Extension Strong Authentication based on requirements

Utrecht, december 2017
Version: 0.3

About this publication

Extension Strong Authentication based on requirements

SURF
P.O. Box 19035
NL-3501 DA Utrecht
T +31 88 787 30 00

info@surf.nl
www.surf.nl/en

Authors
Pieter van der Meulen - SURFnet
Michiel Schok - SURFnet

December 2017

SURF is the collaborative organisation for higher education institutions and research institutes aimed at breakthrough innovations in ICT.
This publication is online available through www.surf.nl/en/publications

Table of Content

Introduction

SURFconext Strong Authentication[1] (SCSA) offers strong authentication as-a service to the SURFnet constituency. SCSA is offered and operated by SURFnet. The software used to run the service is the "Stepup" software that is developed under the OpenConext[2] open source project.

SCSA works seamlessly with SURFconext. SURFconext allows the users of the institutions that together form the SURFnet constituency to use the federated authentication provided by their institutions to authenticate to a wide variety of services. SCSA enhances the strength of the authentication and the identification of the federated authentication provided by the institutions by adding a second authentication factor and a centrally orchestrated identity vetting process. See Figure **1** onder.

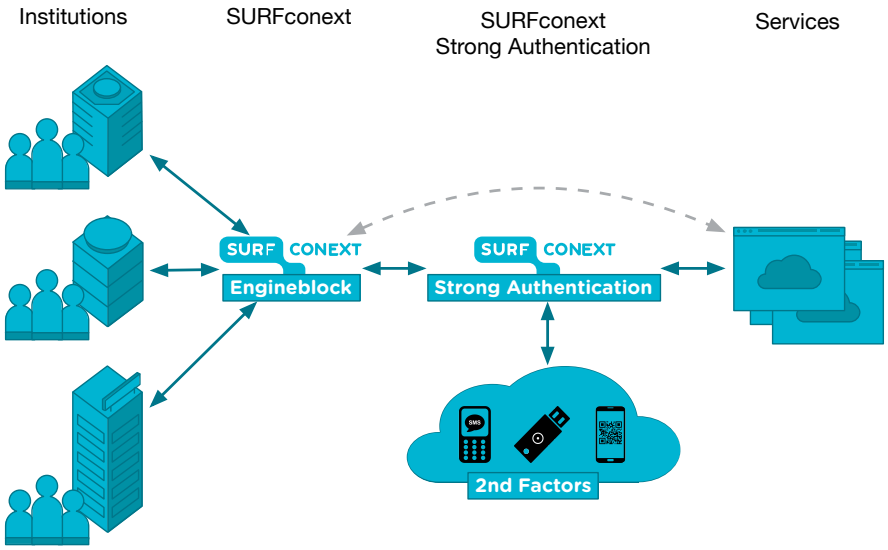


Figure 1: Enhancing the strength of the federated authentication from institutions to services though SURFconext and SURFconext Strong Authentication.

The approach chosen for SCSA has several properties that differentiate it from typical multi factor authentication (MFA) solutions that are worth pointing out:

- There are no technical or infrastructure changes required to the federated authentication infrastructure and identity management systems of the institutions, they do typically have to play a role in the identity vetting process and user support
- Registered users can use strong authentication with all services that are connected to SCSA
- New 2nd factor types and identity vetting processes can be added centrally, without requiring changes to the integration at the institutions or services
- Because of the vetting process that is an integral part of the registration process of SCSA, SCSA raises the assurance level of both the authentication and the identification of the user.

The above properties allow SURFnet to offer and operate SCSA to its constituency as a managed service, with SURFnet, and the services, taking over the burden of managing the technical integration required to make two factor authentication work with existing services. However, these strong points come with two notable limitations:

[1] https://www.surf.nl/en/services-and-products/surfconext/what-is-surfconext/surfconext-strong-authentication/index.html
[2] https://openconext.org

1. Connecting to SCSA requires services to use a web based protocol, for SCSA we chose the SAML 2.0 WebSSO Profile. This excludes non-web protocols like RADIUS.
2. Services must connect to SCSA, and thus to SURFconext. This means that the service will have to make changes to use SCSA. For services that are already connected to SURFconext this is usually a small change, but for others this may be much more work.

1. Limitations of SCSA

In the previous section we pointed out that SURFconext Strong Authentication (SCSA) has two limitations. These limitations are a consequence of the design and technology choices that allow SCSA to be offered as managed service to its institutions. It is worth going into these limitations in more detail because the efforts of overcoming them are what inspired us to develop second factor only (SFO).

1.1.1 First limitation – Services must use a web-based protocol

The SAML 2.0 WebSSO Profile specifies how to use SAML 2.0 authentication from a webbrowser. This means that a service that uses SA must be web based, or at least must be able to use a web browser for authentication. This is an obstacle for so called non-web applications. Examples of a non-web applications are desktop mail clients that use the IMAP protocol to connect to an IMAP server and VPN clients that use RADIUS for authentication.
The gap between the web and the non-web world can be bridged by invoking a webbrowser from the non-web application for authentication. We see that more and more applications, both on the desktop and on mobile devices, are being enabled to allow use of a webbrowser for authentication, or are becoming web based applications altogether. A notable example of both are the newer Microsoft Office 365 clients for the desktop and for mobile devices, where the use of a web browser for authentication from a desktop or mobile application is now possible (branded as “Modern Authentication”). Additionally, Microsoft offers completely web-based versions of its office suite of applications.

1.1.2 Second limitation – Services must connect to SCSA

To be able to use SCSA the service must use SCSA as a SAML IdP. This means that each service must be connected to SCSA, and thus to SURFconext. This means joining the SURFconext federation and conforming with the policy and technical requirements of connecting to SURFconext, and SCSA. Depending on the service this can be a significant investment for all the involved parties: the sponsoring institution, the service provider and also the SURFconext support team. The positive aspect of this route is that once this work is done, you have a service, or an integration with a service that can easily be reused by other institutions.
What we see, is that this poses an obstacle for institutions that want to switch to SCSA because they want to connect all their (existing) applications to SCSA. Institutions, understandably, do not want to bother their users with more than one second factor and issuance process, and do not want the cost and overhead of these, and an additional (on premise) strong authentication solution.

2. Second Factor Only (SFO)

To overcome these limitations, we created an additional authentication interface in SCSA. This interface offers the “raw” second factor authentication which offers more flexibility for integration with other systems. We named this interface “Second Factor Only” (shorthand: SFO) because that name best describes what the extension does from the perspective of SCSA – it only authenticates the second factor of the user, where for normal authentications SCSA will verify both the first and the second factor of the user. An application that uses SFO must perform the first factor authentication of the user itself. The way the applications authenticates the user is up to the application, typically it will use authentication to an existing Windows domain or LDAP. Subsequently, the application will request the SFO extension to initiate the authentication of the second factor. Part of the SFO authentications request must be the identity of the user as it is known to SCSA (i.e. the SURFconext identifier). This means that the application must determine this ID based on first factor authentication that it performs. This SFO authentication process uses the same mechanism (i.e. SAML) for authentication the user as the regular SCSA: all of the second factors that are supported by SCSA are possible (TiQR, SMS, Yubikey, U2F, …), and it is using the same registration database and vetting process.

2.1. Architecture and Development

The second factor authentication protocol is based on the same SAML 2.0 WEB-SSO authentication that is used by SURFconext (SA). The only difference with the authentication process currently offered by SCSA is that the service now must communicate the identity of the user in the authentication request, this because it is the service that performs the first factor authentication, and thus establishes the identity of the user. The solution implemented uses a SAML 2.0 AuthnRequest with a Subject element. It is used to request authentication for a specific user. This is option is part of the SAML 2.0 specification, but is not commonly used. To indicate the type of second factor requested the SP can use the RequestedAuthnContext element in the AuthnRequest, just like a normal SCSA authentication.

More details on SFO:

- Documentation on GitHub: https://github.com/OpenConext/Stepup-Gateway/wiki/Second-factor-query-protocol
- SFO was part of our presentation at TNC2017: https://tnc17.geant.org/core/presentation/44
- Documentation for service providers: https://wiki.surfnet.nl/display/surfconextdev/Second+Factor+Only+%28SFO%29+Authentication

2.2. Security Analysis

To assess the security of the SFO implementation, Computest performed a security audit of the SFO implementation on the Stepup-Gateway in combination with the Microsoft ADFS MFA extension for SCSA which uses SFO.

3. Pilots

After development of the Second Factor Only endpoint on the SCSA service, we developed a demo-application and conducted pilots with 2 institutions.

3.1. SFO demo application in SimpleSAMLphp

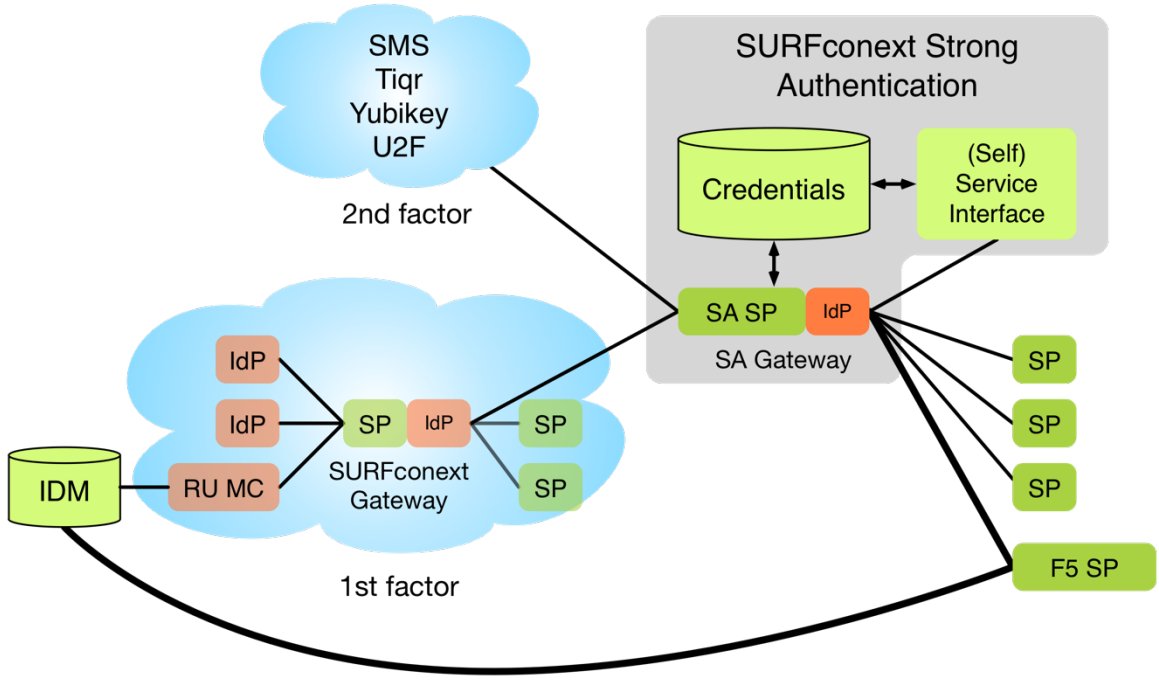
The source code of the demo application can be found at https://github.com/SURFnet/Stepup-SFO-demo. This application demonstrates how an SP can use SFO from SimpleSAMLphp. A feature required for SFO – setting a Subject in an AuthnRequest – was contributed to SimpleSAMLphp.

3.2. SFO with Citrix NetScaler at VUmc

VUmc is using Citrix Netscaler and other Citrix components to facilitate remote access for employees to their remote desktop environment. They expressed a wish to establish multi-factor-authentication to this environment when users were outside of the campus
In close collaboration with Citrix we discovered that a crucial signing step of the authentication request could not be performed by the Citrix firmware. By deploying a proxy (based on SimpleSAMLphp) between Citrix Netscaler and SFO endpoint a feasible workaround was found.
VUmc was satisfied about this solution, and decided to move this to production in Q1 2017.
In June 2017 with the release of firmware v12, Citrix has added the necessary signing natively to Netscaler, removing the need for a workaround using a proxy.

3.3. SFO with F5 Big-IP at Radboudumc

Radboudumc (RUMC) would like to connect their F5 Big-IP to the SFO endpoint of SCSA. A high-level architecture is shown below. The first factor is validated directly at the IDM system of RUMC (and not via the 'traditional way' using the SURFconext Gateway). Subsequently the second factor is validated using the SA Gateway.



At first, F5 was unable to specify the Subject in the SAML authentication request. In close collaboration with RUMC and F5, a specialist of Vosko[3] has implemented this feature and contributed this to a public F5 solution repository.[4]

This solution has been tested, and proved to be working. RUMC expressed the wish to explore other means of secure authentication, and has postponed the decision to bring this to production.

4. Next Steps

In discussion with our constituency the functionality of Second Factor Only seems to solve a problem: how to add step up authentication to internal applications that are not connected to SURFconext for authentication. Therefore, we plan to bring this functionality to production.
In a separate project, an ADFS plugin was developed which uses the SFO endpoint of SCSA to perform second factor authentication. Using this plugin, all applications which use ADFS for authentication can – without modification to the application – use SFO to enhance the authentication.

[3] https://www.vosko.nl
[4] https://devcentral.f5.com/codeshare/surfconext-second-factor-only-sfo-authentication-1012