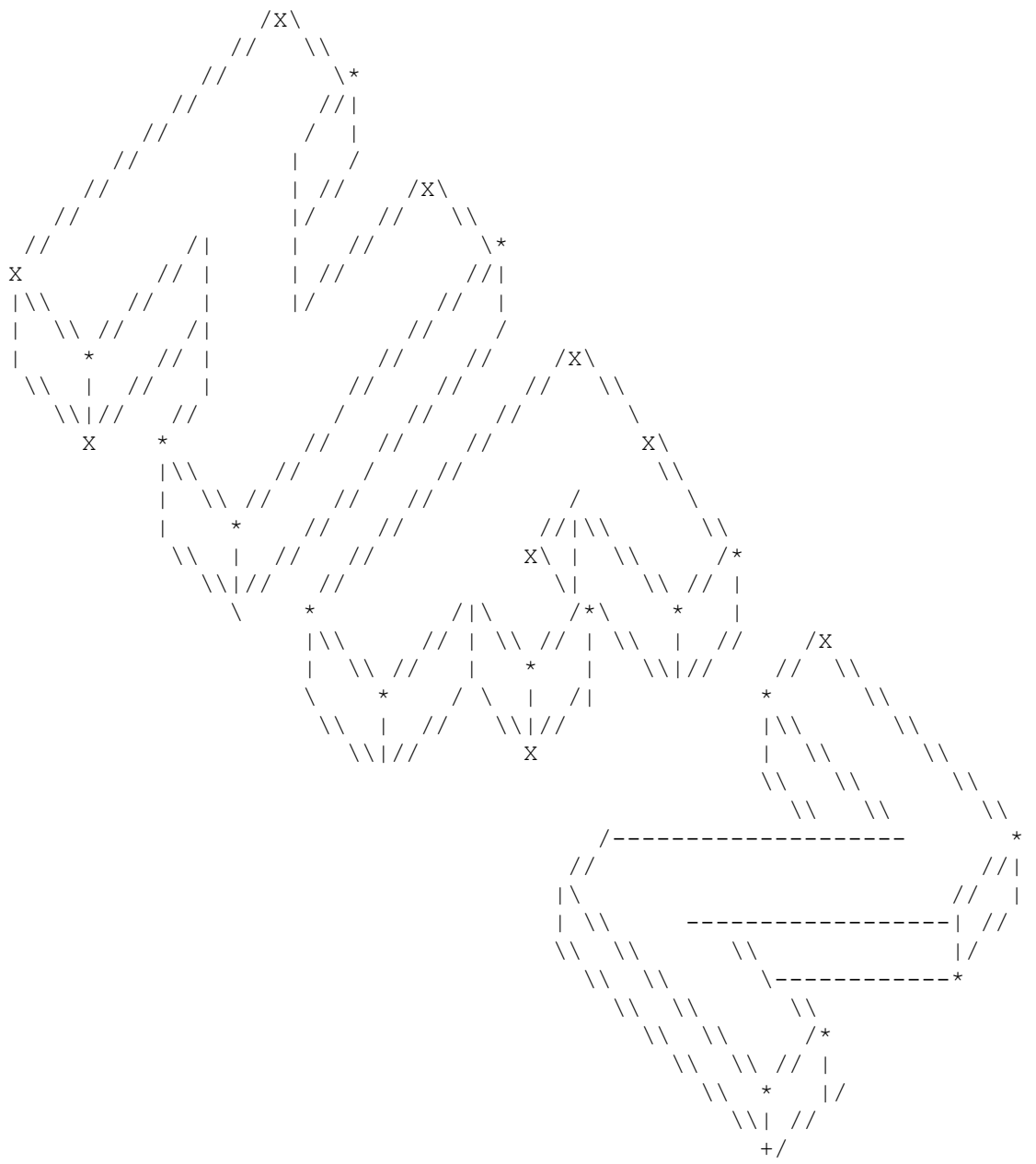
**********No fly Zone Crew**********



----------[E-Zine No 1]----------

Member of the NoFlyZone crew are: []LoRd[V]icio[],Crashes,[D]kl,CityHunter,BIGAlex,
_1/2matto,goony,pex3,Quasar,[Evil],R|Ppy,stefan0,
Capitanmidnight,Pregzt...

----------[Menu Articoli]----------
----------[INTRO]----------

----------[Hacking]----------

----------[THEORY]----------

----------[MISC]----------

Mode E-Zine on:

Prima di tutto xò:

DISCLAIMER:
Tutto il materiale pubblicato in questa zine è di pubblico dominio.
A scopo educativo e di ricerca. Gli autori non si assumono alcuna responsabilità per l'uso di ciò che viene spiegato e di eventuali danni.
Consigli x l'uso: accendere il cervello.

----------[INTRO]----------
----------[[]LoRd[V]icio[]]----------

Tutto iniziò circa 5 mesi fa quando decisi di fondare un chan underground in irc.Scelto il server,rimaneva il nome del chan,a Dkl venne l'ispirazione:NoFlyZone. Vedendo il numeroso afflusso di gente interessata al mondo underground e trovando sul mio cammino ragazzi,ora diventati miei grandi amici di lavoro e nn solo :), preparati,disposti a impegnarsi x uno scopo comune,decisi di fondare la CRew... Eccoci qua,il primo numero di una e-zine ke spero nn finisca come molte altre e ke sia utile anzi utilissima a tutti coloro i quali vogliano conoscere "la rete". La Noflyzone e-zine tratterà diversi argomenti nel modo + kiaro possibile... Sicurezza // Programmazione // Hacking // Cracking // O.S // ecc // ecc //

X ogni rikiesta,info,lamentela,kiarimento,augurio,complimento,e cose di questo genere mandate un e-mail a lordvicio@hotmail.com o fate una capatina in Chan..

Tutto questo nn deve essere di incitamento a far cose illegali o creare problemi ad altre persone ma solo x impararare a difendersi.Vi auguro una buona lettura....
----------*END*----------

----------[INTRO NÝ2]----------
----------[City Hunter]----------

Oggi è il 28 novembre 2001, sono qui a casa, dopo 8 ore di lezione, a cercare di fare un'impaginazione decente. Sono pieno di interrogativi, perchè non ammetterlo. Ci tengo molto a questo progetto, è da tanto che sono appassionato di hacking & c. ed è solo da poco che ne faccio parte un po' più attivamente. Su IRC ho trovato altri con questa passione ed è nato questo gruppo. Sono pieno di speranze e di aspettative per questa zine e mi impegnerà a fondo. Mi auguro che resterete soddisfatti dal nostro lavoro!:-)

Questo è il numero 1 di una nuova zine...l'ennesima direte voi:-)
"Perchè dovremmo leggerla??" mi sembra già di sentirvelo dire! Non so perchè, ma se posso darvi un consiglio sono convinto che qui troverete spesso risposte alle vostre domande e alle vostre curiosità riguardanti tutto (o quasi;-)) ciò che riguarda il mondo underground. Vi chiedo solo un favore: se avete qualcosa da dirci, fatelo in modo costruttivo! Se qualcosa non dovesse essere di vostro gradimento ditelo in modo cortese please!:-) Se qualcosa non vi è chiaro non dovete far altro che contattarci e noi saremo a disposizione. L'ultima richiesta

che ho è questa: pensate a ciò che chiedete e a come lo chiedete. Sinceramente incomincio a stufarmi di quelli che mi chiedono <<mi aiuti a bucare una shell?>> beh...qui cerchiamo di fare qualcosa di buono e con i lamah non vogliamo avere nulla a che farci, quindi se le domande sono queste please join in /dev/null ;-) Finita la solita tiritera passo la parola agli articoli, in fondo sono questi a farvi leggere la zine e non le solite intro:-)

Signori e signore...
No. 1 is Out

----------*END*----------

----------[3]----------
----------[[R|Ppy]]----------
----------[Public FTP protection]----------

****
* TuToRi4l ftp: *
* *
* -Directory protette *
* -Files protetti *
* -Directory nascoste *
****

20/11/2001

[INDICE]

0xbpi-Break Point Information

re all, questo è il mio primo tutorial in assoluto e sarà uno dei tanti che vi dedicher=, per poter trasmettere le informazioni che riesco a trovare o a sniffare dalla rete o per esperienza personale.
Ringrazio subitissimo []LoRd[V]icio[] che mi ha fatto entrare nella crew di Noflyzone (bellissima crew, persone simpaticissime, e colte :pp cheddicci Vicio) sono molto contento di esserne entrato a farne parne. Sarà l'inizio per una formazione personale in questo campo, ancora sconusciuto sotto certi aspetti ;(anche se il tempo a disposizione non è mai troppo ;)
Un grazie a tutti e W Noflyzone :D
Un salutone a tutta la crew, bye.

0x0pr-Premessa

Eccomi, finalmente posso anche io trasmettere ad altri le mie conoscenze ;)

Protocol inizialing:

Avete mai sentito parlare di Directory protette su Server NT o su Unix? beh questo è il tutorial giusto per imparare a crearne un po'.

Quando noi ci connettiamo ad un FTP ovviamente in anonimo, possiamo avere accesso a quasi tutte le cartelle. Dobbiamo subito fare una distinzione, ci sono 2 metodi:

---> 1. Metodo x Microsoft FTP 3.0 - 4.0 - 5.0 - 5.01

```
 ---> 2. Metodo x FTP unix che posso essere WU-ftpd GLftp o altri



0x01-Protezioni in NT
=====================


*************
* IÝ Metodo *
*************

Nello status Windows di FLashfXP notiamo che ci appare il tipo di server ftp
installato:

230 Anonymous user logged in.
SYST
215 Windows_NT version 5.0
PWD
257 "/" is current directory.



Entriamo in ftp con il nostro bel cliente ftp, e diamo un occhiata la struttura
sarà la seguente:

10-25-01  07:23PM       <DIR>
10-22-01  02:14AM       <DIR>          # -=== Tagged by § RIPpy § ===-


Le protezioni che possiamo creare sono le seguenti:

Utilizziamo questi nomi per le cartelle protette: COM1 COM2 COM3 COM4 COM5
AUX PRN NUL, anche se a dire il vero per fare uno cosa un po nascosta ci
conviene utilizzare nomi tipo ~tmp oppure _vti_log.

si crea così:           MKD COM1 / /   ---> MKD COM1[spazio]/[spazio]/


- poi dobbiamo creare una SUBDIRECTORY sotto la com1:

                        MKD COM1 /test/  ---> MKD COM1[spazio]/test/


- per poter entrare nella cartella creata dobbiamo inserire tutto il
  percorso nel nostro caso:

                        123.234.255.3/COM1 /test/

[ la stessa procedura si utilizza con le directory normali del tipo:
        MKD provaprova / / (protetta) ]


- Altra protezione  cartella nascosta (anche se non totalmente)

        MKD / /TEST/   --> MKD /[spazio]/[DIR]/   in questo modo nasconderemo
   la cartelle, puo essere effettuato solo sulla root del nostro ftp. L'unico
   inconveniente è questo, se noi ci spostiamo tramite CHD (change dir) / /
   entreremo in una specia di root quella invisibile e sarà possibile trovare
   anche cartelle create da altri utenti ;)



0x02-Protezioni in Unix, BSD
============================
```

```
**************
* IIÝ Metodo *
**************
```

- Ora vediamo il metodo per proteggere le cartelle su sistemi Unix, Bsd ecc... Dobbiamo fare una piccola considerazione: la stragrande maggiorparte dei sistesmi unix permette i privilegi di scrittura solo nella cartella INCOMING qui dovremo nascondere le nostre cartelle all'interno di quella struttura.

  - Directory Protetta

    ```
    (numero spazi).NOMEDIRECTORY;;(numnero spazi)
    Es.      (6 spazi).testdir;;(5 spazi)
            /      .testdir::     /
    ```

    N.B. con questa protezione per poter entrare nella directory bisogna sapere il numero di spazi usati prima e dopo, non come il metodo di NT specificando la cartella successiva

  - Directory Invisibile

    ```
    (spazio)../
    ES.    / ../ .tagged by;; /scanned by/for []/
           /<1spazio>../....
    ```

```
**********
* Bonus  *
**********
```

Esiste anche una terza protezione, efficace sui files, che funziona esclusivamente su FTP microsoft ma non su unix, quando poi troveremo un ftp non deletable non occorre questa protezione dato che i files non si potranno cancellare OKY????

- Volendo possiamo anche proteggere i files dalla cancellazione dei FUCKING DELETERS semplicemente rinominando i files uploadati sull'ftp e aggiungengo la stringa . / /

Es.

Prima	Dopo	Analisi
mrprn.zip -->	mrprn.zip ./ / -->	mrprn.zip[spazio]./[spazio]/

0x03-Suggerimenti
==========

Suggerimenti per l'utente:

- cercate di nascondere nel subDIR le vostre cartelle protette e ricordatevi di controllare il percorso che sia corretto: se voglio creare una cartella in /~tmp/ciao/ con il flash fxp devo fare MKD COM1 / / e non come vi propone lui /COM1, altrimenti vi crea la cartella sulla root dell'ftp.

- usate nomi di cartelle che si mimetizzino nell'ambiente, che assomiglino a quelle gia presenti

- non sempre gli ftp vi danno tutti i permessi, potete trovare di quelli nondeletable, e persino quelli che non vi fanno vedere il contenuto, è necessario ricordasi le cartelle create.

- NON CANCELLATE MAI NULLA ammeno il pub non sia stato taggato da voi, a quel punto potrette anche deletare l'ftp :DD

- Ricordatevi ad ogni modo che c'è sempre quel qualcuno che è in grado di cancellare le vostre COM1 percui sempre "su con le orecchie" ;)

- Alle volte vi puo capitare che dopo aver eseguito correttamente il login in un ftp anonimo (chiamiamoli pure PUB) vi venga dato un errore del tipo Data Socket Error oppure non si ferma su PASV (nella Task Windows di FlashFXP) e non vi fa piu andare avanti. Non dovete far altro che premere F8 (quick connect) e spostarsi sulla tendina TOGGLES dopodiche disattivate l'opzione PASV MODE (passive mode), per l'uso di upload / download verso un ftp non occorre il pasv mode, nel prossimo tutorial vi spiegher= dove e quando si puo usare.

Saluti a tutti e buona protezione

See you again.....

byez

----> /\ R|Ppy

mail:rippy2k1@hotmail.com

Ringraziamenti e Saluti: a []LoRd[V]icio[] - Crashes - Quasar e a tutti i membri della crew!

----------*END*----------

----------[4]----------
----------[Crashes]----------
----------[Creare una backdoor #1]----------

****
DA UNA PICCOLA IDEA UN GRANDE RISULTATO
****

Prima di presentare il tutorz, un ringraziamento ai miei colleghi d'ufficio ke con molta pazienza hanno atteso per ore al telefono aspettando ke li rikiamassi ....scusate ma ero assorto in una "nube" di pensiero.

Come preparare un Backdoor:

Dunque, dunque, questo tutor è rivolto a tutti quelli ke per qualsiasi ragione di sfogo, volessero prendere possesso di un qualcosa ke nn gli appartiene, anke per il semplice gusto di farlo.
Mettiamo prima un po di musica.Ok, di Backdoor ne troviamo molti sulla rete da SubSeven a BackOrifice ke forse è quello più pericoloso se usato bene, intendo nn da lamerozzi, cmq la mia idea e quella di molti altri è quella di avere un backdoor semplice veloce poco ingombrante e capace di eseguire qualsiasi Task.

Quale linguaggio utilizzare se non il Java forse uno di quelli + diffusi, e probabilmente il + compatibile con tutti i sistemi presenti nella rete e universale (dove lo metti sta e gira...)
Ops, forse sto dimenticando di kiedervi se sapete cosè un Backdoor? Penso di si, ma per ki nn lo sapesse è un code ke attraverso una console sulla Vs makkina vi permetterà di linkarvi con il Pc della vittima. Il resto lo sapete da Voi..è kiaro ke dovete preoccuparvi di sapere se sul PC della vittima ci sia il Java Runtime!

=== Passiamo alle cose serie: per starmi dietro non dovete essere dei program-matori Java , basta solo prestare attenzione,per cui articoleremo il tutor in 3 Parti semplici e veloci:

1) Il Server
2) I Plugin

3) Il Client

k, andiamo avanti. Il Server è quello ke vi dirà quando la Vs vittima è Online, la vittima sarà il CLIENT e il Plugin non è altro ke quella parte di code ke vi permetterà di operare.Il server userà per eseguire le Vs istruzione il ClassLoader, questo vuol dire ke sarà in grado di eseguire qualsiasi cosa voi decidiate di caricare!!

K, gente cominciamo a buttare giù qualcosa:

```
import  java.io.InputStream;
import  java.net.Socket;
import  java.net.ServerSocket;
import  java.net.URL;
import  java.net.URLClassLoader;

// Cominciamo kiamando le varie librerie.
public class BackDoorServer implements Runnable
{
int     srvPort;
int     inputChar;

// La porta door del server è variabile, logicamente sarà quella riservata per
// l'input
ServerSocket    serverDoor;
Socket          commSocket;

// ServerSocket è la porta del Server (2323) mentre il Socket è dove stabiliremo
// la comunicazione
public BackDoorServer() {
this( 2323 );
}
Thread  theDoor         = new Thread( this );
theDoor.start();
// Il server partirà in Thread e inizializzerà la porta 2323
}
public void run() {
try {
serverDoor              = new ServerSocket( srvPort );
//Facciamoci  un bel socket ke ci terrà linkati al Client
Boolean vivo = true;
while( vivo ) {
commSocket      = serverDoor.accept();
// Cosa facciamo ora, mandiamo un segnale di 128bit per
//costruirci un Buffer per la comunicazione con il Client
InputStream     in      = commSocket.getInputStream();
StringBuffer    line    = new StringBuffer( 128 );

while( (inputChar = in.read()) != -1 )
line.append( "" + ( char ) inputChar );
// Ora avrete una Stringa di ritorno dal Client nel Buffer
String  lines   = line.toString();
if( lines.equals( "Fottiti!!" ) ) {
// Se la Stringa è Fottiti, ti conviene kiudere il Pc e piangere
alive =  false;
} else {
{
serverDoor.close();
commSocket.close();
}
```

Perkè non funziona, cazzo nn è possible!!! Ricordate ke cosa vi avevo detto prima ??ke cosa manca ? Il Plugin!!!! Mikioni..bisogna avere il plugin ma questo è un argomento ke affronteremo la prox volta ok? Nel frattempo salvate il file ke avete scritto come vi pare: chiamatelo Backdoor.Java o Mentor.java

insomma fate un po' voi e compilatelo. Ci sentiamo la prossima volta gente!

SALUTI: alla crew, al chan #NoFlyZone #Warez-Planet in particolare a:
/\ LordVicio /\/\LoNeWoLfDeN /\/\ /\/\Cristian84 /\/\ DArklines /\
/\ BigaLex /\/\ Marsio /\/\

********** www.noflyzone-crew.cjb.net **********
********** irc: irc.azzurra.it 6667 #NoFlyZone **********

[Crashes - rocket@freemail.it]

----------*END*----------

----------[5]----------
----------[BIGAlex]----------
----------[MySQL in PHP]----------

Salve a tutti...dato che in tantissimi mi chiedevano di scrivere una nuova guida (e-zines ecc ecc), ho pensato di scrivere un tutorial su come utilizzare il MySQL con il PHP. Il MySQL è un database che è possibile interrogare dall' interno del codice PHP per rendere più dinamiche le proprie pagine web. Ho deciso tuttavia di scrivere il tutorial per tutti coloro che abbiano già un minimo fondamento di PHP (il PHP è grande ed io nn ho abbastanza tempo per trattarlo tutto :), anche perchè esistono già moltissime guide che trattano del PHP, tuttavia molto poche sono quelle che trattano del MySQL, ed ancor di meno quelle che lo fanno in modo chiaro. Per questo ho quindi deciso di scrivere questo tutorial.

++++¤¤} Comandi basilari del MySQL {¤¤++++

Prima di pensare ai comandi basilari del MySQL, dovreste conoscere la struttura: in pratica un database contiene delle 'tabelle' che a loro volta contengono 'righe' e 'colonne'.
Non è possibile scrivere in un database se prima non si costruisce una tabella ed, all'interno della tabella, delle colonne. Le righe possono essere inserite dal PHP quando l'utente che visita un determinato sito web decide ad esempio di registrarsi al sito, inviando i propri dat che vengono appunto salvati in delle righe.
Se non sapete come costruire tabelle, vi rimando all'help di MySQL o allo script di gestione dei database PHPMyAdmin,ottimo per la configurazione dei database in locale ed in remoto.

Passiamo ora ai comandi di base del MySQL:

i comandi di base del MySQL sono pochi: quelli che occorrono per inserire righe, modificarle ed eliminarle.
Un altro comando indispensabile, è SELECT. Vedremo anche l' utilizzo di quest' ultimo.

Per inserire una riga occorre richiamare il comando INSERT INTO.
Ad esempio, vogliamo inserire due variabili nella tabella 'utenti' che contiene due colonne: 'user' e 'pass'. Le variabili che voglio inserire sono $user e $pass.
Il comando da lanciare sarà:

INSERT INTO utenti (user, pass) VALUES ('$user', '$pass');
(nel PHP non occorre inserire il ; dato che viene fatto in automatico. Se inserite un ; riceverete un errore e probabilmente si bloccherà l'esecuzione dello script).

Per modificare una riga (che quindi già esiste), dovete usare il comando UPDATE. Mettiamo ad esempio che io abbia una tabella chiamata 'utenti' che abbia all' interno tre colonne: user email e pass. L'utente vuole modificare l'email e la password. Abbiamo così tre variabili: $user, $email e $pass. Lanciamo quindi il comando per aggiornare il nostro database:

```
UPDATE utenti (email, pass) SET ('$email', '$pass') WHERE user = '$user';
```

In questo modo sostituiremo la vecchia password e la vecchia email con i nuovi dati fornitici dall'utente. (Il WHERE verrà trattato in seguito)

Per eliminare una riga, utilizzeremo invece il comando DELETE. Per utilizzare il comando DELETE, come anche con UPDATE, abbiamo bisogno di una variabile che indichi al database quale riga eliminare, quella che quindi passeremo a 'WHERE', cioè la condizione con la quale il database pu= decidere se eliminare o meno una determinata riga. In questo caso, utilizzeremo la variabile $user, che corrisponde alla colonna user della tabella utenti (il nome utente in genere è unico e quindi dovremmo essere sicuri di eliminare un unico valore, quello legato a quell'utente).Il comando da utilizzare sarà quindi:

```
DELETE FROM utenti WHERE user = '$user';
```

In questo modo tutte le righe con il valore user coincidente con la variabile $user verranno eliminate.

Passiamo ora al comando SELECT: questo comando serve a visualizzare delle righe, in modo da poter ad esempio leggere username e pass per poter effettuare un login. Mettiamo ad esempio di voler leggere i valori di tutte le colonne (*) della tabella utenti (e dato che non utilizziamo la condizione, cioè WHERE, ci verranno proposte anche tutte le righe). Dovremo quindi utilizzare il comando

```
SELECT * FROM utenti;
```

In questo modo riceveremo in una variabile PHP tutte le righe con tutti i valori della tabella utenti.
Nel caso in cui volessimo prendere un solo account, possiamo invece usare la condizione WHERE user = '$user'. Vediamo cosa accade:

```
SELECT * FROM utenti WHERE user = '$user';
```

In questo modo riceveremo tutte i valori di tutte le righe che hanno nella colonna user il valore della variabile $user.

Vediamo ora invece com'è possibile inviare con il PHP le query che permettono di inserire, modificare, cancellare e visualizzare righe.

+++¤¤} I comandi PHP per MySQL {¤¤+++

I comandi per utilizzare i database sono relativamente pochi:
il primo, il più facile e quello più importante è quello relativo alla connessione al database (eheh). Il comando da inserire nel PHP è mysql_connect(host, user, pass).
Con questo comando ci colleghiamo quindi ad un server con hostname (o ip) 'host', nome utente 'user' e password 'pass'.
Faccio subito un esempio:

```
mysql_connect("localhost", "bigalex", "ciauzz");
```

Fatto questo, se il login e l'hostname sono corretti mi collego ad un server.
Per controllare l'avvenuta connessione, il PHP pu= creare una variabile in cui imposta valore 1 se la connessione ha avuto successo, altrimenti valore nullo.
Per controllare, quindi posso fare così:

```
$var = mysql_connect("localhost", "bigalex", "ciauzz");
```

Per controllare l'avvenuta connessione aggiungo una riga...

```
if (!$var) { echo "Impossibile collegarsi al database"; exit; }
```

Et voilà...se non si collega, esce dallo script avvisandomi.
Continuiamo il nostro tour nel mondo del MySQL :)

Il secondo comando (fondamentale) è mysql_select_db(database).
Questo comando ci consente di selezionare un database su cui operare. Per selezionare un database, basta richiamare il comando inviando come variabile database, il nome del nostro database, un esempio rapido di questo comando è mysql_select_db("prova");

Passiamo ancora avanti...ora possiamo fare una query al database. Il comando per eseguirla è mysql_query(sql);
Questo comando restituisce una variabile con il risultato della query (ad esempio con $result = mysql_query($sql); abbiamo in $result il risultato della query).
Proviamo subito a fare una semplice query. Avendo un database chiamato prova ed una tabella chiamata utenti con all'interno due campi: user e pass. E'sottointeso che nel database ci debbano essere delle righe, altrimenti non possiamo richiedere alcun valore! :)Ad esempio, un utente si è registrato col nick bigalex. Vogliamo vedere la sua password per effettuare il login.

```
mysql_connect("localhost", "username", "password");
  /* essendo in localhost è praticamente sicuro che
     avvenga la connessione :) */
mysql_select_db("prova");
$result = mysql_query("SELECT * FROM utenti WHERE nick = 'bigalex'");
```

Abbiamo ora il risultato nella variabile $result. Fin qui tutto ok, ma come facciamo a leggerlo? Facendo echo $result non è possibile essendo un risultato di una query MySQL. Come si fa allora? E beh si utilizza un comando del PHP per leggere le variabili in MySQL, no? ;D
ihihih...allora...per leggere i risultati delle query di MySQL basta usare il comando mysql_fetch_row(result). Questo comando restituisce un array con tutti i campi selezionati (in questo caso *, quindi tutti).
Un esempio:

```
$row = mysql_fetch_row($result);
```

Avremo quindi in $row tutte le colonne richieste nella query. Nel caso in cui le righe siano più di una, bisogna richiedere per ogni riga un mysql_fetch_row.
Procediamo con un esempio:

```
// esistono più righe nel risultato $result!

while ($row = mysql_fetch_row($result))
{
 echo "$row[0] -> $row[1]<br>\n";
}
```

In questo modo per ogni riga ci verrà mostrato la prima colonna, una freccia, e la seconda colonna.

Nel caso in cui vogliamo invece inserire, modificare o cancellare delle righe, ci possiamo affidare al comando mysql_query.
Questo comando infatti non consente solo di mostrare delle righe, ma di fare query di qualsiasi genere.

Una volta fatta una query ed ottenuta una variabile (mettiamo ad esempio $result), possiamo vedere se la query ha avuto effetto su una o più righe (ci viene dato il numero esatto delle righe sulle quali ha avuto effetto quella query).
Procediamo sempre con un esempio:

```
// dopo aver avuto $result

echo "La query ha avuto effetto su " . mysql_affected_rows();
```

Questo comando tuttavia non vale per il comando SELECT. Per quest'ultimo bisogna usare mysql_num_rows($result).
Volendo è possibile assegnare ad una variabile il risultato dei 2 comandi e verificare semplicemente che il valore sia maggiore di 0, in modo da verificare l'avvenuto aggiornamento senza badare al numero di righe su cui ha avuto effetto la query.
Per essere più chiaro faccio un esempio:

```
$num_righe = mysql_num_rows();
if ($num_righe > 0) echo "Il comando ha avuto effetto";
else echo "Il comando non ha avuto effetto";
```

Con questo termino il tutorial, ricordandovi di chiudere la connessione al database con mysql_close();

Nota dell'ultimo minuto: nella costruzione del mio sito, ho avuto dei problemi con le virgolette: se ci fate caso, inserendo una virgoletta in un INSERT INTO si possono avere vari problemi di sicurezza, anche seri!! Per questa ragione i coders del PHP hanno inserito un comando moooolto utile per tutti coloro che giocano come noi con i database :)
Il comando è addslashes e funziona nel seguente modo:
bisogna assegnare ad una variabile il risultato del richiamo di addslashes (variabile). Il risultato sarà compatibile con il MySQL.
Esempio:

```
$variabile = addslashes($variabile);
```

In questo modo aggiungiamo semplicemente dei backslash prima delle virgolette, degli apostrofi e dei backslash, aggiornando la variabile $variabile con il nuovo codice compatibile con MySQL.
Esempio pratico:

```
$variabile = "abc c'è \\comp";
$variabile = addslashes($variabile);
echo $variabile; //ora $variabile sarà diventata "abc c\'è \\\\comp"
```

++++¤¤} Saluti {¤¤++++

In un tutorial pu= mancare tutto, tranne che i saluti :)
Allora, cominciamo dai tank commandos, mighty, il nuovo arrivato ValK, spyro, e tutti gli altri ke nn vedo mai, poi, marsio, la #noflyzone, goliath, aladdin, an4chr0n (ringraziandolo x il logo in flash 5 x il mio sito), blacksoul, dionakra, crashes, e4m, quasar, raptor,shisha, [elektro], sindon2k (che sta facendo il militare), thegass, x[morph]x, holaz, `advanced, delilah, xpterminator, mezzomatto,_kome_, {kurt} ed omega (iihih, chi nn c'era qua, nn c'era neppure nei log del mirc ;D)(sono offesissimo!!ndCityHunter)

Per eventuali chiarimenti mi trovate in chat su azzurranet al canale #hack o alla mail totalmeltdown@libero.it

byezz all!

--
BiGAlex
Everyday is a strike 4 my head!
E-Mail: totalmeltdown@libero.it
SiTE: bigalex.cjb.net (f2s sucks!!!)

----------*END*----------

----------[6]----------

---------------------------------[City Hunter]---------------------------------
---------------------------------[TCP/IP #1]-----------------------------------

Hola a tutti!Eccoci qui con il mio primo tutz per questa nuova zine a cui
auguro un ottimo successo! Questo 1Ý articolo tratterà le fondamenta dei
protocolli TCP/IP per poi proseguire nei prox tut in vari approfondimenti
e usi più maliziosi(spoofing e frammentazione dei pakketti).
Iniziamo coi ringraziamenti:LordVicio per avermi fatto entrare nel gruppo
e a tutti i membri del NfZ e non: XpTerminator,Deli,Marsio,BIGAlex,e poi...boh??
Mi raccomando,diamoci dentro!
Ok...iniziamo subito!

Prima una piccola intro:

```
+---------------------------------------+
|    Protocolli dello strato rete       |
+---------------------------------------+
```

A questa categoria appartengono non solo l'IP ma anche tutti quei protocolli
che, come il suddetto, forniscono un servizio di distribuzione dei datagrammi
e non quello di trasporto e controllo.

```
+-------------------------------+
|   IP(una prima introduzione)  |
+-------------------------------+
```

Come detto sopra è il protocollo atto alla distribuzione dei datagrammi. E' un
protocollo inaffidabile e senza connessione. Inaffidabile perchè non vi è alcun
controllo sull'effettiva ricezione dei pacchetti inviati e senza connessione,
cioè ogni pacchetto ha vita propria, indipendente dagli altri generati dallo
stesso host.

```
+-------------+
|    ICMP     |
+-------------*
```

ICMP sta per Internet Control Message Protocol. Come visto sopra l'IP non dà
alcuna conferma dell'arrivo del pacchetto o di altri errori. L'ICMP serve agli
host a informare i loro corrispondeti di un errore,di un eventuale cambiamento
nelle tabelle di routing o ad effettuare test di raggiungibilità (il comando ping
per es).Gli ICMP viaggiano incapsulati nei pacchetti IP quindi purtroppo sono
soggetti agli stessi problemi dell'IP...possono essere perduti,duplicati e
quant'altro.

```
+-----------+
|    ARP    |
+-----------+
```

ARP sta per Address Resolution Protocol e serve a determinare gli indirizzi
fisici che l'IP utilizzerà per trasmettere i messaggi sulla rete locale.
Non è questa la sede per trattare meglio questo argomento...magari più avanti ne
parler=.

```
+-------------------------------------------------+
|     Protocolli dello strato trasporto           |
+-------------------------------------------------+
```

A questo strato appartengono i due protocolli normalmente usati per le comuni-
cazioni:
il TCP e l'UDP.
Partiamo da quest'ultimo.

```
+-------+
|  UDP  |
+-------+
```

l'User Datagram Protocol è un protocollo utilizzato principalmente all'interno di una singola rete in modo tale che la percentuale di successo della trasmissione sia ottima. Questo è fondamentale perchè,come l'IP, anche l'UDP non fornisce alcuna garanzia sull'arrivo ecc(vedi problemi dell'IP). Questi problemi vengono minimizzati dal programma che usa l'UDP. Viene usato perchè è un protocollo relativamente snello, libero cioè dagli appesantimenti necessari al controllo degli errori.

```
+-----+
| TCP |
+-----+
```

Questo protocollo è progettato per fornire un servizio affidabile di consegna di una sequenza arbitraria di bit.
TCP è ci= che ci permette di trasferirei nostri file senza perdite di dati(cosa che sarebbe molto probabilemente accaduta se avessimo usato solo IP), che dà vita ai vari servizi quali Telnet, FTP, Http ecc.(se vi interessa la specifica ufficiale è nell'RFC 793).

Ok...qui finisce la parte più generale. Ora ci immergiamo nel dettaglio del TCP/IP

```
+--------------------------------------+
|    The magic word of IP protocol     |
+--------------------------------------+
```

Questo è il protocollo principale del set TCP/IP.
Abbiamo detto che fornisce un servizio inaffidabile, cioè che non garantisce l' arrivo e il corretto ordine dei dati trasmessi. Come vedremo in seguito importantissimo per lo spoofing e la gestione dei raw_socket è la gestione dell'header IP.
Diamoci un'occhiata:

```
    +-----------------------------------------------------------------+
    |Version|  IHL  |Type of Service|          Total Length           |
    +-----------------------------------------------------------------+
    |         Identification        |Flags|      Fragment Offset      |
    +-----------------------------------------------------------------+
    |  Time to Live |    Protocol   |         Header Checksum         |
    +-----------------------------------------------------------------+
    |                       Source Address                            |
    +-----------------------------------------------------------------+
    |                    Destination Address                          |
    +-----------------------------------------------------------------+
    |                             data                                |
    +-----------------------------------------------------------------+
    |                            option                               |
    +-----------------------------------------------------------------+
```

Version: beh...ovvio no? La versione dell'IP.
IHL: lunghezza dell'IP.
Total Length: lunghezza totale dell'IP.
Identification: serve a distinguere il pacchetto dagli altri inviati dallo stesso sistema.
TTL: è il numero massimo di router attraverso cui il pacchetto pu= passare. Quando arriva a 0 il pacchetto viene scartato.
Checksum:vedi dopo(nel paragrafo TCP).
Flags e Offset: servono per la frammentazione dei pacchetti.
Gli altri due vi arrangiate:-))
Data e option: beh...i dati e le eventuali opzioni.

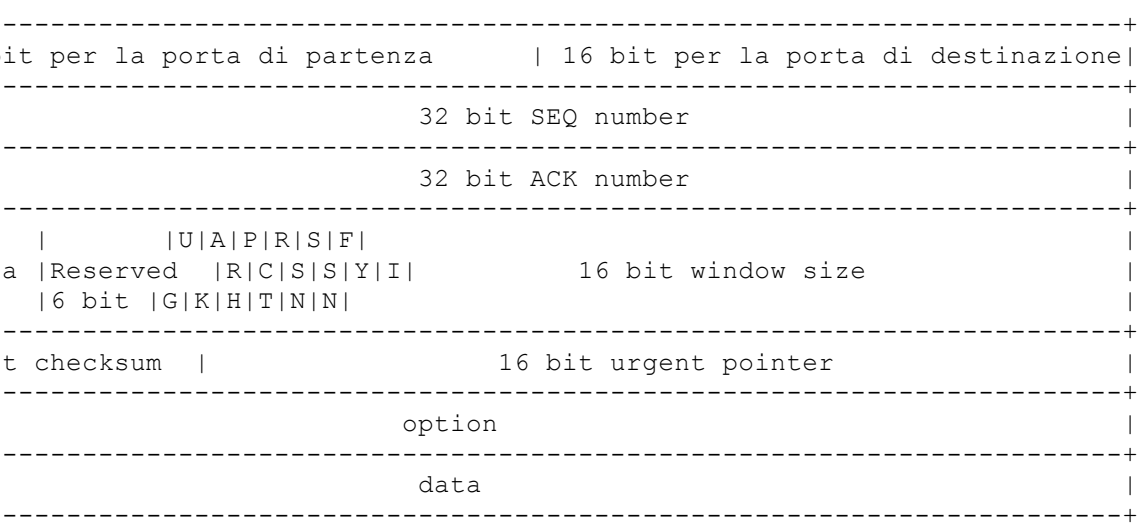
```
+--------------------------------------+
|    The magic word of TCP protocol    |
+--------------------------------------+
```

Il TCP ha il compito di trasformare i dati in datagrammi più piccoli e poi,una volta giunti al destinatario, riconvertirli nel formato originale. Inoltre

rimanda i dati persi al destinatario dove i dati ricevuti vengono riassemblati nel corretto ordine.
Il trasferimento dati avviene in modo bufferizzato, cioè prima di effettuare una trsmissione attraverso IP, il TCP attende di aver riempito un'area di memoria (buffer appunto) con una quantità di dati che valga il costo della trsmissione:)
C'è ancora da tener presente che tutti i dati TCP sono incapsulati in un datagramma IP:

Tutti e tre assieme formano il datagramma IP, gli ultimi due(quelli che iniziano con TCP per intenderci) il segmento inerente al TCP

Qui guardiamo con attenzione il TCP Header:



Ok...direi che alcune parti sono abbastanza chiare...cmq diamo un piccola spiegazione

I primi due campi sono i bit(16) dedicati alla porta di partenza e alla porta di comunicazione(es:io(190.69.69.69) dalla mia porta 23 mi collego a pippo (120.33.33.33) alla sua porta 21...23 e 21 sono le porte. Esempio un po' del cazzo...cmq rende l'idea!)

I due campi da 32 bit sono importantissimi per lo spoofing e sono il numero di sequenza ed il numero di riconoscimento(appunto SEQ e ACK).

I 4 bit sono le dimensioni massime dell'Header TCP.

6 bit riservati e poi le flag che identificano di volta in volta il comportamento di TCP all'interno della connessione(sarebbero tutte quelle letterine).

16 bit per le window. Questo specifica il numero di byte che ogni lato è disposto ad accettare. Faccio un esempietto(odierete i miei esempi alla fine del tutz:-))): appena il computer riceve i dati, lo spazio nel campo Window decresce indicando che il ricevente ha ricevuto i dati. Quando il campo giunge a zero il computer che manda i dati smette di trasmettere.

16 bit per il checksum che viene calcolato solo sull'Header IP e non sui dati trasmessi. Solitamente per calcolarlo si usa un algoritmo rippato dal comando ping:-)

16 bit per l'urgent pointer. E' usato raramente. In sostanza dice al computer remoto di non processare i vecchi dati ricevuti e di ricevere quelli nuovi.

Seguono le opzioni e i dati veri e propri.

Facciamo ancora una piccola parentesi sulle flag del TCP:

SYN	SYNhronize	sincronizza i numeri SEQ
ACK	ACKnowledge	riconosce i dati inviati dal peer
FIN	FINished	l'origine ha finito di inviare dati
RST	ReSeT	interrompe la connessione
PSH	PuSH	invia dati al ricevente il prima possibile
URG	URGent	dati urgenti in arrivo dall'origine

Ok...direi che per il TCP è tutto!

Per questa volta finiamo qui...la prox volta vediamo come due computer si connettono tra loro. Studiate gente, studiate e buon lavoro a tutti!:-)))
Ciauzzzz >>>>>>>>>>HACK tHe FuckInG PlaNeT<<<<<<<<<<<<<

----------*END*----------

----------[7]----------
----------[[]Lord[V]icio[]]----------
----------[Cos'è IPV6]----------

IPV6:

Il nostro attuale protocollo è standarizzatoe la sua attuale versione è la Ipv4. Questo nuovo protocollo è nato innanzitutto per ovviare al problema della quantità di indirizzi presente oggi con il vecchio protocollo evitando di andare incontro alla necessità di mettere in piedi un protocollo TCP/IP nuovo. Ipv4 ha un intestazione di 32 bit in confronto al nuovo IPng ke ne ha ben 128. Uno dei + evidenti benefici, a mio parere, è l'assegnazione automatica dell indentificatore di interfaccia detto EUI-64 e quindi un estrema facilità a metter su una bella rete, che abbia come dato unico questo identificatore di interfaccia per ogni macchina,e ne generi indirizzi Site local e Link local. Poi con un router, si pu= assegnargli un Global IP sempre sfruttando l'indirizzo MAC,dal quale il protocollo ricava l'identificatore di interfaccia.
Attualmente ipv6 nn è un vero e proprio protocollo come lo è il v4: "x ora" non è altro che un tunnelling,ovvero il traffico IPv6 ke viene trasportato in pacchetti IPv4.La maggior parte degli host sono dual-stack, e possono utilizzare i tunnel per instradare i pacchetti IPv6 su porzioni di rete solo IPv4 per raggiungere l'altra IPv6.
Allo stato attuale questo viene fatto dal SIT (Simple Internet Transition)

Principali Caratteristiche del SIT:

Box dual-stack (dette anche IPv4/IPv6)

Incapsulamento di pacchetti IPv6 in pacchetti IPv4 per attraversare parti di rete ancora solo IPv4

Come abbiamo detto ipv6 ha un intestazione di 128 bit.
L'indirizzo e' composto da due campi il cui primo di lunghezza variabile, chiamato Format Prefix (FP) indica il tipo di indirizzo, il secondo e' l'indirizzo vero e proprio.
Gli indirizzi sono rappresentati mediante stringhe testo con il seguente formato: x:x:x:x:x:x:x:x dove le x sono valori esadecimali rappresentanti 16 bit ciascuna.
IPV6 3ffe:13e4:1234:1aef:0000:0000:1234:0001 pu= essere rappresentato come 3ffe:13e4:1234:1aef::1234:1 ..... x rappresentarlo così nn bisogna levare zeri di qua e di la :) ma bisogna seguire una regola: gli zeri (in forma ... :0000:0000:0000:0000:... oppure ...:0:0:0:0:..), possono essere "fusi" in ..::.., ma SOLO UNA VOLTA in tutto l'indirizzo.
3ffe:13e4:0000:0000:34:0000:0000:0001 non potrà mai essere visualizzato come 3ffe:13e4::34::1, ma ad esempio 3ffe:13e4::34:0:0:1.

Tutto kiaro??avete altre domande??venite al chan della crew o mandatemi un e-mail xkè nn ho + intenzione di scrivere :P ... x il suo utilizzo su varie piattaforme consultati gli atri tut della crew...Spero di esser stato kiaro....byez

SALUTI:alla crew,al chan #noflyzone,ai chan #lordspirit #winadmin #hack ....in particolare a LoNeWoLfDeN,Crashes,Cristian84,DArklines.

FUCK:tutti i lamah,alla mia ex,a lordsabotatore al re dei lamah alexmessomalex e a tutta quelli ke fanno le stanze hack in c6 ihhihi

www.vicio84.3000.it
www.noflyzone-crew.cjb.net

dove trovarmi:
c6: vicio84 o lordvicio
irc: irc.azzurra.it 6667 #NoFlyZone nick []LoRd[V]icio[]

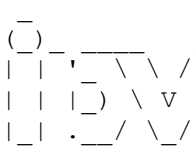
[[]LoRd[V]icio[] -lordvicio@hotmail.com]

----------*END*----------

----------[8]----------
---------- [Quasar]----------
----------[Implementare IPV6 si Linux]----------

__NOFLYZONE CREW__

http://www.noflyzone-crew.cjb.net/



Tutorial forgiato da QUASAR Novembre 2001

Questa nn vuole essere una guida dettagliatissima ma un semplice aiuto a chi vuole senza troppa fatica sperimentare l'ipv6 sulla propria linux box.
Alcune nozioni e approfondimenti sono state prese dalla guida ufficiale all'ipv6 "http://www.linuxhq.com/IPv6/"
"http://www.bieringer.de/linux/IPv6/"

__LA MAPPA DEL TUTORIAL__

[1] Software necessario e installazione

[2] Configurare Il proprio KERNEL

[3] Come creo il mio tunnel ipv6? guida ai Tunnel Broker

[4] Testare il proprio tunnel

[5] Entrare in chat con l'ipv6 :D


```
 _
/ |
| |
| |_
|_(_)  Software necessario e installazione
```

Elenchero' i programmi strettamente necessari da installare

-> iputils-ss010824.tar.gz <-

http://bofh.st/ipv6/downloads/frp.inr.ac.ru/iputils-ss001011.tar.gz

-> net-tools-1.60.tar.bz2 <-

ftp://ftp.netwinder.org/users/p/philib/net-tools/net-tools-1.60.tar.bz2

Per reperirli andate in www.freshmeat.net oppure usate i link da me suggeriti

Una volta scaricati i file date i comandi

```
tar xfvz iputils-ss010824.tar.gz
bunzip -d net-tools-1.60.tar.bz2
tar xfv iputils-ss010824.tar.gz
```

Entrate nelle rispettive cartelle di ognuno dei due leggete i file INSTALL (o readme) dando il comando 'cat INSTALL' normalmente per compilare un pro--gramma bisogna dare i seguenti comandi

```
./configure (nn sempre, ma se presente dare ./configure --help per opzioni)
make
make install
```

Ok ora dovreste avere tutto il necessario per continuare.

```
 ____
|___ \
  __) |
 / __/ _
|_____(_) Configurare Il proprio KERNEL
```

E' possibile abilitare l'ipv6 anche da versioni come 2.2.X basta scaricarsi

la patch, consiglio cmq di utilizzare una versione del kernel >= 2.4.5 per nn avere problemi con la mia guida [www.kernel.org]
Una volta scaricato il sorgente del kernel spostatelo nella cartella /usr/src date questi comandi

```
su root [password]
mv linux-2XX.tar.gz /usr/local
cd /usr/local
mkdir linux-2XX             (a seconda della versione)
mv linux-2XX.tar.gz linux-2XX
cd linux-2XX
tar xfvz linux-2XX.tar.gz
cd linux
```

Ora che avete il kernel decompresso nella vostra cartella compiliamolo, date questi comandi:

```
make mrproper xconfig
```

Vi apparira' un menu' grafico, dovrete abilitare le seguenti opzioni per l'ipv6:

OPZIONE	SCELTA	YES MODULE NO
Code maturity level options	Prompt for development and/or incomplete code/drivers	YES
Neworking options	Packet socket	YES
	Unix domain sockets	YES
	TCP/IP networking	YES
	The IPv6 protocol	YES
File systems	/proc filesystem support	YES

Ora andare su SAVE AND EXIT e dare i seguenti comandi

```
make dep
make clean
make bzImage
make modules
make modules_install
```

oppure

```
make dep clean bzImage modules modules_install
```

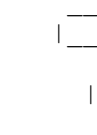
Ora bisogna aggiornare il LILO per far si che all'avvio del pc si possa bootare l'immagine del kernel appena creata, consiglio di fare una copia dell'immagine bzImage nella cartella di /boot

```
cp /usr/src/linux/arch/i386/boot/bzImage /boot/linuxXXX
```

(mettere al posto delle XXX la versione...cosi' gli cambiamo direttamente

```
 il nome e sara' piu' chiaro per noi :D )

Ora aprite il file /etc/lilo.conf ed aggiungete la seguenti righe

image=/boot/linuxXXX
root=/dev/hdXX         (dipende da dove si trova la vostra partizione '/')
label=LinuxXXX
read-only

Dove XXX è il numero del kernel nonche' il nome logico del file :D
Ora sempre da root digitate alla console LILO e aggiornerete il lilo
Bene ora riavviate il computer e scegliete all'avvio la vostra nuova
immagine.
Bene ora per vedere se il kernel riconosce l'ipv6 facciamo una prova,
prendete i diritti di ROOT e scrivete

ifconfig

Dovrebbe darvi il seguente output se nn vedete 'inet6 addr: ::1/128
Scope:Host'
allora l'ipv6 nn è abilitato


lo   Link encap:Local Loopback
     inet addr:127.0.0.1  Mask:255.0.0.0
     inet6 addr: ::1/128 Scope:Host
     [...a noi interessa fino a qui...]

Editare ora il file /etc/protocols e aggiungere

ipv6            41      IPv6            # IPv6
ipv6-route      43      IPv6-Route      # Routing Header for IPv6
ipv6-frag       44      IPv6-Frag       # Fragment Header for IPv6
ipv6-crypt      50      IPv6-Crypt      # Encryption Header for IPv6
ipv6-auth       51      IPv6-Auth       # Authentication Header for IPv6
icmp6           58      IPv6-ICMP       # ICMP for IPv6
ipv6-nonxt      59      IPv6-NoNxt      # No Next Header for IPv6
ipv6-opts       60      IPv6-Opts       # Destination Options for IPv6

Editate il file /etc/hosts ed aggiungete

::1               localhost-v6.localdomain localhost-v6
fe00::0           ip6-localnet
ff00::0           ip6-mcastprefix
ff02::1           ip6-allnodes
ff02::2           ip6-allrouters
ff02::3           ip6-allhosts

[NOTA: sostituire localhost e localdomain se avete dato un nome diverso
       al vostro pc :D]

Ora facciamo un'altra prova date questo comando

ping6 ::1

(ecco l'output)

PING ::1(::1) from ::1 : 56 data bytes
64 bytes from ::1: icmp_seq=1 ttl=64 time=125 usec
64 bytes from ::1: icmp_seq=2 ttl=64 time=109 usec
64 bytes from ::1: icmp_seq=3 ttl=64 time=111 usec
64 bytes from ::1: icmp_seq=4 ttl=64 time=108 usec
64 bytes from ::1: icmp_seq=5 ttl=64 time=156 usec
64 bytes from ::1: icmp_seq=6 ttl=64 time=109 usec
```

Se è tutto ok bene andiamo avanti con la prossima parte



Come creo il mio tunnel ipv6? guida ai Tunnel Broker

Ora che avete tutto perfettamente funzionante dovrete creare un tunnel che vi permetta di dialogare con un'isola ipv6, in genere bisogna registrarsi a un TUNNEL BROKER (distributore di tunnel) che vi dia un IP ipv6 associato al vostro IP normalmente usato con ipv4 (il protocollo odierno) una volta fatto questo spiego ora principalmente cosa bisognera' fare ogni qual volta ci si colleghera' con l'ipv6

-Collegarsi a internet
-Andare sul sito del proprio tunnel broker identificarsi e comunicargli il proprio IP (ipv4)
-Aggiungere un route (strada) all'isola tramite IP e altre cose

Principali TUNNEL BROKER

http://www6.edisontel.com/

http://carmen.cselt.it/ipv6/

http://www.bersafe.it

Grazie a questi tunnel broker potrete usufruire di ipv6 infatti essi forniscono
una rete sovrapposta a internet dove le proprie isole ipv6 comunicano tra loro,
accessibili tramite un gateway e l'indirizzo ipv6 fornito dal server.

NOTA : Ogni qual volta si vuole usufruire del proprio tunnel bisogna comunicare
al TUNNEL BROKER il proprio ip (ipv4) in quanto dinamico, altrimenti nn sapra' associare all'ip (ipv6) il corrispondente ip (ip4)

Testare il proprio tunnel

Ora nn rimane che testare il proprio ipv6 nonche' di attivare l'interfaccia ipv6
ogni qual volta si comunica al TUNNEL BROKER il proprio ip (ipv4)
Una volta comunicato l'ipv4 a un tunnel b. date i seguenti comandi alla vostra shell

```
ifconfig sit0 up
ifconfig sit0 add <metti qui il tuo ipv6>
route -A inet6 add ::/0 gw ::<metti qui l'ipv4 tunnel broker>
```

Ecco ora è tutto ok avete dato un ROUTE (una strada) che porta al vostro tunnel broker e che vi dara' il vostro TUNNEL :D e avete attivato l'interfaccia ipv6, ora nn resta che provarlo per esempio con una bella chattata :D oppure date questo comando

```
ping6 www.6bone.com
```

Se il ping vi risponde allora il tunnel è OK! :D

```
 ____
| ___|
|___ \
 ___) |
|____(_)Entrare in chat con l'ipv6 :D
```

L'ipv6 richiede il porting della maggior parte del software ma per chattare con il protocollo ipv6 si puo' usare gia' da ora BITCHX e XCHAT in quanto è stato gia' implementato, vi diro' ora a grandi linee come compilarsi XCHAT abilitando l'ipv6

-Scaricate l'ultima versione di XCHAT

-Comandi :

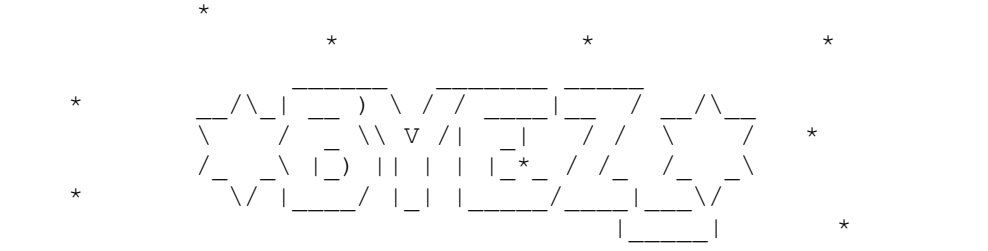
```
tar xfvz xchat_xxx.tar.gz
cd xchat_xxx
(date ./configure --help per vedere le opzioni disponibili)
./configure --enable-ipv6
make
make install
```

Ora avete compilato e installato Xchat con il supporto ipv6 attivo,da ora in poi fornendo un server che supporti l'ipv6 potrete collegarvi con questo protocollo Io vi consiglio il server azzurra dove guarda caso si trova il chan #noflyzone ;D

```
ngnet.azzurra.org
```

Buona chattata!

```
+-------------------------------------------------------------------------+
|#########################################################################|
+-------------------------------------------------------------------------+
```



```
              *
                             *                          *     *

                                         *



     +---------------------------------------------------------------------------+
     |###########################################################################|
     +---------------------------------------------------------------------------+


-------------------------------------*END*---------------------------------------

--------------------------------------[9]----------------------------------------
--------------------------------[goony & haikia]---------------------------------
---------------------------[Virtual Private Network]-----------------------------


   ---------------------------------------------
   Gli autori non sono responsabili dei danni
   futuri causati a cose e/o persone utilizzando
   questo documento... :)
   ---------------------------------------------


 Obiettivo:
 ----------
 Capire cos'è una vpn e realizzare un collegamento criptato tra 2 reti locali.


 Cassetta degli attrezzi:
 ------------------------
 Noi abbiamo utilizzato questi elementi per raggiungere l'obiettivo.
 Voi non siete necessariamente legati ad utilizzare questi...

 - 2 macchine Linux: RedHat 7.1 (www.redhat.com)
                     Trustix Secure Linux 1.5 (www.trustix.net)
   queste avranno la funzione di gateway, firewall...
 - kernel linux 2.4.7 (www.kernel.org)
   quelli di FreeS/WAN consigliano almeno il 2.2.19;
 - FreeS/WAN 1.91 (www.freeswan.org)
   implementazione opensource del protocollo IPSEC per realizzare
   vpn con linux. non è l'unica soluzione (pptp?!) ma a noi
   ci sembra molto valido ed affidabile;
 - tcpdump (www.tcpdump.org)
   bhe lo conoscete...
 - basi di linux & networking... nel caso leggetevi i riferimenti
   che trovate alla fine del documento...
 - fortuna, pazienza! sempre! :)


 Iniziamo con un po' di teoria
 -----------------------------
 Per prima cosa il termine inglese VPN sta per "Vitual Private Network",
 ovvero una rete privata virtuale. Le vpn sono nate con il fine principale
 di collegare in modo sicuro due o più reti private, utilizzando reti
 pubbliche (internet), come mezzo di trasporto dei dati. Quindi in poche
 parole parliamo di vpn quando trattiamo reti private che si estendono
 su reti accessibili pubblicamente (non obbligatoriamente internet) e
 indipendenti dalla tecnologia utilizzata per realizzarle.
 Qui sotto un semplice schema per chiarire la teoria:

   ---------------                          ---------------
   |   rete A    |                          |   rete B    |
   |             | <---> ( internet ) <---> |             |
```

```
  | 192.168.1.0 |                          | 192.168.2.0 |
  ---------------                          ---------------
   Reggio Emilia                              Toronto
```

Una volta che le due reti sono state collegate tra di loro tramite vpn, l'utente appartente alla reta A potrà ad esempio pingare un host della rete B, passando da internet! Figo vero?! Non solo pingare, ma anche lavorare in remoto (ssh), condividere files e stampanti (samba) e tutto quello che vi passa per la testa, abbiate fantasia.
Ora vi chiederete per=, e la sicurezza?! E qui il bello!
Tutti i dati che viaggiano su internet tra le due reti possono (devono!) essere criptati con svariati algoritmi per impedirne l'intercet-tazione e verificarne l'autenticità. Parliamo allora delle 3 caratteristiche fondamentali di una vpn: privatezza, integrità e autenticazione.

1- privatezza: permette che un pacchetto sia ricevuto e leggibile solo e soltanto al destinario di esso, rendendo inefficace l'utilizzo di sniffer da parte di utenti smanettoni;

2- integrità: permette che i dati arrivino a destinazione integri e quindi inalterati durante il tragitto;

3- autenticazione: permette di verificare con efficenza l'identità del mittente, evitando ad esempio fenomeni di spoofing;

Per ottenere questi risultati le vpn utilizzano protocolli di rete particolari, tra i quali uno dei più rinomati è sicuramente IPSEC.

- IPSEC: "IP Security". E' un set di estensioni al protocollo IP che permettono la criptazione di dati e i particolare i 3 elementi sopra descritti. IPSEC offre un servizio simili all'SSL ma lavorando al livello network, per essere cos'ì totalmente trasparente alle diverse applicazioni. IPSEC si compone principalmente di tre protocolli principali:

  - AH (rfc2402): "Authentication Header". Permette l'autenticazione di un pacchetto crittografando con un algoritmo forte l'header IP del pacchetto stesso;
  - ESP (rfc2406): "Encapsulating Security Payload". Permette la privatezza e l'integrità di un pacchetto cifrando il contenuto di esso (payload = campo dati);
  - IKE: "Internet Key Exchange". Permette la negoziazione tra i parametri di connessione...

Questa è solo una piccola introduzione di concetti molto difficili e lunghi da trattare.
Se volete saperne di più consiglio vivamente i link che trovate alla fine di questo documento. Per la realizzazione dell'esperienza queste poche righe sono più che sufficienti.

Installazione

Nella nostra esperienza cosa faremo? Partendo dal presupposto che abbiamo già pronte e configurate due macchine linux collegate ad internet (entrambi con collegamento perenne) che fungono da gateway per due reti private (esempio la 192.168.1.0 e la 192.168.2.0) andremo a collegarle tra di loro, creando un tunnel criptato tra le due macchine e quindi un canale di cumunicazione sicuro tra le due reti. Notare, ovviamente, che le due reti devono avere indirizzi privati diversi.

Iniziamo quindi a configurare il primo gateway, tenendo presente che la stessa configurazione, con solo poche modifiche, sarà utilizzata anche per la seconda macchina. Entrambe avranno un kernel 2.2.19 o superiore con il supporto per il networking. Scarichiamo all'indirizzo ftp://ftp.xs4all.nl/pub/crypto/freeswan/ il pacchetto freeswan-1.91.tar.gz. Scegliamo una directory e s'inzia:

```
note: per installare correttamente FreeS/WAN avremo bisogno di...
         - compilatore C (gcc o egcs)
         - make, path e le solite cosette... ;)
         - glibc
         - GMP (GNU Multi-P recision) library
         - libreire ncurses se volete usare il menuconfig (raccomandato)
         - sorgenti ecc... ecc...

# cd /usr/src
# tar zxvf /usr/local/packages/freeswan-1.9.tar.gz
# cd /usr/src/freeswan-1.9
# make ogo   (invoca "config" per configurare il kernel da linea di comando)
oppure
# make menugo (invoca "menuconfig" per configurare il kernel in modalità
               text-mode. n.b. installatevi le lib ncurses)
oppure
# make xgo (invoca "xconfig" per configurare il kernel con X window)
# make kinstall (installa il nuovo kernel e i moduli se necessari)


L'ultimo comando "kinstall" equivale a fare "make; make install; make
modules ; make modules_install" con i sorgenti in /usr/src/linux.
A questo punto SE tutto è andato bene avremo le librerie necessarie in /usr/local,
gli script necessari per avviare e disattivare i servizi IPsec in /etc/rc.d e i
due file di configurazione /etc/ipsec.conf e /etc/ipsec.secrets.
Facciamo un reboot della macchina, non prima di aver sistemato il lilo...

note: alcune distribuzioni linux permettono di installare FreeS/WAN durante
      l'installazione del sistema:

      - European versions of SuSE Linux (Germany) www.suse.com
      - Conectiva (Brazil) www.conectiva.com
      - the server edition of Corel Linux (Canada) www.corel.com
      - the Polish(ed) Linux Distribution (Poland) www.pld.org.pl
      - Trustix Secure Linux (Norway) www.trustix.net (veramente carina!)

Verifichiamo ora che l'installazione sia avvenuta con successo. Durante il reboot
(date un occhio con il dmesg...) controlliamo che:

- la versione del kernel sia quella corretta e funzionante;
- appaia il messaggio di inizializzazione di KLIPS;
- Pluto sia stato avviato correttamente;

Oltre a questo usiamo...

# ipsec --version         così possiamo vedere che il path è corretto e la
                          versione di IPsec;
# ipsec whack --status    chiede a Pluto informazione sullo stato del processo

Naturalmente le informazioni di debug che riceveremo non saranno ottimali,
dobbiamo ancora configurare il tutto!


Configuriamo:
-------------
Sono due i files di configurazione di IPsec:

/etc/ipsec.conf    per configurare il tutto con le informazioni relative alla
connessione...
/etc/ipsec.secretc contiene la chiave pubblica e privata utilizzate per la
criptazione dei dati...

note:  l'algoritmo di cifratura utilizzato è l?RSA a 2.048 bit, il quale prevede
       che ogni macchina presente nella Vpn possieda una chiave pubblica (nota a
       tutte le macchine della vpn) e una chiave privata.
       Queste chiavi sono generate durante l'installazione, ma potete crearle
```

voi stessi a mano per mezzo del comando IPSEC_RSASIGKEY (http://www.
freeswan.org/freeswan_trees/freeswan-1.91/doc/manpage.d/ipsec_rsasigkey.
8.html),ad esempio lanciando: # ipsec rsasigkey --verbose 2048 >mykey
che genererà le chiavi e le scriverà nel file "mykey". A quel punto
andate a recuperarle e mettetele nel vostro ipsec.secrets. (seguite le
istruzioni sul sito di FreeS/WAN)

ipsec.conf:

```
# basic configuration
config setup
     interfaces="ipsec0=eth0"
     klipsdebug=none
     plutodebug=none
     plutoload=%search
     plutostart=%search

conn %default
     keyingtries=0

conn linux1-linux2
     left=212.*.15.101
     leftsubnet=192.168.1.0/24
     leftnexthop=212.*.15.1
     right=213.*.20.66
     rightsubnet=192.168.2.0/24
     rightnexthop=213.*.20.65
     auto=start
     authby=rsasig
     leftid=@firewalle.foobar.net
     rightid=@fw.barfoo.it
     leftrsasigkey=0x010373f12dd6e1d244895bfc237433bac1c0da...
     rightrsasigkey=0x01037ff85e024bdbee96a64cfdfa3fb3e1f7a...
```

Da come si pu= notare il file è diviso in tre sezioni: la prima e la seconda
definiscono i parametri di carattere generale. Il pi\xf9 importante \xe8
"interface" che definisce tramite quale interfaccia di rete la macchina linux
si collega al mondo esterno. In generale tale parametro \xe8 impostato verso
l'interfaccia alla quale è impostato il defaultroute (default gateway) della
macchina , "ppp0" in caso di connessione con pppd, "ippp0" in caso di
connessione con isdn4linux ecc.
La sezione più importante è invece la terza, che definisce le caratteristiche
delle due macchine.
La "prima" macchina viene identificata come "left", la seconda come "right".

Descriviamo nel dettaglio le singole direttive:

- conn: assegna un nome alla connessione (tunnel) che vogliamo realizzare;
- leftid: nome completo di dominio della prima macchina;
- leftrsasigkey: chiave pubblica della prima macchina
  (prelevabile dal file /etc/ipsec.secrets nel quale è caricata
  la variabile pubkey);
- left: indirizzo Ip pubblico della prima macchina, ovvero quello assegnato dal
  provider al momento della connessione e associato all'interfaccia di
  rete che collega la macchina ad Internet(per esempio, "ppp0");
- leftnexthop: indirizzo Ip del default gateway assegnato dal provider al momento
  della connessione,ovvero la prima macchina visibile facendo un
  traceroute verso internet;
- auto: definisce il modo in cui la vpn deve essere attivata. Se \xe8 impostato
  su start l'avvio avviene tramite script (/etc/rc.d/init.d ecc.);

Per quanto riguarda la "seconda" macchina (right), le impostazioni sono
esattamente le stesse.

Fate direttamente un copia incolla tra le due macchine.

Prima di provare... e il firewall dove lo mettiamo?!

Se utilizzate dei firewall per proteggere le vostre reti, dovete abilitare il passaggio di
pacchetti di determinati protocolli (vi ricordate di AH ed ESP?!) su una porta particolare.
Leggete qui sotto:

- IKE uses the UDP protocol and port 500
- ESP is protocol number 50
- AH is protocol number 51

Nel nostro caso preoccupiamoci della porta 500 UDP e del protocollo 50.

Funzionerà?

A questo punto vediamo se i nostri sforzi saranno premiati. Per prima cosa diamo un occhio ai log.
Utilizziamo per questo il comando: #tail -f /var/log/messages. (magari in fase di testing, associamo alle due variabili "klipsdebug" e "plutodebug" in /etc/ipsec.conf il valore "all" cos'ì da poter analizzare più logs...) Se poi siamo smanettoni diamo un occhio anche qui: #cat /proc/net/ipsec_tncfg
Ora per avviare il tutto utilizzeremo il comando:

```
# ipsec auto --up name
```

su entrambi i gateway. Notare che "name" corrisponde al valore della variabile "conn" in ipsec.conf.
Nel nostro caso "linux1-linux2". Per fermare il servizio utilizzeremo invece il comando:

```
# ipsec auto --down name
```

Note: volendo possiamo anche srivere/utilizzare script del tipo
"/etc/rc.d/init.d/ipsec start" ecc.
per rendere la cosa più veloce ed automatica. Possiamo aggiungere anche una riga di comando in fondo al file "rc.local" per far sì che il servizio IPSEC si attivi in automatico all'avvio del nosti gateway.

Infine proviamo a vedere se l'obiettivo è raggiunto. Ad esempio da una macchina della rete 192.168.1.0 proviamo a pingare una macchina della rete 192.168.2.0.
(Note: non provate a pingare un host dell'altra rete private direttamente dal gateway, FreeS/WAN non lo permette, e per spiegazione controllate sul sito...)
Se il ping ha successo provate con ssh ecc, abbiate un po' di fantasia.. Provate con il "traceroute" da un host all'altro delle due reti: se tutto è corretto dovreste raggiungerlo con un solo solo hop.
In caso contrario controllate:

- connessione ad internet dei gateway;
- regole del firewall;
- configurazione dei due ipsec.conf;
- esattezza della chiavi;
- provate a pingare più host dell'altra rete;
- le macchine in esame sono accese? collegate? hanno firewall?
- utilizzate "tcpdump" direttamente sui due gateway per vedere cosa arriva...
- controllate i vari logs!!!!!

Riferimenti (in ordine semi-sparso!):

- http://www.freeswan.org/freeswan_trees/freeswan-1.91/doc/index.html "FreeS/WAN documentation"

```
 - http://inews.tecnet.it/articoli/Marzo2001/Linux0103.html "Linux + IPsec = Vpn"
   di Piero Baudino"
 - "Virtual Private Network" di Marco Ivaldi su Linux&C. nÝ 19
 - http://www.openbsd.org/faq/faq13.html "Using IPSec - OpenBSD faq"



 ------------------------------------------------
 Autore: goony & haikia
 Data Pubblicazione: Ottobre 01
 Versione Documento: 0.1
 Editor Usato: vi su una macchina OpenBSD 2.8
 Riferimento: goony@inwind.it http://OpenBSD.tzone.it
 ------------------------------------------------


-------------------------------------*END*-----------------------------------------

-------------------------------------[10]-----------------------------------------
-----------------------------------[1/2matto]---------------------------------------
-------------------------------------[SNTP]-----------------------------------------

.............................................................
..........SNTP(Simple Network Time Protocol) su Win2k..........
.............................................................
.......................scritto da m3zz0m4tt0 di #winadmin......
......................................per noflyzone crew......
.............................................................
..
..
.. Chi non si  ricorda i  vecchissimi Personal Computers con le
.. prime  versioni  del  DOS  che all'avviamento  del  sistema
.. chiedevano di inserire la DATA e l'ORA?
.. Forse le cose oggi  sono un po' cambiate,  nei  computers ci
.. sono gli  imprecisi orologi di  sistema che non  costringono
.. gli  utenti  a  dover  digitare  tali  informazioni  ad ogni
.. accensione, lasciando  pero' l'onere  di  supervisionare  al
.. buon funzionamento degli orologi.
..
.. Mi  sto  ponendo una domanda!  come  possiamo  noi imprecisi
.. utenti supervisionare gli imprecisi orologi di sistema?
..
.. Leggendo  qua  e  la  ho  trovato  nel  sito  dell' IETF
.. (Internet Enginering Task Force)  un  RFC  che  potrebbe
.. aiutarci a risolvere queste problematiche.
.. (RFC 2030 Simple Network Time Protocol (SNTP) Version 4
..                       for IPv4, IPv6 and OSI)
..
.. Vediamo ora come utilizzare il  Simple Network Time Protocol
.. su una macchina Windows 2000(perchè proprio il 2000?  perchè
.. è l'uniko Windows che ha saputo affascinarmi).
..
.. innanzitutto  apriamo  la  shell  e  indichiamo  al servizio
.. "Ora di Windows" (lo  potete  trovare  tra gli  strumenti di
.. amministrazione  alla  voce Servizi)  a  quale  server  deve
.. agganciarsi per prelevare l'ora esatta.
..
.. net time /setsntp:tick.usno.navy.mil
..
.. il sistema dovrebbe rispondere con:
..                               esecuzione del comando riuscita
..
.. a  questo  punto  possiamo  richiedere  dalla  shell  una
.. sincronizzazione
..
.. w32tm -once
```

```
..
.. sulla shell windows(o prompt dei comandi per i piu' pignoli)
.. comparira' una cosa tipo:
..
.. W32Time: BEGIN:InitAdjIncr
.. W32Time:    Adj 100144 , Incr 100144  fAdjust 0
.. W32Time: END:Line 2476
.. W32Time: BEGIN:TsUpTheThread
.. W32Time: END  Line 1385
.. W32Time: TimeMMInit()
.. W32Time: Kernel timer : using default maximum resolution
.. W32Time:                MaximumTime = 100144
.. W32Time:                CurrentTime = 100144
.. W32Time: Timer calibrated, looped 1 times
.. [......]
.. W32Time:       END:Line 794
.. W32Time:       END:Line 794
.. W32Time:    END:Line 220
.. W32Time: END:Line 195
.. W32Time: BEGIN:TermTime
.. W32Time:    TimeMMCleanup()
.. W32Time:    BEGIN:FinishCleanup
.. W32Time:       BEGIN:TsUpTheThread
.. W32Time:       END  Line 1385
.. W32Time:       Time service stopped.
.. W32Time:    END:Line 407
..
.. l'ora di Windows è  aggiornata,  non  ci  resta che attivare
.. il relativo servizio per compiere automaticamente quello che
.. noi non  abbiamo mai fatto e che  non inizieremo  mai a fare
..
.. net start w32time
..
.. per sapere su quale  server  la nostra macchina  fa le query
.. basta scrivere
..
.. net time /querysntp
..
..
.. by _1\2MaTto
.. Per contattarmi: mezzomatto@hotmail.com
.. irc.azzurra.it #winadmin #noflyzone


-------------------------------------*END*-------------------------------------

-------------------------------------[11]-------------------------------------
-----------------------------------[[Evil]]-----------------------------------
---------------------------[Comandi base di Linux]---------------------------


        _____________ _____  ____  ______  _____
         ____  ______/ __| | __/ /__/_  _/  _/ /
          __  / __    ____  |  / /  __/ / ___/ /
         __/ _____/     _| |_/ /____/ /____/ /
        __  / ___      ___| |/ / ___/ /_ __/ /_____
       __________/ _________ /  _/____/ _/________/
                                                  .::LinuxBase::.
OoOoOoOoOoOoOoOoOoOoOoOoOoOoOoOoOoOoOoOoOoOoOoOoOoOoOoOoOoOoOoOoOoOoOoO
io non mi ritengo responsabile di come utilizzerete le informazioni
contenute in questa guida in quanto da me offerte a puro scopo
informativo!!
insomma in poche parole NON FATE CAZZATE!
OoOoOoOoOoOoOoOoOoOoOoOoOoOoOoOoOoOoOoOoOoOoOoOoOoOoOoOoOoOoOoOoOoOoOoO

questa guida è copyright di [Evil]
www.evil87.cjb.net
```

Saluti: ReSiNaRo , NoFlyZone-Crew , Ness1 , Do^Sh1n , Tommy_ ,

Fuck To: lamerz

Linux intro

Dedicato a tutti gli abitanti del pianeta windows:
Linux è un OS (Operative System) molto diverso da windows , difatti
a gli utenti di windows è SCONSIGLIATO , installare linux e disinstallare windows!
insomma che ha di diverso linux?
linux a differenza di windows ha puntato molto sulla versabilità del
sistema e sulla sicurezza , difatti in linux con accesso root potete
"modellare" linux a vostro piacimento , solo che non sono dei semplici
doppio click a farvi fare quello che volete , ma dovrete dare comandi
"unix" a linux , che verranno tradotti in linguaggio macchina . E' come
lavorare con windows usando SOLO il dos , solo che è più complicato...
qui di seguito riporter= i comandi più usati con linux ...
P.S prima di usare linux leggete più guide possibili riguardanti questo
OS e incominciate a studiarvi un p= come funzia il dos e telnet (telnet
usa comandi unix)..

Comandi

---

cd sintassi: cd dyrectory 'porta alla directory specificata
-
cd .. riporta alla directory precedente
-
pwd indica la directory remota
-
clear pulisce lo schermo
-
compress sintassi: compress [-v] file 'comprime un file = .Z
-
zcat decomprime un file
-
cp sintassi: cp file directory 'copia un file
-
date da informazioni sulla data
-
ls mostra contenuto directory remota
-
joe editor di testo
-
df informa sullo spazio rimanente sul disco rigido
-
du informa sullo spazio occupato dai vostri file
-
history mostra l'elenco dei comandi usati
-
id informa sul propio id
-
less sintassi: less file 'mostra il contenuto di un file
-
man sintassi: man comando 'mostra informazioni sul comando
-
mkdir sintassi: mkdir directory 'crea nuova directory
-
netstat da informazioni sullo stato della rete
-
pico sintassi: pico file 'editor di testo
-
rm sintassi: rm file 'elimina file

```
-            -
rmdir      sintassi: rmdir directory    'elimina directory
-            -
vi         editor di testo un p= complicato

Spazio     Avanza di una pagina
-
Invio      Avanza di una linea.
-
b          Si sposta indietro di una pagina.
-
man comando   Mostra il manuale corrispondente al comando inserito


/quit Evil rulez


-------------------------------------*END*-------------------------------------

-------------------------------------[12]-------------------------------------
-------------------------------[NoFlyZone Staff]--------------------------------
---------------------------------[Greetings]----------------------------------

Rubo la parola a Vicio dal momento che sono l'impaginatore del numero e quindi
posso scrivere ciò che voglio prima degli altri:-PP:-))
Questo è il primo numero: ci sono articoli nuovi e rivisitazioni di altri...
credo sia un buon lavoro, anche se ci manca ancora una certa maturità generale.
Spero segua presto il numero 2 con tanti nuovi articoli,e, a tal proposito,
invito tutti quelli interessati a contattarci sul canale IRC di azzurra
se si sentono in grado di contribuire! Ben venga gente nuova,ma mi raccomando,
preparati;-)(ricordate prima...lamah,3l33t >>dev\null)
Passo la parola al boss della crew(che forse è già incazzato dal momento che
tutto questo mio parlare non era in programma!:-)))
A presto raga...<<<<Hack The Planet>>>>> CiTyHunet

Re:Lordvicio

Il solito kiakkierone :-),nn mi dilungherò molto...volevo solo fare i dovuti
ringraziamenti....

1 A tutti coloro ke hanno letto l'e-zine
2 A tutti coloro ke hanno seguito la crew da esterni
3 A tutto il chan #NoFlyZone
4 A tutti i membri della crew,ke hanno fatto un lavoro straordinario ..
5 A tutti coloro ke nn ho citato :P


P.S. Quasi dimenticavo di dirvi dove venire a trovarci:-)

www.noflyzone-crew.cjb.net

IRC:    irc.azzurra.it          port: 6667 chan: #noflyzone
        arkshrine.serverirc.com port: 6667 chan: #noflyzone

Con questo finisco davvero di rompervi!:-)
A presto:-)


Mode E-zine OFF.



-------------------------------------*END*-------------------------------------:(


NoFlyZone StaFf!:-)
```