# Security incident report

|  |
| --- |
| **Section 1: Identify the network protocol involved in the incident** |
| The network protocols involved in this incident include:   1. **Domain Name System (DNS)** – Used to resolve the domain names (yummyrecipesforme.com and greatrecipesforme.com) to their respective IP addresses. 2. **Hypertext Transfer Protocol (HTTP)** – Used to request and load webpages from the web server, including the malicious redirection. 3. **Transmission Control Protocol (TCP)** – Handles the communication between the client and server for both HTTP and DNS requests.   The logs indicate that:   * A DNS query was made to resolve yummyrecipesforme.com, and the response returned an IP address. * The browser then initiated an **HTTP GET request** over **port 80**, leading to the download prompt. * After execution of the malware, another **DNS query** was made to resolve greatrecipesforme.com, followed by another **HTTP GET request** to establish a connection with the malicious site. |
|

|  |
| --- |
| **Section 2: Document the incident** |
| The security incident involved an unauthorized modification of the website’s source code due to a **brute force attack** on the admin panel. The hacker, a former employee, successfully guessed the **default admin password** and injected **malicious JavaScript code** into the website.  The JavaScript modification forced visitors to download and execute a file under the pretense of updating their browser. Upon execution, the malware redirected users to greatrecipesforme.com, a spoofed version of the original site, which likely contained further malicious payloads. **Incident Timeline (Based on tcpdump logs):**  1. The **browser initiated a DNS request** to resolve yummyrecipesforme.com, and the DNS server returned an IP address. 2. An **HTTP GET request** was made to yummyrecipesforme.com, where the website prompted users to download an executable file. 3. Once executed, the file initiated a **DNS query for greatrecipesforme.com**, followed by an **HTTP GET request** to its server. 4. The infected machine continued communication over **port 80 (HTTP)** with the malicious server.  **How It Was Discovered:**  * Customers reported unusual behavior, including forced downloads and redirections. * The website owner was **locked out** of the admin panel, suggesting unauthorized access. * A cybersecurity investigation confirmed **unauthorized code injection** in the website’s source code. |

|  |
| --- |
| **Section 3: Recommend one remediation for brute force attacks** |
| To prevent similar brute force attacks in the future, it is recommended that **Two-Factor Authentication (2FA)** be implemented for all admin accounts. **Why is 2FA Effective?**  * Even if an attacker guesses or cracks the password, they would need a second factor (such as a mobile authentication app or a security token) to gain access. * 2FA significantly reduces the likelihood of unauthorized access, even if weak passwords are used. * It enhances overall security without relying solely on strong passwords.   Additionally, **account lockout policies** should be enforced to prevent excessive login attempts, and **default passwords should be changed immediately upon setup**. |