Network Security

Assignment 4B, Wednesday, May 16, 2018, version 1.0

Handing in your answers: Submission via Blackboard (http://blackboard.ru.nl)

Deadline: Wednesday, May 30, 23:59:59 (midnight)

Teaching assistants. Please email *all* of us if you have a question.

- Pol Van Aubel <pol.vanaubel@cs.ru.nl>
- Daan Sprenkels <dsprenkels@science.ru.nl>
- Wouter Kuhnen <w.j.a.kuhnen@student.ru.nl>

The next lecture is in two-and-a-half weeks time. Therefore, assignment 4 is rather large. We split this assignment in three parts so that the grading is distributed somewhat. However, this does not mean that each part has an equal amount of work. When done with one part, move immediately on to the next. Remember that 6 hours per week is the intended time investment for a 3EC course. Don't spend more than 6 hours on part 4a, and make sure that you have 6 hours left for part 4c.

Please turn in all your work in plain text files (program source files are also plain text). If you prefer a document with formatting for whatever reason, like including images, use the PDF format to turn in your work (most editors allow you to export to PDF). Note that it's okay to include images separately and then refer to them from within the text files.

This assignment consists only of theoretical questions.

1. Create a folder called `exercise1` to hold the answers for this exercise.

   Although the lecture focusing on VPNs is in a few weeks time, there are some aspects which we can already cover in assignments. As you should have seen in assignment 4a, sshuttle uses iptables to direct its traffic into the VPN process. There exists another type of VPN software that provides a virtual network interface to route traffic into. OpenVPN and wireguard are both of this type.

   A machine's network stack decides what interface to put outgoing traffic into based on routes. Therefore, when using such a VPN, the routing table contains additional routes to route traffic over the VPN.

   However, this is not as straightforward as simply adding a single route. Therefore, we will examine this approach in this exercise. Bear in mind that when a route has a `via` entry, this traffic is sent to that gateway over the specified interface, `dev`, for routing. If a route does not have a `via` entry, the traffic is sent directly to the target over the specified interface. A routing table therefore usually contains at least two routes: one that specifies how to reach the gateway, and one that specifies how to reach the rest of the internet through that gateway. The latter one is the `default` route.

   For example, my IP address is 145.116.128.31/22. When I'm not connected to my VPN, my routing table looks something like this:

   ```
   $ ip r show
   default via 145.116.128.1 dev wlp3s0
   145.116.128.0/22 dev wlp3s0  proto kernel  scope link  src 145.116.128.31
   ```

   For all the following questions, keep in mind that if routes overlap, the more specific route (i.e. one that applies to a smaller network, a smaller number of hosts) overrides more generic routes. So a route with a netmask of /24 overrides a route covering the same hosts with a netmask of /8. The default route is effectively a route of 0.0.0.0/0.

   Let's say that my VPN runs on a machine with IP address 198.51.100.42. When I connect to my VPN, a new interface (`tap0`) is created, and the routing table is changed (I have slightly altered the output for clarity):

```
$ ip r show
 1. 0.0.0.0/1 via 10.50.9.1 dev tap0
 2. 128.0.0.0/1 via 10.50.9.1 dev tap0
 3. 10.50.9.0/24 dev tap0  proto kernel  scope link  src 10.50.9.60

 4. 10.0.0.0/8 via 145.116.128.1 dev wlp3s0
 5. 172.16.0.0/12 via 145.116.128.1 dev wlp3s0
 6. 192.168.0.0/16 via 145.116.128.1 dev wlp3s0

 7. default via 145.116.128.1 dev wlp3s0
 8. 131.174.117.20 via 145.116.128.1 dev wlp3s0
 9. 145.116.128.0/22 dev wlp3s0  proto kernel  scope link  src 145.116.128.31
10. 198.51.100.42 via 145.116.128.1 dev wlp3s0
```

Other relevant information is in the DHCP leases I got:

```
$ dhcpcd --dumplease wlp3s0
dhcp_server_identifier=131.174.117.20
domain_name_servers=131.174.117.20
ip_address=145.116.128.31
network_number=145.116.128.0
routers=145.116.128.1
subnet_cidr=22
subnet_mask=255.255.252.0

$ dhcpcd --dumplease tap0
dhcp_server_identifier=10.50.9.1
ip_address=10.50.9.60
network_number=10.50.9.0
subnet_cidr=24
subnet_mask=255.255.255.0
```

Internally the VPN uses the network 10.50.9.0/24, as can be seen in the dhcp lease for `tap0`. The dhcp protocol requires periodic communication with the dhcp server to keep the address lease active.

In these questions, when asked "where traffic goes", please answer with the gateway IP address and the interface. If traffic is not sent to a gateway, answer with the direct IP address.

(a) Look at routes 1, 2, and 7. Where does traffic not matched by any of the other routes go, and why?

Route 9 is one of the two original routes, also present when the VPN is not active. Routes 4–6 are always added by my VPN setup script. Note the IP ranges used, and try to imagine the usage scenario for a VPN.

(b) Explain what these routes accomplish. What traffic do they match, where does that traffic go, and why?

(c) Look at the dhcp lease for the `tap0` interface. Explain why route 3 is necessary, in light of what routes 4–6 accomplish.

(d) Look at route 8 and the dhcp lease for the `wlp3s0` interface. Why is route 8 necessary? What happens if it is not present?

(e) Route 10 is, from a functionality point of view, the most important route present. Explain what it does, and what would happen if it was *not* present.

(f) Try to explain what happens when I connect e.g. to an SSH server running on the same machine as the VPN server. Does that traffic get tunneled or not? Explain why.

2. Create a folder called `exercise2` for your answers.

   Take a look at RFC 5508, "NAT Behavioral Requirements for ICMP" (`http://tools.ietf.org/rfc/rfc5508.txt`). Read sections 2 (especially the part on "ICMP Message Classification"), 3, and 4.

   Answer the following 4 questions not from a security, but from a technical point of view.

   (a) What does a NAT do with inbound ICMP Error messages which *do* belong to an existing NAT session, and why?

   (b) Why does a NAT drop inbound ICMP Error messages which *do not* belong to an existing NAT session?

   (c) What does a NAT do with outbound ICMP Error messages which *do* belong to an existing NAT session, and why?

   (d) Why does a NAT drop outbound ICMP Error messages which *do not* belong to an existing NAT session?

   Now read section 10, looking at 9 for the requirements.

   (e) Try to summarize the security considerations of NAT for ICMP. Explain the security concerns, and explain how they are mitigated.

   (f) Suppose an incompetent network administrator decides that ICMP is evil and just blocks all ICMP packets from entering and leaving the network. Give two examples of things that would break.

   Section 9 mentions that a NAT should not support ICMP Source Quench messages. ICMP Source Quench is an ICMP message type which was proposed for congestion control in the early days of networking. A congested router would send a Source Quench message to hosts sending a lot of traffic in order to get them to back off. ICMP Source Quench messages were explicitly deprecated in RFC 6633, "Deprecation of ICMP Source Quench Messages" (`https://tools.ietf.org/html/rfc6633`).

   (g) Why is ICMP Source Quench not an effective method against distributed denial of service attacks against a router?

   (h) Think of a way in which you could have used ICMP Source Quench to perform a denial of service attack against honest communicating hosts in a network.

3. Place the files and directories `exercise1` and `exercise2`, and all their contents, in a folder called `netsec-assignment4b-STUDENTNUMBER1-STUDENTNUMBER2`. Replace `STUDENTNUMBER1` and `STUDENTNUMBER2` by your respective student numbers, and accommodate for extra / fewer student numbers. Make a `tar.gz` archive of the whole `netsec-assignment4b-STUDENTNUMBER1-STUDENTNUMBER2` directory and submit this archive in Blackboard.