Hacker Highschool

SECURITY AWARENESS FOR TEENS



УРОК **4**
ГРАЮЧИ З ДЕМОНАМИ

Hacker Highschool
SECURITY AWARENESS FOR TEENS

ISECOM

УВАГА

Проект Hacker Highschool є засобом навчання і, як в будь-якому навчальному засобі, існує небезпека. Деякі уроки, якщо ними зловживати, можуть призвести до фізичної травми. Також додаткові небезпеки можуть бути там, де ще недостатньо досліджень про можливі наслідки випромінювань від специфічної техніки. Студенти, які використовують ці уроки, повинні перебувати під контролем викладача і, в той же час, заохочуватися на вивчення, практику і заняття. ISECOM не несе відповідальності за застосування інформації, отриманої з даних матеріалів, і за подальші наслідки.

Наступні уроки та книги є відкритими і загальнодоступними на наступних умовах ISECOM:

Всі роботи проекту Hacker Highschool призначені для некомерційного використання з учнями початкової школи, слухачами юнацьких курсів Highschool, і студентами вищих навчальних закладів, приватних організацій або частково для домашнього навчання. Ці матеріали в будь-якій формі не можуть бути використані для продажу. Надання цих матеріалів будь-якому класу, навчальній організації або табору, в яких стягується плата, категорично заборонено без ліцензії, в тому числі на уроки в коледжі, університеті, професійно-технічних заняттях, літніх або комп'ютерних таборах тощо. Для придбання ліцензії відвідайте розділ сайту призначений для Ліцензування: http://www.hackerhighschool.org/licensing.html.

Проект HHS є результатом праці відкритого співтовариства і, якщо Ви знаходите наші труди цінними і корисними, ми просимо Вас підтримати нас шляхом придбання ліцензії, пожертвувань або спонсорства.

Содержание

Співробітники журналу

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

Перекладачі

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

ISECOM

Вступ

У світі існують тисячі мов, якими розмовляють люди, а деякі з них мають десятки діалектів. Ви самі можете знати декілька мов, але шанси на те, що, подорожуючи світом, Ви зможете розмовляти з кожним, кого зустрінете, близькі до нулю.

Так, Ви можете заперечити тим, що математика — це універсальна мова або що мову музики розуміє кожен, але будьмо реалістами. Спробуйте замовити склянку содової з лимоном та кульку морозива, користуючись «універсальними мовами», та подивимося, що у Вас із цього вийшло.

Якщо Вам пощастило побувати в країні, мову якої Ви не знаєте, то спробуйте замовити там лимонад, користуючись волинкою або саксофоном. Запишіть відео цього дійства і надішліть його до ISECOM. Ми дійсно хочемо на це подивитись! Навряд чи ми захочемо це почути, але побачити — точно хочемо.

Але щодня мільйони людей спілкуються один з одним в Інтернеті, користуючись однією спільною мовою. Не всі люди можуть говорити однією мовою; однак наші комп'ютери і мережі можуть так робити.

> Модель, яка використовується в сучасних мережах, називається клієнт-серверною моделлю. Фізичні комп'ютери (хости або сервери) пропонують служби (в UNIX вони називаються демонами **(daemons)**). Згадайте, як працює веб-сервер: він передає сторінку тоді, коли Ви зробите на неї запит. Нічого таємничого тут немає.

Але, насправді, не Ви запитуєте сторінку. Замість Вас це робить веб-браузер, тобто він виконує роль клієнта (або більш формально — цю роль виконує Ваш комп'ютер). У той же час Ваш комп'ютер також може бути і сервером. У цьому вся принадність мереж: Ви робите щось для мене; я щось роблю для Вас.

Помножте цю модель мільйон разів і Ви отримаєте Інтернет. Уявіть: мільйони комп'ютерів пропонують різні сервіси та послуги. Що потрібно для того, щоб бути клієнтом? Та чи можливо це спростувати? (Пошукайте значення цього слова, якщо ви не впевнені, що точно знаєте його. Зрештою, це хакерський курс.)

Готові Ви чи ні, давайте докладніше розглянемо ці питання.

Служби

У Вас є комп'ютер, і Ви знаєте, що на ньому є корисна інформація, або Ви можете приймати участь у загальній галюцинації, створюючи вигляд, що у Вас немає нічого, що має цифрову цінність. Також Ви знаєте, що в інших людей, мільйонів інших людей, є комп'ютери, на яких може бути корисна інформація, не кажучи про ресурси, на зразок процесорів, оперативної пам'яті, дискового простору та пропускної здатності.

Тепер уявіть, що ці люди та їхні комп'ютери з великою ймовірністю мають інформацію, яка представляє для когось інтерес. Єдине питання — як отримати цю інформацію.

Комп'ютери спілкуються один з одним через порти, користуючись протоколами, про які розповідалось в Уроці 3, але це не дозволить Вам читати потоки двійкових даних, якими обмінюються комп'ютери (якщо тільки у Вас є зайвий час). Потрібен спосіб, за допомогою якого можна отримати дані, їх інтерпретувати і представити їх в тій або іншій формі, яку Ви можете використовувати.

Комп'ютери передають дані за допомогою мережевих служб або просто служб. Ці служби дозволяють Вам переглядати веб-сторінки, обмінюватися листами, спілкуватися в чаті та взаємодіяти з віддаленими комп'ютерами. Цим службам відповідають певні номери портів.

Ваш комп'ютер, локальний комп'ютер, використовує програми, які називаються клієнтами, для інтерпретації інформації, яку Ви отримуєте. Ви можете отримати інформацію від сервера (який надає службу/запускає демона), через **Tor** мережу, від **Torrent** сідів або через мережу **peer-to-peer**.

> Звісно, Ваш комп'ютер також може надавати послуги іншим віддаленим комп'ютерам, таким чином виконуючи роль сервера даних або постачальника послуг. Якщо на Вашому комп'ютері з'явилися шкідливі програми, то Ви, можливо, надаєте достатньо багато послуг, про які самі не знаєте.

Приклади клієнтів — веб-браузери (далі браузери, — прим. переклад.), поштові клієнти, програми для чату, Skype, Tor-клієнти, Torrent-клієнти, RSS тощо. Ці застосунки знаходяться на рівні застосунків стеку протоколів TCP/IP. На цьому рівні всі дані, які передані, інкапсульовані, зашифровані, розшифровані, направлені і т. д. нижчими рівнями, конвертуються в щось, що Ви як користувач можете прочитати і зрозуміти.

HTTP і Мережа

Коли ми говоримо про «Інтернет», більшість людей мають на увазі Всесвітню Павутину **(World Wide Web)**. Всесвітня Павутина, або просто Мережа — це не Інтернет, це лише невелика частина доступних служб. Зазвичай під цим поняттям розуміють просто перегляд веб-сторінок через браузер.

> До речі, справжній Інтернет складається із всіх комп'ютерів, маршрутизаторів, дротів, кабельних та бездротових мереж, які переміщують дані різного роду. Мережевий трафік являє собою лише частину всього цього.

Мережа використовує **HTTP (HyperText Transfer Protocol,** протокол передачі гіпертексту**)** і застосунки (клієнти), які називаються браузерами, для доступу до документів на веб**-**серверах (далі серверах — прим. переклад*.)*. Інформація від віддаленого комп'ютера направляється на Ваш комп'ютер за протоколом HTTP, використовуючи зазвичай 80-й порт. Ваш браузер інтерпретує і демонструє Вам оброблену інформацію.

Не всі браузери працюють однаково. Кожен пропонує різні інструменти і відображає HTML-контент трохи (або дуже) по-різному. Питання безпеки і конфіденційності можуть бути вирішені успішно, але в різній мірі. Це означає, що Ви повинні знати, що Ваш браузер може і не може робити, які налаштування та плагіни нададуть Вам ідеальний баланс безпеки та конфіденційності (якщо Вам не подобаються шкідливі програми, реклама, спам та Ваші сусіди, які знають, що Ви полюбляєте дивитися відео про кошенят, які грають в зеленому желе).

Гіпертекстова частина протоколу HTTP відноситься до текстів, які читаються нелінійним способом. Зазвичай Ви читаєте лінійно (тобто послідовно): спочатку сторінка 1, потім сторінка 2; спочатку розділ 1, потім розділ 2; спочатку урок 1, потім урок 2, і так далі. Гіпертекст дозволяє переглядати інформацію в нелінійному порядку. По мірі вивчення чого-небудь Ви можете переходити з одного розділу на інший, переглянути їх повторно або перейти на іншу тему перш, ніж закінчити основну статтю. Ось у чому різниця між гіпертекстом і простим текстом.

У гіпертексті слова та ідеї пов'язуються не тільки зі словами, які безпосередньо оточують їх, але й з іншими словами, зображеннями, відео та музикою. Гіпертекст використовується не тільки в Мережі. Більшість повнофункціональних текстових процесорів дозволяють створювати сторінки у веб, або HTML, форматі; ці сторінки зберігаються локально. Ви читаєте їх у браузері так само, як звичайні веб-сторінки, тільки вони зберігаються на Вашому локальному, а не на віддаленому комп'ютері.

Створити свою веб-сторінку достатньо просто. Найлегший спосіб — це використати один з популярних текстових процесорів, на зразок OpenOffice/LibreOffice Writer, Microsoft Word або WordPerfect. Ці програми дозволяють Вам створювати прості веб-сторінки, поєднуючи текст, гіпертекст та зображення. За їхньою допомогою багато людей зробили достатньо функціональні веб-сторінки (можна навіть використовувати такі прости текстові редактори, як vi, який встановлений на більшості Unix-систем). Серед інших текстових редакторів можна виокремити Microsoft Notepad, Notepad++, SciTe, emacs тощо.

Але ці сторінки нічим не примітні. Цю проблему можна вирішити, використовуючи **CSS,** скрипти та анімацію. Ви можете витратити багато грошей на застосунки для дизайну дивовижних веб-сторінок. Ці застосунки дозволяють створювати цікаві ефекти на веб-сторінці, але вони більш складні у використанні. Однак зазвичай за їхньою допомогою робота полегшується. Більш дешева альтернатива — це взяти один з текстових редакторів, які орієнтовані для роботи з HTML та скриптовими мовами, вивчити синтаксис HTML, скриптів та написати власні веб-сторінки з нуля.

Щойно Ви завершите роботу над дизайном сторінок, Вам потрібно буде викласти їх на якомусь комп'ютері, в тому випадку, якщо хочете, щоб сторінки побачили інші люди. Інтернет**-**провайдери **(Internet Service Providers, ISPs)** надають послугу веб**-**хостингу на власних веб**-**серверах.

Ви можете запустити сервер на своєму домашньому ПК, але в цьому випадку можуть з'явитися деякі проблеми. Інформація, яка зберігається на сервері, доступна тільки тоді, коли сервер увімкнений, правильно функціонує та має відкрите підключення. Тому, якщо Ви хочете запустити сервер зі своєї спальні, Ви повинні тримати комп'ютер увімкненим весь час; Ви маєте бути впевнені, що програма на сервері коректно працює (сюди входить пошук та усунення несправностей комплектуючих, контроль вірусів, хробаків та інших атак, обробка помилок та недоліків всередині самої програми); та відкрите з'єднання з Інтернетом повинно бути стабільним та швидким. Інтернет-провайдери стягують додаткову плату за високу швидкість завантаження та фіксовану (статичну) IP-адресу, тому більшість людей платить за всю роботу третім особам.

Компанії, які надають веб-хостинг (далі хостинг — прим. переклад.), зберігають Вашу інформацію на своїх комп'ютерах. І це дуже добре, адже атаковані будуть їхні сервери, а не Ваші. Хороші компанії з веб-хостингу мають велику кількість резервних серверів і дотримуються політики регулярного резервування даних, таким чином Ваш сайт не зникне безвісти тільки через проблеми з «залізом»; технічна підтримка тримає сервери увімкненими, незважаючи на атаки та помилки в програмах; а низка відкритих підключень до Інтернету дає деяку гарантію від простоїв, перебоїв та зупинок в роботі. Тому все, що від Вас потрібно, — це оформити Вашу веб-сторінку, завантажити її на сервер хостингу, вимкнути ПК та йти спати. Ваша веб-сторінка буде доступна всьому світу, доки Ви будете сплачувати рахунки.

Також можна знайти організації, які пропонують безкоштовний хостинг. Деякі з них фінансуються платною рекламою, тобто будь-хто, хто захоче переглянути Вашу веб-сторінку, спершу побачить чиюсь рекламу. Але їм не доведеться нічого купувати, а Вам не доведеться нічого платити.

Вправи

4.1 Веб-сторінка — це просто текст, який повідомляє браузеру, де повинні розміститися зображення, відео та інші елементи. Ви можете побачити цей текст, переглянувши вихідний код сторінки. Запустіть свій улюблений браузер, перейдіть на ISECOM.ORG і завантажте сторінку. Тепер подивіться на вихідний код. Ви побачите декілька тегів зі словом "meta" в них. Наприклад, meta-charset="utf-8". Що це означає? На що це вказує?

4.2 Знайдіть ще 3 мета-теги і поясніть, на що вони вказують. Можливо, відповідь на це питання Вам доведеться пошукати в Інтернеті, тому ретельно продумайте, які ключові слова Ви будете використовувати для пошуку, щоб бути впевненим, що Ви знайдете правильні відповіді.

4.3 Збережіть вихідний код ISECOM.ORG собі на ПК. Відкрийте його в браузері. Що змінилося? Як Ви думаєте, чим викликані зміни?

4.4 Відкрийте вихідний код ISECOM.ORG в текстовому редакторі і Ви побачите, що це всього лише слова та цифри. Все, що Ви зміните або додасте в цей файл, після збереження вплине на вигляд сторінки в браузері. Приберіть рядки і Ви помітите видалення якихось елементів. Змініть слова і вони відобразяться зміненими. Тепер приберіть все зайве зі сторінки і додайте своє ім'я так, щоб воно виділялося серед інших слів (шрифт напівжирний та більший). Спробуйте. Збережіть. Відкрийте в браузері, подивіться, чи Ви досягли успіху. Ні? Тоді спробуйте ще раз!

Дивіться Пожива для розуму**:** Граємо з ***HTTP*** наприкінці цього уроку, якщо Ви хочете дізнатися більше.

Email — SMTP, POP та IMAP

Другий помітний аспект Інтернету — це, мабуть, електронна пошта. У себе на ПК Ви користуєтесь поштовим клієнтом, який з'єднується з поштовим сервером. Коли Ви створюєте поштовий акаунт (обліковий запис), Ви отримуєте унікальне ім'я у вигляді користувач@домен, і Вам потрібно створити для цього акаунту пароль.

> Існують 2 типи серверів для роботи з електронною поштою: **SMTP (Simple Mail Transfer Protocol,** простий протокол передачі пошти**)**, який відправляє пошту, і поштовий сервер, який отримує Вашу пошту (використовується **POP (Post Office Protocol,** протокол поштового відділення**)** або **IMAP (Internet Message Access Protocol,** протокол доступу до Інтернет-повідомлень**)**).

Протокол SMTP (нагадаємо ще раз) використовується для відправлення електронної пошти. SMTP визначає поля в електронному листі, включаючи поля FROM, TO, SUBJECT, CC та BODY. Старий добрий SMTP не потребує пароля і відправляє все у відкритому вигляді; кожен може прочитати Вашу пошту. Можливо, це був непоганий варіант, коли протокол був тільки розроблений, а Інтернет був невеликим світом, населеним однодумцями. Але він залишив лазівку, яка дозволяла будь-якому користувачу розсилати спам і робити інші капості, на зразок підміни електронної пошти **(email spoofing)**, яка, по суті, означає підміну (spoofing) адреси відправника. Майже всі сучасні поштові сервери використовують Secure SMTP; це означає, що Ви повинні підтвердити свою особу перш, ніж відправляти листи.

У наступних уроках ми покажемо Вам, як працює підміна та як виявити її у заголовках електронної пошти. Ця жменя знань неймовірно швидко може перетворити Вас з «недосвідченої ягнички» у «впевненого вовка».

POP3 (Post Office Protocol, версія **3)** — це протокол, що «зберігає та складає». Поштовий сервер отримує Ваші листи і зберігає їх для Вас, доки Ви не з'єднаєтесь та не завантажите (скинете) Вашу пошту. Відправлення пошти відбувається з використанням SMTP. Це гарний підхід до роботи з електронною поштою в тому випадку, якщо у Вас dial-up з'єднання, оскільки він займає менше часу для відправлення і отримання електронної пошти, а Ви можете читати електронну пошту, навіть не маючи підключення до Інтернету.

IMAP, з іншого боку, за стандартним налаштуванням зберігає Вашу пошту на сервері. Багато корпоративних поштових рішень використовує якийсь варіант IMAP в залежності від постачальників програмного забезпечення. В IMAP Ви можете створювати теки у своїх поштових скриньках і переміщувати листи між ними. Коли Ви з'єднуєтесь з IMAP-сервером, Ваші поштові скриньки і сервер синхронізують теки, їхній вміст та пошту, що видалена. Це вагома перевага: Ви можете отримати всю свою пошту з будь-якого пристрою: ноутбука, телефона або планшета. До того ж, Ви можете завантажити і зберігати пошту в себе на ПК.

Однак цей протокол має й два недоліки: по-перше, вочевидь Вам потрібно обмінюватися більшим об'ємом інформації, тому необхідне більш швидке з'єднання і більша кількість часу. По-друге — обмежений об'єм. Ваш поштовий сервер призначить розмір Вашої поштової скриньки, який не можна перевищити. Якщо Ваша скринька буде повна, Ви не зможете отримувати пошту, доки не видалите інші листи (або не придбаєте більше місця). Зрештою це означає, що корпоративна IMAP-пошта потребує керування даними. Ви повинні переміщувати пошту в локальні сховища та регулярно чистити відправлену пошту, спам та видалені листи для економії простору. Листи з прикріпленнями «знищать» Вас. У наш час, коли є можливість створити безкоштовний обліковий запис з величезним безкоштовним сховищем даних, всі ці

заходи безпеки можуть видатися безглуздими. Доки Ви не отримаєте позов. Або хтось не спробує зламати поштовий сервер і отримати ВСЮ Вашу пошту..

POP і IMAP сервери запитують пароль для отримання доступу до облікового запису. Але обидва протоколи відправляють абсолютно все у відкритому вигляді, в т. ч. і паролі, тому, теоретично, кожен може їх прочитати. Ви повинні використовувати шифрування для маскування процесу входу (наприклад, SSL) та вмісту листа. Ось чому у багатьох поштових клієнтів є прапорець Використовувати *SSL*.

Коли Ви в поштовому клієнті натискаєте кнопку Відправити, відбуваються 2 речі: спершу Ваш клієнт «примушує» Вас зайти на SMTP-сервер (навіть якщо Ви вже зайшли на POP-сервер, чорт забирай!), а потім відправляє саму пошту (через SMTP-протокол).

Така система «набридла» до середини 1990-х, коли сервери почали використовувати протокол, який називається **POP-before-SMTP**: спочатку Ви відправляєте POP-серверу Ваше ім'я користувача і пароль, потім завантажуєте вхідну пошту, а потім SMTP-сервер перевіряє Вас за POP-сервером («З цим хлопцем все ОК?» «Так, я аутентифікував його.») та відправляє Ваші листи. Це непогана економія часу.

Варто пам'ятати одну важливу річ: не дивлячись на використання пароля для захисту, електронна пошта — не варіант для відправлення важливої/засекреченої інформації. Більшість POP-клієнтів і серверів вимагають, щоб Ваш пароль був переданий — незашифрованим — на поштовий сервер. Це не значить, що будь-хто, хто отримує листа від Вас, також отримує Ваш пароль; але це значить, що хтось з потрібними знаннями і інструментами може дізнатись Ваш пароль — та, як наслідок, вміст листів. (Щодо ідей з посилення захисту Ваших листів, див. Урок **9:** Безпека електронних листів.)

Вправи

4.5 Відправте собі листа з Вашого основного облікового запису на Ваш основний обліковий запис. Відправте того ж листа на той же обліковий запис з іншого облікового запису. Наскільки довго йшли 2 листи? Чи є між ними відмінності і чому?

4.6 Перегляньте один лист з того мільйона спаму, який засмічує Вашу скриньку. Чи можете Ви визначити, хто насправді відправляє Вам окремий спам? Чи є в них схована інформація? Якщо є, як хакер може це побачити?

4.7 Чи можете Ви затримати відправлення електронної пошти до певного часу або дня? Придумайте, як можна використати цю особливість, щоб пожартувати над своїми друзями?

IRC

IRC (Internet Relay Chat) — чудове місце, щоб побачити всю принадність неконтрольованості Інтернету в найкращому вигляді. Або в найгіршому. В IRC будь-який учасник, у якого є що сказати, отримує можливість це висловити. IRC також відомий як **Usenet** або групи новин. Кожна група новин має свою назву, під-назву, під-під-назву і т. д.

Цілком ймовірно, Ви вже знайомі з чатами. IRC — це як чат, тільки без правил мережевого етикету, і достатньо часто без модераторів. У каналі IRC Ви можете знайти саме те, що шукаєте, або те, про що ніколи не знали.

Всі правила, які Ви чули про чати, можна застосувати до каналів IRC. Нікому не кажіть Ваше справжнє ім'я. Не давайте номер Вашого мобільного телефону, адресу проживання або номери кредиток. Але отримуйте задоволення! У той же час будьте обережні з наявним контентом. Не все в Інтернеті нешкідливе і не всі люди в Інтернеті хороші.

IRC не є безпечним, і все, що Ви пишете, передається відкритим текстом від одного IRC-сервера до іншого. Ви можете створювати закриті розмови з іншими учасниками IRC, але повідомлення будуть передаватися також у відкритому вигляді. Використання ніка (nickname) забезпечить Вам лише незначну конфіденційність. Якщо Ви плануєте проведення якихось шкідливих або сумнівних дій, не використовуйте один і той самий нік для всіх облікових записів. Використовуючи один нік, Ви даєте чудовий спосіб себе вистежити поліції. Або менш приємним особам.

Теми називаються «каналами». Оскільки в світі існують тисячі каналів, ми даємо Вам URL, в якому перераховані багато з них для подальшого Вашого дослідження до втрати Вами глузду:

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

Якщо у Вас виникли питання щодо повідомлень, які написали інші учасники, Ви можете повідомити про них модератору (якщо він є) або «штовхнути» (**bump**) учасника з цього каналу. Якщо Ви не бажаєте когось чути, Ви завжди можете помістити учасника в «чорний список» і Ви не будете бачити його повідомлення. Можливо, ця тема і так Вам не підходить.

Вправи

4.8 Знайдіть 3 IRC-канали, де обговорюються питання безпеки. Як Ви приєдналися до загальної дискусії? Що Вам потрібно зробити, щоб створити закриту (приватну) розмову з кимось з учасників?

4.9 Який порт використовує IRC?

4.10 В IRC можливий обмін файлами. Як це можна здійснити? Чи хотіли б Ви обмінюватися файлами в IRC?

4.11 Які основні відмінності MIME і SMIME? Коли Ви бачите "S" в абревіатурі, чи означає це щось для Вас як для людини, що схильна до Безпеки (натяк)?

FTP

Старий добрий **FTP (File Transfer Protocol,** протокол передачі файлів**)** зазвичай запущений на 20 і 21 портах. Вгадайте для чого? Це дозволяє Вам обмінюватися файлами між двома комп'ютерами. У той час, як протокол може бути використаний для приватних передач, оскільки в ньому не виконується шифрування, він більш широко використовується для безкоштовних, анонімних FTP-серверів, які пропонують вільний доступ до файлів, на зразок ISO для нових крутих дистрибутивів Linux.

Анонімний FTP колись був єдиним способом для комп'ютерних користувачів обмінюватися файлами по Інтернету. У той час як існує безліч анонімних FTP- серверів для нелегального розповсюдження файлів (що є ефективним методом поширення «цифрових захворювань» (binary disease)), ще більше серверів використовують для легального розповсюдження файлів і програм. Користуючись пошуковою системою, можна знайти сервери, які пропонують анонімні FTP-сервіси. Але пам'ятайте: FTP-логіни передаються у відкритому вигляді. Так, навіть ім'я користувача та пароль. Існує Secure FTP (SFTP), але повсюдно він не використовується.

Більшість анонімних FTP-серверів дозволяють отримати доступ до файлів за допомогою протоколу FTP через браузер. Є також деякі дійсно гарні FTP-клієнти, які працюють як програми управління файлами. Як тільки Ви увійшли на FTP-сервер, Ви можете переміщувати файли до себе на комп'ютер, наче Ви локально переміщуєте файли на своєму ПК. Хіба що для FTP потрібно трохи більше часу для завантаження кожного файлу на Ваш комп'ютер, в основному тому що сервер FTP може бути розташований на іншій стороні планети.

Вправи

4.12 Windows, OSX і Linux поставляються з базовим консольним FTP-клієнтом; для отримання доступу до нього відкрийте командний рядок або термінал і введіть:

```
ftp
```

На вкладці `ftp>` Ви можете написати `help` для отримання списку доступних команд.

```
ftp> help
```

Допускається скорочення команд при введенні. Набір 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	

Базові команди:

Підключення до FTP-серверу *ftp.domain.name*:

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

Список вмісту віддаленої робочої папки:

```
ftp> ls
```

або

```
ftp> dir
```

Перехід на папку *newdir*:

```
ftp> cd newdir
```

Скачати файл *filename* з віддаленого комп'ютера на локальний комп'ютер:

```
ftp> get filename
```

Скачати декілька файлів file1, file2 і file3 з віддаленого комп'ютера на локальний комп'ютер (також Ви можете використовувати шаблони для завантаження декількох файлів з тим же суфіксом або взагалі всіх файлів у папці):

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

Завантажити файл *filename* з локального на віддалений комп'ютер:

```
ftp> put filename
```

Відключитися від віддаленого FTP-сервера:

```
ftp> close
```

Закрити локальний FTP-клієнт:

```
ftp> quit
```

FTP-сесія крок за кроком

Для підключення до анонімного FTP-сервера відкрийте локальний FTP-клієнт:

```
ftp
```

Використайте команду open для підключення до сервера. Команда:

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

з'єднає Ваш FTP-клієнт з анонімним FTP-сервером anon.server. Звичайно, Вам потрібно підставити ім'я реального сервера.

Коли віддалений FTP-сервер встановить з Вами з'єднання, він повідомить Ваш локальний клієнт, а потім запитає ім'я користувача:

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

Для більшості анонімних FTP-серверів в якості імені користувача Ви повинні ввести *anonymous* (або *ftp*). Віддалений FTP-сервер підтвердить, що Ви зайшли як анонімний користувач, і дасть Вам інструкції, що використовувати в якості пароля.

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

У більшості випадків сервер не перевіряє правильність введення адреси пошти, яка введена в якості пароля, що не завадить Вам отримати доступ до сервісу, якщо Ви неправильно ввели адресу. Це вважається порушенням мережевого етикету, але насправді це необхідно: не вказуйте свою реальну адресу електронної пошти! Після того, як Ви ввели пароль, віддалений сервер відправить вітальне повідомлення на локальний комп'ютер.

```
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.
```

Тепер Ви можете використовувати команди ls, dir, cd і get для завантаження файлів з сервера до себе на ПК.

Вправи

4.13 Використовуючи наведені приклади, знайдіть та скачайте файл з анонімного FTP-сервера.

4.14 Використовуючи браузер і пошукову систему, знайдіть анонімний FTP-сервер, на якому є копія Аліси в Країні Чудес, а потім, використовуючи командний рядок FTP-клієнта — НЕ браузер — скачайте файл.

4.15 Які гарні FTP-клієнти Ви знаєте? Чи можуть вони автоматизувати консольне введення і надати зручний графічний інтерфейс? Чи втрачаєте Ви яку-небудь функціональність у порівнянні з командним рядком?

4.16 Чи може Ваш ПК стати FTP-сервером?

Telnet та **SSH**

Telnet дозволяє локальному користувачеві відправляти безліч різних команд на віддалений комп'ютер. Локальний користувач може задавати команди віддаленому комп'ютеру, виконувати різні дії і отримувати дані на локальний комп'ютер, майже як якби він сидів за віддаленим комп'ютером. **Secure Shell (SSH, «**безпечна оболонка**»)** призначений для захищеної шифрованої заміни відкритого тексту telnet.

Більшість версій Windows, OSX і Linux містять клієнт telnet у форматі командного рядка; для отримання доступу до нього, відкрийте командний рядок або вікно терміналу і введіть:

```
telnet
```

Для доступу до сервера telnet Вам необхідні обліковий запис і пароль, встановлені для Вас адміністратором сервера, оскільки програма telnet дозволяє виконувати багато різних дій, і деякі з них можуть серйозно вплинути на безпеку віддаленого комп'ютера.

Telnet раніше використовувався для того, щоб адміністратори комп'ютерів могли дистанційно керувати серверами і забезпечувати підтримку користувачів на відстані. Цією послугою в Інтернеті зараз майже не користуються.

Telnet також може бути використаний для ряду інших завдань, таких як відправлення та отримання електронної пошти та перегляд вихідного коду веб-сторінок (хоча telnet, мабуть, найскладніший спосіб виконання цих завдань). Використання Telnet законне, але він може використовуватися в незаконних або аморальних цілях. Ви можете використовувати telnet для перевірки електронної пошти, а також переглядати не тільки тему повідомлення, а й перші кілька рядків, що дозволить Вам вирішити, чи слід видаляти його, не завантажуючи повідомлення повністю.

Якщо Ви збираєтеся використовувати SSH, переконайтеся, що Ви використовуєте останню версію, оскільки старі версії мають різні вразливості, і багато автоматичних сканерів вразливостей постійно шукають їх в Інтернеті.

Гра почалась: Командуй мною

Темний екран мерехтів перед товстими окулярами дідуся, курсор моргнув в нетерпінні, чекаючи команди. Його сиве рідке волосся ліниво оточувало його зморшкувату голову, дідусь стукав по клавіатурі. Джейс дивилася на безшумного піаніста, що грає на клавіатурі свого комп'ютера, тук, тук, тук, тук. Він повернувся, щоб поглянути в молоді очі Джейс, і посміхнувся їй. «Джейс, я збираюся показати тобі новий світ. Пристебни ремені безпеки», — він підморгнув восьмирічній дівчинці.

Джейс, сидячи в комп'ютерному кріслі, ледь діставала ногами до підлоги, а її дідусь сидів навпроти екрану комп'ютера. Вона чула довгий низький гудок, що виходив з невеликої коробки, що стояла поруч. Біла коробка загорілася зеленими і червоними вогнями, звук, що лунав з неї, став схожий на звук качки, наляканої сміттєвозом. Дідусь схвильовано підняв брови і вдумливо втупився на чорний екран перед ним. Качка замовкла, і всі вогні засвітилися зеленим на телефонній коробці.

Дідусь сказав : «Дивись.»

Зазвичай, коли дідусь каже «Дивись» — то варто очікувати, що щось вибухне або задимиться. Так чи інакше, «дивись» означало, що бабуся буде злитися через його чергову витівку. Джейс подобалося чути ці слова, бо це віщувало початок захоплюючої пригоди.

Екран комп'ютера вийшов із сплячого режиму і вивів банер ASCII-тексту, що оточував слова «Ласкаво просимо до Cline's Bulletin Board System (BBS).» «Ми всередині!» — Дідусь заплескав і спробував «дати п'ять» восьмирічній Джейс. Їх руки розминулися на кілька дюймів, і він мало не вдарив дівчинку по обличчю. Вона засміялася і дідусь також.

Вони обоє подивилися на клавіатуру і на екран комп'ютера. Дідусь схрестив пальці, поки Джейс чесала потилицю, намагаючись з'ясувати, що відбувається. Дідусь почав вводити команди на безшумному піаніно, опустивши голову вниз, як стерв'ятник, який вишукує жертву. Голову вгору, голову вниз, голову вгору, голову... Ой. Він відкинувся на спинку стільця. Дідусь забув щось дуже важливе.

Він зробив паузу і заговорив як вчитель: «Джейс, вибач, я забув розповісти тобі, що тут відбувається. Зараз я підключений до іншого комп'ютера через нашу телефонну лінію. Ця гучна штука он там називається «Модем»; його робота полягає в перетворенні цифрових сигналів в аналогові і навпаки.» Джейс вже багато знала про телефонні системи завдяки любові дідуся до проведення експериментів за будь-якої можливості. 48 вольт при нормальному використанні і 90 вольт під час повідомлення телефонного дзвінка; вона знала більше, ніж будь-який телефонний технік. Стара звичайна телефонна система (або POTS) вже стала предметом жартів між Джейс і її дідусем. Бабуся не розуміла цього гумору, що робило його ще більш смішним.

Телефонні лінії можуть бути використані зацікавленою стороною, але це можна виявити за допомогою регулятора напруги. Відбудеться стрибок напруги і вона буде постійно трохи підвищеною, якщо хтось спробує прослухати лінію. Джейс думала, що дідусь любив свій вольтметр більше, ніж бабусю; він ніколи не виходив з дому без нього. Дідусь зайшов так далеко, що назвав його «Валері». Валері-вольтметр. Це був його найкращий друг, не враховуючи Джейс.

Джейс дивилася на дідуся, її очі світилися від цікавості, вірніше їй цікаво було послухати лекцію про перехід від аналогової до цифрової модуляції з перетворенням звуку в цифровий сигнал. Це в основному саме те, що робить модем. Дідусь продовжував свою лекцію для маленької студентки: «Комп'ютер, до якого я підключений, дозволяє мені підключитися до інших комп'ютерів і отримати послуги, які вони надають.» Дівчинка вловила слово, яке вона не чула раніше — «послуги ».

«Дідусь, що ти маєш на увазі під словом «послуги»?» — запитала дівчинка, очікуючи відповіді, якимось чином пов'язаної з фаст-фудом. «Гарне питання, моя дорога,» — дідусь чекав від Джейс подібне питання. «Мій комп'ютер підключений до мережі комп'ютерів, і в мене є можливість підключення до інших комп'ютерів по всьому світу,» — з радістю відповів він. «Цей модем дозволяє мені розмовляти з цими іншими комп'ютерами, які пропонують доступ до файлів, інформації, спілкування з людьми та інші чудові послуги,

такі як File Transport Protocol, Usenet, IRC, Telnet та електронна пошта.»

Джейс не задовольнила ця відповідь, і це призвело до набагато більшої кількості питань, які посипалися на дідуся. Вона поповнила свій запас питань і почала «обстріл»: «Що таке File Transport Protocol? Що таке MIC? Де знаходиться Telnet? Чи потрібні для електронних листів спеціальні марки? Якого вони кольору в цифровому світі? Хто придумав Usenet? Чому вони називають це Email? Чи знає бабуся про ці послуги? Чому вони називаються послугами? Звідки беруться діти? Звідки взялося желе?»

Дідусеві довелося закрити вуха, щоб захистити себе від натиску питань. «Почекай, почекай, почекай, не так швидко.»

Гра скінчена

DNS

Коли Ви хочете подзвонити другу, Вам потрібно знати правильний номер телефону; коли Ви хочете підключитися до віддаленого комп'ютера, Ви також повинні знати його номер. Можливо, Ви пам'ятаєте з попередніх уроків, що для комп'ютерів в Інтернеті цей номер — це IP-адреса.

Комп'ютерам дуже легко працювати з IP-адресами, але люди вважають за краще використовувати імена, в цьому випадку імена доменів. Наприклад, для підключення до веб-сайту Hacker Highschool введіть www.hackerhighschool.org в адресному рядку веб-браузера. Проте, веб-браузер не може використовувати це ім'я для підключення до сервера, на якому розміщений сайт Hacker Highschool — йому потрібна IP-адреса. Це означає, що локальний комп'ютер повинен мати деякі засоби перетворення доменних імен в IP-адреси. Якби в Інтернеті були тільки сотні або навіть тисячі комп'ютерів, то можна було б створити просту таблицю (файл хостів), що зберігається на комп'ютері, для пошуку цих адрес. Однак, на краще або на гірше, крім існування мільйонів адрес комп'ютерів в Інтернеті, залежності між доменними іменами та IP-адресами постійно змінюються.

Domain Name Service (DNS, сервіс доменних імен**)** використовується для динамічного перетворення доменних імен в IP-адреси (і навпаки). При введенні доменного імені www.domainname.com в адресний рядок Ваш веб-браузер з'єднується з DNS-сервером, обраним Вашим провайдером. Якщо у цього DNS-сервера в базі даних є www.domainname.com, то він повертає IP-адресу Вашому комп'ютеру, дозволяючи Вам підключитися.

Якщо у Вашого DNS-сервера в базі даних немає www.domainname.com, то він надсилає запит на інший DNS-сервер і вони будуть продовжувати відправляти запити іншим DNS-серверам, поки не знайдуть потрібну IP-адресу або не встановлять, що домен має невірне ім'я.

Вправи

4.17 Відкрийте вікно командного рядка і визначте IP-адресу Вашого комп'ютера. Яку команду Ви використовували? Яка у Вас IP-адреса?

4.18 Визначте IP-адресу Вашого DNS-сервера. Яку команду Ви використовували? Що таке IP-адреса DNS-сервера?

4.19 Пропінгуйте www.isecom.org. Чи отримуєте Ви відповідь? Яка IP-адреса відповідає на пінг?

4.20 Чи можете Ви змінити DNS-сервер, яким користується комп'ютер? Якщо так, то змініть конфігурацію комп'ютера так, щоб він використовував інший сервер DNS. Пропінгуйте www.isecom.org знову. Ви отримали ту саму відповідь? Чому?

DHCP

DHCP (**Dynamic Host Configuration Protocol,** протокол динамічної конфігурації хоста**)** дозволяє серверу локальної мережі роздавати IP-адреси в мережі. Серверу надається блок IP-адрес для використання. Коли комп'ютер приєднується до мережі, він отримує IP-адресу. Коли комп'ютер виходить з мережі, його IP-адреса стає доступною для використання іншим комп'ютером.

Такий підхід зручний для великих мереж комп'ютерів, оскільки немає необхідності для кожного комп'ютера мати індивідуально призначену, статичну IP-адресу. Замість цього використовується сервер DHCP. Коли новий комп'ютер підключається до мережі, перше, що він робить, — це запит IP-адреси з сервера DHCP. Як тільки йому була призначена IP-адреса, комп'ютер отримує доступ до всіх послуг в мережі.

Тепер подумайте про наступне. Більшість бездротових мереж пропонують DHCP; це означає, що будь-хто може отримати IP-адресу в цій підмережі. Якщо Ви працюєте в кафе, то це саме те, що Вам підходить, але якщо Ви працюєте в безпечному офісі, то, ймовірно, Ви захочете використовувати фіксовані IP-адреси. Можливі різні варіанти.

З'єднання

У старі недобрі часи комп'ютери підключались до Інтернету через модем. Модеми перетворюють біти у звуки й навпаки; ці дві операції називаються модуляцією та демодуляцією, звідси й назва. Швидкість модему вимірюється в бодах **(baud)** та в бітах за секунду (**bps**). Вища швидкість передачі даних зазвичай означає більш високе значення bps, але Ви також повинні враховувати те, що Ви плануєте робити. Існують певні застосунки — такі як telnetting в **Multi-User Dungeons (MUDs, «**світи, що розраховані на багато користувачів**»)** — для яких двадцятирічний модем в 300 бод все ще прийнятний (за умови, що Ви не занадто швидко друкуєте), в той час як застосунки з більшою пропускною здатністю (як, наприклад, передача потокового відео) часто можуть навантажувати навіть найпотужніший кабель або DSL-з'єднання.

ISP

Процес підключення до Інтернету не такий простий, як здається. Вам потрібно отримати доступ до сервера, який підключить комп'ютер до Інтернету. Сервер виконує усю важку роботу, і він постійно увімкнений. Сервер керується провайдером **(ISP)**.

Інтернет-провайдер має постійну точку присутності PoP (point-of-presence) в Інтернеті, а також сервери, які надають послуги, якими Ви можете користуватися. Але Ви можете запустити ці послуги й самостійно. Наприклад, Ви можете запустити поштовий сервер на локальному комп'ютері, але для цього потрібно, аби Ваш комп'ютер був постійно увімкнений та підключений до мережі, очікуючи сеанси обміну інформацією. Провайдер об'єднує зусилля великої кількості користувачів, тому поштовий сервер працює увесь час. Комп'ютери провайдера використовують високошвидкісне з'єднання для підключення до точки доступу до мережі (**Network Access Point, NAP**). Потім ці точки доступу зв'язуються одна з одною через надшвидкісне з'єднання, яке називається опорною мережею **(backbone)**. Усі ці речі разом складають Інтернет.

Старі звичайні телефонні служби

Старі звичайні телефонні служби **(Plain old telephone service, POTS)** були колись методом доступу до Інтернету, яким користувалися найбільш широко. Його основним недоліком є мала швидкість, але в багатьох випадках це компенсується його доступністю. Більшість національних Інтернет-провайдерів мають велику кількість місцевих номерів доступу, і майже всі як і раніше мають домашню телефонну лінію. У теорії, якщо у Вас акустичний модем та повна кишеня здачі, Ви можете підключитися практично з будь-якого громадського телефону-автомату (якщо Вам вдасться його знайти). Хоча сумніваємося, що Вам дійсно хотілось би так зробити.

Телефонне з'єднання повільне. Найбільш швидкі телефонні модеми розраховані на швидкість 56600 біт за секунду (bps). Що, втім, не відповідає дійсності. Через обмеження потужності, реальна швидкість завантаження складає приблизно до 53000 біт за секунду, а дійсна швидкість, як правило, значно нижча. Ці показники важко порівняти з DSL або кабельними модемами.

Тим не менш, телефонний зв'язок широко доступний, відносно дешевий (а іноді й безкоштовний). Ви би не захотіли торгувати піратськими фільмами по телефонній лінії, тому що це аморально, незаконно та займе Вашу телефонну лінію на всю ніч й, можливо, на всі вихідні, але Ви, безумовно, можете відправляти електронні текстові листи бабусі. Та якщо Ви використовуєте telnet, Ви навіть здатні зробити це за допомогою старенького комп'ютера на DOS-і, який Ви витягли з підвалу.

DSL

Цифрова абонентська лінія (**Digital Subscriber Line**) — це спосіб відправлення великих обсягів інформації по дротам, які вже існують для телефонної лінії. Головною перевагою перед стандартною телефонною службою є те, що цей спосіб набагато швидший, ніж аналогові модеми, і він забезпечує постійне з'єднання. Крім того, DSL дозволяє здійснювати і приймати регулярні телефонні дзвінки, поки Ви підключені до Інтернету. Її головним недоліком є те, що її доступність обмежена тим, наскільки близько Ви знаходитесь до комутаційного обладнання телефонної компанії — якщо Ви мешкаєте дуже далеко вниз по лінії, то Вам не пощастило.

Кабельні модеми

Кабельні шлюзи не використовують традиційні телефонні лінії для підключення до Інтернету. Замість цього вони застосовують коаксіальний кабель (або волоконно-оптичні лінії, якщо Вам дійсно пощастило), наданий кабельною компанією. Як і DSL, кабельні шлюзи дозволяють Вам здійснювати звичайні телефонні дзвінки, поки Ви підключені до Інтернету, і вони забезпечують постійне з'єднання, але кабельні шлюзи, як правило, швидкіші, ніж DSL.

Кабельні шлюзи мають декілька недоліків. По-перше, кабельний шлюз є загальним ресурсом, тому швидкість підключення знижується, коли інші користувачі підключаються до того ж кабелю. По-друге, кабельний доступ є тільки в районах, де кабельні компанії встановили необхідну проводку. Та найсерйозніший недолік полягає в тому, що будь-який трафік, що проходить через кабель, може переглянути будь-який інший користувач, підключений до нього. Це означає, що якщо Ви при підключенні комп'ютера до кабельної лінії не увімкнете брандмауер, усі інші підключені комп'ютери зможуть бачити Ваш комп'ютер та всі його файли. Ви справді бажаєте поділитися з іншими інформацією про свій банківський рахунок?

Wimax

Wimax — це бездротовий спосіб підключення, який зазвичай конкурує з DSL. Він використовується в місцях, де дротову інфраструктуру дуже дорого або неможливо встановити. На рівень сигналу можуть впливати будівлі, дерева або інші великі об'єкти. Деякі версії використовують фіксовану точку доступу, а інші дають Вам мобільний доступ на дійсно великому просторі.

Wifi

Wifi не є способом підключення до провайдера, але це розповсюджений засіб для підключення до Інтернету вдома або у комерційних закладах, таких як торговельні центри чи кафе. Зараз більшість смартфонів і всі ноутбуки використовують Wifi, тому це улюблена мішень для зловмисників. Уявіть себе голим в переповненій кімнаті, коли Ви користуєтесь суспільним Wifi: прикрийтесь, переконайтеся, що ніхто не дивиться на Вас, але кожний бажає поглянути. Звісно, зараз Вам закортіло прочитати урок з безпеки у бездротових мережах, так?

Вправи

4.21 Який тип підключення до Інтернету у Вас вдома? Як Ви про це дізналися? Й найголовніше:

4.22 Хто може бачити Вас в цій мережі? (Як Ви можете це взнати?)

4.23 Яка швидкість Вашого з'єднання? Чи можете Ви покращити її без звернення до Інтернет-провайдера?

4.24 Які додаткові послуги надає Ваш провайдер? Ми вже говорили про послуги, Ваш провайдер може підтримувати декілька.

4.25 Які послуги Ви можете надати зі свого комп'ютера?

Пожива для розуму: Граючи з **HTTP**

HTTP (скорочення від Hypertext Transfer Protocol — протокол передачі гіпертексту) — розташований на вершині стеку TCP/IP й визначений у двох основних RFC:

- 1945 для 1.0 (починаючи з 0.9).

- 2616 для 1.1.

Є декілька істотних оновлень та відмінностей версій 1.0 та 1.1 відносно розширюваності (Extensibility), кешування (Caching), оптимізації пропускної здатності (Bandwidth optimization), керування мережевим з'єднанням (Network connection management), передачі повідомлень (Message transmission), захисту Інтернет-адрес (Internet address conservation), повідомлення про помилки (Error notification), безпеки, цілісності та аутентифікації (Security, integrity, and authentication), угоді про дані (Content negotiation) [3]. Відмінності між 1.0 та 1.1 корисні для отримання інформації про веб-сервер.

В цілому HTTP — це протокол без зберігання стану, у якому клієнт відправляє HTTP-запит на сервер, який, в свою чергу, відправляє HTTP-відповідь: в цьому складається сутність

парадигми запит/відповідь.

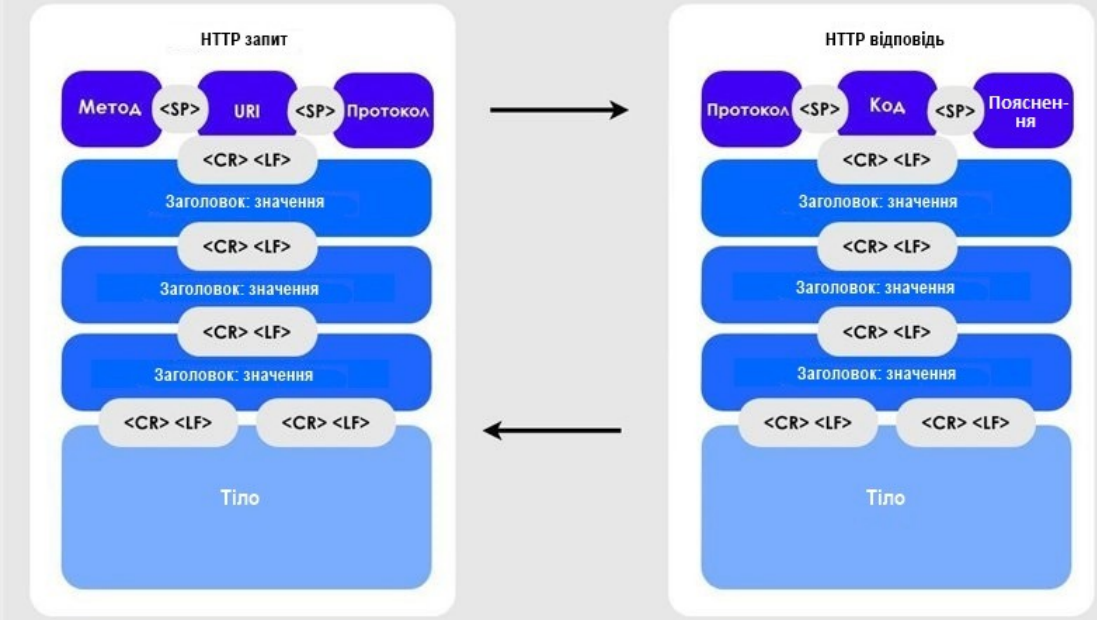

Рисунок **4.1:** HTTP

Як Вам вже, можливо, відомо, ми можемо отримати багато інформації, відправляючи команди на HTTP-сервер. Ми скористаємося декількома базовими мережевими утилітами:

- netcat: набір утиліт TCP/IP
- curl: набір утиліт HTTP
- проксі: такі, як OWASP ZAP або Burpsuite free

Сніффінг з'єднання між Вами та **HTTP**-сервером **HHS**

Використайте проксі для здійснення з'єднання через браузер. Перейдіть за посиланням http://www.hackerhighschool.org і перехопіть свій запит:

```
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
```

і відповідь:

```
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>
[...]
```

Вправи

4.26 Визначте частини запитів через проксі, користуючись діаграмами.

4.27 Чи є у заголовках яка-небудь цікава інформація?

Ваше перше з'єднання, налаштоване вручну

Netcat можна використовувати для з'єднання з веб-сервером, користуючись налаштуваннями портів хоста.

Почніть, ввівши команду:

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

Потім двічі натисніть клавішу Enter.

```
GET / HTTP/1.0
```

Сервер відповість:

```
<!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" />
```

Як Ви бачите, створюється враження, що сторінка отримана з isecom.org, а не з hackerhighschool.org. Чому?

Одне з припущень полягає в тому, що один й той самий хост обслуговує й сайт HHS, й сайт ISECOM. Чи можливий такий варіант?

Щоб розібратися з цим, визначте IP-адресу 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 зараз для www.isecom.org:

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

Non-authoritative answer:
Name: isecom.org
Address: 216.92.116.13
```

Та сама IP-адреса! За допомогою netcat можна відобразити хост, власноруч додавши заголовок Хост (Host) й використовуючи 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 запиту

Ще однією частиною HTTP-запиту, яку можна модифікувати, є його метод. Найчастіше веб-застосунки використовують GET і POST запити, але інші протоколи запитів також можуть бути активні на веб-сервері або сервері застосунків. Серед інших методів, які часто використовуються, можна виділити наступні:

- **OPTIONS** — використовується для визначення параметрів запиту, які підтримуються. Якщо у Вас працює веб-сервер, то пам'ятайте, що надання цієї інформації може привести до різних проблем.
- **GET** — використовується для отримання інформації безпосередньо через URL, наприклад:
  `http://www.usairnet.com/cgi-bin/launch/code.cgi?Submit=Go&sta=KSAF&state=NM`
  Бачите фрагмент рядка після знаку питання? Це дані запиту. Передавати дані таким засобом ризиковано, оскільки їх всі бачать й їх легко змінити.
- **HEAD** — використовується аналогічно методу GET, але сервер не повертає фактичну сторінку.
  Цей метод можна використовувати для визначення варіантів доступу, оптимізації споживання пропускної здатності й – у деяких випадках – обходу засобів керування доступом. Насправді, деякі реалізації ACL перевіряють тільки GET запити. У таких випадках Ви можете виявити вразливість.
- **POST** — використовується для передачі даних веб-застосункам — подібно методу GET — але дані включаються у тіло запита, принаймні, хоча б трохи приховано.
- **PUT** — використовується для розміщенні ресурсів на веб-сервері або для їх оновлення. У багатьох випадках цей метод повинен бути заборонений або захищений засобами керування аутентифікації (Authentication Control). Інакше це може стати чудовою знахідкою для Вас.
- **DELETE** — використовується для видалення ресурсів з веб-сервера.
  Цей метод повинен бути заборонений або захищений засобами керування аутентифікації (Authentication Control) (аналогічно PUT, який представлений вище).
- **TRACE** — використовується на прикладному рівні як зворотня петля (loopback), яка відображає повідомлення.
  Цей метод налагодження повинен бути заборонений, особливо у виробничому середовищі, оскільки він розкриває конфіденційну інформацію й представляє собою вразливість, яка може використовуватися в експлойтах міжсайтового скриптингу.
- **CONNECT** — для використання веб-сервера як проксі.
  Цей метод повинен бути заборонений або захищений засобами керування аутентифікації (Authentication Control), оскільки він дозволяє іншим здійснювати

з'єднання зі сторонніми сервісами, використовуючи IP проксі.

Також враховуйте, що протоколи, які засновані на HTTP, можуть додавати й інші методи (як, наприклад, WebDAV). Ви можете змінювати метод запиту з метою перегляду відповідей сервера (які в чомусь, можливо, становлять інтерес), запиту відомих методів, а також перегляду «реакції» на довільно обрані слова.

Запит **OPTIONS**

Почніть сеанс netcat як зазвичай:

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

Але цього разу не натискайте клавішу Enter двічі. Замість цього введіть наступний рядок:

```
OPTIONS / HTTP/1.1
```

і Ви отримаєте відповідь, схожу на наступну:

```
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
```

Запит **HEAD**

На цей раз, почавши сеанс, введіть 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
```

Дозвольте мені використати Вас як проксі: запит **CONNECT**

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

Вправа

4.28 Скористайтеся netcat (nc) для того, щоб випробувати всі перераховані вище методи запитів на мережевих серверах HHS або сервері, який запущений для тестування. Яку цікаву інформацію Вам вдалось виявити?

Складання сценаріїв **HTTP**-запитів за допомогою **curl**

У деяких випадках тестування веб-застосунків ґрунтується не тільки на відповідях веб-сервера, але й на роботі рівня (веб-)застосунків. Часто можна виявити вразливості веб-застосунків, змінюючи параметри GET і POST, cookies і значення заголовків. Корисною утилітою для bash-скриптингу є команда **curl** — це утиліта командного рядка для запитів веб-сторінок. Але у порівнянні з netcat, логіка роботи curl дещо інша.

Дана команда:

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

не те ж саме, що наступна:

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

Щоб переконатися в цьому, Ви можете ввести параметр -v для детального виводу:

```
# 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">
[...]
```

Як бачите, curl автоматично вибирає версію HTTP 1.1, додає заголовок хоста (host), клієнтського застосунка (user agent) і допустимих форматів ресурсу (accept). З цього випливає важливе правило для хакерів: знай те, з чим працюєш.

На щастя, curl — гарна утиліта, яку можна ретельно налаштувати, задаючи певні параметри.

Для перегляду усіх параметрів введіть `curl –help`.

Серед параметрів команд, схожих на вищенаведений приклад з netcat, можна виділити наступні:

- **-H** для додавання строки заголовка
- **-X** для вибору метода запиту (також відомого як Команда)
- **-d** для додавання POST даних
- **-i** для включення заголовків протоколу у вихідному ресурсі
- **-s** для включення «тихого» режиму, зручного для скриптингу

Використовуючи curl і трохи bash-скриптингу, Ви можете автоматизувати тестування веб-застосунків. Пошук цікавих HTTP-заголовків від сервера можна досить просто автоматизувати за допомогою curl та grep:

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

```
Server: Apache/2.2.22
```

Вправа

4.29 Доповніть вищенаведений сценарій для запиту інших HTTP-заголовків і потенційно корисної інформації.

Посилання та додаткова література

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/

Висновки

Всесвітня павутина — це значно більш широке поняття, ніж Інтернет: крім HTTP є багато інших видів служб. FTP, SSH, DNS, DHCP та багато інших «розкривають вікна» у комп'ютери інших користувачів — у тому числі, і в Ваш. Розуміння того, як Ви підключаєтесь до цих служб («через правильні канали» або якось по-іншому), є ключовим моментом для усвідомлення того, як Ви або Ваш комп'ютер можуть бути атаковані — або як самому провести атаку. Просто пам'ятайте про девіз: зламуйте все, але без шкоди іншим.

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