

MILE
Internet-Draft
Intended status: Standards Track
Expires: January 4, 2018

N. Cam-Winget, Ed.
S. Appala
S. Pope
Cisco Systems
July 3, 2017

Using XMPP Protocol and its Extensions for Use with IODEF
draft-ietf-mile-xmpp-grid-03

Abstract

This document describes how the Extensible Messaging and Presence Protocol (XMPP) [RFC7590] can be used as the framework as transport protocol for collecting and distributing any security telemetry information between any network connected device. As an example, this document describes how XMPP can be used to transport the Incident Object Description Exchange Format (IODEF) information.

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at <http://datatracker.ietf.org/drafts/current/>.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on January 4, 2018.

Copyright Notice

Copyright (c) 2017 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of

the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1.	Introduction	2
1.1.	Glossary of Terms	3
1.2.	Overview of XMPP-Grid	4
1.3.	Benefits of XMPP-Grid	5
2.	XMPP-Grid Architecture	6
2.1.	Using XMPP	7
2.2.	XMPP-Grid Requirements for enabling Information Sharing	8
3.	Example use of XMPP-Grid for IODEF	9
4.	IANA Considerations	11
5.	Security Considerations	11
5.1.	Trust Model	11
5.1.1.	Network	12
5.1.2.	XMPP-Grid Nodes	12
5.1.3.	XMPP-Grid Controller	12
5.1.4.	Certification Authority	12
5.2.	Threat Model	13
5.2.1.	Network Attacks	13
5.2.2.	XMPP-Grid Nodes	14
5.2.3.	XMPP-Grid Controllers	15
5.2.4.	Certification Authority	16
5.3.	Countermeasures	17
5.3.1.	Securing the XMPP-Grid Transport Protocol	17
5.3.2.	Securing XMPP-Grid Nodes	18
5.3.3.	Securing XMPP-Grid Controllers	18
5.3.4.	Limit on search result size	19
5.3.5.	Cryptographically random session-id and authentication checks for ARC	19
5.3.6.	Securing the Certification Authority	20
5.4.	Summary	20
6.	Privacy Considerations	21
7.	Acknowledgements	21
8.	References	21
8.1.	Normative References	21
8.2.	Informative References	22
	Authors' Addresses	23

1. Introduction

XMPP-Grid is intended for use as a secure transport and communications ecosystem for devices, applications and organizations to interconnect, forming an information grid for the exchange of formatted data (e.g. XML, JSON, etc). This document describes how XMPP [RFC7590] serves as the framework and protocols for securely

collecting and distributing security telemetry information between and among network platforms, endpoints, and most any network connected device.

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [RFC2119].

1.1. Glossary of Terms

Capability Provider

Providers who are capable of sharing information on XMPP-Grid.

Publisher

A capability provider sharing content information to other devices participating on XMPP-Grid.

Subscriber

A device participating in XMPP-Grid and subscribing or consuming information published by Publishers on XMPP-Grid.

Topics

Contextual information channel created on XMPP-Grid where a published message by the Publisher will be propagated by XMPP in real-time to a set of subscribed devices.

XMPP-Grid

Set of standards-based XMPP messages with extensions, intended for use as a transport and communications protocol framework between devices forming an information grid for sharing information.

XMPP-Grid Controller

Centralized component of XMPP-Grid responsible for managing all control plane operations.

XMPP-Grid Node

Platform or device that implements XMPP to connect to XMPP-Grid and share or consume security data.

1.2. Overview of XMPP-Grid

XMPP-Grid employs publish/subscribe/query operations brokered by a controller, which enforces access control in the system. This XMPP-based architecture controls what platforms can connect to the "Grid" to share ("publish") and/or consume ("subscribe" or "query") contextual information ("Topics") such as security data needed to support MILE.

Leveraging the XMPP architecture, XMPP-Grid uses the XMPP server to act as a controller, affecting the authentication and authorization of participating XMPP-Grid nodes (Node). Security information may only be published or consumed by authenticated and authorized Nodes using the XMPP publish/subscribe extension defined in [XEP-0060].

The components of XMPP-Grid are:

- o XMPP-Grid Controller (Controller): The Controller manages the control plane of XMPP-Grid operations. As such it authenticates and authorizes platforms connecting to the data exchange grid and controls whether or not they can publish, subscribe or query Topics of security data.
- o XMPP-Grid Node (Node): A Node is a platform or application that has mutually authenticated with the Controller and obtained authorization by the Controller to share and/or consume security data.
- o Data Repository: This is the source of security data available on the Grid and may be a network security platform, management console, endpoint, etc. XMPP-Grid does not mandate a specific information model, but instead remains open to transport structured or unstructured data. Data may be supplied by the security platform itself or by an external information repository.
- o Topic: An XMPP-Grid Topic defines a type of security data that a platform wants to share with other platform(s) and a specified interface by which the data can be obtained.

As enabled by the XMPP architecture, XMPP-Grid is used to exchange security context data between systems on a 1-to-1, 1-to-many, or many-to-many basis. Security data shared between these systems may use pre-negotiated non-standard/native data formats or may utilize an optional common information repository with a standardized data format, such as IODEF. XMPP-Grid is data format agnostic and accommodates transport of whatever format the end systems agree upon.

XMPP-Grid can operate in the following deployment architectures:

- o Broker-Flow: An XMPP-Grid control plane brokers the authorization and redirects the Topic subscriber to Topic publisher directly. In this architecture, the Controller only manages the connection; the security data flow is directly between Nodes using data formats negotiated out-of-band.
- o Centralized Data-Flow: An XMPP-Grid maintains the data within its optional centralized database. In this architecture, the Controller provides a common information structure for use in formatting and storing security context data, such as IODEF, and directly responds to Node publish and Subscribe requests.
- o Proxy-Flow: An XMPP-Grid is acting as proxy, collecting the data from the publisher(s) and presenting it to the subscriber directly. This is used for ad-hoc queries.

Within the deployment architecture, XMPP-Grid may be used in any combination of the following data exchange modes. The flexibility afforded by the different modes enables security information to be exchanged between systems in the method most suitable for serving a given use-case.

- o Continuous Topic update stream: This mode delivers in real-time any data published to a Topic to the Nodes that are subscribed to that Topic.
- o Directed query: This mode enables Nodes to request a specific set of security information regarding a specific asset, such as a specific user endpoint.
- o Bulk historic data query: This mode enables Nodes to request transfer of past output from a Topic over a specific span of time.

1.3. Benefits of XMPP-Grid

Currently, security information standards such as IODEF [RFC7970] defines a data models that has no explicit transport defined and typically are carried over HTTPS as defined in RID [RFC6545].

As security solutions are expanding to expose and share information asynchronously and across network boundaries there is a need for an architecture that facilitates federation, discovery of the different information available, the interfaces used to obtain the information and the need for near real-time exchange of data.

Based on XMPP, XMPP-Grid has been defined to meet those requirements.

2. XMPP-Grid Architecture

XMPP-Grid is an XMPP-based communication fabric that facilitates secure sharing of information between network elements and networked applications connected to the fabric both in real time and on demand (see figure below).

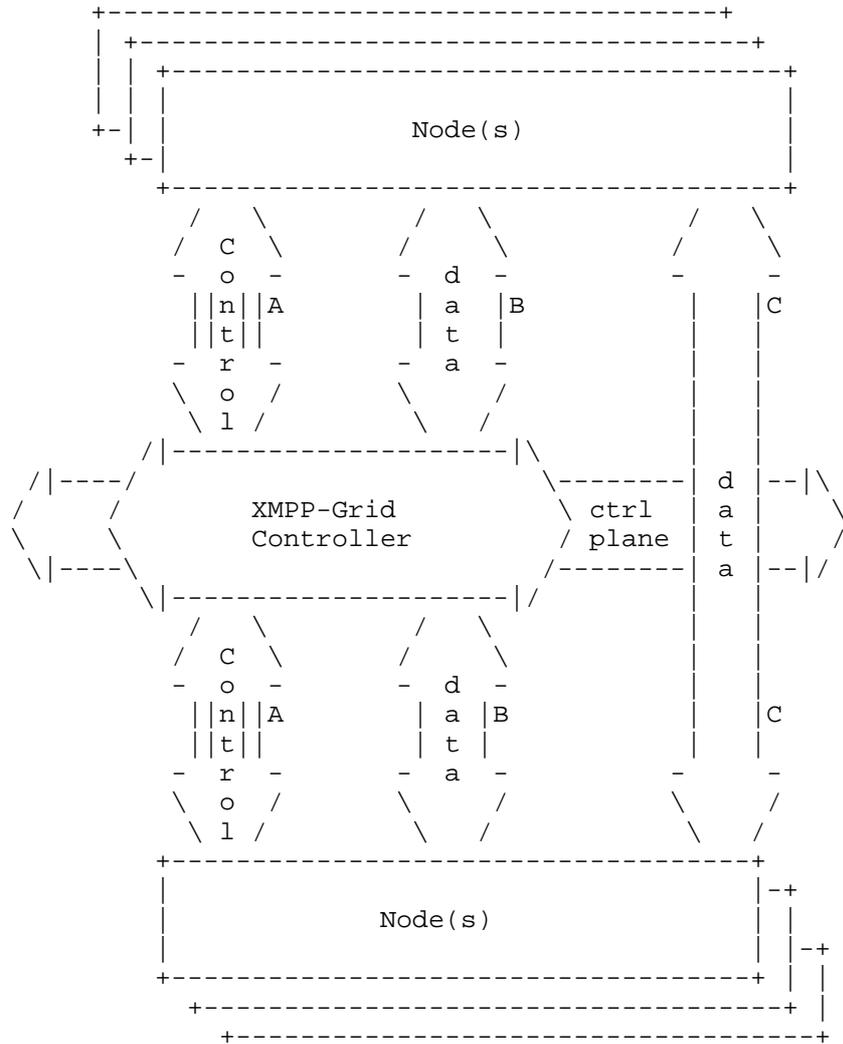


Figure 1: XMPP-Grid Architecture

Nodes must connect to the XMPP-Grid controller to authenticate and establish appropriate authorizations, with appropriate authorization privileges. The control plane messaging is established through XMPP and shown as "A" (Control plane interface) in Figure 1. Authorized nodes may then share data either thru the XMPP-Grid Controller (shown as "B" in Figure 1) or directly (shown as "C" in Figure 1). The data messaging enable Nodes to:

- o Receive real-time events of the published messages from the publisher through Topic subscriptions
- o Make directed queries to other Nodes in the XMPP-Grid with appropriate authorization from the Controller
- o Negotiate out-of-band secure file transfer channel with the peer

2.1. Using XMPP

XMPP is used as the foundation message routing protocol for exchanging security data between systems across XMPP-Grid. XMPP is a communications protocol for message-oriented middleware based on XML. Designed to be extensible, the protocol uses de-centralized client-server architecture where the clients connect to the servers securely and the messages between the clients are routed through the XMPP servers deployed within the cluster. XMPP has been used extensively for publish-subscribe systems, file transfer, video, VoIP, Internet of Things, Smart Grid Software Defined Networks (SDN) and other collaboration and social networking applications. The following are the 4 IETF specifications produced by the XMPP working group:

- o [RFC7590] Extensible Messaging and Presence Protocol (XMPP): Core
- o [RFC6121] Extensible Messaging and Presence Protocol (XMPP): Instant Messaging and Presence
- o [RFC3922] Mapping the Extensible Messaging and Presence Protocol (XMPP) to Common Presence and Instant Messaging (CPIM)
- o [RFC3923] End-to-End Signing and Object Encryption for the Extensible Messaging and Presence Protocol (XMPP)

XMPP offers several of the following salient features for building a security data interexchange protocol:

- o Open - standards-based, decentralized and federated architecture, with no single point of failure

- o Security - Supports domain segregations and federation. Offers strong security via Simple Authentication and Security Layer (SASL) [RFC4422] and Transport Layer Security (TLS) [RFC5246].
- o Real-time event management/exchange - using publish, subscribe notifications
- o Flexibility and Extensibility - XMPP is XML based and is easily extensible to adapt to new use-cases. Custom functionality can be built on top of it.
- o Multiple information exchanges - XMPP offers multiple information exchange mechanisms between the participating clients -
- o
 - * Real-time event notifications through publish and subscribe.
 - * On-demand or directed queries between the clients communicated through the XMPP server
 - * Facilitates out-of-band, direct communication between participating clients
- o Bi-directional - avoids firewall tunneling and avoids opening up a new connection in each direction between client and server.
- o Scalable - supports cluster mode deployment with fan-out and message routing
- o Peer-to-peer communications also enables scale - directed queries and out-of-band file transfer support
- o XMPP offers Node availability, Node service capability discovery, and Node presence within the XMPP network. Nodes ability to detect the availability, presence and capabilities of other participating nodes eases turnkey deployment.

The XMPP extensions used in XMPP-Grid are now part (e.g. publish/subscribe) of the main XMPP specification [RFC7590] and the presence in [RFC6121]. A full list of XMPP Extension Protocols (XEPs) [RFC7590] can be found in <http://xmpp.org/extensions/xep-0001.html> .

2.2. XMPP-Grid Requirements for enabling Information Sharing

This section summarizes the requirements and the extensions used to facilitate the secure sharing of information using XMPP. Knowledge

of the XMPP Protocol and extensions is required to understand this section.

- o Authentication and Authorization: Nodes participating in XMPP-Grid MUST mutually authenticate to the controller using XMPP's authentication mechanisms. Authorization is affected by the controller.
- o Topic Discovery: to facilitate dynamic discovery, Nodes SHOULD support the XMPP Service Discovery [XEP-0030].
- o Publish/Subscribe: to facilitate unsolicited notifications to new or updated security information, Nodes MUST support the XMPP Publish/Subscribe protocol as defined in [RFC7590].

Once a Node has authenticated with the XMPP-Grid controller, it may further register a topic (e.g. information type) to be shared or use the discovery mechanism for determining topics to be consumed. Sharing Information: security information may be shared using registered topics. An example for sharing or consuming the IODEF 1.0 is defined in [XEP-0268].

3. Example use of XMPP-Grid for IODEF

A Node follows the standard XMPP workflow for connecting to the Controller as well as using the XMPP discovery mechanisms to discover the availability to consume IODEF information. The general workflow is summarized in the figure below:

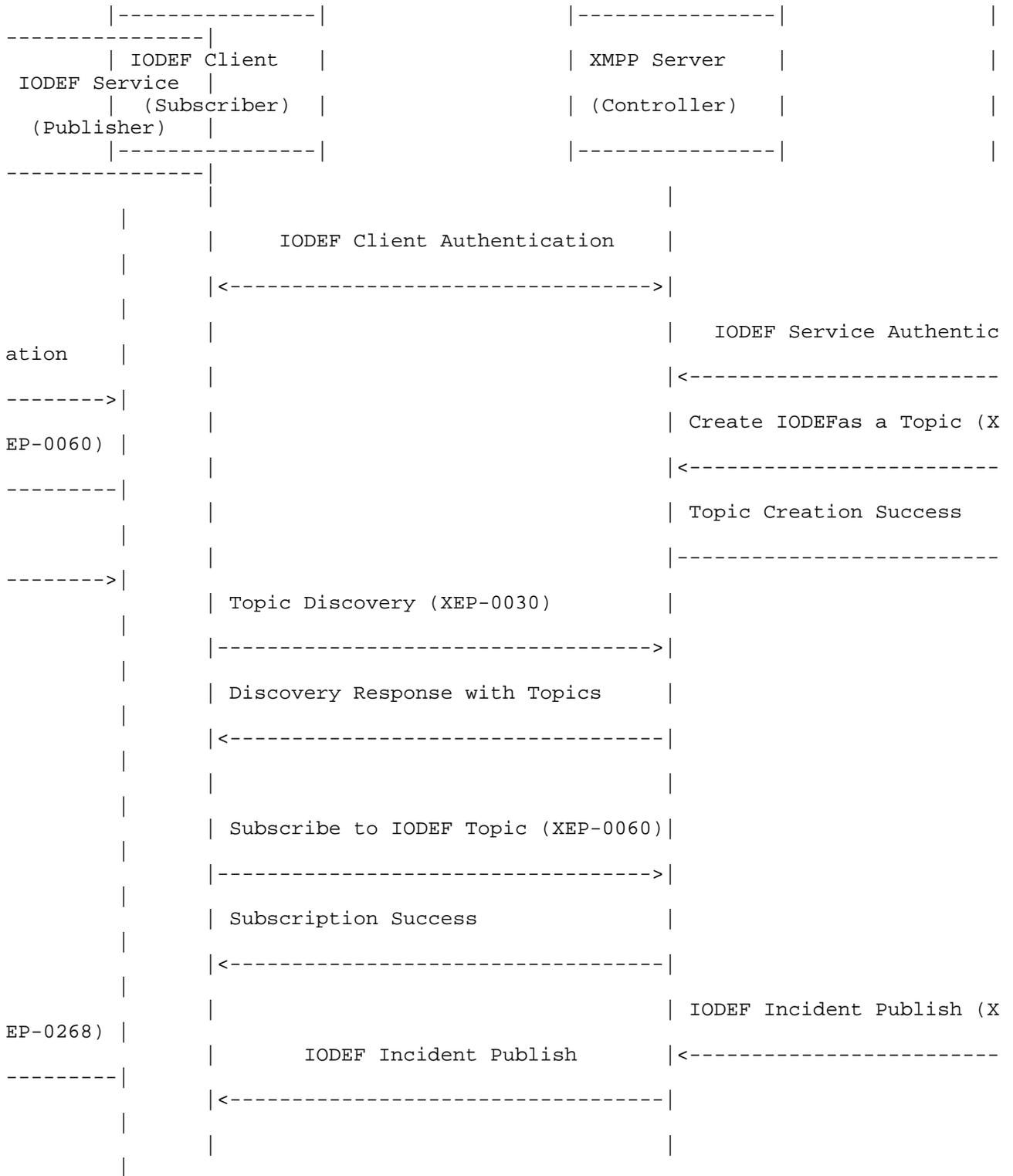


Figure 2: IODEF Example XMPP Workflow

An example XMPP discovery request for an IODEF 1.0 topic is shown below:

```

<iq type='get'
from='iodefclientabc@company.com'
to='pubsub.company.com'
id='nodes1'>

```

```
<query xmlns='http://jabber.org/protocol/disco#items' />
</iq>
```

An example XMPP discovery response for an IODEF 1.0 topic is shown below:

```
<iq type='result'
  from='pubsub.company.com'
  to='iodefclientabc@company.com'
  id='nodes1'>
  <query xmlns='http://jabber.org/protocol/disco#items'>
  <item jid='pubsub.company.com'
    node='incident'
    name='IODEF incident report' />
  </query>
</iq>
```

4. IANA Considerations

IODEF extensions as defined in [XEP-0268] may require IANA considerations and assignment thru the IODEF IANA rules.

5. Security Considerations

An XMPP-Grid Controller serves as an controlling broker for XMPP-Grid Nodes such as Enforcement Points, Policy Servers, CMDBs, and Sensors, using a publish-subscribe-search model of information exchange and lookup. By increasing the ability of XMPP-Grid Nodes to learn about and respond to security-relevant events and data, XMPP-Grid can improve the timeliness and utility of the security system. However, this integrated security system can also be exploited by attackers if they can compromise it. Therefore, strong security protections for XMPP-Grid are essential.

This section provides a security analysis of the XMPP-Grid transport protocol and the architectural elements that employ it, specifically with respect to their use of this protocol. Three subsections define the trust model (which elements are trusted to do what), the threat model (attacks that may be mounted on the system), and the countermeasures (ways to address or mitigate the threats previously identified).

5.1. Trust Model

The first step in analyzing the security of the XMPP-Grid transport protocol is to describe the trust model, listing what each architectural element is trusted to do. The items listed here are assumptions, but provisions are made in the Threat Model and Countermeasures sections for elements that fail to perform as they were trusted to do.

5.1.1. Network

The network used to carry XMPP-Grid messages is trusted to:

- o Perform best effort delivery of network traffic

The network used to carry XMPP-Grid messages is not expected (trusted) to:

- o Provide confidentiality or integrity protection for messages sent over it
- o Provide timely or reliable service

5.1.2. XMPP-Grid Nodes

Authorized XMPP-Grid Nodes are trusted to:

- o Preserve the confidentiality of sensitive data retrieved via the XMPP-Grid Controller

5.1.3. XMPP-Grid Controller

The XMPP-Grid Controller is trusted to:

- o Broker requests for data and enforce authorization of access to this data throughout its lifecycle
- o Perform service requests in a timely and accurate manner
- o Create and maintain accurate operational attributes
- o Only reveal data to and accept service requests from authorized parties

The XMPP-Grid Controller is not expected (trusted) to:

- o Verify the truth (correctness) of data

5.1.4. Certification Authority

The Certification Authority (CA) that issues certificates for the XMPP-Grid Controller and/or XMPP-Grid Nodes (or each CA, if there are several) is trusted to:

- o Ensure that only proper certificates are issued and that all certificates are issued in accordance with the CA's policies

- o Revoke certificates previously issued when necessary
- o Regularly and securely distribute certificate revocation information
- o Promptly detect and report any violations of this trust so that they can be handled

The CA is not expected (trusted) to:

- o Issue certificates that go beyond the XMPP-Grid needs or other constraints imposed by a relying party.

5.2. Threat Model

To secure the XMPP-Grid transport protocol and the architectural elements that implement it, this section identifies the attacks that can be mounted against the protocol and elements.

5.2.1. Network Attacks

A variety of attacks can be mounted using the network. For the purposes of this subsection the phrase "network traffic" should be taken to mean messages and/or parts of messages. Any of these attacks may be mounted by network elements, by parties who control network elements, and (in many cases) by parties who control network-attached devices.

- o Network traffic may be passively monitored to glean information from any unencrypted traffic
- o Even if all traffic is encrypted, valuable information can be gained by traffic analysis (volume, timing, source and destination addresses, etc.)
- o Network traffic may be modified in transit
- o Previously transmitted network traffic may be replayed
- o New network traffic may be added
- o Network traffic may be blocked, perhaps selectively
- o A "Man In The Middle" (MITM) attack may be mounted where an attacker interposes itself between two communicating parties and poses as the other end to either party or impersonates the other end to either or both parties

- o Resist attacks (including denial of service and other attacks from XMPP-Grid Nodes)
- o Undesired network traffic may be sent in an effort to overload an architectural component, thus mounting a denial of service attack

5.2.2. XMPP-Grid Nodes

An unauthorized XMPP-Grid Nodes (one which is not recognized by the XMPP-Grid Controller or is recognized but not authorized to perform any actions) cannot mount any attacks other than those listed in the Network Attacks section above.

An authorized XMPP-Grid Node, on the other hand, can mount many attacks. These attacks might occur because the XMPP-Grid Node is controlled by a malicious, careless, or incompetent party (whether because its owner is malicious, careless, or incompetent or because the XMPP-Grid Node has been compromised and is now controlled by a party other than its owner). They might also occur because the XMPP-Grid Node is running malicious software; because the XMPP-Grid Node is running buggy software (which may fail in a state that floods the network with traffic); or because the XMPP-Grid Node has been configured improperly. From a security standpoint, it generally makes no difference why an attack is initiated. The same countermeasures can be employed in any case.

Here is a list of attacks that may be mounted by an authorized XMPP-Grid Node:

- o Cause many false alarms or otherwise overload the XMPP-Grid Controller or other elements in the network security system (including human administrators) leading to a denial of service or disabling parts of the network security system
- o Omit important actions (such as posting incriminating data), resulting in incorrect access
- o Use confidential information obtained from the XMPP-Grid Controller to enable further attacks (such as using endpoint health check results to exploit vulnerable endpoints)
- o Advertise data crafted to exploit vulnerabilities in the XMPP-Grid Controller or in other XMPP-Grid Nodes, with a goal of compromising those systems
- o Issue a search request or set up a subscription that matches an enormous result, leading to resource exhaustion on the XMPP-Grid Controller, the publishing XMPP-Grid Node, and/or the network

- o Establish a communication channel using another XMPP-Grid Node's session-id

Dependencies of or vulnerabilities of authorized XMPP-Grid Nodes may be exploited to effect these attacks. Another way to effect these attacks is to gain the ability to impersonate an XMPP-Grid Node (through theft of the XMPP-Grid Node's identity credentials or through other means). Even a clock skew between the XMPP-Grid Node and XMPP-Grid Controller can cause problems if the XMPP-Grid Node assumes that old XMPP-Grid Node data should be ignored.

5.2.3. XMPP-Grid Controllers

An unauthorized XMPP-Grid Controller (one which is not trusted by XMPP-Grid Nodes) cannot mount any attacks other than those listed in the Network Attacks section above.

An authorized XMPP-Grid Controller can mount many attacks. Similar to the XMPP-Grid Node case described above, these attacks might occur because the XMPP-Grid Controller is controlled by a malicious, careless, or incompetent party (either an XMPP-Grid Controller administrator or an attacker who has seized control of the XMPP-Grid Controller). They might also occur because the XMPP-Grid Controller is running malicious software, because the XMPP-Grid Controller is running buggy software (which may fail in a state that corrupts data or floods the network with traffic), or because the XMPP-Grid Controller has been configured improperly.

All of the attacks listed for XMPP-Grid Node above can be mounted by the XMPP-Grid Controller. Detection of these attacks will be more difficult since the XMPP-Grid Controller can create false operational attributes and/or logs that imply some other party created any bad data.

Additional XMPP-Grid Controller attacks may include:

- o Expose different data to different XMPP-Grid Nodes to mislead investigators or cause inconsistent behavior
- o Mount an even more effective denial of service attack than a single XMPP-Grid Node could
- o Obtain and cache XMPP-Grid Node credentials so they can be used to impersonate XMPP-Grid Nodes even after a breach of the XMPP-Grid Controller is repaired

- o Obtain and cache XMPP-Grid Controller administrator credentials so they can be used to regain control of the XMPP-Grid Controller after the breach of the XMPP-Grid Controller is repaired

Dependencies of or vulnerabilities of the XMPP-Grid Controller may be exploited to obtain control of the XMPP-Grid Controller and effect these attacks.

5.2.4. Certification Authority

A Certification Authority trusted to issue certificates for the XMPP-Grid Controller and/or XMPP-Grid Nodes can mount several attacks:

- o Issue certificates for unauthorized parties, enabling them to impersonate authorized parties such as the XMPP-Grid Controller or an XMPP-Grid Node. This can lead to all the threats that can be mounted by the certificate's subject.
- o Issue certificates without following all of the CA's policies. Because this can result in issuing certificates that may be used to impersonate authorized parties, this can lead to all the threats that can be mounted by the certificate's subject.
- o Fail to revoke previously issued certificates that need to be revoked. This can lead to undetected impersonation of the certificate's subject or failure to revoke authorization of the subject, and therefore can lead to all of the threats that can be mounted by that subject.
- o Fail to regularly and securely distribute certificate revocation information. This may cause a relying party to accept a revoked certificate, leading to undetected impersonation of the certificate's subject or failure to revoke authorization of the subject, and therefore can lead to all of the threats that can be mounted by that subject. It can also cause a relying party to refuse to proceed with a transaction because timely revocation information is not available, even though the transaction should be permitted to proceed.
- o Allow the CA's private key to be revealed to an unauthorized party. This can lead to all the threats above. Even worse, the actions taken with the private key will not be known to the CA.
- o Fail to promptly detect and report errors and violations of trust so that relying parties can be promptly notified. This can cause the threats listed earlier in this section to persist longer than necessary, leading to many knock-on effects.

5.3. Countermeasures

Below are countermeasures for specific attack scenarios to the XMPP-Grid infrastructure.

5.3.1. Securing the XMPP-Grid Transport Protocol

To address network attacks, the XMPP-Grid transport protocol described in this document requires that the XMPP-Grid messages MUST be carried over TLS (minimally TLS 1.2 [RFC5246]) as described in [RFC2818]. The XMPP-Grid Node MUST verify the XMPP-Grid Controller's certificate and determine whether the XMPP-Grid Controller is trusted by this XMPP-Grid Node before completing the TLS handshake. The XMPP-Grid Controller MUST authenticate the XMPP-Grid Node either using mutual certificate-based authentication in the TLS handshake or using Basic Authentication as described in IETF RFC 2617. XMPP-Grid Controller MUST use Simple Authentication and Security Layer (SASL), described in [RFC4422], to support the aforesaid authentication mechanisms. SASL offers authentication mechanism negotiations between the XMPP-Grid Controller and XMPP-Grid node during the connection establishment phase. XMPP-Grid Nodes and XMPP-Grid Controllers using mutual certificate-based authentication SHOULD each verify the revocation status of the other party's certificate. All XMPP-Grid Controllers and XMPP-Grid Nodes MUST implement both mutual certificate-based authentication and Basic Authentication. The selection of which XMPP-Grid Node authentication technique to use in any particular deployment is left to the administrator.

An XMPP-Grid Controller MAY also support a local, configurable set of Basic Authentication userid-password pairs. If so, it is implementation dependent whether an XMPP-Grid Controller ends a session when an administrator changes the configured password. Since Basic Authentication has many security disadvantages (especially the transmission of reusable XMPP-Grid Node passwords to the XMPP-Grid Controller), it SHOULD only be used when absolutely necessary. Per the HTTP specification, when basic authentication is in use, an XMPP-Grid Controller MAY respond to any request that lacks credentials with an error code similar to HTTP code 401. An XMPP-Grid Node SHOULD avoid this code by submitting basic auth credentials with every request when basic authentication is in use. If it does not do so, an XMPP-Grid Node MUST respond to this code by resubmitting the same request with credentials (unless the XMPP-Grid Node is shutting down).

As XMPP uses TLS as the transport and security mechanisms, it is understood that best practices such as those in [I-D.ietf-uta-tls-bcp] are followed.

These protocol security measures provide protection against all the network attacks listed in the above document section except denial of service attacks. If protection against these denial of service attacks is desired, ingress filtering, rate limiting per source IP address, and other denial of service mitigation measures may be employed. In addition, an XMPP-Grid Controller MAY automatically disable a misbehaving XMPP-Grid Node.

5.3.2. Securing XMPP-Grid Nodes

XMPP-Grid Nodes may be deployed in locations that are susceptible to physical attacks. Physical security measures may be taken to avoid compromise of XMPP-Grid Nodes, but these may not always be practical or completely effective. An alternative measure is to configure the XMPP-Grid Controller to provide read-only access for such systems. The XMPP-Grid Controller SHOULD also include a full authorization model so that individual XMPP-Grid Nodes may be configured to have only the privileges that they need. The XMPP-Grid Controller MAY provide functional templates so that the administrator can configure a specific XMPP-Grid Node as a DHCP server and authorize only the operations and metadata types needed by a DHCP server to be permitted for that XMPP-Grid Node. These techniques can reduce the negative impacts of a compromised XMPP-Grid Node without diminishing the utility of the overall system.

To handle attacks within the bounds of this authorization model, the XMPP-Grid Controller MAY also include rate limits and alerts for unusual XMPP-Grid Node behavior. XMPP-Grid Controllers SHOULD make it easy to revoke an XMPP-Grid Node's authorization when necessary. Another way to detect attacks from XMPP-Grid Nodes is to create fake entries in the available data (honeytokens) which normal XMPP-Grid Nodes will not attempt to access. The XMPP-Grid Controller SHOULD include auditable logs of XMPP-Grid Node activities.

To avoid compromise of XMPP-Grid Node, XMPP-Grid Node SHOULD be hardened against attack and minimized to reduce their attack surface. They should be well managed to minimize vulnerabilities in the underlying platform and in systems upon which the XMPP-Grid Node depends. Personnel with administrative access should be carefully screened and monitored to detect problems as soon as possible.

5.3.3. Securing XMPP-Grid Controllers

Because of the serious consequences of XMPP-Grid Controller compromise, XMPP-Grid Controllers SHOULD be especially well hardened against attack and minimized to reduce their attack surface. They should be well managed to minimize vulnerabilities in the underlying platform and in systems upon which the XMPP-Grid Controller depends.

Network security measures such as firewalls or intrusion detection systems may be used to monitor and limit traffic to and from the XMPP-Grid Controller. Personnel with administrative access should be carefully screened and monitored to detect problems as soon as possible. Administrators should not use password-based authentication but should instead use non-reusable credentials and multi-factor authentication (where available). Physical security measures SHOULD be employed to prevent physical attacks on XMPP-Grid Controllers.

To ease detection of XMPP-Grid Controller compromise should it occur, XMPP-Grid Controller behavior should be monitored to detect unusual behavior (such as a reboot, a large increase in traffic, or different views of an information repository for similar XMPP-Grid Nodes). XMPP-Grid Nodes should log and/or notify administrators when peculiar XMPP-Grid Controller behavior is detected. To aid forensic investigation, permanent read-only audit logs of security-relevant information (especially administrative actions) should be maintained. If XMPP-Grid Controller compromise is detected, a careful analysis should be performed of the impact of this compromise. Any reusable credentials that may have been compromised should be reissued.

5.3.4. Limit on search result size

While XMPP-Grid is designed for high scalability to 100,000s of Nodes, an XMPP-Grid Controller MAY establish a limit to the amount of data it is willing to return in search or subscription results. This mitigates the threat of an XMPP-Grid Node causing resource exhaustion by issuing a search or subscription that leads to an enormous result.

5.3.5. Cryptographically random session-id and authentication checks for ARC

An XMPP-Grid Controller SHOULD ensure that the XMPP-Grid Node establishing an Authenticated Results Chain (ARC) is the same XMPP-Grid Node as the XMPP-Grid Node that established the corresponding Synchronization Source Identifier (SSRC). The XMPP-Grid Controller SHOULD employ both of the following strategies:

- o session-ids SHOULD be cryptographically random
- o The HTTPS transport for the SSRC and the ARC SHOULD be authenticated using the same credentials. SSL session resumption MAY be used to establish the ARC based on the SSRC SSL session.

5.3.6. Securing the Certification Authority

As noted above, compromise of a Certification Authority (CA) trusted to issue certificates for the XMPP-Grid Controller and/or XMPP-Grid Nodes is a major security breach. Many guidelines for proper CA security have been developed: the CA/Browser Forum's Baseline Requirements, the AICPA/CICA Trust Service Principles, etc. The CA operator and relying parties should agree on an appropriately rigorous security practices to be used.

Even with the most rigorous security practices, a CA may be compromised. If this compromise is detected quickly, relying parties can remove the CA from their list of trusted CAs, and other CAs can revoke any certificates issued to the CA. However, CA compromise may go undetected for some time, and there's always the possibility that a CA is being operated improperly or in a manner that is not in the interests of the relying parties. For this reason, relying parties may wish to "pin" a small number of particularly critical certificates (such as the certificate for the XMPP-Grid Controller). Once a certificate has been pinned, the relying party will not accept another certificate in its place unless the Administrator explicitly commands it to do so. This does not mean that the relying party will not check the revocation status of pinned certificates. However, the Administrator may still be consulted if a pinned certificate is revoked, since the CA and revocation process are not completely trusted.

5.4. Summary

XMPP-Grid's considerable value as a broker for security-sensitive data exchange distribution also makes the protocol and the network security elements that implement it a target for attack. Therefore, strong security has been included as a basic design principle within the XMPP-Grid design process.

The XMPP-Grid transport protocol provides strong protection against a variety of different attacks. In the event that an XMPP-Grid Node or XMPP-Grid Controller is compromised, the effects of this compromise have been reduced and limited with the recommended role-based authorization model and other provisions, and best practices for managing and protecting XMPP-Grid systems have been described. Taken together, these measures should provide protection commensurate with the threat to XMPP-Grid systems, thus ensuring that they fulfill their promise as a network security clearing-house.

6. Privacy Considerations

XMPP-Grid Nodes may publish information about endpoint health, network access, events (which may include information about what services an endpoint is accessing), roles and capabilities, and the identity of the end user operating the endpoint. Any of this published information may be queried by other XMPP-Grid Nodes and could potentially be used to correlate network activity to a particular end user.

Dynamic and static information brokered by an XMPP-Grid Controller, ostensibly for purposes of correlation by XMPP-Grid Nodes for intrusion detection, could be misused by a broader set of XMPP-Grid Nodes which hitherto have been performing specific roles with strict well-defined separation of duties.

Care should be taken by deployers of XMPP-Grid to ensure that the information published by XMPP-Grid Nodes does not violate agreements with end users or local and regional laws and regulations. This can be accomplished either by configuring XMPP-Grid Nodes to not publish certain information or by restricting access to sensitive data to trusted XMPP-Grid Nodes. That is, the easiest means to ensure privacy or protect sensitive data, is to omit or not share it at all.

Another consideration for deployers is to enable end-to-end encryption to ensure the data is protected from the data layer to data layer and thus protect it from the transport layer.

7. Acknowledgements

The authors would like to acknowledge the contributions, authoring and/or editing of the following people: Joseph Salowey, Lisa Lorenzin, Clifford Kahn, Henk Birkholz, Jessica Fitzgerald-McKay, Steve Hanna, and Steve Venema. In addition, we want to thank Takeshi Takahashi, Panos Kampanakis, Adam Montville and Chris Inacio for reviewing and providing valuable comments.

8. References

8.1. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <<http://www.rfc-editor.org/info/rfc2119>>.

- [RFC3922] Saint-Andre, P., "Mapping the Extensible Messaging and Presence Protocol (XMPP) to Common Presence and Instant Messaging (CPIM)", RFC 3922, DOI 10.17487/RFC3922, October 2004, <<http://www.rfc-editor.org/info/rfc3922>>.
- [RFC3923] Saint-Andre, P., "End-to-End Signing and Object Encryption for the Extensible Messaging and Presence Protocol (XMPP)", RFC 3923, DOI 10.17487/RFC3923, October 2004, <<http://www.rfc-editor.org/info/rfc3923>>.
- [RFC4422] Melnikov, A., Ed. and K. Zeilenga, Ed., "Simple Authentication and Security Layer (SASL)", RFC 4422, DOI 10.17487/RFC4422, June 2006, <<http://www.rfc-editor.org/info/rfc4422>>.
- [RFC6121] Saint-Andre, P., "Extensible Messaging and Presence Protocol (XMPP): Instant Messaging and Presence", RFC 6121, DOI 10.17487/RFC6121, March 2011, <<http://www.rfc-editor.org/info/rfc6121>>.
- [RFC7590] Saint-Andre, P. and T. Alkemade, "Use of Transport Layer Security (TLS) in the Extensible Messaging and Presence Protocol (XMPP)", RFC 7590, DOI 10.17487/RFC7590, June 2015, <<http://www.rfc-editor.org/info/rfc7590>>.
- [XEP-0030] Hildebrand, J., Millard, P., Eatmon, R., and P. Saint-Andre, "Service Discovery", XSF XEP 0030, July 2010.
- [XEP-0060] Millard, P. and P. Saint-Andre, "Publish-Subscribe", XSF XEP 0060, December 2016.
- [XEP-0268] Hefczyk, A., Jensen, F., Remond, M., Saint-Andre, P., and M. Wild, "Service Discovery", XSF XEP 0268, MY 2012.

8.2. Informative References

- [I-D.ietf-mile-rolie] Field, J., Banghart, S., and D. Waltermire, "Resource-Oriented Lightweight Information Exchange", draft-ietf-mile-rolie-07 (work in progress), May 2017.
- [I-D.ietf-uta-tls-bcp] Sheffer, Y., Holz, R., and P. Saint-Andre, "Recommendations for Secure Use of TLS and DTLS", draft-ietf-uta-tls-bcp-11 (work in progress), February 2015.

- [RFC2818] Rescorla, E., "HTTP Over TLS", RFC 2818, DOI 10.17487/RFC2818, May 2000, <<http://www.rfc-editor.org/info/rfc2818>>.
- [RFC5246] Dierks, T. and E. Rescorla, "The Transport Layer Security (TLS) Protocol Version 1.2", RFC 5246, DOI 10.17487/RFC5246, August 2008, <<http://www.rfc-editor.org/info/rfc5246>>.
- [RFC6545] Moriarty, K., "Real-time Inter-network Defense (RID)", RFC 6545, DOI 10.17487/RFC6545, April 2012, <<http://www.rfc-editor.org/info/rfc6545>>.
- [RFC6546] Trammell, B., "Transport of Real-time Inter-network Defense (RID) Messages over HTTP/TLS", RFC 6546, DOI 10.17487/RFC6546, April 2012, <<http://www.rfc-editor.org/info/rfc6546>>.
- [RFC7970] Danyliw, R., "The Incident Object Description Exchange Format Version 2", RFC 7970, DOI 10.17487/RFC7970, November 2016, <<http://www.rfc-editor.org/info/rfc7970>>.

Authors' Addresses

Nancy Cam-Winget (editor)
Cisco Systems
3550 Cisco Way
San Jose, CA 95134
USA

Email: ncamwing@cisco.com

Syam Appala
Cisco Systems
3550 Cisco Way
San Jose, CA 95134
USA

Email: syaml@cisco.com

Scott Pope
Cisco Systems
5400 Meadows Road
Suite 300
Lake Oswego, OR 97035
USA

Email: scottp@cisco.com