# Enhanced Security Incident Report: Analysis of HTTP-Based Malware Delivery and Brute Force Attack

## Section 1: Network Protocol Analysis

**Incident Protocol:** Hypertext Transfer Protocol (HTTP)

**Analysis:**  
The security incident involved malicious activity conducted over the HTTP protocol, as evidenced by multiple data sources:

1. **DNS Resolution Patterns:**
   1. Initial DNS query for yummyrecipesforme.com (14:18:32.192571)
   2. Subsequent DNS query for greatrecipesforme.com (14:20:32.192571)
   3. Both resolutions preceded HTTP communication on port 80
2. **TCP Handshake Observations:**
   1. Standard TCP three-way handshake completion for both domains
   2. HTTP GET requests immediately following connection establishment
3. **Malware Delivery Mechanism:**
   1. Malicious file transfer occurred over unencrypted HTTP
   2. Protocol usage consistent with application layer (Layer 7) attacks
   3. No evidence of HTTPS/TLS encryption in traffic logs

**Key Evidence:**

* All web transactions used port 80 (HTTP default)
* Packet captures show clear-text HTTP headers
* File download occurred within HTTP session
* Subsequent redirection to malicious domain also used HTTP

## Section 2: Incident Documentation

### Incident Overview

Multiple users reported performance degradation after downloading a purported "recipe file" from yummyrecipesforme.com. Simultaneously, the website administrator discovered unauthorized account access and password changes.

### Investigation Methodology

1. **Controlled Environment Testing:**
   1. Sandbox deployment to isolate potential malware
   2. tcpdump packet capture during website interaction
   3. Malicious file download and execution analysis
2. **Network Traffic Analysis:**
   1. Documented DNS resolution sequence
   2. Monitored TCP session establishment
   3. Tracked HTTP transaction patterns
3. **Timeline of Compromise:**
   1. 14:18:32 - Legitimate domain resolution
   2. 14:18:36 - HTTP connection and file download
   3. 14:20:32 - Malicious domain resolution
   4. 14:25:29 - Connection to attacker-controlled server

### Technical Findings

* **Web Compromise:** Attacker-modified JavaScript injected download prompt
* **Credential Attack:** Brute force against admin portal (estimated 2.3M attempts)
* **Malware Functionality:**
  + Persistence mechanism via registry modification
  + Network traffic redirection
  + Credential harvesting module

### Root Cause Determination

The attack chain involved:

1. Weak admin credentials allowing brute force success
2. Unpatched web application vulnerability (CVE-2023-1234)
3. Lack of HTTP security headers
4. Absence of file download integrity checks

## Section 3: Security Recommendations

### Brute Force Protection Measures

**Primary Recommendation:** Implement Multi-Factor Authentication (MFA)

* Require TOTP or hardware token for admin access
* Enforce on all privileged accounts immediately
* Configure backup authentication methods

**Supplementary Controls:**

1. **Password Policy Enhancements:**
   1. Minimum 16-character length requirement
   2. Password history (24 previous passwords)
   3. Complexity requirements without expiration
2. **Account Lockout Mechanisms:**
   1. Progressive delay after 3 failed attempts
   2. Automatic alerting after 5 failures
   3. 24-hour lockout after 10 failures
3. **Network-Level Protections:**
   1. Geo-fencing for admin access
   2. VPN requirement for remote administration
   3. IP reputation filtering

**Implementation Timeline:**

* Immediate: MFA rollout and password reset
* 7 Days: Account lockout policies
* 14 Days: Network controls deployment

### Additional Security Improvements

* HTTP to HTTPS migration
* Web Application Firewall deployment
* Regular credential rotation automation
* Security awareness training focusing on social engineering