Short Project: Comprehensive Security Assessment and Hardening of a

Small Business Network

Tools used:

* Nmap
* Wireshark
* OpenVas (Greenbone)
* Nessus
* Network: 203.124.43.0/24
* Domain’s info:-

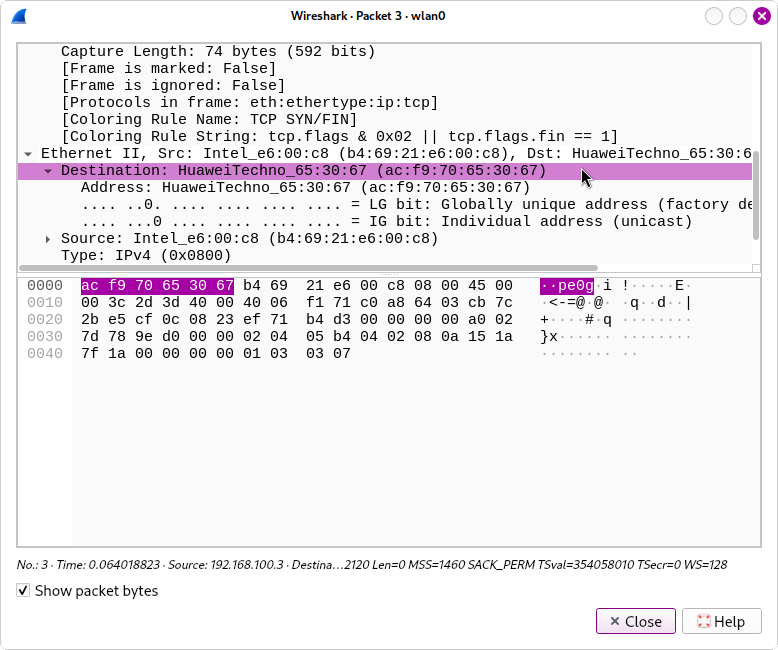


Server’s ip: 203.124.43.229

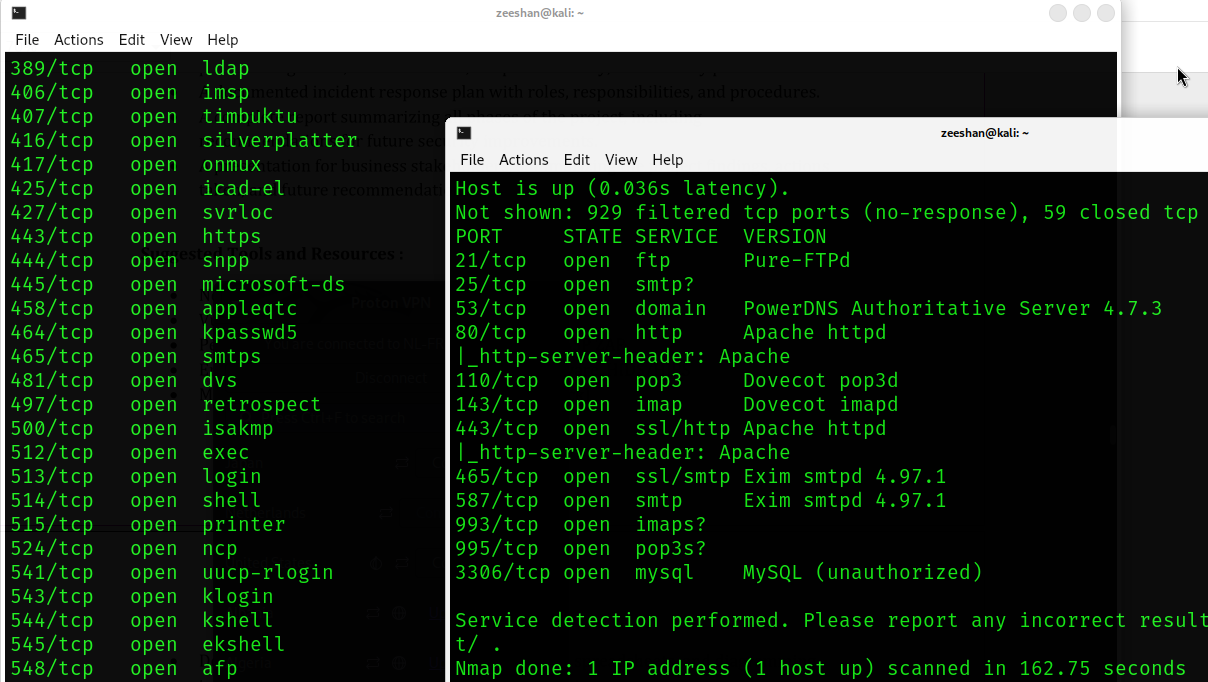
Mac Address: ac:f9:70:65:30:67

Technology: HuaweiTechno

Domain Hosting: host201003.comsatshosting.com



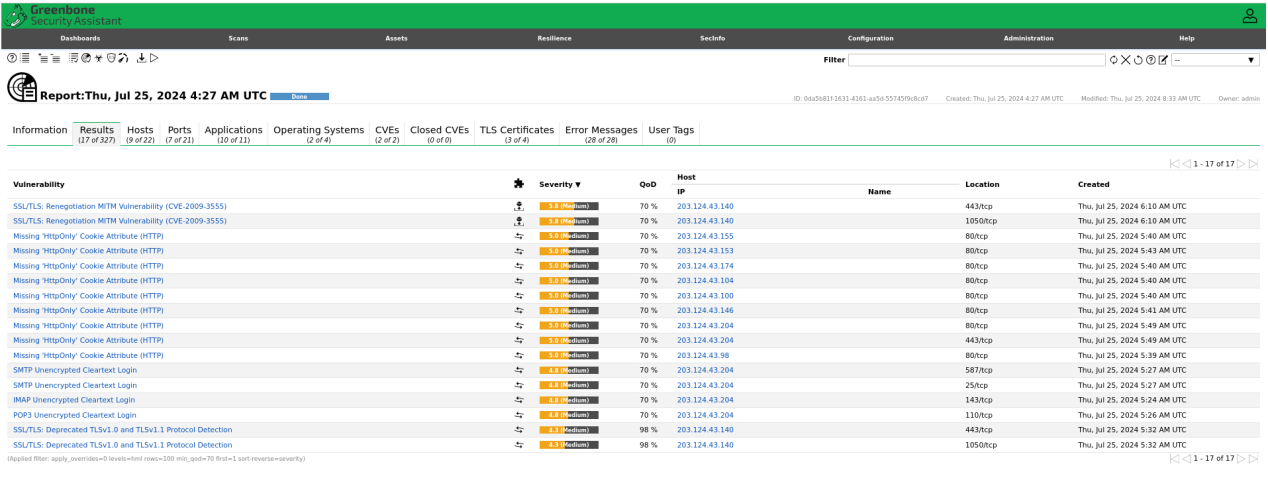
* More than 1000+ ports are opened.



|  |  |
| --- | --- |
| **Ip Address** | **Service** |
| 203.124.43.1 | HP P2000 G3 NAS device |
| 203.124.43.2 | OpenBSD 4.X |
| 203.124.43.21 | Netgear WGR614v7 wireless broadband router |
| 203.124.43.65 | Asus WL-500gP wireless broadband router |
| 203.124.43.66 | Cisco C7200 router (IOS 15) |
| 203.124.43.73 | AVtech Room Alert 26W environmental monitor |
| 203.124.43.80 | D-Link DFL-700 firewall |
| 203.124.43.81 | Cisco Aironet 1250 WAP (IOS 12.4) |
| 203.124.43.87 | D-Link DFL-700 firewall |
| 203.124.43.89 | Cisco Aironet 1250 WAP (IOS 12.4) |
| 203.124.43.96 | D-Link DFL-700 firewall |
| 203.124.43.97 | Cisco Aironet 1250 WAP (IOS 12.4) |
| 203.124.43.98 | Ubiquiti AirMax NanoStation WAP (Linux 2.6.32) |
| 203.124.43.100 | Infomir MAG-250 set-top box |
| 203.124.43.102 | Ubiquiti Pico Station WAP (AirOS 5.2.6) |
| 203.124.43.104 | Ubiquiti Pico Station WAP (AirOS 5.2.6) |
| 203.124.43.107 | Infomir MAG-250 set-top box |
| 203.124.43.108 | Infomir MAG-250 set-top box |
| 203.124.43.111 | D-Link DFL-700 firewall |
| 203.124.43.112 | D-Link DFL-700 firewall |
| 203.124.43.113 | Cisco C7200 router |
| 203.124.43.119 | D-Link DFL-700 firewall |
| 203.124.43.121 | Cisco C7200 router |
| 203.124.43.132 | linux:linux\_kernel |
| 203.124.43.136 | linux:linux\_kernel |
| 203.124.43.137 | linux:linux\_kernel |
| 203.124.43.140 | Huawei embedded |
| 203.124.43.141 | Cyberoam UTM firewall |
| 203.124.43.143 | Adaptec Snap Server 100 NAS device |
| 203.124.43.144 | D-Link DFL-700 firewall |
| 203.124.43.146 | AirMax NanoStation WAP (Linux 2.6.32) |
| 203.124.43.150 | Dropbear sshd (protocol 2.0) |
| 203.124.43.202 | WGR614v7 wireless broadband router |
| 203.124.43.203 | Microsoft Windows |
| 203.124.43.204 | Linux 5.0 - 5.4 |
| 203.124.43.206 | Mail Server Windows |
| 203.124.43.210 | PowerDNS Authoritative Server |
| 203.124.43.225 | AVtech embedded |
| 203.124.43.226 | Server |
| 203.124.43.227 | Server |
| 203.124.43.228 | Server |

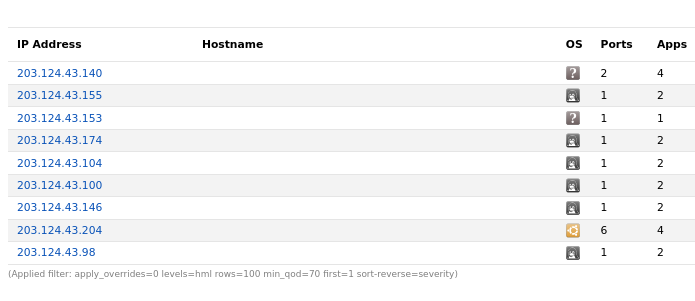
These are the devices connected with the network.

* **Vulnerabilities:-**

****

|  |  |  |
| --- | --- | --- |
| **Vulnerability** | **Severity** | **Ip Address** |
| SSL/TLS: Renegotiation MITM Vulnerability (CVE-2009-3555) | 5.8 | [203.124.43.140](https://127.0.0.1:9392/host/00d3b8d8-92bb-473e-b1e5-820bb46847ae) |
| Missing 'HttpOnly' Cookie Attribute (HTTP) | 5.0 | |  |  | | --- | --- | |  | [203.124.43.155](https://127.0.0.1:9392/host/708c9dbf-2100-4c27-b07f-3a805a67d5ae) | |
| Missing 'HttpOnly' Cookie Attribute (HTTP) | 5.0 | [203.124.43.153](https://127.0.0.1:9392/host/da9204eb-fa37-4ba4-aeb0-a8e4be3e9025) |
| Missing 'HttpOnly' Cookie Attribute (HTTP) | 5.0 | [203.124.43.174](https://127.0.0.1:9392/host/79585a8a-43b6-41d0-9fcb-686ddb96c229) |
| Missing 'HttpOnly' Cookie Attribute (HTTP)  Missing 'HttpOnly' Cookie Attribute (HTTP) | 5.0 | [203.124.43.104](https://127.0.0.1:9392/host/e5c36b6d-906a-4bd2-acda-f574e211f04c) |
| Missing 'HttpOnly' Cookie Attribute (HTTP) | 5.0 | [203.124.43.100](https://127.0.0.1:9392/host/7a34bf03-f94d-4ac6-99f9-ee4728bf7aec) |
| Missing 'HttpOnly' Cookie Attribute (HTTP) | 5.0 | [203.124.43.146](https://127.0.0.1:9392/host/6837e448-f254-4c12-8f77-935302f6cfec) |
| Missing 'HttpOnly' Cookie Attribute | 5.0 | |  |  | | --- | --- | |  | [203.124.43.204](https://127.0.0.1:9392/host/881be3ff-dd19-4efd-a358-14ee57003f36) | |
| Missing 'HttpOnly' Cookie Attribute (HTTP) | 5.0 | [203.124.43.204](https://127.0.0.1:9392/host/881be3ff-dd19-4efd-a358-14ee57003f36) |
| SMTP Unencrypted Cleartext Login | 4.8 | [203.124.43.204](https://127.0.0.1:9392/host/881be3ff-dd19-4efd-a358-14ee57003f36) |
| IMAP Unencrypted Cleartext Login | 4.8 | [203.124.43.204](https://127.0.0.1:9392/host/881be3ff-dd19-4efd-a358-14ee57003f36) |
| POP3 Unencrypted Cleartext Login | 4.8 | [203.124.43.204](https://127.0.0.1:9392/host/881be3ff-dd19-4efd-a358-14ee57003f36) |
| SSL/TLS: Deprecated TLSv1.0 and TLSv1.1 Protocol Detection | 4.3 | |  |  | | --- | --- | |  | [203.124.43.140](https://127.0.0.1:9392/host/00d3b8d8-92bb-473e-b1e5-820bb46847ae) | |

* **Hosts:-**

****

* **TLS certificates:-**

****

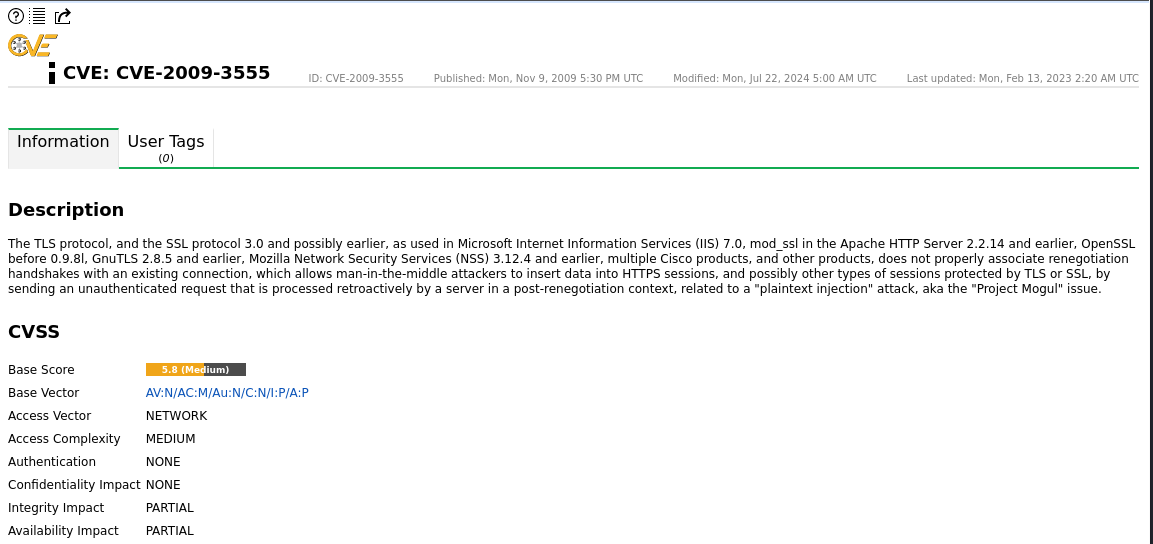
## CVE: CVE-2009-3555

The vulnerability lies in the way the TLS protocol handles renegotiation requests. Renegotiation is a feature that allows a client and server to renegotiate the parameters of an existing TLS connection. However, in the affected versions of the protocol, there was no cryptographic binding between the renegotiation handshake and the original handshake, making it possible for an attacker to inject arbitrary content into the renegotiation process.

### Impact

An attacker could exploit this vulnerability to perform a Man-in-the-Middle (MitM) attack. By intercepting the communication between a client and server, the attacker could inject malicious data into the TLS session, potentially leading to data breaches, session hijacking, or other forms of unauthorized access.

<https://cve.mitre.org/cgi-bin/cvename.cgi?name=CAN-2009-3555>



### Detailed Solution for CVE-2009-3555

CVE-2009-3555, the TLS Renegotiation Vulnerability, requires both immediate and long-term solutions to ensure the security of communications using the TLS protocol. Below is a detailed approach to addressing this vulnerability:

#### 1. Understanding the Vulnerability

Before diving into solutions, it’s important to understand the root cause:

* The vulnerability exists because TLS renegotiation requests are not cryptographically tied to the original handshake.
* This allows an attacker to inject data into the session during the renegotiation process, potentially leading to Man-in-the-Middle (MitM) attacks.

#### 2. Immediate Mitigation

While working on a long-term fix, immediate steps should be taken to mitigate the risk:

* **Disable TLS Renegotiation**: Temporarily disable renegotiation on servers. This can usually be configured in the server’s TLS/SSL settings.
  + For example, in Apache HTTP Server, add the following to the configuration file:

SSLInsecureRenegotiation off

#### 3. Long-term Solution

The ultimate fix involves updating the TLS implementation to support the Renegotiation Indication Extension as defined in RFC 5746. Here’s a step-by-step guide:

**Step 1: Update SSL/TLS Libraries**

**OpenSSL**: Ensure you are using OpenSSL version 0.9.8m or later, which includes the renegotiation fix.

sudo apt-get update

sudo apt-get install openssl

**GnuTLS**: Ensure you are using GnuTLS version 2.8.6 or later.

sudo apt-get update

sudo apt-get install gnutls-bin

**Step 3: Verify Renegotiation Fix**

* Use tools to verify that your server properly supports the renegotiation extension.
  + **Qualys SSL Labs**: Run an SSL Test on your server. Look for the section on renegotiation. It should indicate secure renegotiation support.
  + **OpenSSL Command**: Use the openssl s\_client command to check renegotiation:

openssl s\_client -connect yourserver.com:443 -tls1

**Step 4: Client-Side Updates**

* Ensure that all client software and libraries that connect to your server also support the renegotiation extension.
  + Browsers and other client applications typically update automatically, but verify that your internal applications are using updated libraries.

**Step 5: Testing and Monitoring**

* After implementing the fix, thoroughly test your server to ensure that all services are functioning correctly and securely.
* Continuously monitor for any unusual activity that might indicate attempted exploitation of renegotiation vulnerabilities.

#### 4. Documentation and Training

* Document the changes made and ensure that all IT staff are aware of the implications of the vulnerability and the applied fixes.
* Train relevant personnel on the importance of keeping SSL/TLS libraries and server software up to date.

**CVE: CVE-2015-0204**

CVE-2015-0204, also known as the "FREAK" (Factoring Attack on RSA-EXPORT Keys) vulnerability, affects several SSL/TLS implementations. Here are the key details:

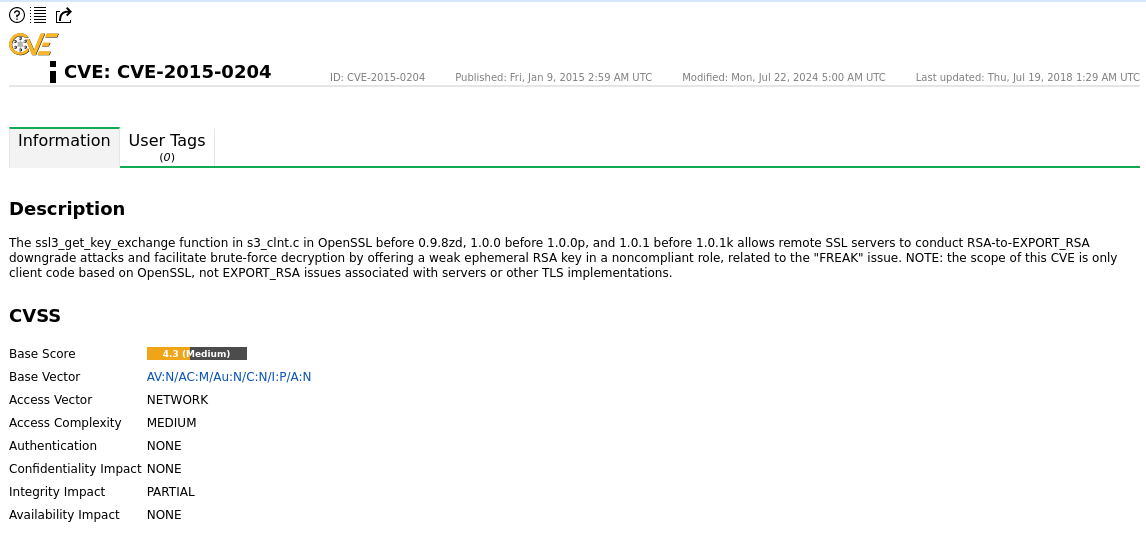
### Overview

* **Vulnerability Name:** FREAK (Factoring Attack on RSA-EXPORT Keys)
* **CVE ID:** CVE-2015-0204
* **Published:** January 8, 2015
* **Description:** This vulnerability allows a man-in-the-middle (MITM) attacker to downgrade the security of an SSL/TLS connection to use weak RSA "export" keys, which can be easily broken, making it possible to intercept and decrypt encrypted communications.

### Affected Systems

* **Software:** OpenSSL, Apple's Secure Transport, and other SSL/TLS implementations.
* **Devices:** A wide range of devices and software that rely on these SSL/TLS libraries for secure communications.

<https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2015-0204>



### Detailed Solution for CVE-2015-0204 (FREAK Vulnerability)

The solution to CVE-2015-0204 involves several steps to ensure that systems are no longer vulnerable to the FREAK attack. Here is a comprehensive guide:

#### Step 1: Identify Affected Systems

**Inventory Assessment**:

* Identify all systems, applications, and devices that use SSL/TLS for secure communications.
* Determine which systems use affected libraries like OpenSSL, Apple's Secure Transport, or others.

**Check Versions**:

* Ensure the versions of SSL/TLS libraries in use. Vulnerable versions of OpenSSL, for example, need to be identified.

#### Step 2: Apply Patches and Updates

**Update OpenSSL**:

* Upgrade to a non-vulnerable version of OpenSSL. The following versions include the fix:
* OpenSSL 1.0.2
* OpenSSL 1.0.1k
* OpenSSL 1.0.0p
* OpenSSL 0.9.8zd
* Use package managers (like apt-get, yum, etc.) to update OpenSSL:

sudo apt-get update

sudo apt-get upgrade openssl

#### Step 3: Configuration Changes

**Disable Export Cipher Suites**:

* Modify server configurations to disable support for export-grade cipher suites. For example, in Apache HTTP Server, edit the SSL configuration file:

SSLCipherSuite HIGH:!aNULL:!MD5:!RC4:!EXP SSLHonorCipherOrder on

**Test Configurations**:

* Use tools like *nmap* or *ssllabs.com* to verify that the export cipher suites are disabled.

**Restart Services**:

* Restart the affected services to apply the new configurations

sudo systemctl restart apache2

sudo systemctl restart nginx

### Step 4: Verification and Monitoring

**Verify Fixes**:

* Use vulnerability scanners or security tools to verify that the FREAK vulnerability has been mitigated.
* Perform tests to ensure that the services are functioning correctly after updates and configuration changes.

**Continuous Monitoring**:

* Keep systems updated with the latest security patches.
* Monitor security advisories for new vulnerabilities and fixes.