1) Would an arbitrary host from the Internet (e.g., 198.51.100.1) be able to establish a new TCP connection to the e-mail server (192.168.1.1) on port 25? If yes, which rule accepts this packet? If no, which rule drops it?

Yes, rule 2 says that for a NEW tcp connection that any host that tries to connect to 192.168.1.1that matches any port is allowed to connect.

2) Would an arbitrary host from the Internet (e.g., 198.51.100.1) be able to establish a new TCP connection to the web server (192.168.1.2) on port 25? If yes, which rule (2—9) accepts this packet? If no, which rule (2—9) drops it?

No, rule 9 would drop it

3) Would the e-mail server (192.168.1.1) be able to establish a new TCP connection to an arbitrary host on the Internet (e.g., 198.51.100.1) on port 80? If yes, which rule accepts this packet? If no, which rule drops it?

Yes, rule 8 states that for a NEW tcp connection that any host between 192.168.1.0 and 192.168.1.254 that tries to connect to any destination that matches any port is allowed to connect.

1. ESTAB 0.0.0.0/0 -> 0.0.0.0/0 : ANY => ACCEPT

2. NEW 0.0.0.0/0 -> 192.168.1.1 : ANY => ACCEPT wrong destination

3. NEW 0.0.0.0/0 -> 192.168.1.2 : 80 => ACCEPT wrong destination

4. NEW 192.168.1.1 -> 0.0.0.0/0 : 25 => ACCEPT wrong port

5. NEW 192.168.1.0/24 -> 192.168.2.1 : 8080 => ACCEPT wrong destination

6. NEW 192.168.2.0/24 -> 0.0.0.0/0 : ANY => ACCEPT wrong source

7. NEW 192.168.1.0/24 -> 192.168.2.0/24 : ANY => DROP wrong destination

8. NEW 192.168.1.0/24 -> 0.0.0.0/0 : ANY => ACCEPT accept

9. OTHERWISE => DROP

4) Would the web server (192.168.1.2) be able to establish a new TCP connection to the database server (192.168.2.1) on port 80? If yes, which rule accepts this packet? If no, which rule drops it?

No, rule 7 states that for a NEW tcp connection that any host between 192.168.1.0 and 192.168.1.254 that tries to connect to any destination from 192.168.2.0 and 192.168.2.254 that matches any port is not allowed to connect.

1. ESTAB 0.0.0.0/0 -> 0.0.0.0/0 : ANY => ACCEPT

2. NEW 0.0.0.0/0 -> 192.168.1.1 : ANY => ACCEPT wrong destination

3. NEW 0.0.0.0/0 -> 192.168.1.2 : 80 => ACCEPT wrong destination

4. NEW 192.168.1.1 -> 0.0.0.0/0 : 25 => ACCEPT wrong port

5. NEW 192.168.1.0/24 -> 192.168.2.1 : 8080 => ACCEPT wrong port

6. NEW 192.168.2.0/24 -> 0.0.0.0/0 : ANY => ACCEPT wrong source

7. NEW 192.168.1.0/24 -> 192.168.2.0/24 : ANY => DROP accept

8. NEW 192.168.1.0/24 -> 0.0.0.0/0 : ANY => ACCEPT

9. OTHERWISE => DROP

5) Would the web server (192.168.1.2) be able to establish a new TCP connection to the database server (192.168.2.1) on port 8080? If yes, which rule accepts this packet? If no, which rule drops it?

Yes, rule 5 states that for a NEW tcp connection that any host between 192.168.1.0 and 192.168.1.254 that tries to connect to 192.168.2.1 that matches port 8080 is allowed to connect.

1. ESTAB 0.0.0.0/0 -> 0.0.0.0/0 : ANY => ACCEPT

2. NEW 0.0.0.0/0 -> 192.168.1.1 : ANY => ACCEPT wrong destination

3. NEW 0.0.0.0/0 -> 192.168.1.2 : 80 => ACCEPT wrong destination

4. NEW 192.168.1.1 -> 0.0.0.0/0 : 25 => ACCEPT wrong port

5. NEW 192.168.1.0/24 -> 192.168.2.1 : 8080 => ACCEPT accept

6. NEW 192.168.2.0/24 -> 0.0.0.0/0 : ANY => ACCEPT

7. NEW 192.168.1.0/24 -> 192.168.2.0/24 : ANY => DROP

8. NEW 192.168.1.0/24 -> 0.0.0.0/0 : ANY => ACCEPT

9. OTHERWISE => DROP

6) Would the e-mail server (192.168.1.1) be able to establish a new TCP connection to the database server (192.168.2.1) on port 25? If yes, which rule accepts this packet? If no, which rule drops it?

Yes, rule 4 states that for a NEW tcp connection that for host 192.168.1.1 that tries to connect to 192.168.2.1 that matches port 25 is allowed to connect.

1. ESTAB 0.0.0.0/0 -> 0.0.0.0/0 : ANY => ACCEPT

2. NEW 0.0.0.0/0 -> 192.168.1.1 : ANY => ACCEPT wrong destination

3. NEW 0.0.0.0/0 -> 192.168.1.2 : 80 => ACCEPT wrong destination

4. NEW 192.168.1.1 -> 0.0.0.0/0 : 25 => ACCEPT accept

5. NEW 192.168.1.0/24 -> 192.168.2.1 : 8080 => ACCEPT

6. NEW 192.168.2.0/24 -> 0.0.0.0/0 : ANY => ACCEPT

7. NEW 192.168.1.0/24 -> 192.168.2.0/24 : ANY => DROP

8. NEW 192.168.1.0/24 -> 0.0.0.0/0 : ANY => ACCEPT

9. OTHERWISE => DROP

7) Would the database server (192.168.2.1) be able to establish a new TCP connection to the web server (192.168.1.2) on port 8080? If yes, which rule accepts this packet? If no, which rule drops it?

Yes, rule 6 states that for a NEW tcp connection that for any host from 192.168.2.0 to 192.168.2.254 that tries to connect to any destination that matches any port is allowed to connect.

1. ESTAB 0.0.0.0/0 -> 0.0.0.0/0 : ANY => ACCEPT

2. NEW 0.0.0.0/0 -> 192.168.1.1 : ANY => ACCEPT wrong destination

3. NEW 0.0.0.0/0 -> 192.168.1.2 : 80 => ACCEPT wrong port

4. NEW 192.168.1.1 -> 0.0.0.0/0 : 25 => ACCEPT wrong port

5. NEW 192.168.1.0/24 -> 192.168.2.1 : 8080 => ACCEPT wrong source

6. NEW 192.168.2.0/24 -> 0.0.0.0/0 : ANY => ACCEPT accept

7. NEW 192.168.1.0/24 -> 192.168.2.0/24 : ANY => DROP

8. NEW 192.168.1.0/24 -> 0.0.0.0/0 : ANY => ACCEPT

9. OTHERWISE => DROP

8) Suppose that you have to ensure that all e-mail traffic from the internal network goes through the e-mail server by extending the firewall rules. Formally, the following requirements must be satisfied:

a. hosts from the internal network should be able to establish new TCP connections to the e-mail server on port 25;

b. hosts from the internal network should not be able to establish new TCP connections to any host on the Internet on port 25;

c. other network traffic should be unaffected by your extension. Write a new firewall rule (or rules) using the syntax of this problem that satisfy the above requirements! Specify before which existing rule (1—9) should the new rule (or rules) be inserted!

These rules can be placed anywhere before rule 9 suchs as before rule 2

NEW 192.168.2.0/24 -> 192.168.1.1 : 25 => ACCEPT

This rule has to be placed before the 3rd rule in order to deny all traffic to the webserver

NEW 192.168.2.0/24 -> 192.168.1.2 : 25 => DROP

Write a Snort rule that raises an alarm (with the message “Yikes!”) when someone not from the internal network tries to access the page wp-login.php on the webserver using HTTP! Try to minimize the number of false matches.

alert tcp !192.168.1.0/24 any -> wp-login.php any (msg: “Yikes!”;)

Which of the variables var1, var2, …, var8 should be considered tainted (e.g., for the purpose of copy-pasting it into a system command to be executed) and why?

The source of the value of var1 is untrustworthy because it is a user input, therefore the variable is tainted

The source of the value of var2 is trustworthy because it is a constant string which is already set which is not modified by any outside sources and does not do anything on its own.

The source of the value of var3 is trustworthy because it does involve a tainted variable but is not used in any command that modifies files, directories, or processes and is only used to evaluate a boolean expression.

The source of the value of var4 is untrustworthy because the variable has its value derived from a concatenation operator from var1, a tainted variable, which means is also tainted.

The source of the value of var5 is trustworthy because it trims a boolean expression which cannot be changed by any external source

The source of the value of var6 is untrustworthy because the variable has its value derived from a concatenation operator from var4, a tainted variable, which means it is also tainted, the capitalization operation means nothing.

The source of the value of var7 is untrustworthy because the variable has its value derived from a string operator from a tainted variable which means is also tainted as var1 can be used to execute a system command and the substring uses 0 which is negligible.

The source of the value of var8 is untrustworthy because it outputs the path in var2, var2 capitalized, var1 which is tainted and var2 which means that a tainted variable is run and a path is outputted which can be used to modify data using the path.