Recentemente, per lavoro, ho preso parte a un'operazione di Red Teaming. Ho trovato l'esperienza interessante ed emozionante poiché è l'unica tipologia di test in cui si possono mettere a frutto tutte le competenze e spaziare nella tipologia di attacchi da effettuare e di bersagli da colpire.

Eseguire l'operazione all'insaputa del Blue Team è quello che caratterizza e diversifica un'operazione diRed Teaming da un normale vulnerability assessment o penetration test.

Il test viene eseguito in modalità black box da parte del team d'attacco che non possiede quindi una conoscenza a priori dell'infrastruttura bersaglio, delle sue contromisure e degli assets critici. È in tutto e per tutto uno scenario reale, il Red Team è posto sullo stesso piano di un attaccante e non gli viene assegnato uno scope preciso e restrittivo quale ad esempio la subnet x o il sito web y.

Inutile dire che, parte fondamentale alla riuscita di una buona operazione di Red Teaming, è far si che il team attaccante rimanga quanto più invisibile e trasparente al Blue team dell'azienda attaccata (Fly under the radar), evitando pertanto di far scattare "allarmi", lasciare tracce nei log, essere rilevati da varie soluzioni quali Firewall, IDS/IPS, HIPS e AV o dal team di difesa.

Quest'articolo si concentrerà pertanto su alcune delle tecniche utilizzate durante l'attacco.

Tecniche Evasive

Dopo aver compromesso una macchina, è necessario mantenere un accesso persistente alla rete, a tal fine è risultata cruciale la scelta dei payload di Metasploit.

In questo caso è stata scelta una shell http dal nome reverse_winhttps, analizziamo assieme il perché di questa scelta.

reverse_winhttps

- Reverse: Le aziende utilizzano soluzioni Firewall con regole più o meno restrittive e, triste ma vero, molto spesso l'unica configurazione presente è rifiutare tutte le connessioni in entrata, specialmente se la richiesta non è stata originata da un computer all'interno della rete.

Fondamentale per la riuscita dell'attacco, una volta compromessa la macchina remota, è quindi ottenere una shell di ritorno. L'impostazione di default prevede che siano gli attaccanti a collegarsi direttamente allashell incontrando in questo caso il blocco del firewall. Impostando l'opzione reverse otteniamo che sia la macchina compromessa a comportarsi da client e a contattare il server di C&C (comando & controllo) degli attaccanti. (E' lo stesso metodo utilizzato dalle botnet)

- Win: In questo caso la macchina target aveva come sistema operativo Windows e il processo che stavamo "exploitando" aveva le caratteristiche necessarie per l'utilizzo di questo metodo.
  Questo metodo permette di caricare il payload come DLL all'interno del processo stesso e quindi di non "spawnare" un altro processo, il che è molto utile per evitare white list di applicazioni presenti sull'host, per evadere alcuni HIPS (Host Intrusion Prevention System) e alcuni Anti Virus che mal supportano la scansione della memoria; specie se il processo è autorizzato, firmato e solo in un secondo momento viene caricata la DLL al suo interno.
- HTTPS: Le aziende più attente alla sicurezza utilizzano anche altre regole per filtrare il traffico in uscita, in maniera tale da permettere solo alcune porte/da alcuni applicativi/con alcuni protocolli. In questa maniera vengono bloccati tutti gli applicativi che provano a comunicare su Internet (riducendo sensibilmente la finestra d'attacco), per esempio: FTP, IRC, TOR, P2P e una gran parte di agent malevoli. La shell HTTP è programmata per ricercare, a intervalli regolari, la connessione con gli attaccanti e la tipologia di comunicazione scelta fa si che il traffico appaia ai dispositivi disposti a protezione del perimetro come una navigazione in internet composta da richieste e risposte.
- FUD: Durante il test, per evitare che i vari prodotti Anti Virus riconoscessero il payload diMetasploit, abbiamo impiegato alcune tecniche (che vi racconterò in un articolo successivo) per offuscare il payload e renderlo pertanto estraneo e irriconoscibile ai database delle firme dei suddetti prodotti.

Alcune delle funzionalità avanzate di questa tipologia di shell sono

- La possibilità di utilizzare proxy.
- Il payload ha una scadenza “hardcoded” al suo interno, l’impostazione di default è una settimana dalla data in cui viene generato, questo per evitare che una connessione dimenticata provi a collegarsi all’infinito. Una volta raggiunto questo valore la shell verrà terminata. ImpostareSessionExpirationTimeout a 0 farà sì che la shell tenti la connessione fintanto che il processo non viene terminato o la macchina target riavviata.

Uscire da una sessione normalmente comporta “killare” l’exploit ma utilizzando il comando detach (avendo cura d’impostare la variabile SessionCommunicationTimeout a 0 in fase di generazione del payload) farà si che la connessione non venga terminata e sia recuperabile; sempre che non venga raggiunta la condizione SessionExpirationTimeout o che il processo venga killato.

Alcune aziende effettuano ulteriori controlli, monitorando il traffico di rete e assicurandosi che il contenuto dei pacchetti in uscita sia accettabile (Egress Filter). Questi controlli sono in grado d’identificare numeri di carte di credito, utenze, dati di login e svariati pattern; tutto questo per accorgersi in tempo reale di eventuali brecce all’interno della rete.

Se vengono visti transitare dati di carte di credito, alcuni di questi controlli interrompono il traffico e avvisano gli amministratori di rete, presupponendo una breccia interna che ha permesso il dump e l’exfiltration dei dati. Per ovviare a questi problemi abbiamo utilizzato una shell HTTPS, cifrando così la connessione e rendendo i dati illeggibili a questi strumenti, riducendo al minimo il rischio di essere individuati.

ICMP Tunnel

Generalmente le aziende bloccano i pacchetti ICMP in entrata ma spesso permettono gli stessi in uscita, grazie a questa configurazione un attaccante può utilizzare pacchetti ICMP per trasferire payload TCP(ICMP Tunnel).
Schema

1. attaccante: connessione TCP a un software PROXY che invia la richiesta TCP al client remoto.
2. client remoto: incapsula payload TCP in pacchetti ICMP ECHO e li invia al PROXY.

3. PROXY: de-incapsula i pacchetti e invia le risposte TCP all'attaccante.

Tra gli strumenti utili per utilizzare questa tecnica troviamo:

- Ping Tunnel
- ICMP Shell
- icmpsh

Parametri TCP

Un metodo che risale agli arbori di internet ma che funziona ancora piuttosto bene per bypassare IDS e IPS è utilizzare i parametri inutilizzati dei pacchetti TCP .

Alcuni dei campi che possono essere utilizzati per questo scopo sono:

- IP Identification: E' necessario aver già stabilito una sessione tra le due parti, dopodiché i dati vengono trasferiti bit a bit all'interno di questo campo.
- TCP initial sequence number: Questo metodo non richiede neanche di stabilire una connessione.
  un pacchetto SYN viene inviato con l'initial sequence number contenente il payload.
  Anche se la risposta è RST il contenuto è già stato estratto.

TCP initial acknowledgement sequence number: più complesso del metodo precedente, è necessario utilizzare un bounce server il cui unico scopo è ricevere i pacchetti e inoltrarli alla macchina dell'attaccante.

- Il client genera un pacchetto TCP SYN con sorgente, l'indirizzo del server degli attaccanti (IP address spoofing) e con destinazione l'indirizzo del bounce server.
- Il valore dell'Initial Sequence Number (ISN) contiene il carattere codificato (ISNq).
- Il Bounce server riceve il pacchetto e risponde con SYN/ACK o RST, dipende dal fatto che la porta sia aperta o chiusa. La risposta viene inviata al server ricevente (quello dell'attaccante) poiché è stato "spoofato" il suo indirizzo.
  La risposta sarà in questo formato SYNB, ACK(ISNq+1).
- Il server dell'attaccante riceve il suddetto pacchetto e recupera il valore dal campo.

Stato dell’arte

Il termine Red Team, ma questo vale quasi esclusivamente per gli USA, include attacchi che non comprendono solo l’intrusione del network target ma che prevedono anche l’utilizzo di altre tecniche quali Social Engineering , Phishing e accesso fisico alla struttura (es. dumpster diving, il superamento di postazioni di controllo ecc).
Magari in futuro bughardy ci racconterà qualche sua esperienza con Opposing Force .

E’ molto difficile portare a termine un’operazione di Red Team in Italia poiché tutto ciò che è Social Engineering incontra una forte resistenza da parte delle singole aziende e dei sindacati, non è infatti possibile tracciare e identificare in maniera univoca il singolo dipendente caduto vittima dell’attacco e il budget per queste operazioni è ridicolo; basti pensare che in un azienda che conta un migliaio di dipendenti vennero “distribuiti” soltanto 5 device USB infetti.

Per concludere, fly under the radar, stay undetected.

Commento Finale

Crediamo che sopratutto in Italia, nell’ultimo periodo, sia morto l’undergound e l’hacking, non per mancanza di idee o capacità ma perchè sono venuti a mancare due requisiti a nostro parere
fondamentali: il punto di ritrovo e la condivisione.
VoidSec quindi si propone di dare a tutti gli appassionati di Sicurezza Informatica (tecnici e non), un punto di ritrovo, di scambio d’idee, di condivisione delle stesse.
Un luogo dove chi sa può restituire alla community, dove gli inesperti possono imparare e avvicinarsi al mondo della Sicurezza Informatica…

Il sito Voidsec

Andate a visitare il bellissimo blog di Voidsec http://voidsec.com/ :-)