HACKTHEBOX

Penetration Test

Mantis

Report of Findings

HTB Certified Penetration Testing Specialist (CPTS) Exam Report

Candidate Name: Lakshya Rastogi

Mantis Ltd.

Version: 1.0

CONFIDENTIAL

Table of Contents

1 Statement of Confidentiality

The contents of this document have been developed by Hack The Box. Hack The Box considers the contents of this document to be proprietary and business confidential information. This information is to be used only in the performance of its intended use. This document may not be released to another vendor, business partner or contractor without prior written consent from Hack The Box. Additionally, no portion of this document may be communicated, reproduced, copied or distributed without the prior consent of Hack The Box.

The contents of this document do not constitute legal advice. Hack The Box's offer of services that relate to compliance, litigation or other legal interests are not intended as legal counsel and should not be taken as such. The assessment detailed herein is against a fictional company for training and examination purposes, and the vulnerabilities in no way affect Hack The Box external or internal infrastructure.

2 Engagement Contacts

Mantis Contacts		
Contact	**Title**	**Contact Email**

Assessor Contact		
Assessor Name	**Title**	**Assessor Contact Email**
Lakshya Rastogi	Security consultant	lakshyarastogi483@gmail.com

3 Executive Summary

Mantis Ltd. (“Mantis” herein) contracted Lakshya Rastogi to perform a Network Penetration Test of Mantis’s externally facing network to identify security weaknesses, determine the impact to Mantis, document all findings in a clear and repeatable manner, and provide remediation recommendations.

3.1 Approach

Lakshya Rastogi performed testing under a “Black Box” approach from , to without credentials or any advance knowledge of Mantis’s externally facing environment with the goal of identifying unknown weaknesses. Testing was performed from a non-evasive standpoint with the goal of uncovering as many misconfigurations and vulnerabilities as possible. Testing was performed remotely from Lakshya Rastogi's assessment labs. Each weakness identified was documented and manually investigated to determine exploitation possibilities and escalation potential. Lakshya Rastogi sought to demonstrate the full impact of every vulnerability, up to and including internal domain compromise. If Lakshya Rastogi were able to gain a foothold in the internal network, Mantis as a result of external network testing, Mantis allowed for further testing including lateral movement and horizontal/vertical privilege escalation to demonstrate the impact of an internal network compromise.

3.2 Scope

The scope of this assessment was one external IP address, two internal network ranges, the mantis.htb.local domain, and any other Active Directory domains owned by Mantis discovered if internal network access were achieved.

In Scope Assets

Host/URL/IP Address	Description
10.10.10.52	mantis.htb.local

3.3 Assessment Overview and Recommendations

During the penetration test against Mantis, Lakshya Rastogi identified 3 findings that threaten the confidentiality, integrity, and availability of Mantis's information systems. The findings were categorized by severity level, with 1 of the findings being assigned a critical-risk rating, 1 high-risk, 1 medium-risk, and 0 low risk. There were also 0 informational finding related to enhancing security monitoring capabilities within the internal network.

The assessment identified a Critical risk path resulting in full Domain Compromise. The attack chain originated from an Information Disclosure vulnerability on a web server, which leaked high-privileged database credentials. Furthermore, the Domain Controller was found to be vulnerable to MS14-068 (Kerberos Checksum Validation). This unpatched vulnerability allowed a standard user to forge a Kerberos Ticket-Granting Ticket (TGT) with Domain Administrator privileges, effectively bypassing all authentication controls.

Key Recommendations

- **Patch Management:** Immediately apply Microsoft security bulletin MS14-068 to all Domain Controllers.
- **Secret Management:** Remove sensitive configuration files (secure_notes) from public web directories.
- **Database Hardening:** Encrypt sensitive user data within the OrchardCMS database.

Mantis should create a remediation plan based on the Remediation Summary section of this report, addressing all high findings as soon as possible according to the needs of the business. Mantis should also consider performing periodic vulnerability assessments if they are not already being performed. Once the issues identified in this report have been addressed, a more collaborative, in-depth Active Directory security assessment may help identify additional opportunities to harden the Active Directory environment, making it more difficult for attackers to move around the network and increasing the likelihood that Mantis will be able to detect and respond to suspicious activity.

4 Network Penetration Test Assessment Summary

Lakshya Rastogi began all testing activities from the perspective of an unauthenticated user on the internet. Mantis provided the tester with network ranges but did not provide additional information such as operating system or configuration information.

4.1 Summary of Findings

During the course of testing, Lakshya Rastogi uncovered a total of 3 findings that pose a material risk to Mantis's information systems. Lakshya Rastogi also identified 0 informational finding that, if addressed, could further strengthen Mantis's overall security posture. Informational findings are observations for areas of improvement by the organization and do not represent security vulnerabilities on their own. The below chart provides a summary of the findings by severity level.

In the course of this penetration test **1 Critical**, **1 High** and **1 Medium** vulnerabilities were identified:

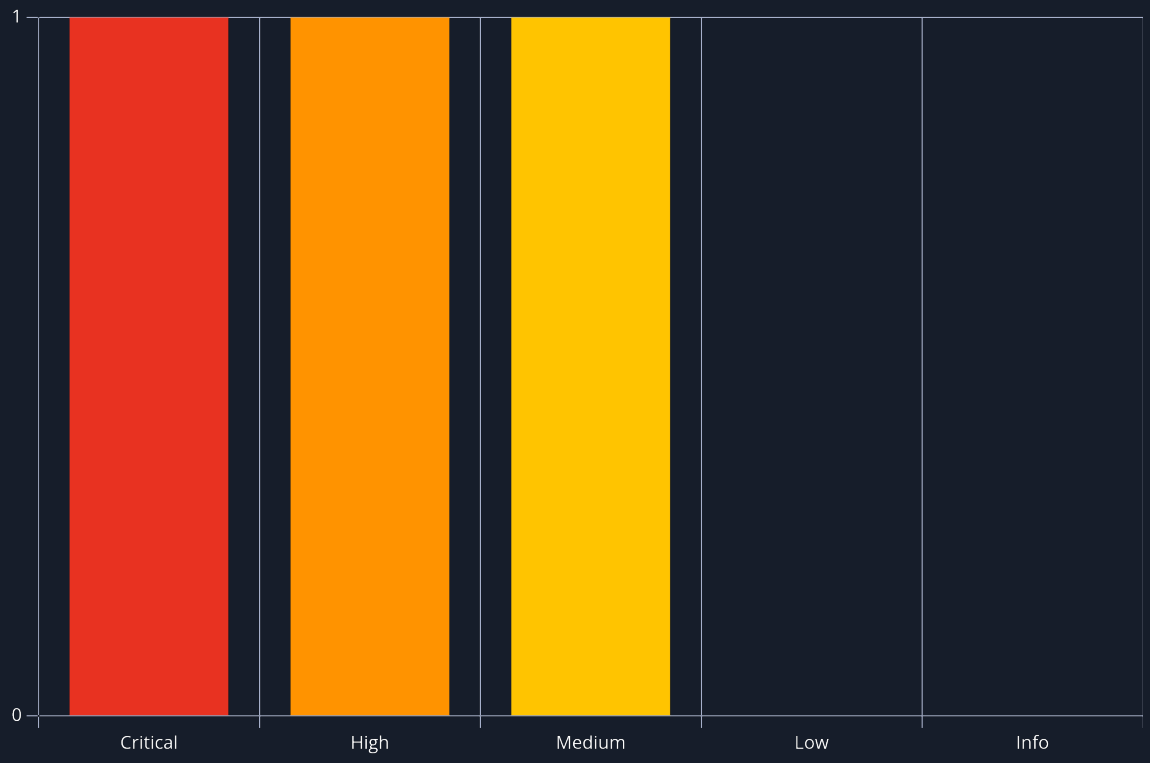


Figure 1 - Distribution of identified vulnerabilities

Below is a high-level overview of each finding identified during testing. These findings are covered in depth in the Technical Findings Details section of this report.

#	Severity Level	Finding Name	Page
1	9.9 (Critical)	Domain Privilege Escalation via Kerberos Checksum Validation (MS14-068)	14

5 Internal Network Compromise Walkthrough

During the course of the assessment Lakshya Rastogi was able gain a foothold via the external network, move laterally, and compromise the internal network, leading to full administrative control over the Mantis Active Directory domain. The steps below demonstrate the steps taken from initial access to compromise and does not include all vulnerabilities and misconfigurations discovered during the course of testing. Any issues not used as part of the path to compromise are listed as separate, standalone issues in the Technical Findings Details section, ranked by severity level. The intent of this attack chain is to demonstrate to Mantis the impact of each vulnerability shown in this report and how they fit together to demonstrate the overall risk to the client environment and help to prioritize remediation efforts (i.e., patching two flaws quickly could break up the attack chain while the company works to remediate all issues reported). While other findings shown in this report could be leveraged to gain a similar level of access, this attack chain shows the initial path of least resistance taken by the tester to achieve domain compromise.

5.1 Detailed Walkthrough

Lakshya Rastogiperformed the following to fully compromise the Mantis

Detailed Attack Narrative

Phase 1: Reconnaissance & Information Disclosure

The assessment began with a comprehensive TCP/UDP port scan of the target infrastructure (`10.10.10.52`). While standard Active Directory services were identified, the assessment team focused on a non-standard HTTP service running on **TCP port 1337**.

Directory enumeration against this web server revealed a hidden location, `/secure_notes/`, which was misconfigured to allow public access without authentication. Inside this directory, a text file with an obfuscated name was discovered. Analysis of the filename string confirmed it was encoded in Base64. Decoding this string revealed a high-entropy password (`m$$ql_S@_P@ssW0rd!`), which was hypothesized to belong to a service administrator.

Phase 2: Service Compromise & Lateral Movement

Leveraging the credentials recovered from the web server, the assessment team attempted to authenticate to the **Microsoft SQL Server (MSSQL)** instance identified on port 1433. The login was successful as the `admin` user.

Once inside the database instance, the team enumerated the application's schema to identify sensitive business data. A specific table within the `OrchardDB` database, named `blog_Orchard_Users_UserPartRecord`, was found to contain user account information. Querying this table exposed the credentials for a valid Active Directory user, `james`. This finding allowed the team to pivot from a database context to a valid domain identity.

Phase 3: Domain Privilege Escalation (MS14-068)

With access to a valid domain user account, the assessment team evaluated the Domain Controller for unpatched vulnerabilities. The system was identified as being vulnerable to **CVE-2014-6324 (MS14-068)**, a critical flaw in the Kerberos Key Distribution Center (KDC) checksum validation process.

To exploit this, the team utilized the `goldenPac.py` tool from the Impacket suite. The exploit successfully manipulated the Kerberos Privilege Attribute Certificate (PAC), allowing the user `james` to forge a valid Ticket-Granting Ticket (TGT) with "Domain Administrator" privileges.

Phase 4: Objective Completion

The forged Kerberos ticket was accepted by the Domain Controller, granting the assessment team `NT AUTHORITY\SYSTEM` level access. This level of access provided complete control over the domain infrastructure, including the ability to retrieve password hashes for all users and persist access indefinitely.

6 Remediation Summary

As a result of this assessment there are several opportunities for Mantis to strengthen its internal network security. Remediation efforts are prioritized below starting with those that will likely take the least amount of time and effort to complete. Mantis should ensure that all remediation steps and mitigating controls are carefully planned and tested to prevent any service disruptions or loss of data.

6.1 Short Term

1. Short-Term Actions (Immediate - 0 to 7 Days)

These actions address critical vulnerabilities that pose an imminent threat to the organization's security posture.

- **Apply Critical Security Updates (MS14-068):** Immediately deploy Microsoft Knowledge Base update **KB3011780** to all Domain Controllers. This patch resolves the Kerberos Checksum Validation vulnerability (CVE-2014-6324), preventing the forgery of high-privileged tickets.
- **Credential Rotation:** Force a password reset for the compromised accounts `admin` (SQL Service) and `james` (Domain User). Ensure the new passwords adhere to a complexity requirement of at least 12 characters.
- **Sanitize Web Directories:** Remove the `/secure_notes/` directory from the public-facing web server on port 1337. Implement a process to verify that no development artifacts or temporary files are deployed to production environments.

6.2 Medium Term

2. Medium-Term Actions (Tactical - 1 to 3 Months)

These actions involve configuration changes and hardening measures to reduce the attack surface.

- **Database Security Hardening:** Modify the `OrchardDB` schema to ensure user credentials in the `blog_Orchard_Users_UserPartRecord` table are not stored in cleartext. Implement strong hashing algorithms (e.g., Argon2id or bcrypt) with unique salts for each user.
- **Web Server Hardening (IIS):** Disable "Directory Browsing" on all IIS web servers to prevent attackers from enumerating file structures. Configure strict Request Filtering to block access to sensitive file extensions (`.config`, `.txt`, `.bak`).
- **Network Segmentation:** Restrict access to the Microsoft SQL Server (Port 1433) using host-based firewalls or network ACLs. Only authorized application servers should be allowed to communicate with the database; direct user access should be blocked.

6.3 Long Term

3. Long-Term Actions (Strategic - 6+ Months)

These actions focus on process improvements and architectural changes to prevent recurrence.

- **Implement Automated Patch Management:** Deploy a centralized patch management solution (e.g., WSUS or SCCM) to ensure critical security updates for the operating system and services are applied within 72 hours of release.

- **Secrets Management Solution:** Transition away from storing credentials in configuration files or databases. Implement an enterprise Secrets Management Vault (e.g., HashiCorp Vault or Azure Key Vault) to dynamically manage and inject credentials into applications at runtime.
- **SIEM Tuning & Monitoring:** Configure the Security Information and Event Management (SIEM) system to alert on Kerberos anomalies, specifically **Event ID 4769** with failure code `0xf`, which serves as a key indicator of Golden Ticket attacks.

7 Technical Findings Details

1. Domain Privilege Escalation via Kerberos Checksum Validation (MS14-068) - Critical

CWE	CWE-287 - Improper Authentication
CVSS 3.1	9.9 / CVSS:3.1/AV:N/AC:L/PR:L/UI:N/S:C/C:H/I:H/A:H
Root Cause	**Overview** Following the compromise of the domain user `james` (detailed in Finding #2), the assessment team evaluated the Domain Controller for known vulnerabilities. The target system was identified as a Windows Server 2008 R2 instance vulnerable to **CVE-2014-6324**, commonly referred to as **MS14-068**. This vulnerability exists in the Kerberos Key Distribution Center (KDC) implementation. It allows an authenticated user to manipulate the **Privilege Attribute Certificate (PAC)** within their Ticket-Granting Ticket (TGT). By exploiting this flaw, the assessment team was able to forge a valid TGT that claimed membership in high-privileged groups, such as "Domain Admins," effectively bypassing all authorization controls.
Impact	**Business Impact** • **Total Domain Compromise:** Successful exploitation grants the attacker full administrative control over the entire Active Directory forest. This includes the ability to access all files, reset any user password, and modify critical infrastructure. • **Loss of Confidentiality & Integrity:** An attacker with this level of access can exfiltrate sensitive intellectual property, install persistent backdoors (Golden Tickets), or deploy ransomware across all domain-joined machines.
Remediation	**Remediation & Mitigation** **1. Apply Security Updates (Critical)** • Immediately install **Microsoft Knowledge Base (KB) 3011780** on all Domain Controllers. This patch corrects the checksum validation process in the Kerberos KDC. • Ensure that all Domain Controllers are running a supported version of Windows Server (Server 2016/2019/2022) to receive ongoing security updates. **2. Monitor for Indicators of Compromise (IoC)** • Configure SIEM alerts for **Event ID 4769** (Kerberos Service Ticket Operations) where the failure code is `0xf` (KRB_ERR_SUM_KDC_GENERIC_ERROR), which often indicates an MS14-068 exploitation attempt. • Monitor for the creation of TGTs with unusually long lifetimes or mismatched PAC signatures.
References	-

Finding Evidence

Proof of Concept (PoC)

Step 1: Vulnerability Identification

The target operating system version and lack of specific security patches indicated potential susceptibility to the MS14-068 vulnerability. The attack requires valid domain credentials, which were previously obtained (`james` / `J@m3s_P[Redacted]`).

Step 2: Exploit Execution (Golden PAC)

The `goldenPac.py` tool from the Impacket suite was used to automate the attack chain. This tool performs the following actions:

1. Requests a TGT for the user `james` .
2. Exploits the vulnerability to rewrite the PAC, adding the user to the "Domain Admins" group.
3. Injects the forged ticket into the current session.
4. Uses the forged ticket to authenticate via SMB (PsExec) and execute a command shell.

Command Used:

```
# Syntax: goldenPac.py domain/user:password@target
python3 goldenPac.py mantis.htb.local/james:'J@m3s_P[Redacted]'@mantis.htb.local
```

Evidence:

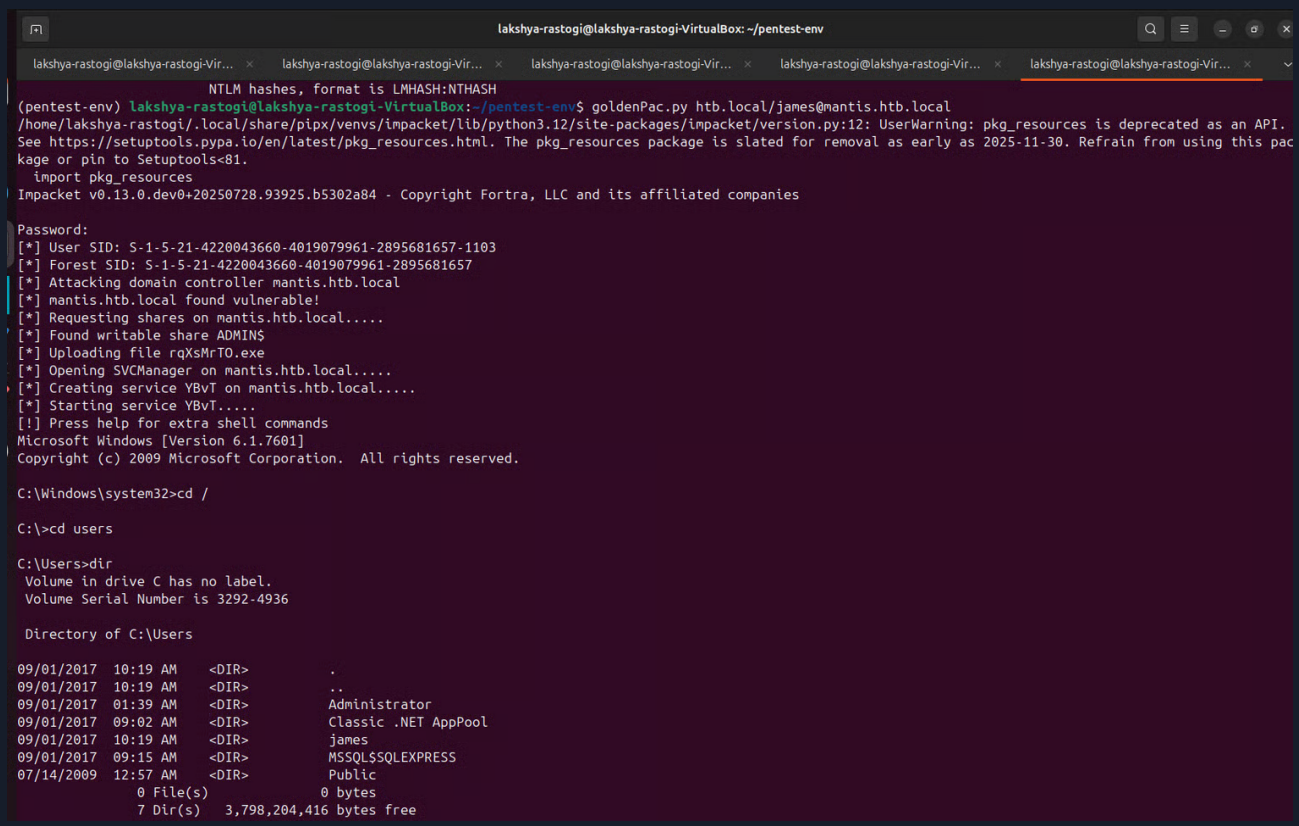


Figure 1: *Your goldenPac.py terminal output showing the "PAC" manipulation and success message*

Step 3: Verification of System Access

The exploit resulted in an interactive command shell on the Domain Controller. The `whoami` command confirmed that the session was running with `NT AUTHORITY\SYSTEM` privileges, the highest level of access on a Windows machine.

Command Used:

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

2. Sensitive Information Disclosure via Publicly Accessible Directory - High

CWE	CWE-200 - Exposure of Sensitive Information to an Unauthorized Actor
CVSS 3.1	7.5 / CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:N/A:N
Root Cause	During the enumeration phase of the web application running on port 1337, the assessment team identified a directory named `/secure_notes/` that was accessible without authentication. This directory contained a file with a non-standard, Base64-encoded filename (`dev_notes_NmQy...txt`). Upon analysis, it was determined that the filename itself concealed sensitive credentials. Decoding the string revealed the cleartext password for the `admin` user account. This credential reuse allowed for direct, unauthorized authentication to the backend Microsoft SQL Server (OrchardDB), bypassing the intended application login flow.
Impact	**Business Impact** The exposure of administrative credentials in a publicly accessible web directory represents a **critical breach of confidentiality**. • **Database Compromise:** Malicious actors can use these credentials to connect directly to the corporate database, allowing them to read, modify, or delete sensitive business data (such as the `blog_Orchard_Users` table). • **Lateral Movement:** As demonstrated later in the assessment, these database privileges provided a pathway to execute system-level commands on the underlying server, escalating the breach from a simple information leak to a full system compromise. • **Reputational Damage:** The storage of cleartext passwords in web-accessible directories indicates a failure in secure development lifecycle (SDLC) practices, potentially leading to compliance violations.
Remediation	TODO REMEDIATION
References	-

Finding Evidence

Here is the **Technical Details & Proof of Concept** section for your SysReptor report.

I have structured this exactly how a professional pentest report requires it: **Step-by-Step Reproduction**. I have left clear `[INSERT SCREENSHOT]` placeholders where you should paste the images you shared with me.

Technical Details

Vulnerability Location:

- **Service:** HTTP (IIS)
- **Port:** 1337
- **URL:** `http://mantis.htb.local:1337/secure_notes/`

- **Vector:** Unsecured Directory Listing & Information Leakage

Step 1: Directory Enumeration

The `ffuf` utility was used to fuzz the web application for hidden directories. The scan identified the `secure_notes` directory, which returned a `301 redirect` status code.

Command Used:

```
ffuf -u http://10.10.10.52:1337/FUZZ -w /snap/seclists/current/Discovery/Web-Content/
directory-list-2.3-big.txt -mc 200,300-399 -t 50
```

Evidence:

```
<SNIP>
/secure_notes (Status: 301)
<SNIP>
```

Step 2: File Identification

Navigating to the `/secure_notes/` directory revealed a file with an obfuscated filename. The file contents themselves contained binary data, but the filename appeared to be a Base64 encoded string.

File URL: `http://10.10.10.52:1337/secure_notes/dev_notes_NmQyNDI0NzE2YzVvNTM0ODczNzUzMDc2NzNlMzI2NDIx.txt.txt`

Evidence:

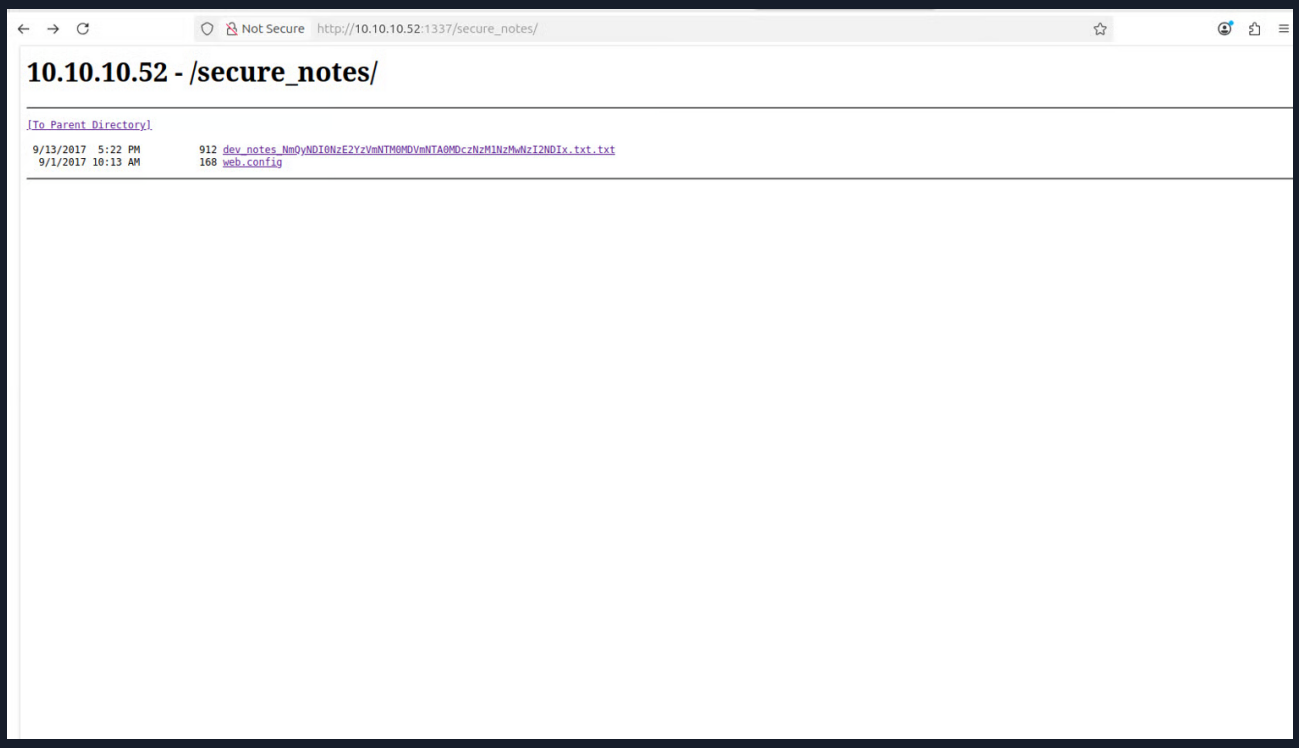


Figure 2: *Browser or curl output showing the directory listing with the long filename*

Step 3: Decoding and Credential Extraction

The filename string was isolated and decoded. The decoding process required two steps:

1. **Base64 Decode:** Resulted in a Hexadecimal string.
2. **Hex to ASCII Conversion:** Resulted in the cleartext password.

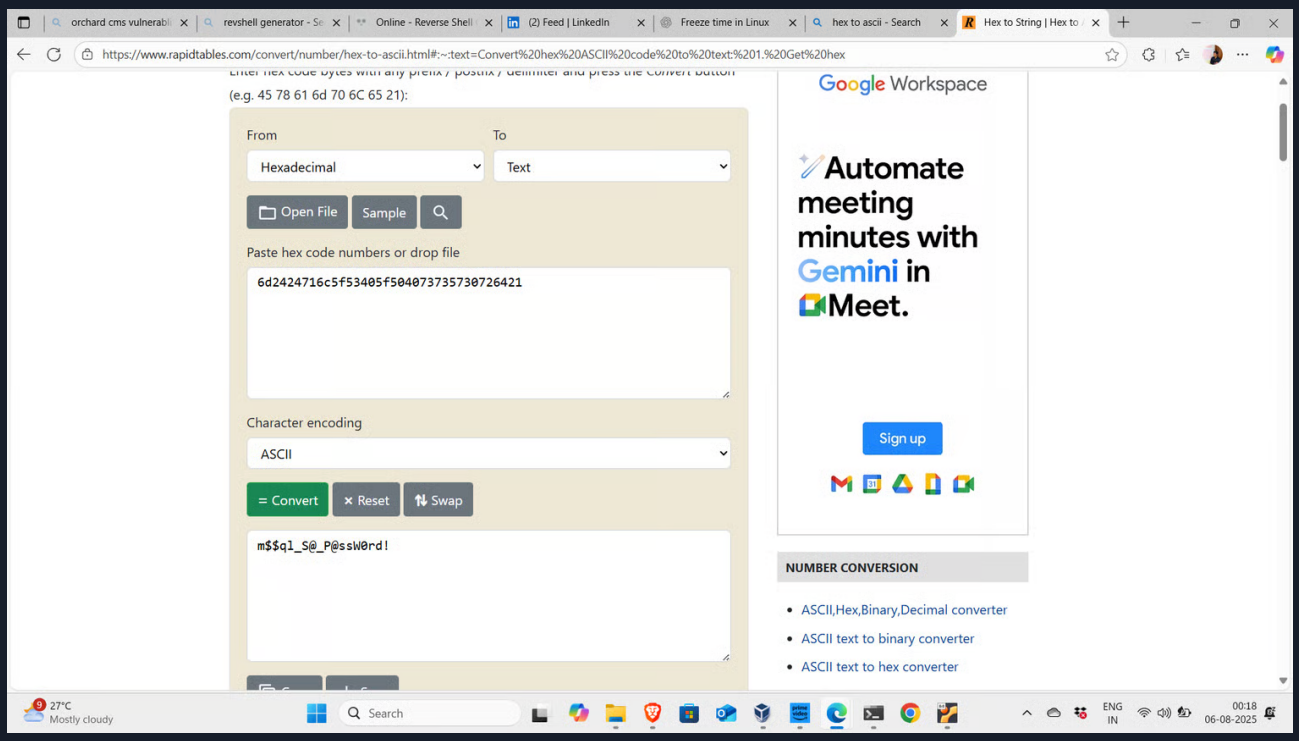
Command Used:

```
# Decoding the Base64 filename to Hex
echo "NmQyNDI0NzE2YzVvNTM0ODczNzUzMDc2NzNlMzI2NDIx" | base64 -d
# Output: 6d2424716c5f53405f504073735730726421 (Hex)

# Converting Hex to ASCII
echo "6d2424716c5f53405f504073735730726421" | xxd -r -p
# Output: m$$ql_S@_<REDACTED>
```

Evidence:

Figure 3: *Your terminal showing the echo command and the final decoded password*



Step 4: Verification of Credentials

To verify the validity of the leaked credentials, an attempt was made to authenticate to the MSSQL service on port 1433 using the `admin` username. The login was successful, confirming the credentials were valid for the database administrator account.

Command Used:

```
mssqlclient.py mantis.htb.local/admin:'m$$ql_S@_P@[Redacted]'@10.10.10.52
```

Evidence:

```
root@kali# mssqlclient.py 'admin:m$$ql_S@_P@ssW0rd!@10.10.10.52'
Impacket v0.9.21 - Copyright 2020 SecureAuth Corporation

[*] Encryption required, switching to TLS
[*] ENVCHANGE(DATABASE): Old Value: master, New Value: master
[*] ENVCHANGE(LANGUAGE): Old Value: None, New Value: us_english
[*] ENVCHANGE(PACKETSIZE): Old Value: 4096, New Value: 16192
[*] INFO(MANTIS\SQLEXPRESS): Line 1: Changed database context to 'master'.
[*] INFO(MANTIS\SQLEXPRESS): Line 1: Changed language setting to us_english.
[*] ACK: Result: 1 - Microsoft SQL Server (120 7208)
[!] Press help for extra shell commands
SQL>
```

3. Insecure Storage of Credentials (Database) - Medium

CWE	CWE-522 - Insufficiently Protected Credentials
CVSS 3.1	4.9 / CVSS:3.1/AV:N/AC:L/PR:H/UI:N/S:U/C:H/I:N/A:N
Root Cause	**Overview** Following the successful compromise of the Microsoft SQL Server (MSSQL) administrative account (detailed in Finding #1), the assessment team proceeded to enumerate the internal database structure to identify sensitive business data. The examination focused on the `OrchardDB` database, which backs the content management system. Within this database, a table named `blog_Orchard_Users_UserPartRecord` was identified. Querying this table revealed that user credentials—specifically for the domain user `james`—were stored in a format that allowed for immediate retrieval. Unlike standard security practices where passwords are hashed and salted (making them unreadable), these credentials appeared to be stored in cleartext or a reversible format. This vulnerability allowed the assessment team to pivot from a database context to a valid Active Directory user context.
Impact	**Business Impact** • **Lateral Movement:** The compromised user `james` was identified as a valid Domain User. Possessing these credentials allowed the assessment team to authenticate to other network services (SMB, Kerberos, RDP) that rely on Active Directory authentication. • **Credential Reuse Risk:** The storage of unhashed passwords suggests that if one component (the database) is breached, all user accounts associated with that application are immediately compromised without the need for offline cracking.
Remediation	TODO REMEDIATION
References	-

Finding Evidence

Attack Chain (Proof of Concept)

Step 1: Authenticated Database Connection

Using the SQL Administrator credentials recovered from the `secure_notes` directory (`admin` / `m$$ql_S@_P[Redacted]`), a connection was established to the MSSQL instance running on port 1433.

Command Used:

```
mssqlclient.py mantis.htb.local/admin:'m$$ql_S@[Redacted]'@10.10.10.52
```

Step 2: Table Enumeration & Data Exfiltration

Once connected, the context was switched to the `OrchardDB` database. A query was executed against the `blog_Orchard_Users_UserPartRecord` table to dump all user records.

Command Used:

```
SQL> USE OrchardDB;
SQL> SELECT * FROM blog_Orchard_Users_UserPartRecord;
```

Step 3: Credential Extraction The query output returned a record for the user `james`. The column containing the password (or password-equivalent data) was clearly visible.

Evidence:

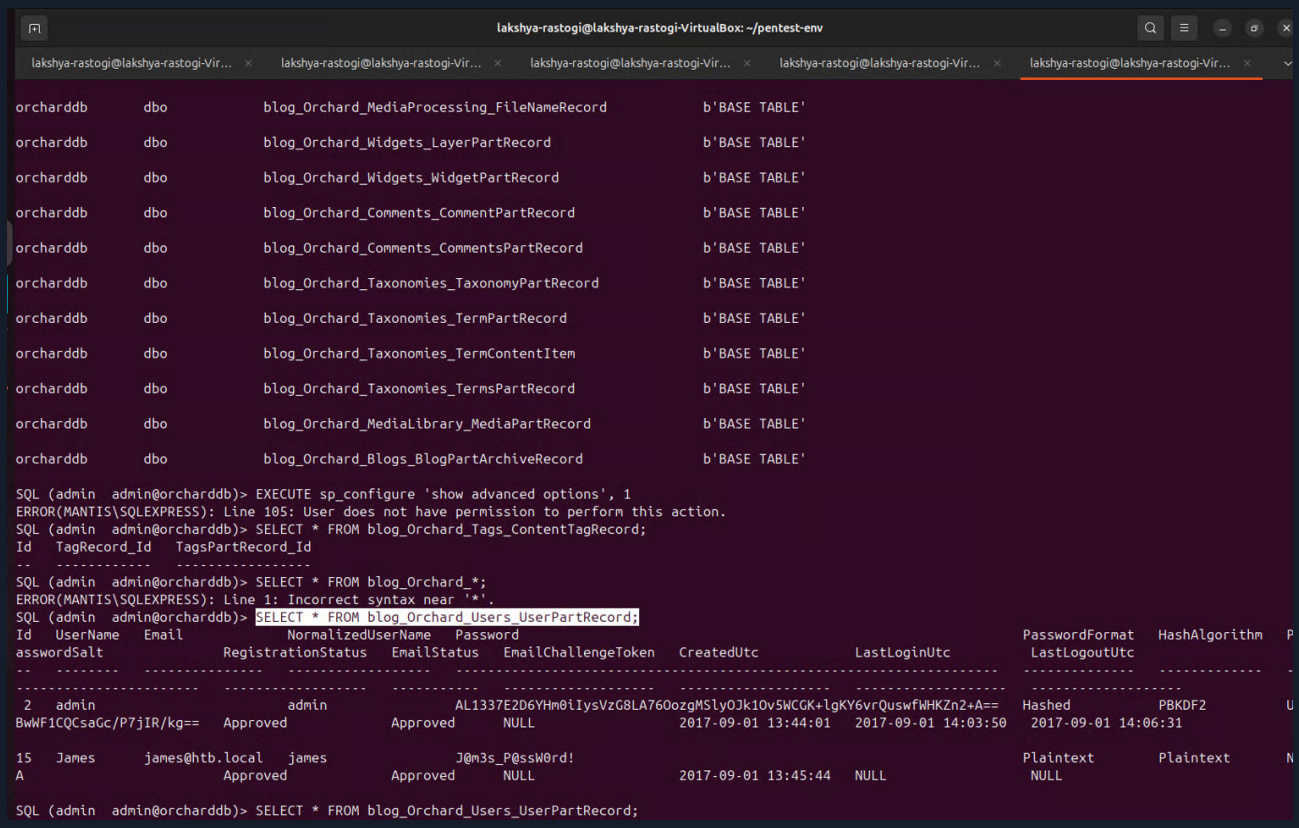


Figure 1: *SQL query result with the 'james' user and password*

Step 4: Verification of Domain Privileges

To confirm the impact of this finding, the extracted credentials for `james` were tested against the Domain Controller. The credentials were found to be valid for the Active Directory environment, confirming the ability to move laterally from the SQL server to the Domain Controller.

Command Used (Verification):

```
root@kali# crackmapexec smb 10.10.10.52 -u james -p '[Redacted]'
SMB         10.10.10.52     445    MANTIS           [*] Windows Server 2008 R2 Standard 7601
Service Pack 1 (name:MANTIS) (domain:htb.local) (signing:True) (SMBv1:True)
SMB         10.10.10.52     445    MANTIS           [+] htb.local\james:[Redacted]
```

A Appendix

A.1 Finding Severities

Each finding has been assigned a severity rating of critical, high, medium, low or info. The rating is based off of an assessment of the priority with which each finding should be viewed and the potential impact each has on the confidentiality, integrity, and availability of Mantis's data.

Rating	CVSS Score Range
Critical	9.0 – 10.0
High	7.0 – 8.9
Medium	4.0 – 6.9
Low	0.1 – 3.9
Info	0.0

A.2 Host & Service Discovery

IP Address	Port	Service	Notes
10.10.10.52	53/tcp	domain	DNS (Domain Name System)
10.10.10.52	88/tcp	kerberos-sec	Kerberos Authentication (KDC)
10.10.10.52	135/tcp	msrpc	Microsoft RPC Endpoint Mapper
10.10.10.52	139/tcp	netbios-ssn	NetBIOS Session Service
10.10.10.52	389/tcp	ldap	Lightweight Directory Access Protocol
10.10.10.52	445/tcp	microsoft-ds	SMB over TCP
10.10.10.52	464/tcp	kpasswd5	Kerberos Change Password Service
10.10.10.52	593/tcp	http-rpc-epmap	HTTP RPC Endpoint Mapper
10.10.10.52	636/tcp	ldapssl	LDAP over SSL
10.10.10.52	1337/tcp	waste	**Non-Standard HTTP Web Server (IIS)**
10.10.10.52	1433/tcp	ms-sql-s	**Microsoft SQL Server**
10.10.10.52	3268/tcp	globalcatLDAP	Microsoft Global Catalog (LDAP)
10.10.10.52	3269/tcp	globalcatLDAPssl	Microsoft Global Catalog over SSL
10.10.10.52	5722/tcp	msdfsr	Microsoft DFSR (Replication)
10.10.10.52	8080/tcp	http-proxy	Alternate HTTP Port
10.10.10.52	9389/tcp	adws	Active Directory Web Services
10.10.10.52	49152+	msrpc	High Ephemeral RPC Ports

A.3 Subdomain Discovery

URL	Description	Discovery Method
Nill	-	-

A.4 Exploited Hosts

Host	Scope	Method	Notes
10.10.10.52 (mantis.htb.local)	Internal Network	MS14-068 (Kerberos Golden PAC)	OS: Windows Server 2008 R2 Role: Domain Controller Access Level: NT AUTHORITY\SYSTEM

A.5 Compromised Users

Username	Type	Method	Notes
admin	SQL Service Account	Information Disclosure	Password: `[Redacted]` Source: Cleartext file in `/secure_notes/` directory.
james	Domain User	Insecure Storage of Credentials	Password: `[Redacted]` Source: Dumped from `OrchardDB` database.
NT AUTHORITY\SYSTEM	Local System	Privilege Escalation	Source: MS14-068 Golden Ticket Attack (Forged PAC).

A.6 Changes/Host Cleanup

Host	Scope	Change/Cleanup Needed
Nil	-	-

A.7 Flags Discovered

Flag #	Host	Flag Value	Flag Location	Method Used
N/A				

End of Report

This report was rendered
by SysReptor with
♥