

BGP Operations and Security

Best Common Practices (RFC7454 + more)

Next-Hop-Self

- When connecting to another provider or at an Internet Exchange use **next-hop-self**
 - so that the next hop to the routes they announce is reachable from your AS

Protecting BGP sessions

- MD5 authentication (RFC2385)
 - To protect the BGP TCP session between peers
 - generates a keyed hash (16-byte) - using the TCP segment and the password

```
router bgp 17821
  neighbor 30.30.30.1 password <key-value>
```

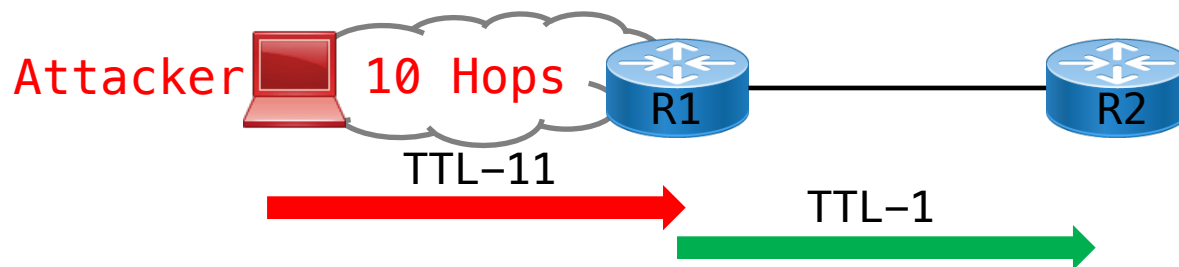
- But susceptible to **collision attacks**
- TCP Authentication Option (RFC5925)
 - Obsoletes RFC2385, but *no known implementations...* please correct ??
 - Key chains (allows moving from one key to another for the same connection), and
 - Support for stronger hash functions

Protecting BGP sessions

- Some organisations dislike MD5, so be flexible
- Do you have any plans to change the key?

Protecting BGP sessions

- GTSM – BGP TTL Security (RFC5082)
 - eBGP has a default TTL of 1
 - Which requires eBGP peers to be directly connected
 - To use any interface other than the directly connected, we generally use `ebgp-multihop`
 - A remote attacker could send spoofed packets with adjusted TTL values to make it seem like its directly connected

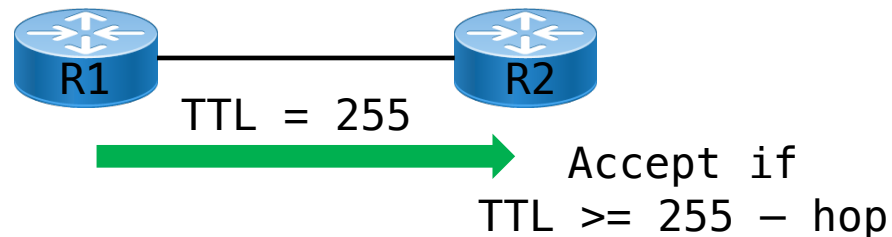


- This way, R2 could be DoSed (will accept the initial TCP SYN) or TCP resets

Protecting BGP sessions

- With GTSM, the TTL between directly connected eBGP peers set to the maximum – 255
 - Only incoming packets with a TTL value equal to or greater than the locally configured value is accepted,
 - Else packet is silently discarded and no ICMP messages generated

```
router bgp 17821
  neighbor <peer-v4/v6 addr> ttl-security hops <1-254>
```

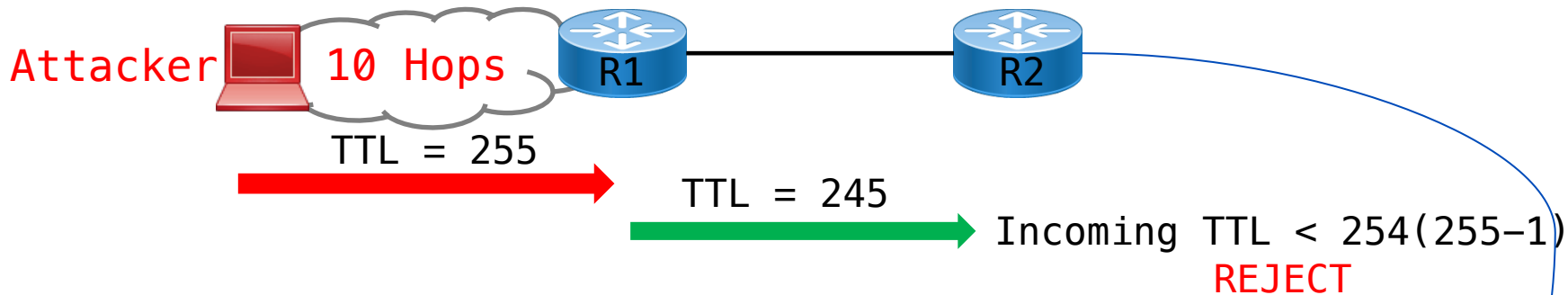


- Needs to be **same on both peers!**

Protecting BGP sessions

- GTSM – BGP TTL Security (RFC5082)
 - Ex: If R2's config was

```
router bgp 17821
neighbor X.X.X.1 ttl-security hops 1
```



```
R2#sh ip bgp neighbors X.X.X.1 | i hop | TTL
External BGP neighbor may be up to 1 hop away.
Connection is ECN Disabled, Minimum incoming TTL 254, Outgoing TTL 255
```

Prefix Filters - Outbound

- To Customers speaking BGP
 - Probably only a *default route*, or
 - WHOLE Internet feed except default and bogons (special use addresses – rfc6890, rfc1918, unassigned blocks)
- To Peers (other ISPs with whom you peer)
 - Send *your prefixes + your downstream customer*, or
 - what you agreed to send
- To Upstream/Transit provider
 - Send *your prefixes + your downstream customers*

Prefix filters - Inbound

- Customers speaking BGP
 - only accept prefixes registered to them, or prefixes registered to their customers
 - If assigned out of your block, you already know
- They may claim to have their own prefix allocated to them from a regional registry (Ex- APNIC)
 - You should check that it is theirs to announce
- Encourage them to aggregate their prefix where possible

Prefix Filters - Inbound

- Customers speaking BGP
 - Only accept their prefixes (their customers)
 - Verify that the prefixes were Assigned or Allocated, and
 - Make sure the prefix length does not exceed /24 (IPv4) and /48 (IPv6) – unless negotiated for TE reason

```
whois -h jwhois.apnic.net 61.45.248.0/21
inetnum:          61.45.248.0 - 61.45.255.255
netname:          APNICTRAINING-AP
descr:           APNIC TRAINING UNIT
descr:           6 Cordelia St. South Brisbane, QLD
country:         AU
org:             ORG-ATU1-AP
admin-c:         AINT1-AP
tech-c:          AINT1-AP
status:          ALLOCATED PORTABLE
mnt-by:         APNIC-HM
mnt-irt:         IRT-ABCINTERNET-SG
last-modified:   2017-08-29T23:00:41Z
source:          APNIC
```

Means it was delegated to an entity

Means it was allocated to the customer, and they can announce the prefix

Prefix Filters - Inbound

- Customers speaking BGP
 - Ex: for a customer with the block 61.45.248.0/21 block

```
router bgp 17821
  address-family ipv4
    neighbor X.X.X.1 prefix-list CUST-PREFIX in
  !
ip prefix-list CUST-PREFIX permit 61.45.248.0/21 le 24
ip prefix-list CUST-PREFIX deny 0.0.0.0/0 le 32
!
```

Prefix Filters - Inbound

- From Peers: Other ISPs/operators with whom you have agreed to exchange routes
 - Only accept their prefixes and their downstream customers (or what was agreed)
 - Verify they have the authority to route those prefixes (and their customers)
 - Typically prefix length should not exceed /24 (IPv4) and /48 (IPv6)
 - Can use RPSL tools like **bgpq3** - <https://github.com/snar/bgpq3>

```
bgpq3 -6A1 PEERv6-IN AS17660
no ipv6 prefix-list PEERv6-IN
ipv6 prefix-list PEERv6-IN permit 2405:d000::/32
ipv6 prefix-list PEERv6-IN permit 2405:d000:7000::/36
```

Prefix Filters - Inbound

- From Peers

- Ex: if a peer has 2001:dc0::/32 and 203.119.96.0/20 prefixes

```
router bgp 17821
  address-family ipv4
    neighbor X.X.X.1 prefix-list PEERv4-IN in
  address-family ipv6
    neighbor X:X:X::1 prefix-list PEERv6-IN in
!
ip prefix-list PEERv4-IN permit 203.119.96.0/20 le 24
ip prefix-list PEERv4-IN deny 0.0.0.0/0 le 32
!
ipv6 prefix-list PEERv6-IN permit 2001:dc0::/32 le 48
ipv6 prefix-list PEERv6-IN deny ::/0 le 128
!
```

Prefix Filters - Inbound

- From Upstream (Transit Provider)
 - Could just be a default route

```
router bgp 17821
  address-family ipv4
    neighbor X.X.X.1 prefix-list DEF-IN in
  address-family ipv6
    neighbor X:X:X::1 prefix-list DEFv6-IN in
!
ip prefix-list DEF-IN permit 0.0.0.0/0
!
ipv6 prefix-list DEFv6-IN permit ::/0
```

Prefix Filters - Inbound

- From Upstream (Transit Provider)
 - the WHOLE Internet feed
 - Do not accept your own prefixes
 - Do not accept bogons
 - special use addresses – rfc6890, rfc1918, and unassigned blocks
 - Do not accept prefix lengths longer than /24 (IPv4) and /48 (IPv6)

Prefix Filters - Inbound

- From Upstream (Transit Provider) – IPv4

```
router bgp 17821
  address-family ipv4
    neighbor X.X.X.1 prefix-list TRANSITv4-IN in
  !
ip prefix-list TRANSITv4-IN deny 0.0.0.0/0                ! Default
ip prefix-list TRANSITv4-IN deny 0.0.0.0/8 le 32         ! Network Zero
ip prefix-list TRANSITv4-IN deny 10.0.0.0/8 le 32       ! RFC1918
ip prefix-list TRANSITv4-IN deny 100.64.0.0/10 le 32    ! RFC6598 shared address
ip prefix-list TRANSITv4-IN deny <your prefix>/X le 32 ! Your address space
ip prefix-list TRANSITv4-IN deny 127.0.0.0/8 le 32     ! Loopback
ip prefix-list TRANSITv4-IN deny 169.254.0.0/16 le 32  ! APIPA
ip prefix-list TRANSITv4-IN deny 172.16.0.0/12 le 32   ! RFC1918
ip prefix-list TRANSITv4-IN deny 192.0.0.0/24 le 32   ! IETF Protocol
ip prefix-list TRANSITv4-IN deny 192.0.2.0/24 le 32   ! TEST1
ip prefix-list TRANSITv4-IN deny 192.168.0.0/16 le 32 ! RFC1918
ip prefix-list TRANSITv4-IN deny 198.18.0.0/15 le 32 ! Benchmarking
ip prefix-list TRANSITv4-IN deny 198.51.100.0/24 le 32 ! TEST2
ip prefix-list TRANSITv4-IN deny 203.0.113.0/24 le 32 ! TEST3
ip prefix-list TRANSITv4-IN deny 224.0.0.0/4 le 32    ! Multicast
ip prefix-list TRANSITv4-IN deny 240.0.0.0/4 le 32    ! Future Use
ip prefix-list TRANSITv4-IN deny 0.0.0.0/0 ge 25      ! Prefixes longer than /24
ip prefix-list TRANSITv4-IN permit 0.0.0.0/0 le 32
```


Prefix Filters - Inbound

- From Upstream (Transit Provider) – IPv6

```
router bgp 17821
  address-family ipv6
    neighbor X:X:X::1 prefix-list TRANSITv6-IN in
  !
  ipv6 prefix-list TRANSITv6-IN deny 2001::/32 le 128           ! Teredo subnets
  ipv6 prefix-list TRANSITv6-IN deny 2001:db8::/32 le 128      ! Documentation
  ipv6 prefix-list TRANSITv6-IN deny 2002::/16 le 128          ! 6to4 subnets
  ipv6 prefix-list TRANSITv6-IN deny <your::/32> le 128        ! Your prefix
  ipv6 prefix-list TRANSITv6-IN deny 3ffe::/16 le 128          ! Old 6bone
  ipv6 prefix-list TRANSITv6-IN deny fc00::/7 le 128           ! ULA
  ipv6 prefix-list TRANSITv6-IN deny fe00::/9 le 128           ! Reserved IETF
  ipv6 prefix-list TRANSITv6-IN deny fe80::/10 le 128          ! Link-local
  ipv6 prefix-list TRANSITv6-IN deny fec0::/10 le 128          ! Reserved IETF
  ipv6 prefix-list TRANSITv6-IN deny ff00::/8 le 128           ! Multicast
  ipv6 prefix-list TRANSITv6-IN permit 2000::/3 le 48          ! Global Unicast
  ipv6 prefix-list TRANSITv6-IN deny ::/0 le 128
```

Bogons

- Not all IP (v4 and v6) are allocated by IANA
- Addresses that should not be seen on the Internet are called "**Bogons**" (also called "**Martians**")
 - RFC6890+RFC1918s + Reserved space
- IANA publishes list of number resources that have been allocated/assigned to RIRs/end-users
 - <https://www.iana.org/assignments/ipv6-unicast-address-assignments/ipv6-unicast-address-assignments.xhtml>
 - <https://www.iana.org/assignments/ipv4-address-space/ipv4-address-space.xhtml>

Bogons

- Commonly found as source addresses of DDoS packets
- We should have ingress and egress filters for bogon routes
 - Should not route them nor accept them from peers
- We could manually craft prefix filters based on the bogon list from IANA
 - But bogon list is dynamic
 - New allocations made out of reserved blocks frequently

Bogon Route Server Project

- In comes the Bogon Route Server project by Team Cymru
 - Provides dynamic bogons information using eBGP multihop sessions
 - Traditional bogons (AS65333)
 - martians plus prefixes not allocated by IANA
 - Full-bogons (AS65332)
 - above plus prefixes allocated to RIRs but not yet assigned to ISPs/end-users by RIRs
- For details:
 - <http://www.team-cymru.org/bogon-reference-bgp.html>



Peering- Bogon Route Servers

- To peer with bogon route servers
 - Write to bogonrs@cymru.com
- You should provide:
 - Your ASN
 - Which bogons you wish to receive
 - Your peering addresses
 - MD5 for BGP?
 - PGP public key (optional)
- It is recommended to have at least 2 (two) peering sessions for redundancy

Maximum Prefix Limit

- It is RECOMMENDED to set the maximum number of prefixes accepted from a peer
 - Prevent memory exhaustion,
 - Prevent impacts of route leaks (where more specifics of big covering prefix gets leaked)

```
router bgp 17821
address-family ip[v4|v6]
neighbor <peer-addr|group> maximum-prefix <max-value> [threshold][restart N] [warning-only]
```

- **max-value**: the max prefix limit
- **threshold**: threshold in % of max limit (by default 75%) to generate warning msg
- **restart**: restart BGP connection after N minutes
- **warning-only**: generate a warning msg when limit is exceeded instead of terminating the BGP session

Maximum Prefix Limit

- Setting the max-prefix limit:
 - Allow future growth
 - Ex: If a peer has a /12 block, allowing upto /24s
 - The number of /24s = 4096 ~ max limit
 - IXPs generally publish the max IPv4/v6 prefixes announced by their route servers (RS)
 - Set your max limit accordingly
 - Ex: @HKIX - max prefix limit ~ [(monthly 2-hr avg)/0.7]

– Examples:

```
neighbor X.X.X.1 maximum-prefix 1000
```

- Drop the peering if more than 1000 prefixes received

```
neighbor X.X.X.1 maximum-prefix 1000 warning-only
```

- Log a warning when it receives more than 1000 prefixes

```
neighbor X.X.X.1 maximum-prefix 1000 90
```

- Logs a warning at 900 prefixes, and drops if more than 1000 received

AS_PATH Filtering

- Do NOT accept/announce prefixes with private ASNs
 - Unless you have customers using private ASNs, or
 - Special arrangements like Cymru Bogon RS

```
router bgp 17821
  address-family ip[v4|v6]
    neighbor <peer-addr|group> remove-private-as [all [replace-as]
```

- Enforce the first ASN in the AS_PATH attribute to be the peer's ASN
 - except if peering with a RS @IXPs

```
router bgp 17821
  bgp enforce-first-as !by default in IOS
```


AS_PATH Filtering

- Limit the AS-PATH length
 - Unusually long AS-PATHs would cause
 - Memory exhaustion, or
 - Issues like below which caused massive global routing updates (~107K per second) for an hour:
 - <https://dyn.com/blog/the-flap-heard-around-the-world/>
 - <https://dyn.com/blog/longer-is-not-better/>

```
bgp-prepend (integer:0-16)
>> bgp-prepend 47868 ~ 47868 mod 256 ~ 252
```

- Some weird announcements:

```
N*> 45.162.216.0/24 4608 24130 7545 6939 263311 268528 268528 268528 268528 268528 268528
268528 268528 268528 268528 268528 268528 268528 268528 268528 i
```

```
N*> 45.162.217.0/24 4608 24130 7545 6939 263311 268528 268528 268528 268528 268528 268528
268528 268528 268528 268528 268528 268528 268528 268528 268528
```

```
0x7F48C876B370 4608 24130 7545 6939 28186 264437 264437 264437 264437 264437 264437
264437 264437 264437 264437 264437 264437 264437 264437 264437 264437 264437 264952 i
```

AS_PATH Filtering

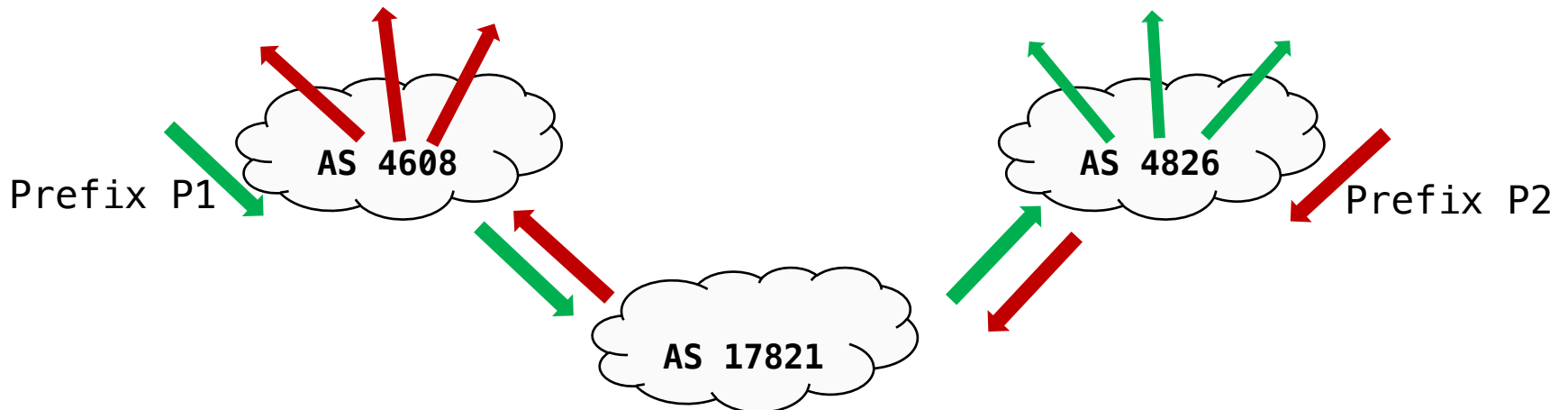
- Limit the AS-PATH length
 - The internet would be around 5-10 ASes deep on average
 - Longest AS-PATHs could be ~ 30 ASNs
 - Consider limiting the AS-PATH length for prefixes you accept

```
router bgp 17821  
  bgp maxas-limit <1-254>
```

RFC 8212 – BGP default reject

- On many platforms, BGP is implicitly permissive!
 - AS-to-AS leaks common during maintenance

```
router bgp 17821
!
neighbor X.X.X.1 remote-as 4608
neighbor X.X.X.1 description eBGP with Upstream-1
!
neighbor A.A.A.1 remote-as 4826
neighbor A.A.A.1 description eBGP with Upstream-2
```



RFC 8212 – BGP default reject

- **RFC8212**- eBGP route propagation without policies
 - Neither accept nor announce routes from/to eBGP peers without explicit policy (import/export) configurations
 - **implicit deny-all** associated with eBGP sessions!
 - Changes to RFC4271:
 - **Decision Process**: *Routes contained in an Adj-RIB-In associated with an EBGP peer SHALL NOT be considered eligible in the Decision Process if no explicit Import Policy has been applied.*
 - **Route Dissemination**: *Routes SHALL NOT be added to an Adj-RIB-Out associated with an EBGP peer if no explicit Export Policy has been applied.*

RFC 8212 – BGP default reject

- But very few known implementations:
 - IOS-XR (all versions)
 - BIRD (2.0.1 and higher)
 - OpenBGPD (6.4 and higher)
 - Nokia SR OS (19.5.R1 and above)
 - <https://github.com/bgp/RFC8212>
- For those OSes that don't support RFC8212 yet
 - Shut the BGP session with the peer (group) during configuration
 - Define and apply explicit export and import policies to the eBGP peer(s)
 - Then no shut the BGP session
 - **Talk to your vendors and force them to support RFC8212!**

Traffic Filters

- **BCP38** (RFC2827)
 - Since **1998!**
 - <https://tools.ietf.org/html/bcp38>
- Only allow traffic with valid source addresses to
 - Leave your network
 - Only packets with source address from your own address space
 - To enter/transit your network
 - Only source addresses from downstream customer address space

uRPF – Unicast Reverse Path

- Unicast Reverse Path Forwarding (uRPF)
 - Router verifies if the source address of incoming packets is in the Forwarding table and also checks the incoming interface
 - **Drop** if not!
 - ***Recommended on customer facing interfaces***

```
(config-if)#ipv6 verify unicast source reachable-via {rx | any}
```

uRPF – Unicast Reverse Path

- Modes of Operation (IOS):

- **Strict:** verifies both source address and incoming interface with entries in the forwarding table

- **Loose:** verifies existence of route to source address

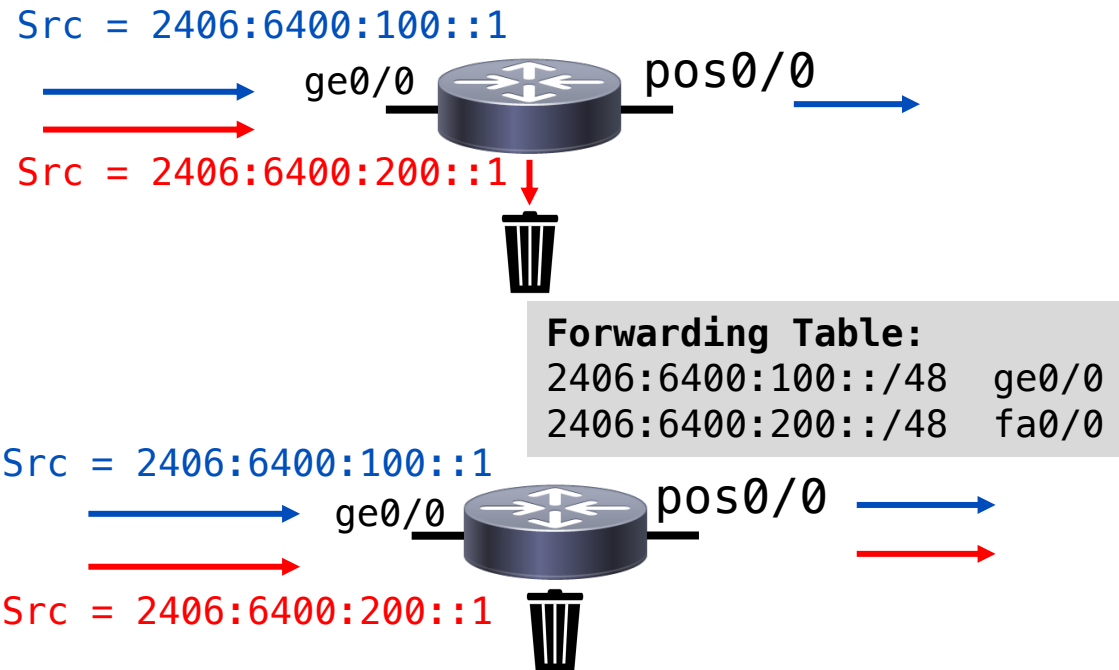


Image source: "Cisco ISP Essentials", Barry Greene & Philip Smith 2002

MANRS

- Mutually Agreed Norms of Routing Security
 - An ISOC led initiative to implement industry best practices to ensure security of routing system
 - <https://www.manrs.org/>
 - Inbound/outbound filtering – prefix/as-path
 - Source address validation – BCP38
 - Coordination – correct & up to date contacts
 - Validation – ROAs/IRR objects



Acknowledgement:

- Philip Smith
- Cisco Systems



Questions

