

SSS PENTESTING



SSS CORPORATION

Date: July 10th, 2024

Penetration Testing Findings Report

Business Confidential

Table of Contents

Table of Contents	2
Confidentiality Statement.....	3
Disclaimer	3
Contact Information	3
Assessment Overview	4
Assessment Components.....	4
External Penetration Test	4
Finding Severity Ratings	5
Risk Factors	5
Likelihood.....	5
Impact	5
Scope	6
Scope Exclusions	6
Client Allowances	6
Executive Summary	7
Scoping and Time Limitations.....	7
Testing Summary	7
Tester Notes and Recommendations	7
Key Strengths and Weaknesses	8
Vulnerability Summary & Report Card	9
External Penetration Test Findings	9
Findings	10
External Penetration Test Findings.....	10
Finding INT-001: Credentials found in source code (critical).....	10
Finding INT-002: Sensitive data stored in plain text (critical).....	12
Finding INT-003: Dormant accounts on the cloud (high)	13
Finding INT-004: Weak password policy (high).....	14
Finding INT-005: Hard coded credentials found in run books (medium).....	15
Finding INT-006: Access tokens available in clear text (medium).....	16

Confidentiality Statement

This document is the exclusive property of Example CORP. This document contains proprietary and confidential information. Duplication, redistribution, or use, in whole or in part, in any form, requires consent of Example CORP or SSS Pen testing.

Example CORP may share this document with auditors under non-disclosure agreements to demonstrate penetration test requirement compliance.

Disclaimer

A penetration test is considered a snapshot in time. The findings and recommendations reflect the information gathered during the assessment and not any changes or modifications made outside of that period.

Time-limited engagements do not allow for a full evaluation of all security controls. SSS Pentesting prioritized the assessment to identify the weakest security controls an attacker would exploit. SSS recommends conducting similar assessments on an annual basis by internal or third-party assessors to ensure the continued success of the controls.

Contact Information

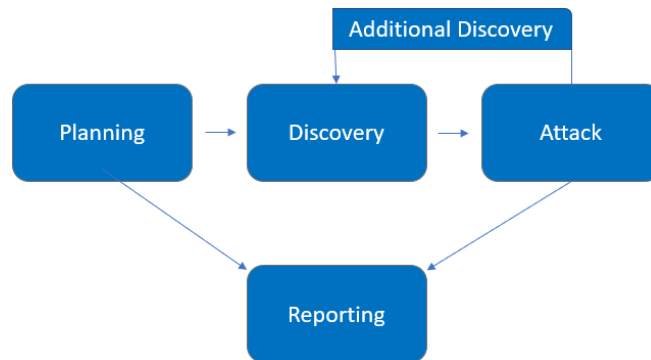
Name	Title	Contact Info
SSS Pentesting		
Sam Shepherd	Penetration Tester	sam@mail.com
Example CORP		
Bob Bobson	Chief Information Security Officer	bob@example.com

Assessment Overview

From June 12th, 2024, to July 10th, 2024, Example CORP engaged SSS to evaluate the security posture of its cloud infrastructure compared to current industry best practices regarding external cloud penetration testing.

Phases of penetration testing activities include the following:

- Planning – Customer goals are gathered, and rules of engagement obtained.
- Discovery – Perform scanning and enumeration to identify potential vulnerabilities, weak areas, and exploits.
- Attack – Confirm potential vulnerabilities through exploitation and perform additional discovery upon new access.
- Reporting – Document all found vulnerabilities and exploits, failed attempts, and company strengths and weaknesses.



Assessment Components

External Penetration Test

An external penetration test is a simulated attack on an organization's external-facing systems and assets conducted by a security professional or team to identify vulnerabilities that could be exploited by malicious actors.

Finding Severity Ratings

The following table defines levels of severity and corresponding CVSS score range that are used throughout the document to assess vulnerability and risk impact.

Severity	CVSS V4 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.
Medium	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 provided regarding items noticed during testing, strong controls, and additional documentation.

Risk Factors

Risk is measured by two factors: Likelihood and Impact:

Likelihood

Likelihood measures the potential of a vulnerability being exploited. Ratings are given based on the difficulty of the attack, the available tools, attacker skill level, and client environment.

Impact

Impact measures the potential vulnerability's effect on operations, including confidentiality, integrity, and availability of client systems and/or data, reputational harm, and financial loss.

Scope

Assessment	Details
External Penetration Test	Example.onmicrosoft.com

Scope Exclusions

Per client request, SSS did not perform any of the following attacks during testing:

- Denial of Service (DoS)
- Phishing/Social Engineering

All other attacks not specified above were permitted by Example Corporation.

Client Allowances

Example Corporation provided SSS the following allowances:

- Authorization of tests on the Azure cloud environment.

Executive Summary

SSS evaluated Example corporations' external security posture through penetration testing from June 12th, 2024, to July 10th, 2024. The following sections provide a high-level overview of vulnerabilities discovered, successful and unsuccessful attempts, and strengths and weaknesses.

Scoping and Time Limitations

Scoping during the engagement did not permit denial of service or social engineering across all testing components.

Time limitations were in place for testing. Cloud penetration testing was permitted for twenty-one (21) business days.

Testing Summary

The assessment evaluated Example CORP's cloud security. The SSS team discovered credentials found in the source code of the website (finding INT-001). This led to initial access to the cloud environment which allowed the SSS team to discover sensitive data in plain text such as credit card information (finding INT-002), access tokens (finding INT-006), and credentials stored in the run books (finding INT-005). Finally, it was found that various users were using weak passwords (finding 004) which can result in initial access, lateral movement, and/or privilege escalation in the cloud environment. For further information on the findings, please review the Technical Findings section.

Tester Notes and Recommendations

Testing results of Example CORP are indicative of an organization undergoing its first penetration test. The findings discovered are vulnerabilities within the cloud environment.

During testing, a reoccurring theme was misconfiguration of storing sensitive data. We recommend Example CORP implement various methods to secure data which would include using encryption and implementing secret management tools.

On a positive note, example CORP had strong logging and monitoring configurations.

Overall, the Example CORP network performed as expected for a first-time penetration test. We recommend that the Example CORP team thoroughly review the recommendations made in this report, correct the findings, and re-test annually to improve their overall security posture.

Key Strengths and Weaknesses

The following identifies the key strengths identified during this assessment:

1. Strong logging and monitoring configurations

The following identifies the key weaknesses identified during this assessment:

1. Source code misconfigurations
2. Credential and key management issues
3. Data protection challenges

Vulnerability Summary & Report Card

The following tables illustrate the vulnerabilities found by impact and recommended remediations:

External Penetration Test Findings

2	2	2	0	0
Critical	High	Medium	Low	Informational

Finding	Severity	Recommendation
<u>External Penetration Test</u>		
INT-001: Credentials found in source code	Critical	Disable old versions of programs that are no longer in use
INT-002: Sensitive data stored in plain text	Critical	Encrypt data at rest
INT-003: Dormant accounts on the cloud	High	Remove and disable user accounts that are no longer in user.
INT-004: Weak password policy	High	Require a mix of character types and a minimum character length of 12 characters.
INT-005: Hard coded credentials found in runbooks	Medium	Secure client secrets in other secret management solutions.
INT-006: Access tokens available in clear text	Medium	When done using Az PowerShell, use the command: Disconnect-Az Account to disconnect

Findings

External Penetration Test Findings

Finding INT-001: Credentials found in source code ([critical](#))

Description:	Through inspecting the source code on the website, old versions of Microsoft blobs were found and further enumerated which ended up containing a file that had hard coded credentials.
Risk:	Likelihood: Very High – The source code is publicly available and therefore can be enumerated and exploited by potentially anyone who visits the site. Impact: Very High - This could result in initial access to the cloud environment.
System:	Example.onmicrosoft.com
Tools Used:	Kali Linux
References:	Keep passwords out of source code – why and how by Falk Tandetzkyy NEW IT Engineering Medium

Evidence:

Figure 1.1: Shows a zip file that is accessible through the source code from an old version of microsoft blobs.

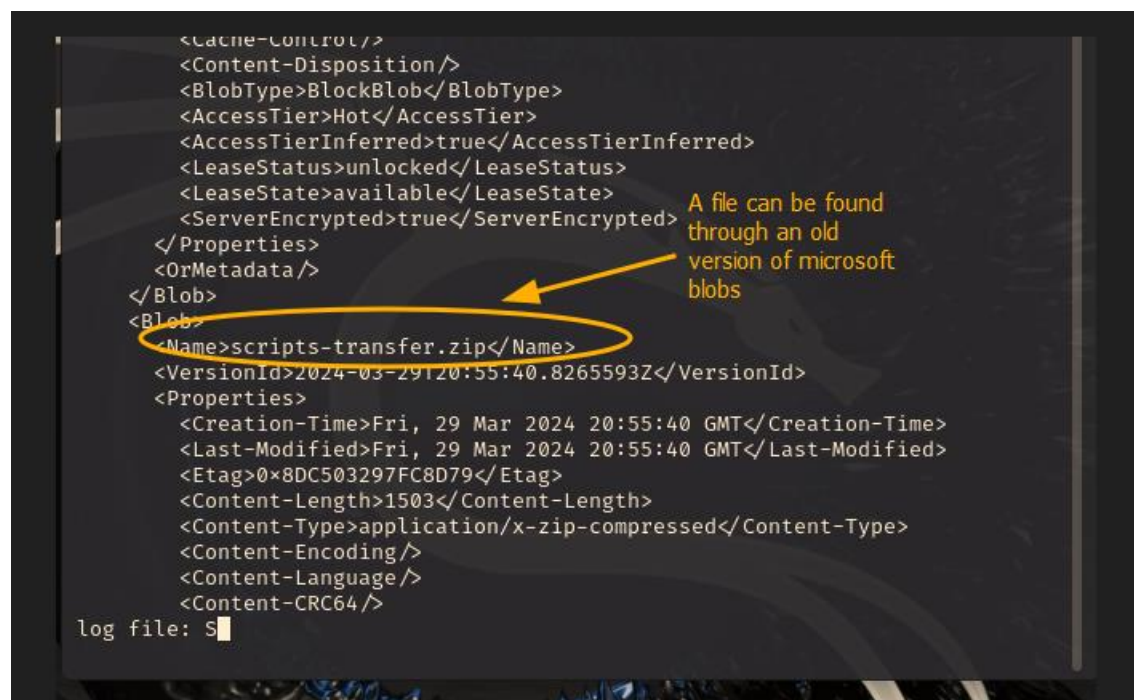
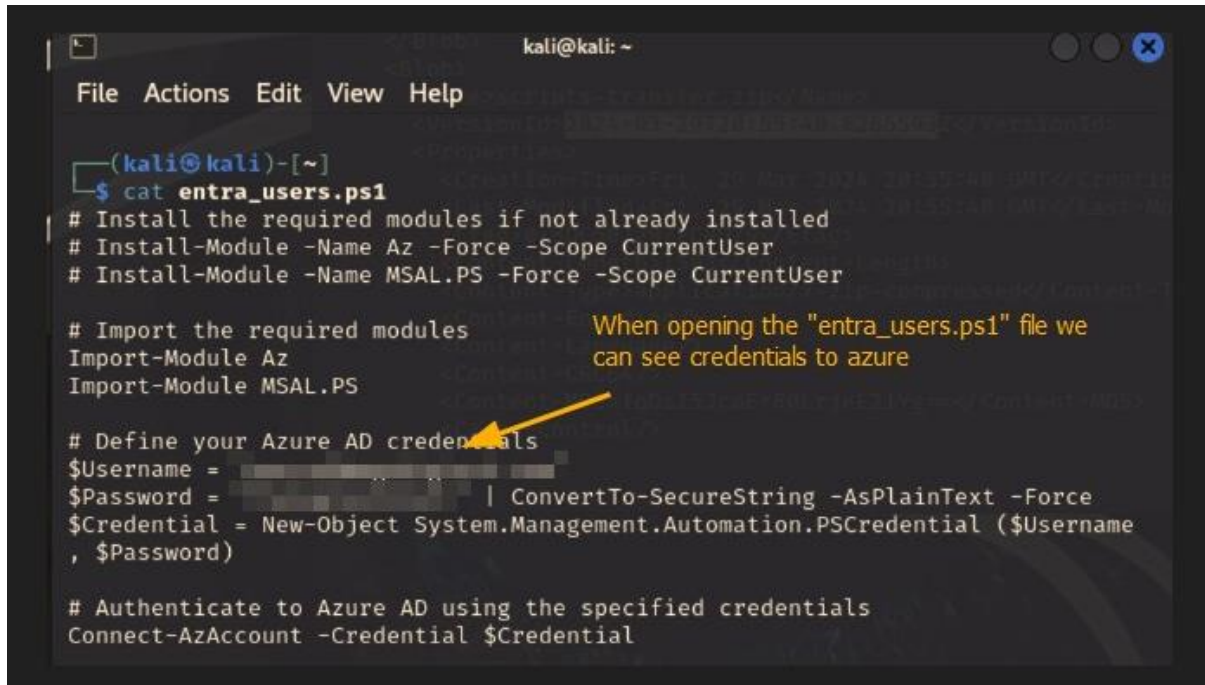


Figure 1.2: Shows credentials from unzipping the file.



```
kali@kali: ~  
File Actions Edit View Help  
  
(kali@kali)-[~]  
$ cat entra_users.ps1  
# Install the required modules if not already installed  
# Install-Module -Name Az -Force -Scope CurrentUser  
# Install-Module -Name MSAL.PS -Force -Scope CurrentUser  
  
# Import the required modules  
Import-Module Az  
Import-Module MSAL.PS  
  
# Define your Azure AD credentials  
$Username = [redacted]  
$Password = [redacted] | ConvertTo-SecureString -AsPlainText -Force  
$Credential = New-Object System.Management.Automation.PSCredential ($Username  
    , $Password)  
  
# Authenticate to Azure AD using the specified credentials  
Connect-AzAccount -Credential $Credential
```

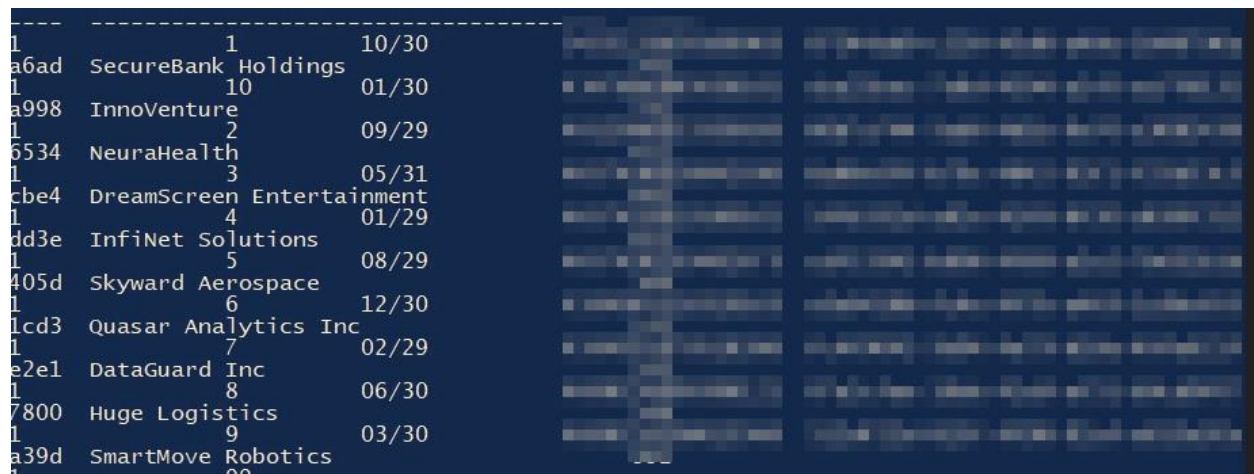
Remediation: Permanently disable old versions of programs that are no longer in use.

Finding INT-002: Sensitive data stored in plain text ([critical](#))

Description:	Data that is stored in plain text means there is no form of obfuscation to safeguard the data if it is compromised.
Risk:	Likelihood: High – Data stored in plain text can be easily exfiltrated by an attacker if an attacker is able to access the data. Impact: Very High – This can result in direct financial loss, reputational damage, and regulatory fines.
System:	Example.onmicrosoft.com
Tools Used:	Mg-graph Az Cli
References:	Top 5 PCI DSS Encryption Requirements - Sprinto

Evidence:

Figure 2.1 Shows a database of credit card numbers stored in plain text.



The screenshot displays a database table with columns for ID, Name, and Expiry Date. The data is as follows:

ID	Name	Expiry Date
1	SecureBank Holdings	10/30
a6ad	10	01/30
a998	InnoVenture	09/29
1	2	05/31
6534	NeuraHealth	01/29
1	3	08/29
cbe4	DreamScreen Entertainment	12/30
1	4	02/29
dd3e	InfiNet Solutions	06/30
1	5	03/30
405d	Skyward Aerospace	
1	6	
lcd3	Quasar Analytics Inc	
1	7	
e2e1	DataGuard Inc	
1	8	
7800	Huge Logistics	
1	9	
a39d	SmartMove Robotics	
1	00	

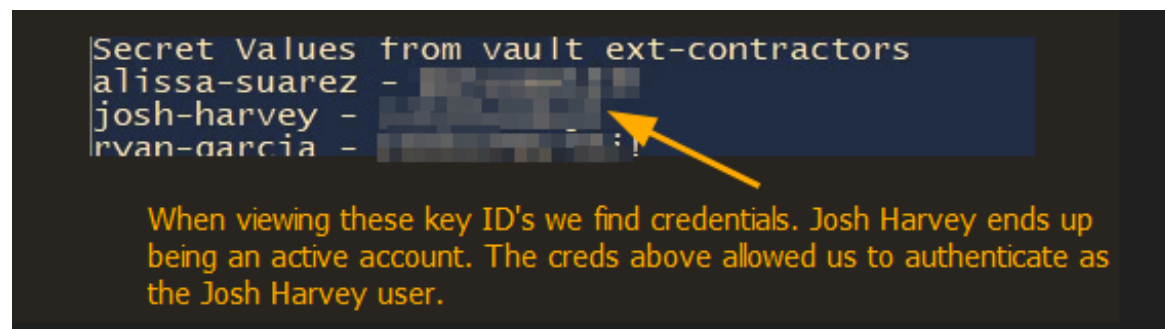
Remediation: Encrypt data at rest.

Finding INT-003: Dormant accounts on the cloud ([high](#))

Description:	Dormant accounts on the cloud can have credentials, permissions, and access to cloud resources that can be exploited by attackers
Risk:	Likelihood: High – If dormant accounts are not removed it is likely that these accounts can be exploited. Impact: High – This can result in data exfiltration, privilege escalation, and lateral movement.
System:	Example.onmicrosoft.com
Tools Used:	Mg-graph Az Cli
References:	Fix user creation and deletion issues in Microsoft Entra ID - Azure Microsoft Learn

Evidence:

Figure 3.1 Shows secrets to various active and inactive user accounts.



Remediation: Disable / de-activate accounts that are no longer in use.

Finding INT-004: Weak password policy ([high](#))

Description:	A weak password policy is when passwords are permitted that lack complexity and length requirements.
Risk:	<p>Likelihood: High – Weak passwords make it easier for attackers to perform password spray attacks and is one of the first types of attacks an attacker uses to gain initial access.</p> <p>Impact: High – Compromised accounts can permit an attacker initial access and to move laterally in the cloud environment.</p>
System:	Example.onmicrosoft.com
Tools Used:	MSOL Spray
References:	Create and use strong passwords - Microsoft Support

Evidence:

Figure 4.1:

```
[*] Now spraying Microsoft Online.
[*] Current date and time: 07/06/2024 09:27:03
PS C:\Users\User\Downloads\Tools\MSOLSpray\MSOLSpray> Invoke-MSOLSpray -UserList C:\Users\User\Downloads\Tools\0h365
serFinder-main\validemails.txt -Password ██████████ -verbose
[*] There are 9 total users to spray.
[*] Now spraying Microsoft Online.
[*] Current date and time: 07/06/2024 09:28:05
VERBOSE: POST with -1-byte payload
VERBOSE: received 3644-byte response of content type application/json; charset=utf-8
[*] SUCCESS! allensmith@samshepherd555gmail.onmicrosoft.com : ██████████
VERBOSE: POST with -1-byte payload
VERBOSE: POST with -1-byte payload
VERBOSE: POST with -1-byte payload
VERBOSE: POST with -1-byte payload
VERBOSE: POST with -1-byte payload
VERBOSE: POST with -1-byte payload
VERBOSE: POST with -1-byte payload
VERBOSE: POST with -1-byte payload
PS C:\Users\User\Downloads\Tools\MSOLSpray\MSOLSpray>
```

Cracked password using a password spray attack permits authentication to azure as the Allen Smith user

Remediation: Require a mix of character types (i.e. special characters and numbers) and a minimum character length of 12 characters. Implement multi-factor authentication to add an extra layer of security beyond passwords.

Finding INT-005: Hard coded credentials found in run books ([medium](#))

Description:	Hard coded credentials found in run books means that credentials are stored directly in scripts used to automated tasks.
Risk:	Likelihood: High – Hard-coded credentials are static and easy for attackers to extract. Impact: Medium – If these credentials are exposed, it can lead to unauthorized access and lateral movement within the cloud environment.
System:	Example.onmicrosoft.com
Tools Used:	Azure
References:	How to Prevent Hardcoded Passwords? - 0360 (offensive360.com)

Evidence:

Figure 5.1: Shows client ID and client secret available in the run book.

Hard coded credentials can be found in the runbook: "SuperRunBook2024." These credentials can be seen by clicking on "view."

```
1 # Hardcoded credentials (Replace this with Managed Service Identity)
2 $clientId = "e9e96018-99c2-453d-87ac-19c826b19103"
3 $clientSecret = "e9e96018-99c2-453d-87ac-19c826b19103"
4 $tenantId = "e9e96018-99c2-453d-87ac-19c826b19103"
5
6 #Authenticate to Azure
7 $credentials = [Microsoft.Azure.Commands.Common.Authentication.Abstractions]::GetDefaultCredentials($tenantId)
8 if ($credentials.Count -gt 0) {
9     $token = $credentials[0]
10     $token
11     Connect-AzAccount -ServicePrincipal -Credential $token
12     -Tenant $tenantId
13     -ApplicationId $clientId
14     -CertificateThumbprint $clientSecret
15 } else {
16     Write-Error "Failed to acquire token."
17     exit
18 }
```

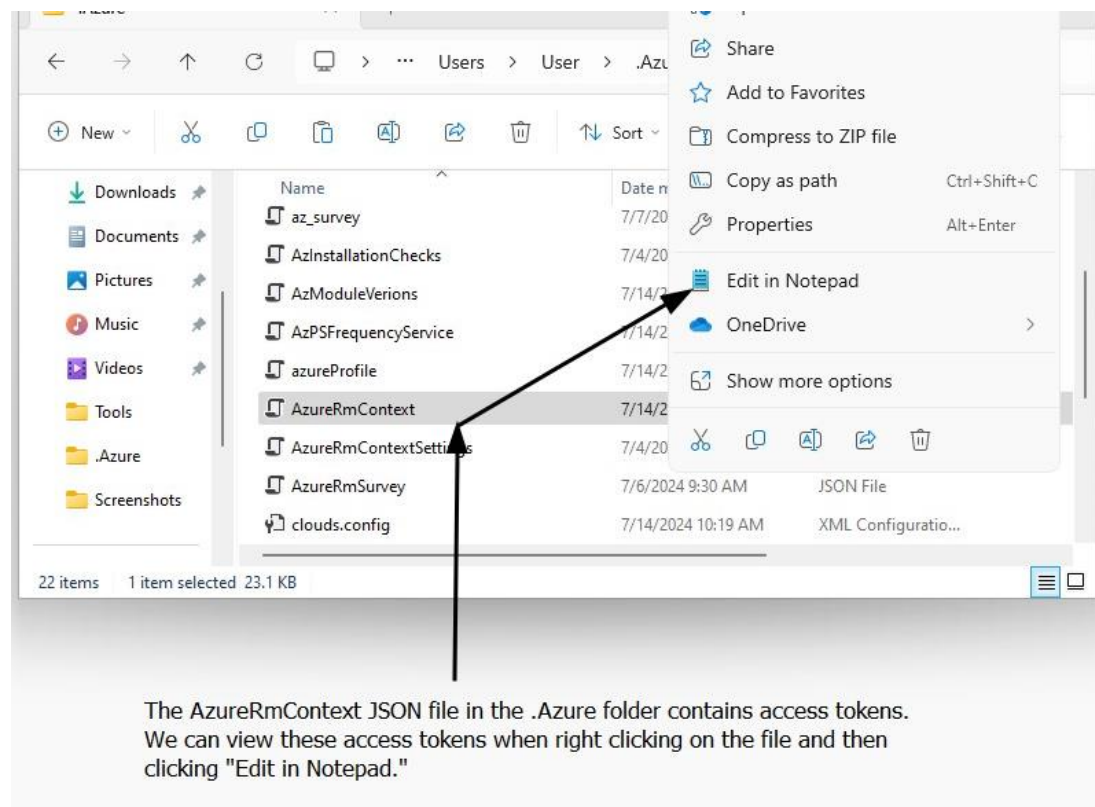
Remediation: Secure client secrets in other secret management solutions.

Finding INT-006: Access tokens available in clear text ([medium](#))

Description:	Access tokens found in clear text can permit authentication of a user and further exploitation of other resources in Entra ID.
Risk:	<p>Likelihood: Medium – Depending on whether users using azure PowerShell disconnect or not can determine if these access tokens can be exploited.</p> <p>Impact: Medium – The impact of this is going to depend on what kind of access tokens are compromised.</p>
System:	Example.onmicrosoft.com
Tools Used:	Az Power shell
References:	security - Securely storing an access token - Stack Overflow

Evidence:

Figure 6.1:



[illegible]

Page 17 of 17