Securely Available Credentials Framework in Cloud Computing Environments

Jaejung Kim*, Seng-phil Hong**, Bong Gyou Lee***, Joon Suk Hwang***

*Department of Computer Science, Sungshin University, Seongbuk-gu, Seoul, Korea
E-mail: jajukim@hotmail.com

**Department of Computer Science, Sungshin University, Seongbuk-gu, Seoul, Korea
E-mail: philhong@sungshin.ac.kr

***Graduate School of Information, Yonsei University, Seodaemun-gu, Seoul, Korea
E-mail: bglee@yonsei.ac.kr, hjs6040@naver.com

Abstract

Due to increasing uses of credentials like certificate through smart phones and smart pads for user authentication in various e-government services and online based financial sector, it is essential to have service provider systems, which allows to access services through a variety of devices. This paper analyzes problems and limitations of the current securely available credential standards. Furthermore, it is also our intention to design N-screen based securely available credential framework (SACF) for internet services that meets framework and protocol requirement of credentials in a cloud computing environments. We designed the secure SACF architecture in cloud computing environments, which not only provides more flexible credentials roaming and digital signing framework but also leads to safer credential management in operating various mobile devices such as smart phone, smart pad, etc.

Key Words: Credentials, Public Key Infrastructure, Cloud Computing, Authentication, N-screen

1. Introduction

Accredited certificate is the most commonly used method for user authentication in various e-government services and online based financial sector[1, 2]. Due to increasing usages of credentials like accredited certificate through smart phones and smart pads, it is necessary to develop new technologies which enable the credentials to utilize in various smart devices and cloud computing environments[3]. In this paper, it is our purpose to suggest a safe and convenient securely available credentials framework that mobile device users can effortlessly use credentials in cloud computing environments. After this brief introduction, the reminder of this paper is organized as follows; section 2 discusses the security issues in clouds and M2M (Machine to Machine) environments and the status of current securely available credentials. Then we propose securely available credentials framework with architecture and protocol is shown in section 3. Section 4 presents overall prototyping of proposed framework and the comparison and verification of our architecture. Finally we conclude the paper and suggest the future research work in section 5.
2. Related Work
2.1 Security issue for clouds and M2M

As cloud computing encompasses many technologies including networks, databases, operating systems, virtualization, resource scheduling, transaction management, load balancing, concurrency control and memory management, there are numerous security issues for cloud computing. Therefore, security issues for many of these systems and technologies are applicable to cloud computing[4]. M2M, which refers to machine to machine, generally means wired or wireless communications, and communications between devices controlled by man and machines. Security issues such as the standardization of smart device authentication, a common security monitoring and control services, human-centric privacy, the security of M2M communication between the various smart devices are occurred in the communication model of society based on smart devices[5].

2.2 Securely available credentials

The SACRED (Securely Available Credentials) Working Group is working on the standardization of a set of protocol for securely transferring credentials among devices. SACRED standards consist of RFC 3157 (Requirements) [6], RFC 3760 (Credential Server Framework) [7], and RFC 3767 (Protocol) [8].

Limitations and problems of existing SACRED standards are both the existing SACRED standards do not provide a detail implementation guideline although it defines the framework and protocol requirements with respect to securely available credential and SACRED protocols do not define a protocol to create a digital signature from credential server using uploaded credential by Client. To solve these problems, let us redefine a secure credential framework that meets the framework and protocol requirements of SACRED and design a credential protocol. In addition, we would like to define a protocol of credential roaming and proxy signature which fits in cloud computing environments.

3. Securely Available Credentials Framework (SACF)
3.1 SACF architecture

The entities of Securely Available Credentials Framework consist of Client, type of devices, Credential Server (CS), Signing Server (SS), and Credential Store(s). Module components are AMM (Account Management Module), CRM (Credential Roaming Module), PSM (Proxy Signature Module, and PCM (Policy Compliance Module) [9].
3.2 Protocol framework

A. Initialization and key sharing operations

1) Bind Protocol / Info Protocol

If Client has his own digital certificate, he can exchange a session key between the Client and Credential server using certificate and private key of both sides. If Client doesn’t have the certificate, Credential Server sends a session key after the server creates secure channel through SSL/TLS or DH key exchange.

B. Account management operations

1) Create Account Protocol

When Client creates user account, Credential Server registers hash value which contains both clients’ unique identifier such as social security number and unique device information such as serial number or MAC address that is used by PC, smart pad, smart phone, etc.

2) Modify Account Protocol and Remove Account Protocol

Client can register new device information or modify the registered device information. If the Client no longer uses the account, a registered account can be removed.

C. Credential roaming operations

1) Credential Upload Protocol

The registration process that Client uploads credentials to Credential Server is as below.
2) Credential Download Protocol from Credential Server

In order to use the credential in a variety of environments, the download procedure of credentials from the credential server is as below.

D. Proxy signature operations

1) Proxy Signature [10] Protocol

The signing process of Client using Signing Server is as below:

Notations and abbreviation for protocol design is as follows;

Table 1. Notations and abbreviation for protocol design

<table>
<thead>
<tr>
<th>Category</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Function</td>
<td>H(): Hash Function, E(): Encrypt Function, D(): Decrypt Function, S(): Create Signature, V(): Verify Signature, £: Compare with</td>
</tr>
</tbody>
</table>
4. Prototyping and Verification

4.1 Implementation and simulation

To demonstrate the feasibility of our architecture, we implemented a prototype system which provides securely credentials management system for a certificate and private key of user. This system is developed using JSP, JAVA, iPhone development toolkit technologies. This image below shows SACP’s signing procedure using iPhone’s Application and user certificate saved in server.

![Image of iPhone user interface of SACP](image)

**Fig. 2. iPhone (iOS) user interface of SACP**

SACP: User’s device requests proxy signature to Signing server and service provider verify that result. PKI: User’s device generates digital signature and service provider verify it. The SACP provide more a time saving and enhanced security than traditional authentication using PKI by providing a centralized authentication and signing method.

Table 2. Comparison between PKI and SACP

<table>
<thead>
<tr>
<th></th>
<th>PKI</th>
<th>SACP</th>
</tr>
</thead>
<tbody>
<tr>
<td>User authentication</td>
<td>Slow</td>
<td>Fast</td>
</tr>
<tr>
<td>Smart Phone</td>
<td>800ms</td>
<td>21ms</td>
</tr>
<tr>
<td>PC</td>
<td>400ms</td>
<td></td>
</tr>
<tr>
<td>UNIX</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Sign verify</td>
<td>16ms</td>
<td>8ms</td>
</tr>
<tr>
<td>Authentication method</td>
<td>Provided by each service</td>
<td>Centralized management</td>
</tr>
</tbody>
</table>

4.2 Comparison and verification

The SACP compare with the old user authentication model as follow;

Table 3. Comparison between Old model and SACP

<table>
<thead>
<tr>
<th>Category</th>
<th>Memory requirements</th>
<th>Process requirements</th>
<th>Proxy signature</th>
</tr>
</thead>
<tbody>
<tr>
<td>Old model</td>
<td>Δ</td>
<td>Δ</td>
<td>X</td>
</tr>
<tr>
<td>SACP</td>
<td>O</td>
<td>O</td>
<td>O</td>
</tr>
</tbody>
</table>

O: provided  Δ: partially provided  X: not provided
5. Conclusion and Future Work

Following is contributions of N-screen based securely available credential framework for internet services that meets framework and protocol requirement of credentials in a cloud computing environments. We designed the secure SACF architecture in cloud computing environments, which not only provides more flexible credentials roaming and digital signing framework but also leads to safer credentials management in operating various mobile devices such as smart phone, smart pad, etc.

The future study will continue to focus the design and implement of our suggested framework, and we will expand to new devices and environments.

References


* First author : Jaejung Kim, E-mail: jajukim@hotmail.com
** Corresponding author: Sengophil Hong, Ph.D.
School of Information Technology, Sungshin Women’s University,
2 Bomun-ro 34ga-gil, Seongbuk-gu, Seoul, Korea
E-mail: philhong@sungshin.ac.kr