Anatomia del Denial of Service

Testo di Luca Fortunato realizzato nell'ambito del corso di Laurea in Informatica.

1 INTRODUZIONE

Il **Denial of Service (D.o.S.)** è un tipo di attacco che ha lo scopo di mettere fuori uso o impedire un servizio agli utenti legittimi.

Esistono principalmente **3 tipologie di D.o.S:**
1. Consumo delle risorse di un sistema.
2. Distruzione o alterazione di informazioni.
3. Distruzione o alterazione fisica delle strutture di una rete.

Questo documento si concentrerà sull'analisi degli attacchi del primo tipo e fornirà alcune strategie di difesa. Va precisato però che attualmente non esiste nulla che possa essere completamente efficace contro il D.o.S. Qualsiasi sistema connesso ad Internet capace di fornire servizi basati sul protocollo tcp (web server, ftp server, mail server ecc.) é soggetto a questa tipologia di attacco.

Di seguito vengono riportati **due tra i più grossi attacchi perpetrati ai danni di Internet**:

1996: Panyx Internet Provider. Uno dei primi attacchi DoS. L'attacco mise in crisi il sistema per più di una settimana rendendo impossibile l'uso del servizio a circa 7000 utenti.

1999: Yahoo, Inc. Uno degli ultimi attacchi DoS che ha provocato l'interruzione per quasi 12 ore di un servizio usato da milioni di utenti ogni giorno.

2 TIPI DI ATTACCHI D.o.S.

2.1 Consumo delle risorse

In questo tipo di attacco l'aggressore cerca di saturare le risorse del sistema "preda" per renderlo inutilizzabile. Qui di seguito vengono riportate le principali tecniche adottate:

Bandwidth consumption: Consiste nel generare una quantità di traffico tale da consumare tutta la banda a disposizione della preda.

Scenario 1 - l'aggressore ha a disposizione una banda molto più ampia delle vittima.
In questo caso l'aggressore inonda semplicemente la vittima con una quantità enorme di pacchetti nulli.

Scenario 2 - la vittima ha più banda dell'aggressore.
In questo scenario l'aggressore usa delle tecniche di amplificazione del traffico riuscendo così a saturare tutta la banda della vittima (**smurfing**).

Resource starvation: Rispetto al tipo di attacco precedente l'aggressore tende a saturare altre risorse del sistema piuttosto che il Bandwidth. Tali componenti possono essere il tempo di CPU, la memoria di sistema , lo spazio su disco ecc. In generale un sistema sotto questo attacco diventa inutilizzabile oppure collassa.

Bug software: In questo caso l'aggressore **sfrutta dei bachi software/hardware presenti nelle strutture della rete per provocare il crash di un sistema**. Ad esempio l'attacco Ip **fragmentation overlap** si basa su un baco dello stack IP usato per il riassemblaggio della sequenza dei pacchetti. La tecnica consiste nel generare una sequenza di pacchetti costruita ad 'arte' in modo da provocare il crash del sistema.

2.2 Distruzione o alterazione di informazioni

Un intruso che riesca a bypassare i dispositivi di sicurezza del firewall potrebbe essere in grado di alterare o distruggere le informazioni e impedire all'utente di proseguire le proprie attività.

2.3 Distruzione o alterazione fisica delle strutture di una rete

La causa di questo attacco può essere un accesso non autorizzato alle strutture fisiche della rete (routers, firewall e qualsiasi altro componente critico del sistema). È un attacco che avviene dall'interno

3 TECNICHE DI ATTACCO

3.1 Spoofing

L'aggressore modifica il valore del campo sorgente del pacchetto che sta per trasmettere. In questo modo il destinatario verrà fuorviato sulla reale provenienza del pacchetto.

3.1.1 Strategie di difesa

La questione non è semplice, infatti una volta che il pacchetto viene consegnato, il destinatario (vittima) non ha strumenti per capire la veridicità dell'indirizzo sorgente.

La soluzione è possibile solo a monte con router configurati in modo da scartare pacchetti con indirizzi sorgenti estranei.
Un sistema di logging potrebbe inoltre segnalare all'amministratore della rete la presenza di un nodo interno con intenzioni ostili.

L'unico caso in cui la vittima può respingere questo tipo di attacco è quando il pacchetto in ingresso ha come indirizzo sorgente un indirizzo locale.
Un pacchetto proveniente dall'esterno non può avere questo valore e deve perciò essere scartato.

3.2 Smurfing/Ping broadcast

È una tecnica utilizzata per amplificare il traffico sulla rete tramite il protocollo ICMP e il comando di PING.
Il **protocollo ICMP** (Internet Control Message Protocol) viene utilizzato per segnalare al mittente dei pacchetti che sono avvenuti degli errori in fase di trasmissione (tipicamente nel caso di frammentazione di pacchetti IP).
L'attacco si basa sul comando PING.

3.2.1 Attacco

L'aggressore per attaccare il sistema esegue le seguenti operazioni:

- Inserisce nei pacchetti ICMP (da spedire come PING) l'indirizzo IP del sistema che vuole attaccare (**spoofing**).
- Sceglie una rete che diventerà il mezzo per il suo attacco (**rete amplificatrice**).
- Inizia a spedire i pacchetti verso l'indirizzo di broadcast della rete amplificatrice.

3.2.2 Conseguenze dell'attacco

L'attacco precedente avrà le seguenti conseguenze:

- Il router della rete amplificatrice, riceve i pacchetti sull'indirizzo di broadcast e li inoltra verso tutti gli indirizzi IP della sua sottorete. Ogni computer presente nella sottorete riceve un pacchetto ICMP e risponde con un altro pacchetto ICMP diretto al mittente del pacchetto ricevuto. Questo valore è però contraffatto e contiene l'indirizzo della vittima.
NB: il router deve permettere la conversione del broadcast da livello IP a livello MAC in modo da coinvolgere tutte le macchine nella sottorete.
- La vittima riceverà una quantità tale di pacchetti ICMP che non riuscirà a gestire.

NB: L'aggressore puó utilizzare anche piú reti di amplificazione aumentando così il traffico.

3.2.3 Stima della potenza dell'attacco

Supponiamo che l'aggressore faccia uso di un modem a 33.6 Kbps e che utilizzi questa banda per 10 broadcast IP di 10 sottoreti. Supponiamo inoltre che ogni sottorete abbia 100 host attivi (valore normale per un provider).

Ad ogni sottorete giungerà quindi un flusso di 3.36Kbps. Se il router di ogni sottorete effettua la conversione del broadcast a livello IP in broadcast nel livello MAC, allora ogni host risponderà con delle echo reply ad un tasso di 3.36 Kbps.

Si avrà quindi :
n° reti * n° host attivi * banda= 10*100*3.36Kbps=3.36 Mbps di echo reply a cui la vittima dovrà rispondere.

3.2.4 Fraggle Attack

È una variante dello smurfing che utilizza il protocollo UDP al posto di ICMP. La tecnica consiste nel trasmettere un pacchetto UDP verso una porta di "ECHO". Il PC, se opportunamente configurato, risponde con un altro pacchetto UDP "echo reply". Il vantaggio di questo attacco è che genera comunque traffico e funziona anche sulle reti che non permettono il ping broadcast.

3.2.5 Strategie di difesa

Metodo di difesa 1
Il problema si può risolvere bloccando al router la possibilità di tradurre i broadcast IP in broadcast MAC per il traffico entrante.
Ad esempio nei router CISCO basta impostare l'opzione no direct IP-broadcast.
All'interno della sottorete però il broadcasting è ancora funzionante ed un hacker potrebbe penetrare in una macchina della rete ed eseguire un broadcast di ping a tutta la rete interna per attuare il suo attacco DoS.

Metodo di difesa 2
Il secondo metodo si basa sull'idea di analizzare il traffico ICMP sulle macchine della rete. Ogni host della rete viene dotata di uno sniffer che controlla i log e blocca il traffico su una certa porta se questo è costituito da pacchetti ICMP echo reply che superano una certa portata. Il metodo però non è tanto efficace poiché tende ad appesantire il carico di lavoro dell'host. Si potrebbe pensare di spostare il controllo sul router ma questo è possibile solo se questo è stato implementato a livello software su una macchina Unix-like. La maggior parte dei router sono oggi dispositivi hardware progettati per un unico compito e quindi poco configurabili.

Metodo di difesa 3
Il terzo metodo si basa su un sistema analogo a quanto visto per lo spoofing: si configura cioè il router da cui potrebbero partire i pacchetti recanti broadcast IP ed echo request in modo tale da non permetterne l'inoltro su Internet.

3.3 SYN flooding

Il **syn flooding** (inondazione) si basa sul protocollo three way handshake utilizzato da TCP per aprire una connessione.

3.3.1 Three way handshake

Il computer A vuole aprire una connessione con il computer B:
- A inoltra a B un pacchetto SYN
- B riceve un SYN da A esegue le seguenti operazioni:
a) inserisce in una tabella delle connessioni la richiesta pendente di A.
b) spedisce ad A un SYN/ACK.
c) si pone in stato di SYN RECEIVED e attende che A confermi la connessione.

In questo caso tra A e B si ha una **half-connection**.

- A riceve il SYN/ACK da B e spedisce a sua volta un altro ACK ritenendo che la connessione sia stabilita.
- B riceve il secondo ACK di A ed esegue le seguenti operazioni:
a) passa dallo stato SYN RECEIVED a ESTABILISHED
b) cancella la richiesta pendente di A dalla tabella delle connessioni.

NB: il nome del protocollo deriva dal fatto che la connessione tra A e B viene stabilita tramite i tre messaggi: SYN,ACK/SYN,ACK.

3.3.2 Attacco

Nello scenario precedente la vittima dell'attacco potrebbe essere B.
L'aggressore A spedisce a B dei SYN il cui indirizzo sorgente è contraffatto (ip spoofing) in modo tale che i messaggi di risposta SYN/ACK della vittima (B) cadano nel vuoto.
A inonda (flooding) B di richieste di connessione allo scopo di far riempire la tabella delle connessioni pendenti (passo b). In questo modo B esaurisce lo spazio disponibile per gestire nuove connessioni.
Lo scopo dell'attacco è perciò quello di lasciare aperte molte half-connection con la vittima.

3.3.3 Attacco stream

È una variante del SYN flooding che consiste nello spedire pacchetti TCP con SYN-ACK. Dato che questi pacchetti anomali non fanno parte di nessuna connessione, occorre un certo tempo perché possano essere gestiti dalla vittima. Se il numero di tali pacchetti ricevuti è elevato, la vittima può sovraccaricarsi ed andare in crash.

3.3.4 Strategie di difesa

Per contrastare l'attacco Syn-Flooding la prima cosa da fare è eliminare la possibilità che ci siano dei pacchetti contraffatti, quindi vale il metodo precedentemente descritto per far fronte allo spoofing.

Metodo di difesa 1
- aumentare la dimensione della tabella delle connessioni
- decrementare il time-out dell'handshake.

Queste due soluzioni sono poco sfruttate perché l'aggressore puó semplicemente continuare a spedire pacchetti piú velocemente di quanto la vittima abbia bisogno per esaudire tutte le richieste.

Metodo di difesa 2: Proteggere la rete
Capire qual è l'indirizzo sorgente dei pacchetti di flooding e far si che il packet filtering del firewall scarti tutti i pacchetti con l'indirizzo sorgente "ostile".
NB: Evitare di annotare in un file di log i pacchetti scartati.

Metodo di difesa 3: TCP interceptor
Tcp interceptor è un **software** che si installa sul server ed è **studiato per proteggere il sistema dagli attacchi Syn-flooding**.

3.3.5 Tcp Interceptor

Questo software può funzionare in tre modalità:
a) **Intercept mode**
b) **Watch mode**
c) **Drop Mode**

3.3.5.1 Intercept mode (modo attivo)

In questa modalità il software intercetta i pacchetti diretti al server. Nel caso in cui arrivi un SYN (richiesta apertura connessione TCP) da una client, cattura la richiesta e gestisce il three way handshake al posto del server.

Descrizione:
- Il client inoltra al server un pacchetto SYN.
- Il tcp interceptor spedisce un SYN/ACK al client.
- Tcp interceptor esegue le seguenti operazioni:

if (riceve il secondo pacchetto di ACK entro un time-out)
then (stabilisce una connessione con il server)
else (termina la connessione con il client)

Per tutta la durata della comunicazione il software riceve i pacchetti dal client e li rispedisce al server. Sia l'apertura della connessione che la comunicazione di dati è assolutamente trasparente al client e al server.

3.3.5.2 Watch mode (modo passivo)

In questa modalità il software non si frappone tra il client e il server ma controlla solo le richieste di connessione pervenute al server. Se una connessione fallisce (non viene stabilita entro un certo time-out) il software interviene e termina la connessione.

3.3.5.3 Drop Mode (modo di difesa)

Quando si verificano le seguenti condizioni:
1) il numero di connessioni incomplete supera un certo valore
2) Il numero di connessioni nell'ultimo minuto supera un certo valore

Il server è considerato sotto attacco e il TCP Interceptor entra in Drop Mode, cioè il software assume un atteggiamento più aggressivo:
- Ogni nuova connessione causa la cancellazione della connessione parziale piú vecchia
- Il time-out viene dimezzato

Il comportamento del software ritorna in modalità normale solo se sia la condizione 1 che la condizione 2 non sono ulteriormente verificate.

3.4 Distribuited Denial of Service (DDOS)

Questo tipo di attacco può essere di tipo **syn flooding** o **smurfing** e **viene effettuato in modo distribuito**. La strategia consiste nel dividere la potenza di attacco su diverse macchine sparse per la rete, mentre una o più coordinano l'attacco.

Esistono diversi DoS tool per attuare attacchi D.o.S distribuiti come ad esempio:

TFN (tribe flood network)
Questo tool si installa di nascosto (troiano) su molti sistemi i quali vengono gestiti da un hacker per coordinare un distributed DoS contro una o più vittime.

Terminologia:
- Zombi o Droni: I sistemi che eseguono a insaputa dell'utente tool DoS
- Master : L'host da cui parte l'attacco
- Slave o Agente: Il nodo che riceve i comandi dal Master e attua l'attacco
- Client: Il processo TFN installato sul master che impartisce i comandi di attacco
- Demone: Il processo TFN installato sullo slave che esegue i comandi di attacco impartiti dal client

TFN2K
É la piú recente versione di TFN che supporta i sistemi operativi Windows NT e UNIX e contiene nuove caratteristiche per rendere piú difficile l'individuazione.
Altri DDoS tools sono Trinoo e Stacheldraht. Quest'ultimo è l'evoluzione di TFN2K utilizzata per l'attacco a Yahoo.

3.4.1 Caratteristiche di TFN2K

- I comandi impartiti dall'aggressore (master) ai nodi (agenti) possono essere di tipo TCP,UDP,ICMP.
- La vittima può essere attaccata con uno dei metodi precedentemente visti (flooding,smurf).
- A differenza del TFN il demone sul nodo slave è completamente silenzioso, cioè non risponde alle richieste del client con degli ACK per non farsi notare.
- Il client, per essere sicuro che il demone riceva i messaggi, manda ogni messaggio 20 volte.
- I comandi TFN2K sono nella seguente forma :

+(id)+(data)
id : è un campo di un singolo byte contenente l'indicativo di un comando.
data : rappresenta i parametri del comando.

- Tutti i comandi sono criptati usando l'algoritmo CAST-256.
- Tutti i pacchetti degli agenti possono essere contraffatti (spoofing).
- Il nome dei processi zombie (demone e client) viene falsificato in modo tale che anche se l'utente controlla la process-list non si accorge di nulla.

3.4.2 Strategia di difesa contro il TFN2K Prevention:

- Per prevenire il TFN2K si deve usare un firewall con funzione di proxy per tutti i servizi supportati o almeno limitare al minimo i servizi che non possono essere utilizzati tramite il proxy. Il proxy infatti può bloccare tutto il traffico TFN2K.
- Inoltre si deve applicare la tecnica di difesa utilizzata contro lo spoofing poiché i pacchetti in uscita dai nodi slave e diretti alla vittima sono contraffatti.

Per scoprire eventuali attacchi con TFN2K si devono eseguire i seguenti controlli:
- Controllare la presenza di processi demone nella lista dei processi.
- Controllare i pacchetti entranti per verificare che al loro interno non ci sia nulla di "sospetto".
- Una semplice scansione dei file tfn(nel master) e td(nello slave) potrebbe rilevare la presenza del TFN2K. Nella maggior parte dei casi questi file sono rinominati ma sia il demone che l'agente contengono un numero di stringhe che possono essere identificate usando un antivirus.

Alcune di queste stringhe vengono riportate qui di seguito:

TFN2K Client (tfn):

[-P protocol]
[-S host/ip]
[-f hostlist]
[-h hostname]
[-i target string]
[-p port]

TFN2K Demone (td):

fork
ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz0123456789+/
/dev/urandom
/dev/random
%d.%d.%d.%d

Nomi file per Unix:
sh
ksh

Nomi file per Windows NT:
command.exe
cmd.exe

TFN2K Demone e Client (tfn and td):

security_through_obscurity : la definizione di questa funzione viene generata durante la compilazione.
D4 40 FB 30 0B FF A0 9F : Questo byte pattern è presente sia nel client che nel demone e rappresenta i primi 8 byte nella tabella di crittazione del CAST-256.
64 64 64 64 ...: è una sequenza di 128 byte contigui che assumo il valore 0x64. Questi valori rilevano la presenza di una tabella statica utilizzata nel metodo Base 64 per la decrittazione.

4 ALTRE TECNICHE DI ATTACCO

4.1 Confondere il sistema IDS (Intrusion Detection System)

La tecnica consiste nell'inviare alla vittima particolari stringhe che possono essere interpretate dall'IDS del firewall come tentativi di intrusione generando così falsi allarmi. Se la quantità e il tipo di pacchetti sono appropriati il sistema IDS si sovraccarica e può in crash. Il vantaggio di un attacco di questo tipo è che si elimina una possibile difesa della vittima.

4.2 Attacchi basati su routing o DNS

L'attaccante manipola le tabelle di routing in modo da deviare tutto o parte del traffico della rete verso una destinazione diversa o nulla.

Articolo originale di Luca Fortunato