PENETRATION TEST REPORT

Generated by KaliPenter

sophie.sarral.io
22/11/2025, 10:34 PM

1. Executive Summary

A penetration test was conducted to identify security deficiencies. The passive reconnaissance scan against sophie.sarral.io was largely unsuccessful due to missing API keys in theHarvester configuration and issues with some of the scanning tools. This resulted in very limited information gathering, indicating a need to properly configure the environment and address tool errors before conducting further reconnaissance. The WHOIS query also failed. Because the scan was largely ineffective, no significant vulnerabilities were identified. The active reconnaissance scan reveals several open ports on the target system (sophie.sarral.io). The presence of services like FTP, SSH, HTTP, RTSP, PPTP, and MySQL indicates a potentially complex system. The lack of WhatWeb and DNSRecon results limits the initial assessment, but the open ports raise concerns about outdated services, default configurations, and potential vulnerabilities. Further investigation is warranted, particularly concerning PPTP (which is deprecated), FTP (if not using secure FTP), and MySQL (regarding version and access control). The closed HTTPS port suggests that encrypted web traffic may not be available. The filtered ports could be due to firewall rules or other network configurations.

2. Scan Overview

Scan ID	Duration
scan-14	14m 32s
Total Findings	**Phases Completed**
10	2

3. Critical Findings

Missing theHarvester API Keys

INFO

TheHarvester is missing API keys for numerous services, significantly limiting its ability to gather information. This includes keys for popular services like Shodan, VirusTotal, Censys, and many others.

`Tool: Passive Recon`

WHOIS Query Failure

INFO

The WHOIS query returned a "Malformed request" error, indicating a problem with the request or the WHOIS service itself. This prevents the retrieval of valuable domain registration information.

`Tool: Passive Recon`

BuiltWith API Errors

INFO

The BuiltWith search returned a 200 status code but with an unexpected mimetype (text/json instead of application/json) and potentially an invalid key. This indicates either a problem with the BuiltWith API itself or an improperly configured API key.

`Tool: Passive Recon`

Sitedossier Captcha Triggered

INFO

The Sitedossier module triggered a captcha, indicating that the tool's requests are being flagged as potentially automated or abusive. This prevents further information gathering using this module.

`Tool: Passive Recon`

Missing API endpoint wordlist

LOW

The Harvester could not find the expected wordlist file for API endpoint scanning, leading to creation of a basic list. And then the temporary file creation also failed.

`Tool: Passive Recon`

Unsecured FTP Service

MEDIUM

The FTP service (port 21) is open. If not configured to use FTPS (FTP over SSL/TLS), all data, including credentials, is transmitted in plaintext, making it vulnerable to eavesdropping and credential theft.

`Tool: Active Recon`

Potentially Vulnerable SSH Service

MEDIUM

The SSH service (port 22) is open. While SSH is generally secure, outdated versions or weak configurations could expose the system to vulnerabilities. The lack of version information from the scan makes this a potential issue that needs further investigation.

`Tool: Active Recon`

Deprecated PPTP Service

HIGH

The PPTP service (port 1723) is open. PPTP (Point-to-Point Tunneling Protocol) is a deprecated and highly vulnerable VPN protocol. It has known security weaknesses that can be easily exploited to intercept traffic and compromise the system.

`Tool: Active Recon`

Unsecured MySQL Service

MEDIUM

The MySQL service (port 3306) is open. If not properly secured, it could allow unauthorized access to the database. This includes using strong passwords, limiting access based on IP address, and ensuring the MySQL version is up to date to patch any known vulnerabilities.

`Tool: Active Recon`

Missing HTTPS Service

LOW

The HTTPS service (port 443) is closed. This implies that the website or application might not be serving traffic over a secure channel. While HTTP is open (port 80), sensitive data should be transmitted over HTTPS to prevent eavesdropping.

`Tool: Active Recon`

4. Mitigation Strategies

1. Missing theHarvester API Keys:

Obtain and configure the necessary API keys in the `/etc/theHarvester/api-keys.yaml` file for all desired data sources. This will allow theHarvester to effectively utilize those services for information gathering.

2. WHOIS Query Failure:

Verify the WHOIS query syntax and ensure the WHOIS service is functioning correctly. Try again later, potentially using a different WHOIS lookup tool or service. Ensure the tool making the request is up-to-date.

3. BuiltWith API Errors:

Verify the BuiltWith API key is correct and that the service is functioning correctly. If the API key is valid, contact BuiltWith support to report the issue.

4. Sitedossier Captcha Triggered:

Change the IP address used for scanning, manually solve the captcha if possible, or wait before rerunning the Sitedossier module. Implement rate limiting and user agent rotation to avoid triggering captchas in the future.

5. Missing API endpoint wordlist:

Verify that the required wordlist exists at the expected path (`/usr/lib/python3/dist-packages/theHarvester/data/wordlists/api_endpoints.txt`) or reinstall the Harvester to ensure proper file integrity. Fix write permission issues for the temporary file in the directory `/usr/lib/python3/dist-packages/theHarvester/data/wordlists/`.

6. Unsecured FTP Service:

Disable the FTP service if not required. If required, enforce FTPS (FTP over SSL/TLS) and disable plaintext FTP. Configure the FTP server to use strong TLS ciphers.

7. Potentially Vulnerable SSH Service:

Ensure the SSH server is running the latest stable version. Enforce strong password policies or use public key authentication. Disable outdated and weak ciphers and MAC algorithms.

8. Deprecated PPTP Service:

Disable the PPTP service immediately. Migrate to a more secure VPN protocol like OpenVPN, IPSec, or WireGuard.

9. Unsecured MySQL Service:

Ensure the MySQL server is running the latest stable version. Implement strong authentication and authorization controls. Restrict access to the MySQL server based on IP address. Regularly review and update user privileges. Consider using a firewall to restrict access to port 3306.

10. Missing HTTPS Service:

Configure HTTPS on the web server. Obtain a valid SSL/TLS certificate and configure the web server to use it. Redirect HTTP traffic to HTTPS.