Denial of Service (DoS)

e

Distributed DoS (DDoS)

 1

Tipologie :

- Consumo di risorse computazionali
- Consumo della banda di trasmissione

In entrambi i casi l'effetto e' di impedire la normale operativita' di un sistema, degradandola o bloccandola del tutto.

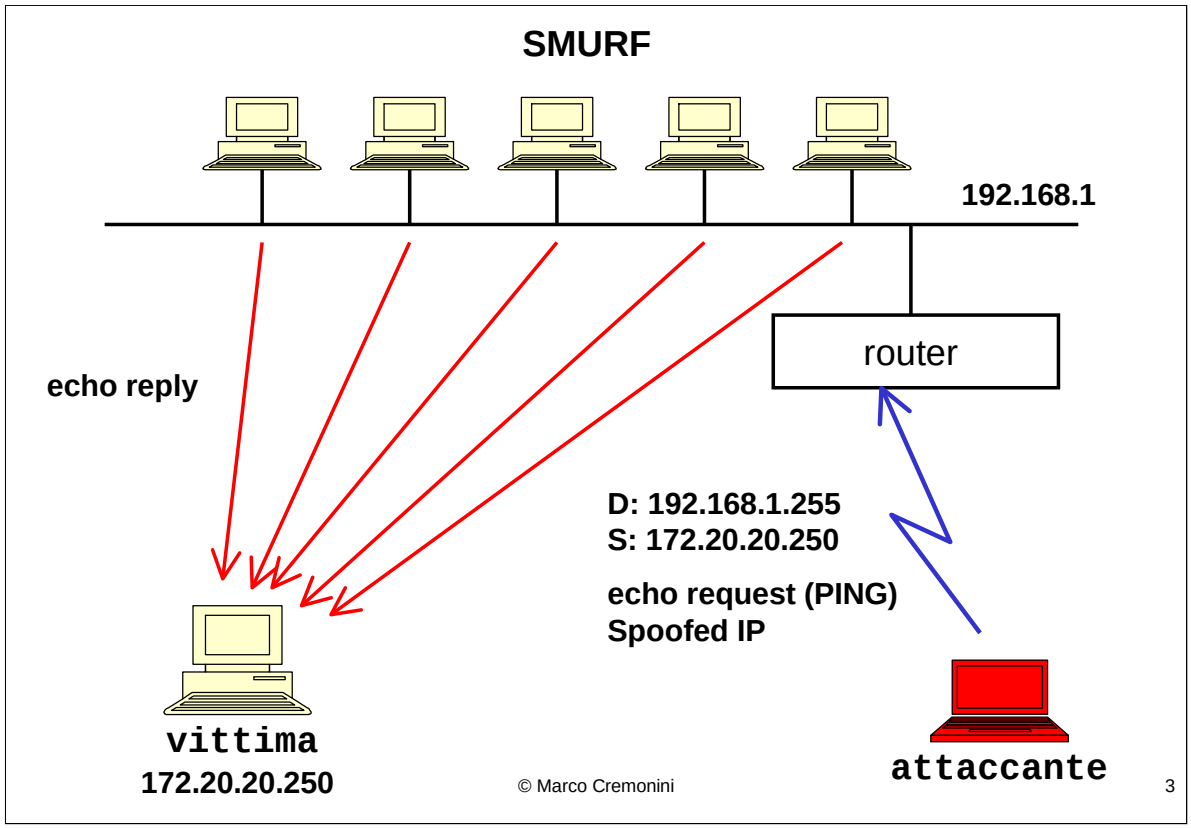
In particolare, si colpisce la capacita' di comunicazione.

 2

Consumo di risorse computazionali: abbiamo gia' visto due esempi, il caso del router BSD che non riesce a gestire la frammentazione e il SYN Flooding di Mitnick.

Consumo della banda di trasmissione: semplicemente, se la connessione di un certo sito supporta al max N byte/sec, viene inviato un traffico di M byte/sec, con M>>N utilizzando, ad esempio dei pacchetti ICMP (echo request, echo reply).

L'obiettivo e l'effetto e' analogo in entrambi i casi.



L'obiettivo di un attacco SMURF e' quello di provocare un denial of service ai danni di una particolare vittima.

Si utilizza la funzionalita' di broadcasting di alcuni host/router per amplificare il traffico in direzione della vittima.

I passi sono quindi:

1) Esistenza di uno o piu' host/router con il servizio di broadcasting attivo;

2) L'attaccante invia un echo request (PING) all'indirizzo di broadcasting (es. A.B.C.255) utilizzando l'indirizzo IP della vittima come sorgente (IP Spoofing);

3) Il PING viene reindirizzato a tutti gli host della sottorete (es. 192.168.1 in figura) i quali rispondono (echo reply) all'indirizzo della vittima.

```
victim.com > 192.168.1.255: icmp echo request
victim.com > 192.168.15.255: icmp echo request
victim.com > 192.168.1.255: icmp echo request
victim.com > 192.168.15.255: icmp echo request
```

DOMANDA: Quanto durano gli effetti di un DoS?

 4

Esempio di pacchetti che la rete che agisce da amplificatore per lo SMURF puo' vedere:

1) Ipotizziamo due sottoreti con broadcast attivo (192.168.1 e 192.168.15);

2) I PING con indirizzo IP della vittima vengono ripetuti per mantenere il traffico di risposte nei confronti della vittima (e quindi gli effetti del DoS).

OSSERVAZIONE

Gli effetti di un attacco di tipo DoS si interrompono non appena cessa l'impulso dell'attaccante (es. i PING nel caso precedente).

Quindi:

DoS → Effetti temporanei, immediatamente pubblico, blocco dell'uso delle risorse

Intrusione → Effetti permanenti, molto spesso non viene resa pubblica, uso illecito delle risorse

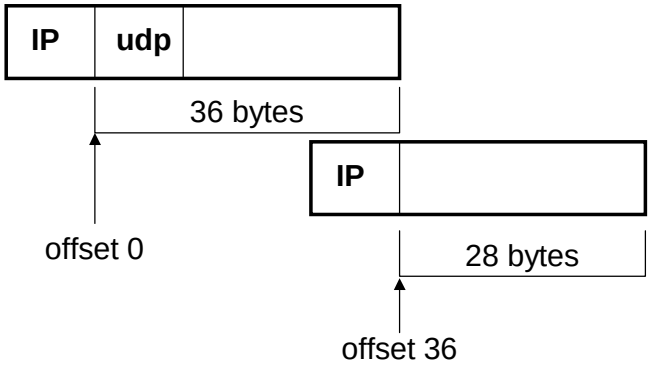
 5

Attacchi di tipo DoS e Intrusioni (compromissioni di sistemi) hanno effetti ed obiettivi molto differenti.

TEARDROP

```
abc.com.45959 > victim.com.3964: udp 28
                                    (frag 242:36@0+)

abc.com.45959 > victim.com.3964: (frag 242:28@24)
```



 6

L'invio di frammenti con offset volutamente inconsistente puo' provocare errori nella gestione della memoria e blocco del sistema (versioni di sistemi operativi non recenti solo).

PING OF DEATH

```
Ping -l 65510 victim.com
```

Dimensione max. pacchetto : 65535 bytes

Header IP : 20 bytes

Header ICMP : 8 bytes

Dimensione max DATI : 65535 -20 - 8 = 65507 bytes

Quindi: al ricevimento dell'ultimo frammento, la dimensione eccessiva puo' provocare crash di sistema.

 7

Anche questo caso nei sistemi operativi piu' recenti e' stato risolto.

Viene comunque ancora rilevato talvolta nei report che mostrano traffico su Internet.

LAND

```
victim.com.80 > victim.com.80 ...

172.20.15.1.31337 > 172.20.15.1.31337 ...
```

CARATTERISTICHE:

- IP Spoofing;
- IP e Porta Sorgente = IP e Porta Destinatario

Questo puo' causare il blocco del sistema operativo.

 8

Stesso commento del caso precedente:

Nei sistemi operativi piu' recenti e' stato risolto.

Viene comunque ancora rilevato talvolta nei report che mostrano traffico su Internet.

Inoltre, e' facilissimo da rilevare.

DISTRIBUTED DoS (DDoS)

- Un DoS che sfrutti l'eccessiva quantita' di traffico ("brute force") e' sempre efficace e molto semplice
- Occorre disporre di un numero adeguato di computer

PROBLEMA: E' possibile disporre di molte computer?

- Moltissimi host facilmente attaccabili in modi semi-automatici (script, worm, email, cross-site scripting, bug)
- Moltissimi computer sempre connessi (utenze professionali, ora ADSL per utenze personali)

A partire dal 1998/1999 gli attacchi di tipo DoS sfruttano architetture distribuite → **DDoS**

 9

Breve storia

1998/1999 – primi tool per DDoS e primi incidenti;

Febbraio 2000 – DDoS (Yahoo!, Amazon.com, CNN, eBay, Buy.com, e*Trade, etc. danni ~$50M)

2000/2001 – moltissimi attacchi DDoS a siti aziendali e governativi (in USA, soprattutto). Molti attacchi a ISP in Europa (es. Tiscali UK e molti altri).

21 Ottobre 2002 – DDoS verso i 13 Root-level DNS. Effetti poco rilevanti (smurf/ping in volume e durata non sufficiente). Grave la dimostrazione di vulnerabilita' dell'infrastruttura Internet.

 10

L'evoluzione degli attacchi di tipo DDoS e' stata quanto mai rapida.

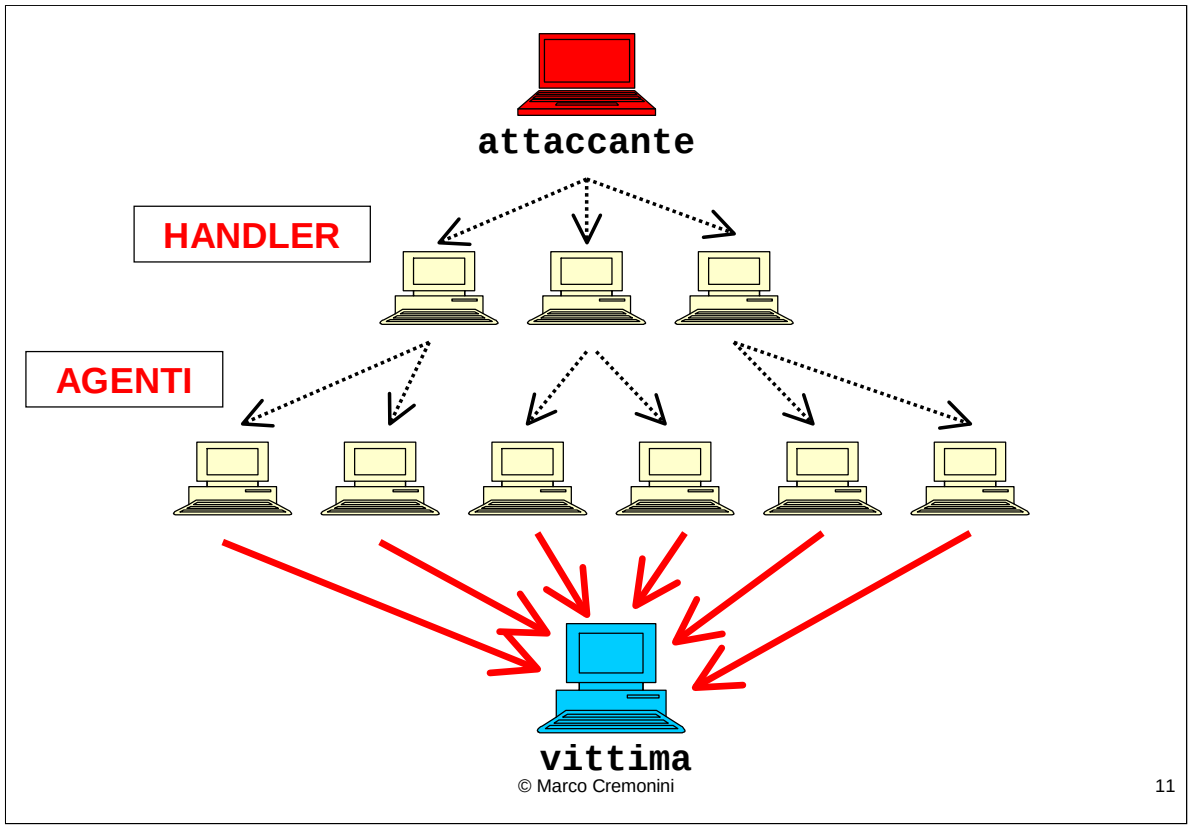
Dalla prima apparizione pubblica di tool e incidenti nel 1998/1999 si e' passati ad un incidente di proporzioni clamorose nel Febbraio 2000 che ha portato all'attenzione mondiale la vulnerabilita' delle societa' interamente basate su eCommerce o eBusiness. Alcune tra le maggiori societa' presenti su Internet hanno subito un attacco coordinato di tipo DDoS che ha reso non operatii i loro collegamenti ad Internet, e di conseguenza i loro siti aziendali, per un arco di tempo di diverse ore.

I danni economici causati da tale interruzione di servizio sono stati stimati essere maggiori di 50 milioni di dollari.

In seguito e' stato individuato un ragazzino canadese quale responsabile dell'azione e incriminato dalla polizia americana. Rimangono forti sospetti sull'esistenza di altri reali responsabili di un attacco di tali proporzioni.

Negli anni seguenti non si sono avuti incidenti di tale proporzione ma una miriade di incidenti minori che hanno coinvolto societa' e siti governativi, negli USA in particolare, e ISP di medie o piccole dimensioni soprattutto in Europa.

Molto recentemente, il 21 Ottobre 2002, un nuovo incidente a portato alla ribalta gli attacchi DDOS. I 13 Root DNS (ovvero i DNS primari che forniscono servizio per tutti gli altri DNS su Internet) hanno subito un attacco che non ha avuto effetti rilevanti. La gravita' del fatto risiede nella dimostrazione di vulnerabilita' dell'intera rete Internet che, se avesse subito un DDoS di proporzioni maggiori, avrebbe potuto essere bloccata a livello mondiale. Sono attualmente in corso indagini per individuare i responsabili. Sono molti gli addetti che valutano quanto accaduto solo come un primo e non ben progettato tentativo di DDoS verso l'intera Internet e, nonostante gli effetti siano stati irrilevanti, lo considerano una dimostrazione pratica della realizzabilita' di tale azione. Alcuni parlano di "inevitabilita'" di un attacco di questo tipo ma di proporzioni ed effetti notevolmente superiori.



Un DDoS viene realizzato attraverso un architettura distribuita di computer e la coordinazione delle loro azioni.

I sistemi compromessi e controllati dall'attaccante sono di due tipi:

HANDLER: gli Handler sono i sistemi (di solito pochi) che ricevono comandi direttamente dall'attaccante e coordinano le azioni degli Agent;

AGENT: gli Agent (possono essere migliaia o decine di migliaia) ricevono comandi dagli handler ed effettuano l'attacco nei confronti della vittima.

Tool per DDoS

Tool	Metodi Utilizzati
Trin00	UDP
TFN	UDP, ICMP, Smurf
TFN2K	UDP, ICMP, Smurf
Stacheldracht	UDP, ICMP, ACK, Smurf
Shaft	UDP, ICMP, Smurf
Mstream	ACK
Trinity	UDP, ICMP, ACK, flags, frammenti

 12

Questi, in ordine cronologico (salvo le varianti), sono i principali tool conosciuti per causare DDoS.

Sono apparsi nell'arco di circa 2 anni.

Caratteristiche dei Tool per DDoS

Metodi di DoS:

- Saturazione banda (Smurf, SYN, UDP, ICMP);
- Pacchetti malformati (Ping of Death, Teardrop, Land, Frammenti)

L'evoluzione piu' rilevante e' stata nei **metodi di controllo e coordinazione** tra attaccante, handler e agenti:

- connessioni TCP e UDP (Trin00, TFN)
- connessioni TCP, UDP e ICMP criptate (TFN2K, Stacheldracht)
- connessioni via IRC (chat) o Instant Messaging (Trinity)

 13

Caratteristiche dei Tool per DDoS (cont.)

Metodi di Spoofing (da TFN in poi):

- random sull intero indirizzo IP;
- random sugli ultimi 8 bit (puo' evitare filtri di egress)

Piattaforme (da TFN2K): Unix, Linux e Windows

Aggiornamento/Riconfigurazione automatica : ad esempio via ssh (comunicazioni crittografate)

Distribuzione : uso di vulnerabilita' note e automatizzabili; allegati di posta elettronica, cross-site scripting, worm.

Host ospitanti agenti per DDoS: DNS, Web Server, utenze personali "always-on", societa' dotate di connessioni veloci, siti di eCommerce, siti militari, universita', etc.

 14

Effetti attacco DDoS

- Un attacco puo' essere diretto ad una sottorete e per questo non venire rilevato dal gestore della rete principale;
- 100-200 agenti possono saturare completamente la banda trasmissiva di un sito di grosse dimensioni;
- Piu' il numero di agenti e' ampio e distribuito geograficamente, maggiore la difficolta' nel bloccarli;
- Effetti DoS collaterali provocati da RST/ACK inviati dalla vittima in risposta agli IP Spoofed.

 15

Preparazione

- Attivita' di scanning ad amplissimo spettro e per blocchi di classi di indirizzi;
- Molto spesso utilizzo di account gratuiti oppure di host residenti in diverse nazioni o da nazioni aventi legislazione non restrittiva per reati informatici;
- Automatizzazione del processo di intrusione, installazione tool e cancellazione delle tracce;

 16

Realizzazione rete di agenti

- Tempo minimo richiesto per compromettere un sistema ed ottenere i diritti di root/amministratore: 3 secondi (dati ricavati da log resi pubblici);
- Tempo minimo richiesto per l'intero processo di intrusione, installazione e cancellazione tracce: 10 secondi;
- Individuate reti di agenti composte da migliaia o decine di migliaia di computer.

 17

Classificazione per STRATEGIA DI SCANNING

- RANDOM
- LISTA
- TOPOLOGIA
- SOTTORETE LOCALE

 18

• RANDOM: ogni host compromesso esegue un'attivita' di scanning su indirizzi IP scelti in maniera casuale al fine di identificare nuove macchine da compromettere. Il traffico creato dagli host compromessi risulta esteso e cio' rende piu' facile l'analisi e l'individuazione di fenomeni anomali (Code Red worm utilizzava questa tecnica);

• LISTA: ogni host compromesso riceve una lista di indirizzi IP sui quali effettuare un'attivita' di scanning a; fine di individuare nuove macchine da compromettere. Questa tecnica rende la propagazione molto veloce evitando azioni sovrapposte di scanning tra differenti host compromessi.

• TOPOLOGIA: ogni host compromesso usa le informazioni locali all'host stesso per individuare nuovi potenziali target. L'esempio tipico e' dato dai worm che si propagano via E-mail che sfruttano gli address book locali per ricavare indirizzi dei destinatari.

• SOTTORETI LOCALI: una ulteriore strategia che viene usata in combinazione con le precedenti prevede di ricercare i possibili nuovi host da compromettere all'interno delle sottoreti locali all'host compromesso. Questa tecnica ha numerosi vantaggi: a) sfrutta relazioni di fiducia o risorse condivise per le intrusioni; b) mantenendo l'attivita' di scanning locale, risulta molto difficilmente individuabile. Casi di worm come Code Red II e Nimda hanno dimostrato l'estrema efficacia di questa strategia.

Classificazione per GRADO DI AUTOMATISMO

- MANUALE
- SEMIAUTOMATICO
- AUTOMATICO

 19

• MANUALE: Solo i primi DDoS venivano lanciati manualmente, quando l'attaccante si loggava sulle macchine compromesse e da esse eseguiva i comandi per il DDoS. Rischioso perche' (relativamente) facile da tracciare;

• SEMIAUTOMATICO: L'attaccante installa script sulle macchine compromesse (HANDLER) per ottenere il controllo di ulteriori macchine e per lanciare il DDoS (SLAVE). Quindi, l'accesso dell'attaccante e' limitato a pochi HANDLER i quali automatizzano la connessione el'invio di comandi agli SLAVE. Molto piu' difficile da tracciare.

• AUTOMATICO: si evita anche la comunicazione diretta tra attaccante ed handler. Il tipo, durata e vittime dell'attacco viene pre-configurato nel codice degli script sia degli handler che degli slave. L'azione richiesta all'attaccante consiste quindi in un singolo comando che avvia l'attacco. Questo puo' essere fornito senza una comunicazione diretta, ad esempio su un canale IRC, attraverso un sito Web, posta elettronica etc.

Classificazione per TIPO DI PROPAGAZIONE

- SORGENTE CENTRALE
- CATENA A RITROSO (Back-chaining)
- AUTONOMA

 20

• SORGENTE CENTRALE: il codice degli script per gestire l'attacco risiede su di un server centrale (es. Sito Web). Su ogni macchina compromessa viene scaricato tale codice dal server centrale. Meccanismo non molto efficiente perche' se individuato il server centrale e disabilitato, la propagazione si interrompe (li0n worm ha usato questo meccanismo);

• BACK-CHAINING: in questa tipologia di propagazione non esiste un server centrale, ma il codice degli script viene scaricato sulla macchina compromessa dalla macchina usata per comprometterla. Meccanismo piu' efficace poiche' non dipende da un unico server centrale, rendendo piu' difficile arrestare la propagazione (Ramen worm utilizza questa tecnica);

• AUTONOMA: con un meccanismo di propagazione autonoma, il codice necessario per il DDoS viene installato al momento della compromissione. (Code Red e la maggior parte dei worm trasmessi per posta elettronica utilizza questa tipologia).

Classificazione per FREQUENZA DELL'ATTACCO

- COSTANTE
- VARIABILE
  - Crescente
  - Fluttuante

 21

• COSTANTE: e' la tipologia piu' utilizzata. La frequenza dei pacchetti inviati durante un DDoS da parte degli agent coinvolti e' solitamente configurata al valore massimo possibile. L'attacco risulta immediatamente visibile e identificabile e intende interdire la connettivita' alle vittime;

• VARIABILE: con questa tipologia invece l'attaccante intende rendere meno visibile e non immediatamente identificabile l'attacco. Le due tipologie note sono:
- CRESCENTE : la frequenza di invio dei pacchetti cresce gradualmente. L'effetto indotto e' di degradare progressivamente i servizi della vittima senza che tale malfunzionamento possa essere immediatamente attribuito ad un'attivita di tipo DDoS;
- FLUTTUANTE : una frequenza di attacco fluttuante prevede che la frequenza di invio dei pacchetti venga modulata in base al comportamento della vittima al fine di indurre ambiguita' nell'interpretazione dei disservizi. Possono ad esempio essere emessi pacchetti ad altissima frquenza per un periodo molto breve di tempo, inducendo temporanei malfunzionamenti. Oppure, l'attacco puo' essere configurato in modo tale da attivare periodicamente gruppi di agenti cosicche' la vittima subisca un DDoS continuo ma l'individuazione degli agenti sia piu' difficlotosa.

Contromisure

- **Non esiste una contromisura specifica**; occorre agire sul contesto che rende possibile la rete di handler e agenti;
- **Spoofing**: INGRESS ed EGRESS filtering (per gli ISP soprattutto);
- **Broadcast**: disabilitarlo quando non usato, impedirne l'accesso dall'esterno di una rete;
- **Collaborazione**: massima collaborazione al momento della segnalazione da parte di chi ospita agenti DDoS e da parte di ISP;
- **Computer non protetti**: sono milioni i computer vulnerabili e potenziali ospitanti di agenti DDOS; mancanza di skill da parte di molti sistemisti;

 22

Report di un caso reale

http://www.grc.com/

The Strange Tale of the

DENIAL OF SERVICE

Attacks Against GRC.COM

by Steve Gibson, Gibson Research Corporation

Nothing more than the whim of a 13-year old hacker is required to knock any user, site, or server right off the Internet.

 23

Il report reso pubblico da Steve Gibson viene considerato come uno dei migliori documenti riguardanti un incidente di tipo DDoS.

Prestare particolare attenzione ai dettagli non-tecnici, che spesso non vengono rilevati ma che invece, come testimoniato da questo caso, possono rappresentare un ostacolo significativo.

Analisi

L'uso di un server IRC fornisce vantaggi in termini di efficenza e privatezza.

- Gli agenti devono solo conoscere il server e il canale al quale connettersi;
- L'attaccante o gli handler non devono conoscere le macchine sulle quali risiedono gli agenti.

I meccanismi di propagazione degli agenti possono quindi essere i piu' vari e non sotto il controllo dell'attaccante (es. attraverso allegati alle email).

- estrema vulnerabilita' della "Internet economy";
- mancanza di contromisure efficaci (soprattutto per motivi non tecnologici).

 24

Il caso presentato rappresenta uno dei report piu' noti riguardanti i DDoS.

Mette in evidenza in maniera chiara la facilita' con la quale una qualsisi organizzazione puo' diventare vittima di tale attacco e la difficolta' di risposta e protezione.

Da un diverso punto di vista pero', quello rappresentato e' un caso eccezionale: solo circostanze eccezionali hanno fatto si' che la societa' divenisse l'obiettivo di tale attacco e di tale insistenza (la ragione fu un commento esposto pubblicamente da S. Gibson a proposito dei cosiddetti "script kiddies").

Nelle circostanze quotidiane, difficilmente una organizzazione che non sia particolarmente esposta (es. ISP, marchi famosi, siti militari/politici, etc.) diventa il target specifico di un attacco del genere. Quindi, come linea generale, il caso di DDoS, per una organizzazione media non rappresenta uno dei pericoli di maggiore priorita'. Molto piu' concreto e' il rischio di diventare agenti o handler.

Un caso a parte potrebbe essere costituito da Pubbliche Amministrazioni o Enti qualora realizzassero servizi di pubblica utilita' attraverso Internet. Questi, per la rilevanza e la eco che produrrebbe un DoS, potrebbero, con maggiore probabilita', essere target di DDoS.

Alcuni esempi (del tutto ipotetici, non vi sono esperienze pregresse): controllo della centrale semaforica di un Comune, controllo del traffico aereo di un aeroporto, etc.

Virus, Trojan Horse e Internet Worm

 1

VIRUS, TROJAN HORSE e WORM: spesso definizioni confuse e non distinte. Terminologia spesso non univoca.

VIRUS : Un programma avente capacita' di

- **REPLICAZIONE**
- **CAMUFFAMENTO**
- **INDURRE MALFUNZIONAMENTI**

Colpiti soprattutto applicazioni in ambiente Windows, sia per la diffusione che ne facilita la replicazione che per le funzionalita' (basate su VB script) inserite in molte applicazioni.

 2

Vedere i molti documenti dal SANS Institute (www.sans.org) in particolare la sezione Reading Room (http://rr.sans.org/)

CONTROMISURE

Sistemi anti-virus: estremamente efficaci, aggiornamento tempestivo.

Criticita'

- Aggiornamento Definizioni del Virus: ON-LINE (contesti aziendali), MANUALE (utenze private e domestiche);
- Presenza di Computer Non Controllati: in contesti aziendali, Ospiti, Consulenti, Utenti in Telelavoro, Dipendenti con Laptop.

 3

Quali contromisure adottare? SISTEMI ANTI-VIRUS.

Sono estremamente efficaci e aggiornati in maniera solitamente molto tempestiva.

L'aspetto critico riguarda tutti i casi in cui l'aggiornamento avviene manualmente (tipicamente per i pc personali e le utenze domestiche). Considerando la frequenza di apparizione di nuovi virus (e worm, descritti in seguito), l'aggiornamento dell'antivirus dovrebbe essere estremamente frequente (settimanalmente).

Di solito il default degli anti-virus per l'aggiornamento e' di 30 o anche 60 giorni, tempo assolutamente troppo ampio nell'attuale situazione.

Spesso poi neppure dopo questo lasso di tempo gli utenti provvedono all'aggiornamento.

Quindi, la causa dell'enorme diffusione dei virus e' principalmente da imputare alla PESSIMA GESTIONE DEGLI AGGIORNAMENTI DEI SISTEMI ANTI-VIRUS DA PARTE DEGLI UTENTI.

In moltissimi contesti aziendali, la gestione degli aggiornamenti degli anti-virus dei computer dei dipendenti avviene in modo centralizzato e on-line. I tempi si possono ridurre drasticamente a poche ore dalla pubblicazione di un nuovo aggiornamento da parte del produttore.

Rimane il problema di quei computer che non possono essere gestiti centralmente(es. Portatili, pc dei dipendenti in telelavoro, pc di proprieta' di ospiti, consulenti etc.). Questi rappresentano i rischi maggiori quali veicoli di propagazione di virus anche all'interno di un ambiente protetto e ben gestito.

TROJAN HORSE : Un programma apparentemente legittimo che nasconde funzionalita' illecite.

A differenza dei virus, non possiede capacita' di autoreplicarsi o di infettare altri file e programmi.

In molti casi, trojan sono stati inseriti in pacchetti sofware di ampia diffusione a insaputa dei produttori.

Molti esempi sia nel mondo Unix/Linux che Windows.

Esempi: servizi tradizionali Unix (telnet, ftp), molto recentemente implementazione open source (OpenSSH) di ssh e sendmail per Linux.

Utility, giochi, programmi freeware per Windows.

Nei due casi recentissimi di OpenSSH e Sendmail, i siti ufficiali sui quali vengono messe a disposizione le nuove versioni sono stati compromessi. Gli intrusori hanno cosi' potuto sostituire le versioni originali con versioni contenenti trojan horse.

Spesso contengono BACKDOOR: attivano servizi di rete su porte non convenzionali che permettono l'accesso remoto e il facile controllo di una macchina.

Veicolano anche gli script per DDoS (handler e Agenti)

Esempi: abilitazione di server SSH, suite complete per il controllo remoto (Back Orifice, SubSeven).

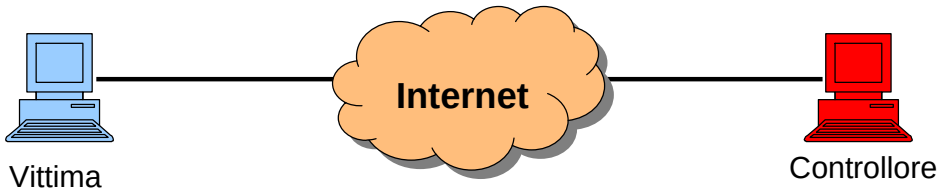
 5

Questo e' l'aspetto certamente piu' grave e preoccupante dei trojan horse.

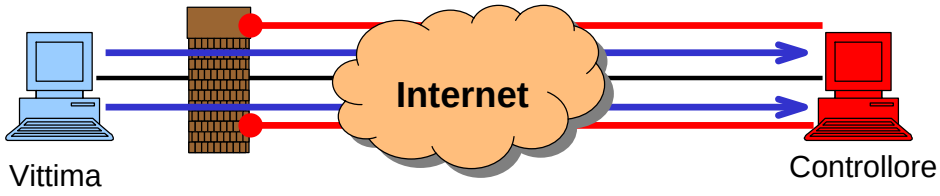
Sono strumenti per la gestione coordinata e automatizzata di larghissime reti di host compromessi, utilizzati sia come passi intermedi per intrusione di sistemi (al fine di rendere piu' ardua la tracciabilita' dei responsabili) che come agenti per DDoS.

EVOLUZIONE DEI TROJAN HORSE

Primo Modello:
indirizzi IP pubblici, nessuna protezione perimetrale



Secondo Modello:
indirizzi IP pubblici, protezione rispetto connessioni entranti



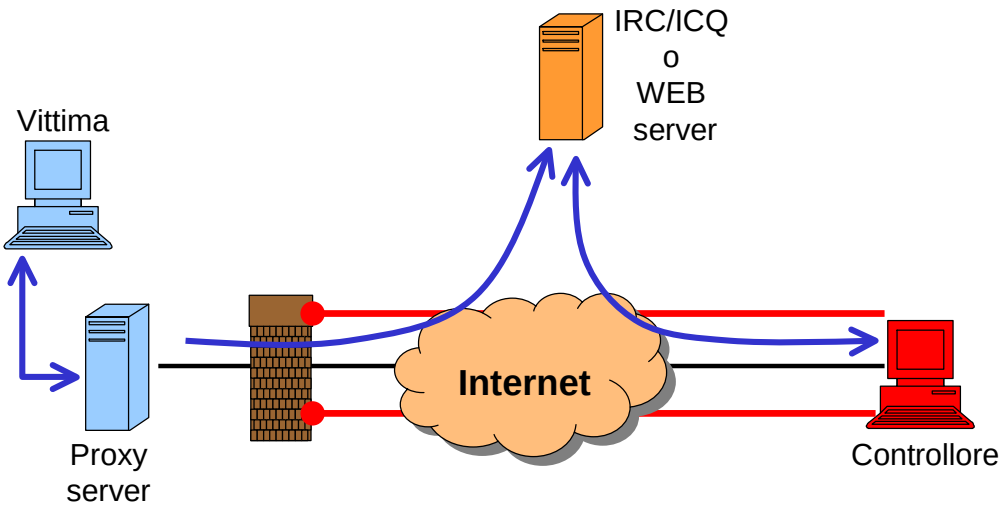
 6

PRIMO MODELLO: il controllore puo' connettersi direttamente alle macchine sulle quali e' presente il trojan horse poiche' l'indirizzo IP e' pubblico e le connessioni in ingresso non vengono filtrate. Oggi questo e' il tipico caso delle utenze domestiche, di molte universita' (cosi' come ancora un numero non irrilevante di realta' aziendali).

SECONDO MODELLO: il controllore non puo' connettersi direttamente alla porta applicativa abilitata dal trojan horse perche' e' in presenza di una protezione perimetrale, pertanto solo alcuni servizi, configurati su porte standard, e alcuni host possono ricevere connessioni dall'esterno. Tuttavia, sessioni iniziate da utenti interni vengono autorizzate senza limitazione e pertanto sono i trojan horse a connettersi ad indirizzi IP o siti Web pre-programmati. In questo modo il controllore riese a mantenere totale gestione degli host attraverso il trojan.

EVOLUZIONE DEI TROJAN HORSE (cont.)

Terzo Modello:
indirizzi IP riservati, filtraggio delle connessioni in entrambi i versi



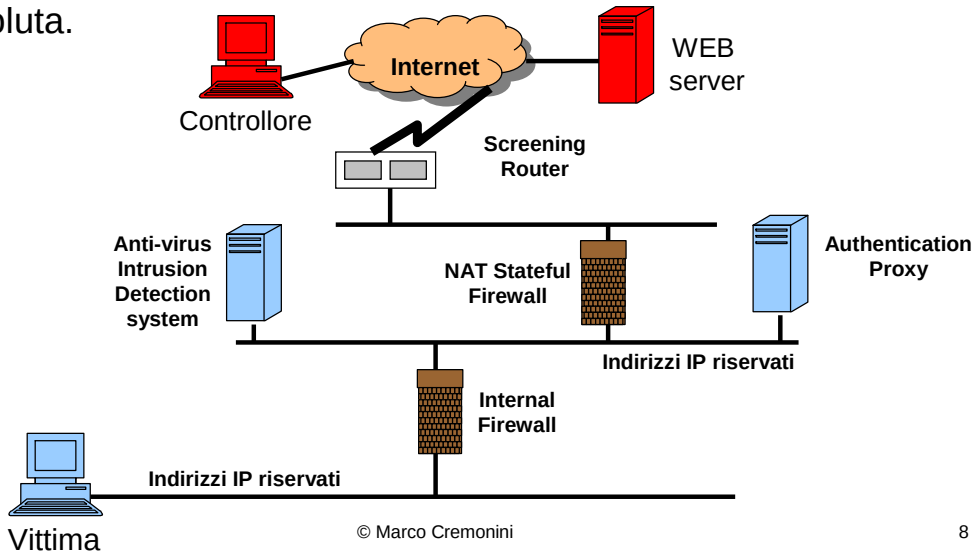
 7

TERZO MODELLO: il controllore non puo' connettersi alla vittima perche' le connessioni in ingresso sono filtrate. Il trojan residente sulla vittima non puo' connettersi direttamente al controllore perche' su rete con IP riservati e connessioni esterne proxate e limitate alle sole autorizzate. Anche in questa configurazione, l'evoluzione dei trojan ha fatto si' che venissero sfruttati quei servizi molto diffusi e spesso abilitati, connessione a chat (IRC/ICQ) e soprattutto sessioni http (WEB) ordinarie.

I comandi emessi dal controllore vengono reperiti dal trojan accedendo a canali di chat riservati oppure a siti web pre-programmati.

EVOLUZIONE DEI TROJAN HORSE (cont.)

Quarto Modello:
indirizzi IP riservati, filtraggio delle connessioni in entrambi i versi, meccanismi e infrastruttura di sicurezza estremamente evoluta.



QUARTO MODELLO: ipotizziamo un ambiente il cui livello di sicurezza e protezione sia elevato, composto, ad esempio, di screened router, stateful firewall, NAT, meccanismi di antivirus, etc.

Ipotizziamo inoltre che agli utenti del sistema, compresa quindi la workstation della vittima sulla quale e' presente il trojan, siano abilitati a navigare sul Web (come comunemente accade).

Infine, ipotizziamo che il controllore gestisca un sito Web, requisito facilmente soddisfabile.

Vediamo come puo' essere anche in questo caso controllato il trojan:

Vittima → Web Server :
http://www.controllore.org/msg.asp?text="Salve%20attendo%20comandi"

Web Server → Vittima : Pagina HTML contenente nell'header "Inizia DDoS verso www.qualcuno.it"

Quindi, il semplice accesso a pagine Web puo' essere utilizzato per un colloquio tra handler e agent in un DDoS o, in generale, tra controllore e trojan, nonostante le eccellenti misure di sicurezza.

WORM : e' un programma che possiede generalmente le seguenti caratteristiche:

- capacita' di scanning e sniffing alla ricerca di nuovi target;
- tecniche di intrusione automatizzate;
- una interfaccia per ricevere comandi;
- proprie capacita' di comunicazione e propagazione.

Un worm e' quindi un vero e proprio programma che automatizza l'intero ciclo di una intrusione.

 9

Tra il 2001 e il 2002 c'e' stato una escalation grave nella diffusione di worm (Code Red, Nimda, Ramen, li0n, etc.) con percentuali di propagazione elevatissime.

I worm sono ad oggi probabilmente il veicolo primario di intrusione e di installazione di script per DDoS, cosi' come di attivazione di backdoor in reti e sistemi altrimenti protetti.

Caratteristiche : **Capacita' di scanning e sniffing**

Automatizzata l'attivita' di scanning e sniffing sul traffico di rete.

SCANNING: Solitamente mirata a ricercare host con servizi corrispondenti alle vulnerabilita' per le quali possiedono gli script che ne automatizzano l'attacco. Esempi tipici, telnet, ftp, ssh, netbios, IIS web server, Microsoft SQL server.

SNIFFING: Possono sia sniffare traffico dalla rete alla ricerca di username/password che loggare gli input della tastiera e successivamente inviare quanto registarato al controllore.

 10

Caratteristiche : **Tecniche di intrusione automatizzate**

Automatizzata l'intrusione sfruttando specifiche vulnerabilita' di servizi di rete, buffer overflow, cgi malprogettati e le molte atre vulnerabilita' dei Web server.

Possiedono gli script che permettono, una volta identificati nuovi target, di compromettere un sistema vulnerabile e propagarsi.

 11

Caratteristiche : **Tecniche di intrusione automatizzate (cont.)**

NOTARE: oltre a script per utilizzare vulnerabilita' note nei sistemi, la maggior parte utilizza **LA TECNICA PIU' DIFFUSA, SEMPLICE ED EFFICACE PER COMPROMETTERE UN SISTEMA**:

L'INVIO DI MESSAGGI DI POSTA ELETTRONICA CON IL WORM STESSO IN UN ALLEGATO ESEGUIBILE

(quest'ultimo requisito non e' necessario per Microsoft Outlook a causa del supporto per VB script eseguiti automaticamente alla ricezione di un nuovo messaggio)

 12

Caratteristiche : **Interfaccia per ricevere comandi**

Per questa funzionalita' i worm presentano caratteristiche mutuate dai Trojan Horse, disponendo quindi di diverse tecniche per comunicare e offrire possibilita' di controllo ad un attaccante.

 13

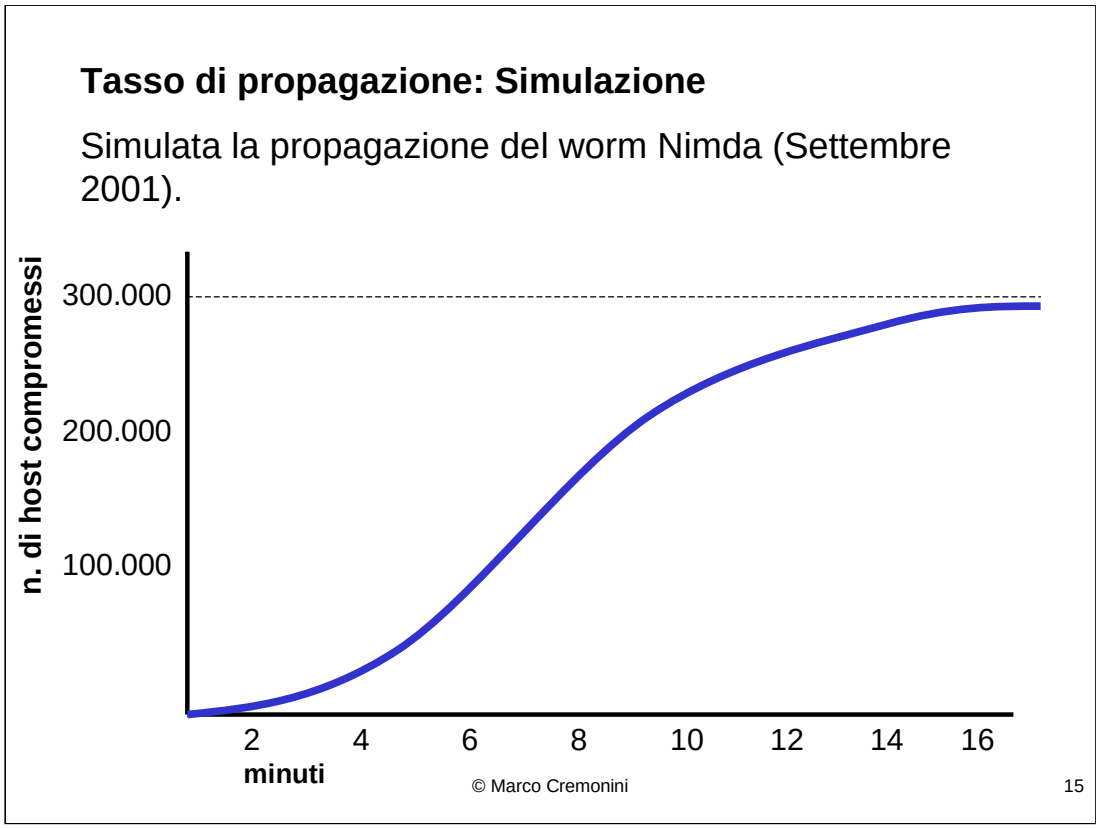
Riferirsi alla discussione precedentemente esposta circa l'evoluzione dei trojan.

Caratteristiche : **Capacita' di comunicazione e propagazione**

- La caratteristica piu' evidente rispetto i virus e i trojan e' l'automatizzazione della propagazione di un worm.
- Completa automatizzazione del ciclo di vita di un'intrusione.
- Reso possibile dal numero di vulnerabilita' esistenti, dalla loro diffusione in programmi di vastissima diffusione e dalla semplicita' nell'ottenere il controllo di un host attaccato.

 14

Mentre un virus o un trojan sono legati ad un particolare programma e sfruttano la copia o il download di tale programma per replicarsi, un worm non e' legato a nessun programma ma procede autonomamente nella propagazione, cercando nuovi target, compromettendoli, trasferendo il software necessario, autoinstallandosi e camuffandosi. Il ciclo poi si ripete per ogni istanza del worm e questo ha provocato il tasso di diffusione elevatissimo in tempi estremamente ridotti dei casi accaduti.



Questo il risultato di una simulazione realizzata dopo la propagazione del worm Nimda (settembre 2001) che mostra chiaramente le capacita' di propagazione degli Internet Worm.

I dati sono ricavati da:

"How to 0wn the Internet in Your Spare Time"

S. Staniford (Silicon Defense), V. Paxon (ICSI Centre for Internet Research) e N. Weaver (UC Berkeley)

2002

Rootkits

 16

Rootkit: tool o collezione di tool per:

- fornire all'intrusore gli strumenti per mantenere il pieno controllo dell'host compromesso senza essere individuato
- nascondere le tracce dell'intrusione

Quindi un rootkit NON e' uno strumento per acquisire i diritti di root. I diritti di root devono essere gia' stati acquisiti in precedenza (vulnerabilita').

 17

Lettura consigliata: i molti documenti dal SANS Institute (www.sans.org) in particolare la sezione Reading Room (http://rr.sans.org/)

Componenti tipici di un rootkit:

- **Backdoor**: una backdoor e' una modalita' non autorizzata che permette l'accesso ad un sistema o computer. Una backdoor serve all'intrusore per accedere facilmente al sistema in momenti successivi.
- **Sniffer**: successivamente all'intrusione, uno sniffer sulla macchina compromessa puo' raccogliere il traffico interno ad una rete (es. Password trasmesse in chiaro);
- **Programmi per modificare i Log**: questi tool modificano i log di sistema eliminando I record relativi all'intrusione;
- **Altri tool**: Agenti DDoS, Client IRC, etc.

 18

Tipi di Rootkit:

per Applicativi (application rootkit): e' la tipologia tradizionale (esistono da fine anni '80 almeno), vengono sostituite applicazioni standard con versioni modificate.

per il Kernel (kernel rootkit): viene modificato il kernel, moduli o system call. Molto difficili da identificare perche' bypassano I controlli a livello applicativo. Piu' recenti, sviluppati soprattutto per Linux e utilizzando la capacita' di linkare dinamicamente moduli (Loadable Kernel Modules - LKM).

Application Rootkit

Programmi modificati per nascondere la presenza dell'intrusore:

`ls`, **`find`**: non mostrano I file e le directory dell'intrusore;

`ps`: non mostrano i processi dei programmi del rootkit;

`netstat`: non mostra le porte aggiuntive aperte dal rootkit;

`killall`: non interrompe i processi dell'intrusore;

`ifconfig`: non mostra il flag PROMISC attivato dallo sniffer (modo promiscuo del driver della scheda di rete);

`crontab`: non mostra i processi schedulati dal rootkit;

`tcpd`, **`syslogd`**: I servizi di logging non loggano l'attivita' dell'intrusore;

Application Rootkit

Servizi di rete con backdoor:

`inetd`: (Internet Daemon) attiva tutti i servizi di rete abilitati (es. telnet, ftp, …). La versione modificata permette all'intrusore di aprire qualunque porta applicativa;

`rshd`: (Remote Shell), la versione modificata restituisce accesso di root all'intrusore utilizzando una password particolare;

`sshd`: una versione modificata di SSH fornisce accesso di root. Utile per avere comunicazioni criptate e non analizzabili da sniffer.

 21

Application Rootkit

Altri programmi:

`fix`: modifica data/tempo e dimensione dei programmi modificati (es. **`ls`**, **`ps`**, …) con I valori di quelli originali;

`wted,z2, zap3`: cancellano o editano le entry relative all'intrusore dai log di sistema in **`/var/log`**, **`/var/adm`**, **`usr/adm`**, **`var/run`**, compresi quelli binari **`wtmp`**/**`utmp`**/**`lastlog`**;

 22

Kernel Rootkit

Kernel: gestione CPU, gestione memoria, driver dei device, gestione I/O;

Moduli linkati dinamicamente: molto usati nei kernel attuali, problemi di dimensione dei kernel attuali (es. Linux, Solaris, FreeBSD), utili per linkare dinamicamente driver di devices, ad esempio.

Modifica di system calls: metodo potente perche' comuni a molte applicazioni/programmi. Manipolando system call o moduli del kernel si ottiene il piu' completo controllo di un computer.

 23

Contromisure

- Gestione rigorosa della configurazione e delle patch.
- Proteggere i sistemi dalle vulnerabilita' piu' utilizzate (es. Vedi lista SANS).
- Utilizzo di tool ad-hoc per rilevare la presenza di rootkit (chkrootkit, rkscan, Carbonite, rkdet). Non sempre efficaci pero'.
- Utilizzo di moduli linkati dinamicamente solo se necessario. Host critici (es. Firewall), se possibile, dovrebbero avere il kernel compilato staticamente per prevenire l'uso di kernel rootkit.
- Protezione perimetrale rigorosa per prevenire intrusioni.

