Security incident report

Section 1: Identify the network protocol involved in the incident
The protocol implicated in the event is HTTP (Hypertext Transfer Protocol). Given the difficulty accessing the web server for yummyrecipesforme.com, it's evident that web page requests entail HTTP traffic. Additionally, our tcpdump analysis of accessing the yummyrecipesforme.com site revealed HTTP protocol usage in the corresponding log file. The malicious file is observed being transmitted to users' computers via the HTTP protocol at the application layer.

Section 2: Document the incident
Numerous clients contacted the website's support desk, reporting that upon visiting the site, they were prompted to download and execute a file supposedly containing new recipes. Subsequently, their personal computers experienced significant slowdowns. Upon attempting to log into the web server, the website owner discovered they were locked out of their account. To investigate, the cybersecurity analyst utilized a sandbox environment to access the website safely without impacting the company's network. Employing tcpdump, the analyst captured network traffic packets generated during interaction with the site. Accepting a download prompt for what purported to be free recipes, the analyst executed the file, leading the browser to redirect to a fraudulent website (greatrecipesforme.com). Upon scrutinizing the tcpdump log, the analyst noted the browser's initial request for the IP address of yummyrecipesforme.com. Subsequent establishment of a connection with the website via the HTTP protocol coincided with the file download and execution. Notably, the log exhibited a sudden shift in network traffic as the browser sought a new IP address for greatrecipesforme.com, ultimately redirecting to it. The senior cybersecurity expert delved into the source code of both websites

and the downloaded file. Their analysis unveiled that an attacker had manipulated the website to induce users into downloading a malicious file disguised as a browser update. Given the website owner's account lockout, it's inferred that the attacker likely employed a brute force attack to compromise the account and alter the admin password, consequently facilitating the execution of the malicious file and compromising end users' computers.

Section 3: Recommend one remediation for brute force attacks
One security measure the team plans to implement to guard against brute force attacks is to prohibit the reuse of previous passwords. Given that the vulnerability exploited in this attack stemmed from the attacker's ability to utilize a default password for login, it's crucial to prevent the use of any old passwords, including default ones, in password resets. Another supportive measure is to enforce more frequent password updates. By ensuring passwords are updated regularly, unauthorized individuals who gain knowledge of a password are less likely to successfully use it if it's changed promptly. Furthermore, implementing two-factor authentication (2FA) is another beneficial solution. 2FA mandates authentication through both a password and a one-time passcode (OTP) sent to the user's email or phone. Once the user verifies their identity using their login credentials and the OTP, they gain access to the system. This additional layer of authentication significantly reduces the likelihood of success for malicious actors attempting brute force attacks, as they would require additional authentication beyond just the password.