

Sistemi di elaborazione dell'informazione - Sicurezza su reti  
Anno 2003/04

# IL WORM BLASTER

## Il superbug di windows

Luigi Alfano - Ulisse Chirico - Nadia MoscarIELlo - Daniele Palumbo - Laura Santoro

*Il worm blaster: il superbug di windows*

### introduzione

#### evoluzione dei worms

- La realizzazione di codice maligno (virus e worm) è cresciuta di pari passo con lo sviluppo di software
- Vengono realizzati, la maggior parte delle volte, al fine di rallentare, violare, distruggere sistemi informatici o reperire informazioni riservate
- Altre volte sono motivo di sfida
- Il fine principale resta ovviamente quello economico

*Il worm blaster: il superbug di windows*

### introduzione

#### il worm blaster

- Creato con l'intento di attaccare il più grande colosso informatico quale la Microsoft al fine di screditarlo
- Il codice del worm sfrutta una grossa vulnerabilità presente in alcuni servizi dei sistemi operativi WindowsXP e Windows2000
- Come ogni worm si diffonde nei computer collegati ad internet e, auto-riproducendosi, continua la sua opera di contagio sulla rete
- La sua semplicità di diffusione è confermata dal fatto che nei soli primi due giorni di attività ha infettato più 120.000 macchine

*Il worm blaster: il superbug di windows*

### autore

#### Jeffrey Lee Parson

- Giovane hacker conosciuto sul web come Teekid
- È stato identificato dagli agenti della sezione crimini informatici dell'FBI attraverso il suo sito web, [t33kid.com](http://t33kid.com)
- Il birbone rischia adesso una pena di circa 10 anni e una multa di circa 260.000 dollari



*Il worm blaster: il superbug di windows*


### scoperta e diffusione

#### ...in due giorni 120.000 vittime

- Il worm è stato rilevato la prima volta l'11 agosto del 2003
- Ha interessato i cronisti di tutto il mondo sia per la ridottissima dimensione (circa 6000 caratteri), sia per la sua rapidità di diffusione senza ausilio di posta elettronica

Il 16 luglio del 2003 la Microsoft rilascia il bollettino di sicurezza MS03-026, nel quale avverte i suoi clienti dell'enorme rischio a cui i loro computer sono esposti

Come si vince dal grafico la crescita del numero delle vittime è avvenuta in maniera esponenziale




*Il worm blaster: il superbug di windows*

### scopi e obiettivi

#### "I love you, Billy"

- Il worm è stato concepito, studiato e realizzato, per attaccare in maniera distribuita i server della microsoft
- La tipologia di questo attacco, definita col termine tecnico **DoS** (denial of service), consiste nel provocare un consistente calo delle prestazioni o addirittura un blocco di un servizio o di un intero sistema
- Una ulteriore prova che conferma l'identità dell'obiettivo da danneggiare è una frase che è visibile solo nel codice





## DoS

### Denial Of Service

- Interruzioni di servizio provocate da sequenze di dati studiate ad arte, ricevendo le quali un sistema networked cessa di funzionare normalmente e, di solito, subisce un degrado di prestazioni più o meno grave, fino all'arresto totale della macchina
- Una variante di questo tipo di tecnica è detta DDoS (Distributed Denial Of Service), dove ad attaccare questa volta non è una singola macchina bensì tante altre che nello stesso istante colpiscono il loro obiettivo



## disagi per l'utente

### ...60 secondi e poi la morte!

- Il maggiore disagio provocato all'utente è il riavvio della macchina a causa di un crash del servizio RPC
- L'icona del disagio è la comparsa della finestra di shutdown che avvisa l'utente dell'inevitabile reboot del sistema
- Il problema sta nel fatto che il servizio accetta anche i messaggi contenenti il codice maligno, che provocano nel sistema una serie di eccezioni che causeranno il riavvio del computer



## danni provocati

### USA: danni per 3 milioni di dollari

- Per ore il worm ha causato il blocco completo dei servizi offerti da grandi organizzazioni di tutto il mondo, dalla *Federal Bank* di Atlanta agli uffici governativi di Hong Kong.
- La motorizzazione del Maryland ha dovuto bloccare le pratiche per più di ventiquattro ore.
- Il potenziale "distruttivo" del worm è di gran lunga superiore. A limitare i danni è stato il periodo della sua diffusione: il mese di agosto, fase dell'anno in cui l'attività di produzione delle aziende rallenta.



## tipologia di attacco

### ...il buffer overflow

- Il worm Blaster utilizza una delle tipologie di attacco più conosciute e usata nel campo della programmazione: IL BUFFER OVERFLOW
- Questa tecnica, concettualmente semplice, ha bisogno di una conoscenza approfondita di alcuni elementi del mondo della programmazione: locazioni di memoria, stack, istruzione successiva, puntatori, etc.
- Disattenzioni e eventuali mancanze di controlli da parte dei programmatori sono la causa della vulnerabilità di molti software.



## Buffer overflow

### introduzione

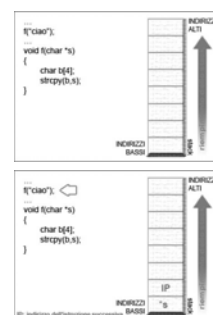
- Ogni programma in esecuzione occupa in memoria delle locazioni (buffer) dove verranno inseriti:
  - Testo: contenente il codice del programma (read only)
  - Stack: contenente variabili locali usate nelle funzioni, l'indirizzo dell'operazione successiva
  - Dati: contenente i dati inizializzati e non
- Tutte queste informazioni necessaria all'esecuzione del programma sono inserite in modalità LIFO (come in uno stack) in maniera sequenziale



## Buffer overflow

### Procedura lecita (1)

- Per semplificare la spiegazione immaginiamo che nello stack siano memorizzate solo i dati e l'indirizzo dell'istruzione successiva (IP)
- Ipotizziamo che una funzione f esegua una procedura lecita all'interno di un programma più complesso
- In questo esempio, la funzione scriverà la parola "Ciao" nello stack utilizzando locazioni di memoria successive a quella destinata all'IP

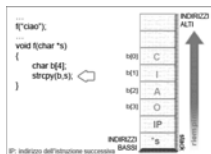
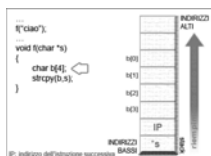




## Buffer overflow

### Procedura lecita (2)

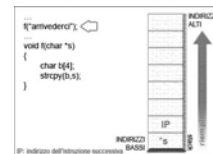
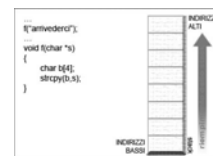
- La procedura lecita alloca la memoria necessaria a tale operazione b[0]...b[3]
- Scrive la parola "Ciao" nelle posizioni di memoria precedentemente allocate
- Esegue l'istruzione successiva spostandosi nella locazione di memoria IP



## Buffer overflow

### Procedura illecita (1)

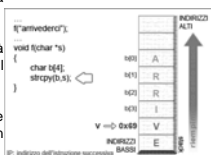
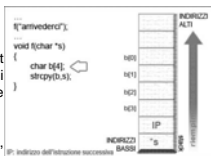
- La procedura precedente è lecita per la coerenza tra input ("CIAO") e locazioni di memoria assegnate (4)
- La funzione però non prevede controlli
- Immaginiamo quindi di scrivere qualcosa di più dimensione maggiore ("Arrivederci"), allocando comunque le quattro posizioni, come da procedura



## Buffer overflow

### Procedura illecita (2)

- L'assenza di controlli sul formato dell'input consente che vengano accettate parole di qualsiasi lunghezza, nonostante le allocazioni di memoria siano comunque 4
- Il programma procederà come deve, fuoriuscendo dall'area di memoria consentita (OVERFLOW)
- L'indirizzo dell'istruzione successiva verrà inevitabilmente modificato, pregiudicando il corretto funzionamento del programma
- In questo caso "V" potrebbe essere l'indirizzo della prima istruzione di un codice maligno



## Come agisce il worm

### L'attacco

- "MSBLASTER.EXE" (l'eseguibile del worm) è di 6176 byte, una volta decompresso, rivela circa 11 Kb di codice capace di sfruttare il bug del servizio DCOM/RPC della microsoft
- Partendo da una postazione già contaminata, il worm attraverso la porta TCP 135 (porta utilizzata dal servizio), invia dati con i quali effettuerà l'attacco
- Nell'80% dei casi l'attacco sarà relativo al sistema operativo Windows XP, nel restante 20% al Windows 2000.



## L'attacco

### le sette fasi

- Immaginando che in rete siano presenti due host: A (infetto) e B (da contagiare), la fase di attacco può essere riassunta in sette fasi specifiche:
  1. L'attesa
  2. Generazione degli indirizzi IP
  3. Attacco al servizio RPC
  4. Controllo del contagio
  5. Lancio della shell CMD.EXE
  6. Download del Worm
  7. Upload registry Keys



## Fase 1: l'attesa

### Impostazioni iniziali

- A (host infetto da MSBlaster) entra in un loop infinito
- Aspetta che la funzione InternetGetConnectedState() ritorna il valore desiderato
- Quando ciò accade il worm sarà sicuro che il computer A è connesso ad Internet e pronto a cercare nuove vittime per il contagio.



## Fase 2: generazione indirizzi IP alla ricerca di B (1)

- La procedura di MSBlaster in **A**, genera gli indirizzi IP di altri sistemi per proseguire nel contagio. Per fare questo usa la seguente procedura
  - Il worm rileva l'IP di **A** A.B.C.D. e ne inserisce i valori nelle variabili
  - Casualmente sceglie un valore tra 1 e 20, "decidendo" se usare l'indirizzo IP dell'host come base per generare una lista di indirizzi da scandire (da 1 a 12), o se creare un indirizzo completamente casuale (da 13 a 20)



## Fase 2: generazione indirizzi IP alla ricerca di B (2)

- Nel caso della scelta dell'IP dell'host **A**, se il suo valore **C** dell'indirizzo IP è maggiore di 20 allora decreterà tale valore di 20
- Il valore **D** dell'IP sarà sempre impostato a 0
- Se il worm sceglie di usare un indirizzo casuale da cui iniziare la scansione allora genererà **A B e C** in questo modo:
  - **A** da 1 a 254
  - **B** da 0 a 253
  - **C** da 0 a 253
  - **D** sempre 0
- Scandendo 20 host per volta, il worm, connettendosi alla porta TCP 135, scoprirà nuove macchine vulnerabili tra cui **B**



## Fase 3: attacco al RPC ...il buco!

- Il worm Blaster in **A**, attraverso la porta TCP 135, manderà speciali argomenti in pacchetti RPC al servizio RPC/DOM della nuova vittima **B**,
- Il servizio RCP/DOM di **B**, non effettua controlli specifici sulla natura del pacchetto ricevuto, né sulla sua consistenza.
- Il pacchetto RPC in realtà nasconderà un attacco di BUFFER OVERFLOW che permetterà di prendere possesso dell'host da infettare.



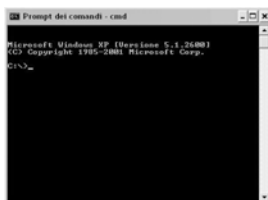
## Fase 4: controllo del contagio L'infezione

- **A**, attraverso la porta TCP 135, controlla se **B** è già infetto
- Effettua una chiamata a funzione che controlla se è già presente l'eseguibile del worm (**GetLastError()**).
- Se riceve il valore del codice di errore di 0xb7 ("Non può creare un file quando questo file già esiste"), MSBlaster termina (**ExitProcess()**)
- Se ciò non avviene allora **A** attiva le socket necessarie alla comunicazione (**WSAStartup()**) e per avviare una comunicazione TFTP necessario per il contagio di **B** (**GetModuleFileName()**)



## Fase 5: la shell CMD.EXE il cavallo di Troia

- MSBlaster di **A**, ormai in controllo di **B**, lancia sulla macchina da infettare, la shell remota CMD.EXE
- La shell funge da "CAVALLO DI TROIA", in quanto permette di eseguire comandi dall'interno della macchina **B**.
- I due host comunicheranno attraverso la porta TCP 4444



## Fase 6: download del worm l'approdo del worm

- Tramite la shell lanciata al passo precedente, **B** invia comandi in remoto per riconnettersi all'host infettante **A**
- **A** rimane in ascolto sulla porta UDP 69, aspettando una richiesta di copia del worm
- **B** richiede tale file ed esegue il download dell'eseguibile msblast.exe, usando il protocollo TFTP
- Il file viene scaricato nella cartella %systemroot%/system32 (cartella di sistema del sistema operativo contagiato) e viene successivamente lanciato



## Fase 7: upload registry keys

### le ultime direttive

- Tramite la stessa shell lanciata al passo 5, **A** apporterà delle modifiche alle Registry Keys del pc infettato.
- Inserirà il valore "windows auto update" = "msblas.exe" nella directory `HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Run`
- Questo farà in modo che il worm venga eseguito ad ogni riavvio del sistema ormai infettato
- A questo punto, l'host infettato eseguirà il ciclo ripartendo dalla FASE 1, continuando e velocizzando la diffusione del worm



## Lo scopo del worm

### Il vero obiettivo del BLASTER

- La vera offensiva del Blaster, quella che cercherà di far danni, avviene parallelamente al contagio
- Sotto determinate condizioni, il worm tenterà di causare un DENIAL-OF-SERVICE contro windowsupdate.com per bloccare i server della microsoft
- L'azione è inoltre eseguita, per impedire ai sistemi, di ottenere la patch (MS03-026) per ovviare alla vulnerabilità dell'interfaccia RPC DCOM
- Il sovraccarico avviene sulla porta 80 del sito attraverso pacchetti SYN e pacchetti HTTP, gli ultimi di quali sono lunghi 40 bytes e sono trasmessi ad intervalli di 50 secondi.



## Condizioni di attacco

### ...se e solo se...

- L'attacco ai server Microsoft avverrà solo se verranno rispettate le seguenti condizioni:
  - Ogni giorno, se il mese corrente è compreso tra Settembre e Dicembre
  - A partire dal giorno 16 se il mese corrente fa parte dei rimanenti
- Inoltre l'attacco DoS opera sotto queste altre 3 condizioni:
  - il worm è eseguito su un sistema WinXP infettato oppure riavviato durante la routine nociva.
  - il worm è eseguito su un sistema W2000 infettato durante la routine nociva e che non è stato riavviato dopo l'infezione
  - il worm è eseguito su un sistema W2000 riavviato dopo l'infezione, durante la routine nociva, e dove l'utente è attualmente registrato



## Conseguenze dell'attacco

### Quali disagi?

- Il numero così alto di sistemi infetti, fa in modo che MSBlaster inondi (e continui ad inondare) i sistemi con traffico sulle porte TCP 135, TCP 4444 ed UDP 69
- La conseguenza principale è il rallentamento del sistema e della rete
- Tale problema è in aggiunta al crash dovuto ai pacchetti RPC costruiti con dati illegali che i sistemi vulnerabili accettano ma non riescono a trattare, causando quei Crash di Sistema e quei continui riavvii di cui prima si è parlato



## Individuazione del worm

### i sintomi sul nostro pc

- Diminuzione sostanziale delle prestazioni della macchina
- Continui riavvii dovuti all'interfaccia RPC della microsoft che, una volta attaccato, riavvia automaticamente l'intero sistema
- Utenti esperti possono accorgersi del contagio analizzando il traffico di pacchetti che coinvolge la propria macchina sulle porte TCP 135 e 4444 e la porta UDP 69 utilizzando il comando NETSTAT
- L'utente inesperto attribuisce tali sintomi ad un uso maldestro di qualche applicazione sulla propria macchina
- Consigliato l'uso di un anti-virus che nella maggior parte dei casi rileva ed elimina automaticamente il worm



## Rimozione del worm (1)

### Come sbarazzarsi del BLASTER

- La rimozione del worm può avvenire in maniera automatica attraverso dei tool scaricabili gratuitamente dalla rete o in maniera manuale
- Per rimuovere il worm manualmente bisogna:
  - Avviare il TASK MANAGER di windows
  - Eliminare il processo MSBLAST.EXE e tutti i processi concorrenti in esecuzione rendendone possibile l'eliminazione
  - Cancellare il file MSBLAST.EXE presente nella cartella C:\Windows\System32
  - Eseguire il REGEDIT
  - Eliminare dalla chiave di registro `HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Run` il valore "windows auto update = msblast.exe"



## Rimozione del worm (2)

### Come sbarazzarsi del BLASTER

- Eliminare dalla cartella STARTUP nel menu start un file chiamato TFTP o simile (responsabile del messaggio di errore che appare all'avvio della macchina)
- Riavviare il computer in modalità normale per avere la possibilità di connettersi ad internet
- Eseguire il windows update dal menu Start, questo installerà la patch che elimina il bug che permette l'attacco e l'intrusione del worm
- Il worm potrà essere ancora presente nel sistema o nel backup del CESTINO o di SYSTEM BIN. Assicurarsi quindi di aver svuotato il CESTINO e di aver disabilitato e poi riabilitato il SYSTEM RESTORE
- Così facendo il sistema può dirsi definitivamente libero dalla minaccia del worm



## Le varianti del worm (1)

### Lo sviluppo del worm

- Queste sono le varianti del blaster secondo una catalogazione della F-SECURE:
  - **LOVESAN.A** è il worm creato da A. Parson che come già detto ha aperto la strada a questa nuova famiglia di worm
  - **LOVESAN.B** rilevata il 13 agosto del 2003, utilizza un dropper che scarica da un sito Web due file, li copia nella directory Windows System ed aggiunge il riferimento nel registro di sistema di Windows nella chiave:  
**HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\Run\**  
Il primo file si chiama **Root32.exe** ed è una *backdoor* mentre il secondo, denominato **teekids.exe** è il worm stesso
  - **LOVESAN.C**: rilevata anch'essa il 13 agosto 2003, è stato cambiato solamente il nome del file in "penis32.exe"



## Le varianti del worm (2)

### Lo sviluppo del worm

- **LOVESAN.F**: variante rilevata il 1 Settembre 2003, di seguito sono elencate le piccole differenze rispetto alla versione originale:
  - Il nome del file è cambiato in "**enbiei.exe**"
  - Il **DDos** target è stato cambiato in "tulasi.ro". Tale indirizzo non esiste per cui l'attacco risulta inefficace
  - aggiunge i seguenti nuovi valori nel registro di sistema:  
**"HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\Run\www.hidro.4t.com"**
- Contiene il seguente messaggio nascosto:  
- Non andare alla facoltà di idrotecnica. Perdi tempo! Birsan, La previdenza sociale ti aspetta!!! F\*\*\*\*\* il diploma!!!!



...fine...