



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	RT1
Contact Name	Andrea Zerbe
Contact Title	Penetration Tester

Document History

Version	Date	Author(s)	Comments
001	04/09/2023	Andrea Zerbe	Many critical vulnerabilities

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.

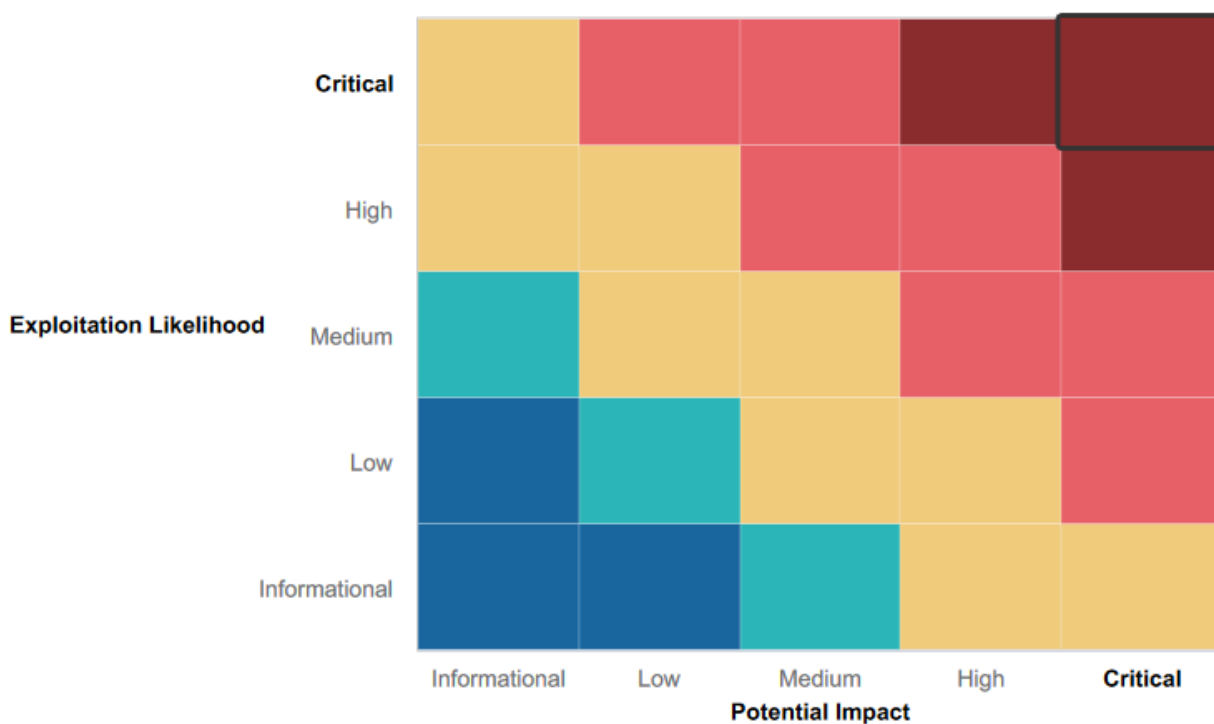
Executive Summary of Findings

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.

- Website is well laid out and easy to navigate
- FTP anonymous login makes mass file sharing easy

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.

- Sensitive information on web application and other open sources
- Web application is susceptible to many attacks
- weak passwords
- open/unfiltered/unmonitored ports
- underutilized firewalls
- simple to no input sanitization
- dated and unpatched software

Executive Summary

Day 1

To begin our penetration test we addressed common vulnerabilities on Rekall's web application. During this exercise our team found many vulnerabilities using a wide variety of methods. Some of the vulnerabilities Rekall's web application is susceptible to include; XSS injection (flags 1, 2, 3), local file inclusion (flags 5, 6), login credentials on login page (flag 8), command injection (flags 10, 11), session management (flag 14), SQL injection (flag 7), directory transversal (flag 15), brute force attacks (flag 12), sensitive data is exposure using robots.txt (flag 9), and PHP injections (flag 13).

[Day 1 Flag Images](#)

Day 2

Today we tested Rekall's network. We started our initial reconnaissance using OSINT tools. After reconnaissance we began scanning Rekall's network finding ports, ip addresses, and other publicly available information using tools such as nmap, cert sh, and nessus. Using the information uncovered in the reconnaissance and scanning phase our team then successfully performed exploitation gaining remote access to Rekall's network. After gaining remote access to Rekall's network our team performed some post exploitation using a meterpreter shell and uncovered confidential information.

[Day 2 Flag Images](#)

Day 3

Today we tested Rekall's network. We started our initial reconnaissance using OSINT tools such as Git Hub. After reconnaissance we began scanning Rekall's network finding ports, ip addresses, and other publicly available information using tools such as nmap. Using the information uncovered in the reconnaissance and scanning phase our team then successfully performed exploitation gaining remote access to Rekall's internal network. After gaining remote access to Rekall's network our team performed some post exploitation using a meterpreter shell and uncovered confidential information. Using compromised information our team the escalated privileges and accessed Rekall's domain controller.

[Day 3 Flag Images](#)

Summary Vulnerability Overview

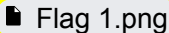
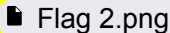
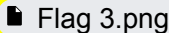
Vulnerability	Severity
XXS injection	HIGH
Local File Inclusion	HIGH
Command Injection	HIGH
Session Management	HIGH
SQL Injection	HIGH
Directory Transversal	HIGH
Brute Force Attacks	HIGH
Sensitive Data Exposure	HIGH
PHP Injections	HIGH
exploit/multi/http/tomcat_jsp_upload_bypass	CRITICAL
exploit/multi/http/apache_mod_cgi_bash_env_exec	CRITICAL
exploit/multi/http/struts2_content_type_ongl	CRITICAL
exploit/unix/webapp/drupal_resetws_unserialize	CRITICAL
Poor password policies	CRITICAL
Anonymous FTP login	MEDIUM
exploit/windows/pop3/seattlelab_pass	CRITICAL
Unnecessary scheduled tasks	MEDIUM
exploit/windows/smb/psexec	CRITICAL

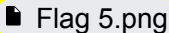
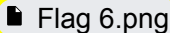
The following summary tables represent an overview of the assessment findings for this penetration test:

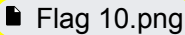
Scan Type	Total
Hosts	192.168.14.35, 192.168.13.10, 192.168.13.11, 192.168.13.12, 192.168.13.13, 192.168.13.14, 172.22.117.20, 172.22.117.10
Ports	80, 8080, 110, 445

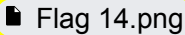
Exploitation Risk	Total
Critical	7
High	9
Medium	2
Low	0

Vulnerability Findings

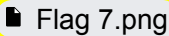
Vulnerability 1	Findings
Title	XXS Injection
Type (Web app / Linux OS / Windows OS)	Web App
Risk Rating	HIGH
Description	XXS (Cross-site scripting) injections occur when an attacker injects malicious code into a vulnerable web page, which is then executed by the victim's browser. This type of attack allows the attacker to steal sensitive information, such as login credentials or session tokens, and can also be used to hijack the victim's session or redirect them to a malicious website.
Images	  
Affected Hosts	http://192.168.14.35/
Remediation	To remediate XXS injections, it is recommended to properly sanitize user input and use encoding techniques to prevent the execution of malicious code.

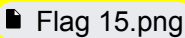
Vulnerability 2	Findings
Title	Local File Inclusion
Type (Web app / Linux OS / Windows OS)	Web App
Risk Rating	HIGH
Description	Local File Inclusion (LFI) is a type of web application vulnerability where an attacker can read or execute local files on a web server. The attacker can use LFI to view sensitive information, such as configuration files, passwords, or private data stored on the server. LFI is typically exploited by injecting specially crafted input that manipulates the web application's file inclusion mechanism.
Images	 
Affected Hosts	http://192.168.14.35/
Remediation	To remediate LFI, it is necessary to properly sanitize user input and implement strong input validation mechanisms. Additionally, it is recommended to use secure coding practices and restrict access to sensitive files by using file permissions and access controls.

Vulnerability 3	Findings
Title	Command Injection
Type (Web app / Linux OS / Windows OS)	Web App
Risk Rating	HIGH
Description	Command injection is a type of security vulnerability that allows an attacker to execute commands on a target system. It occurs when a web application accepts user input that is not properly sanitized and validated, and passes it to an underlying system command. This can allow an attacker to execute commands on the target system with the privileges of the web application, which can lead to data theft, system compromise, or even a full-scale breach.
Images	
Affected Hosts	http://192.168.14.35/
Remediation	To remediate command injection, it is essential to properly sanitize and validate user input. Additionally, implementing input validation and sanitization libraries, and restricting access to system commands can help prevent command injection attacks. It is also recommended to use least privilege access and keep systems and applications up-to-date with security patches.



Vulnerability 4	Findings
Title	Session Management
Type (Web app / Linux OS / Windows OS)	Web App
Risk Rating	HIGH
Description	Session management vulnerabilities refer to security weaknesses in the management of user sessions within a web application. This can include flaws in session authentication, session ID generation, session expiry, and session hijacking prevention. These vulnerabilities can allow an attacker to gain unauthorized access to a user's session or take control of a user's account, resulting in data theft, identity theft, or other malicious activities.
Images	
Affected Hosts	http://192.168.14.35/
Remediation	To remediate session management vulnerabilities, it is necessary to implement secure session management practices, such as using strong session ID generation algorithms, setting appropriate session timeouts, and enforcing secure session communication. Additionally, using SSL/TLS encryption to protect session data in transit, and monitoring user activities for signs of session hijacking can help prevent these vulnerabilities. It is also important to



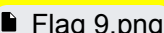
	regularly review and update session management procedures to stay current with emerging threats.
--	--

Vulnerability 5	Findings
Title	SQL Injection
Type (Web app / Linux OS / Windows OS)	Web App
Risk Rating	HIGH
Description	SQL injection is a type of cyber attack that targets web applications that use SQL databases. It occurs when an attacker injects malicious SQL commands into the application's input fields, which are then executed by the database. This can allow an attacker to view, modify, or delete data from the database or even take over the entire system.
Images	
Affected Hosts	http://192.168.14.35/
Remediation	To remediate SQL injections, it is essential to use parameterized queries or prepared statements to ensure that all user input is properly sanitized and validated. Additionally, using least privilege access for database users, implementing proper error handling, and regularly auditing databases for vulnerabilities can help prevent SQL injections. It is also recommended to use web application firewalls and keep systems and applications up-to-date with security patches.


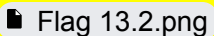

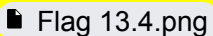
Vulnerability 6	Findings
Title	Directory Transversal
Type (Web app / Linux OS / Windows OS)	Web App
Risk Rating	HIGH
Description	Directory traversal is a web application vulnerability that allows an attacker to access files and directories outside the web server's root directory. This can occur when the application does not properly sanitize user input that includes file path information. An attacker can use directory traversal to view sensitive information, upload and execute malicious files, or even take over the entire system.
Images	
Affected Hosts	http://192.168.14.35/

Remediation	To remediate directory traversal vulnerabilities, it is necessary to implement proper input validation and sanitization mechanisms that filter out special characters and prevent the use of "../" sequences in user input. Additionally, using file access controls and least privilege access to limit the application's file system permissions, and monitoring the web application logs for suspicious activity can help prevent directory traversal attacks. It is also recommended to use web application firewalls and keep systems and applications up-to-date with security patches.
--------------------	---



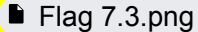
Vulnerability 7	Findings
Title	Brute Force Attacks
Type (Web app / Linux OS / Windows OS)	Web App
Risk Rating	HIGH
Description	Brute force vulnerabilities are a type of cyber attack that involves guessing passwords or keys by trying every possible combination until the correct one is found. This type of attack can be automated using software that tries multiple passwords at a very high speed. Brute force attacks can be used to gain unauthorized access to systems or accounts.
Images	 
Affected Hosts	http://192.168.14.35/
Remediation	Remediation is achieved by implementing strong passwords, rate limiting login attempts, and using multi-factor authentication.

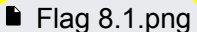
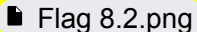
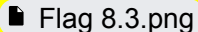
Vulnerability 8	Findings
Title	Sensitive Data Exposure
Type (Web app / Linux OS / Windows OS)	Web App
Risk Rating	HIGH
Description	Sensitive data exposure vulnerability is a security weakness that results in the unauthorized disclosure of sensitive information, such as company information, passwords, or personal identifiable information (PII). This can occur due to poor encryption practices, weak access controls, or insecure data storage methods. Attackers can exploit sensitive data exposure vulnerabilities to steal sensitive information, which can result in identity theft, financial fraud, or other malicious activities.
Images	  

Affected Hosts	http://192.168.14.35/
Remediation	To remediate sensitive data exposure through robots.txt, it is necessary to review and update the robots.txt file to ensure that sensitive directories and files are not being indexed by search engines. Additionally, implementing strong access controls and encryption mechanisms for sensitive data, and regularly auditing web servers and applications for vulnerabilities can help prevent sensitive data exposure. It is also recommended to keep systems and applications up-to-date with security patches and use web application firewalls.





Vulnerability 9	Findings
Title	PHP Injections
Type (Web app / Linux OS / Windows OS)	Web App
Risk Rating	HIGH
Description	PHP injection is a type of cyber attack that targets web applications written in the PHP scripting language. It occurs when an attacker injects malicious PHP code into the application's input fields, which is then executed by the PHP interpreter. This can allow an attacker to execute code on the server, steal sensitive information, or even take over the entire system.
Images	   
Affected Hosts	http://192.168.14.35/
Remediation	To remediate PHP injections, it is essential to use proper input validation and sanitization mechanisms to filter out special characters and prevent the injection of malicious code. Additionally, keeping the PHP interpreter up-to-date with security patches can help prevent PHP injections. It is also recommended to use web application firewalls and least privilege access for PHP users to limit the potential impact of a successful attack.

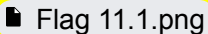


Vulnerability 10	Findings
Title	exploit/multi/http/tomcat_jsp_upload_bypass
Type (Web app / Linux OS / Windows OS)	Linux
Risk Rating	CRITICAL
Description	The exploit/multi/http/tomcat_jsp_upload_bypass is a module in the Metasploit Framework that targets a vulnerability in Apache Tomcat. This vulnerability allows an attacker to upload a malicious JSP file to the server, which can be used to execute code on the target system. The module exploits this

	vulnerability by bypassing the server's file extension and content type filters, and uploading the JSP file with a specially crafted filename.
Images	  
Affected Hosts	192.168.13.10
Remediation	To remediate exploit/multi/http/tomcat_jsp_upload_bypass update Apache Tomcat to the latest version, as the vulnerability has been patched in newer versions of the server. Rekall should consider disabling the manager application, or restricting access to it by configuring authentication and access controls. Implement additional security measures such as web application firewalls, intrusion detection systems, and access controls to prevent unauthorized access and detect any suspicious activity.

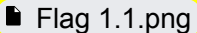
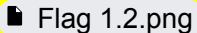
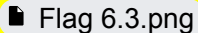
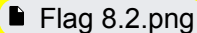
Vulnerability 11	Findings
Title	exploit/multi/http/apache_mod_cgi_bash_env_exec
Type (Web app / Linux OS / Windows OS)	Linux
Risk Rating	CRITICAL
Description	The exploit/multi/http/apache_mod_cgi_bash_env_exec is a module in the Metasploit Framework that targets a vulnerability in the Apache HTTP Server's mod_cgi module. This vulnerability allows an attacker to execute commands on the target system by exploiting a flaw in the handling of environment variables. The module exploits this vulnerability by sending a specially crafted request to the server, which includes a malicious environment variable that executes the attacker's command.
Images	  
Affected Hosts	192.168.13.11
Remediation	To remediate exploit/multi/http/apache_mod_cgi_bash_env_exec update Apache HTTP Server to the latest version, as the vulnerability has been patched in newer versions of the server. Rekall should consider disabling CGI scripts that use the shell, or using a different scripting language that does not rely on the shell. Implement additional security measures such as web application firewalls, intrusion detection systems, and access controls to prevent unauthorized access and detect any suspicious activity.



Vulnerability 12	Findings
Title	exploit/multi/http/struts2_content_type_ongl
Type (Web app / Linux OS / Windows OS)	Linux

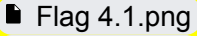
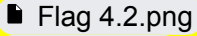
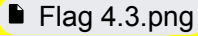
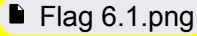
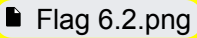
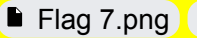
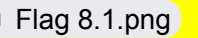
Risk Rating	CRITICAL
Description	The exploit/multi/http/struts2_content_type_onlg is a module in the Metasploit Framework that targets a vulnerability in Apache Struts 2. This vulnerability allows an attacker to execute arbitrary code on the target system by exploiting a flaw in the framework's handling of the Content-Type header. The module exploits this vulnerability by sending a specially crafted HTTP request to the server, which includes an Object-Graph Navigation Language (OGNL) expression that executes the attacker's command.
Images	   
Affected Hosts	192.168.13.12
Remediation	To remediate exploit/multi/http/struts2_content_type_onlg update Struts 2 to the latest version, as the vulnerability has been patched in newer versions of the framework. Rekall should consider implementing additional security measures such as web application firewalls, intrusion detection systems, and access controls to prevent unauthorized access and detect any suspicious activity.

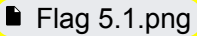
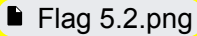
Vulnerability 13	Findings
Title	exploit/unix/webapp/drupal_resetws_unserialize
Type (Web app / Linux OS / Windows OS)	Linux
Risk Rating	CRITICAL
Description	The exploit/unix/webapp/drupal_resetws_unserialize is a module in the Metasploit Framework that targets a vulnerability in Drupal 7 and 8. This vulnerability allows an attacker to execute code on the target system by exploiting a flaw in the way that the reset password feature handles user input. The module exploits this vulnerability by sending a specially crafted HTTP request to the server, which includes a serialized PHP payload that executes the attacker's command.
Images	  
Affected Hosts	192.168.13.13
Remediation	To remediate exploit/unix/webapp/drupal_resetws_unserialize update Drupal to the latest version, as the vulnerability has been patched in newer versions of Drupal. Rekall should consider temporarily disabling the password reset functionality if you are unable to update Drupal immediately. Implement additional security measures such as web application firewalls, intrusion detection systems, and access controls to prevent unauthorized access and detect any suspicious activity.

Vulnerability 14	Findings
-------------------------	-----------------

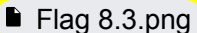
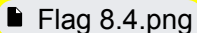
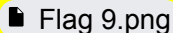
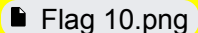
Title	Poor password policies
Type (Web app / Linux OS / Windows OS)	Web app/ Linux/ Windows
Risk Rating	CRITICAL
Description	Poor password policies refer to weak or ineffective password requirements that can leave systems and data vulnerable to unauthorized access. Examples of poor password policies include allowing users to choose easily guessable passwords (such as "password" or "123456"), not requiring regular password changes, and not enforcing minimum password length or complexity requirements.
Images	   
Affected Hosts	172.22.117.20, 172.22.117.10
Remediation	To remediate poor password policies, Rekall can implement strong password policies that require users to choose complex passwords, enforce minimum password length, and require regular password changes. Additionally, Rekall should implement multi-factor authentication mechanisms to add an extra layer of security. Regular security awareness training can also help to educate users on the importance of strong passwords and how to protect their accounts.

Vulnerability 15	Findings
Title	Anonymous FTP login
Type (Web app / Linux OS / Windows OS)	Windows
Risk Rating	MEDIUM
Description	The Anonymous FTP login vulnerability refers to the risk of unauthorized access to files or data on a server that allows anonymous FTP login. This vulnerability can be exploited by attackers who are able to connect to the server using the anonymous FTP login credentials, which typically do not require a username or password. Once connected, attackers may be able to view, download, or modify sensitive files or data, potentially causing damage or exposing sensitive information.
Images	 
Affected Hosts	172.22.117.20
Remediation	To remediate the anonymous FTP login vulnerability, Rekall should disable anonymous FTP login altogether or restrict it to only authorized users. Additionally, Rekall should implement strong password policies and enforce regular password changes to prevent unauthorized access to FTP accounts. Regular monitoring and auditing of FTP logs can also help to identify suspicious activity and prevent unauthorized access.

Vulnerability 16	Findings
Title	exploit/windows/pop3/seattlelab_pass
Type (Web app / Linux OS / Windows OS)	Windows
Risk Rating	CRITICAL
Description	The exploit/windows/pop3/seattlelab_pass is a module in the Metasploit Framework that targets a vulnerability in the Seattle Lab Mail (SLmail) server. The vulnerability allows an attacker to send a specially crafted email message to the server, which can then be used to execute code with SYSTEM privileges. Successful exploitation of this vulnerability can lead to complete compromise of the targeted system.
Images	      
Affected Hosts	172.22.117.20
Remediation	To remediate the exploit/windows/pop3/seattlelab_pass vulnerability, Rekall should apply the latest security patches and updates for the affected Seattle Lab Mail (SLmail) server. Additionally, Rekall should implement strong access controls and restrict access to the server to only authorized users. Regular monitoring and auditing of server logs can also help to identify any suspicious activity and prevent unauthorized access.

Vulnerability 17	Findings
Title	Unnecessary scheduled tasks
Type (Web app / Linux OS / Windows OS)	Windows
Risk Rating	MEDIUM
Description	The Unnecessary scheduled tasks vulnerability refers to the risk of unauthorized access to a system that has scheduled tasks running unnecessarily. Scheduled tasks are automated processes that run on a system at specified intervals, but can also be used by attackers to gain unauthorized access or execute malicious code. This vulnerability can be exploited by attackers who identify and take advantage of scheduled tasks that are not required for normal system operation, potentially leading to system compromise or data exfiltration.
Images	 
Affected Hosts	172.22.117.20

Remediation	To remediate the Unnecessary scheduled tasks vulnerability, Rekall should regularly review and remove any scheduled tasks that are not required for normal system operation. Additionally, Rekall should implement strong access controls and restrict access to scheduled tasks to only authorized users. Regular monitoring and auditing of scheduled tasks can also help to identify any suspicious activity and prevent unauthorized access.
--------------------	--

Vulnerability 18	Findings
Title	exploit/windows/smb/psexec
Type (Web app / Linux OS / Windows OS)	Windows
Risk Rating	CRITICAL
Description	The exploit/windows/smb/psexec module in the Metasploit Framework is a popular module that allows an attacker to execute commands on a remote Windows system using SMB (Server Message Block) protocol. This module takes advantage of a vulnerability in the Windows SMB service that allows an attacker to execute code with SYSTEM privileges on the target system. Successful exploitation of this vulnerability can lead to complete compromise of the targeted system.
Images	   
Affected Hosts	172.22.117.10
Remediation	To remediate the exploit/windows/smb/psexec vulnerability, Rekall should apply the latest security patches and updates for the affected Windows operating systems. Additionally, Rekall should implement strong access controls and restrict access to the SMB service to only authorized users. Regular monitoring and auditing of SMB logs can also help to identify any suspicious activity and prevent unauthorized access.