

колем дрова без помощи топора

крик касперски, aka мышъх, aka souriz, no-email

если бы Microsoft строила автомобили, или здания... нашей цивилизации уже бы давно пришел конец. а все потому, что Microsoft _никогда_ бы не стоила ни того, ни другого. максимум предоставляемый железобетонный каркас с шасси, на которых инженеры-пионеры цепляют что-то совершенно непотребное. в пресловутых экранах смерти и нестабильной работе Windows в подавляющем большинстве случаев виноваты драйвера _сторонних_ производителей и сейчас мышъх расскажет как их обнаружить и удалить из системы

введение

Дефекты проектирования драйвером могут носить самый разный характер от выпадений в голубой экран смерти (BSOD – Blue Screen of Death), до замедления работы компьютера и странностей поведения некоторых, совсем не связанных с драйвером прикладных приложений.

Голубой Экран Смерти замечателен (без всякой иронии!) тем, что явным образом сигнализирует о наличие серьезной проблемы и дает наводку откуда рыть. Зачастую (но далеко не всегда) имя "пронинившегося" драйвера высвечивается непосредственно в правом верхнем углу Экрана Голубой Смерти ([см. рис. 1](#)), однако, там его может и не быть или, что еще хуже, стоят имя совершенно постороннего драйвера.

```
*** STOP: 0x0000001E (0xC0000005,0xBE2D3000,0x00000000,0x00000000)
KMODE_EXCEPTION_NOT_HANDLED
*** Address BE2D3000 base at BE2D2000, DateStamp 39a998cc - w2k_kill.sys
```

Рисунок 1 Голубой Экран Смерти, высвечивающий имя сбойного драйвера (в данном случае `w2k_kill.sys`)

В частности, на мышъх'иной машине драйвер от видео-карты Matrox G450 имеет тенденцию разрушать базовые структуры графической подсистемы Windows 2000, в результате чего, в BSOD'е отображается имя системного драйвера `win32k.sys` ([см. рис. 2](#)), в котором реализована значительная часть функций USER и GDI и который, естественно, тут совсем ни при чем. Так что интерпретация показаний Экраном Голубой Смерти это и магия, и интуиция, и наука, и искусство — всего понемножку.

```
*** STOP: 0x0000001E (0xC0000005,0xA0016B8E,0x00000001,0x5858585C)
KMODE_EXCEPTION_NOT_HANDLED
*** Address A0016B8E base at A0000000, DateStamp 38438ff7 - win32k.sys
Beginning dump of physical memory
Dumping physical memory to disk: .69
```

Рисунок 2 Голубой Экран Смерти, высвечивающий имя базового системного драйвера (`win32k.sys`), пострадавшего из-за нападки другого драйвера, который нам еще предстоит найти

Помимо дефектов драйверов, Голубые Экраны Смерти могут так же вызываться отказами железа, например, разогнанным процессором, неисправной оперативной памятью, кривым контроллером жесткого диска, не до конца воткнутой в слот PCI-картой, неконтактом в одном из разъемов, плохим блоком питания, "вздутих" электролитических конденсаторов на материнской плате (а дуются они по разным причинам: перегрев от рядом расположенного процессора, недостатком керамических конденсаторов, "недоложенных" производителем, в результате чего ВЧ-составляющая идет через электролит и сильно его разогревает, наконец, утечкой ключевых транзистор в узле стабилизатора).

Поэтому, прежде, чем колоть дрова необходимо убедиться, что железо, на котором мы сидим полностью исправно. А как это можно сделать?

разборки с железом

Голубые Экраны Смерти, вызванные сбоями железа, носят стихийный характер, появляясь в непредсказуемое время, не связанное ни с какими конкретными действиями пользователя. Прикладные приложения так же начинают выдавать критические ошибки в самых разных местах, причем (и это самое главное!) коды ошибок, адреса и другая информация выдаваемая системой во всех случаях будет различной!!! (Кстати говоря, драйвера, обрабатывающие асинхронные запросы от устройств ввода/вывода, например беспроводных сетей, ведут себя практически точно так же). Голубые Экраны Смерти, вызванные дефективными драйверами, как правило, возникают при совершении определенного набора действий и содержат более или менее постоянную информацию.

Чтобы снять с железа все подозрения, достаточно подключить к системе еще один жесткий диск, установить на него девственно чистую Windows и поработать на ней некоторое время. Если Голубые Экраны Смерти не исчезнут, значит, действительно, виновато железо и его пришла пора менять. Поиск дефективных компонентов — тема для отдельного разговора, который мы оставим на следующий раз, а пока, засучив рукава, вплотную возьмемся за эти коварные драйвера.

дрова без сертификата сразу в топку

Весь комплект инструментария, необходимый для разработки драйверов (DDK – Driver Development Kit) Microsoft бесплатно распространяет бесплатно вместе с сопутствующей ему документацией, позволяющий любому заточить "как бы" драйвер, с позволения сказать, "как бы" работающий, то есть, ни хвоста не работающий, а постоянно падающий.

Вот чтобы этого беспредела не происходило, Microsoft еще в годы неопалита ввела процедуру сертификации драйверов на соответствие предъявляемых к нему требованиям, после которой драйверу выдавалась цифровая подпись, или... не выдавалась и он отправлялся на доработку. И хотя сертификация — всего лишь формальная процедура, не гарантирующая отсутствие фатальных ошибок и дефектов разработки, часть откровенно "пионерских" драйверов она все-таки отсеивает.

В идеале, в системе следует держать только драйвера, заверенные цифровой подписью. И хотя цифровая подпись - не страховой полис, ее наличие уже указывает на определенный уровень культуры разработки. Драйвера без цифровой подписи — это хуже, чем кот с кошкой в мешке и от них по возможности следует избавляться (тем более, что многие из них являются зловредными программами, устанавливаемыми rootkit'ами или агрессивными защитными механизмами, глубоко проникающих в систему и вызывающих ее нестабильность). Короче, не будет разводить демагогии, а попробуем ответить на один простой вопрос — как составить список драйверов без цифровой подписи?

В этом нам поможет утилита "sigverif.exe", входящая в штатный комплект поставки операционной системы и располагающаяся в каталоге WINNT\System32. Запускаем ее и видим следующее диалоговое окно (см. рис. 3).

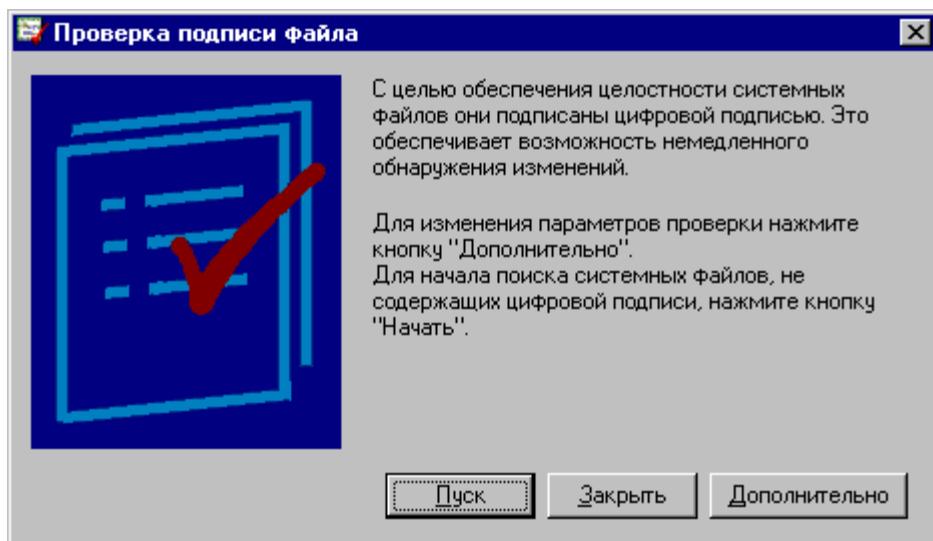


Рисунок 3 штатная утилита sigverif.exe для проверки цифровой подписи драйверов и прочих системных файлов

Нажимаем кнопку "Дополнительно" и во вкладке "Поиск" настраиваем критерии отбора, перемещая радио-кнопку из положения "уведомлять о неподписанных системных файлах" (где она и прозябала по умолчанию) в положение "искать другие файлы, не подписанные цифровой подписью" (см. рис. 4), после чего в "параметрах поиска" открываем бокс "искать файлы следующего типа" и выбираем "*.sys", а ниже указываем папку для поиска "C:\WINNT", обязательно отметив галочку "включая подпапки".

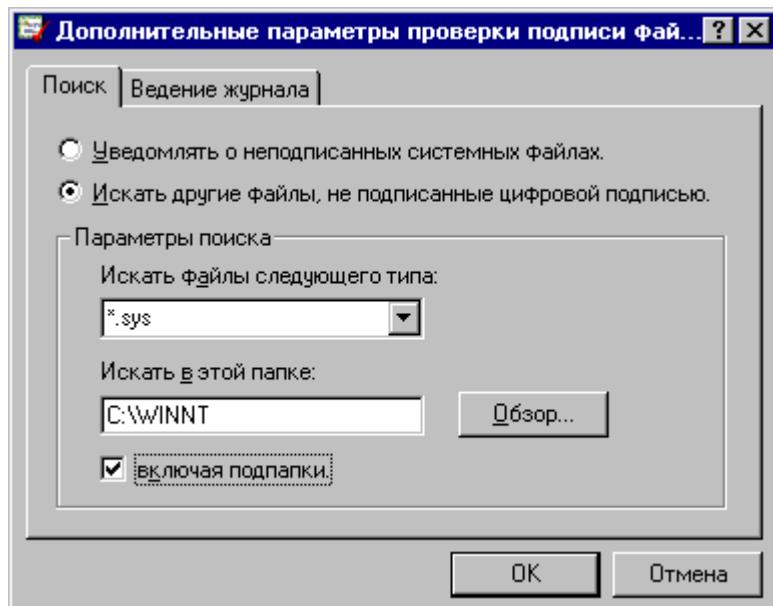


Рисунок 4 задание параметров поиска неподписанных драйверов

Вообще-то, строго говоря, драйвера не обязаны иметь расширение sys и далеко не всегда ограничиваются каталогом "WINNT", находясь в каталогах "своих" приложений, а некоторые приложения и вовсе хранят драйвера... внутри себя! Сразу же после запуска (или в любое другое время) они сохраняют файл на диск в текущую или временную директорию, загружают драйвер в память и... тут же удаляют его с диска! Так поступают не только зловредные вирусы, но и вполне респектабельные программы, вроде некоторых утилит известного исследователя недр Windows Марка Руссиновича.

Поэтому, для чистоты эксперимента нам совсем не помешает получить список драйверов, находящийся в данный момент в памяти и сравнить их с драйверами, находящимися на диске. Слова "в данный момент" ключевые, поскольку, загрузка/выгрузка драйверов может происходить бесплатно без перезагрузки операционной системы, эту операцию желательно выполнить несколько раз, запуская утилиту командной строки "drivers.exe", входящую в состав DDK, который можно бесплатно скачать с сервера компании Microsoft.

Запущенная без каких-либо ключей команной строки, утилита drivers.exe вываливает всю информацию на экран, что не есть хорошо, поскольку драйверов в системе обычно присутствует очень много и на экран они не помещаются (см. рис. 5), однако, религия нам позволяет перенаправить поток вывода в текстовой файл ("drivers.exe >file-name.txt"), открываемый любым текстовым редактором — хоть Word'ом, хоть "Блокнотом". Затем остается только выделить вертикальный блок (чего "Блокнот" не позволяет) и получить список драйверов. Готовенький! Только что из топки! В смысле прямо из ядра операционной системы!

\$drivers.exe								
ModuleName	Code	Data	Bss	Paged	Init		LinkDate	
ntoskrnl.exe	443008	96064	0	749120	136640	Fri May 06 15:44:59 2005		
hal.dll	26080	6048	0	16608	10272	Fri Mar 21 05:04:40 2003		
BOOTVID.dll	5664	2464	0	0	320	Thu Nov 04 04:24:33 1999		
ACPI.sys	92192	9024	0	43488	4448	Wed Jan 15 22:44:22 2003		
WMILIB.SYS	512	0	0	1152	192	Sat Sep 25 22:36:47 1999		
pci.sys	12992	1536	0	31712	4672	Wed Jan 15 22:44:07 2003		
isapnp.sys	14368	832	0	22944	2272	Wed Jan 15 22:43:47 2003		
pcidev.sys	672	32	0	0	128	Wed Jan 15 22:43:03 2003		
PCIINDEX.SYS	4544	480	0	11072	1632	Tue Feb 25 21:31:08 2003		
MountMgr.sys	1088	32	0	23072	2240	Tue Aug 16 12:40:55 2005		
ftdisk.sys	4832	32	0	96000	3392	Fri Dec 03 06:29:58 2004		
Diskperf.sys	1952	32	0	2016	1120	Thu Feb 13 00:34:38 2003		
dmload.sys	2848	64	0	0	608	Wed Jan 15 22:47:06 2003		
dmi.sys	106464	15168	0	0	2752	Wed Jan 15 22:47:04 2003		
PartMgr.sys	576	0	0	7008	1376	Wed Jan 15 22:43:07 2003		
ElbyVCD.sys	10624	128	0	0	1152	Thu Nov 28 13:43:49 2002		
SCSIPORT.SYS	23392	384	0	36640	4064	Thu Jul 14 16:24:06 2005		
atapi.sys	44000	3392	0	22112	8128	Tue Apr 01 22:08:25 2003		
disk.sys	9920	224	0	10976	4832	Wed Jan 15 22:43:05 2003		
CLASSPNP.SYS	14848	64	0	11744	2368	Wed Jan 15 22:42:51 2003		
fltmgr.sys	36928	2304	0	66912	8576	Thu Apr 14 10:59:00 2005		
00Drvled.sys	4864	128	0	4992	2176	Wed Sep 22 15:57:13 2004		
KSecDD.sys	22592	6752	0	33216	1984	Sun Sep 21 04:32:19 2003		
Ntfs.sys	70560	5920	0	394464	12960	Tue May 10 13:20:29 2005		
NDIS.sys	13536	1344	0	130464	6016	Wed Apr 30 03:05:01 2003		
pba.sys	11392	128	0	1536	1376	Wed Jan 15 22:43:26 2003		
Mup.sys	7840	6752	0	62912	3296	Fri Dec 03 06:37:23 2004		
btkrnl.sys	173184	1054400	0	0	1280	Wed Jan 21 05:40:45 2004		
VIDEOPR.T.SYS	6272	96	0	31104	4192	Wed Jan 15 22:47:20 2003		
g400m.sys	68896	45312	0	191616	2144	Thu Aug 17 20:41:50 2000		
ks.sys	22944	64	0	70176	4032	Wed Apr 16 08:02:11 2003		
portcls.sys	41248	10528	0	70144	4224	Wed Apr 16 08:11:22 2003		
sbpcl.sys	27776	9568	0	0	1600	Tue Apr 04 11:47:08 2000		
RTL8139.SYS	11776	512	0	1024	1536	Wed Aug 18 22:33:29 1999		
fdc.sys	18080	256	0	320	3872	Wed Jan 15 22:42:51 2003		
parport.sys	16512	480	0	288	1824	Wed Jan 15 22:47:13 2003		
serial.sys	8768	288	0	31584	9504	Wed Apr 16 08:19:39 2003		

Рисунок 5 построение списка загруженных драйверов с помощью утилиты drivers.exe из комплекта поставки DDK

Если хотя бы один из этих драйверов отсутствует в каталоге C:\WINNT\, то его цифровая подпись проверена не будет!!! Естественно, такой драйвер сразу же привлекает к себе внимание и у нас появляется резонный вопрос — откуда он берется? Сначала сканируем все каталоги на диске и если его там нет, устанавливаем точку останова на функцию CreateFileW в soft-ice и смотрим на передаваемые ей аргументы. Рано или поздно мы встретим наш "незаконнорожденный" драйвер, после чего останется только взглянуть в правый нижний угол экрана soft-ice, где высвечивается имя процесса, породившего его. Впрочем, описание процесса поиска "незаконнорожденных" драйверов выходит за рамки данной статьи. Интересующихся мы отошлем к мышьхиной книге "техника отладки программ без исходных текстов", электронную копию которой можно найти в мышьхиной норе на ftp сервере nezumi.org.ru, так же в процессе написания этой статьи, мышьх воздвиг http сервер, находящийся по тому же адресу — nezumi.org.ru — и пока работающий в тестовом режиме. Так что желающие приглашаются на раскурку травы, а с остальными мы продолжим терзать утилиту sigverif.exe.

После нажатия на "OK", "Пуск" на экране появится градусник, отображающий ход прогресса и жесткий диск начнет шуршать всеми своими головками, какие у него только есть (см. рис. 6).

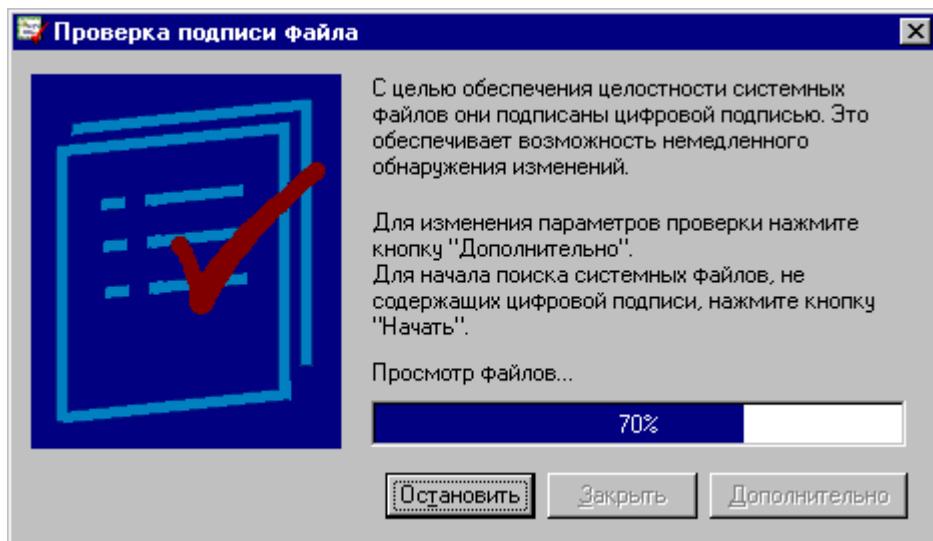


Рисунок 6 отображения хода проверки цифровой подписи драйверов

По завершении работы будет составлен список драйверов _без_ цифровой подписи и выведен на экран (см. рис. 7).

Имя	В папке	Изменен	Тип файла	Версия
cnphetu.sys	c:\winnt\system32\...	31.07.2003	Системный файл	32.99.80.0
cnxtlgn.sys	c:\winnt\system32\...	01.08.2003	Системный файл	29072.99.80.0
elbycodfl.sys	c:\winnt\system32\...	28.11.2002	Системный файл	4.2.0.0
elbycodio.sys	c:\winnt\system32\...	29.11.2002	Системный файл	4.2.0.0
elbyvcd.sys	c:\winnt\system32\...	28.11.2002	Системный файл	4.2.0.0
frmupgr.sys	c:\winnt\system32\...	20.01.2004	Системный файл	1.4.2.8
g400m.sys	c:\winnt\system32\...	17.08.2000	Системный файл	5.12.1.1130
hcmon.sys	c:\winnt\system32\...	10.06.2004	Системный файл	4.5.2.8848
irstusb.sys	c:\winnt\system32\...	29.05.2003	Системный файл	1.26.0.0
ntice.sys	c:\winnt\system32\...	18.04.2005	Системный файл	4.0.1381.1
oodrvled.sys	c:\winnt\system32\...	22.09.2004	Системный файл	1.0.0.275
siwsysm.sys	c:\winnt\system32\...	25.01.2000	Системный файл	Отсутствует
siivid.sys	c:\winnt\system32\...	25.01.2000	Системный файл	4.0.1381.1
teefer.sys	c:\winnt\system32\...	03.10.2001	Системный файл	1.60.1101.0
utilnt.sys	c:\winnt\system32\...	17.04.2000	Системный файл	1.0.0.12
vmnetbridge....	c:\winnt\system32\...	10.06.2004	Системный файл	3.0.0.0
vmnetuserif....	c:\winnt\system32\...	10.06.2004	Системный файл	3.0.0.0
vmparport.sys	c:\winnt\system32\...	10.06.2004	Системный файл	4.5.2.8848

Найдено файлов: 368, Подписано: 333, Не подписано: 35, Не просмотрено: 0.

Рисунок 7 список драйверов, не заверенных цифровой подписью

Некоторые горячие головы предлагают в порядке очищения системы от ереси, удалить все неподписанные драйвера, тогда, мол, все проблемы как хвостом снимет. А как их можно удалить? Самое грубое решение — просто взять их и удалить с диска через FAR'a или проводника (естественно, обладая правами администратора!), но... последствия такой операции могут оказаться весьма плачевными и лучше, кликнув правой клавишей мыши на иконку драйвера в проводнике, найти в "свойствах" имя производителя, по которому можно установить что за приложение/железка установила этот драйвер и де-инсталлировать ее цивилизованным путем, правда, здесь есть одно "но".

На приведенном виде [рисунке 7](#), выделен драйвер g400m.sys, идущий вместе с картой Matrox G450, и несмотря на то, что Matrox совсем не хилая компания, цифровую подпись она не

получила (то ли Microsoft не дала, то ли сама Matrox не захотела заморачиваться). Естественно, после удаления его из системы, о SVGA режиме придется забыть. Можно, правда, сходить к самому Matrox'у, скачав самую последнюю версию драйвера (она уже снабжена цифровой подписью), только вот... и подписанная, и неподписанные версии содержат множество фатальных ошибок, в частности, при стечении определенных обстоятельствах при попытке перейти в overlay mode, система падает в BSOD, поскольку драйвер пытается освободить уже освобожденную память.

Таким образом, наличие/отсутствие цифровой подписи само по себе еще ни о чем не говорит и даже если мы используем только подписанные драйвера, никаких гарантий стабильности это нам не дает.

Вот тут-то мы и переходим ко второй части статьи, а именно — тестированию драйверов в условиях, приближенным к боевым.

устраиваем дровам настоящее испытание

В состав DDK входит замечательная утилита "Driver Verifier", ставящая драйвера раком, то есть создающая для них максимально суровые условия, граничащие с суицидом и экстремалом, в которых вероятность отказа — максимальна, а имя дефективного драйвера определяется с наивысшей точностью (даже если он из-за дефектов разработки не страдает сам, но рушит структуру данных чужих драйверов).

Важно отметить, что "Driver Verifier" — это не лекарство, а только средство диагностики. От сбоев оно все равно не спасет (напротив, увеличит их интенсивность на пару порядков), но зато поможет выявить "подлый" драйвер с достаточной степенью достоверности.

Итак, запускаем "verifier.exe", видим окно "Driver Verifier Manager", идем в закладку "Setting" (см. рис. 8) и переводим радио-кнопку в положение "Verify all drivers", после чего давим кнопку "Preferred Setting", устанавливающую следующие типы проверок ("verification type"): "Special pool" — проверяем драйверам будет отведена специальная область памяти для выделения, не очень быстро работающая, зато способная обнаруживать большинство типов разрушений своих и чужих данных; "Force IRQL checking" — IRQL это уровень запроса прерываний (Interrupt Request Level), и наиболее частой ошибкой разработчиков драйверов является попытка обратится к памяти на таком уровне IRQL, на котором менеджер подкачки не работает и если требуемая страница вдруг окажется вытесненной на диск, система обречется в голубой экран с надписью "IRQL_LESS_OR_EQULAR", форсирование этого режима принудительно вытесняет страницы драйвера на диск, чтобы дефект разработки проявлялся в 100% случаев. Так же полезно взвести галочку "Low resource simulation", чтобы посмотреть как драйвер будет вести себя при катастрофической нехватке системных ресурсов, однако, этого можно и не делать, а вот галочку "Pool tracking" (отслеживание корректности обращения с пулом памяти лучше оставить). Ошибки ввода/вывода ("I/O verification") составляют ничтожную часть от ошибок других типов, поэтому положение этой галки в общем-то совершенно некритично.

Покончив с выбором настроек, нажимаем кнопку "Apply" (применить) и, как нам и предлагаю, перезагружаемся.

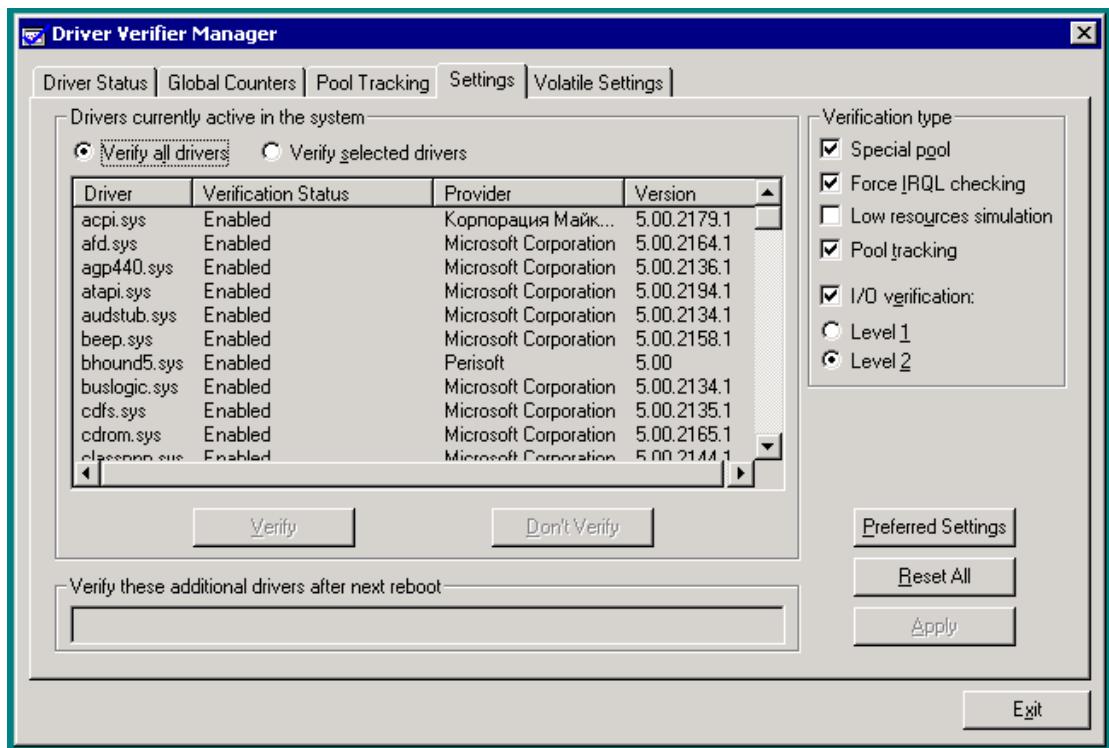


Рисунок 8 выбор проверяемых драйверов и типов проверок

Сразу же после начала загрузки, работа системы _ощутимо_ замедлиться, но это так и должно быть, поскольку сейчас ядро выполняет намного больше проверок чем обычно. При обнаружении ошибок вспыхивает Голубой Экран Смерти с имеем драйвера и некоторой другой информацией, полезной для разработчиков, но бесполезной для нас. Все, что мы можем сделать — обновить драйвер до самой последней версии или отказаться от использования программы (железки), использующей его. Вообще-то, у нас имеется чуть-чуть больше возможностей по розжигу сырых дров, но об этом — чуть позже.

Узнать статус проверки можно в любой момент запуском verifier.exe. В закладке "Driver Status" перечислен статус всех обнаруженных драйверов с пояснением текущей оперативной ситуации. Статус "Loaded" означает, что данный драйвер был загружен и проверен по крайней мере один раз (но, возможно, не полностью, т. е. не все участки драйвера успели отработать). Статус "Unloaded" готовит о том, что драйвер был загружен, проверен (по крайней мере частично) и выгружен системой/программой, использующей его, или по своему собственному желанию. Последнее особенно характерно для драйверов, оставшихся от оборудования которое было удалено путем варварского выдергивания платы расширения из слота, т. е. без выполнения деинсталляции. Оставшийся в живых драйвер сканирует шину, пытаясь нащупать "свое" оборудование, обламывается с поиском, после чего выгружает себя из памяти, кстати говоря, замедляя загрузку системы (иногда очень значительно) и конфликтую с другими драйверами. Однако, не всякий статус "unloaded" признак ненормальности ситуации и прежде, чем удалять такой драйвер, сначала нужно разобраться что это за северный олень такой и откуда от вообще тут взялся, понимаешь.

Статус "Never Loaded" указывает на то, что данный драйвер еще не был загружен, а значит, не был и проверен, следовательно, надо подождать, запуская различные программы, которые могут быть с ним связаны. Впрочем, некоторые драйвера (особенно некорректно деинсталлированные) не загружаются и, соответственно, не проверяются никогда.

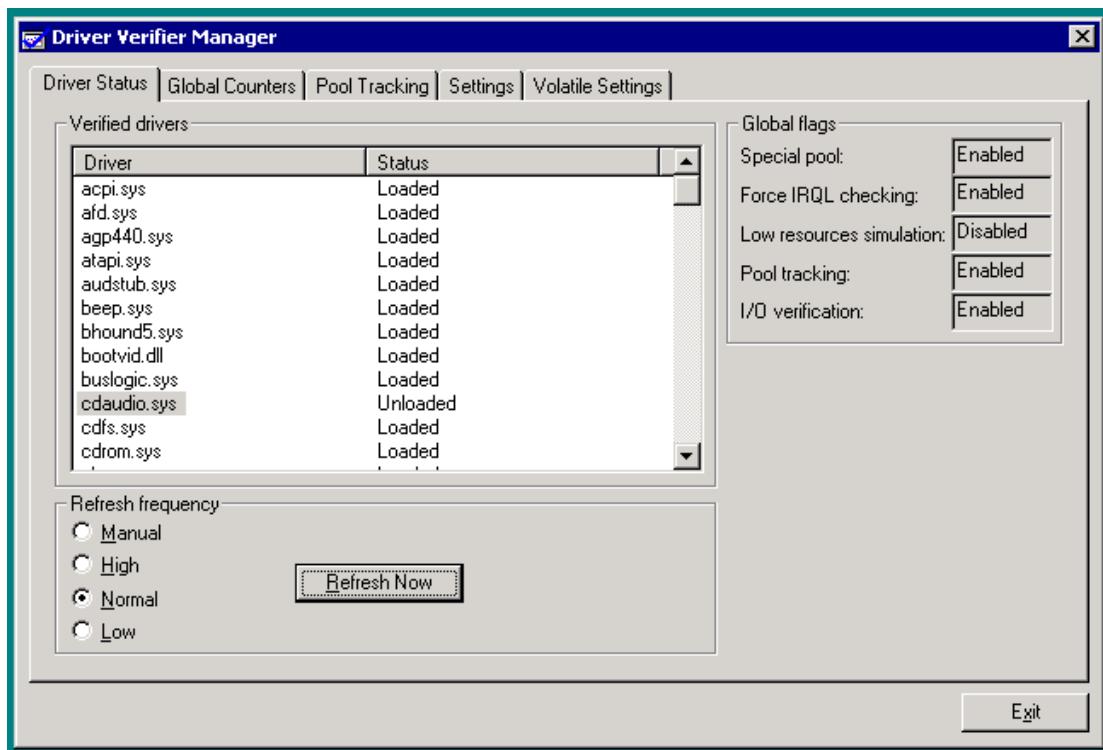


Рисунок 9 просмотр текущего статуса проверки

Поработав с системой в режиме "жесткой проверки" некоторое время (от нескольких часов до нескольких дней) мы выявим практически все дефектные драйвера, от которых страдали ранее, записав их имена на бумажку.

Вернуть систему нормальный режим (т. е. без дополнительных проверок, снижающих производительность), можно с помощью все того же verifier'a. Возвращаемся к закладке "Setting", переводим радио-кнопку в положение "Verify selected drivers" (при этом никакой драйвер не должен быть выделен), давим на "Reset All", затем "Apply" и перезагружаемся. Все! Теперь система работает с нормальной скоростью, но без проверок.

Что делать с сырьими дровами?

А, действительно, что можно сделать с дефектным драйвером? Хакеры, умеющие держать отладчик в руках, при наличии достаточно количества свободного времени, могут дезассемблировать драйвер (благо, по объему драйвера обычно небольшие), найти ошибку, и придумать способ как ее исправить, но... это слишком сложный и трудоемкий путь.

Выбрасывать драйвер (вместе с тем железом/программой, что его использует) тоже не вариант, хотя... если известно, что в Голубых Экранах Смерти виновата звуковая карта неизвестного китайского производителя за 20\$, то у нас появляется вполне весомая мотивация ее заменить на что-то более достойное. Но это, собственно говоря, всем и так понятно и в дополнительных комментариях не нуждается.

Зато далеко не каждый знает, что огромное количество сбоев и Голубых Экранов Смерти связано с тем, что драйвер, разработанный (и протестированный) в однопроцессорной среде, ставят на двухпроцессорную машину. Под "двуихпроцессорностью" здесь понимается как реальная платформа с двумя ядрами, так и Hyper-Threading/многоядерные процессоры. Известно (и подтверждено большим количеством тестов), что домашнему компьютеру два процессора совершенно ни к чему и на подавляющем большинстве приложений увеличение производительности практически не наблюдается.

Поэтому, если система работает нестабильно, а избавиться от дефектного драйвера по тем или иным обстоятельствам никак не удается, можно попробовать залезть в BIOS Setup, превратив свою "виртуальную двухпроцессорную" машину в "однопроцессорную". Аналогичного эффекта можно добиться, открыв файл boot.ini (на компьютерах с Windows NT, Windows 2000 и Windows XP он расположен в корневом каталоге логического диска, на котором установлена система) и добавить к нему ключ /ONECPU (см. листинг 1), после чего перезагрузиться, в надежде, что ошибки исчезнут.

```
[boot loader]
timeout=30
default=multi(0)disk(0)rdisk(0)partition(1)\WINNT
[operating systems]
multi(0)disk(0)rdisk(0)partition(1)\WINNT="Windows 2000 Pro" /fastdetect /SOS
```

Листинг 1 пример типичного файла boot.ini

```
[boot loader]
timeout=30
default=multi(0)disk(0)rdisk(0)partition(1)\WINNT
[operating systems]
multi(0)disk(0)rdisk(0)partition(1)\WINNT="Windows 2000 Pro" /fastdetect /SOS /ONECPU
```

Листинг 2 пример файла boot.ini, настраивающего систему на использование только одного процессора из всех имеющихся

А вот на Windows Vista файла boot.ini нет и хотя существует (временная) возможность сконфигурировать ее загрузочные настройки с помощью специальной утилиты, Microsoft планирует полностью отказаться от этой лазейки, так что остается только BIOS Setup. Впрочем, что касается Vista, то к моменту перехода на нее, разработчики драйверов наверняка обзаведутся многопроцессорными машинами (поскольку, других просто не останется в продаже) и будут тестировать свои извания в многопроцессорном окружении.

Еще один тонкий момент. Помните, мы выше говорили, что наиболее часто встречающаяся ошибка разработчиков драйверов — обращение к вытесняемой памяти на том уровне IRQL, на котором менеджер подкачки не работает и если запрашиваемая страница отсутствует в памяти наступает крах? Очевидным решением будет увеличение оперативной памяти до того объема, при котором вытеснение страниц на диск практически не происходит. При нынешних ценах на память прикупить пару новых "плашек" может позволить себе практически каждый. Но существует и более дешевое (и более элегантное) решение проблемы. А именно — *если параметр "DisablePagingExecutive", находящийся в следующей ветке реестра HKLM\SYSTEM\CurrentControlSet\Control\Session Manager\MemoryManagement, равен 1 (по умолчанию 0), ядерные компоненты вытесняться не будут!* Просто запускаем Редактор Реестра, меняем этот заветный параметр и перезагружаемся (изменения вступают в силу только после перезагрузки), надеясь, что это поможет решить проблему сбоев. А быть может и нет...

ЗАКЛЮЧЕНИЕ

То, что операционные системы семейства Windows (вместе со всем их окружением) падучи и нестабильны — факт, не требующий доказательств. Но вместо того, чтобы ругать Билла Гейтса и криворуких программистов, лучше попробовать разобраться в проблеме и устраниить ее самостоятельно. Это гораздо эффективнее любых матерных слов и проклятий.