目 录

1 keychain

1.1 keychain配置命令

1.1.1 accept-lifetime utc

accept-lifetime utc 命令用来配置用于报文接收的 key 的绝对时间模式的生命周期。

undo accept-lifetime 命令用来恢复缺省情况。

【命令】

accept-lifetime utc *start-time start-date* { **duration** { *duration-value* | **infinite** } | **to** *end-time end-date* }

undo accept-lifetime

【缺省情况】

未配置用于报文接收的 key 的生命周期。

【视图】

key 视图

【缺省用户角色】

network-admin

mdc-admin

【参数】

start-time：开始时间，格式为 HH:MM:SS（小时:分钟:秒），取值范围为 0:0:0～23.59:59。

start-date：开始日期，格式为 MM/DD/YYYY（月/日/年）或 YYYY/MM/DD（年/月/日），MM 的取值范围为 1～12，DD 的取值范围与月份有关，YYYY 的取值范围为 2000～2035。

duration *duration-value*：指定 key 的生命周期从设置的 *start-time* 和 *start-date* 开始所持续时间，取值范围为 1～2147483646，单位为秒。

duration infinite：表示从设置的 *start-time* 和 *start-date* 开始，key 永远可以用来验证接收的报文。

to：指定结束时间和日期。

end-time：结束时间，格式为 HH:MM:SS（小时:分钟:秒），取值范围为 0:0:0～23.59:59。

end-date：结束日期，格式为 MM/DD/YYYY（月/日/年）或 YYYY/MM/DD（年/月/日），MM 的取值范围为 1～12，DD 的取值范围与月份有关，YYYY 的取值范围为 2000～2035。

【使用指导】

只有同时满足如下条件的 key，才是有效 key，才可被应用程序用于对接收的报文进行校验：

- 配置了认证密钥
- 配置了认证算法
- 系统当前的绝对时间处于 accept-lifetime utc 指定的时间范围内

如果应用程序接收到的报文中携带 *key-id* 信息，且该 *key-id* 对应的 key 是有效的，则使用该 key 对接收到的报文进行校验；如果该 *key-id* 对应的 key 是无效的，则报文校验失败。

如果应用程序接收到的报文中没有携带 *key-id* 信息，将使用当前 keychain 中所有的有效 key 对接收到的报文进行校验，如果接收到的报文无法通过任何一个有效 key 的校验，则报文校验失败。

应用程序可以使用多个有效 key 对接收到的报文进行校验。

【举例】

在工作于绝对时间模式的 keychain abc 中，配置用来校验接收报文时 key 1 的生命周期。

```
<Sysname> system-view
[Sysname] keychain abc mode absolute
[Sysname-keychain-abc] key 1
[Sysname-keychain-abc-key-1] accept-lifetime utc 12:30 2015/1/21 to 18:30 2015/1/21
```

1.1.2 accept-tolerance

accept-tolerance 命令用来为 keychain 中的 key 延长其在报文接收时的生命周期。

undo accept-tolerance 命令用来恢复缺省情况。

【命令】

accept-tolerance { *value* | **infinite** }

undo accept-tolerance

【缺省情况】

没有为 keychain 中的 key 延长其在报文接收时的生命周期。

【视图】

keychain 视图

【缺省用户角色】

network-admin

mdc-admin

【参数】

value：为指定 keychain 中的 key 延长其在报文接收时的生命周期，取值范围为 1～8640000，单位为秒。

infinite：为指定 keychain 中的 key 无限延长其在报文接收时的生命周期，即 key 在报文接收的过程中永久有效。

【使用指导】

配置本命令后，由 **accept-lifetime utc** 指定的 key 的生命周期的开始和结束时间都将做相应的延长。当用户需要修改认证双方的校验信息时，可能会出现由于校验信息的不匹配导致的业务中断。为了避免上述情况的发生，可使用该命令为应用协议提供不中断其业务的报文校验服务。

【举例】

为 keychain abc 中的 key 延长其在报文接收时的生命周期，延长值为 100 秒。

```
<Sysname> system-view
[Sysname] keychain abc mode absolute
```

```
[Sysname-keychain-abc] accept-tolerance 100
```

为 keychain abc 中的 key 延长其在报文接收时的生命周期，延长值为无限大。

```
<Sysname> system-view
[Sysname] keychain abc mode absolute
[Sysname-keychain-abc] accept-tolerance infinite
```

1.1.3 authentication-algorithm

authentication-algorithm 命令用来配置 key 的认证算法。

undo authentication-algorithm 命令用来恢复缺省情况。

【命令】

authentication-algorithm { **hmac-md5** | **hmac-sha-256** | **md5** }

undo authentication-algorithm

【缺省情况】

未配置 key 的认证算法。

【视图】

key 视图

【缺省用户角色】

network-admin

mdc-admin

【参数】

hmac-md5：HMAC-MD5 认证算法。

hmac-sha-256：HMAC-SHA-256 认证算法。

md5：MD5 认证算法。

【使用指导】

如果应用程序不支持 **authentication-algorithm** 配置的认证算法，则无法使用该 key 进行报文校验。

【举例】

在工作于绝对时间模式的 keychain abc 中，配置 key 1 的认证算法为 MD5。

```
<Sysname> system-view
[Sysname] keychain abc mode absolute
[Sysname-keychain-abc] key 1
[Sysname-keychain-abc-key-1] authentication-algorithm md5
```

1.1.4 default-send-key

default-send-key 命令用来为 keychain 指定一个缺省发送 key。

undo default-send-key 命令用来恢复缺省情况。

【命令】

default-send-key

undo default-send-key

【缺省情况】

keychain 中不存在缺省发送 key。

【视图】

key 视图

【缺省用户角色】

network-admin

mdc-admin

【使用指导】

当 keychain 中的发送 key 处于非活跃状态时，可以通过使用缺省的发送 key 对报文进行认证。同一个 keychain 中，只能将一个 key 指定为缺省的发送 key，且需要为该缺省的发送 key 配置认证算法和认证密钥。

【举例】

指定 keychain abc 中的 key 1 为缺省发送 key。

```
<Sysname> system-view
[Sysname] keychain abc mode absolute
[Sysname-keychain-abc] key 1
[Sysname-keychain-abc-key-1] default-send-key
```

1.1.5 display keychain

display keychain 命令用来显示 keychain 的信息。

【命令】

display keychain [**name** *keychain-name* [**key** *key-id*]]

【视图】

任意视图

【缺省用户角色】

network-admin

network-operator

mdc-admin

mdc-operator

【参数】

name *keychain-name*：keychain 名称，为 1～63 个字符的字符串，区分大小写。如果不指定本参数，则显示所有 keychain 的信息。

key *key-id*：key 的标识符，取值范围为 0～281474976710655。如果不指定本参数，则显示属于某 keychain 的全部 key 的信息。

【举例】

显示 keychain 的信息。

```
<Sysname> display keychain
```

```
Keychain name          : abc
  Mode                 : absolute
  Accept tolerance     : 0
  TCP kind value       : 254
  TCP algorithm value
    HMAC-MD5           : 5
    MD5                : 3
  Default send key ID  : 2 (Inactive)
  Active send key ID   : 1
  Active accept key IDs: 1 2

  Key ID               : 1
    Key string         : $c$3$vuJpEX3Lah7xcSR2uqmrTK2IZQJZguJh3g==
    Algorithm          : md5
    Send lifetime      : 01:00:00 2015/01/22 to 01:00:00 2015/01/25
    Send status        : Active
    Accept lifetime    : 01:00:00 2015/01/22 to 01:00:00 2015/01/27
    Accept status      : Active

  Key ID               : 2
    Key string         : $c$3$vuJpEX3Lah7xcSR2uqmrTK2IZQJZguJh3g==
    Algorithm          : md5
    Send lifetime      : 01:00:01 2015/01/25 to 01:00:00 2015/01/27
    Send status        : Inactive
    Accept lifetime    : 01:00:00 2015/01/22 to 01:00:00 2015/01/27
    Accept status      : Active
```

表1-1 display keychain 命令显示信息描述表

字段	描述
Keychain name	keychain名称
Mode	keychain的时间模式，取值为absolute，表示绝对时间模式
Accept tolerance	key在报文接收时的容忍度，单位为分钟
TCP kind value	TCP增强认证选项中的类型值
TCP algorithm value	TCP认证算法所对应的算法ID
Default send key ID	缺省发送key的ID、以及缺省发送key的状态
Active send key ID	用于对发送报文进行校验的有效key
Active accept key IDs	用于对接收报文进行校验的有效key
Key ID	key的标识符
Key string	key的认证密钥（密文形式）
Algorithm	key的认证算法： • hmac-md5：HMAC-MD5 认证算法 • hmac-sha-256：HMAC-SHA-256 认证算法

字段	描述
	• md5：MD5 认证算法
Send lifetime	用来校验发送报文时key的生命周期
Send status	用来校验发送报文时，key是否有效： • Active：key 有效 • Inactive：key 无效
Accept lifetime	用来校验接收报文时key的生命周期
Accept status	用来校验接收报文时，key是否有效： • Active：key 有效 • Inactive：key 无效

1.1.6 key

key 命令用来创建一个 key，并进入 key 视图。如果指定的 key 已经存在，则直接进入 key 视图。

undo key 命令用来删除指定的 key 及 key 视图下的所有配置。

【命令】

key *key-id*

undo key *key-id*

【缺省情况】

不存在 key。

【视图】

keychain 视图

【缺省用户角色】

network-admin

mdc-admin

【参数】

key-id：key 的标识符，取值范围为 0～281474976710655。

【使用指导】

一个 keychain 下可以配置多个 key，各个 key 必须指定不同的 *key-id*。

【举例】

创建标识符为 1 的 key，并进入 key 视图。

```
<Sysname> system-view
[Sysname] keychain abc mode absolute
[Sysname-keychain-abc] key 1
[Sysname-keychain-abc-key-1]
```

1.1.7 keychain

keychain 命令用来创建一个 keychain，并进入 keychain 视图。如果指定的 keychain 已经存在，则直接进入 keychain 视图。

undo keychain 命令用来删除指定的 keychain 及 keychain 视图下的所有配置。

【命令】

keychain *keychain-name* [**mode absolute**]

undo keychain *keychain-name*

【缺省情况】

不存在 keychain。

【视图】

系统视图

【缺省用户角色】

network-admin

mdc-admin

【参数】

keychain-name：keychain 名，为 1～63 个字符的字符串，区分大小写。

mode：keychain 的时间模式。

absolute：绝对时间模式。该模式的 keychain 中，key 的生命周期是从指定的起始日期、时间到指定的结束日期、时间，各个日期和时间都是 UTC（UTC，Coordinated Universal Time，国际协调时间）绝对时间，不受系统的时区和夏令时的影响。

【使用指导】

创建 keychain 时必须指定其工作的时间模式。对于已存在的 keychain，不可修改其工作的时间模式。进入已存在的 keychain 视图时可以不指定其工作的时间模式。

【举例】

创建名为 abc 的 keychain，指定其工作在绝对时间模式下，并进入 keychain 视图。

```
<Sysname> system-view
[Sysname] keychain abc mode absolute
[Sysname-keychain-abc]
```

1.1.8 key-string

key-string 命令用来配置 key 的认证密钥。

undo key-string 命令用来恢复缺省情况。

【命令】

key-string { **cipher** | **plain** } *string*

undo key-string

【缺省情况】

未配置 key 的认证密钥。

【视图】

key 视图

【缺省用户角色】

network-admin

mdc-admin

【参数】

cipher：以密文方式设置密钥。

plain：以明文方式设置密钥，该密码将以密文形式存储。

string：密钥字符串，区分大小写。明文密钥为 1～255 个字符的字符串，密文密钥为 33～373 个字符的字符串。

【使用指导】

key 的认证密钥的明文长度有可能会超出应用程序支持的范围，这种情况下应用程序需要截取自己支持的长度范围的明文密钥来对报文进行校验。

【举例】

在 key 视图下，以明文形式设置 key 1 的密钥为 123456。

```
<Sysname> system-view
[Sysname] keychain abc mode absolute
[Sysname-keychain-abc] key 1
[Sysname-keychain-abc-key-1] key-string plain 123456
```

1.1.9 send-lifetime utc

send-lifetime utc 命令用来配置用于报文发送的 key 的绝对时间模式的生命周期。

undo send-lifetime 命令用来恢复缺省情况。

【命令】

send-lifetime utc *start-time start-date* { **duration** { *duration-value* | **infinite** } | **to** *end-time end-date* }

undo send-lifetime

【缺省情况】

未配置用于报文发送的 key 的生命周期。

【视图】

key 视图

【缺省用户角色】

network-admin

mdc-admin

【参数】

start-time：开始时间，格式为 HH:MM:SS（小时:分钟:秒），取值范围为 0:0:0～23.59:59。

start-date：开始日期，格式为 MM/DD/YYYY（月/日/年）或 YYYY/MM/DD（年/月/日），MM 的取值范围为 1～12，DD 的取值范围与月份有关，YYYY 的取值范围为 2000～2035。

duration *duration-value*：指定 key 的生命周期从设置的 *start-time* 和 *start-date* 开始所持续时间，取值范围为 1～2147483646，单位为秒。

duration infinite：表示从设置的 *start-time* 和 *start-date* 开始，key 永远可以用来验证发送的报文。

to：指定结束时间和日期。

end-time：结束时间，格式为 HH:MM:SS（小时:分钟:秒），取值范围为 0:0:0～23.59:59。

end-date：结束日期，格式为 MM/DD/YYYY（月/日/年）或 YYYY/MM/DD（年/月/日），MM 的取值范围为 1～12，DD 的取值范围与月份有关，YYYY 的取值范围为 2000～2035。

【使用指导】

只有同时满足如下条件的 key，才是有效 key，才可被应用程序用于对发送的报文进行校验：

- 配置了认证密钥
- 配置了认证算法
- 系统当前的绝对时间处于 **send-lifetime utc** 指定的时间范围内

同一个 keychain 内的各个 key 使用 **send-lifetime utc** 指定的生命周期不可重叠，以确保在同一时刻，应用程序只使用一个 key 对发送的报文进行校验。

【举例】

在工作于绝对时间模式的 keychain abc 下，配置用来校验发送报文时 key 1 的生命周期。

```
<Sysname> system-view
[Sysname] keychain abc mode absolute
[Sysname-keychain-abc] key 1
[Sysname-keychain-abc-key-1] send-lifetime utc 12:30 2015/1/21 to 18:30 2015/1/21
```

1.1.10 tcp-algorithm-id

tcp-algorithm-id 命令用来配置 keychain 支持的 TCP 认证算法的算法 ID。

undo tcp-algorithm-id 命令用来恢复缺省情况。

【命令】

tcp-algorithm-id { **hmac-md5** | **md5** } *algorithm-id*

undo tcp-algorithm-id { **hmac-md5** | **md5** }

【缺省情况】

MD5 认证算法的算法 ID 是 3，HMAC-MD5 认证算法的算法 ID 是 5。

【视图】

keychain 视图

【缺省用户角色】

network-admin

mdc-admin

【参数】

hmac-md5：采用 HMAC-MD5 认证算法，密钥长度为 16 字节。

md5：采用 MD5 认证算法，密钥长度为 16 字节。

algorithm-id：认证算法的算法 ID，取值范围为 1～63。

【使用指导】

建立 TCP 连接时使用 keychain 认证的应用程序，其发送和接收的 TCP 报文中会携带增强认证选项，增强认证选项中的 *algorithm-id* 字段用来表示认证算法的算法 ID。由于 *algorithm-id* 不是 IANA（Internet Assigned Numbers Authority，因特网地址分配组织）统一定义的，不同的厂商使用 *algorithm-id* 所代表的算法类型不尽相同。在与友商设备互通时，本端对 *algorithm-id* 的配置必须与对端的保持一致，例如，当友商设备中 HMAC-MD5 认证算法的算法 ID 是 3 时，需要在本端设备上配置 **tcp-algorithm-id hmac-md5** 3 命令才能互通。

【举例】

创建名为 abc 的 keychain，并将 HMAC-MD5 认证算法的 *algorithm-id* 配置为 1。

```
<Sysname> system-view
[Sysname] keychain abc mode absolute
[Sysname-keychain-abc] tcp-algorithm-id hmac-md5 1
```

1.1.11 tcp-kind

tcp-kind 命令用来配置 TCP 增强认证选项中的类型值。

undo tcp-kind 命令用来恢复缺省情况。

【命令】

tcp-kind *kind-value*

undo tcp-kind

【缺省情况】

TCP 增强认证选项中的类型值为 254。

【视图】

keychain 视图

【缺省用户角色】

network-admin

mdc-admin

【参数】

kind-value：TCP 增强认证选项中的类型值，取值范围为 28～255，缺省值为 254。

【使用指导】

建立 TCP 连接时使用 keychain 认证的应用程序，其发送和接收的 TCP 报文中会携带增强认证选项，通信双方所指定的增强认证选项中的类型值必须保持一致，否则会导致报文校验失败。

【举例】

在工作于绝对时间模式的 keychain abc 下，配置 TCP 增强认证选项中的类型值为 252。

```
<Sysname> system-view
[Sysname] keychain abc mode absolute
[Sysname-keychain-abc] tcp-kind 252
```