Troubleshooting BGP with Juniper Examples

Joseph M. Soricelli (jms@juniper.net)

NANOG 27, Phoenix, Arizona

Caveats and Assumptions

- The views presented here are those of the author and they do not necessarily represent the views of Juniper Networks
- You will ask a question when you don't understand!
- Other things:
  - Basic knowledge of BGP operation and attributes
  - Understanding of the JUNOS software CLI
  - Some screen captures were modified to fit on the slide
  - All configurations and captures were gathered using JUNOS software version 5.5R2.3

Troubleshooting Mentality

- **Impossible to present an “official” troubleshooting methodology**
  - **Based on experiences**
  - **Take a logical approach (if there is time)**
  - **Shotgun troubleshooting?**
- **Use the tools available from the router**
  - **CLI `show` commands**
  - **Configuration parameters**
  - **Use of `traceoptions` files**

Agenda: Troubleshooting BGP

- ➔ Originating Routes
- Filtering Routes
  - AS Path
  - Prefixes
  - Communities
- Reference Slides

Default Advertisement Rules

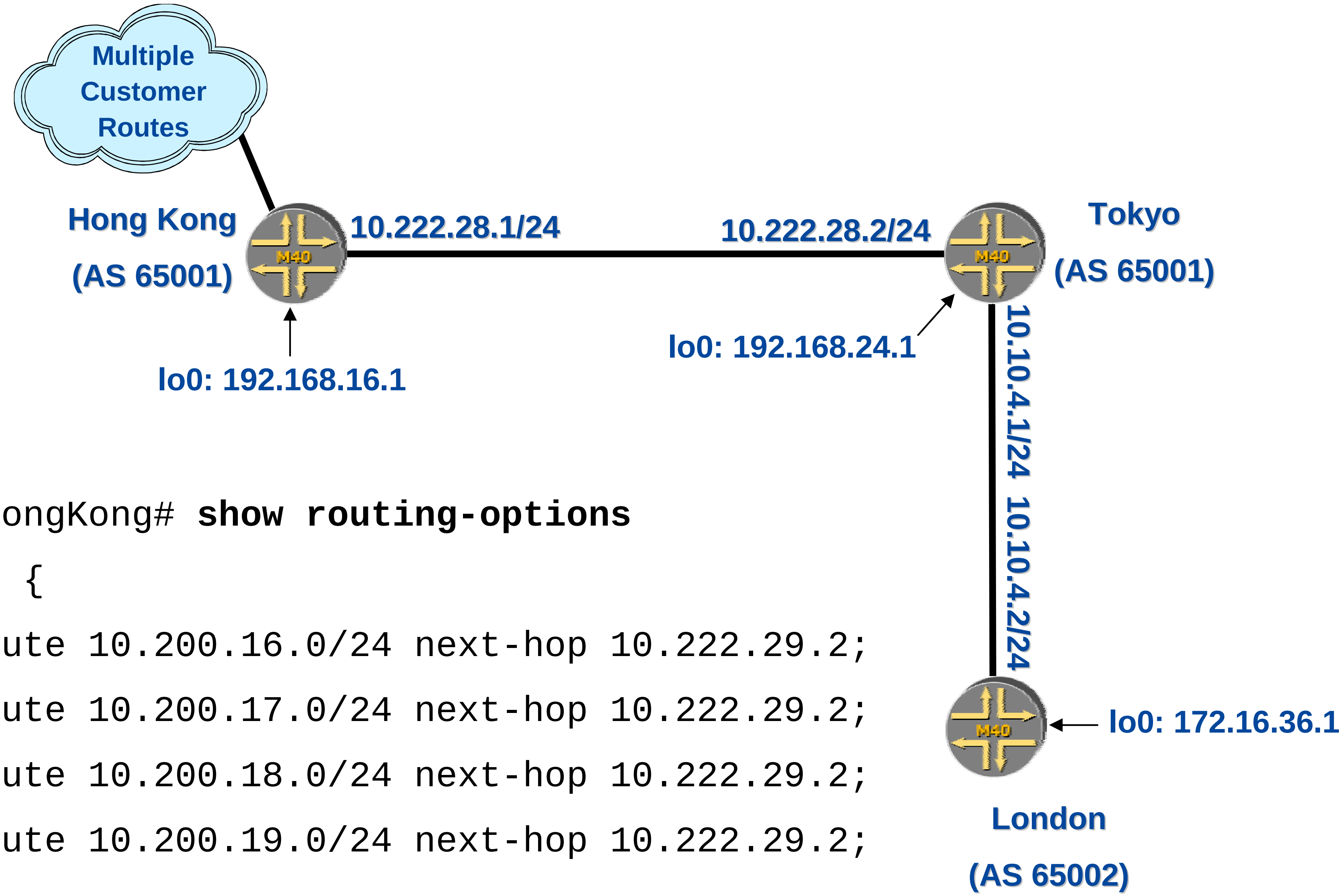
- Advertise only the *active* BGP routes to peers

```
user@HongKong> show route protocol bgp

inet.0: 43 destinations, 43 routes (43 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

192.168.24.0/24    *[BGP/170] 00:00:10, MED 20, localpref 100, from 192.168.24.1
                      AS path: I
                    > to 10.222.28.2 via fe-0/0/0.0
192.168.25.0/24    *[BGP/170] 00:00:10, MED 20, localpref 100, from 192.168.24.1
                      AS path: I
                    > to 10.222.28.2 via fe-0/0/0.0
192.168.26.0/24    *[BGP/170] 00:00:10, MED 20, localpref 100, from 192.168.24.1
                      AS path: I
                    > to 10.222.28.2 via fe-0/0/0.0
192.168.27.0/24    *[BGP/170] 00:00:10, MED 20, localpref 100, from 192.168.24.1
                      AS path: I
                    > to 10.222.28.2 via fe-0/0/0.0
```

Originating Routes

- Any change to the default BGP advertisement rules is accomplished with a routing policy
  - Common framework and language used throughout the JUNOS software
  - Constructed using sets of match and action pairs
- Policy is used to inject new routing information
  - Static routes to customers
  - Locally configured null routes
    - Use `discard` or `reject` for a next-hop option
  - IGP learned routes

Originating Routes — Sample Network



```
[edit]
user@HongKong# **show routing-options**
static {
    route 10.200.16.0/24 next-hop 10.222.29.2;
    route 10.200.17.0/24 next-hop 10.222.29.2;
    route 10.200.18.0/24 next-hop 10.222.29.2;
    route 10.200.19.0/24 next-hop 10.222.29.2;
}
autonomous-system 65001;
```

Advertise the Customer Routes

- **Policy is configured on Hong Kong to send all static routes**

```
user@HongKong> show configuration policy-options
policy-statement send-customer-routes {
    term all-customers {
        from protocol static;
        then accept;
    }
}
user@HongKong> show configuration protocols bgp
group Internal-Peers {
    type internal;
    local-address 192.168.16.1;
    authentication-key "$9$Qeioz/tu0IcrvBIwgJDmPBIEhSe"; # SECRET-DATA
    export send-customer-routes;
    neighbor 192.168.24.1;
}
```

Check for Routes (1 of 2)

- **Multiple methods on the receiving router for verifying the policy worked!**

```
user@Tokyo> show bgp summary
Groups: 2 Peers: 2 Down peers: 0
Table          Tot Paths  Act Paths Suppressed    History Damp State    Pending
inet.0                 4          4          0          0          0          0
Peer               AS      InPkt     OutPkt    OutQ   Flaps Last Up/Dwn State
172.16.36.1     65002        164        168       0       1       49:04 0/0/0
192.168.16.1    65001         88         90       0       0       43:47 4/4/0

user@Tokyo> show route terse protocol bgp source-gateway 192.168.16.1
inet.0: 29 destinations, 29 routes (29 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both
A Destination        P Prf   Metric 1   Metric 2  Next hop        AS path
* 10.200.16.0/24     B 170        100            >10.222.28.1     I
* 10.200.17.0/24     B 170        100            >10.222.28.1     I
* 10.200.18.0/24     B 170        100            >10.222.28.1     I
* 10.200.19.0/24     B 170        100            >10.222.28.1     I
```

Check for Routes (2 of 2)

- The `show route` command has other popular methods

```
user@HongKong> show route advertising-protocol bgp 192.168.24.1
inet.0: 28 destinations, 28 routes (28 active, 0 holddown, 0 hidden)
  Prefix                     Nexthop             MED     Lclpref    AS path
* 10.200.16.0/24             10.222.29.2                 100        I
* 10.200.17.0/24             10.222.29.2                 100        I
* 10.200.18.0/24             10.222.29.2                 100        I
* 10.200.19.0/24             10.222.29.2                 100        I

user@Tokyo> show route receive-protocol bgp 192.168.16.1
inet.0: 29 destinations, 29 routes (29 active, 0 holddown, 0 hidden)
  Prefix                     Nexthop             MED     Lclpref    AS path
* 10.200.16.0/24             10.222.29.2                 100        I
* 10.200.17.0/24             10.222.29.2                 100        I
* 10.200.18.0/24             10.222.29.2                 100        I
* 10.200.19.0/24             10.222.29.2                 100        I
```

Summarize Your Routes

- **Tokyo would like to summarize all routes in AS 65001 towards London**

```
[edit]
user@Tokyo# show policy-options
policy-statement send-aggregate-route {
    term send-aggregate {
        from protocol aggregate;
        then accept;
    }
}

[edit]
user@Tokyo# set routing-options aggregate route 10.200/16
user@Tokyo# set routing-options aggregate route 10.222/16

[edit protocols bgp group External-AS65002]
user@Tokyo# set export send-aggregate-route
```

Aggregation Problems

- **The aggregate routes are sent, but so are some of the more-specific contributing routes**

```
user@Tokyo> show route advertising-protocol bgp 172.16.36.1

inet.0: 31 destinations, 31 routes (31 active, 0 holddown, 0 hidden)

  Prefix                   Nexthop              MED     Lclpref    AS
path

* 10.200.0.0/16            Self                                    I

* 10.200.16.0/24           Self                                    I

* 10.200.17.0/24           Self                                    I

* 10.200.18.0/24           Self                                    I

* 10.200.19.0/24           Self                                    I

* 10.222.0.0/16            Self                                    I
```

- ***What happened here?***

Don’t Forget the Defaults!

- Remember that the default BGP policy advertises all active BGP routes
  - The more-specific contributing routes in our case

```
user@Tokyo> show route protocol bgp terse

inet.0: 31 destinations, 31 routes (31 active, 0 holddown, 0 hidden)

+ = Active Route, - = Last Active, * = Both


A Destination        P Prf   Metric 1   Metric 2  Next hop        AS
path

* 10.200.16.0/24     B 170        100            >10.222.28.1     I

* 10.200.17.0/24     B 170        100            >10.222.28.1     I

* 10.200.18.0/24     B 170        100            >10.222.28.1     I

* 10.200.19.0/24     B 170        100            >10.222.28.1     I
```

Modify the Policy

- **Add a term to the policy that rejects (doesn’t send) the more-specific contributing routes**

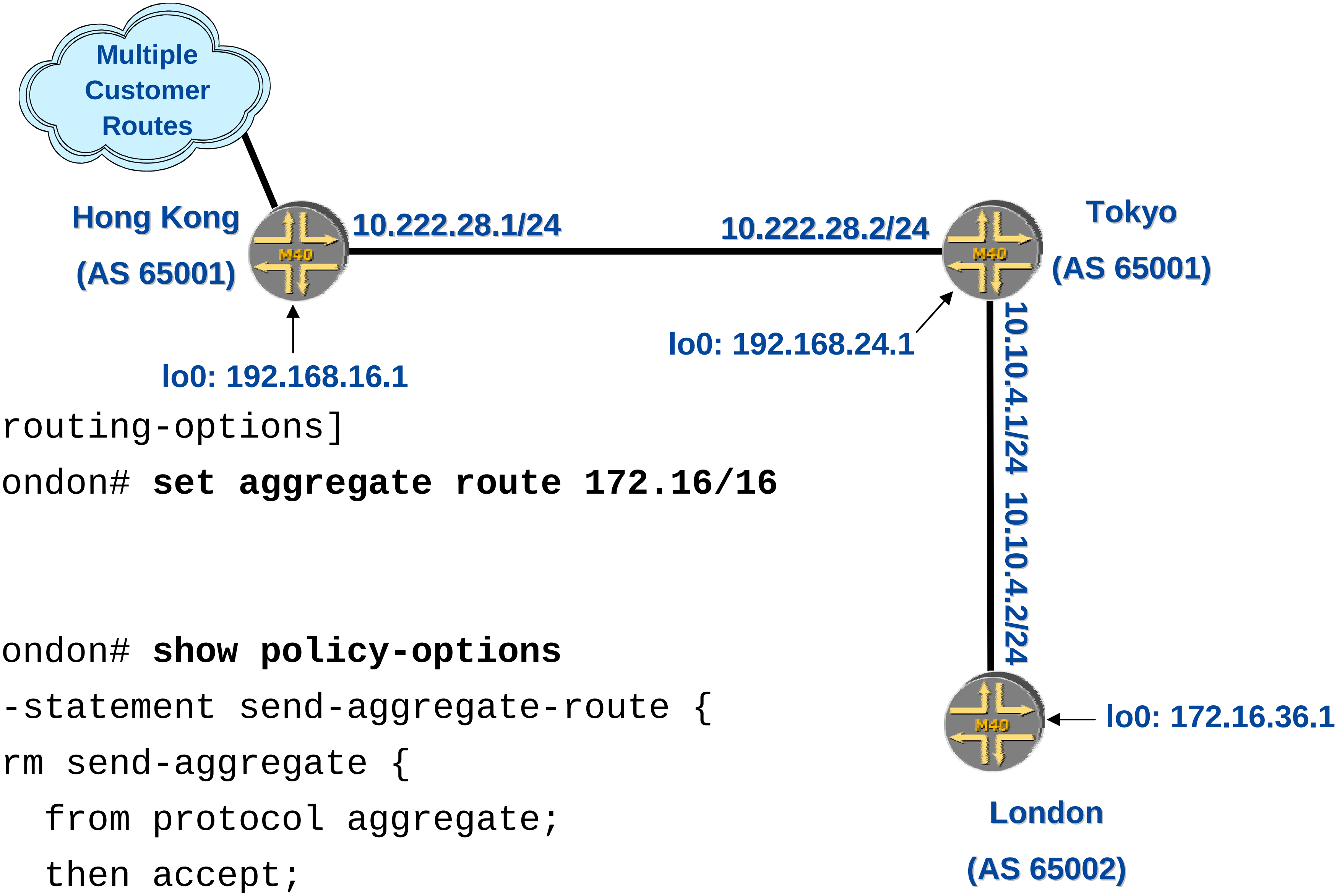
```
[edit policy-options policy-statement send-aggregate-route]
user@Tokyo# set term suppress-specifics from route-filter 10.200/16 longer
user@Tokyo# set term suppress-specifics then reject

[edit policy-options policy-statement send-aggregate-route]
user@Tokyo# show
term send-aggregate {
    from protocol aggregate;
    then accept;
}
term suppress-specifics {
    from {
        route-filter 10.200.0.0/16 longer;
    }
    then reject;
}
```

Successful Aggregation

- **Only the aggregate routes are now sent**

```
user@Tokyo> show route advertising-protocol bgp 172.16.36.1
inet.0: 31 destinations, 31 routes (31 active, 0 holddown, 0 hidden)
  Prefix                    Nexthop              MED     Lclpref    AS path
* 10.200.0.0/16             Self                                    I
* 10.222.0.0/16             Self                                    I

user@London> show route receive-protocol bgp 192.168.24.1
inet.0: 16 destinations, 16 routes (16 active, 0 holddown, 0 hidden)
  Prefix                    Nexthop              MED     Lclpref    AS path
* 10.200.0.0/16             192.168.24.1                            65001 I
* 10.222.0.0/16             192.168.24.1                            65001 I
```

Originating Routes — Sample Network



```
[edit routing-options]
user@London# set aggregate route 172.16/16

[edit]
user@London# show policy-options
policy-statement send-aggregate-route {
    term send-aggregate {
        from protocol aggregate;
        then accept;
    }
}
```

Problems in AS 65001

- **Tokyo received the 172.16/16 route from London**
  - **Sent it to Hong Kong**

```
user@Tokyo> show route receive-protocol bgp 172.16.36.1
inet.0: 32 destinations, 32 routes (32 active, 0 holddown, 0 hidden)
  Prefix                   Nexthop              MED     Lclpref    AS path
* 172.16.0.0/16            172.16.36.1                             65002 I
```

```
user@Tokyo> show route terse protocol bgp 172.16/16
inet.0: 32 destinations, 32 routes (32 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

A Destination        P Prf   Metric 1   Metric 2  Next hop        AS path
* 172.16.0.0/16      B 170        100            >10.10.4.2       65002 I
```

```
user@Tokyo> show route advertising-protocol bgp 192.168.16.1
inet.0: 32 destinations, 32 routes (32 active, 0 holddown, 0 hidden)
  Prefix                   Nexthop              MED     Lclpref    AS path
* 172.16.0.0/16            172.16.36.1                  100        65002 I
```

Problems in AS 65001

- **Hong Kong doesn’t see the route**
  - **We have some clues**

```
user@HongKong> show route receive-protocol bgp 192.168.24.1
inet.0: 29 destinations, 29 routes (28 active, 0 holddown, 1 hidden)

user@HongKong> show route terse protocol bgp 172.16/16
inet.0: 29 destinations, 29 routes (28 active, 0 holddown, 1 hidden)

user@HongKong> show bgp summary
Groups: 1 Peers: 1 Down peers: 0
Table          Tot Paths  Act Paths Suppressed    History Damp State    Pending
inet.0                 1          0          0          0          0          0
Peer               AS      InPkt     OutPkt    OutQ   Flaps Last Up/Dwn State
192.168.24.1    65001       2205       2206       0       0    18:22:04 0/1/0
```

Next Hop Problems

- **The BGP Next Hop is currently set to 172.16.36.1**
  - **Loopback address of London**
  - **Hong Kong doesn’t have a route to that address**

```
user@HongKong> show route hidden extensive
inet.0: 29 destinations, 29 routes (28 active, 0 holddown, 1 hidden)
172.16.0.0/16 (1 entry, 0 announced)
         BGP    Preference: 170/-101
                Next hop type: Unusable
                State: <Hidden Int Ext>
                Local AS: 65001 Peer AS: 65001
                Age: 10:32
                Task: BGP_65001.192.168.24.1+1067
                AS path: 65002 IAggregator: 65002 172.16.36.1
                Localpref: 100
                Router ID: 192.168.24.1
                Indirect next hops: 1
                        Protocol next hop: 172.16.36.1 Indirect next hop: 0 -
```

Next Hop Resolution

- Tokyo alters the BGP Next Hop before advertising the route to Hong Kong
  - Other methods are available

```
[edit]
user@Tokyo# show policy-options policy-statement next-hop-self
term set-nh {
    then {
        next-hop self;
    }
}
[edit]
user@Tokyo# set protocols bgp group Internal-Peers export next-hop-self

user@Tokyo> show route advertising-protocol bgp 192.168.16.1
inet.0: 30 destinations, 30 routes (30 active, 0 holddown, 0 hidden)
  Prefix                   Nexthop              MED     Lclpref    AS path
* 172.16.0.0/16            Self                         100        65002 I
```

Next Hop Resolution

- **Hong Kong now has a usable route**
  - **Loopback address of Tokyo is now the BGP Next Hop**

```
user@HongKong> show route receive-protocol bgp 192.168.24.1
inet.0: 29 destinations, 29 routes (29 active, 0 holddown, 0 hidden)
  Prefix                   Nexthop              MED     Lclpref    AS path
* 172.16.0.0/16            192.168.24.1                 100        65002 I

user@HongKong> show route terse protocol bgp
inet.0: 29 destinations, 29 routes (29 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

A Destination        P Prf   Metric 1   Metric 2  Next hop        AS path
* 172.16.0.0/16      B 170        100            >10.222.28.2     65002 I
```

Troubleshooting Commands

- `show configuration protocols bgp`
- `show bgp summary`
- `show route advertising-protocol bgp` *neighbor*
  - Routes sent by BGP to a specific peer
- `show route receive-protocol bgp` *neighbor*
  - Routes received by BGP from a specific peer
- `show route protocol bgp`
  - All BGP routes installed in the `inet.0` routing table
- `show route hidden` *extensive*
  - All hidden routes in `inet.0`
  - Routes not usable due to BGP Next Hop problems
  - Routes filtered by an inbound `route-filter`

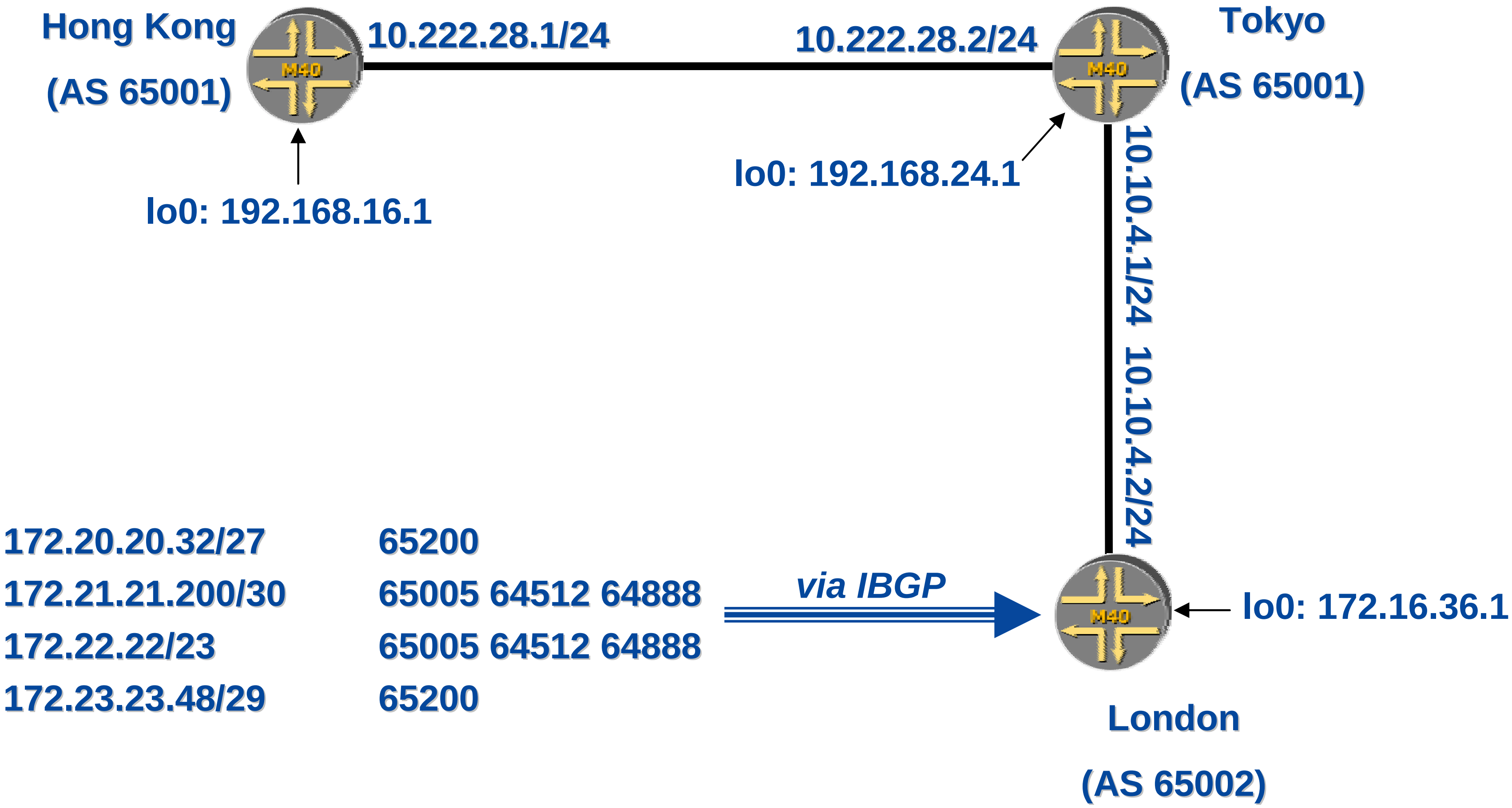
Agenda: Troubleshooting BGP

- Originating Routes
- ➔ Filtering Routes
  - ➔ AS Path
  - Prefixes
  - Communities
- Reference Slides

Filtering and Modifying Routes

- The JUNOS software routing policy language is also used to filter IP prefixes
  - Use a `route-filter` statement to find the appropriate routes
- AS Path regular expressions are used to match routes in a policy
  - Define the regular expression
  - Use the `from as-path` syntax in the policy
- BGP communities are modified using policies
  - Add new communities using `add` or `set` commands
  - Delete existing communities using the `delete` command
- All appropriate BGP attributes are sent by default for all routes
  - This includes any current community values

Filtering Routes — Sample Network



Hong Kong

(AS 65001)

10.222.28.1/24

10.222.28.2/24

Tokyo

(AS 65001)

lo0: 192.168.16.1

lo0: 192.168.24.1

10.10.4.1/24

10.10.4.2/24

172.20.20.32/27	65200
172.21.21.200/30	65005 64512 64888
172.22.22/23	65005 64512 64888
172.23.23.48/29	65200

via IBGP

lo0: 172.16.36.1

London

(AS 65002)

Filtering Routes — AS Path

- London would like to filter all routes from the peer AS of 65200
  - Currently all routes are being sent to Tokyo
  - BGP default policy is advertising them

```
user@London> show route advertising-protocol bgp 192.168.24.1
inet.0: 21 destinations, 21 routes (21 active, 0 holddown, 0 hidden)
  Prefix                   Nexthop           MED     Lclpref    AS path
* 10.200.0.0/16            Self                                 65001 I
* 10.222.0.0/16            Self                                 65001 I
* 172.16.0.0/16            Self                                 I
* 172.20.20.32/27          Self                                 65200 I
* 172.21.21.200/30         Self                                 65005 64512 64888 I
* 172.22.22.0/23           Self                                 65005 64512 64888 I
* 172.23.23.48/29          Self                                 65200 I
```

Filtering Routes — AS Path

- Create a policy that matches on all routes from the AS 65200 peer and reject them
  - Define the regular expression by name
  - Reference the regular expression name in the policy

```
[edit]
user@London# show policy-options
policy-statement filter-on-AS-Path {
    term filter-peer-AS65200 {
        from as-path peer-AS65200;
        then reject;
    }
}
as-path peer-AS65200 "65200 .*";
```

Filtering Routes — AS Path

- **Apply the *filter-on-AS-Path* policy to the appropriate peer group**
  - **Ensure that it is the first policy used by altering the order with the insert command**

```
[edit protocols bgp group External-AS65001]
user@London# show
type external;
local-address 172.16.36.1;
export [ filter-on-AS-Path send-aggregate-route ];
peer-as 65001;
neighbor 192.168.24.1 {
    multihop {
        ttl 2;
    }
}
```

Filtering Routes — AS Path

- **Routes from AS 65200 are no longer sent**

```
user@London> show route advertising-protocol bgp 192.168.24.1
inet.0: 21 destinations, 21 routes (21 active, 0 holddown, 0 hidden)
  Prefix                     Nexthop           MED     Lclpref     AS path
* 10.200.0.0/16              Self                                  65001 I
* 10.222.0.0/16              Self                                  65001 I
* 172.16.0.0/16              Self                                  I
* 172.21.21.200/30           Self                                  65005 64512 64888 I
* 172.22.22.0/23             Self                                  65005 64512 64888 I

user@Tokyo> show route receive-protocol bgp 172.16.36.1
inet.0: 32 destinations, 32 routes (32 active, 0 holddown, 0 hidden)
  Prefix                     Nexthop       MED   Lclpref     AS path
* 172.16.0.0/16              172.16.36.1                     65002 I
* 172.21.21.200/30           172.16.36.1                     65002 65005 64512 64888 I
* 172.22.22.0/23             172.16.36.1                     65002 65005 64512 64888 I
```

Filtering Routes — AS Path

- **London now wants to also restrict all IBGP learned routes**
  - **Only advertise the “nailed up” summary route**
  - **We have an existing policy, so let’s modify it**
  - **“( )” is the same as “^$”**

```
[edit]
user@London# show policy-options
policy-statement filter-on-AS-Path {
    term filter-peer-AS65200 {
        from as-path [ peer-AS65200 ibgp-null-AS-Path ];
        then reject;
    }
}
as-path peer-AS65200 "65200 .*";
as-path ibgp-null-AS-Path “()”;
```

← **Logical OR**

Filtering Routes — AS Path

- **Out configuration doesn’t work right**
  - **The 172.16/16 summary route is not sent**

```
user@London> show route advertising-protocol bgp 192.168.24.1
inet.0: 21 destinations, 21 routes (21 active, 0 holddown, 0 hidden)
  Prefix                   Nexthop           MED     Lclpref    AS path
* 10.200.0.0/16            Self                                 65001 I
* 10.222.0.0/16            Self                                 65001 I
* 172.21.21.200/30         Self                                 65005 64512 64888 I
* 172.22.22.0/23           Self                                 65005 64512 64888 I
```

```
user@Tokyo> show route receive-protocol bgp 172.16.36.1
inet.0: 31 destinations, 31 routes (31 active, 0 holddown, 0 hidden)
  Prefix                   Nexthop         MED   Lclpref  AS path
* 172.21.21.200/30         172.16.36.1                    65002 65005 64512 64888 I
* 172.22.22.0/23           172.16.36.1                    65002 65005 64512 64888 I
```

Filtering Routes — AS Path

- **The current order of the policies is no longer appropriate**
  - **The aggregate route has a Null AS path and it is being rejected by the *`filter-on-AS-Path`* policy**

```
user@London> show route 172.16/16 exact detail
inet.0: 21 destinations, 21 routes (21 active, 0 holddown, 0 hidden)
172.16.0.0/16 (1 entry, 1 announced)
        *Aggregate Preference: 130
                Next hop type: Reject
                State: <Active Int Ext>
                Task: Aggregate
                AS path: I (LocalAgg)
                AS path list:
                AS path: I  Refcount: 2
                Contributing Routes (2):
                        172.16.36.1/32       proto Direct
                        172.16.32.1/32       proto OSPF
```

Filtering Routes — AS Path

- **Use the insert command to change the order of the policies**

```
[edit protocols bgp group External-AS65001]
user@London# show
type external;
local-address 172.16.36.1;
export [ filter-on-AS-Path send-aggregate-route ];
peer-as 65001;
neighbor 192.168.24.1 {
(Information deleted)
user@London# insert export send-aggregate-route before filter-on-AS-Path
user@London# show
type external;
local-address 172.16.36.1;
export [ send-aggregate-route filter-on-AS-Path ];
peer-as 65001;
neighbor 192.168.24.1 {
(Information deleted)
```

Filtering Routes — AS Path

- **All appropriate routes are now sent**

```
user@London> show route advertising-protocol bgp 192.168.24.1
inet.0: 21 destinations, 21 routes (21 active, 0 holddown, 0 hidden)
  Prefix                   Nexthop           MED     Lclpref    AS path
* 10.200.0.0/16            Self                                 65001 I
* 10.222.0.0/16            Self                                 65001 I
* 172.16.0.0/16            Self                                 I
* 172.21.21.200/30         Self                                 65005 64512 64888 I
* 172.22.22.0/23           Self                                 65005 64512 64888 I

user@Tokyo> show route receive-protocol bgp 172.16.36.1
inet.0: 32 destinations, 32 routes (32 active, 0 holddown, 0 hidden)
  Prefix                   Nexthop       MED   Lclpref    AS path
* 172.16.0.0/16            172.16.36.1                    65002 I
* 172.21.21.200/30         172.16.36.1                    65002 65005 64512 64888 I
* 172.22.22.0/23           172.16.36.1                    65002 65005 64512 64888 I
```

Agenda: Troubleshooting BGP

- Originating Routes
- ➔ Filtering Routes
  - AS Path
  - ➔ Prefixes
  - Communities
- Reference Slides

Filtering Routes — Prefixes

- AS 65001 does not want to receive any routes with a subnet mask longer than /24
  - It is currently receiving one such route from London

```
user@Tokyo> show route receive-protocol bgp 172.16.36.1
inet.0: 32 destinations, 32 routes (32 active, 0 holddown, 0 hidden)
  Prefix                   Nexthop       MED    Lclpref    AS path
* 172.16.0.0/16            172.16.36.1                     65002 I
* 172.21.21.200/30         172.16.36.1                     65002 65005 64512 64888 I
* 172.22.22.0/23           172.16.36.1                     65002 65005 64512 64888 I
```

Filtering Routes — Prefixes

- **Create a policy that rejects the unwanted route**

```
[edit]
user@Tokyo# show policy-options
policy-statement filter-bad-prefixes {
    term bad-AS-65002-routes {
        from {
            route-filter 172.21.21.200/30 exact;
        }
        then reject;
    }
}

[edit protocols bgp group External-AS65002]
user@Tokyo# set import filter-bad-prefixes
```

Filtering Routes — Prefixes

- **The filter appears to work perfectly**

```
user@Tokyo> show route receive-protocol bgp 172.16.36.1
inet.0: 32 destinations, 32 routes (31 active, 0 holddown, 1 hidden)
  Prefix                   Nexthop       MED   Lclpref    AS path
* 172.16.0.0/16            172.16.36.1                    65002 I
* 172.22.22.0/23           172.16.36.1                    65002 65005 64512 64888 I

user@Tokyo> show route hidden
inet.0: 32 destinations, 32 routes (31 active, 0 holddown, 1 hidden)
+ = Active Route, - = Last Active, * = Both

172.21.21.200/30    [BGP ] 01:22:33, localpref 100, from 172.16.36.1
                      AS path: 65002 65005 64512 64888 I
                    > to 10.10.4.2 via fe-0/0/1.0
```

Filtering Routes — Prefixes

- **However, the *`filter-bad-prefixes`* policy is not very scalable**
  - **Better to create a policy that rejects all possible routes with a mask greater then /24**
  - **Replace the current `route-filter` to use the `prefix-length-range` option**

```
[edit]
user@Tokyo# show policy-options
policy-statement filter-bad-prefixes {
    term no-more-than-24-bits {
        from {
            route-filter 0.0.0.0/0 prefix-length-range /25-/32;
        }
        then reject;
    }
}
```

Filtering Routes — Prefixes

- **The generic filter still rejects the 172.21.21.200/30 route but other prefixes as well**

```
user@Tokyo> show route receive-protocol bgp 172.16.36.1
inet.0: 34 destinations, 34 routes (31 active, 0 holddown, 3 hidden)
  Prefix                   Nexthop       MED    Lclpref    AS path
* 172.16.0.0/16            172.16.36.1                     65002 I
* 172.22.22.0/23           172.16.36.1                     65002 65005 64512 64888 I

user@Tokyo> show route hidden terse
inet.0: 34 destinations, 34 routes (31 active, 0 holddown, 3 hidden)
+ = Active Route, - = Last Active, * = Both

A Destination        P Prf   Metric 1   Metric 2  Next hop    AS path
  172.21.21.200/30   B            100            >10.10.4.2  65002 65005 64512 64888 I
  172.24.24.64/29    B            100            >10.10.4.2  65002 65005 64512 64888 I
  172.25.25.128/25   B            100            >10.10.4.2  65002 65005 64512 64888 I
```

Agenda: Troubleshooting BGP

- Originating Routes
- ➔ Filtering Routes
  - AS Path
  - Prefixes
  - ➔ Communities
- Reference Slides

Filtering Routes — Communities

- **London doesn’t want routes with 64321:1234 community**

```
user@London> show route receive-protocol bgp 192.168.24.1 detail
inet.0: 24 destinations, 24 routes (24 active, 0 holddown, 0 hidden)
* 10.200.0.0/16 (1 entry, 1 announced)
     Nexthop: 192.168.24.1
     AS path: 65001 I Aggregator: 65001 192.168.24.1
 Communities: 65001:1001
* 10.222.0.0/16 (1 entry, 1 announced)
     Nexthop: 192.168.24.1
     AS path: 65001 I Aggregator: 65001 192.168.24.1
 Communities: 65001:1001
* 10.244.0.0/16 (1 entry, 1 announced)
     Nexthop: 192.168.24.1
     AS path: 65001 I
 Communities: 64321:1234 65001:1001
```

Filtering Routes — Communities

- **Create a policy that rejects the unwanted route**
  - **Create a community name and reference it in the policy**

```
[edit]
user@London# show policy-options
policy-statement filter-on-community {
    term nothing-with-1234 {
        from community AS64321-community;
        then reject;
    }
}
community AS64321-community members 64321:1234;

[edit protocols bgp group External-AS65001]
user@London# set import filter-on-community
```

Filtering Routes — Communities

- **The filter appears to work**

```
user@London> show route receive-protocol bgp 192.168.24.1
inet.0: 24 destinations, 24 routes (23 active, 0 holddown, 1 hidden)
  Prefix                   Nexthop              MED     Lclpref    AS path
* 10.200.0.0/16            192.168.24.1                            65001 I
* 10.222.0.0/16            192.168.24.1                            65001 I

user@London> show route hidden
inet.0: 24 destinations, 24 routes (23 active, 0 holddown, 1 hidden)
+ = Active Route, - = Last Active, * = Both

10.244.0.0/16       [BGP ] 00:05:13, localpref 100, from 192.168.24.1
                      AS path: 65001 I
                    > to 10.10.4.1 via fe-0/0/0.0
```

Filtering Routes — Communities

- **London would also like to remove the current communities on the routes received from Tokyo**
  - **Modify the *`filter-on-community`* policy**

```
[edit]
user@London# show policy-options
policy-statement filter-on-community {
    term remove-AS65001 {
        from community AS65001-community;
        then {
            community delete AS65001-community;
        }
    }
    term nothing-with-1234 {
        from community AS64321-community;
        then reject;
    }
}
community AS64321-community members 64321:1234;
community AS65001-community members 65001:1001;
```

Filtering Routes — Communities

- **The policy appears correct, but it doesn’t appear that anything has changed**

```
user@London> show route receive-protocol bgp 192.168.24.1 detail
inet.0: 24 destinations, 24 routes (23 active, 0 holddown, 1 hidden)
* 10.200.0.0/16 (1 entry, 1 announced)
     Nexthop: 192.168.24.1
     AS path: 65001 I Aggregator: 65001 192.168.24.1
 Communities: 65001:1001

* 10.222.0.0/16 (1 entry, 1 announced)
     Nexthop: 192.168.24.1
     AS path: 65001 I Aggregator: 65001 192.168.24.1
 Communities: 65001:1001
```

Filtering Routes — Communities

- **Let's view things from a different perspective**

```
user@London> show route protocol bgp source-gateway 192.168.24.1
inet.0: 24 destinations, 24 routes (23 active, 0 holddown, 1 hidden)
+ = Active Route, - = Last Active, * = Both
10.200.0.0/16      *[BGP/170] 1d 00:05:00, localpref 100, from 192.168.24.1
                      AS path: 65001 I
                    > to 10.10.4.1 via fe-0/0/0.0
10.222.0.0/16      *[BGP/170] 1d 00:05:00, localpref 100, from 192.168.24.1
                      AS path: 65001 I
                    > to 10.10.4.1 via fe-0/0/0.0

user@London> show route community-name AS65001-community
inet.0: 24 destinations, 24 routes (23 active, 0 holddown, 1 hidden)

user@London>
```

  - **The `receive-protocol` option shows routes before policy actions have occurred**

Filtering Routes — Communities

- **Like before, the current *`filter-on-community`* policy is not very scalable. Let’s change that!**

```
[edit]
user@London# show policy-options
policy-statement filter-on-community {
    term remove-all-communities {
        then {
            community delete all-communities;
        }
    }
    term nothing-with-1234 {
        from community AS64321-community;
        then reject;
    }
}
community AS64321-community members 64321:1234;
community all-communities members *:*;
```

Filtering Routes — Communities

- **We don’t have any communities on the routes in the routing table**

```
user@London> show route detail | match comm
user@London>
```

- **But something has changed.**
  - **The 10.244/16 route is now being accepted**

```
user@London> show route receive-protocol bgp 192.168.24.1
inet.0: 24 destinations, 24 routes (24 active, 0 holddown, 0 hidden)
  Prefix                     Nexthop                  MED     Lclpref     AS path
* 10.200.0.0/16              192.168.24.1                                 65001 I
* 10.222.0.0/16              192.168.24.1                                 65001 I
* 10.244.0.0/16              192.168.24.1                                 65001 I
```

Filtering Routes — Communities

- We've created a logic error in our policy
  - There is no `accept` or `reject` action in the first term
  - All routes have their community removed
  - The second term no longer matches the appropriate route

Filtering Routes — Communities

- **Alter the term order with the insert command**

```
[edit policy-options policy-statement filter-on-community]
lab@London# insert term remove-all-communities after term nothing-
with-1234


[edit policy-options policy-statement filter-on-community]
lab@London# show
term nothing-with-1234 {
    from community AS64321-community;
    then reject;
}
term remove-all-communities {
    then {
        community delete all-communities;
    }
}
```

Filtering Routes — Communities

- **Now things look better!**

```
user@London> show route detail | match comm

user@London>




user@London> show route receive-protocol bgp 192.168.24.1
inet.0: 24 destinations, 24 routes (23 active, 0 holddown, 1 hidden)
  Prefix                   Nexthop              MED     Lclpref    AS path
* 10.200.0.0/16            192.168.24.1                            65001 I
* 10.222.0.0/16            192.168.24.1                            65001 I
```

Troubleshooting Commands

- `show configuration`
- `show route advertising-protocol bgp` ***neighbor***
- `show route receive-protocol bgp` ***neighbor***
- `show route hidden`
- `show route community-name` ***name-of-community***
  - **Displays all routes containing the community value defined in *name-of-community***
- `show route` ***detail***
  - **Displays routes and their communities values, if appropriate**
- `show route detail | match comm`
  - **Displays only community values**
  - **Use to view possible communities in the routing table**

Agenda: Troubleshooting BGP

- Originating Routes
- Filtering Routes
  - AS Path
  - Prefixes
  - Communities
- → Reference Slides

The Match Type Option

- Specifies type of match applied to destination prefix

Match Type	Match if…
`exact`	Prefix-length is *equal* to route’s prefix length
`orlonger`	Prefix-length is *equal to or greater than* route’s prefix length
`longer`	Prefix-length is *greater than* route’s prefix length

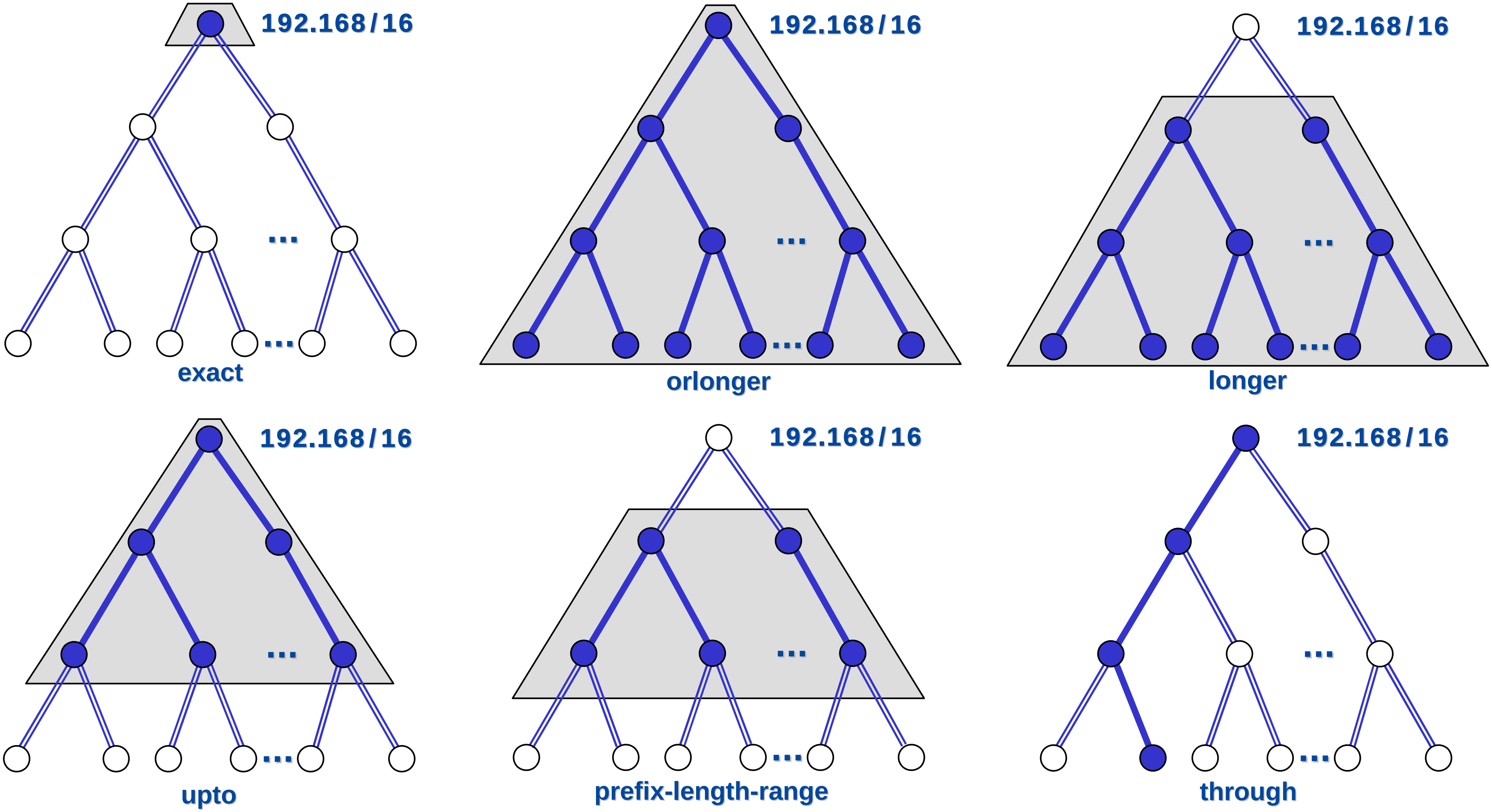
The Match Type Option

- Specifies type of match applied to destination prefix

Match Type	Match if…
`upto`	Route shares most significant bits (as set in prefix-length) and route's prefix length falls between prefix-length and prefix-length2
`prefix-length-range`	Route shares most significant bits and the prefix length is between the two lengths specified
`through`	Route falls exactly between first prefix/prefix-length and second prefix/prefix-length (list of exact matches)

What Matches?

Starting at prefix of 192.168/16, what matches with each option?



Match Type Examples

Prefix	192.168/16 exact	192.168/16 orlonger	192.168/16 longer	192.168/16 upto /24	192.168/16 prefix-length-range /18-/20	192.168/16 through 192.168.16/20
192.0.0.0/8						
192.168.0.0/16	Passes	Passes		Passes		Passes
192.168.0.0/17		Passes	Passes	Passes		Passes
192.168.0.0/18		Passes	Passes	Passes	Passes	Passes
192.168.0.0/19		Passes	Passes	Passes	Passes	Passes
192.168.4.0/24		Passes	Passes	Passes		
192.168.5.4/30		Passes	Passes			
192.168.12.4/30		Passes	Passes			
192.168.12.128/32		Passes	Passes			
192.168.16.0/20		Passes	Passes	Passes	Passes	Passes
192.168.192.0/18		Passes	Passes	Passes	Passes	
192.168.224.0/19		Passes	Passes	Passes	Passes	
192.169.1.0/24						
192.170.0.0/16						

Regular Expression Terms

- Regular expressions take form ***term*** ***<operator>***
- *Terms* are mandatory, and identify the AS number:
  - Can be a single AS number
    - “1024”
  - Can be a complete AS path
    - “1024 2685 3957”
  - Can be a wildcard “.” character which represents a single AS
    - “1024 . 3957”
- Each AS number (not a character) represents a single “entity” to the regular expression parser

Regular Expression Operators

- **Regular expressions take form *term* *<operator>***
- **The operator is an optional pattern matching character that applies to a single term:**
  - **Operators immediately follow the term referenced**
    - **“1024? 2685”**
  - **The pipe (|) operator is used between terms**
    - **“1024 | 2685”**
  - **The dash (-) operator is used between terms**
    - **“1024 – 2685”**
- **One or more term-operator pairs can appear in an AS Path Regular Expression**

AS Path Regex Operators

{m,n}	Match at least *m* and at most *n* repetitions of *term*
{m}	Match exactly *m* repetitions of *term*
{m,}	Match *m* or more repetitions of *term*
*	Match 0 or more repetitions of *term*, same as {0,}
+	Match 1 or more repetitions of *term*, same as {1,}
?	Match 0 or 1 repetitions of *term*, same as {0,1}
|	Match one of the two *terms* on either side of the pipe
-	Used to represent a range
(…),()	Used to group *terms*, or indicate null with no space

Regular Expression Examples

AS Path pattern to match:	Regex:	Example matches:
Exactly one instance of AS 1234	1234	1234
0 or more instances of AS 1234	1234*	1234, 1234 1234, etc., or Null AS Path
0 or 1 instances of AS 1234	1234?	1234 Null AS Path
1 to 3 instances of AS 12 followed by 1 instance of AS 34	“12{1,3} 34”	12 34, 12 12 34, 12 12 12 34
Range of AS numbers to match a single AS	“123 – 125”	123 or 124 or 125

Community Actions: add

Leave existing communities alone and add in the specified value

```
192.168.0.0/24 (2 entries, 1 announced)
    Communities: 64512:567 100:20 50:70 1234:66
```

```
[edit policy-options]
policy-statement community-actions {
    term add-a-community
        then community add test-comm;
    }
}
community test-comm members 65001:1234;
```

```
192.168.0.0/24 (2 entries, 1 announced)
    Communities: 64512:567 100:20 50:70 1234:66 65001:1234
```

Community Actions: `delete`

Remove only the specified values and leave other existing communities alone

```
192.168.0.0/24 (2 entries, 1 announced)
    Communities: 64512:567 100:20 50:70 1234:66
```

```
[edit policy-options]
policy-statement community-actions {
    term add-a-community
        then community delete test-comm;
    }
}
community test-comm members 64512:567;
```

```
192.168.0.0/24 (2 entries, 1 announced)
    Communities: 100:20 50:70 1234:66
```

Community Actions: set

Remove ALL existing communities and add the specified values

```
192.168.0.0/24 (2 entries, 1 announced)
    Communities: 64512:567 100:20 50:70 1234:66
```

```
[edit policy-options]
policy-statement community-actions {
    term add-a-community
        then community set test-comm;
    }
}
community test-comm members 65001:1234;
```

```
192.168.0.0/24 (2 entries, 1 announced)
    Communities: 65001:1234
```

Questions and Comments

- We’ve attempted to show you the tools that allow you to troubleshoot your BGP networks
  - After you master these concepts, you can attack “bigger” problems
- Future topics?
  - Establishing Peers
  - Route Selection
  - Others?
- Feedback on this presentation is highly encouraged
  - jms@juniper.net
- Questions?

Thank you!

http://www.juniper.net

Juniper Networks assumes no responsibility for any inaccuracies in this presentation. Juniper Networks reserves the right to change, modify, transfer, or otherwise revise this information without notice.