Столыпинский вестник

Научная статья

Original article

УДК 004.424

«ОБРАТНАЯ ОБОЛОЧКА» КАК ОДНА ИЗ УЯЗВИМОСТЕЙ ЦЕЛЕВОЙ СИСТЕМЫ

"REVERSE SHELL" AS ONE OF THE VULNERABILITIES OF THE TARGET SYSTEM

Крыгин Никита Дмитриевич, студент бакалавр, Донской государственный технический университет, г. Ростов-на-Дону (344003 Россия г. Ростов-на-Дону, Гагарина 1), krigginnn@rambler.ru

Krygin Nikita Dmitrievich, bachelor student, Don State Technical University, Rostov-on-Don (344003 Russia, Rostov-on-Don, Gagarina 1), krigginnn@rambler.ru

Аннотации: Обратная оболочка, также известная как удаленная оболочка или «оболочка с обратным подключением», использует уязвимости целевой системы, чтобы инициировать сеанс оболочки, а затем получить доступ к компьютеру жертвы. Цель состоит в том, чтобы подключиться к удаленному компьютеру и перенаправить входные и выходные соединения оболочки целевой системы, чтобы злоумышленник мог получить к ней удаленный доступ.

Annotations. The reverse shell, also known as remote shell or "reverse shell", exploits vulnerabilities in the target system to initiate a shell session and then gain access to the victim's computer. The goal is to connect to a remote computer and

redirect input and output shell connections to the target system so that an attacker can access it remotely.

Ключевые слова: уязвимость, информационная безопасность, злоумышленник, анализ портов.

Keywords: vulnerability, information security, intruder, port analysis

Обратные оболочки позволяют злоумышленникам открывать порты к целевым машинам, форсируя связь и обеспечивая полный захват целевой машины. Поэтому это серьезная угроза безопасности. Этот метод также широко используется в тестах на проникновение.

При стандартной атаке удаленной оболочки злоумышленники подключают машину, которой они управляют, к удаленному сетевому хосту цели, запрашивая сеанс оболочки. Эта тактика известна как бинд шелл. Злоумышленники могут использовать обратную оболочку, если удаленный хост недоступен публично (например, из-за защиты брандмауэра или непубличного IP-адреса). Целевая машина инициирует исходящее соединение при атаке с обратной оболочкой и устанавливает сеанс оболочки с прослушивающим сетевым узлом.

Для хостов, защищенных трансляцией сетевых адресов (NAT), может потребоваться обратная оболочка для удаленного обслуживания. Хотя у обратных оболочек есть законное применение, киберпреступники также используют их для проникновения на защищенные хосты и выполнения команд операционной системы. Обратные оболочки позволяют злоумышленникам обходить механизмы сетевой безопасности, такие как брандмауэры.

Злоумышленники могут использовать возможности обратной оболочки с помощью фишинговых писем или вредоносных веб-сайтов. Если жертва устанавливает вредоносное ПО на локальную рабочую станцию, оно инициирует исходящее соединение с командным сервером злоумышленника. Исходящее соединение часто бывает успешным, потому что брандмауэры обычно фильтруют входящий трафик.

Злоумышленник может использовать уязвимости внедрения команд на сервере, чтобы скомпрометировать систему. Во внедренном коде сценарий обратной оболочки предоставляет командную оболочку, позволяющую выполнять дополнительные вредоносные действия.

Пример: обратная оболочка Python

Чтобы понять, как работает обратная оболочка, рассмотрим фрагмент кода, который можно использовать для установки удаленной оболочки на Python:

```
s = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
s.connect(("0.0.0.0", 7777))
os.dup2(s.fileno(), 0)
os.dup2(s.fileno(), 1)
os.dup2(s.fileno(), 2)
p = subprocess.call(["/bin/sh", "-i"])
```

Установление соединения

Эти две строки используются для установления соединения с модулем сокетов Python. Он создает сокет с адресом IPv4, который обменивается данными через TCP.

```
s = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
```

Эта строка указывает, какой IP-адрес и порт должен прослушивать сокет:

```
s.connect(("0.0.0.0", 7777))
```

Перезапись файловых дескрипторов

Интерфейс командной строки Python использует три потока данных для обработки команд оболочки: stdin для входных данных, stdout для выходных данных и stderr для сообщений об ошибках. Внутри они обозначаются как 0, 1 и 2.

Код оболочки теперь использует команду dup2 модуля Python os, которая взаимодействует с операционной системой.

Следующая команда берет дескриптор файла, сгенерированный предыдущей командой сокета, и дублирует его три раза, перезаписывая потоки

данных stdin, stdout и stderr созданным нами обратным сокетом оболочки. s.fileno() относится к файловому дескриптору сокета.

os.dup2(s.fileno(), 0)

os.dup2(s.fileno(), 1)

os.dup2(s.fileno(), 2)

После запуска этих команд три потока данных CLI перенаправляются в новый сокет и больше не обрабатываются локально.

Создание оболочки

Завершающим этапом атаки является запуск модуля подпроцесса Python. Это позволяет обратной оболочке запускать программу как подпроцесс сокета. Подпроцесс. Команда call позволяет нам передать любую исполняемую программу. Передавая /bin/sh, апускаем оболочку Bash как подпроцесс созданного нами сокета.

p = subprocess.call(["/bin/sh", "-i"])

В этот момент оболочка становится интерактивной — любые данные, записанные в оболочку, будут записаны в терминал и прочитаны через терминал, как если бы это была основная системная оболочка. Теперь можно установить обратное соединение с компьютером злоумышленника и позволить ему удаленно выполнять команды на целевой машине.

Соединения с обратной оболочкой часто являются вредоносными, если вы не настроили их явно для целей удаленного администрирования. С точки зрения сервера трудно заблокировать все обратные соединения оболочки при использовании сетевой системы, такой как сервер. Следующие шаги могут помочь вам повысить безопасность вашей системы и снизить риск:

Заблокируйте все исходящие соединения, кроме определенных портов и удаленных IP-адресов для необходимых служб. Для этого используйте песочницу или запускайте свои серверы в минимальных контейнерах.

Настройте прокси-сервер с ограниченными пунктами назначения и строгим контролем. Риск невозможно исключить, учитывая, что

злоумышленники могут создавать обратные подключения оболочки через DNS, но такое усиление может минимизировать риск.

Удалите ненужные интерпретаторы и инструменты, чтобы ограничить выполнение обратного шелл-кода и усложнить злоумышленникам возможность взлома вашей системы. Этот подход не всегда является жизнеспособным решением, так как он практичен только для узкоспециализированных и защищенных серверов, в то время как злоумышленники все еще могут найти работающий сценарий оболочки.

Предотвратите эксплойты, такие как уязвимости внедрения кода. Злоумышленники обычно выполняют сценарии оболочки, используя существующую уязвимость, связанную с внедрением кода, а затем получают привилегии root. Регулярно обновляйте свои веб-приложения и серверы и используйте надежный сканер уязвимостей для их проверки.

Можно использовать большое количество методов для защиты сервера. Дополнительным подходом к предотвращению обратной оболочки является блокирование вредоносного сетевого взаимодействия. Брандмауэры веб-приложений (WAF) и решения для самозащиты приложений во время выполнения (RASP) могут обнаруживать шаблоны связи, которые выглядят как соединение с обратной оболочкой, и блокировать их.

Литература

1. Бабаш, А.В. Информационная безопасность: Лабораторный практикум / А.В. Бабаш, Е.К. Баранова, Ю.Н. Мельников. - М.: КноРус, 2019. - 432 с.
2. Бабаш, А.В. Информационная безопасность. Лабораторный практикум: Учебное пособие / А.В. Бабаш, Е.К. Баранова, Ю.Н. Мельников. - М.: КноРус, 2013. - 136 с.
3. Баранова, Е.К. Информационная безопасность и защита информации: Учебное пособие / Е.К. Баранова, А.В. Бабаш. - М.: Риор, 2017. - 400 с.
4. Баранова, Е.К. Информационная безопасность и защита информации: Учебное пособие / Е.К. Баранова, А.В. Бабаш. - М.: Риор, 2017. - 476 с.

5. Баранова, Е.К. Информационная безопасность и защита информации: Учебное пособие / Е.К. Баранова, А.В. Бабаш. - М.: Риор, 2018. - 400 с.
6. Баранова, Е.К. Информационная безопасность. История специальных методов криптографической деятельности: Учебное пособие / Е.К. Баранова, А.В. Бабаш, Д.А. Ларин. - М.: Риор, 2008. - 400 с.
7. Бирюков, А.А. Информационная безопасность: защита и нападение / А.А. Бирюков. - М.: ДМК Пресс, 2013. - 474 с.
8. Гафнер, В.В. Информационная безопасность: Учебное пособие / В.В. Гафнер. - Рн/Д: Феникс, 2010. - 324 с.

References

1. Babash, A.V. Information security: Laboratory workshop / A.V. Babash, E.K. Baranova, Yu.N. Melnikov. - M.: KnoRus, 2019. - 432 p.
2. Babash, A.V. Information Security. Laboratory workshop: Textbook / A.V. Babash, E.K. Baranova, Yu.N. Melnikov. - M.: KnoRus, 2013. - 136 p.
3. Baranova, E.K. Information security and information protection: Textbook / E.K. Baranova, A.V. Babash. - M.: Rior, 2017. - 400 p.
4. Baranova, E.K. Information security and information protection: Textbook / E.K. Baranova, A.V. Babash. - M.: Rior, 2017. - 476 p.
5. Baranova, E.K. Information security and information protection: Textbook / E.K. Baranova, A.V. Babash. - M.: Rior, 2018. - 400 p.
6. Baranova, E.K. Information Security. History of special methods of cryptographic activity: Textbook / E.K. Baranova, A.V. Babash, D.A. Larin. - M.: Rior, 2008. - 400 p.
7. Biryukov, A.A. Information security: protection and attack / A.A. Biryukov. - M.: DMK Press, 2013. - 474 p.
8. Gafner, V.V. Information Security: Textbook / V.V. Gafner. - Rn / D: Phoenix, 2010. - 324 p.

Для цитирования: Крыгин Н.Д. «ОБРАТНАЯ ОБОЛОЧКА» КАК ОДНА ИЗ УЯЗВИМОСТЕЙ ЦЕЛЕВОЙ СИСТЕМЫ // Научный сетевой журнал «Столыпинский вестник», номер 4/2022.