Ссылка - <http://z-oleg.com/secur/articles/keylogger.php>

|  |
| --- |
| Клавиатурные шпионы |

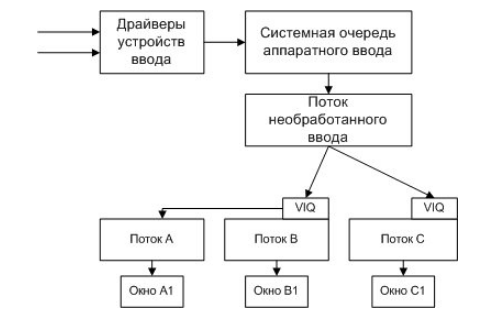
Клавиатурные шпионы образуют большую категорию вредоносных программ, представлющую большую угрозу для безопасности пользователя. Как и Rootkit, о которых  шла речь в предыдущей статье, клавиатурные шпионы не являются вирусами, т.к. не обладают способностью к размножению.

Клавиатурные шпионы - это программа для скрытной записи информации о нажимаемых пользователем клавишах. У термина «клавиатурный шпион» есть ряд синонимов: Keyboard Logger, KeyLogger, кейлоггер; реже встречается термины "снупер", "snoop", "snooper" (от англ. snoop - буквально "человек, вечно сующий нос в чужие дела")

Как правило, современные клавиатурные шпионы не просто записывает коды вводимых клавиш - он "привязывает" клавиатурный ввод к текущему окну и элементу ввода. Кроме того, многие клавиатурные шпионы отслеживают список запущенных приложений, умеют делать "снимки" экрана по заданному расписанию или событию, шпионить за содержимым буфера обмена и решать ряд задач, нацеленных на скрытное слежение за пользователем. Записываемая информация сохраняется на диске и большинство современных клавиатурных шпионов могут формировать различные отчеты, могут передавать их по электронной почте или http/ftp протоколу. Кроме того, ряд современных клавиатурных шпионов пользуются RootKit технологиями для маскировки следов своего присутствия в системе.

Для системы клавиатурный шпион, как правило, безопасен. Однако он чрезвычайно опасен для пользователя – с его помощью можно перехватить пароли и прочую конфиденциальную информацию, вводимую пользователем. К сожелению, в последнее время известны сотни разнообразных келоггеров, причем многие из них не детектируются антивирусами.

Перед описанием основных принципов работы клавиатурного шпиона необходимо рассмотреть модель аппаратного ввода системы Windows. Достаточно подробное описание этой модели можно найти в книге Д.Рихтера «Windows для профессионалов».



При возникновении неких событии ввода (нажатии клавиш, перемещении мыши) события обрабатываются соответствующим драйвером и помещается в системную очередь аппаратного ввода. В системе имеется особый поток необработанного ввода, называемый RIT (Raw Input Thread), который извлекает события из системной очереди и преобразует их в сообщения. Полученные сообщения помещаются в конец очереди виртуального ввода одного из потоков (виртуальная очередь потока называется VIQ – Virtualized Input Queue). При этом RIT сам выясняет, в очередь какого конкретно потока необходимо поместить событие. Для событий мыши поток определяется поиском окна, над которым расположен курсор мыши. Клавиатурные события отправляются только одному потоку – так называемому активному потоку (т.е. потоку, которому принадлежит окно, с которым работает пользователь). На самом деле это не совсем так -  в частности, на рисунке показан поток A, не имеющий очереди виртуального ввода. В данном случае получатся, что потоки A и B совместно используют одну очередь виртуального ввода. Это  достигается при помощи вызова API функции AttachThreadInput, которая позволяет одному потоку подключиться к очереди виртуального ввода другого потока. Следует отметить, что поток необработанного ввода отвечает за обработку специальных сочетаний клавиш, в частности Alt+Tab и Ctrl+Alt+Del.

# Слежение за клавиатурным вводом при помощи ловушек

Данная методика является классической для клавиатурных шпионов. Суть метода состоит в применении механизма ловушек (hook) операционной системы. Ловушки позволяют наблюдать за сообщениями, которые обрабатываются окнами других программ. Установка и удаление ловушек производится при помощи хорошо документированных функций API библиотеки user32.dll (функция SetWindowsHookEx позволяет установить ловушку, UnhookWindowsHookEx - снять ее). При установке ловушки указывается тип сообщений, для которых должен вызываться обработчик ловушки. В частности, есть два специальных типа ловушки WH\_KEYBOARD и WH\_MOUSE - для регистрации событий клавиатуры и мыши соответственно. Ловушка может быть установлена для заданного потока и для всех потоков системы. Ловушка для всех потоков системы очень удобна для построения клавиатурного шпиона.

Код обработчика событий ловушки должен быть расположен в DLL. Это требование связано с тем, что DLL с обработчиком ловушки проецируется системой в адресное пространство всех GUI процессов. Интересной особенностью является то, что проецирование DLL происходит не в момент установки ловушки, а при получении GUI процессом первого сообщения, удовлетворяющего параметрам ловушки.

На прилагаемом компакт-диске имеется демонстрационный «клавиатурный шпион», построенный на основе ловушки. Он регистрирует клавиатурный ввод во всех GUI приложениях и дублирует вводимый текст на своем окне. Данный пример можно использовать для тестирования программ, противодействующих клавиатурных шпионам.

Методика ловушек достаточно проста и эффективна, но у нее есть ряд недостатков. Первым недостатком можно считать то, что DLL с ловушкой проецируется в адресное пространство всех GUI процессов, что может применяться для обнаружения клавиатурного шпиона. Кроме того, регистрация событий клавиатуры возможна только для GUI приложений, это легко проверить при помощи демонстрационной программы.

# Слежение за клавиатурным вводом при помощи опроса клавиатуры

Данная методика основана на периодическом опросе состояния клавиатуры. Для опроса состояния клавиш в системе предусмотрена специальная функция GetKeyboardState, возвращающая массив из 255 байт, в котором каждый байт содержит состояние определенной клавиши на клавиатуре. Данный метод уже не требует внедрения DLL в GUI процессы и в результате шпион менее заметен.

Однако изменение статуса клавиш происходит в момент считывания потоком клавиатурных сообщений из его очереди, и в результате подобная методика работает только для слежения за GUI приложениями. От этого недостатка свободна функция GetAsyncKeyState, возвращающая состояние клавиши на момент вызова функции.

На прилагаемом компакт-диске имеется демонстрационный «клавиатурный шпион», построенный на основе циклического опроса клавиатуры – приложение KD2.

Недостатком клавиатурных шпионов такого типа является необходимость периодического опроса состояния клавиатуры с достаточно высокой скоростью, не менее 10-20 опросов в секунду.

# Слежение за клавиатурным вводом при помощи перехвата API функций

Данная методика не получила широкого распространения, но тем не менее она может с успехом применяться для построения клавиатурных шпионов. Методики перехвата функций API подробно рассматривались в статье, посвященной RootKit. Разница между RootKit и клавиатурным шпионов в данном случае невелика – шпион будет перехватывать функции с целью мониторинга, а не с целью модификации принципов работы и результатов вызова.

Простейшим способом может быть перехват функций GetMessage, PeekMessage и TranslateMessage библиотеки User32, что позволит вести мониторинг всех сообщений, получаемых GUI приложениями.

# Клавиатурный шпион на базе драйвера

Данный метод еще более эффективен, чем описанные выше методы. Возможны как минимум два варианта реализации этого метода – написание и установка в систему своего драйвера клавиатуры вместо штатного или установка драйвера - фильтра. Применение драйвера - фильтра на мой взгляд является наиболее корректной методикой, хороший вариант реализации описан на сайте www.wasm.ru, другой вариант можно найти в Windows DDK (пример называется kbfiltr).

# Аппаратные клавиатурные шпионы

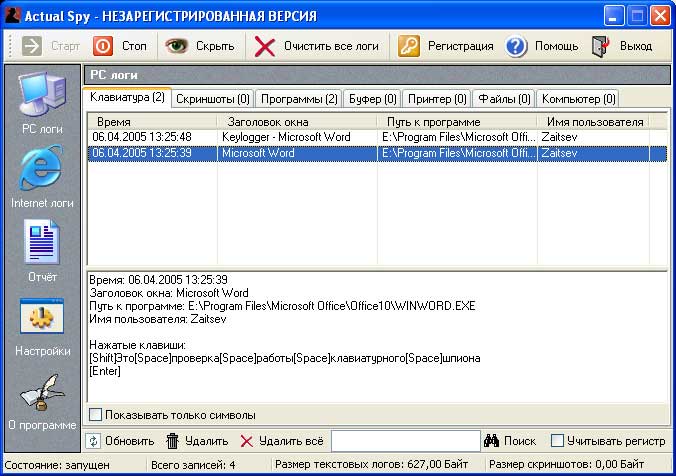
В ходе решения задач по защите от утечки информации часто рассматривают только различные программные средства для шпионажа за работой пользователя. Однако кроме программных возможны и аппаратные средства:

* Установка устройства слежения в разрыв кабеля клавиатуры (например, устройство может быть выполнено в виде переходника PS/2);
* Встраивание устройства слежения в клавиатуру;
* Считывание данных путем регистрации ПЭМИН (побочных электромагнитных излучений и наводок);
* Визуальное наблюдение за клавиатурой

Аппаратные клавиатурные шпионы встречаются намного реже, чем программные. Однако при проверке особо ответственных компьютеров (например, применяемых для совершения банковских операций) о возможности аппаратного слежения за клавиатурным вводом не следует забывать.

# Пример клавиатурного шпиона

В настоящее время сотни клавиатурных шпионов, рассмотрим в качестве примера достаточно распространенную коммерческую программу  ActualSpy

([http://www.actualspy.ru](http://www.actualspy.ru/)). Данная программа может регистрировать клавиатурный ввод (с регистрацией заголовка окна и имени программы), снимать скриншоты экрана по расписанию, регистрировать запуск/останов программ, следить за буфером обмена, принтером, создаваемыми пользователем файлами. Кроме того, в программе реализовано слежение за Интернет-соединениями и посещаемыми сайтами. ActualSpy выбран в качестве примера

Программа имеет простейшую маскировку от обнаружения – она не видна в стандартном списке задач Windows. Для анализа собранной информации программа формирует протоколы в формате HTML. Принцип работы программы ActualSpy основан на ловушке, регистрирующей события клавиатуры.

В качестве других примеров могут выступить

SpyAgent ([http://www.spytech-web.com](http://www.spytech-web.com/)),

ActMon ([http://www.actmon.com](http://www.actmon.com/)),

 SpyBuddy([http://www.actmon.com](http://www.actmon.com/)),

PC Activity Monitor ([http://www.keyloggers.com](http://www.keyloggers.com/)),

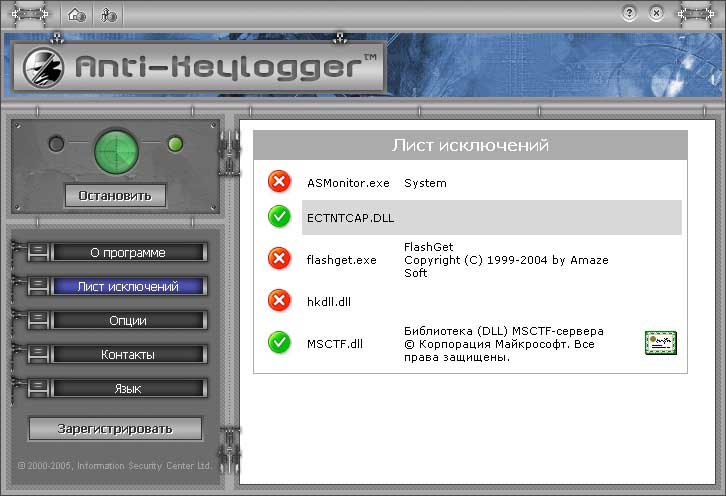
KGB Spy (<http://www.refog.ru/>)…. Этот список можно продолжать очень долго, однако в большинстве случаев современные клавиатурные шпионы имеют примерно одинаковую базовую функциональность и различаются сервисными функциями и качеством маскировки в системе.

# Методики поиска клавиатурных шпионов

1. Поиск по сигнатурам. Данный метод не отличается от типовых методик поиска вирусов. Сигнатурный поиск позволяет однозначно идентифицировать клавиатурные шпионы, при правильном выборе сигнатур вероятность ошибки практически равна нулю. Однако сигнатурный сканер сможет обнаруживать заранее известные и описанные в его базе данных объекты;
2. Эвристически алгоритмы. Как очевидно из названия, это методики поиска клавиатурного шпиона по его характерным особенностям. Эвристический поиск носит вероятностный характер. Как показала практика, этот метод наиболее эффективен для поиска клавиатурных шпионов самого распространенного типа – основанных на ловушках. Однако подобные методики дают много ложных срабатываний. Мои исследования показали, что существуют сотни безопасных программ, не являющихся клавиатурными шпионами, но устанавливающих ловушки для слежения за клавиатурным вводом и мышью. Наиболее распространенный примеры - программы Punto Switcher, словарь Lingvo, программное обеспечение от мультимедийных клавиатур и мышей;
3. Мониторинг API функций, используемых клавиатурными шпионами. Данная методика основана на перехвате ряда функций, применяемых клавиатурным шпионом – в частности, функций SetWindowsHookEx, UnhookWindowsHookEx, GetAsyncKeyState, GetKeyboardState. Вызов данных функций каким либо приложением позволяет вовремя поднять тревогу, однако проблемы многочисленных ложных срабатываний будут аналогичны методу 2;
4. Отслеживание используемых системой драйверов, процессов и сервисов. Это универсальная методика, применимая не только против клавиатурных шпионов. В простейшем случае можно применять программы типа Kaspersky Inspector или Adinf, которые отслеживают появление в системе новых файлов.

# Программы для поиска и удаления клавиатурных шпионов

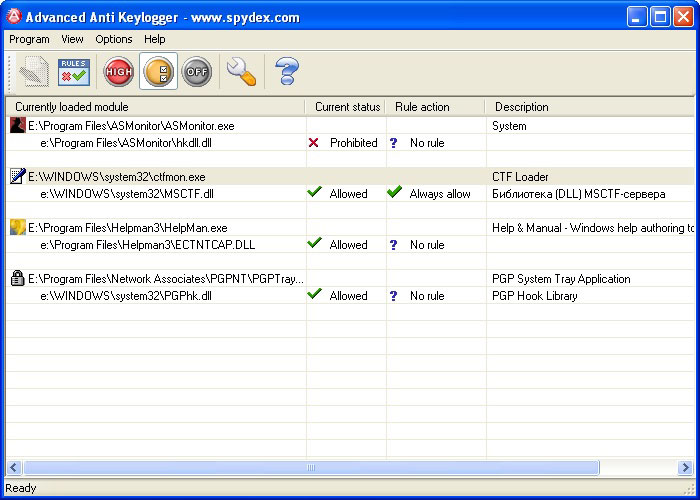
Любой антивирусный продукт. Все антивирусы в той или иной мере могут находить клавиатурные шпионы, однако клавиатурный шпион не является вирусом и в результате пользы от антивируса мало;

Утилиты, реализующие механизм сигнатурного поиска и эвристические механизмы поиска. Примером может служить утилита AVZ, сочетающая сигнатурный сканер и  систему обнаружения  клавиатурных шпионов на базе ловушек;

Специализированные утилиты и программы, предназначенные для обнаружения клавиатурных шпионов и блокирования их работы. Подобные программы наиболее эффективны для обнаружения и блокирования клавиатурных шпионов, поскольку как правило могут блокировать практически все разновидности клавиатурных шпионов.

Из специализированных программ интерес могут представлять коммерческие продукты PrivacyKeyboard и Anti-keylogger (<http://www.bezpeka.biz/>). Интерфейс программы Anti-keylogger показан на рисунке:

Программа Anti-keylogger работает в фоновом режиме и производит обнаружение программ, подозреваемых в слежении за клавиатурой. В случае необходимости можно вручную разблокировать работу любой из обнаруженных программ (например, на рисунке видно, что в список «шпионов» попали MSN Messanger и программа закачки из Интернет FlashGet). Для обнаружение клавиатурных шпионов не применяются базы  сигнатур, обнаружение ведется эвристическими методами.

Тестирование программы показало, что она эффективно противодействует клавиатурным шпионам, основанным на применении ловушек, циклического опроса  и клавиатурного драйвера-фильтра.

Другим примером может служить программа Advanced Anti Keylogger ([http://www.anti-keylogger.net](http://www.anti-keylogger.net/)).

В режиме обучения данная программа по логике работы напоминает Firewall – при обнаружении подозрительной активности выводится предупреждение с указанием имени и описания программы. Пользователь может выбрать действие на сеанс (разрешить, запретить), или создать постоянное правило для правило для приложения.  В ходе тестов Advanced Anti Keylogger уверенно обнаружил все основные разновидности клавиатурных шпионов (на базе ловушки, циклического опроса, драйвера-фильтра). Настройки программы защищаются паролем, который задается в ходе инсталляции.

# Заключение

Клавиатурный шпион не является вирусом, но, тем не менее, представляет большую угрозу для пользователей, поскольку позволяет злоумышленнику следить за работой пользователя и может применяться для похищения конфиденциальной информации, в том числе паролей пользователя. Опасность клавиатурного шпиона может существенно возрасти при его сочетании с RootKit-технологией, которая позволит замаскировать присутствие клавиатурного шпиона. Еще более опасной является троянская или backdoor программа, содержащая клавиатурный шпион – его наличие существенно расширяет функции троянской программы и ее опасность для пользователя.