Running Head: RIP

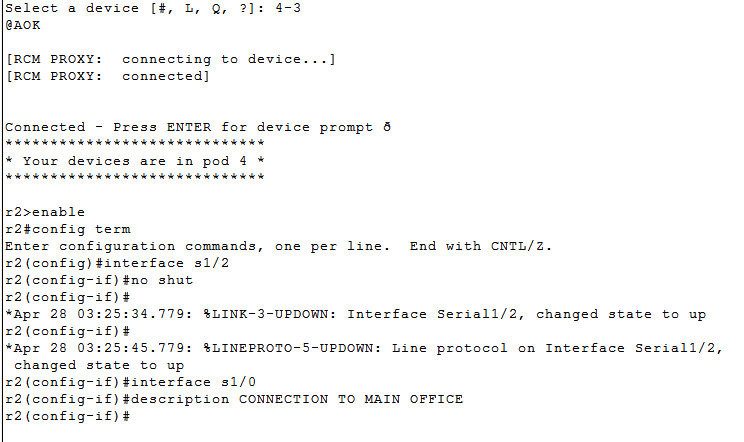
Week 2 Assignment: Setting up RIP Routing Tables

Jered McClure

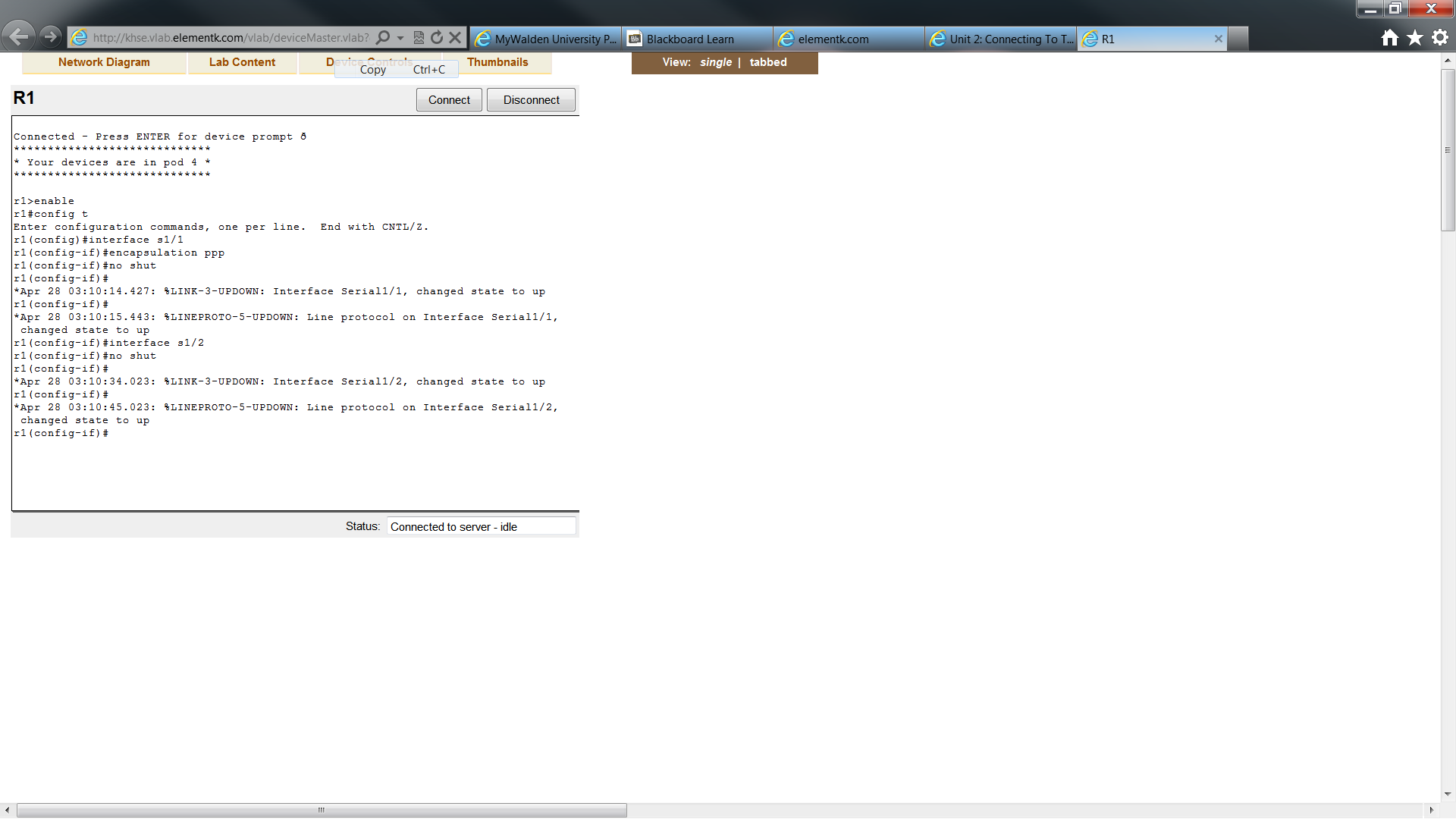
Walden University

Week 2 Assignment: Setting up RIP Routing Tables

Screenshot showing R2 with interface s1/2 enabled and description added to interface s1/0:



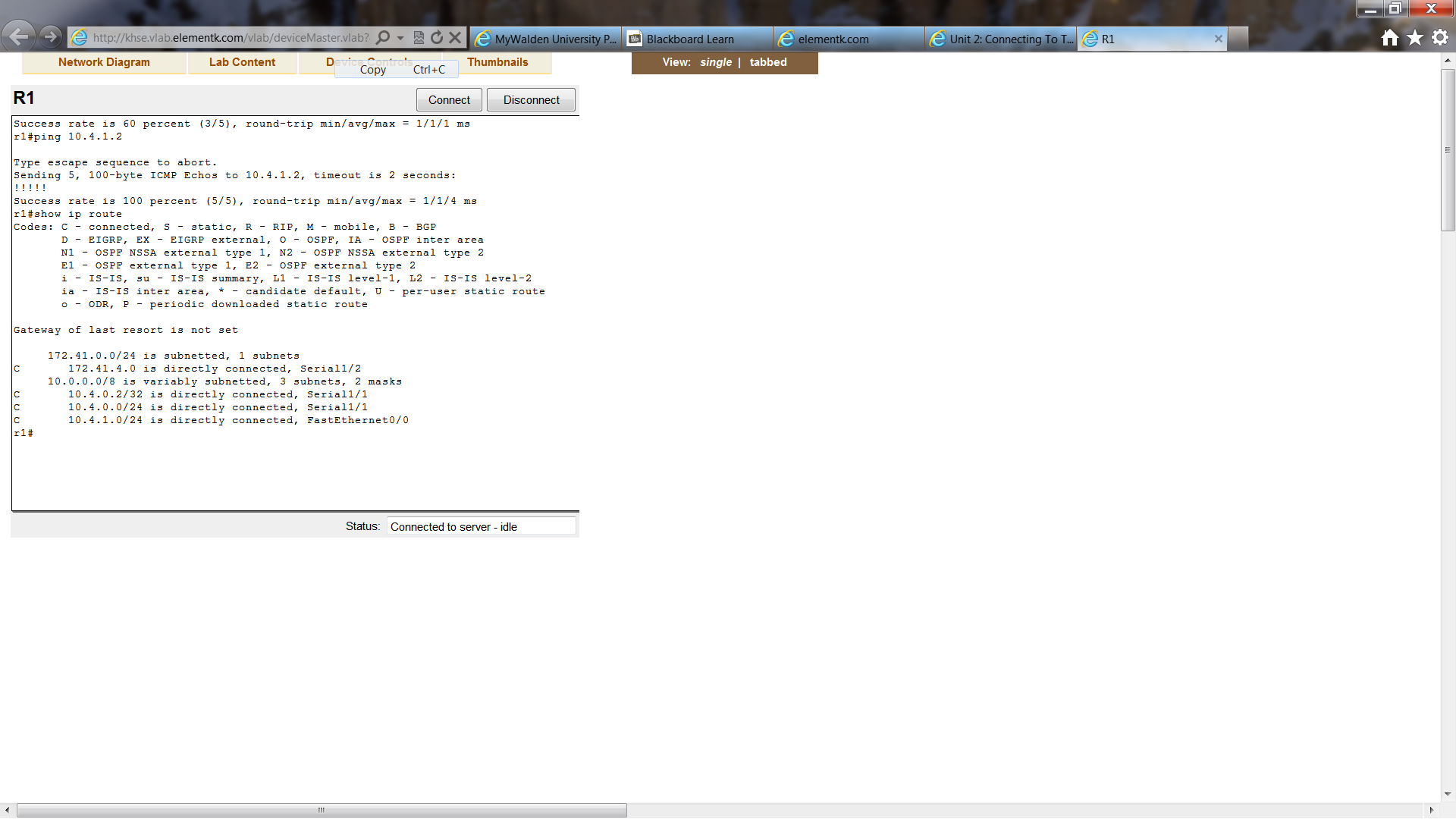
Screenshot of R1 matching the PPP link on R2 and enabling the s1/2 interface.



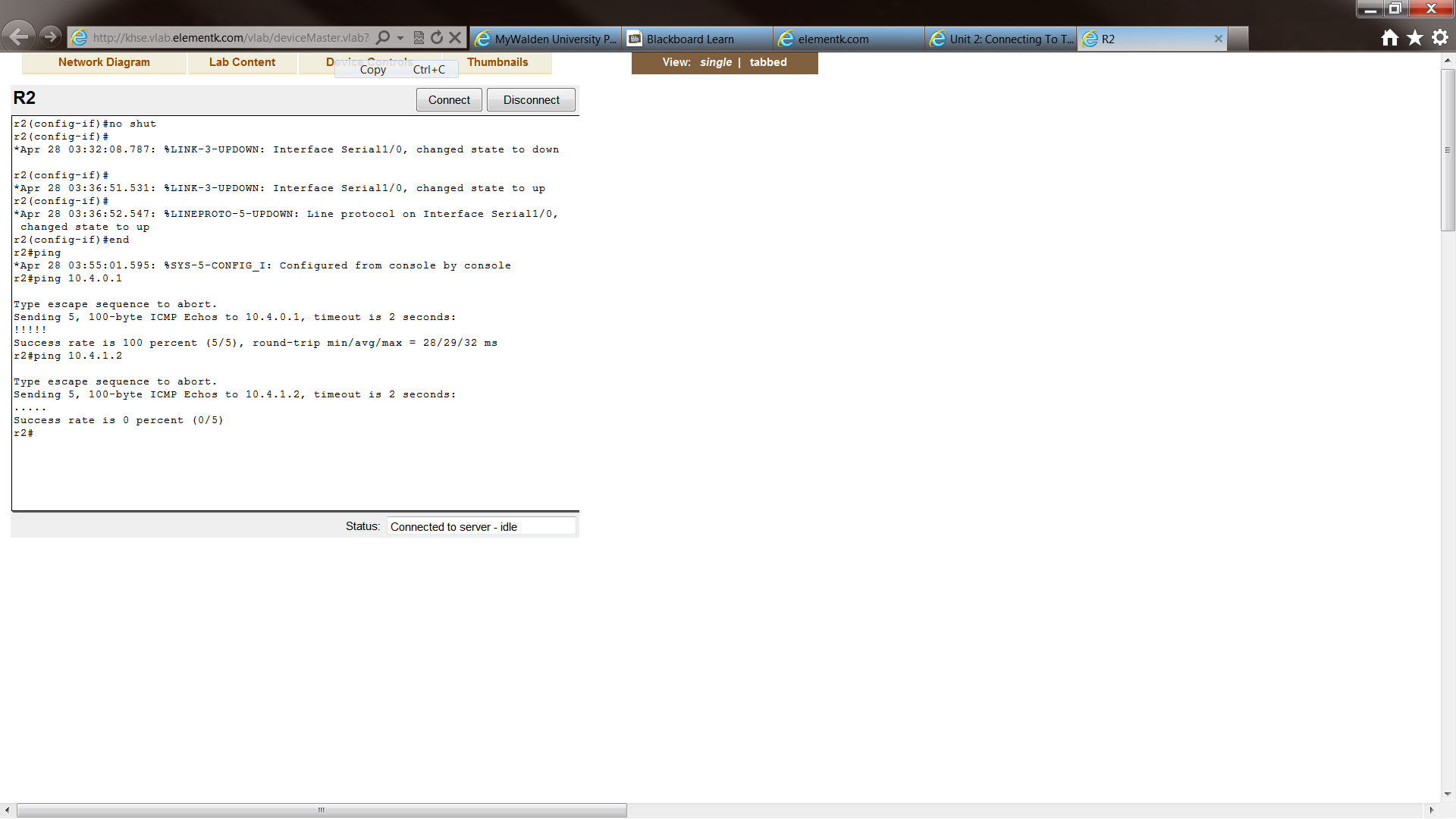
Screenshot showing pings for Branch and Main offices.



Screenshot showing the routing table of R1.



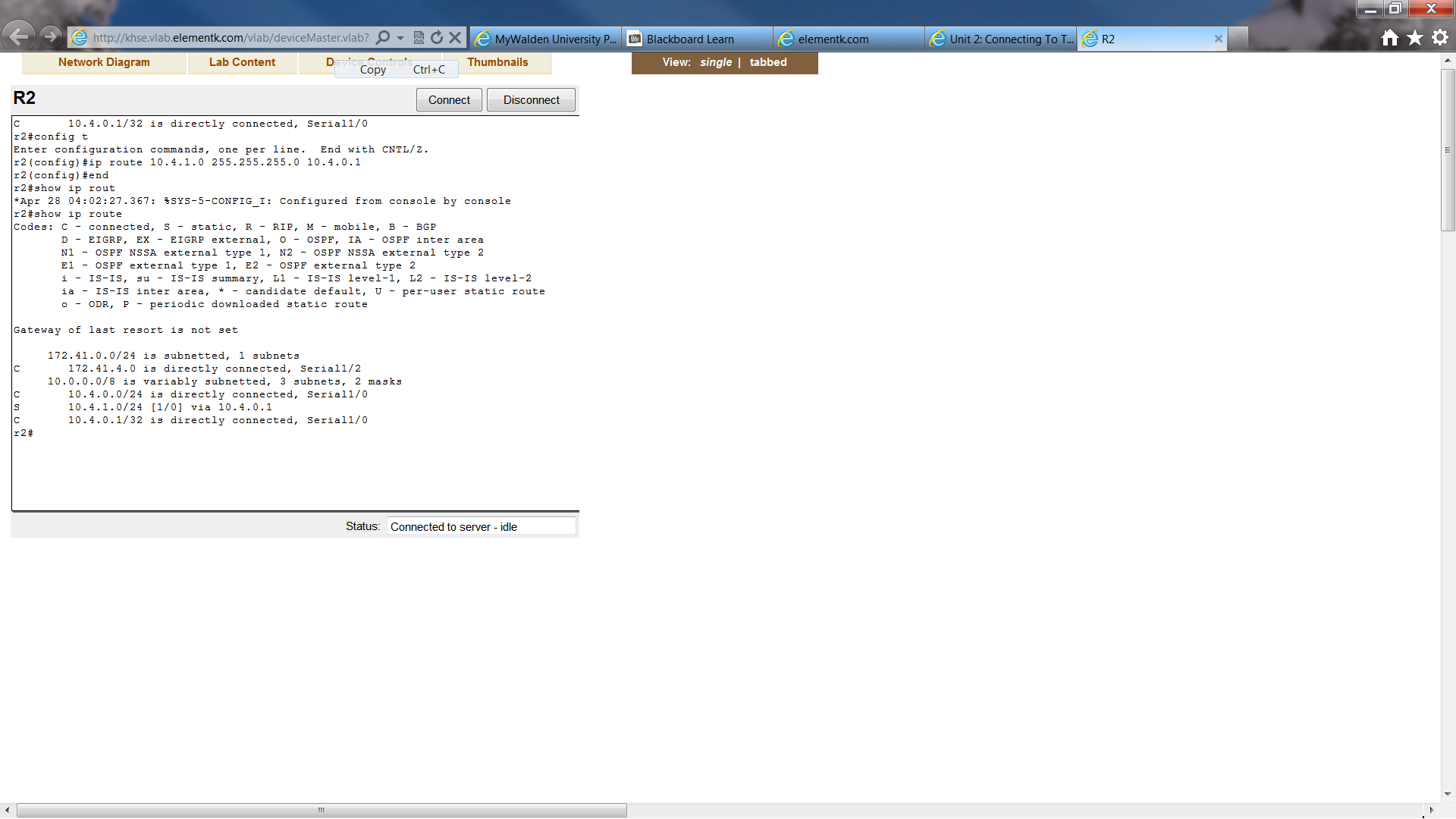
Screenshot showing ping to the main office router from R2 and the Main Office LAN failing from R2 as it is not directly connected and therefore has no route.



Screenshot showing R2’s routing table:



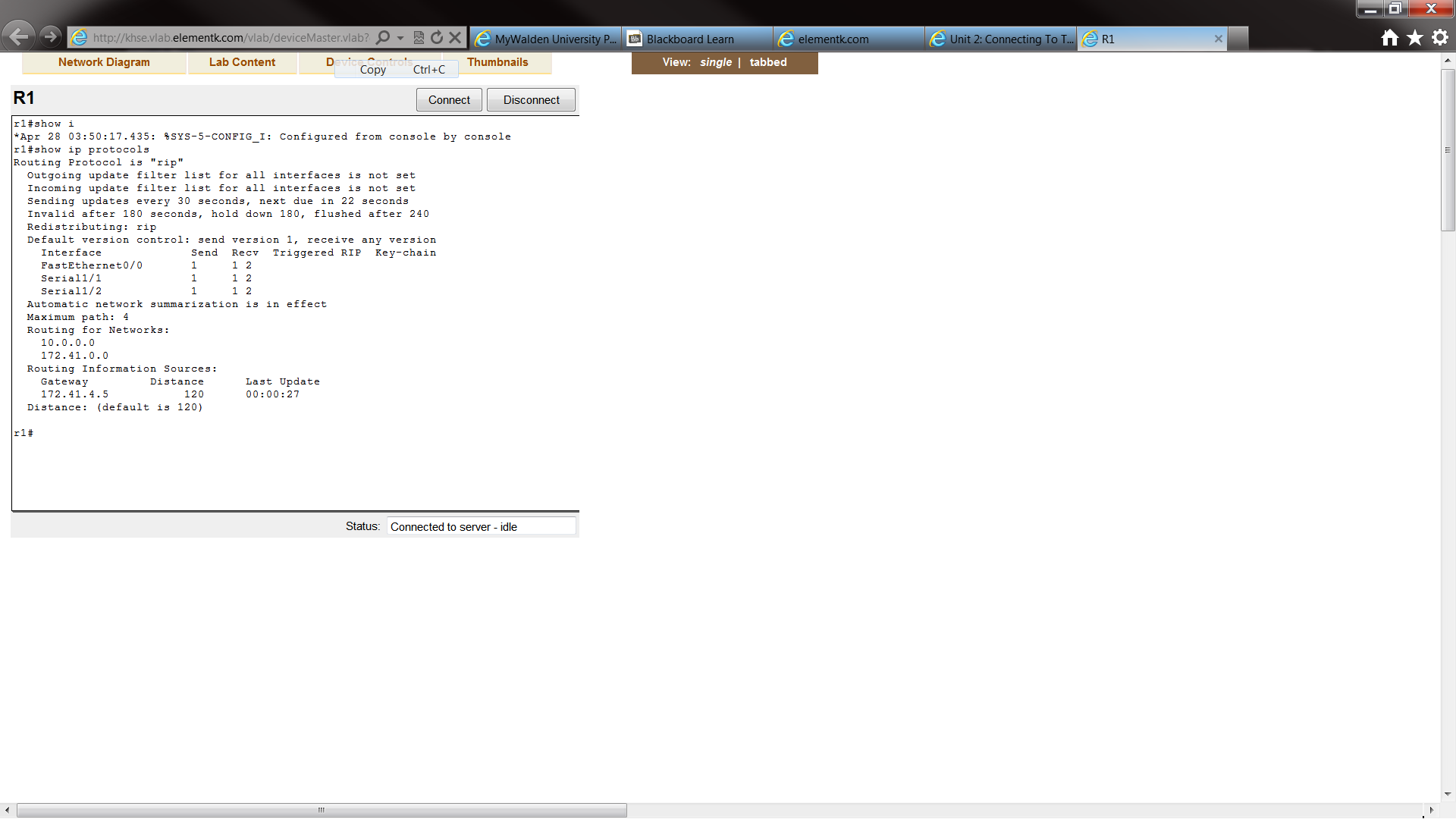
Screenshot showing the static connection to the main office LAN on R2:



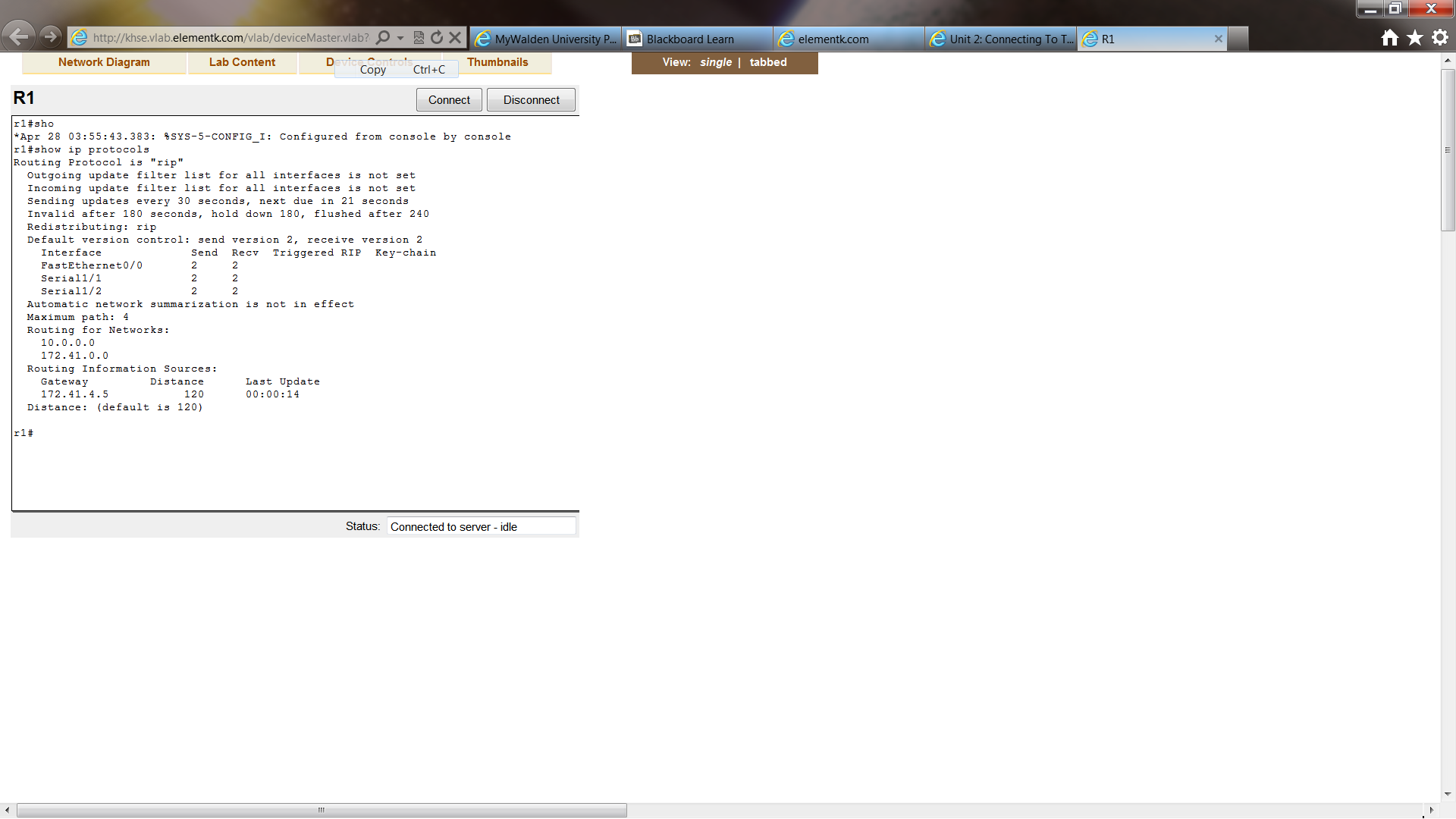
Screenshot showing the configuration of the default route to the main branch and the test ping reply that it is working:



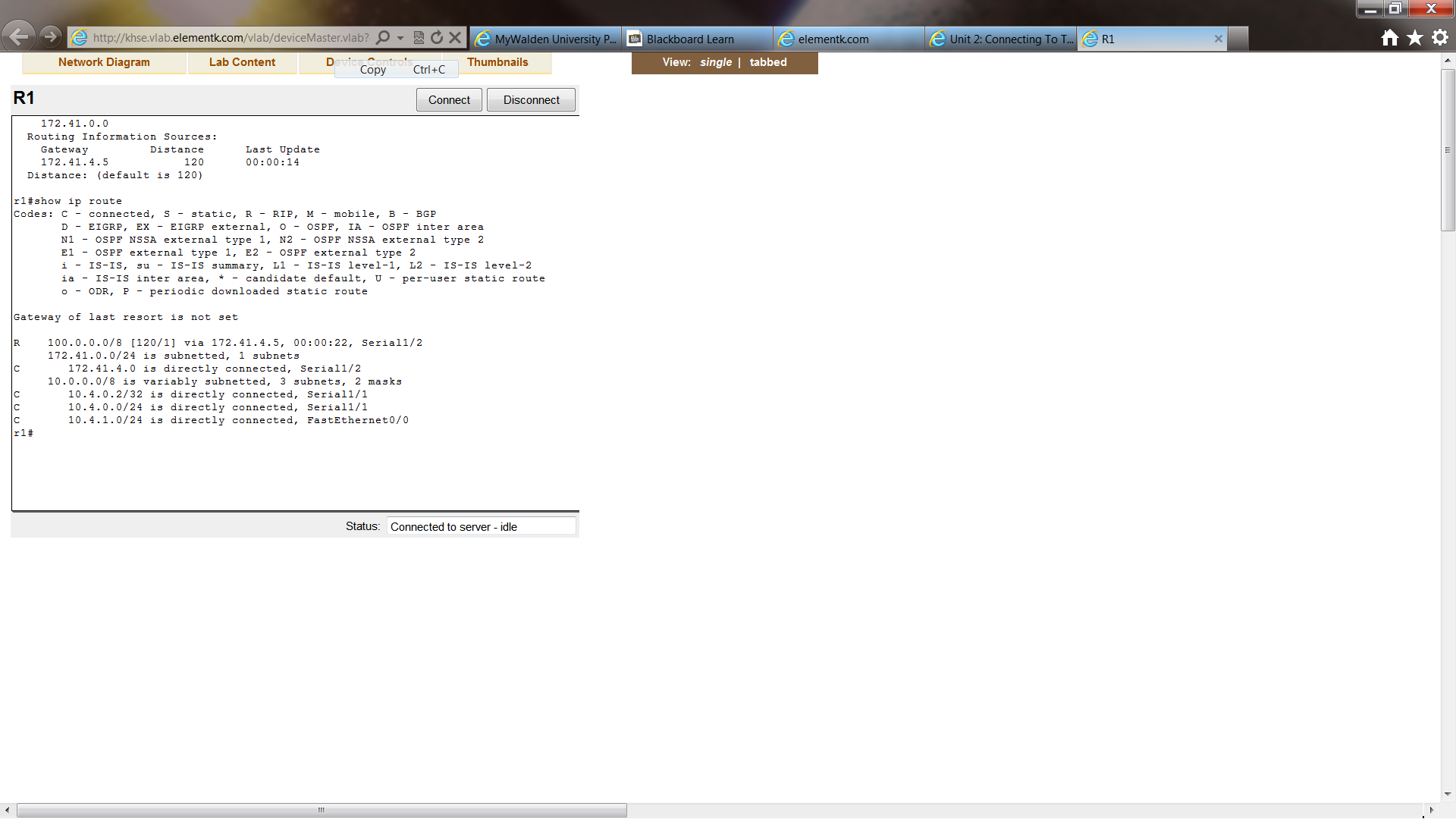
Screenshot showing the routing protocols on R1:



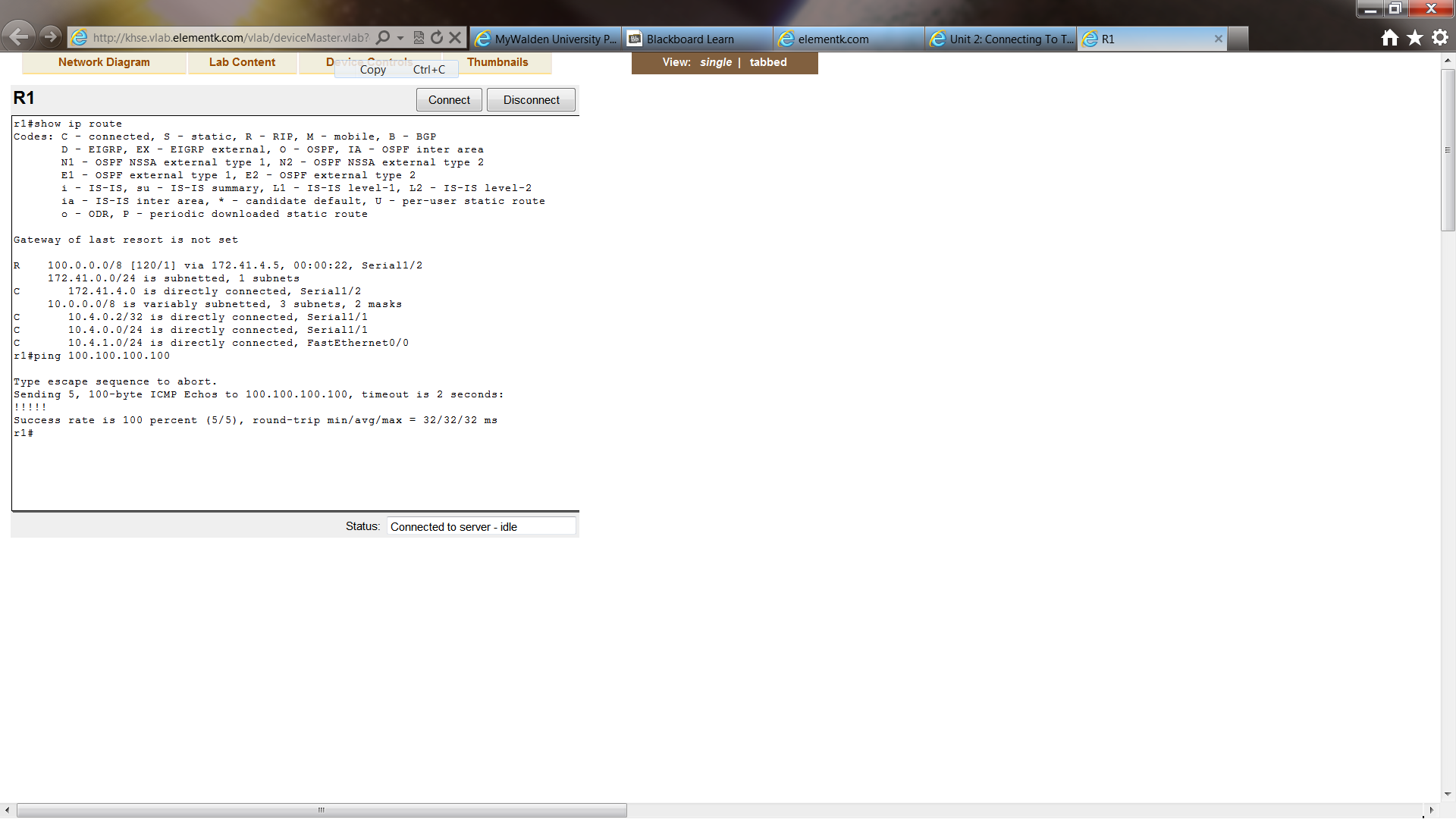
Screenshot showing that automatic network summarization is turned off and version 2 is in effect:



Screenshot showing the routing table of R1 with RIP enabled:



Ping to 100.100.100.100:



Traceroute to 100.100.100.100:



The complexity of setting up a router is unbelievable. Granted, after setting up a few routers the complexity goes away through rote, but for a beginner, or for someone who has no idea about the network layout, it is too much. In fact, the easiest recommendation for setting up large scale networks would be to implement a subjective question menu which then propagates out to any routers which have appropriate rights to the currently connected router.

For instance, the menu could give the options for the different protocols and their functions. Once the appropriate protocol is selected serial ports could be enabled as required automatically. Automation is the key. If the work is menial then spend a few extra hours making it automated and worry about the complicated stuff instead. This is exactly what the movement to GUI operating systems was all about.

In RIP, the cost of each route is determined by the number of hops to that link. However, an administrator can manually change the cost of each hop in a route table to route through higher bandwidth junctions over lower bandwidth junctions, regardless of hop count (Meyers, 2009). That being said, if the hop count is greater than fifteen, RIP cannot see the path.

In a network where the cost of a route is not only determined by the speed of the link but by the monetary value of the link, the metrics for each link would need to be manually input when dealing with RIP. However, if the administrator were to use IS-IS instead, each route would be ranked not just by its number of hop counts, but by the speed, and optionally the expense of those connections (Cisco, 2002, p. 18). This means that the value of all routes would propagate throughout the network. The administrator could then assign monetary values to specific links if so required, thereby lowering their metric value (or raising it) as the need may arise.

Looking back at the routing tables for the business network above, if the network administrator wanted all traffic to the Internet to go through R2 as priority (it has a gateway firewall and cheaper connection to the internet) then the metric of R1 to the internet would need to be higher than the combined metric of R1 to R2 to the internet. Since R1 to R2 to the internet is a metric of 3, R1 to the Internet would need to be set to 4, so that it would only receive traffic when R2 to the internet is unavailable.

There are most definitely paths on the internet longer than 15 hops (note that the work book says 16 hops, but 16 is the metric for “no connection,” therefore there are only 15 effective hops in RIP). For instance, doing a tracert from my computer to [www.whitehouse.gov](http://www.whitehouse.gov) gave me the following:

C:\Windows\system32>tracert -d www.whitehouse.gov

Tracing route to a1128.h.akamai.net [58.26.1.27]

over a maximum of 30 hops:

1 2 ms 1 ms 1 ms 10.0.0.138

2 25 ms 29 ms 25 ms 172.18.221.1

3 27 ms 24 ms 25 ms 172.18.78.34

4 25 ms 25 ms 25 ms 61.9.135.135

5 25 ms 25 ms 25 ms 203.45.29.185

6 28 ms 29 ms 29 ms 203.50.106.130

7 40 ms 39 ms 41 ms 203.50.11.20

8 52 ms 58 ms 60 ms 203.50.11.92

9 82 ms 84 ms 86 ms 203.50.11.17

10 82 ms 82 ms 82 ms 203.50.13.246

11 129 ms 129 ms 129 ms 202.84.141.66

12 133 ms 133 ms 135 ms 202.84.243.82

13 302 ms 303 ms 301 ms 58.27.103.49

14 \* \* \* Request timed out.

15 \* \* \* Request timed out.

16 302 ms 303 ms 303 ms 58.26.1.27

Trace complete.

As for how many hops do I think exist on the internet, that is like asking how long is a piece of string, there are possibly an infinite number of hops depending on how each route on the internet is setup. However, logically, there cannot be an infinite number as a route truncates at some point in time. “A specific TTL number can indicate the maximum range for a packet. For example, zero restricts it to the same host, one to the same subnet, 32 to the same site, 64 to the same region and 128 to the same continent; 255 is unrestricted” (The Linux information Project, 2005).

Because routing tables can contain a large number of redundant routes to subnets, metropolitan routers should not have network summarization turned off. Also, each router needs to have a default route set for external addresses. Another option is to use BGP.

By assigning Autonomous Systems (AS) identifying numbers (ASNs), a routing table can be truncated down to individual areas. Using ASNs shortens the overall size of the routing table at each node down to the ASNs it can see, and the IP tables it possesses. With IPv6 being implemented, routing tables are going to be hit even harder, storing 4 times as much information per address as currently utilized. So making these tables as efficient as possible is a must.

Reference

Cisco. (2002). *Intermediat System-to-Intermediat System Protocol.* Retrieved April 27, 2012, from Cisco: http://www.cisco.com/warp/public/cc/pd/iosw/prodlit/insys\_wp.pdf

Meyers, M. (2009). *CompTIA Netowrk+ Guide to Managing and Troubleshooting Networks* (2nd ed.). McGraw-Hill.

The Linux information Project. (2005, October 14). *Time-to-live Definition.* Retrieved April 27, 2012, from LINFO: http://www.linfo.org/time-to-live.html