

Solucionar problemas de erro do CommPilot "SSL_ERROR_NO_CIPHER_OVERLAP"

Contents

[Introduction](#)

[Prerequisites](#)

[Requirements](#)

[Informações de Apoio](#)

[Configuração da BroadWorks](#)

[Exemplo de laboratório funcional](#)

[Configuração](#)

[Verificação](#)

[Auditoria de conectividade](#)

[Exemplo de laboratório com erro](#)

[Problema](#)

[Configuração](#)

[Verificação](#)

[Auditoria de conectividade](#)

[Resolução](#)

[Verificação de resolução](#)

Introduction

Este documento descreve como configurar e solucionar problemas do BroadWorks para evitar o erro "SSL_ERROR_NO_CIPHER_OVERLAP".

Prerequisites

Requirements

A Cisco recomenda que você tenha conhecimento da plataforma BroadWorks.

Informações de Apoio

Configuração da BroadWorks

Para as versões 22 e posteriores do Broadworks, os protocolos e as cifras são configuráveis via CLI através dos contextos vistos em diferentes níveis de configuração.

```
'Interface/Port specific - low level'
```

```
CLI/Interface/Http/HttpServer/SSLSettings/Protocols
```

```
CLI/Interface/Http/HttpServer/SSLSettings/Ciphers
```

'All interfaces - mid level'

CLI/Interface/Http/SSLCommonSettings/Protocols

CLI/Interface/Http/SSLCommonSettings/Ciphers

'Generic system level - high level'

CLI/System/SSLCommonSettings/JSSE/Protocols

CLI/System/SSLCommonSettings/JSSE/Ciphers

Um contexto chamado SSLCommonSettings se refere a um item menos específico da hierarquia SSL e um contexto chamado SSLSettings se refere a um item mais específico da hierarquia.

Exemplo de laboratório funcional

Configuração

Configuração de baixo nível vinculada à interface e porta específicas sem cifras definidas:

```
CLI/Interface/Http/HttpServer/SSLSettings/Protocols> get 172.16.30.146 443
```

```
Protocol Name
```

```
=====
```

```
TLSv1.1
```

```
TLSv1.2
```

```
TLSv1
```

```
CLI/Interface/Http/HttpServer/SSLSettings/Ciphers> get 172.16.30.146 443
```

```
Cipher Name
```

```
=====
```

```
0 entry found.
```

Verificação

Verifique a configuração com o comando curl comando:

```
$ curl -v -k https://172.16.30.146
* About to connect() to 172.16.30.146 port 443 (#0)
* Trying 172.16.30.146...
* Connected to 172.16.30.146 (172.16.30.146) port 443 (#0)
* Initializing NSS with certpath: sql:/etc/pki/nssdb
* skipping SSL peer certificate verification
* SSL connection using TLS_RSA_WITH_AES_256_CBC_SHA256 <-----
* Server certificate:
* subject:
E=broadworks_tac@cisco.com,CN=*.calo.cisco.com,OU=BroadworksTAC,O=TestIssuer,ST=Veracruz,C=MX
* start date: Apr 04 20:39:56 2022 GMT
* expire date: Apr 04 20:39:56 2023 GMT
* common name: *.calo.cisco.com
* issuer: CN=Root CA test,OU=BroadworksTAC,O=TestIssuer,L=Tecolutla,ST=Veracruz,C=MX
>GET / HTTP/1.1
>User-Agent: curl/7.29.0
>Host: 172.16.30.146
>Accept: */*
>
<HTTP/1.1 302 Found
```

Aqui ele se conectou com êxito via TLSv1.2 com a codificação TLS_RSA_WITH_AES_256_CBC_SHA256.

Auditoria de conectividade

Para verificar os protocolos e as cifras que são aceitos:

```
$ nmap -sV --script ssl-enum-ciphers -p 443 172.16.30.146

Starting Nmap 6.40 ( http://nmap.org ) at 2022-05-09 04:26 EDT
Nmap scan report for r23xsp01.calo.cisco.com (172.16.30.146)
Host is up (0.00013s latency).
PORT STATE SERVICE VERSION
443/tcp open ssl/https?
| ssl-enum-ciphers:
| TLSv1.0:
| ciphers:
| TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA - strong
| TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA - strong
| TLS_ECDHE_RSA_WITH_RC4_128_SHA - strong
| TLS_RSA_WITH_AES_128_CBC_SHA - strong
| TLS_RSA_WITH_AES_256_CBC_SHA - strong
| TLS_RSA_WITH_RC4_128_SHA - strong
| compressors:
| NULL
| TLSv1.1:
| ciphers:
| TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA - strong
| TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA - strong
| TLS_ECDHE_RSA_WITH_RC4_128_SHA - strong
| TLS_RSA_WITH_AES_128_CBC_SHA - strong
| TLS_RSA_WITH_AES_256_CBC_SHA - strong
| TLS_RSA_WITH_RC4_128_SHA - strong
| compressors:
| NULL
| TLSv1.2:
| ciphers:
| TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA - strong
| TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256 - strong
| TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA - strong
| TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384 - strong
| TLS_ECDHE_RSA_WITH_RC4_128_SHA - strong
| TLS_RSA_WITH_AES_128_CBC_SHA - strong
| TLS_RSA_WITH_AES_128_CBC_SHA256 - strong
| TLS_RSA_WITH_AES_256_CBC_SHA - strong
| TLS_RSA_WITH_AES_256_CBC_SHA256 - strong
| TLS_RSA_WITH_RC4_128_SHA - strong
| compressors:
| NULL
|_ least strength: strong
```

Exemplo de laboratório com erro

Problema

Erro observado - "SSL_ERROR_NO_CIPHER_OVERLAP" através do navegador.

```
# curl -v https://172.16.30.146
* About to connect() to 172.16.30.146 port 443 (#0)
* Trying 172.16.30.146...
* Connected to 172.16.30.146 (172.16.30.146) port 443 (#0)
```

```
* Initializing NSS with certpath: sql:/etc/pki/nssdb * CAfile: /etc/pki/tls/certs/ca-bundle.crt
CApath: none
* NSS error -12286 (SSL_ERROR_NO_CYPHER_OVERLAP)
* Cannot communicate securely with peer: no common encryption algorithm(s).
* Closing connection 0 curl: (35) Cannot communicate securely with peer: no common encryption
algorithm(s).
```

Configuração

Configuração de baixo nível vinculada à interface e porta específicas com o protocolo TLSv1.2 definido com a codificação TLSv1.0 TLS_ECDH_RSA_WITH_AES_128_GCM_SHA256 definida:

```
CLI/Interface/Http/HttpServer/SSLSettings/Protocols> get 172.16.30.146 443
```

```
Protocol Name
=====
TLSv1.2
```

```
CLI/Interface/Http/SSLCommonSettings/Ciphers> get
```

```
Cipher Name
=====
TLS_ECDH_RSA_WITH_AES_128_GCM_SHA256
```

Verificação

Verifique a configuração com o comando `curl` comando:

```
$ curl -v -k https://172.16.30.146
* About to connect() to 172.16.30.146 port 443 (#0)
* Trying 172.16.30.146...
* Connected to 172.16.30.146 (172.16.30.146) port 443 (#0)
* Initializing NSS with certpath: sql:/etc/pki/nssdb
* NSS error -12286 (SSL_ERROR_NO_CYPHER_OVERLAP)
* Cannot communicate securely with peer: no common encryption algorithm(s).
* Closing connection 0
curl: (35) Cannot communicate securely with peer: no common encryption algorithm(s).
```

Auditoria de conectividade

Para verificar os protocolos e as cifras que são aceitos:

```
$ nmap -sV --script ssl-enum-ciphers -p 443 172.16.30.146
```

```
Starting Nmap 6.40 ( http://nmap.org ) at 2022-05-09 05:31 EDT
Nmap scan report for r23xsp01.calo.cisco.com (172.16.30.146)
Host is up (0.000049s latency).
PORT STATE SERVICE VERSION
443/tcp open  https?
| ssl-enum-ciphers:
|_ TLSv1.2: No supported ciphers found
```

A partir dos resultados da ferramenta, observa-se que o protocolo TLSv1.2 está disponível, mas não há cifras suportadas.

Resolução

Exclua a cifra TLSv1.1 em `CLI/Interface/Http/SSLCommonSettings/Ciphers` e, em seguida, abra todas as cifras TLSv1.2 novamente (ou adicione uma cifra TLSv1.2).

```
CLI/Interface/Http/HttpServer/SSLSettings/Protocols> get 172.16.30.146 443
Protocol Name
=====
TLsv1.2
```

```
CLI/Interface/Http/HttpServer/SSLSettings/Ciphers> get 172.16.30.146 443
Cipher Name
=====
0 entry found.
```

```
CLI/Interface/Http/SSLCommonSettings/Ciphers> get
Cipher Name
=====
0 entry found.
```

Verificação de resolução

```
$ curl -v -k https://172.16.30.146
* About to connect() to 172.16.30.146 port 443 (#0)
* Trying 172.16.30.146...
* Connected to 172.16.30.146 (172.16.30.146) port 443 (#0)
* Initializing NSS with certpath: sql:/etc/pki/nssdb
* skipping SSL peer certificate verification
* SSL connection using TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256 <-----
* Server certificate:
* subject:
E=broadworks_tac@cisco.com,CN=*.calo.cisco.com,OU=BroadworksTAC,O=TestIssuer,ST=Veracruz,C=MX
* start date: Apr 04 20:39:56 2022 GMT
* expire date: Apr 04 20:39:56 2023 GMT
* common name: *.calo.cisco.com
* issuer: CN=Root CA test,OU=BroadworksTAC,O=TestIssuer,L=Tecolutla,ST=Veracruz,C=MX
>GET / HTTP/1.1
>User-Agent: curl/7.29.0
>Host: 172.16.30.146
>Accept: */*
>
<HTTP/1.1 302 Found
```

```
$ nmap -sV --script ssl-enum-ciphers -p 443 172.16.30.146
```

```
Starting Nmap 6.40 ( http://nmap.org ) at 2022-05-09 05:44 EDT
Nmap scan report for r23xsp01.calo.cisco.com (172.16.30.146)
Host is up (0.000063s latency).
PORT STATE SERVICE VERSION
443/tcp open https?
| ssl-enum-ciphers:
| TLSv1.2:
| ciphers:
| TLS_DHE_RSA_WITH_AES_128_CBC_SHA256 - strong
| TLS_DHE_RSA_WITH_AES_128_GCM_SHA256 - strong
| TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256 - strong
| TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256 - strong
```

Sobre esta tradução

A Cisco traduziu este documento com a ajuda de tecnologias de tradução automática e humana para oferecer conteúdo de suporte aos seus usuários no seu próprio idioma, independentemente da localização.

Observe que mesmo a melhor tradução automática não será tão precisa quanto as realizadas por um tradutor profissional.

A Cisco Systems, Inc. não se responsabiliza pela precisão destas traduções e recomenda que o documento original em inglês ([link fornecido](#)) seja sempre consultado.