Kerberoasting для Red Team

habr.com/ru/articles/697820

hotmanchester

hotmanchester 7 ноя 2022 в 13:36
5 мин

4.3K
Из песочницы
Тестирование на проникновение Active Directory – зрелище не для слабонервных. Стоит только взглянуть на дорожную карту Пентеста Active Directory: “Active Directory Penetration Mind Map” как сразу становится ясным то, что это вовсе не «легкая прогулка». Тем не менее, к настоящему времени исследователями, энтузиастами и другими неравнодушными собрано достаточно большое количество статей и материалов, в которых (по моему скромному мнению) можно найти ответ на любой вопрос, и при этом рассмотреть проблему с разных сторон! По-моему мнению, данный материал возможно «размылит» тот самый замыленный глаз после использования несметного числа утилит и методик при тестировании на проникновение, и возможно заставит задуматься о тех средствах и методах, которые мы применяем в повседневной деятельности!

Более подробно о Kerberoasting вы сможете прочитать у данных авторов:

https://habr.com/ru/post/650889/

https://ardent101.github.io/posts/kerberos_general_attacks/#kerberoasting

Далее будут рассмотрены лишь типовые ошибки RedTeam при проведении данной атаки и предложен способ, который позволит их избежать.

Исходные данные: у нас есть credentials от доменного пользователя (история умалчивает о том, откуда они у нас). По большому счету их наличие может (в теории) открыть перед нами все двери. И одно из тех самых минимальных действий, которое нам доступно, является подключение по протоколу LDAP к контроллеру домена. Существуют разные способы отправки LDAP requests, в данной статье я буду использовать **LDAPAdmin** для более удобного восприятия информации!

SPN (Service Principal Name)

На вид это самая безобидная строка, но по сути, любой аутентифицированный пользователь обладает возможностью запросить Service Ticket для учетной записи [с атрибутом **servicePrincipalName**], затем экспортировать STicket, и ввиду того, что он зашифрован на секрете сервиса, может (~~попытаться~~) подобрать пароль по словарю offline.

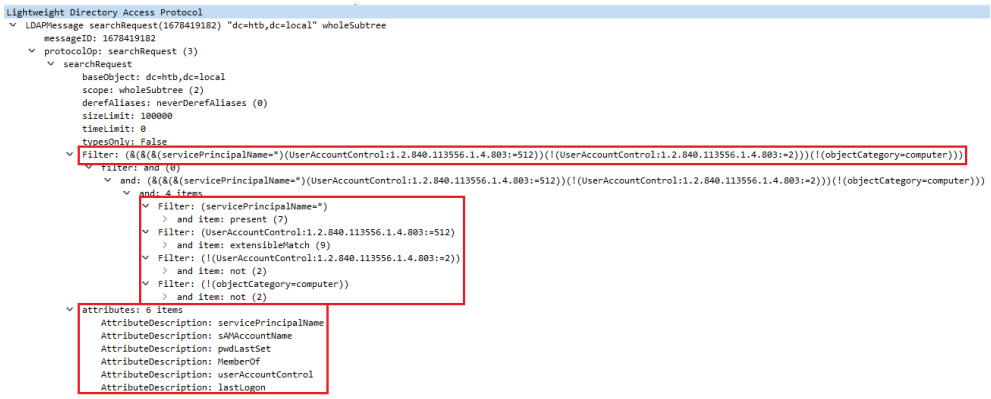
Impacket-GetUserSPNs

Во многих обзорах на Kerberoasting (вполне заслужено) в практической части, для демонстрации данной атаки используют **Impacket-GetUserSPNs**

```
impacket-GetUserSPNs -dc-ip 10.10.10.161  htb.local/svc-alfresco:s3rvice -request -outputfile kerberoastable
```

классический подход к выполнению Kerberosat (HTB Forest)

В описании к опции **–request** говорится, что происходит запрос TGS для пользователей и дальнейший вывод в формате JtR/hashcat формате. Но давайте захватим Wireshark трафик при использовании данной утилиты и посмотрим, а что собственно запрашиваем!



Перехваченный трафик с LDAP запросом

Перед собой мы наблюдаем LDAP request, содержащий определенный набор фильтров. Рассмотрим каждый из них по отдельности!

Filter: (!(objectCategory=computer)) в данном случае говорит, что нас интересуют исключительно пользователи, а не (!) компьютеры, так как пароль к учетным записям компьютеров не является словарным.

Filter: (!(UserAccountControl:1.2.840.113556.1.4.803:=2)) означает следующее: мы ищем активные учетные записи (не отключенные (!))

Значение **2** в поле **UserAccountControl** [обрати внимание, что «**!**» знак – НЕ]

ACCOUNTDISABLE (Учетная запись отключена)	0x0002	2

Десятичная двойка - УЗ отключена

Filter: (UserAccountControl:1.2.840.113556.1.4.803:=512)

Значение 512 в поле UserAccountControl

NORMAL_ACCOUNT (Учетная запись по умолчанию. Обычная активная учетная запись)	0x0200	512

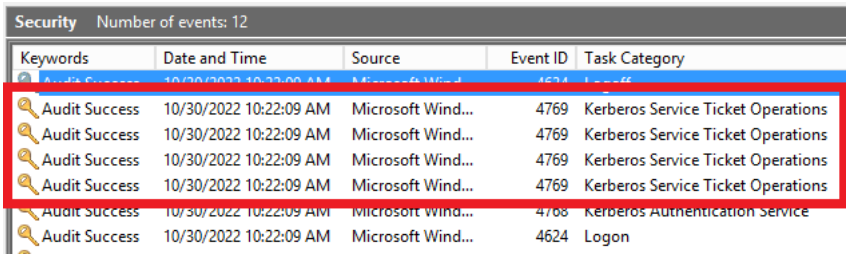
Ищем активную УЗ

Более детальное описание атрибута **UserAccountControl** можно посмотреть:

https://winitpro.ru/index.php/2018/05/14/convertaciya-atributa-useraccountcontrol-v-ad/

Filter: (servicePrincipalName=*) в данном фильтре джокерный символ **«*»** предполагает поиск всех значений. (Берем все, без разбору)

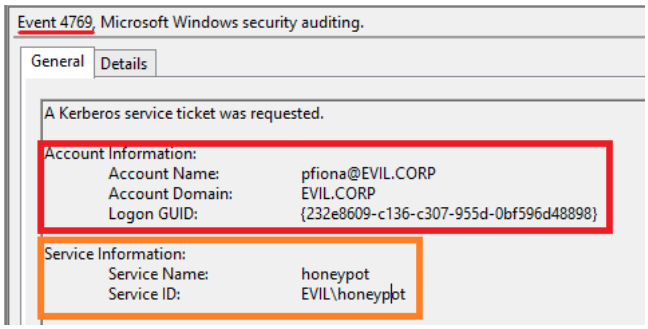
По большому счету, данный набор фильтров – достаточно редкое явление в бескрайних просторах сетевого трафика, а посему по праву может являться «красной тряпкой» для сотрудников подразделений ИБ и программно-аппаратных средств мониторинга сетевой активности. Более того, если мы взглянем на EventLog на контроллере домена, то увидим следующую картину.



Когда мы запрашиваем все Service Tickets для всех учетных записей, то тем самым генерируем множественные **event ID 4769** на контроллере домена. Согласитесь, как минимум выглядит это очень подозрительно…

Honeypot

К одному из способов борьбы с **Kerberoasting** относят создание Honeypot аккаунтов с установленным атрибутом **servicePrincipalName**, и отслеживания запросов к ним. Действительно, при таком прямолинейном подходе, который был продемонстрирован выше этот способ борьбы будет работать прекрасно.



Запрос ST учетной записи Honeypot

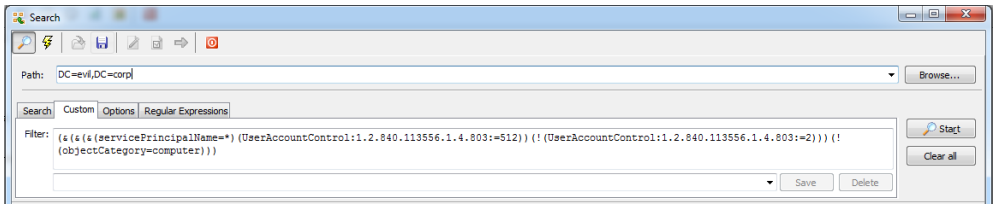
Как мы можем заметить, запрашивая **ServiceTicket** для всех учетных записей, имеющих SPN мы невольно захватили и ST для учетной записи **Honeypot**, при этом **Blue Team** специалист видит не только само событие, но также аккаунт, выполнявший его…

Дорогу осилит идущий

Кажется, мы начинаем понимать, что **самый легкий способ - прямой путь к провалу**! И понимая риски мы стараемся действовать аккуратнее.

Именно поэтому первым нашим действием будет подключение к контроллеру домена (например) через LDAPAdmin с последующим поиском представляющих для нас интерес SPNs.

1) **Используем для поиска учетных записей те же фильтры что и в Impacket-GetUserSPNs**

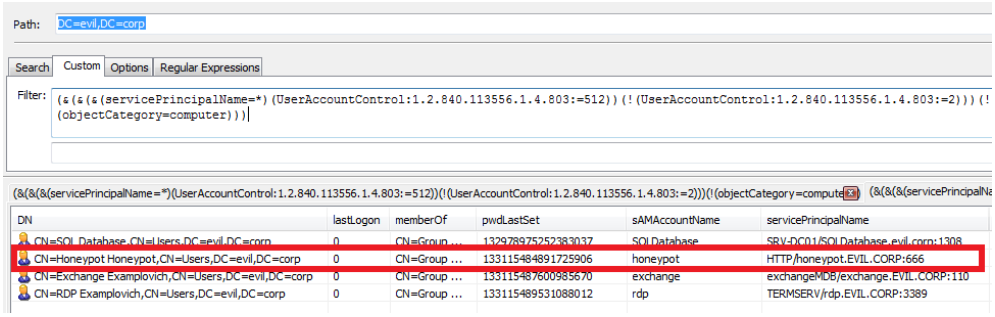


Можем заметить, что используются те же самые фильтры, что и в GetUserSPNs

Это лишь 50% успеха, так как таким образом, мы не запрашиваем TGS а лишь выполняем поиск в дереве. Сетевой трафик с точки зрения LDAP запроса не поменялся, но давайте взглянем на EventLog.

Очень ожидаемо - это ведь только LDAP запрос:)

Ожидаемая картина, так как билеты не запрашивались, то и нечему появляться! (Простите уж за столь очевидные комментарии). Но трафик все такой же грязный. Но почему же мы можем сказать, что это лишь 50% успеха??? Да, действительно, трафик выдает нас с потрохами, но уже сейчас мы можем начать отбор, интересующих нас учетных записей с SPN. А это в теории поможет нам ~~не вляпаться~~ обойти уловку с Honeypot.

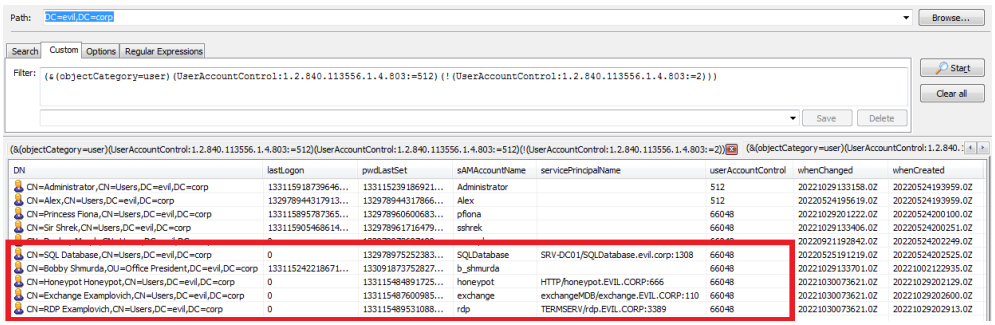


Та самая ловушка

Разумеется, это лишь демонстрация, и никто в здравом уме не будет называть учетную запись Honeypot. А поэтому без дополнительного анализа учетных записей нам не обойтись.

2) Ищем интересные учетки используя свои фильтры

Прелесть LDAPAdmin заключается в его наглядности (кэп). Мы уже сразу можем визуально выделить интересные названия и производить поиск в (кажущихся на первый взгляд интересными) OU. В примере я делаю поиск по всему домену. Но даже так, мы уже избавляемся от столь очевидного фильтра **(servicePrincipalName=*)**



Видоизменяем LDAP запрос и ищем интересные УЗ

Далее нам предстоит отбор только валидных учеток. Сделать это можно с использованием следующего набора атрибутов: **sAMAccountName, servicePrincipalName, userAccountControl, whenCreated, whenChanged, lastLo**

В любом случае, каждый уже сам разрабатывает стратегию поиска, это лишь пример)

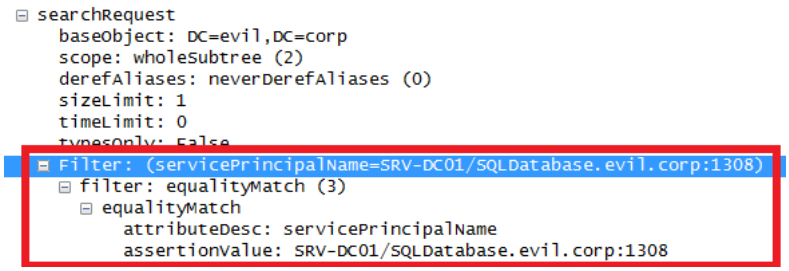
3) Запрос Service Ticket к интересующей нас учетной записи

Ну вот мы и определили нашу потенциальную жертву. А значит пришла пора запрашивать заветный Service Ticket. Сделать это можно с использованием Power Shell скрипта.

https://github.com/cyberark/RiskySPN/blob/master/Get-TGSCipher.ps1

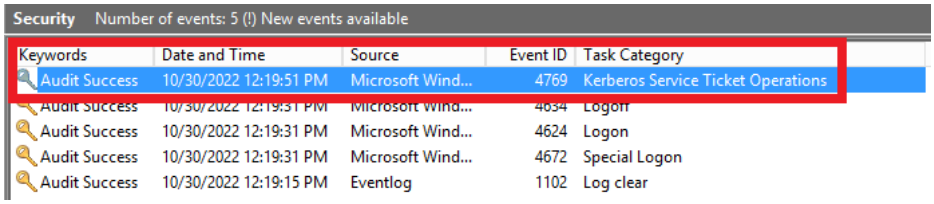
Get-TGSCipher -SPN "SRV-DC01/SQLDatabase.evil.corp" -Format Hashcat

Теперь посмотрим сперва на запрос с помощью Wireshark, а уже потом на EventLog



Согласитесь, выглядит это гораздо приятнее! И вполне может сойти за легитимный запрос.

А что же в EventLog, думаю результат будет ожидаемым)



И действительно, запрос ServiceTicket выполнялся для одной учетной записи!

В данной статье я не открывал Америку, а лишь подчеркнул, на мой взгляд один из очень важных принципов деятельности Red Team – скрытность. Думаю, что если есть шанс остаться незамеченным, то его нужно использовать! Источник Вдохновения.