

IdPv4アップデートに関する情報

✔ Shibboleth SPのアップデートに関してはこちらをご覧ください。⇒[SPv3アップデートに関する情報](#)

✔ V3内のアップデートに関してはこちらをご覧ください。⇒[IdPバージョン3アップデートに関する情報](#)

- バージョン共通
 - アップデートの手順
- IdP 4.1.x から IdP 4.1.x へアップデートする場合の注意点
- IdP 4.0.x から IdP 4.1.x へアップデートする場合の注意点
- IdP 3.x.x から IdP 4.x.x へアップグレードする場合の注意点
 - Javaのバージョン
 - Javaコンテナ
 - 設定ファイルの移行について

バージョン共通

アップデートの手順

shibboleth-identity-provider-4.x.x.tar.gzパッケージを展開したディレクトリで、以下のコマンドで設定ファイルの変更点を確認し、適宜反映した上で、アップデートを実行します。

i /opt/shibboleth-idp/以下に存在しないファイル/ディレクトリはアップデート時に自動的に作成されますが、インストール後修正したファイルのほか、修正していないファイルも一切上書きはされませんので、新バージョンの内容を適宜反映してください。各自で修正していないファイルはdist/以下のファイルで上書きする、各自で修正したファイルは新バージョンでの変更点をマージする形になります。

反映しない場合、旧来の機能は変わらず動作することが保証されますが、新バージョン以降の新機能が（デフォルトで有効な場合と有効化した場合いずれも）正しく動作することが保証されません。このため、将来的な新機能利用も見据えて、アップデート後でもかまいませんのでなるべく早く反映するようにしてください。

特に、edit-webapp/WEB-INF/web.xmlは頻繁に更新されていますので、当該ファイルが存在する場合は[web.xmlの変遷](#)を参照の上必要な変更を反映してください。



uApproveJPをインストールしている場合はsystem/以下の修正が元に戻ってしまうので、アップデート前に展開したディレクトリの当該ファイルを修正した上でアップデートを行うのがお勧めです。system/以下の修正箇所をパッチ形式にしたものを置いておきますので、展開したディレクトリにて適用してください。

[uapprovejp3-system.patch](#)

```
$ patch -p0 < .../uapprovejp3-system.patch
```

ただし、4.1.0以降ではファイルがライブラリの中に入っております。下記の手順で展開したディレクトリの webapp/WEB-INF/lib/idp-conf-impl-4.1.x.jarの中にある当該設定ファイルを更新してJARファイルを上書きしてください。

webapp/WEB-INF/lib/idp-conf-impl-4.1.2.jar
を適当な空のディレクトリでunzipしまして、中の
net/shibboleth/idp/conf/services-system.xml
につきましてid="shibboleth.AttributeFilterService"のbean定義の<constructor-arg name="strategy">を以下のように変更、

```
<constructor-arg name="strategy">
  <bean class="jp.gakunin.idp.attribute.filter.spring.impl.AttributeFilterServiceStrategy"
    depends-on="shibboleth.AttributeRegistryService"
    p:transcoderRegistry-ref="shibboleth.AttributeRegistryService"
    id="ShibbolethAttributeFilter"/>
</constructor-arg>
```

同じく中の
net/shibboleth/idp/flows/intercept/attribute-release-beans.xml
につきまして、id="IsConsentRequiredPredicate"のbean定義のclassを以下のように変更

```
<bean id="IsConsentRequiredPredicate"
  class="jp.gakunin.idp.consent.logic.impl.IsConsentRequiredPredicate" />
```

してください。再度全体をzipしてidp-conf-impl-4.1.2.jarというファイル名にして元のファイルを上書きしてください。

まず、配布物として旧バージョンからの変更点を確認します。

```
# diff -rb -x LICENSE.txt -x bin -x credentials -x doc -x idp_ant*.log -x logs -x metadata -x system /opt/shibboleth-idp/dist/ .
```

Jettyを停止後、インストールスクリプトを実行します。

```
# systemctl stop jetty
# bin/install.sh -Didp.conf.credentials.filemode=640 -Didp.conf.credentials.group=jetty
```

```
Buildfile: /root/shibboleth-identity-provider-4.1.2/bin/build.xml
```

```
install:
Source (Distribution) Directory (press <enter> to accept default): [/root/shibboleth-identity-provider-4.1.2] ?
[Enter] ←入力なし
Installation Directory: [/opt/shibboleth-idp] ?
[Enter] ←入力なし
INFO [net.shibboleth.idp.installer.V4Install:162] - Update from version 4.0.1 to version 4.1.2
INFO [net.shibboleth.idp.installer.BuildWar:103] - Rebuilding /opt/shibboleth-idp/war/idp.war, Version 4.1.2
INFO [net.shibboleth.idp.installer.BuildWar:113] - Initial populate from /opt/shibboleth-idp/dist/webapp to /opt/shibboleth-idp/webpapp.tmp
INFO [net.shibboleth.idp.installer.BuildWar:92] - Overlay from /opt/shibboleth-idp/edit-webapp to /opt/shibboleth-idp/webpapp.tmp
INFO [net.shibboleth.idp.installer.BuildWar:125] - Creating war file /opt/shibboleth-idp/war/idp.war

BUILD SUCCESSFUL
Total time: 8 seconds
#
```

インストールスクリプトが正常に終了したらJettyを起動します。

```
# systemctl start jetty
```

アップデート後、以下のコマンドでバージョンが更新されていることを確認してください。

```
$ /opt/shibboleth-idp/bin/status.sh | grep idp_version
idp_version: 4.1.2
```

IdP 4.1.x から IdP 4.1.x へアップデートする場合の注意点

このバージョンに限ったことではありませんが、Windows MSI installerを使っている場合アップデート時に最大メモリ設定が変更されるという情報があります。Windowsをお使いの場合はアップデート後にチェックするようにしてください。
詳細: ([Shibboleth Wiki](#)) [WindowsInstallation](#) の "Always Check"

IdP 4.0.x から IdP 4.1.x へアップデートする場合の注意点

/opt/shibboleth-idp/system/ディレクトリから多くのファイルが削除されました。
3.1およびそれ以前の配布物のweb.xmlをedit-webapp/WEB-INF/に配置しており、その後の必要な更新を行っていない場合にはsystem/以下への参照が含まれているためエラーになるという情報があります。共通手順にも含めてありますが[web.xmlの変遷](#)を参照し必要な変更を取り込んでください。特にDefaultPlaceholderFileSystemXmlWebApplicationContextという文字列が含まれる場合はエラーになる可能性があります。

また、uApproveJPのインストール・アップデート方法に変更が生じております。詳しくは上記共通手順の注意書きをご参照ください。

4.1からモジュールおよびプラグインの機能が導入されております。以前のバージョンからアップデートした場合は自動的に有効化等されるため問題ありませんが、4.1.xおよびそれ以降を新規インストールした場合、モジュールを有効化しないと使えない機能がございますのでご注意ください。現在有効化されているモジュールは以下のコマンドで確認できます：

```
# /opt/shibboleth-idp/bin/module.sh -l
```

モジュールの例: [属性送信同意画面の設定\(IdPv4\)](#)

4.1からはidp.propertiesの記述が別の複数のファイルに分割記述されるようになりました。ただし4.0.x以前からのアップデートの場合はそれらの記述は適切に無視されます。4.1流にプロパティを分散記述する場合は、idp.propertiesにある重複を排除した上でidp.searchForPropertiesをtrueに設定してください。詳しくはdist/conf/idp.propertiesの冒頭のコメントを参照してください。

4.1の書式に書き換える例: <https://shibboleth.atlassian.net/wiki/spaces/KB/pages/1469908146/Example+4.1+Upgrade>

4.0.1までには属性値ハッシュ計算の問題で属性値が複数ある場合変更されていないのに再同意を求める問題 (idp.consent.compareValues=trueの場合の問題) があるようです。本運用環境を移行する場合にはご注意ください。4.1.0で修正されました。

<https://issues.shibboleth.net/jira/browse/IDP-1660>

LDAPを使った属性取得の例がconf/examples/の下に移動しました。(conf/examples/attribute-resolver-ldap.xml)

IdP 3.x.x から IdP 4.x.x へアップグレードする場合の注意点

下記ページの通りバージョン4.0.0が2020-03-11にリリースされました。下記の注意事項および手順をご確認の上、バージョンアップを行ってください。

<https://wiki.shibboleth.net/confluence/display/NEWS/2020/03/11/Shibboleth+Identity+Provider+V4.0.0+Released>

なお、バージョン3のEoLは2020年末です。未だバージョン3を運用している機関様におかれましては至急V4へのバージョンアップのスケジューリングをお願いします。

<http://shibboleth.net/pipermail/announce/2020-March/000213.html>

学認が提供している技術ガイドも順次更新していきます。uApproveJPはバージョン4対応版を公開中です。TiqrShib等のNIIが提供しているIdPプラグインは現在バージョン4対応版の公開準備中です。

また、アップデート時のノウハウなど情報をお持ちの方がいらっしゃいましたら、[情報交換ML](#)等で共有いただけましたら幸いです。

Javaのバージョン

- Java 11以上が必須です。
- 公式サポートはJavaのLTSのみなので、2020年9月現在ではJava 11のみとなります。

Javaコンテナ

- Shibboleth IdP V3で利用されていたTomcat 7はサポート対象外となりました。
- Shibboleth開発元がサポートしているのはJetty 9.4、またはTomcat 9以上 (参照: [SystemRequirements](#))
 - 開発元はJettyを推奨しています
 - 少なくともServlet 3.1をサポートしたJavaコンテナでなければ動作しません

設定ファイルの移行について

- 名前空間のフラット化が強制され、プレフィックスありの（v2由来の名前空間を使用した）設定ファイルが使用できなくなります
- IdP v3.4ではフラット化していない場合、DEPRECATEDのwarningとしてログに出力されます
 - 3.4.0以降も細かい改善が続けられていますので、v3.4系の最新版で確認してください
 - 静的に解析しているものと動的に解析しているものがあり、3.4系の最新版でしばらく動かして一通り認証フローの動作確認を行ってみる必要があります
 - フラット化の対応手順はこちら
⇒<https://meatwiki.nii.ac.jp/confluence/x/F4Z7AQ>
- その他にも、DEPRECATEDのwarningとなる対象が複数あります
 - 影響の大きいものとして、<Dependency>要素はDEPRECATEDです
 - 内容によって<InputAttributeDefinition>もしくは<InputDataConnector>で置き換えてください。例えば

```
<resolver:Dependency ref="myLDAP" />
```

は（myLDAPがDataConnectorとして定義されていると仮定して）以下で置き換えます。attributeNamesには使用する属性名を（複数ある場合はスペース区切りで）列挙してください:

```
<InputDataConnector ref="myLDAP" attributeNames="..." />
```

また、

```
<resolver:Dependency ref="eduPersonAffiliation" />
```

は（eduPersonAffiliationがAttributeDefinitionで定義されていると仮定して）以下で置き換えます:

```
<InputAttributeDefinition ref="eduPersonAffiliation" />
```

- LegacyPrincipalConnector
以下のwarningが出る場合は

```
WARN [DEPRECATED:118] - Spring bean 'c14n/LegacyPrincipalConnector', (c14n/subject-c14n.xml): This will be removed in the next major version of this software; replacement is <remove>
```

conf/c14n/subject-c14n.xml の以下の部分を削除してください。

```
<!--
This is installed to support the old mechanism of using PrincipalConnectors in the attribute resolver
to map SAML Subjects back into principals. If you don't use those (or this is a new install) you can
remove this.
-->
<ref bean="c14n/LegacyPrincipalConnector" />
```

フラット化への対応により<PrincipalConnector>は削除されているはずですので、当該部分を削除しても問題ありません。

- 詳細は**Shibbolethの本家の情報**を参照してください。
 - 変更のある要素への置き換え方法へのリンクを含めて記載されています
- **Shibboleth IdP V4への準備として、V3.4系最新版（3.4.8）にてwarningが出なくなるまで設定ファイルを修正することを推奨します**
- すでにフラット化とDEPRECATED対応済みの学認テンプレートを**配布中**ですのでこちらも参考にしてください。ただし最新版attribute-resolver.xmlテンプレートにはV4向けのAttribute Registry対応も入っておりますので、本ページ（V3からのアップグレード）の文脈では参考にしないでください。具体的に言うと、<AttributeEncoder>の行は消さないでください。あくまでフラット化とDEPRECATEDな要素・属性の置き換えの参考として参照してください。
- LDAP周りで、使用するライブラリがJNDIからUnboundIDに変更になることにより以下の通り設定によってはV4への移行後にエラーになる可能性が若干ございます。
 - LDAPのURL（ldap.propertiesの idp.authn.LDAP.ldapURL）がスラッシュ(/)で終わる場合はうまく動作しませんのでスラッシュを除去してください。
 - 検索フィルタ（ldap.propertiesの idp.attribute.resolver.LDAP.searchFilter）に空白が含まれるとActive Directory等との連携に問題が発生する場合がございますので、空白を除去してください。
 - **LDAPConnector**もしくは**JAASAuthnConfiguration**にてJNDI特有のプロパティを使っている場合問題が発生します。プロパティ名に"jndi"を含むものもしくは下記"binary"にご注意ください。代替のものに置き換えてください。
 - 特にバイナリ属性（objectGUID等）については3.4.5よりLDAPConnectorにて<BinaryAttributes>要素がサポートされておりますのでこれで代替してください。プロパティ名は"java.naming.ldap.attributes.binary"となっております。
 - JNDI特有のプロパティとは、例えば、attribute-resolver.xmlの<DataConnector>に以下のような指定がある場合該当します。

```
<LDAPProperty name="com.sun.jndi.ldap.connect.timeout" value="500"/>
```

その他、ldpのディレクトリの中やJavaのディレクトリの中に jndi.properties というファイルが存在しその中で指定しているという場合があるようですのでご注意ください。

- **上記問題の対象の場合は、V3.4系最新版で以下に記載されている手順でUnboundIDを使うようにして動作確認することを推奨します**
 - V3.4.4以降で以下の行をldap.propertiesに追加すればJNDIでなくUnboundIDを使うようになります。

```
idp.ldap.provider=org.ldap.provider.unboundid.UnboundIDProvider
```

- V3系でUnboundIDが使われていることの確認は、idp.propertiesに

```
idp.loglevel.ldap=INFO
```

を追加してログレベルを変更の上再起動し、下記のように"Setting ldap provider to"が UnboundIDProvider になっていることを確認してください。

```
2020-03-02 09:46:50,446 - - INFO [org.ldap.DefaultConnectionFactory:192] - Setting ldap provider to org.ldap.provider.unboundid.UnboundIDProvider
2020-03-02 09:46:50,453 - - INFO [org.ldap.DefaultConnectionFactory:192] - Setting ldap provider to org.ldap.provider.unboundid.UnboundIDProvider
2020-03-02 09:46:50,453 - - INFO [org.ldap.DefaultConnectionFactory:192] - Setting ldap provider to org.ldap.provider.unboundid.UnboundIDProvider
```

確認後、idp.propertiesに追加した行を削除してログレベルを元に戻してください。

- 詳細はこちら: [LDAPonJava \(v4\)](#) および [LDAPonJava>8 \(v3\)](#)