# Security incident report

| **Section 1: Identify the network protocol involved in the incident** |
| --- |
| The network protocol involved in the incident is the HTTP protocol on the TCP/IP Application layer.  Where on the 5th time stamp (14:18:36.786589) on the DNS/HTTP traffic log the log entry with the code HTTP: GET / HTTP/1.1 shows the browser is requesting data from yummyrecipesforme.com with the HTTP: GET method using HTTP protocol version 1.1. This could be the download request for the malicious JavaScript file. |
|

| **Section 2: Document the incident** |
| --- |
| Multiple customers emailed yummyrecipesforme’s helpdesk complaining that the company’s website had prompted them to download a file to update their browsers. The customers claimed that, after running the file, the address of the website changed and their personal computers began running more slowly.  In response to this incident, the website owner tries to log in to the admin panel but is unable to, so they reach out to the website hosting provider.  I create a sandbox environment to observe the suspicious website behavior. Running the network protocol analyzer tcpdump, then type in the URL for the website, yummyrecipesforme.com. As soon as the website loads, it prompted to download an executable file to update your browser. You accept the download and allow the file to run. Then my browser redirects to a different URL, greatrecipesforme.com, which is designed to look like the original site. However, the recipes are now posted for free on the new website.  The logs show the following process:  ✓The browser requests a DNS resolution of the yummyrecipesforme.com URL.  ✓The DNS replies with the correct IP address.  ✓The browser initiates an HTTP request for the webpage.  ✓The browser initiates the download of the malware.  ✓The browser requests another DNS resolution for greatrecipesforme.com.  ✓The DNS server responds with the new IP address.  ✓The browser initiates an HTTP request to the new IP address.  A senior analyst confirms that the website was compromised. The analyst checks the source code for the website. They notice that javascript code had been added to prompt website visitors to download an executable file. Analysis of the downloaded file found a script that redirects the visitors’ browsers from yummyrecipesforme.com to greatrecipesforme.com.  The cybersecurity team reports that the web server was impacted by a brute force attack. The disgruntled baker was able to guess the password easily because the admin password was still set to the default password. Additionally, there were no controls in place to prevent a brute force attack. |

| **Section 3: Recommend one remediation for brute force attacks** |
| --- |
| For the brute force attack remediation, there are numerous counter measures to implement but I will recommend Enforcing two-factor authentication (2FA), this put in place a two layer authentication for access control, it implement strong password set up, and there's a log strategy of OTP for notification of access and random password creation.  Since the vulnerability that lead to this attack was the attacker’s ability to use a default password to log in, it’s important that we prevent any old passwords such as default passwords from being used to reset the password. Another supportive measure is to require more frequent password updates, so in case any unauthorized person becomes aware of the password, they are less likely to be able to use that password if the password is updated sooner than later. |