Hacker Highschool

SECURITY AWARENESS FOR TEENS



УРОК 4
ИГРАЯ С ДЕМОНАМИ

Hacker Highschool
SECURITY AWARENESS FOR TEENS

ISECOM

WWW.ISECOM.ORG - WWW.OSSTMM.ORG - WWW.HACKERHIGHSCHOOL.ORG - WWW.BADPEOPLEPROJECT.ORG - WWW.OSSTMMTRAINING.ORG

ВНИМАНИЕ

Проект Hacker Highschool является средством обучения и, как в любом обучающем средстве, существует опасность. Некоторые уроки, если ими злоупотреблять, могут привести к физической травме. Также дополнительные опасности могут быть там, где ещё недостаточно исследований о возможных последствиях излучений от специфической техники. Студенты, использующие эти уроки, находятся под контролем преподавателя, и, в тоже время, должны быть мотивированы на изучение материалов и непрерывную практику. ISECOM не несёт ответственности за применение информации, полученной из данных материалов и за дальнейшие последствия.

Все представленные здесь материалы являются открытыми и общедоступными в соответствии с положениями и условиями организации ISECOM:

Все материалы проекта Hacker Highschool предназначены для некоммерческого использования в работе с учениками средних государственных или частных школ, техникумов, студентами высших учебных заведений, слушателями младших курсов Hacker Highschool и учащимися на дому. Эти материалы в любой форме не могут быть использованы для продажи. Обучение по этим материалам в обучающей организации, техникумах, университетах, профессионально-технических заведениях, летних или компьютерных лагерях и других организациях, в которых взимается плата за обучение, категорически запрещено без приобретения лицензии. Для более подробного ознакомления с условиями использования либо приобретения лицензии для коммерческого использования материалов, посетите раздел сайта предназначенный для Лицензирования http://www.hackerhighschool.org/licensing.html.

Проект HHS является результатом труда открытого сообщества и, если Вы находите наши труды ценными и полезными, мы просим Вас поддержать нас путём приобретения лицензии, пожертвований, либо спонсорства.

Содержание

Сотрудники журнала

Pete Herzog, ISECOM
Glenn Norman, ISECOM
Marta Barceló, ISECOM
Chuck Truett, ISECOM
Kim Truett, ISECOM
Marco Ivaldi, ISECOM
Bob Monroe, ISECOM
Jaume Abella, ISECOM
Greg Playle, ISECOM
Simone Onofri, ISECOM
Guiomar Corral, Barcelona
Ashar Iqbal

Переводчики

Vadim Chakryan, Kharkiv National University of Radio Electronics
Dmitriy Pichuev, Ukrainian Engineering Pedagogical Academy
Andrii Sezko, Kharkiv National University of Radio Electronics
Olena Boiko, Kharkiv National University of Radio Electronics

ISECOM

Введение

В мире существуют тысячи языков, на которых разговаривают люди, а некоторые из них имеют десятки диалектов. Вы сами можете знать несколько языков, но шансы на то, что, путешествуя по миру, Вы сможете говорить с каждым, кого встретите, близки к нулю.

Да, Вы можете возразить тем, что математика — это универсальный язык или что язык музыки понимает каждый, но будем реалистами. Попробуйте заказать стакан содовой с лимоном и шарик мороженого, пользуясь «универсальными языками», и посмотрим, что у Вас из этого получится.

Если Вам посчастливилось побывать в стране, язык которой Вы не знаете, то попробуйте заказать там лимонад, используя волынку или саксофон. Запишите видео этого действа и отправьте его в ISECOM. Мы действительно хотим на это посмотреть! Вряд ли мы захотим это услышать, но увидеть — точно хотим.

Но ежедневно миллионы людей общаются друг с другом по Интернету, используя один общий язык. Не все люди могут говорить на одном языке; однако наши компьютеры и сети могут так делать.

> Модель, которая используется в современных сетях, называется клиент-серверной моделью. Физические компьютеры (хосты или серверы) предлагают службы (в UNIX они называются демонами **(daemons)**). Вспомните, как работает веб-сервер: он передает веб-страницу тогда, когда Вы её запросите. Ничего таинственного здесь нет.

Но, на самом деле, не Вы запрашиваете страницу. Вместо Вас это делает веб-браузер, то есть он выполняет роль клиента (или более формально — эту роль выполняет Ваш компьютер). В то же время Ваш компьютер также может быть и сервером. В этом вся прелесть сетей: Вы делаете что-то для меня; я что-то делаю для Вас.

Умножьте эту модель миллион раз и Вы получите Интернет. Представьте: миллионы компьютеров предлагают различные сервисы и услуги. Что нужно для того, чтобы быть клиентом? И возможно ли это ниспровергнуть? (Поищите значение этого слова, если Вы не уверенны, что точно знаете его. В конце концов, это хакерский курс.)

Готовы Вы или нет, давайте подробнее рассмотрим эти вопросы.

Службы

У Вас есть компьютер, и Вы знаете, что на нём есть полезная информация, или Вы можете участвовать в общей галлюцинации, создавая вид, что у Вас нет ничего, что имеет цифровую ценность. Также Вы знаете, что у других людей, миллионов других людей, есть компьютеры, на которых может быть полезная информация, не говоря о ресурсах, вроде процессоров, оперативной памяти, дисковом пространстве и пропускной способности.

Теперь представьте, что эти люди и их компьютеры с большой вероятностью имеют информацию, представляющую для кого-то интерес. Единственный вопрос — как получить эту информацию.

Компьютеры общаются друг с другом через порты, используя протоколы, о которых было рассказано в Уроке 3, но это не позволит Вам читать потоки двоичных данных, которыми обмениваются компьютеры (если только у Вас есть лишнее время). Нужен способ, при помощи которого можно получить данные, их интерпретировать и представить их в той или иной форме, которую Вы можете использовать.

Компьютеры передают данные посредством сетевых служб или просто служб. Эти службы позволяют Вам просматривать веб-страницы, обмениваться письмами, общаться в чате и взаимодействовать с удалёнными компьютерами. Эти службы сопоставлены с номерами портов.

Ваш компьютер, локальный компьютер, использует программы, называемые клиентами, для интерпретации информации, которую Вы получаете. Вы можете получить информацию от сервера (который предоставляет службу/запускает демон), через **Tor** сеть, от **Torrent** сидов или по сети **peer-to-peer**.

> Конечно, Ваш компьютер также может предоставлять услуги другим удалённым компьютерам, таким образом выполняя роль сервера данных или поставщика услуг. Если на Вашем компьютере появились вредоносные программы, то Вы, возможно, оказываете довольно много услуг, о которых сами не знаете.

Примеры клиентов — веб-браузеры (далее браузеры, — прим. перевод.), почтовые клиенты, программы для чата, Skype, Tor-клиенты, Torrent-клиенты, RSS и так далее. Эти приложения находятся на уровне приложений стека протоколов TCP/IP. На этом уровне все данные, переданные, инкапсулированные, зашифрованные, расшифрованные, направленные и т. д. нижними уровнями, конвертируются в нечто, что Вы, как пользователь, можете прочесть и понять.

HTTP и Сеть

Когда мы говорим про «Интернет», большинство людей подразумевают Всемирную Паутину **(World Wide Web)**. Всемирная Паутина, или просто Сеть — это не Интернет, это лишь малая часть доступных служб. Обычно она подразумевает просто просмотр веб-страниц через браузер.

Кстати, настоящий Интернет состоит из всех компьютеров, маршрутизаторов, проводов, кабельных и беспроводных сетей, которые перемещают данные различного рода. Сетевой трафик представляет собой только часть всего этого.

Сеть использует **HTTP (HyperText Transfer Protocol,** протокол передачи гипертекста**)** и приложения (клиенты), называемые браузерами, для доступа к документам на веб**-**серверах (далее серверах — прим. перевод.*)*. Информация от удалённого компьютера направляется на Ваш компьютер по протоколу HTTP, используя обычно 80-й порт. Ваш браузер интерпретирует и показывает Вам обработанную информацию.

Не все браузеры одинаково работают. Каждый предлагает разные инструменты и отображает HTML-контент немного (или очень) по-разному. Вопросы безопасности и конфиденциальности могут быть решены успешно, но в разной степени. Это означает, что Вы должны знать, что Ваш браузер может и не может делать, какие настройки и плагины дадут Вам идеальный баланс безопасности и конфиденциальности (если Вы не любите вредоносные программы, рекламу, спам и Ваших соседей, которые знают, что Вы любите смотреть видео про котят, играющих в зеленом желе).

Гипертекстовая часть протокола HTTP относится к текстам, читаемым нелинейным способом. Обычно Вы читаете линейно (т. е. последовательно): сначала страница 1, потом страница 2; сначала глава 1, потом глава 2; сначала урок 1, потом урок 2, и так далее. Гипертекст позволяет просматривать информацию в нелинейном порядке. По мере изучения чего-либо Вы можете переходить с одной главы на другую, просмотреть их повторно или перейти на другую тему прежде, чем закончить основную статью. Вот в чем разница между гипертекстом и простым текстом.

В гипертексте слова и идеи связываются не только со словами, которые непосредственно окружают их, но и с другими словами, изображениями, видео и музыкой. Гипертекст используется не только в Сети. Большинство полнофункциональных текстовых процессоров позволяют создавать локально хранимые страницы в веб, или HTML, формате. Вы читаете эти страницы в браузере так же, как обычные веб-страницы, только они хранятся на Вашем локальном, а не на удалённом компьютере.

Создать свою веб-страницу достаточно просто. Самый лёгкий способ — это использовать один из популярных текстовых процессоров, вроде OpenOffice/LibreOffice Writer, Microsoft Word или WordPerfect. Эти программы позволяют Вам создавать простые веб-страницы, сочетая текст, гипертекст и изображения. С их помощью многие люди сделали достаточно функциональные веб-страницы (можно даже использовать такие простые текстовые редакторы, как vi, который установлен на большинстве Unix-систем). Среди других текстовых редакторов можно выделить Microsoft Notepad, Notepad++, SciTe, emacs и т. д.

Но эти страницы ничем не примечательны. Эту проблему можно решить, используя **CSS,** скрипты и анимацию. Вы можете потратить много денег на приложения для дизайна причудливых веб-страниц. Эти приложения позволяют создавать интересные эффекты на веб-странице, но они более сложны в использовании. Тем не менее, они обычно делают работу в целом легче. Более дешевая альтернатива — это взять один из текстовых

редакторов, ориентированных для работы с HTML и скриптовыми языками, изучить синтаксис HTML, скриптов и написать собственные веб-страницы с нуля.

Едва Вы закончите с дизайном страниц, Вам нужно будет где-то выложить их, в том случае, если хотите, чтобы страницы увидели другие люди. Интернет-провайдеры **(Internet Service Providers, ISPs)** предоставляют услугу веб-хостинга на собственных веб-серверах.

Вы можете запустить сервер на своём домашнем ПК, но в этом случае могут возникнуть некоторые проблемы. Информация, хранимая на сервере, доступна только тогда, когда сервер включен, правильно функционирует и имеет открытое подключение. Поэтому, если Вы хотите запустить сервер из своей спальни, Вы должны держать компьютер включенным все время; Вы должны быть уверены, что программа на сервере корректно работает (сюда входит поиск и устранение неисправностей комплектующих, контроль вирусов, червей и других атак, обработка ошибок и изъянов внутри самой программы); и открытое соединение с Интернетом должно быть стабильным и быстрым. Интернет-провайдеры взимают дополнительную плату за высокую скорость загрузки и фиксированный (статический) IP-адрес, поэтому большинство людей платит за всю работу третьим лицам.

Компании, предоставляющие веб-хостинг (далее хостинг — прим. перевод.), хранят Вашу информацию на своих компьютерах. И это отлично, ведь атакованы будут их серверы, а не Ваши. Хорошие компании по веб-хостингу имеют множество резервных серверов и соблюдают политику регулярного резервирования данных, таким образом Ваш сайт не исчезнет без вести только из-за проблем с «железом»; техническая поддержка держит серверы запущенными, несмотря на атаки и ошибки в программах; а ряд открытых подключений к Интернету даёт небольшую гарантию от простоев, перебоев и остановок в работе. Поэтому всё, что от Вас требуется, — это оформить Вашу веб-страницу, загрузить её на сервер хостинга, выключить ПК и идти спать. Ваша веб-страница будет доступна всему миру, пока Вы будете платить по счетам.

Также можно найти организации, предлагающие бесплатный хостинг. Некоторые из них финансируются платной рекламой, то есть любой, кто захочет посмотреть Вашу веб-страницу, сперва увидит чью-ту рекламу. Но им не придётся ничего покупать, а Вам не придётся ничего платить.

Упражнения

4.1 Веб-страница — это просто текст, который сообщает браузеру, где должны размещаться изображения, видео и другие элементы. Вы можете увидеть этот текст, просмотрев исходный код страницы. Запустите свой любимый браузер, перейдите на ISECOM.ORG и загрузите страницу. Теперь посмотрите исходный код. Вы увидите несколько тегов со слово "meta" в них. Например, meta-charset="utf-8". Что это значит? На что это указывает?

4.2 Найдите еще 3 мета-тега и объясните, на что они указывают. Возможно, ответ на этот вопрос Вам придётся поискать в Интернете, так что тщательно продумайте, какие ключевые слова Вы будете использовать при поиске, чтобы быть уверенным, что Вы найдёте правильные ответы.

4.3 Сохраните исходный код ISECOM.ORG себе на ПК. Откройте его в браузере. Что изменилось? Как Вы думаете, чем вызваны изменения?

4.4 Откройте исходный код ISECOM.ORG в текстовом редакторе и Вы увидите, что это всего лишь слова и цифры. Всё, что Вы измените или добавите в этот файл, после сохранения повлияет на вид страницы в браузере. Удалите строки и Вы заметите удаление каких-то элементов. Измените слова и они отобразятся изменёнными. Теперь уберите всё лишнее со страницы и добавьте своё имя так,

чтобы оно выделялось среди других слов (шрифт полужирный и бОльший). Попробуйте. Сохраните. Откройте в браузере, посмотрите, добились ли Вы успеха. Нет? Тогда продолжайте пробовать!

Смотрите Пища для ума**:** Играем с ***HTTP*** в конце этого урока*,*
если Вы хотите копнуть глубже.

Email — SMTP, POP и IMAP

Второй заметный аспект Интернета — это, вероятно, электронная почта. У себя на ПК Вы пользуетесь почтовым клиентом, который соединяется с почтовым сервером. Когда Вы создаете почтовый аккаунт, Вы получаете уникальное имя в виде пользователь@домен, и Вам нужно создать для этого аккаунта пароль.

Существуют 2 типа серверов для работы с электронной почтой: **SMTP (Simple Mail Transfer Protocol,** простой протокол передачи почты**)**, который отправляет почту, и почтовый сервер, который извлекает Вашу почту (используется **POP (Post Office Protocol,** протокол почтового отделения**)** или **IMAP (Internet Message Access Protocol,** протокол доступа к Интернет**-**сообщениям**)**).

Протокол SMTP (напомним ещё раз) используется для отправки электронной почты. SMTP определяет поля в электронном письме, включая поля FROM, TO, SUBJECT, CC и BODY. Старый добрый SMTP не требует пароля и отправляет всё в открытом виде; каждый может прочитать Вашу почту. Возможно, это был неплохой вариант, когда протокол был только разработан, а Интернет был небольшим миром, населённым единомышленниками. Но он оставил лазейку, которая позволяла любому пользователю рассылать спам и делать другие гадости, вроде подмены электронной почты **(email spoofing)**, которая, по существу, означает подмену (spoofing) адреса отправителя. Почти все современные почтовые серверы используют Secure SMTP; это значит, что Вы должны подтвердить свою личность прежде, чем отправлять письма.

В следующих уроках мы покажем Вам, как работает подмена и как обнаружить её в заголовках электронной почты. Эта горсть знаний неимоверно быстро может превратить Вас из «неопытной овечки» в «уверенного волка».

POP3 (Post Office Protocol, версия **3)** — это «хранящий и складывающий» протокол. Почтовый сервер получает Ваши письма и хранит их для Вас, пока Вы не соединитесь и не скачаете (сбросите) Вашу почту. Отправка же почты происходит с использованием SMTP. Это хороший подход к работе с электронной почтой в том случае, если у Вас dial-up соединение, поскольку оно занимает меньше времени для отправки и получения электронной почты, а Вы можете читать электронную почту даже не имея подключения к Интернету.

IMAP, с другой стороны, по умолчанию хранит Вашу почту на сервере. Множество корпоративных почтовых решений используют какой-либо вариант IMAP в зависимости от поставщиков программного обеспечения. В IMAP Вы можете создавать папки в своих почтовых ящиках и перемещать письма между ними. Когда Вы соединяетесь с IMAP-сервером, Ваши почтовые ящики и сервер синхронизируют папки, их содержимое, входящую и удалённую почту. Это веское преимущество: Вы можете получить всю свою почту с любого

устройство: ноутбука, телефона или планшета. К тому же, Вы можете скачать и хранить почту у себя на ПК.

Однако у этого протокола есть и два недостатка: во-первых, очевидно, что Вам нужно обмениваться бОльшим объёмом информации, поэтому необходимо более быстрое соединение и большее количество времени. Во-вторых — ограниченный объём. Ваш почтовый сервер назначит размер Вашего почтового ящика, который нельзя превышать. Если Ваш ящик будет полон, Вы не сможете принимать почту, пока не удалите другие письма (или не купите больше места). В конечном счёте это означает, что корпоративная IMAP-почта требует управления данными. Вы должны перемещать почту в локальные хранилища и регулярно очищать отправленную почту, спам и удалённые письма в целях экономии места. Письма с вложениями «уничтожат» Вас. В наше время, когда есть возможность создать бесплатную учётную запись с огромным бесплатным хранилищем данных, все эти предосторожности могут показаться глупыми. Пока Вы не получите иск. Или кто-то не попытается взломать почтовый сервер и получить ВСЮ Вашу почту..

POP и IMAP серверы требуют пароль для получения доступа к учётной записи. Но оба протокола отправляют абсолютно всё в открытом виде, в т.ч. и пароли, поэтому, теоретически, каждый может их прочитать. Вы должны использовать шифрование для маскировки процесса входа (например, SSL) и содержимого письма. Вот почему у многих почтовых клиентов есть флажок Использовать *SSL*.

Когда Вы в почтовом клиенте нажимаете кнопку Отправить, происходит 2 вещи: сперва Ваш клиент заставляет Вас войти на SMTP-сервер (даже если Вы уже вошли на POP-сервер, чёрт побери!), а затем отправляет саму почту (через SMTP-протокол).

Такая система «надоела» к середине 1990-х, когда серверы начали использовать протокол, называемый **POP-before-SMTP**: сначала Вы отправляете POP-серверу Ваше имя пользователя и пароль, затем загружаете входящую почту, а потом SMTP-сервер проверяет Вас по POP-серверу («С этим парнем всё ОК?» «Да, я аутентифицировал его.») и отправляет Ваши письма. Это хорошая экономия времени.

Стоит помнить одну важную вещь: несмотря на использование пароля для защиты, электронная почта — не вариант для отправки важной/засекреченной информации. Большинство POP-клиентов и серверов требуют, чтобы Ваш пароль был передан — незашифрованным — на почтовый сервер. Это не значит, что любой, кто получает письмо от Вас, также получает Ваш пароль; но это значит, что кто-то с нужными знаниями и инструментами может узнать Ваш пароль — и, как следствие, содержимое писем. (Касательно идей по усилению защиты Ваших писем, см. Урок **9:** Безопасность электронных писем.)

Упражнения

4.5 Отправьте себе письмо с Вашей главной учётной записи на Вашу главную учётную запись. Отправьте то же письмо на ту же учётную запись из другой учётной записи. Насколько долго шли 2 письма? Есть ли между ними отличия и почему?

4.6 Просмотрите одно письмо из того миллиона спама, который засоряет Ваш ящик. Можете ли Вы определить, кто действительно шлёт Вам отдельный спам? Есть ли в них спрятанная информация? Если есть, как хакер может это увидеть?

4.7 Можете ли Вы задержать отправку электронной почты до определённого времени или дня? Придумайте, как можно использовать эту особенность, чтобы пошутить над своими друзьями?

IRC

IRC (Internet Relay Chat) — прекрасное место, чтобы увидеть всю прелесть неконтролируемости Интернета в лучшем виде. Или в худшем. В IRC любой участник, у которого есть что сказать, получает возможность это высказать. IRC также известен как **Usenet** или группы новостей. Каждая группа новостей имеет своё название, под-имя, под-под-имя и так далее.

Вполне вероятно, Вы уже знакомы с чатами. IRC — это как чат, только без правил сетевого этикета, и достаточно часто без модераторов. В канале IRC Вы можете найти именно то, что ищете, или то, о чём никогда не знали.

> Все правила, которые Вы слышали о чатах, применимы к каналам IRC. Никому не говорите Ваше настоящее имя. Не давайте номер Вашего мобильного телефона, адрес проживания или номера кредиток. Но получайте удовольствие! В то же время будьте осторожны с доступным контентом. Не всё в Интернете невредоносно и не все люди в Интернете хорошие.

IRC не безопасен, и всё, что Вы пишете, передаётся открытым текстом от одного IRC-сервера к другому. Вы можете создавать закрытые разговоры с другими участниками IRC, но сообщения будут передаваться также в открытом виде. Использование ника (nickname) обеспечит Вам лишь незначительную конфиденциальность. Если Вы планируете проведение каких-либо вредоносных или сомнительных действий, не используйте один и тот же ник для всех учётных записей. Используя один ник, Вы даёте отличный способ себя выследить полиции. Или менее приятным личностям.

Темы называются «каналами». Поскольку в мире существуют тысячи каналов, мы даём Вам URL, в котором перечислены многие из них для дальнейшего Вашего исследования до потери Вами рассудка:

```
http://www.nic.funet.fi/~irc/channels.html
```

Если у Вас возникли вопросы по сообщениям, написанным другим участником, Вы можете сообщить о них модератору (если он есть), либо «пнуть» (**bump**) участника из этого канала. Если Вы не желаете кого-либо слышать, Вы всегда можете поместить участника в «черный список», а его сообщения не будут Вам видны. Может быть эта тема и так Вам не подходит.

Упражнения

4.8 Найдите 3 IRC-канала, где обсуждаются вопросы безопасности. Как Вы присоединились к общей дискуссии? Что Вам надо сделать, чтобы создать закрытое (личное) общение с кем-то из участников?

4.9 Какой порт использует IRC?

4.10 В IRC возможен обмен файлами. Как это можно осуществить? Хотели бы Вы обмениваться файлами в IRC?

4.11 Каковы главные отличия MIME и SMIME? Когда Вы видите "S" в аббревиатуре, означает ли это что-либо для Вас как Безопасно (намёк) мыслящей личности?

FTP

Старый добрый **FTP (File Transfer Protocol,** протокол передачи файлов**)** обычно запущен на 20 и 21 портах. Угадайте для чего? Это позволяет Вам обмениваться файлами между двумя компьютерами. В то время, как протокол может быть использован для частных передач, т. к. в нём не использует шифрование, он более широко используется для бесплатных, анонимных FTP-серверов, которые предлагают свободный доступ к коллекциям файлов, вроде ISO для новых крутых дистрибутивов Linux.

Анонимный FTP когда-то был единственным вариантом для компьютерных пользователей обмениваться файлами по Интернету. В то время как существует множество анонимных FTP-серверов для нелегального распространения файлов (что является отличным методом распространения «цифровых заболеваний» (binary disease)), ещё больше серверов используют для легального распространения файлов и программ. Пользуясь поисковой системой, можно найти серверы, которые предлагают анонимные FTP-сервисы. Но помните: FTP-логины передаются в открытом виде. Да, даже имя пользователя и пароль. Существует Secure FTP (SFTP), но повсеместно он не используется.

Большинство анонимных FTP-серверов позволяют получить доступ к файлам с помощью протокола FTP через браузер. Есть также некоторые действительно хорошие FTP-клиенты, которые работают как программы управления файлами. Как только Вы вошли на FTP-сервер, Вы можете перемещать файлы к себе на компьютер так же, как Вы локально перемещаете файлы на своем ПК. Разве что для FTP требуется немного больше времени для загрузки каждого файла на Ваш компьютер, в основном потому что сервер FTP может быть расположен на другой стороне планеты.

Упражнения

4.12 Windows, OSX и Linux поставляются с базовым консольным FTP-клиентом; для получения доступа к нему откройте командную строку или терминал и введите:

```
ftp
```

На вкладке `ftp>`, Вы можете написать `help` для получения списка доступных команд.

```
ftp> help
```

Допускается сокращение команд при вводе. Набор команд:

!	delete	literal	prompt	send
?	debug	ls	put	status
append	dir	mdelete	pwd	trace
ascii	disconnect	mdir	quit	type
bell	get	mget	quote	user
binary	glob	mkdir	recv	verbose
bye	hash	mls	remotehelp	
cd	help	mput	rename	
close	lcd	open	rmdir	

Базовые команды:

Подключение к FTP-серверу *ftp.domain.name*:

```
ftp> open ftp.domain.name
```

Список содержимого удалённой рабочей папки:

```
ftp> ls
```

или

```
ftp> dir
```

Переход на папку *newdir*:

```
ftp> cd newdir
```

Скачать файл *filename* с удалённого компьютера на локальный компьютер:

```
ftp> get filename
```

Скачать несколько файлов *file1*, *file2* и *file3* с удалённого компьютера на локальный компьютер (также Вы можете использовать шаблоны для скачивания нескольких файлов с тем же суффиксом или вообще всех файлов в папке):

```
ftp> mget file1 file2 file3
```

Загрузить файл *filename* с локального на удалённый компьютер:

```
ftp> put filename
```

Отключиться от удалённого FTP-сервера:

```
ftp> close
```

Закрыть локальный FTP-клиент:

```
ftp> quit
```

FTP-сессия шаг за шагом

Для подключения к анонимному FTP-серверу откройте локальный FTP-клиент:

```
ftp
```

Используйте команду open для подключения к серверу. Команда:

```
ftp> open anon.server
```

соединит Ваш FTP-клиент с анонимным FTP-сервером *anon.server*. Естественно, Вам нужно подставить имя реального сервера.

Когда удалённый FTP-сервер установит с Вами соединение, он уведомит Ваш локальный клиент, а затем спросит имя пользователя:

```
Connected to anon.server.
220 ProFTPD Server (Welcome . . . )
User (anon.server:(none)):
```

Для большинства анонимных FTP-серверов в качестве имени пользователя Вы должны ввести *anonymous* (или *ftp*). Удалённый FTP-сервер подтвердит, что Вы зашли как анонимный пользователей, и даст Вам инструкции по тому, что использовать в качестве пароля.

```
331 Anonymous login ok, send your complete email address as your
password.
```

```
Password:
```

В большинстве случаев сервер не проверяет правильность ввода адреса почты, введённого в качестве пароля, что не помешает Вам получить доступ к сервису, если Вы неправильно ввели адрес. Это считается нарушением сетевого этикета, но на самом деле это необходимо: не указывайте свой реальный адрес электронной почты! После того, как Вы ввели пароль, удалённый сервер отправит приветственное сообщение на локальный компьютер.

```
230-
   Welcome to ftp.anon.server, the public ftp server of anon.server. We
   hope you find what you're looking for.
   If you have any problems or questions, please send email to
   ftpadmin@anon.server
   Thanks!
230 Anonymous access granted, restrictions apply.
```

Теперь Вы можете использовать команды ls, dir, cd и get для скачивания файлов с сервера к себе на ПК.

Упражнения

4.13 Используя приведенные примеры, найдите и скачайте файл с анонимного FTP- сервера.

4.14 Используя браузер и поисковую систему, найдите анонимный FTP-сервер, на котором есть копия Алисы в Стране Чудес, а потом, используя командную строку FTP-клиента – не браузер – скачайте файл.

4.15 Какие хорошие FTP-клиенты Вы знаете? Могут ли они автоматизировать консольный ввод и предоставить удобный графический интерфейс? Теряете ли Вы какую-либо функциональность по сравнению с командой строкой?

4.16 Может ли Ваш ПК стать FTP-сервером?

Telnet и SSH

Telnet позволяет локальному пользователю отправлять множество различных команд на удалённый компьютер. Локальный пользователь может задавать команды удалённому компьютеру, выполнять различные действия и получать данные на локальный компьютер, почти как если бы он сидел за удалённым компьютером. **Secure Shell (SSH, «**безопасная оболочка**»)** предназначен для защищённой шифрованной замены открытого текста telnet.

Большинство версий Windows, OSX и Linux содержат клиент telnet в формате командной строки; для получения доступа к нему, откройте командную строку или окно терминала и введите:

```
telnet
```

Для доступа к серверу telnet Вам необходимы учётная запись и пароль, установленные для Вас администратором сервера, так как программа telnet позволяет выполнять много различных действий, и некоторые из них могут серьёзно повлиять на безопасность удалённого компьютера.

Telnet раньше использовался для того, чтобы администраторы компьютеров могли удалённо управлять серверами и обеспечивать поддержку пользователей на расстоянии. Этой услугой в Интернете сейчас почти не пользуются.

Telnet также может быть использован для ряда других задач, таких как отправка и получение электронной почты и просмотра исходного кода веб-страниц (хотя telnet, пожалуй, самый сложный способ выполнения этих задач). Многие из этих вещей являются законными, но они могут использоваться в незаконных или аморальных целях. Вы можете использовать telnet для проверки электронной почты, а также просматривать не только тему сообщения, но и первые несколько строк, что позволит Вам решить, следует ли удалять его, не загружая сообщение целиком.

Если Вы собираетесь использовать SSH, убедитесь, что Вы используете последнюю версию, поскольку старые версии имеют различные уязвимости, и многие автоматические сканеры уязвимостей постоянно ищут их в Интернете.

Игра началась: Командуй мной

Темный экран мерцал перед толстыми очками дедушки, курсор моргнул в нетерпении, ожидая команды. Его седые редкие волосы лениво окружали его морщинистую голову, дедушка стучал по клавиатуре. Джейс смотрела на бесшумного пианиста, играющего на клавиатуре своего компьютера, тук, тук, тук, тук. Он повернулся, чтобы посмотреть в молодые глаза Джейс, и улыбнулся ей. «Джейс, я собираюсь показать тебе новый мир. Пристегни ремни безопасности», — он подмигнул восьмилетней девчонке.

Джейс, сидя в компьютерном кресле, едва доставали ногами до пола, а её дедушка сидел напротив экрана компьютера. Она слышала длинный низкий гудок, исходящий из небольшой коробки, стоявшей рядом. Белая коробка загорелась зелёными и красными огнями, звук, издаваемый ею, стал похож на звук утки, напуганной мусоровозом. Дедушка взволнованно поднял брови и вдумчиво уставился на чёрный экран перед ним. Утка умолкла, и все огни зажглись зелёным на телефонной коробке.

Дедушка сказал: «Смотри.»

Обычно, когда дедушка говорит «Смотри» — то можно ожидать, что что-то взорвётся или задымится. Так или иначе, «смотри» означало что бабушка будет злиться из-за его очередной выходки. Джейс нравилось слышать эти слова, потому что это предвещало начало захватывающего приключения.

Экран компьютера вышел из спящего режима и вывел баннер ASCII-текста, окружавшего слова «Добро пожаловать в Cline's Bulletin Board System (BBS).» «Мы внутри!» — дедушка захлопал и попытался «дать пять» восьмилетней Джейс. Их руки разминулись на несколько дюймов, и он чуть не ударил девочку по лицу. Она засмеялась и дедушка тоже.

Они оба посмотрели на клавиатуру и на экран компьютера. Дедушка скрестил пальцы, пока Джейс чесала затылок, пытаясь выяснить, что происходит. Дедушка начал вводить команды на бесшумном пианино, опустив голову вниз, как стервятник, выискивающий жертву. Голову вверх, голову вниз, голову вверх, голову... Ой. Он откинулся на спинку стула. Дедушка забыл что-то очень важное.

Он сделал паузу и заговорил как учитель: «Джейс, извини, я забыл рассказать тебе, что здесь происходит. Сейчас я подключён к другому компьютеру через нашу телефонную линию. Эта шумная штука вон там называется «Модем»; его работа заключается в преобразовании цифровых сигналов в аналоговые и наоборот.» Джейс уже много знала о

телефонных системах благодаря дедушкиному стремлению к проведению экспериментов при любой возможности. 48 вольт при нормальном использовании и 90 вольт во время уведомления телефонного звонка; она знала больше, чем любой телефонный техник. Старая обычная телефонная система (или POTS) уже стала предметом шуток между Джейс и её дедушкой. Бабушка не понимала этого юмора, что делало его ещё более смешным.

Телефонные линии могут быть использованы заинтересованной стороной, но это можно обнаружить с помощью регулятора напряжения. Произойдёт скачок напряжения, и оно будет постоянно немного повышенным, если кто-то попытается прослушать линию. Джейс думала, что дедушка любил свой вольтметр больше, чем бабушку; он никогда не выходил из дома без него. Дедушка зашёл так далеко, что назвал его «Валери». Валери — вольтметр. Это был его лучший друг, не считая Джейс.

Джейс смотрела светящимися от любопытства глазами на дедушку, вернее ей любопытно было послушать лекцию по переходу от аналоговой к цифровой модуляции с преобразованием звука в цифровой сигнал. Это в основном именно то, что делает модем. Дедушка продолжал свою лекцию для маленькой студентки: «Компьютер, к которому я подключён, позволяет мне подключиться к другим компьютерам и получить услуги, которые они предоставляют.» Девочка уловила слово, которое она не слышала раньше — «услуги».

«Дедушка, что ты имеешь в виду под словом «услуги»?» — спросила девочка, ожидая ответа, каким-то образом связанного с фаст-фудом. «Отличный вопрос, моя дорогая,» — дедушка ожидал от Джейс подобный вопрос. «Мой компьютер подключён к сети компьютеров, и у меня есть возможность подключения к другим компьютерам по всему миру,» — с радостью ответил он. «Этот модем позволяет мне разговаривать с этими другими компьютерами, которые предлагают доступ к файлам, информации, общению с людьми и другие замечательные услуги, такие как File Transport Protocol, Usenet, IRC, Telnet и электронная почта.»

Джейс не удовлетворил этот ответ, и это привело к гораздо большему количеству вопросов, которые подряд посыпались на дедушку. Она пополнила свой запас вопросов и начала «обстрел»: «Что такое File Transport Protocol? Что такое MIC? Где находится Telnet? Нужны ли для электронных писем специальные марки? Какого они цвета в цифровом мире? Кто придумал Usenet? Почему они называют это Email? Знает ли бабушка об этих услугах? Почему они называются услугами? Откуда берутся дети? Откуда взялось желе?»

Дедушке пришлось закрыть уши, чтобы оградить себя от натиска вопросов. «Подожди, подожди, подожди, помедленнее.»

Игра окончена

DNS

Когда Вы хотите позвонить другу, Вам нужно знать правильный номер телефона; когда Вы хотите подключиться к удалённому компьютеру, Вы также должны знать его номер. Возможно, Вы помните из предыдущих уроков, что для компьютеров в Интернете этот номер — это IP-адрес.

Компьютерам очень легко работать с IP-адресами, но люди предпочитают использовать имена, в этом случае имена доменов. Например, для подключения к веб-сайту Hacker

Highschool введите www.hackerhighschool.org в адресной строке веб-браузера. Тем не менее, веб-браузер не может использовать это имя для подключения к серверу, на котором размещён сайт Hacker Highschool — ему нужен IP-адрес. Это означает, что локальный компьютер должен иметь некоторые средства перевода доменных имён в IP-адреса. Если бы в Интернете были только сотни или даже тысячи компьютеров, то можно было бы создать простую таблицу **(**файл хостов**)**, хранящуюся на компьютере, для поиска этих адресов. Однако, к лучшему или худшему, помимо существования миллионов адресов компьютеров в Интернете, зависимости между доменными именами и IP-адресами постоянно меняются.

Domain Name Service (DNS, сервис доменных имён**)** используется для динамического перевода доменных имён в IP-адреса (и наоборот). При вводе доменного имени www.domainname.com в адресную строку Ваш веб-браузер соединяется с DNS-сервером, выбранным Вашим провайдером. Если у этого DNS-сервера в базе данных есть www.domainname.com, то он возвращает IP-адрес Вашему компьютеру, позволяя Вам подключиться.

Если у Вашего DNS-сервера в базе данных нет www.domainname.com, то он посылает запрос на другой DNS-сервер, и они будут продолжать посылать запросы другим DNS-серверам, пока не найдут нужный IP-адрес или не установят, что домен имеет неверное имя.

Упражнения

4.17 Откройте окно командной строки и определите IP-адрес Вашего компьютера. Какую команду Вы использовали? Какой у Вас IP-адрес?

4.18 Определить IP-адрес Вашего DNS-сервера. Какую команду Вы использовали? Что такое IP-адрес DNS-сервера?

4.19 Пропингуйте www.isecom.org. Получаете ли Вы ответ? Какой IP-адрес отвечает на пинг?

4.20 Можете ли Вы изменить используемый компьютером DNS-сервер? Если да, то измените конфигурацию компьютера так, чтобы он использовал другой сервер DNS. Пропингуйте www.isecom.org снова. Вы получили тот же ответ? Почему?

DHCP

DHCP (**Dynamic Host Configuration Protocol,** протокол динамической конфигурации хоста**)** позволяет серверу локальной сети раздавать IP-адреса в сети. Серверу предоставляется блок IP-адресов для использования. Когда компьютер присоединяется к сети, он получает IP-адрес. Когда компьютер выходит из сети, его IP-адрес становится доступным для использования другим компьютером.

Такой подход удобен для больших сетей компьютеров, так как нет необходимости для каждого компьютера иметь индивидуально назначенный, статический IP-адрес. Вместо этого используется сервер DHCP. Когда новый компьютер подключается к сети, первое, что он делает, — это запрос IP-адреса с сервера DHCP. Как только ему был назначен IP-адрес, компьютер получает доступ ко всем услугам в сети.

Теперь подумайте о следующем. Большинство беспроводных сетей предлагают DHCP; это означает, что любой желающий может получить IP-адрес в этой подсети. Если Вы работаете в кафе, то это именно то, что Вам подходит; но если Вы работаете в безопасном офисе, то, вероятно, Вы захотите использовать фиксированные IP-адреса. Возможны разные варианты.

Соединения

В старые недобрые времена компьютеры подключались к Интернету через модем. Модемы преобразовывают биты в звуки и обратно; эти две операции называются модуляция и демодуляция, отсюда и название. Скорость модема измеряется в бодах **(baud)** и в битах в секунду (**bps**). Высшая скорость передачи данных обычно означает более высокое значение bps, но Вы также должны учитывать то, что Вы планируете делать. Есть определённые приложения — такие как telnetting в **Multi-User Dungeons (MUDs, «**многопользовательские миры**»)** — для которых двадцатилетний модем в 300 бод все ещё приемлемый (при условии, что Вы не слишком быстро печатаете), в то время как приложения с большой пропускной способностью (как, например, передача потокового видео) часто могут нагружать даже самый мощный кабель или DSL-соединения.

ISPs

Процесс подключения к Интернету не так прост, как кажется. Вам нужно получить доступ к серверу, который подключит компьютер к Интернету. Сервер выполняет всю тяжёлую работу, и он постоянно включен. Сервер управляется провайдером **(ISP)**.

Интернет-провайдер имеет постоянную точку присутствия PoP (point-of-presence) в Интернете, а также серверы, предоставляющие услуги, которыми Вы можете пользоваться. Но Вы можете запустить эти услуги и самостоятельно. Например, Вы можете запустить почтовый сервер на локальном компьютере, но для этого нужно, чтобы Ваш компьютер был постоянно включен и подключен к сети, ожидая сеансы обмена информацией. Провайдер объединяет усилия большого количества пользователей, поэтому почтовый сервер работает всё время, вместо того чтобы бездействовать. Компьютеры провайдера используют высокоскоростное соединение для подключения к точке доступа к сети (**Network Access Point, NAP**). Затем эти точки доступа связываются друг с другом через сверхскоростное соединение, называемое опорной сетью **(backbone)**. Все эти вещи вместе и составляют Интернет.

Старые обычные телефонные службы

Старые обычные телефонные службы **(Plain old telephone service, POTS)** были когда-то наиболее широко используемым методом доступа к Интернету. Его основным недостатком является низкая скорость, но во многих случаях это компенсируется его доступностью. Большинство национальных Интернет-провайдеров имеют большое количество местных номеров доступа, и почти все по-прежнему имеют домашнюю телефонную линию. В теории если у Вас акустический модем и полный карман сдачи, Вы можете подключиться практически из любого общественного телефона-автомата (если Вам удастся его найти). Хотя сомневаемся, что Вам действительно хотелось бы так сделать.

Телефонное соединение медленное. Самые быстрые телефонные модемы рассчитаны на скорость 56600 бит в секунду (bps). Что, впрочем, не соответствует действительности. Из-за ограничений мощности реальная скорость загрузки составляет примерно до 53000 бит в секунду, а действительная скорость, как правило, значительно ниже. Эти показатели трудно сравнивать с DSL или кабельными модемами.

Тем не менее, телефонная связь широко доступна, относительно дешёвая (а иногда и бесплатная). Вы бы не захотели торговать пиратскими фильмами по телефонной линии, потому что это аморально, незаконно и займёт Вашу телефонную линию на всю ночь и, возможно, на все выходные, но вы, безусловно, можете отправлять текстовые письма бабушке. И если Вы используете telnet, Вы даже можете сделать это с помощью старенького компьютера на DOS-е, который Вы вытащили из подвала.

DSL

Цифровая абонентская линия (**Digital Subscriber Line**) — это способ отправки больших объёмов информации по проводам, которые уже существуют для телефонной линии. Основным преимуществом перед стандартной телефонной службой является то, что этот способ намного быстрее, чем аналоговые модемы, и он обеспечивает постоянное соединение. Кроме того, DSL позволяет совершать и принимать регулярные телефонные звонки, пока Вы подключены к Интернету. Её основным недостатком является то, что её доступность ограничена тем, насколько близко Вы находитесь к коммутационному оборудованию телефонной компании — если Вы живете слишком далеко вниз по линии, то Вам не повезло.

Кабельные модемы

Кабельные шлюзы не используют традиционные телефонные линии для подключения к Интернету. Вместо этого они используют коаксиальный кабель (или волоконно-оптические линии, если Вам действительно повезло), предоставляемый кабельной компанией. Как и DSL, кабельные шлюзы позволяют Вам совершать обычные телефонные звонки, пока Вы подключены к Интернету, и они обеспечивают постоянное соединение, но кабельные шлюзы, как правило, быстрее, чем DSL.

Кабельные шлюзы имеют несколько недостатков. Во-первых, кабельный шлюз является общим ресурсом, поэтому скорость подключения понижается, когда другие пользователи подключаются к тому же кабелю. Во-вторых, кабельный доступ есть только в районах, где кабельные компании установили необходимую проводку. И самый серьезный недостаток заключается в том, что любой трафик, проходящий через кабель, может просмотреть любой другой пользователь, подключенный к нему же. Это означает, что если Вы при подключении компьютера к кабельной линии не включите брандмауэр, все остальные подключенные компьютеры смогут видеть Ваш компьютер и все его файлы. Вы действительно хотите поделиться с другими информацией о своем банковском счёте?

Wimax

Wimax — это беспроводной способ подключения, который обычно конкурирует с DSL. Он используется в местах, где проводную инфраструктуру слишком дорого или невозможно установить. На уровень сигнала могут влиять здания, деревья или другие крупные объекты. Некоторые версии используют фиксированную точку доступа, а другие дают Вам мобильный доступ на действительно большом пространстве.

Wifi

Wifi не является способом подключения к провайдеру, но это распространенный способ для подключения к Интернету дома или в коммерческих учреждениях, таких как торговые центры или кафе. Большинство смартфонов и все ноутбуки сейчас используют Wifi, поэтому это излюбленная мишень для злоумышленников. Представьте себя голым в переполненной комнате, когда Вы пользуетесь общественным Wifi: прикройтесь, убедитесь, что никто не смотрит на Вас, но каждый хочет посмотреть. Конечно, теперь Вам захотелось прочитать урок по безопасности в беспроводных сетях, верно?

Упражнения

4.21 Какой тип подключения к Интернету у Вас дома? Как Вы это узнали? И самое главное:

4.22 Кто может видеть Вас в этой сети? (Как Вы можете это узнать?)

4.23 Какая скорость Вашего соединения? Можете ли Вы улучшить её без обращения к Интернет-провайдеру?

4.24 Какие дополнительные услуги предоставляет Ваш провайдер? Мы уже говорили об услугах, Ваш провайдер может поддерживать несколько.

4.25 Какие услуги Вы можете предоставить со своего компьютера?

Пища для ума: Играя с **HTTP**

HTTP (сокращение от Hypertext Transfer Protocol — протокол передачи гипертекста) — расположен на вершине стека TCP/IP и определён в двух основных RFC:

- 1945 для 1.0 (начиная с 0.9).

- 2616 для 1.1.

Есть несколько существенных обновлений и отличий версий 1.0 и 1.1 относительно расширяемости (Extensibility), кэширования (Caching), оптимизации полосы пропускания (Bandwidth optimization), управления сетевым соединением (Network connection management), передачи сообщений (Message transmission), защиты Интернет-адресов (Internet address conservation), уведомления об ошибках (Error notification), безопасности, целостности и аутентификации (Security, integrity, and authentication), соглашении о данных (Content negotiation) [3]. Различия между 1.0 и 1.1 полезны для получения информации о веб-сервере.

В целом HTTP — это протокол без сохранения состояния, в котором клиент отправляет HTTP-запрос на сервер, который, в свою очередь, отправляет HTTP-ответ: в этом состоит суть парадигмы запрос/ответ.

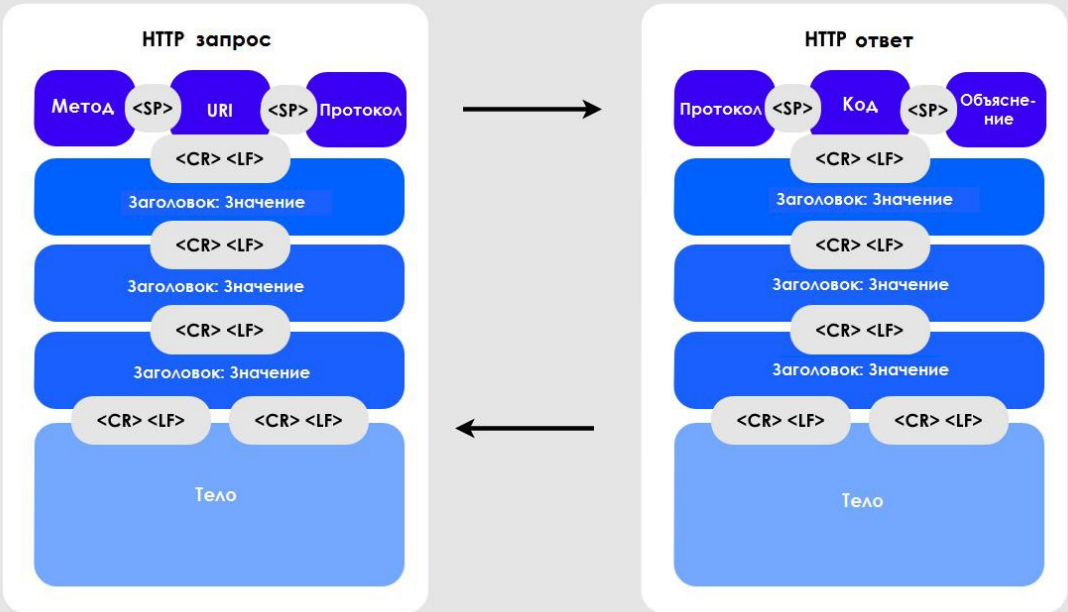

Рисунок **4.1:** HTTP

Как Вам уже, возможно, известно, мы можем получить много информации, отправляя команды на HTTP-сервер. Мы воспользуемся несколькими базовыми сетевыми утилитами:

- netcat: набор утилит TCP/IP
- curl: набор утилит HTTP
- прокси: такие, как OWASP ZAP или Burpsuite free

Сниффинг соединения между Вами и **HTTP**-сервером **HHS**

Используйте прокси для осуществления соединения через браузер. Перейдите по ссылке http://www.hackerhighschool.org и перехватите свой запрос:

```
GET / HTTP/1.1
Host: www.hackerhighschool.org
User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10.8; rv:11.0) Gecko/20100101 Firefox/11.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-us,en;q=0.5
Accept-Encoding: gzip, deflate
Proxy-Connection: keep-alive
```

и ответ:

```
HTTP/1.1 200 OK
Content-Length: 10376
Date: Fri, 03 Feb 2013 09:11:17 GMT
Server: Apache/2.2.22
Last-Modified: Mon, 06 Feb 2013 09:31:18 GMT
ETag: "2f42-4b8485316c580"
Accept-Ranges: bytes
Identity: The Institute for Security and Open Methodologies, The Institute for Security and Open Methodologies
P3P: Not supported at this time, Not supported at this time
Content-Type: text/html
Connection: keep-alive

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd"[]><html xmlns="http://www.w3.org/1999/xhtml" dir="ltr" lang="en-US" xml:lang="en"><head><meta http-equiv="Content-Type" content="text/html; charset=UTF-8" /><title>Hacker Highschool - Security Awareness for Teens</title>
[...]
```

Упражнения

4.26 Определите части запросов через прокси, пользуясь диаграммами.

4.27 Есть ли в заголовках какая-нибудь интересная информация?

Ваше первое соединение, настроенное вручную

Netcat можно использовать для соединения с веб-сервером, пользуясь настройками портов хоста.

Начните, введя команду:

```
nc www.hackerhighschool.org 80
```

Затем дважды нажмите клавишу Enter.

```
GET / HTTP/1.0
```

Сервер ответит:

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd"[]>
<html xmlns="http://www.w3.org/1999/xhtml" dir="ltr" lang="en-US"
xml:lang="en"><head>
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8" />
<title>ISECOM - Institute for Security and Open Methodologies</title>
<meta name="description" content="Description" />
```

Как Вы видите, создаётся впечатление, что страница получена с isecom.org, а не с hackerhighschool.org. Почему?

Одно из предположений заключается в том, что один и тот же хост обслуживает и сайт HHS, и сайт ISECOM. Такой вариант возможен?

Чтобы разобраться с этим, определите IP-адрес hackerhighschool.org:

```
nslookup www.hackerhighschool.org
[...]
Non-authoritative answer:
www.hackerhighschool.org      canonical name = hackerhighschool.org.
Name: hackerhighschool.org
Address: 216.92.116.13
```

А теперь для www.isecom.org:

```
nslookup isecom.org
[...]

Non-authoritative answer:
```

```
Name: isecom.org
Address: 216.92.116.13
```

IP-адрес тот же! С помощью netcat можно отобразить хост, вручную добавив заголовок Хост (Host) и используя HTTP 1.1:

```
GET / HTTP/1.1
Host: www.hackerhighschool.org

HTTP/1.1 200 OK
Content-Length: 10376
Date: Fri, 03 Feb 2013 09:11:17 GMT
Server: Apache/2.2.22
Last-Modified: Mon, 06 Feb 2013 09:31:18 GMT
ETag: "2f42-4b8485316c580"
Accept-Ranges: bytes
Identity: The Institute for Security and Open Methodologies, The Institute for Security and Open Methodologies
P3P: Not supported at this time, Not supported at this time
Content-Type: text/html
Connection: keep-alive
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd"[]>
<html xmlns="http://www.w3.org/1999/xhtml" dir="ltr" lang="en-US" xml:lang="en"><head>
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8" />
<title>Hacker Highschool - Security Awareness for Teens</title>
```

Метод запроса

Ещё одной модифицируемой частью HTTP-запроса является его метод. Чаще всего веб-приложения используют GET и POST запросы, но другие протоколы запросов тоже могут быть активны на веб-сервере или сервере приложений. Среди других часто используемых методов можно выделить следующие:

- **OPTIONS** — используется для определения поддерживаемых параметров запроса. Если у Вас работает веб-сервер, то помните, что предоставление этой информации может привести к различным проблемам.
- **GET** — используется для получения информации непосредственно через URL, например:
  `http://www.usairnet.com/cgi-bin/launch/code.cgi?Submit=Go&sta=KSAF&state=NM`
  Видите фрагмент строки после вопросительного знака? Это данные запроса. Передавать данные таким способом рискованно, поскольку они находятся на виду у всех и их легко

изменить.

- **HEAD** — используется аналогично методу GET, но сервер не возвращает фактическую страницу.
  Этот метод можно использовать для определения вариантов доступа, оптимизации потребления пропускной способности и – в некоторых случаях – обхода средств управления доступом. На самом деле, некоторые реализации ACL проверяют только GET запросы. В таких случаях Вы можете обнаружить уязвимость.
- **POST** — используется для передачи данных веб-приложениям — подобно методу GET — но данные включаются в тело запроса, по крайней мере, хоть немного вне пределов видимости.
- **PUT** — используется для размещения ресурсов на веб-сервере или для их обновления.
  Во многих случаях этот метод должен быть запрещён или защищён средствами управления аутентификацией (Authentication Control). Иначе это может стать замечательной находкой для Вас.
- **DELETE** — используется для удаления ресурсов с веб-сервера.
  Этот метод должен быть запрещён или защищён средствами управления аутентификацией (Authentication Control) (аналогично PUT, представленному выше).
- **TRACE** — используется на прикладном уровне как обратная петля (loopback), которая отображает сообщения.
  Этот метод отладки должен быть запрещён, особенно в производственной среде, поскольку он раскрывает конфиденциальную информацию и представляет собой уязвимость, так как может использоваться в эксплойтах межсайтового скриптинга.
- **CONNECT** — для использования веб-сервера в качестве прокси.
  Этот метод должен быть запрещён или защищён средствами управления аутентификацией (Authentication Control), поскольку он позволяет другим осуществлять соединение со сторонними сервисами, используя IP прокси.

Учитывайте также, что протоколы, основанные на HTTP, могут добавлять и другие методы (как, например, WebDAV). Вы можете изменять метод запроса с целью просмотра ответов сервера (в чём-то, возможно, представляющих интерес), запроса известных методов, а также просмотра «реакции» на произвольно выбранные слова.

Запрос **OPTIONS**

Начните сеанс netcat как обычно:

```
# nc www.hackerhighschool.org 80
```

Но в этот раз не нажимайте дважды клавишу Enter. Вместо этого введите следующую строку:

```
OPTIONS / HTTP/1.1
```

и Вы получите ответ, похожий на следующий:

```
Host: www.hackerhighschool.org
HTTP/1.0 200 OK
Date: Tue, 07 Feb 2013 08:43:38 GMT
Server: Apache/2.2.22
```

```
Allow: GET,HEAD,POST,OPTIONS
Identity: The Institute for Security and Open Methodologies, The
Institute for Security and Open Methodologies
P3P: Not supported at this time, Not supported at this time
Content-Length: 0
Content-Type: text/html
```

Запрос **HEAD**

В этот раз, начав сеанс, введите метод HEAD.

```
# nc www.hackerhighschool.org 80
HEAD / HTTP/1.1
Host: www.hackerhighschool.org

HTTP/1.0 200 OK
Date: Tue, 07 Feb 2013 08:41:14 GMT
Server: Apache/2.2.22
Last-Modified: Fri, 13 Feb 2013 15:48:14 GMT
ETag: "3e3a-4bd916679ab80"
Accept-Ranges: bytes
Content-Length: 15930
Identity: The Institute for Security and Open Methodologies
P3P: Not supported at this time
Content-Type: text/html
Age: 45
Connection: close
```

Позвольте мне использовать Вас как прокси: запрос **CONNECT**

```
# nc www.hackerhighschool.org 80
CONNECT http://www.isecom.org/ HTTP/1.1
Host: www.hackerhighschool.org
```

Упражнение

4.28 Воспользуйтесь netcat (nc) для того, чтобы испытать все перечисленные выше методы запросов на сетевых серверах HHS или сервере, запущенном для тестирования. Какую интересную информацию Вам удалось обнаружить?

Составление сценариев **HTTP**-запросов с помощью **curl**

В некоторых случаях тестирование веб-приложений основывается не только на ответах

веб-сервера, но и на работе уровня (веб-)приложений. Часто можно обнаружить уязвимости веб-приложения, изменяя параметры GET и POST, cookies и значения заголовков. Полезной утилитой для bash-скриптинга является команда **curl** — это утилита командной строки для запросов веб-страниц. Но по сравнению с netcat, логика работы curl немного другая.

Данная команда:

```
# curl http://www.isecom.org
```

не то же самое, что следующая:

```
# nc www.isecom.org 80
GET / HTTP/1.1
```

Чтобы убедиться в этом, Вы можете ввести параметр -v для подробного вывода:

```
# curl -v http://www.isecom.org/
* About to connect() to www.isecom.org port 80 (#0)
*   Trying 216.92.116.13...
* connected
* Connected to www.isecom.org (216.92.116.13) port 80 (#0)
> GET / HTTP/1.1
> User-Agent: curl/7.26.0
> Host: www.isecom.org
> Accept: */*
>
* HTTP 1.0, assume close after body
< HTTP/1.0 200 OK
< Date: Tue, 07 Feb 2013 09:29:23 GMT
< Server: Apache/2.2.22
< Last-Modified: Fri, 13 Feb 2013 15:48:14 GMT
< ETag: "3e3a-4bd916679ab80"
< Accept-Ranges: bytes
< Content-Length: 15930
< Identity: The Institute for Security and Open Methodologies
< P3P: Not supported at this time
< Content-Type: text/html
< Age: 247
< Connection: close
<
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
```

```
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd"[]>
<html xmlns="http://www.w3.org/1999/xhtml" dir="ltr" lang="en-US"
xml:lang="en">
[...]
```

Как видите, curl автоматически выбирает версию HTTP 1.1, добавляет заголовок хоста (host), клиентского приложения (user agent) и допустимых форматов ресурса (accept). Из этого следует важное правило для хакеров: знай то, с чем работаешь.

К счастью, curl — хорошая утилита, которую можно тщательно настроить, задавая определённые параметры.

Для просмотра всех параметров введите `curl –help`.

Среди параметров команд, похожих на вышеприведенный пример с netcat, можно выделить следующие:

- -H для добавления строки заголовка
- **-X** для выбора метода запроса (также известного как Команда)
- **-d** для добавления POST данных
- **-i** для включения заголовков протокола в выходном ресурсе
- **-s** для включения «тихого» режима, удобного для скриптинга

Используя curl и немного bash-скриптинга, Вы можете автоматизировать тестирование веб-приложений. Поиск интересных HTTP заголовков от сервера можно достаточно просто автоматизировать с помощью curl и grep:

```
# curl -siX HEAD http://www.isecom.org/ | grep "Server:"
Server: Apache/2.2.22
```

Упражнение

4.29 Дополните вышеприведенный сценарий для запроса других HTTP-заголовков и потенциально полезной информации.

Ссылки и дополнительная литература

http://www.ietf.org/rfc/rfc1945.txt

http://www.ietf.org/rfc/rfc2616.txt

http://www8.org/w8-papers/5c-protocols/key/key.html

http://netcat.sourceforge.net/

http://curl.haxx.se/

Выводы

Всемирная паутина — это значительно более широкое понятие, чем Интернет: кроме HTTP есть много других видов служб. FTP, SSH, DNS, DHCP и многие другие «раскрывают окна» в компьютеры других пользователей — в том числе, и в Ваш. Понимание того, как Вы подключаетесь к этим службам («через правильные каналы» или как-то по-другому), является ключевым моментом для осознания того, как Вы или Ваш компьютер могут быть атакованы — или как самому провести атаку. Просто помните о девизе: взламывайте всё, но без ущерба другим.

Today's teens are in a world with major communication and productivity channels open to them and they don't have the knowledge to defend themselves against the fraud, identity theft, privacy leaks and other attacks made against them just for using the Internet. This is the reason for Hacker Highschool.

The Hacker Highschool project is the development of security and privacy awareness learning materials for junior high and high school students.

Hacker Highschool is a set of lessons and a practical means of making hackers. Beyond just providing cybersecurity awareness and critical Internet skills, we need to teach the young people of today how to be resourceful, creative, and logical, traits synonymous with hackers. The program contains free security and privacy awareness teaching materials and back-end support for teachers of accredited junior high, high schools, and home schooling. There are multiple workbooks available in multiple languages. These are lessons that challenge teens to be as resourceful as hackers, including safe Internet use, web privacy, researching on the internet, avoiding viruses and Trojans, legalities and ethics, and more.

The HHS program is developed by ISECOM, a non-profit, open-source research group focused on security awareness and professional security development and accreditation.

Hacker Highschool
SECURITY AWARENESS FOR TEENS

ISECOM