Hacker Highschool

SECURITY AWARENESS FOR TEENS



LEZIONE 4
GIOCHIAMO CON I DEMONI

Hacker Highschool
SECURITY AWARENESS FOR TEENS

ISECOM

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

ATTENZIONE

Il progetto Hacker Highschool è uno strumento di apprendimento e come tutti gli strumenti di apprendimento non è esente da pericoli. Alcune lezioni, se usate in modo improprio, possono causare danni fisici. Eventuali pericoli possono emergere anche in caso non si sia svolta una sufficiente ricerca in merito agli effetti di particolari tecnologie. Gli studenti che usano queste lezioni dovrebbero essere incoraggiati ad imparare, provare e testare. Ad ogni buon conto ISECOM non potrà essere ritenuto responsabile per un uso improprio di quanto esposto.

Le seguenti lezioni ed esercizi sono "open" e disponibili pubblicamente alle seguenti condizioni e termini stabiliti da ISECOM:

Tutti i contenuti del progetto Hacker Highschool vengono forniti per uso non-commerciale per gli studenti delle scuole elementari, scuole medie inferiori e scuole medie superiori sia per le istituzioni pubbliche che per quelle private, ammettendone l'uso per le esercitazioni a casa. Non è ammessa la riproduzione del materiale per la vendita. L'utilizzo del materiale presente in queste lezioni è consentito per i corsi di ogni tipo che prevedono il pagamento di una tassa/quota d'iscrizione o frequenza, previa acquisizione di regolare licenza. Sono soggetti a tale norma anche i corsi presso le università, campi estivi e tutto quanto sia inteso come formazione. Per acquistare una licenza è possibile visitare la sezione LICENSE della pagina web della HHS all'indirizzo web http://www.hackerhighschool.org/licensing.html.

Il progetto Hacker Highschool rappresenta lo sforzo di una comunità "open". Pertanto se trovi utile questo materiale ti invitiamo a supportarci tramite l'acquisto di una licenza, attraverso una donazione o una sponsorizzazione.

Indice

Hanno contribuito

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

Per la versione in lingua italiana

Raoul Chiesa, ISECOM (Coordinatore Team di lavoro edizione italiana)
Matteo Benedetti, Security Brokers SCpA
Ing. Selene Giupponi, Security Brokers SCpA
Francesco Mininni, Ing. PhD, Uff. E.I.
Riccardo Trifonio, Mar.Ca. CC
Dott. Sandro Bruscino, CISSP
Dott.ssa Sophia Danesino, I.I.S. "G.Peano" TO

ISECOM

Introduzione

Esistono migliaia di differenti lingue usate dall'uomo alcune delle quali presentano dozzine di dialetti. Puoi conoscere varie lingue, ma la probabilità che tu possa essere in grado di girare il mondo e parlare con tutti quelli che incontri è prossima allo zero.

Ok, potresti obiettare che la matematica è un linguaggio universae o che la musica parla a tutti, ma siamo realistici. Prova ad ordinare un bicchiere di soda con una scorza di limone e una pallina di gelato usando uno dei questi "linguaggi universali" e vedi quanto vai lontano.

Se ti capita di passare in un paese di cui non parli la lingua, ti prego di mandare un video ad ISECOM in cui usi delle cornamuse o un sassofono per ordinare una soda. Vogliamo veramente vederlo! Magari non ci interessa sentirlo, ma di sicuro vogliamo vederlo.

Ma ogni giorno milioni di persone comunicano tra loro usando un solo comune linguaggio su internet. Gli essere umani potranno non parlare tutti la stessa lingua; tuttavia, i nostri computer e le reti lo fanno.

Il modello che usiamo nelle attuali reti è il **modello client-server.** I computer fisici (**host** o **server**) offrono **servizi** (che in ambiente UNIX sono chiamati **daemons: disk access and execution monitors –** adesso vai a vantarti con qualcuno). Pensa ad un server web: ti serve una pagina web quando glielo chiedi. Nessun mistero.

Ma in realtà "tu" non hai richiesto quella pagina; lo ha fatto il tuo browser web, ed in questo senso è un **client** (o per l'esattezza, lo è il tuo computer). Allo stesso tempo, il tuo computer può essere anche un server. Questo è il bello delle reti: tu fai questo per me; io faccio questo per te.

Moltiplica questo modello un milione di volte, ed avrai ottenuto Internet. Considera questo: milioni di computer stanno offrendo un qualche tipo di servizio. Che cosa server per essere un client? Ed è possibile **sovvertire** tutto questo? (Vai a cercare questo verbo se non sei assolutamente certo di cosa significhi. Dopotutto questo è un corso sull'hacking).

Pronti o meno, tuffiamoci nell'argomento.

Servizi

Tu hai un computer, e sai che ci sono informazioni importanti su di esso, oppure potresti partecipare all'allucinazione collettiva secondo cui non hai nessun contenuto digitale di valore. Sai anche che altre persone, milioni di altre persone, hanno anchesse dei computer e che anche questi contengono informazioni interessanti, senza contare risorse utili quali processori, memoria, spazio di archiviazione e banda.

Ora, puoi supporre che tutte quelle persone, e quei computer, possano molto probabilmente contenere informazioni interessanti per qualcuno. L'unico problema è come avere accesso a tutte queste informazioni.

I computer comunicano tra loro in modo semplice attraverso delle porte, usando i protocolli di cui abbiamo parlato nella Lezione 3, ma quello che leggi non sono i reali flussi di dati binari che i computer si scambiano tra loro (a meno che tu non abbia una buona quantità di tempo libero). Il tuo computer ha bisogno di un modo per ricevere i dati, interpretarli per te e presentarteli in una forma che tu possa usare.

Il modo nel quale i computer trasferiscono i dati è attraverso i **servizi di rete**, o più semplicemente **servizi**. Questi servizi ti consentono di visitare pagine web, di scambiare email, chattare, e di interagire remotamente con altri computer. Questi servizi sono mappati su determinate porte.

Il tuo computer, il **computer locale**, utilizza programmi chiamati **client** per interpretare le informazioni che ricevi. Potresti ricevere informazioni da un server (che fornisce un servizio/esegue un demone), tramite una rete **Tor**, da **seeder Torrent** o attraverso reti **peer-to-peer**.

> Ovviamente anche il tuo computer a sua volta può fornire servizi ad altri computer, agendo da vero e proprio server o fornitore di servizi. Se nel tuo computer è presente del malware, potresti stare fornendo alcuni servizi di cui non sei al corrente.

Esempi di client includono i browser web, i client email, i programmi per chat, Skype, client Tor, client Torrent, RSS e così via. Queste sono le applicazioni a **livello applicazione** dello pila protocollare TCP/IP. A livello applicazione, tutti i dati che vengono trasmessi, incapsulati, criptati, decriptati, indirizzati e così via dagli strati inferiori vengono trasformati in modo che tu, l'utente, possa leggerli e comprenderli.

HTTP ed il Web

Quando parliamo di "Internet", molte persone pensano in realtà al **World Wide Web**. Il World Wide Web, o più semplicemente il **Web**, non è Internet, è solo una piccola porzione dei servizi disponibili. Di solito riguarda solo il visitare delle pagine web tramite un browser.

> A proposito, la vera Internet è composta da tutti i computer, i router, i cavi, le connessione ed i sistemi wireless che movimentano tutte le informazioni. Solo una frazione di questi riguarda il traffico web.

Il web utilizza il protocollo **HTTP** o **HyperText Transfer Protocol** e delle applicazioni (i client) chiamate **browser web** per accedere ai documenti sui **server web**. L'informazione dal computer remoto viene inviata al tuo computer locale usando il protocollo HTTP, solitamente sulla porta 80. Il tuo browser web interpreta quella informazione e la visualizza sul tuo computer.

Non tutti i browser sono uguali. Ognuno offre strumenti diversi e mostra i contenuti HTML in modo leggermente (o radicalmente) diverso. I problemi legati alla sicurezza ed alla privacy possono essere gestiti più o meno bene. Questo vuol dire che dovresti sapere cosa può fare o meno il tuo browser, e quali settaggi e plugin ti danno quel perfetto equilibrio tra sicurezza e privacy (a meno che non ti piacciano malware, pubblicità, spam e che il tuo vicino sappia che ti piace vedere i gattini giocare con le gelatine verdi).

La parte di **ipertesto** del protocollo HTTP si riferisce al modo non lineare in cui lo leggi. Normalmente leggi in modo lineare: pagina 1, quindi pagina 2; capitolo 1, poi il capitolo 2; lezione 1 quindi lezione 2 e così via. Un ipertesto ti consente di accedere ad una informazione in modo non lineare. Puoi saltare di argomento in argomento, mentre impari, poi torni indietro e magari vai a leggere un'altra informazione prima di finire l'articolo da cui sei partito. Questa è la differenza tra un ipertesto ed un testo semplice.

Nell'ipertesto, le parole e le idee si collegano non solo alle parole che le circondano direttamente, ma anche con altre parole, immagini, contenuti video ed audio. L'ipertesto non è presente solo nel Web. La maggior parte degli elaboratori di testi ti consente di creare in locale pagine in formato web, o HTML. Leggi queste pagine nel tuo browser allo stesso modo di quelle presenti sul web, solo che si trovano sul tuo computer e non su uno remoto.

È facile creare la tua propria pagina web. Il modo più semplice per farlo è di usare uno dei tanti elaboratori di testi come OpenOffice/LibreOffice Writer, Microsoft Word, o WordPerfect. Questi programmi ti consentono di produrre semplici pagine web, combinando testo, ipertesto ed immagini. Molte persone hanno realizzato pagine web in questo modo (o anche con semplici editor di testo come vi, presente sulla maggior parte delle piattaforme Unix). Altri editor di testi includono Microsoft Notepad, Notepad++, SciTe, emacs ed altri.

Ma queste pagine non sono "flashy". Flashy vuol dire **CSS** e **script** ed animazioni. Puoi spendere molto denaro su applicazioni per lo sviluppo di pagine web accattivanti. Queste applicazioni ti consentono di creare interessanti effetti sulla tua pagina web, ma sono più difficili da usare. Anche se, nel complesso rendono tutto più semplice. L'alternativa a basso costo è di prendere uno degli editor di testo personalizzati per lavorare su HTML e linguaggi di scripting, imparare la sintassi HTML e di scripting scriversi interamente a mano la propria pagina web.

Una volta che le pagine sono pronte, avrai bisogno di un computer su cui metterle, se vuoi che anche gli altri le vedano. Gli **Internet Service Provider (ISP)** forniscono servizi di **web hosting** sui loro server.

Puoi anche far girare il tuo server web da casa, sul tuo computer, ma ci sono tutta una serie di problemi. Le informazioni saranno accessibili solo quando quel server sarà acceso, operativo e con una connessione funzionante. Quindi se vuoi tenere in funzione un server web nella tua camera, dovrai lasciare il computer sempre acceso; devi essere sicuro che il programma che fa girare il server web funzioni correttamente tutto il tempo (il che

include risolvere problemi hardware, tenere sotto controllo virus, worm ed altri attacchi, e dover fare i conti con gli immancabili bug ed errori del software stesso); inoltre devi tenere una connessione aperta su internet, che deve essere stabile e veloce. Gli ISP fanno pagare un extra per una maggiore velocità di upload ed un indirizzo IP statico, e questo è il motivo per cui ci si affida ad altri per fare tutto questo.

Chi fornisce servizi di hosting tiene le tue pagine sui suoi computer. È conveniente lasciare che siano i loro server ad essere attaccati invece dei tuoi. Una buona compagnia di hosting avrà molteplici server, ridondati, ed una politica di backup ricorrente, in modo da evitare che il tuo sito scompaia per un semplice problema hardware; lo staff di supporto manterrà i server in funzione nonostante attacchi e bug del software; e varie connessioni ad Internet forniscono una certa garanzia di non interruzione. Quindi tutto quello che devi fare è disegnare la tua pagina, caricarla sul server della compagnia di hosting, spegnere il tuo computer ed andare a dormire. La tua pagina sarà raggiungibile da tutto il mondo, fintanto che continuerai a pagare per il servizio.

È possibile anche trovare delle aziende che offrono gratuitamente i servizi di hosting web. Alcune di queste vengono finanziate dalle pubblicità, il che vuol dire che chiunque voglia visitare la tua pagina dovrà prima guardare lo un qualche spot pubblicitario. Ma loro non dovranno comprare niente, e tu non dovrai pagare niente.

Esercizi

4.1 Una pagina web è un semplice testo che dice al browser dove immagini, video ed altre cose dovrebbero essere. Puoi farti un'idea di come si presenta guardando il sorgente della pagina. Apri il tuo browser preferito, indirizzalo su ISECOM.ORG e carica la pagina. Ora osserva il sorgente. Vedrai alcuni tag contenenti la parola "meta". Ad esempio, il primo è meta-charset="utf-8". Che cosa vuol dire? A cosa serve?

4.2 Trova altri 3 meta tag e spiega a cosa servono. Potresti dover cercare sul web per capirne il significato quindi pensa attentamente a quali parole chiave userai nella tua ricerca per essere sicuro di ricevere le risposte giuste.

4.3 Salva il codice sorgente della pagina ISECOM.ORG. Trascinalo nel browser. Cosa è cambiato? Perché pensi che sia cambiato?

4.4 Apri il codice sorgente della pagina ISECOM.ORG in un editor di testo e vedrai che sono solo parole e numeri. Qualunque cosa cambierai o scriverai nella pagina cambierà come la pagina verrà visualizzata quando la salverai e la trascinerai nel browser. Cancella qualcosa e vedrai che scomparirà. Cambia le parole e vedrai che compariranno come le hai scritte. Ora cancella tutto il resto dalla pagina ed aggiungi il tuo nome così che venga mostrato più largo e marcato delle altre parole. Prova. Salva. E trascinalo nel browser e vedi se hai avuto successo. No? Continua a provare!

*Leggi **Nutrite la Vostra Mente: Giocare con l'HTTP** alla fine di questa lezione per approfondire l'argomento.*

Email – SMTP, POP e IMAP

Il secondo aspetto più visibile di Internet è probabilmente la posta elettronica. Sul tuo computer utilizzi un client email, che si connette ad un server di posta. Quando configuri il tuo account email, utilizzi un nome univoco nel seguente formato **user@domain** e devi creare una password.

Ci sono due componenti: **SMTP (Simple Mail Transfer Protocol)**, che *invia* la posta, ed il server di posta, **POP (Post Office Protocol)** o **IMAP (Internet Message Access Protocol)**, che *recupera* la tua posta.

Il protocollo SMTP (te lo ricorderemo ancora) è usando per *inviare* la posta. SMTP definisce i **campi** in una email, includendo i campi FROM, TO, SUBJECT, CC e BODY. La vecchia versione di SMTP non richiede una password ed invia tutto in chiaro; chiunque può leggere la tua posta. Questo poteva non essere un problema ai tempi in cui il protocollo venne progettato ed Internet era un piccolo mondo abitato da persone gentili. Ma ha lasciato la possibilità ad ogni utente di inviare **spam** e di fare altre spiacevoli cose come lo **spoof** delle email, che in pratica vuol dire mentire (spoofing) riguardo all'indirizzo del mittente. Praticamente tutti i server di posta moderni utilizzano Secure SMTP, che vuol dire che devi dare prova della tua identità prima di poter inviare una email.

Nelle prossime lezioni ti mostreremo come funziona lo spoofing e come fare a riconoscerlo negli header di una email. Queste semplici nozioni possono trasformarti da pecora in lupo in modo incredibilmente veloce.

POP3 (Post Office Protocol version 3) è un protocollo "conserva e scarica". Il server riceve la tua posta e la conserva per te, fino a che non ti colleghi e la scarichi. La tua posta in uscita viene inviata tramite SMTP. Questo è un buon approccio per gestire le email se hai una connessione dial-up, dal momento che richiede meno tempo per inviare e ricevere la posta, e la puoi leggere offline.

IMAP, d'altro canto, di default mantiene la tua posta sul server. Molte implementazioni a livello corporate per la posta elettronica fanno uso di una qualche forma di IMAP, a seconda del software utilizzato, In IMAP, puoi creare delle cartelle nella tua cassetta postale e spostare i messaggi tra queste cartelle. Quando ti colleghi al server IMAP, le tue cassette postali ed il server sincronizzano le cartelle, i contenuti, le email in ingresso e la posta cancellata. Questo ha non pochi vantaggi: puoi consultare tutta la tua posta da qualunque computer o device che usi: portatile, totem, smartphone o tablet. Inoltre puoi scaricare e conservare le email in dei file personali sul tuo computer.

Tuttavia, ci sono anche due inconvenienti: il primo, ovviamente, è che hai bisogno di scambiare più informazioni, quindi ti servirà una connessione più veloce e più tempo. Il secondo è che lo spazio a disposizione è limitato. Il server di posta ti assegnerà una cassetta postale di una certa dimensione che non dovrai superare. Se la riempi, non potrai ricevere altri messaggi a meno che non cancellerai qualche messaggio (o pagherai per avere più spazio). In definitiva questo vuol dire che la posta aziendale basata su IMAP deve essere gestita. Devi archiviare la posta localmente e pulire la cartella della posta inviata, dello spam, ed il cestino in modo ricorrente per conservare spazio. Le email contenenti allegati ti distruggeranno. In quest'epoca degli account email gratuiti con enorme spazio di archiviazione, tutta questa manutenzione potrebbe sembrare stupida. Fino a che non vieni citato. O qualcuno compromette il server di posta e ruba TUTTE le tue email.

Sia i server POP che IMAP richiedono una password per accedere al tuo account. Ma entrambi i protocolli inviano *tutto* in chiaro, comprese le password, e quindi chiunque potenzialmente potrebbe leggerle. Devi usare una qualche forma di crittografia per rendere illeggibile il processo di login (come SSL) ed i contenuti della posta. Questo è il motivo per il quale molti client email presentano la spunta *Usa SSL*.

Quando clicchi sul bottone *Invia* nel tuo client di posta, accadono due cose: prima il tuo client ti obbliga ad effettuare l'accesso al server SMTP (anche se hai già fatto quello per il server POP, maledizione!), e solo dopo invia la tua posta in uscita (tramite il protocollo SMTP).

Questo divenne fastidioso dalla metà degli anni 90, quando i server iniziarono ad usare un protocollo chiamato **POP-before-SMTP**: prima invii al server POP nome utente e password, e la tua posta in ingresso viene scaricata, dopo il server SMTP verifica le tue credenziali tramite il server POP ("Questo tipo è a posto?" "Si, l'ho autenticato io.") ed invia i tuoi messaggi. È un piacevole risparmio di tempo.

Una cosa importante da ricordare è, nonostante sia protetta da password, la posta elettronica non va usata per inviare informazioni in modo sicuro. Molti client e server POP richiedono che la password venga comunicata – in chiaro – al tuo server di posta. Questo non vuol dire che chiunque riceva una email da te riceva anche la tua password; ma vuol dire che qualcuno con le giuste conoscenze ed i mezzi opportuni può sniffare la tua password – così come il contenuto delle tue email. (Per idee su come rendere la tua posta elettonica più sicura, leggi **Lezione 9: Sicurezza delle Email**).

Esercizi

4.5 Invia una email a te stesso dal tuo account principale allo stesso account. Invia la stessa email sempre all'account principale ma da un altro account, ad esempio uno di quelli gratuiti disponibili online (dai, sappiamo che ce l'hai). Quanto ci mettono i due messaggi ad arrivare? Se c'è differenza, perché?

4.6 Apri uno dei miliardi di messaggi di spam che intasano la tua casella in entrata. Puoi stabilire chi è il vero mittente di quella particolare email? Ad esempio, c'è una qualche forma di informazione nascosta nel messaggio? Se c'è, come fa un bravo hacker a vederla?

4.7 Puoi ritardare l'invio di una email fino ad una certa ora o giorno (il che può veramente mandare all'aria il concetto di "Deniability")? Puoi pensare ad un bel modo di usare l'invio ritardato per fare casino con i tuoi amici?

IRC

IRC (Internet Relay Chat) è un ottimo posto per osservare la natura ribelle di Internet al suo massimo. O peggio. Su IRC, chiunque abbia qualcosa da dire ha l'occasione di farlo. IRC è anche conosciuta come **Usenet** o **news group**. Ogni news group ha il suo proprio nome, sub-nome, sub-sub-nome e così via.

Potrebbero esserti familiari le chat room. IRC è proprio come una chat room, solo che non ci sono regole oltre una **netiquette** di base, e abbastanza spesso non ci solo moderatori. Potresti trovare esattamente quello che stai cercando su un canale IRC, o qualcosa che non sapevi neanche esistesse.

> Tutte le regole di cui hai sentito parlare riguardo alle chat room sono applicabili ai canali IRC. Non dire a nessuno il tuo vero nome. Non dare il tuo numero di telefono, il tuo indirizzo, o i tuoi numeri di conto corrente. Ma divertiti! Se vai in giro, stai attento ai contenuti che sono disponibili. Non tutto su Internet è libero da malware, e non tutti su Internet sono gentili.

IRC non è sicura e tutto quello che digiti viene inviato in chiaro tra un server IRC e l'altro. Puoi impostare conversazioni private tra te ed un altro membro IRC ma anche quelle vengono trasmesse in chiaro. L'utilizzo di un nickname ti garantirà giusto un po' di privacy. Se stai pensando di commettere azioni malevole o sgradevoli, non usare lo stesso nickname per ogni account. Usare lo stesso nickname è un ottimo modo per venire rintracciati dalla polizia. O da persone molto meno gradevoli.

Gli argomenti sono chiamati "channels" o canali. Dal momento che ci sono migliaia di canali, ti forniamo una lista di URL dove ne sono indicati molti in modo che tu non impazzisca a cercarli:

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

Se stai avendo problemi con i commenti fatti da un altro membro, puoi o farlo presente al moderatore (se c'è), o ottenere che quella persona venga **espulsa** dal channel. Se non vuoi sentire quello che qualcuno ha da dire, puoi sempre bloccare o ignorare i suoi messaggi. Forse quell'argomento non fa per te in ogni caso.

Esercizi

4.8 Trova tre canali IRC riguardanti tematiche di sicurezza. Come fai ad unirti alla conversazione pubblica? Cosa devi fare per intrattenere una conversazione privata con una persona?

4.9 Che porta usa IRC?

4.10 È possibile scambiare file tramite IRC. Come fai? Vorresti scambiare file tramite IRC?

4.11 Qual'è la maggior differenza tra MIME e SMIME? Quando vedi una "S" in un acronimo, vuol dire qualcosa di speciale per te in quanto persona orientata alla *Sicurezza* (suggerimento)?

FTP

Il vecchio **File Transfer Protocol (FTP)** tipicamente gira sulle porte 20 e 21. Indovina: ti permette di trasferire file tra due computer. Anche se può essere utilizzato per il trasferimento di file privati, dal momento che non usa la crittografia è più comunemente usato per server FTP anonimi e gratuiti che offrono accesso pubblico a collezioni di file, come ad esempio la ISO di quella nuova distribuzione Linux.

L'FTP anonimo una volta era il metodo principale tra utenti di computer per scambiarsi file su Internet. Mentre ci sono molti server FTP anonimi usati per distribuire file illegalmente (che è un modo simpatico per diffondere malattie digitali), molti altri sono usati legalmente per distribuire programmi e file. Puoi trovare server che offrono servizi FTP anonimi nei soliti modi, come ad esempio i motori di ricerca. Ma ricorda: il login FTP viene effettuato in chiaro. Si, anche se stiamo parlando di username e password (questa cosa è debole o no?). Esiste un secure FTP (SFTP) ma non viene usato diffusamente.

La maggior parte dei server FTP anonimi ti consente di accedere ai loro file usando il protocollo FTP da un browser web. Ci sono anche molti ottimi client FTP che funzionano come un programma per la gestione dei file. Una volta che hai effettuato l'accesso al server FTP, puoi spostare i file sul tuo computer nello stesso modo in cui muovi i file tra le cartelle dello stesso. L'FTP impiega giusto un po' più di tempo a scaricare ogni file sul tuo computer, principalmente per il fatto che il server potrebbe trovarsi dall'altra parte del pianeta.

Esercizi

4.12 Windows, OSX e Linux hanno un semplice client FTP a riga di comando; per accedervi, apri il prompt dei comandi o una finestra di terminale e digita:

```
ftp
```

Al prompt `ftp>`, puoi scrivere `help`, per visualizzare la lista dei comandi disponibili.

```
ftp> help
Commands may be abbreviated. Commands are:
!               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
```

I comandi base sono:

Connettersi al server FTP chiamato *ftp.domain.name*:

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

Visualizzare la lista dei contenuti della cartella di lavoro remota:

```
ftp> ls
```

o

```
ftp> dir
```

Cambiare la cartella di lavoro remota passando alla directory chiamata *newdir*:

```
ftp> cd newdir
```

Scaricare un file chiamato *filename* dal computer remoto a quello locale:

```
ftp> get filename
```

Scaricare molteplici file chiamati *file1*, *file2*, e *file3* dal computer remoto a quello locale (puoi anche usare delle wildcard per scaricare più file con lo stesso suffisso, o tutti i file in una cartella):

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

Caricare un file chiamato *filename* dal computer locale a quello remoto:

```
ftp> put filename
```

Scollegarsi dal server FTP remoto:

```
ftp> close
```

Chiudere il client FTP:

```
ftp> quit
```

Una sessione FTP, passo per passo

Per connettersi ad un servizio ftp anonimo, prima apri il tuo client FTP:

```
ftp
```

Usa il comando *open* per connetterti al server. Il comando

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

connette il client FTP al server FTP anonimo chiamato *anon.server*. Sostituisci il nome di un server reale, ovviamente.

Quando il server FTP remoto accetta la tua connessione, si identificherà al tuo client, e dopo chiede di inserire un nome utente.

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

Per la maggior parte dei server FTP anonimi, dovresti inserire la parola *anonymous* (o *ftp*) come nome utente. Il server FTP riconoscerà che ti stai connettendo come un utente anonimo, e ti darà istruzioni su quale password usare.

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

Spesso, il server non verifica la validità dell'indirizzo email inserito come password, quindi non ti impedirà di accedere al servizio se inserisci un indirizzo non valido. Questo viene considerato come una inosservanza della netiquette, ma è in realtà è necessario: non fornire il tuo indirizzo email! Dopo aver inserito una password, il server remoto invierà un messaggio di benvenuto al tuo computer.

```
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.
```

Adesso puoi usare i comandi *ls*, *dir*, *cd* e *get* per scaricare in locale i file dal server remoto.

Esercizi

4.13 Usando questi esempi, trova e scarica un file da un server FTP anonimo.

4.14 Usa il tuo browser web ed un motore di ricerca per trovare un server FTP anonimo su cui è presente una copia di *Alice nel Paese delle Meraviglie*, quindi, usando il client FTP a riga di comando – non il tuo browser – scarica il file.

4.15 Quali sono i migliori client FTP in circolazione? Possono automatizzare tutte le funzioni di quelli a riga di comando e fornirti una bella interfaccia grafica? Perdi qualcuna delle funzionalità disponibili a riga di comando?

4.16 Il tuo computer potrebbe fungere da server FTP?

Telnet e SSH

Telnet permette ad un utente locale di inviare un'ampia gamma di comandi ad un computer remoto. Questo gli consente di istruire il computer remoto ad effettuare delle operazioni ed a restituire i dati al computer locale, quasi come se si fosse seduti d'avanti alla tastiera del computer remoto. **Secure Shell (SSH)** è inteso come un sostituto sicuro e criptato del telnet (che lavora in chiaro).

Ancora, la maggior parte delle versioni di Windows, OSX e Linux forniscono un semplice client telnet a riga di comando; per accedervi, apri il prompt dei comandi o una finestra di terminale e digita:

```
telnet
```

Per accedere ad un server telnet, avrai bisogno di un account e di una password configurate per te da un amministratore del server, dal momento che puoi fare un sacco di cose con telnet, alcune delle quali potrebbero compromettere seriamente il server.

Telnet venivca usato per consentire agli amministratori di controllare remotamente i server e per fornire supporto agli utenti a distanza. Questo servizio è parte della vecchia Internet e non viene usato ormai molto.

Telnet può essere usato anche per una serie di altri compiti, come inviare e ricevere email e visualizzare il codice sorgente delle pagine web (anche se usare telnet è probabilmente il modo più difficile per fare queste cose). Molte di queste cose sono legittime, ma possono essere abusate per motivi illegali o immorali. Puoi usare telnet per controllare la tua posta, e vedere, non solo il soggetto, ma anche le prime righe di messaggio, il che ti consente di decidere se cancellare o meno l'email senza scaricare l'intero messaggio.

Se decidi di utilizzare SSH, assicurati di usare la versione attuale, perché le vecchie versioni presentano varie vulnerabilità, e molti scanner automatici di vulnerabilità su Internet sono alla costante ricerca di queste versioni.

Game On: Comandami

Lo schermo nero rifletteva sugli spessi occhiali del nonno mentre il cursore lampeggiava impazientemente, in attesa di un comando. Con la sua sottile capigliatura grigia che copriva pigramente la sua testa rugosa, il nonno digitò sulla tastiera. Jace osservava il silenzioso suonatore di piano suonare la tastiera del suo computer, tap, tap, tap, tap. Lui sorrise a Jace voltandosi a guardare nei suoi giovani occhi. "Jace, sto per mostrarti un nuovo mondo la fuori. Allacciati la cintura," strizzando l'occhio alla bimba di otto anni.

I piedi di Jace toccavano a malapena terra sulla sedia per computer con suo nonno dietro allo schermo del computer. Lei sentì il tono di chiamata del telefono provenire da una piccola scatola affianco. La scatola bianca si accendeva con luci verdi e rosse quando il tono cambiò sembrando un'anatra che veniva ingoiata da un tritarifiuti. Il nonno sollevò i suoi sopraccigli eccitati e fissò con tutte le forze lo schermo nero di fronte a lui. L'anatra smise di piangere e tutte le luci sulla scatola erano verdi.

Il nonno disse, "Guarda questo."

Solitamente quando il nonno diceva, "guarda questo" qualcosa esplodeva o usciva del fumo nero da qualcosa. In ogni caso, "guarda questo" significava che la nonna si sarebbe arrabbiata per qualche suo errore. Jace però amava udire quelle parole, perché erano l'eccitante anticipazione di qualche fantastico evento.

Lo schermo del computer si risvegliò dal suo scuro torpore con un banner di testo ASCII che contornava le parole "Welcome to Cline's Bulletin Board System (BBS)." "Siamo dentro," il nonno applaudì, tentando di dare il cinque alto a Jace. Le sue mani mancarono la sua di svariati centimetri e lui quasi la colpì sulla faccia. Lei rise così come il nonno.

Guardavano entrambi avanti ed indietro alla tastiera ed allo schermo del computer. Il nonno si sfregava le dita mentre Jace si arrovellava, cercando di capire cosa stesse succedendo. Il nonno iniziò a digitare comandi sul silenzioso pianoforte, capo chino sui tasti come farebbe un avvoltoio mangiando una carcassa. Testa su e giu, su e. Oops. Si mise dritto. Il nonno aveva dimenticato qualcosa di molto importante.

Si fermò e parlò come un insegnante. "Jace, scusami, ho dimenticato di dirti cosa sta succedendo. Proprio ora, sono collegato ad un altro computer tramite la nostra linea telefonica. Quella cosa rumorosa che sta lì si chiama "Modem" ed il suo compito è di convertire i segnali digitali in analogici e viceversa". Jace sapeva già fin troppo sui sui sistemi telefonici vista la passione del nonno di giocarci ogni volta che ne aveva la possibilità. 48 volt durante il normale uso e 90 volt mentre squilla, lei sapeva più di quello che doveva sapere un tecnico dei telefoni. Plain old telephone system o POTS era uno scherzo tra nonno e lei. La nonno non sembrava cogliere lo scherzo del POTS il che lo rendeva ancora più divertente.

Le linee telefoniche possono essere intercettate, ma questa cosa può essere rilevata usando un regolatore di tensione. La tensione della linea telefonica avrebbe presentato un picco momentaneo rimanendo leggermente elevata se qualcuno provava ad intercettarla. Jace pensava che il nonno amasse il suo voltmetro più della nonna; non usciva mai di casa senza. Arrivò addirittura a dargli un nome "Valerie". Valerie il voltmetro. Era il suo miglior amico, a parte Jace.

Jace alzò i suoi occhi saccenti al cielo, ancor di più durante la lezione sulla modulazione analogica e digitale, che converte il suono in un segnale digitale. Quello è più o meno ciò che fa un modem. Il nonno continuava la sua lezione alla riluttante studentessa, "Il computer a cui sono connesso mi consente di connettermi ad altri computer e di giocare con qualsiasi servizio essi forniscano". Le sue orecchie si drizzarono su una parola che non aveva mai sentito prima, "servizi".

"Nonno, cosa intendi per 'servizi?'" chiese la ragazza curiosa aspettandosi una risposta riguardante il fast food. "Eccellente domanda, mia cara," il nonno stava aspettando che Jace facesse una domanda come quella. "Il mio computer è connesso ad una rete di computer, ho la possibilità di connettermi ad altri computer in tutto il mondo," rispose volentieri. "Questo modem mi consente di parlare a questi altri computer che forniscono l'accesso a file, informazioni, persone con cui parlare, ed altre cose meravigliose. Questi computer offrono servizi come File Transfer Protocol, Usenet, IRC, Telnet, ed Email."

Jace non era soddisfatta della risposta che gli era stata fornita e solitamente questo portava a molte altre domande sparate a raffica al nonno. Lei caricò la sua cartucciera di domande ed iniziò la carneficina: "Che cos'è File Transfer proto cosa? Che così MIC? Dov'è Telknit? Per l'Imail ci vogliono francobolli speciali? Chi ha inventato Usednet? Perché la chiamano Imail? La nonna sa dei tuoi servizi? Perché li chiamano servizi? Da dove vengono i bambini? Da dove viene la gelatina?"

Il nonno dovette coprirsi le orecchie per difendere il suo cervello dal massacro di domande. "Aspetta, aspetta, aspetta, rallenta".

Game Over

DNS

Quando vuoi chiamare un amico al telefono, devi conoscere il suo numero di telefono; anche quando vuoi collegarti ad un computer remoto, devi conoscere il suo numero. Potresti ricordare dalle precedenti lezioni che, per i computer su Internet, questo numero è l'indirizzo IP.

Gli indirizzi IP sono gestiti con facilità dai computer, ma noi umani preferiamo usare dei nomi, in questo caso **nomi di dominio**. Ad esempio, per connettersi al sito Hacker Highschool, digita www.hackerhighschool.org nella barra di indirizzo di un browser web. Tuttavia, il browser non può usare questo nome per connettersi al server che ospita il sito Hacker Highschool – ha bisogno dell'indirizzo IP. Questo vuol dire che il tuo computer deve avere un qualche modo per tradurre i nomi di dominio in indirizzi IP. Se c'erano solo centinaia, o anche migliaia di computer su Internet, allora ti sarebbe stato possibile avere una semplice tabella (o file **hosts**) sul tuo computer per poter ritrovare questi indirizzi. Tuttavia, piaccia o meno, non solo ci sono milioni di computer su Internet, ma le corrispondenze tra nomi di dominio ed indirizzi IP cambiano continuamente.

Domain Name Service (DNS) viene usato per tradurre dinamicamente i nomi di dominio in indirizzi IP (e viceversa). Quando digiti il nome di dominio *www.domainname.com* nel tuo browser, esso contatta il server DNS scelto dal tuo ISP. Se quel server DNS ha *www.domainname.com* nel suo database, allora restituisce l'indirizzo IP al tuo computer, consentendoti di connetterti.

Se il tuo server DNS non ha nel suo database *www.domainname.com*, allora invia una richiesta ad un altro server DNS, e continueranno ad inviare richieste ad altri server DNS fino a che uno fornisce il corretto indirizzo IP, o stabilisce che il nome di dominio non è valido.

Esercizi

4.17 Apri una finestra a riga di comando ed identifica l'indirizzo IP del tuo computer. Quale comando hai usato? Qual'è il tuo indirizzo IP?

4.18 Identifica l'indirizzo IP del tuo server DNS. Quale comando hai usato? Qual'è l'indirizzo IP del server DNS?

4.19 Pinga *www.isecom.org.* Ricevi una risposta? Quale indirizzo IP risponde al ping?

4.20 Puoi istruire il tuo computer ad usare un server DNS differente? Se si, modifica la configurazione del tuo computer affinché usi un altro server DNS. Pinga nuovamente *www.isecom.org.* Ricevi la stessa risposta? Perchè?

DHCP

DHCP o **Dynamic Host Configuration Protocol** consente ad un server di rete locale di fornire indirizzi IP all'interno della rete. Al server viene dato un blocco di indirizzi IP da utilizzare. Quando un computer si unisce alla rete, riceve un indirizzo IP. Quando un computer si scollega, il suo indirizzo IP diventa disponibile per un altro computer.

Questo è utile in reti di computer estese, dal momento che non è necessario assegnare ad ogni computer un indirizzo IP statico. Si usa invece, appunto, un server DHCP. Quando un nuovo computer si connette alla rete, la prima cosa che fa è richiedere un indirizzo IP al server DHCP. Una volta che gli viene assegnato un indirizzo IP, il computer ha accesso a tutti i servizi presenti nella rete.

Ora rifletti su questo. Molte reti wifi utilizzano DHCP, nel senso che *chiunque* può prendere un indirizzo IP su quella sottorete. Se gestisci un bar è esattamente quello che vuoi, ma in un ufficio sicuro, potresti invece considerare di usare indirizzi IP statici. Dipende...

Connessioni

Ai vecchi tempi, i computer si connettevano ad Internet attraverso un modem. I modem traducono i bit in suoni e viceversa, **mod**ulando e **dem**odulando, da qui il nome. La velocità del modem si misura in **baud** e **bps**, o bit al secondo. Un maggior numero di baud di solito significa più bps, ma devi anche considerare che cosa hai in mente di fare. Ci sono alcune applicazioni – come collegarsi via telnet ai **Multi-User Dungeons (MUDs)** – per le quali un modem a 300 baud vecchio di vent'anni sarebbe ancora accettabile (ammesso che la tua velocità di scrittura non fosse così buona), mentre applicazioni che richiedono molta banda come ad esempio lo streaming video possono spesso spremere anche le più veloci connessioni via cavo o DSL.

Gli ISP

Tu non puoi prendere e chiamare Internet. Hai bisogno di accedere ad un server che connetterà il tuo computer ad Internet. Il server fa tutto il lavoro pesante, ed è sempre acceso. Il server è gestito da un **Internet Service Provider (ISP)**.

Un ISP ha un costante punto di accesso (point-of-presence) su Internet, ed ha dei server che forniscono servizi che puoi usare. Ma anche tu puoi fornire questi servizi. Ad esempio, puoi fornire un server di posta sul tuo computer, ma questo richiederà che il tuo computer sia sempre acceso e connesso ad una rete, in attesa di quei brevi momenti in cui c'è scambio di informazioni. Un ISP, tuttavia, consolida gli sforzi di un ampio numero di utenti, quindi il server di posta lavora continuamente, invece di starsene in attesa senza far nulla. I computer dell'ISP utilizzano una connessione ad alta velocità per connettersi ad un **Network Access Point (NAP)**. Questi NAP si connettono poi tra loro tramite connessioni ultraveloci chiamate **backbone**. Tutte queste cose insieme formano Internet.

Plain Old Telephone Service

Plain old telephone service (POTS) una volta era il metodo più usato per accedere ad Internet. Il suo principale svantaggio è la bassa velocità, ma in molti casi è una risorsa per la sua ampia disponibilità. Molti ISP nazionali hanno un vasto numero di numeri di accesso locale, e praticamente tutti ancora hanno un telefono ed una linea. In teoria, se tu avessi un modem acustico ed un sacco di monetine in tasca, potresti connetterti da quasi ogni telefono pubblico (se ne trovi uno). Non che tu voglia farlo veramente.

POTS è lento. I più veloci modem telefonici riportano una velocità teorica di 56,600 bit al secondo (bps). Tuttavia non è realmente così, come viene scritto in piccolo. I vincoli di potenza limitano la velocità di download reale a circa 53,000 bps e quella effettiva è solitamente molto inferiore. Non c'è paragone con la DSL o il modem via cavo.

Detto questo, il servizio telefonico è ampiamente diffuso, e gli ISP basati sul POTS sono relativamente economici (e a volte gratuiti). Non vorresti scambiare film piratati tramite POTS, perché è immorale, illegale e tiene occupata la tua linea telefonica tutta la notte e forse tutto il weekend, ma potresti certamente inviare delle innocenti email testuali alla tua nonnina. E se usassi il telnet, potresti farlo addirittura con una polverosa macchina DOS ritrovata in cantina.

DSL

Una **Digital Subscriber Line (DSL)** è un sistema per inviare grandi quantità di informazioni sui fili già esistenti utilizzati per il POTS. Il suo maggior vantaggio rispetto al POTS è che è molto più veloce dei modem analogici, e fornisce una connessione permanente. Inoltre, ti consente di fare e ricevere le telefonate mentre sei connesso ad Internet. Il suo più

grande svantaggio è che la sua disponibilità è limitata da quanto sei vicino alla cabina della compagnia telefonica – se vivi in una zona troppo distante, non hai possibilità.

I Modem via Cavo

I gateway via Cavo non usano le tradizionali linee telefoniche per connettersi ad Internet. Usano invece un cavo coassiale (o se sei veramente fortunato, la fibra ottica) fornito dalle compagnie via cavo. Come la DSL, i gateway via cavo ti consentono di fare e ricevere telefonate mentre sei connesso ad Internet, e forniscono una connessione permanente, ma sono generalmente più veloci della DSL.

I gateway via cavo hanno alcune pecche di base. La prima è che l'accesso è una risorsa condivisa, quindi la tua velocità di connessione verrà ridotta quando ci sono altre persone sul tuo stesso cavo. La seconda è che l'accesso via cavo è disponibile solo in aree dove le relative compagnie hanno installato i cablaggi necessari. E la più importante di tutte è che tutto il traffico che metti sul cavo può essere visto da ogni altro utente su quel cavo! Questo vuol dire che se colleghi il tuo computer al gateway via cavo e non usi un firewall, chiunque altro nel vicinato può vedere il tuo computer e tutti i suoi file. Vuoi veramente condividere le informazioni relative al tuo conto bancario in questo modo?

Wimax

Wimax è un sistema di connessione senza fili che compete con la DSL. È usato in posti dove è troppo difficile o costoso installare un'infrastruttura cablata. La forza del segnale può essere influenzata da edifici, alberi o altri grossi oggetti. Alcune versioni usano un access point fisso, ma altre ti danno accesso in mobilità su aree veramente ampie.

Wifi

Il Wifi non è un modo per connettersi al tuo ISP ma è un sistema di rete comunemente usato per connettersi ad Internet da casa o da attività commerciali come centri commerciali o bar. La maggior parte degli smartphone e tutti i portatili ora usano il Wifi, che quindi è uno dei bersagli favoriti dagli attaccanti. Pensa a te stesso come nudo in una stanza piena di gente quando usi il Wifi pubblico: copriti, accertati che nessuno ti stia guardando ma supponi che tutti vogliano farlo. Ovviamente leggerai anche la lezione sulla Sicurezza del Wireless, giusto?

Esercizi

4.21 Che tipo di connessione internet hai a casa, se ne hai una? Come puoi stabilirlo? E soprattutto:

4.22 Chi puoi vedere su quella rete? (Come puoi capirlo?)

4.23 Quanto è veloce la tua connessione? Puoi migliorare la tua velocità senza contattare il tuo ISP?

4.24 Quali altri servizi fornisce il tuo ISP? Abbiamo già parlato di servizi; il tuo ISP potrebbe supportarne svariati.

4.25 Quali servizi puoi fornire dal tuo computer?

Nutri la tua Mente: Giocando con l'HTTP

HTTP, l'acronimo di Hypertext Transfer Protocol, lavora al livello più alto della pila TCP/IP come descritto nei due principali RFC:

- 1945 per l'1.0 (basato su 0.9).
- 2616 per l'1.1.

Ci sono alcune sostanziali differenze e migliorie dalla 1.0 alla 1.1 riguardanti l'Estensibilità, l'utilizzo della Cache, l'ottimizzazione della Banda, la gestione della connessione di Rete, la trasmissione del Messaggio, la notifica di Errori, la Sicurezza, l'Integrità e l'autenticazione, la negoziazione del Contenuto (3). Le differenze tra 1.0 ed 1.1 sono informazioni utili riguardanti un server web.

Si dice che l'HTTP è un protocollo "stateless" nel quale un Client invia una Richiesta HTTP al Server, che invia una Risposta HTTP: il paradigma Request/Response.

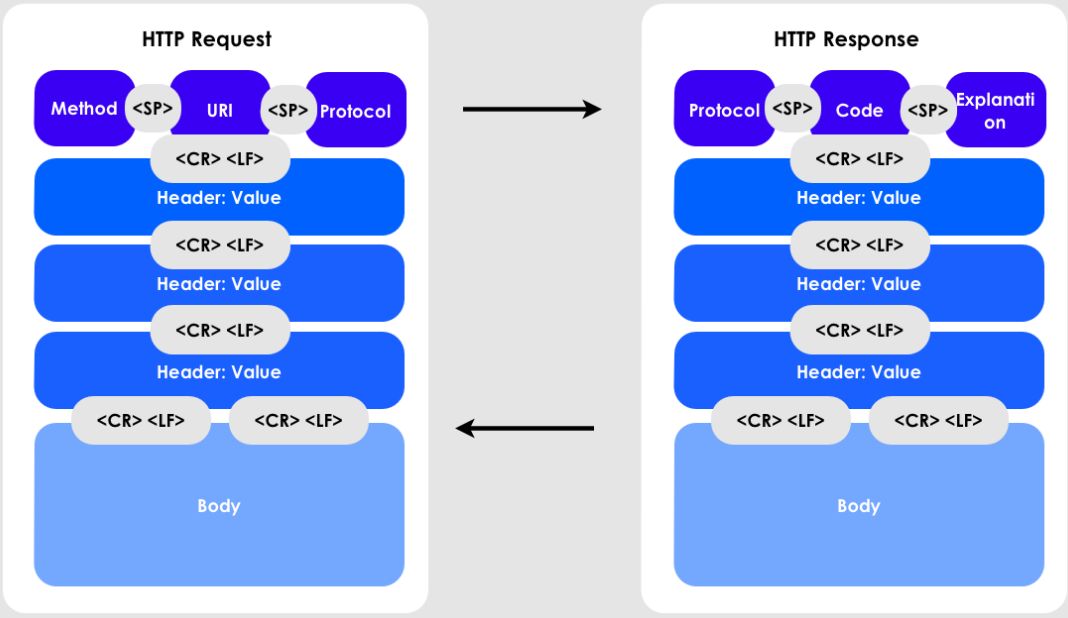

Figura 4.1: HTTP

Come saprai possiamo ottenere molte informazioni inviando dei comandi ad un server HTTP. Useremo alcuni tool di rete di base:

- netcat: il toolkit TCP/IP
- curl: il toolkit HTTP
- proxy: come OWASP ZAP o Burpsuite free

Sniffare la Connessione Tra Te ed il Server HTTP di HHS

Usa un proxy per connettere il tuo browser. Vai su http://www.hackerhighschool.org ed intercetta la tua richiesta:

```
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
```

e risposta:

```
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>

[...]
```

Esercizi

4.26 Usando il proxy identificare le parti delle richieste indicate nei diagrammi.

4.27 Ci sono informazioni interessanti negli header?

La tua prima Connessione Manuale

Netcat può essere usato per connettersi ad un server web settando la porta corretta.

Inizia digitando:

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

Quindi premi *Enter* due volte.

```
GET / HTTP/1.0
```

The server will reply:

```
<!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" />
```

Come puoi vedere la pagina risulta provenire da isecom.org e non da hackerhighschool.org. Come mai?

Un'ipotesi potrebbe essere che sullo stesso host girino entrambi i siti HHS e ISECOM. È possibile?

Per scoprirlo, controlla l'indirizzo IP di hackerhighschool.org:

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

And now for www.isecom.org:

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

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

Lo stesso indirizzo IP! Usando netcat è possibile mostrare l'host aggiungendo manualmente l'header Host ed usando 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>
```

Il Metodo Request

Un'altra parte di una richiesta HTTP che può essere modificata è il Metodo Request. Solitamente le applicazioni web usano le richieste GET e POST, ma su un web server od un application server potrebbero essere attivi altri protocolli di richiesta. I metodi comuni sono:

- **OPTIONS** – usato per chiedere quali sono i tipi di richiesta supportati.
  Se stai gestendo un server web, tieni presente che fornire questo tipo di informazioni potrebbe essere fonte di problemi.
- **GET** – usato per recuperare informazioni direttamente dall'URL, ad esempio:
  `http://www.usairnet.com/cgi-bin/launch/code.cgi?Submit=Go&sta=KSAF&state=NM`
  Vedi tutto quello che c'è dopo il punto interrogativo? Quelli sono i dati della richiesta. Passare le richieste in questo modo è rischioso, perché sono in bella vista, ed è facile modificarle.
- **HEAD** – usato come GET mail server non restituisce una pagina reale.
  Può essere usato per identificare gli Accessi, ottimizzare il consumo di banda e – in alcuni casi – bypassare il controllo accessi. Infatti alcune implementazioni delle ACL verificano solo le richieste di tipo GET. In questo caso hai trovato una Vulnerabilità.
- **POST** – usato per inviare dati alle applicazioni web – come GET – ma i dati vengono trasmessi nel Request Body, non in piena vista o almeno in parte.
- **PUT** – usato per allocare le risorse su un server web o per aggiornarle.
  In molti contesti questo metodo dovrebbe venire disabilitato o protetto tramite Autenticazione. In altri contesti questa è una piacevolissima scoperta.
- **DELETE** – usato per liberare risorse su un server web.
  Questo metodo dovrebbe venire disabilitato o protetto tramite Autenticazione. Vedi PUT.
- **TRACE** – usato come un loopback a livello applicazione che riflette i messaggi.
  Questo metodo di debug andrebbe disabilitato, soprattutto in ambiente di produzione per motivi di Confidenzialità e perché introduce una Vulnerabilità potendo essere sfruttato per condurre attacchi di tipo Cross Site Scripting.
- **CONNECT** – per usare il server web come un proxy.
  Andrebbe disabilitato o protetto tramite Autenticazione perché consente ad altri di connettersi a servizi di terze parti usando il proxy IP.

Considera inoltre che altri protocolli basati su HTTP possono aggiungere altri metodi, come WebDAV. Puoi modificare il Metodo Request in modo da osservare le risposte del server in cerca di informazioni interessanti, per chiedere i metodi conosciuti ed addirittura parole arbitrarie.

Richiedere OPTIONS

Avvia la sessione netcat come sempre:

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

Ma non premere *Enter* due volte. Invece, digita il seguente testo:

```
OPTIONS / HTTP/1.1
```

e riceverai una risposta del tipo:

```
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
```

Richiedere HEAD

Questa volta, dopo aver aperto la tua sessione, utilizza l'opzione 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
```

Lascia che ti usi come un Proxy: la Richiesta CONNECT

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

Esercizi

4.28 Usa netcat (nc) per provare tutti i metodi di Richiesta sopraelencati sui server HHS o su un server appositamente configurato. Che tipo di informazioni interessanti riesci a ricavare?

Scrivere delle richieste HTTP con curl

Alcuni Test per Applicazioni Web sono basati non solo sulla risposta del Server Web ma anche a Livello di Applicazione (Web). Spesso puoi trovare delle vulnerabilità delle applicazioni web alterando i parametri GET e POST, modificando i cookie e giostrando sugli header. Uno strumento utile di scripting, è il comando **curl**, un tool per richiedere una pagina web da riga di comando. Ma curl aggiunge della logica a netcat.

Richiedere:

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

non è la stessa cosa di:

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

Puoi verificarlo usando l'opzione -v per ottenere un output più dettagliato o "verbose":

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

Come puoi vedere curl seleziona automaticamente la versione 1.1 di HTTP, aggiunge l'header Host, lo user agent ed accept. Il che ci porta ad un'importante regola per gli hacker: conoscere i propri tool.

Fortunatamente curl è un simpatico tool che può essere personalizzato usando varie opzioni.

Per una lista completa digita `curl -help`

Alcune opzioni per ottenere funzioni simili al precedente esempio netcat sono:

- **-H** per aggiungere un header
- **-X** per selezionare un metodo di richiesta (anche noto come Comando)
- **-d** per aggiungere dati di POST
- **-i** per includere gli header del protocollo nell'output
- **-s** per abilitare il modo silenzioso, utile per gli script

With curl and some bash scripting you can automate web application testing. Looking for interesting HTTP headers from a server can be automated simply with curl and grep:

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

Exercise

4.29 Expand the script above to request more HTTP headers and potentially useful information.

Riferimenti ed Approfondimenti

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/

Conclusioni

Il World Wide Web è nel complesso molto più di Internet: ci sono tutta una serie di servizi oltre al solo HTTP. FTP, SSH, DNS, DHCP e molti altri offrono finestre sui computer degli altri – e sul tuo. Capire come ti colleghi a questi servizi, sia "tramite i canali appropriati" che in altro modo, è fondamentale per comprendere come tu o il tuo computer potete essere attaccati – o potete attaccare. Ricorda sempre il motto: fai Hacking su tutto, ma non fare male a nessuno.

Al giorno d'oggi i ragazzi vivono in un mondo in cui possono accedere ai principali canali di comunicazione, ma non hanno le conoscenze per difendersi contro le frodi, i furti d'identità, le violazioni della privacy ed altri attacchi che subiscono quotidianamente per il semplice fatto di utilizzare Internet.
È per questoche esiste Hacker Highschool.

Il progetto Hacker Highschool punta a sviluppare dei materiali per l'apprendimento e la formazione su temi della sicurezza e della privacy per gli studenti delle scuole medie e superiori.

Hacker Highschool è composto da un set di lezioni ed esempi pratici per diventare degli hacker.

Oltre a renderli consapevoli riguardo a temi di cybersecurity ed a fornire le skill necessarie per navigare su Internet, dobbiamo insegnare ai giovani di oggi ad essere pieni di risorse, creativi e ad usare la logica, tutti tratti distintivi di un hacker. Il programma contiene materiali didattici su sicurezza e privacy e supporta gli insegnanti di scuole medie, superiori e private accreditate. Queste lezioni offrono delle sfide ai ragazzi per stimolarli ad essere creativi come un hacker e trattano l'utilizzo sicuro di Internet, la privacy sul web, le ricerche su Internet, evitare virus e trojan, temi legali ed etici ed altro.

Il programma HHS è sviluppato da ISECOM, un gruppo di ricerca no profit, open source, concentrato sulla sensibilizzazione alla sicurezza ed allo sviluppo della sicurezza professionale ed al suo accreditamento.

Hacker Highschool
SECURITY AWARENESS FOR TEENS

ISECOM