BGP and the Internet

Transit and Internet Exchange Points

Definitions

- **Transit – carrying traffic across a network, usually for a fee**

  traffic and prefixes originating from one AS are carried across an intermediate AS to reach their destination AS

- **Exchange Points – common interconnect location where several ASes exchange routing information and traffic**

ISP Transit Issues

- Only announce default to your BGP customers unless they need more prefixes
- Only accept the prefixes which your customer is entitled to originate
- If your customer hasn't told you he is providing transit, don't accept anything else

ISP Transit Issues

Many mistakes are made on the Internet today due to incomplete understanding of how to configure BGP for transit

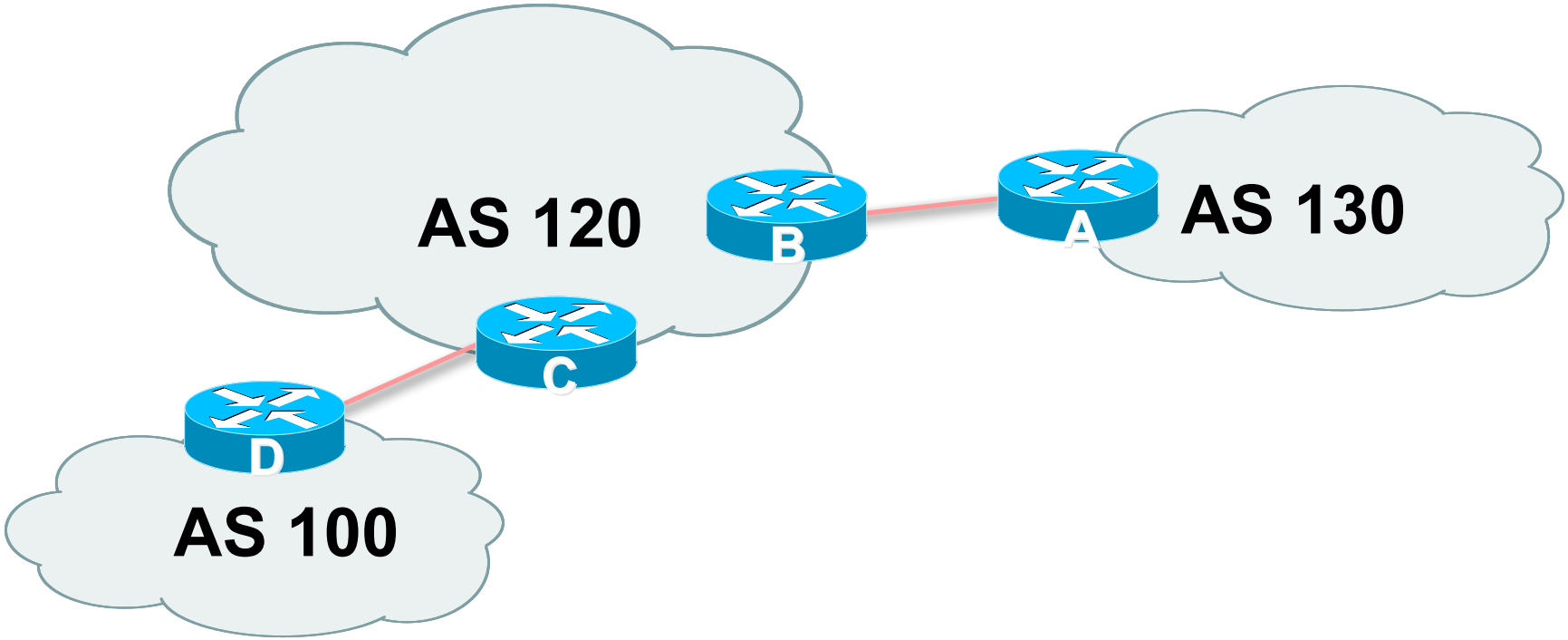
ISP Transit Provider

Simple Example

ISP Transit

- **AS130 and AS100 are stub/customer ASes of AS120**
  - **they may have their own peerings with other ASes**
  - **minimal routing table desired**
  - **minimum complexity required**

ISP Transit



- **AS120 is transit provider between AS130 and AS100**

ISP Transit

- **Router A Configuration**

```
router bgp 130
 network 121.10.0.0 mask 255.255.224.0
 neighbor 122.12.10.2 remote-as 120
 neighbor 122.12.10.2 prefix-list upstream out
 neighbor 122.12.10.2 prefix-list default in
!
ip prefix-list default permit 0.0.0.0/0
ip prefix-list upstream permit 121.10.0.0/19
!
ip route 121.10.0.0 255.255.224.0 null0
```

ISP Transit

- **Router B Configuration**

```
router bgp 120
 neighbor 122.12.10.1 remote-as 130
 neighbor 122.12.10.1 default-originate
 neighbor 122.12.10.1 prefix-list Customer130 in
 neighbor 122.12.10.1 prefix-list default out
!
ip prefix-list Customer130 permit 121.10.0.0/19
ip prefix-list default permit 0.0.0.0/0
```

- **Router B announces default to Router A, only accepts customer /19**

ISP Transit

- **Router C Configuration**

```
router bgp 120
 neighbor 122.12.20.1 remote-as 100
 neighbor 122.12.20.1 default-originate
 neighbor 122.12.20.1 prefix-list Customer100 in
 neighbor 122.12.20.1 prefix-list default out
!
ip prefix-list Customer100 permit 109.0.0.0/19
ip prefix-list default permit 0.0.0.0/0
```

- **Router C announces default to Router D, only accepts customer /19**

ISP Transit

- **Router D Configuration**

```
router bgp 100
 network 109.0.0.0 mask 255.255.224.0
 neighbor 122.12.20.2 remote-as 120
 neighbor 122.12.20.2 prefix-list upstream out
 neighbor 122.12.20.2 prefix-list default in
!
ip prefix-list default permit 0.0.0.0/0
ip prefix-list upstream permit 109.0.0.0/19
!
ip route 109.0.0.0 255.255.224.0 null0
```

ISP Transit

- **This is simple case:**
  - **if AS130 or AS100 get another address block, it requires AS120 and their own filters to be changed**
    - **some ISP transit provider are better skilled at doing this than others!**
  - **May not scale if they are frequently adding new prefixes**

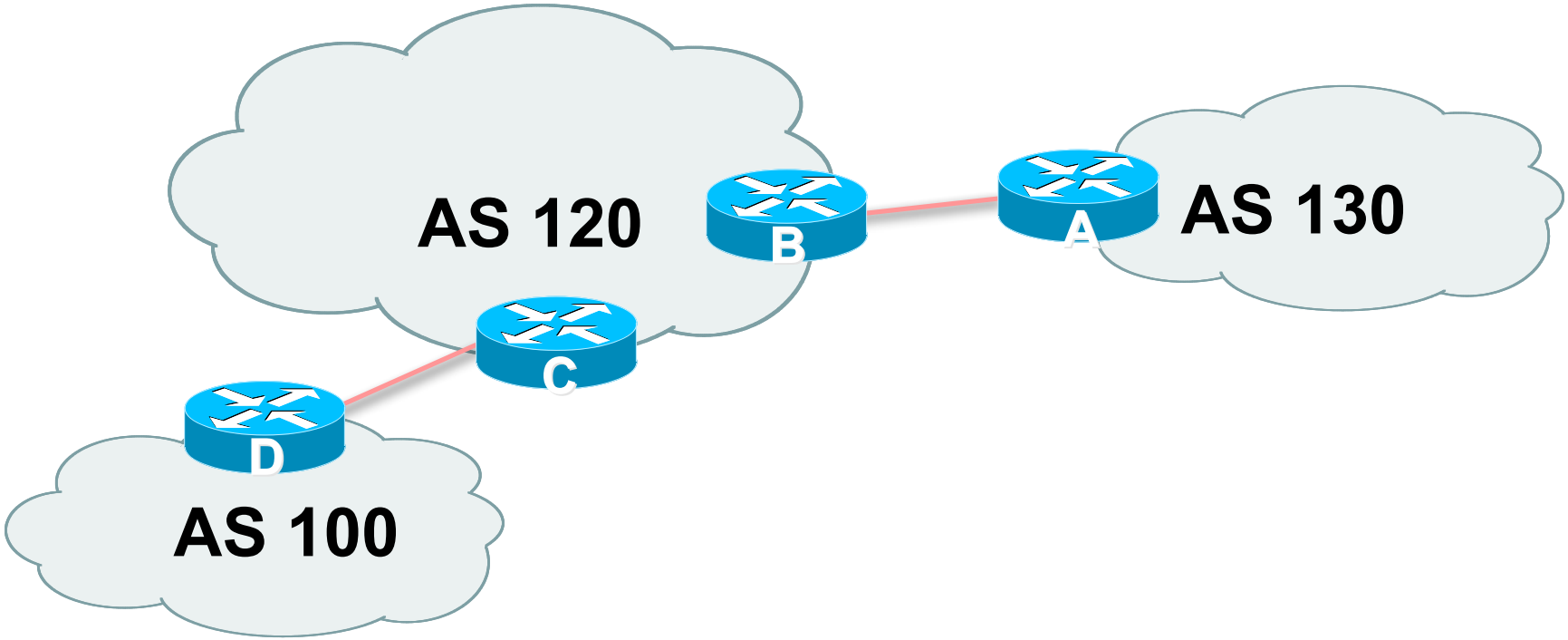
ISP Transit Provider

More complex Example 1

ISP Transit

- **AS130 and AS100 are stub/customer ASes of AS120**
  - **AS120 provides transit between AS130 and AS100 only**
  - **AS120 does not provide Internet connectivity to AS130**

ISP Transit



- **AS120 is transit provider between AS130 and AS100**

ISP Transit

- **Router A Configuration**

```
router bgp 130
 network 121.10.0.0 mask 255.255.224.0
 neighbor 122.12.10.2 remote-as 120
 neighbor 122.12.10.2 prefix-list upstream out
 neighbor 122.12.10.2 prefix-list bogons in
!
ip prefix-list upstream permit 121.10.0.0/19
!
ip route 121.10.0.0 255.255.224.0 null0
```

ISP Transit

- **Router B Configuration**

```
router bgp 120
 neighbor 122.12.10.1 remote-as 130
 neighbor 122.12.10.1 prefix-list Customer130 in
 neighbor 122.12.10.1 prefix-list bogons out
 neighbor 122.12.10.1 filter-list 15 out
!
ip as-path access-list 15 permit ^$
ip as-path access-list 15 permit ^100$
ip prefix-list Customer130 permit 121.10.0.0/19
```

- **Router B announces AS120 and AS100 prefixes to Router A, only accepts customer /19**

ISP Transit

- **Router C Configuration**

```
router bgp 120
 neighbor 122.12.20.1 remote-as 100
 neighbor 122.12.20.1 default-originate
 neighbor 122.12.20.1 prefix-list Customer100 in
 neighbor 122.12.20.1 prefix-list default out
!
ip prefix-list Customer100 permit 109.0.0.0/19
ip prefix-list default permit 0.0.0.0/0
```

- **Router C announces default to Router D, only accepts customer /19**

ISP Transit

- **Router D Configuration**

```
router bgp 100
 network 109.0.0.0 mask 255.255.224.0
 neighbor 122.12.20.2 remote-as 120
 neighbor 122.12.20.2 prefix-list upstream out
 neighbor 122.12.20.2 prefix-list default in
!
ip prefix-list default permit 0.0.0.0/0
ip prefix-list upstream permit 109.0.0.0/19
!
ip route 109.0.0.0 255.255.224.0 null0
```

ISP Transit

- **AS130 only hears AS120 and AS100 prefixes**
  - **inbound AS path filter on Router A is optional, but good practice (never trust a peer)**
  - **inbound Martian prefix-list filters are mandatory on all Internet peerings**

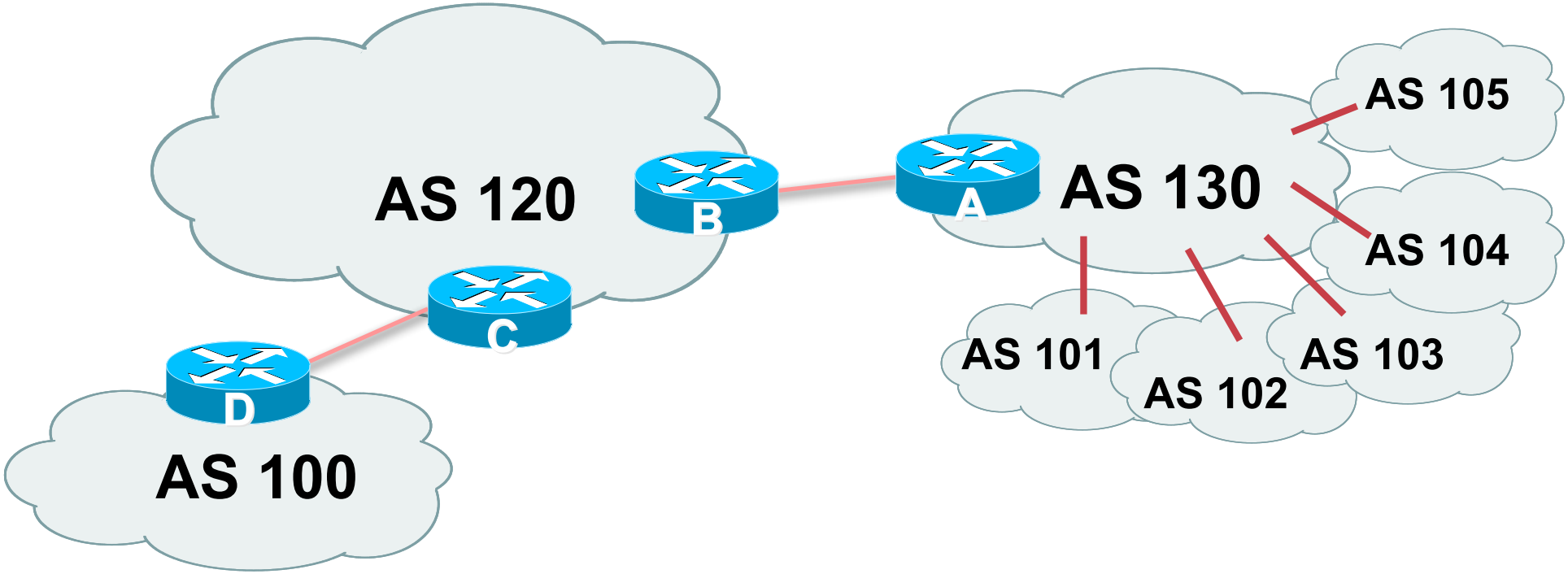
ISP Transit Provider

More complex Example 2

ISP Transit

- **AS130 and AS100 are stub/customer ASes of AS120**
  - **AS130 has many customers with their own ASes**
    - **AS105 doesn't get announced to AS120**
  - **AS120 provides transit between AS130 and AS100**

ISP Transit



- **AS130 has several customer ASes connecting to its backbone**

ISP Transit

- **Router A Configuration**

```
router bgp 130
 network 121.10.0.0 mask 255.255.224.0
 neighbor 122.12.10.2 remote-as 120
 neighbor 122.12.10.2 prefix-list upstream-out out
 neighbor 122.12.10.2 filter-list 5 out
 neighbor 122.12.10.2 prefix-list upstream-in in
!
ip route 121.10.0.0 255.255.224.0 null0 250
!
..next slide
```

ISP Transit

```
!
! As-path filters..
ip as-path access-list 5 permit ^$
ip as-path access-list 5 permit ^(101_)+$
ip as-path access-list 5 permit ^102$
ip as-path access-list 5 permit ^103$
ip as-path access-list 5 permit ^104$
ip as-path access-list 5 deny ^105_
!
..next slide
```

ISP Transit

```
! Outbound Martian prefixes to be blocked to eBGP peers
ip prefix-list upstream-out deny 0.0.0.0/8 le 32
ip prefix-list upstream-out deny 10.0.0.0/8 le 32
ip prefix-list upstream-out deny 127.0.0.0/8 le 32
ip prefix-list upstream-out deny 169.254.0.0/16 le 32
ip prefix-list upstream-out deny 172.16.0.0/12 le 32
ip prefix-list upstream-out deny 192.0.2.0/24 le 32
ip prefix-list upstream-out deny 192.168.0.0/16 le 32
ip prefix-list upstream-out deny 224.0.0.0/3 le 32
ip prefix-list upstream-out deny 0.0.0.0/0 ge 25
! Extra prefixes
ip prefix-list upstream-out deny 121.10.0.0/19 ge 20
ip prefix-list upstream-out permit 0.0.0.0/0 le 32
..next slide
```

ISP Transit

```
! Inbound Martian prefixes to be blocked from eBGP peers
ip prefix-list upstream-in deny 0.0.0.0/8 le 32
ip prefix-list upstream-in deny 10.0.0.0/8 le 32
ip prefix-list upstream-in deny 127.0.0.0/8 le 32
ip prefix-list upstream-in deny 169.254.0.0/16 le 32
ip prefix-list upstream-in deny 172.16.0.0/12 le 32
ip prefix-list upstream-in deny 192.0.2.0/24 le 32
ip prefix-list upstream-in deny 192.168.0.0/16 le 32
ip prefix-list upstream-in deny 224.0.0.0/3 le 32
ip prefix-list upstream-in deny 0.0.0.0/0 ge 25
! Extra prefixes
ip prefix-list upstream-in deny 121.10.0.0/19 le 32
ip prefix-list upstream-in permit 0.0.0.0/0 le 32
!
```

ISP Transit

- **Router B Configuration**

```
router bgp 120
 neighbor 122.12.10.1 remote-as 130
 neighbor 122.12.10.1 prefix-list bogons in
 neighbor 122.12.10.1 prefix-list bogons out
 neighbor 122.12.10.1 filter-list 10 in
 neighbor 122.12.10.1 filter-list 15 out
!
ip as-path access-list 15 permit ^$
ip as-path access-list 15 permit ^100$
```

Router B announces AS120 and AS100 prefixes to Router A, and accepts all AS130 customer ASes

ISP Transit

- **Router C Configuration**

```
router bgp 120
 neighbor 122.12.20.1 remote-as 100
 neighbor 122.12.20.1 default-originate
 neighbor 122.12.20.1 prefix-list Customer100 in
 neighbor 122.12.20.1 prefix-list default out
!
ip prefix-list Customer100 permit 109.0.0.0/19
ip prefix-list default permit 0.0.0.0/0
```

- **Router C announces default to Router D, only accepts customer /19**

ISP Transit

- **Router D Configuration**

```
router bgp 100
 network 109.0.0.0 mask 255.255.224.0
 neighbor 122.12.20.2 remote-as 120
 neighbor 122.12.20.2 prefix-list upstream out
 neighbor 122.12.20.2 prefix-list default in
!
ip prefix-list default permit 0.0.0.0/0
ip prefix-list upstream permit 109.0.0.0/19
!
ip route 109.0.0.0 255.255.224.0 null0
```

ISP Transit

- **AS130 only hears AS120 and AS100 prefixes**
  - **inbound AS path filter on Router A is optional, but good practice (never trust a peer)**
  - **Special Use Address prefix-list filters are required on all Internet peerings**

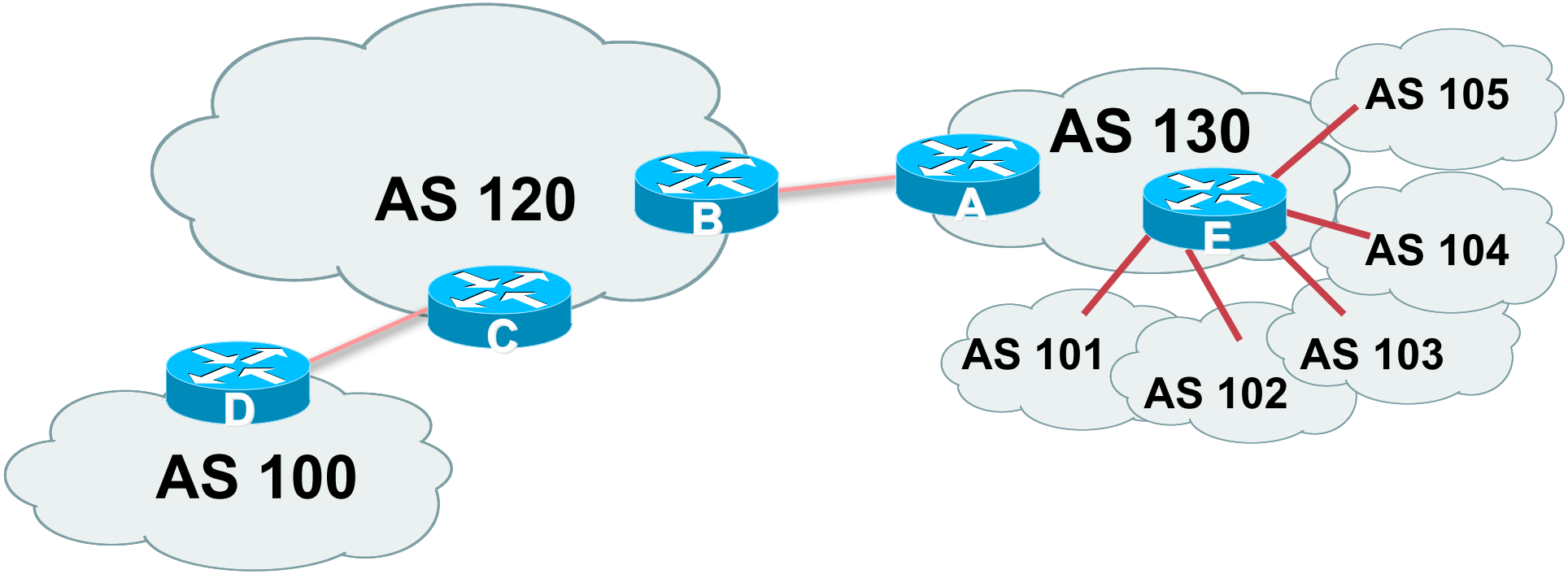
ISP Transit Provider

More complex Example 3

ISP Transit

- AS130 and AS100 are stub/customer ASes of AS120
  - AS130 has many customers with their own ASes
    - AS105 doesn't get announced to AS120
  - AS120 provides transit between AS130 and AS100
- Same example as previously but using communities

ISP Transit



- AS130 has several customer ASes connecting to its backbone

ISP Transit

- **Router A configuration is greatly simplified**
  - **all prefixes to be announced to upstream are marked with community 130:5100**
  - **route-map on outbound peering implements community policy**
  - **Martian prefix-lists still required**

ISP Transit

- **Router A Configuration**

```
router bgp 130
 network 121.10.0.0 mask 255.255.224.0 route-map setcomm
 neighbor 122.12.10.2 remote-as 120
 neighbor 122.12.10.2 prefix-list upstream-out out
 neighbor 122.12.10.2 route-map to-AS120 out
 neighbor 122.12.10.2 prefix-list upstream-in in
!
ip route 121.10.0.0 255.255.224.0 null0 250
!
..next slide
```

ISP Transit

```
!
ip community-list 5 permit 130:5100
!
! Set community on local prefixes
route-map setcomm permit 10
 set community 130:5100
!
route-map to-AS120 permit 10
 match community 5
!
```

- upstream-in and upstream-out prefix-lists are the same as in the previous example

ISP Transit

- **Router E Configuration**

```
router bgp 130
 neighbor x.x.x.x remote-as 101
 neighbor x.x.x.x default-originate
 neighbor x.x.x.x prefix-list customer101 in
 neighbor x.x.x.x route-map bgp-cust-in in
 neighbor x.x.x.x prefix-list default out
 neighbor x.x.x.x remote-as 102
 neighbor x.x.x.x default-originate
 neighbor x.x.x.x prefix-list customer102 in
 neighbor x.x.x.x route-map bgp-cust-in in
 neighbor x.x.x.x prefix-list default out
..next slide
```

ISP Transit

```
 neighbor s.s.s.s remote-as 105
 neighbor s.s.s.s default-originate
 neighbor s.s.s.s prefix-list customer105 in
 neighbor s.s.s.s route-map no-transit in
 neighbor s.s.s.s prefix-list default out
!
! Set community on eBGP customers announced to AS120
route-map bgp-cust-in permit 10
 set community 130:5100
route-map no-transit permit 10
 set community 130:5199
```

Notice that AS105 peering has no route-map to set the community policy

ISP Transit

- AS130 only announces the community 130:5100 to AS120
- Notice how Router E tags the prefixes to be announced to AS120 with community 130:5100
- More efficient to manage than using filter lists

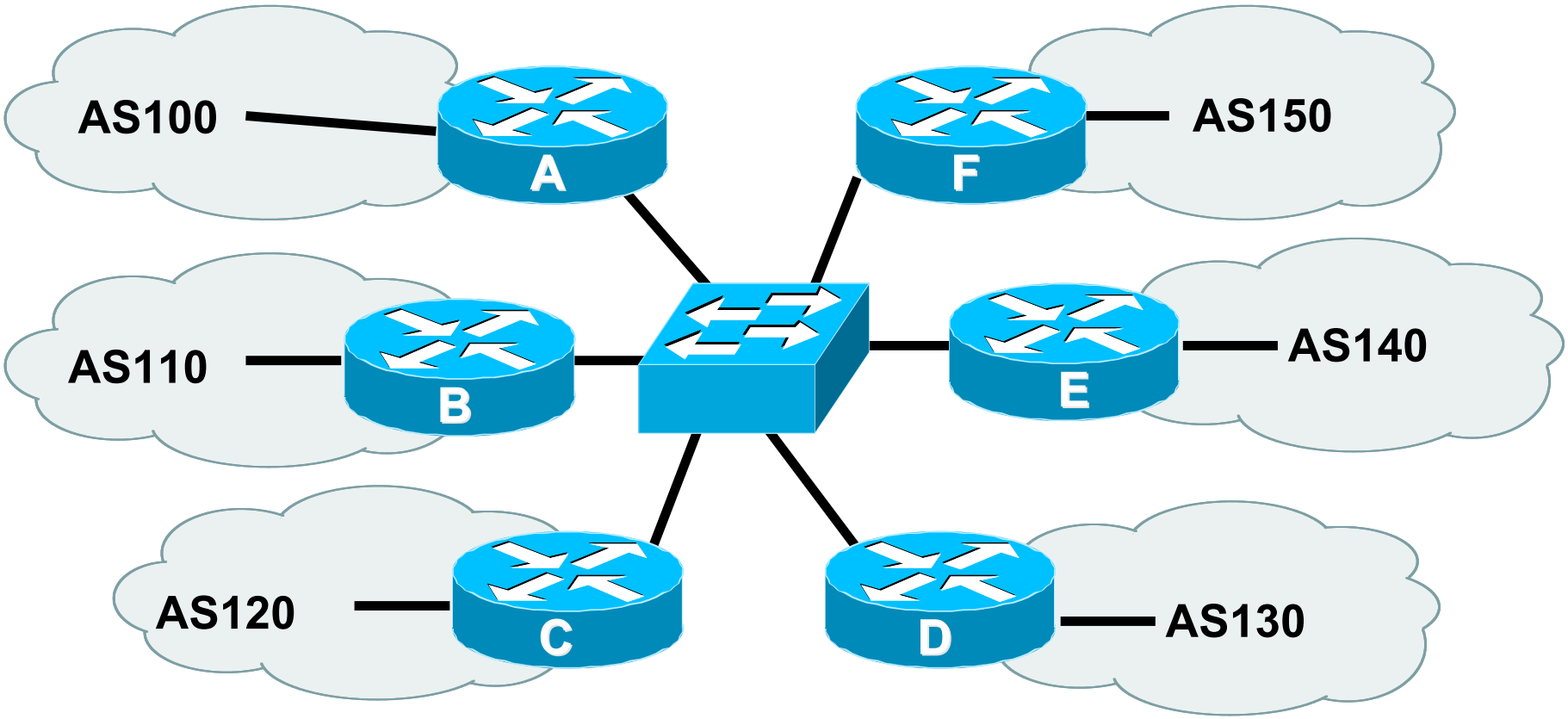
Exchange Points

Simple Example

Exchange Point Example

- **Exchange point with 6 ASes present**
  - **Layer 2 – ethernet switch**
- **Each ISP peers with the other**
  - **NO transit across the IXP allowed**

Exchange Point



each of these represents a border router in a different autonomous system

Exchange Point
Router A configuration

```
interface fastethernet 0/0
 description Exchange Point LAN
 ip address 120.5.10.2 mask 255.255.255.224
 ip verify unicast reverse-path
 no ip directed-broadcast
 no ip proxy-arp
 no ip redirects
!
router bgp 100
 neighbor ixp-peers peer-group
 neighbor ixp-peers send-community
 neighbor ixp-peers prefix-list myprefixes out
 neighbor ixp-peers route-map set-local-pref in
```

..next slide

Exchange Point

```
neighbor 120.5.10.2 remote-as 110
neighbor 120.5.10.2 peer-group ixp-peers
neighbor 120.5.10.2 prefix-list peer110 in
neighbor 120.5.10.3 remote-as 120
neighbor 120.5.10.3 peer-group ixp-peers
neighbor 120.5.10.3 prefix-list peer120 in
neighbor 120.5.10.4 remote-as 130
neighbor 120.5.10.4 peer-group ixp-peers
neighbor 120.5.10.4 prefix-list peer130 in
neighbor 120.5.10.5 remote-as 140
neighbor 120.5.10.5 peer-group ixp-peers
neighbor 120.5.10.5 prefix-list peer140 in
neighbor 120.5.10.6 remote-as 150
neighbor 120.5.10.6 peer-group ixp-peers
neighbor 120.5.10.6 prefix-list peer150 in
```

Exchange Point

```
ip route 121.10.0.0 255.255.224.0 null0
!
ip prefix-list myprefixes permit 121.10.0.0/19
ip prefix-list peer110 permit 122.0.0.0/19
ip prefix-list peer120 permit 122.30.0.0/19
ip prefix-list peer130 permit 122.12.0.0/19
ip prefix-list peer140 permit 122.18.128.0/19
ip prefix-list peer150 permit 122.1.32.0/19
!
route-map set-local-pref permit 10
 set local-preference 150
!
```

Exchange Point

- Configuration of the other routers in the AS is similar in concept
- Notice inbound and outbound prefix filters
  - outbound announces myprefixes only
  - inbound accepts peer prefixes only
- Notice inbound route-map
  - Set local preference higher than default ensures that local traffic crosses the exchange

Exchange Point

- **Ethernet port configuration**

  use *ip verify unicast reverse-path*

  helps prevent “stealing of bandwidth”

- **IXP border router must NOT carry prefixes with origin outside local AS and IXP participant ASes**

  helps prevent “stealing of bandwidth”

Exchange Point

- **Issues:**
  - **AS100 needs to know all the prefixes its peers are announcing**
  - **New prefixes requires the prefix-lists to be updated**
- **Alternative solutions**
  - **Use the Internet Routing Registry to build prefix list**
  - **Use AS Path filters (could be risky)**

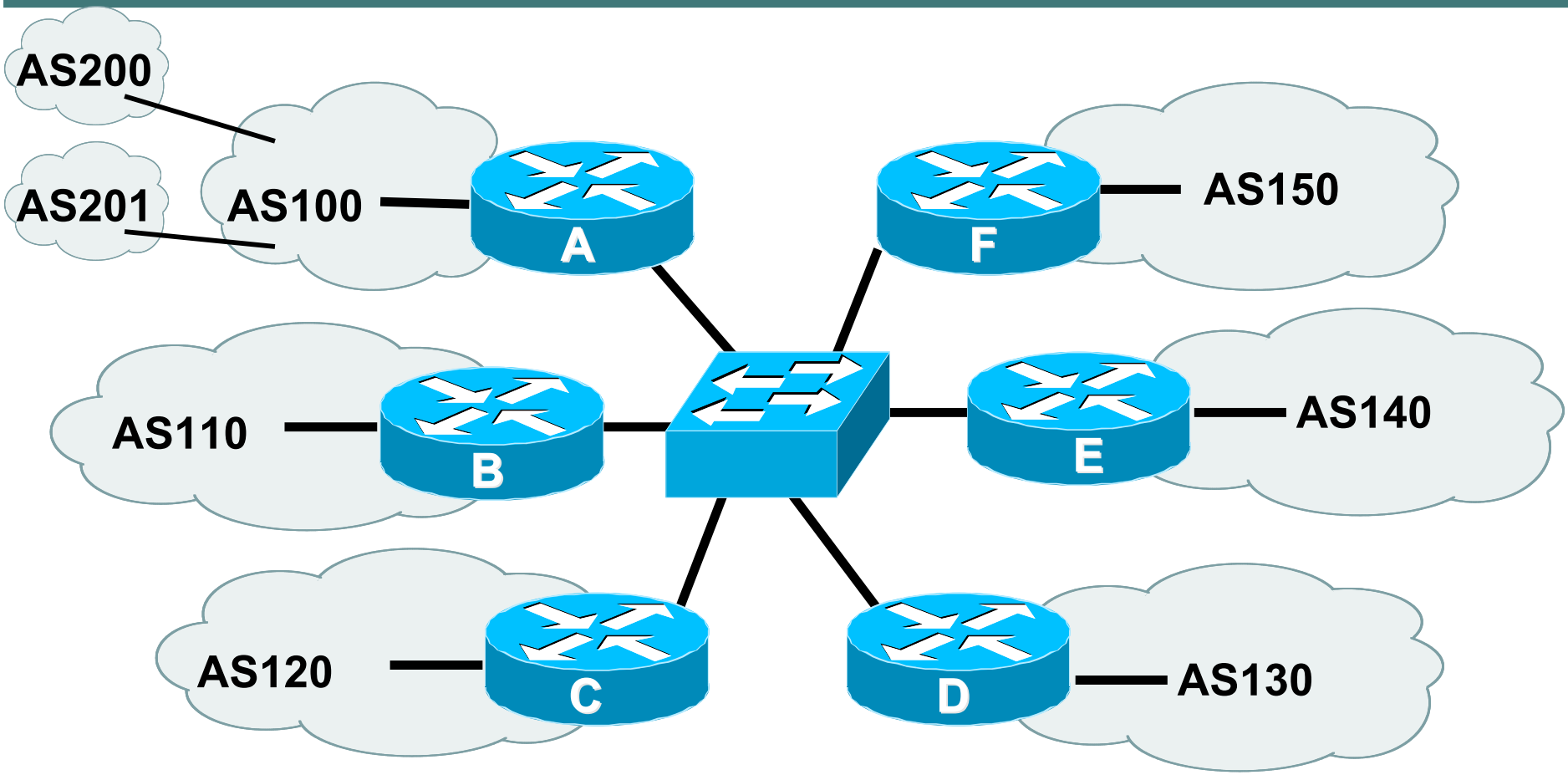
Exchange Points

More Complex Example

Exchange Point Example

- **Exchange point with 6 ASes present**
  - **Layer 2 – ethernet switch**
- **Each ISP peers with the other**
  - **NO transit across the IXP allowed**
  - **ISPs at exchange points provide transit to their customers**

Exchange Point



each of these represents a border router in a different autonomous system

Exchange Point
Router A configuration

```
interface fastethernet 0/0
 description Exchange Point LAN
 ip address 120.5.10.2 mask 255.255.255.224
 ip verify unicast reverse-path
 no ip directed-broadcast
 no ip proxy-arp
 no ip redirects
!
router bgp 100
 neighbor ixp-peers peer-group
 neighbor ixp-peers send-community
 neighbor ixp-peers prefix-list bogons out
 neighbor ixp-peers filter-list 10 out
 neighbor ixp-peers route-map set-local-pref in
```

..next slide

Exchange Point

```
neighbor 120.5.10.2 remote-as 110
neighbor 120.5.10.2 peer-group ixp-peers
neighbor 120.5.10.2 prefix-list peer110 in
neighbor 120.5.10.3 remote-as 120
neighbor 120.5.10.3 peer-group ixp-peers
neighbor 120.5.10.3 prefix-list peer120 in
neighbor 120.5.10.4 remote-as 130
neighbor 120.5.10.4 peer-group ixp-peers
neighbor 120.5.10.4 prefix-list peer130 in
neighbor 120.5.10.5 remote-as 140
neighbor 120.5.10.5 peer-group ixp-peers
neighbor 120.5.10.5 prefix-list peer140 in
neighbor 120.5.10.6 remote-as 150
neighbor 120.5.10.6 peer-group ixp-peers
neighbor 120.5.10.6 prefix-list peer150 in
```

Exchange Point

```
ip route 121.10.0.0 255.255.224.0 null0
!
ip as-path access-list 10 permit ^$
ip as-path access-list 10 permit ^200$
ip as-path access-list 10 permit ^201$
!
ip prefix-list myprefixes permit 121.10.0.0/19
ip prefix-list peer110 permit 122.0.0.0/19
ip prefix-list peer120 permit 122.30.0.0/19
ip prefix-list peer130 permit 122.12.0.0/19
ip prefix-list peer140 permit 122.18.128.0/19
ip prefix-list peer150 permit 122.1.32.0/19
!
route-map set-local-pref permit 10
 set local-preference 150
```

Exchange Point

- Notice the change in router A's configuration

  filter-list instead of prefix-list permits local and customer ASes out to exchange

  prefix-list blocks Special Use Address prefixes – rest get out, could be risky

- Other issues as previously

BGP and the Internet

Transit and Internet Exchange Points