DEVELOPMENT OF SECURE CLOUD TRANSMISSION PROTOCOL (SCTP) ENGINEERING PHASES: MULTILEVEL SECURITY & CRYPTOGRAPHY

Dinesha H A¹ and Dr. Vinod K Agrawal²

¹PhD Student, Dept of ISE, PESIT, Bangalore, India
²Professor, Dept of ISE, PESIT Bangalore, India

ABSTRACT

Cloud computing technology provides various internet-based services. Many cloud computing vendors are offering cloud services through their own service mechanism. These mechanisms consist of various service parameters such as authentication, security, performance, availability, etc. Customer can access these cloud services through web browsers using http protocols. Each protocol has its own way of achieving the request-response services, authentication, confidentiality and etc. Cloud computing is an internet-based technology, which provides Infrastructure, Storage, Platform services on demand through a browser using HTTP protocols. These protocol features can be enhanced using cloud specific protocol, which provides strong authentication, confidentiality, security, integrity, availability and accessibility. We are proposing and presenting the secure cloud transmission protocol (SCTP) engineering phases which sits on top of existing http protocols to provide strong authentication and confidentiality using multi-models. SCTP has multi-level and multi-dimensional approach to achieve strong authentication and multi-level security technique to achieve secure channel. This protocol can add on to existing http protocols. It can be used in any cloud services. This paper presents proposed Protocol engineering phases such as Service Specification, Synthesis, Analysis, Modelling, and Implementation model with test suites. This paper represents complete integration of our earlier proposed and published multilevel techniques.

KEYWORDS:

Authentication, Confidentiality, Multi-dimensional, Multi-level, Security, SCTP;

1. INTRODUCTION

A protocol is a set of rules which governs in between two communication ports. Protocol engineering describes as an application of formal methods and software engineering in the development of communication protocol. TCP/IP, UDP and HTTP are the major existing communication protocols used in between communication parties. Many internet communication and web services are possible with the internet protocol. Cloud computing is an internet service which provides on demand platforms, storage, infrastructure and related services. It has been facing many security issues as reported in literature [1-10]. Poor identity and access management procedures, implementation of poor access control, procedures create many threat opportunities. Cloud authorization and Authentication issues are reported by TCS Innovation Labs and many
more organizations [1]. HP Labs in 2011 reported demerits on the lack of user control, unauthorized secondary usage, access, audit and lack of customer trust [2]. Accenture lab in 2011, reported concerns around cloud access authentication, authorization and access control, encrypted data communication, multi-device access, multi-level authentications and user identity management [3]. CA technologies in 2013 raised issues pertaining to identification and authentication of users’ survey calculation before granting access to cloud information or infrastructure [4]. Nelson Gonzalez1*, Charles Miers1,4, Fernando Red’iigo1, Marcos Simpl’icio1, Tereza Carvalho1, Mats N’aslund2 and Makan Pourzandi3[5], mentioned regarding identity and access management, enabling authentication for cloud solutions while maintaining security levels and availability for customers and organizations. They mentioned the user access, authentication and privacy as novel concerns [5].

Cloud computing can be improved by specialized security design and security protocol, which works with existing protocol and activates during cloud transaction to ensure strong authentication, security and confidentiality. Many cloud authentication protocols were proposed and used to avoid authentication issues. Kerberos protocol is fruitless with one flaw of replay attack. The Open ID authentication protocol is luckless in phishing vulnerabilities [6]. O Auth Protocol reports Kerberos on several aspects and thus has comparable advantages and drawbacks [6][7]. All these protocols rely on user’s memorable passwords [6]. A zero knowledge authentication protocol and Sedici 2.0 protocol is used in third party authentication, but it depends on passwords and authentication [6]. Many cloud protocols have been introduced such as Gossip protocol [7], Dynamic auditing protocol [8], Access protocol [9], Cloud Fault Tolerance Protocol [10], Cloud Gossip Protocol for Dynamic Resource Management Agent-Based User Authentication [11] and Access Control-2013 [12-14]. Many cloud authentication schemes are proposed in literature such as Graphic Password Authentication [15], Biometric Authentication [16], Secured Biometric Authentication [16,17], RFID- based authentication [18] and Eid Authentication [19,20] with their own limitations. Literature study about cloud authentication issues, security importance, proposed solution demerits motivate us to take an objective of cloud strong authentication and secure channel to achieve customer trust, privacy, confidentiality and satisfaction. In this paper, we propose to secure cloud transmission protocol, which works with existing internet protocols to ensure strong authentication, privacy, customer trust, security and confidentiality.

Proposed secure cloud transmission protocol (SCTP) has its own communication technique, technology, algorithms to achieve strong authentication, security and confidentiality [21]. SCTP may be a solution to the issues reported in literature, such as i) Authenticated Access based on user types (privileged access rights) ii) Data protection and Integrity iii) Poor identity and access management procedures, and Implementation of poor access control and procedures. Proposed SCTP has multi-dimensional password generation and authentication system [22], Multilevel Authentication technique [23] and Multi-level Cryptography system with Metadata and Lock Approach for Storing Data in Cloud [24]. This protocol can be further improved with cloud usage profile- based intruder detection system [25].

Figure 1 presents SCTP phases from requirements specification to Implementation. Different Phases are: i) SCTP requirements specifications to ensure strong authentication and secure channel ii) SCTP Service synthesis to make the error- free protocol specification and to combine multiple protocol specifications into an error- free protocol specification. iii) SCTP Modelling and specifications to present the exchange sequences, multi-level authentication,
multi-dimensional password, multi-level Cryptography and Finite state machine modelling. iv) **SCTP implementation** to take the SCTP specification and develop SCTP software modules. v) **SCTP verification** to verify if the SCTP specification actually realizes the SCTP service specification or not vi) **SCTP Validation** to check that SCTP specification does not get into deadlock, unspecified reception, and live lock errors. vii) **Conformance testing** to test the given a SCTP specification, generate test-suite for proving complete testing the SCTP functionalities.

This paper is an integration of earlier proposed and published multilevel authentication, multidimensional password generation and multilevel cryptography techniques. It is organized in the following manner: Chapter 2, presents the SCT protocol requirement specification, Chapter 3, presents SCT protocol synthesis and modelling, Chapter 4, presents SCT protocol analysis, Chapter 5, presents SCT protocol implementation modules, Chapter 6, presents SCT protocol conformance testing. Chapter 7 concludes the paper along with future enhancement.

### 2. SCT PROTOCOL REQUIREMENT SPECIFICATION AND SYNTHESIS

This section describes the SCTP requirements specification in cloud computing. Given below are the two major specifications:
Requirement Specification 1: Strong Authentication

Cloud computing provides on demand storage, infrastructure (Virtual Machines), platforms (development space) services to customers. The customer may use these services in storing their confidential data, developing important product codes, designing important software, creating software environment, testing their products/software, writing research reports and others manual, etc. Hence, each operation of customers can be considered as confidential. The vendor must make necessary arrangement in restricting the user operation based on their privileges. It shall not be accessed and breached by unauthenticated user within and outside the organization. Hence, authentication and authorization are important service specifications in cloud computing. These services and operations are only accessible to authenticated organization for its authorized users. Two important security aspects here are: i) Organization/customer must be authenticated. ii) Particular user must be authorized to perform the particular operation.

Requirement Specification 2: Secure Channel

The secure channel is an important requirement. Customer data must send or upload in secured channel. It is possible with cryptographic technique. Customer data must release from the customer place in a cipher text form. Cloud storage service acts as a container for cipher texts. The vendor provides only storages to the customer. The vendor may use much better cryptographic algorithms to store customer data in encrypted form. Though vendor has many techniques/Service Level Agreements, it is difficult to make customer understand and earn customer trust and satisfaction. Hence, we recommend customer side encryption to achieve the secure channel and trust.

We can conclude secure cloud transmission protocol with two important service specifications, they are: i) Strong Authentication ii) Secure Channel

Figure 2 shows the SCTP service specification in cloud computing services wherein it shows the major specification described above, along with the different attack scenarios. The requirements specification is presented in such a way that i) Inside attacker or Unauthorized accesses are to be restricted, ii) Outside attacker must not be able to break the confidentiality and security iii) Cloud services must be accessed by authorized customer and specific operation must permit only authenticate privileged user iv) Data uploading must take place with the only authenticated user in secure cipher form over secure channel.
SCTP string authentication executes while accessing the cloud service. SCTP secure channel executes while uploading the data to the cloud. Both are independent and can happen parallel. Hence, the error free protocol specification can be combined in SCTP service specification 1 & 2.

3. SCT PROTOCOL SYNTHESIS AND MODELING

This section describes the SCT protocol synthesis and FSM modelling.

SCT Protocol Synthesis

As described in section 2, Strong authentication and secure channel are the two main requirement specifications of SCTP. These objectives can be achieved using multi-model structure. Strong authentication can be achieved using the multi-dimensional password system and multi-level authentication (MLA) technique which are described in [22] and [23]. Secure channel can be achieved using multi-level cryptography with Metadata and Lock approach for storing data in the cloud which is described in [24]. Hence, protocol specification can be derived as i) Multi-level-Multi-dimensional password (MDP) Authentication and ii) Multi-level cryptography (MLC). While customer is accessing cloud services like PaaS, IaaS, SaaS and related [26], SCTP must execute multi-level and multi-dimensional approach. When customer initiates SaaS and related operations, SCTP must execute multi-level cryptography approach to create secure channel.
SCT Protocol Modelling

Figure 3 shows the finite state machine (FSM) modelling of SCTP.

Figure 4: FSM-SCTP

Figure 3(a)(b) shows the finite state machine for SCTP, finite states derived for both objectives. Fig 3 (a) shows the authorization and authentication finite state model using multi-level and multi-dimensional password system. It clearly mentions service usage as final/finite stage and the attacker is trap state, and remaining as a transition stage. Fig 3(b) shows the secure channel finite state model using multi-level cryptography. Send/Upload is marked as a final/finite state, stop is marked as a trap state. Remaining states refer to transition state which indicates the MLC process.

Table 1 represents the state transition table for FSM-SCTP (a) where it shows cloud usage final state feasible with only valid transition. Table 2, represents the state transition table for FSM-SCTP (b) which presents secure channel state transition. Algorithm 1 SCTP_STRONG_AUTHENTICATION presents the steps to create strong authentication. Algorithm 2 SCTP_SECURE_CHANNEL represents steps to achieve the secure channel.
Table 1 state transition table for FSM-SCTP (a)

<table>
<thead>
<tr>
<th>Current State</th>
<th>Input</th>
<th>Next State</th>
</tr>
</thead>
<tbody>
<tr>
<td>Generate MDP</td>
<td>S</td>
<td>Check MDP For authentication</td>
</tr>
<tr>
<td>Authenticate MDP</td>
<td>S</td>
<td>Enter into MLA</td>
</tr>
<tr>
<td>1(^{st}) Level Authenticate</td>
<td>S</td>
<td>Enter Second Level</td>
</tr>
<tr>
<td>2(^{nd}) Level Authenticate</td>
<td>S</td>
<td>Enter Nth Level</td>
</tr>
<tr>
<td>Nth Level Authenticate</td>
<td>S</td>
<td>Allow for cloud usage</td>
</tr>
<tr>
<td>Any above state</td>
<td>Fail</td>
<td>Consider as attacker/misbehave</td>
</tr>
</tbody>
</table>

Table 2 state transition table for FSM-SCTP (b)

<table>
<thead>
<tr>
<th>Current State</th>
<th>Input</th>
<th>Next State</th>
</tr>
</thead>
<tbody>
<tr>
<td>Metadata</td>
<td>Yes</td>
<td>Cryptography</td>
</tr>
<tr>
<td>Cryptography</td>
<td>Yes</td>
<td>Lock</td>
</tr>
<tr>
<td>Lock</td>
<td>Yes</td>
<td>Send/Upload</td>
</tr>
<tr>
<td>Any above state</td>
<td>No</td>
<td>Stop Sending/SaaS</td>
</tr>
</tbody>
</table>

Algorithm 1: SCTP\_STRONG\_AUTHENTICATION

Input: Image1..n, Text1..n, MDPi
Output: D, SaaS, PaaS, IaaS

Step 1: Cloud connection establishment between vendor and customer over internet

C\(\leftarrow\)V

Step 2: Generate Multi-dimensional password system using multi-format inputs for multiple levels

K\(_1\) \(\leftarrow\) MDP\_Generation (Image1, Text1, Image n, Text n) /*To detect outside attacker*/

K\(_2\) \(\leftarrow\) MDP\_Generation (Image1, Text1, Image n, Text n) /*To detect inside attacker*/

K\(_n\) \(\leftarrow\) MDP\_Generation (Image1, Text1, Image n, Text n) /*To ensure access privileges and grant access*/

MDP\_Generation (Image1, Text1, Imagen, Textn)
{
//Arithmetic/Logic Operation for input Images
If (Images number > 1) then
  Img\_Opn= Arithmetic\_Logic(Image1,Image2..Imagen)
//Extract Features of final image after arithmetic or logic operation i.e Img\_Opn
ImageFeature= Extract Features (Img\_Opn)
// Combine all text inputs in random way
If(texts_number>1) then
PasswordText= RandomMix(Text1,Text2,...Textn)
// Combine final text and final features of image
MDP=Combined _File (ImageFeature, PasswordText)
Return MDP
}

Step 3: Allow to enter MDP input, Check entered MDP input against Generated MDP data base i.e K1, K2,K3..Kn

Initialize j←1
Repeat the below steps until j=n
MDPi← Entered MDP Input
IF MDPi=Kj THEN
Then j←j+1
ELSE
Flag←J
Go to Step 4
END IF
Go to Step 5

Step 4: Analyze the flag value
Switch (Flag)
case 1:
D←Trigger True positive and display outside attack;
break;
case 2:
D← Trigger true positive and display inside attacker;
break;
case n:
D← Trigger true positive and display misused privileges;
End Switch

Try again! Go to step 3

Step 5: Allow accessing cloud services and performing intending operations.
Confirm Concatenated MDPi (1,2...,n) = Concatenated (K1,K2...Kn)
Then C←SaaS, PaaS, IaaS

Algorithm 2: SCTP_SECURE _CHANNEL

Input: plain data Pd
output: cipher data Cd

Step 1: Start Multilevel cryptography executes metadata, encryption and end with lock
Md← Metadata (Pd)
Repeat below step till all encryption level completes

\[ Ed \leftarrow \text{Encrypt (Md, Key)} \]

\[ Cd \leftarrow \text{Lock}(Ed) \]

**Step 2: Send, upload to cloud SaaS**

- IF (Cd) THEN
  - \[ \text{SaaS} \leftarrow Cd \]
- ELSE
  - Goto Step 1
- End

**Step 3: While retrieving Cd, MLC executes Unlock, Decrypt, Reverse Metadata**

- Ed \leftarrow \text{Unlock (Cd)}

Repeat below step till all decryption level completes

\[ Md \leftarrow \text{Decrypt(Ed,Key)} \]

\[ Pd \leftarrow \text{~Metadata (Md)} \]

**Step 4: End of Multi-level cryptography**

**SCTP Verification & Validation**

The above FSM, transition and algorithms verify and confirm the requirements specification of strong authentication and secure channel discussed in section 2. MDP-MLA and MLC techniques are applied, modelled in wireless sensor-cloud integration using an ant colony routing algorithm using Petri-nets theory [27][28].

**SCTP Validation against deadlock:** Deadlock occurs, when processes chain waits for some action to occur, when resources are occupied and not released and when complex and parallel action takes place. SCTP MLA, MDP and MLC functionalities are designed to happen one after another, no resources and inputs are shared, no parallel action takes place as it travels from level to level, hence there will be no deadlock. This issue has been taken care of. The unspecified reception isn’t happening due to protocol design consideration for internal fixed levels and inputs from a specific limited user. The unspecified reception issue is incorporated.

**SCTP Validation against live lock:** Livelock is a special case of resource starvation. In SCTP, we can correlate livelock resource starvation for MDP input/process, MLA movement and MLC data/process. MDP-MLA occurs in sequence from level to level. No parallel level executes at a time. MLC designed in an independent way in sequence, hence livelock issue may not occur.

4. **SCT PROTOCOL ANALYSIS**

This section analyzes the SCTP protocol MDP password strength for MLA authentication. It analyses secure channel and its performance by creating attacker, unauthorized user scenarios.
Applying set theory for analyzing MDP in MLA, let us assume 3 inputs for MDP generation and 3 levels like L1, L2 and L3 for authentication. MDP generation takes both images (img1, img2,.. Imgn) and texts (text1, text2 ..textn) passwords. The universal sets U = { img1, img2, img3, img4, img5, text1,text2,text3,text4} can be declared to handle 3 MDP in 3 MLA. The corresponding sub sets for each level are derived below:

L1= {img1, img2, text1}
L2= {img3, text2, text3}
L3= {img4, img5, text4}

Then, the venn diagram is as in figure 4

MLA=Concatenate (MDP (L1), MDP (L2),MDP (L3)) //Final authentication for accessing the service.
Refer to MDP MLA venn diagram 4 figure, MDP confidential inputs for different levels are different in nature, L1,L2 and L3 are considered as disjoint sets. \[1\]

Refer to universal sets the U can be

\[ U = L1 \cup L2 \cup L3 \quad & \quad \text{Null} = L1 \cap L2 \cap L3 \]

Then
Number of sample space (total confidential data) can be defined as number of confidential inputs at each individual levels L1, L2 and L3 as defined below:

\[ n(s) = n(L1 \cup L2 \cup L3)=n(L1) + n(L2) + n(L3) \]

Number of confidential data required to break the system is \( n(s) \).

We used probability theory to analyze SCTP features and estimates the probability of breaking. Probability models can greatly help system in optimizing multi- mode levels and making safe decisions to have proved MLD MDP and MLC.

Since it is disjoint and algorithm, designed to not to occur simultaneously, L1, L2, L3…..Ln is considered as a mutually exclusive event.

\( A1, A2, \cdots, An \) is a finite sequence of mutually exclusive events in \( S \) (\( A1 \cap \cdots \cap Aj = 0 \) for \( i\neq j \)), then probability to break the system ‘p’ can be derived as below:
Probability to break each level is \( P(A_i) \), levels may go up to ‘\( \infty \)’. ‘p’ can be probability of breaking the complete system.

Now, to analyze the probability to get successful three inputs to break each level using brute force or dictionary attack which contain ‘n’ guessed inputs. Analyze the probability to get successful n(L1UL1UL3) in a guessed ‘m’ dictionary or brute force lists. The derivation after applying the law of probability formula for MLA_MDP. Probability of breaking event E can be defined as probability of levels \( P(L_1), P(L_2), \ldots P(L_n) \) breaking.

\[
P(E) = P(L_1)P(E/L_1) + P(L_2)P(E/L_2) + P(L_3)P(E/L_3) + \ldots + P(L_n)P(E/L_n)
\]

Probability of breaking each level with ‘n’ confidentiality input breaks is referred as \( p(A_k/E) \). It can be defined using Bayes theorem as below:

For \( n=1,2,3,..n \) levels,

\[
P(An|E) = \frac{P(An)P(E|An)}{P(E)}
\]

\[
\Rightarrow \frac{P(An)P(E|An)}{P(E)} = \frac{P(L_1)P(E/L_1) + P(L_2)P(E/L_2) + P(L_3)P(E/L_3) + \ldots + P(L_n)P(E/L_n)}{P(E)}
\]

Above Bayes theorem expression proves that probability of breaking proposed MLA system i.e \( P(An/E) \) depends on the attacker success on each levels, i.e \( L_1, L_2, \ldots L_n \). i.e on \( P(L_1)P(E/L_1), P(L_2)P(E/L_2), P(L_3)P(E/L_3) + \ldots + P(L_n)P(E/L_n) \) events. Hence, it proves the criticality of breaking confidential inputs.

**Analyzing the Strength of Multi-dimensional Password**

The National Institute of Standards and Technology (NIST) password meter tries to estimate the entropy of a password mainly based on their length. A password strength meter is a function i.e \( f=\sum x \rightarrow R \), that takes as input a string (or password) \( x \) over an alphabets \( \sum \) and outputs a number \( s \), a score, which is a measure of the strength of string \( x \) as a password. The output is, in general, a real number indicating the password strength.

Consider currently using normal textual password length \( L \), from a set of \( N \) possible symbols. The number of possible passwords can be found by increasing the number of symbols to the power \( L \), i.e. \( N^L \). Increasing either \( L \) or \( N \) will strengthen password. The strength of a random password can be measured by the information entropy is just the base-2 logarithm or \( \log_2 \) of the number of possible passwords. Assuming each symbol in the password is produced independently, a random password's information entropy, \( H \), is given by the formula as below:

Where, \( N \) is the number of possible symbols, \( L \) is the number of symbols in the password. \( H \) is measured in bits. In MDP inputs such as images, texts and operations gives increased \( N \) and \( L \) which results in increased strength compared to textual passwords.
The strength of a password is the amount of work an adversary needs to break the password. Consequently, the optimal strategy for an attacker is to guess passwords in increasing order of strength and decreasing order of probability, i.e., more likely passwords are tried before less likely ones. This also motivates the definition of guessing entropy, which gives the average number of passwords an attacker has to guess before finding the correct one. Let $X$ be a random variable with finite domain $D$ and $P(X = d_i) = p_i$, ordered with decreasing probabilities $p_i < p_j$ for $i < j$. The guessing entropy $G(X)$ is defined as

$$G(X) = \sum_{i=1}^{\left|D\right|} (i \cdot p_n(x = i))$$

**An Ideal Password Strength Meter Definition 1**

Let us fix probabilities $P: \Sigma^* \rightarrow [0, 1]$ on the space of passwords (i.e., strings over a certain alphabet). An ideal password checker $f(x)$ is given by the function $f(x) = -\log(P(x))$. We denote this password strength meter as “ideal”, as the order which is given by this function is the same as the order with which passwords are guessed in an optimal guessing attack. Consequently, the following two functions $f'(x) = 1/P(x)$ and $f''(x) = R P(x)$ also constitute ideal password checkers, where $R P(x)$ is the rank of $x$ according to the distribution $P$, i.e., if the probabilities $p_i = P(x_i)$ are ordered with $p_i < p_j$ for $i < j$, then $R P(x_i) = i$.

An adaptive password strength meter (APSM) proposed in [29] $f(x, L)$ is a function $f: \Sigma^*(\Sigma^*)^k \rightarrow \mathbb{R}$, that takes a string (or password) $x$ over an alphabet $\Sigma$ and a password file $L$ containing a number of passwords as input and outputs a score $S$. Password database $L$ contains a number of passwords sampled from the same distribution, and the task of the password strength meter is to estimate the strength of the password $x$ based on his estimation of $P$ [29].

Over the last years, Markov models have proved to be very useful for computer security in general and for password security in particular. For example, Narayanan et al. [29] showed the effectiveness of Markov models to password cracking [29].

For breaking the password:

$$f(c) = -\log_2(\prod_{i=1}^{n} P(ci|ci-1,..ci-1))$$

MLA-MDP password generated in multi-level with combinations of special and alphanumerical characters in more than 30 characters. The generated password in the above equation, proves MDP strength.

**Strong Authentication**

The multi-dimensional password generation system uses multiple inputs to generate password and generated password is used in authenticating the user in multiple levels. Hence, to analyze the proposed protocol authentication and security, we used probability theory. Below section presents the (a) probability of breaking the MDP-MLA authentication and getting strong authentication and (b) probability of breaking MLC security and getting secure channel. We compare our work to three different password checkers that are widely used today- the NIST, Google and Microsoft.
password checkers. They were chosen because of their popularity and because they are representative for the techniques employed currently for password strength meters.

(a). Probability of breaking the authentication

To break the MDP-MLA system one should know, all the ‘N’ confidential inputs to create MDP’s for the all the ‘M’ levels. MDP=1,2,…N MLA = 1,2,… M. Hence, the probability of breaking by identifying N inputs in M levels is found. Probability of breaking MDP-MLA can be \( P(\text{B})/P(\text{S}) \) = i.e \( 1/2^N/M \) for N=4 and M=4 the result will be \( \Rightarrow \frac{1}{16} \Rightarrow 0.015625 \).

Below table 3 executes the proposed authentication with different M inputs and N values.

<table>
<thead>
<tr>
<th>Values</th>
<th>Meaning</th>
<th>Formula</th>
<th>Conclusion</th>
</tr>
</thead>
<tbody>
<tr>
<td>N=0 M=0</td>
<td>No security</td>
<td>1</td>
<td>Certain to break</td>
</tr>
<tr>
<td>N=1 M=1</td>
<td>Single input password for only 1 level</td>
<td>0.5</td>
<td>Chance to break</td>
</tr>
<tr>
<td>N=2 M=2</td>
<td>Two input password for 2 levels</td>
<td>0.125</td>
<td>Difficult to break</td>
</tr>
<tr>
<td>N=3 M=3</td>
<td>Three input password (MDP) for 3 levels</td>
<td>0.0416</td>
<td>Unlikely to break</td>
</tr>
<tr>
<td>N=4 M=4</td>
<td>Four input password (MDP) for 4 levels</td>
<td>0.015625</td>
<td>Strong Authentication</td>
</tr>
</tbody>
</table>

Figure 4: MDP-MLA Strong authentication

Figure 4, represents the strong authentication with increased inputs and levels. MDP MLA achieves strong authentication as the number of inputs and levels increases. Initial M=N=0 value creates zero authentication, M=N=1 generates weak system, M=N=2 produces an average system, M=N=3 and 4 creates a strong authentication system. MDP_MLA system is stronger than single level authentication or 2 levels with single/double input password. It cannot be broken with dictionary and other brute force and similar attacks.
EXPERIMENTATION & DISCUSSION

Case Study Example: Consider a university ‘X’ as a customer for keeping all its affiliated college’s student’s admission, examinations, marks card and other related data. The huge student’s information cannot be maintained in single PC or server. Cloud Storage service can be used for best repository. It may be private or public cloud storage service. Let us take marks card data as a sensitive data which should not modify or update by unauthenticated user. A single password authentication, biometric, smart card, single level authentication cannot be used to access the marks card storage as it may breach confidentiality. Confidentiality cannot be depend on single person and privileges. So let us, introduce multi-level and multi-dimensional password authentication system which authenticates users in multiple levels and authorizes access privileges for university marks card storage.

MDP authentication for accessing cloud services such as Storage as a service could be the best solution to authenticate and authorize the user. Using multiple authentication, in a first level/Top level, Vice chancellor/University Head authenticates the cloud services, In second level Examination, head authenticates second level for accessing the data inside the department, third level (may be bottom level) is used to authenticate and authorize privileges to perform a particular operation. At each Level, MDP techniques take place. Generated MDP has to be tested in multiple levels. MDP generator uses images and texts combination. Images may be university unique logo, Individual signature in image form, Seal etc. Text may be confidential textual inputs like password, university name and etc.

Experiment: Assuming the number of levels is 3, multi-level and multi-dimensional inputs are 3 in each level, image arithmetic and logic operation as add, random mix function for texts, Consider the table below and let us do the operation.

Discussion:

Since images are defined by company individuals, Image size, RGB, Pixel and other image factors are difficult to guess. Even if a hacker breaks for single image, they will have multiple image challenges along with textual passwords to break. Assuming the successful hacking of all MDP inputs, the hacker will face the challenge of breaking multiple levels. Hence, single hackers and system cannot do this. Brute force attack and dictionary attack can identify the textual passwords, not images, even with further improvements of attack breaking the images multilevel break cannot be done.

Experimentation

Assuming a number of levels and number of inputs to MLA MDP is 3, Image operation is multiplication and table below is assumed confidential data, then the concatenated password obtained in all three levels is described below:
### Table 4: Secure channel with MLC

<table>
<thead>
<tr>
<th>Values</th>
<th>Meaning</th>
<th>( P(B)/P(S) = 1/2^N )</th>
<th>Conclusion</th>
</tr>
</thead>
<tbody>
<tr>
<td>N=0</td>
<td>No security</td>
<td>1</td>
<td>Certain unsecure channel</td>
</tr>
<tr>
<td>N=1</td>
<td>Single level encryption</td>
<td>0.5</td>
<td>Chance to make it unsecure</td>
</tr>
<tr>
<td>N=2</td>
<td>Cryptography with lock</td>
<td>0.25</td>
<td>Difficult to unsecure</td>
</tr>
<tr>
<td>N=3</td>
<td>Cryptography, lock and metadata</td>
<td>0.125</td>
<td>Unlikely to unsecure</td>
</tr>
<tr>
<td>N=4</td>
<td>2level cryptography, lock and metadata</td>
<td>0.0625</td>
<td>Fully secure channel</td>
</tr>
</tbody>
</table>

**Figure 5: MLC Secure Chanel**

Comparison of multi-level authentication with single level authentication when brute force attacks and dictionary attacks happen for 10 to 10000 times. Shown below is the derived formula to compare single level attacks and multilevel attacks for 10 - 10000. \( N \) times [24], where, \( n \)=number of times attacks, \( j \) = number of success, \( p \) = probability of success in each try \( b(n,p,j) = \binom{n}{j} p^j q^{n-j} \). The comparison graph is shown in figure 6.
5. SCT PROTOCOL IMPLEMENTATION MODULES

This section describes SCTP implementation modules. In software engineering, UML class diagram is very well used to represent the system before actual software development. Hence, we use a class diagram to represent the SCTP implementation modules. Class diagram describes a SCTP object that shares the same attributes, operations, relationships, and semantics.

Figure 7: SCTP class diagram
Figure 7 shows the SCTP class diagram where major modules of the protocol are presented as classes. SCTP is the main class which has MDP, MLA and MLC as subclasses. SCTP attributes are derived from sub classes called constructors. Each constructor initiates the class objects to perform the MDP, MLA and MLC operations.

6. SCT PROTOCOL CONFORMANCE TESTING

This section describes the SCTP testing suites, which confirm the requirement of SCTP. It also presents verification and validation test cases for SCTP. It describes the major positive, negative and security test cases required to test protocol. Basic functionality verifications, testing with negative values and security test have been done. Test cases are derived from major functionalities of SCTP i.e MLC, MDP and MLA. Hence, completeness of test cases can be confirmed by Table 5, containing 20 major test cases. It tests the SCTP MDP, MLA and MLC functionalities. Test cases prefixed with MDP refer to MDP basic functionalities. Test cases prefixes with MLA refer to MLA basic functionalities. Test cases prefixed with MLC refer to MLC basic functionalities. Hence, it confirms the complete SCTP Testing.

Table 5: SCTP functionality test cases

<table>
<thead>
<tr>
<th>S.No</th>
<th>Test Case And Type</th>
<th>Description</th>
<th>Expected Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>MDP_GENERATE, ‘+’ve</td>
<td>MDP operation to generate MDP</td>
<td>The function should create MDP password by taking confidential inputs.</td>
</tr>
<tr>
<td>2.</td>
<td>MDP_AUTHENTICATE, ‘+’ve</td>
<td>MDP operation to authenticate MDP</td>
<td>The function must authenticate the given password with MDP password.</td>
</tr>
<tr>
<td>3.</td>
<td>MLA_LEVEL_MOVE, ‘+’ve</td>
<td>MLA operation to move the levels</td>
<td>Levels should move to its inner level as MDP authenticates.</td>
</tr>
<tr>
<td>4.</td>
<td>MLA_PASSWORD_FLOW, ‘+’ve</td>
<td>PF operation to carry passwords from one level to the next level</td>
<td>The password should carry to its lower level when successfully authenticates happens at a higher level.</td>
</tr>
<tr>
<td>5.</td>
<td>MLA_ALLOW_ACCESS_ATEND, ‘+’ve</td>
<td>Allow operation to get successful access</td>
<td>After clearing all levels with a valid password, user must be able to access the service.</td>
</tr>
<tr>
<td>6.</td>
<td>MLC_CREATE, ‘+’ve</td>
<td>MLC operation to create secure channel</td>
<td>Must create a secure channel by generating cipher texts.</td>
</tr>
<tr>
<td>7.</td>
<td>MDP_NO_INPUT, ‘-’ve</td>
<td>Test with no inputs and random texts</td>
<td>It should be able to generate an error message to get confidential inputs.</td>
</tr>
<tr>
<td>8.</td>
<td>MLA_NULL_AUTHENTICATE, ‘-’ve</td>
<td>Test with null</td>
<td>It should not authenticate the null.</td>
</tr>
<tr>
<td></td>
<td>Test Case Description</td>
<td>Expected Outcome</td>
<td></td>
</tr>
<tr>
<td>---</td>
<td>-----------------------------------------------------------</td>
<td>--------------------------------------------------------</td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>MLA_NULL_PASSWORD_LEVEL MOVEMENT, '-' ve</td>
<td>Test to move levels with null value</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>No level should move when null MDP enters</td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>MLA_INVALID_INPUTS, '-' ve</td>
<td>Test with random invalid inputs</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Must notify the vendor and customer about invalid input try (true positive)</td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>MLA_DIRECT_LEVEL_JUMP, security</td>
<td>Test to jump to lower level directly</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Must not allow to jump lower levels and trigger true positive</td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>MLC_DIRECR_LEVEL_JUMP security</td>
<td>Test to jump higher levels directly</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Must not allow for higher level operation and trigger true positive</td>
<td></td>
</tr>
<tr>
<td>13</td>
<td>MLA_ZERO_LEVEL, '-' ve</td>
<td>Test with 0 value for in MLA</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Should generate error message to initiate MLA levels when it is zero</td>
<td></td>
</tr>
<tr>
<td>14</td>
<td>MLC_ZERO_LEVEL, '-' ve</td>
<td>Test with 0 value in MLC</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Should generate error message to initiate MLC levels when it is zero</td>
<td></td>
</tr>
<tr>
<td>15</td>
<td>MLC_NO_METADATA, '-' ve</td>
<td>Test with 0 METADATA value</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Should generate error message to initiate MLC Metadata when it has no value</td>
<td></td>
</tr>
<tr>
<td>16</td>
<td>MLC_NO_LOCK_KEY, '-' ve</td>
<td>Test with no lock key and unlock key</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Should generate error message to initiate MLC lock/unlock key when it has no value</td>
<td></td>
</tr>
<tr>
<td>17</td>
<td>MLC_NO_CRYPTO_KEY, '-' ve</td>
<td>Test with no encryption/decryption key</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Should generate error message to initiate MLC crypto key when it has no key value</td>
<td></td>
</tr>
<tr>
<td>18</td>
<td>MLC_LOCK_UNLOCK, '+' ve</td>
<td>LOCK/UNLOCK operation for locking and unlocking files</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>MLC LOCK and Unlock operation must execute properly</td>
<td></td>
</tr>
<tr>
<td>19</td>
<td>MLC_METADATA, '+' ve</td>
<td>Operation to arrange/rearrange data</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>The MLC Metadata operation must arrange/rearrange properly</td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>MLC_RM_CRYPTOGRAPHY Cancel, security</td>
<td>Test to remove MLC operation for ML cryptography</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>MLC cryptography must create an error message and trigger true positive</td>
<td></td>
</tr>
</tbody>
</table>
CONCLUSION AND FUTURE ENHANCEMENT

Secure cloud transmission protocol can be used in a cloud computing transaction where the strong authentication and secure channel are the mandatory requirements from the customer. It can be used as an add on to the existing HTTP protocol. SCTP achieves strong authentication by means of MDP_MLA system. SCTP achieves secure channel by means of MLC. This protocol solves access rights, authentication, privilege, access, authorization, privacy, confidential issues and achieves customer trust and satisfaction. SCTP is also a solution to brute force attacks, dictionary attack, phishing attack and it provides strong security. This protocol can be enhanced by adding usage profile-based intruder detection and prevention system. This paper can be further enhanced with detailed investigation, comparisons, usefulness and advantages of proposed techniques.

ACKNOWLEDGMENT

Our sincere thanks to Prof. K N B Murthy, Principal and Prof. Shylaja S S, HOD, Department of Information Science and Engineering, PES Institute of Technology, Bangalore, for their constant encouragement.

REFERENCES

[4] Ponemon Institute, Security of Cloud Computing Users Study, CA Technologies Independently conducted by Ponemon Institute, LLC Publication Date: March 2013


Cloud-based RFID Authentication, Wei Xie1, Lei Xie2, Chen Zhang1, Quan Zhang1, Chaojing Tang1. 2013 IEEE International Conference on RFID, 978-1-4673-5750-0/13, 168-175, 2013.


Adaptive Password-Strength Meters from Markov Models, Claude Castelluccia, Markus D’urnmuth, Daniele Perito.