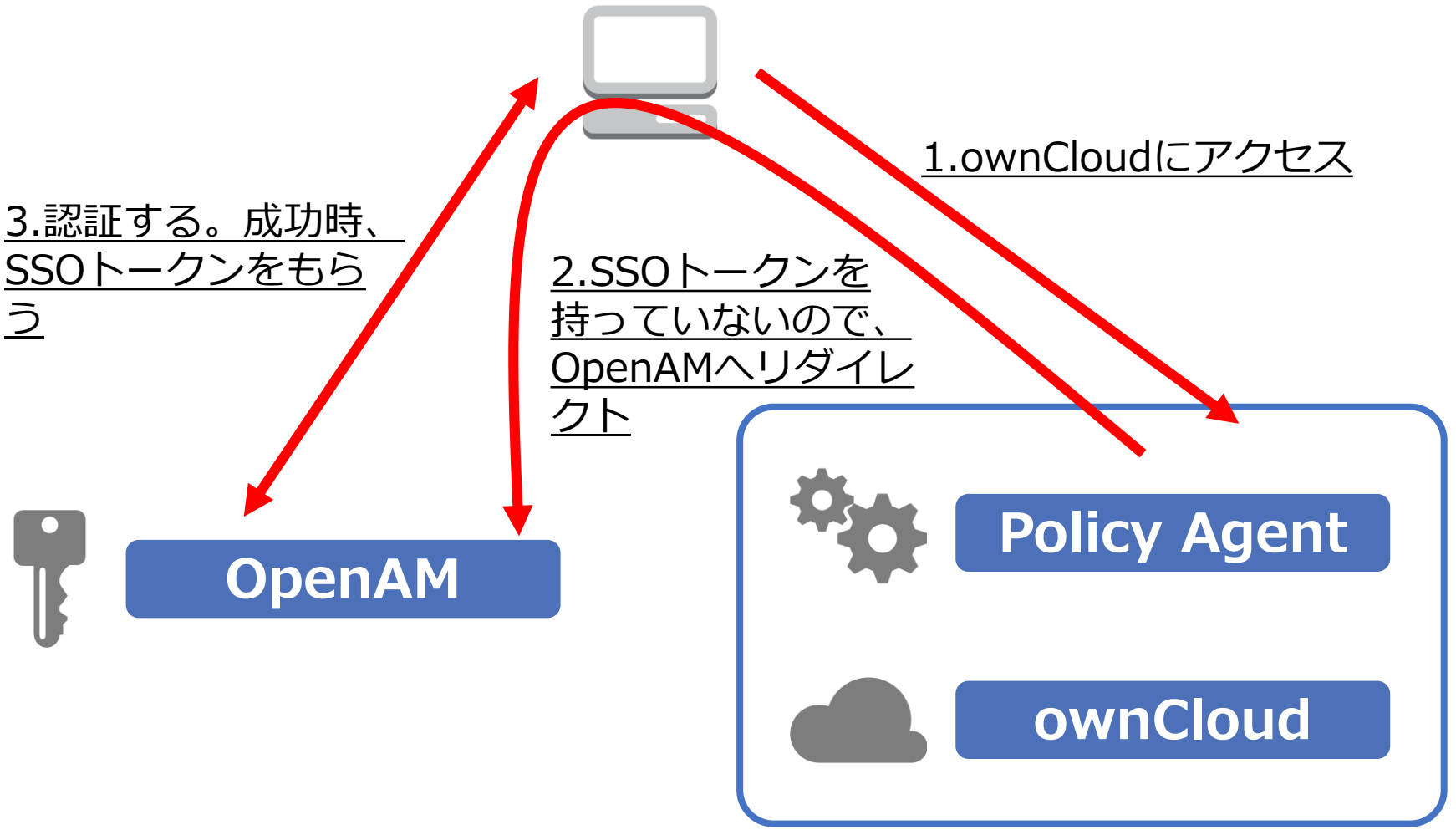
□ 構成



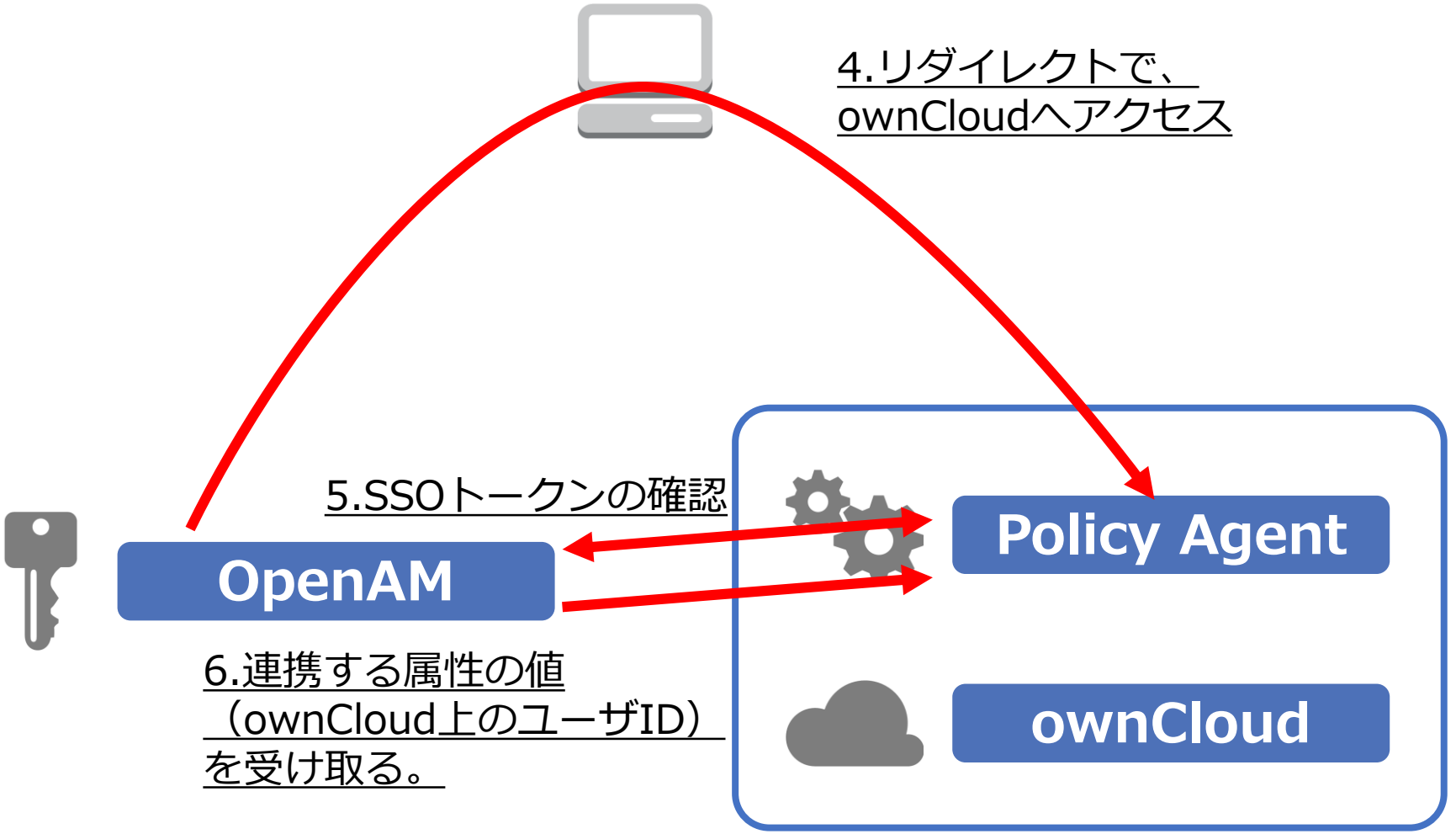


Let’s try ①
～OpenAMに認証を委譲する～

本日のデモ その１ - フロー①



本日のデモ その１ - フロー②



本日のデモ その１ - フロー③

7.連携された属性を
HTTP_HEADERに
セット

OpenAM

Policy Agent

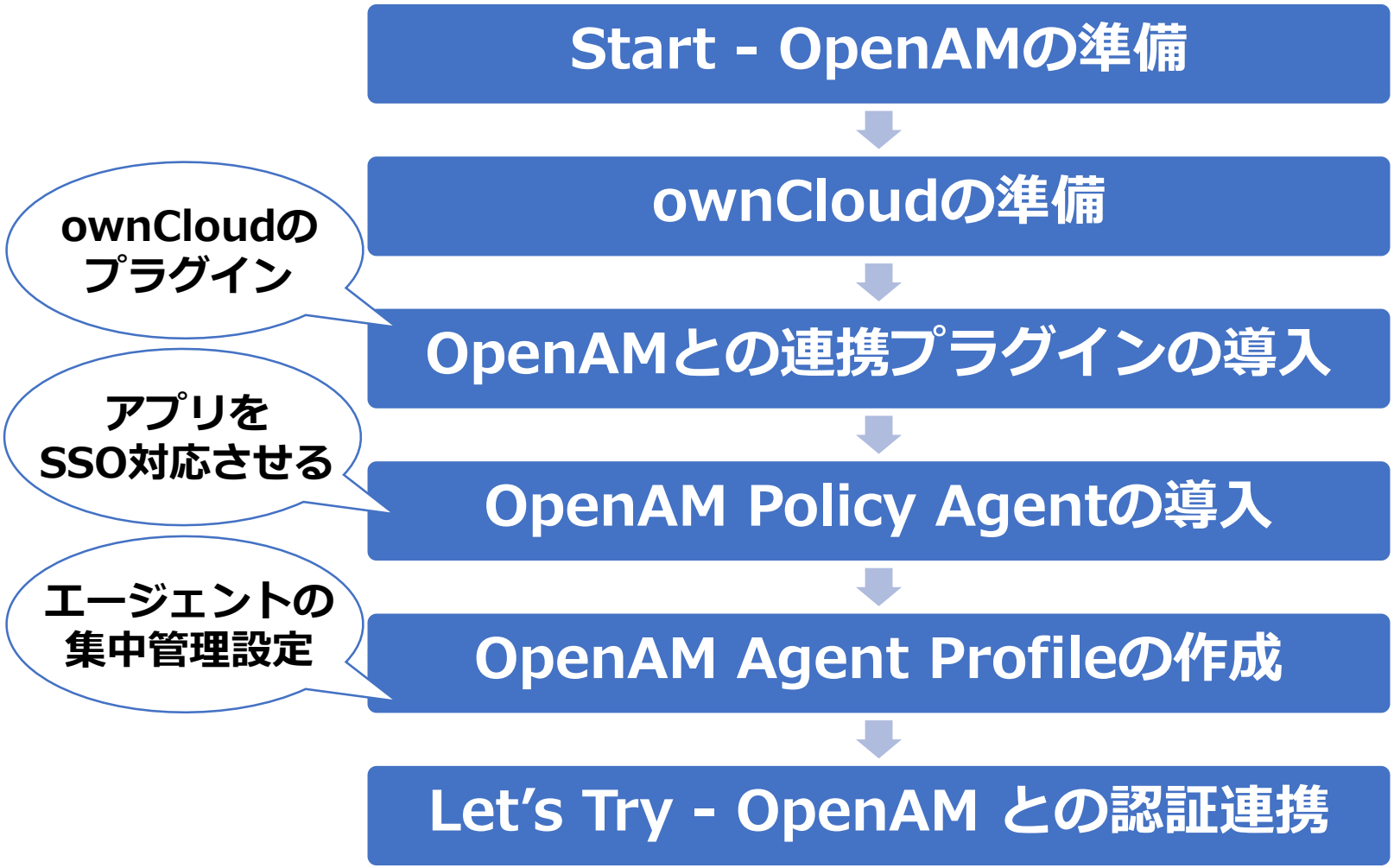
8.HTTP_HEADER付き
のリクエスト

ownCloud

9.Policy Agentから連携
されたHTTP_HEADERを
取得して認証実施する。

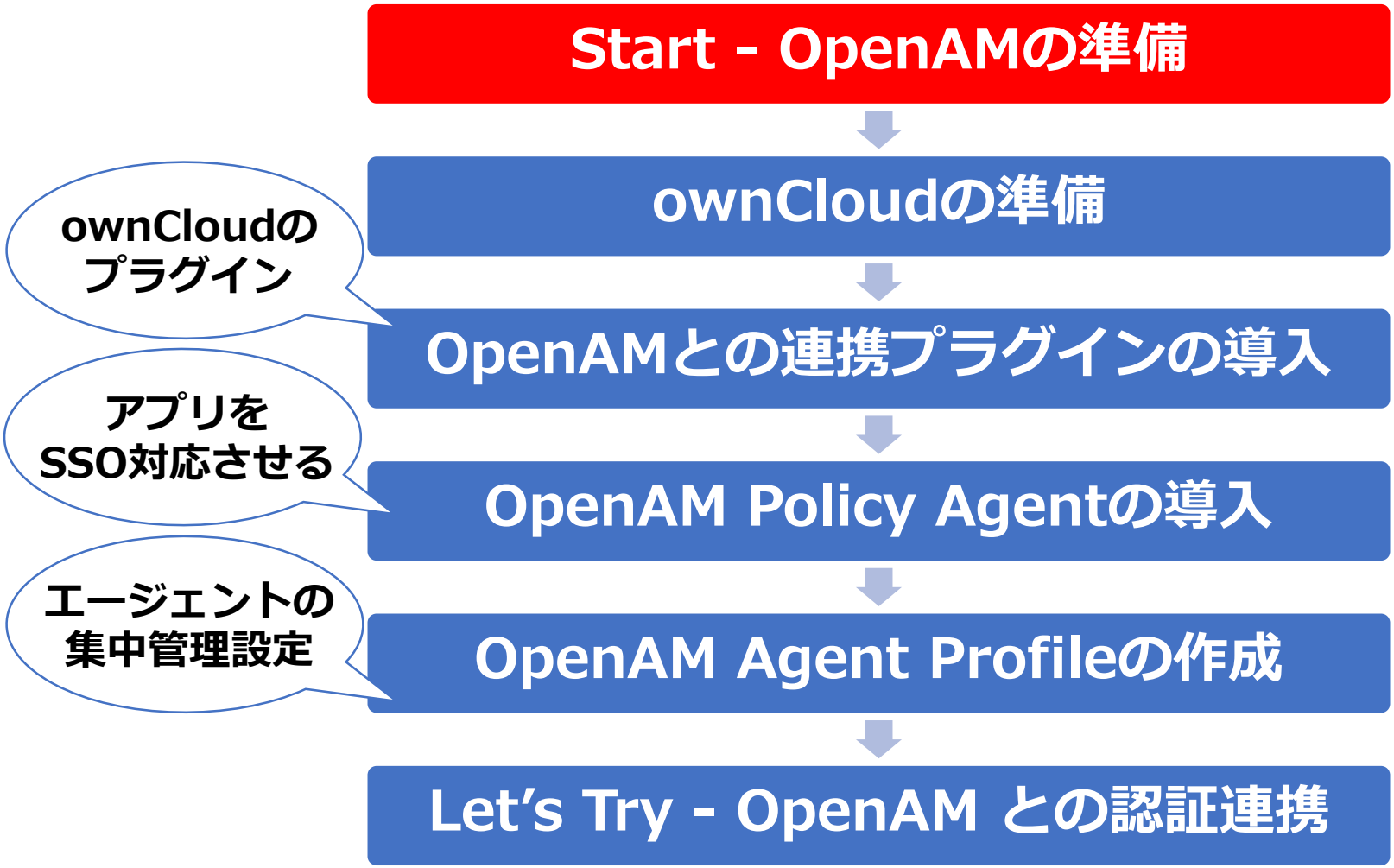
本日のデモ その１ - 連携までの流れ

◘ 手順のサマリ



本日のデモ その１ - 連携までの流れ

◘ 手順のサマリ



作業を開始する前にやっておくこと

- FQDNを決めておくこと
  - OpenAM
    →login.ossxusers.themistruct.net
  - ownCloud
    →owncloud.ossxusers.themistruct.net

OpenAMはドメインCookieを
SETしてくるので、FQDNは重要

- /etc/hostsで名前解決を実施しておくこと

- SSHやWEBのアクセスができるよう、Security Gatewayなどに穴を開けておくこと

OpenAMの準備

- 当社社員が執筆した「OpenAMインストール手順」を参考に準備する

（参考）http://www.ogis-ri.co.jp/pickup/themistruct/lineup/1228554_7683.html

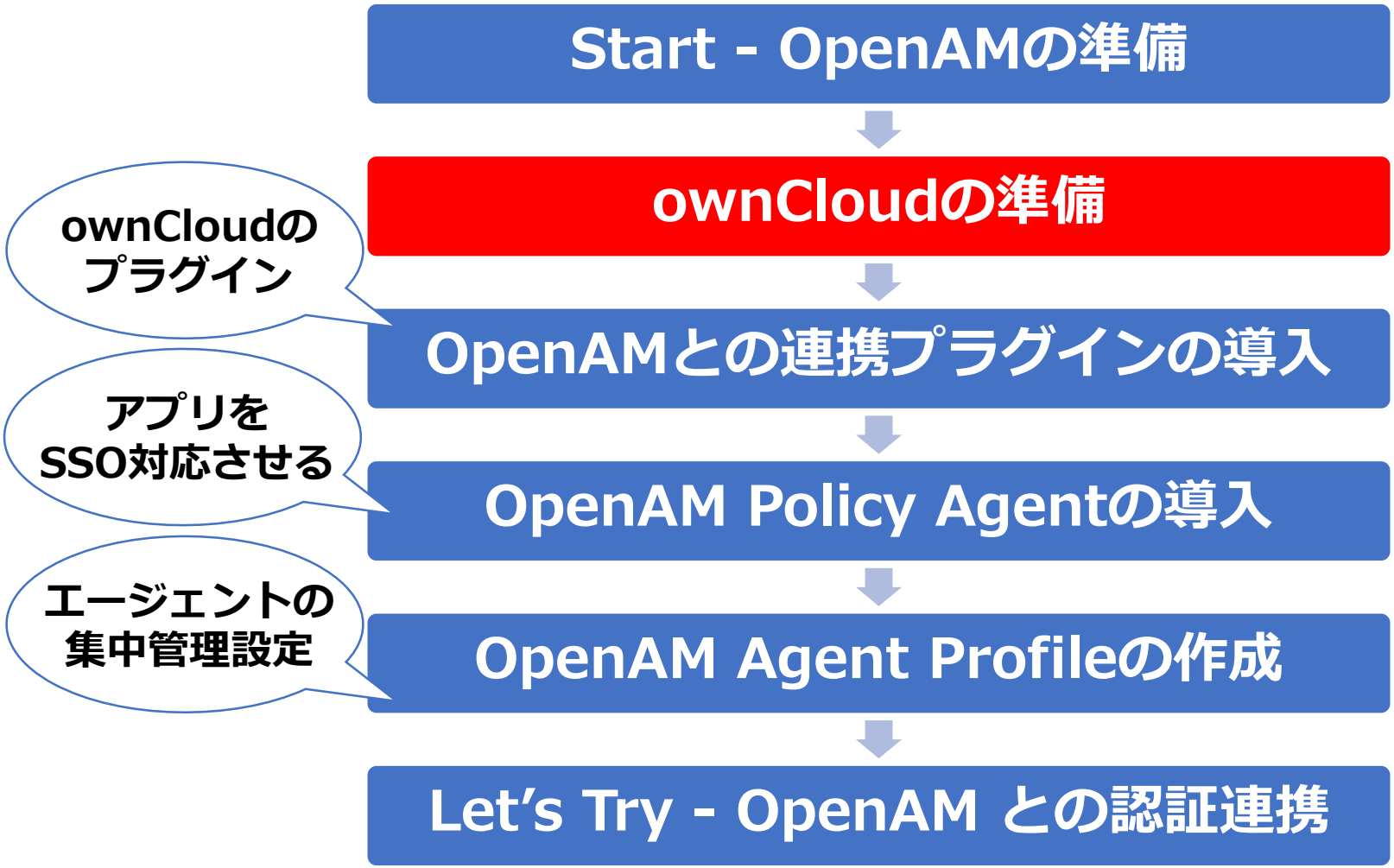
管理画面へアクセスできる状態

出来上がるもの

OpenAM
OGIS
インストール　で検索

本日のデモ その１ - 連携までの流れ

◘ 手順のサマリ



ownCloudの準備

- ownCloudをセットアップ！

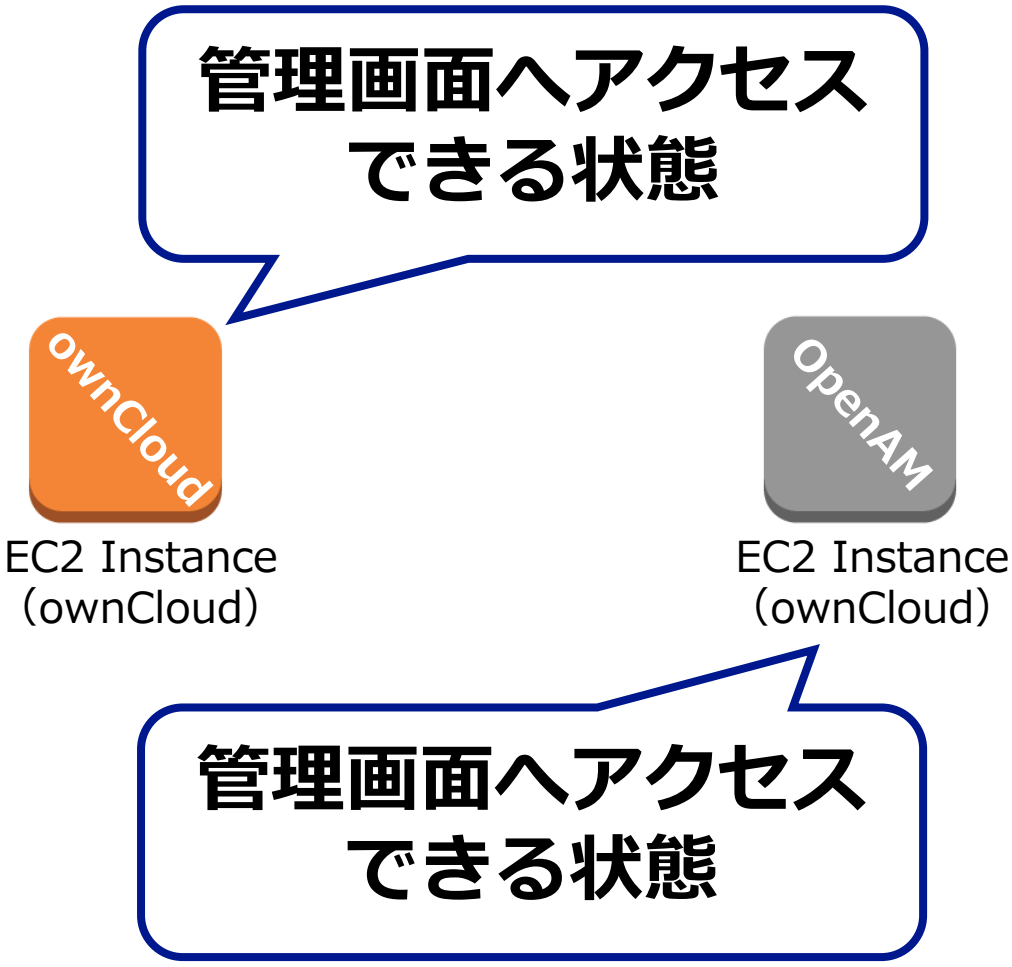
```
# yum -y install httpd24 php56 php56-pdo php56-gd && ¥
    wget https://download.owncloud.org/community/owncloud-8.1.0.zip ¥
    -O temp.zip; unzip temp.zip -d /var/www/html/; ¥
    rm -f temp.zip && chkconfig httpd on; service httpd start && ¥
    chown -R apache. /var/www/html/owncloud/
```

- Web経由で、管理者
  アカウントを作成する。

ブラウザで「http://FQDN or IP:80/owncloud/」▶
にアクセスすると右の画面が表示される。

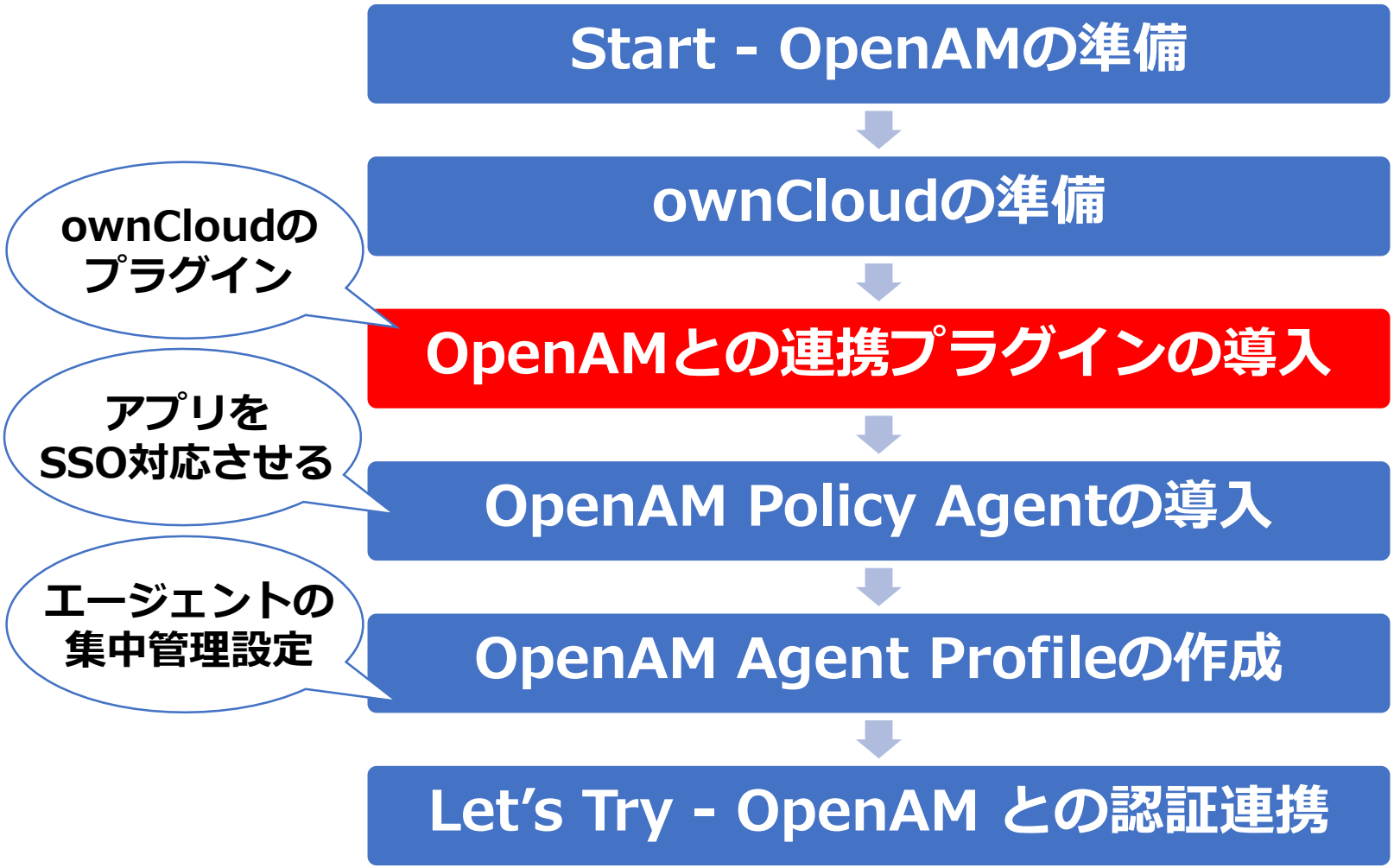
ownCloudの準備

- ここまでの手順でできあがるもの



本日のデモ その１ - 連携までの流れ

◘ 手順のサマリ



OpenAMとの連携プラグインの導入

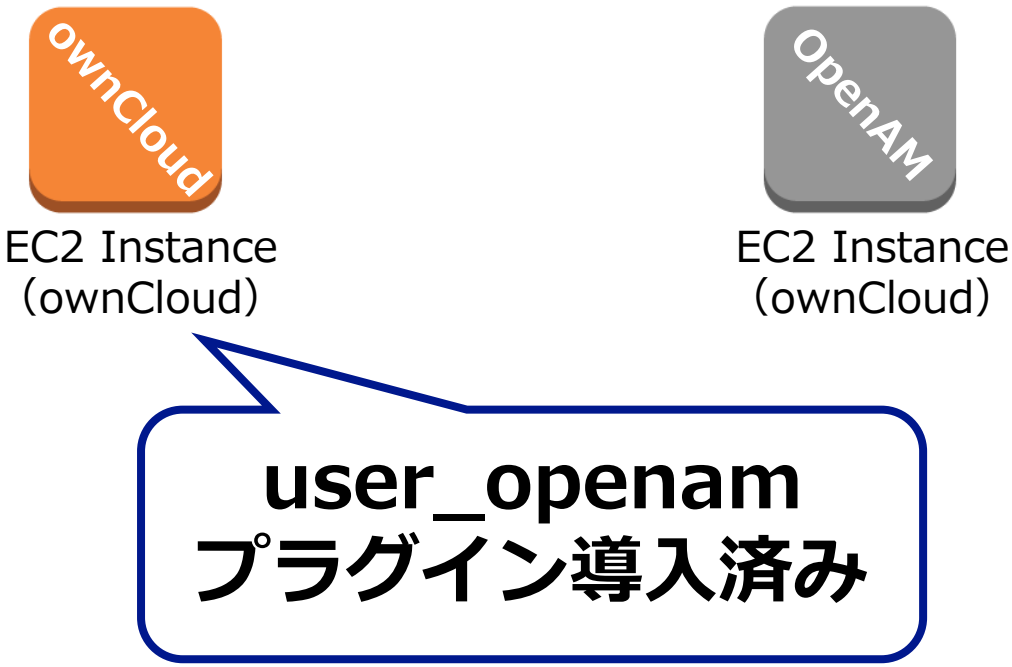
- ビーグッド・テクノロジー 高橋様 製のownCloudプラグインの導入
  - https://github.com/ukitiyan/user_openam
  - リクエストにのっているHTTP_HEADERの値でユーザを認証するプラグイン

```
# wget https://github.com/ukitiyan/user_openam/archive/master.zip ¥
  -O temp.zip; unzip temp.zip -d /var/www/html/owncloud/apps; ¥
  rm -f temp.zip; ¥
  mv /var/www/html/owncloud/apps/user_openam-master ¥
  /var/www/html/owncloud/apps/user_openam
```

- 管理画面＞ファイル＞アプリ＞無効＞User OpenAM Backend
  →「有効にする」

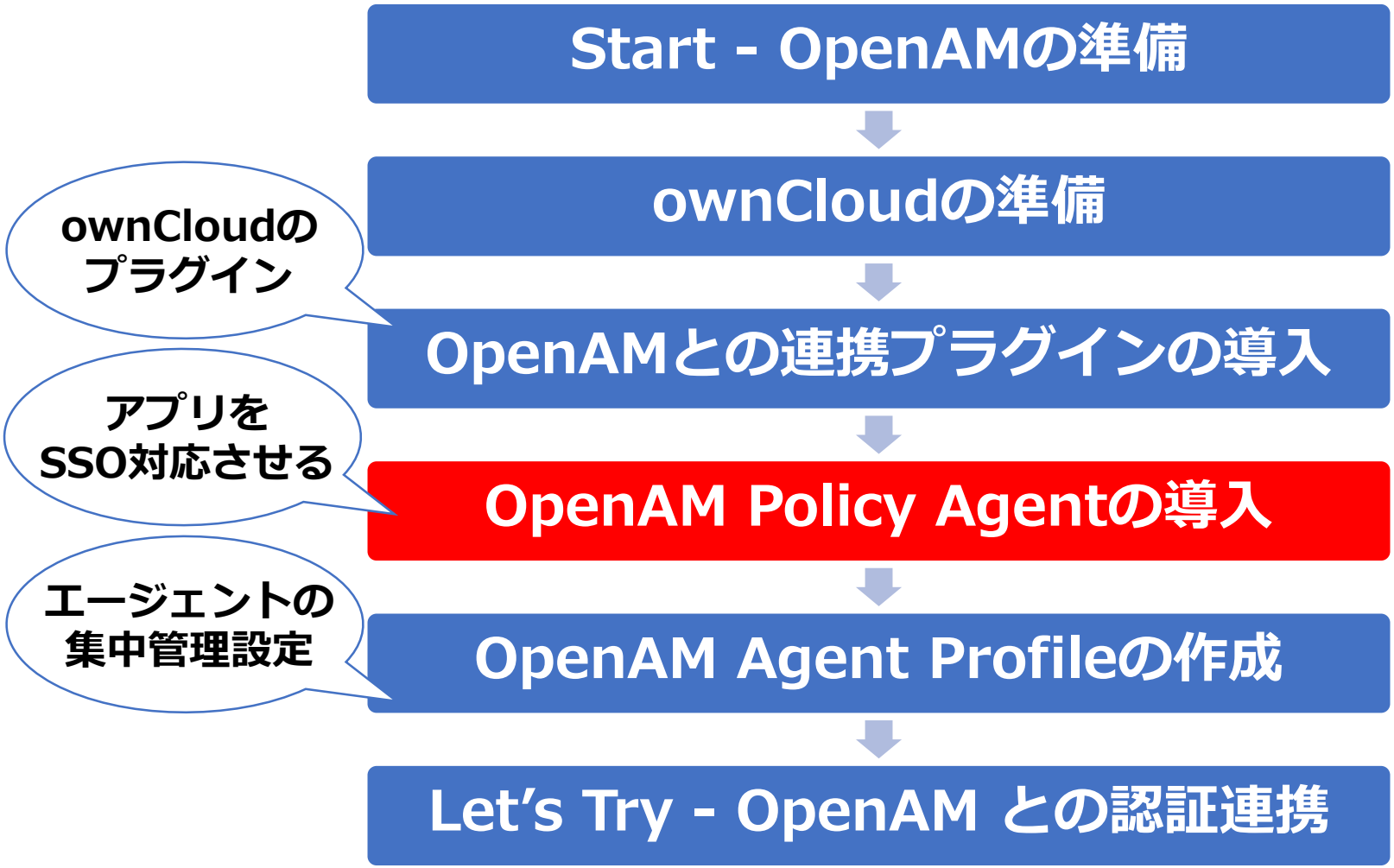
ownCloudの準備

- ここまでの手順でできあがるもの



本日のデモ その１ - 連携までの流れ

◘ 手順のサマリ



OpenAM Policy Agentの導入（１）

- Policy Agentをダウンロードして、展開

```
# wget
http://download.forgerock.org/downloads/openam/webagents/nightly/
Linux/apache_v24_Linux_64_agent_4.0.0-SNAPSHOT.zip -O temp.zip
&& unzip temp.zip -d /opt/
```

- Policy AgentがOpenAM上の設定ファイルを取得する際に使うパスワードの設定

```
# echo POLICY-AGENT-PASSWORD > ¥
  /opt/web_agents/apache24_agent/bin/agentpw
```

OpenAM Policy Agentの導入（２）

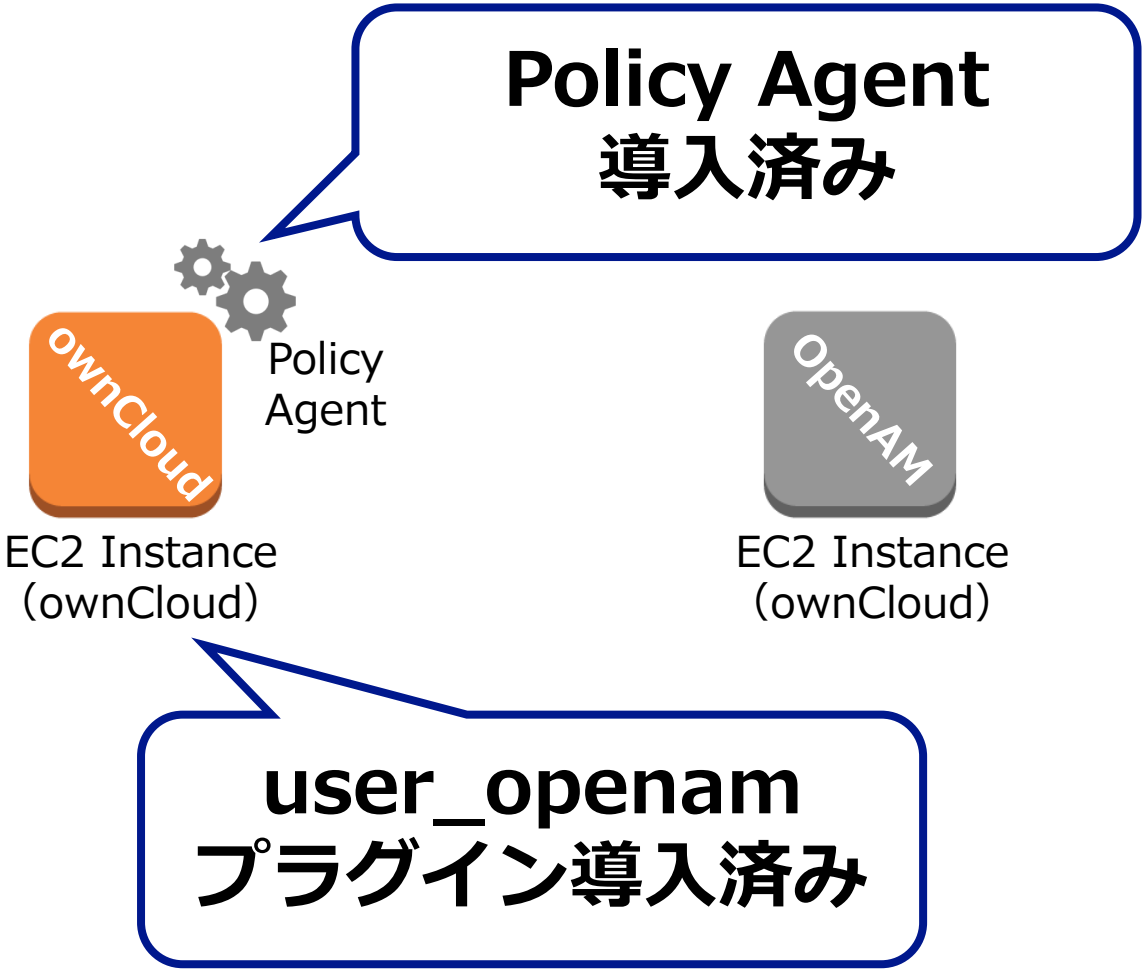
- Policy Agentのインストール（メッセージを省略しています）

```
# /opt/web_agents/apache24_agent/bin/agentadmin --install

Enter the Apache Server Config Directory Path [/opt/apache24/conf]:
/etc/httpd/conf

OpenAM server URL: http://login.ossxusers.themistruct.net:80/sso

Agent URL: http://owncloud.ossxusers.themistruct.net:80

Enter the Agent Profile name: owncloud-agent

Enter the path to the password file: password

Please make your selection [1]: 1

Thank you for using OpenAM Policy Agent
```

Policy AgentがOpenAM上の設定ファイルを取得する際に使うIDとパスワードの設定

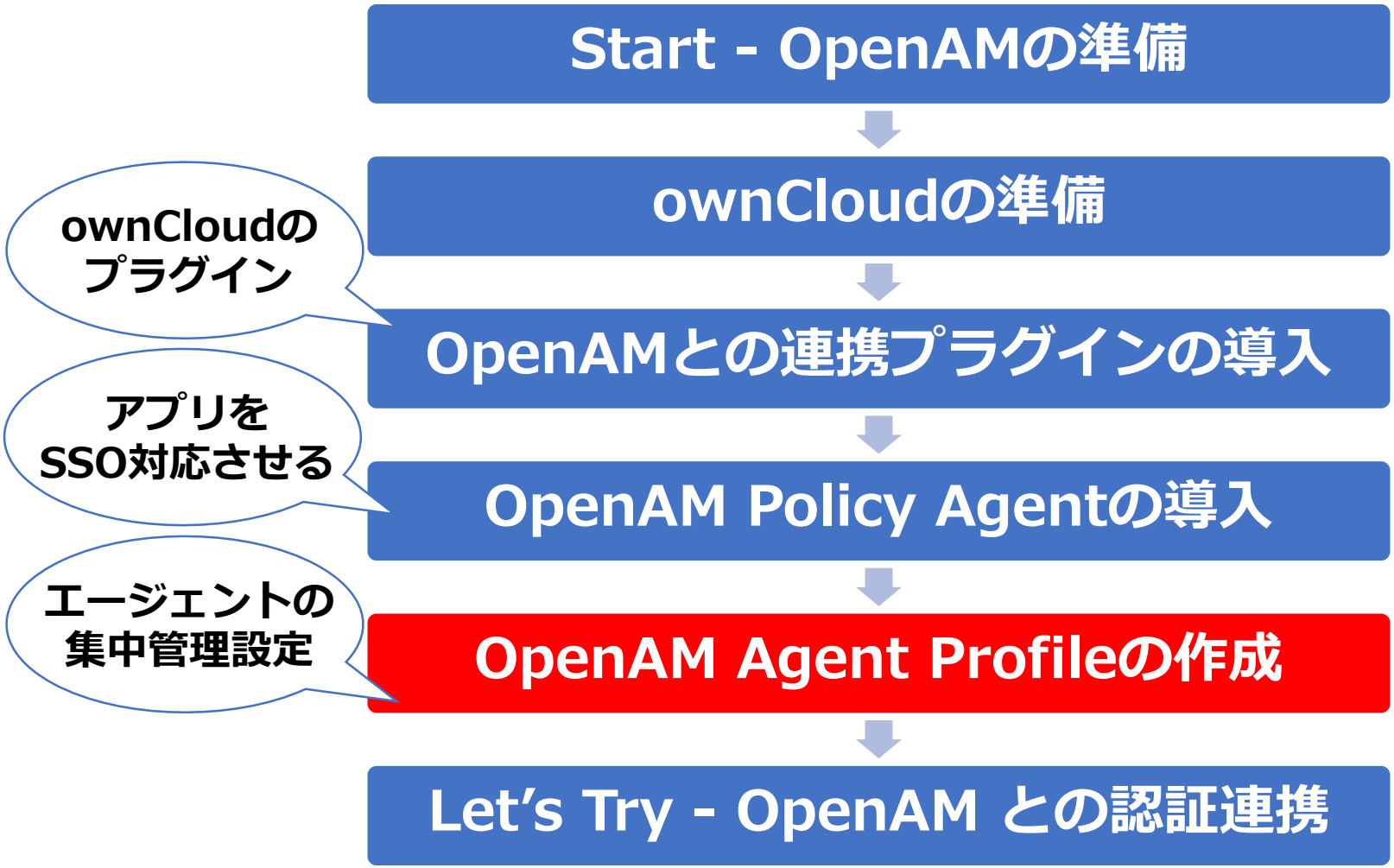
OpenAM Policy Agentの導入（3）

- ここまでの手順でできあがるもの



本日のデモ その１ - 連携までの流れ

◘ 手順のサマリ



OpenAM Agent Profileの作成（１）

設定画面

- OpenAMの管理画面に「amadmin」ユーザーでログイン
- アクセス制御＞最上位のレルム＞エージェント＞新規 から以下の設定を実施。

新しい Web

* 名前: owncloud-agent

* パスワード: ••••••••

* パスワードの再入力: ••••••••

Policy AgentがOpenAM上の設定ファイルを取得する際に使うIDとパスワードの設定

設定: ○ ローカル ◉ 集中

エージェントプロパティーが格納されている場所。「ローカル」は、エージェントが実行されているサーバーです。

* サーバー URL: http://login.ossxusers.themistruct.net:80/sso

プロトコル://ホスト:ポート/deploymentUri (たとえば、http://opensso.sample.com:58080/opensso)

* エージェント URL: http://owncloud.ossxusers.themistruct.net:80

プロトコル://ホスト:ポート (たとえば、http://agent1.sample.com:1234)

OpenAM Agent Profileの作成（２）

設定画面

- エージェントプロファイル名＞一般＞SSOのポリシー強制モード →「有効」
  - 認可を行わない設定

- エージェントプロファイル名＞アプリケーション＞プロファイル属性処理
  - ownCloudのプラグイン（user_openam）にHTTP_HEADERを連携する設定

本日のデモ - 認証強化の手順

◘ 手順のサマリ

OTP認証の
インスタンス

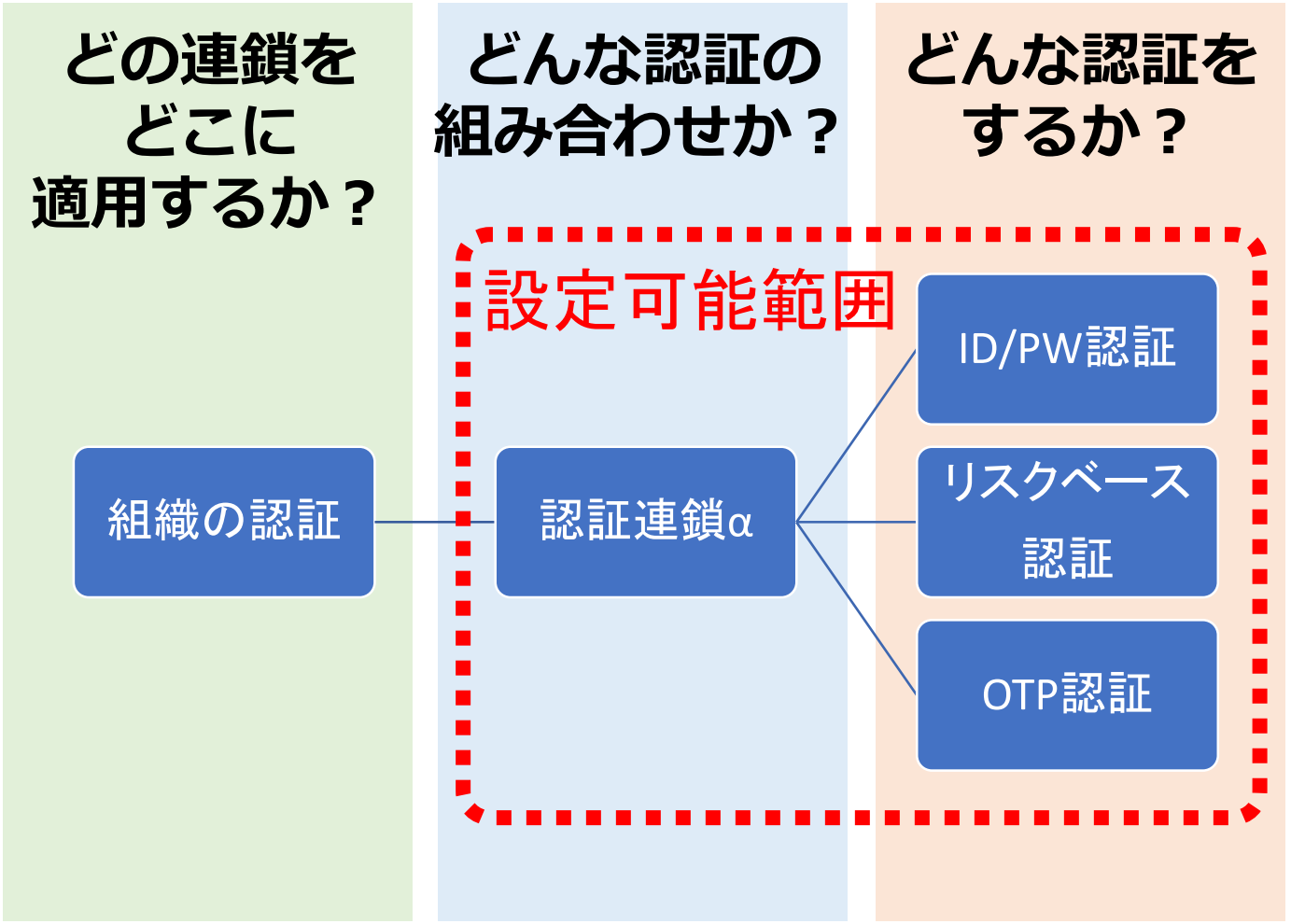
モジュールインスタンスの作成

インスタンスを
組み合わせる

認証連鎖の設定

Let’s Try - OTP認証で
ownCloudへログイン

OpenAMにおける認証連鎖設定の構造



本日のデモ - 認証強化の手順

◘ 手順のサマリ

OTP認証の
インスタンス

インスタンスを
組み合わせる

モジュールインスタンスの作成

↓

認証連鎖の設定

↓

**Let’s Try - OTP認証で
ownCloudへログイン**

認証強化の設定 – モジュールインスタンスの作成

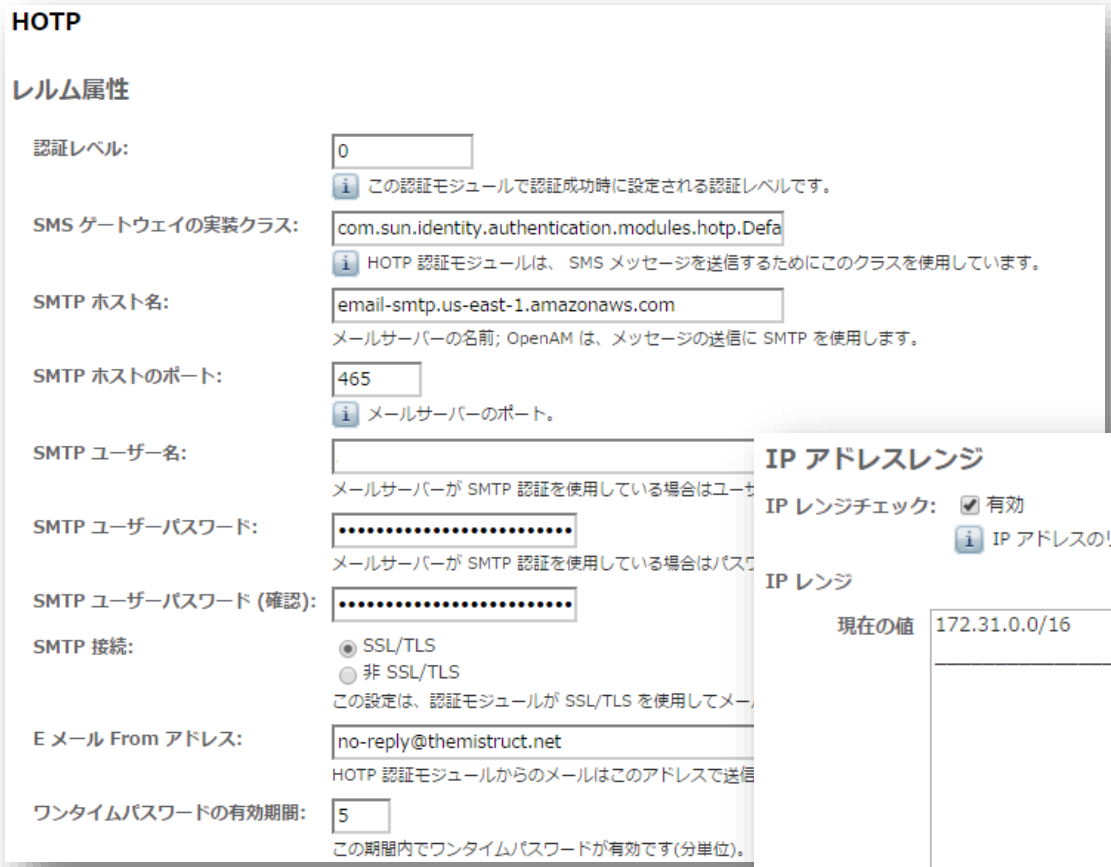
- ID/PW認証（インスタンス名：DataStore）
  - デフォルトで設定済みなので、割愛

- リスクベース認証（インスタンス名：RiskBased）
  - ソースIPが172.31.0.0/16以外なら認証失敗
  - 他にも、CookieやHeader、最終ログインからの経過時間、ユーザーの属性（所属や役職）で制御可能。

- HOTP認証（インスタンス名：HOTP）
  - OTPをユーザーのmail属性に設定されたメールアドレスに送信する
  - 他にもSMSでの送信が可能。

認証強化の設定 – モジュールインスタンスの

設定画面



▲HOTPの設定画面

▼RiskBasedの設定画面

本日のデモ - 認証強化の手順

◘ 手順のサマリ

OTP認証の
インスタンス

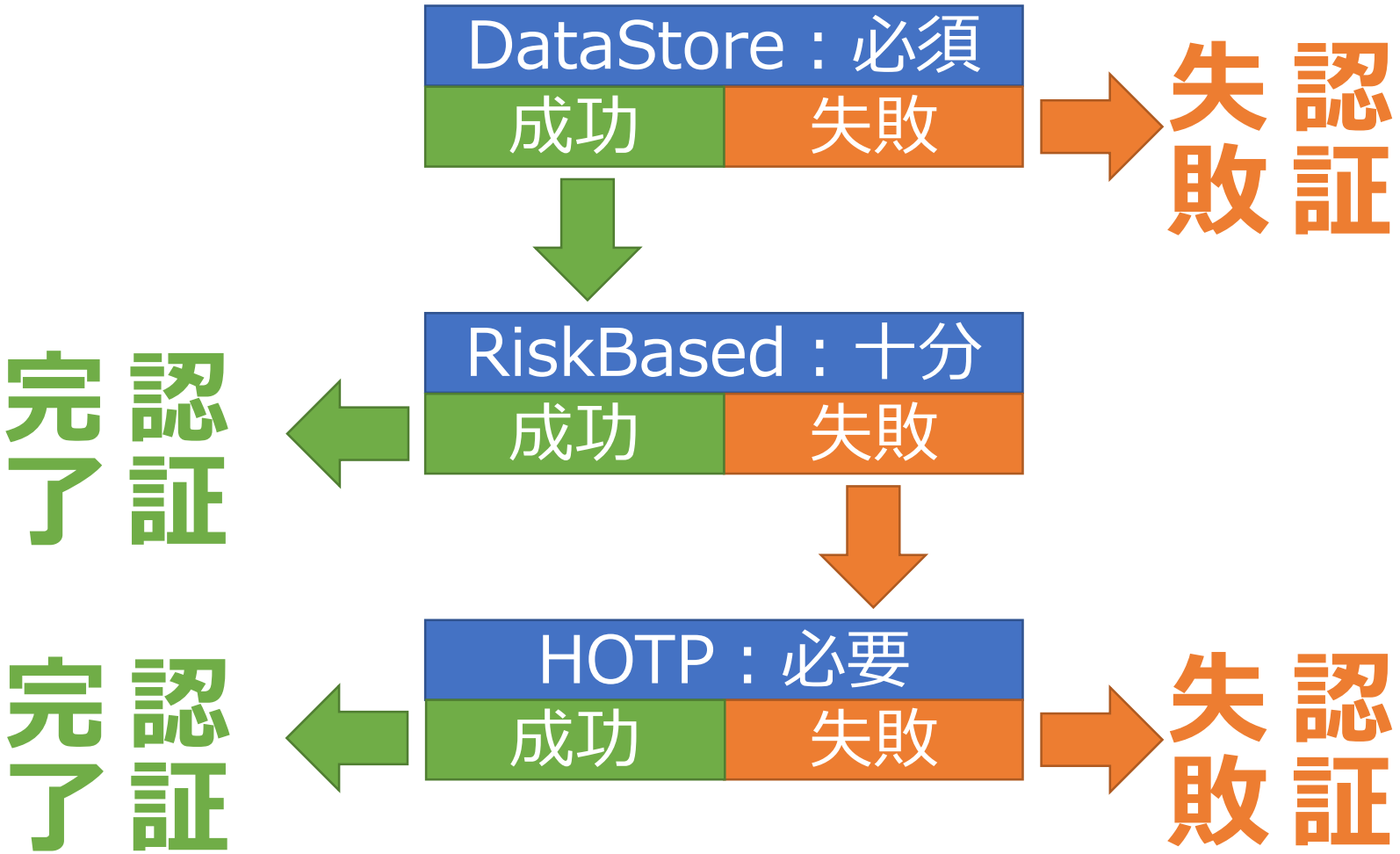
モジュールインスタンスの作成

インスタンスを
組み合わせる

認証連鎖の設定

Let’s Try - OTP認証で
ownCloudへログイン

認証強化の設定 – 認証連鎖

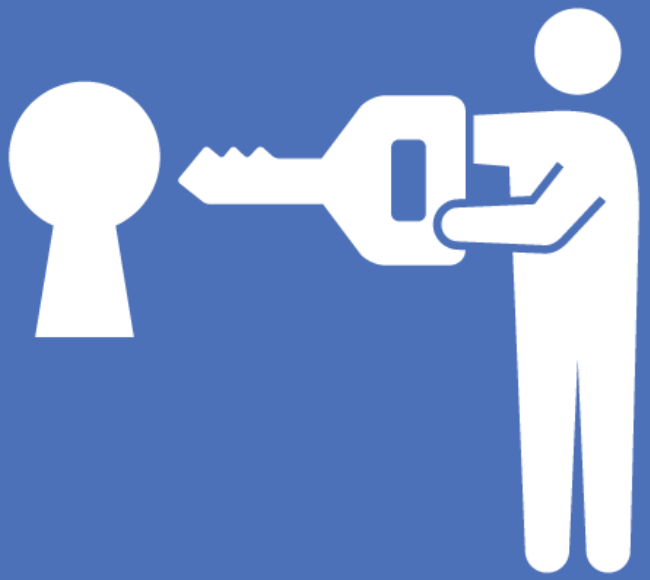


認証強化の設定 – 認証チェーン

設定画面

orgAuthN - プロパティー

必要：成功したら連鎖から抜ける。
必須：成功したら次の連鎖に進める
十分：成功したら連鎖から抜ける。
　　失敗したら次の連鎖に進める。



Let’s try ②
~ownCloudでMFA~

番外編：ownCloudのanonymous userの対応

- 認証除外URL
  - OpenAMでの認証を必要とせず、アクセスできるURLをホワイトリストで設定できる

- 設定方法
  1. 管理者でOpenAMにログイン
  2. アクセス制御＞最上位のレルム＞エージェント＞エージェント名＞アプリケーション＞適用されないURL 処理
  3. 適用されないURLに対象のURLを追加
  4. http://owncloud.ossxusers.themistruct.net/owncloud/index.php/s/*

まとめ

まとめ

ownCloudは「簡単に」、「既存のリソースを使って」、
「やりたいこと」ができる

ownCloudをインターネットに公開する際は、
認証強化をすること

ownCloudとOpenAMと連携して、
認証強化を実装することができる

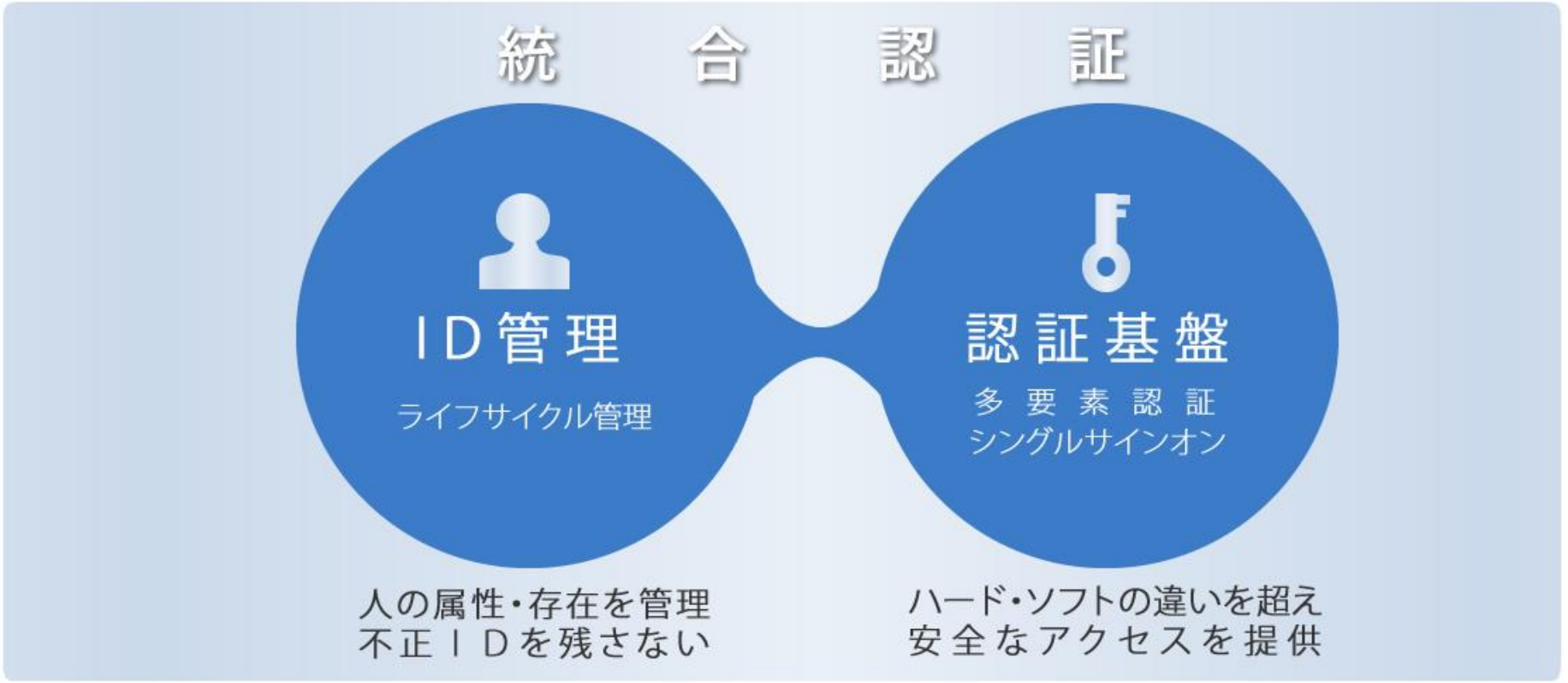
OpenAMでは様々な認証手段を提供しており、
設定だけで使うことができる

認証を数珠つなぎにする「認証連鎖」を駆使して、
企業のポリシーにあった、認証をユーザに提供できる

OpenAMは昔に比べて、だいぶ小慣れてきた！

当社
ソリューションの
ご紹介

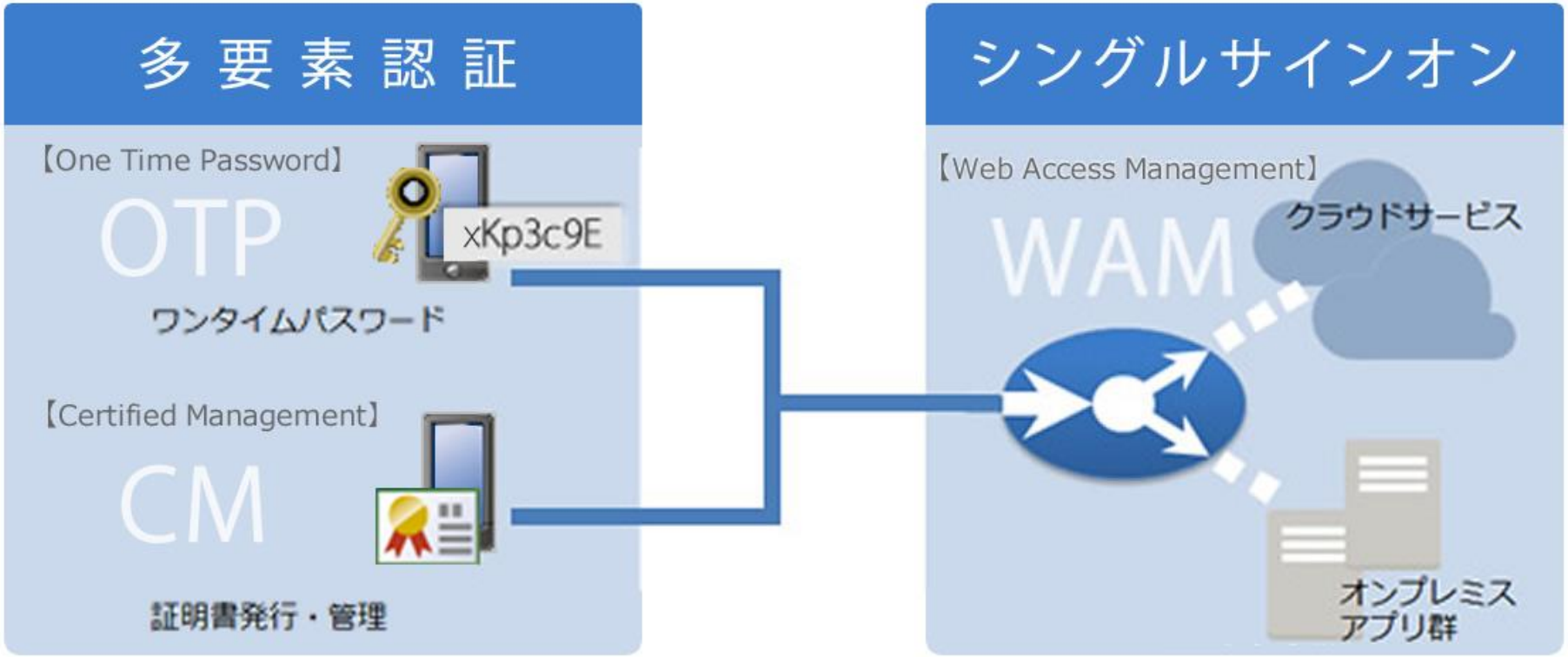
ThemiStruct は統合認証ソリューション



多要素認証やシングルサインオンなどアクセスマネジメント

1. 認証基盤ソリューション

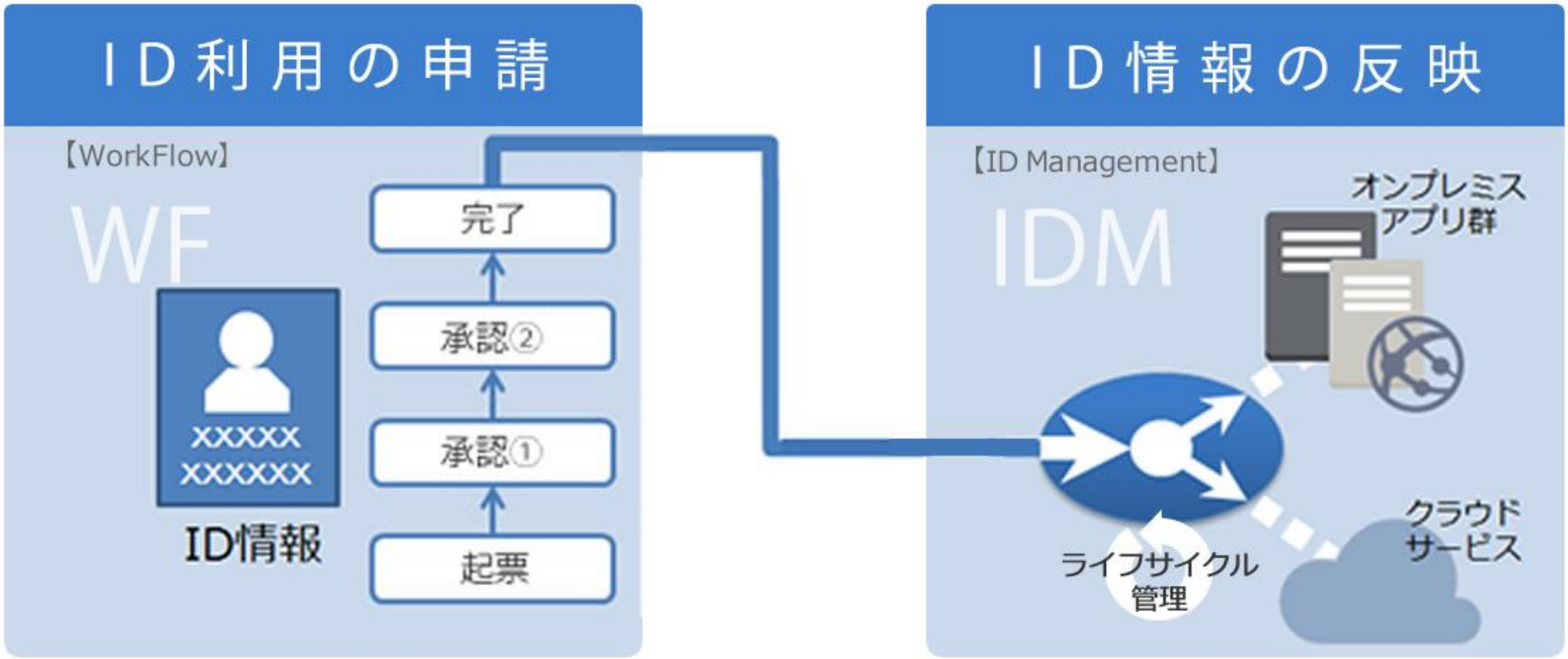
クラウド×スマートデバイスを多要素認証でセキュリティ強化し
シングルサインオンで利便性を向上します



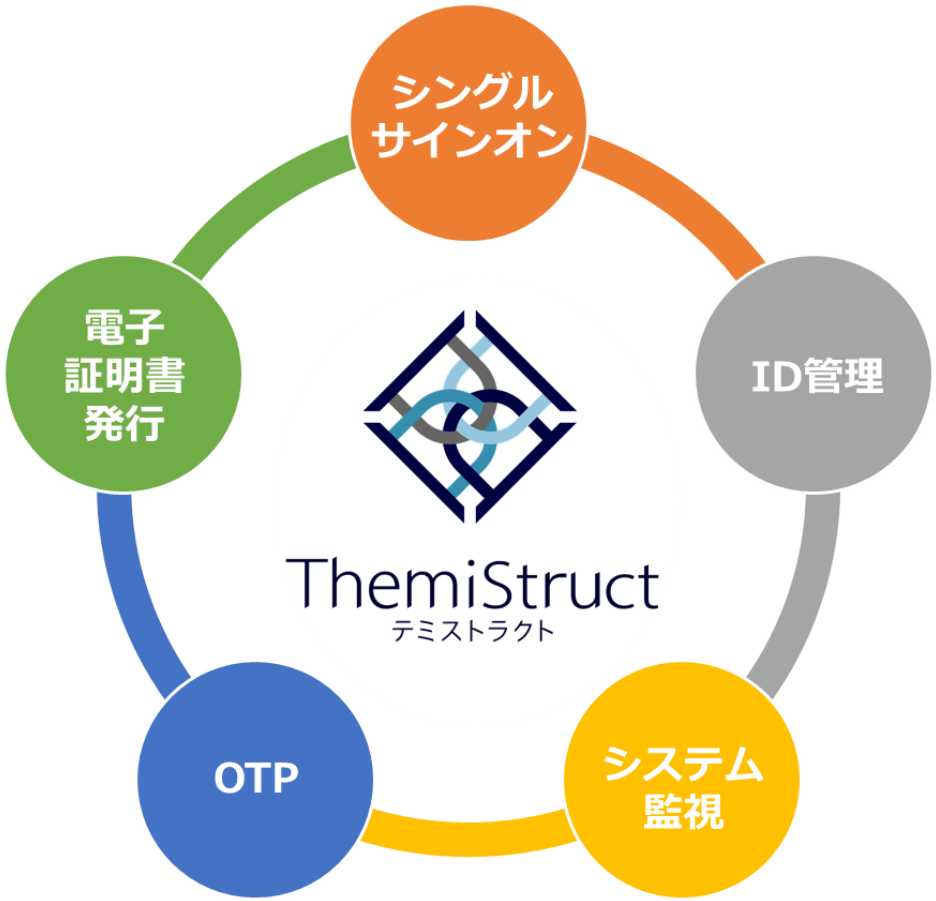
IDのライフサイクル管理

2. ID管理ソリューション

ID情報を申請フローを用いて収集し対象システム群へ反映
有効期限による管理および人事異動などにスマートに対応可能です



ThemiStructとは？



ThemiStruct(テミストラクト)は、オージス総研が持つOSS活用のノウハウと導入実績に裏打ちされた統合認証ソリューションです。

テクニカルサポート

- 利用方法などの問合せへの回答
- 障害時の調査、回避策や代替案の提示、復旧の技術支援

プロフェッショナルサービス

- 要件実現方法の相談、回答
- 概念実証、技術検証の支援
- 自社で実施する開発、構築の技術支援

システムインテグレーションサービス

- お客様の要望に応じたシステムの実現を責任を持ってお引受け

ThemiStruct-WAM

テミストラクト

- OpenAM (trunk) がベース
- 専用のインストーラー（Linux、Windows Serverに対応）
- 当社オリジナルの日本語マニュアル
- スマートデバイスに対応するレスポンシブUI
- パスワード管理強化のためのアドオンモジュール群
- 高度なリスクベース認証を実現するインベントリ認証オプション

ThemiStruct-IDM

テミストラクト

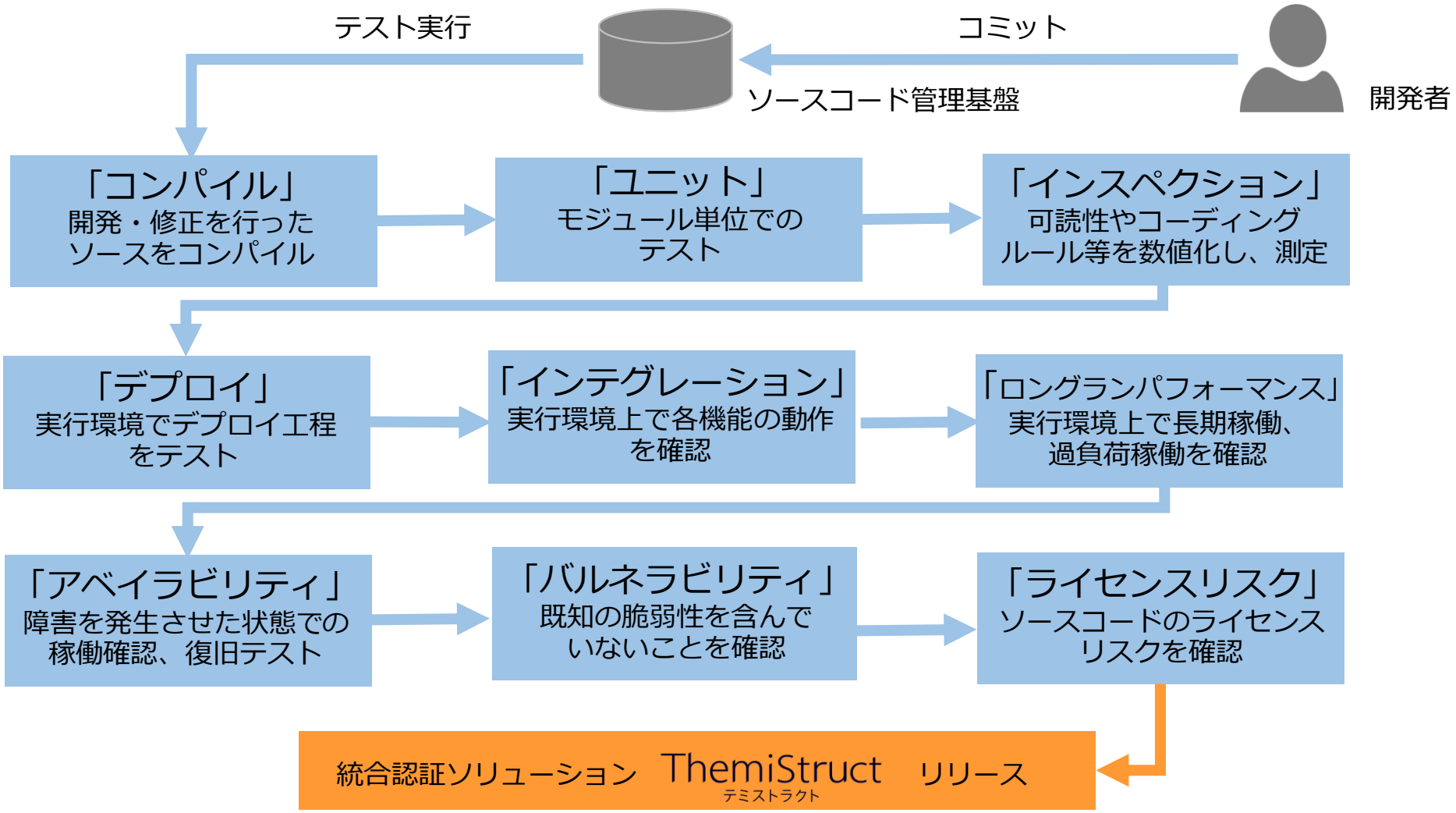
- OpenIDM (trunk) がベース
- 専用のインストーラー
- 当社オリジナルの日本語マニュアル
- 権限委譲に対応した専用のWebUI (SSI)
- 組織、グループ、ロールの管理
- プロビジョニングのスケジューリング、予約への対応
- …

ThemiStruct-CM

テミストラクト

- EJBCA (Enterprise版) がベース
- 当社オリジナルの日本語マニュアル
- 証明書の一括登録、一括ダウンロード
- 秘密鍵がエクスポートできない証明書発行への対応
- デバイスアクセスコントロールへの対応
- …

OSSを安心して活用いただくための取り組み



Amazon Web Services上に展開されたテスト基盤

- テストターゲットの自動デプロイによるクリーンなテスト環境の構築
- 自動インテグレーションテスト
- スポットインスタンスを活用した大規模なパフォーマンステスト
- etc…

当社は APN (AWS Partner Network)
コンサルティングパートナー です。

ライセンスリスクの確認

- 使用しているOSS、著作権者、ライセンス条件を正しく把握
- 権利の瑕疵の発生を予防
- OSS自動検出ツール「Palamida™」 を使用

当社は 米国PALAMIDA社の
パートナー です。

サポート力向上のための更なる挑戦

- ソースコード静的解析ツール「COVERITY®」を使った、不具合発見と修正へのトライに着手。
  - クラッシュ
  - リソースリーク
  - 脆弱性
  - …

当社は 米国シノプシス社（旧：コベリティ社）の
販売代理店 です。

ThemiStruct
テミストラクト

ご清聴ありがとうございました

OGIS-RI

【お問合せ先】
TEL: 03-6712-1201 / 06-6871-7998
E-mail: info@ogis-ri.co.jp