SecTULab: A Moodle-Integrated Secure Remote Access Architecture for Cyber Security Laboratories

Joachim Fabini
joachim.fabini@tuwien.ac.at
Institute of Telecommunications,
TU Wien
Vienna, Austria

Alexander Hartl
alexander.hartl@tuwien.ac.at
Institute of Telecommunications,
TU Wien
Vienna, Austria

Fares Meghdouri
fares.meghdouri@tuwien.ac.at
Institute of Telecommunications,
TU Wien
Vienna, Austria

Claudia Breitenfellner
claudia.breitenfellner@tuwien.ac.at
Institute of Telecommunications,
TU Wien
Vienna, Austria

Tanja Zseby
tanja.zseby@tuwien.ac.at
Institute of Telecommunications,
TU Wien
Vienna, Austria

ABSTRACT

The Covid-19 crisis has challenged cyber security teaching by creating the need for secure remote access to existing cyber security laboratory infrastructure. In this paper, we present requirements, architecture and key functionalities of a secure remote laboratory access solution that has been instantiated successfully for two existing laboratories at TU Wien. The proposed design prioritizes security and privacy aspects while integrating with existing Moodle eLearning platforms to leverage available authentication and group collaboration features. Performance evaluations of the prototype implementation for real cyber security classes support a first estimate of dimensioning and resources that must be provisioned when implementing the proposed secure remote laboratory access.

CCS CONCEPTS

• **Applied computing** → **Distance learning**; • **Networks** → **Firewalls**; • **Security and privacy** → *Multi-factor authentication.*

KEYWORDS

cyber security teaching, remote access laboratories, eLearning

ACM Reference Format:
Joachim Fabini, Alexander Hartl, Fares Meghdouri, Claudia Breitenfellner, and Tanja Zseby. 2021. SecTULab: A Moodle-Integrated Secure Remote Access Architecture for Cyber Security Laboratories. In *The 16th International Conference on Availability, Reliability and Security (ARES 2021), August 17–20, 2021, Vienna, Austria.* ACM, New York, NY, USA, 11 pages. https://doi.org/10.1145/3465481.3470034

1 INTRODUCTION

The last years have witnessed a high and continuously increasing student interest and demand for cyber security classes and laboratories. Because of common space limitations and costs, cyber security laboratories are densely populated, establishing a main bottleneck in the maximum number of students per semester or per supervision unit. During pandemics, restrictions with respect to minimum laboratory seating distance and maximum number of students per lab turned out to be prohibitive for presence cyber security laboratories.

While lectures could be converted to distance teaching with acceptable effort, the first year of the Covid 19 crisis has identified access to laboratories as a critical factor in cyber security teaching. This development was paralleled by a huge increase – in some cases almost doubling – of the student count enrolled for communication networks and cyber security master classes and labs at our institute. One potential reason is that the short-notice canceling of other labs triggered the migration of students towards compulsory labs that managed the conversion to distance learning labs within shortest time.

At our university, key challenges in the conversion of presence laboratories to distance learning laboratories turned out to be, among others, the following:

(1) `Specialized hardware and software:` Planned for presence face-to-face teaching, many lab Personal Computers (PCs) are equipped with special software, hardware, or special licenses. It is not possible to run them on the students' home PCs.
(2) `Student PC restrictions:` The creation of a virtual image of the lab computer to enable the execution of the lab (as a virtual guest, e.g. with VMware, VirtualBox, Xen, etc.) on the students' home computers is usually not possible due to legal or technical reasons. Technical reasons include, but are not limited to the high resource demands for data analysis in typical cyber security laboratories. Additionally, the virtual lab image risks to compromise the learning objective whenever students can exploit their local root privilege to analyze the virtual lab image. Assignments in communication network or cyber security classes frequently target the discovery of

ARES 2021, August 17–20, 2021, Vienna, Austria

ACM ISBN 978-1-4503-9051-4/21/08…$15.00
https://doi.org/10.1145/3465481.3470034

secrets. These secrets are commonly hidden within protocol communications or virtual network topologies that reside within the lab PC image. Privileged student access to the running lab PC image allows analysis of the image, which can reveal the needed secrets. Obfuscation is a possible but demanding countermeasure that may not be applicable to all assignments.

(3) `Remote access protocols:` Protocols that allow remote access to lab PCs include Remote Desktop Protocol (RDP) or Virtual Network Computing (VNC). These are unsecured by default and should not be used over the Internet without additional protection, as they open preferred gateways for attackers. Observers can reproduce input and screen by recording packets on the network. Even if a Virtual Private Network (VPN) to the campus network is used, observers can eavesdrop on the protocols within the campus or the institute network. End-to-End (e2e) encryption is therefore desirable, but challenging to achieve.

(4) `Institute firewall:` Lab PCs are typically located or connected behind the institute firewall. Remote access to these labs depends on selectively opening the institute firewall - which can lead to serious security breaches.

(5) `Group cooperation:` Labs relying on student group cooperation are challenging to implement with remote access and cause additional complexity.

This paper presents goals, architecture, performance evaluation and lessons learned from the design, implementation and management of SecTULab: a Moodle-integrated, secure remote access solution to an existing laboratory infrastructure at the Institute of Telecommunications of TU Wien, Austria. Although this paper focuses on two particular use cases, we emphasize that many of the presented concepts, primitives, and implementations are generic in nature and reusable for other application areas, too. Basic prerequisites being met – most notably the need for secure remote access to standard networked lab PCs and an existing Moodle installation – the proposed architecture may support secure group access to laboratories for a wide variety of studies like, e.g., architectural sciences, chemistry, computer science, etc. , as well. However, there is no "one-size-fits-all" solution to securing remote lab access. In some cases the earlier-mentioned prerequisites may be a prohibitive factor for deploying parts of the SecTULab architecture. In particular, successful cyber security teaching formats like capture-the-flag style competitions or others do not fit into the targeted lab pattern.

Due to time and resource pressure the first prototype's goal was defined pragmatical: to support about 80 master class students in completing the lab without compromising the existing network security architecture of the institute. An additional barrier in achieving the goal was the need for configurations in central infrastructure like servers and switches while facing a ban on entering the campus buildings. The result was a first prototype, which, even though developed and implemented within a short time, worked and delivered promising results.

In the subsequent TU-Wien-sponsored SecTULab project the prototype was extended, modularized and documented. This improved version was then tested by converting a cyber security teaching laboratory to distance learning in fall and winter 2020. The lessons learned from the realization of these two prototypes will be presented in this paper.

The remainder of this paper is structured as follows: a summary of related work in section 2 is succeeded by section 3 that compiles a list of high-level goals and requirements for secure remote lab access. Section 4 discusses architectural decisions and design options of a secure remote lab access implementation that meets the aforementioned requirements and goals. Subsequently, section 5 presents selected lab monitoring results and evaluations, followed by a review of lessons learned and limitations of the remote lab access. Section 6 concludes and summarizes the paper.

2 RELATED WORK

Many institutions offer lab access to their students not only onsite but also via the Internet. Several publications address remote access for teaching purposes, a recent one [5] proposes a Linux based variant.

Some authors suggest virtualization solutions for teaching purposes in different flavors, e.g. [2], [6], [7], [4], or [12]. In [11] the focus is set on load balancing and selection of virtual machines to share the load.

Some publications focus on the design of lab content specifically tailored for remote access, e.g. [3] and [9], or [10] using Docker containers. Several authors, e.g. [8] and [1] introduce a cloud based solution.

In our publication, we propose an architecture that enables secure remote access to lab PCs for groups of students. Two main characteristics differentiate it from earlier work: first, we target e2e secured access from the student's home PC to one of the standard lab PCs that are part of an existing on-site cyber security laboratory. Second, the solution is tightly integrated with the existing campus Moodle eLearning platform and campus Single-Sign-On (SSO) in order to support group based lab access and additional security improvements.

2.1 Contribution of this work

The following list summarizes the main benefits of the proposed security framework in supporting cyber security teaching and the main contributions of this paper:

- Requirements, design, and architecture of a framework that supports secure remote access to existing laboratory PCs.
- Seamless integration with Moodle, one of the main eLearning platforms and existing SSO infrastructures.
- Explicit provisioning of architectural support for enabling student group collaboration in times of distance learning.
- Advanced security features that minimize the attack surface of the existing infrastructure – networks and systems.
- Energy saving abilities by starting resources on request only and stopping them whenever idle.
- Software distribution and maintenance platform to enable reproducibility in the laboratory.
- Evaluation of the resource usage of two remote laboratories as an initial guideline for the dimensioning of similar remote access laboratories.

3 HIGH-LEVEL GOALS AND REQUIREMENTS

This section summarizes high-level goals and requirements on the secure remote access to existing laboratory infrastructure.

Project goal is the development of a generic architecture including documentation and implemented sample modules to support the provision of cyber security group labs with secure remote access to lab PCs. The generic solution mandates that the proposed access is limited to secure remote control of laboratory PCs using a mouse, keyboard, and screen (i.e., "remote desktop").

The vision is that well-documented lab modules will be made available within the campus network as building blocks of a template lab. By adapting and integrating template lab parts, lab managers of other institutes can implement secure remote access for their lab infrastructure, too. Open protocols, pre-built modules and sample implementations are complemented by community-supported mailing lists to support installation, configuration and maintenance of the secure remote lab access.

The following sub-goals and requirements are central to the proposed secure remote lab access solution:

- `Secure and privacy-preserving remote access`: Implement secure and privacy-preserving remote access for students from their home laptops to existing laboratory PCs to support distance teaching of cyber security classes. User and control data must be encrypted and protected e2e against eavesdropping and man in the middle (MITM) attacks.
- `Multiple platforms`: Support the most common platforms with one single, generic implementation: Linux and Microsoft Windows as lab PC operating system, Linux, MacOS, and Microsoft Windows as student PC operating system.
- `Moodle integration`: The remote lab access solution is to be integrated with Moodle or other eLearning platforms to control access to lab PCs. Topics of relevance include, but are not limited to: Moodle authentication (e.g., leverage existing campus SSO authentication functionality, including optional two-factor authentication), the use of Moodle groups (e.g., to assign the same lab PC to all students of a group), as well as an additional reduction of the attack surface of the remote access lab.
- `Group support`: Support and on request enforce student group collaboration in solving their laboratory assignments.
- `Reproducible research`: all PCs in the laboratory should rely on identical operating system and software packages, ideally identical images. Similarly, controlled and centralized roll-out and installation of updates should be preferred over uncontrolled local updates.
- `Minimum attack surface`: The additional attack surface created by remote student access to the existing institute and laboratory infrastructure must be minimized. Security mechanisms should prevent attackers from the Internet or intranet from misusing remote lab access to attack the institute's infrastructure. A single hardened publicly accessible server interface with dynamically controlled firewall rules should be sufficient for secure remote lab access.
- `Scalability`: The lab should scale well with demands in terms of network connectivity and physical or virtual lab PCs.
- `Automation`: The setup and integration of additional physical or virtual lab PCs should require a minimum of human intervention. The cleanup, lab setup and configuration for a new laboratory term should be to a large extent automated while satisfying the reproducibility criterion mentioned earlier.
- `Monitoring`: Monitor required resources, in particular network and server capacities, both for physically existing computers and for virtual lab PCs. The resulting, anonymized data is expected to build up a knowledge base that supports lab administrators in sizing hardware and network for various requirements and prototypical use cases. Even though requirements differ, we argue that empirical values for typical use cases can serve as an important indication of feasibility and dimensioning for potentially interested parties.
- `Energy savings`: enable substantial energy savings by starting laboratory PCs on request only and shutting them down when no longer needed.
- `Free software`: Use exclusively freely available, well established and maintained, royalty-free, preferably open source software for the implementation.
- `Security teaching`: Laboratories that teach security should rely on open state-of-the art security implementations to counter the commonly encountered security-by-obscurity paradigm. The architecture and mechanisms used to secure the laboratory infrastructure will, therefore, become themselves topics in the security teaching context.

3.1 Hardware and Networking Requirements

Minimum networking requirement for the secure remote access lab is that the lab firewall's outer interface is assigned a public Internet Protocol version 4 (IPv4) and/or Internet Protocol version 6 (IPv6) address. All PCs within the lab must be networked, reachable from the firewall's internal interface and have sufficient computing capacity for running a remote desktop server like VNC or RDP.

For the fully-featured secure lab access solution that we present in the following sections, some additional components and features have been used. Due to the modular structure these parts can be omitted or replaced. The following list summarizes the extended component list:

- `Lab PCs`: standard PC, equipped with one or more network interfaces, decent hardware, memory and storage that can handle the lab assignment and a remote desktop server. These highly generic requirements should be satisfied by most labs (whether cyber-security or not).
- `Lab firewall`: standard server hardware running Linux, equipped with one or more network interfaces that can range from 1 GBit/s copper to 10 GBit/s or faster optical network interfaces, depending on the lab size and expected remote lab access traffic.
- `Connectivity`: The lab firewall's outer interface – or at least port ranges of it – must be reachable from the public Internet. The path capacities within the lab and on the lab firewall's outer interface must match the demands of the lab. The existing lab PCs should ideally be connected to a dedicated, isolated Virtual Local Area Network (VLAN) and

IP subnet (private Internet Protocol (IP) address range is sufficient).

- `Switch:` A managed switch that supports VLANs, trunks and interface aggregation (bonds) is recommended, in particular when the remote access solution is planned to scale later on with demands.
- `Campus Infrastructure:` Enhanced security features that are proposed in the following rely on existence and connectivity of a campus Moodle hosting the course's eLearning web page, potentially integrated with a campus SSO service.

More details will be identified in the following discussion on secure remote access architecture and implementation.

4 ARCHITECTURE AND IMPLEMENTATION

Figure 1 depicts a schematic block diagram of the secure lab access architecture, its building blocks and interactions.

4.1 Basic Architecture

Main goal of the proposed security architecture is an e2e secured, privacy-preserving remote desktop connection between the student's home PC at the upper left and one available laboratory network PC at the lower right of Figure 1.

Three distinct firewalls must be traversed on this behalf by the remote lab access traffic. First is the existing campus firewall (vertical rectangle, at the left) that separates the Internet from the trusted campus network but imposes restrictions on very specific services only (for instance mail). Second, the existing institute firewall (centered horizontal rectangle) isolates the institute network from the outer world. The institute firewall blocks incoming traffic except the one targeting hosts connected to a dedicated Demilitarized Zone (DMZ) subnet. The DMZ is isolated at layer 2 from the main institute network by using a dedicated VLAN.

Third firewall and main functionality-providing component to transit is the newly added lab firewall, identified by an orange rectangle at the center of Figure 1. The lab firewall has an outer interface in the institute's DMZ labeled `Public.lab`, a control interface `Ctrl.lab` that will be described in subsection 4.3, and the internal interface as standard gateway of the laboratory network. In our setup the lab network uses a private IPv4 range and a public IPv6 prefix due to the shortage on public IPv4 addresses. Therefore, the lab firewall controls IPv4 and IPv6 traffic and implements a Network Address Translation (NAT) for IPv4 connections to the institute DMZ and the Internet.

The laboratory itself, as shown in the lower right corner of Figure 1 uses an own IPv4 and IPv6 connected subnetwork that is isolated from the institute network at layer 2 by using an own VLAN. The lab currently consists of 23 standard common-of-the-shelf physical PCs[1] and a flexible number of virtual lab PCs that are running on a hypervisor. Virtual lab PCs can be instantiated on demand and connected to the laboratory VLAN/subnet to complement the existing base of physical lab PCs. Physical and virtual lab PCs run centrally-administered, identical images, meaning operating system, software installations, and configurations.

Student laboratory accounts are configured on the lab PCs as centralized Network Information Service (NIS) users with NFS-shared home directories. The lab networking infrastructure is configured to isolate all existing lab PCs. That is, security measures prevent any two lab PCs – either physical or virtual – from seeing or accessing each other, despite being connected to the same VLAN and subnet. Firewall rules on PCs furthermore prohibit direct TCP access to the NFS share by unprivileged processes, i.e. by students.

4.2 Hardened Network Security

Following the basic architecture description this subsection presents solutions that improve the network security and minimize the attack surface resulting from offering remote lab PC access to students.

We argue that the underlying topology depicted in Figure 1 is typical for many universities and teaching institutions. In particular most universities deploy campus and institute firewalls, campus Moodle installations for distance teaching and campus SSO services that the proposed security solutions rely on.

4.2.1 *Campus Single-Sign-On.*

As first step (message `(1)` in Figure 1) the student who plans remote access to the lab visits the cyber security course landing page hosted by the campus Moodle. Access to the course page is conditioned by student authentication (optional Two-Factor Authentication (TFA)) with the Campus SSO service `(2)` and successful enrollment for the cyber security course.

4.2.2 *Lab Credential Distribution.*

Lab credential distribution was done in person in the past and has become a challenge for distance learning. The Moodle integration of the lab supports automated and secure distribution of lab PC access credentials: only the SSO authenticated student can view his/her own credentials stored in Moodle.

4.2.3 *E2E encrypted traffic.*

A mandatory requirement for the presented security architecture was e2e encryption of the entire remote access protocol traffic. Remote desktop access protocols, most notably RDP or VNC, use non-encrypted data transfer or weak encryption algorithms when streaming remote access data. Potential MITM on the transmission path can therefore eavesdrop and compromise the transmitted screen data whenever such protocols are sent unencrypted.

Using one of the available VPNs – either campus or institute VPN – can, on one hand, protect the most exposed part of the path, between the student's PC and the campus firewall or institute firewall. On the other hand, this architecture lacks traffic protection on the subpath between campus firewall or institute firewall and lab PC.

This is the main reason why we opted for a Secure Shell (SSH) e2e tunnel between student PC and lab PC for VNC remote access data. The SSH tunnel terminates at the end hosts, such that unencrypted VNC data is available exclusively within the end user devices, being protected against eavesdropping on the entire network path.

As a further benefit, by using SSH for PC access, we provide students with the possibility for uncomplicated and secure access of the lab PC's file system and simple terminal tasks can be performed using SSH directly without requiring VNC.

[1] Standard PC, Core i7, 16GB of RAM to cope with the data processing requirements of the lab. For remote access a substantially lower resource profile is sufficient.

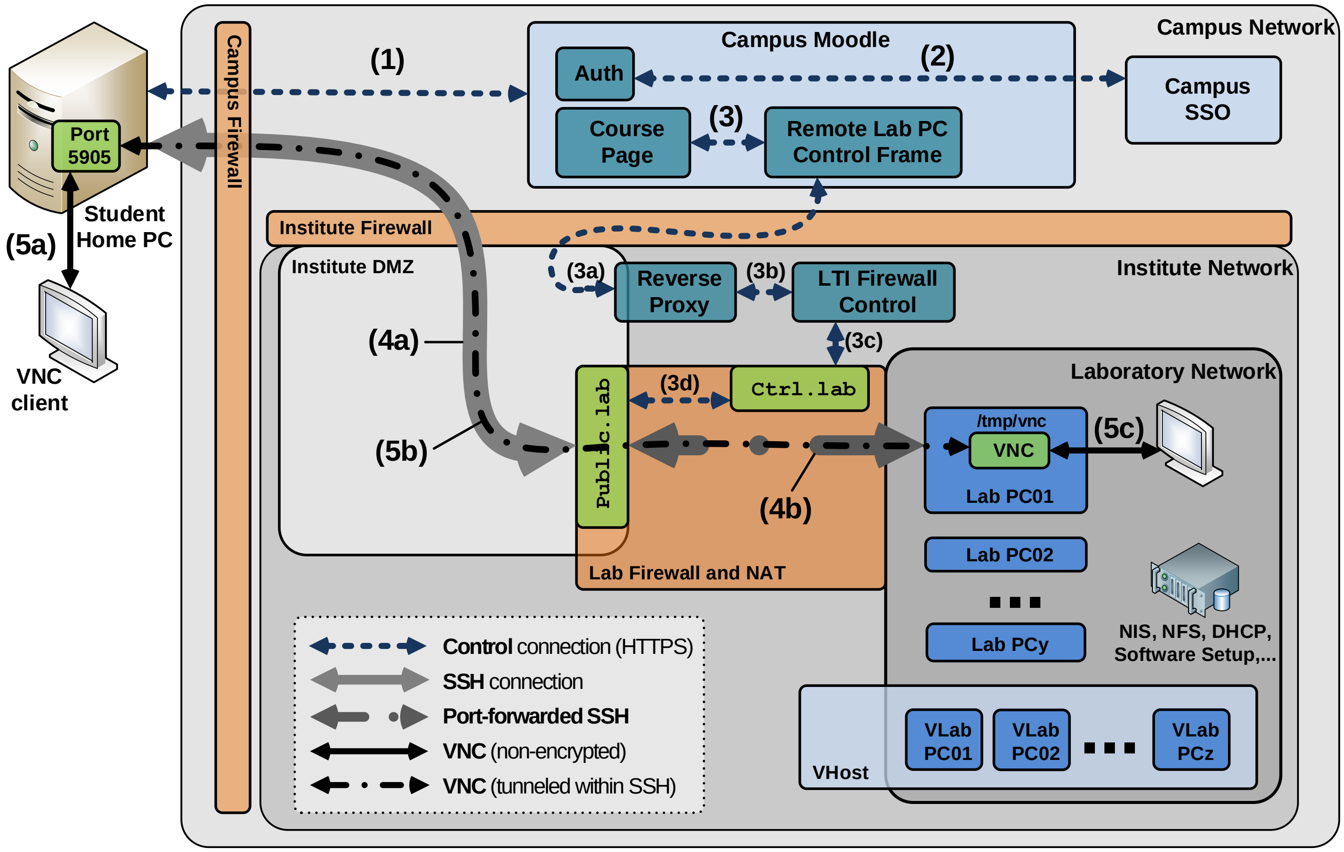


Figure 1: Generic Architecture of Secure Distance Learning Laboratory

4.3 Lab Firewall Interface Access

The lab network can use either a private IP address range (case in which the lab firewall acts as a NAT) or public IP addresses. The outer lab firewall interface in the DMZ drops by default all incoming traffic in both cases. In particular the standard SSH port on the lab firewall's outer interface is closed.

In order to support remote student access while minimizing the attack surface, the lab firewall leverages student data obtained from the Moodle course page. The student home PC's public IP address is recorded when the student accesses the Moodle course page and subsequently used by the lab firewall as source IP address filter for a newly opened communication port.

Going more into technical details, Moodle communication relies on Learning Tools Interoperability (LTI), which is a framework defining interfaces to allow the extension of Moodle courses by custom tools. The course web page includes a `Remote Lab PC Control Frame (3)` as illustrated in Figure 1.

Following successful Moodle authentication for the course, a student can request access to a free lab PC using this frame. The frame data is computed by an `LTI Firewall Control` web server in the institute network. The connection to the `LTI Firewall Control` is TLS-protected and makes use of a reverse proxy in the DMZ (`(3a)` and `(3b)`).

Whenever a student requests remote access to a lab PC, the `LTI Firewall Control` executes the following steps outlined in the sequence diagram in Figure 2:

(1) Verify availability of a lab PC.
(2) Start the lab PC.
(3) Allocate an available random high port on the lab firewall's outer interface and instantiate a source IP address filter for this port to allow the student home PC's address only and drop all other sources.
(4) Establish forwarding rules to dispatch traffic between the allocated port on the external lab firewall interface and the started lab PC's standard SSH port.
(5) On successful completion report the newly opened lab firewall port number to the user via the LTI interface.

On viewing the successful lab PC allocation result the student starts an SSH session with VNC port forwarding on his home PC to the transport address shown in the LTI frame (message `(4a)` in Figure 1)[2]. The lab firewall accepts the student PC's source address on its outer port and forwards the SSH packets to the allocated lab

[2] An example command line for Linux lab PC using the Unix VNC socket /tmp/vnc on the lab PC is shown below. The use of port 5905 on the student's home PC for SSH forward is preferred over the standard VNC server port 5900: the latter may be in use by a locally running VNC server, causing questions and need for student support. ssh -L 5905:/tmp/vnc -p 30999 student01@public.lab

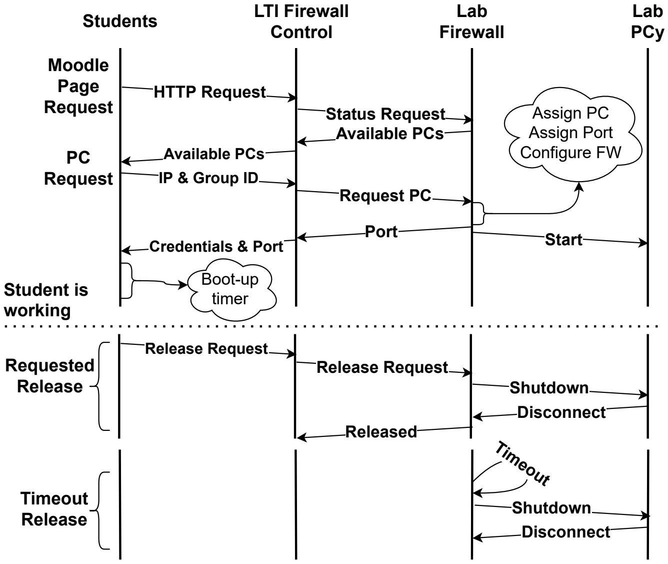


Figure 2: Lab PC request procedure.

PC's standard SSH port (4b). The user authenticates with the lab PC using the NIS credentials obtained from Moodle and opens an SSH session.

For lab PC access the user starts a local remote desktop viewer – in this example we assume VNC – and connects to the local configured SSH forwarding port (e.g., localhost:5905). Data sent to and received from this port (5a) is securely and transparently relayed by SSH between the student PC and the configured remote port on the lab PC (5b, 5c).

By default the lab firewall allows outgoing user-traffic from the lab to the Internet but prevents visibility and data exchange between lab PCs. This policy can be changed by modifying corresponding firewall (iptables) rules.

4.4 Group Support

Integration with the campus Moodle and LTI eases group support as part of the presented remote access architecture. The `LTI Firewall Control` server that has been outlined in subsection 4.3 has access to student group membership data.

On incoming student requests for a new lab PC via Moodle and LTI, the `LTI Firewall Control` server verifies if one of the requesting student's group members has an active lab PC session. Whenever an active lab PC and session is found, the new request will be assigned to the same lab PC. Otherwise a new lab PC is chosen from the existing pool and started for the new request.

At a technical level, assignment of several students to the same lab PC is implemented by the lab firewall redirecting distinct ports on its outer interface `public.lab` to the same SSH port of the physical or virtual lab PC. Following successful SSH e2e connection establishment, the group colleagues can share the same screen – for instance by accessing the same VNC socket on the lab PC when using VNC as remote access viewer.

Students who are members of the same group were advised to stay permanently connected, discuss and cooperate via live web teleconferencing systems while solving their lab assignments. The lab administration did neither provide this service, nor was there a recommendation or endorsement of a specific third-party product. Hence, students were free to use the teleconferencing tool(s) that suited best their needs and preferences.

4.5 Scalability

Essential requirement for the secure remote lab access is that the solution scales well, in particular with respect to network capacity. The remote access network traffic per user depends to a large extent on the specific lab content. Complex full-screen motion video preview or processing may originate huge network traffic for remote access viewing. Many cyber security labs rely on capturing network data with tools like Wireshark and applying subsequent data processing chains. We targeted mainly this use case and found it to generate acceptable traffic, even in the case of log scrolling or life Wireshark capture.

Nevertheless, we argue that network capacity bottlenecks for remote lab access are most likely to involve the lab firewall's network interfaces. We tried several alternatives to overcome such limitations for optical and for copper interfaces and present the solutions in the following.

- `Interface Aggregation`: The server hardware that was planned to serve as lab firewall initially featured two 1Gb/s copper interfaces. Having interface aggregation (bonding) support on the connecting switch we added an additional 4-port 1 GBit/s PCIe card to the firewall. Then we configured two bonds on the firewall and on the switch, one incoming and one outgoing, respectively, each one having a total capacity of 3 GBit/s. The link capacity per session is still limited to 1 GBit/s but with increasing user count the firewall's network throughput was effectively tripled.
- `Trunks and VLANs`: In addition to copper bonds we experimented with optical 10 GBit/s SFP+ interfaces to increase the lab firewall capacity. Extending the lab firewall server by adding a twin PCIe 10 GBit/s interface card yielded a substantial network capacity improvement. However, the limiting factor turned out to be the single optical link between firewall and server cabinet and one single available optical port on the connected switch. On short notice it was impossible to connect an additional optical link between the two servers (located in distinct buildings) and to purchase and configure an additional managed switch for the existing switch stack. Therefore the use of distinct physical optical interfaces for the outer and the inner lab firewall interface was not feasible.
  The adopted solution was to configure the single optical link as trunk that transports two distinct VLANs on both, lab firewall and switch. Logical VLAN interface support on the Linux-based lab firewall offers two distinct network interfaces for the two enabled VLANs. Minimum configuration changes – enabling VLAN, adding VLAN interfaces and modifying the interface names in the firewall configuration – yielded a substantial capacity increase with just one single optical link connectivity without any compromise on lab firewall security.

4.6 Reproducibility

One main requirement of the cyber-security lab is to support addition and installation of new lab PCs with a minimum of effort and to enforce identical replicas of lab PCs. Identical means that all lab PCs must run an identical version of the operating system plus identical software installation in identical configuration. This requirement is paramount in minimizing potential uncertainty sources, i.e., support the reproducibility of research.

As main consequence, software is not being installed locally, on lab PCs on request but exclusively as part of an automated image installation. Lab administrators can trigger an automated Preboot Execution Environment (PXE) boot of lab PCs that reinstalls the complete system image based on a local mirror that caches Linux software packages.

An identical hardware and software environment on all lab PCs is of paramount importance, e.g., whenever implementing cyber security competitions that are part of lab B. Student groups must have a fair means to evaluate the functionality and performance of their solution against the provided reference scoreboard. The proposed reproducibility setup ensures that the final group submissions (being evaluated on a well-protected, but identical submission PC) will (a) be run in an identical environment as the one of the lab PC, including all prerequisites, and (b) match the performance of the preliminary student tests.

4.7 Physical vs. Virtual Lab PCs

When comparing in-presence cyber-security labs against remote access labs, the latter have the benefit of time-based load balancing. While in-presence labs typically require at least one fixed lab time slot allocated per group per week, remote labs are open 24/7. Potential bottlenecks may arise from the need for supervised working time.

Increasing the number of available lab PCs can help in decreasing the supervision and teaching effort. This is why we evaluated the addition of virtual lab PCs. First experiments with virtual guests on top of KVM/qemu/libvirt based hypervisor are promising. These virtual lab PCs can be integrated with the PXE-based boot process that has been deployed for the physical lab PCs. The number of concurrent virtual guest instances, as well as the degree of parallelism both depend on available resources on the hypervisor and the specific application.

A decent resource planning should safeguard comparable user experience for virtual and physical lab PCs. Exceptions apply, most notably in the case of strict timing requirements and applications depending on (near) real-time behavior.

4.8 Energy Saving

Physical lab PCs build the basis of the cyber security lab. During preliminary lab evaluations it turned out the automated shutdown of unused lab PCs is an essential feature. Beside the main benefit of substantial energy savings, when shutting down unused PCs, we can additionally guarantee students to start with a clean environment, which is free of relics from previous groups like, e.g., background tasks.

4.8.1 Starting and Stopping. By default all lab PCs are turned off. The activated Wake-On-LAN (WoL) feature on the lab PCs allow the lab firewall to trigger lab PC boot remotely. Once a student requests a lab PC, the firewall selects an available PC from the pool and sends the WoL trigger packet to this lab PC. By operating WoL-capable network hardware we can start a PC remotely for administrative purposes. Hence, in conjunction with PXE boot, this setup can perform a complete and fully automated, unattended reinstallation of a lab PCs from a remote location (see subsection 4.6).

When a PC is no longer needed, the lab firewall shuts it down automatically. For this purpose, the lab firewall initiates an SSH connection to the appropriate PC and issues a shutdown command. In addition to the benefits mentioned above, we achieve a basic form of availability monitoring using this shutdown procedure. Hence, if a PC becomes unresponsive due to, e.g., kernel freezes or hardware failures, the establishment of SSH shutdown connections fails. In this case, the lab firewall periodically retries to initiate the shutdown and no longer assigns the PC to students until the PC is back online and the SSH shutdown has successfully been issued. To cope with unlikely situations when a PC is already shut down at the time of website- or timeout-triggered release, the firewall sends WoL packets in parallel to shutdown requests. Running PCs will not be affected by the WoL packets, whereas shut down PCs will eventually wake up and become available for the controlled shutdown procedure. Since the lab firewall is aware of non-responsive lab PCs, it might be optionally possible to trigger alarms, e.g., by sending an email message if a PC remains non-responsive for a specified period. In our case, the necessity for email notifications did not arise.

Evidently, this approach meets only basic demands of availability monitoring. In particular, if a PC freezes during shutdown or boot or if WoL fails, the problem is not immediately detected. However, at latest when an unresponsive PC is assigned to students, the problem is detected at the next shutdown attempt and the PC is identified as failed.

4.8.2 Timeout. Students are asked for releasing their lab PC using the dedicated controls within the Moodle web page as soon as it is no longer needed. The lab PC is then shut down and becomes available to other students. This is the preferred release method that students are instructed to conform to.

However, since releasing a lab PC is not required for solving lab exercises, experience has shown that students tend to forget to perform the release manually. An analysis revealed that less than half of the student sessions ended with planned student-triggered lab PC shutdowns. The majority of lab PCs continued to run despite no student being logged in. This raised the need for an automatic lab PC release after a certain idle time.

On the downside, an automatic release procedure might lead to data loss if the determination of idle PCs is unreliable and a PC is shut down while a student is working. For example, not being active in the Moodle platform is a very weak indicator for the student not working actively on his assigned lab PC. The adopted solution was to implement a timeout mechanism on the lab firewall. This module monitors open ports on the lab firewall's outer (Internet) interface for user traffic. If the lab firewall reports no traffic within the timeout interval (conservatively set to half an hour) then it infers on an idle lab PC. Consequently, the lab firewall triggers release of all associated resources – in particular closing of all open ports for this lab PC on the firewall's outer interface – and, eventually, a lab PC shutdown. As long as students have any SSH connection active, no timeout is triggered. Since remote desktop traffic is tunneled through SSH, also any active VNC connection ensures that no timeout is triggered.

4.9 Generic Applicability

The cyber security classes and laboratories described throughout this paper required the use of Linux as operating system on the lab PCs, which recommended the use of VNC as remote access protocol. Non-technical students or laboratories relying on specific licenses may prefer Microsoft Windows as lab PC operating system. When using Windows on lab PCs, our described techniques can be used analogously. In particular, recent versions of Windows provide an SSH server and protocols like RDP or VNC can be used for remote desktop access, using TCP port forwarding to tunnel the remote desktop protocol over SSH.

Instead of NIS and Network File System (NFS), the corresponding techniques and protocols of Windows environments like Server Message Block (SMB) and Active Directory can be used. While WoL is agnostic to the used operating system, we experimented with methods to remotely power off Windows PCs and found that a shutdown can be initiated either using SSH, as we did with Linux, or using the Windows Management Instrumentation (WMI), both requiring a privileged user for issuing the command.

The proposed secure remote access architecture is, therefore, flexible with respect to the lab PC operating system. We recommend the use of Linux as operating system for the deployed controlling server infrastructure, as it yields versatile interfaces for controlling firewalls and obtaining information about active firewall rules. We have not evaluated the use of server operating systems other than Linux for the lab's controlling server infrastructure.

5 EVALUATION AND LESSONS LEARNED

We evaluated our solution for two distinct classes on cyber security, and collected data on resource utilization and usage of our lab PCs. We pursued several goals when collecting the data.

- We aimed to measure the amount of network traffic due to lab usage. We were particular interested in the network bandwidth required for VNC traffic of a substantial number of simultaneously working students.
- By analyzing whether student accesses are bursty or rather are uniform during a given time interval, we wanted to investigate how many lab PCs have to be held available for a certain number of students
- We furthermore wanted to analyze how and when the lab infrastructure is used depending on approaching exercise deadlines and officially supervised lab timeslots.
- Finally, we aimed to analyze whether students work together for solving exercises.

5.1 Lab Characteristics

We implemented the solution for two courses, henceforth referred to as lab A and lab B, respectively. Lab A was held during summer term 2020 with about 80 students and lab B during winter term 2020 with about 60 students. The subjects of both courses were centered around cyber security, hence students had to work on a PC to solve exercises. Lab A consisted of 4 exercises whereas Lab B consisted of 5 exercises. An exercise was made available as soon as the student had solved the previous exercise. Additionally, students had to hand in one report per group per exercise subject to respective deadlines for lab A and only three reports for lab B (combined reports for exercises 1-2 and 3-4). For the second and third report of lab B early submission options were offered that yielded extra bonus points. Students who did not manage to submit by the early deadline missed the bonus points but received hints that supported the solving of the assignments.

For lab A, only the last exercise could be solved directly on the students' home computers, and students had to use the lab infrastructure for the remaining ones. For lab B, 40% of exercises (assignments 3 and 4) could be solved on the students' home computers.

Students were able to access the lab infrastructure 24h a day. However, we offered one timeslot per week when students could get support from supervisors via online meeting tools. We had the a-priori expectation that these reserved timeslots would be the most critical points in time considering resource usage.

5.2 Resource Utilization

Figure 3 shows the distribution of transfer rate of the outer (Internet) interface on the lab firewalls of both labs, observed during the respective semester.

Considering lab PC usage, Figure 4 shows the observed distribution of occupied lab PCs during the respective semester, closely resembling a geometric distribution. Hence, in our settings neither the network bandwidth nor the number of available lab PCs were limiting factors for students.

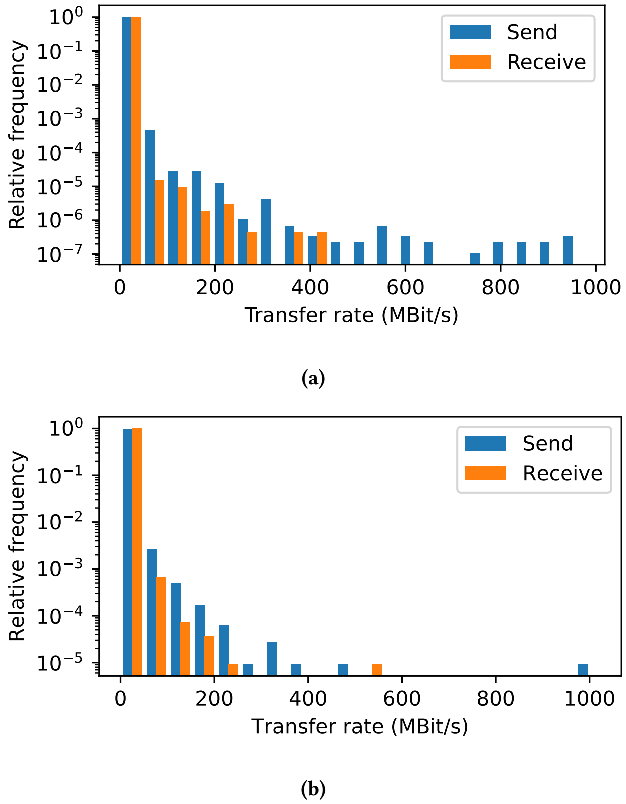
Even though we explicitly asked students to release the lab PC reservation when they no longer need it, among all release operations 45% for lab A and 32% for lab B were issued by timeout. Hence, if all release operations had to be performed manually by students, we would for both labs have hit the limit of available lab PCs with high probability, reinforcing the importance of a timeout mechanism. We received no student complaints about inappropriately triggered timeout releases.

5.3 Student Activity

Figure 5 shows the activity of students throughout the lab. For lab A, we provided supervised timeslots on Tuesdays and Wednesdays, leading to discernible peaks mainly on Tuesdays. Furthermore, as expected, report deadlines have an effect on lab usage.

Worth mentioning is the remarkable increase in student activity before the first early report deadline for lab B in Figure 5 (lower figure, first yellow vertical bar). The extra bonus points seem to be an incentive for students to invest extra effort during a weekend. Overall, student activity is well distributed over time, as Figure 5 and the averaged lab PC usage distribution per working day in Figure 6 confirm.

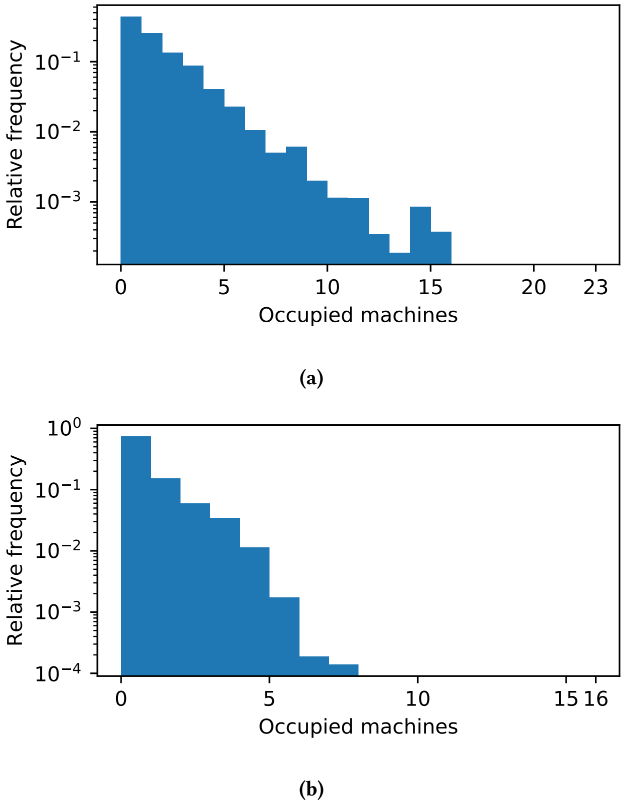
A main benefit of our solution is the opportunity for students to work jointly on solving their exercises. To analyze whether



(a)

(b)

Figure 3: Utilization of Internet-facing lab firewall interface during active times for lab A (upper) and lab B (lower figure).



(a)

(b)

Figure 4: Distribution of occupied lab PCs for lab A (23 lab PCs, upper figure) and lab B (16 lab PCs, lower figure).

students use this offer, we determined for each student the total time when both group members were connected simultaneously and the overall total time when only one was connected, and computed the quotient of both values to measure the amount of group work. Figure 7 shows the distribution of the group work fraction. Hence, without further measures, the sole possibility to work jointly in a group is hardly sufficient to motivate students for group work.

5.4 Lessons Learned

First and most important lesson learned is that existing cyber security laboratories can be converted to distance learning laboratories using secure remote lab access. Secure remote access to existing laboratory PCs can be realized with acceptable effort, using common-of-the-shelf hardware and open software. Some extra effort must be invested for seamlessly integrating secure remote access with authentication and group features of well-established campus eLearning platforms like Moodle.

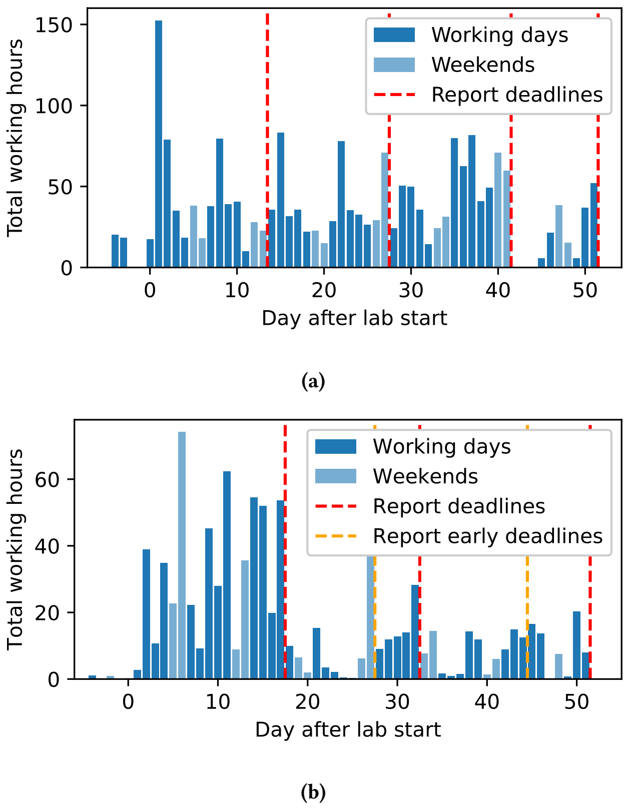
Second lesson learned is that an offering of group cooperation features does not necessarily mean that students use them. If lab assignments resemble the mode of the courses described in this paper, we recommend to motivate students to cooperate with their team partner(s). This motivation can be either an explicit enforcement – for instance a group can not advance to the next assignment unless all group members have confirmed the submission of the prior one – or implicitly, by crafting assignments in such a way that their solving process is supported by (or depends on) group cooperation. However, we definitely recommend positive motivation over enforcement. The effort that students invested into achieving extra bonus points for early submission supports this position as detailed in subsection 5.3.

Third lesson learned is the lacking reliability of explicit manual resource release. The inactivity timeout mechanism that we implemented for remote lab access turned out to be an invaluable help and crucial for successful lab operation. Either because of unavoidable technical issues like, e.g., authentication timeouts, or because of human omission, lab PC reservations are not released reliably. Without a reliable timeout mechanism this may lead to shortage of lab PCs. However, the implemented timeout mechanisms worked reliably, such that neither resource usage nor availability of lab PCs were limiting factors in our setup.

5.5 Limitations

Students' feedback on the secure remote lab access was almost exclusively positive. The cyber security lab participants were thankful for the option to attend the lab remotely and solve their assignments from home.

A minor – mainly technical – challenge turned out to be the command-line SSH tunnel. Few lab participants who were used to



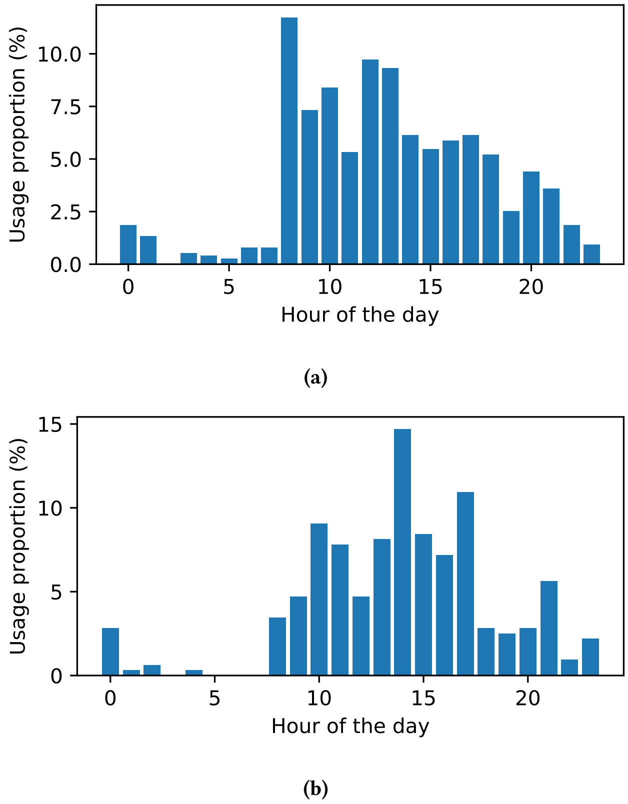
(a)

(b)

Figure 5: Daily usage of the lab infrastructure over entire lab duration for lab A (upper figure) and lab B (lower figure).

case-insensitive operating systems failed to connect and asked for support because of misspelled command line parts. However, this experience raises some question marks as the secure remote lab access is planned to be generic in its concept. If such difficulties are encountered by students of electrical engineering and computer sciences, who are the main audience of the presented cyber security labs, the situation may become worse if the remote access solution is used by students less skilled in terms of communication networks (architects, chemistry, etc.). However, we are confident that wrapper scripts can automate the connection procedure even for non-technical students.

A second potential limitation is related to the use of Microsoft Windows as target operating system for lab PCs. In preliminary tests, we identified a limitation for group access using the RDP protocol. The proposed secure remote access is fully agnostic to the specific tunneled remote access protocol (VNC or RDP) or lab PC operating system. However, the client releases of Microsoft Windows (in particular Windows 10) do not support simultaneous RDP connections from distinct sources to the same target PC. Lab PCs having installed a Windows server license allow two simultaneous RDP sessions (even more when using Microsoft's terminal server license model) but by default these do not share the same screen. As a work-around we recommend a free or commercial VNC server



(a)

(b)

Figure 6: Average distribution of lab PC usage over 24h working day for lab A (upper figure) and lab B (lower figure).

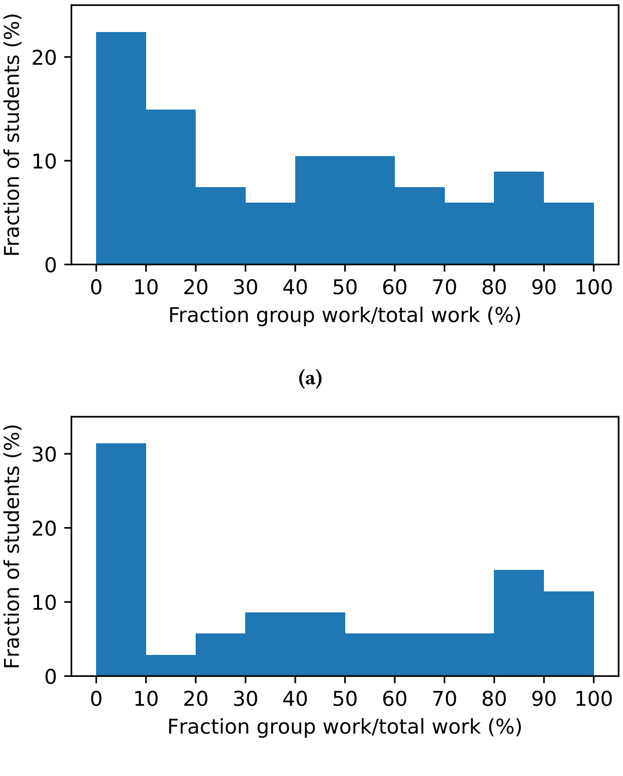
to be installed on the lab PC, which works on Windows client operating system versions, too. We emphasize that this limitation is not one of the proposed secure remote access solution but a limitation of the operating system vendor.

6 CONCLUSIONS AND FUTURE WORK

This paper presented goals, architecture and evaluation of a secure remote access for existing cyber security lab PC infrastructure. Stressing the benefits of integration with campus Moodle group and authentication features, the proposed solution is a well-suited case study for cyber security classes, teaching how to secure and protect existing assets.

It is planned to make the lab modules and documentation available for campus-wide use by July 2021. However, for a research institute it is challenging to plan for high-quality software maintenance and long-term support. This is why a stable source base and a template laboratory configuration will be published to the campus GitLab repository. Once this source base is available, the campus IT department will assume repository ownership and take over maintenance and long-term support. Subsequently, the role of our group is planned to be the one of a user and regular contributor to the secure remote lab access framework.

A final word is due on the lab teaching objective. In interviews, students preferred the flexibility of permanently accessible remote



(a)

(b)

Figure 7: Distribution of fraction of group work for lab A (upper figure) and lab B (lower figure).

labs over fixed on-site lab hours. The working hours distribution shown in Figure 6 unconditionally supports this statement. However, the (subjective) impression of the teaching staff is that distance learning labs fall short of a central component of education, the human factor. In particular, we argue that the cooperation between groups – consisting of questions and explanations, which are tolerated to a certain extent during on-site labs – contributed a substantial share to the overall knowledge and comprehension acquisition of the participants during on-site labs. Despite offering student discussion fora, as well as online- and email support, we did not find evidence of a comparable "learning while explaining" component for remote labs.

Once on-site labs are permitted again, a likely scenario is that compulsory on-site cyber security lab hours will be complemented by optional remote lab access outside of the lab busy-time. We believe that this solution combines the positive aspect of both, comprehension and flexibility.

ACKNOWLEDGMENTS

The secTUlab project for secure remote lab access is supported by the .digitalOffice of TU Wien under grant number DCL017. The authors thankfully acknowledge Gernot Vormayr's initial implementation of the on-site lab that served as a base for subsequent secure remote access extensions and campus Moodle control.

REFERENCES

[1] Mhd Wael Bazzaza and Khaled Salah. 2015. Using the cloud to teach computer networks. In *2015 IEEE/ACM 8th International Conference on Utility and Cloud Computing (UCC)*. IEEE, 310–314.

[2] Charles Border. 2007. The Development and Deployment of a Multi-User, Remote Access Virtualization System for Networking, Security, and System Administration Classes. *SIGCSE Bull.* 39, 1 (March 2007), 576–580. https://doi.org/10.1145/1227504.1227501

[3] Young B. Choi, Shinyoung Lim, and Tae H. Oh. 2010. Feasibility of Virtual Security Laboratory for Three-Tiered Distance Education. In *Proceedings of the 2010 ACM Conference on Information Technology Education* (Midland, Michigan, USA) *(SIGITE '10)*. Association for Computing Machinery, New York, NY, USA, 53–58. https://doi.org/10.1145/1867651.1867666

[4] Te-Shun Chou and John Jones. 2018. Developing and Evaluating an Experimental Learning Environment for Cyber Security Education. In *Proceedings of the 19th Annual SIG Conference on Information Technology Education* (Fort Lauderdale, Florida, USA) *(SIGITE '18)*. Association for Computing Machinery, New York, NY, USA, 92–97. https://doi.org/10.1145/3241815.3241855

[5] Jeremy Diederich, Xianping Wang, Niharika Dayyala, Sundeep Inti, and Ying Luo. 2020. USB Linux: An IT Lab Instruction Tool During COVID-19. In *Proceedings of the 21st Annual Conference on Information Technology Education* (Virtual Event, USA) *(SIGITE '20)*. Association for Computing Machinery, New York, NY, USA, 273–278. https://doi.org/10.1145/3368308.3415372

[6] Ji Hu, Christoph Meinel, and Michael Schmitt. 2004. Tele-Lab IT Security: An Architecture for Interactive Lessons for Security Education. *SIGCSE Bull.* 36, 1 (March 2004), 412–416. https://doi.org/10.1145/1028174.971440

[7] Steven Rigby and Melissa Dark. 2006. Designing a Flexible, Multipurpose Remote Lab for the IT Curriculum. In *Proceedings of the 7th Conference on Information Technology Education* (Minneapolis, Minnesota, USA) *(SIGITE '06)*. Association for Computing Machinery, New York, NY, USA, 161–164. https://doi.org/10.1145/1168812.1168843

[8] Khaled Salah, Mohammad Hammoud, and Sherali Zeadally. 2015. Teaching cybersecurity using the cloud. *IEEE Transactions on Learning Technologies* 8, 4 (2015), 383–392.

[9] Dino Schweitzer and Jeff Boleng. 2009. Designing web labs for teaching security concepts. *Journal of Computing Sciences in Colleges* 25, 2 (2009), 39–45.

[10] Johannes Sianipar, Christian Willems, and C. Meinel. 2016. A Container-Based Virtual Laboratory for Internet Security e-Learning. *The international journal of learning* (2016).

[11] Johannes Sianipar, Christian Willems, and Christoph Meinel. 2017. Team Placement in Crowd-Resourcing Virtual Laboratory for IT Security e-Learning. In *Proceedings of the 2017 International Conference on Cloud and Big Data Computing* (London, United Kingdom) *(ICCBDC 2017)*. Association for Computing Machinery, New York, NY, USA, 60–66. https://doi.org/10.1145/3141128.3141146

[12] Muhammad Wannous and Hiroshi Nakano. 2010. NVLab, a Networking Virtual Web-Based Laboratory that Implements Virtualization and Virtual Network Computing Technologies. *IEEE Transactions on Learning Technologies* 3, 2 (2010), 129–138. https://doi.org/10.1109/TLT.2009.31