Security incident report

|  |
| --- |
| **Section 1: Identify the network protocol involved in the incident** |
| The HTTP protocol was the one affected by the incident. By running tcpdump and accessing the yummyrecipesforme.com website, we were able to identify the issue and gather evidence in the log file. The captured protocol and traffic activity were logged in a DNS & HTTP traffic log file, which helped us come to this conclusion. Our investigation revealed that the malicious file was being transmitted to users' computers through the HTTP protocol at the application layer. |
|

|  |
| --- |
| **Section 2: Document the incident** |
| 1. During the sandbox testing phase, the log shows the source computer using port 5244 to send a DNS resolution request to the DNS server for the correct destination URL. The reply comes back from the DNS server with the correct destination URL.  2. The website then initiates an HTTP:GET request requesting data from yummyrecipesforme.com, which could be the download request for the malicious file.  3. Traffic is then routed from the source computer to the DNS server again using port 5244, but this time the DNS server routes the traffic to a new IP address(192.0.2.172) and its associated URL (greatrecipesforme.com.http). The traffic changes to a route between the source computer and the spoofed website. The port number(.56378) on the source computer has changed again when redirected to a new website. |

|  |
| --- |
| **Section 3: Recommend one remediation for brute force attacks** |
| One recommended remediation would be to limit the number of login attempts. This would be effective against brute force attacks because:  1. By limiting the number of login attempts, an attacker's ability to try numerous username and apssword combinations would be significantly reduced. This introduces a time delay between each attempt, slowing down the attack process.  2. Brute force attacks rely on systematically attempting all possible combinations until a valid username and password pair is found. By imposing limits on login attempts, it becomes increasingly difficult for attacked to succeed within a reasonable timeframe.  3. This would deter automated attacks, suck as a dictionary attack. The attackers tools are rendered less effective because the attacker's script would quickly exhaust its limit and fail to gain unauthorized access.  4. Limiting login attempts could be combined with monitoring mechanisms that track failed login attempts and raise alets when suspicious activity is detected. This would provide an opportuntiy for a security person to identify and respond to ongoing attacks, such as blocking the attacking IP address. |