Cybersecurity MSc: Identification, Authentication and Authorization 2020-21

Project (2nd part): User-centric identity management

June 23, 2021 Due date: Jun 27, 2021

Changelog

- v1.1 - Mention to SPs changed to IdPs in Section 2.2.
- v1.0 - Initial version.

1 Introduction

Nowadays, Service Providers (SPs) accessed by users' browser redirect the latter to an Identity Provider (IdP) selected by the former. IdPs usually start by authenticating the user that shows up redirected by an SP, and at the end of the authentication they send a set of identity attributes about that user to the SP. Such identity attribute provisioning may be presented by the IdP to the user but, ultimately, user have no effective control on which attributes are, in fact, provided.

Besides this lack of evidence, that can undermine the confidence of the users in such identification systems, IdPs can track the set of SPs with which the users interact with. And, for privacy sake, this should not happen.

2 Homework

The work consists on adapting the first part of this project to follow a different, user-centric identity management paradigm. To that end, the authentication and identification flow should become as follows:

- The SP redirect the user to their Helper application, providing 4 elements:
  - The SP identity;
  - The set of identity attributes it needs from the user;
  - The identity and location of the IdP that can provide those attributes;
  - A URL for collecting the requested set of attributes, signed, provided by the suggested IdP.
- The Helper application presents the SP request to the user, in a human-understandable way, collects their consent, and proceeds to authenticate the user in the IdP (using the protocols of the 1st part of the project).
- The Helper application gets the intended attributes from the IdP, presents their value to the user, and sends them to the SP. In this step, it is fundamental to hide any details about the SP from the IdP.
- The SP validates the IdP signature in order to accept the user identity attributes.

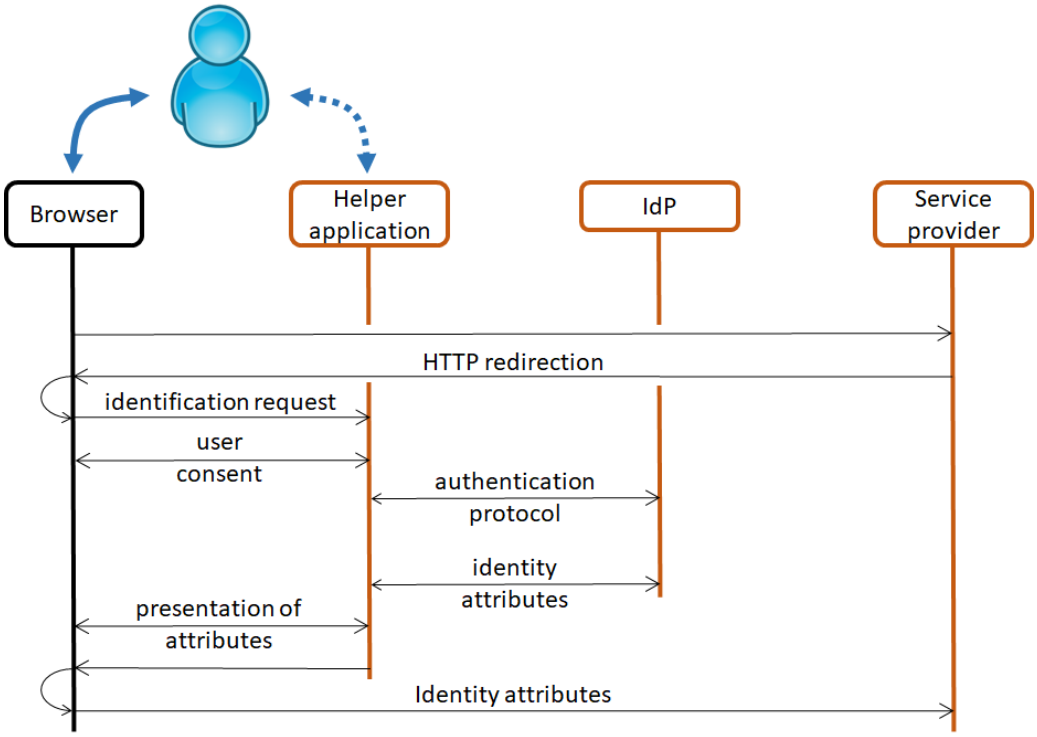

Figure 1: Architecture of the system to implement. The Helper application should run locally to the browser

2.1 Secure tunneling

The communication between the Helper application and the IdP can be performed using consecutive HTTP redirections through the user browser. This strategy can benefit from an HTTPS connection between the browser and the IdP, relying on the certificate verification performed by the browser during the setup of the HTTPS session with the IdP.

Alternatively, the Helper application can directly address the IdP, as illustrated in Figure 1, acting as a direct HTTP client. In this case, the Helper application must encrypt and authenticate the communication, which can be implemented using key material provided by the IdP to the Helper application. This provisioning, however, needs to forced by the Helper application, unlike in the previous project. Such forcing can be easily performed with an extra redirection through the browser just to collect that key material.

In both cases, the secure tunneling of the protocols between the Helper application and IdPs is assured, which is beneficial for preventing the eavesdropping of the authentication and identification protocol messages, which could be used to implement off-line password guessing attacks and identity eavesdropping.

2.2 IdPs signatures on identity attributes

In this model, IdPs no longer know to which SP they provide the users' identity attributes. Therefore, the term "signature", which, in some scenarios, can be replaced by a Message Authentication Code (MAC) computed from a shared key, is not suitable here; consequently, IdPs need to use asymmetric credentials.

In this project, you can either use certificates or pre-shared public keys for dealing with IdP signatures.

2.3 On the use of SAML

If you have used SAML in the 1st project, you could only reuse it for handling authentication responses, since authentication requests do not allow SPs to specify the set of identity attributes they require.

This is established by the metadata exchanged between SPs and IdP when they agree on the way they will interact (including their authentication and HTTP endpoints), and we do not want that to happen any more.

Therefore, in this project you can use whatever you like to format the identity-related requests and responses involving an SP, an IdP and the Handler application, as long as it is understood by all of them. Do not forget to explain this new format in your report.

2.4 Protocol collapsing

Once a user shares an asymmetric credential with an IdP, they can easily skip the authentication protocol and proceed to the identity protocol. In fact, they can use those credentials to sign identity requests and to decrypt identity responses.

3 Delivery

Send your code to the course teachers through Elearning (a submission link will be provided). Include a report, with no more than 30 pages, describing the authentication protocols defined and implemented, the overall sequence of operations considered, the interfaces used and their parameters, some relevant implementation details (not complete copies of the code!) and the results achieved.

4 Evaluation

This 1st part of the project will be evaluated as follows:

- Identification protocol: 30%;
- IdP authentications: 20%;
- Helper application interface for identity attributes: 30%;
- Written report, with a complete explanations of the strategies followed and the results achieved: 20%