EXTION

CYBER SECURITY INTERNSHIP

STEPHEN O. JOHN-EBOWE

PROJECT 1: NETWORK VULNERABILITY ASSESSMENT

13th AUGUST, 2024

1.0 INTRODUCTION

Vulnerability assessment is a fundamental process in cybersecurity that focuses on identifying and evaluating potential weaknesses within an organization's infrastructure, systems, and applications. This methodical approach aims to detect security gaps that could be exploited by attackers, thereby compromising the integrity, confidentiality, or availability of critical data and resources. By systematically analyzing various components such as networks, software, and hardware, vulnerability assessment helps organizations understand their security posture and prioritize remediation efforts to strengthen their defenses.

In the rapidly evolving digital landscape, the significance of vulnerability assessment has grown exponentially. As organizations increasingly rely on complex technological systems and interconnected networks, the potential entry points for cyber threats have multiplied. Effective vulnerability assessments allow organizations to proactively identify these vulnerabilities before they can be exploited, thereby mitigating the risk of cyber-attacks, data breaches, and other security incidents. This proactive stance not only safeguards sensitive information but also helps in maintaining trust and compliance with regulatory requirements.

A comprehensive vulnerability assessment typically employs a combination of automated tools and manual techniques to identify security flaws. Automated scanning tools can quickly detect known vulnerabilities across a broad range of systems, while manual testing and expert analysis provide deeper insights into more complex or subtle weaknesses. The results of these assessments are then used to develop a risk management strategy that includes remediation plans, ongoing monitoring, and periodic reassessment to adapt to new and emerging threats.

Overall, vulnerability assessment plays a crucial role in the broader framework of risk management and cybersecurity. By uncovering and addressing security weaknesses, organizations can enhance their resilience against potential threats, ensuring the protection of their digital assets and maintaining operational continuity in an increasingly hostile cyber environment.

1.1 Aims and Objectives

The aim of this project is to perform a comprehensive network vulnerability assessment. The following objectives are thus outlined;

1. To undertake a complete security audit of a test web platform http://target.dummy.sh/.

2. To use necessary tools to check for vulnerabilities and software exploits that may be present.
3. To identify vulnerabilities ranging from critical design flaws to simple misconfigurations.
4. To document the vulnerabilities so that developers can easily identify and reproduce the findings.
5. To create guidance to assist developers with remediating the identified vulnerabilities.

1.2 Proposed Deliverables

The network vulnerability assessment project is typically undertaken to assess the security of a network, identify vulnerabilities and recommend measures to mitigate those vulnerabilities. The project's expected deliverables can be summarized as follows:

1. **Comprehensive Security Assessment:** The primary deliverable is a detailed report outlining the security status of the website or web application. This report includes an assessment of potential vulnerabilities, security weaknesses, and a risk analysis.

2. **Vulnerability List:** A list of specific security vulnerabilities and weaknesses found during the audit, along with their severity ratings, is provided. This could include issues such as SQL injection, Cross-Site Scripting (XSS), Cross-Site Request Forgery (CSRF), and more.

3. **Recommendations:** The project should offer specific recommendations for mitigating or resolving identified vulnerabilities. These recommendations might include code changes, security configuration adjustments, and best practices to follow.

4. **Documentation:** Detailed documentation of the audit process, testing methodologies, and tools used should be included to ensure transparency and repeatability.

5. **Prioritization:** A prioritized list of vulnerabilities, with critical issues highlighted, helps the organization focus on the most urgent security concerns.

The project is expected to deliver a comprehensive assessment of the website's security, including a list of vulnerabilities and recommendations for mitigation. The project is needed to protect sensitive data, ensure compliance with regulations, prevent cyberattacks, and maintain an organization's reputation. The long-term benefits encompass improved security, cost savings, regulatory compliance, enhanced user trust, reduced legal liability, and opportunities for innovation and growth.

2.0 METHODOLOGY

A thorough vulnerability assessment methodology involves several key phases, each designed to systematically identify, analyze, and address potential security weaknesses within an organization's systems and infrastructure. Below is a detailed methodology for conducting an effective vulnerability assessment:

1. **Preparation and Planning**

   The initial phase of a vulnerability assessment involves defining the scope and objectives of the assessment. This includes identifying the systems, networks, applications, and assets to be evaluated, as well as understanding the organizational context and specific security requirements. Key tasks in this phase include:

   - **Defining Scope**: Determine which systems, applications, and network segments will be included in the assessment. This may involve discussions with stakeholders to understand critical assets and potential risks.
   - **Setting Objectives**: Establish clear goals for the assessment, such as identifying security weaknesses, prioritizing risks, or ensuring compliance with regulatory standards.
   - **Gathering Information**: Collect relevant data about the target environment, including network diagrams, system configurations, and existing security controls. This information helps in understanding the context and potential vulnerabilities.

2. **Information Gathering**

   The information gathering phase involves collecting detailed information about the systems and network architecture to identify potential vulnerabilities. This phase typically includes:

   - **Network Scanning**: Use network scanning tools to identify live hosts, open ports, and services running on the network. Tools such as Nmap can be employed to map the network and discover active devices.
   - **System Enumeration**: Gather detailed information about operating systems, applications, and configurations. This may include software versions, patch levels, and system configurations that could reveal potential vulnerabilities.

- **Threat Intelligence**: Utilize threat intelligence sources to understand current and emerging threats that could impact the target environment. This includes reviewing recent vulnerabilities, exploits, and attack trends relevant to the systems in scope.

3. **Vulnerability Identification**

In this phase, the collected information is analyzed to identify vulnerabilities that could be exploited by attackers. Key activities include:

- **Automated Scanning**: Deploy vulnerability scanning tools, such as Nessus, OpenVAS, or Qualys, to automatically detect known vulnerabilities based on a database of security issues. These tools scan the systems for misconfigurations, outdated software, and other security weaknesses.
- **Manual Testing**: Conduct manual testing to identify vulnerabilities that automated tools might miss. This includes techniques such as penetration testing, where ethical hackers simulate attacks to uncover more subtle or complex vulnerabilities.
- **Code Review**: For applications, perform a detailed code review to identify security flaws in the source code, such as coding errors, insecure coding practices, or other weaknesses.

4. **Vulnerability Analysis**

Once vulnerabilities are identified, they need to be analyzed to determine their potential impact and exploitability. This involves the following:

- **Risk Assessment**: Evaluate the severity of each identified vulnerability based on factors such as the ease of exploitation, potential impact on the organization, and the likelihood of exploitation. This helps prioritize vulnerabilities that pose the greatest risk.
- **Impact Analysis**: Assess the potential consequences of each vulnerability, including data breaches, system compromise, or disruption of services. This analysis helps in understanding the potential damage and prioritizing remediation efforts.

5. **Reporting and Documentation**

The results of the vulnerability assessment are documented and presented in a detailed report. This report should include the following:

- **Executive Summary**: Provide a high-level overview of the findings, including the most critical vulnerabilities and recommended actions for addressing them.
- **Detailed Findings**: Include a comprehensive list of identified vulnerabilities, their risk levels, and evidence supporting the findings. This may also include screenshots, logs, or other relevant data.
- **Remediation Recommendations**: Offer actionable recommendations for mitigating or eliminating the identified vulnerabilities. This may involve patching software, changing configurations, or implementing new security controls.
- **Risk Management**: Suggest strategies for managing residual risks that cannot be fully mitigated, including compensating controls or additional monitoring.

6. **Remediation and Follow-Up**

After the assessment report is delivered, the remediation phase involves addressing the identified vulnerabilities. Key tasks include:

- **Implementing Fixes**: Apply patches, update configurations, or make other changes as recommended in the assessment report. Ensure that remediation efforts are tested to confirm that vulnerabilities have been effectively addressed.
- **Verification**: Conduct follow-up scans or tests to verify that the vulnerabilities have been successfully mitigated and that no new issues have been introduced.
- **Continuous Monitoring**: Establish a process for ongoing monitoring and periodic reassessment to detect new vulnerabilities and ensure that the organization’s security posture remains robust.

7. **Review and Improvement**

Finally, review the entire vulnerability assessment process to identify lessons learned and areas for improvement. This phase involves:

- **Post-Assessment Review**: Evaluate the effectiveness of the assessment process, including the accuracy of findings and the efficiency of remediation efforts.
- **Process Improvement**: Implement improvements based on feedback and lessons learned to enhance future vulnerability assessments and overall security practices.

By following this comprehensive methodology, organizations can effectively identify and address vulnerabilities, thereby strengthening their security posture and reducing the risk of cyber threats.

2.1 Vulnerability Assessment

A vulnerability assessment is a systematic process that aims to identify, classify, and prioritize potential weaknesses or vulnerabilities in an organization's information systems, networks, applications, and infrastructure. It is a testing process used to identify and assign severity levels to as many security defects as possible in a given timeframe. This process may involve automated and manual techniques with varying degrees of rigor and an emphasis on comprehensive coverage. Using a risk-based approach, vulnerability assessments may target different layers of technology, the most common being host-, network-, and application-layer assessments. The primary goal of a vulnerability assessment is to proactively discover and assess security flaws before malicious actors can exploit them. This process helps organizations take preemptive measures to enhance their security posture and protect sensitive data.

Here are the key components and steps involved in a vulnerability assessment:

1. **Asset Identification:** Begin by identifying all the assets within your organization that need to be assessed. This includes hardware, software, servers, workstations, mobile devices, and network components.

2. **Vulnerability Scanning:** Utilize automated scanning tools and software to systematically scan and analyze these assets for known vulnerabilities. These tools often use a database of known vulnerabilities and security misconfigurations to identify potential weaknesses.

3. **Vulnerability Assessment:** Once vulnerabilities are identified, they are assessed for their severity and potential impact. This involves evaluating the potential risk associated with each vulnerability, considering factors such as likelihood of exploitation and the potential damage it could cause.

4. **Risk Classification:** Vulnerabilities are typically categorized based on their severity, usually using a scoring system such as the Common Vulnerability Scoring System (CVSS). This helps prioritize which vulnerabilities should be addressed first.

5. **Prioritization:** Based on the risk assessment, vulnerabilities are prioritized, and a plan is developed to remediate or mitigate them. High-risk vulnerabilities typically receive immediate attention, while lower-risk ones may be addressed in a phased approach.

6. **Remediation:** Remediation involves taking steps to fix or mitigate the identified vulnerabilities. This may include applying security patches, reconfiguring systems, updating software, or implementing security controls to reduce the risk.

Vulnerability assessments are a fundamental component of an organization's overall cybersecurity strategy. They provide valuable insights into an organization's security posture, help reduce the risk of data breaches and cyberattacks, and support compliance efforts. It's essential to conduct vulnerability assessments regularly and respond promptly to mitigate identified risks to enhance overall security.

2.2 Vulnerability Assessment Tools Used

Vulnerability scanners are valuable tools that search for and report on what known vulnerabilities are present in an organization's IT infrastructure. Using a vulnerability scanner is a simple, but critical security practice that every organization can benefit from. These scans can give an organization an idea of what security threats they may be facing by giving insights into potential security weaknesses present in their environment. Many organizations use multiple vulnerability scanners to ensure they're getting full coverage of every asset, creating a complete picture.

Vulnerability assessment tools are versatile tools for network administrators and cybersecurity professionals. However, it's important to note that unauthorized scanning of networks or systems is illegal and unethical. Always ensure to have the necessary permissions, proper authorization and legal rights before using a vulnerability scanner on a network or system. Over the years, many different scanners have been developed, providing a lot of different options and features. The vulnerability assessment tools used for this project include Urlscan.io, Nmap and Intruder Vulnerability Scanner.

2.2.1 Urlscan.io

Urlscan.io is an online service that allows users to scan and analyze URLs (Uniform Resource Locators) and websites for potential security threats and suspicious activities. It is a tool commonly used by cybersecurity professionals to examine the safety and integrity of web content by capturing and analyzing web page interactions, including HTTP requests and JavaScript execution, to identify and report on potential threats, such as phishing attempts, malware distribution, or other security issues. Users can submit URLs to the platform, and urlscan.io will provide detailed reports and insights into the web content's behavior and security risks. Urlscan.io has several uses in the field of cybersecurity and web security analysis, some of which include the following:

1. **Threat Detection:** Urlscan.io is used to identify and analyze potential security threats and malicious activities associated with URLs and websites, including phishing attempts, malware distribution, and other security risks.
2. **URL and Website Analysis:** It allows users to submit URLs or website links for comprehensive analysis, providing insights into the site's behavior, server details, domain information, and more.
3. **Phishing Detection:** Security professionals use urlscan.io to detect and report on phishing websites, helping to protect individuals and organizations from falling victim to phishing attacks.
4. **Malware Identification:** The service is instrumental in identifying websites that may be distributing malware or serving as a conduit for malicious code, enabling users to take appropriate action to protect their systems.
5. **Suspicious Behavior Monitoring:** Urlscan.io captures and analyzes web page interactions, including HTTP requests and JavaScript execution, to detect any unusual or suspicious behaviors that might indicate a security threat.
6. **Security Research:** Cybersecurity researchers and professionals use urlscan.io to gather data on web-based threats and trends, contributing to the development of threat intelligence.
7. **Incident Response:** During incident response activities, urlscan.io can be used to investigate potentially compromised websites, providing insights into the extent of an incident and the tactics used by attackers.
8. **Web Application Analysis:** Security professionals use urlscan.io to assess and analyze the security of web applications, identifying vulnerabilities, misconfigurations, and areas of concern.
9. **Information Gathering:** The platform can be used for passive reconnaissance to gather information about a target website, including IP addresses, DNS details, and other metadata.
10. **Public Resource for Sharing Threat Data:** Urlscan.io is often used to share threat data and reports with the broader cybersecurity community, enabling collective awareness and protection against emerging threats.

Urlscan.io is a versatile and valuable resource for web security analysis, threat detection, and security research, benefiting both individuals and organizations seeking to protect themselves from cyber threats.

2.2.2 Nmap

Nmap (Network Mapper) is a powerful and versatile open-source network scanning tool used for discovering and analyzing network hosts and services. Nmap is widely used by network administrators, security professionals, and ethical hackers to gather information about network devices and identify potential security vulnerabilities. Key features of Nmap include the following:

1. **Host Discovery:** Nmap can be used to discover hosts on a network by sending ICMP Echo requests, TCP SYN/ACK packets, and other probes to determine which hosts are online and responsive.
2. **Port Scanning:** It can scan and identify open ports on network hosts, which helps in understanding which services are running and accessible.
3. **Service Enumeration:** Nmap can identify the services running on open ports, such as identifying web servers, FTP servers, SSH, and other applications. It also retrieves version information when available.
4. **Operating System Detection:** Nmap can attempt to identify the operating system of a target host based on various network characteristics and response patterns.
5. **Scripting Engine:** Nmap includes a scripting engine (Nmap Scripting Engine or NSE) that allows users to create and run custom scripts to perform more advanced tasks like vulnerability scanning, banner grabbing, and service enumeration.
6. **Timing and Performance Options:** Users can control the speed and aggressiveness of scans using various timing options, making it possible to balance thoroughness with speed.
7. **Output Formats:** Nmap can produce output in multiple formats, including text, XML, and grepable formats, making it easy to process and analyze the scan results.
8. **Extensive Options:** Nmap offers a wide range of scanning techniques, such as SYN, FIN, XMAS, NULL, and more, each with its own unique purpose.
9. **Security Auditing:** Nmap is frequently used for security audits and vulnerability assessments. It can identify vulnerabilities, misconfigurations, and security weaknesses in network services.
10. **Script Repository:** Nmap has a vast repository of NSE scripts that can be used to automate and extend its functionality.

2.2.3 Intruder Vulnerability Scanner

Intruder vulnerability scanner is a cloud-based vulnerability scanning and assessment platform designed to help organizations identify and manage security vulnerabilities in their IT infrastructure and web applications. Intruder vulnerability scanner provides automated scanning and reporting capabilities to help businesses improve their cybersecurity posture. Here are some key features about this tool:

1. **Automated Scanning:** Intruder automates the process of scanning networks, web applications, and internet-facing systems for vulnerabilities. It uses a database of known vulnerabilities and security checks to identify potential weaknesses.

2. **Asset Discovery:** The platform helps organizations discover and inventory their assets, including devices, servers, applications, and websites that need to be scanned for vulnerabilities.

3. **Vulnerability Assessment:** Intruder assesses the severity of identified vulnerabilities and provides detailed reports, including information about the risk, potential impact, and remediation recommendations.

4. **Prioritization:** The system categorizes vulnerabilities based on their severity, allowing organizations to prioritize which vulnerabilities to address first.

5. **User-Friendly Dashboard:** Intruder provides a user-friendly dashboard that allows users to monitor the progress of scans, view reports, and access actionable insights about their security posture.

6. **Alerts and Notifications:** Users can configure alerts and notifications to stay informed about new vulnerabilities or changes in their security status.

7. **Integration:** Intruder may offer integrations with other security tools and platforms, allowing organizations to incorporate vulnerability assessment into their broader security strategy.

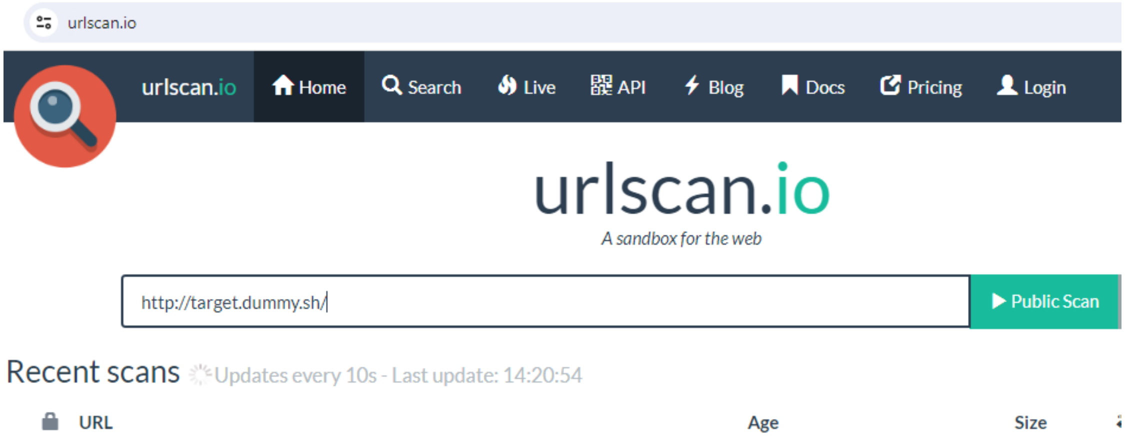
8. **Compliance:** The platform may assist organizations in meeting compliance requirements by identifying vulnerabilities that could lead to compliance violations.

9. **Custom Scanning:** Intruder may allow users to customize scans to focus on specific areas of their infrastructure or applications.

10. **Continuous Monitoring:** Regular scans can be scheduled to ensure that newly discovered vulnerabilities or changes in the environment are promptly addressed.

Intruder vulnerability scanner gives a real view of the attack surface combining continuous network monitoring, automated vulnerability scanning, and proactive threat response in one platform. With actionable results prioritized by context, Intruder helps focus on fixing what matters, bringing an easy effectiveness to vulnerability management. Intruder keeps track of the attack surface, showing where and how there may be vulnerability, prioritizing issues and filtering noise.

2.3 Web Analysis Using Urlscan.io



urlscan.io

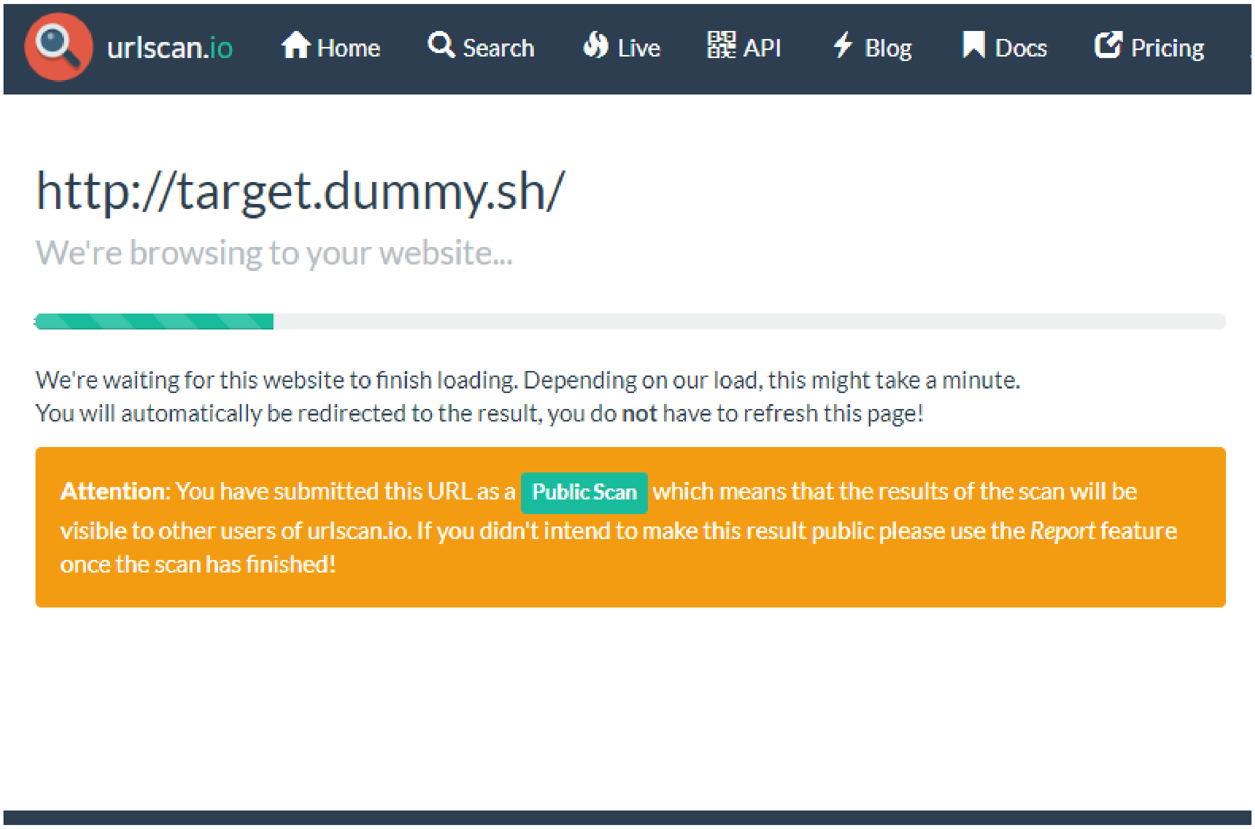
A sandbox for the web

http://target.dummy.sh/

Recent scans Updates every 10s - Last update: 14:20:54

URL	Age	Size

Fig. 1.0: Web Analysis Using UrlScan.io



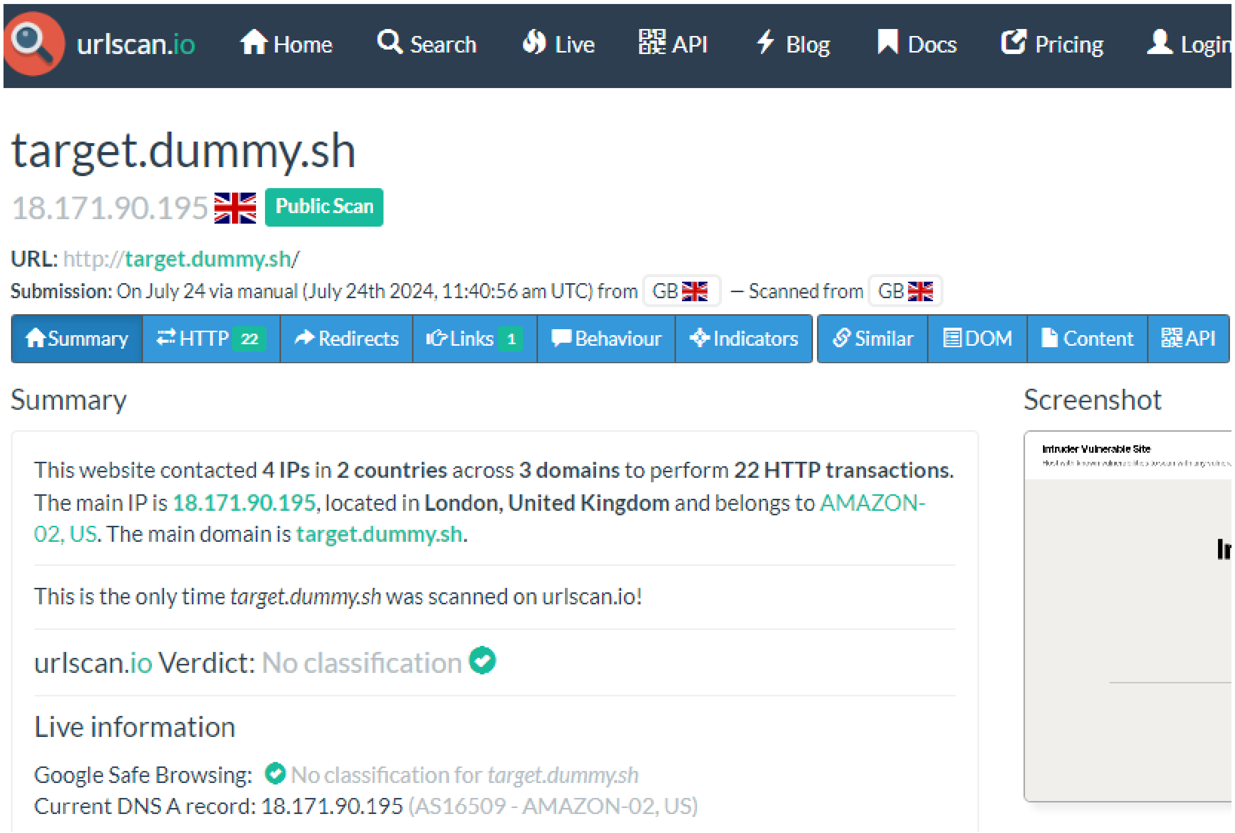
http://target.dummy.sh/

We're browsing to your website...

We're waiting for this website to finish loading. Depending on our load, this might take a minute.
You will automatically be redirected to the result, you do **not** have to refresh this page!

Attention: You have submitted this URL as a **Public Scan** which means that the results of the scan will be visible to other users of urlscan.io. If you didn't intend to make this result public please use the *Report* feature once the scan has finished!

Fig. 1.1: Loading the Website on UrlScan.io



target.dummy.sh

18.171.90.195 Public Scan

URL: http://target.dummy.sh/

Submission: On July 24 via manual (July 24th 2024, 11:40:56 am UTC) from GB — Scanned from GB

Summary

This website contacted **4 IPs** in **2 countries** across **3 domains** to perform **22 HTTP transactions.** The main IP is **18.171.90.195**, located in **London, United Kingdom** and belongs to AMAZON-02, US. The main domain is **target.dummy.sh**.

This is the only time *target.dummy.sh* was scanned on urlscan.io!

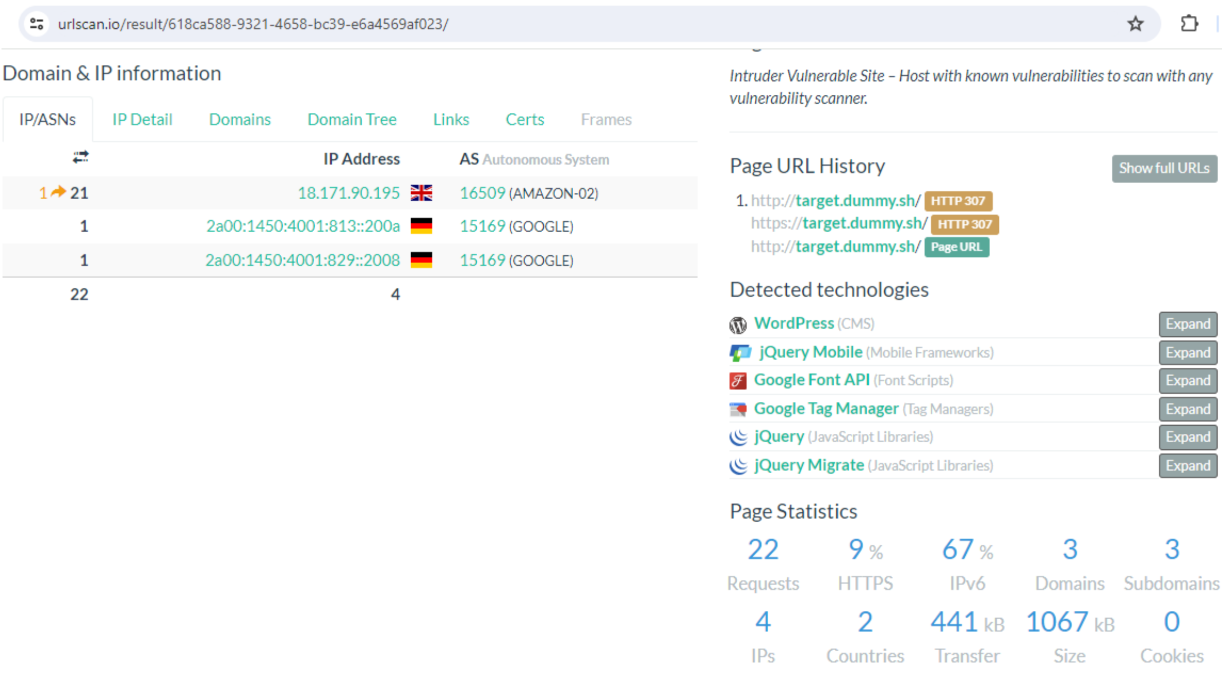
urlscan.io Verdict: No classification ✓

Live information

Google Safe Browsing: ✓ No classification for *target.dummy.sh*
Current DNS A record: 18.171.90.195 (AS16509 - AMAZON-02, US)

Screenshot

Fig. 1.2: Web Analysis of the Target Website



Domain & IP information

⇄	IP Address	AS Autonomous System
1 ↗ 21	18.171.90.195	16509 (AMAZON-02)
1	2a00:1450:4001:813::200a	15169 (GOOGLE)
1	2a00:1450:4001:829::2008	15169 (GOOGLE)
22	4	

Intruder Vulnerable Site – Host with known vulnerabilities to scan with any vulnerability scanner.

Page URL History

1. http://target.dummy.sh/ HTTP 307
https://target.dummy.sh/ HTTP 307
http://target.dummy.sh/ Page URL

Detected technologies

- WordPress (CMS)
- jQuery Mobile (Mobile Frameworks)
- Google Font API (Font Scripts)
- Google Tag Manager (Tag Managers)
- jQuery (JavaScript Libraries)
- jQuery Migrate (JavaScript Libraries)

Page Statistics

22	9 %	67 %	3	3
Requests	HTTPS	IPv6	Domains	Subdomains
4	2	441 kB	1067 kB	0
IPs	Countries	Transfer	Size	Cookies

Fig. 1.3: URL History and Further Analysis

The results of the urlscan.io scan for the domain "target.dummy.sh".

1. Target Information:
    - URL scanned: http://target.dummy.sh/
    - IP address: 18.171.90.195
    - Location: London, United Kingdom
    - Hosting provider: AMAZON-02, US
2. Scan Details:
    - Conducted on July 24th, 2024, at 11:40:56 am UTC
    - Originated from and scanned from GB (Great Britain)
    - This is the first and only time this domain has been scanned on urlscan.io
3. Scan Results:
    - The website contacted 4 IPs in 2 countries across 3 domains
    - 22 HTTP transactions were performed during the scan
    - The scan found 1 link on the website
4. Security Verdict:
    - urlscan.io Verdict: No classification (indicated by a green checkmark)
    - Google Safe Browsing: No classification for target.dummy.sh
5. Technical Details:
    - Current DNS A record: 18.171.90.195 (AS16509 - AMAZON-02, US)
6. Scan Options:
    - The scan was labeled as a "Public Scan"
    - Various analysis tabs are available: Summary, HTTP, Redirects, Links, Behaviour, Indicators, Similar, DOM, Content, and API

Observations:

1. The scanned website appears to be a test or dummy site, given its domain name (target.dummy.sh).
2. The site is hosted on Amazon Web Services (AWS) infrastructure, despite showing a UK IP location.
3. The scan didn't flag any security issues, suggesting the site is likely benign.
4. The limited number of IPs, countries, and domains contacted indicates a relatively simple website structure.
5. This being the first scan of the domain on urlscan.io could mean it's a new site or one that hasn't attracted much attention before.

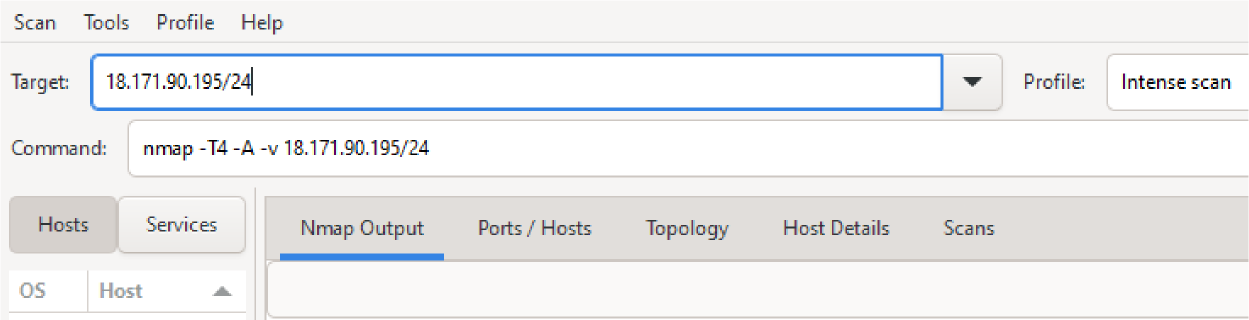
Overall, this scan provides a security and structural overview of a seemingly non-malicious website, likely used for testing or development purposes, hosted on AWS infrastructure.

In today's digital landscape, websites play a crucial role in our interactions with businesses, information, and entertainment. However, lurking beneath the surface of a seemingly innocuous website can be hidden dangers like malware, phishing attempts, and vulnerabilities. This is where web platform analysis comes into play, and urlscan.io emerges as a powerful tool for dissecting and understanding websites. Imagine encountering a suspicious URL. Clicking on it could expose you to malware or lead you to a phishing site designed to steal your information. urlscan.io acts as a safe haven. You can submit the URL to urlscan.io, and it will analyze the website without you ever having to visit it. This safe analysis creates a comprehensive snapshot of the website, including its content, structure, and behavior. This snapshot allows you to assess potential threats and make informed decisions about whether to interact with the website. Beyond identifying malicious content, urlscan.io provides a detailed breakdown of a website's functionality. It analyzes factors like the presence of scripts, forms, cookies, and other elements that can be used for legitimate purposes or malicious activities. This in-depth analysis empowers security professionals, developers, and website owners to understand how a website operates and identify any potential vulnerabilities that could be exploited by attackers. Urlscan.io goes beyond just analyzing websites. It actively scans for malicious content and checks the website's reputation against known blacklists. This real-time threat detection allows you to identify and mitigate potential security risks before they can cause harm. For businesses, this proactive approach can prevent data breaches, financial losses, and reputational damage.

Urlscan.io provides a valuable historical record of website snapshots. This allows you to track changes made to a website over time and identify any suspicious modifications. For instance, if a website was previously legitimate and then suddenly starts exhibiting malicious behavior, historical snapshots can help identify the point of change and facilitate a faster response. Security is a collaborative effort. urlscan.io allows you to share website scans with colleagues and security teams, facilitating communication and coordinated responses to potential threats. This promotes information sharing within organizations and across the broader security community. Urlscan.io offers a comprehensive suite of tools for web platform analysis. Its ability to safely analyze unknown URLs, provide deep insights into website functionality, and actively detect threats makes it a valuable asset for anyone concerned with online security. Whether you're a security professional, developer, or simply an internet user who wants to stay safe online, urlscan.io empowers you to navigate the web with greater confidence and awareness.

Web platform analysis and network scanning are complementary tools in the arsenal of cybersecurity professionals. While they serve distinct purposes, they often work in tandem to provide a comprehensive view of an organization's digital footprint. Network scanning involves probing systems, networks, and applications for vulnerabilities, open ports, and potential entry points for attackers. This process helps identify weaknesses in the overall network infrastructure. On the other hand, web platform analysis focuses specifically on examining individual websites and web applications for vulnerabilities, malicious code, and other security risks. The interconnection between these two disciplines is evident in several ways. Firstly, network scanning can identify exposed web servers, which are prime targets for attackers. Once identified, web platform analysis can be used to assess the security posture of these web applications in detail. Secondly, web platform analysis can uncover vulnerabilities that might not be detected by traditional network scanning tools. For instance, a web application might be vulnerable to SQL injection or cross-site scripting, which require a deeper analysis of the application's code and functionality. By combining network scanning and web platform analysis, organizations can achieve a more holistic understanding of their security posture. This comprehensive approach helps identify and address vulnerabilities before they can be exploited by malicious actors. Additionally, the findings from web platform analysis can inform network security policies. In essence, network scanning provides a broad overview of an organization's network, while web platform analysis offers a deep dive into specific web applications. By working together, these two techniques create a robust security framework that helps protect against a wide range of cyber threats.

2.4 Network Scanning Using Nmap

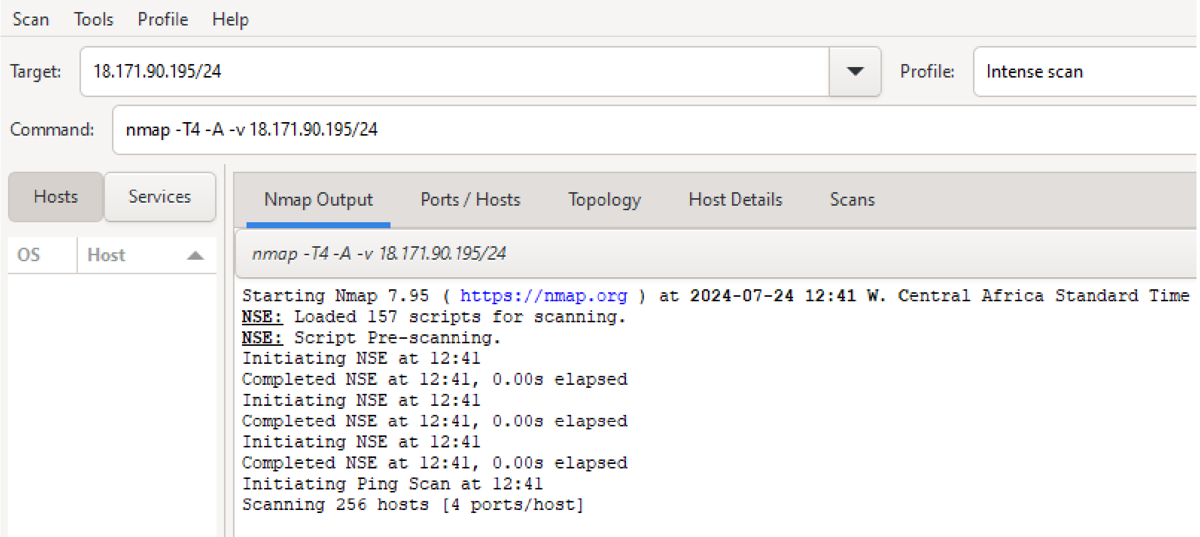


Target: 18.171.90.195/24

Profile: Intense scan

Command: nmap -T4 -A -v 18.171.90.195/24

Fig. 1.4: Nmap Scanning



Target: 18.171.90.195/24

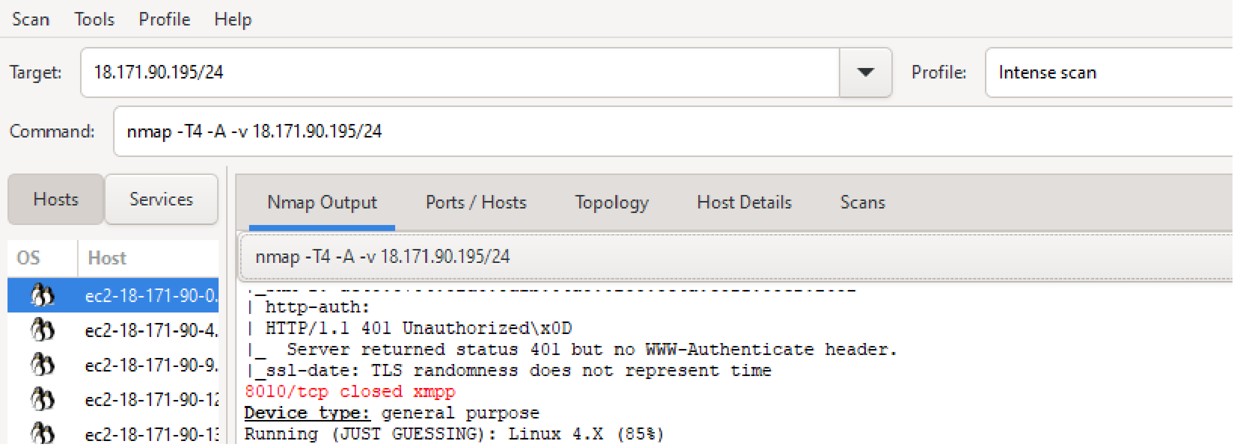
Profile: Intense scan

Command: nmap -T4 -A -v 18.171.90.195/24

nmap -T4 -A -v 18.171.90.195/24

```
Starting Nmap 7.95 ( https://nmap.org ) at 2024-07-24 12:41 W. Central Africa Standard Time
NSE: Loaded 157 scripts for scanning.
NSE: Script Pre-scanning.
Initiating NSE at 12:41
Completed NSE at 12:41, 0.00s elapsed
Initiating NSE at 12:41
Completed NSE at 12:41, 0.00s elapsed
Initiating NSE at 12:41
Completed NSE at 12:41, 0.00s elapsed
Initiating Ping Scan at 12:41
Scanning 256 hosts [4 ports/host]
```

Fig. 1.5: Network Scanning of the Target IP



Target: 18.171.90.195/24

Profile: Intense scan

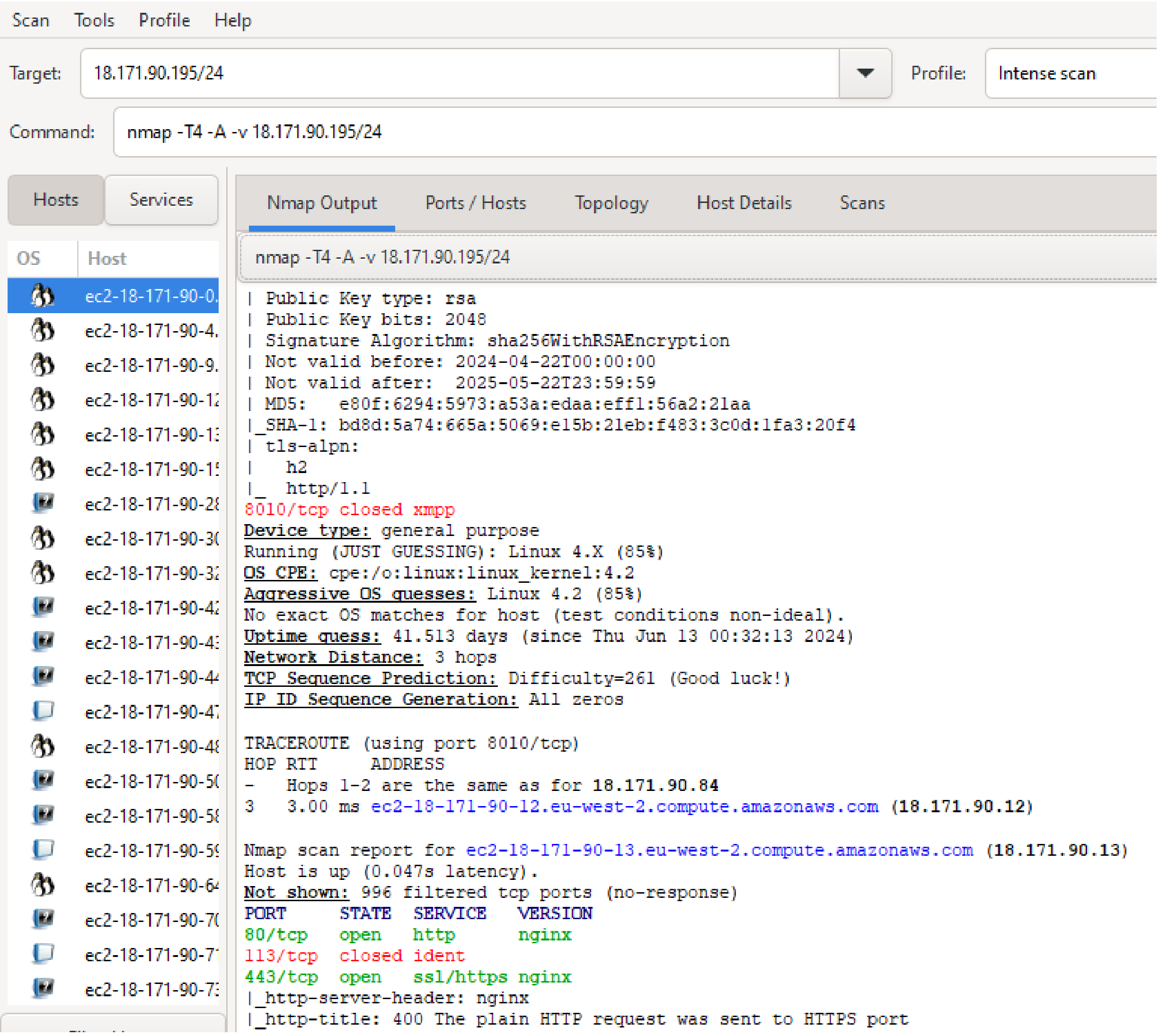
Command: nmap -T4 -A -v 18.171.90.195/24

ec2-18-171-90-0.
ec2-18-171-90-4.
ec2-18-171-90-9.
ec2-18-171-90-12
ec2-18-171-90-13

nmap -T4 -A -v 18.171.90.195/24

```
| http-auth:
| HTTP/1.1 401 Unauthorized\x0D
|_  Server returned status 401 but no WWW-Authenticate header.
|_ssl-date: TLS randomness does not represent time
8010/tcp closed xmpp
Device type: general purpose
Running (JUST GUESSING): Linux 4.X (85%)
```

Fig. 1.6: Identification of Closed Ports



Target: 18.171.90.195/24

Profile: Intense scan

Command: nmap -T4 -A -v 18.171.90.195/24

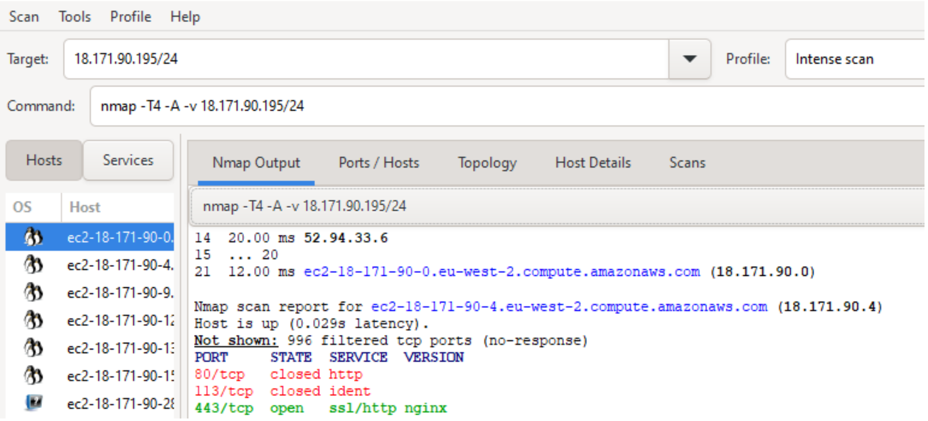
```
ec2-18-171-90-0.
ec2-18-171-90-4.
ec2-18-171-90-9.
ec2-18-171-90-12
ec2-18-171-90-13
ec2-18-171-90-15
ec2-18-171-90-28
ec2-18-171-90-30
ec2-18-171-90-32
ec2-18-171-90-42
ec2-18-171-90-43
ec2-18-171-90-44
ec2-18-171-90-47
ec2-18-171-90-48
ec2-18-171-90-50
ec2-18-171-90-58
ec2-18-171-90-59
ec2-18-171-90-64
ec2-18-171-90-70
ec2-18-171-90-71
ec2-18-171-90-73
```

```
nmap -T4 -A -v 18.171.90.195/24
| Public Key type: rsa
| Public Key bits: 2048
| Signature Algorithm: sha256WithRSAEncryption
| Not valid before: 2024-04-22T00:00:00
| Not valid after:  2025-05-22T23:59:59
| MD5:   e80f:6294:5973:a53a:edaa:eff1:56a2:21aa
|_SHA-1: bd8d:5a74:665a:5069:e15b:21eb:f483:3c0d:1fa3:20f4
| tls-alpn:
|   h2
|_  http/1.1
8010/tcp closed xmpp
Device type: general purpose
Running (JUST GUESSING): Linux 4.X (85%)
OS CPE: cpe:/o:linux:linux_kernel:4.2
Aggressive OS guesses: Linux 4.2 (85%)
No exact OS matches for host (test conditions non-ideal).
Uptime guess: 41.513 days (since Thu Jun 13 00:32:13 2024)
Network Distance: 3 hops
TCP Sequence Prediction: Difficulty=261 (Good luck!)
IP ID Sequence Generation: All zeros

TRACEROUTE (using port 8010/tcp)
HOP RTT     ADDRESS
-   Hops 1-2 are the same as for 18.171.90.84
3   3.00 ms ec2-18-171-90-12.eu-west-2.compute.amazonaws.com (18.171.90.12)

Nmap scan report for ec2-18-171-90-13.eu-west-2.compute.amazonaws.com (18.171.90.13)
Host is up (0.047s latency).
Not shown: 996 filtered tcp ports (no-response)
PORT     STATE  SERVICE   VERSION
80/tcp   open   http      nginx
113/tcp  closed ident
443/tcp  open   ssl/https nginx
|_http-server-header: nginx
|_http-title: 400 The plain HTTP request was sent to HTTPS port
```

Fig. 1.7: Identification of Closed and Open Ports



Target: 18.171.90.195/24

Profile: Intense scan

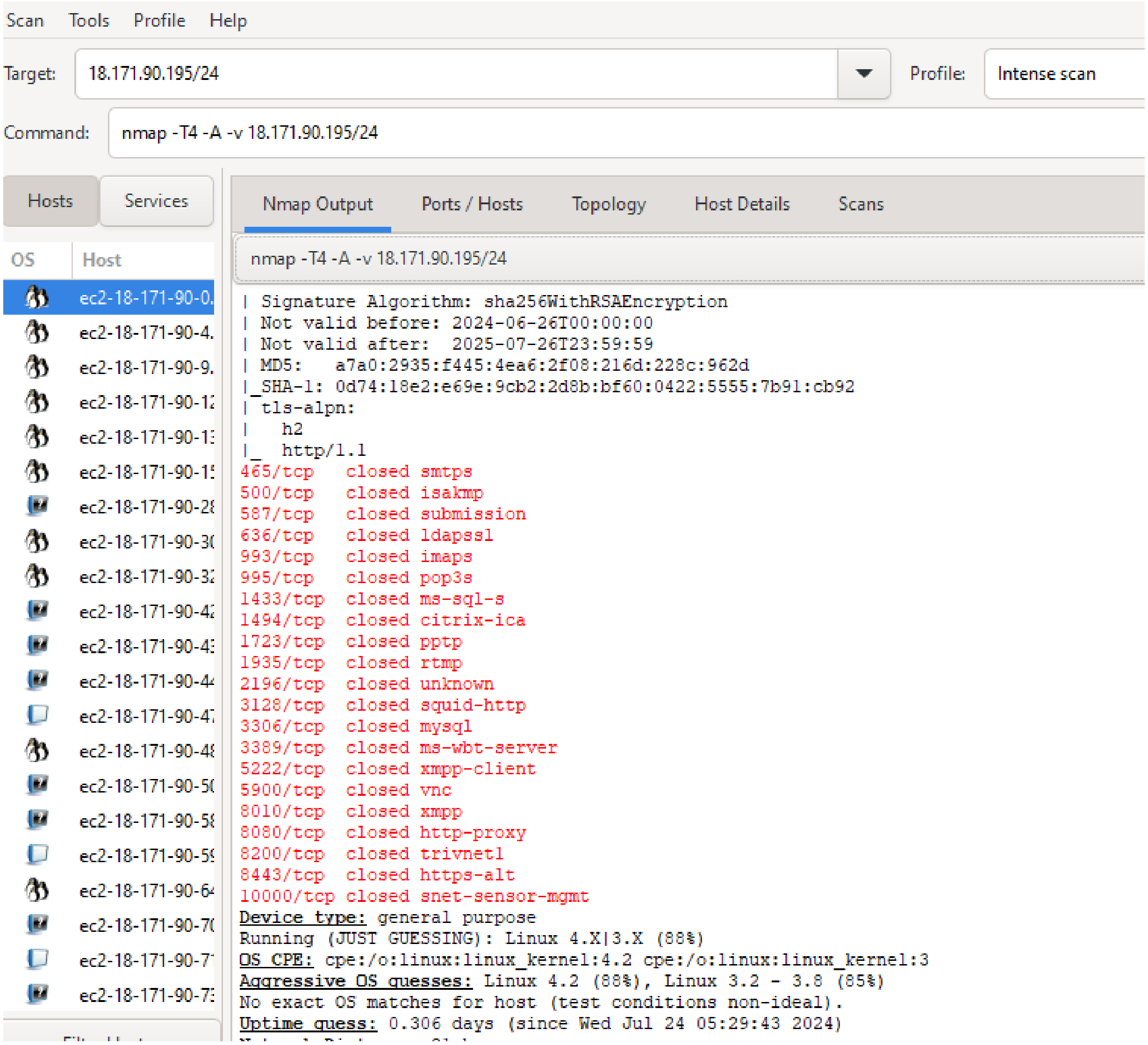
Command: nmap -T4 -A -v 18.171.90.195/24

```
ec2-18-171-90-0.
ec2-18-171-90-4.
ec2-18-171-90-9.
ec2-18-171-90-12
ec2-18-171-90-13
ec2-18-171-90-15
ec2-18-171-90-28
```

```
nmap -T4 -A -v 18.171.90.195/24
14  20.00 ms 52.94.33.6
15  ... 20
21  12.00 ms ec2-18-171-90-0.eu-west-2.compute.amazonaws.com (18.171.90.0)

Nmap scan report for ec2-18-171-90-4.eu-west-2.compute.amazonaws.com (18.171.90.4)
Host is up (0.029s latency).
Not shown: 996 filtered tcp ports (no-response)
PORT     STATE  SERVICE  VERSION
80/tcp   closed http
113/tcp  closed ident
443/tcp  open   ssl/http nginx
```

Fig. 1.8: Further Port Identification



Target: 18.171.90.195/24

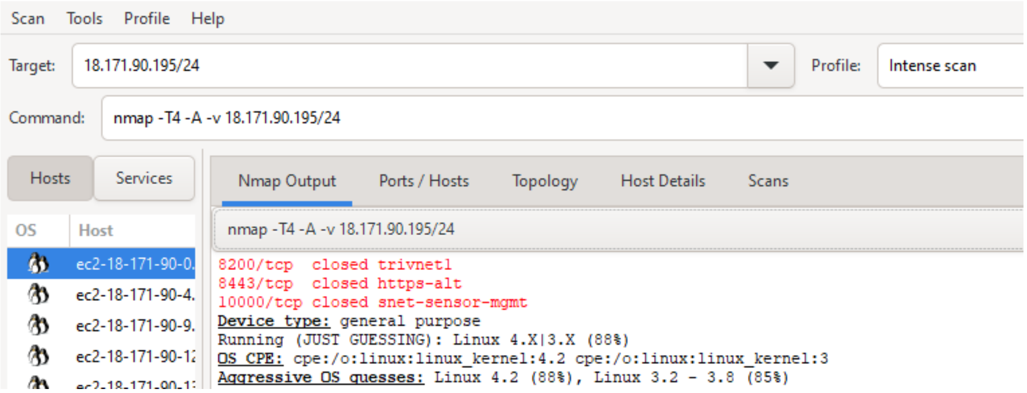
Profile: Intense scan

Command: nmap -T4 -A -v 18.171.90.195/24

```
nmap -T4 -A -v 18.171.90.195/24
```

```
| Signature Algorithm: sha256WithRSAEncryption
| Not valid before: 2024-06-26T00:00:00
| Not valid after:  2025-07-26T23:59:59
| MD5:   a7a0:2935:f445:4ea6:2f08:216d:228c:962d
|_SHA-1: 0d74:18e2:e69e:9cb2:2d8b:bf60:0422:5555:7b91:cb92
| tls-alpn:
|   h2
|_  http/1.1
465/tcp   closed smtps
500/tcp   closed isakmp
587/tcp   closed submission
636/tcp   closed ldapssl
993/tcp   closed imaps
995/tcp   closed pop3s
1433/tcp  closed ms-sql-s
1494/tcp  closed citrix-ica
1723/tcp  closed pptp
1935/tcp  closed rtmp
2196/tcp  closed unknown
3128/tcp  closed squid-http
3306/tcp  closed mysql
3389/tcp  closed ms-wbt-server
5222/tcp  closed xmpp-client
5900/tcp  closed vnc
8010/tcp  closed xmpp
8080/tcp  closed http-proxy
8200/tcp  closed trivnetl
8443/tcp  closed https-alt
10000/tcp closed snet-sensor-mgmt
Device type: general purpose
Running (JUST GUESSING): Linux 4.X|3.X (88%)
OS CPE: cpe:/o:linux:linux_kernel:4.2 cpe:/o:linux:linux_kernel:3
Aggressive OS guesses: Linux 4.2 (88%), Linux 3.2 - 3.8 (85%)
No exact OS matches for host (test conditions non-ideal).
Uptime guess: 0.306 days (since Wed Jul 24 05:29:43 2024)
```

Fig. 1.9: Identification of More Closed Ports



Target: 18.171.90.195/24

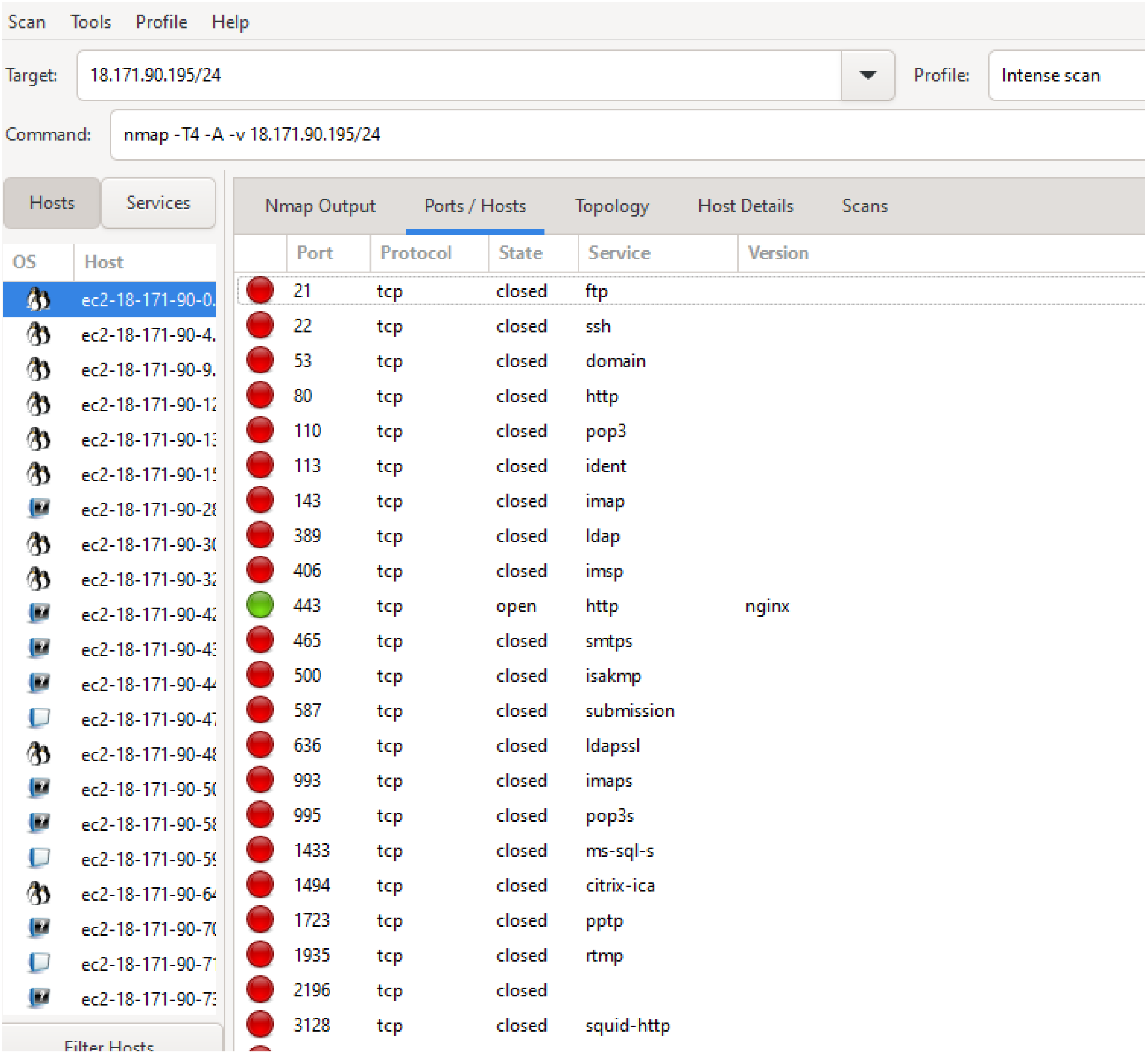
Profile: Intense scan

Command: nmap -T4 -A -v 18.171.90.195/24

```
nmap -T4 -A -v 18.171.90.195/24
```

```
8200/tcp  closed trivnetl
8443/tcp  closed https-alt
10000/tcp closed snet-sensor-mgmt
Device type: general purpose
Running (JUST GUESSING): Linux 4.X|3.X (88%)
OS CPE: cpe:/o:linux:linux_kernel:4.2 cpe:/o:linux:linux_kernel:3
Aggressive OS guesses: Linux 4.2 (88%), Linux 3.2 - 3.8 (85%)
```

Fig. 2.0: Port Identifictaion



Target: 18.171.90.195/24

Profile: Intense scan

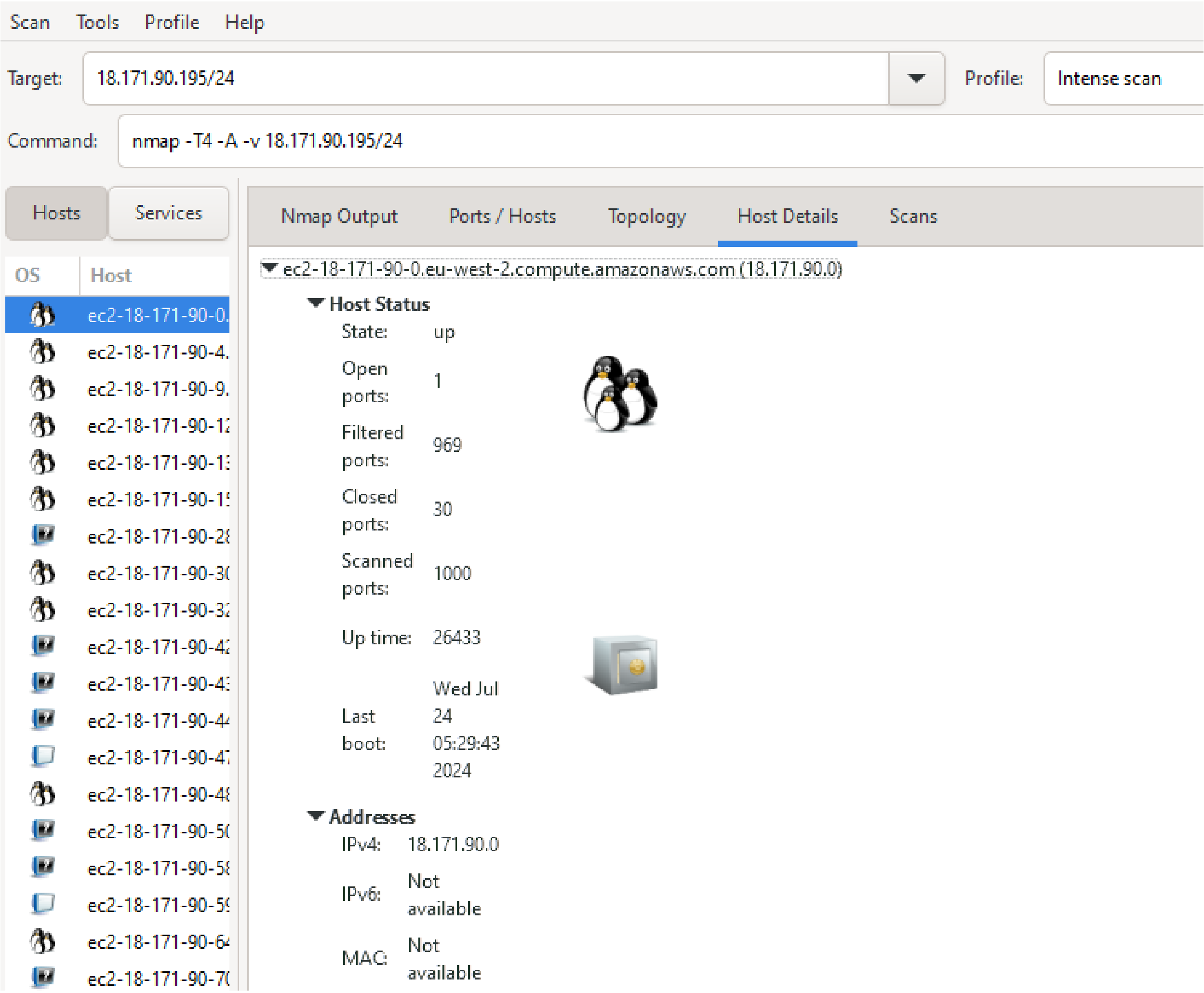
Command: nmap -T4 -A -v 18.171.90.195/24

OS	Host
	ec2-18-171-90-0.
	ec2-18-171-90-4.
	ec2-18-171-90-9.
	ec2-18-171-90-12
	ec2-18-171-90-13
	ec2-18-171-90-15
	ec2-18-171-90-28
	ec2-18-171-90-30
	ec2-18-171-90-32
	ec2-18-171-90-42
	ec2-18-171-90-43
	ec2-18-171-90-44
	ec2-18-171-90-47
	ec2-18-171-90-48
	ec2-18-171-90-50
	ec2-18-171-90-58
	ec2-18-171-90-59
	ec2-18-171-90-64
	ec2-18-171-90-70
	ec2-18-171-90-71
	ec2-18-171-90-73

	Port	Protocol	State	Service	Version
🔴	21	tcp	closed	ftp	
🔴	22	tcp	closed	ssh	
🔴	53	tcp	closed	domain	
🔴	80	tcp	closed	http	
🔴	110	tcp	closed	pop3	
🔴	113	tcp	closed	ident	
🔴	143	tcp	closed	imap	
🔴	389	tcp	closed	ldap	
🔴	406	tcp	closed	imsp	
🟢	443	tcp	open	http	nginx
🔴	465	tcp	closed	smtps	
🔴	500	tcp	closed	isakmp	
🔴	587	tcp	closed	submission	
🔴	636	tcp	closed	ldapssl	
🔴	993	tcp	closed	imaps	
🔴	995	tcp	closed	pop3s	
🔴	1433	tcp	closed	ms-sql-s	
🔴	1494	tcp	closed	citrix-ica	
🔴	1723	tcp	closed	pptp	
🔴	1935	tcp	closed	rtmp	
🔴	2196	tcp	closed		
🔴	3128	tcp	closed	squid-http	

Fig. 2.1: Port Scanning Summary

Nmap provides detailed information about ports, active hosts and services running on those hosts. Ports are communication endpoints for network services. Each port is associated with a specific service or application running on a host. Nmap scans for open ports to identify which services are available and which may be vulnerable. Hosts refer to devices or systems on the network that Nmap discovers during the scan. Each host is identified by its IP address and may have a hostname if it is properly resolved. Services are the applications or daemons running on the open ports. Nmap attempts to identify these services by performing banner grabbing or version detection. By analyzing Nmap scan results, the network security posture, potential vulnerabilities and steps to mitigate risks effectively can be understood.



Target: 18.171.90.195/24

Profile: Intense scan

Command: nmap -T4 -A -v 18.171.90.195/24

OS	Host
	ec2-18-171-90-0.
	ec2-18-171-90-4.
	ec2-18-171-90-9.
	ec2-18-171-90-1
	ec2-18-171-90-1
	ec2-18-171-90-1
	ec2-18-171-90-2
	ec2-18-171-90-3
	ec2-18-171-90-3
	ec2-18-171-90-4
	ec2-18-171-90-4
	ec2-18-171-90-4
	ec2-18-171-90-4
	ec2-18-171-90-4
	ec2-18-171-90-5
	ec2-18-171-90-5
	ec2-18-171-90-5
	ec2-18-171-90-6
	ec2-18-171-90-7

ec2-18-171-90-0.eu-west-2.compute.amazonaws.com (18.171.90.0)

Host Status

State: up

Open ports: 1

Filtered ports: 969

Closed ports: 30

Scanned ports: 1000

Up time: 26433

Last boot: Wed Jul 24 05:29:43 2024

Addresses

IPv4: 18.171.90.0

IPv6: Not available

MAC: Not available

Fig. 2.2: Nmap Scan Summary

Key Steps in the Scan:

1. **Script Pre-scanning**:
   - Loaded 157 NSE (Nmap Scripting Engine) scripts for scanning.
   - Conducted preliminary script scans, but these completed with zero elapsed time, indicating no significant pre-scan activities.
2. **Ping Scan**:
   - Scanned 256 hosts for their availability by sending ping requests.
   - This phase completed in 3.69 seconds.

- DNS resolution was performed for 72 hosts in parallel, completing in 2.52 seconds.

3. **Host Discovery**:
    - The majority of the hosts in the 18.171.90.x IP range were reported as "host down," meaning they did not respond to the ping requests.
    - Specific IP addresses such as 18.171.90.1, 18.171.90.2, and so on up to 18.171.90.229 were tested, with most being unresponsive.
4. **Port Scanning (SYN Stealth Scan)**:
    - Targeted 64 hosts to check for open ports.
    - The scan discovered open ports on multiple hosts:
        - **Port 443 (HTTPS)**: Found open on numerous hosts (e.g., 18.171.90.0, 18.171.90.4, 18.171.90.12, etc.).
        - **Port 22 (SSH)**: Found open on host 18.171.90.32.
        - **Port 80 (HTTP)**: Found open on an unspecified IP in the scanned range (the PDF content was truncated here).

Observations:

1. **Host Availability**:
    - A significant number of hosts were reported as down, which could indicate:
        - The hosts are truly offline or unresponsive to ping requests.
        - ICMP (ping) traffic might be blocked by firewalls or network policies.
2. **Common Open Ports**:
    - **Port 443 (HTTPS)** is widely open, suggesting these hosts are likely running web servers or services over SSL/TLS.
    - **Port 22 (SSH)**, found open on one host, indicates it might be running an SSH server, typically used for secure remote login.

- **Port 80 (HTTP)**, commonly used for web servers, was also detected as open on at least one host.

Results

- **Successful Discoveries**:
    - The scan successfully identified several hosts with open ports, mainly port 443.
    - This information can be used for further analysis, such as identifying the services running on these ports or conducting security assessments.
- **Recommendations**:
    - Investigate the hosts with open ports to ensure they are properly secured, especially those with ports 443 and 22 open.
    - Check firewall settings to confirm if the "host down" responses are expected behavior.
    - Ensure that critical services exposed to the internet are adequately protected against vulnerabilities.

The Nmap scan provided a snapshot of the network's visible and responsive hosts, highlighting key open ports and potential areas for further security review. This information is crucial for network administrators and security professionals in managing and securing their networks.

Network scanning is a critical component of cybersecurity, serving as a foundational tool for identifying vulnerabilities and potential threats within an organization's IT infrastructure. It involves systematically examining systems, networks, and applications for open ports, vulnerabilities, and misconfigurations. By conducting regular network scans, organizations can proactively identify and address potential security weaknesses before they are exploited by malicious actors. This proactive approach is essential in preventing data breaches, financial losses, and reputational damage. Network scanning helps organizations comply with industry regulations and standards, such as PCI DSS and HIPAA, which often require regular vulnerability assessments. Furthermore, network scanning is instrumental in maintaining the overall health and performance of a network. By identifying unused or unnecessary services, organizations can optimize resource allocation and reduce the attack surface. Additionally, network scans can help detect unauthorized devices or systems connected to the network, which could pose a security risk. In conclusion, network scanning is an indispensable aspect of modern cybersecurity. It provides organizations with the visibility and insight needed to protect their valuable assets and maintain a strong security posture. By incorporating network scanning into a comprehensive security strategy, organizations can significantly reduce their risk of falling victim to cyberattacks.

2.4 Vulnerability Assessment Using Intruder Vulnerability Scanner

Scan Info

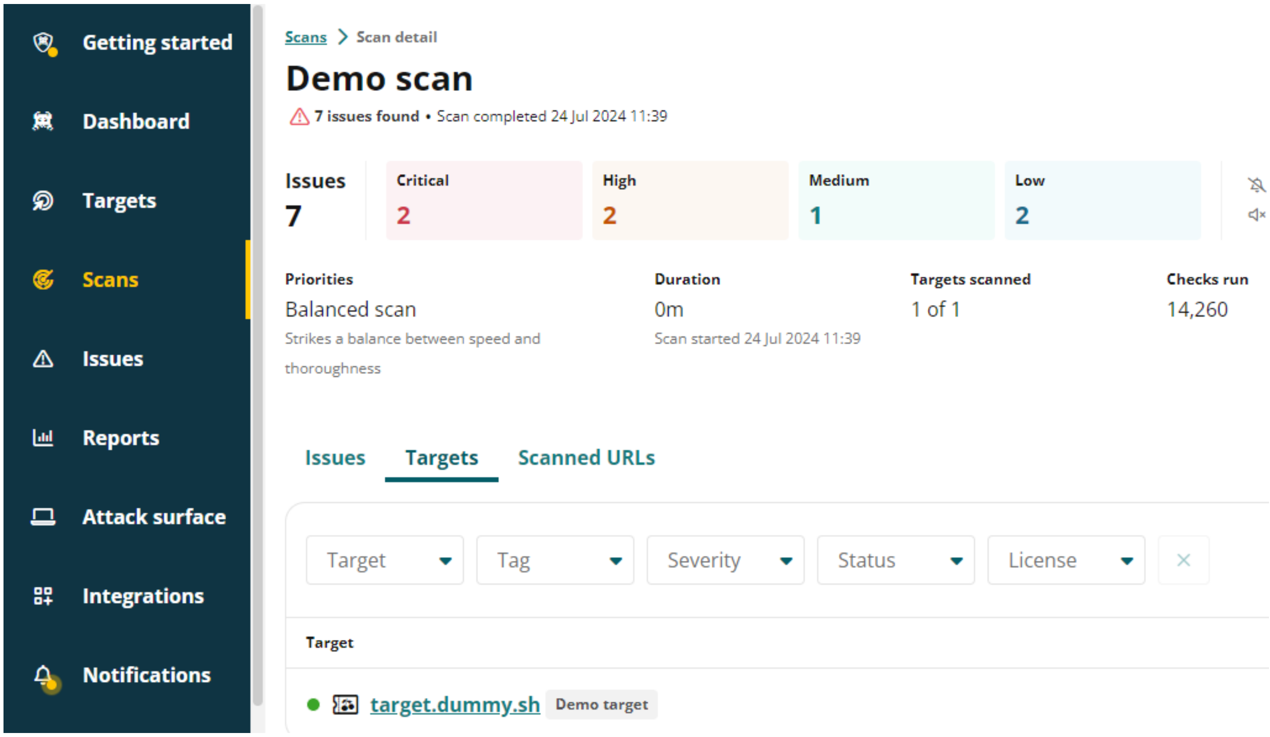
Targets included in this scan: http://target.dummy.sh/

Total Checks: 14,260

Targets: 1

Issues Discovered: 7

Noise Items: 38



Scans > Scan detail

Demo scan

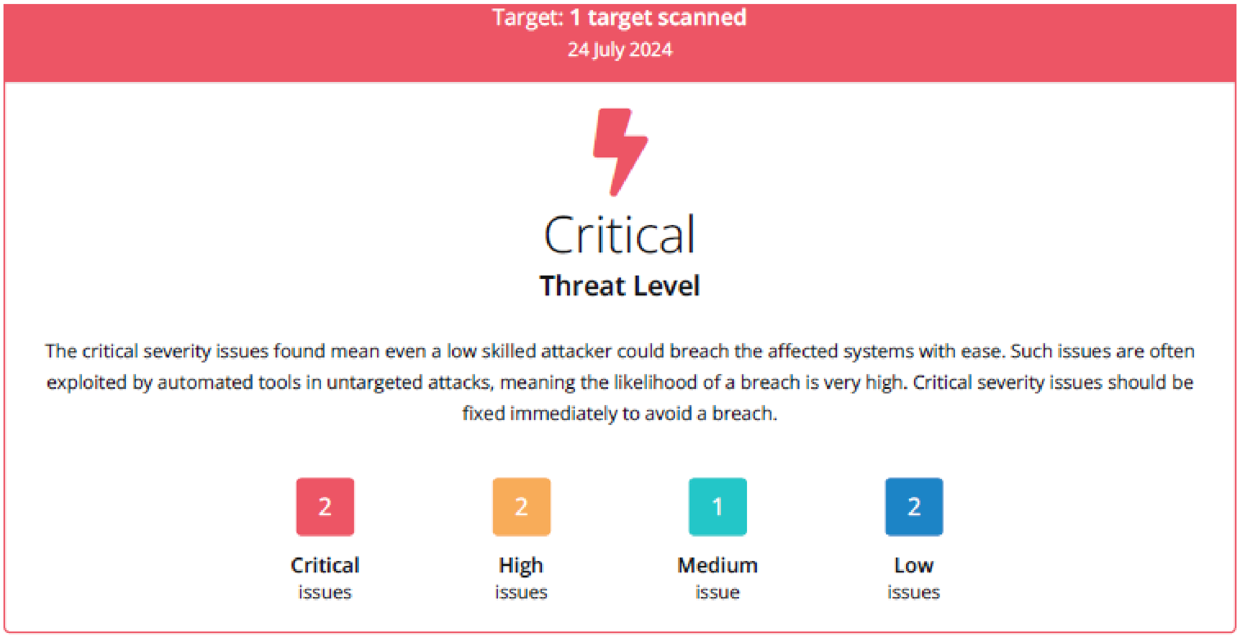
7 issues found • Scan completed 24 Jul 2024 11:39

Issues	Critical	High	Medium	Low
7	2	2	1	2

Priorities	Duration	Targets scanned	Checks run
Balanced scan Strikes a balance between speed and thoroughness	0m Scan started 24 Jul 2024 11:39	1 of 1	14,260

Target
● target.dummy.sh Demo target

Fig. 2.3: Vulnerability Assessment Using Intruder Vulnerability Scanner



Target: **1 target scanned**
24 July 2024

Critical
Threat Level

The critical severity issues found mean even a low skilled attacker could breach the affected systems with ease. Such issues are often exploited by automated tools in untargeted attacks, meaning the likelihood of a breach is very high. Critical severity issues should be fixed immediately to avoid a breach.

2	2	1	2
Critical issues	**High** issues	**Medium** issue	**Low** issues

Fig. 2.4: Severity of Threat Level

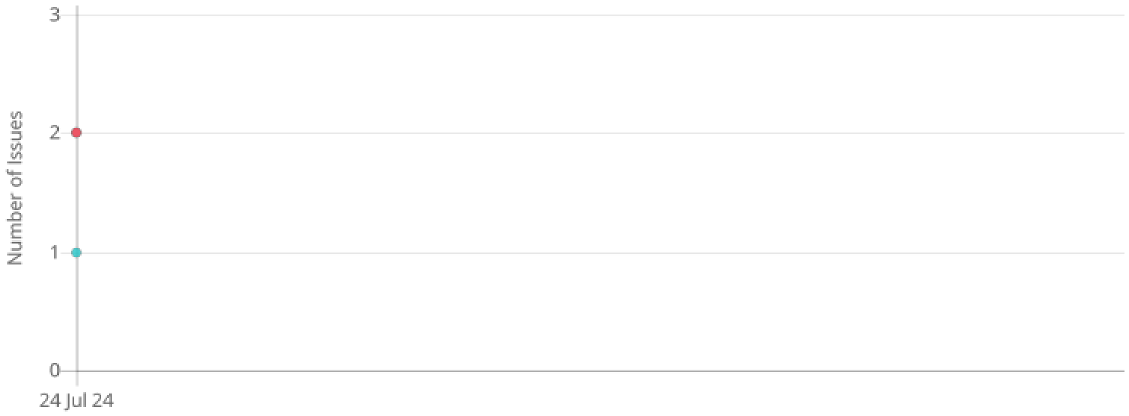
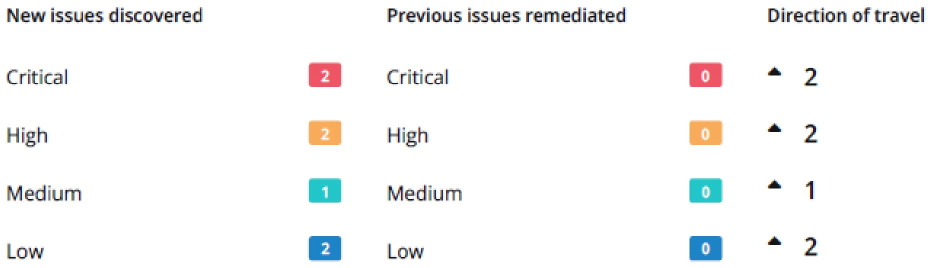


Fig. 2.5: Exposure Over Time



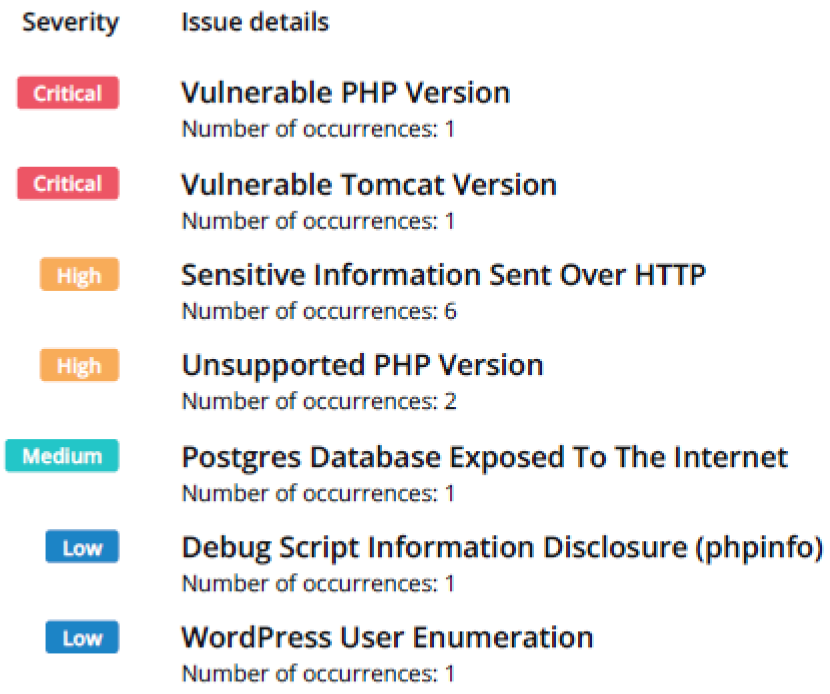
New issues discovered		Previous issues remediated		Direction of travel
Critical	2	Critical	0	▲ 2
High	2	High	0	▲ 2
Medium	1	Medium	0	▲ 1
Low	2	Low	0	▲ 2

Fig. 2.6: Differences Since Last Assessment

2.5 Vulnerability Scan Checklist

1. Vulnerable Software and Hardware
    - Web servers, e.g. Apache, Nginx.
    - Mail servers, e.g. Exim.
    - Development software, e.g. PHP.
    - Network monitoring software, e.g. Zabbix, Nagios.
    - Networking systems, e.g. Cisco ASA.
    - Content management systems, e.g. Drupal, Wordpress.
    - Other well-known weaknesses, e.g. 'Log4Shell' and 'Shellshock'.
2. Web Application Vulnerabilities
    - Checks for multiple OWASP Top Ten issues.
    - SQL injection.
    - Cross-site scripting (XSS).
    - XML external entity (XXE) injection.
    - Local/remote file inclusion.
    - Web server misconfigurations.
    - Directory/path traversal, directory listing and unintentionally exposed content.
3. Attack Surface Reduction
    - Publicly exposed databases.
    - Administrative interfaces.
    - Sensitive services, e.g. SMB.
    - Network monitoring software.
4. Information Leakage
    - Local directory path information.
    - Internal IP Addresses.
5. Encryption Weaknesses
    - 'Heartbleed', 'CRIME', 'BEAST' and 'ROBOT'.
    - Weak encryption ciphers & protocols.
    - SSL certificate misconfigurations.
    - Unencrypted services such as FTP.

6. Common Mistakes and Misconfigurations
    - VPN configuration weaknesses.
    - Exposed SVN/git repositories.
    - Unsupported operating systems.
    - Open mail relays.
    - DNS servers allowing zone transfer.



Severity	Issue details
Critical	Vulnerable PHP Version Number of occurrences: 1
Critical	Vulnerable Tomcat Version Number of occurrences: 1
High	Sensitive Information Sent Over HTTP Number of occurrences: 6
High	Unsupported PHP Version Number of occurrences: 2
Medium	Postgres Database Exposed To The Internet Number of occurrences: 1
Low	Debug Script Information Disclosure (phpinfo) Number of occurrences: 1
Low	WordPress User Enumeration Number of occurrences: 1

Fig. 2.7: Vulnerability Assessment Summary

2.6 Vulnerability Assessment and Mitigation Recommendation

1. Vulnerable PHP Version (Critical)

Description: The version of PHP in use contains a number of known security vulnerabilities which could be used to compromise the system or affect its availability. PHP is a scripting language usually used for web development. For further detail on specific vulnerabilities affecting the installed version of software, please see the scanner output and refer to: http://www.cvedetails.com/vulnerability-list/vendor_id-74/product_id-128/PHP-PHP.html

Mitigation Recommendation: Upgrade the version of PHP in use to the latest available supported version.

Occurrences	Version	First Seen
target.dummy.sh : 80 (tcp)	7.4.16	24-07-2024, 11:39:06 UTC

2. Vulnerable Tomcat Version (Critical)

Description: The version of the 'Apache Tomcat' web server in use contains a number of known security vulnerabilities which could be used to compromise the system or affect its availability. Apache Tomcat is a web server for running Java-based applications. For further information on specific vulnerabilities affecting the installed version of software, please see the scanner output. Please also see the vendor's security advisory and alerts page at: https://tomcat.apache.org/security.html

Mitigation Recommendation: Upgrade the version of Apache Tomcat in use to the latest available supported version.

Occurrences	Version	First Seen
target.dummy.sh : 8000 (tcp)	9.0.55	24-07-2024, 11:39:06 UTC

3. Sensitive Information Sent Over HTTP (High)

Description: The application does not require a secure connection (HTTPS) to be used when sending sensitive information. HTTPS is the secure extension of the HTTP protocol. For specifics on the type of information being sent in this way, see the scanner output. Passwords and other data submitted over a connection lacking HTTPS are vulnerable to capture by an attacker who is suitably positioned to view traffic sent between the user and application, in what is referred to as a Man-in-the-Middle (MitM) attack. Connections made over HTTPS are encrypted using Transport Layer Security (TLS), so that they remain confidential and unaltered, end-to-end. Without a connection using HTTPS, a MitM attacker can impersonate your application, or view and tamper with data sent between the client and the application. MitM attacks could be carried out by any malicious party located in areas such as the client's own network (e.g. using an unsecured WiFi network at a café or gym), within the client or server's internet Service Provider (ISP), or within the server's hosting infrastructure.

Mitigation Recommendation: All applications handling sensitive data should use a secured HTTPS connection to protect client-server communications. Where cookies are used for transmitting session tokens, the secure flag should be set to prevent transmission without encryption.

Occurrences	Path	First Seen
target.dummy.sh : 80 (tcp)	/wp-login.php	24-07-2024, 11:39:05 UTC
target.dummy.sh : 80 (tcp)	/wp-login.php?redirect_to=http%3A%2F%2Ftarget.dummy.sh%2Fwpadmin%2F&reauth=1	24-07-2024, 11:39:05 UTC
target.dummy.sh : 8000 (tcp)	/examples/jsp/security/protected/index.jsp	24-07-2024, 11:39:05 UTC
target.dummy.sh : 8000 (tcp)	/examples/jsp/security/protected	24-07-2024, 11:39:06 UTC
target.dummy.sh : 8000 (tcp)	/manager/html	24-07-2024, 11:39:06 UTC
target.dummy.sh : 8000 (tcp)	/manager/status	24-07-2024, 11:39:06 UTC

4. Unsupported PHP Version (High)

Description: Security updates or patches and therefore leaves the system exposed to unpatched security weaknesses. For further information, please see: http://php.net/supported-versions.php

Mitigation Recommendation: Upgrade to a supported version or uninstall the software and find a secure alternative which benefits from security patches.

Occurrences	Version	First Seen
target.dummy.sh : 80 (tcp)	7.4.16	24-07-2024, 11:39:06 UTC
target.dummy.sh : 8080 (tcp)	7.3.33	24-07-2024, 11:39:06 UTC

5. Postgres Database Exposed To The Internet (Medium)

Description: A Postgres database, which is usually intended only to be accessible on local networks (i.e. not exposed to the public) was discovered exposed to the internet. Databases are designed to be repositories for business information and should never be directly exposed to the internet. Exposing this database to the internet increases your organization’s risk in three ways:

- An attacker could attempt to use default, common or stolen credentials to login.
- An attacker could use privately owned/publicly unknown vulnerabilities to compromise the database.
- An attacker could exploit newly released vulnerabilities before you have time to patch the system.

Mitigation Recommendation: If this database does not need to be accessed over the internet, implement a firewall policy to block access from outside the internal network. If remote access is required, firewall policy should only permit access to allowed IP addresses, or access should be secured through a VPN.

Occurrences	First Seen
target.dummy.sh : 5432 (tcp)	24-07-2024, 11:39:05 UTC

6. Debug Script Information Disclosure (phpinfo) (Low)

Description: A page which makes a call to the "phpinfo()" function was found to be publicly accessible on the web server. This page discloses information about the system which may be useful to an attacker during the reconnaissance phase of an attack. Information disclosed includes OS level version and user information, web server version details and configuration information.

Mitigation Recommendation: If the page is not necessary to the operation of the system, it should be removed. If it is necessary but does not need to be exposed to the internet, then access to it should be restricted to authorized users.

Occurrences	Path	First Seen
target.dummy.sh : 8080 (tcp)	/phpinfo.php	24-07-2024, 11:39:05 UTC

7. WordPress User Enumeration (Low)

Description: The WordPress application is affected by a weakness which allows an attacker to discover valid usernames used to log into the site. This could be used by an attacker to discover valid usernames, and mount an automated password-guessing (brute force) attack against the login panel in an attempt to gain unauthorized access. These types of attack are made much easier with a user enumeration flaw, since usernames do not also have to be guessed. A successful attack may yield access to the WordPress admin console, which could allow an attacker to deface the site, change its content, or gain access to privileged information. If strong passwords have been used throughout and/or a multi-factor authentication mechanism is in place, the risk of this issue is greatly reduced. Please note that WordPress is weak to user enumeration by default, and that it cannot currently be fixed by upgrading your version of WordPress.

Mitigation Recommendation: Consider modifying the WordPress application to prevent user enumeration. This can be done by modifying the application code, but there also exist 3rd-party WordPress plugins which implement protections against this kind of attack. One popular third-party plugin which prevents user enumeration can be found at: https://engb.wordpress.org/plugins/stop-user-enumeration/.It is also recommended to switch off the default administrative user 'admin', or change its name. This account is commonly left as an active administrator, which gives attackers an easy place to start guessing passwords of a highly privileged account.

Occurrences	First Seen
target.dummy.sh : 80 (tcp)	24-07-2024, 11:39:06 UTC

3.0 ANALYSIS OF RESULTS

After the successful completion of the web vulnerability scanning and assessment, 2 critical severity issues, 2 high severity issues, 1 medium severity issue and 2 low severity issues were discovered. The critical and high severity issues found means even a low skilled attacker could breach the affected systems with ease. Such issues are often exploited by automated tools in untargeted attacks, meaning the likelihood of a breach is very high.

Critical vulnerability is a severe and potentially exploitable flaw in the network's security that poses a significant threat to the confidentiality, integrity, and availability of data and services. Immediate attention and remediation are necessary to prevent malicious actors from exploiting this vulnerability and compromising the platform. Hence, critical severity issues should be fixed immediately to avoid a breach.

High vulnerability signifies a significant security issue within the network that could potentially lead to data breaches, service disruptions, or unauthorized access. While not as severe as a critical vulnerability, it still demands prompt attention and remediation to mitigate the risk and protect the network and its users from potential security threats.

Medium vulnerability indicates a notable security concern within the network that, while not as critical as high or critical vulnerabilities, still requires attention. Addressing this issue is important to enhance the overall security posture of the network and reduce the risk of potential security incidents. It should be prioritized in the remediation process to maintain a robust security framework.

Low vulnerability represents a minor security concern in the network. While it may not pose an immediate and significant threat, it's still advisable to address it to maintain a strong security posture. This low-level vulnerability should be remediated as part of regular security maintenance to prevent potential future risks or vulnerabilities from accumulating.

Recommendation: The time between new vulnerabilities emerging and hackers exploiting them is now days, not weeks or months. For organizations who need a more mature approach to cyber security, continuous monitoring and scanning can help detect different levels of threats to systems. Also, internal systems can also be hacked with a little extra effort, e.g., by an email or web page link that exploits known unpatched software or an employee's device. An agent-based scanner can be installed on each machine to be protected.

4.0 TARGETTED VERSUS ARCHIEVED OUTPUT

In the vulnerability assessment plan, the targeted outputs included specific deliverables and outcomes. Here's a list of common targeted outputs along with what have been achieved:

1. **Comprehensive Security Assessment:** A fairly comprehensive assessment of the network security status, including an identification of vulnerabilities, security weaknesses, and a risk analysis. This output helped determine the overall security posture of the network.
2. **Vulnerability List:** A list of specific security vulnerabilities found during the assessment was recorded, accompanied by severity ratings. This output provides a clear overview of the security weaknesses that need attention.
3. **Recommendations:** Specific recommendations for mitigating or resolving identified vulnerabilities. These recommendations included code changes, configuration adjustments, and best practices to improve security.
4. **Documentation:** Detailed documentation of the audit process, including testing methodologies, tools used, and assessment findings. This documentation serves as a valuable reference and resource for stakeholders.
5. **Prioritization:** A prioritized list of vulnerabilities, with critical issues highlighted. Prioritization ensures that the organization focuses on addressing the most urgent security concerns.

5.0 CONCLUSION

Vulnerability assessment is a critical component of an organization's cybersecurity strategy, providing a systematic approach to identifying, evaluating, and addressing potential security weaknesses. By conducting regular assessments, organizations can uncover vulnerabilities within their systems, applications, and network infrastructures before they can be exploited by malicious actors. This proactive approach allows for the early detection of security gaps and helps in prioritizing remediation efforts based on the severity and potential impact of identified vulnerabilities.

A thorough vulnerability assessment involves several key steps, including reconnaissance, scanning, and analysis. Initially, security professionals gather information about the network and systems to identify potential points of entry. Scanning tools, such as Nmap and vulnerability scanners, are then employed to detect open ports, running services, and known vulnerabilities. This is followed by an in-depth analysis of the scan results to understand the nature of each vulnerability, its exploitability, and the potential consequences if exploited. The findings are then used to develop a prioritized list of vulnerabilities, guiding the remediation process effectively.

Effective vulnerability assessment not only helps in fortifying an organization's defenses but also plays a crucial role in compliance with regulatory requirements and industry standards. Many regulations, such as GDPR and PCI-DSS, mandate regular vulnerability assessments as part of a broader security framework. By adhering to these requirements, organizations can avoid costly fines and reputational damage, ensuring they meet legal and industry expectations for data protection and security. However, the process of vulnerability assessment is not a one-time task but an ongoing practice. The threat landscape is constantly evolving, with new vulnerabilities and attack vectors emerging regularly. Therefore, continuous monitoring and periodic reassessment are essential to maintain a robust security posture. Integrating vulnerability assessments with other security practices, such as penetration testing and incident response, further enhances an organization’s ability to adapt to changing threats and vulnerabilities.

Vulnerability assessment is an indispensable practice for safeguarding information systems and maintaining cybersecurity resilience. By identifying and addressing vulnerabilities before they can be exploited, organizations not only protect their assets and data but also build a stronger foundation for long-term security. As technology and threats continue to evolve, maintaining a proactive and adaptive approach to vulnerability management will remain a cornerstone of effective cybersecurity strategy.