GFD-C

Category: Community Practice Documents

CA Operations WG

September 2004

OCSP Requirements for Grids

Status of this Memo

This is an informational track document.

.

Copyright Notice

Abstract

Grids use X.509 certificates for authentication and authorization. These certificates have built-in lifetimes, but this is insufficient: lists of revoked certificates are required by many relying parties, and should be used by every relying party, in order to eliminate lost, compromised, or otherwise-invalid certificates from use. Commercial credit and debit cards are managed in an analogous fashion. The Online Certificate Status Protocol (OCSP) is a protocol that can be used to provide this service for Grid stakeholders. OCSP is a simple query protocol, relieving its clients of the burden of managing lists of revoked certificates. Since the OCSP protocol is made to be flexible and extensible, certificate validation services beyond reporting of contents of certificate revocation lists (CRLs) could be provided. The Grid presents considerable challenges for such a service, however. To be suitable for Grid use, OCSP services must be discoverable. Grid administrators need to develop interoperability methods, "chaining" methods from one OCSP to another, and replication techniques.

Table of Contents

1 Introduction

Grids use X.509 certificates for authentication and as a basis for authorization. A reliable, secure Grid infrastructure depends on the integrity of these certificates. X.509 certificates have built-in lifetimes, but this is not adequate to deal with every aspect of certificate life cycle and management. Certificates can be lost by their owners, can be "compromised", or the justification for holding the certificate may no longer apply. These certificates need to be revoked before the certificate expiration date is reached. Distribution of certificate revocation lists (CRLs) support this need. CRL distribution and maintenance in the Grid has proven difficult. The Online Certificate Status Protocol (OCSP) provides a simple query protocol for clients to perform revocation check lookups on certificates without the need to maintain up-to-date sets of CRLs from a number of different certification authorities.

OCSP is a product of the IETF PKIX working group, and the current version is described in [RFC2560].

2 Practical Considerations

[In this section, we discuss which software packages support this, such as revisions of openssl, gsi, java, and any other important support api's; browser support; Apache web servers; other web servers or the like.

We may need to discuss incompatibility or limitations (for instance, Microsoft IE doesn't support OCSP directly). Identify missing software / software requiring development or integration issues that stakeholders may need to deal with]

3 Protocol overview

In OCSP, a relying party (client) identifies the location of an OCSP responder (server) in one of two ways:

1. Local configuration: a table associating issuer names (CA distinguished names) with one or many URLs of OCSP responders to contact.

2. Self-described: the URL(s) of the OCSP responder(s) knowing the status of a subscriber's certificate is specified in the certificate itself, as an *AuthorityInfoAccess* (*AIA*) certificate extension.

The relying party sends an OCSP request message to the identified OCSP responder, normally carried over HTTP or HTTPS using HTTP POST[1] operations. The request identifies one or several[2] *subscriber certificates*, identified by issuer name (hashed), issuer key (hashed), certificate serial number, and possible *AIA* information. The request may be signed (mainly intended for authorization and billing purposes), in which case the signing certificate is attached. In addition, the request may also contain a *nonce*, a random sequence of bytes that render the request unique.

The OCSP responder in turn provides a signed response that contains the timestamp of the signature and current status of the certificate(s) identified in the request. Any nonce is copied ad verbatim to the request. The responder typically also includes its signing certificate, and certification path.

The possible certificate status codes returned by the OCSP responder are *Good*, *Revoked*, and *Unknown*. The *Unknown* state indicates that the responder is unable to answer the request for *some* reason: it might not know of the CA that issued the certificate, a local copy of the revocation database is not up-to-date, and so on.

In case of an error, OCSP defines a set of common error codes which are sent back to the relying party *non-signed*.

The OCSP RFC specifies three cases for when a relying party may accept an OCSP response:

1. The response is signed by the CA that issued the subscriber certificate(s). This is specified by the RFC, but is rarely used in practice.
2. The response is signed by an *Authorized responder*, with a direct delegated authority from the CA. This delegation is identified by inclusion of *OCSPSigning* in the *extendedKeyUsage* extension of the responder's certificate.

---

[1] HTTP GET is supported as well but not as common and only suitable for trivial use of OCSP due to the message size limitation.

[2] While OCSP supports querying of multiple certificates in a single request, it is rarely used in practice, and the support in common off-the-shelf implementations for this mode of operation is questionable.

3. The response is signed by a *Trusted responder*, a responder explicitly trusted by the relying party (local configuration). Relying parties must know about potential Trusted responders ahead of time.

While use of Authorized responders above is the most scalable and maintainable configuration, it creates a circular dependency problem in that relying parties may wish to know whether the responder certificate has been revoked or not before accepting the signature of the OCSP response[3]. To address this and other similar problems, a certificate can be marked with the *ocsp-nocheck* extension, indicating to relying parties to not attempt to verify the revocation status of the certificate.

The *ocsp-nocheck* extension is typically combined with relatively short-lived certificates and close supervision of the issuer. It can also be used to mitigate anticipated load problems: for instance, the TLS certificate of a central server may be equipped with the *ocsp-nocheck* extension (and necessary operational improvements to mitigate the risks associated with that) to eliminate OCSP queries otherwise triggered by all clients connecting to that server.

The relying party performs additional checks before accepting an OCSP response: these include validating the *freshness* of the OCSP response (checking that OCSP response's *producedAt* timestamp is within the allowed lifetime), that the responded nonce matches what was sent, and so on.

4 Client requirements

For OCSP to be used in wide area, multi-organizational Grid environments, support for it must be integrated in any path validation software used.

4.1 Network connectivity requirements

OCSP clients MUST be able to send OCSP requests over HTTP or HTTPS, which may affect the network and firewall policies of a site.

[3] Or, in the case of OCSP over HTTPS, the client wants to validate the TLS certificate of the server hosting the OCSP responder, and whose revocation status can only be obtained by connecting to the OCSP responder…

4.2 Revocation source requirements

In some scenarios, CRLs provide better a better means of processing revocation than OCSP; for instance, in the case of server-side validation, it is beneficiary from a performance point of view to have all revocation information cached locally. In other scenarios, CRLs may be used as a backup source of revocation in case contacting the OCSP responder fails, e.g. due to temporary network outage. In case of an OCSP rollout, any and all CAs will not be able to provide an OCSP service overnight, but rather the most likely scenario is that relying parties will see a gradual transition from CRL to OCSP as the primary source of revocation information.

To accommodate for this, clients MUST be capable of handling both CRLs and OCSP, and it MUST be a configurable option which source of revocation to prefer and which to use as a backup on a per-issuer basis.

4.3 Caching of responses

It is often the case that an application interacts to some other system component on a frequent basis. Instead of each interaction triggering a new OCSP request, caching of responses SHOULD be supported by the client. We note that OCSP responses with reported certificate status *Revoked* can be cached indefinitely.

It MUST be possible to configure the maximum size and lifetime/freshness of the entries in an OCSP response cache.

4.4 Responder discovery

At a minimum, the OCSP client MUST be able to locate the OCSP responder using the methods specified in Section 3 . Local configuration has precedence over any service locator information located in the certificate's *AIA* extension. A default responder for "all other" issuers SHOULD be configurable as well.

To allow for high-availability and load balancing, it SHOULD be possible to associate each issuer name with more than one responder URL.

4.5 Nonce

Large-scale OCSP responder implementations, such as the ones deployed by VeriSign, continuously pre-produce OCSP responses in the background, for maximum throughput.

Pre-produced OCSP responses can not be used to service nonced requests. For this reason, OCSP clients SHOULD NOT make use of nonce in the requests.

(Nonce is useful to ensure up-to-date, unique responses from the OCSP responder, necessary in deployments with strong audit requirements, such as financial markets. For the general Grid community, we consider the use of nonce a bit of an overkill.)

4.6 AuthorityInfoAccess

OCSP clients SHOULD include any information about an Authorized responder location (as indicated by OCSP service locator URLs in the *AIA* extension of the subscriber certificate) in its OCSP request.

4.7 Error handling and the Unknown status code

It MUST be configurable how the OCSP clients handle errors and the *Unknown* status code: in most cases, such a failure means "try somewhere else", indicating to the client to probe other possible candidate sources (including other OCSP responders) for revocation information. In case the resulting status after an exhausted search is still an error or status *Unknown*, the client SHOULD interpret that as *Revoked* with status *OnHold* (that is, a non-definite revocation state), unless otherwise configured.

5 Server Requirements

Our experience with OCSP servers has been with commercial products with considerable capabilities (and in some cases, considerable limitations). A server based on open-source software such as openssl or a java implementation should be developed in tandem with this requirements definition.

5.1 Performance and key protection

Most OCSP responder implementations easily handle up to ~100 requests per second on a commodity desktop workstation. Use of cryptographic hardware may accelerate that further, and also provide adequate protection of the signature keys. We do not require the use of hardware protection, but RECOMMEND it.

5.2 Responder certificate

An OCSP responder implementation MUST support for live updates of the signature key material. (Use case: short-lived certificates signed with the *ocsp-nocheck* extension – see Section 3).

The responder SHOULD support for handling multiple signing certificates simultaneously: one for each issuer that it service revocation information for. (Use case: an Authorized responder servicing more than one CA)

5.3 Revocation sources

CRLs, local databases, replication of those, …

5.4 Transponder mode

Transponder: forward requests to other responders, forward reply back. Combined with caching they are still useful.

5.5 Proxy certificates

Provide the means for a user to register proxy certs with a responder so they can be revoked

6 CA / Certificate Issuer Requirements

6.1 End entity certificates

The service locator (URL) of a responder SHOULD be included in all certificates issued by all CAs for which the responder is an Authorized responder. The service locator is encoded in an *AuthorityInfoAccess* certificate extension, as specified in [RFC2560].

6.2 Certifying Authorized responders

A CA's Authorized responder SHOULD be issued a certificate with the *ocsp-nocheck* extension as well as *OCSPSigning* in the *extendedKeyUsage* extension, as described in [RFC2560]. Direct use of the CA's private key for OCSP response signing is not recommended.

6.3 Revocation information propagation

A common source of revocation information for a responder is by periodically downloading CRLs. For Authorized responders in particular, removing the delay introduced by this pull model is desirable.

When a CA issues a new CRL, it SHOULD initiate an immediate update (push) of the revocation information available to all its Authorized responders.

7 Service architecture

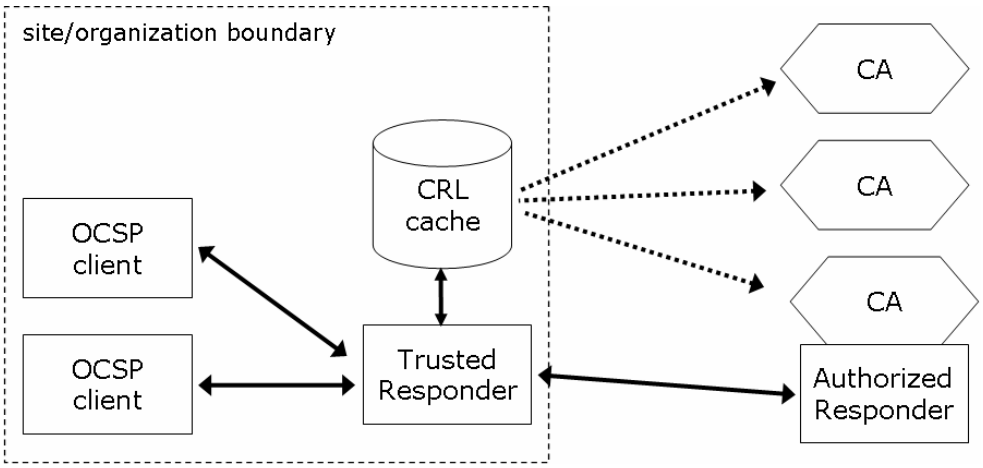

Figure 1.Overall OCSP service architecture

The overall OCSP service architecture for Grids is depicted in Figure 1. It econsists of three classes of components

- Authorized OCSP responders
- Trusted OCSP responders
- OCSP clients

All of the components may appear in multiple instances and should be configurable according to the needs of their operators.

A typical site configuration includes at least two Trusted responders configured to serve certificate status information for every accepted CA from its CRL or by requesting an Authorized responder whenever such exists (indicated by local configuration or presence of a service locator URL in the incoming OCSP request, in turn copied by the OCSP client from the subscriber certificate). In the latter case, the Trusted responders may act in transponder mode, and CRL information may be used as a fallback source of revocation information.

All software verifying certificates SHOULD use OCSP to request certificate status from the Trusted responders. In case the Trusted responders are unavailable, an Authorized responder may be requested directly, or a CRL may be obtained and consulted, unless such actions would conflict with local site policy.

7.1 Authorized responders

Conforming CAs make use of Authorized responders to provide a primary source of certificate status information. The CA is responsible for keeping the Authorized responder's revocation information updated at all times, as described in 6.3.

The operator of an Authorized responder SHOULD ensure high availability of the OCSP service, for instance by operating several responder independent instances or make use of fault-tolerant or mirrored systems.

7.2 Trusted responders

Trusted responders are usually operated by sites or at an organizational level, to provide OCSP service to local OCSP clients, thus centralizing the complexity of the OCSP/CRL configuration, and minimizing the need for outgoing network connectivity.

Trusted responders SHOULD enable caching of OCSP response, to reduce the load on the Authorized responders.

Trusted responders MAY operate in transponder mode. However, this requires that the OCSP clients are configured to trust any external responders whose OCSP responses are forwarded to the client (e.g., by only act as transponder when contacting an Authorized responder).

7.3 OCSP clients

It is RECOMMENDED that OCSP clients are configured to use the Trusted responder(s) as the only source of certificate status information. OCSP Clients SHOULD NOT make use of nonce in their requests.

8 Glossary

None

Intellectual Property Statement

The GGF takes no position regarding the validity or scope of any intellectual property or other rights that might be claimed to pertain to the implementation or use of the technology described in this document or the extent to which any license under such rights might or might not be available; neither does it represent that it has made any effort to identify any such rights. Copies of claims of rights made available for publication and any assurances of licenses to be made available, or the result of an attempt made to obtain a general license or permission for the use of such proprietary rights by implementers or users of this specification can be obtained from the GGF Secretariat.

The GGF invites any interested party to bring to its attention any copyrights, patents or patent applications, or other proprietary rights which may cover technology that may be required to practice this recommendation. Please address the information to the GGF Executive Director.

Full Copyright Notice

IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE INFORMATION HEREIN WILL NOT INFRINGE ANY RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE."

References

[RFC2560] Myers et al, *X.509 Internet Public Key Infrastructure Online Certificate Status Protocol – OCSP*. Available at http://www.ietf.org/rfc/rfc2560.txt

[RFC3280] Housley et al, *Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile.* Available at http://www.ietf.org/rfc/rfc3280.txt