Setting up eBGP

BGP for networks who peer: Part 3b

Wolfgang Tremmel
wolfgang.tremmel@de-cix.net
BGP (new) Webinars Overview

- 01 - Prefixes and AS numbers
- 02 - BGP Introduction
- 03a - Setting up iBGP
- 03b - Setting up eBGP
- 04 - Becoming multi-homed
- 05 - BGP Best Path Selection
BGP (new) Webinars Overview

- 01 - Prefixes and AS numbers
- 02 - BGP Introduction
- 03a - Setting up iBGP
- 03b - Setting up eBGP
- 04 - Becoming multi-homed
- 05 - BGP Best Path Selection
BGP - not re-inventing the wheel
BGP - not re-inventing the wheel

→ BGP uses TCP for transport
BGP - not re-inventing the wheel

- BGP uses TCP for transport
- so no need to re-implement features TCP already provides, like
  - reliable transport
  - flow control
  - framing
BGP - not re-inventing the wheel

- BGP uses TCP for transport
- so no need to re-implement features TCP already provides, like
  - reliable transport
  - flow control
  - framing
- as long as the TCP session is up, BGP assumes its neighbors are up
- and have all the information sent to them
BGP - (re)distributing prefixes
BGP - (re)distributing prefixes

- a BGP speaking router
- learns prefixes
BGP - (re)distributing prefixes

- a BGP speaking router
- learns prefixes
- distributes prefixes to its BGP neighbors
BGP - (re)distributing prefixes

- a BGP speaking router
- learns prefixes
- distributes prefixes to its BGP neighbors

- Everything BGP learns from external
  - it distributes internal
  - it distributes external

- Everything BGP learns from internal
  - it distributes external
Everything BGP learns from external it distributes external

Everything BGP learns from internal it distributes external
Everything BGP learns from external
it distributes external
it distributes internal
Everything BGP learns from internal
it distributes external
Everything BGP learns from **external**

- it distributes **external**
- it distributes **internal**

Everything BGP learns from internal
- it distributes **external**
Everything BGP learns from **external**

→ it distributes **external**

→ it distributes **internal**

→ Everything BGP learns from **internal**

→ it distributes **external**
Everything BGP learns from **external**

it distributes **external**
Everything BGP learns from **external**

it distributes **external**

AS64500

AS64496

AS64497

AS64511
Everything BGP learns from external.
It distributes external.
Everything BGP learns from **external**
- it distributes **external**

You need filtering!
You need filtering!
You need filtering!

- You have multiple sources of prefixes
- Upstream provider(s)
You need filtering!

- You have multiple sources of prefixes
- upstream provider(s)
- peering(s)
You need filtering!

- You have multiple sources of prefixes
  - upstream provider(s)
  - peering(s)
  - customer(s)
You need filtering!

- You have multiple sources of prefixes
  - upstream provider(s)
  - peering(s)
  - customer(s)
  - your own prefixes!
You need filtering!

- You have multiple sources of prefixes
  - upstream provider(s)
  - peering(s)
  - customer(s)
  - your own prefixes!
- And destinations to which you announce prefixes
  - upstream providers

AS64496
upstream

your own prefixes

AS64511
customer
You need filtering!

- You have multiple sources of prefixes
  - upstream provider(s)
  - peering(s)
  - customer(s)
  - your own prefixes!
- And destinations to which you announce prefixes
  - upstream providers
  - peerings
You need filtering!

→ You have multiple sources of prefixes
  → upstream provider(s)
  → peering(s)
  → customer(s)
  → your own prefixes!
→ And destinations to which you announce prefixes
  → upstream providers
  → peerings
  → customers
Prefixes
Sources and Destinations

AS64496 upstream

DE CIX

customers

AS64497 upstream

DE CIX

customers

your own prefixes
Prefixes
Sources and Destinations

AS64496 upstream

your own prefixes

AS64497 upstream

DE CIX

customers
Prefixes
Sources and Destinations

AS64496 upstream

DE CIX

customers

your own prefixes

AS64497 upstream

DE CIX

customers
Easy filtering for beginners

- Deny everything outgoing
- Allow everything incoming

- Open filters step by step to allow certain prefixes through
Easy filtering for beginners

→ Deny everything outgoing

→ Allow everything incoming

route-map upstream-out deny 100
!
route-map upstream-in permit 100
!

→ Open filters step by step to allow certain prefixes through
Easy filtering for beginners

➔ Deny everything outgoing
➔ Allow everything incoming

➔ Open filters step by step to allow certain prefixes through

route-map upstream-out deny 100
route-map upstream-in permit 100

ip prefix-list my-networks permit 198.51.100.0/24
route-map upstream-out permit 50
  match ip address prefix-list my-networks
route-map upstream-out deny 100
BGP Session Setup
BGP Session Setup

- BGP uses TCP for transport
- TCP already provides **reliable** transport
- but a bit more is needed
**BGP Session Setup**

- BGP uses **TCP for transport**
- TCP already provides **reliable** transport
- but a bit more is needed
- some information exchange at **setup**
BGP Session Setup

- BGP uses **TCP for transport**
- TCP already provides **reliable** transport
- but a bit more is needed
  - some information exchange at **setup**
  - some mechanism for **keepalive**
BGP Session Setup

- BGP uses TCP for transport
- TCP already provides reliable transport
- but a bit more is needed
  - some information exchange at setup
  - some mechanism for keepalive
- a state model and timers
State model for a BGP session (incomplete)
State model for a BGP session (incomplete)
State model for a BGP session (incomplete)

- Idle
- Connect
- Open Sent
- Open Confirm
- Established

Active
State model for a BGP session (incomplete)

- Idle
- Connect
- Open Sent
- Open Confirm
- Established

Active
State model for a BGP session (incomplete)
State model for a BGP session (incomplete)

- Idle
- Connect
- Open Sent
- Open Confirm
- Established
- Active
State model for a BGP session (incomplete)

- Idle
- Connect
- Open Sent
- Open Confirm
- Established
- Active
State model for a BGP session (incomplete)

- Idle
- Connect
- Open Sent
- Open Confirm
- Established

Active
Experiment: Setup eBGP

experiment 02a + ./2c-02-solution-announce-prefix
Summary

- BGP uses TCP
- eBGP is BGP between Autonomous Systems
- BGP distributes prefixes
  - from external to internal
  - from internal to external
  - from external to external
- Filtering!
Thank you!