

Лабораторная работа №3

Кибербезопасность предприятия

НКНБД-01-22; Аристид Жан, Акопян Сатеник,
Кадров Виктор, Нве Манге Хоце Херсон Мико,
Эспиноса Висилита Кристина Микаела,
НПИБД-01-22; Старикив Данила, НФИБД-02-22;
Чемоданова Ангелина

Содержание

1 Цель работы	4
2 Теоретическое введение	5
2.1 Легенда “Защита интеграционной платформы”	5
2.2 Описание сценария	5
2.3 Пояснения уязвимостей	6
3 Выполнение лабораторной работы	7
3.1 Первый инцидент	7
3.1.1 Обнаружение уязвимости	7
3.1.2 Устранение уязвимости “SQL-инъекция”	9
3.1.3 Последствие Web portal meterpreter	11
3.2 Второй инцидент	12
3.2.1 Обнаружение уязвимости	12
3.2.2 Устранение уязвимости “Отключенная защита антивируса”	13
3.2.3 Последствие Admin meterpreter	15
3.3 Третий инцидент	16
3.3.1 Обнаружение уязвимости	16
3.3.2 Устранение уязвимости “Слабый пароль учетной записи” . .	18
3.3.3 Последствие AD User	18
4 Вывод	22
Список литературы	23

Список иллюстраций

3.1 Сканирование на предмет SQL-инъекций	7
3.2 Детектирование SQL-инъекции	8
3.3 Загрузка вредоносного файла и выставление права доступа на выполнение	8
3.4 Карточка первого инцидента	9
3.5 Поиск места уязвимого параметра	9
3.6 Параметры уязвимой функции	10
3.7 Измененная функция actionView с проверкой типа параметра \$id	11
3.8 Завершение сессии с нарушителем	11
3.9 Настройки Windows Defender	12
3.10 Карточка второго инцидента	13
3.11 Удаление записи DisableAntiSpyware в реестре	14
3.12 Интерфейс Windows Defender	14
3.13 Включение Real-time Protection	15
3.14 Соединение с машиной нарушителя	15
3.15 Остановка процесса	16
3.16 Детектирование Remote Desktop Protocol	17
3.17 Карточка третьего инцидента	17
3.18 Изменение пароля администратора	18
3.19 Лог добавления нового пользователя	19
3.20 Удаление пользователя hacker в AD User & Computers	20
3.21 Удаление пользователя hacker в AD User & Computers	20
3.22 Результат проделанной работы	21

1 Цель работы

Основная цель данной лабораторной работы заключается в выполнении тренировки “Защита контроллера домена предприятия” в роли команды “Blue Team” (CSIRT – Computer Security Incident Response Team). В процессе выполнения работы необходимо освоить практические навыки выявления, анализа и устранения уязвимостей в различных системах, а также освоить навыки отработки действий по нейтрализации последствий успешных атак[1].

2 Теоретическое введение

2.1 Легенда “Защита интеграционной платформы”

Внешний злоумышленник находит в интернете сайт Компании и решает провести атаку на него с целью получения доступа к внутренним ресурсам компании. Обнаружив несколько уязвимостей на внешнем периметре и закрепившись на одном из серверов, Злоумышленник проводит разведку корпоративной сети с целью захватить контроллер домена.

Квалификация нарушителя средняя. Он умеет использовать инструментарий для проведения атак, а также знает техники постэксплуатации.

Злоумышленник обладает опытом проведения почтовых фишинговых рассылок[2].

2.2 Описание сценария

Последовательность действий нарушителя следующая:

1. Нарушитель проводит сканирование сети 195.239.174.0/24 и находит веб-сервер. Далее сканирует веб-сервер на предмет SQL-инъекций утилитой sqlmap. Нарушитель генерирует php reverse shell, используя найденную SQL-инъекцию, загружает вредоносный файл на веб-сервер. Для закрепления на хосте нарушитель устанавливает meterpreter-соединение.
2. Нарушитель определяет маршрут к сети 10.10.2.0/24, сканирует сеть и находит почтовый сервер. Нарушитель генерирует письмо с вредоносным

вложением и отправляет администратору.

3. Администратор открывает письмо, запускается вредоносный скрипт.
4. Нарушитель получает контроль над компьютером администратора и meterpreter-сессию.
5. Нарушитель находит AD&DNS сервер, проверяет, открыт ли порт 3389 (стандартный порт RDP). В случае открытого порта 3389 пытается с помощью инструмента hydra получить доступ к AD&DNS серверу, перебирая пароль по словарю. Для закрепления на контроллере домена нарушитель добавляет нового привилегированного пользователя.

2.3 Пояснения уязвимостей

Уязвимости и последствия:

- SQL-инъекция -> Web portal meterpreter
- Отключенная защита антивируса -> Admin meterpreter
- Слабый пароль учетной записи -> Добавление привилегированного пользователя

3 Выполнение лабораторной работы

3.1 Первый инцидент

3.1.1 Обнаружение уязвимости

Сетевой сенсор ViPNet IDS NS[3] детектирует события сканирования веб-сервера на предмет SQL-инъекций(рис. 3.1), использование определенного типа инъекции (Blind SQL-Injection)(рис. 3.2), а также загрузку вредоносного файла с php скриптом и выставление права доступа на выполнение(рис. 3.3).

События						
События за последние 24 часа	Фильтр	Сброс	Изменить	Сортировка	Показать	Скрыть
Y... Дата и время	Название правила					
17:38:33.601 23.10.... AM EXPLOIT Generic Possible XSS in URI: 'script' in request var 2	web-ap...	TCP	195.239.174.11	57858	10.10.1.20	80
17:38:33.601 23.10.... AM EXPLOIT Generic URL XSS attempt	web-ap...	TCP	195.239.174.11	57858	10.10.1.20	80
17:38:33.601 23.10.... ET WEB_SERVER Possible SQL Injection Attempt UNION SELECT in HTTP URI	web-ap...	TCP	195.239.174.11	57858	10.10.1.20	80
17:38:33.601 23.10.... AM EXPLOIT Generic Path Traversal in HTTP URI var 3	web-ap...	TCP	195.239.174.11	57858	10.10.1.20	80
17:38:28.370 23.10.... AM SQL Generic SQLi in HTTP URI: 'SELECT CASE' query	web-ap...	TCP	195.239.174.11	54026	10.10.1.20	80
17:38:28.370 23.10.... AM SQL [ET] Generic SQLi in HTTP URI: 'SELECT VERSION' query	web-ap...	TCP	195.239.174.11	54026	10.10.1.20	80
17:38:28.337 23.10.... AM SQL Generic SQLi in HTTP URI: 'SELECT CASE' query	web-ap...	TCP	195.239.174.11	54010	10.10.1.20	80
17:38:28.209 23.10.... AM SQL Generic SQLi in HTTP URI: 'UNION SELECT' query	web-ap...	TCP	195.239.174.11	53934	10.10.1.20	80
17:38:28.209 23.10.... AM SQL Generic SQLi in HTTP URI: 'SELECT CONCAT' query	web-ap...	TCP	195.239.174.11	53934	10.10.1.20	80
17:38:28.209 23.10.... ET WEB_SERVER MYSQL SELECT CONCAT SQL Injection Attempt	web-ap...	TCP	195.239.174.11	53934	10.10.1.20	80
17:38:28.209 23.10.... AM SQL Generic SQLi in HTTP URI: 'UNION SELECT' query var2	web-ap...	TCP	195.239.174.11	53934	10.10.1.20	80
17:38:28.197 23.10.... AM SQL Generic SQLi in HTTP URI: 'SELECT CONCAT' query	web-ap...	TCP	195.239.174.11	53926	10.10.1.20	80
17:38:28.197 23.10.... AM SQL Generic SQLi in HTTP URI: 'UNION SELECT' query	web-ap...	TCP	195.239.174.11	53926	10.10.1.20	80
17:38:28.197 23.10.... ET WEB_SERVER MYSQL SELECT CONCAT SQL Injection Attempt	web-ap...	TCP	195.239.174.11	53926	10.10.1.20	80
17:38:28.197 23.10.... AM SQL Generic SQLi in HTTP URI: 'UNION SELECT' query var2	web-ap...	TCP	195.239.174.11	53926	10.10.1.20	80
17:38:28.178 23.10.... AM SQL Generic SQLi in HTTP URI: 'UNION SELECT' query	web-ap...	TCP	195.239.174.11	53898	10.10.1.20	80

Рис. 3.1: Сканирование на предмет SQL-инъекций

События						
События за последние 24 часа						
Y...	Дата и время	Название правила	Класс	Про... IP-адрес источ... Порт ... IP-адрес получа... Порт полу...		
●	17:38:45.015 23.10...	AM SQL Generic SQLi in HTTP URI: 'ORDER BY' query	web-ap...	TCP 195.239.174.11 52800 10.10.1.20 80		
●	17:38:44.998 23.10...	AM SQL Generic SQLi in HTTP URI: 'ORDER BY' query	web-ap...	TCP 195.239.174.11 52786 10.10.1.20 80		
●	17:38:44.987 23.10...	AM SQL Generic SQLi in HTTP URI: 'ORDER BY' query	web-ap...	TCP 195.239.174.11 52774 10.10.1.20 80		
●	17:38:44.970 23.10...	AM SQL Generic SQLi in HTTP URI: 'ORDER BY' query	web-ap...	TCP 195.239.174.11 58170 10.10.1.20 80		
●	17:38:39.950 23.10...	ET WEB_SERVER SQL Injection Select Sleep Time Delay	web-ap...	TCP 195.239.174.11 58186 10.10.1.20 80		
●	17:38:39.950 23.10...	AM SQL SLEEP function in GET [Possible Blind SQL Injection Attempt]	client-si...	TCP 195.239.174.11 58186 10.10.1.20 80		
●	17:38:39.950 23.10...	AM SQL Generic SQLi in HTTP URI: 'SELECT FROM' query	client-si...	TCP 195.239.174.11 58186 10.10.1.20 80		
●	17:38:39.950 23.10...	AM SQL Generic SQLi in HTTP URI: 'SELECT SLEEP' query	web-ap...	TCP 195.239.174.11 58186 10.10.1.20 80		
●	17:38:39.940 23.10...	ET WEB_SERVER SQL Injection Select Sleep Time Delay	web-ap...	TCP 195.239.174.11 58164 10.10.1.20 80		
●	17:38:39.940 23.10...	AM SQL SLEEP function in GET [Possible Blind SQL Injection Attempt]	client-si...	TCP 195.239.174.11 58164 10.10.1.20 80		
●	17:38:39.940 23.10...	AM SQL Generic SQLi in HTTP URI: 'SELECT FROM' query	client-si...	TCP 195.239.174.11 58164 10.10.1.20 80		
●	17:38:39.940 23.10...	AM SQL Generic SQLi in HTTP URI: 'SELECT SLEEP' query	web-ap...	TCP 195.239.174.11 58164 10.10.1.20 80		
●	17:38:38.233 23.10...	AM SQL Generic SQLi in HTTP URI: UNION SELECT query	web-ap...	TCP 195.239.174.11 10.10.1.20 80		
●	17:38:38.233 23.10...	AM SQL Generic SQLi in HTTP URI: 'SELECT CONCAT' query	web-ap...	TCP 195.239.174.11 10.10.1.20 80		
●	17:38:38.233 23.10...	ET WEB_SERVER MYSQL SELECT CONCAT SQL Injection Attempt	web-ap...	TCP 195.239.174.11 10.10.1.20 80		
●	17:38:38.233 23.10...	AM SQL Generic SQLi in HTTP URI: 'UNION SELECT' query var2	web-ap...	TCP 195.239.174.11 10.10.1.20 80		

Рис. 3.2: Детектирование SQL-инъекции

События						
События за последние 24 часа						
Y...	Дата и время	Название правила	Класс			
●	17:38:45.540 23.10...	AM SQL Generic SQLi in HTTP URI: 'SELECT LOAD_FILE' query	web-application			
●	17:38:45.540 23.10...	ET WEB_SERVER Possible SQL Injection SELECT CAST in HTTP URI	attempted-admin			
●	17:38:45.513 23.10...	AM SQL DUMPFILE function in GET - Possible SQL Injection Attempt	web-application			
●	17:38:45.513 23.10...	AM SQL Generic SQLi in HTTP URI: 'SELECT DUMPFILE' query	web-application			
●	17:38:45.513 23.10...	AM EXPLOIT Generic Possible PHP Injection: hex encoded PHP Tag in HTTP request	web-application			
●	17:38:45.455 23.10...	AM SQL DUMPFILE function in GET - Possible SQL Injection Attempt	web-application			
●	17:38:45.455 23.10...	AM SQL Generic SQLi in HTTP URI: 'SELECT DUMPFILE' query	web-application			
●	17:38:45.455 23.10...	AM EXPLOIT Generic Possible PHP Injection: hex encoded PHP Tag in HTTP request	web-application			
●	17:38:45.404 23.10...	AM SQL Generic SQLi in HTTP URI (double encoded) var 1	web-application			
●	17:38:45.404 23.10...	AM SQL Generic SQLi in HTTP URI: 'INTO OUTFILE' query	web-application			
●	17:38:45.404 23.10...	ET WEB_SERVER Possible SQL Injection INTO OUTFILE Arbitrary File Write Attempt	web-application			
●	17:38:45.404 23.10...	AM EXPLOIT Generic Possible PHP Injection: hex encoded PHP Tag in HTTP request	web-application			
●	17:38:45.360 23.10...	AM SQL Generic SQLi in HTTP URI: 'SELECT CASE' query	web-application			
●	17:38:45.360 23.10...	ET WEB_SERVER Possible attempt to enumerate MS SQL Server version	attempted-admin			
●	17:38:45.347 23.10...	AM SQL Generic SQLi in HTTP URI: 'SELECT CASE' query	web-application			
●	17:38:45.347 23.10...	AM SQL [ET] Generic SQLi in HTTP URI: 'SELECT VERSION' query	web-application			

Событие 17:38:45.455 23.10.2025

Событие	Источник	Получатель	Пакет
---------	----------	------------	-------

Общая информация

Дата и время	17:38:45.455 23.10.2025
Интерфейс захвата	eth2
Уровень важности	Высокий
Тип события	Сигнатурное событие
Протокол	TCP
Код события	3000708
Клиентское приложение	sqlmap/1.7.2#stable (https://sqlmap.org)
Доменное имя ресурса	195.239.174.95

Правило анализа

Класс	web-application-attack
Группа	sql
Название	AM SQL DUMPFILE function in GET - Possible SQL Injection Attempt

Описание:

Правило обнаруживает атаки на реляционную базу данных SQL.

Текст:

```
alert tcp $EXTERNAL_NET any -> $HOME_NET $HTTP_PORTS (msg:"AM SQL DUMPFILE function in GET - Possible SQL Injection Attempt";flow: to_server,establish;
```

Рис. 3.3: Загрузка вредоносного файла и выставление права доступа на выполнение

Заполним карточку первого инцидента(рис. 3.4).

The screenshot shows a dark-themed incident report card. At the top left is the title 'SQL-инъекция'. Below it are tabs for 'Основная информация' (Main information) and 'Чат' (Chat), with 'Основная информация' being the active tab. On the right, there's a red button labeled 'Закрытый' (Closed). The main content area is divided into several sections: 'Дата и время события' (Event date and time) showing '20.10.2025 19:30'; 'Описание' (Description) stating that a PHP web server on port 80 has a vulnerable service that allows for a reverse shell via the 'id' parameter; 'Индикаторы компрометации' (Compromise indicators) listing suspicious files, processes, or high connection counts to a single file; 'Рекомендации' (Recommendations) suggesting changes to configuration files; and 'Прикреплённые файлы' (Attached files). On the right side, there are sections for 'Оценка' (Rating) with five stars, 'Автор' (Author) listed as 'Чемоданова Ангелина' with email '@1132226443@pfur.ru', 'Ответственный' (Responsible) listed as 'Кадров Виктор' with email '@1132226454@pfur.ru', and 'Источник' (Source) showing '195.239.174.11'. Below these is a section for 'Поражённые активы' (Affected assets) with the IP '10.10.1.20'.

Рис. 3.4: Карточка первого инцидента

3.1.2 Устранение уязвимости “SQL-инъекция”

Решение: известно, что \$id является уязвимым параметром, следует проверять тип данного параметра. Требуется найти место кода, где данный параметр считывается из GET запроса (рис. 3.5).

The terminal session shows a user logging in as root on a system named 'webportal1'. The user navigates to the '/var/www/html/htdocs/polygon/' directory and runs 'ls' to list files. Then, they open the 'NewsController.php' file in nano editor. A cursor is shown pointing to the word 'NewsController.php' in the file.

Рис. 3.5: Поиск места уязвимого параметра

Считывание параметра сайта происходит в функции actionView() в файле NewsController.php (рис. 3.6).

The screenshot shows a Windows desktop environment. In the center is a terminal window titled 'user@webportal: ~' with the file 'File: controllers/NewsController.php' open. The code displays several functions: 'actionIndex', 'actionView', and 'actionAddComment'. The 'actionAddComment' function includes a POST request with parameters like 'name', 'email', 'text', and 'post_id'. Below the terminal, the Windows taskbar is visible with icons for various applications including Putty, Remote Desktop, and WinSCP. The system tray shows the date and time as 10/20/2025 at 7:51 PM.

```
user@webportal: ~
File: controllers/NewsController.php

class NewsController extends Controller
{
    public function actionIndex()
    {
        $models = News::model()->findAll();
        foreach ($models as $key => $model) {
            $models[$key][4] = News::model()->commentsCount($model[0]);
        }
        $this->render('news/index', array('model'=>$models));
    }

    public function actionView()
    {
        $id = $_GET['id'];
        $model = News::model()->findId($id);
        $comments = Comment::model()->findByAttributes(array('post_id'=>$id));
        $this->render('news/view', array('model'=>$model, 'comments'=>$comments));
    }

    public function actionAddComment()
    {
        $model = Comment::model()->create(array(
            'name'=>$_POST['name'],
            'email'=>$_POST['email'],
            'text'=>$_POST['text'],
            'post_id'=>$_POST['post_id'],
        ));
        header('Location: ' . $_SERVER['HTTP_REFERER']);
        exit();
    }
}
```

Рис. 3.6: Параметры уязвимой функции

Для проверки типа \$id используется функция `is_numeric`, которая возвращает True в случае, если \$id – число, иначе – False. В случае успешной проверки параметр \$id будет передаваться в запрос, иначе – запрос будет статичным и независимым от \$id (рис. 3.7).

```

GNU nano 2.7.4                               File: controllers/NewsController.php
Modified ^

<?php

class NewsController extends Controller
{
    public function actionIndex()
    {
        $models = News::model()->findAll();
        foreach ($models as $key => $model) {
            $models[$key][4] = News::model()->commentsCount($model[0]);
        }

        $this->render('news/index', array('model'=>$models));
    }

    public function actionView()
    {
        $id = $_GET['id'];
        if (!is_numeric($id)){
            $id = 1;
        }
        $model = News::model()->findById($id);
        $comments = Comment::model()->findByAttributes(array('post_id'=>$id));
        $this->render('news/view', array('model'=>$model, 'comments'=>$comments));
    }

    public function actionAddComment()
    {
        $model = Comment::model()->create(array(
            'name'=>$_POST['name'],
            'email'=>$_POST['email'],
            'text'=>$_POST['text'],
            'post_id'=>$_POST['post_id'],
        ));
    }
}

```

Get Help WC Write Out WM Where Is XK Cut Text J Justify C Cur Pos Y Prev Page M\ First Line
 Exit WR Read File R Replace U Uncut Text T To Spell G Go To Line N Next Page M/ Last Line

Рис. 3.7: Измененная функция actionView с проверкой типа параметра \$id

После внесения изменений в файл конфигурации и проверки значения параметра \$id уязвимость SQL-инъекции успешно устранена[4].

3.1.3 Последствие Web portal meterpreter

Нарушитель устанавливает shell сессию с веб- порталом PHP. Для обнаружения последствия необходимо проверить сокеты уязвимой машины при помощи утилиты ss с ключами -tp(рис. 3.8).

На скриншоте изображено активное соединение веб-портала с IP-адресом нарушителя (195.239.174.11). Необходимо устраниТЬ соединение, воспользовавшись командой kill (рис. 3.8).

Local Address:Port	Peer Address:Port
10.10.1.20:43908	195.239.174.11:49444
10.10.1.20:48842	10.10.1.25:5044
10.10.1.20:82954	10.10.2.17:28004
10.10.1.20:50236	195.239.174.125:puppet
10.10.1.20:proxxy	10.10.1.253:61576
10.10.1.20:ssh	10.10.1.253:46696
fffff:10.10.1.20:http	::ffff:195.239.174.11:37466

Рис. 3.8: Завершение сессии с нарушителем

В результате выполнения команды сессия с нарушителем завершена, последствие Web portal meterpreter успешно устранено[4].

3.2 Второй инцидент

3.2.1 Обнаружение уязвимости

Один из способов проверки состояния защиты в реальном времени Windows Defender – в Powershell ввести команду Get-MpPreference и проверить значение параметра DisableRealtimeMonitoring. Если значение – True, то защита в реальном времени выключена(рис. 3.9).

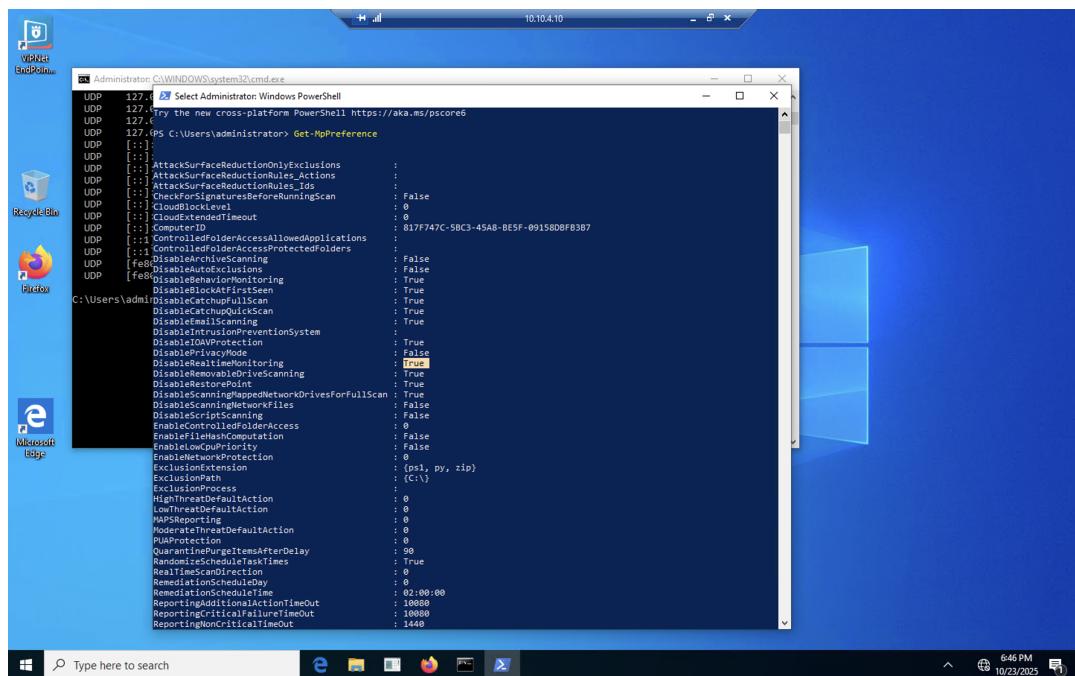


Рис. 3.9: Настройки Windows Defender

На вышеупомянутом рисунке изображено значение «true» параметра DisableRealtimeMonitoring, что означает отключенную защиту антивируса на узле.

Заполним карточку второго инцидента(рис. 3.10).

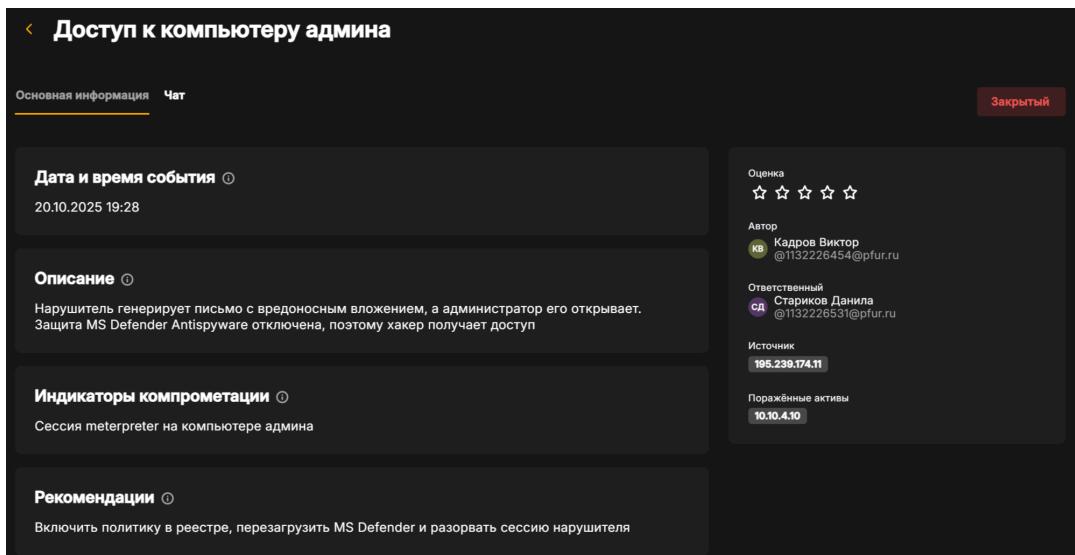


Рис. 3.10: Карточка второго инцидента

3.2.2 Устранение уязвимости “Отключенная защита антивируса”

Решение: на узле Administrator Workstation вручную удалить запись в реестре или через консоль, используя команду(рис. 3.11): REG DELETE «HKLM/SOFTWARE/Policies/Microsoft/Windows Defender» /v DisableAntiSpyware. Подтвердить действие, далее в Windows Defender перезапустить Virus & Threat Protection (рис. 3.12) и включить Real-time Protection (рис. 3.13).

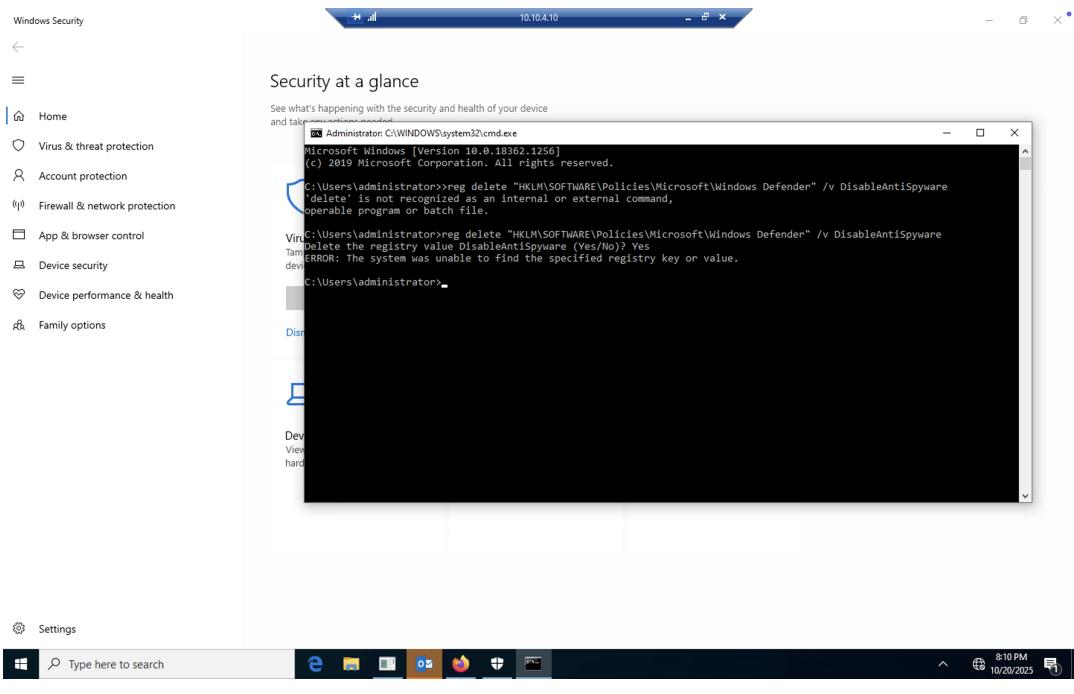


Рис. 3.11: Удаление записи DisableAntiSpyware в реестре

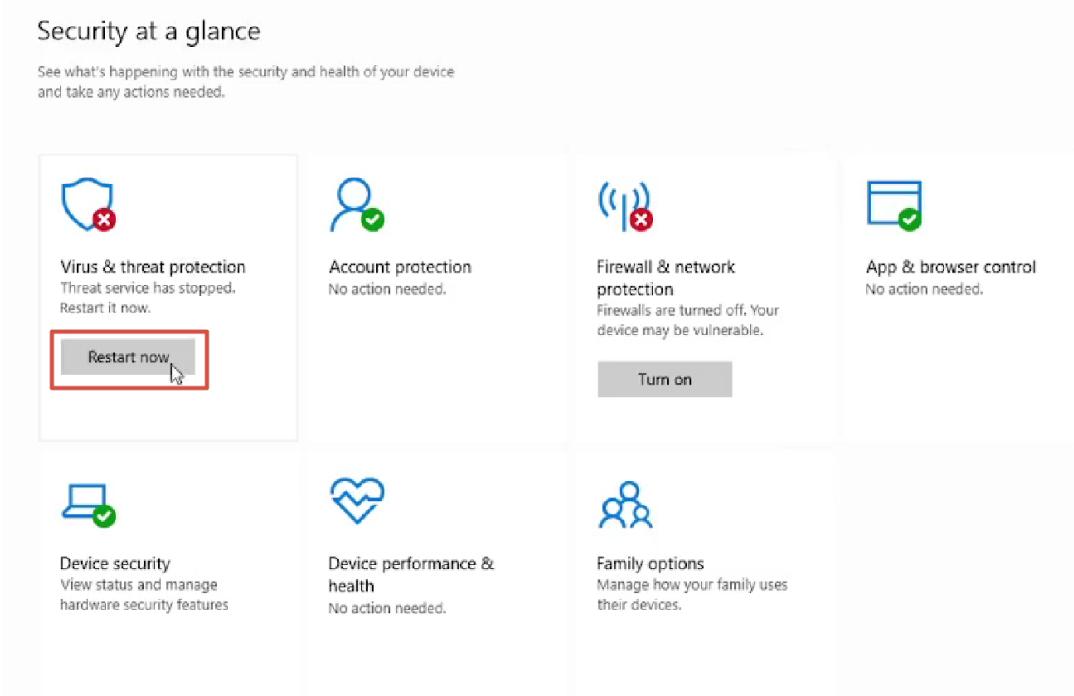


Рис. 3.12: Интерфейс Windows Defender

❖ Virus & threat protection settings

View and update Virus & threat protection settings for Windows Defender Antivirus.

Real-time protection

Locates and stops malware from installing or running on your device. You can turn off this setting for a short time before it turns back on automatically.



Рис. 3.13: Включение Real-time Protection

После удаления записи реестра и включения защиты антивирусной программы Microsoft Defender необходимо перезагрузить Windows[4].

3.2.3 Последствие Admin meterpreter

Установленную сессию с нарушителем можно обнаружить при помощи утилиты netstat с ключами -ano(рис. 3.14).

Proto	Local Address	Foreign Address	State	PID
TCP	0.0.0.0:135	0.0.0.0:0	LISTENING	984
TCP	0.0.0.0:445	0.0.0.0:0	LISTENING	4
TCP	0.0.0.0:3389	0.0.0.0:0	LISTENING	1004
TCP	0.0.0.0:5040	0.0.0.0:0	LISTENING	7476
TCP	0.0.0.0:5985	0.0.0.0:0	LISTENING	4
TCP	0.0.0.0:47001	0.0.0.0:0	LISTENING	4
TCP	0.0.0.0:49664	0.0.0.0:0	LISTENING	728
TCP	0.0.0.0:49665	0.0.0.0:0	LISTENING	536
TCP	0.0.0.0:49666	0.0.0.0:0	LISTENING	1208
TCP	0.0.0.0:49667	0.0.0.0:0	LISTENING	1764
TCP	0.0.0.0:49670	0.0.0.0:0	LISTENING	2192
TCP	0.0.0.0:49671	0.0.0.0:0	LISTENING	2876
TCP	0.0.0.0:49672	0.0.0.0:0	LISTENING	728
TCP	0.0.0.0:49693	0.0.0.0:0	LISTENING	3168
TCP	0.0.0.0:49712	0.0.0.0:0	LISTENING	684
TCP	10.10.4.10:130	0.0.0.0:0	LISTENING	4
TCP	10.10.4.10:3389	10.10.4.12:63818	ESTABLISHED	1004
TCP	10.10.4.10:49811	10.10.2.11:443	ESTABLISHED	4132
TCP	10.10.4.10:49820	10.10.2.11:443	ESTABLISHED	4132
TCP	10.10.4.10:50177	10.10.1.25:5044	ESTABLISHED	10496
TCP	10.10.4.10:56234	195.239.174.11:444	ESTABLISHED	8952
TCP	10.10.4.10:56355	10.10.2.11:443	ESTABLISHED	6452
TCP	10.10.4.10:56842	10.10.2.11:443	ESTABLISHED	6452
TCP	10.10.4.10:59182	10.10.2.15:80	ESTABLISHED	6676
TCP	10.10.4.10:60170	10.10.2.15:80	TIME_WAIT	0
TCP	10.10.4.10:60171	10.10.2.15:80	TIME_WAIT	0
TCP	10.10.4.10:60172	10.10.2.15:80	TIME_WAIT	0
TCP	10.10.4.10:60173	10.10.2.15:80	TIME_WAIT	0
TCP	10.10.4.10:60174	10.10.2.15:80	TIME_WAIT	0

Рис. 3.14: Соединение с машиной нарушителя

Для устранения необходимо завершить сессию с машиной нарушителя.

Например, при помощи команды taskkill /f /pid (рис. 3.15).

The screenshot shows a Windows PowerShell window titled "Select Administrator: Windows PowerShell" running on a host with IP 10.10.4.10. The window displays a list of network connections (TCP and UDP) with their local and remote addresses, ports, and PID. A command is being typed at the bottom:

```
PS C:\Users\administrator> taskkill /f /pid 8952
SUCCESS: The process with PID 8952 has been terminated.
PS C:\Users\administrator>
```

Рис. 3.15: Остановка процесса

В результате выполнения команды сессия с машиной нарушителя завершена, последствие Admin meterpreter успешно устранено[4].

3.3 Третий инцидент

3.3.1 Обнаружение уязвимости

С помощью ViPNet IDS NS[3] в сетевом трафике обнаруживаются множественные попытки подключения к хосту AD&DNS с портом 3389, сканирование системы, что может говорить о попытках подбора пароля. Также если мы зайдем на сам узел MS Active Directory, откроем Viewer Properties, перейдем в необходимую директорию с событиями (TerminalServices...), то сможем увидеть событие с кодом 1149, которое говорит о том, что пользователю удалось подключиться по RDP(рис. 3.16).

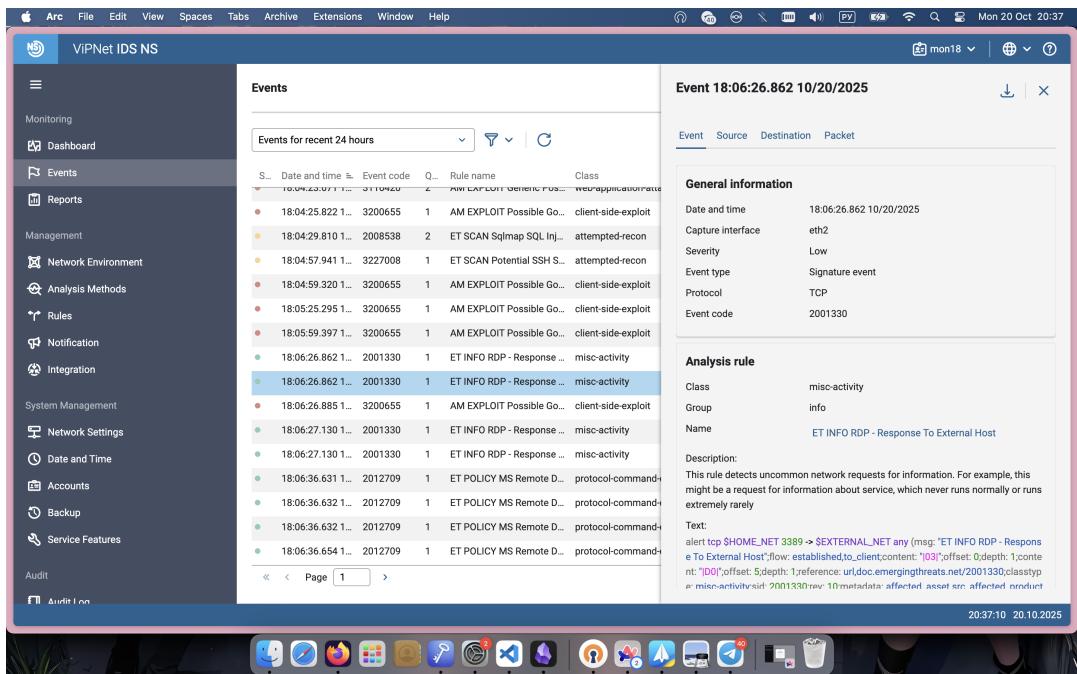


Рис. 3.16: Детектирование Remote Desktop Protocol

Заполним карточку третьего инцидента(рис. 3.17).

Рис. 3.17: Карточка третьего инцидента

3.3.2 Устранение уязвимости “Слабый пароль учетной записи”

Решение: изменить пароль к учетной записи администратора на более сложный, не содержащийся в словарях(рис. 3.18).

```
C:\Users\Administrator>net user Administrator *
Type a password for the user:
Retype the password to confirm:
The command completed successfully.

C:\Users\Administrator>_
```

Рис. 3.18: Изменение пароля администратора

На вышеупомянутом рисунке изображена смена пароля администратора на узле MS Active Directory командой «`net user Administrator *`». В результате изменения ненадежного пароля уязвимость успешно устранена[4].

3.3.3 Последствие AD User

Добавление нового привилегированного пользователя можно отследить с помощью аудита событий входа в учетную запись Windows security, где появится событие с ID 4720. Необходимо перейти в Event Viewer и в Windows Logs – Security, затем применить фильтр на логи. Ниже показан лог, генерируемый при добавлении нового пользователя(рис. 3.19).

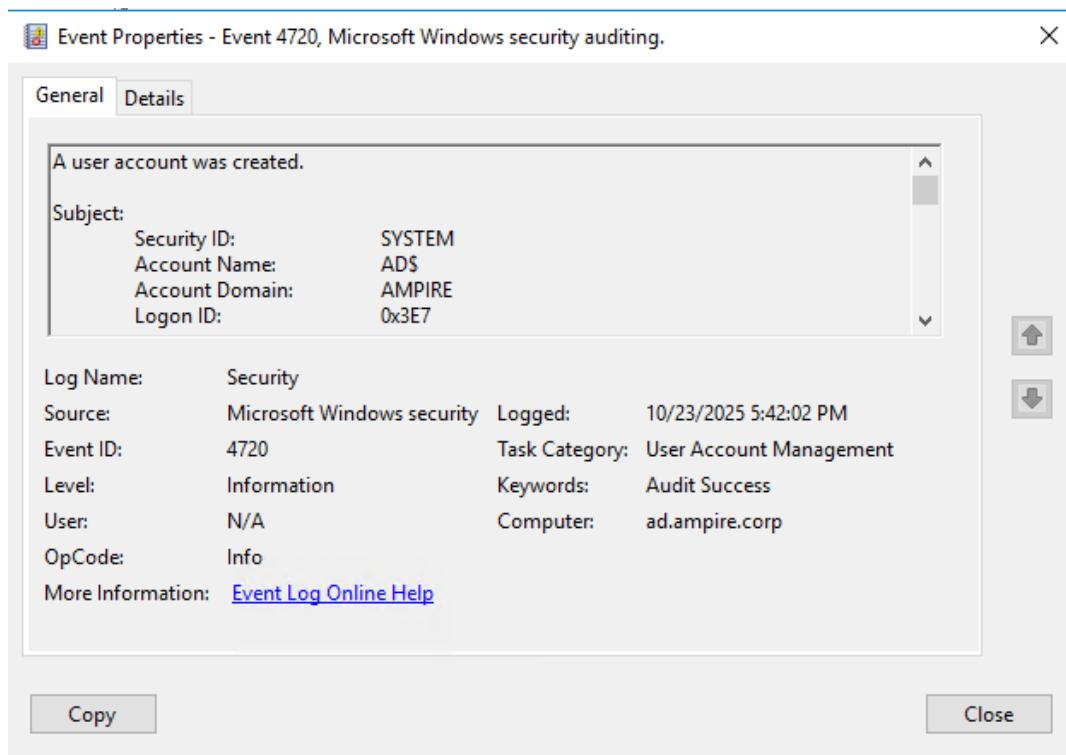


Рис. 3.19: Лог добавления нового пользователя

Для удаления пользователя необходимо зайти в Administrative Tools – Active Directory Users and computers. Затем во вкладке Users найти и удалить нового привилегированного пользователя с именем «Hacked»(рис. 3.20 - рис. 3.21).

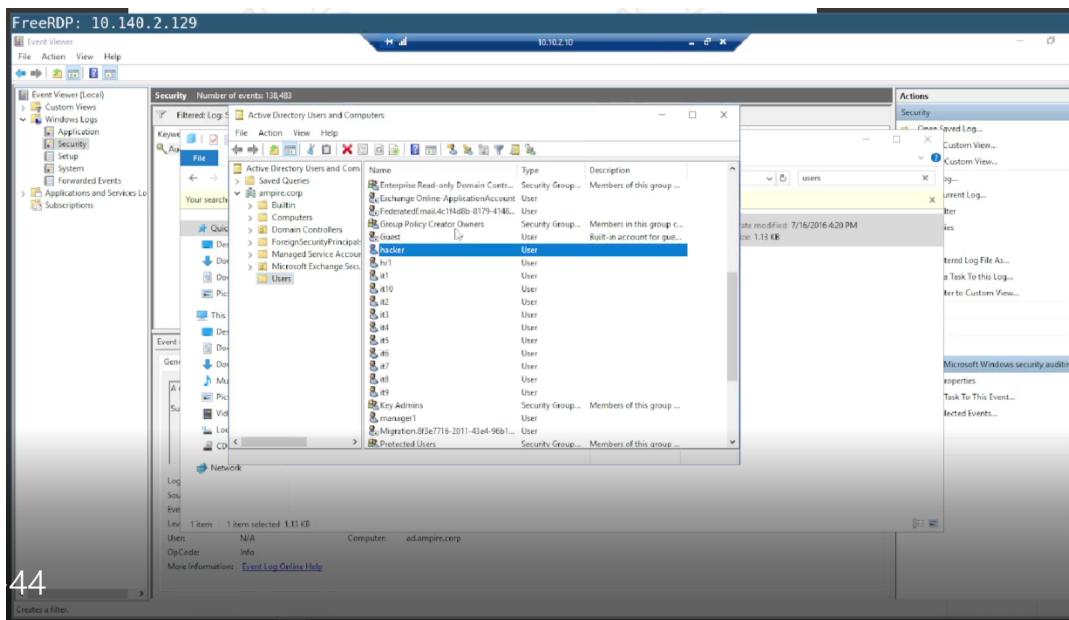


Рис. 3.20: Удаление пользователя hacker в AD User & Computers

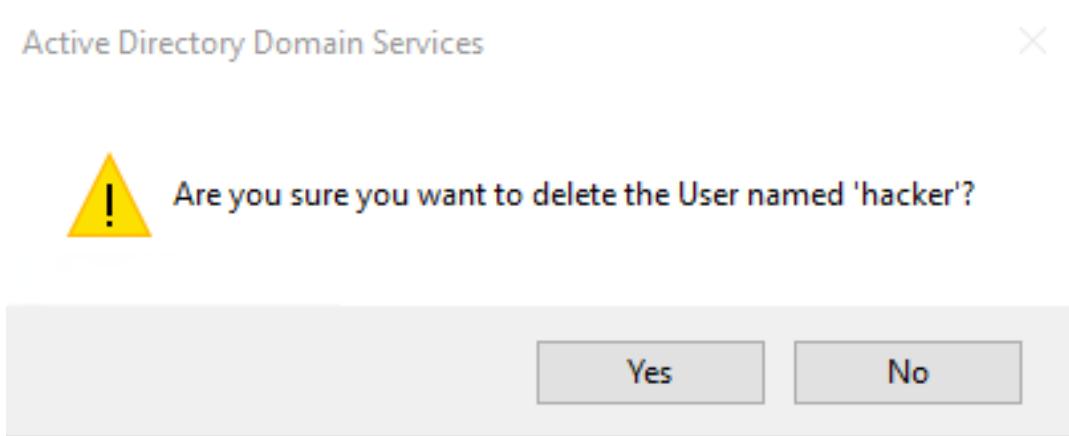


Рис. 3.21: Удаление пользователя hacker в AD User & Computers

В результате выполнения вышеупомянутых действий привилегированный пользователь удален, последствие AD User успешно устранено[4].

Результат проделанной работы(рис. 3.22).

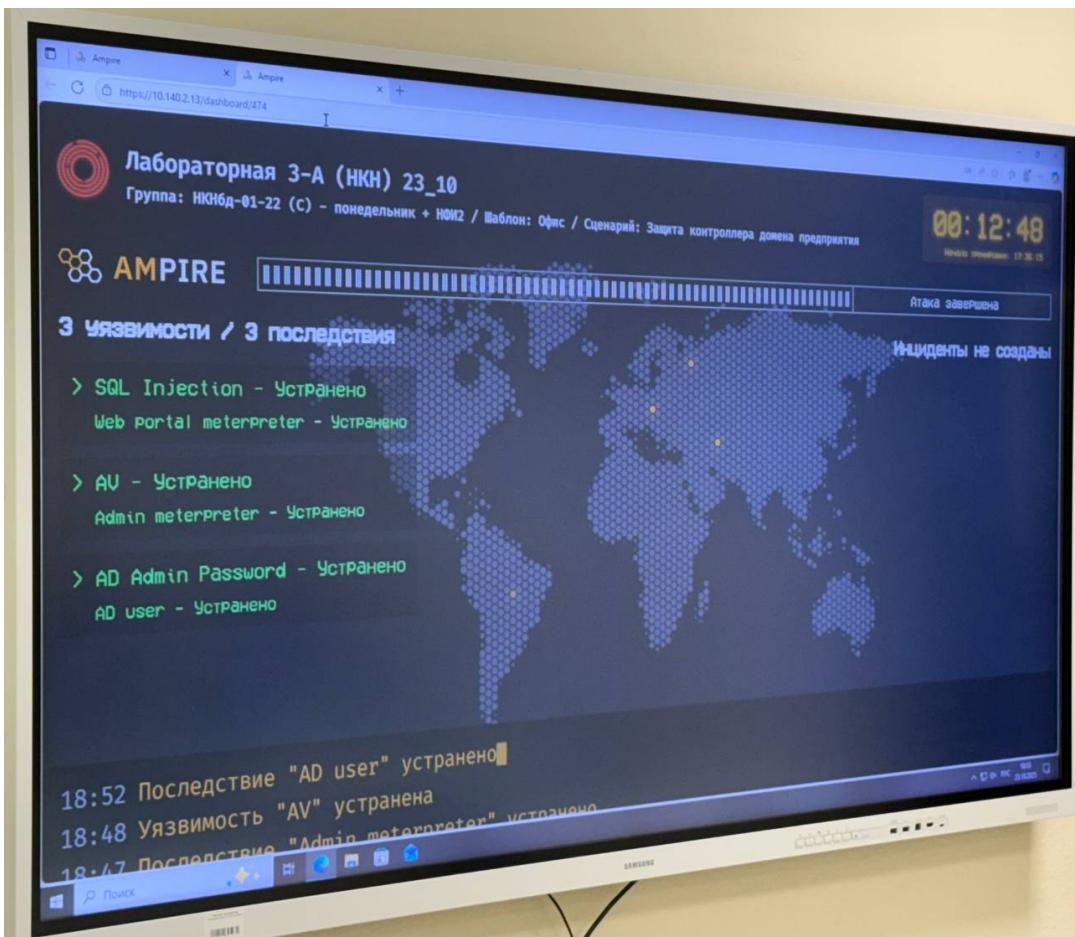


Рис. 3.22: Результат проделанной работы

4 Вывод

В ходе выполнения данной лабораторной работы мы выполнили тренировку “Защита контроллера домена предприятия” в роли команды “Blue Team” (CSIRT – Computer Security Incident Response Team). В процессе выполнения работы освоили практические навыки выявления, анализа и устранения уязвимостей в различных системах, а также освоили навыки отработки действий по нейтрализации последствий успешных атак.

Список литературы

1. AM Threat Intelligence Portal [Электронный ресурс].
2. Легенда сценария [Электронный ресурс].
3. Сетевой сенсор системы обнаружения атак программно-аппаратный комплекс ViPNet IDS NS 3 Руководство администратора [Электронный ресурс].
4. Методическое пособие преподавателя [Электронный ресурс].