Shib2SPConfig

Az Shibboleth 2 SP-t a `shibboleth.xml` állományon keresztül konfigurálhatjuk. Ebben a leírásban feltételezzük, hogy az SP konfigurációja a `/etc/shibboleth` könyvtárban van.

Alapszerkezet

Mindenekelőtt megmutatjuk a `shibboleth.xml` fájl alapszerkezetét, majd alább az egyes szerkezeti elemeket részletesen is tárgyaljuk, majd a fejezet végén egy teljes, működő konfigurációt mutatunk be.

```
<SPConfig xmlns="urn:mace:shibboleth:sp:config:2.0"
    xmlns:conf="urn:mace:shibboleth:sp:config:2.0"
    xmlns:md="urn:oasis:names:tc:SAML:2.0:metadata"
    logger="shibboleth/syslog.logger" clockSkew="180">

    <Extensions/>

    <OutOfProcess logger="shibboleth/shibd.logger"/>

    <InProcess logger="shibboleth/native.logger"/>

    <Listener/>

    <StorageService/>
    <SessionCache/>
    <ReplayCache/>
    <ArtifactMap/>

    <RequestMapper/>

    <ApplicationDefaults id="default" policyId="default"
        entityID="https://sp.example.org/shibboleth"
        homeURL="https://sp.example.org/index.html"/>

    <SecurityPolicies/>
```

```
</SPConfig>
```

Látható, hogy a szerkezet keretét egy `<SPConfig>` elem adja, ez fogja közre a különböző összetevők részletes konfigurációit. Az `<SPConfig>` opcionális attribútumai

- **`logger`** Annak a konfigurációs fájlnak a helyét adhatjuk meg, amelyben a loggolási tulajdonságok kerültek definiálásra. Alapértelmezés szerint ez a `shibboleth/shid.logger` fájl.
- **`clockSkew`** A legtöbb elosztott rendszerhez hasonlóan a Shibbolethnél is nagyon fontos, hogy szinkronban legyenek a rendszerben résztvevő elemek órái. Mivel komoly sebezhetőséget jelentene, ha a szerverek közti üzeneteken nem lenne megjelölve a feladás időpontja, ezért ezek az üzenetek időbélyeggel ellátottak, s minden rendszer elem csak egy bizonyos időnél nem régebbi üzenetekkel hajlandó foglalkozni. Ezt az értéket tudjuk itt megadni. Alapértelmezés szerint 3 perc, azaz 180 másodperc az értéke.

Összetevők

`<Extensions>`

`<OutOfProcess>`

`<InProcess>`

`<Listener>`

`<StorageService>`

`<SessionCache>`

`<ReplayCache>`

`<ArtifactMap>`

<RequestMapper>

A RequestMap megadja azokat a címeket (Host és Path), amelyeket a Shibboleth SP kezelni fog. Szerkezete:

```
<RequestMap applicationId="default">
    <Host name="www.example.org">
        <Path name="secure" authType="shibboleth" requireSession="true"/>
    </Host>
    <Host name="admin.example.org" applicationId="admin" authType="shibboleth"
requireSession="true">
        <AccessControl>
            <Rule require="affiliation">faculty@osu.edu student@osu.edu</Rule>
        </AccessControl>
    </Host>
</RequestMap>
```

A RequestMap több Host elemet is tartalmazhat, a Host elem 0 vagy több Path elemet tartalmazhat.

!!! danger "Figyelem"

```
Ha 1-nél nagyobb mélységű könyvtárat (pl. a `/shibtest/shibreq` nevűt) szeretnénk védeni,
akkor **nem** adhatjuk meg a *name* paraméterben a "shibtest/shibreq" értéket, hanem egymásba
ágyazott Path elemeket kell használni. A *name* paraméter nem tartalmazhat '/' karaktert.
```

Az egyes elemeknél paraméterekkel szabályozhatjuk, hogy az SP milyen módon kezelje a hostot vagy az útvonalat. A paraméterek felüldefiniálhatók. A legfontosabb paraméterek az alábbiak (ezek ugyanúgy használhatók Host-nál mint Path-nál):

- **`requireSession`**: ha értéke "true", akkor az SP csak akkor továbbítja a HTTP request-et az alkalmazás ill. a webszerver felé, ha sikerült létrehozni egy autentikált session-t. Ha "false", akkor az alkalmazás felelős azért, hogy létrehozza a Shibboleth session-t (ún. lazy session) Alapértelmezés: "false"
- **`applicationId`**: lehetőség van arra, hogy bizonyos helyekre érkező kérésekre az SP más és más módon próbáljon meg session-t létrehozni, ezt ún. Shibboleth Application-ben konfigurálhatjuk. Ha nem adunk meg értéket, akkor a "default" application-nél megadott értékek vonatkoznak majd a session-re.

<ApplicationDefaults>

Változat #1
document-uploader hozta létre 2025-08-07 12:06:37 CEST
document-uploader frissítette 2025-08-07 12:06:38 CEST