Gestor de contrasenyes en línia

José Francisco Mata Ayuso

Enginyeria Tècnica en Informàtica de Sistemes

Consultora: Cristina Pérez Solà

13 de juny de 2014

Universitat Oberta de Catalunya
www.uoc.edu

Com la tortuga de la faula d'Isop, però sense llebre amb qui competir, sinó el temps, aquesta memòria clou un camí començat ja fa alguns anys.

El meu agraïment a "l'equip de la UOC", i a nivell personal a tots els que m'han estat suportant i encoratjant durant aquest camí, especialment a la Sandra i a dues persones que "indirectament" m'han ajudat a "tombar la closca", el meu pare, i el Luca, encara que no sé cap a quin costat.

Resum

Aquest document s'ha elaborat seguint una de les iniciatives proposades com Treball de Final de Carrera dins l'àrea de Seguretat Informàtica, consistent en la creació d'una eina en línia, amb les corresponents garanties de seguretat i privacitat, des d'on un usuari registrat, fent ús de la majoria dels dispositius d'accés a Internet al seu abast, pugui emmagatzemar i posteriorment consultar, modificar o eliminar les seves contrasenyes de serveis habituals, com ara correu electrònic, xarxes socials, portals privats, connexions, etc., només tenint que recordar l'usuari i contrasenya d'accés a la utilitat.

Per desenvolupar aquesta solució s'han codificat en HTML i PHP les funcions bàsiques d'alta d'usuari, inici de sessió, creació, modificació i eliminació de serveis, fent ús de JavaScript, incloent llibreries obertes o funcions ja existents com Jquery, o de comunicacions asíncrones amb el servidor, com AJAX. Per allotjar el lloc web s'ha implementat un servidor Apache amb el corresponent certificat públic autosignat i l'habilitació dels serveis HTTPS, per garantir la seguretat de les comunicacions, MySQL com a gestor de base de dades i PHP.

Tenint en compte la importància de la preservació de les dades emmagatzemades es presenta AES com algorisme de xifratge, fent servir la contrasenya principal com clau d'encriptació, i la funció hash SHA2, més bits de salt, com elements addicionals de protecció de comunicacions i possible accés a les taules, tant per a la fase de registre com la codificació del nom d'usuari i contrasenya principal abans de ser desats. En darrer terme s'ha creat un únic usuari SQL només amb permís d'execució de les crides a rutines anteriorment creades per l'administrador, evitant el risc del llenguatge de consultes, *querys*, més vulnerable i susceptible a vies d'accés a les dades, la seva manipulació o eliminació, com les anomenades injeccions SQL.

Índex

Índex de figures

1 Introducció

1.1 Justificació del Treball

Davant la utilització de credencials d'usuari i contrasenyes, a l'entorn de les noves tecnologies, per accedir a diferents programes i serveis, ja sigui locals o en xarxa, com ara el correu electrònic, xarxes socials, banca en línia, carpetes de continguts sincronitzats o compartits, es planteja la creació d'una eina en línia des d'on permetre l'accés a les contrasenyes des dels diferents dispositius, ordinador de sobretaula, portàtil, tauleta tàctil, telèfon intel·ligent.

Aquest eina també vol oferir a l'usuari l'avantatge de només haver de recordar-se'n d'un parrell de dades, el nom i clau per accedir, per contra s'ha de garantir la confidencialitat no només amb la codificació de les dades contingudes sinó també mentre viatgen per la xarxa, per no comprometre totes les dades .

1.2 Objectius

L'objectiu principal és dissenyar i implementar un gestor de contrasenyes segur a la xarxa.

- Per assolir l'objectiu anterior s'haurà de dissenyar un lloc web que incorpori les següents funcionalitats:
  - Funcionalitat per registrar un usuari nou, reduint al màxim les dades per garantir la privacitat, com ara un "usuari" i una paraula o frase clau, que fer servir con clau d'encriptació. Per altra banda s'ha de fixar el nom d'usuari a un mínim de 4 dígits, controlant l'existència d'altra igual, i la clau a una complexitat mitja o alta.
  - Funcionalitat per emmagatzemar nom o àlies del servei o lloc, codi d'usuari, contrasenya.
  - Funcionalitat per editar el grup de dades guardades.
  - Funcionalitat per donar de baixa dades.
  - Funcionalitat que verifiqui la fortalesa de les claus.

- Respecte a la seguretat, per garantir la privacitat dels usuaris, el sistema es dissenyarà tenint en compte aquestes premisses :
  - Assegurar que ningú pugui obtenir dades alienes emmagatzemades o comprometre les credencials d'altres usuaris, tant administradors com algú que pugui interceptar les comunicacions o accedir al servidor.
  - Crear un lloc segur i de confiança, tipus https, on allotjar l'aplicació.
  - Assegurar que cap usuari pugui eliminar dades que no siguin les seves o un intrús el contingut de les taules.

1.3 Enfocament i mètode

Tenint en compte l'abastament del projecte i per facilitar la seva aplicació, s'ha optat per fer servir un esquema de cicle de vida clàssic o en cascada, que com es va veure a l'assignatura d'Enginyeria del Programari consta de les fases de requisits, anàlisis i disseny, implementació, proves i manteniment, si bé s'ha simplificat una mica més ometent la fase de manteniment ja que aquesta darrera fase s'hauria de portar a terme una vegada posada la utilitat a disposició dels usuaris, i això queda fora dels objectius inicials d'aquest treball.

1.4 Planificació

Fase d'anàlisis de requisits

- Avaluar sistemes i llocs similars.
- Avaluar tècniques d'atac de contrasenyes i captura de dades per tractar de contrarestar-les.
- Avaluar els diferents sistemes d'encriptació i seleccionar el més adient tant per la fase d'inici de sessió, com l'emmagatzematge de les dades.
- Avaluar el tipus de bases dades per trobar el que compleixi amb els requeriments de seguretat i entorn.
- Avaluar criteris de fortalesa de contrasenyes per la funcionalitat de classificació.
- Avaluar els sistemes de comunicació que millor garanteixin la seguretat entre client i servidor.

Fase de disseny

- Disseny de l'entorn client i servidor.
- Disseny dels protocols criptogràfics.
- Disseny de la pàgina inicial.
- Disseny de les pàgines amb totes les funcions.
    - Registre d'usuari nou.
    - Avaluació de fortalesa de contrasenya.
    - Introducció de dades a guardar.
    - Modificació de dades.
    - Eliminació de dades.
- Disseny de l'estructura de la base de dades on guarda les dades.

Implementació

- Configurar el servidor segur on allotjar l'aplicació.
- Crear la base de dades on guardar les dades.
- Generar els certificats de seguretat pel servidor.
- Implementar les pàgines dissenyades.
- Implementar el codi de programació per les diferents funcions dissenyades.

Proves

- Registrar alguns usuaris, introduir dades, modificar i eliminar.
- Accedir com administrador a les bases de dades i verificar que no és comprensible la informació.
- Capturar paquets durant la comunicació per avaluar que no hi ha dades sense encriptar.

1.5 Productes obtinguts

Una vegada completades les diferents fases planificades s'ofereix un conjunt de pàgines PHP, HTML i llibreries JavaScript, que juntament amb la base de dades MySQL, poden ser instal·lades a un servidor Apache, amb PHP i MySQL com a gestor de base de dades. A continuació, per oferir l'eina de gestió de contrasenyes segons els objectius i fites indicats al document, s'instal·la un certificat públic del lloc i s'habilita els serveis HTTPS. Per últim, es crea un usuari SQL restringit amb només permisos d'execució de rutines.

2 Sistemes de gestió de contrasenyes

2.1 Avaluació de sistemes i llocs similars

Prenent com a primer referent les utilitats ofertes per l'INTENCO, Instituto Nacional de Tecnologías de la Comunicación, s'ha fet una aproximació per tractar de veure punts comuns, funcionalitats i, sense aprofundir, tecnologia i seguretat emprades.

Utilitats d'ús local:

- **KeepassX**. Utilitat multiplataforma, Windows, Linux, MacOS X, que es pot considerar una base de dades xifrada amb AES o Twofish, fent servir una clau de 256 bits o un arxiu de sistema, emmagatzemat a qualsevol suport de dades (USB, CD, etc). També ofereix una utilitat per generar contrasenyes.

- **LockCrypt**. Ofereix més plataformes on ser instal·lada com Windows Mobile i telèfons intel·ligents/PDA amb J2ME. En aquest cas el sistema de treball s'estructura en subgrups on guardar les dades aplicant formats, nombre de caps i tipus, en funció de les plantilles predefinides o generant noves. El sistema de xifrat també és AES o Twofish de 256 bits, amb clau o fitxer de sistema.

Utilitats en línia:

- **LastPass**. Basada a la instal·lació d'un connector al navegador, permet guardar el lloc i paraules de pas introduïdes durant la navegació i iniciar sessió des del seu lloc web amb un sol clic una vegada guardada la informació.

- **Passpack**. Mitjançant el navegador fa servir una contrasenya d'accés i altra de "empaquetat", per donar major seguretat. Per altra banda, una vegada introduïts l'usuari i primera paraula de pas, utilitza un control de CAPTCHA[1], consistent en fer clic a un quadre negre per introduir la segona clau.

- **ClipperZ**. Tot i no trobar-se als oferts per l'INTECO, és el referent principal d'aquest treball ja que la seva estructura i utilitats són les que més s'apropen als objectius plantejats. Un lloc web des d'on una vegada registrat l'usuari pot accedir i gestionar les "targetes", que és com denomina els tipus d'estructura de dades a emmagatzemar, com ara contrasenya de lloc web, compte bancari, targeta de crèdit, llibreta d'adreces, personalitzat, o accés directe a un lloc web mitjançant les utilitats ofertes per ClipperZ.

1 CAPTCHA són les sigles de Completely Automated Public Turing Test to tell Computers and Humans Apart, test controlat per una màquina, d'aquí prova inversa de Turing, per verificar que l'usuari és humà. Un dels més utilitzats és el quadre d'imatge amb fons degradats i lletres distorsionades que l'usuari ha de teclejar.
Wikipedia. Captcha. [Data de consulta: 3/4/2014]. http://es.wikipedia.org/wiki/Captcha

2.2 Tècniques d'atac de contrasenyes.

Atenent al concepte de mecanisme de protecció i acreditació d'identitat que s'explica dins del mòdul 3 de Seguretat en Xarxes de Computadors, *la idea bàsica de l'autenticació basada en contrasenyes és que l'usuari A envia la seva identitat (el seu identificador d'usuari, el seu nom de login, etc.) seguida d'una contrasenya secreta x_A (una paraula o combinació de caràcters que l'usuari pugui memoritzar). El verificador B comprova que la contrasenya sigui vàlida, i si ho és dóna per bona la identitat d'A.* Algunes de les tècniques d'atac són:

- Obtenció del **llistat de contrasenyes en clar**: Aquesta vulnerabilitat es produeix quant l'usuari guarda les contrasenyes sense protecció o el verificador fa servir una matriu tipus usuari, contrasenya, encara que sigui en una carpeta amb permisos restringits, i algú llegeix les dades o aconsegueix accedir a la carpeta.
- **Atac de diccionari**: Quant es troben codificades, per evitar la situació anterior, un possible atac és utilitzar paraules existents al diccionari, per anar provant si es tracta de la contrasenya d'accés.
- **Força bruta**: En aquest cas és més costosa computacionalment, ja que prova de trobar la contrasenya amb totes les combinacions possibles, lletres, números, signes, majúscules, etc.
- **Diccionari i força bruta**: Si bé és una mescla de les anteriors és el més emprat pel programari "trenca-contrasenyes", per reduir els costos de la força bruta i basant-se en l'Enginyeria Social, que indica la utilització de combinacions de paraules i números.

En el cas de la utilitat s'opta per encriptar tant les dades d'inici de sessió com el contingut guardat, però possiblement amb la utilització de funcions hash, per la qual cosa s'ha de valorar els atacs com ara:

- Atac taules arc de Sant Marti: Consistent en anar provant la llista de funcions hash fins trobar la que possiblement s'ha fet servir a la codificació per obtenir la contrasenya, en un temps computacional "acceptable" i comprometre tota la informació.

Altres amenaces a tenir en compte en fer servir una solució en línia és la **captura de dades durant les comunicacions,** ja sigui **"escoltant"** amb programes detectors (*sniffers*) els paquets que circulen per la xarxa, en cas de comunicacions sense autentificació de servidor ni client, generant l'atacant les seves pròpies claus públiques i privades, fent creure a les dues parts que és l'interlocutor, en l'anomena **"home al mig"**.

2.3 Criteris de fortalesa de contrasenyes

Tot i que no existeix un criteri únic per determinar la fortalesa d'una contrasenya, sí hi ha recomanacions fonamentades en criteris matemàtics, com ara les possibles combinacions amb repetició dels elements utilitzats, per determinar que la **longitud mínima recomanada és de 8 dígits,** que dificulta els atacs de força bruta en temps computacionalment acceptables. Altres responen més al comportament de l'ésser humà i tracten **d'evitar que la clau compleixi patrons de comportament**, com per exemple la substitució de determinats caràcters alfabètics per numèrics, concatenació de **paraules de diccionari o noms propis** amb nombres, **dates**, etc, que serien vulnerables als atacs de diccionari.

Després d'avaluar alguns algorismes que ofereixen valors numèrics per tractar de determinar aquesta fortalesa, alguns tan complexes com el fet servir per "Silent Circle"[1], un servei per encriptar les comunicacions, es planteja avaluar només que la longitud sigui superior a 8, i calcular el grau de complexitat en funció d'una simplificació de l'**entropia**, amb la següent fórmula:

$$H = L\frac{\log N}{\log 2}$$

On H és el grau d'entropia, mitjana de la quantitat d'informació que contenen els dígits de la contrasenya sobre el total dels possibles de l'alfabet, N, dividit en aquest cas en 5 grups, per incentivar l'ús de majúscules, que comptabilitzaran com 26 elements, a l'igual que les minúscules, números, amb 10, i símbols amb 32, obtenint en el cas de la intervenció d'almenys un element de cada grup un alfabet de 94 elements. Per últim es divideix aquest grau entre 10 per obtenir una escala més visual, on el valors del **0 al 3** es consideren **febles**, **4 a 6 mitjans**, i superiors a **6 forts**.

1Silent Circle's password strenght algorithm http://andrew.hedges.name/experiments/password-strength/ (4/4/2014)
Silent Circle. Página Oficial de la companyia https://silentcircle.com/ (4/4/2014)

3 Anàlisi de requisits

3.1 Funcionals

El lloc ha de permetre identificar-se de manera unívoca, per garantir que les dades guardades o a les quals s'accedeixen són les d'aquell usuari en qüestió. A més a més, la informació en clar s'haurà de gestionar a la part client de la solució per tal de que no viatgi per la xarxa. Pel que fa a la informació continguda a la base de dades, s'ha de facilitar a l'usuari una manera àgil d'afegir, modificar o eliminar les seves dades, sempre sense comprometre-les.

3.2 De seguretat

Garantir la inintel·ligibilitat de les dades, tant per part dels administradors del sistema amb accés físic als continguts, com durant els intercanvis entre client i servidor per algú que pugi interceptar o "escoltar" les comunicacions.

3.3 Dades

Els requeriments pel que fa a la base de dades són bàsicament emmagatzemar de manera segura la informació *nom d'usuari, contrasenya general,* i les dades dels *serveis*, facilitar la seva gestió mitjançant la seva integració i compatibilitat amb els llenguatges de funcionament tant de l'entorn client com del propi servidor.

3.4 Comunicacions

Com s'ha de facilitar l'accés des de qualsevol dispositiu i connexió a Internet el més adequat és treballar a nivell de protocols de transport, com ara SSL (Secuer Sockets Layer) o TLS (Transport Layer Security), que ofereixen **confidencialitat,** ja que la informació va en paquet xifrats mitjançant claus simètriques determinades a l'inici de sessió, diferents pels enviats del client al servidor que del servidor al client, **autenticació d'entitat,** ja que amb un protocol de repte-resposta, el client pot verificar que es tracta del servidor, en obtenir el seu certificat i per tant la clau pública amb què s'ha signat, i de **missatge,** ja que a cada paquet es pot incloure el codi MAC per controlar la no manipulació dels paquets, **eficiència** amb l'especificació d'un identificador de sessió i algorismes de compressió de dades, **i extensibilitat** ja que permet la incorporació, de trobar-se, nous algorismes més eficients o segurs.

4 Disseny

4.1 Arquitectura

Com ja s'ha indicat anteriorment, la base d'aquesta utilitat és facilitar l'accés des de qualsevol dispositiu amb connexió a Internet, per tant al costat client s'ha d'utilitzar un estàndard. Pel que fa al servidor es planteja fer servir un servidor web amb protocol segur interconnectat a un de base de dades on guardar tota la informació.

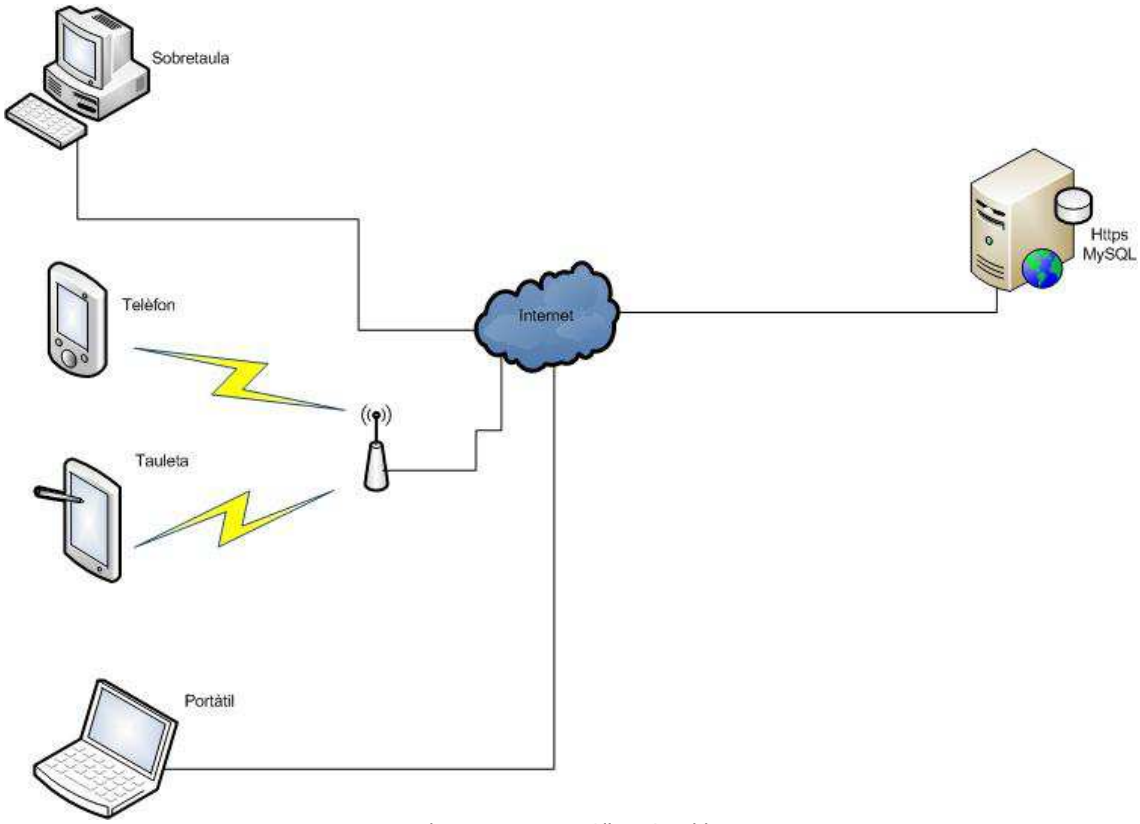

Figura 1 – Entorn Client Servidor

4.2 Protocols criptogràfics

En base a l'anàlisi de la fase anterior, es determina que intervindran dos protocols de codificació, un per al registre i inici de sessió, i un altra per a l'encriptació de les dades en general. Per al registre s'utilitzarà una funció hash del nom d'usuari, que per la seva propietat d'unidireccionalitat ofereix gran dificultat per conèixer el valor en clar, **més bits de salt** per tractar d'evitar un possible atac per la tècnica de les "taules de Sant Martí" en cas d'obtenir aquesta dada, que s'enviarà i guardarà al servidor juntament amb el hash de la contrasenya principal xifrada amb el nom d'usuari com a clau. Per altra banda, durant la fase d'inici de sessió, el servidor generarà un número aleatori del qual sumat amb el hash de la contrasenya principal xifrada, es generarà una altra funció hash que servirà per a la validació del repte-resposta per part del servidor. Pel que fa a la funció hash SHA2 pot ser l'elecció i AES l'algorisme de xifrat per les dades en general, fent servir com a clau la contrasenya principal.

4.3 Decisions de disseny relatives a la seguretat

Amb la informació analitzada fins al moment es considera adient el **xifrat en bloc**, consistent en el xifrat de blocs de longitud determinada amb una clau compartida, per la seva velocitat de xifrat relativament acceptable per aquest tipus de solució. Per altra banda, el **AES** (Advanced Encryption Standard) ofereix millors garanties de seguretat que el DES (Data Encryption Standard).

Durant el registre d'un nou usuari, aquest introduirà la contrasenya d'accés, que en cap cas ha de viatjar en clar. Una vegada registrat el client, per iniciar sessió és pot utilitzar un protocol de repte-resposta amb número aleatori que haurà generat el servidor, enviat com resposta el hash de la contrasenya d'accés encriptada amb el nom d'usuari i el número aleatori codificats, permetent al server descodificar i verificar la identitat amb les dades guardades durant el registre.

4.4 Decisions de disseny relatives al desat de dades

Dins els Sistemes de Bases de Dades Relacionals s'ha avaluat **Firebird**, un sistema de codi obert alliberat per Borland al 2000, del qual destacaríem la bona seguretat basada en usuaris/rols, escalabilitat mitjana i multiplataforma pel que fa a sistemes operatius, i **MySQL**, un dels sistemes multiusuari i multifil més estesos sota llicència "oberta" (GNU GPL), amb **bones característiques de seguretat,** com ara l'ús de taules hash en memòries temporals, sistema de contrasenyes i privilegis segurs amb verificació a l'ordinador amfitrió, trànsit de contrasenyes en connectar-me al servidor, i respecte a la integració, molt utilitzat dins entorns webs com ara plataformes PHP, fet que el fa inicialment més adient a les necessitats plantejades.

4.5 Disseny del procés d'autenticació

Atenent a la importància d'aquest apartat es considera més adient tractar-lo amb més detall. Bàsicament, s'ha tractat de dissenyar un procés d'autentificació que no comprometi el contingut de les dades, optant per utilitzar una única contrasenya que serà la mateixa que es farà servir a la codificació de la informació.

4.5.1 Donar d'alta un usuari

Com no existeixen dades l'usuari omple el formulari amb les dades en clar de l'usuari i paraula de pas.

Al client:

- Es genera un número aleatori com bits de salt (*bitSalt*).
- Es verifica que el nom d'usuari té com a mínim 4 dígits.
- Es verifica que la paraula de pas té un criteri de "complexitat" mitjà o fort.
- Es calcula el xifratge de la paraula de pas amb el nom d'usuari com clau (*pswX*).
- Es calcula el hash(pswX).
- Es calcula el hash(hash(usuari)+bitSalt).

S'envia del client al servidor les dades **per ser guardades dins la base de dades**:

- Hash(hash(usuari)+bitSalt) $\rightarrow$ hUS.
- BitSalt
- Hash(parulaPasXifrada) $\rightarrow$ hPswX.

Al servidor:

- Insereix la informació a la taula *usuaris(us, salt, psw).*

Fi del procés, es torna a la pàgina principal per tal que l'usuari inicií sessió si ho desitja.

4.5.2 Autenticar-se

Des de la pàgina principal l'usuari omple el formulari amb el nom d'usuari i paraula de pas donats d'alta a la fase de registre.

Al client:

- Es demana un número aleatori de sessió al servidor (nAle).
- Es verifica que el nom d'usuari té com a mínim 4 dígits.
- Es calcula el xifratge de la paraula de pas amb el nom d'usuari com a clau.
- Es calcula el hash(hash(paraulaPasXifrada)+numeroAleatori) → (hPswXN).
- Es calcula el hash(usuari) → hU.

S'envia del client al servidor les dades:

- hU
- hPswXN

Al servidor:

- Existeix un número aleatori guardat com variable de sessió ($_SESSION["nAle"]=mt_rand())
- Se cerca dins la taula *usuaris* aquell registre que el camp usuaris.us = hash(hU+usuaris.salt) i hash(usuaris.psw + nAle = hPswXN.

S'envia del servidor al client:

- "Cert" i el valor d'usuaris.us, o "fals" i un missatge d'error.
- Se cerca dins la taula *serveis* tots els registres on el camp *usUs* = *usuari.us*

4.5.3 Donar d'alta dades de servei

Una vegada autentificat.

Al client:

- Hi ha:
  - Paraula clau en clar inserida durant la fase d'autenticació (*psw*).
  - El valor usuaris.us retornat pel server a l'autenticació.

- Fent clic un botó s'afegeix una línea per omplir les dades en clar referents al nou servei:
  - Servei, adreça, nom d'usuari, contrasenya
- En fer clic al botó "Acceptar", xifra les 4 dades amb el valor de *psw* com clau.
  - ServeiX, adreçaX, usuariX, contrasenyaX.

S'envia del client al servidor les dades **per ser guardades dins la base de dades**:

- Usuaris.us.
- ServeiX que amb usuaris.us formaran la clau única.
- adreçaX
- UsuariX
- ContrasenyaX.

Al servidor:

- Insereix la informació la taula *serveis(usUs, servei, adre, us, contra).*

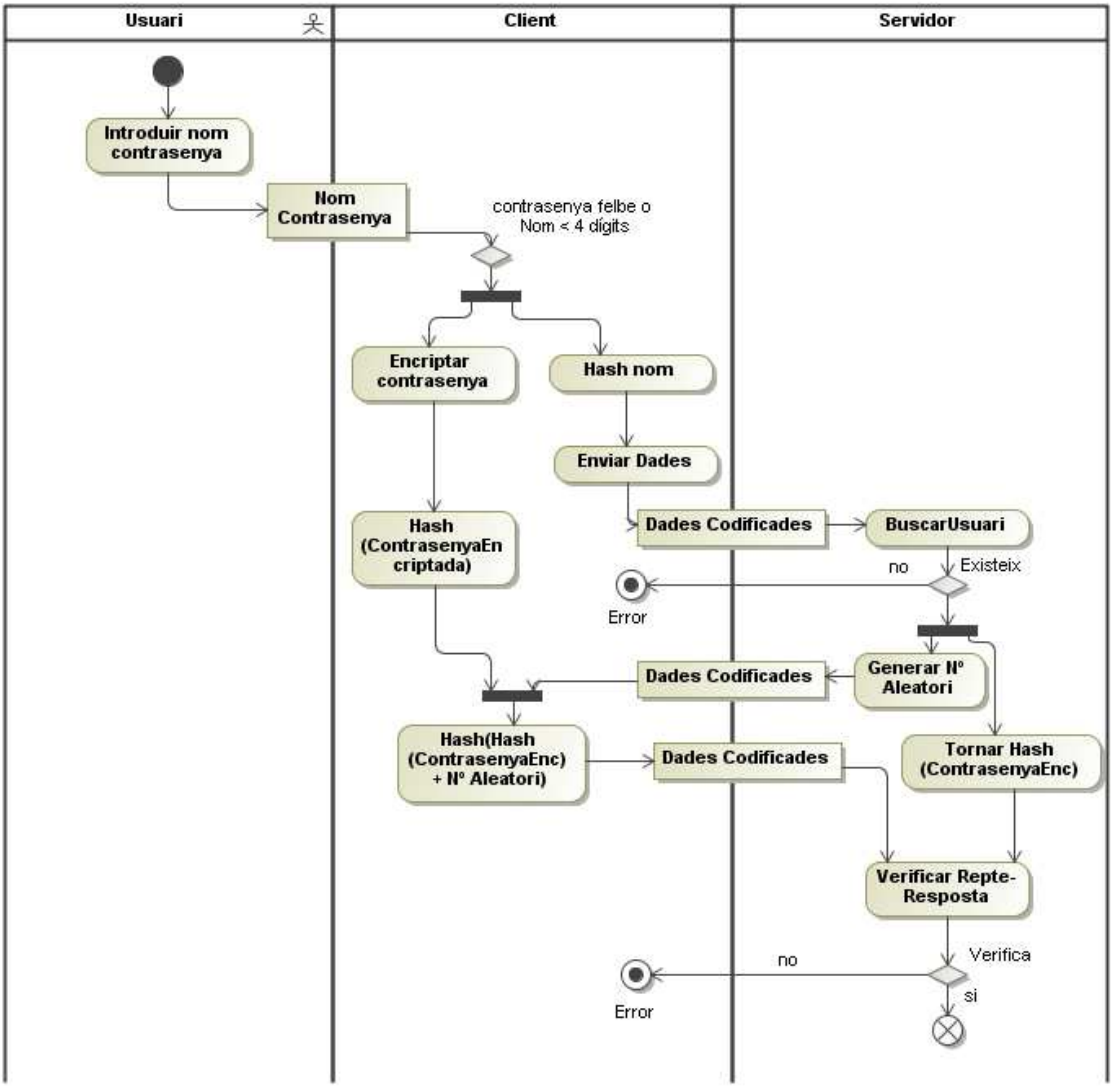

Figura 2 – Diagrama d'activitats Inici de sessió

Aclarir que a "buscar usuari" s'efectuarà una cerca dins la taula d'usuaris on es compleixi que la dada enviada pel client més el camp de bits de salt emmagatzemat sigui cert, per després amb el número aleatori de sessió, verificar si la contrasenya compleix el repte resposta esperat, que tracta de dificultar l'atac en cas d'haver-hi algú "escoltant" la xarxa.

4.6 Xifratge de dades

Com ja s'ha comentat a l'apartat de la fase de disseny sobre els protocols criptogràfics, pel xifratge de les dades de serveis s'ha optat per AES amb la contrasenya principal com clau.

4.7 Disseny de la base de dades

Utilitzant MySQL es crearà una base de dades amb dues taules, una on es guardarà les dades de registre d'usuari, que són el camp únic d'identificador d'usuari, que com s'ha indicat l'apartat de protocol criptogràfic, és el hash del hash del nom d'usuari més els bits de salt, els bits de salt i el hash de la contrasenya principal encriptada. A l'altre, es desaran els serveis a gestionar dins els camps nom del servei, adreça, usuari del servei, contrasenya i identificació d'usuari, vincle amb la taula anterior i que juntament amb nom de servei formaran una clau única d'aquesta taula.

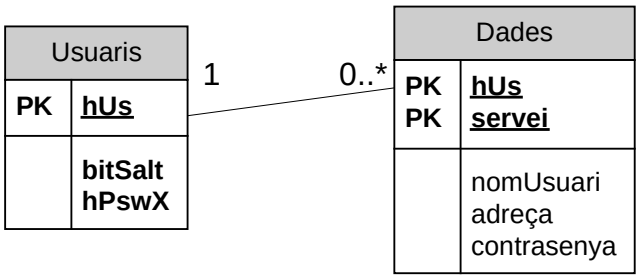

Figura 3 – Relació bases de dades

5 Implementació

5.1 El servidor

Per aquest entorn de proves i tenint en compte el maquinari a l'abast, s'ha optat per fer servir un ordinador amb Windows XP com servidor, amb l'aplicació WAMP Server, que aglutina els serveis d'un servidor Apache, MySQL i PHP, integrats i pràcticament a punt per funcionar sense gaires modificacions, com ara habilitar el servei segur SSL, modificar la contrasenya per defecte del super usuari root a MySQL.

5.2 La base de dades

S'ha creat una base de dades amb un únic usuari restringit que només pot executar rutines prèviament creades per l'administrador, que farem servir per manegar les dades contingudes a les dues taules, com ara afegir, modificar o eliminar registres des de l'aplicació, per tractar d'evitar el risc de les injeccions de codi SQL, tècnica utilitzada per obtenir accés a les dades o la seva eliminació.

5.3 Generació dels certificats

Per la implementació del servidor segur SSL, fent servir l'aplicació openssl del servidor Apache, s'ha generat un parell de claus RSA de 2048 bits, xifrades amb triple DES, amb què posteriorment autosignar el certificat X.509, on s'ha inclòs entre altres paràmetres el nom virtual del lloc, *www.projecte_p_pas.cat*, com "nom comú" del certificat que assegura la identitat del servidor.

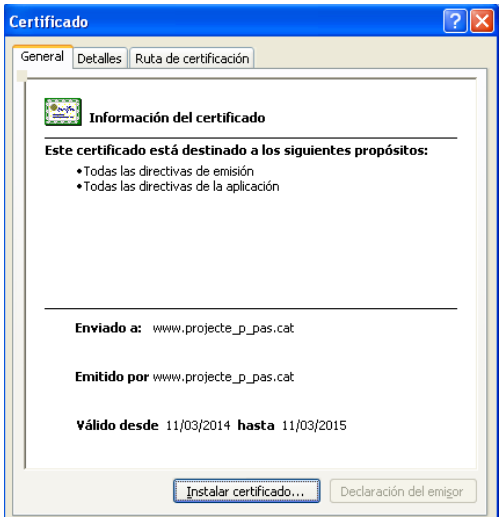

Figura 4 – Captura de la informació general del certificat instal·lat

5.4 Implementació de pàgines

Seguint amb la finalitat d'utilitzar l'eina en el major tipus de dispositius i sistemes, així com no sobrecarregar ni el disseny ni els complements o programes addicionals pel funcionament, es plantegen totes les pàgines només amb els textos imprescindibles i botons essencials per al funcionament, centrant els esforços en la programació de les funcions i la seguretat, prioritat principal d'aquest treball.

Figura 5 – Pàgina principal

5.4.1 Registre d'usuari nou

En el moment del registrar, el nou usuari introduirà un nom d'usuari, més llarg de 4 dígits, i la contrasenya principal, que s'avaluarà amb la funció de càlcul de fortalesa assignant-li un número del 0 al 10 indicant el nivell i jugant amb el color de la font, vermell si va de 0 a 3, groc de 4 a 6, verd de 7 a 10, **no acceptant contrasenyes febles,** també s'inclourà un text amb les recomanacions bàsiques per a la creació de la contrasenya principal.

Figura 6 – Pàgina de registre d'usuari

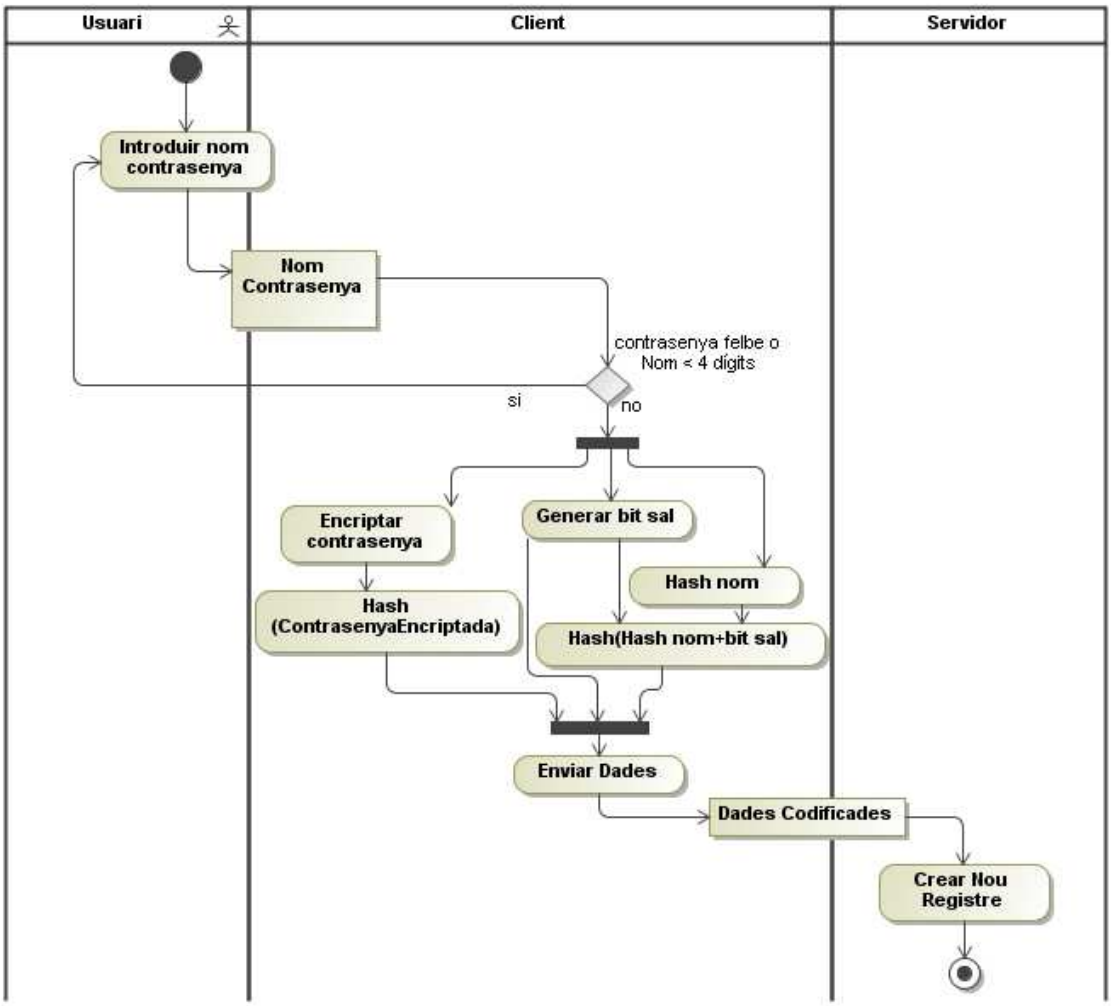

Figura 7 – Diagrama d'activitats del registre d'usuari

5.4.2 Avaluació de fortalesa de la contrasenya

Com s'ha indicat a la fase d'anàlisi no es permetrà l'ús de contrasenyes més curtes de 8 dígits i criteri de complexitat inferior a 4, segons el càlcul simplificat de l'entropia especificat, que incentiva per un costat l'ús dels diferents elements, majúscules, minúscules, números i símbols, oferint un "alfabet" major, i per altre claus més llargues.

5.4.3 Alta i modificació de dades

Pel que fa a l'alta i modificació d'informació a emmagatzemar, després d'iniciada la sessió, es mostrarà una pàgina amb el botó "Nou servei" per donar d'alta un registre nou o seleccionar el botó "Modificar" d'un dels registres guardats per poder modificar el contingut tornant-se a encriptat en desar-se.

Figura 8 – Pàgina llista de serveis

5.4.4 Introducció de dades a guardar

Inicialment es planteja crear un únic formulari per introduir les dades noves amb els camps: servei, adreça, nom d'usuari, com camps de text, i contrasenya amb la funció d'avaluació de contrasenya, on una vegada emplenat serà encriptat.

Figura 9 – Pàgina d'introducció de dades

5.4.5 Eliminació de dades

Com al cas anterior, mitjançant un botó “Eliminar” al costat del registre desitjat es podrà sol·licitar l’eliminació de les dades, mostrant una pantalla d’advertència d’esborrat i demanant la confirmació per evitar l’acció accidental.

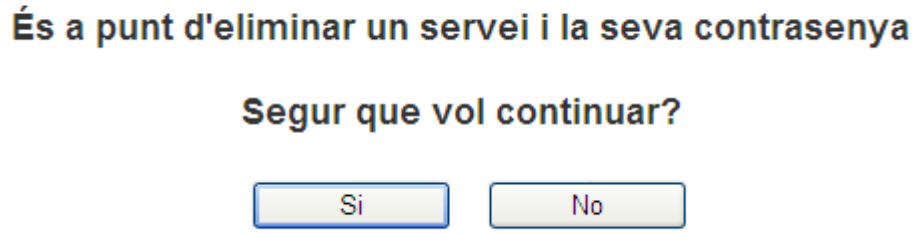

Figura 10 – Missatge de confirmació d’eliminació

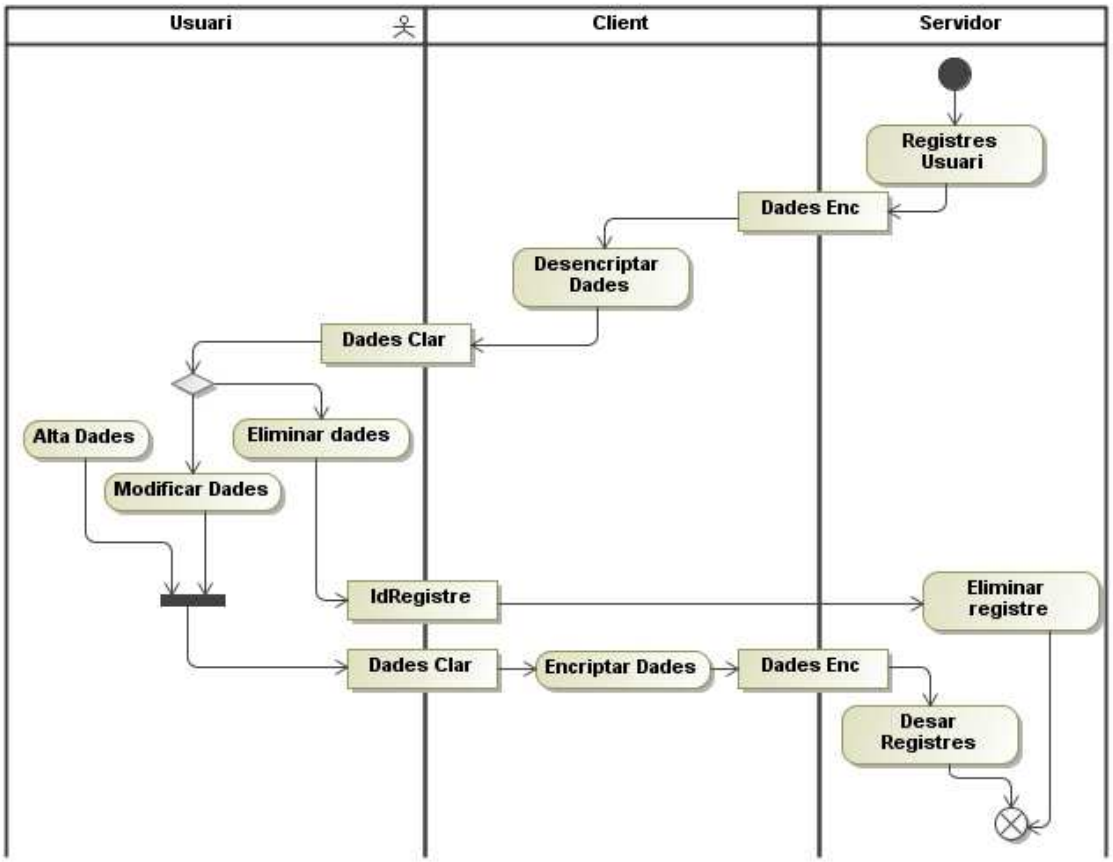

Figura 11 – Diagrama d’activitats gestió de registre de dades

5.5 Programar funcions

Basant-nos en l'estàndard HTML i la seva compatibilitat amb el diferents navegadors s'ha utilitzat aquest llenguatge com la base per programar les funcions del costat client afegint funcions de llibreries JavaScript, ja siguin ja existents, com ara "crypto-js", referents als protocols i funcions d'encriptació, o creant noves com ara la que avalua el criteri de fortalesa de la contrasenya segons el nostre projecte en particular.

Per altre costat i amb la finalitat tractar de centrar totes de dades sensibles, com ara la informació sense codificar, a la pàgina principal però amb la necessitat d'enviar i rebre dades al servidor s'han utilitzat algunes tècniques i utilitat d'AJAX (Asynchronous JavaScrip And XML) i interfície DOM (Document Object Model).

En referència a les funcions pròpies del servidor, com ara les dades se sessió o interacció amb les bases de dades SQL s'ha emprat PHP. Pel que fa a les bases de dades, s'ha optat per gestionar-les via rutines MySQL, on s'envien o es reben les dades com paràmetres d'entrada o sortida sense la possibilitat de que, més enllà de l'administrador, es pugi modificar les consultes indegudament.

5.5.1 Funció d'encriptació d'alta d'usuari i inici de sessió

Atent al protocol explicat a l'apartat 4.5 sobre l'autenticació i més en concret a l'alta d'usuari, s'ha optat per utilitzar la funció CryptoJS.AES de la llibreria JavaScript crypto-js però amb la codificació en UTF8 del nom d'usuari com "clau" i el hash d'usuari com vector inicial, ja que en aquest cas els bits de salt de la funció més els propis inclosos al procés dificultaven la posterior comprovació durant l'inici de sessió, on s'ha utilitzat els mateixos paràmetres a CryptoJS.AES.

6 Proves

6.1 Registre d'usuaris, ús de l'aplicació

A les primeres proves s'ha creat un parell d'usuaris amb els corresponents serveis. Per tractar de valorar millor la usabilitat, s'ha sol·licitat a dues persones alienes al desenvolupament del projecte la seva col·laboració destacant la percepció per part de totes dues d'un inconvenient en el fet de tenir que complir amb els requeriments de fortalesa de la contrasenya, tenint que explicar a una de elles què s'entén per "símbol".

Pel que fa a maquinari i navegadors, s'ha avaluat els processos a un portàtil amb Firefox i un sobretaula amb Internet Explorer 8, Firefox, Google Chrome.

6.2 Comprovació de la inintel·ligibilitat de les dades

Tot i accedir com administrador a les taules i extreure el seu contingut, com es pot comprovar a les taules adjuntes, les dades no poden ser interpretades complint-se una de les premisses bàsiques d'aquest projecte.

hUs	bitSalt	hPswX
9cb5c22bbe4878e3da581eef9ed0aef0247f68551bb3a675ce6f49eeca03cbae	9718301624525330	3be2e6fcf4f5c786a1549d5a045c667970ee06f3f0ddc05db7
a0444a564ac73b89eb13500008b2b406e4419a6ec2a340326b52d2feaa6e9c40	972425595391541	aaa05f5df2c2d7b62e41cc2cb926929a84f136b812dc60e218

Figura 12 – Captura contingut taula "usuaris"

En el cas dels serveis s'ha utilitzat la funció CryptoJS.AES només amb el text a codificar, que són servei, adreça, usuari i contrasenya, i la paraula de pas mestra com clau de codificació, generant la mateixa funció el vector inicial i bits de salt necessaris per generar un objecte de tipus matriu de paraules que compleix amb el format OpenSSL, fet que propicia que tot i codificar el mateix text amb la mateixa clau el resultat sigui diferent, bàsicament per la aleatorietat dels bits de salt, ocultant aquest fet.

Servei	U2FsdGVkX18uCt6Dag5yJOv5qE+xpRu6BGU4yaGuA8U=	U2FsdGVkX19dCWZT3K83am1Cggj0LEIoXfz+qdFRUeA=
Adreça	U2FsdGVkX18eJI5JhTasVLEy5WWCfpnN/KeooLsrxlQ=	U2FsdGVkX19BP5Qzb2eYLasMTky8R1PQwBUtDfvZtwk=
Usuari	U2FsdGVkX1973T63XuEKRaF6qQ+yfyUqS9iJ40Zga68=	U2FsdGVkX1/ZOA4O9Uf5lmfkls3o60cPwEfWFMTB0T0=
Contrasenya	U2FsdGVkX198yimuaP7+KXonVZQOZDSl0oL4Og4H8hg=	U2FsdGVkX1/dd8ZzL3qh9VwZe3AhNPVVT1G3n/NC5yI=

Figura 13 – Captura contingut taula "serveis"

6.3 Comprovació de la privacitat de les comunicacions

Tenint en compte l'entorn de proves amb adreces IP "privades" (192.168.2.0/24), només s'ha comprovat amb l'analitzador de protocols "Wireshark" el correcte establiment de les comunicacions via https una vegada instal·lat el certificat al client, quedant pendent una "escolta" més en detall a les diferents etapes, focalitzant l'alta d'un nou usuari i l'inici de sessió.

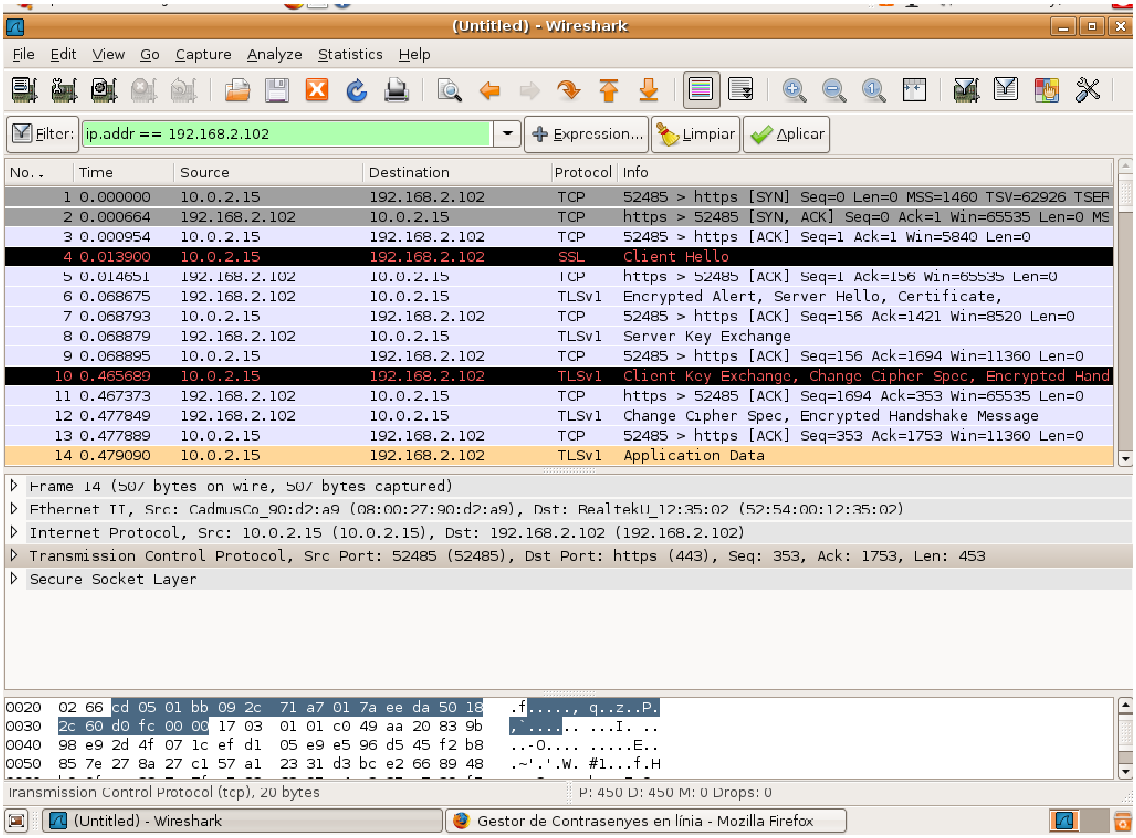

Figura 14 – Captura de paquets

7 Conclusions

Arribat a aquest punt on finalitza el temps planificat per portar a terme aquest treball, existeix la sensació de que si bé s'han complert la majoria d'objectius fixats, s'han dedicat més recursos dels estimats per assolir-los, potser en gran mesura pel fet d'haver d'invertir temps addicional en conèixer les tecnologies i recursos existents actualment respecte al moment en que es va tracta a l'assignatura en qüestió, i més concretament en referència a l'entorn JavaScript i la seva vessant asíncrona AJAX.

Per altre banda, es valora positivament el fet d'afermar altres conceptes com ara les bases d'encriptació, l'anàlisi del concepte d'entropia, que finalment ha estat incorporada de manera simplificada al criteri de fortalesa de la contrasenya principal, i aprendre una nova utilitat sobre gestió de bases de dades com són les rutines SQL, que ofereixen més seguretat i menys riscos, en cas de que aquestes s'hagin de posar a disposició d'un usuari final.

7.1 Propostes de futures millores o funcionalitats

Amb l'objectiu de continuar desenvolupant i millorant aquest prototip de producte, i tractant de concretar alguns punts, més enllà d'un repàs general en profunditat després de l'ús per part de més usuaris, es proposa:

- Gestionar l'eliminació d'usuaris o bé de manera activa, on una vegada iniciada la sessió es pugui clicar un botó *"Donar-se de baixa"* i després dels missatges d'advertència i corresponents confirmacions, s'esborrin les dades tant d'usuari com dels seus serveis, o bé com una rutina de manteniment del servidor, on afegint un camp a la taula d'usuaris tipus "darrer accés" i transcorregut un període de temps, un any per exemple, executi una rutina en el servidor que les elimini.

- Crear una funció que generi contrasenyes aleatòries, amb el criteris de fortalesa ja definits, que una vegada acceptades puguin ser utilitzades als serveis.

- Mostrar inicialment a la taula de serveis els camps de contrasenyes inintel·ligibles per evitar que algú les pugui visualitzar "per sobre de les espatlles", fent-les llegibles en fer clic a sobre i ocultant-se en clicar per segon cop.

Bibliografia

Instituto Nacional de Técnologias de la Comunicación. “Listado de útiles gratuitos – Gestor de contraseñas”. [En línia], [Consultada el 5/4/2014]
http://cert.inteco.es/software/Proteccion/utiles_gratuitos/Utiles_gratuitos_listado/?idLabel=2230254&idUser=&idPlatform=

Wikipedia.(2014). “Password Strength”. [En línia], [Consultada el 7/4/2014]
http://en.wikipedia.org/wiki/Password_strength

Wikipedia.(2014). “Entropy (Information Theory)”, [En línia]. [Consultada el 7/4/2014]
http://en.wikipedia.org/wiki/Information_entropy

Jordi Herrera Joancomartí, Joaquín García Alfaro, Xavier Perramón Tornil. (2008). “Seguretat en Xarxes de Computadors”. Barcelona: FUOC.

Wikipedia. (2014). “MySQL”. [En línia], [Consultada el 8/4/2014]
http://es.wikipedia.org/wiki/MySQL

Wikipedia. (2014). “Firebird”. [En línia], [Consultada el 8/4/2014]
http://es.wikipedia.org/wiki/Firebird

Instituto Nacional de Técnologias de la Comunicación. “Cómo crear una contraseña segura”. [En línia], [Consultada el 10/4/2014]
http://cert.inteco.es/Proteccion/Recomendaciones/Crear_una_contrasena_segura/

AMPLE Scaffolder software. (2010). “The password meter”. [En línia], [Consultada el 10/4/2014]
http://www.passwordmeter.com/

Josep Domingo Ferrer, Jordi Herrera Joancomartí, Helena Rifà Pous. (2006). “Criptografia”. Barcelona: FUOC.

Benet Campderrich Falgueras. (2004). “Enginyeria del programari”. Barcelona: FUOC.

Piero Berni Millet, Dídac Gil de la Iglesia. (2010). “Laboratorio de PHP y MySQL”. Barcelona : FUOC.

Miller González Leon. “Como crear y configurar un certificado SSL en WAMP para navegación segura por HTTPS”. [Blog, 22/7/2012], [Consultada el 21/04/2014]
http://millertaker.blogspot.com.es/2012/07/como-crear-y-configurar-un-certificado.html

Cripto-JS. JavaScript implementations of standard and secure cryptographic algorithms
https://code.google.com/p/crypto-js/

The PHP Documentation Group. “Manual de PHP”. [En línia], [Consultada el 9/5/2014]
http://www.php.net/manual/es/index.php

W3Schools. “JavaScript Tutorial”. [En línia], [Consultada el 2/5/2014]
http://www.w3schools.com/js/default.asp

W3Schools. “Jquery Tutorial”. [En línia], [Consultada el 9/5/2014]
http://www.w3schools.com/jquery/default.asp

W3Schools. “AJAX Tutorial”. [En línia], [Consultada el 9/5/2014]
http://www.w3schools.com/ajax/default.asp

W3Schools. “AJAX JSON”. [En línia], [Consultada el 9/5/2014]
http://www.w3schools.com/json/default.asp

The PHP Group. “Manual de PHP”. [En línia], [Consultada el 12/5/2014]
http://es1.php.net/manual/es/index.php

ORACLE. “MySQL 5.5 Reference Manual”. [En línia], [Consultada el 30/04/2014]
http://dev.mysql.com/doc/refman/5.5/en/

The PHP Group. “Inyección de SQL”. [En línia], [Consultada el 12/5/2014]
http://www.php.net/manual/es/security.database.sql-injection.php

Tarlogic Security. “Cómo generar sesiones en PHP de forma segura”. [Blog, 7/9/2012], [Consultada el 3/6/2014]
https://www.tarlogic.com/como-generar-sesiones-en-php-de-forma-segura