HACKER JOURNAL

Anno 2 - N. 37
6 Novembre - 20 Novembre 2003

Boss: theguilty@hackerjournal.it
Editor: grand@hackerjournal.it
Contributors: Salvatore Aranzulla, Bismark.it, Il Coccia, DaMe`, Lele, pctips , Angelo Rosiello, >>>----Robin---->

DTP: Cesare Salgaro
Graphic designer: Dopla Graphic S.r.l. info@dopla.com
Copertina: Gregory Peron
Publishing company
4ever S.r.l.
Via Torino, 51
20063 Cernusco S/N (MI)
Fax +39/02.92.43.22.35
Printing
Roto 2000
Distributore
Parrini & C. S.PA.
00189 Roma - Via Vitorchiano, 81-
Tel. 06.33455.1 r.a.
20134 Milano, V.le Forlanini, 23
Tel. 02.75417.1 r.a.
Abbonamenti
Staff S.r.l.
Via Bodoni, 24
20090 Buccinasco (MI)
Tel. 02.45.70.24.15
Fax 02.45.70.24.34
Lun. - Ven. 9,30/12,30 - 14,30/17,30
abbonamenti@staffonline.biz

Pubblicazione quattordicinale registrata al Tribunale di Milano il 25/03/02 con il numero 190.
Direttore responsabile - Editore Luca Sprea

HJ: INTASATE LE NOSTRE CASELLE
Ormai sapete dove e come trovarci, appena possiamo rispondiamo a tutti, anche a quelli incazzati.
redazione@hackerjournal.it

hack·er (hãk'ər)

"Persona che si diverte ad esplorare i dettagli dei sistemi di programmazione e come espandere le loro capacità, a differenza di molti utenti, che preferiscono imparare solamente il minimo necessario."

IL GIROTONDO DI OFFICE

Se non questo mese, sicuramente succederà il mese prossimo: una quarantina di riviste specializzate in informatica usciranno tutte con la stessa parola in copertina, Office 2003. Personalmente, la cosa mi riguarderà solo di rimbalzo: probabilmente sarà un po' più difficile aprire qualche documento (che sicuramente contiene solo informazioni che avrei potuto tranquillamente ricevere in un formato standard), magari avrò qualche problema con i messaggi email inviati con le nuove funzionalità di Outlook...

Ma l'argomento che volevo affrontare è un altro. Gironzolando nei bassifondi di Internet, si leggono post di "furboni" felici e soddisfatti di avere già installato la loro copia pirata. Un messaggio in particolare mi ha colpito, perché faceva la stima dei soldi risparmiati usando copie pirata di Office negli ultimi anni. Dal tono e da ciò che scriveva, era evidente che questa persona non usava Office per quelle sue caratteristiche uniche che lo rendono insostituibile in certi casi (principalmente, in ambienti dove l'automazione di ufficio è strettamente basata sulle funzionalità Microsoft), ma solo per leggere e scrivere qualche testo, creare qualche semplice tabella o database. Stanti così le cose, caro amico, forse dovresti rifare un po' i conti, perché ci sono delle voci di spesa che non hai considerato.

Ora memoria e hard disk costano poco, ma nel corso degli anni quanto spazio hai dovuto dedicare al mastodonte di Redmond, mentre potevi tranquillamente usare Wordpad o qualche più snello software freeware o shareware? Non hai dovuto aumentare la memoria RAM in occasione di qualche aggiornamento, o addirittura cambiare computer, perché Word non riusciva più nemmeno a stare dietro alla tua velocità di digitazione? E quanto ti sono costati in termini di tempo, problemi, scocciature, banda e rallentamenti, i mille virus e Worm che esistono e proliferano solo per colpa dei gravissimi problemi di sicurezza di Outlook?

Insomma, perché invece di calcolare quanto hai risparmiato, non provi a capire quanto ti è costato scegliere di usare Office, anche se non ne hai mai pagato il prezzo?

Inutile provare a chiedere queste cose al diretto interessato: conosco già la risposta più probabile. "Ma tutti lo usano, e devo poter essere in grado di aprire l'ultima presentazione PowerPoint con il calendario delle Veline svelate". Calendario che molto probabilmente è stato creato su una copia pirata di Office, e distribuito attraverso mille copie pirata di Outlook. E proprio qui sta il lato perverso della faccenda: copio Office perché tutti lo usano, e così facendo spingo altri a usare Office.

È un loop apparentemente infinito. Giro giro tondo, copia il mondo, cracca la terra, tutti giù per terra!

grand@hackerjournal.it

SEGNALAZIONI DAL FORUM...

Nel canale forum generale c'è un post chiamato "Sicurezza DB e relative tecniche" di mattemax80. Nel post si chiedono maggiori chiarimenti sulla protezione di DB (si cita il numero 32 della rivista, dove si tratta SQL injection).
Altro post che considero interessante è quello nella sezione newbie del canale sicurezza postato da metal_lord "attacco ricorrente". Nel post chiede info su un presunto attacco durante le sezioni di connessione usando p2p e mIRC (il firewall informa di portscanning da parte di azzurra, tiscali e/o telecom...).

(Walther)

FREE HACK NET

freeHACKnet è il servizio gratuito di collegamento a Internet targato Hacker Journal: indirizzo email @hackerjournal.it con 5 Mbyte, accesso super veloce fino a 128 Kbit al secondo (ISDN multilink PPP), server newsgroup, controllo anti virus e anti spam. Niente abbonamento, nessuno sbattimento, paghi solo la tariffa telefonica urbana.
Corri subito a iscriverti su
www.hackerjournal.it/freeinternet

I vostri siti...

Vorrei segnalare il mio Portale Informatico www.lhcsite.tk. Io e i miei "collaboratori" speriamo che con questo messaggio la LHC

possa diventare quello che tutti noi speriamo...Un VERO Portale sulla Sicurezza Informatica.

Nuova password!

Ecco i codici per accedere alla Secret Zone del nostro sito, dove troverete gli arretrati, informazioni e approfondimenti interessanti. Con alcuni browser, potrebbe capitare di dover inserire due volte gli stessi codici. Non fermatevi al primo tentativo!

user: ana3
pass: ar2

Prossimo Guestbook

Per te un computer portatile è... una cosa da fighetti? Il futuro del PC? Il sogno che non puoi permetterti? Oppure cosa? Rispondi con una decina di parole, e invia il tutto a guestbook@hackerjournal.it. Tutte le risposte verranno pubblicate sull'ultima pagina del prossimo numero.

mailto:
redazione@hackerjournal.it

USCIRE DALL'AZIENDA

Un mio caro amico lavora come impiegato in un'azienda multinazionale dove tutti i PC sono conessi in rete, se per sbaglio dovesse installare un programma come winmx, kazaa, o altri verrebbe subito intercettato dalla sede principale all'estero e quindi richiamato per disinstallare subito il software (cosa che gli e' gia' successa installando un vocabolario online). Mi ha chiesto se c'e' una possibilita' di poter scaricare Mp3 o altro utilizzando programmi sopracitati o in altri modi senza essere intercettato (anche perche' i suoi superiori tramite vie traverse riescono a fare cio' e sia a lui che a me non sembra giusto).
Scusate se non sono stato in grado di esprimermi chiaramente ma purtroppo seguo la vostra rivista da poco e sto' incominciando adesso a capirci qualcosa!

Notturno

L'azienda ha il diritto di limitare o impedire l'uso privato delle proprie risorse, e l'uso di stratagemmi (che in effetti esistono) potrebbe causare dei guai al tuo amico qualora venisse scoperto. Piuttosto, bisogna precisare una cosa: per poter sorvegliare a distanza i dipendenti, con qualsiasi metodo (quindi anche con l'uso di software di controllo dell'uso del PC), l'azienda è tenuta a comunicare ai rappresentanti sindacali i metodi, le finalità e i limiti del controllo. Il fatto che il controllo avvenga dalla sede centrale, all'estero, non la esime da questo dovere.

MITNICK IN ITALIANO

Vorrei sapere se possibile trovare, tradotto in lingua italiana, il libro di Kevin Mitnik di cui voi avevate parlato e pubblicato una parte in un numero passato di HJ.

Federico

Sì, il libro è stato pubblicato nella collana "Serie Bianca" di Feltrinelli col titolo "L'arte dell'inganno - I consigli dell'hacker più famoso del mondo". Costa 15 euro e lo puoi trovare in ogni libreria.

FAQ YOU

Come si fa ad avere il controllo di un altro computer? Che programmi bisogna avere per connettersi a questo computer preso in considerazione? Bisogna avere xforza l'IP del computer a cui vogliamo connetterci? Tramite l'email si puo' risalire all'IP? In questo caso che programmi bisogna utilizzare?
Vi ringrazio anticipatamente x la risposta che sono sicuro sarà esauriente alle mie domande.

(lettera firmata)

Fai male a essere così sicuro. Infatti, non rispondiamo a queste domande, specialmente se formulate in questo modo. Ovviamenete, leggendo HJ e cercando di imparare qualcosa dalla documentazione disponibile, puoi arrivare da solo alle giuste conclusioni.

COPERTINE E TESTI MUSICALI

Ciao! Avrei un quesito da porvi. Sto per aprire un sito web; il sito dovrebbe contenere informazioni riguardanti gruppi misicali (nome della band, logo della band, storia della band, discografia, copertine, testi dei brani, traduzioni, foto varie del gruppo, ecc). Devo chiedere particolari autorizzazioni al fine di evitare la chiusura del sito web o magari qualche denuncia?? Mi hanno consigliato di specificare "l'appartenenza del Copyright nelle pagine riguardanti la band"!! Questo risolverà il problema??

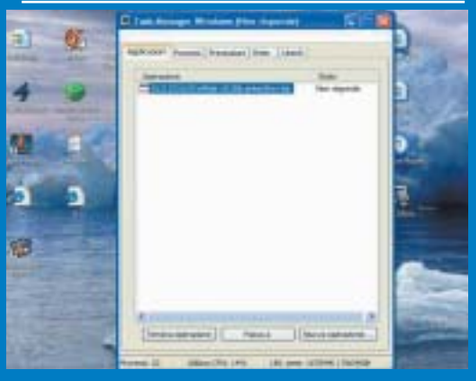
Piero

In teoria, sono protetti dal diritto d'autore la copertina e i testi. Se per la copertina in genere non ci sono problemi, a meno che non metti un'immagine ad alta risoluzione che può essere utile per stampare copertine di CD pirata (in fin dei conti, fai promozione al gruppo...), il problema vero è rappresentato dai testi, che non possono essere pubblicati in forma integrale. E' però consentito citarne alcuni passaggi all'interno di un articolo (per esempio, una critica o una recensione).

ANTISPAM INTRUSIVO

Partecipo a diversi FORUM e MailingList presenti in rete. Recentemente, in uno di questi ultimi ci è giunto l'AVVISO che riporto qui sotto.

Tech Humor



La sezione che più mi piace della rivista sono le Tech Humor. In allegato a questa e-mail c'è un'immagine insolita, il mitico Task Manager di Windows, il programma che serve a sbloccare gli altri programmi, si ritrova bloccato!
Per me è una stranezza, ma può darsi che mi stia sbagliando, se la cosa fà sorridere anche voi vi prego di bubblicarla.

Neo88.de

la sua e-mail:
Oggetto: XXXXXXXX
Inviata il: 15-10-2003
a: xxxxx@email.it
non puo' essere recapitata in quanto la casella xxxxx@email.it e' protetta con CiaoSpam, il servizio di protezione dalla posta indesiderata di Email.it
Se vuole recapitare correttamente la sua e-mail occorre seguire, entro 3gg da oggi e solo per questa prima volta, la seguente procedura di auto-certificazione
https://secure.email.it/cgi-bin/cs.pl?id=xxxxxxxxxx&lang=ITA
Alla scadenza del 3° giorno, in mancanza della auto-certificazione, l'e-mail XXXXXX verra' eliminata e quindi non recapitata al destinatario.
[...]

A parte che ogni appartenente alla MailingList è dotato di software Antivirus - Firewall e chi più ne ha più ne metta, e di conseguenza siamo già sufficentemente protetti da attacchi di differenti tipi, non è questo un classico esempio di "INTRUSIONE" non richiesta agli appartenenti alla Mailing medesima (prima dialogavamo senza problemi e all'improvviso compare "questa cosa" che ci blocca il servizio)? Il Forum riguarda attività sportive ed altro, ma non ha nulla a che vedere con l'indirizzo xxxxx@email.it, e men che meno le nostre e-mail erano state indirizzate ad esso. Potreste darci un Vs. parere sulla questione ed eventualmente indicarci i metodi per eludere tali "condizionamenti"?

Pier Paolo (Snake_bo2003)

Evidentemente, il proprietario dell'indirizzo xxxxx@email.it (la prima parte è stata cancellata) è iscritto alla mailing list, e ha richiesto l'attivazione del servizio CiaoSpam di email.it. Da qualche tempo, questo servizio di posta utilizza questo sistema anti-spam, effettivamente un po' invasivo. In pratica, ogni email con mittente sconosciuto al destinatario viene messa in sospeso; se il mittente vuole far pervenire il suo messaggio al destinatario, deve visitare il sito di email.it e seguire la procedura, che non può essere automatizzata. In questo modo si neutralizzano gli strumenti di mass-mailing tipici degli spammer. L'idea è interessante, e le intenzioni lodevoli (limitare lo spam, appunto), ma il periodo di tre giorni è decisamente troppo, troppo corto. Se io mando un messaggio a un utente di email.it e per tre giorni non mi collego a Internet, il mio messaggio non arriverà a destinazione.

CONTROLLO REMOTO

Ciao a tutti. Sono un ragazzo che legge assiduamente la vostra rivista. Sul numero 34 nell'articolo "CONTROLLO REMOTO IN C" veniva spiegato come poter collegare 2 pc tramite la rete e far eseguire dei comandi a "distanza". La mia domanda era come e dove trovare materiale informativo sulle procedure in C per leggere dei file o spegnere il computer e sopratutto far eseguire dei comandi dos da C dato che avendo due computer collegati in rete a casa in due stanze diverse mi scoccia ogni volta fare la staffetta per controllarli contemporaneamente.Ho già provato su internet ma sia la lentezza del mio collegamento sia l'inettitudine del saper cercare nella rete mi hanno fatto perdere la speranza. Ora sono nelle vostre mani...

rasputin84

PRECISAZIONE SUL CRACK DEI PIN

Sul numero 35 di hj a pag. 14 c'è un piccolo errore nella descrizione della tecnica di crack del PIN di una carta Bancomat.
Nell'articolo si afferma che l'HSM darà come valido il PIN 0000 se nel PIN vero è presente il numero 3 o la lettera C.
In realtà l'HSM dà una risposta positiva quando nel PIN reale (esadecimale) non sono presenti né il numero 3 né la lettera D.
Inoltre, il numero medio di tentativi da eseguire per determinare tutte le cifre del PIN è 7 (4 nella migliore delle ipotesi e 10 nella peggiore).

bisi

La correzione è giusta. L'HSM risponde no se la cifra è presente e sì se non è presente. E' stato un errore di distrazione perché la conclusione cui arriva il lettore si evince dall'articolo stesso. Chiedo comunque scusa.
Per quanto riguarda la seconda correzione il lettore sbaglia. Ipotizziamo infatti di avere un PIN 1111.
l'HSM risponde no al tentativo relativo alla cifra 1 e alla lettera B e sì a tutti gli altri tentativi. Dal momento che non siamo sicuri di quante cifre uguali sono presenti nel PIN è necessario fare tutti e 10 i tentativi a meno che non abbiamo ricevuto già 4 NO. In quel caso possiamo fermarci ma in tutti gli altri casi è necessario continuare fino al decimo tentativo per essere sicuri della composizione del PIN.

dec0der

NEWS

HOT!

ALTRO GIRO ALTRA FALLA

Anzi cinque. Questo è il numero di bug, quattro critiche e una importante, che attualmente affligge diverse versioni di Windows. Quali danni possono causare? Bazzecole. Il nostro computer rischia "solo" di contrarre i peggiori virus, semplicemente in seguito alla visita di un sito Internet o alla ricezione una mail in html. Tutti i prudenti che si sentono al sicuro perché non aprono allegati sconosciuti stiano all'erta. Ma il consiglio vale per tutti: di corsa ad aggiornare Windows, sul sito di Microsoft update.

IL PIENO GRAZIE

Tra un po' per usare il portatile dovremo fare il pieno alle batterie. Non è una battuta. Ormai la ricerca sulle fuel cell, le batterie a combustibile, sta facendo passi da gigante. Con circa 300 cc al giorno di alcool metilico, oggi il prototipo della batteria presentata da NEC Computer può garantire a un notebook il funzionamento di circa cinque ore. Ma l'obiettivo è quello di arrivare a quaranta ore di autonomia, vale a dire a una settimana lavorativa. A breve insomma vedremo altre abitudini caratterizzare il nostro lunedì mattina. Dopo i classici cappuccio, cornetto e quotidiano, un rabbocco di carburante alla batteria del portatile e... via verso una nuova settimana lavorativa.

LIBERO UN PO' MENO LIBERO

Tutto quello che è gratis prima o poi smette di esserlo. Sembra il triste finale di una favola, e invece è la realtà di molti servizi free online. Finito il periodo delle vacche grasse per i download p2p, adesso tocca alle mail. Dal primo novembre, chi ha una casella di posta con dominio libero.it, inwind.it, iol.it e blu.it, può gestire il traffico postale tramite programmi dedicati (Eudora, Outlook ecc) solo accedendo alla connessione tramite Libero e Libero Free. Gli altri o la controllano via browser o devono pagare. Pagare una somma mensile che va da 1 euro e 25 a 2 euro e 50, a seconda dello spazio Web desiderato, per abbonarsi ai servizi MalL e MailXL. Libero cerca di inzuccherare la pillola ricordando che per ogni nuovo abbonamento, verrà devoluto un euro all'UNICEF per la costruzione di una scuola in Congo. Ma noi, abituati ad approfittare della loro generosità senza limitazioni, ci metteremo un po' far mente locale. Soprattutto perché ci hanno avvertiti del cambio di rotta con soli dieci giorni di anticipo. Anche se a caval donato non si guarda in bocca, siamo convinti che potevano sforzarsi di dare un margine maggiore di preavviso.
Con una connessione Libero sarà invece ancora possibile scaricare messaggi di altri provider, sempre che questi lo permettano.

PIÙ VELOCI CON LA LUCE

Il prezzo del carburante è salito alle stelle? Bene, noi saliremo alle stelle senza carburante, alimentando i motori degli aerei con il laser. Questo è ciò che sognano alcuni scienziati della NASA, che hanno messo a punto un aereo il cui motore funziona con elettricità generata da celle fotosensibili. Il velivolo in questione pesa poco più di trecento grammi, è realizzato con leggerissimo legno di balsa e fibra di carbonio, e ha un'apertura alare di circa novanta centimetri. Il prototipo, con le dovute migliorie, potrebbe rivoluzionare il modo di funzionare dei satelliti. Niente del genere in previsione per aerei civili. Stiano tranquilli dunque i piccioni dei nostri cieli. Al momento non corrono nessun rischio di finire arrostiti dai raggi laser di rifornimento.

ARRIVA LA PANTERA

Tranquilli: niente proteste, niente occupazioni in Università. Panther, è semplicemente il nome di battesimo dell'ultima release del Mac OS X. La versione 10.3 dello Unix di Apple ha almeno centocinquanta nuove funzionalità rispetto la precedente. Tra le tante eccone alcune in ambito server. Il tool Server Admin, che permette l'amministrazione del sistema e la gestione dei software open source integrati in Mac Os X; Samba 3, che gestisce login e supporto delle Home Directory dei client Windows; l'application server Jboss, per utilizzare le applicazioni J2ee. Infine l'Open Directory 2, per l'hosting di directory Ldap e servizi di autenticazione Kerberos scalabili.
Inoltre con Panther Server, l'utente può usare sia un Mac che un Pc Windows per accedere al proprio account.

SARÀ BELLO O BRUTTO?

È grosso, molto grosso, e fornisce prestazioni quaranta volte superiori alla norma. Le signore non si facciano illusioni e i maschietti tirino un sospiro di sollievo: è un computer quello di cui stiamo parlando. Un gigantesco computer che potrebbe rivoluzionare il sistema di previsioni del tempo. Il bestione si chiama Earth Simulator, e oltre a essere spaventosamente potente, è anche spaventosamente esteso. Figurarsi che occupa la superficie di ben quattro campi da tennis. Insomma non è roba da appoggiare sulla scrivania del Colonnello Bernacca e dei suoi degni successori. Tanta vastità è dovuta al fatto che è composto da ben 640 nodi linkati tra loro da oltre 83.000 cavi. Il computer dovrebbe essere in grado di fornire sia previsioni su piccola scala, anche su aree non più vaste di dieci chilometri quadrati, e soprattutto dovrebbe riuscire a prevedere con sufficiente anticipo eventi straordinari come ondate di caldo e tempeste.

SCIMMIE, VIDEOGIOCHI E RICERCA

Mentre alcune scimmie si divertono con un videogioco, chi è rimasto vittima di paresi o ha problemi che gli impediscono il movimento degli arti nutre una speranza in più di poter comunicare col mondo. Sembra impossibile un connubio tra due realtà apparentemente così distanti. E invece esite. Il dottor Miguel Nicolelis, ha condotto un esperimento impiantando minuscoli dispositivi nel cervello di due scimmie invitate poi a interagire con un videogame. A un certo punto quando le scimmie si sono accorte di poter controllare i gioco con il pensiero senza dover ricorrere al movimento, hanno smesso di muoversi e hanno continuato a giocare semplicemente "decidendo" che mosse fare. Gli elettrodi impiantati infatti trasmettono segnali a un sistema di computer in grado di decifrarli e di tradurli in movimento. L'utilità di un simile ritrovato per chi non è in grado di muoversi è lampante. Garantito, nessuna brutalizzazione per gli animali. I dispositivi introdotti nel cervello hanno la sezione inferiore a quella di un capello umano. Addirittura, si dice che si stia sperimentando la procedura anche su un campione di soggetti umani. Ma per ora sulla faccenda grava una cortina di riservatezza.

E GLI UTENTI STANNO A GUARDARE

Ancora battaglie nell'ambito del download musicale. Questa volta però gli utenti fanno da spettatori. Sono i colossi a mordersi tra loro. La scaramuccia coinvolge la newyorkese E-Data e big del calibro di Tiscali, Microsoft MSN e OD2 di Peter Gabriel. Secondo E-Data il brevetto del download le appartiene e dunque tutte le società che forniscono questo tipo di servizio a pagamento, in America e in altri nove stati europei, stanno violando diritti d'autore. Quindi devono pagare la licenza. In passato E-Data ha già vinto alcuni contenziosi sull'argomento. Vediamo come se la caverà con questo.

HOT!

VELOCE MA SEMPLICE

Arriva dalla Finlandia un possibile concorrente della ADSL. Proposto da

Teleste, ETTH (Ethernet To The Home) è un servizio che promette di trasmettere dati alla velocità di 10 mega al secondo per l'utenza residenziale e fino a 50 mega al secondo (in futuro anche 100) per l'utenza business. Il grosso vantaggio di questa offerta è che utilizza le vecchie strutture, solo leggermente modificate. I dati viaggiano infatti sui normali cavi coassiali, senza necessità di nuove cablature e apparecchi specifici. Basta una normale presa Ethernet e siamo pronti per usare ETTH.

L'INTIMO CHE TI SALVA LA VITA

Niente a che vedere con cinture di castità o diavolerie del genere. Cose serie. Gli scienziati del Eindhoven's Philips Research Labs hanno messo a punto slip, canottiere e reggiseno salvavita. Gli indumenti hanno dei sensori collegati a un telefono mobile in grado di inviare un segnale d' allarme e chiamare un'ambulanza nel caso che chi li indossa sia colto da un attacco di cuore. L'invenzione ottima per i cardiopatici, potrebbe essere un perfido spunto per creare un modello di intimo antipalpeggiamento e antisfrugugliamento per malcapitate adolescenti figlie di genitori bacchettoni. Ma anche fosse, in quattro e quattr'otto si troverebbe una sc…crhackatoia. Come era il detto? L'ormone fa l'adolescente hacker?

HOT!

MOZILLA SI RINNOVA

Nuova versione in arrivo per il browser open source Mozilla. Le migliorie riguardano il client e-mail/newsgroup MailNews e l'editor HTML Composer. Tutti e due ora dispongono di un correttore ortografico, mentre Composer garantisce una gestione più sofisticata del layout delle pagine Web. La release 1.5 è un primo passo verso la fusione di Mozilla con il browser Firebird e il client di posta elettronica Thunderbird.

SBAGLIANDO SI IMPARA

La RIAA aggiusta il tiro. Se è di qualche tempo fa la notizia che la più potente organizzazione di discografici aveva toppato clamorosamente denunciando nel mucchio, persone che nulla o poco centravano con lo sharing e il downloading illegale, gli ultimi aggiornamenti provenienti da quel fronte sono un po' diversi. Prima di partire in quarta con azioni legali, la RIAA ha scoperto che forse è più intelligente ed economicamente vantaggioso cercare al mediazione. A differenza dei 261 malcapitati della prima ondata di denunce, i 204 del secondo giro, sono stati avvertiti con un ultimatum che li invita a una composizione extragiudiziale senza spese legali.

SALUTE SCHEDATA

Torna alla ribalta la questione della carta sanitaria elettronica. Proprio in questi giorni dovrebbe infatti essere discusso il decreto che ne stabilirà l'eventuale attivazione e normativa. Se il decreto 269 venisse approvato e trasformato in legge, si rischierebbe schedatura di massa della situazione sanitaria dei cittadini italiani. Con tutti i rischi che ne conseguono per il mantenimento del diritto di privacy.Oltre al problema della garanzia della privacy, sorgono altri interrogativi che riguardano per lo più i costi della manovra. Quanto costerà realizzare una simile mole di tessere magnetiche? Quanto costerà installare apparecchiature in grado di trasmettere le informazioni su ricette e prestazioni sanitarie? E infine, quanti bei soldi guadagnerebbero le aziende telefoniche con tutto questo gran viaggiare di dati?

ECCHIP! SALUTE

Dall'Oriente vengono le principali sindromi influenzali, dall'Oriente vengono nuove tecnologie per una diagnosi precoce delle stesse. Pare che l'istituto di genetica di Singapore abbia messo a punto un dispositivo elettronico grande quanto una monetina che a contatto con saliva o secrezioni nasali è in grado di rilevare e riconoscere eventuali agenti patogeni. Il chip, che sta per essere testato in maniera massiva, si rivelerà particolarmente utile per malattie che allo stadio iniziale presentano gli stessi sintomi. Pensiamo alla SARS, alla comune influenza o ad altre patologie dell'apparato respiratorio. La diagnosi differenziale precoce aiuterà a contenere il rischio di epidemie e a ottenere un maggior numero di guarigioni.

UNA MELA SU WINDOWS

I tempi sono proprio cambiati. Una volta il Mac era il Mac e il PC era il PC. E un muro tracciato di netto divideva le due piattaforme, così come le due fazioni di utenti. Oggi non solo la linea di confine è più incerta. Addirittura Apple ha realizzato un software niente meno che per Windows, iTunes. Ed è già scompiglio per tante ragioni. Sony cerca di correre ai ripari pensando a servizi concorrenziali. Napster 2 è pronto per essere rilasciato, sebbene gravato da pesanti limitazioni. Anche le polemiche non mancano. Il servizio di acquisto musica iTunes per Windows è attivo solo da pochi giorni e solo negli Stati Uniti, eppure sono molti i bug segnalati. I più futili riguardano l'estetica. Qualcuno si lamenta che non supporti le skin. I più seri riguardano problemi di installazione e blocchi di Windows. Che si tratti di problemi reali e non di voci messe in giro dalla concorrenza lo dimostra la dichiarazione dello stesso direttore marketing di Apple. In alcuni casi, ha ammesso il portavoce, dopo l'installazione di iTunes, può verificarsi un congelamento del PC. Per concludere il rosario di lamentele, aggiungiamo quelle di chi denuncia che il sistema di trasmissione dei brani di iTunes sia a bassa protezione e quindi favorisca la pirateria informatica.

ACCESSORI PER PIGRONI

Stufi del classico mouse un po' babbione che più che clicca e trascina non capisce? Aprite le orecchie e la home pagine della FingerWorks (http://www.fingerworks.com/igesture.html). Questa azienda americana ha progettato una tavoltetta che sfrutta i normali gesti della mano per comunicare con il computer. Il futuristico dispositivo chiama Gesture Pad e costa 159 dollari. I gesti da fare sulla tavoletta sono molto intuitivi: si uniscono le dita pizzicando, per tagliare un pezzo di testo, si muove un dito come per aprire uno sportello per aprire un file e via dicendo. Oltre alla Gesture Pad, sempre della stessa ditta, c'è TouchStream Keyboard, una tastiera che comprende anche la Gesture Pad.

SSL BATTE IPSEC

Pare che ormai la tendenza sia questa: alle Vpn basate su Ipsec si preferiscono Vpn basate su Ssl. Lo aveva annunciato F5 Networks quando aveva presentato il nuovo FirePass Controller, ma allora poteva sembrare un giudizio di parte. E invece, oggettivamente questa soluzione ha i suoi vantaggi. A differenza delle Ipsec, che pur restando una buona soluzione necessitano di personale qualificato per procedure di installazione, gestione e manutenzione, la FirePass permette di autorizzare dinamicamente gli accessi alle applicazioni necessarie agli utenti. In più supporta web host, terminal server, cellulari, applicazioni client-server e Windows Desktop. Il FirePass Controller garantisce uguali prestazioni su sistemi Unix/Linux che utilizzano applicazioni X-Windows. Sempre di F5 Networks ci sono poi i server FirePass di uRoam, permettono l'accesso basato su Web a ogni applicazione di rete da qualsiasi client, senza configurazioni specifiche per i client e modifiche alle risorse di back-end.

HO VISTO LA LUCE: ERA UNA TASTIERA

Per restare in tema di tecnosfizi, vale la pena ricordare Canesta Keyboard, la regina di tutte le tastiere virtuali. Di fatto tale tastiera non esiste. Non perché non l'abbiano ancora inventata, ma perché la sua sagoma viene proiettata dai un fascio di luce su una qualsiasi superficie piatta. Digitando sui quadratini illuminati della "tastiera", grazie a un laser che decifra i nostri movimenti, impartiremo comandi a palmari e cellulari. Incredibile. Ah e naturalmente per non farci patire la nostalgia per il caro buon vecchio hardware, mentre digitiamo su sui tasti virtuali potremo ascoltare un piacevole ticchettio di macchina per scrivere old style.
Per saperne di più:
http://www.canesta.com

SIAMO PRONTI AD ESSERE SICURI?

È stato approvato lo scorso giugno, ma entrerà in vigore dal primo gennaio 2004 il Testo Unico, documento che contiene le norme per la tutela della sicurezza e della privacy in ambito aziendale. Molte sono le novità in materia stabilite dal documento, anche se di fatto le aziende che si dichiarano pronte ad adottarle sono ancora poche. Secondo i dati di un sondaggio stilato tra giugno e settembre in 400 aziende, un terzo delle quali del milanese, i risultati non sono poi così confortanti. È andata così così per i firewall: il 71% dichiara di averli, ma solo il 40% risulta averne uno adeguato.Quasi tutti risultano a norma riguardo l'antivirus: il 99% del campione delle aziende intervistate dichiara di averne uno in uso, anche se non tutti sono adeguati. Il responsabile della sicurezza, una figura professionale ormai divenuta obbligatoria, al momento è adottato solo dal 49% delle aziende interpellate. I peggiori risultati se li è guadagnati la firma digitale: la ha adottata solo il 17% delle aziende italiane. La sicurezza è importante. Adeguatevi aziende, adeguatevi.

HOT!

SUSE ARRIVA A SCUOLA

Scuole, studenti, università e organizzazioni no profit, volete risparmiare? Potete aderire al programma lanciato a fine ottobre da SuSe Linux Education Program che permette di acquistare server, sistemi operativi e firewall con uno sconto anche del 40%. Oltre al notevole risparmio l'iniziativa è vantaggiosa perché consente agli istituti, di personalizzare le soluzioni a seconda delle proprie esigenze.
Per informazioni: http://www.suse.de/it/

FINZIONE FINO A QUANDO?

Dei processi non proprio ortodossi se ne è sentiti un po' di tutti i colori. Ma quello istruito dall'avvocato Martine Rothblatt li batte tutti, senza appello. Si tratta di una causa intentata niente meno che da un'intelligenza artificiale contro la società a cui appartiene e che ha deciso di "spegnerla". Naturalmente anche se condotto nell'assoluto rispetto delle regole processuali vigenti, si tratta di un processo finto. Ma siamo sicuri di dovere attendere così tanto prima di vederne uno del genere nelle aule del tribunale della nostra città? A volte la fantascienza è meno lontana di quanto si creda…

Anche i veri non rapinano

Ma come, per mesi vi abbiamo stressato con l'etica dell'hacker, ripetendo che bisogna distinguere tra

I giornalisti ancora oggi identificano l'hacker con il criminale informatico, ma quando non lo fanno distinguono tra hacker "buono", colui che non causa danni, e "cracker" dando per scontato che il termine voglia dire **hacker "cattivo"**. Descrivono il cracker come un "criminale" o un "pirata informatico", uno che ama bucare i sistemi, entrare e **violare la sicurezza solo per il gusto di esserci riuscito** e con l'obiettivo di sottrarre dati o danneggiarli. Per dimostrare quanto affermato, **ricorrono persino al Jargon File che in genere, per altre questioni, fingono di ignorare**. Ma ad una lettura più attenta del gergo hacker si scopre che il cracker non è mai definito, almeno apertamente, criminale e che eludere la sicurezza di un sistema non è un crimine nemmeno per un "hacker buono".

Jargon File

Nasce come dizionario per tradurre il gergo degli hacker in una lingua "comprensibile dagli umani". Vi si possono infatti trovare le definizioni di più di 2000 termini. A differenza di un normale dizionario, però, rivestono molta importanza anche le varie prefazioni e appendici, mirate a definire meglio la figura e l'etica dell'hacker. Qualcosa di simile al Jargon File esiste anche in italiano, anche se non si dilunga sulla filosofia hacker. È il Gergo Telematico curato da Maurizio Codogno (http://xmau.com/gergo).

>> La prima abilità di un hacker

Se il significato 6 del Jargon File descrive l'hacker come **"un esperto o un entusiasta di qualsiasi tipo"**, il significato 5 limita la competenza dell'hacker a un unico campo. E', infatti, "un esperto di un particolare programma, o uno che ci lavora frequentemente", "un hacker di UNIX". Quando si tratta di veri esperti in un settore, però, i termini più appropriati sono **wizard** ("mago") o **guru** ("santone"). I significati 1 e 7 invece non solo danno una definizione più precisa di hacker, ma spiegano molto bene quale sia l'abilità di base di qualunque hacker. Ciò che l'hacker sa fare veramente è "**studiare un sistema (anche non informatico), scoprirne debolezze, peculiarità e caratteristiche nascoste, e utilizzarle per scavalcare o aggirare i limiti**", persino quando è "**un ficcanaso maligno che tenta di scoprire informazioni delicate frugando qua e là**" (significato 8). Quando si parla di hacker e in particolare di questa sua abilità, non si può non parlare di sicurezza. Un hacker, e quindi non solo un cracker, studia e conosce così bene i sistemi informatici che può scoprirne facilmente i buchi (bugs, bachi) attraverso i quali può aggirare tutti i sistemi di sicurezza ed entrare nel cuore della macchina fino a prenderne il controllo. Oppure sa entrare nel codice sorgente del software, individuare quelle porzioni di codice che fanno funzionare male o che limitano il programma, può modificarlo, migliorarlo ed anche in questo caso, se vuole, può aggirare le restrizioni. Un hacker sa fare la stessa cosa anche con i sistemi umani che pure non sono privi di bug, come dimostrato da alcune sue attività ben note, che si svolgono al di fuori della rete e del mondo dei computer. Il **Tech Model Railroad Club** del MIT promosse ad esempio il **Midnight Requisitoring Committee**, incursioni notturne nei magazzini e il **Lock Hacking**, l'hackeraggio di serrature delle porte per portar via i componenti necessari per costruire macchinari più efficienti o per usare gli strumenti contenuti nelle stanze chiuse. Ci sono poi il **social engi-**

CRACKER le banche!

ra hacker e cracker, e ora cambiamo idea? Non proprio, ma leggete cosa ha da dire a proposito DaMe`...

neering, l'ingegneria sociale, una serie di metodi per spacciarsi per un altro ed ottenere così delle informazioni riservate; il **vadding**, l'esplorazione di posti dove le persone comuni normalmente non hanno accesso; il **trashing**, rovistare nei rifiuti, ad esempio di "Mamma Telecom", per saperne di più sul mondo delle comunicazioni in Italia (S. Chiarelli & A. Monti, Spaghetti Hacker).

>> Cracker: il Jargon File insegna

Sfruttando le debolezze di un sistema o di un programma si può ottenere ciò che non si è autorizzati ad avere. Alcuni dei metodi applicati dagli hacker sono **in contrasto con il diritto di proprietà**, e finalizzati quasi tutti all'accesso a informazioni a cui non si avrebbe diritto. Questo spiegherebbe in parte perché gli hacker siano temuti, considerati delle figure estremamente scomode, ma di certo non giustifica l'uso da parte dei giornalisti del termine "criminale" neanche quando è associato al cracker. Nel Jargon File si afferma che il cracker è "**uno che elude la sicurezza di un sistema**" (breaks security on a system), sa cioè violare le "serrature", i codici di accesso o i sistemi di protezione dei software e dei sistemi informatici. Non c'è scritto che rompe o danneggia i sistemi! Il termine è stato coniato nel 1985 circa dagli hacker "**in difesa contro l'uso scorretto del termine "hacker" da parte dei giornalisti che lo intendevano nel significato 8 del Jargon File**" e cioè come "ficcanaso maligno" (malicius meddler). "Un precedente tentativo di instaurare il termine "worm" in questo senso nel 1981-82 circa su USENET, fu un fallimento".
Il termine cracker viene quindi scelto per distinguere l'hacker da **uno che fruga e rovista** (by poking around) nei sistemi, non per distruggere ma per scoprire informazioni delicate (to discover sensitive information). Non viene però chiarito se siano queste informazioni ad essere distrutte o ad essere utilizzate in maniera illecita e cioè per altri scopi che vadano ben oltre il puro piacere di essere riusciti a "scoprirle". Il cracker entra in un sistema senza permesso e questo può anche essere considerato illegale, ma da qui a dire che ciò che è "illegale" sia anche "criminale" ce ne vuole! Certo è che il Jargon File descrive il cracker, non come un criminale, ma come un "maligno", "**una separata e più bassa forma di vita**" (a separate and lower form of life) e persino come un "**perdente**" (pretty losing), perché non riesce ad immaginare un modo più interessante di utilizzare il computer che quello di penetrare nei sistemi informatici altrui. Ciò che fa un cracker, sa farlo anche un hacker, ma "Mentre ci si aspetta che qualunque vero hacker abbia crackato per diletto e conosca molte delle tecniche di base, chiunque abbia passato lo "stato larvale" **ci si aspetta che**

abbia superato il desiderio di farlo". L'atto di penetrare in un sistema informatico, infatti, "contrariamente al mito diffuso non richiede una qualche misteriosa brillantezza, ma piuttosto persistenza e la tenace ripetizione di utili e ben noti trucchetti e lo sfruttamento di debolezze comuni nella sicurezza dei sistemi che si intende attaccare. Di conseguenza **la maggior parte dei cracker sono solo hacker mediocri**" (mediocre hackers).

>> L'assalto degli script kiddies e dei lamer

Anche l'hacker, come si è visto, sa aggirare i limiti dei sistemi e dei software,

ma il suo interesse non è rivolto esclusivamente a questa attività. Un tempo praticare il cracking per un hacker significava soprattutto **sviluppare da soli i programmi per entrare nei sistemi informatici**, essere ottimi programmatori, conoscere il funzionamento dei software, il linguaggio macchina, i comandi diretti del processore. A partire dagli anni 80, con la diffusione della cultura informatica, la situazione cambia. Si avvera il sogno di Apple e Microsoft: "un computer in ogni scrivania e in ogni casa" e sono sempre più numerose le riviste informatiche che insegnano programmazione e persino tecniche particolarmente avanzate. Negli anni 90 il computer è già diventato un elettrodomestico e le riviste, ora un po' più scadenti, spiegano soprattutto come usare i programmi. Ed è proprio in questi anni che **si affermano gli hacker mediocri**! Chiunque abbia un minimo di conoscenze informatiche, e ormai sono pochi a non averne, e disponga di alcuni specifici software, sviluppati ovviamente da altri, ma disponibili in rete, **può facilmente lanciare degli attacchi di vario genere o nascondere la propria identità per penetrare in un sistema informatico**. Come nel caso dei metodi applicati ai sistemi umani, qualche volta basta semplicemente essere furbi e un po' psicologi. Tutti possono praticare il cracking, e non solo gli hacker allo stato larvale, ma **anche i veri e propri criminali**, quelli che tanto per intenderci rubano i codici delle carte di credito per poi trarne profitto personale. E persino i ragazzini che i giornalisti definiscono hacker e che in realtà altro non sono che degli **script kiddies** o, quando danneggiano, dei **lamer** o **wannabe**, in pratica delle persone davvero poco esperte, che non dispongono neanche delle conoscenze di base necessarie.

"Il passaggio di consegne del mondo dei computer dagli hacker alla gente comune, spiega Valerio Capello nel suo documento *Essere Hacker*, ha certamente avuto degli effetti generali positivi, ma si e' rivelato un'arma a doppio taglio, soprattutto con l'avvento di Internet: chiunque oggi può avere degli strumenti potentissimi per danneggiare gli altri, delle vere e proprie *armi digitali*, senza avere alcuna idea di come questi funzionino e come debbano essere *maneggiati*. **Si può finire in galera con la convinzione di aver perpetrato soltanto un simpatico scherzo**, anche se un po' di cattivo gusto".

>> La mancanza di pudore dei giornalisti

I veri hacker, quelli che sviluppano la tecnica e realizzano dei programmi con cui poterla sfruttare ed hanno un senso etico molto forte, verso la metà degli anni 80 sentono l'esigenza di **prendere le distanze da coloro che agiscono senza alcuna motivazione valida** e senza neanche essere tecnicamente eccellenti. Non vogliono essere confusi con gli hacker mediocri di quegli anni identificati da forze di sicurezza e dai media coi criminali, "per permettersi di reprimere indiscriminatamente qualsiasi forma di dissenso sociale verso le politiche economiche dominanti di sviluppo delle nuove tecnologie della comunicazioni" (Di Corinto e Tozzi), forme di dissenso che gli hacker insieme ai programmatori, agli attivisti sociali e gli agitatori intellettuali hanno sempre praticato. I giornalisti, ancora oggi, **non distinguono mai, tra hacker-cracker, criminali e semplici lamer**. Preferiscono descrivere persino i lamer come degli hacker e chiamare cracker o criminale chiunque eluda la sicurezza di un sistema. "Nessuno, specialmente un giornalista, sostiene Capello, dovrebbe confondere un hacker con il povero sprovveduto finito in galera per aver utilizzato con troppa leggerezza qualche programma che gli è capitato tra le mani (anche se forse usare il termine hacker fa più notizia... **La differenza tra gli hacker**

HJ ALERT!

Nel nostro paese (e non solo) è illegale penertare nei sistemi altrui anche senza commettere vandalismi, furti o altro.
Quindi a prescindere da questioni morali, NON FATE CA..ATE!

>> Anche i cracker hanno un'etica

Bene e male per gli hacker non hanno lo stesso significato che per le persone comuni e le istituzioni. Penetrare un sistema, per un hacker, non è un atto criminale, ma **una sfida intellettuale**. Il fine non è danneggiare o provocare un danno a qualcuno, e neanche il guadagno personale, ma **trovare un mezzo di penetrare le sue difese**. Lo scopo primo per un hacker è sempre quello di acquisire nuove conoscenze o migliorare quelle che già ha. Prova piacere nell'esplorazione e nella scoperta di nuovi modi per superare i propri limiti oltre che quelli dei sistemi con o attraverso i quali opera. Sfida innanzitutto se stesso e la propria abilità. Per un hacker è un dovere etico facilitare l'accesso all'informazione e per molti hacker non ci sono informazioni a cui non si ha diritto di accesso. "Il ragionamento è semplice, spiegano Di Corinto e Tozzi. Se l'informazione è potere e la tecnologia il suo veicolo, per opporsi al monopolio dell'informazione "che serve a dominare le masse" ogni mezzo è legittimo per redistribuire informazione e conoscenza", persino penetrare un sistema. Anche un hacker quindi all'occorrenza può utilizzare il cracking e il cracker può sproteggere un programma o irrompere in un sistema per motivi etici, perchè è convinto che si possa migliorare o perché ritiene che certe informazioni e certi saperi debbano essere diffusi e condivisi da tutti. Insomma c'è una grande differenza, ad esempio, tra chi copia software per distribuirlo agli altri e chi copia il software per rivenderlo e trarne profitto. Penetrare illegalmente in un sistema protetto "per finalità etiche positive può rientrare nella definizione di cracker, **se invece avviene per scopi individuali e di profitto la definizione corretta è quella di criminale informatico**".
Il cracking, quindi, potrà anche essere considerato illegale dalle istituzioni, ma di certo va distinto dalla volgare pirateria e dai criminali, perché parafrasando un intervento di L. Felsenstein anche i "**Real Crackers Don't Rob Banks**".

e i giornalisti è che i primi hanno un'etica, i secondi neanche il senso del pudore)".
Se questi stessi giornalisti consultassero la voce "etica hacker" del Jargon File e non si limitassero solo a quella di "cracker", scoprirebbero che penetrare un sistema non è considerato neanche dai veri hacker un crimine. A proposito dell'etica hacker infatti si legge: "La convinzione che penetrare nei sistemi per divertimento ed esplorazione è eticamente a posto, finchè il cracker non commette furto, vandalismo, o diffusione di informazioni confidenziali".
I veri criminali, per gli hacker, **sono quelli che non rispettano l'etica hacker ed entrano nei sistemi solo per danneggiare**. Nel Jargon File è considerato "criminal" solo il Dark-side hacker ("Hacker del Lato Oscuro"), colui che è "sedotto dal Lato Oscuro della Forza". Benchè sia altrettanto esperto di computer e assetato di conoscenza, abbia la stessa abilità e dignità di un hacker, è più proiettato verso il male anziché il bene. Il suo "orientamento" lo rende elemento "potenzialmente" (e quindi anche in questo caso "non necessariamente") pericoloso per la comunità.

DaMe`
www.dvara.net/HK

BIBLIOGRAFIA E SITOGRAFIA

http://info.astrian.net/jargon/
Jargon File (con possibilità di effettuare ricerche)

http://www.s0ftpj.org/bfi/online/bfi7/bfi07-02.html
Valerio Capello - Essere Hacker

Di Corinto e T.Tozzi - Hacktivism. La libertà nelle maglie della rete
In particolare i paragrafi:
1.1.2. Gli Hackers non sono tutti uguali:
http://www.hackerart.org/storia/hacktivism/1_1_2.htm
3.4.3. L'underground Telematico, Il Phreaking e i Crackers:
http://www.hackerart.org/storia/hacktivism/3_4_3.htm

Dove c'è Knoppix c'è casa

Una distribuzione Linux su CD e una chiave di memoria USB (o un disco Zip) costituiscono il computer più portatile del mondo.

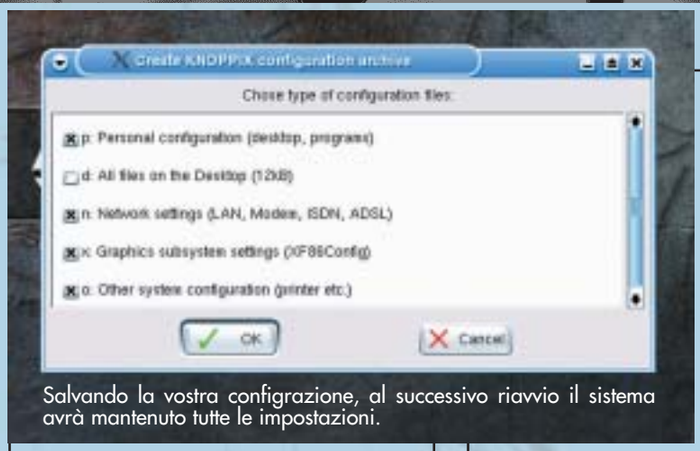
Recentemente, accanto alle tradizionali distribuzioni Linux e alle mini-distro avviabili direttamente da floppy sono comparse le cosiddette Live-CD, ovvero **distribuzioni funzionanti interamente da CD complete di interfaccia grafica**. Queste ultime, non intaccando in alcun modo le il disco fisso e i dati presenti su esso, hanno ovviamente permesso a moltissimi utenti

Windows di poter **provare le prime volte Linux senza dover affrontare il tanto temuto ri-partizionamento**; d'altro canto, moltissimi fan del pinguino ne hanno sempre una copia con sé e, siano essi al lavoro o a scuola, approfittano di ogni occasione per **impossessarsi di un computer libero e poter avviare in pochi minuti il loro sistema operativo preferito**. Tra le tante Live-Distro presenti in Internet, alcune delle quali scritte da zero ed altre derivate invece dalle distribuzioni "tradizionali", si è decisamente affermata negli ultimi tempi **Knoppix** (www.knopper.net), creata dall'ingegnere tedesco Klaus Knopper e particolarmente apprezzata per le sue caratteristiche.

Potete memorizzare la vostra home sia in un drive USB che all'interno di una partizione esistente.

>> Ottime caratteristiche

Principale punto di forza di questa distribuzione, basata peraltro sull'affidabile **Debian**, è la sua capacità di **rilevare e configurare in maniera completamente automatica tutto l'hardware installato** e il sistema grazie ad una serie di script ad-hoc.
Di fronte a tutto ciò, la scelta è indubbiamente ardua: se da un lato un Live-Cd consente di disporre di un sistema completo up'n'running in pochi minuti, un sistema installato completamente personalizzabile dove poter lavorare e, ad esempio, **modificare e salvare files nella propria home o sul desktop** (proprio come accade con altri sistemi operativi) è una comodità non da poco... In Linux le impostazioni e i file personali vengono infatti salvati nella home directory dell'utente e, dal momento che nei Live-CD la directory home **/home è solitamente memorizzata in un ramdrive** (ovvero un disco virtuale in memoria RAM), **ogni**



Salvando la vostra configrazione, al successivo riavvio il sistema avrà mantenuto tutte le impostazioni.

volta che il computer viene spento tutti i dati vanno perduti.
Tuttavia utilizzando Knoppix e un drive USB rimovibile potrete creare anche voi la vostra "**KNOPPIX Persistent Home**" e così, di computer in computer, di reboot in reboot, avere sempre a portata di mano i documenti personali, i bookmarks o gli script personalizzati.

>> Creare la Home permanente...

Per prima cosa dovrete disporre di una partizione dove poter creare l'immagine della vostra home directory; utilizzando una **chiave USB** la "portabilità" è massima ma potreste anche decidere di creare un immagine sul vostro **disco rigido**, a patto che l'unità sia stata formattata con un filesystem supportato da Linux sia in lettura che in scrittura (ad esempio ext2, ext3 e fat32 ma NON NTFS!). Connettete quindi il dispositivo ed avviate la macchina, **evitando però montare il dispositivo una volta caricato il sistema**. A questo punto dal menù principale '**KNOPPIX**' accedete al sottomenù '**Configure**' e cliccate infine su '**Create a persistent KNOPPIX Home directory**'; comparirà quindi un finestra di conferma da leggere con attenzione prima di cliccare su '**Yes**' e procedere quindi nella creazione della vostra Home permanente. A questo punto dovete selezionare la partizione in cui salvare i vostri dati (il dispositivo USB solitamente viene indicato come **/dev/sda1**) e decidere quindi se dedicare l'intera partizione alla home; in questo caso è consigliabile optare per il "**No**" e specificare manualmente la dimensione, lasciando magari dello spazio libero da utilizzare per lo scambio di dati tra i diversi sistemi operativi. Verrà quindi creato un file immagine della vostra home chiamato **knoppix.img** e potrete anche decidere se **cifrarlo o meno** (verrà utilizzando l'algoritmo AES con una chiave di 256bit); in caso affermativo, la frase che dovrete specificare (**di almeno 20 caratteri**) vi verrà richiesta nuovamente ogni qualvolta cercherete di montare al boot la vostra home personalizzata.

>> ...e montarla al boot

Una volta spento il sistema, estraete il vostro drive USB e tenetelo sempre a portata di mano. Indipendentemente dal PC su cui vi troverete a lavorare, vi basterà infatti aggiungere al boot il parametro di caricamento del kernel "**home=[nome del dispositivo]**" nel caso in cui sappiate quale sia il nome assegnatogli dal sistema (**/dev/sda1** nel nostro caso ad esempio) o, più semplicemente, "**home=scan**" per lasciare a Knoppix il compito di rintracciare la vostra /home.

Le chiavi di memoria USB vengono viste come normali dischi dal sistema operativo, e sono molto pratiche per spostare velocemente dati tra due postazioni diverse, o per tenere sempre con sé le informaizoni più importanti. I costi si stanno sempre più abbassando, e oggi si può comprare una chiave da 128 Mbyte con circa 60 euro.

Cheat Codes

Knoppix permette all'avvio di passare al kernel moltissimi parametri: in questo modo è possibile personalizzare il caricamento del sistema (specificando la lingua, la tastiera, il wm da utilizzare) e risolvere eventuali problemi nel riconoscimento dell'hardware installato, forzando ad esempio il caricamento di moduli specifici o evitando la ricerca di determinate tipologie di dispositivi. Per avere un'idea di quali siano le decine di opzioni possibili è sufficiente premere F2 al prompt iniziale che compare appena avviato il computer da CD-Rom; inoltre lo stesso elenco è presente sul CD all'interno della cartella Knoppix (knoppix-cheatcodes.txt) ed ogni parametro è spiegato in dettaglio da un apposito documento della "Knoppix Linux Documentation" (http://www.knoppix.net/docs/index.php/CheatCodes). Molto probabilmente per molti di voi sarà comunque sufficiente digitare al boot

```
knoppix lang=it
```

ma, chi volesse specificare anche la risoluzione o un Window Manager alternativo, potrebbe ad esempio aggiungere anche

```
screen=800x600 desktop=xfce
```

e così via... Infine ricordate che Knoppix utilizza come predefinita la tastiera americana; il simbolo di "uguale" è quindi riproducibile al boot premendo non SHIFT+0 bensì il tasto della "ì" (i accentata).

Con 'Save KNOPPIX configuration' e 'KNOPPIX Persistent Home" il vostro desktop vi seguirà ovunque.

Le figlie di Knoppix

Oltre che essere estremamente versatile e potente nel riconoscimento dell'hardware installato, Knoppix è liberamente modificabile poichè rilasciata sotto licenza GNU GPL ed è estremamente semplice da personalizzare; esistono persino appositi tool che permettono in pochi passi di crearne una versione contenente solo i pacchetti desiderati. Per questo motivo la tecnologia alla base di questo Live-Cd è stata utilizzata anche da altre distribuzioni (Yoper, Gentoo) per creare versioni live del proprio prodotto e molti altri utenti non hanno tardato a distribuire in Rete versioni modificate di Knoppix per gli scopi più disparati. Esistono infatti versioni localizzate in turco, ceco, giapponese, greco, spagnolo, ungherese e anche in Italiano (KnopILS - http://knopils.linux.it); Echelonlinux, L.A.S., Overklockix, Phlak o Knoppix STD sono state invece studiate per l'analisi e la manutenzione della propria rete e del computer. Accanto poi alle versioni ultra compatte (nell'ordine dei 50 Mb!) ma altrettanto funzionanti quali DamnSmallLinux, Flonux, gKnx e MicroKnoppixISO troviamo Gnoppix, il cui DE predefinito non è KDE bensì Gnome, la modulare Morphix e KnoppiXMAME, che dal nome non ha bisogno di ulteriori presentazioni. Per una lista completa ed aggiornata dei Live-CD derivati da Knoppix fate comunque riferimento alla pagina ufficiale http://www.knoppix.net/docs/index.php/.

>> E la configurazione?

Nello stesso menu di Configurazione di Knoppix è presente anche una voce "**Save KNOPPIX Configuration**"; altro non è che uno script del tutto simile a quello precedentemente visto che consente di **salvare tutti i dati della configurazione** dell'ambiente grafico, della rete e di altre impostazioni di sistema. Anche in questo caso vi verrà chiesto su quale dispositivo salvare il file di configurazione e, al successivo riavvio, basterà aggiungere come parametro di boot **myconfig=scan** per ritrovare il sistema tale e quale a come l'avevate lasciato. In particolare è possibile utilizzare un semplice floppy per questa operazione e utilizzare al boot il cheatcode "**floppyconfig**". Più semplice di così...

Lele

Il menu di configurazione di Knoppix offre tutti gli strumenti necessari per farvi sentire sempre come a casa.

FLASH E IDENTIFICAZIONI

Una accoppiata non molto sicura sul Web!

Flash è una tecnologia realizzata da Macromedia per la creazione di siti e filmati animati ampiamente utilizzati per il web. Le animazioni create in flash possono essere utilizzate nei più svariati modi, per esempio si può passare dalla realizzazione di un sito alla creazione di un gioco, grazie soprattutto agli script che possono essere inseriti nei filmati e grazie alla facile integrazione con i linguaggi di scripting dal lato server, tipo l'ASP.

>> Schermate di identificazione

I file in flash (.swf) presenti in molti siti sono dei file compilati, cioè una volta finalizzati per il web **non sarebbe più possibile** ritornare al file sorgente (riconoscibile invece dall'estensione .fla).
Ultimamente si stanno diffondendo in rete delle semplici schermate di identificazione create in flash per realizzare delle **sezioni protette**, non molto sicure e purtroppo facilmente superabili.
Queste schermate di identificazione vengono create integrando nei filmati degli script, in questo caso degli **script per verificare username e password...**
Una schermata di identificazione in flash è di questo tipo: **campo** in cui inserire username, **campo** in cui inserire password e infine un **tasto** per inviare i dati, che chiameremo Invia. Cliccando sul tasto Invia viene eseguita l'azione "**on click**", ossia "al click", inserendo in questa azione uno script è possibile facilmente realizzare una procedura di identificazione.
Il compito dello script sarà quello di **verificare una semplice condizione**: se l'username inserito è uguale, per esempio, a "x" e la password inserita è uguale a "x", i dati sono corretti e si viene identificati altrimenti viene mostrato un messaggio di errore.

>> I decompilatori

Come per i file exe (programmi), anche per i file swf (file flash compilati) esistono numerosi "**decompilatatori**" che permettono di **vedere come è fatto il filmato flash,** e di conseguenza osservare le azioni e gli script associati, per esempio, al tasto Invia.
Per questo motivo, ritengo poco sicure e facilmente superabili queste schermate di identificazione in flash. Basterà infatti **decompilare il filmato in flash**, precedentemente salvato sul proprio PC, utilizzando uno dei numerosissimi decompilatori presenti in rete, trovare le azioni e gli script del filmato e di conseguenza username e password per superare in un batter d'occhio la protezione.

Infine volevo precisare che le condizioni degli script sono di questo tipo:

```
IF condizione then
  (la condizione è vera)
else
  (altrimenti)
end if
```

Quindi **sconsiglio vivamente a tutti i siti** di usare queste schermate di identificazione in flash ma di affidarsi per esempio a dei linguaggi di programmazione per il web più sicuri, tipo l'ASP o il PHP!

Salvatore Aranzulla
mirabilweb@tiscali.it
http://www.mirabilweb.tk

IL PROGRAMMA "ACTION SCRIPT VIEWER"

Fra i migliori decompilatori flash, da ricordare è Action Script Viewer un programma che permette di analizzare le varie parti di un filmato flash, decompilandolo. Èpossibile scaricare da http://www.buraks.com/asv/1.html una versione demo di ASV, inoltre nello stesso sito troverete maggiori informazioni e dettagli su questo decompilatore flash.

Cancella le tracce

Se si va a pescare in quell'enorme barattolo di marmellata che è Internet,

Ogni volta che si visita un singolo sito Web, rimangono delle tracce in diverse posizioni del computer: Cronologia, URL digitati (che non sono la stessa cosa…), cookie, file temporanei, cache… Chiunque abbia accesso allo stesso computer, può facilmente **ricostruire le pagine che abbiamo visitato**, i **file** scaricati, persino le **parole chiave** inserite in moduli e motori di ricerca. Inoltre, spyware e siti Web senza scrupoli possono registrare sul nostro computer programmini e componenti fastidiosi. Ecco un bell'aresenale di strumenti gratuiti che possono essere molto utili per **cancellare le proprie tracce** e **riprendere il controllo** del PC.

Exposed!

www.reddragonsoftware.com

Windows 98/ME/2000/XP

Volete farvi un'idea di quali siano le informazioni che il browser memorizza automaticamente da qualche parte del computer? E magari eliminarle selettivamente, scegliendo cosa tenere? Exposed è l'ideale. Il programma mostra le anteprime di immagini, video, musica, animazioni Flash e molti altri elementi, suddivisi per categorie. La ricerca può essere effettuata anche solo per quei file registrati in un certo periodo di tempo. Ogni elemento può essere salvato altrove sul disco (anche organizzato in "collezioni"), oppure cancellato.

RapidBlaster Killer

www.wilderssecurity.net/specialinfo/rapidblaster.html

Windows 98/ME/2000/XP

RapidBlaster è un motore pubblicitario parassita che si installa come task di avvio di Windows e mostra banner scaricati da Internet. Si installa quasi come un virus, attraverso ActiveX diffusi da siti senza scrupoli, ed è abbastanza difficile da rimuovere manualmente, visto che cambia costantemente nome e posizione. Questo programma analizza i programmi in esecuzione e, una volta trovato RapidBlaster termina il processo e rimuove la chiave di registro e ogni file collegato.

AbsoluteShield Internet Eraser Lite

www.internet-track-eraser.com

98/ME/NT/2000/XP

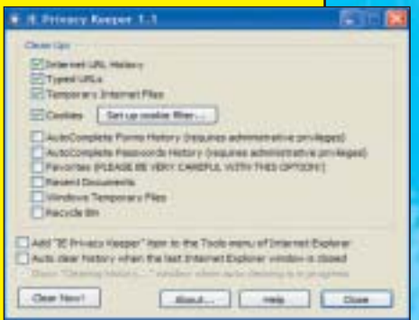
Questo programma protegge la privacy dell'utente rimuovendo le tracce delle navigazioni Internet e dell'utilizzo del computer. Può cancellare al volo la Cronologia del browser, la cache, i cookie, gli URL digitati, il cestino e le cartelle dei file temporanei.

È possibile usare un metodo di cancellazione sicura dei file, che riscrive varie volte il disco in modo da evitare ogni possibile recupero dei dati. Come optional, c'è un efficace blocco delle finestre popup. Del programma esiste una versione più avanzata, a pagamento, ma tutte le funzionalità citate sono disponibili in quella gratuita.

IE Privacy Keeper

www.unhsolutions.net/IEPK/

Windows 98/ME/2000/XP



Nell'esatto momento in cui chiuderete l'ultima finestra aperta di Internet Explorer, IE Privacy Keeper cancellerà tutte le tracce della vostra sessione Internet. Si può scegliere di cancellare la Cronologia, gli URL digitati, i File Temporanei Internet, i Cookie, i Documenti Recenti, la cronologia dell'Auto Completamento, i file temporanei di Windows e svuotare il Cestino. IE Privacy Keeper può lavorare in automatico, oppure manualmente dal menu Tools di Internet Explorer.

delle tue navigazioni

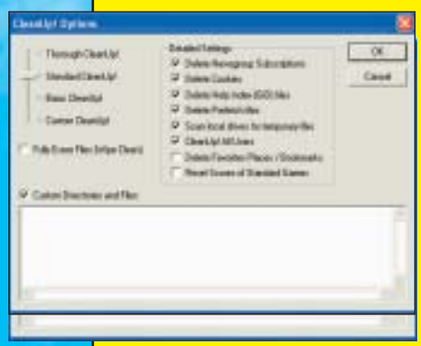
è difficile non lasciare ditate appiccicose sull'hard disk. Ecco come eliminarle...

Windows CleanUp!

http://cleanup.stevengould.org

98/ME/NT/2000/XP

Oltre a cancellare i dati sensibili che rimangono registrati sul computer dopo una sessione Internet, Windows CleanUp è molto utile per rimuovere parecchi di quei file spazzatura che si accumulano nel tempo sull'hard disk, aumentandone la frammentazione e riducendo lo spazio disponibile. Offre quattro livelli di intervento, l'ultimo dei quali è indicato per ripulire completamente un hard disk prima di vendere il computer (per esempio, rimuove anche i Preferiti). È anche possibile impostare alcune directory personalizzate, e ripulirle periodicamente o a un esplicito comando.

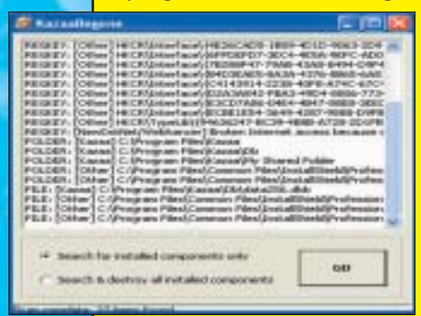


KazaaBegone

www.spywareinfo.com/~merijn/

Windows 98/ME/2000/XP

Il programma di file sharing Kazaa è noto per installare molti componenti indesiderati (Spyware, insomma), che non vengono rimossi automaticamente quando si disinstalla il programma. KazaaBegone si occupa proprio di questo, e rimuove ogni traccia dell'installazione di Kazaa, in ogni sua versione, compresi i software aggiuntivi che potrebbero non essere facilmente individuabili.

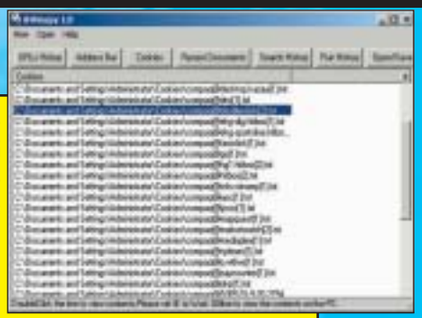


WinSpy

www.acesoft.net/winspy/index.html

98/ME/NT/2000/XP

A differenza dei programmi citati finora, WinSpy non serve ad eliminare i file che rimangono sul disco dopo una navigazione Internet. Al contrario, serve per rivelarli, utilizzando una pratica interfaccia. Si possono visualizzare gli URL inseriti, la Cronologia, la cache del browser, i cookie, i documenti aperti di recenti e altro ancora. Mi raccomando, non usatelo per violare la privacy di altre persone…



GooDelete History

www.pppindia.com/goodelete

Windows 98/ME/NT/2000/XP

La toolbar di Google è uno strumento utile per facilitare la ricerca su Internet attraverso il popolare motore di ricerca, ma apre le porte a un ulteriore rischio per la privacy. Memorizza infatti ogni ricerca effettuata, permettendo a chiunque abbia accesso al computer di vedere a quali argomenti ci siamo interessati. In effetti è possibile cancellare l'intera lista delle parole chiave cercate, ma in certi casi potremmo voler eliminare soltanto una o due voci, conservando quelle utili. GooDelete History serve proprio a questo.

Empty Temp Folders

www.danish-shareware.dk/soft/emptemp/

Windows 98/NT/ME/2000 e IE5

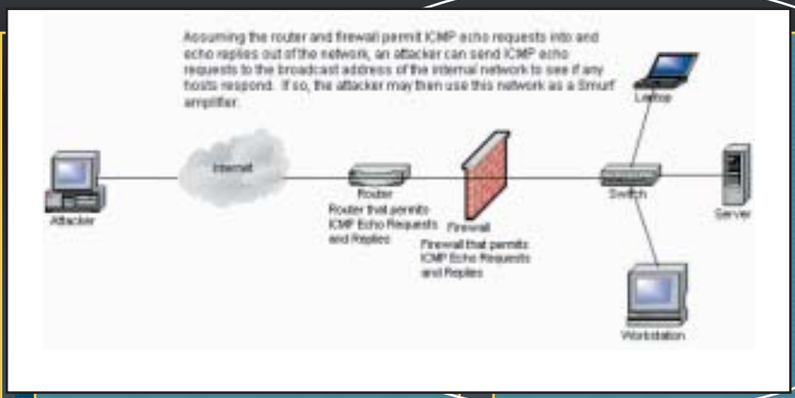
Questo è probabilmente uno degli strumenti più utili presentati in queste pagine. Come Windows CleanUp permette di assicurare la privacy eliminando le tracce della navigazione Internet e dei file aperti, e anche di cancellare i file temporanei ingombranti, ma le sue possibilità di personalizzazione lo rendono uno strumento ben più versatile. È infatti possibile impostare una lista di elementi che non devono essere cancellati. In questo modo si può ripulire l'archivio dei cookie ma conservare quelli di cui si fa effettivamente uso (come i cookie per il login automatico sul forum di Hacker Journal). L'interfaccia molto pulita e la facilità d'uso lo rendono adatto anche ai principianti, mentre gli utenti più smaliziati potranno usarne le funzionalità più avanzate.

NON C'ENTRANO I PUFFI!

Il piu famoso attacco DDOS prende il nome dai simpatici ometti blu noti al mondo come "smurfs" pur non imitandone la simpatia.

Un tempo l'attacco di **Denial of Service** era comunemente considerato l'ultima spiaggia di un aggressore ad un sistema informatico, quando tutti i tentativi d'accesso si erano esauriti. Il suo utilizzo, infatti, a parte **provocare il momentaneo disservizio del sistema attaccato**, non arrecava né danni che non si potessero riparare al limite con un semplice riavvio del sistema (mi riferisco ai vecchi sistemi NT), né vantaggi rilevanti per l'attaccante quali una shell o anche soltanto un nome utente. A causa però della relativa facilità con cui un attacco di questo tipo può essere messo in atto, **è diventato lo strumento principe dei cosiddetti script kiddies o lamer che dir si voglia**.

Tra i DDOS sono annoverate varie tipologie di attacco. Noi qui ci limiteremo a prendere in considerazione quelle che hanno come leit-motiv la **saturazione della banda a disposizione del sistema target**.



>> Premesse teoriche

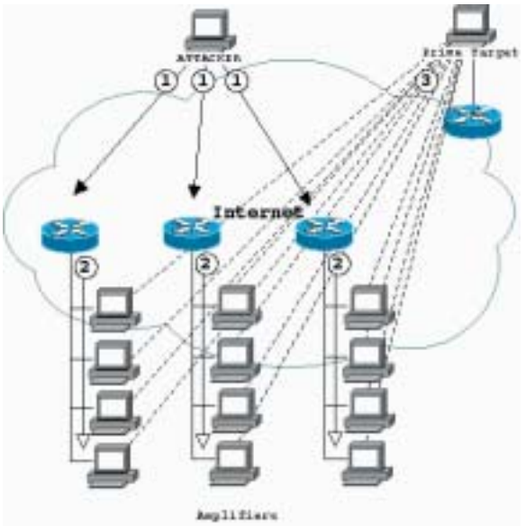
La premessa teorica che sta alla base degli attacchi a saturazione di banda è l'**amplificazione dei pacchetti**, ossia la possibilità di indirizzare verso la rete obiettivo un grosso quantitativo di dati, sfruttando una rete intermedia che ne amplifichi (ossia ne moltiplichi) la quantità. Per cui possiamo configurare uno scenario di questo tipo con la presenza di tre attori:

- **La rete target**
- **La rete intermedia**
- **La sorgente dei pacchetti**

Secondo il tipo di pacchetto sorgente utilizzato, ossia il pacchetto cha da avvio al processo, l'attacco prende nomi diversi: **smurf** quando si utilizzano pacchetti **ICMP**, **fraggle** con i pacchetti **UDP**.

>> SMURF

L'attacco smurf prende il suo nome dall'exploit in c che per primo ha sfruttato questa vulnerabilità del set di protocolli TCP/IP (e che potete trovare nella secret zone di HJ). Esso si basa sull'invio da parte dell'attaccante di un pacchetto



ICMP di **ECHO_REQUEST spoofato**, vale a dire costruito con un indirizzo di origine diverso da quello reale, **verso gli indirizzi broadcast di una rete che abbia una grossa disponibilità di banda**. I pacchetti ICMP ECHO_REQUEST sono utilizzati normalmente per il **ping,** cioè per la verifica della presenza in rete di un computer o dispositivo di rete, il quale **risponde al mittente con un pacchetto ICMP ECHO_REPLY**. Gli indirizzi di broadcast sono, all'interno di una sottorete, degli indirizzi dedicati non assegnabili a singoli dispositivi di rete, che sono utilizzati per l'invio simultaneo di messaggi a **tutte le macchine** appartenenti alla sottorete cui appartiene l'indirizzo di broadcast stesso. L'attaccante quindi può creare delle **ECHO_REQUEST fasulle** con un indirizzo sorgente uguale a quello del sistema che vuole colpire. A questo punto il router della rete "amplificatrice", reindirizza la richiesta di ping a **tutti i computer** appartenenti alla sottorete che rispondono all'unisono in direzione del computer vittima. Immaginate quindi cosa può accadere se un attaccante invia uno stream di pacchetti ICMP ECHO_REQUEST che generi un traffico di 768 kbps ad una rete con 100 computer: in breve **viene prodotto un traffico di circa 76,8 Mbps di pacchetti ICMP ECHO_REPLY verso la vittima**.

Capite bene, quindi, come si possa facilmente, magari con più azioni coordinate, saturare in breve tempo la banda anche di una grossa struttura come Yahoo o Ebay o Microsoft.

>> Fraggle

L'attacco fraggle si basa invece sull'utilizzo di **pacchetti UDP** indirizzati verso particolari porte che sono la **chargen** (porta **19**) che sta per character generator e la **echo** (porta **7**). Il primo servizio su UDP genera uno stream di caratteri che viene inviato verso il computer connesso, il secondo genera un echo cioè restituisce al mittente i pacchetti ricevuti. Spoffando anche in questo caso l'indirizzo ip sorgente è possibile creare un effetto molto simile a quello dello smurf con in più la possibilità di **generare un ping pong all'interno di una rete tra la porta chargen di una macchina e la porta echo di un'altra**. Questo tipo di attacco è meno temuto poiché i servizi che vengono sfruttati generalmente non servono molto e quindi **possono essere disabilitati** senza danneggiare il normale funzionamento della rete.

>> Difendersi dallo smurf

Diverso è il discorso per quanto riguarda lo smurf poiché **disabilitare il ping è quantomeno improponibile** visto l'importanza del servizio. Nonostante ciò **vi sono degli accorgimenti** che consentono di evitare che la propria rete reciti il ruolo di vittima, di attaccante o di rete amplificatrice.

>> Evitare di essere la fonte di un attacco

Per evitare che qualcuno dalla nostra rete effettui un attacco di smurf è sufficiente **filtrare i pacchetti in uscita** attraverso un firewall in modo che gli indirizzi sorgente dei pacchetti siano tutti provenienti dalla propria sottorete e che **vengano quindi scartati quelli spoofati**. A tal proposito la Cisco ha aggiunto tempo fa nel suo IOS (più o meno dalla versione 12.0 in poi) l'opzione di filtraggio "**ip verify unicast reverse-path**". Questa opzione fa in modo che il router, per ogni pacchetto ricevuto, effettui un confronto attraverso la tabella di routing tra l'indirizzo ip sorgente "dichiarato" sul pacchetto e il MAC address dell'interfaccia di rete da cui è stato ricevuto. Se il router nota una mancata corrispondenza, scarta il pacchetto.

>> Evitare di essere la rete amplificatrice

Anche in questo caso è necessario operare sui router per far fronte alla minaccia. È infatti necessario **evitare che i router inoltrino verso i computer della propria rete i pacchetti ICMP indirizzati all'ip di broadcast**. In base all'rfc 1812 "Requirements for IP version 4 Routers", ogni router **può disporre** di un opzione che disabiliti la ricezione di messaggi broadcast da una rete ester-

COME TRACCIARE STREAM DI PACCHETTI SPOOFFATI

Riuscire a risalire alla provenienza di un pacchetto con ip spooffato non è un'operazione semplice ma neanche impossibile. Alcune opzioni che si possono inserire nelle ACL dei dispositivi Cisco, che montano IOS dalla 12.0 in poi, consentono la registrazione dei log sia completi (opzione "log-input") sia limitati cioè senza MAC address e interfaccia di provenienza (opzione "log") con un minore utilizzo di risorse. Il metodo consiste nel verificare attraverso interrogazioni arp ai vari dispositivi se vi è una corrispondenza tra il MAC address dell'interfaccia di provenienza ed il suo indirizzo ip risalendo all'indietro attraverso i vari hop fino a giungere al MAC address finale e quindi all'ip dell'attaccante. Questo metodo, che per questioni di spazio non possiamo esporvi nel dettaglio in questa sede, lo trovate notevolmente approfondito e corredato da esempi pratici al seguente indirizzo: http://www.cymru.com/Documents/tracking-spoofed.html .
Questa tecnica è comunque fuori dalla portata del singolo utente però vi fa comprendere come una ricerca coordinata dalle forze dell'ordine può giungere all'origine dell'attacco.

na ma **DEVE disporre** di un opzione che disabiliti l'inoltro di questi messaggi verso le interfacce della rete interna. Attualmente, infatti, nella maggior parte dei router, anche in seguito all'ulteriore rfc 2644 di Daniel Senie, la configurazione di default disabilita l'inoltro di messaggi broadcast provenienti da reti esterne. In particolare nei sistemi Cisco con IOS dal 12.0 in poi il comando è il "**no ip directed-broadcast**". Per avere informazioni sulla configurazione relativa al vostro router vi consiglio di far riferimento alla documentazione presente sul sito del vendor.
Altra alternativa praticabile è di disabilitare sul singolo computer la risposta ad un ECHO_REQUEST diretta ad un indirizzo di broadcast. Questa soluzione però ha i suoi lati negativi giacché il ping broadcast **è anche uno strumento utilizzato da alcuni programmi di diagnostica**.

>> Evitare di essere vittima di un attacco

Evitare di cadere vittima di un attacco smurf è possibile. Si possono, infatti, **filtrare i pacchetti ECHO_REPLY** in ingresso sui dispositivi di confine della rete ossia i router. Questo però non impedisce che la banda venga saturata ma che il "packet storm" **non raggiunga i computer della LAN** compromettendone la comunicazione interna. Qualora l'obiettivo dell'attacco sia ad esempio un server web, questo tipo di approccio non impedisce un DDOS a chi cerca di collegarsi e di utilizzare le risorse del sito. A tal proposito, l'IOS di Cisco dalla versione 12.0 in poi prevede, oltre al semplice filtraggio del tipo di pacchetto incriminato, anche la **limitazione del traffico ad un certo numero di kbps attraverso il committed access rate (CAR)**. Questo consente di evitare di bloccare del tutto gli ECHO_REPLAY cosa che impedirebbe comunque il ping ai computer interni alla rete. Tutti questi accorgimenti però impediscono che i pacchetti vengano loggati ossia che il router ne registri la provenienza, che può essere utile per risalire all'origine dell'attacco.
Concludiamo qui il nostro articolo ripromettendoci di ritrattare l'argomento descrivendo altri attacchi DDOS nelle prossime uscite.

Roberto 'decOder' Enea
enea@hackerjournal.it

Link utili...

ftp://ftp.isi.edu/in-notes/rfc2267.txt
Documento di Paul Ferguson della Cisco Systems e di Daniel Senie di BlazeNet relativo ai metodi di protezione contro gli attacchi DDOS

ftp://ftp.rfc-editor.org/in-notes/rfc1812.txt
L'rfc 1812 "Requirements for IP version 4 Routers" di Baker della Cisco Systems

ftp://ftp.rfc-editor.org/in-notes/rfc2644.txt
L'rfc 2644 "Changing the Default for Directed Broadcasts in Routers" di Daniel Senie

ftp://ftp.rfc-editor.org/in-notes/rfc1122.txt
L'rfc 1122 "Requirements for Internet Hosts — Communication Layers" tratta i requisiti software degli host su internet
www.netscan.org e http://www.powertech.no/smurf/ questi siti forniscono un utile servizio che vi permette di verificare se la vostra rete è vulnerabile all'attacco smurf

AL MIO COMANDO

Programmazione della shell con GNU/Linux – Seconda parte

Nella prima puntata di questo mini-corso abbiamo trattato tutti gli elementi necessari per avere una conoscenza basilare della programmazione di shell in GNU/Linux. Se non l'avete letta, vi invito a farlo poiché la comprensione dei concetti che analizzeremo a breve potrebbe risultarvi difficoltosa.

>> Lo standard error

Lo **standard error** è un particolare tipo di output che viene utilizzato per i messaggi di errore. Nella puntata precedente avevamo analizzato invece gli operatori di ridirezione dell'input e dell'output, senza soffermarci sullo standard error. In questa sezione invece studieremo meglio tale standard.

L'operatore **'2>'** svolge una funzione simile a quella dell'operatore per lo standard output '**>**': esso memorizza i dati presenti nello standard error in un file. Analizziamo insieme questo esempio:

```
#!/bin/bash
# Salvo l'output secondo lo
standard error del comando
"XFree86 -version"
XFree86 -version 2>
xfree86_version.txt
```

Il comando "XFree86 -version" infatti mostra dei messaggi servendosi dello standard error. Noi li abbiamo salvati all'interno del file xfree86_version.txt, leggibile con un normale editor di testo.

È possibile ridirigere contemporaneamente lo standard error e quello ouput su un file usando l'operatore '**&>**'.

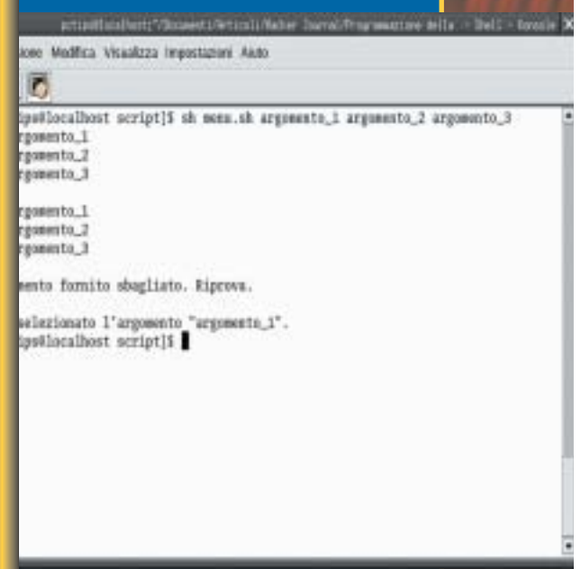
Esistono anche altri modi per effettuare la ridirezione: per maggiori informazioni leggete la pagina man di bash.

>> Liste di comandi

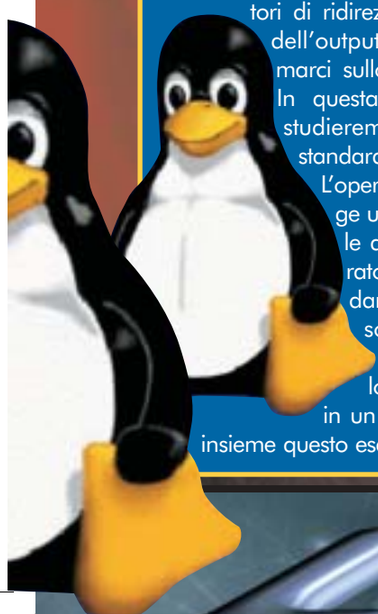
La shell bash permette all'utente di raggruppare in un'unica riga diversi comandi da eseguire. Tale riga prende il nome di **lista di comandi**.

Il modo più semplice per creare una lista è quello di separare ciascun comando dall'altro con un **punto e virgola**:

```
#!/bin/bash
# Lista di comandi separati
con l'uso del punto e vir-
gola
echo "Prima istruzione" ;
echo "Seconda istruzione" ;
echo "Terza istruzione"
```



Ogni lista termina con un **carattere di fine linea**, con **un punto e virgola** oppure con il simbolo '**&**' (che abilita l'esecuzione in background di un comando).



Le liste di comandi possono essere raggruppate anche all'interno di parentesi tonde oppure graffe.
I comandi tra **parentesi tonde** vanno separati come nel caso precedente e vengono eseguiti in una **subshell**. Per questo motivo assegnamenti di variabili e comandi interni non avranno effetto sul resto del programma. Ecco un esempio:

```
#!/bin/bash
# Lista di comandi racchiusa tra parentesi tonde
( echo "Prima istruzione" ; echo "Seconda istruzione" ) ; echo "Terza istruzione"
```

Come avrete potuto notare, è possibile utilizzare anche **liste multiple**.
Le liste di comandi racchiuse tra **parentesi graffe** vengono eseguite all'interno della shell principale e, a differenza di quelle tra parentesi tonde, assegnamenti di variabili ed esecuzioni di comandi interni avranno effetto su tutto il programma. Analizziamo insieme questo esempio:

```
#!/bin/bash
# Lista di comandi racchiusa tra parentesi graffe
{
echo "Prima istruzione"
echo "Seconda istruzione"
} ; echo "Terza istruzione"
```

Le parentesi graffe vanno espresse su **righe diverse rispetto alle istruzioni**.
Prima di concludere questa sezione, occorre fare una precisazione: il **valore di uscita** di una lista di comandi racchiusa tra parentesi è quello dell'**ultimo comando eseguito** al suo interno.
Secondo lo standard POSIX, ogni comando dopo aver svolto il suo compito emette un valore di uscita (o exit status): generalmente se esso è uguale a zero il comando è stato eseguito correttamente; in caso contrario no. Tale discorso si applica anche alle funzioni.

>> Operatori di controllo

Gli **operatori di controllo** svolgono la funzione di delimitatori all'interno delle liste di comandi proprio come il punto e virgola, ma, a differenza di quest'ultimo, leggono il valore di uscita del programma appena eseguito.
L'operatore di controllo '**&&**' esegue il comando successivo solo se il valore di uscita di quello precedente è zero (esecuzione corretta). Esso corrisponde all'operatore booleano AND. Analizziamo insieme questo esempio:

```
#!/bin/bash
# Verifico la presenza del comando bash
which bash && echo "Il comando bash esiste"
```

Questo programma verifica la presenza del comando bash e, in caso affermativo, mostra un messaggio sullo schermo.
L'operatore '**||**' svolge il compito opposto: esso esegue il comando successivo solo se il valore di uscita di quello precedente è diverso da zero (esecuzione fallita) e corrisponde all'operatore booleano OR. Ecco un esempio:

```
#!/bin/bash
# Verifico la presenza del comando bash3
which bash3 || echo "Il comando bash3 non esiste"
```

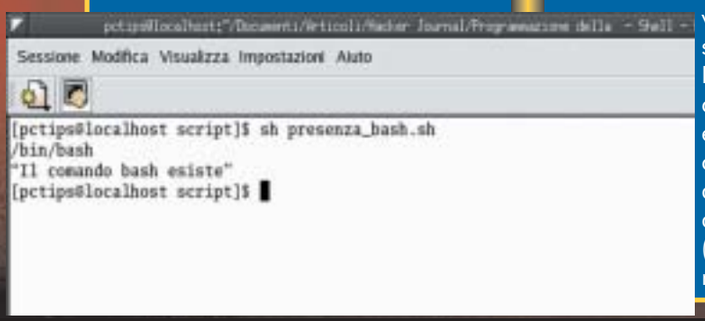
>> Parametri di shell

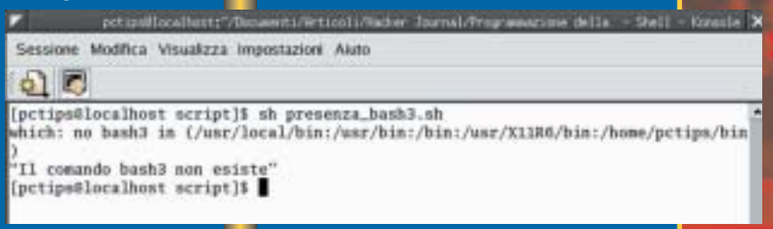
Un parametro è una variabile che rappresenta un elemeto particolare nell'attività della shell. Essi possono essere utilizzati all'interno degli script di shell come delle variabili normali, ma non possono essere sovrascritti.
Un **parametro posizionale** corrisponde ad un **argomento fornito alla shell** al momento dell'esecuzione di un programma. Il primo argomento specificato corrisponde alla variabile **1** (ma per maggiore chiarezza generalmente viene usata la dicitura **$1**), il secondo argomento alla variabile 2 e così via:

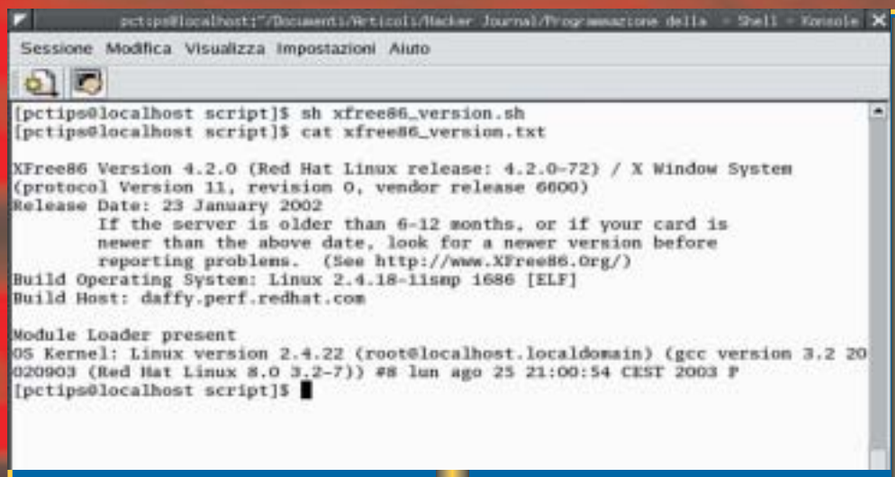
```
#!/bin/bash
# Mostro il primo argomento fornito al programma
echo -n "Il primo argomento fornito al programma è: $1"
```

Quando si utilizza un parametro a due o più cifre, quest'ultime vanno racchiuse tra **parentesi graffe** (Es. 10 => ${10}).
Esistono poi dei **parametri speciali** costituiti da uno zero o da un simbolo particolare. Essi sono:

• **$0** Equivale al nome della shell o dello script in esecuzione. Se uno script viene avviato digitando "bash script.sh", tale parametro corrisponderà al nome dello script. Se invece uno script viene avviato digitando "bash -c







script.sh", tale parametro corrisponderà al primo argomento eventualmente fornito al programma in questione.

- **$*** Equivale all'insieme di tutti i parametri posizionali a partire dal primo. Se usato tra apici doppi, tale parametro corrisponderà ad un'unica parola comprendente tutti i parametri posizionali specificati separati dal primo carattere (generalmente uno spazio) contenuto nella variabile speciale IFS, che analizzeremo nella prossima puntata.
- **$@** Svolge una funzione analoga al precedente, ma se racchiuso tra doppi apici genera una serie di parole, ciascuna composta dal contenuto del rispettivo parametro posizionale.
- **$#** Restituisce il numero di parametri posizionali specificati.
- **$?** Restituisce il valore di uscita dell'ultimo comando eseguito in foreground.
- **$-** Restituisce una serie di lettere che costituisce l'insieme delle modalità configurabili con il comando interno set (digita "help set" per maggiori informazioni).
- **$$** Restituisce il PID della shell. Se usato in una subshell, restituisce il PID della shell principale.
- **$!** Restituisce il PID dell'ultimo comando eseguito in background.
- **$_** Equivale all'ultimo argomento del comando precedente.

Imparare tutti i parametri speciali in una sola volta è inutile: a mio avviso è preferibile sperimentarli uno per uno direttamente sul campo.

>> Il costrutto select

Questo costrutto permette all'utente di effettuare una scelta attraverso la digitazione di un valore sulla tastiera. Può essere usato da un programmatore per realizzare dei menu. Ecco un esempio:

```
#!/bin/bash
# Genero un menu le cui
voci sono gli argomenti che
l'utente mi ha fornito
if [ $# = "0" ]
then
echo "Nessun argomento for-
nito."
else
select SCELTA in $@
do
if [ ! $SCELTA ]
then
echo "Argomento fornito
sbagliato. Riprova."
else
echo "Hai selezionato l'ar-
gomento \"$SCELTA\"." &&
break
fi
done
fi
```

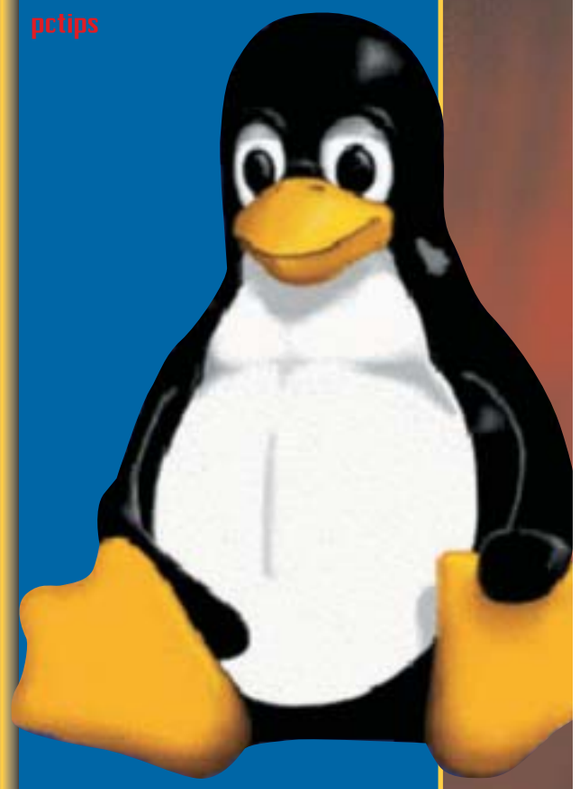
La prima istruzione if serve a verificare se l'utente ha fornito dei parametri al programma. In caso affermativo, viene creato un menu attraverso **select**. La parola **in** indica alla shell il **campo di azione** per definire gli elementi del menu: esso può essere costituita da una o più **variabili**. Se **in** è **assente**, allora si fa riferimento all'insieme di tutti i parametri posizionali (ovvero il parametro speciale '**$**@').

Dopo aver effettuato una selezione, la shell esegue le istruzioni comprese tra **do** e **done**. Il costrutto select è un'**istruzione iterativa** ed esegue ciclicamente le stesse operazioni fino a quando non viene eseguito **break** oppure **return**. Nella prossima puntata analizzeremo le altre istruzioni iterative disponibili.

>> Conclusione

Nella seconda puntata di questo minicorso abbiamo conosciuto nuovi costrutti e nuove peculiarità della shell bash che già ora ci consentono di scrivere i nostri script personali. Nella prossima puntata analizzeremo le variabili di shell, il calcolo aritmetico e le istruzioni iterative for, while ed until. Nel frattempo, provate a scrivere qualche piccolo programma adoperando i concetti che avete acquisito in questo minicorso. Buona fortuna :)

pctips



L'ACCESSO AI FILE

Per un programma, l'accesso ai file registrati su disco è una caratteristica fondamentale per rendere più pratiche l'acuisizione e la produzione dei dati.

Nell'articolo precedente abbiamo esaminato gli output a video e gli input tramite tastiera. In questo articolo ci concentreremo sull'**utilizzo dei file** sia come fonte di **input**, come risultato di una elaborazione e quindi **output**.
Un esempio su tutti: una volta che si è scritto un programma in un qualsiasi linguaggio (ma di quelli che necessitano una compilazione!) e si è salvato sotto forma di file, questo sarà il **file di input** per il compilatore, che una volta assolto il suo compito fornirà come risultato di uscita il file compilato (eseguibile), che è **l'output** del compilatore.
In generale, avremo la necessità di utilizzare un file (o più di uno) come input **quando il numero di dati è elevato**; pensate ad un database, oppure ad una serie di valori numerici. Sarebbe impensabile che ogni volta un utente immetta tali dati da tastiera questo perché: costituirebbe un inutile perdita di tempo ed inoltre aumenta vertiginosamente la probabilità di errore nella digitazione, soprattutto se i dati inseriti sono di tipo numerico.
Appare invece senz'altro più scontato il fatto che il risultato della nostra elaborazione (output) sia **memorizzato ad esempio su disco sotto forma di file**, proprio per poi essere riutilizzato e rielaborato in futuro, magari proprio come input!

>> Fasi dell'accesso ai file

In tutti i linguaggi possiamo schematizzare la gestione dei file nella seguente maniera:
1) Definiamo il **nome del file** (ossia attuiamo una procedura di assegnazione).
2) Predisponiamo l'**apertura del file** ed esplicitiamo la finalità per cui il file deve essere aperto; tipicamente si può aprire un file in **lettura**, **scrittura** o nella modalità **append** (aggiunta). La modalità apertura di un file in scrittura, apre il file e qualora il file non esiste lo crea; ma attenzione che se invece il file esiste, viene **sovrascritto e quindi tutto il suo contenuto viene cancellato!** Invece la modalità "append", di apertura di un file permette di aggiungere dati ad un file già esistente senza sovrascriverlo e quindi senza incorrere nella perdita di dati. Tuttavia la modalità append (in generale) necessita che il file a cui si devono aggiungere i dati in coda, sia un file esistente; ossia mentre l'opzione "scrittura" crea un file da zero, "append", in tal caso, genera un errore perché non gestisce la creazione del file.
3) **Prelievo** (se si è scelta la modalità lettura) o **inserimento** (se si è scelta la modalità scrittura o append) dei dati su file.
4) **Chiusura del file** ed effettiva memorizzazione dello stesso su disco.

MODALITA' APERTURA FILE

	Linguaggio C	Pascal	Visual Basic
	"r" "r+"	reset	input
	"w" "w+"	rewrite	output
	"a" "a+"	append	append

COMANDI GESTIONE FILE

	Linguaggio C	Pascal	Visual Basic
	fopen ( ) fclose ( )	assign ( ) close ( )	Open Close #
	fscanf ( )	readln ( ) read ( )	Input #
	fprintf ( )	writeln ( ) write ( )	Print # Write #

Vediamo alcuni semplici esempi di gestione di file nei 3 linguaggi che abbiamo preso come punto di riferimento:

>> Linguaggio C

Si voglia realizzare un programma che stampi a schermo i dati numerici (interi) contenuti in un file di nome "**readC.txt**", e li memorizzi in un altro file di nome "**writeC.txt**".
Di seguito è riportata una codifica in linguaggio C:

```
C1    #include <stdio.h>
C2      #define filelettura "readC.txt"
C3      #define filescrittura "writeC.txt"
C4      main ()
C5      {int dato;
C6           FILE *puntatore1,*puntatore2;
C7           puntatore1 = fopen(filelettura,"r");
C8         puntatore2 = fopen(filescrittura,"w");
C9         while (!feof(puntatore1)){
C10               fscanf(puntatore1,"%i",&dato);
C11               printf ("%i\n",dato);
C12              fprintf
(puntatore2,"%i\n",dato);}
C13       fclose(puntatore1);
C14       fclose(puntatore2);}
```

Naturalmente i caratteri C1… C14 non fanno parte del codice da compilare, ma servono in questa sede per commentare il codice:
C1 — C3 intestazioni (direttive al preprocessore); viene inclusa la libreria standard per la gestione degli input e degli output (**C1**), associamo (**#define**) al file "**readC.txt**" il nome simbolico di "**filelettura**" (**C2**) e procediamo in maniera analoga per il file "**writeC.txt**" (**C3**).
C4 funzione **main ( )** che deve essere contenuta all'interno di ogni programma in linguaggio C; il corpo di tale funzione è delimitato da parentesi graffe.
C5 dichiariamo la variabile dato come una variabile intera; all'interno di questa variabile scriveremo i dati prelevati dal "**filelettura**".
C6 vengono dichiarati in questo modo (**FILE *nome_puntatore**) 2 puntatori che serviranno nella gestione dei file; ponete attenzione nello scrivere FILE a caratteri maiuscoli e ad anteporre un asterisco (*) al nome del puntatore.
C7 al puntatore di nome "**puntatore1**" associamo il file di sola lettura (infatti è specificato "**r**") di nome "**filelettura**".
C8 analogamente al puntatore di nome "**puntatore2**" associamo il file aperto in scrittura (infatti è specificato "**w**") di nome "**filescrittura**". Il linguaggio C è molto elastico; si può aprire un file anche in modalità aggiunta (append) specificando "**a**", oppure aprire un file in scrittura/lettura con "**r+**", creare un file per scrittura/lettura con "**w+**" ed anche aggiungere dati o creare un file per scrittura/lettura "**a+**".
C9 — C12 ciclo che viene eseguito finchè il "puntatore1" non punta alla fine del file (**C9**); leggiamo da file attraverso il comando **fscanf( )** quello puntato dal "**puntatore1**" e lo memorizziamo nella variabile dato (**C10**). Quindi stampiamo a schermo il contenuto della variabile dato (**C11**), e infine scriviamo sul file gestito dal "**puntatore2**", quello contenuto all'interno della variabile dato (**C12**).
C13 — C14 chiudiamo i 2 file che sono stati manipolati.

>> Pascal

Il programma che vogliamo realizzare deve permettere all'utente di inserire da tastiera una serie di righe di testo (stringhe). Quando l'utente digita la stringa "**FINE**" significa che è terminata l'immissione del testo e si può procedere alla sua memorizzazione sotto forma di file; naturalmente la stringa "FINE" non dovrà apparire all'interno del file memorizzato in quanto non fa parte del testo ma è stata utilizzata solo come espediente per indicare il termine dell'immissione da tastiera.
Un esempio di codice potrebbe essere il seguente:

```
C1 var file1:text;
C2 var a: string;
C3 begin
C4         assign (file1,'c:\pascal.txt');
C5         rewrite (file1);
C6         repeat
C7                readln (a);
C8                if a <> 'FINE' then writeln
(file1,a);
C9         until a = 'FINE';
C10         close (file1);
C11 end.
```

C1 - C2 costituiscono la parte dichiarativa del programma; abbiamo definito una variabile di nome "file1" di tipo testo e la generica variabile "a" come una stringa.
C3 e C11 indicano rispettivamente l'inizio e la fine del corpo programma in pascal.
C4 è il comando di assegnazione, associamo al file di nome "file1" il suo vero nome e percorso di memorizzazione "C:\pascal.txt" attraverso il comando **assign (nomefile, 'percorso_di_memorizzazione');**
C5 con il comando **rewrite (nomefile)** stiamo precisando che il file deve essere aperto in scrittura (se esiste viene sovrascritto!); qualora avessimo voluto utilizzare il file in lettura avremmo dovuto usare il comando **reset (nomefile)**, se invece avessimo voluto utilizzare il file per aggiungere dei dati in coda allo stesso il comando da utilizzare sarebbe stato **append (nomefile)**.

C6 - C9 rappresenta il ciclo che viene ripetuto fino a quando la stringa immessa non coincide con la parola "FINE".
C7 viene richiesta l'immissione da tastiera di una stringa
C8 condizione che mi permette di memorizzare sul file tutte le stringhe digitate tranne la stringa "FINE". Si noti che per attuare la scrittura su file si è usato semplicemente il comando **writeln** (si poteva anche usare write) come se fosse un normale output su schermo; in realtà il comando **writeln** ha questa volta bisogno di 2 parametri secondo la sintassi:
writeln (nomefile, variabile_da_scrivere); quindi con l'esempio il contenuto della stringa di nome "a" viene memorizzato all'interno del file di nome "file1". Naturalmente qualora avessimo dichiarato il file aperto per la lettura il comando **writeln** (o write) sarebbe stato sostituito con l'analogo per la lettura **readln** (o read).
C10 infine possiamo quindi chiudere il file attraverso il comando **close (nomefile);**
Si noti che riguardo alla gestione dei file passando da un compilatore ad un altro ci potrebbe essere una leggera variazione della sintassi.
Un altro comando che potrebbe essere utile (ma ce ne sono anche molti altri) è il comando per cancellare un file: **erase (nomefile)**; naturalmente per non generare errori il file deve esistere e non deve essere aperto.

>> Visual basic

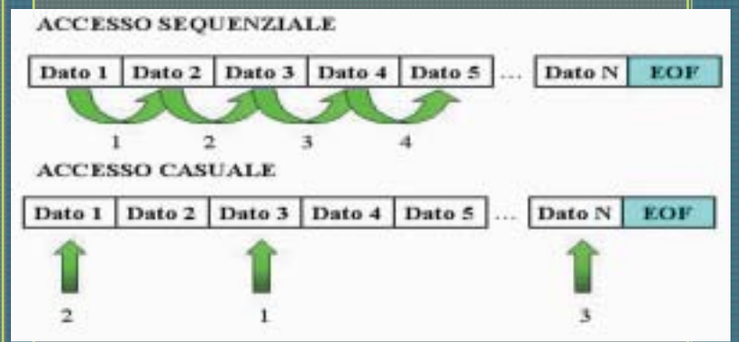
Questa volta si voglia realizzare un programma che trasferisca i dati contenuti in un file di nome "**file1.txt**" in un file di nome "**file2.txt**" il quale all'inizio (prima del trasferimento dei dati) e alla fine (dopo il trasferimento dei dati) presenti una stringa di delimitazione, costituita da una serie di trattini (**-**); inoltre sia alla fine cancellato il file sorgente (**file1**).
Un esempio di codice potrebbe essere il seguente:

```
C1   Dim riga As String
C2   Open "c:\file1.txt" For Input As #1
C3   Open "c:\file2.txt" For Output As #2
C4     Print #2, "-----------"
C5     Do While Not EOF(1)
C6         Input #1, riga
C7         Print #2, riga
C8     Loop
C9     Close #1
C10    Print #2, "-----------"
C11    Close #2
C12    Kill ("c:\file1.txt")
```

Vediamo in dettaglio le varie istruzioni:
C1 dichiariamo la variabile stringa di nome "riga" che servirà a memorizzare i dati letti dal file di nome "file1.txt".
C2 apriamo il file "file1.txt" in modalità di lettura (**input**) al quale viene associato il numero #1 come identificativo del file.
C3 apriamo il file "file2.txt" in modalità di scrittura (**output**) e gli associamo il numero #2; se avessimo voluto aprire un file per aggiungere dei dati, avremmo dovuto aprirlo con la modalità **append** (NOTA: nel visual basic se non esiste il file viene creato, e quindi non si genera alcun errore).
C4 e C10 attraverso il comando **print #numero_file** possiamo scrivere dei dati all'interno del file; in questo esempio scriviamo le righe di delimitazione. Per scrivere su di un file, in visual basic si potrebbe usare anche il comando **write #numero_file**; l'unica differenza con il comando print, è che il comando write separa i dati con delle virgole.
C5 — C8 è un ciclo che impone che finchè il "file1" non sia terminato (Not EOF(1)/ EOF sta per "end of file"), preleviamo i dati dal file1 attraverso l'istruzione **input #numero_file** (riga **C6**) li memorizziamo all'interno della variabile stringa di nome "riga" e successivamente li scriviamo sul "file2" attraverso l'istruzione print (riga **C7**).
C9 e C11 attraverso il comando **close #numero_file** chiudiamo i 2 file sui quali abbiamo operato.
C12 con il comando **kill (percorso_nome_file)** cancelliamo il file; è una sorta di comando "delete" (del) all'interno del visual basic.

>> Conclusione

Abbiamo preso in esame, nei tre linguaggi di riferimento, gli elementi base per maneggiare i file, ossia: la scrittura e la lettura di file ad accesso sequenziale (stream) di tipo testo, in cui i dati sono visti come una sequenza ordinata e quindi non si può procedere a sbalzi nel prelevare i dati e nello scriverli; tale operazione è invece consentita nei file ad accesso casuale (random). Naturalmente i file ad accesso sequenziale hanno una gestione più semplice, ma nel contempo sono ovviamente più limitati.



>> Nel prossimo articolo ...

Nel prossimo articolo prenderemo in esame le **funzioni** e le **procedure,** e vedremo come rendano più veloce, elegante, ordinata ed efficiente la programmazione.

>>--Robin-->
RobinHood.Sherwood@libero.it

SHELLCODING

Come sfruttare un programma già attivo per far eseguire al computer operazioni non previste (e non autorizzate...).

Uno **shellcode** non è altro che un insieme di **istruzioni espresse in modo tale da poter essere attuate mentre un altro programma è già in corso** (OPCODE).
Varie esemplificazioni di come uno shellcode possa essere inserito a tempo di esecuzione in una applicazione, ci sono fornite dalla maggior parte degli exploit attualmente in circolazione.
Affinchè una vulnerabilità possa essere sfruttata, (sia essa un "buffer/heap overflow", un "format-string bug" o quant'altro), è indispensabile **iniettare uno shellcode,** proprio perché non si sta facendo altro che prendere il controllo del sistema attraverso un'applicazione già in esecuzione.
Lo scopo di questo articolo esula dall'esplorare tutte le possibili tecniche adottate nel corso del tempo per iniettare uno shellcode e renderne possibile l'esecuzione; ci limiteremo infatti a **carpire i concetti essenziali che ne caratterizzano l'arte**.

I REGISTRI

Prima di passare al codice assembler e di seguito a quello binario, è necessario dare un rapido sguardo ai registri del processore per capirne il ruolo nel linguaggio assembler.
L'architettura presa in considerazione è **Intel-x86**, montante **sistema operativo Linux**.
Tutti i registri della piattaforma Intel sono a 32 bit, scindibili in sotto sezioni da 16 ed 8 bit, al fine di permettere uno sfruttamento euristico della memoria (vedere tabella).

EAX, AX, AH, AL sono detti registri accumulatore e possono essere usati per operazioni aritmetiche o riguardanti l'Input/Output o chiamate di Interrupt ecc. Vedremo infatti come è possibile utilizzarli per effettuare delle chiamate di sistema (system calls).

EBX, BX, BH, BL sono detti registri base e sono utilizzati come puntatori base per l'accesso alla memoria. Utilizzeremo questi registri per passare gli argomenti delle chiamate di sistema. Questi, sono a volte usati anche per memorizzare il valore di ritorno di un interrupt.
(Quando facciamo uso di una open(), il valore del descrittore del file è memorizzato nel registro EBX).

ECX, CX, CH, CL sono detti registri contatore.

EDX, DX, DH, DL sono i registri per i dati e possono essere impiegati per operazioni aritmetiche, chiamate di interrupt e per alcune operazioni di Input/Output.

I REGISTRI DELL'ARCHITETTURA INTEL-X86

32 bit	16 bit	8 bit (parte alta)	8 bit (parte bassa)
EAX	AX	AH	AL
EBX	BX	BH	BL
ECX	CX	CH	CL
EDX	DX	DH	DL

>> Semplici Istruzioni Assembler

L'assembler a cui ci accostiamo, adotta la sintassi **AT&T** ed è denominato **"Inline Assembly"**.
Il nome dei registri è preceduto dal simbolo '**%**', per cui se deve essere utilizzato il registro eax, è necessario scrivere **%eax**.
Se si fa riferimento a costanti numeriche si fa precedere il valore dal simbolo '**$**'.
Di seguito sono riportate le istruzioni più utilizzate nel linguaggio assembler, indispensabili per realizzare il nostro primo shellcode.

MOV Questa istruzione ci consente di spostare un valore in un registro, o perfino il contenuto di un registro presso un altro.

```
mov $0x4, %al sposta il valore 0x4 nel registro al
mov %eax, %ebx sposta il contenuto di eax in ebx
```

PUSH Pone un valore nello stack.
POP Preleva un valore dallo stack e lo memorizza in un registro o in una variabile.
INT Invoca un interrupt.

```
int $0x80 consente di passare il controllo al kernel.
```

>> Fase di Codifica

L'algoritmo da implementare in assembler e di seguito in codice binario (in versione esadecimale), è una semplice stampa a video della stringa "**WWW.ROSIELLO.ORG**".
L'algoritmo risolutivo in C è il seguente:

```
int main()
{
write(0, "WWW.ROSIELLO.ORG", 16);
      exit(0);
}
```

Per effettuare una **write()** ed una **exit()**, dobbiamo effettuare le relative **sys-call**.
È possibile trovare in Linux la libreria "**unistd.h**" in cui sono contenute tutte le chiamate di sistema.

```
angelo@rosiello.org$ cat /usr/include/asm-
i386/unistd.h
/*
 * This file contains the system call numbers.
 */
#define __NR_exit    1      _ Ecco la nostra exit()
#define __NR_fork    2
#define __NR_read    3
#define __NR_write   4      _ Ecco la nostra write()
#define __NR_open    5
……….
……….
write(0, "WWW.ROSIELLO.ORG", 16);
```

Il primo argomento "**0**" non è altro che lo **Standard Output** (video) al fine di stampare a video la stringa che compare come secondo argomento. L'ultimo parametro "**16**" indica la lunghezza della stringa.
Cerchiamo di realizzare tutto questo in assembler.

```
xor %eax, %eax
```

Pulisce il registro %eax

```
xor  %ebx, %ebx
xor  %edx, %edx

push %eax
```

Inserisce **NULL** nello stack, terminando la stringa. In modo tale che non appaiano caratteri indesiderati.

```
push $0x47524f2e #push GRO. sullo stack
push $0x4f4c4c45 #push OLLE sullo stack
push $0x49534f52 #push ISOR sullo stack
push $0x2e575757 #push .WWW sullo stack
```

Le quattro push su riportate inseriscono la stringa "**WWW.ROSIELLO.ORG**" nello stack secondo la codifica esadecimale.
Come è possibile notare, a causa della sua politica di funzionamento, è necessario porre nello stack la stringa **completamente rovesciata**.
Il descrittore che indica lo **Standard Output** è associato al registro **%ebx** che contiene al momento il valore **0**, quindi non dobbiamo indicare altro **(write(0, ...))**.

```
mov %esp, %ecx #sposta il contenuto di esp in ecx
```

A questo punto, l'indirizzo della stringa si trova nel registro **%esp** (ricordiamo che **esp** viene incrementato/decrementato solo mediante delle **pop/push**) e ne poniamo il contenuto nel registro **%ecx**. Questo consentirà al processore di ritrovare la posizione della stringa nello stack **(write(0, stringa, ...))**.

```
mov $0x10,%dl #size 16 bytes
```

Esattamente come in C indichiamo che la stringa ha dimensione 16 byte **(write(0, stringa, 16))**.

```
mov $0x4,%al  #syscall per write
```

Poniamo nel registro **eax** (nella parte inferiore **al**) il codice della **write**.

```
int $0x80    #esegui la syscall
```

Passiamo mediante questo interrupt il controllo al kernel, che si occuperà dell'esecuzione della write.

L'implementazione di exit(0) è ancora più semplice.

```
exit(0);

xor     %eax, %eax
xor     %ebx, %ebx
```

I registri eax ed ebx sono puliti.

```
mov     $0x1, %al    #syscall per exit
```

Poniamo il valore della exit in al.

```
int     $0x80     #esegui syscall
```

Passiamo il controllo al kernel.

>> Compilazione ed Esecuzione

Dopo aver realizzato il programma in assembler, bisogna compilarlo. Il compilatore che viene utilizzato è **Gcc**. La comodità dello **"inline assembly"** è proprio quella di poter essere compilato via Gcc.
Tutto quello che occorre fare è scrivere il programma assembler secondo la seguente metodologia.

```
angelo@rosiello.org:~/shellcode$ cat rosiello.c
#include <stdio.h>
int main(){
__asm__("           // notazione inline
// inseriamo il codice assembler
        xor     %eax,%eax
        xor     %ebx,%ebx
        xor     %edx,%edx
        push    %eax
        push    $0x47524f2e
        push    $0x4f4c4c45
        push    $0x49534f52
        push    $0x2e575757
        // %ebx = file_descriptor => 0 = stdout
        mov     %esp,%ecx
        mov     $0x10,%dl
        mov     $0x4,%al
        int     $0x80

        // exit(0)
        xor     %eax,%eax
        xor     %ebx,%ebx
        mov     $0x1,%al      #syscall for exit
        int     $0x80         #execute the syscall
// fine del codice assembler
");                 // notazione inline
}


angelo@rosiello.org:~/shellcode$ gcc rosiello.c -o
rosiello
angelo@rosiello.org:~/shellcode$ ./rosiello
WWW.ROSIELLO.ORG    _ stringa a video
```

>> Codifica binaria

L'ultimo passo da compiere è la codifica in codice binario. Per il nostro scopo utilizzeremo il debugger **gdb**.

```
angelo@rosiello.org:~/shellcode$ gdb rosiello
(gdb) disas main
Dump of assembler code for function main:
0x80482f4 <main>:        push   %ebp
0x80482f5 <main+1>:      mov    %esp,%ebp
0x80482f7 <main+3>:      sub    $0x8,%esp
0x80482fa <main+6>:      and    $0xfffffff0,%esp
0x80482fd <main+9>:      mov    $0x0,%eax
0x8048302 <main+14>:     sub    %eax,%esp
0x8048304 <main+16>:     xor    %eax,%eax
0x8048306 <main+18>:     xor    %ebx,%ebx
0x8048308 <main+20>:     xor    %edx,%edx
0x804830a <main+22>:     push   %eax
0x804830b <main+23>:     push   $0x47524f2e
0x8048310 <main+28>:     push   $0x4f4c4c45
0x8048315 <main+33>:     push   $0x49534f52
0x804831a <main+38>:     push   $0x2e575757
0x804831f <main+43>:     mov    %esp,%ecx
0x8048321 <main+45>:     mov    $0x10,%dl
0x8048323 <main+47>:     mov    $0x4,%al
0x8048325 <main+49>:     int    $0x80
0x8048327 <main+51>:     xor    %eax,%eax
0x8048329 <main+53>:     xor    %ebx,%ebx
0x804832b <main+55>:     mov    $0x1,%al
0x804832d <main+57>:     int    $0x80
End of assembler dump.
```

Il nostro codice comincia da **<main+16>** e termina a **<main+57>**.
Per ottenere l'opcode basta adottare la seguente tecnica.

```
(gdb) x/bx main+16
0x8048304 <main+16>:     0x31    _OPCODE
(gdb)
0x8048305 <main+17>:     0xc0    _OPCODE
(gdb)
0x8048306 <main+18>:     0x31    _OPCODE
…..
```

battendo invio fino a <main+57>.
A questo punto è necessario porre tutto secondo questa forma:

```
"\x31\xc0\x31…..".
"\x31\xc0\x31\xdb\x31\xd2\x50\x68\x2e\x4f"
"\x52\x47\x68\x45\x4c\x4c\x4f\x68\x52\x4f"
"\x53\x49\x68\x57\x57\x57\x2e\x89\xe1\xb2"
"\x10\xb0\x04\xcd\x80\x31\xc0\x31\xdb\xb0"
"\x01\xcd\x80"
```

Per potere compilare ed eseguire il nostro shellcode, basta strutturare il programma C secondo semplici canoni.

```
angelo@rosiello.org:~/shellcode$ cat shellcode.c
#include <stdio.h>
char shellcode[]=
"\x31\xc0\x31\xdb\x31\xd2\x50\x68\x2e\x4f"
"\x52\x47\x68\x45\x4c\x4c\x4f\x68\x52\x4f"
"\x53\x49\x68\x57\x57\x57\x2e\x89\xe1\xb2"
"\x10\xb0\x04\xcd\x80\x31\xc0\x31\xdb\xb0"
"\x01\xcd\x80";
main()
{
      void (*routine) ();
(long) routine = &shellcode;
printf("Size: %d bytes.\n", sizeof(shellcode));
routine();
}


angelo@rosiello.org:~/shellcode$ gcc shellcode.c -o
shellcode
angelo@rosiello.org:~/shellcode$ ./shellcode
Size: 44 bytes.
WWW.ROSIELLO.ORG
```

>> Conclusioni

Realizzare uno shellcode è **relativamente semplice**, basta un po' di pazienza e di pratica.
La sua importanza in alcune applicazioni di basso livello è di cruciale importanza, basti pensare alla realizzazione di un exploit mirante ad **evidenziare la vulnerabilità** di un programma.
L'intenzione era quella di fare un po' di luce su questo argomento, che molto spesso viene trascurato, con la speranza di non annoiare il lettore bensì di invogliarlo ad approfondire gli aspetti che, non per negligenza, ma per questioni di "spazio" sono stati solo brevemente accennati.

Angelo Rosiello-angelo@rosiello.org-http://www.rosiello.org

Tornano gli ABBONAMENTI!

Abbonati a

HACKER JOURNAL

25 numeri della rivista + il mitico CAPPELLINO HJ con ricamato il nostro logo, al prezzo di € 50,00

Dopo un periodo di pausa, tornano alla grande i servizi di abbonamento e arretrati. La gestione non sarà effettuata dalla redazione, ma da una struttura esterna, che accetterà pagamenti in conto corrente postale o via carta di credito. Per informazioni, bisogna contattare la Staff srl ai seguenti recapiti:

Tel. 02/45702415 (dal Lunedì al Venerdì, ore 9.30/12.30 - 14.30/17.30)
Fax 02/45702434
abbonamenti@staffonline.biz

Potete trovare i moduli da compilare e tutte le istruzioni all'indirizzo:
www.hackerjournal.it/abbonamenti