Изучаем Adversarial Tactics, Techniques & Common Knowledge (ATT@CK). Enterprise Tactics. Часть 4

habr.com/ru/articles/428602

Андрей Макеев

Повышение привилегий (Privilege Escalation)

Ссылки на все части:

Эскалация привилегий — это результат действий, которые позволяют злоумышленнику или вредоносной программе получить более высокий уровень разрешений в атакуемой системе или сети. Техники эскалации привилегий описывают методы, с помощью которых противник, получив непривилегированный доступ в атакуемую систему, используя различные «слабости» системы может получить права локального администратора, system или root. Использование злоумышленниками учетных записей пользователей с правами доступа к конкретным системам или разрешениями на выполнения определенных операций также может рассматриваться как эскалация привилегий.

Автор не несет ответственности за возможные последствия применения изложенной в статье информации, а также просит прощения за возможные неточности, допущенные в некоторых формулировках и терминах. Публикуемая информация является свободным пересказом содержания MITRE ATT&CK.

Важно отменить, что некоторые техники, описываемые в матрице ATT@CK, одновременно включены в несколько этапов цепочки атаки, например, перехват поиска DLL может использоваться как для закрепления доступа путём несанкционированного выполнения вредоносной DLL, так и для повышения привилегий путём запуска DLL в процессе, работающем в контексте более привилегированного пользователя.

Манипуляции с маркерами доступа (Access Token Manipulation)

Система: Windows
Права: Пользователь, администратор
Описание: Злоумышленники могут использовать маркеры доступа (Access Token), чтобы совершать свои действия в различных пользовательских или системном контекстах безопасности, таким образом избегая обнаружения вредоносной активности. Противник может использовать функции Windows API для копирования маркеров доступа из

существующих процессов (Token stealing), для этого он должен находиться в контексте привилегированного пользователя (например, администратора). Кража маркеров доступа обычно используется для повышения привилегий с уровня администратора до уровня System. Противник также может использовать маркер доступа учетной записи для аутентификации в удаленной системе, если у этой учетной записи есть нужные разрешения в удаленной системе.
Рассмотрим несколько способов злоупотребления маркерами доступа:

- Кража и олицетворение токенов. Олицетворение токенов — это способность ОС запускать потоки в контексте безопасности, отличном от контекста процесса, которому принадлежит этот поток. Другими словами, олицетворение токенов позволяет совершать какие-либо действия от имени другого пользователя. Противник может создать дубликат маркера доступа с помощью функции *DuplicateTokenEX* и использовать *ImpersonateLoggedOnUser*, чтобы вызвать поток в контексте залогиненного пользователя или использовать *SetThreadToken*, чтобы назначить маркер доступа в поток.
- Создание процесса с помощью маркера доступа. Злоумышленник может создать маркер доступа с помощью функции *DuplicateTokenEX* и далее использовать его с *CreateProcessWithTokenW* для создания нового процесса, работающего в контексте олицетворяемого пользователя.
- Получение и олицетворение маркеров доступа. Противник, имея логин и пароль пользователя, может создать сеанс входа в систему с помощью API-функции *LogonUser*, которая вернёт копию сессионного маркера доступа нового сеанса, и далее с помощью функции *SetThreadToken* назначить маркер для потока. Metasploit Meterpreter и CobaltStrike имеют инструментарий для манипуляций с маркерами доступа с целью повышения привилегий.

Рекомендации по защите: Чтобы в полной мере использовать вышеописанную тактику злоумышленник должен обладать правами администратора системы, поэтому не забывайте ограничивать привилегии обычных пользователей. Любой пользователь может обмануть маркеры доступа если у него есть легитимные учетные данные. Ограничьте возможность создания пользователями и группами маркеров доступа:
GPO: Computer Configuration > [Policies] > Windows Settings > Security Settings > Local Policies > User Rights Assignment: Create a token object
Так же определите, кто может заменять маркеры процессов локальных или сетевых служб:
GPO: Computer Configuration > [Policies] > Windows Settings > Security Settings > Local Policies > User Rights Assignment: Replace a process level token

Модификация исполняемых файлов приложений «специальные возможности Windows» (Accessibility Features)

Система: Windows
Права: Администратор
Описание: Приложения «специальные возможности» (экранная лупа, экранная клавиатура и т.п.) могут запускаться с помощью комбинаций клавиш до входа пользователя в систему. Злоумышленник может подменить файлы запуска этих программ

или изменить способ их запуска и открыть командную консоль или получить бэкдор без входа в систему.

- *C:\Windows\System32\sethc.exe* — запускается 5-кратным нажатием клавиши Shift;
- *C:\Windows\System32\utilman.exe* — запускается нажатием комбинации Win+U.

В WinXP и более поздних версиях sethc.exe и utilman.exe могут быть заменены, например, на cmd.exe, впоследствии при нажатии нужной комбинации клавиш cmd.exe запуститься до входа в Windows с привилегиями System.
В Vista и более поздних версиях нужно изменить ключ реестра, который настраивает cmd.exe или другую программу в качестве отладчика, например, для ultiman.exe. После правки реестра и нажатии нужной комбинации клавиш на экране входа в систему или при подключении к хосту по RDP выполнится cmd.exe с правами System.
Есть ещё программы Windows, которые могут использоваться при реализации данной техники атаки:

- *C:\Windows\System32\osk.exe;*
- *C:\Windows\System32\Magnify.exe;*
- *C:\Windows\System32\Narrator.exe;*
- *C:\Windows\System32\DisplaySwitch.exe;*
- *C:\Windows\System32\AtBroker.exe.*

Рекомендации по защите: Настройте запуск обязательной сетевой аутентификации удаленных пользователей до создания RDP-сеанса и отображения экрана входа в систему (*включено по умолчанию в Windows Vista и более поздних версиях*). Используйте Remote Desktop Gateway для управления соединениями и настройкой безопасности RDP.

Модификация ключа AppCert DLLs

Система: Windows
Права: Администратор, System
Описание: Библиотеки DLL, указанные в значении ключа AppCertDLLs загружаются в каждый процесс, который вызывает часто используемые функции API: *CreateProcess, CreateProcessAsUser, CreateProcessWithLoginW, CreateProcessWithTokenW, WinExec*. Значением ключа AppCertDLLs можно злоупотреблять, вызвав загрузку вредоносной DLL и запустив определенные процессы. AppCertDLLs хранится в следующем разделе реестра:
HKEY_LOCAL_MACHINE\System\CurrentControlSet\Control\Session Manager.

Рекомендации по защите: Применяйте всевозможные средства блокировки потенциально-опасного программного обеспечения и загрузки неизвестных DLL-библиотек, например AppLocker и DeviceGuard.

Модификация ключа AppInit DLLs

Система: Windows
Права: Администратор, System

Описание: DLL-библиотеки, указанные в значении ключа AppInit_DLLs, загружаются в каждый процесс, который загружает user32.dll. На практике, это почти каждая программа. AppInit_DLLs хранится в следующих разделах реестра:

- *HKEY_LOCAL_MACHINE\Software\Microsoft\Windows NT\CurrentVersion\Windows;*
- *HKEY_LOCAL_MACHINE\Software\Wow6432Node\Microsoft\Windows NT\CurrentVersion\Windows.*

Значением ключа AppInit_DLLs можно злоупотреблять для превышения привилегий, загружая вредоносные DLL и запуская определенные процессы. Функциональность AppInit_DLLs отключена в Windows 8 и более поздних версиях, когда активирована безопасная загрузка.

Рекомендации по защите: Рассмотрите возможность использования ОС версии не ранее Windows 8 и включения безопасной загрузки. Применяйте всевозможные средства блокировки потенциально-опасного программного обеспечения и загрузки неизвестных DLL-библиотек, например AppLocker и DeviceGuard.

Злоупотребление подсистемой совместимости приложений (Application Shimming)

Система: Windows
Права: Администратор
Описание:Microsoft Windows Application Compatibility Infrastructure/Framework создана для обеспечения совместимости программ с обновлениями Windows и изменениями кода ОС. Система совместимости использует так называемые shim («прокладки») — библиотеки, выступающие в качестве буфера между программой и ОС. С помощью shim-кэша система определяет необходимость использования shim-прокладок (хранятся в виде БД типа .sdb). В файлах .sdb хранятся различные процедуры для перехвата кода приложения, его обработки и дальнейшего перенаправления в ОС. Перечень всех shim-прокладок, установленных установщиком (sdbinst.exe) по умолчанию храниться в:

- *%WINDIR%\AppPatch\sysmain.sdb*;
- *HKLM\Software\Microsoft\Windows NT\CurrentVersion\AppCompatFlags\InstalledSDB*.

Кастомные shim-базы хранятся в:

- *%WINDIR%\AppPatch[64]\Custom;*
- *HKLM\Software\Microsoft\Windows NT\CurrentVersion\AppCompatFlags\Custom*.

Для обеспечения защиты в пользовательском режиме исключена возможность изменения ядра ОС с помощью shim-прокладок, а для их установки необходимы права администратора. Однако некоторые shim-прокладки могут использоваться для обхода контроля учетных записей (UAC), DLL-инъекций, отключения Data Execution Prevention и Srtucture Exception Handling, а так же перехвата адресов памяти. Использование злоумышленником shim-прокладок позволяет повысить привилегии, установить бэкдоры, отключить защиту ОС, например Защитник Windows.

Рекомендации по защите: Способов предотвращения Application shiming не так много. Отключение совместимости приложений не рекомендуется во избежание проблем со стабильностью работы ОС. Microsoft выпустила KB3045645, которое удалит флаг «auto-elevate» в файле sdbinst.exe для предотвращения использования shim-системы для обхода UAC.

Обход контроля учетных записей (Bypass User Account Control)

Система: Windows
Права: Пользователь, администратор
*Описание: И*звестно множество способов обхода UAC, самые распространенные из которых реализованы в проекте UACMe. Регулярно обнаруживаются новые способы обхода UAC, подобные злоупотреблению системным приложением eventvwr.exe, которое может выполнить бинарный файл или скрипт с повышенными правами. Вредоносные программы также могут быть внедрены в доверенные процессы, которым UAC разрешает повышение привилегий не запрашивая пользователя.
Для обхода UAC с помощью eventvwr.exe в реестре Windows модифицируется ключ: *[HKEY_CURRENT_USER]\Software\Classes\mscfile\shell \open\command*.
Для обхода UAC с помощью sdclt.exe в реестре Windows модифицируются ключи: *[HKEY_CURRENT_USER]\Software\Microsoft\Windows\CurrentVersion\App Paths\control.exe; [HKEY_CURRENT_USER]\Software\Classes\exefile\shell\runas\command\isolatedCommand.*

Рекомендации по защите: Удаляйте пользователей из локальной группы администраторов в защищаемых системах. По возможности включите в параметрах UAC наивысший уровень защиты.

Перехват поиска DLL (DLL Search Order Hijacking)

Система: Windows
Права: Пользователь, Администратор, System
Описание: Техника заключается в эксплуатации уязвимостей алгоритма поиска приложениями файлов DLL, необходимых им для работы (MSA2269637). Зачастую директорией поиска DLL является рабочий каталог программы, поэтому злоумышленники могут подменять исходную DLL на вредоносную с тем же именем файла.
Удаленные атаки на поиск DLL могут проводиться когда программа устанавливает свой текущий каталог в удаленной директории, например, сетевую шару. Также злоумышленники могут напрямую менять способ поиска и загрузки DLL заменяя файлы .manifest или .local, в которых описываются параметры поиска DLL. Если атакуемая программа работает с высоким уровнем привилегий, то подгруженная ею вредоносная DLL также будет выполняться с высокими правами. В этом случае техника может использоваться для повышения привилегий от пользователя до администратора или System.

Рекомендации по защите: Запрет удаленной загрузки DLL (включено по умолчанию в Windows Server 2012+ и доступно с обновлениями для XP+ и Server 2003+). Включение безопасного режима поиска DLL, который ограничит каталоги поиска директориями типа *%SYSTEMROOT%* до выполнения поиска DLL в текущей директории приложения.

Включение режима безопасного поиска DLL:
Computer Configuration > [Policies] > Administrative Templates > MSS (Legacy): MSS: (SafeDllSearchMode) Enable Safe DLL search mode.
Соответствующий ключ реестра:
HKLM\SYSTEM\CurrentControlSet\Control\Session Manager\SafeDLLSearchMode.

Рассмотрите целесообразность аудита защищаемой системы для устранения недостатков DLL с помощью таких инструментов как модуль PowerUP в PowerSploit. Не забывайте про блокировку вредоносного и потенциально-опасного ПО, а так же выполнение рекомендаций Microsoft.

Перехват поиска Dylib (Dylib Hijacking)

Система: macOS
Права: Пользователь
Описание: Техника основана на уязвимостях алгоритмов поиска динамических библиотек dylib в macOS и OS X. Суть заключается в определении dylib, которые подгружает атакуемое приложение и последующем размещении вредоносной версии dylib с тем же именем в рабочей директории приложения. Это приведёт к загрузке приложением dylib, которая размещена в рабочем каталоге программы. При этом вредоносная Dylib будет выполнятся с правами доступа атакуемого приложения.

Рекомендации по защите: Запрет записи пользователями файлов в каталоги поиска dylib. Аудит уязвимостей с помощью Dylib Hijacking Scanner от Objective-See.

Эксплуатация уязвимостей для повышения привилегий (Exploitation for Privilege Escalation)

Система: Windows, Linux, macOS
Права: Пользователь
Описание: Противники могут повысить привилегии в атакуемой системе используя уязвимости в программном обеспечении.

Рекомендации по защите: Регулярное обновление ПО на всех защищаемых рабочих станциях, серверах, сетевом оборудовании и других устройствах подключенных к защищаемой сети. Проводите анализ типов угроз, уязвимостей, программ-эксплойтов, которые можно использовать против защищаемой организации. Так же рекомендуется применение систем защиты от эксплойтов, например, Windows Defender Exploit Guard (WDEG) для Windows 10 или Enhanced Mitigation Experience Toolkit (EMET) для более ранних версий Windows.

h3EWM-инъекции (Extra Window Memory Injection)
Система: Windows
Права: Администратор, System
Описание: Техника заключается в злоупотреблении дополнительной памятью окна Windows, так называемой Extra Window Memory (EWM). Размер EWM — 40 байт, подходит для хранения 32-битного указателя и часто используется для указания ссылки на

процедуры. Вредоносные программы в ходе цепочки атаки, могут размещать в EWM указатель на вредоносный код, который в последствие будет запущен процессом инфицированного приложения.

Рекомендации по защите: Учитывая, что техники EWM-инъекций основаны на злоупотреблении функциями разработки ОС усилия по защите необходимо направить на предотвращение запуска вредоносных программ и инструментов злоумышленников. Хорошей практикой является выявление и блокирование потенциально-опасного ПО с помощью AppLocker, организации белого списка приложений или применения политик ограничения программного обеспечения Software Restriction Policies.

Недостатки разрешений на уровне файловой системы (File System Permissions Weakness)

Система: Windows
Права: Пользователь, Администратор
Описание: Суть техники заключается в подмене исполняемых файлов, которые автоматически запускаются различными процессами (например, при загрузке ОС или в определенное время, в случае если права на исполняемые файлы настроены неверно). После подмены вредоносный файл будет запущен с правами процесса, таким образом если процесс имеет более высокий уровень доступа злоумышленник сможет осуществить эскалацию привилегий. В рамках данной техники злоумышленники могут пытаться манипулировать двоичными файлами служб Windows.
Другой вариант атаки связан с недостатками алгоритмов в работе самораспаковывающихся установщиков. В процессе инсталляции ПО, установщики зачастую распаковывают различные полезные файлы, в том числе .dll и .exe, в каталог %TEMP%, при этом они могут не устанавливать соответствующие разрешения для ограничения доступа к распаковываемым файлам, что позволяет злоумышленникам совершать подмену файлов и, как следствие, повысить привилегии или обойти контроль учетных записей, т.к. некоторые установщики выполняются с расширенными правами.

Рекомендации по защите: Ограничение прав учетных записей, чтобы только администраторы могли управлять службами и взаимодействовать с бинарными файлами, используемыми службами. Отключение в UAC возможности повышения привилегий для стандартных пользователей. Параметры UAC хранятся в следующем разделе реестра:

[HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Policies\System].

Для автоматического отклонения запросов на повышение привилегий необходимо добавить ключ:

«ConsentPromptBehaviorUser»=dword:00000000.

Для контроля работы установщиков необходимо добавить ключ:

«EnableInstallerDetection»=dword:00000001, который будет требовать ввода пароля для установки программ.

Перехват вызовов функций Windows API (Hooking)

Система: Windows
Права: Администратор, System
Описание: API функции Windows обычно хранятся в DLL-библиотеках. Техника Hooking заключается в перенаправлении вызовов API-функций посредством:

- Hook-процедур — встроенных в ОС процедур, которые выполняют код при вызове различных событий, например, нажатие клавиш или перемещение мыши;
- Модификации адресной таблицы (IAT), в которой хранятся указатели на API-функции. Это позволит «обмануть» атакуемое приложение, заставив его запустить вредоносную функцию;
- Непосредственного изменения функции (сплайсинг), в ходе которого меняются первые 5 байт функции, вместо которых вставляется переход на вредоносную или иную функцию, определенную злоумышленником.

Подобно инъекциям, злоумышленники могут использовать hooking для исполнения вредоносного кода, маскировки его выполнения, доступа к памяти атакуемого процесса и повышения привилегий. Злоумышленники могут захватывать вызовы API, включающие параметры, содержащие аутентификационные данные. Hooking обычно применяется руткитами для скрытия вредоносной активности в системе.

Рекомендации по защите: Перехват событий в ОС является частью нормальной работы системы, поэтому какое либо ограничение данной функциональности может негативно влиять на стабильность работы законных приложений, например антивирусного ПО. Усилия по предотвращению применения техник перехвата необходимо сосредоточить на более ранних этапах цепочки атаки. Обнаружить вредоносную hooking-активность можно с помощью мониторинга вызовов функций SetWindowsHookEx и SetWinEventHook, использования детекторов руткитов, анализа аномального поведения процессов.

IFEO-инъекции (Image File Execution Options Injection)

Система: Windows
Права: Администратор, System
Описание: Механизм Image File Execution Options (IFEO) позволяет запускать вместо программы её отладчик, заранее указанный разработчиком в реестре:

- *HKLM\Software\Microsoft\Windows NT\CurrentVersion\Image File Execution Options/[executable]*
- *HKLM\SOFTWARE\Wow6432Node\Microsoft\Windows NT\CurrentVersion\Image File Execution Options\[executable]*, где *[executable]* — это исполняемый двоичный файл отладчика.

Подобно инъекциям, значением *[executable]* можно злоупотреблять запуская произвольный код, чтобы повысить привилегии или закрепиться в системе. Вредоносные программы могут использовать IFEO на для обхода защиты, регистрируя отладчики, которые перенаправляют и отклоняют различные системные приложения и приложения безопасности.

Рекомендации по защите: Описываемая техника основана на злоупотреблении штатными средствами разработки ОС, поэтому какие-либо ограничения могут вызвать нестабильность работы законного ПО, например, приложений безопасности. Усилия по предотвращению применения техники IFEO-инъекций необходимо сосредоточить на более ранних этапах цепочки атаки. Обнаружить подобную атаку можно с помощью мониторинга процессов с флагами *Debug_process* и *Debug_only_this_process*.

Запуск демонов (Launch Daemon)>

Система: macOS
Права: Администратор
Описание: Техника заключается в изменении злоумышленником параметров сервисов системного уровня запуска — Launch Daemon, указанных в plist-файлах. При загрузке системы процесс Launchd загружает параметры сервисов (демонов) из plist-файлов расположенных в следующих директориях:

- /System/Library/LaunchDeamons;
- /Library/LaunchDeamons.

Launch Daemon могут создаваться с администраторскими привилегиями, но выполнятся под учетной записью root, таким образом злоумышленник может реализовать эскалацию привилегий. Разрешения plist-файлов должны быть root:while, однако сценарий или программа, указанные в нём, могут иметь менее строгие разрешения. Поэтому злоумышленник может изменить исполняемые файлы, указанные в plist, и, таким образом, модифицировать текущие системные сервисы для закрепления в системе или эскалации привилегий.

Рекомендации по защите: Ограничьте привилегии пользователей, чтобы только авторизованные администраторы могли создавать Launch Daemon. Рассмотрите возможность мониторинга создания в системе plist-файлов с помощью таких приложений как KnockKnock.

Новые службы (New Service)

Система: Windows
Права: Администратор, System
Описание: Имя доступ в систему, злоумышленники могут создавать новые службы и настраивать их автоматический запуск. Имя службы может быть замаскировано с использованием имён, характерных для операционной системы. Службы могут быть созданы с привилегиями администратора, но запускаться от имени System. Сервисы могут создаваться из командной строки, с помощью средств удаленного доступа с функциями взаимодействия с Windows API или с помощью стандартных средств управления Windows и PowerShell.

Рекомендации по защите: Ограничьте права пользователей на создание новых служб, чтобы только уполномоченные администраторы могли это делать. Применяйте AppLocker и Software Restriction Policy.

Перехват пути (Path Interception)

Система: Windows
Права: Пользователь, администратор, system
Описание: Техника перехвата пути заключается в помещении исполняемого файла в директорию, из которой приложение запустит его вместо целевого файла. Атакующий может использовать следующие методы:

- Несуществующие пути. Пути к исполняемым файлам служб хранятся в ключах реестра и могут иметь один или несколько пробелов, например, *C:\Program Files\service.exe*, если атакующий создаст в системе файл *C:\Program.exe*, то Windows при обработке пути запустит его вместо целевого файла службы.
- Неправильная конфигурация переменных окружения. Если в переменной PATH путь *C:\example* предшествует *c:\Windows\System32* и существует файл *C:\example\net.exe*, то при вызове команды net, будет выполнен *C:\example\net.exe*, а не *c:\Windows\System32\net.exe*.
- Перехват порядка поиска (Search order hijacking). Когда не задан полный путь к исполняемому файлу, Windows, как правило, ищет файл с указанным именем в текущем каталоге, затем осуществляет поиск в системных каталогах. Например, файл «example.exe» при выполнении запускает cmd.exe с аргументами для выполнения команды net use. Атакующий может поместить в каталог расположения example.exe файл net.exe и он будет запущен вместо утилиты *c:\Windows\System32\net.exe*. Кроме того, если атакующий поместит файл net.com в каталог с файлом net.exe, то Windows выполнит net.com в соответствии с порядком исполнения, определенном в системной переменной PATHEXT.

Перехват порядка поиска файлов также применяется для выполнения DLL с помощью техники DLL Search Hijacking.

Рекомендации по защите: Выделите кавычками пути, указанные в файлах конфигураций, сценариях, переменной PATH, настройках служб и ярлыках. Помните о порядке поиска исполняемых файлов и используйте только полные пути. Выполните очистку старых ключей реестра, оставшихся от удаленного ПО, чтобы в реестре не осталось ключей, указывающих на несуществующие файлы. Установите запрет на запись пользователями системы в корневой каталог C:\ и системные каталоги Windows, ограничивайте права на запись в каталоги с исполняемыми файлами.

Модификация файлов Plist (Plist Modification)

Система: macOS
Права: Пользователь, Администратор
Описание: Злоумышленники могут модифицировать plist-файлы, указывая в них собственный код для его исполнения в контексте другого пользователя. Файлы свойств plist, расположенные в */Library/Preferences* выполняются с повышенными привилегиями, а plist из *~/Library/Preferences* выполняются с привилегиями пользователя.

Рекомендации по защите: Предотвратите изменение файлов plist, сделав их доступными только на чтение.

Модификация Port Monitors в Диспетчере печати (Port Monitors)

Система: Windows
Права: Администратор, System
Описание: Атакующий может организовать выполнение произвольной DLL от имени System при каждой загрузке Windows с помощью злоупотребления настройками Диспетчера печати (Spoolsv.exe). Для взаимодействия с устройствами печати Spoolsv.exe использует так называемые мониторы порта (port monitor) — это DLL-библиотеки, с помощью которых посредством LAN, USB, LPT или COM-интерфейса на устройства печати передаются низкоуровневые команды. Вышеописанные DLL хранятся в *C:\windows\system32* и регистрируются в реестре: *HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Print\Monitors*.
Port Monitor можно установить с помощью API функции AddMonitor или напрямую через редактирование вышеуказанного раздела реестра.

Рекомендации по защите: Организуйте блокирование потенциально-опасного ПО и применяйте инструменты контроля запуска приложений.

Инъекция кода в процесс (Process Injection), Ten Process Injection Techniques

Система: Windows, Linux, macOS
Права: Пользователь, администратор, system, root
Описание: Процессные инъекции — это метод выполнения произвольного кода в адресном пространстве отдельно живущего процесса. Запуск кода в контексте другого процесса позволяет получить доступ к памяти инъецируемого процесса, системным/сетевым ресурсам и, возможно, повышенные привилегии. Процессные инъекции также могут использоваться во избежание возможного обнаружения вредоносной активности средствами безопасности. Техники реализации инъекций в процессы основны на злоупотреблении различными механизмами, обеспечивающими многопоточность выполнения программ в ОС. Далее рассмотрены некоторые подходы к выполнению инъекции кода в процесс.

Windows
• DLL-инъекции. Выполняются посредством записи пути к вредоносной DLL внутрь процесса с её последующим выполнением путём создания удаленного потока (Remote thread — поток, который работает в виртуальном адресном пространстве другого процесса). Иными словами, вредоносное ПО записывает на диск DLL, а затем использует функцию подобную CreateRemoteTread, с помощью которой будет вызвана функция LoadLibrary в инъецируемом процессе.
• PE-инъекции (Portable executable injection) основаны на злоупотреблении особенностями выполнения в памяти PE-файлов, таких как DLL или EXE. Вредоносный код записывается в процесс без записи каких-либо файлов на диске, а затем с помощью дополнительного кода или путем создания удаленного потока вызывается его исполнение.

• Захват выполнения потока (Thread execution hijacking) включает в себя инъекции вредоносного кода или пути к DLL напрямую в поток процесса. Подобно технике Process Hollowing, поток сначала должен быть приостановлен.
• Инъекции в процедуры асинхронного вызова (Asynchronous Procedure Call (APC) injection) предполагают вложение вредоносного кода в очередь APC-процедур (APC Queue) потока процесса. Один из способов APC-инъекций, получивший название «Инъекция ранней пташки (Earle Bird injection)», предполагает создание приостановленного процесса в котором вредоносный код может быть записан и запущен до точки входа процесса через APC. AtomBombing — это другой вариант инъекции, который использует APC для вызова вредоносного кода, ранее записанного в глобальную таблицу атомов (Global atom table).
• Инъекции в локальное хранилище потока (Thread Local Storage (TLS) injection) предполагают манипуляции с указателями памяти внутри исполняемого PE-файла для перенаправления процесса на вредоносный код.

Mac и Linux
• Системные переменные LD_RPELOAD, LD_LIBRARY_PATH (Linux), DYLIB_INSERT_LIBRARIES (macOS X) или интерфейс прикладного программирования dlfcn (API) могут использоваться для динамической загрузки библиотеки (общего объекта) в процесс, который в свою очередь может использоваться для перехвата вызовов API из запущенных процессов.
• Системный вызов Ptrace может использоваться для подключения к запущенному процессу и изменения во время его выполнения.
• /proc/[pid]/mem обеспечивает доступ к памяти процесса и может использоваться для чтения/записи произвольных данных, однако такой метод редко применяется из-за сложности его реализации.
• Захват VDSO (Virtual dynamic shared object) позволяет осуществить инъекцию кода во время исполнения двоичных файлов ELF, манипулируя заглушками кода из linux-vdso.so.
Вредоносные программы обычно используют инъекции кода в процесс для доступа к системным ресурсам, благодаря которым злоумышленник может закрепиться в системе и выполнять другие изменения в атакуемой среде. Более сложные образцы могут выполнять множественные инъекции процессов для затруднения своего обнаружения.

Рекомендации по защите: Методы инъецирования кода в процессы основаны на злоупотреблении штатными функциями ОС прямое воздействие на которые может привести к нестабильной работе законного ПО и продуктов безопасности. Усилия по предотвращению применения техник перехвата необходимо сосредоточить на более ранних этапах цепочки атаки. Применяйте инструменты блокировки потенциально-опасного ПО, такие как AppLocker. Применяйте Yama в качестве превентивной меры от инъекций кода в ptrace, ограничив использование ptrace только привилегированными пользователями. Дополнительные меры защиты могут включать развертывание модулей безопасности ядра, обеспечивающих расширенный контроль доступа и ограничение процессов. К таким средствам относятся SELinux, grsecurity, AppArmor.

Инъекции в SID-истории (SID-History Injection)

Система: Windows
Права: Администратор, система

Описание: Всякий раз, когда объект перемещается из одного домена в другой создается новый SID, который становится основным objectSID. Предыдущие SID продолжают храниться в свойстве sIDHistory, таким образом обеспечивая сохранение прав, которые имел объект до междоменной миграции. Злоумышленники, при наличии прав администратора, могут вставлять в SID-History собранные ранее SID, чтобы выполнить действие от имени более привилегированных групп доступа или учетных записей, такие как администраторы домена.

Рекомендации по защите: В ОС Windows Server версии 2003 и выше по умолчанию включена Фильтрация SID (SID Filtering), которая предполагает удаление или фильтрацию всех SID, кроме доверенного домена, однако данная опция может быть умышленно отключена, чтобы разрешить междоменный доступ.
Основные способы фильтрации SID:
• Отключение SIDHistory в параметрах доверительных отношений (трастов) между лесами доменов с помощью команды: *netdom trust /domain: /EnableSIDHistory:no*;
• Применение SID Filter Quarantining. Это гарантирует, что объект, содержащий SID, отличный от доверенного домена, не сможет пройти аутентификацию в доверяющем домене. SID Filter Quarantining применяется для внешних трастов с помощью выполнения команды: *netdom trust /domain: /quarantine:yes*.
Применение SID Filtering между доменами одного леса не рекомендуется. Если домен в лесу ненадёжен, то он не должен быть членом леса, в такой ситуации необходимо сначала разделить доверенные и ненадежные домены на отдельные леса, а затем применить SID Filtering для трастов между лесами.

Планирование задач (Scheduled Task)

Система: Windows
Права: Пользователь, администратор, система
Описание: Такие утилиты как at, schtasks и Планировщик задач Windows могут использоваться для планирования запуска программ и сценариев, которые будут выполнятся в определенную дату и время. Задачу можно запланировать в удаленной системе, при условии, что для проверки подлинности используется RPC, и включен общий доступ к принтерам и файлам. Планирование задач в удаленной системе требует прав администратора. Злоумышленник может использовать удаленное выполнение кода для получения прав System или для запуска процесса под определенной учетной записью.

Рекомендации по защите: Ограничение привилегий пользователей. Применение инструментов, таких как модуль PowerUP в PowerSploit, которые могут использоваться для поиска слабых мест в разрешениях запланированных задач. Отключение возможности запуска задач от имени System, отключение в политике безопасности параметра "*Разрешить операторам сервера планировать задачи*" и включение параметра "*Назначение прав пользователя: Увеличить приоритет планирования*".

Слабости разрешений параметров служб в реестре (Service Registry Permissions Weakness)

Система: Windows

Права: Администратор, System
Описание: Если разрешения пользователей и групп позволяют изменять в реестре Windows значения ключей, в которых хранятся параметры служб, то злоумышленники могут напрямую модифицировать ключи, в которых хранятся пути к исполняемым файлам запуска служб или использовать различные инструменты управления службами — sc.exe, PowerShell или Reg. Атакующие так же могут менять параметры, связанные с отказом служб, например, FailureCommand, указывающие команду, которая будет выполнятся в случае отказа или преднамеренного повреждения службы. Параметры служб хранятся в *HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\services*.

Рекомендации по защите: Убедитесь, что пользователи защищаемой системы не могут изменять в реестре ключи, хранящие параметры системных компонентов. Используйте всевозможные средства блокирования потенциально-опасного ПО, например, Windows AppLocker.

Setuid и Setgid

Система: Linux, macOS
Права: Пользователь
Описание: Setuid и Setgid — это флаги прав доступа в Unix-системах, которые разрешают пользователю запускать исполняемые файлы с правами владельца или группы исполняемого файла. Если приложение необходимо запустить от имени root, то вместо того, чтобы создавать запись в файле sudo, пользователь может указать флаг Setuid или Setgid. Противники могут злоупотреблять флагами Setuid и Setgid чтобы выполнить shell escape (приём, когда в консольном режиме какое-либо приложение использует файл открытый в другом приложении) либо воспользоваться уязвимостью приложения с флагами Setuid и Setgid и выполнить код в контексте различных пользователей. При просмотре атрибутов файла командой ls -l вышеописанные флаги обозначаются символом «s» вместо «x». Утилита chmod может устанавливать флаги Setuid и Setgid с помощью команды *chmod 4777 [файл]* или *chmod u + s [файл]*.

Рекомендации по защите: Сведите количество программ с установленными флагами Setuid и Setgid к минимуму.

Элементы автозапуска (Startup Items)

Система: macOS
Права: Администратор
Описание: Атакующий может использовать устаревший, но до сих пор работающий в macOS Sierra, механизм автозапуска приложений с помощью StartupItems для настройки запуска своего кода с правами root во время загрузки ОС. StartupItems — это каталог в */Library/Startupitems*, командный сценарий и файл свойств StartupParameters.plist. Сценарий и файл свойств должны находится на верхнем уровне иерархии: */Library/Startupitems/[MyStartupItem]*.

Рекомендации по защите: Поскольку механизм StartupItems является устаревшим, то запрет записи в каталоге */Library/Startupitems/* позволит избежать создания элементов автозагрузки.

Sudo

Система: Linux, macOS
Права: Пользователь
Описание: Противники могут пользоваться недостатками конфигурации Sudo для выполнения команд от имени других пользователей или порождения процессов с более высокими привилегиями. Параметры Sudo хранятся в файле */etc/sudoers*, для редактирования этого файла необходимы повышенные привилегии. Файл sudoers описывает какие команды пользователи могут запускать от имени других пользователей или групп, это позволяет пользователям работать большую часть времени с минимальными правами и только при необходимости повышать привилегии. Однако, в файле sudoers можно указать пользователей, для которых пароль запрашиваться не будет: *username ALL=(ALL) NOPASSWD: ALL*.

Рекомендации по защите: Файл sudoers должен быть отредактирован так, чтобы пользователи всегда вводили пароль при выполнении Sudo. Auditd в Linux может генерировать предупреждение всякий раз, когда реальный и эффективный ID пользователя не совпадают (это происходит, когда пользователь использует sudo).

Sudo Caching

Система: Linux, macOS
Права: User
Описание: Различные вредоносные программы, как, например, *OCX Proton Malware*, могут злоупотреблять настройками sudo, чтобы выполнить код от имени root без ввода пароля. Поскольку инструментарий sudo был создан для системного администрирования в нём есть некоторые полезные функции такие как *timestamp_timeout* — этот параметр хранит количество времени в минутах между запусками sudo в течение которого команда не будет запрашивать ввод пароля root. Sudo имеет возможность кэширования учетных данных в течение некоторого времени. Отметка о времени последнего запуска Sudo хранится в */var/db/sudo* и служит для определения заданного таймаута. Кроме того, существует переменная *tty_tickets*, которая обрабатывает каждый новый сеанс терминала изолированно, таким образом таймаут в одном экземпляре консоли не повлияет на таймаут в другом экземпляре.

Рекомендации по защите: Установите значение параметра *timestamp_timeout = 0*, чтобы система требовала ввод пароля root при каждом выполнении sudo. Включите параметр *tty_tickets*, чтобы предотвратить возможность реализации атаки через сеансы командной консоли.

Valid Accounts

Описание: Злоумышленники могут украсть учетные данные определенного пользователя

или учетную запись службы с помощью техник доступа к учетным данным, захватить учетные данные в процессе разведки с помощью социальной инженерии. Скомпрометированные учетные данные могут использоваться для обхода систем управления доступом и получения доступа к удаленным системам и внешним службам, таким как VPN, OWA, удаленный рабочий стол или получения повышенных привилегий в определенных системах и областях сети. В случае успешной реализации сценария злоумышленники могут отказаться от вредоносных программ, чтобы затруднить своё обнаружение. Так же злоумышленники могут создавать учетные записи используя заранее определенные имена и пароли для сохранения резервного доступа в случае неудачных попыток использования других средств.

Рекомендации по защите: Применение парольной политики, следование рекомендациям по проектированию и администрированию корпоративной сети для ограничения использования привилегированных учетных записей на всех административных уровнях. Регулярные проверки доменных, локальных учетных записей и их прав с целью выявления тех, которые могут позволить злоумышленнику получить широкий доступ. Мониторинг активности учетных записей с помощью SIEM-систем.

Web Shell

Система: Windows, Linux, macOS
Описание: Web Shell может использоваться злоумышленником в качестве шлюза доступа в вашу сеть или избыточного доступа в атакуемую систему, как резервного механизма закрепления в случае обнаружения и блокирования основных каналов доступа в атакуемую среду. *Рекомендации по защите:* Убедитесь, что ваши внешние веб-серверы регулярно обновляются и не имеют известных уязвимостей, которые позволяют злоумышленникам загрузить на сервер файл или сценарий с последующим исполнением. Проверьте, что разрешения учетных записей и групп с правами управления серверами не совпадают с учетными записями внутренней сети, которые могут быть использованы для входа на веб-сервер, запуска Web shell или закрепления на Web-сервере. Web Shell трудно обнаружить, т.к. они не инициируют подключения и их серверная часть может быть маленькой и безобидной, например, PHP-версия оболочки China Chopper Web выглядит как строчка:
[?php eval($_POST ['password']);]