



Project Moonshot

Briefing paper for IETF 77, Anaheim.

Josh Howlett, JANET(UK)

Sam Hartman, Painless Security, LLC

18 March, 2010

Image © Luc Viatour (<http://www.lucnix.be>)

This page left intentionally blank

Abstract

Project Moonshot is an effort to address a number of issues that have been observed with SAML-based federation. These include: the difficulty of using a federated identity with applications that are not HTTP user agents; the so-called 'Identity Provider discovery' and 'Multiple Affiliation' problems; and inter-federation.

Project Moonshot proposes an architecture for addressing these issues; in addition, it proposes a number of other novel benefits that may improve the ease of using and deploying SAML-based federation.

Project Moonshot is a direct response to growing experience with SAML federations within the Research and Education networking community. This paper presents JANET(UK)'s understanding of the use cases that have emerged in that community along with an overview of the Moonshot technical approach. This paper is presented together with preliminary specifications and an analysis of the technical feasibility. Together, these contributions solicit feedback and interest from IETF and other interested technical communities.

Table of Contents

1. Introduction.....	5
2. Use-cases.....	6
2.1. Learning from SAML Web SSO.....	6
2.2. Service Outsourcing.....	6
2.3. High Performance Computing.....	7
2.4. Entity trust establishment.....	7
3. Expected benefits.....	8
4. Proposed architecture.....	10
4.1. An overview and comparison with eduroam.....	10
4.2. The SAML EAP Profile.....	11
4.3. Composing the EAP SAML Profile.....	12
4.3.1. Beyond Web SSO.....	12
4.3.2. Scalable Trust.....	12
5. Scalable Trust.....	13
5.1. Moonshot Trust 101.....	13
5.1.1. Step 1: Fully collapsed.....	13
5.1.2. Step 2: Basic federation.....	13
5.1.3. Step 3: Introducing the Trusted Third Party.....	14
5.1.4. An interlude: the Key Negotiation Protocol.....	15
5.1.5. Step 4: Using the KNP to establish dynamic AAA relationships.....	15
5.1.6. Step 5: Recursive discovery.....	15
5.2. Application of the Key Negotiation protocol to the 'Entity trust establishment' use-case.....	16
6. Implementation.....	18
6.1. Client modifications.....	18
6.2. Service modifications.....	18
7. Moonshot planning.....	19
8. Conclusions.....	20
9. Acknowledgements.....	21

1. Introduction

"[I]f you go for a complete client stack revamp [...] then I would shoot for the moon."

Scott Cantor, REFEDS mailing list, 22 October 2009.

The TERENA EMC2 task-force¹ is a group composed primarily of representatives from the European National Research and Education Network (NREN) community, but also from other similar organisations and communities from most other world regions. The task-force has a work item called "Beyond Web SSO" that is charged with investigating the use of federated identity in applications that do not require the use of an HTTP user agent, such as a web browser. The discussion to date has focussed on identifying and developing use-cases, rather than proposing specific technical approaches.

JANET(UK), the NREN for the United Kingdom, has undertaken work to develop such a technical approach. This work was presented at the "Beyond Web SSO" Birds of a Feather session at TNC 2009². Significant progress has since been made in developing this work, and it is now felt that this work is sufficiently advanced to bear the consideration of an experimental implementation. Within JANET(UK), this item of work is known as "Moonshot".

The goal of Moonshot is to develop a general approach for associating a federated identity with arbitrary Internet protocols. This is intended to address use-cases from two categories of general problems:

- I. **Beyond Web SSO:** federated identity for applications that are not bound to HTTP user agents.
- II. **Scalable Trust:** flexible and scalable trust establishment between network entities.

While these categories of use-case may appear unrelated, Moonshot contends that they can both be modelled similarly; and subsequently addressed using the same technical approach.

Early modelling suggested a number of benefits for the actors that participate in federated identity systems. These were often surprising and, on occasion, implausible. Therefore, prior to investing significant effort, JANET(UK) decided that it would be prudent to solicit the opinion of an independent expert (Sam Hartman of Painless Security LLC) to confirm the feasibility of the technical approach and the proposed benefits.

Moonshot and the technical feasibility report³ were presented at the Vienna meeting of TERENA EMC2 task-force in February 2010. Based on feedback received at that meeting, JANET(UK) is presenting the proposal to a broader technical community for review. This paper describes the motivating use cases and an overview of the architecture. The feasibility analysis describes how the proposal relates to existing technologies and to similar efforts. Preliminary specifications are available for several components of the system.

The technical feasibility of Moonshot is only one of the aspects of the proposal that requires consideration. The technical approach demands modifications to the client and server stacks that might themselves be infeasible for more practical reasons, such as the business considerations of the parties that provide those parts of the stack or manage them on behalf of the end-users. These modifications are discussed in the feasibility analysis. Therefore, the feasibility of Moonshot can be usefully discussed in the light of these types of considerations, and not only on its technical properties.

JANET(UK) believes that any solution to the problems discussed in this document will require acceptance from a broad technical community: standard development organisations, vendors, deployers and so forth. While the NREN community may be able to successfully implement a solution independently, an independent solution is likely to fail to obtain traction more widely. It will probably be replaced by another solution that may be less optimal for the use cases motivating the independent solution.

Therefore the goal of this paper is to provide a high-level summary of Moonshot in order to inform the discussion in that broader technical community, in the hope this community can work towards a community-wide consensus on this – or some other proposal – that might provide sufficient impetus to develop a solution that meets JANET(UK)'s use cases, as well as any other use cases identified in the broader discussion.

¹ See <http://www.terena.org/activities/tf-emc2>

² A recording of the presentation is available at <http://www.surfmedia.nl/medialibrary/item.html?id=0166XS8ba9lkhOhGxXqPOGCp>

³ The Technical Feasibility Analysis is available at <http://www.painless-security.com/blog/2010/02/12/moonshot1>

2. Use-cases

“To infinity, and beyond!”

Buzz Lightyear

This section describes the initial proposed target use-cases for Moonshot. It is expected that these use-cases should all be satisfied using the same technical approach. In the present phase of the project, these use-cases are being continually developed and expanded, and supplemented with explanations of how the proposed technical approach can be applied to address these use-cases.

2.1. Learning from SAML Web SSO

One of the motivations of Project Moonshot is to learn from experience of existing SAML federations. Two problems have emerged from the application of SAML in large-scale Web SSO implementations; Moonshot intends to provide solutions to these problems in the use case categories it covers.

The first problem is that of multiple affiliations. Today, when users wish to use their SAML identity, they are directed to a “Where are You From” (WAYF) service⁴ (sometimes known as a Discovery service). That site typically remembers their most recent identity provider and directs them there. The problem is that users who have multiple affiliations and thus multiple identities don't have good mechanisms for choosing the correct identity. For example, in the case of the UK Access Management Federation⁵ (which is operated by JANET(UK)) it is not unusual for users to have identities from two organisations (for example, if they teach at one Institution and are enrolled on a course at another). As the scope of this federation increasingly expands to cover the K12 sector, these users may also be issued with identities as parents (for example, to obtain access to a child's attainment record). The privileges that are accorded to these identities may be quite different, and consequently users must take care to ensure that they are using the appropriate identity for a particular context. Research indicates this problem will become more significant as federations grow; for example, a recent JISC study⁶ concluded that “It is likely that the issues associated with Multiple Affiliations will become more acute”.

The second problem is the identity discovery problem. When a user connects to a relying party, that party must determine which identity provider to contact for assertions pertaining to the user. Identity provider discovery is typically performed by the relying party re-directing the user to the WAYF, which presents a list of all identity providers in the UK Access Management Federation. This currently includes hundreds of identity providers which presents significant usability issues. These are likely to become increasingly pronounced as the federation expands in scale and scope, and inter-federates with other NREN operated federations⁷.

In addressing the Moonshot use cases, JANET(UK) wishes to develop a solution that does not suffer from these disadvantages.

2.2. Service Outsourcing

Many universities are outsourcing services such as e-mail, instant messaging and calendar to providers such as Google or Microsoft. During a meeting to discuss federated authentication, a representative from an outsourcing provider predicted that a majority of universities will outsource their mail or other services within the next few years. Outsourcing providers typically do support SAML for web access to e-mail and other content⁸.

Providers also frequently support protocols such as IMAP and XMPP. To use these clients, the providers typically require that a database of credentials be synchronized to the provider. There are obvious security disadvantages to this approach. In addition, synchronizing password updates can be difficult. Support costs are increased when it takes time for password resets to propagate to an external provider.

4 For example, see <https://spaces.internet2.edu/display/SHIB/DiscoveryService>

5 See <http://www.ukfederation.org.uk>

6 See <https://gabriel.lse.ac.uk/twiki/bin/view/Projects/MultAffiliations/>

7 For example, see <http://www.geant.net/Services/EndUserApplicationServices/Pages/eduGAIN.aspx>

8 For example, see <http://code.google.com/googleapps/faq.html#auth>

These universities would benefit from the ability to use federated authentication from IMAP, XMPP and calendaring clients.

An additional requirement is to support federated identity management for authorization of collaboration between organisations. Consider a user from one institution who wishes to authorize access to a resource for a user from another institution. For web access, things tend to work relatively well. However for client access, it works less well if the authorized institution does not use the same provider as the authorizing institution. The provider is unlikely to have passwords for clients from the authorized institution since that institution normally does not use the provider. On the other hand, if federated authentication were used, the provider would not need prior knowledge of the account.

This use case belongs to the 'Beyond Web SSO' category.

2.3. High Performance Computing

The UK High Performance Computing (HPC) community has expressed an interest in two federated use-cases. First, federated authentication may be an important part of business continuity planning. When one data centre fails, it would be desirable to use resources of other data centres to offset computational losses. The second reason is to support HPC computing as a service. Under this use case, organisations who do not have their own computational resources could purchase access to resources from another institution.

Both of these use cases take advantage of federated authentication because they leave account provisioning to the organisation performing authentication. The organisation providing continuity or HPC as a service need not pay the costs of managing or synchronizing provisioning information except for its own users. Auditing mechanisms need to be robust enough to support cost recovery.

Access to computational resources is typically accomplished using Secure Shell (SSH). To support the business continuity requirement, access to storage would also need to be federated. Protocols used to access storage include NFS, CIFS, WEBDAV, and a variety of clustered file-systems.

For SSH, the client and server modifications are likely to be trivial. In addition, SSH is an important part of the system administrator's tool kit and may find other interesting applications in addition to console access (remote invocation of scripts, file transfer, etc.). Finally, this use case provides evidence of the application of federated identity in the absence of an HTTP user agent, improving confidence in the proposed approach.

There is also a perception that the X.509 technology that is commonly used to manage access to HPC systems imposes significant burdens on both the users of these systems, and also the administrators who may be required to undertake identity management and certification authority responsibilities that they are not properly resourced to perform. The central IT function of the institution is already performing the identity management responsibilities as part of delivering the institution's federation services, and there is a desire to make use of that rather than operate a special purpose authentication and authorisation system.

This use case belongs to the 'Beyond Web SSO' category.

2.4. Entity trust establishment

The goal of this use-case is to address the issues associated with the key and metadata management strategies that are commonly used to establish trust between entities in large-scale federated systems, such as the NRENs' federations. This use-case belongs to the 'Scalable Trust' category. The solution to this use-case should address:

- establishing the trustworthiness of metadata and the entity that it describes in real-time.
- support for any arbitrary and untrusted metadata distribution method.

This use-case has been prioritised for two reasons. First, the trust-related issues are likely to become more pronounced as federations grow and inter-federate. Secondly, if we are successful in federating non-web applications then these new entities will further compound the problem.

Several other potential use-cases have been identified, but these have not been prioritised pending more explicit user requirements.

3. *Expected benefits*

"I can't believe that!" said Alice.

"Can't you?" the queen said in a pitying tone. "Try again, draw a long breath, and shut your eyes."

Alice laughed. "There's no use trying," she said. "One can't believe impossible things."

"I dare say you haven't had much practice," said the queen. "When I was your age, I always did it for half an hour a day. Why, sometimes I've believed as many as six impossible things before breakfast."

'Alice in Wonderland', Lewis Carroll

This section enumerates the expected benefits from the proposed architecture. As to be expected from a proposal in the early stages of development, there are varying degrees of confidence in the technical feasibility of these. An estimated level of confidence is offered in parentheses.

- Users
 - Sign-on using one or more identities to desktop applications that support the technology (high).
 - Selection of an identity using an "identity selector", achieved by extending existing technology that is already widely deployed, addressing the so-called "discovery" and "multiple affiliation" problems (high).
- Institutions
 - Permits users to use federated identity with a range of services, improving the usability of these services and reducing the effort required to support different authentication technologies and credentials (high).
 - Addresses or significantly mitigates the effects of some of the problems associated with the conventional Web SSO profiles, including the "multiple affiliation" problem by providing a user friendly and manageable system for selecting an identity (high).
 - Increases the ROI made in federated identity services, by expanding its use to a greater range of applications (high).
- Service Providers
 - Introduces the benefits of SAML-based federated identity to new types of services (high).
 - Addresses or mitigates the effects of some of the problems associated with the conventional Web SSO profiles, including the "discovery" problem, by providing a user friendly and manageable system for selecting an identity (high).
 - The technology, when used with a web browser, could co-exist with conventional Web SSO profiles, providing a convenient transition mechanism (high).
- Federation operators
 - Permits the use of role descriptors in SAML metadata that do not include keys or credentials of any kind, or references to these (but does not require this) (medium).
 - Permits the use of unsigned SAML metadata while providing a means to demonstrate trustworthiness, including real-time revocation, of this metadata in a way that is semantically equivalent to the use of a signature (but does not require this) (medium)

- Permits the use of any kind of metadata distribution mechanism that may or may not be trusted (medium).
- SAML implementations
 - Provides a SAML-based SSO profile to incorporate to support federated identity without requiring significant application-specific profiling (high).
 - Entities can use almost any type of credential to authenticate itself; communicating SAML implementations do not need to understand each others credential types (medium).
 - Credential and key management delegated entirely outside of the SAML implementation (but does not require this) (medium).
- Standards developers
 - Provides a SAML-based SSO profile to incorporate to support federated identity without requiring significant application-specific profiling (high).

As this list shows, there is a high level of confidence attributed to those benefits associated with the 'Beyond Web SSO' category of use-cases. However, there is a lower level of confidence attributed to those benefits associated with the 'Scalable Trust' category of use-cases. The reasons for this are described in section 5.2.

It is worth recalling that 'Scalable Trust' is not a necessary dependency of 'Beyond Web SSO', and that merely satisfying the latter category would be a significant step forwards. Nonetheless, for the reasons given in section 2.4. addressing both categories is nonetheless highly desirable.

4. Proposed architecture

"We shape our buildings; thereafter they shape us."

Winston Churchill

This section describes the Moonshot architecture at a level of detail that should provide readers conceptually familiar with the OASIS SSTC's SAML specifications and the IETF's AAA-related specifications with sufficient understanding of the proposal to allow a degree of technical analysis.

4.1. An overview and comparison with eduroam

Moonshot's federation approach borrows successful strategies from eduroam⁹, a network access federation within the NREN community. Eduroam is a federation of RADIUS servers run by hundreds institutions across approximately 50 countries that provides network access to roaming users who can use the credentials issued to them by their "home" institution to obtain seamless access to the "visited" institution's network.

Figure 1 below illustrates this re-use and the high-level similarities between eduroam and Moonshot.

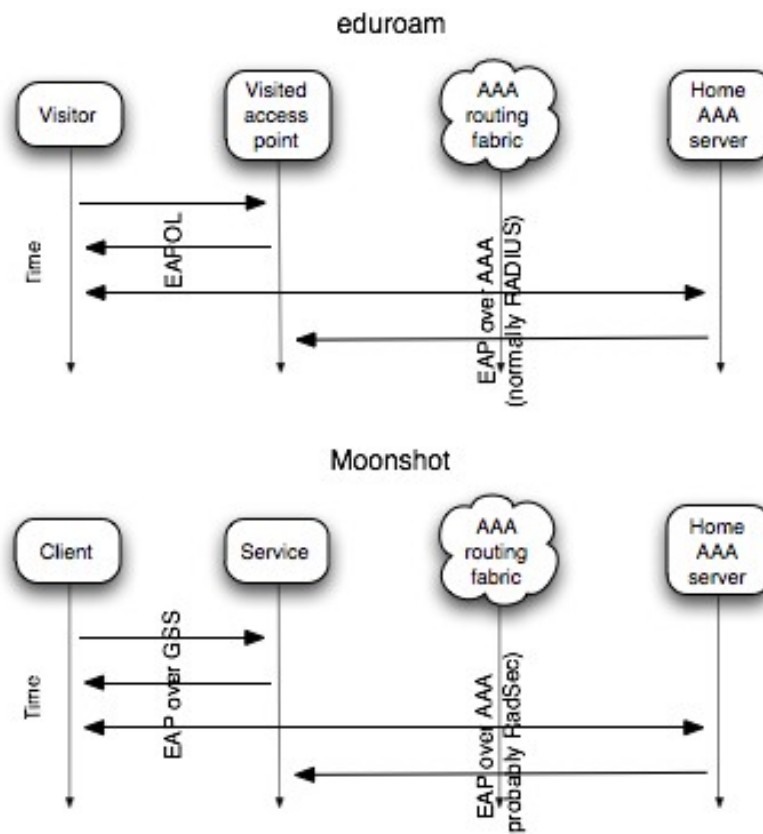


Figure 1: Comparison between eduroam and Moonshot

This high-level comparison can be misleading, and it is essential to note the following points:

- Moonshot is not dependent on eduroam, in the sense that it is totally independent of the type of connectivity.
- The meanings of "client" and "service" depicted in the diagram are intended in the loosest possible sense. For example, the use of these terms to denote an SSH client and server is an unambiguous

9 <http://www.eduroam.org>

way of describing these actors within a 'Beyond Web SSO' use-case. However, they might equally apply to the client and server relationship between the AAA nodes that are themselves part of the fabric that connects the SSH client and server. That is, the AAA fabric can apply the same mechanism for establishing trust between its components to establish trust ('Scalable Trust'), as the mechanism used by its connected application clients and servers to establish a federated identity ('Beyond Web SSO').

4.2. The SAML EAP Profile

The architecture proposes a common protocol exchange that is profiled in a manner that is conducive to its application in a broad range of use-cases. Further application-specific profiling may be necessary (for example, to address the unique personalisation and authorisation requirements of the application protocol), but it is expected that this type of profiling could be addressed through attribute profiles that would not modify the common protocol exchange.

There are three actors in this profile: the **client**, **service** and **trust authority** (TA). As implied at the end of the previous section, nomenclature is somewhat problematic in the Moonshot, and it is expected that these terms will be revised as the architecture matures.

In the simplest scenario, the protocol exchange is performed over the following channels.

1. The client and TA engage in an EAP authentication via the service, which acts as an EAP pass-through authenticator. EAP provides support for many different types of credentials.
2. This EAP exchange is bound to an application protocol using the Generic Security Services API (GSS-API) between the client and the service. The GSS-API provides an application-independent transport.
3. The EAP exchange is bound to any AAA transport between the service and the TA.
4. In the case of a successful EAP authentication, the TA issues a SAML assertion to the service by binding this to the AAA transport.

The architecture therefore requires the development and implementation of two key technologies:

- An EAP GSS mechanism that provides an EAP "lower layer" for GSS-API, as used in step 2.
- A binding of SAML to the AAA transport (most likely RADIUS, and by extension RadSec and DIAMETER), as used in step 4.

Some additional work to existing specifications will be required, primarily to ease the integration of these security technologies. In many cases the necessary work is either planned or under-way (for example, EAP channel bindings and GSS-API naming extensions); in the remaining cases, the required work is likely to be perceived as desirable and therefore the risk of interference is low.

The *EAP GSS mechanism* and the *SAML AAA Binding* are composed to create the *SAML EAP Profile*. Figure 2 below illustrates this composition.

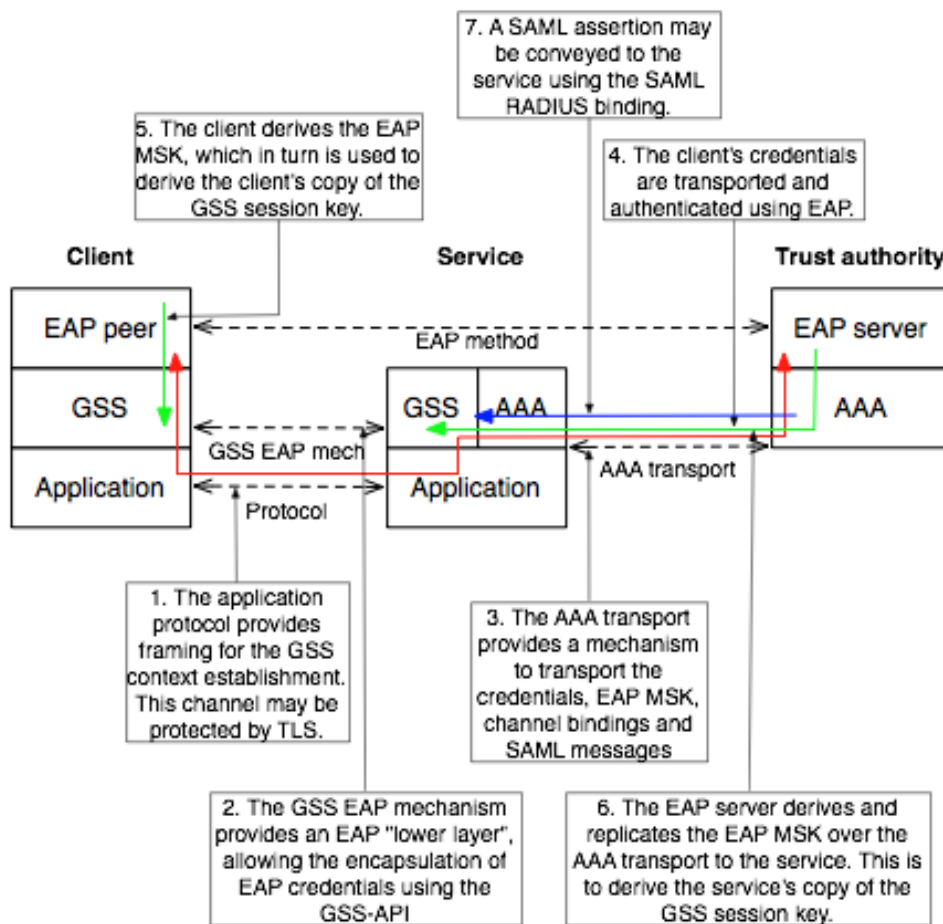


Figure 2: The SAML EAP Profile

4.3. Composing the EAP SAML Profile

The *SAML EAP Profile* is designed in a relatively abstract manner so that it can be easily composed with a diverse range of application protocols, with the intent to minimise the level of application-specific profiling.

4.3.1. Beyond Web SSO

There is a high level of confidence that the SAML EAP Profile could be composed with many application protocols (supporting GSS-API or SASL) without significant effort; this is considered particularly true of the initial target use-cases ("Learning from SAML Web SSO", "Service Outsourcing", "High Performance Computing") within this category.

It is worth noting that the client's EAP and GSS functions do not need to be implemented on the same system. These can be implemented on distinct systems, providing that there exists a trusted signalling mechanism that connects these to allow transfer of EAP packets, and any derived keys and names if authenticated. This opens the possibility of using an user's smart-phone's supplicant (or some other type of portable and personal hardware token) to drive the EAP exchange via, for example, a Bluetooth or USB connection to the terminal that the user is using. The user could use his smart-phone to select an identity when required by the remote service, even while allowing the terminal's GSS layer to establish trust in the remote service. This would not result in the exposure of the user's credentials to the terminal, and may significantly improve the user's experience of using federated identity across a range of terminals.

4.3.2. Scalable Trust

The composition of the SAML EAP Profile that is proposed to realise the Scalable Trust category of use-cases is less obvious, and is described in the following section.

5. Scalable Trust

"In general, SAML itself defines nothing related to trust management [...] In fact, absolutely nothing in the standards world addresses this. Makes writing this stuff fun."

<https://spaces.internet2.edu/display/SHIB/TrustManagement>

The composition of the SAML EAP Profile to effect 'Scalable Trust' is essentially a specialisation of the composition used to effect 'Beyond Web SSO' described in the previous section. In this case, however, the client and server actors use the EAP SAML Profile to broker AAA trust relationships (i.e., box 3 in Figure 2).

5.1. Moonshot Trust 101

This composition can be hard to understand for two reasons. First, it relies on recursion at two different layers in the stack. The reader may initially find it difficult to maintain the state of these recursions mentally, until he becomes familiar with the patterns in which they are applied. Secondly, it makes use of some of the lesser known capabilities of the GSS cryptographic tool-kit. To assist the reader in understanding the trust model, this section starts from a simple model and then incrementally ratchets up the complexity.

5.1.1. Step 1: Fully collapsed

The 'Beyond Web SSO' composition illustrated in Figure 2, in which both the client and the service share a direct AAA trust relationship with the Trust Authority (TA), is the simplest manifestation of trust. This configuration is represented in Figure 3 below.

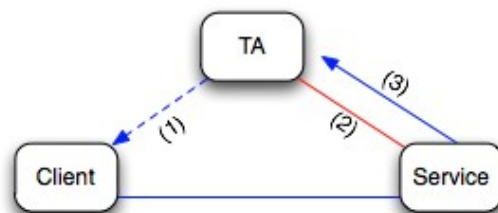


Figure 3: Simple AAA trust

The following steps occur:

1. The out-of-band establishment of an EAP credential between the TA and the Client.
2. The out-of-band establishment of an AAA credential between the TA and the Service.
3. The authentication of the Client's EAP credential to the TA, via the Service's AAA relationship with TA.

In step 3, the TA validates both the EAP and AAA credentials and asserts this to both parties. This also allows mutual authentication of the TA. Therefore, the Client and Service can both verify the identity of each other providing that they trust the TA to make this determination correctly.

As this model illustrates, a TA is a node that is trusted to make claims about the identity of other nodes that have authenticated using a AAA or EAP credential.

However, while this model is a useful explanatory tool, it is of limited practical interest because it only has a single TA. Evidently, a federated model should have two or more TAs.

5.1.2. Step 2: Basic federation

Figure 4 below illustrates the simplest federated model.

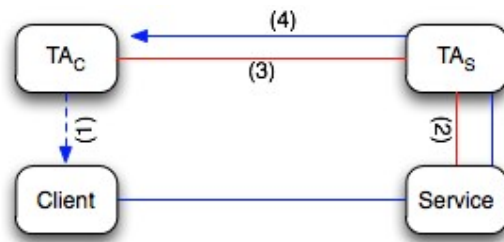


Figure 4: Simplest federated model

The following steps occur:

1. The out-of-band establishment of an EAP credential between TA_C and the Client.
2. The out-of-band establishment of an AAA credential between TA_S and the Service.
3. The out-of-band establishment of an AAA credential between TA_C and TA_S .
4. The authentication of the Client's EAP credential to TA_C via the Service's AAA transitive relationship with TA_C through TA_S .

This model is essentially the present eduroam trust model (where the proxy fabric has been collapsed). The only significant difference is that EAP, as used in eduroam, does not provide a means for TA_C to signal the identity of the Service to the Client, owing to the lack of EAP channel bindings (this is an area of work under active development in IETF, and is discussed in more depth in the *Feasibility Analysis*).

This model requires that the TAs share a trust relationship that has been established out-of-band. However, clearly this results in an exponentially expanding key management problem as the number of TAs in the system is incremented. It would be much more efficient if the trust relationship could be established dynamically between TA_C and TA_S .

5.1.3. Step 3: Introducing the Trusted Third Party

Figure 5 below illustrates how this can be achieved by reference to a mutually trusted TA (TA_{TTP}). Note that this model is a direct composition of the previous two models, except that the EAP authentication between the Client and TA_C is enabled by means of a trust relationship that has been established dynamically between TA_C and TA_S by means of an EAP authentication between TA_S and TA_{TTP} .

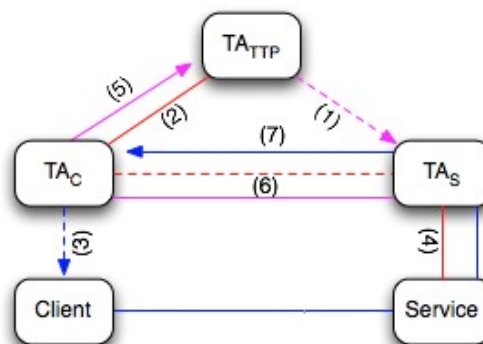


Figure 5: Using a mutually trusted TA to dynamically establish a trust relationship between two TAs.

The following steps occur:

1. The out-of-band establishment of an EAP credential between TA_{TTP} and TA_S .
2. The out-of-band establishment of an AAA credential between TA_{TTP} and the TA_C .
3. The out-of-band establishment of an EAP credential between TA_C and the Client.

4. The out-of-band establishment of an AAA credential between TA_S and the Service.
5. The authentication of the TA_S 's EAP credential to TA_{TTP} , via TA_C 's AAA relationship with TA_{TTP} .
6. The dynamic establishment of an AAA credential between TA_C and TA_S resulting from (5).
7. The authentication of the Client's EAP credential to TA_C via the Service's AAA transitive relationship with TA_C through TA_S .

The critical steps in this model are 5 and 6 where the AAA relationship between TA_C and TA_S is dynamically established. The following section describes how this is achieved.

5.1.4. *An interlude: the Key Negotiation Protocol*

The Key Negotiation Protocol (KNP) is proposed as a general mechanism for establishing a security context between two entities (mutually authenticated names and key material).

The KNP is simply the SAML EAP Profile as bound to HTTP, with minor profiling to support REST-based interactions.

The client invokes the KNP against the remote responder. The responder advertises the TAs that it trusts. The client selects an EAP credential that has been issued to it by one of the advertised TAs (if one exists) and invokes an EAP authentication exchange. If authentication is successful, both actors share a GSS security context. This context contains a session key and authenticated names for the client and responder.

5.1.5. *Step 4: Using the KNP to establish dynamic AAA relationships*

The reader may have observed that there is an element of redundancy with the introduction of TA_{TTP} . The TAs that trust it must normally possess two credentials: an EAP credential and an AAA credential. This can be reduced to a single EAP credential if the trusting TA uses the EAP credential and the MKNP (against the KNP endpoint of TA_{TTP}) to negotiate a AAA credential dynamically. This credential is derived from the GSS session key using the GSS Pseudo Random Function, and applied to the TLS PSK cipher that protects the AAA protocol.

In addition to reducing the number of credentials in the system, it also avoids the problem of agreeing a common AAA credential type between the TAs. Each TA can use whatever type of credentials are supported by the TA_{TTP} that it is associated with.

Figure 6 below illustrates the long-term credential distribution implied in the case of a single TA_{TTP} .

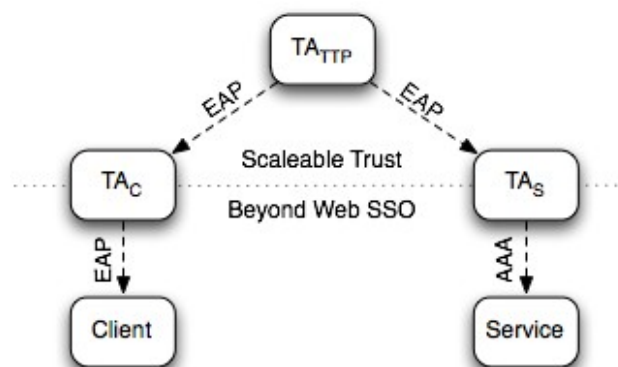


Figure 6: Moonshot long-term credential distribution

As discussed previously, the long-lived EAP credentials issued by TA_{TTP} may be used to generate short-lived AAA keys between TA_C and TA_S . These keys should be cached to avoid incurring redundant KNP exchanges. The lifetime of these keys is a local policy decision.

5.1.6. *Step 5: Recursive discovery*

It is easy to imagine the case where two TAs do not share a common TA_{TTP} , but may still wish to establish a

dynamic AAA relationship. These TAs could, using the model described above, both obtain a new EAP credential from their opposite's respective TA_{TTP} ; but clearly this may result in TAs being forced to maintain a large number of long-lived EAP credentials.

Moonshot proposes a model that allows a TA to recurse through a graph of TA_{TTP} using the KNP until it discovers a TA_{TTP} that shares a trust relationship with a TA_{TTP} with whom it also shares a trust relationship. This allows it to authenticate using the EAP credential issued by this TA_{TTP} , establishing a GSS context on the penultimate TA_{TTP} . This GSS context is used to establish a short-lived EAP credential at this TA_{TTP} , which it subsequently uses to authenticate against the previous TA_{TTP} in the chain. In this way, the TA chases back through the chain of TA_{TTP} . Figure 7 below illustrates this process.

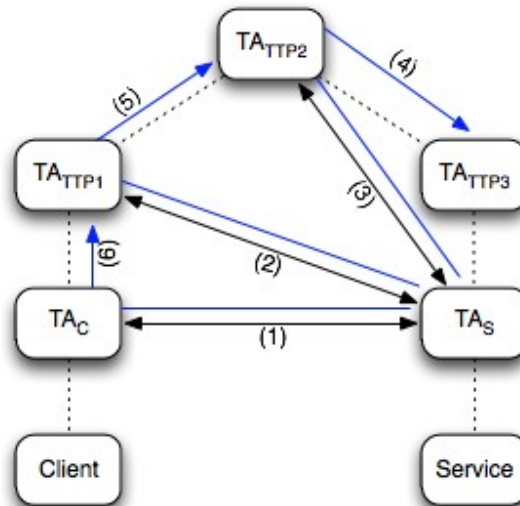


Figure 7: Recursive discovery

The following steps occur:

1. TA_S invokes the KNP against TA_C ; it fails to discover a common TA_S and begins search of the graph from TA_{TTP1} , which TA_C trusts.
2. TA_S invokes the KNP against TA_{TTP1} ; again it fails to discover a common TA and continues searching the graph by moving to TA_{TTP2} , which TA_{TTP1} trusts.
3. TA_S invokes the KNP against TA_C ; this time it discovers a common TA, TA_{TTP3} .
4. TA_S authenticates using its EAP credential against this TA, establishing a short-lived EAP credential at TA_{TTP2} .
5. TA_S authenticates using its short-lived EAP credential for TA_{TTP2} against TA_{TTP1} , establishing a short-lived EAP credential at TA_{TTP1} .
6. TA_S authenticates using its short-lived EAP credential for TA_{TTP1} against TA_C , establishing a dynamic AAA trust relationship.

Clearly, performing this process of discovery and key negotiation is expensive. However, this can be mitigated by each TA maintaining the state of the graph (from its perspective) and the short-lived EAP credentials that it has negotiated with each TA (TAs already need to maintain state for the long-lived credentials, and so this is not a new demanding implementation requirement). With this information, a TA can significantly optimise its search of the graph. For example, in the scenario described above, TA_S may already have cached information for TA_{TTP1} from a previous MKNP exchange. In this case, the search of the graph would terminate quickly.

5.2. Application of the Key Negotiation protocol to the 'Entity trust establishment' use-case

In this use-case a SAML entity, E_1 , needs to establish trust in another SAML entity, E_2 .

Entity E_1 invokes the KNP against the entity E_2 . The KNP endpoint is determined by dereferencing the entity identifier. The context resulting from the KNP exchange contains, as noted previously, names authenticated by the TTP and a GSS session key. This name is a globally unique identifier for the SAML metadata instance

that is, according to the TTP, valid for this entity. This identifier is expressed using a SAML metadata extension whose value contains a hash of the metadata (less the extension element).

Entity E_1 can consider the metadata to be “true” if it can match the identifier obtained from the GSS context to the identifier given in the SAML metadata extension, and verify the asserted hash.

This exchange has four properties that are of particular interest:

1. As described previously, a shared key may be obtained using the GSS Pseudo Random Function from the negotiated GSS context. This key may be used to protect messages exchanged by these entities; this may be used (but is not mandatory) in place of the certificates, keys, names or references thereof that are traditionally used in SAML metadata role descriptor elements. This may significantly reduce the key management and distribution issues associated with SAML federation.
2. The observant reader may have noticed that the means by which the SAML metadata is obtained by the caller of the web-service has not been described. That is because it is out-of-scope, for the reason that any kind of distribution mechanism can be used. The trust that can be attributed to metadata is entirely divorced from the means by which it has been obtained.
3. As with the situation described in section 5.1.5. , these entities do not need to share a common credential technology.
4. Recursive discovery, described in section 5.1.6. , provides a solution to inter-federation.

However, it is worth noting the following points of caution:

- The use of symmetric cryptography results in the loss of non-repudiation. There are no known use-cases where non-repudiation is necessary requirement, but it seems possible that these may exist. In these cases, this form of key management may not be appropriate.
- The use of real-time trust brokering in a SAML system is novel. There are no particular grounds for concern; only that it differs significantly from the conventional trust establishment strategies.

Further analysis of these points is warranted.

6. Implementation

“The half minute which we daily devote to the winding-up of our watches is an exertion of labour almost insensible; yet, by the aid of a few wheels, its effect is spread over the whole twenty-four hours.”

Charles Babbage

This section provides a summary of the major modifications that would be required to the client and service in order to implement Moonshot. The impact on the AAA server is not considered here, because these are relatively minor modifications. A complete list of modifications can be found in section 9.2 of the *Feasibility Analysis*; this document also provides significantly more information about the reasons for the modifications and the effort required to implement these.

6.1. Client modifications

The following modifications may be needed to the client.

1. It is expected that the majority of clients will require modifications to support Moonshot, even those that support GSS-API. This is largely due to assumptions made by clients owing to the prevalent use of Kerberos with the GSS-API. It is believed that these modifications would be straightforward or even trivial; however, making these modifications requires the co-operation of the developers of the application. These political aspects are considered much more problematic than the technical considerations. For the purposes of the planned proof-of-concept, JANET(UK) anticipates making modifications to open-source applications that are readily amenable to such modifications. As discussed in the *Feasibility Analysis*, it may be possible to circumvent this problem by making the EAP GSS mechanism 'look' like the Kerberos GSS mechanism, thereby avoiding modification to the application, but confidence is low that this would be practical.
2. The client will require software to manage identity selection, the credentials associated with these identities, and the EAP protocol exchanges necessary to authenticate these. It is anticipated that this requirement would be addressed by using a modified supplicant. All modern operating systems are provided with supplicants; however, persuading vendors to support the necessary extensions is unlikely to be trivial, particularly in light of the eduroam community's experience with the Microsoft Windows supplicant. In addition, the supplicant would need to be extended to support EAP channel bindings. For the purposes of the planned proof-of-concept, JANET(UK) anticipates using one of the open-source supplicant implementations (and most probably Open1X).
3. A GSS library supporting the EAP GSS mechanism would need to be available on the client. For the purposes of the planned proof-of-concept, JANET(UK) anticipates using one of the open-source GSS implementations.

6.2. Service modifications

The following modifications may be needed to the server.

1. As with the client (see point 1 in the previous section), many server applications would also need to be modified to support the use of the EAP GSS mechanism.
2. The service application would need to be modified to use GSS-API naming extensions to obtain any SAML assertions provided by the SAML EAP Profile (and make use of these).
3. As with the client (see point 3), a GSS library would need to be available on the server.
4. Server applications that wanted to use the dynamic trust establishment protocol would need to implement this.

7. *Moonshot planning*

"If you've done six impossible things this morning, why not round it off with breakfast at Milliways – the Restaurant at the End of the Universe!"

The Restaurant at the End of the Universe, Douglas Adams.

Project Moonshot is presently in its first major phase, which terminates in April 2010. By this point, the following deliverables will have been produced:

- *Feasibility Analysis* paper, that provides an independent opinion on the proposal.
- *EAP GSS Mechanism* Internet Draft, that will be the focus of discussion at a planned Bar BoF at IETF 77 in March 2010.
- *Use-case and application* paper, that describes the use-cases under consideration and the way in which Moonshot is expected to address these (the present paper)
- *Explanatory slide-set*, that will be used for presenting Project Moonshot to audiences.
- *Strategy paper*, that will provide an independent opinion on strategies for addressing the use-cases with Moonshot or other technical approaches.

If it proceeds, the second major phase of Project Moonshot is expected to run from April 2010 until August 2011. The focus of this phase will be the development of standards and implementations resulting in a proof-of-concept that demonstrates the initial target use-cases.

In summary, JANET(UK)

1. is actively seeking the participation of other parties for this second phase, and has already contacted some who are known to be also actively investigating similar use-cases.
2. believes that a co-operative effort is significantly more likely to succeed, through the pooling of resources, the development of consensus, and the resulting aggregate mind-share.
3. is not committed to the technical approach described in this paper, and is open to other technical approaches that achieve similar ends.

8. *Conclusions*

"That's all folks!"

Porky Pig

This paper has attempted to describe the architecture that is presently under consideration by Project Moonshot. This architecture is intended to provide a single technical approach for addressing the 'Beyond Web SSO' and 'Scalable Trust' categories of use-case.

In particular, it is believed that Moonshot could provide solutions for a range of issues associated with the use of SAML-based federated identity on significant scales. This includes solutions to the following known issues:

- Using a SAML-based federated identity for applications that are not HTTP user agents.
- The 'Identity Provider discovery' problem.
- The 'Multiple Affiliation' problem.
- Large-scale inter-federation.

In addition, it may provide the following novel benefits:

- Avoid the use for keys or key names in SAML metadata, greatly reducing the issues associated with key management and distribution, by keying from a negotiated GSS context.
- The ability to use any kind of SAML metadata distribution mechanism, trusted or not.
- Entities do not need to share a common credential technology.

Some architectural questions remain; in particular:

- Does non-repudiation for SAML protocol messages matter? If so, this may prevent the use of SAML keying from the GSS context.
- Is the real-time brokering of trust described in Moonshot appropriate for deployments?

A proof-of-concept that demonstrates the Moonshot architecture is considered feasible, but obtaining wider traction in the wider technical community, beyond the proof-of-concept, will present some challenges.

JANET(UK) welcomes comments and participation from all parties. There is also an open mailing list; to join, send an email to listserv@jiscmail.ac.uk with a body of

SUBSCRIBE moonshot-community <forename surname>

Alternatively, it is possible to subscribe at the following URL:

<https://www.jiscmail.ac.uk/cgi-bin/webadmin?A0=moonshot-community>

9. *Acknowledgements*

The authors are grateful for the support from colleagues at JANET(UK). In addition, they wish to acknowledge the following individuals who have contributed directly or indirectly to this document:

Scott Cantor, Ohio State University
Antonio Gomez Skarmeta, University of Murcia
Leif Johansson, NORDUnet
Diego Lopez, RedIRIS
Gabriel Lopez, University of Murcia
Nicolas Williams, Sun Microsystems, Inc
Sascha Neinert, University of Stuttgart
Remco Poortinga – van Wijnen, Surfnet
Candido Rodriguez, RedIRIS
Paul Sangster, Symantec Corporation
Klaas Wieranga, Cisco Systems, Inc.
Simon Wilkinson, University of Edinburgh
Ian Young, SDSS

The authors are also grateful for the discussions with members of the JANET(UK) and TERENA EMC2 communities and elsewhere who have contributed to the development of the use-cases described in this document.