Authentication System for Online Examination Using Keyed-Hash Message Authentication Code and Multi-agent System

Yin Lai Winn Tint, Swe Zin Hlaing
University of Computer Studies, Yangon
yin.lei.winn.tint@gmail.com,szh.ucsy@gmail.com

Abstract

Agents are independent pieces of software capable of acting autonomously in response to input from their environment. A multi-agent system is a system composed of numerous agents. Besides, it can be used to solve problems which are difficult for an individual agent to solve. With the growth of Internet and distributed computing, more and more web applications employed agent based applications. Security is an important issue for the widespread development of application based on software agent. When users were simply accessing information on a network, simple passwords were usually sufficient to protect information. However, the increasingly distributed computing environment are often operated by multiple parties and password-cracking programs are available that can break typical passwords in a short period of time. Consequently, the need for stronger levels of authentication has become critical. This paper describes agents that used Keyed-Hash Message Authentication Code (HMAC) to authenticate valid user to ensure proper access and control of the web based online examination agent.

1. Introduction

Recent development in Web Technologies and using AI techniques to support efforts in making the web more intelligent and provide higher-level services to its users leads to the agent based intelligent web application. Multi agent system is one composed of multiple interacting software components known as agents, which are typically capable of cooperating to solve problems that are beyond the abilities of any individual member. Two classes of services are crucially needed for a secure Internet infrastructure. These include access control services and communication security services. Access control service protects Internet resources from unauthorized use, whereas communication security service ensures confidentiality and integrity of data transmitted over the network, in addition to nonrepudiation of services to the communication entities. An important prerequisite for access control is user authentication, the process that establishes the identity of a user. In the context of the Internet, we assume authentication is handled by the communication security services. Without proper authentication, an attacker can personate anyone and can gain access to the resources maintained by the agent based web application. A Message Authentication Code (MAC) is basically a message digest with an associated key. It produces a short value based on its input data and the key. In theory, someone with the same key can produce the same MAC from the same input data.

This paper presents the agent based online examination system ,which is designed to give the quick way of assessing the approximate level of a student's knowledge of English grammar and usage. The remainder of this paper organized as follows. We describe about agents and web application in section 2. In section 3, authentication is presented. Keyed-Hash Message Authentication (HMAC) is proposed in section 3. Proposed system is stated in section 4.Conclusion is given in section 5 and references are given in section 6.

2. Related work

In the recent progress, agent technology has been applied in many interesting web-based services like decision support systems and virtual learning environment, Internet applications. Many security mechanisms and technologies have been developed for enhancing the security of user authentication. The author Xi-Yang has researched one-time password based remote authentication dial-in user service (RADIUS) authentication in a generic security service application programming interface (GSS-API). The focus of this research is to investigate the use of one-time password and RADIUS authentication as a protection facility for a GSS-API mechanism [6].

Woo T. Y. C., and S.S. Lam pointed out mechanism and implementation for authentication system in distributed system [5]. Security for multi-agent system based on various cryptographic algorithms was presented in [1]. It focuses on message security for agent.

The authors Somchart Fugkeaw, Piyawit Manpanpanich, and Sekpon Juntapremjitt have proposed the public key infrastructure (PKI) based authentication scheme and the multi-agent technique for authentication approaches to support multi-clients in using a multi-application environment for controlling access role to client [3].

3. Authentication

Authentication is the process to identify and assure the genuine, authorized user in the system. In other words, authentication is the great challenge and extremely important in most network applications. The authentication methods have been categorized into the followings.

User name / Password authentication: Usernames and passwords are the most common form of authentication in use today for its simplicity.

Key-file-based authentication: Instead of using passwords, a system can authenticate by using encrypted "key-file", which normally has far larger data size than typical password, thus more secure. For example, SSH (Secured Shell) client can connect and login to the server by using key file instead of a password. In this case, a user cannot login by password, and must prepare a valid key file for the authentication.

Biometrics authentication: Password is handy for authentication, but is actually a very weak method because anyone who knows the password can login to a system. In case the system needs to recognize a particular person, we should use some forms of biometrics authentication like Fingerprint, Hand palm geometry, etc.

In the proposed system, in order to avoid the weakness in username/password based authentication, HMAC mechanism has been applied to be the strong and secure authentication system.

3.1 Keyed-Hash Message Authentication Code (HMAC)

HMAC is a method of ensuring that a message was generated by someone with access to a shared secret. HMAC makes use of some sort of one-way hashing function (like MD5 or SHA-1) to encrypt the secret along with a message.

Let ***H*** be the hash function. For simplicity of description we may assume *H* to be MD5 or SHA-1.*H* takes inputs of any length and produces l-bit output (for l=128 for MD5 and 160 for SHA-1). Let **Text** denote the data to which MAC function is to be applied and let **K** be the message authentication secret key shared by the two parties. It should not be larger than 64 bytes, the size of a hashing block, and, if shorter, zeros are appended to bring its length to exactly 64 bytes. MAC used to further different 64 byte strings ipad and opad.

ipad = the byte 0x36 repeated 64 times

opad = the byte 0x5C repeated 64 times

The function HMAC takes the key **K** and **Text** and produces $HMAC_k$ (**Text**) =

$$H(K \oplus \text{opad}, H(K \oplus \text{ipad}, \text{Text}))$$

Namely,

1. Append zeros to the end of the K to create a 64 byte string
2. XOR (bitwise exclusive-OR) the 64 byte string computed in step (1) with **ipad**
3. Append the data stream **Text** to the 64 byte string resulting from step (2)
4. Apply H to the stream generated in step (3)
5. XOR (bitwise exclusive OR) the 64 byte string computed in step (1) with **opad**
6. Append the H result from step (4) to the 64 byte string resulting from step (5).
7. Apply H to the stream generated in step (6) and output the result.

The recommended length of the key is at least *l* bits. A longer key does not add significantly to the security of the function, although it may be advisable if the randomness of the key is considered weak.

HMAC uses the hash function H as a black box. No modifications to the code for H are required to implement HMAC. This makes it easy to use library code for H and, also makes it easy to replace a particular hash function, such as MD5, with another, such as SHA, should the need to arise.

3.2 Message Digest 5 (MD5)

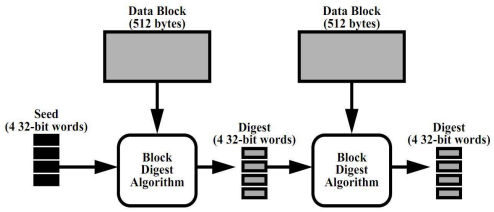

Figure: 1 MD5 block-chained digest algorithm.

MD5 is an extensively used cryptographic hash function with a 128-bit hash value. A MD5 hash is typically expressed as a 32 digital hexadecimal number. MD5 algorithm takes input message of arbitrary length and generates 128-bit long output hash. MD5 hash algorithm consists of 5 steps.

- Append Padding Bits
- Append Length
- Initialize MD Buffer
- Process Message in 16-Word Blocks
- Output

MD5 is a block-chained digest algorithm, computed over the data in phases of 512-byte blocks organized as little-endian 32-bit words (Figure 1). The first block is processed with an initial seed, resulting in a digest that becomes the seed for the next block. When the last block is computed, its digest is the digest for the entire stream. This chained seeding prohibits parallel processing of the blocks [2].

Each 512-byte block is digested in 4 phases. Each phase consists of 16 basic steps, for a total of 64 basic steps. Each step updates one word of a 4-word accumulated digest, using the entire intermediate digest as well as block data and constants.

4. Proposed System

The proposed system is composed of four agents named collector agent, provider agent, database agent and rank maker agent. Collector agent is to collect the students' information and communicate with the proper agent to go on the process based on the user requirements. Provider agent generates HMAC for authenticated student access and provides questions and necessary information for the students. Rank maker agent evaluates the students' marks and ranks the suitable level for the student. Database agent can be asked by other agents whenever they want to take all the transactions with the server database.

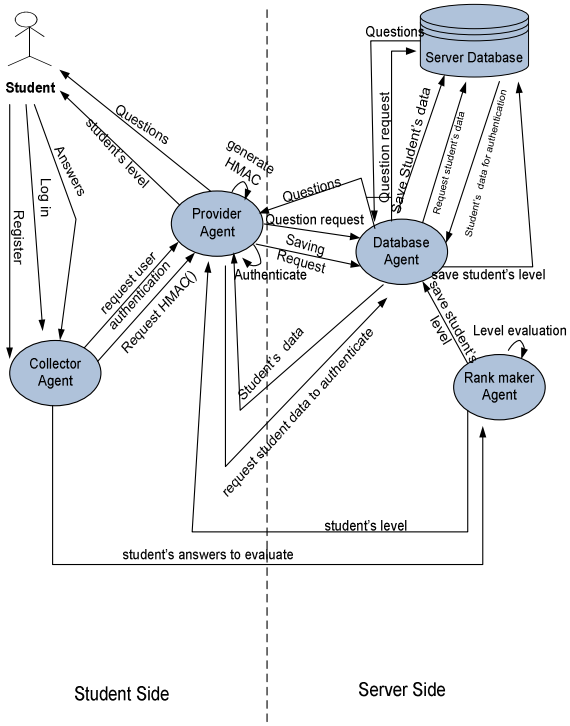

Figure: 2 Proposed system

4.1 Registration Process

Every student, who is not the member of the system, has to register first. Registration process of the system requires you to provide the information such as user name, password, NRC, father name etc. After the form is submitted, the server first verifies that the user is unique and then the system creates a HMAC by digesting user's unique information (user name, password and NRC). The resulting HMAC value is stored in the database. The system does not store the password in the database .Moreover, if the user take the exam many times within a month, they can guess answers and their actual English level is difficult to be determined. So, the system records the timestamp of the user's last log in order to avoid taking the exam repetitively during one month. Registration process of the system takes as shown in Figure 3.

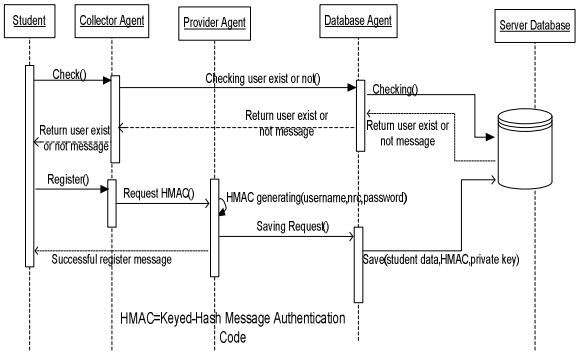

Figure: 3 Registration Process of the system

4.2 Log in process

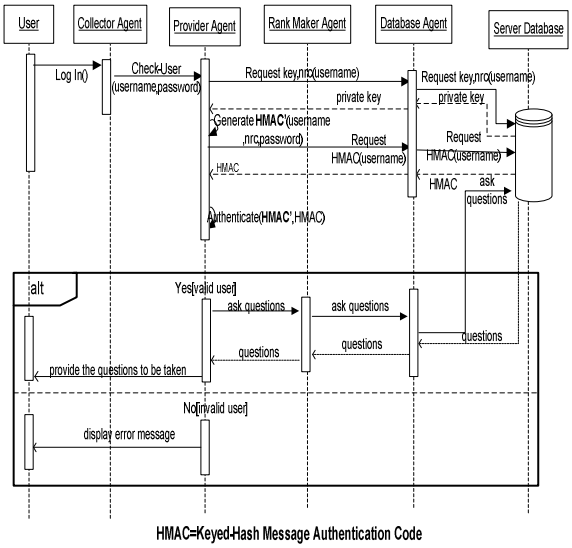

Figure: 4 Log in Process of the system

Once the registration is complete, the user has to log in using the user id and password given at the time of registration. When the user is logged into the system, the system performs the authentication as follows.

- It first retrieves NRC from the stored database for that user.
- It creates HMAC hash value on user data (user name, password and NRC) and the resulting hash value is sent to server along with user name.
- The system checks the corresponding HMAC hash value for that user. If the HMAC value from the client

and HMAC value from the database is identical, the system still checks the user's last log in timestamp to meet the system's time constraint as we mentioned above. If the timestamp is okay, user is allowed to take the exam and can enroll the class via online.

Log in process of the system performs as shown in Figure 4.

4.3 Database Design of the System

There are five tables used in the proposed system. They are student, result, timetable, course and question tables. Student table is used to record student information such as student name, login name etc. The most important fields in the student table are hmactext, privatekey and NRC which are used for authentication process. Last login field in the student used to hold the date and time when user last logged in. Result table is used to store the final level of the student. Timetable is used to hold timetable information for each course and course table presents the course information like course name and duration of such courses. Question table is used to store questions for every level.

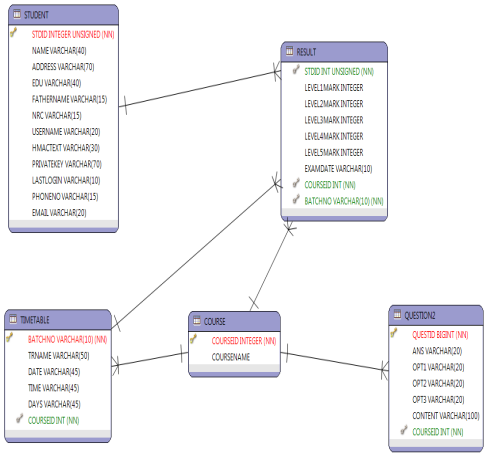

Figure: 5 Database design

4.4 Analysis

Figure 6 shows the execution times for various cryptographic algorithms in JCE package. The chart shows that HMAC-MD5 is the faster than all other cryptographic algorithms, and it is well suited for the online agent authentication system. The bar with various colors showed the message size and execution time for the algorithm base on their input size is presented in the left hand size. In Table 1, the execution time of the algorithms are in accordance to the fact that stream ciphers (e.g. RC4 algorithm) execute faster than block ciphers (e.g. DES algorithm) and cryptographic hash functions (e.g. HMAC-MD5) have least execution time than others. Execution time was measured in seconds for each algorithm based on the presence and absence of I/O (input, output) file sizes.

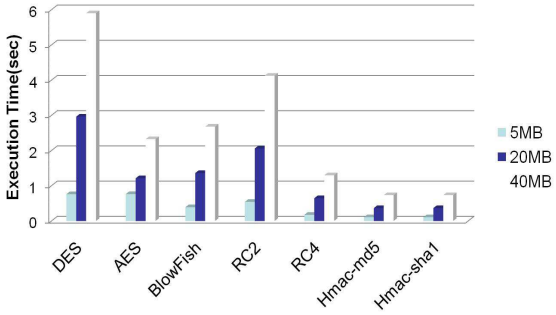

Figure: 6 Analysis of execution time for JCE encryption without File I/O for three different input file sizes

Table: 1 Results of Execution time for JCE Encryption with I/O and without I/O for three different input sizes.

JCE algorithms	With I/O			Without I/O		
	5Mb	20Mb	40Mb	5Mb	20Mb	40Mb
DES	1.705	7.174	11.673	0.761	2.964	5.902
AES	1.539	4.200	7.771	0.764	1.215	2.326
Blow Fish	1.185	2.854	7.404	0.391	1.362	2.680
RC2	1.325	4.877	8.946	0.544	2.060	4.126
RC4	1.025	3.288	6.509	0.177	0.648	1.299
Hmac-MD5	0.731	2.320	3.905	0.107	0.367	0.734
Hmac-SHA1	0.731	2,320	3.905	0.107	0.367	0.734
RSA(1024)	10.966			10.511		
RSA(2048)	11.353			11.008		

5. Conclusion

The security of the agent in Web environment is one of the most important issues to protect improper access to the system. Cryptographic algorithm could be used in conjunction with agent technology to improve the security level of the multi agent system. In this paper, the online examination of rating English skills that used keyed-hash message authentication code (HMAC) for clients' authenticity has been discussed. All in all, this kind of system could be said secured authentication system that saves user's cost and time because of the independence of classroom, time and place.

6. References

[1]Faith Tekbacak, "Developing a security mechanism for software agent", Master Thesis, Izmir Institute of Technology, July 2006.

[2] Rivest, R., “The MD5 Message-Digest Algorithm,” RFC-1321, MIT LCS and RSA Data Security, Inc., April 1992.

[3]Somchart Fugkeaw, Piyawit Manpanpanich, and Sekpon Juntapremjitt "Multi-Application Authentication based on

Multi-Agent System", IAENG International Journal of Computer Science, 33, 2.May 2007.

[4]Vani Karimijji, "Analysis of capabilities and performance of JAVA Cryptography Extension (JCE)".

[5]Woo, T. Y. C., and S.S. Lam, "Authentication for Distributed Systems", Volume 25, Number 1, January 1992.

[6] Xi Yang, " The Use of One-Time Password and RADIUS Authentication in a GSS-API Architecture", Master Thesis KTH Information and Communication Technology, Sweden 2006.