Day-10

Insufficent Logging & Monitoring

[Task 31] [Day 10] Insufficient Logging and Monitoring

When web applications are set up, every action performed by the user should be logged. Logging is important because in the event of an incident, the attackers actions can be traced. Once their actions are traced, their risk and impact can be determined. Without logging, there would be no way to tell what actions an attacker performed if they gain access to particular web applications.

The bigger impacts of these include:
• regulatory damage: if an attacker has gained access to personally identifiable user information and there is no record of this, not only are users of the application affected, but the application owners may be subject to fines or more severe actions depending on regulations.
• risk of further attacks: without logging, the presence of an attacker may be undetected. This could allow an attacker to launch further attacks against web application owners by stealing credentials, attacking infrastructure and more.

The information stored in logs should include:
• HTTP status codes
• Time Stamps
• Usernames
• API endpoints/page locations
• IP addresses

These logs do have some sensitive information on them so its important to ensure that logs are stored securely and multiple copies of these logs are stored at different locations.
As you may have noticed, logging is more important after a breach or incident has occurred. The ideal is case is having monitoring in place to detect any suspicious activity. The aim of detecting this suspicious activity is to either stop the attacker completely or reduce the impact they've made if their presence has been detected much later than anticipated.

Common examples of suspicious activity includes:
• multiple unauthorised attempts for a particular action (usually authentication attempts or access to unauthorised resources e.g. admin pages)
• requests from anomalous IP addresses or locations: while this can indicate that someone else is trying to access a particular user's account, it can also have a false positive rate.
• use of automated tools: particular automated tooling can be easily identifiable e.g. using the value of User-Agent headers or the speed of requests. This can indicate an attacker is using automated tooling.
• common payloads: in web applications, it's common for attackers to use Cross Site Scripting (XSS) payloads. Detecting the use of these payloads can indicate the presence of someone conducting unauthorised/malicious testing on applications.

Just detecting suspicious activity isn't helpful. This suspicious activity needs to be rated according to the impact level. For example, certain actions will higher impact than others. These higher impact actions need to be responded to sooner thus they should raise an alarm which raises the attention of the relevant party.
Put this knowledge to practise by analysing this sample log file.

#1
What IP address is the attacker using?

49.99.13.16

#2
What kind of attack is being carried out?

brute force

login-logs.txt

200 OK	**12.55.22.88**	**jr22**	**2019-03-18T09:21:17 /login**
200 OK	**14.56.23.11**	**rand99**	**2019-03-18T10:19:22 /login**
200 OK	**17.33.10.38**	**afer11**	**2019-03-18T11:11:44 /login**
200 OK	**99.12.44.20**	**rad4**	**2019-03-18T11:55:51 /login**
200 OK	**67.34.22.10**	**bff1**	**2019-03-18T13:08:59 /login**
200 OK	**34.55.11.14**	**hax0r**	**2019-03-21T16:08:15 /login**
401 Unauthorised	**49.99.13.16**	**admin**	**2019-03-21T21:08:15 /login**
401 Unauthorised	**49.99.13.16**	**administrator**	**2019-03-21T21:08:20 /login**
401 Unauthorised	**49.99.13.16**	**anonymous**	**2019-03-21T21:08:25 /login**
401 Unauthorised	**49.99.13.16**	**root**	**2019-03-21T21:08:30 /login**