R&E ROUTING SECURITY
BEST PRACTICES

Grover Browning
Karl Newell
RFC 7454
BGP Operations & Security
Feb, 2015

Agenda

• Background / Community Development
• Overview of the RFC
• ACL’s on BGP Sessions
• Rate Limiters on BGP
• MD5/TCP-AO
• Internal Spoof Blocking
• GTSM
• To Dampen or not to Dampen
• Max-Prefixes
• AS-PATH Filtering
• Private AS Rejection
• First AS Rejection
• Advertising Private ASN’s
• Next Hop Filters
• Community Scrubbing
• In/Out Customer/Peer Filters
And most importantly …

Keeping Things Up to Date!
Community & Internet2

Backoffice and Community
ACL’s on BGP Sessions – filter TCP/179

- Good Peer
- Bad Guy
- Good Peer
- Bad Guy
- My Router
ACL’s on BGP Sessions  

JunOS Example

Lo0 {
    unit 0 {
        family inet {
            filter {
                input input-strict-in }}
    }
}

term allow-bgp {
    from {
        source-prefix-list {BGP-PEERS; }
        protocol tcp;
        port 179;}
    then {
        accept;}

prefix-list BGP-PEERS-ROUTING-INSTANCE {
    apply-path "protocols bgp group <*> neighbor <*>";
ACL’s on BGP Sessions – Cisco IOS XR

ipv4 access-list input-strict-in
   remark Only permit BGP session for peers
   permit tcp host 192.0.3.1 eq bgp any
   permit tcp host 192.0.3.1 any eq bgp
   deny tcp any eq bgp any
   deny tcp any any eq bgp
   ...
   deny any any

interface type interface-path-id
   ipv4 access-group input-strict-in ingress
Rate Limiters for BGP

Neighbor Router

Raubritter

My Routers RE

20 mbit TCP/179
Rate Limiters for BGP

JunOS Example

term limit-tcp-syn {
  from {
    source-prefix-list {
      BGP-PEERS;
      MSDP-PEERS;}
    protocol tcp;
    tcp-flags "(syn & !ack) | fin | rst";}
  then {
    policer 500K-drop;
    next term;}
}

policer 500K-drop {
  if-exceeding {
    bandwidth-limit 500k;
    burst-size-limit 62k;}
  then discard;}

Rate Limiters for BGP – Cisco IOS XR

lpts pifib hardware police
  flow bgp-cfg-peer 2000
  flow bgp-default 2500
  flow bgp-known 1500
MD5/TCP-AO

Secure Neighbor Router

7deca1987c2a945a a39067d476d2e353

Hey – look at all my routes and communities

Neighbor Router

I XP

My Router
MD5/TCP-AO

JunOS Example

```
bgp {
  group ext {
    type external;
    peer-as 65530;
    neighbor 172.16.2.1;
    authentication-key-chain bgp-auth;
    authentication-algorithm md5;
  }
  authentication-key-chains {
    key-chain bgp-auth {
      tolerance 30;
      key 0 {
        secret "$9$5TJDi....F6A"; ## SECRET-DATA
        start-time “2011-6-23.20:19:33 -0700”; }
      key 1 {
        secret "$9JGDiqW....puh."; ## SECRET-DATA
        start-time “2012-6-23.20:19:33 -0700”; }
  }
```
MD5/TCP-AO – Cisco IOS XR

MD5
router bgp 65500
  neighbor 192.0.2.1
    remote-as 65555
    password encrypted 123abc

TCP-AO
router bgp 65500
  neighbor 192.0.2.1
    remote-as 65555
    keychain bgp-auth

key chain bgp-auth
  key 0
  key-string $9$5TJDi....F6A
  cryptographic-algorithm [HMAC-MD5 | HMAC-SHA1-12]
  send-lifetime 20:19:33 june 23 2012 infinite
  accept-lifetime 20:19:33 june 23 2012 infinite
Spoof Blocking

Bad Guy

Bad Guy

TCP RST
Src: 192.0.2.140

TCP RST
Src: 192.0.2.23

My Router
192.0.2.0/24 for iBGP
filter connector-in {
    interface-specific;
    term border-exceptions {
        from {
            source-prefix-list {
                border-subnets;}
        then {
            accept;}}
    term border-discard {
        from {
            source-prefix-list {
                INTERNAL;}}
        then {
            syslog;
            next term;}}
    prefix-list border-subnets {
        apply-path "interfaces <*> unit <*> family inet address <*>";}
}
Spoof Blocking – Cisco IOS XR

```
ipv4 access-list input-strict-in
remark Deny internal iBGP spoofed sources
deny ip 192.0.2.0/24 any
permit tcp host 192.0.3.1 eq bgp any
permit tcp host 192.0.3.1 any eq bgp
deny tcp any eq bgp any
deny tcp any any eq bgp
... deny tcp any any eq bgp

deny any any

interface type interface-path-id
ipv4 access-group input-strict-in ingress
```
GTSM

Bad Guy

IP TTL 255

Intermediate Router

IP TTL 254

My Router
GTSM

bgp {
  group toAS2 {
    type external;
    peer-as 2;
    ttl 255;
    neighbor 10.1.2.3;
  }
}

JunOS Example

filter ttl-security {
  term gtsm {
    from {
      source-address {
        10.1.2.3/32; }
      protocol tcp;
      ttl-except 255;
      port 179; }
    then {
      discard; }
  }
  term else {
    then {
      accept;
    }
  }
}
GTSM – Cisco IOS XR

router bgp 65500
neighbor 192.0.2.1 ttl-security hops 1
To Dampen or not to Dampen – RFC 7196

Flaky Route
I'M HERE
My Router
Flaky Route
I'M NOT HERE
My Router

I'll ignore you for a while
byp {  
    damping;
group ext {  
    type external;
    import damp;
    export send-direct;
    neighbor 10.0.0.1 {  
        peer-as 100;
    }
    damping aggressive {
        half-life 15;
        suppress 6000;
    }
    damping timid {
        half-life 15;
        suppress 12000;
    }
    damping dry {
        disable;
    }
    policy-statement damp {
        term 1 {
            from {
            Route-filter 10.128.0.0/9 exact damping dry;
            route-filter 0.0.0.0/0 prefix-length-range /0/-/8 damping timid;
            route-filter 0.0.0.0/0 prefix-length-range /17/-/32 damping aggressive;
        }  
    }
To Dampen or not to Dampen – Cisco IOS XR

```
bgp dampening [ half-life [ reuse suppress max-suppress-time ] ]

router bgp 65500
  remote-as 65555
  address-family ipv4 unicast
    bgp dampening 15 750 6000 60
```
Max-Prefixes

Neighbor Router

18 IPv4 prefixes

My Router

Small neighbor that decides to transit for the Internet

628,332 IPv4 prefixes
Max-Prefixes

JunOS Example

```plaintext
Protocols {
  bgp {
    group CONNECTORS6 {
      type external;
      metric-out igp;
      family inet6 {
        unicast {
          prefix-limit {
            maximum 50;
            teardown 90;
          }
        }
      }
    }
  }
}
```
Max Prefixes – Cisco IOS XR

maximum prefix maximum [ threshold ]

router bgp 65500
neighbor 192.0.2.1
  remote-as 65555
  address-family ipv4 unicast
  maximum prefix 1000 90
First AS Rejection

AS 65555

Destination 192.168.0.0/16
AS Path 65555 65000

AS 65000

Destination 192.168.0.0/16
AS Path 65000

AS 65222

Destination 192.168.0.0/16
AS Path 65000

My Router
First AS Rejection – JunOS Example

set protocols bgp enforce-first-as
First AS Rejection – Cisco IOS XR

Cisco supports this by default. It can be disabled.

```
router bgp 65500
  bgp enforce-first-as disable
```
Advertising Private ASN

AS 65500

Destination: 192.0.2.0/24  AS Path: 65500

AS 209

Destination: 192.0.2.0/24  AS Path: 209 65500
Private ASNs

JunOS Example

```plaintext
policy-statement SANITY-IN {
  term block-private-asn {
    from as-path PRIVATE;
    then reject;
  }
  as-path PRIVATE ".* (64512-65535) .*";
}
```

!!Don’t Forget!!
32bit AS
Reject Routes with Private ASNs – Cisco IOS XR

route-policy check-as-PRIVATE
  if as-path passes-through (ios-regex `.* (64512-65535) .*`) then
drop
else
  pass
endif
end-policy

router bgp 65500
  neighbor 192.0.2.1
  address-family ipv4 unicast
  route-policy check-as-PRIVATE in
Remove Private ASNs  Outbound

JunOS Example

```plaintext
bgp {
    group ext {
        type external;
        neighbor 192.168.20.1 {
            remove-private;
            peer-as 200;
        }
    }
}
```

!!Don’t Forget!!
32bit AS
Remove Private ASNs Outbound – Cisco IOS XR

```
router bgp 65500
 neighbor 192.0.2.1 remove-private-as
```
**Next Hop Filters**

Set the Next Hop to that of the peer – ignore the advertised Next Hop (unless IXP or Blackhole injector).

<table>
<thead>
<tr>
<th>Destination</th>
<th>Next Hop</th>
<th>AS Path</th>
</tr>
</thead>
<tbody>
<tr>
<td>192.0.2.0/24</td>
<td>192.0.2.222</td>
<td>65500</td>
</tr>
</tbody>
</table>

My Router

AS 65500
192.0.2.1
Next Hop Filters

Protocols {
    bgp {
        neighbor 1.2.3.4 {
            IMPORT NEXT-HOP-CHECK;
        }
    }
}

policy-statement NEXT-HOP-CHECK {
    from
    protocol bgp;
    then {
        next-hop 1.2.3.4;
    }
}
Next Hop Filters – Cisco IOS XR

route-policy NEXT-HOP-CHECK ($nexthop)
  set next-hop $nexthop
end-policy

router bgp 65500
  neighbor 192.0.2.1
  address-family ipv4 unicast
    route-policy NEXT-HOP-CHECK(192.0.2.1) in
policy-statement REMOVE-COMMS-OUT {
    term remove {
        then {
            community delete DISCARD;
        }
    }
}

group FEDNET {
    type external;
    metric-out igp;
    export [REMOVE-COMMS-OUT];
}

community DISCARD members 11537:911;
community-set DISCARD
  11537:911
end-set

route-policy REMOVE-COMMS-OUT
  delete community in DISCARD
end-policy

router bgp 11537
  neighbor 192.0.2.1
  address-family ipv4 unicast
    route-policy REMOVE-COMMS-OUT out
Inbound Customer Filters

<table>
<thead>
<tr>
<th>Destination</th>
<th>AS Path</th>
</tr>
</thead>
<tbody>
<tr>
<td>192.0.2.0/24</td>
<td>65500</td>
</tr>
<tr>
<td>8.8.8.0/24</td>
<td>65500</td>
</tr>
</tbody>
</table>
neighbor 205.233.255.32 {
  description "[RE] UNY | I2-S1234";
  import [ SANITY-IN SET-PREF UNY-IN CONNECTOR-IN ];
  peer-as 1234;
}

policy-statement UNY {
  term participant {
    from {
      protocol bgp;
      prefix-list-filter UNY-PARTICIPANT orlonger;
    }
    then next policy;  
  }
  then reject;  
}

prefix-list UNY-PARTICIPANT {
  130.18.0.0/16;
  130.74.0.0/16;
}
Inbound Customer Filters – Cisco IOS XR

prefix-set MY-NEIGHBOR
  10.0.0.0/8 le 24,
  192.168.0.0/16 le 24
end-set

route-policy NEIGHBOR-PREFIX ($list)
  if destination in $list
    then pass
  else drop
endif
end-policy

router bgp 65500
  neighbor 192.0.2.1
  address-family ipv4 unicast
  route-policy NEIGHBOR-PREFIX(MY-NEIGHBOR) in
In/Out Filtering

AS 65500
192.0.2.1

Destination
192.0.2.0/24
10.0.0.0/8

AS Path
65500
65500

My Router
In/out Filtering

JunOS Example

route-filter 0.0.0.0/0 prefix-length-range /28-/32;

policy-statement SANITY-OUT {
  term block-martians {
     from {
       route-filter 0.0.0.0/0 exact;
       route-filter 10.0.0.0/8 orlonger;
       route-filter 127.0.0.0/8 orlonger;
       route-filter 169.254.0.0/16 orlonger;
       route-filter 172.16.0.0/12 orlonger;
       route-filter 192.0.2.0/24 orlonger;
       route-filter 192.88.99.1/32 exact;
       route-filter 192.168.0.0/16 orlonger;
       route-filter 198.18.0.0/15 orlonger;
       route-filter 224.0.0.0/4 orlonger;
       route-filter 240.0.0.0/4 orlonger;
     }
     then reject;
   }
}

export [ SANITY-OUT etc]
In/Out Filtering – Cisco IOS XR

prefix-set BOGONS
  0.0.0.0/0,
  0.0.0.0/8 le 32,
  10.0.0.0/8 le 32,
  127.0.0.0/8 le 32,
  169.254.0.0/16 le 32,
  172.16.0.0/12 le 32,
  192.0.0.0/24 le 32,
  192.0.2.0/24 le 32,
  192.168.0.0/16 le 32,
  198.18.0.0/15 le 32,
  198.51.100.0/24 le 32,
  203.0.113.0/24 le 32,
  224.0.0.0/4 le 32
end-set

route-policy Deny-Bogons
  if destination in BOGONS
    then drop
  else pass
endif
end-policy

router bgp 65500
  neighbor 192.0.2.1
    address-family ipv4 unicast
      route-policy Deny-Bogons
Keeping Things up to Date

You MUST keep the filters up to date!
Or else don’t do it!
Recent Updates to BGP

- [https://www.youtube.com/watch?v=V6Wsq66-f40&feature=youtu.be](https://www.youtube.com/watch?v=V6Wsq66-f40&feature=youtu.be)
- [https://pc.nanog.org/static/published/meetings/NANOG71/1425/20171003_Snijders_Recent_Bgp_Innovations_v1.pdf](https://pc.nanog.org/static/published/meetings/NANOG71/1425/20171003_Snijders_Recent_Bgp_Innovations_v1.pdf)

- RFC 7999 - BLACKHOLE Community, 65535:666
- RFC 8212 - Default External BGP (EBGP) Route Propagation Behavior without Policies
- RFC 8092 - BGP Large Communities Attribute
- RFC 8203 - BGP Administrative Shutdown Communication
The Internet2 Process...

- Security did an initial evaluation and prepared a paper.
- Architecture recommended which to implement
- Operations interpreted the recommendation & implemented.
RPKI

• Resource Public Key Infrastructure
• https://tools.ietf.org/html/rfc6480
• Interactive tutorials - http://www.securerouting.net/
• RPKI Session
Mind your MANRS

- Mutually Agreed Norms for Routing Security
- https://www.routingmanifesto.org/

1. Prevent propagation of incorrect routing information
2. Prevent traffic with spoofed source IP addresses
3. Facilitate global operational communication and coordination between network operators
4. Facilitate validation of routing information on a global scale
BCP 38 & 84

• BCP 38
  – Network Ingress Filtering

• BCP 84
  – Ingress Filtering for Multihomed Networks
uRPF

- unicast Reverse Path Forwarding

  - Strict – each incoming packet is tested against the FIB and if the incoming interface is not the best reverse path the packet check will fail

  - Feasible – the FIB maintains alternate routes to a given ip address. If the incoming interface matches with any of the routes associated with the ip address, then the packet is forwarded

  - Loose - each incoming packet's source address is tested against the FIB. The packet is dropped only if the source address is not reachable via any interface on that router