Security incident report

Section 1: Identify the network protocol involved in the incident

The network protocol involved in the incident is HTTP protocol version 1.1

Section 2: Document the incident

A baker executed a brute force attack by repeatedly entering known default passwords for the administrative account until the ultimately obtained the correct login credentials. The baker then accessed the admin panel to change the website's source code by embedding a javascript function that prompted visitors to the site to download and run a file. Once the file was run, customers were redirected to a fake version of the website which made the seller's recipes available for free. The website owner became aware of this incident several hours after the attack when multiple customers emailed yummyrecipesforme's helpdesk with complaints that the website prompted them to run a file to update their browsers and that running this file changed the website address and slowed their computers' functions. The owner notified us to investigate the incident after being unable to login to the admin panel in response to the customers' complaints.

I used a sandbox environment to run tcpdump on the website's URL. When the website loaded, I was prompted to download and run a file, which I did. The browser then redirected me to a different URL, greatrecipesforme.com, which appeared similarly designed to yummyrecipesforme.com but had the recipes sold by the company freely available.

The tcpdump logs showed that the browser requested a DNS resolution of the website's URL, and that the DNS replied with the correct IP address. However, the browser then initiated an HTTP request for the webpage and initiated a download of malware before requesting another DNS resolution, this time for the fake site greatrecipesforme.com. The DNS server responded with a new IP address, and the browser initiated an HTTP request to the new IP address.

A senior analyst confirmed that javascript code was added to the website to prompt visitors to download an executable file, which contained a script that redirected visitors' browsers to the spoofed site. The cybersecurity team found that a brute force attack was used to guess the admin password, which was still set to the default password. No controls were in place at the time to prevent brute force attacks.

Section 3: Recommend one remediation for brute force attacks

I recommend implementing a password policy that requires strong passwords and limits the number of incorrect password attempts that can be made before access is suspended. Requiring strong passwords would make it more difficult for the disgruntled baker to guess the admin credentials, and limiting the number of incorrect attempts that can occur before access is suspended would prevent the baker from continuing to test different default passwords indefinitely until stumbling across the correct credentials.