Обзор технологии Intel SMEP и её частичный обход на ОС Windows 8

АРТЁМ ШИШКИН

Positive Research Center

Moscow
2012

ОГЛАВЛЕНИЕ

АННОТАЦИЯ

В данной статье приводится обзор нового аппаратно реализованного средства безопасности, представленного компанией Intel, и описание его поддержки в ОС Windows 8. Наряду с другими средствами безопасности оно усложняет эксплуатацию уязвимостей на целевой системе. Однако, если данные средства некорректно сконфигурированы, они могут стать в принципе бесполезными. В данной статье показан дефект безопасности на x86 версиях ОС Windows 8, который приводит к обходу технологии Intel SMEP.

1. ВВЕДЕНИЕ

С приходом нового поколения процессоров Intel на базе архитектуры Ivy Bridge было представлено новое средство безопасности. Оно называется Intel SMEP, что расшифровывается как “Supervisor Mode Execution Prevention” — предотвращение исполнения кода в режиме супервизора. Технология заключается в предотвращении выполнения кода, расположенного на пользовательской странице, при текущем уровне привилегий равном 0. С точки зрения атакующего, данное средство значительно усложняет эксплуатацию уязвимостей режима ядра, потому как в данных условиях отсутствует место для хранения шелл-кода. Обычно при эксплуатации уязвимости режима ядра атакующий выделяет буфер с шелл-кодом в пользовательском режиме и затем активирует уязвимость, получая контроль над исполнением кода и переопределяя точку исполнения на содержимое подготовленного буфера. Таким образом, если атакующий не может исполнить свой шелл-код, вся атака бессмысленна. Конечно, существуют различные техники, такие как возвратно-ориентированное программирование (ROP) для эксплуатации

уязвимостей с полезной нагрузкой. Однако бывают и такие случаи, когда среда исполнения позволяет обходить ограничения безопасности при некорректной конфигурации. Рассмотрим подробнее технологию Intel SMEP, а также её программную поддержку, заявленную в ОС Windows 8.

2. АППАРАТНАЯ СОСТАВЛЯЮЩАЯ ТЕХНОЛОГИИ INTEL SMEP

Данный раздел включает в себя обзор технологии Intel SMEP на основе документации Intel.

SMEP является частью механизма защиты страниц памяти. На самом деле она использует уже существующий флаг в записи таблицы страниц — флаг U/S (флаг User/Supervisor, 2-й бит). Этот флаг указывает на то, является данная страница страницей пользовательского режима или режима ядра. Владелец страницы определяет наличие доступа к данной странице памяти, то есть, если страница принадлежит ядру операционной системы, которая исполняет код в привилегированном режиме, доступ к ней из пользовательского приложения невозможен.

SMEP включается и выключается при помощи управляющего регистра CR4 (20-й бит). Установка данного бита изменяет влияние флага U/S на доступ к страницам памяти. При попытке исполнения кода, расположенного на пользовательской странице, в привилегированном режиме аппаратно генерируется ошибка страницы (page fault) в результате нарушения прав доступа (права доступа описаны в главе 4.6 тома 3 [1]).

Как видно, SMEP генерирует не исключение общего нарушения защиты (#GP), но ошибку страницы (#PF). Таким образом, ОС должна обработать нарушение механизма SMEP в обработчике ошибок страниц. Эта деталь нам понадобится в дальнейшем при анализе программной поддержки механизма SMEP.

3. ПРОГРАММНАЯ ПОДДЕРЖКА ТЕХНОЛОГИИ SMEP

Поддержка SMEP может быть определена при помощи инструкции “cpuid”. Результат исполнения запроса “cpuid” для уровня 7 с подуровнем 0 (входные параметры EAX = 7, ECX = 0) указывает на поддержку SMEP на данном процессоре — для этого необходимо проверить 7-й бит регистра EBX.

64-разрядная версия Windows 8 проверяет поддержку SMEP при инициализации загрузочных структур, заполняя переменную “KeFeatureBits”:

KiSystemStartup() → KiInitializeBootStructures() → KiSetFeatureBits()

То же самое происходит в x86 версии Windows 8:

KiSystemStartup() → KiInitializeKernel() → KiGetFeatureBits()

Переменная “KeFeatureBits” в дальнейшем используется при обработке ошибок страниц.

Если процессор поддерживает технологию SMEP, ОС включает её, устанавливая 20-й бит регистра CR4. На x86 версии она включается также во время старта системы, в фазе 1 в функции KiInitMachineDependent() и затем инициализируется для каждого ядра процессора, инициируя межпроцессорное прерывание, которое в итоге вызывает функцию KiConfigureDynamicProcessor(). То же самое происходит на x64 версии ОС, за исключением того, что в ней отсутствует функция KiInitMachineDependent().

Таким образом, включение SMEP и инициализация переменной “KeFeatureBits” происходит при старте системы. Другой частью

программной поддержки SMEP является код обработчика ошибок страниц. В Windows 8 была добавлена новая функция — MI_CHECK_KERNEL_NOEXECUTE_FAULT(). Внутри неё происходит проверка на нарушение технологий SMEP и NX. Если произошло нарушение прав доступа, связанное с SMEP или NX, то ОС показывает пользователю синий экран смерти с кодом ошибки “ATTEMPTED_EXECUTE_OF_NOEXECUTE_MEMORY”:

KiTrap0E()/KiPageFault() → MmAccessFault() → … →

→ MI_CHECK_KERNEL_NOEXECUTE_FAULT()

Данная функция реализована только в Windows 8.

4. СПОСОБ ОБХОДА SMEP НА ОС WINDOWS И ПРОТИВОДЕЙСТВИЕ ЕМУ

Логично предположить, что если нельзя хранить шелл-код в пользовательском адресном пространстве, необходимо найти способ его внедрения в пространство ядра. Наиболее очевидным решением в данном случае является использование объектов Windows, таких как WinAPI (таймеры, события, секции и т.д.) или GDI (контексты устройств, палитры и т.д.). Доступ к ним осуществляется косвенным образом через функции WinAPI, которые в свою очередь используют системные вызовы. Суть в том, что тело объекта хранится в памяти ядра, а его поля могут быть изменены в пользовательском режиме. Таким образом, атакующий способен передать необходимые байты шелл-кода из пространства пользователя в пространство ядра.

Также очевидно, что атакующему необходимо знать, где именно в пространстве ядра находится тело используемого им объекта. Для этого необходим метод раскрытия информации о пространстве ядра, поскольку, как мы помним, пользовательскому приложению недоступны страницы

памяти ядра, в том числе и для чтения. Такие методы существуют в ОС Windows [2].

Из вышеописанного следует, что теоретически возможно обойти SMEP благодаря раскрытию информации о пространстве ядра в ОС Windows. Однако SMEP дополняется таким механизмом защиты, как использование пулов, помеченных как неисполняемые, для выделения памяти для объектов в Windows 8.

В рамках данной работы были проверены на пригодность для внедрения шелл-кода в пространство ядра различные объекты Windows. Объекты WinAPI хранятся в подкачиваемом и неподкачиваемом пулах. Объекты GDI хранятся в подкачиваемом пуле сессии (session pool). Все они являются неисполняемыми в Windows 8. Более того, в соответствии с результатом сканирования таблиц страниц количество использующихся страниц из исполняемых пулов теперь крайне мало. Все буферы данных теперь являются неисполняемыми. Большинство исполняемых страниц (например, образы драйверов) недоступны для записи.

4.1. Недостаток конфигурации

Как упоминалось выше, все объекты в Windows 8 теперь хранятся в неисполняемых пулах. Это утверждение справедливо для x64 версии и частично для x86 версии ОС. Недостатком является пул сессии. Он помечен как исполняемый на x86 версии Windows 8. Следовательно, можно использовать подходящий объект GDI для хранения шелл-кода в памяти ядра.

Наиболее подходящим объектом для этой цели является GDI Palette — объект палитры. Он создается при помощи функции CreatePalette() и соответствующей заполненной структуры LOGPALETTE. Данная структура содержит в себе массив структур PALETTEENTRY, которые определяют цвета и их использование в логической палитре [5]. Смысл в том, что в отличие от других объектов GDI при создании палитры нет валидации на содержимое массива цветов. Атакующий может хранить любые цвета в

своей палитре. Таким же образом он может хранить байты шелл-кода. Адрес тела объекта палитры может быть получен при помощи разделяемой таблицы GDI. Содержимое палитры хранится по некторому смещению (в нашем случае 0x54). Однако данное смещение знать не обязательно, поскольку шелл-код можно хранить где-нибудь после заполнения палитры инструкциями NOP. Схема обхода SMEP представлена на рисунке 1.

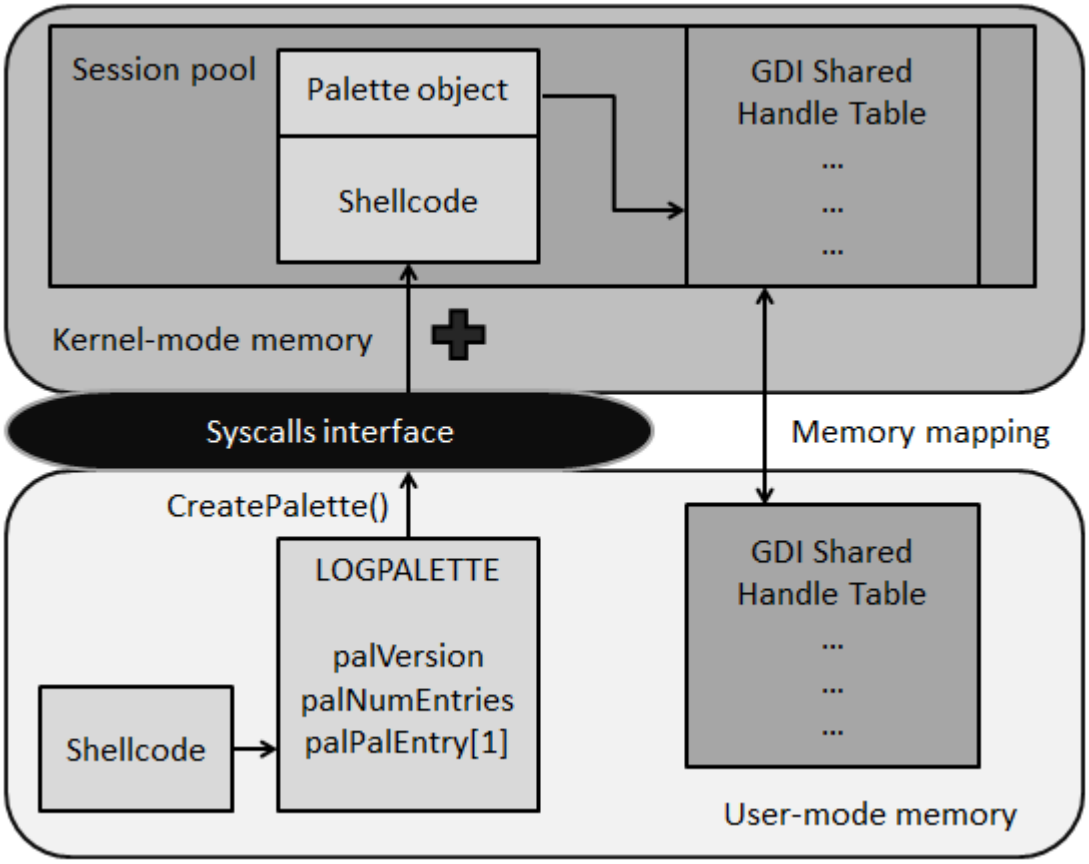

Рисунок 1. Схема обхода SMEP на x86 версии Windows 8

Объект палитры предоставляет достаточное количество байт для хранения объемного шелл-кода. Однако в действительности всё, что нужно атакующему, это отключить SMEP. Это можно сделать простым сбросом 20-го бита управляющего регистра CR4, после чего атакующий получит возможность исполнения кода из пользовательского буфера уже без каких-либо ограничений на размер.

Разумеется, при использовании пула сессии существуют некоторые ограничения. Во-первых, он является подкачиваемым, следовательно, необходимо учитывать уровень запроса на прерывание (IRQL) при эксплуатации уязвимости режима ядра. Во-вторых, пул сессии проецируется в соответствии с текущей пользовательской сессией, значит, также необходимо учитывать текущую сессию. И в-третьих, в многопроцессорной среде управляющие регистры присутствуют на каждом ядре, следовательно, необходимо использовать привязку потока к ядру для отключения SMEP на конкретном ядре процессора.

4.2. Другие вектора атак для обхода SMEP

Как упоминалось выше, возвратно-ориентированное программирование может успешно использоваться для обхода средства защиты SMEP, поскольку в данном варианте необязательно хранить подготовленный заранее шелл-код. Вместо этого можно использовать фрагменты кода, уже хранящиеся в памяти ядра (например, драйвера).

Также существует возможность эксплуатации драйверов сторонних производителей, которые пока не используют неисполняемые пулы для хранения данных.

5. ЗАКЛЮЧЕНИЕ

В данной статье был описан принцип работы технологии Intel SMEP и её программная поддержка в Windows 8. Также был показан вариант обхода данной технологии в определенных случаях ввиду возможности раскрытия информации об адресном пространстве ОС и частичному применению механизмов защиты. Тем не менее, в том виде, в каком

реализована поддержка SMEP на x64 версиях Windows 8, она может считаться достаточно надежной и способной предотвратить различные атаки с использованием уязвимостей режима ядра.

6. ДАЛЬНЕЙШАЯ РАБОТА

Дальнейшая работа связана с исследованием распространенных драйверов сторонних производителей, которые всё ещё используют исполняемые пулы, и методов раскрытия информации, необходимой для эффективной эксплуатации таких драйверов. На данный момент это направление считается лучшим для исследования методов обхода SMEP.

ССЫЛКИ

[1] Intel: Intel® 64 and IA-32 Architectures Developer's Manual: Combined Volumes. Intel Corporation, 2012.

[2] Mateusz “j00ru” Jurczyk: *Windows Security Hardening Through Kernel Address Protection.* http://j00ru.vexillium.org/blog/04_12_11/Windows_Kernel_Address_Protection.pdf

[3] Mateusz “j00ru” Jurczyk, Gynvael Coldwind: *SMEP: What is it, and how to beat it on Windows.* http://j00ru.vexillium.org/?p=783

[4] Ken Johnson, Matt Miller: *Exploit Mitigation Improvements in Windows 8.* Slides, Black Hat USA, 2012.

[5] MSDN: *Windows GDI.* *http://msdn.microsoft.com/en-us/library/windows/desktop/dd145203(v=vs.85).aspx*

[6] Feng Yuan: Windows Graphics Programming Win32 GDI and DirectDraw®. Prentice Hall PTR, 2000.

[7] Mark Russinovich, David A. Solomon, Alex Ionescu: Windows® Internals: Including Windows Server 2008 and Windows Vista, Fifth Edition. Microsoft Press, 2009.

О КОМПАНИИ

Positive Technologies — лидер европейского рынка систем анализа защищенности и соответствия стандартам. Компания входит в число наиболее динамично развивающихся участников рынка ИТ, демонстрируя ежегодный рост более 50%. Офисы и представительства Positive Technologies расположены в Москве, Лондоне, Риме, Сеуле и Тунисе.

Разработанные экспертами компании программные продукты заслужили международное признание в сфере практической информационной безопасности.

Продукты

Система контроля защищенности и соответствия стандартам MaxPatrol помогает обеспечивать безопасность корпоративных информационных систем и формировать комплексное представление о реальном уровне защищенности IT-инфраструктуры организации. Система позволяет контролировать выполнение требований государственных, отраслевых и международных стандартов, таких как Федеральный закон № 152-ФЗ «О персональных данных», СТО БР ИББС, ISO 27001/27002, SOX 404, PCI DSS. В MaxPatrol объединены активные механизмы оценки защищенности, включая функции системных проверок, тестирования на проникновение, контроля соответствия стандартам — в сочетании с поддержкой анализа различных операционных систем, СУБД и веб-приложений.

Система анализа защищенности XSpider более 10 лет является признанным лидером среди средств сетевого аудита ИБ. На сегодняшний день это один из лучших интеллектуальных сканеров безопасности в мире. Более 1000 международных компаний успешно используют XSpider для анализа и контроля защищенности корпоративных ресурсов.

Услуги

Компания Positive Technologies специализируется на проведении комплексного аудита информационной безопасности, на оценке защищенности прикладных систем и веб-приложений, тестировании на проникновение и внедрении процессов мониторинга информационной безопасности. Статус PCI DSS Approved Scanning Vendor позволяет проводить работы по проверке соответствия данному стандарту.

Исследования

Positive Research — один из крупнейших в Европе исследовательских центров в области информационной безопасности. В его задачи входит анализ передовых тенденций IT-индустрии, а также их использование для развития продуктов и сервисов компании. Эксперты центра проводят исследовательские и конструкторские работы, анализ угроз и уязвимостей, содействуют разработчикам в устранении ошибок в различных системах и приложениях.

Лицензии

Свою деятельность Positive Technologies осуществляет на основе лицензий ФСБ, ФСТЭК и Министерства обороны РФ. Продукты компании сертифицированы ФСТЭК, Минобороны и ОАО «Газпром» (по системе ГАЗПРОМСЕРТ), а ее специалисты участвуют в работе различных международных ассоциаций: Web Application Security Consortium, (ISC)², ISACA, Certified Ethical Hacker, Center for Internet Security.

Клиенты

В числе заказчиков Positive Technologies — более 1000 государственных учреждений, финансовых организаций, телекоммуникационных и розничных компаний, промышленных предприятий России, стран СНГ и Балтии, а также Великобритании, Германии, Голландии, Израиля, Ирана, Китая, Мексики, США, Таиланда, Турции, Эквадора, ЮАР и Японии.

Вклад в развитие индустрии

Принимая активное участие в жизни отрасли, Positive Technologies выступает организатором международного форума по информационной безопасности Positive Hack Days и развивает SecurityLab — самый популярный русскоязычный портал о ИБ.

www.ptsecurity.com
pt@ptsecurity.com
+7 (495) 744 01 44