Finesse-Fehler "SSLPeerUnverifiedException" für Gadgets, die auf Servern mit CA-Signatur gehostet werden

Inhalt

Einleitung

In diesem Dokument werden die Schritte zur Fehlerbehebung in einem Szenario beschrieben, in dem eine von der Zertifizierungsstelle (Certificate Authority, CA) signierte Zertifikatskette für einen externen Webserver, der ein Gadget hostet, auf Finesse hochgeladen wird, das Gadget jedoch nicht geladen wird, wenn Sie sich bei Finesse anmelden, und der Fehler "SSLPeerUnverifiedException" angezeigt wird.

Mit freundlicher Unterstützung von Gino Schweinsberger, Cisco TAC Engineer.

Voraussetzungen

Anforderungen

Cisco empfiehlt, dass Sie über Kenntnisse in folgenden Bereichen verfügen:

- SSL-Zertifikate
- Finesse-Verwaltung
- Windows Server-Administration
- Paketerfassungsanalyse mit Wireshark

Verwendete Komponenten

Die Informationen in diesem Dokument basieren auf folgenden Software-Versionen:

- Unified Contact Center Express (UCCX) 11.x
- Finesse 11.x

Die Informationen in diesem Dokument beziehen sich auf Geräte in einer speziell eingerichteten Testumgebung. Alle Geräte, die in diesem Dokument benutzt wurden, begannen mit einer gelöschten (Nichterfüllungs) Konfiguration. Wenn Ihr Netzwerk in Betrieb ist, stellen Sie sicher, dass Sie die möglichen Auswirkungen aller Befehle verstehen.

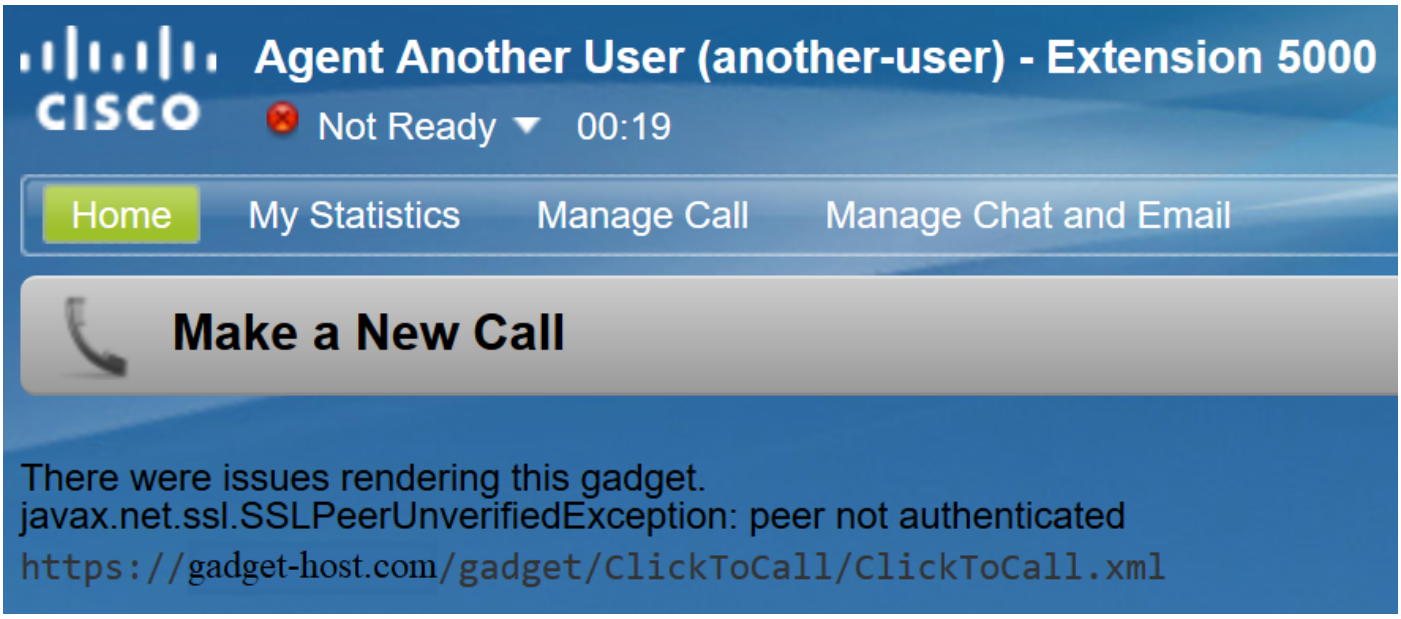
Hintergrundinformationen

Dies sind die Bedingungen für das Auftreten des Fehlers:

- Annahme: Die Zertifikatvertrauenskette wird nach Finesse hochgeladen
- Stellen Sie sicher, dass die richtigen Server/Dienste neu gestartet wurden.
- Angenommen, das Gadget wurde dem Finesse-Layout mit einer HTTPS-URL hinzugefügt und die URL ist erreichbar.

Dieser Fehler wurde bei der Anmeldung des Agenten bei Finesse festgestellt:

"Bei der Wiedergabe dieses Gadgets sind Probleme aufgetreten.
javax.net.ssl.SSLPeerUnverifiedException: Peer nicht authentifiziert"



Probleme

Szenario 1: Der Hosting-Server handelt unsichere TLS aus

Wenn Finesse Server eine Verbindungsanforderung an den Hosting-Server stellt, kündigt Finesse Tomcat eine Liste der unterstützten Verschlüsselungsschlüssel an.

Einige Chiffren werden aufgrund von Sicherheitslücken nicht unterstützt.

Wenn der Hostserver einen dieser Chiffren auswählt, wird die Verbindung abgelehnt:

- TLS_DHE_RSA_WITH_AES_256_CBC_SHA
- TLS_DHE_RSA_WITH_AES_128_CBC_SHA

Diese Chiffren nutzen bekanntermaßen schwache ephemere Diffie-Hellman-Schlüssel, wenn sie

die Verbindung aushandeln. Aufgrund der Logjam-Verwundbarkeit sind diese eine schlechte Wahl für TLS-Verbindungen.

Folgen Sie dem TLS-Handshake-Prozess bei der Paketerfassung, um festzustellen, welche Verschlüsselung ausgehandelt wird.

1. Finesse präsentiert seine Liste der unterstützten Chiffren im **Client Hello-**Schritt:

```
▼ TLSv1 Record Layer: Handshake Protocol: Client Hello
    Content Type: Handshake (22)
    Version: TLS 1.0 (0x0301)
    Length: 67
  ▼ Handshake Protocol: Client Hello
      Handshake Type: Client Hello (1)
      Length: 63
      Version: TLS 1.0 (0x0301)
    › Random: 5cacb293b5efdb4cf1bb34464d7de9f5060b00a9beb81d29…
      Session ID Length: 0
      Cipher Suites Length: 24
    ▼ Cipher Suites (12 suites)
        Cipher Suite: TLS_RSA_WITH_AES_256_CBC_SHA (0x0035)
        Cipher Suite: TLS_DHE_RSA_WITH_AES_256_CBC_SHA (0x0039)
        Cipher Suite: TLS_DHE_DSS_WITH_AES_256_CBC_SHA (0x0038)
        Cipher Suite: TLS_RSA_WITH_AES_128_CBC_SHA (0x002f)
        Cipher Suite: TLS_DHE_RSA_WITH_AES_128_CBC_SHA (0x0033)
        Cipher Suite: TLS_DHE_DSS_WITH_AES_128_CBC_SHA (0x0032)
        Cipher Suite: TLS_RSA_WITH_RC4_128_SHA (0x0005)
        Cipher Suite: TLS_RSA_WITH_3DES_EDE_CBC_SHA (0x000a)
        Cipher Suite: TLS_DHE_RSA_WITH_3DES_EDE_CBC_SHA (0x0016)
        Cipher Suite: TLS_DHE_DSS_WITH_3DES_EDE_CBC_SHA (0x0013)
        Cipher Suite: TLS_RSA_WITH_RC4_128_MD5 (0x0004)
        Cipher Suite: TLS_EMPTY_RENEGOTIATION_INFO_SCSV (0x00ff)
      Compression Methods Length: 1
    › Compression Methods (1 method)
```

2. Für diese Verbindung wurde **TLS_DHE_RSA_WITH_AES_256_CBC_SHA** vom Hosting-Server während des **Server-Hello-**Schritts ausgewählt, da dieser höher in der Liste der bevorzugten Chiffren steht.

```
TLSv1 Record Layer: Handshake Protocol: Multiple Handshake Messages
  Content Type: Handshake (22)
  Version: TLS 1.0 (0x0301)
  Length: 2557
  Handshake Protocol: Server Hello
    Handshake Type: Server Hello (2)
    Length: 77
    Version: TLS 1.0 (0x0301)
    Random: 5cacb292c4d7183627f620a066f9b6ce6460dcb849b59cae…
    Session ID Length: 32
    Session ID: 4c290000ce66098cc994a33e193b0da1244cb9f083f69c26…
    Cipher Suite: TLS_DHE_RSA_WITH_AES_256_CBC_SHA (0x0039)
    Compression Method: null (0)
    Extensions Length: 5
    Extension: renegotiation_info (len=1)
  Handshake Protocol: Certificate
  Handshake Protocol: Server Key Exchange
    Handshake Type: Server Key Exchange (12)
    Length: 1032
    Diffie-Hellman Server Params
  Handshake Protocol: Server Hello Done
    Handshake Type: Server Hello Done (14)
    Length: 0
```

3. Finesse sendet eine schwerwiegende Warnung und beendet die Verbindung:

```
TLSv1 Record Layer: Alert (Level: Fatal, Description: Internal Error)
  Content Type: Alert (21)
  Version: TLS 1.0 (0x0301)
  Length: 2
  Alert Message
```

Lösung

Um die Verwendung dieser Chiffren zu verhindern, muss der Hosting-Server so konfiguriert werden, dass diese eine niedrige Priorität erhalten, oder sie müssen vollständig aus der Liste der verfügbaren Chiffren entfernt werden. Dies kann auf einem Windows-Server mit dem Windows-Gruppenrichtlinien-Editor (gpedit.msc) erfolgen.

> **Hinweis**: Weitere Informationen über die Auswirkungen von Logjam in Finesse und die Verwendung von gpedit finden Sie unter:

Szenario 2: Das Zertifikat verfügt über einen nicht unterstützten Signaturalgorithmus

Windows Server-Zertifizierungsstellen können neuere Signaturstandards verwenden, um Zertifikate zu signieren. Selbst wenn dadurch die Sicherheit erhöht wird als bei SHA, werden diese Standards nur in geringem Umfang außerhalb von Microsoft-Produkten übernommen, und Administratoren stoßen wahrscheinlich auf Interoperabilitätsprobleme.

Finesse Tomcat vertraut auf den Sicherheitsanbieter SunMSCAPI von Java, um die Unterstützung für die verschiedenen von Microsoft verwendeten Signaturalgorithmen und Verschlüsselungsfunktionen zu ermöglichen. Alle aktuellen Versionen von Java (1.7, 1.8 und 1.9) unterstützen nur die folgenden Signaturalgorithmen:

- MD5 mit RSA
- MD2mitRSA
- KEINE mitRSA
- SHA1 mitRSA
- SHA256mit RSA
- SHA384mit RSA
- SHA512mit RSA

Es empfiehlt sich, die Java-Version auf dem Finesse-Server zu überprüfen, um festzustellen, welche Algorithmen in dieser Version unterstützt werden. Die Version kann mithilfe des folgenden Befehls vom Root-Zugriff aus überprüft werden: **java -version**

```
Using username "root".
Last login: Tue Apr 16 13:11:00 2019 from
[root@uccx12pub ~]# java -version
java version "1.7.0_181"
OpenJDK Runtime Environment (rhel-2.6.14.8.el6_9-i386 u181-b00)
OpenJDK Server VM (build 24.181-b00, mixed mode)
[root@uccx12pub ~]#
```

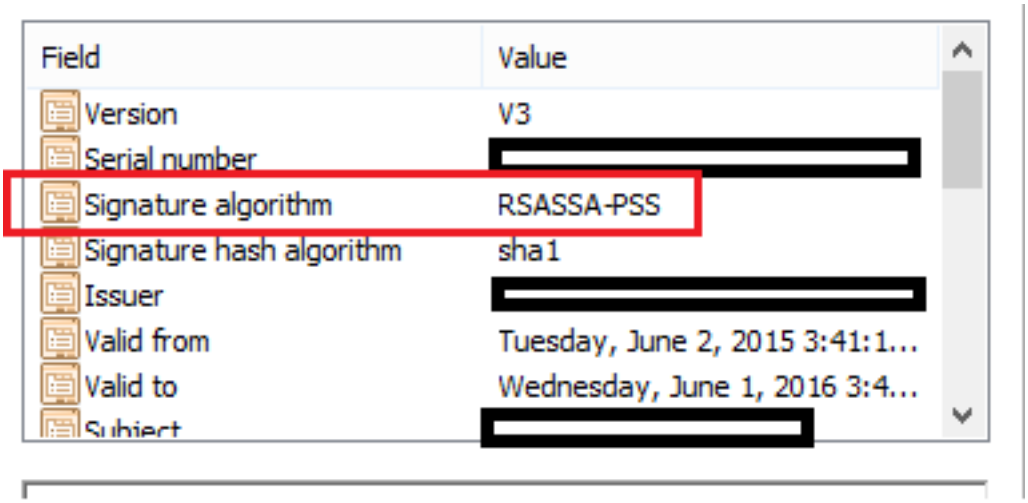
> **Hinweis**: Weitere Informationen zum Java SunMSCAPI-Anbieter finden Sie unter https://docs.oracle.com/javase/8/docs/technotes/guides/security/SunProviders.html#SunMSCAPI

Wenn ein Zertifikat mit einer anderen als den oben aufgeführten Signaturen bereitgestellt wird, kann Finesse das Zertifikat nicht zum Herstellen einer TLS-Verbindung mit dem Hosting-Server verwenden. Dazu gehören Zertifikate, die mit einem unterstützten Signaturtyp signiert sind, aber von Zertifizierungsstellen ausgestellt wurden, deren eigene Zwischen- und Stammzertifikate mit anderen Signaturen signiert sind.

Wenn Sie sich eine Paketerfassung ansehen, schließt Finesse die Verbindung mit einer "schwerwiegenden Warnung: Certificate Unknown"-Fehler, wie im Bild gezeigt.

```
Secure Sockets Layer
⊟ TLSv1.2 Record Layer: Alert (Level: Fatal, Description: Certificate Unknown)
    Content Type: Alert (21)
    Version: TLS 1.2 (0x0303)
    Length: 2
  ⊟ Alert Message
      Level: Fatal (2)
      Description: Certificate Unknown (46)
```

An dieser Stelle ist erforderlich, um die vom Hosting-Server vorgelegten Zertifikate zu überprüfen und nach nicht unterstützten Signaturalgorithmen zu suchen. Es ist üblich, **RSASSA-PSS** als den problematischen Signaturalgorithmus zu betrachten:



Wenn ein Zertifikat in der Kette mit RSASSA-PSS signiert ist, schlägt die Verbindung fehl. In diesem Fall zeigt die Paketerfassung, dass die Stammzertifizierungsstelle RSASSA-PSS für ihr eigenes Zertifikat verwendet:

```
Certificates (3906 bytes)
    Certificate Length: 1728
  Certificate: 308206bc308205a4a00302010202137400000243b805da9... (id-at-commonName=[redacted]
    signedCertificate
    algorithmIdentifier (sha256WithRSAEncryption)
      Padding: 0
      encrypted: e6230df257be9d34c0f57bc2f88c081c4186aaad092c8155...
    Certificate Length: 1114
  Certificate: 308204563082033ea00302010202131600000000a93cd17d6... (id-at-commonName=[redacted]Issuing Authority[redacted]
    signedCertificate
    algorithmIdentifier (sha256WithRSAEncryption)
      Padding: 0
      encrypted: 889be6a1125c758cd0009b392d3b90a69b64546dcee09c84...
    Certificate Length: 1055
  Certificate: 3082041b308202cfa00302010202107b70dbb7c2760da74f... (id-at-commonName=[redacted]Root CA [redacted]
    signedCertificate
    algorithmIdentifier (id-RSASSA-PSS)
      Algorithm Id: 1.2.840.113549.1.1.10 (id-RSASSA-PSS)
    RSASSA-PSS-params
      Padding: 0
      encrypted: d8e9151adc76b4e55f9277fce916613ce26199e3b50dcb54...
```

Lösung

Um dieses Problem zu beheben, muss ein neues Zertifikat von einem Zertifizierungsstellenanbieter ausgestellt werden, der nur einen der unterstützten SunMSCAPI-Signaturtypen verwendet, die wie oben erläutert in der gesamten Zertifikatskette aufgeführt sind.

> **Hinweis**: Weitere Informationen zum RSASSA-PSS-Signaturalgorithmus finden Sie unter https://pkisolutions.com/pkcs1v2-1rsassa-pss/

> **Anmerkung:** Dieses Problem wird im Defekt CSCve79330 verfolgt

Informationen zu dieser Übersetzung

Cisco hat dieses Dokument maschinell übersetzen und von einem menschlichen Übersetzer editieren und korrigieren lassen, um unseren Benutzern auf der ganzen Welt Support-Inhalte in ihrer eigenen Sprache zu bieten. Bitte beachten Sie, dass selbst die beste maschinelle Übersetzung nicht so genau ist wie eine von einem professionellen Übersetzer angefertigte. Cisco Systems, Inc. übernimmt keine Haftung für die Richtigkeit dieser Übersetzungen und empfiehlt, immer das englische Originaldokument (siehe bereitgestellter Link) heranzuziehen.