Introduction to Computer Security

Question 1 ***Intrusion Detection***

FooCorp is deciding which intrusion detection *method* to employ in a few target scenarios. In the following parts, consider which of the intrusion detection methods learned in class would be most appropriate (NIDS, HIDS, or logging), and justify why.

Q1.1 FooCorp is hosting a web application over HTTPS and needs to detect any use of blacklisted characters in real time.

> **Solution:** FooCorp should use a HIDS. While the style of detection is ideal for a NIDS, it is useless in this specific scenario because TLS hinders a NIDS from reading the application data.

Q1.2 FooCorp is hosting a web application over HTTP and wants to pass all user traffic through an anomaly detection algorithm (which uses some computationally expensive mAcHinE LeARniNg). The web application needs to have low latency when many users are online during the day.

> **Solution:** FooCorp should use logging with overnight analysis. Using a NIDS or HIDS would cause the web application to have increased latency since running the analysis is expensive.

Q1.3 FooCorp uses the Simple Mail Transfer Protocol (SMTP) for email and wants to be able to quickly detect phishing attacks against any of their internal computers. SMTP runs on port 25 and is unencrypted.

> **Solution:** FooCorp should use a NIDS here. They only need one device to be able to monitor the whole corporation and it will be real-time detection.

Q1.4 FooCorp doesn't trust its employees and sets-up a NIDS to monitor their traffic. However, many employees use TLS, hindering what can be monitored.

FooCorp decides to turn their NIDS into a *Man-in-the-Middle*, giving it a certificate that all the employee's computers trust. Whenever an employee visits a website they complete a TLS handshake with the NIDS, the NIDS connects to the requested website using TLS, and any traffic between the employee and website is forwarded across the two TLS links by the NIDS.

Which security principle does this violate? Describe everything an attacker can do if they compromise the NIDS.

Solution: Separation of privilege. The NIDS has complete control over the contents, integrity, and authenticity of any connection.

The only certificate being verified by the client is the NIDS's so an attacker can reroute hosts to a malicious website and they wouldn't be able to tell. In other words, all the guarantees of TLS are lost.

The NIDS essentially has the same power as a MiTM with HTTP. Except it's even worse since there's a false sense of security due to the presence of TLS.

FooCorp now needs to decide which intrusion detection *technique* to employ in a few target scenarios. In the following parts, consider which technique would be most appropriate (signature-based, anomaly-based, specification-based, or behavioral), and justify why.

Q1.5 FooCorp wants to detect script kiddies (hackers who primarily use publicly available tools or exploits).

Solution: Signature-based detection would be best here since you can compile a list of all the things the hacker might do.

Q1.6 FooCorp wants to detect a seasoned l33t h4x0r who crafts custom exploits for each attack.

Solution: Behavioral/anomaly-based detection will be best here since the hacker will most likely use clever, possibly novel, techniques to pull of the attack. They can most likely avoid writing code that violates specifications or matches known signatures.

(Also 'l33t h4x0r' is slang for 'elite hacker' if anyone asks lol)

Q1.7 FooCorp wants to detect publicly-available malware that a hacker manually tweaks to avoid signature checks.

Solution: Behavioral detection will be best here since you have a good idea of how the attack works.

Q1.8 FooCorp wants to detect any attempts by their employees to access the protected `/etc/passwd` file.

Solution: Specification detection is best here. Simply set up rules that flag any syscalls which access `/etc/passwd`.

Question 2 ***Low-level Denial of Service***

In this question, you will help Mallory develop new ways to conduct denial-of-service (DoS) attacks.

CHARGEN and ECHO are services provided by some UNIX servers. For every UDP packet arriving at port 19, CHARGEN sends back a packet with 0 to 512 random characters. For every UDP packet arriving at port 7, ECHO sends back a packet with the same content.

Mallory wants to perform a DoS attack on two servers. One with IP address A supports CHARGEN, and another with IP address B supports ECHO. Mallory can spoof IP addresses.

Q2.1 Is it possible to create a single UDP packet with no content which will cause both servers to consume a large amount of bandwidth?

- If yes, mark 'Possible' and fill in the fields below to create this packet.
- If no, mark 'Impossible' and explain within the provided lines.

● Possible ○ Impossible

If possible, fill in the fields:

Source IP: ____**B**____ Destination IP: ____**A**____
Source port: ____**7**____ Destination port: ____**19**____

If impossible, why?

____

____

Solution: Source IP: B, port: 7. Destination IP: A, port: 19. Source and destination can be flipped. Notice this will create a chain of CHARGEN and ECHO that will generate a lot of network traffic.

Q2.2 Assume now that CHARGEN and ECHO are now modified to only respond to TCP packets (post-handshake) and not UDP. Is it possible to create a single TCP SYN packet with no content which will cause both servers to consume a large amount of bandwidth? Assume Mallory is off-path from the two servers.

- If yes, mark 'Possible' and fill in the fields below to create this packet.
- If no, mark 'Impossible' and explain within the provided lines.

○ Possible ● Impossible

If possible, fill in the fields:

Source IP: ______________ Destination IP: ______________
Source port: ______________ Destination port: ______________
Sequence #: ______________ Ack #: N/A

If impossible, why?

____

____

Solution: Impossible. As seen in previous question, source/destination IP has to be B/A for the chain to work. If you send a SYN packet to A pretending to be B, A will send SYN-ACK to B, which won't respond since it never sent a SYN. The connection won't be established.

Question 3 ***A Tour of Tor***

As a reminder, when connecting to a normal website through Tor, your computer first queries the Tor "consensus" to get a list of all Tor nodes, and using this information it connects to the first Tor node and, from there, creates a circuit through the Tor network, eventually ending at an exit node.

Q3.1 (4 min) Consider the scenario where you are in a censored country and the censor choses not to block Tor, the censor is the adversary, and no Tor relays exist within this country. How many Tor relays must your traffic pass through, including the exit node, to prevent the censor from blocking your traffic.

● One

○ Two

○ Three

○ Four

○ Tor doesn't stop this adversary

Solution: The censor doesn't block Tor and the relay is outside of the country, so one hop will get you safely past the censor. The censor will see you sending packets to an encrypted Tor relay but will not be able to determine who you're actually communicating with.

This is equivalent to using a VPN where the VPN server is in a different country.

Q3.2 (4 min) Consider the scenario where you are the only user of Tor on a network that keeps detailed logs of all IPs contacted. You use Tor to email a threat. The network operator is made aware of this threat and that it was sent through Tor and probably originated on the operator's network. How many Tor relays must your traffic pass through, including the exit node, to guarantee the network operator can't identify you as the one who sent the threat?

○ One

○ Two

○ Three

○ Four

● Tor doesn't stop this adversary

Solution: Since you are the only user of Tor, the network operator just needs to look at the IP of the only person trying to connect to a Tor relay. The network operator can look through the list of IPs and see that you contacted a Tor relay regardless of how many relays you use.

Q3.3 (4 min) Consider the scenario where there is a single hostile Tor node but you don't know that node's identitity, and that node can be an exit node. You want to keep confidential from this node what HTTP sites you are visiting through Tor. How many Tor relays must your traffic pass through, including the exit node, to guarantee this adversary can't know what sites you visit?

○ One

● Two

○ Three

○ Four

○ Tor doesn't stop this adversary

Solution: If you only use a single relay, then if that relay is hostile they will be able to see your request and the site you're visiting. If you use two relays, the first relay cannot see your request, and the second can see your request but doesn't know who it's from. So in either case, you are protected.

In other words, if the second relay is positioned between you and the hostile node, the hostile node will not know the request originated from you since it only sees the incoming request coming from "that other node." If the second relay is positioned between the hostile node and your destination, then while the hostile node knows the request comes from you, it doesn't know the destination since it forwards the request to "that other node."

Q3.4 (4 min) Consider the scenario where there are mulitple independent hostile Tor nodes but you don't know their identities, and these nodes can be exit nodes. You want to keep confidential from all these nodes what HTTP sites you are visiting through Tor. How many Tor relays must your traffic pass through, including the exit node, to guarantee that every independent hostile node can't know what sites you visit?

○ One

● Two

○ Three

○ Four

○ Tor doesn't stop this adversary

Solution: The solution is the same as the previous question. Since the hostile nodes are independent (non-colluding), it doesn't matter that there are multiple. No individual node can ever know both your identity and the request as long as you use at least two relays.

Q3.5 (4 min) Consider the scenario where there are multiple colluding hostile Tor nodes but you don't know those nodes identities, and these nodes can be exit nodes. You want to keep confidential from all these nodes what HTTP sites you are visiting through Tor. How many Tor relays must your traffic pass through, including the exit node, to guarantee that the colluding system of hostile nodes can't know what sites you visit?

○ One

○ Two

○ Three

○ Four

● Tor doesn't stop this adversary

Solution: Now, since the hostile nodes are colluding, you cannot ever be sure you are anonymous since you could get "unlucky" and have every node in your path be colluding hostile nodes.

Note that in real life, using three relays makes the probability of this happening negligible (assuming a certain amount of randomness in relay selection).

Q3.6 (4 min) Consider the scenario where there is a single hostile Tor node but you don't know that node's identity, and that node can be an exit node. You want to have data integrity for the HTTP sites you are visiting through Tor. How many Tor relays must your traffic pass through, including the exit node, to guarantee this adversary can't manipulate the data you receive from the sites you visit?

○ One

○ Two

○ Three

○ Four

● Tor doesn't stop this adversary

Solution: The exit node could modify the HTTP response without detection before forwarding the HTTP response to you.