Access Control for Shared Remote Laboratories

Verónica Mateos, Luis Bellidoa and Víctor A. Villagrá

Dpto. de Ingeniería de Sistemas Telemáticos, Universidad Politécnica de Madrid, Madrid, Spain
Email: {vmateos, lbellido, villagra}@dit.upm.es

Thomas Richter and Alberto Gallardo

University of Stuttgart, Stuttgart, Germany
Email: {richter, gallardo}@ rus.uni-stuttgart.de

Remote experiments can be shared and exchanged among educational institutions in the form of reusable learning objects. The paper discusses different solutions for authentication and authorization needs when "Lab Learning Objects" are used through the Learning Management System of an institution to run a remote experiment based on physical resources (rigs) provided by an external experiment provider organization. The solutions are based on the use of SCORM as a framework for learning objects, the definition of a third-party rig booking system and the use of Shibboleth as an optional framework for a federated authentication and authorization.

Keywords: *remote experiment, access control, authorization, LMS, booking system*

ACM Classification:

Categories and subject descriptors: *C.2.4 [Computer-Communication Networks]: Distributed Systems – client/server, distributed applications; D.2.11 [Software Engineering]: Software Architectures – domain-specific architectures; D.4.6 [Operating Systems]: Security and Protection – access controls; K.3.1 [Computers and Education]: Computer Uses in Education – computer-assisted instruction (CAI); K.6.5 [Management of Computing and Information Systems]: Security and Protection – authentication*

General terms: *Design, Management, Security*

1. Introduction

Sharing remote laboratories on-line is a need acknowledged by many educational institutions. Remote laboratories are often based on scarce and expensive rigs, physical resources that can typically be used only by one person or cooperating group at a time. When teachers and students from the institution owning a rig are its only users, access control can be based on closed and simple solutions. But rigs are often idle most of the time, so opening the access to these rigs for users in other institutions can be beneficial for both the institution owning the rig and the institutions accessing a piece of equipment that can be costly. It is then necessary to implement new access control mechanisms for the rigs to make it simple for the institutions owning rigs to share the remote laboratory, so external users can run experiments accessing their rigs.

This paper discusses the design of a technical solution for controlling the access to remote laboratories based on the definition of a rig booking system and three different user roles: content

Manuscript received: 29 December 2011
Communicating Editor: José-Luis Sierra-Rodríguez

providers who are providing experiments using a rig at their institution premises, teachers who are including remote experiments in the curriculum of the subjects they teach, and students who access the remote experiments.

This document is structured as follows. Section 2 provides a discussion of the pedagogical frameworks supported by sharing access to remote laboratories. Section 3 gives an overview on the technical framework. Section 4 describes the architecture proposed to share remote experiments and gives an overview of the rig booking system. Section 5 discusses different solutions related to the authentication and authorization for accessing remote laboratories. Section 6 describes how the solutions have been implemented in the LiLa project (LiLa, 2011). Finally, Section 7 provides a discussion of the proposal in relation to existing work and some conclusions and future work in this area.

2. Pedagogical Implications

One of the main current movements in education motivating the work described in this paper is the use of open educational resources (OERs). Bissell and Ahrash (2009) describe OERs as "digitized materials offered freely and openly for educators, students, and self-learners to use and reuse for teaching, learning and research". Additionally, remote laboratories have increased in use and become a common feature in universities throughout the world. The technical solutions described in the paper aim at supporting the view of a remote laboratory as a kind of OER.

As with any other type of OERs, it is important to define what is the granularity of the access to a remote laboratory. An OER may be an entire course, a complete book, or a more granular piece, such as a single learning object (Downes, 2007). The use of experiments in education can range nowadays from teacher-centred education to student-led education. A whole range of learning scenarios can be thought of within the two ends of the spectrum. In order to accommodate the different learning scenarios, our focus is on accessing remote laboratories using "Lab Learning Objects" (LLOs). Thus, most of the discussion is focusing mainly on the technical aspects of the access to the remote laboratory. However, as it is explained in the following section, these LLOs are based on the SCORM standard, which means that they can be easily integrated into popular learning management systems (LMSs), such as Blackboard and Moodle.

The decision on focusing on LLOs is also related to the level of reusability and the challenge of contextualisation for learning objects. A very simple learning object offering access to a laboratory not contextualised within a pedagogical context has less effectiveness than a highly contextualised learning object. On the other hand, simpler resources can be easier to contextualise and reuse, because they need no "de-contextualisation" and can be adapted directly for a specific purpose (Watson, Coble, Bhave, Smallbone and Kraft, 2012). A possible framework for creating and using LLOs is to include a low level of contextualization within the LLO so teachers referring to remote experiments during a class can give specific assignments that can be carried out with the experiments, with the support of other tools found in current LMSs. But LLOs can also be used in other frameworks, for example in scenarios using online collaborations that do not necessarily involve pedagogical institutions, in which different resources and tools are used by students to get acquainted with theory in an engaging way to discuss problems.

The number of OERs produced worldwide is growing. One can look at the more than 200 institutions of higher education that are members of the OpencourseWare consortium as of April 2012 (http://www.ocwconsortium.org). However, sharing remote experiments as LLOs is not as

straightforward as sharing other kinds of OERs, e.g., sharing a video or audio version of a lecture on YouTube or a set of slides on slideShare. Hilton (2010) believes that a baseline definition of 'open' for an OER should mean that it can be freely reused and redistributed. This would facilitate the use of OERs in different pedagogical frameworks. However, LLOs have necessary access restrictions, because they are giving access to physical resources that can typically be used only by one person or cooperating group at a time. In Section 7 we compare the technical solutions proposed in this paper to other initiatives for accessing remote laboratories online. We expect that these solutions will contribute to the reusability of remote experiments in different pedagogical frameworks.

3. Technology Overview

3.1 SCORM Technology and LLOs

Sharable Content Object Reference Model (SCORM) (Scorm, 2001) is a set of technical standards, specifications and guidelines designed to meet the functional requirements of the Advanced Distributed Learning (ADL) initiative. These standards are aimed at e-learning products like Learning Management Systems.

SCORM specifies that content should be packaged in a ZIP file and described in a XML file, named imsmanifest.xml (the "manifest file"). The XML file contains all the information the LMS needs to deliver the content, like information about how to launch each SCORM learning object (SCO), the description of the structure and resources of the package and, optionally, metadata that describes the course and its parts. "A SCO is a collection of one or more assets [electronic resources that can be rendered by a Web client] that represent a single launchable learning resource […]. A SCO represents the lowest level of granularity of a learning resource that is tracked by an LMS […]." (ADL, 2009). A SCO is the smallest runnable unit in the SCORM model.

Sharing remote experiments requires the maximal portability and reusability of the experiments, and it is the reason why SCORM standards are used as a framework for learning objects in this proposal.

In that context, experiments are available via LLOs ("Lab Learning Objects"). An LLO is a SCO with additional constraints extended by additional metadata. LLOs are learning objects that will include all the necessary elements to access a remote laboratory or an experiment, so they should obey the SCORM standard for learning objects and be packaged in ZIP containers similar to

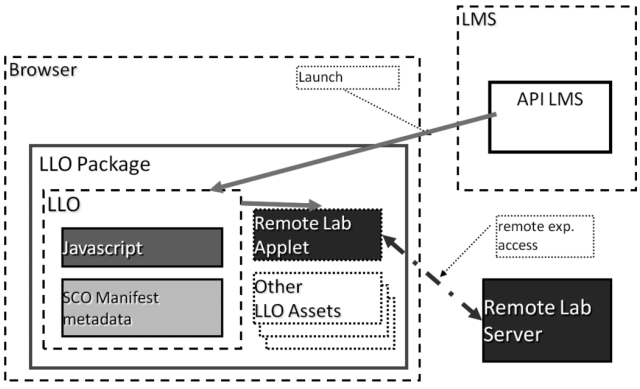


Figure 1: Lab Learning Object components

regular SCOs. Typically, an LLO will embed an applet in order to execute the remote laboratory or virtual experiment. Figure 1 shows the internal components of an LLO and the relations between an LLO, the LMS and a remote laboratory.

The LLO can be displayed using external learning management systems (LMS) where the LLO needs to be plugged in. Having LLOs based on SCORM makes it easier and possible to download the LLO from the remote lab server to deploy it in other SCORM compliant LMS.

3.2 Federated Identity Management

Federated Identity Management (FIDM) refers to a set of technologies, standards and use-cases which serve to enable the portability of identity information across different security domains. The main purpose is that users of one domain can securely access data or systems of another domain included in the federation, without needing redundant user administration. Typical use-cases involve cross-domain, single sign-on, cross-domain user attribute exchange, and so on.

Without using identity federation, remote users must present identities and authentication information for each of the services they want to access. Part of this information is the same in all of the services, and users have to repeat it several times. By contrast, in federated environments, a remote user must present the authentication and authorization information only once. This information gives users access to the resources hosted in the different servers of the federation.

FIDM allows users to reuse electronic identities, saves administrators redundant work in maintaining user accounts and provides a consistent, trustworthy infrastructure component.

3.2.1 Shibboleth

The Shibboleth System (Shibboleth, 2011) is a standards-based, open source software package for web single sign-on across or within organizational boundaries. It allows sites to make informed authorization decisions for individual access of protected online resources in a privacy-preserving manner.

Shibboleth allows the deployment of a federated identity management system, in which the components of the identity management system are distributed across organizations, with each organization trusting the other to perform the functions of the components they host. There are two roles in this kind of systems: organizations providing identities, which would like to provide applications only the information required to make authorization decisions, and organizations providing applications or services that would like to manage only the identity information required for their applications. In the second case, this characteristic reduces the risk of storing or accidentally releasing sensitive information they do not need.

The functional components of a Shibboleth implementation support these two roles. There is an identity provider (IdP) component, which has been implemented as a J2EE component, and a service provider (SP) which has been implemented as a C++ Apache module. There is also an optional "Where are you from?" (WAYF) service that redirects users to the right IdP.

In the context of this work, Shibboleth is proposed as the authentication and authorization technology for LMSs, due to the wide deployment of Shibboleth in educational organizations in Europe.

In the use case of LiLa (Section 6) Shibboleth is also used for authentication and authorization in the LiLa portal. This facilitates the creation of a federated identity management system comprising the organizations providing virtual laboratories and remote experiments, LMSs and LiLa as the organization that will provide a repository for experiments and laboratories. LiLa

enhances the use of remote experiments by including services based on the inclusion of metadata (search and catalog creation), and materials to facilitate the inclusion of the experiments in the curricula of different universities.

4. Architecture for Sharing Remote Laboratories

Developing a technical integrated and organization framework for the mutual exchange of experiments requires opening the access to the rigs for users in other educational institutions. Content providers from the institution owning a rig will provide access to the remote experiments using SCORM objects, and teachers from educational institutions will include these learning objects in the curriculum of their universities. Typically, only one user can access the rig at a time, so content providers will create reservations for accessing them.

4.1 Proposed Architecture

Figure 2 depicts the architecture proposed to control the access to shared laboratories and remote experiments in organizations belonging to a federation. There are four main entities: users or students who want to execute experiments, educational institutions – typically universities having LLOs to run remote experiments through their own LMS, remote laboratories which provide the experiments or virtual laboratories, and a booking system that manages the reservations of the experiments.

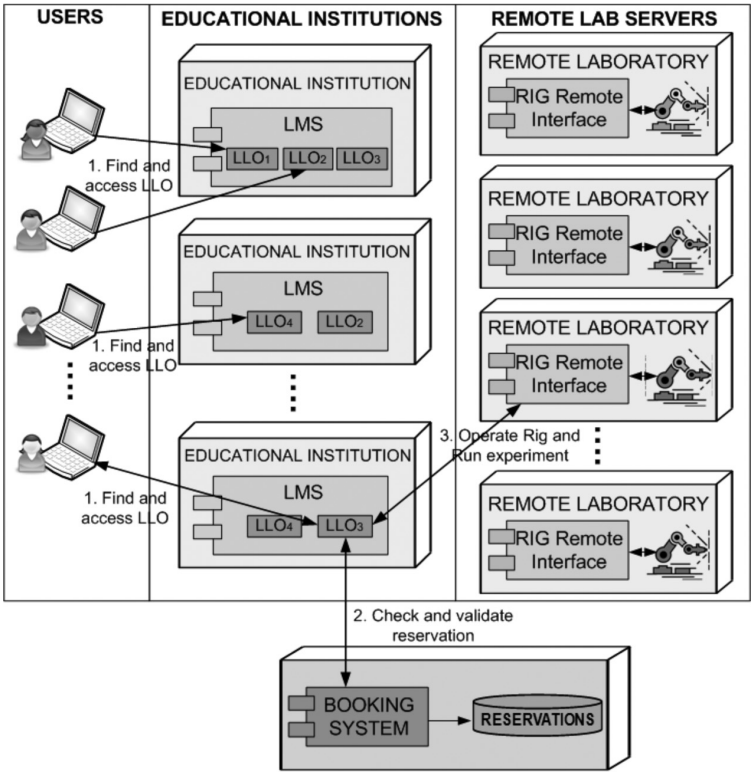


Figure 2: Architecture for controlling access to shared laboratories

A student, who wants to run an experiment belonging to a course of her university, goes to the web site of the corresponding LMS where remote experiments have previously been uploaded. Then, she can find and access the right experiment (available via an LLO), once she is authenticated by the system (1). The student's Web browser will download the LLO and contact the booking system to check access permissions (2). If the student has a valid reservation, she can run the experiment by operating the rig located in an external remote laboratory (3).

4.1.1 Client Side: Web Browser

Students run the remote experiments or virtual laboratories from within their JavaScript capable web browsers. When a student accesses an experiment which requires booking, the web browser will load the LLO content and execute the JavaScript code included in the LLO. This JavaScript will contact the corresponding Booking System, which will check if the student has a valid reservation for this experiment at this time. If the process is successful, the experiment will start by operating the remote rig. In order to do that, the LLO running in the browser will need to communicate with an API running in the remote server that interfaces with the rig. The definition of this API is beyond the scope of this research; typically, Flash or Java based interfaces implement this API, sometimes LabView plugins are employed as well.

4.1.2 Learning Management Systems

A Learning Management System can be described as a software application designed to deliver, track, report on and manage learning content, student progress and student interaction. The most popular LMSs are web-based, in order to facilitate access to learning content and administration. In the architecture proposed, the work is focused on the standards and technologies related to web-based LMSs, specifically the SCORM standard to create learning content that can be re-used by different LMSs. Virtual experiments and remote laboratories access will be contained in learning objects based on SCORM. It will be possible to use these objects in external LMSs, such as ILIAS (Ilias, 2011) or Moodle (Moodle, 2011).

In the architecture proposed, the educational institutions already have some kind of LMS; lecturers who use these LMSs will download experiment LLOs from Remote Labs, and will upload them into their SCORM compliant LMS. The LLOs will be deployed in such environments.

For students, this model provides the advantage of presenting a consistent view of the learning material, i.e. students never have to leave the LMS of the university to complete their homework.

However, linking the LLO to a booking system will be necessary for experiments that require a reservation. In this proposal, a URL for an external global booking system will typically be included into the LLO as metadata so the institutions providing access to their remote laboratories can use this global booking system for their experiments with almost no extra cost for them.

4.1.3 Remote Laboratories

Content providers give access to virtual experiments and remote laboratories. Typically, to run experiments in remote laboratories a remote rig needs to be operated. Rigs are the remote hardware that remote experiments operate on: the scarce resources, the expensive physical set-ups, the bulky and noisy machines, or the fragile robots. The rig is operated through a remote server that interfaces with the rig and provides an API that can be accessed by a remote client. A content provider will package all the necessary elements to access a remote laboratory in a zip

container similar to regular SCOs. This zip file is the LLO that will be provided to the institutions using the remote laboratory.

Several experiments could operate on the same rig. This is the reason why the booking system included in the architecture has been designed to administer rigs, instead of experiments, as it is explained below. A user books a rig, not an experiment. If the rig related to an experiment requires reservation, information about the rig must be included into the LLO, for example, in the SCO manifest metadata.

4.2 Booking System

If the experiments require booking, content providers need to create reservations for the rig controlling the remote laboratory using a booking system (BS). To create the reservation, they have to provide the identifier of the rig, the name of the rig, and time-slots within which students can access the rig. It is possible to specify many time-slots, e.g. "Monday, Wednesday and Friday from 10 am. to 2 pm". The URL of the BS needs to be included in the LLO as metadata. This URL will be used by the JavaScript code in the LLO to contact the BS to check and validate if a user has a valid reservation.

Before accessing an experiment that requires a reservation, a user must book the rig associated with the experiment by selecting an available time-slot in the calendar provided by the booking system. Then, the user will be able to access the experiment at the selected time and date.

One of the main advantages of the architecture is the usage of an external booking system. Neither the remote laboratories nor the LMSs must have their own booking system.

The booking system could be based on a set of calendars provided by external sites, such as Google Calendar, or a booking system such as the one provided by the LiLa portal, which has been designed based on the rig concept.

The Rig Booking System provides the functionality to schedule the access to the remote rigs with the aim to facilitate the student access to the experiments. It is implemented as a client-server solution. The server is a web application that runs on an application server. It implements a RESTful API that can be invoked from any external HTTP client (for example, a web browser running JavaScript). The server persists all reservation-related data in a database. The persisted data include information about the rigs, time-slots reserved and a codified unique user identifier. This booking system is built upon three different user roles: content providers, teachers and students, and five key elements: experiments, rigs, reservation for teachers, reservation for students and tickets.

The architecture and the conceptual model of the Rig Booking System are specified in (Mateos, Gallardo, Richter, Bellido, Debicki and Villagrá, 2011).

5. Rig Access Control

When providing access to remote experiments for users from within the institution owning the rig it is possible to provide scheduling and user permission checks through the institution portal or LMS. However, when LLOs are downloaded to an external LMS, implementing the access control for the experiment is not straightforward. This section discusses the authentication and authorization mechanisms proposed to overcome this problem.

Using a SSO system, such as Shibboleth, as an authentication and authorization mechanism in all the educational institutions and the remote laboratories would be a good solution, but

organizations providing access to LLOs through an LMS will many times want to use their own authentication mechanisms such as LDAP. In this case it is not convenient to assume that the authentication and authorization attributes could be provided exclusively through a SSO system such as Shibboleth, so the requirement is to include additional code within an LLO to control the access to the experiment from external systems. This code becomes an integral part of the experiment (it runs in the students' web browsers) and is responsible for checking against a RESTful (Richardson and Ruby, 2007) server that the user of an experiment can access the experiment at this time. If this is not the case, the code will render an informative message and will redirect the user to a webpage that will allow her to make a new reservation.

5.1 Learning Object Authentication and Authorization: Simple Validation

Figure 3 shows the scenario in which the LMS performs the user authentication and authorization, and the schedule control related to an LLO is carried out by the Booking System Server, using the user AA data obtained from the LMS.

When a user tries to access the LLO using her own LMS, she needs to sign in using the corresponding authentication mechanism of her institution. The LLO can get this security information using the SCORM API provided by the LMS.

Before launching the Remote Laboratory Applet, the LLO needs to check the user rights, i.e. if the user has a reservation for the experiment at the current time. In order to do that, the LLO will contact the BS Server hosted centrally or in the server of the institution (step 1 in the figure); in the request, the LLO sends the rig ID and the user ID obtained from the LMS "on the fly". The booking system checks if the user has a booking for running the requested experiment at this moment. In return, the BS provides a number equivalent to the user's rights. This number will be analyzed by the JavaScript code within the LLO. If the user has a valid booking to access the experiment, the virtual experiment or remote laboratory applet will start (step 2 and 3 in the picture). If this is not the case, the code will render an informative message and will redirect the user to a webpage (hosted in the booking system server) to allow her to make a new reservation, or to manage a previous one. The communication between the LLO and the REST interface of the BS will be based on AJAX, using the XMLHttpRequest object. The URL of the HTTP request is the booking system URL and it is included in the LLO.

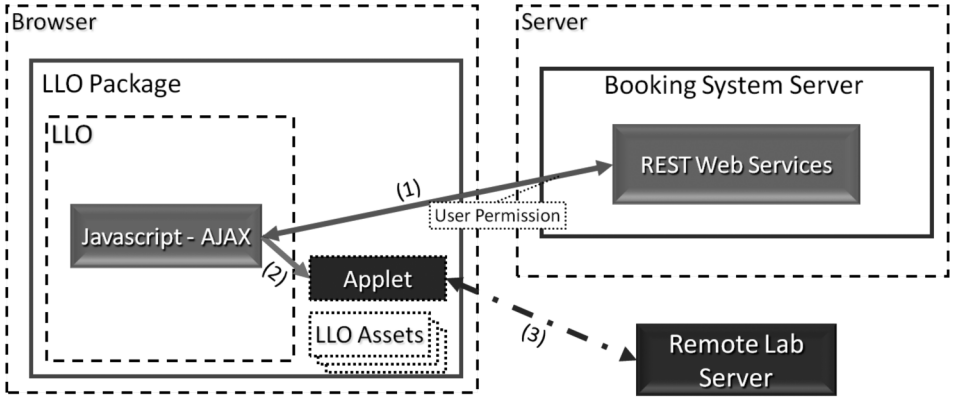


Figure 3: Authentication and Authorization for LLO using simple validation

This solution of implementing the users' access control using exclusively a JavaScript code that runs on the client side (the students' web browser) has an important security consequence: students with appropriate programming skills could locally edit the JavaScript code and overpass the access control. Thus, this solution is far from ideal from the security point of view.

But even with this security flaw, the advantage of this solution is that there is no need to add anything at the Remote laboratory server side or at the LMS server side. Typically, the administrators of the institutions do not want to change servers to access the LLOs in their institutions.

5.2 Learning Object Authentication and Authorization: Double Access

In this subsection, another authentication and authorization mechanism is proposed. This solution tries to overcome the security problem described above adding a higher security level to the process of controlling the external access to the experiments.

Figure 4 following depicts the scenario in which the LMS performs the user Authentication and Authorization, and the scheduling related to an LLO is the responsibility of the BS and also of the "Remote Lab Server".

The user Authentication and Authorization process is the same as that specified in the previous solution. The user is authenticated and then authorized to book the experiment and to access it from an external LMS by using the authentication system provided by the institution where the LMS is hosted.

As it is explained above, the LLO includes some information about the LLO, such as the rig ID. Also, the LLO can get the user ID by retrieving the value *cmi.core.student_id* using the API of the LMS. When the user tries to access the experiment, the LLO contacts the interface REST of the BS in order to get the user permission to access the experiment at this time (step 1). The BS provides a number as the result of the request and the JavaScript code analyzes this number and determines if the user can access the experiment or if she needs to make a reservation.

The main difference with the first mechanism is that in this case, the user ID, the identifier of the LLO and the user permission, will be passed as parameters to the applet when launched. These parameters might also include other authorization parameters depending on the level of

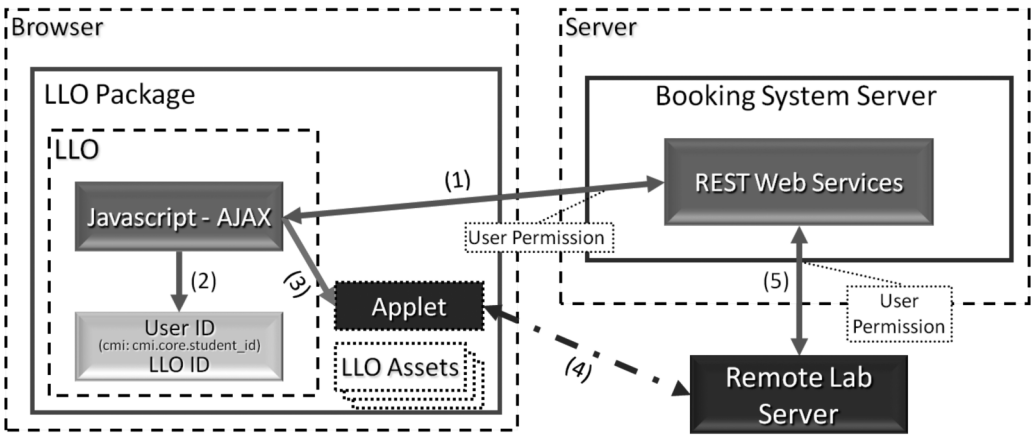


Figure 4: Authentication and Authorization for LLO using double validation

integration of the remote laboratory with scheduling mechanisms. Before executing the experiment, the remote laboratory server can then check against the REST interface of the BS server that the user has a valid reservation for the experiment at the current time, using the parameters of the applet. In order to do this, the remote lab server has to know the URL of the booking system. Implementing this communication between the Remote Lab Server and the REST BS server is a decision of the RLS administrator.

Adding this second check from the server side of the experiment avoids the problem of a user modifying the LLO code to get rid of the access control.

However, this solution requires including additional code in the remote server-side to get the parameters set by the LLO and to contact the BS server to check if the user has a valid reservation. The scheduling mechanisms of the booking system must be well-known and established in order to integrate the remote laboratory with the BS.

A good alternative to this double checking is to use a private key infrastructure technology such as a digital signature, which ensures integrity, authenticity and non-repudiation.

5.3 Learning Object Authentication and Authorization: Digital Signature

"*A digital signature is a mathematical scheme for demonstrating the authenticity of a digital message or document*" (Indrasena, Bhat, Chetwavani and Purushotham, 2011). That is, a digital signature authenticates the identity of the sender of a message and ensures that the content of the message has not been changed or altered in transit, since it was signed by the sender.

Typically, a digital signature is an encrypted version of a message, attached together with a message. A digital signature scheme consists of three steps:

- The sender creates the digital signature using a private key and a message by means of a signing algorithm. This private key has an associated public key; both were generated by a key-pair generation algorithm.
- The sender sends the message and the digital signature to a recipient.
- The recipient verifies the signature using the corresponding public key, which is found in the sender's X.509 certificate. The recipient accepts or rejects the message's claim to authenticity.

Using this asymmetric cryptography process as an authentication and authorization mechanism to control the access to the experiments from external users, prevents the Remote Lab Server from having to connect to the BS to check if the authorization information sent by the LLO as a parameter of the applet has been changed in transit.

Figure 5 illustrates how to use a digital signature solution to validate the user authentication and authorization data.

The schema of this proposed solution based on digital signature is the following:

1. The user tries to run one experiment (by accessing the LLO) using an external LMS, typically the LMS of her institution. She must be authenticated beforehand.
2. The LLO will contact the REST interface of the Booking System using the user ID and the LLO identifier (step 1).
3. The Booking System checks if the user has a valid reservation for this experiment, and generates a message, called *User Data* in the picture. This message contains information such as the *user ID*, the *LLO identifier*, *user permissions*, and *time slot* of the reservation.

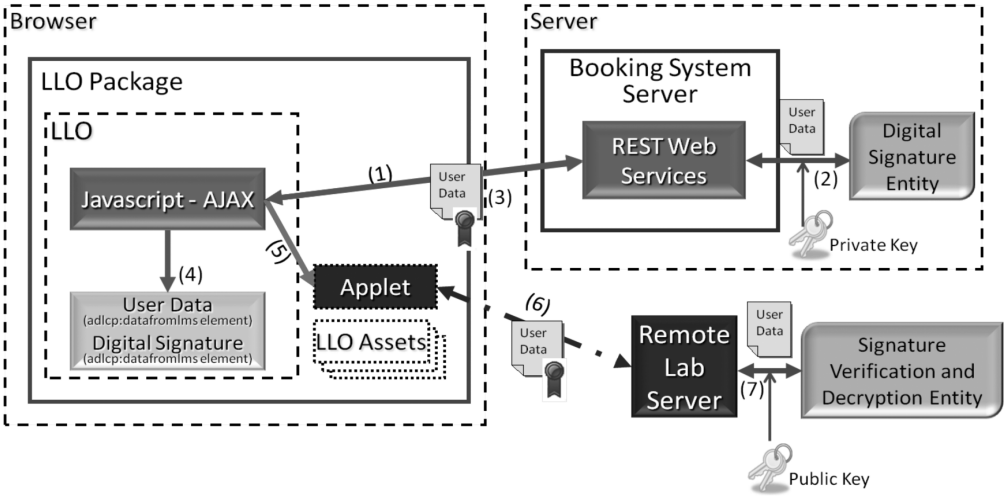


Figure 5: Authentication and Authorization for LLO based on digital signature

4. The message is encrypted by the sender (BS server) using its own private key. This encrypted message is called a Digital Signature (step 2).
5. In return for the request of the LLO, the BS will provide the digital signature attached to the original message (*User Data*). Also, the user rights are provided (step 3).
6. The JavaScript code analyzes the permission of the user, and determines if the user can access the experiment or if she needs to make a reservation. If the user can access the experiment, the original message and the digital signature will be passed as parameters onto the applet when launched (steps 4, 5 and 6).
7. Before executing the experiment, the remote laboratory server (RLS) must verify the signature, using the BS public key, which is found in the sender's X.509 certificate. If the BS is a trusted entity, the RLS decrypts the signature with the key and gets the message (step 7).
8. The RLS compares the message decrypted with the original message. If they are exactly equal, the recipient can be confident that the message has come from the BS and has not changed since he signed it. In that situation, the experiment will start. If the messages are not equal, the signature is not valid for that message, so the user cannot execute the experiment.

The communication between the LLO and the REST interface of the BS will be based on AJAX, using the XMLHttpRequest object. The URL of the HTTP request is the booking system URL, and it is included in the LLO, as in the previous solutions.

Using digital signatures, a higher security level is added to the control access, because the BS signs the user data, and the RLS verifies the authenticity of the message by using the corresponding public key. Therefore, there is no need to include additional code in the remote server-side to contact the BS server.

5.4 Learning Object Authentication and Authorization: Using Shibboleth

Accessing remote laboratories through the web can also take advantage of the solutions proposed for federated identity management systems, using Shibboleth. In this case, the federated identity

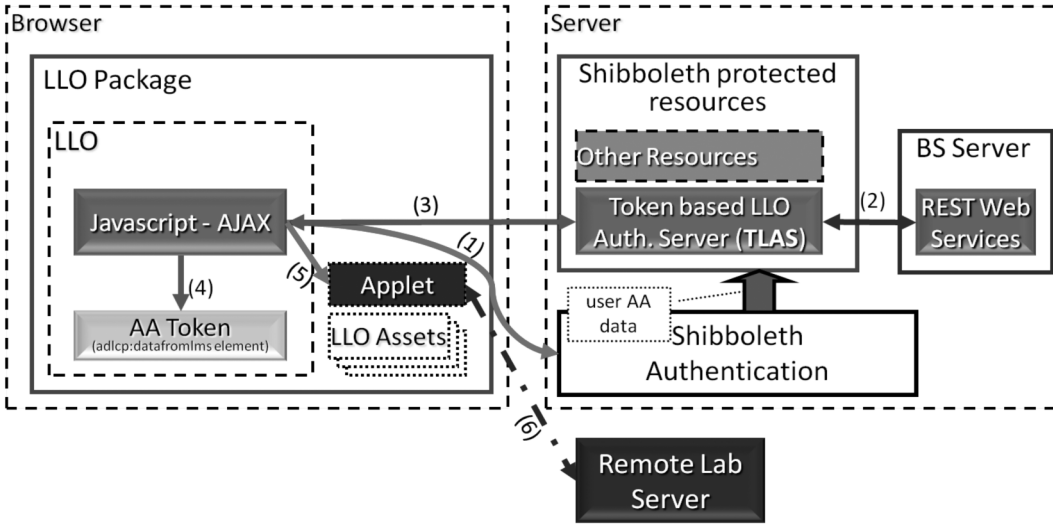


Figure 6: Authentication and Authorization for Lab Learning Objects based on LLO Auth. Server Token

management system would consist of organizations providing virtual laboratories and remote experiments, organizations providing access to those experiments through their LMSs and a common booking system.

Figure 6 depicts the scenario in which Shibboleth is used to perform User Authentication and Authorization, while the authorization related to an LLO is carried out by a specific "Token based LLO Authorization Server" (TLAS).

An LLO includes some information or a security token, the LLO AA Token, for example in the SCO Manifest *adlcp:datafromlms* data element. An LLO is designed to get this information before launching the Remote Laboratory Applet, by retrieving the value *cmi.launch_data*.

Once the LLO AA Token is retrieved, the LLO will contact the TLAS using the token and in return, the TLAS will provide some piece of information which is needed to start the virtual experiment / remote laboratory applet. This piece of information will be passed as a parameter to the applet when launched. In the case of a remote laboratory applet, the parameter could be the URL needed to contact the remote laboratory server, which might include authorization parameters, depending on the level of integration of the remote laboratory with the booking system authorization and scheduling mechanisms. In the case of a virtual experiment, in which the applet does not need access to a server, the parameter could be some access code needed to unlock the virtual experiment. The communication between the LLO and the TLAS will be based on AJAX, using the XMLHttpRequest object.

The task of the token-based LLO Authorization Server (TLAS) will be aided by Shibboleth. This server will be accessed through an Apache httpd server and protected by a Shibboleth Service Provider. This means that the TLAS will be able to perform the LLO authorization task using the user AA data provided by Shibboleth. This will make it possible to have an authorization schema where an LLO is only authorized if the user belongs to a certain organization that has obtained the access rights to the LLO. In this schema, the AA token includes information that links the LLO and the organization.

In the case of an experiment for which previous reservations are required, the TLAS should also be linked to the booking system infrastructure, in order to only give access to the applet in case there is a valid reservation for the user.

Other alternatives have been considered. For example, AA attributes and scheduling information could be provided exclusively through Shibboleth. However, this would mean that in order to take advantage of remote laboratory reservations, an external LMS would need to include a Shibboleth Service Provider.

Another alternative is to include the functions to check AA and scheduling status in the *LMSInitialize()* function that is provided by SCORM compliant LMSs systems. This could be an interesting alternative, since some of the most popular LMS are open source so the LLO AA and scheduling mechanisms could be integrated in LMSs such as ILIAS or Moodle.

6. Use Case: LiLa

Library of Labs (LiLa, 2011) is an initiative of different European universities and companies for granting access to shared virtual laboratories and remote experiments. LiLa aims to enhance the learning of the natural sciences and engineering. Some university courses have developed simulations, known as 'virtual laboratories', to help make abstract concepts more graphic and concrete. Moreover, remote hardware is being made accessible online as 'remote laboratories' in order to provide practical experience in a more flexible way. LiLa aims to make as many of these virtual and remote laboratories centrally accessible as possible, and provide a secure access control mechanism when necessary.

The architecture of LiLa is similar to the architecture proposed above, but including a new entity, called the LiLa Portal. The LiLa Portal is a repository where the content providers from the Remote Labs in the LiLa community can upload their virtual and remote experiments as LLOs. The bulk of the experiments currently in the portal belong to the subject area of Physics, but other fields are represented as well. The main content providers in LiLa are the following (Boehringer and Robbe, 2011):

- The Technical University Berlin provides their 'RemoteFarm'; an online laboratory of remotely controlled physics experiments used in undergraduate education.
- The University of Cambridge provides a reconfigurable chemical reactor 'Weblab' which can be used for teaching several subjects on the Chemical Engineering curriculum.
- The Aristotle University of Thessaloniki contributes their Nanotechnology Remote Lab (NRL).
- Other content providers have developed virtual experiments or simulations which give access to experiments which cannot easily be performed in reality. Among them are the Universities of Stuttgart and Basel, the Cambridge-based company CMCL innovations, and the University of Linköping with its OpenModelica simulation language.

If the online experiment is a "remote experiment", like the Cambridge Chemical Reactor Weblab, all software that is needed on the user side to control the remote equipment is wrapped into an LLO, as it is explained above. In some instances, this would mean that the user would be given access and control over a 'virtual machine' via a "remote desktop" from where the experiment is controlled. This variation had to be introduced for security reasons. Remote experiments also require prior booking. To book and control the access to the experiment, LiLa uses two of the four authentication mechanisms proposed above: simple validation and double access. The solution based on digital signature and the solution using Shibboleth are a work in progress. In the next

subsection, the experiment 'Chemical Reactor WebLab' is used as an example to show how access control and booking works in practice.

6.1 Specific Use Case: Controlling Access to 'Chemical Reactor WebLab'

Cambridge WebLabs (Coble, Smalbbone, Bhave, Watson, Braumann and Kraft, 2010) is a 'Cambridge-MIT Institute' initiative for developing remote experiments for the teaching of Chemical Engineering. At the present time, two Weblabs experiments have been developed: one on chemical reaction engineering and one on process control. The first one is called 'Chemical Reactor Weblab' and it consists of a continuous Stirred Tank Reactor controlled by Industrial Standard PCS7. The experiment can be configured for operation in non-ideal mode or well mixed for process control.

This remote lab has been uploaded to the LiLa portal as an LLO, and it is available for running remotely. This LLO is available as content packages in two ways:

- Teachers can download the LLO and reuse them in their own LMSs.
- Students can access and execute the LLO using the portal or through the LMS of their universities. Also, they can search and comment on the experiments using the portal. In order to run LLOs through the portal, LiLa implements a SCORM RTS, allowing the communication between experiments and LiLa.

When uploading this LLO to LiLa, the content provider has to specify the rig for the experiment that students need to book. This experiment requires booking, so the content provider must select the URL of the corresponding Booking System, and create reservations by defining time slots in which the experiment will be available. LiLa portal provides a calendar interface that allows content providers and teachers to create reservations.

Once the reservations for users are created, the students can access experiments. To control the access to the remote experiment, LiLa uses two of the four authentication and authorization mechanism proposed above: simple validation and double access. When a content provider is wrapping the experiment into an LLO, she has to choose one of the authentication and authorization mechanisms for each experiment. If simple validation is used, the content provider does not need to include any additional code; the portal will embed the necessary JavaScript code into the LLO. In the case of double access, the portal will also embed the JavaScript code into the LLO, but content providers will have to include some modifications:

- They need to add JavaScript code such as the following when building the LLO:

```
var lilaAuthUID = lila_checkBooking.getLilaUser();
var lilaAuthRID = lila_checkBooking.getLilaRig();
```

  Using these two variables *lilaAuthUID* and *lilaAuthRID* the content provider will also add the needed code to pass those two values to the Remote Lab server to check if the user has permissions to access the experiment.

- On the remote lab server-side, the content provider has to contact the RESTful interface of the BS accessing a URL like the following:

```
https://BS_URL/rigaccess/<lilaAuthUID>/<lilaAuthRID>
```

  The booking system will check if the user (<*lilaAuthUID*>) has a valid reservation for the requested rig (<*lilaAuthRID*>). In return, it will provide "true" or "false" as a response.

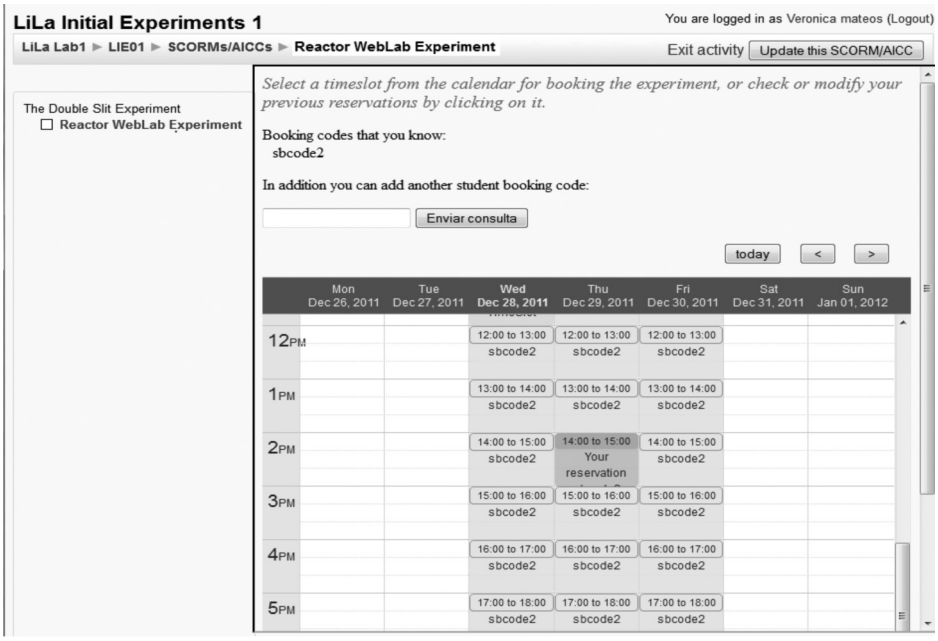


Figure 7: Calendar interface for making a reservation when a student tries to access an experiment through the LMS without a valid booking

In the use case of the chemical reactor weblab, the authentication mechanism used is double access.

Regardless of the authentication and authorization mechanism used for access control, the student has to book a time slot before running the experiment. If she tries to access the experiment without a reservation, the user will be redirected to a calendar in which she can make a reservation. Figure 7 shows a view of a calendar in which the user has selected an available time slot on Thursday from 14:00 to 15:00. Then, if she tries to run the experiment at that time, the booking system will return "true" and the experiment will start correctly.

6.2 Evaluation Results

Three questionnaires based on the roles that a user can have in the LiLa portal (Student, Teacher and Content Provider) were prepared, tested and revised. In this evaluation, content providers, teachers and students were asked about the usefulness of using virtual or remote experiments for learning, their satisfaction with the portal functionality, and the usefulness of the access control mechanism provided. Most of the results were positive; some of them are shown in Figure 8 (Boehringer and Robbe, 2011):

For example, one of the most important questions for the evaluation is the acceptance and perception of the online experiments by students. The students' answers were unequivocal: they enjoy running online experiments and many say that their interest and motivation increased by using the online experiment. This result is especially motivating for the teachers since experience has shown that the subject matters are difficult for the students to understand, and the increase in motivation is expected to correlate with learning success. Another issue is the access control for

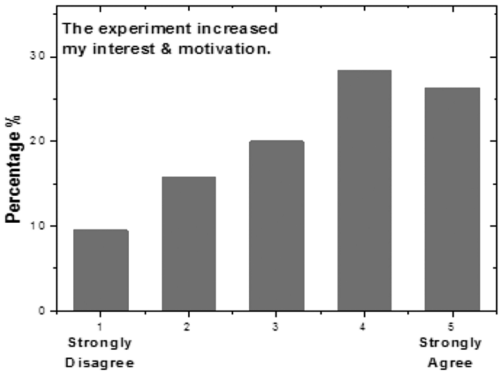


Figure 8: Evaluation results regarding to using virtual and remote experiments for learning

the remote experiments. Some of the remote experiments used by the students involved the use of a booking system and access control. The analysis of the student questionnaire showed that more than 70% of the students who used the access control declared that it was easy to make a reservation via the LiLa Portal booking system.

7. Discussion and Conclusions

Controlling the access to remote laboratories using a booking system is a need that many projects have addressed in different ways. A solution with very strong support for access booking and distributed and federated user account management is iLabs (MIT, 2011). However iLabs does not have good support for integrating with third-party learning management systems and though it is open source itself, it requires Windows Server machines and it uses the Microsoft SQL Server and Visual Studio.NET.

Other projects base their solutions on extensions to the functionality of a particular LMS so they cannot be reused independently from it. For example, the MARVEL project (Ferreira and Cardoso, 2005) and its successor EDIPE (Uran, Hercog and Jezerni, 2007) include a booking system for Moodle (Moodle, 2011); WebLab (Cirka, Kvasnica and Fikar, 2008) of the Slovak University of Technology in Bratislava is also module for Moodle; and ReLEEP, developed by the Qatar University (Khalil, Hasna, Benammar, Chaabane and Ben amar, 2009) also uses a Moodle specific solution extending the "Meeting Room Booking System" (MRBS, 2011) to accommodate their needs.

The proposal presented on this paper provides a solution that is not particular to any LMS, since it is based on an external booking system that can be viewed in an integrated manner in LMSs supporting the SCORM standard. The solution also addresses the separation of three roles in a distributed manner: content providers giving access to their laboratories, teachers providing access to those external laboratories through their own institution LMS and students accessing the remote laboratories. The solution has been designed with the reusability and flexibility as the main goals, basing the implementations on open source software. Some of the solutions discussed for remote laboratories access control have already been implemented and are being used in the LiLa Portal. The alternative solutions presented, i.e. using digital signatures and using Shibboleth, are currently a work in progress in the context of the LiLa portal.

Acknowledgements

This work was partly funded by the eContentPlus EU project ECP-2008- EDU-428037 'LiLa' (Library of Labs: Dissemination of Remote and Virtual Laboratories for Natural Sciences and Engineering).

References

ADL. (2009): Sharable content object reference model ® 2004 4th Edition Run-Time Environment Version 1.0.

BISSELL, AHRASH N. (2009): Permission granted: Open licensing for educational resources, *Open Learning: The Journal of Open, Distance and e-Learning*, 24(1): 97–106.

BOEHRINGER, D. and ROBBE, T. (2011): Project LiLa: Final report. http://www.lila-project.org/resources/Documents/files/D1-8_annual-public-report_final.pdf Accessed 12-Apr-2012.

CIRKA, L., KVASNICA, M. and FIKAR, M. (2008): WebLab module for the Moodle Learning Management System. *Proceedings of the 9th International Conference Virtual University 2008*, E-academia Slovaca.

COBLE, A., SMALLBONE, A., BHAVE, A., WATSON, R., BRAUMANN, Q. and KRAFT, M. (2010): Delivering authentic experiences for engineering students and professionals through e-labs. *Education Engineering (EDUCON)*, 1085–1090.

DOWNES, S. (2007): Models for sustainable open educational resources. *Interdisciplinary Journal of Knowledge and Learning Objects*, 3: 29–44.

FERREIRA, J. and CARDOSO, A. (2005): A Moodle extension to book online labs. *Int. J. Online Engineering*, 1(2).

HILTON III, J., WILEY, D., STEIN, J. and JOHNSON, A. (2010): The four 'R's of openness and ALMS analysis: Frameworks for open educational resources. *Open Learning*, February, 25(1): 37–44

ILIAS. (2011): ILIAS (Integriertes Lern-,Informations- und Arbeitskooperations-System) Learning Management. http://www.ilias.de/docu/. Accessed 12-Apr-2012.

INDRASENA, M., BHAT, P.J., CHETWAVANI, R. and PURUSHOTHAM, M. (2011): Establishment of public key infrastructure for digital signatures. *Computer Engineering and Intelligent Systems*, 2(6).

KHALIL, A., HASNA, M., BENAMMAR, M., CHAABANE, M. and BEN AMAR, C. (2009): Development of a remote lab for electrical engineering program. *Signals, Circuits and Systems (SCS), 2009 3rd International Conference on*, 1–5.

LILA. (2011): LiLa – Library of Labs. http://www.lila-project.org. Accessed 12-Apr-2012.

MATEOS, V., GALLARDO, A., RICHTER, T., BELLIDO, L., DEBICKI, P. and VILLAGRÁ, V.A. (2011): LiLa Booking System: Architecture and Conceptual Model of a Rig Booking System for On-Line Laboratories. *Int. J. Online Engineering*, 7(4).

MIT (2011): MIT iCampus: iLabs. http://icampus.mit.edu/iLabs. Accessed 12-Apr-2012.

MOODLE (2011): Moodle.org: Open-source community-based tools for learning. http://moodle.org. Accessed 27-Dec-2011.

MRBS (2011): MRBS: Introduction. http://mrbs.sourceforge.net. Accessed 12-Apr-2012.

RICHARDSON, L. and RUBY, S. (2007): RESTful Web Services. *O'Reilly Media, Inc*.

SCORM. (2001): Advanced distributed learning initiative, sharable content object reference model (SCORM) Version 1.2. The SCORM content aggregation model. 2001. http://www.adlnet.org. Accessed 12-Apr-2012.

SHIBBOLETH. (2011): A project of the Internet2 Middleware initiative. http://shibboleth.internet2.edu. Accessed 12-Apr-2012.

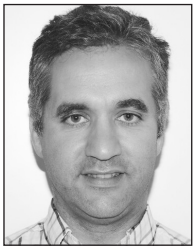
URAN, S., HERCOG, D. and JEZERNIK, K. (2007): Remote control laboratory with Moodle booking system. *Industrial Electronics, 2007. ISIE 2007. IEEE International Symposium on*, 2978–2983.

WATSON, R., COBLE, A., BHAVE, A., SMALLBONE, A. and KRAFT, M. (2012): Collaborative sustainability strategies for online laboratories. *Internet Accessible Remote Laboratories: Scalable E-Learning Tools for Engineering and Science Disciplines*, 23: 468-490, IGI Global.

Biographical Notes


Verónica Mateos

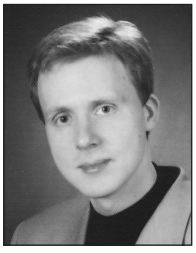
Verónica Mateos *has been a PhD student at the Department of Telematics Systems Engineering at Technical University of Madrid (UPM) since 2008. She received the MS degree in telecommunications engineering from the Technical University of Madrid (UPM) in 2008. Her research interests are in the area of security networks, and ontologies and semantic web. She has also participated on several national and international research projects.*


Luis Bellido

Luis Bellido *is associate professor of Telematic Systems Engineering at the Technical University of Madrid. He received his PhD degree in telecommunication engineering from the same university. For the last years he has been involved in research and development activities related to quality of experience evaluation, semantic web, multilingual web and learning support technologies. He has undertaken research projects in these fields and acquired an extensive research, development and technical management experience through his involvement in a number of European Commission funded projects in collaboration with research and industrial organizations.*


Víctor A. Villagrá

Víctor A. Villagrá *has been an associate professor in telematics engineering at the Technical University of Madrid (UPM) since 1992. He has been involved in international research projects related with network management, advanced services design and network security, as well as different national projects. He is author or co-author of more than 70 scientific papers and is author of a textbook about security in telecommunication networks.*


Thomas Richter

Thomas Richter *received his Master of Science in physics and mathematics at the University of Technology, Berlin, and received his PhD in mathematical physics on the Quantum Hall Effect at the University of Technology, Berlin, at the SFB 288 – Special Research Forum on Quantum Mechanics and Mathematical Physics. In the following two years, he worked for Algovision Technology GmbH in Berlin as project manager in the field of image compression/JPEG 2000. From 2002 to 2004, he moved back to the Berlin Mathematical Research Center Matheon to develop Virtual Laboratories for research and education purposes. In 2007, he moved to the Computing Center of the Unviersity of Stuttgart to continue his research interest in remote and virtual experiments. Thomas Richter has published numerous papers on virtual experiments and compression technology.*


Alberto Gallardo

Alberto Gallardo *is a PhD student at the Department of New Media in Research and Teaching (NFL) at the University of Stuttgart Computing Centre. He received the MSc degree in computer science from the Technical University of Madrid (UPM), in 2001. He has worked for the air traffic-control industry developing mission-critical software from 2000 until 2008, and since then for the e-learning community, developing web content management solutions. His research interests are in the area of software architecture, software engineering, and image processing.*