Capitolo 1

Conosci il tuo nemico

Lanciare un'applicazione web su Internet è un compito arduo. I passaggi da compiere per poter distribuire un'applicazione web possono essere onerosi: progettare e programmare le pagine web, aggiungere l'interattività utilizzando JavaScript, implementare i servizi backend e collegarli a un archivio dati, scegliere una piattaforma di hosting e registrare un nome di dominio. Il risultato, ovviamente, vale la pena: il vostro sito web sarà immediatamente disponibile per miliardi di utenti, grazie alla magia di Internet. Tuttavia, non tutti questi utenti hanno buone intenzioni. Internet ospita un complesso ecosistema di script, bot e hacker che cercheranno di sfruttare eventuali falle di sicurezza nella vostra applicazione web per fini "poco gentili. Questo è probabilmente l'aspetto più sconcertante dello sviluppo web: dopo tutto il lavoro che avete dedicato alla creazione della vostra applicazione web, qualcuno arriverà fin da subito a prendere a calci le gomme e a graffiarvi la carrozzeria.

Poiché state leggendo questo libro, probabilmente siete sviluppatori attenti a questi rischi per la sicurezza e desiderate imparare a proteggervi. Questo libro è una guida completa alla sicurezza web: imparerete a proteggere le vostre applicazioni web nel browser, in rete, sul server e a livello di codice. Introdurrò, inoltre, i principi della sicurezza che possono essere applicati a ciascun livello di astrazione.

Prima di addentrarci nei dettagli, tuttavia, vale la pena di indagare su chi sono questi malintenzionati che si aggirano in Internet, che cosa li motiva e quali strumenti utilizzano. Parliamo degli hacker.

In questo capitolo

Capire come gli hacker vi attaccano e perché

L'*hacking* è, nel suo senso più letterale, un tentativo di ottenere un accesso non autorizzato a un sistema software. Ma questa definizione non rende giustizia all'ampia varietà di malviventi e "poco di buono" che popolano Internet, anche se comprende alcune aree grigie che non è il caso di considerare attività di hacking. Condividere i dati di accesso a Netflix con un altro membro della famiglia vi rende fore hacker?
Dovremmo invece spostare il nostro ambito per considerare gli hacker veri e propri, i criminali che prenderanno di mira la vostra applicazione web. Queste persone usano Internet per commettere i loro crimini praticamente da quando esiste Internet. Gli hacker possono essere classificati a grandi linee come *black hat*, che compiono atti illegali per un guadagno economico o politico, o come hacker *white hat*, che tentano di identificare le vulnerabilità prima che i black hat possano sfruttarle. Le grandi aziende spesso mettono in palio dei premi, *bug bounties*, per quest'ultimo gruppo, da elargire a chiunque riesca a trovare difetti nella loro strategia di sicurezza prima che lo facciano dei malintenzionati. Questa pratica ha portato all'ascesa degli hacker *grey hat*, i quali, invece di sfruttare una vulnerabilità, la segnalano, se ritengono che sia più redditizio.

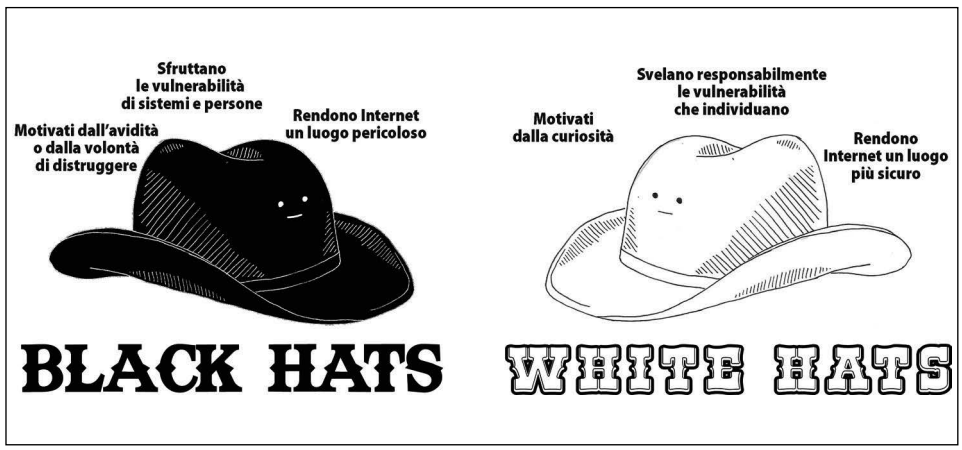

Figura 1.1

Gli hacker su entrambi i lati del "fronte" utilizzano appositi strumenti e script automatizzati per rilevare le vulnerabilità. Questi strumenti sono generalmente open source e facili da ottenere. Molti hacker utilizzano *Kali Linux*, una distribuzione personalizzata di Linux contenente gli strumenti più diffusi di hacking e digital forensic. Gli hacker white hat utilizzano Kali Linux come parte delle loro attività di *penetration testing*, per scansionare un sistema client alla ricerca di punti di accesso vulnerabili, nell'ambito di una valutazione della sicurezza. I black hat utilizzano quegli stessi strumenti per individuare le vulnerabilità che possono sfruttare.
Il mondo dei white hat comprende anche i *ricercatori che operano nel campo della sicurezza*, che cercano di scoprire, documentare e condividere le informazioni sulle vulnerabilità presenti nei software più diffusi. Un ricercatore potrebbe scoprire una vulnerabilità su un noto server web Java, come Apache Tomcat, per esempio, per poi mostrare agli autori del software come si manifesta tale vulnerabilità. Quando viene resa disponibile una patch

software per risolvere il problema, tali vulnerabilità vengono catalogate nel database CVE (*Critical Vulnerability and Exposure*) gestito da MITRE Corporation, un'organizzazione no-profit specializzata nella sicurezza dei computer. Spesso tali vulnerabilità sono catalogate con numeri CVE.

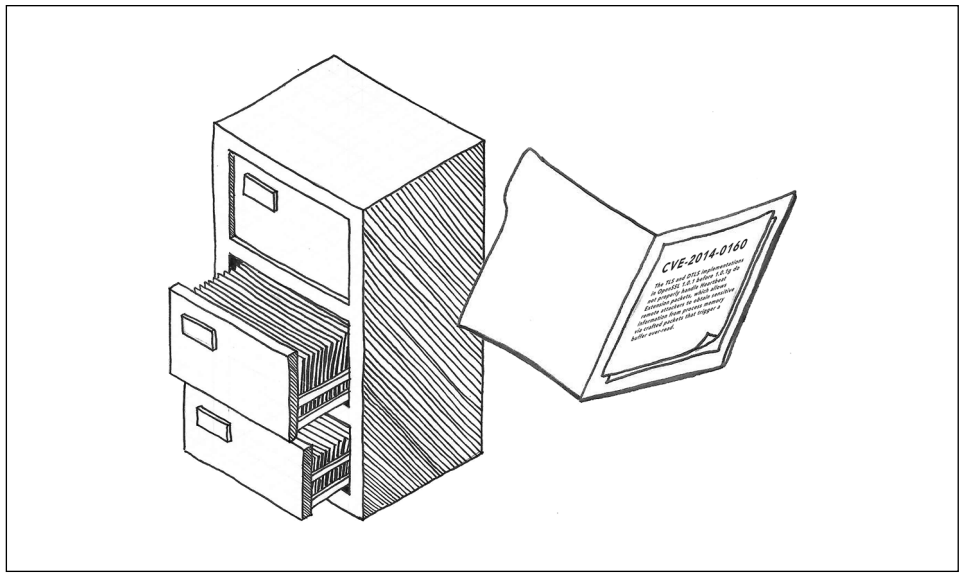

Figura 1.2

Non appena viene pubblicata una nuova CVE, e talvolta anche prima, diventano disponibili anche degli exploit proof-of-concept. Gli *exploit* sono frammenti di codice che mostrano come la vulnerabilità può essere utilizzata per eseguire attività dannose, come l'introduzione di codice ostile in un sistema vulnerabile. Tali exploit vengono rapidamente incorporati in strumenti di hacking come *Metasploit*, comunemente utilizzato dagli hacker sia black hat sia white hat per sondare le vulnerabilità dei siti web. Gli hacker black hat accumulano anche conoscenze sulle vulnerabilità che hanno scoperto, cercando di mantenerle attive il più a lungo possibile, in modo che non vengano risolte.
Lo sfruttamento delle vulnerabilità del software non è nemmeno l'unico strumento nella cassetta degli attrezzi dei criminali. L'*ingegneria sociale* è un processo volto a conquistare la fiducia di una vittima e a persuaderla a divulgare informazioni riservate, come le credenziali d'accesso. L'ingegneria sociale può essere svolta di persona, al telefono o tramite canali di messaggistica. Potreste avere una certa familiarità con le e-mail di *phishing*, che tentano di indurre la vittima a svelare la propria password. Gli hacker ottengono molto successo con lo *spear phishing*, eseguendo ricerche in background per prendere di mira specifiche persone (spesso nei reparti amministrativi delle aziende). Questa forma di frode ha una controparte nelle app di messaggistica e nei social media.
Alcuni dei crimini informatici più audaci degli ultimi anni sono stati assistiti da *insider*: dipendenti o consulenti disonesti che decidono di vendere o divulgare segreti aziendali o proprietà intellettuali o causare altri tipi di danni. Avere un insider ostile in azienda è una delle situazioni più difficili da cui proteggersi, quindi le aziende a rischio tendono a limitare l'accesso ai dati, in base alle effettive necessità del singolo utente.

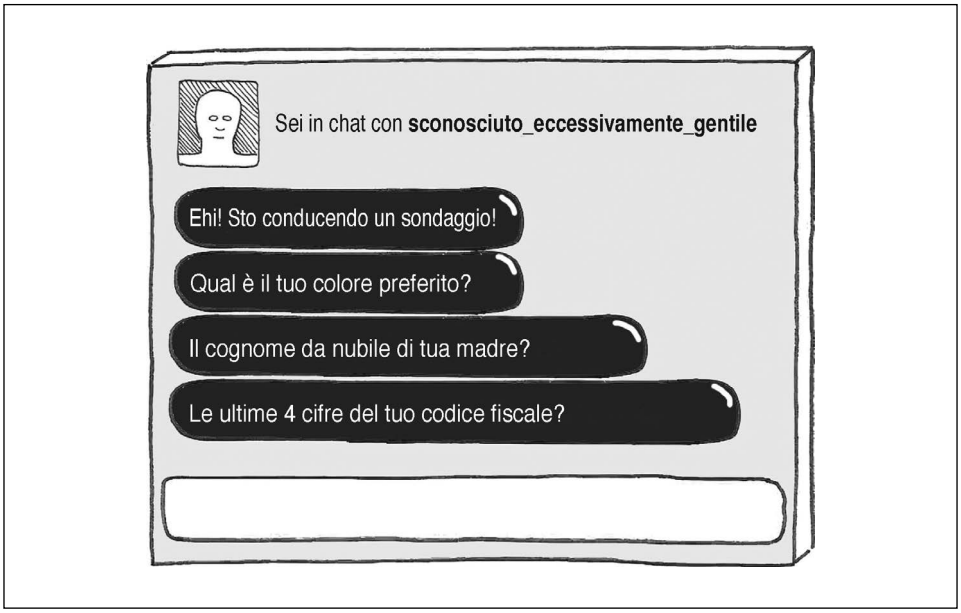

Figura 1.3

Perché la criminalità informatica è così comune? La risposta, ben poco sorprendente, è che può essere un'attività piuttosto redditizia. C'è un'economia sommersa di siti che forma il *Dark Web*, dove gli hacker rivendono dati sottratti, numeri di carte di credito, vulnerabilità individuate e perfino server violati. I pagamenti avvengono in criptovalute, cosa che li rende molto difficili da tracciare. Poiché il Dark Web è accessibile solo tramite il browser Tor, che rende anonimo ogni accesso, questi mercati operano impunemente e sono estremamente difficili da colpire da parte delle forze dell'ordine.

Oltre a vendere sul Dark Web i dati sottratti, i criminali informatici ricorrono all'estorsione per chiedere denaro direttamente alle loro vittime. Il *ransomware* è una forma di software dannoso che crittografa i file di una vittima e ne impedisce l'accesso finché non viene pagato un riscatto in criptovaluta. Vari tipi di aziende, oleodotti, operatori sanitari, fornitori di carne e catene alberghiere, sono state tutte vittime di gravi attacchi e sono state costrette a pagare per sbloccare i propri server. Il ransomware è così onnipresente che gli autori di tale software adottano una sorta di franchising, rendendo i loro strumenti liberamente disponibili ai gruppi di hacker black hat, in cambio di una percentuale sui riscatti riscossi. Gli hacker, a volte, offrono perfino dei "canali di supporto" alle vittime che necessitano di assistenza per decrittografare i propri file system dopo il pagamento di un riscatto.

Vale la pena di notare che non tutto l'hacking viene effettuato per ragioni finanziarie. L'*hacktivismo* è una forma di hacking svolto per ragioni politiche da formazioni che vogliono promuovere la propria causa. Gli obiettivi degli hacktivisti sono spesso lodevoli, per esempio abbattere i siti di social media utilizzati dagli estremisti svelando (*doxing*) i loro aderenti, colpendo regimi repressivi o facendo trapelare documenti dai paradisi fiscali. Anche lo spionaggio informatico gioca un ruolo chiave nella guerra moderna, e i gruppi di hacker più formidabili sono solitamente sponsorizzati da Stati. I gruppi di hacker che

rientrano in questa categoria utilizzano sofisticate tecniche di sorveglianza per colpire le loro vittime. I ricercatori che operano nel campo della sicurezza tracciano tali *minacce persistenti avanzate* (APT – *Advanced Persistent Threat*) monitorando le tecniche che utilizzano. La comunità dedicata alla sicurezza assegna a ogni APT un nome in codice divertente, come *Cozy Bear* (un gruppo di hacker russi) o *Charming Kitten* (un gruppo di guerra informatica del governo iraniano) in diretto contrasto con il caos che provoca.

Figura 1.4

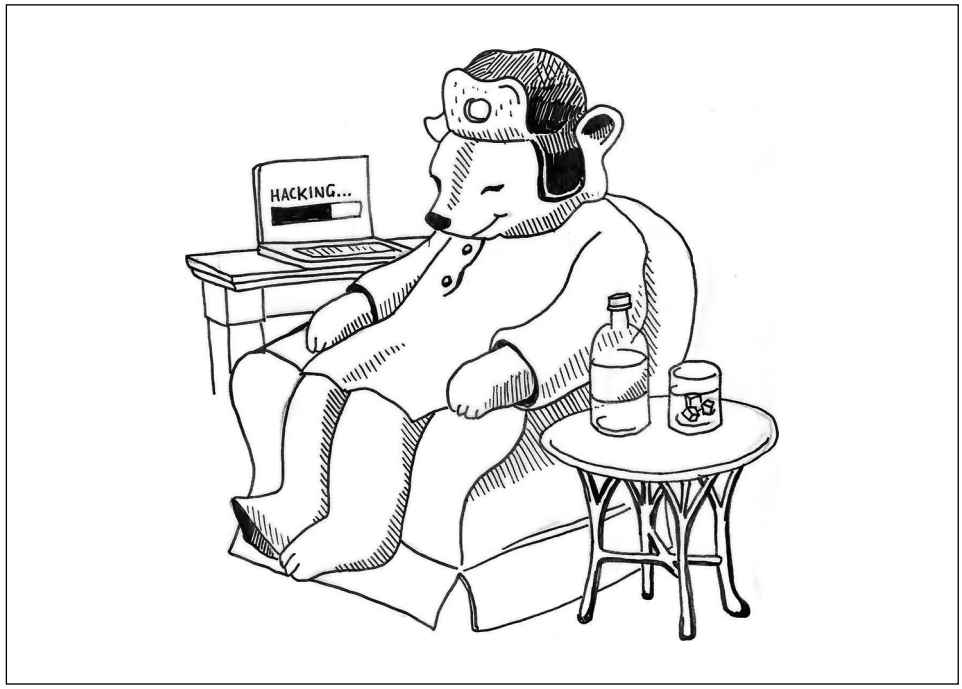

Figura 1.5

Sopravvivere alle conseguenze di un hacking

Ora che abbiamo fatto la conoscenza con i nostri avversari, consideriamo che cosa significa essere vittima di un hacker. Proprio come l'*hacking* descrive una vasta gamma di attività, anche cadere vittima di un hacking può avere tutta una varietà di risultati, con diversi gradi di gravità.

La conseguenza più diretta di un attacco da parte di hacker è che la vostra applicazione web non sarà più disponibile per gli utenti legittimi. Questo tipo di hacking è chiamato attacco DoS (*Denial of Service*). Per raggiungere questo scopo, non è nemmeno necessario che gli strumenti di hacking penetrino nel perimetro di sicurezza; un hacker può semplicemente bombardare i vostri server con un numero così elevato di richieste che agli altri visitatori non rimarrà più alcuna risorsa disponibile.

Nonostante la loro relativa mancanza di sofisticatezza, gli attacchi DoS possono essere difficili da prevenire. Gli attacchi DDoS (*Distributed Denial of Service*), in particolare, utilizzano migliaia di server per inviare richieste simultaneamente da vari indirizzi IP (*Internet Protocol*), rendendo quasi impossibile il blocco delle richieste dannose sulla base della loro origine. Nel 2016, il provider DNS (*Domain Name System*) Dyn è stato vittima di uno dei più grandi attacchi DDoS della storia, che ha reso indisponibili alcuni dei siti web più noti al mondo, da `Amazon.com` a `Zillow.com`.

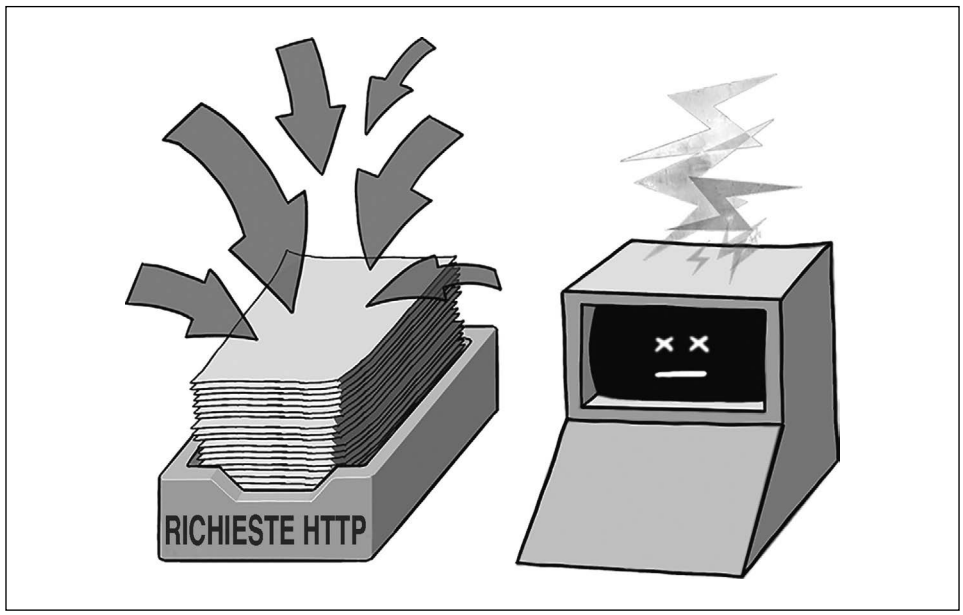

Figura 1.6

Un'altra potenziale conseguenza dell'hacking della vostra applicazione web è che l'hacker la utilizzerà come "rampa di lancio" per prendere di mira i vostri utenti. L'inserimento di codice JavaScript dannoso in un sito web è chiamato XSS (*Cross-Site Scripting*), una vulnerabilità comune che esamineremo nel Capitolo 6. Il codice JavaScript dannoso può causare vari tipi di problemi, deviando gli utenti su altri siti, verso truffe e frodi,

oppure può essere utilizzato per osservare le attività della vittima sul sito host stesso. Gli script di *keylogging* possono acquisire il nome-utente e la password degli utenti quando effettuano il login. Sui siti web finanziari, gli script di *skimming* possono essere utilizzati per sottrarre i dati della carta di credito.

Sottrarre le credenziali d'accesso è un obiettivo comune per gli hacker, perché i nomi utente e le password raccolti possono essere poi venduti sul Dark Web. Le credenziali per i siti di social media più popolari, come Facebook, vengono acquistate da chi poi le utilizza per promuovere truffe (no, vostro zio non si è messo improvvisamente a vendere occhiali da sole a prezzi superscontati; probabilmente il suo account è stato violato e poi rivenduto). Le credenziali sottratte hanno anche un uso secondario: poiché molti tendono a riutilizzare nomi utente e password su più siti web, un hacker può provare le credenziali sottratte in altri siti web, conducendo attacchi di *tipo password spraying*. In alternativa, un hacker può prendere di mira un singolo sito, provando contemporaneamente un intero database di password sottratte per condurre un attacco di *credential stuffing*.

Il modo più rapido con cui un hacker può sottrarre più credenziali in blocco consiste nel trovare un modo per accedere e scaricare il contenuto del database delle password. Tali *violazioni dei dati* rappresentano spesso lo scenario peggiore per molte aziende, perché i dati sono la loro risorsa principale. Nomi utente e password non sono gli unici dati sensibili archiviati nei database; gli hacker possono ottenere token di accesso per servizi di terze parti, registri di chat, segreti commerciali, informazioni di identificazione personale e numeri di carte di credito. In molti paesi, le aziende che subiscono violazioni dei dati sono obbligate per legge a rivelare ai clienti la portata della violazione, il che causa immancabilmente danni alla loro reputazione.

Un hacker che riesca a ottenere *l'accesso in scrittura* al database di una vittima acquisisce la capacità di espandere la portata del proprio attacco. Potrebbero essere in grado di iniettare nel database degli script JavaScript dannosi che verranno visualizzati sulle pagine del sito web delle vittime. Oppure potrebbero inserire file dannosi (per esempio ransomware) che gli utenti del sito verranno indotti a scaricare.

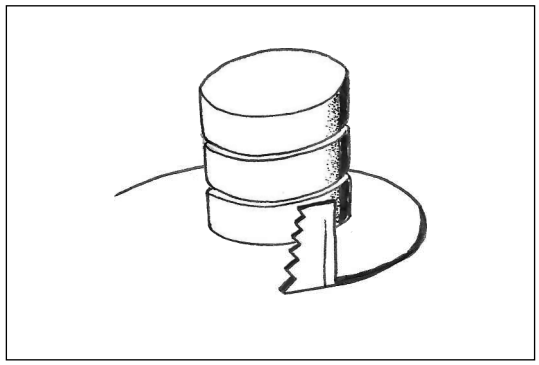

Figura 1.7

Gli hacker che hanno preso piede nel vostro sistema cercheranno di *estendere i loro privilegi*, fino ad acquisire un accesso completo ai vostri server. Gli strumenti che utilizzano a questo scopo si chiamano *rootkit*; gli hacker tentano di accedere all'account root del vostro server, che detiene i privilegi più elevati. Un hacker che abbia ottenuto l'accesso

root può iniziare a utilizzare le vostre risorse di elaborazione per i propri scopi. Rendere il server parte di una *botnet*, una rete controllata centralmente di computer violati, chiamati *bot*, consentirà loro di estrarre criptovaluta, inviare e-mail di phishing, commettere frodi sui clic (utilizzando bot per aumentare artificialmente le visualizzazioni delle pagine) e svolgere molte altre attività fraudolente. L'accesso ai server violati può essere rivenduto sul Dark Web, quindi le vostre risorse di elaborazione potrebbero essere state rivendute a vostra insaputa.

Rilevare i server violati è una sfida impegnativa anche per le società dedicate, che svolgono questo lavoro in modo professionale. In genere, il rilevamento richiede la scansione delle attività insolite sulla rete, la ricerca dei file sospetti sul file system o il rilevamento di picchi inspiegabili nell'utilizzo delle risorse. Per complicare ulteriormente le cose, oggi i gruppi di hacker cercano di *mimetizzarsi*, imitando i processi esistenti e utilizzando solo servizi accessibili localmente, per evitare di essere scoperti.

Quanto dovremmo essere paranoici?

Gli hacker sono minacce ben presenti nella vita reale, e i risultati dei loro sforzi possono essere catastrofici. Le aziende che subiscono un attacco da parte di hacker vanno incontro a danni finanziari e di reputazione. Dopotutto, chi mai vorrebbe utilizzare un servizio che fa trapelare le vostre informazioni private? Inoltre, una violazione dei dati può avere ripercussioni legali, se si dimostra che la vittima non ha prestato la dovuta attenzione nel proteggere i propri sistemi. È così che gli attacchi informatici hanno portato molte aziende alla bancarotta.

Prima di farvi prendere dal panico, tuttavia, fate un passo indietro e valutate realisticamente il livello di minaccia rappresentato dagli hacker per la vostra azienda o organizzazione. Considerare chi vorrebbe attaccarvi e che cosa potrebbe cercare di fare si chiama *modellazione delle minacce*.

Figura 1.8

La portata della minaccia rappresentata dagli hacker dipende da quanto siete grandi come obiettivo e da che cosa gli hacker potrebbero guadagnare violando i vostri sistemi. Gli enti governativi, i fornitori di energia e i servizi finanziari sono obiettivi di alto profilo. Anche qualsiasi settore che archivi informazioni riservate, come la sanità o l'istruzione, è ad alto rischio. Anche la dimensione della vostra organizzazione è un fattore; ottenere l'accesso alla rete di una grande azienda (in gergo *big-game hunting*) è molto redditizio per un hacker. Se lavorate per un'organizzazione che opera in uno di questi settori, molto probabilmente potrete contare su un team di sicurezza interno che controllerà i sistemi e monitorerà gli accessi sospetti. Questo team si assumerà parte dell'onere di considerare i rischi per la sicurezza, consentendovi di concentrarvi sulla scrittura di codice sicuro. Se mai verrete convocati a una riunione a porte chiuse per discutere di un *evento con priorità zero* (*P0*), sappiate che il team di sicurezza della vostra azienda ha applicato una matrice di modellazione delle minacce e ha ritenuto che qualcosa costituisse una minaccia critica. Gli hacker, tuttavia, sono opportunisti e utilizzeranno strumenti per cercare in Internet dei server web affetti da vulnerabilità note, indipendentemente dalla loro importanza. Questo tipo di scansione delle vulnerabilità è qualcosa di cui voi, come sviluppatori, dovreste preoccuparvi. Dovreste anche cercare eventuali difetti nella vostra base di codice che potrebbero essere sfruttati, come funzioni di autenticazione malfunzionanti o una mancanza di controllo degli accessi. Il solo fatto di prendere precauzioni, come correggere le vulnerabilità più evidenti nel codice e rendersi un bersaglio più difficile, spesso fa sì che gli hacker passino a prede più facili.

Da dove iniziare a proteggersi?

Questo libro intende fare da guida nello scrivere codice sicuro e rilevare le vulnerabilità presenti nelle vostre applicazioni web. Potete leggerlo tutto o approfondire i capitoli che ritenete più rilevanti per voi, ma ciò vi darà certamente un vantaggio sulla protezione delle vostre app. Tuttavia, probabilmente desiderate iniziare il vostro viaggio verso la sicurezza fin da subito, quindi questo paragrafo presenta alcune cose che potete iniziare a fare mentre approfondite il resto del libro.

Tenete traccia delle nuove vulnerabilità

Le vulnerabilità *zero-day* descrivono i problemi di sicurezza appena resi pubblici (in altre parole, sono passati "zero giorni" dalla loro divulgazione). Gli hacker coglieranno l'opportunità di sfruttare le vulnerabilità zero-day, quindi spetta al vostro team tenere traccia delle nuove vulnerabilità e applicare le patch di sicurezza non appena diventano disponibili. Quando viene annunciato una vulnerabilità zero-day, inizia per voi una corsa contro il tempo.

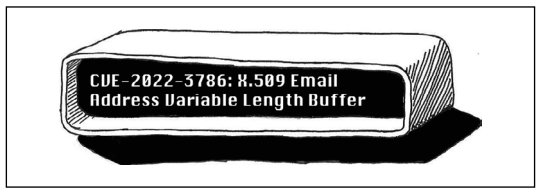

Figura 1.9

I social media e i siti di notizie sono dalla vostra parte se volete tenervi aggiornati sugli avvisi relativi alla sicurezza. X e Reddit vi terranno aggiornati se seguite i leader nel campo o vi iscrivete ai sub-Reddit. Le vulnerabilità più gravi, come *Log4Shell*, una vulnerabilità legata all'esecuzione di codice in modalità remota della libreria di logging di Java `Log4J`, fanno notizia su tutti i principali siti tecnologici, come TechCrunch e Ars Technica.

Scoprite quale codice state distribuendo

Per mantenere sicura la vostra applicazione web, dovete sapere quale codice esegue. È impossibile sapere quali librerie vulnerabili vengono richiamate dal vostro codice e, quindi, quali patch dovete applicare, a meno che non sappiate quali *dipendenze* sono state usate nel processo di release. Il Capitolo 5 illustra come eseguire la distribuzione dal sistema di controllo del codice sorgente e come utilizzare un gestore delle dipendenze. Se non riuscite a determinare a colpo d'occhio quale codice è in esecuzione sulla vostra applicazione web, rendete una priorità la risoluzione di questa situazione.

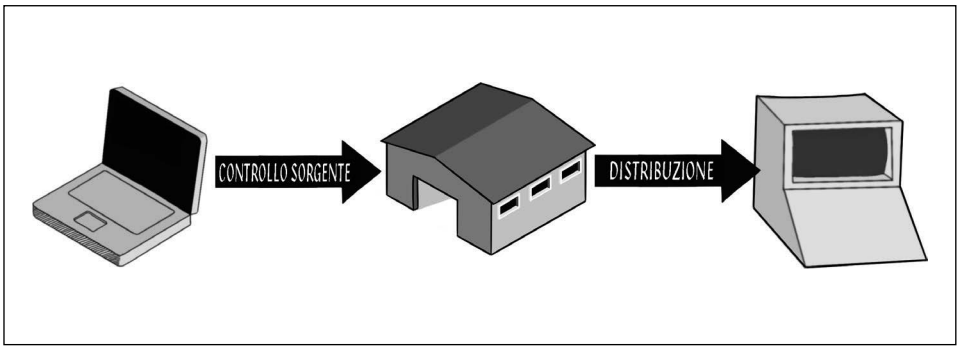

Figura 1.10

Registrate e monitorate l'attività

Potreste non sapere mai di essere stati vittima di un hacking, a meno che non disponiate di informazioni sufficienti per diagnosticarlo. Dovreste sempre essere in grado di visualizzare i registri in tempo reale (log) di un'applicazione web, per osservare come vengono effettuati gli accessi. Il vostro codice dovrebbe rilevare e segnalare gli errori imprevisti. Infine, dovreste avere un sistema di monitoraggio su ciascuna applicazione web, in modo da poter vedere quante richieste gestisce al secondo e il tempo medio di risposta della vostra applicazione. La registrazione sui log, la segnalazione degli errori e il monitoraggio aiutano anche con l'*analisi forense*, per capire a posteriori come un hacker sia riuscito a violare i vostri sistemi.

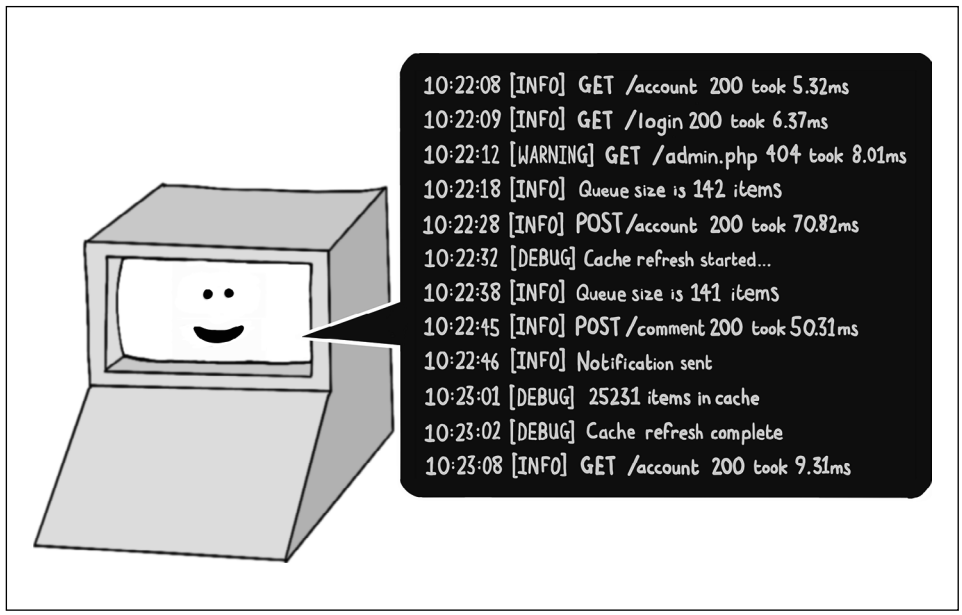

Figura 1.11

Trasformate i membri del vostro team in esperti di sicurezza

La migliore difesa contro gli attacchi degli hacker consiste nell'avere un intero team alla ricerca di incidenti della sicurezza e potenziali vulnerabilità. Le revisioni del codice possono individuare problemi relativi alla sicurezza prima che il codice stesso venga rilasciato e il fatto di avere un intero team di sviluppatori ben addestrati che effettuano controlli incrociati sul lavoro degli altri vi metterà in una posizione di maggior sicurezza. Incoraggiate i vostri colleghi a rispolverare le loro conoscenze sulla sicurezza e a esprimere sempre le loro idee sui potenziali problemi relativi alla sicurezza nelle riunioni del team.

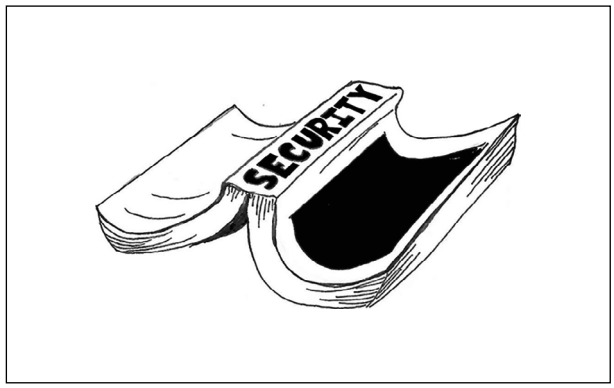

Figura 1.12

Rallentate

Spesso, i problemi relativi alla sicurezza a livello di codice si verificano quando un team ha troppa fretta di rispettare le scadenze. Assicuratevi che il ciclo di vita del vostro sviluppo vi conceda il tempo di condurre revisioni e analisi attente del codice, soprattutto se state rielaborando *codice legacy*, ovvero codice scritto da qualcuno che ormai è passato ad altre società o progetti. Può essere difficile considerare la sicurezza a fronte di scadenze troppo ravvicinate, ma questa attività richiede sicuramente meno tempo che affrontare le conseguenze di un hacking.

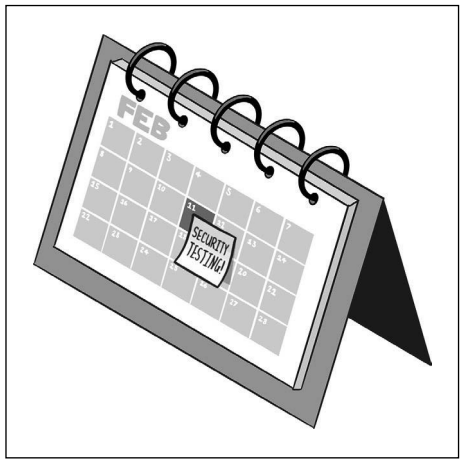

Figura 1.13

Riepilogo

- Gli hacker prenderanno di mira le vostre applicazioni web per ottenere un guadagno finanziario, la notorietà o per ragioni politiche.
- Gli hacker utilizzano vari tipi di strumenti e tecniche sofisticate, e la vendita di dati sottratti o la distribuzione di ransomware può essere un'attività redditizia.
- Se il vostro sito web viene violato, potrebbe essere messo offline, potrebbero violare i vostri dati, potrebbero prendere di mira i vostri utenti o potrebbero infettare di bot i vostri server.
- Il vostro profilo di rischio dipende dalle dimensioni della vostra azienda e del vostro settore, ma nessuno è al riparo da una scansione delle vulnerabilità.
- Tenete traccia delle vulnerabilità, tenete traccia delle dipendenze, assicuratevi che il sistema sia per voi trasparente, educate il team di sicurezza e inserite tutte le patch nel ciclo di vita dello sviluppo e otterrete vantaggi immediati.