NETSENSE

ISPAB

Routing Security

Best Current Operational Practices

IGP Best Practices

- IGP carries infrastructure routes
  - Loopback and backbone P2P
  - Can also include IXP LAN block
- Do NOT carry customer route
  - Even if you assign IP to your customers
- Avoid route redistribution in IGP
  - If required, do it in a strictly controlled way
  - Route maps and policies can be used

BGP Best Practices

- IBGP is used to carry:
  - Full/partial Internet prefixes across backbone
  - Customer prefixes
- EBGP is used to:
  - Exchange prefixes with other ASes
  - Implement routing policy
- While configuring eBGP
  - Must configure Inbound and Outbound policy
  - Even if you don’t need any filter

Routing Hygene

- Use neighbour authentication for both IGP, iBGP and eBGP
- DO NOT:
  - Distribute BGP prefixes into an IGP
  - Distribute IGP routes into BGP
  - Use an IGP to carry customer prefixes

Border Router Best Practices

- Use inbound filter to reject unwanted routes from upstream
- Use outbound filter to advertise only your+customers prefix
- Originate default route in IGP
  - All other routers in the backbone will receive it
- If you have only one border router or you receive only default route from transits
  - No need to advertise routes towards your Core router
  - IGP will carry the default route that is originated at the border

IX Peering Router Best Practices

- Use separate router(s) for IX peering
  - Do not use your transit router to peer with IX
- Originating routes from peering router is NOT recommended
  - Originate from Core
- Configure default route towards NULL
  - Blackhole any traffic other then your own and of your customers
- Carry IX LAN prefix within your infra using IGP
  - Configure the link as passive interface
- Use BGP filters
  - Inbound: Reject default route and accept all other
  - Outbound: Only permit your own and customers’ prefixes

Static Route Towards Customer Router

- Interface flap will result in prefix withdraw and reannounce
  - Configure persistent route configuration
  - Cisco:

```
ip route 100.100.1.0 255.255.255.0 172.16.1.2 permanent
```

  - Juniper:

```
set static route 100.100.1.0/24 next-hop 172.16.1.2 passive
```

Pull-up Route for BGP Advertisement

- Many ISPs redistribute static routes into BGP rather than using the network statement
- Better to use pull-up route
  - Can discurd traffic to unused destination IP
  - Cisco:

```
R1(config)# ip route 100.100.1.0 255.255.255.0 null0
R1(config-router-af)# network 100.100.1.0 255.255.255.255
```

  - Juniper:

```
set static route 100.100.1.0/24 discard
set policy-options policy-statement net-out term static from
route-filter 100.100.1.0/24 exact
set policy-options policy-statement net-out term statics then
accept
set protocols bgp export net-out
```

NETSENSE

NULL Route for Aggregated Prefix

- If the aggregated prefix is further divided into subnets and not all the subnets are in use
  - That might enable processing of packets towards unused addresses
  - Traffic coming from outside might face routing loops for unused destinations
- If BGP is used, NULL route for aggregated prefix should already be in place
  - See the previous slide for “Pull-up route”

Bogon Route Filtering (IPv4): Cisco

```
ip prefix-list in-filter deny <Your own prefixes> le 32
ip prefix-list in-filter deny 0.0.0.0/8 le 32
ip prefix-list in-filter deny 10.0.0.0/8 le 32
ip prefix-list in-filter deny 100.64.0.0/10 le 32
ip prefix-list in-filter deny 127.0.0.0/8 le 32
ip prefix-list in-filter deny 169.254.0.0/16 le 32
ip prefix-list in-filter deny 172.16.0.0/12 le 32
ip prefix-list in-filter deny 192.0.0.0/24 le 32
ip prefix-list in-filter deny 192.0.2.0/24 le 32
ip prefix-list in-filter deny 192.168.0.0/16 le 32
ip prefix-list in-filter deny 198.18.0.0/15 le 32
ip prefix-list in-filter deny 198.51.100.0/24 le 32
ip prefix-list in-filter deny 203.0.113.0/24 le 32
ip prefix-list in-filter deny 224.0.0.0/3 le 32
ip prefix-list in-filter deny 0.0.0.0/0 ge 25
ip prefix-list in-filter permit 0.0.0.0/0 le 24
```

Bogon Route Filtering (IPv4): Juniper

```
set policy-options policy-statement ebgp-martian term reserved from route-filter <your own prefixes> exact reject
set policy-options policy-statement ebgp-martian term reserved from route-filter 0.0.0.0/8 orlonger reject
set policy-options policy-statement ebgp-martian term reserved from route-filter 10.0.0.0/8 orlonger reject
set policy-options policy-statement ebgp-martian term reserved from route-filter 127.0.0.0/8 orlonger reject
set policy-options policy-statement ebgp-martian term reserved from route-filter 128.0.0.0/16 orlonger reject
set policy-options policy-statement ebgp-martian term reserved from route-filter 172.16.0.0/12 orlonger reject
set policy-options policy-statement ebgp-martian term reserved from route-filter 191.255.0.0/16 orlonger reject
set policy-options policy-statement ebgp-martian term reserved from route-filter 192.0.2.0/24 orlonger reject
set policy-options policy-statement ebgp-martian term reserved from route-filter 223.255.255.0/24 orlonger reject
set policy-options policy-statement ebgp-martian term reserved from route-filter 224.0.0.0/3 orlonger reject
```

Bogon/Martian Route Filtering (IPv6)

```
ipv6 prefix-list v6in-filter deny <Your own prefixes> le 128
ipv6 prefix-list v6in-filter permit 64:ff9b::/96
ipv6 prefix-list v6in-filter deny 2001::/23 le 128
ipv6 prefix-list v6in-filter deny 2001:2::/48 le 128
ipv6 prefix-list v6in-filter deny 2001:10::/28 le 128
ipv6 prefix-list v6in-filter deny 2001:db8::/32 le 128
ipv6 prefix-list v6in-filter deny 2002::/16 le 128
ipv6 prefix-list v6in-filter deny 3ffe::/16 le 128
ipv6 prefix-list v6in-filter permit 2000::/3 le 48
ipv6 prefix-list v6in-filter deny ::/0 le 128
```

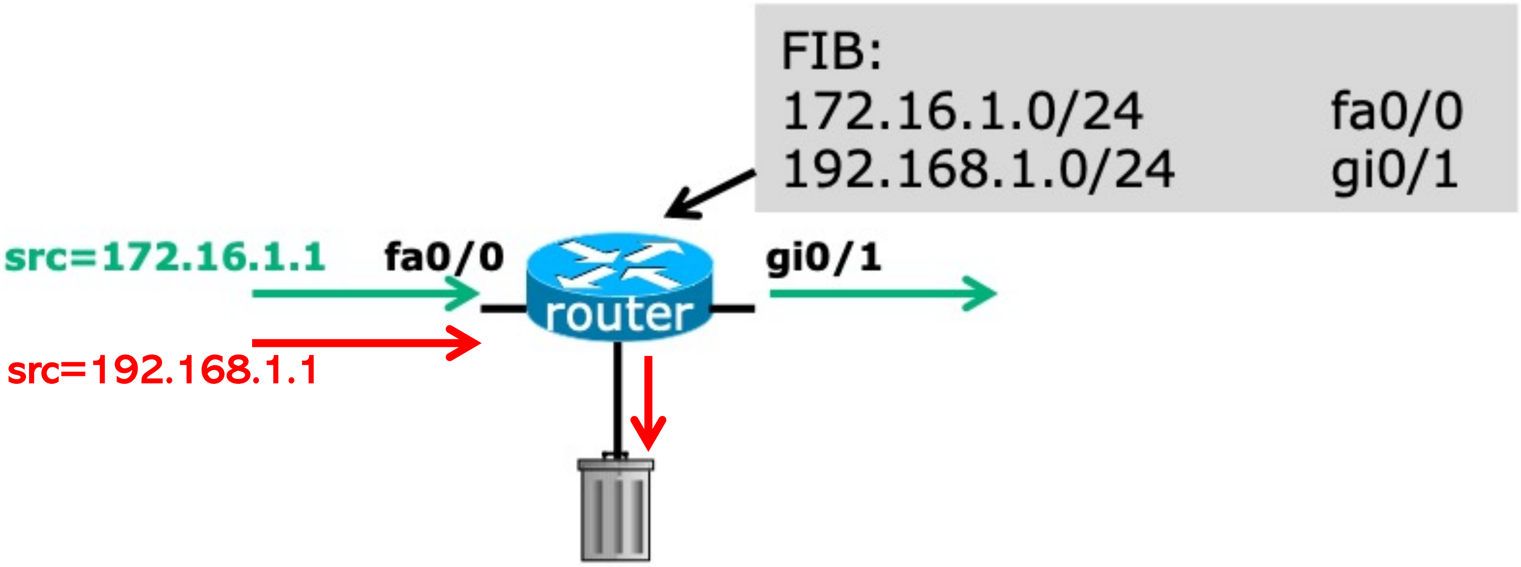
Redistribution Example

```
ip route 100.64.0.0 255.255.255.248 Serial 5/0 permanent
!
router bgp 100
 address-family ipv4
  redistribute static route-map static-to-bgp
<snip>
!
route-map static-to-bgp permit 10
 match ip address prefix-list ISP-block
 set origin igp
<snip>
!
ip prefix-list ISP-block permit 100.64.0.0/26 le 31
```

uRPF/BCP38

- Unicast Reverse Path Forwarding
- There are two modes for uRPF:
  - Strict Mode
    - Source address must be reachable via the source (incoming) interface
    - Typically used in Access Networks
  - Loose Mode
    - Source address must be in the FIB
    - Typically used to drop non-routed address space
    - Also can be used when asymmetric traffic flows are present (for example, when multihoming)

uRPF: Strict Mode



FIB:

172.16.1.0/24	fa0/0
192.168.1.0/24	gi0/1

src=172.16.1.1

src=192.168.1.1

fa0/0

router

gi0/1

- Router compares source address of incoming packet with FIB entry
  - If FIB entry interface matches incoming interface, the packet is forwarded
  - If FIB entry interface does not match incoming interface, the packet is dropped

uRPF Config Example: Cisco

Strict mode:

```
ip verify unicast source reachable-via rx allow-self-ping
```

Loose mode:

```
ip verify unicast source reachable-via any allow-self-ping
```

uRPF Config Example: Juniper

Strict Mode:

```
[edit]
interfaces {
    so-0/0/0 {
        unit 0 {
            family inet {
                rpf-check;
} } } }
```

Loose Mode:

```
[edit]
interfaces {
    so-0/0/0 {
        unit 0 {
            family inet {
                rpf-check;
                mode loose;
} } } }
```

Too Long AS Path in BGP: Limit Max AS

Cisco:

```
neighbor x.x.x.x maxas-limit 10
```

Juniper:

```
set policy-options policy-statement block-very-long-paths term
LONG-AS-PATH from as-path too-many-AS

set policy-options policy-statement block-very-long-paths term
LONG-AS-PATH then reject

set policy-options as-path too-many-AS ".{10,}”

set protocols bgp group external-bgp import block-very-long-paths
```

BGP Max Prefix Limit

Cisco:

```
neighbor <x.x.x.x> maximum-prefix <max> [restart N]
[<threshold>] [warning-only]
```

Juniper:

```
accepted-prefix-limit {
    maximum number;
    teardown <percentage-threshold> idle-timeout
    (forever | minutes);
}
```

Remove Private AS from BGP AS Path

Cisco:

```
neighbor <x.x.x.x> remove-private-as
```

Juniper:

```
set protocols bgp group external set neighbor <x.x.x.x>
remove-private
```

`as-override` can be used instead of removing the private AS for cases where the private AS is replaced with a Public ASN.

Multihop TTL Security

Cisco:

```
neighbor <x.x.x.x> multi-hop 5
```

Juniper:

```
bgp {
     group external-peers {
          type external;
          neighbor x.x.x.x {
               multihop ttl 5;
          }
     }
}
```

GTSM

Cisco:

```
neighbor <x.x.x.x> ttl-security hops 1
```

Juniper:

```
filter ttl-security {
  term gtsm {
    from {
        source-address {
          x.x.x.x/32;
        }
        protocol tcp;
        ttl-except 255;
        port 179;
    }
    then {
        discard;
    }
  }
  term else {
    then {
        accept; } } }
```

```
ge-1/0/0 {
  unit 0 {
    family inet {
       filter {
         input gtsm;
       }
    }
  }
}
```

OSPF Authentication

Cisco:

```
interface GigabitEthernet2/0
 ip ospf authentication message-digest
 ip ospf message-digest-key 1 md5 MYPASSWORD
```

Juniper:

```
set protocols ospf area 0.0.0.0 interface so-0/2/0
authentication md5 5 key MYPASSWORD
```

BGP Authentication

Cisco:

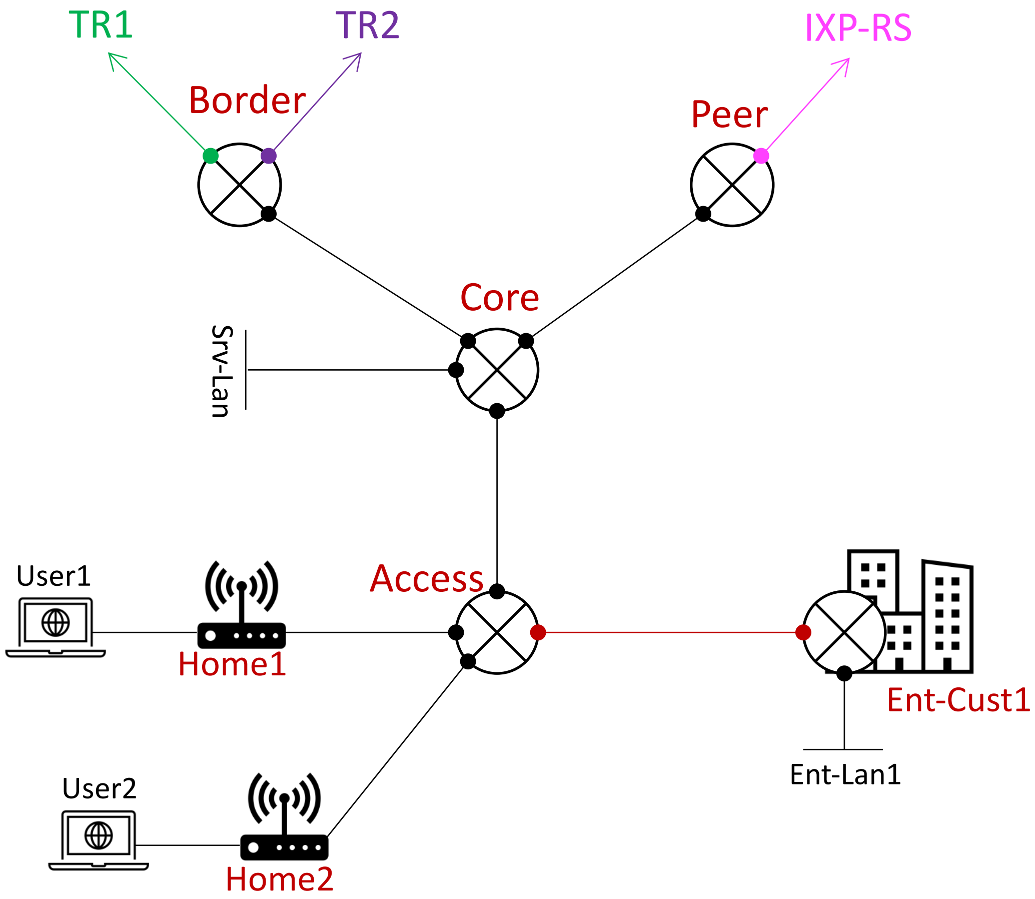
```
neighbor 192.0.2.70 password MySecretPassword
```

Juniper:

```
[edit security authentication-key-chains key-chain bgp-auth]
set key 0 secret this-is-the-secret-password
set key 0 start-time 2011-6-23.20:19:33-0700
set key 1 secret this-is-another-secret-password
set key 1 start-time 2012-6-23.20:19:33-0700

[edit protocols bgp group external]
set authentication-key-chain bgp-auth
set authentication-algorithm md5
```

Routing BCP Summary



NETSENSE

Questions?