**MIS 6330: Cybersecurity Fundamentals**

**Individual Homework 7**

1. SMTP is the standard protocol for transferring mails over a TCP connection. The server listens to TCP port 25 and the user typically uses a port above 1023. Suppose you wish to build a packet filter ruleset allowing certain inbound and outbound SMTP traffic. All other traffic is thus blocked. You generate the following ruleset:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Rule | Direction | Source | Destination | Protocol | Dest Port | Action |
| A | In | External | Internal | TCP | 25 | Permit |
| B | Out | Internal | External | TCP | >1023 | Permit |
| C | Out | Internal | External | TCP | 25 | Permit |
| D | In | External | Internal | TCP | >1023 | Permit |
| E | Either | Any | Any | Any | Any | Deny |

* 1. Your host has the IP address 172.16.1.1. A benign outside source tries to send an e-mail to your host, using SMTP, from a remote host with the IP address 192.168.3.4. Four typical packets for this scenario are as follows:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Packet | Direction | Source | Destination | Protocol | Dest Port | Action |
| 1 | In | 192.168.3.4 | 172.16.1.1 | TCP | 25 | Permitted |
| 2 | Out | 172.16.1.1 | 192.168.3.4 | TCP | 1234 | Permitted |
| 3 | Out | 172.16.1.1 | 192.168.3.4 | TCP | 25 | Permitted |
| 4 | In | 192.168.3.4 | 172.16.1.1 | TCP | 1357 | Permitted |

* 1. Someone from the outside world (10.1.2.3) attempts to open a connection from local port 5150 on a remote host to the Web proxy server on on port 8080 on one of your local hosts (172.16.3.4) to carry out an attack. The packets involved are as follows:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Packet | Direction | Source | Destination | Protocol | Dest Port | Action |
| 1 | In | 10.1.2.3 | 172.16.3.4 | TCP | 8080 | Permitted |
| 2 | Out | 172.16.3.4 | 10.1.2.3 | TCP | 5150 | Permitted |

Yes, this attack will succeed as *packet filters* are stateless filters where only   
the values like direction, source, destination, protocol, etc are checked.  
 In the given case, both packets have values that are permitted by the   
*packet filters* and thereby, those packets will be permitted.

*Packet filters* won’t check the information of state i.e., if host initiated communication   
or not, what is the current state of the communication and so on. Due to this,   
anyone can send packets like this attacker and if it follows the rules of   
*packet filters*, packets will be permitted.

1. *Stateful filters* are more secure than *packet filters* as:
   1. *Packet filters (or stateless filters)* check the source and destination and some other static values for a packet and if they are not restricted by rules, it allows them.  
      This is a problem when an attacker would send packets to your machine with source address that is not restricted by rules of *packet filters* and these packets will be allowed to reach you.
   2. *Stateful filters* store the information of state i.e., with which destination, you have initiated communication, what is the current state of the communication and so on.

This would help in the case where attacker tries to send the packets from his address which is not present in stored state information (since you didn’t initiate communication). The stateful filters would then block these packets, even though, the source and destination are not restricted by rules.

This way, the packets that come to user’s network are only from the sources which are allowed and with whom the user started to communicate.

1. Main differences between *Double Bastion T* and *Double Bastion Inline* are as follows:

|  |  |
| --- | --- |
| *Double Bastion T* | *Double Bastion Inline* |
| Internal network can be divided into different networks guarded by a separate internal firewall for each part of internal network | Internal network is guarded by one single internal firewall |
| Multiple internal firewalls give more flexibility to use different rules for each internal firewall as per requirement of the network it is guarding | Flexibility is comparatively less as there is one single ruleset present in the only internal firewall for internal network |
| Load is relatively less on each internal firewall as different parts of internal networks are guarded by its separate firewall | Load is more since there is one internal firewall guarding the whole internal network |
| Cost is relatively higher as different firewalls need to be configured for each part of internal network | Cost is relatively lower as only one internal firewall to be configured |
| Each of the internal firewalls need time for configuration and due to it, it would take longer to configure this type of network | With only one firewall to be configured, relatively less time is needed to configure |

In my opinion, *Double Bastion T* is more secure as:

* Each of the different components of network have different requirements for rules.

If rules are added within a single firewall, it’d have to be less strict, otherwise it’d block lot of traffic.

* Instead, if rules are added in each firewall based on network component the firewall is protecting, we can have more stricter rules for some components and less strict rules for other components of the network.
* Also, if changes are to be done for rules for one of the network component, it’d be easier to do that in *Double Bastion T* as each component has a different firewall. This way, it’d not affect the rest of the network.

However, this is not possible in *Double Bastion Inline* and single ruleset will have to be changed which can affect the other components of the network.

* Also, if one of the components get attacked/infected, separate firewalls will prevent the spread of it to other components of the networks.
* Also, debugging and maintaining firewalls of components of network is easier if we know which firewall has a problem.