OSEI TUTU II INSTITUTE FOR ADVANCED ICT STUDIES, GHANA
AFFLIATED TO
KWAME NKRUMAH UNIVERSITY OF SCIENCE AND TECHNOLOGY, KUMASI
SCHOOL OF GRADUATE STUDIES

AN AUTHENTICATION ARCHITECTURE BASED ON THE CONCEPT OF SINGLE SIGN-ON (KNUST AS A CASE STUDY)

A DISSERTATION PRESENTED TO THE INSTITUTE IN PARTIAL FULFILMENT OF THE REQUIREMENT FOR THE AWARD OF THE DEGREE OF MASTER OF SCIENCE IN ADVANCED ICT STUDIES (MANAGEMENT OPTION)

BY
KWABENA OHENE APAU NYANTENG
(Bsc Electrical Engineering)
OCTOBER, 2009

OSEI TUTU II INSTITUTE FOR ADVANCED ICT STUDIES, GHANA
AFFLIATED TO
KWAME NKRUMAH UNIVERSITY OF SCIENCE AND TECHNOLOGY,
KUMASI
SCHOOL OF GRADUATE STUDIES

AN AUTHENTICATION ARCHITECTURE BASED ON THE CONCEPT OF SINGLE SIGN-ON (KNUST AS A CASE STUDY)

A DISSERTATION PRESENTED TO THE INSTITUTE IN PARTIAL FULFILMENT OF THE REQUIREMENT FOR THE AWARD OF THE DEGREE OF MASTER OF SCIENCE IN ADVANCED ICT STUDIES (MANAGEMENT OPTION)

BY
KWABENA OHENE APAU NYANTENG
(Bsc Electrical Engineering)
OCTOBER, 2009

DECLARATION

I hereby declare that this study was under taken independently and it is my original work. It is not replication of any work either published or unpublished. All references made in this study are due acknowledged. Finally, all aspects of this study have been discussed with and approved by my supervisor, Dr. K. O. Boateng.

Signature .... Date 21-10-09

KWABENA O. A. NYANTENG

(STUDENT)

I declare that this dissertation was written under my supervision and that the Student has been consistent in his interaction with me for guidance and direction. He has my consent to present it for assessment.

Signature .... Date 21/10/09

DR. K. O. BOATENG

(SUPERVISOR)

DEDICATION

This study is dedicated to my parents,

Dr. Victor Kwame Nyanteng

and

Mrs. Phyllis Nyanteng

ACKNOWLEDGEMENTS

I would first and foremost like to thank God for granting me the strength to complete this research and my masters program.

I would like to express my gratitude to my supervisor Dr. K. O. Boateng, for assisting me with this thesis. I thank him for his guidance and contributions leading to the completion of the research.

I also thank all staff and lecturers of the Osei Tutu II institute of Advanced ICT for providing me with the necessary foundation for this research study.

I would like to thank all members of the KNUST University Information Technology Services (UITS) for their valuable contributions and time. They were very helpful indeed.

I am grateful to my parents for their encouragement and belief in me. And to my siblings who have continuously supported me throughout the period of the program.

I express a special 'thank you' to Miss Prisca Amoah who has been an inspiration to me through this campaign. Her encouragement and support were instrumental in this accomplishment.

Lastly, I thank everyone else who in one way or another has helped me through this time.

KWABENA O. A. NYANTENG
OCTOBER, 2009

ABSTRACT

Most enterprises and institutions today have many applications available to its users across a local area network (LAN). In order to increase productivity, more users rely more on such applications. This translates into user and administrative problems. The most common problems are associated with multiple password retention, password complexities and multiple user account management. Users, who access multiple applications in institutions such as the case study institution (KNUST), require multiple passwords. Single Sign-On has been hailed as a solution to deal with the usability and security problems associated with multiple user authentications.

The goal of this thesis is to determine an authentication architecture that would be suitable for the environment of KNUST and other similar institutions. This has entailed a review of existing single sign-on architectures and operating mechanisms, a study of the case study institution's ICT setup, the design of the proposed authentication architecture and its concept of operations. The driving factors of the design are functionality and security. Other factors include availability and feasibility of integration.

The results will serve as a blueprint for a more detailed design and the future development of such architecture in KNUST and similar institutions.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

AMDB	Application Mapping Database
APMS	Application Mapping Software
APS	Application Server
ASDB	Application Session Database
CA	Certificate Authority
DAS	Distributed Authentication Server
DAUS	Distributed Authorization Server
FA	Fingerprint Agent
FMS	Fingerprint Management Software
FR	Fingerprint Reader
FPCDB	Fingerprint Credential Database
IT	Information Technology
KNUST	Kwame Nkrumah University of Science and Technology
LDAP	Lightweight Directory Access Protocol
LSDB	Login Session Database
MAC	Media Access Control
NOC	Network Operations Centre
PWCDB	Password Credential Database
SQL	Structured Query Language
SRI	Server Resource Index
SSO	Single Sign-On
UR	User Registry

Chapter One

Introduction to the Thesis

1.0 Introduction

This chapter introduces the reader to the subject matter and the domain of the thesis. It discusses the scope, objectives, justification and organization of the thesis. The research and sub research questions are also presented here whiles explaining the problem statement. Limitations encountered within the study are described.

1.1 Background to the study

Information Technology (IT) systems need to be robust enough to help enterprises accomplish their mission. IT has the capability to provide services and applications across an enterprise network. As the number of services and applications increase, the complexity of the IT setup also increases to accommodate such growth. As business processes depend more on IT, more users and more user accounts are created in the IT system. This growth introduces a variety of user-management problems. (Nitai Alush-Aben 2005)

On a daily basis users need to enter a combination of usernames and passwords to access IT resources. Some of these resources are the windows system, email and other applications. Ufinity's (2003) whitepaper mentions the four major classes of applications as dumb terminal, client-server, web browser and micro-browser. Usually, users have to perform separate authentications in order to access each service and application they require. They need to periodically re-authenticate to switch between resources or after periods of inactivity. This manner of authenticating users wastes time and can impact actual working time. This is more so in enterprises with a security policy enforcing regular password changes, lengths and complexities. (Secude 2002)

With so many passwords to remember for each of the IT systems resources, it is human nature to forget them from time to time. The username and password combination which is referred to as Single Factor Authentication is still the most popular amongst businesses and enterprises. It is however no longer a reliable security measure because passwords are easily compromised by social engineering methods, physical access to computers and elementary hacking techniques. Social engineering methods take advantage of users who use simple passwords. For example using usernames or passwords generated from personal information. Users also write down passwords and have them in close proximity to workstations. Unauthorized access to such a location can easily compromise the user's password and the security of the enterprise. (Secude 2002)

Norbert Steinhauser, the Vice President of Business Management at SECUDE IT Security said that:

"Release your users from passwords so they can focus on more important issues."

Multiple passwords for users ~~create~~ problems for users, IT administrators and the enterprise as a whole. When passwords are forgotten, users lose productivity time and calls are made to IT help desks for support. Secude (2002) states that Gartner assumes a user will call a company help desk up to 19 times a year as a result of password issues. In a large enterprise with many employees, this can be a substantial figure.

IT administrators who work within decentralized IT systems are faced with a difficult task of managing the system. They need to enforce security policy across all machines and services in the enterprise. They need to be able to manage user access rights and accountability for use of those rights. This can be very challenging in a decentralized system. A concept that aims to reduce the frustrations associated with multiple user authentications is known as Single Sign-On. (Parker 1995)

Single Sign-On (SSO) is a concept that describes a user of different application services who authenticates only once to a distributed system. This authentication transparently replicates to the end applications as required. Single Sign-On has evolved into an important requirement for enterprises. An important aspect of SSO is reducing frustrations associated with using and managing multiple passwords. It also aims to address stronger security and user identity management. (Parker 1995)

SSO is classified and categorized in many ways. Some of which have similar meanings and different names. SSO can be characterized by types of services (Understanding Enterprise SSO 2009), architecture classifications (Ufinity 2003) , single/multiple credential set, (Clercq 2002), (Bui 2005) and modes of operation (Parker 1995). Other characterizations of SSO also exist. These classifications will be looked at further in the next chapter.

Enterprises such as the Kwame Nkrumah University of Science and Technology (KNUST) have such network setups. There are various applications and services available to users across the network that each require unique authentication. Users within KNUST who use multiple services are also tasked with remembering multiple passwords. This results in an enterprise network with high administrative overhead and no centralized system to coordinate user accountability.

1.2 Statement of the problem

KNUST together with other large enterprises cannot avoid the use of multiple applications and services in their daily activities. KNUST does not have a system wide authentication system that can provide administrative control across these applications. KNUST needs to control user access rights and monitor accountability for those rights in a way that is spanned across all system resources. The absence of such a system in KNUST makes system wide accountability and authorization a difficult issue to deal with. The thesis will try to answer the following main research question:

What kind of authentication architecture will be appropriate for KNUST?

1.3 Objectives of the study

This thesis hopes to research into SSO architectures and mechanisms that will aid the design of an authentication architecture for KNUST.

The thesis aims to present an outline of such architecture with relevant regard to KNUST's current setup and future ICT strategy. This thesis will serve as a reference when KNUST decides to choose or develop an authentication system that will coordinate across all system components and provide accountability and system security. The eventual implementation of this system in KNUST will improve authentication, accountability, security and IT management.

1.3.1 Research Questions

The problem statement from the previous section constitutes the main research question. The following sub research questions will provide direction in answering the main research question:

- What are the different classifications of central authentication?
- What is SSO, how does it work and what kinds exist?
- What types of information systems are available in KNUST?
- What is the current state of authentication and authorization across IT services in KNUST?
- What future projections does KNUST have for IT services?

1.4 Justification

KNUST has five strategic objectives. One of these objectives states that KNUST aims:

"*To expand and modernise the physical infrastructure and facilities of the university.*"

There is an ICT objective of KNUST which is:

"*To ensure the sustainable management of the university ICT resources through the creation of appropriate institutional framework.*"

(KNUST 2006)

This thesis lays a foundation for establishing a modern authentication system that is suitable, scalable and secure for the environment. The results of the research will help improve the management of KNUST's ICT resources by enforcing authentication security and accountability for using ICT resources. The results intend to propose a system that will improve existing issues in authentication and security, whiles conforming to KNUST's strategic and ICT objectives.

1.5 Methodology

The methodology used in this thesis involves review of literature as well as qualitative research methods. Structured interviews are used to acquire information about the IT infrastructure and applications used in KNUST. The steps used in performing the research are discussed in the chapter 3.

1.6 The scope of the study

This thesis is focused on proposing an appropriate architecture and concept of operations for a centralized authentication system in KNUST. The architecture was designed with functionality and feasibility of integration as the driving factors. This resulted in a design which considered user convenience, reliability and security. The actual implementation of the architecture is strongly recommended in the near future, but is not tackled in the current work.

1.7 Organization of the study

This thesis is organized into five chapters. Chapter 1 introduces the thesis domain, the problem statement, research questions and objectives. Chapter 2 presents literature review in SSO and KNUST's ICT objectives. Chapter 3 describes the steps used in conducting the research. Chapter 4 discusses the results of the study at KNUST and continues further to present the proposed architecture suitable to the KNUST situation. Also presented is the concept of operations and its related issues. Chapter 5 concludes the thesis and presents recommendations for future work.

1.8 Limitations of the study

The limitation in this study is the unavailability of current information in the area of the SSO architectures and operating mechanisms. The preferred source for such information would be the IEEE digital library also known is ieeexplore. The researcher, however, was not able to gain access to the library. A lot more could have been studied if access to ieeexplore was possible.

1.9 Summary

This chapter provided an introduction to the thesis topic, objectives and methodology. The justification for the thesis in the case study is proven. The chapter puts the reader in the domain of the thesis. The following chapter provides a review of literature in the thesis domain.

Chapter Two

Literature Review

2.0 Introduction

This chapter will proceed to understand in detail some pertinent topics related to the thesis domain. The previous chapter described the topics to be discussed here. The topics are central authentication scheme, single sign-on, and KNUST ICT status and direction. This compares SSO with the other major central authentication scheme - password management (PM). An explanation of SSO, its type's, benefits and threats is also found here. Finally, the draft KNUST ICT policy is reviewed to help define ICT strategy and objectives.

2.1 Central Authentication and Identity Management

Single Sign-On (SSO) and Password Management are two major authentication schemes that provide central administration. These concepts exist within the larger context of Identity Management architectures and systems. The Identity Management Framework comprises of other concepts such as Web Access Control, Delegated Administration, User Provisioning, Web-based user self-service, and Directory Services. SSO and PM have emerged due to important trends in corporate information technology.

- The increase and complexity of IT services has had an impact on the increase of user management problems.

- More and more business processes are depending on IT and this translates into more user accounts to be managed.
- Enterprises have evolved to include vendors, customers and suppliers as users into their information systems. This increases security risks and support costs.
- User management problems, increased user accounts and increased security risks have also increased data insecurity.
- Security and audit regulations such as Sarbanes-Oxley and HIPPA require that public companies monitor and control data access. They require that access to data is done on a person-by-person basis as well as function by function basis.

This has led the IT industry to develop a host of solutions that vary in strengths, weaknesses, architectures, pros and cons. These solutions include Single Sign-On (SSO) and Password Management (PM). (Nitai Alush-Aben 2005)

2.1.1 SSO vs. Password Management

Single sign-on systems allow users to log in once and have access to multiple applications without logging on to each application separately. This increases system usability and improves user efficiency. SSO systems have authentication severs that typically hold user credentials. These credentials are passed on to various applications providing the appropriate access rights. SSO will be reviewed in more detail in the next section.

Password Management systems enable users to log into multiple applications with a single username and password. They log on separately to each application, but with the same username and password. Users need to remember only one password which is synchronized across all applications. PM systems consist of a PM server which is responsible for creating, resetting and restoring passwords for users. Users interact with the server for these password operations and the server synchronizes the changes across all application servers. PM servers have a self-service feature which allows users to perform password changes, resets and unlocks without external assistance.

SSO and PM systems can both be used to help solve the problems associated with remembering multiple passwords. This sometimes makes it difficult to see the basic differences between the two. (Nitai Alush-Aben 2005)

SSO means authenticating only once. PM will use the same password to authenticate multiple times. Beyond solving password problems, SSO goes further than PM to provide greater administrative overhead in managing user authentication. It provides easier centralized management for user auditing, monitoring and control.

2.2 Single Sign On

'Single sign-on (SSO) is a form of technology that eases the authentication process for users and IT administrators.' (Search Security(a) 2008)

Whatis.com (What is Authentication? - A definition from whatis.com 2007), defines Authentication as the process of determining whether someone or something is, in fact, who or what it is declared to be. In private and public computer networks (including the Internet), authentication is commonly done through the use of logon passwords.

Authentication is not restricted to passwords. There are three factors of authentication. Something you know, such as a user ID and password; something you have, such as a smart card; and something you are, which refers to a physical characteristic, like a fingerprint that is verified using biometric technology. These factors can be used alone, or they can be combined in pairs to build a stronger authentication strategy in what is known as two-factor authentication. (Search Security(b) 2008)

When users try to access applications, they are prompted to enter authentication credentials. In a typical SSO system, application servers request for authentication credentials directly from an SSO authentication server. The server holds authentication credentials for each user. The server then passes the approved authentication credential to the application server that houses the application the user is requesting (Search Security(a) 2008).

2.2.1 Advantages and Disadvantages of SSO

There are many advantages of adopting SSO into authentication systems. The most distinct ones offered are:

- Users no longer need to remember multiple passwords in order to gain access to multiple applications and services. They only require a single password and are asked to enter it only once.
- SSO helps IT staff reduce the cost of managing an endless number of passwords. Help desk costs are reduced as a result of fewer users calling in for password resets and other related password issues.

- SSO facilitates managing of audit logs and monitoring of user accounts. Administrative tasks such as tracking user activities and removing inactive accounts are easier to manage. This improves organisational security and is in accordance with the requirements of the Sarbanes-Oxley Act (SOX). (Search Security(a) 2008)

Some of the disadvantages of SSO are:

- Weak passwords in a single sign-on environment can pose a significant risk. If a user's password is compromised it could result in unauthorized access to protected resources and information. This is evident if passwords are the primary authenticating mechanisms.
- SSO reduces the need to remember many passwords but still does not stop users from forgetting passwords. This is the case because of complex password policies which try to stop users from using simple passwords. Such password policies involve password length, composition and life spans.
- SSO systems are, most of the time, costly to implement. This is due to the requirement of new servers and hardware necessary for the implementation of the system.

The above disadvantages presented by SSO are normally taken into account when developing SSO systems. Password policies are incorporated into the authentication mechanism of the system to eliminate weak password creation. In addition alternate types of authentication can be used instead of or in conjunction with passwords to assist users with memory issues. Various SSO architectures have many different requirements. Each has different cost, security and adaptability implications. The balance of these is normally a trade-off that is mostly affected by resources available for the SSO project.

2.2.2 SSO Services

There are three types of SSO services available today as described by the Microsoft Development Network. (Understanding Enterprise SSO 2009) These are Windows Integrated Single Sign-On, Extranet Single Sign-On (Web SSO), and Server-Based Intranet Single Sign-On.

Windows integrated single sign-on allows users to access multiple applications within a network that uses a common authentication mechanism. SSO of this nature verifies a user's credentials after he/she logs on to the network. The credentials indicate the users access rights. Users can use any resource as long as it has the common authentication mechanism. Kerberos is such an authentication mechanism. (Understanding Enterprise SSO 2009)

Extranet Single Sign-On which is more commonly known as WebSSO is web related. This service allows users to use a single log in to gain access to multiple resources over the internet. With a single set of credentials the user can access multiple websites owned by different companies. An example of this SSO service is the Microsoft Passport Network for consumer based – applications. (Understanding Enterprise SSO 2009)

Server-Based Intranet Single Sign-On is a type of service that allows single sign on across different heterogeneous systems across an enterprise network. These systems may use different authentication mechanisms and have their own user directory. In this service users can use a single set of credentials to access resources from both windows and non-windows based domains. (Understanding Enterprise SSO 2009)

2.2.3 SSO Operating Architectures

There are several SSO solutions that have been developed by industry and academia. These can be classified by two main categories. These are SSO solutions with a single set of credentials and those with multiple credentials (Bui 2005), (Clercq 2002). Solutions with a single set of credentials manage one common authentication mechanism during the user authentication process. This solution is generally used in network domain environments where the IT systems will generally use the same authentication mechanism. SSO with multiple credentials on the other hand would operate across two or more domains, with each domain requiring a different authentication mechanism. SSO with a multiple credential set can either be client side caching or server side caching. This describes the location of the credential cache during SSO.

In this literature review, the researcher only presented SSO solutions with a single set of credentials. This is because the case study institution has a single domain structure and a single authentication mechanism. Additional information about SSO solutions with a multiple set of credentials can be found in (Clercq 2002).

These SSO categories all use one form of operation or the other. For a single set of credentials, the operating architectures that were reviewed were token-based, PKI (public key infrastructure)-based, proxy-based, identity provider redirection and database replication architectures. (Bui 2005) (Clercq 2002).

2.2.3.1 Token-Based SSO

In this SSO, users receive cryptographic tokens from the authentication server after successful authentication. When the user requests an application from an application server, the token is used as proof of identity. The application server performs cryptographic processing on the token to verify that the token is valid and the user is authentic. Token SSO relies on shared keys that represent a trust between application servers and the authentication server. For added security some tokens are time-stamped with passwords. Synchronized clocks from a time server allow synchronized time. At predetermined intervals the token generates new passwords that are unique to the token and accepted only within a given time window. An example of token-based SSO is the Kerberos authentication protocol which uses symmetric cryptography to provide authentication for client-server applications. Symmetric cryptography uses the same encryption key between the client and the server application. (Bui 2005) (Clercq 2002) (Tech-Faq 2009).

2.2.3.2 PKI- Based SSO

Public Key Infrastructure (PKI) based SSO uses certification authorities (CA) for user registration. During the registration process, users present credentials for proof of identity. The CA generates public – private key pairs, and creates certificates using the user's public key. The user generates tokens using his private key and certificate. The token is then used for authentication and SSO. The difference between PKI and Token based SSO is that PKI uses asymmetric cryptography and Token uses symmetric cryptography. PKI also has a unique registration process. Some SSO solutions combine both Token and PKI based SSO. (Bui 2005) (Clercq 2002).

2.2.3.3 Proxy-Based SSO

In proxy based SSO the authentication server acts as a proxy for the subsequent sign on processes. The user initially authenticates to the central authentication server. When a request is made to an application server by the user, the authentication server itself provides the application server with the appropriate user credentials for that user. Proxy SSO is popular when servers have different authentication mechanisms. The users will have multiple credentials. The authentication server (proxy server) manages all credentials for all users in a database. Proxy SSO can be used for both single credential and multiple credential solutions. Proxy SSO uses little modification to the end systems of a network to enable SSO. (Bui 2005) (Clercq 2002).

2.2.3.4 Database Replication

This form of SSO operation is the simplest. When users authenticate to the central authentication server, the users logged in information is stored in a database. All currently logged in users are stored in this database (session database). The database is then broadcasted to all servers. When a user wishes to request an application, the server authenticates the client with its copy of the session database. This creates a master slave relationship between the central authentication server database and the application server's databases. (Bui 2005) (Clercq 2002).

2.2.3.5 *Identity-Provider Redirection*

This SSO type is used over the internet to access resources that are located on websites in different domains. Any requests made to resources on a website (via web browser) are redirected to an identity provider. The user is authenticated to the identity provider which returns authentication information back to the web browser. This is normally in the form of a browser cookie. The browser is then directed back to the website with access to the resources according to the browser cookie. Microsoft Passport is a well known example of identity-provider SSO. (Bui 2005) (Clercq 2002).

2.2.3.6 *Comparison*

The above architectures are compared according to scalability, implementation requirements, potential security risks, security benefits and bottlenecks. (Bui 2005)

Scalability

This looks at the ability of the architecture to accommodate user expansion. All the architectures can increase the number of authentication servers to handle increases in user requests. Database replication can implement more servers to augment user load. The servers will still be seen as slaves to the master server. Token based and PKI can also adopt authentication server replication. However, there should be a master server to maintain consistency. PKI will adopt a master CA that will verify the certificates issued by other CA's. Identity-provider redirection can provide multiple servers manage user load. This also needs a master server for consistency. (Bui 2005)

Implementation Requirements

To use token based SSO, both clients and servers need to be able to use tokens, before the application can be used. A homogeneous environment that uses the token is necessary. As such additional code is required at application servers and clients. PKI also requires conformity to the use of certificates by both clients and servers. Database replication requires that all servers use the same authenticating format. With proxy based SSO clients and server need very minimal if any modification at all. Identity-provider redirection clients require standard web browsers. The services and applications being provided need to be web based. (Bui 2005)

Security Risks

Token based SSO functions by caching tokens on client machines. Unsecure clients can lead to stolen or reused tokens. With time-stamped tokens, insecure server synchronizing can introduce rogue servers in the network. If rogue CA's are present in the network, revoked certificates can be reused. Proxy based and identity-provider redirections are susceptible to man-in-the-middle attacks. This happens when an attacker poses as the proxy server or redirection server and gain access to the user's username and password. If databases are not updated adequately in Database replication, the outdated information may lead to unauthorized access. For all the architectures, compromise of the authentication server is a genuine threat. (Bui 2005)

Security Benefits

Database replication allows a failsafe authentication database due to the presence of slave servers. A slave server can replace the authenticating sever in the case of a failure. This failsafe is a benefit for all architectures that employ multiple authentication servers. That can be for proxy, token, pki and identity-provider type SSO's. Time-stamped tokens provide increased security against compromised tokens. PKI based SSO allows direct client server authentication through registration and examination of server and client certificates. Security benefits in proxy and identity-provider redirection depend on the security mechanism being used at the authenticating server. (Bui 2005)

Bottlenecks

This relates to the performance of the architectures by possible points of failure within the architectures. Database replication has a performance bottleneck based on the frequency of authentication information. The slaves are dependent on the ability of the master sever to provide them with updated information. The bottleneck in token based SSO is the authentication server. This server responds to each user request for each application. The bottleneck of the PKI based architecture is the maintenance of certificates. That is checking for expired and revoked certificates. This is a general performance of PKI. The proxy authentication server and the identity provider are the bottlenecks in proxy SSO and identity-provider redirection SSO respectively, just like for token based SSO. (Bui 2005)

2.2.4 Client Side and Server Side Architectures

This classification of SSO architecture is proposed by Ufinity. (Ufinity 2003) The aim of the SSO architecture is to propagate user identity and sessions across heterogeneous severs. Four major classes of applications that reflect the evolution of applications are considered in achieving this aim. The applications are dumb terminal, client-server, web browser, and micro-browser. Under this classification we examine client-side SSO, server-side SSO and Hybrids.

2.2.4.1 Client-Side SSO

This involves automating the keystrokes of a user's logging in process. Login profile scripts need to be defined to replay the login procedure. As a benefit capable clients can log into almost any system. The disadvantage however is that such scripts break easily when rigid assumptions change. Client-side SSO can be either:

- Form-filling client with local credential store or
- Form-filling client with central credential store

Form-filling client with local credential store

Form-filling agents are installed on the user's computers. The agents are in constant operation. These agents monitor the computer for any application login prompts. Upon detection the agent retrieves the necessary passwords and logs in the user. User credentials are stored locally on the hard disk or within smartcards. Users may need to keep one credential for each application being used. No central administration is possible in this architecture and applications need to manage their own access control registries.

This architecture suits dumb terminal and client server applications. This action is described in Figure 2.2-1

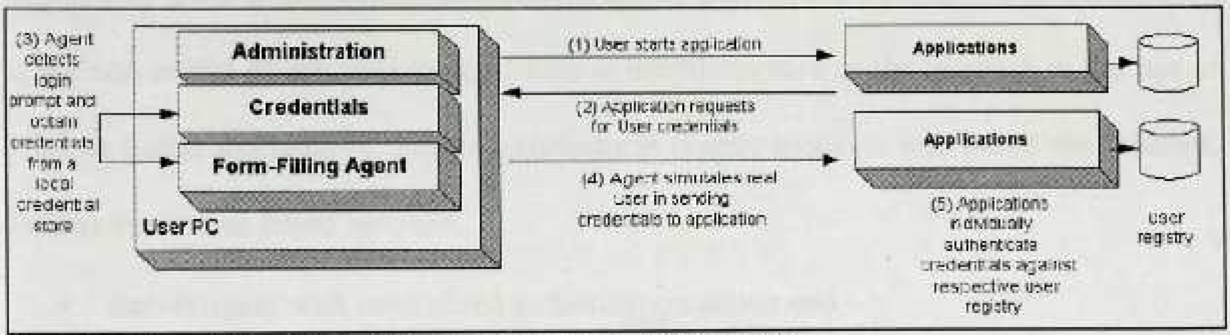
(3) Agent detects login prompt and obtain credentials from a local credential store
Administration
Credentials
Form-Filling Agent
User PC
(1) User starts application
(2) Application requests for User credentials
(4) Agent simulates real user in sending credentials to application
Applications
Applications
(5) Applications individually authenticate credentials against respective user registry
user registry

Figure 2.2-1 Form filling client with local credential store. (Ufinity 2003)

Form-filling client with central credential store

This architecture is an improvement of the above. It provides a better management system by providing central credential control. The form filling agents retrieve the user's control from a central credential server instead of a local store. This is represented in Figure 2.2-2

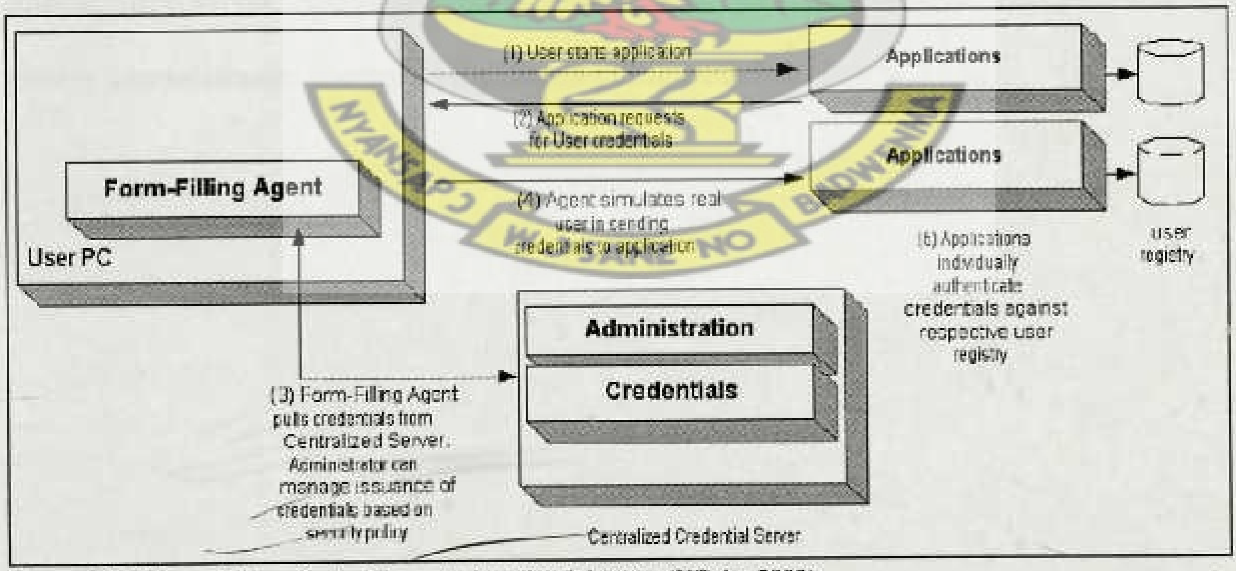
(1) User starts application
(2) Application requests for User credentials
Form-Filling Agent
(4) Agent simulates real user in sending credentials to application
User PC
Applications
Applications
(6) Applications individually authenticate credentials against respective user registry
user registry
Administration
Credentials
(3) Form-Filling Agent pulls credentials from Centralized Server. Administrator can manage issuance of credentials based on security policy
Centralized Credential Server

Figure 2.2-2 Form filling client with central credential store. (Ufinity 2003)

2.2.4.2 Server-Side SSO

Server-side SSO architectures are concerned with performing authentication verification at the server end of the SSO system. This is useful because of the increase in the use of browser based applications. Server architectures mainly focus on web based applications. Two architectures found here are:

- Server agent with centralized authorization server and
- Reverse proxy with central authorization server

Server agent with centralized authorization server

This architecture uses agents that are installed on the application servers. There are no agents present on the client computers. The user first logs into the central authorization server. The server then issues a session credential for the user's browser. The credential allows the user to SSO into all application servers based on the server agent. In this architecture the user registry for application servers is managed centrally by the authorization server. Application servers need not perform this function. This provides greater administrative control as illustrated in Figure 2.2-3.

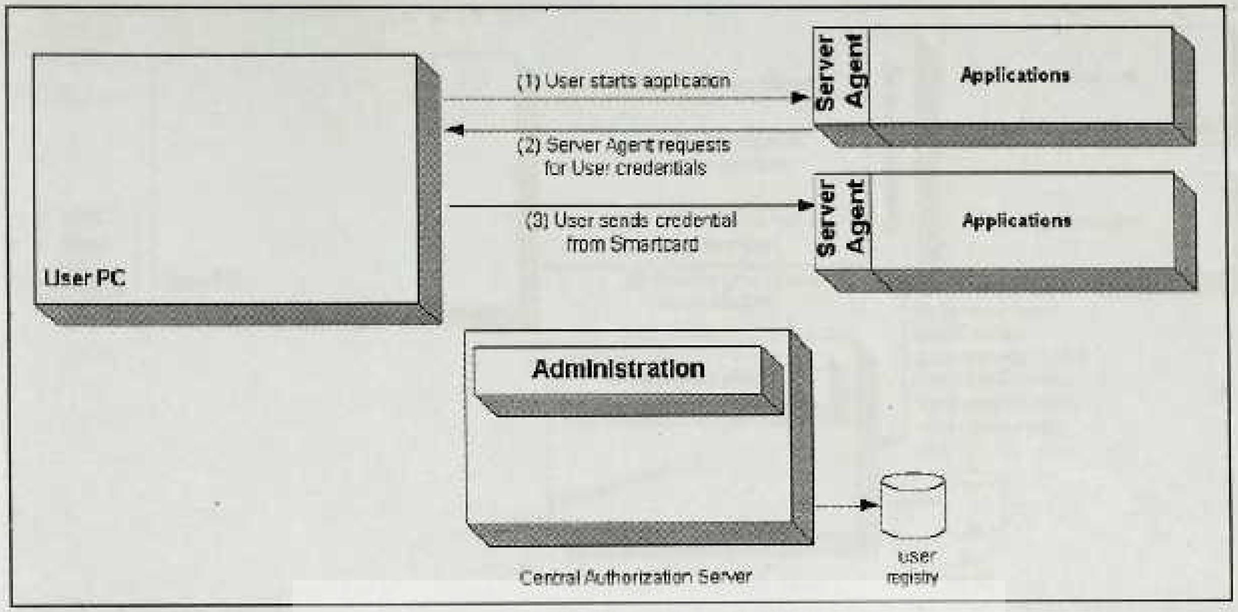
User PC
(1) User starts application
(2) Server Agent requests for User credentials
(3) User sends credential from Smartcard
Server Agent
Applications
Server Agent
Applications
Administration
Central Authorization Server
user registry

Figure 2.2-3 Server agent with centralized authorization server. (Ufinity 2003)

Reverse proxy with central authorization server

This type of architecture is a variation of the server agent architecture. A special server which is referred to as a reverse proxy server monitors and traps any HTTP request made to the application servers. No server agents are installed on the application servers. The requests are allowed only if they have a valid session credential. The reverse proxy architecture is shown in Figure 2.2-4.

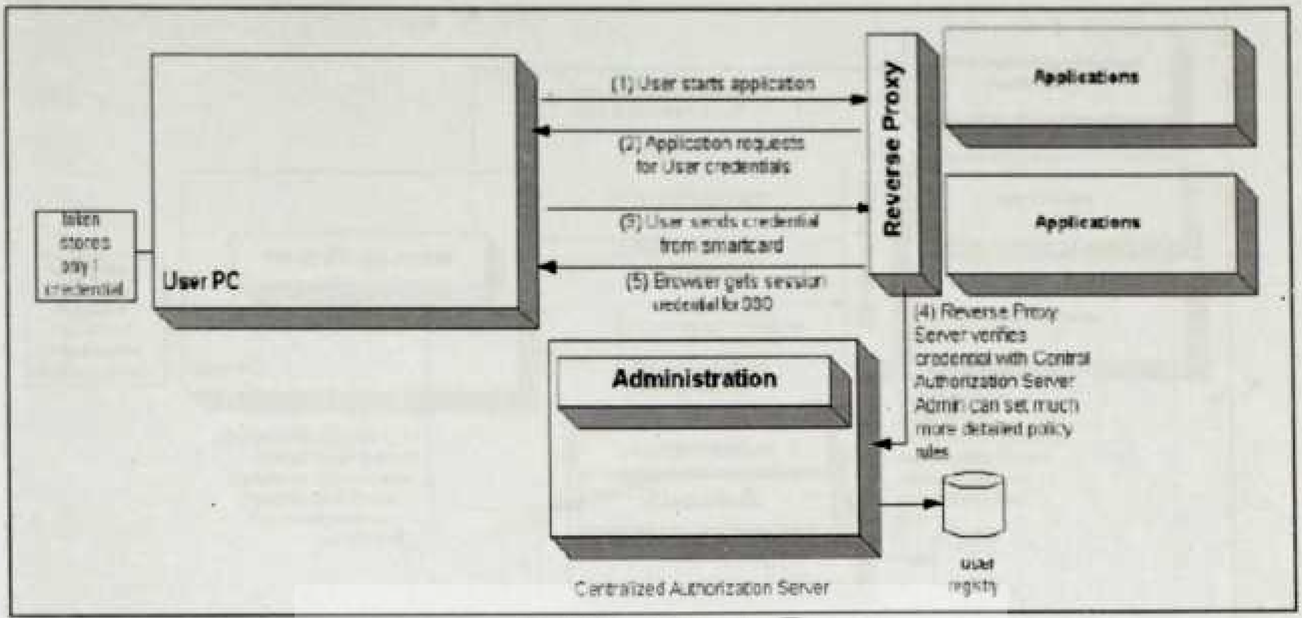
(1) User starts application
(2) Application requests for User credentials
(3) User sends credential from smartcard
(5) Browser gets session credential for SSO
Reverse Proxy
Applications
Applications
token stores only 1 credential
User PC
(4) Reverse Proxy Server verifies credential with Central Authorization Server Admin can set much more detailed policy rules
Administration
User registry
Centralized Authorization Server

Figure 2.2-4 Reverse Proxy with central authorization server. (Ufinity 2003)

2.2.4.3 *Hybrids*

There are possible combinations of client and server side SSO architectures. These hybrid architectures increase performance of the SSO. It also improves the deployment strategy for enterprises. An example of hybrid architecture is one between the form-filling client with central credential store and the reverse proxy with central authorization server. The form-filling client performs authentications for traditional client-server applications. The central authentication server provides credentials for the client-server applications. The credentials can be cached by the client. Web applications are controlled by the reverse proxy server. The server monitors http requests and verifies credentials with the central authorization server. Figure 2.2-5 shows the hybrid architecture described above.

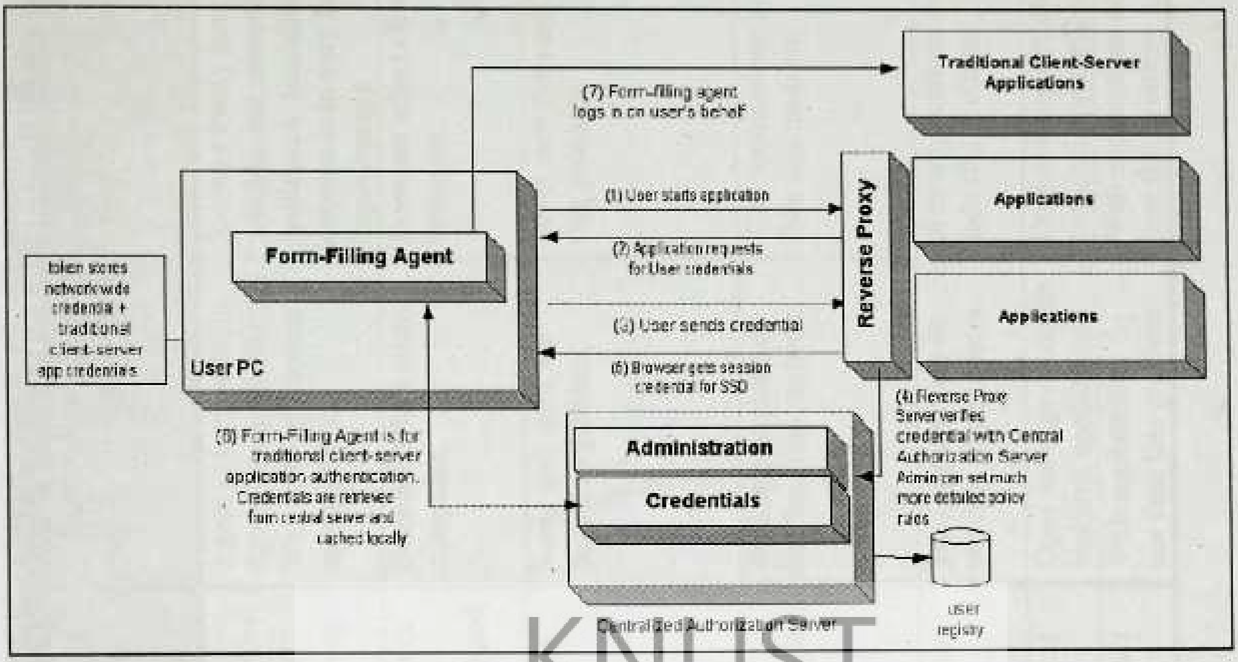
Traditional Client-Server Applications
(7) Form-filling agent logs in on user's behalf
(1) User starts application
(2) Application requests for User credentials
(3) User sends credential
(6) Browser gets session credential for SSO
Reverse Proxy
Applications
Applications
Form-Filling Agent
token stores networkwide credential + traditional client-server app credentials
User PC
(5) Form-Filling Agent is for traditional client-server application authentication. Credentials are retrieved from central server and cached locally
Administration
Credentials
(4) Reverse Proxy Server verifies credential with Central Authorization Server Admin can set much more detailed policy rules
Centralized Authorization Server
user registry

Figure 2.2-5 Hybrid Architecture. (Ufinity 2003)

2.2.4.4 Comparisons

A comparison of the client-side, server-side, and hybrid architectures are compared in Table 2.2-1

CLIENT-SIDE SSO		HYBRID	SERVER-SIDE SSO	
Form-filling agent with local credential store	Form-filling agent with central credential store	Hybrid Agent Client side: Traditional Apps Server side: Web Apps	Server Agent with Central Authorization Sever	Reverse Proxy with Central Authorization Server
Agent is not 100% reliable in detecting login prompts. Slow computers increase this problem.	Same as with client side problem.	Reliability problem is reduced minimally. Traditional apps do not change as quickly as web applications.	Agent is very reliable. Agent is scalable for Web SSO and access management.	Reverse Proxy is very reliable, but presents a bottleneck. Scalability is expensive in very large deployments.
Computers will slow down due to continuously detecting login events and 'spying' on keystrokes.	Same as with client side problem.	Reduced problem than in client-side. FF agent monitors specific apps and not all web access.	Does not affect client computer.	Does not affect client computer.
Login scripts are created for each web application. If a user uses multiple browsers, he will need multiple login profiles for each browser. Also some browser upgrades will require script upgrades.	Same as with client side problem.	Same as with client side problem.	Sever side does not have this problem.	Sever side does not have this problem.
When application are changed or modified login scripts are also changed. 'Screen scraping' with form filling is highly intolerant to changes in login scripts. This can disrupt the SSO.	Same as with client side problem.	Problem is lessened compared to general client side.	Web applications do not face this problem.	Web applications do not face this problem.
Agent needs to be configured according to password renewal and expiration policies for each application.	Renewal and expiration can be done by central credential store.	Web applications do not face this problem	Web applications do not face this problem	Web applications do not face this problem
Supports dumb terminal applications	Supports dumb terminal applications	Supports dumb terminal applications	Does not support dumb terminal applications	Does not support dumb terminal applications
Maintaining and managing tokens can be expensive	Token replacement is easier through server.	Web applications do not face this problem	Web applications do not face this problem	Web applications do not face this problem

Table 2.2-1 Client-side, Server-side, Hybrid architecture comparison

2.3 KNUST Background and Composition

The Kwame Nkrumah University of Science and Technology (KNUST) was established in 1951. It was then known as the Kumasi Institute of Technology. KNUST was founded to provide tertiary education specifically in the area of science and technology. KNUST had the objective of being a catalyst for technological development in the country.

The academic activities are performed by six colleges, which are, Agriculture and Natural Resources, Architecture and Planning, Art and Social Sciences, Engineering, Health Sciences, and Science. Other administrative functions are performed by other units such as the main administration, finance office, planning unit public relations office. KNUST has a student population of about 24,000 and a staff population of over 700. Amongst the staff are 16 personnel who constitute the IT department known as University Information Technology Services (UITS). (KNUST 2006)

2.3.1 KNUST ICT objectives

The KNUST ICT policy and development plan is meant to contribute to the realisation of national standards in ICT. KNUST as a result has a set of ICT objectives that will achieve this goal. In relation to the thesis domain are the following objectives:

- "*to extend ICT services to all Units of the University*" (KNUST 2006)

  With the provision of more services there will be more users and more accounts to manage. Ease of authentication and management will allow this objective to be facilitated.

- "*To ensure sustainable management of the university's ICT resources through the creation of appropriate institutional framework.*" (KNUST 2006)

  Planning and eventually implementing an authentication system that can be administered centrally will enforce this policy to the letter.

2.3.2 KNUST ICT setup and projection

KNUST has a Local Area Network that extends across 90 percent of the campus. This LAN allows for the distribution of non-centralized applications across the entire university. A fibre-optic backbone connects multiple network segments together through a TCP/IP network. The network operating centre (NOC) is at the centre of the network housing most of the network infrastructure and distributing common services to the various segments of the LAN. The segments which are made up of colleges and other units have sub networks which serve other internal applications to users.

According to the KNUST ICT policy KNUST aims to integrate the relevant university administrative functional units into an enterprise system. This system will automate the University administrative functions to provide an effective and efficient management process. There are a number of information systems that are seen as relevant administrative functional systems which must be part of the proposed enterprise system.

These are:

- Academic Record Management Information System (ARMIS)
- Library Management Information System (LMIS)

- Human Resources Management Information System (HRMIS)
- Financial Management Information System (FMIS)
- Hospital Management Information System (HMIS)
- Project Management Information System (PMIS)
- Security Management Information System (SMIS)
- Decision Support System (DSS)

(KNUST 2006)

2.3.3 KNUST Make or Buy Policy

KNUST has a make or buy policy which states that university management will decide whether or not an application should be developed 'in-house' or acquired from external sources based on the following key considerations:

In-House Development criteria

- *A customised ICT application or service that is totally responsive to the institution's very specific needs.*
- *Increased ease in developing software due to the growth of Rapid Application Development tools and systems.*
- *Ease of adapting software to rapidly changing user needs without having to co-ordinate the requirements with vendors.*
- *Developing professional competence in software development.*

External Source Acquisition Criteria

- *Ability to gain access to specialised skills that cannot be retained or for which there is insufficient need to have continuously available.*
- *Cost. Building software is still extremely costly.*
- *Staff utilization.*
- *Ability to make short-term commitment for ICT development support instead of having to make major investment in staff recruitment and professional training.*

(KNUST 2006)

2.4 Summary

This chapter presented an overview of single-sign-on concepts and architectures according to operating architectures and client/server side architectures. A basic review of password management was made in comparison to SSO as central authentication systems. Relevant factors of the KNUST ICT policy were looked at. The chapter presented the research with necessary background theory and concepts required to suggest and design suitable authentication architecture for KNUST.

Chapter Three

Research Methodology

3.0 Introduction

This chapter will explain the process by which the research was done. It provides the necessary direction for others who would want to perform similar research on the topic of the thesis. The sampling procedure used to collect primary data was explained as well the data types used within the research.

3.1 Data Collection

The data collected was both primary and secondary in nature. Internet research was used to collect secondary data on issues required to understand the SSO concept. Data was also collected to determine KNUST's IT strategy and objectives. This data was collected and reviewed to facilitate the selection and development of the authentication architecture. Primary data was collected from KNUST's IT personnel by means of a structured interview. The interview was guided by specific questions in order to focus on specific topics. Interviews were used to ensure each question was completely understood by the interviewee. Any doubts concerning the questions were clarified by the researcher before the answers were given. Average time for each interview was 15 to 20 minutes. The interview intended to allow healthy discussions into the areas being investigated. Provision was made for the interviewees to make additional comments into matters arising from software and application administration.

3.2 Sampling Procedure

The sampling population was taken to be the ICT personnel within KNUST. They represent the specific members of staff who have the required information about the case study institution. The ICT personnel have three main work areas. These are network operations, web applications development and other applications development. The network division has the most members. They have 10 personnel who are tasked with network administration within the main operating centre, college operating centres and the university ICT centre. They also control network operations within other areas such as the administration. The software development division has four members. These develop client/server applications which are being used within the colleges and other units of the university. The web team has only two members and control the university web site and web related applications. Representatives from the web applications, software development, and each deployment area for the network divisions constituted the sampling units for the study. The objective of interacting with this sample population was to get relevant information about the types of information services and applications running in KNUST that currently require authentication. The total population of ICT personnel is 19 and out of this, 12 personnel were selected across the three divisions and network deployment areas. The distribution by IT division is shown in the Table 3.2-1.

ICT Division	ICT Staff Population	ICT Staff Interviewed
Network Operations Centre	3	1
Colleges	6	5
ICT Centre	2	1
Administration & Other Units	2	2
Software Development	4	2
Web Development	2	1

Table 3.2-1 Sampling of KNUST ICT Staff

3.3 Research Design

The methodology is made up of a series of 6 interrelated steps that were used to conduct the research study. These steps are explained below:

Step 1

Step 1 is a review of existing literature. This step is further divided into sub steps which are aimed at understanding the theoretical concepts associated with the research study.

Step1.1 This sub-step aims to study central authentication mechanisms. This will compare methods of central authentications under the larger context of identity management.

Step1.2 This next sub-step will investigate available literature on the case study institution (KNUST) to determine current and future IT policies and security objectives.

Step1.3 This sub-step looks into the central authentication methods and singles out one considered to be most suitable to the situation of the case study institution. This method is then studied in detail.

Step 2

Step 2 is made up of a structured interview of ICT personnel of the case study institution (Kwame Nkrumah University of Science and Technology). The interview is to be conducted with structured questions that elicits responses to give an insight into the current ICT situation in the institution. The interview will help to identify some of the requirements necessary in performing step 3.

The following are the objectives of the interview;

- Collect primary data on the information services in the institution.
- Collect primary data on frequency of authentication errors.
- Obtain a brief idea on state of SSO in the institution.
- Investigate the types of authentication in the institution (password, others)
- Find out the capacity of the information services to perform user auditing
- Find out any other useful information to help in the design of the authentication architecture.

Step 3

Design of a conceptual model: The design will be based on existing architectures from literature. The model will be suited to the case study institution according to the data collected in steps 1 and 2.

Step 4

Perform model analysis in order to develop a concept of operations and some failsafe mechanisms in the event of failure. The analysis will generate the concept of operations and failure scenarios.

Step 5

Formulate assumptions made for the model (if any). In addition, advantages and disadvantages of the model are formulated based on analysis in step 4.

Step 6

Figure out a recommendation for future work (if necessary).

3.4 Summary

This Chapter provides the reader with a description of the steps used to carry out the research. The order of relevance for each step was indicated. The next chapter proceeds to discuss the results of steps 2, 3, 4 and 5 in the methodology above.

Chapter Four

Results and Discussion

4.0 Introduction

This chapter presents the responses of the interviewees. The responses analyzed and presented according to KNUST's ICT setup, applications and services provided, and the requirements of the proposed system.

4.1 Interview Results

The following results were received from the 12 interviewees.

Question 1

All the respondents answered positively to the first question, indicating that they understood the concept of central authentication. This helped introduce the respondents to the area and prepared them for the question to follow.

Question 2

None of the respondents had heard of SSO as a terminology. Two of them, however identified the concept as existent in foreign university structure.

Question 3

All ~~the re~~spondents confirmed that SSO is not currently implemented in KNUST.

Question 4

Eleven respondents (91.67%) identified the username and password mechanism as the main authentication mechanism within KNUST. One person indicated that smartcards (swipe cards) were also used. They are used for physical authentication into the premises of the KNUST Network Operating Centre.

Question 5

Each of the respondents identified applications and services available within KNUST. Most respondents had a decent awareness for applications that existed outside their specific job functions. The applications and services with their classification (proposed by Ufinity 2003) are shown in Figure 4.1-1.

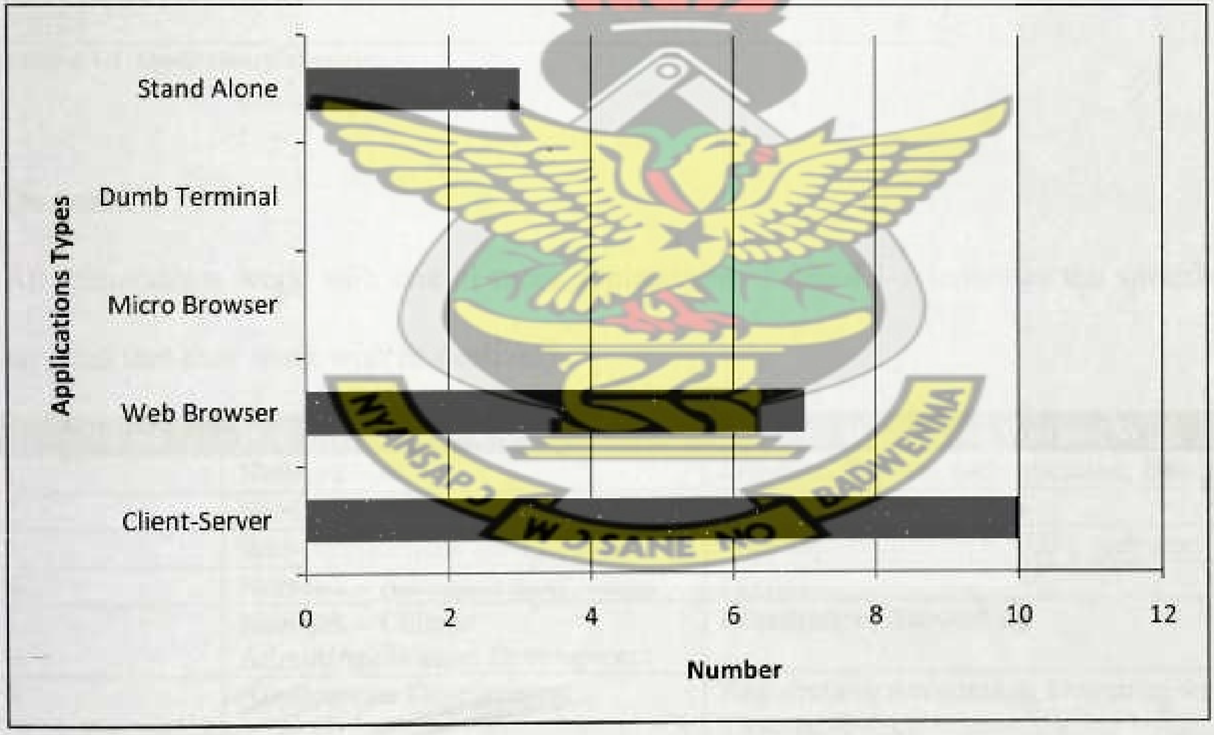

Figure 4.1-1 Application Types

The client-server applications are clearly in the majority.

The Specific Applications and their types are shown in Table 4.1-1.

Application Name	Application Type
1 Registration (Interbase)	Client - Server
2 Examination (Interbase)	Client - Server
3 Accounting (Interbase)	Client - Server
4 Library (Interbase)	Client - Server
5 Timetable (Interbase)	Client - Server
6 Registration (Sql)	Client - Server
7 Examination (Sql)	Client - Server
8 Accounting (Sql)	Client - Server
9 Time-Table (Sql)	Client - Server
10 Human Resource (Interbase)	Client - Server
11 Payroll (Dbase)	Stand Alone
12 Procurement	Stand Alone
13 Admissions	Stand Alone
14 E – Learning	Web Browser
15 E- Mail	Web Browser
16 Internet Browsing	Web Browser
17 Staff Profile	Web Browser
18 Student Forums	Web Browser
19 D Space	Web Browser
20 Online Educational Resource (OER)	Web Browser

Table 4.1-1 Application Categories

Question 6

All respondents work with one or more applications. Table 4.1-2 indicates the specific services that they work with respectively.

Respondent	ICT Division	Services/Application Worked With
1	Network – NOC	Wireless, Domain authentication, Email
2	Network – Administration Admin	Human Resource system
3	Web Applications Development	Web applications on KNUST web site
4	Network – Administration Admin	Payroll
5	Network – College Admin/Application Development	Registration, Accounting
6	Applications Development	Registration, Accounting, Examination, Time-table
7	Applications Development	Registration, Examination, Library
8	Network – ICT Centre	Student Access Control System
9	Network – College Admin	Email, applications and services accessibility.
10	Network – College Admin	Email, applications and services accessibility.

11	Network – College Admin	Email, applications and services accessibility.
12	Network – College Admin	Email, applications and services accessibility.

Table 4.1-2 Respondent Application Areas

Question 7

Some respondents work in the capacity to perform authentication management within the various applications and services. Table 4.1-3 shows the respondents who perform authentication management and the applications they manage.

Respondent	Authentication Management?	Management Description
1	yes	Email and Wireless Access Control
2	yes	HR system Access Control
3	yes	Web Applications Access Control
4	no	-
5	yes	Registration and Accounting Software(sql applications). Access Control
6	yes	Registration and Accounting Software(interbase applications). Access Control
7	yes	Examination and Library Software(interbase applications). Access Control.
8	yes	Access Control and user management
9	no	-
10	no	-
11	no	-
12	no	-

Table 4.1-3 Respondent - Authentication Management

Access Control in the table above refers to the creation of users, allocation of permissions and troubleshooting access related problems. The respondents who answered no to the above question do not manage authentication within the applications they work with. They however respond to user problems and forward the issue to the appropriate administrator.

Question 8

The frequency of encountering problems is shown in Figure 4.1-2.

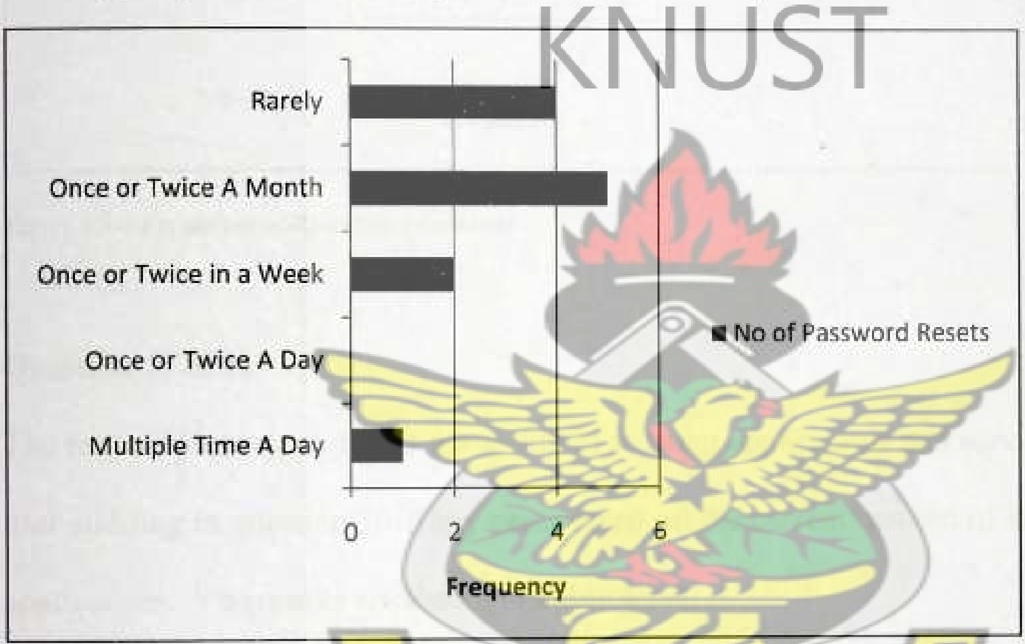

Figure 4.1-2 Frequency of Encountering Problems with User Authentication

Question 9

The frequency of resetting passwords (question 9) is shown in Figure 4.1-3.

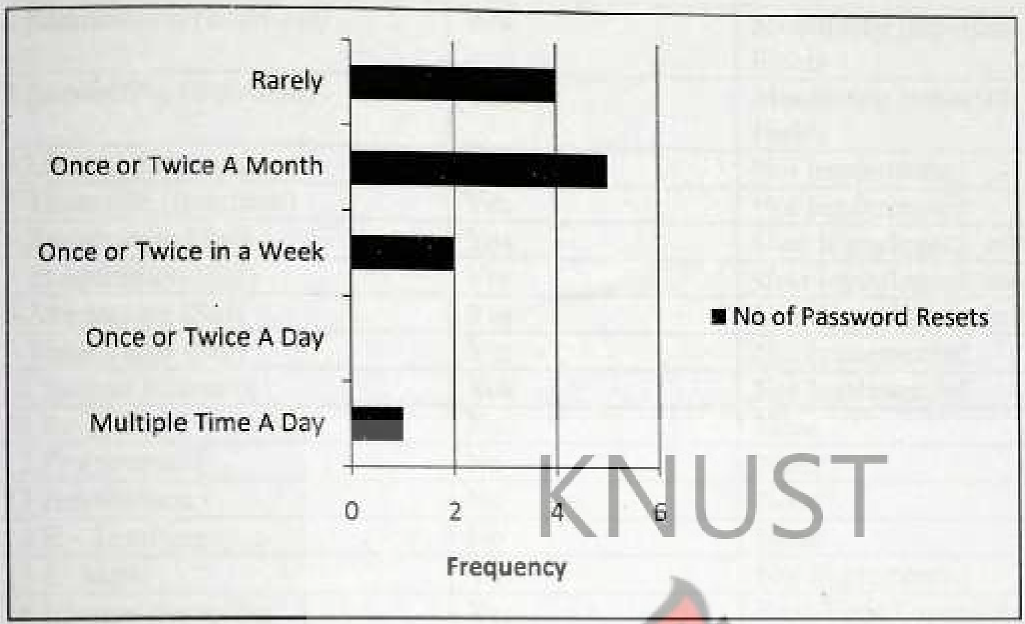

Figure 4.1-3 Frequency of Resetting Passwords

Question 10 & 11

The respondents responded to the ability of existing applications and services to provide user auditing in question 10. They commented on the current method of auditing for the applications. The results are shown in Table 4.1-4.

Application	Capability of Auditing	Current Auditing Implementation
1 Registration (Interbase)	Yes	Monitoring Important Database Fields
2 Examination (Interbase)	Yes	Monitoring Important Database Fields
3 Accounting (Interbase)	Yes	Monitoring Important Database Fields
4 Library (Interbase)	Yes	Not Implemented
5 Timetable (Interbase)	Yes	Not Implemented
6 Registration (Sql)	Yes	User login/logout information
7 Examination (Sql)	Yes	User login/logout information
8 Accounting (Sql)	Yes	User login/logout information
9 Time-Table (Sql)	Yes	Not Implemented
10 Human Resource	Yes	Not Implemented
11 Payroll	No	None
12 Procurement	No	None
13 Admissions	No	None
14 E – Learning	No	None
15 E- Mail	Yes	Not Implemented
16 Internet Browsing	Yes	Real-Time Capability, however no report generation
17 Staff Profile	Yes	Time of visit
18 Student Forums	No	None
19 D Space	Yes	Time of visit
20 Online Educational Resource (OER)	No	None

Table 4.1-4 Application Auditing Capability

Question 12

The respondents provided additional comments of relevance to the applications and services within KNUST. These are enumerated below.

- The use of usernames and passwords for authentication is not a secure mechanism in the institution. Many users have weak and insecure passwords. There have been many incidents where 'hackers' have infiltrated email accounts and impersonated the users, within the organization.
- The interbase database software which is used to develop some of the client-server applications recommends minimal log creation. It explains within help

files, that extensive log creation for all tables in a database demands a large storage space.

4.2 General ICT Setup

The ICT staff that were interviewed provided necessary insight into the existing systems and infrastructure in KNUST. The KNUST ICT setup consists of a network operating centre (NOC) that serves as the main control room and heart of the university network. The centre houses the core of the university's ICT infrastructure. The NOC also contains servers that host the commonly used internet and email services. Other applications that are used by multiple users within the university are stored in application servers which are found in other segments of the local area network (LAN). These LAN segments logically form sub-networks for colleges and other administrative units. Each LAN segment has a miniature network operating centre which houses a domain controller and an internet proxy server. The internet proxy server is used for bandwidth management and does not require authentication.

4.3 Applications and Services

A basic look at the type of applications being used within the university is presented in Table 4.3-1. All the applications and services listed in the table were developed by staff of the University Information Technology Service (UITS). This is with the exception of the internet and email services provided by the NOC and the e-learning services found under the Administration and other units.

KNUST Division	Applications/Services Provided	Application Types
College of Engineering	Registration, Examinations, Time-table, Accounting, Library management. Email and Internet from the NOC.	Client – Server and web applications
College of Science	Accounting, registration, exams processing. Email and Internet from the NOC.	Client – Server and web applications
College of Arts and Social Sciences	Uses application from engineering. Email and Internet from the NOC.	Client – Server and web applications
College of Agriculture and Natural Resources	Uses applications from engineering. Email and Internet from the NOC.	Client - Server and web applications
College of Architecture and Planning	Uses applications from engineering. Email and Internet from the NOC.	Client – Server and web applications
College of Health Sciences	Same as College of Science	Client – Server and web applications
Administration and other Units	Human resource, Payroll, Procurement, Admissions, E-Learning Facility. Some Legacy Applications.	Client – Server, Web applications and stand-alone applications
Network Operating Centre	Internet Service, Email, Staff profile, Student forums, Dspace application, OER(online educational resource)	Web Applications
ICT Centre	Internet Services	Web Application

Table 4.3-1 KNUST ICT Applications and Services

The college of engineering applications used by engineering were developed using Interbase. These applications have a user registry which authenticates and authorizes users with varying levels of access. Other colleges who use this service are added into the user registry. The user registry is within the interbase application.

The college of science applications were developed on the Microsoft SQL platform. The science applications also have their own user registry as an SQL database. The web applications are developed with PHP and Java (JSP). The user registry used for authenticating users is located in an SQL database which is separate from the application.

The email system was not developed in-house. It runs on software called Zimbra. The zimbra user registry uses an open LDAP database.

From the information revealed here, it is clear that applications that fall under the client-server and web categories are in the majority. There are however a few legacy applications also still running in KNUST. Some of these were developed in-house some years ago with programming languages such as DBASE.

4.4 Authentication and Security

The only form of authentication used for applications within KNUST is the username and password system. This is a security concern within the institution. A KNUST network administrator who administers user accounts within the domain and email systems reported of unsecure passwords used within these IT systems. Most users do not yet require domain authentication to access the user computers and the LAN. Applications and services on the other hand require usernames and password. Email is used by a majority of the staff. In February 2009, about 70 percent of the passwords used for email access were simple and unsecure. The administrator identified that the most common problem with managing user accounts is password resetting due to forgotten passwords. Recently the use of weak user passwords resulted in identity theft from 'hackers'. These hackers have impersonated users through their email accounts for malicious purposes. The administrator made mention of the initiative to introduce password complexity polices to force strong password usage.

4.5 Authentication System Requirements

In order to design a suitable authentication architecture for KNUST, an analysis of its requirements was made. The requirements are based on current IT setup, authentication problems faced and scalability.

- The architecture to be designed must be able to improve authentication security in KNUST. It must either strengthen or augment the existing authentication mechanism. The previous section describes the use of weak passwords by many users of the email system in particular. An introduction of password complexity policies will not entirely deter 'hackers' from obtaining passwords. It will also force users to adopt inadequate methods to remember passwords. For example users will write down passwords and may even keep them in close proximity to their computers. This will increase risk from physical security intrusions.
- There should be very minimal modification to existing infrastructure especially the application servers. Access control rights within application servers needs to be made available to authentication and authorization processes.
- The architecture must be scalable enough to allow the addition of more applications and services. The review of the KNUST ICT policy (KNUST 2006) shows the projected target for application acquisition.
- There should be proper synchronization between architectural components that perform similar functions. Where necessary distributed authentication should be considered to fit KNUST sub-network administration and LAN-wide administration.

- There should be adequate failure recovery scenarios in place to address component failure.
- The architecture should properly log all user authentication attempts, successes and failures. Cohesion between components should exist to prevent simultaneous logins and security breaches.
- The architecture should adequately perform auditing that is coordinated across the entire LAN.

4.6 Summary

This chapter presented relevant information required for the development of an authentication architecture for the case study institution. The next chapter proceeds to present the proposed authentication architecture.

Chapter Five

Proposed Authentication architecture:

Design and Concept of Operations

5.0 Introduction

This chapter presents the proposed authentication architecture for the case study institution. It continues to explain the concept of operations and describes some advantages and disadvantages of the proposed architecture.

5.1 KNUST Proposed Authentication architecture

Based on review of literature (Clercq 2002), (Bui 2005), (Ufinity 2003), (Futronic 2006) and knowledge of the ICT systems from the ICT staff in KNUST, an architecture was developed to perform central authentication in KNUST. The architecture overview is shown in Figure 5.1-1. The architecture adopts a hybrid-type design (Ufinity 2003) and operates with database replication, proxy and internet-provider redirection SSO mechanisms. The architecture is distributed in nature and it uses the fingerprint biometric as the primary authentication mechanism.

The main components of the architecture on a single network segment include:

- Distributed authentication server (DAS)
- Distributed authorization server (DAUS)
- Application servers (APS)
- User computers with USB fingerprint readers (FR)

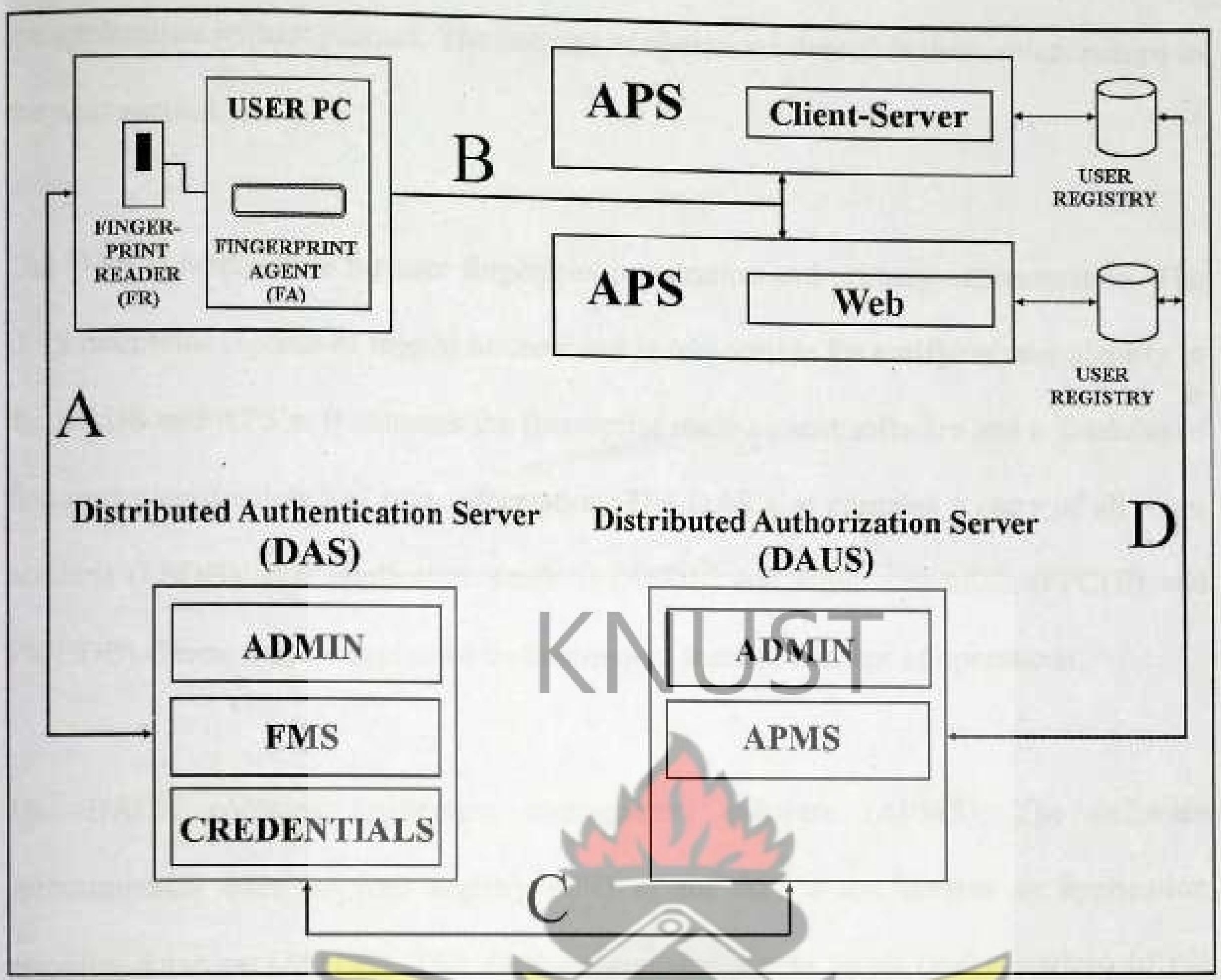

Figure 5.1-1 KNUST Proposed Architecture – General Concept.

Figure 5.1-1 shows the architecture with its main components. It also shows the relationships that form the concept of operations. The relationships are indicated as A, B, C and D in the diagram. Relationship A shows the user process authentication as well as the user registration process. Relationship B is the application request process. This is a transparent authentication process that occurs between a user computer and an application server. This process is single sign-on because the user does not need to re-enter authentication information. Relationship C is the verification process and relationship D is the authorization process. Verification and authorization occur within the application request process. The concept of operations describes these relationships in the next section.

The DAS is responsible for user fingerprint registration and primary authentication. The DAS maintains records of logged in users and is responsible for verifying user identity to the DAUS and APS's. It contains the fingerprint management software and a database of fingerprint credentials and user information. The DAS also contains a copy of all login sessions (LSDB), user application sessions (ASDB) and login credentials (FPCDB and PWCDB). These will be explained further in next section, concept of operations.

The DAUS contains application management software (APMS). The software communicates with the user registry (UR) of the APS's and creates an application mapping database (AMDB). This database contains access levels (authorization) of the users within the user registry of the APS. The various APS's contain the various client-server and web applications that exist in KNUST. Due to the heterogeneous nature of the different APS's a DAUS is allocated to each network segment with APS's.

The architecture, if incorporated into the KNUST IT setup will have several authorization servers and authentication servers. The KNUST Network Operations Centre (NOC) and the miniature network operations centres of the sub-networks will typically have one DAS for user registration and authentication. If the sub-network contains APS(s), then it will have a DAUS as well. ~~KNUST~~'s NOC has a DAUS to manage the main web applications used throughout the institution. The DAS in each sub-network will also hold a database of all fingerprint readers (FR), as well as a database of the MAC addresses of all user computers. These databases enrich the content of the audit logs that will be generated by the system. These audit logs ensure user accountability within the system. The MAC address database and FR database are not represented in the architecture diagrams, but are relevant to the security of the architecture.

To allow for fingerprint authentication, portable USB fingerprint readers (FR) are attached to all user computers. Fingerprint management software (FMS) is installed on all DAS's and a fingerprint agent (FA) on the user computers. The adoption of the FR, FMS and FA into the architecture can be made possible through third party providers of fingerprint recognition technology. An example of such a provider is Futronic. (Futronic 2006) Futronic offers a series of hardware and software products based on their proprietary fingerprint recognition algorithm. A diagram showing a KNUST LAN overview of the distributed architecture is shown in Figure 5.1-2. The diagram also shows the servers and server entities within the KNUST NOC and KNUST network-segments.

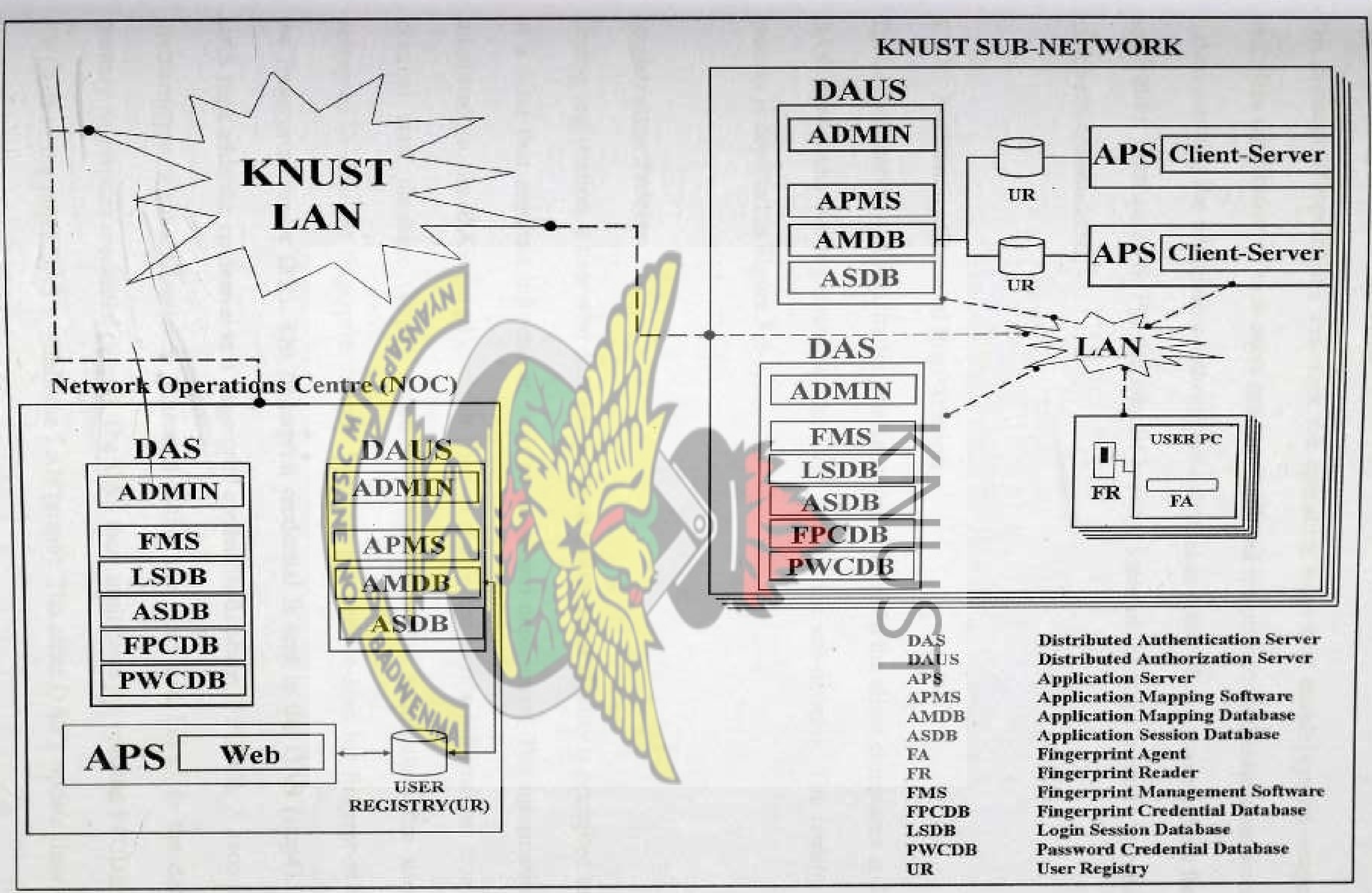
KNUST SUB-NETWORK
DAUS
ADMIN
APMS
AMDB
ASDB
UR
UR
APS Client-Server
APS Client-Server
LAN
DAS
ADMIN
FMS
LSDB
ASDB
FPCDB
PWCDB
USER PC
FR
FA
KNUST LAN
Network Operations Centre (NOC)
DAS
ADMIN
FMS
LSDB
ASDB
FPCDB
PWCDB
DAUS
ADMIN
APMS
AMDB
ASDB
APS Web
USER REGISTRY(UR)
DAS Distributed Authentication Server
DAUS Distributed Authorization Server
APS Application Server
APMS Application Mapping Software
AMDB Application Mapping Database
ASDB Application Session Database
FA Fingerprint Agent
FR Fingerprint Reader
FMS Fingerprint Management Software
FPCDB Fingerprint Credential Database
LSDB Login Session Database
PWCDB Password Credential Database
UR User Registry

Figure 5.1-2 KNUST Proposed Architecture – LAN Overview

5.2 Concept of Operations

The concept of operations describes the operating steps that enable system usage. It describes registration of new users into the DAS and the primary authentication process. It then explains the transparent authentication that takes place when a user requests for an application from an APS. For the sake of brevity, some of the steps in these operations have been consolidated.

5.2.1 Authentication and Registration

The authentication and registration processes occur between the client computers and the DAS. New users are registered via the DAS within their sub-network. The registration process is depicted in Figure 5.2-1.

Registration Process

During registration, a user who does not yet have a fingerprint identity is prompted to fill in a form that requests information such as name and department. The information is submitted to the DAS (step1), which tries to verify the user's information from an external staff database (step2). Upon confirmation of the information the user is prompted to scan his fingerprint (step3). The user proceeds to scan his fingerprint with the fingerprint reader (FR). The fingerprint credential is sent to the DAS (step4). The DAS then adds the credential to a fingerprint credential database (FPCDB). A secondary username/password credential is generated within the DAS and linked to the default primary fingerprint credential (step5). The DAS then multicasts the updated FPCDB and PWCDB to the other DAS's within the LAN (step6). The other DAS's update their own copies of the FPCDB and PWCDB (step7). Then a confirmation of the update is sent to the initial DAS by the other DAS's (step8). The DAS then sends a confirmation to the user that registration has been successful (step9). During registration, at least one other DAS must send confirmation of the FPCDB and PWCDB before the registration can be successful.

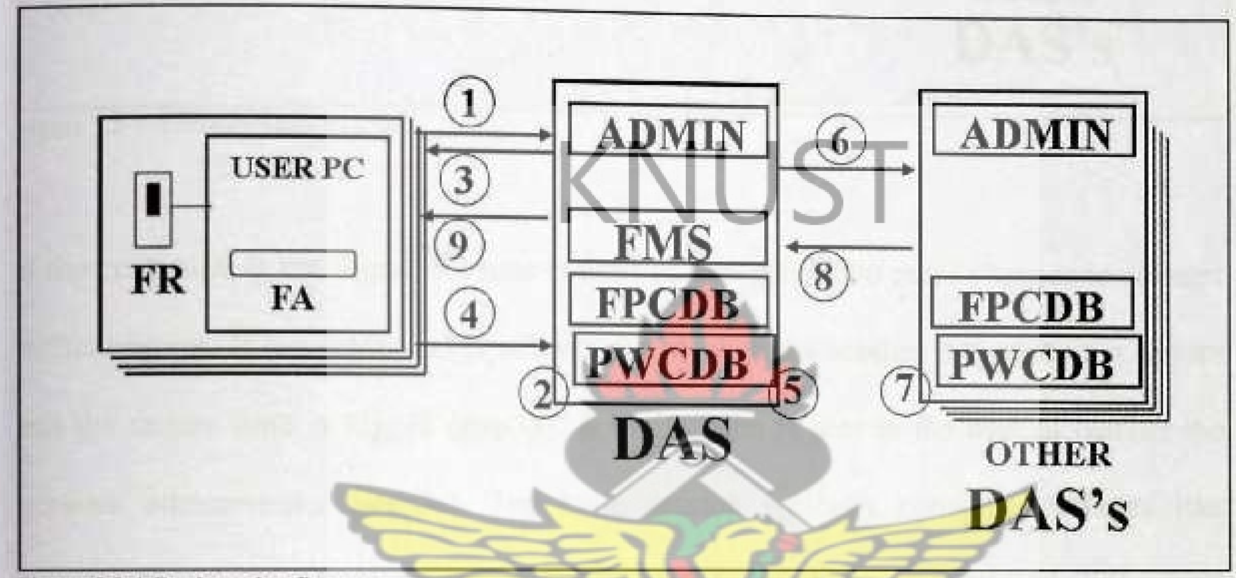

Figure 5.2-1 Registration Process

Authentication Process

Users are required to authenticate to a DAS prior to accessing the network and any applications. The user authentication process is shown in Figure 5.2-2. The user inputs his fingerprint credential via a FR connected to a user computer. The credential is sent to the DAS (step1). The DAS creates a login session in the login session database (LSDB) (step2). The DAS checks the credential against the local FPCDB. If the credential is found, the login session is tagged as successful. The login time is added to the session (step3a).

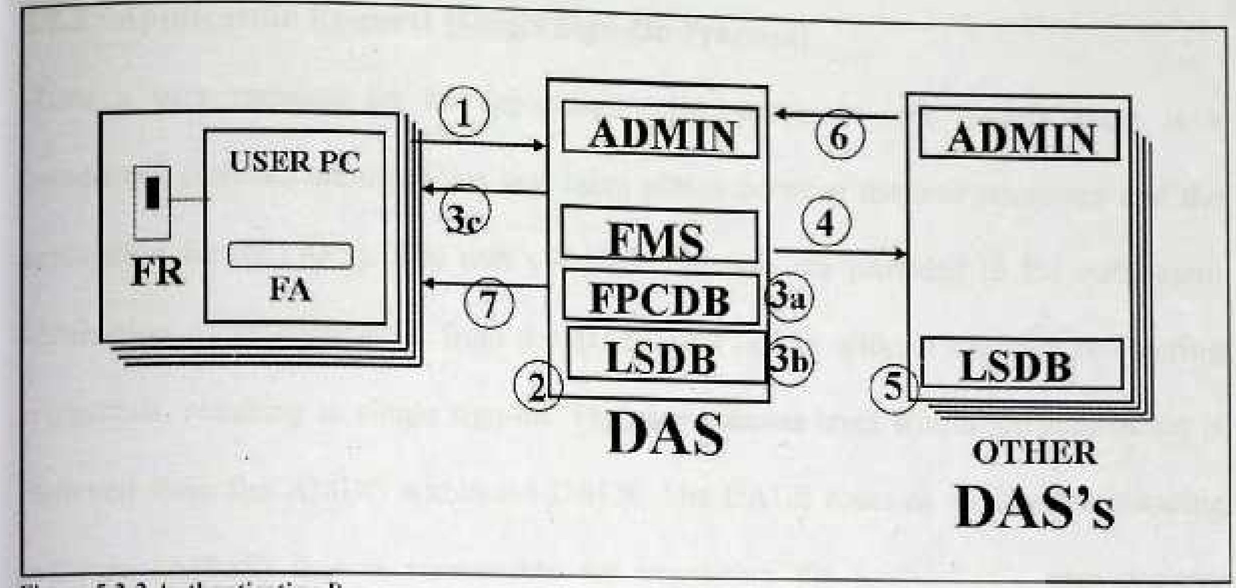

Figure 5.2-2 Authentication Process

If the credential is not found, the user is notified and given two more chances to attempt authentication. If the credential is still not found, then the session is tagged as a failure and the failure time is logged (step3b). A notification is sent to the user to contact the network administrator (step3c). The login session typically contains attributes like session id, login/logoff success time, login/logoff failure time, credential id, fingerprint reader id and MAC address of user computer. These attributes will be used to maintain a reliable audit trail across the architecture. The DAS sends the LSDB update to the other DAS's within the LAN by multicast (step4). The DAS's update their local copies of the database (step5) and send confirmation to the initial DAS (step6). The user receives confirmation of authentication and can now access applications (step7).

5.2.2 Application Request (Single Sign-On Process)

When a user requests for an application (client-server or web based), there is a transparent authentication process that takes places between the user computer and the application server (APS). The user's login credentials are provided to the APS upon verification of authentication from the DAS. This occurs without the user re-entering credentials, resulting in single sign-on. The user's access level within the application is retrieved from the AMDB within the DAUS. The DAUS contains application mapping software (APMS) that is responsible for translating the application's user registry information into a viable format to be used by the AMDB. This database contains a mapping of user credentials in the application (with access levels) to the user's fingerprint credentials in the DAS. The application request process is shown in Figure 5.2-3.

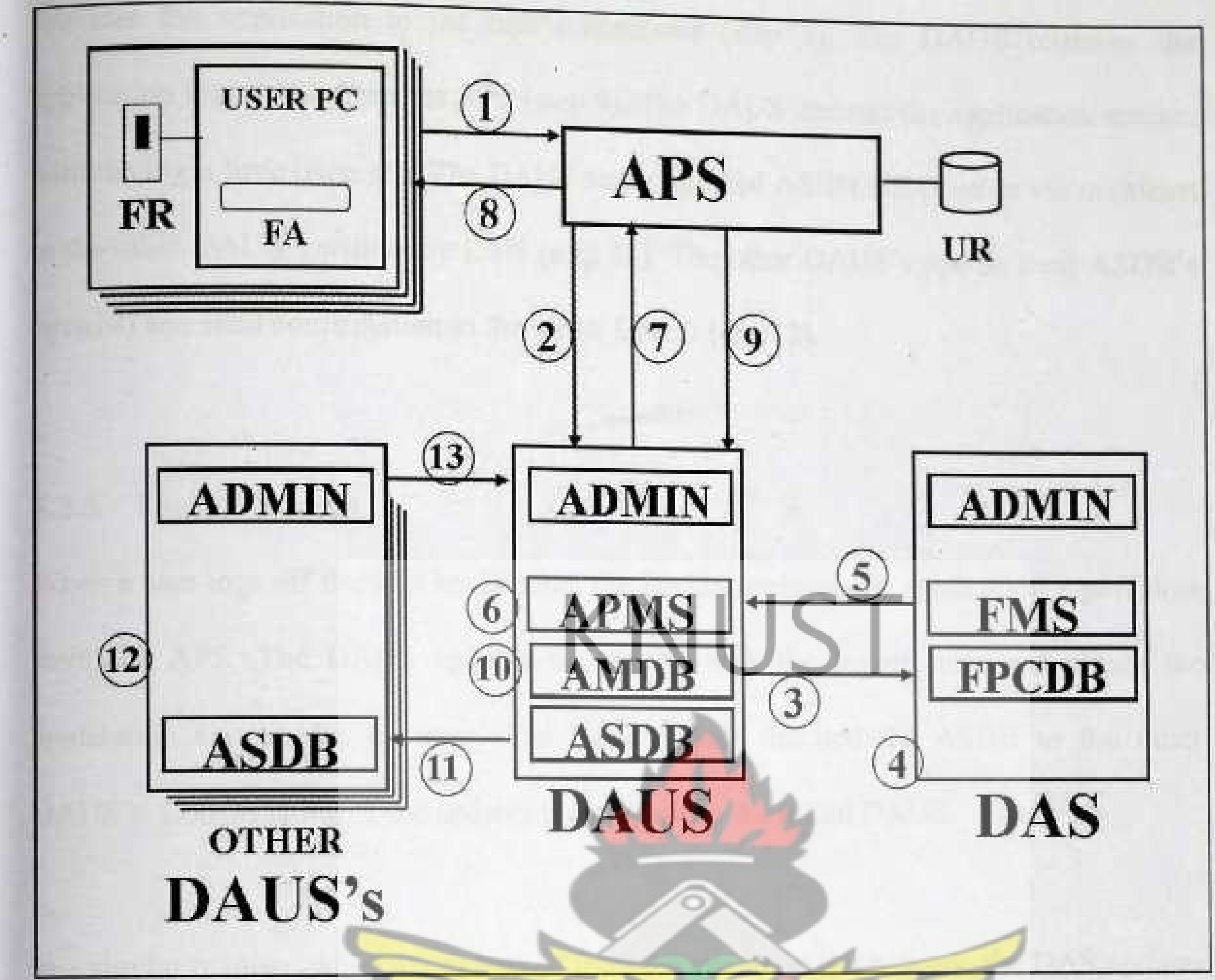

Figure 5.2-3 Application Request Process

The user requests for a client-server or web application from an application server (step 1). The APS requests user login credentials from the DAUS (step 2). DAUS requests user verification from the DAS (step 3). The DAS verifics the user as valid and logged in from the LSDB (step 4). DAS sends the logged in user's credential to the DAUS (step 5). The DAUS creates a user application session from the user credential and the AMDB (step 6). The session contains attributes similar to the login session within the LSDB. The session also contains the appropriate user access level provided by the AMDB. The DAUS sends the user session information to the APS (step 7). The APS authenticates the user and provides the application to the user's computer (step 8). The DAUS retrieves the application login time from the APS (step 9). The DAUS updates the application session with the login time (step 10). The DAUS sends updated ASDB information via multicast to the other DAUS's within the LAN (step 11). The other DAUS's update their ASDB's (step12) and send confirmation to the initial DAUS (step13).

5.2.3 Logoff Process

When a user logs off from an application, the DAUS retrieves the application logoff time from the APS. The DAUS updates the ASDB with the logoff time and closes the application session for the user. The DAUS sends the updated ASDB to the other DAUS's. Confirmation of the updates is sent back to the initial DAUS.

In a similar manner, when a user logs of from the system in it's entirety, the DAS updates the session with the logoff time. The updated session is updated across the other DAS's and confirmations are sent back to the initial DAS.

5.2.4 Server Database Replication

Database replication is widely used within this architecture as can be seen in the sections above. Each update performed to a database within the DAS or a DAUS is replicated across the LAN. When an administrator makes changes to the user registry of an application, the DAUS updates the changes in the AMDB. Then like the other databases it sends an update to the DAUS's.

The only database that is not replicated like the others is the password credential database (PWCDB). This database is reserved for finger print failure situations. It contains a unique mapping of a generated username and password for each fingerprint credential in the FPCDB. An explanation of the usage of the FPCDB for authentication as a result of a specific failure scenario is found in section 5.2.7.1

The constant updates with confirmations that occur are security checks to prevent simultaneous logins for a single user. This would be more evident when there is a problem with the fingerprint authentication mechanism for an end user and is described in section 5.2.7.1 The updated servers also ensures that in the case of a DAS or DAUS failure, another DAS or DAUS can perform the required function. This increases system availability and provides redundancy for the server components.

5.2.5 Server Resource Index (SRI)

The server index is performance metric for the DAS and DAUS that gives an indication of the server's resource insensitivity. This metric indicates the load on the DAS based on average login sessions generated daily. The DAUS metric would be based on average application sessions generated. The index will thus show which DAS and DAUS has the lowest utilization. An optimum performance level for this index will be used to identify overused servers within sub-networks. The sub-networks can then be divided into smaller segments with additional servers to optimize performance within the system.

5.2.6 Auditing

Auditing is made possible because of the various databases containing login sessions and application sessions within the system. The attributes of these sessions which are constantly updated and replicated allow for comprehensive audit reports. Audit reports can be created according to user, APS, DAS or DAUS activity. For example user activity reports are generated from a user's fingerprint credential. The corresponding login sessions and application sessions provide his details over a period of time. Details can be successful logins, failed logins, login durations, applications accessed and access times.

5.2.7 Failure Recovery

The proposed authentication architecture has capabilities to handle a number of failure scenarios. These include fingerprint reader failure, logout failure and DAS or DAUS failure. Combinations of failure scenarios are also possible but such analysis is left for future work.

5.2.7.1 Fingerprint Reader (FR) Failure

In the event of a FR failure when a user tries to authenticate to the DAS, the following operations will take place:

- The user receives a notification to contact the system administrator.
- The administrator initiates a fingerprint override sequence. This override sequence deactivates the fingerprint credential and activates the password credential as the primary credential for the user.

- The DAS retrieves the associated username for the user in question. This username is directly related to the user's fingerprint credential in the FPCDB. The user's profile is queried and the username is retrieved by the administrator. The username is retrieved together with a temporary password that must be changed at first logon by the user. The temporary password is also configured with a time limit, after which it is disabled. This is a security measure put in place to check the availability and lifespan of such temporary passwords in the system.
- The system administrator then relays the information about the username and password to the user via the most secure channel available within the case study institution.
- The user authenticates to the DAS with the username and password and then single sign on resumes.

5.2.7.2 Log Off Failure.

The following situations relate to Log off Failure.

- A user's computer looses power and shuts down
- A network failure disrupts the connection to application server and network.
- A user forgets to logout from an application and shuts down the computer
- A user forgets to log off from the system in its entirety.

In any of these events, login sessions and application sessions fail to receive log off times and remain open. The following occurs when the user attempts to log in the next time.

A. Fingerprint Authentication.

- The user logs in again with his fingerprint credential.
- The DAS checks the LSDB and finds a current session with the credential that is still open.
- The DAS tags the session as log off failed and updates the failure time.
- The DAS closes the session, and reopens a new session for the user.

B. Password Authentication

- In the case of password authentication, when an open session is found, the user is prompted to contact the system administrator. This is an attempt to determine the validity of the new user attempting to log in to the system.
- The system administrator performs the user verification action before closing the session which is tagged as log off failed. User verification can be verbal or visual. The method used is dependent on the user in question and the system administrator.
- The user is informed to re-enter password information.

5.2.7.3 DAS or DAUS Failure

- In the event that a DAS located in a sub-network cannot authenticate users or initiate login sessions, the users are redirected to an alternate DAS within another sub-network. The alternate DAS will have an updated FPCDB so there will be no inconsistencies. The alternate DAS will be chosen by the value of the DAS's SRI. The DAS with the lowest SRI will become the alternate DAS.

- Likewise if a DAUS in a sub-network cannot initiate application sessions or adequately update the AMDB, the role of that DAUS is transferred to an alternate DAUS with the lowest SRI.

5.3 Assumptions of the Architecture

The following assumptions are made on aspects of the environment that the authentication system is not able to control. For example physical security and the trustworthiness of the remote components of the architecture (sub-network components).

- It is an assumption that the IT environment provides the servers with appropriate physical security that reflects the value of the IT assets protected by the authentication system.
- All LAN components are appraised to have the appropriate level of trust in order to properly enforce network security.
- It is assumed that KNUST users are aware of security implications and accountability for improper misconduct.
- It is an assumption that the KNUST ICT policy is regularly reviewed and updated to adequately accommodate new security threats and user needs.

5.4 Advantages of the Architecture

The following are advantages of the authentication architecture:

- It allows centralized user management.
- It has failsafe mechanisms against fingerprint reader failure, log off failure and DAS and DAUS failure.
- It has components that work together to provide effective audit management.
- Availability is increased through the replication of authentication function to sub-networks.
- More users and applications can easily be accommodated by providing more servers.

5.5 Disadvantages of the Architecture

The following are some of the limitations of the architecture:

- The reliability of the system depends heavily of the update processes between databases. Expensive servers are needed to ensure quick and consistent updates.
- Implementation of the fingerprint credential can be costly, since every computer requires a portable USB fingerprint reader.
- In order to accommodate increase in users and applications, more servers will be required. This results in increased cost and more strain on the updating process.
- The logs generated by the system will demand a large amount of server hard drive space.

5.6 Summary

This chapter described the proposed authentication architecture based on information obtained from interviewing KNUST IT personnel and review of literature. The concept of operations, failure recovery pros and cons were also described. Assumptions were made that reflect aspects beyond the control of the system.

Chapter Six

Conclusions and Recommendations

6.0 Introduction

This chapter presents future work and the conclusions of the research study. Various aspects including the concept of operations and failure recovery will be refined in future iterations of this research. Some recommendations for future work are discussed below.

6.1 Conclusions of Study

Introductory literature into user authentication through single sign-on has revealed a number of benefits for adopting such an authentication architecture. Nowadays, large institutions and organizations such as the case study institution use multiple applications and services provided by IT to perform their daily business processes. Typically each application and service requires its own authentication. There is a significant level of difficulty associated with using and managing such IT systems. These institutions have been found to be candidates that stand to benefit from implementing a SSO authentication architecture. As a result various SSO architectures and SSO operating mechanisms were studied and compared based on a number of factors. The result of the study presented architectures and mechanisms that were modified to suite the case study institution and similar organizations.

The resulting architecture created a way to maintain existing application servers in the KNUST by retrieving access control levels into an application session database.

The design also opted to use fingerprint biometric authentication as the primary source of user authentication, with the username/password mechanism as a failsafe. The concept of operations for the architecture explains a rudimentary session creation system within databases that are constantly updated throughout the network. The sessions within the database system allows the introduction of a comprehensive audit management scheme. The results of this research will serve as a framework for future design and specification of a complete authentication architecture with single sign on capabilities.

6.2 Recommendations

The design presented in the research is a preliminary one. Not all aspects have been clearly defined. Also some mechanisms need to be determined before prospective design work can be done. The following recommendations for the authentication system define some of the aspects and mechanisms required before design work can be initiated:

- Specifications for the update process between databases need to be determined. A secure encryption scheme needs to be adopted to ensure that database updates are not transmitted across the LAN in plaintext format.
- The application mapping software needs to be specified or developed. The software must be able communicate with the different user registry formats that the various application servers have. So far interbase, SQL, and LDAP type formats have been identified. The software must be able to translate the access level information within these user registries into a suitable format to be used by

the application mapping database. Detailed requirements for such software need to be determined.

- A possible method for accommodating large log files from login sessions and application sessions needs to be proposed. Database software such as Interbase prompts database administrators to activate log creation for only the most relevant database fields due to resource intensiveness.
- Requirements for each of DAUS need to be specified in order to determine whether the functions of the DAS and DAUS can be merged as one server. This will reduce the number of servers required for implementation.
- A future upgrade to this design is to allow the application mapping software to communicate directly between the application servers (APS) and the primary credentials activated for users. This will speed up the transparent authentication system.

References

1. Bui, Sonia. "Single Sign-On Solution for MYSEA services." Msc Thesis, 2005.

2. Clercq, Jan De. "Single Sign-On Architectures." 2002.

3. *Enterprise Single Sign-On:Easing the authentication process.* November 11, 2008. Available:http://searchsecurity.techtarget.com/tip/0,289483,sid14_gci1338507,00.html (accessed May 3, 2009).

4. *Exploring Authentication Methods: How to develop secure systems.* November 10, 2008. Available:http://searchsecurity.techtarget.com/generic/0,295582,sid14_gci1123443,00.html (accessed May 3, 2009).

5. Futronic. *Fingerprint authentication, Fingerprint authentication server.* 2006. Available:http://www.biometric-fingerprint.com/fingerprint-authentication-server.html (accessed June 26, 2009).

6. Imprivata. *Expert Advice - 20 Practical Tips on Authentication and Access Management from Practised Professionals.* Imprivata Inc., 2009.

7. KNUST. "ICT for Knowledge Development (ICT4KD)." *Information and Communications Policy.* KNUST, November 2006.

8. Michael Liou. *Enterprise Single Sign-On Best Practise Considerations.* White Paper, CA, 2007.

9. Nitai Alush-Aben. *Selecting The Right Solution - SSO vs Password Management.* 3AExperts LLC DBA, 2005.

10. Parker, T A. "Single Sign-On Systems - The Technologies and the Products." *European Convention on Security and Detection - Publication No 408.* IEE, 1995.

11. Secude. "Simple, Secure Enterprise Single Sign-On - Advantages for your Company." White Paper, Switzerland, 2002.

12. Ufinity. *SSO Architecture Comparisons.* White Paper, Ufinity, 2003.

13. "Understanding Enterprise SSO." *Microsoft Cooperation Web Site.* 2009. Available:http://msdn.microsoft.com/en-us/library/aa745042.aspx.htm (accessed April 15, 2009).

14. *What is Authentication? - A definition from whatis.com.* June 4, 2007. Available:http://searchsecurity.techtarget.com/sDefinition/0,,sid14_gci211621,00. (accessed May 4, 2009).

15. *What is authorization? - A definition from whatis.com.* January 13, 2006. Available:http://searchsecurity.techtarget.com/sDefinition/0,,sid14_gci211622,00. (accessed May 4, 2009).

Appendix

ICT PERSONNEL STRUCTURED INTERVIEW

Name:
IT Division:
Location:

1. Do you understand the concept of centralized authentication?

2. Have you ever heard of single-sign on?

3. Do you know if there is any such implementation within the institution's ICT architecture?

4. How many forms of user authentication are you aware of in this institution?

5. What information services within this instituion require authentication?

6. Do you work with any of these information services provided by UITS? (If yes, which ones?)

   (If no skip to question 10) Classification of services.

7. Do any of your responsibilities require you to manage usernames/passwords or other means of user authentication? (If yes, in what capacity)

8. How often do you encounter problems with user authentication?

   (Multiple times in a day/once or twice a day/once or twice a week/once or twice a month/rarely)

9. Specifically, how often do you have to retrieve or reset user passwords?

   (Multiple times in a day/once or twice a day/once or twice a week/once or twice a month/rarely)

10. Do you know if any of these information services are capable of providing user auditing? (explain if necessary)

11. Currently how is user auditing carried out?

12. Do you have any additional comments related to any of the above issues?