Peyrin & Ryan
Summer 2020

CS 161
Computer Security

Discussion 9

Web Security I

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 *Cross-site not scripting*

Consider a simple web messaging service. You receive messages from other users. The page shows all messages sent to you. Its HTML looks like this:

```
Mallory: Do you have time for a conference call?
Steam: Your account verification code is 86423
Mallory: Where are you? This is <b>important!!!</b>
Steam: Thank you for your purchase
       <img src="https://store.steampowered.com/assets/thankyou.png">
```

The user is off buying video games from Steam, while Mallory is trying to get ahold of them.

Users can include **arbitrary HTML code** messages and it will be concatenated into the page, **unsanitized**. Sounds crazy, doesn't it? However, they have a magical technique that prevents *any* JavaScript code from running. Period.

Discuss what an attacker could do to snoop on another user's messages. What specially crafted messages could Mallory have sent to steal this user's account verification code?