

Université Abou Bekr Belkaid
Tlemcen Algérie



جامعة أبي بكر بلقايد

République Algérienne Démocratique et Populaire
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique
UNIVERSITE ABOU BEKR BELKAID TLEMEN
FACULTE DE TECHNOLOGIE

DEPARTEMENT DE GENIE ELECTRIQUE ET ELECTRONIQUE



MEMOIRE

Pour l'obtention du diplôme de
MASTER en

Réseaux mobiles et services de télécommunications

Réalisé par

Mr. HADOU MOULAY ABDELAZIZ

THEME

Contrôle à distance d'un mobile

Soutenu en 26 MAI 2015 devant le Jury :

Mr. ZERROUKI Hadj

M.C.B à l'Université de Tlemcen

Président

Mr. HADJILA Mourad

M.C.B à l'Université de Tlemcen

Examineur

Mr. MERZOUGUI Rachid

M.C.A à l'Université de Tlemcen

Encadreur

Année universitaire : 2014-2015

Remerciements

Au terme de ce travail, je tiens à remercier

Notre DIEU

tout puissant de m'avoir donné la foi, la force et le courage

Monsieur ZERROUKI HADJ

Pour avoir accepté de présider le jury de ma soutenance.

Mon encadreur Monsieur MERZOUGUI RACHID

Pour ses qualités humaines et professionnelles, pour son encadrement, ses directives,

Ses remarques constructives, et sa disponibilité.

Monsieur HADJILLA MOURAD mon examinateur

De me faire l'honneur d'être membre examinateur du jury

Monsieur FEHAM MOUHAMED

Pour avoir accepté d'être étudiant au spécialité RMST

Tous les personnels qui me soutien

Pour leurs encouragements continus et leurs aides précieuses.

HADOU MOULAY ABDELAAZIZ

Je dédie ce travail à

*Mes parents,
Ma fiancée,
Mon frère et Ma sœur,
Et tous mes amis et ma famille.*

HADOU MOULAY ABDELAAZIZ

Table des matières

Remerciement
Dédicaces
Table des matières
Table des figures
Liste des tableaux
Glossaire

Introduction générale..... 1

Chapitre I : Réseaux d'accès radio mobiles

I.1. Introduction.....	2
I.2. Standard des réseaux d'accès mobile	2
I.2.1 1G.....	2
I.2.2 Norme GSM (2G)	3
I.2.3 GPRS (2.5G).....	4
I.2.4 HSCSD ou EDGE	5
I.2.5 UMTS (3G).....	5
I.2.6 Technologie HSDPA (3.5G)	6
I.2.7 Technologie HSUPA (3.75G).....	6
I.2.8 Technologie 4G.....	6
I.3. Passerelle WAP	7
I.4. Modèle TCP/IP.....	9
I.5. Conclusion	11

Chapitre II : Smartphone et le système d'exploitation mobile Android

II.1. Introduction	12
II.2. Définition Smartphone.....	12
II.3. Historique.....	12

II.4. Les principaux acteurs du marché des Smartphones	13
II.5. Systèmes d'exploitation pour les Smartphones	14
II.5.1. iOS de Apple Inc	14
II.5.2. Windows Phone de Microsoft.....	14
II.5.3. BlackBerry OS de Research In Motion (RIM)	15
II.5.4. Bada de Samsung Electronics.....	15
II.5.5. Firefox OS de Mozilla	15
II.5.6. Android de Google Inc	15
II.5.6.1. Etude de la plateforme Android.....	16
II.5.6.2. Différentes versions	16
II.5.6.3. Architecture Android	18
II.5.6.3.a. Premier niveau : noyau linux	19
II.5.6.2.b. Deuxième niveau : Bibliothèques C / C++	20
II.5.6.2.c. Troisième niveau : Framework de développement	21
II.5.6.2.d. Quatrième niveau : applications	22
II.6. Conclusion	22

Chapitre III : Réalisation et mise en œuvre

III.1. Introduction	23
III.2. cahier de charges	23
III.3. Description le système de contrôle a distance	24
III.4. Technologie de modèle de contrôle	25
III.5. Partie serveur	25
III.5 .1 Site web	25
III.5.1.1. Pourquoi un site web?	25
III.5.1.2. Présentation de site web de projet	26
III.5.1.3. Architecture de site web	26
III.5.1.4. Partie hors session (avant Authentification).....	27
III.5.1.5. Partie après la session.....	29
III.5.1.6. Page Accueil.....	29
III.5.1.7. Page mon serveur	29

Tables des matières

III.5.1.8. Page mon client	31
III.5.1.9. Page contrôle	31
III.5.1.9.a. Page Appareil	32
III.5.1.9.b. Page SMS	33
III.5.1.9.c. Pa page appels	33
III.5.1.9.d. Page camera.....	34
III.5.1.9.e. Page audio.....	34
III.5.1.9.f. Page de localisation.....	34
III.5.1.9.g. Page SHELL.....	35
III.5.2 Fonctionnement du serveur de contrôle	35
III.5.2.1. génération de serveur d'écoute et son client.....	35
III.5.2.2. Exécution des commandes	36
III.5.2.3. ScriptGET.PL.....	36
III.6 Partie Client	37
III.7 Mode de connexion	37
III.8 Autres fichiers nécessaires pour le projet	38
III.9 Conclusion	38
Conclusion general	39
Référence & bibliographie.....	40

Table des figures

Figure I.1 Architecture réseau GSM 4

Figure I.2 Pile WAP et les composants Internet 8

Figure I.3 Systèmes en couches 9

Figure II.1 Logo D'Appel IOS..... 14

Figure II.2 Evolution des versions d'Android 16

Figure II.3 Architecture de système Android 18

Figure II.4 architecture de ces médiathèques..... 21

Figure III.1 Architecture du système de contrôle 25

Figure III.2 Architecteur de site web 26

Figure III.3 Page d'accueil de site web 27

Figure III.4 Page d'inscription sur le site 27

Figure III.5 Addition d'une ligne dans le tableau d'utilisateurs 28

Figure III.6 Page d'accès à l'espace d'utilisateurs..... 28

Figure III.7 Entête d'utilisateur fin3 29

Figure III.8 Changements dans le fichier sudoers 30

Figure III.9 Page mon serveur 30

Figure III.10 Page « mon client » 31

Figure III.11 Entête de contrôle 31

Figure III.12 Identification d'appareil contrôlé 32

Figure III.13 Informations sur les interfaces dans le téléphone 32

Figure III.14 Page de téléchargement des messages..... 33

Figure III.15 Exemple d'affichage les messages récupérés 33

Figure III.16 Page de téléchargement du journal d'appels..... 34

Figure III.17 Page de localisation par Google map 34

Figure III.18 Scénario de lancement l'écouteur et génération de client 35

Figure III.19 Scénario d'applique d'une commande 36

Figure III.20 Mode de connexion entre les clients et le serveur 37

Liste des tableaux

Tab I.1 Différentes normes de téléphonie mobile et les vitesses qu'elles offrent 7
Tab II.1 Nombres de Smartphones systèmes expédition à travers le monde..... 13
Tab II.2 Part de marché mondiale des systèmes d'exploitation mobiles 13

Glossaire



Glossaire

A

AAC: Advanced Audio Coding.

AMPS: Advanced Mobile Phone System.

AMPS: Abbingdon Music Research.

API: Application Programming Interface.

ARM: Advanced Risc Machine.

B

BASH: Bourne-Again Shell.

BSD: Berkeley Software Distribution.

C

CES : Consumer Electronics Show.

CPU: Central Processing Unit.

E

EDGE: Enhanced Data for GSM Evolution.

ETACS: Extended Total Access Communication System.

F

FTP: File Transfer Protocol.

G

GPRS: Global Packet Radio Service.

GSM: Global System for Mobile communication.

H

HSCSD: High-Speed Circuit-Switched Data.

HSDPA: High Speed Downlink Packet Access.

HSUPA: High Speed Uplink Packet Access.

HTC: High Tech Computer.

HTTP: Hyper Text Transfer Protocol.

I

ID: Identity.

IMT: International Mobile *Télécommunications*.

iOS: *Iphone Operating System*.

IP: Internet Protocol.

IT: Intervalle de Temps

J

JPG: Joint Photographic Group.

L

LTE: Long Term Evolution.

M

MIPS: Moving Pictures Experts Group.

MP3: Motion Picture Expert Group– Audio Layer 3.

MPEG4: Moving Picture Experts Group 4.

O

OHA: Open Handset Alliance.

OS: *Operating System*.

P

PC: Personal Computer.

PCS: Personal Communications Service

PDA: Personal Digital Assistant.

PDC: *Professional Darts Corporation*.

PERL: Practical Extraction and Report Language.

PHP: *Hypertext Preprocessor*.

PNG: Portable Network Graphics.

PSK: Phase-Shift Keying.

R

RIM: Research In Motion.

RTC: Réseau Téléphonique Commuté

S

SMS: Short Message System.

SQL: Structured Query Language.

T

TACS: Total Access Communication System.

TDMA: Time Division Multiple Access.

TCP: Transmission Control Protocol.

U

UIT: Universal Institute of Technology.

UMTS: Universal Mobile Télécommunications System.

UDP: User Datagram Protocol

W

WAE: Wireless Application Agent.

WSP: Wireless Session Protocol.

WDP: Wireless Datagram Protocol.

WTLS: Wireless Transport Layer Security.

WTP: Wireless Transaction Protocol.



Introduction générale

Introduction générale

Les progrès conjoints de la microélectronique, des technologies de transmission sans fil et des applications embarquées ont permis de produire à coût raisonnable des terminaux mobiles de haute technologie comme les Smartphones et les tablettes PC.

Actuellement la société Apple à travers son Smartphone « iPhone», sa tablette PC « iPad » et son système d'exploitation « iPhone OS» est en forte concurrence avec la communauté Open Handset Alliance (OHA) qui englobe Google, Motorola, HTC, Samsung, etc. Cette dernière équipe ses terminaux mobiles par le système d'exploitation mobile « Android OS».

Ce projet présente la conception d'un système de télé-contrôle, permettant de commander à distance un téléphone portable doté de l'Android en utilisant un site web abrégé sur un serveur.

Pour répondre aux besoins et objectifs de ce projet de fin d'étude, nous avons organisé notre travail en trois chapitres :

- Le premier chapitre expose le développement des générations des réseaux cellulaires.
- Le chapitre II décrit les Smartphones, leurs systèmes d'exploitation et plus particulièrement le système Android.
- Le chapitre III est une description détaillée de la conception et l'implémentation de système de contrôle à distance.



Chapitre I

Réseaux d'accès radio mobiles

I.1. Introduction

Né dans les années 1980, le téléphone mobile a connu une diffusion très rapide. Aujourd'hui, pour une population mondiale de 7 milliards de personnes, on compte plus de 6.8 milliards d'abonnés au téléphone mobile. Soit environ 96,2% des habitants (UIT-2014).

C'est véritablement un nouveau secteur de l'industrie mondiale qui s'est créé, regroupant notamment constructeurs de circuits électroniques, de terminaux mobiles, d'infrastructures de réseaux, développeurs d'applications et de services et opérateurs de réseaux mobiles.

Avant d'expliquer l'état actuel des technologies utilisées, aujourd'hui dans le téléphone portable ou le Smartphone, il me semble intéressant de rappeler l'évolution de ces techniques, cela a pour avantage d'une part de savoir de quoi on est parti pour mieux se positionner à l'heure actuelle et d'autre part de voir l'évolution des réseaux mobiles.

I.2. Standard des réseaux d'accès mobile

Les progrès technologique dans le domaine des réseaux de télécommunications mobiles, ont vu l'apparition des technologies numériques au début des années 1990. En Europe (*GSM*), au Japon (*PDC*) et aux Etats Unis (*PCS*) [1]. L'évolution du réseau radio mobile analogique (*1G*) vers *GSM* (dit de 2ème génération «*2G*») après *UMTS* (dit de 3ème génération «*3G*») ensuite vers la «*4G*» (4ème génération) passe par des générations intermédiaires comme le *GPRS*, *HSCSD* ou *EDGE* (dites «*2.5G*»), *HSDPA* (*3.5G*) et *HSUPA* (*3.75G*) qui seront présentés dans les paragraphes suivants.

I.2.1. Norme 1G

La première génération de téléphonie mobile (notée 1G) possédait un fonctionnement analogique et était constituée d'appareils relativement volumineux. Il s'agissait principalement des standards suivants :

- *AMPS* apparu en 1976 aux Etats-Unis, constitue le premier standard de réseau cellulaire. Utilisé principalement Outre-Atlantique, en Russie et en Asie, ce réseau analogique de première génération possédait de faibles mécanismes de sécurité rendant possible le piratage de lignes téléphoniques.
- *TACS* est la version européenne du modèle AMPS. Utilisant la bande de fréquence de 900 MHz, ce système fut notamment largement utilisé en Angleterre, puis en Asie (Hong-Kong et Japon).
- *ETACS* est une version améliorée du standard *TACS* développé au Royaume-Uni utilisant un nombre plus important de canaux de communication [2].

Les réseaux cellulaires de première génération ont été rendus obsolètes avec l'apparition d'une seconde génération entièrement numérique.

I.2.2. Norme GSM (2G)

Le service le plus important dans les réseaux cellulaires *GSM* est le service de la voix. Cette technologie a pour premier rôle de permettre des communications entre abonnés mobiles et abonnés du réseau fixe (*RTC*). Le réseau *GSM* s'interface avec le réseau *RTC* et comprend des commutateurs. Il se distingue par un accès spécifique: la liaison radio.

La satisfaction de l'utilisateur final se traduit par trois contraintes de fonctionnement:

- L'abonné doit pouvoir joindre n'importe qui, n'importe quand et n'importe où.
- Après établissement de la communication, la conversation est audible et compréhensible par les deux interlocuteurs.
- La ligne téléphonique n'est pas coupée en cours de communication.

Pour l'opérateur, ceci se traduit par trois contraintes techniques:

- Une couverture nationale
- Un dimensionnement correct des liens radio et réseau (pour la disponibilité des ressources)
- La mise en place de mécanismes efficaces de gestion du lien radio (pour la minimisation du taux de coupure) [1].

La Figure I.1 montre L'architecture du GSM :

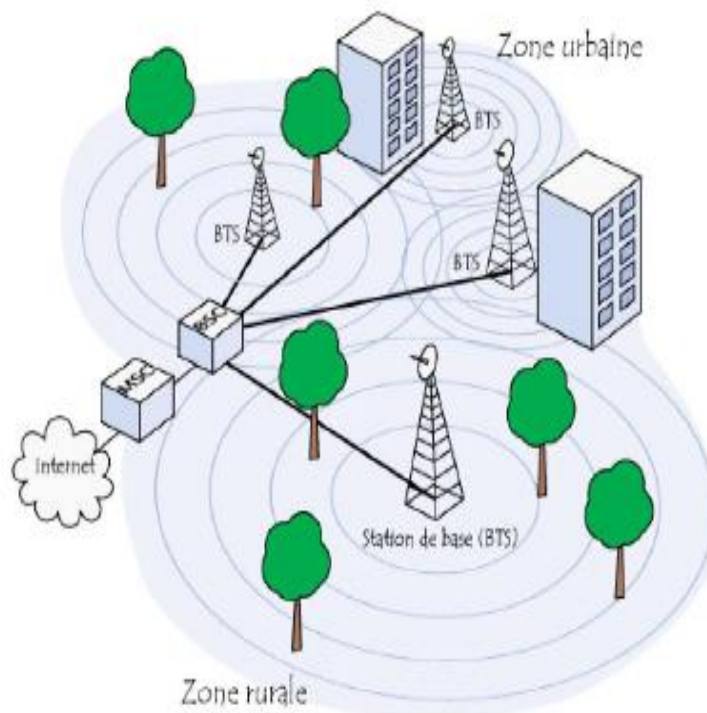


Figure I.1 Architecture réseau GSM

I.2.3.GPRS (2.5G)

Le standard *GPRS* représente une évolution majeure de la norme *GSM* et une transition vers la troisième génération, on parle généralement de *2.5G* pour classer ce standard. L'exploitation du mode de transfert des données par paquets et l'augmentation des débits ouvrent la porte aux communications mobiles multimédia. Ce standard peut être considéré comme un réseau de données à part entière (commutation de paquet) qui dispose d'un accès radio réutilisant une partie du réseau *GSM*[3]. Le réseau *GPRS* est relié à différents réseaux de données par l'intermédiaire de l'Internet (Protocole *IP*). Pour cela, il

est indispensable qu'un terminal *GPRS* dispose d'une adresse *IP* dont le champ réseau est spécifique à ce type de support.

Les débits théoriques autorisés par cette génération (de *9.6 Kbps* à *171.2 Kbps*) permettent d'envisager de nombreuses applications tels que la consultation du Web, le transfert de fichiers, la transmission de vidéo compressée, etc. La facturation en *GPRS* se fait selon le volume échangé plutôt qu'à la durée de connexion, ce qui signifie notamment qu'il peut rester connecté sans surcoût.

I.2.4.HSCSD ou EDGE

EDGE représente une seconde forme d'évolution des systèmes *2G*. Il s'agit d'une simple évolution de la technologie *GSM/GPRS* et du système *TDMA* permettant d'obtenir un débit qui peut aller jusqu'à *384 Kbps*. Mais c'est beaucoup moins performant que la *3G* et son rendement optimal est obtenu lorsqu'il est combiné avec un réseau de commutation par paquet (*GPRS*).

Ce standard utilise une nouvelle modulation du signal (*8PSK*) qui permet d'améliorer l'efficacité spectrale et la capacité du réseau. Un terminal mobile dans un réseau *EDGE* est capable de transmettre et de recevoir sur plusieurs intervalles de temps (*IT*) [1]. Cette technique permet de définir des circuits allant jusqu'à *6 IT* alloués aux mobiles par trame *TDMA* ; ce qui permet d'envisager des débits de l'ordre *19.2 Kbps*, *28.8 Kbps*, *38.4 Kbps*, *48 Kbps*, *56 Kbps* ou *64 Kbps* suivant le nombre des canaux alloués

I.2.5.UMTS (3G)

La norme *UMTS* est une évolution de la deuxième génération à la troisième génération (*3G*). Elle constitue une voie royale pour le développement de produits et de services multimédias. Les technologies développées autour de cette norme conduisent à une amélioration significative des services et des vitesses de transfert avec des débits supérieurs à *144 Kbps* et pouvant aller jusqu'à *2 Mbps*. Cette amélioration des débits est rendue possible grâce à l'évolution des technologies radio qui autorisent une meilleure efficacité spectrale et l'exploitation de bandes de spectre de fréquences supérieures à celles utilisées par la technologie *GSM* [4].

I.2.6. Technologie *HSDPA* (3.5G)

La troisième génération (*UMTS*) impose des investissements énormes pour des débits qui ne sont pas éloignés de ceux du *GPRS* et se voit rapidement remplacée par la nouvelle génération (3.5G), appelée *HSDPA*. Cette nouvelle technologie offre des débits pouvant aller au-delà de 10 *Mbps* sur le lien descendant [5]. Le *HSDPA* peut être installé et déployé sans changer radicalement l'architecture de l'*UMTS*. Il permet du coup aux opérateurs de réaliser cette évolution à moindre coût.

I.2.7. Technologie *HSUPA* (3.75G)

La 3.75G est une variante de *HSDPA* sur la voie montante. *HSUPA* peut être considéré comme le successeur de *HSDPA*, il porte un débit montant (Up Link) à 5,8 *Mbps* théorique, le flux descendant (*Down Link*) étant de 14 *Mbps* comme en *HSDPA*[6]. L'intérêt de cette technologie sera d'offrir la possibilité d'émettre vers d'autres mobiles toutes sortes des documents électroniques (images, vidéos...) favorisant ainsi le développement de l'Internet haut débit mobile.

I.2.8. Technologie 4G

La 4eme génération des standards pour la téléphonie mobile succédant à la 2G et la 3G, elle permet le « très haut débit mobile », c'est-à-dire des transmissions de données à des débits théoriques supérieurs à 100 Mb/s, voire supérieurs à 1 Gb/s (débit minimum défini par l'*UIT* pour les spécifications *IMT-Advanced* (en)). En pratique, les débits sont de l'ordre de quelques dizaines de Mb/s selon le nombre d'utilisateurs, puisque la bande passante est partagée entre les terminaux actifs des utilisateurs présents dans une même cellule radio[7].

Le **tableau I.1** récapitule les différentes normes de téléphonie mobile citées auparavant et les vitesses qu'elles offrent (en *Mbps* : *Méga bit par seconde*) :

GÉNÉRATION	SURNOM	PROTOCOLE	DÉBIT MAXIMUM THÉORIQUE (MBPS)	DÉBIT MOYEN PRATIQUE (MBPS)
<i>1G</i>		Radiocompas 2000	analogique	analogique
<i>2G</i>	2.5G 2.75G	GSM GPRS EDGE	0,009 0,171 0,384	0,002 0,05 0,064
<i>3G</i>	3G+ 3G++ H+ Dual Carrier	UMTS HSPA HSPA+ DC-HSPA+	1,9 14,4 21 42	1,2 3.6 5 10
<i>4G</i>		LTE LTE-Advanced	150 1000	40 100

Tab I.1 Différentes normes de téléphonie mobile et les vitesses qu'elles offrent.

I.3. Passerelle WAP

Le protocole WAP, destiné aux applications sans fil, est un ensemble de standards et technologie permettant de faire fonctionner des applications sur un terminal mobile (les téléphones portables, PDA, les assistants numériques personnels...).

Le protocole WAP est défini selon des couches, c'est-à-dire des niveaux d'abstraction des données (dans l'esprit du modèle OSI) afin de séparer les différents traitements des données nécessaires pour effectuer la transaction.

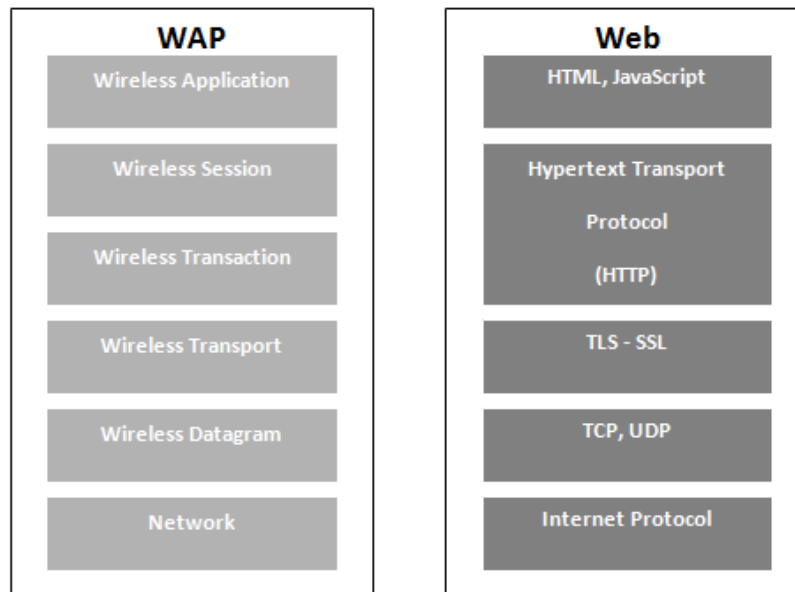


Figure I.2 Pile WAP et les composants Internet.

- **La couche application, WAE** Fournit un environnement d'application destiné au développement et à l'exécution des applications et des services dits portables.
- **La couche session, WSP** Fournit des méthodes pour l'organisation des échanges des informations entre les applications client / serveur. Elle permet d'établir une session entre un client et un serveur c'est-à-dire de définir les paramètres de connexion pour effectuer des transactions. La couche WSP permet ainsi à la couche application de bénéficier de deux types de sessions différentes:
 - Session orientée connexion dans laquelle la couche session va interagir avec la couche transaction.
 - Session orientée non-connexion dans laquelle la couche session va directement agir au niveau de la couche transport pour l'envoi de datagrammes bruts. WSP est dans son ensemble l'équivalent du protocole HTTP (dans sa version 1.1). On retrouve d'ailleurs un bon nombre d'implémentations identiques au HTTP dans WSP.
- **La couche transaction, WTP** Fournit des méthodes permettant de gérer les performances au niveau transactionnel.

- **La couche sécurité, WTLS** Est une couche optionnelle qui fournit, lorsqu'elle est implémentée, des méthodes permettant d'authentifier et de sécuriser des connexions entres applications.
- **La couche de transport, WDP** Est la couche la plus basse de la pile WAP, qui abrite les couches supérieures des services de transports (bearer) offerts par l'opérateur.

I.4. Modèle TCP/IP

TCP/IP est né de la réflexion de chercheurs américains. IP est un protocole qui permet d'envoyer des informations élémentaires de machine à machine. Les chercheurs ont développé un autre protocole de nom TCP. Le nom de TCP/IP a donc été choisi en référence à ces deux principaux protocoles qui le caractérisent. Aujourd'hui, ce modèle intègre beaucoup d'autres protocoles (FTP, SMTP, HTTP ...). TCP/IP est un protocole qui nécessite une coopération des OS (Systèmes d'exploitation) des machines.

TCP/IP est très répandu, car sa robustesse a été prouvée (quelques millions de machines interconnectées dans le monde). Tous les applicatifs réseaux doivent pouvoir communiquer entre eux, quelque soit l'architecture ou la plateforme utilisée. Pour cela, les opérations sur les réseaux ont été divisées en plusieurs phases de base, de manière à simplifier la portabilité des applicatifs sur toutes les plateformes [9], c'est ce qu'on appelle en couche. Un standard a alors été créé, normalisé par l'OSI sous la référence OSI-RM, utilisant 7 couches distinctes.

L'architecture TCP/IP (Figure I.3) est similaire à ce modèle en couche, mais ne dispose que de 4 couches dans la plupart des cas.

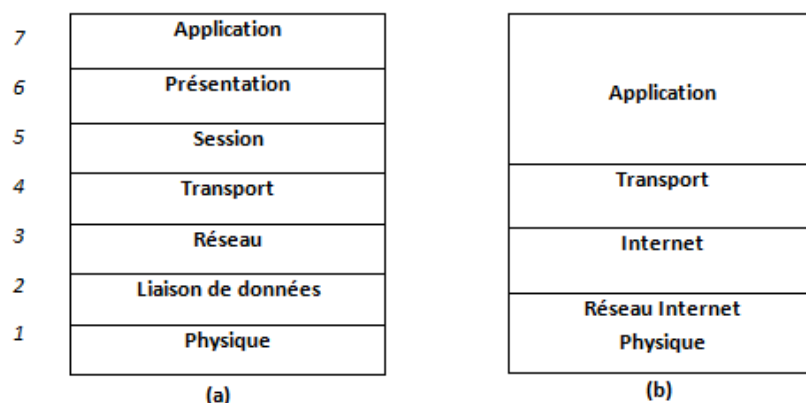


Figure I.3 Systèmes en couches.
(a): Modèle de référence OSI, (b): Modèle TCP/IP (Internet)

Les couches 5 à 7 du modèle OSI sont des couches dites d'application. Elles sont orientées application, et fournissent une interface entre une application et le réseau. Les couches 1 à 4 sont des couches dites de liaison. Ce sont elles qui se chargeront du routage, afin de correctement acheminer les paquets d'un point à un autre.

- **Protocole IP** Le protocole IP est le cœur du fonctionnement de l'internet. Il assure sans connexion un service non fiable de délivrance de datagrammes IP. Le service est non fiable car il n'existe aucune garantie pour que les datagrammes IP arrivent à destination. Certains peuvent être perdus, dupliqués, retardés, altérés ou remis dans le désordre. Le mode de transmission est non connecté car IP traite chaque datagramme indépendamment de ceux qui le précèdent et du suivant.
- **Le protocole UDP** Ce protocole utilise IP pour acheminer, d'un ordinateur à un autre, en mode non fiable des datagrammes qui lui sont transmis par une application. UDP n'utilise pas d'accusé de réception et ne peut pas donc garantir que les données ont bien été reçues. Il ne réordonne pas les messages si ceux-ci n'arrivent pas dans l'ordre dans lequel ils ont été émis et il n'assure pas non plus de contrôle de flux. Cependant, UDP fournit un service supplémentaire par rapport à IP car il permet de distinguer plusieurs applications destinataires sur la même machine par l'intermédiaire des ports.
- **Protocole TCP** Contrairement à UDP, TCP est un protocole qui procure un service de flux d'octets orienté connexion et fiable. Les données transmises par TCP sont encapsulées dans des datagrammes IP en y fixant la valeur du protocole à 6.
- **Le protocole HTTP** Http est un protocole de niveau application suffisamment léger et rapide pour une communication Client-serveur. Il peut fonctionner sur n'importe quelle connexion fiable. Les Clients http les plus connus sont les navigateurs Web permettant à l'utilisateur final d'accéder à un serveur de données. Il existe aussi des systèmes pour récupérer automatiquement les données sur Internet tel que les aspirateurs. Mis à part la récupération de données, les systèmes d'information évolués nécessitent d'autres fonctions telles que la possibilité d'effectuer des recherches, les fonctions de remise à jour et l'annotation. Les messages sont transmis sous une forme similaire à celle de la messagerie électronique (E-mail) et des extensions MIME.

I.5. Conclusion

Le développement des réseaux mobiles n'a pas cessé d'accroître, plusieurs générations ont vu le jour (*1G*, *2G*, *3G*, *4G* et prochainement la *5G* pas encore mise en œuvre). L'amélioration de ces générations mobile apporte sur le débit et la bande passante. Ces différentes générations exploitent des terminaux modernes tels que les Smartphones. Le chapitre suivant sera consacré à ces types de terminaux, leurs caractéristiques et leurs systèmes d'exploitation mobile.



Chapitre II

**Smartphones et le système
d'exploitation mobiles Android**

II.1. Introduction

Le Smartphone est un compagnon de tous les instants et un véritable ordinateur de poche pour gérer le quotidien. Ce dernier est passé en l'espace de quelques années d'un outil professionnel à un objet de divertissement grand public.

Dans ce chapitre, nous allons essayer de voir les concepts de base des Smartphones et nous nous intéressons plus précisément à la plateforme de Google Inc. Android

II.2. Définition de Smartphone

Un Smartphone est un terme d'origine anglo-saxonne, signifiant littéralement « téléphone intelligent ». Il désigne un téléphone mobile évolué disposant des fonctions d'un assistant numérique personnel, d'un appareil photo numérique et d'un ordinateur portable. La saisie des données se fait le plus souvent par le biais d'un écran tactile ou d'un clavier.

II.3. Historique

Paradoxalement à l'évolution des technologies, l'avènement du Smartphone a été relativement long à se mettre en place. Les principaux acteurs du marché (à l'époque, Nokia, Ericsson, Motorola) se faisant hésitant pour se lancer du fait des nombreuses contraintes de normalisation technique.

En effet, si à l'époque, chaque mobile GSM possédait son propre environnement d'exploitation, aucun d'entre eux n'était prêt à laisser rentrer Microsoft, à l'époque le seul acteur capable de lancer une plateforme dédiée mobile, permettant d'échanger des données et entièrement compatible avec les ordinateurs sous Windows [8].

II.4. Les principaux acteurs du marché des Smartphones

Aujourd'hui, les acteurs principaux du marché des Smartphones, sont Apple, constructeur de l'iPhone et propriétaire de son OS (iOS), Google propriétaire de son OS Android (Open source), Microsoft, propriétaire de son environnement Windows Mobile et les nombreux constructeurs de Smartphones, anciens ou récents, ayant pratiquement tous abandonné leur environnement propriétaire, au profit de Windows Mobile ou d'Android.

Les deux tableaux suivants donnent la part du marché des principaux systèmes d'exploitation et le nombre d'expédition mondiale de Smartphones :

SMARTPHONES OS EXPÉDITION (MILLION)	<i>T4 2013</i>	<i>2013</i>	<i>T4 2014</i>	<i>2014</i>
Android	<i>227.3</i>	<i>780.8</i>	<i>291.7</i>	<i>1042.7</i>
APPLE IOS	<i>51.0</i>	<i>153.4</i>	<i>74.5</i>	<i>192.7</i>
MICROSOFT WIN MOBILE	<i>9.6</i>	<i>35.8</i>	<i>11.3</i>	<i>38.8</i>
AUTRES SYSTEMES	<i>2.3</i>	<i>20.0</i>	<i>26</i>	<i>9.3</i>
TOTAL	<i>290.2</i>	<i>990.0</i>	<i>380.1</i>	<i>1283.5</i>

Tab II.1 Nombre des Smartphones systèmes expédition à travers le monde

SMARTPHONES OS parts de marché	<i>T4 2013</i>	<i>2013</i>	<i>T4 2014</i>	<i>2014</i>
Android	<i>78.3 %</i>	<i>78.9 %</i>	<i>76.7 %</i>	<i>81.2 %</i>
APPLE IOS	<i>17.6 %</i>	<i>15.5 %</i>	<i>19.6 %</i>	<i>15.0 %</i>
MICROSOFT WIN MOBILE	<i>3.3 %</i>	<i>3.6 %</i>	<i>3.0 %</i>	<i>3.0 %</i>
AUTRES SYSTEMES	<i>0.8 %</i>	<i>2.0 %</i>	<i>0.7 %</i>	<i>0.7 %</i>
TOTAL	<i>100 %</i>	<i>100 %</i>	<i>100 %</i>	<i>100 %</i>

Tab II.2 Part de marché mondiale des systèmes d'exploitation mobiles

Android est l'OS mobile le plus utilisé du monde. Cela fait déjà plusieurs années que cette situation persiste. En 2014, les choses n'ont fait que s'améliorer pour le système

de Google. En effet plus un milliard de terminaux tournant sous Android ont été vendus [8].

Pour cela, nous avons choisi le système Android pour utiliser comme le système de mobile qu'on veut contrôler.

II.5. Systèmes d'exploitation pour les Smartphones

II.5.1. IOS d'Apple Inc.

Le téléphone iPhone, l'iPod Touch et la tablette iPad utilisent tous le système d'exploitation iOS, qui est dérivé de Mac OS X. Les applications tierces sont supportées depuis juillet 2008. La boutique Apple propose 300 000 applications.



Figure II.1 Logo d'Appel IOS

II.5.2. Windows Phone de Microsoft

Le système Windows CE était largement distribué en Asie. Les deux variantes de ce système d'exploitation et Windows Mobile 6 standard lancé en février 2007 ont été peu adoptées.

Le 15 février 2010 Microsoft a lancé un nouveau système d'exploitation pour mobile, Windows Phone 7. Il inclut des services de Microsoft comme Windows Live, Zune, Xbox Live et Bing. Il intègre aussi des fonctionnalités média sociaux tels Facebook, Twitter, Snapchat... Comme Windows Phone 7 est une nouvelle plate-forme, il n'existe aucune compatibilité avec les anciennes applications Windows Mobile.

Les versions actuelles sont Windows Phone 7, Windows Phone 8 et Windows Phone 8.1, Microsoft rachète en 2013 la division mobile de Nokia afin de pouvoir améliorer et faire progresser son OS.

II.5.3. BlackBerry OS de Research In Motion (RIM)

Ce système fut créé à l'origine en 1999 pour le marché des hommes d'affaires. Il occupait 18,2% de parts de marché au 2e trimestre 2010. Récemment, RIM a favorisé la création d'application web et supporte les applications multimédia. À ce jour, BlackBerry's App World dispose de 6 000 applications.

II.5.4. Bada de Samsung Electronics

Ce système d'exploitation a été développé par la société coréenne Samsung qui devrait être installé sur ses téléphones traditionnels. Confirmant, à moyen terme la fusion des mobiles simples avec les Smartphones.

Le premier appareil qui exploite Bada est le Wave et fut lancé en 2010. Il s'agit d'un téléphone 100 % tactile qui dispose d'une boutique appelé Samsung Apps. Cette boutique possède maintenant plus de 5 000 applications. Samsung supporte aussi Android sur 30% de ses téléphones (en 2010).

Le 25 février 2013, Samsung annonce qu'il va arrêter le développement de Bada pour se concentrer, à la place, sur le développement de dizain.

II.5.5. Firefox OS de Mozilla

Firefox OS (précédemment connu sous le nom de Boot to Gecko ou B2G) est un système d'exploitation mobile libre proposé et développé par Mozilla Corporation [9].

II.5.6. Android de Google Inc

Android de Google Inc. fut développé par une petite startup qui fut achetée par Google qui poursuit activement son développement. Android distribué sous licence open source, est une variante de Linux. Google a lancé Open Handset Alliance qui regroupe des grands constructeurs et développeurs de logiciels (tel qu'Intel, HTC, ARM, Samsung, Motorola and eBay) [10].

II.5.6.1. Etude de la plateforme Android

Android est un système d'exploitation pour Smartphones, tablettes tactile, PDA et terminaux mobiles. Utilisant le noyau Linux optimisé pour un usage mobile, et sur une machine virtuelle Java assez fortement modifiée, nommée Dalvik JVM. Le développement se fait donc en Java, mais su cette JVM spécifique à Android.

II.5.6.2. Différentes versions

Depuis l'introduction du téléphone HTC Dream (T-Mobile G1, le premier Smartphone sous Android) on constate une explosion du nombre de téléphones qui l'ont adopté. De Q2 de 2009 au second trimestre 2010, la part de marché d'Android est passée de 1,8 % à 17,2 % avec un taux de croissance de 850 %. Et le nombre d'applications a fortement augmenté.

Le 15 novembre 2011, Android atteint les 52,5% de part du marché mondial des smartphones³. En septembre 2014, la part de marché mondiale Android est passée à 81%.



Figure II.2 Evolution des versions d'Android

En 2007, le 5 novembre, l'OHA a été officiellement annoncée, ainsi que son but: développer des standards open sources pour appareil mobile. Le premier standard annoncé a été Android, une plateforme pour appareils mobiles basée sur un kernel linux 2.6. En octobre 2008, apparait la première version d'Android qui n'avait pas reçu de nom. Cette version c'est avérée être la β du système [10].

➤ **Android 1.5** Cupcake corrigea le manque d'API et rendit le système plus utilisable. Depuis, Android 1.6, 2.0 et 2.1 ont apporté d'importantes améliorations respectivement sur les fonctionnalités et sur l'interface graphique du système.

- **Android 2.2** Froyo a fortement mis l'accent sur la synergie avec Internet. L'envoi d'applications et de liens instantanés depuis un ordinateur est désormais possible. Aussi, Google annonce-t-elle que le navigateur Chrome intégré à 2.2 est le navigateur mobile le plus rapide au monde grâce à l'intégration du moteur JavaScript V8.

- **Android 3.0** Honeycomb est spécialement étudié pour les tablettes tactiles. Les premiers modèles devaient être annoncés au CES 2011. On y apprend quelques nouveautés comme la prise en charge de la vidéo-conférence via Gtalk, la nouvelle interface Gmail ou encore le lecteur de livre électronique Google. La refonte graphique de l'interface utilisateur est assez réussie, plus d'informations devraient suivre dont sûrement des éclaircissements sur l'intégration ou non de l'interface de cette version d'Android sur les futurs Smartphones.

- **Android 4.0** Devrait arriver très vite (mi 2011) pour rajouter encore plus de fonctionnalités aux terminaux. Pour le développement, ces nouvelles versions d'Android devraient proposer de nouveaux composants permettant de réaliser des applications avec une ergonomie plus adaptée aux tablettes tactiles.

Remarque :

Android 3.0 et Android 4.0 devraient apporter plus d'outils aux constructeurs leur permettant de proposer des tablettes tactiles, qui seront capables de rivaliser (surtout au niveau de l'ergonomie) avec Ipad.

- **Android 4.1** Le 27 juin 2012 lors du Google I/O 2012, Google annonce la version 4.1 dénommée Jelly Bean basée sur le noyau Linux 3.0.31 et dont la principale nouveauté est une amélioration des fonctionnalités et des performances de l'interface utilisateur. Les améliorations des performances font partie du Project Butter (Projet Beurre en français) qui utilise le triple buffering.

- **Android 4.4** Le 3 septembre 2013, Google annonce que la version 4.4, portera le nom de KitKat alors que plusieurs rumeurs annonçaient le nom de Key Lime Pie (tarte au citron vert). Cette version est sortie le 4 novembre 2013.

- **Android 5.0** Annoncée le 15 octobre 2014 et sortie le 3 novembre 2014, Android 5.0 Lollipop est une évolution majeure d'Android qui propose de nombreuses modifications et nouveautés, et qui étend sa disponibilité sur de nouveaux supports tels que la télévision, la voiture ou les montres connectées [12].

II.5.6.3. Architecture Android

L'architecture de la Platform Android se décline, selon une démarche bottomup, en quatre principaux niveaux que sont :

- Le noyau Linux.
- Les bibliothèques et l'environnement d'exécution.
- Le module de développement d'application.
- Les différentes applications



Figure II.3 Architecture de système Android

Android est basé sur un kernel linux Au-dessus de cette couche, on retrouve les bibliothèques C/C++ utilisées par un certain nombre de composants du système Android. Au-

dessus des bibliothèques, on retrouve l'Android Runtime. Cette couche contient les bibliothèques natives du Framework ainsi que la machine virtuelle exécutant les applications [13].

Au-dessus de la couche "Android Runtime" et des bibliothèques natives, on retrouve le Framework permettant au développeur de créer des applications. Enfin au-dessus du Framework, il y a les applications, et voilà une brève explication sur chaque couche qui constitue le système.

II.5.6.2.a. Premier niveau : noyau linux

Android est basé sur un kernel linux 2.6 → 3.2 mais ce n'est pas linux. Il ne possède pas de système de fenêtrage natif (X window system). La glibc n'étant pas supportée, Android utilise une libc customisée appelée Bioniclibc.

Android utilise un kernel avec différents patches pour la gestion de l'alimentation, le partage mémoire, etc. permettant une meilleure gestion de ces caractéristiques pour les appareils mobiles.

Android n'est pas Linux mais il est basé sur un kernel Linux. Pourquoi sur un kernel linux ? Le kernel linux a un système de gestion mémoire et de processus reconnu pour sa stabilité et ses performances.

- Le modèle de sécurité utilisé par linux, basé sur un système de permission, est connu pour être robuste et performant. Il n'a pas changé depuis les années 70.
- Le kernel linux fournit un système de driver permettant une abstraction avec le matériel. Il permet également le partage de bibliothèques entre différents processus, le chargement et le déchargement de modules à chaud.
- Le kernel linux est entièrement open source et il y a une communauté de développeurs qui l'améliorent et rajoutent des drivers.

C'est pour les points cités ci-dessus que l'équipe en charge du noyau a décidé d'utiliser un kernel linux.

II.5.6.2.b. Deuxième niveau : Bibliothèques C / C++

Android dispose d'un ensemble de bibliothèques C / C++ utilisées par les différents composants du système Android. Elles sont offertes aux développeurs à travers le framework Android. Voici quelques-unes:

Système de bibliothèque C – une mise en œuvre dérivée de BSD de la bibliothèque C standard du système (libc), destinés aux systèmes embarqués basés sur Linux.

Comme cela a été dit précédemment, Android ne supporte pas la glibc, donc les ingénieurs d'Android ont développé une bibliothèque C (libc) nommé Bionlibc. Elle est optimisée pour les appareils mobiles et a été développée spécialement pour Android.

Les ingénieurs d'Android ont décidé de développer une libc propre à la plateforme Android car ils avaient besoin d'une libc légère (la libc sera chargée dans chaque processus) et rapide (les appareils mobiles ne disposent pas de CPU puissant).

La Bionlibc a été écrit pour supporter les CPU ARM, bien que le support x86 soit présent. Il n'y pas de support pour les autres architectures CPU telles que PowerPC ou MIPS. Néanmoins, pour le marché des appareils mobiles, seulement l'architecture ARM est importante.

Cette libc est sous licence BSD. Elle reprend une grande partie du code des glib issue d'OpenBSD, FreeBSD et NetBSD Ces caractéristiques importantes :

Elle pèse environ 200Ko, soit la moitié de la glibc implémentation des threads (POSIX thread) a été complètement réécrite pour supporter les threads de la machine virtuelle Dalvik. De ce fait, la Bionlibc ne supporte pas les threads POSIX.

Les exceptions C++ et les "wide char" ne sont pas supportés Médiathèques – basée sur PacketVideo de OpenCore; les bibliothèques permettant la lecture et l'enregistrement audio et vidéo, ainsi que la gestion des fichiers image, y compris MPEG4, H.264,MP3, AAC, AMR, JPG et PNG.

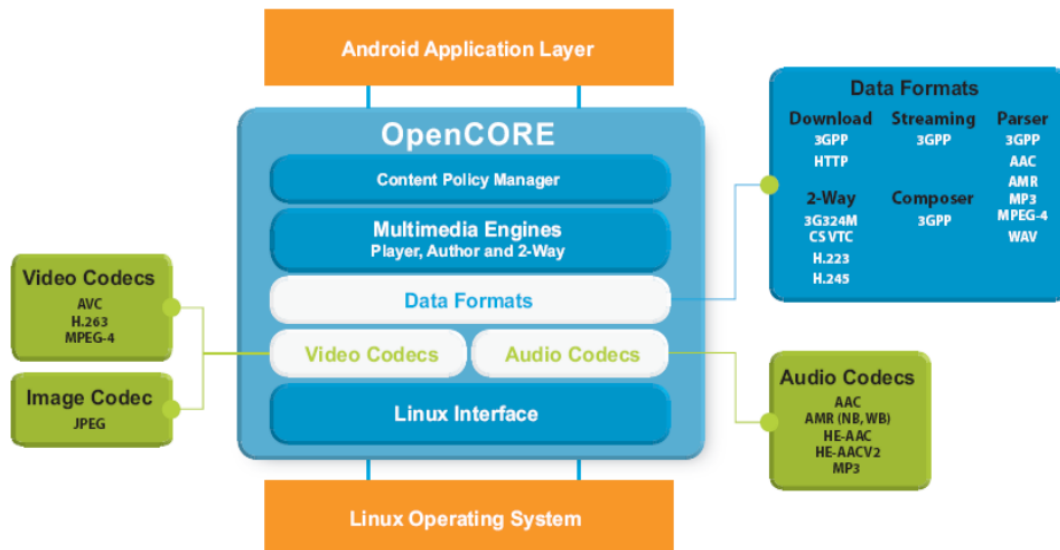


Figure II.4 Architecture de ces médiathèques

II.5.6.2.c. Troisième niveau : Framework de développement

En fournissant une plateforme de développement ouverte, Android offre aux développeurs la possibilité de créer des applications extrêmement riches et innovantes. Les développeurs sont libres de profiter du matériel périphérique et informations sur la localisation d'accès, exécuter des services d'arrière-plan, définir des alarmes, ajouter des notifications à la barre d'état.

Les développeurs ont un accès complet au même framework API utilisé par les applications de base. L'architecture d'application est conçue pour simplifier la réutilisation des composants; n'importe quelle application peut publier ses capacités et n'importe quelle autre application peut alors faire usage de ces capacités (soumis à des contraintes de sécurité appliquées par le framework). Ce même mécanisme permet aux composants d'être remplacés par l'utilisateur.

Toutes les applications sous-jacentes forment un ensemble de services et de systèmes, compris:

- Un jeu extensible de vues qui peuvent être utilisées pour construire une application.
- Des fournisseurs de contenu qui permettent aux applications d'accéder aux données d'autres applications (telles que les contacts), ou de partager leurs propres données gestionnaire de ressources.
- Un gestionnaire de notification qui permet à toutes les demandes d'afficher des alertes personnalisées dans la barre d'état.
- Un gestionnaire d'activité qui gère le cycle de vie des applications et propose une navigation commune [10].

II.5.6.2.d. Quatrième niveau : applications

Android est fourni avec un ensemble d'applications dont un client email, une application SMS, un calendrier, un service de cartographie, un navigateur... toutes écrites en JAVA.

II.6. Conclusion

Dans ce chapitre, nous avons identifié les Smartphones et leurs principaux systèmes d'exploitation en détaillant le système Android, que l'on va contrôler à distance à l'aide des outils présentés dans le chapitre suivant.



Chapitre III
Réalisation et mise en œuvre

III.1. Introduction

Dans la réalisation d'un projet de contrôle à distance d'un mobile, plusieurs paramètres sont pris en considération : le système d'un mobile choisi pour le contrôler à distance, le système du serveur de contrôle, mode de lien de connexion et le plus important le mode d'échanges des commandes entre le serveur et le client.

Dans le cadre de ce projet, la mise en œuvre du contrôle à distance nécessite un téléphone portable sous-système Android et un serveur sous système d'exploitation Linux et autres outils.

III.2.Cahier de charges

L'objectif de notre travail est de concevoir un système de contrôle à distance d'un mobile sous « Android » via un site Web hébergé sur un serveur.

Le « *Android remote contrôl systèm* » proposé dans ce mémoire permet de :

- Etablir un lien de connexion direct entre le serveur (Linux serveur) et le client (mobile).
- Transférer et appliquer successivement les commandes à distance sur le client.
- Identifier la version du système d'Android du client et les informations des interfaces de connexion utilisé.
- Récupérer tout le répertoire de contacts de mobile.
- Récupérer le journal des appels effectués par le mobile.
- Récupérer tous les SMS échangés par le mobile (émis et reçus).
- Prendre une photo à distance.
- Localisation de mobile.
- Télécharger n'importe quel fichier existant dans le téléphone mobile.
- Appliquer les commandes directes sur le « Shell » de système Android de mobile.

Le prototype à réaliser doit contenir :

- Un serveur web sous-système « *Linux* ».
- Les outils de programmation *PERL*, *PHP* et *BASH*.
- Projet Metasploit au niveau du serveur.
- Un téléphone portable avec système Androïde.
- Un serveur de base de données (*MySQL*).

III.3. Description de système de contrôle à distance

Le système est composé sur la base de client / serveur. Le client est défini par le mobile qu'on veut contrôler et le serveur est un site web.

Dans notre cas, le serveur intègre les parties suivantes :

- Kali Linux version 1.0.9 (c'est la meilleure distribution de Linux d'offensive sécurité).
- Un site web hébergé avec Apache web server.
- Metasploit comme une base de connexion transmetteur de commandes.
- Des scripts en *PERL* et *BASH* pour relier entre l'interface (en *PHP*) et le cœur de système Linux « SHELL » pour l'échange des commandes ou bien récupérer les données.

Tous ces éléments sont reliés entre eux pour construire une base de contrôle à distance d'un mobile.

Dans la partie suivante nous expliquons tous les éléments de ce système à partir de site web utilisé par l'utilisateur jusqu' au téléphone portable.

III.4. Technologie de modèle de contrôle

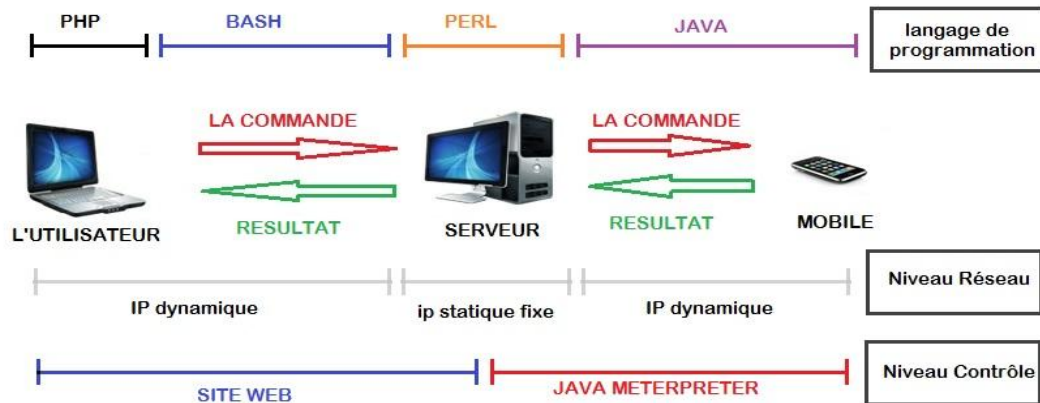


Figure III.1 Architecture du système de contrôle.

La Figure III.1 présente une description générale sur le système de contrôle, il existe deux parties générales dans ce projet, partie serveur et partie client.

III.5. Partie serveur

La partie serveur est composée par les éléments suivants :

III.5.1 Site web

III.5.1.1. Pourquoi un site web ?

Nous choisissons d'utiliser un site web parce qu'il satisfait aux plusieurs conditions :

- Par un site web, on peut augmenter le nombre d'utilisateurs de l'application, il suffit d'avoir un navigateur web installé dans n'importe quel équipement (*PC, Smartphone, tablette, PDA...*)
- On peut relier facilement le site web avec les autres outils au cœur du serveur de contrôle.

- On n'a pas besoin de fichier d'installation ou un système d'exploitation spécifique pour utiliser l'application, ce qui crée une diversité d'utilisateurs et aussi respecte le principe de mobilité.

III.5.1.2. Présentation de site web de projet

Nous avons utilisé le langage *PHP* pour programmé notre site web, Le langage *PHP* a justement été conçu pour créer des sites "vivants" (on parle de sites dynamiques).

Un site dynamique n'est pas figé, les pages vues par les internautes sont calculées et exécutées en fonction de chaque visite, numéro d'article demandé et heure ou jour de visite.

Le site web que nous développons est appelé (*Andoid Remote contrôl System*), le permet à chaque utilisateur avoir son propre profile pour contrôler son équipement (téléphone portable).

Une base de données est rattachée au site web pour assurer l'authentification, ce site est relié directement au cœur du serveur avec des scripts de type *BASH*, il a aussi des relations indirectes avec des scripts *PER Let* la base de contrôle *Metasploit_*

III.5.1.3. Architecture de site web

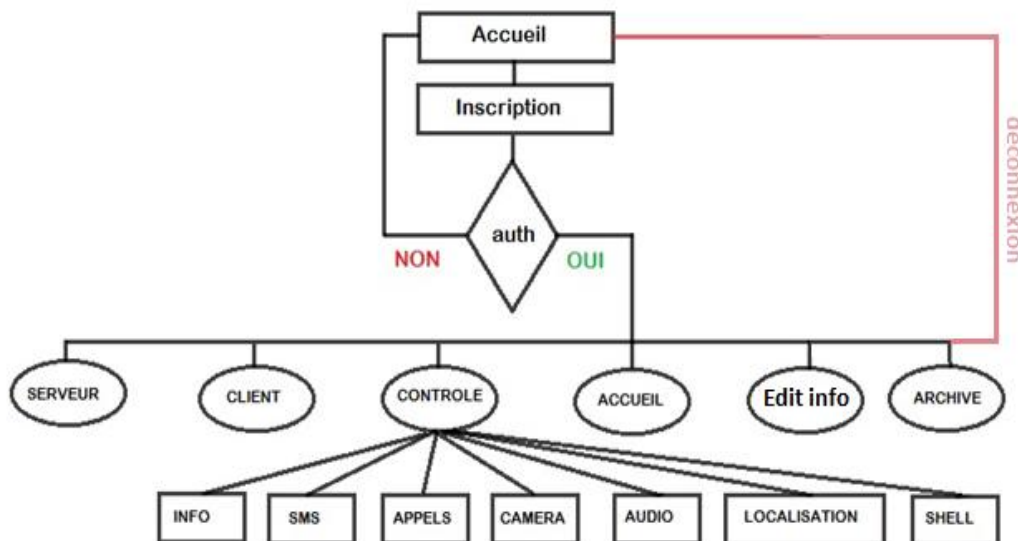


Figure III.2 Architecteur de site web

Le site web est composé de plusieurs parties. Il se divise en général en deux états :

- Etat hors session.
- Etat après session.

III.5.1.4. Hors session : Avant authentification

Lorsque l'utilisateur accède au site web il va trouver la page d'accueil. Cette dernière permet aux nouveaux utilisateurs de s'inscrire ou accéder à l'espace d'utilisateurs.

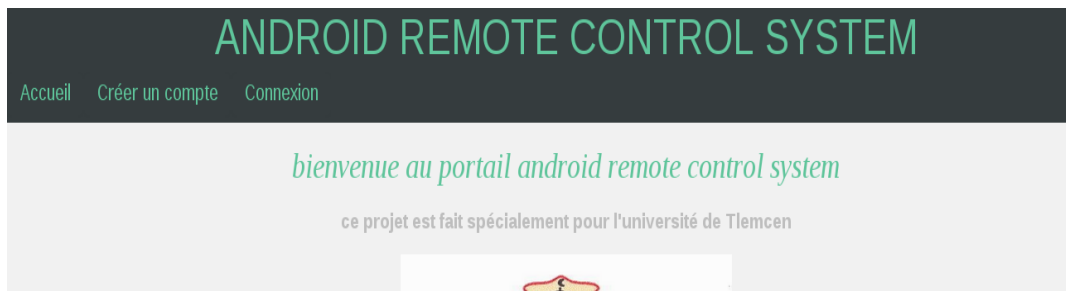


Figure III.3 Page d'accueil de site web



Figure III.4 Page d'inscription sur le site

Après que l'utilisateur remplit toutes les cases, les informations sont envoyées vers la base de données pour créer un nouvel utilisateur.

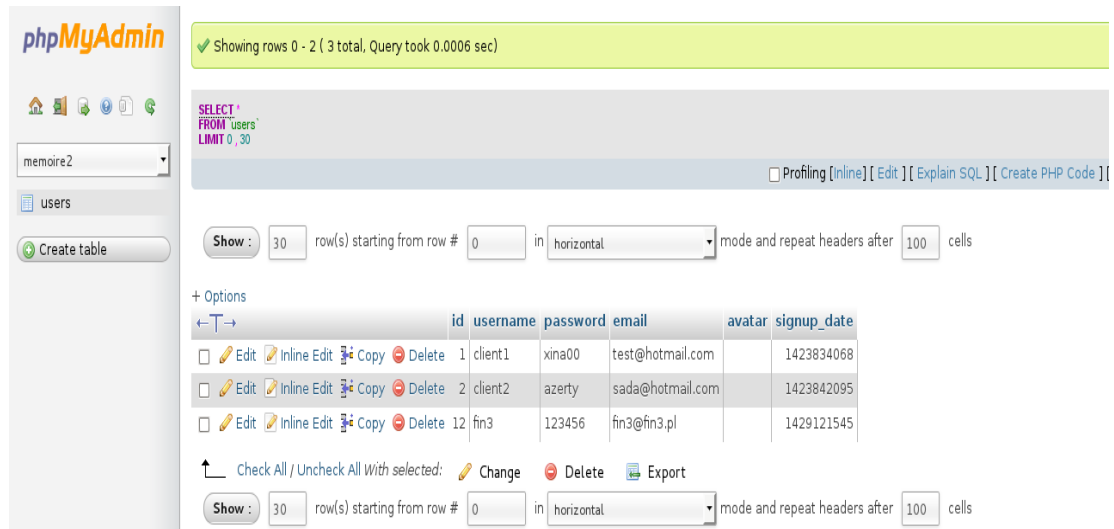


Figure III.5 Addition d'une ligne dans le tableau d'utilisateurs

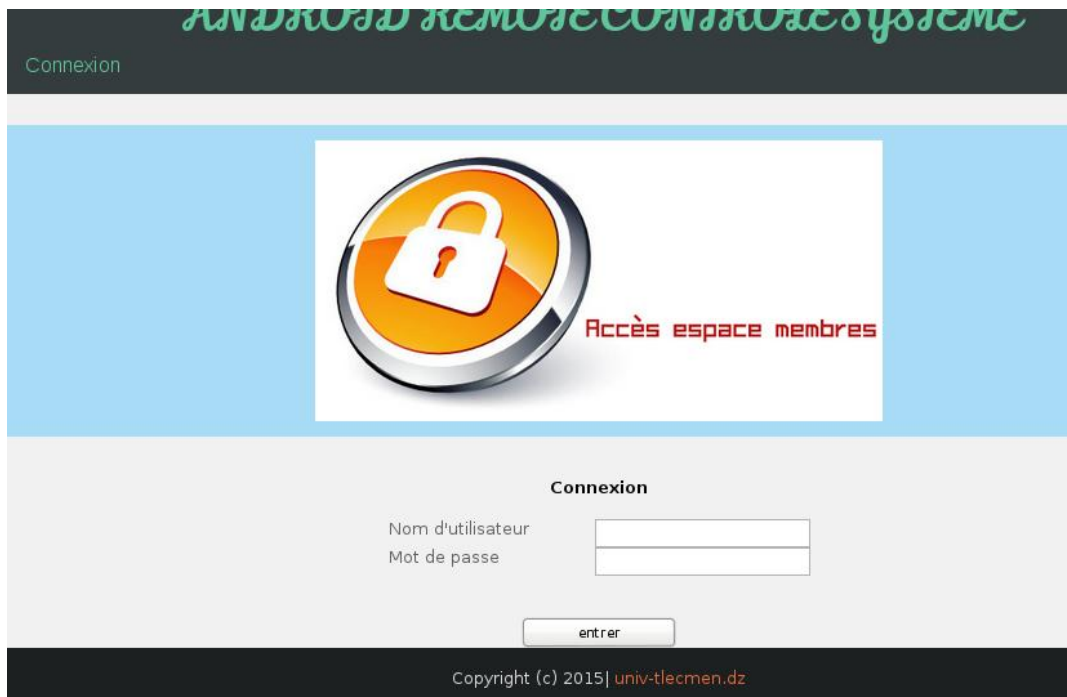


Figure III.6 Page d'accès à l'espace d'utilisateurs

III.5.1.5. Après session

Lorsque vous accédez dans l'espace d'utilisateur, l'entête de corps de la page web est changé, il contient autres options comme « mon serveur » « mon client » « contrôle » « édité info » et « archive ».



Figure III.7 Entête d'utilisateur fin3.

III.5.1.6. Page Accueil

La page « accueil » joue un rôle important, elle prend votre pseudo et crée un répertoire avec le nom de ce dernier afin de l'utiliser comme un espace de stockage pour les informations récupérées par le serveur, comme le fichier *SMS* et les coordonnées de localisation.

III.5.1.7. Page Mon Serveur

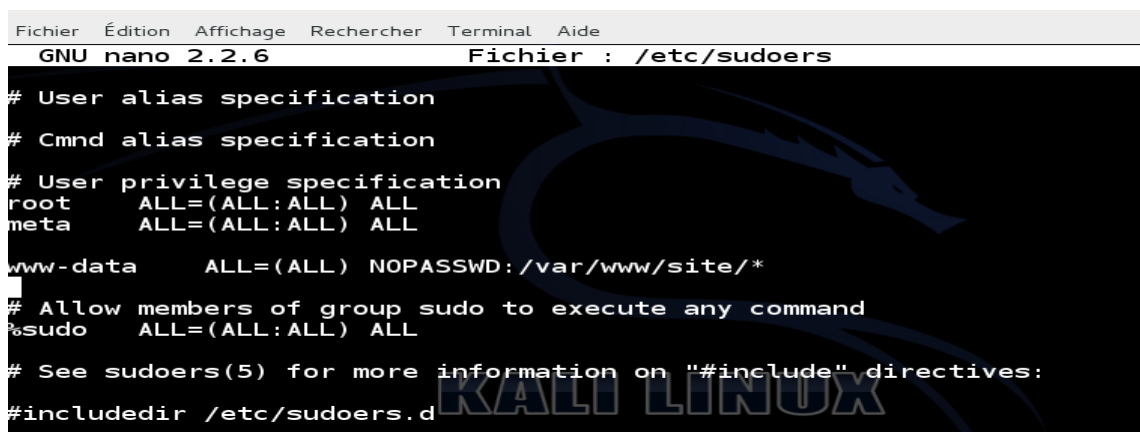
La page « Mon Serveur » est reliée directement avec le cœur de serveur : «SHELL» et le projet « Metasploit ».

Cette page est utilisée pour lancer votre serveur d'écoute de Metasploit avec un port bien spécifique. Chaque utilisateur a un port de l'écoute parallèle, tout dépend de son *ID*.

Il est impossible d'avoir deux utilisateurs avec le même port de l'écoute, cette option rend la séparation des connexions entre chaque utilisateur avec son client et autre utilisateur avec autre client.

A cause des conditions de sécurité, la permission du serveur web « www-data » ne peut pas appliquer une commande sous *SHELL linux* pour lancer directement l'écouteur de *Metasploit*.

Pour cela nous sommes obligés d'augmenter la permission d'utilisateur web "www-data" vers la permission de « root » par l'addition d'une ligne dans le fichier « sudoers » qui se trouve au répertoire /etc/. Nous avons également reliés le site avec des scripts *BASH* en appliquant la commande de lancement de l'écouteur.



```
Fichier  Édition  Affichage  Rechercher  Terminal  Aide
GNU nano 2.2.6  Fichier : /etc/sudoers

# User alias specification
# Cmnd alias specification
# User privilege specification
root    ALL=(ALL:ALL) ALL
meta    ALL=(ALL:ALL) ALL

www-data    ALL=(ALL) NOPASSWD: /var/www/site/*
#
# Allow members of group sudo to execute any command
%sudo    ALL=(ALL:ALL) ALL

# See sudoers(5) for more information on "#include" directives:
#include /etc/sudoers.d
```

Figure III.8 Changements dans le fichier sudoers

Après la consultation de la page « Mon Serveur », les propriétés de votre serveur d'écoute (comme *IP*, port, ... etc.) sont affichées.



Figure III.9 Page mon serveur

III.5.1.8. Page Mon Client

La page « Mon Client » permet de créer votre propre client. Le client est un agent de connexion (reverse-connexion) qu'on va installer dans le téléphone portable pour connecter avec le serveur.

Il va utiliser le même principe de séparation des ports qui est basé sur votre *ID* dans la base de données. Le client est un payload programmé en java d'*Android* qui établit une connexion avec le serveur.

La page « mon client » affiche les propriétés de votre client d'écoute comme le nom d'agent et le port.

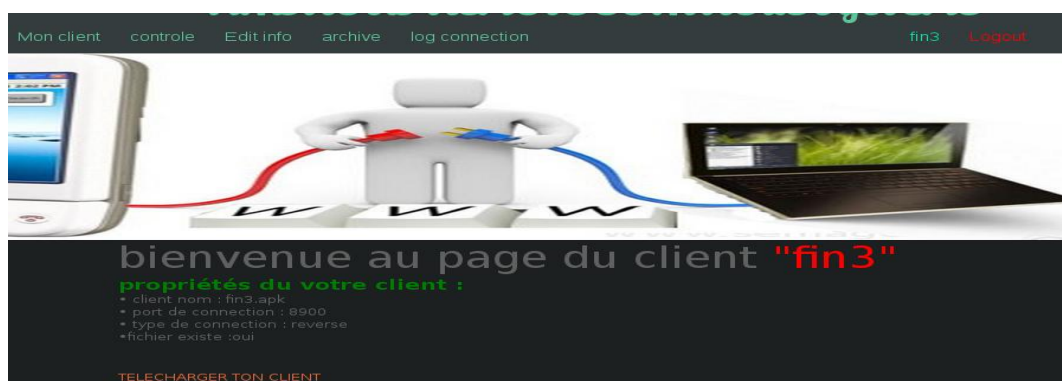


Figure III.10 Page « mon client »

III.5.1.9. Page Contrôle

Dans la page « Contrôle », un deuxième entête sera ajouté pour appliquer les commandes de contrôles. Ce dernier contient les éléments suivants :



Figure III.11 Entête de contrôle

III.5.1.9.a. Page Appareil

Dans la page « Appareil », on va trouvé toutes les informations du client tels que : les informations sur la version *Andoid* , routage et les informations sur tous les processus en cours d'exécution sur le téléphone portable.



Figure III.12 Identification d'appareil contrôlé

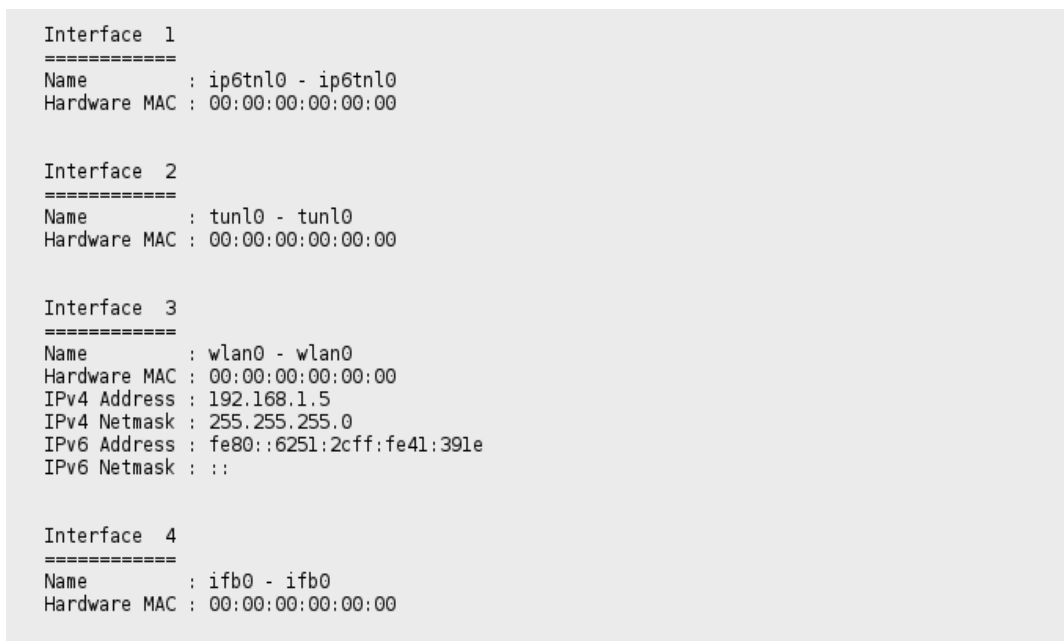


Figure III.13 Informations sur les interfaces dans le téléphone

III.5.1.9.b. Page SMS

La page « SMS » est utilisée pour consulter tout les messages dans le téléphone portable.

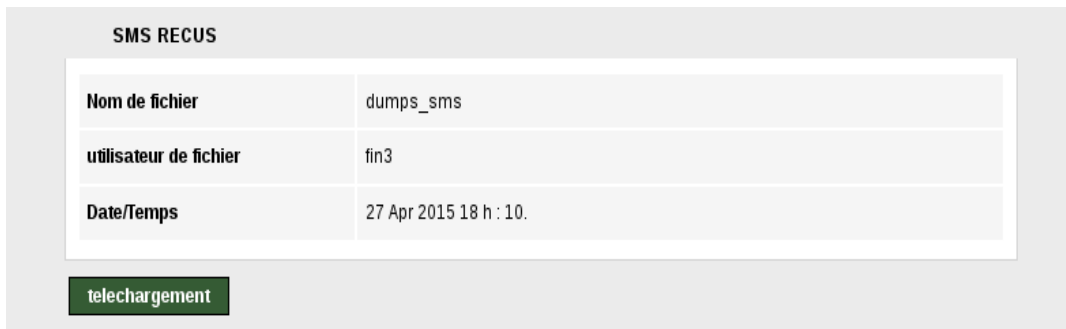


Figure III.14 Page de téléchargement des messages.

```

=====
[+] Sms messages dump
=====
Date: 2015-04-18 23:30:29 +0200
OS: Android 4.2.2 - Linux 3.4.5 (armv7l)
Remote IP: 192.168.1.5
Remote Port: 50358

#1
Type : Incoming
Date : 2015-04-18 23:51:44
Address : 213799717022
Status : NOT_RECEIVED
Message : $U$?iSecurity a publi une mise jour: ?????????? ??????? ??????? , ?????? ?? ???...?. https

#2
Type : Incoming
Date : 2015-04-18 18:37:53
Address : 213799717022
Status : NOT_RECEIVED
Message : $U$?iSecurity a publi une mise jour: ??? ?????? ??? ???? ??????? ?? ???????...?. https:/

#3
Type : Outgoing
Date : 2015-04-17 23:44:29
Address : +213773404952
Status : NOT_RECEIVED

```

Figure III.15 Exemple d'affichage les messages récupérés.

III.5.1.9.c. Page Appels

On utilise la page « Appels » pour consulter le journal des communications fait par le téléphone portable. On peut aussi afficher tout le répertoire des contacts sauvegardés sur le téléphone portable.



Figure III.16 Page de téléchargement du journal d'appels.

III.5.1.9.d. Page Camera

Dans la page « Camera », vous pouvez prendre une photo depuis le téléphone portable et l'afficher.

III.5.1.9.e. Page Audio

La page « Audio » utilisée pour l'enregistrement vocal à partir du microphone de téléphone portable. La durée d'enregistrement est définie par l'utilisateur du site web.

III.5.1.9.f. Page de localisation

La page de localisation utilisée pour afficher les coordonnées, le premier élément affiche les coordonnées du téléphone portable contrôlé à distance et le deuxième élément affiche exactement la position du téléphone portable à l'aide de Google Map. Il existe deux types d'affichage : par satellite et par la carte (Figure III.16).

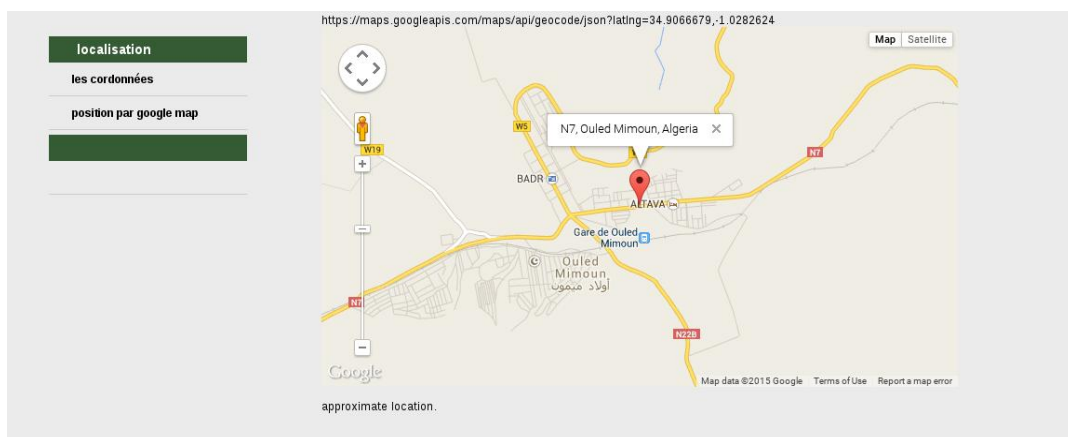


Figure III.17 Page de localisation par Google map.

III.5.1.9.g. Page SHELL

La page « *SHELL* » est utilisée pour appliquer les commandes directement au *SHELL* de mobile. Avec cette option on peut contrôler totalement le téléphone portable (l'éteindre ou exécuter quelque programmes et même télécharger n'importe quel fichier existant dans le téléphone portable).

III.5.2. Fonctionnement du serveur de contrôle

III.5.2.1. Génération de serveur d'écoute et son client

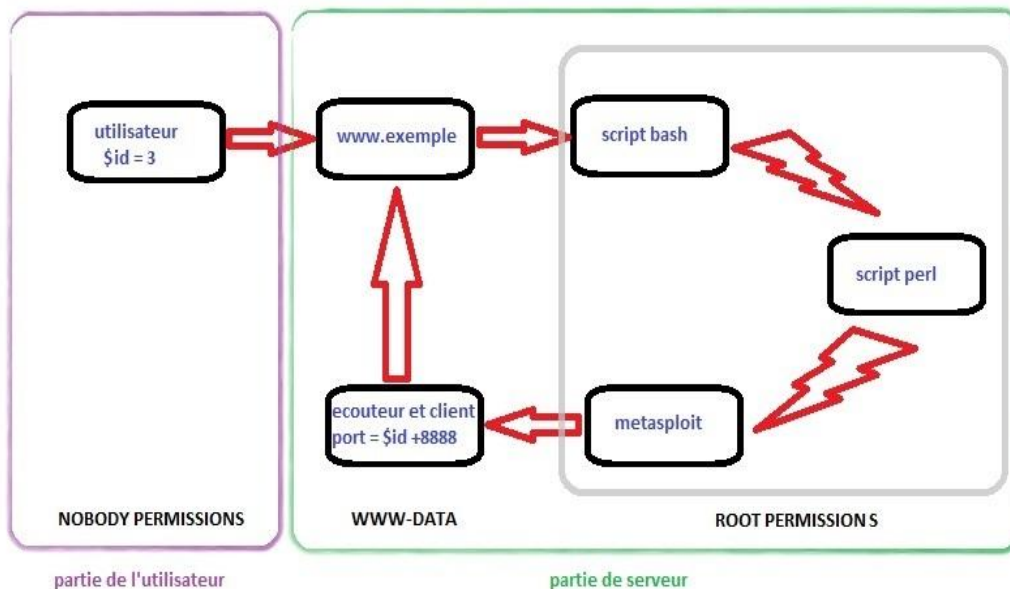


Figure III.18 Scénario de lancement l'écouteur et génération de client

La **Figure III.18** indique un scénario de lancement d'un écouteur qui a un port depend d'*ID* d'utilisateur dans la base de données. Dans notre cas nous avons choisil'addition d'un constant 8888 pour éviter les ports de services(*ftp* :21 ,*http* :80...).

III.5.2.2. Exécution des commandes

L'exécution d'une commande depuis l'utilisateur se passe par le serveur, l'utilisateur change le contenu de fichier « command.txt » qui existe dans son propre repertoire.

Ensuite, le script listner.pl utilise une fonction qui s'appelle « expect » pour remarquer les changements dans le fichier command.txt (**figure III.19**).

Le résultat va être envoyé par le mobile vers le serveur qui le sauvgarde dans le fichier « log.log1 ».

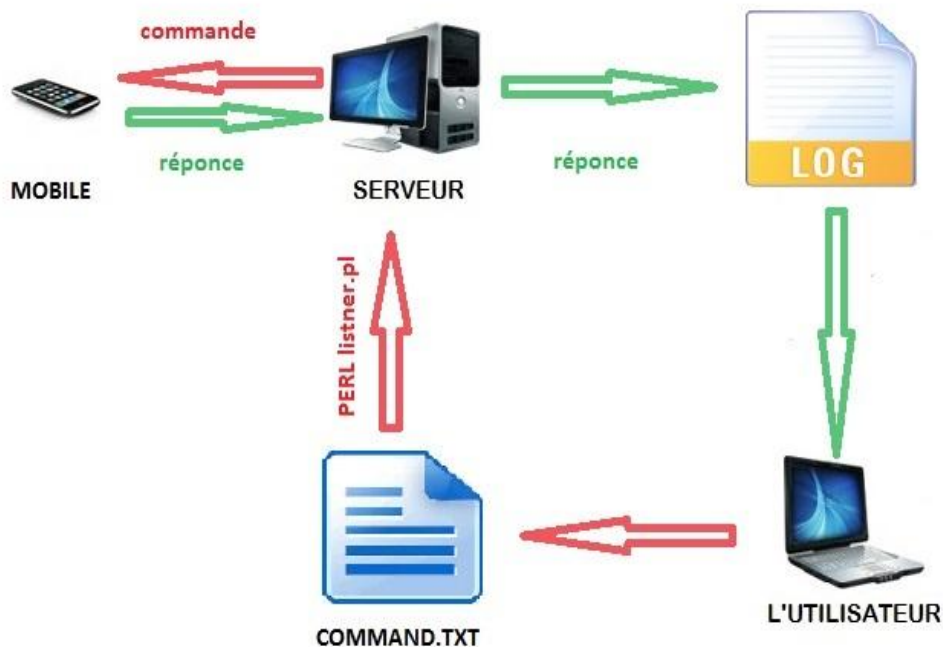


Figure III.19 Scénario d'applique d'une commande

III.5.2.3. Script GET.PL

Après la connexion du client vers le serveur, le fichier « get.pl » commence automatiquement de récupérer les données de téléphone portable affichées par les pages web de site, par exemple: les informations du système du client, la localisation, SMS, appels ...etc.

Les commandes qui ont besoin d'un temps réel pour l'exécution, ne sont pas appliquées par script « get.pl », comme prendre une photo ou bien appliquer une commande dans le *SHELL* de portable.

III.6.Partie Client

La partie Client est simple, il faut juste installer l'agent (payload) qui est généré par le Metasploit et lancé, on peut télécharger ce payload à partir de la page « mon client ».

III.7.Mode de connexion

La connexion entre le client et le serveur est une connexion par paquets, les échanges des informations ou les commandes de contrôle se font par un port bien spécifique. Le client est connecté d'une façon « reverse-connexion » avec le serveur qui a une adresse *IP* fixe connue par le client. Ce qui assure la mobilité de client et la multi-connexion entre plusieurs appareils et le serveur en même temps.

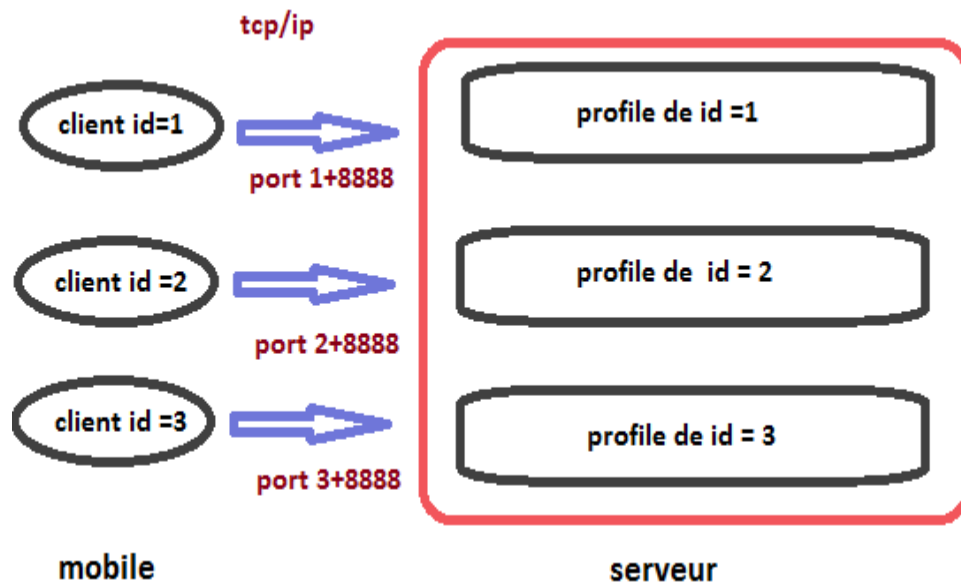


Figure III.20 Mode de connexion entre les clients et le serveur

III.8. Autres fichiers nécessaires pour le projet

Le fichier « config.php » est situé à la racine de votre site, il permet de faire le lien entre la base de données et le site.

L'installation de *module* «*Expect.pm* » est effectuée à l'aide de son panneau de configuration cpan par la commande « Install expect ».

III.9. Conclusion

L'exploitation des Smartphones introduit de nouvelles problématiques et crée des nouveaux besoins. Ils permettent à des usagers (éventuellement mobiles) d'avoir accès à des services indépendamment de leurs positions physiques. Dans cette nouvelle vision, nous avons proposé un système de contrôle à distance pour les équipements Android et par conséquent, d'exploiter ces types des terminaux pour des domaines hors les fonctionnalités de la téléphonie.

Conclusion générale

Conclusion générale

Dans ce projet de fin d'étude, l'objectif visé consistait à concevoir et réaliser un système de télé-contrôle d'un téléphone mobile. Ce système se compose en deux parties :

- Une partie qui s'occupe des configurations d'utilisateur et l'interprétation des commandes.
- Une partie pour recevoir les commandes, appliquer et envoyer le résultat.

Cette solution, non coûteuse et facilement réalisable, est adaptée aux appareils portatifs assurant le contrôle à distance des appareils Android à n'importe où depuis n'importe quel position. C'est dans cette vision que d'autres services, associés aux téléphones mobiles et destinés au domaine extra-communication seront développés.

Enfin, ce travail s'inscrit dans le cadre des projets innovants et a impact socio-économique développé dans laboratoire de recherche des Systèmes et Technologies de l'Information et de la Communication (STIC) de l'Université de Tlemcen.

Références & Bibliographie

Référence & Bibliographie

[1] A. RADU, «Évaluation de la Qualité de Service par l'utilisateur final dans les systèmes mobiles», Thèse de doctorat en Informatique et Télécom de l'Université de Mame-La-Vallée, France, Mars, 2004.

<http://pelleas.univ-mlv.fr/document/UMLV-2004-000235-PDF.pdf>

[2] Frédéric Jacquenod « Normalisation des réseaux »

<http://portail.jacquenod.net/Web/Tutoriaux/normalisation.pdf>

[3] P. GODLEWSKI, X. LAGRANGE, S. TABBANE, «Réseaux GSM-DCS», 4e Édition Hermès, Paris, France, 1999.

ISBN: 2-7462-0028-7.

<http://www.decitre.fr/livres/RESEAUX-GSM-DCS.aspx/9782746200289>.

[4] U. HORN, R. KELLER et N. NIEBERT, «Services mobiles interactifs – La convergence de la radiodiffusion et des communications mobiles », UER – Revue Technique, N° 281, pp. 1–10, Automne 1999.

http://www.ebu.ch/en/technical/trev/trev_281-umts_f.pdf

[5] G. PUJOLLE, «Les réseaux», Édition EYROLLES, Paris, France, 2008.

ISBN: 978-2-212-11757-8.

http://www.friendlyduck.com/AF_TA/rel/index.cfm?RST=UNF&TAD=420036.

[6] K. IBRAHIMI, «Gestion des ressources des réseaux mobiles de nouvelle génération par rapport à la mobilité des utilisateurs», Thèse de doctorat en Informatique, Sciences de l'Université d'Avignon et des Pays de Vaucluse France & Université Mohammed V-Agdal Rabat - Maroc, Laboratoire LIA, Avignon, France, Laboratoire LIMIARF, Rabat, Maroc, 20 Novembre 2009.

http://tel.archives-ouvertes.fr/docs/00/45/36/44/PDF/2009AVIG0169_0_0.pdf

[7] Rabah MERAIHI, «Gestion de la qualité de service et contrôle de topologie dans les réseaux ad hoc», Thèse présentée en Informatique et Réseaux pour obtenir le grade de docteur de l'Ecole nationale supérieure des télécommunications, ENST, Paris, France, 2003.

http://pastel.archives-ouvertes.fr/docs/00/49/99/43/PDF/Rabah_Meraihi.pdf

- [8] Définition et historique du Smartphone - SCT TELECOM
<http://www.sct-telecom.fr/glossaire/smartphone/>
- [9] wikipedia Système d'exploitation mobile
fr.wikipedia.org/wiki/Système_d'exploitation_mobile
- [10] Khaoula MRABET & Nessrine TRABELSI Application de messagerie simple sur Android, Rapport de projet de VAP RSM
<http://www-public.it-sudparis.eu/~afifi/rapport%20projet%20RSM%20android.pdf>
- [11] Elabed nawfal , Hachmi abbess , Sami ben amor Système d'exploitation Android [ppt] <http://ekldata.com/mba.eklablog.com/perso/se%20pour%20mobile/systeme%20d---exploitation%20android.pptx>
- [12] Wikipedia Android_Lollipop
http://fr.wikipedia.org/wiki/Android_Lollipop
- [13] http://www-igm.univ-mlv.fr/~dr/XPOSE2008/android/archi_comp.html

Résumé

Le contrôle à distance d'un système mobile acquit de plus en plus l'attention d'une nouvelle société qui propose des services à valeur ajoutée dédiés aux différents domaines tout assurant aux utilisateurs, le confort, la fiabilité, la sécurité et la confidentialité.

L'objectif du présent travail consiste à concevoir et réaliser un système de télécontrôle permettant de commander à distance les fonctionnalités d'un téléphone mobile sous Android. Un tel projet aurait nécessité un gros moyen ainsi qu'une grosse infrastructure, aujourd'hui, un simple téléphone peut suffire.

Mots clés : téléphone portable, contrôle à distance, *Android*, *Linux*, *Métasploit*.

ABSTRACT

The remote control of a mobile system gained more and more attention to a new company that offers added value services dedicated to different fields while ensuring users, comfort, reliability, security and confidentiality.

The objective of this work is to design and implement a remote control system for remotely controlling mobile phone functions running with Android. Such a project would need a huge means and a big infrastructure; today a simple phone can be sufficient.

Key words: mobile phone, remote control, *Android*, *Linux*, *Métasploit*.