OSPF Section
Last Updated: 4/26/15

Netref Entries:
Linux Configuration: https://netref.soe.ucsc.edu/osnl/39
IOS Configurations: https://netref.soe.ucsc.edu/osnl/37

Concepts you should already know:
  - What are link-state algorithms?
  - Class Lecture: https://classes.soe.ucsc.edu/cmpe151/Spring15/content/lectures/Lecture_3.pdf

Reading OSPF database tables:

**WHY USE SERIAL INTERFACE?** The serial links are used because based on previous experience, the routers were not able to detect link going down if a different connection was used. (HOLD SHIFT when click the add link icon to select the serial link)

Part 1 - Single Area OSPF
R1

```
conf t
int s0/0
ip add 10.0.1.1 255.255.255.0
no shut
end
```

R2

```
conf t
int s0/0
ip add 10.0.1.2 255.255.255.0
no shut
int fa1/0
ip add 10.0.0.2 255.255.255.0
no shut
end
```

R3

```
conf t
int fa0/0
ip add 10.0.0.3 255.255.255.0
no shut
int fa1/0
ip add 10.0.2.3 255.255.255.0
no shut
end
```

R4

```
conf t
int fa1/0
ip add 10.0.2.4 255.255.255.0
no shut
end
```

PC1:

```
ifconfig eth0 10.0.0.1/24
```

**Test connectivity:**

<table>
<thead>
<tr>
<th>R1</th>
<th>R2</th>
<th>R3</th>
<th>PC1</th>
</tr>
</thead>
<tbody>
<tr>
<td>ping 10.0.1.2</td>
<td>ping 10.0.0.3</td>
<td>ping 10.0.2.4</td>
<td>ping -c2 10.0.0.2</td>
</tr>
</tbody>
</table>
Setting up OSPF as a single area

<table>
<thead>
<tr>
<th>R1</th>
<th>R2</th>
<th>R3</th>
<th>R4</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>conf t</code></td>
<td><code>conf t</code></td>
<td><code>conf t</code></td>
<td><code>conf t</code></td>
</tr>
<tr>
<td><code>router ospf 1</code></td>
<td><code>router ospf 1</code></td>
<td><code>router ospf 1</code></td>
<td><code>router ospf 1</code></td>
</tr>
<tr>
<td><code>network 10.0.0.0</code></td>
<td><code>network 10.0.0.0</code></td>
<td><code>network 10.0.0.0</code></td>
<td><code>network 10.0.0.0</code></td>
</tr>
<tr>
<td><code>0.0.255.255 area 0</code></td>
<td><code>0.0.255.255 area 0</code></td>
<td><code>0.0.255.255 area 0</code></td>
<td><code>0.0.255.255 area 0</code></td>
</tr>
<tr>
<td><code>end</code></td>
<td><code>end</code></td>
<td><code>end</code></td>
<td><code>end</code></td>
</tr>
</tbody>
</table>

Note: OSPF configuration uses wildcard mask to declare range.

PC1:

```
/etc/rc.d/init.d/zebra start
/etc/rc.d/init.d/ospfd start
telnet localhost 2604
    pw = root
enable
conf t
router ospf
network 10.0.0.0/16 area 0
router-id 10.0.0.1
no passive eth0
end
show ip ospf database
```

//zebra is our routing daemon
//telnet-ing into port 2604 allows us to configure our ospf service
//no passive tells PC1 to actively participate in route advertisements

Discussion: Why are there two tables in the “show ip ospf database” command?

Test connectivity:

<table>
<thead>
<tr>
<th>R1</th>
<th>PC1 (open a new terminal)</th>
</tr>
</thead>
<tbody>
<tr>
<td>ping 10.0.2.4</td>
<td>ping -c2 10.0.1.1</td>
</tr>
<tr>
<td></td>
<td>ping -c2 10.0.2.4</td>
</tr>
</tbody>
</table>

Discussion: We didn’t do any static routing, how does PC1 know where to issue the ping when the other devices are outside of its subnet prefix?
Take a look at PC's routing table, the routes advertised from OSPF are added to the PC's route table.
PC1: route -n

Now OSPF is enabled on our network we should see how long it takes to converge.

1. On PC1 indefinite ping to R1
2. R1-R2 disconnect serial
3. Wait until ping begins showing “Destination Host Unreachable”
4. Reconnect R1-R2
5. Stop ping once connection is back
   a. count the number of misses (1 second each) - **LOOK AT THE SEQ NUMBER**

```plaintext
64 bytes from 10.0.1.1: icmp_seq=5 ttl=254 time=34.3 ms
64 bytes from 10.0.1.1: icmp_seq=6 ttl=254 time=51.2 ms
64 bytes from 10.0.1.1: icmp_seq=7 ttl=254 time=45.5 ms
64 bytes from 10.0.1.1: icmp_seq=8 ttl=254 time=80.5 ms
From 10.0.0.2 icmp_seq=34 Destination Host Unreachable
From 10.0.0.2 icmp_seq=35 Destination Host Unreachable
From 10.0.0.2 icmp_seq=36 Destination Host Unreachable
ping: sendmsg: Network is unreachable
ping: sendmsg: Network is unreachable
ping: sendmsg: Network is unreachable
ping: sendmsg: Network is unreachable
ping: sendmsg: Network is unreachable
ping: sendmsg: Network is unreachable
ping: sendmsg: Network is unreachable
ping: sendmsg: Network is unreachable
64 bytes from 10.0.1.1: icmp_seq=69 ttl=254 time=21.5 ms
64 bytes from 10.0.1.1: icmp_seq=70 ttl=254 time=55.3 ms
64 bytes from 10.0.1.1: icmp_seq=71 ttl=254 time=17.2 ms
64 bytes from 10.0.1.1: icmp_seq=72 ttl=254 time=75.6 ms
```
Setting up OSPF as multiple areas

We are re-configuring the same topology!

<table>
<thead>
<tr>
<th>R1</th>
<th>R2</th>
<th>R3</th>
<th>R4</th>
</tr>
</thead>
<tbody>
<tr>
<td>conf t</td>
<td>conf t</td>
<td>conf t</td>
<td>conf t</td>
</tr>
<tr>
<td>router ospf 1</td>
<td>router ospf 1</td>
<td>router ospf 1</td>
<td>router ospf 1</td>
</tr>
<tr>
<td>no network 10.0.0.0</td>
<td>no network 10.0.0.0</td>
<td>no network 10.0.0.0</td>
<td>no network 10.0.0.0</td>
</tr>
<tr>
<td>0.0.255.255 area 0</td>
<td>0.0.255.255 area 0</td>
<td>0.0.255.255 area 0</td>
<td>0.0.255.255 area 0</td>
</tr>
<tr>
<td>network 10.0.1.0</td>
<td>network 10.0.1.0</td>
<td>network 10.0.0.0</td>
<td>network 10.0.2.0</td>
</tr>
<tr>
<td>0.0.0.255 area 1</td>
<td>0.0.0.255 area 1</td>
<td>0.0.0.255 area 2</td>
<td>0.0.0.255 area 2</td>
</tr>
<tr>
<td>end</td>
<td>end</td>
<td>end</td>
<td>end</td>
</tr>
</tbody>
</table>

PC1:

```
conf t
router ospf
no network 10.0.0.0/16 area 0
network 10.0.0.0/24 area 0
end
```

Test connectivity:
<table>
<thead>
<tr>
<th>R1</th>
<th>PC1 (open a new terminal)</th>
</tr>
</thead>
<tbody>
<tr>
<td>ping 10.0.2.4</td>
<td>ping -c2 10.0.1.1</td>
</tr>
<tr>
<td></td>
<td>ping -c2 10.0.2.4</td>
</tr>
</tbody>
</table>

Discussion: Area 0 is our network backbone. What is the point of configuring the network this way? Think scalability and AS…. What are the new roles of R2 and R3? (ABR)

Other useful tools for your lab write up:

**IOS:** show ip ospf neighbor

**wireshark:** filter for OSPF message and pay attention to how routes are updated and how they propagate through the network.