Network Security III

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.

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

(b) 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.

(c) 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.

(d) 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.

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.

(e) FooCorp wants to detect script kiddies (hackers who primarily use publically available tools or exploits)

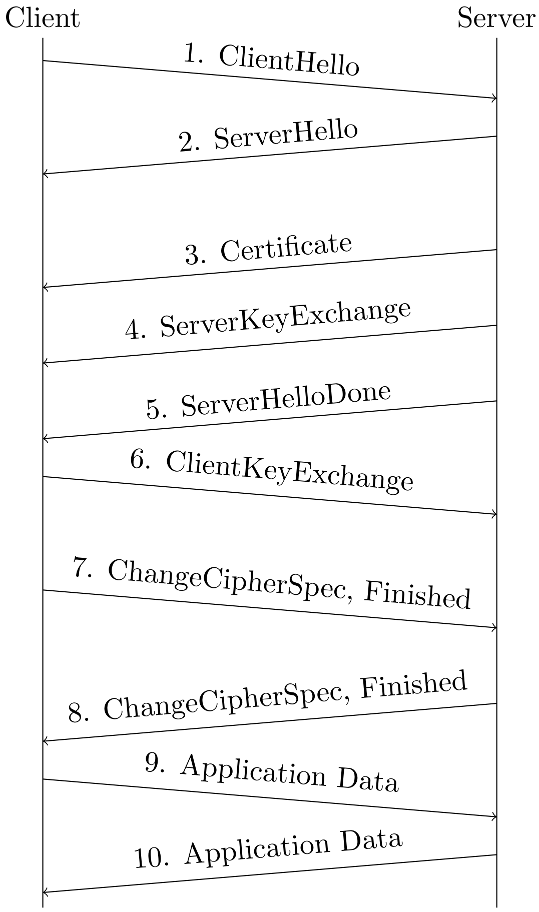
(f) FooCorp wants to detect a seasoned l33t h4x0r who uses crafts custom exploits for each attack

(g) FooCorp wants to detect publically-available malware that a hacker manually tweaks to avoid signature checks

(h) FooCorp wants to detect any attempts by their employees to access the protected `/etc/passwd` file

Question 2 *TLS protocol details*

Depicted below is a typical instance of a TLS handshake.



1. Client sends 256-bit random number R_b and supported ciphers

2. Server sends 256-bit random number R_s and chosen cipher

3. Server sends certificate

4. DH: Server sends $\{g, p, g^a \bmod p\}_{K_{\text{server}}^{-1}}$

5. Server signals end of handshake

6. DH: Client sends $g^b \bmod p$
   RSA: Client sends $\{PS\}_{K_{\text{server}}}$

Client and server derive cipher keys C_b, C_s and integrity keys I_b, I_s from R_b, R_s, PS

7. Client sends MAC(dialog, I_b)

8. Server sends MAC(dialog, I_s)

9. Client data takes the form $\{M_1, \text{MAC}(M_1, I_b)\}_{C_b}$

10. Server data takes the form $\{M_2, \text{MAC}(M_2, I_s)\}_{C_s}$

Figure 1: TLS 1.2 Key Exchange

(a) What is the purpose of the *client random* and *server random* fields?

(b) ClientHello and ServerHello are not encrypted or authenticated. Explain why a man-in-the-middle cannot exploit this. (Consider both the Diffie-Hellman and RSA case.)

(c) Note that in the TLS protocol presented above, there are two cipher keys C_b and C_s. One key is used only by the client, and the other is used only by the server. Likewise, there are two integrity keys I_b and I_s. Alice proposes that both the server and the client should simply share one cipher key C and one integrity key I. Why might this be a bad idea?

(d) The protocol given above is a simplified form of what actually happens. After step 8 (ChangeCipherSpec), the protocol as described above is still vulnerable. What is the vulnerability and how could you fix this?

Question 3 ***TLS threats***

An attacker is trying to attack the company Boogle and its users. Assume that users always visit Boogle's website with an HTTPS connection, using ephemeral Diffie-Hellman. You should also assume that Boogle does not use certificate pinning. The attacker may have one of three possible goals:

1. Impersonate the Boogle web server to a user
2. Discover some of the plaintext of data sent during a past connection between a user and Boogle's website
3. Replay data that a user previously sent to the Boogle server over a prior HTTPS connection

For each of the following scenarios, describe if and how the attacker can achieve each goal.

(a) The attacker obtains a copy of Boogle's certificate.

(b) The attacker obtains the private key of a certificate authority trusted by users of Boogle.

(c) The attacker obtains the private key corresponding to an old certificate used by Boogle's server during a past connection between a victim and Boogle's server. Assume that this old certificate has been revoked and is no longer valid. Note that the attacker does not have the private key corresponding to current certificate.