Security incident report

Section 1: Identify the network protocol involved in the incident
The network protocol involved in this incident is HTTP (Hyper Text Markup Language). Since the issue was accessing the webserver, we know that the request for a webpage from the server involves http protocol. Furthermore, on accessing the tcpdump logs, we can see that the http protocol is used to request data from the yummyrecipesforme.com website. This could be the download request for the malicious file. The malicious file is being transported to the user's device using the http protocol at the application layer.

Section 2: Document the incident
Multiple customers contacted the company website's helpdesk stating that the website had prompted them to download a file to access free recipes. They claimed that after running the file, the address of the website changed and their computers starting running slow ever since. The website owner then tried to log in to the admin panel but was locked out of the account. The cybersecurity analysts used a sanboxed environment to observe the website's suspicious behaviour. Then, the analysts ran tcpdump to capture the network traffic packets produced by interacting with the website. The analyst, on accessing the website, was prompted to download and run a file claiming to provide free recipes. On accepting the download and allowing the file to run browser redirected the analyst to a fake website (greatrecipesforme.com). The analyst inspected the tcpdump log, which shows that the browser first creates a DNS request for the IP address of yummyrecipesforme.com. The server replies with the correct IP address, using which the browser initiates an HTTP request for the yummyrecipesforme webpage. Then the browser initiates the download of the malware. The logs showed a sudden change in network traffic as the browser initiated a DNS request for a website greatrecipesforme.com. The network traffic was then rerouted to the new IP address for the greatrecipesforme.com website.

A senior security analyst checked the source code for the website and confirmed that it had been compromised. The analyst discovered that an attacker had manipulated the source code by adding a javascript code to prompt website users to the download an executable file. Analysis of the file found a script that redirects the visitors' browsers from yummyrecipesforme.com to greatrecipesforme.com. Since the website owner had been locked out of their administrator account, the analysts believe that the server was impacted by a brute force attack. The attacker was able to guess the password so easily because it was set to one of the default passwords. Additionally no controls were set in place to prevent against a brute force attack.

Section 3: Recommend one remediation for brute force attacks
One security measure the team plans to implement is disallowing the use of previous passwords. This measure addresses the specific scenario in which the attacker was able to guess the administrator password easily because it had been set to a default value. Another proposed measure is the adoption of multi-factor authentication (MFA), such as one-time passwords (OTPs), biometrics, or similar methods. With MFA, users must confirm their identity using both their login credentials and an additional verification method (e.g., an OTP or biometric authentication) before gaining access to the system. Additionally, the team recommends implementing controls to monitor login attempts. In cases where SIEM (Security Information and Event Management) tools detect unusual activity—such as login attempts at odd hours or multiple incorrect password entries—these incidents must be thoroughly investigated. To prevent potential malicious activity during the investigation, access should be temporarily restricted until the issue is resolved. These measures are aimed at mitigating brute force attacks and enhancing the overall security posture of the system.