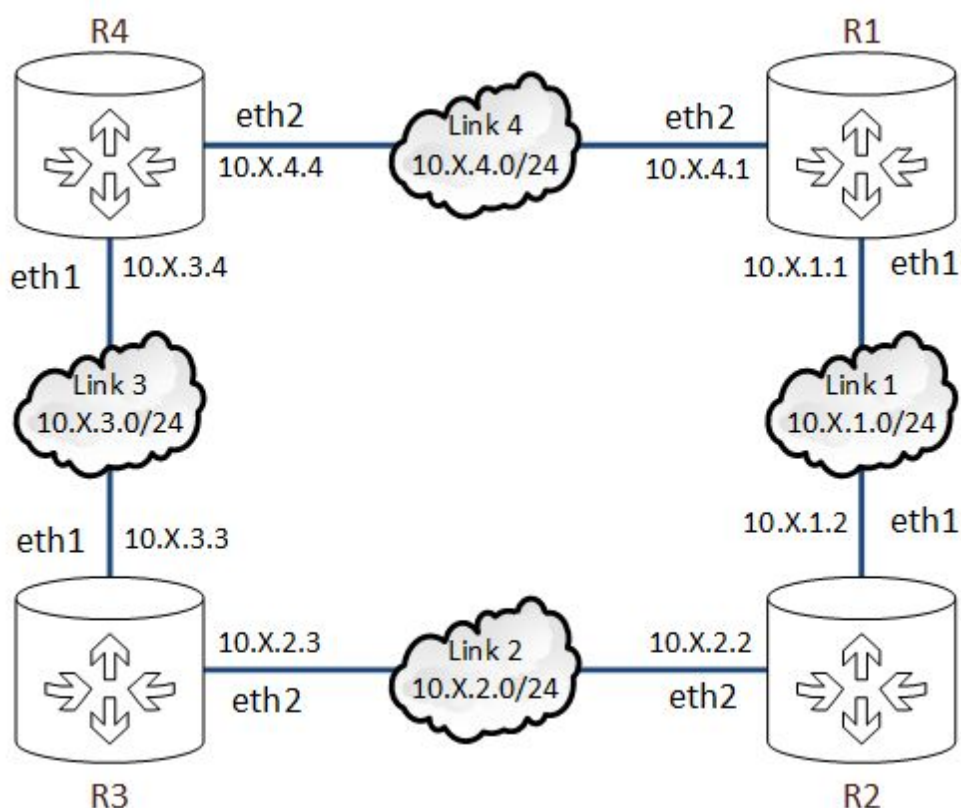


eBGP LAB

Zielsetzung:

Ziel dieses Labs ist es pro Gruppe zwei eBGP Sitzungen mit den beiden Upstream ISPs AS201701 und AS49009 aufzubauen. Über beide Peerings soll ein Announcement eures IP Spaces in einem Aggregat 10.X.0.0/16 erfolgen. Diese Route werden die beiden ISPs an andere Gruppen weiterverteilen.



Schritt 1: Aufbau der eBGP Sitzungen

Die Router R1 und R2 verfügen jeweils über eine bisher unbeachtete Netzwerkschnittstelle: eth3. eth3 ist das Interface zum Upstream ISP. Auf dem Link zum Upstream wird das IP-Netz 100.64.X.0/31 (ISP-A an R1) bzw 100.65.X.0/31(ISP-B an R2) verwendet. Die Adresse .0 ist der Router des ISP. Die .1 ist euer Router.

Eure AS Nummer lautet: 65500 + Gruppennummer

Konfiguriere die eBGP Sitzungen auf R1 und R2!

Schritt 2: Überprüfe den Zustand der BGP Sitzung

```
birdc> show protocols all ISP-A
```

Tipp: Active bedeutet nicht dass eine Sitzung etabliert wurde. Was ihr braucht ist der Established Zustand!

Schritt 3: Existenz der zu announcenden Route sicherstellen

Stelle Sicher, dass die Routen, welche Ihr Announcen wollt, in den Routing Tabellen eurer Edge Router (Die Router welche mit den ISPs verbunden sind) vorhanden ist. Hierzu verwendet man typischerweise eine statische unreachable route.

```
birdc> show route for 10.X.0.0/16
```

Schritt 4: Erstelle einen Filter welcher euer Subnetz an die ISPs exportiert/announced

```
filter isp-out {  
  if net = 10.X.0.0/16 then {  
    accept;  
  }  
  reject;  
}
```

Schritt 5: Erstelle einen Filter der alle Routen akzeptiert

```
filter accept-all {  
  accept;  
}
```

Schritt 6: Configuriere die erstellten Filter als import/export filter für die BGP Sessions

```
protocol bgp ISP-A {  
  neighbor x.x.x.x as 201701;  
  local as 655xx;  
  
  export filter isp-out;  
  import filter accept-all;  
}
```

Schritt 7: Prüfe die empfangenen Routen

```
birdc> show route protocol ISP-A
```

Schritt 8: Prüfe die exportierten Routen:

```
birdc> show route export ISP-A
```

Zeit sich auch die Routing Tabelle mal wieder anzusehen...

```
birdc> show route
```

Schritt 9: Versuche einen der Edge Router einer anderen Gruppe zu pingen

```
ping 10.X.0.1
```

Schritt 10: Konfiguriere eine LOCAL_PREF von 1000 zu ISP-B (sodass 100% des ausgehenden Traffic über ISP-B euer Netz verlassen)

```
function set_top_local_pref() {  
  bgp_local_pref = 1000;  
}
```

```
filter isp-b {  
  if net = 10.X.0.0/16 then {  
    set_top_local_pref();  
    accept;  
  }  
  reject;  
}
```

Anschließend per traceroute / mtr testen!

Schritt 11: Verlängere den AS_PATH zu ISP-B sodass 100% des Traffics über ISP-A zu euch gelangen

```
function prepend_once() {  
  bgp_path.prepend(<ASN>);  
}
```

```
filter isp-b-out {
```

```
if net = 10.X.0.0/16 then {  
    prepend_once();  
    accept;  
}  
reject;  
}
```

Schritt 12: Lasst euch von einer anderen Gruppe pingen/tracerouten und prüft wo der Traffic in euer Netz kommt!

Tipp: `tcpdump -n -i <INTERFACE>`