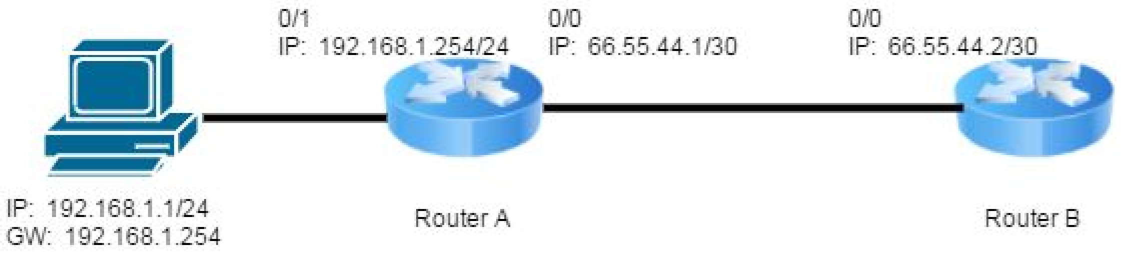
Network Address Translation (NAT) Background:

Network address translation is a feature designed mostly to save public IPv4 address space, allowing for hosts on a LAN to share the same or multiple public IP addresses. This can be done in 3 ways. **Static NAT** is a one to one mapping where a single LAN IP address translates to a single public or external address. **Dynamic NAT** uses pool of public addresses and when an internal host needs to communicate outside the network, an available address is selected from the pool and is used as the host's external IP. Finally, **NAT overload** is a port based translation method where multiple LAN hosts share a single public IP address and traffic is differentiated based on port numbers assigned by the NAT router.

NAT Topology Diagram 1:

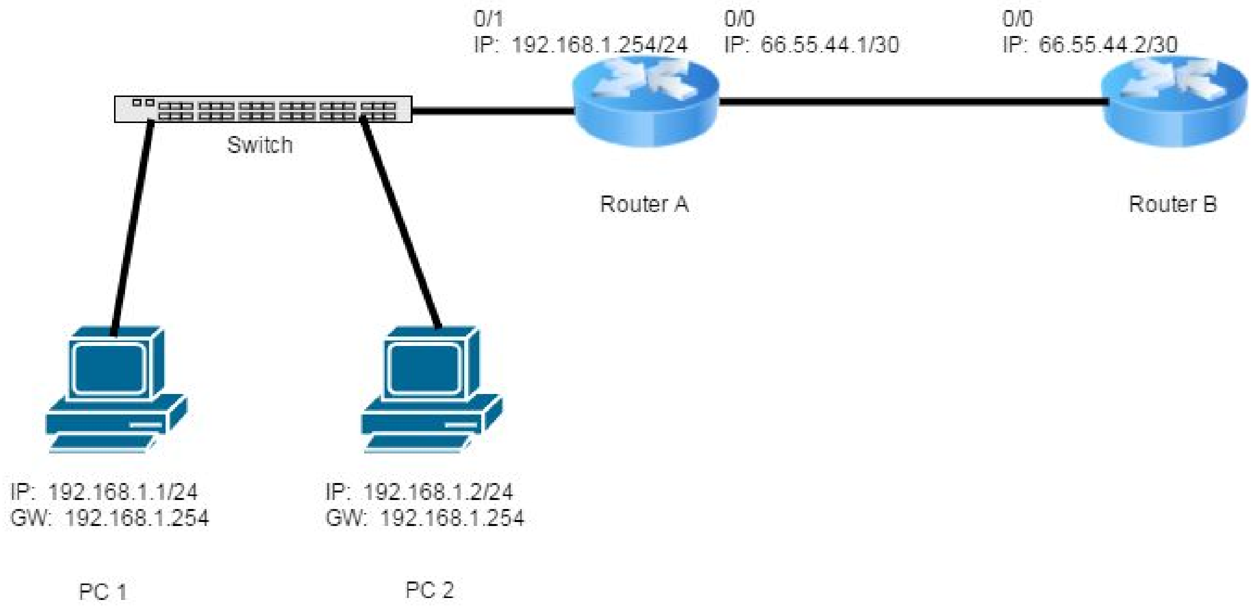


NAT Lab Instructions:

1. Build the topology shown above. Directly connect routers 1 and 2, assigning the IP addresses as shown.
2. Connect a computer to interface 0/1 of router A and assign the IP addresses as shown.
3. Try pinging from the computer to Router B's interface 0/0 address. Why does the ping fail? Hint: use the "`show ip route`" command.

4. Fix this issue by using static NAT to translate the 192.168.1.1 address to the address assigned to Router A's 0/0 interface.
5. Try pinging from the computer to Router B's interface 0/0 address again. Why does the ping succeed now? What is NAT actually doing?
6. Undo the static NAT translation and then move on.

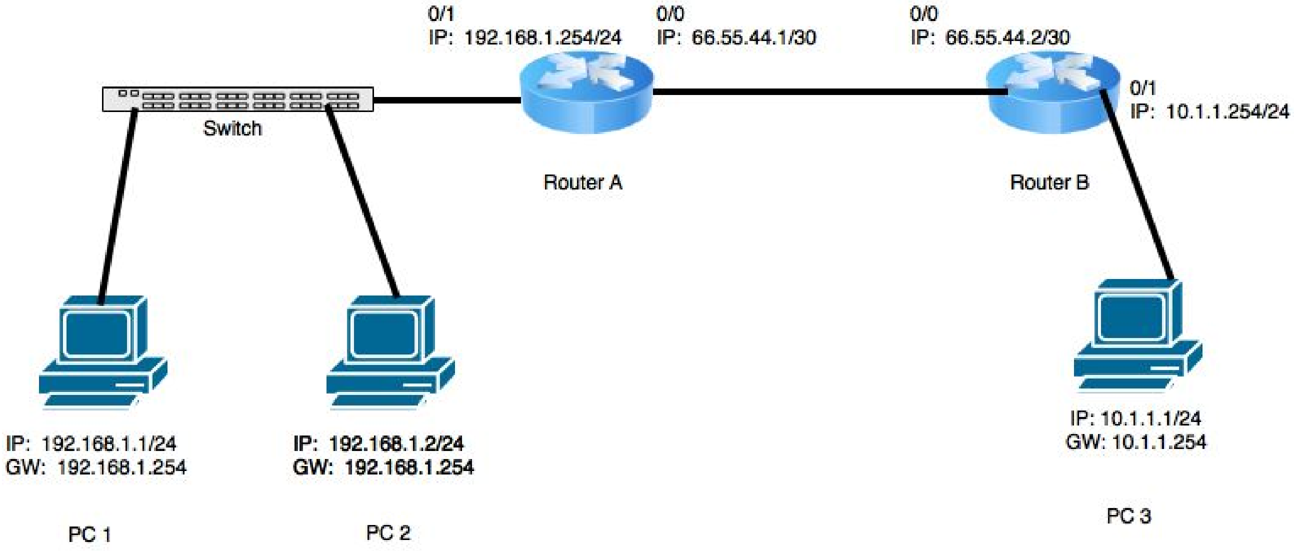
NAT Topology Diagram 2:



7. Modify your topology by adding a switch behind Router A's 0/1 interface and plug two computers into the switch.
8. Next, assign the IP address 192.168.1.2/24 to the second PC you connected.
9. Configure router A to use port address translation (NAT overload), translating all traffic from the 192.168.1.0/24 network to the IP address of router A's 0/0 interface.
10. Test your work by pinging from BOTH computers to router B's 0/0 interface. What is NAT doing in this case? Try running the "`show ip nat translations`" command.

Access Control Lists (ACL) Background:

ACLs are essentially used to classify data. Many usually make the assumption that ACLs are to block access to a network, but that is just one of the many uses of them. An ACL can be used with Quality of Service, for example. But for the purposes of this Build-It-Night we will just be focusing on the use as a security feature. There are two types of ACLs: extended and standard. **Standard** ACLs only filter traffic based on the source IP address. Standard ACLs can be numbered 1-99 and 1300-1999 and are typically applied closest to the destination. **Extended** ACLs filter traffic based on source and destination addresses as well as port numbers and protocols. The valid numbers for extended ACLs are 100-199 and 2000-2699 and are typically applied as close to the source as possible.



ACL Lab Instructions:

1. Configure PC3 and add the extra subnet to Router B as shown in the topology above.
2. Remove NAT and add static routes to each router using the following command:
   a. ip route 0.0.0.0 0.0.0.0 65.55.44.X (where X is 1 or 2)
3. Set Up a Filezilla FTP server on PC3.

4. Try to ping from PC1 and PC2 to PC3.
5. Try to browse the FTP share on PC3 from both PC1 and PC2.
6. Create a **standard** ACL to block PC1 from accessing FTP on PC3.
    a. Where did you apply it?
    b. Does it work?
    c. Can you still ping PC3?
7. Remove the standard ACL
8. Create an **extended** ACL to block PC1 from accessing FTP on PC3.
    a. Where did you apply it?
    b. Does it work?
    c. Can you still ping PC3? (you should be able to if the ACL was written correctly)