A secure electronic examination protocol using wireless networks

Jordi Herrera-Joancomartí, Josep Prieto-Blázquez, Jordi Castellà-Roca
Estudis d'Informàtica i Multimèdia
Universitat Oberta de Catalunya
Av. Tibidabo 39, 08035 Barcelona
Email: {jordiherrera,jprieto,jcastellar}@uoc.edu

Abstract

Electronic examinations are those examinations performed through a computer where questions and answers are computer files rather than sheets of paper. The temptation of using electronic examinations as a distance learning evaluation tool is obvious. However, security concerns must be considered. In this paper, we present a secure electronic examination protocol. Using wireless technology, we propose a trade-off solution between examination security (in terms of examination result correctness) and examination flexibility.

Keywords: *electronic examinations, wireless networks, security, cryptography, e-learning.*

1 Introduction

Different approaches can be taken to define electronic examinations. We identify the concept of an electronic examination with those examinations that are performed through a computer and that produce an examination solution that can be stored in an electronic file. The main property of electronic examinations is that both questions and answers are in electronic format. This fact has interesting implications in terms of managing examination information. For instance, in a test based examination [3], the examination evaluation can be automated and the results can be obtained just after finishing the examination.

The benefits of electronic examinations are specially important in an e-learning or virtual learning environment where students, teachers and the institution itself are widespread and connected using computer networks. Having examination information in electronic format simplifies all information management processes.

Virtual electronic examinations, where examinations are performed within the distance paradigm, are difficult to implement due to the complexity of student authentication and control. In fact, virtual electronic examinations are a superset of electronic voting, a more studied field [5]. On one hand, there is the user authentication problem, since in both cases, electronic examinations and electronic voting, the user must to be indisputably authenticated. Further more, in electronic voting, the vote must be a personal decision so vote coercion or vote sale must be prevented. In the same way, virtual examinations must ensure that the authenticated user is the one and only that solve the examination.

For that reason, virtual electronic examinations implemented so far rely either on a trusted student model (for instance CNAP model [2]) or in a trusted environment model. The trusted student model assumes students do not cheat and follow all the rules properly. However, in some scenarios, this could not be a realistic model since student trust cannot be assumed. In fact, students themselves reject the trusted student model because beyond social institution acknowledgement, evaluation correctness ensures credibility of the degrees that the institution issues.

The trusted environment model for virtual electronic examinations needs a trusted physical place in order for students to sit examinations. Furthermore, the trusted environment model assumes that computers and platforms used in electronic examinations are trusted. Those constraints reduce drastically the potential virtuality of examinations. However security goals can be achieved.

In this paper we propose a secure electronic examination protocol. Our protocol allows students to take electronic examinations in a trusted environment. Wireless networks are used in order not to rely on fixed infrastructure. The proposed secure examination protocol can be implemented in a mobile electronic examination lab that can be installed with no cost and without any infrastructure needs. This property is extremely important for pure e-learning institutions that do not have any fixed "real infrastructure and rent buildings in order to conduct their student examinations.

0-7695-2108-8/04 $ 20.00

The rest of the paper is organized as follows. Section 2 describes a general view of our secure electronic examination protocol. In Section 3 descriptions of entities and protocols are presented. In section 4 security of the proposed electronic examination protocol is discussed. Finally, conclusions are presented in section 5.

2 General description

The scenario for our system is a standard university classroom with a wireless access point connected to a server (namely examination server, ES). In case the classroom has wired network, the server can be placed anyway in the network otherwise, the server can be placed in the classroom connected to the access point. Furthermore, every student will be provided with an examination terminal, ET, a computer with a wireless card. For portability reasons, examination terminals are conceived of as a laptops that do not have major installation requirements, even for power supply.

Since electronic examinations need an inherent infrastructure (basically computers and networks) the maximum flexibility for that infrastructure is the use of mobile devices and wireless networks. If communication between examination terminals and examination server is performed through wireless networks, mobile electronic examination lab portability is ensured since no infrastructure in needed in the classroom where the examination is performed.

The general examination protocol is depicted in Figure 1.

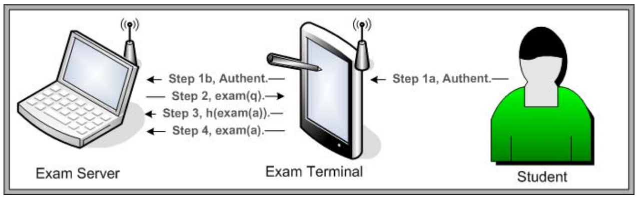

Figure 1. The general examination protocol

The examination set-up starts when every student chooses one of the examination terminals to take the examination. Previously, the student identity has been checked by the teacher using traditional means like for instance and personal ID university card. Then, the student authenticates to the examination server through the examination terminal using the authentication protocol (Step 1a, 1b). After that, the student is ready to take the examination.

When the examination time begins, the examination server executes the examination download protocol (Step 2) where examination questions, $exam(q)$, are sent to the examination terminal. Students have the specified time to solve the examination.

Once the student finishes the examination, examination answer $exam(a)$ must to be sent to the examination server. Prior to sending the examination answer, the student executes the examination registration protocol (Step 3) in order to obtain a time-stamp over the examination answer hash $h(exam(a))$. This time-stamp will ensure examination answer correctness at a given time in case some problems occur during its transmission. The examination upload protocol (Step 4) sends the examination answer to the examination server and the examination server will return a reception receipt. Finally, examination questions and answers are deleted from the ET.

3 Architecture description

Three different entities and four protocols can be identified in this architecture.

3.1 Entities

Students are those who want to sit the examination. Main requirements for students are due the authentication protocol (see Subsection 3.2). More precisely, since the authentication protocol uses public key cryptography, students must have a key pair of a public key cryptosystem, namely $\{PK_T, SK_T\}$ and a certificate, $Cert_T$, issued by a trusted third party.

Different approaches can be used to enable users to perform digital signatures, from USB key tokens to smart cards or even mobile devices [7].

The **examination terminal** is a computer the student uses to answer the examination. Basic examination terminals requirements are the following:

1. Examination terminals must enable students to answer examination questions.
2. Wireless connection between examination terminals and the examination server is also required.
3. Examination terminals must allow students to authenticate in front of the examination server.

First requirement determines the device type and depends on the examination format itself. For instance, a multiple-choice test can be conducted on a PDA, while a normal examination must be performed in a device with keyboard. Furthermore, if the examination requires handwriting, a tablet PC can be used.

The wireless connection requirement implies examination terminals must be provided with wireless cards, for instance a IEEE 802.11g PCMCIA card for a laptop device.

The last requirement assumes that students can provide to examination terminals all information needed in the authentication protocol (passwords, keys, certificates, etc.).

As we pointed out before, the authentication protocol could be performed by a student mobile device, so examination terminals should be able to communicate with those personal devices, for instance using WPAN [1] technology like Bluetooth.

We assume the examination terminal is provided by the examiner so it is not a student personal device. In this way, the device software can be more controlled.

The **examination server** is a computer that performs all administration operations during the examination. The examination server requirements are listed below:

1. Examination server must to be able to perform student authentication.
2. Examination server must have enough memory to store all examination questions and answers.
3. Examination server must provide a time-stamp service to register the examination (see examination registration protocol in subsection 3.2).
4. Examination server must provide simultaneous examination terminal connections through a wireless network.

The examination server is the equivalent of the teacher in traditional examinations. The examination server is responsible for student authentication, custody of examination questions and answers and time examination control.

3.2 Protocols

Four main protocols can be identified in our architecture.

The **authentication protocol** is the most important protocol. Its implications are relevant from the security point of view and for system performance since authentication may allow or may not allow students to take the examination.

Student authentication can be performed in different ways but for the correctness of the process we use a challenge-response authentication scheme based on public key cryptography which provides strong authentication. That means every student must have a key pair of a public key cryptosystem as it has been pointed out in Section 3.1.

Following the ideas presented in [7], the authentication protocol is a mutual entity authentication protocol based on the one proposed by Diffie *et alter* [4]. The protocol is a three pass Diffie-Hellman variant that establishes a shared session key between the student and the examination server. At the end of the protocol, the student has been authenticated in front of the examination server and also the examination server has been authenticated in front of the student.

The authentication protocol is depicted in Figure 2. For simplicity, we assume examination terminal acts on behalf of the student and whenever necessary the student provides sensible information like private keys.

$$
\begin{aligned}
ET \longrightarrow ES \quad &: \quad g^{r_T} && (1)\\
ES \quad &: \quad K = (g^{r_T})^{r_S} \\
ET \longleftarrow ES \quad &: \quad g^{r_S}, E_K\{Sig_S(g^{r_S}, g^{r_T})\}, Cert_S && (2)\\
ET \quad &: \quad K = (g^{r_S})^{r_T} \\
ET \longrightarrow ES \quad &: \quad E_K\{Sig_T(g^{r_S}, g^{r_T}), Cert_T\} && (3)\\
ET \longleftarrow ES \quad &: \quad E_K(Id_{E1}, Id_{E2}, \cdots, Id_{En}) && (4)\\
ET \longrightarrow ES \quad &: \quad E_K(Id_{Ei}) && (5)
\end{aligned}
$$

Figure 2. Authentication protocol

Subscripts are used to identify the owner of each element, so PK_T stands for the public key of the student (managed by examination terminal) while r_S is a random number generated by the examination server.

In step (1), the examination terminal takes g a generator of a multiplicative group in which discrete logarithms are hard to compute. Then it generates a random value r_T and then g^{r_T} is sent to the examination server. The examination server then computes a symmetric key $K = (g^{r_T})^{r_S}$ using a random value r_S. In step (2) the examination server computes g^{r_S} and signs (g^{r_S}, g^{r_T}) with his private key SK_S. The resultant signature is encrypted using a symmetric algorithm with key K generated in step (1). The value g^{r_S} together with the certificate of the examination server $Cert_S$ and the encrypted value of the signature is sent to the examination terminal. Then the examination terminal can compute the symmetric key K, he obtains the digital signature and validates it. In step (3) the examination terminal signs g^{r_S}, g^{r_T} with the private key SK_T and encrypts the resulted signature and the certificate $Cert_T$ with the shared key K. At this stage both student and examination server are authenticated. In step (4) the examination server sends to the examination terminal identification Id_{Ei} of all possible examinations (even could be only one) the student can perform in this classroom, that day at the given time. The identification Id_{Ei} contains course information like name of the subject, term of the examination, etc. Finally, in step (5) the student chooses and commits to the examination to be taken.

The **examination download protocol** takes place at the initial examination time. Figure 3 depicts the process.

The examination server sends examination questions, $exam(q)$, to the examination terminal. The information sent is encrypted using the session key K in order to en-

$$
\begin{array}{rcl}
ES \longrightarrow ET & : & M = E_K(exam(q)|t) \\
ET & : & D_K(M) = exam(q)|t \\
ES \longleftarrow ET & : & Sig_T\{h(exam(q)|t)\}
\end{array}
$$

Figure 3. Examination download protocol

sure examination server authentication. Message, M, contains also the current time, t. The examination terminal decrypts the message, M, and digitally signs a examination questions hash followed by the temporal tag. The resulted signature is returned to the examination server as a receipt in order to prove the beginning examination time.

After the execution of this protocol, the student has the examination questions in her terminal and can start to answer them.

The **examination registration protocol** is performed once the student has finished the examination and before the examination is sent to the examination server in the examination upload protocol. The examination registration protocol is a time-stamp protocol that ensures integrity of the data, examination answer $exam(a)$, at a given time.

A time-stamp protocol is a two-step protocol (like the one depicted in Figure 4) where in the first step the user sends the data to be time-stamped and in the second step the time stamp authority, in our case the examination server, returns a time-stamp structure for the given value.

$$
\begin{array}{rcl}
ET \longrightarrow ES & : & h(exam(a)) \\
ET \longleftarrow ES & : & timstp = Sig_S\{h(exam(a))|t)\}
\end{array}
$$

Figure 4. Examination registration protocol

Any time-stamp proposal protocol [6, 9] can be used as a examination registration protocol.

Although the registration operation can be performed during the examination upload, we execute it before in order to use a standard time-stamp protocol. Furthermore, the examination answer file size could cause overload network problems during the examination upload. For instance, a typical examination with 150 students sending all the examination files at the same time gives a 45 MB traffic, assuming a 5 page handwrite examination produced by a tablet PC takes 300 KB in average. Using a examination registration protocol before the upload itself, only a typical 160 bit hash string transaction is needed to ensure what exactly the student did until the ending examination time.

Once examination answer is time-stamped, examinations can be upload even after the examination ending time avoiding network overload.

During the **examination upload protocol**, the student sends the examination answer to the examination server encrypted with the shared key K. The examination server decrypts the file, computes the hash value of the received information and ensures it matches with the hash value he has previously time-stamped. In that case the examination server sends a acknowledgement receipt to the student. Then, examination questions and answers are deleted from the ET.

Figure 5 depicts the process:

$$
\begin{array}{rcl}
ET \longrightarrow ES & : & M = E_K(exam(a)) \\
ES & : & D_K(M) = exam(a) \\
ES & : & h(exam(a)) \overset{?}{=} h(exam(a)) \\
ET \longleftarrow ES & : & OK
\end{array}
$$

Figure 5. Examination upload protocol

4 Security assessment

As we pointed out at the beginning of the paper, our electronic examination protocol follows the environment trust model rather than the student trust model. That means every device used in our protocol is a trusted device. This fact is important since restrictions on the connections between examination terminals and between examination terminals and other networks can be controlled. Furthermore, having ET as a trusted device implied that possible attacks from those devices are not possible since the examination time that will have a student to corrupt the ET is normally too short.

In next subsections we analyze every protocol involved in the architecture and we describe security properties they achieve.

4.1 The authentication protocol

The adoption of the protocol described in section 3.2 satisfies different security properties as it is pointed out in [8].

It provides mutual authentication since both the examination terminal and the examination server must authenticate each other through their digital signatures and certificates. Examination authentication is important since the student must to be sure the examination he is taken is a correct one served by the real examination server.

Furthermore, the authentication protocol generates a shared key between both parties. Key generation is performed by mutual agreement so joint key control, mutual implicit key authentication and mutual assurance of key freshness is provided. This shared key is used to protect the communication channel between the examination server and the examination terminal.

Finally, confidentiality of student identity is provided against an eavesdropping attack, since identity information is encrypted with the shared key before its transmission.

Security properties described above prevents different possible attacks such as source substitution attack, signer verification attack, content verification attack and codebook attack, among others (see [8] for details).

4.2 Examination download protocol

The examination download protocol uses symmetric key encryption to authenticate the examination terminal. This fact reduces the computational cost of the examination server since avoid public key operations. The key length used in this protocol does not need to be too long, even it is for authentication purposes, since the time slot between the authentication protocol, when the key is generated, and the examination download protocol is typically short.

Examination download protocol also ensures the initial time of the examination. Notice that the student digitally signs the hash value of the examination followed by a time tag. This signature proves that the student starts that particular examination at a given time.

An important security issue is the fact that students should not be able to communicate between each other during the examination. This communication obviously include oral communication, but also computer based communication.

4.3 Examination registration protocol

Security of examination registration protocol is obtained because a standard time-stamp protocol is used in order to register the examination. Notice that the protocol ends with a registration receipt of examination answer hash. Such receipt verifies the exact examination answer content at the ending examination time.

4.4 Examination upload protocol

The examination upload protocol allows the exam be send to the examination server. Confidentiality and integrity of examination answers is ensured since communication between examination terminal and the examination server is encrypted using the shared key generated during the authentication protocol.

Furthermore, examination answers integrity with respect the answer file generated at the ending examination time is also ensured. Notice that the examination server check hash value of the received examination answer is the same hash value he has previously received. This mechanism ensures all students have the same time to finish taking the examination and that time does not depend on who upload the examination first or on the examination server connection speed.

5 Conclusions and further research

In this paper a secure electronic examination protocol using wireless networks has been presented.

Our electronic examination protocol follows the environment trust model rather than the student trust model. That means every device used in our protocol is a trusted device and student trust is not required. Although this could seem a restriction, wireless technology together with mobile devices allows us to create a mobile examination lab that can be easily installed in any room.

Further research will be focused in increasing examination flexibility but maintaining the same level of security. For example by allowing examination terminals be untrusted devices, then students could use their own labtop to take the examinations students.

Acknowledgments

We would like to thank the anonymous referees for helpful comments on the submitted version of this paper. This work is supported by Spanish MCYT and FEDER funds ref. TIC2001-0633-C03-03 STREAMOBILE.

References

[1] IEEE Std 802.15.1, June 2002. `http://ieee802.org/15/pub/TG1.html`.

[2] Cisco networking academy program - cnap. 2003.

[3] I. G. L. Consortium. Ims question and test interoperability specification. Stand:24.07.2002, 2003.

[4] W. Diffie, P. C. van Oorschot, and M. J. Wiener. Authentication and authenticated key exchanges. *Designs, Codes and Cryptography*, 2(2):107–125, June 1992.

[5] C. E. S. Group. e-voting security study. Issue 1.2, Jul 2002.

[6] N. W. Group. Rfc 3161: Internet x.509 public key infrastructure time-stamp protocol. Internet Activities Board, 2001.

[7] J. Herrera-Joancomart'ı and J. Prieto-Blázquez. A personal authentication scheme using mobile technology. In *Proceedings of the Information Technology: Coding and Computing ITCC'2003*, pages 253–257. IEEE Computer Society, 2003.

[8] K. M. C. Horn, G.; Martin. Authentication protocols for mobile network environment value-added services. *IEEE Trans. on Vehicular Technology*, 51(2):383–392, March 2002.

[9] A. G.-T. A. R. K. Wouters, B. Preneel. Towards an xml format for timestamps. In *Proceedings of the ACM workshop on XML Security*, 2002.