Potential Role of the SIM in the Internet in a Secure Single Sign-On (SSO) Authentication Solution for Online Banking: Design, and Evaluation of Security and Performance Issues.

Ohwadua, Emmanuel Obaro

Student Number: 10059669

Supervisor: Keith Mayes

Submitted as part of the requirements for the award of the

MSc in Information Security at Royal Holloway, University of London.

I declare that this document is all my own work and that I have acknowledged all quotations from the published or unpublished works of other people. I declare that I have also read the statements on plagiarism in Section 1 of the Regulations Governing Examination and Assessment Offenses and in accordance with it I submit this project report as my own work.

Signature Date

ACKNOWLEDGEMENT

My sincere thanks go to my Supervisor – Dr Keith Mayes and my Advisor – Professor Kenny Paterson for their time spent, and invaluable advice in producing this report. I equally wish to thank all my lecturers and non-academic staff of Information Security Group (ISG) for their understanding and support throughout my studies at Royal Holloway.

My gratitude to Vodafone Group, Netherlands for the opportunity given to me for my research internship, and in particular many thanks to the R&D team – Patrick Waters, Paul Lahaije, Najib Koraichi, Aguibou Barry, Laurent Eschenauer, Alard Weisscher, Menw Hurkens, and every other member not mentioned for their kind support and cooperation during my internship without which I would not have been able to produce this report.

Lastly, my special appreciation goes to Professor Fred Piper, Dr Chez Ciechanowicz, Professor Peter Wild, and Mrs Pauline Stoner for a memorable ISG experience in Royal Holloway.

TABLE OF CONTENTS

TABLE OF FIGURES

LIST OF ABBREVIATIONS

B2B: Business-to-Business

B2C: Consumer-to-Business

GSM: Global System for Mobile communications

SAML: Security Assertions Markup Language

XML: Extensible Markup Language

HTML: Hypertext Markup Language

SIM: Subscriber Identity Module

USIM: Universal Subscriber Identity Module

SOAP: Simple Object Access Protocol

SSO: Single Sign-On

UML: Unified Modeling Language

WS: Web Server

SP: Service Provider

RP: Relying Party

AP: Authentication Provider

IDP: Identity Provider

OP: OpenID Provider

SMS: Short Message Service

IMSI: International Mobile Subscriber Identity

TMSI: Temporary Mobile Subscriber Identity

PIN: Personal Identification Number

ATM: Automatic Teller Machine

CNP: Customer Not Present

POS: Point of Sale

EMV: Europay MasterCard and Visa

IID: Internet Identity

PKI: Public Key Infrastructure

SSL: Secure Socket Layer

TLS: Transport Layer Security

ME: Mobile Equipment

AuC: Authentication Centre

HLR: Home Location Register

VLR: Vistor Location Register

MSISDN: Mobile Subscriber Integrated Services Digital Network

XRES: Expected Response

SRES: Subscriber Response

RAND: Random Number

WAP: Wireless Access Protocol

OTA: Over the Air

APDU: Application Protocol Data Unit

SAT: SIM Application Toolkit

NFC: Near Field Communication

EXECUTIVE SUMMARY

Internet identity has over the years been a serious challenge to online communities since the emergence of e-commerce. Unlike in the physical world where identity instruments such as ID cards, international passports, and even biometric data are carried by the individual owner and presented upon request while physically present, the identification instruments on the internet is conducted by the submission of the identification details/attributes of the identification instrument without the physical presence of the individual. In other words, once the owner of an identification instrument is separated from the instrument, then logically any individual can claim the identity of another person or even fake an identity that does not exist physically. This brings us to the thorny issue of identity theft and multiple identities on the internet. This research paper is concerned with this thorny issue of internet identity with emphasis on customer's authentication in online banking.

Since the emergence of online banking, it has been shrouded with incidences of fraud ranging from phishing, pharming, username/password abuse, and malware attacks resulting to loss of millions of investors fund, credibility crisis and user apathy/frustration. With all these in mind and various solutions that have attempted to deal with this problem over time, our proposed solution in this paper is concerned with the ubiquitous SIM in a SIM-based SSO authentication for online banking.

Chapter 1 is concerned with the introduction, the definition of the problem, and our research objectives intertwined by a brief review of the challenges of current username/password based SSO authentication solutions. In chapter 2 we discussed related

literatures bordering on standards/specifications such as Microsoft Web Services, Liberty Alliance, OASIS, and OpenID on internet identity. We equally discussed Microsoft .NET Passport as an example of an SSO authentication application and finally concluded the chapter with a peep into GSM U(SIM) authentication and security services. Our chapter 3 focused on the protocol design architectures consisting of protocol flows, challenge/response mechanisms, and system use case representation using sequence diagrams. This was followed by a security and performance evaluations of the design, and demonstration of proof-of-concept in chapter 4 with online banking as our use case.

In chapter 5, we carried out a further security and performance analysis to identity possible security challenges and proposed appropriate cryptographic mechanisms to deal with those potential attacks. We equally considered the pros/cons of the system and finally concluded with implementation considerations. Finally, our chapter 6 deals with conclusion and recommendations for future research.

CHAPTER 1

INTRODUCTION

The internet, over the years has grown in size and complexity as more and more service providers and users are embracing the cyberspace for their day-to-day transactions – from buying and selling, banking, social services, to B2C and B2B communications across the internet. But as the volume of activities of e-commerce increases with service providers delivering more content online, the internet has become more sophisticated with attendant increases in fraud, identity theft, and privacy concerns stemming from increasingly criminal practices such a phishing, pharming and malware. The numerous service delivery websites across the internet have resulted to the multiplicity of usernames and passwords users must remember/memorize in order to access protected resources on these sites. The variety of methods of authenticating to these sites result not only in user frustrations and dissatisfaction, but also harmful practices such as re-using of the same login credentials at many sites, writing down of login account details, or using browser cookies to store the username and password for each service provider's site (1). Thus accessing services on the internet has become an ordeal each time a user tries to login, to the extent that a user account risks the chance of being blocked after trying for more than once necessitating another round of re-

registration process. The frustration did not even end there, as some service providers in an attempt to improve the security of the authentication process, requires user's to enter other confidential personal details such as the maiden name of your mother, or the characters of certain positions in your passwords – any mistake – the user's account is blocked.

1.1 Current SSO Solutions and Challenges

The nightmares that users pass through on the internet in username/password based authentication solution are countless, and vary from service-provider to service-provider. In the search for a better solution to alleviate the pains of users, various SSO (Single Sign-On) authentication solutions have been proposed and developed, amongst which are .NET Passport based on Web Service (WS-*) Specification, OpenSSO promoted by Sun Java, Novell SecureLogin etc. SSO is an authentication mechanism that allows a user to access protected resources of various service providers (SP) within a network by a single login credential. Having registered with a service provider via an authentication provider (center), the user at the point of login is required to enter his/her login details once to the authentication (identity) provider (AP/IDP) which authenticates the user on behalf of all the SP that belong to the network. However, despite the robustness of the username/password based SSO, the solution is still fraught with operational and technical challenges ranging from vulnerability of username/password,

phishing, privacy issues of user's personal information stored by identity providers, and even suspicion or mistrust among the tripartite parties – the user, the AP and SP.

1.2 The SIM SSO Solution

The inadequacies of the username/password based SSO have further necessitated the quest for an alternative solution that would at least, address the limitations inherent in the SSO technology. This is where the potential role of the SIM (subscriber identity module) to providing identity in the internet is considered to offer a better and sustainable SSO solution for internet identity management. The SIM is a smart card – tamper-resistant device that is issued by GSM operators such as Vodafone, to enable subscriber's access to mobile telecommunication services. The SIM is plugged into a mobile equipment – handset, and together identifies users to the mobile network, secures communication with the network, and stores subscribers information such as phone book, SMS (Short Message Service). It equally enables the network provider to offer value-added services to the subscribers. The SIM has over a decade of its existence provides quality authentication services to its users coupled with other security services such as confidentiality and integrity of communication infrastructures, which has endeared the SIM to over *"2 billion"* (2) subscribers worldwide. This research project is therefore leveraging on the success story of the ubiquitous SIM with its service availability and trust, to extend its horizon to providing

additional value-added services – internet identity authentication for online communities to replace the frustrating generic username/password authentication mechanism.

1.3 Objectives

The goal of this project is to provide an authentication system that will give users peace of mind; that will remove the frustrations and annoyances associated with username/password authentication mechanisms while at the same time ensuring a safer and more trustworthy internet. To the service providers, the gains of this system will translate to a more efficient and effective service delivery as the issue of bureaucratic username/password authentication mechanisms and instances of forgotten-passwords would give way to a better, more friendly and enduring e-commerce and participation by online communities. Above all, our main objective is to eliminate phishing attacks which have resulted to millions of investor's money lost over the years to internet banking fraud.

In addition, this authentication mechanism will provide a stable and long-term solution that will give anonymity to the user's identity by disclosing the least amount of identifying information – only the IMSI (International Mobile Subscriber Identity) of the user's SIM. In other words, the authentication service (identity) providers such as Vodafone for instance, will only be required to store in their database the IMSI and mobile numbers of users which according to Cameron in its *"Laws of Identity"* satisfies

"Minimal Disclosure for a Constrained Use" – that it is best to acquire identifying information on a *"need to know"* basis; and to retain it only on a *"need to retain"* basis (3). Thus if these principles are adhered to, we can achieve a substantial information security risk reduction in our internet identity management.

Summarizing, our objectives are:

- To replace username/password authentication solution in online banking – "what you know" with "what you have".
- To prevent phishing attack.
- To provide user anonymity.
- To build a system that is convenient with end-user and gives user control over their identity information.

1.4 Motivation

In realizing the above objectives, we were stimulated by the ubiquity of the SIM with its 24/7 service availability, together with its proven security services such as user's anonymity as well as the confidentiality, and integrity of communications infrastructure to provide a better authentication mechanism for accessing restricted resources on the internet. In addition, the unabated increase of phishing attacks on online banking despite various internet identity solutions equally generates our enthusiasm in

contributing to the search for a secure and sustainable solution that will promote the trust and confidence of online communities in internet banking.

1.5 Methods of Research

Our research approach shall rely on the security infrastructure of the tamper-resistant SIM, to design a single sign-on (SSO) application based on a unique identity of the SIM – IMSI. In this design, the GSM operator shall offer the services of the identity/authentication provider in the form of value-added services using its existing infrastructure. In other words, a user is authenticated with what you own/have – SIM and phone handset, and not what you know – static password. Our use case shall be user/customer authentication in online banking. The design shall be modeled using UML (unified modeling language) to define the use cases, followed by a demonstration of the proof-of-concept application of the design. The protocol flows, protocol messages, and protocol actions will be described and evaluated. Our design shall be based on OpenID specification. OpenID specification provides a way to prove that end users have control over their identifying information and it does this without the relying party (service provider) needing access to end user authentication credentials such as password or other private information such as email address (4).

1.6 Chapter Summary

In this chapter, we have defined the statement of our problem and the background of study followed by a brief review of the challenges of current username/password based SSO authentication solutions. We have equally proposed the SIM SSO authentication for internet identity management as an alternative solution. This chapter is concluded by the statements of our objectives, our motivation and finally, research approach. In chapter two, we shall be reviewing relevant literatures to highlight their various frameworks, strengths, and weaknesses. In particular, we shall discuss some notable SSO specifications such as WS-*, Liberty Alliance, OpenID, and Oasis. In addition, we shall discuss GSM security technologies and its authentication process.

CHAPTER 2

REVIEW OF RELATED LITERATURE

There is no doubt that the internet has become the sine-qua-non to various forms of activities ranging from commerce, business, governance, games, social interactions to sharing of information, creation and distribution of media, and education. For e-commerce and e-business, the driving force has been cost minimization and access to world market as the internet has no boundaries. "*The internet is a loosely-organized global collaboration of autonomous, interconnected networks, supports host-to-host communication through voluntary adherence to open protocols and procedures defined by Internet Standards"* (5). As most activities conducted in the "physical" world establish their presence in the "virtual" world (the internet), in the same vein, the identity of players such as individuals, organizations, and devices have to acquire online/internet identities in order to be authenticated to facilitate service delivery in the "cyberspace". In the cyberspace, you cannot trust who you communicate with; people aren't who (or even where) they claim they are (6). Thus, this challenge was the thrust for the development of various authentication factors that we now have today.

2.1 Authentication Methods

Authentication is a process by which a user – a person, device or piece of software proves their identity to a system or application offering a service, which can be categorized as follows (7):

- what you know – e.g. username/password, PIN;
- what you have – e.g. smart card, hardware security token;
- what you are – e.g. biometric such as fingerprint;
- some combination of the three.

When two authentication methods are used to establish an identity, we refer to such authentication as a two-factor authentication; an example is the use of a smart card and PIN in ATM transactions. In other words, multiple authentication factors are a means of improving the security of the authentication system.

2.1.1 What you know

This is an access control mechanism that is designed to accept something that a user has knowledge of such as a username/password or a PIN (personal identification number) in order to be authenticated to a system. The system requiring authentication could be a service provider whose service is hosted via a website on the internet, or POS (point of sale) terminal and ATM (automated teller machine). The POS terminal is

usually for the payment of goods in supermarket checkouts or retail shops while the ATM is an unattended cash dispenser terminal operated by banks to dispense cash to customers 24/7. Both the POS and ATM require two authentication features which may be regarded as two-factor authentication – PIN and a smart card. The PIN is usually a 4 to 6 digit number for ease of memorization. During the authentication process, a user is prompted by the terminal to enter the PIN after the insertion of the smart card to perform the authentication operation. On the other hand, to access a protected resource from a service provider (SP) via the SP's website, the user is required to enter his/her authentication credentials such as a username/password which may be as long as 6 to 15 alphanumeric or ASCII characters depending on the authentication security policy of the organization. In the terminal operations, particularly in POS, the merchant had some clue to the authenticity of the customer and his presented credentials, because the *"Cardholder was Present"* (2); whereas accessing a protected resource in a website, the merchant has no clue about the authenticity of the customer except the claim or authentication credentials such as the username/password entered at login.

2.1.2 What you have

This authentication approach involve the use of a device such as magnetic-stripe card, smart card or a hardware security token to perform authentication operations, and in most cases the user may be required to equally have secret PIN. A magnetic-stripe

card consists of a magnetic stripe located at the back of the card where the user's data can be read and written to. During authentication, the magnetic stripe is read by pulling it across a read head, either manually or automatically, and the user may be required to enter his PIN as an additional authentication factor in a two-factor mechanism (8). An example of this identification scheme is a staff identity card. The difference between a magnetic-stripe and a smart card is that a smart card – which may be referred to as a chip card, is a tamper-resistant device that consists of embedded integrated circuit and can be used for transmitting, storing and processing data. Smart card can be classified by chip type into memory chip and micro-controller chip; and by data transmission method into contacts, contactless and dual interface. A Good example of smart card is EMV card such as Visa and MasterCard. A smart card is usually combined with a user's PIN for authentication at POS or ATM terminals. A token on the other hand, is a hardware security device that is used to generate random numbers in the form of a PIN that a user is required to enter during online authentication session. An Example is the security token issued by banks to their customers for online transaction.

2.1.3 What you are

This authentication option involves the use of biometric measurement of the user. Biometrics is the science of measuring various aspects of living – typically human beings, making analytical judgments on these measurements, and taking appropriated

decision based on those judgments (9). In biometric identity scheme, an individual identity is confirmed by “who she is” rather than “what she possesses” such as an ID card, or “what she remembers” (e.g., a password) (10). Examples of commonly used biometrics are fingerprints, face, signature, voice, and iris. Biometrics applications are commonly found in identification system that requires high security such as border control and military installations. Biometric identification system can also be used for crime control in the form of “negative recognition” to prevent an individual from claiming multiple identities. This is one of the main differences with other identification schemes such as password, PINs, tokens which are used for “positive recognition” while negative recognition can only be achieved through biometric system. Negative recognition operates on identification mode where the biometric data of an individual is compared to an entire database of biometric templates to find a match; it is a one-to-many comparison. In positive recognition, the biometric system operates in verification mode where the biometric data of an individual is mapped on a one-to-one basis with her biometric template stored in the system database to establish a matching which may either be a true or false match. The main significance of positive recognition is to prevent many people from claiming the same identity which is equally the main function performed by traditional authentication schemes such as password, ID cards, and smart cards.

2.2 Internet Identity Authentication Mechanisms

Internet identity (IID) is a unique piece of data that is associated with a user, entity or principal. An entity, principal or subject is a person, organization, software program, machine, or other thing making a request to access a protected resource, which may be a web page or a piece of data in a database (11). The IID may be a username and password created by the user in accordance with the authentication policy of the service provider, or may be generated by a device such as a hardware security token as in the form of a PIN. Unlike in the “real world” where a merchant will be able to have a physical interaction with his customers, in the cyberspace the reverse is the case – the customer can assume any identity which may or may not reflect the actual identity of the customer. For fear of stolen or abuse of personal details, the customer is always afraid of giving his/her real personal credentials such as name, date of birth, address or credit card details. Apart from the identity theft and privacy issues, users may have to grapple with the agony of remembering or memorizing multiple authentication credentials for various service providers which may lead to insecure practice of writing down the credentials or using one credential for many websites. Over the years, these challenges have necessitated the development of various internet authentication frameworks and specifications to help developers to design tailored solutions for secure and efficient service delivery in the cyberspace. Notable among these specifications are Web Services (WS-*), Liberty Alliance and Oasis.

2.3 Web Services (WS-*) Specifications

In April 2002, IBM Corporation and Microsoft Corporation proposed a standard for security within a Web service environment that defines a comprehensive Web service security model that supports, integrates, and unifies several security models, mechanisms, and technologies in a way that enables a variety of systems to securely interoperate in a platform-independent and language-neutral manner of linking applications within organizations, across enterprises, and across the internet (12). The collaborations objectives is to design a framework and standard-based architecture for developers to build secure and inter-operable web services infrastructure in a distributed and heterogeneous cyberspace environment that would ensure safe and efficient B2C and B2B communications in the internet while ensuring the integrity, confidentiality and reliability through the application of an elaborate security model. The model is designed in a modular form encapsulated into a broad set of specifications – Web Services family (WS-*) that consists mainly of WS-Security, WS-Policy, and WS-Trust that forms layers of the model. Other layers such WS-Privacy are incorporated into these three basic specifications. The federation provides benefits of single sign-on among the federating units by providing a flexible mechanism to authenticate users from federating partner organizations while ensuring uniform common policies and mechanism across disparate local identity systems (11).

2.3.1 WS-Security Specification

This specification is the plank by which other layers of the “WS-*” architecture is developed. It defines the basic mechanisms for integrity, confidentiality and authentication of messages exchanged in the web service environment through the use of tokens. *Specifically, the WS-Security profile specifications describes how to encode username tokens, X.509 (PKI certificates) tokens, SAML tokens, REL tokens, and Kerberos tokens as well as how to include opaque encrypted keys as a sample of different binary token types* (13) . This specification provides developers with a platform for SOAP functionality to be deployed in designing web services application. SOAP (Simple Object Access Protocol) is an application layer protocol in the Internet Protocol Suite that is responsible for the exchange of structured messages in accordance with XML (Extensible Markup Language) format within a web services environment.

2.3.2 WS-Policy Specification

The web services policy framework provides a general purpose model and corresponding language to describe policies of federating entities in a web services environment (14). This specification defines a standard guidelines and procedures for messages between a sender and a receiver in a web services environment and such specification includes some basic parameters such as authentication schemes, transport protocol selection, and privacy policy. The WS-policy provides a single policy

language to streamline the exchange of messages in a heterogeneous web service environment in a consistent and efficient manner.

2.3.3 WS-Trust Specification

Trust is the foundation of any security model, and trust can be defined as *the expression between parties that one party to a relationship will believe statements (claims) made by another party; it is based on evidence – history, experience, documents etc., - and personal risk tolerance* (15). This layer of WS-* family builds on the foundation provided by the WS-Security mechanism framework and defines additional security mechanisms and extensions for the exchange of security credentials among federating units by providing standards for issuing, renewing, cancelling and validating security credentials as well as procedures to establish and broker trust relationships (16). In other words, it is required that an incoming message prove a set of claims such as name, key, permission or capability, otherwise if a message arrives without having the required proof of claim, then the recipient will not be able to recognize the origin of such message, and would therefore ignore or reject it.

2.3.4 WS Federation

A federation is a business-to-business relationship that has been established to share information securely based on trust and agreed security mechanisms. Thus the mechanisms defined in WS-Security, WS-Security Policy and WS-Trust as discussed above provides the foundation and model for WS-Federation. The value of establishing a federation is to facilitate single sign-on of users across the federating members. The federating members are the resource (service) providers and each federating unit is established by an identity provider. WS-Federation does not restrict users to a specific authentication token format, but instead builds on the WS-Trust encapsulation mechanism that allows protocol processing to remain independent of the type of token being transmitted, and this enhances the interoperability between the federating entities (15) . The WS-Federation provides the framework for inter-federation relationship such that the security token of a user can be federated outside a federated unit to grant users access to resources from other federating unit.

2.4 Liberty Alliance Project

Liberty Alliance was founded in September 2001 with the goal of establishing open technical specifications, standards, guidelines and best practices for internet identity management in order to drive a new level of security, trust and privacy requirements in

consumer-to-business and business-to-business online communications – it currently has membership of over 150 corporate organizations, institutions and governments across the globe (17). As obtained in WS-* specifications above, the Liberty Alliance is built around the concept of Federated Network Identity model that offers variable approaches for establishing standardized, platform-independent, web-based single sign-on with simple federated identities which gave birth to Liberty Identity Federation (ID-FF) version 1.0. Other follow-up specifications and guidelines are Identity Web services (ID-WSF), Liberty Interoperable certification program, an open source openliberty.org, Identity governance framework, Liberty Identity Assurance framework (IAF), and Concordia project amongst others. We shall discuss only those specifications that are relevant to our studies, while interested readers can consult the references for those specifications not covered here.

2.4.1 Liberty Identity Federation (ID-FF)

ID-FF specification offers a variable approach that is platform independent for implementing a single sign-on with federated identities based on a modular model with due consideration to the security and privacy of user's identity credentials. The users are empowered by privacy policy of the federation to be in control of their identity credentials, and other such attributes such as shopping habits and preferences to be shared with any SP as the user wishes. In ID-FF, the specification supports two or more federated groups that have

business relationships to “re-federate” to form what Liberty referred to as the *“circle of trust”*, provided such relationship is predicated upon prior agreement between the identity (authentication) providers (AP) and the SP on one hand, and provided too that the user’s consent is equally obtained. The key liberty objectives include (18):

- Enable users to have control over their internet identity credentials.
- Enable service providers to specify their domain security and privacy policies without third-party participation.
- Provides an open single sign-on standard based on federated network identity framework.
- Create a platform-independent network identity infrastructure that supports all current and emerging network access devices.

2.4.2 Liberty Identity Web Services (ID-WSF)

The Liberty Identity Web Services specification provides a framework for the exchange of structured and standardized messages between an SP and another in an inter-domain communication or between an SP and a client without compromising security. In order for a standard message to be passed between two domains, it is important that they agree on a standard communication policy such that the origin of the message can be verified at the recipient’s end; it is equally necessary to be able to ascertain whether the message is protected against eavesdropping or message

integrity. The overall objective is to establish trust between communicating parties by handling messages in accordance with agreed policy language. Although Liberty WSF does not specify a particular policy language, but it provides a number of places where policy may both be specified and enforced (19). In Liberty WSF, its web services are generally classified into three:

- Identity-based – specifies the requirements for consumer-to-business communication, in particular the policy guiding the authentication credentials of users.
- Identity-consuming – specifies the format for messages that are delivered to users where the SP needs to retrieve the users information – for instance via a subscribers database, for the purpose of delivering a service.
- Basic – specifies the structure or language of messages in either B2C or B2B communications e.g., SOAP envelope or TLS/SSL.

2.4.3 Liberty Identity Governance Framework (IGF)

This framework is concerned with the requirements for developers, implementers and information owners on how identity-related information should be stored, retrieved, updated, transmitted and the use of such information. Identity-related data is any information such as name, address, date of birth, medical history etc., that relates to an individual, device, or application that may perform actions or may be acted upon in a

system or service (20). The Liberty IGF specifies a framework for the access control policy of identity-related information of individuals held in a database with a view to protecting the data from unauthorized access, unauthorized disclosure, and unauthorized modification, and to establish a standard procedure that will enhance appropriate documentation and auditing of controls. The IGF is specifically designed amongst other things to support (21):

- Developers to build access control applications for identity-related data from a wide range of sources.
- Administrators and implementers to define, enforce, and audit policies concerning the privacy of data.

The IGF specification consists of four parts which include (22):

- CARM (Clients Attribute Requirement Markup Language) – which defines identity requirements such as the type of identity-related data required by applications and the application usage policy of the data.
- AAPML (Attribute Authority Policy Markup Language) – which defines the access rights or security assertions and any other use policies of the data.
- IGF Enabled Protocols – specifies the protocol mechanisms that link application to database.
- Developer APIs/Tools – developers can establish their identity requirements using any development tool of their choice.

2.4.4 Liberty Identity Assurance Framework (IAF)

The Identity Assurance is a framework that is established by Liberty Alliance to act as a benchmark by which identity-related applications can be assessed and evaluated in terms of their security policies, guidelines and privacy of identity credentials. The objective is to build confidence and encourage the formation of trusted identity federation, best practice, and to promote uniformity and interoperability amongst identity providers, with a specific focus on specifying criteria for issuing assurance level associated with identity credentials (23). The framework provides for cross certification and accreditation of service providers and identity providers to provide a baseline for trust, confidence and validation of identity federations while at the same time streamlining the certification process for online identity communities.

2.5 OASIS Standards

OASIS (Organization of Advanced Structured Information Standards) is a global non-profit consortium of various organizations and individual members founded in 1993 under the name of SGML (Standard Generalized Markup Language), and later changed its name to OASIS in 1998 (24). Oasis, just like WS-* family and Liberty Alliance was established to standardize framework for the development, collaboration, convergence, best practice and adoption of open standards in the IT industry for web services, web

security, e-business and e-governance. The core OASIS standards include – SGML, XML, Schemas, XSL/XSLT/XPath, XLink, XML Query, CSS and SVG.

2.5.1 SGML

SGML (ISO 8879(E)) is one of the generic markup languages in web services standard for the definition of device-independent, system-independent procedure of representing texts in electronic form (25). SGML has been replaced by other markup languages such as HTML and XML for producing web pages and various web services in the World Wide Web.

2.5.2 XML

The Extensible Markup Language (XML) is a subset of SGML and enables the generic SGML command to be issued, transmitted, received, and processed on the Web, and it has been designed for ease of implementation and interoperability with HTML (26). Thus XML standard is the vehicle by which developers create web pages/content and enables structured messages to be transmitted within the internet. For instance, SAML can be used to define a security token that contain security assertions based on XML.

2.6 OpenID Specifications

OpenID is an open, decentralized, free framework for user-centric digital identity that was established in 2005 by an open source community to provide necessary infrastructure for single sign-on and assist in the promotion and support for expanded adoption of OpenID (27). OpenID specifications include:

- **OpenID Authentication 2.0** – provides a base service to enable users have control over their authentication credentials in a free and decentralized manner without the Service Providers (SP) or Relying Party requiring access to end-users authentication token such as email and passwords. OpenID is not tied to the use of cookies but uses only HTTPs requests and responses, so it does not require any special capabilities of the client browser or other special software installation (28).
- **OpenID Provider Authentication Policy Extensionn1.0** – this extension may be used with the OpenID authentication specification and it provides a framework that enables the Relying Party to request the OpenID Provider to implement additional local authentication policies when authenticating users. Such authentication policies may be multi-factor authentication or phishing-resistant method (29). The assumption here is that the Relying Party (RP) must trust the OpenID Provider (OP) to implement the additional policies and the assurance level need to be communicated to the RP.

Other OpenID specifications are **OpenID Attribute Exchange 1.0** – that defines the framework for exchanging identity information between endpoints (30) and, **OpenID Simple Registration Extension 1.0** – that produces a standardized reference list of user's identity attributes that may be required by various RPs during user's registration of new account with OP. Examples of OpenID providers are myopenID, verisignlabs, claimed, myID.net, and myvidoop.

2.7 SSO Applications

So far in the preceding sections of this chapter, we have dwelt on the three major Web Services standards – WS-*, Liberty Alliance, and OASIS. Although we have so many SSO applications today – both proprietary and open source, however we shall be restricting our discussion here to a common example of SSO application - .NET Passport. Other examples of SSO are OpenSSO offered by Sun Microsystems, SecureLogin by Novell, and SP Sign-On by Unisys.

2.7.1 Microsoft .NET Passport

.NET Passport is a user authentication technology that is based on the WS-* standard and MS .NET framework that uses web security mechanisms such as SSL, HTTP, cookies and cryptographic primitives to offer single sign-on solutions to all users of Microsoft online services such Hotmail, SharePoint Online, Office Communication

Online etc. Other service providers equally federate with Microsoft identity service to form Microsoft identity federation which has recently been transformed to the Microsoft “identity metasystem” – the Windows Live ID Service. Unlike the .NET passport authentication mechanism that uses only email and username/password pair of authentication token, the Windows Live IDs is based on Windows Live ID accounts which can be authenticated using traditional username/password pairs, strong passwords and security PIN combinations, and smart cards , and also supports the use of self-issued Windows CardSpace formerly “infocards” (31).

2.8 SIM Security Services

SIM is a smart card that is tamper-resistance with embedded integrated electronic circuit or chip which has the capability for storing, processing and transmitting data (8). When a smart card is classified according to the type of chip used then we have two types – memory chip and microprocessor chip cards. On the other hand, if we classify smart cards according to data transmission methods, then we have three types – contacts, contactless and dual interface cards. The SIM (Subscriber Identity Module) falls under the microprocessor type of chip and it has contact interface. In accordance with ISO 7816-1,2, a SIM card generally has two sizes and layout – the ID-1 SIM which is large (just like the size of a standard ID card), and the Plug-in SIM which has a width of 25 mm, and a height of 15 mm (32). Our emphasis here is on the Plug-in type, which

is inserted into the mobile equipment (ME) together which can be regarded as Mobile Station (MS). The SIM is issued to subscribers by mobile (network) operators and it is independent on type of mobile equipment. The security goal of the SIM may be summarized as follows:

- Authenticate the subscribers (users) to the network by protecting the network from unauthorized use such as cloning, and providing appropriate billing information.
- Provides for users anonymity by protecting against the location of users during call.
- Provides for the confidentiality and integrity of users data during communication on the radio link between the mobile station and the base station.

2.8.1 SIM Authentication

The SIM authentication architecture is based on a long-term secret key, K_i, a unique subscriber identifier, IMSI (Integrated Mobile Subscriber Identity), authentication algorithm, A3, and cipher algorithm, A8 (33). These items are required to be present at both the network authentication centre (AuC) and the SIM for authentication to take place. The authentication of the SIM to the network is required whenever the mobile equipment (ME) housing the SIM is powered-on or whenever the subscriber out of his

home location register (HLR) to the vicinity of a visitor location register (VLR) particularly in a roaming situation.

When the ME is powered-on, the SIM requests for the subscribers PIN (personal identification number) to unlock the SIM if the SIM is PIN-locked and thereafter it retrieves the IMSI and forward it through the ME to the Base Station for onward transmission through the HLR to the AuC. The IMSI is a unique identifier that maps uniquely to the MSISDN and it is assigned by the network operator to disguise the MSISDN as part of the user anonymity security service offered by GSM operators. However, the IMSI is not always transmitted to VLR during roaming but instead a temporary IMSI known as TMSI is generated and transmitted to VLRs for authentication of subscribers. When a SIM requests for authentication either when a ME is powered-on or during a call, the AuC retrieves the secret K_i and generates a random challenge, RAND and together used to produce a XRES (expected response) with the A3 algorithm, and K_c with the A8 algorithm which together with the RAND form the authentication vector in triplet which is forwarded to the HLR. The HLR stores the XRES and the K_c, and forwards the RAND to the SIM which is used by the SIM to produce its own SRES (subscriber response) and K_c. The SRES is then forwarded to the HLR for comparison with the earlier XRES produced by the AuC, if they match, then the SIM is authenticated to the network and the K_c (cipher key) is now used to cipher messages between the mobile station and the base station during a call to provide for message

confidentiality and integrity on the radio link. The AuC produces the authentication triplets in batches for use by the HLRs or VLRs to facilitate service delivery on the network. It is equally instructive to note that the K_i does not leave the AuC and the SIM to guide against cloning and other vulnerabilities that may be associated with the knowledge of the secret key.

2.8.2 3G/USIM Authentication

3G (3rd Generation)/Universal SIM (USIM) is a removable module like the SIM, but unlike the SIM which is a mono-application smart card, the USIM supports value-added services in the form of independent applications which can even run in parallel in a multi-application environment (33). Due to some vulnerability that was observed in GSM services using SIM, the USIM was developed to plug those vulnerabilities. As part of the enhancements, mutual authentication, MAC, anonymity key (AK), integrity key (IK), sequence number (SQN), and authentication management field (AMF) were all included in the authentication challenge. Unlike in the SIM, the authentication vector (AV = challenge) now consist of 5 elements (quintuplet): RAND|| XRES||CK||IK||AUTN; where the authentication token (AUTN) = SQN $\oplus$ AK||AMF||MAC – the authentication token (AUTN) support in addition to mutual authentication, management and anti-replay attacks (2).

2.9 Chapter Summary

We have tried in this chapter to review relevant literatures related to SSO applications on the internet. The issues surrounding Internet Identity Management – from security, privacy to user's convenience were discussed and the various challenges were equally elaborated. Various internet identity standards and specifications were treated – they include Microsoft WS-* specifications, Liberty Alliance Project specifications, OASIS, and the most recent OpenID specifications. As an example of common SSO application, we discussed Microsoft .NET Passport (Windows Live ID), and finally concluded the chapter with a peep into U(SIM) authentication and security services.

Our next chapter shall dwell on the application design, description of protocol flows and messages, as well as definition and description of use cases.

CHAPTER 3

DESIGN AND DESCRIPTION OF APPLICATION

In the previous chapter, we reviewed various web services standards and specifications, and have decided to base our design on the OpenID identity model, but instead of username/password-based authentication, it was replaced by a more secure and user-friendly SIM-based authentication. So one of our assumptions here is that our application shall be provided by a GSM operator such as Vodafone, because we want to leverage on the existing security infrastructures of GSM network (see 2.8), we therefore assume that every internet user that would access this service must have a GSM phone. Our use case in developing this application is user authentication in online banking.

Online banking, also known as internet banking, electronic banking, cyberbanking, virtual banking, or home banking, includes various banking activities conducted from home, business, and on the road as an alternative to the customer's physical presence at the bank branch (34). Online banking is developed to save time and money for users, while at the same time offering the banks an inexpensive alternative to branch banking with higher returns on investment. Some of the banking activities that customers can carry out using online banking include bank account

balances, account statements, bills payment, intra and interbank money transfers, and telephone banking amongst others. However, in many banks, the story have not been the same, as bank fraud caused by insider abuse and online bank criminal gangs have threatened this business innovation brought about by the emergence of the internet. Various bank frauds range from online identity theft through email scam, phishing, password guessing, dictionary attack on password, brute-force attack on password, and stealing of bank account details. According to an online banking fraud survey conducted by the BBC news (35), concluded that the UK has seen an 8,000% increase in online banking fraud; excerpt is given below:

> The UK has seen an 8,000% increase in fake internet banking scams in the past two years, the government's financial watchdog has warned. The Financial Services Authority (FSA) told peers it was "very concerned" about the growth in "phishing". Phishing involves using fake websites to lure people into revealing their bank account numbers.
>
> The amount stolen is still relatively small but it is set to go up by 90% for the second year running, peers heard. Between January and June 2005, the number of recorded phishing incidents was 312, the Lords science and technology committee was told. The figure for the same period this year was 5,059, according to banking trade body Apacs figures. (35)

In related publications in subsequent years, Gartner published another survey captioned, *"Gartner Survey Shows Phishing Attacks Escalated in 2007; More than $3 Billion Lost to These Attacks"* (36). Also, in 2008 there is another publication by the BBC news captioned, *"Phishing attacks soar in the UK"* (37). It is therefore obvious that incidents of online banking fraud have continued to be on the increase resulting to loss of millions of investor's money. However, despite the increases in the incidents of online banking fraud, the banks and the customers still embrace this technology because of its enormous benefits as compared to the traditional and unwieldy branch banking. Although, various solutions have been pouring out over the years to attack this phenomenal fraud, the situation still appears bleak as fraudsters are equally advancing in their act everyday to outwit developers.

Having identified one of the main vulnerability in most online authentication solutions, we have decided to develop an application that will replace the username/password-based authentication with what the user possesses – the SIM. Therefore, our main objective in this use case is to prevent phishing attack, and to replace username/password authentication which is prone to password guessing, dictionary attack, or brute-force attack. We equally wish to make internet banking experience to be devoid of its user frustrations due to forgotten passwords and the resultant effect of insecure practice of storing passwords.

For the purpose of this paper, our design architecture shall be limited to front-end design consisting of the major actors/entities, and the exchange of messages. Thus this chapter, is structured into three parts – system architecture, message flows representation and actions description to describe the interactions between various objects, and finally the security considerations.

3.1 System Architecture

Our system architecture is broken down into three modules – user registration mechanism, sign-in mechanism and single sign-on mechanism. The user registration module is for users who are registering with a service provider for the first time in order to access the SP's protected service; the sign-in module is for existing users of the openID internet service, while the last module – single sign-on is for users already authenticated by the identity provider (IDP) with valid session identity and wish to access other service providers (SPs) within that authentication session. The system architecture is followed by two interaction diagrams – a simple challenge/response mechanism and sequence diagrams to describe the sequence of operations during each process.

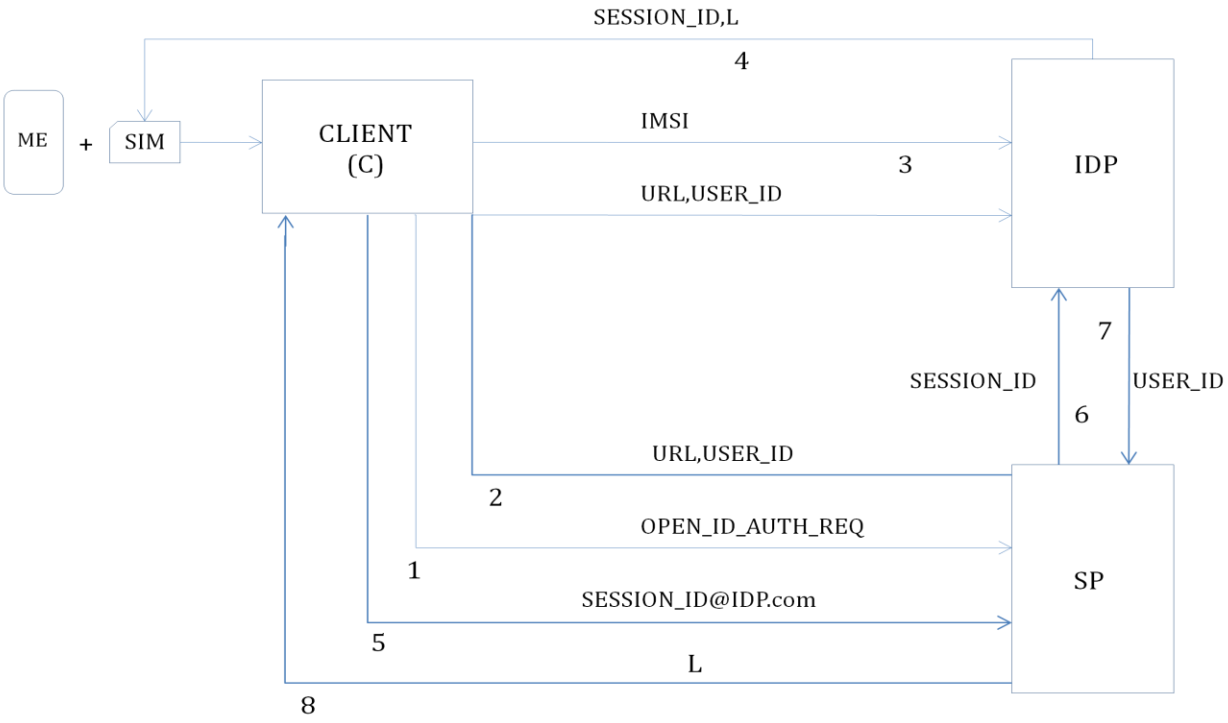


Figure 1: User registration mechanism

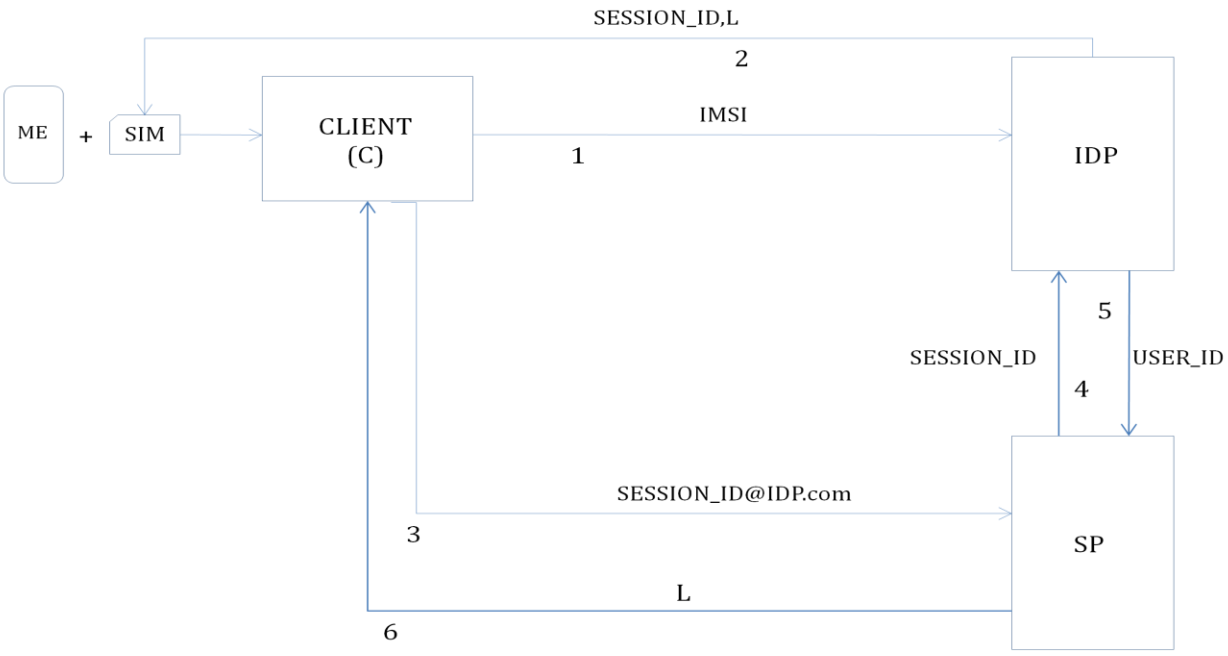


Figure 2: Sign-in mechanism

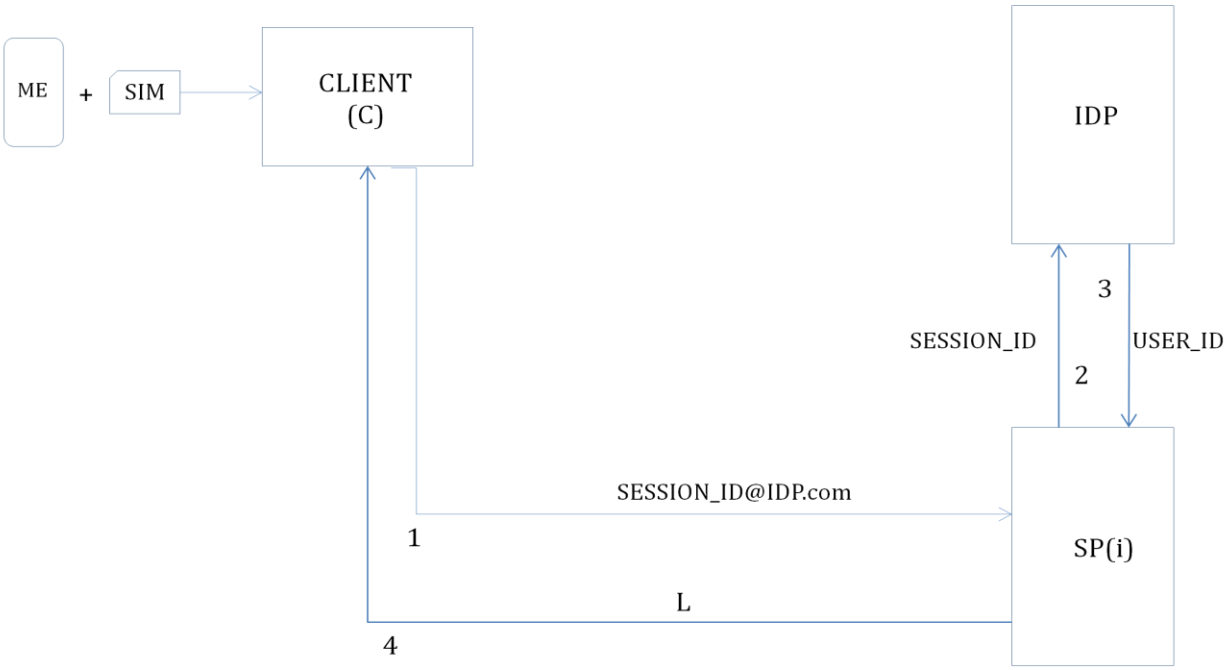


Figure 3: Single sign-on mechanism

3.1.1 Entities Definition in System Architecture

The entities/actors involved in the systems architecture as contained in figures 1 to 3 are briefly described below:

- ME (Mobile Equipment) – refers to the mobile phone handset.
- SIM (Subscriber Identity Module) – is the tamper-resistant smart card issued by the GSM operators to the users and it is inserted into the mobile equipment (see 2.8). It contains the IMSI, authentication algorithms, secret key, and other basic functionality and features that ensure that mobile services are offered to the subscriber securely and efficiently.
- Client (C) – is the user's browser running on a PC, laptop or PDA.

- IDP (Identity Provider) – is a server that offers the authentication service and it includes back-end facilities such as user's database.
- SP (Service Provider) – these are the business websites where various protected services are being offered on the internet. In this paper, Service Provider can equally be referred to as Relying Party according to OpenID specifications but we shall use SP throughout.
- OPENID_AUTH_REQ – refers to openID authentication request issued by the user for openID authentication.
- URL (Uniform Resource Locator) – is the website address of the SP.
- USER_ID – refers to user's unique registered customer identity and it is generated by the SP which is required for customer identification and audit purposes.
- IMSI (International Mobile Subscriber Identity) – is a unique identifier for the SIM which is issued by the GSM network operator and stored in the SIM. It is used during SIM authentication to identify the SIM and it equally gives users a form of anonymity during communication as only the network operator can associate the IMSI with the SIM. The IMSI usually contain about 15 digits long – with the first 3 representing the Mobile Country Code (MCC), either next 2 or 3 representing the Mobile Network Code (MNC) and the last digits representing the Mobile Subscriber Identification Number (MSIN) (32).

- SESSION_ID – is a unique session identity issued by the IDP to the user during registration and sign-in process and it serves as a kind of ticket within the session to access services at various SP's belonging to a particular openID network. It equally acts as a form of a dynamic password or one-time-password (OTP) since it is only valid for one authentication session.
- L (liveness) – is the time to live for the SESSION_ID and the SESSION_ID expires once L expires.
- SESSION_ID@IDP.com – is formulated in accordance with OpenID framework but it replaces the static "user_opendID" in most OpenID implementations.
- SP(i) – refers to the i-th SP in a single sign-on process; where i=1,2,3,....,n.

3.2 Description of System Architecture

The challenge/response authentication mechanism involving the various objects in the system architecture is described below. We shall commence with the registration/enrolment process consisting of steps 1 – 8.

3.2.1 User Registration Process

The registration process consists of steps 1-8:

1. In order to register for online services, a user/customer requests for openID registration by sending the message – (openID_auth_req) to the SP; which in our use case is a bank. We shall use user or customer or client to mean the same in our application. However, a client here can equally be the browser of the user's PC or laptop.
2. Here the SP processes the request in (1) above and then issue a unique customer internet identity – User_ID for the user/customer. This User_ID shall be associated with the user/customer details at the end of a successful registration for the online banking service. This measure is being introduced as part of the user anonymity security service as only the bank can be able to associate this user data/attribute to his/her account details. At the successful creation of the User_ID, it is forwarded together with its URL in a re-direct link through the client to the Identity Provider (IDP).
3. At the same time, (2) above is forwarded to the IDP via the client's browser, the client is prompted to enter its unique user identifier – in our design, we have chosen this unique identifier to be the IMSI of the user's SIM. As you already know (see 3.1.1), the IMSI plays the same role in SIM authentication in GSM networks. The only question here is how the client's browser would read this unique identity to be forwarded to the IDP. We shall propose three methods – via Bluetooth, NFC's enabled phone or mobile phone with WAP (Wireless Access

Protocol) services. We shall elaborate on these technologies later in the next chapter. Once the IDP receives the IMSI, it is used to retrieve the user's MSISDN from the IDP's HLR/AuC, and these four attributes – IMSI, MSISDN, URL, and User_ID are stored in the IDP's database which shall be used for the authentication of the user whenever the user requires an online banking service.

4. However after (3) above, a particular user can have as many additional attributes as may be desired by the user which can be used for discovery service by the IDP. It is important to emphasize here that apart from the compulsory 4 attributes and additional Session_ID which we shall discuss here, every other attributes submitted by the user is at its own discretion. For instance a user may decide to submit its credit/debit card details for online payment. The purpose for this flexibility in our design is to put the control of user's data at their grip to satisfy privacy issues of user's authentication credentials, (see 1.1) for user's privacy concerns. In this line 4, upon the receipt of the 3 attributes and the subsequent retrieval of the user's MSISDN, a session identity together with attached live time – (Session_ID,L) is forwarded to the user's phone via his MSISDN. The session_ID occupies a temporary field location in the user's database with the IDP and it expires once L expires. At this point the user is now allowed into the openID members directory of the IDP where an http link to all the members of the IDP openID are displayed using a drop down list.

5. Upon registration/enrolment, and having obtained a session_ID from the IDP, the user now selects his/her bank's URL's link and is taken back to the authentication page of the bank (SP). At this web page, the user is required to submit its Session_ID for verification and equally for the bank to obtain the user's internet identity from the IDP, because in the openID network, the SP only knows the user by this User_ID and this is the user's attribute that it shares with the IDP. It may be necessary to emphasise here that the Session_ID is concatenated with the URL – Session_ID@ IDP.com, in the form of email address; this is to inform the SP that the user is being authenticated by the that particular IDP openID, it therefore creates room for the SP to belong to as many OpenID authentication federation as may be desired without any conflict.
6. Upon receipt of the user's Session_ID@IDP.com, the SP then extracts the session_ID and forwards it to the IDP for two reasons – to obtain the user's internet identity – User_ID, and as a further security layer built onto the authentication process. Just like in the physical world where a traveller presents his/her passport/visa at the port of entry, the immigration officer still has to submit the document for further verification before the visitor is allowed to pass – this is our thinking.
7. In step 7, the IDP receives the Session_ID and uses it to locate the temporary field location and with it, is able to retrieve the User_ID corresponding to the

Session_ID and forwards it to the SP. Another reason why the application is designed in this form is to equally give the SP a role to play in the authentication process, so that the SP would have the final decision whether to allow the user or not. In a way, it provides a form of check and balances in the authentication process.

8. Finally, the SP receives the confirmation in the form of the User_ID because it is only the IDP that can associate the current Session_ID with the User_ID, so that the user is now granted access to carry out his/her online banking activities and the L attached to the user's session_ID indicates that the session has a livenes of L.

3.2.2 User Log-in Process

From the system architecture diagram in Figure 2 and the exchange of messages in the registration process above, observe that the architecture in Figure 1 and Figure 2 are the same except for steps 1 and 2 of the registration process which is removed from the sign-in process. Every other step remains the same as explained in the registration process above.

3.2.3 Single Sign-on Process

In 3.2.1 and 3.2.2, we have explained the user registration and sign-in processes assuming the user intends to access only the services of one SP. Now in another scenario, the user may desire to access other SP during his/her authentication session with the Session_ID at his kitty, all that is required is for the user to present this session identity to his new SP for verification so long the user has a valid registration with the new SP, say SP(i). Once the registration is ascertained, steps 1 to 4 of the registration process is circumvented and the user goes straight to step 5 to submit his/her session identity – Session_ID@IDP.com to the SP(i). The entire process now passes through steps 4 to 8 as described above and the user is successfully logon to the services of the new service provider, SP(i).

3.2.4 De-registration Process

We equally found it expedient to inform that should the user desire not to continue with the services of a particular SP, he is at liberty to do so with ease. All he requires is to activate a de-registration button and the de-activation request will be passed to the IDP together with his unique identity – IMSI, the IDP then sends the user a de-activation confirmation request via his mobile phone, this confirmation is submitted at his/her SP, the SP verifies and the user is finally de-listed from the user database of the particular SP.

3.3 Challenge/Response Mechanism

The challenge/response mechanism is split into three – user registration/enrolment process, user log-in process, and single sign-on process:

1. C → SP : OPEN_ID_AUTH_REQ
2. SP → IDP : URL,USER_ID
3. C → IDP : IMSI
4. IDP → C : SESSION_ID,L
5. C → SP : SESSION_ID@IDP.com
6. SP → IDP : SESSION_ID
7. IDP → SP : USER_ID
8. SP → C : L

Figure 4: User registration challenge/response mechanism

1. C → IDP : IMSI
2. IDP → C : SESSION_ID,L
3. C → SP : SESSION_ID@IDP.com
4. SP → IDP : SESSION_ID

5. IDP → USER_ID

6. SP → C : L

Figure 5: Sign-in challenge/response mechanism

1. C → SP : SESSION_ID@IDP.com

2. SP → IDP : SESSION_ID

3. IDP → SP(i) : USER_ID

4. SP → C : L

Figure 6: Single Sign-on challenge/response mechanism

3.4 Sequence Diagram

We equally have three sequence diagrams representation corresponding to each stage of the authentication system as contained in the system architecture and challenge/response mechanisms above. The sequence diagram displays the objects, the actions performed by each object and the object that requests for such action. From the system architecture above, there are basically 5 objects/entities as contained in the boxes but the SIM and mobile equipment together form the mobile station. The direction of the arrows indicates the direction of flow of messages, while the vertical dotted lines (----) represents events at each entity.

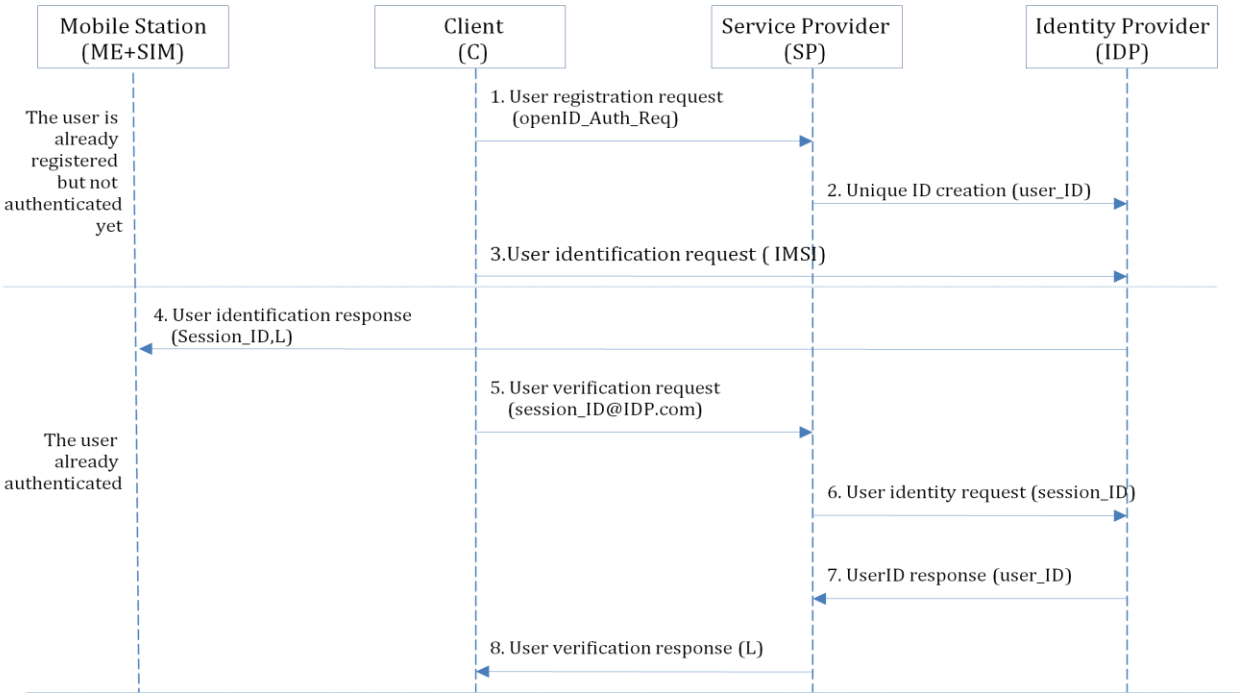


Figure 7: Object interactions in a registration/enrollment process

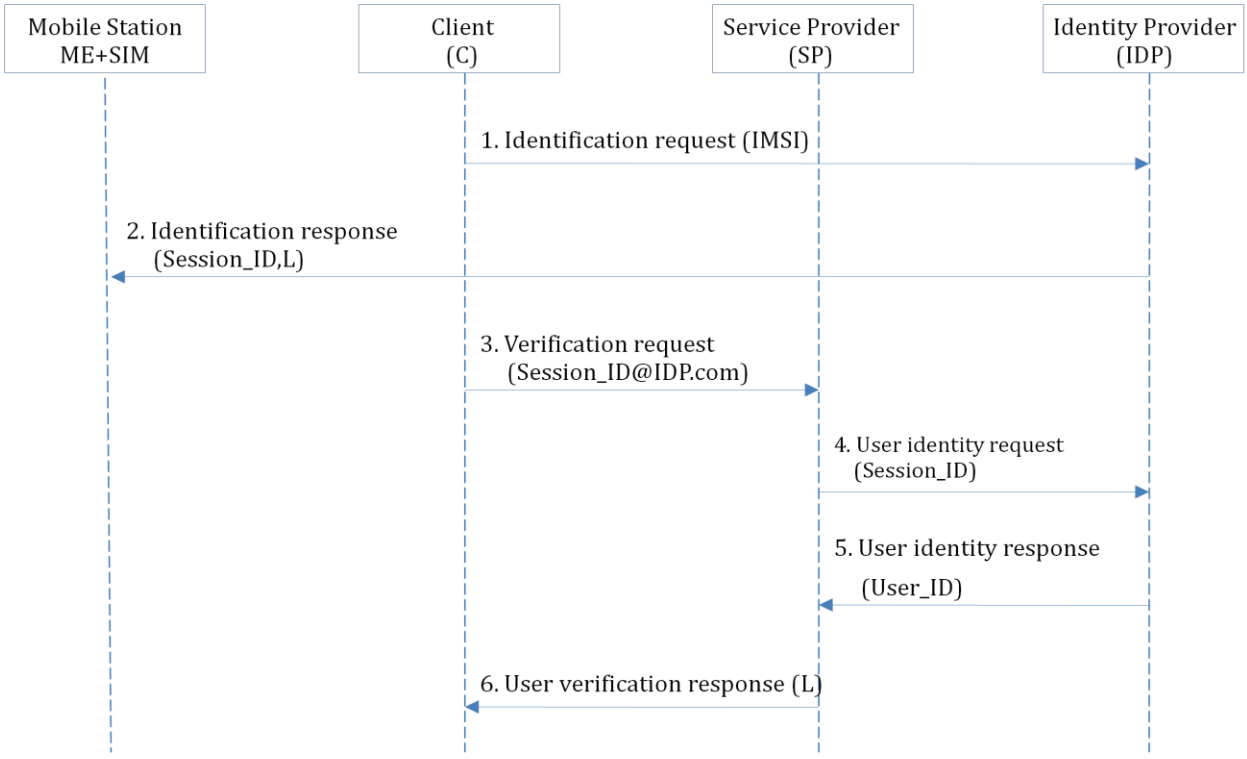


Figure 8: Object interaction in a sign-in process

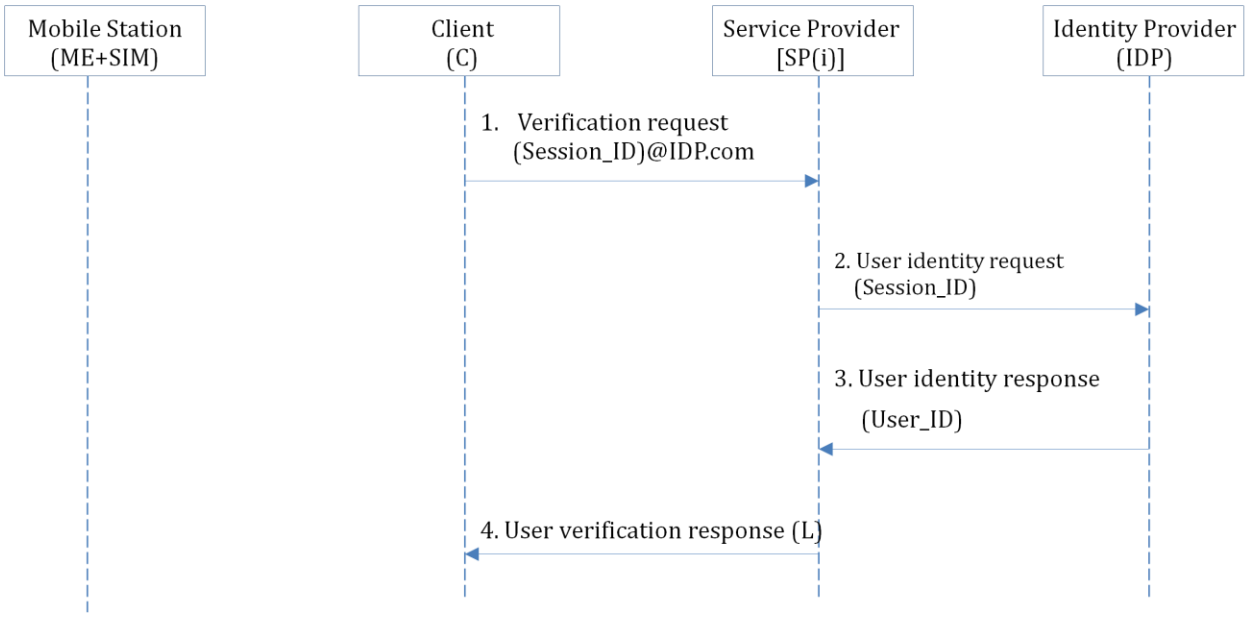


Figure 9: Object interactions in a single sign-on process

3.5 Security Risk Evaluations

In this section, we shall be considering the potential vulnerabilities and perceived threat to enable us determine the risk associated with the operability of the application. Our methodology shall be to examine the application using a step-wise approach to enable us identify possible weakness, threats and impact if any, in a particular operation or message path, and to determine appropriate security mechanism that could be applied to prevent an attacker from exploiting any such vulnerability. In this wise, we shall use the challenge/response mechanism to isolate each particular operation or exchange of message/authentication path to enable us carry out the potential risk assessment and treatment.

1. C → SP : openID_Auth_Req

   Risk assessment: This is just a request made by the client for openID registration option, so an attacker has nothing to learn and so, no any form of identifiable risk associated with this interaction.

2. SP → IDP : URL,USER_ID

   Risk assessment: From Figure 1 above, observe that this is a re-direct link through the browser of the client to the IDP. The reason for this re-direct is to enable the client's browser to query the client's mobile phone for the entry of a third field, the SIM's unique identity – IMSI, that the IDP requires for the user's identification. A lot of attacks can take place here, but what would the attacker learn. Well, the attacker may want to masquerade as the user in the future – man-in-the-middle attack. From our analysis, we want to assume that this path could be vulnerable to two basic attacks – eavesdropping and replay attacks.

   Risk treatment:

   (i) To secure the path between SP – C and C – IDP with SSL, this will prevent eavesdropping and replay attacks. SSL/TLS (Secure Socket Layer now known as Transport Layer Security) is a communication protocol in the application layer that is used to secure the end-end communication link between two points over the internet. It provides data integrity and confidentiality of messages over the networks.

(ii) On the alternative, the SP can encrypt the message with the IDP's public key such as RSA – this will prevent eavesdropping but not replay attack. So, to prevent a replay attack, a time stamp (T) need to be added to the message before encryption, which can be re-written:

$$\{URL,USER\_ID,T\}_{Kidp}.$$

where Kidp is the public key of the IDP.

3. C → IDP : IMSI

Risk Assessment: The IMSI is a sensitive data in SIM authentication in GSM network, and so it has to be protected first from eavesdropping attack. Another vulnerability that is of concern is the path between the phone and the PC/laptop. Two attacks can take place here resulting from man-in-the-middle attack – replay attack, and the other – injection of another IMSI by an attacker.

Risk treatment: The only attack that may result to any impact in the above scenario is the eavesdropping because of the sensitivity to GSM network. However, this can be prevented by encrypting the IMSI with the public key of the IDP. The other two attacks – IMSI injection and replay attacks has already been made ineffective with encryption, however, other security measures have equally been incorporated in the application between lines 4 to 7 that nullifies these attacks. Thus, incorporating this control measure into (3), we now have a revised form as:

$$C \rightarrow IDP : \{IMSI\}_{Kidp}$$

4. IDP → C : SESSION_ID,L

   Risk assessment: The session_ID is one of the most sensitive user credential in the design, if an attacker can get hold of it, then the attacker can masquerade as the user within the live-span of that session (L). This identity attribute will be sent to the user's phone via short message service (SMS).

   Risk treatment: Since we are leveraging on the security of the GSM's network, then this message is secure because it is encrypted by the OTA (Over the Air) server of IDP to the SIM using their cipher key, i.e.,

$$IDP \rightarrow C : E_{kc}(SESSION\_ID,L)$$

5. C → SP : SESSION_ID@IDP.com

   Risk assessment: The path between C and SP can be vulnerable to man-in-the-middle attack, such that the attacker can masquerade as the user within the live-time (L) of the session_ID.

   Risk treatment: We have earlier suggested that this path be protected by SSL, so this is equally the best panacea to this threat.

6. SP → IDP : SESSION_ID

   Risk assessment: Just as we discussed in 5 above, this path is susceptible to man-in-the-middle attack, so an attacker must not get hold of the session_ID.

Risk treatment: In addition to the SSL channel between the SP and IDP, the SP can encrypt the message with the public key of the IDP which we suggested in message 2 above. We now have;

$$SP \rightarrow IDP : \{SESSION\_ID\}_{Kidp}$$

7. IDP → SP : USER_ID

   Risk assessment: As explained earlier, the USER_ID is the identity attribute that the SP uses in tracking the customer that is shared with the IDP. If an attacker gets hold of this attribute without the session_ID, it is meaningless.

   Risk treatment: The above assessment notwithstanding, we already suggested in 7 above that this path be supported by SSL, in addition the IDP can encrypt the USSER_ID with the public key of the SP such that the revised flow is:

$$IDP \rightarrow SP : \{USER\_ID\}_{Ksp}$$

8. SP → C : L

   Risk assessment: As discussed in 3.1.1 above, L is just the session period assigned to the user and therefore carries no message that would be useful to an attacker. The session expires as soon as the condition attached to it occurs, such as inactivity over a period of say 10 minutes the session can expire.

Based on the security risk evaluations, a revised form of the challenge/response mechanism – figure 4 is displayed below. The other scenarios as contained in figures 5 and 6 follow as well:

1. C → SP : OPEN_ID_AUTH_REQ

2. SP → IDP : $\{URL,USER\_ID,T\}_{Kidp}$

3. C → IDP : $\{IMSI\}_{Kidp}$

4. IDP → C : $E_{kc}(SESSION\_ID,L)$

5. C → SP : SESSION_ID@IDP.com

6. SP → IDP : $\{SESSION\_ID\}_{Kidp}$

7. IDP → SP : $\{USER\_ID\}_{Ksp}$

8. SP → C : L

Figure 10: Revised challenge/response mechanism

On a general note, it is obvious that a panacea to most identifiable risks in the design can be mitigated by configuring the IDP and the SP servers to use SSL channel during the authentication session. We equally want to re-emphasise that our main design objective is to prevent phishing attacks in online banking, and we believe that we have achieved this goal together with user convenience – no more username and password associated problems. Apart from the username/password replacement which is obvious, the reader may want to ask how phishing attack has been prevented in our application. Simple; recall that from the excerpts that we quoted at the introduction of this chapter, we explained the antics of attackers in phishing attacks. One style is by email scam – requesting the user/customer to renew his/her account details and if the

user clicks on the requested link, he is directed to a website that appears the same as the original website of the bank. There, the user's details will now be stolen. Even in another style, when a user issues an http request using the URL of the bank, the user could be re-directed to another website that equally appears the same to trick the user to enter its details. In both scenarios, it is obvious that our application has replaced the point of failure/vulnerability – username and password by a stronger authentication – the SIM. Suppose an attacker successfully tricked a user to its website, what information can he get from the user – the SIM unique identity – IMSI; and so what would be the usefulness of the IMSI to the attacker – no use! The attacker learns nothing from the IMSI of the user's SIM as he cannot associate it with his MSISDN, only the GSM network operator can – thus it has given the user a sort of anonymity as the attacker cannot associate the IMSI with the user. From our evaluation, the only way an attacker can succeed is when the user's phone is stolen, and even so, the user has to be advised to always pin-lock his SIM to give enough room for such incident to be reported so that the SIM can be blocked by the network operators.

3.6 Performance Evaluations

It is obvious that one of the most common hand-held devices that can be found on any individual today is the mobile phone – whether at home, in the office or on the highway – mobile phone is everywhere; so to perform authentication only requires the

presence of this phone which is always with you. Unlike the hardware security token issued by the banks such as RSA secureID requires a PIN and has to be carried about. Some of the numerous benefits of this design can be summarised below:

- no more multiple username and passwords;
- no more memorization of username and password;
- no more user frustrations due to fear of forgetting their password;
- no more insecure practices of storing passwords such as writing down password, storing password using browser's cookies, or storing passwords in computer files;
- no more multiple authentication token such as a two factor - username/password and hardware token;
- no more inconvenience of carrying hardware authentication token.

To the service providers, no more phishing attacks nightmares, in addition, the overheads due to intermittent reset of user's forgotten passwords would have been saved, also there would be no need for hardware authentication token with its attendant overhead costs to the banks. Above all, this design will restore the confidence of internet banking to online communities.

3.7 Chapter Summary

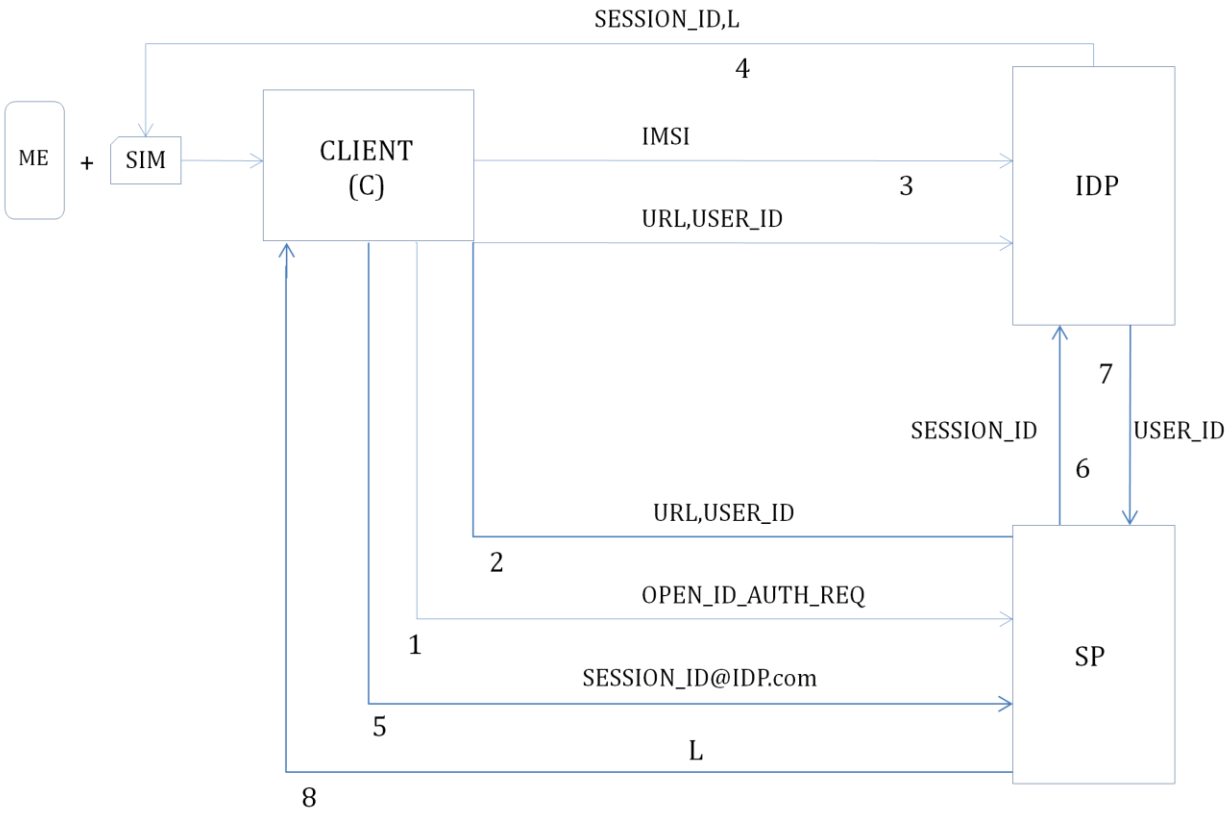
In this chapter, our emphasis is on the application architectures and the description of the interactions and exchange of messages between objects – client, service providers and the identity providers. Our design consists mainly of the front-end with message flow representation using challenge/response mechanisms and sequence diagrams. We concluded this chapter by carrying out the security and performance evaluations of the application.

In the next chapter, we shall give a proof-of-concept demonstration of the operability of our design.

CHAPTER 4

DEMONSTRATION OF PROOF-OF-CONCEPT

The use case in our demonstration is a secure customer authentication in online banking. The principal actors in our demo are the Client represented by the Customer, the Service Provider represented by Holloway Bank, and the OpenID Provider represented by Vodafone OpenID. The Mobile Station i.e., the SIM and mobile equipment, and the customer's PC/laptop shall altogether form the Client in our demonstration. Please note that the principal actors are all hypothetical cases for the purpose of this research paper. In this chapter, we shall be demonstrating the "implementability" and operability of the application using simple web services tools such as wamp server, notepad, HTML and JavaScript. Then on the user side, we need a mobile handset (ME) with either NFC-enabled SIM and NFC (Near Field Communication) reader or mobile phone with Bluetooth capability, or mobile phone with USB cable connector. Wamp server shall serve as our web server. In the demo, we shall go through all the protocol steps from 1 to 8 that encompass the three scenarios of registration, login and single sign-on. In order for us to give a proper understanding and follow-up, we shall re-display the system architecture as obtained in: Figure 1: User registration mechanism



Stage 1: A customer with its browser displays the web page of his/her bank using the bank's url: http://www.holloway_bank.com and the user is directed to the authentication page as displayed in the web page below. In our demonstration, the URL will be displayed by the wamp server as: file:///C:/wamp/www/holloway_bank/authentication.htm. In the Holloway Bank authentication web page, a customer is required to either register for online banking services to obtain an internet identity or login with his/her session_ID if he has already registered and has equally submitted himself for identification by the openID provider – Vodafone openID.

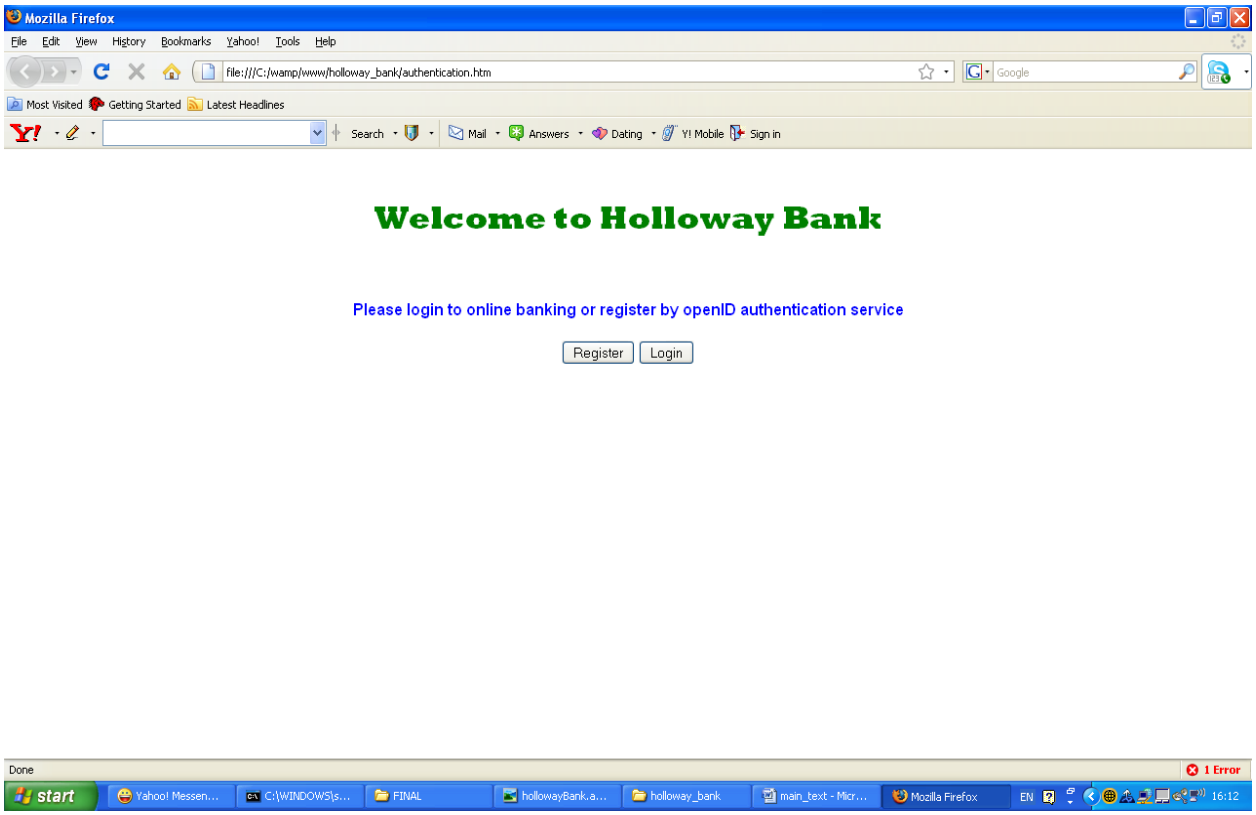


Figure 11: file:///C:/wamp/www/holloway_bank/authentication.htm

Observe from above figure that the customer is prompted to either register or login. For now, we want to assume that the customer wants to register, so he clicks on the registration button which is equivalent to step 1 in our protocol runs:

Customer → Holloway_bank : I want to register by openID

Upon the receipt of the registration request, Holloway_bank then assigns a unique customer internet identity – USER_ID which together with the URL is forwarded to Vodafone OpenID Service by redirecting the browser to Vodafone OpenID. Suppose the user_ID assigned to the customer is 74895361. The protocol runs appears thus:

Holloway_bank → Vodafone_openID : This is my URL and the customer's internet ID (74895361)

In other words, the two protocol runs in steps 1 and 2 takes place simultaneously as you can observe in the Figure 1 above. Meanwhile at Holloway bank server's end, the just created user_ID is held in a cookie pending the completion of the registration process so that the field can be added to the customer's record in Customer Database as the customer's internet identity.

Stage 2: At this stage, the customer is requested by the Vodafone OpenID to submit its unique SIM identity – the IMSI. Three methods can be used to achieve this – by NFC phone with a reader or laptop with NFC capability, Bluetooth or USB data cable. Alternatively, if the phone has WAP services or a PDA is used, then the operation can be run directly from the phone's WAP browser. The protocol runs for this stage is given as:

Vodafone_openID → Customer : Please provide your SIM's unique ID – IMSI

From the figure below and the corresponding protocol runs, Get SIM_ID is a query by the browser to the SIM once the link or phone is detected to provide the unique SIM identity – IMSI. An application plug-in is required to be installed at the

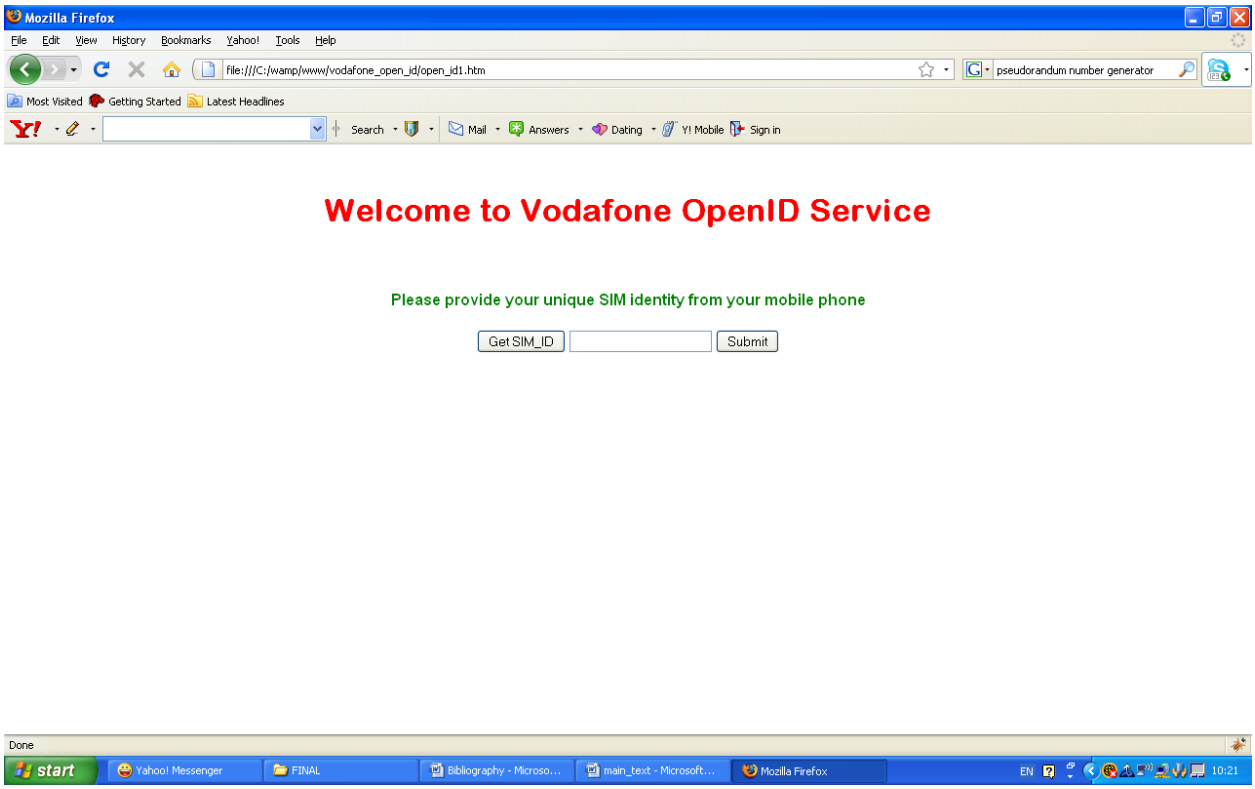


Figure 12: file:///C:/wamp/www/vodafone_open_id/open_id1.htm

Vodafone OpenID server, and on the SIM – an applet needs to be installed to respond to the APDU command from the phone to implement this request without compromising the SIM's security. APDU (Application Protocol Data Unit) is a communication unit between the SIM and the reader – in this case – the mobile equipment (38).

Stage 3: This operation follows immediately when the customer clicks on the Get SIM_ID button – the browser queries the phone and an APDU command is sent to the

SIM, the SIM then replies with an APDU response containing the IMSI and is displayed on the Get SIM_ID input field. The protocol runs is given below:

Customer → Vodafone openID : This my SIM's unique ID – IMSI

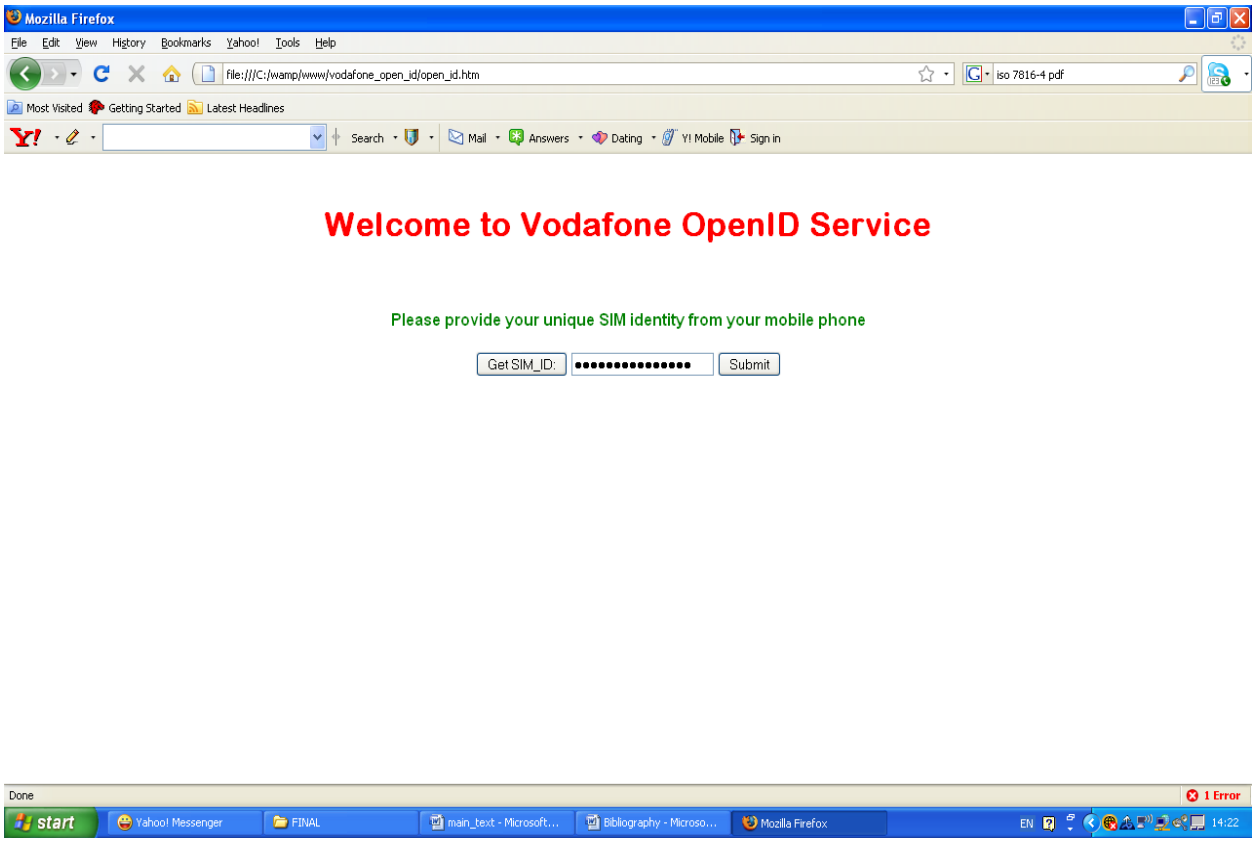


Figure 13: file:///C:/wamp/www/vodafone_open_id/open_id.htm

Observe from the figure that input field of the Get SIM_ID has been filled with the IMSI ready to be transmitted to Vodafone openID server for processing. The customer then clicks on the submit button to submit the IMSI to Vodafone openID.

Step 4: At this point, Vodafone receives the IMSI and uses it to retrieve the customers MSISDN from Vodafone's network HLR/AuC. After the receipt of the MSISDN, a record is then created for the customer's identity attributes in the database of Holloway_bank in Vodafone's directory. The fields are – User_ID, IMSI and MSISDN. Next, Vodafone then generates a temporary random number of about 8 to 10 digits long which we shall refer to as Session_ID and forwards it to the customer via the customer's phone as a text message using the MSISDN. The live-time, of the session_ID is L. The session_ID - 625aG17hWs is displayed on the customer's phone as contained in the diagram below:

Figure 14: SESSION_ID received by the customer on his mobile phone

Here is the protocol run:

Vodafone openID → Customer : Here is your session_ID & liveness of L

Once the session_ID has been sent to the customer, Vodafone openID now gives permission to the customer to enter Vodafone openID member's directory web page to enable the customer select the service provider that he wishes to access.

Observe from the figure below that the customer is now required to select one from the list of members to login with its session_ID. In the step that follows, the customer will have to select a member that he/she has registered with – in other words, he must have internet identity already assigned to him in the form of User_ID. He will then require the session_ID already received via his phone to login for verification and final access to online banking services.

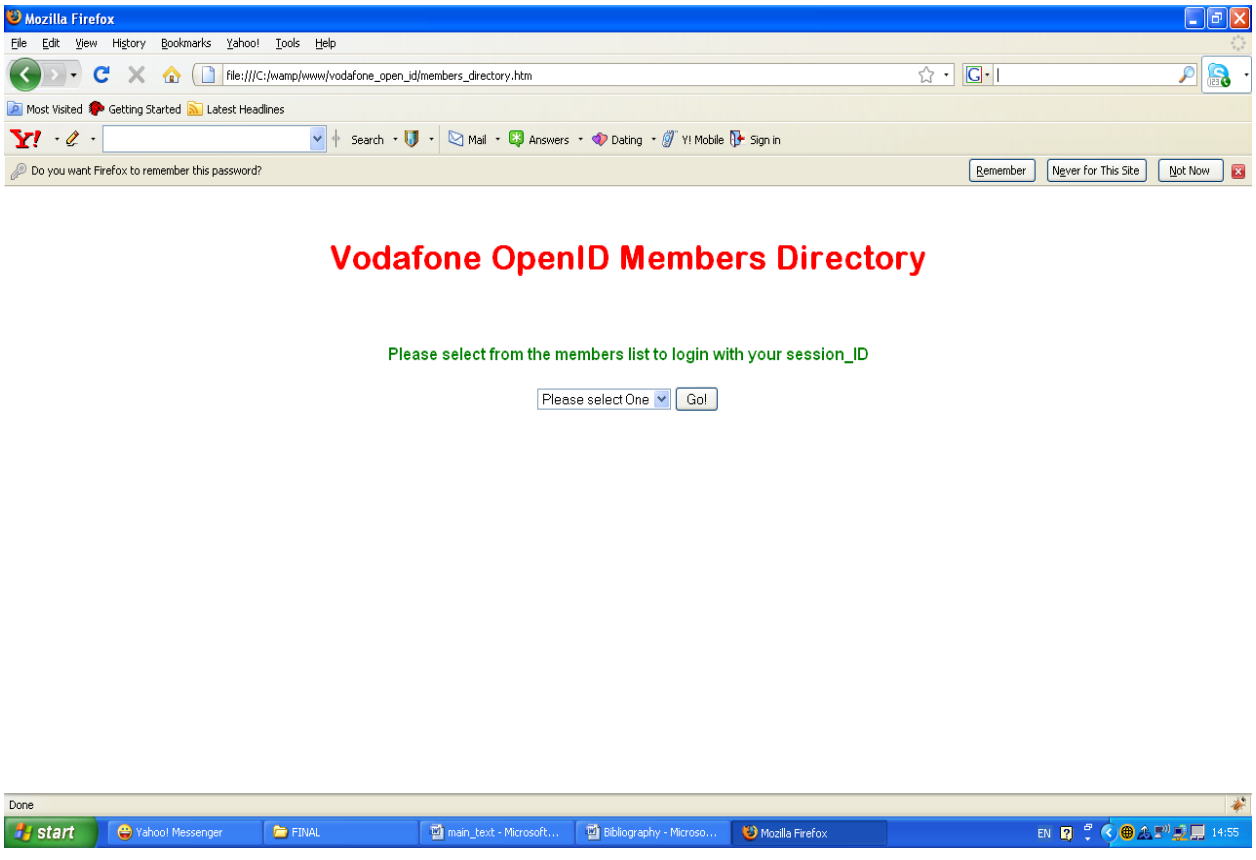


Figure 15: file:///C:/wamp/www/vodafone_open_id/members_directory.htm

Stage 5: Since the customer has just registered with Holloway bank – in other words, the customer already has internet identity – User_ID assigned to him earlier. So the customer then selects Holloway bank from the members list as displayed in the figure below:

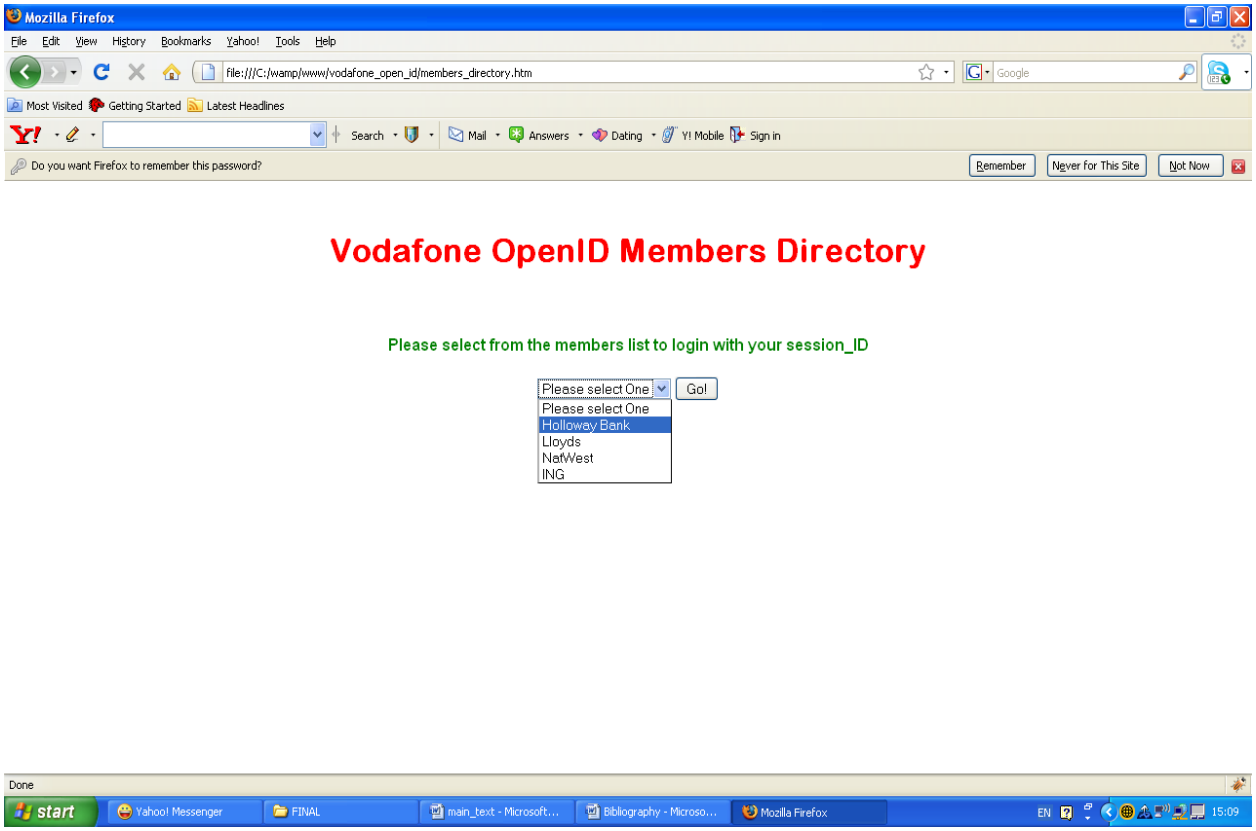


Figure 16: file:///C:/wamp/www/vodafone_open_id/members_directory.htm

From the figure, a drop down list is displayed containing the members of the Vodafone openID. For the purpose of this paper, we have on the list 4 members consisting of Holloway Bank, Lloyds, NatWest, and ING. Since the customer banks with Holloway and he has internet ID – 74895361 assigned to him already by Holloway, he

then selects Holloway from the list as indicated above. There is a dedicated link using http protocol from Vodafone member's directory to every member of the OpenID. This, in a way prevents any form of phishing attack as the customer is taken directly to the website of the SP.

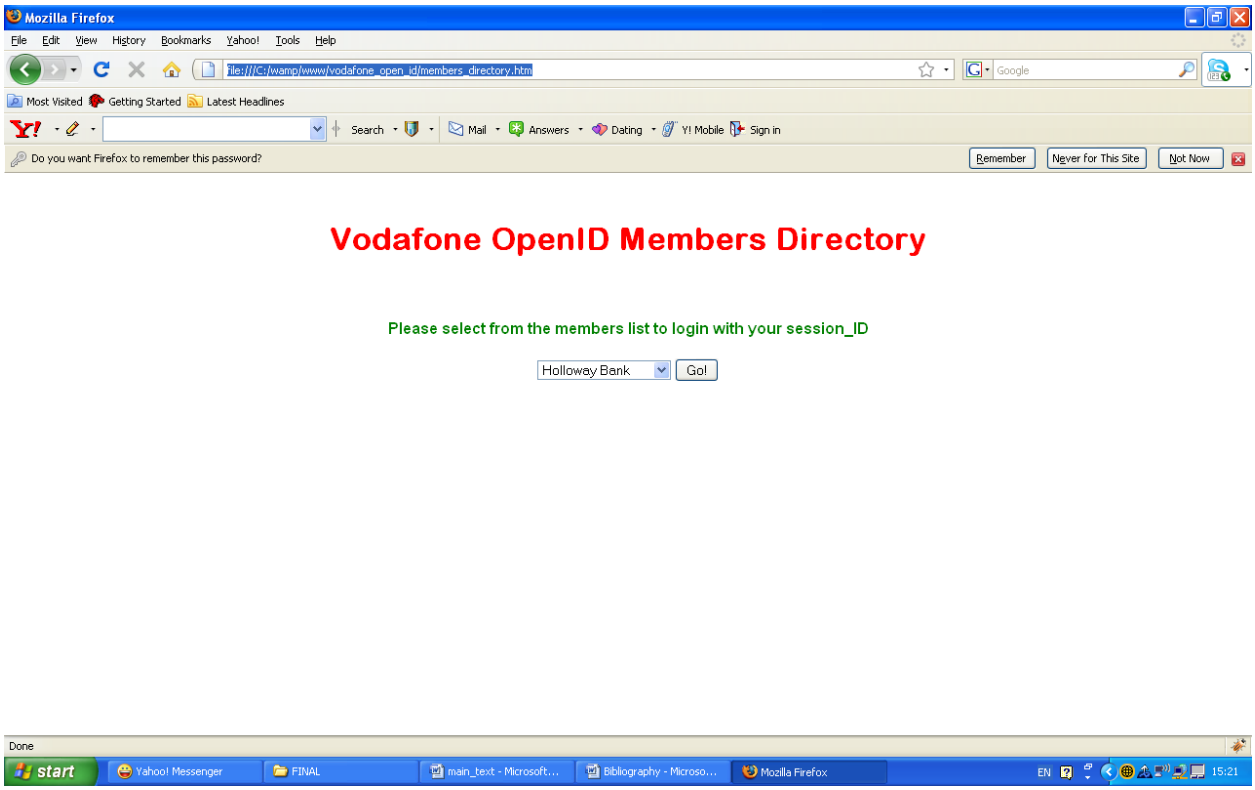


Figure 17: file:///C:/wamp/www/vodafone_open_id/members_directory.htm

From the figure, the customer's selection has been displayed in the input field as we can see. The customer then clicks on the "GO!" button and is redirected to the authentication page of Holloway bank for login with his session_ID.

Stage 6: At this point, the customer has been redirected to the authentication page of Holloway as displayed earlier in Figure 11. However, this time around since he has registered for online banking already with a valid internet ID assigned to him by Holloway, he then clicks on the Login button and a pop-up box shows up for him to enter his Session_ID, and required to concatenate it with the URL of Vodafone - Vodafone_open_id.com using "@". This requirement is necessary to accommodate any need for the SP (Holloway bank) to belong to other OpenID provider without conflict.

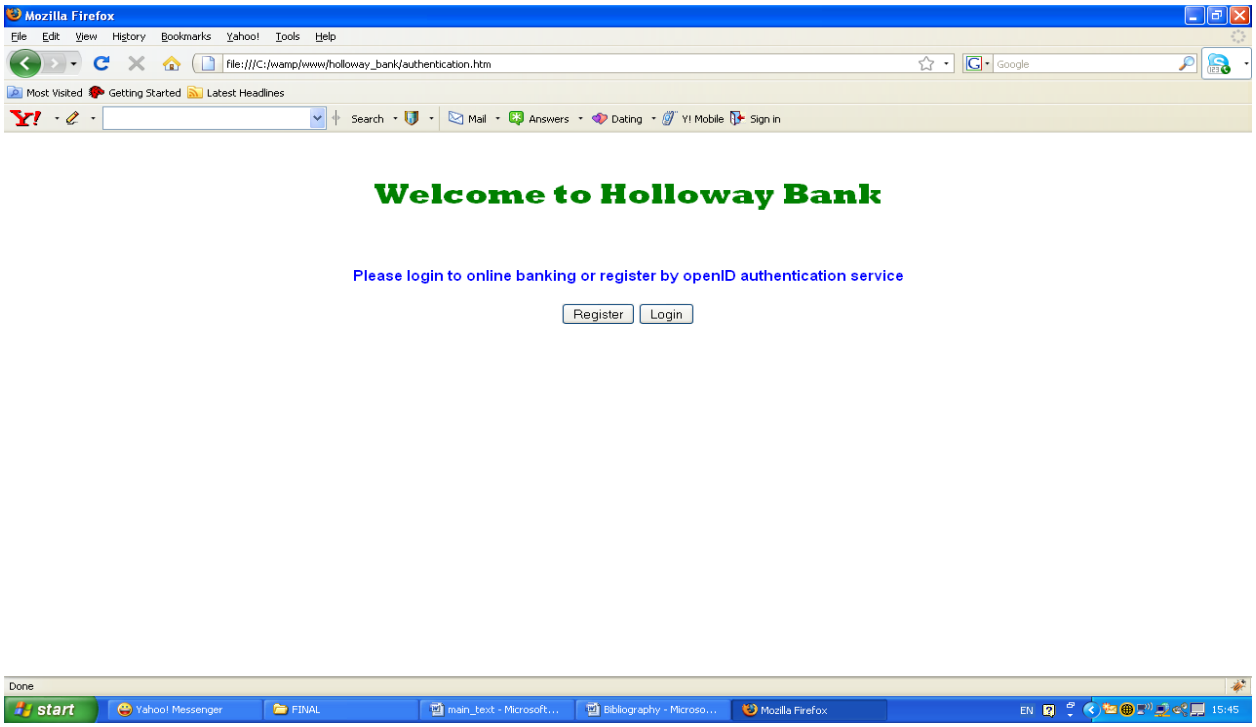


Figure 11: file:///C:/wamp/www/holloway_bank/authentication.htm

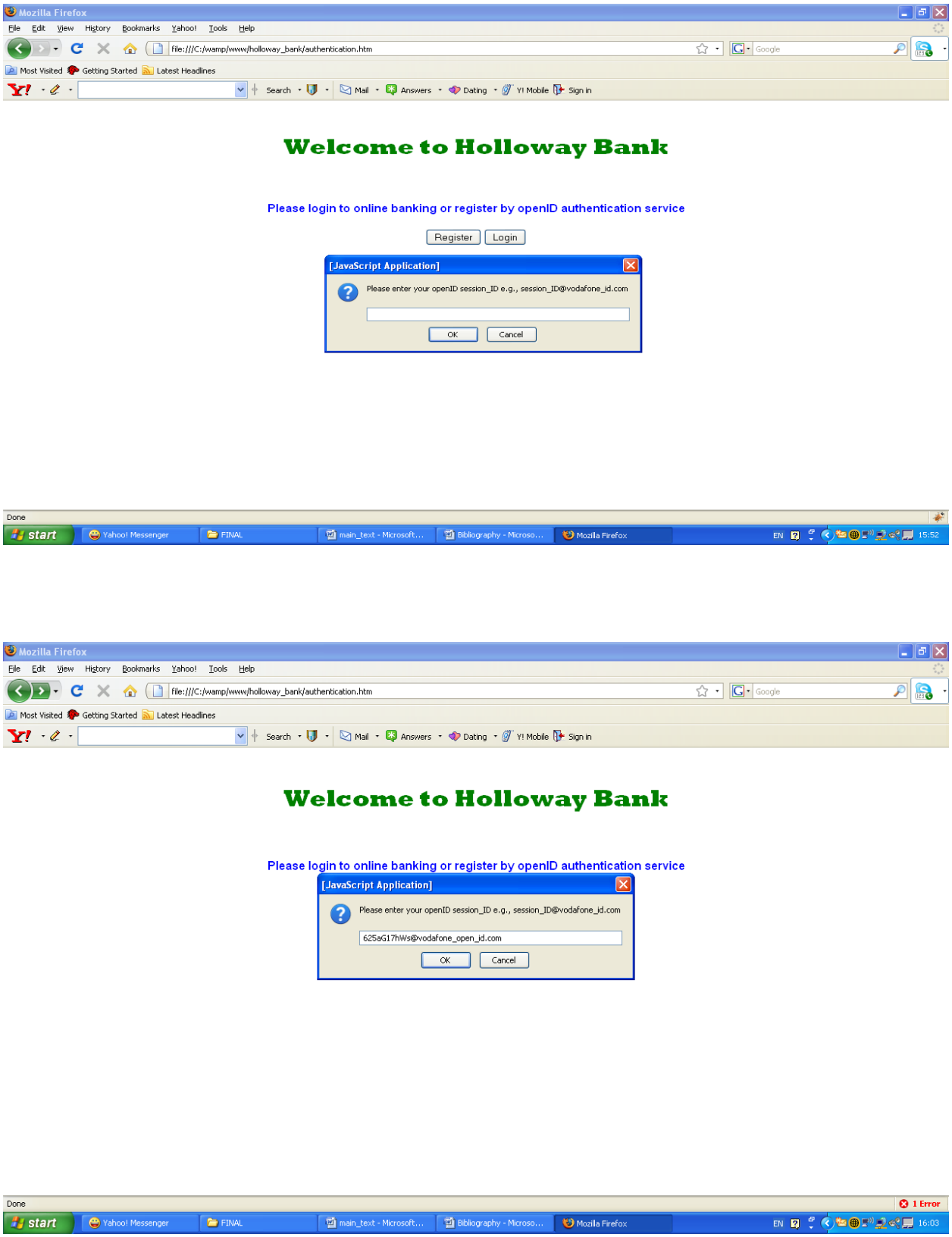


Figure 18: file:///C:/wamp/www/holloway_bank/authentication.htm

From above figure, the customer has just typed in his session_ID@vodafone_open_id.com in the form of "625aG17hWs@vodafone_open_id.com". Upon the receipt by Holloway, it retrieves the Session_ID and forwards it to Vodafone for verification as follows:

Holloway_bank → Vodafone openID : Please verify the customer with session_ID - 625aG17hWs & let me have his user_ID

Two tasks will be carried out here by Vodafone – verify the authenticity of the session_ID - 625aG17hWs and after which, retrieve the internet identity of the customer – User_ID and forward it to Holloway. The protocol runs is given as:

Vodafone openID → Holloway_bank : session_ID confirmed, here is the user_ID (74895361) of the customer

Stage 7: This is the last stage of the demonstration process. Holloway bank receives the session_ID confirmation once the user_ID is forwarded to it by Vodafone. So upon receipt of the user_ID – 74895361 by Holloway, a match is performed after which the customer is then allowed to the home page of Holloway online banking services. This can be shown by the protocol runs and diagram below:

Holloway_bank → Customer : You have been authenticated, you may carry out your online banking activities as desired

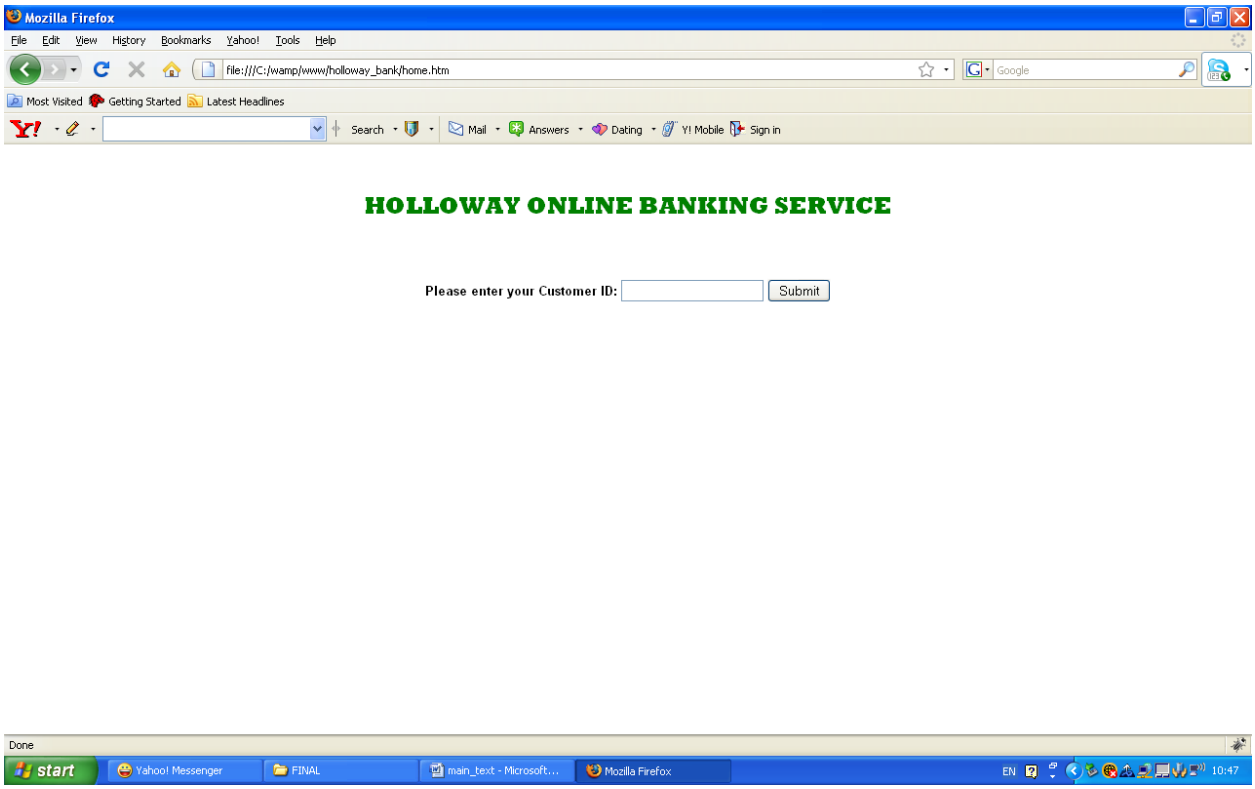


Figure 19: file:///C:/wamp/www/holloway_bank/home.htm

4.1 Chapter Summary

We devoted this chapter exclusively to demonstrating the proof-of-concept of our design using simple web services tools to simulate the operation of the application. Our use case is SIM-based secure customer authentication mechanism in online banking. A hypothetical bank – Holloway bank represented the Service Provider while Vodafone OpenID Authentication Service represented the IDP. The next chapter shall focus on the analysis of our design with emphasis on the security challenges, the prons/cons of the application, and finally a peep into the implementation requirements.

CHAPTER 5

ANALYSIS

In the previous chapter, we have been able to demonstrate the workability of our design, so we want to look at some specific challenges to the application vis-à-vis implementation requirements.

5.1 Accessing the SIM

One of the challenges of this application is how to access the IMSI in the SIM without compromising its security. In chapters 3 and 4, we have identified three possible methods to securely communicate and retrieve data from the SIM. We have USB data cable, Bluetooth and NFC-enabled phone and an NFC reader or an NFC capable laptop. The purpose of these three technologies is to enable the phone and the PC/ laptop to communicate. On the other hand, if the phone has WAP services or PDA is used, then the operation can be carried out directly without a physical connectivity medium. In either case, a plug-in has to be installed at the IDP server to enable the browser to query the SIM for the IMSI via the phone without compromising the security of the SIM. A couple of such plug-ins has been developed by these device manufacturers which may be obtained either freely on the internet or by purchase. In

other to separate this application from the normal GSM authentication network, a PKI-enabled SIM such as RSA would be required, and the public key of the IDP has to be stored in the SIM. This public key will be required by the SIM for the encryption of the IMSI which will be sent to the IDP during authentication. An applet would be required to be installed in the SIM to perform the encryption function and process the request to respond to the APDU command that will be sent to the SIM for the retrieval of the IMSI. This applet can only be installed by the GSM operator that issues the SIM via a SIM Application Toolkit because they have the key to access the SIM file system. *SIM Application Toolkit is a set of applications and related procedures which may be used during the network operation phase of a GSM* (39).

5.2 Threat Analysis

In chapter 3 we touched briefly on the security considerations of the various exchange of messages that took place in the application design to enable us determine the appropriate cryptographic mechanism to incorporate in the design framework. In this section we shall isolate each potential type of attack on the application as a whole, analyze it, and examine the extent of impact it would have in the event of occurrence. This is to enable appropriate cryptographic mechanism to be recommended for protect against such threat.

5.2.1 Eavesdropping

Basically, there are two sensitive user's credentials that an attacker may be interested in – the IMSI and Session_ID. In the case of the IMSI, three areas of vulnerabilities have been identified. They are the path between the SIM and the handset, the connection between the handset and the PC or laptop, and finally from the client to the IDP server. If the IMSI is not encrypted on these paths, then an attacker can eavesdrop on the IMSI, however, the IMSI is not sent in the clear; it is encrypted by the public key of the IDP stored in the SIM as earlier discussed. On the other hand, the session_ID is equally vulnerable at three points on the design framework – first, when session_ID is forwarded to the user's phone via SMS from the IDP, when the user submits it to the SP at point of login, and finally when the SP forwards it to the IDP for verification. If an attacker gets hold of the session_ID, he can impersonate the user. The cryptographic mechanism that we have introduced is first for the session_ID to have a liveness (L) after which it expires. This will reduce the attacker's time. Secondly, the SMS between the IDP and the mobile station is encrypted by the cipher key (k_c) by OTA (Over the Air) server, so no eavesdropping can succeed on this path. The last point of vulnerability is the path between the client and the SP, and the SP and IDP. We therefore recommend that the channel for the exchange of the session_ID between the client and the SP should be protected by SSL and likewise between the SP and the

IDP. Equally, the SP and the IDP can secure their exchange of message here with their PKI (public key infrastructure).

5.2.2 Man-in-the-middle attack

There is one point of vulnerability that this type of attack may take place – the transmission of the IMSI between the client and the IDP. An attacker may decide to spoof the IMSI and inject another IMSI for transmission to IDP. Unfortunately, this type of threat has been taken care of as each IMSI has its own record and any IMSI not registered with the IDP will be ignored. But one security threat it can cause is probably denial of service attack to the user. However, the attacker cannot succeed because the IMSI is not sent in the clear; it is always encrypted by the public key of the IDP, so the attacker cannot recognize what the message actually contains.

5.2.3 Replay attack

The point of vulnerability for a replay attack to take place is when the IMSI is being transmitted to the IDP during identification stage. Unfortunately, if an attacker is already in possession of the IMSI probably captured in a previous session, whenever he transmits the encrypted IMSI to the IDP for authentication, the IDP will then send the session_ID, also encrypted to the user via SMS, unfortunately, the attacker cannot

receive the session_ID because he is not in possession of the phone. Even if he intercepts the session_ID on its way to the user, he cannot make use of it because it is encrypted with the cipher key (k_c). Similarly, assuming the attacker was able to eavesdrop on the session_ID at the point of login by the user, it will be difficult for the attacker to succeed because the channel is protected by SSL/TLS and even at that, the session_ID has a liveness of L after which the session_ID expires, because the session_ID is meant for only one session.

5.2.4 Denial of service attack (DoS)

We already mentioned it when discussing man-in-the-middle attack in 5.2.2 above. However, depending on the motive of the attacker, otherwise the attacker cannot derive any economic benefit from this attack. Even at that, the chance of succeeding is still very slim as he may need to install a malware on the browser of the client to be able to spoof the encrypted IMSI, and malware can be detected or prevented by installing personal firewalls and anti-virus programs in your system.

5.2.5 Stolen mobile phone

If a mobile phone is stolen with the intent of impersonating the user, then the attacker may need to know the SP(s) that the user must have registered with otherwise

he will not be able to masquerade as the user. If on the other hand the attacker already has such information, then the next stumbling block is the PIN of the SIM. So to counter this type of attack, a user must make sure that his SIM is PIN-protected. Another measure will be for a user to contact the mobile service provider for the SIM to be blocked immediately. But we must say that this attack is the most potent of all the attacks discussed, so the best prevention is for a user to careful with his/her phone.

5.3 Pros and Cons

In this session we shall be evaluating the benefits of this application and the possible challenges.

5.3.1 Pros

In considering the various benefits of this application, we shall itemize them as follows:

i. User experience – this application was designed with user convenience in mind. The frustrating experiences with username/password authentication have been eliminated with improved security and peace of mind. All user needs to login at anytime is his/her mobile phone, which is always with him.

ii. User anonymity – this application provides for user anonymity as an attacker cannot associate any of the authentication credentials such as IMSI or session_ID to the user except the IDP as against the use of credentials such as username, email or MSISDN that are used in most application which an attacker can associate with the owner/user.

iii. Privacy concerns – the issue of privacy of users credentials such as email, username, and other private information is no longer in contention as the only credential that may be considered private in this design is the MSISDN (phone number), and this does not leave the IDP who is equally the issuer.

iv. Prevent spams – spamming is one of the potential attacks prevalent in most authentication application. Spamming is an unsolicited communication in the form of email, SMS, or even physical correspondence through postal address. This attack is totally prevented as such users contacts are not involved in the design. The mobile phone number does not leave the IDP.

v. Prevents phishing – this is one of the cardinal objective of our design – phishing was elaborately discussed at the introductory part of chapter 3. Firstly, there is no way a user can receive an unsolicited mail, and secondly, if a user is tricked to another website masquerading as the website of the SP, the identification process, which is the submission of the IMSI is meaningless to the attacker as he cannot use it for any attack.

vi. Lower operations overhead – all the overhead resulting from forgotten and reset of password have been prevented. It therefore reposes more confident for the online communities.

vii. Added-value – to the network operators, it is another way of rewarding their customers by continuously evolving services that will add value to the relationship. And since this design leverages on the existing network infrastructure, the authentication service can be offered at a fractional cost to the operator's equivalent to the cost of an SMS per authentication session excluding the cost of set-up.

5.3.2 Cons

The challenges/cons of this application can be summarized thus:

i. The security of the SIM – since the application involves reading a value – the IMSI from the SIM, it may expose the SIM to other form of attack common in the internet such as the injection of malware.

ii. Identity Providers – this application is designed for the mobile operators in mind.

iii. Stolen phone – as discussed above, the point of failure of this application may be when the users mobile phone is stolen and it was not reported early

enough for the SIM to be blocked or if the user is careless enough not to PIN-protect his SIM.

5.4 Implementation requirements

Although the appropriateness of the inclusion of this session may be subjective since our design does not include detailed design, however, the implementation requirement given here can be considered as a guide rather than normative requirements. We shall itemize these requirements as much as possible:

5.4.1 Device requirements

In considering this requirement, we shall break it down into three – representing the Client/user, SP and IDP. At the client side, a user must own a mobile phone in addition to a means of connectivity between the mobile phone and a PC or laptop. We have already given three options – USB data cable, Bluetooth, or NFC-capable phone and PC/laptop. Today, we have very common technology that can act as an add-on device to the SIM that will automatically make the SIM NFC-enabled. A very good example is Waver, and a good example of an NFC reader is Tikitag/Touchatag. An interested reader can find information about these devices at: www.bladox.com. On the other hand, if a user has a mobile phone that has WAP services, then there will be no

need for any connectivity medium as the WAP browser with the appropriate plug-in installed at the IDP side should be able communicate with the SIM securely. The issue of SIM security has been discussed previously. Off course, apart from the mobile phone and connectivity, a user must have access to a PC or laptop with internet access to be able to browse the internet. We equally recommended the installation of personal firewalls and anti-virus software at the client side to prevent malware attack.

On the SP's side, no additional hardware or software devices is required except a Servlet that will be able to receive, process and respond to requests from the user and the IDP. This servlet shall be provided by the IDP to the SP for the application, so both the SPand IDP should be able to reach a mutual understanding on that issue.

On the IDP's side, we already said that this application is targeted at the mobile operators, so this Identity Service is an added-value service and requires very little investment. The additional device requirements are a plug-in for communication between the client's browser and the SIM/phone, an applet installed in the SIM that will ensure secure communication. At the server's end, it is not necessary to bundle this application into their AuC (authentication Centre), a back-end server can be installed to manage the database of the users and be able to communicate with the SP directly. In order to simplify the operation, particularly matching the user's IMSI with their corresponding MSISDN, the entire database of IMSI and their corresponding MSISDN

can be copied to the IDP database with regular updates. This will isolate the IDP from any form of interference with the normal operations of the HLR/AuC.

5.4.2 Memory requirement

At the user's end, the issue of memory requirement does not arise as there is no new major application that will be required except the drivers of the plug-in device such as the NFC reader or Bluetooth. The SIM applet required for the application will require very little memory capacity. At the IDP's end, any standard database server should suffice.

5.5 Chapter Summary

In the analysis of our design, we discussed the security considerations identifying the various possible attacks and the appropriate security mechanisms to deal with those potential attacks. We equally considered the prons/cons of the system and finally concluded with a not too detail implementation requirements. The next chapter is our last and final chapter of this report and shall consist of the conclusion of the entire report and possible recommendation for future research.

CHAPTER 6

CONCLUSION AND RECOMMENDATIONS

Since the upturn of the 21st century, the internet has become a sine-qua-nun of business, governance, education and social networking. Thus the criminality that pervades the society at large now found a better attraction on the cyberspace. Unlike in the physical world, a criminal in China can commit a cybercrime in the United States at the comfort of his home with just a PC/laptop with connectivity to the internet. Now, how do we identify the cyber criminal? This brings us to the thorny issue of identity on the internet which is the root of the subject of this research paper. Unlike in the physical world where the instruments of identity such as ID card, passport and biometric data is tied to the individual owner, on the internet, identity instruments are separated from the owner. On the internet, you can create your own identity, and even have multiple identities for different scenarios – this is the precursor to identity theft and privacy concerns on the internet.

Online banking is one of the most potent areas at the moment where internet identity has resulted to significant damage and loss to banks – a peep into the statistics of online fraud (35) in banks since the beginning of this 21st century is mind-burgling and it continues to rise despite myriads of solutions being developed to combat this menace.

The subject matter of this research paper – SIM based internet identity for customer authentication in online banking is a child of the search for a secure and sustainable solution to user internet identity on the cyberspace. In the reviews of related literatures to the subject matter of this paper, we observe that several organisations are being set up over the years with the responsibility of providing standards and specifications for developers to build applications to tackle internet identity problems not only for online banking but for online services in general. The most recent of such organisations is OpenID (27) that came on-stream in 2005, while others are Microsoft Webs Services (15), Liberty Alliance (17), and OASIS (24). During the course of this review, we equally found several applications based on these specifications that have been developed by various vendors some of which are still in use today. Some of these applications are RSA SecureID, Open SSO offered by Sun Microsystems, SecureLogin by Novell, SP Sign-on by Unisys, Microsoft .NET Passport and myriads of unknown proprietary applications that are in use today. Out of the lot, we selected Microsoft .Net passport based on Microsoft WS-* specifications partly because of its popularity, and studied it in detail. We observed that despite the popularity of Microsoft products, .Net Passport faced several challenges such as mutual suspicion between the tripartite parties involved – the Identity Provider (Microsoft), Service Providers and users. The mutual suspicion resulted from mutual trust and privacy concerns of user's authentication credentials. Other issue is the collaboration of identity providers across the internet for

the provision of the internet identity services. This led to the low success of .Net Passport and the metamorphosis to the current Microsoft Winows Live ID (31).

One of the most common points of failure that we found to be prevalent in almost all the applications that we reviewed is the username/password. Username and password is vulnerable to so many attacks ranging from phishing, pharming, identity theft, malware attack, password cracking, and owner abuse. This point of vulnerability was the ignition for our design by replacing the username/password – “what the user knows” with “what the user possesses” – the ubiquitous SIM. Another area of attraction for our SIM based application is the proven security of GSM services and its 24/7 availability. Our design is based on OpenID specifications with GSM operators as the Identity Providers.

6.1 Recommendations

In our analysis, we have tried to present the pros/cons of this application within our perceived judgement and we found the connectivity and the communication between the SIM and the PC/laptop quite challenging because of the security implication of this task. Although, we gave our opinion, but we believe there is still much to be done in terms of practical implementation and operation. On the whole, we found the result of

this research quite exciting because of its re-usable potentials for online payment in Customer Not Present (CNP) transactions on the internet.

BIBLIOGRAPHY

1. **Microsoft Corporation.** Microsoft's Vision for an Identity Metasystem. [Online] May 2005. [Cited: 06 June 2009.] http://msdn.microsoft.com/en-us/library/ms99642.aspx.

2. **Mayes, K E and Markantonakis, K.** *Smart Cards, Tokens, Security and Applications.* New York : Springer, 2008.

3. **Cameron, K.** The Laws of Identy. [Online] May 2005. [Cited: 6 June 2009.] http://msdn.microsoft.com/en-us/library/ms99456.aspx.

4. **OpenID.** OpenID Authentication 2.0 - Final. [Online] 2.0, 05 December 2007. [Cited: 9 June 2009.] http://openid.net/specs/openid-authentication-2_0.html.

5. **Network Working Group.** The Internet Standards Process -- Revision 3. [Online] October 1996. [Cited: 17 June 2009.] ftp://ftp.rfc-editor.org/in-notes/bcp/bcp9.txt.

6. **Merchant, Guy.** Identity, Social Networks and Online Communication. [Online] 2006. [Cited: 19 June 2009.] http://dx.doi.org/10.2304/elea.2006.3.2.235.

7. **Purser, Steve.** *A Practical Guide to Managing Information Security.* Massachusetts : Artech House, 2004.

8. **Rankl, Wolfgang and Effing, Wolfgang.** *Smart Card Handbook.* 3. England : John Wiley, 2003.

9. **Tipton, Harold F. and Krause, Micki, [ed.].** *Information Security Management Handbook.* 5. New York : Auerbach Publications, 2005.

10. **Jain, Anil K., Ross, Arun and Prabhakar, Salil.** An Introduction to Biometric Recognition. *IEEE Transactions on Circuits and Systems for Video Technology.* Vol. 14, No. 1, January 2004, pp. 4-20.

11. **Windley, Phillip J.** *Digital Identity.* California : O'Reilly Media, 2005.

12. **IBM, Microsoft.** Security in a Web Services World: A Proposed Architecture and Roadmap. [Online] 1, 7 April 2002. [Cited: 21 June 2009.] http://download.boulder.ibm.com/ibmdl/pub/software/dw/library/ws-secmap.pdf.

13. **IBM Corporation.** web Services Security. [Online] 1 March 2004. [Cited: 21 June 2009.] http://www.ibm.com/developerworks/webservices/library/specification/ws-secure/.

14. **W3C.** Web Services Policy 1.5 - Framework. [Online] 4 September 2007. [Cited: 17 June 2009.] http://www.w3.org/TR/ws-policy/ws-policy-framework.pdf.

15. **IBM and Microsoft.** Understanding WS-Federation. [Online] 1.0, 28 May 2007. [Cited: 06 June 2009.] http://msdn.microsoft.com/en-us/library/bb498017.aspx.

16. **Gudgin, Martin and Nadalin, Anthony (ed.).** WebServices Trust Langguage (WS-Trust). [Online] February 2005. [Cited: 18 June 2009.] http://spcs.xmlsoap.org/we/2005/02/trust/WS-Trust.pdf.

17. **Liberty Alliiance Project.** History. [Online] [Cited: 01 July 2009.] http://www.projectliberty.org/liberty/about/history.

18. **Liberty Alliance Project.** Lberty ID-FF Architecture Overview. [Online] 1.2-errata-v1.0, 2004-2005. [Cited: 02 July 2009.]

http://www.projectliberty.org/resource_center/specifications/liberty_alliance_id_ff_1_2_specifications.

19. —. Liberty ID-WSF: A Web Services Framework . [Online] 2004. [Cited: 03 07 2009.] http://www.projectliberty.org/liberty/resource_center/papers/liberty_id_wsf_a_web_services_framework_pdf.

20. **Hunt, Phillip and Mirshra, Prateek.** Identity Governance Framework. [Online] November 2006. [Cited: 04 July 2009.] http://www.oracle.com/technology/tech/standards/idm/igf/pdf/IGF-Overview-02.pdf.

21. **Liberty Alliance Project.** Identity Governance. [Online] [Cited: 04 07 2009.] http://www.projectliberty.org/strategic_initiatives/identity_governance.

22. **Aisien, John.** The Identiy Governance Framework, Liberty Alliance's Private Initiative. [Online] October 2007. [Cited: 04 July 2009.] http://www.projectliberty.org/strategic_initiatives/identity_governance.

23. **Liberty Assurance Project.** Liberty Identity Assurace Framework. [Online] 1.1, 2007-2008. [Cited: 04 July 2009.] http://www.projectliberty.org/strategic_initiatives/identity_assurance.

24. **OASIS.** About OASIS. [Online] [Cited: 05 July 2009.] http://www.oasis-open.org/who/.

25. —. A Gentle Introduction to SGML. [Online] [Cited: 05 July 2009.] http://xml.coverpages.org/general.html#hist.

26. **W3C.** Extensible Markup Language (XML). [Online] 1.0, 10 February 1998. [Cited: 05 July 2009.] http://www.w3.org/TR/1998/REC-xml-19980210.pdf.

27. What is OpenID? [Online] [Cited: 15 July 2009.] http://openid.net/what/.

28. **OpenID.** OpenID Authentication 2.0 - Final. [Online] 05 December 2007. [Cited: 15 July 2009.] http://openid.net/specs/openid-authentication-2.0.html.

29. —. OpenID Provider Authentication Policy Extension 1.0. [Online] 30 December 2008. [Cited: 15 July 2009.] http://openid.net/specs/openid-provider-authentication-policy-extension-1_0.html.

30. —. OpenID Attriute Exchange 1.0. [Online] 5 December 2007. [Cited: 16 July 2009.] http://openid.net/specs/openid-attribute-exchange-1_0.html.

31. **Microsoft.** Introduction to Windows Live ID. [Online] 2008 February. [Cited: 06 June 2009.] http://msdn.microsoft.com/en-us/library/bb288408.aspx.

32. **ETSI.** GSM Technical Specification, GSM 11.11. [Online] 5.3.0, July 1996. [Cited: 09 July 2009.] http://www.3gpp.org/ftp/Specs/archive/11_series/11.11/.

33. **Hillebrand, Friedhelm, [ed.].** *GSM and UMTS: The Creation of Global Mobile Communication.* England : John Wiley, 2001.

34. **Turban, Efraim, et al.** *Electronic Commerce (A Managerial Perspective.* New Jersey : Prentice-Hall, 2000.

35. Online banking fraud 'up 8,000%. [Online] 13 December 2006. [Cited: 30 July 2009.] http://news.bbc.co.uk/2/hi/uk_news/politics/6177555.stm.

36. **Stamford, Conn.** Gartner Survey Shows Phishing Attacks Escalated in 2007; More than $3 Billion Lost to These Attacks. [Online] 17 December 2007. [Cited: 30 July 2009.] http://www.gartner.com/it/page.jsp?id=565125.

37. Phishing attacks soar in the UK. [Online] 15 April 2008. [Cited: 30 July 2009.] http://news.bbc.co.uk/2/hi/technology/7348737.stm,.

38. **ISO/IEC.** ISO/IEC 7816-4 Second edition. [Online] 15 January 2005. [Cited: 12 August 2009.] http://webstore.iec.ch/preview/info_isoiec7816-4%7Bed2.0%7Den.pdf.

39. **ETSI.** GSM 11.14. [Online] 5.2.0, December 1996. [Cited: 14 August 2009.] http://www.ttfn.net/techno/smartcards/GSM11-14V5-2-0.pdf.

APPENDIX

SOURCE CODES

```
<!-- Holloway bank authentication page -->
<html>
  <head>
    <script type="text/javascript">
      function show_prompt()
      {
        var session_id=prompt("Please enter your openID session_ID
        e.g., session_ID@vodafone_id.com");
        if (session_id!=null && session_id!="")
       {
        document.write("<a href='home.htm'>Click to continue</a>")
       }
        else
       {
        document.write("You have not registered!");
        }
        }
    </script>
  </head>
<body>
<br/>
<center>
  <p><h1 style="font-family:rockwell extra bold;color:green">Welcome
to Holloway Bank</h1></p>
<br/>
  <p><h4 style="font-family:arial;color:blue">Please login to online
banking or register by openID authentication service
  </h4></p>
```

```
  <form action="file:///C:/wamp/www/vodafone_open_id/open_id1.htm"
method="post">
    <button id="button1">Register</button>
    <input type="button" onclick="show_prompt()" value="Login" />
  </form>
</center>
</body>
</html>



<!-- Holloway bank home page -->
<html>
  <body>
  <br/>
  <center>
    <p><h2 style="font-family:rockwell extra
bold;color:green">HOLLOWAY ONLINE BANKING SERVICE
    </h2></p>
    <br/>
    <form name="input" action="html_form_submit.asp" method="get">
      <h5 style="font-family:arial">Please enter your Customer ID:
      <input type="password" name="password" />
      <input type="submit" value="Submit" />
      </h5>
    </form>
</center>
</body>
</html>



<!-- Vodafone OpenID -->
<html>
  <head>
    <script type="text/javascript">
      function get_sim_id()
    </script>
```

```
  </head>
<br/>
<body>
<center>

  <p><h1 style="font-family:arial rounded mt bold;color:red">Welcome
to Vodafone OpenID Service</h1></p>
  <br/>
  <p><h4 style="font-family:arial;color:green">Please provide your
unique SIM identity from your mobile phone</h4>
  </p>
  <form
action="file:///C:/wamp/www/vodafone_open_id/members_directory.htm"
method="post">
    <input type="button" onclick="get_sim_id()" value="Get SIM_ID:" />
    <input type="password" name="password" value="mmmmmmmmmmmmmmm" />
    <input type="submit" value="Submit" />
  </form>
</center>
</body>
</html>



<!--Vodafone openID member directory -->
<html>
  <head>
    <script type="text/JavaScript">
      function display_web_page()
     {
      var url=document.combobox.selected.value
      document.location.href=url
     }
    </script>
  </head>
<br/>
<body>
<center>
```

```
  <p><h1 style="font-family:arial rounded mt bold;color:red">Vodafone
OpenID Members Directory</h1></p>
  <br/>


  <p><h4 style="font-family:arial;color:green">Please select from the
members list to login with your session_ID</h4>
  </p>
  <form name="combobox">
    <select name="selected">
    <option value="" SELECTED>Please select One
    <option
value="file:///C:/wamp/www/holloway_bank/authentication.htm">Holloway
Bank</option>
    <option value="http://www.lloyds.com">Lloyds</option>
    <option value="http://www.natwest.com">NatWest</option>
    <option value="http://www.ing.com">ING</option>
  </select>
    <input type=button value="Go!" onClick="display_web_page();">
  </form>
</center>
</body>
</html>
```