*Journal of Universal Computer Science, vol. 10, no. 2 (2004), 145-157*
*submitted: 1/8/03, accepted: 11/2/04, appeared: 28/2/04 © J.UCS*

# Securing Web-Based Exams

**Olivier Sessink**
(Wageningen University, the Netherlands
olivier.sessink@wur.nl)

**Rik Beeftink**
(Wageningen University, the Netherlands
rik.beeftink@wur.nl)

**Johannes Tramper**
(Wageningen University, the Netherlands
hans.tramper@wur.nl)

**Rob Hartog**
(Wageningen University, the Netherlands
rob.hartog@wur.nl)

**Abstract:** Learning management systems may offer web-based exam facilities. Such facilities entail a higher risk to exams fraud than traditional paper-based exams. The article discusses security issues with web-based exams, and proposes precautionary measures to reduce the risks. A security model is presented that distinguishes supervision support, software restrictions, and network restrictions. Solutions to security problems are tools to supervise and monitor web-based exams, measures for exam computers with Windows and Linux, and secure network setups in common network architectures. The article intends to raise risk awareness among faculty in higher education, and to help technical staff to implement precautions.

**Key Words:** web-based exam, assessment, security, supervision, fraud prevention
**Categories:** K.3.0, K.3.1

## 1 Introduction

Since a few years, almost every institute in higher education deploys one or more learning management systems (LMSs) as a facility for students and staff. Many of these systems use Internet for communication and they often have a web-interface. This means essentially that the system can be accessed using a web browser. A number of LMSs feature a test and exam facility: the Blackboard learning management system, for example, has its *assessment* facility [Blackboard, 2002].

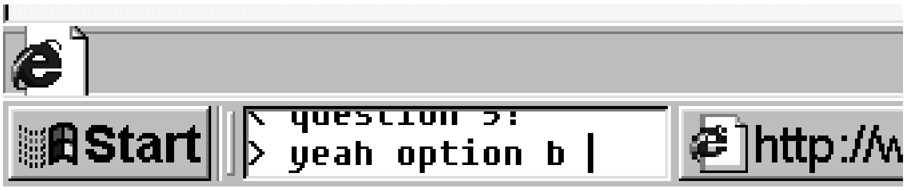
In this article the term web-based exam refers to a situation in which a student accesses questions and submits answers by a web browser and in which the exam results (partially or completely) determine the final grade for the subject. When the test is used for final grading it is important to assure that the student took the test in a satisfactory setting. Each exam may require a different setting. Commonly, the student should not have help from other people; access to answers from other students is not allowed either. Often, the student should not have access to the Internet (apart from access to the LMS itself), to a book, or to personal notes. Sometimes, however, students are allowed access to a book ('open-book exams') or even to the web ('open-web exams'). With the shift in learning goals towards comprehension, application, analysis, synthesis, and evaluation, more and more exams become open-book.

Most of these requirements may be met, to some extent, by traditional supervision; faculty members, however, should be aware that students have many more possibilities for fraud in a computer-room than in a traditional classroom. The aim of this article therefore is twofold: The first aim is to raise awareness of faculty in higher education of the possibilities that students have with web-based exams (sections 2 and 6). The second aim is to help technical staff with several solutions to support digital supervision (section 3) and to secure the computer facilities (section 4 and 5).

## 2 Security issues

For faculty members, it is important to realize that the new generations of students have a high level of computer skills. Students may exchange or acquire answers to exam questions in ways that most faculty members are not aware of. We estimate, for example, that some 80% of all students at Wageningen University use instant messengers like ICQ, Trillian, AOL, MSN, etcetera, on a regular basis, while the number of faculty users is negligible.

By its nature, a web-based exam may provoke unwanted communication. Since the exam is web-based, the exam computer should have a network connection. A student might misuse this connection to communicate with other students.

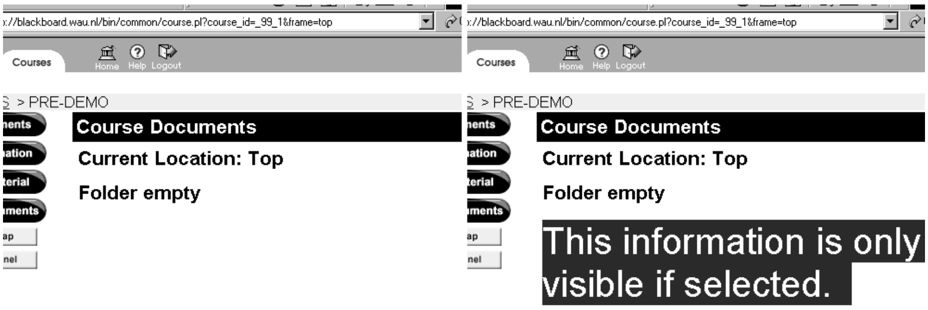


question 3:
> yeah option b |

*Figure 1: A chat program may go unnoticed.*

Chat programs and instant messaging programs provide network communication. Users may customize the look of some of these programs, which is called *skinning.* Skinning may be used to conceal these programs on the computer desktop, or to disguise them. In the example screen-shot [Fig. 1], a chat program is hidden in the windows task-bar. Such a program easily remains unnoticed by the supervisor in the computer room. ICQ, Trillian, MSN, and mIRC, for example, are freely available on the Internet.

It even is not necessary to install such communication software on the exam computer itself, because many public web sites offer chat facilities. Also, students can install these facilities on their personal home pages. Even the chat facility from the LMS itself might be used to communicate during an exam.


://blackboard.wau.nl/bin/common/course.pl?course_id=_99_1&frame=top
Courses
Home
Help
Logout
> PRE-DEMO
nents
ation
terial
ments
ap
nel

Course Documents

Current Location: Top

Folder empty

This information is only visible if selected.

*Figure 2: Screenshots of a Blackboard page. The normal view on the left contains a blank area. On the right, the apparently blank area is selected and hidden text is revealed.*

Besides chat facilities there are many more options to exchange answers. Almost any communication program may be used to exchange answers. Students can upload answers to exam questions to their homepage with an ftp program, making the answers available to their fellow students. The telnet program might be used by students to login to a common account on a Unix server. An email program might be used to send answers to another student. Some LMSs even enable users to setup their own page. This page then may be used to add a chat facility to the LMS, or to make answers available to other students. A skilled student could even make a web site that resembles the look of the exam facility, but actually contains the answers to the exam. An example is shown in [Fig. 2]: on the left, a page looks empty; on the right, previously invisible text is selected and thus appears. This is just a short list of possibilities, many more can be found.

Other important issues are the identity and the whereabouts of the student who submits the exam. Students may, for example, exchange passwords and submit each other's exams. Students may also leave the exam room prematurely, and log on to the LMS from an external location and continue the exam with help from others. Moreover, a student may also pretend to take the exam, while another student from an external location is actually submitting the exam.

On the Internet, a computer's Internet Protocol (IP) address is often used to restrict access to a service (for example access to a fileserver), or to register from which computer a service has been accessed. On many networks, however, it is easy to change the IP address of a computer and to adopt the address of another computer (known as *IP spoofing*). Using IP spoofing, a student at an external computer could pretend to submit his exam from a computer within the exam room. Such a disguise is especially easy on wireless networks, because physical access to the network (e.g. UTP cable) is not required.

A number of features in LMSs are meant to guarantee the identity of the student. LMSs may limit exam access to a certain time period. Multiple log-ins by a single user may be prohibited as well. Some LMSs may accept client log-ins from a specific IP address range only. As stated before, such measures are vulnerable to IP spoofing. Some systems offer password protection of the exam. The password can be made available just prior to the start of the exam. This can stop students to log in into the exam from external locations.

Several security issues are not specific to web-based exams and therefore beyond the scope of the present contribution. An important example is the security of the server. The server stores all questions, all answers, and after the exam all grades. Another example concerns the computer room setup; it is often not particularly suited for exams. Many computer rooms are setup for tutorials, and all displays face the same direction. Rows of screens are usually within short distance, thus allowing students to glance at other participant's screens. Because web-based exams usually consist of multiple-choice questions this increases the risks for fraud.

## 3 Digital supervision

As indicated above, most security issues involve electronic communication. Below, the prevention of unwanted electronic communication will be discussed. Faculty members should, however, anticipate that some students, while accepting the traditional need for supervision during exams, will claim that being “watched electronically” will influence their performance.

### 3.1 Supervising the client computer

There are several possibilities to detect which software is being used by a student. The supervisor may request a process listing and check all programs a user is running. Also, there are several software suites available to view the contents of a user’s display without disturbing the user. Visual Network Computing (VNC) [Richardson et al. 1998] and PCAnywhere [Symantec 2002], for example, may be used to remotely monitor any user’s screen. If students are aware that the supervisor may monitor their displays, this awareness will probably prevent them from using the computer for unwanted communication.

### 3.2 Supervising the network

In addition to supervision of the client software on the student’s computer, also the network traffic between computers may be monitored. To this end, a packet analyzer may be used. For open-web exams in particular, such network analysis will make sense. Students are allowed to use the web to find information, but not for mutual communication. A packet analyzer analyzes and logs all network traffic and thus may be used for prevention and to identify student communication. If students are aware of such personal traffic logging, they will most likely refrain from chatting and network communication. In the case that a student is suspected of cheating, furthermore, the network logs may be used to ascertain that the student really did.

Most networks in PC rooms use Ethernet (IEEE 802.3). The exam computers can be on the same Ethernet segment (shared), on several shared segments (bridged), or all on separate segments (switched). On a shared network, all network traffic is visible to all hosts in the segment. In such a case, the network analyzer should be connected to the same segment as the client computer. On a switched network, it often is not feasible to apply a packet analyzer without some additional effort. In such a case, the network traffic is visible only to the source and to the destination host. Either the switch should be configured to send all traffic to the analyzer, or the analyzer should be connected upstream from the switch. Most switches at Wageningen University, unfortunately, do not provide such configuration options. In addition it is often impossible to connect a packet analyzer to the upstream network hardware (e.g. fiber backbones). In many cases the network is neither switched nor shared but bridged; some hosts might see other traffic while others might not. A detailed analysis of the network topology is then required to decide whether a packet analyzer is feasible or not.

A powerful tool for packet analysis is offered by Ethereal, an open-source application that logs and analyses network traffic [Sharpe, Warnicke 2002]. As an alternative, it may analyze traffic logs that are provided by the TCPdump utility. Any Linux computer running Ethereal can be used to log the network traffic. If TCPdump is used as an intermediate, even non-Linux computers can be used once they have been booted from diskette or CDROM. Logs can then later be analyzed by Ethereal. This setup may be expanded to a professional setup in which dedicated remote controlled network loggers send their logs to a central facility for analysis. Such logs are very large, however, and in our experience analysis is time consuming.

If the exam computers use a proxy server, the data in the proxy log may be used for analysis. If students browse the web during an open-web exam, the proxy logs can be used to analyze the web sites that are visited; students who use chat facilities at some web site can easily be identified. Proxy logs are plain text files and small in size when compared with network logs; less effort is thus required for analysis. If traffic analysis relies on proxy logs, it should, obviously, be impossible to circumvent the proxy server. This can be ascertained by securing the client computer (section 4.2 and 4.3) or by blocking other network traffic (section 4.5 and 4.6).

## 4 A four level security model

To address the issues in electronic communication, a four level security model is used to classify security measures for exam computers. The solutions involve the configuration of the exam computer, both for Windows systems and Unix systems, and the configuration of the upstream network.

| *Layer* | *Description* |
|---|---|
| I User interface restrictions | Configuration of exam computer restricting parts of user interface |
| II File system restrictions | Configuration of exam computer restricting availability of or access to executables |
| III Local network restrictions | Configuration of exam computer restricting network access |
| IV Upstream network restrictions | Configuration of upstream network hardware or servers restricting the exam computer’s network access |

*Table 1: A four level security model*

The levels in table 1 can be compared to the ISO OSI reference model [Siyan et al. 1997]. Levels I and II secure the exam computer in layer 7 of the OSI model. Level III secures the exam computer in layer 3 of the OSI model. Level IV secures network hardware or servers in layer 3 trough 7 of the OSI model.

| *Layer* | *Description* |
|---|---|
| 7 – Application | Application specific services |
| 6 – Presentation | Converting the information |
| 5 – Session | Session services and activity management |
| 4 – Transport | End to end communication control |
| 3 – Network | Routing and switching the information in the network |
| 2 – Data link | Error control and grouping of data |
| 1 – Physical | Transmission over the physical medium |

*Table 2: The ISO OSI reference model*

Unix systems are quite secure by default, and offer a wide range of tools to restrict the user. Linux, being the most popular Unix-like operating system, will be used as an example. Not everybody will require the extensive security features of a Unix system; therefore measures for both Windows and Linux will be described.

The first three levels concern configuration of the exam computer's operating system. Consequently, these solutions are as safe as the operating system is. On a default Linux system, students will not be able to change the configuration without root access. On a Windows 98 system, however, a skilled student can easily change the configuration. The exam computer, furthermore, should only boot from its hard disk drive and it should have a BIOS password set to assure that the configuration cannot be changed. If this measure is omitted any user can boot the exam computer from a diskette or CDROM and obtain full access to its hard disk drive. The user could, for example, install a chat program or remove network restrictions.

### 4.1 Superfluous software

In many exam settings, only a web browser and some specific software should be used during the exam. The use of *'superfluous software'* to communicate with other students, or to help answering exam questions should be prevented. Preventive measures to this end are presented in levels I and II.

### 4.2 Level I – User interface restrictions

Windows has its security architecture built around so-called policies. These policies are stored on a server, and are retrieved upon a successful domain logon. Policies may be used to disable the registry editor, to restrict execution of applications and to disable the control panel. Many of these policies are, unfortunately, easy to circumvent. To restrict execution of applications, for example, the system administrator needs to specify full path names of executables. If, however, a user copies the executable to another location and renames it, the executable is still executable. The option to disable registry edit tools, furthermore, only disables the Microsoft registry edit tools. Many third-party registry tools are still functional.

Windows 2000, in combination with a Windows 2000 domain, features so-called group policies that allow tighter control over the desktop environment. An interesting feature is the “local security policy”. Using a template, the exam computer can be secured against software installation, or against a change of configuration. Again, however, a skilled user can circumvent these policies.

To overcome the lack of security options, the Department of Animal Sciences at Wageningen University developed a browser that runs inside a screensaver. This makes the underlying system inaccessible to the users from the moment the screensaver starts. The browser does not have any menu or toolbar, so the user cannot change any browser setting. This screensaver application is developed for Windows 98; minor changes might be required to run this on top of Windows NT or 2000.

There also exist dedicated exam applications that make the underlying system inaccessible; Questionmark has such a dedicated application available [Questionmark, 2002].

Linux allows several different measures. If only one browser window is needed during the examination, the window manager can be disabled. This measure is quite powerful and requires little effort. The window manager is the application that manages the placement, resizing and starting of all application windows. Without a window manager, a user will only see the web browser, without title bar, close button, or minimize button. The user thus cannot start new applications, iconify a chat window, or hide it under the web browser. It is, furthermore, impossible to switch between multiple web browser windows, although several web browsers can show several sites in one window using tabs. Only if the web browser itself starts another program, the student has access to that program. If, for example, the browser configuration is accessible and no measures against superfluous software are taken, a student could change the browser’s configuration in such a way that an xterminal is spawned for a pdf file. The xterminal can then be used to start any application. It is therefore important to secure the browser configuration and to take measures in level II. The Mozilla browser we used in our tests failed to function without a window manager. Several other browsers, however, did run correctly.

If multiple windows are desired for e.g. an open-web exam, there are also several very minimal window managers that can be used. The Lightweight Window Manager (LWM) allows multiple windows, but there is no menu and there are no icons. Several other window managers can be used for restricted setups, like sawfish, scwm and wm2.

The browser functionality can be reduced as well. We managed to configure the Mozilla browser to hide all menus and toolbars, disabling, therefore, access to most functionality (e.g. changing the configuration). Xmodmap can disable function keys and modifier keys (control and alt), thus disabling shortcut-key actions (e.g. Open URL). This also blocks access to other shortcut-key functionality such as ‘change virtual console’ (control-alt-F1) or ‘terminate X server’ (control-alt-backspace). The Xfree86 X server, the default X server on most Linux distributions, can easily be configured to disable all but the left mouse button, thus disabling for example the right-click pop-up menu.

### 4.3 Level II – File system restrictions

To fully prevent access to superfluous software, the exam computer needs to be stripped from such software. A search for executables yielded over 200 executables in the c:\windows\ directory of a typical Windows 98 system and yielded over 400 executables in the c:\winnt\ directory of a typical Windows 2000 system. Most of these executables (e.g. telnet.exe, ftp.exe, winpopup.exe) have no uninstall program, and, therefore, have to be selected manually. A problem that arises with such a stripped system is that service packs cannot be installed anymore. If a service pack is critical nevertheless (e.g. service packs against the '*smbdie'* exploit or the older '*winnuke'* exploit), the exam computer has to be reinstalled from scratch.

On exam computers with the NTFS file system, it is possible to limit the access to specific executables for a group of users. As with executable stripping this requires quite some effort and detailed knowledge of which executables are required for a normal functioning desktop. An advantage over stripping is that users in a different group can still use the computer as a normal desktop. Service packs, moreover, will also install correctly.

A Linux system can be stripped as well to remove superfluous software. A typical Linux system, however, has over a thousand executables installed. Furthermore, there is a high likelihood that security updates cannot be installed after stripping. The perl interpreter, for example, can be used for network connections, but is often required to install packages.

All common Linux file systems allow access control to executables. User groups can be denied access to an executable. Because it is well documented which executables are required for a functional desktop, it is very well possible to restrict access to all other executables. If not correctly configured, however, such a setup might interfere with normal system operation. The advantage over stripping is that security updates and such might fail on a stripped system but they will install correctly on a system with file system restrictions.

Another measure for the same problem is to place the exam user in a *chrooted* environment. Chroot is the irreversible change-root utility. It sets a certain directory as root directory. Users in a chrooted environment have, therefore, only access to files within or below their root directory; all other files are invisible. A chrooted environment is, therefore, a very powerful option to remove superfluous software from the user's environment. The advantage of a chrooted environment over modified file system permissions is that a chrooted environment, by its nature, cannot interfere with the normal system operation. Setting up a functional chrooted environment is, however, not a trivial task and requires knowledge and effort.

### 4.4 Securing the network

Essentially, networks are communication channels. Since any communication channel may in principle be used to exchange answers, or to find online answers, (e.g. in the on-line Encyclopædia Britannica), a secure network is essential. Fortunately many exams require communication with an LMS only: the fewer the number of communication channels, the fewer the options for misuse.

Instead of asking which type of communication should be disabled, it is better to ask which communication should be allowed. Often only network traffic to the LMS is required, but an exam computer might require more communication channels in order to operate correctly.

Exam computers have some basic requirements. To access the LMS, the exam computer has to resolve the host name of the LMS; the exam computer thus needs access to a Domain Name Service (DNS) server. If DNS access is disabled, the exam computer needs different means to resolve the LMS host name (e.g. the /etc/hosts file on Linux systems). The web browser on the exam computer, furthermore, might be configured to use a proxy server; proxy access thus may be necessary.

Many client computers depend on some Network Operating System (NOS; e.g. Novell, Windows NT, or NIS). Obviously, NOS communication should be allowed, if the client computers depend on it. By doing so, however, a lot of NOS services will be available too. Many network operating systems feature communication facilities such as chat, email, and file sharing. Critical inspection of those facilities is important if NOS access is required for an exam computer, and additional restrictions might be necessary.

### 4.5 Level III – Local network restrictions

Crippling the exam computer’s DNS configuration might disable much communication functionality with little effort. Many communication applications depend on the DNS service, and thus will fail without a proper configuration. Obviously, the browser on the exam computer should be able to access the LMS; it thus requires LMS host-name resolution to function properly. Even without access to an external DNS server, such name resolution is possible, provided that the LMS name and IP address are stored locally in the client's hosts file (/etc/hosts on Unix, c:\windows\hosts on windows). It should be kept in mind, however, that most unwanted communication programs function without DNS server access if the user knows the correct IP address by heart. Additional restrictions at levels I and II are, therefore, recommended.

Another simple measure is to configure the browser on the exam computer to use the LMS server as proxy server. This will limit the browser to pages on the exam server. This measure only restricts the browser, but it requires very little effort. The protocol for proxy servers, unfortunately, is slightly different from the http protocol. Some browsers, therefore, will not function with this configuration, most notably older Mozilla versions.

| | |
|---|---|
| 192.168.10.20 | client computer, with network card eth0 |
| 192.168.10.1 | gateway for client computer |
| 192.168.100.30 | LMS server running the exam |

the routing setup for that client would be:

```
route add -host 192.168.10.1 eth0
route add -host 192.168.100.30 gw 192.168.10.1
```

*Figure 3: Installing a crippled routing table on a Linux system*

A firewall-like solution on the exam computer is to install a crippled routing table. A simple script can be executed on every client just before the exam [Fig. 3], requiring little effort. The exam computer cannot reach anything besides all routers and the exam server after removing the default gateway and installing a static route to the exam server. Also peer-to-peer communication between student computers can be disabled with a static routing table. This is comparable to filtering in the ISO OSI Network layer.

On Linux clients with the *dhclient* DHCP client software, the /etc/dhclient-exit-hook script can be used to automatically setup this secured routing table after the DHCP information is received.

### 4.6 Level IV – Upstream network restrictions

Restrictions in the network are much more secure than restrictions on the exam computer. Network hardware is usually in a locked room, physical access is, therefore, not possible.

The most common way to restrict network communication is to install a firewall directly upstream from the exam computers (filtering on the ISO OSI Network and Transport layers) and to disable any peer-to-peer communication between exam computers (filtering on the ISO OSI Data link layer). Most universities already employ routers and switches that can be used to realize firewall functionality. To stop peer-to-peer communication, a switch can be configured to put all exam computers into a private Virtual Local Area Network (VLAN, [IEEE-SA, 1998]) with isolated ports. To restrict upstream communication, the router can be configured as firewall for this VLAN. Even if the network hardware is only used for the exam computers, such a network setup requires detailed knowledge. If the network hardware also services other clients, however, the complexity of the setup could increase dramatically.

If the network between the computer room and the exam server is trusted, IP spoofing can be disabled as well using VLAN technology. The switch ports for the exam computers should be configured to put all exam computers into a separate VLAN. The router should be configured to allow only traffic on that specific VLAN to use the IP range of the computer room.

If the network hardware for the exam computers has no firewall capabilities, there are several options. Linux and OpenBSD, for example, offer secure firewall functionality; any ordinary computer can be converted into a firewall. Another option is to use a network interface card (NIC) with embedded firewall on every exam computer. An example is the 3Com *embedded firewall solution* [3Com, 2002]. It offers central administration, and can update the firewalls for all exam computers simultaneously.

A proxy server may be used as an extra security extension to a firewall setup, since it allows filtering on the ISO OSI Application layer. It can be used to disable for example the chat facility in the LMS itself. In the closed-web situation, the firewall should allow traffic to the proxy server only, so all other protocols besides http will be stopped, and the proxy server should allow connections to the LMS only. In an open-web exam, however, the proxy server should allow connections to any web site; the firewall should still allow connections to the proxy server only. The Squid proxy server can be used for these purposes [Pearson, 2002]. It has a high performance and supports very flexible filtering. Setting up a proxy server for logging requires little effort, but configuring it to filter specific URLs requires detailed knowledge and much more effort.

If the network between the exam computers and the exam server is not trusted, a Virtual Private Network (VPN) can be used to stop IP spoofing. Setting up a VPN is, however, not trivial. The exam server should only allow connections to the exam from IP addresses within the private VPN range. The exam computers should have VPN software installed, or a VPN gateway should be available on the same network segment. If a VPN gateway is used, it should only allow access to the VPN from the exam computers. A VPN setup, furthermore, has the advantage that it is possible to link the VPN access to the users identity. A number of VPN implementations can interface with smart cards, fingerprinting, and iris-scan technology. The most dominant framework for VPNs is Internet Protocol Security (IPsec), developed by the Internet Engineering Task Force (IETF) [Kent, Atkinson, 1998].

## 5 Security requirements for specific exams

Depending on the exam requirements, some security levels need more attention than others. Apart from the exam requirements, several factors from the organization will also affect the decisions. For example the available hardware, the knowledge and experience of the technical staff, organizational issues like how access to network hardware is organized, and also the number of exam computers, will affect application of the four level security model.

Securing a web-based exam should start with measures on level IV. In many computer rooms, however, the network architecture is not designed to restrict communication. Only some high-end switches can be configured to disable peer-to-peer communication. The upstream network, furthermore, might be a fiber backbone; a firewall or filtering proxy server thus cannot be put into place easily. If the number of exam computers is very high, new network hardware such as the 3Com embedded firewall solution is probably feasible; it can secure large numbers of exam computers with little effort. If securing level IV is not a viable option (e.g. only a small number of exam computers), then level III needs extra attention.

In section 5.1 and 5.2 the four level security model will be applied to two very common situations. These two examples also demonstrate that the situation in which an open-web exam requires additional software during the examination is the most difficult situation to secure. In section 5.3, the four level security model is applied to the situation at Wageningen University.

### 5.1 Open-web exams

For open-web exams it is not an option to make all network resources unavailable to the user. The user should be allowed to use the World Wide Web (WWW) to gather information; security levels III and IV should thus allow access to the Internet for http (port 80) or to a proxy server that allows unrestricted http access. Because the WWW offers a lot of options for data exchange, and because the WWW access is unrestricted, logging is needed on levels III and IV. In such a situation with open WWW access, it is important to assess security issues at levels I and II critically.

Some exams only need a limited part of the World Wide Web. In such a situation, levels III and IV can be configured to allow access to those specific web sites. If those web sites do not have facilities to for data exchange, the situation is not much different from a closed-web exam.

### 5.2 Extra software needed at the exam

If some specific software is needed during the exam, for example Matlab, it will be very hard to secure levels I and II. There is a considerable effort needed to find the capabilities of a program. Matlab for example, can start other software and it can open network connections. Because Matlab can start other programs it will be extremely difficult to secure level I. Because it can open network connections it is important to pay extra attention to level III and IV.

### 5.3 Implementation example

Most web-based exams only require a web browser on the client. At Wageningen University, both a Windows-based exam client, and a very secure Linux-based exam client were developed. The Windows-based client is secured on level I and II. The browser on this client is configured to use the LMS server as proxy server; only the browser is thus secured on level III. The Linux-based client is fully secured on level I, II, and III. Level IV is not secured because of both the organizational structure at Wageningen University and because of the network topology in the computer rooms. A local ICT department administers the exam computers. A different department, however, administers the network hardware. Securing all four levels would, therefore, require much more organization than securing levels I, II, and III. The exam computers, furthermore, are connected to the same network hardware as other, normal, client computers. A lot of effort is, therefore, required to configure the network for each computer room separately. Because the exam computers are sufficiently secured at levels I, II, and III it is not worthwhile to secure level IV.

## 6 Conclusion

Web-based exams are more vulnerable to fraud than regular exams. This article describes the most important security issues for web-based exams. It also presents a comprehensive set of measures organized in a four level security model. The levels present user interface restrictions on the client, file system restrictions on the client, communication restrictions on the client, and communication restrictions on the network. The security model supports selection of a specific combination of measures for a specific exam setting. When applying the model, the number of exam computers, the organizational structure, the experience and knowledge of the technical staff, and the available facilities should be taken into account.

Organizations that plan to offer web-based exams should take into account that restrictive measures are necessary during these exams. Investing in network hardware or changing the network topology could reduce the effort required to implement these measures.

**Acknowledgements**

The authors would like to thank Gert Klein and Gerard Folkerts for their information and help on Windows security.

## References

[Blackboard, 2002] Blackboard: “Blackboard Learning System”; (2002) http://products.blackboard.com/cp/release6/LSR6WP.pdf

[IEEE-SA, 1998] IEEE-SA Standards Board.: “IEEE 802.1Q: Virtual bridged local area networks”; (1998) http://standards.ieee.org/getieee802/download/802.1Q-1998.pdf

[Kent, Atkinson, 1998] Kent, S., Atkinson, R.: “RFC2401: Security Architecture for the Internet Protocol”; (1998) http://www.ietf.org/rfc/rfc2401.txt

[Pearson, 2002] Pearson, O.: “Squid, a user's guide”; (2002)

http://squid-docs.sourceforge.net/latest/html/

[Siyan et al. 1997] Siyan K., Hawkins N., Wettern, J.: “Inside Tcp/Ip: A Comprehensive Introduction to Protocols and Concepts”; New Riders Publishing (1997)

[Questionmark, 2002] Questionmark: “Perception Secure Browser product info”; (2002) http://www.questionmark.com/uk/infosheets/perception_secure_browser.pdf

[Richardson et al. 1998] Richardson T., Stafford-Fraser Q., Wood K., Hopper A.: “Virtual Network Computing”; IEEE Internet Computing, Vol.2 No.1, Jan/Feb (1998) 33-38.

[Symantec 2002] Symantec.: “PcAnywhere fact sheet“; (2002)

http://enterprisesecurity.symantec.com/content/displaypdf.cfm?pdfid=35&EID=0

[Sharpe, Warnicke 2002] Sharpe R., Warnicke E.: “Ethereal user guide”; (2002)

http://www.ethereal.com/distribution/docs/user-guide.pdf

[3Com, 2002] 3Com: “Embedded firewall solution datasheet”; (2002) http://www.3com.com/other/pdfs/products/en_US/400741.pdf