жучки в электронных письмах

крис касперски, no-email

электронная почта была и остается одним из основным инструментов хакера (достаточно вспоминать письма с зловредными вложениями или эксплуатацию "дыр" популярных почтовых клиентов). атакующие применяют все более и более изощренные приемы, зачастую неизвестные широкой аудитории администраторов и специалистов по безопасности. сегодня мы поговорим о… ссылках на картинки и покажем на что они способны, а способы они на многое!

введение

Хорошо подготовленная и продуманная атака ("пионеров" и вандалов мы в расчет не берем) начинается с тщательного сбора информации о жертве, реконструкции топологии локальной сети (если у жертвы есть сеть), определении типа и версий используемого программного обеспечения, выявления защитных комплексов (спам-фильтров, брандмауэров, систем обнаружения вторжений, etc), а так же составления графика работы жертвы для осуществления атаки в наиболее "удобное" с точки зрения хакера время.

Существует множество утилит, позволяющих сканировать локальные сети, не взирая на брандмауэр (например, знаменитый nmap), однако, все они работают только с ущербленными (или неверно настроенными) брандмауэрами и, к тому же, легко распознаются системами обнаружения вторжений, что, естественно, не входит в планы атакующего, желающего остаться незаметным.

Неожиданным хакерским подспорьем оказалась… электронная почта. Отвечая на хакерское письмо (вполне безобидное со всех точек зрения), жертва сама того не подозревая, включает в заголовок ответа не только данные об установленном программном обеспечении, но и более "тонкую" информацию (например, номера локальных портов, по стратегии назначения которых, при длительной переписке, можно выявить наличие транслятора сетевых адресов или сделать вывод о степени загруженности узла).

Более "продвинутые" хакерские письма, "закодированные" в HTML-формате, используют Java-скрипты и ссылки на внешние (подконтрольные хакеру) ресурсы, что позволяет собрать намного больше информации и не требует от жертвы ответа — достаточно просто просмотреть письмо (а в популярных почтовых клиентах просмотр включен по умолчанию и очень мало кто из пользователей отключает его).

Java-скрипты, относящиеся к "тяжелой артиллерии", мы рассматривать не будем, вооружившись одной лишь "снайперской винтовкой" — внешними ссылками на картинки, возможности которых сильно недооцениваются как хакерами, так и специалистами безопасности.

>>> врезка сокрушая барьеры или HTML-во вложениях

Outlook Express (и некоторые другие почтовые клиенты), обрабатывают HTML-содержимое электронных писем весьма специфическим образом, помещая их в "песочницу" с минимальными привилегиями, которые в лучшем (для хакера) случае совпадают с привилегиям удаленных Internet-узлов, то есть Java-скрипт не имеет доступа ни к каким локальным файлам и далеко не во всех случаях может устанавливать соединение даже с тем сервером откуда он был загружен, что существенно затрудняет атаку.

Совсем другое дело — HTML-страницы, открываемые с локального диска. Поскольку, никакого "узла-родителя" здесь нет, Java-скрипту, как правило, дозволено устанавливать любые TCP/IP соединения (которым, естественно, не препятствует брандмауэр), а так же обращаться (пускай и не без ограничений) к локальным файлам. В любом случае, локально отрываемый HTML-документ, автоматически попадает в зону "доверенных узлов", обладающую (по умолчанию) наивысшими привилегиями. Существует большое количество атак, тем или иным образом вынуждающим жертву сохранить HTML-страницу на диск и открыть ее, чтобы повысить уровень привилегий Java-скриптов.

HTML-вложения — в этом смысле идеальный вариант. Чтобы просмотреть содержимое вложения, его необходимо открыть, а при этом _всегда_ происходит копирование на диск. Даже если жертва не выбирает опцию "сохранить", почтовый клиент копирует HTML-вложение во временный файл, передавая его браузеру со всеми вытекающими отсюда последствиями. Как

защититься от атак подобного типа? Очень просто — ассоциировать HTML/HTM-расширения со специальным браузером, в настройках которого отключить поддержку скриптов, ActiveX компонентов и всего-всего-всего. Или же (если браузер поддерживает различные зоны безопасности, как, например, IE) "урезать" права локальной зоны.

руководящая идея

Внедряем в HTML-письмо ссылку на картинку, расположенную на подконтрольном хакеру WEB-сервере, и отправляем его жертве. В момент просмотра письма, почтовый клиент (если только в нем не отключена загрузка картинок) обращается к хакерскому WEB-серверу, передавая в заголовке запроса огромное количество "чувствительной" информации. Плюс точное время открытия письма жертвой, плюс IP-адрес, плюс еще много чего.

Грубо говоря, внедренная картинка в данном случае работает как классический "счетчик" наподобие того же SpyLog'a, собирающего множество информации, однако, имея собственный Web-сервер, можно выявить намного больше "интимных" деталей, раскрытие которых существенно упрощает атаку.

Начнем со спам-фильтров, обрабатывающие графические изображения (а, с учетом популярности графического спама, такие фильтры получают все большее и большее распространение). Очевидно, чтобы отделить зерна от плевел, фильтр должен загрузить картинку, следуя указанной ссылке. Если фильтр установлен непосредственно на почтовом сервере жертвы (или перед ним), хакерский WEB-сервер "поймает" GET-запрос _сразу_ же после отправки письма (или спустя короткое время с учетом очереди обработки писем) даже если жертва спит мертвым сном и не заботится о проверке почты. Как вариант, спам-фильтр может быть установлен на локальной машине (интегрирован в почтовый клиент), работающей на "автопилоте" (то есть остающейся включенной на ночь) и загружающей письма по POP3 протоколу каждые 3-5 минут. Может ли хакер различить две этих ситуации? Конечно же может! Внешний спам-фильтр имеет свой собственный IP, отличающийся от IP-адреса клиента, что выдает его с головой!

Поскольку, последнее время обнаружено огромное количество "дыр" в графических библиотеках, то антивирусы, установленные на почтовых серверах, осуществляют автоматический поиск вирусов в изображениях, формируя GET-запрос на хакерский сервер, причем, этот GET-запрос очень характерный. Для экономии трафика антивирусы (как правило) загружают только заголовок (где, собственно говоря, и находятся искаженные поля, приводящие к переполнению буфера). Остальное содержимое файла их не интересует, в результате чего мы получаем очень характерный GET-запрос, более того — уникальный для каждого типа антивируса. Поскольку, длина заголовка большинства графических файлов варьируется в очень широких пределах, размер запрашиваемого блока у всех антивирусов различен, что позволяет отличить NOD32 от KAV (например). Тоже самое относится к системам обнаружения вторжений и многим брандмауэрам, "скрещенных" с антивирусами. Мода пихать в один продукт кучу функционала хакерам только на руку.

Так же можно выявить и наличие Proxy-сервера, GET-запрос которого отличается от GET-запроса "чистого" почтового клиента. "Хорошие" (в хакерском смысле этого слова) Proxy-сервера даже сообщают внутренний IP-адрес жертвы. Красота!!! А если у жертвы на локальном компьютере установлены "баннерорезалки" (обычно работающие как локальные Proxy-серверы), они так же будут обнаружены, поскольку их GET-запросы отличается от GET-запросов почтовых клиентов.

Кэширующие Proxy-сервера распознаются вообще элементарно. Если мы отправили письмо по нескольким адресам, принадлежащим одной фирме, а картинка с хакерского сервера оказалась загружена лишь однажды, то, либо остальные письма не были открыты, либо же картинка была скэширована. Кстати, далеко не все сотрудники компании проверяют корреспонденцию со своего рабочего места. Многие из них используют альтернативные каналы выхода в Сеть, да и сама локальная сеть компании зачастую разбита на несколько независимых сегментов и хакер без труда выяснит что это за сегменты!!! Впрочем, на практике, обычно достаточно найти хотя бы одно слабое звено — сотрудника, проверяющего корпоративный ящик со своего домашнего компьютера (естественно, не защищенного). Зная его IP и версию операционной системы, хакер запросто забросит туда зловредную программу, заражающую другие программы (и сменные носители типа флеш-карт), в конечном счете проникающие на рабочую станцию, подключенную к корпоративной локальной сети.

Сбор статистики — еще одно хорошее подспорье для атаки. Отправляя серию писем и наблюдая время их открытия, хакер построит достаточно точный график работы жертвы, что

само по себе не является большим секретом, но в совокупности со всеми остальными данными дает богатую пищу к размышлениям.

Кстати, Outlook Express (и другие популярные почтовые клиенты/браузеры) по умолчанию загружают только по три картинки за раз, и следующие GET-запросы формируют только после получения ответа от WEB-сервера. Любое другое поведение указывает на то, что картинки загружаются не почтовым клиентом и не браузером, а каким-то программным комплексом, расположенным между ними, тип которого можно выявить по характеру и содержимому GET-запросов.

Кстати, о браузерах и WEB-почте. Просто поразительно какое количество информации браузеры передают в GET-запросе!!! Достаточно часто поле Referrer содержит все необходимые данные для несанкционированного входа в чужую текущую почтовую сессию — хакер может открыть почтовый ящик жертвы как свой собственный и делать все, что ему заблагорассудится (конечно, одновременная работа с двух разных IP, как правило, блокируется, но если жертва выходит из ящика не сделав Logout, то сессия еще будет удержится некоторое время, позволяя хакеру дорваться до ящика).

Выявить наличие трансляторов сетевых адресов несколько сложнее, но все-таки возможно. Для этого необходимо проследить за номерами локальных портов, характер назначения которых позволяет выявить не только наличие транслятора, но так же его тип и загруженность узла (т. е. сколько других TCP/IP соединений осуществляется за единицу времени). Впрочем, техника выявления NAT'ов — это отдельная большая тема.

А вот еще один интересный факт. Операционные системы семейства Windows имеют довольно слабый DNS stub (не путать с DNS-клиентом, который вообще может быть отключен). Генерируя довольно предсказуемые номера UDP-портов и идентификаторов (при активном DNS-клиенте, который в Windows 2000 и выше включен по умолчанию, UDP порт вообще постоянен), они допускают возможность "подмятия" DNS. Хакер генерирует подложный ответ, который тем не менее воспринимается системой как правильный и… последствия не заставляют себя ждать. Единственная проблема в том, что в отличие от DNS-серверов, рабочие станции не генерируют большое количество DNS-запросов в единицу времени и послать подложный ответ, опередив настоящий DNS, достаточно трудно. Если только… не напичкать письмо кучей ссылок на картинки! Тогда… а вот тогда… даже страшно сказать, что произойдет. Ну, если жертва выходит в Сеть со своего домашнего ПК, то атака провалиться даже не начавшись, поскольку штатные клиенты генерируют не более трех GET-запросов одновременно. И хакеру будет очень трудно успеть послать подложный запрос. А вот если это корпоративная сеть с кэширующим Proxy или анти-спам фильтром, то картинки загружаются общим скопом и вероятность того, что поддельный DNS-отклик будет воспринят как правильный значительно повышается (естественно, все картинки должны быть расположены на разных узлах).

Допустим, хакер подделал DNS-отклик. Что тогда? А то, что антивирус загрузит совсем не ту картинку, на которую указывает ссылка. И если исходная картинка содержит в себе вирус, антивирус его не поймает, зато его поймает жертва, открывающая письмо!

Обращаю внимание читателя, что во всех вышеописанных случаях ответа получателя на хакерское письмо не требуется!

Рисунок 1 внешний вид "жучка" и фрагмент HTML-кода с внешней ссылкой на картинку

>>> врезка дыры в обработчиках изображений

Листая Security Focus, просто не устаешь поражаться как много дыр содержится в обработчиках графических изображений. В настоящий момент уязвимы практически все форматы: wmf, bmp, pic, gif, png, jpg, swf… И хотя большинство ошибок уже исправлено, далеко не все пользователи установили заплатки, что делает их уязвимыми для удаленных атак.

Практика показывает, что локальные антивирусы (DrWeb, KAV, NOD32) обращают внимание лишь на вложения, но пропускают ссылки на внешние ресурсы. Антивирусы, установленные на "магистральных" каналах, проверяют и то, и другое, однако, исследование проведенное автором, показало, что таких антивирусов раз два и обчелся и потому ссылки на зловредные картинки работают просто превосходно, а вот вложения давятся локальными вирусами весьма эффективно.

>>> врезка почтовые ресурсы с жучками

Существует достаточно большое количество вполне легальных служб, предоставляющих услуги внедрения жучков в письма с целью определения точного времени открытия письма абонентом (не говоря уже о подтверждении факта успешной доставки). Как вы понимаете, эта техника радикальным образом отличается от стандартной процедуры запроса на получение — по умолчанию почтовые клиенты выводят диалоговое окно на экран, позволяя пользователю либо подтвердить получение письма, либо сделать вид, что оно до него не дошло.

Вот пара ссылок на подобные ресурсы: http://www.didtheyreadit.com/ и http://www.msgtag.com/home/ (между прочим, бесплатные и работающие по принципу proxy-сервера, то есть пересылающие письма со своего доменного имени).

от идеи до модели или лабораторная работа N1

Заканчиваем с теорией и вплотную переходим к практическим экспериментам. Мы будем использовать почтовые программы Outlook Express и The Bat. К сожалению, штатным образом ни одна из них не позволяет вставлять ссылки на картинки, без присоединения самой картинки к письму (что, естественно, не входит в наши планы). Отсутствие встроенного HTML-редактора препятствует элементарной правке полей, однако… не нужно быть гением, чтобы обойти все эти ограничения.

ОК, берем Outlook Express, создаем новое письмо в HTML-формате и вставляем гиперссылку в текст сообщения (именно гиперссылку, а не картинку!). Сохраняем его на диск через меню Файл/Сохранить как… и получаем вполне текстовой .eml, который можно править в текстовом редакторе:

```
From: "Kris Kaspersky" <kpnc@sendmail.ru>
To: <kpnc@aport.ru>
Subject: test 666
Date: Sat, 26 Apr 2008 03:00:58 +0400
MIME-Version: 1.0
X-Priority: 3
X-MSMail-Priority: Normal
X-Unsent: 1
X-MimeOLE: Produced By Microsoft MimeOLE V6.00.2800.1506

------=_NextPart_000_025C_01C8A749.C39D0D40
Content-Type: text/html;
        charset="koi8-r"
Content-Transfer-Encoding: quoted-printable

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">
<HTML><HEAD>
<META http-equiv=3DContent-Type content=3D"text/html; charset=3Dkoi8-r">
<META content=3D"MSHTML 6.00.2800.1515" name=3DGENERATOR>
<STYLE></STYLE>
</HEAD>
<BODY bgColor=3D#ffffff>
<A=20 href=3D"http://nezumi.org.ru/souriz/temp/love.gif">
http://nezumi.org.ru/souriz/temp/love.gif</A>
</FONT></DIV></BODY></HTML>

------=_NextPart_000_025C_01C8A749.C39D0D40--
```

Листинг 1 письмо, созданное Outlook Express, сохраненное на диск и открытое в текстовом редакторе (все email адреса в заголовке — недействительные)

Находим строку **<A=20href=3D"http://nezumi.org.ru/souriz/temp/love.gif"> http://nezumi.org.ru/souriz/temp/love.gif</A>** и переписываем ее следующим образом: **<IMG SRC=3D"http://nezumi.org.ru/souriz/temp/love.gif"/>**. Не забывайте про префикс "3D" иначе ничего не получится! Некоторые хакеры выставляют размеры картинки в 0x0 пикселей, чтобы она не была видна, но это не есть хорошо и слишком подозрительно. Лучше вставить обычную картинку, например, свой логотип или что-то еще.

Естественно, вместо адреса http://nezumi.org.ru/ вы должны указать что-то свое, воздвигнув собственный WEB-сервер (рекомендую бесплатный SMALL HTTP – smallsrv.com) с доменным именем, которое можно зарегистрировать даже имея в своем распоряжении всего лишь динамический IP с хлипким каналом (большая пропускная способность нам не понадобиться, мы же не спамом занимаемся).

Открываем модифицированный файл в Outlook Express'e (для этого достаточно дважды щелкнуть по нему мышкой в Проводнике) и отправляем его адресату. Смотрим на консоль нашего WEB-сервера, отображающую текущие запросы (или пытаем лог). Поскольку, в данном случае мы отсылаем файл самому себе, то в момент просмотра полученного письма, в логе сервера тут же появляется новая запись с нашим IP-адресом и прочими параметрами. Кстати, обратите внимание, что почтовые службы mail.ru и gmail.com _не_ осуществляют проверку внешних картинок ни при отправке, ни при приеме письма. А еще говорят, что они защищают нас от спама!!! Выходит, что вся защита сводится к ведению black-листов IP-адресов (ну и, может быть, фильтрации текстового содержимого). Если же картинка загружается с внешнего ресурса, спамерам достаточно менять IP-адреса (для чего идеально подходит boot-net, то есть подконтрольная хакеру сеть машин с внедренным back-door'ом) и спам свободно пройдет сквозь фильтры, правда, нагрузка на наш WEB-сервер существенно возрастет, но исходящий трафик обычно либо очень дешев или вообще бесплатен, а объем входящих GET-запросов не так уж и велик.

Ладно, а как осуществить тоже самое при помощи The Bat? В меню "Tools" выбираем Import messages/From .MSG/.EML-flies, и наше письмо появляется в папке входящих, откуда мы копируем его в исходящие и говорим "Send Queued Mail".

Как и в предыдущем случае, смотрим в лог WEB-сервера и делаем соответствующие выводы.

практический пример атаки

Рассмотрим конкретный пример использования жучков для реконструкции топологии сети и получения прочих "интимных" данных. Это, конечно, не атака, а только ее начальная фаза. Ничего нелегального мы совершать не собираемся. Все _абсолютно_ законно. Кстати, аналогичная техника используется корпорацией Intel в ее почтовой рассылке — все картинки там представлены в виде ссылок на внешний HTTP-сервер. Возможно, это сделало с целью уменьшения объема рассылаемых писем, возможно, для слежения за подписчиками. Кто знает? Но никто же не пытается засудить Intel, поскольку, никакого состава преступления в ее действиях нет. Вот так же и с нами.

Объектом исследований выступит популярная антивирусная служба www.virustotal.com, внизу главной страницы которой мы видим адрес info@virustotal.com. Вот на него мы и отправим письмо, созданное по вышеописанной методике, только вместо <kpnc@aport.ru> поставим <info@virustotal.com>. Обратный адрес может быть любым, это не суть важно. Главное, чтобы он не был в black-листе и сервер-отправитель не относился к категории заблокированных. Лучше всего использовать свой собственный SMTP-сервер, кстати говоря, в состав SMALL HTTP сервера входит замечательный SMTP!

Через несколько секунд после отправки письма с жучком мы ловим первый GET-запрос (см. листинг 2).

```
!->25/04 13:34:43 [81.26.151.146:2785>80] (t1 547)
GET /souriz/temp/love.gif HTTP/1.0
User-Agent: Mozilla/4.0 (compatible; Lotus-Notes/6.0; Windows-NT)
Accept-Language: ru
Host: nezumi.org.ru
Accept: text/html
Accept: text/x-html
Accept: application/html
Accept: application/x-html
Accept: text/plain
Accept: image/gif
Accept: image/jpeg
Accept: multipart/*
Accept: application/x-x509-user-cert
Accept: application/x-x509-ca-cert
Accept: */*

!->25/04 13:34:43 [81.26.151.146:2785>80] (t1 548)
!->HTTP  in:408 out:8735 /souriz/ Time:130
```

Листинг 2 первый GET-запрос, пойманный нашим WEB-сервером

А спустя короткое время — (приблизительно через две минуты) наш WEB-сервер ловит еще один GET-запрос (см. листинг 3):

```
!->25/04 13:36:55 [85.62.90.20:54329>80] (t1 549)
```

```
GET /souriz/temp/love.gif HTTP/1.0
Host: nezumi.org.ru
Accept-Language: es
Accept: */*
User-Agent: Mozilla/4.0 (compatible; MSIE.6.0; Windows NT 5.1)
Via: 1.1 noname:8888 (squid/2.6.STABLE17)
Cache-Control: max-age=259200
Connection: keep-alive

!->25/04 13:36:55 [85.62.90.20:54329>80] (t1 550)
!->HTTP  in:255 out:8735 /souriz/ Time:10
```

Листинг 3 второй GET-запрос, пойманный нашим WEB-сервером

Попробуем разобраться в этом хозяйстве, попутно отметив, что IP-адрес самого virustotal'a равен 74.53.201.162, а так же приведем несколько вполне типичных GET-запросов от известных агентов (см. листинги 4 — 6). Первый запрос (см. листинг 2) совершенно нетипичен ни для браузеров, ни для почтовых клиентов, ни для Proxy-серверов, что наводит на мысль — а не является ли это спам-фильтром? Поскольку, запрашивается вся картинка целиком (размер out совпадает до последнего байта), то это скорее именно спам-фильтр, а не антивирус.

Второй GET-запрос (см. листинг 3) — типичный кэширующий proxy (сравните его с листингом 6). Кстати, первый запрос не мог принадлежать кэширующему Proxy, иначе бы второй запрос уже бы не появился. Не мог быть первый запрос и просто Proxy-сервером, ибо разница во времени (~2 минуты) слишком велика для "проксирования", но вполне типична для спам-фильтра/антивируса на довольно загруженном узле.

Оба GET-запроса достаточно характеры и позволяют выявить версию программного обеспечения, для чего достаточно запустить какой-нибудь сетевой сканер, например, X-Spider, сообщающий нам, что узел 85.62.90.20 не отвечает на запросы, а 81.26.151.146 работает под управлением UNIX'a, имеет кучу открытых портов (53, 80, 123, 443, 993, 2222) и содержит уязвимость в OpenSSH, приводящую к возможности выполнения произвольного кода (см. рис. 2). Что ж, неплохое начало!

Рисунок 2 данные сканирования узла 81.26.151.146

```
!->25/04 11:05:02 [83.239.33.46:3338>80] (t1 9199)
GET /souriz/temp/love.gif HTTP/1.1
Host: nezumi.org.ru
User-Agent: Mozilla/5.0 (Windows; Windows NT 5.0; en-US; rv:1.8) Gecko Firefox/1.5
Accept: text/xml,application/xml,application/xhtml+xml,text/html; image/png,*/*
Accept-Language: en-us,en;q=0.5
Accept-Encoding: gzip, deflate
Accept-Charset: ISO-8859-1,utf-8;q=0.7,*;q=0.7
Keep-Alive: 300
Connection: keep-alive
```

Листинг 4 GET-запрос FireFox'a

```
!->25/04 11:30:17 [83.239.33.46:3415>80] (t1 174)
GET /souriz/temp/love.gif HTTP/1.1
Accept: */*
Accept-Language: ru,ja;q=0.5
Accept-Encoding: gzip, deflate
If-Modified-Since: Mon, 21 Nov 2005 18:06:17 GMT
User-Agent: Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.0)
Host: nezumi.org.ru
Connection: Keep-Alive
```

Листинг 5 Outlook Express

```
!-<25/04 12:40:53 [74.53.201.162:80<4008] (t1 388) <HTTP/1.0 200 OK
Date: Fri, 25 Apr 2008 08:16:15 GMT
Server: Apache
Last-Modified: Tue, 19 Jun 2007 11:32:26 GMT
ETag: "519b2-15df-43340aaf5c680"
Accept-Ranges: bytes
Content-Length: 5599
Cache-Control: max-age=604800
Expires: Fri, 02 May 2008 08:16
```

```
!->25/04 12:40:54 [127.0.0.1:4005>3127] (t1 389) >Proxy in:6049 out:333
```

Листинг 6 Proxy-сервер (SMALL-HTTP)

Теперь попробуем выяснить, кому какие IP-адреса принадлежат и как они распределены в пространстве на поверхности планеты Земля. В этом нам поможет одна из множества on-line служб, например, http://www.leader.ru/. Открыв ее в браузере, видим в правом верхнем углу окошко whois, вводим туда "www.virustotal.com" и получаем следующую информацию (см. листинг 7):

```
Hostname:              www.virustotal.com
IP:                    74.53.201.162
Reverse name:          viruskill2.hispasec.com
Preferable MX:         mail.hispasec.com

Owner:                 ThePlanet.com Internet Services, Inc.
Location:              Capitol, City: Houston, PostalCode: 77002, Country: US
Contact Information:   74.52.0.0 - 74.55.255.255
CIDR:                  74.52.0.0/14,
NetType:               Direct Allocation,
NameServer             NS1.THEPLANET.COM, NameServer: NS2.THEPLANET.COM,

Domain Information:    VIRUSTOTAL.COM
Owner:                 Hispasec Sistemas
Location:              Edificio Bic Euronova, Parque Tecnologico Andalucia, ES
Contact Information:   Hispasec Sistemas bernardo@hispasec.com, Network Solutions, LLC.
Name Servers:          DNS.HISPASEC.COM, DNS.HISPASECSISTEMAS.COM
```

Листинг 7 информация об имени www.virustotal.com

Теперь испытаем IP-адрес 85.62.90.20 (второй GET-запрос нашего "жучка"), пойманным Web-сервером (см. листинг 8):

```
Hostname:              Cannot be resolved
IP:                    85.62.90.20
Preferable MX:         inc.wanadoo.es

Network Information:   UNI2-NET
Network Range:         85.62.0.0 - 85.62.255.255
Owner:                 Addresses IP for corporate ABI clients, France Telecom Espaca
Location:              Parque Empresarial La Finca, Edificio 9,  Madrid, Spain
```

Листинг 8 информация об узле 85.62.90.20

Ага! Уже есть кое-что интересное! Адрес 85.62.90.20 никак не связан с узлом www.virustotal.com и входит в совершенно другую подсеть, впрочем, так же принадлежащую испанскому провайдеру. Учитывая тот факт, что 85.62.90.20 не отвечает на запросы, не имеет доменного имени и в качестве почтового сервера использует внешнюю почтовую службу inc.wanadoo.es, мы с 99% вероятностью можем предположить, что это Proxy-сервер (squid/2.6.STABLE17) организации, обслуживающей www.virustotal.com.

А что на счет адреса 81.26.151.146? И вот тут нас ждет неожиданный, но очень приятный сюрприз (см. листинг 9):

```
Hostname:              msk-gw.drweb.com
IP:                    81.26.151.146
Preferable MX:         mx.drweb.com

Network Information:   NAUKANET
Network Range:         81.26.151.0 - 81.26.151.255
Owner:                 Moscow
Location:              OOO Nauka-Svyaz,3ya ulitsa Yamskogo Polia,125124 Moscow,Russia
Contact Information:   Vadim Vakhrushin, noc@naukanet.ru,
                       +7 495 5029092, +7 495 9373412, Vladimir Schedrin,
                       noc@naukanet.ru, +7 495 5029092, +7 495 9373412
```

Листинг 9 информация об узле 81.26.151.146

Выходит, что сотрудники virustotal'a используют Dr.Web в качестве "магистрального" почтового сервера, точнее, его использует их провайдер, чем и объясняется столь длительное время проверки такого короткого письма (узлы провайдера обычно довольно сильно загружены). Как уже говорилось выше, данный узел содержит уязвимость, приводящую к возможности выполнения удаленного кода, а, значит, мы можем захватить контроль не только

над перепиской компании-разработчика virustotal'a, но и многих других фирм. И все это мы выяснили всего за несколько минут, обладая минимальными познаниями и хакерскими навыками!!!

Рисунок 3 кто есть кто в Интернете

заключение

Представляют ли "жучки" серьезную угрозу для безопасности сети? Трудно дать однозначный ответ. И да, и нет. С одной стороны, "жучки" позволяют хакеру добывать "интимную" информацию, действуя совершенно легальным и не привлекающим к себе внимания путем, но… если система не содержит дыр (все заплатки установлены, используется надежное программное обеспечение и т. д.), вся эта информация полностью бесполезна. Хакер либо откажется от атаки, либо будет искать другие пути.

Если же вы все-таки хотите заблокировать жучков, достаточно установить пакетный фильтр, анализирующий поступающую корреспонденцию и "выкусывающий" все ссылки на внешние картинки.

>>> врезка какую информацию может собрать img-жучок

- подтвердить открытие письма жертвой;
- установить точное время открытия письма;
- определить наличие Proxy-сервера у жертвы;
- выявить наличие трансляторов сетевых адресов;
- определить наличие кэширующего Proxy-сервера у провайдера жертвы;
- определить используемое программное обеспечение с точностью до версии;
- определить наличие спам-фильтров и/или антивирусов (а иногда и их версию);
- определить IP-адрес жертвы (чаще — внешний, реже — внутренний и внешний);
- выявить "банерорезалки", аннонимайзеры, контентные фильтры и др. локальные утилиты;