X.509v3 TLS Feature Extension

TLS Feature Extension

```
<ietf=draft-hallambaker-tlsfeature std trust200902 id sec>
<version>07
<author>Phillip Hallam-Baker
    <lastname>Hallam-Baker
    <initials>P. M.
    <firstname>Phillip
    <organization>Comodo Group Inc.
    <email>philliph@comodo.com
<bibliography="Bibliography.xml">
<bibliography="Cache.xml" cache="true">
```

Abstract

The purpose of the TLS Feature extension is to prevent downgrade attacks that are not otherwise prevented by the TLS protocol. In particular, the TLS Feature extension may be used to mandate support for revocation checking features in the TLS protocol such as OCSP stapling. Informing clients that an OCSP status response will always be stapled permits an immediate failure in the case that the response is not stapled. This in turn prevents a denial of service attack that might otherwise be possible.

Definitions

Requirements Language

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 RFC 2119 [!RFC2119].

TLS Feature

In order to avoid the confusion that would occur in attempting to describe an X.509 extension describing the use of TLS extensions, in this document the term 'extension' is reserved to refer to X.509v3 extensions and the term 'feature' is used to refer to a TLS extension.

Purpose

The purpose of the TLS Feature extension is to prevent downgrade attacks that are not otherwise prevented by the TLS protocol.

Since the TLS protocol itself provides strong protection against most forms of downgrade attack including downgrade attacks against cipher suite choices offered and client credentials, the TLS Feature is only relevant to the validation of TLS protocol credentials. In particular to the revocation status of the server credentials presented.

At the time of writing, the only TLS feature extensions that are relevant to the revocation status of credentials are the Certificate Status Request extension (status_request) and the Multiple Certificate Status Extension (status_request_v2). These extensions are used to support in-band exchange of OCSP tokens, otherwise known as OCSP stapling. These extensions are described in [!RFC6066] and [!RFC6961].

The OCSP stapling mechanism described in [!RFC6066] permits a TLS server to provide evidence of valid certificate status inband. When this information is provided inband, the privacy, performance and reliability concerns arising from the need to make a third party connection during the TLS handshake are eliminated. A client cannot however draw any conclusion from the absence of inband status information unless it knows that the legitimate server would have provided it. The status information might have been omitted because the server does not support the extension or because the server is witholding the information intentionally, knowing the certificate to be invalid.

The inclusion of a TLS feature extension advertising the status_request feature in the server end entity certificate permits a client to fail immediately if the certificate status information is not provided by the server. The need to query the OCSP responder is eliminated entirely. This improves client efficiency and more importantly prevents a denial of service attack against the client by either blocking the OCSP response or mounting a denial of service attack against the OCSP responder.

Since the TLS Feature extension is an option, it is not likely that an attacker attempting to obtain a certificate through fraud will choose to have a certificate issued with this extension. Such risks are more appropriately addressed by mechanisms such as Certification Authority Authorization DNS records RFC 6844 [!RFC6844] that are designed to prevent or mitigate mis-issue. Nevertheless a Certification Authority MAY consider the presence or absence of a required TLS feature as one factor in determining the level of additional scrutiny a request should be subject to.

A server offering an end entity certificate with a TLS feature extension MUST satisfy a client request for the specified feature unless this would be redundant as described below. Otherwise clients MAY refuse connection. It is important therefore that a Certification Authority only issue certificates that specify features that match the configuration of the server and that the server is capable of verifying that its configuration is compatible with the feature declaration of the certificates it offers. Ideally, the TLS feature declaration would be specified by the certificate request generator as part of the certificate issue process.

This document describes the use of the TLS feature in PKIX end entity and certificate signing certificate and a mechanism that MAY be used to describe support for the specified features in-band for the most commonly used certificate registration protocol.

Syntax

The TLS Feature extension has the following format:

```
tls-feature OBJECT IDENTIFIER ::=  { id-pe  24 }

Features ::= SEQUENCE OF INTEGER
```

The TLS Feature Extension SHOULD NOT be marked critical. RFC 5280 [!RFC5280] requires that implementations that do not understand critical extensions MUST reject the certificate. Marking the TLS

Feature Extension critical breaks backward compatibility and is not recommended unless this is the desired behavior.

TLS Feature

The object member Features is a sequence of TLS extension identifiers (features, in this specification's terminology). If these features are requested by the client in its ClientHello message, then they MUST be present in the server's ServerHello.

This specification does not require a TLS client to offer or support any TLS feature regardless of whether it is specified in the server certificate's TLS Feature extension or not. In particular a client MAY request and a server MAY support any TLS extension regardless of whether it is specified in a TLS Feature extension or not.

If a TLS Feature extension specifies a TLS feature, a server offering the certificate MUST support the extension specified and MUST comply with any specific requirements specified for that feature in this document or in the document that specifies the TLS feature.

Use

Certificate Signing Request

If the certificate issue mechanism makes use of the PKCS#10 Certificate Signing Request (CSR) [!RFC2986], the CSR MAY specify a TLS Feature extension as a CSR attribute. A server or server administration tool should only generate key signing requests that it knows can be supported by the server for which the certificate is intended.

Certificate Signing Certificate

When present in a Certificate Signing Certificate (i.e., CA certificate with the key usage extension value set to keyCertSign), the TLS Feature extension specifies a constraint on valid certificate chains. Specifically, a certificate that is signed by a Certificate Signing Certificate that contains a TLS Feature extension MUST contain a TLS Feature extension which MUST offer the same set or a superset of the features advertised in the signing certificate.

While relying parites (i.e., clients) MAY reject certificates that do not comply with this requirement, the use of TLS Feature extension in Certificate Signing Certificates is primarily intended for use by parties seeking to evaluate the performance of certificate issuers and MAY be ignored by clients.

End Entity Certificate

When specified in a server end entity Certificate (i.e. a certificate that specifies the id-kp-server EKU), the TLS Feature extension specifies criteria that a server MUST meet to be compliant with the feature declaration.

In the case that a client determines that the server configuration is inconsistent with the specified feature declaration it MAY reject the TLS configuration.

In the case that a client determines that the server configuration is inconsistent with a feature declaration specifying support for the TLS status_request extension it SHOULD reject the TLS configuration.

Processing

Certification Authority

A CA SHOULD NOT issue certs with a TLS Feature extension unless there is an affirmative statement to the effect that the end entity intends to support the specified features. For example the use of a Feature extension in the CSR or through an out of band communication.

Server

A TLS server certificate containing a TLS Feature extension MAY be used with any TLS server that supports the specified features. It is not necessary for the server to provide support for the TLS Feature extension itself. Such support is nevertheless desirable as it can reduce the risk of administrative error.

A server SHOULD verify that its configuration is compatible with the TLS Feature extension expressed in a certificate it presents. A server MAY override local configuration options if necessary to ensure consistency but SHOULD inform the administrator whenever such an inconsistency is discovered.

A server SHOULD support generation of the Feature extension in CSRs if key generation is supported.

Client

A client MUST treat a certificate with a TLS Feature extension if the features offered by the server do not contain all features present in both the client's ClientHello message and the TLS Feature extension

In the case that use of TLS with a valid certificate is mandated by explicit security policy, application protocol specification or other means, the client MUST refuse the connection. If the use of TLS with a valid certificate is optional, a client MAY accept the connection but MUST NOT treat the certificate as valid.

Acknowledgements

This proposal incorporates text and other contributions from participants in the IETF and CA-Browser forum. In particular, Robin Alden, Richard Barnes, Viktor Dukhovni, Stephen Farrell, Gervase Markham, Yoav Nir, Tom Ritter, Jeremy Rowley, Stefan Santesson, Ryan Sleevi, Brian Smith, Rob Stradling and Sean Turner

Security Considerations

Alternative Certificates and Certificate Issuers

Use of the TLS Feature extension to mandate support for a particular form of revocation checking is optional. This control can provide protection in the case that a certificate with a TLS Feature is compromised after issue but not in the case that the attacker obtains an unmarked certificate from an issuer through fraud.

The TLS Feature extension is a post-issue security control. Such risks can only be addressed by security controls that take effect before issue.

Denial of Service

A certificate Issuer could issue a certificate that intentionally specified a feature statement that they knew the server could not support.

The risks of such refusal would appear to be negligible since a Certification Authority could equally refuse to issue the certificate.

Cipher Suite Downgrade Attack

The TLS Feature extension does not provide protection against a cipher suite downgrade attack. This is left to the existing controls in the TLS protocol itself.

IANA Considerations

On approval, IANA shall add in the SMI Security for PKIX Certificate Extension (1.3.6.1.5.5.7.1) registry the following entry:

Decimal	Description	References
24	id-pe-tlsfeature	{this RFC}