Secure Web System in a Cloud Environment

BIBESH POKHREL

Master of Science Thesis
Stockholm, Sweden 2012
TRITA-ICT-EX-2013:3
Acknowledgement

First and foremost I want to thank Prof. Dr. Sead Muftic for his supervision during this Thesis work. His guidance has helped in every step of this work toward its conclusions.

I would also like to thank Dr. Feng Zhang and Dr. Abdul Ghafoor for their support and assistance to make this Thesis successful. Also heartfelt thanks to all of SecLab colleagues, specially, Davit Hakobyan, Sanjaya Dahal, Daniel Gomez Villanueva and Arunendra Nath Pandey for sharing their knowledge and insight during this Thesis.

Finally, I would like to pay my sincere thanks to my parents and loved ones for all their love, support and trust and providing an opportunity for my graduation.
Abstract

Advent of cloud computing has brought a lot of benefits for users based on its essential characteristics. Users are attracted by its costs per use service and rapidly deploy their applications in the cloud and scale by using virtualization technology without investing in their own IT infrastructure. These applications can be accessed through web based technology, such as web browsers or mobile apps. However, security becomes a major challenge when user’s data and applications are stored in a remote server in a virtualized environment and Internet is medium for accessing them. Internet is always prone to known and unknown threats and a successful breach in the security in cloud environment could lead to a massive loss to property, data and thereafter future of cloud computing technology.

In order to meet the challenges of security needs in cloud computing, security architecture is presented in this Thesis. This Cloud Security Architecture delivers security solutions to deployed applications in the cloud as a service. Security solutions that are delivered by the architecture are Authentication, Authorization, Identity Management and Access Control. With these security solutions by Cloud Security Architecture, the Thesis proposes Secure Web System that incorporates secure authentication and privacy enhancing applications in cloud environment. Authentication utilizes the use of the smart card technology and thus is able to provide robustness to the procedure. Further, two more methods of authentication, browser certificate and username/password based give flexible approach when smart card is not available to clients. Applications deployed in a cloud environment would provide security and privacy for users while searching for any query in remote search engine or browsing a remote web server.

Thus, the Thesis lays a foundation towards approaching security and privacy for applications that are deployed in Cloud Security Architecture and building up a Secure Web System.
Contents

Acknowledgement .............................................................................................................. 1
Abstract .............................................................................................................................. 2
List of Figures ....................................................................................................................... 5
Chapter 1: Introduction ....................................................................................................... 6
  1.1 Problem Statement ................................................................................................. 7
  1.2 Purpose and Goals ............................................................................................... 8
  1.3 Research Methodology ....................................................................................... 8
  1.4 Thesis Outline ..................................................................................................... 9
Chapter 2: Background ..................................................................................................... 10
  2.1 Introduction to Cloud Computing ..................................................................... 10
  2.2 Cloud Computing Architecture ...................................................................... 11
  2.3 Security ............................................................................................................. 13
  2.4 Privacy .............................................................................................................. 15
Chapter 3: Cloud Security Architecture ......................................................................... 16
  3.1 Central Cloud Security Architecture ................................................................ 16
  3.2 Components ..................................................................................................... 17
    3.2.1 Central Security Server ............................................................................ 17
    3.2.2 Cloud Client Station ............................................................................... 18
    3.2.3 Portal Security Server ............................................................................ 19
    3.2.4 Cloud Portal ........................................................................................... 19
  3.3 Secure Protocols ............................................................................................... 20
    3.3.1 Strong Authentication Protocol .............................................................. 20
    3.3.2 Single-Sign-On Protocol ........................................................................ 22
    3.2.3 Secure Session Protocol ........................................................................ 22
Chapter 4: Design of the Secure Web System ................................................................. 24
  4.1 Overview of Secure Web System ....................................................................... 24
  4.2 Components of the Secure Web System ............................................................. 25
    4.2.1 Cloud Login Client ................................................................................... 26
    4.2.2 Security Proxy Server .............................................................................. 27
4.2.3 Secure Search and Browse Application .................................................................29
4.3 Security ..................................................................................................................29
4.4 Privacy ..................................................................................................................30
Chapter 5: Prototype Implementation ...........................................................................32
  5.1 Cloud Login Client ..............................................................................................32
  5.2 Secure Search Application ..................................................................................34
  5.3 Secure Browse Application ..................................................................................37
Chapter 6: Conclusion and Future Work .......................................................................39
  6.1 Conclusions ........................................................................................................39
  6.2 Future Work ........................................................................................................40
Acronyms .....................................................................................................................41
References ....................................................................................................................42
List of Figures

Figure 1: Layered Architecture of Cloud Vs Delivery Model.........................................................12
Figure 2: Simple Cloud Architecture .............................................................................................13
Figure 3: Cloud Security Architecture ...........................................................................................16
Figure 4: Components of the Central Security System .................................................................18
Figure 5: Cloud Client Station .......................................................................................................18
Figure 6: Portal Security Server ..................................................................................................19
Figure 7: Cloud Portal ..................................................................................................................20
Figure 8: Strong Authentication Protocol ....................................................................................21
Figure 9: Single-Sign-On Protocol ................................................................................................22
Figure 10: Secure Session Protocol ..............................................................................................23
Figure 11: Architecture of the Secure Web System ......................................................................25
Figure 12: Flow Diagram of Cloud Login Client ...........................................................................27
Figure 13: Secure Communication with Portal Security Server ....................................................28
Figure 14: Secure Search and Browse Applications .....................................................................29
Figure 15: Local Smart Card Authentication Login Screen ..........................................................33
Figure 16: Authentication with Web Browser Certificate Screen .................................................33
Figure 17: Authentication with Username-Password Screen .........................................................34
Figure 18: Page for Submitting Search Query ..............................................................................35
Figure 19: Display of Search Result ..............................................................................................36
Figure 20: Configuring of the Browser to use Proxy ...................................................................37
Figure 21: Browsing with Streaming Proxy ..................................................................................38
Chapter 1: Introduction

Security of information and resources are the most important aspect in any computer communication environment. It become ever so demanding when this communication takes place in an open place such as internet. Internet is the fundamental part of today’s cloud computing paradigm. It is challenging to replace traditional in-house IT infrastructures. These IT infrastructures are very resource and cost intensive. Cloud computing, on the other hand, through its three delivery models - Software as a Service (SaaS), Infrastructure as a Service (IaaS) and Platform as a Service (PaaS), provide many benefits, such as fast deployment, pay-for-use, scalability, ubiquitous network access, rapid elasticity, etc. An estimate of $95 billion of computing market and 12% of worldwide software market will move to cloud around year 2015 [1], but security is a major issue for IT executive to move the application into cloud [2].

The access to cloud-based applications is based on Internet related standards. A web browser is most common way to access the Internet. Cloud based applications are developed using web based technologies. However, vulnerability of web browser against security attacks is a major drawback for implementing cloud applications. SSL is the most popular way of communicating securely using web browsers in the Internet. However, security researchers Thai Duong and Juliano Rizzo in September 2011 have demonstrated a Browser Exploit Against SSL/TLS (BEAST) attack at Ekoparty Security Conference\(^1\) [4, 5] to obtain authentication tokens such as HTTP cookies [6].

An organization wants similar security solutions in their traditional infrastructure to be implemented while accessing any application in cloud with a web browser. Therefore, these cloud-based applications must be developed which support advanced security standards, solutions and features. Authentication of users, authorization for resources, protection of data and communication and privacy of user’s identity are the most fundamental security requirements in any application. These requirements are only met when we support advanced encryption standards, digital signature standards, and security requirement for cryptographic modules. Encryption of sensitive and private data with strong cryptographic algorithms will preserve privacy and security. Additionally, use of public key certificates (PKI) for hashing and

\(^1\) http://www.ekoparty.org//eng/index.php
digital signature, along with strong encryption and decryption standard ensures confidentiality, non-repudiation and some form of authentication in a cloud environment. Authentication is based on the use of certificates which assert identity of a user based on private and public key pair. The Private Key is associated with a user’s public key in such a way that only the bearer of the private key can decrypt the message encrypted with corresponding public key. The information encrypted with private key of a user can be decrypted with his public key. In addition, applying a digital signature scheme of that private key encryption will assert the identity of the originator of the message. It becomes important than to protect this public-private key pair. More importantly the private key must be kept secure enough so that no intruder would be able to access it and use it to impersonate a user.

Smart card technology is emerging in this regard as their storage and processing capacities are steadily increasing [3]. These smart cards can not only be used to store sensitive or private data, such as user certificate, but also to perform advanced cryptographic functions. This enables to build applications that provide the most advanced security features based on smart card technologies. Smart cards are already gaining popularity by financial institutions in the form of debit or credit card to withdraw money from the machines. Cloud build applications need to take advantage of this technology in order to deliver secure applications to their users. This thesis aims to provide these security solutions as a service in a cloud environment for a secure web system. Secure web system constitutes secure authentication and secure communication of messages for searching and browsing in a cloud environment. It uses benefits of central security architecture for cloud system, which will also be presented.

1.1 Problem Statement

We discussed issues regarding security in cloud applications. The risk of security breach is considerable in a cloud portal where a user connects to cloud with some form of authentication mechanism. Generally, this authentication mechanism is based on username and password, which is very prone to attacks, such as brute force attacks and dictionary attacks. Communication channel between a portal and user is only protected with SSL. SSL, as discussed earlier, is also vulnerable to attacks. This can directly result in compromise of user’s data,
privacy and even the system itself. Security solutions to mitigate such attacks are not comprehensively implemented and described.

Further, tracking of users via client’s identifiable information, such as cookies or IP addresses during browsing of internet, is the most prevalent in modern web technologies. Even some major search engines track user behavior, such as click pattern and queries [25] to deliver their services or sell that information to third parties, what violates user privacy.

1.2 Purpose and Goals
The purpose of this research is to identify and design all components necessary for a secure web system in a cloud environment. These components will support security protocols, certification protocols, SAML protocols, strong authentication protocols, etc. Thus, the system designed in this research enables secure communication with cloud portal and protection of user’s data and privacy of communications.

Taking this into consideration, the goal of this research is to design a secure web system. The secure web system consists of two main parts:

- **Secure Client**: The function of secure client is to establish secure authentication with Authentication Server in a cloud security infrastructure. The authentication established can have three assurance level:
  - High Assurance: FIPS 196 strong authentication based on FIPS 201 (PIV) smart cards
  - Medium Assurance: IE-based certificate for authentication
  - Low Assurance: username/password for authentication

- **Secure Browsing and Searching**: A proxy based cloud application for secure browsing and searching with security and privacy enhancing features provided by cloud security architecture and components.

1.3 Research Methodology
Research is a scientific and systematic search of information that is relevant to some specific topic. The term scientific means that the information gathered is interrelated from the facts. It
is systematic observation, classification and interpretation of data [7]. Research methodology concerns with conducting the research scientifically. It is a science of principles and procedure in research.

This research is based on design science research methodology. Design science research is fundamentally problem solving paradigm in engineering [8]. It must produce an artifact that could be a construct (vocabulary and symbols), models (abstractions and representations), methods (algorithms and practices), and instantiations (implemented and prototype systems) [9]. The main step in this research method are problem identification and motivation, define the objectives for a solution, design and development, demonstration, evaluation and communication [10]. This approach closely correlates to this current research: the first step correlates to problem statement, second to the purpose and goals of this research, design and development are discussed in chapters 2, 3 and 4, demonstration is in chapter 5, evaluation in chapter 6 and consequently the complete thesis report corresponds to communication. Thus, it is the main motivation for using design science methodology for this research.

1.4 Thesis Outline

Chapter 1 discusses introduction to cloud computing and current provisions of authentication in cloud access mechanisms. This lays a background for the research problem statement for designing Secure Web System for a cloud environment. In Chapter 2, the concept of cloud is elaborated with regards to the Thesis area. This Chapter also presents the security and privacy issues arising in Secure Web System and authentication mechanism, while accessing the cloud services. Chapter 3 describes the Cloud Security Architecture, its components and secure protocols in that architecture. Chapter 4 presents the design of our Secure Web System by taking Cloud Security Architecture as a core for delivering security. It also presents the components of Secure Web System, such as Cloud Login Client for authentication of clients, and Secure Search and Secure Browse applications designed for security and privacy for its users. Security and privacy issues provided by these applications, along with Secure Web System, are also presented in this chapter. The prototypes of these components are presented in the following Chapter 5. Finally, the conclusions and potential future work of this Thesis are presented in Chapter 6.
Chapter 2: Background

This chapter provides the introduction to the technologies and the issues that are related to the current research.

2.1 Introduction to Cloud Computing

Cloud Computing is a recent paradigm which delivers hosted services over the network, typically the Internet. Cloud delivers computing resources, such as hardware and software. Unlike traditional IT infrastructure, customers or users need only a computer and Internet connection to access services offered by the cloud. NIST defines cloud computing as “Cloud computing is a model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction” [11]. The essential characteristics of cloud computing offer several advantages, such as decrease in cost by pay-per-use, rapid scalability, unilateral provisioning of computing capabilities without providers interaction, resources pooling by dynamically assigning and reassigning different physical and virtual resources according to demand and ease of access to information from any geographical location or client platforms, such as web browsers or mobile apps. Generally, there are four types of clouds: public, private, community and hybrid, also known as deployment models.

Public Cloud: In a public cloud application, storage, infrastructure and all other resources are openly available to use for public. General public can access these resources through the internet and the cloud provider can be government, business or any academic organization.

Private Cloud: Private cloud is meant to be used only by a single organization. It may be located within the organizational premise or outside and can be managed by organization or by a third party.

Community Cloud: The cloud which is shared by group of organizations having similar or shared concerns, such as mission, compliance, policy, jurisdiction, security, etc is a community cloud.

Hybrid Cloud: Hybrid cloud are the combination of two or more clouds, private, public or community.
Apart from deployment models, cloud computing has basically three service models by which cloud providers deliver their services to their customers. These models are also referred as SPI model, where SPI stands for software, platform and infrastructure, respectively.

**Infrastructure as a Service (IaaS):** Infrastructure as a service provides the consumer all the fundamental computing resources, such as processing capacity, storage, network, or servers. Consumers can deploy any operating system or application. However, cloud provider still manages the underlying cloud infrastructure. The consumer has control over his operating system, storage and deployed applications.

**Platform as a Service (PaaS):** In a platform as a service model, the consumers deploy their applications in the cloud provider’s infrastructure. Application must be created using programming languages and tools supported by cloud providers. Consumers have control only over their applications or application hosting environment and the underlying infrastructure, such as routers, storage, operating system or servers are managed by the provider [12].

**Software as a Service (SaaS):** Software as a service model delivers applications to users that are provided in a cloud infrastructure. Applications are managed and run by the providers and users only access those applications through thin client applications, such as a web browser.

**2.2 Cloud Computing Architecture**

Cloud computing architecture can be defined as “the structure of the system, which comprise on-premise and cloud resources, services, middleware, and software components, geo-location, the externally visible properties of those, and the relationships between them” [13]. It facilitates to discuss, describe and develop system-specific architecture using a common framework of reference [14]. Figure 1 shows simple three layered reference architecture in comparison with the service delivery model of the cloud.
The top layer exposes user interfaces or APIs to clients for access to the three cloud services. Clients can access through a simple web browsers or some cloud application specific to the provider. However, the dependency of the three models is optional meaning that it is not necessary that SaaS applications are built on the top of PaaS component, and PaaS component on top of IaaS component [14].

Figure 2 shows a more abstract representation of cloud architecture with respect to the security requirements. Cloud portal hosts different applications which are represented as App-1 and App-2 in the Figure 2. In order to access these applications there should be an authorization point which blocks any unauthorized user from accessing the applications. This authorization point is represented as APP. Before any user request for a particular application should be authenticated. The point where any user is prompted for authentication is called CAP. This is the place where user first connects to and submits its authentication credentials. Only after successful authentication, a user is able to continue requesting a cloud service. A user can use variety of methods for connection with this point, such as Wi-Fi, Internet or 3G/4G mobile network.
2.3 Security

Security is one of the main factors determining the success of a new technology [15]. Cloud computing faces a challenge in this regard as it is projected to grow to a $207 billion industry by 2016 [16]. In the traditional organizational IT environment, the infrastructure are situated behind the firewall and protected by the security tools for their environment. However, the basic security requirements, such as authentication and authorization, confidentiality and integrity and communication security still apply to cloud computing environment. Apart from these, with SaaS delivery model of cloud computing another factor that contributes to security concerns are reliability and security of web browsers, which are widely used to access the services [12].

Authentication is the primary factor in any secure system. The identity of a user has to be established, so that only a legitimate user can access the system. Access to services must be provided only when additional authorization information is provided by that user. Generally, username/password credentials are used to prove the authenticity of a user in the Internet through a web browser. The most common methods, such as brute force attack and dictionary
attacks, have made possible to guess the password and gain unauthorized access to the system. When system requires these credentials to be transferred over the network encrypted for authentication is also vulnerable to attacks with the tools like John the Ripper\(^2\), which makes it possible to get the plaintext. Web browsers are not capable of supporting other advanced technology used for the purpose of authentication, such as strong authentication protocol based on PKI with smart card, which provides strong security mechanisms in the authentication process. When a user enters data or information in cloud computing, data are transmitted through the Internet and finally reaches the desired location where the cloud infrastructure is located. As the Internet is an open place it is very probable that the user data can be tampered with during the transmission by some malicious attacker. A strong measure, such encryption of user’s data and information, computing hash or message authentication code, are required to ensure that confidentiality and integrity of user’s data are guaranteed. Furthermore, attacks such as man-in-the-middle attack, can severely compromise the security of the system. Digital signature schemes must be required to counteract, so that the origin of messages and authenticity of sender could be verified. Hence authenticity and non-repudiation in the system could also be provisioned.

Security of a communication channel between a client and a cloud portal is also an important issue. Generally, communication between these two is secured by SSL/TLS. It is a cryptographic protocol for encrypting the communication channel at the communication layer. However, as discussed in the previous chapter, this procedure has been shown to be vulnerable to BEAST attack. Furthermore, it does not provide all the security services, such as key management or smart card based PKI that are needed to provide end-to-end security for a secure communication in a cloud computing environment. Further, web browser itself may be contaminated with malicious programs or add-ons. This malicious software could leak user’s sensitive information or even hijack user’s session on behalf of an attacker. Thus, robust security mechanisms must be in place in order to ensure that end-users are securely and reliably using their cloud resources.

\(^2\) [http://www.openwall.com/john/]
2.4 Privacy

Privacy is considered as a fundamental human right in United Nation Universal Declaration of Human Rights and European Convention on Human Rights [17] and grants exclusive protection to arbitrary interference to one’s personal information or property. In information technology scenario, privacy may refer to the control of one’s information that can be used, collected or processed by any party involved in an information exchange. The information that a user wants to control is personal identity information, such as name and address, sensitive information such as race, health, biometric information or financial information, usage data, such as browsing history, product usage history or search history and unique device identifiers, such as IP address or hardware MAC addresses.

In a cloud computing environment data are processed or stored remotely and platform is shared with many users through virtualization technology [17]. There can be a leak or exposure of user’s information. Service provider could also sell its customers’ data to other third party business. Cookies, web beacons and other trackers are often used to collect information about users. Big companies such as Google Inc., Microsoft and Quantcast Corp maintain huge database of customer profiles by tracking user’s online activities, which later can be used to target advertisement or sell it to other companies [18]. Data that are collected by tracking technology can also be used to deliver dynamically personalized services based on user’s location, preference, calendar and social networks [17]. Furthermore, Google even keeps record of your search queries, IP address, browser type, browser language, date and time of request, cookies and even the hyperlink that are clicked after getting a search result [19]. These logs may reveal sensitive and personal information about the user and severely intrude into one’s privacy. Hence, it is evidently important to protect user’s personal and sensitive information.

Privacy invasive technology can be counteracted with privacy enhancing technologies. These technologies should be implemented during service and application development period as every developer could not be trained at privacy standard or international privacy legislation.
Chapter 3: Cloud Security Architecture

This chapter presents and describes different components and protocols that are associated with cloud security architecture.

3.1 Central Cloud Security Architecture

Central cloud security architecture contains four main functional components and three main secure protocols. Four main functional components are central security system, cloud station, portal security server, and cloud portal. Strong authentication protocol, single sign-on and secure session protocol are three secure protocols. These components and protocols are introduced in our cloud infrastructure to provide security as a service for a web based applications. Various security functions are accessed as web services. The architecture, shown in figure 3, contains several independent components with their own well defined tasks. Thus it is easy to make this system highly scalable and manageable.

Figure 3: Cloud Security Architecture
3.2 Components

As stated earlier, this architecture contains four main components: central security server, cloud client station, portal security server, and cloud portal.

3.2.1 Central Security Server

Figure 4 shows the central security system of the cloud security infrastructure and which comprises of four components: IDMS server, LCA server, SAML/PDP server, and Authentication server. These servers can be located either on different machines or inside a single machine, what depends upon implementation. The Authentication server is located in front of other servers and acts as a gateway. It transparently handles requests sent form either cloud station or portal security server to IDMS, LCA or PDP servers depending upon the nature of the request. The function of LCA server is to issue certificate to users, application or other servers used for authentication in a cloud system. It also checks the validity of certificates used in the authentication process. IDMS server provides identity services in this cloud architecture. It keeps detail information about all cloud users, servers and system components. It also handles the verification of identity in the authentication process, as requested by Authentication Server. PDP server supports the process of authorization. It contains XACML policy list of users. Based on this policy list, the server makes decision to accept or deny any request made by PEP Server on behalf of any user. It is also responsible for issuing SAML tickets in the initial authentication process, which facilitates Single-Sign-On protocol.
3.2.2 Cloud Client Station

Cloud client station is simply a workstation with a web browser. It also contains a login client that handles additional functions required for authentication purpose. Authentication is usually strong authentication protocol using a certificate. The cloud client station is also able to support high assurance authentication with certificates based on smart cards. Secure channel established by secure protocol is used to communicate with portal security server or with the cloud portal to access web applications.
3.2.3 Portal Security Server
Portal security server is located between cloud client station and cloud portal and transparently handles access and authorization function for any application. It acts as a secure gateway to the cloud portal. The main unit in this server is the PEP Server. The function of the PEP Server is to provide access control. Access control decisions are made in conjunction with the PDP Server. When any user demands access to any application it has to present security credentials to the PEP Server, which is a SAML ticket received form the PDP Server during authentication process. The token is then passed to the PDP Server to check its validity. The PDP server replies back accept or deny response, based on ticket validity. The PEP server then grants access if the reply is accept and blocks access upon receiving deny response.

![Portal Security Server](image)

Figure 6: Portal Security Server

3.2.4 Cloud Portal
Cloud portal is a secure web space available to cloud customers after successful authentication. It provides service hosted by the cloud for its customers. Different cloud providers have their own design of cloud portal by which they can deliver their back-end services. Thus, the portal serves as front-end mechanism for accessing the resources. But the most basic characteristic of this portal is that it must be accessible through web browsers. Otherwise, it fails to take advantage of most prolific application, a web browser, which is commonly used to get the resources form the Internet.

Client through a web browser logs into the portal and consumes the resources. Therefore, cloud portal is just a web server providing back-end services which are web-based applications.
3.3 Secure Protocols

In our cloud architecture, three security protocols describe the nature of the communication between different components. The first protocol describes remote authentication based on FIPS 196 strong authentication protocol [20]. This process takes place between cloud client station and the Authentication Server. After this Single-Sign-On protocol is initiated between the client station and the Portal Security Server. Upon successful completion, secure session protocol is initiated which secures communications between client station and cloud portal. All three of these protocols used here were proposed in the CyptoNET [21].

3.3.1 Strong Authentication Protocol

Strong authentication protocol takes place between two architectural components: client station and the Central Security Server. The purpose of this process is to authenticate a user to a remote server. This protocol is the extension of NIST’s Entity Authentication Using Public Key Cryptography, FIPS 196. In this authentication process, user’s certificate is validated by the LCA server and user’s identity by the IDMS server in the Central Security System. Authentication Server acts as a proxy and mediates communications between them.

At first, client initiates strong authentication protocol by sending PIV authentication certificate to the Authentication Server. The Authentication Server then checks the validity of the certificate by sending it to the LCA server. It also verifies the identity of the client by checking its Distinguished Name by the IDMS server. If both, the certificate and identity, are verified,
Authentication Server generates a random number, \( R_s \) and sends it to the client. Client then signs the hash of that random number with its private key corresponding to PIV authentication certificate and sends to the Authentication Server along with its own random number, \( R_{\text{user}} \). The server then can verify this signature using public key in the PIV certificate received in the earlier step. Then, it computes hash of the random number sent by the client. This hash value is signed by the server using its private key. This signed value and the server’s certificate are sent back to the client. The client then verifies signed value by using public key from the PIV certificate sent by the server. It also verifies the certificate and identity of the server by contacting LCA and IDMS server.

At the end of this protocol, as shown in Figure 8, the Authentication Server requests from the PDP server to issue a SAML ticket for that user. The user identity is again verified by the PDP server by asking IDMS server. If the identity is verified, it issues a SAML ticket for that user which contains information such as ticket-id, identity of the user, timestamp and the address of

![Figure 8: Strong Authentication Protocol](image)
PDP server. The ticket is also digitally signed by its own private key. This ticket is then passed down to the Authentication server and then back to user and stored in a smart card.

3.3.2 Single-Sign-On Protocol

Single-Sign-On protocol is used to seamlessly access any resources without requiring further authentication. It takes advantage of the SAML ticket obtained in the previous step of Strong Authentication Protocol. When any application server initiates Single-Sign-On protocol, the client fetches SAML ticket and digitally signs it with its private key. This signed ticket along with clients’ digital certificate is sent to the PEP server. The PEP server also signs this user signed ticket. Then PEP server sends user signed SAML ticket, user certificate and the ticket signed by it to the PDP server. The PDP server verifies the signature on the ticket. It also verifies the SAML ticket against the issued ticket in its database. If the ticket matches with the one that was issued by it, a permit message is sent back to the PEP server. Upon receiving positive reply form PDP server, it grants access to the client and starts secure session protocol.

![Figure 9: Single-Sign-On Protocol](image)

3.2.3 Secure Session Protocol

Secure session protocol is initiated after the completion of Single-Sign-On protocol. The objective of the secure session protocol is to establish a onetime session key that is used to
transfer message between application server and the client. The PEP server requests from the client its key exchange certificate just after the completion of the Single-Sign-On protocol. The client fetches the certificate and sends back to the PEP server. The PEP server verifies the certificate chain and the Distinguished Name in the certificate with the identity stored in its session container. The session container contains an active session object for a client and is established after successful completion of Single-Sign-On protocol. It also contains the session key and session ID. If the identity and certificate are verified, PEP server generates a symmetric session key and session ID. These two are signed by PEP’s digital signature private key and enveloped by client’s key exchange public key. This protects message sent to the client. Client extracts the session key and session ID by decrypting with its private key corresponding to key exchange certificate when the digital signature of the PEP server is verified. Then, the client communicates messages in PKCS#7SignedAndEnvelopedData using symmetric session key to envelop and digital signature private key for signing the data.

**Figure 10: Secure Session Protocol**
Chapter 4: Design of the Secure Web System

This chapter outlines the design of secure web system which will be deployed in the cloud portal within the architecture described in the previous chapter. The secure web system is designed to provide three primary applications which will be built around the security architecture and components, also described in chapter 3. First, it provides a login for a client, which will help to establish authentication with the cloud central security server. The other two applications are secure web search and secure web browser providing security and privacy for users.

4.1 Overview of Secure Web System

The concept of a proxy architecture is used in the design of the secure web system. Furthermore, as a cloud application, it supports standard web-based protocols, security features and network protocols. This is achieved by the use of the cloud security architecture, components and protocols, described in the previous chapter. The first system component is a login client that will be installed in every client station. The purpose of this login client is to provide FIPS-196 based mutual strong authentication, using FIPS-201 compliant smart cards. Additionally, it also supports medium level assurance and low level assurance authentication for the client station with no smart card. Medium level assurance is achieved using user certificate of the web browser for medium level authentication and for low level assurance authentication it uses only username/password pair provided by the user.
Secure search and secure browsing are two cloud applications. These are simple web applications running on a web server deployed in the cloud portal. Basically they are simple proxy servers, through which a user can search or browse any remote services. When a client wants the access these applications, Portal Security Server acts as an access control point and also provides security of data and communication. Further, it also provides security module for a client browser. The security module is a software program downloaded into a client browser for the session which handles security functions related to smart card operations. This facilitates application developer by providing all the security features inside cloud security architecture without concerning of security for their individual applications. Figure 11 represents the design architecture of the secure web system with secure search and secure browse application.

4.2 Components of the Secure Web System

The secure web system comprises three components. These components are: cloud login client, security proxy server, and secure search and browse applications. Other components shown in Figure 11 are part of cloud security architecture, already discussed in Chapter 3.
4.2.1 Cloud Login Client

Cloud login client is a software program that runs on a client station. This program must be acquired by the client in some secure mechanism which is not discussed in this report. The primary function of this program is to establish secure authentication with the cloud’s central security server. Basically, mutual authentication is established with the central security server based on FIPS-196 standard. Therefore, it utilizes PIV-201 compliant smart card. Smart card contains user sensitive information, such as private key, public key certificates, authentication tokens, and cryptographic function needed for secure communication. It does not depend on the use of vulnerable external resources [22]. Thus, authentication with smart cards provides high level assurances. However, not all client stations are equipped with smart cards. In order to incorporate such clients, two other authentication mechanisms are supported by the login client.

In the second approach, the login client is able to authenticate with remote server using certificate of the client’s web browser. This authentication is only applied in the absence of smart card. Medium assurance level of authentication is achieved by such process. Lastly, if neither the smart card nor web browser certificate is available, the user is able to authenticate by providing username and password pair. Since authentication using this pair is very vulnerable, it is considered as low assurance authentication. The flow diagram of Cloud Login Client is shown in the Figure 12.
4.2.2 Security Proxy Server

Security proxy server is the component of portal security server. It is responsible for security functions, such as processing of protected message, creating of secure channel and enforcement of authorization policies. It is also responsible for supporting a security provider to client station web browser. Security provider has also an ActiveX component that uses Cryptographic Service Provider. This extends the functionality of standard web browser to communicate with smart card and perform various cryptographic functions. Hence, the standard web browser is now security enabled. The logical representation of this is shown in Figure 13.
Figure 13: Secure Communication with Portal Security Server

On the client side, the security provider is responsible for redirecting HTTP requests to and from client standard web browser. On the server side, the security proxy server handles this function. During Single-Sign-On process, client web browser will fetch SAML ticket from smart card and digitally sign with the help of ActiveX component. This is passed to Security Proxy Server through security provider. Then the process proceeds as described in Chapter 3.3. Secure session is established after the completion of Single-Sign-On protocol. In secure session all the messages are protected using PKCS#7SignedAndEnvelopedData between client station and Security Proxy Server. The first destination of any message sent by client station is always the Security Proxy Server.

It always validates signed data and opens the contents of encrypted message. If the message is addressed to itself, it can process the request and package its response back to the client. When the destination of the message is for another server, usually addressed in the URL of the message, it will simply forward that message.

Proxy architecture also facilitates storing of encrypted web pages in the web server. Web pages are stored in the PKCS#7 message format so that only Security Proxy Server is able to access them. Symmetric key is used to encrypt web pages and they are signed by the Private Key that corresponds to the digital signature certificate. When an authorized user requests web page, it retrieves the page from the web server. It unpacks the PKCS#7 protected web page and sends it
to the user through the secure session discussed earlier. The system described here is based on Secure Web System of CryptoNET [21].

4.2.3 Secure Search and Browse Application

Secure search and browse applications are simple proxies that facilitate search and browsing of some external server. The main purpose of this application is to provide security and privacy to users during this process. It is deployed in a cloud portal so that Security Proxy Server transparently handles the security function.

![Diagram of Secure Search and Browse Applications](image)

Secure search application receives search query as a HTTP request. It then submits that query to an external search engine. The external search engine retrieves search results and sends back to the secure search application as the response. That result is sent to Proxy Security Server which forwards it to the client station as a PKCS#7SignandEnvelopedData through secure session established beforehand. The browsing application takes similar approach. Client station requests a particular webpage which is addressed by its URL. This information is sent to the Proxy Security Server in a secure session which then forwards it to secure browse application. The requested webpage is retrieved by secure browse application. It is then sent to the Proxy Security Server which forwards it to the client station.

4.3 Security

Secure web system provides security in all aspects for secure search and browsing process. Primarily, it employs authentication mechanism which is very strong and robust. This system
requires the client to be authenticated with the Central Security Sever using FIPS-196 strong authentication protocol. The authentication is mutual, so that both the client and server are verified. This provides security against spoofing and repudiation attacks. Further application of smart cards gives very high level of assurance in that process. Smart cards are tamper resistance. They hold user sensitive information, such as certificates, SAML tickets, etc which could not be easily accessed by any intruder.

Access to any application, in this case secure search or secure browse, is controlled through XACML policies. Only user with valid SAML ticket is authorized to access resources in this system. SAML tickets are also digitally signed. Therefore, no intruder can tamper with or spoof them for unauthorized access. XACML policy files are stored encrypted, so that no intruder can tamper with it. Web pages are also secured in the PCKS#7 formats. This protects from tampering by any intruder.

Our secure web system also strongly protects messages that are transferred between client station and the server. Usually, this communication is protected only by the SSL/TLS. This approach does not provide adequate security in this system, because this communication goes through the Internet. Thus, PKCS#7SignandEnvelopedData message format is used to transfer message between the client station and the server. So it protects from tampering or spoofing of messages by any intruder.

4.4 Privacy

Privacy of users during search and browsing is primarily achieved through the proxy architecture. Secure search and browse application acts as a proxy for fulfilling client request. The client makes a search request using a query. This query is passed to secure search application. Then the application makes query request to a particular search engine on clients’ behalf. In this way, the search engine is unaware of the client making that particular search request. Thus, we effectively hide the location of user (IP address) initiating that search request. The same applies to the secure browse method. In this case, the application retrieves webpage from external web server and sends it back to the client station.
The other way by which the privacy is enforced is by encryption. The search request or the URL of request is encrypted in *PKCS#7* format. Therefore, any intruder will not be able to look at those data. Also, the use of SAML ticket helps to preserve the identity of a user within the system. The ticket contains authentication information asserted by an authority. It does not contain the real identity of the user for which the ticket is issued. Therefore, an intruder that is able to intercept the ticket would not be able to find out the real identity of that ticket holder.
Chapter 5: Prototype Implementation

This chapter describes prototype implementation of the applications discussed in Chapter 4. The applications that are developed are Cloud Login Client, Secure Search and Secure Browse applications. It is considered that the cloud security infrastructure provides all the security functions for the application. Since Secure Search and Secure Search application have to communicate with external search engine and web server, data transmitted from these applications to external entities cannot be encrypted. The communication between these takes places using standard internet protocol, basically HTTP. However, the communication between the application developed and the clients are protected using PKCS#7SignandEvelopedData with the help of the Portal Security Server.

5.1 Cloud Login Client

The Cloud Login Client is a Visual C++ application developed using Microsoft Visual Studio. This client must be installed in every client station and helps with establishing authentication with Central Security Server of cloud security infrastructure. The address of Authentication Server is preconfigured into the client module.

After the installation of the Cloud Login Client, its icon appears in the client station. The user has to double click on its icon to start the application. It then first checks the connection with Authentication Server specified in its configuration file. If it is not able to find the Authentication Server, the program terminates with a message indicating that the connection cannot be established. When the connection with the Authentication Server is establish, it checks for the smart card presence in the client station. If smart card is found in the client station, it presents a small dialog screen as shown in Figure 15 for entering the PIN for authentication with the smart card. The smart cards are FIPS-201 (Personal Identity Verification) compliant.

The user provides the PIN which will establish local authentication of the user with the smart card. With a series of APDU commands the application is able to fetch user authentication certificate from the smart card and thereafter transparently handle the FIPS-196 based strong
authentication with the Authentication Server. After successful authentication, it stores the SAML ticket obtained at the end of the process into the smart card.

If smart card is not found in the client station, the Cloud Login Client scans for the certificate stored in the client station. Then, the client is presented with the user-interface that lists all the certificates stored and the certificates stored in those stores. The user can choose from the list of certificates in the store to use it for the authentication process, as shown in Figure 16.

The user can also terminate browser authentication process and choose authentication with simple username-password pair. By clicking on “Cancel” button on the above interface, a simple dialogue box is presented to enter username and password, as shown in figure. When OK button is clicked, the password based authentication process begins and the program is terminated when “Cancel” is clicked.

![Figure 15: Local Smart Card Authentication Login Screen](image)

![Figure 16: Authentication with Web Browser Certificate Screen](image)
5.2 Secure Search Application

Secure Search Application is a simple proxy-based web application. It is built as J2EE dynamic web application using Servlet and Java Server Pages (JSP). The JSP page serves as visual element that is used for receiving user search query and displaying the results of that query. The Servlet is processing user search query, making search request to the appropriate search engine, collecting the results, and then dispatching those results back to the client in another JSP page.

For the purpose of this Thesis, Google search engine is used as an external search engine. The reason for selecting it is fairly simple. It is very popular search engine in most of the world [23], has huge database of indexed web pages [24], has detailed documentation for its Custom Search API\(^3\) to programmatically retrieve search result using RESTful request, and more importantly its free for limited number of search queries per day that is useful for the purpose of the Thesis.

The purpose of this application is making a proxy that hides the client from the external search engine from identifying the actual location of origination of the request. These is achieved through the use of Servlet technology that acts as a proxy for making request to the search engine and sends back results to the client upon receiving them from the search engine. In our prototype implementation, a JSP page is provided with a text area to enter a search query as shown in figure 18.

\(^3\) https://developers.google.com/custom-search/v1/overview
After a user enters a search query and presses “Submit Query” button, the query is passed to its serving Servlet. The Servlet then takes that search query and composes a search request to the search engine, in this case Google, through an API call. The syntax of making a search request is found in the Custom Search API documentation. The Servlet receives the search result of the query requested. This result is passed down to the user, originally submitting the request, and displayed in another JSP page, as shown in Figure 19.
Figure 19: Display of Search Result
5.3 Secure Browse Application

Secure Browse is a Java application. It serves the function of a streaming proxy. As a proxy, it resides in the middle between a client making a request and a remote server fulfilling that request. Whenever a user wants to browse any remote server, proxy information has to be set into the user’s browser. The proxy setting corresponds to the address of the Secure Browse application running on the Proxy Server and the port it is listening to. After setting up the Proxy Server in browser user may access any remote server by specifying its URL in the address field of the browser. The request now is directed towards the Secure Browse application, instead of directly going to that remote server address. Secure Browse application reads URL and makes request to that address on behalf of the client. The response it gets from that remote server is then passed back to the client. Thus, the remote server only sees the Secure Browse application making that request not the actual client.

With the current implementation, the streaming proxy runs at localhost (IP address of 127.0.0.1) and listens on port 8088. Then browser’s setting has to be configured so that the entire request is directed towards above address and port as shown in Figure 20.

![Figure 20: Configuring of the Browser to use Proxy](image-url)
After the setting is completed, user may enter any web address into the address bar of the web browser. This request is always first sent to the streaming proxy listening on port 8088. The streaming proxy then fetches the requested web page and streams back to the client, as shown in Figure 21. Secure Browse application at present serves only the primary function of browsing through proxy at the current implementation. Due to limited time available during this Thesis, other functionalities such as cookie manipulation have not been implemented. Cookies contain identifiable information about the client. Currently, any cookies that are passed down to the streaming proxy are sent to the remote server. Therefore, it does not hide the client and preserve the privacy of the client making the request. More work to be done in this area is left intentionally for future work of this Thesis.

![Figure 21: Browsing with Streaming Proxy](image)
Chapter 6: Conclusion and Future Work

6.1 Conclusions

This Thesis provides the concept of the secure web system in a cloud environment and applications that are delivered to users in a secure manner. With the introduction of the Central Security System, security can be provided as a service for any application running in a cloud. Thus, a cloud application developer can use security services provided by the cloud security infrastructure, such as SSO, authorization and authentication and only focus on their application logic.

Central Security System enables to perform a strong authentication protocol for any client. It provides ability to authenticate a client using smart cards, thus enhancing system’s robustness of authentication mechanisms. At the same time incorporating browser’s certificate and username/password in the authentication process, the system is able to initiate authentication process in the absence of a smart card. Thus, a flexible solution is provided for the authentication process. Messages between a client and the cloud are always protected using strong cryptographic standard. Every user in the system must be authorized to access any application in the cloud system. Central Security System and Portal Security Sever provide strong confidentiality for data and communications. With these security benefits of the Cloud Security Architecture, a secure web system is implemented which enhances security of data and communications. Users are tracked whenever they browse the Internet or search for any query. Our two privacy preserving applications were implemented in the cloud system. Security and privacy of user search or browse requests are preserved for a client by strong cryptographic encryption format for data and communication. Proxy nature of the application then hides actual clients from remote servers.

Thus, Central Security Architecture benefited to the development of a secure web system that delivers security and privacy to users while searching or browsing the Internet. Security features were mostly provided by the Cloud as a form of Security-as-a-Service and the application developed mostly focus on its logic of preserving privacy, the primary goal of the Cloud Security Architecture.
6.2 Future Work

At the end of this Thesis, there are potential future works that could be done related to the current work. Some of the important aspects are highlighted below.

- The thesis has presented authentication mechanism with browser certificate and username/password along with smart card. However, mechanism of securely storing SAML ticket using a browser certificate and username/password based mechanism is not dealt in regards to working with Central Security Server and PDP Server.

- Future work could be done in the area of delivering encrypted web pages by Portal Security Server to the client. This proposition is highlighted in the thesis, but it has not been implemented during this Thesis.

- Currently, the Secure Browse application is only streaming proxy. Client privacy can be effectively preserved only if client’s IP address and cookies are hidden during HTTP request to the remote server via proxy. This can be taken as future work for further improving the nature of Secure Browse application with regards to security and privacy.
Acronyms

**APDU** Application Protocol Data Unit  
**API** Application Programming Interface  
**APP** Application Access Point  
**CAP** Cloud Access Point  
**FIPS** Federal Information Processing Standard  
**HTTP** Hypertext Transfer Protocol  
**IDMS** Integrated Database Management System  
**IP** Internet Protocol  
**LCA** Local Certification Authority  
**MAC** Media Access Control  
**NIST** National Institute of Standards and Technology  
**PDP** Policy Decision Point  
**PEP** Policy Enforcement Point  
**PIV** Personal Identity Verification  
**SAML** Security Assertion Markup Language  
**SSL** Secure Socket Layer  
**TLS** Transport Security Layer  
**URL** Uniform Resource Locator  
**XACML** eXtensible Access Control Markup Language
References


http://www.searchenginehonesty.net/search_engine_spying.html