Don’t Break the Internet:

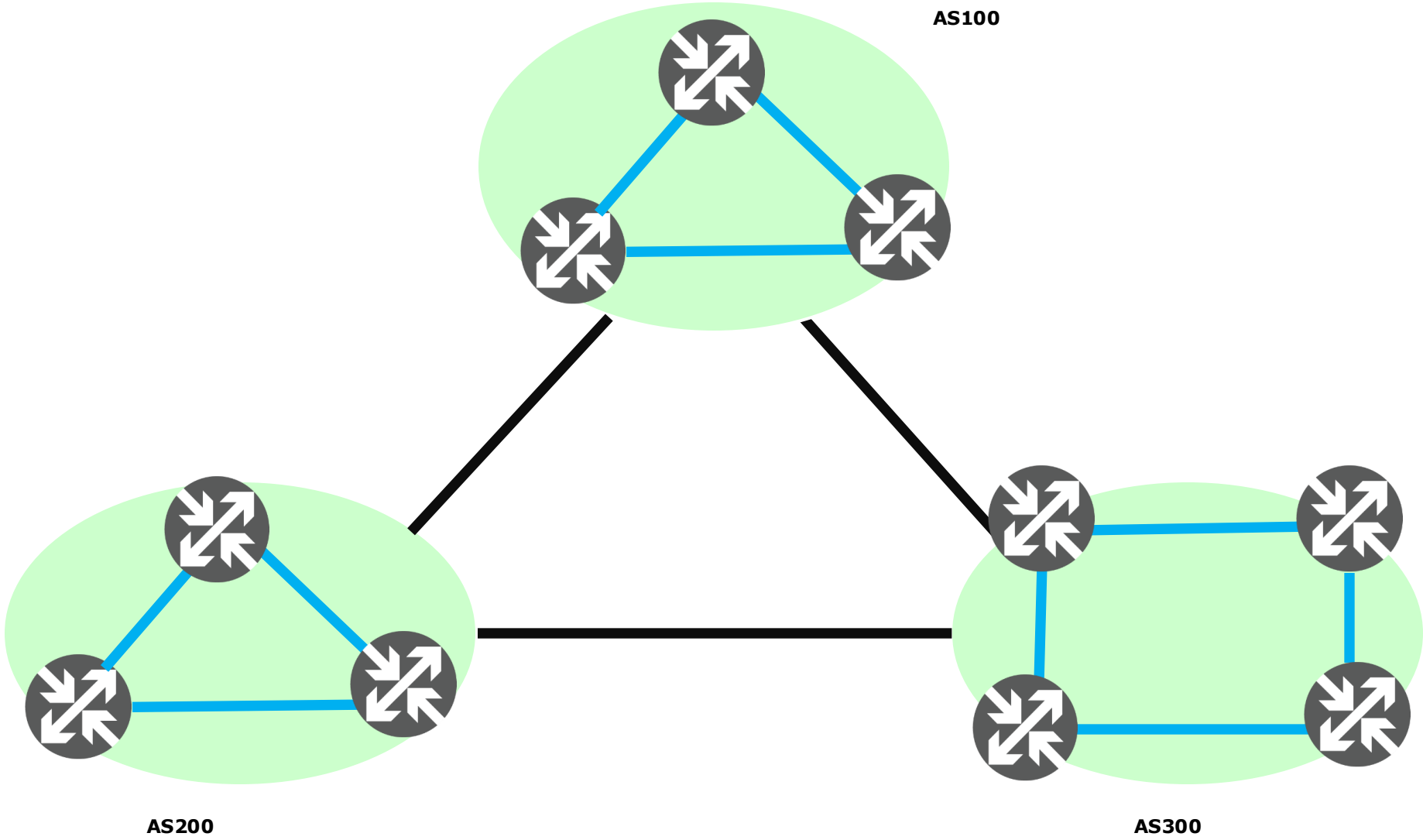
BGP Security Best Practices that Matter

Sheryl (Shane) Hermoso

Nov 2025

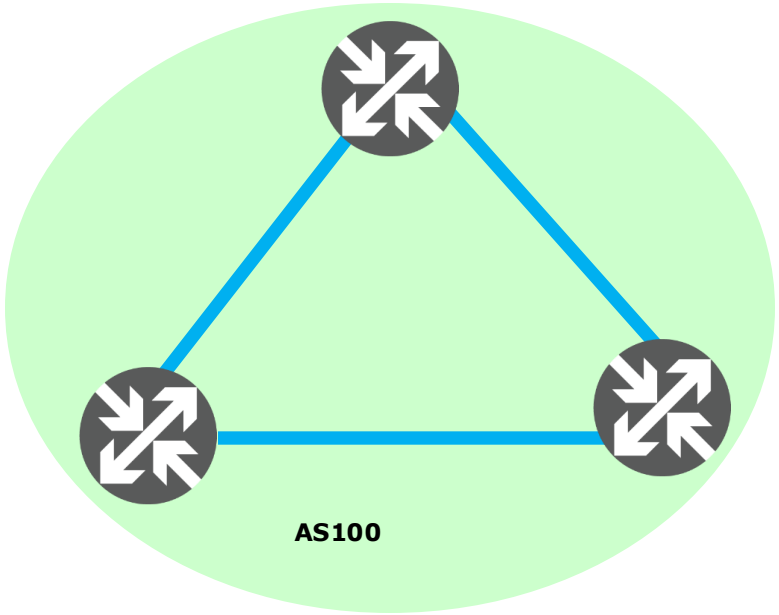
BGP in a nutshell

Interconnect Networks or ASes

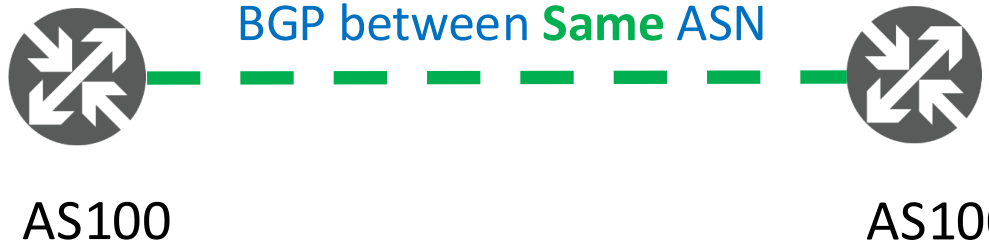


Autonomous System (AS)

Definition:
A network (or group of networks) under a single administrative control with uniform routing policies.



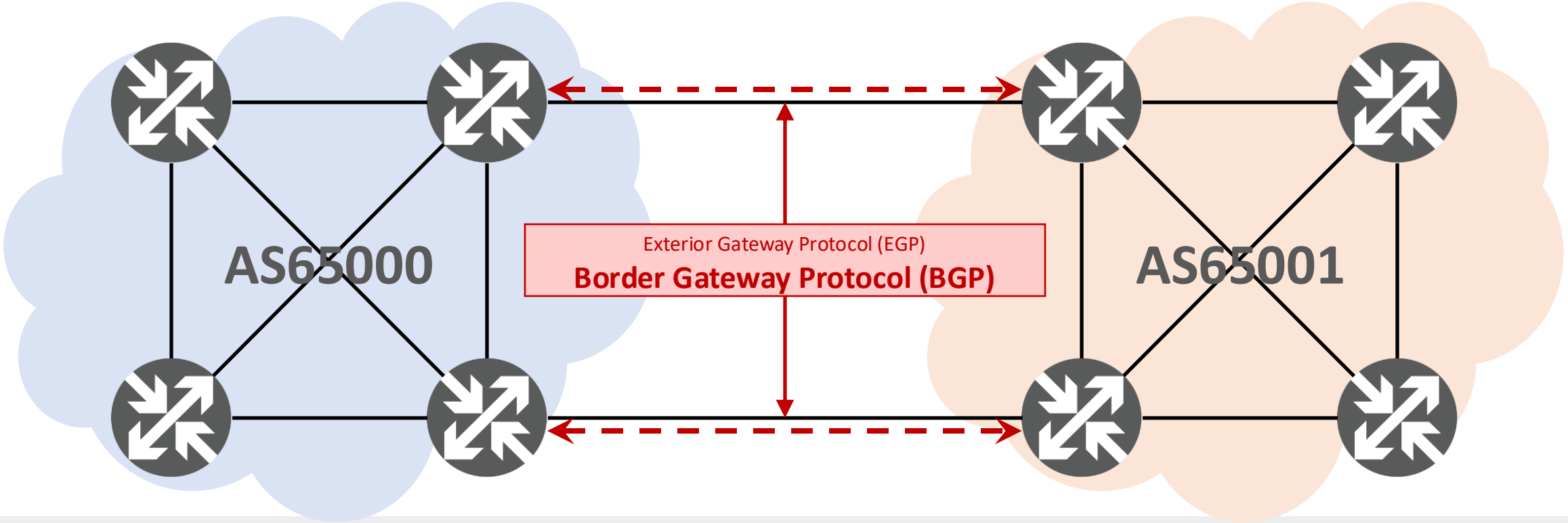
iBGP



eBGP

Border Gateway Protocol (BGP)

- An Exterior Gateway Protocol (EGP).
- Designed for exchanging routing information between **different Autonomous Systems**.



BGP Security Best Practices

Why BGP Best Practices Matter

- Misconfigurations or attacks can cause large-scale disruptions.

- **Prefix hijacking**, **route leaks**, **lack of filters**, **BGP session hijacking** are some ways to break the Internet

- BGP security practices:
  - Prevent service outages
  - Enhances global routing security
  - Protect your network … and OTHERS too
  - Ensure routing stability and trust

BGP Built-in Problem

- Routing works by RUMOUR
  - It is only based on trust, no built-in security

- Assumes everyone is correct (and honest)
  - No verification of the correctness of prefixes or AS paths

- No protection for message integrity or authenticity
- No validation of AS authority to announce routes
- No verification of path attribute authenticity.

BCP to address Vulnerabilities

According to RFC 7454 and supporting best practices:

- **Protect BGP Speakers**
  - Apply data-plane filters (e.g., uRPF) to block spoofed packets
  - Use control-plane ACLs to block unauthorized access to TCP port 179

- **Secure BGP Sessions**
  - Use MD5 (RFC 2385) or TCP-AO (RFC 5925) for peer authentication
  - Prefer TCP-AO over MD5 for stronger algorithms (e.g., HMAC-SHA1) and key rotation without session disruption

BCP to address Vulnerabilities

- **Route Filtering**
  - Implement prefix/AS-path filters (deny bogons routes)
  - Set max-prefix limits to prevent route floods
- **IRR Registration**
  - Maintain accurate routing data in IRR databases to support automated filtering (aut-num)
- **RPKI Validation**
  - Register routes in RPKI (ROA) to validate origin authenticity
  - Deploy ROV (Route Origin Validation) to reject invalid routes

#1 BGP Session Security

- Use MD5 or TTL security to protect BGP sessions.
- Prevents spoofing or session hijacking.
- Configure MD5 authentication or TTL hop limits.

Config Example – Block Spoofing (uRPF)

Strict Mode (for single-homed interfaces)

Cisco

```
interface GigabitEthernet0/0
 ip verify unicast source reachable-via rx
```

Juniper

```
set interfaces ge-0/0/0 unit 0 family inet rpf-check
```

Loose Mode (for multi-homed interfaces):

Cisco

```
interface GigabitEthernet0/0
 ip verify unicast source reachable-via any
```

Juniper

```
set interfaces ge-0/0/0 unit 0 family inet rpf-check mode loose
```

Config Example – Protect TCP-179

Cisco

```
ip access-list extended CoPP

 permit tcp host 10.10.10.2 host
192.168.1.1 eq 179  # Allow trusted peer

 deny tcp any any eq 179
# Block others


# Apply ACL to Control-Plane

!

control-plane

 service-policy input CoPP
```

Juniper

```
firewall {
  family inet {
    filter BGP-CONTROL-PLANE {
      term ALLOW-BGP-PEERS {
        from {
          source-address {
            10.10.10.2/32;   # Trusted peer
          }
          destination-port bgp;
        }
        then accept;
      }
      term BLOCK-OTHERS {
        then discard;
      }
    }
  }
}
apply:
set interfaces lo0 unit 0 family inet filter input
BGP-CONTROL-PLANE
```

Config Example – TCP-AO

Cisco

```
key chain BGP-TCPAO tcp
 key 1
  send-id 10
  recv-id 20
  cryptographic-algorithm aes-128-cmac
  key-string encrypted 066A0D020D1C470B1E
  send-lifetime 12:00:00 Mar 18 2025 infinite
 !
!
router bgp 65001
 neighbor 203.0.113.2
  remote-as 65002
  ao BGP-TCPAO include-tcp-options
  address-family ipv4 unicast
 !
!
```

** If TCP-AO is not supported, use MD5 **

Juniper

```
security {
    authentication-key-chains {
        key-chain BGP-TCPAO {
            key 1 {
                secret "$9$Hsd4QbvfhT6m";  # AES-128-CMAC-96 encrypted
                start-time "2025-03-18.12:00:00 +0000";
                algorithm ao;
                ao-attribute {
                    send-id 20;    # Matches Cisco's recv-id
                    recv-id 10;    # Matches Cisco's send-id
                    tcp-ao-option enabled;
                    cryptographic-algorithm aes-128-cmac-96;
                }
protocols {
    bgp {
        group EBGP {
            neighbor 203.0.113.1 {
                authentication-algorithm ao;
                authentication-key-chain BGP-TCPAO;
            }
        }
```

#2 Prefix and AS-Path Filtering

- Route filtering policies control which prefixes are advertised to or received from BGP peers

- RFC 2827/BCP 38 Ingress Packet Filtering

- **Prefix filters** only allow known, legitimate prefixes from peers/customers
  - This helps prevent route hijacks and leaks.

- **AS-path filters** restrict route advertisement using AS-PATH and BGP communities.
  - This helps avoid route leaks and policy violations.

Config Example - Filtering

Drop bogons when receiving and advertising **IPv4** & **IPv6**

```
Bogon IPv4 Prefixes:
    0.0.0.0/8 (This network)
    10.0.0.0/8 (Private-use networks)
    100.64.0.0/10 (Carrier-grade NAT)
    127.0.0.0/8 (Loopback)
    169.254.0.0/16 (Link-local)
    172.16.0.0/12 (Private-use networks)
    192.0.0.0/24 (IETF Protocol Assignments)
    192.0.2.0/24 (TEST-NET-1)
    192.168.0.0/16 (Private-use networks)
    198.18.0.0/15 (Benchmarking)
    198.51.100.0/24 (TEST-NET-2)
    203.0.113.0/24 (TEST-NET-3)
    224.0.0.0/4 (Multicast)
    240.0.0.0/4 (Reserved for future use)
    255.255.255.255/32 (Limited broadcast)
```

Additional considerations:

- Filter prefixes longer than /24 (0.0.0.0/0 ge 25)
- Include your own network ranges
- Regularly update this list as allocations change

Config Example – Prefix Filters

Cisco

```
! Create inbound prefix list for IPv4 Unicast EBGP with AS65000
ip prefix-list AS65000-IN seq 10 permit 0.0.0.0/0 le 24
ip prefix-list AS65000-IN seq 1000 deny 0.0.0.0/0 le 32

! Create outbound prefix list for IPv4 Unicast EBGP with AS65000
ip prefix-list, AS65000-OUT seq 10 permit 10.0.0.0/8
ip prefix-list AS65000-OUT seq 1000 deny 0.0.0.0/0 le 32

! Apply prefix filters to respective BGP peer
router bgp 64512
 address-family ipv4 unicast
  ! Keep a copy of original received prefixes
  neighbor EBGP-AS65000-IPV4 soft-reconfiguration inbound
  neighbor EBGP-AS65000-IPV4 prefix-list AS65000-IN in
  neighbor EBGP-AS65000-IPV4 prefix-list AS65000-OUT out
 exit-address-family
```

Juniper

```
# Create inbound policy for IPv4 Unicast EBGP with AS65000
top edit policy-options policy-statement EBGP-AS65000-IPV4-IN
    set term 10 from route-filter 0.0.0.0/0 upto /24
    set term 10 then accept
    set term 1000 then reject
# Re-create outbound policy for IPv4 Unicast EBGP with AS65000
top delete policy-options policy-statement EBGP-AS65000-IPV4-OUT
top edit policy-options policy-statement EBGP-AS65000-IPV4-OUT
    set term 10 from route-filter 10.0.0.0/8 exact
    set term 10 then accept
    set term 1000 then reject
# Apply routing policies to respective BGP peer
top edit protocols bgp
    set group EBGP-AS65000-IPV4 import EBGP-AS65000-IPV4-IN
    set group EBGP-AS65000-IPV4 export EBGP-AS65000-IPV4-OUT
```

Config Example – AS-Path Filters

Cisco

```
! Create inbound AS Path filter for
! IPv4 Unicast EBGP with AS65001
ip as-path access-list 101 permit ^65001(_65001)*$
ip as-path access-list 101 deny .*

! Create outbound AS Path filter for
! IPv4 Unicast EBGP with AS65001
ip as-path access-list 102 permit ^$
ip as-path access-list 102 permit ^64512(_64512)*$
ip as-path access-list 102 deny .*

! Apply AS Path filters to respective BGP peer
router bgp 65000
 address-family ipv4 unicast
  ! Keep a copy of original received prefixes
  neighbor EBGP-AS65001-IPV4 soft-reconfiguration
inbound
  neighbor EBGP-AS65001-IPV4 filter-list 101 in
  neighbor EBGP-AS65001-IPV4 filter-list 102 out
 exit-address-family
```

Juniper

```
# Create AS Path group for AS65001's IPv4 Unicast
paths
top edit policy-options as-path-group AS65001-IPV4
    set as-path 10 "^65001(65001)*$"

# Create AS Path group for locally originated IPv4
Unicast paths
top edit policy-options as-path-group LOCAL-IPV4
    set as-path 10 "^$"

# Create AS Path group for customer's IPv4 Unicast
paths
top edit policy-options as-path-group CUSTOMER-IPV4
    set as-path 10 "^64512(64512)*$"
```

#3 Max Prefixes on a Peering

- Set limits on the number of accepted prefixes per peer.
  - Logs warning messages on a configured threshold
  - Clears BGP session when prefixes exceeded the limit

- This helps prevents router overload from misconfigurations.

- Regular monitoring and adjustment may be required.

Config Example – Maximum-Prefix

Cisco

```
router bgp 65001
  neighbor 192.0.2.1 remote-as 65002
  neighbor 192.0.2.1 maximum-prefix 1000 80 60
```

Juniper

```
set protocols bgp group bgp-peer-in family inet unicast prefix-limit maximum 1000
set protocols bgp group bgp-peer-in family inet unicast prefix-limit maximum 2000
warning-only
set protocols bgp group bgp-peer-in family inet unicast prefix-limit maximum 5000
threshold 80 restart-interval 5
```

#4 Internet Routing Registry (IRR)

- IRR is a globally distributed routing information database
- Policies are stored using the routing policy specification language (RPSL)
- Filter out route advertisements not described in the registry

```
aut-num:        AS17821
as-name:        APNICTRAINING-ISP
descr:          ASN for APNICTRAINING LAB ISP
country:        AU
mp-import:      afi ipv6.unicast from AS65001 2406:6400:10::2 at
2406:6400:10::1 action community.append(17821:65001); pref=200;
accept <^AS65001+$> AND RS-APNICTRAINING:AS65001
mp-import:      afi ipv6.unicast from AS65002 2406:6400:14::2 at
2406:6400:14::1 action community.append(17821:65002); pref=200;
accept <^AS65002+$> AND RS-APNICTRAINING:AS65002
mp-export:      afi ipv6.unicast to AS65001 2406:6400:10::2 at
2406:6400:10::1 action aspath.prepend (AS17821,AS17821);
announce ANY AND NOT FLTR-MARTIAN-V6
mp-export:      afi ipv6.unicast to AS65002 2406:6400:14::2 at
2406:6400:14::1 announce ANY AND NOT FLTR-MARTIAN-V6
org:            ORG-ATU1-AP
admin-c:        AT480-AP
tech-c:         AT480-AP
abuse-c:        AA2310-AP
mnt-lower:      MAINT-AU-APNICTRAINING
mnt-lower:      MAINT-AU-APNICTRAINING
mnt-routes:     MAINT-AU-APNICTRAINING
mnt-by:         APNIC-HM
mnt-irt:        IRT-APNICTRAINING-AU
mnt-irt:        IRT-APNICTRAINING-AU
last-modified:  2021-01-26T23:58:34Z
source:         APNIC
```

Mp-import/export attribute contains inbound/outbound IPv4 or IPv6 routing policy of the AS.

Tools – bgpq4

- BGP filter generator
- Can generate configurations such as prefix-lists, extended access-lists, policy-statement terms and as-path lists based on IRR data

Cisco

```
# bgpq4 -l APNIC-IN AS4608

no ip prefix-list APNIC-IN
ip prefix-list APNIC-IN permit 103.0.0.0/16;
ip prefix-list APNIC-IN permit 103.246.136.0/22;
ip prefix-list APNIC-IN permit 110.76.136.0/22;
ip prefix-list APNIC-IN permit 202.12.29.0/24;
ip prefix-list APNIC-IN permit 203.30.127.0/24;
ip prefix-list APNIC-IN permit 203.30.127.0/24;
ip prefix-list APNIC-IN permit 203.119.76.0/23;
ip prefix-list APNIC-IN permit 203.119.92.0/23;
```

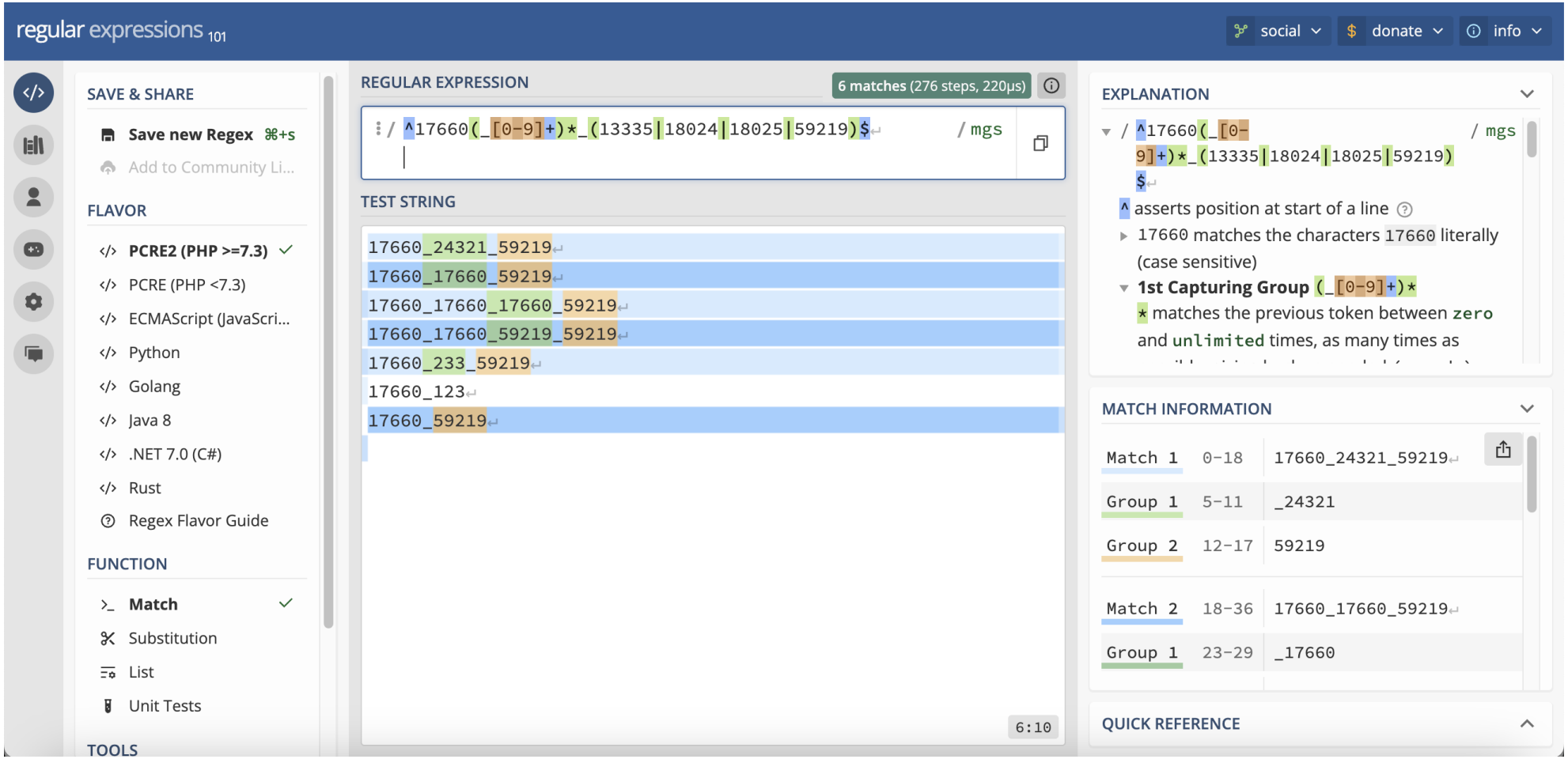
Juniper

```
# bgpq4 -Jl APNIC-IN AS4608

Policy-options {
Replace:
  prefix-list APNIC-IN {
         103.0.0.0/16;
         103.246.136.0/22;
         110.76.136.0/22;
         202.12.29.0/24;
         203.30.127.0/24;
         203.119.76.0/23;
         203.119.92.0/23;

  }
}
```

https://github.com/bgp/bgpq4

Tools – Regex101

- the ultimate regular expression tester



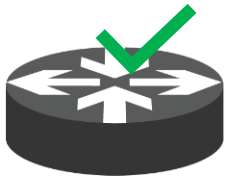
https://regex101.com/

#5 Route Origin Validation (ROV) Filtering

- Validate that routes come from authorized ASN.
- This helps prevent prefix hijacking.

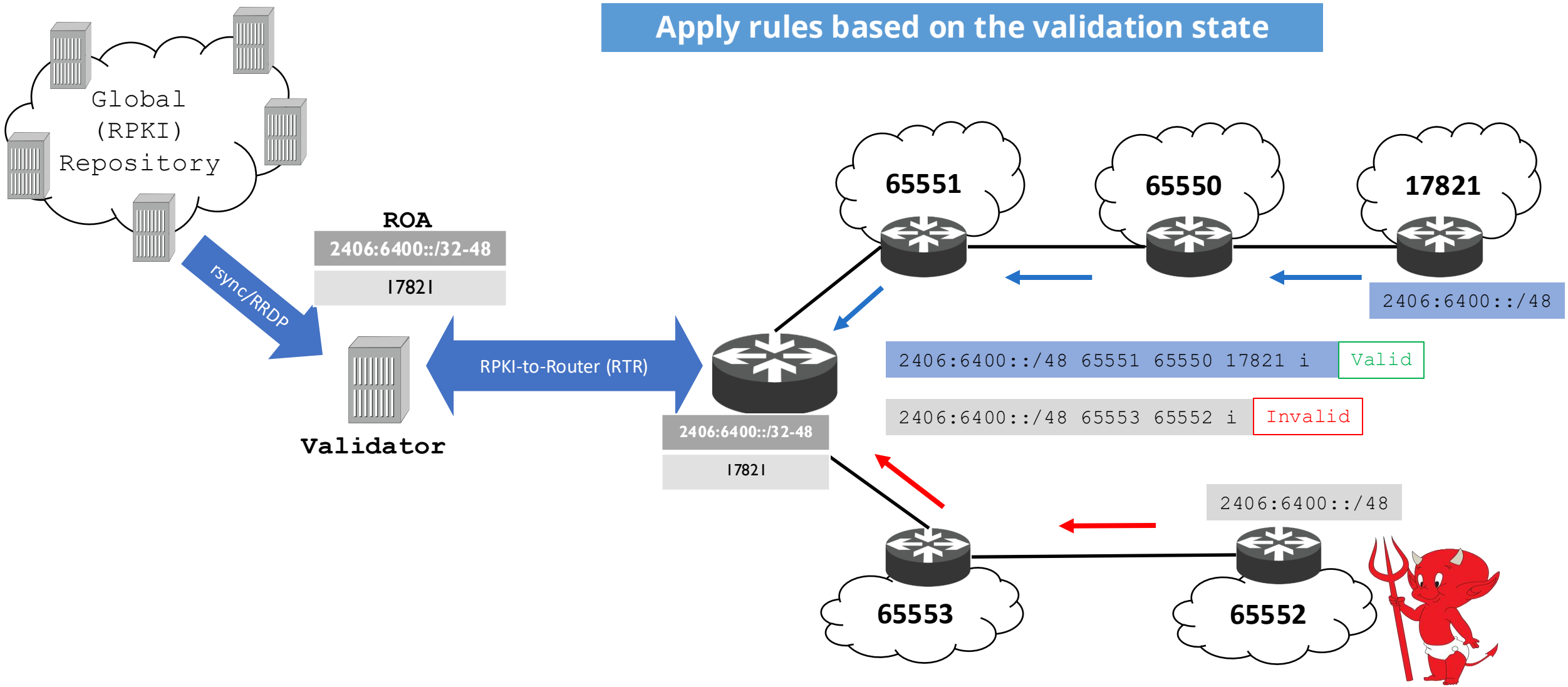
RPKI
robust security framework for verifying the association between resource holders and their Internet number resources

Phase 1: ROA (Signing origin)
Resource holders must create their ROA objects, which gets published to the RPKI repo



Phase 2: ROV (Validating origin)
Routers are validating route entries against the RPKI cache

Route Origin Validation (ROV)



Secure your BGP now!

- **Filter! Filter! Filter!**
- Review your prefix and AS-path filters.
- Set max-prefix limits.
- Deploy RPKI validation.
- Secure your BGP sessions.
- Monitor BGP announcements.

Thank You!


Questions