1. **Определение эсклалации привелегий**

Эскалация привилегий представляет собой процесс, в ходе которого злоумышленники пытаются получить более высокий уровень доступа к системе, чем тот, который был предназначен для них. В контексте информационной безопасности, это означает, что атакующий стремится получить расширенные права или привилегии, которые позволяют ему иметь контроль над системой, обходя стандартные ограничения.

1. **Цели атаки**

Злоумышленники преследуют несколько целей при попытках эскалации привилегий. Во-первых, расширение своих привилегий позволяет им иметь более полный контроль над целевой системой или сетью. Это может включать в себя доступ к конфиденциальной информации, возможность модификации системных настроек или даже запуск вредоносных программ.

Во-вторых, эскалация привилегий может служить средством для скрытного нахождения в системе. Обладание более высокими привилегиями может позволить злоумышленнику избегать обнаружения, так как он может обходить средства мониторинга и обнаружения вторжений.

Наконец, эскалация привилегий часто является ключевым этапом в цепочке атак. После того, как злоумышленник получил расширенные права, ему становится легче продвигаться дальше в системе и достигать своих целей.

1. **Недостаток передачи сообщений Windows**

Когда пользователь нажимает клавишу, в активное окно отправляется сообщение о нажатии клавиши. Когда Windows решает, что приложение должно перерисовать свою рабочую область, она отправляет сообщение этому приложению. Фактически, при возникновении любого события, о котором приложение должно быть уведомлено, ему отправляется сообщение. Эти сообщения помещаются в очередь и обрабатываются приложением в порядке очереди.

Это очень надежный механизм управления приложениями. Однако в Win32 есть недостаток в механизме управления этими сообщениями. Любое приложение на данном рабочем столе может отправлять сообщение любому окну на том же рабочем столе, независимо от того, принадлежит ли это окно отправляющему приложению или нет, и независимо от того, хочет ли целевое приложение получать эти сообщения. Нет механизма для аутентификации источника сообщения; сообщение, отправленное злонамеренным приложением, невозможно отличить от сообщения, отправленного ядром Windows.

1. **Shatter attack**

О данном технике атак стало известно после публикации «Chris Paget. Exploiting design flaws in the Win32 API for privilege escalation» В своё докладе он рассказал как можно воспользоваться вышеописанным недостатком Windows. На примере программы Network Associates VirusScan v4.5.1. Цель - обмануть VirusScan и запустить свой код для повышения привилегий. Процесс разбивается на четыре этапа.

*Этап 1: Поиск окна*

Нам необходимо найти элемент управления, в который можно вводить текст. Нажимаем кнопку "New Task" в консоли VirusScan, затем обращаемся к Shatter, чтобы получить идентификатор этого окна. Теперь мы можем взаимодействовать с окном, работающим с более высокими привилегиями.

*Этап 2: Устранение ограничений*

Теперь, когда у нас есть идентификатор окна, мы можем отправлять ему любые сообщения, и оно будет их выполнит. Сперва убеждаемся, что у нас достаточно места для нашего исполняемого кода.

*Этап 3: Внедрение Shellcode*

Теперь давайте попробуем вставить что-то в поле. Очистите поле VirusScan, включите блокнот, введите текст и скопируйте его. Затем отправим VirusScan сообщение "Вставить содержимое буфера обмена" (WM\_PASTE), и наш текст появится в поле. Процесс вставки шелл-кода описан в докладе.

*Этап 4: Выполнение кода*

Этот этап требует навыков работы с отладчиком. Подключаемся к процессу avconsol.exe, ищем адрес шелл-кода, а затем выполняем его. После этого мы получаем привилегии локальной системы.

Важно помнить, что некоторые этапы можно автоматизировать, и приложения могут иметь свои обработчики исключений. Таким образом, атака Shatter представляет серьезную угрозу, требующую внимания разработчиков и пользователей Windows.

Этим завершается практический пример атаки Shatter, демонстрируя её потенциальные последствия и подчеркивая важность безопасности при разработке и использовании программного обеспечения под Windows.

1. Борьба с Shatter Attack

С течением времени, проблема Shatter Attack привлекла внимание разработчиков и инженеров по безопасности. В данной главе мы рассмотрим меры, принятые Microsoft для предотвращения и минимизации рисков, связанных с Shatter Attack.

*Патчи и Частичные Решения*

В декабре 2002 года Microsoft реагировала на угрозу, выпустив патч для систем Windows NT 4.0, Windows 2000 и Windows XP. Этот патч ограничил использование "shatter attack", но коснулся лишь служб, поставляемых с Windows. Однако сама архитектура осталась без изменений, и угроза существовала для остальных приложений и служб.

*Комплексные Изменения в Windows Vista*

С появлением Windows Vista, Microsoft приняла более радикальные меры для борьбы с Shatter Attack. Во-первых, сеанс 0 был выделен исключительно для системных процессов, а пользователь больше не имеет прямого доступа к этому сеансу. Во-вторых, было внедрено ограничение на отправку большинства сообщений от процессов с низкими привилегиями к процессам с высокими привилегиями - User Interface Privilege Isolation (UIPI). Эти изменения существенно повысили уровень безопасности системы.

Пример: Internet Explorer 7 воспользовался нововведением UIPI, чтобы ограничить взаимодействие компонентов визуализации с остальной системой.

Эти комплексные изменения в архитектуре Windows Vista привели к существенному уменьшению уязвимости к Shatter Attack и подчеркнули стремление Microsoft к обеспечению надежности и безопасности своей операционной системы.