глубоководное погружение в недра vista/longhorn

крис касперски ака мыщъх, no-email

чем _реально_ vista отличается от своей предшественницы XP? какие _объективные_ преимущества она дает? с какими _проблемами_ нам придется столкнуться при переходе? и стоит ли этот переход того? как обстоят дела с удобством использования, производительностью, безопасностью? мыщъх перерыл все ядро с сопредельными территориями, отделив зерна от шелухи рекламных плевел и теперь готов поделиться результатами своих исследований с общественностью

введение

Обсуждения целесообразности перехода на висту совершенно безосновательны. Как будто у нас есть выбор — переходить или… не переходить. Такие решения принимаются на высоком корпоративном уровне и не нами, а за нас. Неизбежность перехода на висту в исторической перспективе совершенно очевидна. Пройдет совсем немного времени и Microsoft прекратит поддержку XP (поддержка w2k еще не прекращена, но легальным путем ее уже не добыть), появятся программы и оборудование, работающие только на висте, а сама виста окажется предустановленной на миллионах компьютерах.

Мы можем лишь затянуть переход на висту, но предотвратить его не в силах. Отношение к самой висте у мыщъх'а многократно менялось по ходу исследований: от абсолютной неприязни, до желания выдрать ядро висты и скрестить его с w2k, получив операционную систему своей мечты, в процессе осуществления которой мыщъх неожиданно разглядел демонический лик, скрывающийся за ангельским интерфейсом и после глубокой депрессии высел на измену, граничащую с суицидом. Как стыдно за мир, в котором приходится жить, но… обо всем по порядку.

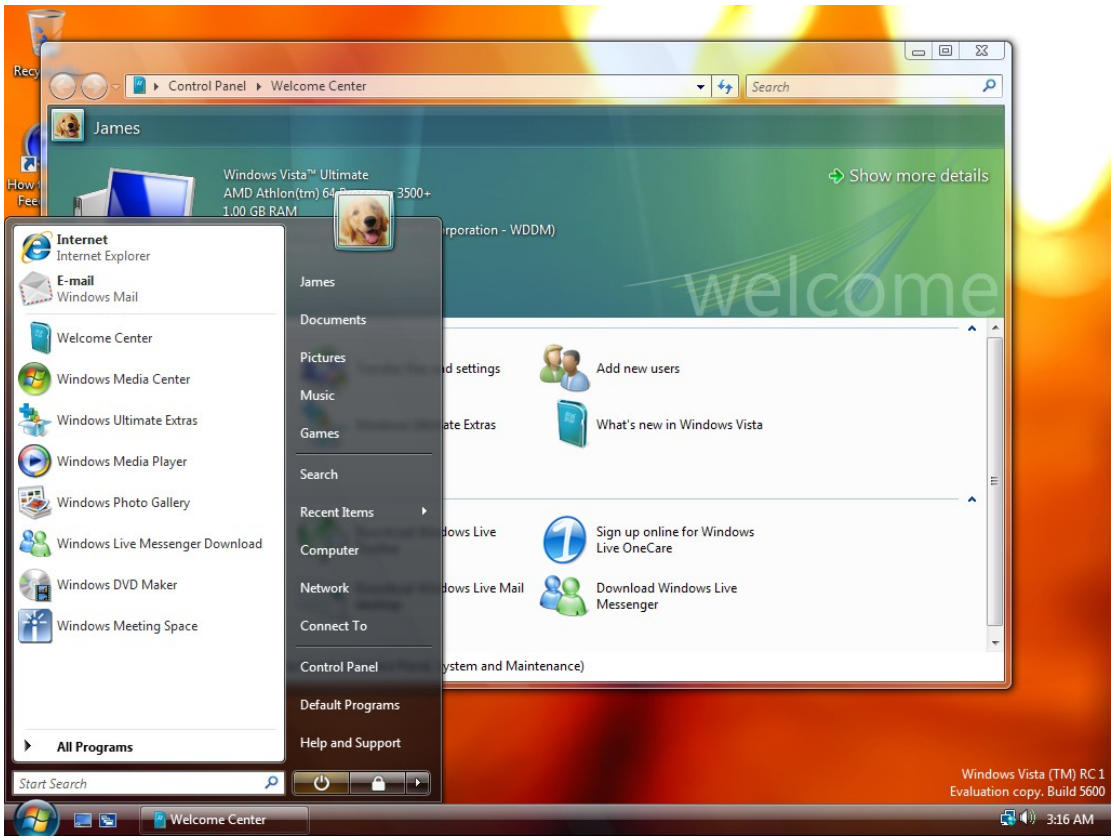

Рисунок 1 новый ангельский интерфейс "aero", скрывающий демонический лик ядра

>>> врезка основные достопримечательности висты

- завышенные требования к оборудованию;
- оптимизация файла подкачки (на системах страдающих недостатком памяти);
- оптимизация под многопроцессорные системы (от 2х двуядерных ЦП и свыше);
- **меньшая вероятность потери данных в случае сбоев или отключения питания;**
- новый пользовательский интерфейс с кучей спецэффектов, реализованный на .NET'е;
- ощутимые тормоза и потеря производительности вследствие двух последних пунктов;
- переписанный с нуля сетевой стек содержит кучу дыр, делающих висту **небезопасной**;
- поддержка нового железа (в частности: ACPI 2.0, PCI Express, Hybrid-носителей и т. д.);
- **поддержка старого железа и программного обеспечения _значительно_ ухудшена;**

что виста нам готовит

Microsoft радикально оптимизировала ядро, однако большая часть улучшений относится к многопроцессорным машинам (двуядерные процессоры не в счет) и менеджеру файла подкачки (при нынешних ценах на память вспоминать о подкачке просто смешно, имея всего лишь 512 Мбайт на W2K от нее можно полностью отказаться). Остальные механизмы оптимизации проявляют себя лишь при работе с приложениями, жадных до памяти, или интенсивном дисковом вводе/выводе, что типично для серверов, и совсем нетипично для рабочих станций. Но даже этот выигрыш "скомпенсирован" тормозами, порожденными усиленной защитой реестра и файловой системы от непреднамеренного разрушения, что опять-таки больше полезно для серверов, чем для рабочих станций. Про возможность "горячего" добавления оперативной памяти и процессоров можно было бы даже и не упоминать, если бы материнские платы не выгорали при этом до основания. Одной поддержкой со стороны операционной системы тут не обойтись, требуется специальное оборудование, изначально рассчитанное на такие издевательства и применяющиеся исключительно в мощных серверах, как правило, объединенных в кластеры (подробности можно найти в официальной презентации от MS: download.microsoft.com/download/f/0/5/f05a42ce-575b-4c60-82d6-208d3754b2d6/MemoryManagerInWindows.ppt).

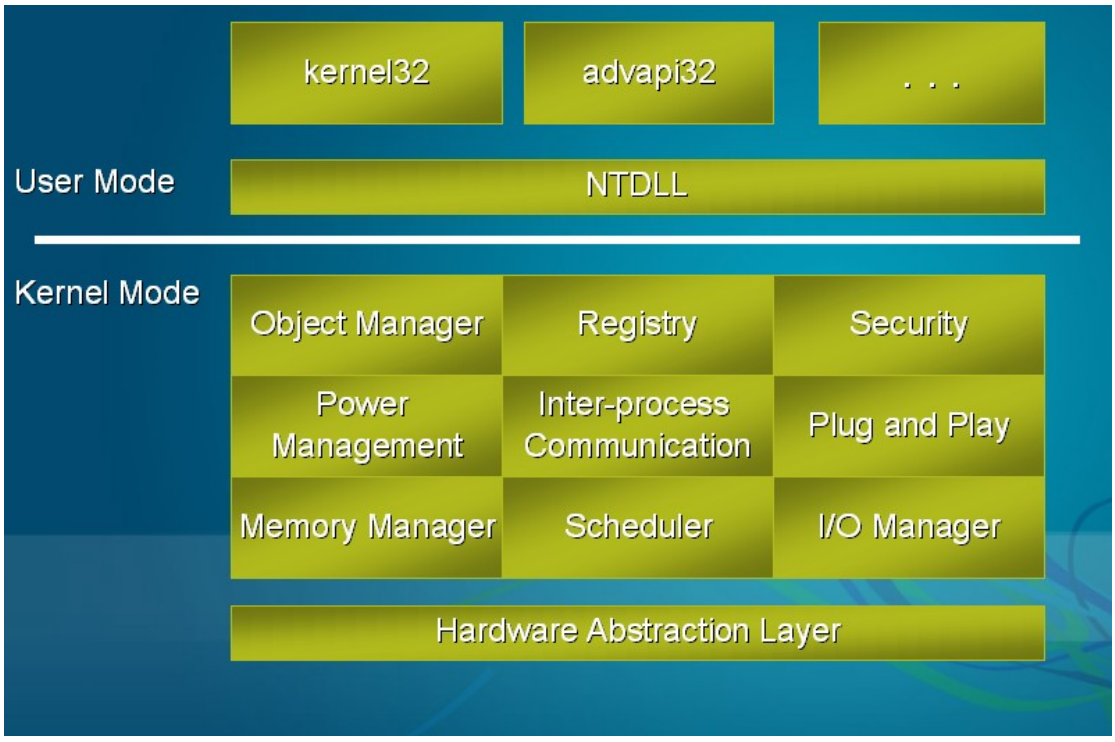

Рисунок 2 архитектура ядра 32-разрядной версии висты

Поверх ядра Microsoft взгромоздила множество новых служб, в том числе и глубоко ненавистную многим программистам платформу .NET (представляющую по сути тот же самый Visual Basic, только в другом обличие) и "аэродинамический" интерфейс с кучей спецэффектов, пожирающих оперативную память и процессорные такты в неимоверных количествах. То есть, вместо обещанного ускорения, мы получим конкретные тормоза.

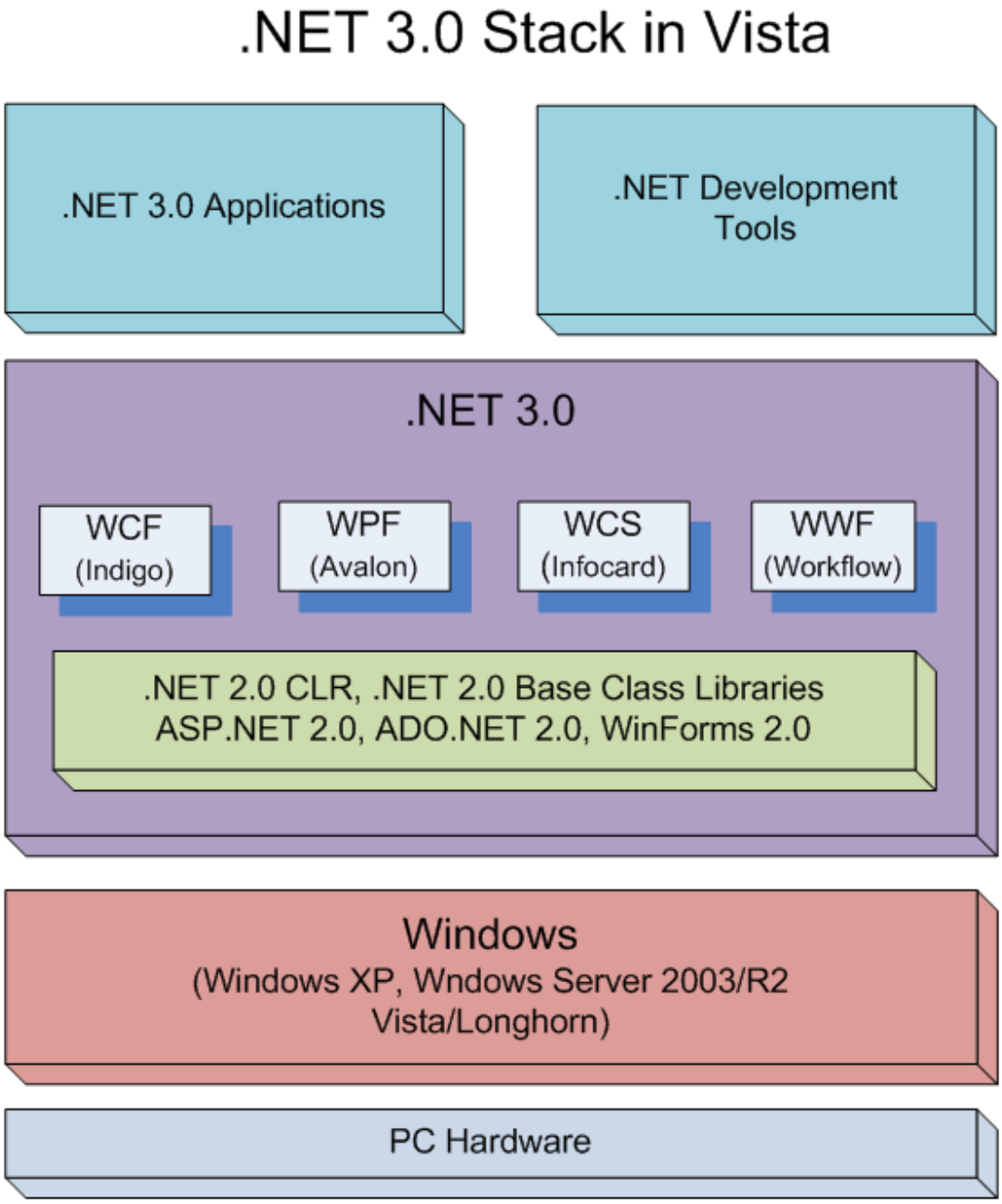

Рисунок 3 платформа .NET, натянутая на ядро

Ладно, хрен с нею, с производительностью. Microsoft уже приучила нас, что каждая последующая версия Windows работает медленнее предыдущей и требует намного более мощного железа. **Основное кредо висты — это безопасность, точнее — полное отсутствие таковой**. Формально, разработчики предприняли целый комплекс "противотанковых" мер (то есть, мер, направленных против тех кто в танке): рандомизацию адресного пространства, контроль целостности служебных структур динамической памяти с шифровкой магическим словом по XOR, изоляцию нулевой сессии от пользовательских приложений, понижение уровня привилегий некоторых сетевых сервисов и т. д., и т. п. (полный перечень содержится в официальном документе: http://download.microsoft.com/download/9/c/5/9c5b2167-8017-4bae-9fde-d599bac8184a/kernel-en.doc), но, во-первых, все это уже давно было реализовано сторонними разработчиками в тот же защитном комплексе BufferShield (подробнее о котором можно прочитать в статье: "переполнение буфера на системах с неисполняемым стеком", лежащей на мыщъх'ином ftp), только теперь пользователь получает a-la BufferShield в одной коробке с

Windows _без_ возможности его отключения (даже если он ему на хрен не нужен), а, во-вторых (и это самое важное!), разработчики похоронили старый сетевой стек, переписав его с нуля и один хвост знает сколько _новых_ ошибок допустили при этом.

Рисунок 4 они похерили старый сетевой стек и написали новый

Корпорация Symantec провела свое собственное расследование (отчет можно найти на: http://www.symantec.com/avcenter/reference/ATR-VistaAttackSurface.pdf), в результате которого пришла к весьма неутешительным выводам: качество реализация далеко от идеала и по степени защищенности новый сетевой стек значительно уступает старому стеку из XP. А тут еще как на грех в сентябре 2006 хакер Johnny Cache открыл **принципиально новый тип удаленных атак**, основанный на ошибках синхронизации потоков и допускающий захват управления с ядерными привилегиями. Угроза распространяется на все компьютеры, оснащенные сетевыми устройствами, обрабатывающими асинхронные запросы (беспроводные и ИК адаптеры, DSL-модемы, голубые зубья и т. д.)

Несмотря на предоставленные ей дампы памяти, подтверждающие успешное воздействие на регистр EIP, Microsoft никак не отреагировала на происходящее, переложив вину на разработчиков драйверов, в которых и была обнаружена ошибка. А ведь ошибки подобного типа носят повсеместный характер (в данном случае они обнаружились в драйверах от весьма нехилой конторы по имени Intel) и операционная система не предоставляет _никаких_ средств защиты и навряд ли предоставит их в дальнейшем, поскольку разрушения данных, происходящие при "срыве" синхронизации носят весьма специфичный характер.

Хакерская мысль не стоит на месте, а неуклонно движется вперед. Microsoft же, тем временем, добавляет в Висту новые методы синхронизации, упрощающие программирование драйверов и ликвидирующие часть проблем (см. соответствующую врезку), но… ни сегодня, ни завтра, писать драйвера специально под Висту никто не будет (даже такой гигант, как Intel), поскольку рыночная доля w2k, XP и Server 2003 слишком велика для того, чтобы разработчик мог использовать API, присутствующие в одной лишь Висте.

>>> врезка ссылки на новые методы синхронизации

- https://msdn.microsoft.com/library/default.asp?url=/library/en-us/dllproc/base/synchronization_functions.asp
- http://msdn.microsoft.com/library/default.asp?url=/library/en-us/dllproc/base/condition_variables.asp
- http://msdn.microsoft.com/library/default.asp?url=/library/en-us/dllproc/base/using_condition_variables.asp
- http://msdn.microsoft.com/library/default.asp?url=/library/en-us/dllproc/base/one-time_initialization.asp

- http://msdn.microsoft.com/msdnmag/issues/06/04/Deadlocks/default.aspx

>>> врезка версии висты

Операционная система Windows NT 6.0 существует в двух версиях, известных под торговыми марками **Windows Vista** и **Windows Server Longhorn**, каждая из которых представлена в двух редакциях — под 32-битную (x86) и 64-битную (x86-64) платформы. Если разница между сервером и рабочей станцией очевидна и не требует дополнительных комментариев, то преимущества и недостатки 64-битной редакции заслуживают развернутого объяснения.

Миллионы программ и мегатонны оборудования, работающие на x86 версии NT, не позволили Microsoft'у основательно перетряхнуть ядро без потери обратной совместимости. И хотя совместимость все-таки пострадала (список несовместимых программ можно найти на: www.iexbeta.com/wiki/index.php/Windows_Vista_Beta_2_Software_Compatibility_List), но все же не столь радикально, как в 64-битной версии, спроектированной с чистого листа, без оглядки на совместимость, поскольку ни оборудования, ни программного обеспечения под нее еще не существовало.

Воспользовавшись "подарком" от AMD, Microsoft перенесла NT на платформу x86-64, превратив систему в настоящую **тюрьму** (хотя реклама уверяет нас, что это — крепость). Именно в 64-битной редакции NT реализована защита ядерных функций от перехвата (без которых немыслимо создание качественных антивирусов, брандмауэров и прочих программ подобного типа), именно здесь цифровая подпись драйверов является обязательной, а прикладного программного обеспечения — желательной.

Все это сделано ради двух целей: монополизации рынка системного программирования в руках Microsoft и позиционирования своей системы как защищенной от грабежа premium content'a, что очень нравится Голливуду и другим медиа-магнатам. Борьба с малварью — всего лишь прикрытие!

Остается надеяться, что рыночная доля x86-64 **_никогда_** не окажется настолько значительной, чтобы Microsoft смогла похоронить 32-версию Windows, лишив нас возможности выбора, а выбирать следует именно x86, тем более, что Intel выпустила удачную линейку двуядерных процессоров Pentium 4 D (впрочем, на поклонников продукции AMD этот призыв не распространяется).

vista x86-64 – nightmare edition

В 64-разрядной редакции висты (работающей на платформе AMD x86-64) появилось множество "улучшений", отсутствующих в 32-битной версии. Microsoft полностью пересмотрела политику безопасности, надежно защитив ядро от… легальных пользователей системы, в том числе и администраторов, при этом оставив достаточное количество лазеек для малвари.

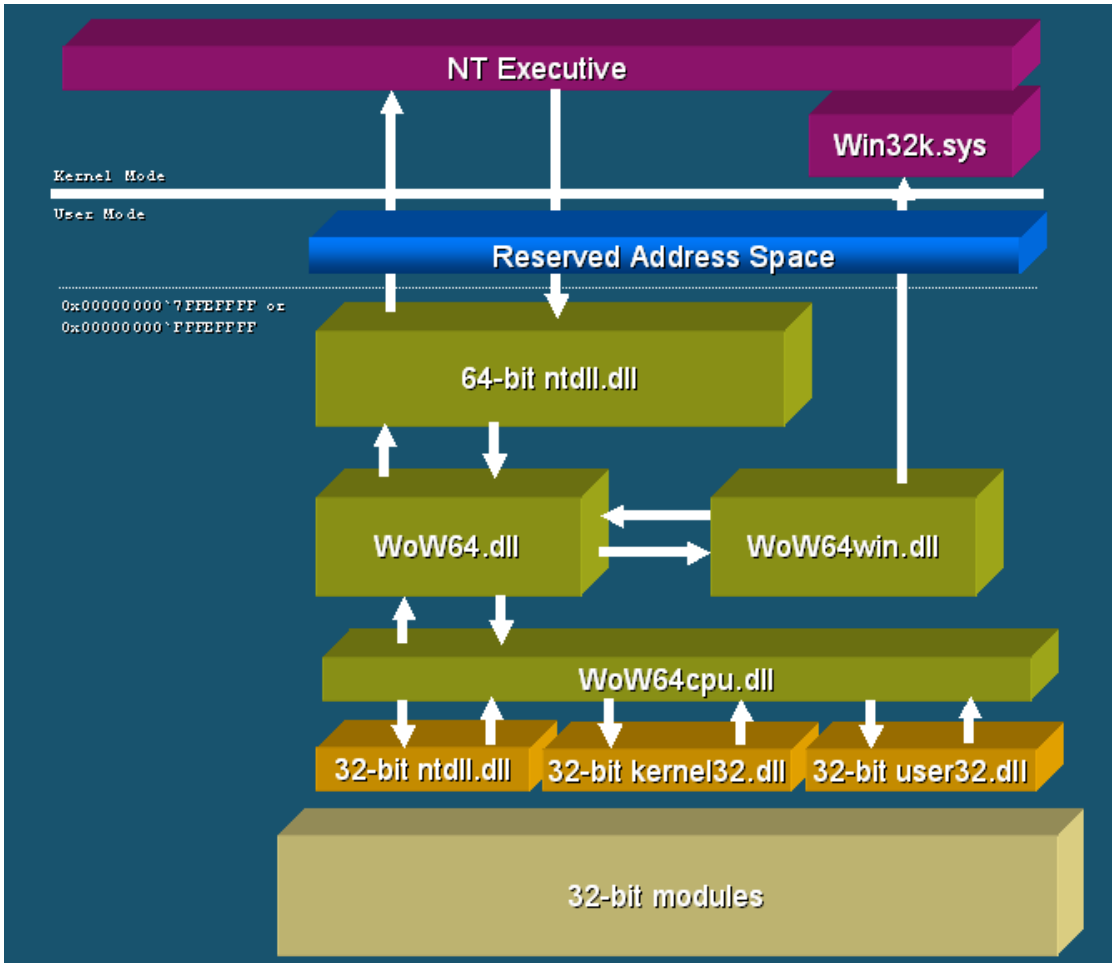

Рисунок 5 архитектура ядра 32-разрядной версии висты

Вот две ключевые технологии, впервые появившиеся еще в XP/Server 2003 SP1, но анонсированные только с приходом висты: **контроль целостности ядра** и **обязательное требование цифровой подписи для всех драйверов**.

Начнем с контроля целостности ядра, для легитимного взаимодействия с которым Microsoft предоставила множество документированных (и еще больше недокументированных) API-функций. Модифицировать ядро, вмешиваясь в его внутренние структуры, _крайне_ нежелательно. Малейшая небрежность проектирования и/или реализации ведет к нестабильной работе системы, голубым экранами смерти, дырам в системе безопасности, а в некоторых случаях и потере всех данных. Проанализировав отчеты об ошибках, Microsoft пришла к выводу, что в большинстве сбоев Windows виновата не она, а программное обеспечение, созданное сторонними разработчиками, модифицирующими ядро "пионерскими" способами, то есть без просчета последствий всех возможных ситуаций.

Технология **Patch-Guard**, реализованная на x86-64 системах, призвана положить конец этому безобразию раз и навсегда. 32-битную версию Microsoft решила не трогать, поскольку в противном случае огромное количество программ тут же бы отказали в работе (подробности можно найти на blog'e одного из сотрудников Microsoft, занимающегося этой проблемой: http://blogs.msdn.com/windowsvistasecurity/archive/2006/08/11/695993.aspx).

Как известно, операционные системы семейства NT используют два кольца защиты из четырех, предоставляемых процессорами семейства x86. Почему? Дело в том, что NT изначально проектировалась как переносимая система, а некоторые из процессоров, на которые ее планировалось перенести, содержали только два кольца, вынуждая разработчиков ориентироваться на наиболее "спартанскую" конфигурацию.

При переносе системы на платформу x86-64 у Microsoft появился реальный шанс забить на "спартанские" конфигурации, давно умерших процессоров и "развести" ядро, драйвера и прикладной код по трем разным кольцам, защитив ядро от пагубных воздействий драйверов на аппаратном уровне на все 100%, однако… Microsoft пошла своим путем, ограничилась периодической проверкой целостности основных структур, вызывая "сторожевую" процедуру

приблизительно один раз в 5-10 секунд. Хорошая получилась защита, нечего сказать… Малварь буквально рыдает от счастья. 5 секунд — это же целая вечность для процессора, успевающего выполнить за это время миллионы машинных команд, с легкостью отключающих Patch-Guard, поскольку защита и зловредный код обладают одинаковыми привилегиями. Так что на хакеров эта защита не распространяется (описание техники обхода Patch-Guard'a можно найти в статье "Bypassing PatchGuard on Windows x64" — http://uninformed.org/index.cgi?v=3&a=3&t=sumry, и в презентации Жанны Рутковской "Rootkit Hunting vs. Compromise Detection", подготовленной для федеральной конференции Black Hat: invisiblethings.org/papers/rutkowska_bheurope2006.ppt — да! да! да! она девушка и хакер одновременно).

А вот легальным разработчикам антивирусов, брандмауэров и прочих программ подобного рода приходится либо сворачивать свой бизнес, либо бухаться в колени к Microsoft и просить предоставить им "ручку" в виде соответствующего вызова API, а в идеале — интегрировать их продукт в ядро (но разработчиков много, а интегрировать можно только одного и вовсе не факт, что он будет лучшим из всех имеющихся). Собственно говоря, кое-какой API для этого появилось еще в NT и всякий желающий мог установить свой собственный фильтр, контролирующий сетевой трафик или содержимое открываемых файлов. Почему же тогда разработчики предпочли модифицировать ядро системы? Да потому, что это надежнее! Легкость установки легального фильтра компенсируется легкостью его снятия, не говоря уже о том, что все имеющиеся на данный момент фильтры работают на довольно высоком уровне, что позволяет зловредным программам легко обходить их! Ладно бы Microsoft закрыла ядро отдельным кольцом, защитив его и от "хороших", и от "плохих" программ. Так ведь нет! Легальные программы вынуждены либо отключать Patch-Guard, что чревато далеко идущими последствиями, либо становится жертвой rootkit'ов.

Чтобы никакой зловредный код не смог пробиться на уровень ядра, Microsoft заблокировала загрузку драйверов без цифровой подписи. Даже обладая администраторскими правами владелец системы не может загрузить неподписанный драйвер. Снять блокировку можно тремя путями: подключить ядерный отладчик, при старте системы нажать <F8> или отредактировать опции загрузки в boot.ini (стоп! в висте уже нет boot.ini и опции загрузки хранятся в бинарном формате, манипулировать которым можно штатной утилитой BCDedit). Так же, в состав SDK входит тестовая цифровая подпись, содержащая слово "test" и предназначенная исключительно для отладочных целей. Ни один из этих способов для коммерческих продуктов, разумеется, не пригоден и финальная версия драйвера, должна быть подписана полноценной цифровой подписью, которую в настоящей момент уполномочена выдавать только одна компания — Verisign. Сертификат начального уровня стоит $500 и выдается только американским фирмам или фирмам, имеющим свое представительство в США. Подробности о политике цифровой подписи читайте в официальном документе "Kernel-Mode Code Signing Walkthrough" от Microsoft: http://download.microsoft.com/download/9/c/5/9c5b2167-8017-4bae-9fde-d599bac8184a/KMCS_Walkthrough.doc.

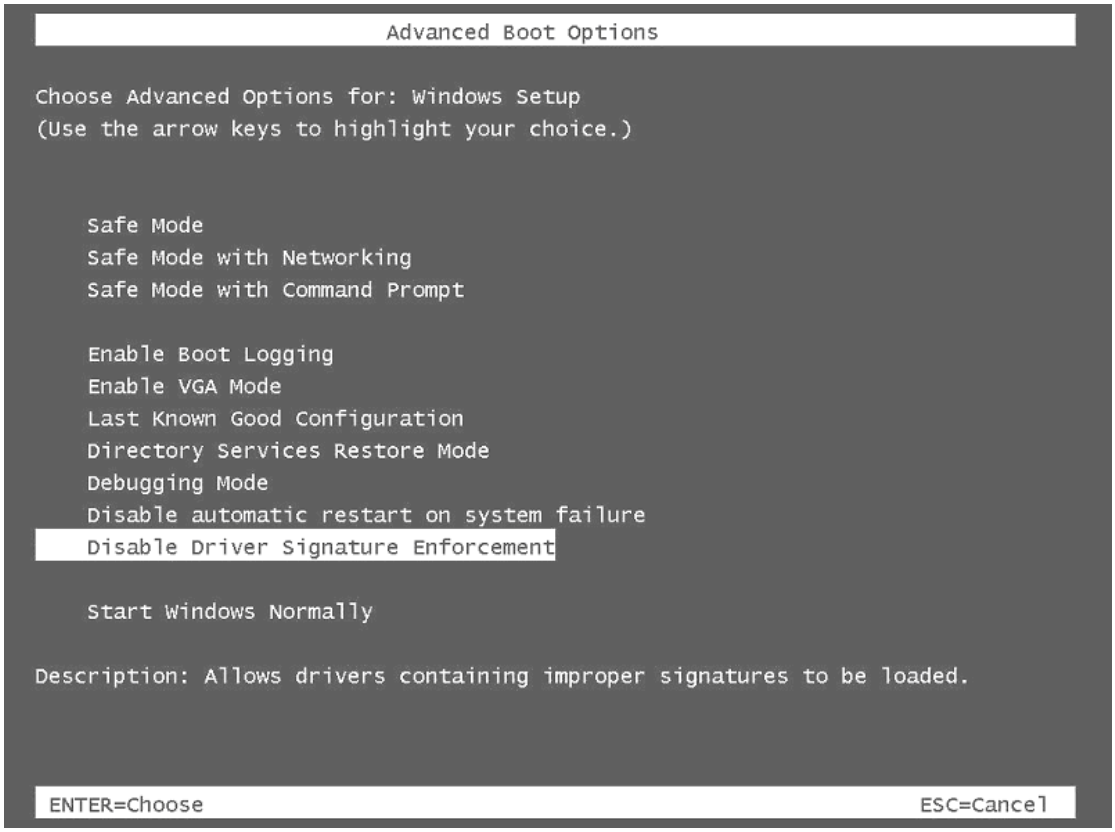

Рисунок 6 отключение проверки цифровой подписи драйверов при загрузке системы

Логика Microsoft такова: не будет сертификата — не будет и подписи, а раз не будет подписи, хакер не сможет загрузить зловредный драйвер, модифицирующий ядро и устанавливающий rootkit, скрывающий малварь от глаз администратора. На самом деле, борьба с малварью никогда не поднималась на такой высокий уровень, да и процедура сертификации носит чисто формальный характер, порочность которого уже была продемонстрирована компонентами ActiveX – вы доверяете фирме "John Doe" из местечка хухры-мухры?

Просто Microsoft хочет укрепить свои позиции на рынке, вытесняя сторонних разработчиков и позиционируя свою платформу как идеальное средство для просмотра premium media content'a следующего поколения. Фактически, все изменения в висте крутятся вокруг DRM — Digital Rights Management — Управление Цифровыми Правами. Microsoft гарантирует, что зашифрованный цифровой медиа-поток данных нигде не будет перехвачен злостными пиратами. Ерунда, конечно. Сграбить его — плевое дело (и такие утилиты _уже_ написаны), а вот у легальных пользователей системы появляются огромные проблемы. Даже если они не смотрят фильмы, и не слушают музыку, все равно они вынуждены мириться с многочисленными ограничениями, налагаемыми этими технологиями.

Виста — это первая система, в который администратор не бог, а… образно говоря, заключенный. Пускай, даже самый старший среди всех заключенных. Что это меняет? Свобода в обмен на… эй, кто там сказал "безопасность"?! Отсутствие рычагов управления делает администратора безвластным и неспособным обнаружить присутствие чего-то постороннего, тем более, что и обнаруживать-то его нечем. Все защитные средства (антивирусы, брандмауэры) вынуждены работать на высоком уровне через скудный набор API-функций и зловредному вирусу ничего не стоит "поднырнуть" под них и как следует замаскироваться. Ничего не напоминает? Ты (администратор) видишь сурка? Вот, и я (антивирус) не вижу. А он есть!

Пробиться на уровень ядра можно и без цифровой подписи, что наглядно продемонстрировала на американской конференции Black Hat Жанна Рутковская, воспользовавшись тем, что файл подкачки доступен на секторном уровне через устройство "\\.\C:" предварительно запустив программу, "скушавшую" всю доступную память и заставившую операционную систему вытеснять код драйверов на диск: http://www.invisiblethings.org/papers/joanna%20rutkowska%20-%20subverting%20vista%20kernel.ppt;

И хотя реакция Microsoft была на удивление спокойной (подумаешь, подломали бету!), хакеры уже потирают руки и сворачивают штопором хвост в предвкушении новой серии атак, а производители железа и разработчики драйверов пьют горькую, матерясь всеми словами которые только знают (а, заодно, изобретают много новых слов), прикидывая во что им обойдется перенос уже отлаженного кода на новую систему и его сертификация. Многие системные программисты окажутся выдавлены с рынка. Пользователям придется обновить железо, а вместе с ним и значительную часть своих любимых программ, многие из которых уже давно заброшены и не поддерживаются.

Вот такая, значит, напряженная ситуация. Конечно, с течением времени все эти проблемы будут обходится всеми возможными путями. В сети появится множество программ, отключающих ненужные защитные механизмы и возвращающие администратору все необходимые права. Производительность железа через несколько лет возрастет настолько, что системным требованиям висты будет удовлетворять даже самый дешевый компьютер. К тому же, технологии виртуализации, уже появившиеся в процессорах Intel Pentium (Vanderpool)/AMD Althorn (Pacifica) и поддерживаемые, в частности, VM Ware 5.5, увеличивают скорость аппаратной эмуляции во много раз, позволяя запускать несколько операционных систем одновременно. Это снимает проблему (не)совместимости программного обеспечения, но оставляет машину уязвимой перед сетевыми атаками.

Лично для себя мыщъх решил, что будет сидеть на w2k столько, сколько это вообще возможно, после чего мигрирует на FreeBSD, где царит полная свобода, где решения принимаю я, а не парни из Реймонда!

>>> супер-врезка цели Microsoft

- защита от малвари (прикрытие);
- упрочнение своих позиций и вытеснение сторонних разработчиков с рынка;
- позиционирование своей системы как защищенной от грабежа premium content'а;

генеалогия висты

Анатомически, виста представляет собой слегка "доработанное" ядро **Server 2003 SP1** (чем, собственно, и объясняется ее ярко выраженная серверная ориентация), с переписанным сетевым стеком, новым пользовательским интерфейсом и кучей выброшенных вещей, в частности: исчез "продвинутый пользовательский интерфейс", теперь есть только один тип интерфейса — "для дэбилов", Windows Messenger был удален без какой-либо замены, эту же участь разделил NetMeeting, вытесненный Windows Meeting Space; Microsoft наконец-то отодрала Internet Explorer и теперь он уже не часть системы, а отдельный компонент, за который по-видимому придется платить конкретные деньги; популярная тема "Luna" оказалась приговоренной к расстрелу без объяснения причин и без всякого следствия (ну кому она в самом деле мешала?! или на DVD места не хватило?). Протокол MS-CHAP v1 более не поддерживается, как не поддерживаются материнские платы без ACPI и ворох другого "морально устаревшего железа".

Другими словами, **Виста — это гибрид, полученный путем скрещивания изуродованного Server 2003 S1 с урезанной и покаленной XP.** Полный перечень отсутствующих фич можно найти на бесплатной энциклопедии wikipedia: http://en.wikipedia.org/wiki/Features_new_to_Windows_Vista#XP_features_excluded;

А вот обещанная и широко разрекламированная файловая система WinFS, к счастью, так и не была реализована ("к счастью" потому, что в противном случае мы бы теряли сотни гигабайты данных, только потому что кого-то не научили программировать), подробнее см: http://en.wikipedia.org/wiki/WinFS.

Другая, чуть менее разрекламированная фича под названием NGSCB (Next-Generation Secure Computing Base — Компьютерная База Безопасности Нового Поколения) разделила туже участь: http://en.wikipedia.org/wiki/Next-Generation_Secure_Computing_Base;

>>> врезка виста - внутри яйца

Сроки выхода операционной системы, ранее известной под кодовым именем Longhorn, переносились неоднократно, уже написанный код хоронился заживо и вновь переписывался с нуля. Поговаривали даже, что Longhorn не выйдет _никогда_ а если и выйдет, то на это уродище никто добровольно не перейдет. Поскольку промежуточные билды просачивались в

файлообменные сети с завидной регулярностью, сейчас мы можем восстановить полную хронологию разработки системы: http://en.wikipedia.org/wiki/Development_of_Windows_Vista.

Мнение самих разработчиков о качестве кода можно узнать из жаркой дискуссии, разгоревшейся на blogspot'e (см. http://minimsft.blogspot.com/2006/03/vista-2007-fire-leadership-now.html), большинство постеров по понятным причинам не оставляют своих имен и один черт разберет, кто из них действительно работает в Microsoft, а кто просто прикидывается. Тем не менее, имея живое подтверждение в виде дистрибутива висты на руках, мы можем "фильтровать базар", выделяя из общего гвалта правдивую информацию.

заключение

Установив висту на свой компьютер (вы ведь все равно установите ее, верно?) не обращайте внимание на то, как она тормозит. Это идет индексация всех файлов для быстрого поиска, так что ее низкая производительность в этот момент — не показатель. После завершения индексации с системой будет можно вполне комфортно работать даже на однопроцессорной машине, хотя, кластер, конечно бы не помешал.