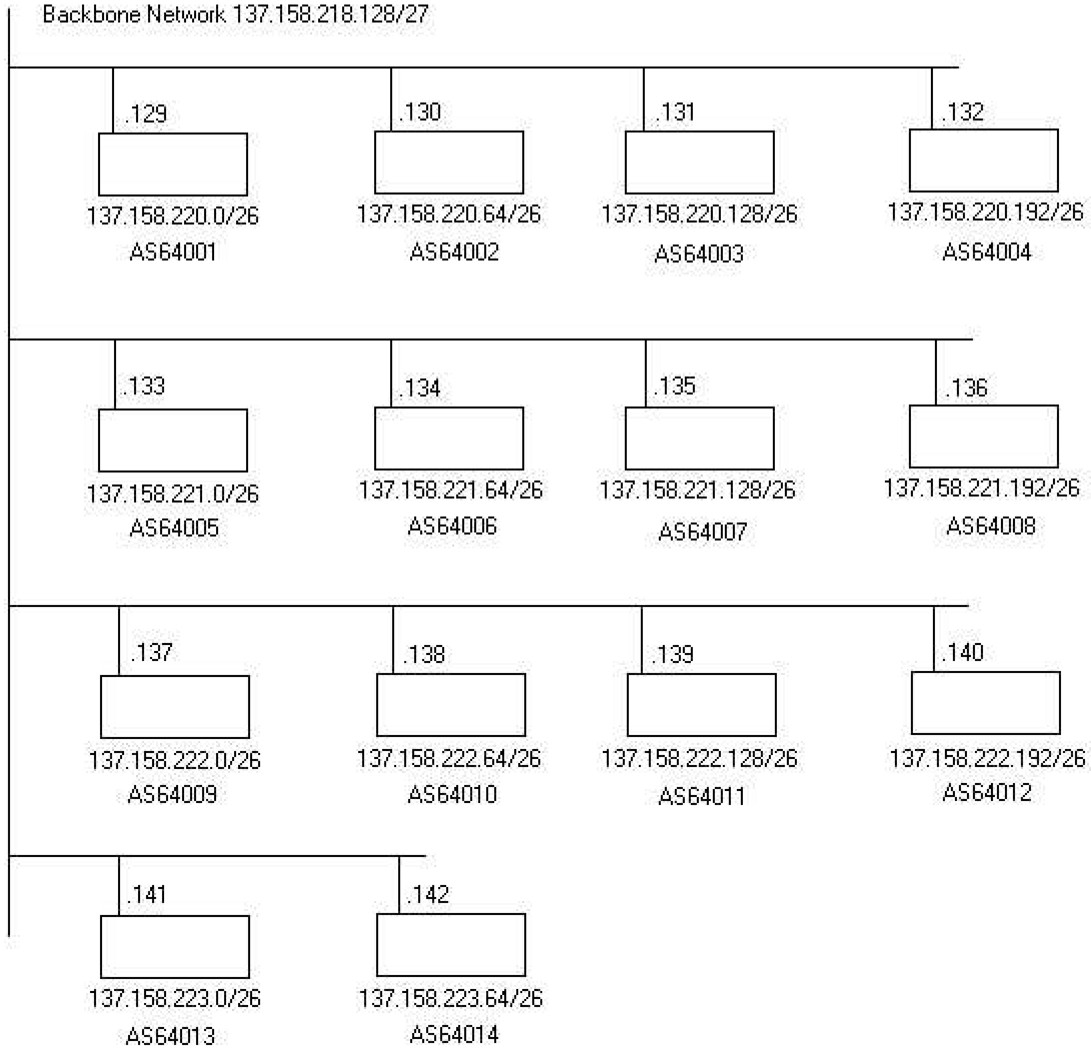
BGP Tutorial AFNOG2000

Class IP Assignments



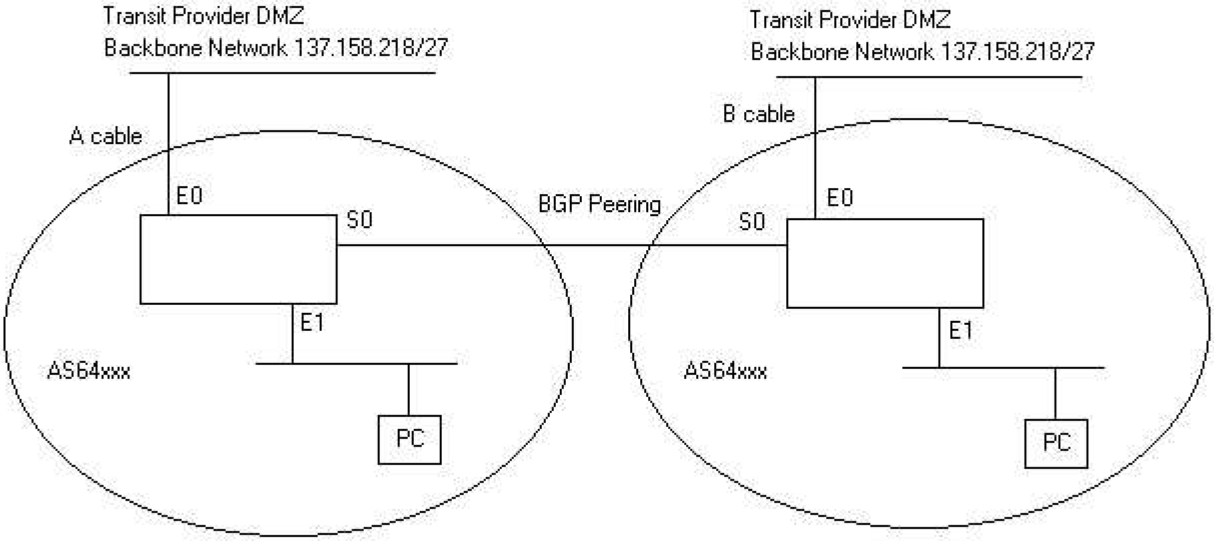
Note: Refer to this sheet for your IP assignments throughout the BGP lab exercises.

Anatomy of a BGP Configuration

```
ip bgp-community new-format
!
router bgp 3582                                          - Initial BGP config statement with ASN
 no synchronization                                      - Do not synchronize with the IGP
 no auto-summary                                         - no classfull boundaries
 bgp dampening                                           - dampen route flaps
 network 128.223.0.0 nlri unicast multicast              - distribute into URIB and MRIB
 neighbor ebgp peer-group nlri unicast multicast         - Initial peer-group configuration
 neighbor ebgp send-community                            - Send communities to peers
 neighbor ebgp version 4                                 - BGP version 4 negotiation
 neighbor ebgp soft-reconfiguration inbound              - Store updates & allow reads. Without reset
 neighbor ebgp prefix-list sanity-filter in              - filter all incoming prefixes against this prefix-
list
 neighbor ebgp prefix-list my-routes out                 - filter all outgoing prefixes against this prefix-
list
 neighbor 207.98.66.11 remote-as 3701                    - neighbor peering session
 neighbor 207.98.66.11 description NERO Transit (noc@nero.net)      - neighbor description
 neighbor 207.98.66.11 update-source POS 6/0/0           - peering session connection source
 neighbor 207.98.66.11 peer-group ebgp                   - inherit configuration from peer-group
 neighbor 198.32.162.9 remote-as 6108                    - neighbor peering session
 neighbor 198.32.162.9 description The Pond (jhill@pond.net)        - neighbor description
 neighbor 198.32.162.9 update-source FastEthernet 5/0/0      - peering session connection source
 neighbor 198.32.162.9 peer-group ebgp                       - inherit configuration from peer-group
 neighbor 198.32.162.9 route-map THE-POND-IN in              - process all incoming advertisements
!                                                                  through route-map.
!
ip prefix-list my-routes seq 5 permit 128.223.0.0/16        - prefix-list filtering all outgoing
advertised
ip prefix-list my-routes seq 50 deny 0.0.0.0/0 le 32               prefixes.
!
ip prefix-list pond seq 5 permit 205.240.25.0/24            - prefix-list filtering all incoming
advertised
ip prefix-list pond seq 10 permit 199.217.92.0/22                  prefixes from peer.
ip prefix-list pond seq 15 permit 206.40.128.0/22
ip prefix-list pond seq 50 deny 0.0.0.0/0 le 32
!
ip prefix-list sanity-filter description "Deny default, RFC1918, net 10"      - no bogons
ip prefix-list sanity-filter seq 10 deny 127.0.0.0/8 le 32
ip prefix-list sanity-filter seq 15 deny 10.0.0.0/8 le 32
ip prefix-list sanity-filter seq 20 deny 172.16.0.0/12 le 32
ip prefix-list sanity-filter seq 25 deny 192.168.0.0/16 le 32
ip prefix-list sanity-filter seq 30 deny 192.0.2.0/24 le 32
ip prefix-list sanity-filter seq 50 permit 0.0.0.0/0 le 32
!
route-map THE-POND-IN permit 5                              - simple route-map example
 match ip address prefix-list pond
```

BGP Lab Exercise 1
Static-default transit / Local BGP peering

Topology for router pairs at each table:



1. Set upstream transit routing

    a. Assign IP address to E0
    b. Set static route to your provider's next-hop
        A cable - 137.158.218.158
        B cable - 137.158.218.157
    c. Ping upstream devices to ensure connectivity

2. Assign a portion of your IP space to the subnet on E1

    a. Your choice – NOTE: You will need to reserve space for your peering link
    b. Configure PC with an address on the E1 subnet

3. Arrange for local peering

    a. Connect serial cable between your S0 and the adjacent S0 on the same table.
    b. Coordinate /30 assignment for the peering link with your table peer.
    c. Assign an IP address on S0 from your agreed/arranged peer.

4. Configure BGP peering with table peer.

```
Router#config t
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#router bgp 64109
Router(config-router)#no synchronization
Router(config-router)#no auto-summary
Router(config-router)#bgp dampening
Router(config-router)#network x.x.x.x mask x.x.x.x
Router(config-router)#neighbor x.x.x.x remote-as 64110
Router(config-router)#neighbor x.x.x.x description My Right Peer
Router(config-router)#neighbor x.x.x.x update-source s0
Router(config-router)#neighbor x.x.x.x prefix-list my-routes out
Router(config-router)#neighbor x.x.x.x prefix-list my-peer in
Router(config-router)#exit
```

Now the two above mentioned prefix-lists must exist

```
Router(config)#ip prefix-list my-routes seq 5 permit x.x.x.x/x
Router(config)#ip prefix-list my-routes seq 50 deny 0.0.0.0/0 le 32

Router(config)#ip prefix-list my-peer seq 5 permit x.x.x.x/x
Router(config)#ip prefix-list my-peer seq 50 deny 0.0.0.0/0 le 32
```

Exit config, and write.
Let's see if the peering is up:

```
Router#sh ip bgp summary
Neighbor        V    AS MsgRcvd MsgSent   TblVer  InQ OutQ Up/Down
State/PfxRcd
123.123.123.2   4  64110   4417    2185   304297    0    0 1m        1
```

Other commands to monitor BGP:

```
Router#sh ip route
Router#sh ip bgp
Router#sh ip bgp neighbor
```

5. Checking connectivity from your router

   a. Traceroute to your peers host. What do you notice?

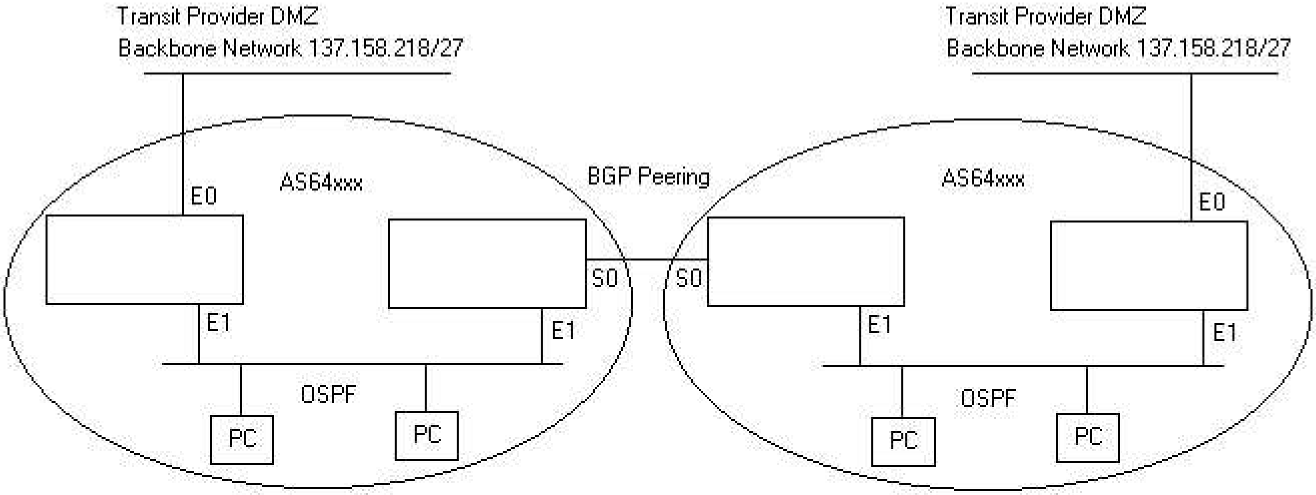
Lab Exercise 1b:

Telnet to: 198.32.162.100

```
route-views.oregon-ix.net>sh ip bgp x.x.x.x
```

BGP Lab Exercise 2
Static-default transit / OSPF IGP / Local BGP Peering

Topology for table pairs:



1. Set upstream transit routing

   a. Assign IP address to E0
   (if you are the peering router in the above config, please remove any previous configuration for E0 and any static default routes)
   b. Set static default route to provider's next hop
   A cable – 137.158.218.158
   B cable – 137.158.218.157
   c. Ping upstream device to ensure connectivity.
   d. Remove all BGP configuration on your transit router.

2. Assign a portion of your IP space to the E1 subnet

   a. Note: your router, and your table neighbors' router are now ONE ISP.
   b. Combine your IP space, and assign a portion of it to the subnet on E1

3. Configure OSPF on E1

   a. Use instructions from Monday's handout to configure OSPF
   b. Be sure all external interfaces of your ISP are passive OSPF
   c. Note: both routers are in the same OSPF area

4. Arrange for local peering

   a. Connect serial cable between your S0 and the adjacent S0 on the same table.
   b. Coordinate /30 assignment for the peering link with your table peer.
   c. Assign an IP address on S0 from your agreed/arranged peer.

5. Modify your BGP configuration on your peering router to reflect your new IP space

   a. Change your route statement in BGP to populate RIB with new IP space.

b. Change your prefix-list statement to reflect your new IP space.
c. Change prefix-list statement to reflect your new peer's IP space.
(Follow exercise 1 for commands)

6. Check connectivity from your two routers

   a. Traceroute to one of your peers hosts from each of your routers. What's different?
   b. Issue show route command on each router. What's different?

***CAUTION* This should NEVER been done ordinarily – ESPECIALLY with BGP full routes**

7. Make your BGP acquired peer routes available throughout your network

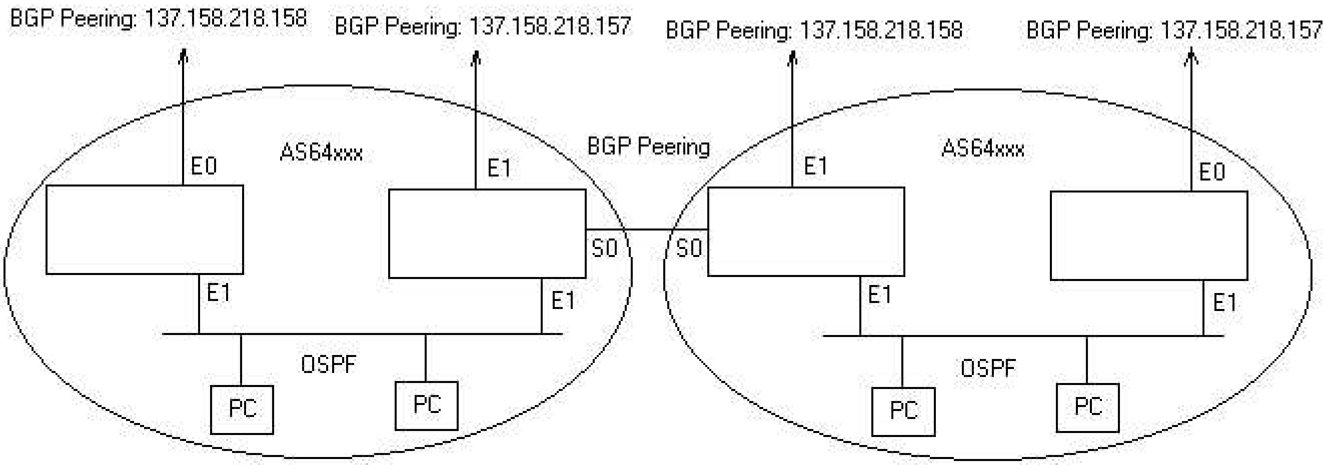
```
Router(config)#router ospf xxx
Router(config-router)#redistribute bgp (your ASN)
```

8. Check connectivity again from your routers

   a. Same steps as in 5.
      What's different? What's the same?

BGP Lab Exercise 3
Dual-homed / OSPF IGP / IBGP / Local peering

Topology for table pairs / tables:


BGP Peering: 137.158.218.158
BGP Peering: 137.158.218.157
BGP Peering: 137.158.218.158
BGP Peering: 137.158.218.157
AS64xxx
E0
E1
BGP Peering
S0
S0
E1
AS64xxx
E0
E1
E1
E1
E1
OSPF
PC
PC
OSPF
PC
PC

1. Remove static default routes to transit provider

```
Router(config)#no ip route x.x.x.x
```

2. Disable BGP peering with neighbor table

```
Router(config)#router bgp (your ASN)
Router(config-router)#no neighbor x.x.x.x
```

3. Add prefix-list for sanity filter

   - refer to Anatomy of a BGP Configuration

4. Arrange dual-homed peering with your transit provider

   a. A Cable connected router peers with AS64513 at 137.158.218.158
   b. B Cable connected router peer with AS64513 at 137.158.218.157
   c. Filter incoming routes from transit provider through the sanity-filter prefix-list

```
Router(config)#router bgp (your ASN)
Router(config-router)#neighbor 137.158.218.158 remote-as 64513
Router(config-router)#neighbor 137.158.218.158 soft-reconfiguration inbound
Router(config-router)#neighbor 137.158.218.158 prefix-list sanity-filter in
Router(config-router)#neighbor 137.158.218.158 prefix-list my-routes out
```

NOTE: Transit provider will not be able to offer you full routes, as you don't have enough ram.

   d. You will need to coordinate with your transit provider Engineer for peering. They will need to know your AS number, your peering IP address(s), and the prefixes you will be advertising.
   e. Once both peering is up, look at your bgp route table. What do you notice?
   f. Traceroute around the conference net. What do you notice?

5. Configure an IBGP session between your two routers

On each router:

```
Router(config)#router bgp (your ASN)
Router(config-router)#neighbor (E1 IP address of OTHER router) remote-as (YOUR
ASN)
Router(config-router)#neighbor x.x.x.x soft-reconfiguration inbound
Router(config-router)#neighbor x.x.x.x send community
Router(config-router)#neighbor x.x.x.x version 4
```

6. Look at your bgp table. What do you notice? Traceroute. What do you notice?

7. Arrange for you transit provider to provide MEDs (Multi-Exit Descriminator)

8. Traceroute around again. What do you notice?

9. Bring peering back up with your row neighbor. (refer to Exercise 1 & 2)

10. What do you notice?

11. Okay, so now you're a pro! You realize your transit connection is on a layer 2 exchange where all of your local ISPs (tables) also connect. So arrange peering with as many of the other tables as you can stand. ;-)