DAME: On-demand Internet-scale SAML Metadata Exchange

Michael Grabatin, Wolfgang Hommel, Stefan Metzger, and Daniela Pöhn
Leibniz Supercomputing Center
Munich Network Management Team
85748 Garching n. Munich, Germany
Email: [grabatin,hommel,metzger,poehn]@lrz.de

***Abstract*—Inter-organizational IT service access based on the Security Assertion Markup Language (SAML), the predominant standard for Federated Identity Management (FIM), suffers from metadata scalability issues when Identity Providers (IDPs) and Service Providers (SPs) from different federations are involved. This article presents Dynamic Automated Metadata Exchange (DAME) for SAML-based FIM and its open source implementation, GÉANT-TrustBroker, which is currently in preparation for pilot operations within the pan-European research and education network, GÉANT. Based on the DAME metadata broker architecture and workflows, the concept of Internet-scale dynamic virtual federations is introduced and life-cycle management concepts are discussed; special emphasis is put on the risk management aspects of GÉANT-TrustBroker.**

Keywords–Federated Identity Management; SAML; Shibboleth; Inter-Federation; Trust-Management.

I. INTRODUCTION

Identity & access management (I&AM) is the umbrella term for managing users and their permissions. While I&AM can be applied to individual IT services, such as a web application, I&AM architectures typically cover the majority of all IT services within an organization. For example, higher education institutions use I&AM systems to manage the accounts of all of their students, staff, faculty, guests, and alumni along with their individual access rights to email servers, file storage, learning management systems, and other IT services. I&AM has many challenging organizational aspects, such as defining responsibilities for data quality and master systems for individual information, but its implementation technology has matured over the past 15 years. Typically, central Lightweight Directory Access Protocol (LDAP) based directory services or other database management systems aggregate all the required data and make it available to the I&AM-connected IT services.

Given the sensitivity of the personally identifiable information (PII) stored within I&AM systems, read access is only granted to trusted IT services in a selective manner. For example, IT services, which only need to authenticate users based on their usernames and passwords, will not be allowed to also read, for example, their email addresses and telephone numbers. Therefore, I&AM systems authenticate the IT services that make use of them and are often operated in firewall-protected internal networks. As a consequence, I&AM systems are not suited for inter-organizational use cases, such as multiple users from different universities and industry partners accessing a web-based collaboration platform as part of a research project.

Federated Identity Management (FIM) provides partial solutions for inter-organizational use cases. In its basic form, it assigns the role of Identity Providers (IDPs) and Service Providers (SPs) to organizations: IDPs are the home organizations of users and provide authentication as well as authorization services, whereas SPs operate IT services that can be used by multiple IDPs. Sets of at least one IDP and one SP are referred to as federations. In higher education, several dozens of national federations have been established over the past 10 years, such as InCommon in the United States, SWITCH-AAI in Switzerland, and DFN-AAI in Germany. In industry, federations are typically established for sector-specific supply chains, such as the pan-European automotive platform Odette. The Security Assertion Markup Language (SAML) is the predominant technology in both professional areas, whereas consumer-oriented Internet services often make use of more lightweight approaches such as OpenID Connect.

The large-scale real-world application of FIM is subject to two major distinct challenges: First, IDPs and SPs need each other's metadata, i. e., information about technical communication endpoints and server certificates for message signatures and encryption. Second, IDPs must provide information about their users, referred to as *user attributes*, in a data format compatible to the SP and its IT service. Existing federations solve the first problem by first centrally aggregating the metadata of each IDP and SP and then distributing the complete metadata package to each participating organization. The second problem is typically solved by defining a federation-wide user data model, commonly referred to as *federation schema*. Both solutions work well for average-size federations, but hit a dead end when users want to access IT services across federations' borders, e. g., in international research or cross-industry-sector projects: while inter-federations, such as eduGAIN, attempt to aggregate and distribute the SAML metadata of several national federations, the organizational overhead as well as the technical performance impact of huge metadata sets deters many organizations from participating. Also, given the heterogeneity of federation schemes, successful user attribute exchange is limited to their intersection, leaving many IT services with a lack of information about individual users that limits their functionality.

In [1], we presented a SAML metadata broker for dynamic federations and inter-federations. It supports the user-triggered, on-demand exchange of SAML metadata between pairs of IDPs and SPs whenever a user from a specific IDP attempts to access a particular SP service for the first time. It significantly simplifies the organizational and technical aspects of SAML setups across existing federations' borders and optimizes the technical scalability by avoiding the aggregation of metadata that is not relevant to individual organizations. It also supports inter-federation user attribute exchange by providing a repository, which allows for the sharing and re-use of conversion rules. Along with several improvements, the approach has since been refined as follows: First, the protocol has been

2015, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

formally specified in the IETF Internet-Draft *Integration of Dynamic Automated Metadata Exchange into the SAML 2.0 Web Browser SSO Profile* (DAME). Second, an open source implementation based on the popular FIM software suite Shibboleth, called GÉANT-TrustBroker (GNTB), has been developed and tested within the pan-European research and education network GÉANT; it currently is being prepared for multi-national pilot operations and scheduled for integration into the GÉANT service portfolio as a part of the ongoing, EC-funded GÉANT GN4 project.

In this article, the background, design rationale, and current state of both DAME and the GÉANT-TrustBroker implementation are presented in detail. It is structured as follows: In Section II, related scientific work and practical approaches are discussed. Section III then explains the chosen broker-based approach along with its architecture and workflows. The concept of dynamic virtual federations along with their life-cycle and management procedures are then detailed in Section IV. Afterwards, the GÉANT-TrustBroker implementation is presented in Section V, followed by a discussion of its risk management aspects in Section VI. The article is concluded by a summary and outlook to ongoing work in Section VII.

II. Related Work

Though FIM is used in the recent years and many theoretical and practical solutions were designed, scalable and at the same time secure solutions are rarely found. All related work, which was investigated, concentrates on only one particular aspect and does not see the problem as a whole. First practical solutions are shown, before scientific approaches are explained.

A. Practical Approaches

Although SAML does not specify that SAML metadata of each participating entity, i. e., IDP, SP, and attribute authority, needs to be aggregated and exchanged beforehand, it is the current practice. In order to aggregate and exchange metadata, several federations have established metadata registry tools. The Swiss federation SwitchAAI was the first NREN federation to develop a so-called Resource Registry [2], where entities can register their metadata and update information. Based on all uploaded metadata, the national metadata file is aggregated, which then can be downloaded by the participants. Though the national web tool helps entities to manage their information, many manual steps are required and the local configuration needs to be updated manually.

Public Endpoint Entities Registry (PEER) by Ian Young et al. [3] is another practical solution. The implementation of PEER is called REEP and can be used by any entity, independent of the federation and the protocol used. Though PEER moves the metadata aggregation from federations one layer up to a central service, the metadata is still aggregated. Another drawback are the manual steps, e. g., to generate an attribute filter adjusted to the IDP.

Another way to distribute metadata is the submitted Internet-Draft (I-D) Metadata Query Protocol by Ian Young [4], which has a profile for SAML environments. In this approach, metadata can be retrieved by hypertext transfer protocol (HTTP) GET requests, which allow dynamic metadata distribution. Therefore, Metadata Query Protocol solves the problem of huge aggregated metadata files, while manual steps are needed to adjust the local configuration. Furthermore, Metadata Query Protocol does not suggest a workflow to exchange metadata on-demand and establish trust between two entities, i. e., SP and IDP.

B. Scientific Approaches

The scientific approach of Federated Attribute Management and Trust Negotiation (FAMTN) by Bhargav-Spantzel et al. [5] assumes that each SP can act as an IDP. Since no IDP exists, the user information need to be stored at the users. Internal users of the FAMTN system are supposed to perform negotiations by exploiting their single sign-on (SSO) ID without repeating identity verifications. External users need to declare all their attributes in the first communication, in order to receive a temporary user ID. At the second communication, the SSO ID is exploited, though it could be misused for attacks. It might appear that a provider needs less or more attributes, leading to violations of data minimization or further negotiations between providers.

Arias Cabarcos' et al. approach of IdMRep [6] shifts from pre-configured cooperation to dynamic trust establishment by a distributed reputation-based mechanism based on local dynamic trust lists (DTLs) and external reputation data. DTLs can, e. g., receive recommendations from other entities, when a cooperation was successfully ended. Hence, the cooperation runs through different phases: receiving and evaluating information, local calculation of the risk and trust values, dynamic decision based on available information, and monitoring and adjusting trust level. This mechanism does not work well for new entities. Because of the amount of data processing required for all external and internal trust information especially in inter-federations, this results in yet another bottleneck. It is vulnerable to Sybil attacks. Furthermore, the problem of different attributes, syntax, and semantics is not considered.

The approach Dynamic Identity Federation by Md.Sadek Ferdous and Ron Poet [7] also concentrates on the dynamic trust. Dynamic Identity Federation distinguishes between fully trusted, semi-trusted, and untrusted entities. Authenticated users are allowed to add SPs to their IDPs, while SPs add the IDPs to their local trust anchor list for further usage. The user establishes the trust by generating a code at his first authentication. He then informs the SP about the code and the EntityID of the IDP. After verification, the SP generates a request with two invisible fields, i. e., MetaAdd and ReturnTo. Both fields are used for the metadata exchange, while the IDP needs to evaluate the value of MetaAdd. When the user gives his consent, the IDP adds the chosen SP to the list of semi-trusted entities. Semi-trusted entities are not allowed to receive sensitive attributes. Untrusted entities are given the National Institute of Standards and Technology (NIST) level of assurance (LoA) 1. If the SP is not known by the IDP, a proxy could be used complicating the trust establishment. The trust establishment via the user generating and forwarding a code is not user friendly, while both invisible fields are not necessary. The fragmentation into trusted, semi-trusted, and untrusted entities as well as the usage of NIST LoA 1 does not reflect real world with its different LoA schemes and the trust relationships.

In sum, different aspects can be adopted, though neither approach tries to solve the problem as a whole. While the Metadata Query Protocol is a scalable approach for distributing metadata, it needs to be included in a scalable architecture

2015, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

for dynamic trust establishment. The trust framework needs to reflect real world, while being flexible. IdMRep could be added as a trust layer on top of LoA. Furthermore, the solution needs to be secure for the participants.

III. SAML Metadata Broker

The project GÉANT-TrustBroker was established within GÉANT to address the challenges of SAML metadata exchange. The central trusted third party (TTP) GNTB is, as described in [1], [8], and [9], an on-demand repository for metadata and conversion rules. It extends existing discovery services, formerly known as WAYF (Where Are You From?), in order to locate the appropriate IDP. As both entities, i. e., IDP and SP are known by the TTP, metadata can be exchanged on-demand, if triggered by the user. In order to exchange and integrate the metadata automatically into the local configuration, IDPs and SPs need an extension for communicating with the TTP, as shown in Figure 1.

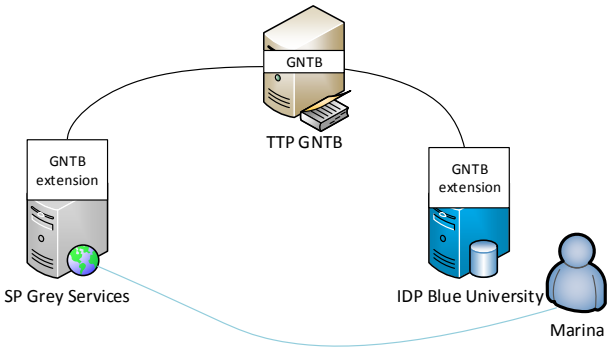

Figure 1. Basic architecture for dynamic metadata exchange with GNTB

By automating the metadata exchange, GNTB simplifies the discovery of entities and establishes the technical trust in dynamic virtual federations, while it improves the scalability of metadata release. The cooperation is not limited to an existing federation or inter-federation. Instead, the metadata can be exchanged across borders, making the federation virtual. As the metadata is not aggregated beforehand at the different providers, but exchanged on-demand, the size of the metadata files integrated at each provider is reduced. If the user trusts the SP, he can trigger the technical trust establishment at first time use of the SP, as described in the next section. The metadata is then exchanged and automatically integrated into the local configuration of the user's chosen IDP and the requested SP by extensions of the predominate software. Because the TTP keeps track of the established technical trust relationships, it can trigger the download of updated metadata information if needed. Furthermore, a conversion rule repository is provided, in order to extend and translate the amount of attributes used in collaborations. In the following section the different workflows are explained in detail. Last but not least, the architecture of this approach is visualized.

A. Workflows

In this section, three different types of workflows will be explained: management workflows, conversion rule workflow, and the core workflow. *Management workflows* on the one hand allow SP and IDP administrators to register, upload, update, and delete metadata information as well as attribute conversion rules. Uploading metadata information requires a proof-of-ownership verification step. This can be technically implemented by creation of a specific resource in the document root of the web service for that domain with a specific, random string given. Once created, the administrator can trigger the verification process and, if receiving an 200 OK status code in the response message, the metadata information will be inserted. Alternatively, certificate based verification or simple mechanisms, e. g., comparison of the entities name with the mail address' domain of the logged in user can also be implemented. This degree of automation keeps humans on the broker side out of the loop, so newly registered entities do not have to wait for manual approval of their application.

Conversion rule workflow: Since SP and IDP are usually not members of the same (inter-)federation, syntax and semantics of the user attributes, i. e., the attribute schema used, vary. Fortunately, because the metadata of a SP usually contains information about the required attributes, the IDP can determine if it can fulfill the attribute requirements directly or further attribute conversion will be required. In the latter case, the IDP can now check whether suitable rules are available at GNTB. This step can be automated by scripts. If suitable rules were found, these will be automatically downloaded and integrated into IDP's attribute resolver and filter configuration. This conversion rule workflow is not part of the core workflow, but can be triggered by it.

On the other hand, the *core workflow* (presented in Figure 2) builds up the provider pairing or virtual federation. This core workflow was specified as an Internet-Draft and submitted to the IETF as DAME.

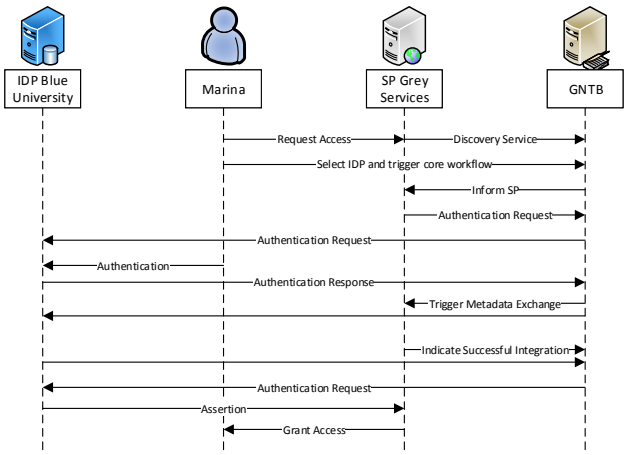

Figure 2. DAME core workflow for provider pairing

Explaining the core workflow, we assume that researcher Marina from an IDP Blue University, member of the federation Blue, requests access to a protected resource provided by SP Grey Services, which is not a member of the same federation. The often seen embedded discovery service on the SP lists all already trusted IDPs. We assume that Marina's IDP is not listed

2015, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

there, so she can trigger the DAME workflow. Comparable to other typical SAML-based workflows, Marina is redirected to GNTB, technically speaking to its centralized discovery service component. Provided that both IDP Blue University and SP Grey Services are already registered and uploaded their metadata to the TTP using the management functions mentioned previously, Marina can pick the IDP she wants to use. Rather than redirecting Marina directly to the chosen IDP for authentication, GNTB passes the information about the selected IDP back to the requested SP. If Grey Services decides that users from that chosen IDP can be accepted, it sends a generated SAML authentication request to GNTB, which temporarily stores it. GNTB, now in the role of a regular SP, generates a new SAML authentication request and redirects Marina to her chosen IDP Blue University. This two-part user authentication is necessary to prevent malicious users to add arbitrary IDPs' metadata to any SP and vice versa. After successful user authentication and receiving the SAML assertion in the corresponding response message, GNTB triggers the IDP and SP afterwards to download and integrate each other's metadata. This can be done either by using Young's Metadata Query Protocol, explained in Section II, or any other appropriate mean, like a simple web service or REST API function, as described in the next section. After updating each others' configuration, GNTB forwards the temporarily stored SAML authentication request to the IDP. Unless forced user re-authentication is required by the SP, the IDP immediately responds with a SAML authentication assertion to Grey Services and Marina's browser is redirected back to the requested service and access will be granted. If Marina inadvertently has chosen her IDP, which Grey Services already trusts, a regular FIM authentication workflow without further involvement of GNTB is initiated. Analogous, if the metadata information has been exchanged and the technical trust has been established successfully, GNTB is not involved anymore.

In order to manage these workflows, a TTP was designed, which interacts with IDPs' and SPs' extensions.

B. Architecture

In this section, the architecture, internal data model of the TTP, and the data access layer are described. Besides the GNTB core service providing a centralized discovery service for IDP selection and storing metadata information on each provider entity, an DAME extension has to be installed at IDPs and SPs to enable metadata exchange and automatic integration as well as the attribute conversion rule handling.

While both metadata information and attribute conversion rules are stored in TTP's file system, a relational database is used to support the provided management functions. In contrast to the tables described in [1], the proof of concept has additional tables to realize all added functionalities (in alphabetical order), e. g:

- attributes: This table stores information on source or destination attributes, which can be used in attribute conversion rules. To identify the attributes the unique object identifiers, e. g., urn:oid:1.3.6.1.4.1.5923.1.1.1.6 (eduPersonPrincipalName) are used.
- convRules: This table contains information about an attribute conversion rule. Besides a unique identifier of the rule, its status, creation date, a short description, the owner information, the location in the file system is stored. The result of the attribute conversion is expressed as target, which links to the appropriate attribute.
- metadata: Comparable to the convRules table, this table contains information about the provider entities' metadata, e. g., the unique entityID, the location of the metadata file stored on the TTP's file system, a short comment, creation data, and owner information.
- organization: Each provider can be associated with an organization.
- providers: Stores all providers and relevant information like the entityID .
- providerWhitelist and providerBlacklist: The usage of a whitelist or blacklist enables IDP or SP administrators to explicitly allow or reject certain providers and, therefore, the metadata exchange. It is based on DNS domain names and is intertwined with the validation of new entities regarding domain ownership. As the permissions to add and change data is validated and administrators can only use this functionality for their own entity, spoofing is prevented.
- providerUserRelationship: Information about the association between provider entities and users.
- ruleDependencies: Attribute conversion rules converts some input attributes into a target attribute. This table stores information about the source attributes required for conversion.
- ruleStatus: This table contains the available rule status.
- spIdPRelationsship: Stores information about the SP to IDP relationship, i. e., information about existing virtual federations.
- users: Information about the users registered at the TTP, i. e., administrators of provider entities.

Administrators of IDPs and SPs can use basic features, such as the registration of new metadata and uploading or searching for appropriate attribute conversion rules via the web interface and to further automate some management tasks by using provided command-line-tools. The GNTB's core service, therefore, provides an application programming interface (API) consisting of a number of API functions, which were described in [1]. The API function for downloading conversion rules is publicly available, as they do not contain PII. All other functions are classified as internal use only, authentication required and additionally restricted to own account or organization. For user management, the creation of new, updating or deleting existing accounts exists. Before registration of a new provider entity and uploading its metadata, it has to be verified that this entity does not already exist to avoid duplicates. Also, for the proof-of-ownership of the registered metadata or to ensure syntactically correctness of the metadata file as well as notification of administrators, the API provides appropriate validation functions.

To support the core service, the data access layer provides function to trigger the download of the metadata information. The download can be done by the Metadata Query Protocol or any other method. The extensions, installed on the IDPs and SPs, allow the automated integration of the downloaded

2015, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

metadata as well as attribute conversion rules. This results in immediate use of a service by the user.

IV. Federations in SAML Metadata Broker

The metadata is exchanged on demand between IDP and SP as described in the previous section. Therefore, as the metadata is not aggregated and then distributed as a whole any more, static federations are technically not needed. The metadata is exchanged on-demand between cooperating IDPs and SPs. When IDPs and SPs have integrated each other's metadata, dynamic virtual federations can be built. This depends on the situation, e. g., if only one IDP-SP pair cooperates, it is a bilateral federation. If more IDPs and SPs cooperate, they can dynamically build a federation. The concept of dynamic virtual federations is described in this section, followed by the design of a federation administration tool for those more fixed federations, which require opt-in. Both, dynamic virtual federations and the federation administration tool, are available with an extended version of the GNTB TTP.

A. Concept of Dynamic Virtual Federations

Dynamic virtual federations are built dynamically, depending on the needs of the users. The second dynamic aspect is the dynamic appearance of the federation. They are built dynamically, new organizations or single providers can join, while others can leave. The dynamic virtual federation can be closed, when the project or reason for the cooperation ends. This means that the size of the federation is dynamically adjusted. The degree of dynamics depends on the reason for the federation. While project and cooperation federations have a shorter length of life, national federations are less dynamic. If the federation is closed, e. g., due to an official project cooperation, the size of the federation will not change, whereas open federations have greater dynamics in relation to the size. Another aspect are service level agreements for financial services, which need to be in place beforehand. This also has impact on the dynamics of a federation. Virtual means the federation is orthogonal to existing static federations. The federation has members from different federations, which want to cooperate. Existing structures are suspended or weakened, in order to allow efficient international cooperation. Based on the characteristics dynamic and virtual, the defined term of national federations disappears. Hence, a federation is a cooperation of members, i. e., IDPs and SPs, which cooperate due to needs of users. The dynamic virtual federation can be characterized as follows:

- The structure of the cooperation can be an ad-hoc federation, a hub-and-spoke federation as well as an identity network.
- The amount of members is flexible. A bilateral federation is possible as well as a fixed number of participants and, most likely, a complex structure.
- The structure of the group depends on the needs of the participants. It can be open, open with restrictions and closed, although closed federations are opposed to the characteristic dynamic and, therefore, not likely.
- The dimension of the federation is open. It can be local, regional, national or international.
- The organizational dimension is intra-federation, though dynamic virtual inter-federations can be established by federations.
- The duration of the federation depends on the requirements and can be limited to the project length or fixed-terms.
- The sort of collaboration can be project, virtual organization, or by other reasons.
- The coordination depends on the requirements and can be implicit, explicit or mixed structure.
- The process of establishment is spontaneous, event-driven or as needs arise. Planned establishments are possible for, e. g., projects.
- The circle of trust can be anything but static.
- The degree of commitment is probably unwritten agreements, as long as contracts and service level agreements are not needed.
- The trust relationship between members is most likely direct, though it can be indirect as well.

When two or more dynamic virtual federations need to cooperate, dynamic virtual inter-federations can to be established. The establishment is likewise dynamic and virtual. If there are enough connects between the participating federations established, e. g., at least 20 percent of all possible connections, the TTP GNTB can automatically build an inter-federation. In between, the (inter-)federation can change, when entities opt-in, while others opt-out. If the (inter-)federation is not needed anymore, e. g., if a project or other sort of cooperation is terminated, the (inter-)federation can be closed. One precondition is the approval of the federations to build an inter-federation. If the federations do not want a dynamic inter-federation, they can use the federation administration tool to establish a static inter-federation.

B. Federation administration tool

In order to help managing federations and inter-federations requiring formal opt-in, a federation administration tool at the GNTB TTP needs to be implemented with the following functionalities, among others:

- establish a federation,
- define an application process,
- accept and reject possible members,
- establish and update policies,
- suspend members, and
- change permissions.

As policies, described in Extensible Markup Language (XML) files, need to be uploaded, changed, obeyed, and deleted, a policy management is needed. The policies are, similar to metadata and conversion rules, stored in a policy repository. Based on policies and other requirements, federation administrators can decide, if an IDP or SP is allowed to join a federation. By applying for membership in a federation, the entity accepts the policies. This also results in quality assurance similar to the current practice in national NREN federations. The core workflow is as follows:

Step 1: An entity, i. e., an IDP or SP, would like to join a federation. The desire is expressed by applying to a federation via the administrative web interface.

Step 2: The application is stored at the TTP as a status and the federation administrator is informed. The

2015, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

federation administrator checks if all requirements are fulfilled. Depending on the requirements, this step can be automated.

Step 2a: If the entity is controlled manually, the federation administrator needs to check policies, other requirements and/or audits.

Step 2b: If the federation requires a specific certificate, it needs to be issued and sent to the entity.

Step 3: The federation accepts/denies the entity. The result is stored in the database of the TTP.

The same basic workflow is used, if a federation wants to participate in an inter-federation. The federation's members should be notified about the result. If an existing federation or inter-federation decides to use the TTP, all members can be bulk imported, though they need to accept the membership officially. In order to represent the basic federation workflow with its status in the database, federations and inter-federations first need to be able to register and assign roles to administrators. Policies and other requirements must be stored or referenced in the database as well. The status of the federation respectively inter-federation is important as it can be currently added, updated or, e. g., after a project, deleted. The same appears for policies. When a policy was updated, members need to be notified and, in the worst case, checked against it. Furthermore, the status of the relationship between entity and federation as well as federation and inter-federation is crucial. This information, as stored in the database, can be used for federation and inter-federation statistics.

In contrast to the current situation with different federation tools, these pre-defined workflows minimize the problems, described by Harris [10]. The biggest improvement is made to the metadata flow problem. The upstream and downstream of metadata varies by federations. As a single tool with predefined interfaces is used and the metadata is exchanged on demand, the problem disappears. At the same time, work load from the federation administrators is shifted to the TTP. In order to allow both types of (inter-)federations, dynamic virtual and fixed (inter-)federations, these were investigated and a federation administration tool was designed. The dynamic virtual federation can reuse the federation administration tool by fully automating the workflow when a certain percentage of technical trust was established. These additional functionalities can be seamlessly integrated into the proof of concept implementation. The proof of concept implementation of the TTP and the extensions for IDPs and SPs, required by both dynamic virtual federations and the federation administration tool, is described in the next section.

V. Proof of Concept

The primary goal of the proof of concept implementation is to show the possibility that an existing SAML implementation can be extended to support the DAME protocol without breaking SAML and the interoperability between the different parties and without complicating the authentication workflow for the user. Additionally, the coexistence of federations and the dynamic metadata exchange introduced by DAME is to be shown.

The proof of concept implementation was done by extending the Shibboleth SAML implementation. Shibboleth was chosen because it is the primary SAML implementation used across European institutions, followed by simpleSAMLphp. The documentation of installing and running Shibboleth is available from different sources, like [11] and [12]. Also, high profile extensions, like uApprove [13], demonstrate that it is possible to extend Shibboleth and have many IDP administrators install your extension. Additionally, Shibboleth provides all three components that need to be extended in order to implement DAME. The Identity Provider, Service Provider, and a Centralized Discovery Service (CDS). The first and second, IDP and SP, must be extended to support the automated, user initiated, exchange of metadata. The latter is used as a discovery service, where users select their IDPs, and is extended to provide a web-based management interface of the TTP for the participating providers. The scenarios and the evaluation of the implementation is discussed later onwards in this section.

The proof of concept network running on virtual machines was also used to demonstrate and test that different versions and installation methods of the Shibboleth base software, which can be used with the DAME extensions. All machines are Debian based Linux virtual machines. As Debian 8 (Jessie) was released recently, the upgrade from Debian 7 to 8 could be tested on some machines, others are deliberately still running the old Debian version. This reflects a real wold scenario where systems are not always immediately updated to the latest version. The software versions used for testing are Apache Tomcat 6, 7, and 8 as the Java Web Application Server for the IDP as well as Apache web server 2.2 and 2.4 for the SPs. The TTP has only been tested on a Tomcat 7 server, but, as the CDS is very similar to the IDP and there were no issues running the IDP on Tomcat 6, 7 or 8, the CDS and the TTP extension are very likely to have no issues as well.

A. TTP Discovery Service

The trusted third party consists of three modules. First, the discovery service, to allow users to pick their IDP, second, the core metadata and conversion rule exchange mechanism, and third, the management interface for IDP and SP administrators to register and manage their providers. In the proof of concept implementation all three modules have been combined in an extension to the Shibboleth centralized discovery service. The CDS was chosen because it already implements the first module, the discovery service. Extending the existing implementation also made sense as the SAML discovery protocol is not modified by the extension and reusing an existing implementation decreases the chance of creating incompatible protocol versions. The only interface between the TTP and the CDS is the CDS' access all metadata registered at the TTP. The CDS also has to be notified, if a new IDP is registered or an existing one is modified. This is achieved by generating a complete metadata file that is including all registered IDPs, whenever there are changes to the available IDPs. This file can then be included by the CDS. The second and third modules, the actual TTP, are also part of the extension to the CDS, because this way the TTP can be distributed as a single extension. The installation of extensions on the CDS is very similar to the IDP, which makes installing the extension especially easy for administrators that are used to installing or updating extensions for the IDP.

Besides the CDS and the TTP, another discovery service is needed to implement DAME efficiently. The DAME protocol should only be used the first time, if the metadata of the IDP

2015, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

is not available at the SP. So, the SP or the user need a way of determining whether the metadata is already available or not. In the proof of concept, this is done via the Shibboleth embedded discovery service (EDS). The EDS is installed on the same system as the SP and a user is redirected to this EDS for IDP discovery first. The EDS can then display all IDPs that are available at the SP. If the EDS knows about the IDP, which the user would like to use, the DAME protocol is never used and a regular SAML SSO can be done. If the desired IDP is not already listed at the EDS, the user can then be forwarded to the CDS at the TTP. This forwarding is implemented by adding a button to the web page generated by the EDS that will forward the original request to the CDS at the TTP. The EDS is mostly written in Java Script and has to be configured using a separate Java Script file. The EDS does not supply a method of implementing extensions, so, to implement the necessary changes, the EDS itself needed to be modified.

In contrast, the discovery service part of the CDS does not need to be modified. The CDS can be configured to read one or multiple metadata files and then extract the required information about the name and the communication endpoint of the IDPs. After the user selects an IDP, the CDS can relay the information about the selected IDP back to the SP as it normally would. The SP is then responsible of initiating the communication with the TTP, in order to trigger the metadata exchange.

The core TTP implementation consists of a module that handles the dynamic metadata exchange according to DAME. This module is also part of the CDS extension. This allows a more efficient communication between different modules. As described in the DAME workflow in Section III-A, this module receives an authentication request from a SP and validates its signature, in order to verify that the SP is legitimately trying to contact the IDP specified in the request. Following the SAML standard, this can only be done if the authentication request by the SP is not encrypted, as the signature is placed inside the encryption, which, as the message is directed at the IDP, the TTP cannot decrypt. Encrypted authentication requests are therefore discouraged, to provide security, while transmitting the request HTTPS should be used instead. After verification, the authentication request is stored in the users session and the TTP issues its own authentication request to the IDP to verify the user actually holds a valid account at the IDP, before initiating the metadata exchange.

The metadata exchange itself is done by getting the DAME extension element from the provider's metadata. This element must be supplied and specifies the location of the communication endpoint for initiating the metadata exchange. The TTP then sends an HTTP request to this endpoint and indicates, for which EntityID metadata should be downloaded and from where. The location of the metadata, therefore, does not have to be at the TTP itself. The current implementation of the TTP only allows for this as there is no way of specifying a remote location in the management interface, but this could easily be added.

The conversion rule exchange is done similarly. The IDP determines, which attributes the SP requests from the downloaded SP metadata. If the IDP is missing some or all of them, it can then asks the TTP, if there are conversion rules available that would use the attributes the IDP can provide to build the missing attributes. The TTP replies with a XML-formatted list of rules, which could be of use. The IDP can then filter the results and pick the rule it prefers, potentially the IDP could also try to activate multiple rules in a sandbox environment until it finds the one that works best. Alternatively, a reputation system could be implemented, so that IDPs prefer conversion rules already used or issued by the federation they are in. Votes then could be cast, e.g., by federations or IDPs that successfully tested and use the specific rules; however, as reputation systems are vulnerable to misuse and conversion rules affect sensitive personal data, any implementation that runs without supervisory control constitutes a risk. We therefore plan to gather practical experiences regarding how problematic redundant conversion rules become in the real world and will then address it as necessary in future work. The conversion rule interface is done via regular HTTP calls, so that also any other tool or extension could be used to query for conversion rules.

One remaining problem related to conversion rules is that they are currently specifically designed for the Shibboleth IDP. As the IDP uses XML files for configuration, the conversion rule is an EXtensible Stylesheet Language Transformation (XSLT) file adding new XML elements to the configuration. This method is not suited for simpleSAMLphp or other IDP implementations using a different format. In the future, an abstract syntax for conversion rules needs to be designed, which can automatically generate the correct conversion rule based on the IDP software.

The last part of the CDS extension is the management interface of the TTP. For the TTP to be usable, it needs to provide some core methods for the providers and their administrators, e. g.:

- User registration: Provider administrators are able to register at the TTP, in order to manage one or more of their systems. The user management also supports multiple users being able to manage the same provider.
- Provider registration: The provider administrators can register their providers at the TTP. The registration requires a unique EntityID and can be extended by an description of the provider. Additionally, a provider can be assigned to be part of a federation.
- Provider verification: To prevent obvious misuse of the TTP, an automated method of verifying that the person registering a provider is actually allowed to do so, has been implemented. The administrator must currently place a file with a randomly generated file name on a web server at the host name of the EntityID. Other methods, like email verification, are also possible.
- Metadata management: After registering at the TTP, the provider administrators are able to upload and modify the metadata of their provider.
- Conversion rule management: Administrators of the registered providers are able to create and modify conversion rules.

To manage the user, provider, and conversion rule information, a MySQL database is used. The metadata and conversion rule files are stored on the file system and referenced in the database. The metadata files are named by calculating a SHA-1 hash of the EntityID and appending a timestamp. This way, multiple versions of a metadata file can be stored and the

2015, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

resulting file name contains only ASCII characters and is of fixed length.

B. IDP Software

In order to support the DAME protocol, the Identity Provider needs to be extended. The extension implements a new communication endpoint, which can be used by the TTP to trigger the download of new metadata, and a new metadata provider component, which manages the downloaded metadata and provides it to the other IDP components, like the authentication module.

The original proof of concept implementation was done for the Shibboleth IDP version 2, as it was the current version at that time and version 3 was not used by any production IDP. Because version 3 is now released and no longer in beta status, the DAME extension has been converted to an extension for IDP version 3. This was also a chance to improve the proof of concept implementation. The conversion of an extension between versions 2 and 3 is not trivial as much of the underlying structures and interfaces have been changed. But the version 3 extension is much easier to install and maintain from an administrator's perspective.

In general, the IDP extension adds three new modules to the IDP: The communication endpoint to receive metadata synchronization requests from the TTP, a metadata provider to manage the downloaded metadata, and a method of implementing conversion.

The communication endpoint is a relative straight forward HttpServlet in the IDP version 2 extension and a Spring Webflow in the IDP version 3 extension. To prevent misuse, it first checks if the request is originating from a trusted source. This source is identified by its IP address and has to be configured by the IDP administrator. If the request is allowed, the metadata is downloaded and passed to the metadata provider as described below. After the metadata has been synchronized, the attributes requested by the SP are compared to the attributes available at the IDP and, if some are missing, conversion rules are requested from the TTP.

The IDP is designed to support multiple metadata providers. Two general examples of the metadata providers, the IDP is shipped with, are file based metadata providers, that just read a local metadata file placed on the IDP by an administrator, and HTTP metadata providers, which periodically download the metadata file from a remote location and cache it locally. A special metadata provider is the chaining metadata provider, it can be used to combine multiple other metadata providers together. The IDP extension adds another type, the DAME metadata provider.

In the IDP version 2 extension, the DAME metadata provider was just able to read the files, which were downloaded via the DAME protocol. The version 3 extension is more advanced as it is basically a chaining metadata provider and uses a file backed HTTP metadata provider for each SP. The key difference to the chaining metadata provider is that the DAME metadata provider can be modified during runtime of the IDP. This is necessary to dynamically add new metadata. If a new metadata file is synchronized using the DAME protocol, the URL of the metadata is saved in a local file. The filename is the SHA-1 hash of the EntityID and the extension ".xml.loc" designates that this is the file containing the original location of the metadata file. The file based approach has been chosen over a database to keep the number of dependencies small. Those URLs are necessary to initiate all file backed HTTP metadata resolvers if the IDP is started. The metadata itself is stored by the file backed HTTP metadata resolvers as the SHA-1 hash of the EntityID and the extension ".xml". The local copy of the metadata ensures that the metadata is always available even if the TTP or the entity hosting the metadata cannot be reached.

The conversion rule synchronization mechanism of the IDP extension creates a backup of the relevant configuration files "attribute-resolver.xml" and "attribute-filter.xml". The downloaded conversion rule XSLT file is then applied to the "attribute-resolver.xml" file. The XSLT can lookup the XML id of other attributes it depends on to reference them properly. The "attribute-filter.xml" file is extended by a XSLT file, which is distributed with the IDP extension. It is used to limit the release of the converted attribute to the SP, which requested the attribute. After modifying these files, the related IDP components need to be reloaded for the changes to become active. The extension is able to do this without restarting the whole IDP.

C. SP Software

The SP extension is written for the SP module, which can be used with the Apache web server. It is very similar to the IDP extension. Because the SP is written C/C++, while the IDP and CDS are written in Java, there cannot be a joint extension for both. The SP only exists as version 2 at the moment, so there is only one extension. Special about the SP is that it consists of two modules, which need to communicate via inter process communication. One module is included as a library into the Apache web servers processes and the other runs as a standalone daemon. This prevents the Apache web server needing to load all libraries and their dependencies, which are required to parse and process the SAML messages. For that reason, the part included in the Apache web server is called "shibd_lite", whereas the daemon that processes the messages is called "shibd".

Because the SP extension is written in C/C++, it currently needs to be build on the target SP. Unless the administrator builds the SP from source, fetching the dependencies and compiling the extension can take some time. The extension's documentation contains a description of how to build it using Debian Linux.

The extension contains a communication endpoint for initiating the metadata exchange and a metadata provider for managing the downloaded metadata. The conversion rule part does not need to be implemented for the SP, as all attribute conversion is done by the IDP.

The communication endpoint of the SP does the same checks to prevent misuse as the IDP and then downloads the metadata to a file named after the SHA-1 hash of the EntityID. The DAME metadata provider is on the same level as the metadata provider of the IDP version 2 extension. It reads all available metadata files and has methods to dynamically add new files during runtime.

D. Evaluation of the Implementation

Figure 3 shows the general setup of the environment used to demonstrate the proof of concept. It consists of multiple

2015, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

virtual machines that each were assigned specific roles and federations. On the one hand, the federation setup was used to determine the amount of work, which needed to be done for setting up a federation and to compare this to setting up the TTP, and, on the other hand, to test scenarios, where some providers were available right through the federation and others could be added dynamically using DAME.

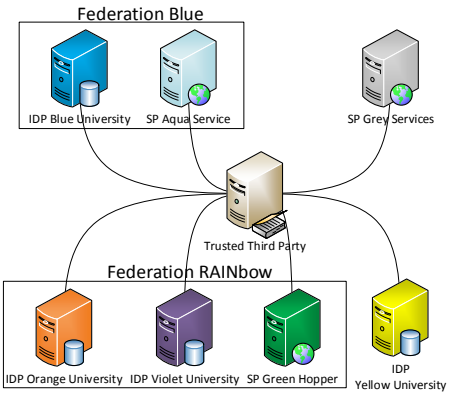

Figure 3. Overview of the proof of concept setup.

Building on the example given in Section III-A, one tested scenario included Blue University, Grey Services, and the Yellow University cooperating on the project COLORado. Marina from Blue University requests access to the SP of Grey Services, which runs a simple project collaboration tool. This tool should be used for sharing of project-related files, wiki web pages, group calendar, and has an integrated online Skype status check plugin. As both universities and the SP are not part of a common federation or inter-federation, they do not have each other's SAML metadata. Therefore, Marina chooses the federated login. Because the SP does not know her IDP, the Blue University, its embedded discovery service does not allow the selection of her IDP directly. As the organizations are set up for the GNTB, the SP's discovery service is configured to allow forwarding the discovery request to the GNTB DAME TTP. Marina chooses this option and is presented with a list of all IDPs currently available at the GNTB. After selecting her home IDP at the GNTB discovery service and subsequently authenticating there, Marina is redirected to the SP Grey Services. In the background, SP and IDP have exchanged each other's metadata and integrated it into the local configuration. A consent management tool, like uApprove, shows Marina the transmitted attributes and she needs to give her informed consent. After confirmation, she will be successfully logged-in to the collaboration tool. However, the integrated skype plugin does not yet contain Marina's Skype-ID, because the *skypeID* attribute could not be found. Marina informs her IDP administrator Azuro. Checking the SP's metadata, Azuro logs onto the GNTB web application and adds a new conversion rule that derives the *skypeID* attribute from *schacUserPresenceID*, which he knows is available at his IDP Blue University. With the conversion rule in place, Marina can use the Skype plugin as intended. We assume that at a later point in time, user Sunny from Yellow University tries to access the SP of Grey Services as well. Because both IDPs use *schacUserPrecenceID*, the attribute *skypeID* can automatically be created by re-using the attribute conversion rule from Blue University. Therefore, if Sunny chooses his IDP at the GNTB discovery service, triggering the metadata exchange. Based on the information in the SP's metadata, the correct conversion rule is downloaded and integrated into Sunny's IDP. He can directly use the Grey Services collaboration tool with the Skype plugin.

To evaluate the benefits of the GNTB extension, the test environment was setup as displayed in Figure 3. In order to measure how efficient the setup is, the manual steps for exchanging the metadata to setup the scenario are compared against the number of manual metadata exchanges needed to set up GNTB. Additionally, as an important metric it was determined how fast the metadata exchange actually could be done, as the aim was to do the exchange completely transparent to the user. The proof of concept implementation shows that the exchange can be done in under two seconds. However, it has to be noted that in this case all hosts are on the same virtual network and that real world usage would see more latency in the communication between servers. To ensure that the original authentication request from the SP is only forwarded to the IDP after the metadata has been exchanged and both providers have reload their configuration, a 10 second delay has been implemented. This should be more than enough time for the providers to finish reloading, while not being overly annoying to the user. The user also only needs to wait those extra 10 seconds if she is the first to use the specific SP–IDP combination. A more refined procedure, in which the current status is being periodically polled, will be added to minimize waiting times for users in real-world deployments, making the overall system more robust regarding latencies of any kind, including, e.g., delays due to insufficient Internet connectivity of mobile users.

To test how many manual metadata exchanges would be necessary, three different scenarios, in which GNTB could be used, are analyzed. The following description of scenarios and evaluation does not specifically include the amount of extra work that is required to install the necessary extensions at each provider. This amount of work is not specific to the DAME protocol but to its implementation. The installation and configuration of the DAME extensions could be heavily automated and is time efficient with the version 3 IDP. Table I summarizes the results.

1) Intra-federation: Within a federation GNTB could be used to exchange the metadata of the federation members. In large federations, this could improve scalability because only small subsets of identity and service providers ever need to exchange metadata and communicate with each other. The federation could deploy their own GNTB instance and reuse existing infrastructure, to get recent metadata files from their members.

To build a federation, like "Federation Blue", in the test environment, the members, i. e., Blue University, and Aqua Service, need to apply at the federation for membership and send their metadata to the federation. Both providers need to go through this procedure. To be more general, all n providers of a federation first need to register by sending their metadata to the federation. Afterwards, each provider must add the federation's metadata to its configuration, another n operations. In total $2n$ metadata exchange operations by all provider administrators. If the federation would be using GNTB, the number of operations would be less. Each provider has to register at the TTP (n

2015, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

operations), but only the IDP providers would need to add the TTPs metadata to their configuration. If n_{idp} is the number of IDPs, a total of $n+n_{idp}$ operations would be needed to set up an environment with n providers. n_{idp} cannot be greater than n, the total amount of providers, thus $n + n_{idp} \leq 2n$ and the GNTB approach would reduce the overall work needed to be done to setup a federation. In the proof of concept environment shown in the figure $n = 2, n_{idp} = 1 : 2 + 1 < 2 \cdot 2 \rightarrow 3 < 4$

2) Inter-federation: Between multiple federations, the aggregated metadata file, which contains the metadata of all members, is even bigger than in federations, thus the amount of never used metadata is even higher. In this scenario, the federations could pass the metadata of their members to the GNTB instance of the inter-federation, which would be easier than all providers sending their metadata to their federation and the inter-federation. One possibility for growing federations and inter-federations is the use of a still to developed distributed GNTB, run by all participating federations.

Suppose the federations Blue and RAINbow want to join an inter-federation BlueRAINbow. The easiest way would be for the federation managers to send the federation metadata to the inter-federation GNTB instance and include the TTPs metadata in their own metadata distribution. This would be independent of whether the federations are using GNTB or the classic metadata aggregation. This would lead to $2i$ operations if i is the number of federations joining the inter-federation. From a technical view, this requires the same amount of metadata exchanges whether GNTB is used or not. Unfortunately, the TTP implementation does not support any bulk provider registration yet. This would be necessary if a federation would like to add all providers at once. With the current implementation $\sum_{x=1}^{i} n_x$ providers would need to register at the TTP and $\sum_{x=1}^{i} n_{idp_x}$ IDPs would need to integrate the TTPs metadata. This would result in $i = 2, n_1 = 2, n_2 = 3, n_{idp_1} = 1, n_{idp_2} = 2 : (2 + 3) + (1 + 2) = 8$ necessary metadata exchanges from the figure scenario.

3) No federations: Without any prior federations, each provider must register and upload its metadata to a GNTB instance itself, as described above in the COLORado example. As there is no existing infrastructure, it must be decided who runs a GNTB instance that can be used in that way. For example, larger projects or projects with much fluctuation of members could setup their own instance to manage the exchange of metadata between the members. This example uses the two federation-less providers SP Grey Services and IDP Yellow University as well as IDP Blue University to get a large enough test case.

Without a TTP and a federation, each provider would need to include everyone else's metadata and send its own metadata to everyone else, which would result in $2n(n-1)$ operations. With 3 providers, there are already $2 \cdot 3(3-1) = 12$ metadata exchanges. When a TTP is set up, this situation basically becomes the intra-federation case, which needs only $n + n_{idp} = 3 + 2 = 5$ metadata exchanges.

4) Summary of the Evaluation: Table I shows the manual metadata exchanges needed as described above. In the formula, n is the number of providers participating in building a federation, n_{idp} the number of IDPs in a federation, and i the number of federations joining a inter-federation. It is shown that in all cases, with the exception of the inter-federation case, using GNTB requires less manual metadata exchange operations and is, therefore, easier to maintain for administrators and quicker. The inter-federation case could be improved for GNTB to be equally good as the manual method by implementing the mass import of providers.

TABLE I. Comparison of manual and GNTB metadata exchange operations

	Manual	GNTB
Intra-federation	$2n$	$n + n_{idp}$
Inter-federation	$2i$	$\sum_{x=1}^{i} n_x + \sum_{x=1}^{i} n_{idp_x}$
No federations	$2n(n-1)$	$n + n_{idp}$

Because GNTB aims to make the metadata exchange more dynamic and remove the fixed structures of federations by using virtual federations, any combination of the scenarios above can be represented. A provider can be a member of multiple GNTB instances, so that it can be part of a project GNTB and of the GNTB of its federation and/or inter-federation. In order to reduce the amount of registrations, these distributed GNTB instances should cooperate. The then extended core workflow and the register of the GNTB instances still need to be developed. If the shortcut, which allows federations to add all their members in a bulk operation for use in an inter-federation scenario, is implemented, each test case is equal to or better than the currently used approaches with regards to the number of manual metadata exchanges necessary. This is also true in the case that a mix of the scenarios needs to be represented as this does not add any metadata exchange overhead. But not only the number of manual metadata exchanges is the same or smaller, the size of the metadata files is reduced as well. The inter-federation BlueRAINbow would, e. g., normally aggregate all metadata, which means at least 5 providers per metadata set. If, as an example, only IDP Blue University and IDP Orange University cooperate with SP Green Hopper the size contains with GNTB only 2 entities for the SP, while 1 for both IDPs.

VI. Risk Management

In preparation of pilot phase of the GNTB prototype and to achieve a technology readiness level TRL7, security-related questions have to be answered. Following existing good practices and international standards, e. g., ISO/IEC 27001, a risk assessment takes place. As presented in [14], we operationalize and support this continuous management process by applying our risk management template. Because GNTB allows the trust establishment between authentication and authorization infrastructure components, which are used to store, exchange, and process personally identifiable information by the user's IDP and an arbitrary SP, assuming that both are registered at the TTP, the criticality all of these have to be set to (very) high and implementing appropriate security measures is nearly unavoidable.

The first step in risk management, establishing the risk management context, requires the definition of primary and secondary assets. Primary assets are usually the core business processes and workflows of an organization. In this case, GNTB makes the immediate access to online services provided by previously unknown or untrusted SPs possible and thus could be seen as an enabler and innovator for the Research and Education community to collaborate and share data across

2015, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

organizations and national or federation borders. The more technical, secondary assets support these processes and usually are categorized as the used hard- and software, the information exchanged and processed by the service's components. Operationalizing risk management focuses on the technical GNTB components: IDP and SP software extension, the TTP, the processed PII, and the exchanged metadata information and attribute conversion rules with all their dependencies to internal components like used databased, the underlying network infrastructure, libraries and operational details regarding the well-known objectives (confidentiality, integrity, and availability).

Possible events threatening GNTB's components are, e. g., flooding the TTP with metadata exchange requests. Describing such an event, using our threat scenario template, we have external actors triggering this harmful event, the threat type or category is malicious and the attacks aim at the violation of the availability in form of a complete service interruption and limitation of the services' usability. Another threat could be the download faked metadata information to compromise the IDPs or SPs and to lead them to trust each other and release sensitive user data to a malicious SP.

Assessing these example threat events, their likelihood as well as impact, especially the latter one from a privacy perspective, must be seen as high. The high risk value resulting requires further action. So, to overcome the first threat, GNTB requires a user authentication before the metadata exchange will be triggered as well as the SP can check due to sending back information about the user's IDP selection, if there was a previous access request, and, finally, an integrated rate limiter slows down the number of allowed requests to the TTP. Countermeasures to solve the second issue are, e. g., that the IDP checks if the source IP address of the metadata trigger message, as described in Section V-B, corresponds to that one configured by the IDP's administrator and prevents download metadata information from faked TTP instances; given that any transport is based on TCP/IP and successfully completed TLS handshakes, primitive attacks such as source IP address spoofing are not explicitly addressed here. Furthermore, the validation of the entity also prevents the registration of faked entities.

By listing all assets, analyzing the risks of all components and the dynamic metadata exchange itself, all possible risks were regarded. Based on the risks, possible attacker models and counter measurements, i. e., technical, organizational respectively preventive, detective, and responsive, were inspected. This lead to a protocol, which is as secure as possible, and to a secure designed GNTB TTP. Nevertheless, the local software and the TTP need to be monitored. Further risks can be mitigated by control by federation operators and the use of assurance frameworks.

VII. Conclusion

The DAME on-demand internet-scale SAML metadata exchange enables user-triggered exchange of metadata between IDPs and SP across current federations' borders. Furthermore, it enables the re-use of conversion rules, in order to further automate and accelerate the technical trust establishment. Last but not least, the scalability of the metadata exchange in federations and inter-federations is improved. The approach GÉANT-TrustBroker supports the fully automated technical setup of FIM-based authentication/authorization data exchange. Therefore, it increases the automation and scalability of former manual implementation steps by administrators. Consequently, the users can immediately use a new service.

While the DAME workflow allows the metadata exchange between IDP and SP, which are not part of a federation, but form a dynamic virtual federation, the federation management tool helps federation operators to formally establish federation, where official opt-in is required. The approach GÉANT-TrustBroker was implemented extending the SAML implementation Shibboleth and evaluated based on several scenarios. The implementation shows a scalable approach for SAML metadata exchange, where the duration of the metadata exchange is convenient for the end user. The amount of metadata exchanges is the same or smaller. At the same time the size of the metadata file is reduced. In order to have a secure service, the risk management was applied in Section VI and taken into account during the design of the GNTB. The current state of the protocol has been submitted as an Internet-Draft to the IETF to initiate a standardization process; a second implementation based on SimpleSAMLphp is currently being worked on. With its first international setup being deployed as a part of the eduGAIN service operated by the pan-European research and education network GÉANT, practical experiences with a large number of participating organizations and users will be gathered over the next few years.

Further research topics relate to the level of assurance respectively the trust between two entities. Though the technical trust is exchanged via the metadata, the quality of the entity could be assured or estimated by a level of assurance. As explained above, an abstract format for conversion rules would help to make these rules usable for different implementations. Furthermore, distributed TTPs should be investigated in order to have cooperating TTPs as it is not likely that only one TTP is operated worldwide.

Acknowledgment

The research leading to these results has received funding from the European Community's Seventh Framework Programme under grant agreement no 605243 (Multi-gigabit European Research and Education Network and Associated Services — GÉANT).

The authors wish to thank the members of the Munich Network Management (MNM) Team for helpful comments on previous versions of this paper. The MNM-Team, directed by Prof. Dr. Dieter Kranzlmüller and Prof. Dr. Heinz-Gerd Hegering, is a group of researchers at Ludwig-Maximilians-Universität München, Technische Universität München, the University of the Federal Armed Forces, and the Leibniz Supercomputing Centre of the Bavarian Academy of Sciences and Humanities.

References

[1] W. Hommel, S. Metzger, and D. Pöhn, "A SAML Metadata Broker for Dynamic Federations and Inter-Federations," in Proceedings of INFOCOMP 2014, The Fourth International Conference on Advanced Communications and Computation. IARIA, 2014, pp. 132–137.

[2] "SWITCH – AAI Resource Registry," 2015, URL: https://rr.aai.switch.ch/ [accessed: 2015-07-28].

[3] "PEER 0.20.0: Python Package Index," 2015, URL: https://pypi.python.org/pypi/peer [accessed: 2015-07-28].

2015, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

[4] I. A. Young, "Metadata Query Protocol – draft-young-md-query-05," Work in Progress, 2015.

[5] A. Bhargav-Spantzel, A. Squicciarini, and E. Bertino, "Trust Negotiation in Identity Management," IEEE Security and Privacy, vol. 5, no. 2, Mar. 2007, pp. 55–63.

[6] P. A. Cabarcos, F. Almenárez, F. G. Mármol, and A. Marín, "To Federate or Not To Federate: A Reputation-Based Mechanism to Dynamize Cooperation in Identity Management," Wireless Personal Communications, 2013, pp. 1–18. [Online]. Available: http://dx.doi.org/10.1007/s11277-013-1338-y

[7] M. S. Ferdous and R. Poet, "Dynamic identity federation using security assertion markup language (saml)," in Policies and Research in Identity Management. Springer Berlin Heidelberg, 2013, pp. 131–146.

[8] W. Hommel, S. Metzger, and D. Pöhn, "Géant-TrustBroker: Dynamic, Scalable Management of SAML-Based Inter-federation Authentication and Authorization Infrastructures," in ICT Systems Security and Privacy Protection. Springer Berlin Heidelberg, 2014, pp. 307–320.

[9] W. Hommel, S. Metzger, and D. Pöhn, "Project GÉANT-TrustBroker – Dynamic Identity Management across Federation Borders," in Networking with the World, The 30th Trans European Research and Education Networking Conference, 19-22 May, 2014, Dublin, Ireland, Selected Papers. TERENA, 2014.

[10] N. Harris, "The Interfederation Problem," 2014, URL: https://blog.refeds.org/a/201 [accessed: 2015-07-28].

[11] "Shibboleth – Installation," 2015, URL: https://wiki.shibboleth.net/confluence/display/SHIB2/Installation [accessed: 2015-07-28].

[12] "SWITCH – Support – SWITCHaai," 2015, URL: https://www.switch.ch/aai/support [accessed: 2015-07-28].

[13] "SWITCH – uApprove – User Consent Module for Shibboleth Identity Providers," 2015, URL: https://www.switch.ch/aai/support/tools/uapprove [accessed: 2015-07-28].

[14] W. Hommel, S. Metzger, and M. Steinke, "Information Security Risk Management in Higher Education Institutions: From Processes to Operationalization," in Proceedings of the 21th congress of the European University Information Systems Organisation. EUNIS, 2015, pp. 190–201.

2015, © Copyright by authors, Published under agreement with IARIA - www.iaria.org