Сбор информации: 1 Часть. – Telegraph

telegra.ph/Sbor-informacii-1-CHast-06-22

Life-Hack - Linux/Хакинг/Хакер/ИБ/Osint June 22, 2024

Представьте, что инженеры, разработавшие корпоративную сеть, сидят с вами за одним столом и демонстрируют вам огромную схему, из которой становится понятно строение зон и подсетей, расположение компонентов и почему сеть устроена именно так. Ваша задача на этапе сбора информации в ходе теста на проникновение заключается в том, чтобы максимально приблизиться к этому уровню понимания без помощи сетевых инженеров. Чем больше информации вы получите, тем выше ваши шансы обнаружить слабое место.

Пожалуй основной инструмент на этапе сбора информации - Nmap. Помимо проверки открытых/закрытых портов Nmap может идентифицировать сервис, слушающий открытый порт, и его версию, а иногда помогает определить ОС. В Nmap есть поддержка скриптов для сканирования (NSE — Nmap Scripting Engine). С использованием скриптов возможно проверить уязвимости для различных сервисов (если, конечно, для них есть скрипт, либо можно всегда написать свой) или побрутить пароли от различных сервисов.

Подробнее об этом замечательном инструменте мы уже писали ранее:

1) https://telegra.ph/NMAP-1-chast-05-07

2) https://telegra.ph/NMAP-2-chast-05-07

3) https://telegra.ph/NMAP-3-chast-05-07

Nmap позволяет составить подробную карту сети, получить максимум информации о запущенных сервисах на хостах в сети, а также превентивно проверить некоторые уязвимости. Nmap также имеет гибкие настройки сканирования, возможна

настройка скорости сканирования, количества потоков, количества групп для сканирования и т.д.

Бывают ситуации, когда приходится работать в сильно сегментированной сети и сканирование по маске 255.255.0.0 займет очень много времени. Рассмотрим способ решения этой проблемы.

```
Nmap -T4 -iL targets.txt
```

Это очень распространенная строка сканирования, которую многие люди используют для первоначальной разведки. На /24 вы можете получить достойные результаты за разумное количество времени с помощью этого метода. Для сканирования /16 CIDR-диапазона требуется другой подход.

```
nmap -vvv -Pn -sS -p 21-23,25,53,111,137,139,445,80,443,3389,5900,8080,8443 --min-rtt-timeout 275ms --max-rtt-timeout 350ms --max-retries 1 --max-scan-delay 0 --min-hostgroup 128 --min-rate 5500 -iL UpHosts.txt -oA Tuned_FullPort
```

-Pn: пропуск обнаружения хостов

-n: разрешение DNS отключено (при использовании с /16, может обойтись в несколько часов дополнительного сканирования). Можно выполнить позже, когда будет получен список активных хостов, и не терять так много времени.

-p: список портов

—min-rtt-timeout и —max-rtt-timeout: флаги отвечают за время отклика. Получив среднее время эха ICMP, можно добавить 25 мс для –min-rtt-timeout и 100 мс для –max-rtt-timeout. Судя по скриншоту ниже, при среднем времени пинга около 250 мс я могу установить минимум в 275 мс и максимум в 350 мс. Эти цифры являются приблизительной отправной точкой. Вполне возможно, что вы сможете немного сократить время, если результаты по-прежнему выглядят хорошо.

```
┌──(anorman㉿kali)-[~/Documents]
└─$ ping -c 5 123.59.211.123
PING 123.59.211.123 (123.59.211.123) 56(84) bytes of data.
64 bytes from 123.59.211.123: icmp_seq=1 ttl=128 time=256 ms
64 bytes from 123.59.211.123: icmp_seq=2 ttl=128 time=259 ms
64 bytes from 123.59.211.123: icmp_seq=3 ttl=128 time=259 ms
64 bytes from 123.59.211.123: icmp_seq=4 ttl=128 time=259 ms
```

—max-retries: сколько раз вы хотите, чтобы Nmap повторил попытку отправки пакета (я никогда не замечал, чтобы несколько повторных попыток давали мне лучшие результаты).

—max-scan-delay: еще один базовый и распространенный флаг задержки между пакетами. Так как мы собираемся действовать быстро, установим этот флаг в 0.

—min-hostgroup: этот флаг может помочь ускорить процесс, но только до определенного момента (мне нравится устанавливать значение 128, но при очень больших сканах я выбираю 512 или 1024, чтобы попытаться захватить большие группы).

–min-rate: это может быть опасно в зависимости от сети, в которой вы находитесь, и может привести к перегрузке, если вы не будете осторожны.

Конечным результатом таких настроек будет возможность отсканировать /16 примерно за 15 минут.