Autenticació d'alumnes en un aula sense Internet

Roger Huertas

Resum– L'autenticació sense Internet dels dispositius mòbils dels alumnes d'una classe de secundària. Aquest és l'objectiu que s'ha aconseguit en el desenvolupament del projecte denominat Auth4Classroom. En aquest informe s'indica quin és el mecanisme d'autenticació utilitzat i amb quines tecnologies disponibles en dispositius mòbils s'ha solucionat la problemàtica anterior. Tenint present els coneixements tècnics dels usuaris finals, hem desenvolupat una solució que no fos complexa d'utilitzar. A més s'ha pensat en un sistema que pugui afegir nous mecanismes d'autenticació en un futur. El resultat aconseguit és una eina de software que permet autenticació offline.

Paraules clau– Autenticació, Offline, Seguretat, Mòbil, Wi-Fi, Wi-Fi Direct, P2P, QR, NFC, Android, MAC

Abstract– Authentication of the mobile devices of a classroom of high-school students. Without Internet. That is the goal achieved during the development of the project called Auth4Classroom. In this report we will explain what the authentication scheme used is, and the technologies used for solving the problem we initially faced. Having in mind the end-user's tech knowledge, we've developed a solution that is not very complex to use. At the same time it's done in a way that allows for new authentication mechanisms to be developed and introduced, exapanding the available options. The end result is a software tool that allows offline authentication.

Keywords– Authentication, Offline, Security, Mobile, Wi-Fi, Wi-Fi Direct, P2P, QR, NFC, Android, MAC

1 Introducció

Tenim el següent problema. Un professor de secundària vol que els seus alumnes utilitzin els seus dispositius mòbils per realitzar deures. Per fer-ho, cal una aplicació mòbil que sigui capaç d'autenticar als alumnes (i només als seus alumnes) presents a l'aula. Un cop tingui els alumnes autenticats podrà enviar preguntes i exercicis per fer a casa. Tant els alumnes com el professor tots disposen de dispositius mòbils, però no disposen d'Internet a classe, o com a molt el professor és l'únic que en disposa (Wi-Fi).

En aquest treball explicarem com hem dut a terme el projecte que dóna solució a aquesta situació. Ens centrarem exclusivament en el desenvolupament d'una proposta que solucioni el problema d'autenticació plantejat. Definim un mecanisme d'autenticació que es simple d'utilitzar i proporciona un mínim grau de seguretat. En particular es vol evitar que el mecanisme d'autenticació sigui susceptible a atacs de suplantació d'identitat (Man In The Middle attacks). Aquests atacs ens afectarien en cas que utilitzéssim algun mecanisme de connexió amb xarxes mòbils (com ara Wi-Fi o Bluetooth) i podrien tenir conseqüències a les notes dels alumnes. En la següents seccions indiquem quins són els objectius a assolir en el projecte que busca solucionar el problema descrit. Seguidament explicarem les tecnologies utilitzades i quina informació rellevant de referència hem revisat. Indicarem la metodologia utilitzada pel desenvolupament del projecte, i descriurem la proposta plantejada i com hem dut a terme la seva implementació. Farem èmfasi en l'apartat d'autenticació, explicant quins mecanismes hem plantejat i utilitzat. També distingirem en quina és la funcionalitat nucli (referent a l'autenticació) de la nostra aplicació i la funcionalitat que està més relacionada amb la interfície. Finalment avaluarem els objectius i resultats assolits i indicarem les conclusions obtingudes.

• E-mail de contacte: roger.huertas@e-campus.uab.cat
• Menció realitzada: Tecnologies de la Informació
• Treball tutoritzat per: Àngela Fabregues (TIC)
• Curs 2016/17

Objectiu	Prioritat
1. Definir un mecanisme d'autenticació	Prioritari
2. Implementar autenticació tecn. mòbils	Prioritari
3. Identificar als alumnes individualment	Important
4. Implementar la proposta en Android	Important
5. Implementar la proposta en iOS	Opcional

TAULA 1: PRIORITAT DELS OBJECTIUS

2 OBJECTIUS

L'objectiu principal del projecte Auth4Classroom és permetre que el dispositiu mòbil del professor autentiqui als seus alumnes (i només als seus alumnes presents a l'aula) mitjançant una aplicació mòbil o similar. Una vegada autenticats els alumnes, el professor podrà publicar preguntes. Els alumnes podran visualitzar aquestes preguntes en els seus dispositius mòbils, i enviar les respostes al professor. Aquest objectiu principal el podem desglossar en els següents objectius:

1. Definir un mecanisme d'autenticació que pugui funcionar sense necessitat d'Internet.

2. Implementar aquest mecanisme d'autenticació amb les tecnologies mòbils disponibles.

3. Poder identificar als alumnes (i el seu dispositiu mòbil) individualment.

4. Programar la solució escollida pel sistema operatiu Android.

5. Opcionalment, si és viable, fer l'aplicació compatible amb sistema operatiu iOS.

Cal recalcar la importància dels 3 primers objectius. Sobre tot és molt important identificar als alumnes, ja que es volen evitar casos no desitjats, com ara per exemple que un alumne vulgui fer trampa fent-se passar per un altre alumne per respondre un exercici. En quant a l'apartat més tècnic (relacionat també amb la identificació dels alumnes) volem evitar atacs de suplantació d'identitat, és a dir, que un dispositiu/usuari es faci passar per un altre.

A la taula 1 s'indica la prioritat de cadascun dels objectius.

3 ESTAT DE L'ART

Hi han diverses aplicacions que gestionen la publicació de preguntes/exercicis perquè els alumnes les puguin respondre. Dues de les més conegudes són Kahoot i Socrative. Malauradament la primera es tracta d'una aplicació web i necessita Internet. A més, no autentica de cap manera als alumnes que responen les preguntes, sinó que simplement permet accedir a l'aula virtual a tothom que tingui el codi d'accés, i al final d'una sèrie de preguntes, mostra el resultat. Socrative en canvi disposa d'app mòbils, però requereixen connexió a Internet. Funciona de manera similar a Kahoot en el sentit de què el professor crea una aula virtual. Aquí els alumnes sí que es registren a l'aula i quan responen a les preguntes el professor pot veure qui ha respost, però no hi ha autenticació. A més també necessita Internet per funcionar.

Hem analitzat diversos mètodes d'autenticació i altres tecnologies que ens podien servir de referència pel desenvolupament. Un dels mètodes d'autenticació estudiats és el conegut com a Universal 2nd Factor (U2F) [1], que consisteix en una autenticació de dos factors. La seva especificació i implementació open-source definida a Yubico ens ha permès obtenir molta informació valuosa. Aquest tipus d'autenticació es basa en un factor físic per autenticar al client. És a dir, el client disposa d'alguna mena de clau que només pot tenir en un dispositiu físic (en el cas de Yubico es tracta d'un USB). Necessita aquesta clau per poder autenticar-se, i com que únicament autenticarem dispositius que disposin d'aquest clau, estem augmentant molt el nivell de seguretat.

Wi-Fi Direct i la seva implementació de la tecnologia Peer-to-Peer (abreviat P2P) [2] [3] ens ha permès estudiar com implementar una xarxa ad-hoc sense necessitat d'Internet. Bàsicament, ens permet formar grups de dispositius amb la tecnologia P2P, on un dispositiu és el servidor i tots els altres actuen com a clients. Tots els establiments de connexió no necessiten internet, ja que es basen en Wi-Fi Direct, una tecnologia basada en Bluetooth que estableix xarxes Wi-Fi entre dispositius, sense necessitat d'un punt d'accés (PA) que actuï com a intermediari. Aquesta tecnologia està present en dispositius Android, i conté múltiples guies sobre la seva implementació [4].

Vam estudiar la viabilitat de la tecnologia NFC com a mecanisme d'autenticació offline [5]. També vam trobar una API (Application Programming Interface) web de lectura i escriptura de NFC tags [6]. Malauradament es tracta d'un esborrany (draft) que encara es troba en desenvolupament. Es díficil pensar en un dispositiu mòbil avui dia que no disposin d'almneys una càmera. A més de utilitzar-se per fer fotografies, la càmera es pot utilitzar per escanejar codis de barres. Vam decidir que seria bona idea utilitzar codis QR per la transmissió d'informació, i aprofitar la càmera d'aquests dispositius per escanejar-los. La llibreria open source ZXing [7] ens permet integrar la funcionalitat de generació i escaneig de QR amb el dispositiu mòbil.

4 METODOLOGIA

En aquesta secció descriurem breument quina és la metodologia que hem utilitzat per al desenvolupament del projecte. La majoria de metodologies orientades a desenvolupament de projectes de software, parteixen de l'assumpció del fet que hi ha diversos membres a l'equip (o almenys diversos rols duts a terme per diverses persones, on es pot donar el cas que un mateix membre de l'equip pugui tenir diversos rols) i estan enfocades a potenciar el treball en equip, la comunicació i la coordinació. Exemples d'aquestes metodologies serien Scrum o Lean Software Development (LSD).

Com que en aquest projecte només hi ha un únic membre en l'equip desenvolupador, que hauria de cobrir tots els rols que s'especifiquen en aquests tipus de metodologies, a la pràctica no té massa sentit aplicar-les, ja que possiblement l'esforç d'adaptar completament aquesta metodologia d'equip per un sol membre pot representar una tasca molt més complexa que el mateix desenvolupament, cosa que no resultaria gaire viable.

Per tant en comptes d'escollir una metodologia existent, ens vam inspirar en alguns elements que funcionen bé en aquests tipus de metodologies i que no requereixen la presència de diversos membres d'equip per especificar les pautes a seguir.

S'ha utilitzat un concepte similar al "Sprint"de Scrum. La manera de funcionar es la següent. Hi ha un cronograma d'activitats global, però cada mes s'especifiquen una sèrie de tasques a realitzar.Es fa un seguiment de quan s'han realitzat aquestes tasques i qualsevol inconvenient sorgit. En finalitzar el mes s'avaluen les tasques realitzades, quines tasques no s'han realitzat i per què, i s'indica qualsevol canvi de planificació o inconvenient sorgit.

Hem pogut disposar de les instal·lacions de l'Institut d'Investigació en Intel·ligència Artificial (IIIA-CSIC [8]) pel desenvolupament del projecte. Al servidor web de l'IIIA-CSIC es troba una pàgina on s'indiquen les instruccions per instal·lar i utilitzar la solució implementada. Pel que fa al desenvolupament del codi, s'han realitzat diversos tests unitaris amb Java JUnit 4, utilitzant la técnica de Test Driven Development per desenvolupar la funcionalitat nucli de l'aplicació. El codi está disponible al github de l'IIIA-CSIC.

5 AUTENTICACIÓ

Pel que fa a l'autenticació, vam decidir basar-nos en les tecnologies següents: Wi-Fi P2P i QR [9]. Tenint en compte això, vam dissenyar una seqüència d'autenticació que pogués servir de base per ambdós mètodes. Ens vam inspirar en el protocol d'autenticació U2F. Donats dos dispositius mòbils. Un dispositiu del professor i un dispositiu de l'alumne. Pel que respecta a l'autenticació ens referirem al dispositiu de l'alumne com a "Client" i al dispositiu del professor com a "Servidor".

La figura 1 mostra l'esquema d'autenticació. A l'esquerra tenim l'alumne (Student), al centre tenim l'aplicació client i a la dreta tenim el servidor que farà les tasques de CA (Certification Authority) en el procés d'autenticació. L'alumne polsarà un botó de connexió i el seu dispositiu establirà connexió amb el servidor. L'establiment d'aquesta connexió serà mitjançant alguna tecnologia de xarxa mòbil disponible.

Una vegada hagi finalitzat l'establiment de la connexió, el dispositiu client enviarà la informació de l'alumne. Aquest procés l'hem denominat handshake. El servidor utilitzarà aquesta informació per revisar i identificar si aquest alumne i dispositiu ja s'han autenticat prèviament. En cas afirmatiu finalitza el procés d'autenticació. En cas negatiu, és a dir, encara no s'ha autenticat aquest dispositiu, l'hem d'autenticar.

Per poder autenticar el dispositiu des del servidor generarem el que anomenarem un challenge d'autenticació. Challenge en anglès vol dir desafiament, per tant traduït literalment sería "desafiament d'autenticació". Aquest challenge pot ser moltes coses: una pregunta, un número, una contrasenya... Una vegada generat aquest challenge, volem enviar-lo al dispositiu client. No obstant volem evitar casos no desitjats com ara que s'intercepti i/o es modifiqui aquest challenge. Per tant haurem de tenir això present a l'hora d'enviament de dades per la xarxa. La forma en què hem solucionat aquest inconvenient és tenint mecanisme d'autenticació que requereixen que els dispositius estiguin a prop l'un de l'altre. Una vegada s'ha enviat el challenge d'autenticació al dispositiu client, aquest generará la resposta d'autenticació. Aquesta resposta d'autenticació pot ser tenir un contingut tan senzill com el propi challenge o pot contenir informació addicional. Quan el dispositiu servidor rebi la resposta verificarà que sigui correcta. En cas afirmatiu l'autenticació es produeix amb èxit. Si no s'ha arribat a aquest pas o el missatge de resposta rebut és diferent, l'autenticació falla.

Basant-nos en aquest esquema, hem desenvolupat dos mecanismes d'autenticació diferents. Un és l'autenticació mitjançant la tecnologia de Wi-Fi Direct P2P i l'altre és mitjançant l'escaneig i generació de codis QR.

5.1 Autenticació P2P

Inicialment vam pensar a requerir que els dispositius que volien ser autenticats mitjançant P2P, haguéssin d'enviar una clau que no se'ls havia enviat prèviament, però va resultar ser poc pràctic, per les seguents raons:

- Un dels requeriments que ens vam marcar és que la clau ha de ser la més senzilla possible. Aquest requeriment és a causa del fet que escriure al mòbil (sobretot símbols i números) és generalment molest i poc còmode. A més volem que la interacció per part de l'alumne hagi de ser mínima. Això significa que no podem tenir una clau que obligui als alumnes a introduir un gran nombre de caràcters com a resposta d'autenticació. Es trigaria molt de temps i seria molt enrevessat.

- Encara que pot semblar que és una obvietat, requereix que els alumnes introdueixin la clau per poder autenticar-se. Ens podem trobar alumnes no col·laboratius, que no acabin el mecanisme d'autenticació, causant diversos problemes. Un dels problemes més significatius seria l'augment de l'overhead computacional del dispositiu mòbil del professor (el servidor) que tindria connexions obertes i inactives, que no s'utilitzen per res.

- Se'ns complica el control d'errors perquè hem de fer front al problema de claus mal escrites. La solució més senzilla (que no la més òptima) per aquest problema és que els alumnes tornessin a introduir la clau. Encara i així no podem solucionar la inanició, que significa que hi pot haver dispositius d'alumnes que mai s'autentiquin perquè introdueixen sempre la clau de forma errònia.

Per tant hem d'idear un sistema que pugui autenticar als alumnes sense que aquests hagin d'intervenir, o intervinguin mínimament. Això ho hem fet eliminant la necessitat dels alumnes d'introduir la contrasenya. Així doncs la solució d'autenticació proposada segueix la següent seqüència:

1. El dispositiu del professor (servidor) escoltarà peticions de connexió Wi-Fi P2P.

2. Els dispositius alumne (clients) iniciaran la connexió amb el dispositiu del professor pressionant un botó. Sabran quin és el dispositiu al qual han de connectar-se, perquè el professor indicarà quin és el nom del seu

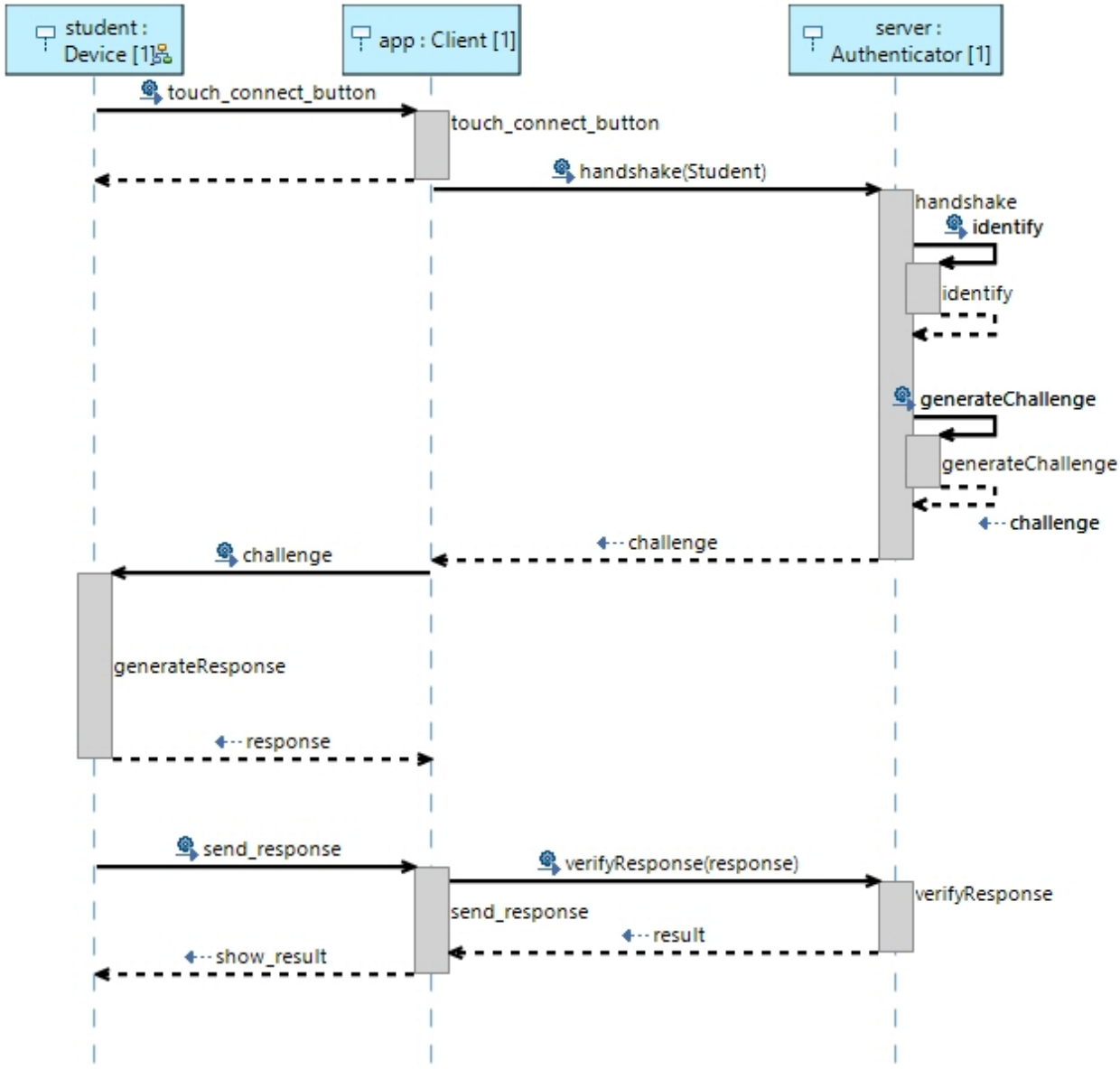

Fig. 1: Esquema d'autenticació

dispositiu. Per aquest motiu vam decidir incloure un mecanisme que permetés tant a alumnes com a professor canviar aquest nom.

3. Una vegada connectats els dispositius, l'alumne encara s'ha d'autenticar. Al dispositiu de l'alumne l'hi apareix un nou botó per autenticar-se. Al pressionar aquest botó, s'envia un missatge de salutació a l'altre dispositiu, el del professor. En aquest missatge s'indica quin estudiant és i les dades del seu dispositiu.

4. El dispositiu del professor rep aquest missatge i comprova si aquest estudiant s'ha autenticat prèviament, amb la informació del seu dispositiu. Gràcies a l'identificador únic, la MAC, podem identificar als dispositius de forma única. Si no s'ha autenticat, se li envia (a l'alumne) el challenge d'autenticació. Aquest challenge consisteix en una contrasenya numèrica. A més a més del challenge enviem les dades del professor i del seu dispositiu. Si el dispositiu client ja estava autenticat, no fem res i tanquem la connexió.

5. El dispositiu de l'alumne rep aquest missatge. Comprova que el dispositiu emissor té la mateixa MAC que s'indica al contingut del missatge. Si no hi ha cap diferència o error en el contingut del missatge, es torna a enviar la contrasenya al dispositiu emissor.

6. El dispositiu professor rep aquesta contrasenya i si és correcte, l'autenticació es duu a terme amb èxit. Les dades de l'alumne es modifiquen i actualitzen per les rebudes. En cas que el dispositiu del professor no hagi "seleccionat" cap alumne, se'n crea un de nou i es guarda a la llista. A les seccions següents s'explica amb més detall com s'ha implementat la interfície de l'aplicació.

Tot l'intercanvi de missatges es fa xifrat. Així doncs tenim la certesa que altres usuaris que interceptin les comunicacions no poden veure el contingut dels missatges.

Per eliminar els atacs de Man In The Middle (un dispositiu que intenta suplantar a un altre en el procés d'intercanvi de missatges) hem inclòs la informació dels dispositius en

els missatges que s'envien. Al rebre un missatge, el receptor comprova que efectivament el dispositiu que l'hi ha enviat el missatge es tracta del mateix que s'indica en el seu contingut (es comprova que la MAC dins del missatge coincideix amb la de l'emissor). En cas contrari l'autenticació falla i es tanca la connexió.

5.2 Autenticació QR

Per dissenyar l'autenticació mitjançant QR hem hagut d'utilitzar la llibrería open source ZXing. Aquesta llibreria conté la funcionalitat necessària per codificar missatges en QR i descodificar-los. A més a més té la funcionalitat requerida per integrar l'escaneig de QR amb la càmera dels dispositius mòbils. En aquest cas vam modificar una mica la seqüència d'autenticació respecte a la visió inicial, per fer-la més senzilla. Inicialment haviem pensat que el dispositiu del professor generaria el challenge d'autenticació (generaria un QR) i el dispositiu de l'alumne enviaria aquest challenge via Wi-Fi P2P. Al realitzar proves d'implementació vam veure que era massa complicació. Es va optar per una seqüència d'autenticació més senzilla, que és la següent:

1. El dispositiu del professor genera un QR amb un missatge. Aquest missatge conté el challenge d'autenticació, que es tracta d'una passphrase (contrasenya) generada aleatòriament. També introduïm les dades del professor en el missatge.

2. El dispositiu de l'alumne escaneja aquest QR. Al descodificar el QR es guarden les dades del professor, ja ens interessa tenir una mínima informació de qui ens ha autenticat. En aquesta informació que guardem també s'inclou la MAC del dispositiu del professor.

3. El dispositiu de l'alumne genera un nou QR on hi guarda un missatge amb la passphrase rebuda i les dades de l'alumne.

4. El dispositiu del professor escaneja aquest últim QR generat per l'alumne i l'autentica. Al decodificar el QR comprova que la passphrase efectivament és correcta i per tant l'autenticació és un éxit. En cas contrari l'autenticació es considera que ha fallat. Les dades de l'alumne rebudes al QR s'utilitzen per actualitzar les del llistat que té el dispositiu del professor.

La figura 2 mostra el procés d'escaneig d'un codi QR entre dos dispositius. Per aquest mecanisme d'autenticació no s'ha utilitzat xifrat, ja que no fa falta. Inicialment es va realitzar una prova de generació de QR amb el contingut del missatge xifrat amb una clau RSA, però l'únic que vam aconseguir va ser empitjorar el rendiment de l'escaneig del QR (trigava molt més) sense realment afegir cap millora en l'autenticació. Com que el QR el generarem al principi del procés d'autenticació en el dispositiu del professor, i cada vegada que tornem a generar un QR canviem de passphrase, no ens cal preocupar-nos d'escaneigs d'aquest QR no desitjats. Per tant podem evitar aquest problema.
L'enviament de missatges es fa mitjançant JSON, que és un tipus de format que ens permet enviar tipus de dades diferents encapsulats en un mateix missatge.

Fig. 2: Imatge que mostra el procés d'escaneig d'un codi QR

6 MÒDUL D'AUTENTICACIÓ

Necessitàvem una aplicació que ens proporcionés la funcionalitat necessària per a tota la gestió de l'autenticació. Aquesta funcionalitat inclou, a més dels mecanismes d'autenticació anteriorment explicats, totes les dades d'aplicació necessàries. Hem desenvolupat el mòdul d'autenticació en llenguatge de programació Java.

Respecte al desenvolupament de la mòdul d'autenticació, volíem separar aquesta part del que és el desenvolupament d'interfície mòbil. Així doncs vam definir un model de domini que inclou les especificacions de la nostra aplicació. L'aplicació consta de les següents parts:

1. Classes de l'aplicació:

   1.1. Mòbil: Aquesta és una classe bàsica que ens serveix per guardar dades sobre un dispositiu. Com a atributs conté la MAC del dispositiu, un codi hash basat en la MAC, i el nom del dispositiu (aquest últim és opcional).

   1.2. Estudiant: Estudiant és una classe de dades que representa a l'alumne. Conté informació bàsica com ara el nom i cognoms, l'identificador (NIU o similar), el correu electrònic, la institució i el departament al qual pertany. També conté una instància de la classe Mòbil.

   1.3. Professor: El Professor és una classe de dades similar a l'anterior. Conté la informació del professor i també la del seu dispositiu. Per tant també conté una instància de la classe Mòbil. A diferència de la classe Estudiant, el professor serà una classe amb instància única en l'aplicació.

   1.4. Aula: Aquesta classe és bàsicament una llista d'Estudiants.

   1.5. Autenticador: La classe Autenticador és una classe abstracta que defineix els mètodes d'autenticació. Les classes filles hereten i implementen aquests mètodes, concretant-los. Actualment disposem de dues classes filles derivades d'aquesta. Una és la que pertany a l'autenticació via QR i l'altra a l'autenticació via contrasenya (password), utilitzada per P2P.

   1.6. Client: aquesta classe conté funcions que realitza el dispositiu client. Bàsicament es tracta de

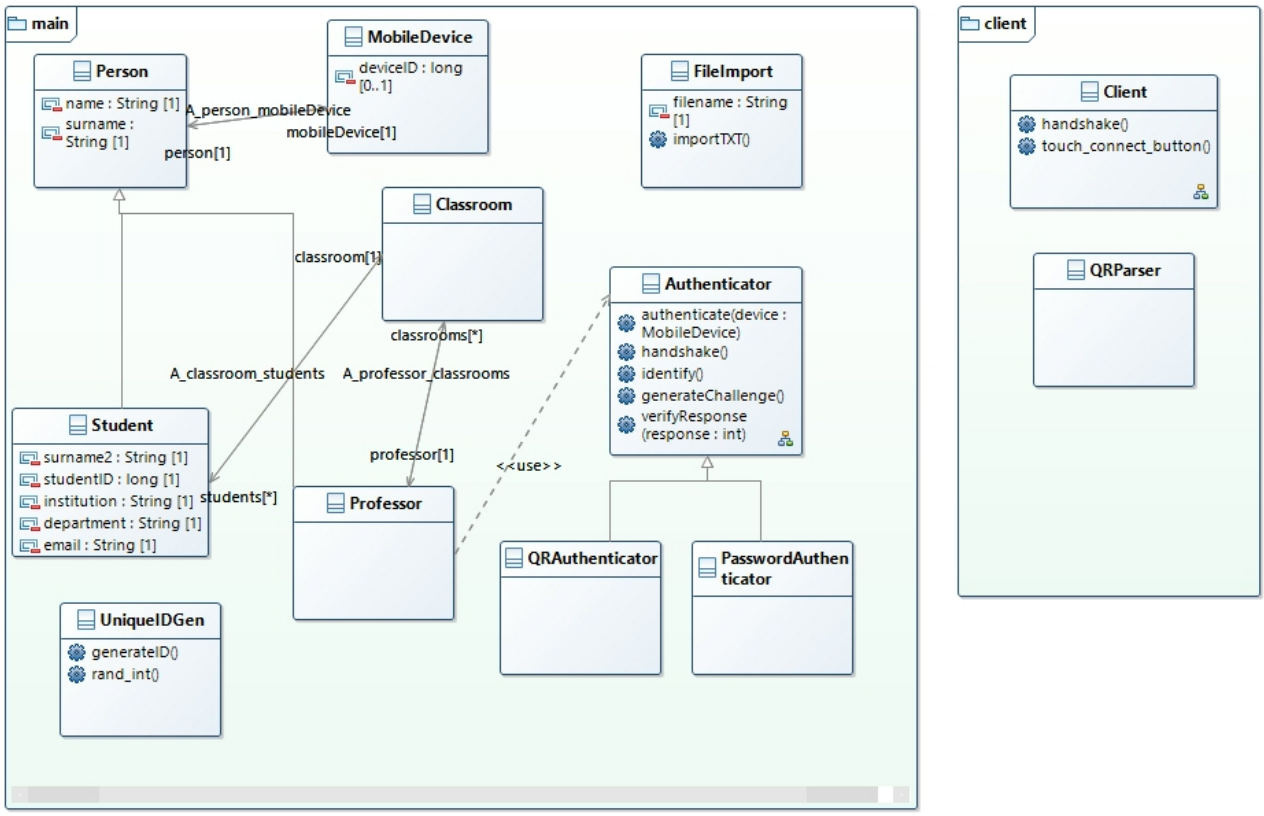

Fig. 3: Diagrama de classes del mòdul d'autenticació

diverses funcions de generació de missatges en format JSON.

2. Relacions entre classes:

   (a) Aula-Estudiant. Com ja hem indicat prèviament, la classe Aula conté una llista d'Estudiants. La classe Aula doncs pot tenir fins a N Estudiants, però cada Estudiant només pertany a una única Aula. Aquests estudiants poden trobar-se en estats diferents.

   (b) Professor-Autenticador. El Professor és una classe que supervisa el procés d'autenticació, a més de participar-hi.

   (c) Estudiant-Professor. L'estudiant serà autenticat per un Professor.

   (d) Professor-Aula: Un professor pot tenir assignades diverses aules, però una aula només pertany a un professor.

   (e) Client-Autenticador: El Client serà el que estableixi connexió amb el servidor. La classe Autenticador (i derivades) pertany al servidor i s'encarrega de generar el challenge d'autenticació. La classe Client conté funcions de generació de missatges de resposta al servidor.

A més a més de totes les classes anteriorment descrites, hem necessitat classes per realitzar les funcions d'importació de fitxers de text, parsejar missatges JSON, xifrat i desxifrat de dades, i altres utilitats com ara la generació de nombres aleatoris o identificadors numèrics únics i sequencials.
La figura 3 mostra el diagrama de classes de la mòdul d'autenticació.

7 Aplicacions Android

Una vegada tenim plantejada i programada la funcionalitat del mòdul d'autenticació, necessitem programar la funcionalitat que defineix la interfície de l'aplicació. Aquesta part l'hem realitzat amb Java Android Studio. Android Studio és l'entorn de desenvolupament oficial que permet programar aplicacions per Android. Utilitza llenguatge de programació Java.

En el cas del nostre treball hem hagut de realitzar dues interfícies d'aplicació diferents, una pel dispositiu de l'alumne i un altre pel dispositiu del professor. Ambdues interfícies utilitzen el mòdul d'autenticació definida prèviament, encara que no en la seva totalitat. Per exemple, l'aplicació de l'alumne haurà d'utilitzar les funcions de la classe Client, però no necessita per res la classe Autenticador. Això és a causa del fet que no realitza el procés d'autenticació (simplement respon als missatges rebuts). A més de desenvolupar l'aspecte de les aplicacions (la interfície gràfica) s'han d'integrar totes les funcionalitats nucli correctament. Hem de lligar la generació dels challenge d'autenticació amb la generació de codis QR i l'enviament de dades via P2P.

Pel que respecta a P2P hem hagut de desenvolupar una infraestructura bastant complexa. L'avantatge és que Android ja suporta Wi-Fi P2P i no hem necessitat llibreries externes per desenvolupar aquesta part. Hem hagut d'integrar les funcionalitats de cerca de dispositius, connexió i desconnexió.

La connexió de Wi-Fi P2P originalment és unidireccional. Entre els dos dispositius que es connecten, s'escull automàticament un que fa de servidor i un altre que fa de

client. El servidor s'anomena Group Owner. El client és l'únic que pot enviar dades al servidor perquè coneix la seva IP i MAC. El servidor a priori no guarda cap informació del client que s'ha connectat i per tant no pot enviar-li cap missatge o dada, només escoltar peticions.

La forma de solucionar aquest problema és indicar un paràmetre que marca la prioritat d'un dispositiu per ser escollit com a Group Owner. Al dispositiu client que inicia la connexió li indicarem un zero, ja que no volem que sigui Group Owner. Al dispositiu servidor, que és el que està escoltant peticions i rep aquesta petició de connexió, li indicarem el valor més alt de prioritat. Això ens assegura que sempre serà el Group Owner. Una vegada definit aquest valor, en acceptar la connexió, guardarem les dades del dispositiu que se'ns ha connectat. Això ho podem fer amb certa complicació si accedim a les dades del socket que està escoltant les peticions al servidor. Aquest socket quan rep una connexió guarda la IP i la MAC de qui s'ha connectat. Recuperant i guardant aquestes dades ja podrem respondre des del servidor al client, aconseguint una comunicació bidireccional.

Un altre inconvenient que vam haver de solucionar va ser el dels noms P2P. El dispositiu mòbil té assignat un nom. Per defecte ve amb el nom de la marca del telèfon, és a dir, quelcom similar a "Xperia-XA321". Aquest nom es pot canviar però s'ha d'anar a preferències del sistema, i és una mica complicat de trobar l'opció per un usuari normal. Així doncs vam implementar el mecanisme que permetia canviar el nom des de la nostra aplicació. Per poder realitzar aquesta tasca hem necessitat una llibreria anomenada Reflection Utils. Aquesta llibreria permet accedir a mètodes del sistema protegits. Com que el mètode que canvia el nom efectivament es tracta d'un mètode protegit, normalment només podríem accedir des de les preferències del sistema, i no des de la nostra aplicació. Amb aquesta llibreria podem accedir lliurament de la mateixa manera que si fos un mètode públic o no protegit.

8 APLICACIONS IOS

Si disposàvem de més temps, volíem portar el desenvolupament a iOS. Una vegada vam acabar d'integrar les funcionalitats nucli i de desenvolupar els mecanismes d'autenticació, comprovant que funcionaven correctament, vam realitzar un estudi de portabilitat. Aquest estudi tenia com a objectiu estudiar si ens era viable desenvolupar en el temps que ens quedava l'aplicació per iOS.

Vam realitzar aquest estudi durant la tercera setmana de maig del 2017, amb el que encara teníem 3 setmanes de temps per realitzar les tasques restants d'implementació de la interfície d'Android i a més, mirar de portar l'aplicació per iOS.

Després de realitzar aquest estudi vam deduir que malauradament no ens seria possible portar l'aplicació a iOS. Els motius són els següents:

- No tenim experiència prèvia amb el llenguatge de programació que utilitza el sistema operatiu iOS. Hauríem d'aprendre des de zero un llenguatge nou.
- iOS no disposa de llibreries natives que sí que té Android, com ara Wi-Fi P2P o JSON. Hauríem de cercar llibreries externes o buscar mecanismes alternatius.
- La llibreria ZXing per generació i escaneig de QR té una aplicació client en iOS, però no el codi font disponible.
- No disposàvem de temps suficient. Amb Android sí que teníem experiència prèvia i a més disposàvem de llibreries natives, i tot i així vam trigar un mes i mig en realitzar la implementació de la Interfície.

En conclusió, no era gens realista esperar aconseguir en tres setmanes de desenvolupament el que havíem desenvolupat en sis. A més lidiàvem amb molts altres factors de risc com ara que no teníem experiència d'haver treballat amb iOS prèviament i el suport de moltes llibreries que hem utilitzat no existeix en iOS. Finalment vam decidir que la portabilitat a iOS no era viable. Per realitzar aquesta portabilitat necessitaríem més temps de desenvolupament.

9 RESULTATS

Vam realitzar diversos experiments. Per la majoria de temps de desenvolupament vam disposar de dos dispositius mòbils amb els que realitzar moltes proves de funcionalitat. Per l'app del professor es tracta d'un mòbil Sony Xperia XA amb versió d'Android Marshmallow v.6.0 i per l'app de l'alumne vam utilitzar un Alcatel Pixi 4, també amb la mateixa versió d'Android.

Després vam realitzar un experiment amb 15 persones (1 professor + 14 alumnes).

Dos dels dispositius eren IPhone (el 13,33% del total) per tant no vam poder provar el funcionament de l'aplicació en aquest dos dispositius. Tots els altres dispositius (el 86,67% restant) eren versions d'Android Kitkat v.4.3 o superior, amb el que no vam tenir cap problema de compatibilitat per la instal·lació i funcionament. Per l'app del professor vam utilitzar el mateix dispositiu que per les proves un a un, un Sony Xperia XA.

Tots els dispositius amb les corresponents versions d'Android es poden veure a la taula 2.

El temps de preparació, que inclou la descàrrega i instal·lació de les apps dels alumnes (que es va realitzar "in situ"), va ser de 40 minuts. Després vam realitzar amb el dispositiu del professor una autenticació de tots els dispositius dels alumnes mitjançant autenticació QR. En realitzar aquest procés vam trigar molt poc (8 minuts), comparat amb el temps de preparació. Vam autenticar tots els dispositius correctament i sense incidències. Un cop autenticats els dispositius via QR, vam tornar a mirar d'autenticar els dispositius, aquesta vegada utilitzant el mecanisme d'autenticació P2P. Aquí sí que vam trigar més que amb l'altre mecanisme, pel fet que vam patir petits inconvenients. No obstant això, vam aconseguir autenticar als 12 dispositius Android.

Hem extret dades positives d'aquests experiments. L'aplicació de l'alumne és senzilla d'utilitzar, i pràcticament no va fer falta explicar als usuaris com funcionava i de seguida van aprendre a fer-la funcionar. L'autenticació via QR és molt ràpida, ja que l'escaneig dels codis es fa en pocs segons. L'autenticació P2P, encara que és cert que vam trobar dificultats amb les connexions i desconnexions peer-to-peer (P2P), realitza el mecanisme d'autenticació sense errors. Així doncs extraiem conclusions positives del feedback rebut i creiem que les aplicacions desenvolupades proporcionen autenticació sense Internet de forma satisfactòria.

Marca mòbil	Sistema Operatiu
Samsung Galaxy S4	Android Lollipop (v.5.1.1)
LG Nexus 5	Android Lollipop (v.5.1.1)
Sony Xperia E	Android KitKat (v.4.4)
Bq Aquaris E5	Android Lollipop (v.5.1)
Sony Xperia L	Android KitKat (v.4.4)
HTC One X9	Android Lollipop (v.5.1)
Sony Xperia Z	Android Marshmallow (v.6.0)
Samsung Galaxy Note 5	Android Marshmallow (v.6.0)
Samsung Galaxy S4	Android Lollipop (v.5.1.1)
Huawei P8	Android Marshmallow (v.6.0)
Sony Xperia Z1	Android Marshmallow (v.6.0)
Samsung Galaxy S5	Android Lollipop (v.5.1.1)
Sony Xperia XA	Android Marshmallow (v.6.0)
IPhone 5	iOS v.9.3.5
IPhone 6s	iOS v.10.3.1

TAULA 2: VERSIONS MÒBIL DEL EXPERIMENT AMB 15 PERSONES

10 CONCLUSIONS

Pel que respecta als objectius inicials que ens havíem marcat, hem assolit tots els que vam definir com obligatoris. Hem aconseguit implementar una proposta que soluciona el problema que ens havien plantejat inicialment. Hem dissenyat un esquema d'autenticació basat en U2F que pot servir de base per ser utilitzat amb diverses tecnologies disponibles en els dispositius mòbils. Hem dissenyat un esquema d'autenticació que hem utilitzat per desenvolupar dos mecanismes amb tecnologies diferents, un amb autenticació QR i l'altre amb Wi-Fi P2P. Complim el requisit d'utilitzar tecnologies sense Internet i a més a més donem versatilitat a l'usuari final, ja que té dos mecanismes que pot utilitzar. Identifiquem als alumnes de forma individual pel seu dispositiu mòbil. Hem implementat una mòdul que simula l'autenticació de llistats d'alumnes. Aquesta funcionalitat ha servit de base per desenvolupar un parell d'aplicacions que utilitzen les tecnologies esmentades per realitzar autenticació de dispositius mòbils propers. Aquestes aplicacions s'han desenvolupat pel sistema operatiu Android.

Durant el desenvolupament d'aquest projecte hem après que tot i tenir una planificació inicial definida, sempre tenim risc de trobar problemes inesperats. De fet ens hem trobat amb desafiaments i hem hagut de resoldre inconvenients que no ens era possible planificar i identificar inicialment, com ara les limitacions en el pas de missatges via Wifi P2P o canvis en el funcionament d'accés a fitxers en Android. A més, tot i que no hem sigut ambiciosos inicialment, hem hagut d'oblidar-nos de desenvolupar el mecanisme d'autenticació NFC (que vam estudiar i indicar com a opcional) per manca de temps.

Primerament teníem planificat realitzar la implementació en Android. Vam escollir Android perquè és el sistema operatiu amb major percentatge d'utilització al mercat. A més disposàvem de coneixements i experiència prèvia [10]. En cas de disposar de més temps pel desenvolupament del projecte, aquests són els objectius que voldríem desenvolupar com a línies futures:

1. Afegir mecanisme d'autenticació mitjançant tecnologia NFC.

2. Desenvolupar l'aplicació pel sistema operatiu iOS.

AGRAÏMENTS

M'agradaria agrair a la tutora del TFG Àngela Fabregues per l'ajuda i feedback proporcionats durant el desenvolupament del treball. Els valuosos suggeriments i consells proporcionats em van ajudar a coordinar el meu projecte, especialment en la redacció d'aquest informe.

A més, voldria agrair a les assignatures de "Xarxes" i "Tecnologies Avançades d'Internet" pels coneixements que m'han proporcionat sobre sockets, xarxes adhoc i xarxes P2P. També a les assignatures "Informació i Seguretat" i "Garantia de la Informació i Seguretat", ja que gràcies als coneixements sobre esquemes de xifrat i autenticació vaig tenir una idea de com havia de començar a treballar. Sense aquests coneixements de base el desenvolupament d'aquest projecte no hauria estat possible o hauria resultat molt difícil de realitzar.

Gràcies també a la valuosa experiencia obtinguda desenvolupant una aplicació d'Android a l'assignatura "Disseny del Software" vaig ser capaç de tenir la confiança necessària per elaborar una planificació amb la seguretat que podria complir el deadline.

Finalment també m'agradaria agrair a l'assignatura de "Gestió de Projectes" per l'experiència obtinguda elaborant documents tècnics.

REFERÈNCIES

[1] FIDO Alliance, "U2F - FIDO Universal 2nd Factor Authentication v1.1," *FidoAlliance Organization*, September 2015.

[2] Wi-Fi Direct Alliance, "Wi-fi peer-to-peer technical specification v1.5," *Specifications, Wi-Fi Direct Alliance Propietary*, August 2014.

[3] Network Working Group, " Peer-to-Peer (P2P) Architecture: Definition, Taxonomies, Examples, and Applicability," *RFC5694*, November 2009.

[4] Android Developers, "Android Developer Guides," *developer.android.com*, 2017.

[5] M. Q. Saeed and C. D. Walter, "Off-line NFC Tag Authentication," *IEE Xplore Document*, 2017.

[6] Web NFC Community Group, " W3C Web NFC API Draft," *W3c.github.io*, 2016.

[7] S. Owen and A. Coeur, "GitHub - zxing/zxing: Official ZXing ("Zebra Crossing") project home," *Zebra Crossing Project*, 2017.

[8] Consejo Superior de Investigaciones Científicas, "Institut d'Investigació en Intel·ligència Artificial," *IIIA-CSIC*.

[9] ISO/IEC 18004:2015, " Information – Automatic identification and data capture techniques – QR Code barcode symbology specification," *ISO/IEC Standards*, February 2015.

[10] D. Griffiths and D. Griffiths, *Head First Android Development*. O'Reilly Media, 2015, ch. 2-3,5,8.