NAME

ip-xfrm - transform configuration

SYNOPSIS

ip [*OPTIONS*] **xfrm** { *COMMAND* | **help** }

ip xfrm *XFRM-OBJECT* { *COMMAND* | **help** }

XFRM-OBJECT := **state | policy | monitor**

ip xfrm state{ **add | update** } *ID* [*ALGO-LIST*] [**mode** *MODE*] [**mark** *MARK* [**mask** *MASK*]] [**reqid** *REQID*] [**seq** *SEQ*] [**replay-window** *SIZE*] [**replay-seq** *SEQ*] [**replay-oseq** *SEQ*] [**flag** *FLAG-LIST*] [**sel** *SELECTOR*] [*LIMIT-LIST*] [**encap** *ENCAP*] [**coa** *ADDR*[/*PLEN*]] [**ctx** *CTX*]

ip xfrm state allocspi *ID* [**mode** *MODE*] [**mark** *MARK* [**mask** *MASK*]] [**reqid** *REQID*] [**seq** *SEQ*] [**min** *SPI* **max** *SPI*]

ip xfrm state{ **delete | get** } *ID* [**mark** *MARK* [**mask** *MASK*]]

ip xfrm state{ **deleteall | list** } [*ID*] [**mode** *MODE*] [**reqid** *REQID*] [**flag** *FLA G-LIST*]

ip xfrm state flush[**proto** *XFRM-PR OTO*]

ip xfrm state count

ID := [**src** *ADDR*] [**dst** *ADDR*] [**proto** *XFRM-PROTO*] [**spi** *SPI*]

XFRM-PROTO := **esp | ah | comp | route2 | hao**

ALGO-LIST := [*ALGO-LIST*] *ALGO*

ALGO := { **enc | auth** } *ALGO-NAME ALGO-KEYMAT* |
auth-trunc *ALGO-NAME ALGO-KEYMAT ALGO-TRUNC-LEN* |
aead *ALGO-NAME ALGO-KEYMAT ALGO-ICV-LEN* |
comp *ALGO-NAME*

MODE := **transport | tunnel | beet | ro | in_trigger**

FLAG-LIST := [*FLAG-LIST*] *FLAG*

FLAG := **noecn | decap-dscp | nopmtudisc | wildrecv | icmp | af-unspec | align4**

SELECTOR := [**src** *ADDR*[/*PLEN*]] [**dst** *ADDR*[/*PLEN*]] [**dev** *DEV*] [*UPSPEC*]

UPSPEC := **proto** { *PROTO* |
{ **tcp | udp | sctp | dccp** } [**sport** *PORT*] [**dport** *PORT*] |
{ **icmp | ipv6-icmp | mobility-header** } [**type** *NUMBER*] [**code** *NUMBER*] |
gre [**key** { *DOTTED-QUAD* | *NUMBER* }] }

LIMIT-LIST := [*LIMIT-LIST*] **limit** *LIMIT*

LIMIT := { **time-soft | time-hard | time-use-soft | time-use-hard** } *SECONDS* |
{ **byte-soft | byte-hard** } *SIZE* |
{ **packet-soft | packet-hard** } *COUNT*

ENCAP := { **espinudp | espinudp-nonike** } *SPORT DPORT OADDR*

ip xfrm policy{ **add | update** } *SELECTOR* **dir** *DIR* [**ctx** *CTX*] [**mark** *MARK* [**mask** *MASK*]] [**index** *INDEX*] [**ptype** *PTYPE*] [**action** *ACTION*] [**priority** *PRIORITY*] [**flag** *FLAG-LIST*] [*LIMIT-LIST*] [*TMPL-LIST*]

ip xfrm policy{ **delete | get** } { *SELECTOR* | **index** *INDEX* } **dir** *DIR* [**ctx** *CTX*] [**mark** *MARK* [**mask** *MASK*]] [**ptype** *PTYPE*]

ip xfrm policy{ deleteall | list } [*SELECTOR*] [**dir** *DIR*] [**index** *INDEX*] [**pt ype** *PTYPE*] [**action** *ACTION*] [**priority** *PRIORITY*]

ip xfrm policy flush [**ptype** *PTYPE*]

ip xfrm policy count

SELECTOR := [**src** *ADDR*[/*PLEN*]] [**dst** *ADDR*[/*PLEN*]] [**dev** *DEV*] [*UPSPEC*]

UPSPEC := **proto** { *PROTO* |
{ **tcp | udp | sctp | dccp** } [**sport** *PORT*] [**dport** *PORT*] |
{ **icmp | ipv6-icmp | mobility-header** } [**type** *NUMBER*] [**code** *NUMBER*] |
gre [**key** { *DOTTED-QUAD* | *NUMBER* }] }

DIR := **in | out | fwd**

PTYPE := **main | sub**

ACTION := **allow | block**

FLAG-LIST := [*FLAG-LIST*] *FLAG*

FLAG := **localok | icmp**

LIMIT-LIST := [*LIMIT-LIST*] **limit** *LIMIT*

LIMIT := { **time-soft | time-hard | time-use-soft | time-use-hard** } *SECONDS* |
{ **byte-soft | byte-hard** } *SIZE* |
{ **packet-soft | packet-hard** } *COUNT*

TMPL-LIST := [*TMPL-LIST*] **tmpl** *TMPL*

TMPL := *ID* [**mode** *MODE*] [**reqid** *REQID*] [**level** *LEVEL*]

ID := [**src** *ADDR*] [**dst** *ADDR*] [**proto** *XFRM-PROTO*] [**spi** *SPI*]

XFRM-PROTO := **esp | ah | comp | route2 | hao**

MODE := **transport | tunnel | beet | ro | in_trigger**

LEVEL := **required | use**

ip xfrm monitor[**all** | *LIST ofXFRM-OBJECTS*]

DESCRIPTION

xfrm is an IP framework for transforming packets (such as encrypting their payloads). This framework is used to implement the IPsec protocol suite (with the **state** object operating on the Security Association Database, and the **policy** object operating on the Security Policy Database). It is also used for the IP Payload Compression Protocol and features of Mobile IPv6.

ip xfrm state add	add new state into xfrm
ip xfrm state update	update existing state in xfrm
ip xfrm state allocspi	allocate an SPI value
ip xfrm state delete	delete existing state in xfrm
ip xfrm state get	get existing state in xfrm
ip xfrm state deleteall	delete all existing state in xfrm
ip xfrm state list	print out the list of existing state in xfrm
ip xfrm state flush	flush all state in xfrm
ip xfrm state count	count all existing state in xfrm
ip xfrm monitor	state monitoring for xfrm objects

ID is specified by a source address, destination address, transform protocol *XFRM-PROTO*, and/or Security Parameter Index *SPI*. (For IP Payload Compression, the Compression Parameter Index or CPI is used for *SPI*.)

XFRM-PROTO
specifies a transform protocol: IPsec Encapsulating Security Payload (**esp**), IPsec Authentication Header (**ah**), IP Payload Compression (**comp**), Mobile IPv6 Type 2 Routing Header (**route2**), or Mobile IPv6 Home Address Option (**hao**).

ALGO-LIST
contains one or more algorithms to use. Each algorithm *ALGO* is specified by:

- the algorithm type: encryption (**enc**), authentication (**auth** or **auth-trunc**), authenticated encryption with associated data (**aead**), or compression (**comp**)
- the algorithm name *ALGO-NAME* (see below)
- (for all except **comp**) the keying material *ALGO-KEYMAT*, which may include both a key and a salt or nonce value; refer to the corresponding RFC
- (for **auth-trunc** only) the truncation length *ALGO-TRUNC-LEN* in bits
- (for **aead** only) the Integrity Check Value length *ALGO-ICV-LEN* in bits

Encryption algorithms include **ecb(cipher_null)**, **cbc(des)**, **cbc(des3_ede)**, **cbc(cast5)**, **cbc(blowfish)**, **cbc(aes)**, **cbc(serpent)**, **cbc(camellia)**, **cbc(twofish)**, and **rfc3686(ctr(aes))**.

Authentication algorithms include **digest_null**, **hmac(md5)**, **hmac(sha1)**, **hmac(sha256)**, **hmac(sha384)**, **hmac(sha512)**, **hmac(rmd610)**, and **xcbc(aes)**.

Authenticated encryption with associated data (AEAD) algorithms include **rfc4106(gcm(aes))**, **rfc4309(ccm(aes))**, and **rfc4543(gcm(aes))**.

Compression algorithms include **deflate**, **lzs**, and **lzjh**.

MODE specifies a mode of operation for the transform protocol. IPsec and IP Payload Compression modes are **transport**, **tunnel**, and (for IPsec ESP only) Bound End-to-End Tunnel (**beet**). Mobile IPv6 modes are route optimization (**ro**) and inbound trigger (**in_trigger**).

FLAG-LIST
contains one or more of the following optional flags: **noecn**, **decap-dscp**, **nopmtudisc**, **wildrecv**, **icmp**, **af-unspec**, or **align4**.

SELECTOR
selects the traffic that will be controlled by the policy, based on the source address, the destination address, the network device, and/or *UPSPEC*.

UPSPEC
selects traffic by protocol. For the **tcp**, **udp**, **sctp**, or **dccp** protocols, the source and destination port can optionally be specified. For the **icmp**, **ipv6-icmp**, or **mobility-header** protocols, the type and code numbers can optionally be specified. For the **gre** protocol, the key can optionally be specified as a dotted-quad or number. Other protocols can be selected by name or number *PROTO*.

LIMIT-LIST
sets limits in seconds, bytes, or numbers of packets.

ENCAP
encapsulates packets with protocol **espinudp** or **espinudp-nonike**, using source port *SPORT*, destination port *DPORT* , and original address *OADDR*.

ip xfrm policy add	add a new policy
ip xfrm policy update	update an existing policy
ip xfrm policy delete	delete an existing policy
ip xfrm policy get	get an existing policy
ip xfrm policy deleteall	delete all existing xfrm policies
ip xfrm policy list	print out the list of xfrm policies
ip xfrm policy flush	flush policies
ip xfrm policy count	count existing policies

SELECTOR
selects the traffic that will be controlled by the policy, based on the source address, the destination address, the network device, and/or *UPSPEC*.

UPSPEC
selects traffic by protocol. For the **tcp**, **udp**, **sctp**, or **dccp** protocols, the source and destination port can optionally be specified. For the **icmp**, **ipv6-icmp**, or **mobility-header** protocols, the type and code numbers can optionally be specified. For the **gre** protocol, the key can optionally be specified as a dotted-quad or number. Other protocols can be selected by name or number *PROTO*.

DIR selects the policy direction as **in**, **out**, or **fwd**.

CTX sets the security context.

PTYPE
can be **main** (default) or **sub**.

ACTION
can be **allow** (default) or **block**.

PRIORITY
is a number that defaults to zero.

FLAG-LIST
contains one or both of the following optional flags: **local** or **icmp**.

LIMIT-LIST
sets limits in seconds, bytes, or numbers of packets.

TMPL-LIST
is a template list specified using *ID*, *MODE*, *REQID*, and/or *LEVEL*.

ID is specified by a source address, destination address, transform protocol *XFRM-PROTO*, and/or Security Parameter Index *SPI*. (For IP Payload Compression, the Compression Parameter Index or CPI is used for *SPI*.)

XFRM-PROTO
specifies a transform protocol: IPsec Encapsulating Security Payload (**esp**), IPsec Authentication Header (**ah**), IP Payload Compression (**comp**), Mobile IPv6 Type 2 Routing Header (**route2**), or Mobile IPv6 Home Address Option (**hao**).

MODE specifies a mode of operation for the transform protocol. IPsec and IP Payload Compression modes are **transport**, **tunnel**, and (for IPsec ESP only) Bound End-to-End Tunnel (**beet**). Mobile IPv6 modes are route optimization (**ro**) and inbound trigger (**in_trigger**).

LEVEL

can be **required** (default) or **use**.

The xfrm objects to monitor can be optionally specified.

AUTHOR

Manpage revised by David Ward <david.ward@ll.mit.edu>