（IdP運用担当者向け）attribute-filterの自動生成機能を使う

目次

このページに書いてある手順に従うことにより、学認に参加しているSPについて、attribute-filter.xml を手で修正することなく、学認申請システムでの選択に従ってIdPから属性送信を行うことができるようになります。

0. 機能および前提条件

属性送信対象SPの追加・削除のみが操作可能で、個々の属性を送信するか否かをIdP管理者が操作することはできません。SPが学認に申請した通りに属性を送ります。

各SPが要求する属性はこちら⇒SP接続情報

現在提供しているのは以下の3種類のattribute-filter.xml設定ファイルです:

1. （uApproveJPをインストールしていない）Shibboleth IdP 4.x向け (target=idp4)
    - フラット化されたもので3.2.x以降に適用可能です
2. uApproveJPプラグインがインストールされたShibboleth IdP 3.x向け (target=uapprovejp3)
3. （uApproveJPをインストールしていない）Shibboleth IdP 3.x向け (target=idp3)
4. uApprove.jp 2.2.1がインストールされたShibboleth IdP 2.4.x向け (target=uapprovejp221)

適切なパラメーターを付与してご利用ください。

以下のマニュアルではIdPv4向けの設定をご紹介しています。

任意属性については uApproveJP の機能により利用者が送信するかどうかを選択できます。

Shibboleth IdP 2.4.4およびそれ以降をご利用ください。それ以前のバージョンでは、リモートからのattribute-filterの取得停止、およびメタデータ取得停止の問題が発生することが確認されております。

- 学認申請システムが生成したattribute-filterのattributeIDに記載されている値と、IdPの /opt/shibboleth-idp/conf/attribute-resolver.xml に記載されているidの値が一致していること
  V3版のattribute-resolver.xmlテンプレートをアップグレードして利用していることを想定しています。⇒テンプレート
  テンプレートV4版のattributeIDの変更（organizationName→o等）へは近日中に対応の見込みです。それまでは暫定的にattribute-resolver.xmlおよびattribute-filter.xmlの設定を元の値に修正してご利用ください。

  attribute-resolver.xmlテンプレートを更新する際には、idの変更を含む場合がありますのであわせてattribute-filter.xmlの確認および修正も行ってください。

- 学認申請システムが提供するattribute-filterの提供元を指定するために、IdPの**申請書ベースID**が必要です。申請書ベースIDは、学認申請システムの承認済みIdPの詳細画面から確認することができます。
- 例として示す設定ファイルの書き方は既存のattribute-filter.xmlと、学認申請システムが提供するattribute-filterの両方を読むことを想定した内容となっています。
- 下記の設定を行うかどうかに関わらずデフォルトでは既存のattribute-filter.xmlも定期的に再読み込みされます（デフォルト15分おきで、idp.service.attribute.filter.checkIntervalにより変更可）。更新途中のattribute-filter.xmlを放置するとそれを読み込んで誤動作する可能性がありますのでご注意ください。

1. IdPにattribute-filterの自動読み込み設定を行う

IdPに対して以下のどちらかの手順を行ってください。全てが自動化されることに不安がある場合は、後者の方法を選択して間に検証手順を挟んでください。

a) 学認申請システムが生成したattribute-filterを直接読み込む方法

1. /opt/shibboleth-idp/conf/services.xml に学認申請システムからattribute-filterをダウンロードするための設定を追加します。 c:url には https://office.gakunin.nii.ac.jp/ProdFed/export/attribute_filter/申請書ベースID 、 c:backingFile にはダウンロードしたファイルを保存するためのパスを指定してください。例として、学認IdP（申請書ベースID:PI0025JP）のattributer-filterをIdPのconfディレクトリ配下にダウンロードするための設定を以下に示します。

```
    ...
    <util:list id ="shibboleth.AttributeFilterResources">
        <value>%{idp.home}/conf/attribute-filter.xml</value>
        <bean class="net.shibboleth.ext.spring.resource.FileBackedHTTPResource"
            c:client-ref="GakuNinAttributeFilterHttpClient"
            c:url="https://office.gakunin.nii.ac.jp/ProdFed/export/attribute_filter/PI0025JP?target=idp4"
            c:backingFile="/opt/shibboleth-idp/conf/attribute-filter-fromoffice-backing.xml" />
    </util:list>
    <bean id="GakuNinAttributeFilterHttpClient" parent="shibboleth.HttpClientFactory"
        p:connectionTimeout="%{idp.httpclient.connectionTimeout:PT1M}"
        p:connectionRequestTimeout="%{idp.httpclient.connectionRequestTimeout:PT1M}"
        p:socketTimeout="%{idp.httpclient.socketTimeout:PT1M}"
        p:maxConnectionsTotal="%{idp.httpclient.maxConnectionsTotal:100}"
        p:maxConnectionsPerRoute="%{idp.httpclient.maxConnectionsPerRoute:100}" />

    <util:list id ="shibboleth.NameIdentifierGenerationResources">
    ...
```

   PI0025JP の部分を置き換えてください。
2. Jettyを再起動して、設定を反映します。

```
$ sudo systemctl restart jetty
```

> この方法では、Jettyを非root権限で動かしている場合は、confディレクトリに新規ファイルを書き込めない可能性があります。その場合は c:backingFile に指定するファイルをあらかじめ作成し、ownerを変更するなどの事前準備が必要となります。
> 例（ユーザjettyで動かしている場合）：
>
> ```
> $ ls -ld /opt/shibboleth-idp/conf
> drwxr-xr-x 2 root root 4096 10月  8 17:50 /opt/shibboleth-idp/conf
>
> $ sudo touch /opt/shibboleth-idp/conf/attribute-filter-fromoffice-backing.xml
> $ sudo chown jetty:jetty /opt/shibboleth-idp/conf/attribute-filter-fromoffice-backing.xml
> ```

b) 学認申請システムが生成したattribute-filterをローカルにダウンロードした上で読み込む方法

1. 学認申請システムから自機関のIdP向けのattribute-filterをダウンロードします。wgetコマンドの引数には https://office.gakunin.nii.ac.jp/ProdFed/export/attribute_filter/申請書ベースID?target=idp4 を指定します。例として、学認IdP（申請書ベースID:PI0025JP）のattributer-filterをダウンロードするときのコマンドを以下に示します。PI0025JP の部分を置き換えてください。

```
$ wget https://office.gakunin.nii.ac.jp/ProdFed/export/attribute_filter/PI0025JP?target=idp4
```

2. ダウンロードしたattribute-filterを任意のディレクトリに移動します。例として、IdPのconfディレクトリに移動し、ファイル名を attribute-filter-fromoffice.xml に変更しています。

```
$ sudo mv PI0025JP¥?target=idp4 /opt/shibboleth-idp/conf/attribute-filter-fromoffice.xml
```

3. /opt/shibboleth-idp/conf/services.xml にダウンロードしたattribute-filterを読み込む設定を追加します。例として、上で配置したattributer-filterを（デフォルト15分おき(idp.service.attribute.filter.checkInterval)に）読み込むための設定を以下に示します。

```
    ...
    <util:list id ="shibboleth.AttributeFilterResources">
        <value>%{idp.home}/conf/attribute-filter.xml</value>
        <value>%{idp.home}/conf/attribute-filter-fromoffice.xml</value>
    </util:list>

    ...
```

4. Jettyを再起動して、設定を反映します。

```
$ sudo systemctl restart jetty
```

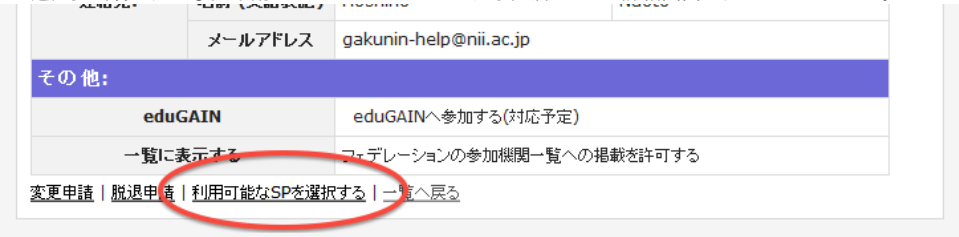
この方法ではattribute-filterダウンロードの部分は自動化されませんので、定期的に手動で実行するなり、後述2.の操作後に実行するなり、cronで実行するなりして、`attribute-filter-fromoffice.xml` を置き換えてください。

いずれの場合も、以下の手順により動作確認が完了したら、既存の attribute-filter.xml に設定されている重複した学認参加SPに対するフィルタ設定を削除しておくほうが混乱がないでしょう。

2. 属性送信するSPを追加・変更する

ここでの操作が影響するのはSPへ属性を送信するか否かのみです。SPによっては別途接続申し込みが必要なものがあり、これは個別に行っていただく必要があります。詳細は学認のSP一覧をご参照ください。

1. IdP運用担当者の方が学認申請システムにログインし、承認済みIdPの詳細画面を表示してください。



ページ最下部に「利用可能なSPを選択する」リンクがありますのでクリックしてください。

.

2. SP選択画面が表示されます。
まず、「特に指定しない（全てのSPで表示されます）」にチェックが入っていないことを確認してください。

ページ先頭にある説明書きと注意書きは、別の機能についてのものですので、無視してください。

このページでチェックを入れたSPに属性が送信されるようになりますので、慎重に行ってください。

.

3. 最後にページ最下部の「登録」ボタンをクリックしてください。

.

4. 上述の直接読み込むIdP設定の通りに行っていれば、最大でも15分後には選択したSPに属性を送信するようになります。

3. 注意事項

この自動生成では、IdP管理者が選択できるのはSP単位で、選択されたSPについては、SP管理者が申請した通りの属性を送信するようになります。つまりIdP版の場合は

- SP管理者が必須および任意と指定した属性は、Shibboleth IdPにて利用者に選択権のない（送信しないことを選択できない）属性となります。
- その他の属性は送信されません。

uApproveJP版の場合は

- SP管理者が必須と指定した属性は、uApproveJPにて利用者に選択権のない（送信しないことを選択できない）属性となります。
- SP管理者が任意と指定した属性は、uApproveJPにて利用者が送信するか否かを選択できる属性となります。
- その他の属性は送信されません。

特定のSPについて特定の属性を送信したくない（例えば任意属性となっているものを利用者への選択肢として表示させない）場合は、以下のような記述を attribute-filter.xml （上記手順で追加したファイルではなく既存のもののほうです）に追加してください。

```
<!--  Additional Deny Policy for <SP名> -->
<AttributeFilterPolicy id="DenyPolicyfor<SP名>">
    <PolicyRequirementRule xsi:type="Requester" value="<SPのentityID>" />

    <AttributeRule attributeID="<属性名>">
                <DenyValueRule xsi:type="ANY" />
    </AttributeRule>
</AttributeFilterPolicy>
```

また、本機能で自動化されるのはフィルタ設定部分のみですので、SPに適切に属性を送信するためには、別途attribute-resolver.xmlで適切な属性値を生成する必要がある場合があります。（例えば、電子ジャーナルサービスでのeduPersonEntitlement属性の common-lib-terms）
本機能を使用していて一部のSPにつながらない場合はattribute-resolver.xmlの設定をご確認ください。

4. その他

現在、学認申請システムによって自動生成されるフィルタの構成は、以下のようになっています。

1. チェックを入れたSP群のうち、そのメタデータに RequestedAttribute(AttributeConsumingService) が記載されていれば、その指定に従って送信。
   （SPを PolicyRequirementRule に列挙し、メタデータに従った PermitValueRule(xsi:type="uajpmf:AttributeUapprove" requestedOnly="true") で16属性列挙）
2. チェックを入れた個々のSPについて、かつ上記にあてはまらない場合は、学認申請システムに登録された属性を列挙して、送信。
   （必須は xsi:type="basic:ANY" として、オプショナルは xsi:type="uajpmf:AttributeUapprove" として。
   これによりオプショナル属性は利用者にチェックボックスが提示される。）

また、ダウンロードおよびダウンロードされた設定の読み込みの詳細な挙動は以下の通りです。

1. 15分に1回a)ではダウンロードを試み、ないしb)ではダウンロードされた設定の読み込みを行います。(idp.service.attribute.filter.checkInterval)

以上