OpenAM 14 管理者マニュアル

オープンソース・ソリューション・テクノロジ (株)

更新日	2021 年 6 月 1 日
リビジョン	1.7

目次

1 はじめに

1.1 本書の目的

本文書は弊社提供の OpenAM 14 の管理者向けマニュアルです。

1.2 前提情報

本文書では、特に記載の無い場合は、以下の環境を利用していることを前提として記載します。適宜、実際にお使いの環境に読み替えてください。

OpenAM サーバ：

【属性】	【値】
ホスト名	sso.example.co.jp
ポート番号	TCP/8080
URI	http://sso.example.co.jp:8080/openam/

OpenLDAP サーバ：

【属性】	【値】
ホスト名	ldap.example.co.jp
ポート番号	TCP/389 (LDAP)、TCP/636 (LDAPS)
URI	ldap://ldap.example.co.jp/ (LDAP) ldaps://ldap.example.co.jp/ (LDAPS)

1.3 OpenAM インストールパス

本文書では、OpenAM がインストールされたパスを"`{OPENAM_INSTALL}`"と表記します。特別に設定していなければ、パスは「`/opt/osstech/share/tomcat/webapps/openam`」になります。

2 OpenAM の管理

本章では、OpenAM を管理する上での基本事項を記載します。

2.1 管理者アカウント

OpenAM における管理用アカウントは、以下の 2 種類存在します。本文書において「管理者」と記されていた場合、通常は 1. の amadmin を指します。

1. OpenAM の管理者アカウント（amadmin）
   OpenAM 自体の管理者アカウントです。
2. OpenDJ の管理者アカウント（cn=Directory Manager）
   OpenAM の内部で動作する OpenDJ（設定保存用のディレクトリサーバー）の管理者アカウントです。

2.2 管理者コンソールへのログイン

OpenAM の機能は、Web インタフェースで管理・設定が可能です。管理者向けの Web インタフェースを「管理コンソール」と呼びます。管理コンソールへのログイン方法について説明します。

1. ブラウザで OpenAM にアクセスします。
   http://sso.example.co.jp:8080/openam/
2. 管理者アカウントである「amadmin」でログインします。

以降の説明の中で「OpenAM に管理者ユーザーでログインする」、又は「管理コンソールを開く」といった表記があった場合は、上記の手順で管理者アカウントでログインし、管理コンソールにアクセスした状態であることを意味します。

2.3 管理者コンソールのトップ画面 (レルム選択画面) への戻り方

OpenAM に管理者ユーザーでログインすると、トップ画面（レルム選択画面）が表示されます。OpenAM 14 の Web 画面では、XUI と呼ばれる新しい画面系と、旧来の画面系とが混在しており、それぞれ以下の方法でトップ画面に戻ることができます。

- 新しい画面系から戻る場合（以下のいずれか）
  - 画面左上の OpenAM ロゴをクリックする
  - 画面上部のメニューで「レルム」→「全て表示」をクリックする
- 旧来の画面系から戻る場合
  - 「アクセス制御」タブや「アクセス制御へ戻る」ボタンをクリックする
    （アクセス制御に関するタブやボタンが画面上に無い場合は、それが出てくるまで「〇〇へ戻る」ボタンを押す。）

以後、「管理者コンソールのトップ画面（に戻る）」や「レルム選択画面（に戻る）」といった表記があった場合は、上記の手順を実施してください。

2.4 管理コンソールへのログインパスワード変更

管理コンソールへのログインパスワードの変更方法は、以下の通りです。

1. OpenAM に管理者ユーザーでログインする。
2. 「Top Level Realm（/）」→「対象」→「amAdmin」とクリックしていきます。
3. amAdmin の個人プロファイルを設定する画面において、「パスワード」の「編集」をクリックします。
4. 「新しいパスワード」と「パスワードの再入力」に新しいパスワードを入力し、「了解」ボタンをクリックします。

以上で完了です。冗長化構成を採用している場合、上記の作業を 1 台のサーバ上で実行すれば、全てのサーバーにおける amadmin のパスワードが変更されます。

2.5 OpenDJ への接続パスワード変更

OpenAM の初期設定により amadmin と「cn=Directory Manager」には同じパスワードが設定されますが、管理コンソール上で amadmin のパスワードを変更しても「cn=Directory Manager」の方のパスワードは 変更されません。「cn=Directory Manager」のパスワードを変更しなくても機能上は問題ありませんが、本節ではこの変更方法を記載します。

「cn=Directory Manager」のパスワードは以下の手順で実施します。詳細な内容は、本節内の各項にて記載します。

1. OpenDJ に「cn=Directory Manager」のパスワードを追加する
2. OpenAM のディレクトリパスワードを変更する

3. OpenDJ から「cn=Directory Manager」の旧パスワードを削除する
4. （冗長化構成を採用している場合）OpenDJ のレプリケーション管理者アカウント「cn=admin,cn=Administrators,cn=admin data」のパスワードを変更する。

なお、冗長化構成を採用している場合、全てのサーバーを同時に停止する必要はありません。ただし、各手順を全てのサーバーに適用した後、次の手順に移ってください。

＜注意＞　設定データストアである OpenDJ は OpenAM の動作に必要な設定情報が保存されています。Directory Manager アカウントのパスワード変更が不完全の場合は OpenAM が正常に動作しなくなりますので、実施する際には必ずバックアップを取得して慎重に操作を行ってください。

2.5.1 OpenDJ に「cn=Directory Manager」のパスワードを追加する

1. OpenAM(Tomcat) を停止します。

```
# /sbin/service osstech-tomcat stop
```

2. 現在の OpenAM の設定をバックアップしておきます。

```
# cd /opt/osstech/var/lib/tomcat/data
# cp -a openam openam.bak
```

3. OpenDJ に付属のコマンドを使い、新しいパスワードのハッシュ値を取得します。なお、このコマンドは Tomcat の実行ユーザーである “tomcat” で実行する必要があります。新しいパスワードは “-c” オプションでコマンドライン引数として渡す必要があります。コマンド履歴上にパスワードを残したくない場合は、コマンドをシェルスクリプトに記述して実行してください。

```
# su tomcat -s /bin/bash
$ /opt/osstech/var/lib/tomcat/data/openam/opends/bin/encode-password -s SSHA512
-c "newpasswd"

Encoded Password:  "{SSHA512}jrZ9VxJKj1XXICWU1E0KKmPXc9YhoM14tEqBx5TeFe+Pm3
KLGiJ7/4N0/CxSh0c/D1h2DO#Bbl0h34KF95cTpmMwQWsVWfnpR"
```

4. OpenDJ 用の LDIF ファイルを編集します。

/opt/osstech/var/lib/tomcat/data/openam/opends/config/config.ldif ファイルを開き、「dn: cn=Directory Manager,cn=Root DNs,cn=config」エントリに userpassword 属性を追加し、新しいパスワードのハッシュ値を記述します。一時的にですが、新旧パスワードに対応する userpassword 属性が 2 つ存在することになります。

```
dn: cn=Directory Manager,cn=Root DNs,cn=config
objectClass: organizationalPerson
objectClass: top
objectClass: person
objectClass: inetOrgPerson
objectClass: ds-cfg-root-dn-user
ds-cfg-alternate-bind-dn: cn=Directory Manager
cn: Directory Manager
sn: Manager
givenName: Directory
userpassword: {SSHA512}UE2gem0He9SNmlfxDhyzl5wkWbM1gagn4OYaeAdD604qMakE70h
Nm3y+3oogQulZEUFvz4cR2AF8ap88uXDAj+SaJKbevNAi
userpassword: {SSHA512}jrZ9VxJKj1XXICWU1E0KKmPXc9YhoM14tEqBx5TeFe+Pm3KLGiJ
7/4N0/CxSh0c/D1h2DOBbl0h34KF95cTpmMwQWsVWfnpR
```

5. OpenAM(Tomcat) を起動します。

```
# /sbin/service osstech-tomcat start
```

6. 冗長化構成を採用している場合、同様の作業を全てのサーバーに対して実施します。

2.5.2 OpenAM のディレクトリパスワードを変更する

OpenAM に保存されている OpenDJ 接続のためのパスワードを変更します。

1. OpenAM 管理コンソールを開き、「デプロイメント」→「サーバー」→対象サーバー→「ディレクトリ設定」と選択します。「バインドパスワード」を新しいパスワードに変更し、保存します。(冗長化構成を採用している場合、全てのサーバーの設定を変更します)
2. OpenAM 管理コンソールのトップ画面に戻り、「/ (Top Level Realm)」 →「データストア」→「embedded」と選択します。「LDAP バインドパスワード」と「LDAP バインドパスワード (確認)」を新しいパスワードに変更し、保存します。
3. OpenAM 管理コンソールのトップ画面に戻り、「/ (Top Level Realm)」 →「サービス」

→「ポリシー設定」を開きます。「LDAP バインドパスワード」と「LDAP バインドパスワード (確認)」を新しいパスワードに変更し、保存します。

2.5.3 OpenDJ から「cn=Directory Manager」の旧パスワードを削除する

1. OpenAM(Tomcat) を停止します。

```
# /sbin/service osstech-tomcat stop
```

2. OpenDJ 用の LDIF ファイルを編集します

   /opt/osstech/var/lib/tomcat/data/openam/opends/config/config.ldif に記述されている古いパスワードを削除します。config.ldif ファイルにある「dn: cn=Directory Manager,cn=Root DNs,cn=config」エントリの userpassword 属性のうち、古いパスワードの属性を削除します。結果的に userpassword 属性は新パスワードに対応するものだけが存在することになります。

```
dn: cn=Directory Manager,cn=Root DNs,cn=config
objectClass: organizationalPerson
objectClass: top
objectClass: person
objectClass: inetOrgPerson
objectClass: ds-cfg-root-dn-user
ds-cfg-alternate-bind-dn: cn=Directory Manager
cn: Directory Manager
sn: Manager
givenName: Directory
userpassword: {SSHA512}jrZ9VxJKj1XXICWU1E0KKmPXc9YhoM14tEqBx5TeFe+Pm3KLGiJ
7/4N0/CxSh0c/D1h2DOBbl0h34KF95cTpmMwQWsVWfnpR
```

3. OpenAM を起動します。

```
# /sbin/service osstech-tomcat start
```

4. OpenAM 起動後にログイン画面が表示され、amadmin でログイン可能であり、各種設定が正常に読み込めているか確認します。
5. 冗長化構成を採用している場合、同様の作業を全てのサーバーに対して実施します。

2.5.4 cn=admin,cn=Administrators,cn=admin data のパスワード変更

冗長化構成を採用している場合、レプリケーション用のアカウントのパスワードは cn=Directory Manager と合わせておく必要があります。冗長化構成を採用していない場合は、本手順を実施する必要はありません。

下記のコマンドを実行し、cn=admin,cn=Administrators,cn=admin data のパスワードを変更します。

```
# /opt/osstech/var/lib/tomcat/data/openam/opends/bin/ldappasswordmodify \
  --port 50389 \
  --hostname localhost \
  --bindDN "cn=Directory Manager" \
  --bindPassword "[cn=Directory Manager のパスワード]" \
  --authzID "cn=admin,cn=Administrators,cn=admin data" \
  --newPassword "[cn=Directory Manager のパスワード]"
```

以上で完了です。

3 ユーザー管理

本章では、OpenAM におけるユーザー管理機能の基本的な事項について説明します。

3.1 外部ディレクトリサーバとの連携

OpenAM はシングルサインオンのためのソフトウェアであり、その機能の中心は認証・認可にあります。OpenAM のユーザー管理機能はシステムを構成するための十分な機能を有しておりません。そのため、ユーザー管理については、OpenLDAP などの外部ディレクトリサーバをご利用ください。

また、OpenAM のユーザー情報は、基本的に LDAP で管理されていることを前提としています。MySQL などの RDBMS を利用することも可能ですが、グループの機能を利用できないことや、複数の値をもつ属性を利用できないなど、機能の一部に制限があります。

3.2 ユーザー

OSSTech 版 OpenLDAP をユーザー情報の保存先（ユーザーデータストア）に利用する場合は、以下のオブジェクトクラスを利用します。その他の LDAP ソフトウェアを利用する場合は、その仕様や設計に合わせて適宜変更してください。

- top
- person
- inetOrgPerson
- organizationalPerson
- inetUser
- iplanet-am-user-service
- iplanet-am-session-service
- iplanet-am-managed-person
- iPlanetPreferences
- sunAMAuthAccountLockout
- sunFMSAML2NameIdentifier

4 ユーザー認証

本章では、OpenAM のユーザー認証機能の概要について説明します。

4.1 認証モジュールとユーザーデータストア

OpenAM のユーザー認証機能では、主に以下の 2 種類のモジュールが関係します。

1. 認証モジュール
   ユーザー認証を行うモジュール
2. ユーザーデータストア
   認証モジュールによって識別されたユーザーの情報を扱うモジュール

認証モジュールとユーザーデータストアとで別のディレクトリサーバーを利用する構成・設定も可能ですが、通常は、同一のディレクトリサーバーを利用します。

4.2 認証手法

OpenAM で実現できる認証手法は、大別すると以下の各項のようになります。

4.2.1 ログイン URL

下記の OpenAM ログイン URL にブラウザでアクセスし、OpenAM のログイン画面から認証します。

```
http://sso.example.co.jp:8080/openam/
```

認証モジュールとして Windows デスクトップ SSO などを設定している場合など、設定によっては画面が表示されない場合もあります。

4.2.2 フェデレーション

OpenAM が他のアイデンティティプロバイダーと連携している場合、アイデンティティプロバイダーから渡された認証情報を元にログインを行います。たとえば、OpenAM が SAML 2.0 のサービスプロバイダーとして動作している場合、アイデンティティプロバイダーから渡されるアサーションによって認証を行います。

4.2.3 REST API

REST API でログインすることも可能です。”X-OpenAM-Username” ヘッダーにユーザー名、”X-OpenAM-Password” ヘッダーにパスワードを指定して REST API のエンドポイントへ POST リクエストを送信します。 以下に例を示します。

```
$ curl \
 --request POST \
 --header "X-OpenAM-Username: demo" \
 --header "X-OpenAM-Password: changeit" \
 --header "Content-Type: application/json" \
 --data "{}" \
 http://sso.example.co.jp:8080/openam/json/authenticate

{ "tokenId": "AQI...*", "successUrl": "/openam/console" }
```

認証に成功すると、JSON 形式で認証トークンが返却されます。たとえば、この値を OpenAM の認証用クッキー (iPlanetDirectoryPro) に設定することで、クッキーが存在するブラウザは OpenAM に認証済みとして扱われます。

5 レルム設定

本章では、OpenAM のレルムについて説明します。

5.1 レルムとは

「レルム」とは OpenAM の設定をまとめた単位であり、組織毎に認証を分ける場合などに利用します。レルム毎に以下の項目を個別に設定することが可能です。

- 認証モジュール
  詳細は「7 認証モジュール設定」の章を参照
- ユーザーデータストア
  詳細は「6 ユーザーデータストア設定」の章を参照
- DNS ドメイン (OpenAM の DNS エイリアス)
  詳細は「5.3.1 ドメイン名で指定する」の項を参照
- ポリシー
  詳細は別紙「Policy Agent リファレンスマニュアル」を参照
- SAML
  詳細は別紙「OpenAM SAML 設定ガイド」を参照
- ログイン画面、ログアウト画面、各種エラー画面のデザイン (JSP、CSS、画像ファイルなど)

OpenAM インストール時に作成されるデフォルトのレルムをトップレルム（Top Level Realm）と呼び、トップレルム以外のレルムをサブレルムと呼びます。たとえば以下の目的でサブレルムを定義します。

- OpenAM 管理者用のレルムと一般ユーザー用のレルムを分ける
  最も一般的なレルムの利用方法です。トップレルムを管理者用のレルムとして利用し、一般ユーザー用にサブレルムを作成します。
- 顧客毎にレルムを分ける
  複数の顧客 (企業) に対して OpenAM のサービスを提供する場合（マルチテナント）に、各顧客 (企業) にサブレルムを割り当てる利用方法です。
- 子会社/部門/学部などをもとにしてレルムを分割する
  一つの組織 (企業や大学) 内で OpenAM のサービスを提供するにあたり、部分組織毎にサブレルムを割り当てて管理を分散させたい場合の利用方法です。

以下の各節では、サブレルムの作成方法とサブレルムへのアクセス方法について説明します。

5.2 サブレルムの作成

サブレルムの作成方法を説明します。

1. OpenAM に管理者ユーザーでログインします。
2.「新規レルム」をクリックします。
3.「名前」にレルム名を入力します。ここでは「usr」と入力したとします。
4.「レルムまたは DNS のエイリアス」に、このレルムに割り当てるドメイン名を追加します。例えば「sub.example.co.jp」などです。ドメイン名を割り当てない場合は、この手順は実施する必要はありません。
5.「レルムの状態」にチェックが入っていることを確認します。
6.「作成」をクリックします。

以上で完了です。

5.3 サブレルムへのアクセス

OpenAM を利用する一般ユーザーは、自身のアカウントが登録されているユーザーデータストアが割り当たっているレルムを指定してログインする必要があります。レルムを指定する方法として、以下の 2 通りの方法があります。

1. ドメイン名で指定する
2. URL のクエリ文字列で指定する

以下、それぞれの方法について解説します。

5.3.1 ドメイン名で指定する

OpenAM のレルムには、そのレルム専用のドメイン名を割り当てることが可能です（Apache HTTP Server のバーチャルホストのような機能です）。以下のように、レルム毎に異なるホスト名を指定してブラウザからアクセスします。

- レルム A（例えば、企業 A のレルム）
  - http://com-a.example.co.jp:8080/openam/
- レルム B（例えば、企業 B のレルム）
  - http://com-b.example.jp:8080/openam/

ドメイン名を指定してアクセスする場合は、各レルムの設定にドメイン名を追加する必要があります。サブレルムの作成のようにレルム作成時でも設定できますが、レルム作成後に設定する場合の手順は以下のようになります。

1. OpenAM に管理者ユーザーでログインします。
2. 対象レルムをクリックし、「プロパティ」ボタンを押します。
3. 「レルムまたは DNS のエイリアス」欄に、このレルムに割り当てるドメイン名を追加します。
4. 「保存」をクリックします。

OpenAM は、クライアントから送られる HTTP ヘッダーの Host ヘッダーの値を見てレルムを判別します。そのため、ドメイン名でレルムを指定する場合は、アクセスしたいレルムのドメイン名を含む URL で OpenAM にアクセスする必要があります。
ただし、後述する「5.3.2 URL のクエリ文字列で指定する」の方法でレルムを指定する場合は、「レルムまたは DNS のエイリアス」へのドメイン名の追加は不要です。

レルムへのドメイン名の割り当ての説明は以上ですが、さらにクッキードメインの確認・設定や、トップレルムへのリダイレクトの回避などの設定が必要になります。詳細は 10. マルチテナントを参照してください。

5.3.2 URL のクエリ文字列で指定する

レルムを指定するにあたり、OpenAM の URL に特定のクエリ文字列を指定することで行うことも可能です。以下に例を示します。

- レルム A（例えば、企業 A のレルム）
  - http://sso.example.co.jp:8080/openam?realm=＜レルム A のレルム名＞
- レルム B（例えば、企業 B のレルム）
  - http://sso.example.co.jp:8080/openam?realm=＜レルム B のレルム名＞

6 ユーザーデータストア設定

本章では、OpenAM のユーザーデータストアの設定方法について説明します。

ユーザーデータストアとは、OpenAM のユーザー情報を保存・参照するためのデータベースです。OpenAM はユーザーデータストアとして OpenLDAP 等の外部のディレクトリサーバーを利用します。ユーザーデータストアは、OpenAM の初期設定の完了後に、必要に応じて追加することが出来ます。

6.1 OpenLDAP ユーザーデータストア

OpenLDAP を OpenAM のユーザーデーターストアとして利用するための設定手順を説明します。

1. OpenAM に管理者ユーザーでログインします。
2. 対象のレルムをクリックし、「データストア」をクリックします。
3. 選択したレルムが「/ (Top Level Realm)」以外のレルムの場合は、「データストア」の「embedded」をチェックし、「削除」をクリックします。
4. 「データストア」の「新規」をクリックします
5. 「名前」欄に任意の名前を入力し、「タイプ」の「OpenLDAP」をクリックして、「次へ」をクリックします。
6. LDAP サーバーの設定画面が表示されるため、パラメータを設定します。
   以下はパラメータの設定例です。

=== サーバー設定 セクション ===

【項目名】	【説明】
LDAP サーバー	LDAP サーバーのホスト名とポート番号を「: (コロン)」で区切って入力します。デフォルトで入力されている値は削除します。 例 : ldap.example.co.jp:389
LDAP バインド DN	LDAP サーバーへ接続するための管理者 DN を入力します。 例 : cn=oam,dc=example,dc=co,dc=jp
LDAP バインドパスワード	LDAP サーバーへ接続する管理者 DN のパスワードを入力します。

【項目名】	【説明】
LDAP バインドパスワード (確認)	同上
LDAP 組織 DN	LDAP の root suffix を入力します。 例 : dc=example,dc=co,dc=jp
LDAP Connection Mode	LDAP サーバーへ暗号化通信で接続する場合は、LDAPS や StartTLS を選択します。平文で通信する場合は LDAP を選択します。
LDAPv3 プラグイン検索範囲	LDAP の検索範囲を指定します。一般的には「SCOPE_SUB」を選択します。

=== ユーザー設定 セクション ===

【項目名】	【説明】
LDAP ピープルコンテナネーミング属性	ユーザーエントリが保存されている OU などを指定する場合のみ入力します。例えば、ユーザーエントリが ou=Users,dc=example,dc=co,dc=jp 以下に登録されている場合は、「ou=Users」の「ou」のみをここに入力します。 例 : ou
LDAP ピープルコンテナ値	ユーザーエントリが保存されている OU などを指定する場合のみ入力します。例えば、ユーザーエントリが ou=Users,dc=example,dc=co,dc=jp 以下に登録されている場合は、「ou=Users」の「Users」のみをここに入力します。 例 : Users

=== グループ設定 セクション ===

（グループ設定セクションは、OpenAM から LDAP のグループを参照する場合のみ入力します。）

【項目名】	【説明】
LDAP グループコンテナネーミング属性	グループエントリが ou=Groups,dc=example,dc=co,dc=jp 以下に登録されている場合は、「ou=Groups」の「ou」のみをここに入力します。 例 : ou
LDAP グループコンテナ値	グループエントリが ou=Groups,dc=example,dc=co,dc=jp 以下に登録されている場合は、「ou=Groups」の「Groups」のみをここに入力します。 例 : Groups
グループメンバーシップの属性名	ユーザーエントリに保存される、所属グループの DN を保持する属性名を入力します。 例 : memberOf
一意のメンバーの属性名	グループエントリに保存される、所属メンバーの DN を保持する属性名を入力します。 例 : uniqueMember
デフォルトグループメンバーのユーザー DN	OpenAM から LDAP グループを追加する場合、グループには必ず一人以上のメンバーが含まれている必要があります。OpenAM はグループ作成時に、このパラメーターで指定したメンバーをグループメンバーとして追加します。 例: cn=dummy,dc=example,dc=co,dc=jp

7. 画面右上の「終了」ボタンをクリックします。
8. OpenAM 管理コンソールのトップに戻る→ 対象のレルム →「対象」を開き、OpenLDAP に登録されているユーザー/グループが表示されることを確認します。表示されない場合は、設定に誤りがあります。

以上で完了です。

6.2 Active Directory ユーザーデータストア

Active Directory を OpenAM のユーザーデーターストアとして利用するための設定手順を説明します。

1. OpenAM に管理者ユーザーでログインします。

2. 対象のレルムをクリックし、「データストア」をクリックします。
3. 選択したレルムが「/ (Top Level Realm)」以外のレルムの場合は、「データストア」の「embedded」をチェックし、「削除」をクリックします。
4. 「データストア」の「新規」をクリックします
5. 「名前」欄に任意の名前を入力し、「タイプ」の「Active Directory」をチェックして、「次へ」をクリックします。
6. Active Directory サーバーの設定画面が表示されるため、以下の項目を入力します。追加で記入する項目、デフォルトの状態から変更する項目のみ記載します。

=== サーバー設定 セクション ===

【項目名】	【説明】
LDAP サーバー	Active Directory サーバーのホスト名とポート番号を「: (コロン)」で区切って入力します。デフォルトで入力されている値は削除します。
LDAP バインド DN	AD の管理者権限を持つユーザーの DN を入力します。 例：CN=Administrator,CN=Users,dc=ms,dc=osstech,dc=co,dc=jp
LDAP バインドパスワード	バインドユーザーのパスワードを入力します。
LDAP バインドパスワード (確認)	同上
LDAP 組織 DN	AD のベース DN を入力します。 例：ドメインが「ms.osstech.co.jp」の場合は「dc=ms,dc=osstech,dc=co,dc=jp」などのように入力します。
LDAP Connection Mode	AD サーバーへ暗号化通信で接続する場合は、LDAPS や StartTLS を選択します。平文で通信する場合は LDAP を選択します。 なお、暗号化通信する場合は、「7.4.1 Active Directory との LDAPS 通信設定」を実施する必要があります。
LDAPv3 プラグイン検索範囲	検索範囲を指定します。一般的には「SCOPE_SUB」を選択します。

7. 画面右上の「終了」ボタンをクリックします。
8. OpenAM 管理コンソールのトップに戻る→ 対象のレルム →「対象」を開き、Active Directory に登録されているユーザー/グループが表示されることを確認します。表示されない場合は、設定に誤りがあります。

以上で完了です。

6.3 その他の設定

6.3.1 ユーザー属性のキャッシュ時間の変更

OpenAM はユーザーデータストアから検索したユーザー属性情報をキャッシュしますが、デフォルトのキャッシュ時間は無制限に設定されています。そのため、ユーザーデータストア側でユーザー属性を変更しても OpenAM の動作には変化がなく、キャッシュ上の属性情報が使われ続けます。

ここでは、属性情報の変更が一定時間後には反映されるよう、キャッシュ時間を設定する手順を説明します (例として、有効期限を 10 分間に設定します)。

1. OpenAM に管理者ユーザーでログインします。
2.「設定」→「デフォルトサーバー」をクリックします。
3.「詳細設定」をクリックします。
4. 以下の 3 つの拡張プロパティーを追加します。

【プロパティー名】	【プロパティー値】
com.sun.identity.idm.cache.entry.expire.enabled	true
com.sun.identity.idm.cache.entry.default.expire.time	10 (キャッシュの有効期限は 10 分)
com.sun.identity.idm.cache.entry.user.expire.time	10 (キャッシュの有効期限は 10 分)

5.「変更の保存」ボタンをクリックします。
6. OpenAM を再起動します。

```
# /sbin/service osstech-tomcat restart
```

以上で完了です。

6.3.2 OpenAM データストアのユーザー動的生成

OpenAM では、認証モジュールとユーザーデータストアとで異なるディレクトリサーバーを参照することが可能です。この場合、認証したユーザーの情報がユーザーデータストアに存在しないケースが発生します。

OpenAM には、認証したユーザーがユーザーデータストアに存在しない場合に、ユーザーデータストアが参照しているディレクトリサーバーに自動でユーザー情報を登録する機能が用意されています。

ここでは、この機能を有効化する手順を説明します。

1. OpenAM に管理者ユーザーでログインします。
2. 対象のレルム →「認証」→「設定」ボタンをクリックします。
3.「ユーザープロファイル」→「ユーザープロファイル」と辿り、「動的」を選択します。
4.「変更の保存」ボタンをクリックします。

以上で完了です。

7 認証モジュール設定

本章では、OpenAM の認証モジュールの設定方法について説明します。

7.1 データストア認証モジュール

ユーザーデータストアに設定したディレクトリサーバーを利用して認証するモジュールです。デフォルトで利用する認証モジュールになります。

7.2 OpenLDAP 認証モジュール

OpenLDAP 認証モジュールは、OSSTech 版 OpenAM 独自の認証モジュールです。LDAP 認証モジュールをカスタマイズして、OpenLDAP への親和性を向上させています。設定方法は別紙「OpenLDAP 認証モジュール利用手順書」をご覧ください。

7.3 LDAP 認証モジュール

LDAP 認証モジュールの設定方法について説明します。

1. OpenAM 管理コンソールを開き、対象のレルム →「認証」→「モジュール」→「モジュールの追加」と辿ります。
2. 「モジュール名」に任意の名前を入力し、「タイプ」から「LDAP」を選択して、「作成」ボタンをクリックします。
3. 認証モジュールの設定画面にて各パラメーターを入力し、「変更の保存」をクリックします。

以下はパラメーターの例です。実際の値は LDAP サーバーの設計に合わせてください。

【設定項目】	【説明】
プライマリ LDAP サーバー	ldap.example.co.jp:389 (デフォルトで入力されている値は削除します)
ユーザー検索の開始 DN	ou=Users,dc=osstech,dc=co,dc=jp (デフォルトで入力されている値は削除します)
バインドユーザー DN	cn=oam,dc=osstech,dc=co,dc=jp
バインドユーザーパスワード	cn=oam のパスワードを入力

【設定項目】	【説明】
ユーザープロファイルの取得に使用する属性	uid
認証するユーザーの検索に使用する属性	uid
ユーザー検索フィルタ	(objectclass=inetorgperson)
検索範囲	サブツリー
LDAP 接続モード	LDAP (セキュアなプロトコルを使用する場合は、LDAPS または StartTLS を選択する。)
ユーザー DN をデータストアに返す	無効 (有効のチェックをはずす)

4. 左側のメニューより、「認証」→「認証連鎖」→「認証連鎖の追加」をクリックします。
5. 「認証連鎖名」に任意の名前を入力し、「作成」をクリックします。
6. 認証連鎖の設定画面が表示されますので、「モジュールの追加」をクリックします。
7. 「モジュールの選択」のプルダウンから先程作成した認証モジュールの名前を選択し、「基準の選択」は「Required」を選択します。
8. 「OK」をクリックし、認証連鎖の設定画面に戻ったら、「変更の保存」をクリックします。
9. 左側のメニューより、「認証」→「設定」→「コア」をクリックします。
10. 「コア」の「組織認証設定」で、先程作成した認証連鎖の名前を選択し、「変更の保存」をクリックします。

以上で完了です。

認証連鎖変更後の管理者コンソールへのログイン方法については「7.11.2 認証連鎖変更後の amadmin のログイン URL」を参照してください。

7.4 Active Directory 認証モジュール

Active Directory 認証モジュールの設定方法について説明します。

7.4.1 Active Directory との LDAPS 通信設定

まず、Active Directory サーバーと OpenAM 間で LDAPS(SSL) 通信を行なうための設定手順について説明します。LDAPS 通信を行なうためには、以下の設定作業を行なう必要があります。

1. Active Directory サーバーに Active Directory 証明書サービスをインストールする。
2. Active Directory サーバーの証明書をエクスポートする。
3. エクスポートした Active Directory サーバーの証明書を OpenAM サーバーにインポートする。

以降の節で、上記の設定作業の詳細な手順を説明します。

7.4.1.1 証明書サービスのインストール

Active Directory サーバーに Windows 証明書サービスをインストールします。以下の手順は Active Directory サーバー上で実施します。(ここでは、ネットワーク上に Windows Server 2016 の Active Directory が 1 台のみ存在することを想定し、一例として示すものです。実際には、Microsoft 社のマニュアルなどを参照し、システム要件に合致する内容で実施してください。)

1.「サーバーマネージャー」を起動して「役割と機能の追加」を選択し、「役割と機能の追加」ウィザードを開きます。
2.「開始する前に」画面、「インストールの種類」画面、「サーバの選択」画面では、「次へ」を押します。
3.「サーバーの役割の選択」画面に移り、ここでは「Active Directory 証明書サービス」を選択し、「機能の追加」を押します。
4.「サーバーの役割の選択」画面に戻るので、「次へ」を押します。
5.「機能の選択」画面、「Active Directory 証明書サービス」画面では、「次へ」を押します。
6.「役割サービスの選択」画面では、念の為「証明機関」にチェックが入っていることを確認し、「次へ」を押します。
7.「インストールオプションの確認」画面では「インストール」を押します。
8.「インストールの進行状況」画面でインストールが完了したら、「閉じる」を押します。
9.「サーバーマネージャー」の「通知」アイコンを押し、「対象サーバーに Active Directory 証明書サービスを構成する」を選択することで、「AD CS の構成」ウィザードを開きます。

10.「資格情報」画面では、「次へ」を押します。
11.「役割サービス」画面では、「証明機関」にチェックを入れ、「次へ」を押します。
12.「セットアップの種類」、「CA の種類」、「秘密キー」、「CA の暗号化」、「CA の名前」の各画面では、「次へ」を押します。
13.「有効期間」画面では、有効期間を必要に応じて変更し、「次へ」を押します。
14.「CA データベース」画面では、「次へ」を押します。
15.「確認」画面では「構成」を押し、「結果」画面で「閉じる」を押します。
16. Active Directory サーバーを再起動します。

7.4.1.2 証明書のエクスポート

Active Directory の証明書をエクスポートします。 以下の手順は Active Directory サーバー上で実施します。

1.「サーバーマネージャー」を起動して、「ツール」→「証明機関」を選択します。
2. 証明機関として表示されているサーバー名を右クリックし、「プロパティ」を開きます。
3. CA 証明書を選択し、「証明書の表示」を押します。
4.「詳細」タブを選択し、右下の「ファイルにコピー」をクリックします。
5.「エクスポートファイルの形式」画面で「Base64 encoded X509」を選択し、「次へ」をクリックします。
6. 出力ファイルを指定し、「次へ」をクリックします。
7.「完了」をクリックします 。
8. 証明書ファイルが作成されます。作成されたファイルを OpenAM サーバーにコピーしておきます。

7.4.1.3 OpenAM 側での証明書インポート

Active Directory サーバー側でエクスポートした証明書を、OpenAM サーバーの JRE(Java Runtime Environment) のキーストアにインポートします。作業は OpenAM サーバー上で実施します。

事前に Active Directory の証明書を OpenAM サーバー上にコピーしておきます (ここでは、sso.cert というファイル名であると仮定します)。

以下のコマンドを実行し、証明書をインポートします。

```
# cp sso.cert /etc/pki/ca-trust/source/anchors
# update-ca-trust extract
```

以上で証明書のインポートは完了です。OpenAM サーバーが稼働する Tomcat を再起動します。

```
# /sbin/service osstech-tomcat restart
```

以上で Active Directory へ LDAPS 接続するための事前準備は完了です。

7.4.2 Active Directory 認証設定

本章では、OpenAM へのログイン時の認証を、Active Directory に登録されているユーザー ID/パスワードで認証するための設定手順について説明します。

7.4.2.1 Active Directory 認証モジュール設定

Active Directory 認証モジュールを有効化する手順を説明します。以下の手順で設定します。

1. OpenAM に管理者ユーザーでログインします。
2. 対象レルム →「認証」→「モジュール」を開きます。
3.「モジュールの追加」ボタンをクリックします。
4.「名前」に任意のモジュール名を入力し、「タイプ」は「Active Directory」を選択して、「作成」ボタンをクリックします。
5. 各パラメーターを入力し、「変更の保存」をクリックします。
   以下はパラメーターの例です。実際の値は Active Directory の設計に合わせてください。

【パラメータ名】	【設定値】
プライマリ Active Directory サーバー	AD のホスト名 (FQDN):636 （デフォルトで登録されている「localhost:50389」は削除する）
セカンダリ Active Directory サーバー	セカンダリ Active Directory サーバーが存在する場合は入力する

【パラメータ名】	【設定値】
ユーザー検索の開始 DN	AD のベース DN を入力する 例：ドメインが「ms.osstech.co.jp」の場合は 「dc=ms,dc=osstech,dc=co,dc=jp」などのように入力する （デフォルトで登録されている DN は削除する）
バインドユーザー DN	AD の管理者権限を持つユーザーの DN を入力する 例： CN=Administrator,CN=Users,dc=ms,dc=osstech,dc=co,dc=jp
バインドユーザーパスワード	バインドユーザーのパスワードを入力する
ユーザープロファイルの取得に使用する属性	sAMAccountName （デフォルトで登録されている「uid」は削除する）
認証するユーザーの検索に使用する属性	sAMAccountName （デフォルトで登録されている「uid」は削除する）
検索範囲	サブツリー
LDAP Connection Mode	LDAPS
ユーザー DN をデータストアに返す	「有効」のチェックを外す

6. 左側メニューの「認証」→「認証連鎖」→「認証連鎖の追加」をクリックします。
7. 「新規認証連鎖」画面の「名前」に任意の認証連鎖の名前を入力し、「作成」ボタンをクリックします。
8. 「モジュールの追加」ボタンをクリックします。
9. 「モジュールの選択」プルダウンで作成したモジュールを、「基準の選択」は「Required」を選択し、「OK」ボタンをクリックします。
10. ActiveDirectory 認証連鎖の画面に戻るので、「変更の保存」ボタンをクリックします。
11. 左側メニューの「認証」→「設定」→「組織認証設定」にて、作成した認証連鎖を選択し、「変更の保存」ボタンをクリックします。

以上で完了です。

認証連鎖変更後の管理者コンソールへのログイン方法については「7.11.2 認証連鎖変更後の amadmin のログイン URL」を参照してください。

7.5 証明書認証モジュール

証明書認証モジュールは、クライアント証明書の Subject のユーザー名で認証するモジュールです。証明書認証モジュールを利用するためには、OpenAM サーバーにクライアント証明書を送付するようにシステムを構成する必要があります。

7.5.1 認証モジュールと認証連鎖の設定方法

証明書認証モジュールの設定方法について記載します。ここでは SSL 終端が OpenAM サーバーであることを前提としています。

1. OpenAM に管理者ユーザーでログインします。
2. 対象レルム →「認証」→「モジュール」を開きます。
3.「モジュールの追加」ボタンをクリックします。
4.「名前」に任意のモジュール名を入力し、「タイプ」は「証明書」を選択して、「作成」ボタンをクリックします。
5. 左側メニューの「認証」→「認証連鎖」→「認証連鎖の追加」をクリックします。
6.「新規認証連鎖」画面の「名前」に任意の認証連鎖の名前を入力し、「作成」ボタンをクリックします。
7.「モジュールの追加」ボタンをクリックします。
8.「モジュールの選択」プルダウンで作成したモジュールを、「基準の選択」は「Required」を選択し、「OK」ボタンをクリックします。
9. 認証連鎖の画面に戻るので、「変更の保存」ボタンをクリックします。
10. 左側メニューの「認証」→「設定」→「組織認証設定」にて、作成した認証連鎖を選択し、「変更の保存」ボタンをクリックします。

以上で完了です。

認証連鎖変更後の管理者コンソールへのログイン方法については「7.11.2 認証連鎖変更後の amadmin のログイン URL」を参照してください。

7.5.2 SSL 終端がロードバランサーの場合の追加設定

ロードバランサーが SSL 終端である場合、OpenAM サーバーにはクライアント証明書が送信されません。この場合は、ロードバランサーが OpenAM へのリクエストの HTTP ヘッダーに証明書を設定する必要があります。OpenAM 側では、HTTP ヘッダーからクライアント証明書を取得するための設定が必要です。

1. OpenAM に管理者ユーザーでログインします。
2. 対象レルム →「認証」→「モジュール」を開きます。
3. 作成していた証明書認証モジュールをクリックします。
4. 各パラメーターを入力し、「変更の保存」をクリックします。

【パラメータ名】	【設定値】
信頼できるリモートホスト	none を削除し、「ロードバランサーの IP アドレス」を指定する。
クライアント証明書用 HTTP ヘッダー名	ロードバランサーで証明書をセットする際の HTTP ヘッダー名を指定する。

以上で完了です。

7.5.3 クライアント証明書を LDAP に格納して検証する場合の追加設定

クライアントによって提示された証明書が LDAP に格納されている証明書と一致するか、チェックさせることができます。LDAP 側では、UserCertificate 属性に証明書を入れておく必要があります。OpenAM 側の設定は以下の通りです。

1. OpenAM に管理者ユーザーでログインします。
2. 対象レルム →「認証」→「モジュール」を開きます。
3. 作成していた証明書認証モジュールをクリックします。
4. 各パラメーターを入力し、「変更の保存」をクリックします。

【パラメータ名】	【設定内容】
LDAP で証明書を照合	チェックを入れる
証明書が格納されている LDAP サーバー	LDAP サーバーの情報
LDAP 検索開始 DN	検索開始する地点の DN
LDAP サーバーの主体ユーザー	LDAP に bind するユーザ
LDAP アクセスに SSL を使用	使用する場合はチェックを入れる

以上で完了です。

7.6 HOTP 認証モジュール

HOTP 認証モジュールは、電子メールでワンタイムパスワードをユーザーに発行するモジュールです。ワンタイムパスワードは「なりすまし」等の不正を防ぎ、認証の精度を高めるために有効な手段のひとつです。具体的には以下のように動作します。

1. ユーザは OpenAM にユーザ ID とパスワードを入力し、通常の認証を受けます。
2. 上記で認証された後、ユーザは OpenAM にワンタイムパスワードの発行を要求します。
3. OpenAM はユーザの登録済みメールアドレスに対してワンタイムパスワードを送付します。
4. ユーザはメールを受信し、送られて来たワンタイムパスワードを OpenAM のログイン画面に入力します。

以上でログイン処理が完了です。ワンタイムパスワードの生成には HMAC（Keyed-Hashing for Message Authentication Code）を使用し、セキュリティを考慮した作りになっています。

この認証方式を使うためには、メールを送信するための SMTP サーバと、ワンタイムパスワードを受け取るための電子メールクライアントが必要です。

ワンタイムパスワードは、必ず他の認証方式（多くの場合は、ユーザ ID/パスワードによる認証）と組み合わせて使用します。ワンタイムパスワードはユーザデータストアに登録されているメールアドレス宛てに送信されますが、そのメールアドレスを知るためにはユーザを認証してユーザデータストアからメールアドレスを取得する必要があるため、そのような認証連鎖を設定します。

7.6.1 認証モジュールと認証連鎖の設定方法

ここでは、ワンタイムパスワードを使用した認証連鎖の設定例を説明します。

1. OpenAM に管理者ユーザーでログインします。
2. 対象レルム →「認証」→「モジュール」を開きます。
3.「モジュールの追加」ボタンをクリックします。
4.「名前」に任意のモジュール名を入力し、「タイプ」は「HOTP」を選択して、「作成」ボタンをクリックします。
5. 以下の例のように各パラメーターを入力し、「変更の保存」をクリックします。

【項目】	【設定値】
SMTP ホスト名	SMTP サーバのホスト名です。 例：smtp.example.co.jp
SMTP ホストのポート	SMTP サーバの接続ポート番号です。 例：465（SSL の場合）
SMTP ユーザー名	SMTP サーバの接続ユーザ名です。 ※ SMTP 認証が無い SMTP サーバーの場合でも、適当な文字列を入力してください。空にすると「SMTP ホスト名」で指定した SMTP サーバーではなく、localhost の SMTP サーバーに接続するためです。
SMTP ユーザーパスワード	SMTP サーバの接続パスワードです。 ※ SMTP 認証が無い SMTP サーバーの場合でも、適当な文字列を入力してください。空にすると「SMTP ホスト名」で指定した SMTP サーバーではなく、localhost の SMTP サーバーに接続するためです。
SMTP 接続	SSL/TLS 接続を推奨します。
ワンタイムパスワードの有効期間	ワンタイムパスワードの有効期間を分単位で設定します。ここで設定した時間よりも認証モジュールのセッションタイムアウト時間の方が短い場合は、認証画面のセッションタイムアウトに伴いワンタイムパスワードも無効になります。
ワンタイムパスワードの長さ	ワンタイムパスワードの桁数を選択します。ワンタイムパスワードはランダムな数字となります。
ワンタイムパスワードの配布	「電子メール」に変更します。

6. 左側メニューの「認証」→「認証連鎖」→「認証連鎖の追加」をクリックします。
7. 「新規認証連鎖」画面の「名前」に任意の認証連鎖の名前を入力し、「作成」ボタンをクリックします。
8. 「モジュールの追加」ボタンを 2 回使用し、以下の例のように認証連鎖を構成します。

【モジュールの種類】	【基準】
データストア	Requisite
HOTP	Required

9. 「認証連鎖」の画面にて、「変更の保存」ボタンをクリックします。
10. 左側メニューの「認証」→「設定」→「組織認証設定」にて、作成した認証連鎖を選択し、「変更の保存」ボタンをクリックします。

以上で完了です。

認証連鎖を設定する際に、一つ目の DataStore 認証（ユーザ ID・パスワードによる認証方式のひとつ）の条件が「Requisite」になっていることに注意して下さい。この段階で失敗した場合には、そこで認証処理を中断し、ログイン失敗の画面を表示します。この認証に成功してユーザが特定されたら、ワンタイムパスワード認証に進みます。「OTP コードの要求」ボタンをクリックすると、ワンタイムパスワードがメールで送信されます。

認証連鎖変更後の管理者コンソールへのログイン方法については「7.11.2 認証連鎖変更後の amadmin のログイン URL」を参照してください。

7.6.2 メールメッセージの変更

ワンタイムパスワード認証モジュールにより送信されたメールは、デフォルトでは以下のようなメッセージとなります。

```
件名：OpenAM のワンタイムパスワード
本文：OpenAM のワンタイムパスワード: 77382746
```

これらのメッセージは、以下のプロパティファイルにて設定することができます。プロパティファイルは Unicode エスケープ形式で記述されているため、native2ascii コマンドなどで UTF-8 に変換して編集し、再び native2ascii で Unicode エスケープ形式に戻す必要があります。

```
英語：{OPENAM_INSTALL}/WEB-INF/classes/amAuthHOTP.properties
日本語：{OPENAM_INSTALL}/WEB-INF/classes/amAuthHOTP_ja.properties
```

デフォルトの設定ファイルは以下の場所にあります。これを任意の場所で解凍し、中身のプロパティファイル群を前記の場所にコピー・編集することになります。

```
{OPENAM_INSTALL}/WEB-INF/lib/openam-auth-hotp-14.x.x.jar
```

メールのメッセージは以下のプロパティです。

```
messageSubject=OpenAM のワンタイムパスワード
messageContent=OpenAM のワンタイムパスワード:
```

この部分を編集することで、任意のメッセージを表示させることも可能です。メッセージに改行を入れる場合は、以下のように、改行したい箇所に改行コード「\n」を記述します。

```
messageContent=OpenAM のワンタイムパスワードです。\nXX 分以内に入力してください。
```

設定変更を行い、OpenAM を再起動することで、変更が反映されます。なお、冗長構成を採っている場合は、全てのサーバで設定を行う必要があります。

7.7 OATH 認証モジュール

OATH 認証モジュールは、ワンタイムパスワードで認証するモジュールです。OATH 認証モジュールは、データストア認証モジュールなどと組み合わせて利用する必要があります。

OATH 認証では、OpenAM サーバーとユーザーのクライアント端末（スマートフォンなど）が互いに共通鍵を保持します（OpenAM 側では、データストアのユーザーエントリの属性において、ユーザー毎に異なる鍵を保持します）。OpenAM は、ユーザーが送信したパスワードと自身が生成するパスワードとが一致するかどうかを確認します。

OATH 認証には以下の 2 つの方式があります。

- HMAC-Based One-time Password Algorithm (HOTP)：生成回数ベース
- Time-Based One-Time Password Algorithm (TOTP)：時刻ベース

7.7.1 前提条件

ここでは、以下のオブジェクトクラスと属性を利用することを前提としています。ディレクトリサーバーにスキーマを追加する必要があります。

【属性】	【値】
オブジェクトクラス	am-auth-oath-service
秘密鍵の属性	am-auth-oath-secret-key

【属性】	【値】
カウンタ属性	am-auth-oath-hotp-counter
最終ログイン時間属性	am-auth-oath-last-login-time

OpenAM14 では、OpenAM 用 LDAP スキーマパッケージ「osstech-openam-ldapschema」内に上記の属性定義を含んだ oath.schema がありますので、こちらをご利用ください。

7.7.2 認証モジュールと認証連鎖の設定方法

1. OpenAM に管理者ユーザーでログインします。
2. 対象レルム →「認証」→「モジュール」を開きます。
3.「モジュールの追加」ボタンをクリックします。
4.「名前」に任意のモジュール名を入力し、「タイプ」は「OATH」を選択して、「作成」ボタンをクリックします。
5. 各パラメーターを入力し、「変更の保存」をクリックします。

【項目】	【設定値】
秘密鍵の属性名	am-auth-oath-secret-key
使用する OATH アルゴリズム	HOTP または TOTP を選択します。
カウンタ属性名	am-auth-oath-hotp-counter
最終ログイン時間属性	am-auth-oath-last-login-time

また、以下のパラメータを任意で調整することができます。

【項目】	【設定値】
ワンタイムパスワードの長さ	生成する OTP の桁数。 6 以上の数字を入力してください。
秘密鍵の最小桁数	秘密鍵として許容される 16 進数文字の桁数。

【項目】	【設定値】
HOTP ウインドウサイズ	アルゴリズムが HOTP の場合に使用。 OpenAM と OTP 生成デバイスの HOTP の生成回数のズレを許容する範囲。
チェックサム数字の追加	OTP にチェックサムを追加するかどうか。 チェックサムとは OTP が正しく生成されたかどうかを判断するための追加の桁です。 使用する OTP 生成デバイスが対応している場合のみ True に設定することができます。
トランケーションオフセット	HOTP 利用時のオフセットの値。使用する OTP 生成デバイスが対応している場合のみ 1 以上の値を設定することができます。
TOTP タイムステップ期間	アルゴリズムが TOTP の場合に使用。 TOTP を生成する時間間隔 (秒)。
TOTP タイムステップ数	アルゴリズムが TOTP の場合に使用。 OpenAM が検証対象とする TOTP のステップ数。
最大許容クロックドリフト	アルゴリズムが TOTP の場合に使用。 OpenAM と OTP 生成デバイスの TOTP のステップ数のズレを許容する範囲。[a]

[a] 入力された TOTP は「最大許容クロックドリフト」「TOTP タイムステップ数」両方を満たす必要があるので、「最大許容クロックドリフト」の値が「TOTP タイムステップ数」の値以上となるように設定してください。
例えば両方を 0 とした場合、有効期間を過ぎた TOTP は認証に失敗します。両方を 1 とした場合、TOTP 入力中に有効期間が過ぎてしまった場合でも 1 ステップまでであれば認証成功とします。

6. 左側メニューの「認証」→「認証連鎖」→「認証連鎖の追加」をクリックします。
7. 「新規認証連鎖」画面の「名前」に任意の認証連鎖の名前を入力し、「作成」ボタンをクリックします。

8.「モジュールの追加」ボタンを 2 回使用し、以下の例のように認証連鎖を構成します。

【モジュールの種類】	【基準】
データストア	Requisite
OATH	Required

9.「認証連鎖」の画面にて、「変更の保存」ボタンをクリックします。
10. 左側メニューの「認証」→「設定」→「組織認証設定」にて、作成した認証連鎖を選択し、「変更の保存」ボタンをクリックします。

引き続きユーザーデータストアの設定を行います。

7.7.3 ユーザーデータストアの設定

1. OpenAM 管理コンソール→対象レルム→「データストア」を開きます。
2. 対象のデータストアをクリックします。
3. パラメーターを追加します。

【項目】	【設定値】
LDAP ユーザーオブジェクトクラス	am-auth-oath-service
LDAP ユーザー属性	am-auth-oath-secret-key am-auth-oath-hotp-counter am-auth-oath-last-login-time

4. 画面右上の「保存」をクリックします。

以上で OpenAM 側の設定は完了です。

7.7.4 ユーザーエントリへの鍵の設定

OATH 認証を利用するためには、ディレクトリサーバーの各ユーザーエントリへ、秘密鍵などの属性を設定する必要があります。

1. 下の例のように、byte の乱数を鍵として生成する。 最終的に取得する 16 進文字列が認証モジュールと認証連鎖の設定方法で設定した「秘密鍵の最小桁数」以上となる

ように、乱数は「秘密鍵の最小桁数」の半分以上の byte 数で生成してください。鍵長は 10 の倍数 (byte) にすることを推奨します。（ここでは 20 bytes）

```
Hex [9542F3DB9A2A340949B34EEB8713ED49C40737A3]
```

2. 上記 1. で生成した乱数を 16 進文字列で取得します。
3. ユーザーエントリに秘密鍵などの属性を設定します。

【属性】	【設定値 (例)】
am-auth-oath-secret-key	9542F3DB9A2A340949B34EEB8713ED49C40737A3
am-auth-oath-hotp-counter	0
am-auth-oath-last-login-time	0

7.7.5 クライアント端末への鍵の登録

OATH 認証を利用するためには、クライアント端末に秘密鍵を設定する必要があります。ここでは、クライアント端末としてスマートフォン、鍵生成ソフトウェアとして Google Authenticator を前提として説明します。

1. 秘密鍵を Base32 文字列で取得します。

```
Base32 [SVBPHW42FI2ASSNTJ3VYOE7NJHCAON5D]
```

2. スマートフォンに Google Authenticator をインストールします。
3. 以下の URL にアクセスして登録用の QR コードを生成します。

【場合】	【QR コード生成方法】
HOTP の場合	=== フォーマット === http://chart.apis.google.com/chart?chs=150x150&cht=qr&chl=otpauth://hotp/＜ラベル＞?secret=＜秘密鍵の Base32 の値＞ === 例 === http://chart.apis.google.com/chart?chs=150x150&cht=qr&chl=otpauth://hotp/HOTP?secret=SVBPHW42FI2ASSNTJ3VYOE7NJHCAON5D
TOTP の場合	=== フォーマット === http://chart.apis.google.com/chart?chs=150x150&cht=qr&chl=otpauth://totp/＜ラベル＞?secret=＜秘密鍵の Base32 の値＞ === 例 === http://chart.apis.google.com/chart?chs=150x150&cht=qr&chl=otpauth://totp/TOTP?secret=SVBPHW42FI2ASSNTJ3VYOE7NJHCAON5D

4. Google Authenticator で上記の QR コードを読み込みます。

以上で設定は完了です。

7.8 ForgeRock Authenticator (OATH) 認証モジュール

ForgeRock Authenticator (OATH) 認証モジュールは OATH 認証モジュールと同様の機能を持ちますが、秘密鍵の生成と登録が認証モジュール内に組み込まれているため、管理者が設定する必要はありません。OTP 生成デバイスが未登録のユーザーがログインしようとすると、デバイスの登録を促すメッセージが表示されます。登録画面に進むと自動で生成された QR コードが画面に表示されます。これを読み取ることで OpenAM と OTP 生成デバイスで秘密鍵が共有され、デバイスの登録が完了します。デバイスの登録が完了すると、以後はそのデバイスで生成した OTP で認証することが可能となります。

7.8.1 認証モジュールと認証連鎖の設定方法

1. OpenAM に管理者ユーザーでログインします。
2. 対象レルム →「認証」→「モジュール」を開きます。
3.「モジュールの追加」ボタンをクリックします。
4.「名前」に任意のモジュール名を入力し、「タイプ」は「ForgeRock Authenticator(OATH)」

を選択して、「作成」ボタンをクリックします。

5. 「発行者の名前」に任意の名前を入力します。この名前が OTP 生成デバイス側に表示されます。
6. 必要に応じてその他のパラメータを変更し、「変更の保存」をクリックします。各パラメータの意味については OATH 認証モジュールの設定方法を参照してください。
7. 左側メニューの「認証」→「認証連鎖」→「認証連鎖の追加」をクリックします。
8. 「新規認証連鎖」画面の「名前」に任意の認証連鎖の名前を入力し、「作成」ボタンをクリックします。
9. 「モジュールの追加」ボタンを 2 回使用し、以下の例のように認証連鎖を構成します。

【モジュールの種類】	【基準】
データストア	Requisite
ForgeRock Authenticator(OATH)	Required

10. 「認証連鎖」の画面にて、「変更の保存」ボタンをクリックします。
11. 左側メニューの「認証」→「設定」→「組織認証設定」にて、作成した認証連鎖を選択し、「変更の保存」ボタンをクリックします。

以上で設定は完了です。

7.9 Windows デスクトップ SSO 認証モジュール

Windows デスクトップ SSO(統合 Windows 認証) の設定方法について説明します。なお、ホスト名などを以下のように仮定します。

【項目】	【値】
Active Directory ドメイン名	example.co.jp (Kerberos レルム名は EXAMPLE.CO.JP)
Active Directory ホスト名	ad.example.co.jp

7.9.1 時刻の同期

Windows デスクトップ SSO の機能は、内部的に Kerberos を利用しています。そのため、関連するサーバー間 (Active Directory、OpenAM) で時刻が正確に同期されている必要があ

ります。

7.9.2 Windows Active Directory の設定

Active Directory の設定について説明します。作業は Active Directory サーバー上で実施します。

7.9.2.1 Windows DesktopSSO 設定用のユーザーの作成

Active Directory に Windows デスクトップ SSO 設定用のユーザーを作成します。ユーザー名は何でもかまいません。ここでは、“openam” という名前のユーザーを作成したと仮定します。
暗号化方式として AES256-SHA1 を使用するために、作成したアカウントが AES256-SHA1 をサポートするように設定を変更します。

1. 「Active Directory ユーザーとコンピューター」から「example.co.jp」→「Users」→「openam」をクリックします。
2. 「アカウント」タブをクリックします。
3. 「アカウントオプション」の「このアカウントで Kerberos AES 256 ビット暗号化をサポートする」にチェックを入れ、「適用」をクリックします。

7.9.2.2 keytab ファイルの作成

「Windows デスクトップ SSO 設定用のユーザーの作成」で作成したユーザーに対して、OpenAM 用のサービスプリンシパル名 (SPN) を登録し、keytab ファイルを作成します。
以下のコマンドを実行して keytab ファイルを作成します。ここでは、keytab ファイル名を openam.keytab とします。

```
> ktpass -out openam.keytab -princ HTTP/sso.example.co.jp@EXAMPLE.CO.JP
  -mapuser "openam" -pass password -crypto All -ptype KRB5_NT_PRINCIPAL
```

「HTTP/sso.example.co.jp」の部分には、OpenAM がシングルサーバー構成である場合は OpenAM サーバーの FQDN を指定します。OpenAM が冗長化構成 (サイト構成) の場合は、ロードバランサーの FQDN を指定します。
-mapuser には作成したユーザーを指定します。ユーザー名表記 (openam) で keytab ファイル作成が失敗する場合は UPN 形式を指定してください。
作成した keytab ファイル (openam.keytab) を OpenAM サーバーにコピーします。

7.9.3 OpenAM の設定

7.9.3.1 keytab ファイルの配置

Active Directory 上で作成した keytab ファイルを OpenAM サーバー上に配置し、ファイルパーミッションを設定します。任意のディレクトリに保存可能ですが、パーミッションは root ユーザーと Tomcat プロセス実行ユーザーのみ読み取り可能とすることにご注意ください。

```
# mkdir /opt/osstech/var/lib/tomcat/data/openam/openam/private
# mv openam.keytab /opt/osstech/var/lib/tomcat/data/openam/openam/private
# chown -R root:tomcat /opt/osstech/var/lib/tomcat/data/openam/openam/private
# chmod 750 /opt/osstech/var/lib/tomcat/data/openam/openam/private
# chmod 640 /opt/osstech/var/lib/tomcat/data/openam/openam/private/openam.keytab
```

7.9.3.2 暗号化方式として RC4-HMAC-NT を使用する場合

＜注意＞ RC4-HMAC-NT は暗号化強度が弱いため、以下に示す方法は非推奨です。やむを得ず RC4-HMAC-NT を使う必要がある場合のみ実施してください。

OpenAM サーバーが RHEL8 かつ暗号化方式として RC4-HMAC-NT を使用する場合、デフォルトで RC4-HMAC-NT が許容されていないため、OS 全体の暗号化ポリシーを変更する必要があります。コマンドのオプションについてはご利用の OS のバージョンに依存しますので、詳しくは Red Hat のドキュメントを参照してください。

- OpenAM サーバーが RHEL8.0〜8.2 の場合

8.2 以前で RC4-HMAC-NT を有効化する方法については Red Hat のドキュメントをご覧ください。

- OpenAM サーバーが RHEL8.3 以降の場合

```
# update-crypto-policies --set DEFAULT:AD-SUPPORT
```

変更後、OS を再起動してください。

7.9.3.3 認証モジュールの設定

OpenAM の認証モジュールを設定します。

1. OpenAM に管理者ユーザーでログインします。

2. 対象レルム →「認証」→「モジュール」を開きます。
3.「モジュールの追加」ボタンをクリックします。
4.「名前」に任意のモジュール名を入力し、「タイプ」は「Windows デスクトップ SSO」を選択して、「作成」ボタンをクリックします。
5. 各パラメーターを入力し、「変更の保存」をクリックします。以下はパラメータの例です。

【項目】	【設定値】
サービス主体	HTTP/sso.example.co.jp@EXAMPLE.CO.JP
Keytab ファイル名	/opt/osstech/var/lib/tomcat/data/openam/openam/private/openam.keytab
Kerberos レルム	EXAMPLE.CO.JP
Kerberos サーバー名	ad.example.co.jp
ドメイン名を含む主体を返す	無効
信頼された Kerberos レルム	[設定なし]
レルム内のユーザー検索	無効
Template JSP for NTLM	[設定なし]

6. 左側メニューの「認証」→「認証連鎖」→「認証連鎖の追加」をクリックします。
7.「新規認証連鎖」画面の「名前」に任意の認証連鎖の名前を入力し、「作成」ボタンをクリックします。
8.「モジュールの追加」ボタンをクリックします。
9.「モジュールの選択」プルダウンで作成したモジュールを、「基準の選択」は「Required」を選択し、「OK」ボタンをクリックします。
10. 認証連鎖の画面に戻るので、「変更の保存」ボタンをクリックします。
11. 左側メニューの「認証」→「設定」→「組織認証設定」にて、作成した認証連鎖を選択し、「変更の保存」ボタンをクリックします。

12. OpenAM の再起動を実施します。

以上で完了です。

認証連鎖変更後の管理者コンソールへのログイン方法については「7.11.2 認証連鎖変更後の amadmin のログイン URL」を参照してください。

7.9.4 Internet Explorer の設定変更

ドメインに参加している Windows 端末の Internet Explorer の設定を変更します。

まず、OpenAM の URL を「ローカルイントラネット」のサイトに追加します。

- 「インターネットオプション」→「セキュリティ」→「ローカルイントラネット」→「サイト」→「詳細設定」に OpenAM のサイトを追加します。以下のような URL となります。
  - http://sso.example.co.jp

次に、Internet Explorer の設定で「統合 Windows 認証」を有効化します。

1. 「インターネットオプション」 → 「詳細設定」 を開きます。
2. 「統合 Windows 認証を使用する」にチェックを入れます。
3. コンピューターを再起動します。

以上で設定は完了です。

7.10 アダプティブリスク認証モジュール

アダプティブリスク認証モジュールは、認証するユーザーの要素からリスク値を判断し、リスクが高い場合には追加の認証を求めるモジュールです。そのため、データストア認証や OpenLDAP 認証といった認証モジュールに合わせ、さらに HOTP 認証などの追加認証を行う種類のモジュールと組み合わせて利用します。また、リスクの高いログイン成功に対し、事前に登録されたユーザーのアドレス宛てにメールを送信することも可能です。

7.10.1 リスク評価の種類

アダプティブリスク認証で判定できるリスクの種類を示します。

【種類】	【評価内容】
認証失敗	直前の認証処理時に失敗していたかをチェックします。直前のログイン時にパスワードミスで失敗していればリスクが高いと判定します。
IP アドレスレンジ	OpenAM に IP アドレスの範囲を設定します。ユーザーが IP レンジの範囲内でなければリスクが高いと判定します。
IP アドレス履歴	ユーザープロファイルの属性に格納されている IP アドレスと一致するかをチェックします。属性名は任意の名前を使用できます。属性値は、|(パイプ) で区切り複数の IP アドレスを定義することができます。(例：192.168.0.1|192.168.0.2)。履歴と一致しなければリスクが高いと判定します。
既知の Cookie	指定された Cookie を保持しているかどうかチェックします。Cookie がなければリスクが高いと判定します。
最終ログインからの経過時間	前回ログイン時から経過時間をチェックします。ログイン成功時に暗号化されたログイン時刻が Cookie にセットされ、OpenAM は前回のログイン時刻を取得します。Cookie が無い場合や最終ログイン時刻が設定した期間以前であればリスクが高いと判定します。
プロファイル属性	ユーザープロファイルにある属性と値をチェックします。設定した属性名の属性値がないユーザーはリスクが高いと判定します。
デバイス登録 Cookie	デバイス登録識別子 Cookie を保持しているかチェックします。デバイス登録識別子 Cookie とは、User-Agent や Accept-*系のリクエスト HTTP ヘッダー、ユーザー ID などを暗号化した値で、認証成功時に OpenAM が設定します。Cookie がなければリスクが高いと判定します。
位置情報	クライアントの IP アドレスから国コードを判別し、許容する国かチェックします。IP アドレスが設定された国以外であればリスクが高いと判定します。

【種類】	【評価内容】
位置情報履歴	クライアントの IP アドレスから国コードを判別し、ユーザープロファイルの属性に格納されている履歴のリストと比較します。判別された国コードが履歴のリストに含まれていない場合、リスクが高いと判定します。
リクエストヘッダー	リクエスト HTTP ヘッダーに指定した値がセットされているかチェックします。セットされていなければリスクが高いと判定します。

7.10.2 認証モジュールと認証連鎖の設定方法

ここでは、ユーザーの IP アドレスでリスク判定をする場合の設定例を示します。IP アドレスが範囲外の場合に HOTP 認証を求めます。なお、事前にデータストア認証及び HOTP 認証が設定されていることを前提としています。

1. OpenAM に管理者ユーザーでログインします。
2. 対象レルム →「認証」→「モジュール」を開きます。
3.「モジュールの追加」ボタンをクリックします。
4.「名前」に任意のモジュール名を入力し、「タイプ」は「アダプティブリスク」を選択して、「作成」ボタンをクリックします。
5. 各パラメーターを入力し、「変更の保存」をクリックします。以下はパラメータの例です。

【項目】	【設定値】
IP レンジチェック	有効
IP レンジ	リスクが低いと判断する IP レンジを設定します。

6. 左側メニューの「認証」→「認証連鎖」→「認証連鎖の追加」をクリックします。
7.「新規認証連鎖」画面の「名前」に任意の認証連鎖の名前を入力し、「作成」ボタンをクリックします。
8.「モジュールの追加」ボタンを 3 回使用し、以下の例のように認証連鎖を構成します。

【モジュールの種類】	【基準】
データストア	Requisite
アダプティブリスク	Sufficient
HOTP	Requisite

9. 「認証連鎖」の画面にて、「変更の保存」ボタンをクリックします。
10. 左側メニューの「認証」→「設定」→「組織認証設定」にて、作成した認証連鎖を選択し、「変更の保存」ボタンをクリックします。

以上で設定は完了です。

7.10.3 認証ポストプロセスクラスの設定

リスクベース認証には、前回のログイン成功時に設定した LDAP 属性や Cookie 値を利用して判定するリスク評価があります。

- IP アドレス履歴
- 既知の Cookie
- デバイス登録 Cookie
- 最終ログインからの経過時間

これらを動作させるためには、認証ポストプロセスクラスを設定する必要があります。認証ポストプロセスクラスを設定することで、認証成功時に LDAP 属性の更新や Cookie の設定が行われるようになります。

1. OpenAM 管理コンソール → 対象レルム → 「認証」→「設定」を開きます。
2. 「ポスト認証プロセス」の「認証ポストプロセスクラス」に「org.forgerock.openam.authentication.modules.adaptive.Adaptive」を追加します。
3. 「変更の保存」ボタンをクリックします。
4. OpenAM を再起動します。

以上で設定は完了です。

7.11 持続 Cookie 認証モジュール

通常 OpenAM が発行する Cookie はセッション Cookie であり、ブラウザを閉じることでセッションは失われます。持続 Cookie 認証モジュールは、認証成功時に有効期限付きの

Cookie を発行することで、ブラウザを閉じても認証状態を継続させることができるモジュールです。

7.11.1 組織認証用鍵ペアの作成

持続 Cookie 認証モジュールでは、公開鍵暗号化方式で暗号化した情報を持つ Cookie を作成します。まず JDK の keytool コマンドを利用して鍵ペアを作成します。ここでは、OpenAM が利用するデフォルトのキーストアに追加することを前提としています。

```
$ keytool -genkeypair \
-keyalg rsa \
-alias top-realm \
-dname "CN=sso.example.co.jp,OU=development,O=EXAMPLE,
L=Shinagawa,ST=Tokyo,C=JP" \
-keypass  changeit \
-keystore /opt/osstech/var/lib/tomcat/data/openam/openam/keystore.jks \
-storepass changeit \
-validity 3650 \
-keysize 2048
```

キーストアのパスワード (-storepass) 及び秘密鍵のパスワード (-keypass) は、OpenAM のサーバー設定のキーストアの設定に合わせる必要があります。”changeit” は OpenAM が利用するデフォルトのキーストアのパスワードです。

7.11.2 組織認証用の証明書エイリアスの変更

前節で作成した証明書を利用するよう、OpenAM の設定を変更します。

1. OpenAM 管理コンソール → 「設定」 → 「認証」 → 「コア属性」を開きます。
2. 「セキュリティ」の「組織認証の証明書のエイリアス」に「top-realm」を設定します。
3. 「変更の保存」ボタンをクリックします。

引き続き、認証モジュールと認証連鎖の設定を行います。

7.11.3 認証モジュールと認証連鎖の設定方法

ここでは、持続 Cookie 認証を使用した認証連鎖の設定例を説明します。なお、事前にデータストア認証が設定されていることを前提にしています。

1. OpenAM に管理者ユーザーでログインします。
2. 対象レルム → 「認証」→「モジュール」を開きます。

3.「モジュールの追加」ボタンをクリックします。
4.「名前」に任意のモジュール名を入力し、「タイプ」は「持続 Cookie」を選択して、「作成」ボタンをクリックします。
5. 各パラメーターを入力し、「変更の保存」をクリックします。以下はパラメータの例です。

【項目】	【設定値】
アイドルタイムアウト	Cookie が無効になるまでの、リクエスト間の最大アイドル時間を指定します (時間単位)。
生存期間	Cookie の最大生存期間を指定します (時間単位)。

6. 左側メニューの「認証」→「認証連鎖」→「認証連鎖の追加」をクリックします。
7.「新規認証連鎖」画面の「名前」に任意の認証連鎖の名前を入力し、「作成」ボタンをクリックします。
8.「モジュールの追加」ボタンを 2 回使用し、以下の例のように認証連鎖を構成します。

【モジュールの種類】	【基準】
持続 Cookie	Sufficient
データストア	Requisite

9.「認証連鎖」の画面にて、「変更の保存」ボタンをクリックします。
10. 左側メニューの「認証」→「設定」→「組織認証設定」にて、作成した認証連鎖を選択し、「変更の保存」ボタンをクリックします。

引き続き、認証ポストプロセスクラスの設定を行います。

7.11.4 認証ポストプロセスクラスの設定方法

持続 Cookie 認証は、前回ログイン成功時に設定した Cookie 値を利用して認証します。そのためには認証ポストプロセスクラスを設定する必要があります。認証ポストプロセスクラスを設定することで、認証成功時に Cookie の設定が行われるようになります。

1. OpenAM 管理コンソール → 対象レルム →「認証」→「設定」を開きます。
2.「ポスト認証プロセス」の「認証ポストプロセスクラス」に以下を追加します。

「org.forgerock.openam.authentication.modules.persistentcookie.PersistentCookieAuthModule」

3.「変更の保存」ボタンをクリックします。
4. OpenAM を再起動します。

以上で設定は完了です。

7.12 認証連鎖

認証連鎖とは、認証モジュールの組み合わせを定義したものです。１番目の認証に失敗しても２番目の認証で成功すれば全体の認証を成功とするパターン、１番目と２番目の認証の両方に成功する必要があるパターンなど、さまざまな認証連鎖を定義することが可能です。

7.12.1 認証連鎖の設定

認証連鎖の設定方法を記載します。利用する認証モジュールは事前に定義しておく必要があります。

1. OpenAM 管理コンソール → 対象レルム →「認証」→「認証連鎖」を開きます。
2.「認証連鎖の追加」をクリックします。「新規認証連鎖の作成」画面の「認証連鎖名」に認証連鎖の名前を入力し、「作成」ボタンをクリックします。
3.「モジュールの追加」ボタンをクリックし、認証に利用する順番で認証モジュール（のインスタンス）を追加していきます。
4. 各認証モジュール（のインスタンス）に対して適用する基準を選択します。

【基準】	【動作】
Optional	認証モジュールの成功は必要ではありません。成功したかどうかに関係なく、リスト内の次の認証モジュールに進みます。
Sufficient	認証モジュールの成功は必要ではありません。成功すると、リスト内の次の認証モジュールに進みません。失敗すると、リスト内の次の認証モジュールに進みます。
Required	認証モジュールの成功が必要です。成功したかどうかに関係なく、リスト内の次の認証モジュールに進みます。
Requisite	認証モジュールの成功が必要です。成功すると、リスト内の次の認証モジュールに進みます。失敗すると、リスト内の次の認証モジュールに進みません。

5. オプションを設定します。一例として、２番目に設定した認証モジュールで１番目の認証モジュールで入力したパスワードを利用したい場合は、２番目の認証モジュールのオプションに下記の値を入力します。

【キー】	【値】
iplanet-am-auth-shared-state-enabled	true
iplanet-am-auth-shared-state-behavior-pattern	useFirstPass

6. 各「モジュールの編集」画面では、「OK」ボタンを押します。
7. 「認証連鎖」の画面にて、「変更の保存」ボタンをクリックします。
8. 左側メニューの「認証」→「設定」→「組織認証設定」にて、作成した認証連鎖を選択し、「変更の保存」ボタンをクリックします。

以上で完了です。

認証連鎖変更後の管理者コンソールへのログイン方法については「7.11.2 認証連鎖変更後の amadmin のログイン URL」を参照してください。

7.12.2 認証連鎖変更後の amadmin のログイン URL

認証連鎖の設定を変更してデフォルトとは異なる認証モジュールを有効に変更した場合、OpenAM は管理者アカウントである amadmin も新たに設定された認証連鎖で認証しようと試みます。そのため、通常のログイン URL からログインを試みると、amadmin の正しいパスワードを入力したにもかかわらずエラーとなります。

amadmin でログインするためには、URL に「service=adminconsoleservice」パラメータを付加します。

```
https://sso.example.co.jp/openam/XUI/#login/&service=adminconsoleservice
```

7.13 アカウントロックアウト設定

連続して認証に失敗した場合にアカウントをロックアウトする方法を説明します。

なお、本節は OpenAM の機能によってロックアウトを実現する場合について説明するものであり、OpenLDAP のパスワードポリシーの機能を用いてロックアウトを実現する場合とは異なります。そちらの場合は、別資料「OpenLDAP パスワードポリシー運用ガイド」を参照ください。

7.13.1 アカウントロックアウト方式

アカウントロックアウトには以下の 2 種類の方式があります。性能の面からはメモリに保持する方式が推奨されます。パスワードの総当たり攻撃を防ぐ目的であれば、メモリに保持する方式で問題ありません。

- ロックアウト情報をメモリに保存する方式
  - アカウントロックアウト情報をメモリに保持します。
  - OpenAM が複数台構成の場合は情報がサーバー間で共有されないため、ロックアウト後に別のサーバーにアクセスした場合、認証操作が可能となる場合があります。
- ロックアウト情報をユーザーデータストアに保存する方式
  - アカウントロックアウト情報をユーザーデータストアに保持します。
  - OpenAM が複数台構成の場合もサーバー間で情報が共有されます。
  - 認証処理毎に LDAP への更新処理が発生するため、メモリに保持する場合と比べ負荷が高くなります。

7.13.2 ロックアウト設定の有効化

1. OpenAM 管理コンソール → 対象のレルム→「認証」→「設定」→「アカウントロック」を開きます。
2. 設定値を編集します。主な設定値は以下のとおりです。

【項目】	【設定値】
ログイン失敗時のロックアウトモード	有効（チェックする）
アカウントロックされる認証失敗回数	アカウントがロックアウトされるまでの認証失敗回数です。
認証失敗回数が加算される期間(分)	認証失敗回数をカウントする間隔です。この時間内に連続して認証に失敗すると、アカウントがロックアウトされます。

【項目】	【設定値】
ユーザーに警告を出すまでの失敗回数	認証失敗回数がこの回数に達した際に、ログイン画面に警告メッセージが表示されます。 警告メッセージを表示しない場合は 0 を設定します。
ログイン失敗時のロックアウト持続時間 (分)	アカウントがロックアウトされた後、ロックアウトを解除するまでの時間を指定します。
無効な試行をデータストアに格納する	アカウントロックアウト情報 (認証失敗回数、アカウントロック日時など) を保持する方法です。 メモリに保存する場合は無効（チェックを外す）に、 ユーザーデータストアに保存する場合は有効 （チェックする）にします。

3.「変更の保存」ボタンをクリックします。

以上で完了です。

7.13.3 ロックアウトの強制解除

ロックアウト状態がユーザーデータストアの属性として保存されている場合、この属性値を変更・削除することで、ロックアウトを強制的に解除できる場合があります。

ロックアウト持続時間の設定、及び、ロックアウト方式によって、採れる解強制除方法が異なります。

【ロックアウト持続時間】	【ロックアウト方式】	【強制解除方法】
0 分（管理者が解除するまで永続）	-	inetUserStatus 属性を Inactive から Active に変更する。
1 分以上	メモリ	なし（ロックアウト持続時間の経過を待つ）
1 分以上	データストア	sunAMAuthInvalidAttemptsData 属性を 削除する。

ユーザーデーターストアの情報は OpenAM にキャッシュされるため、属性を更新・削除した後も、OpenAM にキャッシュが残っている間はロックアウトが継続されます。ロック

アウト解除を即時に反映させたい場合は、属性を更新・削除した後に OpenAM を再起動する必要があります。

「sunAMAuthInvalidAttemptsData」属性には XML 形式のデータが保存されます。各パラメータの意味は以下の表の通りです。

【項目】	【説明】
InvalidCount	連続して認証に失敗した回数
LastInvalidAt	最後に認証に失敗した時刻 (UNIX 時間。ミリ秒。)
LockedoutAt	アカウントがロックアウトされた時刻
ActualLockoutDuration	アカウントがロックアウトされる期間

7.14 共通オプション

認証に関して設定する項目です。設定箇所は「対象のレルム」→「認証」→「設定」です。

7.14.1 ログイン失敗時に返すデフォルトの URL

「ログイン失敗時に返すデフォルトの URL」に値を入れると、以下のエラーが発生した場合には、設定した URL に遷移します。

- 認証エラー
  - ユーザー ID/PW 間違いなど
- ユーザーが非アクティブの場合
- ユーザーデータストアにユーザーが存在しない場合

以下のエラーの場合は、設定した URL には遷移しません。

- 認証時のセッションタイムアウト
- 認証後のセッションタイムアウト

8 セッション管理

8.1 OpenAM 全体の最大セッション数の変更

OpenAM 上で生成可能な最大セッション数の設定方法を説明します。

1. OpenAM 管理コンソールの「デプロイメント」→「サーバー」→対象サーバー→「セッション」を開きます。
2. 「最大セッション数」の鍵をクリックして変更可能状態にします。
3. 「最大セッション数」に許可する最大のセッション数を入力します。
4. 「変更の保存」をクリックします。

以上で完了です。

8.2 ユーザーセッション数を制限する

ユーザー毎のセッション数（同一ユーザーの同時ログイン数）を制限する方法を説明します。

1. OpenAM 管理コンソールの「設定」→「グローバルサービス」→「セッション」を開きます。
2. 「グローバル属性」の「Session Quotas」の欄までスクロールし、値を変更します。
   - 「割り当て制限を有効」の「オン」を選択します。
   - 「セッション制限がいっぱいになった場合に生じる動作」を選択します。

【設定項目】	【説明】
DENY_ACCESS	新しいセッションの作成要求が拒否される
DESTROY_NEXT_EXPIRING	次の有効期限切れセッションが破棄される
DESTROY_OLDEST_SESSION	最も古いセッションが破棄される
DESTROY_OLD_SESSIONS	以前のすべてのセッションが破棄される

3. 「動的属性」の「アクティブなユーザーセッション」の値を変更します。
4. 画面右下の「保存」をクリックします。
5. OpenAM を再起動します。

以上で完了です。

8.3 セッションタイムアウト時間を設定する

OpenAM では以下のセッションタイムアウト時間を設定可能です。

【設定項目】	【説明】
最大セッション時間	セッションが期限切れになるまでの時間を分単位で指定します。アイドル状態とならず OpenAM へアクセスし続けていた場合でも、ここで設定した時間が経過するとセッションが削除され、ユーザーは再び認証を要求されます。
最大アイドル時間	セッションが期限切れになるまでの無操作時間を分単位で指定します。無操作時間がここで設定した時間だけ続くと、ユーザーは再び認証を要求されます。

設定手順は以下となります。

1. OpenAM 管理コンソールの「設定」→「グローバルサービス」→「セッション」を開きます。
2. 「動的属性」の設定値を変更します。
   - 最大セッション時間 (分)
   - 最大アイドル時間 (分)
3. 画面右下の「保存」をクリックします。

以上で完了です。

注意点として、「最大アイドル時間」は「最大セッション時間」よりも短い時間に設定する必要があります。「最大アイドル時間」を「最大セッション時間」よりも長い時間に設定した場合は、「最大セッション時間」が経過してもセッションが破棄されません。

8.4 Cookie ドメインの管理

OpenAM が発行する Cookie のドメインを追加/変更/削除する方法を説明します。

1. OpenAM 管理コンソールの「設定」→「グローバルサービス」→「システム」タブ→「プラットフォーム」を開きます。
2. 「Cookie ドメイン」のドメイン名を必要に応じて追加/削除します。

3. 画面右上の「保存」をクリックします。

以上で完了です。

「Cookie ドメイン」に複数のドメインが登録されている場合は、それぞれのドメインを domain 属性に持つ Cookie がクライアントに対して発行されます。

9 冗長化構成（サイト構成）

冗長化構成（サイト構成）とは、OpenAM を 2 台以上構築する構成です。ロードバランサーの背後に配置された複数の OpenAM サーバー群をサイトと呼びます。本章では、2 台の OpenAM を冗長化構成にするものと仮定し、サイトを構成する手順を説明します。

設定作業は 1 号機に対して行います。各 OpenAM の URL は以下のようなものであると仮定します。

- OpenAM1 号機 : http://oam1.example.co.jp:8080/openam/
- OpenAM2 号機 : http://oam2.example.co.jp:8080/openam/
  - インストール時に「既存の配備に追加します」を選択して構築しているものとします。
- ロードバランサー : http://sso.example.co.jp/openam/

9.1 サイト構成の追加

1. OpenAM 管理コンソールの「デプロイメント」→「サイト」→「サイトの追加」を開きます。
2. 「名前」にサイト名を入力します。ここでは、「site1」と入力します。
3. 「Primary URL」にロードバランサーの URL を入力します。ここでは、「http://sso.example.co.jp/openam」と入力することにします。URL の最後に「/(スラッシュ)」をつけない点にご注意ください。
4. 画面右下の「作成」ボタンをクリックします。
5. 続いて、作成したサイトに実サーバーを登録します。「デプロイメント」→「サーバー」を開き、一覧にあるサーバーのうち一つをクリックします (ここでは、1 号機をクリックしたとします)。
6. 「サイト」→「親サイト」から、先程作成した「site1」を選択し、「変更の保存」ボタンをクリックします。
7. 「サーバー」画面に戻り、1 号機のサーバー URL の下に親サイトとして設定した「site1」が表示されていることを確認します。
8. 「サーバー」画面の一覧にある他のサーバーに対しても同様の設定を行います。

以上で完了です。

9.2 DNS エイリアスの設定

サイト構成を行うと、最上位のレルムに自動的に DNS のエイリアスとして「sso.example.co.jp (ロードバランサーの FQDN)」が追加されます。ロードバランサーの FQDN をサブレルムの DNS エイリアスとして設定する場合は、以下の手順で変更します。

1. 「管理者コンソールのトップ画面」→「Top Level Realm」→「プロパティ」を開きます。
2. 「レルムまたは DNS のエイリアス」から「sso.example.co.jp (ロードバランサーの FQDN)」を削除し、「変更の保存」をクリックします。
3. 「管理者コンソールのトップ画面」→該当のサブレルム→「プロパティ」を開きます。
4. 「レルムまたは DNS のエイリアス」に「sso.example.co.jp (ロードバランサーの FQDN)」を追加し、「変更の保存」をクリックします。

以上で完了です。

10 マルチテナント

本章では、OpenAM のサブレルムに組織や企業を割り当て、マルチテナントに対応した認証サーバーとして OpenAM を設定する方法について説明します。

10.1 事前検討事項

マルチテナント対応の設定を行うためには、事前に各テナントに割り当てるドメイン名を決定しておく必要があります。例として「company1.example.com」いうドメインをサブレルムに割り当てるものと仮定します。

10.2 サブレルムの作成とドメイン名の割り当て

サブレルムを作成し、前節で用意しておいたドメイン名でサブレルムにログインできるように設定します。

まず、サブレルムを作成します。

1. OpenAM 管理コンソールの「アクセス制御」を開きます。
2. 「5.2 サブレルムの作成」の手順でサブレルムを作成し、「レルムまたは DNS のエイリアス」に先の項で用意しておいたドメイン名を追加します。

次に、OpenAM が発行する Cookie のドメインに「10.1 事前検討事項」で用意しておいたドメインを追加します。

1. OpenAM 管理コンソールの「設定」 →「グローバルサービス」 →「システム」タブ →「プラットフォーム」を開きます。
2. Cookie ドメインに、「10.1 事前検討事項」で用意しておいたドメインを追加します。この例の場合では「company1.example.com」となります。
3. 画面右上の「保存」をクリックします。

このままですと、「http://company1.example.com:8080/openam/」という URL で OpenAM にアクセスしても、トップレルムの DNS エイリアスに割り当てられたドメイン名の URL にリダイレクトされてしまいます。

そこで、「http://company1.example.com:8080/openam/」という URL で継続的に OpenAM にアクセスできるように、以下の設定を行います。

1. シングル構成の場合は OpenAM 管理コンソールの「デプロイメント」 →「サーバー」

の OpenAM サーバーを開きます。サイト構成などで複数の同構成の OpenAM サーバーが存在する場合は「設定」→「デフォルトサーバー」を開きます。

2. 「詳細設定」を開きます。
3. 以下のプロパティを追加します。

【プロパティ名】	【プロパティ値】
com.sun.identity.server.fqdnMap[company1.example.com]	company1.example.com

4. 画面右下の「変更の保存」をクリックします。
5. OpenAM を再起動します。

以上で完了です。

「http://company1.example.com:8080/openam/」という URL で OpenAM にアクセスし、トップレルムのドメインの URL(http://sso.example.co.jp:8080/openam/) にリダイレクトしないことを確認します。

11 セキュリティ設定

11.1 goto パラメータのドメイン制限

OpenAM は goto というクエリパラメータに URL を指定することで、ログイン成功後にクライアントを任意の URL に遷移させることができますが、この機能はフィッシング詐欺に利用される可能性があります。

そのため、goto パラメータに指定できる URL は制限することが推奨されます。ここでは、この制限を OpenAM 上で行う方法を説明します。

11.1.1 設定方法

1. OpenAM 管理コンソールにて、対象のレルム→「サービス」→「サービスの追加」を開きます。
2. 「Validation Service」を選択します。
3. 「Valid goto URL Resources」にて、以下の例のように URL を追加していきます。

【URL の分類】	【URL】
OpenAM 用 URL 例	http://sso.example.co.jp:8080/* http://sso.example.co.jp:8080/*?*
リバースプロキシ用 URL 例	https://rp.example.co.jp/* https://rp.example.co.jp/*?*

4. 「作成」ボタンをクリックします。
5. OpenAM を再起動します。

以上で完了です。

11.1.2 goto パラメータの注意事項

URL の記法はエージェントのポリシーと同じです。例えば、クエリストリングを含む URL を許可する場合は、以下のように 2 種類指定する必要があります。

- https://rp.example.co.jp/*
- https://rp.example.co.jp/*?*

特定のドメインを含む URL を goto パラメーターとして指定可能にする場合は、ワイルドカードとして「-*-」を指定します (OpenAM では One-Level Wildcard と呼びます)。 以下に例を示します。

- https://-*-.example.co.jp/*
  - スキームが https で、かつ「.example.co.jp」というドメインを含む URL を goto パラメーターに指定可能です。「https://www.example.co.jp」や「https://ap.sso.example.co.jp」などを指定することができます。

11.2 Cookie に Secure 属性を付与する

HTTPS 通信の場合のみ Cookie を送信するよう、OpenAM が発行する Cookie に Secure 属性を付与することができます。

この設定を行うと HTTP のアクセスでは動作しなくなるため、十分に注意してください。

設定方法は以下の通りです。

1. OpenAM 管理コンソールの「設定」→「デフォルトサーバー」→「セキュリティ」を開きます。
2. 「Cookie」の「セキュリティー保護された Cookie」にチェックします。
3. 「変更の保存」ボタンをクリックします。
4. OpenAM を再起動します。

以上で完了です。

11.3 Cookie に HttpOnly 属性を付与する

JavaScript が Cookie を操作できないように、OpenAM が発行する Cookie に HttpOnly 属性を付与することができます。

ただし、XUI が有効な場合 (デフォルト：有効) は、セッション用のクッキー (デフォルト：iPlanetDirectoryPro) に対しては HttpOnly 属性は付与されませんことをご留意ください。

設定方法は以下の通りです。

1. OpenAM 管理コンソールの「設定」→「デフォルトサーバー」→「詳細設定」を開きます。
2. 「com.sun.identity.cookie.httponly」プロパティーの値を「true」に変更します。
3. 画面右下の「変更の保存」ボタンをクリックします。
4. OpenAM を再起動します。

以上で完了です。

11.4 モジュールベースの認証

OpenAM はリクエストパラメーターに「認証モジュール名」や「認証レベル」をセットして特定の認証方法を指定してログインできます。

この方法はセキュリティ上の問題が発生する可能性があります。例えば多要素認証を使用する (認証連鎖で複数の認証モジュールを設定) ケースです。認証連鎖で「データストア認証モジュール」「HOTP 認証モジュール」を Require と設定した場合、 通常は両方の認証モジュールの認証が必要です。しかし、認証方法として「データストア認証」を指定することで ID/パスワードだけでログインできます。

「モジュールベースの認証」を “無効” にすると、これらの指定を無効化します。具体的には次のような認証が不可となります。

- OpenAM ログイン時に「認証モジュール名」を指定してのログイン
- OpenAM ログイン時に「認証レベル」を指定してのログイン
- 「認可」の「ポリシーセット」の「条件」の設定で「認証モジュール」や「認証レベル」を使用したセッションアップグレード (追加の認証)

一般的な OpenAM の運用としてログイン時に特定の認証モジュールや認証レベルを指定可能とする必要はありません。「認可」の設定で「認証レベル (以上)」を使用しない限り本設定は “無効” にすることを推奨します。

“無効” と設定した場合 ssoadm コマンドに影響がありますが、こちらには回避方法があります。詳細は「コマンドライン 利用手順書」を参照ください。

11.4.1 設定方法

1. OpenAM 管理コンソールにて、対象のレルム→「認証」→「設定」→「セキュリティ」を開きます。
2. 「モジュールベースの認証」のチェックを外します。
3. 「変更の保存」を押します。

12 Cookie に SameSite=None 属性値を付加する設定

2020 年から主要なブラウザ (Chrome、Edge、Firefox) において Cross-site request forgery(CSRF) 攻撃を防ぐために、クロスサイト (スキーム、またはドメインをまたぐ) リクエストでの Cookie セットに関するデフォルトの振る舞いが変更になりました。この変更によって、クロスサイトで URL 遷移を行う処理中に意図した Cookie がセットされずに問題が発生する場合があります。

参考情報

- https://developers-jp.googleblog.com/2019/11/cookie-samesitenone-secure.html
- https://www.osstech.co.jp/support/am2020-1-1

12.1 影響を受ける可能性がある構成

- Agent でクロスドメイン SSO を利用しており、ポリシーで一部の URL に対してより強固な認証を定義しており、かつ XUI を無効化している
- SAML2 IdP として動作しており、認証コンテキストクラスでより強固な認証を定義しており、かつ XUI を無効化している
- SAML2 認証を利用している
- SAML2 SP として動作している
- SAML2 シングルログアウトを利用している

12.2 対応方法

OpenAM の発行する Cookie に [SameSite=None; Secure] 属性を付加する設定を行い、旧来通りブラウザに Cookie をセットできるようにします。[SameSite=None; Secure] は OpenAM を HTTPS で利用している必要があります。

この設定が可能なのは、OpenAM14-14.0.0-7 以降のパッケージです。必要に応じて OpenAM パッケージをアップデートして下さい。

12.2.1 OpenAM 設定

12.2.1.1 Cookie に Secure 属性を付与する

SameSite 属性の付与には Secure 属性も追加する必要があります。Secure 属性の追加方法は Cookie に Secure 属性を付与するを参照してください。

12.2.1.2 全ての Cookie に SameSite=None を付加する設定方法

この設定の場合は、OpenAM が発行する全ての Cookie に SameSite=None が付加されます。

1. OpenAM 管理コンソールの「設定」 → 「デフォルトサーバー」 →「セキュリティ」 →「Cookie」タブを開きます。
2. 「デフォルト SameSite 値」プルダウンメニューから「None」にします。
3. 画面右下の「変更の保存」ボタンをクリックします。
4. OpenAM を再起動します。

12.2.1.3 指定した Cookie にのみ SameSite=None を付加する設定方法

この設定は上の「デフォルト SameSite 値」設定より優先されます。「デフォルト SameSite 値」を「設定しない」に設定し、「Cookie 毎の SameSite 値」を設定した場合、設定の Cookie 以外には SameSite=None が付加されません。

1. OpenAM 管理コンソールの「設定」 → 「デフォルトサーバー」 →「セキュリティ」 →「Cookie」タブを開きます。
2. 「Cookie 毎の SameSite 値」に初期値 [amlbcookie=None,authenticationStep=None] のように、SameSite 属性を付加する Cookie 名と値を [=] でペアとし、複数の Cookie を指定する場合は [,] カンマ区切りで入力します。
3. 画面右下の「変更の保存」ボタンをクリックします。
4. OpenAM を再起動します。

12.2.2 SameSite=None 属性に非互換のブラウザを除外する設定

一部の古い非互換ブラウザでは SameSite=None 属性に対応しておらず正常に動作しません。そのため非互換ブラウザを SameSite 属性セット対象から除外する設定が必要です。OpenAM では除外する条件を正規表現で記載する機能が用意されており、デフォルトで定義されております。お使いのクライアント環境で問題がある場合は、適宜ブラウザ種別の正規表現を追加又は削除して対応下さい。

- 設定ファイル

```
/opt/osstech/etc/tomcat/webapps/openam/WEB-INF/classes/
 SameSiteIncompatibleClient.properties
```

- 非互換ブラウザ名の正規表現例

```
iOS_12=\\(iP.+; CPU .*OS 12[_\\d]*.*\\) AppleWebKit\\/
macOS_10_14_Safari=\\(Macintosh;.*Mac OS X 10_14[_\\d]*.*\\)
 AppleWebKit\\/.*Version
```

設定の反映には osstech-tomcat サービスの再起動が必要です。

13 ログ

本章では、OpenAM が出力するログについて説明します。

13.1 標準ログ (監査ログ)

13.1.1 出力内容

- 出力場所：/opt/osstech/var/lib/tomcat/data/openam/openam/log
- OpenAM の標準ログ (監査ログ) が出力されます。ファイルの種類は以下の通りです。

【ファイル】	【内容】
access.csv	あらゆるアクセスリクエストに対して、誰が、いつ、何を、そして結果といった情報を記録したログ。
activity.csv	エンドユーザに起因するセッションオブジェクトの状態変更（生成、更新、削除）のログ。
authentication.csv	いつどのようにユーザーが認証されたか、また、関連するイベントについてのログ。
config.csv	設定が変更された際に、いつ誰によってなされたかのログ。

13.2 統計情報ログ

13.2.1 出力内容

- 出力場所：/opt/osstech/var/lib/tomcat/data/openam/openam/stats
- 統計情報が出力されます。標準機能で以下の統計情報をログ出力可能です。
  - 最大ユーザーセッション数
  - ユーザーデーターストア (OpenLDAP) のキャッシュヒット数
  - ポリシー情報の利用状況
  - 認可処理数
- ログは指定間隔 (秒) で定期的に取得可能です。また、次の項で示す手順により、統計情報ログの出力を無効化することも可能です。

13.2.2 統計情報ログの無効化/有効化

統計情報ログの無効化/有効化の手順について説明します。

1. OpenAM 管理コンソールの「デプロイメント」 → 「サーバー」 →対象サーバー→「セッション」を開きます。
2. 「統計情報」の「状態」の鍵をクリックして変更可能状態にします。
3. 「状態」のドロップダウンリストから選択します。
   - 統計情報ログを出力する場合は「ファイル」を選択します。
   - 統計情報ログを出力しない場合は「オフ」を選択します。
4. 「変更の保存」をクリックします。
5. サーバー一覧に表示されている全てのサーバーに対して同様の設定を行います。

以上で完了です。

13.3 デバッグログ

13.3.1 出力内容

- 出力場所：/opt/osstech/var/lib/tomcat/data/openam/openam/debug
- OpenAM のデバッグログが出力されます。OpenAM に問題が発生した際はログレベルを変更し、デバッグログを取得します。

13.3.2 ログレベルの変更

デバッグログのログレベルについて説明します。以下の 3 つのログレベルを設定できます。

【ログレベル】	【出力内容】
エラー	エラーメッセージだけがログに書き込まれます。
警告	エラーメッセージおよび警告メッセージがログに書き込まれます。
メッセージ	エラーメッセージ、警告メッセージ、および情報メッセージがログに書き込まれます。最も情報量の多いログとなります。

問題が発生した場合は、ログレベル「メッセージ」のログを取得することで原因を調査することが可能です。ここでは、OpenAM のメッセージレベルのログ (デバッグ) を取得する方法を説明します。ログレベルの変更は全ての OpenAM サーバーに対して実施する必要が

あります。

1. OpenAM 管理コンソールの「デプロイメント」 → 「サーバー」 →対象サーバー→「一般」→「デバッグ」を開きます。
2. 「デバッグレベル」を「メッセージ」へ変更します。
3. 「変更の保存」ボタンをクリックします。
4. 同様の手順で、サーバー一覧に表示されている全てのサーバーのログレベルを変更します。

以上で完了です。

13.4 OpenDJ ログ

- 出力場所：/opt/osstech/var/lib/tomcat/data/openam/opends/logs
- OpenAM の設定情報を保存する内蔵 LDAP である OpenDJ のログが出力されます。OpenAM 管理コンソールから設定を変更した際や、複数の OpenAM 間で設定情報が複製 (レプリケーション) された際にログが出力されます。

13.5 Tomcat ログ

- 出力場所：/opt/osstech/var/log/tomcat
- Tomcat のログが出力されます。

14 起動状態の監視

本章では、OpenAM の起動状態の監視の方法について説明します。
OpenAM には本目的用の JSP が用意されており、これに HTTP GET リクエストを送信することで起動状態を監視できます。 送信する HTTP GET リクエストとレスポンスは以下のようになります。

リクエスト

```
GET /OpenAM デプロイ名/isAlive.jsp HTTP/1.0
```

レスポンス (正常な場合)

```
HTTP/1.1 200 OK
Server: Apache-Coyote/1.1
Set-Cookie: JSESSIONID=61E6019B55F9CC8A6FB731D6B37F5472; Path=/OpenAM デプロイ名
Content-Type: text/html
Content-Length: 112
Date: Wed, 11 Dec 2019 00:49:09 GMT
Connection: close

<html>
<head>
    <title>OpenAM</title>
</head>
<body>
<h1>Server is ALIVE: </h1>
</body>
</html>
```

異常が発生している場合には、上記の「HTTP/1.1 200 OK」以外のレスポンスが返ります。

15 トラブルシューティング

15.1 インストール時のエラーメッセージ

エラーメッセージ	Configuring Directory Server ….. Failed to create debug directory
エラー詳細	OpenAM が自分のホスト名を名前解決できない場合に表示されます。DNS サーバの設定や/etc/hosts の内容を再確認し、OpenAM サーバが自分のホスト名を名前解決できるか確認してください。

15.2 設定が複製されない

エラーメッセージ	ー
エラー詳細	OpenAM はデフォルトで設定の変更履歴を 3 日分しか保持しません。そのため、複数台構成で特定の OpenAM サーバを停止させ続けた場合、設定のレプリケーションが動作しなくなる場合があります。 特定の OpenAM サーバを長時間停止させている場合には設定変更を行わないでください。

15.3 ログイン時のエラーメッセージ

エラーメッセージ	「そのような組織はみつかりません」
エラー詳細	OpenAM の URL にアクセスした際に、ログイン画面が表示されることもなくこのメッセージが表示された場合は、レルムの DNS エイリアス設定が正しくありません。同一の FQDN が複数のレルムに登録されています。複数のレルムに同一の DNS エイリアスを登録することはできないため、レルムの設定を変更してください。

エラーメッセージ	「ユーザーにはこの組織におけるプロファイルがありません。システム管理者に連絡してください。」
エラー詳細	ユーザーデータストアにおけるユーザー検索に失敗しています。以下のような場合にこのメッセージが表示されます。
発生パターン 1	以下の前提条件の場合に発生します。 ＊認証モジュールで、ユーザーデーターストアとは別の LDAP を参照している。 ＊認証モジュールの設定オプションで「ユーザー DN をデータストアに返す」が有効になっている。 この場合、認証モジュールでのユーザー認証成功後にユーザーの DN をデータストア検索処理に渡し、データストア側でユーザー検索を行いますが、存在しない DN を渡されるためユーザーエントリを見つけることができず、エラーが表示されます。

発生パターン 2	以下の前提条件の場合に発生します。 ＊認証モジュールで、ユーザーデーターストアと同じ LDAP を参照している。 ＊ユーザーデータストアのオプションで、「LDAP ピープルコンテナネーミング属性」「LDAP ピープルコンテナ値」を設定している。 ＊「LDAP ピープルコンテナ値」で指定した OU の配下にさらに OU(仮に、サブピープルコンテナ OU と呼ぶものとする) を追加し、サブピープルコンテナ OU 配下にユーザーを登録している。 この場合、ログイン後の一部 LDAP 検索処理で、ユーザー ID と「LDAP ピープルコンテナ値」で指定した OU から成るベースオブジェクトのみを対象とした検索処理が実行されます。そのため、サブピープルコンテナ OU 配下のエントリの場合は検索に失敗し、結果的にログインに失敗してエラーメッセージが表示されます。

エラーメッセージ	502 Proxy Error、503 Service Temporarily Unavailable
エラー詳細	OpenAM(Tomcat) の最大プロセス数に対して Apache の最大プロセス数が多い場合、OpenAM 側の処理が追い付かず、エラーが発生する場合があります。Apache と OpenAM の最大プロセス数は同じ値になるよう設定してください。

エラーメッセージ	all: process_request(): size of render data buffer too small for pointer

エラー詳細	OpenAM に渡されたクエリストリングが長すぎる (2048 バイト以上) 場合に表示されます。goto パラメータ等に長い URL が渡されると発生するので、長い URL を渡さないようご注意ください。

15.4 レルム

エラー内容	サブレルムで指定した DNS エイリアスでログイン画面にアクセスし認証に成功後、トップレルムの FQDN が返される。
エラー詳細	DNS エイリアス名に誤りがある可能性があります。OpenAM サーバーの FQDN、サイト構成にした場合はサイトの FQDN が DNS エイリアスに正しく設定されているかご確認ください。

16 改版履歴

- 2019 年 12 月 11 日 リビジョン 1.0
  - 初版作成
- 2021 年 3 月 1 日 リビジョン 1.1
  - 「12 Cookie に SameSite=None 属性値を付加する設定」を追加
- 2021 年 3 月 2 日 リビジョン 1.2
  - アダプティブリスク認証モジュールの新機能を追記
- 2021 年 3 月 31 日 リビジョン 1.3
  - 「セキュリティ設定」に「モジュールベースの認証」を追記
- 2021 年 4 月 28 日 リビジョン 1.4
  - 設定情報ディレクトリのパスを修正
- 2021 年 5 月 18 日 リビジョン 1.5
  - TOTP のパラメータに関する記述を追記
  - ForgeRock Authenticator (OATH) の利用手順を追記
- 2021 年 5 月 27 日 リビジョン 1.6
  - マルチテナントの場合の Cookie の設定方法を修正
- 2021 年 6 月 1 日 リビジョン 1.7
  - DesktopSSO で使用する暗号化方式を AES256-SHA1 に変更