DISCLAIMER

Il materiale contenuto nel drive è stato raccolto e richiesto tramite autorizzazione ai ragazzi frequentanti il corso di studi di Informatica dell'Università degli Studi di Salerno. Gli appunti e gli esercizi nascono da un uso e consumo degli autori che li hanno creati e risistemati per tanto non ci assumiamo la responsabilità di eventuali mancanze o difetti all'interno del materiale pubblicato.

Il materiale sarà modificato aggiungendo il logo dell'associazione, in tal caso questo possa recare problemi ad alcuni autori di materiale pubblicato, tale persona può contattarci in privato ed elimineremo o modificheremo il materiale in base alle sue preferenze.

Ringraziamo eventuali segnalazioni di errori così da poter modificare e fornire il miglior materiale possibile a supporto degli studenti.

CoScienze
Associazione

Alfonso Cannavale

CENNI STORICI E TERMINOLOGIA

Hacker Classificazione

- **Cracker**: programmatori specializzati nell'infrangere sistemi di sicurezza per sottrarre o distruggere dati.
- **Script Kiddie**: cracker che adoperano script scritti da altri, non essendo in grado di produrli da sé.
- **Phracher**: rubano programmi che offrono servizi telefonici gratuiti o penetrano computer e database di società telefoniche.
- **Phreaker**: utilizzano informazioni telefoniche (numeri telefonici, carte telefoniche,...) per accedere ad altri computer.

CERT: Computer Emergency Response Team

Il CERT è un team di esperti nell'ambito della sicurezza che si occupa di identificare il tipo di incidenti, quantificare le perdite economiche e analizzare le vulnerabilità dei prodotti.

Virus poliformico

Un virus polimorfico è un tipo di malware che si distingue per la sua capacità di mutare il proprio codice in modo continuo, rendendolo difficile da rilevare e combattere per gli antivirus. Questi virus hanno la caratteristica di modificare la propria struttura o comportamento ad ogni infezione, il che li rende molto più sfuggenti rispetto ai virus tradizionali.

Per raggiungere questa capacità camaleontica, il virus polimorfico utilizza una tecnica di cifratura del proprio codice con una chiave diversa ogni volta che infetta un sistema. Questa chiave è essenziale per decifrare il codice e far eseguire le istruzioni del virus al momento opportuno. La particolarità sta nel fatto che la chiave necessaria per decifrare il codice è conservata internamente al virus stesso, rendendola accessibile solo al momento dell'esecuzione.

Il risultato di questo processo è che il virus polimorfico può assumere forme diverse ad ogni nuova infezione, con un codice apparentemente diverso ogni volta. Questa costante mutazione rende estremamente difficile per gli antivirus rilevare e bloccare il malware utilizzando le tradizionali signature note. Poiché il codice del virus appare diverso ad ogni esecuzione, i motori antivirus faticano a identificarlo in modo efficace, aumentando così la sua efficacia nel diffondersi e nel compromettere sistemi informatici.

Macro Virus

Un macro virus è un tipo di virus informatico che infetta i file di documento che contengono macro, come ad esempio documenti di Microsoft Word o fogli di calcolo di Excel. Questi virus sfruttano le macro, che sono sequenze di comandi che automatizzano compiti all'interno di documenti, per diffondersi e danneggiare i sistemi informatici. Una volta attivati, i macro virus possono eseguire azioni dannose come la cancellazione di file, la modifica dei dati o la diffusione della propria copia ad altri documenti.

DOS e DDOS

Un attacco ***DoS (Denial of Service)*** è un tentativo di rendere un servizio o una risorsa non disponibili agli utenti legittimi, sovraccaricando il sistema bersaglio con un alto volume di richieste. Questo può causare il blocco del servizio, impedendo agli utenti di accedervi. D'altra parte, un attacco ***DDoS (Distributed Denial of Service)*** è simile a un attacco DoS, ma coinvolge molteplici dispositivi distribuiti in rete, noti come "botnet", che coordinano e amplificano gli sforzi per sovraccaricare il sistema bersaglio con traffico dannoso. Questo rende molto più difficile mitigare l'attacco, poiché proviene da molteplici fonti e può essere più difficile da individuare e bloccare.

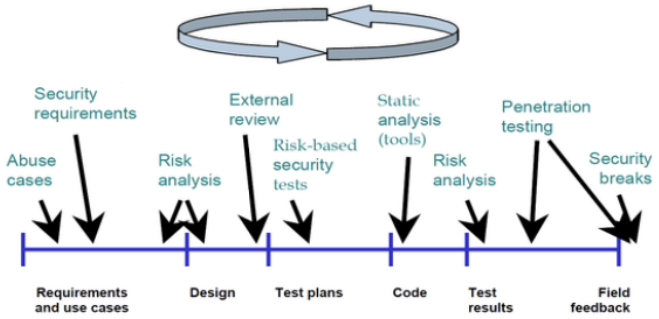
Phishing

Il phishing è una forma di attacco informatico in cui gli aggressori cercano di ingannare le vittime per ottenere informazioni confidenziali, come username, password o dati finanziari, o per installare software dannoso sul loro dispositivo. Questo avviene di solito attraverso l'invio di e-mail, messaggi istantanei o siti web contraffatti che sembrano provenire da fonti affidabili o di fiducia. Gli aggressori utilizzano tecniche di ingegneria sociale per convincere le vittime a fornire queste informazioni sensibili, ad esempio facendosi passare per istituti finanziari, società di servizi o siti di social media. Una volta che le vittime cadono nella trappola del phishing, i loro dati possono essere utilizzati per frodi finanziarie o per altri scopi dannosi.

LEZIONI APPRESE

- Il termine *hacking* ha cambiato radicalmente significato nel tempo.
    - Da desiderio di manifestare la propria superiorità a un complesso insieme di attività per ottenere un vantaggio economico e politico.
- L'hacking in stile black hat finisce quasi sempre nello stesso modo (con una denuncia, un processo o una condanna penale).

- Bisogna imparare a difendersi e per farlo è necessario capire come avvengono gli attacchi.
- Difendersi è più complesso rispetto ad attaccare.
    - All'attaccante può bastare una singolo falla, ma il difensore deve individuare tutte le falle e ripararle.
    - L'attaccante può scegliere metodi e obiettivo dell'attacco, mentre il difensore deve adattarsi all'attaccante.
    - L'attaccante spesso è visto come un eroe se ha successo, mentre il difensore è visto come un perdente se fallisce.
    - L'attaccante conosce bene gli strumenti che usa, mentre il difensore può scontrarsi con tecniche mai viste prima.
- Gli obiettivi di un attaccante sono molteplici: un apparato hardware, un software (sistema operativo, libreria o applicazione), una procedura o algoritmo.
- Il programmatore deve considerare minacce e obiettivi in tutto il ciclo di vita del software.



- La storia si ripete e gli errori commessi sono quasi sempre gli stessi, quindi le risposte agli incidenti sono quasi sempre le stesse.
    - Il programmatore ha a sua disposizione una grande arma: la storia… che gli insegna quello che NON deve fare.

TERMINOLOGIA

- **Asset:** è un'entità generica che può interagire col mondo circostante e la sua natura dipende dal contesto. Potrebbe essere un dispositivo hardware, software, un dato, un algoritmo o anche una persona. Se l'asset è un software allora esso potrà interagire con l'utente ed è importante capire in che modo avviene l'interazione. Un utente può interagire con un asset in tre modi: *correttamente*, *non correttamente in modo involontario* o *non correttamente in modo malizioso*. Un uso non corretto di un asset può comportare rischi gravi, tra cui: furto di dati sensibili (password o conti correnti), beni preziosi o denaro, modifica o distribuzione di informazioni sensibili, compromissione di servizi.
- **Minacce (threat)**: è una potenziale causa di incidenti, il risultato è un danno all'asset. Le minacce possono tramutarsi in due modi: accidentale o doloso. Microsoft propose una classificazione delle minacce sotto il nome di ***STRIDE***:
    - ***S**poofing*, quando un utente si spaccia per un'altra entità;
    - ***T**ampering*, modifica non autorizzata delle informazioni;
    - ***R**epudiation*, ripudiare o negare l'esecuzione di un azione;
    - ***I**nformation Disclosure*, divulgazione di informazioni sensibili;
    - ***D**enial of Service*, negazione di un servizio;
    - ***E**levation of Privilege*, elevazione dei privilegi.
- **Attaccante:** colui che interagisce con l'asset in modo malizioso e doloso, cercando di ottenere un fine. L'attaccante è alla ricerca di un malfunzionamento sfruttabile che si possa tramutare in un exploit. Ha lo scopo di tramutare una minaccia in realtà, motivato dal conseguire un vantaggio. Esistono vari tipi di attaccanti tra cui:
    - *White hat (ethical hacker)*: viola asset per fini non maliziosi, come stimare il livello di sicurezza;

    - *Black hat*: viola asset per fini maliziosi o per tornaconto personale;
    - *Gray hat*: viola asset e, in cambio di denaro, si offre di irrobustirli;
    - *Hacktivist*: viola asset per fini ideologici, politici o religiosi. Svolge attività di cyber terrorismo e rende accessibili al pubblico documenti confidenziali;
    - *Nation state*: team di attaccanti sponsorizzati da una nazione;
    - *Organized criminal gang*: team di attaccanti che viola asset per profitti illegali.
- **Bug**: è un errore di implementazione dell'asset.
- **Difetto**: è una deviazione dell'asset da requisiti e specifiche di progetto.
- **Debolezza (weakness)**: è un difetto che potrebbe rendere reale una minaccia, la presenza di una debolezza non indica necessariamente che l'asset possa essere compromesso (deve essere raggiunto dall'attaccante e deve poter essere violato).
- **Vulnerabilità**: è una debolezza che un attaccante è in grado di usare per tramutare una minaccia in realtà. È la somma di tre fattori:
    - *debolezza esistente*;
    - *accessibilità* dell'attaccante alla debolezza;
    - *capacità dell'attaccante di sfruttare la debolezza* per conseguire un vantaggio.
- **Exploit**: è una procedura che è in grado di sfruttare una vulnerabilità (permette di trasformare una minaccia in realtà). Una volta che l'attaccante ha individuato una debolezza, la trasforma in una vulnerabilità sfruttabile, dopodiché è in grado di causare un comportamento inatteso dell'asset, ottenendo un vantaggio.
- **Vettore di attacco**: è uno strumento qualsiasi attraverso il quale si può veicolare una vulnerabilità (modalità di attacco). Può essere una connessione TCP verso un server, una shell locale, una linea telefonica incustodita.
- **Superficie di attacco** di un asset: è l'insieme di tutti i suoi vettori di attacco e misura l'esposizione dell'asset agli attacchi.
- **Politica di sicurezza (security policy)**: definisce in modo non ambiguo il livello di sicurezza di un asset. Risponde ad una serie di domande del tipo: Che significa che "l'asset è sicuro"? Da quali interazioni ci si vuole difendere? Da quali utenti ci si vuole difendere?. La politica di sicurezza nasce spesso da un'***analisi dei rischi (risk analysis)*** che cerca di identificare cosa potrà andare storto con l'asset, quanto sarà probabile un incidente o quanto costerà un incidente.
- **Meccanismi di sicurezza**: è uno strumento che consente di attuare una politica di sicurezza. Ci sono tre diverse categoria:
    - *Meccanismi di prevenzione*, tendono ad impedire le interazioni tra un asset e un utente, come un firewall;
    - *Meccanismi di rilevazione*, controllano le interazioni tra un asset e un utente;
    - *Meccanismi di reazione*, per ripristinare il sistema in seguito ad un incidente.

  Operazioni tipiche dei meccanismi di sicurezza sono:
    - *Autenticazione*, verificare che l'utente che si presenta all'asset è effettivamente chi dice di essere;
    - *Controllo degli accessi*, verificare che l'utente ha i diritti per accedere all'asset;
    - *Auditing*, monitorare e registrare le interazioni di un utente con l'asset;
    - *Azione*, svolgere azioni correttive per far rispettare la politica di sicurezza.
- **Prevenzione**: un *asset soggetto a prevenzione* non è in grado di interagire con nessuno. L'aspetto positivo è che non è attaccabile dai malintenzionati, ma l'aspetto negativo è che non è utilizzabile anche dagli utenti normali.
  È necessario un *aumento dell'esposizione dell'asset* affinchè gli utenti possano fruirne, come l'apertura di porte TCP in un servizio di rete, piuttosto che disconnessione totale, e una lettura di input in una applicazione locale, piuttosto che nessun input.
- **Rilevazione**: con l'aumento dell'esposizione aumentano anche i rischi. I rischi vanno controllati con *meccanismi di rilevazione,* come il controllo del traffico sulle porte TCP aperte o il controllo degli input passati a una funzione.

Asset e sicurezza

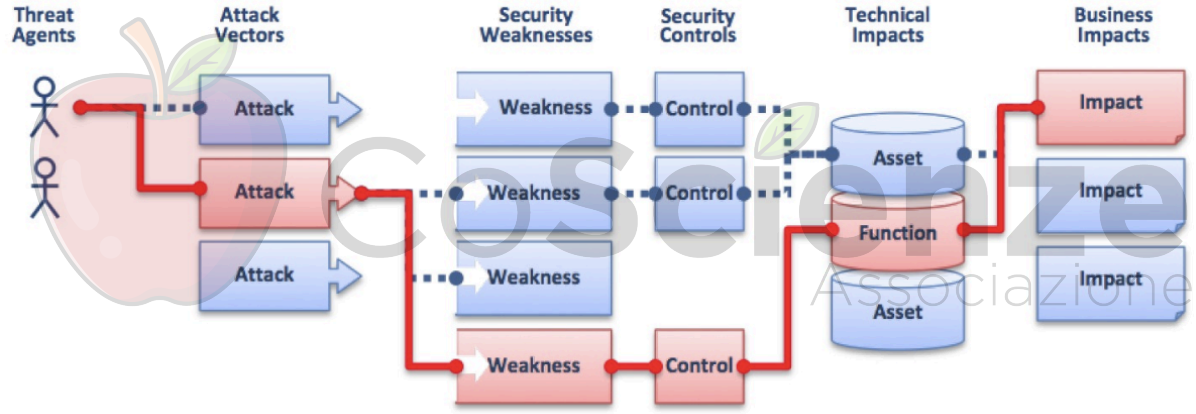
Per quanto riguarda la ***sicurezza di un asset***, dipende da che funzionalità offre un asset. Un asset non esposto al pubblico e che non offre funzionalità è più sicuro, ma molto inutile. Pertanto, è desiderabile un asset che offre funzionalità ma che sia ben protetto, questi tipi di asset però possono incorrere in abusi non autorizzati -> le funzionalità esposte da un asset implicano in *rischio di abuso*. Esempi di abuso includono la violazione della triade ***CIA***:

- ***Confidentiality***: è stato causato un accesso in lettura non autorizzato, ovvero l'utente sfruttando l'asset riesce ad accedere a dati sensibili. Pertanto, bisogna impedire l'interazione in lettura tra un asset e un utente non autorizzato.
- ***Integrity***: è stato causato un accesso, non solo in lettura, ma anche in scrittura, non autorizzato. Pertanto, bisogna impedire l'interazione in scrittura tra asset e un utente non autorizzato.
- ***Availability***: bisogna rendere disponibili le funzioni di un asset a utenti esplicitamente autorizzati.

La proprietà più importante da considerare è l'*integrità* rispetto alla confidenzialità, ad esempio in un software che gestisce un conto corrente non si vuole rendere noto all'attaccante quante soli si ha sul conto, ma soprattutto non si vuole concedere all'attaccante eventuali modifiche sul conto corrente.

In alcuni scenari la *disponibilità (availability)* può essere di intralcio, cioè la privacy di un utente può implicare la mancata disponibilità di informazioni e servizi a terzi, in quanto per proteggere la privacy si impedisce a terzi di utilizzare determinati servizi.

Come opera un attaccante



Un attaccante individua una vulnerabilità nella sicurezza di un sistema che può sfruttare per ottenere accesso non autorizzato (un attaccante individua in una superficie di attacco un vettori di attacco e può sfruttare una debilezza nella sicurezza del sistema -> se sfruttabile diventa una vulnerabilità). Se l'attaccante ottiene il controllo dell'asset, questo potrebbe portare al furto di dati, interruzione dei servizi o altri danni. In sostanza, l'attaccante sfrutta le debolezze nel sistema per ottenere il controllo e causare danni o rubare informazioni.