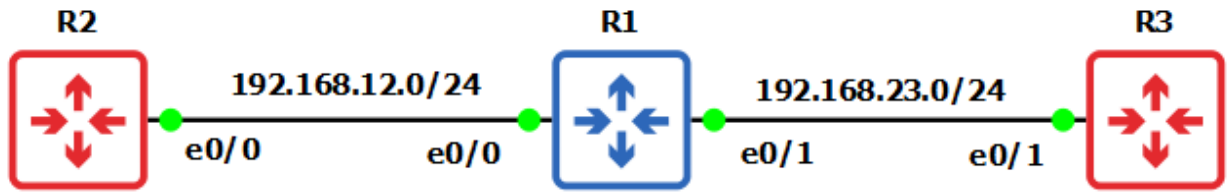
## Debug and Conditional Debug:

- Debugs are useful tools when troubleshooting specific issues in Cisco devices.
- Cisco IOS debugging is very useful feature for the troubleshooting various issues.
- Debug provides invaluable, real-time information for troubleshooting a problem.
- Debugging must be run on Cisco devices with caution as it is very CPU intensive.
- It may result in Router/Switch crash if low CPU memory is available on the device.
- Before debugging, look at your CPU load with the show processes cpu command.
- Verify that you have plenty of CPU available before beginning debugs command.
- Routers can also log messages to an internal buffer to an external syslog server.
- Debug commands help to show the real time information on Cisco device.
- Debug facility used to keep track in case of events or protocol errors in devices.
- Debug facility used to keep track the internal messages taking place in devices.
- Conditional debug is very useful to filter out some of the debug information.
- Conditional debug is a powerful tool to filter out unnecessary debug output.
- It allows IOS to display information that matches only specified parameters.
- Conditionally debugging generates debugging messages matching given condition.
- Such as Conditionally debugging messages for one interface or sub-interface.
- It allows to show debug info that matches certain interface, MAC address etc.
- Access-list can be referenced in the debug ip packet command to narrow down.
- Turn on debugging on Cisco devices for all interfaces that meet specified conditions.
- Debugs can produce a huge amount of information, which overload the Router.
- Use conditional debug to limit information based on specific interface, or protocol.
- Using no debug all or undebug all does not remove the condition debugging.

| Debugging & Condition Debugging | |
|---|---|
| R1# debug ip icmp | R1#debug condition ? |
| R1# debug ip packet | R1#debug condition interface f0/0 |
| R1# debug interface | R1#undebug condition interface f0/0 |
| R1# debug eigrp packets hello | R1#show debugging |
| R1#show debug condition | R1#no debug all |




When Enable RIP debugging on R1. Showing RIP debug information from both interfaces.

```
R1#debug ip rip
RIP protocol debugging is on
R1#
*Mar 16 16:47:42.034: RIP: received v2 update from 192.168.23.3 on Ethernet0/1
*Mar 16 16:47:42.034:      23.0.0.0/8 via 0.0.0.0 in 1 hops
R1#
*Mar 16 16:47:46.116: RIP: received v2 update from 192.168.12.2 on Ethernet0/0
*Mar 16 16:47:46.116:      23.0.0.0/8 via 0.0.0.0 in 1 hops
```

If only, want to see the debug information from one interface then use a debug condition. Using debug condition only can see RIP debug information from the Ethernet 0/0 interface.

```
R1#debug condition int e0/0
R1#debug ip rip
RIP protocol debugging is on
R1#
*Mar 16 16:50:33.816: RIP: received v2 update from 192.168.12.2 on Ethernet0/0
*Mar 16 16:50:33.816:      23.0.0.0/8 via 0.0.0.0 in 1 hops
R1#
*Mar 16 16:50:45.207: RIP: sending v2 update to 224.0.0.9 via Ethernet0/0 (192.168.12.1)
*Mar 16 16:50:45.207: RIP: build update entries
*Mar 16 16:50:45.207:   192.168.23.0/24 via 0.0.0.0, metric 1, tag 0
```

To remove condition debug put no in the begging of the command and type yes.

```
R1#no debug condition interface e0/0
This condition is the last interface condition set.
Removing all conditions may cause a flood of debugging
messages to result, unless specific debugging flags
are first removed.

Proceed with removal? [yes/no]: yes
Condition 1 has been removed
```