目 录

1 SSL

说明

设备运行于 FIPS 模式时，本特性部分配置相对于非 FIPS 模式有所变化，具体差异请见本文相关描述。有关 FIPS 模式的详细介绍请参见“安全配置指导”中的“FIPS”。

1.1 SSL配置命令

1.1.1 ciphersuite

ciphersuite 命令用来配置 SSL 服务器端策略支持的加密套件。

undo ciphersuite 命令用来恢复缺省情况。

【命令】

非 FIPS 模式下：

ciphersuite { dhe_rsa_aes_128_cbc_sha | dhe_rsa_aes_128_cbc_sha256 | dhe_rsa_aes_256_cbc_sha | dhe_rsa_aes_256_cbc_sha256 | ecdhe_ecdsa_aes_128_cbc_sha256 | ecdhe_ecdsa_aes_128_gcm_sha256 | ecdhe_ecdsa_aes_256_cbc_sha384 | ecdhe_ecdsa_aes_256_gcm_sha384 | ecdhe_rsa_aes_128_cbc_sha256 | ecdhe_rsa_aes_128_gcm_sha256 | ecdhe_rsa_aes_256_cbc_sha384 | ecdhe_rsa_aes_256_gcm_sha384 | exp_rsa_des_cbc_sha | exp_rsa_rc2_md5 | exp_rsa_rc4_md5 | rsa_3des_ede_cbc_sha | rsa_aes_128_cbc_sha | rsa_aes_128_cbc_sha256 | rsa_aes_256_cbc_sha | rsa_aes_256_cbc_sha256 | rsa_des_cbc_sha | rsa_rc4_128_md5 | rsa_rc4_128_sha } *

undo ciphersuite

FIPS 模式下：

ciphersuite { ecdhe_ecdsa_aes_128_cbc_sha256 | ecdhe_ecdsa_aes_128_gcm_sha256 | ecdhe_ecdsa_aes_256_cbc_sha384 | ecdhe_ecdsa_aes_256_gcm_sha384 | ecdhe_rsa_aes_128_cbc_sha256 | ecdhe_rsa_aes_128_gcm_sha256 | rsa_aes_128_cbc_sha | rsa_aes_128_cbc_sha256 | rsa_aes_256_cbc_sha | rsa_aes_256_cbc_sha256 } *

undo ciphersuite

【缺省情况】

SSL 服务器端策略支持所有的加密套件。

【视图】

SSL 服务器端策略视图

【缺省用户角色】

network-admin

【参数】

dhe_rsa_aes_128_cbc_sha：密钥交换算法采用 DHE RSA、数据加密算法采用 128 位的 AES、MAC 算法采用 SHA。

dhe_rsa_aes_128_cbc_sha256：密钥交换算法采用 DHE RSA、数据加密算法采用 128 位的 AES_CBC、MAC 算法采用 SHA256。

dhe_rsa_aes_256_cbc_sha：密钥交换算法采用 DHE RSA、数据加密算法采用 256 位的 AES、MAC 算法采用 SHA。

dhe_rsa_aes_256_cbc_sha256：密钥交换算法采用 DHE RSA、数据加密算法采用 256 位的 AES_CBC、MAC 算法采用 SHA256。

ecdhe_ecdsa_aes_128_cbc_sha256：密钥交换算法采用 ECDHE ECDSA、数据加密算法采用 128 位的 AES_CBC、MAC 算法采用 SHA256。

ecdhe_ecdsa_aes_128_gcm_sha256：密钥交换算法采用 ECDHE ECDSA、数据加密算法采用 128 位的 AES_GCM、MAC 算法采用 SHA256。

ecdhe_ecdsa_aes_256_cbc_sha384：密钥交换算法采用 ECDHE ECDSA、数据加密算法采用 256 位的 AES_CBC、MAC 算法采用 SHA384。

ecdhe_ecdsa_aes_256_gcm_sha384：密钥交换算法采用 ECDHE ECDSA、数据加密算法采用 256 位的 AES_GCM、MAC 算法采用 SHA384。

ecdhe_rsa_aes_128_cbc_sha256：密钥交换算法采用 ECDHE RSA、数据加密算法采用 128 位的 AES_CBC、MAC 算法采用 SHA256。

ecdhe_rsa_aes_128_gcm_sha256：密钥交换算法采用 ECDHE RSA、数据加密算法采用 128 位的 AES_GCM、MAC 算法采用 SHA256。

ecdhe_rsa_aes_256_cbc_sha384：密钥交换算法采用 ECDHE RSA、数据加密算法采用 256 位的 AES_CBC、MAC 算法采用 SHA384。

ecdhe_rsa_aes_256_gcm_sha384：密钥交换算法采用 ECDHE RSA、数据加密算法采用 256 位的 AES_GCM、MAC 算法采用 SHA384。

exp_rsa_des_cbc_sha：满足出口限制的算法套件。密钥交换算法采用 RSA、数据加密算法采用 DES_CBC、MAC 算法采用 SHA。

exp_rsa_rc2_md5：满足出口限制的算法套件。密钥交换算法采用 RSA、数据加密算法采用 RC2、MAC 算法采用 MD5。

exp_rsa_rc4_md5：满足出口限制的算法套件。密钥交换算法采用 RSA、数据加密算法采用 RC4、MAC 算法采用 MD5。

rsa_3des_ede_cbc_sha：密钥交换算法采用 RSA、数据加密算法采用 3DES_EDE_CBC、MAC 算法采用 SHA。

rsa_aes_128_cbc_sha：密钥交换算法采用 RSA、数据加密算法采用 128 位 AES_CBC、MAC 算法采用 SHA。

rsa_aes_128_cbc_sha256：密钥交换算法采用 RSA、数据加密算法采用 128 位的 AES_CBC、MAC 算法采用 SHA256。

rsa_aes_256_cbc_sha：密钥交换算法采用 RSA、数据加密算法采用 256 位 AES_CBC、MAC 算法采用 SHA。

rsa_aes_256_cbc_sha256：密钥交换算法采用 RSA、数据加密算法采用 256 位的 AES_CBC、MAC 算法采用 SHA256。

rsa_des_cbc_sha：密钥交换算法采用 RSA、数据加密算法采用 DES_CBC、MAC 算法采用 SHA。

rsa_rc4_128_md5：密钥交换算法采用 RSA、数据加密算法采用 128 位的 RC4、MAC 算法采用 MD5。

rsa_rc4_128_sha：密钥交换算法采用 RSA、数据加密算法采用 128 位的 RC4、MAC 算法采用 SHA。

【使用指导】

为了提高安全性，SSL 协议采用了如下算法：

- 数据加密算法：用来对传输的数据进行加密，以保证数据传输的私密性。常用的数据加密算法通常为对称密钥算法，如 DES_CBC、3DES_EDE_CBC、AES_CBC、RC4 等。使用对称密钥算法时，要求 SSL 服务器端和 SSL 客户端具有相同的密钥。
- MAC（Message Authentication Code，消息验证码）算法：用来计算数据的 MAC 值，以防止发送的数据被篡改。常用的 MAC 算法有 MD5、SHA 等。使用 MAC 算法时，要求 SSL 服务器端和 SSL 客户端具有相同的密钥。
- 密钥交换算法：用来实现密钥交换，以保证对称密钥算法、MAC 算法中使用的密钥在 SSL 服务器端和 SSL 客户端之间安全地传递。常用的密钥交换算法通常为非对称密钥算法，如 RSA。

通过本命令可以配置 SSL 服务器端策略支持的各种算法组合。例如，**rsa_des_cbc_sha** 表示 SSL 服务器端策略支持的密钥交换算法为 RSA、数据加密算法为 DES_CBC、MAC 算法为 SHA。

SSL 服务器接收到 SSL 客户端发送的客户端加密套件后，将服务器支持的加密套件与 SSL 客户端支持的加密套件比较。如果 SSL 服务器支持的加密套件中存在 SSL 客户端支持的加密套件，则加密套件协商成功；否则，加密套件协商失败。

需要注意的是，如果多次执行本命令，则新的配置覆盖原有配置。

【举例】

指定 SSL 服务器端策略支持如下加密套件：

- 密钥交换算法为 DHE RSA、数据加密算法为 128 位的 AES、MAC 算法为 SHA
- 密钥交换算法为 RSA、数据加密算法为 128 位的 AES、MAC 算法为 SHA

```
<Sysname> system-view
[Sysname] ssl server-policy policy1
[Sysname-ssl-server-policy-policy1] ciphersuite dhe_rsa_aes_128_cbc_sha
rsa_aes_128_cbc_sha
```

【相关命令】

- **display ssl server-policy**
- **prefer-cipher**

1.1.2 client-verify enable

client-verify enable 命令用来配置 SSL 服务器端要求对 SSL 客户端进行基于数字证书的身份验证。**undo client-verify enable** 命令用来恢复缺省情况。

【命令】

client-verify enable

undo client-verify enable

【缺省情况】

SSL 服务器端不要求对 SSL 客户端进行基于数字证书的身份验证。

【视图】

SSL 服务器端策略视图

【缺省用户角色】

network-admin

【使用指导】

SSL 通过数字证书实现对对端的身份进行验证。数字证书的详细介绍，请参见“安全配置指导”中的“PKI”。

如果执行了 **client-verify enable** 命令，则 SSL 客户端必须将自己的数字证书提供给服务器，以便服务器对客户端进行基于数字证书的身份验证。只有身份验证通过后，SSL 客户端才能访问 SSL 服务器。

【举例】

配置 SSL 服务器端要求对 SSL 客户端进行基于数字证书的身份验证。

```
<Sysname> system-view
[Sysname] ssl server-policy policy1
[Sysname-ssl-server-policy-policy1] client-verify enable
```

【相关命令】

- **display ssl server-policy**

1.1.3 display crypto version

display crypto version 命令用来显示算法库的版本号。

【命令】

display crypto version

【视图】

任意视图

【缺省用户角色】

network-admin

network-operator

【使用指导】

相同的算法库版本号表示了一套相同的密码学算法。

【举例】

显示当前设备算法库的版本号。

```
<Sysname> display crypto version
```

```
7.1.3290
```

表1-1 display crypto version 命令显示信息描述表

字段	描述
7.1.3290	版本号信息，格式为7.1.X，其中7.1表示Comware V700R001，X表示算法库的版本号。

1.1.4 display ssl client-policy

display ssl client-policy 命令用来显示 SSL 客户端策略的信息。

【命令】

display ssl client-policy [*policy-name*]

【视图】

任意视图

【缺省用户角色】

network-admin

network-operator

【参数】

policy-name：显示指定的 SSL 客户端策略的信息，为 1～31 个字符的字符串，不区分大小写。如果不指定本参数，则显示所有 SSL 客户端策略的信息。

【举例】

显示名为 policy1 的 SSL 客户端策略的信息。

```
<Sysname> display ssl client-policy policy1
 SSL client policy: policy1
     SSL version: SSL 3.0
     PKI domain: client-domain
     Preferred ciphersuite:
         RSA_AES_128_CBC_SHA
     Server-verify: enabled
```

表1-2 display ssl client-policy 命令显示信息描述表

字段	描述
SSL client policy	SSL客户端策略名
SSL version	SSL客户端策略使用的SSL协议版本
PKI domain	SSL客户端策略使用的PKI域
Preferred ciphersuite	SSL客户端策略支持的加密套件
Server-verify	SSL客户端策略的服务器端验证模式，取值包括： • disabled：不要求对 SSL 服务器进行基于数字证书的身份验证 • enabled：要求对 SSL 服务器进行基于数字证书的身份验证

1.1.5 display ssl server-policy

display ssl server-policy 命令用来显示 SSL 服务器端策略的信息。

【命令】

display ssl server-policy [*policy-name*]

【视图】

任意视图

【缺省用户角色】

network-admin

network-operator

【参数】

policy-name：显示指定的 SSL 服务器端策略的信息，为 1～31 个字符的字符串，不区分大小写。如果不指定本参数，则显示所有 SSL 服务器端策略的信息。

【举例】

显示名为 policy1 的 SSL 服务器端策略的信息。

```
<Sysname> display ssl server-policy policy1
 SSL server policy: policy1
     PKI domain: server-domain
     Ciphersuites:
         DHE_RSA_AES_128_CBC_SHA
         RSA_AES_128_CBC_SHA
     Session cache size: 600
     Client-verify: enabled
```

表1-3 display ssl server-policy 命令显示信息描述表

字段	描述
SSL server policy	SSL服务器端策略名
PKI domain	SSL服务器端策略使用的PKI域
Ciphersuites	SSL服务器端策略支持的加密套件
Session cache size	SSL服务器端可以缓存的最大会话数目
Client-verify	SSL服务器端策略的客户端验证模式，取值包括： • disabled：不要求对客户端进行基于数字证书的身份验证 • enabled：要求对客户端进行基于数字证书的身份验证

1.1.6 pki-domain (SSL client policy view /SSL server policy view)

pki-domain 命令用来配置 SSL 客户端策略或 SSL 服务器端策略所使用的 PKI 域。

undo pki-domain 命令用来恢复缺省情况。

【命令】

pki-domain *domain-name*

undo pki-domain

【缺省情况】

没有指定 SSL 客户端策略或 SSL 服务器端策略所使用的 PKI 域。

【视图】

SSL 客户端策略视图/SSL 服务器端策略视图

【缺省用户角色】

network-admin

【参数】

domain-name：PKI 域的域名，为 1～31 个字符的字符串，不区分大小写。

【使用指导】

如果通过本命令指定了 SSL 客户端策略使用的 PKI 域，则引用该客户端策略的 SSL 客户端将通过该 PKI 域获取客户端的数字证书。

如果通过本命令指定了 SSL 服务器端策略使用的 PKI 域，则引用该服务器端策略的 SSL 服务器将通过该 PKI 域获取服务器端的数字证书。

【举例】

配置 SSL 客户端策略所使用的 PKI 域为 client-domain。

```
<Sysname> system-view
[Sysname] ssl client-policy policy1
[Sysname-ssl-client-policy-policy1] pki-domain client-domain
```

配置 SSL 服务器端策略所使用的 PKI 域为 server-domain。

```
<Sysname> system-view
[Sysname] ssl server-policy policy1
[Sysname-ssl-server-policy-policy1] pki-domain server-domain
```

【相关命令】

- **display ssl client-policy**
- **display ssl server-policy**
- **pki domain**（安全命令参考/PKI）

1.1.7 prefer-cipher

prefer-cipher 命令用来配置 SSL 客户端策略支持的加密套件。

undo prefer-cipher 命令用来恢复缺省情况。

【命令】

非 FIPS 模式下：

prefer-cipher { dhe_rsa_aes_128_cbc_sha | dhe_rsa_aes_128_cbc_sha256 | dhe_rsa_aes_256_cbc_sha | dhe_rsa_aes_256_cbc_sha256 | ecdhe_ecdsa_aes_128_cbc_sha256 | ecdhe_ecdsa_aes_128_gcm_sha256 |

ecdhe_ecdsa_aes_256_cbc_sha384 | **ecdhe_ecdsa_aes_256_gcm_sha384** | **ecdhe_rsa_aes_128_cbc_sha256** | **ecdhe_rsa_aes_128_gcm_sha256** | **ecdhe_rsa_aes_256_cbc_sha384** | **ecdhe_rsa_aes_256_gcm_sha384** | **exp_rsa_des_cbc_sha** | **exp_rsa_rc2_md5** | **exp_rsa_rc4_md5** | **rsa_3des_ede_cbc_sha** | **rsa_aes_128_cbc_sha** | **rsa_aes_128_cbc_sha256** | **rsa_aes_256_cbc_sha** | **rsa_aes_256_cbc_sha256** | **rsa_des_cbc_sha** | **rsa_rc4_128_md5** | **rsa_rc4_128_sha** }

undo prefer-cipher

FIPS 模式下：

prefer-cipher { **ecdhe_ecdsa_aes_128_cbc_sha256** | **ecdhe_ecdsa_aes_128_gcm_sha256** | **ecdhe_ecdsa_aes_256_cbc_sha384** | **ecdhe_ecdsa_aes_256_gcm_sha384** | **ecdhe_rsa_aes_128_cbc_sha256** | **ecdhe_rsa_aes_128_gcm_sha256** | **ecdhe_rsa_aes_256_cbc_sha384** | **ecdhe_rsa_aes_256_gcm_sha384** | **rsa_aes_128_cbc_sha** | **rsa_aes_128_cbc_sha256** | **rsa_aes_256_cbc_sha** | **rsa_aes_256_cbc_sha256** }

undo prefer-cipher

【缺省情况】

非 FIPS 模式下：

SSL 客户端策略支持的加密套件为 **rsa_rc4_128_md5**。

FIPS 模式下：

SSL 客户端策略支持的加密套件为 **rsa_aes_128_cbc_sha**。

【视图】

SSL 客户端策略视图

【缺省用户角色】

network-admin

【参数】

dhe_rsa_aes_128_cbc_sha：密钥交换算法采用 DHE RSA、数据加密算法采用 128 位的 AES、MAC 算法采用 SHA。

dhe_rsa_aes_128_cbc_sha256：密钥交换算法采用 DHE RSA、数据加密算法采用 128 位的 AES_CBC、MAC 算法采用 SHA256。

dhe_rsa_aes_256_cbc_sha：密钥交换算法采用 DHE RSA、数据加密算法采用 256 位的 AES、MAC 算法采用 SHA。

dhe_rsa_aes_256_cbc_sha256：密钥交换算法采用 DHE RSA、数据加密算法采用 256 位的 AES_CBC、MAC 算法采用 SHA256。

ecdhe_ecdsa_aes_128_cbc_sha256：密钥交换算法采用 ECDHE ECDSA、数据加密算法采用 128 位的 AES_CBC、MAC 算法采用 SHA256。

ecdhe_ecdsa_aes_128_gcm_sha256：密钥交换算法采用 ECDHE ECDSA、数据加密算法采用 128 位的 AES_GCM、MAC 算法采用 SHA256。

ecdhe_ecdsa_aes_256_cbc_sha384：密钥交换算法采用 ECDHE ECDSA、数据加密算法采用 256 位的 AES_CBC、MAC 算法采用 SHA384。

ecdhe_ecdsa_aes_256_gcm_sha384：密钥交换算法采用 ECDHE ECDSA、数据加密算法采用 256 位的 AES_GCM、MAC 算法采用 SHA384。

ecdhe_rsa_aes_128_cbc_sha256：密钥交换算法采用 ECDHE RSA、数据加密算法采用 128 位的 AES_CBC、MAC 算法采用 SHA256。

ecdhe_rsa_aes_128_gcm_sha256：密钥交换算法采用 ECDHE RSA、数据加密算法采用 128 位的 AES_GCM、MAC 算法采用 SHA256。

ecdhe_rsa_aes_256_cbc_sha384：密钥交换算法采用 ECDHE RSA、数据加密算法采用 256 位的 AES_CBC、MAC 算法采用 SHA384。

ecdhe_rsa_aes_256_gcm_sha384：密钥交换算法采用 ECDHE RSA、数据加密算法采用 256 位的 AES_GCM、MAC 算法采用 SHA384。

exp_rsa_des_cbc_sha：满足出口限制的算法套件。密钥交换算法采用 RSA、数据加密算法采用 DES_CBC、MAC 算法采用 SHA。

exp_rsa_rc2_md5：满足出口限制的算法套件。密钥交换算法采用 RSA、数据加密算法采用 RC2、MAC 算法采用 MD5。

exp_rsa_rc4_md5：满足出口限制的算法套件。密钥交换算法采用 RSA、数据加密算法采用 RC4、MAC 算法采用 MD5。

rsa_3des_ede_cbc_sha：密钥交换算法采用 RSA、数据加密算法采用 3DES_EDE_CBC、MAC 算法采用 SHA。

rsa_aes_128_cbc_sha：密钥交换算法采用 RSA、数据加密算法采用 128 位 AES_CBC、MAC 算法采用 SHA。

rsa_aes_128_cbc_sha256：密钥交换算法采用 RSA、数据加密算法采用 128 位的 AES_CBC、MAC 算法采用 SHA256。

rsa_aes_256_cbc_sha：密钥交换算法采用 RSA、数据加密算法采用 256 位 AES_CBC、MAC 算法采用 SHA。

rsa_aes_256_cbc_sha256：密钥交换算法采用 RSA、数据加密算法采用 256 位的 AES_CBC、MAC 算法采用 SHA256。

rsa_des_cbc_sha：密钥交换算法采用 RSA、数据加密算法采用 DES_CBC、MAC 算法采用 SHA。

rsa_rc4_128_md5：密钥交换算法采用 RSA、数据加密算法采用 128 位的 RC4、MAC 算法采用 MD5。

rsa_rc4_128_sha：密钥交换算法采用 RSA、数据加密算法采用 128 位的 RC4、MAC 算法采用 SHA。

【使用指导】

为了提高安全性，SSL 协议采用了如下算法：

- 数据加密算法：用来对传输的数据进行加密，以保证数据传输的私密性。常用的数据加密算法通常为对称密钥算法，如 DES_CBC、3DES_EDE_CBC、AES_CBC、RC4 等。使用对称密钥算法时，要求 SSL 服务器端和 SSL 客户端具有相同的密钥。
- MAC（Message Authentication Code，消息验证码）算法：用来计算数据的 MAC 值，以防止发送的数据被篡改。常用的 MAC 算法有 MD5、SHA 等。使用 MAC 算法时，要求 SSL 服务器端和 SSL 客户端具有相同的密钥。

- 密钥交换算法：用来实现密钥交换，以保证对称密钥算法、MAC 算法中使用的密钥在 SSL 服务器端和 SSL 客户端之间安全地传递。常用的密钥交换算法通常为非对称密钥算法，如 RSA。

通过本命令可以配置 SSL 客户端策略支持的算法组合。例如，**rsa_des_cbc_sha** 表示 SSL 客户端支持的密钥交换算法为 RSA、数据加密算法为 DES_CBC、MAC 算法为 SHA。

SSL 客户端将本端支持的加密套件发送给 SSL 服务器，SSL 服务器将自己支持的加密套件与 SSL 客户端支持的加密套件比较。如果 SSL 服务器支持的加密套件中存在 SSL 客户端支持的加密套件，则加密套件协商成功；否则，加密套件协商失败。

需要注意的是，如果多次执行本命令，则新的配置覆盖原有配置。

【举例】

配置 SSL 客户端策略支持的加密套件为：密钥交换算法采用 RSA、数据加密算法采用 128 位 AES_CBC、MAC 算法采用 SHA。

```
<Sysname> system-view
[Sysname] ssl client-policy policy1
[Sysname-ssl-client-policy-policy1] prefer-cipher rsa_aes_128_cbc_sha
```

【相关命令】

- **ciphersuite**
- **display ssl client-policy**

1.1.8 server-verify enable

server-verify enable 命令用来配置客户端需要对服务器端进行基于数字证书的身份验证。

undo server-verify enable 命令用来配置客户端不要求对服务器端进行基于数字证书的身份验证，默认 SSL 服务器身份合法。

【命令】

server-verify enable

undo server-verify enable

【缺省情况】

SSL 客户端需要对 SSL 服务器端进行基于数字证书的身份验证。

【视图】

SSL 客户端策略视图

【缺省用户角色】

network-admin

【使用指导】

SSL 通过数字证书实现对对端的身份进行验证。数字证书的详细介绍，请参见“安全配置指导”中的“PKI”。

如果执行了 **server-verify enable** 命令，则 SSL 服务器端需要将自己的数字证书提供给客户端，以便客户端对服务器端进行基于数字证书的身份验证。只有身份验证通过后，SSL 客户端才会访问该 SSL 服务器。

【举例】

配置 SSL 客户端需要对 SSL 服务器端进行基于数字证书的身份验证。

```
<Sysname> system-view
[Sysname] ssl client-policy policy1
[Sysname-ssl-client-policy-policy1] server-verify enable
```

【相关命令】

- **display ssl client-policy**

1.1.9 session cachesize

session cachesize 命令用来配置 SSL 服务器上可以缓存的最大会话数目。

undo session cachesize 命令用来恢复缺省情况。

【命令】

session cachesize *size*

undo session cachesize

【缺省情况】

SSL 服务器上可以缓存的最大会话数目为 500 个。

【视图】

SSL 服务器端策略视图

【缺省用户角色】

network-admin

【参数】

size：缓存的最大会话数目，取值范围为 100～1000。

【使用指导】

通过 SSL 握手协议协商会话参数并建立会话的过程比较复杂。为了简化 SSL 握手过程，SSL 允许重用已经协商出的会话参数建立会话。为此，SSL 服务器上需要保存已有的会话信息。本命令用来限制可以保存的会话数目。如果缓存的会话数目达到最大值，SSL 将拒绝缓存新协商出的会话。

【举例】

配置 SSL 服务器上可以缓存的最大会话数目为 600 个。

```
<Sysname> system-view
[Sysname] ssl server-policy policy1
[Sysname-ssl-server-policy-policy1] session cachesize 600
```

【相关命令】

- **display ssl server-policy**

1.1.10 ssl client-policy

ssl client-policy 命令用来创建 SSL 客户端策略，并进入 SSL 客户端策略视图。

undo ssl client-policy 命令用来删除已创建的 SSL 客户端策略。

【命令】

ssl client-policy *policy-name*

undo ssl client-policy *policy-name*

【缺省情况】

设备上不存在任何 SSL 客户端策略。

【视图】

系统视图

【缺省用户角色】

network-admin

【参数】

policy-name：SSL 客户端策略名，为 1～31 个字符的字符串，不区分大小写。

【使用指导】

SSL 客户端策略视图下可以配置 SSL 客户端启动时使用的 SSL 参数，如使用的 PKI 域、支持的加密套件等。只有与应用层协议，如 DDNS（Dynamic Domain Name System，动态域名系统），关联后，SSL 客户端策略才能生效。

【举例】

创建 SSL 客户端策略 policy1，并进入 SSL 客户端策略视图。

```
<Sysname> system-view
[Sysname] ssl client-policy policy1
[Sysname-ssl-client-policy-policy1]
```

【相关命令】

- **display ssl client-policy**

1.1.11 ssl renegotiation disable

ssl renegotiation disable 命令用来关闭 SSL 重协商。

undo ssl renegotiation disable 命令用来恢复缺省情况。

【命令】

ssl renegotiation disable

undo ssl renegotiation disable

【缺省情况】

允许 SSL 重协商。

【视图】

系统视图

【缺省用户角色】

network-admin

【使用指导】

关闭 SSL 重协商是指，不允许复用已有的 SSL 会话进行 SSL 快速协商，每次 SSL 协商必须进行完整的 SSL 握手过程。关闭 SSL 重协商会导致系统付出更多的计算开销，但可以避免潜在的风险，安全性更高。

通常情况下，不建议关闭 SSL 重协商。本命令仅用于用户明确要求关闭重协商的场景。

【举例】

关闭 SSL 重协商。

```
<Sysname> system-view
[Sysname] ssl renegotiation disable
```

1.1.12 ssl server-policy

ssl server-policy 命令用来创建 SSL 服务器端策略，并进入 SSL 服务器端策略视图。

undo ssl server-policy 命令用来删除已创建的 SSL 服务器端策略。

【命令】

ssl server-policy *policy-name*

undo ssl server-policy *policy-name*

【缺省情况】

设备上不存在任何 SSL 服务器端策略。

【视图】

系统视图

【缺省用户角色】

network-admin

【参数】

policy-name：SSL 服务器端策略名，为 1～31 个字符的字符串，不区分大小写。

【使用指导】

SSL 服务器端策略视图下可以配置 SSL 服务器启动时使用的 SSL 参数，如使用的 PKI 域、支持的加密套件等。只有与 HTTPS 等应用关联后，SSL 服务器端策略才能生效。

【举例】

创建 SSL 服务器端策略 policy1，并进入 SSL 服务器端策略视图。

```
<Sysname> system-view
[Sysname] ssl server-policy policy1
[Sysname-ssl-server-policy-policy1]
```

【相关命令】

- **display ssl server-policy**

1.1.13 ssl version disable

ssl version disable 命令用来关闭对应版本号的 SSL 协商功能。

undo ssl version disable 命令用来恢复缺省情况。

【命令】

非 FIPS 模式下：

ssl version { **ssl3.0 | tls1.0** | **tls1.1** } *** disable**

undo ssl version { **ssl3.0 | tls1.0 | tls1.1** } *** disable**

FIPS 模式下：

ssl version { **tls1.0** | **tls1.1** } *** disable**

undo ssl version { **tls1.0** | **tls1.1** } *** disable**

【缺省情况】

非 FIPS 模式下：

允许使用 SSL3.0、TLS1.0、TLS1.1、TLS1.2 版本的协商功能。

FIPS 模式下：

允许使用 TLS1.0、TLS1.1、TLS1.2 版本的协商功能。

【视图】

系统视图

【缺省用户角色】

network-admin

【参数】

ssl3.0：SSL 协商的版本为 SSL3.0。

tls1.0：SSL 协商的版本为 TLS1.0。

tls1.1：SSL 协商的版本为 TLS1.1。

【使用指导】

当对系统安全性有较高要求时可以关闭 SSL3.0、TLS1.0 和 TLS1.1 版本的协商功能。

- 如果系统视图下关闭了某版本的 SSL 协商功能，但 SSL 客户端策略视图下配置了对应的协议版本，则该 SSL 客户端策略仍可使用该版本的 SSL 协商功能。
- 如果系统视图下同时关闭了 SSL3.0、TLS1.0 和 TLS1.1 版本的 SSL 协商功能，则 SSL 服务器端策略只能使用 TLS1.2 版本进行 SSL 协商，否则与客户端设备协商使用的版本。
- 需要注意的是，如果通过本命令关闭了指定版本的 SSL 协商功能，并不会同时关闭比其更低版本的 SSL 协商功能，例如，**ssl version tls1.1 disable** 命令仅表示关闭了 TLS1.1 版本的 SSL 协商功能，不会同时关闭 TLS1.0 版本。

【举例】

关闭 SSL 3.0 版本。

```
<Sysname> system-view
[Sysname] ssl version ssl3.0 disable
```

1.1.14 version

version 命令用来配置 SSL 客户端策略使用的 SSL 协议版本。

undo version 命令恢复缺省情况。

【命令】

非 FIPS 模式下：

version { **ssl3.0** | **tls1.0** | **tls1.1** | **tls1.2** }

undo version

FIPS 模式下：

version { **tls1.0** | **tls1.1** | **tls1.2** }

undo version

【缺省情况】

SSL 客户端策略使用的 SSL 协议版本为 TLS 1.0。

【视图】

SSL 客户端策略视图

【缺省用户角色】

network-admin

【参数】

ssl3.0：版本为 SSL 3.0。

tls1.0：版本为 TLS 1.0。

tls1.1：版本为 TLS1.1。

tls1.2：版本为 TLS1.2。

【使用指导】

如果多次执行本命令，则新的配置覆盖原有配置。

系统视图下可以关闭 SSL3.0/TLS1.0/TLS1.1 版本(配置命令 **ssl version disable**)。如果使用本命令配置的 SSL 协议版本，在系统视图下被关闭， SSL 客户端仍可使用该版本。

对安全性要求较高的环境下，建议为不要为 SSL 客户端指定 SSL3.0 版本。

【举例】

配置 SSL 客户端策略使用的 SSL 协议版本为 TLS 1.0。

```
<Sysname> system-view
[Sysname] ssl client-policy policy1
[Sysname-ssl-client-policy-policy1] version tls1.0
```

【相关命令】

- **display ssl client-policy**