Itis Enea Mattei Sondrio

ESAME DI STATO A.S. 2010/2011

PHP e gestione della sicurezza in Rete

Sviluppo di un sito web

CLASSE VE - INFORMATICA

Andrea Polini **Pietro Caspani**

Indice

Introduzione

In un'era in cui Internet si pone come principale mezzo di comunicazione e di informazione, è necessario aumentare la coscienza critica dell'utente che giorno dopo giorno si trova a contatto con la rete al fine di consentirgli una navigazione in sicurezza, priva di rischi.
A tale scopo si è deciso di trattare l'argomento "Sicurezza informatica", la quale si presenta come requisito fondamentale per garantire agli utenti Internet una navigazione sicura.

Senza delle garanzie sulla sicurezza quanti sarebbero gli utenti che quotidianamente navigherebbero su Internet?

Probabilmente gli utenti sarebbero in numero minore rispetto agli 1,97 miliardi rilevati dalla società svedese Royal Pingdom nel giugno 2010. Questo perché Internet è oggi sinonimo non solo di "Social Network" e quindi di comunicazione ma anche di "vendite online" le quali, in forte crescita, consentono all'utente domestico di acquistare dei prodotti a prezzi vantaggiosi comodamente dalla poltrona di casa.
Nel momento in cui i pagamenti avvengono direttamente online diventano quindi fondamentali delle policy che tutelino i clienti al fine di incoraggiare l'utente ad acquistare online.
Non meno rilevante è il problema della privacy: ognuno di noi almeno una volta registrandosi ad un sito Internet si è chiesto dove i propri dati personali potessero finire, se in qualche modo qualcuno potesse spiarci e se le nostre password fossero realmente sicure.

E' possibile che la profezia di Orwell si stia avverando?
Siamo destinati a vivere in una società da Grande Fratello?

Allo scopo di mostrare le principali implementazioni di sicurezza di un sito Internet è stato sviluppato un sito web che consente la vendita di prodotti online.
Il linguaggio di programmazione lato server utilizzato è PHP e nella parte conclusiva del lavoro verranno riportati alcuni segmenti di codice significativo impiegati nello sviluppo del sito.

Principali obiettivi della sicurezza informatica

La sicurezza informatica punta solitamente a cinque obiettivi principali :

- *L' integrità: la protezione dei dati e delle informazioni nei confronti delle modifiche del contenuto, accidentali oppure effettuate da terze parti;*

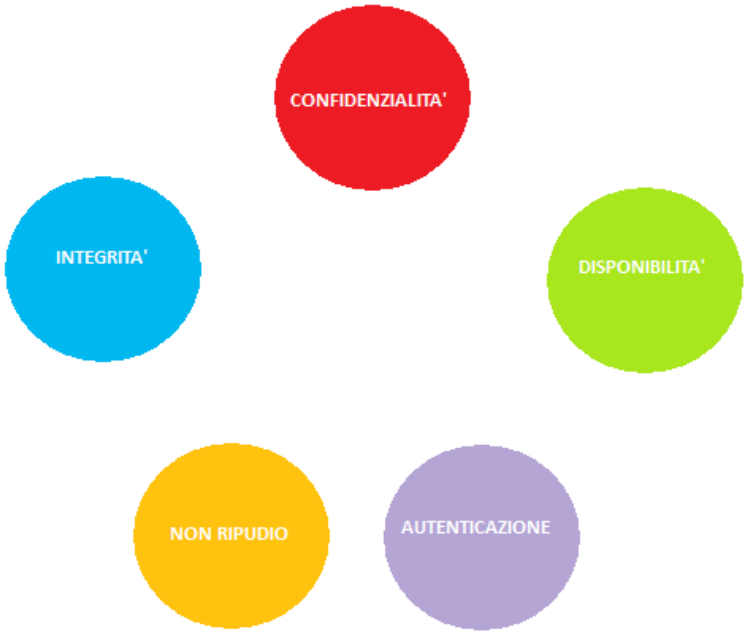
- *La Confidenzialità: si intende la protezione dei dati e delle informazioni scambiate tra un mittente e uno o più destinatari nei confronti di terze parti. Diventa particolarmente importante garantire questa proprietà quando il sistema utilizzato è insicuro, come ad esempio la rete internet. In un sistema che garantisce la confidenzialità, una terza parte che entra in possesso delle informazioni scambiate tra mittente e destinatario non è in grado di ricavarne alcun contenuto informativo, questo perché si ricorre a meccanismi di cifratura. Non esistono però meccanismi di protezione sicuri in assoluto;*

- *La disponibilità: le risorse informatiche e le informazioni sono accessibili agli utenti autorizzati nel momento in cui servono;*

- *Il non ripudio: grazie al PKI(Public key Infrastracture) è possibile fornire la prova incontestabile di una avvenuta spedizione o di una avvenuta ricezione di dati in rete. Il non ripudio assume due modalità:*

    1. *non ripudio della sorgente: prova chi è il mittente dei dati in una transazione;*
    2. *non ripudio della destinazione: prova che i dati sono arrivati ad uno specifico destinatario*

- *L'autentificazione, che consiste nell'assicurare che solo le persone autorizzate abbiano acceso alle risorse.*



Gestione del rischio

Oltre agli obiettivi di sicurezza è bene introdurre nella trattazione altri termini comunemente utilizzati e poi analizzarli separatamente:

- *Beni da proteggere*
- *Minacce alla sicurezza*
- *Vulnerabilità del sistema informatico*
- *Impatto causato dall'attuazione della minaccia*

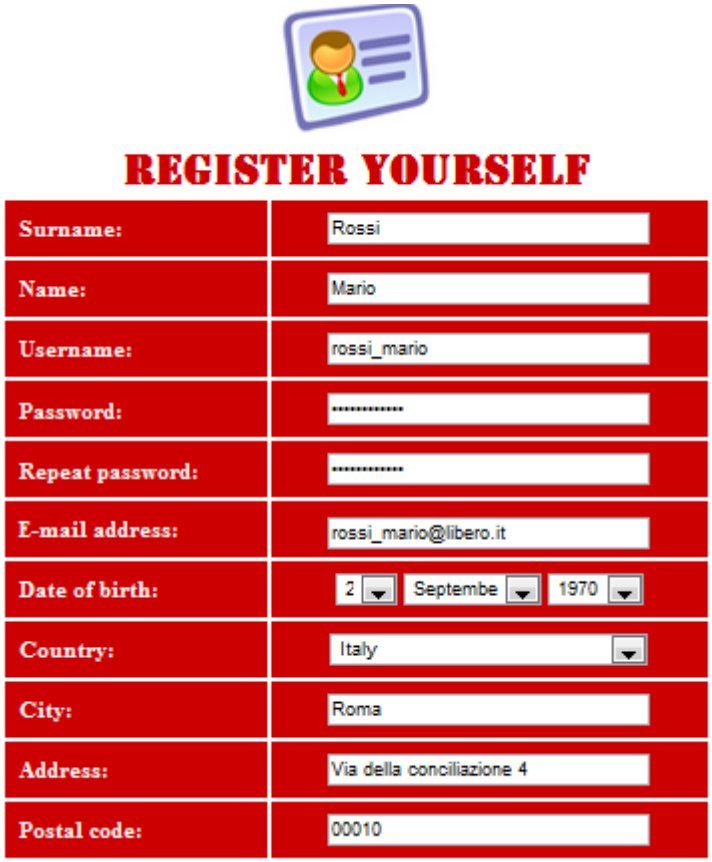
Beni da proteggere

Il bene è qualsiasi cosa materiale o immateriale che debba essere protetto: in campo informatico tra i beni di un ente ci sono le risorse informatiche, il personale, le informazioni, la documentazione, l'immagine dell'Ente.
Per un individuo, tra i beni da proteggere, ci sono oltre alle risorse informatiche, anche le informazioni personali e la privacy.
I beni però non sono tutti uguali e possono essere distinti in:

- *Beni primari: si tratta dei dati veri e propri*
- *Beni che servono per proteggere i beni primari: Password*
- *Altri esempi di beni secondari sono le attrezzature che consentono all'hardware di funzionare con continuità e sicurezza: gruppi di continuità, condizionatori, alimentatori.*

Generalmente le informazioni pubblicate su un sito web richiedono disponibilità e integrità ma non riservatezza, proprietà che invece è richiesta dalle password.
Quando avviene la registrazione ad un sito internet, i dati personali vengono memorizzati in un database, e tra i dati personali vengono comprese anche la password.
Per evitare che gli stessi amministratori del sito oppure terze parti che riescono ad infiltrarsi nel sistema possano sfruttare le generalità degli utenti con scopi fraudolenti è necessario che le password vengano criptate sfruttando degli algoritmi di crittazione.
Nel caso specifico si va ad analizzare l'algoritmo SHA-1 implementato due volte per porre rimedio ad alcuni limiti individuati recentemente da alcuni studiosi.

Quello che segue è un esempio di registrazione ad un sito:



Dopo aver acconsentito ai termini di sicurezza del sito la registrazione è avvenuta

Ciò che appare nel Database è questo:

+ Opzioni

	id	cognome	nome	username	mail	nascita	coins	stato	citta	indirizzo	codicepostale	password
	18	Rossi	Mario	rossi_mario	rossi_mario@libero.it	1970-09-25	2	Italy	Roma	Via della conciliazione 4	10	4297653b530342f12ece8f45b310ab8b3443f022

Seleziona tutti / Deseleziona tutti *Se selezionati:*

Il campo Password risulta criptato: 4297653b530342f12ece8f45b310ab8b3443f022

Algoritmo SHA-1 (Secure Hash Algorithm -1)

SHA-1 è una funzione hash crittografica la cui specifica originale fu pubblicata nel 1993 dal NIST (National Institute of Standards and Technology). La versione del 1993 è però indicata spesso con SHA-0 dato che è stata rivista nel 1995 e in quell'anno ne è stata pubblicata la nuova versione: SHA-1.

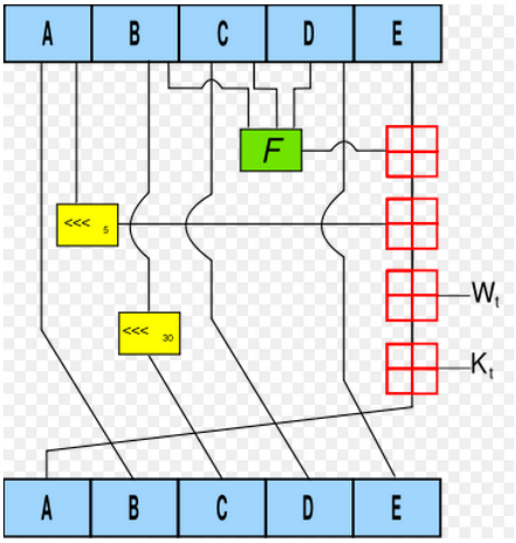
Funzionamento:

***Passo 1** (Imbottitura): Al messaggio originale vengono aggiunti dei bit di "imbottitura" affinché la lunghezza finale del messaggio risulti congruente a 448 (mod 512), così facendo la lunghezza di "messaggio + imbottitura" è pari ad un numero 64bit più piccolo di un multiplo di 512bit.*

***Passo 2** (Aggiunta lunghezza): Alla sequenza di bit (messaggio + imbottitura) creata durante il passo 1 viene aggiunto un intero unsigned di 64bit contenente la lunghezza del messaggio originale. Alla fine di questi primi due passi otteniamo una sequenza di bit che è un multiplo di 512.*

***Passo 3** (Inizializzazione del buffer MD): Un buffer di 160bit suddiviso in 5 registri da 32bits ciascuno viene creato per la memorizzazione di alcuni passaggi intermedi. I 5 registri verranno convenzionalmente indicati con (A,B,C,D,E) ed inizializzati con i seguenti valori esadecimali:*

1. *A = 67452301*
2. *B = EFCDAB89*
3. *C = 98BADCFE*
4. *D = 10325476*
5. *E = C3D2E1F0*



***Passo 4** (Elaborazione dei blocchi da 512bit): La sequenza di bit "messaggio+imbottitura+lunghezzaMessaggio" viene divisa in blocchi da 512bit, che identificheremo con B_n con n che va da 0 a L. Il fulcro dell'algoritmo SHA-1 è chiamato compression function ed è formato da 4 cicli di 20 passi ciascuno.*
I cicli hanno una struttura molto simile tra di loro se non per il fatto che utilizzano una differente funzione logica primitiva. Ogni blocco viene preso come parametro di input da tutti e 4 i cicli insieme ad una costante K e i valori dei 5 registri. Alla fine della computazione otterremo dei nuovi

valori per A,B,C,D,E che useremo per la computazione del blocco successivo sino ad arrivare al blocco finale L.

Quello che segue è lo Pseudocodice SHA-1:

```
h0 = 0x67452301
h1 = 0xEFCDAB89
h2 = 0x98BADCFE
h3 = 0x10325476
h4 = 0xC3D2E1F0

   for i from 16 to 79
      w[i] = (w[i-3] xor' w[i-8] xor w[i-14] xor w[i-16]) ' 1

   a = h0
   b = h1
   c = h2
   d = h3
   e = h4

   Main loop:
   for i from 0 to 79
      if 0 ≤ i ≤ 19 then
         f = (b and c) or ((not b) and d)
         k = 0x5A827999
      else if 20 ≤ i ≤ 39
         f = b xor c xor d
         k = 0x6ED9EBA1
      else if 40 ≤ i ≤ 59
         f = (b and c) or (b and d) or (c and d)
         k = 0x8F1BBCDC
      else if 60 ≤ i ≤ 79
         f = b xor c xor d
         k = 0xCA62C1D6

      temp = (a leftrotate 5) + f + e + k + w[i]
      e = d
      d = c
      c = b leftrotate 30
      b = a
      a = temp

   Add this chunk's hash to result so far:
   h0 = h0 + a
   h1 = h1 + b
   h2 = h2 + c
   h3 = h3 + d
   h4 = h4 + e
digest = hash = h0 append h1 append h2 append h3 append h4
```

Minacce alla sicurezza

In campo informatico quando si parla di minaccia si intende un'azione potenziale, accidentale o deliberata che può portare alla violazione di uno o più obiettivi di sicurezza. Le minacce informatiche si possono classificare secondo la loro origine:

- *Naturale*
- Umana

Naturale

La minaccia di tipo naturale è quella causata da fattori atmosferici come può essere un *allagamento per forti piogge che può interrompere la disponibilità dei servizi informatici. Per fronteggiare tale evenienza bisogna prevedere un piano di Disasterer Recovery.*
Per Disaster Recovery (brevemente DR) si intende l'insieme di misure tecnologiche e organizzative atte a ripristinare sistemi, dati e infrastrutture necessarie all'erogazione di servizi di business a fronte di gravi emergenze. Si stima che la maggior parte delle grandi imprese spendano fra il 2% ed il 4% del proprio budget IT nella pianificazione della gestione dei disaster recovery, allo scopo di evitare perdite maggiori nel caso che l'attività non possa continuare a seguito della perdita di dati ed infrastrutture IT. Delle imprese che hanno subito disastri con pesanti perdite di dati, circa il 43% non ha più ripreso l'attività, il 51% ha chiuso entro due anni e solo il 6% è riuscita a sopravvivere nel lungo termine. I disastri informatici con ingenti perdite di dati nella maggioranza dei casi provocano quindi il fallimento dell'impresa o dell'organizzazione, ragion per cui investire in opportune strategie di recupero diventa una scelta quasi obbligata.
Il Disaster Recovery Plan (DRP) (in italiano, Piano di disaster recovery) è il documento che esplicita tali misure. Esso fa parte del più ampio Business Continuity Plan (BCP).
Affinché una organizzazione possa rispondere in maniera efficiente ad una situazione di emergenza, devono essere analizzati:

- *I possibili livelli di disastro*
- *La criticità dei sistemi/applicazioni.*

Per una corretta applicazione del piano, i sistemi devono essere classificati secondo le seguenti definizioni:

- *Critici*

Le relative funzioni non possono essere eseguite senza essere sostituite da strumenti (mezzi) di caratteristiche identiche. Le applicazioni critiche non possono essere sostituite con metodi manuali. La tolleranza in caso di interruzione è molto bassa, di conseguenza il costo di una interruzione è molto alto.

- *Vitali*

Le relative funzioni possono essere svolte manualmente, ma solo per un breve periodo di tempo. Vi è una maggiore tolleranza all'interruzione rispetto a quella prevista per i sistemi critici, conseguentemente il costo di una interruzione è inferiore, anche perché queste funzioni possono essere riattivate entro un breve intervallo di tempo (generalmente entro cinque giorni).

- *Delicati*

Queste funzioni possono essere svolte manualmente, a costi tollerabili, per un lungo periodo di tempo. Benché queste funzioni possano essere eseguite manualmente, il loro svolgimento risulta comunque difficoltoso e richiede l'impiego di un numero di persone superiore a quello normalmente previsto in condizioni normali.

- *Non-critici*

Le relative funzioni possono rimanere interrotte per un lungo periodo di tempo, con un modesto, o nullo costo per l'azienda, e si richiede un limitato (o nullo) sforzo di ripartenza quando il sistema viene ripristinato.
Le procedure applicative, il software di sistema ed i file che sono stati classificati e documentati come critici, devono essere ripristinati prioritariamente. Applicazioni, software e file classificati come critici hanno una tolleranza molto bassa alle interruzioni. La criticità di applicazioni, software di sistema e dati, deve essere valutata in funzione del periodo dell'anno in cui il disastro può accadere. Un piano d'emergenza deve prevedere il ripristino di tutte le funzioni aziendali e non solo il servizio ICT centrale. Per la definizione del DRP devono essere valutate le strategie di ripristino più opportune su: siti alternativi, metodi di backup, sostituzione degli equipaggiamenti e ruoli e responsabilità dei team. La prolungata indisponibilità del servizio elaborativo derivante in particolare situazione di disastro, e quindi dei servizi primari, rende necessario l'utilizzo di una strategia di ripristino in sito alternativo.

Umana

Le minacce di questo tipo sono quelle con una deliberata origine umana. Il pc attaccato può cadere sotto il controllo esterno e non essere completamente disponibile per il proprietario.
Un recente studio sulle minacce alla sicurezza informatica condotto da Microsoft su un campione composto da oltre seicento milioni di pc in 117 paesi del mondo, ha rivelato che il veicolo di propagazione maggiore dei virus, nel 2010, sono stati gli ormai diffusissimi social network, ambienti sempre più frequentati in cui gli utenti vengono attaccati dal cosiddetto social engineering malware. Si tratta di applicazioni che si celano dietro un semplice link condiviso tra amici, che trattano temi attuali che inducono l'ignara vittima ad aprire il link stesso. Recentissimo è il caso di alcuni link che promettevano di aprire un video sulla morte di Osama bin Laden o immagini esclusive sul Royal

Wedding e che invece non facevano altro che attivare il codice maligno. Un altro tipo di attacco molto in voga lo scorso anno è stato quello chiamato Adware. In sostanza sono dei semplicissimi programmi che aprono un numero elevatissimo di volte banner pubblicitari rallentando la navigazione e setacciando il pc alla ricerca di dati personali. In Italia il più diffuso è il PornPop (con il 16% di attacchi), che si attiva navigando su alcuni siti per soli adulti. I danni per la privacy sono ovvi ma ben peggiori sono quelli che potrebbero derivare dall'eventuale furto di codici d'accesso a conti bancari o per l'utilizzo di carte di credito. Gli utenti spesso tendono a sottovalutare questo tipo di attacchi, il più delle volte facilmente prevenibili, semplicemente aggiornando antivirus e browser utilizzato per le ricerche sul web.

Questi virus per social network dal punto di vista tecnico sono molto semplici e tipici di qualche anno fa. Con l'impressionante crescita di siti come Facebook, Twitter ed altri, sono tornati in voga e, nel solo 2010, la percentuale di attacchi di questo tipo è balzata dall'8 all'84 per cento. L'unica spiegazione possibile è che la grande quantità di nuovi utenti che arrivano online trascinati dalla voglia di restare in contatto con gli amici e con i propri cari, non sono particolarmente preparati sui fondamentali della sicurezza. I social network vivono sulla fiducia tra gli utenti di cui i malintenzionati approfittano.

Un altro problema molto attuale e che mette in serio pericolo la privacy degli utenti è sicuramente la protezione delle reti wireless. Questo tipo di rete è ormai molto diffuso vista la sua comodità nel collegare senza fili diversi terminali ad internet. Il tentativo di accesso a tali reti si può classificare come minaccia umana ed è probabilmente la meno conosciuta pur essendo di fatto una delle più pericolose. Un malintenzionato accedendo alla rete wireless di un qualsiasi utente può nella migliore delle ipotesi utilizzarla per navigare in internet a spese dell'utente stesso. Ben peggiore è il caso in cui questa persona utilizza la rete per commettere reati a cui poi dovrà rispondere l'intestatario dell'abbonamento in quanto responsabile degli accessi. Un altro problema è che chi compie queste azioni può tenere sotto controllo tutto ciò che passa sulla rete, comprese ricerche sul web effettuate dall'ignara vittima e conversazioni private, per esempio quelle in chat, oltre alla possibilità concreta di ottenere password da utilizzare poi per avere accesso ad ulteriori dati riservati.

I malintenzionati, anche in questo caso, sfruttano le scarse conoscenze tecniche delle loro vittime. Il caso più semplice è quello delle reti non protette da password. In questa evenienza chiunque può accedere alla rete senza nessun tipo di problema.

Più complicato, ma non impossibile, è violare una rete protetta da password.

Un client che intenda connettersi ad una rete Wi-fi deve innanzitutto le caratteristiche dell'access point cui collegarsi (deve quindi localizzarlo) ed in secondo luogo connettersi superando la procedura di autenticazione prevista.
Le metodologie definite dello standard 802.11 affinché un client possa localizzare un access point sono due, che potremmo definire rispettivamente come metodo passivo ed attivo: tramite beacon frame inviati a cura dell'access point oppure a mezzo di frame di probe request/response scambiati dalle parti.
Al riguardo del primo metodo, gli access point inviano di default frame beacon ad intervalli regolari (ogni decimo di secondo) al fine di palesare la loro presenza e di presentare le proprie caratteristiche, in funzione di un facile riconoscimento soprattutto da parte dei nuovi host. L'interfaccia di rete di ogni host capta questi frame.
Gli host che già conoscono i parametri di configurazione si connetteranno automaticamente attraverso il proprio programma di network manager o file di configurazione che sia.
I beacon frame contengono (nel loro body): nome della rete e MAC dell'access point (ESSID/BSSID), canale, standard 802.11xx e rate di trasmissione supportato dall'access point, timestamp, parametri relativi al livello fisico (DSSS,FHSS,..), cifratura usata, e via dicendo: sono informazioni assolutamente necessarie affinché un client possa connettersi all'access point.
Per ciò che concerne il secondo metodo, gli host inviano richieste "sonda", di tipo diretto o broadcast, alle quali l'AP risponderà con pacchetti contenenti informazioni analoghe a quelle viste.
In entrambi i metodi è possibile che l'access point, se istruito, non si comporti come detto.
Una volta che sia stato definito, da parte dell'host client, l'access point cui collegarsi, il client medesimo invia una richiesta di autenticazione ad esso. Il processo tramite il quale il dispositivo di accesso autentica il client dipende dall'opzione utilizzata, Open authentication, WEP o WPA, e, negli ultimi due casi prevede un four-way handshake.

L'assenza di connessione fisica tra i dispositivi facenti parte delle rete wireless rende obbligatorio un sistema di autenticazione e scambio dati sicuro, al fine di soddisfare ai requisiti minimi di confidenzialità ed integrità dei dati trasmessi: se in una rete cablata è il cavo stesso a confinare i segnali elettromagnetici entro un perimetro definito (onde guidate), così non è per le onde elettromagnetiche irradiate in aria dalle antenne, che, potenza permettendo, potrebbero essere intercettate da chiunque disponesse di un apparato di ricezione compatibile con le specifiche elettriche di progetto.
In caso di autenticazione avvenuta con successo, l'AP invia una authentication response che indica l'avvenuta autenticazione. Subito dopo, l'host invia una richiesta di associazione, contenente obbligatoriamente l'identificativo dell'access point (SSID) cui intende connettersi ed informazioni su di esso (ad esempio data rate supportati). Nella successiva risposta di associazione dell'access point sono contenuti alcuni parametri, dei quali d'interesse è l'ID di associazione.

WEP (Wired Equivalent Privacy)

È il vecchio protocollo standard 802.11 utilizzato per autenticare i partecipanti, che condividono un medesimo segreto (password o chiave privata che dir si voglia), e cifrare conseguentemente le

trasmissioni che intercorrono tra access point e schede di rete wireless degli host in una rete Wi-Fi, a livello quindi data link.

I passi seguiti per autenticare una stazione Wi-Fi client all'access point seguono il 4-way handshake a sfida:

- *l'host client invia la richiesta di autenticazione all'access point;*
- *l'access point risponde con un frame di challenge o sfida;*
- *l'host ritorna all'access point il challenge cifrato con la chiave segreta;*
- *l'access point decifra il frame con la medesima chiave segreta condivisa e lo confronta con la sfida inviata.*

Se i due risultano uguali allora l'access point comunica all'host che l'autenticazione ha avuto successo; viceversa lo informa che la connessione gli viene negata.

Si noti che non è richiesta autenticazione dell'access point, quindi il client non ha la reale certezza di esser connesso all'access point legittimo!
Le chiavi WEP hanno lunghezza da 40 bit (5 caratteri soltanto!) a 104 bit e vengono usate, nel modo che vedremo a breve, per cifrare tutto il traffico scambiato tra gli end-point.
Wired Equivalent Privacy ha centrato appieno l'obiettivo postosi di rendere sicure le trasmissioni wireless quanto quelle via cavo (Ethernet): per nulla! In realtà, la sicurezza dei dati trasmessi via etere è anche peggiore, dacché entrare a far parte di una rete LAN cablata scassinando la porta di ingresso di un ufficio, per un attaccante esterno, è qualcosa più complicato e rischioso che starsene comodamente nel parcheggio sotto di esso con un'antenna fatta con le confezioni delle patatine puntata nel verso dell'access point.

WPA e WPA2

Molto simili ma differenti per motivi storici, sono il risultato riuscito del tentativo di colmare le lacune del progenitore WEP. WPA è stato rilasciato dalla Wi-Fi Alliance nella fretta di rendere sicura una wireless nel più breve tempo possibile, mentre l'802.11i o WPA2 è da intendersi oggigiorno come standard ufficiale.
Per l'autenticazione degli host, WPA e WPA2 prevedono due modalità: la prima, enterprise, mediante server di autenticazione RADIUS (WPA-RADIUS e WPA2/RADIUS o semplicemente WPA e WPA2 rispettivamente), che distribuisce differenti chiavi agli utenti, e la seconda, personal, tramite una (unica) chiave segreta condivisa, chiamata pre-shared key (PSK, da cui WPA/PSK e WPA2/PSK rispettivamente), modalità di gran lunga più adatta per ambienti di small office e casalinghi come suggerito dallo stesso nome.
WPA utilizza TKIP (Temporal Key Integrity Protocol), protocollo crittografico che a sua volta utilizza RC4, analogamente al WEP, ma con chiavi a 128 bit ed un vettore di inizializzazione (IV) a 48bit, che costituirebbero da soli un passo innanzi non da poco.
L'aspetto più importante, tuttavia, è che, oltre a variare la chiave di cifratura di ogni pacchetto, TKIP contiene un meccanismo di variazione continua della chiave segreta condivisa ad ogni nuova associazione host-access point. Il condividere la chiave PSK con altri host ed utilizzare sempre la medesima costituiva un grosso problema per la tecnologia WEP (non che essa avesse particolari punti d'eccellenza, s'intende..).

WPA inoltre utilizza un sistema di "firma" (CRC) del payload più sofisticato e sicuro che quanto utilizzato dal WEP, che permetteva addirittura, tramite attacco man-in-the-middle, di modificare il contenuto di un frame dati e ricalcolarne il CRC.

Nota la teoria delle comunicazioni wireless ci occuperemo ora, per puro esempio didattico, di come sia semplice entrare a far parte di una rete in modo illegittimo, sfruttando per lo più le (enormi) falle di sicurezza dei protocolli più vecchi (WEP) e le possibili configurazioni approssimative degli access point, compiute da chi installa la rete, anche in presenza dei nuovi protocolli (WPA e WPA2), ciò che vedremo successivamente.

Brute Force

Oltre ad esser stato inizialmente dotato, come detto, di chiavi molto piccole (40 bit massimi effettivi, cioè 5 caratteri se consideriamo una normale codifica ad 8 bit per carattere), che hanno fatto la felicità dei cracker amanti degli attacchi di forza bruta, il protocollo ha mostrato di possedere falle (più d'una) nella sua stessa logica, specie nell'implementazione dell'algoritmo di cifratura RC4, che utilizza internamente.

Attacchi statistici

Dire che un protocollo presenta vulnerabilità significa che esso è passibile di attacchi di criptoanalisi, che risultano essere (quasi) indipendenti dalla lunghezza delle chiavi utilizzate.
Il fulcro del problema non è l'algoritmo RC4 in sé, che rimane sicuro, ma il modo in cui viene utilizzato nel contesto al fine di creare l'encryption key di ogni messaggio scambiato: quando WEP utilizza RC4 per cifrare un pacchetto dati, utilizza una stringa o vettore di inizializzazione (IV) di dimensione molto piccola (24 bit, cioè soltanto 2^24, ovvero circa 16 milioni di combinazioni), scambiata in chiaro tra gli end point.

Siano:

- ***key**: la chiave segreta WEP condivisa;*
- ***IV**: il vettore di inizializzazione di 24 bit;*
- ***msg**: il messaggio da scambiare (messaggio in chiaro);*
- ***ch(M)**: il checksum del messaggio in chiaro.*

Il messaggio cifrato scambiato sarà: Cmsg = [msg+ch(msg)]XOR[RC4(Key+IV)]

("+" indica qui l'operatore di concatenazione tra stringhe).

La chiave privata condivisa concatenata con il vettore di inizializzazione formano l'encryption key, che, "codificato" tramite RC4, cifra il messaggio in chiaro ed il suo checksum. Assieme ad esso (Cmsg) viene scambiato in chiaro anche l'IV.
Varie vulnerabilità del protocollo sono state nel tempo trovate ed altrettanti tool creati al fine dell'exploit tramite esse, tool che si basano per la maggior parte sull'asserto che segue. Data la

forma del messaggio scambiato, è possibile che alcuni bit del keystream dipendano da alcuni bit dell'encryption key: se questi pacchetti vengono sniffati (e salvati in un file) è quindi possibile ottenere la chiave segreta attraverso attacchi di analisi statistica (conoscendo sempre il vettore di inizializzazione, essendo inviato in chiaro). Il tutto in pochi minuti, come vedremo a breve.

Chiavi statiche

Un'altra problematica da considerare è l'assenza di una politica di gestione delle chiavi: più host che condividono la medesima chiave privata per lunghi periodi di tempo, implicano la possibilità di venirne a conoscenza da parte di esterni se qualche host viene in qualche modo compromesso o rubato. Questa problematica è ovviamente di validità generale in simili contesti.

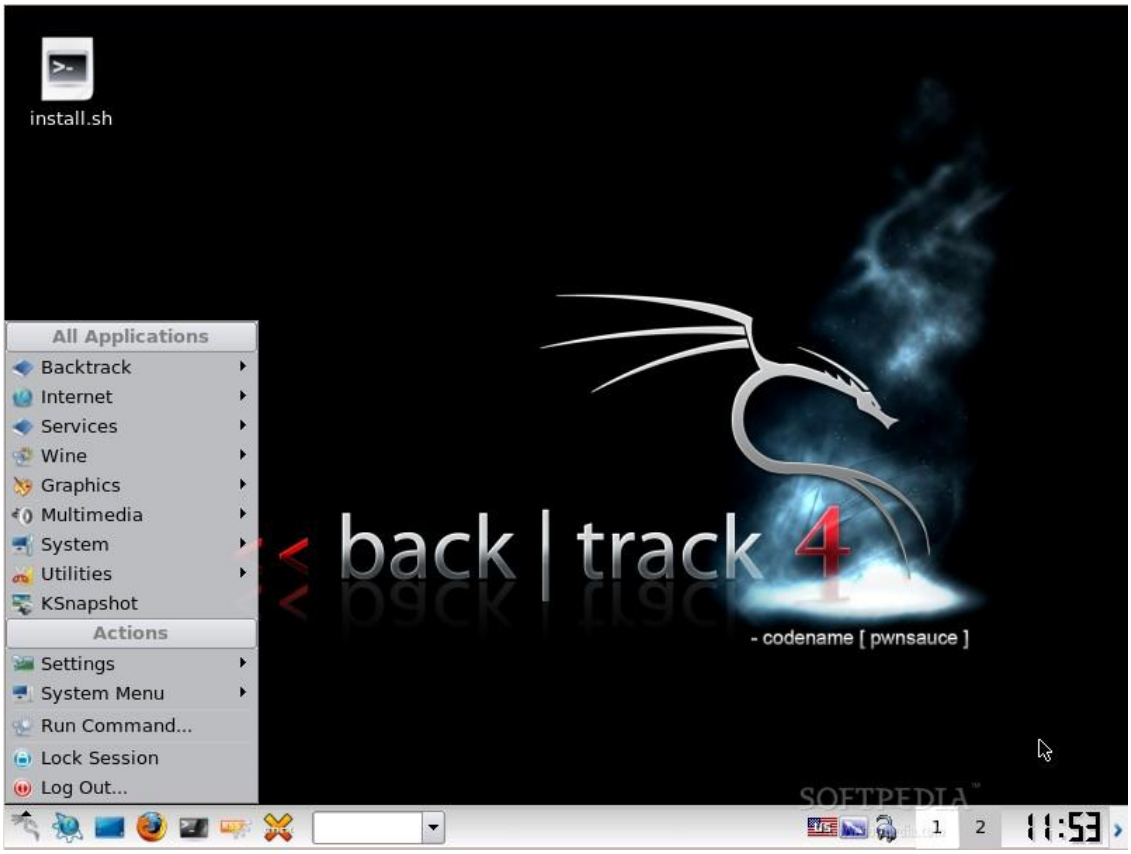
Tutto questo dal punto di vista teorico. A questo punto la domanda è sorta spontanea. Possibile che si possa davvero, con strumenti totalmente gratuiti, riuscire ad ottenere la password di una rete wireless protetta senza che il proprietario si accorga di nulla? Ovviamente ci abbiamo provato e in effetti con alcune conoscenze informatiche e un po' di pazienza si può fare. Occorre precisare che per effettuare questi test abbiamo utilizzato una rete di nostra proprietà, in quanto è assolutamente illegale tentare di ottenere la password di una rete wireless senza il consenso del proprietario.
Possiamo affermare dopo le nostre prove che le reti protette con il vecchio protocollo WEP sono veramente insicure: con i giusti programmi, un minimo di competenza tecnica ed un po' di pazienza, è possibile entrare in questo tipo di infrastrutture senza grossi problemi, anche in caso si stesse utilizzando un PC molto datato.

Il discorso è completamente diverso per le reti protette da WPA oppure WPA2: le possibilità di riuscire a forzare una rete wireless che utilizzi uno di questi sistemi è decisamente ridotta. Per la precisione, nel corso delle prove è emerso chiaramente che non è possibile per ora accedere ad una rete WPA/WPA2 configurata per utilizzare una chiave di rete sufficientemente robusta.

E ' quindi altamente consigliato l'utilizzo di una password di una certa lunghezza, composta da caratteri alfanumerici che non abbiano un senso logico. L'errore di molte persone è scegliere come chiave di rete una parola sensata. A questo punto anche le reti protette da wpa e wap2 diventano vulnerabili in quanto online si possono recuperare, per ogni lingua, dei veri e propri dizionari con decine di migliaia di password spesso utilizzate dagli utenti.

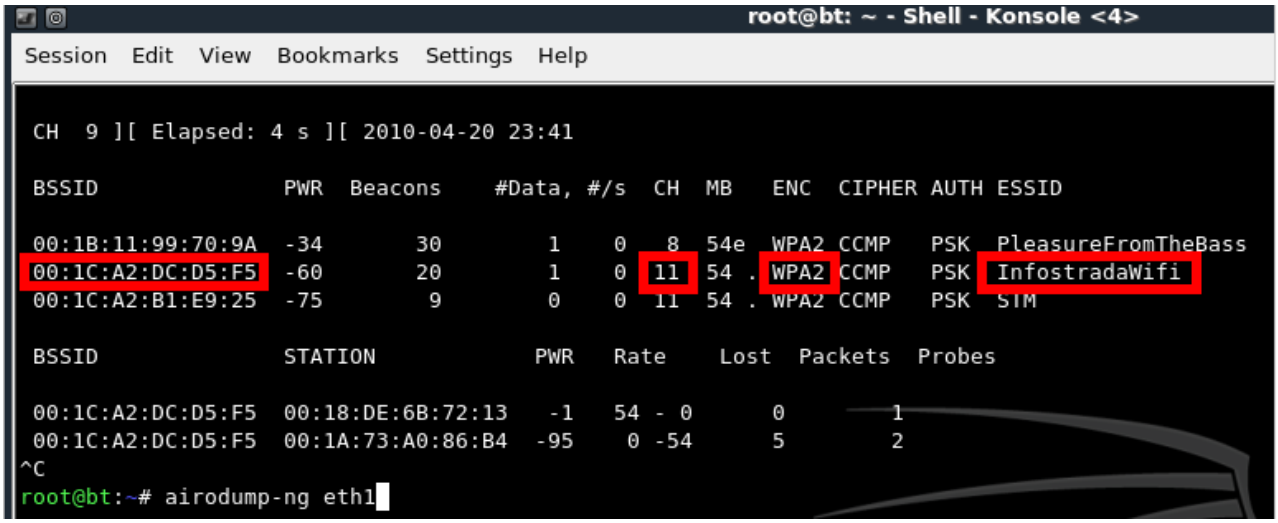
Passiamo quindi alla prova pratica:

Per raggiungere il nostro scopo, dovremo utilizzare una distribuzione Linux appositamente progettata per testare la sicurezza delle reti: lo strumento si chiama Backtrack che è liberamente scaricabile. BackTrack si presenta così:

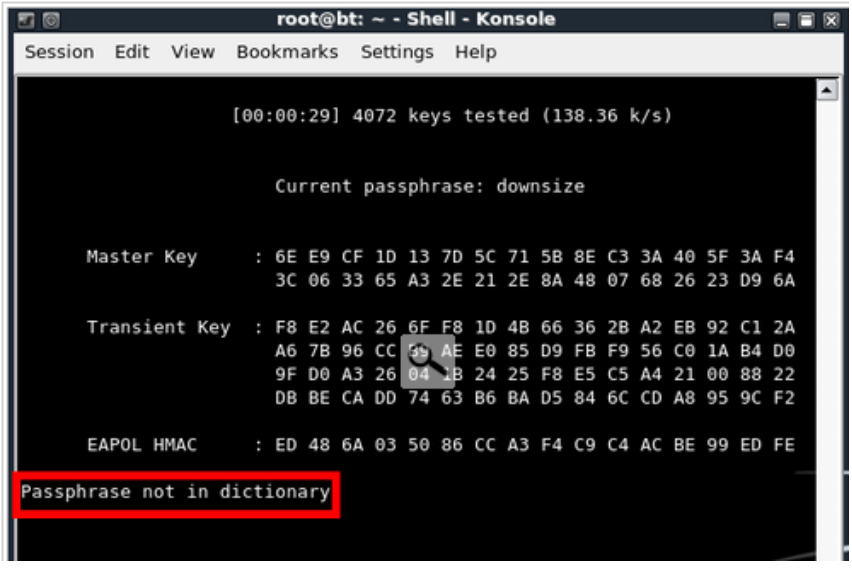


Il procedimento verrà descritto a grandi linee per evitare che qualcuno possa utilizzare questo approfondimento per scopi non legali.
Dopo una serie di operazioni di configurazione della scheda wifi che deve essere impostata in modalità di monitoraggio, si passa alla raccolta delle informazioni sulle reti che la scheda stessa rileva.
Si ottiene una schermata con queste informazioni:

- *BSSID: è l'indirizzo fisico dell'access point. CH (sesta colonna): è il canale sul quale opera l'access point in questione*
- *ENC: indica il protocollo crittografico utilizzato dalla rete. Può essere WEP, WPA, WPA2.*
- *ESSID (ultima colonna): è il nome della rete wireless, così come visualizzato anche da Windows*



Dopo una serie di specifici comandi che omettiamo, si arriva finalmente alla fase finale della procedura in cui bisogna solamente attendere che un client scambi un pacchetto "speciale" con l'access point.
Un cracker che cercasse di violare la rete dovrà prepararsi ad ***un'attesa abbastanza lunga*** *prima di riuscire ad intercettare le informazioni necessarie: questo speciale pacchetto viene inviato solamente al momento in cui si instaura la connessione fra un client e l'access point.*
Dopo un po' d'attesa vengono intercettate le informazioni necessarie per far partire la procedura che può restituire la password della rete.
Qui entra in gioco il dizionario. Come accennato prima contengono, per ogni lingua, decine di migliaia di possibili password. Quest'ultimo passaggio prevede la prova di ogni singola chiave presente nel dizionario. Se viene trovata la procedura si ferma e la password viene comunicata all'utente. Nelle prove effettuate sulle nostri reti e su reti di altre persone, con il loro permesso, abbiamo purtroppo notato che questa procedura è veramente efficace. Infatti molte persone tendono a scegliere una password piuttosto semplice che è appunto contenuta in questi dizionari. La soluzione migliore rimane quella di scegliere una password lunga e complessa così da complicare la vita all'eventuale malintenzionato. Inoltre ci si può difendere permettendo l'accesso all'AP di determinati dispositivi identificati univocamente dal mac adress. In questo modo la sicurezza della rete cresce in maniera notevole e l'hacker di turno si troverà di fronte questa schermata:



Vulnerabilità del sistema informatico

Nonostante sia da scongiurare o comunque da evitare è possibile che un sistema informatico sia soggetto ad alcune vulnerabilità.
Una vulnerabilità è un errore che genera ripercussioni sulla sicurezza.

Spesso le problematiche che risiedono in un sistema sono legate ad errori, tecnicamente chiamati Bachi(Bug) all'interno del software.
Si possono identificare le seguenti tipologie di vulnerabilità:

Vulnerabilità dei componenti informatici:
- *Vulnerabilità di Sistemi/Applicazioni.*
- *Vulnerabilità dei Protocolli.*

Vulnerabilità strutturali:
- *Vulnerabilità legate alle architetture di rete.*

Vulnerabilità organizzativo/procedurali:
- *Vulnerabilità procedurale.*
- *Vulnerabilità organizzativa.*

Vulnerabilità di Sistemi/Applicazioni:
- *Vulnerabilità del Software:*

Overflow: buffer overflow, stack overflow e heap overflow.
Format String: vulnerabilità delle format function (fprintf, printf, sprintf, etc...)
- *Errori di configurazione: configurazioni poco robuste e/o inadeguate, ad esempio utenze non protette.*

Vulnerabilità dei Protocolli:
- *Debolezze di progettazione: permettono attacchi di tipo Spoofing, Hijacking e Sniffing.*
- *Errori implementativi dello stack di rete: permettono attacchi di tipo DoS/DDoS.*

Un malintenzionato potrebbe sfruttare una delle problematiche sopra elencate per corrompere un sistema informativo. E' possibile ad esempio che qualcuno sfrutti un errore del software o un punto d'accesso ad esso (backdoor) per accedere ai dati e alle informazioni. Oppure che qualcuno "spii" le operazioni compiute all'interno del sistema, catturando magari i dati che viaggiano sulla rete (sniffing) oppure verificando il traffico in ingresso e uscita da un particolare computer (port scanning).

In ultimo un haker esperto potrebbe introdurre in rete un programma apparentemente innocuo che invece rende inutilizzabile alcune funzionalità, deturpa, corrompe o distrugge le informazioni sensibili (trojan e virus).
Pensare ai danni che in una Pubblica Amministrazione possano produrre queste problematiche se trascurate spaventano e preoccuperebbero chiunque. Immaginiamo se qualcuno riuscisse a corrompere un database e venisse a conoscenza delle informazioni personali di migliaia di persone, ad esempio codici fiscali, coordinate bancarie, informazioni sanitarie, contrattuali o dati di fatture.
Peggio ancora se una persona esterna riuscisse ad entrare in un sistema dopo aver spiato i dati di accesso di un responsabile (login e password) e che quindi abbia la possibilità di operare indisturbato con il software per fini poco leciti. Costui potrebbe modificare alcuni valori all'interno della banca dati operando in maniera pulita e quindi difficilmente individuabile ad una prima analisi dei responsabili della sicurezza.
I casi a cui si può andare incontro sono innumerevoli, la cosa certa è che tutti in una maniera o in un'altra portano danni alla sicurezza di un sistema informativo. Risulta quindi necessario porre importante interesse alle tecniche di sicurezza onde evitare che un malfunzionamento o un attacco crei danni anche irreparabili.

Tra le tecniche più utilizzate ci sono:
- *Antivirus*
- *Antispyware*
- *Firewall*
- *Firma digitale, Crittografia*
- *Backup*
- *Intrusion Detection System (IDS)*
- *Network Intrusion Detection System (NIDS)*
- *Sistema di autenticazione*

Analizziamo la tecnica del Backup

Backup

La tecnica del backup è da ritenersi indubbiamente una delle più utilizzate in qualsiasi ambiente di lavoro, soprattutto nel momento in cui le informazioni presenti in database crescono e si aggiornano rapidamente durante le attività lavorative giornaliere.

Il backup è la copia dei file di un sistema in un'unità rimovibile, allo scopo di recuperare i dati in caso di malfunzionamenti o errori. Con il termine backup si intende sia la copia dei file, sia la procedura necessaria a copiare i dati.
Anche se i motivi per cui si eseguono i backup sono molteplici, si può affermare che il backup

serve a ripristinare i dati persi.

La perdita dei dati può avvenire per svariati motivi, solitamente:
• Cause legate agli operatori e ai sabotatori, persone fisiche (80%)
• Cause tecniche: (14%)
• Cause ambientali: (6%)

Il verificarsi di una qualsiasi delle cause elencate potrebbe significare il blocco dell'intero sistema informatico di un'azienda per un periodo lungo, nonché la perdita di dati importanti.
Immaginiamo ad alcune situazioni cui un'azienda o una Pubblica Amministrazione possono andare incontro. Potrebbero ad esempio non essere disponibili i server, costringendo gli utenti a lavorare senza collegamento a Internet, senza posta elettronica, senza fax e senza gestionale. Anche dopo la reinstallazione e riconfigurazione dei server potrebbero esserci disagi, ove non fosse possibile recuperare i file con i dati. Questo potrebbe comportare il reinserimento manuale di tutte le fatture e tutti gli ordini, tutta l'anagrafica clienti e dipendenti e la perdita di tutta la corrispondenza via posta elettronica. Se proviamo solo un attimo a considerare tali situazioni di disagio in un meccanismo complesso come può essere un'università con la perdita di tutti i dati riguardanti studenti, professori, dipendenti; in un attimo verrebbero perse le carriere universitarie, i dati di corsi e esami, nonché la banca dati degli uffici tecnici. E' chiaro come la gravità di fenomeni del genere blocchino per intero anche le più minime funzionalità di un meccanismo che entrerebbe di diritto in uno stato di caos totale.
Le procedure di backup non possono evitare il verificarsi delle cause, ma permettono il ripristino del sistema e il recupero dei dati in periodi brevi (i tempi dipendono dalla politica scelta).
Il backup può essere gestito in molti modi differenti e utilizzando supporti diversi, la scelta dipende dai mezzi a disposizione (quanto si vuole spendere in tempo e denaro), dalla priorità di ripristino (per quanto tempo si può permettere di tenere bloccato il sistema informatico?) e da altri fattori analizzati in seguito. Per creare una procedura di backup efficiente è quindi necessaria una attenta analisi del sistema informativo, al fine di definire una strategia di backup adatta alle esigenze aziendali e che garantisca il recupero dei dati persi, in tempi ragionevoli.

Si devono prendere in considerazione molti fattori, i più importanti sono:
• di quali file è opportuno eseguire il backup?
• si esegue un backup di rete o più procedure sui singoli PC?
frequenza dei backup, quante volte farli? Uno al giorno, uno alla settimana oppure uno al mese?
• quando si deve eseguire la procedura di copia dei dati? Alle otto di ogni mattina, alla sera, etc.
• che tipo di backup eseguire?
• quali tecnologie utilizzare?
La scelta della strategia da seguire dipende essenzialmente quindi dalla natura dei dati da salvare,

dalla loro frequenza di utilizzo e aggiornamento. Ad esempio i responsabili della sicurezza potrebbero decidere di effettuare il backup di un determinato archivio ogni pomeriggio sul finire delle attività lavorative dopo aver considerato che tale archivio subisce importanti modifiche ogni giorno (ad esempio i dati di fatture o richieste); d'altro canto su un archivio potrebbe essere effettuato il backup anche mensilmente qualora questo subisca poche modifiche (ad esempio l'anagrafica dei dipendenti).

Impatto causato dall'attuazione della minaccia

Si può considerare l'Impatto come la conseguenza dell'attuazione di una minaccia ed esso varia a secondo della natura dei beni colpiti e dagli obiettivi di sicurezza violati.

"Se un utente di una generica rete aziendale si connette con il suo portatile dall'estero ad Internet senza protezione(firewall, antivirus, aggiornamenti di sicurezza del Sistema Operativo)ed apre una e-mail infetta e al ritorno propaga il virus in tutta l'azienda, l'impatto può essere grave e coinvolgere tutti gli obiettivi di sicurezza."

Agente della minaccia: UTENTE
Vulnerabilità: CATTIVA CONFIGURAZIONE E FALLE DI SICUREZZA DEL SO
Minaccia: CATTIVE ABITUDINI E INCOPETENZA DELL'UTENTE

Un impatto può includere il blocco temporaneo della rete e dei computer per consentire l'azione dei tecnici che tenteranno un ripristino delle risorse causando in alcuni casi la perdita di dati: nemmeno i dati di Backup potrebbero essere al sicuro!

Sviluppo di un sito web

Dopo aver approfondito quest'argomento abbiamo deciso di sviluppare un nuovo sito web da zero, per renderci conto delle difficoltà che si incontrano in un lavoro del genere.
Contemporaneamente anche un nostro compagno di classe, Ivan Piani, aveva pensato alla stessa idea e per questo è iniziata una collaborazione per portare a termine un progetto piuttosto complesso.
Sicuramente questa è stata un'esperienza davvero utile in quanto ha permesso di provare concretamente quanto imparato in teoria.
Principalmente abbiamo utilizzato HTML, PHP E JS.
Dietro ad una semplice pagina web si nasconde un lavoro di mesi e i problemi che si incontrano sono svariati: da quelli più banali come la scelta del nome del sito, a quelli più complessi che hanno richiesto giornate intere per essere risolti.
Per testare tutte le problematiche si poteva creare il classico sito di compravendite online.
Ma volevamo fare qualcosa di nuovo, qualcosa di poco conosciuto in Italia. Ecco perché è nata l'idea di creare un nostro sito di aste al centesimo. Come detto queste tipologie di vendita in Italia sono poco conosciute. All'estero invece sono ormai realtà da anni. In breve le aste al centesimo funzionano così: il sito web vende dei crediti con i quali partecipare a delle particolari aste. L'oggetto in vendita ha una base d'asta di zero euro ed un timer settato ad un certo valore (poniamo per esempio 30s). All'inizio dell'asta stessa il timer comincia a scorrere all'indietro ed ogni utente che ha acquistato i crediti ha diritto a piazzare la sua offerta la quale incrementa il prezzo di un centesimo e fa tornare il timer al suo valore iniziale. Quando il timer arriva a zero l'ultimo offerente si aggiudica l'oggetto.
Per poter avere un simile sistema funzionante occorre gestire degli utenti e proteggere i loro dati personali inseriti nel momento della registrazione, gestire il login e la loro pagina personale.
Per un lavoro di questo tipo il linguaggio php studiato quest'anno si prestava alla perfezione.

Gestione degli utenti

Gli utenti sono il fulcro del sito web ed è necessario gestire e conservare in un database i loro dati inseriti durante la registrazione, garantire che non vengano rubati da malintenzionati ed inoltre prevedere la criptazione della password con la quale accederanno all'area riservata. La password non può essere conservata in chiaro all'interno del db e di conseguenza nemmeno gli amministratori del sito possono risalire alla stessa. Questo per una questione di onestà e correttezza nel confronto degli utenti i quali hanno diritto a mantenere privata la loro chiave di accesso al sito.
Per la gestione degli iscritti al sito si prestano perfettamente due linguaggi studiati in quest'ultimo anno scolastico:

- *Php per la registrazione degli utenti, il login e il logout, il recupero della password*
- *MySql per gestire le tabella in cui sono inseriti i dati degli utenti*

Per prima cosa creiamo la tabella per salvare informazioni sugli utenti:

```
CREATE TABLE utenti (

   id INT UNSIGNED NOT NULL AUTO_INCREMENT,
  name VARCHAR(30) NOT NULL,
  surname VARCHAR(30) NOT NULL,
  username VARCHAR(30) NOT NULL,
  password CHAR(40) NOT NULL,
  date date NOT NULL,
  stato varchar(50) NOT NULL,
  citta varchar(50) NOT NULL,
  postalcode int(50) NOT NULL,
  PRIMARY KEY(id)
);
```

La tabella utenti contiene nove campi:

- ***Il campo id** conterrà un numero univoco per identificare il record contenete i dati dell'utente loggato*
- ***I campi name e surname** conterranno rispettivamente nome e cognome dell' utente*
- ***Il campo username** conterrà il nome utente necessario per effettuare il login*
- ***Il campo password** conterrà la password (criptata attraverso la funzione SHA1) che l'utente dovrà utilizzare per effettuare il login*
- *Il campo date conterrà la data di nascita dell'utente*
- ***Il campo stato, citta** (volutamente senza l'accento perché per evitare problemi di interpretazione del codice è preferibile non usare caratteri particolari) e **postalcode** rappresentano la provenienza dell'utente iscritto.*

*La tabella è molto semplice. Da sottolineare è l'utilizzo della **funzione SHA1** (facente parte del core di funzioni di MySQL): data una stringa in input, questa funzione restituisce un valore di 40 bit irreversibile che rappresenta univocamente la stringa data (vedere la parte teorica a riguardo).*

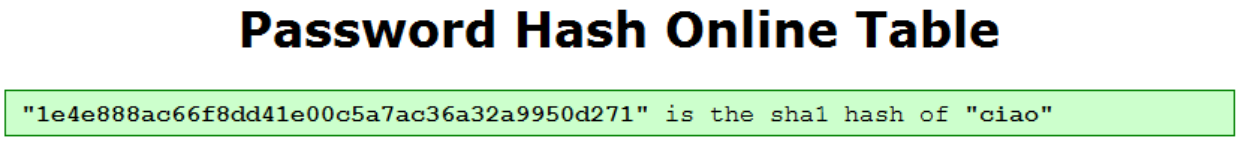
Nella registrazione che spiegheremo più avanti si potrà notare la scelta di criptare due volte con l'algoritmo sha1 la password dell'utente. Dal punto di vista teorico basterebbe criptare la password una sola volta in quanto sha1 genera in output una stinga irreversibile. Nella pratica, vista anche l'ampia diffusione di tale funzione, con password semplici scelte dagli utenti, basta una banalissima ricerca su google per trovare dizionari che restituiscono la stringa iniziale passata a sha1.

Per esempio se l'utente sceglie una password semplice come "ciao" la funzione sha1 restituisce questa stringa:

1e4e888ac66f8dd41e00c5a7ac36a32a9950d271

Come detto dal punto di vista teorico non si può in nessun modo tornare all'input "ciao". Ma anche in questo caso i malintenzionati hanno trovato il metodo di aggirare il problema. Infatti esistono molti siti web che gratuitamente forniscono l'output di sha1. Alcuni di questi siti però, all'insaputa dell'utente, conservano in dei database le parole inserite dagli utenti con il rispettivo risultato in sha1, creando così dei veri e propri dizionari. Quest'ultimi si trovano molto facilmente online. Basta infatti inserire in un qualsiasi motore di ricerca, come google la stringa criptata dalla funzione per trovare la parola che l'ha generata. In questo specifico caso:

E questo è la pagina del primo sito che la ricerca su Google fornisce:



Ovviamente questa situazione era una grossa minaccia alla sicurezza delle password degli utenti. Per cui siamo arrivati ad una conclusione tanto semplice quanto efficace. Una seconda criptazione del risultato ottenuto dalla prima criptazione avrebbe risolto il problema. Infatti è possibile dimostrare che la stessa password scelta prima come "ciao" non si riesce a riottenere nemmeno con l'ausilio dei dizionari appositi in quanto questi sono utili per una sola criptazione.

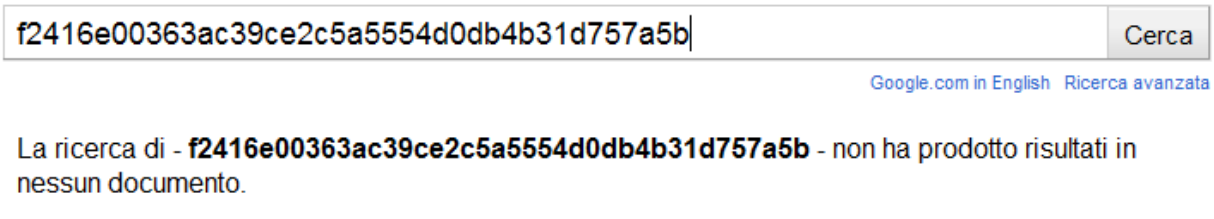
Come visto prima il risultato della funzione sha1 che ha come ingresso la stringa "ciao" è il seguente:

1e4e888ac66f8dd41e00c5a7ac36a32a9950d271

Passando alla funzione questa stringa otteniamo:

f2416e00363ac39ce2c5a5554d0db4b31d757a5b

Allo stesso modo di prima cerchiamo su Google se si riesce ad ottenere la password decrittata "ciao" associata a tale risultato:



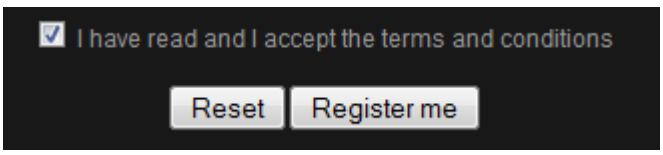
Come previsto Google non fornisce nessun risultato ed è quindi provata la sicurezza delle password gestita in questo modo.

Registrazione degli utenti

Per permettere la registrazione bisogna creare una pagina html in cui l'utente può inserire i suoi dati personali. Una volta completato il form i dati inseriti vengono passati ad una pagina php che effettua alcuni controlli. Se questi sono tutti superati l'utente verrà indirizzato ad una pagina in cui gli viene comunicato il successo dell'operazione. All'utente si presenta questo form:

REGISTER YOURSELF

Surname:	Rossi
Name:	Mario
Username:	rossi_mario
Password:	••••••••••••
Repeat password:	••••••••••••
E-mail address:	rossi_mario@libero.it
Date of birth:	2 Septembe 1970
Country:	Italy
City:	Roma
Address:	Via della conciliazione 4
Postal code:	00010



Appena l'utente completa tutto il form ed accetta i termini e le condizioni del sito, può premere il bottone register me che manderà i dati ad una pagina php. Di seguito sono riportati solo alcuni estratti.

```
<?php
```

- *Con questa riga di codice viene assegnata ad una variabile chiamata cognome il valore corrispondente inserito nel form di registrazione.*

```
$cognome=$_POST['cognome'];
```

- *Ovviamente tutti i dati devo finire in un tabella di un database. Per testare il lavoro in locale si può utlizzare EasyPhp, un unico programma che trasforma il computer in un server, esattamente come quello che si dovrebbe affittare per ospitare un sito web, fornisce un database e permette di installare in locale (cioè sul proprio computer in una cartella specifica) il lavoro prima di pubblicarlo online, in questo modo si può lavorare allo sviluppo del sito in tutta comodità. I parametri per connettersi al database in locale sono questi:*

```
$mysql = new mysqli('localhost', 'user', 'iamuser', 'password');
```

- *Qui cominciano i controlli sui campi: in questo specifico punto controlla che non siano stati lasciati vuoti dei campi. Se si riscontra l'errore viene restituisca la pagina adatta per comunicarlo all'utente:*

```
if($nome==null || $cognome==null || $username==null || $password==null || $ripetipass==null || $nascita==null || $mail==null || $indirizzo==null || $stato==null || $citta==null || $postalcode==null)

{

        echo '<meta http-equiv="refresh" content="1;url=register_failed_all.php">';

}
```

- *Un altro controllo fondamentale è quello sulla password inserita. Infatti è sempre buona norma chiedere all'utente di inserirla due volte per essere sicuri che abbia inserito veramente la password che voleva. Il controllo da fare è la coincidenza tra la password inserita e quella ripetuta.*

```
if($password!=$ripetipass)
        echo '<meta http-equiv="refresh" content="1;url=register_failed_pass.php">';
```

Se non dovesse andare a buon fine anche in questo caso viene restituita una pagina con l'errore.

- *Fondamentale nella gestione degli utenti è che ognuno scelga un username diverso per evitare doppioni che creerebbero solo confusione. Anche per questo è previsto un controllo. La variabile $row rappresenta il numero delle righe ed è inizializzata a zero.*

```
$row=0;
```

Con la query si cerca se l'username inserito nell'utente è già presente nel database e più specificatamente nella tabella utenti.

```
$query = "SELECT username FROM utenti WHERE username LIKE '$username'";
if ($stmt = $mysql->prepare($query))
{
```

```
        $stmt->execute();

        $stmt->store_result();
        $row=$stmt->num_rows;

        $stmt->close();

    }
```

A questo punto torna utile la variabile $row. Infatti se la query precedente restituisce una riga significa che nella tabella era già presente l'username scelta dall'utente. A questo punto, nel controllo che segue, viene generata la pagina d'errore

```
    if($row!=0)

    {

        echo '<meta http-equiv="refresh"
        content="1;url=register_failed_username.php">';

    }
```

Una volta conclusi i controlli sui campi si può procedere con l'inserimento vero e proprio dei dati nella tabella utenti:

```
{

$mysql->query("INSERT INTO utenti VALUES
(null,'$cognome','$nome','$username','$mail','$nascita','$stato','$citta','$indirizzo','$postalcode',sh
a1(sha1('$password')))");

                $mysql->close();

echo '<meta http-equiv="refresh" content="1;url=register_success.php">'

            } ?>
```

A questo punto si può restituire all'utente la pagina di avvenuta registrazione:

Congratulations, you are now registered in our website!
You have got 2 free coins, start use them!

Login degli utenti

Per permettere agli utenti iscritti di vedere una pagina personale a cui accedere con le proprie credenziali (nome utente e password) abbiamo utilizzato anche in questo caso il linguaggio php. Il risultato finale è il seguente e vedremo come dietro ad un semplice click ci sia in realtà molto codice.

Tutto lo script php lavora ovviamente sulla stessa tabella "utenti" creata in precedenza e in cui , grazie alla registrazione, sono salvati i dati degli utenti. Per il corretto funzionamento del login è stato necessario sviluppare tre moduli php:

- *checklogin.php*

```
<?php

// connessione al server e selezione db

//host name
$host="127.0.0.1";
//username mysql
$username="user";
//password mysql
$password="iamuser";
//nome db
$db_name="lowbay";
//nome tabella
$tbl_name="utenti";

mysql_connect("$host", "$username", "$password")or die("cannot connect");
mysql_select_db("$db_name")or die("cannot select DB");

// user e pass presi dal form html

$myusername=$_POST['myusername'];
```

```
$mypassword=sha1(sha1($_POST['mypassword']));

$myusername = stripslashes($myusername);
$mypassword = stripslashes($mypassword);
$myusername = mysql_real_escape_string($myusername);
$mypassword = mysql_real_escape_string($mypassword);
$count=0;
if($count==0)
{
        //query che  controlla la presenza dell'utente nel db

        $sql="SELECT * FROM utenti WHERE username='$myusername' and
password='$mypassword'";
        $result=mysql_query($sql);
        $count=mysql_num_rows($result);
        if($count==0)
}

//Se la query ha restituito un solo record si procede con il reindirizzamento dell'utente alla
pagina login_success.php

if($count==1){

session_start();
$_SESSION['myusername']=$myusername;
$_SESSION['mypassword']=$mypassword;
//session_register("myusername");
//session_register("mypassword");
header("location:../login_success.php");
}
else {
echo'Wrong username or password';
}
?>
```

In questo modulo, come del resto in tutti gli script php, bisogna inizialmente specificare i parametri per la connessione al database e alla tabella su cui lavorare. L'username e la password inseriti dall'utente vengono recuperati grazie alla funzione $_POST. Inoltre è molto importante utilizzare anche le funzioni stripslashes() e mysql_real_escape_string() che eliminano dai parametri inseriti dall'utenti eventuali caratteri speciali, per esempio gli accenti, che potrebbero portare a dei

problemi nella successiva esecuzione delle query. Quest'ultima controlla nel database se esiste un nome utente associato alla password passata dal form html. Con la successiva istruzione si contano il numero di righe ottenute dal risultato (può essere solo zero o uno in quanto nella registrazione, come mostrato si erano implementati i dovuti controlli per evitare registrazioni di più persone con lo stesso nome utente). A questo punto se il risultato ottenuto è uno significa che nel database è presente l'utente e viene quindi reindirizzato ad una nuova pagina, chiamata login_success.php in cui potrà consultare alcune informazioni personali. Vengono anche inizializziate le sessioni che si possono definire come l'arco di tempo in cui viene monitorata la connessione di un utente. Durante questo arco di tempo è possibile conservare informazioni sulla navigazione accessibili da ogni pagina collegata alla sessione.
La sessione inizia quando l'utente accede al sito e finisce quando lo abbandona, chiudendo il browser oppure quando effettua il logout.

- *Login_success.php*

```
<?
session_start();
if(!session_is_registered(myusername)){
header("location:main_login.php");
}
?>

<html>
<body>

//qui va specificato il codice html per la costruzione della pagina personale dell'utente

</body>
</html>
```

In questo modulo php viene prima di tutto fatto un controllo sulla sessione. Se non è registrata correttamente l'utente viene rimandato alla pagina principale. Invece se tutto funziona correttamente visualizza una pagina html con le sue informazioni personali.

- *Logout.php*

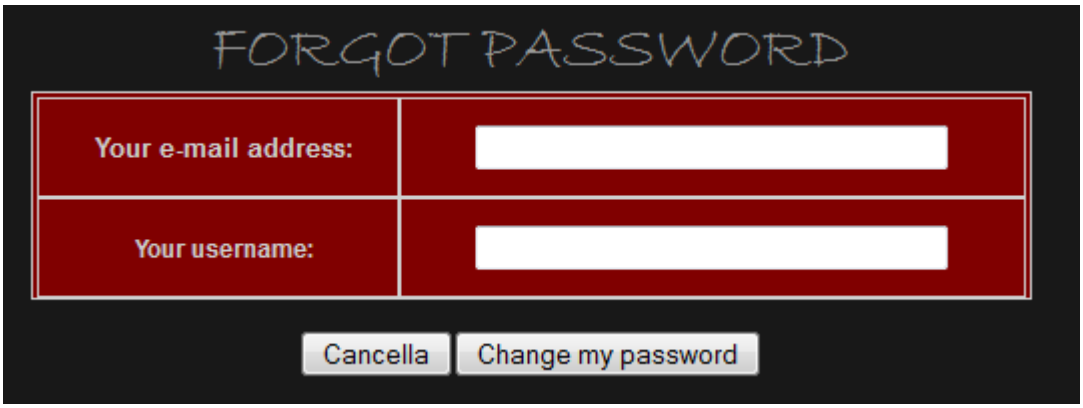
Questo modulo è molto semplice e come accennato in precedenza ha il solo scopo di distruggere la sessione creata dall'utente:

```
<?
session_start();
session_destroy();
?>
```

Recupero password

Altro aspetto fondamentale nella gestione degli utenti è il recupero della password in caso questa venga dimenticata. Molto spesso capita di non ricordare più i dati d'accesso ad un sito web e quindi è necessario prevedere tale eventualità.

Premendo sulla scritta Forgot password? l'utente è mandato in un'altra pagina html in cui dovrà inserire alcuni dati per dimostrare di essere veramente lui davanti al pc e non qualche malintenzionato che ha il solo scopo di rubare la password.
Il form html si presenta così:



L'utente deve solo inserire l'e-mail e l'username scelte al momento della registrazione.
Quando poi preme il bottone Change my password i dati vengono passati al modulo php inviapass. Qui innanzitutto viene controllata la veridicità dei dati inseriti. Se questi sono realmente i dati di un utente viene inviata a quest'ultimo una e-mail con una nuova password di 8 caratteri generati casualmente da un'apposita funzione. In questo modo anche se ad inserire i dati fosse stata una persona diversa dall'utente, c'è sempre la sicurezza dell'indirizzo e-mail a cui comunque solo l'utente può accedere. Per inviare le mail in php ci sono sostanzialmente due metodi: il primo è l'utilizzo della funzione mail(). In realtà è ormai superata e con molti gestori telefonici non funziona più. Questa funzione lavora sulla porta 25, che non prevede l'autenticazione. Per una questione di

sicurezza è quindi preferibile non usarla. Molto più pratica ed interessante è l'utilizzo della classi class.phpmailer e class.smtp. Questi sono moduli php open source, ovvero disponibili in rete per chiunque. Vanno semplicemente richiamati nella propria pagina php di invio mail. Di seguito c'è il codice, con le opportune spiegazioni per l'invio di una nuova password all'utente.

```
<?php

//Recupero dei dati inseriti nel form html visto in precedenza

$mailinput=$_POST['mail'];
$user=$_POST['user'];

//Connessione al database e query che controlla l'effettiva veridicità dei dati inseriti

 $conn=mysql_connect("127.0.0.1","user","iamuser") or die("Cannot connect");
 mysql_select_db("hippobids") or die("Cannot select database");
 $query="SELECT mail,password FROM utenti WHERE mail='".$mailinput."' and username='".$user."';";
 $res=mysql_query($query,$conn) or die ("Query failed");
 $rows=mysql_num_rows($res);

//Se il controllo non va a buon fine l'utente riceverà il seguente errore:

 if($rows == 0)
 echo "Il tuo indirizzo mail non è nei nostri archivi <br>"; //pagina che dice failed mail

//Invece se i dati corrispondono viene generata la nuova password casuale di 8 caratteri

  if($rows == 1)
  {
       //genera la nuova password
       $nChar = 8;
       $newpass= substr(md5(rand(0, 1000000)), 0, $nChar);

//Importazione delle classi phpmailer e smtp essenziali per il corretto invio della mail

        include("class.phpmailer.php");
        include("class.smtp.php");
        $mail= new PHPMailer();
        $body= '<html>
```

```
        $mail->IsSMTP();
        $mail->SMTPAuth = true;
//Naturlamente vanno specificati tutti i vari parametri. Per il suo invio abbiamo deciso di utlizzare i server SMTP messi a disposizione gratuitamente dalla gmail di Google (Dove ci sono gli asterischi vanno specificati i dati personali del proprio account gmail)

        $mail->SMTPSecure = "SSL";
        $mail->Host  = "ssl://smtp.gmail.com";
        $mail->Port  = 465;
       $mail->Username  = *********"; // specificare l'username gmail
       $mail->Password   = "**********";  // specificare la password gmail
       $mail->From     = "**********"; //specificare l'indirizzo del mittente
       $mail->FromName = "*******"; //nome del mittente
       $mail->Subject      = "Ecco la tua nuova password!"; //oggetto del messaggio
      $mail->MsgHTML($body);
       $mail->AddAddress("$mailinput");
       $mail->IsHTML(true);

//Controllo sull'effettivo invio della mail. Se non va a buon fine viene restituito l'errore
       if(!$mail->Send())
       {
         echo "An error occurred while sending mail";
       }
       else
//Invece se è tutto ok la mail viene inviata all'indirizzo specificato in precedenza dall'utente. Inoltre viene eseguita la query d'aggiornamento. Infatti nel database verrà sostituita la vecchia password con quella nuova appena generata

        echo"<div style='text-align: center;'><img style='width: 72px; height: 72px;' src='../../images/completed.png' align='center'><big > An e-mail with a new password has been sent to your mail address, check it and login back</big></div>";
       $aggiornapassdb=" UPDATE utenti SET password ='".sha1(sha1($newpass))."' WHERE mail='".$mailinput."' and username='".$user."';";
       $res2=mysql_query($aggiornapassdb,$conn) or die ("Query failed");
       mysql_close($conn);

   }
?>
```

Questo è quello che l'utente riceve nella sua casella di posta elettronica:

Una volta ottenuta la nuova password può subito effettuare il login (come mostrato in precedenza nel database la password è già stata aggiornata). A quel punto l'utente si trova davanti un semplice form html in cui potrà inserire una nuova password a suo piacimento.
Il recupero password poteva essere fatto in molti modi diversi. Questo ci è sembrato il più sicuro in quanto solo l'utente che ha accesso alla sua casella e-mail può riuscire a cambiare la password diminuendo drasticamente la possibilità di un furto di password.

Scelta del nome e acquisto del dominio

In questo approfondimento abbiamo mostrato solo una parte del sito web sviluppato ovvero quella relativa alla sicurezza in rete ed in particolar modo legata al php, argomenti di studio di quest'ultimo anno scolastico. Come si può facilmente intuire il progetto è ben più ampio. Non a caso oltre a noi due ci ha lavorato anche il nostro compagno di classe Ivan Piani. Dopo mesi di prove ed ore di lavoro possiamo affermare che il sito è pronto.
Una delle ultime operazioni da fare è stata la scelta del nome e il conseguente acquisto del dominio. Dopo qualche giorno siamo arrivati al nome attuale: Hippobids.
Hippo sta ad indicare il simpatico ippopotamo scelto come mascotte mentre bids ricorda la natura del sito in cui tramite delle puntate si possono acquistare degli oggetti.
Dal punto di vista tecnico, oltre all'acquisto del dominio (hippobids.com), è stato necessario anche l'acquisto di uno spazio Mysql per gestire il nostro database e le relative tabelle.

Hippo, la nostra simpatica mascotte

Con estrema soddisfazione siamo ora online, momentaneamente con una versione ridotta del sito visto che stiamo raccogliendo le prime iscrizioni che stanno arrivando da ogni parte d'Italia. Tutto quello che si vede online è ciò che è spiegato in questa ricerca.
In conclusione possiamo affermare con assoluta certezza che quando si prova a mettere in pratica quanto studiato in teoria si possono incontrare diverse difficoltà ma allo stesso tempo risulta interessante e soddisfacente risolvere problemi sempre diversi, problemi che difficilmente i libri di testo riportano

Conclusione

Il lavoro proposto è stato realizzato grazie ad un'intensa collaborazione tra noi compagni durante l'anno scolastico, spesso organizzando pomeriggi di studio e approfondimento e sfruttando anche la preziosa collaborazione di alcuni docenti dell'istituto.
Fondamentale durante l'anno scolastico è stata l'attività di Stage formativo svolta presso Bankadati di Sondrio, il reparto informatico della Credito Valtellinese, dove attraverso i nostri Tutor abbiamo appreso importanti concetti che si sono dimostrati utili al fine del lavoro realizzato.
Questo lavoro rappresenta, ci auguriamo degnamente, la conclusione di un ciclo di studi che ha lasciato ampio spazio anche ai rapporti d'amicizia: se non fosse stato così, probabilmente non avremmo deciso di collaborare alla realizzazione di questo progetto.
Anche per questo motivo siamo soddisfatti della nostra scelta di approfondire un argomento di interesse comune.
Un ulteriore ringraziamento va ad Ivan Piani, un nostro compagno di classe che appassionandosi alla realizzazione del sito web, è intervenuto più volte sostenendoci e aiutandoci concretamente nella parte pratica del lavoro.
Infine non possono mancare i più sinceri ringraziamenti nei confronti dei nostri docenti che con il loro lavoro sono diventati "guide" durante questi cinque anni di scuola, anni che comunque non dimenticheremo facilmente qualsiasi sarà il nostro percorso futuro.

Fonti

Sicurezza Informatica - Aranzulla Salvatore – 2007
Task Corso di Informatica 3 - Piero Gallo Fabio Salerno
Sistemi di elaborazione e trasmissione delle informazioni, Reti di calcolatori - Hoepli - P.Levi
http://sicurezzacittadino.caspur.it
http://www.slideshare.net
http://www.megalab.it/
http://it.kioskea.net
http://www.donatantonio.net