

Openid-connect to SAML interoperability

Introduction

OpenID-Connect has a clear design suitable for both frontend and backend.

SAML has a clear design for the frontend, but the backend usage is harder as the security in SAML cannot be placed at transport layer. Instead, it must be placed at document level. Additionally, it requires intensive use of cryptographic algorithms for signature and encryption.

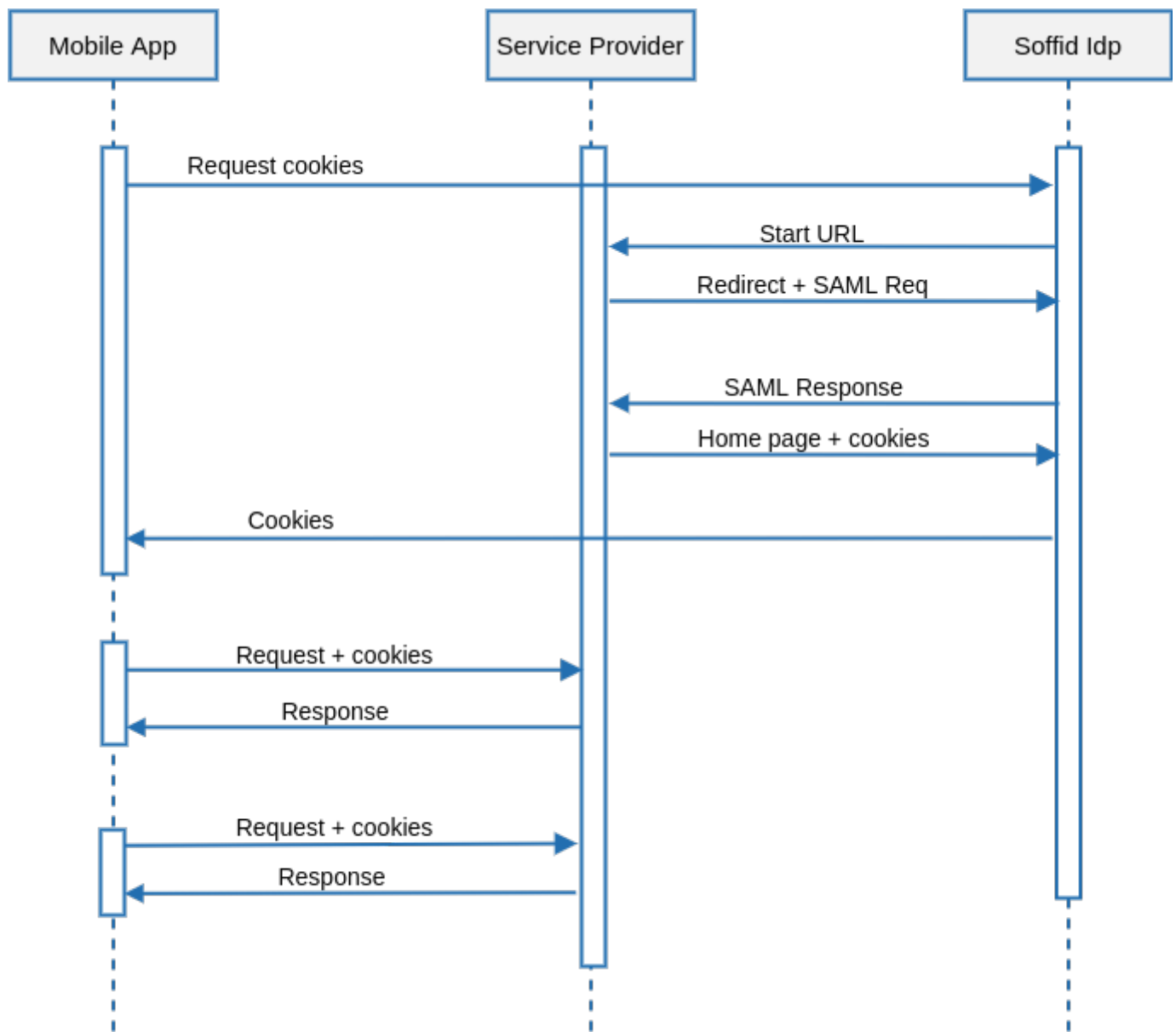
That's why some applications put a SAML frontend protection for both the frontend and relay on the session cookies generated by the frontend for backend access.

The problem arises when one service provider needs to invoke some services from a SAML enabled application that does not support or implement WS-Security.

To solve it, **Soffid Identity Provider** provides a service to get the session cookies required to access to a SAML application.

Data flow

The rest service **/userinfo/impersonate?url=....** will do the job, and will return the cookies to use to act upon the target application impersonating the current user.



Request

```
POST https://<YOUR_SERVER>:2443/userinfo/impersonate?url=http://targetapplication/
Accept: application/json
Content-type: application/x-www-form-urlencoded
Authorization: Basic dGVzdDp0ZXN0
[
  {
    "path": "/",
    "domain": "samltest.id",
    "name": "_shibsession_64656661756c7468747470733a2f2f73616d6c746573742e69642f73616d6c2f7370",
    "value": "_fa49874951dd05c18a0f68642c0736e9"
  },
  {
    "path": "/",
    "domain": "samltest.id",

"name": "_opensaml_req_ss%3A88b0af3e1ff47c911257490bc1a5749dfda1670948a563cec2fdf9e8a799f2c4",
    "value": ""
  }
]
```

Parameters

- **URL:** is the access URL for the target application.
- **Authorization:** contains the oauth token.

Response

The response contains the list of cookies to send to the target application.

```
[
  {
    "path": "/",
    "domain": "samltest.id",
    "name": "_shibsession_64656661756c7468747470733a2f2f73616d6c746573742e69642f73616d6c2f7370",
    "value": "_fa49874951dd05c18a0f68642c0736e9"
  },
  {
    "path": "/",
    "domain": "samltest.id",

"name": "_opensaml_req_ss%3A88b0af3e1ff47c911257490bc1a5749dfda1670948a563cec2fdf9e8a799f2c4",
    "value": ""
  }
]
```

Request

Once the application has got the list of cookies, it can invoke the target application URL

POST <https://targetapplication/api/service1>

Accept: application/json

Content-type: application/json

Cookie: cookie1=value1

As security measures, the impersonation profile must be enabled, and the source application must be entitled to use it against the target application

Revision #9

Created 21 September 2021 14:35:13 by pgarcia@soffid.com

Updated 21 June 2022 14:48:06 by pgarcia@soffid.com