

Харлэн Карви

**Криминалистическое исследование
Windows**



Проведение анализа при ограниченном бюджете

Содержание этой главы:

§ Документирование анализа

§ Инструменты

ü Краткое изложение

ü Быстрое повторение

ü Часто задаваемые вопросы

Введение

Похоже, что расследование инцидента или проведение компьютерно-технической экспертизы недоступно для некоторых специалистов из-за стоимости различных коммерческих инструментов. Однако это касается не только людей, увлеченных своим хобби, и тех, кто заинтересован в тщательном исследовании этой увлекательной (во всяком случае, для меня) области. Это касается учебных заведений: курсы по судебной компьютерной экспертизе предлагаются не только в ведущих университетах, но также и в общественных колледжах. Стоимость коммерческих инструментов касается сотрудников правоохранительных органов и даже консультантов (таких как я, например – ведь у нас у всех есть бюджет, которого нужно придерживаться). Было бы неплохо иметь доступ ко всем выпускаемым инструментам, не так ли? Безусловно, но с точки зрения бюджета это непрактично.

Кроме того, это не особенно нужно. Коммерческие приложения (EnCase, FTK, ProDiscover и т. д.) – это всего лишь инструменты. Каждый инструмент имеет свои достоинства и недостатки, а подготовленные, компетентные эксперты, прежде чем выбирать инструменты или приложения, которые помогут им во время анализа, понимают, какие задачи им предстоит выполнить. Ведь, как я уже неоднократно говорил, прошло то время, когда для проведения судебного анализа эксперты просто загружали образ клонированных данных в специализированную программу и нажимали кнопку в ее интерфейсе. Самое главное в судебном анализе – понимать, какие артефакты вам доступны, и иметь логический, обоснованный и подробный план или процесс для сбора и интерпретирования данных. С этой точки зрения вы не привязаны к отдельному коммерческому приложению (при отсутствии определенных требований, вынуждающих вас его использовать) и можете взамен попробовать недорогие или бесплатные инструменты или приложения, необходимые для вашего анализа.

Совет
Сейчас подходящее время рассмотреть тему антикриминалистических инструментов. Антикриминалистические инструменты – это инструменты (а в некоторых случаях приемы), используемые злоумышленником, чтобы осложнить работу экспертов, например, изменяя отметки времени МАС (время изменения, доступа и создания) или стирая данные (или «доказательства») с накопителя. Многие считают, что целью антикриминалистических средств являются определенные коммерческие приложения, но это не так. Конечно, на популярных конференциях по компьютерной безопасности были представлены доклады, в которых говорилось о том, как помешать работе эксперта, использующего программу EnCase, но на самом деле антикриминалистические инструменты и приемы нацелены на эксперта, а не на определенные приложения. Эксперт, который понимает это, находится на шаг впереди, а не позади, злоумышленника.

В каждой из глав этой книги рассматривались, описывались и/или демонстрировались инструменты, используемые для определенных целей, но в каждом случае были представлены только основные возможности и характеристики отдельного инструмента. Задача этой главы – познакомить читателей с некоторыми другими инструментами, такими как шестнадцатеричные редакторы, средства для перехвата и анализа пакетов и т. д., которые помогут им приступить к работе. Существует множество инструментов, которые можно использовать, и многие из них даже не были разработаны для анализа. Однако кто-то узнал, что эти инструменты полезны благодаря некоторым из предлагаемых ими функциональных возможностей. Не следует рассматривать эту главу как полное и исчерпывающее руководство по всем инструментам, которые вам могут потребоваться. В лучшем случае эта глава служит для того, чтобы показать вам, что есть другие варианты, помимо тех, которые недоступны из-за стоимости программного продукта или стоимости обучения, связанного с этим продуктом.

Наконец, если вы знакомы со мной или читали любую из моих книг, то знаете, что я фанат языка программирования Perl. Некоторые могут даже сказать, что Perl – это мой молоток, а все, что я вижу, – это гвоздь. И они будут правы. А если серьезно, то Perl может быть чрезвычайно мощным инструментом, например, когда вам нужно проанализировать несколько сотен мегабайт журналов веб-сервера на наличие признаков атаки с внедрением SQL-кода и преобразовать шестнадцатеричное кодирование или кодирование символов, чтобы получить фактическую внедренную команду и найти другие зараженные системы. То, на что у вас могло бы потребоваться несколько дней, теперь можно сделать за несколько минут. Я убедился в этом сам и демонстрировал это другим с помощью написанных мной скриптов, большая часть которых легла в основу приложения RegRipper (www.regripper.net). Я слышал, как другие говорили то же самое – как, благодаря умению коллеги работать с Perl, удавалось сократить несколько дней обработки данных вручную до пары часов, используя Perl-скрипт. Это не значит, что Perl – *единственный* доступный язык программирования, потому что подойдет любой язык, с которым вам удобно работать, например Python. Благодаря Дейву Роту (Dave Roth) я предпочитаю Perl.

Документирование анализа

Я знаю, что мы уже говорили о документировании в других главах этой книги, а сейчас мы снова рассматриваем эту тему. Это связано с тем, что тема документирования чрезвычайно важна, особенно потому, что технические специалисты не очень любят этим заниматься. Что касается меня, то я никогда не любил ничего документировать, пока не понял, что случается, когда сведения об анализе не документируются. Например, мне приходила в голову какая-нибудь замечательная мысль, или я находил отличный инструмент или прием для анализа, а три месяца спустя я не мог вспомнить, что это было. И я это не задокументировал! Тема документирования повторяется в этой книге просто потому, что она так важна.

Еще одна важная тема в этой книге – необходимость повторяемости в работе (будь то сбор данных или анализ), которая достигается посредством документирования. Повторяемость, которая по существу представляет собой возможность взять те же данные, следовать тому же процессу, используя те же инструменты, и получить те же результаты, – это основополагающий принцип судебной науки. Одна из причин, по которой документирование должно вести к повторяемости, состоит в том, что эксперты не всегда находятся на месте. Один специалист или эксперт может выполнить работу, а затем, когда несколько месяцев спустя этот эксперт будет в отпуске, или ему дадут другое задание, у кого-нибудь могут возникнуть вопросы относительно результатов. Другой эксперт должен суметь включиться в работу и, используя исходные данные и записи первого эксперта, повторить тот же процесс и (будем надеяться) получить те же результаты. То же касается работы, к которой эксперту, возможно, придется обратиться

год спустя; без правильно составленной документации эксперт вряд ли сможет вспомнить, что он сделал.

При проведении любого судебного анализа в первую очередь необходимо иметь способ для документирования ваших действий. Ведь если вы выполняете анализ, но не документируете его, то *его не происходит*. Несмотря на то, что технари, похоже, не любят этого делать, документирование анализа имеет первостепенную важность для ваших действий. Документацию нужно (а не *следует*) вести достаточно подробно и понятно, чтобы позволить другому специалисту или эксперту понять, что вы сделали, и подтвердить ваши действия. Кроме того, документация должна быть достаточно подробной и понятной, чтобы вы смогли найти ваши записи об анализе годичной давности (или более старые) и подтвердить свои действия. Под словом *подтвердить* я имею в виду, что, используя те же данные и те же инструменты (вы ведь указали в своих записях применяемые инструменты и их версии, не так ли?), вы или другие специалисты должны получить те же результаты.

Задумайтесь об этом. Предположим, вы провели анализ и, после завершения экспертизы и составления итогового отчета, положили накопитель в свой сейф, ожидая окончательного решения по делу. Затем, месяц спустя, возникает вопрос относительно какой-нибудь информации в вашем отчете, и вам нужно вернуться к этим данным и подтвердить какой-нибудь аспект вашего анализа. Но это происходит через месяц (полгода или год), а ваша скорость работы такова, что за это время вы не только провели несколько экспертиз, но вам также дали другое задание, и исходные данные нужно передать другому эксперту. Как стыдно и неприятно вам будет, если другой эксперт не сможет, используя те же данные и те же инструменты, получить ваши результаты? Как вы это объясните? Большинство из нас, вероятно, сказал бы что-то вроде «Вы неправильно это сделали» или «Вы использовали не ту версию инструмента», не так ли? Ну, а как бы это выглядело, если бы *вы* не смогли повторить свои собственные результаты?

Документировать свои действия – важно, но еще важнее – документировать свои действия так, чтобы кто-то другой мог получить ваши результаты. Итак, как это сделать? Я всегда считал, что лучший подход – быть кратким. Я видел много специалистов, которые были слишком многословными в своих записях, и их фактические действия терялись в прозе. Предположим, вы подозреваете, что веб-сервер, возможно, подвергся атаке с внедрением SQL-кода. Наиболее логично было бы искать признаки такой атаки в журналах веб-сервера. Если веб-сервер – это IIS (Internet Information Server) от Microsoft, а серверная база данных – это Microsoft SQL Server, то логично было бы начать с поиска применения расширенной хранимой процедуры *xp_cmdshell* в журналах веб-сервера, потому что обычно вы не увидите этого в журналах веб-сервера. Итак, допустим, вы создали новый проект ProDiscover, добавили в него образ веб-сервера, а затем выполнили поиск строки «xp_cmdshell» в журналах веб-сервера. Ваши записи могут выглядеть следующим образом:

- § Создал проект с именем «intrusion_20081030» в программе ProDiscover 5.0. Добавил образ веб-сервера, сохранил проект.
- § Выполнил поиск строки «xp_cmdshell» в журналах веб-сервера (укажите полный путь), используя функцию поиска программы ProDiscover; обнаружил несколько совпадений поиска в журналах «ex081002.log» и «ex081003.log».

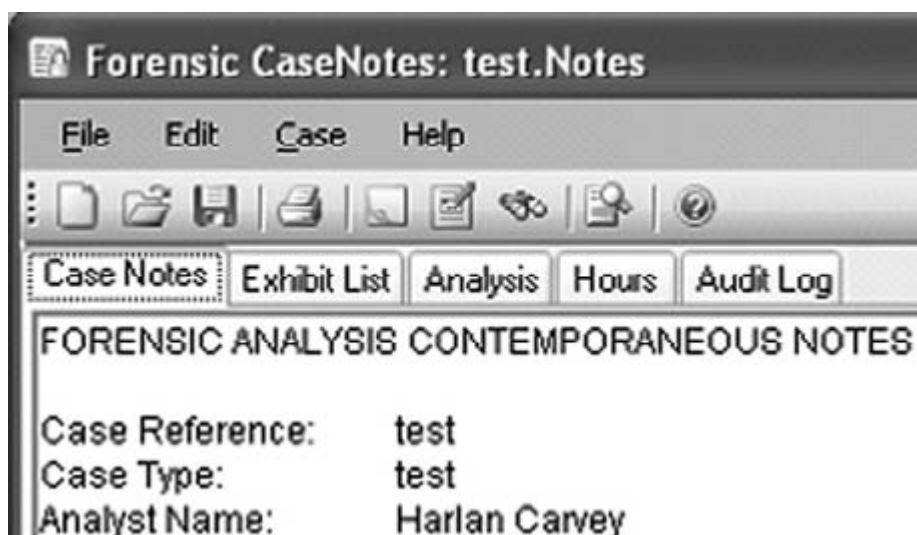
Вот так. Просто, кратко, по существу, но при этом понятно и подробно. В данном случае вы указали, что искали (xp_cmdshell), что использовали для выполнения поиска (функцию поиска программы ProDiscover 5.0), и что обнаружили (совпадения в двух файлах журналов). Например, запись «проанализировал файлы журналов» или «выполнил поиск в файлах журналов» ничего не говорит о том, что вы сделали. В каких файлах журналов вы выполняли поиск? Что вы искали? Если вы выполняли поиск по ключевым

словам, то по каким именно? Каким образом вы выполняли поиск? Используя `grep` или функцию «Поиск» (“Search”) в Windows (например, экспортировали файлы журналов из образа при помощи FTK Imager, а затем нажали кнопку «Пуск» (“Start”), выбрали функцию «Поиск» (“Search”) и указали параметр «Файлы и папки» (“For Files and Folders”))? Какими были результаты? Видите, как неполная документация создает вопросы, а не отвечает на них? Кроме того, не будьте излишне многословными; как я уже говорил, мне встречались записи настолько пространные, что сведения о фактической проделанной работе и результатах было абсолютно невозможно найти. Добавление дат на каждой странице или к каждому событию делает ваши записи более достоверными, а если над одним крупным проектом работает несколько членов группы, то инициалы или подписи сотрудников после сведений о выполненных ими заданиях могут быть настоящим преимуществом в дальнейшем.

Добавление информации об используемых инструментах и их версиях позволит вам легче восстановить и подтвердить результаты. Версия инструмента имеет большое значение, особенно если версии инструмента отличаются друг от друга наличием критических обновлений. Это в особенности важно, если вы используете антивирусное приложение для проверки монтированного образа или только нескольких файлов экспортированных из образа. Информация о версии модуля проверки, а также о версии антивирусной базы, может иметь огромное значение, особенно когда через две недели выяснится, что файлы, которые вы проверяли, являются вредоносными.

К числу важных аспектов документирования анализа также относится обоснование ваших умозаключений. Эксперты очень редко дают выводы относительно чего-либо, основываясь только на одном виде данных; в большинстве случаев для этого сопоставляется несколько дополнительных элементов данных. Например, если бы мне нужно было определить, когда пользователь входил в систему Windows, то в первую очередь я бы начал поиск в файле куста «Security». Анализ одного раздела в том файле куста покажет мне, включен ли аудит событий входа в систему. Если включен, то я буду искать соответствующие записи о событиях в журнале событий безопасности. Я бы также проверил файл куста «SAM» на наличие времени последнего входа пользователя в систему, а также файл куста «NTUSER.DAT» в каталоге профиля пользователя на наличие признаков активности в исследуемый период времени. Все эти сведения можно использовать для того, чтобы определить, когда пользователь входил в систему. Кроме того, ссылки на внешние источники, например, на статьи из базы знаний Microsoft, – отличный способ обосновать результаты анализа.

Один из инструментов, который, по моему мнению, очень удобен для ведения документации, – это Forensic CaseNotes от британской компании QCC Information Security (www.qccis.com/?section=casenotes). Forensic CaseNotes – отличный инструмент для ведения записей об экспертизе. Он бесплатный, настраиваемый и довольно универсальный. Как правило, загрузив программу CaseNotes на компьютер и установив ее, я сначала создаю настраиваемые вкладки, отвечающие моим потребностям, как показано на илл. 9.1.



Илл. 9.1. Фрагмент графического интерфейса программы Forensic CaseNotes после выполненных мной настроек.

На илл. 9.1 показано несколько вкладок. Одну вкладку я создаю для информации об исследуемых объектах (“Exhibit List”). В ней хранятся записи обо всех имеющихся у меня носителях, а также примечания, касающиеся клонирования данных. Информация для этой вкладки берется из моих рабочих листов о клонировании данных. Также я создаю вкладку «Часы» (“Hours”). Я использую ее для учета времени, потраченного на работу, которая непосредственно относится к делу; эта информация включается в счет клиенту. Это чрезвычайно важно для консультантов. Возможно, самая важная вкладка – это «Анализ» (“Analysis”). Здесь я записываю информацию о фактической работе, которую выполняю ежедневно. Иногда я работаю над анализом сам, иногда – как руководитель группы и также управляю работой других членов группы или работой другой организации из компании клиента. Когда вся информация доступна в одном месте, я могу увидеть, что уже сделано, и нужно ли сделать запрос на дополнительные рабочие часы.

Одно из преимуществ программы CaseNotes состоит в том, что вы можете добавлять изображения (снимки экрана, цифровые фотографии и т. д.) в текстовые поля под каждой вкладкой. При проведении анализа я часто копирую и вставляю в содержимое вкладок команды, используемые в различных инструментах командной строки, фрагменты выходных данных таких инструментов, а также какую-нибудь особенную информацию, например, схемы расположения выводов для адаптеров. Вся эта информация может быть использована в будущем, например, другим экспертом, или быть очень полезной при составлении отчета. Часто добавление фотографии или схемы в отчет может избавить вас от множества объяснений и недоразумений.

Что еще я указываю в своих записях? Все. Серьезно. Сюда относятся ссылки на информацию из Интернета, которую я использовал как часть своего анализа, например, ссылки на статьи из базы знаний Microsoft и даже на хакерские сайты, если они имеют отношение к экспертизе (и подтверждены дополнительными данными); снимки экрана, изображения всего, что относится к экспертизе, и т. д.; версии используемых инструментов, а также ссылки на отдельные инструменты или скрипты, созданные мной, чтобы облегчить управление имеющимися данными, и информация об этих инструментах и скриптах. В некоторых случаях, если используется несколько инструментов, я архивирую их в другом месте для последующего использования.

Следует отметить, что имели место случаи, когда эксперты не могли получить доступ к файлу CaseNotes после того, как защитили его паролем и закрыли его. Также имейте в виду, что программа CaseNotes сохраняет содержимое ваших записей в файле в собственном формате, то есть, если вы используете CaseNotes, не следует ожидать, что вы сможете открыть файл с записями в программе «Блокнот» и найти там доступную для чтения информацию.

Еще один доступный инструмент – диспетчер записей NoteCase (<http://notecase.sourceforge.net>). Я не пробовал использовать NoteCase или, собственно говоря, другие инструменты для ведения записей о деле. Однако если вы ищете что-то, что позволит вашим записям быть более доступными, то лучший способ для этого – задать формат или создать контрольный список в программе Microsoft Word. Большинство коммерческих организаций использует программу Microsoft Word; вы можете сохранить файлы в нескольких форматах, а различные платные (Adobe) и бесплатные (PDFCreator) инструменты позволят вам преобразовать эти файлы в формат PDF. Вкладки, которые я использовал в CaseNotes, можно легко переделать в виде колонтитулов в документе Microsoft Word Excel, и вы даже можете использовать таблицы или встроенные электронные таблицы Excel для учета рабочих часов.

Общая идея в том, что у вас есть *несколько* способов для документирования своей работы последовательным и доступным для проверки образом. Я не упомянул о еще одной очень хорошей причине вести записи. Что произойдет, если вас вызовут в суд давать показания о какой-либо вашей работе. Вы вспомните все подробности и нюансы экспертизы или расследования шестимесячной или годичной давности? В моей практике было несколько случаев, когда мне задавали вопрос (не в суде и не во время дачи письменных показаний под присягой) о работе, которую я выполнял некоторое время (четыре, шесть или девять месяцев) тому назад, и мне приходилось обращаться к своим записям, чтобы убедиться, что я понял, о какой экспертизе идет речь.

Инструменты

Проведение расследования инцидента или судебного анализа сопряжено с большим числом различных действий. В связи с этим вам нужен подходящий инструмент для соответствующей задачи – но какой из инструментов лучше всего подходит вам? Цель этого раздела – показать вам ряд инструментов, полезных для различных задач, чтобы вы могли начать свой путь по дороге открытий.

Инструменты, перечисленные в этой главе, находятся в открытом доступе, и большинство из них можно бесплатно использовать в рамках лицензионного соглашения. Некоторые инструменты представляют собой ознакомительные версии, и если вы захотите продолжать пользоваться ими, вам, возможно, придется приобрести их полные версии.

Создание образов данных

Клонирование данных – это то, чем в основном занимаются эксперты и специалисты по расследованию инцидентов, а создание образов из систем – это лишь часть этой работы.

dd

dd – это утилита, о которой думает большинство экспертов, когда дело касается создания образа данных. dd – собственная команда Linux/UNIX, которая (согласно справочным руководствам, например <http://linuxreviews.org/man/dd>) «преобразовывает или копирует файл» и очень часто считается стандартной утилитой для этой цели. Существует несколько версий этой утилиты, и некоторые из них обладают другими возможностями; однако все они по существу выполняют одну и ту же основную функцию – позволяют создавать образы накопителей или томов.

Одна из версий dd, разработанная Джорджем Гарнером-младшим (George M. Garner, Jr.), предназначена для работы в ОС Windows и является частью пакета Forensic Acquisition Utilities (<http://gmgsystemsinc.com/fau>). Этот пакет утилит позволяет создавать образы систем, передавать их по сети (если у вас нет доступного локального накопителя), использовать сжатие, а также вычислять и проверять хэш-значения, чтобы обеспечить целостность клонированных данных.

Инструменты и ловушки...

Применение dd для создания образа накопителя работающего компьютера

Большинство специалистов считает, что такие инструменты, как dd, предназначены только для клонирования данных накопителей, извлеченных из компьютеров – подключите накопитель к устройству блокирования записи и используйте dd для создания образа. Конечно, это предпочтительный метод, но иногда его невозможно осуществить. В одном из случаев из-за характера сетевой инфраструктуры клиента и воздействия, которое оказало бы разборка компьютера и отключение его от сети, чтобы клонировать данные накопителя, мы решили использовать стандартную команду dd (SUSE Linux 9), чтобы создать образ физического НЖМД работающего компьютера. Мы тщательно задокументировали причину и процесс для этого подхода, в том числе информацию о версиях используемых утилит (dd, split и т. д.) в наших записях о деле и в отчете.

dcfldd (<http://dcfldd.sourceforge.net>) – еще одна свободно доступная версия инструмента dd, которая также работает в ОС Windows. Утилита dcfldd была написана Ником Харбором (Nick Harbour). На странице веб-сайта Sourceforge, посвященной dcfldd, эта утилита описывается как «расширенная версия GNU dd» с такими дополнительными возможностями, как проверка создания образа или стирания данных, хэширование, отправка данных журнала в другие команды или в файл и т. д. Все эти функции чрезвычайно полезны не только для того, чтобы обеспечить целостность образа и позволить вам создать образ по сети (при клонировании данных работающего компьютера вы не захотите записывать файл-образ на фактический НЖМД, который вы клонируете), но и для того, чтобы удалить или стереть образ после завершения анализа.

Инструменты и ловушки...

Образы формата dd

Все чаще и чаще эксперты понимают, что необходима определенная стандартизация во всех аспектах их работы. Это также относится к созданию образов. В арсенале специалиста должно быть несколько способов для создания образов, например, с помощью аппаратных блокираторов записи (которые применяются для создания образа напрямую с накопителя на накопитель, а также для создания образа посредством программного обеспечения, и т. д.), и средств (инструменты и приемы) для клонирования данных работающего компьютера. Кроме того, группам по расследованию инцидентов следует предусматривать в своих стандартных процедурах типовой формат, в котором должны (если это возможно) создаваться образы.

Почему это так важно? До того как недавно были выпущены обновленные версии приложений судебного анализа, мне представилась возможность помочь в проведении экспертизы, во время которой образ одного накопителя был создан в формате dd, а образ другого накопителя из того же компьютера – в формате, характерном для одного приложения судебного анализа. В той ситуации я мог бы ограничиться использованием одного отдельного приложения для судебного анализа.

Использование унифицированного формата для создания образов также важно по ряду других причин. Во-первых, это повышает ваш уровень профессионализма в глазах ваших клиентов, а также коллег. Поверьте, когда я в первый раз начал изучать подробности экспертизы, в проведении которой я собирался помочь, и увидел два образа НЖМД из одного компьютера, созданных в разных форматах, моя первая мысль была: «У этих ребят вообще был план работы?».

Кроме того, даже не смейте думать, что вы будете единственными, кто увидит эти образы. В моей практике было несколько случаев, когда, после того как я завершил экспертизу и предоставил отчет, клиент просил передать ему образы, а не просто очистить

накопители и отправить их назад. Всегда будьте готовы вернуть образы клиенту или передать их кому-нибудь другому для анализа; наличие унифицированных форматов образов (а также документации) просто является признаком профессионализма.

Во-вторых, необходимость в унифицированном формате образа естественно ведет к документированию, которое решает вопросы, связанные не только с приемами, используемыми для создания образа, но и с обоснованием причин, по которым вам нужно было отойти от стандартных рабочих процедур. В целом это свидетельствует о более профессиональном и тщательном подходе к работе.

FTK Imager

Как FTK Imager, так и FTK Imager Lite, можно бесплатно загрузить с сайта AccessData.com (www.accessdata.com). FTK Imager Lite – это «облегченная» версия инструмента FTK Imager, которую для использования можно распаковать и записать на компакт-диск или скопировать на флеш-накопитель. На веб-сайте AccessData.com есть статья с перечнем файлов, которые вам потребуются из архива FTK Imager, если вы захотите запускать этот инструмент с компакт-диска или флеш-накопителя.

Я считаю, что программа FTK Imager может быть чрезвычайно полезной в нескольких случаях. Когда мне нужно было провести экспертизу образов, созданных при помощи EnCase, а программа EnCase была недоступна (или я просто не хотел ее использовать), я открывал файлы формата .E0x в FTK Imager и либо извлекал отдельные файлы, либо создавал образ в формате dd. Кроме того, я использовал FTK Imager для проверки файловых систем созданных образов, в том числе образа ОС SUSE Linux 9 с файловой системой ReiserFS. Конечно, я также использовал программу FTK Imager для клонирования данных, либо запуская ее вместе с правильно подключенным устройством блокирования записи, либо запуская ее с компакт-диска и создавая образ накопителя из компьютера с ОС Windows на НЖМД, подключенный по USB (или сохраняя образ на других носителях/в других местах).

FTK Imager можно также использовать для открытия файлов .vmdk виртуальной машины VMware. В моей практике были случаи, когда системы, работающие в виртуальной среде VMware, были частью сетевой инфраструктуры или даже систем, данные которых нужно было клонировать и анализировать. Поэтому, возможно, самый легкий способ «клонировать» такие системы – просто копировать файлы .vmdk (и .vmem, при наличии таковых) из хостовой системы. В FTK Imager можно выбрать пункт «Добавить исследуемый объект» (“Add an Evidence Item”), чтобы просмотреть файловую систему и извлечь отдельные файлы, или выбрать пункт «Создать образ накопителя» (“Create Disk Image”), чтобы преобразовать файл .vmdk (или образ .E0x) в формат dd, SMART или E0x. Эти функции очень полезны, когда коммерческие инструменты могут не распознавать формат .vmdk или могут быть сложнее, чем необходимо, для работы, которую вам нужно выполнить.

Если вы не хотите создавать собственные образы данных, или у вас нет для этого необходимых средств, существуют веб-сайты, откуда можно загрузить образы, предлагаемые для тестирования инструментов или как часть задачи. Это отличная возможность, предоставляемая некоторыми очень толковыми пользователями. В конце концов, разве может быть лучший способ сообщить о каком-нибудь направлении, методе или приеме анализа, чем описать его, а затем предложить пользователям некоторое средство, чтобы попробовать практический подход к обучению? Большинство образов публикуется вместе с определенными задачами или вопросами, которые должны помочь специалисту определить направление анализа. По своему опыту я хорошо знаю, что такие инструкции, как «найти все подозрительные или вредоносные действия» могут привести только к множеству оплачиваемых часов, которые нельзя вернуть. Публикуемые задачи являются не только отличным средством для оттачивания навыков анализа, но и замечательным примером того, какой должна быть экспертиза с самого начала.

Одним из первых найденных мной ресурсов со свободно доступными образами был проект CFReDS на сайте Национального института стандартов и технологий (NIST). Файлы-образы для задачи о взломе «Hacking Case» (www.cfreds.nist.gov/Hacking_Case.html) включали в себя не только образ формата dd, разделенный на части, но и образы EnCase или EWF (формат Expert Witness; программа Expert Witness была предшественницей EnCase) для тех, кто хочет попрактиковаться с другими инструментами.

Еще один сайт, содержащий несколько образов данных и сценариев тестирования, – это Digital Forensics Tool Testing (<http://dfft.sourceforge.net>), основанный доктором Брайаном Кэрриэром (Brian Carrier). На сайте предоставляется несколько специальных образов для тестирования инструментов судебного анализа, но, как и в случае с другими сайтами, эти образы можно использовать в качестве основы для развития и совершенствования навыков анализа, а также для ознакомления с различными приложениями судебного анализа.

Ланс Мюллер (Lance Mueller) предлагает интересные практические задачи в своем блоге ForensicKB.com (www.forensickb.com/search?q=practical). Ланс любезно предоставляет практические сценарии вместе с небольшими образами (~400 Мб), созданными из ОС Windows XP, в сжатом формате .E0x/EWF. Если у вас нет лицензионной EnCase, не волнуйтесь: FTK Imager легко откроет эти образы и позволит вам экспортировать из них файлы или просто создать образ в формате dd из файла в EWF-формате. Некоторые комментарии к этим статьям в блоге Ланса Мюллера также дадут вам представление о том, что искали и что нашли другие эксперты.

Анализ образа

После того как вы создали образ, проверили хэш-значения образа и накопителя и задокументировали весь процесс, вам потребуются средства для открытия образа и выполнения основных функций анализа, являющихся необходимой частью вашей работы. В этой книге мы рассматривали несколько инструментов, предназначенных для этих целей, в том числе для открытия целого файла-образа, просмотра структуры файловой системы в целом, выполнения поиска и т. д. или просто для извлечения отдельных файлов (файлов кустов реестра, файлов журналов событий и т. д.) для анализа.

The SleuthKit

Набор инструментов The Sleuth Kit (TSK; www.sleuthkit.org) был разработан доктором Брайаном Кэрриэром и предоставляет внутренние компоненты для средства Autopsy Forensic Browser. TSK – набор инструментов командной строки, позволяющих исследовать и анализировать файловые системы в файлах-образах. Инструменты TSK также доступны для операционных систем Windows; однако на момент написания этой книги средство Autopsy Forensic Browser не было адаптировано для собственного формата Windows (хотя все инструменты можно компилировать для Windows, используя подсистему Cygwin).

Инструменты TSK можно использовать в системах Windows почти так же, как в Linux, но с некоторыми оговорками. Во-первых, по словам доктора Брайана Кэрриэра, существует проблема с символами подстановки в командной строке, и вам придется указывать все части разделенного образа по порядку. Это означает, что если вы анализируете файл-образ, разделенный на несколько частей, вам нужно будет указать каждую из них, как показано ниже:

команда [параметры] образ1 образ2 образ3...

Здесь пригодится программа FTK Imager, которая может повторно собрать части образа в единый файл-образ. FTK Imager без проблем повторно соберет части образов других форматов, например, образы своего собственного формата или образы, созданные такими программами, как EnCase от компании Guidance Software. Можно также использовать команду *type* в Windows, чтобы повторно собрать части разделенного образа, созданного в формате необработанных данных:

```
D:\images>type image.001 > image_all.img
D:\images>type image.002 >> image_all.img
D:\images>type image.003 >> image_all.img
...
```

TSK может открывать образы таких форматов, как dd, EWF (т. е. программы Expert Witness/EnCase) и AFF (www.sleuthkit.org/sleuthkit/desc.php). Утилита «fls.exe» (версии 3 из набора TSK) для платформы Windows может анализировать образы необработанных данных (dd), образы EWF и образы необработанных данных, разделенные на части:

```
D:\tools\tsk>fls -i list
Supported image format types:
  raw (Single raw file (dd))
  ewf (Expert Witness format (encase))
  split (Split raw files)
```

На веб-сайте SleuthKit и других онлайн-ресурсах доступно много документов, в которых описывается, как совместно использовать различные инструменты командной строки для проведения анализа образа. Например, статьи о способах анализа файловой системы (http://wiki.sleuthkit.org/index.php?title=FS_Analysis) и о создании временной шкалы активности в системе (<http://wiki.sleuthkit.org/index.php?title=Timeline>) предоставляют множество чрезвычайно полезной информации об инструментах TSK. Возможно, лучший источник информации об этих инструментах – вики-сайт TSK (http://wiki.sleuthkit.org/index.php?title=Main_Page).

К простым примерам использования инструментов TSK относится применение утилиты *dls* для сбора данных из свободного пространства из файла-образа. Следующую команду можно использовать для извлечения данных из свободного пространства из созданного ранее файла-образа:

```
dls -A image.dd > unalloc.dls
```

Извлечение данных из свободного пространства из файла-образа может быть полезным при восстановлении файлов, а также при выполнении поиска строк или поиска с помощью утилиты *grep* по этому свободному пространству, например, при поиске номеров кредитных карт, IP-адресов или адресов электронной почты.

Следующая команда предоставит вам информацию о файловой системе в файле-образе:

```
fsstat -f ntfs image.dd
```

Команда *fsstat* показывает информацию о файловой системе, метаданных и содержимом из файла-образа. Например, применение этой команды к образу данных ОС Windows XP возвращает следующую информацию о файловой системе:

```
FILE SYSTEM INFORMATION
-----
File System Type: NTFS
Volume Serial Number: 98B0A679B0A65D8E
```

OEM Name: NTFS
Version: Windows XP

Можно получить большую часть той же информации на работающем компьютере, используя утилиту «fsutil.exe». Например, следующая команда возвращает информацию (в том числе серийный номер тома), похожую на результаты утилиты «fsstat.exe», но на работающем компьютере:

```
C:\>fsutil fsinfo volumeinfo C:\
C:\>fsutil fsinfo ntfsinfo C:
```

Серийный номер тома создается во время форматирования накопителя и вычисляется из этого времени; его можно частично применять для обозначения образа данных при использовании совместно с другой документацией.

Возможно, самый полезный для эксперта инструмент из набора TSK – это «fls.exe» (<http://wiki.sleuthkit.org/index.php?title=Fls>), который перечисляет имена файлов и каталогов в файловой системе и разделяет их вертикальной чертой, что позволяет использовать эти данные для создания временной шкалы при помощи скрипта «mactime.pl». Например, следующая команда применяется ко всему файлу-образу, рекурсивно выполняясь в каталогах и подкаталогах:

```
D:\tools\tsk>fls -m c: -r d:\cases\xp\xp.001
```

Параметр *-m* позволяет перед перечнями файлов и каталогов добавлять имя используемой точки монтирования (в данном случае C:\). Чаще всего выходные данные команды перенаправляются в выходной файл, а затем обрабатываются таким инструментом, как «mactime.pl» или ex-tip (средство создания временной шкалы, разработанное Майклом Клоппертом (Michael Cloppert); https://www2.sans.org/reading_room/whitepapers/forensics/32767.php), чтобы упорядочить информацию о файловой системе в удобочитаемый и понятный формат.

Инструменты и ловушки...

Временные шкалы

Такие инструменты, как «fls.exe» из набора TSK, «mactime.pl» и ex-tip, разработанный Майклом Клоппертом, предоставляют полезные функциональные возможности с открытым исходным кодом для создания временных шкал активности в файловой системе. Вместе с тем, так как это инструменты с открытым исходным кодом, их функциональные возможности легко расширить. Например, во временную шкалу можно также включить любой другой источник информации, имеющий отметку времени, из ОС Windows, в том числе разделы реестра (а также параметры, данные которых имеют отметки времени), записи журналов событий и даже содержимое других файлов (инструмент ex-tip содержит фильтр для файлов журналов McAfee OnAccessScan, и можно написать фильтры для других журналов антивирусных приложений, для файла журнала «setupapi.log» и т. д.). Все, что для этого нужно, – придерживаться правильного стандарта (как показано на вики-странице TSK для утилиты «fls.exe»). Кроме того, данные можно вручную добавлять в промежуточный файл, прежде чем информация в нем будет отфильтрована и упорядочена с помощью такого инструмента, как например ex-tip, что позволяет вводить дополнительные данные, которые эксперт, возможно, захочет включить во временную шкалу. Приведение записей к общему формату позволяет импортировать их в другие инструменты, например, в Zeitline (<http://projects.cerias.purdue.edu/forensics/timeline.php>).

Информация о других командах для инструментов из набора TSK доступна в двухстраничном PDF-документе на сайте CyberGuardians (www.cyberguardians.org/docs/ForensicsSheet.pdf).

На веб-сайте Sourceforge также есть версия утилиты Selective File Dumper для Windows, которая называется FUNDL (сокращенно от File Undelete, рус. *средство восстановления файлов*) и использует инструменты из набора TSK (<http://sfdumper.sourceforge.net/fundl.htm>).

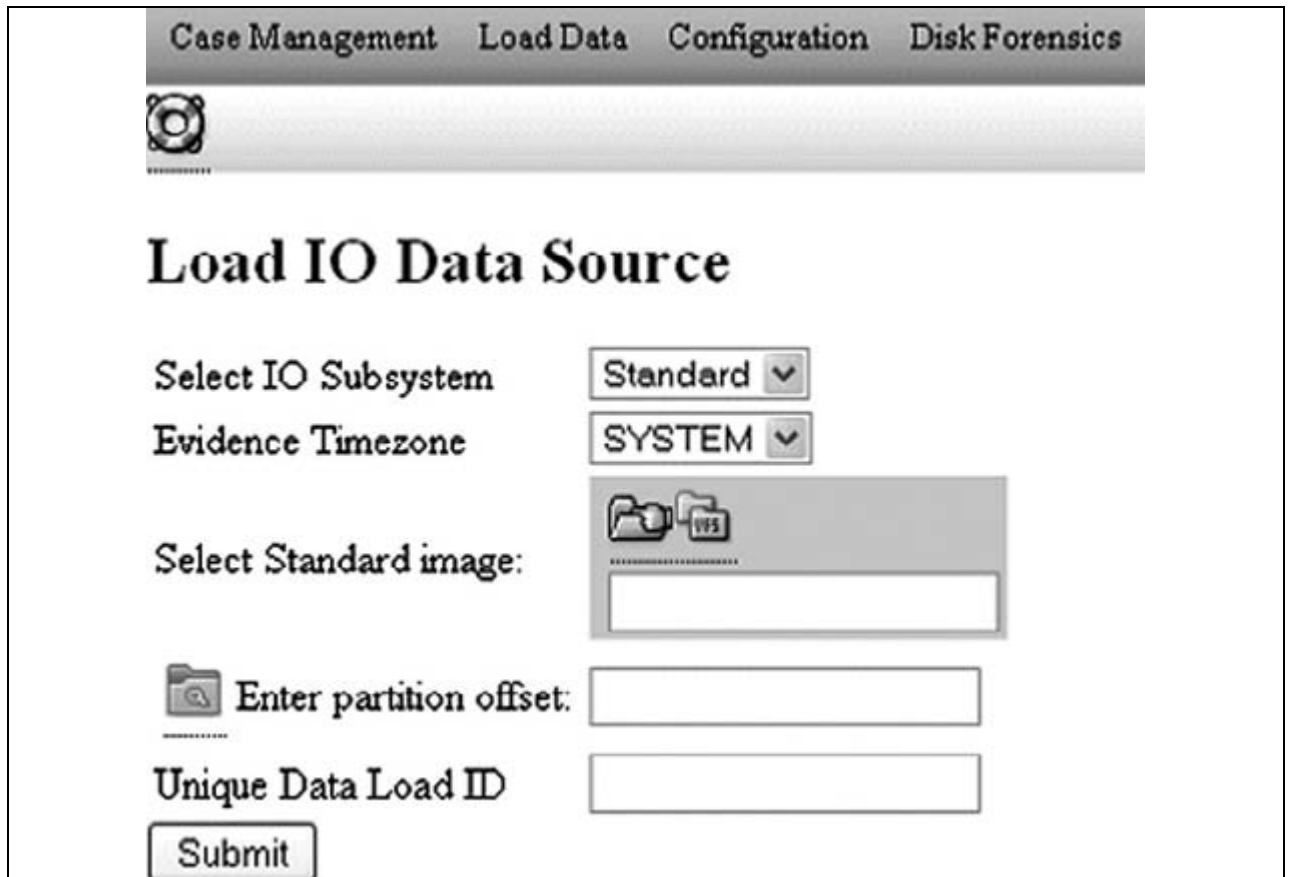
Инструменты и ловушки...

Форматы образов

Ранее в этой главе я говорил о необходимости стандартизации форматов образов. Цель процесса стандартизации – достичь последовательности и профессионализма. Некоторые организации могут использовать только одно коммерческое приложение для судебного анализа и иметь отличную причину принять в качестве стандарта формат образа, разработанный для этого приложения. Другие организации, например, консалтинговые фирмы, могут выбрать в качестве стандарта более доступный формат (т. е. dd), чтобы предоставить возможность доступа к большему количеству приложений для судебного анализа, что в свою очередь предусматривает возможность проверки данных и т. д. В 2008 году компания Technology Pathways выпустила программу ProDiscover версии 5.0, в которой появилась возможность открывать образы формата EWF. После конференции DFRWS 2008 доктор Майкл Коэн (Michael Cohen) разработал версию своего приложения PyFlag специально для работы в ОС Windows. В апреле 2008 года доктор Брайан Кэрриер выпустил версии инструментов Sleuthkit, которые были скомпилированы так, чтобы работать только под управлением ОС Windows. Эти инструменты (несмотря на то, что на момент написания этой книги с ними нельзя было работать без средства Autopsy Forensic Browser (нужно было использовать Cygwin-версии инструментов)) предоставляют через командную строку доступ к образам в формате dd, EWF (посредством libewf) и AFF (посредством afflib; www.afflib.org).

PyFlag

После конференции DFRWS 2008 доктор Майкл Коэн (Michael Cohen) выпустил разработанную специально для ОС Windows версию своего приложения PyFlag для судебного анализа и анализа журналов (www.pyflag.net/cgi-bin/moin.cgi/PyFlagWindows). Эта версия PyFlag известна под названием PyFlagWindows или WinPyFlag. Как бы вы не решили называть ее, не забудьте несколько раз великодушно поблагодарить доктора Коэна за его щедрый вклад в сообщество. Приложение PyFlag было некоторое время доступно для ОС Linux, а теперь широкие возможности PyFlag доступны тем экспертам, которым более комфортно работать в среде Windows. После загрузки и установки PyFlag для Windows согласно инструкциям на вики-сайте PyFlagWiki вам нужно только запустить файл «FlagHTTPServer.py», дважды щелкнув по нему левой кнопкой мыши, и ввести в веб-браузере адрес <http://127.0.0.1:8000>. На илл. 9.2 показана часть интерфейса PyFlag, запущенного посредством браузера Firefox в ОС Windows.



Илл. 9.2. Фрагмент интерфейса PyFlag в браузере Firefox в ОС Windows.

После установки PyFlag вы можете использовать его как обычно, так же, как если бы вы работали в Linux. Приложение PyFlag содержит инструменты TSK и позволяет эксперту объединять в одно «дело» файлы-образы, данные журналов и перехваченные пакеты. Доктор Коэн также включил в состав PyFlag инструменты платформы Volatility, что позволяет эксперту добавлять в дело дампы памяти.

Во время конкурса «Судебное родео» на конференции DFRWS 2008 (www.dfrws.org/2008/rodeo.shtml) доктор Коэн использовал PyFlag для проведения анализа, выполняя поиск в предоставленных данных (дамп памяти и образ, созданный с флеш-накопителя), чтобы ответить на вопросы, поставленные в задаче.

ProDiscover Basic

ProDiscover – отличное приложение для анализа данных, которое мне посчастливилось использовать, начиная с версии 3; пятая версия была выпущена летом 2008 года. Мне нравится использовать это приложение для анализа образов данных, клонированных из ОС Windows, потому что оно позволяет мне увидеть много информации в едином, интуитивно понятном, хотя и достаточно простом, интерфейсе. Независимо от того, выполняю ли я проверку файловой системы в образе, провожу ли быстрый или подробный анализ, я во многих случаях предпочитаю начать с ProDiscover.

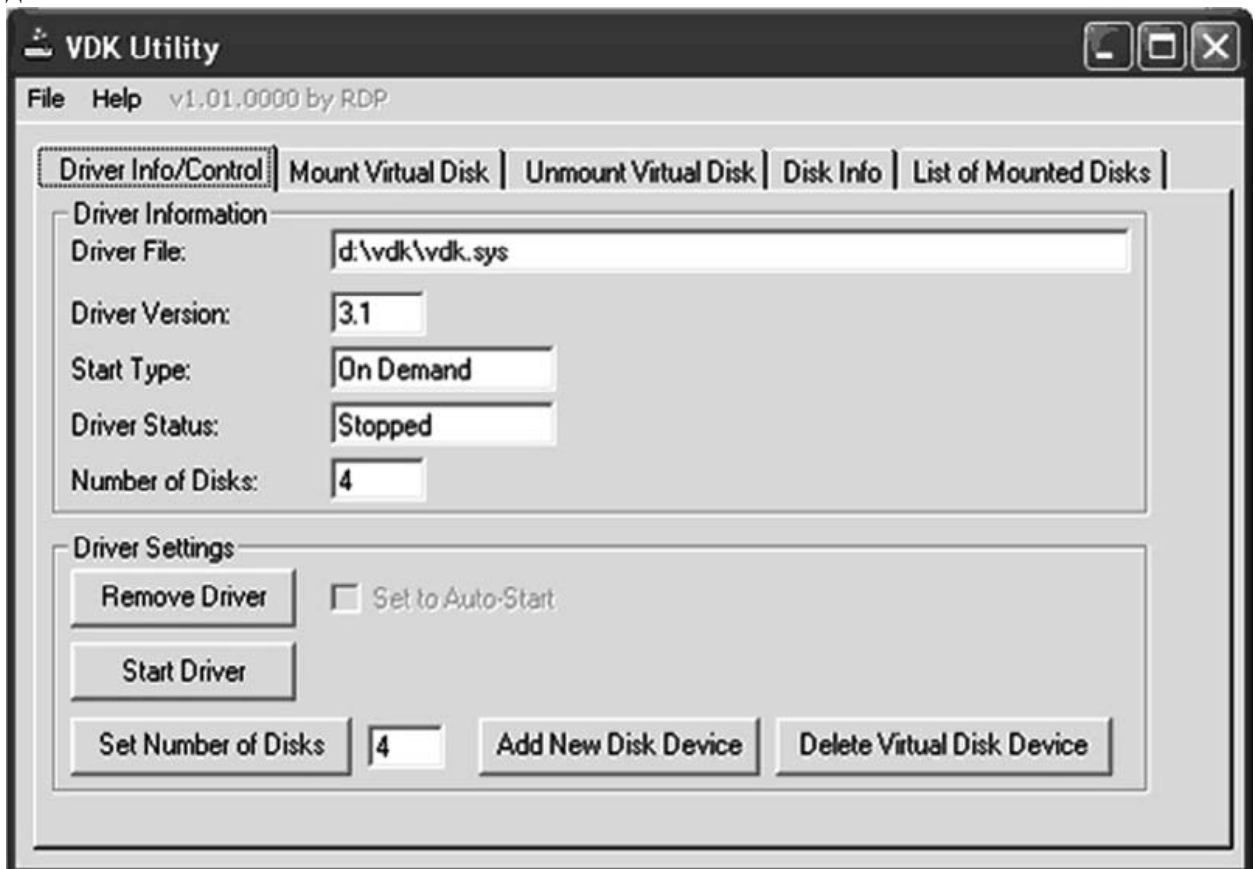
Крис Браун (Chris Brown), владелец компании Technology Pathways и автор книги *Computer Evidence: Collection and Preservation*, предоставляет возможность бесплатно загрузить и использовать базовую версию (Basic) приложения ProDiscover. Хотя версию Basic нельзя сравнить с возможностями полной версии приложения, она все равно является полезным инструментом.

Недостаток использования приложения ProDiscover состоит в том, как оно обрабатывает файлы-образы, разделенные на части. Образы, созданные в виде целых файлов, можно добавлять в файл проекта ProDiscover, но для того чтобы добавить образ, разделенный на несколько частей, нужно создать pds-файл. Файл формата .pds состоит из

сведений о заголовке и полного, упорядоченного списка всех частей разделенного образа. При добавлении образа в проект нужно выбрать pds-файл, а не первую часть разделенного образа (как вы бы сделали в программе FTK Imager, например).

Монтирование файла-образа

Помимо того, что файл-образ можно открывать в приложениях для анализа, его также можно монтировать как файловую систему в режиме только для чтения, чтобы образ отображался в вашем компьютере как накопитель. При соблюдении осторожности (используемое приложение должно быть настроено на монтирование файловой системы в режиме только для чтения) и обеспечении защиты файла-образа (то есть используйте копию данных, а не исходные данные, не забудьте установить права доступа к файловой системе, чтобы предотвратить запись в файл-образ, и т. д.) это может быть очень эффективным средством для различных видов анализа. Помимо программ, упомянутых ранее в этой книге (SmartMount от компании ASRData и Mount Image Pro от компании GetData), существует бесплатный инструмент Virtual Disk Driver (VDK; <http://chitchat.at.infoseek.co.jp/vmware/vdk.html>), позволяющий выполнять то же самое. VDK – это драйвер устройств, предоставляющий возможность монтировать на компьютере файл-образ как накопитель. При использовании этого инструмента вместе с графическим интерфейсом VDKWin (<http://petruska.stardock.net/Software/VMware.html>), показанном на илл. 9.3, вам нужно только нажать несколько кнопок, чтобы файловая система была смонтирована и доступна на вашем компьютере, используемом для анализа данных.

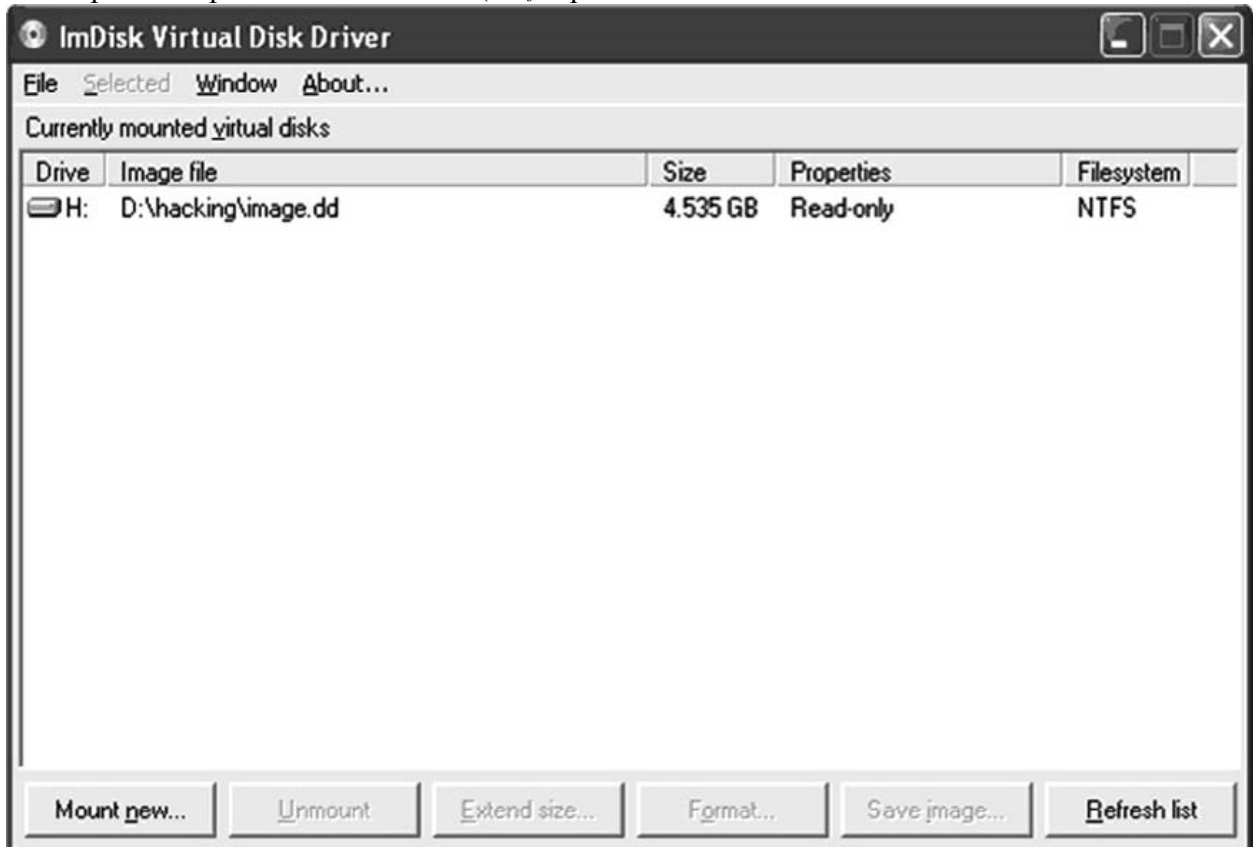


Илл. 9.3. Графический интерфейс VDKWin.

VDKWin устраняет некоторые сложности (и вероятность совершить ошибку) при работе с драйвером «vdk.sys», но насколько полезными являются инструменты такого типа? Я надеюсь, что к этому времени вы уже поняли, что эксперт не должен быть ограничен только одним способом выполнения работы; пока вы соблюдаете должный

уровень осторожности и документируете, что вы делаете (и почему), процесс, который вы используете для анализа файлов-образов, будет зависеть от ваших потребностей (или от стандартных рабочих процедур вашей организации, в зависимости от обстоятельств). Однако не исключено, что некоторые методы анализа будут недоступны из-за того, что они не предусмотрены в коммерческом приложении, которое вы используете, или, если они предусмотрены, цена, назначенная производителем за это приложение, выходит за пределы ваших финансовых возможностей.

Еще один бесплатный инструмент для монтирования образов – это IMDisk (версия 1.1.3 была выпущена 5 декабря 2008 года, см. www.ltr-data.se/opencode.html), драйвер виртуального диска, устанавливающий утилиту командной строки и добавляющий элемент панели управления, который предоставляет доступ к графическому интерфейсу для драйвера. На илл. 9.4 показан интерфейс пользователя программы IMDisk, в которой монтирован образ как накопитель (H:\) в режиме только для чтения.



Илл. 9.4. Интерфейс программы IMDisk, в которой монтирован образ как накопитель H:\.

Совет

Microsoft предоставляет (хотя не поддерживает и не рекламирует) бесплатный инструмент, который называется Virtual CD-ROM Control Panel for XP. Этот инструмент создает в панели управления Windows XP виртуальное устройство чтения компакт-дисков, которое можно использовать для монтирования файлов в формате .iso (обычно создаваемых с CD- или DVD-дисков) как файловые системы.

Прямая ссылка на этот инструмент довольно длинная, но ее можно найти на веб-сайте Microsoft (<http://msdn.microsoft.com/en-us/subscriptions/aa948864.aspx>; прокрутите примерно две трети страницы вниз), а также в таких блогах как RaDaJo (<http://radajo.blogspot.com/2006/09/mounting-cddvd-iso-images-in-windows.html>) и help.net (<http://weblogs.asp.net/pleloup/archive/2004/01/15/58918.aspx>).

Анализ файлов

Часто эксперту нужно исследовать отдельные файлы, а не всю файловую систему или целый том. Во многих случаях эти файлы могут иметь собственные форматы (например, файлы корзины INFO2 в Windows, которые мы рассматривали в главе 5), и для них может отсутствовать подходящее средство просмотра.

Утилиты хэширования

При извлечении файлов из образа данных вы, возможно, захотите вычислить криптографический хэш этих файлов, чтобы позднее проверить их целостность. Алгоритмы хэширования – это криптографические вычисления, при которых обычно берутся входные данные переменной длины и возвращаются уникальные выходные данные фиксированной длины. Если хотя бы один бит в файле изменится, то хэш-значение изменится также, что свидетельствует о полезности хэш-значений файлов. Джесси Корнблум (Jesse Kornblum) разработал программу хэширования MD5Deep (<http://md5deep.sourceforge.net>), которая создает и сравнивает не только хэш-значения MD5 для файлов, но и хэш-значения SHA-1, SHA-256, Tiger и Whirlpool. Все программы хэширования имеют интерфейс командной строки, что позволяет использовать их в пакетных файлах, чтобы автоматизировать их развертывание.

Хэш-значения файлов можно использовать не только для проверки целостности данных, но и для того, чтобы быстро определить, был ли файл, с которым вы работаете, уже идентифицирован как вредоносный. На веб-сайт VirusTotal (www.virustotal.com) можно передать хэш файла для сравнения по базе данных, а не отправлять весь файл. Поэтому, если интересующий вас файл очень большой, или вы не хотите отправлять по Интернету копии вредоносных файлов, вы можете использовать возможность этого веб-сайта, чтобы выполнить быструю проверку. Ведь это не потребует от вас больших усилий и сделает ваши записи о деле или итоговые отчеты более полными независимо от того, являетесь вы консультантом или сотрудником правоохранительных органов.

Еще один инструмент хэширования от Джесси Корнблума – это ssdeep (<http://ssdeep.sourceforge.net>), отличное средство для вычисления нечетких хэш-значений. Этот способ хэширования позволяет сравнивать похожие, но не идентичные файлы путем определения вероятности сходства между файлами. Я использовал этот инструмент хэширования при сравнении двух файлов примерно одинакового размера и с одним и тем же именем, которые были собраны в результате расследований двух разных инцидентов, и обнаружил, что эти файлы были похожи на 98–99 процентов.

Шестнадцатеричные редакторы

Хороший шестнадцатеричный редактор, с которым вам удобно работать, может быть незаменимым инструментом для судебного анализа. Вам часто будут встречаться двоичные файлы, которые нужно будет открыть и просмотреть, а текстовые процессоры просто не смогут перевести эти данные в подходящий формат. В моей практике было несколько случаев, когда я получал необычные ответы из приложений для анализа, или когда Perl-скрипт, который я писал для анализа двоичного содержимого файла, просто не работал, и мне приходилось открывать этот файл в шестнадцатеричном редакторе, чтобы просмотреть его двоичное/шестнадцатеричное содержимое и увидеть, в чем может быть проблема. Примером того, как я применял этот способ, является исследование различий между файлами упреждающей выборки в Windows XP и Vista (см. главу 5).

Я предпочитаю редактор UltraEdit (www.ultraedit.com), потому что я использую его как в качестве среды программирования, так и в качестве шестнадцатеричного редактора. Мне нравятся многие возможности этого приложения (в нем видны номера строк, и если Perl-скрипт не работает, я могу быстро найти ошибку), поэтому я с удовольствием плачу за него номинальную плату. К другим возможностям этого приложения относится

выделение синтаксических конструкций Perl, автоматический отступ, возможность открывать *действительно* большие журналы и двоичные файлы и параллельно просматривать шестнадцатеричное и двоичное содержимое файла. Однако, прежде чем я выбрал UltraEdit, я изучил несколько бесплатных приложений, чтобы узнать, какие функциональные возможности доступны и каким из них я отдаю предпочтение. Во время этого изучения мне встретилось несколько других приложений, например:

- § Cygnus Hex Editor Free Edition (www.softcircuits.com/cygnus/fe)
- § XVI32 (www.chmaas.handshake.de/delphi/freeware/xvi32/xvi32.htm)
- § Free Hex Editor Neo от компании HDD Software (www.hhdsoftware.com/Products/home/hex-editor-free.html)
- § HexEdit (www.physics.ohio-state.edu/~prewett/hexedit)

Имейте в виду, что это только примеры шестнадцатеричных редакторов, и вам, как эксперту, нужно выяснить, какой из них подходит вам. Если вы заинтересованы в изучении других опций, Википедия содержит сравнительную таблицу шестнадцатеричных редакторов (http://en.wikipedia.org/wiki/Comparison_of_hex_editors) для различных платформ, которую вы, возможно, захотите просмотреть или даже использовать в качестве основы для собственного исследования необходимых вам инструментов.

Сетевые инструменты

Надеюсь, что, читая эту книгу, вы поняли, что анализ систем, особенно во время расследования инцидентов, не ограничен только анализом операционных систем хост-компьютеров (хотя данному вопросу уделяется основное внимание в этой книге). Часто признаки взлома, вторжения или заражения вредоносной программой можно в первую очередь обнаружить в оповещениях систем обнаружения вторжений, в необычных данных журнала брандмауэра или просто в подозрительном сетевом трафике. В других ситуациях дополнительную информацию о вторжении (например, обмен данными между системами, место возникновения сетевых соединений, место назначения исходящих данных (в случае кражи информации)) можно будет определить только посредством сбора и анализа сетевых данных. Сбор и анализ сетевых данных – отдельная тема, которая выходит за рамки данной книги, но она так важна, что здесь будет представлено несколько необходимых инструментов, которые можно использовать для этих целей.

Сканирование

Одна из главных проблем, которая стоит перед сообществом людей, занимающихся расследованием инцидентов и проведением судебных анализов, заключается (по моему скромному мнению) в отсутствии взаимодействия между этими специалистами и специалистами, занимающимися поиском и устранением уязвимостей. Уязвимости выявляются и подтверждаются едва ли не ежедневно, и вскоре после этого рабочий эксплойт, использующий одну из новых уязвимостей, может быть опубликован в Интернете или обнаружен во время расследования инцидента. Кроме того, существуют компании, бизнес-модель которых состоит в том, чтобы искать, находить и подтверждать уязвимости в программных продуктах, а затем обеспечивать своим клиентам защиту от этих уязвимостей.

Отсутствие взаимодействия проявляется в том, что, когда обнаруживается уязвимость, очень редко проводятся исследования, чтобы определить артефакты, оставшиеся во взломанной системе после использования эксплойта. Для того чтобы уязвимость была успешно использована, в системе, как правило, должна быть служба или программа, которая прослушивает сетевой порт (например, Microsoft SQL Server прослушивает TCP-порт 1433, ожидая соединений) и подвержена действию эксплойта. Таким образом, исследователь должен иметь определенные средства подтверждения того,

что эксплойт был применен успешно. Результатом успешного применения эксплойта будет взломанная система, которую позже можно проанализировать на наличие артефактов, связанных с эксплойтом.

Приложения сканирования используются при оценке уязвимостей с целью определить возможные бреши в системе безопасности инфраструктуры, чтобы затем можно было разработать приоритетный, комплексный план для уменьшения поверхности атаки этой инфраструктуры. Это означает, что в результате определения уязвимостей и применения последующих мер по их устранению (установка исправлений для системы, загрузка обновлений и настройка безопасной конфигурации для приложений) значительно уменьшаются возможности злоумышленника получить доступ к сетевой инфраструктуре. Те же приложения сканирования можно использовать во время анализа с целью определить поверхность атаки исследуемых систем, чтобы у вас был способ установить, как эти системы могли быть взломаны. Например, в октябре 2008 года корпорация Microsoft выпустила «внеплановое» исправление, обозначенное как MS08-067 (<http://blogs.technet.com/msrc/archive/2008/10/23/ms08-067-released.aspx>) и устраняющее критическую уязвимость в службе Windows Server. Если бы вы исследовали взломанную систему Windows XP два месяца спустя и обнаружили, что критически важное исправление для данной уязвимости не было установлено, этот факт мог бы указать вам, в каком направлении проводить анализ, особенно если вредоносная программа, используемая в системе после применения эксплойта, была бы такой новой, что ее нельзя было обнаружить антивирусным приложением.

Существует несколько свободно доступных инструментов, которые могут помочь вам при проведении анализа, особенно если вы пытаетесь определить, какие уязвимости или «векторы атаки», *возможно*, использовались в инциденте. Вы можете использовать эти инструменты на работающих компьютерах, а также можете загрузить образ при помощи LiveView (<http://liveview.sourceforge.net>) и затем сканировать систему (системы, загруженные при помощи LiveView, по умолчанию не имеют возможностей работы в сети), чтобы получить представление относительно того, какая уязвимость, возможно, существовала в системе. Например, можно воспользоваться программой Baseline Security Analyzer (<http://technet.microsoft.com/en-us/security/cc184924.aspx>) от Microsoft, чтобы сканировать систему и определить, какие обновления или исправления для отдельных уязвимостей отсутствуют в системе.

Сканирование сети позволяет получить очень полезную информацию во время расследования инцидента или при проведении анализа образа данных. Но имейте в виду, что, для того чтобы данный тип сканирования оказался эффективным при проведении анализа образа, этот образ должен быть загружен в среду с поддержкой работы в сети. Возможно, самая популярная утилита для сканирования сетей – это Nmap (www.Nmap.org). Помимо простого сканирования портов, Nmap может определять операционную систему компьютера и службы, а с появлением графического интерфейса Zenmap в Nmap добавилась возможность составлять карту топологии сети.

Инструменты и ловушки...

Инструменты, поддерживающие Nmap

Существует несколько бесплатных инструментов, которые помогут вам проанализировать результаты утилиты Nmap. Один из таких инструментов, который особенно полезен при широкомасштабных сканированиях, – это fe3d (<http://projects.icapsid.net/fe3d>), средство визуализации данных, обладающее возможностью отображать выходные данные сканирований Nmap в графическом формате. Кроме того, специально для работы с утилитой Nmap разработано несколько Perl-модулей, в том числе Nmap::Scanner, Nmap::Parser и Nmap::Parser::XML. Последние два модуля позволяют анализировать выходные данные Nmap, упорядочивая и предварительно обрабатывая их (т. е. выполняя поиск отдельных систем или служб и т. д.) при необходимости.

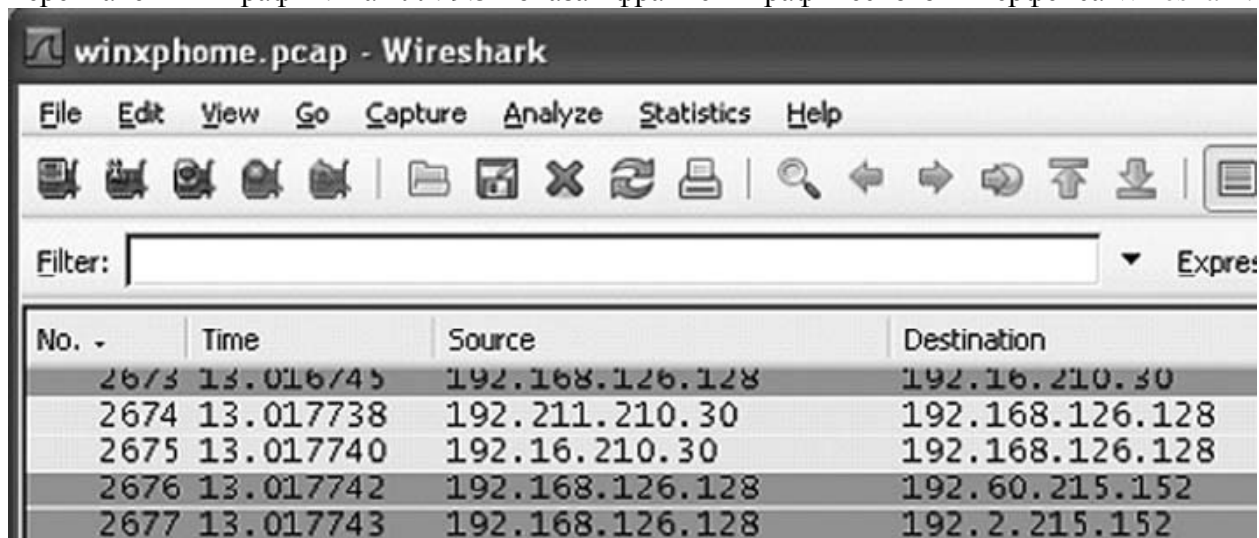
Сканирование системы не ограничивается простым сканированием портов и определением операционной системы компьютера и имеющихся служб. Сканирование на наличие уязвимостей, например, может быть важной частью анализа и продвинуть этот анализ на один шаг вперед. Отличные инструменты для данной цели – это Nessus (www.nessus.org/nessus) и Sara (www-arc.com/sara), причем Nessus является самым популярным и известным из этих двух. Оба инструмента находятся в списке 100 лучших инструментов обеспечения безопасности в сети (<http://sectools.org>) наряду с другими приложениями сканирования.

Перехват и анализ пакетов

Еще одна задача, с которой вы можете столкнуться во время расследования инцидента, – перехват и анализ сетевого трафика. Независимо от того, как вы будете выполнять эту задачу: самостоятельно, совместно с сотрудниками ИТ-отдела клиента или получите перехваченный трафик в виде данных от другого эксперта, вы можете столкнуться с необходимостью перехватывать и анализировать трафик.

Два популярных инструмента для перехвата и анализа сетевых пакетов для ОС Windows – это Wireshark (www.wireshark.org) и NetworkMiner (<http://sourceforge.net/projects/networkminer>). Эти бесплатные инструменты чрезвычайно полезны для любого специалиста по расследованию инцидентов.

На момент написания этой книги приложение Wireshark версии 1.0.3 было доступно для платформы Windows. Wireshark позволяет не только перехватывать сетевой трафик (на основе выбранного вами сетевого интерфейса), но и анализировать перехваченный трафик. На илл. 9.5 показан фрагмент графического интерфейса Wireshark.



Илл. 9.5. Фрагмент графического интерфейса приложения Wireshark версии 1.0.3.

Одна из особенностей Wireshark, которую я нахожу очень полезной, – возможность полностью собирать повторно TCP-поток. Для этого, когда сетевой трафик будет загружен в Wireshark, щелкните по пункту «Анализ» (“Analyze”) в строке меню и выберите команду «Следить за TCP-поток» (“Follow TCP Stream”) в раскрывающемся меню. Wireshark будет следить за потоком и полностью соберет повторно содержимое данных, передаваемых по TCP. Эта функция может быть очень полезной при изолировании отдельного соединения, а также при восстановлении диалога (трафик между двумя конечными точками). Например, вы можете восстановить веб-страницы, которые видит пользователь, сообщения электронной почты, обмен данными с сервером контроля и управления в ботнете или обмен незашифрованными мгновенными сообщениями. Wireshark предоставляет возможность проводить подобный анализ с UDP- и SSL-пакетами.

Инструменты и ловушки...

Перехваченный сетевой трафик

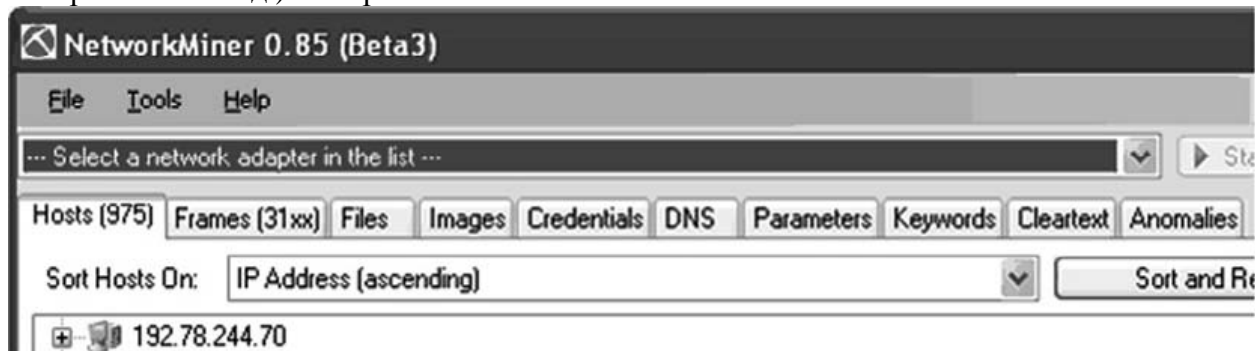
Пока мы говорим о перехвате сетевого трафика, я здесь хочу напомнить вам, как перехваченный трафик вписывается в общую картину расследования инцидентов. Многие инциденты связаны с тем или иным сетевым компонентом – система заражается в результате загрузки какого-нибудь объекта из Интернета, а затем заражение распространяется на другие системы в сети; злоумышленник получает доступ к системе и управляет ей, или бот попадает в систему и обращается к серверу контроля и управления, ожидая команд. Многие инциденты, независимо от их типа, связаны в некоторой степени с каким-нибудь сетевым компонентом. Вот почему перехваченный сетевой трафик может быть очень полезным источником данных. Во-первых, вы найдете информацию о самих пакетах, в том числе IP-адреса и порты источника и назначения. Эта информация позволит вам определить (1) компьютеры, связанные с инцидентом (на основе IP-адреса), и (2) какие программы, возможно, связаны с инцидентом, если вам удастся сопоставить сведения о порте с энергозависимыми данными (результаты утилит «tcpvcon.exe», «netstat.exe» или данные, полученные в результате анализа дампа памяти), собранными как минимум с одного исследуемого компьютера. Во-вторых, информация в этих пакетах (часто собранная повторно из «диалогов» по ТСР) может очень много рассказать о том, обмен какими данными совершался. Такая информация будет чрезвычайно полезна, когда дело связано с несанкционированным копированием данных (т. е. возникает вопрос о том, какие данные были похищены из системы).

В строке меню Wireshark также есть пункт «Статистика» (“Statistics”), предоставляющий доступ к инструментам, которые помогут вам сузить область поиска или отфильтровать огромное количество данных, чтобы найти ту самую иголку в стоге сена. Можно просмотреть общие статистические данные о перехваченных пакетах, подробный список сетевых диалогов в перехваченных пакетах или просто список конечных точек. Все эти сведения помогут вам тщательно исследовать килобайты или даже мегабайты данных.

Иногда графический интерфейс предоставляет больше возможностей, чем вам необходимо, и использование инструментов с интерфейсом командной строки, возможно, будет более целесообразным. В таких случаях можно также работать с приложением Wireshark, в состав которого входит несколько инструментов командной строки, в том числе tshark, tcpdump и dumpcap. Согласно информации на веб-сайте Wireshark, каждый из этих инструментов, как и любой другой, имеет свои преимущества и недостатки. Хотя инструменты командной строки отлично подходят для загрузки на удаленные компьютеры или для запуска из других мест в сети с целью перехвата сетевого трафика, утилита tcpdump по умолчанию перехватывает только первые 68 байт, усекая информацию. Еще один инструмент командной строки – это windump (www.winpcap.org/windump), который не только перехватывает сетевой трафик, как tshark и dumpcap, но и может быть использован с соответствующими драйверами для сбора сетевого трафика посредством точек доступа беспроводной сети.

Приложение NetworkMiner версии 0.85 (бета) доступно для ОС Windows. NetworkMiner описывается на веб-сайте Sourceforge как «сетевой инструмент судебного анализа для ОС Windows, который может определять операционную систему, имя компьютера и открытые порты компьютеров в сети посредством перехвата и анализа сетевых пакетов или анализа файла PCAP». Кроме того, «NetworkMiner также может извлекать из сетевого трафика файлы, передаваемые по сети». Эти возможности делают

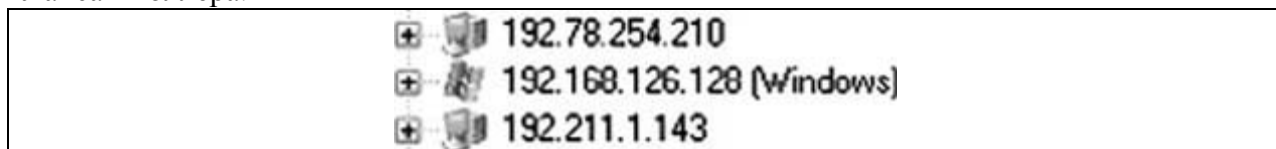
NetworkMiner чрезвычайно важным инструментом для специалиста по расследованию инцидентов. Как показано на илл. 9.6, графический интерфейс приложения NetworkMiner содержит несколько вкладок для отображения автоматически анализируемой информации (в том числе файлов, учетных данных пользователей, изображений и т. д.) из перехваченных сетевых пакетов.



Илл. 9.6. Фрагмент интерфейса приложения NetworkMiner версии 0.85 (бета).

Для ОС Linux существует инструмент tcpxtract (<http://tcpxtract.sourceforge.net>) разработанный Ником Харбором (Nick Harbour), который представляет собой утилиту восстановления файлов, предназначенную для перехваченного сетевого трафика. Tcpxtract позволяет выполнять в перехваченном сетевом трафике поиск файлов на основе библиотеки сигнатур файлов. Хотя инструмент tcpxtract не доступен для систем Windows, приложение NetworkMiner обладает похожими функциональными возможностями.

Снимок экрана со страницы проекта NetworkMiner на веб-сайте Sourceforge показывает, что это приложение может определять операционную систему компьютера посредством анализа перехваченных пакетов. NetworkMiner использует функциональные возможности, заимствованные у утилиты p0f (<http://lcamtuf.coredump.cx/p0f.shtml>), чтобы определять операционную систему компьютера пассивно, посредством перехвата сетевых пакетов, а не при помощи активного сканирования (как Nmap). На илл. 9.7 показано, как NetworkMiner определяет операционную систему компьютера, используя перехваченные пакеты, предоставленные как часть одной из практических задач по судебному анализу Ланса Мюллера.



Илл. 9.7. Фрагмент интерфейса NetworkMiner, в котором показано, как приложение определяет операционную систему компьютера.

Еще один инструмент (с графическим интерфейсом пользователя) для перехвата и анализа сетевого трафика – это PacketMon (www.analogx.com/contents/download/network/pmon.htm). Похоже, что на момент написания этой книги PacketMon не обновлялся, и он не обладает таким богатством функций, как Wireshark или NetworkMiner, но это полезный инструмент, который поможет вам познакомиться с основами анализа перехваченного сетевого трафика.

Инструмент командной строки (для тех, кто любит такие вещи), который может помочь при анализе сетевого трафика, – это ngrep (<http://ngrep.sourceforge.net/download.html>), версия утилиты grep, применяемая к сетевому уровню и позволяющая использовать расширенные регулярные или шестнадцатеричные выражения, чтобы выполнять поиск по шаблону в перехваченном сетевом трафике.

Все эти четыре инструмента позволяют получить доступ к перехваченному сетевому трафику; первые три – инструменты с графическим интерфейсом пользователя, которые предоставляют возможность перехватывать сетевой трафик, а также анализировать его. Эти инструменты позволяют не только перехватывать сетевой трафик, но и работать с перехваченным сетевым трафиком, предоставленным из любого

источника. Это означает, что, если вы расследуете инцидент, а специалисты клиента уже собрали сетевой трафик, вы можете использовать вышеупомянутые инструменты, чтобы проанализировать эти собранные данные, при условии что они имеют приемлемый формат. Большинство инструментов предоставляют доступ к перехваченным данным в формате tcpdump, который, как и формат dd для образа данных, может считаться стандартом де-факто для перехваченного сетевого трафика.

Совет

Осенью 2008 года компания NetWitness выпустила приложение Investigator, которое бесплатно доступно по адресу <http://download.netwitness.com/download.php?src=DIRECT>. Используя Investigator, эксперт может быстро импортировать файл в формате .pcap, содержащий собранный сетевой трафик, или просто перехватывать сетевой трафик самостоятельно. Investigator описывается как «приложение семейства продуктов NextGen для интерактивного анализа возможных угроз», которое предоставляет эксперту возможность проводить «произвольный контекстуальный анализ необработанных сетевых данных». Это чрезвычайно полезный инструмент для эксперта, но не забудьте внимательно прочитать лицензионное соглашение перед загрузкой и использованием этого инструмента.

Инструменты и ловушки...

Snort

Одно из бесплатных приложений, которое достаточно редко выбирают для анализа перехваченного сетевого трафика, – это Snort (www.snort.org). Данное приложение широко известно как бесплатная система обнаружения вторжений (СОВ). Когда я впервые узнал о приложении Snort, это была обычная СОВ, но в последующие годы были приложены большие усилия, направленные на усовершенствование этого инструмента, и в результате его также стали называть системой предотвращения вторжений. Одна из особенностей Snort состоит в том, что приложение может не только прослушивать сетевой интерфейс (в неизбирательном режиме), перехватывая и фильтруя трафик работающей сети, но и с тем же успехом работать, используя только файлы с перехваченным сетевым трафиком. Указывая Snort прочитать сетевой трафик из файла в формате .pcap и обработать перехваченный трафик при помощи наборов правил, вы получаете возможность, похожую на использование утилиты ngrer с предварительно созданными наборами фильтров, некоторые из которых намного сложнее, чем регулярные выражения. Эта возможность может помочь вам предварительно обработать данные, чтобы уменьшить их объем. Представьте ситуацию, когда сетевой червь быстро распространяется в сетевой инфраструктуре. Перехваченный сетевой трафик можно использовать, чтобы определить, какие системы обмениваются данными в сети, а в крупных сетях, как правило, происходит множество «нормальных» обменов данными, что может легко обескуражить эксперта. Используя Snort (и при условии, что имеется сигнатура для обмена данными этого червя в сети), вы можете легко «найти иголку в стоге сена», выполнив предварительную обработку данных с таким уровнем детализации, который намного выше, чем у других инструментов.

Утилиты поиска

Всякий раз, когда я разговариваю с коллегами об основных возможностях приложений для судебного анализа, то одной из важнейших функций, необходимой для любого подобного приложения называется возможность выполнять поиск, в том числе поиск по ключевым словам и поиск с использованием регулярных выражений. Операции поиска обычно используются как прием для уменьшения объема исследуемых данных; используя ключевые слова или регулярные выражения, эксперт может прочесывать

мегабайты или даже гигабайты данных, чтобы найти элементы (файлы, записи журналов событий и т. д.), которые актуальны для проводимого им анализа.

Инструменты и ловушки...

Поиск в реестре

Поиск данных формата ASCII или Unicode в большинстве файлов ОС Windows, как правило, является простым процессом. Однако поиск в файлах кустов реестра может быть связан с некоторыми трудностями. Например, эксперты должны обращать самое пристальное внимание на пути в реестре. Путь, содержащий строку «SessionManager», полностью отличается от пути со строкой «Session Manager». Точно так же, если раздел или параметр реестра находится в пути, содержащем «Windows NT», не нужно выполнять поиск «WindowsNT». Как и при проведении большинства видов поиска, орфография играет важную роль. Помимо этого, не вся информация в реестре хранится в чистом формате ASCII или Unicode. Некоторая информация закодирована в параметре DWORD (4 байта), и этот параметр нужно сопоставить с разделом, чтобы интерпретировать его данные. В некоторых параметрах DWORD значение «0» может означать, что функция включена, тогда как в других параметрах то же самое может быть обозначено значением «1». В других случаях определенная функциональная возможность может быть закодирована в двоичном параметре каким-нибудь образом. Поэтому не удивляйтесь, если путем поиска по ключевым словам или регулярным выражениям не удалось найти признаки искомых данных в файлах кустов. Иногда, самое главное при проведении поиска – знать, что вы ищите, и корректировать поиск вручную.

Утилиты поиска, указанные в этом разделе, предназначены для применения к файлам на работающем компьютере, то есть их можно использовать во время исследования работающей системы или после того, как вы загрузите образ как систему, работающую в реальном времени. Большинство коммерческих приложений для анализа данных имеют встроенную функцию поиска (а некоторые, такие как FTK и X-Ways Forensics, предоставляют возможности индексирования), а также предварительно сконфигурированные строки для поиска с использованием регулярных выражений.

Отличный источник утилит поиска в файлах и файловых системах – это веб-сайт *GNU utilities for Win32* (<http://unxutils.sourceforge.net>). Этот сайт содержит архив UNIX-подобных утилит, предоставляющих большое количество функциональных возможностей, с которыми знакомы многие администраторы UNIX, хотя все эти инструменты предназначены для работы в Windows. Эти инструменты можно легко добавлять в пакетные файлы и скрипты для использования при проведении поиска или в других операциях по уменьшению объема исследуемых данных.

Помимо этих инструментов, существует несколько версий утилиты grep для платформы Windows. На самом деле, таких версий две, и обе они называются «grep for Windows»; одна доступна на сайте Sourceforge (<http://gnuwin32.sourceforge.net/packages/grep.htm>), а другая – на сайте InterLog (<http://pages.interlog.com/~tcharron/grep.html>). Обе версии предоставляют похожие функциональные возможности.

В некоторых случаях вам, возможно, придется выполнять поиск отдельных элементов или строк, например, номеров социального страхования или номеров кредитных карт. Эти элементы попадают под определение «конфиденциальные данные», как указано в законе SB-1386 штата Калифорния и в стандарте защиты информации в индустрии платежных карт, разработанном Visa. В связи с этим возможны случаи, когда вам придется выполнять поиск такого вида данных в качестве отдельной функции вашего анализа. К счастью, существует несколько инструментов, которые можно использовать для этих целей. Один из таких инструментов – это Spider (www.cit.cornell.edu/security/tools), который предназначен для проверки наборов файлов

(файлов на накопителе, веб-сайтов и т. д.) на наличие конфиденциальных данных, таких как номера социального страхования или номера кредитных карт. В результате запуска программы Spider создается файл журнала с информацией обо всех файлах, содержащих конфиденциальные данные.

Еще один полезный инструмент для поиска номеров кредитных карт – это ccsrch (<http://sourceforge.net/projects/ccsrch>). Ccsrch – это утилита командной строки для Windows, которая может искать непрерывные и незашифрованные номера кредитных карт, а также данные дорожек магнитной полосы карт. Условия форматирования для данных первой и второй дорожек на магнитной полосе кредитной карты включают в себя номер кредитной карты или основной номер счета (PAN) в виде непрерывных данных, то есть последовательность чисел без разрывов, пробелов или тире. Результаты поиска утилиты ccsrch, которые содержат имя файла, а также найденный номер, отправляются на стандартное устройство вывода, что позволяет легко перенаправить их в файл.

Ниже представлен список ресурсов, которые помогут вам при выполнении поиска:

- § справочная информация по регулярным выражениям (www.regular-expressions.info);
- § форматы номеров кредитных карт (http://en.wikipedia.org/wiki/Credit_card_number);
- § регулярные выражения для номеров кредитных карт (www.regular-expressions.info/creditcard.html).

Инструменты и ловушки...

Поиск конфиденциальных данных

При поиске конфиденциальных данных любого типа необходимо удостовериться, что вы полностью понимаете характер и формат этих данных, а также то, что на самом деле означают результаты поиска. Так, поиск номеров социального страхования и номеров кредитных карт может быть связан с определенными проблемами, касающимися форматирования. Большинство экспертов распознают форматы этих номеров, но нужно удостовериться, что, например, поиск номера кредитной карты, состоящего из 16 цифр, включает в себя поиск, который отвечает необходимым критериям не только для номера кредитной карты (т. е. длина, начальные цифры, проверка с использованием алгоритма Луна), но и для прямой последовательности чисел без разрывов, а также последовательности чисел, содержащей пробелы или тире в соответствующих местах. Еще одна трудность, связанная с поиском конфиденциальных данных, состоит в том, что необходимо тестировать свои инструменты, чтобы определить, какие форматы данных они могут искать. Некоторые инструменты выполняют поиск только непрерывных последовательностей чисел (как в номерах кредитных карт и номерах социального страхования), тогда как другие могут выполнять поиск этих номеров, отформатированных с пробелами или тире.

Краткое изложение

Основа экспертизы – это не инструменты, которые вы используете, а ваша методика работы. Хорошая методика работы не зависит от того, какой инструмент вы применяете: коммерческий пакет программ для судебного анализа, бесплатное приложение с открытым исходным кодом или специально созданный Perl-скрипт. Самое главное – знать, на какие вопросы вам нужно ответить, где искать данные, и как затем правильно извлечь эти данные и интерпретировать их в отчете. Учитывая эти факторы и свои основные принципы, вы сможете выбрать правильный инструмент для необходимой задачи, будь то извлечение данных для анализа или подкрепление полученной информации другими данными.

Быстрое повторение

Документирование анализа

- § Документирование – важная часть любой экспертизы. Документация должна быть понятной, точной и достаточно подробной, чтобы вы или (в особенности) другие эксперты могли позднее повторить и подтвердить результаты.
- § Многие сотрудники ИТ-отделов и специалисты по расследованию инцидентов должны передавать собранные данные в суд и обращать особое внимание на соответствие необходимым для этого стандартам. Главной отличительной особенностью вашего анализа будет ваша документация (приемы работы и действия должны быть задокументированы так, чтобы их смог понять и при необходимости повторить другой специалист).

Инструменты

- § Существует несколько бесплатных или недорогих инструментов, которые могут более чем в достаточной мере заменить или даже расширить функциональные возможности, свойственные коммерческим пакетам приложений. Обладая определенными знаниями и заранее обдумывая свои намерения, вы сможете увеличить, заменить или даже превзойти возможности, имеющиеся в таких приложениях.
- § Как и в случае с другими аспектами судебного компьютерного анализа и расследования инцидентов, различные коммерческие инструменты имеют свои преимущества и недостатки. Бывают случаи, когда необходимо использовать коммерческое приложение для анализа и представления данных. Однако иногда бесплатные инструменты предоставляют больший уровень глубины и наглядности данных и намного быстрее дают ответы на имеющиеся вопросы.

Часто задаваемые вопросы

Вопрос: Мне нужно провести анализ данных, которые я собрал. Какие инструменты мне использовать?

Ответ: Как и в любых других случаях, все зависит от обстоятельств. Серьезно. Прежде чем решать, какой инструмент использовать, вам нужно определить и задокументировать цели анализа, потому что от них зависят все этапы экспертизы. Вы ищете незаконные изображения? Вы хотите узнать, кем (т. е. с использованием какой учетной записи) были загружены, открыты или просмотрены эти изображения? У вас есть несколько мегабайт или гигабайт журналов IIS-сервера, и вам нужно определить, была ли совершена атака с внедрением SQL-кода? Не существует одного единственного инструмента, который бы подходил к каждой ситуации, и во многих случаях выбор инструмента зависит от личных предпочтений; я выполняю анализ файлов журналов, используя Perl-скрипты, тогда как другие предпочитают использовать программу Log Parser от Microsoft.

Вопрос: Каким образом перехваченный сетевой трафик может быть полезен для экспертизы?

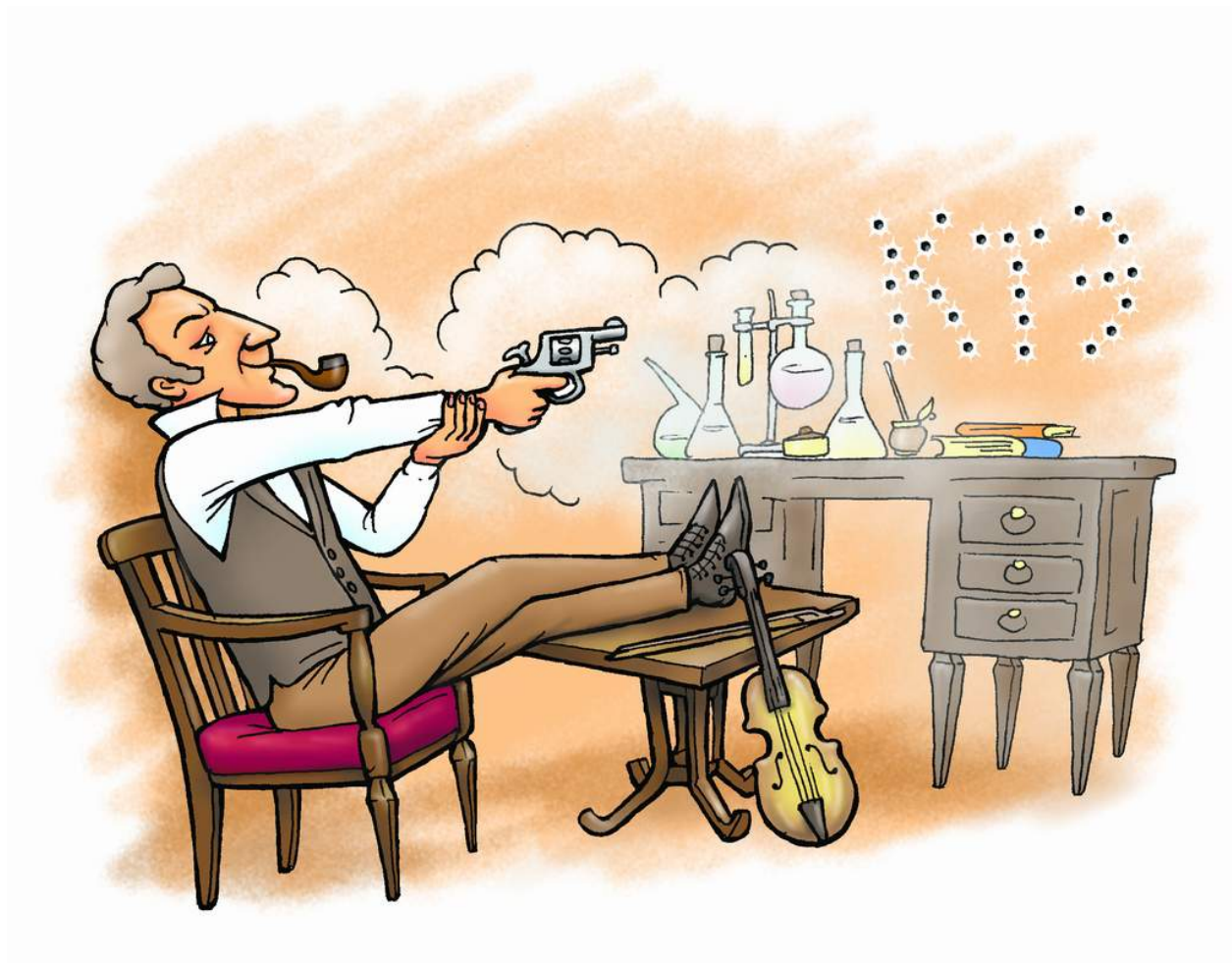
Ответ: Перехваченный сетевой трафик содержит много полезной информации, которую можно связать с системой (IP-адрес), а также с процессом, выполняющимся в этой системе, посредством сопоставления сведений о порте в заголовке пакета с энергозависимыми данными (сетевые подключения, процесс, сопоставление процесса с портом) из системы. Содержимое пакетов может подсказать вам, какие данные передавались в систему или из нее.

Вопрос: Во время экспертизы у меня есть несколько источников данных (т. е. перехваченные сетевые пакеты, журналы сетевых устройств, журналы серверов и образы систем), которые мне нужно сопоставить и связать между собой? Как лучше всего это сделать?

Ответ: В данном случае – посредством документирования. Я не знаю ни одного пакета приложений, который позволяет добавлять, анализировать и сопоставлять несколько источников данных, за исключением PyFlag.

Содержание

Введение	2
Документирование анализа	3
Инструменты	7
Создание образов данных	7
dd	7
FTK Imager	9
Анализ образа	10
The SleuthKit	10
PyFlag	13
ProDiscover Basic	14
Монтирование файла-образа	15
Анализ файлов	17
Утилиты хэширования	17
Шестнадцатеричные редакторы	17
Сетевые инструменты	18
Сканирование	18
Перехват и анализ пакетов	20
Утилиты поиска	23
Краткое изложение	25
Быстрое повторение	26
Часто задаваемые вопросы	26



<http://computer-forensics-lab.org>

Перевод:
Бочков Д.С.
Капинус О.В.
Михайлов И.Ю.