# Security incident report

| **Section 1: Identify the network protocol involved in the incident** | |
| --- | --- |
| The primary network protocol identified in the tcpdump log is **HTTP (Hypertext Transfer Protocol)**.  HTTP traffic is visible on **port 80**, which operates at the **Application Layer** of the TCP/IP model.  Additionally, the **DNS protocol** (over UDP and sometimes TCP) is also used for name resolution, shown via requests to dns.google.domain.  These protocols are built on top of the **TCP protocol**, which operates at the **Transport Layer**. | |
|

| **Section 2: Document the incident** |
| --- |
| The incident involved unusual activity and increased HTTP traffic between the user machine and multiple web servers (e.g., yummyrecipesforme.com and greatrecipesforme.com) as captured in the tcpdump log.  **Detailed Timeline and Behavior:**   * At 14:18:32, a DNS A-record query for yummyrecipesforme.com was made to dns.google.domain, and a valid IP response was returned. * Shortly after, a complete **TCP three-way handshake** was initiated successfully with the destination server using port 80 (HTTP). * An HTTP GET request was made, followed by consistent traffic flow on the HTTP protocol. * Later, at 14:20:32, another DNS query was made for greatrecipesforme.com, with similar successful connection patterns. * At 14:25:29, another TCP handshake and HTTP GET request occurred to greatrecipesforme.com, also resulting in a stable HTTP session.   **Observations:**   * No anomalies such as dropped SYNs or RST responses are shown, but the volume and repetition suggest the possibility of an early stage reconnaissance or brute-force browsing/scripted scanning.   **Source:**  All information is sourced from the packet data in tcpdump traffic log.docx, which logs packet-level network communications. |

| **Section 3: Recommend one remediation for brute force attacks** |
| --- |
| **Enable account lockout policies after a limited number of failed login attempts.**  **Explanation:**  This approach is effective because it prevents attackers from continuously guessing usernames or passwords. Once a user account reaches a defined number of failed attempts (e.g., 5), the account is temporarily locked or requires additional verification. This drastically slows down brute force methods and helps detect unauthorized access attempts in real time.  Additional enhancements like CAPTCHA and two-factor authentication (2FA) can further strengthen the protection. |