Step-up Authentication as-a Service  
Abstract of a 25 minute Presentation for TNC2017  

Pieter van der Meulen, SURFnet*
pieter.vandermeulen@surfnet.nl

Keywords  
Multi-factor authentication, level of assurance, federation, SAML 2.0, cloud services

1 Abstract

1.1 Introduction

At TNC2013 SURFnet presented\[1\] a study\[2\] on the architecture and processes for a multi-factor authentication (MFA) cloud service for services in a SAML federation. Continuing on that work we evaluated existing solutions, piloted with vendors of MFA solutions and decided to develop a new service, developed this service and a business model for the service and finally took the service in production as “SURFconext Strong Authentication”\[3\] in August 2015. In this talk we present the ideas we developed while designing and building the software for this service and the lessons we learned while running it in production for almost two years\[4\].

In this abstract we describe what the “SURFconext Strong Authentication” service is, and what sets it apart from other MFA solutions. Then we discuss the architecture of the solution and describe how we employed SAML in novel ways to solve several architectural problems. Next we discuss how we employed the “Event Sourcing” design pattern and how this is a good fit this kind of service. Finally, we look forward to some future planned work.

1.2 What is Step-up Authentication as-a-Service (Step-up)?

The software that runs SURFconext Strong Authentication is called “Step-up Authentication as-a-Service”, or “Step-up” for short. It distinguishes itself from existing multi-factor authentication solutions in that it:

- supports the complete NIST SP 800-63\[4\] / ISO/IEC 29115 assurance process in an existing SAML federation infrastructure by providing:
  - self-service user registration and 2nd factor token issuance;
  - identity vetting distributed to registration authorities at the institutions;
  - two-factor step-up authentication at multiple assurance levels.
- can be offered “as-a-service” without requiring technical integration at the SAML Identity Provider (IdP) and minimal integration at the Service Provider (SP);
- is a completely open source solution that is part of OpenConext\[5\], but does not require any other OpenConext components; Other parties are actively looking at using Step-up. For instance, ELIXIR\[6\];
- allows for integration of new 2nd factor token types without changes to existing IdPs or SPs;
- is completely web based.

1.3 Architecture and Design

Step-up consists of a SAML proxy with additional components for supporting the complete life cycle of a 2nd factor. From self-service registration by the user, to vetting by a Registration Authority (RA) to revocation.

In the illustration on the following page the 1st factor IdP is the existing institutional IdP of the user. The 2nd factor IdP is offered as part of the service. Each 2nd factor IdP provides authentication for a type of 2nd factor, e.g. Yubikey\[7\] or Tiqr\[8\].

---

*SURFnet, Postbus 19035, 3501 DA Utrecht, The Netherlands
\[1\] At the start of TNC2017 the service will have been in production for 21 months
\[2\] The Step-up software is currently being moved under the OpenConext project
1.4 Security and Reliability

We paid special attention to the reliability and security of the Step-up software:
- Only the middleware can write to the database, the middleware is not accessible from the internet.
  Only the Self-service and RA components can access the middleware.
- For SAML login to an SP the gateway, the database, the user’s 1st factor IdP and, when step-up is required, the user’s 2nd factor IdP must be available.
- For redundancy, the production setup uses a MariaDB Galera[9] cluster to run in two different locations.

1.5 Using SAML for integration

Step-up uses SAML extensively for connecting and integrating Step-up components internally, and also for connecting external components to Step-up. Below we describe three cases where we employed SAML in a novel way to solve an architectural problem.

1.5.1 Generic Step-up Service Provider (GSSP)

The API between the 2nd factor IdP and the gateway uses SAML only. To allow the IdP to handle both the registration of new tokens and the authentication of previously registered tokens we add a Subject element to the SAML AuthnRequest during authentication. This optional element is defined in the SAML 2.0 Core standard[10]. There are two use-cases for the 2nd factor IdP:
1. It receives an AuthnRequest without a Subject: register a new token, and return the identifier of the token in the Subject of the Assertion.
2. It receives an AuthnRequest with a Subject: authenticate a previously registered token, and return the identifier of the token in the Subject of the Assertion.
A 2nd factor IdP that implements this interface is known as a Generic Step-up Service Provider (GSSP).

1.5.2 Second Factor Only (SFO) authentication

A lesson learned is that institutions want to use Step-up not only for cloud services, but also for adding MFA to their existing internal applications. Here we ran into the problem of bridging the gap between the web-based SAML world and internal native windows applications. We developed a GSSP like SAML based solution to authenticate only the 2nd factor of the user. Through our institutions we’re actively working with vendors like Citrix and F5 to make this SFO[11] authentication available on their application gateways.

1.5.3 Transparent proxy

Step-up reuses the functionality in the 1st factor IdP as much a possible instead of reimplementing it. An IdP typically does more than just authenticating users. It decides what users may access a specific SP, what attributes to release, ask for consent, creates per SP pseudonyms etc. For this the IdP needs to know the SP behind the Step-up proxy. The Step-up proxy on the other hand needs to know the user ID so it can look-up the registered 2nd factor of the user. Sending the SP identifier though the proxy to the IdP is commonly done using a Scoping/RequesterID element in the AuthnRequest. Passing the user ID from the IdP though the proxy to the SP required a novel solution. To send both IDs the IdP:
- sets the ID for the proxy in the Subject of the Assertion.
- sets the ID for the SP in the eduPersonTargetedID attribute.
To create the Assertion for the SP the proxy copies all attributes that it received from the IdP and replaces the Subject in the Assertion with the value in the eduPersonTargetedID attribute.
1.6 Event Sourcing

Step-up uses the Broadway\[12\] Event Sourcing library to handle persistence. A typical database reflects the current state of the system. Some history may be kept in the form of audit logs, but typically not enough to recreate the current state of the system. When using Event Sourcing all the events that lead up to the current state of the system are kept. An event is a state change of an object in the system. These events can be replayed to recreate the state of the system at any point in time. To prevent having to replay the events each time the current state is needed, projections are used. A projection is the reflection of the state of the system for a particular purpose. E.g. a table of the active tokens, the RA’s etc. Event Sourcing turned out to be a very good fit for Step-up because automatically provides a complete audit trail and prevents loosing any history. The latter meant we did need to find a solution to handle the right to be forgotten privacy requirement.

1.7 Future work

Several new features are on the roadmap for Step-up. We expect to be able to present on some of these at TNC2017:

- Add a SAML attribute provider to release the assurance capabilities of a user authentication to a SP.
- Use Policy decisions from a XACML\[13\] PDP for deciding the Required assurance level and allowed the 2\textsuperscript{nd} factors for a user.
- Additional 2\textsuperscript{nd} factor integrations (Vasco) using GSSP and adding institution specific 2\textsuperscript{nd} factors.

Vitae

Pieter van der Meulen

Pieter van der Meulen is a technical product manager at SURFnet where he works on innovation projects involving identity, trust and security. He combines his work in innovation with operation of SURFnet’s SURFconext and SURFconext Strong Authentication services. Since graduating from university Pieter has had various roles in software development, from programming to design and architecture and management. Pieter joined SURFnet in 2011.

References

[12] https://github.com/qandidate-labs/broadway