Final Report of the Per-Entity Metadata Working Group

(12/7/2016)

Repository ID: TI.5.1
Authors: Scott Koranda <https://orcid.org/0000-0003-4478-9026>
David Walker <https://orcid.org/0000-0003-2540-0644>
The Per-Entity Metadata Working Group
Sponsor: InCommon Technical Advisory Committee
Superseded documents: (none)
Proposed future review date: December 1, 2017
Subject tags: federation, metadata

Table of Contents

1. Executive Summary

2. Introduction

3. Current State of Metadata Distribution

4. Per-Entity Metadata Distribution

5. Risks of Per-Entity Metadata Distribution
    - 5.1. Unavailability of the MDQ Service
    - 5.2. Poor Responsiveness of the MDQ Service
    - 5.3. Network Failure or Isolation of the Metadata Consumer
    - 5.4. Security Related Risks
    - 5.5. Unavailability of the Metadata Production Infrastructure
    - 5.6. Cost
    - 5.7. MDQ Client Software and Risk Mitigation

6. MDQ Service Architecture
    - 6.1. Existing Infrastructure
        - 6.1.1. Producing Local Metadata
        - 6.1.2. Importing Global Metadata
    - 6.2. Adding Per-Entity Metadata to the Infrastructure
        - 6.2.1. Content Delivery Network Based Distribution
        - 6.2.2. Traditional Server-Based Distribution

7. Requirements for the InCommon MDQ Service
    - 7.1. Security
    - 7.2. Availability
    - 7.3. Responsiveness
    - 7.4. Metadata Production
    - 7.5. Monitoring

8. Other Issues
    - 8.1. Discovery
    - 8.2. Deployment Profiles
    - 8.3. Local Site Caching
    - 8.4. Open Access

9. InCommon Per-Entity Distribution Roadmap
    - 9.1. Short Term (1-3 months)
    - 9.2. Medium Term (2-12 months)
    - 9.3. Long Term (12-24 months)

9.4. Longer Term (24+ months)

10. Appendix: State of MDQ support in IdP and SP software

11. Appendix: Per-Entity Metadata Working Group Charter

12. Appendix: Working Group Participants

1. Executive Summary

In its 10+ years, the InCommon federation has grown from serving a very limited number of applications ("Service Providers" or "SPs") with an equally small number of participants ("Identity Providers" or "IdPs" - mostly large research universities), to a federation that now supports thousands of different SPs across approximately 450 IdPs. This growth and InCommon's recent production support for the eduGAIN interfederation service have caused a rapid increase in the size of InCommon's metadata aggregate, a large file containing information about all interfederated SPs and IdPs. The growth of the federation and the metadata aggregate puts the federation at risk of becoming a victim of its own success. The aggregate model for metadata distribution parallels how services used a "hosts" file for hostname resolution before DNS existed, and like the hosts file the single large metadata aggregate file has reached the end of its sustainability.

Per-entity metadata distribution addresses scalability and sustainability by enabling metadata consumers (SPs and IdPs) to obtain just the metadata they need, when they need it, rather than consuming the full aggregate. Whenever a consumer requires the metadata for another entity, it uses the Metadata Query (MDQ) Protocol to query an MDQ service and retrieve the metadata for only that single entity.

In order to sustain InCommon's capacity for growth, the Per-Entity Metadata Working Group recommends the following.

- InCommon must deploy an MDQ service.
- Availability of the MDQ service should be at least 99.99%. It should be engineered so that 99% of all queries are satisfied within 200ms, exclusive of network latency.
- Utilization of InCommon's MDQ service will require reconfiguration of participants' Identity Providers (IdPs) and Service Providers (SPs). InCommon should provide communication and education to facilitate with that work.
- SAML implementations other than Shibboleth and SimpleSAMLphp (*e.g.*, Microsoft ADFS, Ping Identity, Ellucian/WSO2) will likely require community pressure to support federation-distributed metadata and per-entity distribution.
- The per-entity metadata support in Shibboleth and SimpleSAMLphp should be enhanced to mitigate the effect of network and service outages and slowdowns affecting InCommon's MDQ service. InCommon should advocate resources and community support for those efforts.
- As a short-term measure, InCommon should produce a metadata aggregate targeted at Service Providers that contains only the metadata for Identity Providers in order to temporarily address operational issues for Service Providers caused by the current size of the full metadata aggregate.
- Support for IdP discovery in the absence of an aggregate was explicitly not part of the charge for this working group. It is, nonetheless, critical to address before SPs providing

discovery services for their users can no longer handle the growing aggregate. Another working group should be formed quickly to address discovery.

2. Introduction

This is the final report of InCommon's Per-Entity Metadata Working Group, which was charged by the InCommon Technical Advisory Committee with the following tasks:

1. Develop a roadmap for addressing the immediate needs for reduced aggregate size, as well as intermediate milestones along a trajectory to a sustainable future state, based on the MDQ protocol for per-entity distribution of federation metadata.
2. Address issues related to reliance on this new model, including but not limited to:
    a. High availability
    b. Performance
    c. Site redundancy
3. Develop requirements, risks, and recommended risk mitigation strategies for a production per-entity metadata service delivered by InCommon, including a firm definition of the scope of the service, aligned with the immediate needs addressed in the roadmap.
4. Advise InCommon staff on implementation of a solution, based on the requirements of the service.
5. Compile the outcomes of these investigations into a report to the TAC.

In its 10+ years, the InCommon federation has grown from serving a very limited number of applications with an equally small number of participants (mostly large research universities), to a federation that now supports thousands of different applications across approximately 650 active participants. This and InCommon's recent production support for the eduGAIN interfederation service have caused rapid growth in the size of InCommon's metadata aggregate, putting the federation at risk of becoming a victim of its own success.

Metadata aggregates, that is, metadata made up of more than one SAML entity descriptor element, are static lists of entity descriptors that are aggregated, validated, signed and distributed to consumers of federation metadata in a single, large file. This model is analogous to how hostname resolution was done before DNS existed, using a "hosts" file, and it has reached the end of its sustainability the same way the hosts file did long ago.

The metadata aggregate distribution strategy has a number of major drawbacks:

1. An error in a single entity descriptor can cause denial-of-service for consumers of the aggregate when malformed entity descriptors are created erroneously or imported from other federations.

2. Significant amounts of memory are needed to process the aggregate - now on the order of gigabytes. This will only increase over time, is a waste of deployer resources, and precludes resource-constrained deployments from full federation participation.
3. Increased bandwidth is utilized by the Federation Operator to distribute a large file that consumers almost certainly don't need in its entirety.
4. Every IdP and SP requires increased time and bandwidth to obtain and process the aggregate, thus increasing the time to start up a SAML deployment. At the aggregate's current size, this has already become a critical issue for some deployments.

3. Current State of Metadata Distribution

Since its inception InCommon has realized the federation trust fabric as a monolithic digitally signed SAML metadata file, aggregating the metadata for all IdPs and SPs (entities). Today the InCommon Federation operator generates three large files colloquially known as the preview, main, and fallback aggregates. All deployments are encouraged to retrieve an InCommon metadata aggregate on at least a daily basis, and to verify the authenticity of the aggregate by checking the digital signature of the file using the well-known InCommon metadata signing certificate. Metadata consumers download an aggregate file from an HTTP server operated directly by InCommon and housed in an Internet2 data center. A second HTTP server operated in a geographically distant location acts as a hot standby for InCommon aggregate metadata distribution that can be put into service when necessary.

InCommon federation operators have recognized for a number of years that distributing and consuming large monolithic aggregates containing all entities would not scale as the number of entities increased over time. In early 2016, federation operators began importing metadata from international federations as part of InCommon's participation in eduGAIN. At that time, the InCommon metadata aggregate file grew dramatically in size. It continues to grow steadily as federations export more entities to eduGAIN and more federations participate in eduGAIN. With the growth of the InCommon aggregate, metadata consumers operated by InCommon Participants have reached a tipping point, with resource-constrained deployments experiencing slow startup times and even crashes due to the size of the aggregate.

The vast majority of InCommon metadata consumers do not operationally need to consume and have available the SAML metadata for every entity since most participate in transactions with only a handful of relying parties. Even entities that do actively transact with many unique relying parties do so relatively infrequently. Further, the nature of SAML federation is such that no IdP needs to consume metadata about any other IdP and no SP needs to consume metadata about any other SP. These observations, taken as a whole, indicate not only that aggregate metadata distribution has reached the end of its useful life, but also that the needs of the vast majority of federation participants can be met using a new model — Per-Entity Metadata Distribution.

4. Per-Entity Metadata Distribution

Per-entity metadata distribution addresses the problems of large, monolithic aggregates and exploits the relatively modest metadata consumption needs of most consumers by enabling them to obtain just the metadata they need, when they need it, rather than consuming the full aggregate. Whenever a consumer requires the metadata for another entity, it uses the Metadata Query (MDQ) Protocol to query an MDQ service and retrieve the metadata for only that single entity.

The benefits of per-entity metadata distribution via the MDQ protocol include:

- Reduced memory and resource consumption by a metadata consumer since it need only request and consume metadata for entities with which it needs to federate.
- Reduced load on the metadata distribution service since consumers only query for and download the actual entity descriptors they need.
- Decoupling of entity descriptors so that errors for any single entity need not impede the distribution and consumption of the metadata for all other entities.

5. Risks of Per-Entity Metadata Distribution

While a transition to per-entity distribution of the InCommon metadata would help reduce resource consumption, network traffic, and resolve the brittleness of large aggregates, it is not without risk. Below, we examine categories of risk and discuss changes in the risk posture for InCommon Participants and the Federation as a whole as part of a transition away from monolithic aggregates to per-entity metadata distribution. These risks are presented in no particular order.

5.1. Unavailability of the MDQ Service

When an MDQ consumer, either IdP or SP, requires the metadata for an entity, it must query an MDQ service to obtain the metadata. If the MDQ service is unavailable and cannot answer the query, the consumer does not receive the necessary metadata for the entity, resulting in a service disruption for the MDQ consumer.

An MDQ service may be unavailable for any number of reasons, some inherent and some not:

- Server failures including power failures and resource exhaustion, be it memory or disk.
- Failed or misconfigured software components such as HTTP web servers.
- Incorrect or incomplete DNS entries that prevent resolution of the IP address(es) for the service.

- Network outages for any network component without redundancy between the MDQ consumer and the service.
- Outages caused by poor consumer software implementations.
- DDoS and other attacks on the service by malicious actors.

Each of these reasons for service disruption exist today in InCommon's current metadata distribution strategy. However, since most IdPs and SPs obtain the metadata periodically, recover gracefully from download errors, and the signed aggregate is valid for two weeks, such disruptions are transparent and do not commonly result in user-visible failures.

The change in risk posture for InCommon Participants when transitioning to per-entity metadata distribution is that service disruptions in the MDQ service are more likely to result in visible failures. Caching MDQ query results by consumers does not entirely mitigate this risk, but does provide moderate outage tolerance assuming prior successful queries. While other mitigations, including preloading caches and failover to tiered services are discussed below, it is immediately apparent that the transition to per-entity metadata distribution substantially increases the high availability (HA) and related requirements for InCommon metadata distribution services.

5.2. Poor Responsiveness of the MDQ Service

An MDQ service may be available but may not respond quickly enough to any individual MDQ query. Specifically, the service may take a relatively long time to return a valid and expected response to the client. Since the IdP or SP making the query cannot continue the SSO web flow until it receives the correct response from the MDQ service the user that initiated the SSO flow will see a delayed and degraded SSO experience.

An MDQ service may respond slowly for any number of reasons including:

- A lack of service or server capacity so that the service is unable to respond in a timely way.
- Degraded performance of other services on which an MDQ service may depend.
- Degraded network performance for any part of the network between the consumer and the service.
- DDoS and other attacks on the service by malicious actors.

Again, each of these reasons for service degradation exist today in InCommon's metadata aggregate distribution strategy, but since most IdPs and SPs download the metadata aggregate asynchronously to specific user behavior, and the signed aggregate is valid for two weeks, such degradations are transparent and do not result in a degradation of the sign-on experience for users.

The change in risk posture for InCommon Participants when transitioning to per-entity metadata distribution is that a slow and unresponsive MDQ service will result in a degraded user

experience. Mitigations for this risk are discussed below, but again it is obvious that the transition to per-entity metadata distribution substantially increases the requirements on the InCommon metadata distribution service for responsiveness, including capacity and the ability to meet peak demand during particular times in the academic calendar such as late August and early September.

5.3. Network Failure or Isolation of the Metadata Consumer

Above, we considered network failures and performance factors that can contribute to MDQ service degradation, but here we consider specifically the risk inherent in per-entity distribution when the network connecting a campus or other organization to the Internet fails.

There is no change in risk posture for off campus users since they cannot contact either the campus IdP nor any SP on campus and this is the same regardless of whether the campus is relying on the InCommon monolithic metadata aggregate or an InCommon MDQ service.

On campus users can contact the campus IdP[1] but not off campus SPs. Again the risk posture is no different whether relying on an aggregate or an MDQ service, although the error a user experiences may manifest differently depending on where in the SSO flow the lost connectivity to the Internet is first encountered.

On campus users can, however, contact both the campus IdP and campus SPs during such a network event and it is expected that the basic intra-campus services continue to operate normally. If a campus relies on its own mechanisms and not InCommon-signed metadata for bootstrapping the trust between the campus IdP and SPs then again there is no change in risk posture since there is no reliance on the InCommon trust fabric for interoperability between the campus IdP and campus SPs.

Some campuses or organizations do, however, rely on InCommon metadata for bootstrapping the trust between the campus IdP and campus SPs. That is, they submit metadata for both the IdP and SPs into InCommon metadata and configure both the IdP and SPs to consume the InCommon metadata. For these campuses the change to per-entity metadata distribution does result in a change in risk posture since the inability to query an MDQ service external to campus during a network isolation event will result in intra-campus SSO flows failing. Those campuses may require mitigation strategies before adopting per-entity metadata distribution.

5.4. Security Related Risks

A full and detailed analysis of the security risks associated with the per-entity distribution of InCommon metadata is out of scope here. Rather, we consider one specific change in the security-related risk posture.

[1] We do not consider here the details of a campus operating its IdP off site, perhaps in the cloud.

InCommon digitally signs each of the metadata aggregates so that consumers can verify the integrity of the download using the well-known InCommon metadata signing certificate. Weaknesses in XML digital signature implementations have been found in the past, however, and will plausibly be discovered in the future. Malicious actors could exploit such weaknesses and prepare a rogue file to distribute to their target. A successful attack requires the target to consume the rogue file.

Another possible attack is the substitution of old metadata that is no longer current, but still within its validity period, for the current metadata. That an MDQ query occurs in-band and just-in-time may allow an attacker to induce a query to the MDQ service on demand and so more readily attempt to intercept the query from the client and inject the old metadata.

Use of TLS transport can mitigate this risk, assuming the MDQ client verifies the MDQ server's certificate as being authentic and either self- issued or issued by a trusted certificate authority. There are, however, issues of how well that certificate's private key can be protected in a content delivery network that the working group did not explore extensively. In particular, the metadata signing certificate should not be used as the end-entity server certificate for TLS.

5.5. Unavailability of the Metadata Production Infrastructure

It is assumed that a common technical and process infrastructure will be used to produce metadata for both per-entity and aggregate distribution. If the servers, equipment, or people involved in the daily production of metadata are not available, additions, updates, and deletions of metadata will not occur. Continued distribution of the current set of metadata will be uninterrupted, however.

The impact of missing a production cycle is the same for per-entity distribution as it has always been for aggregate production. Under normal circumstances, the impact of not producing metadata on a particular day is low. There have been, however, emergency situations where it has been necessary to produce more than once in a single day. Loss of the metadata production infrastructure on such a day could have severe implications.

5.6. Cost

The infrastructure to support per-entity metadata distribution will require reconfiguration and upgrades over time to address an increasing, sometimes unpredicted, workload.

Early experience with MDQ service deployments including both the InCommon MDQ pilot project and the initial rollout of an MDQ service for the UK Access Management Federation indicates that this will not require very large expenditures, even if the future InCommon MDQ service leverages commercial content delivery networks (CDNs). Still, it behooves InCommon to track workload and cost over time to ensure that sufficient resources are available when needed.

5.7. MDQ Client Software and Risk Mitigation

Much of the risk detailed above for a per-entity metadata distribution architecture for the InCommon Federation can be mitigated by carefully designing, deploying, and operating the MDQ service. An MDQ service that is always available and responsive goes a long way to addressing much of the risk of transitioning to a per-entity approach.

Service outages do happen, however, and even well designed deployments can face uncertain technical challenges as the InCommon and wider communities continue to grow and add new entities to the metadata. It follows that MDQ consumers have a role in reducing and mitigating the risks of per-entity metadata distribution. That role then directly translates into technical requirements for the MDQ consumer software stacks operated by InCommon Participants.

Specifically, mitigating the risks of MDQ service unavailability, poor responsiveness, and high latency can only be achieved if MDQ clients (SPs and IdPs) have the capability to detect and then appropriately respond to those service conditions. Such capabilities should include:

- A persistent caching mechanism that retains previously-retrieved metadata across software restart so that it may be re-used if the software is restarted when the MDQ service is not available. A likely mechanism is caching to local disk and then consumption from the cache on restart.
- A mechanism for pre-loading metadata for high-value IdPs and SPs and keeping it available. This enables successful operation the first time a high-value entity's metadata is needed, even if the MDQ service is not available.
- The ability to detect a failed query, retry appropriately, and after repeated completed but failed queries failover to a secondary MDQ service. A complete implementation would include the ability to mark an MDQ service as unavailable for some time but later test again and return to using it when the service is again available and completing successfully.
- Likewise the ability to detect unresponsive (hanging) MDQ services or MDQ services that do not answer queries fast enough and similarly retry, mark as unavailable, and then later test for restoring into service such MDQ services.

Clients implementing the capabilities above should allow administrators to tune thresholds for detecting and responding to failures to accommodate local deployment needs.

The Shibboleth development team has added significant capabilities in version 3.3 of the Shibboleth Identity Provider. Clients without such capabilities can still leverage a per-entity metadata distribution infrastructure and interoperate with MDQ services but they risk lower availability for their users. As the predominant client software used in InCommon, the working group recommends that the InCommon community request that the SimpleSAMLphp

development team add these capabilities. Specific guidance is detailed below and included in the discussion of timelines.

6. MDQ Service Architecture

As noted above, the design and implementation of the MDQ service itself will have the greatest impact on mitigating the risks of per-entity metadata distribution. Before discussing risk mitigations and translating them into specific requirements on the InCommon MDQ service, however, it is helpful to examine proposed MDQ service architectures to help put them into context.

While all risks described above should be addressed, the singular requirement for a per-entity metadata distribution architecture for the InCommon Federation is that users do not observe any change in the behavior of their InCommon-backed (SAML) authentication flows during and after the transition from aggregate-based distribution to per-entity distribution. This requirement distills down to each relying party, both IdPs and SPs, the requirement that every time a client queries an MDQ service for metadata for a particular entity, the query is answered (high availability) and answered quickly (high responsiveness or low latency) with very high probability. Put simply, any MDQ service operated by InCommon must "just work" in the same way that DNS services "just work".

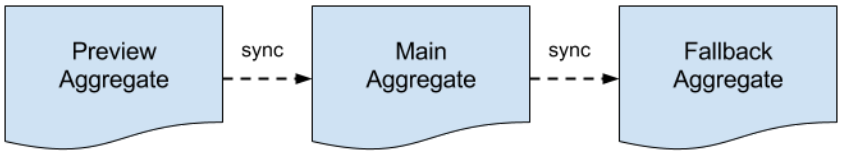
The architecture for the MDQ service will share much of the infrastructure to produce metadata that already exists to create the aggregates. Where it differs is in the addition of processing steps to sign each entity's metadata, as well as a highly available and responsive distribution layer.

6.1. Existing Infrastructure

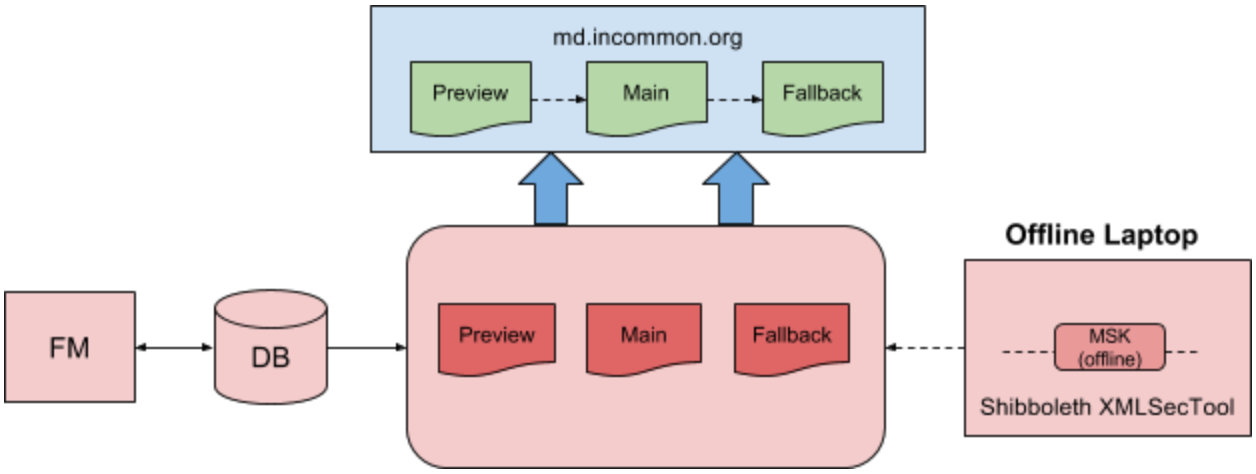
6.1.1. Producing Local Metadata

InCommon metadata is processed daily. Metadata submitted by InCommon Site Administrators via the Federation Manager (FM) is vetted and approved by the InCommon Registration Authority at approximately 2:30 pm ET every Internet2 business day. Fresh aggregates are signed and published immediately thereafter, at approximately 3:00 pm ET. See the InCommon Hours of Operation page for more detail.

InCommon distributes multiple metadata aggregates for various purposes. Structural changes to metadata (which often involve extension schema) are systematically pushed through a pipeline of aggregates to avoid breakage. Clients consume whatever aggregate is most appropriate for their particular deployment. Of special interest is the fallback aggregate, a temporary alternative for deployments that experience metadata issues as changes are pushed through the pipeline.



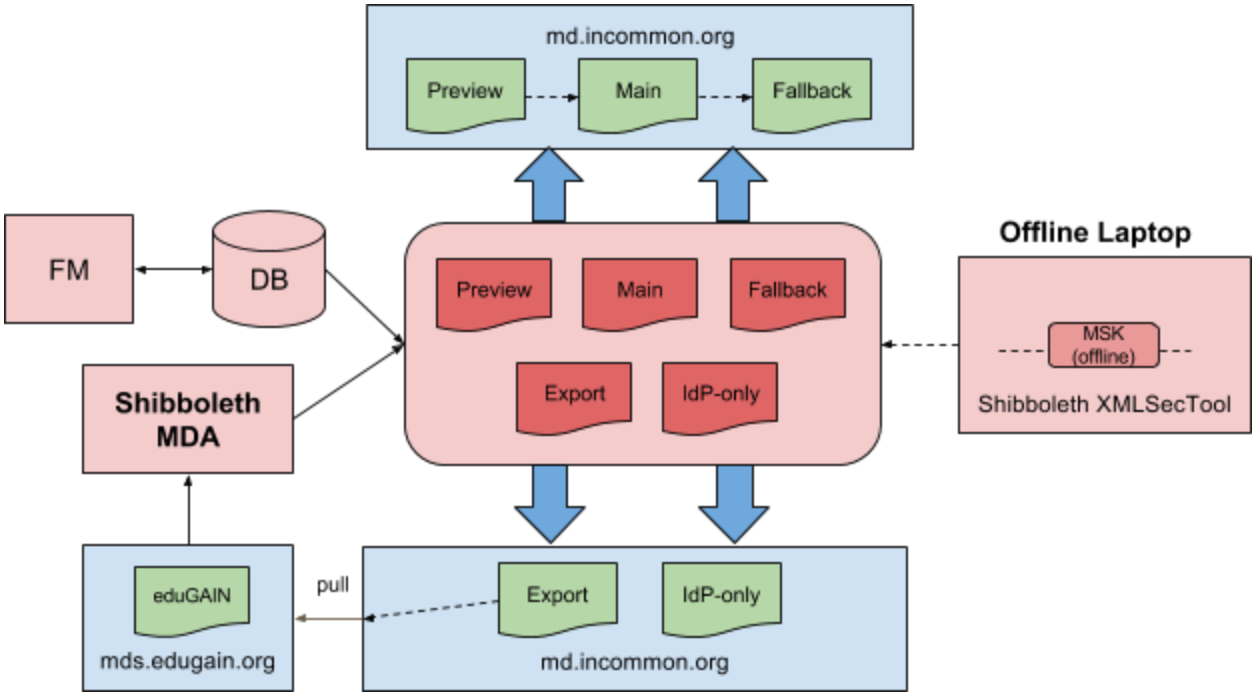
Each aggregate is digitally signed for authenticity and integrity. The daily metadata signing process is basically a manual process. The metadata signing key resides on an offline laptop stored in a safe with strict access controls.



6.1.2. Importing Global Metadata

By the end of Q1 2016, InCommon was fully integrated with the eduGAIN metadata aggregation service. During that time, InCommon began importing global metadata directly into the InCommon metadata aggregate. Likewise InCommon exported local metadata to eduGAIN.

As a consequence of the eduGAIN integration, there are now two distinct sources of metadata: 1) local metadata registered by InCommon, and 2) global metadata registered by other federations. The daily metadata signing process combines entity descriptors from both sources into a single, comprehensive metadata aggregate. The diagram below illustrates the expanded infrastructure that incorporates eduGAIN metadata (compare with the diagram shown in the previous section).



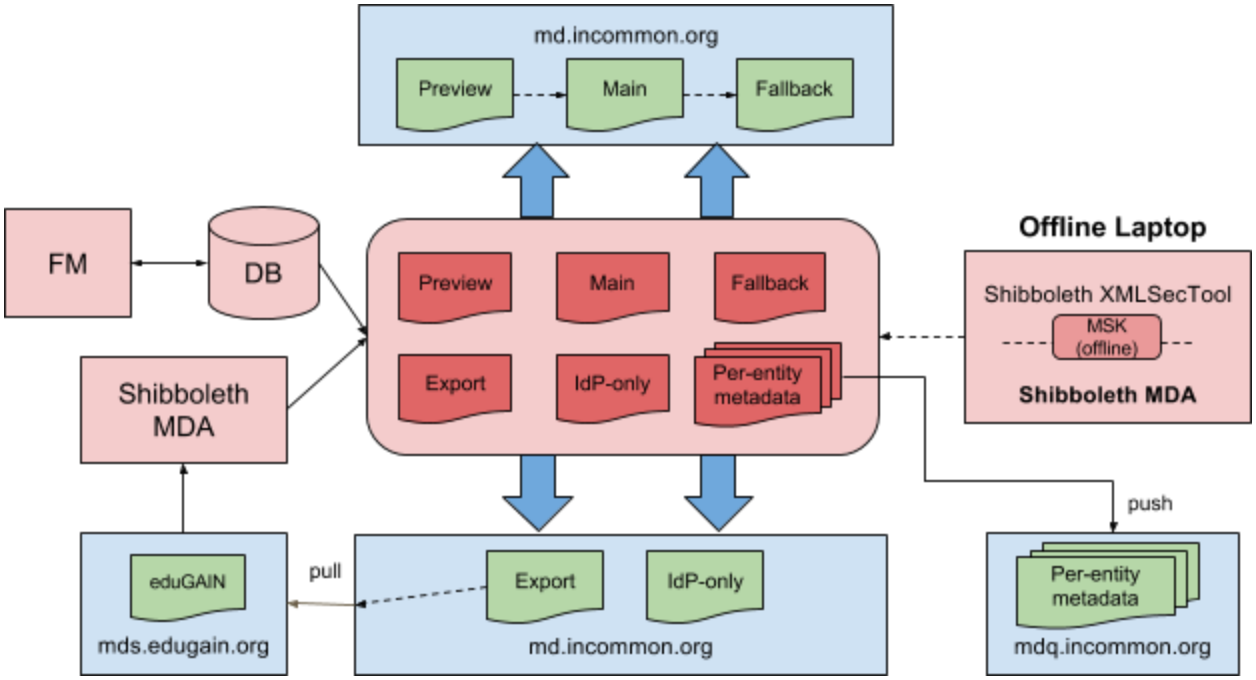
In preparation for the daily metadata signing process described in the previous section, a repetitive metadata import process ensures that fresh global metadata is available for aggregation and signing at 3:00 pm ET. The import process is implemented using a customized instance of the Shibboleth Metadata Aggregator software, which filters imported metadata according to published technical policy rules.

To complete the circular flow of metadata among federations, a subset of entity metadata registered by InCommon is assembled into an export aggregate and made available for download by eduGAIN operations. As a matter of policy, IdP metadata registered by InCommon is exported to eduGAIN by default whereas InCommon SP owners explicitly opt into metadata export.

The initial eduGAIN integration caused the InCommon metadata aggregate to nearly double in size. To compensate for the ever-increasing size of the metadata file, and because some SP deployments will be unable to leverage per-entity metadata (at least initially), an IdP-only aggregate for SP deployments was introduced in October 2016. IdP deployments, on the other hand, are expected to leverage per-entity metadata as soon as it becomes available.

6.2. Adding Per-Entity Metadata to the Infrastructure

In order to provide per-entity metadata distribution, two things must be added to the infrastructure, processing to sign each entity's metadata, and a highly available and responsive distribution layer. This is illustrated below:



The new processing is shown as "Per-entity metadata" in the center bubble, and the new distribution layer is the "mdq.incommon.org" box in the lower right.

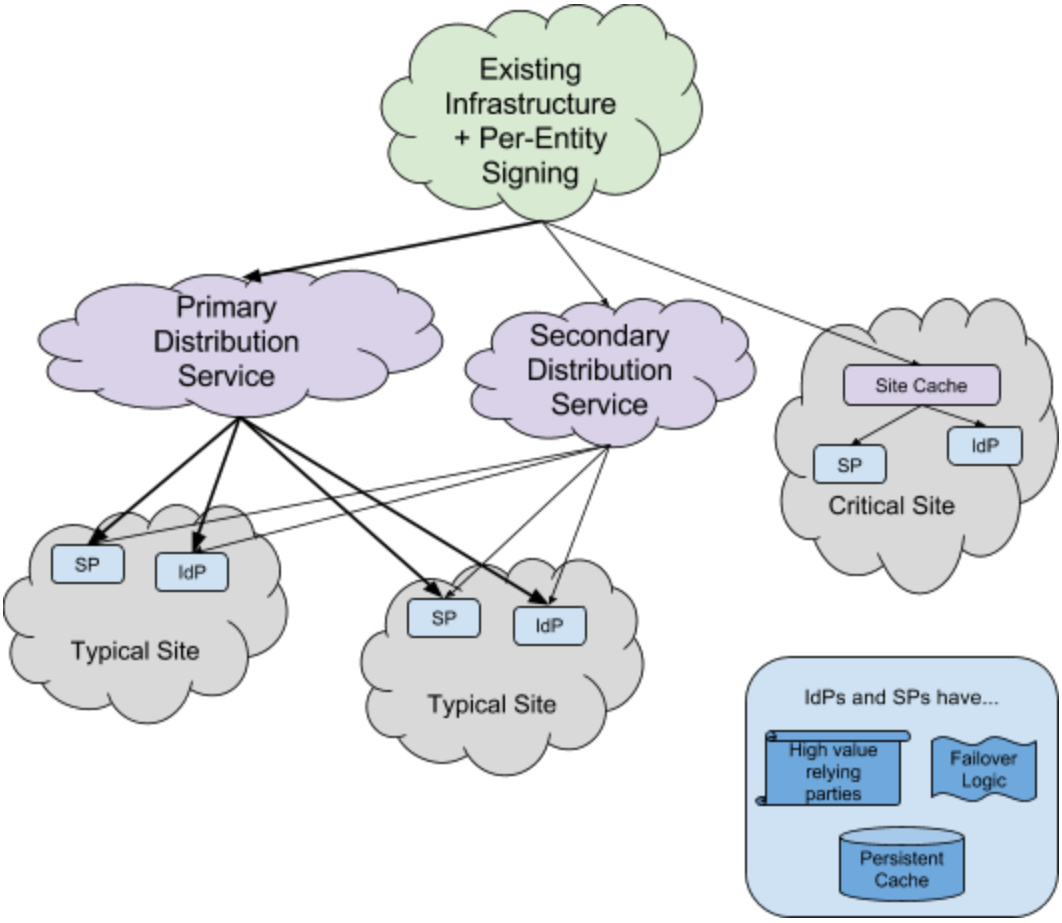
The working group's consensus was that the MDQ service must provide very high availability to the federation's IdPs and SPs, at least 99.99%. Unfortunately, popular content distribution services, such as commercial CDNs, typically guarantee only 99.9% availability. This is the equivalent of about 43 minutes of downtime per month.

In order to achieve at least 99.99% availability (4.3 minutes of downtime per month), the group recommends that both primary and second distribution services be deployed to "ride out" outages in the primary service. The existing SAMLbits CDN, likely augmented with additional nodes contributed by InCommon, could be a good candidate for the secondary service.

Note that the combination of a highly available primary distribution service, a secondary distribution service, and MDQ clients with appropriate configuration for failover and sophisticated persistent caching, tunable for high-value relying parties, will provide a solution for the large majority of InCommon Participants.

Finally, it is not expected that most deployers will leverage a local caching service or out of band cache filling processes, but the opportunity exists for extremely critical sites, or sites with problematic Internet connectivity to implement a local metadata cache.

The following diagram illustrates this distribution architecture.



In order to meet the high availability and low latency requirements for distribution, the working group has discussed two strategies for the primary and secondary distribution services:

1. Content delivery network based distribution
2. Traditional server based distribution

Either strategy or both could be selected for the deployed solution architecture.

6.2.1. Content Delivery Network Based Distribution

A content delivery network (CDN) is a distributed network of proxy servers deployed in multiple data centers that serve content to consumers with high availability and high performance. CDNs serve a large portion of web content today, including many of the standard JavaScript libraries, Cascading Style Sheets (CSS), and images and other media files. Besides better performance and availability, CDNs also offload the traffic served directly from a content provider's origin infrastructure and can provide a degree of protection from DoS attacks by using their large distributed server infrastructure to absorb the attack traffic.

Considerations for the use of CDNs for distribution include:

- Pros
    - CDNs already mitigate common risks to cloud-based services, such as DDoS.
    - Capacity scaling is automatic.
    - The cost will likely be lower, although this will require more detailed analysis.
- Cons
    - Most CDNs are optimized for use by browsers which can, for example, switch quickly among multiple IP addresses for a server. Clients of services like MDQ may or may not be that agile.
    - CDNs must have access to the private key used for TLS. While multiple mitigation strategies are typically provided, they are only partial. The security risks must be analyzed.
    - MDQ is new to InCommon; so is the use of CDNs. It may be prudent to introduce only one new technology at a time.

6.2.2. Traditional Server-Based Distribution

The alternative to use of CDNs is to deploy a more traditional server-based distribution layer, operated by Internet2. Such an infrastructure could be based either in the cloud or in geographically-distributed Internet2 data centers.

Considerations for this alternative include:

- Pros
    - The service would be optimized to the needs of MDQ client software, rather than browsers.
    - Access to the private key used for TLS can be managed directly by Internet2.
    - The architecture is more familiar. Fewer new technologies would be introduced simultaneously.
- Cons
    - The service will need to address other risks, such as DDoS attacks, that are inherently addressed by CDNs.
    - Capacity scaling will likely not be as easy/automatic.
    - The cost will likely be higher, although this will require more detailed analysis.

7. Requirements for the InCommon MDQ Service

InCommon's MDQ service must be designed and deployed to address the risks and other issues described in this document. Specific requirements are detailed below.

7.1. Security

The security of the MDQ service must, as much as possible, be equivalent to existing aggregate service. As noted above, the nature of MDQ may increase the risk of a metadata consumer receiving out of date information that is still within its validity period, due to a man-in-the-middle attack between the distribution layer and the consumer; this must be mitigated through the use of TLS unless some other mitigation is determined to be more appropriate.

7.2. Availability

Any MDQ client operated by an InCommon Participant must find the service available. Any time a client queries the service and the service does not respond in compliance with the MDQ specification due to any issue with the service delivery infrastructure is known as an outage. The time period during which the service suffers no outages is known as the service uptime. The monthly service uptime percentage is the percentage of client service transactions in which the service responds to client queries and delivers the requested metadata without error. The InCommon MDQ service must realize a monthly service uptime percentage of at least 99.99%.

This figure does not include or address outages which occur due to failures of the network or infrastructure at sites running per-entity metadata clients. These failures are not addressable by federation operations, and are explicitly out of scope for service availability targets.

7.3. Responsiveness

The distribution layer of the InCommon MDQ service must provide a response time of no more than 200ms[2] for at least 99% of all queries received each month from a test probe on or near the Internet2 backbone. The test probe will select one entity per minute, retrieve its metadata, and record the response time. InCommon will post monthly reports of response time distribution on the web.

It is understood that not all InCommon participants will experience the response times observed by the test probe, depending on each participant's server and network topology with respect to the Internet2 backbone. It is also understood that it is the participants' observation of response time that is truly important, if difficult or impossible to generalize into this service specification. For this reason, response time measurements should also be taken from participants' sites. We recommend that InCommon identify representative participants and ask them to contribute records of their response times for inclusion with the monthly reports. We also recommend that TIER include metadata query response times in its current efforts to instrument Shibboleth software.

[2]200ms is a fraction of the overall response time during an SSO flow that is not expected to significantly change the user experience.

7.4. Metadata Production

The working group does not recommend any changes to the processes and systems that produce the metadata that is distributed both as aggregates and via the MDQ service. In particular, we do not see a need to change the current once-per-day metadata production or the two-week metadata validity period. If at some time, however, InCommon were to decide to leverage per-entity distribution to institute more frequent, or real-time, publishing of metadata updates, then the availability of the metadata production infrastructure should be addressed in light of such new service commitments. A decision to publish metadata more frequently would require reconsideration of the metadata validity and caching intervals. Shortening the validity interval decreases the risk old metadata being reintroduced by a malicious actor, but it also increases the risk that current metadata will expire during an outage of the metadata production infrastructure. Shortening the caching interval shortens the time required for metadata updates to propagate to IdPs and SPs, but it also increases load on the distribution infrastructure. Achieving a proper balance for both parameters will be essential if more frequent publishing is desired at some time in the future.

7.5. Monitoring

In addition to the monitoring for responsiveness described in section 7.3, InCommon Operations should monitor the availability and performance of the Per-Entity Metadata Service from multiple geographic locations on a regular basis to demonstrate compliance with the service requirements in this section. These results should be made transparent and accessible to all federation participants , in a manner to be determined by Federation Operations.

8. Other Issues

8.1. Discovery

In order to authenticate a user and retrieve attribute information about them, an SP must redirect the user to their correct IdP. This process of determining which IdP to use is called Discovery. Discovery take many forms, but many of them rely upon the SP having a list of all IdPs in metadata so that its users can select from any of the available IdPs. With per-entity distribution, said list is not readily available; SPs that rely upon it thus cannot migrate entirely to per-entity metadata in its current form.

Discovery was explicitly not part of this working group's charge, but a solution is critical for any SP that provides a discovery service for its users. InCommon should convene a working group to address this issue quickly to assure that affected SPs have a path forward before the growing metadata aggregate impacts them.

8.2. Deployment Profiles

The MDQ protocol adds new criteria that must be addressed in the deployment of SAML services. InCommon should work with its own Deployment Profile Working Group to assure consideration of MDQ in their specification. At a minimum, a requirement to support MDQ and per-entity metadata distribution should be specified.

8.3. Local Site Caching

Some sites will have higher requirements for availability and/or responsiveness than is offered by the primary and secondary CDNs. Likely reasons for this are insufficient Internet connectivity for the site or heavy reliance on metadata for SAML-based authentication internal to the site. Such sites can create local caching servers, or even a private CDN, to address this. They can also deploy site-specific out of band processes to pre-fill caches for local services and high-value partners. While InCommon would not support such local infrastructure, it should provide documentation to enable a site to support its own.

InCommon should track and document lessons it learns about the effectiveness of caching and other parameters to improve availability and performance. The TAC should consider creation of a working group to work with InCommon Operations on these issues.

8.4. Open Access

The software, documentation *etc*. for the MDQ service must be openly available to others wishing to follow in InCommon's footsteps. This applies to the MDQ software itself, as well as tools for managing and monitoring the service.

9. InCommon Per-Entity Distribution Roadmap

9.1. Short Term (1-3 months)

There is concern that many SPs will soon be impacted by the current, growing metadata aggregate. As a short-term workaround, InCommon will deploy an IdP-only aggregate file on a daily basis.

9.2. Medium Term (2-12 months)

Upon acceptance of this report, InCommon develops a detailed plan and allocate resources for the deployment of per-entity metadata services. This plan will include the following elements:

- A detailed architecture that addresses the risks and service commitments described in this document
- Formal requests to the Shibboleth and SimpleSAMLphp development communities to address the capabilities to weather short service outages described in MDQ Client Configuration and Caching Capabilities
- Deployment of the detailed architecture
- A communications plan, including
    - What participants must do
    - What participants may choose to do
    - What participants cannot do (*e.g.*, discovery) with workarounds, if possible
    - Timeline of events (conditioned on the timing of Shibboleth and SimpleSAMLphp development)
    - Participant feedback

9.3. Long Term (12-24 months)

Per-entity metadata distribution is in production during this period. A solution for discovery that is analogous to existing discovery approaches is identified and deployed. Operational issues are discovered and resolved. IdPs and SPs that consume metadata, but have not adopted per-entity metadata distribution, are increasingly experiencing problems due to the size of the aggregate.

9.4. Longer Term (24+ months)

The vast majority of IdPs and SPs that consume metadata have migrated to per-entity metadata distribution in the 24-36 month time frame, even those that were delayed due to discovery issues. More work is done to improve discovery strategies. InCommon develops a plan for the future of aggregate distribution in the 36-48 month time frame, depending on how vast that majority is.

10. Appendix: State of MDQ support in IdP and SP software

The working group reached out to software providers to determine their support for MDQ. At this time, Shibboleth and SimpleSAMLphp are the only two that have released or are working on MDQ support; they are also the two implementations that support federation-provided metadata via any distribution method.

We remind sites that operate SAML software stacks other than Shibboleth or SimpleSAMLphp that only those projects have historically and consistently supported functionality highly desired for the best interoperability in the higher education and research federations.

This table captures current and future state of client software capable of requesting and consuming per-entity metadata via the Metadata Query Protocol.

Client Software	Supports MDQ protocol?	Notes on current capability	Security Model(s)	Known future capabilities or enhancements?
Shibboleth SP (current: V2.6.0)	Yes	See the Dynamic MetadataProvider topic in the Shibboleth wiki. This feature (first introduced in SP V2.0) is mostly untested (which means there are probably bugs) but is already being enhanced in response to this group's feedback.	XML Signature, TLS validation against explicit anchors	New "file://" feature in SP V2.6.0 Committed to add additional caching support in 2017.
Shibboleth IdP (current: V3.3)	Yes	See the DynamicHTTPMetadataProvider topic in the Shibboleth wiki. This feature (new in IdP V3.3) is is probably the most capable client implementation available but has seen little use to date.	XML Signature, TLS validation against explicit anchors	V3.3 has introduced substantial caching enhancements in line with the group's suggestions.

SimpleSAMLphp (current: V1.14.8)	Yes	MDQ metadata handler merged on March 16, 2015. There is no formal documentation (search for "MDQ" in config.php). This feature is mostly untested.	XML Signature (via cert fingerprint)	
ADFS 2.0 (Server 2008 and Server 2008 R2)	Partial	ADFS will fetch and cache a single SAML EntityDescriptor at a configured endpoint location beginning with "https://"	TLS	
ADFS 3.0 (Server 2012 R2)	Partial	ADFS will fetch and cache a single SAML EntityDescriptor at a configured endpoint location beginning with "https://"	TLS	
ADFS 4.0 (Server 2016 Tech Preview)	Partial	ADFS will fetch and cache a single SAML EntityDescriptor at a configured endpoint location beginning with "https://"	TLS	This version may load an aggregate
Ping	No	Ticket filed for next release to enable the needed 'Accepts' header value.	TLS	

11. Appendix: Per-Entity Metadata Working Group Charter

Problem Statement

In its 10+ years, the InCommon federation has grown from serving a very limited number of applications with an equally small number of participants (mostly large research universities), to a federation that now supports thousands of different applications across approximately 650 active participants. Add to this a rapid growth in the size of InCommon metadata, due to InCommon's production support for the eduGAIN interfederation service, and the federation is at risk of becoming a victim of its own success.

Metadata aggregates, that is, metadata made up of more than one SAML entity descriptor element, are static lists of entity descriptors that are aggregated, validated, signed and distributed to consumers of federation metadata. This model is analogous to how hostname resolution was done before DNS existed, using 'hosts' files, and it has reached the end of its sustainability the way that solution did long ago. Aggregates are inherently brittle - an error in a single entity descriptor can cause issues loading an entire aggregate.

Additionally, very large metadata aggregates, as InCommon now distributes on a daily basis, have a number of other major drawbacks:

1. Increased bandwidth use to distribute a large file that consumers almost certainly don't need in its entirety
2. Inefficient use of client bandwidth to download a large aggregate on a regular basis
3. Increased time to canonicalize (XML document normalization) a large XML document so that the signature on it may be verified - thus increased time to start up a SAML deployment consuming a large aggregate
4. Increased memory needed to canonicalize a large XML document - now on the order of gigabytes, and this will only increase over time. A waste of deployer resources.
5. Intentional or unintentional denial-of-service for consumers of an entire aggregate based on malformed entity descriptors imported from other federations.

To address these and other concerns with the aggregate, InCommon's previous Metadata Distribution Working Group[1] recommended a test deployment of the Metadata Query Protocol (MDQ)[2]. For over two years, InCommon has been running an MDQ testbed to gain experience with the technology and this new model. This new working group is charged with items necessary to allow InCommon Ops to move this technology into a production-ready service.

Stakeholders/Influencers/Influences

Different audiences can impact different aspects of this problem:

1. SAML deployers - IdP, SP, AA, Discovery Services, etc. of various implementations: Shibboleth, SimpleSAMLphp, ADFS, Ping, etc.
2. SAML implementers - Latest versions of both Shibboleth and SimpleSAMLphp support the MDQ protocol, but implementation issues may exist that have not been found due to the need for operational exercising of these features. Other implementations such as ADFS may be enabled to participate in the federation in ways they have not been able to previously.
3. InCommon Operations and Internet2 Technical Services Group (TSG) - Running a highly reliable service that responds to requests for entity descriptors in real-time is a service delivery model that is new for InCommon and will require additional resources to support.
4. Participants - what needs do they have for local per-entity metadata installations to allow for local generation and consumption of per-entity site-specific metadata? Do they have a need for a local copy of a cache of per-entity metadata for redundancy reasons? Etc….
5. International community - how will an InCommon per-entity metadata service align with plans that other federation operators may have?

Charter

The Per-Entity Metadata Working Group will:

1. Work based on the premise that InCommon will be moving toward per-entity MDQ[2] protocol-based distribution of metadata.
2. Develop a roadmap for addressing the immediate needs for reduced aggregate size, as well as intermediate milestones along a trajectory to a sustainable future state, to be determined. The first items on this roadmap should include building a production service which allows production SAML deployments to exercise their per-entity metadata capabilities, and include checkpoints to improve the service and software when issues are encountered. If short-term steps such as InCommon producing separate IdP and SP feeds are deemed necessary, these items should be included in this roadmap. This roadmap should also address the issue of continued creation (or eventual decommissioning) of multi-entity aggregates.
3. Address issues and questions that have arisen about the process of moving from where we are to relying on this new model, including but not limited to:
    a. High availability
    b. Performance
    c. Site redundancy
4. Develop requirements, risks, and recommended risk mitigation strategies for a production per-entity metadata service delivered by InCommon, including a firm definition of the scope of the service, aligned with the immediate needs addressed in the roadmap from (1).

5. Advise InCommon staff on implementation of a solution, based on the requirements of the service documented in (4).
6. Compile the outcomes of these investigations into a report to the TAC

Explicitly out-of scope is:

1. A 'full DNS' model which would require changes to the MDQ protocol or current software implementations of the protocol.
2. Other items that need to be resolved by the international community. These items should be addressed via appropriate forums such as REFEDS, or better yet, the IETF.
3. Determining a concrete roadmap for ceasing production of multi-entity aggregates. This item is important, but must be the work of a later group, after we build experience using a production-quality per-entity metadata service.
4. A solution to the IdP discovery problem in light of per-entity metadata. Discussion or debate of options is reasonable, as long as the WG's main deliverables are not sidetracked.
5. Choice of a per-entity metadata server application or specific testing of software related to a specific choice of server application. That is the realm of operationalizing a production service and is up to InCommon staff.
6. Support for querying entities by anything other than entityID (already put out-of-scope by previous work: https://spaces.internet2.edu/x/BoGDAg)

Membership

Membership in the Working Group is open to all interested parties. Solicitation will take place on lists such as the InCommon Participants list and the REFEDS list, explicitly seeking international participation. Members join the Working Group by subscribing to the mailing list, participating on the phone calls, and otherwise actively engaging in the work of the group.

Work Products

1. September, 2016 - Draft Report to the TAC, Report out at TechExchange
2. November, 2016 - Final Report to the TAC

Related Resources

1. Metadata Distribution Working Group recommendation on pilot study of MDQ
2. MDQ protocol draft
3. Draft call for participation in MDQ testbed (restricted access)

12. Appendix: Working Group Participants

- Jorj Bauer, Temple University
- Scott Cantor, Ohio State University
- Steve Carmody, Brown University
- Paul Caskey, Internet2
- Tommy Doan, Southern Methodist University
- Michael Domingues, <http://orcid.org/0000-0001-6978-2803>, University of Iowa
- Paul Engle, Rice University
- Lukas Hämmerle, SWITCH
- Walter Hoehn, University of Memphis
- Chris Hubing, <http://orcid.org/0000-0002-8565-1966>, Internet2
- John Kazmerzak, <https://orcid.org/0000-0002-6575-3340>, University of Iowa
- IJ Kim, Internet2
- Scott Koranda, <https://orcid.org/0000-0003-4478-9026>, LIGO, Chair
- Tom Mitchell, GENI
- Kevin Morooney, InCommon / Internet2
- Chris Phillips <http://orcid.org/0000-0001-5567-4916>, CANARIE
- Phil Pishioneri, Penn State
- Nick Roy, InCommon / Internet2
- Tom Scavo, InCommon / Internet2
- Rhys Smith, Jisc
- David Walker, <https://orcid.org/0000-0003-2540-0644>, InCommon/Internet2, Flywheel
- Ann West, InCommon / Internet2
- Ian Young, InCommon / Internet2