Analysis of NAS Device Vulnerabilities and Using Nmap

Dušan Kocić[0009-0000-7451-7107], Goran Stamenović[0000-0001-6241-0425], and Nemanja Zdravković[0000-0002-2631-6308]

Faculty of Information Technology, Belgrade Metropolitan University
Tadeuša Košušća 63, 11000 Belgrade, Serbia

dusan.kocic.6382@metropolitan.ac.rs
goran.stamenovic@metropolitan.ac.rs
nemanja.zdravkovic@metopolitan.ac.rs

Abstract. Network-Attached Storage (NAS) devices are widely deployed in enterprise and consumer environments as centralized storage solutions for file sharing, backup, and IoT integration. However, improper configuration, outdated firmware, and lack of network segmentation can expose these devices to severe security risks. This paper presents a practical vulnerability assessment of a real-world Wi-Fi network, with a specific focus on NAS exposure, using the Nmap tool and its Nmap Scripting Engine. In this paper, we apply host discovery, TCP SYN scanning, service and version detection, and vulnerability probing through publicly documented CVEs. The analysis reveals multiple critical vulnerabilities which enable unauthenticated administrator password reset and allows full system compromise. Additional findings included directory traversal and credential exposure vulnerabilities on a router, an exposed UPnP service, and a legacy phpMyAdmin Local File Inclusion flaw.
The results demonstrate how a single misconfigured NAS device, combined with router-level vulnerabilities and lack of segmentation, can compromise the integrity of an entire network. The study underscores the importance of continuous vulnerability assessment, timely firmware updates, secure configuration, and proactive network hardening practices.

Keywords: NAS · Nmap · NSE · Network security · Network auditing · NAS security.

1 Introduction

In today's digital environment, the demand for efficient and reliable data storage solutions continues to grow. Network-Attached Storage (NAS) devices have emerged as a popular and practical choice, offering a centralized file storage that enables multiple users and heterogeneous client devices to retrieve data from centralized disk capacity, unlike traditional storage solutions which are attached directly to a personal computer, a NAS provides a centralized storage repository that can be accessed by multiple users at the same time [1]. In the context of IoT (Internet of Things) NAS devices are used as a centralized point of storage, processing and security of the data that IoT devices produce in their environment, especially in "smart" housing, industrial IoT and healthcare [2].

Research Paper
DOI: https://doi.org/10.46793/BISEC25.214K
Part of ISBN: 978-86-89755-40-4

NAS devices also serve as a file sharing solution, where it is possible for multiple users to have access to the same files. Such functionality is facilitated by platform agnosticism, which enables networked clients to retrieve files regardless of device architecture, utilizing protocols such as Server Message Block (SMB), Apple Filling Protocol (AFP), Network File System (NFS), and Web Distributed Authoring and Versioning (WebDAV) [3]. Widely used for backup purposes, especially in small and medium-sized businesses and IT environments requiring dependable and secure data storage, NAS devices provide backup operations for computers, servers, mobile platforms and IoT systems. The ever growing number of deployed NAS devices, both in enterprise and consumer environments, raise a critical concerns regarding their security posture. Many NAS system operate with default or old system configurations, which expose vulnerable services such as SMB, File Transfer Protocol (FTP), Hyper Text Transfer Protocol (HTTP), and often lack proper segmentation or encryption [4]. These factors make them attractive targets for malicious actors seeking unauthorized aces, data stealing, or ransomware deployment [5]. Given their role as core storage units in business, academic, and home environments, where data accessibility and reliability are paramount, NAS devices must be properly secured. Ensuring their safe operation requires proactive configuration and protection against cyber threats. Interconnected devices beside NAS, inside a Wi-Fi network often store sensitive data or provide access points into one's infrastructure, if left unchecked, those devices become prime targets for attackers [6].

One of the most effective tools for assessing the security of NAS device and other network hardware for its security is Nmap (Network Mapper), using Nmap is considered as a critical step in securing a Wi-Fi network. Nmap is a powerful open-source tool used for network discovery and security auditing, it allows administrators and security personel to scan devices on a network, detect open ports and running services, identify operating systems and firmware versions and to discover vulnerabilities and misconfigurations in network. Widely adopted in penetration testing and routine network maintenance, Nmap plays a critical role in securing modern Wi-Fi networks and guarding sensitive infrastructure [7].

Besides their role as a centralized storage units, NAS devices often serve as integral components within broader network infrastructure. When improperly configured or left unmonitored, they can cause significant security risks, not only to the data they store, but to the entire network that they inhabit. A vulnerable NAS device can potentially run outdated services, or act as a gateway for lateral movement, allowing attackers to infiltrate into other systems within the same Wi-Fi network, therefore harming the entirety of one network infrastructure. In this context, evaluating the security of NAS device, cannot be isolated from assessing the security of one network as a whole. Open-source tools like Nmap are used as an essential tool in this process. Nmap enables thorough scanning of all devices connected to a network, which makes the program crucial for uncovering security gaps that might otherwise remain hidden in network. In the course of this research, a network scan using Nmap exposed several critical vulnerabilities within a local Wi-Fi environment. Among them, a particularly severe case

involved a vulnerable NAS device, a Seagate BlackArmor NAS 110/220/440, exposing a known flaw that allowed unauthenticated administrator password reset via a publicly accessible script. This discovery became a pivotal point in the study, highlighting how a single misconfigured NAS unit can harm not only the integrity of its stored data, but the security of the entire network [8].

In addition to the NAS vulnerability, the scan revealed several other high-risk exposures, including an open UPnP port (49185/tcp) running a potentially vulnerable version, a phpMyAdmin instance affected by CVE-2005-2869 LFI (Local File Inclusion), and traces of CVE-2009-3733 present in legacy Vmware server configurations. These findings illustrate how multiple weak points, can coexist within the same network, all together significantly increase the overall risk of security in network infrastructure [9, 10].

All of the above named vulnerabilities were identified using Nmap, underscoring the tool's effectiveness in uncovering hidden problems and threats across diverse device types. A detailed analysis of this case, with special attention to NAS device will be presented in the following Sections of this paper. Section 2 deals with the methodology used. Section 3 gives the results and discussion, while Section 4 concludes the paper.

2 Methodology

This research employed a practical, hands-on approach to network vulnerability assessment with main orientation to NAS devices found in the given network using Nmap, a open-source tool for network discovery and security auditing. The objective of research was to identify potential weakness in a real-world Wi-Fi environment through scanning techniques, including ping sweep, port scanning and vulnerability scanning. The methodology involved next key steps – identification of devices connected to a private Wi-Fi network, analysis of the networks gateway, testing the gateway for possible vulnerabilities and evaluation of the discovered security issues. Our environment during this endeavor was a local coffee shop, which has its own WiFi network, that is protected by a password, which was already known. The network infrastructure itself consisted of a router, a NAS device (Seagate BlackArmor NAS 110/220/440) and client devices that were mutually visible, indicating the absence of segmentation in tested network, as shown in the Fig. 1. While the specific operational role of the NAS within the café infrastructure was not known, its presence and exposure to the network made it a relevant target for analysis.

Tools used while testing the shown network were Nmap, PowerShell, Online CVE Databases and various online sources for vulnerability analysis. Nmap, as a primary tool for network scanning and vulnerability detection, was used for identifying active host, mapping the network, scanning open ports, detecting running services, and gathering system-level information about devices within the tested Wi-Fi network. Complementing Nmap's core capabilities, the Nmap Scripting Engine (NSE) was used to detect and confirm the presence of known vulnerabilities through targeted script execution. The NSE is considered as one

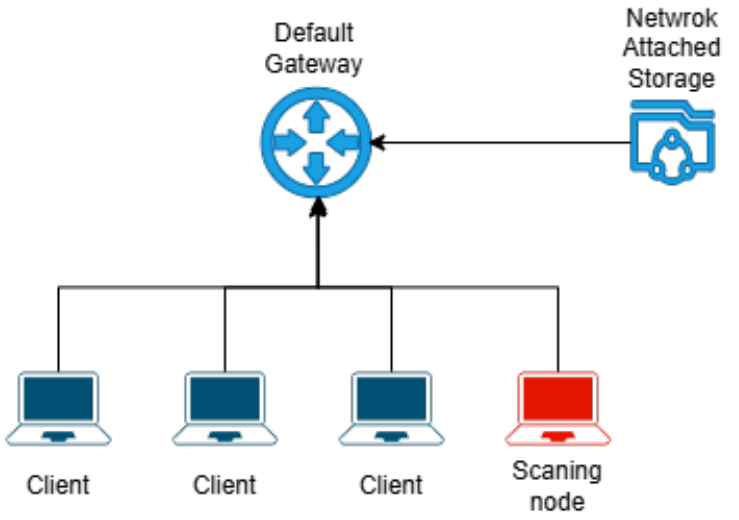

Fig. 1. Network diagram.

of Nmap's most powerful and flexible features, as it allows users to execute custom scripts written in Lua programming language, targeting specific protocols, services, know vulnerabilities and in some cases its able to even exploit them directly. In this study NSE scripts were used to probe devices for misconfigurations, outdated services and known CVEs [7]. This process included using scripts from the "vuln" category, which automate checks for exploitable conditions like unauthenticated access, weak encryption and more. Together, PowerShell, which was used for executing commands, Nmap and its NSE engine provided a thourough scanning, providing both surface-level mapping and device detection and in-depth vulnerability detection within the given network environment.

After connecting to the given Wi-Fi environment, from the scaning node (personal device used for testing), the process of assessing the security posture of devices in the environment, a series of structured Nmap scans were conducted. Each scan type was selected and executed on its ability to reveal specific aspects of the network and its potential security risk.

Initial step involved identifying active hosts within the Wi-Fi network using a ping sweep. This scan showed which devices were online and responsive to our Internet Control Message Protocol (ICMP) or Adress Resolution Protocol (ARP) requests:

```
nmap -sn 192.168.1.0/24
```

This command was used to map the entire network and enumerate all reachable devices, including the NAS unit and default gateway. We have seen that all of the connected device were mutually visible and have concluded that there was no segmentation of the network. From this ping sweep we have also gathered

information about the IP address of our router, which became the main target of our further scans.

The following step of assessment included a comprehensive scan that was performed on the networks default gateway.

```
nmap -sS -sV -O -Pn 192.168.1.1
```

This command used a variation of multiple techniques. TCP SYN scan (`-sS`) to identify open ports without completing full Transfer Control Protocol (TCP) handshakes, minimizing detection. Service and version detection (`-sV`) to enumerate active services and expose their software versions. Operating system fingerprinting (`-O`) to determine the underlying OS based on TCP/IP behavior. No ping (`-Pn`) to bypass ICMP-based host discovery, ensuring the scan proceeds even if ping responses are blocked [7].

The main goal of this step was to profile the router acting as the default gateway, revealing its exposed services and potential further vulnerabilities and security risks. This scan provided insight into the routers configuration, its open ports and whether it could serve as a pivot point for lateral movement inside the network. The router had a number of open TCP ports and open services. Among them were multiple ports that could pose a security risk, which will be discussed later in this paper. The scan confirmed that the device was one hop away, indicating direct accessibility from any client that was connected to the tested network at that time.

Following the initial profiling of the default gateway, we have conducted a targeted vulnerability scan using NSE:

```
nmap -script vuln 192.168.1.1
```

This command uses a collection of NSE scripts to check for known vulnerabilities associated with public CVEs. The goal of this step was to potentially detect exploitable vulnerabilities across visible hosts in the interconnected environment. By using Nmap's NSE, the scan aimed to uncover publicly documented CVEs that could be exploited by an potential attacker within the network. In this case, the scan revealed a Local File Inclusion (LFI) vulnerability in the grab_globals.lib.php component of phpMyAdmin, specifically CVE-2005-3299. This detection indicated that there is a need to do a deeper security check, especially in an unsegmented Wi-Fi network.

In the final phase of the assessment, a focused scan was executed against the router using a custom command that combined port targeting, vulnerability detection, and exploit probing:

```
nmap -p 21,22,23,25,53,80,110,139,143,443,445,587,8080
-script vuln,exploit -script-args=unsafe=1
-oN scan.txt 192.168.1.1
```

This scan was designed to target specific high-risk ports, invoke both vulnerability (`vuln`) and exploitation (`exploit`), also including aggressive probing (`-script-args=unsafe=1`), allowing execution of scripts that may trigger real exploit conditions. After the scan was finished we had an text file where the result was stored (scan.txt). The purpose of this step was to simulate a realistic attacker's approach on the given network. Unlike the first steps, which were

passive enumeration, this step actively tested whether services running on the gateway, or other indirectly visible devices had any of the documented CVEs and known exploit vectors.

In our final step we have seen a couple of serious security risks, that had a significant exploitation risk. Despite the scan being targeted at the router itself, the result showed a critical vulnerability on a Seagate BlackArmor NAS device present in the same network, it is documented under the CVE-2012-2568, and it allows unauthenticated administrator password reset via the NAS's web interface. This confirmed the NAS was fully exposed within the local network and reachable from any client device, this combined with other known vulnerabilities that were found exposed this networks significant security flaws making it a unsecure network and an easy target for potential attackers.

The pseudocode for the previously explained steps is given below.

```
nmap -sn 192.168.1.0/24
// basic ping sweep used to discover active host
nmap -sS -sV -O -Pn 192.168.1.1
// tcp scanning and service detection
// performed at the router
nmap -script vuln 192.168.1.1
// basic vulnerability detection
nmap -p 21,22,23,25,53,80,110,139,143,443,445,587,8080
--script vuln,exploit --script-args=unsafe=1 -oN scan.txt 192.168.1.1
// port targeting, vulnerability detection, and exploit probing
```

3 Results and discussion

3.1 Discovering active devices and network segmentation

Host discovery is the process of finding live (available) hosts on a network and its considered one of the earliest phases of network reconnaissance. One of the very first steps in any network reconnaissance mission is to discover active hosts and reduce a set of IP ranges into a list of active or potentially interesting hosts. Scanning every port of every single IP address is slow and in most cases unnecessary. In practice the list of interesting hosts depends greatly on the scan purpose. For instance network administrators may be interested in hosts running a specific service, while security auditors may care about every single device with an IP address. For this diverse purposes, the used program Nmap offers a wide variety of options for customizing the techniques for host discovery. In our endeavor we have used a "No port scan", otherwise called ping sweep or ping scan, as shown below [7]:

`nmap -sn 192.168.1.0/24`

The default host discovery done with -sn consists of an ICMP echo request, TCP SYN to port 443, TCP ACK to port 80, and an ICMP timestamp request by default. This option tells Nmap not to do a port scan after host discovery, and only print out the available hosts that responded to the host discovery

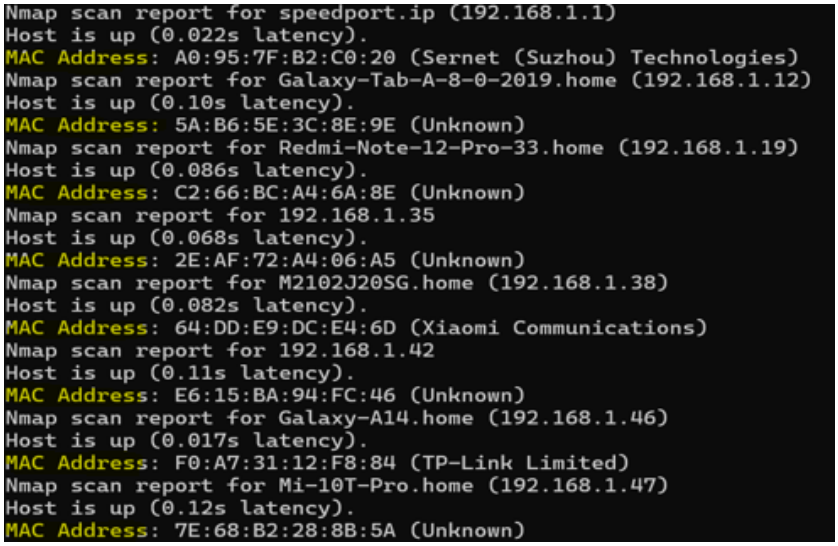

Fig. 2. Ping sweep results.

probes. Ping scan (`-sn`) allows light reconnaissance of a target network without attracting much attention. Both system administrators and potential attackers find this scan valuable. In this paper, during our endevour, after we ran this command in PowerShell, we have gotten the next results shown in Fig. 2 [7].

This scan identified multiple active hosts within the local network, including a router (192.168.1.1), and all of the other personal devices that were connected. While the scan successfully revealed several reachable hosts, it is important to note that ping sweep relies on ICMP and ARP responses, and may not detect devices that were configured to ignore such requests. This configuration became evident later on, when a vulnerability scan revealed the presence of a Seagate BlackArmor NAS device that was not visible in the initial host discovery, but was reachable and vulnerable via service-level exposure. This scan however revealed that this network did not have any segmentation whatsoever, based on number of physically visible devices, and number of devices that were shown by the done ping sweep.

3.2 TCP port scanning and service identification

When TCP scan is performed, one gets a detailed insight into how a target device communicates over the network, it is a foundational technique used to enumerate services exposed by target host. Since almost all network applications communicate over TCP, scanning TCP ports reveals which services are active and reachable. By identifying open ports and services working on them, potential attackers can map the attack surface of a device and determine which

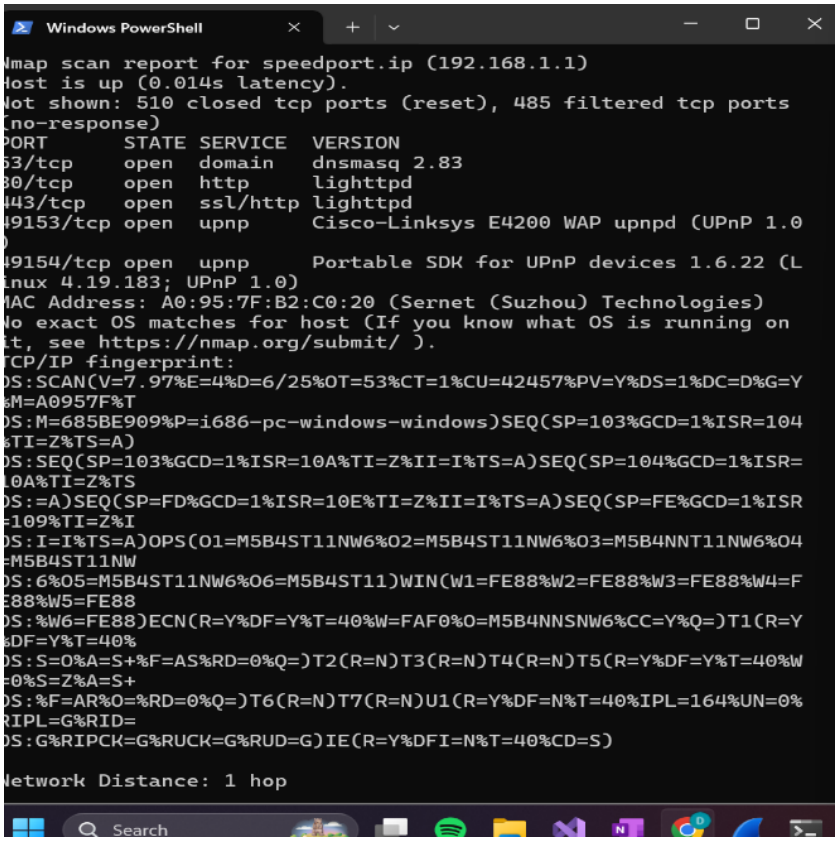

Fig. 3. TCP SYN scanning.

components are worthy of further inspection. In this paper, a TCP SYN scan (`-sS`) was chosen, its commonly referred to as a half-open scan. A half-open scan works by sending the initial SYN packet, without completing the handshake by sending the ACK packet. By doing so, the scanner can determine whether a port is open or closed, all while avoiding the full connection process that would typically be logged by the target system [7].

Additionally, our command used a service/version detection (`-sV`) and OS fingerprinting (`-O`) to gather deeper insights into the software stack and operating system characteristics. When scanning a device, discovering open ports is only the first step. To assess risk, we need to know what services are running and which versions, because vulenrabilities are tied to certain software versions. Nmap's `-sV` enables service and version detection by sending probes to open ports and analyzing the responses from them. Then it compares these responses to a large database of known service fingerprints. All of the above named allow Nmap to identify service name (e.g., HTTP, DNS), service version (e.g., OpenSSH,

Apache, dnsmasq) and version number (e.g., OpenSSH 7.6p1, Apache 2.4.54). To enumerate open ports and identify active services on the default gateway, a combined TCP scan was used as shown below.

```
nmap -sS -sV -O -Pn 192.168.1.1
```

This scan performed TCP SYN scanning (`-sS`), service and version detection (`-sV`), operating system fingerprinting (`-O`) and ping bypass (`-Pn`), making sure the scan proceeds even if the host block ICMP requests. When this command was executed, we have gotten the next results shown in Fig. 3.

Figure 3 shows a few open ports and services that can introduce potential risks in this environment, especially Universal Plug and Play (UPnP) services, that are particulary dangerous in consumer networks. UPnP is a service that allows devices on the same local network to discover each other and automatically connect through standard networking protocols (such as TCP/IP HTTP, and DHCP). UPnP can also modify router settings to open ports into a firewall to facilitate the connection of devices outside of a network. Though the UPnP protocol is safe, it can facilitate insecure connections. A UPnP protocol could permit devices with critical vulnerabilities to connect to network and sensitive resources. For instance, the U.S department of homeland security urged all businesses to disable their UPnP following a cyberattack in 2013 impacting tens of millions of devices [11]. Because of above given reasons and the fact that its difficult to determine if a connection could facilitate a malware infection, it is the best advised to disable UPnP ports.

3.3 Checking for known vulnerabilities (NSE scripts and CVE refferences)

To assess whether the target device is affected by known vulnerabilities, this phase employed the Nmap Scripting Engine (NSE). NSE is a powerful extension of Nmap that allows users to run custom scripts for advanced scanning tasks. These scripts are written in Lua programming language and can preform a wide range of functions such as malware detection, vulnerability detection, vulnerability exploitation, backdoor detection and more sophisticated version detection. Nmap includes hundreds of scripts, and the vuln category specifically targets known vulnerabilities by probing services and analyzing responses.

These scripts reference CVE identifiers to match detected software versions against documented exploits. CVE (Common Vulnerabilities and Exposures) is an international standard of set of security threats that are included in a reference system that outlines publicly known risks. The CVE threat list is maintained by the MITRE Corporation, a nonprofit organization that runs federal government-sponsored research and development centers. CVE defines vulnerabilities as a mistake within software code, which enables an attacker to gain direct unauthorized access to computer systems and networks and spread malware [12]. This typically allows attackers to pose as system administrators or superusers with full access privileges to corporate resources. Nmap includes a set of vulnerability detection scripts in the vuln category. These scripts send crafted probes to services, look for specific behaviors or misconfigurations and

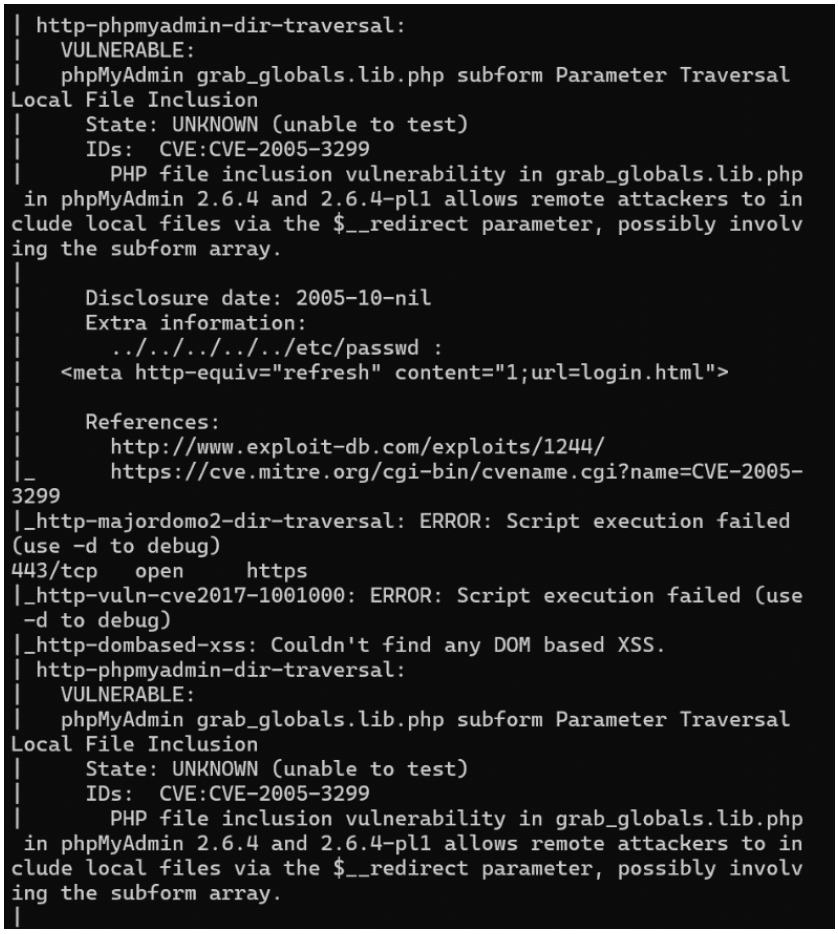

Fig. 4. Scanning for known vulnerabilities.

compare responses to known exploit signatures. If a service matches a known vulnerability, the script will output the CVE identifier and a brief description of the issue. Sometimes, this scripts can show links to advisories or exploit databases. To preform this part of the assessment, the following script was used:

```
mmap --script vuln 192.168.1.1
```

This command instructs Nmap to run all scripts in the `vuln` category against the target IP. These scripts test for known vulnerabilities in services previously identified on the device, including DNS, HTTP, HTTPS, and UPnP. After the execution of this command we have gotten the following results, shown below in Fig. 4.

During execution, the scan identified a Local File Inclusion (LFI) vulnerability affecting phpMyAdmin, specifically in the file grab_globals.lib.php. This

vulnerability is referenced as CVE-2005-3299, and it allows remote attackers to include local files via the `$__redirect` parameter in HTTP requests. LFI allows the attackers to read sensitive files on the server and if combined with other vulnerabilities it may lead to remote code execution. Nmap's vuln script for CVE-2005-3299 simulates the attack by crafting a request that includes the `$__redirect` parameter with a malicious value, which was shown in "Extra information" section of the report. If the server responds with content from that file, or even just fails in a predictable way, the script flags the vulnerability as confirmed. In this case, when Nmap used the directory traversal payload which is used in LFI attacks, to move up five directories from the current location and access the system file (/etc/passwd), the server responded with a HyperText Markup Language (HTML) meta tag (`<meta http-equiv="refresh" content="1;url=login.html>`) that tells the browser to redirect the user to login.html after 1 second.

This suggest that the server received the malicious LFI request, but instead of exposing file content, it redirected the response. In most cases, this behavior means that the LFI vulnerability exists, but authentication is required to exploit it fully [9].

3.4 Vulnerability detection on NAS devices

NAS devices are commonly deployed in local networks with the goal of centralized file sharing, backup, media streaming and integration with IoT devices [13, **?**]. In most environments NAS units are connected directly to the default gateway via Ethernet or WiFi. During initial reconnaissance, a ping sweep was performed to identify active hosts on the network, however the scan did not did not directly discover a NAS device. Certain NAS devices do not respond to ICMP echo request, likely due to firewall rules or ICMP filtering, power-saving or OS-level restrictions. As a result, NAS device was not detected during the initial step. However during the vulnerability scanning on the router, the NAS was later identified. A vulnerability scan directed at the default gateway revealed a known CVE associated with a NAS service. This behavior is possible in scenarios where the router exposes NAS functionality through port forwarding, UPnP, or integrated web interfaces. In such configurations, the router acts as a proxy or relay for NAS services, meaning that a scan of the routers IP can show responses from services actually hosted on the NAS. This indirect detection method highlights the importance of scanning not only endpoint devices but also the network infrastructure that may expose or relay access to them. In our endeavour the following script was executed directly onto the router:

```
nmap -p 21,22,23,25,53,80,110,139,143,443,445,587,8080
-script vuln,exploit -script-args=unsafe=1
-oN scan.txt 192.168.1.1
```

This command is used to scan a list of commonly used ports and invoke both `vuln` and `exploit` script categories. The `-script-args=unsafe=1` flag enables more aggressive check that simulate exploit conditions, and the output of this

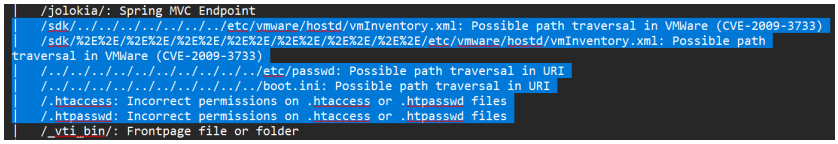

Fig. 5. Scan for `vuln` categories.

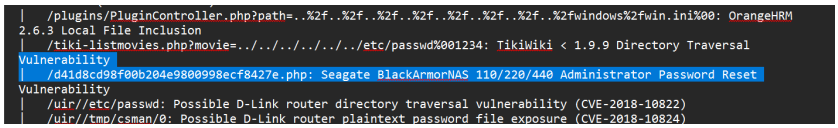

Fig. 6. Scan for `exploit` categories.

command is further saved on a text file named scan.txt. When this command was executed, we got the following result, shown below in Figs. 5 and 6.

from Figs. 5 and 6 we can see that there is a few critical vulnerabilities in this network, among them, was the Seagate BlackArmor NAS administrator password reset vulnerability, formally documented as CVE-2012-2586. This vulnerability affects Seagate NAS models 110, 220 and 440, and it allows remote attacker to reset the administrator password without any authentication by accessing a static PHP endpoint (d41d8cd98f00b204e9800998ecf8427e.php) on the device. Presence of this vulnerability implies full compromise of the present NAS, including unauthorized access to data stored on the NAS, configuration files, and potentially other devices on the network. The exploitation of this vulnerability is fairly easy. A potential attacker, needs to locate NAS device's IP address, and just send a direct HTTP GET request to the vulnerable endpoint (http://192.168.1.x/d41d8cd98f00b204e9800998ecf8427e.php).

This action will trigger the NAS device to reset the administrator password to a default or blank value, or even allow the attacker to set a new one. After that step, attacker can then log into the administrator panel and gain full control over the device. This attack has the following impact:

- Full administrative access to the NAS
- Exposure of sensitive files and backups
- Potential pivoting into the other network hardware
- Loss of data confidentiality and integrity

According to the Nation Institute of Standards and Technology (NIST), the found vulnerability in Seagate BlackArmor NAS, was assigned a CVSS v2 Base Score of 10.0 (HIGH), which is the maximum possible severity. CVSS (Common Vulnerability Scoring System) is a standardized framework for rating the severity of software vulnerability, it evaluates and ranks reported vulnerabilities in a standardized and repeatable way. The corresponding vector string is (AV:N/AC:L/Au:N/C:C/I:C/A:C) which means that this vulnerability is:

– Remotely exploitable over the network (AV:N)
– Requires no special conditions or complexity (AC:L)
– Requires no authentication (Au:N)
– Results in complete compromise of confidentiality (C:C)
– Allows full modification of system integrity (I:C)
– Enables complete disruption of availability (A:C)

This combination represents the highest possible severity rating under CVSS v2 and reflects a worst case scenario where the attacker gains full control over the given NAS device, with minimal effort while doing so. In the context of this paper, CVE-2012-2568 serves as an main example of how NAS vulnerabilities can effect the integrity of the whole network, and how NAS vulnerabilities can be discovered indirectly through router scans [8, 15].

In this scan additional vulnerabilities, worth of mention, were also detected. The scan found a possible directory traversal vulnerability in VMware Server, which allows remote attackers to read arbitrary files via unspecified vectors. On the second picture we can see two possible vulnerabilities CVE-2018-10822 and CVE-2018-10824. The CVE-2018-10822 is a directory traversal vulnerability in the web interface on D-Link router, where the device allows remote attackers to read arbitrary files via a `/..` or `//` after `GET /uir` in an HTTP request. Second vulnerability, much more severe, the CVE-2018-10824, is also on the D-Link router, and its possible that the administrative password I stored in plaintext in the `/tmp/csman/0` file. An attacker having a directory traversal (or LFI) can easily get full router access with this vulnerability [16, ?].

3.5 Discussion

The results obtained through the layered scanning methodology that was used, beginning with the devices discovery, followed by TCP scanning and service identification and vulnerability probing, reveal critical weaknesses in the tested network environment. Named findings have a direct implications for data confidentiality, device integrity and overall network security.

The scan revealed several high risk vulnerabilities across both router and NAS device. Based on the severity of the found vulnerabilities, the most severe one was Seagate BlackArmor NAS administrator reset (CVE-2012-2568). This vulnerability, as already mentioned, allows remote attackers to rest the administrator password via a static PHP file through NAS web interface. With a CVSS v2 Base Score of 10.0, it represents a complete compromise of NAS device. Once exploited, an attacker can gain full access to stored data, configuration settings, and can potentially pivot into the internal structure of the network. For this vulnerability, Seagate Software includes a fix, which addresses the previously publicized security hole, users with this vulnerability should update the software to the newer version.

Continuing with the second major security concern, we have discovered D-Link Router Directory Traversal and Password exposure (CVE-2018-10822, CVE-2018-10824). These vulnerabilities affect the router, and they allow attacker to

read arbitrary files and extract plaintext credentials from exposed endpoints such as `/tmp/csman0` and `/ect/passwd`. In the case of CVE-2018-10824, the administrative password is stored in the plaintext in the `/tmp/csman/0` file, allowing the attacker that has a directory traversal (or LFI) to easily gain full router access. On the other hand, CVE-2018-10822, allows remote attackers to read any file on the routers file system by sending a crafted HTTP request to the `/uir` endpoint (e.g. `GET /uir//etc/passwd`). Individually, each vulnerability is serious, but combined, the can potentially enable full compromise. CVE-2018-10822 gives attackers access to arbitrary files and CVE-2018-10824 ensures that oe of those files contains plaintext administrator credentials. For CVE-2018-10822, recommended fixes are firmware update or blocking the access to `/uir` and similar endpoints using routers Access Control Lists (ACL) or external firewalls. For CVE-2018-10824, its also advised to update firmware to a newer version, or to ensure the routers clear /tmp directories on reboot or restrict access to them.

Next vulnerability, worthy of mentioning is phpMyAdmin Remote File Inclusion (CVE-2005-3299). This vulnerability affects phpMyAdmin versions 2.6.4 and 2.6.4-pl1, allowing attackers to include local files via the `__redirect` parameter in `grab_globals.lib.php`. If phpMyAdmin is hosted on the router, this flaw can be used to access sensitive configuration files or escalate privileges. In the tested environment, router responded to this vulnerability with a login html page, which means that its somewhat protected, but can still be potentially exploited under certain conditions. The simplest fix for CVE-2005-3299 is to upgrade to phpMyAdmin 2.6.4-pl2 or any later version. As this router already redirects users to login.html page, its advised to change the login credentials on that page.

Its also worthy of mentioning the found open port `49153/tcp` with UPnP. Port 49253 with open UPnP can expose potentially vulnerable services that could be exploited for remote access or configuration manipulation. This exposure raises a bigger risk, if the router uses outdated firmware or the UPnP service is externally reachable. The main security implication is unauthorized port forwarding, where malicious applications can silently open ports to internal services bypassing firewall rules. Recommendations for this situation is disabling UPnP in the routers web interface unless there is a explicit need for internal device discovery, or updating the firmware to the latest version available from the Cisco/Linksys to patch known vulnerabilities.

Table 1 shows CVSS scores of each vulnerability mentioned, their severity, and likelihood to be exploited across the globe in the next 30 days (EPSS) [18].

Table 1. CVSS vulnerability scores.

CVE ID	CVSS Score	Severity	EPSS
CV-2012-2568	10.0	Critical	Low (1.10%)
CVE-2018-10822	7.5	High	High(86.73%)
CVE-2018-10824	9.8	Critical	High(44.03%)
CVE-2005-3299	5.0	Medium	Moderate (8.85%)

4 Conclusion

This research was done with the goal of systematical analysis of the security of a local network environment, with a focus on NAS exposure, and its impact on overall network stability and security. In this paper, we have mentioned how certain router misconfigurations, and ignorance of network maintenance can, overtime, become a serious security risk, especially in a network with on segmentation, that has a not so small number of users on daily basis. Through active scanning and targeted enumeration we have found several critical vulnerabilities which have potential to endanger data confidentiality, integrity, and availability of the whole network.

The most severe finding was the presence of CVE-2012-2568 on a Seagate BlackArmor NAS device, which was found when probing exploits on the default gateway, although that device was not directly identified in the initial ping sweep scan. This vulnerability allowed unauthenticated remote password reset via a vulnerable PHP script which was accessible through devices web interface. This security flaw, which is a product of ignorance and neglect of the software updates that need to be done from time to time, enabled full administrative takeover of the NAS without any credentials. Although this CVE is rarely seen with a EPSS score of 1.10%, it can still be found in some devices and represent a serious threat with major exploitation risk. Additionally, a chained vulnerability involving CVE-2018-10822 and CVE-2012-10824 was discovered on a D-Link router. The first allowed arbitrary file reads via directory traversal, while the second exposed plaintext administrative credentials. Together they enabled remote access and lateral movement toward internal assets of the network. Other notable weaknesses included exposed UPnP service on port 49153, which could be used for unauthorized port forwarding or device enumeration. Also, there were found legacy vulnerabilities such as CVE-2005-3299 (phpMyAdmin LFI) and CVE-2009-3733 (VMware Server file disclosure) were also detected, highlighting the risk of outdated software still present in the environment.

All of previously mentioned findings underscore the importance of continuous vulnerability assessment, timely patch management, and secure configuration of network services. Furthermore, the research emphasized the need for user education, particularly regarding the risks of exposing administrative interfaces and enabling insecure protocols like HTTP, UPnP, or unauthenticated web services.

Highlighting the importance of security of "ghost" NAS devices, and how they can pose as a major security risks, this paper analyzed one of those cases and underscored how combined vulnerabilities can absolutely compromise the security of the network environment. In conclusion, this assessment demonstrated that even a small network can harbor high-impact vulnerabilities, if the network has no proper maintenance strategy or there is a neglect of security concerns. Regular testing, combined with proactive hardening and informed user practices, remains essential to maintaining a resilient and secure network infrastructure, no matter the network size.

References

1. Amazon Web Services: What is NAS (Network-Attached Storage)? https://aws.amazon.com/what-is/nas/ (2025), accessed: 2025-11-01
2. Khandale, S.P.: Iot based network attached storage. Int. J. Sci. Technol. Eng **12**(11), 2247–2255 (2024)
3. Duong, M.: What Protocol Should You Choose For Your NAS? https://www.thegalah.com/choosing-right-protocol-your-nas-afp-vs-smb-nfs-iscsi (2023), accessed: 2025-11-01
4. Dimitrijević, N., Zdravković, N., Bogdanović, M., Mesterovic, A.: Advanced Security Mechanisms in the Spring Framework: JWT, OAuth, LDAP and Keycloak. In: Proceedings of the 14th International Conference on Business Information Security (BISEC 2023). pp. 64–70 (2024)
5. Yasui, H., Inoue, T., Sasaki, T., Tanabe, R., Yoshioka, K., Matsumoto, T.: SPOT: In-depth Analysis of IoT Ransomware Attacks Using Bare Metal NAS Devices. Journal of Information Processing **32**, 23–34 (2024)
6. Zhou, X., Wang, P., Zhou, L., Xun, P., Lu, K.: A survey of the security analysis of embedded devices. Sensors **23**(22), 9221 (2023)
7. Lyon, G.F.: Nmap network scanning: The official Nmap project guide to network discovery and security scanning. Insecure (2009)
8. NIST: National Vulnerability Database: CVE-2012-2568. https://nvd.nist.gov/vuln/detail/CVE-2012-2568 (2012), accessed: 2025-11-01
9. NIST: National Vulnerability Database: CVE-2005-2869. https://nvd.nist.gov/vuln/detail/CVE-2005-2869 (2005), accessed: 2025-11-01
10. NIST: National Vulnerability Database: CVE-2009-3733. https://nvd.nist.gov/vuln/detail/cve-2009-3733 (2009), accessed: 2025-11-01
11. Kost, E.: What is UPnP? https://www.upguard.com/blog/what-is-upnp (2025), accessed: 2025-11-01
12. Fortinet: What is a CVE? Meaning & Definition How Does CVE Define Vulnerabilities? https://www.fortinet.com/resources/cyberglossary/cve (2025), accessed: 2025-11-01
13. Fang, Z., Fu, H., Gu, T., Hu, P., Song, J., Jaeger, T., Mohapatra, P.: Iota: A framework for analyzing system-level security of iots. In: 2022 IEEE/ACM Seventh International Conference on Internet-of-Things Design and Implementation (IoTDI). pp. 143–155. IEEE (2022)
14. Bakhshi, T., Ghita, B., Kuzminykh, I.: A review of iot firmware vulnerabilities and auditing techniques. Sensors **24**(2), 708 (2024)
15. CVEdetails: CVEdetails, Vulnerability Details : CVE-2012-2568. https://www.cvedetails.com/cve/CVE-2012-2568/ (2012), accessed: 2025-11-01
16. NIST: National Vulnerability Database: CVE-2018-10822. https://nvd.nist.gov/vuln/detail/cve-2018-10822 (2018), accessed: 2025-11-01
17. NIST: National Vulnerability Database: CVE-2018-10824. https://nvd.nist.gov/vuln/detail/cve-2018-10824 (2018), accessed: 2025-11-01
18. Barricade Cyber Solutions: EPSS Lookup Tool. https://epsslookuptool.com/ (2025), accessed: 2025-11-01