

Educating the Next Generation MSc in Cyber Security

• Anders Carlsson • Iryna Sokolianska • Alexander Adamov



ENGENSEC



2013
2017



ENGENSEC

Educating the Next generation
experts in Cyber Security



Co-funded by the
Tempus Programme
of the European Union

Educating the Next Generation MSc in Cyber Security

• Anders Carlsson • Iryna Sokolianska • Alexander Adamov



ENGENSEC



2013
2017



ENGENSEC

Educating the Next generation
experts in Cyber Security



Co-funded by the
Tempus Programme
of the European Union



ENGENSEC

Educating the Next generation
experts in Cyber Security

Acknowledgements

This work would not have been possible without the support of the steering group:

Iryna Sokolianska, Kharkiv, Ukraine; Alexander Adamov, Kharkiv, Ukraine; country coordinators prof. Evgenij Duravkin, Ukraine, Marina Borisova, Russian Federation, Tomasz Surmacz, Poland and DCI. Heiko Held, BKA, Germany.

I am also grateful to all the members in the course developer teams and others whom I have had the pleasure to work with and learning security issues and teaching skills during the project.

I am especially indebted to Professor Rune Gustavsson, BTH, Sweden, who has been my mentor and supportive.

I sincerely hope that this book will give you guidance or ideas to develop or improve a MSc in Cybersecurity.

*Dr. Anders Carlsson,
General Manager of ENGENSEC.EU
BTH, Sweden*

Content

Introduction	6
Partners	10
Dissemination Activities	18
Master’s Thesis Framework	20
Course Modules	
■ Course Module 1. Advanced Network and Cloud Security	46
■ Course Module 2. Wireless and Mobile Security	54
■ Course Module 3. Secure Software Development	66
■ Course Module 4. Malware Analysis	70
■ Course Module 5. Web Security	78
■ Course Module 6. Penetration Testing and Ethical Hacking . . .	88
■ Course Module 7. Digital Forensics	96
Cloud Based Remote Security Lab (ReSeLa)	102
Articles	
■ Overview of the Course in “Wireless and Mobile Security”	104
■ Secure Software Programming Problems	120
■ Education Challenges and Development of Advanced Malware Analysis Course	130

Development of Informatics Students' Competence in the Field of Information Security of Internet Resources	135
Teaching Penetration Testing: Focusing on Hacking Techniques or Methodology	144
Digital Forensics Cases Overview	155
The Cloud Threat Landscape	182
Validation of Results	187

Introduction

The EU Project ENGENSEC — Educating The Next Generation Experts in Cyber Security was realized in 2014–2017 (<http://engensec.eu>).

The objective was to create a new Master's Security as a response to cyber security threats, by educating the next generation experts.

This task was addressed in the following six ways:

- ✓ First, identify an education framework stressing engineering fundamentals.
- ✓ Second, identify a suitable Master's program framework.
- ✓ Third, identify relevant course modules to be developed and validated.
- ✓ Fourth, identify and implement a suitable flexible experimental environments.
- ✓ Fifth, validate the developed courses and methods.
- ✓ Sixth, dissemination and publications.

The education framework chosen is CDIO² — Conceive, Design, Implement, Operate. The developed course material, including labs, follows the CDIO model (<http://www.cdio.org>).

1

The new Master's framework.

In order to identify challenges in future information societies as well as analysis of the skills and knowledge demanded from future security experts a questionnaire was sent to 52 companies. The results of this investigation was identification of 7 new courses.

2

Development of these new course models.

- ✓ Advanced network & cloud security
- ✓ Wireless & mobile security
- ✓ Secure software development
- ✓ Malware Analysis
- ✓ Web security
- ✓ Penetration testing and ethical hacking
- ✓ Digital forensics

3

To facilitate the educational and teaching environment of ENGENSEC courses, the Cloud based Remote Security Lab (Resela) was designed, implemented and validated.

Validations of the results of ENGENSEC have been by:

- ✓ External peer-to-peer evaluations
- ✓ Train the trainer courses and seminars
- ✓ Summer schools for students

4

Those issues are addressed.

Dissemination of the project's results have been by:

- ✓ Setting up and maintaining a webpage
- ✓ Dissemination activities
- ✓ Scientific publication

5

These issues are presented.

ENGENSEC

Project Info

Project title: Educating the Next generation experts in Cyber Security: the new EU-recognized Master's program (ENGENSEC).

Project number:

544455-TEMPUS-1-2013-1-SE-TEMPUS-JPCR.

Duration: 01 Dec 2013 — 30 Nov 2017.

Target group: university students, teachers, IT specialized public, government, industry organizations.

Website: www.engensec.eu

Project objective

The main objective of the ENGENSEC project is to create new Master's program in area of IT Security as a response on current and emerging Cyber Security threats by educating the next generation experts in this area, thus supporting development of e-based economics, life and society in partner countries.

The ENGENSEC proposal addressed challenges in information society development concerned Cyber Security and privacy in e-oriented world. This project introduces advanced Master's curriculum in Cyber Security to teach the next generation experts, based on previous best practices in Double Degree implementation among EU universities, ECTS and mutual degree recognition.

Project Management

General manager, Coordinator for EU partners:

Anders Carlsson, Blekinge Institute of Technology (Karlskrona, Sweden), anders.carlsson@bth.se

Coordinator for UA partners:

Arkadii Snihurov, Kharkiv National University of Radio Electronics (Kharkiv, Ukraine), arkadii.snihurov@nure.ua

Coordinator for RU partners:

Marina Borisova, P.G. Demidov Yaroslavl State University (Yaroslavl, Russia), marinafon@yandex.ru

Quality assurance: Chairman is Prof. **Rune Gustavsson**, KTH Stockholm, rune.gustavsson@ics.kth.se

Project activities

1. Defining framework of joint Master's program.
2. Cloud-based security lab development.
3. Development of the joint course curriculum.
4. Develop new and further develop existing courses
5. Teacher training.
6. Effective quality control ensured and project management.
7. Dissemination of new Master's programs benefits
8. Give pilot courses in a summer school.
9. Prepare for participating Universities to launching new Master's program.

Expected results

1. Prepare well qualified professionals in emerging areas of Cyber Security by offering recognized Double degree Master's program.
2. Develop set courses on emerging areas of Cyber Security addressed to Digital Forensics, Ethical Hacking, Penetration Testing, Web Security and Malware Analysis.
3. Develop a cloud-based laboratory environment, which can be used remotely from partner's University.
4. Establish Life Long Learning framework by offering set of training courses for specialized public and government entities focused on computer crime and computer Investigation.
5. Harmonize Bachelor programs to unify courses objectives, curriculum content, skills and knowledge requirements.
6. Define structure of the new Master's program, create and adopt courses recognized by all consortium partners.
7. Enable student and academic staff mobility for exchanging of best practices to raise the level of international cooperation in area of Cyber Security.

Region managers

-  **Anders Carlsson** (*anders.carlsson@bth.se*) — Contractor, EU Coordinator, Coordinator of Blekinge Institute of Technology (Karlskrona/ Sweden).
-  **Arkadii Sniurov** (*arkadii.sniurov@nure.ua*) — UA Coordinator, Coordinator of Kharkiv National University of Radio Electronics (Kharkiv/ Ukraine).
-  **Marina Borisova** (*marinafon@yandex.ru*) — RU Coordinator, Coordinator of P.G. Demidov Yaroslavl State University (Yaroslavl/ Russia).



EU managers

- Anders Carlsson** (*anders.carlsson@bth.se*) — Contractor, EU Coordinator, Coordinator of Blekinge Institute of Technology (Karlskrona/ Sweden).
- Dariusz Caban** (*dariusz.caban@pwr.wroc.pl*) — Coordinator of Wroclaw University of Technology (Wroclaw/Poland).
- Leo Truksans** (*leo.truksans@lu.lv*) — Coordinator of University of Latvia (Riga/ Latvia).
- Spyros Kokolakis** (*sak@aegean.gr*) — Coordinator of University of the Aegean-Research Unit (MYTILENE/Greece).
- Nadja Golz** (*Nadja.golz@bka.bund.de*) — Coordinator of The German Federal Criminal Police Office (Wiesbaden/Germany).



UA managers

- Arkadii Sniurov** (*arkadii.sniurov@nure.ua*) — UA Coordinator, Coordinator of Kharkiv National University of Radio Electronics (Kharkiv).
- Sergey Toliupa** (*izi_duikt@ukr.net*) — Coordinator of State University of Information and Communication Technologies (Kyiv).
- Lukova-Chuiko Nataliya** (*lukova@ukr.net*) — Coordinator of State University of Telecommunication (Kyiv).
- Yuriy Lakh** (*julakh@polynet.lviv.ua*) — Coordinator of Lviv Polytechnic National University (Lviv).
- Kostiantyn Korsun** (*kkorsun@uisg.com.ua*) — Coordinator of Ukrainian Information Security Group (Kiev), Non-governmental Organization «Ukrainian Information Security Group» (Kyiv).
- Marharyta Schetkina** (*marharyta.schetkina@aiesec.net*) — Coordinator of AIESEC Ukraine (Kyiv).
- Olena Shershun** (*olena1.shershun@aiesec.net*) — coordinator of Ukrainian, Youth Non-governmental Organization AIESEC in Ukraine (Kyiv).
- Vasyl Satsyuk** (*tzi_kh@ukr.net*) — Coordinator of Kharkiv state center for information security (Kharkiv), Kharkiv State Regional Science and Research Center on Information Security (Kharkiv).
- Alla Rybalko** (*a_rybalko@mon.gov.ua*) — contact person at Ministry of Education and Science, Head of the sector of University Education within Department of Higher Education.

Introduction



RU managers

- **Marina Borisova** (*marinafon@yandex.ru*) – RU Coordinator, Coordinator of P.G. Demidov Yaroslavl State University (Yaroslavl/ Russia).
- **Viktor Petrikin** (*petrikin@klgtu.ru*) — Coordinator of Kaliningrad State Technical University (Kaliningrad).
 - **Andrey Krasov** (*krasov@pisem.net*) — Coordinator of Saint Petersburg State University of Telecommunications (Saint-Petersburg).
 - **Igor Kotenko** (*ivkote@comsec.spb.ru*) — Coordinator of Saint-Petersburg Institute for Information and Automation (Saint-Petersburg).
 - **Eduard Lysenko** (*tani@region.adm.yar.ru*) — Coordinator of Department of Informatization and communication of Yaroslavl (Yaroslavl).
 - **Alexander Dorofeev** (*adorofeev@uc-echelon.ru*) — Coordinator of Autonomous non-profit organization «Training center «Echelon» (Moscow).
 - **Farit Muzipov** (*muzipov@npsib.org*) — Coordinator of The partnership of experts of information security (Moscow).
 - **Alexander Levit** (*alexlove89@mail.ru*) — Coordinator of Yaroslavl regional public organization «Union of students» (Yaroslavl).



Project Management



BTH BLEKINGE
INSTITUTE OF
TECHNOLOGY



BLEKINGE INSTITUTE OF TECHNOLOGY (KARLSKRONA, SWEDEN)

Blekinge Institute of Technology (BTH) is a globally attractive knowledge community within applied IT and innovation for sustainable growth. BTH was founded in 1989 which means that we are a young institute who manage education and research in new ways, but still with good quality. At Blekinge Institute of Technology (BTH) education and research are of a very high international standard and learning is the focal point for students, teachers and researchers alike. BTH is a recognized university under government control. It is the most distinctly profiled higher education institution in Sweden, with a clear focus on applied information technology and innovation for sustainable growth.

BTH offers study programmes and single subject courses in first, second and third cycles. 7,200 students study at our five schools. Around 35 professors and more than 100 associate and assistant professors work on two campuses.



Wrocław University
of Science and Technology



WROCLAW UNIVERSITY OF SCIENCE AND TECHNOLOGY (WROCLAW, POLAND)

The university has been functioning under the current name since 1945. Since the beginning of its existence, it has been an important centre of technical education. It belongs to the best technical universities in Poland — nearly 34 000 students study under the guidance of 2 000 academic teachers, at the 12 faculties, as well as in the 3 regional branches. It takes high position in the annual rankings of Polish universities. Recently, it has been announced the best Polish technical university in Webometrics Ranking of World's Universities. The mission of the University is to shape the creative and critical personalities of students and define the directions of development in science and technology. The education is strongly linked with scientific research and the needs of economy and is consistent with standards of the European Higher Education Area.



Bundeskriminalamt



THE GERMAN FEDERAL CRIMINAL POLICE OFFICE (WIESBADEN, GERMANY)

BKA has been supporting and assigned partner for example of the ISEC 2009 that funded Cybercrime training project, led by Europol. KI32 led the ISEC 2010 project in Cybercrime training, where three courses were by BKA. KI32 is the Specialized Criminal Police Training unit of the federal and state police in Germany develops, organizes and disseminates trainings for the German Police.

KI32 took the lead of the development of the National Cybercrime training concept in Germany 2003–2006 and provides this training since 2007. Together with experts from Cybercrime forensics units they have developed a training for court experts. The training was disseminated in cooperation with the University of Mannheim. Annually since 2007, KI32 has been organizing Cybercrime trainings for European Law Enforcement with OLAF for 60 ~ 130 participants each.

PARTNERS



**LATVIJAS
UNIVERSITATE**
ANNO 1919

UNIVERSITY OF LATVIA (RIGA, LATVIA)

The UL Faculty of Computing in cooperation with the UL Institute of Mathematics and Computer Science provides higher academic as well as professional education and conducts research in the fields of computer science and information technologies. The studies are organized in 4 levels based on the pyramid principle:

1. The college level prepares programmers and computer network administrators by offering knowledge and skills that correspond to the standards of the first level professional higher education.
2. The bachelor's level requires mastering not only programming, but also designing and development of complex programme systems.
3. The master's level provides skills how to analyze and design large systems as well as manage projects.
4. The doctoral level prepares specialists of the highest qualification for the development of large and complex projects as well as for work in higher educational institutions.

Five education fields of computing are offered.



UNIVERSITY OF THE AEGEAN (MYTILENE, GREECE)

The University of the Aegean is one of the youngest Universities in Greece, and was established in 1984. It comprises 16 departments located in the islands of Samos, Lesbos, Chios, Rodos, Syros, and Limnos in the Aegean Archipelagos. It has a wide range of educational activities, including more than 25 MSc programmes, more than 20 summer schools and about 50 e-learning training programmes.

The Information & Communication Systems Security Lab (Info-Sec-Lab) includes 7 academic staff members and more than 12 research scientists and engineers. The members of the Info-Sec-Lab have been involved in several R&D projects on Security and Privacy, security investigations, audits, security conference committees, and security reviews. Most of the R&D projects have been funded by either European Commission programmes or the Greek government. The lab also supports an MSc programme on Information & Communication Systems Security.



**UNIVERSITY
OF THE AEGEAN**





STATE UNIVERSITY OF INFORMATION AND COMMUNICATION TECHNOLOGIES (KYIV, UKRAINE)

The University is a leading educational institution in Ukraine and is famous for its scientists in infocommunications and information security industry. During the training curriculum the teachers consult and coordinate their educational programmes with the leading telecommunications operators. A new field of knowledge such as infocommunications is proposed at the University. This master programme focuses on modern technologies: digital TV, the 4th generation of mobile networks, future networks and broadband access networks.

The graduates of university are highly qualified specialists in the areas of: telecommunication systems and networks, radio equipment and telecommunications, optoelectronic system, information security systems. The specialized academic councils work for these specialties.



LVIV POLYTECHNIC NATIONAL UNIVERSITY (LVIV, UKRAINE)

Research area of the Information Security Department is connected primarily with investigation and development of technical information security equipment, computer network security systems, new principles of construction and methodological design of equipment for immitance testing, with development of theoretical bases of analysis and design of pulse-numeric functional converters. Another research area — construction of computer network security system, using honeypot systems, is rather perspective in the sphere of computer systems and network protection, as honeypot-systems could be used as an effective instrument of network attacks indicating and analysis. The department is closely cooperating with the Department of Information Technology Security as well as with the Western regional educating-scientific information security center of the Lviv Polytechnic National University.



KHARKIV STATE CENTER FOR INFORMATION SECURITY (KHARKIV, UKRAINE)

Kharkov State Regional Scientific and Research Center on a Technical Protection of the Information was established in 1997. The center has special license from the State government authority on specialized communication and information protection which allows providing services to public, government and commercial enterprises in the field of working environment securing, information security and its protection. The center contributes to the government and authorities on state policy development in the area of information protection and security. The staff of the center is 70 % Ph.Ds. and researchers that allows to run numerous research and development activities in the areas of information security, secured systems design, electro-magnetic compatibility.



PARTNERS



AIESEC

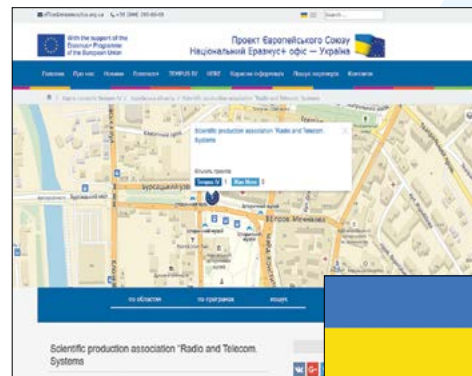
AIESEC UKRAINE (KYIV, UKRAINE)

AIESEC is an international youth organization that helps develop skills and personal qualities, get new experience by taking part in international programs and projects in the organization. AIESEC unites active young people from 113 countries. Members of the organization are students and graduates who develop personally and professionally towards forming themselves as leaders, and they want to know how to change the world for the better. We believe that developing the skills and personal qualities, we enable young people to influence the future of the world. AIESEC offers young people the opportunity to develop as leaders through a complex experience that is formed by three main elements: Experience in leadership; International Programs AIESEC; Global educational environment. Since 1994 AIESEC in Ukraine has brought together more than 1,000 students from 70 universities. Local Committees of organization work in 19 cities of Ukraine.



SCIENTIFIC PRODUCTION ASSOCIATION "RADIO AND TELECOM. SYSTEMS" (KHARKIV, UKRAINE)

Science and Research company Radio and Telecommunication Systems is a private company which works in the area of telecommunication technologies research and development, information security and protection. Among its primary activities are development, implementation and maintenance of information systems facilities, protection of the information which is transmitted through acoustic, electromagnetic fields with electrical signals; consulting services in areas of communication technologies and information security; development and production of complex technical information security systems, to protect information from unauthorized access. The company has several licenses from government authorities which permit appropriate activities and services in telecom fields.



MINISTRY OF EDUCATION AND SCIENCE OF UKRAINE (KYIV, UKRAINE)

Ministry of Education and Science of Ukraine is the main government institution in Ukraine, responsible for development and implementation of the national policy in education and science. The main objectives of the Ministry are determination of the prospects and priorities of education and science, innovation and information, intellectual property development in Ukraine. The Ministry ensures development of educational, research, innovation, information potential of Ukraine; integrates national education and science, information resources, intellectual property; carries out regulatory and legal framework; defines the strategy for monitoring of the education quality and controls its realization; provides licensing and accreditation of higher educational institutions.





UKRAINIAN INFORMATION SECURITY GROUP (KIEV, UKRAINE)

NGO UISG is a legal entity for Ukrainian information security community which includes ~ 1500 INFOSEC experts. The community members discuss actual problems/issues in the field of cyber security in groups NGO UISG and UISG in LinkedIn network. The founders of the Group believe that they can unite efforts of Ukrainian professionals:

- for further popularization and advancement of modern InfoSec ideas and practices;
- to become “symbol of trust” in the sphere of cyber threats mitigation within Ukraine as well as abroad.

NGO UISG is the organizer of annual national INFOSEC Conference UISGCON.



KHARKIV NATIONAL UNIVERSITY OF RADIO ELECTRONICS

(KHARKIV, UKRAINE)

Kharkiv National University of Radio Electronics (KhNURE) is one of the oldest schools in Ukraine. It's research and educational capabilities in electrical engineering, electronics, telecommunications, information and computer technologies have no equals in Ukraine and CIS. Today the University is a home for over 12 thousand students majoring in 34 specialties on in the 8 faculties. Teaching staff includes 135 professors and 340 post-graduate degree holders, 9 Honorary Scientists of Ukraine and 25 members of national and international Academies. The University has many specialized laboratories and centers, such as the center for distance learning, an electronic library, student television center, laboratory of satellite television and other laboratories (of the electronic technologies, information security, consumer electronics, microprocessor devices, bank technologies and others). KhNURE hosts National Association “Antenna” and Eastern-Ukrainian department of IEEE.



N U R E

Kharkiv National University
of Radio Electronics



DEPARTMENT OF INFORMATIZATION AND COMMUNICATION OF YAROSLAVL (YAROSLAVL, RUSSIA)

The Department of Informatization and Communication of Yaroslavl Region in the Russian Federation is the government authority of Yaroslavl Region. The Department develops and realizes state politics concerning regional informatization and communication, formation of electronic government, television and radio broadcasting in the territory of Yaroslavl region. The department organizes technical security of information and technical reconnaissance counterwork during information processes in the administration of Yaroslavl Region.



PARTNERS

P.G. DEMIDOV YAROSLAVL STATE UNIVERSITY

(YAROSLAVL, RUSSIA)

Yaroslavl State University (YSU) was founded in 1803. Today YSU is a leading higher educational institution of the Upper-Volga region with 7800 students. It is a classical multi-discipline educational institution training students in 70 majors (on bachelor's and master' programmes). The university primarily serves Yaroslavl Region but also has students from other parts of Russia and from abroad. As a state university YSU reports directly to the Federal Ministry of Education. The development of fundamental science was always the top-priority task for the university. YSU provides 43 bachelor programmes, 28 master programmes and 41 post-graduate programmes. The introduction of the new master programme in Cyber Security is based on students' interests in the following faculties: IT, Mathematics, Physics. The faculty of Mathematics has "Computer security" area of study.

The university is actively involved in the internalization of higher education. It has a number of long-term partnerships with higher educational institutions of the USA , EU, NIS countries.



SAINT-PETERSBURG STATE UNIVERSITY OF TELECOMMUNICATIONS

(SAINT-PETERSBURG, RUSSIA)

The Bonch-Bruевич Saint-Petersburg State University of Telecommunications holds a leading position not only among universities of communications, but also among other technical universities in Russia. The staff of our university are highly skilled experts in the field of telecommunications, broadcasting, information communication, multimedia, information systems, information security, advertising and television. The university provides 15 educational programmes to 10,000 students. There is a tiered system of continuous training at the university: high school — lyceum — college — university. Graduate education includes doctoral and post-doctoral study programmes, as well as in-service training programmes. The educational process is conducted in close cooperation with leading companies in the telecommunications sector.

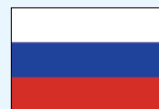


AUTONOMOUS NON-PROFIT ORGANIZATION «TRAINING CENTER «ECHELON»

(MOSCOW, RUSSIA)

The training center specializes in providing corporate and open trainings in information security. Our main courses are: protection of personal data; technical protection of confidential information; cryptography; ethical hacking; ISO 27001: introduction, implementation and internal audit; administering of Astra Linux Special Edition; certification of software; common criteria;

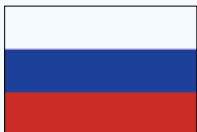
We have partner relationships with Bauman Moscow State Technical University and National Research Nuclear University «MEPhI». The training center is a member of ACP programme of British Standards Institution.



SAINT-PETERSBURG INSTITUTE FOR INFORMATION AND AUTOMATION OF THE RUSSIAN ACADEMY OF SCIENCE

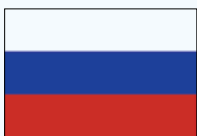
(SAINT-PETERSBURG, RUSSIA)

The information security and computer network protection are the key research areas of SPIIRAS. It is these problems that are investigated within international and federal projects by a set of laboratories such as the laboratory of computer security problems, the laboratory of cryptology and others. For example, the laboratory of computer security problems currently conducts research in security information and event management (EU Project MASSIF), embedded system security (EU Project SecFutur), development of methods for integrated network protection against network attacks based on hybrid multi-agent modeling and simulation of computer counteraction, and design of the intelligent security services in critical infrastructures (the project of Russian Ministry of Education and Science), etc. The institute employees share their knowledge in information security with students when lecture specialized courses at the basic departments affiliated to the leading St. Petersburg universities.



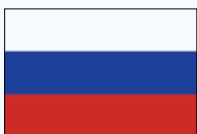
NGO «THE PARTNERSHIP OF EXPERTS OF INFORMATION SECURITY» (MOSCOW, RUSSIA)

Noncommercial Partnership «The partnership of experts of information security» was registered on September 7, 2009. Today it is the community of experts, who have professional point of view on different problematic issues in the Cyber Security sphere. Moreover it is the club for communication and experience sharing between specialists in the information security. The activity of Noncommercial Partnership members is focused on the investigation and practice of the up-to-date approaches and technologies in the cryptographic protection area and information security. The members of Noncommercial Partnership regularly take part in Cyber Security conferences. These conferences are held not only in regional, but also in federal and international levels. The Noncommercial Partnership members are oriented to up-to-date decisions in the Cyber Security sphere.



YAROSLAVL REGIONAL PUBLIC ORGANIZATION «UNION OF STUDENTS» (YAROSLAVL, RUSSIA)

Yaroslavl regional public organization «Union of students» works with 9 universities (including branches) of Yaroslavl and Yaroslavl region. The members of «Union of students» are 1000 students. The main goal of the organization is to create favourable conditions for students' self-actualization. The directions of activities of «Union of students» are patriotic education, development of a volunteer base, student self-actualization in creative work, cross-cultural communication development, staff assistance, works during 17 years. Organization got "best student and youth NGO" status in Yaroslavl region many times. There are about 500 social awards.



PARTNERS

KALININGRAD STATE TECHNICAL UNIVERSITY (KALININGRAD, RUSSIA)

Kaliningrad State Technical University (KSTU) is the oldest higher education institution in Russia and the only in the Region that gives education in almost all of the industrial and economic brunches. The University offers education at various levels, namely, Bachelor, Professional Diploma, Master, Post-graduate and Doctorate, as well as professional development for specialists. Since 2011 in compliance with Bologna process KSTU has developed a new multi-level training system — Bachelor — Master — PhD — Doctorate Degrees. KSTU together with other universities train professionals in Information Technology and Computing Machinery, Complexes, Systems and Networks, Data Processing and Control Systems, and offer interdisciplinary course in Applied Information Technology in Economics.



In recent years the Bachelor's Programme in Information Security was introduced to the training curricula. The next step in this area is the development of Master Degree modules in Computers Security.

7
countries

Ukraine

Greece

Germany

Partners

Sweden

Poland

Latvia

Russia

Academic —
universities

Governmental —
authorities



Students —
association of students

22
organizations



Industrial —
noncommercial organizations
in the information security area

REPRESENTATION OF THE PROJECT ON INTERNET

- Information about the project, opportunities, reports of working groups and development teams are available on the website <http://engensec.eu>
- Community of the project in social networks <https://www.facebook.com/engensec/>
- On-line date portal was developed and used during the project <http://owncloud.engensec.eu>

1

SUMMER SCHOOLS — 3

- 28 June — 8 July 2016 in **Lviv National Polytechnic University**, 18 — 31 July, 2016 in **Saint Petersburg State University of Telecommunications**, 17–28 July 2017 in **Lviv Polytechnic National University**.
- As a result, we got feedback from the students regarding the quality and content of developed courses.
- Student skill competition took place at the end of each Summer School.

2

INTERNATIONAL CONFERENCES — 20

- The participants of the project presented their reports at the international conferences in Estonia, the Czech Republic, Ukraine, Russia and Sweden on cyber security issues.
- The report of the technology which RESELA is based on was presented OpenStack Summit. Barcelona, Spain.

3

ENGENSEC IT SECURITY ACADEMY — 5

- The main idea of open lectures and workshops is to share IT security experience with everybody who really wants to get it! The entrance was free.

4

ARTICLES IN MEDIA ABOUT THE PROJECT — 17

- Information about some results of the project was published in the print media of the partner universities of the project and in other publications, such as specialized national journals and newspapers on information security issues.

5

DISSEMIN ACTIV

ACTIVITIES



ENGENSEC

Educating the Next generation
experts in Cyber Security

ENGENSEC book



A book with a description of the project and its results, a brief presentation of each course and a virtual laboratory.

RESELA WORKSHOPS — 6

- The development team held working meetings, as well as training of technical specialists and a presentation of a remote laboratory for partners

6

DISSEMINATION VISITS — 10

- Presentational visits with reports on the project, its activities and possible benefits were conducted for academic, business and military and financial organizations

7

ABOUT THE PROJECT ON THE SITES OF PARTNERS — 12

- Partner organizations through their websites inform about the progress of the project and its capabilities for potential participants and those who are interested in improving cyber security skills

8

THE MEETINGS AND WORKSHOPS — 19

- About 3-4 working meetings and workshops inside the teams and also for partners were conducted during the project for each of 7 courses.

9

TEACHER TRAININGS — 8

- The training on the topic the organization of remote teams, the communication in international teams were conducted for course leader.
- A series of webinars on the topics of visualizing educational information and using of interactive methods of teaching was also held.

10

Master's Thesis Framework

Guidelines for Degree Projects

Modification of the ENGENSEC project by:

Anders Carlsson Department of Computer Science & Engineering, BTH

The document was originally created by:

Jürgen Börstler, Department of Software Engineering, BTH

Niklas Lavesson, Department of Computer Science & Engineering, BTH

Veronica Sundstedt, Department of Creative Technologies, BTH

PREFACE

This document summarizes information regarding the degree project courses at the Faculty of Computing at Blekinge Institute of Technology. The document covers degree projects at the Bachelor's, one-year Master's, and two-year Master's level in computer science, digital game development, telecommunications systems, and software engineering as well as the degree projects in the Master of Science in Engineering (MSE¹) programmes.

These guidelines are intended as a complement to the reference literature and the introductory lectures.

The main part of this document concerns assessment. Assessment rubrics are described in detail in the appendices. This should help students in project development and documentation, as well as supervisors, reviewers and examiners in performing a uniform assessment of degree projects. Assessment and grading must always be based on the explicit learning outcomes defined in the most recently approved syllabus ("course plan") for the degree project course in question. Our ambition has been to develop assessment procedures and templates that capture these learning outcomes.

The generic degree project process and the corresponding templates, documents, instructions, and other items have been developed and improved by several individuals in recent years. Above all, we would like to thank Professor Robert Feldt who initiated and developed the first versions of the process and many of the supporting documents. Additionally, we would like to thank our colleagues at the Faculty of Computing who have contributed with new material and contents, insightful ideas, and suggestions for improvement.

CHAPTER 1. INTRODUCTION

The degree project is conducted at the end of the education programs that lead to Bachelor's, Master's, and Master of Science in Engineering (MSE) degrees. The purpose of the project for a student is to practice the necessary skills used to independently define, plan, conduct, and present a project. The projects focus on research, on development, or contain a blend of both. At Blekinge Institute of Technology (BTH), the Master's projects should have a clear research focus while the Bachelor's and MSE projects can be focused

¹ MSE = Swedish "Civilingenjör", a five-year Engineering degree that should not be confused with "Civil Engineering". Please note that the MSE programmes received a separate, slightly different process and own templates in 2016.

on development. Regardless of the type of degree, it is important to note that all projects must have sound scientific basis.

A typical degree project of 30 ECTS corresponds to 20 weeks of full-time work, i.e. 800 hours per person. Each ECTS corresponds to roughly 26.7 hours of work. It is important to start working hard and continuously on the project from the very beginning, since it will be very difficult to compensate for delays.

During the planning, execution, and presentation of the degree project, the student is expected to apply the knowledge and skills acquired during his or her education to independently and deeply explore one or more areas of the main subject. At the end, the student will present, orally and in writing, the results of the degree project. In addition, the student will review and critically appraise another student's thesis.

These guidelines are divided into four brief chapters (besides Chapter 1. Introduction) and two appendices. Chapter 2. Critical thinking and reasoning provides a short introduction to the reasoning that is expected during development and documentation of a degree project. Chapter 3. Academic writing gives a very brief introduction to the key issues in academic writing. Chapter 4. Presenting and defending the thesis and Chapter 5. Critical appraisal of another thesis (opposition) provide some details about thesis presentations and oppositions, respectively.

The appendices describe in detail the rules, instructions, and templates of the degree project courses. The first appendix contains all assessment rubrics. These templates are mainly used by the examiner during the assessment and grading of various examination parts. Two of these templates are also used by the supervisor and the appointed independent reviewers to assess and recommend grades for the project plan and the final version of the thesis. The second appendix contains instructions for assessing the various aspects that constitute each ground for judgment in the templates. The third appendix contains instructions concerning the mandatory document templates for the project plan and the thesis.

CHAPTER 2. CRITICAL THINKING AND REASONING

Carrying out a degree project requires continuous decision making. Which topic is worthwhile investigating? Which research questions are relevant? Which methods are appropriate to answer them? Etc.

The reasoning behind these questions should be as objective as possible and based on the best information available. Our thinking should be critical and not biased by a lack of information or our personal beliefs and prejudices.

The Foundation for Critical Thinking [7] provides guidelines for critical thinking and reasoning along the eight elements of thought: purpose, question/problem, assumptions, point of view, evidence, theories/concepts, interpretations/conclusions, and implications/consequences.¹

All reasoning has a *purpose* or a goal. To reason well, your purpose must be clearly stated, reasonable and fair. "Fair" means that other points of view should be considered unbiased.

All reasoning addresses a specific *problem*, a question, or an issue. To solve a problem, it must be solvable. It is therefore important to state research questions clearly and precisely and to understand what is needed to answer them.

All reasoning starts from underlying *assumptions*. It is important to understand how these assumptions affect your reasoning, whether or how they can be justified and how alternative assumptions might change your point of view.

¹ The following paragraphs are based on various sources available from the Foundation for Critical Thinking, see <http://www.criticalthinking.org> 3

All reasoning is done from some *point of view*. Different stakeholders might have different points of view. It is important to acknowledge other viewpoints and to discuss them fairly.

All reasoning is based on *evidence*. Your reasoning can only be as sound as the data it is based on. It is therefore important to gather the right type of data and make sure that it is relevant for the type of reasoning you have in mind.

All reasoning is shaped by the *theories*, models and concepts that are used to interpret the evidence. It is therefore important to state and explain these theories, models and concepts clearly.

All reasoning makes interpretations and draws *conclusions* to make sense of the gathered evidence with respect to the theories, models and concepts used. Conclusion should be strictly based on the available evidence and it should be made clear how underlying assumptions might affect these conclusions.

All reasoning has implications and *consequences*. These can be scientific, technical, social, environmental and ethical. It is important to show awareness of positive as well as of negative implications and consequences.

CHAPTER 3. ACADEMIC WRITING

The purpose of academic writing is to produce a text that communicates knowledge in an unambiguous way. It is the responsibility of a writer to convey that knowledge clearly, in a readable and understandable way. Academic writing should be economic, objective and accurate.

The process of academic writing helps a writer to deepen his or her understanding of a topic. Putting thoughts and ideas on paper helps to get a clearer idea about a topic. Without a clear idea about a topic, it is difficult to describe it clearly. It is also difficult to decide what is important and what is not. Without knowing what is important, it is difficult to decide what to include and what not. It is therefore strongly recommended to start writing early and to edit, reedit, and restructure the text continuously.

Please note that editing comprises more than just proofreading for spelling and grammar issues. A thesis should have a clear line of reasoning that leads a reader through the text. Sections and subsections should be in an intuitive and logical order and use expressive headings that support readability and understandability.

A good thesis should be clear, concise, coherent and cohesive. Think about what you want to tell your reader and tell it as clearly and concisely as possible. Try to avoid long-winded sentences that are difficult to read and understand. All content must be relevant. The content must also fit well together. Ideas that are presented together should also belong together, i.e. your ideas should be connected and related to each other (→ coherence). These connections on idea level should be supported by connections on actual text level. The sentences, paragraphs and sections should create a logical flow of those ideas (→ cohesion).

Please refer to the course textbook *Writing for Computer Science* [16] for more information on academic writing.

You can find further useful resources regarding academic writing on the Student Portal at https://studentportal.bth.se/web/studentportal.nsf/web.xsp/faculty_of_computing.

CHAPTER 4. PRESENTING AND DEFENDING THE THESIS

Each student will present his or her degree project in person at BTH. Presentations will be scheduled at three occasions each year¹ and an approved project plan and an approved thesis draft is required to be allowed to present.

¹ Presentation dates are usually in the end of January, early June and the end of September. The exact dates are available at https://studentportal.bth.se/web/studentportal.nsf/web.xsp/faculty_of_computing.

The total scheduled time for a 30 ECTS (15 ECTS) thesis is 1 hour (45 minutes), which is split as described below. Presenters (and opponents) are supposed to stay within the allocated time. Too short or too long a presentation (or opposition) might lead to a failing grade.

1. The student presents his or her work (20 minutes for Bachelor theses and 25–30 minutes for Master theses).
2. The opponent comments the thesis and raises questions and issues (5–10 minutes).
3. The examiner comments the thesis and raises questions and issues.
4. The supervisor comments the thesis and raises questions and issues.
5. The audience comments the thesis and raises questions and issues.

The main goal of a degree project presentation is to explain in detail what you have achieved and what we (the research community) can learn from your work and/or experience. The presentation should be on a suitable academic level for the target audience. Since your audience are fellow students and faculty, you do not need to describe everything from scratch. But you should not try to “show off”. A presentation that most people do not understand will not earn you a good grade.

It can feel quite challenging to present a degree project in 30 minutes or less, it is therefore very important to focus a presentation on a clear message. *What are the main points you want to get across?* Try to focus on the main points. You cannot cover everything. It is as important to figure out what you need to talk about then what you can ignore.

Start out by telling the audience *who* you are, *what* you are going to talk about and *why* this topic is interesting/relevant. Try to summarize what you have covered every now and then to keep your audience on track. Since people will not remember much from your presentation, make sure to end your presentation with your main contribution. A clear “take away message” or some other unusual element (a live demo, an interesting video, or a creative diagram) might also be helpful to make your talk memorable.

Besides actual timing and contents, it is also important to actually present well and use visuals that support your presentation. You should face the audience, talk clearly and loudly, and in a reasonable speed. Talking too fast will exhaust the audience, talking too slow and as a result the audience might get bored. Talking faster, you might be able to cover more material, but the audience will not be able to follow you and probably not remember much from your presentation.

Make sure to rehearse your presentation, preferably with an audience. A script may help you to remember what you want to say, but *don't read your visuals and don't learn and present your presentation by heart*. The audience expects to be spoken to and not read to; the latter feels artificial and will not be appreciated.

During the defense, you will also be asked questions that go beyond your actual degree project. It is therefore important that you know the area well. Of course, you cannot know everything but you should at least be able to discuss a topic in a professional and knowledgeable way. To support your audience, it might be a good idea to number your slides for easier reference.

¹ Time distribution might vary a little from course to course. Dates for your particular course will be available on the course homepage.

CHAPTER 5. CRITICAL APPRAISAL OF ANOTHER THESIS (OPPOSITION)

Each student will critically appraise another thesis. This critical appraisal (opposition) should take 5–10 minutes. Too short, too long, or too trivial an opposition might lead to a failing grade.

The purpose of the opposition is threefold. First, you (the opponent) should demonstrate that you can objectively and critically evaluate a substantial piece of academic work. Second, you should demonstrate that you can put a piece of scientific work into context and ask meaningful questions about its content. Third, the opponent's comments and questions should help to assess whether the presenter can defend his or her work reasonably well.

An opposition should be constructive, i.e. not simply criticizing the work but also suggesting the ways for improvement. That means that you might need to read up on the topic of the thesis you are assigned to for opposition. In the oral opposition, you should try to engage the presenter(s) in a discussion about the work. Questions that can simply be answered by “yes” or “no” should therefore be kept to a minimum.

After the opposition, the opponent shall send his or her opposition report directly and timely to the presenter(s).

APPENDIX A. ASSESSMENT RUBRICS

The assessment rubrics are used by different constellations of examiners, supervisors, and appointed independent reviewers to assess project plans, theses, presentations, defenses, written and oral oppositions. It is important to remark that the supervisors and independent reviewers are not actually grading any examination part. Their assessments only serve as an input to the examiner. The examiner is finally and formally responsible for assessing and grading the different examination parts.

All examination parts are assessed using the 2-level or 3-level scale, except the thesis for which the 7-level scale is used. The difference between the 2-level and the 3-level scale is that the latter includes the assessment *Fail — Minor complementation required*. The grade of an examination part is based on a summarized assessment of several aspects of a number of assessment criteria, unless an alternative approach is explicitly stated. Table A.1 describes the various assessment scales. The assessment criteria and aspects for all examination parts are summarized in Table A.2.

To achieve a passing grade, all assessment criteria must receive a value of at least 1. If any of the assessment criteria receives value “–” (or 0), the thesis as a whole will receive a corresponding failing grade, independently of how well the other criteria are evaluated.

Assessment	2-level scale	3-level scale	7-level scale	Value
Fail — Substantial further work required	U	U	F	–
Fail — Minor complementation required	U	UX	FX	0
Acceptable	G	G	E	1
Satisfactory	G	G	D	2
Good	G	G	C	3
Very good	G	G	B	4
Excellent	G	G	A	5

☞ **Table A.1.** Assessment scales and values.

Assessment Criterion	Project Plan	Thesis	Pres. & Defense	Written Oppos.	Oral Oppos.
Process		X			
Contents	X	X		X	
Contribution		X			
Presentation	X	X	X	X	X
Overall impression		X			

 **Table A.2.** Overview of assessment criteria for all examination parts.

If the grade is *Fail — Substantial further work required*, the thesis needs to be resubmitted at a later regular submission deadline. If the grade is *Fail — Minor complementation required*, the student is given a time-limited opportunity (usually about 4 weeks¹) to complement the examined part in question. After complementation, the thesis will be either graded as Fail (U/F) or as Pass (G/E), grades A–D are not possible. The thesis will be awarded the grade Fail (U/F) if:

- the thesis has not been submitted to the examiner within the designated deadline after an oral presentation and defense;

or if

- the examiner determines that the complementation was insufficient to fulfill the requirements for a Pass grade (for example, because some assessment aspect still does not reach a value ≥ 1). Each student has five opportunities to be examined to achieve a passing grade on any of the examination parts of the degree project course. After two such opportunities, a student is entitled to request a change of examiner. Such a request must be made in a written form to the university's vice chancellor. After five failing grades in any of the examination parts, a student can no longer register or re-register for the same degree project course. When all assessment aspects have received a value ≥ 1 , a weighted mean is computed to determine the final grade. The computation is based on the values of all assessed aspects for all assessment criteria. The weighted mean is then rounded off to the nearest integer². This integer is then translated to the corresponding grade according to the 7-level scale. The weighting of assessment criteria differs between degree projects of different types (see Table A.3).

A.1. Assessment rubrics for project plans

Through the project plan, the student should demonstrate broad knowledge and a deepened understanding of methodologies relevant for the subject. The project plan should describe a clearly defined and delimited degree project with the purpose to creatively, critically, and independently answer one or more identified and formulated research questions within at least one specific area of the subject.

¹ The time given for complementation might differ from course to course. 2As in the Swedish rounding rules; http://en.wikipedia.org/wiki/Swedish_rounding.

² As in the Swedish rounding rules; http://en.wikipedia.org/wiki/Swedish_rounding.

Assessment Criterion	Bachelor	1-year Master	2-year Master	MSE
Process	0.20	0.20	0.10	0.20
Contents	0.40	0.40	0.40	0.40
Contribution	0.00	0.10	0.25	0.20
Presentation	0.40	0.30	0.25	0.20

☞ **Table A.3.** Weighting of criteria for various types of theses.

Criterion	Aspect	Value
Contents	Background analysis	
	Problem formulation	
	Method selection and application	
	Knowledge and understanding	
	Critical thinking and scientific character	
	Planning	
Presentation	Disposition and layout	
	Adherence to academic standards	
	Coherence and cohesion	
	Language	

☞ **Table A.4.** Assessment rubrics for project plans.

The project plan should contain a critical review of relevant scientific literature and it should clarify the student's view and understanding of the state-of-the-art and the state-of-practice in the selected area of study. The project plan must identify any potential risk related to the selected approach and how they can be mitigated. The project plan should also describe specific needs of knowledge gathering necessary for completing the degree project. The student must define a clear and adequate time and activity plan and describe suitable approaches to answer the formulated research questions within the given constraints.

The assessment rubrics for project plans are presented in Table A.4. Since the project plan merely describes *proposed work*, the assessment of several aspects is based on the expected outcomes, assuming that the student pursues the project according to the project plan.

A.2. Assessment rubrics for theses

Through the thesis, the student should demonstrate broad knowledge and a deepened understanding of methodology in the subject. The thesis shall contain the description of a clearly defined and delimited degree project for which the purpose was to creatively, critically, and independently answer one or more identified and formulated research questions within at least one specific area of the subject. The student should, through the thesis, demonstrate significantly deepened knowledge in the selected area of study, which comprises deepened insights of the state-of-the-art and the state-of-practice¹. The thesis must contain a critical review of related scientific literature within the selected area of study. Furthermore, the thesis must clearly and in detail describe and motivate a suitable approach to answering the formulated research question(s). The con-

¹ For a Bachelor-level thesis, the requirements are not as strong as formulated here. Please consult the current course descriptor for your particular course for details.

ducted degree project must contribute to knowledge development in the subject area³. The thesis should contain a clear description of findings as well as an insightful and scientifically based analysis and synthesis of these findings. The contribution should be discussed in relation to the relevant scientific, engineering, societal, and ethical aspects. The thesis must adhere to the standards and requirements of scientifically grounded work.

Criterion	Aspect	Value
Process	Independence, initiative and creativity	
	Openness to critique and supervision	
	Planning and execution	
Contents	Background analysis	
	Problem formulation	
	Method selection and application	
	Knowledge and understanding	
Contribution	Critical thinking and scientific character	
	Overall contribution	
	Traceability and validity	
	Synthesis	
Presentation	Disposition and layout	
	Adherence to academic standards	
	Coherence and cohesion	
	Language	
Overall impression*		

☞ **Table A.5.** Assessment rubrics for theses. The first two aspects of the Process criterion are only assessed by supervisors.

* Overall impression represents a quantitative assessment of the overall impression of the thesis. It can be used to communicate strengths or weaknesses that are not captured well by existing criteria/aspects.

The assessment rubrics for theses are presented in Table A.5. The aspects for “Contents” and “Presentation” are almost identical for project plans. The meaning of some of the aspects, however, differ between the two artefacts since project plans are assessed from the perspective of expected outcomes instead of actual outcomes.

A.2.1. Example

The examiner has weighted together the reviews from the supervisor and independent reviewers with his/her own impression with the following overall results for the criteria Process, Contents, Contribution, and Presentation:

$Process = \{1, 2, 1\}$
 $Contents = \{2, 4, 1, 2, 3\}$
 $Contribution = \{2, 3, 2\}$
 $Presentation = \{4, 3, 2, 1\}$

According to the weighting scheme for a 2-year Master thesis project (see Table A.3), the grade is computed as follows:

$$0.1\left(\frac{1+2+1}{3}\right) + 0.4\left(\frac{2+4+1+2+3}{5}\right) + 0.25\left(\frac{2+3+2}{5}\right) + 0.25\left(\frac{4+3+2+1}{4}\right) = 2.398$$

The mean is rounded according to the Swedish rounding rules and the resulting value, 2, corresponds to grade D. If there was reasonable consensus among the supervisor and independent reviewers and the examiner's own impression is not much different from that, the examiner will likely award grade D. In case of large differences between assessments, the examiner discusses the assessments with the supervisor and independent reviewers to agree on a fair grade.

A.3. Assessment rubrics for oral presentations and defenses

During the presentation and defense of the degree project, the student should demonstrate the necessary skills to communicate orally and describe clearly his or her conclusions and the basis of these conclusions in a dialogue with the examiner, senior lecturers from the subject area, students from the subject area who are on the same level, and lay people. Table A.6 describes the criteria for judgment and their aspects for oral presentations and defenses. Only the examiner assesses the oral presentations and defenses.

Criterion	Aspect	Value
Presentation	Oral description of problem and method	
	Oral description of analysis and contribution	
	Overall communication skills	
	Argumentation and discussion abilities	

 **Table A.6.** Assessment rubrics for oral presentations and defenses.

A.4. Assessment rubrics for the written opposition

Through the opposition report, the student should demonstrate his or her ability to provide constructive criticism and feedback and communicate his or her conclusions after reading another student's thesis. The student should review the basis for the provided criticism and conclusions. The constructive criticism, the conclusions and the arguments behind them, must be summarized in a report to be read by the examiner as well as the author of the reviewed thesis. The opposition report must include the results of a systematic and critical review in which sufficient consideration has been put on all relevant scientific, engineering, societal, and ethical aspects of the thesis' contribution.

Table A.7 describes the criteria for judgment and their aspects for the written opposition report. Only the examiner assesses the written opposition reports. The opponent is expected to first describe the results of the critical review and then a list of clear and concise suggestions for necessary and recommended revisions. The necessary changes are those changes that must be performed in order for the opponent to consider a Pass grade for the thesis. The recommended changes are those changes that should be performed in order to increase the quality (and grade) of the thesis.

Criterion	Aspect	Value
Contents	Description of critical review	
	Description of necessary revisions and recommended revisions	
Presentation	Analysis and argumentation	
	Language	

 **Table A.7.** Assessment rubrics for the written opposition.

A.5. Assessment rubrics for the oral opposition

Through the oral opposition, the student should demonstrate his or her abilities to provide constructive criticism and discuss his or her conclusions about the other student's thesis based on the conducted review. The opponent should review the arguments behind the criticism and the conclusions. The constructive criticism, the conclusions and the arguments behind them should be discussed in a dialogue with the author of the thesis on a level that makes it possible for the participating students, lay people, and researchers to appreciate and understand the core of the discussion. The provided criticism must be based on a systematic and critical review of the thesis in question in which consideration has been put on all relevant scientific, engineering, societal, and ethical aspects of the thesis' contribution.

Table A.8 describes the criteria for judgment and their aspects for the oral opposition. Only the examiner assesses and grades the oral opposition. The opponent is expected first to present balanced and constructive criticism (the aspect Overall communication skills) and then lead a constructive and critical discussion (the aspect Analysis and argumentation skills) during which the respondent must defend his or her thesis.

Criterion	Aspect	Value
Presentation	Overall communication skills	
	Analysis and argumentation	

 **Table A.8.** Assessment rubrics for the oral opposition.

APPENDIX B. EVALUATION AND WEIGHTING OF ASPECTS

All degree project courses share the same assessment criteria (see Table A.2) but their weighting differs depending on the type of degree project. Table A.3 describes the relative weights of the assessment criteria for Bachelor's, One-year Master's, Two-year Master's, and Master of Science of Engineering (MSE¹) degree projects.

¹ MSE = Swedish "Civilingenjör", a five-year Engineering degree that should not be confused with "Civil Engineering".

The project plan and the thesis are assessed by the supervisor, the examiner, and the appointed independent reviewers. Since the course grade is based only on the assessment of the thesis, the weighted mean of aspect values is computed only for this examination part. The weighting of aspects is determined by the criterion weight as exemplified in detail in Section A.2.1.

In the following subsections, we provide textual descriptions for the assessment levels of all aspects for all assessment criteria. These textual descriptions should be used as guides ("rubrics") for the assessment. Please note that there is only one textual description for Fail, but that there are two assessment levels for Fail, as described in Table A.1. Please do also note that although we do not provide textual descriptions for levels 2 (satisfactory) and 4 (very good), these levels are nevertheless used for assessment.

Assessment level Fail — Minor complementation required (Value "0") indicates that the level of performance is close to level Acceptable, i.e. when only a little extra-work is necessary to reach level Acceptable. In all other cases, i.e. when all or most of the conditions described in row Fail are true, the assessment result should be Fail — Substantial further work is required (Value "–").

B.1. Evaluation of aspects for project plans and theses

Since the rubrics for project plans and theses are very similar, they are described together here. It is indicated where the criteria or aspects differ.

B.1.1. Process (theses only)

Independence, initiative and creativity	
Fail (–, 0) ²	Lets the supervisor be the main driver. Does not provide own suggestions for planning, execution, contents, or focus on the project. Insufficient or defective status/progress reporting.
Acceptable (1)	Takes some initiative in driving the project. Initiates some contacts with the supervisor. Proposes few and simple or initial own ideas. Status/progress reporting provides only limited information.
Satisfactory (2)	...
Good (3)	The main driver of the project. Keeps contact with the supervisor, based on agreed terms. Proposes and develops own ideas, which are creative to some degree. Regular and informative status/progress reports.
Very good (4)	...
Excellent (5)	Drives the project independently. Keeps the supervisor informed. Creates, develops, and realizes own creative ideas that fit the context well.

² Remember: Value "0" indicates that only a little extra-work is required to reach level Acceptable. When all or most of the conditions described for Fail are true, the assessment result should be "–".

Openness to critique and supervision	
Fail (–, 0)	Inability to develop/revise method(s) or content(s) despite of recurring constructive feedback. Little personal or content-related development.
Acceptable (1)	Acceptable progress regarding problem solving, creativity and academic writing based on supervision. Mistakes and defects are corrected based on direct feedback. Little extrapolation of comments/feedback to other parts of his/her work.
Satisfactory (2)	...
Good (3)	Responsive and adaptive for supervision. Develops ability to self-identify mistakes, errors, and improvement opportunities. Good extrapolation of feedback to other parts of his/her work.
Very good (4)	...
Excellent (5)	Excellent ability to request and utilize appropriate forms of supervision. Well-developed ability to examine his/her own or others' work critically. Excellent extrapolation and adaptation of feedback to the rest of his/her work.

Planning and execution	
Fail (–, 0)	Unrealistic planning that lacks a meaningful risk analysis. Irregular work of insufficient quality. Little progress.
Acceptable (1)	Acceptable planning that considers some relevant risks. Regular work of acceptable quality. Acceptable progress. Little to no status tracking and re-planning.
Satisfactory (2)	...
Good (3)	Good planning with basic risk management (i.e. risk identification, impact analysis, and mitigation). Systematic execution of plan. Steady progress. Sufficient status tracking and re-planning.
Very good (4)	...
Excellent (5)	Excellent planning and very good risk management. Systematic execution of plan. Steady progress. Excellent status tracking. Regular reconsideration/revision of plans.

B.1.2. Content (project plans and theses)

Problem formulation	
Fail (–, 0)	Goals, aims and objectives are poorly related to the background analysis. Scope is unclear, much too ambitious or much too broad for a thesis project. The actual research questions are irrelevant, poorly formulated or poorly related to the goals, aims and objectives. Expected results/outcomes are unclear or irrelevant.
Acceptable (1)	Goals, aims and objectives are sufficiently related to the background analysis. Scope is acceptable for a thesis project. Some research questions are relevant and related to the goals, aims and objectives. Indicates expected results/outcomes.
Satisfactory (2)	...
Good (3)	Goals, aims and objectives are clearly derived from the background analysis. Scope fits a thesis project well. Research questions are formulated well. They are relevant and clearly related to the stated goals, aims and objectives. Expected results/outcomes are relevant. Indicates stakeholders who could benefit from the results/outcomes.
Very good (4)	...
Excellent (5)	Goals, aims and objectives are clearly derived from the background analysis. Scope is very clearly stated and very ambitious (but not over-ambitious) for a thesis project. Research questions are formulated excellently and creatively. They are relevant, related clearly and explicitly to the stated goals, aims and objectives. Describes how various stakeholders would benefit from the expected results/outcomes.

Problem formulation	
Fail (–, 0)	Goals, aims and objectives are poorly related to the background analysis. Scope is unclear, much too ambitious or much too broad for a thesis project. The actual research questions are irrelevant, poorly formulated or poorly related to the goals, aims and objectives. Expected results/outcomes are unclear or irrelevant.
Acceptable (1)	Goals, aims and objectives are sufficiently related to the background analysis. Scope is acceptable for a thesis project. Some research questions are relevant and related to the goals, aims and objectives. Indicates expected results/outcomes.
Satisfactory (2)	...
Good (3)	Goals, aims and objectives are clearly derived from the background analysis. Scope fits a thesis project well. Research questions are formulated well. They are relevant and clearly related to the stated goals, aims and objectives. Expected results/outcomes are relevant. Indicates stakeholders who could benefit from the results/outcomes.
Very good (4)	...
Excellent (5)	Goals, aims and objectives are clearly derived from the background analysis. Scope is very clearly stated and very ambitious (but not over-ambitious) for a thesis project. Research questions are formulated excellently and creatively. They are relevant, related clearly and explicitly to the stated goals, aims and objectives. Describes how various stakeholders would benefit from the expected results/outcomes.

Method selection and application	
Fail (–, 0)	Inadequate method selection or poor application of method. Clear motivation, description, or analysis of the selected method is missing.
Acceptable (1)	Acceptable selection of method and acceptable application of method. The selected method is motivated, described, and analyzed in an acceptable way. At least one alternative method is briefly discussed.
Satisfactory (2)	...
Good (3)	Good selection of methods that represent together a comprehensive and appropriate approach. Good application of method(s). The method collection is well motivated, described, and analyzed. Alternative methods are discussed thoroughly.
Very good (4)	...
Excellent (5)	Excellent selection and application of methods that demonstrates a thorough understanding of the chosen field and the strengths, weaknesses, and limitations of different methods.

Knowledge and understanding	
Fail (–, 0)	Lacks a broad knowledge and understanding in software engineering. Lacks substantially deeper knowledge of at least one sub-area of software engineering. Lacks a deeper insight into current research and development.
Acceptable (1)	Demonstrates a broad knowledge and understanding in software engineering. Indicates substantially deeper knowledge of at least one sub-area of software engineering. Indicates a deeper insight into current research and development.
Satisfactory (2)	...
Good (3)	Demonstrates a broad knowledge and understanding in software engineering. Demonstrates substantially deeper knowledge of at least one sub-area of software engineering. Demonstrates a deeper insight into current research and development.
Very good (4)	...
Excellent (5)	Clearly demonstrates a broad knowledge and understanding in software engineering. Clearly demonstrates substantially deeper knowledge of at least one sub-area of software engineering. Clearly demonstrates a deeper insight into current research and development.

Critical thinking and scientific character

Fail (–, 0)	Inability or unwillingness to objectively and critically evaluate the relevance, reliability and validity of information/claims. Lacks evidence-based reasoning.
Acceptable (1)	Indicates some ability to objectively and critically evaluate the relevance, reliability and validity of information/claims. Key facts, claims and reasoning are relevant, objective, critical and based on evidence. Demonstrates some insight into the consequences that follow weakly supported claims.
Satisfactory (2)	...
Good (3)	Demonstrates an ability to objectively and critically evaluate the relevance, re-liability and validity of information/claims. Most facts, claims and reasoning are relevant, objective, critical and based on evidence. The work conveys some insight into the consequences that follow weakly supported claims.
Very good (4)	...
Excellent (5)	Excellent ability to objectively and critically evaluate the relevance, reliability and validity of information/claims. Excellent evidence-based reasoning abilities; all facts, claims and reasoning are relevant, objective, critical and based on evidence. The work conveys excellent insight into the consequences that follow weakly supported claims.

Planning (project plans only)

Fail (–, 0)	Trivial or unrealistic WBS (Work Breakdown Structure) that is not supported. Considers only trivial risks. Lacks impact analysis and mitigation strategy/plan.
Acceptable (1)	Feasible but not very realistic WBS and schedule. Insufficient analysis regarding required resources. Identifies some relevant non-trivial risks. Ignores some important risks and/or identifies irrelevant risks. Simplistic impact analysis and mitigation strategy/plan.
Satisfactory (2)	...
Good (3)	Realistic WBS, schedule and expectations about required resources. Identifies most relevant and important risks. Ignores irrelevant risks. Meaningful impact analysis and mitigation strategy/plan.
Very good (4)	...
Excellent (5)	Excellent WBS, schedule and analysis of required resources. Identifies all important risks and further relevant risks. Excellent impact analysis and mitigation strategy/plan.

B.1.3. Contribution (theses only)

Overall contribution	
Fail (–, 0)	Lacks contribution or potential contribution is not discussed or analyzed. Research questions have not been answered. Aims and objectives of the thesis have not been reached.
Acceptable (1)	The contribution to research and/or practice has been described, but not with sufficient clarity and objectivity. Most research questions have been answered in an acceptable way. Some aims and objectives of the thesis have been achieved.
Satisfactory (2)	...
Good (3)	The contribution to research and/or practice has been described clearly, objectively and critically. All research questions have been answered satisfactory. Aims and objectives of the thesis have been largely achieved.
Very good (4)	...
Excellent (5)	Scientific contribution of a high quality equivalent to a publishable conference or journal article.

Traceability and validity	
Fail (–, 0)	Lacks the description of relevant validity threats. Lacks the discussion of how obvious threats to validity might affect the results/conclusions. Conclusions are unclear and are not based on actual data/results and their analysis.
Acceptable (1)	Describes several relevant validity threats. Indicates how they might affect the results/conclusions. Conclusions are largely based on actual data/results and their analysis.
Satisfactory (2)	...
Good (3)	Discusses most relevant validity threats. Potential countermeasures have been described clearly. There is a clear thread of evidence from the data and their analysis through the results to the conclusions.
Very good (4)	...
Excellent (5)	All relevant validity threats are described and discussed clearly. Where appropriate, countermeasures have been taken and/or described clearly. There is a clear thread of evidence from the data and their analysis through the results to the conclusions.

Synthesis	
Fail (–, 0)	Lacks synthesis. Just restates or summarizes existing works and own materials.
Acceptable (1)	Limited synthesis. Indicates the ability to combine information from existing works and/or own materials to explain or discuss some phenomenon in a new way.
Satisfactory (2)	...
Good (3)	Satisfactory synthesis. Reorganizes and summarizes information from existing works and own materials and combines these into explanations/discussions/conclusions that go beyond the analysis of individual works or results. Answers many “So what?” questions.
Very good (4)	...
Excellent (5)	Clear and logical of reasoning. Excellent reasoning and organization of thoughts based on analyses of existing works and own materials to derive new knowledge. Strong focus on “So what?” questions.

B.1.4. Presentation (project plans and theses)

Disposition and layout	
Fail (–, 0)	Chapter and section headings are generic and not adapted to the specific topic. Tables, figures and illustrations are of a poor quality and have an inconsistent ad-hoc design.
Acceptable (1)	Chapter and section headings are somewhat adapted to the topic. Tables, figures and illustrations are of an acceptable quality. The design/layout of tables, figures and illustrations is somewhat consistent.
Satisfactory (2)	...
Good (3)	Systematic disposition. Chapter and section headings are adapted well to the topic and support the readability and comprehensibility of the text. Tables, figures and illustrations are of a good quality and have a consistent design/layout.
Very good (4)	...
Excellent (5)	Excellent and systematic disposition. Chapter and section headings are adapted excellently to the topic and support the readability and comprehensibility of the text very well. Tables, figures and illustrations are of an excellent quality and have a consistent and professional design/layout.

Adherence to academic standards

Fail (–, 0)	Lacks the most important and/or the most recent sources. The format and layout of references and citations are incomplete and inconsistent. Poor and inconsistent usage of tables, figures, illustrations and lists. Does not adhere to academic standards and conventions.
Acceptable (1)	Lacks some important and/or recent sources. The format and layout of references and citations are mostly complete and somewhat consistent. Acceptable, but inconsistent use of tables, figures, illustrations and lists. Academic standards and conventions are not always adhered to.
Satisfactory (2)	...
Good (3)	Cites all important sources. The format and layout of references and citations are complete and consistent. Uses tables, figures, illustrations and lists consistently where appropriate. Adheres to most academic standards and conventions.
Very good (4)	...
Excellent (5)	Cites all relevant sources. Flawless formatting and layout of references and citations. Excellent and consistent use of tables, figures, illustrations. Adheres to all academic standards and conventions.

Coherence and cohesion

Fail (–, 0)	There are no clear threads of reasoning throughout the report. The content does not fit logically together; ideas and arguments are not connected (coherence). Sentences, paragraphs and (sub)sections are not connected on a textual level (cohesion), i.e. the contents of the thesis give an impression of random order on idea level as well as on text level.
Acceptable (1)	There are some clear threads of reasoning. Coherence is acceptable in the key parts of the report (the most important ideas and arguments are presented in a logical order). Cohesion is acceptable in the key parts of the report (sentences and paragraphs build a logical flow of text).
Satisfactory (2)	...
Good (3)	There are clear threads of reasoning throughout the report. Coherence is good; in general, ideas and arguments fit together logically. Cohesion is good; in general, sentences in a paragraph, paragraphs in a section, and sections in a chapter build a natural flow of text.
Very good (4)	...
Excellent (5)	There is a single and clear thread of reasoning throughout the report. Coherence is excellent; all ideas and arguments fit together excellently. Cohesion is excellent; all sentences in a paragraph, all paragraphs in a section, and all sections in a chapter build a natural flow of text.

Language	
Fail (–, 0)	Poor. The text contains many spelling/grammar errors. The text is difficult to read and to understand due to language issues. The text does not meet the requirements for academic writing.
Acceptable (1)	Acceptable language processing. There are still many linguistic shortcomings, but the text is readable and understandable.
Satisfactory (2)	...
Good (3)	Good language processing. The subject is treated with clear prose that largely meets the requirements for academic writing. Some linguistic shortcomings remain, but the text is easy to read and easy to understand.
Very good (4)	...
Excellent (5)	Excellent language processing. The subject is treated with very clear prose that meets the requirements for scientific text of a very high quality.

B.2. Evaluation of aspects for oral presentations and defenses

B.2.1. Presentation

Oral description of problem and method	
Fail (–, 0)	Neither problems nor methods are motivated sufficiently. Descriptions are generally superficial and indistinct. The audience has difficulties understanding what was done and why.
Acceptable (1)	Acceptable motivation of problems and methods. Descriptions are too brief or too comprehensive but of an acceptable depth/level, so that the audience, in general, can understand most of what was done and why.
Satisfactory (2)	...
Good (3)	Clear and concise motivation of problems and methods. Descriptions are of adequate length and sufficient depth/level, so that the audience can understand clearly what was done and why.
Very good (4)	...
Excellent (5)	Problems and methods are described and motivated very clearly and concisely. The depth/level of the descriptions allows the audience to get deep insights on what was done and why. Problem aspects that are relevant outside the main topic of the thesis are discussed. Methods are compared systematically with related work.

Oral description of analysis and contribution

Fail (–, 0)	Neither analyses nor contributions are discussed or motivated clearly. Descriptions are generally superficial and indistinct. The audience has difficulties to understand conclusions and contributions.
Acceptable (1)	Acceptable motivation of analyses and contributions. Descriptions are too brief or too comprehensive but of an acceptable depth/level, so that the audience, in general, can understand conclusions and contributions.
Satisfactory (2)	...
Good (3)	Clear and concise motivation of analyses and contributions. Descriptions are of adequate length and sufficient depth/level, so that the audience can understand conclusions and contributions clearly.
Very good (4)	...
Excellent (5)	Analyses and contributions are described and motivated very clearly and concisely. The depth/level of the descriptions allows the audience to get deep insights into conclusions and contributions. The impact of contributions in areas outside the main topic of the thesis are discussed clearly. Results are systematically compared with related work.

Overall communication skills

Fail (–, 0)	The presentation is either too long or too short. The presentation is too vague or too superficial and not on an academic level. Poor language. Poor visuals.
Acceptable (1)	The presentation is mostly on time. The presentation is mostly on an acceptable academic level. Acceptable language. Acceptable visuals.
Satisfactory (2)	...
Good (3)	The presentation is on time. The presentation is on an adequate academic level. Good language. Good visuals. Maintains interest of audience.
Very good (4)	...
Excellent (5)	The presentation is on time. The presentation is on a very high academic level. Excellent language. Excellent visuals. Maintains interest of audience.

Argumentation and discussion abilities

Fail (–, 0)	Insufficient, inaccurate or very vague answers. Too brief or excessively long answers. Poor ability to handle critique constructively and objectively and to argue in defense.
Acceptable (1)	Most answers are clear and generally accurate but mostly too brief or lengthy. Acceptable ability to handle critique and discuss basic positions objectively and constructively. Indicates knowledge of the broader area of the topic.
Satisfactory (2)	...

Argumentation and discussion abilities

Good (3)	All answers are insightful, clear, factual, accurate and of adequate length/depth. Good ability to handle critique and engage in basic technical and scientific discussions objectively and constructively. Demonstrates good knowledge of the broader area of the topic.
Very good (4)	...
Excellent (5)	All answers are insightful, clear, factual, accurate and of adequate length/depth. Excellent ability to respond critique and engage in deep technical and scientific discussions objectively and constructively. Demonstrates excellent knowledge even beyond the actual topic.

B.3. Evaluation of aspects for written oppositions

B.3.1. Contents

Description of critical review

Fail (–, 0)	The report is sloppily executed and contains numerous errors. The critical review is superficial and generic. The presentation is generally unclear and inadequate.
Acceptable (1)	The critique is mainly objective, relevant and accurate. Contains some language and content errors. Acceptable structure and length/depth.
Satisfactory (2)	...
Good (3)	The critique is specific, balanced, objective, relevant and accurate. Contains very few language and content errors. Well-structured, well-organized and of adequate length/depth.
Very good (4)	...
Excellent (5)	The critique is very specifically balanced, objective, relevant and accurate. Basically free of errors. Excellent structure, organization and length/depth. Comments are systematically arranged and each comment is clearly motivated scientifically. Related work is cited as necessary.

Description of necessary revisions and recommended revisions

Fail (–, 0)	Suggestions are carelessly described and not organized systematically. The text contains numerous errors. Unclear how the suggestions would improve the reviewed thesis. The presentation is generally unclear.
Acceptable (1)	Suggestions are mostly constructive, relevant and accurate. In general, it is clear how the suggestions can be implemented and how they might improve the reviewed thesis. Contains some language and content errors.
Satisfactory (2)	...
Good (3)	Suggestions are constructive, highly relevant and accurate. It is clear how the suggestions can be implemented and how they will improve the reviewed thesis. Contains very few language and content errors. Suggestions are well-structured, well-organized and of adequate length/depth.

Description of necessary revisions and recommended revisions	
Very good (4)	...
Excellent (5)	Suggestions are very constructive, highly relevant and accurate. Quite clear how the suggestions can be implemented and how they will improve the reviewed thesis. Basically error-free. Excellent structure, organization and length/depth. Suggestions are systematically arranged and each suggestion is clearly motivated scientifically. Related work is cited as necessary.

B.3.2. Presentation

Analysis and argumentation	
Fail (–, 0)	Analysis and argumentation are missing, incorrect, not based on reasonable assumptions or do not rest on a scientific basis.
Acceptable (1)	Analysis and argumentation are based on some evidence. The level of analysis is generally acceptable for the discussed issues.
Satisfactory (2)	...
Good (3)	Analysis and argumentation are based on scientific evidence. Clear analysis and reasoning contribute to insights in the chosen area.
Very good (4)	...
Excellent (5)	Excellent analysis and argumentation. Advanced analysis and stringent reasoning contribute to substantial insights in the chosen area.

Language	
Fail (–, 0)	Poor. The text contains many spelling/grammar errors. The text is difficult to read and to understand due to language issues. The text does not meet the requirements for academic writing.
Acceptable (1)	Acceptable language processing. There are still many linguistic shortcomings, but the text is readable and understandable.
Satisfactory (2)	...
Good (3)	Good language processing. The subject is treated with clear prose that mainly meets the requirements for academic writing. Some linguistic shortcomings remain, but the text is easy to read and easy to understand.
Very good (4)	...
Excellent (5)	Excellent language processing. The subject is treated with very clear prose that meets the requirements for scientific text of a very high quality.

B.4. Evaluation of aspects for oral oppositions

B.4.1. Presentation

Overall communication skills	
Fail (–, 0)	The opposition is either too long or too short. The questions are too vague, too superficial or too generic and not on an academic level. Most questions are irrelevant. Poor language.
Acceptable (1)	The opposition is mostly on time. The questions are generally relevant and on an acceptable academic level. Acceptable language.
Satisfactory (2)	...
Good (3)	The presentation is on time. All questions are relevant and specific and on an adequate academic level. Good language. Maintains interest of audience.
Very good (4)	...
Excellent (5)	The presentation is on time. All questions are highly relevant, very specific and on a high academic level. Excellent language. Maintains interest of audience.

Analysis and argumentation	
Fail (–, 0)	Irrelevant, inaccurate or very vague questions. Very short or excessively long questions. More statements and general, generic comments than actual questions.
Acceptable (1)	Most questions are clear, as well as relevant and accurate. Indicates ability to present critique objectively and constructively.
Satisfactory (2)	...
Good (3)	All questions are clear, relevant and accurate. Demonstrates good ability to present critique objectively and constructively. Indicates ability to engage in technical and scientific discussions.
Very good (4)	...
Excellent (5)	All questions are very insightful, very clear, highly relevant and accurate. Demonstrates excellent ability to present critique objectively and constructively. Demonstrates excellent ability to engage in deep technical and scientific discussions.

APPENDIX C. DOCUMENT TEMPLATES

The project plan, the opposition report and the thesis shall be written using official document templates. Refer to Degree projects at DIDD, DIKO, DIPT, and DIKR at the Student Portal on the BTH website to find these document templates. This chapter describes necessary and recommended content as well as additional formal rules and requirements that govern the content of the documents to be produced in the degree project courses.

For all documents, the length should be appropriate for the content. That means that you should write concise and to the point. A longer document will not necessarily be better. On the contrary, adding irrelevant, inappropriate or repetitive content will likely result in lower marks. For details on academic writing you should consult Chapter 3.

C.1. Document template for the project plan

The project plan should include formal data about the thesis project, like title, author, person ID, e-mail-address, and supervisor's name(s) plus the following sections, preferably in this order: introduction, aim, objectives, research questions, method, expected outcomes, time and activity plan, risk management, and references.

The introduction should describe the background and context of the proposed work. It should identify a problem and motivate its relevance. Furthermore, the introduction should review related work and describe how the proposed work fits in. The aim should describe the overall goal of the thesis in one or two brief sentences. Compared to this overall goal, the objectives are more clearly defined. The objectives are usually measurable and should be completed before specific deadlines. The research questions section may include one or more research questions, or one or more hypotheses. In some cases, this section may include both hypotheses and research questions. In general, research questions should be concretely and clearly defined. Aims, objectives and research questions/hypotheses should be clearly related to each other. The method section should propose a suitable approach for answering the research questions together with a motivation for choosing the proposed approach. Expected outcomes briefly state the main expected outcomes. The time and activity plan divides the work into reasonable tasks and a reasonable schedule. In risk management, various problems that may threaten the timely delivery of acceptable results are briefly investigated.

C.2. Document template for monograph theses

The thesis is the main artifact of a degree project. Accepted theses will be published electronically. It is therefore important that all theses have a uniform and professional appearance and, to some extent, a predictable breakdown of the content. A uniform appearance is particularly important with respect to the front and back of the title page. The title page should therefore be produced by filling in the document template for theses with the necessary information.

The title page should be followed by a structured abstract which should summarize the report by describing: context, objectives, methods, results and conclusions. At least one and a maximum of three keywords must be specified in succession after the structured abstract.

The next page can be set aside for thanks or acknowledgements to those who helped you during the thesis work. The content and format of this page is up to you, but the text should only highlight the people who actually contributed to the thesis in some way. These persons should be arranged by the extent of their respective contributions. Usually this means that the student would like to thank the supervisor, any other contributing colleagues, fellow students or family members.

The order of the main content of the thesis depends on the actual topic of the thesis. Different genres, research areas or research methods have different traditions. Since there is no "one-size-fits-all" structure, you need to investigate carefully what is appropriate for your particular topic. The following items should, however, be included for all types of theses (but not necessarily in this order): table of contents, introduction (including background, motivation, a clear problem description/motivation, and aims and objectives), related work, methodology, results, analysis, discussions, conclusions and bibliography/references. Including an item does not necessarily mean that you need to include a separate section with this particular heading. It just means that this particular type of content should be included. Heading names should be chosen so that they fit your particular topic.

The thesis text should have a scientific basis and be free from grammatical errors. Your supervisor is not responsible for pointing out language, style and grammar issues. This would furthermore be a waste of precious subject expert time. Remember to write concise and to the point. Theses should usually not exceed 80 pages in length excluding appendices.

C.3. Document template for compilation-style theses

A thesis can be also written as a scientific journal article instead of a monograph.

A journal article presents the materials in a more concise form and usually contains less basic information about the research methodology (since it targets a key audience with more knowledge of the theory and methods of the research area in question than the general population). Due to this reason and since the journal article format is rather dense in general, it is difficult to include certain content that is required (and assessed) in theses. The compilation-style thesis therefore needs to include an introductory chapter that precedes the journal article and which includes a more thorough description and motivation of the research methodology (and any additional topics that are required in the degree project course but which are not included in the journal article).

C.4. Document template for opposition reports

The opposition report comprises four main parts: introduction, critical review, required changes and recommended changes.

The introduction gives a brief overview over the content of reviewed thesis, results and contributions from the reviewer's point of view. The critical review highlights the strengths and weaknesses of the reviewed thesis. Required changes point out issues that need to be resolved in the final version of the thesis. Recommended changes point out further issues that are less critical but would help to improve the thesis further.

BIBLIOGRAPHY

- [1] **Mikael Berndtsson, Jörgen Hansson, Björn Olsson and Björn Lundell.** Thesis Projects: A Guide for Students in Computer Science and Information Systems. Springer Science & Business Media, 2nd edition, 2008. Main textbook.
- [2] **Pär Blomkvist and Anette Hallin.** Method for Engineering Students: Degree Projects Using the 4-Phase Model. Studentlitteratur, 2015. This text is available in Swedish and in English.
- [3] **Douglas W Cunningham and Christian Wallraven.** Experimental Design: From User Studies to Psychophysics. CRC Press, 2011.
- [4] **Christian W Dawson.** Projects in Computing and Information Systems — A Student's Guide. Addison-Wesley, 2nd edition, 2009.
- [5] **Rolf Dobelli.** The Art of Thinking Clearly. Harper Paperbacks, reprint edition, 2014. Explains numerous common threats to validity in everyday examples.
- [6] **Daniel Kahneman.** Thinking, Fast and Slow. Penguin, 2011. Nobel prize winner explains how humans think.
- [7] **Richard Paul and Linda Elder.** The Miniature Guide to Critical Thinking: Concepts & tools. Foundation for Critical Thinking Press, 5th edition, 2008.
- [8] **Colin Robson and Kieran McCartan.** Real World Research. Wiley, 4th edition, 2016.
- [9] **William Strunk and White E B.** The Elements of Style. Pearson, 4th edition, 1999. The classics of writing and composition.

- [10] **William Trochim, James Donnelly and Kanika Arora.** Research Methods: The Essential Knowledge Base. Wadsworth Publishing, 2nd edition, 2015.
- [11] **Gabe T. Wang and Keumanjae Park.** Student Research and Report Writing: From Topic Selection to the Complete Paper. Wiley, 2016.
- [12] **Joseph M Williams and Joseph Bizup.** Style: The Basics of Clarity and Grace. Pearson, 5th edition, 2014.
- [13] **Joseph M Williams and Gregory G Colomb.** The Craft of Argument. Pearson, 3rd edition, 2007.
- [14] **Claes Wohlin, Per Runeson, Martin Höst, Magnus C Ohlsson, Björn Regnell and Anders Wesslén.** Experimentation in Software Engineering — An Introduction. Springer, 2nd edition, 2012.
- [15] **William Zinsser.** On Writing Well: The Classic Guide to Writing Nonfiction. Harper Perennial, 30th anniv. rep. edition, 2016.
- [16] **Justin Zobel.** Writing for Computer Science. Springer Science & Business Media, 3rd edition, 2014. Main textbook.

Course Module 1

Advanced Network and Cloud Security



Purpose

- Learn to detect the vulnerabilities of clouds, software-defined networks, virtualization technologies and to develop solutions for their protection.

Outcoming skills

- Protect cloud infrastructures against cyber attacks.
- Research new trends of vulnerabilities.
- Protect software-defined networks.

Outcoming knowledge

- Software-defined networks.
- OpenFlow protocol principles.
- SDN controller security.
- Flex VPNs configuration, IPS/IDS deep packet inspection.
- Virtualization principles: issues, security threats and solutions.
- Network virtualization principles and overlays.
- Security and privacy issues in cloud computing.
- Cloud confidentiality, integrity availability, accountability, privacy.

Result

- Lectures in total — **16.**
- Laboratory works in total — **12.**
- Workshops in total — **2.**
- Trainings in total — **2.**

Course Module 1

Advanced Network and Cloud Security



Course name: Advanced Network and Cloud Security.

Level: Master Programme.

Developers:

- *Manager:* Igor Ushakov, The Bonch-Bruевич Saint-Petersburg State University of Telecommunications, Russian Federation.
- *Manager:* Anders Carlsson, Blekinge Institute of Technology, Sweden.
- Igor Kotenko, SPIIRAS, Russian Federation.
- Andrey Krasov, The Bonch-Bruевич Saint-Petersburg State University of Telecommunications, Russian Federation.
- Kurt Tutschku, Blekinge Institute of Technology, Sweden.
- Elena Niemkova, Lviv Politechnic National University, Ukraine.

Purpose of the course

The purpose of the course is to provide knowledge how to detect the vulnerabilities of clouds, software-defined networks, virtualization technologies and to develop solutions for their protection.

Relevance of the course

The course teaches the fundamentals of cloud, SDN, virtualization technologies, IPS/IDS, advanced VPN protection.

One workshop “Train the trainer” was held.

The course has been adopted by two universities:

- The Bonch-Bruевич Saint-Petersburg State University of Telecommunications, Russian Federation.
- Kharkiv National University of Radio Electronics, Ukraine.

Lectures

Lecture 1. Introduction to advanced networking

By the end of this module/lesson students will know/be able to:

- understand changes on the Internet;
- know service network architectures;
- understand softwarization of future networks.

Lecture 2. Security and privacy in cloud computing: state of the art

By the end of this module/lesson students will know/be able to:

- describe cloud computing: definition and architecture;
- know security and privacy challenges;
- describe attribute-driven methodology for representing cloud security and privacy.

Lecture 3. Virtualization: issues, security threats and solutions

By the end of this module/lesson students will know/be able to:

- know virtualization basics;
- understand properties and approaches of virtualization security issues.

Lecture 4. SDN principles

By the end of this module/lesson students will know/be able to:

- understand general concepts of SDN;
- know how to work with SDN controller;
- know API services;
- know examples of network emulators and simulator for SDN.

Lecture 5. OpenFlow

By the end of this module/lesson students will know/be able to:

- know OpenFlow basics;
- understand Open vSwitch;
- define briefly what Open vSwitch (OVS) is;
- know the architecture of OVS;
- look briefly at the use of OVS in OpenStack.

Lecture 6. Network virtualization and overlays

By the end of this module/lesson students will know/be able to:

- know OpenFlow basics;
- understand Open vSwitch;
- define briefly what Open vSwitch (OVS) is;

- know the architecture of OVS;
- look briefly at the use of OVS in OpenStack.

Lecture 7. Layer 3 VPN

By the end of this module/lesson students will know/
be able to:

- describe VPN definition;
- describe key threats to WANs and remote access;
- describe VPN types;
- describe IPsec, IKE v.1 and v.2 protocols;
- describe different types of VPN including FlexVPN.

Lecture 8. Virtual private networks: SSL/TLS VPN

By the end of this module/lesson students will know/
be able to:

- describe SSL and TLS protocols;
- main goals of TLS protocol;
- know how server authenticated TLS session works;

- know attacks on TLS;
- know possible ways to attack mitigation.

Lecture 9. SDN security

By the end of this module/lesson students will know/
be able to:

- understand origins of SDN;
- know security management.

Lecture 10. Intrusion detection & prevention systems

By the end of this module/lesson students will know/
be able to:

- know intrusion detection and prevention principles;
- know IDPS technologies: components and architecture; types of detected events;
- understand open IDPS (Snort & Suricata).

Lecture 11. OpenStack security

By the end of this module/lesson students will know/
be able to:

- understand what OpenStack is;
- know how Keystone works;
- know how to work with TLS proxies and HTTP services;
- understand secure reference architectures.

Lecture 12. Cloud confidentiality

By the end of this module/lesson students will know/
be able to:

- know threats to cloud confidentiality;
- understand confidentiality defense strategies.



Lecture 13. Cloud integrity

By the end of this module/lesson students will know/be able to:

- know threats to cloud integrity;
- know integrity defense strategies based on provable data possession;
- know integrity defense strategies on third party auditor;
- understand methods against dishonest computing.

Lecture 14. Cloud availability

By the end of this module/lesson students will know/be able to:

- know threats to cloud availability;
- understand availability defense strategies.

Lecture 15. Data accountability in cloud systems

By the end of this module/lesson students will know/be able to:

- threats to cloud accountability;
- accountability defense strategies.

Lecture 16. Cloud privacy

By the end of this module/lesson students will know/be able to:

- know threats to cloud privacy and privacy defense strategies;
- understand fully homomorphic encryption, privacy manager and trusted hardware token;
- know privacy issue caused by data indexing, privacy-as-a-service and privacy preserving framework.

Laboratory works

Lab 1. Installation of virtualization system and VMs running

Goal: obtain skills in installation of virtualization system including virtual machines creation.

Lab 2. Virtual network creation

Goal: gain skills in virtual network creation, compare resulted topology with real network.

Lab 3. Install OpenFlow controller in virtual machine

Goal: gain skills in OpenFlow controller configuration.

Lab 4. TLS middleboxes

Goal: gain skills in TLS middlebox configuration.

Lab 5. DMVPN configuration

Goal: gain skills in DMVPN configuration, explore the creation process of dynamic VPN tunnels.

Lab 6. OpenStack configuration

Goal: gain skills in OpenStack configuration.

Lab 7. SDN controller security

Goal: obtain skills in SDN security mechanisms.

Lab 8. Creation of the fingerprint/deep packet inspection with Snort/Suricata

Goal: gain skills in signature creation and testing with open source IPS.

Lab 9. Virtualization security

Goal: gain skills in virtualization security and attacks, including attacks on violation of accessibility, confidentiality and integrity.

Lab 10. Confidentiality and integrity: attacks and defense in cloud systems

Goal: gain skills in attacks related to exploitation of hypervisor vulnerabilities, including attacks on violation of confidentiality and integrity.

Lab 11. Availability: attacks and defense in cloud systems

Goal: gain skills in DoS attacks, SYN-flood attacks and DNS amplification.

Lab 12. Assessment lab

Goal: verify the obtained skills and knowledge.

Curricula-Syllabus

Activity	Lectures	Practical works	Laboratory works	Tests	Exams	Classroom hours total	Individual work and self study hours total	Total hours	Credits
Hours	34	—	40	2	4	80	172 (including 36 hours for preparatory exam in RU)	252	7 = 252 hours in RU

1. The purpose and objectives of the course

Aim

Learn to detect the vulnerabilities of clouds, software-defined networks, virtualization technologies and to develop solutions for their protection

Learning Outcomes

At the end of the session a successful student will

know:

- ✓ Software-defined networks;
- ✓ OpenFlow protocol principles;
- ✓ SDN controller security;
- ✓ Flex VPNs configuration, IPS/IDS deep packet inspection;
- ✓ Virtualization principles: issues, security threats and solutions;
- ✓ Network virtualization principles and overlays;
- ✓ Security and privacy issues in cloud computing;

- ✓ Cloud confidentiality, integrity availability, accountability, privacy;

be able to:

- ✓ protect cloud infrastructures against cyber-attacks;
- ✓ research new trends of vulnerabilities;
- ✓ protect software-defined networks.

Preknowledge

- ✓ Communication protocols, TCP/IP stack;
- ✓ general programming skills;
- ✓ concepts of cloud system building, threats specific to cloud architecture.

Assessment of acquired skills

The analysis of vulnerability is executed on all types covered in this course (cloud, SDN, virtualization) and actions are taken to eliminate and minimize the attack.

2. Course content

2.1. Lectures

Lectures	Hours	Comments
Lecture 1. Introduction to advanced networking	2	
Lecture 2. Security and privacy in cloud computing: state of the art	2	
Lecture 3. Virtualization: issues, security threats and solutions	2	
Lecture 4. SDN principles	4	
Lecture 5. OpenFlow	2	
Lecture 6. Network virtualization and overlays	2	
Lecture 7. Layer 3 VPN (IPSec, MPLS VPN, GRE, DMVPN, FlexVPN, GET VPN)	2	
Lecture 8. Layer 4 VPN (SSL/TLS) – middleboxes	2	
Lecture 9. SDN controller security	2	
Lecture 10. Intrusion detection and prevention systems (advanced)	2	
Lecture 11. OpenStack security	2	
Lecture 12. Cloud confidentiality: threats and defense	2	
Lecture 13. Cloud integrity: threats and defense	2	
Lecture 14. Cloud availability: threats and defense	2	
Lecture 15. Data accountability in cloud systems	2	
Lecture 16. Cloud privacy	2	
Total lectures	34	

2.2. Laboratory works

Labs	Hours	Comments
Lab 1. Installation of virtualization system and VMs running	2	
Lab 2. Virtual network creation. Comparison with real network	4	
Lab 3. Installation of OpenFlow controller in virtual machine (pre-installation). Usage of virtual switch. Deployment of Open vSwitch (OpenFlow commands).	4	
Lab 4. TLS middleboxes	2	

Labs	Hours	Comments
Lab 5. DMVPN configuration	4	
Lab 6. OpenStack configuration	4	
Lab 7. SDN controller security	4	
Lab 8. Creation of the fingerprint/deep packet inspection with Snort/Suricata	4	
Lab 9. Virtualization security	4	
Lab 10. Confidentiality and integrity: attacks and defense in cloud systems	4	
Lab 11. Availability: attacks and defense in cloud systems	4	
Lab 12. Assessment lab	8	
Total labs	40	

2.3. Individual and self-study works

Chapter/Lecture	Example topics of individual tasks and topics for self-study	Hours
Chapter 1	Intro self-study	28
Chapter 2	Software-defined network principle self-study using references from the lecture notes	36
Chapter 3	Advanced VPNs self-study using references from the lecture notes	36
Chapter 4	Control plane security and network control elements self-study	36
Chapter 5	Cloud security self-study using references from the lecture notes	36
Total		172

2.4. Tests and exams

Chapter/Lecture	Hours
Lectures	Final test
Labs	Assessment lab
FINAL	Final Exam
Total	14

Additional sources



1. "The Zettabyte Era: Trends and Analysis", Cisco White Paper
<https://www.cisco.com/c/en/us/solutions/collateral/service-provider/visual-networking-index-vni-hyperconnectivity-wp.html>



2. Software-Defined Networking (SDN) Definition
<https://www.opennetworking.org/sdn-definition/>



3. Openflow — enabling innovation in your network
<https://www.opennetworking.org/sdn-resources/openflow/>



4. Security guidance for critical areas of focus in Cloud Computing v3.0
<https://downloads.cloudsecurityalliance.org/assets/research/security-guidance/csaguide.v3.0.pdf>



5. Mitigation of Distributed Denial of Service Attacks in the Cloud
<https://www.degruyter.com/view/j/cait.2017.17.issue-4/cait-2017-0040/cait-2017-0040.xml>



4. Internetworking with TCP/IP Volume One (6th Edition)
<https://www.amazon.com/Internetworking-TCP-IP-One-6th/dp/013608530X?tag=routerfr-20>

Official sources (public)



1. CCIE Routing and Switching Certification Guide (4th Edition)
<https://www.amazon.com/CCIE-Routing-Switching-Certification-Guide/dp/1587059800?tag=routerfr-20>



2. Routing TCP/IP, Volume 1 (2nd Edition)
<https://www.amazon.com/Routing-TCP-IP-1-2nd/dp/1587052024?tag=routerfr-20>



3. Computer Networking: A Top-Down Approach (6th Edition)
<https://www.amazon.com/Computer-Networking-Top-Down-Approach-6th/dp/0132856204?tag=routerfr-20>

Course Module 2

Wireless and Mobile Security

Purpose

- To provide knowledge in security architecture of wireless and mobile communication systems, information threats models, vulnerabilities and protection abilities in wireless networks. The students will obtain practical skills in penetration testing of wireless networks and in usage of various protection methods.

Outcoming skills

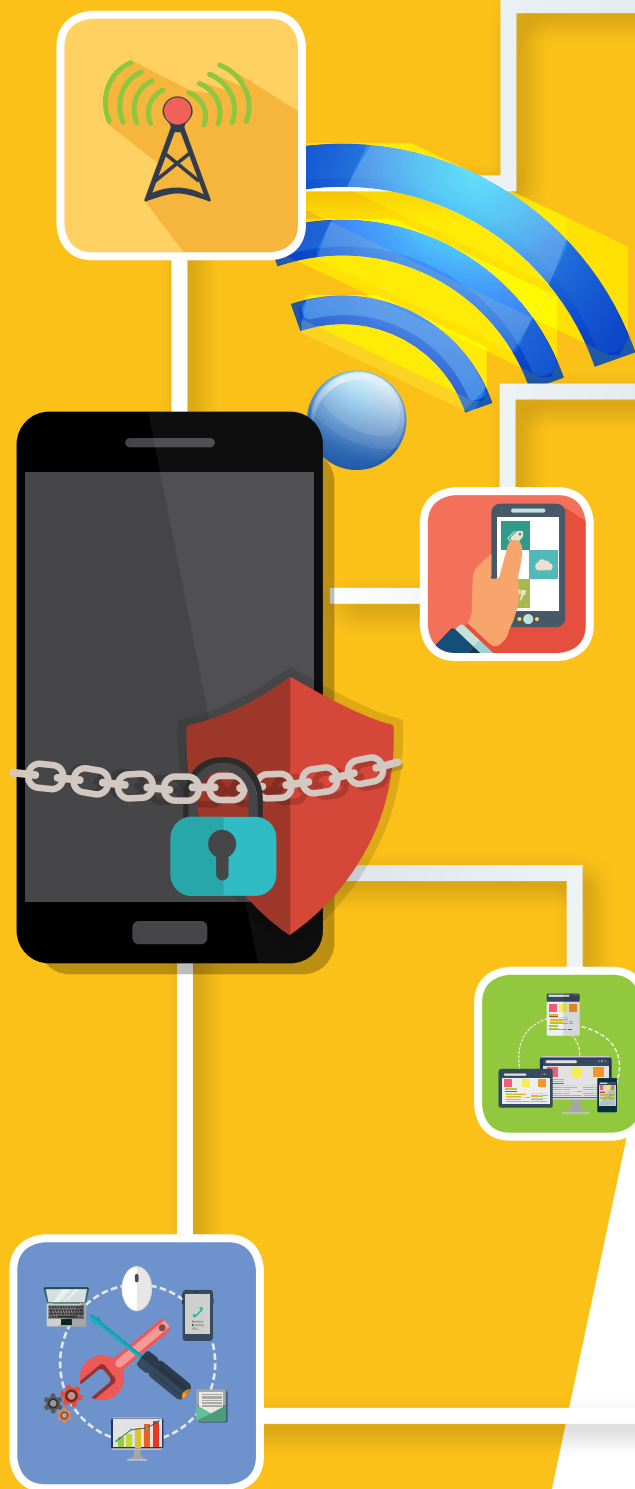
- Design a wireless infrastructure.
- Choose wireless configuration.
- Create security policies.
- Set permission rankings.
- Conduct an audit of the service network.
- Match the current network with other wired and wireless networks.
- Ensure the availability and scalability of the wireless network.
- Protect a mobile device and a communication channels against cyber attacks.

Outcoming knowledge

- Wi-Fi protocols and their weaknesses.
- techniques and tools for Bluetooth, ZigBee, etc., protocol analysis and compromising methods.
- embedded and cyber physical system features and security aspects.
- security of mobile networks (GSM, CDMA protocols).

Result

- Lectures in total — **21**.
- Laboratory works in total — **10**.
- Workshops in total — **1**.
- Trainings in total — **1**.





Course name: Wireless & Mobile Security.

Level: Master Programme.

Developers:

- *Manager:* Maryna Yevdokymenko, Kharkiv National University of Radio Electronics, Ukraine.
- *Manager:* Anders Carlsson, Blekinge Institute of Technology, Sweden.
- Vladimir Sokolov, State University of Telecommunications, Ukraine.
- Dmytro Ageiev, Kharkiv National University of Radio Electronics, Ukraine.
- Iveniiia Kuzminykh, Kharkiv National University of Radio Electronics, Ukraine.

Purpose of the course

To provide knowledge in security architecture of wireless and mobile communication systems, information threat models, vulner-

abilities and protection abilities in wireless networks. The students will obtain practical skills in penetration testing of wireless networks and in usage of various protection methods.

Relevance of the course

The course of «Wireless and Mobile Security» is very important due to the intensive development of wireless networks. The course covers a comprehensive wireless and mobile security overview including the design, planning, installation and maintenance of wireless network security infrastructures.

The focus is on the security techniques available for wireless local area networks and for wireless devices used to access the Internet. This course assists in understanding of wireless security requirements and their implementation.

Lectures

Lecture 1. Threat models of wireless networks

By the end of this module/lesson students will know/
be able to:

- threat modeling terminology;
- specifics of wireless communication from a security standpoint;
- eavesdropping;
- wardriving;
- wireless denial-of-service attack;
- wireless man-in-the-middle attack;
- evil twin;
- evolution of malicious programs and antivirus solutions.

Lecture 2. Audit tools of wireless networks

By the end of this module/lesson students will know/
be able to:

- wireless network survey;
- software types for wireless audit;
- hardware types for wireless audit.

Lecture 3.

Basic protection techniques

By the end of this module/lesson students will know/
be able to:

- wireless security policy;
- basic WLAN encryption and authentication requirements;
- additional activities that support protection.

Lecture 4.

The wireless threats

By the end of this module/lesson students will know/
be able to:

- wireless impact on traditional security approaches, signal exposure threats, common misconceptions in wireless security, wireless LAN and MAN signal leakage, information disclosure threats, DoS attacks, rogue AP attacks, wireless protocol deficiencies, anonymity attacks, home user threats.

Wireless and Mobile Security

Lecture 5. Wireless LAN organizations and standards

By the end of this module/lesson students will know/be able to:

- understand wireless standards bodies, role of the Wi-Fi Alliance for interoperability testing, capabilities and features of WPA and WPA2, IETF standards, understand the RADIUS and EAP protocols;
- identify and understand the enterprise impact of security-pertinent wireless standards including: 802.11z "Direct Link Setup", 802.11ac "Gigabit over Wi-Fi", 802.11af "Wi-Fi in TV White Space";
- obtain information about the work of standards bodies and working group resources.

Lecture 6. Sniffing of wireless networks: tools, techniques and implementation

By the end of this module/lesson students will know/be able to:

- use wireless sniffing as an analysis mechanism, understand WLAN card operating modes, use sniffing in managed mode, sniffing in monitor mode, understand advantages of RFMON sniffing, RFMON implementations;
- monitor mode sniffing;
- analyze wireless traffic with Tcpdump, Wireshark and Kismet;
- sniff wireless, use Wireshark, identify wireless networks with Kismet, map wireless networks with gpsmap, Google Maps, Google Earth.

Lecture 7. IEEE 802.11 MAC: in-depth

By the end of this module/lesson students will know/be able to:

- common capabilities of the IEEE 802.11 MAC, understand the architecture and operating of ad-hoc and infrastructure networks, phases of station authentication and association, understand the operation and behavior of IEEE 802.1X authentication;
- identify capabilities and features of EAP types including PEAP, EAP/TLS, TTLS, EAP-FAST;
- Packet framing on wireless networks, understand the 802.11 header format and fields, significance of FromDS and ToDS fields, 802.11 address field ordering and behavior, 802.11 management frames and data encoding, 802.11 management action frames, decoding frames in hex.

Lecture 8. Network level security of wireless networks

By the end of this module/lesson students will know/be able to:

- firewall for wireless network;
- network topologies;
- policy of filtering traffic;
- VPN for wireless network;
- main cases of VPN usage in wireless networks;
- overview of VPN protocols.

Lecture 9. Attacking WEP

By the end of the module the students will know/be able to know:

- introduction to WEP technology, WEP key selection, IV transmission, WEP framing;
- understand the XOR truth table;
- introduction to RC4, WEP ICV processing, WEP encryption process, WEP decryption process;
- WEP failures including lack of replay protection, weak message integrity check, no key rotation mechanism, the cases when initialization vector is too short, challenge/response reveals PRGA and key is reversible from ciphertext;
- reliable mechanisms for exploiting and decrypting WEP networks;
- apply WEP failures to other network protocols.

Lecture 10. Exploiting ZigBee

By the end of this module/lesson students will know/be able to:

- introduction to ZigBee, ZigBee use cases and deployment;
- attacker interest in ZigBee and industrial control systems;
- ZigBee and IEEE 802.15.4 physical and MAC layer architecture;
- ZigBee and IEEE 802.15.4 security mechanisms;
- authentication and cryptographic controls;
- weaknesses in ZigBee key provisioning and management mechanisms;
- tools for eavesdropping and manipulation of ZigBee networks;
- exploit ZigBee over-the-air (OTA) key provisioning;
- locate ZigBee devices with signal analysis tools.

Lecture 11. Bluetooth technology and its security features

By the end of this module/lesson students will know/be able to:

- Bluetooth technology introduction, assessing of the Bluetooth protocol stack;
- Bluetooth classic device analysis, procedure for joining a Bluetooth piconet, physical layer components;
- Bluetooth Low Energy (4.0) technology analysis, use cases, deployment models and structure;
- Bluetooth profiles and application features, Bluetooth security options, leveraging of Bluetooth link authentication and encryption;
- techniques for auditing and identifying Bluetooth devices, techniques for locating Bluetooth transmitters on Windows and Android platforms;
- Bluetooth policy and device configuration best practices.

Lecture 12. Bluetooth vulnerability, attacks and countermeasures

By the end of this module/lesson students will know/be able to:

- understand Bluetooth pairing, analyze the Bluetooth authentication exchange and associated protocols, attack the Bluetooth pairing process, implement PIN attacks;
- attack the Bluetooth E0 encryption algorithm;
- sniffing of Bluetooth networks, hacker techniques for building Bluetooth sniffers, interaction with Bluetooth networks using the Ubertooth One;
- exploit Bluetooth non-discoverable mode, discover non-discoverable devices;
- exploit Bluetooth profile vulnerabilities;
- Bluetooth device auditing, Bluetooth protocol fuzzing techniques, device enumeration.

Lecture 13. Deficiencies in TKIP networks

By the end of this module/lesson students will know/be able to:

- TKIP improvements over WEP networks including keying, message integrity checks (MIC), IV sequencing;
- QoS deficiencies and the TKIP break, TKIP replay attacks;
- TKIP countermeasure to DoS attacks;
- TKIP plaintext recovery attacks;
- Applying of TKIP failures to modern cryptographic systems.

Lecture 14. Leveraging of Wi-Fi DoS attacks

By the end of this module/lesson students will know/be able to:

- understand the impact of DoS attacks, differentiate persistent and non-persistent DoS attacks, IEEE 802.11 DoS attack targets including PHY, MAC and client attacks;
- physical medium attacks with the Wave Bubble, common jammers;
- IEEE 802.11 MAC attacks, authentication and association floods, deauthentication and disassociation floods, Beacon DS Set DoS;
- Impact of IEEE 802.11w and management frame protection and DoS attacks;
- IEEE 802.11 medium management techniques, hidden node problem, RTS/CTS medium management, medium reservation attacks, RTS/CTS co-opting;
- client attacks including rogue AP DoS, NULL SSID DoS, 802.1X authentication flood.

Lecture 15. Wireless fuzzing for bug discovery

By the end of this module/lesson students will know/be able to:

- value of protocol fuzzing for fault determination in wireless networks;
- implement fuzzing testing operationally, scope, monitor, record and analyze results;
- when to use fuzzing as a test mechanism;
- strategies for vulnerability disclosure.

Lecture 16. WLAN IDS analyst techniques

By the end of this module/lesson students will know/be able to:

- introduction to IDS concepts, differentiation of true positives from false positives, events of interest assessment;
- WIDS deployment models including overlay, integrated, and hybrid deployments;
- techniques to identify attacks including signature analysis, trend analysis and anomaly analysis;
- evaluating of attacks through traffic analysis, several examples;
- evaluating of WIDS systems, event aggregation, light bulb deployment, secure communication protocols, intrusion protection services, integration with third-party IDS systems;
- WIDS deployment considerations including facility coverage, dwell time, logging fidelity, event storage, trend analysis.

Wireless and Mobile Security

Lecture 17. Deploying a secure wireless infrastructure

By the end of this module/lesson students will know/ be able to:

- recommendations how to manage an authentication architecture, leverage the RADIUS protocol for authentication validation, RADIUS data encoding rules, EAP transmitted over RADIUS;
- understand the impact of a compromised CA, “evil twin” attack;
- recommendations and preferences how to select an EAP type;
- understand the advantages and disadvantages of EAP/TLS, PEAP, PEAPv1, PEAPv2, TTLS, EAP-FAST, PEAP-EAP-TLS;
- summary and recommendations how to select an EAP type.

Lecture 18. Configuring and securing wireless clients

By the end of this module/lesson students will know/ be able to:

- manage client certificate trust policies, default Windows root CA trust;
- four techniques for deploying of a new root certificate authority: manual, web-server delivery, scripted web-server delivery, automatic trust with GPO;
- manage client configuration settings with Windows, cached authentication credentials with PEAP on Win-

dows WZC, deploy GPO settings for preferred wireless network, specify the configuration and settings of preferred WZC networks, edit and implement wireless-specific GPO policies, recommendations for securing PEAP through GPO;

- manage third-part wireless manager tools with the Funk Odyssey supplicant, create a custom installer with Odyssey manager.

Lecture 19. Protection methods in mobile technologies

By the end of this module/lesson students will know/ be able to:

- GSM network security mechanisms.

Lecture 20. GSM network security mechanisms, vulnerability, attacks and countermeasures.

By the end of this module/lesson students will know/ be able to:

- GSM/GPRS vulnerability, attacks and countermeasures.

Lecture 21. 3G and LTE Security

By the end of this module/lesson students will know/ be able to:

- problems with GSM security;
- new security features of 3G;
- new security features of LTE.

Laboratory works

Lab 1. Wireless network mapping

Goal: Obtain data on wireless networks and visualize the relationships between the elements.

Lab 2. Monitor network traffic

Goal: Collect wireless network data using sniffer and analyze this data on vulnerabilities.

Lab 3. WEP and WPS hacking technologies

Goal: Consider the common methods of hacking Wi-Fi networks.

Lab 4. Research of radio frequency Wi-Fi resources in 2.4–2.5 GHz range

Goal: Consider different ways to obtain information about the radiation levels in wireless networks without the use of special spectrum analyzers.

Lab 5. Wi-Fi network DoS attacks

Goal: Consideration of DoS ways to attack the Wi-Fi network.

Lab 6. Wi-Fi fuzzing

Goal: Make fuzz testing of wireless access point and to conduct security analysis of the part of PCI DSS standard that is responsible for the wireless network.

Lab 7. OTA

Goal: Check the firmware updates over the network. To analyze the data transmitted over the network during the upgrade. To investigate the possibility of substitution of transferred firmware.

Lab 8. Research of stress loading of wireless network

Goal: Assemble a sample of the test access point-based on single-board computer. To explore one of the security aspects of wireless infrastructure — availability.

Lab 9. 125 kHz RFID sniffing

Goals: Consider the work EM-Marin (EM4100 or EM4102) protocol for 125 kHz RFID and sniff it.

Lab 10. Crunch password list generation

Goals: Obtain a relevant password list by crunching password list generation tool on Kali Linux.



Curricula-Syllabus

Activity	Lectures	Practical works	Laboratory works	Tests	Exams	Classroom hours total	Individual work and self study hours total	Total hours	Credits
Hours	42	—	40	2	4	88	164 (including 36 hours for preparatory exam in UA)	252	7 = 252 hours in UA

1. The purpose and objectives of the course

Aim

Provide knowledge in security architecture of wireless and mobile communication systems, information threat models, vulnerabilities and protection abilities in wireless networks. The students will obtain practical skills in penetration testing of wireless networks and using of various protection methods.

Learning Outcomes

At the end of the course a successful student will

understand:

- ✓ Wi-Fi protocols and their weaknesses;
- ✓ Techniques and tools for Bluetooth, ZigBee, analysis of other protocols and compromising methods;
- ✓ Embedded and cyber physical system features and security aspects;
- ✓ Security of mobile networks (GSM, CDMA protocols);

be able to:

- ✓ design a wireless infrastructure;
- ✓ choose wireless configuration;
- ✓ create security policies;
- ✓ set permission rankings;
- ✓ conduct an audit of the service network;
- ✓ match the current network with other wired and wireless networks;
- ✓ ensure the availability and scalability of the wireless network;
- ✓ protect a mobile device and a communication channel against cyber attacks.

Preknowledge

- ✓ Standards for wireless interaction (IEEE 802.11, 802.16.4, 802.15.4, 3GPP, etc.).
- ✓ Basics of hardware, networking, OS, cryptology, information theory.
- ✓ Basics of wireless technologies (Wi-Fi, LTE, Bluetooth, WiMAX, CDMA, GSM). The principle of channel formation in these technologies. Protocols used in these technologies. The network access organizing principle in different technologies.

Assessment of acquired skills

- ✓ Know vulnerabilities and how to use them in wireless networks.
- ✓ Identify threats of malicious entering or access to wireless networks. Specialized network equipment used to secure wireless networks. To design protected (subject to threats) wireless networks.

2. Course content

2.1. Lectures

Chapter/Lecture	Content	Hours
Chapter 1. Wireless Networks Threat Models		6
Lecture 1.1	Wireless Networks Threat Models	2
Lecture 1.2	Wireless Networks Audit Tools	2
Lecture 1.3	Basic Protection Techniques	2
Chapter 2. Wireless Data Collection and Wi-Fi MAC Analysis		8
Lecture 2.1	The Wireless Threats Wireless impact on traditional security approaches, signal exposure threats, common misconceptions in wireless security, wireless LAN and MAN signal leakage, information disclosure threats, DoS attacks, rogue AP attacks, wireless protocol deficiencies, anonymity attacks, home user threats.	2
Lecture 2.2	Wireless LAN Organizations and Standards - Understanding of wireless standards bodies, role of the Wi-Fi Alliance for interoperability testing, capabilities and features of WPA and WPA2, IETF standards, understanding of RADIUS and EAP protocols; - Identifying and understanding of the enterprise impact of security-pertinent wireless standards including: 802.11z "Direct Link Setup", 802.11ac "Gigabit over Wi-Fi", 802.11af "Wi-Fi in TV White Space"; - Obtaining of information about standards bodies work and working group resources.	2
Lecture 2.3	Sniffing of Wireless Networks: Tools, Techniques and Implementation - Using of wireless sniffing as an analysis mechanism, understanding of WLAN card operating modes, sniffing in managed mode, sniffing in monitor mode, advantages of RFMON sniffing, RFMON implementations; - Monitor mode sniffing; - Analysis of wireless traffic with Tcpdump, Wireshark and Kismet; - How to sniff Wireless, use Wireshark, identify wireless networks with Kismet, map wireless networks with gpsmap, Google Maps, Google Earth.	2
Lecture 2.4	IEEE 802.11 MAC: In Depth - Common capabilities of the IEEE 802.11 MAC, understanding of the architecture and operating of ad-hoc and infrastructure networks, phases of station authentication and association, understanding of the operation and behavior of IEEE 802.1X authentication; - Identifying of capabilities and features of EAP types including PEAP, EAP/TLS, TTLS, EAP-FAST; - Packet framing on wireless networks, understanding of the 802.11 header format and fields, significance of From DS and To DS fields, 802.11 address field ordering and behavior, 802.11 management frames and data encoding, 802.11 management action frames, decoding frames in hex.	2

Chapter/Lecture	Content	Hours
Chapter 3. Wireless Tools and Information Analysis		4
Lecture 3.1	Network Level security of wireless networks. Using of VPNs and Firewalls.	2
Lecture 3.2	Attacking WEP • Introduction to WEP technology, WEP key selection, IV transmission, WEP framing; • Understanding of the XOR truth table; • Introduction to RC4, WEP ICV processing, WEP encryption process, WEP decryption process; • WEP failures including lack of replay protection, weak message integrity check, no key rotation mechanism, also when initialization vector is too short, challenge/response reveals PRGA and key is reversible from ciphertext; • Reliable mechanisms for exploiting and decrypting WEP networks; • Applying WEP failures to other network protocols.	2
Chapter 4. Bluetooth, DECT and ZigBee Attacks		6
Lecture 4.1	Exploiting ZigBee • Introduction to ZigBee, ZigBee use cases and deployment; • Attacker interest in ZigBee and industrial control systems; • ZigBee and IEEE 802.15.4 physical and MAC layer architecture; • ZigBee and IEEE 802.15.4 security mechanisms; authentication and cryptographic controls; • Weaknesses in ZigBee key provisioning and management mechanisms; • Tools for eavesdropping on and manipulating ZigBee networks; • Exploiting ZigBee Over-the-Air (OTA) key provisioning; • Locating ZigBee devices with signal analysis tools.	2
Lecture 4.2	Bluetooth technology and its security features • Bluetooth technology introduction, assessing the Bluetooth protocol stack; • Bluetooth Classic device analysis, procedure for joining a Bluetooth piconet, physical layer components; • Bluetooth Low Energy (4.0) technology analysis, use cases, deployment models and structure; • Bluetooth profiles and application features, Bluetooth security options, leveraging of Bluetooth link authentication and encryption; • Techniques for auditing and identifying of Bluetooth devices, techniques for locating Bluetooth transmitters on Windows and Android platforms; • Bluetooth policy and device configuration best practices.	2

Chapter/Lecture	Content	Hours
Lecture 4.3	Bluetooth vulnerability, attacks and countermeasures - Understanding of Bluetooth pairing, analysis of the Bluetooth authentication exchange and associated protocols, attack on the Bluetooth pairing process, implementation of PIN attacks; - Attack on the Bluetooth E0 encryption algorithm; - Sniffing of Bluetooth networks, hacker techniques for building Bluetooth sniffers, interaction with Bluetooth networks using the Ubertooth One; - Exploiting of Bluetooth non-discoverable mode, discovering of non-discoverable devices; - Exploiting of Bluetooth profile vulnerabilities, audio recording attacks, exploiting of Bluetooth headsets, Bluetooth device impersonation attacks; - Bluetooth device auditing, Bluetooth protocol fuzzing techniques, device enumeration.	2
Chapter 5. Advanced Wi-Fi Attack Techniques		6
Lecture 5.1	Deficiencies in TKIP Networks - TKIP improvements over WEP networks including keying, message integrity checks (MIC), IV sequencing; - QoS deficiencies and the TKIP break, TKIP replay attacks; - TKIP countermeasure DoS attacks; - TKIP plaintext recovery attacks; - Application of TKIP failures to modern cryptographic systems; - Vendor failures exacerbating TKIP flaws.	2
Lecture 5.2	Leveraging of Wi-Fi DoS Attacks - Understanding of the impact of DoS attacks, differentiation of persistent and non-persistent DoS attacks, IEEE 802.11 DoS attack targets including PHY, MAC and client attacks; - Physical medium attacks with the Wave Bubble, common jammers; - IEEE 802.11 MAC attacks, authentication and association floods, deauthentication and disassociation floods, Beacon DS Set DoS; - Impact of IEEE 802.11w and management frame protection and DoS attacks; - IEEE 802.11 medium management techniques, hidden node problem, RTS/CTS medium management, medium reservation attacks, RTS/CTS co-opting; - Client attacks including rogue AP DoS, NULL SSID DoS, 802.1X authentication flood.	2
Lecture 5.3	Wireless Fuzzing for Bug Discovery - Value of protocol fuzzing for fault determination in wireless networks; - Implementation of fuzzing testing operationally, scoping, monitoring, recording and analyzing of results; - When to use fuzzing as a test mechanism; - Strategies for vulnerability disclosure.	2

Wireless and Mobile Security

Chapter/Lecture	Content	Hours
Chapter 6. Wireless Security Strategies and Implementation		6
Lecture 6.1	WLAN IDS Analyst Techniques • Introduction to IDS concepts, differentiation of true positives from false positives, assessment of events of interest; • WIDS deployment models including overlay, integrated, and hybrid deployments; • Techniques for identifying of attacks including signature analysis, trend analysis and anomaly analysis; • Evaluation of attacks through traffic analysis, several examples; • Evaluation of WIDS systems, event aggregation, light bulb deployment, secure communication protocols, intrusion protection services, integration with third-party IDS systems; • WIDS deployment considerations including facility coverage, dwell time, logging fidelity, event storage, trend analysis.	2
Lecture 6.2	Deploying a Secure Wireless Infrastructure • Recommendations for managing an authentication architecture, leveraging the RADIUS protocol for authentication validation, RADIUS data encoding rules, EAP transmitted over RADIUS; • Understanding of the impact of a compromised CA, «evil twin» attack; • Recommendations and preferences for selection of an EAP type, understanding of the advantages and disadvantages of EAP/TLS, PEAP, PEAPv1, PEAPv2, TTLS, EAP-FAST, PEAP-EAP-TLS; • Summary and recommendations for selection of an EAP type.	2
Lecture 6.3	Configuring and Securing Wireless Clients • Management of client certificate trust policies, default Windows root CA trust; • Four techniques for deploying of a new root certificate authority: manual, web-server delivery, scripted web-server delivery, automatic trust with GPO; • Management of client configuration settings with Windows, cached authentication credentials with PEAP on Windows WZC, deploying of GPO settings for preferred wireless network, specification of the configuration and settings of preferred WZC networks, edition and implementation of wireless-specific GPO policies, recommendations for securing PEAP through GPO; • Management of third-part wireless manager tools with the Funk Odyssey supplicant, creation of a custom installer with Odyssey manager.	2
Chapter 7. Protection methods in mobile technologies		6
Lecture 7.1	Protection methods in mobile technologies.	2
Lecture 7.2	WMAN and mobile network security; GSM network security mechanisms; GSM/GPRS vulnerability, attacks and countermeasures.	2
Lecture 7.3	LTE mobile network security features.	2

**2.2. Laboratory works**

Content	Hours	Comments
Lab 1. Wireless Network Mapping	4	
Lab 2. Monitor Network Traffic	4	
Lab 3. WEP and WPS Hacking Technologies	4	
Lab 4. Research of Radio Frequency Wi-Fi Resources in 2.4–2.5 GHz Range	4	
Lab 5. Wi-Fi Network DoS Attacks	4	
Lab 6. Wi-Fi Fuzzing	4	
Lab 7. OTA	4	
Lab 8. Research of Stress Loading of Wireless Network	4	
Lab 9. Crunch Password List Generation	4	
Lab 10. 125 kHz RFID Sniffing [Facultative]	(4)	
Total	36(40)	

2.3. Individual and self-study works

Chapter/ Lecture	Example topics of individual tasks and topics for self-study	Hours
Chapter 1	Intro self-study	20
Chapter 2	Wireless Client Attacks	24
Chapter 3	Introduction to hashing mechanisms, understanding of HMAC hashes. WPA2 key hierarchy architecture and establishment mechanisms.	24
Chapter 4	Exploiting range in Bluetooth networks, Bluetooth attacks including rogue APs, Bluesnarfing, Blueline; exploiting of Bluetooth deficiencies on mobile devices.	24
Chapter 5	Impact of range in a DoS attack, IEEE 802.11 committee stance on DoS attacks, defensive measures.	24
Chapter 6	Management of third-part wireless manager tools with the Funk Odyssey supplicant, creation of a custom installer with Odyssey manager.	24
Chapter 7	CDMA/UMTS mobile network security features.	24
Total		164

2.4. Tests and Exams

Chapter/Lecture		Hours
Lectures	Final test	2
Labs	Assessment lab	8
FINAL	Final Exam	4
Total		14

Course Module 3

Secure Software Development

Purpose

- Learn to write software which is secure, reliable and easy to maintain.

Outcoming skills

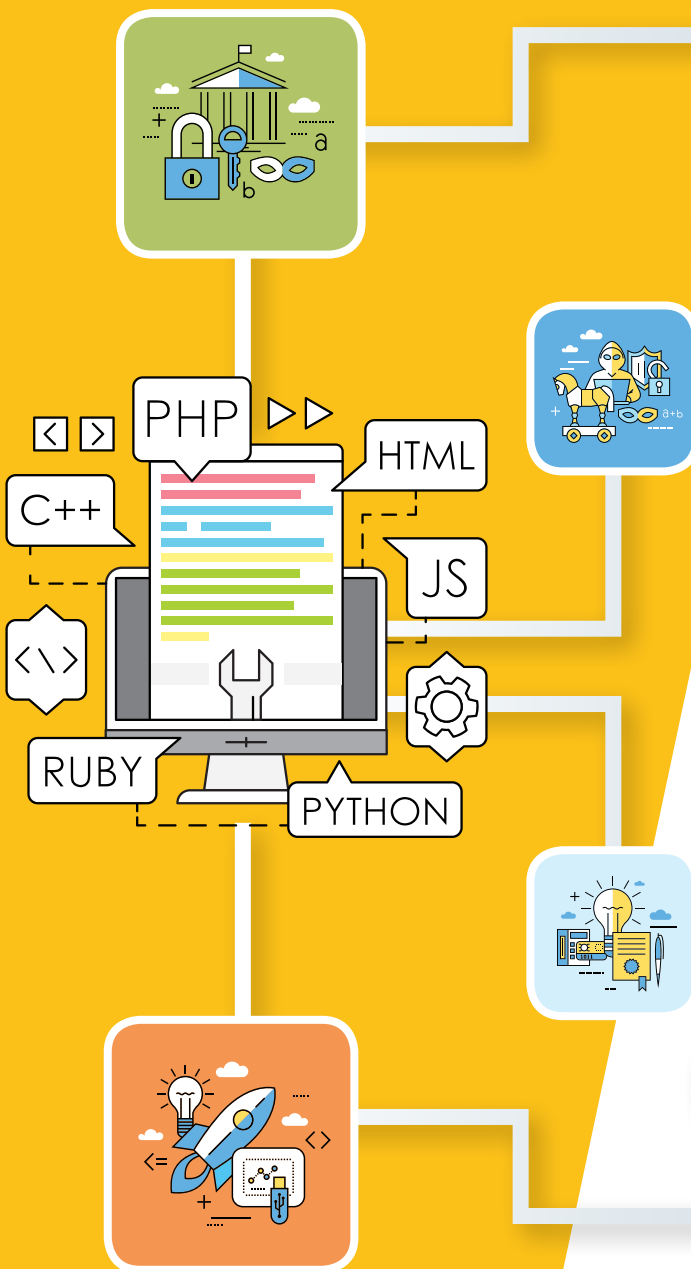
- Perform security assessments.
- Recognize typical programming errors.
- Apply security patterns to programming.
- Write a reliable and maintainable code.
- Debug, analyze code.

Outcoming knowledge

- Software development lifetime.
- Security patterns.
- Security and privacy assessments.
- Safety and reliability of software.
- Defensive programming techniques.
- Good practices in programming.
- Reverse engineering.

Result

- Lectures in total — **15**.
- Laboratory works in total — **9**.
- Workshops in total — **2**.
- Training in total — **1**.





Course name: Secure Software Development

Level: Master Programme

Developers:

- Tomasz Surmacz, Wrocław University of Technology and Science, Poland
- Evgenij Duravkin, Kharkiv National University of Radio Electronics, Ukraine
- Anders Carlsson, Blekinge Institute of Technology, Sweden.
- Andrey Krasov, The Bonch-Bruевич Saint-Petersburg State University of Telecommunications, Russian Federation.

Purpose of the course

The purpose of the course is to teach students the process of developing software which is secure, reliable and easy to maintain. The whole System/Software Development Life Cycle process is described in detail with special attention given to security issues of software development process. Common programming errors and typical vulnerabilities are discussed, as well as testing techniques. High-level approach is also included, focusing on various stages of SDLC, such as software specification, development, testing and maintenance. Labs accompa-

nying the lectures give students opportunity to test their newly gained knowledge in practice. The course is addressed to students who may become programmers/system developers in future, as well as those, who will not develop the software by themselves but rather manage the process as team leaders or managers.

Relevance of the course

The currently developed computer software becomes more and more networked and accessible through different hardware platforms. The shift from a self-contained program to a client-server model or a web-based query-response programming requires even more strict standards for proper programming techniques to maintain the required level of reliability and security of the final result. This applies not only to software related to security issues but also to commonly used applications. Inadequate programmer experience or lack of foresight in some seemingly unrelated simple functions may have major consequences to the overall system security by making the application vulnerable to various kinds of network attacks and exploit scripts.

Two workshops "Train the trainer" were held and the parts of the course were presented during summer schools in Lviv in 2016 and 2017.

Lectures

Lecture 1. SDL requirements

By the end of this module/lesson students will know/
be able to:

- know the principles of Software Development Lifetime;
- know the security requirements;
- know what is the security policy.

Lecture 2. Software risk management

By the end of this module/lesson students will know/
be able to:

- know SRM methodologies;
- know risk prevention, mitigation and correction techniques;

- identify different risks according to the learned taxonomy;
- use methodological framework for software risk management.

Lecture 3. Threat modeling

By the end of this module/lesson students will know/
be able to:

- know the STRIDE methodology;
- recognize typical attack methods on software;
- know various security threats;
- be aware of MITRE Common Attack Enumeration lists;
- be aware of the OWASP project.

Lecture 4. Security patterns

By the end of this module/lesson students will know/be able to:

- know the basic principles of achieving better security in programs;
- know patterns of typical tasks: user identification, authentication, access control, process management, secure execution, file management, etc.;
- know patterns relating to cloud computing and OS administration.

Lecture 5. Code obfuscation

By the end of this module/lesson students will know/be able to:

- know obfuscation: its purpose and techniques;
- know when to use obfuscation;
- recognize different obfuscation methods;
- use the acquired knowledge for malicious software reverse engineering and other applications;

Lecture 6. Code obfuscation 2

By the end of this module/lesson students will know/be able to:

- learn principles of obfuscation transformations;
- learn various obfuscation transformations by examples;
- learn details of obfuscation algorithms.

Lecture 7. Software watermarking

By the end of this module/lesson students will know/be able to:

- know software watermarking principles and purpose;
- know basic concepts;
- learn applications of watermarking and see examples;
- be aware of attack methods on watermarking systems.

Lecture 8. Reverse engineering

By the end of this module/lesson students will know/be able to:

- learn basic principles and concepts;
- know basic reverse engineering techniques;
- be able to analyze static and dynamic models;
- know what tools to use for reverse engineering and how to use them;
- know what is patching.

Lecture 9. Security testing

By the end of the module the students will know/be able to know:

- know types of security testing;
- know and be able to use vulnerability scanners;
- know about penetration testing techniques;
- learn basic principles of security auditing and ethical hacking.

Lecture 10. Best practices in programming

By the end of this module/lesson students will know/be able to:

- principles of good code maintainability;
- importance of proper commenting, naming and indentation;
- examples of good and bad habits in code writing.

Lecture 11. Defensive programming

By the end of this module/lesson students will know/be able to:

- defensive techniques in everyday life;
- defensive programming principles;
- programming style;
- examples of various defensive programming techniques;
- assertions and invariants.

Lecture 12. Programming problems and buffer overflows

By the end of this module/lesson students will know/be able to:

- OWASP project and MITRE;
- common Weakness Enumeration list;
- memory corruption and boundary checks;
- memory management in computer systems;
- buffer overruns, symlinks, race conditions and other sources of problems.

Lecture 13. Synchronization mechanisms in programming

By the end of this module/lesson students will know/be able to:

- necessity of process synchronization;
- atomicity, semaphores, critical sections;
- dining philosophers and other examples;
- resource starvation and deadlocks;
- mutexes and conditional variables in multi-threaded programs.

Lecture 14. Critical sections in Java

By the end of this module/lesson students will know/ be able to:

- definitions;
- atomicity in Java applications;
- semaphore-based solutions;
- reader-writer problem;
- barriers.

Lecture 15. Revision control systems

By the end of this module/lesson students will know/ be able to:

- know the purpose of RCS systems and their advantages;
- learn sample systems: RCS, CVS, SVN, GIT;
- know examples and best practices.

Laboratory works

Lab 1. Secure programming principles

Goal: familiarize with the principles of defensive programming.

Lab 2. Buffer overflows

Goal: test the principles of buffer overflow problems and learn how to avoid them.

Lab 3. Synchronization in distributed programs

Goal: learn how to use semaphores and mutexes in concurrent/multi-threaded programs.

Lab 4. Reverse engineering

Goal: gain skills in using debuggers and code analyzers to discover program hidden details.

Lab 5. Patching

Goal: learn how to reverse-engineer running programs and modify their binary code to change the default behavior.

Lab 6. Java Disassemblers

Goal: learn disassembling techniques for Java byte-code.

Lab 7. Java Obfuscator

Goal: study of the functional capabilities of Java of an obfuscator.

Lab 8. Windows Disassemblers

Goal: study of the functional capabilities of disassemblers (OllyDbg).

Lab 9. Data encryption on a microcontroller

Goal: study programming essentials on microcontrollers, learn the Arduino programming language program hidden details.



Course Module 4

Malware Analysis

Purpose

- Learn to detect and analyse malware.



Outcoming skills

- Detection malware.
- Analysis malware.
- Providing incident response to cyber attacks.
- Preventing infection.



- Types of cyber threats.
- Cyberattack vectors.
- Static analysis methods.
- Dynamic analysis methods.
- Disassembling (x86/ARM).
- Incident response procedure.
- Malware detection technologies.

- Lectures in total — 17.
- Laboratory works in total — 11.
- Practice works in total — 5.
- Workshops in total — 3.
- Training in total — 1.



Outcoming knowledge

Result

Course name: Malware Analysis

Level: Master Programme

Developers:

- *Manager:* Alexander Adamov, Kharkov National University of Radio Electronics, Ukraine.
- *Co-Manager:* Anders Carlsson, Blekinge Institute of Technology, Sweden.
- Vladimir Obrizan, Kharkov National University of Radio Electronics, Ukraine.
- Dmitry Komashinskiy, F-Secure Malware Researcher, Finland; SPIIRAS, Spb., Russian Federation.
- Dragos Illie, senior researcher, BTH, Sweden.

Purpose of the course

The purpose of the course is to provide knowledge in malware analysis for Windows, Android and iOS platforms. The students will obtain practical skills in reverse engineering as well, as in static and dynamic analysis of malicious applications designed for both: ARM and x86 architectures.

Relevance of the course

The course teaches real-world threats analysis, as the majority of modern cyber attacks are performed using malware or, so called, “cyber weapon”. Nowadays, an analysis of such threats requires a special type of education, practically oriented in malware and antivirus technologies.

Three workshops “Train the trainer” have been held.

The course has been adopted by three universities:

- Blekinge Institute of Technology, Sweden.
- The Bonch-Bruевич Saint-Petersburg State University of Telecommunications, Russian Federation.
- Kharkiv National University of Radio Electronics, Ukraine.

Lectures

Lecture 1. History of Computer Threats

By the end of this module/lesson students will know/
be able to:

- Evolution of malicious programs and antivirus solutions.

Lecture 2. Taxonomy of Computer Threats

By the end of this module/lesson students will know/
be able to:

- Main classes of malicious programs and potentially unwanted programs.
- Examples.
- Naming conventions.

Lecture 3. Attack Techniques and Infection Vectors

By the end of this module/lesson students will know/
be able to:

- Typical attack vectors and techniques via: removable drives, browser extensions, P2P networks, etc.
- Examples.

Lecture 4. x86 Disassembly

By the end of this module/lesson students will know/
be able to:

- understand ASM code.
- recognize typical C constructs in ASM code.
- x86 architecture.

Lecture 5. Phishing

By the end of this module/lesson students will know/
be able to:

- Definition of phishing attack.
- Classification of phishing attacks.
- Detection of phishing attacks.

Lecture 6. Online Banking Threats

By the end of this module/lesson students will know/
be able to:

- Overview of online banking threats.
- Classification of online banking attacks.
- Examples.
- Methods of protection.

Lecture 7. Ransomware: Cryptolockers

By the end of this module/lesson students will know/
be able to:

- Definition of Ransomware and Cryptolockers.
- Classification.
- Examples.
- Methods of protection and decryption.

Lecture 8. Botnets

By the end of this module/lesson students will know/
be able to:

- Botnet definition.
- Botnet topologies.
- Communication protocols.
- Examples.
- Methods of detection.

Lecture 9. Mobile Threats in iOS

By the end of the module the students will know/be
able to know:

- The history of viruses and vulnerabilities for Apple iOS.
- Understand security guidelines available in Apple iOS.
- Understand security coding guidelines.

Lecture 10. Mobile Threats in Android

By the end of this module/lesson students will know/
be able to:

- Overview of Android architecture.
- Introduction to Android security.
- Evolution of Android threats.
- Examples of malicious programs for Android.
- Methods of reverse engineering.
- Patching of APK files.
- Exploits.

Lecture 11. Social Networking Threats

By the end of this module/lesson students will know/
be able to:

- Overview of Web threats.
- Overview of Social Networking threats.
- Examples of attacks in social networks.

Lecture 12. Vulnerabilities and Exploits

By the end of this module/lesson students will know/
be able to:

- Definition of a vulnerability and an exploit.
- Web exploits.
- PDF exploits.
- MS Office exploits.
- others.

Lecture 13. Drive-by attacks

By the end of this module/lesson students will know/
be able to:

- Definition of a drive-by attack.
- Typical use cases.

Lecture 14. Rootkits/Bootkits

By the end of this module/lesson students will know/
be able to:

- Overview of Windows kernel.
- Definition of a rootkit.
- Evolution of rootkits.
- Examples.
- Anti-rootkits.

Lecture 15. Targeted Attacks

By the end of this module/lesson students will know/
be able to:

- A targeted attack model.
- Stages of targeted attacks.
- Penetration techniques.
- Reconnaissance techniques.
- Examples.

Lecture 16. Incident Response

By the end of this module/lesson students will know/
be able to:

- An incident response model and procedure.
- A place of malware analysis in the incident response model.

Lecture 17. Threat hunting

By the end of this module/lesson students will know/
be able to:

- Create Yara rule to detect the necessary class of threat.
- Use the hunting capabilities of Virustotal.
- Sort out Virustotal notifications for the triggered threats.

Laboratory works

Lab 1. Malware Detection and Removal

Goal: to familiarize with the principles of operation and management of free signature scanners on the operating system Microsoft Windows.

Lab 2. Malware Static Analysis

Goal: to gain skills in static analysis on malicious programs.

Lab 3. Malware Dynamic Analysis

Goal: to gain skills in static analysis on malicious programs.

Lab 4. Analysis of Exploits

Goal: to gain skills in analysis of exploits.

Lab 5. Reverse Engineering x86/ARM

Goal: to obtain the skills of static program analysis using disassembler in order to detect the presence of malicious content, to explore the basic techniques of disassembling of applications.

Lab 6. Advanced Unpacking

Goal: to gain skills in determining the presence of the packer executable files, to identify the packer; to explore the basic methods of extracting executable files.

Lab 7. Rootkits (optional)

Goal: to gain skills in determining the presence of rootkits in the system; to explore special tools (Anti-Rootkits) for rootkit detection and deactivation; to obtain knowledge of Windows kernel debugging.

Course Module 4

Malware Analysis

Lab 8. Android Malware Analysis

Goal: to gain skills in disassembling and debugging malicious Android applications.

Lab 9. Data Mining with RapidMiner

Goal:

- to familiarize with the basic approach; to build heuristic systems based on Data Mining techniques aimed at malware detection procedures;
- to get practical skills of using the RapidMiner platform for solving tasks from the scope of malware detection and analysis.

Lab 10. Data Mining with Maltego

Goal:

- to learn how to use data mining for malware detection and incident response;
- to get practical skills of using the Maltego visualisation platform for solving tasks from the scope of malware detection and analysis;
- to build an analysis graph with existing entities and transforms (e.g. Virustotal, Paterva) for collecting information about malware and malicious URLs.

Lab 11. Assessment Lab

Goal: to verify the obtained skills and knowledge.

Practice works

Practice work 1. Analysis of host-based attributes for phishing URLs

Finding the patterns based on host-based attributes collected with the help of the WhoIs, GeoIP, and PhishTank services to detect phishing URLs.

Practice work 2. x86 Disassembly: Typical C Constructs

Learning C Constructs in IA32 Assembler.

Practice work 3. x86 Disassembly: Analysis of decryption (deobfuscation) routine used by malware

Learning malware decryption and deobfuscation routines in IA32 Assembler.

Practice work 4. Exploits: Debugging Stack Overrun example

- exploiting stack overflow vulnerability in MS Visual Studio under debugger.

Practice work 5. Debugging Android CrackMe example

- Debugging Android native code to guess the password.



Curricula-Syllabus

Activity	Lectures	Practical works	Laboratory works	Tests	Exams	Classroom hours total	Individual work and self study hours total	Total hours	Credits
Hours	48	10	48	2	4	112	158	270	7.5

1. The purpose and objectives of the course

Aim

Learn to detect and analyse malware.

Learning Outcomes

At the end of the session a successful student will

know:

- ✓ types of cyber threats;
- ✓ cyberattack vectors;
- ✓ static analysis methods;
- ✓ dynamic analysis methods;
- ✓ disassembling (x86/arm);
- ✓ incident response procedure;
- ✓ malware detection technologies;

be able to:

- ✓ detect malware;
- ✓ analyse malware;

- ✓ provide incident response to cyberattacks;
- ✓ prevent infection.

Preknowledge

- ✓ Communication protocols.
- ✓ General programming skills (C/C++, Java).
- ✓ Assembler IA32.
- ✓ Tetworking, Information Security, Web technologies (Java Script).
- ✓ Architecture of Web resources, languages and protocols used in their design and operation (PHP, SQL, HTML, HTTP, IP, TCP, UDP, IP, JAVA, JavaScript).
- ✓ Concepts of vulnerabilities and exploits.
- ✓ Privacy and cyber security laws.

Assessment of acquired skills

- ✓ Ability to detect and analyze Windows and Android malware.
- ✓ Ability to create analysis report.

Course Module 4

Malware Analysis

2. Course content

2.1. Lectures

Lecture	Hours	Comments
Lecture 1. History of computer threats	2	
Lecture 2. Classification of computer threats	2	
Lecture 3. Attack techniques and infection vectors	2	
Lecture 4. Disassembly	6	Demo in IDA, Debugging Android CrackMe
Lecture 5. Phishing	2	
Lecture 6. Online banking threats	2	
Lecture 7. Ransomware: cryptolockers	4	Demo: analysis of available cryptolockers
Lecture 8. Botnets	4	
Lecture 9. Mobile threats in iOS	2	
Lecture 10. Mobile threats in Android	4	
Lecture 11. Social networking threats	2	
Lecture 12. Vulnerabilities and exploits	4	Demo: StackOverrun C program
Lecture 13. Drive-by attacks	2	
Lecture 14. Rootkits/bootkits	2	Modern bootkit examples: NotPetya and Petya ransomware
Lecture 15. Targeted attacks and industrial malware	4	Demo: CozyDuke APT
Lecture 16. Incident Response	2	
Lecture 17. Threat hunting	2	Creating Yara rules to hunt on Virustotal. By the end of this lesson students will know/be able to: • Create Yara rule to detect the necessary class of threat • Use the hunting capabilities of Virustotal • Sort out Virustotal notifications for the triggered threats
Total — 17	48	

2.2. Practice works

Practice work	Hours	Comments
Practice work 1. Analysis of host-based attributes for phishing URLs	2	Finding the patterns based on host-based attributes collected with the help of the WhoIs, GeoIP, and PhishTank services to detect phishing URLs
Practice work 2. x86 Disassembly: Typical C Constructs	2	Learning C Constructs in IA32 Assembler
Practice work 3. x86 Disassembly: Analysis of decryption (deobfuscation) routine used by malware	2	Learning malware decryption and deobfuscation routines in IA32 Assembler
Practice work 4. Exploits: Debugging Stack Overrun example	2	Exploiting stack overflow vulnerability in MS Visual Studio under debugger
Practice work 5. Debugging Android CrackMe example	2	Debugging Android native code to guess the password
Total — 5	10	

2.2. Laboratory works

Labs	Hours	Comments
Lab 1. Malware Detection and Removal	4	Detection of cryptolockers, finding a corresponding decryptor, creation of Yara rules to detect new threats
Lab 2. Malware Static Analysis	4	Usage of static analysis tools such as PEView, PEID, and IDA Disassembler to collect information about a file such as type, format, creation date and time, manufacturer, digital certificate, etc.
Lab 3. Malware Dynamic Analysis	4	Usage of ProcessExplorer, ProcessMonitor, and Wireshark to monitor malicious behavior
Lab 4. Analysis of Exploits	4	JS exploits deobfuscation using Malzilla
Lab 5. Reverse Engineering x86/ARM	4	Analysis of the GoogleDropper malware
Lab 6. Advanced Unpacking	4	UPX, Aspack, WinUpack, FSG, and Morphin packers
Lab 7. Rootkits (optional)	4	Only for WinXP. Usage of Kernel Debugger and WinDBG
Lab 8. Android Malware Analysis	4	Analysis of Android.SimpleLocker cryptolocker and creation of decryption tool
Lab 9. Data Mining with RapidMiner	4	Analysis of PE32 attributes with RapidMiner
Lab 10. Data Mining with Maltego	4	Collecting data (indicators of compromise) with Maltego CE
Lab 11. Assessment Lab	8	Custom — to be created by the teacher and includes tasks from preceding labs covering the following topics: • Windows malware analysis • Android malware analysis • Data mining for malware detection
Total — 11	48	

2.3. Individual and self study works

Chapter/Lecture	Example topics of individual tasks and topics for self study	Hours
Disassembling	IA32 Assembler	40
Disassembling	ARM Assembler	40
Data Mining	Data mining concepts and methods. Usage of RapidMiner.	78

2.4. Tests and Exams

Chapter/Lecture	Control Activities	Hours
Lectures	Final Test 1	2
Labs	Assessment Lab	8
FINAL	Final Exam	4
Total		14

Course Module 4

Malware Analysis

The section 'Additional material and sources' should include:



- 1.** Computer Threats (the introductory course book)
https://drive.google.com/open?id=0B6BXqICLL6H_YUkzVmJldmJrRWM



- 2.** Reverse Engineering for Beginners by Dennis Yurichev, October 2017, 1058 pp., available at http://beginners.re/Reverse_Engineering_for_Beginners-en.pdf



- 3.** Practical Malware Analysis. The Hands-On Guide to Dissecting Malicious Software by Michael Sikorski and Andrew Honig, February 2012, 800 pp., available at <https://www.nostarch.com/malware>



- 4.** Malware Analyst's Cookbook and DVD: Tools and Techniques for Fighting Malicious Code October 2010, 744 pp., available at <http://eu.wiley.com/WileyCDA/WileyTitle/productCd-0470613033.html>



- 5.** Android Hacker's Handbook, 576 pp., 2014, available at <https://www.wiley.com/en-us/Android+Hacker%27s+Handbook-p-9781118608647>



- 6.** iOS Hacker's Handbook, 408 pp., 2012, available at <https://www.wiley.com/en-us/iOS+Hacker%27s+Handbook-p-978111824075>

Course Module 5

Web Security

Purpose

- Learn to detect the vulnerabilities of Web applications and to develop solutions for protection.

Outcoming skills

- Protect web services against cyber attacks. Apply OWASP;
- Perform research on information security of client-server system;
- Protect client & servers on web-platforms.

Outcoming knowledge

- Architecture of web applications protocols (HTTP, SOAP, etc.) — common vulnerabilities and weaknesses, open vulnerability databases;
- Security aspects of applying PHP, Java-Scripts, SQL, etc. languages;
- Web authentication mechanisms and techniques for authentication bypass;
- Application coding errors (SQL Injection, CSS attacks, etc.);
- Existing web-resource vulnerabilities (SQL Injection, brute force, XSS, etc.) and how to deal with them at the design stage and during operation. Design patterns for secure web applications;
- XSS attacks, SQL Injection including vulnerabilities and how to resolve them. Security architecture which is typical for web services;
- Internal and external penetration testing, wireless penetration, PenTesting and Security Tools;
- OWASP.

Result

- Lectures in total — **13.**
- Laboratory works in total — **18.**
- Workshops in total — **2.**
- Trainings in total — **2.**

Course Module 5

Web Security

DESCRIPTION

One of the areas of modern development of information technology is the active use of a variety of web applications: official websites of companies, electronic trading platforms, remote banking systems, public service portals. Web-based systems and cloud services are increasingly replacing enterprise applications based on specialized client software.

The use of web technologies as the means of delivering information largely determines the universal interface between a person and a computer. The importance of web technologies is largely determined by the fact that they are, first of all, the integration technologies of various sources and types of information. Web technologies allow you to create easy-to-learn, easily accessible, extremely cheap, quickly updated the information, dialog and reference systems.

However, the vulnerabilities of web applications make them accessible via the Internet for both users and hackers. Vulnerabilities allow attackers to steal confidential information, make unauthorized changes to data, violate the availability of the application.

Unsurprisingly, web applications create a large number of problems related to information security, since it is the vulnerabilities of Web applications that are one of the main vectors of attacks on information systems. Currently, the problem of ensuring the security of web applications is very relevant.

It is assumed that the students, by the time the course begins, have listened to the courses "Computer networks", "Network technologies" or similar ones, and also have basic ideas about the technologies used in modern web applications (HTTP, SSL / TLS, HTML / CSS, JavaScript / AJAX / DOM).



Course name: WebSecurity.
Level: Master Programme.

Developers:

- *Manager:* Viktor Petrikin, Kaliningrad State University, Russian Federation.
- Sergey Syroezhkin, Kaliningrad State Technical University, Russian Federation.
- Yuriy Lakh, Lviv Polytechnic National University, Ukraine.
- Iryna Sokolianska, Blekinge Institute of Technology, Sweden.
- Anders Carlsson, Blekinge Institute of Technology, Sweden.
- Iveniiia Kuzminykh, Kharkiv National University of Radio Electronics, Ukraine.

Purpose of the course

The purpose of the course "Web Security" is to develop the scientific and educational potential of the trainee in the field of web technology security. The main objective is to provide students with theoretical knowledge and practical skills in the search for vulnerabilities in web applica-

tions, analyze the security of web applications and evaluate the effectiveness of protection measures.

Relevance of the course

Modern web resources have become an integral part of the information assets of the absolute majority of today's organizations, regardless of what they do. Web applications are susceptible to security threats more than other assets because they are available from the Internet at the same time both for users and hackers. The relevance of protecting web applications is only growing because methods and techniques for hacking attacks are constantly developing and improving. Ensuring an adequate level of protection of web-based systems requires highly skilled developers in the field of information security.

The course has been adopted by three universities:

- Blekinge Institute of Technology, Sweden.
- Lviv Polytechnic National University, Ukraine.
- Kharkiv National University of Radio Electronics, Ukraine.

Lectures

Chapter 1. Implementation of protection methods in information exchange protocols

Lecture 1.1. Protocol HTTP

By the end of this module/lesson students will know/
be able to:

- the Hypertext Transfer Protocol;
- HTTP Requests and Responses, Methods and Messages
- cookies;
- status codes.

Lecture 1.2. Protocols HTTP/S, SOAP

By the end of this module/lesson students will know/
be able to:

- HTTPS (Hyper Text Transfer Protocol over Secure Sockets);
- the Secure Sockets Layer (SSL) Protocol;
- Symmetric and Asymmetric Encryption;
- Intercepting Proxies and HTTPS;
- using Simple Object Access Protocol (SOAP).

Lecture 1.3. Privacy protection of client/server application due to Post Office Protocol, Simple Mail Transfer Protocol and Internet Message Access Protocol

By the end of this module/lesson students will know/
be able to:

- the Simple Mail Transfer Protocol (SMTP);
- the Post Office Protocol (POP3);
- the Internet Message Access Protocol (IMAP).

CHAPTER 2. Web-attacks (vulnerabilities)

Lecture 2.1. Web-systems and web-application architecture.

By the end of this module/lesson students will know/
be able to:

- Web Systems and Web Application Architecture;
- Objects of Protection / Attacks;
- Classification of Web Attacks (Vulnerabilities).

Lecture 2.2. Authentication

By the end of this module/lesson students will know/
be able to:

- Brute Force;
- Insufficient Authentication;
- Insufficient Password Recovery (Weak Password Recovery Validation).

Lecture 2.3. Authorization

By the end of this module/lesson students will know/
be able to:

- Credential/Session Prediction;
- Insufficient Authorization;
- Insufficient Session Expiration;
- Session Fixation;
- Session hijacking.

Lecture 2.4. Client-side Attacks

By the end of this module/lesson students will know/
be able to:

- Content Spoofing;
- Cross-Site Scripting, XSS;
- Cross-Frame Scripting (XFS) or Iframe Injection;
- Clickjacking;
- HTTP Response Splitting;
- Cross Site Request Forgery, CSRF;
- JSON hijacking.

Lecture 2.5. Command Execution

By the end of this module/lesson students will know/
be able to:

- Buffer Overflow;
- Format String Attack;
- LDAP Injection;
- OS Commanding;
- SQL Injection;
- SSI Injection;
- XPath Injection.

Lecture 2.6. Information Disclosure

By the end of this module/lesson students will know/
be able to:

- Buffer Overflow;
- Format String Attack;

- LDAP Injection;
- OS Commanding;
- SQL Injection;
- SSI Injection;
- XPath Injection.

Lecture 2.7. Logical Attacks

By the end of this module/lesson students will know/
be able to:

- directory Indexing;
- Web Server/Application Fingerprinting;
- Information Leakage;
- Path Traversal;
- Predictable Resource Location.

CHAPTER 3. Development of protected sites. OWASP projects

Lecture 3.1. Securing web application technologies (SWAT)

By the end of this module/lesson students will know/
be able to:

- error handling and logging;
- data protection;
- authentication;

- input and output handling;
- configuration and operations;
- session management;
- access control.

Lecture 3.2. Introduction to OWASP Testing Project

By the end of this module/lesson students will know/
be able to:

- the OWASP Testing Project;
- principles of Testing;
- Testing Techniques Explained;
- serving Security Test Requirements;
- security Tests Integrated in Development and Testing Workflows;
- security Test Data Analysis and Reporting;
- testing Tools.

Lecture 3.3. Security audit of web applications

By the end of this module/lesson students will know/
be able to:

- basic concepts of auditing Web applications;
- methodology of organization and conducting audit Web applications.

Laboratory works

CHAPTER 1. Injections Flaws

Laboratory work 1. Command injection

Goal: to practice to execute any command on the hosting OS and understand of parameter injection.

Laboratory work 2. Numeric SQL Injection

Goal: to practice how perform Numeric SQL Injection in OS command, scripts, database queries; to understand how Intercept the post request and appending a SQL statement that always resolves to true.

Laboratory work 3. String SQL Injection

Goal: to practice how perform SQL Injection using a string parameter (not an integer), especially data that will used in OS command.

Laboratory work 4. Log spoofing

Goal: to practice how to perform the attack and how the attacks based on fooling the human eye in log files works.

Laboratory work 5. XPATH Injection Attacks

Goal: to practice how to perform XPath Injection attacks; to understand how the attacks works and how using XPATH Injection attacks to elevate the privileges on the web site if the xml data is being used for authentication (such as an xml based user file).

Laboratory work 6. Database Back Door Attacks

Goal: to practice how to create database Back Door Attacks.

To understand how to exploit a vulnerable query to create a trigger that is called by the database management system upon the execution of another database operation like insert, select, update or delete and set other data.

Laboratory work 7. Blind Numeric SQL Injection

Goal: to practice how to exploit this type of Injection (numeric values are used instead of strings) and to understand how the attacks works.

Laboratory work 8. Blind String SQL Injection

Goal: to practice the mechanism of using Blind String SQL Injections in hacker attacks and to understand how the attacks works, the difference from the usual SQL injection.

CHAPTER 2. Cross Site Scripting (XSS)

Laboratory work 1. Stored XSS Attack

Goal: to practice how to perform Stored XSS Attack and to understand how the attacks works.

Laboratory work 2. Reflected XSS Attack

Goal: to practice how to perform Reflected XSS Attack and to understand how the attacks works.

Laboratory work 3. Cross Site Request Forgery (CSRF)

Goal: to practice how to perform Cross Site Request Forgery (CSRF) attacks and to understand how the attacks works.

Laboratory work 4. CSRF Prompt by-pass

Goal: to practice how to perform Cross Site Request Forgery (CSRF) attacks containing multiple requests to

by-pass a scriptable user-prompt and to understand how the attacks works.

Laboratory work 5. CSRF Tocket By-Pass

Goal: to practice how to perform CSRF attacks on sites that use tokens to mitigate CSRF attacks, but are vulnerable to CSS attacks; to understand how the attacks works.

CHAPTER 3. AJAX Security

Laboratory work 1. Client Side Filtering.

Goal: to practice how to discover and get sensitive information and how to defence confidence information on your site and to understand how the attacks works.

Laboratory work 2. DOM-Based cross-site scripting

Goal: to practice how to perform malicious code XSS injection, use the vulnerability to inject malicious code into The Document Object Model (DOM) and how to correct the flaws in the code to address the vulnerability; to understand how the attacks works.

Laboratory work 3. DOM Injection

Goal: to practice how the attacks works for some applications that uses AJAX; to understand how the key validation process works in a system with an activation key.

Laboratory work 4. XML Injection

Goal: to practice how to perform XML Injection attacks and how the attacks works; to understand the behavior of the AJAX application.

Laboratory work 5. JSON Injection

Goal: to practice how to perform JSON Injection Attacks and how the attacks works.

Curricula-Syllabus

Activity	Lectures	Practical works	Laboratory works	Tests	Exams	Classroom hours total	Individual work and self study hours total	Total hours	Credits
Hours	46	—	66	4	4	120	90	210	7

1. The purpose and objectives of the course

Aim

Learn to detect the vulnerabilities of Web applications and to develop solutions for protection.

Learning Outcomes

At the end of the course a successful student will

understand:

- ✓ architecture of web application protocols (HTTP, SOAP, etc.) — common vulnerabilities and weaknesses, open vulnerability databases;
- ✓ security aspects applying PHP, JavaScripts, SQL, etc. languages;
- ✓ web authentication mechanisms and techniques for authentication bypass;
- ✓ application coding errors (SQL injection, CSS attacks, etc.);
- ✓ present vulnerability of web resources (SQL injection, brute force, XSS, etc.) and how to deal with them at the design stage and during operation. Design patterns for secure web applications;
- ✓ XSS attacks, SQL injection, include vulnerability and how to resolve them. Security architecture typical for Web-services;
- ✓ internal and external penetration testing, wireless penetration, Pen Testing and security tools;
- ✓ OWASP.

be able to:

- ✓ protect web services against cyber attacks, apply OWASP;
- ✓ perform research on information security of client-server system;
- ✓ protect client & servers on web-platforms.

Preknowledge

- ✓ communication protocols;
- ✓ general programming skills;
- ✓ networking, information security, web technologies (PHP, Java, etc.);
- ✓ architecture of web resources, languages and protocols used in their design and operation (PHP, SQL, HTML, HTTP, IP, TCP, UDP, IP, JAVA, JAVA Script).
- ✓ safety standards of web services, variety of web vulnerabilities;
- ✓ privacy and cyber security protocols and legislations.

Assessment of acquired skills

Analysis of student's abilities to address web application vulnerability of different types and suggested actions to eliminate and minimize the effects of penetrations.

2. Course content

2.1. Lectures

Chapter/Lecture	Content	Hours
Chapter 1. Implementation of protection methods in information exchange protocols		4
Lecture 1.1. Protocol HTTP	1. The Hypertext Transfer Protocol. 2. HTTP Requests and Responses, Methods and Messages. 3. Cookies. 4. Status Codes.	2
Lecture 1.2. Protocols HTTP/S, SOAP	1. HTTPS (Hyper Text Transfer Protocol over Secure Sockets). 2. The Secure Sockets Layer (SSL) Protocol. 3. Symmetric and Asymmetric Encryption. 4. Interception of Proxies and HTTPS. 5. Usage of Simple Object Access Protocol (SOAP).	1
Lecture 1.3. Privacy protection of client/server application due to Post Office Protocol, Simple Mail Transfer Protocol and Internet Message Access Protocol	1. The Simple Mail Transfer Protocol (SMTP). 2. The Post Office Protocol (POP3). 3. The Internet Message Access Protocol (IMAP).	1
Chapter 2. Web-attacks (vulnerabilities)		32
Lecture 2.1. Web systems and web application architecture	1. Introduction. 2. Web Systems and Web Application Architecture. 3. Objects of Protection / Attacks. 4. Classification of Web Attacks (Vulnerabilities).	4
Lecture 2.2. Authentication	1. Introduction. 2. Brute Force. 3. Insufficient Authentication. 4. Insufficient Password Recovery (Weak Password Recovery Validation).	4
Lecture 2.3. Authorization	1. Introduction. 2. Credential/Session Prediction. 3. Insufficient Authorization. 4. Insufficient Session Expiration. 5. Session Fixation. 6. Session Hijacking.	6
Lecture 2.4. Client-side attacks	1. Introduction. 2. Content Spoofing. 3. Cross-Site Scripting, XSS. 4. Cross-Frame Scripting (XFS) or Iframe Injection. 5. Clickjacking. 6. HTTP Response Splitting. 7. Cross Site Request Forgery, CSRF. 8. JSON Hijacking.	6

Course Module 5

Web Security

Chapter/Lecture	Content	Hours
Lecture 2.5. Command execution	1. Introduction. 2. Buffer Overflow. 3. Format String Attack. 4. LDAP Injection. 5. OS Commanding. 6. SQL Injection. 7. SSI Injection. 8. XPath Injection.	6
Lecture 2.6. Information disclosure	1. Introduction. 2. Buffer Overflow. 3. Format String Attack. 4. LDAP Injection. 5. OS Commanding. 6. SQL Injection. 7. SSI Injection. 8. XPath Injection.	6
Lecture 2.7. Logical attacks	1. Introduction. 2. Directory Indexing. 3. Web Server/Application Fingerprinting. 4. Information Leakage. 5. Path Traversal. 6. Predictable Resource Location.	6
Chapter 3. Development of protected sites. OWASP projects		10
Lecture 3.1. Methodologies of the development of protected sites	1. Securing Web Application Technologies (SWAT). 2. Error Handling and Logging. 3. Data Protection. 4. Authentication. 5. Input and Output Handling. 6. Configuration and Operations. 7. Session Management. 8. Access Control.	2
Lecture 3.2. Introduction to OWASP testing project	About the OWASP Testing Project. Principles of Testing. Testing Techniques Explained. Deriving of Security Test Requirements. Security Tests Integrated in Development and Testing Workflows. Security Test Data Analysis and Reporting. Testing Tools.	6
Lecture 3.3. Security audit of web applications	1. Basic concepts of web application audit. 2. Methodology of organization and conducting of web application audit.	2
Total lectures — 13		46

2.2. Laboratory works

Chapter/Labs	Hours
Chapter 1. Injection Flaws	26
Lab 1. Command injection	2
Lab 2. Numeric SQL Injection	2
Lab 3. String SQL Injection	2
Lab 4. Log spoofing	4
Lab 5. XPATH Injection Attacks	4
Lab 6. Database Back Door Attacks	4
Lab 7. Blind Numeric SQL Injection	4
Lab 8. Blind String SQL Injection	4
Chapter 2. Cross Site Scripting (XSS)	20
Lab 1. Stored XSS Attack	4
Lab 2. Reflected XSS Attack	4
Lab 3. Cross Site Request Forgery (CSRF)	4
Lab 4. CSRF Prompt Bypass	4
Lab 5. CSRF Token Bypass	4
Chapter 3. AJAX Security	20
Lab 1. Client Side Filtering	4
Lab 2. DOM-based Cross-site Scripting	4
Lab 3. DOM Injection	4
Lab 4. XML Injection	4
Lab 5. JSON Injection	4
Total laboratory works — 18	66

2.3. Tests and exams

Chapter/Lecture	Hours
Chapter 2	2
Test 1. Web Application Security	
Test 2. Web Application Threats and Vulnerabilities	2
FINAL	4
Total	8

Additional sources



1. [Free E-Guide] Mitigate Business Risk Strategically With Application Security Management
<https://securityintelligence.com/mitigate-business-risk-strategically-with-application-security-management/>



4. Security · Code · Systems
<https://bitquark.co.uk/blog/>



2. OWASP Online Academy
https://www.owasp.org/index.php/OWASP_Online_Academy



5. Bug Bounty Reference
<https://github.com/ngalongc/bug-bounty-reference/blob/master/README.md>



3. Geekboy | Security Researcher
<http://blog.geekboy.ninja/>



4. Threat Modeling: Designing for Security
https://www.amazon.com/s/ref=nb_sb_noss_2?url=search-alias%3Daps&field-keywords=ISBN-13%3A+978-1118809990&rh=i%3Aaps%2Ck%3AISBN-13%3A+978-1118809990

Official sources (public)



1. Web Application Security, A Beginner's Guide
https://www.amazon.com/Web-Application-Security-Beginners-Guide-ebook/dp/B006DVPWX0/ref=mt_kindle?encoding=UTF8&me=



2. The Web Application Hacker's Handbook: Finding and Exploiting Security Flaws
https://www.amazon.com/s/ref=nb_sb_noss?url=search-alias%3Daps&field-keywords=ISBN-13%3A+860-1404288999%0B&rh=i%3Aaps%2Ck%3AISBN-13%3A+860-1404288999%0B



3. Hacking Exposed Web Applications, Third Edition
https://www.amazon.com/s/ref=nb_sb_noss?url=search-alias%3Daps&field-keywords=ISBN-13%3A+978-0071740647%0B&rh=i%3Aaps%2Ck%3AISBN-13%3A+978-0071740647%0B

Course Module 6

Penetration Testing and Ethical Hacking

Purpose



- Develop and apply the methodology of penetration testing for discovering of vulnerabilities.

Outcoming skills



- Use of tools for ethical hacking
- Prepare a report with findings and recommendations.

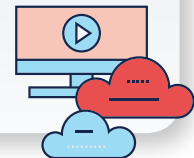
Outcoming knowledge



- Common terms in ethical hacking: *vulnerability, exploit, payload, zero day*.
- Penetration testing methodology.
- How to exploit common vulnerabilities.
- How to perform online and offline password attacks.
- How to perform man-in-the-middle attacks.
- How vulnerability scanner works.

Result

- Lectures in total — **12**.
- Laboratory works in total — **20**.
- VM Kali Linux and Metasploitable — **2**.
- Workshops in total — **2**.
- Trainings in total — **2**.



Description

The purpose of the course "Penetration testing and ethical hacking" is to understand the methodology of penetration testing for discovering of vulnerabilities.

Students should have basic knowledge of OS, DBMS, Web-servers, basic Linux administration skills, communication protocols: IP/TCP/UDP/FTP/HTTP, use of search engines like Google, privacy and cyber security legislations.

At the end of the course a successful student will understand: common terms in ethical hacking: vulnerability, exploit, payload, zero day, penetration testing methodology, how to exploit common vulnerabilities, how to perform online and offline password attacks,

how to perform man-in-the-middle attacks, how vulnerability scanner works.

Course materials include lectures and laboratory work.

Course name: Penetration testing and ethical hacking.

Level: Master Programme.

Developers:

- **Manager:** Alexander Dorofeev, Training Center Echelon, Russian Federation.
- Anders Carlsson, Blekinge Institute of Technology, Sweden.
- Andrian Piskozub, Lviv Politechnik National University, Ukraine.

Lectures

Lecture 1. Vulnerability assessment methodologies

By the end of this module/lesson students will know/
be able to:

- Understand the basic terms of penetration testing.
- Penetration testing process and steps.

Lecture 2. Ethical hacking tools overview

By the end of this module/lesson students will know/
be able to:

- Common tools and approaches for penetration testing.

Lecture 3. Structured approaches for information gathering

By the end of this module/lesson students will know/
be able to:

- Open Source Intelligence methods.
- Types of gathered information and its criticality.
- Discover the sources of information.

Lecture 4. Google for penetration testers

By the end of this module/lesson students will know/
be able to:

- Understand how search engines work.
- Google advanced search operators.
- Google Hacking Database and Google Docs usage.

Lecture 5. Technical information gathering

By the end of this module/lesson students will know/
be able to:

- Gather technical information from open sources.
- Usage of tools for technical information gathering.

Lecture 6. Vulnerability analysis

By the end of this module/lesson students will know/
be able to:

- Types of vulnerabilities.
- Manual search for vulnerabilities.
- Automated search for vulnerabilities.
- Vulnerability analysis tools.

Lecture 7. Basic exploitation. Metasploit Framework

By the end of this module/lesson students will know/
be able to:

- Exploit Databases.
- Local and Remoted exploitation.
- Types of payloads.
- Man-in-the-middle attacks.

Lecture 8. Password attacks

By the end of this module/lesson students will know/
be able to:

- Online and offline password attack types.
- Art of manual password guessing.
- Pass the hash attack.

Lecture 9. Exploitation of Web-applications

By the end of this module/lesson students will know/
be able to:

- Typical structure of Web-application.
- Common web-application vulnerabilities.
- Use the OWASP tools and guidelines.
- Use the web security testing tools.

Lecture 10. Social engineering

By the end of this module/lesson students will know/
be able to:

- Understand social engineering approaches.
- Understand psychological terms and concepts used in social engineering.
- Understand the technical side of social engineering.
- Use the social engineering psychological techniques.
- Use the social engineering technical tools.

Lecture 11. Exploitation using client-side attacks

By the end of this module/lesson students will know/
be able to:

- Understand the concepts of client-side exploitation.
- Use the browser exploitation framework project overview.

Lecture 12. Maintaining Access

By the end of this module/lesson students will know/
be able to:

- Understand the techniques of maintaining access.
- Use Meterpreter.

Laboratory works

Lab 1. Basic configuration of Kali Linux and Metasploitable 2

Goal: configure the private penetration testing laboratory for further sharpening of practical skills.

Lab 2. Usage of Google for OSINT in penetration testing projects

Goal: gain skills in finding of useful information via open sources and in automated way for further attacks on the target of evaluation.

Lab 3. Technical reconnaissance of IT infrastructure

Goal: gain skills in finding of useful technical information needed for further attacks on the target of evaluation.

Lab 4. Using Maltego for reconnaissance

Goal: gain skills in usage of Maltego for structured information gathering.

Lab 5. Port scanning

Goal: gain skills in usage of port scanning tools to get information about network services used on a remote system.

Lab 6. Manual vulnerability assessment

Goal: gain skills in analysis of information found on reconnaissance stage to find suitable exploits via vulnerability databases.

Lab 7. Usage of vulnerability scanners for vulnerability assessment

Goal: gain skills in automation of vulnerability analysis process.

Lab 8. Configuration review

Goal: gain skills in analysis of services configuration to find weak and exploitable points.

Lab 9. Specific scanning tools

Goals: gain skills in working with diverse set of scanning, information gathering and enumeration tools.

Lab 10. ARP-spoofing attack

Goals: gain skills in conducting of man-in-the-middle attacks exploiting the ARP protocol weaknesses.

Lab 11. Use of Metasploit command line for vulnerability exploitation

Goal: gain skills in Metasploit Framework usage.

Lab 12. Use of Armitage for vulnerability exploitation

Goal: gain skills in automation of Metasploit Framework processes using Armitage graphical interface.

Lab 13. Database password attacks

Goal: gain skills in attacks on databases.

Lab 14. Password attacks for different services

Goal: gain skills in usage of different online and offline tools and techniques of password cracking for diverse network services.

Lab 15. Scanning of web application

Goal: gain skills in information gathering about evaluated web application needed for further attacks on it.

Lab 16. Password bruteforcing of web application

Goal: gain skills in attacks on different types of web application authorization schemes.

Lab 17. OS command execution through web server

Goal: gain skills in exploitation of web services where remote code execution is possible.

Lab 18. SQL injection and offline password cracking

Goal: gain skills in SQL injections to get information from hidden areas of web service database and offline password attacks to crack the possibly grabbed credentials.

Lab 19. Exploitation of XSS vulnerability

Goal: gain skills in exploitation of stored, reflected and DOM-based XSS attacks.

Lab 20. Client exploitation with BeEF

Goal: gain skills in social engineering attacks and exploitation of user behavior to infect and control the remote machine.

Lab 21. Rootkit installation and usage

Goal: gain skills in maintaining access to a target system that has already been taken under attacker's control using rootkits.

Curricula-Syllabus

Activity	Lectures	Practical works	Laboratory works	Tests	Exams	Classroom hours total	Individual work and self study hours total	Total hours	Credits
Hours	24	—	38	2	4	80	10	90	3

1. The purpose and objectives of the course

Aim

Understand the methodology of penetration testing for discovering of vulnerabilities.

- ✓ how to perform man-in-the-middle attacks;
- ✓ how vulnerability scanner works;

be able to:

- ✓ use of tools for ethical hacking;
- ✓ prepare a report with findings and recommendations.

Learning Outcomes

At the end of the course a successful student will

understand:

- ✓ common terms in ethical hacking: vulnerability, exploit, payload, zero day;
- ✓ penetration testing methodology;
- ✓ how to exploit common vulnerabilities;
- ✓ how to perform online and offline password attacks;

Preknowledge

- Basic knowledge of OS, DBMS, Web-servers;
- basic Linux administration skills;
- communication protocols: IP/TCP/UDP/FTP/HTTP;
- use of search engines like Google;
- privacy and cyber security legislations.

Assessment of acquired skills

Analysis of student's abilities to perform different security tests.

2. Course content

2.1. Lectures

Chapter/Lecture	Content	Hours
Chapter 1. Introduction to the penetration testing		4
Lecture 1.1. Vulnerability assessment methodologies	• Basic terms. • What is hacking and ethical hacking? • Who are real hackers and what they do. • Penetration testing methodology: OSTMM, ISSAF, etc. • Penetration testing project management. • Hacking tools overview. • Applicable laws. • Dealing with third parties. • Social engineering issues. • Logging. • Reporting. • Scope. Links to other courses.	2
Lecture 1.2. Ethical hacking tools overview	• Ethical hacking workplace: Kali Linux. • Targets: Metasploitable 2, etc. • Portscanners. • Vulnerability scanners. • Exploitation frameworks.	2
Chapter 2. Intelligence gathering		6
Lecture 2.1. Structured approaches for information gathering	• Open Source Intelligence methods. • Structured analytic techniques overview. • Types of collected information: ✓ Business information (financial, clients, suppliers, partners). ✓ Information about technical infrastructure. ✓ Employees. ✓ Discovering sources of the information.	2
Lecture 2.2. Google for penetration testers	• How search engines work. • Google advanced search operators. • Google Hacking Database.	2
Lecture 2.3. Gathering technical information	• Discovering IP-addresses. • Tracerouting. • Using Maltego. • Using theHarvester. • DNS zone transfer. • DNS brute-forcing.	2

Course Module 6

Penetration Testing and Ethical Hacking

Chapter/Lecture	Content	Hours
Chapter 3. Vulnerability analysis		2
Lecture 3.1. Vulnerability Analysis	• Types of vulnerabilities. • Manual search for vulnerabilities. • Automated search for vulnerabilities. • Vulnerability Analysis tools.	2
Chapter 4. Exploitation		4
Lecture 4.1. Basic exploitation. Metasploit Framework	• What is an exploit? • Exploit Databases. • Google for penetration testers: <i>www.exploit-db.com</i>. • Local and remoted exploitation. • Metasploit Framework overview. • Types of payloads. • Man-in-the-middle attacks.	2
Lecture 4.2. Password attacks	• Password attacks: online and offline. • Password hashes. • Art of manual password guessing. • Pass-the-hash attack.	2
Chapter 5. Exploitation of Web-applications		2
Lecture 5.1. Exploitation of Web-applications	• Typical structure of Web-application. • Common web-vulnerabilities. • OWASP projects. • OWASP testing guide overview. • Google Hacking. Google Hacking Database (GHDB). • Web security testing tools: ✓ Web-scanners. ✓ Local proxies. ✓ Fuzzers. ✓ Specialized browsers and browser plugins.	2
Chapter 6. Social engineering		2
Lecture 6.1	• Social engineering. • The Social engineering Toolkit project overview.	2
Chapter 7. Exploitation using client-side attacks		2
Lecture 7.1	• Client-side exploits. • The browser exploitation framework project overview.	2
Chapter 8. Maintaining access		2
Lecture 8.1	• Maintaining access techniques. • Meterpreter usage.	
Total lectures — 12		24

2.2. Practical works

N/A

2.3. Laboratory works

Chapter	Content	Hours
Chapter 1	Introduction to the penetration testing	2
	Lab 1.1. Basic configuration of Kali Linux and Metasploitable 2.	
Chapter 2	Intelligence gathering	4
	Lab 2.1. Using Google for OSINT in penetration testing projects. Lab 2.2. Technical reconnaissance of IT infrastructure. Lab 2.3. Using Maltego for reconnaissance.	
Chapter 3	Vulnerability analysis	6
	Lab 3.1. Port scanning. Lab 3.2. Manual vulnerability assessment. Lab 3.3. Using vulnerability scanners for vulnerability assessment. Lab 3.4. Configuration review. Lab 3.5. Specific scanning tools.	
Chapter 4	Exploitation	10
	Lab 4.1. ARP spoofing attack. Lab 4.2. Use of Metasploit command line for vulnerability exploitation. Lab 4.3. Use of Armitage for vulnerability exploitation. Lab 4.4. Database password attacks. Lab 4.5. Password attacks for different services.	
Chapter 5	Exploitation of Web-applications	6
	Lab 5.1. Scanning of web-application. Lab 5.2. Password brute-forcing of web-application. Lab 5.3. OS command execution through web-server. Lab 5.4. SQL-injection and offline password cracking. Lab 5.5. Exploitation of XSS vulnerability.	
Chapter 7	Exploitation Using Client-side Attacks	1
	Lab 7.1. Client exploitation with BeEF.	
Chapter 8	Maintaining access	2
	Lab 8.1. Rootkit installation and usage.	
Total laboratory works — 8		31

Course Module 6

Penetration Testing and Ethical Hacking

2.4. Individual and self-study works

Chapter/Lecture	Example topics of individual tasks and topics for self-study	Hours
Chapter 1	Creating home laboratory for penetration testing using VirtualBox	2
Chapters 2–7	Recreating labs in home laboratory	8
Total		10

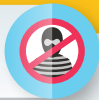
2.5. Tests and Exams

Chapter/Lecture		Hours
Chapter 1	Test 1	0,25
Chapter 2	Test 2	0,25
Chapter 3	Test 3	0,25
Chapter 4	Test 4	0,25
Chapter 5	Test 5	0,25
Chapter 6	Test 6	0,25
Chapter 7	Test 7	0,25
Chapter 8	Test 8	0,25
FINAL	Final Exam	3
Total		5

Course Module 7

Digital Forensics

Purpose



- This course acquaints students with computer forensics. The course is aimed to provide students with the skills of real-world cases investigation by various aspects of digital forensics, including operation systems, network, file system and memory analysis. The presence of such kind of experience will help the students in their future life not only if they work in digital forensics area, but for preventing the traces from unpremeditated damage for their further analysis by forensics specialists.



Outcoming skills



- Select, install, set up and apply required digital forensics tools.
- Find and prevent evidences from unpremeditated damage for their further analysis.
- Select and apply existing practices for collection of evidences in various cases.
- Perform forensics investigation in various operating system environments.
- Conduct basic file system, memory and network forensics investigation.
- Write a forensics report according to the existing templates.

Outcoming knowledge



- General concepts of digital forensics, including brief knowledge in local legal aspects.
- The main features of various forensics tools and methods.
- How to prevent the traces from unpremeditated damage for their further analysis by forensics specialists.
- How to perform forensics investigation in various operating system environments, including Windows, Linux and Mac OS analysis.
- How to perform forensics investigation in various file systems, including FAT, NTFS, Ext and HFS file systems.
- How to perform forensics investigation in computer networks, including traffic dumps and network application features analysis.
- How to perform live forensics investigation, including file system and memory analysis.
- The main principles and features of forensics report writing.

Result



- Lectures in total — **18.**
- Laboratory works in total — **9.**
- Practical works in total — **9.**
- Workshops in total — **3.**
- Trainings in total — **3.**
- Laboratory virtual machine — **1.**



Description

The purpose of the course «Digital forensics» is to provide students with the skills of real-world cases investigation by various aspects of digital forensic, including operation systems, network, file system and memory analysis. The presence of such kind of experience will help the students in their future life not only if they work in digital forensic area, but for preventing the traces from unpremeditated damage for their further analysis by forensic specialists.

It is assumed that the students have basic knowledge of network communication protocols, operation systems basics.

At the end of the course a successful student will understand: general concepts of digital forensic, including brief knowledge in local legal aspects; the main features of various forensic tools and methods; how to prevent the traces from unpremeditated damage for their further analysis by forensic specialists; how to perform forensic investigation in various operating system environments, including Windows, Linux and Mac OS analysis; how to perform forensic investigation in various file systems, including FAT, NTFS, Ext and HFS file systems; how to perform forensic investigation in computer networks, including traffic dumps and network application features analysis; how to perform live forensic investigation, including file system and

memory analysis; the main principles and features of forensic report writing. A successful student will have ability to: select, install, set up and apply required digital forensic tools; find and prevent evidences from unpremeditated damage for their further analysis; select and apply existing practices for collection of evidences in various cases; perform forensic investigation in various operating system environments; conduct basic file system, memory and network forensic investigation; write a forensic report according to the existing templates.

Course name: Digital Forensics.

Level: Master Programme.

Developers:

Managers: Anders Carlsson, Blekinge Institute of Technology, Sweden and Andrey Chechulin, Saint-Petersburg Institute for information and automation of the Russian Academy of science, Russian Federation.

- Benjamin Dorsch, The German Federal Criminal Police Office, Germany.
- Andreas Kuehn, The German Federal Criminal Police Office, Germany.
- Cemil Yesil, PP Westhessen, Germany.
- Victor Völzow, Police Academie Hessen, Germany.
- Dr. Martin Mink, Police Academie Hessen, Germany.

Curricula-Syllabus

Activity	Lectures	Practical works and consultations	Laboratory works	Tests	Exams	Classroom hours total	Individual work and self-study hours total	Total hours	Credits
Hours	36	18+20	36	4	6	120	90	210	7

LECTURE PLAN

Chapter/Lecture	Content	Hours
Chapter 1. Basics of Digital Forensics		4
Lecture 1.1. Introduction	Course introduction: • Course structure. • Pre-knowledge. • Outcomes. Basics of Digital Forensics: • Meaning of Digital Forensics. • History. • Standard operating procedures. • References / literature for self-education. • Report writing & documentation methodologies.	2
Lecture 1.2. General aspects of digital forensics	• Legal aspects of digital forensics. • Standards for identification, collection, acquisition and preservation of digital evidence (e.g., ISO/IEC 27037). • Tool description with a short overview of TSK, xmount, guymager, ewf-tools, etc. • How to set up a forensics workstation.	2
Chapter 2. File System Analysis		12
Lecture 2.1. Data acquisition	Creating of an image for digital forensic investigation: • Description of tools. • Linux commands. Image Formats: • dd (raw format). • ewf (evidence file compressible). Hashing: • Integrity control of evidences. • MD5, SHA1, SHA256 (problem of hash collision).	2
Lecture 2.2. Partitioning	Partitioning: • Features of physical and logical volumes. • Tools for partition analysis. • MBR. • GPT. • RAID overview.	2
Lecture 2.3. Windows file systems. FAT	• FAT main features. • Tools for FAT volume analysis.	2
Lecture 2.4. Windows file systems. NTFS	• NTFS main features (resident / not resident, attributes, etc.). • Tools for NTFS volume analysis.	2
Lecture 2.5. Linux file system	• Ext main features (superblocks, inodes, group descriptor, etc.). • Tools for NTFS volume analysis.	2

Course Module 7

Digital Forensics

Chapter/Lecture	Content	Hours
Lecture 2.6. Mac OS file system	- HFS and HFS+ main features (catalog files, permissions, attributes, etc.). - Tools for HFS volume analysis.	2
Chapter 3. Operation System Analysis		12
Lecture 3.1. Windows artifact analysis (1/2)	- Windows logs. - Prefetch. - Tools for Windows artifact analysis (libevt, libevtx, regripper, etc.).	2
Lecture 3.2. Windows artifact analysis (2/2)	- Registry. - VSS. - Tools for Windows artifact analysis (libvhadow, regripper, etc.).	2
Lecture 3.3. Windows internet application analysis	- Browsers. - Messengers. - P2P artifacts. - Tools for Windows application artifact analysis (SQLite browser, etc.).	2
Lecture 3.4. Windows application analysis	- Files metadata analysis. - Encryption (bitlockers). - Tools for Windows application artifact analysis.	2
Lecture 3.5. Linux artifact analysis	- Configuration. - Log analysis. - History / User homefolder.	2
Lecture 3.6. Mac OS artifact analysis	- Plist. - Log analysis. - User library.	2
Chapter 4. Other Sources for Forensic Investigation		8
Lecture 4.1. Network forensics	- Sniffing of network traffic. - Application layer analysis. - Tools for network forensics (Wireshark, Ettercap, etc.).	2
Lecture 4.2. Live forensics	- How to handle live machines. - Volatile data features and sources. - Acquisition of volatile data (Windows, Linux, Mac OS).	2
Lecture 4.3. SSD forensics	- SSD data acquisition features. - Tools for SSD data acquisition.	2
Lecture 4.4. Memory forensics	- Memory analysis basics. - Volatility. - RAM-dump analysis.	2
Total lectures — 18		36

LABORATORY WORKS

Chapter/Lab	Content	Hours
Chapter 1	Basics of Digital Forensics	4
Lab 1.1. Legal aspects	• Indication of national legal aspects. • A group or a person give presentation devoted to national laws (for example, computer crime / policies / data protection laws / privacy laws).	4
Chapter 2	File System Analysis	12
Lab 2.1. Recovering of deleted partition and rebuilding of Raid	• Usage of diskeditor to change values in the image, so partition will be able to mount automatically. • Reassembling of the separate Raid image with mdadm to rebuild the raid-system. • Report writing.	4
Lab 2.2. NTFS file system analysis	• Recovering of deleted files in NTFS file system. • Report writing.	4
Lab 2.3. HFS file system analysis	• HFS image analysis. • Report writing.	4
Chapter 3	Operation System Analysis	12
Lab 3.1. Windows artifact analysis	• Usage of free tools for Windows examination (libvhadow, regripper). • Investigation of operation system artifacts. • Report writing.	4
Lab 3.2. Windows application artifact analysis	• Analyzing of unknown application behavior. • Students have to develop a methodology for unknown application analyzing and to write a report about their findings.	4
Lab 3.3. Mac OS artifact analysis	• Investigation of Mac operating system artifacts. • Report writing.	4
Chapter 4	Other Sources for Forensic Investigation	12
Lab 4.1. Network forensic analysis	• Wireshark usage. • Usage of case analysis. • Report writing.	4
Lab 4.2. Conducting of memory analysis	• Work with volatility data in memory. • Extraction of encryption keys from memory dump. • Report writing.	4
Total laboratory works — 9		36

Course Module 7

Digital Forensics

PRACTICAL WORKS

Chapter/Practice	Content	Hours
Chapter 1	Basics of Digital Forensics	2
Practice 1.1. First steps in setting up a forensic workstation	• Setting up a Linux-based forensic machine. • ISO-image (Fedora). • Install Linux. • Set up of Internet access. • Download & install CERT-repository. • Installation of the forensic tools.	2
Chapter 2	File System Analysis	6
Practice 2.1. Creation of images in several formats and comparison of them	• How to use a prepared forensic machine. • Imaging of a physical media with write protection.	2
Practice 2.2. Practicing with tools for FAT analysis	• Mounting images (mount/xmount). • File recovery. • Recovering of a deleted file on FAT System using diskeditor.	2
Practice 2.3. Practicing with tools for Ext analysis	• Analyzing of an ext image.	2
Chapter 3	Operation System Analysis	6
Practice 3.1. Practicing with tools for Windows analysis	• Using of free tools for Windows examination (libevt, libevtX, regripper). • Eventlogs and prefetch analysis.	2
Practice 3.2. Practicing with tools for Windows application analysis	• Using of free tools for examination of Windows application internet artifacts (SQLlite-browser).	2
Practice 3.3. Practicing with tools for Linux application analysis	• Analyze a Linux image for traces in configuration, logs and history of Linux operation system.	2
Chapter 4	Other Sources for Forensic Investigation	4
Practice 4.1. Practicing with tools for live forensics	• Cold boot, inception. • Acquisition of volatile data in Windows.	2
Practice 4.2. Practicing with tools for SSD forensics	• Using of free tools for collecting and analyzing of SSD images.	2
Total practical works — 9		18

Cloud Based Remote Security Lab (ReSeLa)

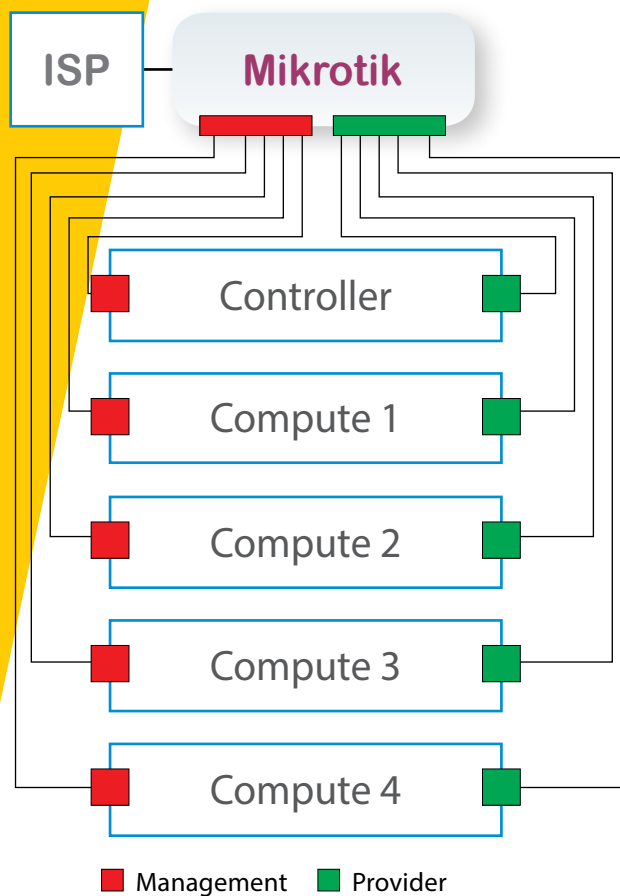


Development was supervised by Anders Carlsson, first year it was Leo Truskans and Martin Baluonis from University of Latvia in Riga that created the test system in 2014. The production system was created by Anders Carlsson and his students from spring 2015 to final version ReSeLa+ in 2017.

Course developer teams made a good work in testing and supported with useful ideas.

ReSeLa is a configurable training environment. Remote Security Lab (ReSeLa) was identified to be a training backbone of the program ENGENSEC. It is an integrated cloud system. The aim of ReSeLa is to support security training and education by selecting appropriate configurations of courses from the ENGENSEC program.

STRUCTURE OF RESELA



On the conceptual level ReSeLa consists of the following three components:

- ReSeLa Panel — front end.
- SDN router — firewall and VPN backend.
- OpenStack — IaaS backend, because it has wide functionality, has well-defined and stable API, is modular in its architecture and is well-supported by the industry.

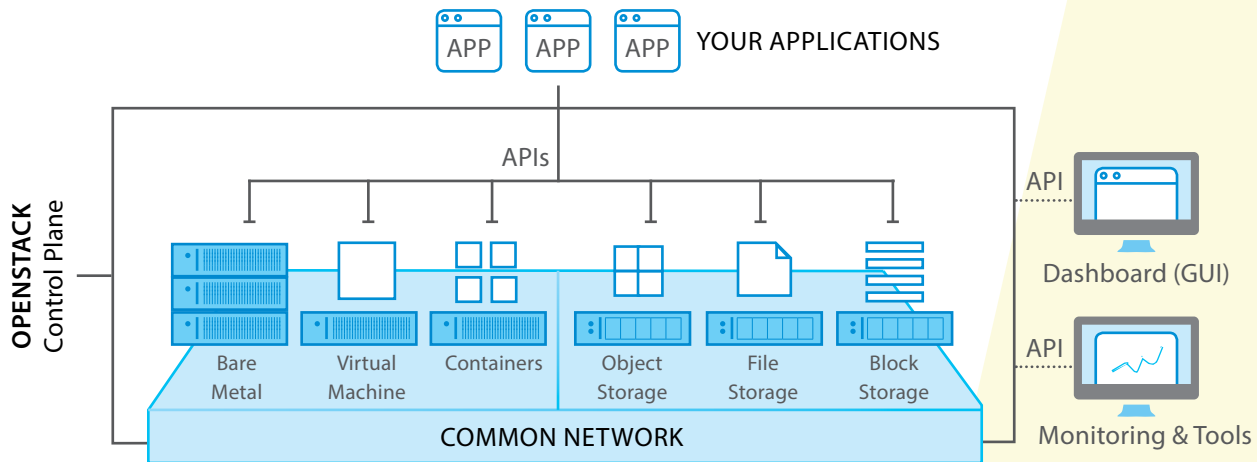
MySQL database is used for its internal data structures.

OpenStack is a cloud operating system that controls storage, large pools of compute, and networking resources throughout the datacenter.

Network policy defines the inbound/outbound access to the Lab network.

Cloud Based Remote Security Lab (ReSeLa)

STRUCTURE OF OPENSTACK



TECHNICAL REQUIREMENTS

The following list specifies the minimum requirements:

1. Mikrotik router (1)

Router specifications:

- ✓ Eight (8) or more Gigabit Ethernet ports.
A Gigabit Ethernet connection is recommended for optimal use. ReSeLa has been tested with the CCR1016-12G and CCR1036-12G-4S models.

Paraphernalia:

- ✓ One (1) Ethernet or fiber cable for Internet connection.
- ✓ Two (2) Ethernet cables for Controller connections.
- ✓ Two (2) Ethernet cables per Compute node.

2. Controller node (1)

- ✓ OS: Ubuntu Server 16.04.
- ✓ CPU: 1 processor.
- ✓ RAM: 12 GB minimum.
- ✓ Storage: 64 GB.

3. Compute nodes (1+)

- ✓ OS: Ubuntu Server 16.04.
- ✓ CPU: 1 processor.
- ✓ RAM: 10 GB.
- ✓ Storage: 100 GB total for the Compute nodes.

4. Public IP and domain name

Make sure to have a public IP-address which can be used for the Internet interface on the MikroTik router. One address is enough for a complete ReSeLa site. You must also have a domain name which can be bound to that public IP-address to make the ReSeLa application available and allow the use of encrypted web communication.

5. Ports open

For ReSeLa to work correctly, make sure that port 25 (TCP) is opened toward the outside, so the ReSeLa application can send mails to users.

ADVANTAGES

ReSeLa provides the following high-level functionality:

- Using and managing Lab with web access.
- Lab provisioning on demand.
- A concept of Virtual Lab as a singular object.
- Customizable network policies, isolated Lab networks.
- Direct connections to Lab instances with the help of VPN access points.
- Network policy defines the inbound/outbound access of the Lab network.
- All users and administrators have user accounts according to their role.

Articles

Overview of the Course in “Wireless and Mobile Security”

*Maryna Yevdokymenko, Department of Infocommunication Engineering,
Kharkiv National University of Radio Electronics, Kharkiv, Ukraine.
E-mail: marina.ievdokymenko@nure.ua*

*Volodymyr Sokolov, Department of Information Cybersecurity,
State University of Telecommunications, Kyiv, Ukraine.
E-mail: vladimir.y.sokolov@gmail.com*

Abstract — This paper provides an overview of “Wireless and Mobile Security” course. The course offers practical study of security issues and features concerning wireless security. The program of the course efficiently interleaves systematic theoretical knowledge and practical work. The theoretical part of the course includes basic information about the architecture of wireless networks, as well as available in this area to modern standards and protection mechanisms built into the equipment for wireless networks. It is also proposed an effective method for integrating a wireless network with the existing network infrastructure, taking into account all aspects of security. More than 50 percent of teaching time is devoted to practical work on the protection of wireless networks.

During the course skills to work with software NetStumbler, Kismet, AirSnort, Aircrack, and other monitoring wireless and network tools will be acquired. Particular attention is paid to the use of the most common tools of audit wireless networks, both commercial, and open source. In conclusion, a comprehensive approach to wireless security will be offered for each wireless technology.

Keywords: *Wireless Network; Wireless Security; Wireless Threats; Arduino IDE; Bluetooth; DECT; ZigBee; Rogue Access Point; RADIUS; WPA; WPA2; AES; TKIP; PEAP; TTLS; EAP-FAST; PololuWixel; Access Point, Spectrum Analyzer.*

INTRODUCTION

Wireless and mobile networks are rapidly extending their capabilities. One of the most beneficial features of wireless networks is that they support user mobility in a convenient way. The downside is that wireless networks are more susceptible to attacks than their wired counterparts. This increased vulnerability mainly stems from the lack of physical connections and the broadcast nature of radio communications. It is, therefore, important to provide appropriate security measures for wireless networks, which ensure the robustness of their operation even in case of malicious attacks.

In practice, many companies and organizations still use and deploy vulnerable wireless gear, often in their default configurations. This is most often due to poor security awareness or a lack of understanding of the risks and ramifications. Home computer users and businesses that lack security are putting themselves at risk of many serious problems, including (but not limited to):

- Identity theft and all the problems caused by it.
- Corruption of important files and data, installing viruses, etc.
- Other people reading your private details and information.
- Stealing of credit card numbers, bank account details and secure information.

Articles

Overview of the Course in “Wireless and Mobile Security”

- Large bills for the download of huge amounts of data by other people.
- Being reported for the download of illegal or copyrighted materials.
- Having your computer used as a part of a network to attack others.
- Handing control of your computer to those with unwelcome agendas.

Early wireless networks heavily leaned on Virtual Private Networks to provide Layer 3 security, which — aside from the additional overhead of encapsulation and challenges of roaming, Quality of Service, client support and scalability — left the IP network vulnerable to attacks.

Today current security features are ineffective, costly, or non-Universal. Home users want something they can figure out that works without having to purchase anything extra. Network administrators also consider cost, but their primary concern has to be making the network available to most of their users while still offering authentication and protection from intruders.

The goal of the course in “Wireless and Mobile Security” is to provide an overview of what is required to provide a secure communication channel in a wireless environment. The focus is on the security techniques available for wireless local area networks and for wireless devices used to access the Internet. This course assists in understanding wireless security requirements and their implementation.

A non-exhaustive list of topics to be taught includes:

- Security basics of the wireless network;
- Wireless data collection and Wi-Fi MAC analysis;
- Wireless tools and Information analysis;
- Client, crypto, and enterprise attacks;
- Bluetooth, DECT and ZigBee attacks;
- Advanced Wi-Fi attack techniques;
- Wireless security strategies, and implementation;
- Protection methods in mobile technologies.

Also, the purpose of the course is to provide knowledge in the security architecture of wireless and mobile communication systems, information threats models, vulnerabilities and protection abilities in wireless networks. After the course in “Wireless Security” the students will obtain practical skills in penetration testing of wireless networks and using various protection methods and will be able to:

- Design a wireless infrastructure;
- Choose wireless configuration;
- Create security policies;

- Set permissions rankings;
- Conduct an audit of the service network;
- Match the current network with other wired and wireless networks;
- Ensure the availability and scalability of the wireless network;
- Protect a mobile device and communication channels against cyberattacks.

So, the main contributions of the course in “Wireless and Mobile Security” are summarized as follows. Firstly, in this course a review of security threats and vulnerabilities at different protocol layers and commencing from the physical layer up to the application layer is presented,. Secondly, the family of security protocols and algorithms used in the existing wireless networks are summarized, such as the Bluetooth, Wi-Fi, WiMAX, and Long-Term Evolution (LTE) standards. Additionally, course provides a review on various wireless sniffers and tools as well as their detection and prevention techniques and proposes some of the open challenges in wireless security.

A. Security Basics of the Wireless Network

Security is especially important for Wi-Fi wireless networks. Hackers can easily intercept wireless network traffic over open air connections and extract information like passwords and credit card numbers. Several Wi-Fi network security technologies have been developed to combat hackers, of course, although some of these technologies can be defeated relatively easily.

As wireless communication and the Internet become truly interoperable, users will want this communication channel to be secure and available when needed. For a message sent using this communication channel, the user expects assurance of:

- Authentication (the sender and receiver are who they say they are).
- Confidentiality (the message cannot be understood except by the receiver).
- Integrity (the message was not altered).

The goal of this course is to provide an overview of what is required to provide a secure communication channel in a wireless environment. The focus is on the security techniques available for Wireless Local Area Networks (WLAN) and for wireless devices (e.g. cell phones and PDA's) used to access the Internet.

Wireless networking provides many advantages but it is also coupled with new security threats and alters the organization’s overall information security risk profile. Although implementation of technological solutions is the usual response to wireless security threats and vulnerabilities, wireless security is primarily a management issue. Effective management of the threats associated with wireless technology requires a sound and thorough assessment of risk given the environment and development of a plan to mitigate identified threats. (Нет источника для этой цитаты)

The wireless networks consist of such basic components: the transmission of data using radio frequencies; access points that provide a connection to the organizational network and/or the client’s devices (laptops, PDAs, etc.); and users. Each of these components provides an avenue for attack that can result in the compromise of one or more of the three fundamental security objectives of confidentiality, integrity, and availability.

Wireless Threats. Wireless networks are exposed to various threats and vulnerabilities. Wireless net-

works are vulnerable because of their open medium nature. The networks have dynamically changing topology and also there is no centralized monitoring and management. Generally, a threat model is used to classify threats or attacks. The attacks are classified into two categories namely passive attacks and active attacks.

Passive Attacks. Passive attacks are those attacks that only snoop the traffic without modifying the data or traffic between the two nodes. In this the attacker does not disrupt the normal routine, but only tries to gather valuable information from snooping the traffic. Examples of passive attacks are Eavesdropping, Man-in-the-Middle attack, Traffic Analysis etc. Active (Изменить на Passive attack?) attack categories are WEP key cracking, evil twin AP and AP phishing, etc.

Active attacks. Active attacks are those attacks that either modifies the data being exchanged or dropping of packets in network. Following are a list of active attacks in WLAN technology: Unauthorized Access, Rogue Access Point, Man-in-the-Middle Attack (MITM), Denial-of-Service, Reply Attack, Session Hijacking.

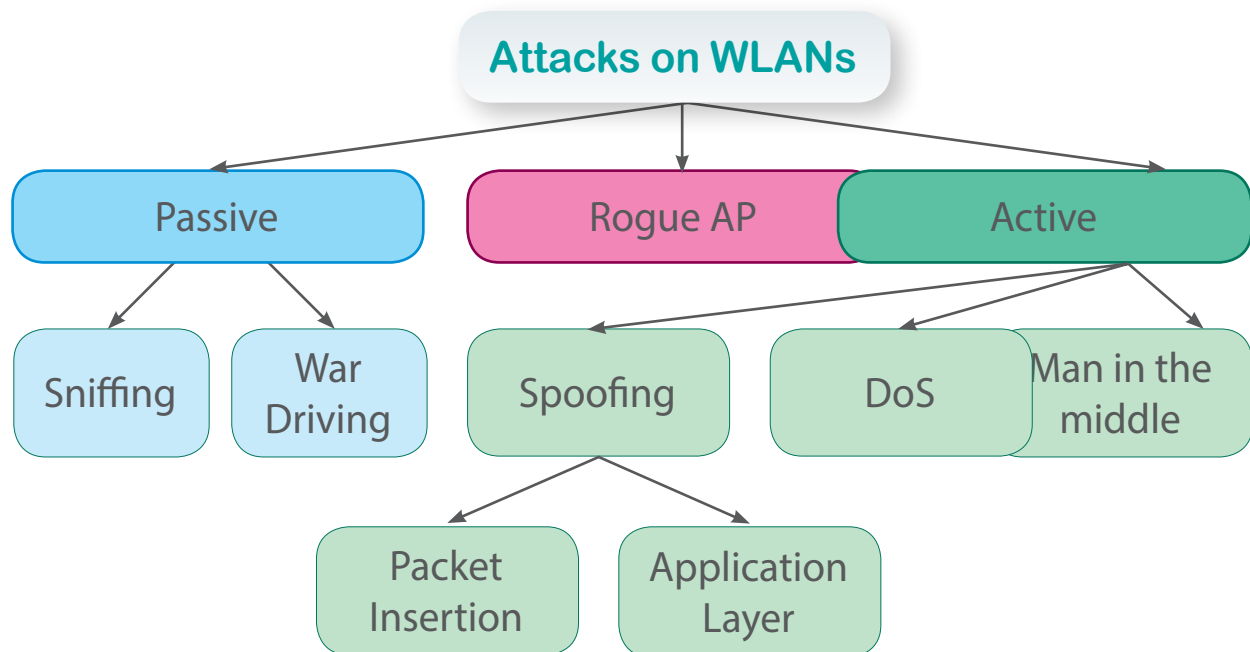


Fig. 1. Attacks on WLANs

Articles

Overview of the Course in “Wireless and Mobile Security”

There are also attacks against the various layers of TCP/IP protocol stack. The attacks can be shown as follows (see Table I).

Table I. Attacks against the various layers of OSI model

Layer	Attacks
Application	DoS, WORM, CSRF, SQL, Injection, Viruses
Transport	Session hijacking, covert channel
Network	Black hole, Byzantine, Wormhole, Rushing, Sybil
MAC layer	Signal jamming, sniffing

Application layer:

The application layer is responsible for running the services. The attacks that are made against this layer are Denial of Service, executing malicious code, worms, etc. The defense mechanism for this layer is proper maintenance of sequence numbers and also maintaining the routing information to avoid denial of service.

Transport layer:

The transport layer is vulnerable to session hijacking which can lead to an attacker taking over the session of a legitimate user and perform malicious actions. Another attack is the covert channel, i.e. hiding information in network header and using a channel for communication that goes undiscoverable. The defense mechanisms for the session hijacking are change of session ID immediately after logging into a session.

Network layer:

The attacks on network layer mainly involve the attacks on the various routing protocols used in ad hoc networks such as Ad hoc On-Demand Distance Vector (AODV) protocol and the Dynamic Source Routing (DSR) protocol. The various attacks taking place on these routing protocols are Black hole, Byzantine, Sybil and Wormhole attacks. These attacks are discussed in detail further.

MAC layer:

The attacks on MAC layer are associated with the channel allocation for the wireless medium. Various attacks such as signal jamming can lead to denial of service. Sniffing can lead to modification of the packets being transferred in the wireless medium. To avoid sniffing packets can be encrypted at the sender side and decrypted at the receiver side. There are various encryption techniques and the standards being used include WPA and WEP.

Multi-layer:

The attacks that occur in any layer of the network protocol stack come under this category. The attacks included are spoofing attacks, denial of service attack etc.

This section mainly focuses on:

- Wireless impact on traditional security approaches;
- Signal exposure threats;
- Common misconceptions in wireless security;
- Wireless LAN and MAN signal leakage;
- Information disclosure threats;
- DoS attacks;
- Rogue AP attacks;
- Wireless protocol deficiencies;
- Anonymity attacks;
- Home user threats.

Thus, in the first part of the course, all existing attacks in wireless networks will be analyzed.

In addition, the following issues will be discussed:

- Wireless standards bodies, role of the WiFi Alliance for interoperability testing, capabilities and features of WPA and WPA2, IETF standards, RADIUS, and EAP protocols.
- Enterprise impact of security-pertinent wireless standards including: 802.11z “Direct Link Setup”, 802.11ac “Gigabit over Wi-Fi”, 802.11af “Wi-Fi in TV White Space”.
- Information about standards bodies work and working group resources.

As a result, the first section of the course “Wireless and Mobile Security” will cover all the principles of wireless LAN organizations, standards and the wireless threats.

Also, the risks and threats using BYOD will be discussed, as it can lead to data breaches and increased liability for the organization.

B. Wireless LAN Assessment Techniques

The second section of the course will present the important question of the wireless LAN assessment techniques.

In fact, companies should conduct regular, periodic security reviews to ensure that changes to the WLAN don't make the system vulnerable to hackers. A review once each year may suffice for low risk networks, but a review each quarter or more often may be necessary if the network supports high risk information (e.g., financial data, postal mail routing, manufacturing control functions, etc.).

When performing a wireless LAN security assessment, consider completing the following steps:

1. Review existing security policies.
2. Review the system architecture and configurations.
3. Review operational support tools and procedures.
4. Interview users.
5. Verify configurations of wireless devices.
6. Investigate physical installations of access points.
7. Identify rogue access points.
8. Perform penetration tests.
9. Analyze security gaps.
10. Recommend improvements.

In the course different methods and analysis for a WLAN audit will be discussed. For example, the application of the method of fingerprinting is associated with the fact that identifying devices connected to a network (i.e., device fingerprinting) has become of critical importance to ensure, among other security services, access control to the network. In the same vein, there has also been a need to understand the type of a device that is connected to a network (i.e., device type fingerprinting). Device fingerprinting seeks to uniquely identify devices on a network without considering existing easily forgeable identifiers (e.g., IP and MAC addresses). On the other hand, device type fingerprinting can be used to determine if a device belongs to a particular cohort. This method includes three categories: OS fingerprinting, host fingerprinting, and device type/driver fingerprinting.

The following WLAN assessment techniques will be described: client post-processing analysis with Kismet XML files, identifying the authentication and encryption options used on the WLAN with Kismet and Wireshark, techniques for mapping the range of indoor and outdoor WLANs, assessing traffic captured in monitor mode for information disclosure, identifying multicast protocols

with MAC analysis, evaluating encrypted traffic and proprietary encryption functions, etc.

After the course the students will have a full understanding of the Rogue AP Analysis (*Fig. 2.*).

Rogue AP Analyse includes:

Defining and understanding rogue networks

How attackers exploit rogue networks

Types of rogue networks

Examples of malicious Rogue AP compromises

Ad-hoc rogue networks

Behavior and spread of the "Free Public Wi-Fi" ad-hoc networks

Windows bridging and the Ad-hoc threat

SOHO devices as a node threat, threat of Windows soft APs

 **Fig. 2. Rogue AP analysis**

Based on the analysis, different techniques will be offered for identifying rogue devices (such as: wired-side AP fingerprinting, wired-side MAC prefix analysis, wireless-side warwalking, wireless-side client monitoring, wireless-side IDS, Nmap rogue AP scripting analysis) and for preventing rogue APs (*Fig. 3.*).

Articles

Overview of the Course in “Wireless and Mobile Security”

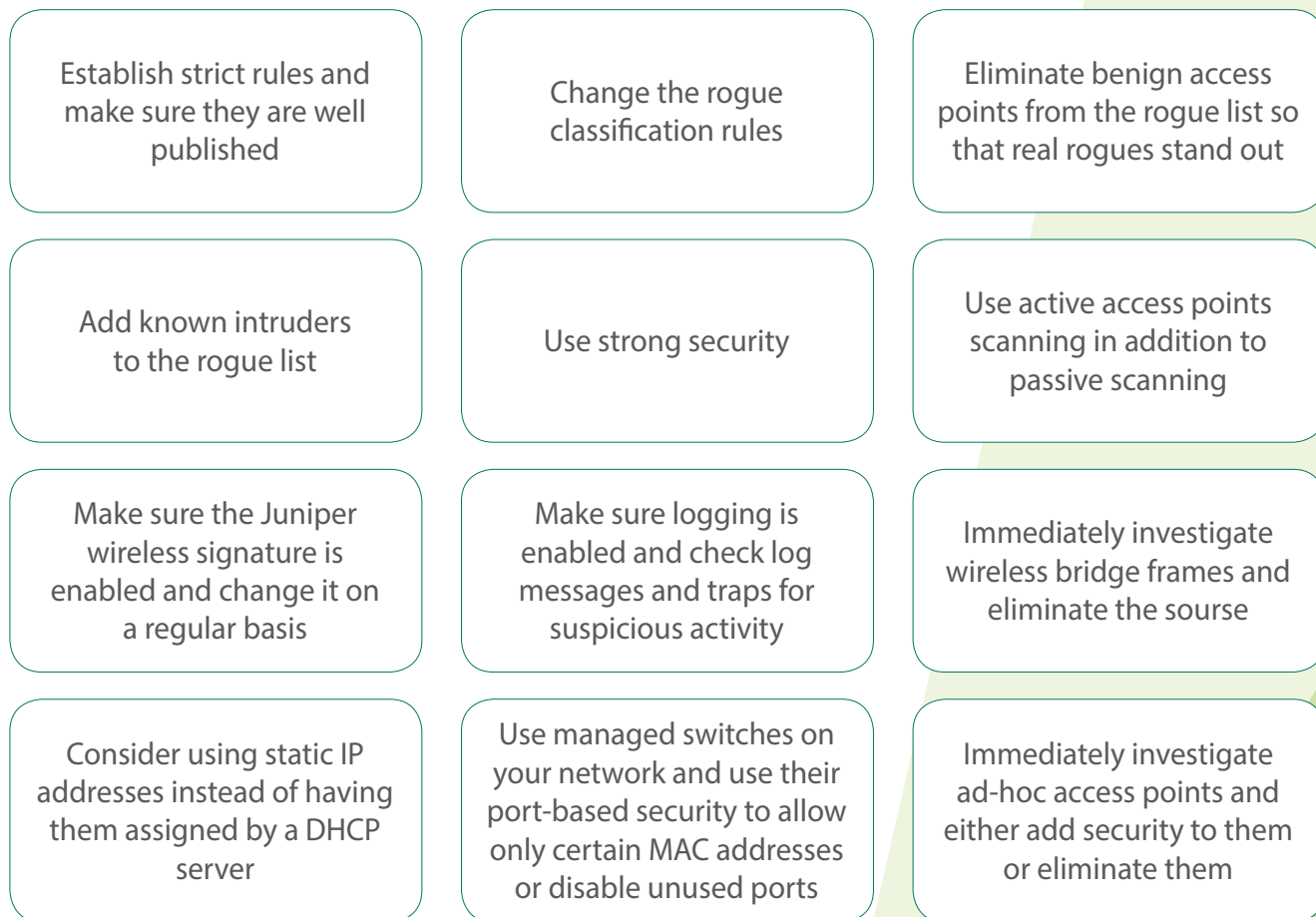


Fig. 3. Preventing rogue access points

Currently, there is already a number of different methods of protection:

- Encryption protocol WEP;
- WPA encryption protocol;
- WPA2 protocol;
- 802.1X security standard;
- Filtering by MAC address;
- Hiding the SSID;
- Denying access to the settings of the access point or router through the wireless network.

However, there is no difficulty in overcoming a wireless network security system. Therefore, it is necessary to use a comprehensive approach to protect a wireless network. For this, the course will consider all methods of protection,

starting with methods from Client, Crypto, Enterprise and finishing Advanced attack.

C. Client, Crypto, Enterprise, and Advanced Attacks

If in a wireless network of small dimensions using standard security methods is usually enough, in large networks, it is necessary to use a more thorough approach to security issues.

For a more detailed understanding of Enterprise attacks, the course will look in detail at the following issues:

- Understanding the risks and challenges of legacy authentication sources, how PEAP addresses this weakness using TLS.

- Understanding TLS tunnel establishment exchange and validation, behavior of PEAP Phase 1 and PEAP Phase 2 connections, identity disclosure in PEAP supplicants.
- Differences between WPA2-PSK and WPA-Enterprise authentication, EAPOL-Key distribution and use, PMK generation and delivery from RADIUS, PTK derivation and key rotation mechanisms.
- Attacks against PEAP networks including authentication attacks, Man-In-The-Middle attacks, EAPOL key-distribution attacks, client-specific attacks.
- Exploiting weaknesses in certificate validation mechanisms in Windows, Apple iOS, and Android platforms.
- Protecting PEAP networks, WZC recommended supplicant configuration properties, mitigating PEAP username disclosure with third-party supplicants, client firewall devices and wireless security recommendations.

Advanced Wi-Fi attack techniques are discussed in more detail. The most compatible wireless hardware configurations nowadays are WPA + WPA2 Mixed Mode, WPA-TKIP, WPA2-AES and WPA-AES, WPA2-TKIP.

D. Wireless Security Strategies and Implementation

The main goals of the course in “Wireless and Mobile Security” is to provide modern methods and techniques of preventing different types of existing attacks. Part of these methods are using Intrusion Detection System (IDS)/Intrusion Prevention System (IPS). The WIDS is the software that detects an attack on a wireless network or wireless system. A network IDS (NIDS) is designed to support multiple hosts, whereas a host IDS (HIDS) is set up to detect illegal actions within the host. Most IDS programs typically use signatures of known cracker attempts to signal an alert. Others look for deviations of the normal routine as indications of an attack.

1. WIDS Deployment Models:

WIDS deployment models can classify by three WIDS model categories: overlay deployment, integrated deployment, and hybrid deployment.

WIDS overlay deployment model (Fig. 4) has next advantages and disadvantages.

Advantages:

1. Ability to deploy a monitoring solution that is separate from the wireless transport provider.
2. The organization can leverage the overlay WIDS product to monitor for the vulnerability, if a vulnerability is discovered in the transport network implementation equipment.
3. Allowing organizations to monitor for the presence of unauthorized wireless devices.

Disadvantages:

1. Cost.
2. Organizations must deploy wireless sensor devices in all areas that require WIDS monitoring.
3. The lack of integration with the transport network can constrain the analysis capabilities of the overlay model.

The sensors passively listen to the events on the wireless network and use various analysis techniques to identify attacks.

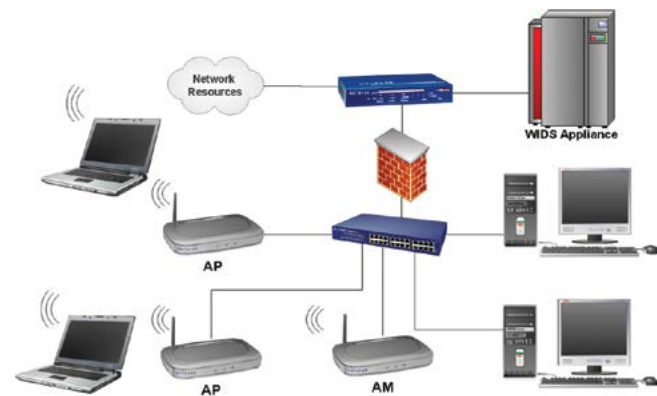


Fig. 4. Overlay WIDS Example, where AM — Air Monitors, AP — Access Point

2. WIDS integrated deployment model:

By using WIDS integrated deployment model the access point operates as both a transport provider, and a WIDS sensor, identifying and reporting attacks.

WIDS integrated deployment model characterizes the following advantages and disadvantages.

Articles

Overview of the Course in “Wireless and Mobile Security”

Advantages:

1. More cost-effective solution for organizations.
2. Has knowledge of dynamic encryption keys.

Disadvantages:

1. The lack of resources.
2. Must provide service and analysis capabilities, restricting the AP's ability to devote available CPU and memory resources to monitoring tasks.
3. Constrain to monitoring the frequency for which it is servicing users.

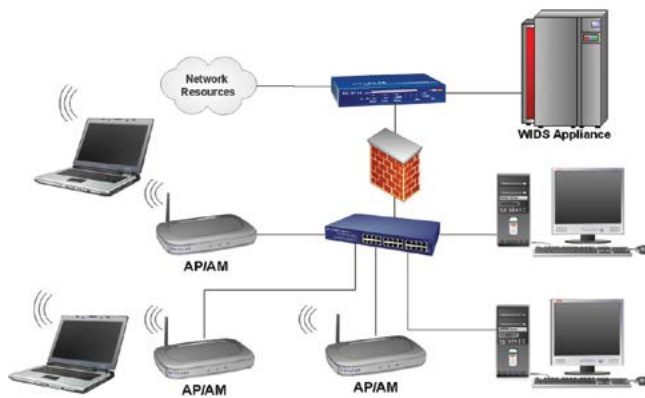


Fig. 5. Integrated WIDS Example, where Dual-purpose AP/ AM devices

3. WIDS Hybrid Deployment Model:

WIDS Hybrid Deployment Model uses both dual-purpose APs and dedicated AMs for intrusion detection and protection. Analysis is performed by a centralized controller similar to what is used with an overlay model, rather than the approach used in an integrated WIDS deployment, where processing is handled by distributed access points.

Advantages:

1. Increased flexibility in deployment.
2. Focused analysis mechanisms.
3. More comprehensive attack detection.
4. Powerful response mechanisms.
5. Using a centralized access controller that integrates an identity-based ICSA-certified firewall.

Disadvantages:

1. Cost.
2. The lack of resources.

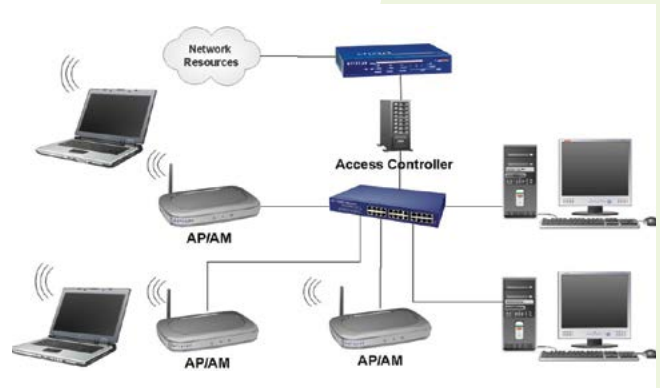


Fig. 6. Integrated WIDS example, where Dual-purpose AP/ AM devices

Methods for detection of attacks in modern Intrusion Detection Systems (IDS) not sufficiently elaborated in terms of the formal model of attacks, and for them it is quite difficult to strictly evaluate properties such as computational complexity, accuracy, etc. Intrusion detection methods are divided into methods for anomaly detection and misuse detection methods. The second type of methods are most of today's commercial systems (Cisco IPS, ISS RealSecure, NFR) — they use a signature (expert) detection methods. There are a lot of academic researches in the field of anomaly detection, but in industrial applications, they are used sparingly and with great caution, as these systems generate a large number of false positives. For expert systems the main problem is the low efficiency of detecting unknown attacks (adaptability). Low adaptability still remains a problem, although such advantages as a low computational complexity and low cost deployment of such systems determine dominance in this area.

4. Methods of Intrusion Detection:

In this section of the course the following criteria were selected for the comparative analysis of intrusion detection methods:

The level of system monitoring. This criterion determines the level of abstraction of the analyzed events in

the protected system, and determines the limits of applicability of the method for the detection of attacks on networks. The following levels are considered in this review:

- HIDS — Host-based IDS.
- NIDS — Network-based IDS.
- AIDS — Application-based IDS.
- Hybrid — A combination of different levels of observer.

Verifiability of the method. This criterion allows to evaluate whether a person can (for example, a skilled operator of IDS or expert) to reproduce the sequence of steps for the adoption of a decision on the presence of an attack by comparing the input and output of IDS. For example, signature-based methods will be considered verifiable, and clustered — not. Verifiability allows the correctness of the method and its implementation at any given time for peer review. The property of this method is important to the operation of the intrusion detection system in real environment as a means of gathering evidence about the attacks.

Adaptive methods. Assessment of the stability of the method to small changes in the implementation of the attacks, which do not change the result of the attack. Adaptability is the only significant advantage of “alternative” methods of detection of attacks before “signature-based”. Lack of adaptability does not allow protection system to respond quickly to unknown attacks and requires the organization of the system some regular updating databases of known attacks, similar to anti-virus systems.

Stability. Output method does not depend on the protected system. The stability problem is acute for statistical methods that analyze the absolute values of the parameters of performance and utilization of network resources and components. Trained in the same network recognizer can be sustained in this network, and unstable in all other networks. This stability will be called local. Since learning is usually an “expensive” procedure because it requires the use of large amounts of resources and time, the number of training procedures should be minimized. Methods for detection of attacks, analyzing the input semantics, are more stable than statistics.

Computational complexity. The complexity of this method of analysis is considered in the acquisition mode, without taking into account the possible preliminary stages of setup and training. This criterion is a key problem for intrusion detection in telecommunication networks. Complexity Estimation is sublinear, linear, quadratic, etc.

Based on the comparative analysis it can be concluded that none of the above open IDS does fully conform to the criteria of the “ideal” IDS. The main disadvantage is the lack of adaptability to unknown attacks and the inability to analyze the behavior of objects in a telecommunications network at the same time, at all levels.

Based on the analysis and identified disadvantage the requirements have been developed for modern IDS [6–9] that can be put into the development of a hybrid adaptive intrusion detection method based on signature analysis and state transition. It requires to:

- Cover all classes of attacks.
- Allow to analyze the behavior of the telecommunications network to be protected at all levels: network, nodes, and the level of individual applications.
- Be adaptable to unknown attacks (using an adaptive method of intrusion detection).
- Scale to telecommunications networks of different classes (from small local networks of class “home office” to large multi-segment and switched enterprise networks, providing centralized management of all IDS components).
- Have integrated mechanisms to respond to an attack.
- Be protected against attacks on IDS components, including interception control or attacks “denial of service”.
- Be open.

So, the role of intrusion detection systems for information security has increased steadily. The cost of systems and the cost of exploitation for commercial IDS is very high. Given the growing popularity of open source software and its significantly lower cost, it appears that the role of free IDS will grow even faster.

Organizations should consider using multiple types of IDPS technologies to achieve more comprehensive and accurate detection and prevention of malicious activity. For most environments, a combination of network-based and host-based IDPSs is needed for an effective IDS solution. Wireless IDPSs may also be needed if the organization determines that its wireless networks need additional monitoring or if the organization wants to ensure that rogue wireless networks are not in use in the organization’s facilities.

The course identifies deploying a secure wireless infrastructure, configuring and securing wireless clients:

- Recommendations for managing an authentication architecture, leveraging the RADIUS protocol for authentication validation, RADIUS data encoding rules, EAP transmitted over RADIUS.

Articles

Overview of the Course in “Wireless and Mobile Security”

- Understanding the impact of a compromised CA, “evil twin” attack.
- Recommendations and preferences for selecting an EAP type, understanding the advantages and disadvantages of EAP/TLS, PEAP, PEAPv1, PEAPv2, TTLS, EAP-FAST, PEAP-EAP-TLS.
- Four techniques for deploying a new root certificate authority: manual, web-server delivery, scripted web-server delivery, automatic trust with GPO.
- Managing client configuration settings with Windows, cached authentication credentials with PEAP on Windows WZC, deploying GPO settings for preferred wireless network, specifying the configuration and settings of preferred WZC networks, editing and implementing wireless-specific GPO policies, recommendations for securing PEAP through GPO.
- Managing third-part wireless manager tools with the Funk Odyssey supplicant, creating a custom installer with Odyssey manager.

Besides security in Wi-Fi network, Protection methods in Bluetooth, DECT, and ZigBee and Mobile technologies are also reviewed in the course.

E. Practical Skills in the Course in “Wireless and Mobile Security”

During the course, students acquire the skills to work with software NetStumbler, Kismet, AirSnort, aircrack and other wireless network monitoring tools. Particular attention is paid to the use of the most common tools of audit wireless networks, both commercial and open source.

The main topics for practice are listed below:

1. Wireless Network Mapping:

The purpose of this practice is to obtain data on wireless networks and visualize the relationships between the elements.

After the study, the students must know and be able to:

- Collect data about wireless devices and their geolocation.
- Write a small Python scripts for data conversion.
- Make a wireless network mapping.
- Build a “client to access point” and “client to probe” relationship graphs.
- Use GPS module in wireless network scanning.

- Parse airodump-ng results and to export its as JSON.
- Make a GPS track on Google maps (for independent work).

2. Monitor Network Traffic:

The purpose of this practice is to collect wireless network data using sniffer and analyze this data on vulnerabilities.

After the study, the students must know and be able to:

- Choose a sniffer for the monitoring of network traffic.
- Restrict on packet processing.
- Collect data from network sniffer.
- Write their own sniffer based on external tools.

3. WEP and WPS Hacking Technologies:

The purpose of this practice is to consider the common methods of hacking Wi-Fi networks.

After the study, the students must know and be able to:

- Conduct attacks on Wi-Fi.
- Set up wireless access point based on known vulnerabilities.
- Test an AP on WEP and WPS vulnerabilities.
- Disable insecure services.

4. Research of Radio Frequency Wi-Fi Resources in 2.4–2.5 GHz Range:

The purpose of this practice is to consider different ways to obtain information about the radiation levels in wireless networks without the use of special spectrum analyzers.

After the study, the students must know and be able to:

- Know ways to obtain information about the energy utilization of channels.
- Know how to get workload of the frequency range.
- Use a mobile phone to receive a list of available wireless networks.
- Determine the workload channel.
- In this practice there are two types of spectrum analyzers used:
- Fourier analyzer (FFT analyzer).
- Analyzers operating in accordance with the heterodyne principle.

In the course a lot of spectrum analyzers for the 2.4–2.5 GHz ISM band will be implemented to connect via USB-interface (FFT analyzer). For example, Ubiquiti AirView2, MetaGeek Wi-Spy, Wi-Detector, as well as on the basis of different sets of debugging (such as TI eZ430-RF2500) or network interface cards (e.g., Atheros AR92xx and AR93xx with Spectral Scan mode [3]). These devices have a number of drawbacks: the price; the difficulty of obtaining data that are usually tied to a particular program; the inability to change the firmware of devices. In addition, there are of course many homemade projects, usually based on TI CC2500 chip and Cypress 693x, as well as modules on their basis.

5. Wi-Fi Fuzzing:

The purpose of this practice is to make fuzz testing of wireless access point and to conduct security analysis of the part of PCI DSS standard that is responsible for the wireless network.

After the study, the students must know and be able to:

- Know different ways of Wi-Fi fuzzing.
- Know wireless part of PCI DSS standard.
- Make a fuzz testing of an AP.
- Security analysis of AP in accordance with PCI DSS standard.

Fuzz testing or fuzzing is a black box software testing technique, which basically consists in finding implementation bugs using malformed/semi-malformed data injection in an automated fashion.

A fuzzer is a program which injects automatically semi-random data into a program/stack and detects bugs. The data-generation part is made of generators, and vulnerability identification relies on debugging tools. Generators usually use combinations of static fuzzing vectors (known-to-be-dangerous values), or totally random data. New generation fuzzers use genetic algorithms to link injected data and observed impact. Such tools are not public yet.

6. Wi-Fi Network DoS Attacks:

The purpose of this practice is to consider DoS ways to attack the Wi-Fi network.

After the study, the students must know and be able to:

- Know types of DoS attacks on the Wi-Fi network.
- Know typical signs of DoS attack.
- Test an AP by DoS attacking.
- Get an answer on the status of running AP.

A denial of service (DoS) occurs when a system is not providing services to authorized clients because of resource exhaustion by unauthorized clients. In wireless networks, DoS attacks are difficult to prevent, difficult to stop an on-going attack; the victim and its clients may not even detect the attacks. The duration of such DoS may range from milliseconds to hours. A DoS attack against an individual station enables session hijacking.

Jamming the air waves. A number of consumer appliances such as microwave ovens, baby monitors, and cordless phones operate on the unregulated 2.4 GHz radio frequency. An attacker can unleash large amounts of noise using these devices and jam the airwaves so that the signal to noise drops so low, that the wireless LAN ceases to function. The only solution to this is RF proofing the surrounding environment.

Flooding with associations. The AP inserts the data supplied by the station in the Association Request into a table called the association table that the AP maintains in its memory. The IEEE 802.11 specifies a maximum value of 2007 concurrent associations to an AP. The actual size of this table varies among different models of APs. When this table overflows, the AP would refuse further clients. Having cracked WEP, an attacker authenticates several non-existing stations using legitimate-looking but randomly generated MAC addresses. The attacker then sends a flood of spoofed associate requests so that the association table overflows. Enabling MAC filtering in the AP will prevent this attack.

Forged dissociation. The attacker sends a spoofed Disassociation frame where the source MAC address is set to that of the AP. The station is still authenticated but needs only to reassociate and sends Reassociation Requests to the AP. The AP may send a Reassociation Response accepting the station and the station can then resume sending data. To prevent Reassociation the attacker continues to send Disassociation frames for a desired period.

Forged deauthentication. The attacker monitors all raw frames collecting the source and destination MAC addresses to verify that they are among the targeted victims. When a data or Association Response frame is observed, the attacker sends a spoofed Deauthentication frame where the source MAC address is spoofed to that

of the AP. The station is now unassociated and unauthenticated, and needs to reconnect. To prevent a reconnection, the attacker continues to send Deauthentication frames for a desired period. The attacker may even rate the limit of the Deauthentication frames to avoid overloading the already congested network. The mischievous packets of Disassociation and Deauthentication are sent directly to the client, so these will not be logged by the AP or IDS, and neither MAC filtering nor WEP protection will prevent it.

Obviously, the primary thing they can do is to force stations (clients) off of a given network, causing a DoS attack. It can also use death attacks to reveal otherwise hidden SSIDs (not included in beacon frames) by disconnecting the clients, and then monitoring for Probe Requests which always contain the SSID.

7. Over the Air (OTA)

The purpose of this practice is to check the firmware updates over the network; to analyze the data transmitted over the network during the upgrade; to investigate the possibility of substitution of transferred firmware.

After the study, the students must know and be able to:

- Know principles of OTA.
- Know weak spots of OTA.
- Create the firmware.
- Provide online firmware.

OTA update is the process of loading the firmware to ESP module using Wi-Fi connection rather than a serial port. Such functionality became extremely useful in case of limited or no physical access to the module.

OTA may be done using:

- Arduino IDE.
- Web Browser.
- HTTP Server.

Arduino IDE option is intended primarily for software development phase. The two other options would be more useful after deployment, to provide module with application updates manually with a web browser or automatically using a http server.

In any case first firmware upload has to be done over a serial port. If OTA routines are correctly implemented in a sketch, then all subsequent uploads may be done over the air.

There is no imposed security on OTA process from being hacked. It is up to developer to ensure that updates are allowed only from legitimate / trusted sources. Once update is complete, module restarts and new code is executed. Developer should ensure that application running on module is shut down and restarted in a safe manner. Chapters below provide additional information regarding security and safety of OTA process.

8. Research of Stress Loading of Wireless Network:

The purpose of this practice is to assemble a sample of the test access point based on single-board computer; to explore one of the security aspects of wireless infrastructure — availability.

After the study, the students must know and be able to:

- Restrict imposed on the wireless infrastructure.
- Know methods of collecting system information in OS Linux.
- Establish basic network services for wireless access points in OS Linux.
- Work with the indicator displays (OLED).

Availability is ensuring that authorized users can access and work with information assets, resources, and systems they need, while providing the required performance. Ensuring availability includes measures to support access to information, despite the possibility of interference, including system failure and deliberate attempts to violate availability. The protection of access to and the capacity of mail service would be an example.

Ensuring availability is to identify possible points of failure and liquidation of these points. Strategies for reducing the negative consequences of failure can be management and technology.

The first step is to identify potential points of failure in the network infrastructure. These mission-critical devices, such as switches and routers, as well as the basic terms of the functioning of servers, such as DNSs, need to be analyzed in terms of a possible failure and its impact on the functioning of the IT capabilities. This is related to risk management — identify and minimize risk.

From the standpoint of availability, the following definitions can be given.

Reliability — the ability of a system or an individual component to perform its required function under certain conditions in the specified time period.

Redundancy — the creation of one or more copies (backup) systems, which are available in the event of primary system failure or the presence of the additional capabilities of the system for the organization of its resiliency.

Resiliency — method of operation, in which the functions of the system component (such as CPU, server, network or database) run redundant components in case of failure or a planned shutdown major component. The ability of a system or component to continue to function normally in case of failure of equipment or software.

It is necessary to analyze the possible points of failure in the following components: data, system components, network topology, routers and switches, some critical services.

9. 125 kHz RFID Sniffing

The purpose of this practice is to consider the work of EM-Marine (EM4100 or EM4102) protocol for 125 kHz RFID and sniff it.

After the study, the students must know and be able to:

- Know EM-Marine protocol.
- Receive data from RFID.
- Sniff and analyze data from RFID.

EM4100 (EM4102, EM-Marine) is a format of contactless radio frequency ID cards by the company EM Microelectronic-Marine (one of the most popular in Ukraine).

They belong to a class of passive RFID cards, because they do not have a built-in power supply. They operate in the frequency range 125 kHz and have a unique number of 40-bit.

They are available in a variety of forms (the most common Clamshell cards, ISO 7810 cards, key rings). ISO-card can be issued in addition to the magnetic stripe identification number, made by stamping, a field for the signature of the cardholder. Personalization ISO-cards is made by thermal printing, screen printing, offset printing. Personalization Clamshell-cards are made using labels that are applied to all the necessary information.

The reader generates a magnetic field frequency of 125 kHz. Once in the magnetic field of the reader, the card receives power and begins to cyclically modulate the magnetic field of the reader signal in which its identifica-

tion code is encrypted. The range of labels is ranging from 5–10 to 60–70 centimeters, depending on the structural elements tags and readers.

Modulation method of the carrier is amplitude. Data encryption — Manchester. 64 bits, including 40 bits of proper unique number, the special synchronization sequence and parity check bits are cyclically transmitted.

The main use is the control of access to the premises and a car park. A distinctive feature of identity cards Em-marine is lower cost compared to other proximity-card standard (e.g., HID or Mifare).

F. Equipment and Software Components for the Workplace.

Many manufacturers of wireless equipment insert algorithms, but the spectrum is scanned only in the location area of AP, thus, they do not consider particularly customer location. AP starts at the clear channel at the place of its location, which to some extent improves the performance of the entire network, but does not make it the optimal.

Usually existing wireless cards are used for collecting the information, but their range of visibility is often limited only by the IEEE 802.11 standard networks (and some maps do not even see “hidden” networks). Because universal method should consider the use of additional independent devices (spectrum analyzers).

For conducting good practice our course offers a spectrum analyzer for separate scheme. The control unit implemented on Arduino Nano v3.0 (with 3.3V supply) and the receiver — on TI CC2500+PA+LNA module with external antenna. Scheme is equipped with OLEDs 0.96” 128×64 I2C and SPI SSD1306 for visualizing the measurement results. The program is written in Arduino IDE and compiled with GCC.

The Fig. 7 shows a diagram of the connection module and power to the device. This device can be operated with a controller that receives data via a USB interface (the virtual COM port). On two screens displayed range of 2400.01–2503.40 MHz with spacing in 405.5 kHz. And Fig. 8 shows an example of display device assembly in a transparent case.

It was found that the control unit memory is not enough (only 2 KB of RAM) to analyze the available channels. Also, this device does not fulfill one of the requirements — must be non-directional antenna, and the half-wave dipole has a distinct polarization.

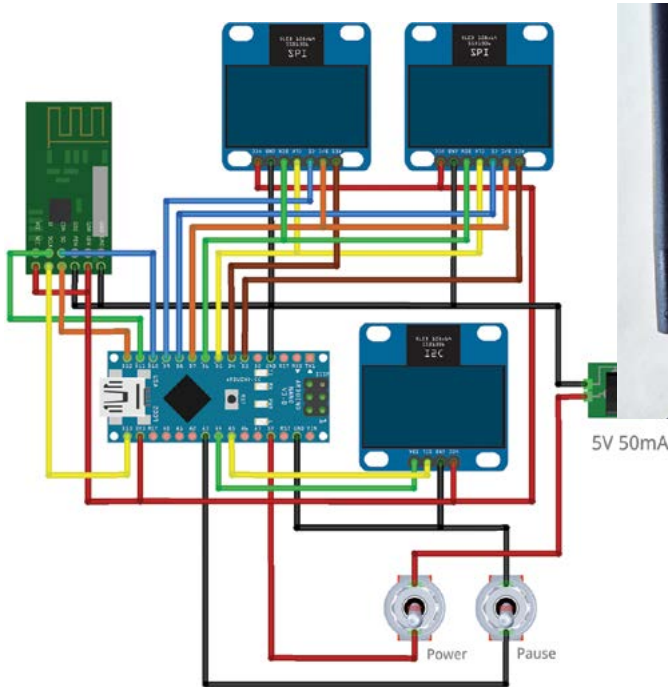


Fig. 7. Schematic diagram of separated receiver and control unit

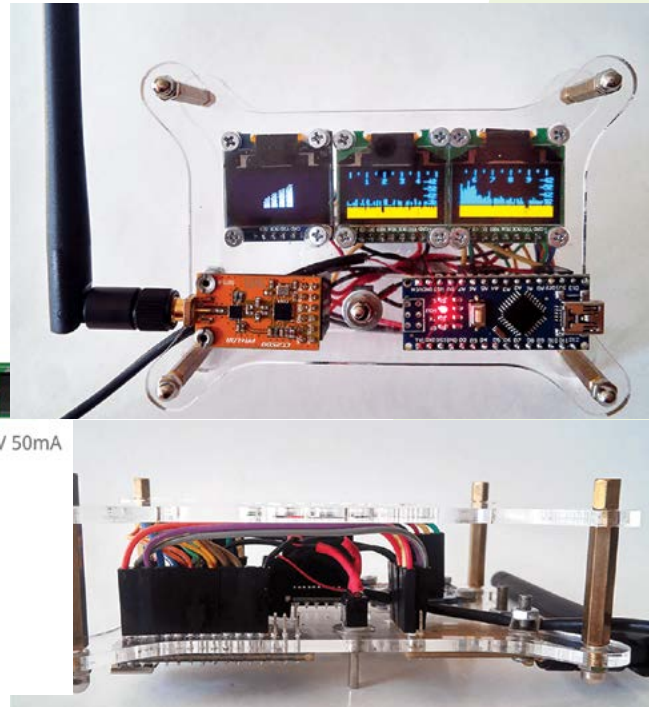


Fig. 8. Separated receiver with external antenna and control unit

To implement the analyzer on a single chip Pololu Wixel has been selected, whose RAM size is 4 KB, the non-directional antenna, five times lower power consumption, almost one and a half times better resolution and SDK with detailed documentation.

Spectrum analyzer on PololuWixel (CC2511F32) with SPI and/or I2C OLED's SSD1306. The spectral width is 2403.47–2476.50 MHz with spacing in 286.4 kHz on two SPI displays. Displays available channels on I2C display. This scheme takes less than 10mA (on 5V).

Prototype can be assembled in a clear acrylic case for Raspberry Pi, but can be built more compactly. Button with a red cap — switch on, and the second one — pause. The left screen displays ZigBee and Wi-Fi channels (not all channels fall within the available range).

So, for practice in the course “Wireless and Mobile Security” the following equipment will be used:

1. Arduino Nano v3.0.
2. TI CC2500+PA+LNA module with external antenna.
3. PololuWixel.
4. OLED's SSD1306 SPI or I2C.
5. Raspberry Pi 3.

6. GPS module (e.g., NEO-6M).

7. UART to TTL adapter with 3.3 V levels (optionally).

8. NodeMCU (ESP-12E).

In the process of learning the course, students will acquire skills to work with different software components: Kismet, AirSnort, Ettercap, Wixel SDK, Arduino IDE, Wi-FiBeaconJam, WIDS, WIPS, Aircrack and other wireless network monitoring tools.

SUMMARY

The world of wireless and mobile devices is evolving day-to-day, with many individuals relying solely on their wireless devices at workplace and at home. The growing use of mobile devices demands that organizations become more educated in securing this growing technology and determining how to protect their assets best. Using case studies and real-world events, it goes on to discuss risk assessments, threats, and vulnerabilities of wireless networks, as well as the security measures that should be put in place to mitigate breaches.

The wireless networks consist of four basic components: the transmission of data using radio frequencies; access points that provide a connection to the organizational network and/or the client devices (laptops, PDAs, etc.); and users. Each of these components provides an avenue for attack that can result in the compromise of one or more of the three fundamental security objectives of confidentiality, integrity, and availability.

Due to the intensive development of wireless networks course in “Wireless and Mobile Security” is very important. The course covers a comprehensive wireless and mobile security overview including the design, planning, installation, and maintenance of wireless network security infrastructures.

The course in “Wireless and Mobile Security” provides knowledge in security architecture of wireless and mobile communication systems, information threats models, vulnerabilities and protection abilities in wireless networks. The students will obtain practical skills in penetration testing of wireless networks by using various protection methods.

REFERENCES

- [1] Wireless Geographic Logging Engine database <https://wgle.net/graph-large.html>.
- [2] **Astapenya V. M., Sokolov V. Yu.** “Modified accelerating lens as a means of increasing the throughput, range and noise immunity of IEEE 802.11 systems,” ICATT’2015 Proceedings of the X Anniversary International Conference on Antenna Theory and Techniques, Kharkiv, Apr. 2015, pp. 267–269.
- [3] “CC2500 Low-Cost Low-Power 2.4 GHz RF Transceiver,” Texas Instruments, 2016, 97 p.
- [4] “Pololu Wixel User’s Guide,” Pololu Corporation, 2015, 67 p.
- [5] **V. Buryachok, G. Gulak, V. Sokolov.** “Miniaturization of Wireless Monitoring Systems 2.4–2.5 GHz Band,” Proceedings of the II International Scientific-Technical Conference on Actual Problems of Science and Technology, Kiev, Dec. 2015, p. 41.
- [6] “nRF24L01 Single Chip 2.4GHz Transceiver Product Specification,” Nordic Semiconductor ASA, Version 2.0, July 2007, 74 p.
- [7] **Graham E., Steinbart P. J.** Wireless Security. 2006.
- [8] Cisco. Dictionary attack on Cisco LEAP vulnerability, Revision 2.1, 19 July 2004.
- [9] CSI. CSI/FBI Computer Crime and Security Survey. 2004.
- [10] **Hopper D. I.** (2002). Secret Service agents probe wireless networks in Washington.
- [11] IEEE 802.11-2007, New York, NY, USA. 2007.
- [12] IEEE 802.11i-2004, New York, NY, USA. 2004.
- [13] **Bellardo J., Savage, S.** 802.11 denial-of-service attacks: Real vulnerabilities and practical solutions. In: Proceedings of the 12th USENIX Security Symposium, Berkeley, CA, USA, USENIX Association, 2003.
- [14] **Aime M. D., Calandriello G., Liyo A.** Dependability in wireless networks: Can we rely on Wi-Fi IEEE Security and Privacy 5, p. 23–29, 2004.
- [15] **Devine C., d’Otreppa T., Beck M.** Aircrack-ng. 2009. <http://www.aircrack-ng.org>.
- [16] **Smith J.** Denial of service: Prevention, modelling and detection. 2007.
- [17] **Glass S., Muthukkumarasamy, V.** A study of the TKIP cryptographic DoS attack. In: ICON 2007: Proceedings of the 15th IEEE International Conference on Networks, New York, NY, USA, IEEE, p. 59–65. 2007.
- [18] **Tews E., Beck M.** Practical attacks against WEP and WPA. In: WiSec ’09: Proceedings of the second ACM conference on Wireless network security, New York, NY, USA, ACM, p. 79–86. 2009.
- [19] IEEE: IEEE 802.11e-2005, New York, NY, USA. 2005.
- [20] **Halvorsen F. M., Haugen O., Eian M., Mjølunes S. F.** An improved attack on TKIP. In: NordSec ’09: Proceedings of the 14th Nordic Conference on Secure IT Systems, Berlin, Heidelberg, Springer-Verlag, p. 120–132. 2009.
- [21] IEEE: IEEE 802.11h-2003, New York, NY, USA. 2003.
- [22] **Harkins D.** Attacks against Michael and Their Countermeasures. IEEE 802.11 Working Group Document 03/211r0, New York, NY, USA. 2003.
- [23] The OpenWrt Project: OpenWrt. 2009. <http://www.openwrt.org>.
- [24] **Malinen J.** Hostapd: IEEE 802.11 AP, IEEE 802.1X / WPA / WPA2 / EAP / RADIUS Authenticator. 2009. <http://hostap.epitest.fi/hostapd>.
- [25] **Zarate J.** Tomato Firmware (2009) <http://www.polarcloud.com/tomato>.
- [26] Cisco Systems Inc.: Enterprise Mobility 4.1 Design Guide, San Jose, CA, USA. 2009.
- [27] **M. Beck.** Enhanced TKIP michael attacks. Retrieved 4 February, 2013, from http://download.aircrack-ng.org/wiki-files/doc/enhanced_tkip_michael.pdf.
- [28] **J. Bellardo and S. Savage.** 802.11 denial-of-service attacks: real vulnerabilities and practical solutions. In Proceedings of the USENIX Security Symposium, 2003.

Articles

Overview of the Course in “Wireless and Mobile Security”



- [29] **K. Bicakci** and **B. Tavli**. Denial-of-service attacks and countermeasures in IEEE 802.11 wireless networks, 2009.
- [30] **A. Stubblefield, J. Ioannidis, A. D. Rubin**. A key recovery attack on the 802.11b wired equivalent privacy protocol (wep). *ACM Trans. Inf. Syst. Secur.*, 7(2), 2004.
- [31] **E. Tews, M. Beck**. Practical attacks against WEP and WPA. In *Proceedings of the second ACM conference on Wireless network security, WiSec '09*, 2009.
- [32] **Y. Todo, Y. Ozawa, T. Ohigashi, M. Morii**. Falsification attacks against WPA-TKIP in a realistic environment. *IEICE Transactions*, 95-D (2), 2012.
- [33] **F. M. Halvorsen, O. Haugen, M. Eian** and **S. F. Mjølunes**. An improved attack on TKIP. In *14th Nordic Conference on Secure IT Systems, NordSec '09*, 2009.
- [34] **B. Harris, R. Hunt**. Review: TCP/IP security threats and attack methods. *Computer Communications*, 22(10):885–897, 1999.
- [35] **J. Huang, J. Seberry, W. Susilo, M. W. Bunder**. Security analysis of michael: The IEEE 802.11i message integrity code. In *EUC Workshops*, pp. 423–432, 2005.

Secure Software Programming Problems

*Tomasz Surmacz, Wrocław University of Technology,
Wybrzeże Wyspiańskiego 27, 50-370, Wrocław, Poland.
E-mail: tomasz.surmacz@pwr.wroc.pl*

SECURE PROGRAMMING

Many security problems come from improper configuration or unwanted interaction between different programs, but many of them are actually consequences of errors in software development — laziness of programmers, lack of experience, tight schedules and deadlines that have to be met, or just the underestimated role of software testing.

The “Top 25 most dangerous software errors” document [5] lists the most dangerous programming problems considered today that may lead to serious software vulnerabilities. The short summary of these errors is shown in table I. It includes various aspects of programming — from specifics of various programming languages to web programming and constructing database queries, so the given scores are influenced mostly by the impact on the overall security, which puts a clear stress on WWW access and parts of programming techniques which are web-related. Another notable web-oriented document is the OWASP Testing Guide [6]. Nevertheless, it is equally important to avoid the well-known problems when writing programs that will be compiled and run in any operating system, no matter if they are WWW-accessible or run as standalone programs.

When considering shell programming and writing standalone programs, the most common errors include problems that allow exploiting setuid programs, such as:

- buffer overflows;
- signal disposition and process group setup;
- `printf()` format string attacks;
- relative path in `exec()`;
- `/tmp` or other publicly writable directory race conditions.

There are many general rules for safe programming and techniques. One of the very basic ones requires processes to operate on the least possible level of required privileges and dropping them whenever they are not needed. Other ones assume that both programmers and users can make errors, and it is the responsibility of a computer program to deal properly with all kinds of problems with data sanity checks, for example by checking various assertions throughout the execution of the program.

The *defensive programming* techniques cover the whole process of software design that covers such aspects, as writing down the required functionality, splitting the whole program into functional objects, reducing the design complexity, writing and testing pieces of software as they are written, writing usable documentation, and finally — writing software that is reusable, bug-free, predictable and running properly in spite of its unforeseeable usage.

Proper software writing techniques include:

- dropping extra privileges when they are not needed;
- ‘black list’ of functions that should never be used in suid programs;
- adhering to the specification of the program;
- keeping in mind the re-usability of code and proper documentation;
- proper testing of all components of a written program.

Table I. Top 25 programming errors according to [5]

Rank	Score	ID	Name
1	93.8	CWE-89	Improper Neutralization of Special Elements used in an SQL Command "SQL Injection"
2	83.3	CWE-78	Improper Neutralization of Special Elements used in an OS Command "OS Command Injection"
3	79.0	CWE-120	Buffer Copy without Checking Size of Input "Classic Buffer Overflow"
4	77.7	CWE-79	Improper Neutralization of Input During Web Page Generation "Cross-site Scripting"
5	76.9	CWE-306	Missing Authentication for Critical Function
6	76.8	CWE-862	Missing Authorization
7	75.0	CWE-798	Use of Hard-coded Credentials
8	75.0	CWE-311	Missing Encryption of Sensitive Data
9	74.0	CWE-434	Unrestricted Upload of File with Dangerous Type
10	73.8	CWE-807	Reliance on Untrusted Inputs in a Security Decision
11	73.1	CWE-250	Execution with Unnecessary Privileges
12	70.1	CWE-352	Cross-Site Request Forgery (CSRF)
13	69.3	CWE-22	Improper Limitation of a Pathname to a Restricted Directory "Path Traversal"
14	68.5	CWE-494	Download of Code Without Integrity Check
15	67.8	CWE-863	Incorrect Authorization
16	66.0	CWE-829	Inclusion of Functionality from Untrusted Control Sphere
17	65.5	CWE-732	Incorrect Perm. Assignment for Critical Resource
18	64.6	CWE-676	Use of Potentially Dangerous Function
19	64.1	CWE-327	Use of a Broken or Risky Cryptographic Algorithm
20	62.4	CWE-131	Incorrect Calculation of Buffer Size
21	61.5	CWE-307	Improper Restriction of Excessive Authentication Attempts
22	61.1	CWE-601	URL Redirection to Untrusted Site "Open Redirect"
23	61.0	CWE-134	Uncontrolled Format String
24	60.3	CWE-190	Integer Overflow or Wraparound
25	59.9	CWE-759	Use of a One-Way Hash without a Salt

BUFFER OVERFLOW

Most of currently used programming languages provide support for tables or other structures whose elements can be iterated, as well as local variables and recursion, which impose a certain model of architecture of the memory usage in a program. Also many languages that are compiled (such as assembler, C, C++, Pascal, etc.) enforce strict memory allocation using various functions, but without the actual boundary checking for the allocated data.

No matter what the programming language is used, the architecture and a memory layout of the running application are similar for all operating systems. From the point of view of a processor, all the physical memory is divided into pages of the same size (typically 4, 8 or 32 kB each, depending on the operating system and the memory management unit used). These pages are then used to construct segments that are available for running processes. Each process has access to three different segments: the program code, the data and the stack. Segments are an integral part of process context data, so whenever there is a context switch in a processor (when a multitasking CPU switches from executing of one process to another one) all segment registers of the processor are loaded with values associated with the currently running process. This is shown in fig. 1.

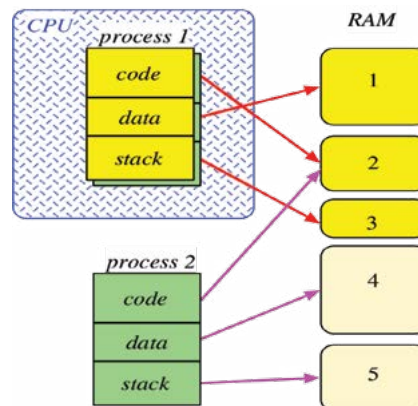


 **Fig. 1.** Segments in a context switch

The same mechanisms are used to provide virtual memory to processes. Virtual to physical address mappings are provided by segment tables. Each segment table contains a list of physical memory pages that belong to that segment. Two different processes may use the same virtual addresses and actually access different regions of memory, as their segment tables are independent of each other and may point to different physical memory pages through the same virtual address (see fig. 2). On the other hand, a memory segment may be shared between different processes, even if they are referred to by different virtual addresses.

What is more important, the stack is also used for storing the return address from the function (so that the control can be transferred back to the calling function when the called one finishes). Local data (variables declared inside a function) is also allocated on the stack just as the function starts execution in a process called stack-frame setup. The function frame is prepared as shown in fig. 3, and the assembly code equivalent of the listed program is shown in fig. 4.

If you are not familiar with assembly language, here are some basic facts that will help you understand the code: The % sign denotes the processor registers, **eax**, **edx**, **esi** and **edi**, are general purpose registers, **esi** and **edi** are also typically used to pass parameters when

calling a function and **eax** — to return the function value. **rsp** is the stack pointer, **rbp** is the “base pointer”. The **mov** (quick, byte, word, long) moves the data from its first argument to the second, the **push** command copies its argument to the location pointed by the stack pointer and automatically decreases **rsp**, and **pop** retrieves the previously pushed value from the stack. The notation **\$64** means the decimal number 64, and **-52(%rbp)** means a memory location addressed by using the contents of register **rbp** with a displacement of -52 bytes (i.e. **rbp-52**). The rest of opcodes, such as **sub**(tract), **j(u)mp**, **c(o)mp**(are) or **ret**(urn) should be self-explanatory.

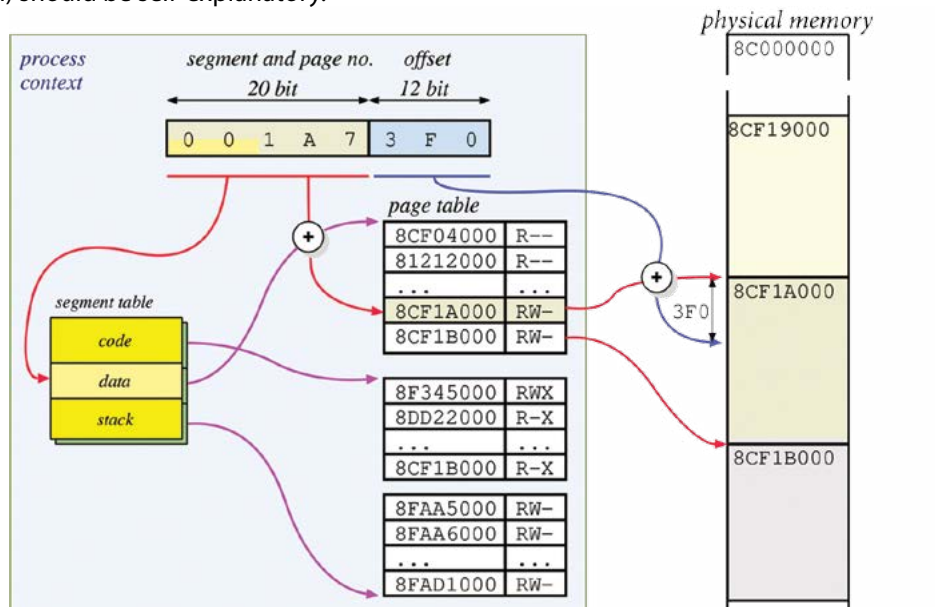


Fig. 2. Virtual memory management through memory pages and segments

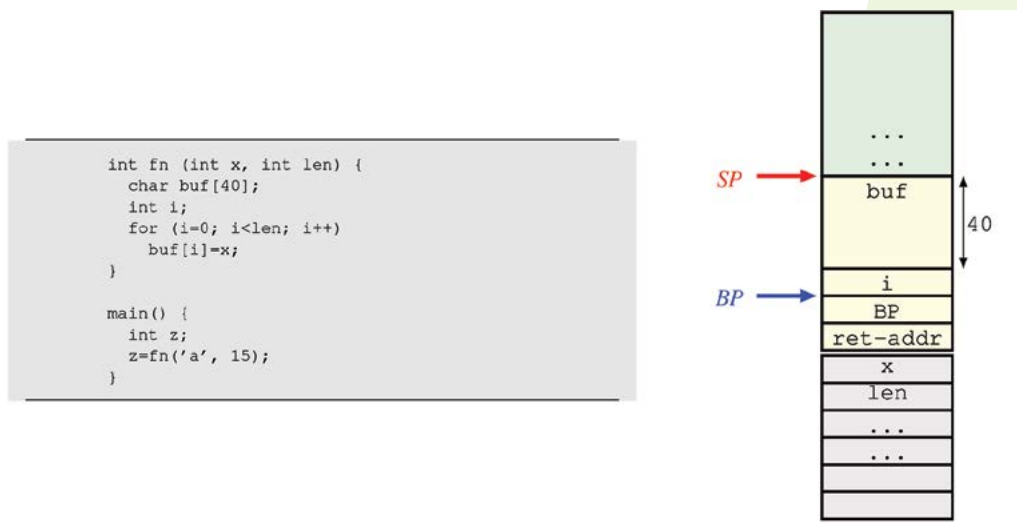


Fig. 3. Stack frame of a function

```

1  fn:
2      pushq   %rbp
3      movq    %rsp, %rbp
4      subq    $64, %rsp
5      movl    %edi, -52(%rbp)      # local copy of x
6      movl    %esi, -56(%rbp)      # local copy of len
7      movl    $0, -4(%rbp)         # i=0
8      jmp     .L2
9  .L3:
10     movl    -52(%rbp), %eax
11     movl    %eax, %edx
12     movl    -4(%rbp), %eax        # i
13     cltq
14     movb    %dl, -48(%rbp,%rax)
15     addl    $1, -4(%rbp)         # i++
16  .L2:
17     movl    -4(%rbp), %eax        # i <
18     cmpl    -56(%rbp), %eax      # len ?
19     jl      .L3
20     popq    %rbp
21     ret
22  main:
23     pushq   %rbp
24     movq    %rsp, %rbp
25     subq    $16, %rsp
26     movl    $15, %esi            # parameter: 15
27     movl    $97, %edi            # parameter: 'a'
28     call    fn                   # call function
29     movl    %eax, -4(%rbp)
30     leave
31     ret
  
```

Fig. 4. Assembly-code representation of program from fig. 3

Throughout the function the **SP** (or **ESP**) register may change its value back and forth as values are pushed onto the stack, so instead the **BP** (or **EBP**) — base pointer register is used to point to some fixed address on the stack and refer to all the function parameters and local variables by addresses relative to this base value. In the process of frame setup, first — in line 2 the old value of **BP** is pushed to the stack (so it can be restored when the function finishes) and the current value of **SP** is copied to **BP**. Then the space is allocated for local variables by subtracting the required amount of space from the **SP** register (line 4). From that point on, local variables may be addressed by using **BP** register with negative offsets, and parameters passed to function are addressed by **BP** with positive offsets such as **BP+8** or **BP+12**. Memory addressed by **BP+0** holds the old value of **BP** and **BP+4** — the return address.

Between labels `fn` and `.L3` the stack frame is setup, then the main loop is executed between `.L3` and `.L2`. At `.L2` the loop condition is tested and either the loop is continued at `.L3`, or the old value of `BP` is fetched from the stack (line 20) and the function returns. You can also see a similar stack frame setup code in the main function of the program in lines 23–25 and 30.

Now consider what happens if the function is called with the `x` parameter set to the value larger than the size of the allocated buffer, e.g. 60 (see fig. 5). The data copying loop in this case will happily fill the whole buffer, then continue with overwriting memory locations immediately after the buffer. This includes the old `BP` value stored on stack, as well as the return address (and parameters passed to the function). When the loop finally finishes and the CPU executes the `ret` instruction, the return address is already smashed, so the program control is transferred to some unknown location (based on value used to fill in the buffer). If we are lucky, this will point outside the address space available to the running process, and the process will be killed with “Memory access violation” signal. If we are less fortunate, the address will point to some random location within the address space of the program, so the jump will succeed and the program will continue to run, but not as it was expected to do. In the above example we used a simple `for` loop to illustrate the problem, but typically buffer overruns happen when calling system or library functions for copying data without proper boundary checking — `strcpy()` being the most notorious one. Targeted attacks on buffer overflows usually provide a specially prepared strings of binary data that gets copied using `strcpy()`, `scanf()`, `gets()` or any other vulnerable function present in the attacked program. This data will contain the machine code of a program that is going to be executed and other data to fill up and overwrite the destination buffer, but it must be prepared in such a way, that the data copied to the stack overwrites the function return address with a pointer to the stack where the injected program starts. Predicting this value is the only tricky part of the exploit, but most of the programs are very consistent in stack allocation between different program runs, so this is rather a trivial problem that can be easily solved by running the process under a debugger such as `gdb`.

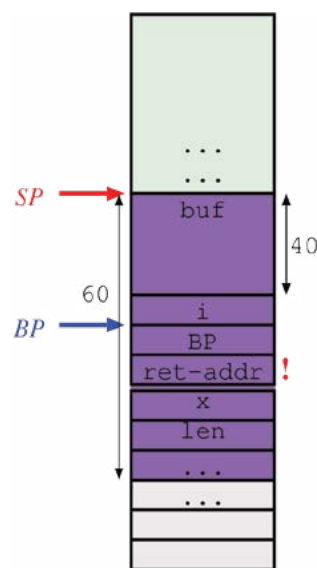


Fig. 5. Overwriting of the buffer on the stack

As some variables are stored on the stack (i.e. variables declared as *local* inside a function or a code block), it is really important to check boundaries of any write operations that may operate on locally declared tables or vectors. Let's consider a function definition as shown in fig. 6.

```
int sample(int n, char *ptr) {  
    int i;  
    char buf[100];  
  
    sprintf(buf, "Read %d samples from %s\n", n, ptr);  
    ...  
}
```

 **Fig. 6.** Buffer overflow problem

This function may be called with any parameters, so when the *sprintf()* function gets called it will happily copy whatever is passed as the second parameter to the buffer allocated on the stack. The buffer's size is set to 100 bytes and some of these will be allocated for the printed text, so effectively, there remains about 80 bytes for the remaining data. If the string passed by the *ptr* parameter is longer, the *sprintf()* will not stop, but continue copying the bytes past the end of the tab buffer, overwriting whatever is there — the **variable i**, the function frame and the function return address, so when finally the function will end, the program execution will not be transferred to the place where the function was called but to some random memory location.

Programs that contain buffer overrun errors are vulnerable to attacks that can be done on different levels. A simple exploit script that is designed ad-hoc will probably result in a *Denial of Service* (DoS) by crashing the program. A more sophisticated exploit may be crafted for a particular program and can result in a much more dangerous *Remote Code Execution* (RCE).

Buffer overrun problem gained widespread attention after being published in the Phrack magazine in 2001 [7], but it seems it is still one of the most important but still overlooked problems in writing applications.

The problem may manifest itself in many different situations, whenever some data is being copied between memory locations without the proper boundary checking. It is especially dangerous if the data is copied from standard input (*stdin*), program parameters (*argv*) or environment variables (*getenv()*), as it enables an attacker to control easily what is being copied to the overwritten stack and actually exploit the error by supplying a specially crafted *shellcode* — a piece of assembly code placed on the stack which when executed, runs a shell, possibly with the privileges obtained from the program with a buffer overrun problem.

The actual dangerous copying may be done by different means, some of which include:

- using *strcpy()* instead of *strncpy()*;
- using *gets()* instead of *fgets()*;
- improper usage of *sscanf()*;
- improper usage of pointers and wrong loop ending conditions.

History shows that most of the programs had this kind of problem. A lot of open source programs have been cleaned from dangerous system calls once buffer overflows gained widespread attention, but still a lot of problems may exist undetected in closed source pro-

grams or even in open source that is not maintained very actively. In some cases it may be even easier to find an exploit for a buffer overrun bug hidden in a program than to fix it, as it is quite simple to analyse its binary version, looking for calls to functions such as *gets()* or *strcpy()*. They need to be listed explicitly in order to allow dynamic linking of system libraries. Any calls to these functions found in a program are a clear invitation for further analysis and writing an exploit.

The best advice for avoiding this kind of problem is of course to pay attention to boundary checking and always make sure that memory copying functions are bound by the limits of the maximum size of destination buffer. Thus, never use functions such as *strcpy()* or *gets()* for reading of user-provided data. However, as the problem is widespread, there are also some provisions in the development tools to help avoid this problem.

One of them is the compiler feature called 'stack canary' or simply a 'canary'. The compiler adds some extra code to both the function entrance code, where the stack frame is created and setup, and to the function exit code, where the return statement is about to be executed. The frame setup code puts an extra value on the stack between the local variables and the function return address. The value is usually chosen at random at compile time. The exact location of the canary in the stack frame may also vary but should always be between the local variables and the function return address.

The canary is always initialized when the function starts and function's exit code simply checks for the canary's value whether it is still there. If the value is different than expected, it means that the canary has been overwritten in some unexpected way, so instead of allowing a return to some unpredictable location, a system call is made to terminate the running process. This way, if a buffer overrun is detected, it is stopped before the injected code can be executed. The program experiencing such a behaviour should nevertheless be rewritten in a proper way, but at least the potential damage can be prevented in most of the cases. Note however, that if the canary location and value is predictable, a carefully chosen exploit attack may still get undetected if the injected data overwrites the canary with the same value as it was already written in the frame setup code. This is why this value has to be randomized at compile time — or even better: at runtime. This second approach is a little bit harder to implement, as the function return code must be able to verify what was the original value of the canary and in this case it cannot be hardcoded. A typical approach will require the the setup code to place another copy of the canary in a place where it cannot be overwritten (i.e. at the beginning of the stack frame, before the space allocated for the local variables), or even split this into several pieces and disperse this information over several places. This way any statically-prepared exploit will be useless, as the probability of overwriting the canary with its own value becomes practically zero.

The canary trick is now the default behaviour of most of the C compilers, so basically you do not need anything to do in order to use it, but rather you need a special compiler option `-fno-stack-protect` to turn it off. It may be desirable sometimes for the performance reasons in some time-critical functions that are called very often. However, if you are really going to turn it off, make sure that the unprotected code does not contain any potentially dangerous statements.

Another technique that can help against buffer overruns is tightening permissions on memory segments and preventing execution

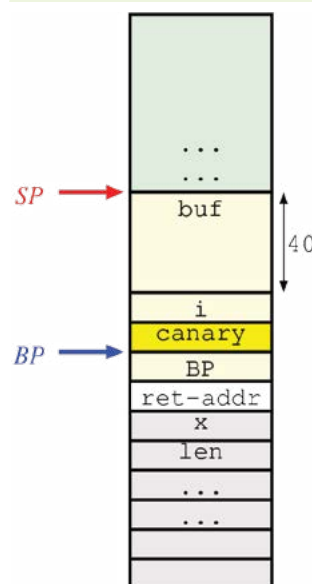


Fig. 7. Buffer overflow problem

of data from stack and data segment (this is called 'Data Execution Prevention (DEP)'). This however depends heavily on operating system support and cannot be used everywhere. Memory pages which are organized in segments contain also permission bits similar to those available for files — read, write and execute. Most often they are ignored, i.e. all memory pages are created with all rwx bits set, but there is no real need for pages belonging to data segment or stack segment to actually use the x permission bit (and the code segment does not need the w bit). We need of course the read and write bits for the stack segment to function properly, but if the x bit is taken out and the memory permissions are enforced by the processor and its memory management unit, any typical buffer overrun attack becomes ineffective. It is possible of course to overwrite the stack data with the exploit script and even make the processor jump through the ret statement to the stack address containing the injected exploit code, but as soon as the processor fetches this code in execution mode from a memory page that has no execution permission, the processor exception is generated, which will basically kill the running process, unless it is dealt with in some exception handling routine. In any case, the injected code will not be executed.

Finally, a technique called 'Address Space Layer Randomization (ASLR)' can be used. This is a feature that randomizes where various parts of the program are located in memory when the program is run. E.g. the stack base address or memory heap starting address will be chosen differently each time the program is run. This complicates attacking of a vulnerable program as it is much harder to prepare an exploit that will match the memory layout of the attacked program. However, the vulnerability stays open — typically, an exploit will be reduced from Remote Code Execution to program crashing (i.e. DoS), but the program is still exploitable and more sophisticated exploits can be written that adapt to this introduced randomness, so ASLR should not be treated as a remedy.

FILE PERMISSIONS

File permissions play an important role securing access to various system resources as well as individual user files and should not be overlooked when writing any application. Very often temporary files are needed by programs, which are usually created in the `/tmp` directory — if they are readable by anybody, some internal state of the program may be revealed, and if the programmer did not care about the `umask` value and created a file with all-write permissions, anybody else may disrupt the execution of the program by modifying such a temporary file.

The basic rule about file permissions when writing programs is to keep the permissions to the minimum, i.e. only enough for the program to work properly and within the defined specification but not more. So it may respect the default `umask` (i.e. defined by the user running a program) when creating output files, but for temporary files or any other transient operations it is safer to set the `umask` to 066, so that anybody else will not be able to read/write these files. If the program needs special privileges and is run in `suid` mode, even more care has to be taken for accessing files and producing any output. Access to files that are supposed to be world-readable should be done only after dropping special privileges (by using the `seteuid()` function). Failing to do so may result in race conditions or problems related to symbolic links usage.

Some well-known permission problems are specific to particular versions of the operating system and some are rather the effect of lazy programming and may appear every-

where. Older systems allowed deleting files owned by other people if they were placed in publicly writable directories, such as `/tmp`, as the all-write access bit set for such a directory allows it. Modern systems still may allow this, but the sticky bit (`+t`) set on the directory makes these operations safe, i.e. disallowing any file manipulation of files owned by other users. This also helps reducing risks of race conditions, but still some other measures have to be taken to prevent them from happening.

Default `umask` needs to be changed especially in servers — the programs that once executed, will run endlessly in a loop, waiting for clients to connect. Also it is worth knowing, that if a server is run from the system startup scripts (located in `/etc/rc?.d` directories), its `umask` value will probably be set to 0, so a server program should not assume anything and set this value to some more appropriate value, regardless of the initial value.

DEFAULT DIRECTORIES AND RELATIVE PATHS

For programs that run as a service or a system *daemon* (i.e. once started, they do not exit, but service new clients as they appear) it is usually desirable to provide some way of reinitialization or re-checking of the initial state. Typically, if such a process receives a `SIGHUP` signal, it will reopen all log files, re-read its config file and, generally, restart its operations. As the program name is kept in `argv[0]`, the simplest way is to use this name in the `execv()` or `execvp()` call (so that all initial parameters are also preserved). However, if the program has been started with a relative path or from a path which was a symlink in the `/tmp` directory, this may lead actually to the start of a completely different program if that symlink has been changed in the meantime.

In suid programs care must be taken to drop all special privileges before proceeding with any of the `exec()` system calls. If the privileges are really needed, the `exec()` function will take care of them (i.e. grant them on execution of a file read from the disk), while not dropping them may lead to using them in a program that should be run with privileges of a normal user.

REFERENCES

- [1] **W. Cheswick**. "An evening with Berferd, in which a cracker is lured, endured, and studied" Proceedings of the Winter USENIX Conference (San Francisco), January 1992.
- [2] **S. Garfinkel, G. Spafford and A. Schwartz**. Practical Unix & Internet Security, O'Reilly & Associates, Inc., 3rd edition, 2003. [Online]. Available: <http://www.oreilly.com/catalog/9780596003234>; <http://www.oreilly.com/catalog/puis3>
- [3] **M. D. Bauer**. Building secure servers with Linux. O'Reilly & Associates, Inc., 2nd edition, 2005. [Online]. Available: <http://www.oreilly.com/catalog/9780596006709/>
- [4] **C. Stoll**. The Cuckoo's Egg. Pocket, 1st edition, 1990.
- [5] **The MITRE Corporation**. Top 25 most dangerous software errors. Dec. 2010. [Online]. Available: <http://cwe.mitre.org/top25/index.html>
- [6] **The Open Web Application Security Project**. "OWASP testing guide" <http://www.owasp.org/>, Aug 2009. [Online]. Available: <http://www.lulu.com/product/file-download/owasp-testing-guide/5310591>
- [7] **Aleph One**. Smashing the stack for fun and profit. Phrack, vol. 49, Apr. 2001, <http://www.phrack.org/archives/49/>. [Online]. Available: <http://phrack.org/issues/49/14.html>

Education Challenges and Development of Advanced Malware Analysis Course

*Alexander Adamov, Kharkov National University of Radio Electronics, Ukraine.
E-mail: oleksandr.adamov@nure.ua*

ABSTRACT

The article describes the challenges of Advanced Malware Analysis (AMA) course development for the new EU-recognized Master's programme within the Educating the Next Generation Experts in Cyber Security (ENGENSEC) project. The development results in the discussion on future work that can be done to improve the course. The advantage of the practices of continuous integration and continuous delivery in educational course development is shown to keep up with the latest types of cyber threats.

1. Introduction

Modern cyber attacks are becoming more sophisticated year after year and require more and more skills to analyse, detect and counteract threats nowadays [1]. Zero-day exploits, spear-phishing, watering hole attack vectors are used to deliver evasive malware into an organization's security perimeter [2]. As a result, there is a high demand for skilled cybersecurity experts [3] to protect organizations on the private and government level.

The aim is to develop the course that will be used to teach malware analysis to the next generation of security experts who can whether participate in the incident response process analyzing malware or collect threat intelligence to be used by the deployed security solutions.

2. Problem overview

The problem lies in a lack of security professionals [3] capable of analysis of complex malware such as Advanced Persistent Threats (APT) delivered through targeted attacks and used for cyber espionage purposes and the latest versions of ransomware that employ network worm capabilities and leave no chance to decrypt encrypted data other than to pay a ransom [1,2].

According to [4], malware analysis courses were not being taught until 2011 in academic institutions all over the world. Currently, the Malware Analysis course is presented in the educational programs of the following universities:

In the United States:

- Sam Houston State University [5].
- University of Maryland Baltimore County [6].
- University of Maryland [7].
- James Madison University [8].
- University of Missouri [9].
- University of Massachusetts Boston [10].
- Stevenson University [11].
- Roger Williams University [12].
- Rensselaer Polytechnic Institute [13].
- George Mason University [14].

Articles

Education Challenges and Development of Advanced Malware Analysis Course

In the United Kingdom:

- University College London [15].
- University College Dublin [16].
- University of York [17].
- De Montfort University [18].
- Staffordshire University, SSTK-11054 [19].

In Europe:

- Norwegian University of Science and Technology, IMT4116 [20].

The discovered US courses in malware analysis are based on Practical Malware Analysis book by Michael Sikorski and Andrew Honig issued in 2011 [21]. For example, in George Mason University [14], the course is self-paced by students that follow the mentioned book.

3. Similar developments

The malware analysis currently is not well represented in an academic realm as stated by the US researchers from Sam Houston State University [22]. They presented an overview of malware related courses and noticed the lack of such courses in the academic world.

Another work emphasizes the importance of looking at cultural differences across nations in cybersecurity education to reveal attackers' motivation and biases [23].

The discovered malware analysis courses and trainings are mentioned below:

1. The course "Computer viruses and tools to combat them" by Kaspersky Lab. The course was included to the bachelor's degree programme of Kharkiv National University of Radio Electronics (KhNURE). It has been recently updated with ENGENSEC materials and now covers the first half of the AMA course (4 ECTS) [24].
 - *Pros:* One of the first courses about computer threats and antivirus technologies.
 - *Cons:* The course is aimed mostly at studying of Kaspersky Lab's products describing an installation and configuration process.
2. Practical Malware Analysis. The Hands-On Guide to Dissecting Malicious Software by Michael Sikorski and Andrew Honig. This book methodologically describes the base malware analysis techniques in a well-organized manner. It includes both theoretical definitions and practical exercises specially developed for the book. Even though the book materials cannot be used in the course, they might perfectly complement the developed AMA course and help students in self-study [21].
 - *Pros:* The book contains many practical examples and tasks. It explains malware analysis on two levels: basic and advanced.
 - *Cons:* The content of the book is proprietary and cannot be included in the course.
3. Malware Analysis Crash Course by the Flare Team of Mandiant and FireEye Company, BlackHat USA 2017. The course is two-day training given by the cyber security experts of Mandiant and FireEye [25].
 - *Pros:* The training covers advanced topics in Malware Analysis and is given by the industry experts.
 - *Cons:* It lasts only for two days and is rather expensive (\$4000).
4. Malware Discovery and Basic Analysis by Michael Gou [26].
 - *Pros:* Malware analysis automation and infrastructure is considered during the training.
 - *Cons:* The training is patchy and covers only malware analysis basics.
5. Malware Analysis Training by Robert Simmons [27].
 - *Pros:* The training is devoted to open source tools, covers advanced topics in Malware Analysis and is given by the industry experts.
 - *Cons:* The training targets only usage of the following tools for malware analysis: Cuckoo Sandbox, IDS/IPS Bro, Volatility Framework.
6. GIAC Malware Analysis Certification: GREM by SANS Institute [28].
 - *Pros:* The training covers advanced topics in Malware Analysis and is given by the industry experts.
 - *Cons:* The two-day course costs \$4000.
7. European Cybercrime Training and Education Group (ECTEG) Malware Investigations [29].
 - *Pros:* The course describes the techniques used to set up a sinkhole server and shut down a botnet.
 - *Cons:* The one-week course does not cover reverse engineering and the disassembly of PE files.

4. Background work

The following steps were taken by the authors of the course to develop malware-oriented educational materials in the past:

1. Creation of the community of cyber security experts and development of educational materials for lectures and practice work within the "Baltic Academic IT-Security Exchange" project during 2011–2013.
2. The university course called 'Computer Threats: Methods of Detection and Analysis' that was developed by Alexander Adamov in 2009 based on Kaspersky Lab training course 'Computer viruses and means of combating them' [30] within the collaboration between Kaspersky Academy and Kharkiv National University of Radio Electronics. The course later was adopted as a part of the Master Programme in Kharkiv National University of Radio Electronics [24] and has been recently rewritten to include the materials of the ENGENSEC AMA course.

5. Results

The AMA course [31] has been developed within the ENGENSEC project. It is 7.5 ECTS has and includes 112 classroom and 158 self-study hours. The course package includes:

- The slides and multimedia for 17 lectures
- The study guide for 5 practice works
- The study guide for 10 laboratory works
- The video guides for 5 laboratory works
- The evaluation test
- The image of the virtual machine with the laboratory environment

The course has been already included into the cybersecurity master programme by the following universities:

- Blekinge Institute of Technology, DV2567, Sweden [32].
- Kharkiv National University of Radio Electronics, Ukraine [33]. The Advanced Malware Analysis is split into two courses: "Computer viruses and tools to combat them" for bachelor students and the second one is named "Computer Threats: Detection and Analysis Methods" for master students in KhNURE [24].
- The Bonch-Bruевич St. Petersburg State University of Telecommunications, Russia [34].

6. Validation

The course has been tested in the iterative manner with further integration of the newly developed materials in Blekinge Institute of Technology, Sweden and Kharkiv National University of Radio Electronics, Ukraine during the last three years (2015, 2016, 2017).

The course received good feedback from students collecting 3.44 points out of 4 (86% of satisfaction) during the course run in 2016 in Blekinge Institute of Technology. However, in the feedback, the students, who had evaluated the course, argued about too many practical assignments and tight course schedule.

The students, who had successfully passed the AMA course, were graduated and hired as cybersecurity experts by the leading security consulting companies, such as Outpost24 [35], Deloitte [36] and TrueSec AB [37].

7. Conclusions

As a result of collaboration between the universities and industry experts within the ENGENSEC project, the AMA course has been developed, verified and adopted to the master programme of the project members' curriculum during 2013–2017.

The developed AMA course has the following advantages and disadvantages:

Advantages:

- The unique content for studying of reverse engineering and malware analysis which differs from the content of the Practical Malware Analysis book by Sikorski M. and Honig A. used in the US universities.
- Includes practical examples of analysis of "in the real world" malware used in cyberattacks.
- Based on usage of free and open source software.
- Considers antivirus industry standards and best practices.
- The materials are constantly updating to cover new threats, such as the latest versions of ransomware, and recently discovered targeted attacks.
- Tested and adopted by the universities, members of the ENGENSEC project, in three different countries and received the evaluation from both teachers and students.

Disadvantages:

High prior knowledge requirements that include: cryptology, system programming, Windows system administration, C/C++, Python, assembler IA32/ARM, networking.

Teachers need to be trained to run the AMA course.

8. Future work

The AMA course can be extended to address the following topics in future:

- Critical infrastructure cyberattacks.
- Decoding of hardcoded configurations for the malware used in the SaaS model.
- More data mining technologies that can be applied for detection and analysis of cyberattacks. For example, application of the clustering analysis for cyber attack detection based on network traffic and memory dumps.
- The dedicated module for learning how to create Yara and IDS/IPS rules to detect specific classes of cyberattacks.

REFERENCES

- [1] **Guerrero-Saade J. A., Raiu C., Baumgartner K.** Kaspersky Security Bulletin: Threat Predictions for 2018, available at <https://securelist.com/ksb-threat-predictions-for-2018/83169/>
- [2] GReAT, APT Trends report Q3 2017, Kaspersky Lab, available at <https://securelist.com/apt-trends-report-q3-2017/83162/>
- [3] Mitigating the Cybersecurity Skills Shortage. Top Insights and Actions from Cisco Security Advisory Services, 2015, available at <https://www.cisco.com/c/dam/en/us/products/collateral/security/cybersecurity-talent.pdf>
- [4] **Ledin G.** Inside Risks: The Growing Harm of Not Teaching Malware: Revisiting the need to educate professional to defend against malware in its various guises, Viewpoints Communications of the ACM, vol. 54, № 2, Feb 2011.
- [5] COSC 5330: Malware, Sam Houston State University, US, available at <http://graduate-catalog-2015-2016.shsu.edu/home/course-descriptions.html#dfsc>
- [6] CMSC491: Malware, University of Maryland Baltimore County, available at <http://www.csee.umbc.edu/courses/undergraduate/CMSC491malware/>
- [7] CMIT 340: Malware Analysis, University of Maryland University College, available at <https://www.umuc.edu/academic-programs/course-information.cfm?course=cmit340>
- [8] CS 640: Malware Analysis, James Madison University, available at <https://www.jmu.edu/cs/graduate/ms-digital-forensics/courses/cs-640/index.shtml>
- [9] CMP_SC 4440: Malware Analysis and Defense, University of Missouri, http://catalog.missouri.edu/courseofferings/cmp_sc/
- [10] IT 421: Digital forensics/Malware Analysis, University of Massachusetts Boston, available at https://www.umb.edu/academics/course_catalog/course_info/ugrd_IT_2017%20Fall_421
- [11] CYBF 685: Malware Detection, Analysis, and Prevention, Stevenson University, available at <http://www.stevenson.edu/online/academics/graduate-programs/cyber-forensics/>
- [12] SEC 620: Malware Analysis and Malicious Software, Roger Williams University, available at <http://rwu.edu/sites/default/files/downloads/admission/cybenecurity-brochure.pdf>
- [13] CSCI 4972/6963: Malware Analysis, Rensselaer Polytechnic Institute, available at <http://security.cs.rpi.edu/courses/malware-spring2013/> and <https://github.com/RPISEC/Malware/>
- [14] CFRS 761-001: Malware Reverse Engineering, George Mason University, available at http://cfrs.gmu.edu/assets/Fall17/CFRS761_Fall17.pdf
- [15] COMPGA16: Malware, University College London, available at http://www.cs.ucl.ac.uk/current_students/syllabus/compqa/compqa16_malware/
- [16] COMP41650: Malware Investigations, University College Dublin, available at https://sisweb.ucd.ie/usis!/W_HU_MENU.P_PUBLISH?p_tag=MODULE&MODULE=COMP41650&MAJR=T146
- [17] COM00095M: Malware & Other Malfeasance, University of York, available at <https://www.york.ac.uk/students/studying/manage/programmes/module-catalogue/module/COM00095M/2017-18>
- [18] Malware Analysis, De Montfort University, available at <http://www.dmu.ac.uk/study/courses/postgraduate-courses/cyber-security/cyber-security-msc-degree.aspx>
- [19] SSTK-11054: Staffordshire University, available at <http://www.staffs.ac.uk/course/SSTK-11054.jsp>
- [20] IMT4116: Reverse Engineering and Malware Analysis, Norwegian University of Science and Technology, available at <https://www.ntnu.edu/studies/courses/IMT4116>

- [21] Practical Malware Analysis. The Hands-On Guide to Dissecting Malicious Software by Michael Sikorski and Andrew Honig, February 2012, 800 pp., available at <https://www.nostarch.com/malware>
- [22] **Shashidhar N., Cooper P.** Teaching malware analysis: The design philosophy of a model curriculum, 4th International Symposium on Digital forensics and Security (ISDFS) 2016, Little Rock, AR, USA, 2016, available at <http://ieeexplore.ieee.org/document/7473529/>
- [23] **Hwee-Joo Kam, Katerattanakul P.** Diversifying cybersecurity education: A non-technical approach to technical studies, IEEE Frontiers in Education Conference (FIE) 2014, 22-25 Oct 2014, Madrid, Spain, available at <http://ieeexplore.ieee.org/document/7044197/>
- [24] Cybersecurity Courses at the Design Automation Department, Kharkiv National University of Electronics, <http://nure.ua/uk/university/structure/departments/kiu/apvt/disciplini-vilnogo-viboru-studenta/>
- [25] Malware Analysis Crash Course by the Flare Team of Mandiant and FireEye Company, BlackHat USA 2017, July 22-23, 2017, Las Vegas, NV, US, available at <https://www.blackhat.com/us-17/training/schedule/index.html#malware-analysis-crash-course-5748>
- [26] Malware Discovery and Basic Analysis by Michael Gou, hackmiamicon5, May 19-21, 2017, Miami, FL, US, available at <https://www.hackmiami.com/live-training-seminars>
- [27] Malware Analysis Training by Robert Simmons, hackmiamicon5, May 19-21, 2017, available at <https://www.hackmiami.com/live-training-seminars>
- [28] GIAC Malware Analysis Certification: GREM, SANS Institute, available at <https://digital-forensics.sans.org/certification/grem>
- [29] European Cybercrime Training and Education Group (ECTEG) Malware Investigations, 2017, available at http://ecteg.eu/c_malwareinvestigations.html
- [30] Computer viruses and means of combating them: Training course, M. : Kaspersky Lab, 2012. — 608 P., available at <http://www.intuit.ru/studies/courses/1042/154/info>
- [31] New Cyber Security Course for Master Students, NioGuard Security Lab Blog, 2017, available at <https://nioguard.blogspot.se/2017/07/new-cyber-security-course-for-master.html>
- [32] Blekinge Institute of Technology, <https://www.bth.se/eng/>
- [33] Kharkiv National University of Radio Electronics, <http://nure.ua/en/>
- [34] Federal State Budget-Financed Educational Institution of Higher Education The Bonch-Bruевич Saint-Petersburg State University of Telecommunications, <https://www.sut.ru/eng/>
- [35] Outpost24, <https://outpost24.com/>
- [36] Deloitte, <https://www2.deloitte.com/global/en.html>
- [37] TrueSec AB, <https://www.truesec.se/>

Development of Informatics Students' Competence in the Field of Information Security of Internet Resources

(*Informatization of Education and Science*. — № 3 (35). — 2017, P. 38–49)

Rudinskiy Igor D. FSBEI HE «Kaliningrad State Technical University», Kaliningrad, Russia, Doctor of Pedagogics, Professor of the Department of Control Systems and Computer Engineering. E-mail: idru@yandex.ru.

Mikhaylovskiy Mikhail Yu. FSBEI HE «Kaliningrad State Technical University», Kaliningrad, Russia, Postgraduate student of the Department of Control Systems and Computer Engineering. E-mail: mih.mu93@mail.ru

Abstract. The article analyzes the peculiarities of the internet resources created with the help of content management systems. It serves to justify the importance of the informatics students' competence development in the field of information security of internet resources. The article describes the structure and content of the given competence and shows what approach should be taken to develop it.

Keywords. *information security, internet resource, content management system, educational technology, professional competence.*

Introduction. The rapid development of Internet technologies in the last 10 years has led to a situation where both business and government are abandoning the provision of services on the ground in favor of the Internet. Modern sites are not just information pages, but full-fledged Internet resources through which the site owner (physical or legal person, authority, etc.) interacts with the visitor of the site, usually acting as a customer-consumer of services. Increasingly, through Internet resources, financial exchange or the exchange of confidential information is carried out. One of the most common types of interaction on the Internet with the use of confidential information is Internet commerce [1]. According to statistics provided by the research agency Data Insight, which specializes in the e-commerce market, according to the results of 2014, more than 195 million purchases worth about 560 million rubles were made in Russia. On average, for each buyer, there were about 7 orders per year. On-line purchases were issued by more than 25 million people, that is 37% more than in the previous year [2]. At the same time, the problems of the vulnerability of Internet resources, which have, as surprisingly as it sounds, pedagogical nature, become

more important. For a detailed analysis of the emerging situation, let us consider the most common technology of developing and maintaining websites.

Problem analysis. Currently, the majority of both large and small sites are implemented using ready-made platform solutions, including open ones (OpenGL license) [3]. Many customers are trying to minimize the cost of the project that leads to a sharp increase in the popularity of free content management systems (CMS), such as Wordpress, Joomla, Drupal, etc. [3]. Due to the apparent simplicity, many sites based on such platforms are developed and maintained by their users themselves or with the involvement of newcomers in the field of information technology (IT). This often leads to poor-quality implementation of the Internet resource, that, in turn, causes vulnerabilities in the code generated by CMS tools.

One of the key problems in the field of information security of Internet resources is the low awareness of both owners and operational staff of the websites about threats to the integrity and confidentiality of data that are used for various purposes by Internet resources. After completing the CMS security research in 2014, its authors attempted to contact the owners of about 130 sites that were infected with viruses or were on the "black lists" of search engines. More than half of the owners whose websites were actively used to promote a product or a service did not know about the existence of problems with their Internet resources. At the same time, a third of them continued to spend money on service or promotion of the site [4].

In our opinion, the results of this study indicate that the site staff (administrators, editors, content managers, etc.) in most cases have insufficient level of training in the field of information security, and at that these employees focus on

promotion of the content of the site and not on its protection from intruders. The availability of the majority of free CMS and the simplicity of their usage lead to the fact that many owners of Internet resources, who do not have a specialized education in the IT field, refuse the help of specialists and independently develop and maintain their own website. In order to save financial resources, users themselves configure the CMS that supports the site, thus, it often leads to vulnerabilities - incorrect code fragments used by hackers to crack Internet resources for fraudulent usage purposes of both the site itself and the data stored on it. This is largely due to the fact that owners of Internet resources do not have confidence in that the most of young IT professionals and university graduates offering them their professional services have appropriate competencies for such service.

Unfortunately, these doubts should be recognized as such that are not without merit. The analysis of the qualification requirements for the competencies of university graduates, which are included in the federal state educational standards for higher education (FSES HE) in the IT areas of training, demonstrates the lack of professional competences among these requirements in the field of identifying and eliminating vulnerabilities in Internet resources which is one of the most important aspects of providing their information security. Thus, since 2014, in the current FSES HE, insufficient attention is paid to the problem of teaching students to provide information security of Internet resources in the direction of training 10.03.01 "Information Security". In particular, among the list of professional tasks, Internet technologies are not mentioned at all, and information security issues are mentioned only in the most general context: "Installation, configuration, operation and maintenance of components of the information security system in working order, taking into account the established requirements" [5].

In the list of competencies that a graduate of this field of training should possess, Internet technologies are mentioned only in a very incomplete formulation of the overall professional competence OPC-4 (*"The ability to understand the essence and significance of information in the development of modern society, to solve standard problems of professional activity on the basis of information and bibliographic culture with the use of information and communication technologies"*), whereas Internet technologies are not at all mentioned in the formulations of the professional competences.

In 2016, in the FSES HE in the direction of training 09.03.01 — Informatics and computer science (qualification

"Bachelor"), very common in the universities of the Russian Federation, the problem of teaching students to provide information security of Internet resources is paid even less attention to. In particular, among the professional tasks that a graduate of this training area must solve in the framework of professional activity, there appears only one task in which the Internet technologies are mentioned in a very general way (in the formulation of the above-mentioned FSES HE — Web-technologies) without any coordination with the range of problems concerning information security (IS) provision of Internet resources: ***"Application of Web-technologies in the implementation of remote access in client / server systems and distributed computing"*** [6].

In the list of competencies that a graduate of this area should possess, information security is mentioned only in a very incomplete formulation of the overall professional competence of the OPC-5 (*"The ability to solve standard tasks of professional activity based on information and bibliographic culture using information and communication technologies and taking into account the basic requirements of information security"*), whereas neither the problems of information security, nor the internet technologies are mentioned in the formulations of professional competences.

The analysis shows clearly the insufficient attention given by the developers of the FSES HE in IT areas to the issues of information security of Internet resources. In our opinion, this is the reason for the urgency of supplementing the targeted results of educational activities for the above-mentioned FSES HE with a new professional competence directly aimed at providing IS of Internet resources, processing the work programs of the relevant academic disciplines for mastering the necessary technologies, methods and tools within them.

Definition of the proposed professional competence. Taking into account the above, we consider it necessary to supplement the list of professional competencies of the students studying the field of training 09.03.01 — Informatics and computer technology and related areas, with a competence ***"The ability and readiness to apply modern methods and technologies of processing information to provide information security of Internet resources"***. Guided by GOST 33244-2015 "Information technologies. Teaching, education and training. Conceptual Reference Model of Competence and Related Objects" and using GOST R 52807-2007 "Guidelines for the Assessment of the Competence of Project Managers" as a model, we will offer a description of the developed competence, given in Table 1.

Table I. Description of the competence “The ability and willingness to apply modern methods and technologies for processing information to provide information security of Internet resources”

Elements of competence (qualification characteristics)	Criteria of professional conformity	Levels of competence element formation
Cognitive component		
Knowledge of sources, risks, forms, methods and signs of attacks on Internet resources.	Knowledge of the classification of attacks, understanding of the principles of their actions, knowledge of the main types of attack sources, the severity of their impact.	1. Threshold level: to know what is called an «attack on the Internet resource», the principles of its conduct and the consequences of implementation, to be able to recognize its obvious signs; 2. Basic level: to know the classification of attack types and their hidden signs, the severity of the impact of specific types of attacks, the main sources of attacks of each type and their most likely consequences; 3. High level: to know how to organize and conduct attacks; to know the methods and technologies for detecting, fixing and blocking attacks.
Knowledge of the classification of vulnerabilities in Internet resources, the causes of their appearance, the degree and types of damage they cause.	Knowledge of the classification of vulnerabilities, their types, the ability to recognize the causes of their appearance, knowledge of the basics of network technologies, the fundamentals of server operation. Knowledge of the degree of harm, its types and places of manifestation.	1. Threshold level: to know the basics of network technologies, the basics of server operation; to know the classification of vulnerabilities, the extent and types of damage caused by them; 2. Basic level: to know the causes of the occurrence of the most common vulnerabilities and their signs; 3. High level: to know the likely places of occurrence of the most common vulnerabilities, the ways of their localization and elimination.
Knowledge of the ways and means to eliminate identified vulnerabilities.	Knowledge of the main ways to eliminate vulnerabilities, plug-ins to protect the Internet resource, web scanners, principles of authentication and protection of used content.	1. Threshold level: to know the main plug-ins for protecting the Internet resource, web scanners, the principles of authentication and protection of used content; 2. Basic level: to know the most popular ways to remove vulnerabilities by fixing code, the principles of managing rights to data access; 3. High level: to know how to fix vulnerabilities at the level of designing a web server.

Elements of competence (qualification characteristics)	Criteria of professional conformity	Levels of competence element formation
Knowledge of regulative documentation on information security provision of Internet resources.	Knowledge of federal laws and standards, as well as other federal and departmental regulations in the field of information security.	1. Threshold level: to know the structure and essence of the federal legislation in the field of information protection and information security; 2. Basic level: to know the purpose of federal laws and major departmental regulations in the field of information protection and information security; 3. High level: to know the content of federal laws and departmental regulations in the field of information protection and information security of Internet resources, to be able to analyze and evaluate the practice of their enforcement.
Knowledge of information protection concepts.	Knowledge of legal and organizational principles of information protection. Knowledge of the technical and programmatic principles of information protection, their classification and application areas.	1. Threshold level: to know the basic legal and organizational principles of information protection and their classification; 2. Basic level: to know the content of legal and organizational principles of information protection; 3. High level: to know the content and to be able to explain the consequences of implementation of legal, organizational and technical principles of information protection.
Knowledge of authentication and authorization methods of users of Internet resources.	Knowledge of the main methods of authentication and authorization used in modern Internet resources; knowledge of software constructs that implement authentication and authorization functions in the original program code and in the applicable CMS.	1. Threshold level: to know the main methods, algorithms and means of authentication and authorization used in modern Internet resources; 2. Basic level: to know the principles of using of the main methods, algorithms and means of authentication and authorization used in modern Internet resources; 3. High level: to know and / or to be able to create software constructs that implement authentication and authorization functions in the original software code and in the applicable CMS, to be able to determine where they are implemented.
Knowledge of strategies, principles for compiling and implementing policies for providing information security of an Internet resource.	Knowledge of the basic strategies of security policies, their essence; knowledge of the principles of compiling and implementing policies to provide information security of Internet resources, their legal bases and implementation examples.	1. Threshold level: to know the basic strategies of security policies; 2. Basic level: to know the essence of the basic strategies of security policies, to know examples of the implementation of the strategy of information security policies of large IT projects and IT companies; 3. High level: to know the principles of compiling and implementing policies to provide information security of Internet resources, to know their legislative basis.

Elements of competence (qualification characteristics)	Criteria of professional conformity	Levels of competence element formation
Technological component		
The ability to use the most common content management systems (CMS) for development and maintenance of an Internet resource.	The ability to customize applied CMS and to manage their content.	1. Threshold level: to be able to explain the operation principles of the most common CMS and the features of their application; 2. Basic level: to be able to use the administration panels of the most common CMS and perform basic configuration; 3. High level: to be able to purposefully change the program code of standard CMS files, create graphical design of an Internet resource, apply authentication, distribution, data storage and use fine-tuning of backups, data caching, registration of Internet resource visits and user rights.
A good command of encryption and coding information methods.	The ability to apply the basic methods of encryption for cryptographic protection of information with a public and private key, as well as methods of antinoise coding.	1. Threshold level: to be able to explain the principles and applications of the most common methods of encryption and coding of information, their differences and classification; 2. Basic level: to know how to use well-known encryption methods with public and private keys, methods of antinoise coding; 3. High level: to be able to develop new methods of encryption or antinoise coding or create original implementations of well-known methods.
A good command of data backup methods.	The ability to apply modern data backup methods used in Internet resources.	1. Threshold level: to be able to explain the basic principles and methods of backup, to be able to apply the technology of data copying to a maintaining Internet resource; 2. Basic level: to be able to apply methods of data backup implemented in the applied CMS including plug-in using; 3. High level: to be able to select and implement a specific technology of data backup to the Internet resource by modifying the program code of the Internet resource or by configuring the settings and parameters directly on the data server of the Internet resource.

Elements of competence (qualification characteristics)	Criteria of professional conformity	Levels of competence element formation
A good command of technologies for detecting intrusion into Internet resources.	The ability to apply modern web scanners, antivirus products, the ability to program in HTML and PHP with SQL, the ability to identify and recognize SQL injections.	1. Threshold level: to be able to work with common anti-virus programs and web scanners, write and understand simple programs in PHP and HTML with SQL, to recognize the main kinds and types of SQL injections; 2. Basic level: to be able to implement simple code scanning functions in PHP and HTML languages, to be able to form simple SQL queries and identify simple SQL injections; 3. High level: to be able to perform advanced scanning of an Internet resource using web scanners and anti-virus programs; to be able to program complex loops and functions in PHP and HTML, form complex SQL queries and identify hidden SQL injections.
A good command of methods of scanning and optimization of Internet resources using plug-ins.	The ability to configure and apply advanced security plug-ins to optimize the performance and analyze the integrity of the CMS.	1. Threshold level: to be able to explain the purpose, principles of construction and operation of plug-ins, to know the classification and purpose of typical plug-ins of the most common CMS; 2. Basic level: to be able to select, install and configure typical plug-ins of the most common CMS; 3. High level: to be able to tweak specific plug-ins of the most common CMS, to be able to create original plug-ins.
A good command of methods for eliminating of vulnerabilities and their consequences in the program code.	The ability to work with the program code, block certain functions to identify the site with a vulnerability, analyze the code for integrity and correctness, delete or correct fragments of the code without affecting the operability of the Internet resource, conduct its audit.	1. Threshold level: to be able to explain the purpose of a fragment of the program code, to block certain procedures and functions; 2. The basic level: the ability to analyze the program code, if necessary, highlight the incorrect or damaged components, logically organize the elements and code snippets, the ability to assign access rights to program files; 3. High level: the ability to delete or correct code snippets within a working Internet resource without affecting its functionality, the ability to label the program code, the ability to re-scan and audit it for security testing.

Articles

Development of Informatics Students' Competence in the Field of Information Security of Internet Resources



Elements of competence (qualification characteristics)	Criteria of professional conformity	Levels of competence element formation
Motivational component		
Willingness to apply their knowledge of how to implement threats («active search») not to harm the researched Internet resource and its owner.	Understanding of the information value accessed by a specialist with a good command of the technology of active searching for vulnerabilities in Internet resources, and the need to use it only to improve the information security of the Internet resource and not for other purposes.	Gradation by levels is not provided. To understand social importance and importance of activities to improve and provide information security of Internet resources. To be aware of the responsibility for providing the security of the surveyed Internet resource, to understand the consequences of the leakage of information stored on it, to be ready to guarantee its non-proliferation. To be interested in the success of the measures taken to improve the information security of the Internet resource both from the standpoint of one's own professional prestige and with the goal of achieving a financial result.
Readiness to respond immediately to threats.	Awareness of the need to be ready to have irregular working hours and to fulfill urgent requests for searching, identifying and eliminating vulnerabilities in the customer's Internet resource at any time, including non-working time.	
Personal component		
Diligence.	Readiness for the fact that it may take a long time to find and fix vulnerabilities in a particular Internet resource.	Gradation by levels is not provided. From the standpoint of a customer, the presence of these personal qualities is a prerequisite for a person to allow them an information security audit of an Internet resource.
Flexibility.	Readiness for the fact that a request for an urgent search, identification and removal of vulnerabilities can occur anywhere and anytime. A necessary condition for the realization of such a request is the availability of high-speed mobile Internet access.	
Resiliency.	Readiness for the fact that the process of identifying, localizing, identifying and eliminating vulnerabilities can be accompanied by a large number of erroneous actions.	
Consciousness.	Readiness and ability to resist the temptations of using available confidential information for mercenary ends.	

Elements of competence (qualification characteristics)	Criteria of professional conformity	Levels of competence element formation
Customer's trust.	Ability to prove to the owner of the Internet resource their own decency in the aspect of not using confidential information contained on the Internet resource for personal purposes or in the interests of third parties.	
Professional reputation.	The availability of positive results which are documented or recognized by authoritative members of the professional community and represent activities to provide information security of Internet resources.	
Admission to work with limited information.	In the situations defined by the legislation, the availability of an official permit (admission) to work with information of limited distribution.	

In the above mentioned description of the competence, we did not purposefully point out the names of specific CMS, plug-ins, algorithms and other instrumental and technological tools, since the ability to select, master and effectively use them must be predetermined by both the current situation in the market of IT products and services and by requests of specific customers or employers. The universality of the qualification characteristics of the proposed professional competence will allow a future IT specialist to adapt successfully to the rapid development of web technologies and web-programming tools, including the cross-platform growth of programming languages and the Internet resources created with their help. A good command of this competence will allow an IT specialist to administer any Internet resource effectively, to respond to attempts of hacking or other unauthorized penetration in proper time. To eliminate the consequences of successful penetrations, the full stop of the Internet resource will not be required, and the time for its full debugging will be significantly reduced.

The technology of teaching students to search actively and fix vulnerabilities in Internet resources. In our opinion, the relevance and necessity of development of the proposed competence among the gradu-

ates of IT training directions is due to the fact that the technological audit of information security (IS) is one of the mandatory stages of a life cycle of an organization's IS management system implemented in accordance with the requirements of the international ISO / IEC standard 27001: 2005 [8]. The results of the technological audit not only are of great value by themselves as an objective and independent (in the case of an external audit) assessment of the security of an Internet resource, but also they are necessary for conducting an analysis of information risks and, therefore, for the effective management of the organization's IS.

Nowadays, the procedure for conducting technological audit of IS is almost not formalized. The general principles for audit are standardized (stated, for example, in ISO 19011 or NSA INFOSEC Assessment Methodology), also there are corporate standards and specialized techniques applied in individual companies or for specific information systems (for example, CBR standards), but there are no general rules and applicable in practice universal methods. For this reason, the development of educational technologies for the training of computer science students in the audit of information security of various components of management systems acquires particular urgency.

Within the framework of the proposed competence, we are developing a technology for training computer science students to provide information security of both projected and existing Internet resources with the help of so-called active vulnerability search [9, 10]. The "active search" of vulnerabilities in Internet resources refers to testing the resource for unauthorized penetration (English language sources — penetration testing), in which the possible techniques and tools necessary to penetrate the Internet resource in order to gain access to confidential information or to damage data are modelled. To date, this method is considered to be the most effective one [7]. The tasks of such testing are the definition and analysis of the structure of the web application, the identification of all possible vulnerabilities of the protected Internet resource and the disadvantages of the system of access delimitation among its users.

The main point of the technology lies in the usage of a system of forms, methods, means and techniques of training that provide the development students' ability to identify and eliminate the vulnerabilities which are typical for the used CMS, without compromising the operability of the Internet resource being researched. A virtual laboratory stand is used as a research object and an online resource — it is an online store with numerous vulnerabilities. The virtual laboratory stand is adjusted as close as possible to the actual conditions of functioning of the Internet resource, which almost completely reproduces the functionality of a real online store, including interaction with banking (implementation of a test version of bank processing), payment and accounting systems.

In the process of training, special attention is paid both to the formation of students' personal qualities necessary for the successful detection of disguised vulnerabilities and to the formation of a sustainable moral and ethical attitude to the inadmissibility of using confidential information contained in the Internet resource being researched, for personal purposes or in the interests of third parties. These qualities, which are combined with clearly defined professional and other motivations for identifying and eliminating vulnerabilities in the object under investigation, predetermine the readiness of a future IT specialist for professional activities related to the information security of Internet resources.

LIST OF SOURCES

1. **E. Byrkova.** Statistics of Internet commerce in the countries of the world: main trends and indicators. — 2014. Available at [provad.pf / 16849statistika-intepnet-topgovli-v-stpanah-mipa-osnovnye-tendentsii-i-pokazateli.html](http://provad.pf/16849statistika-intepnet-topgovli-v-stpanah-mipa-osnovnye-tendentsii-i-pokazateli.html) (Accessed 24.10.2016).
2. Internet trading in Russia — 2014. Annual report. A joint study of Data Insight, InSales and PayU. — 2014. Available at <http://www.datainsight.ru/ecommerce2014> (Accessed 15.10.2016).
3. **Oksana Mamchueva.** CMS market research: which CMS are preventing the sites from entering the TOP. — 2013. Available at https://www.searchengines.ru/issledovanie_ry.html (Accessed 18.10.2016).
4. **Investigation** of CMS-system security in 2014. Available at <http://www.ruward.ru/sitesecure/sms-survey/> (Accessed 18.10.2016)
5. **Federal** state educational standard of higher education in the field of training 10.03.01 Information security (bachelor's level). Available at <http://www.consultant.ru/cons/cgi/online.cgi?req=doc&base=LAW&n=209323&fld=134&dst=100131,0&rnd=0.4348703610454252#0> (Accessed 20.10.2016)
6. **The Instruction** of 12 January 2016 p. 5 on the approval of the federal state educational standard of higher education in the field of training 09.03.01 Computer science and computer engineering (bachelor's level). Available at <http://www.consultant.ru/cons/cgi/online.cgi?req=doc&base=LAW&n=194218&fld=134&dst=100001,0&rnd=0.20243341273000426#0> (Accessed 20.10.2016)
7. **Bibina E. S., Osadchaya A. D.** Methods of searching for vulnerabilities in Web applications. — 2016. Available at <http://nauka-rastudent.ru/27/3308/> (Accessed 20.10.2016).
8. **Markevich M.** Principles of conducting an active information security audit of a company. Available at https://dsec.ru/ipmresearchcenter/article/principles_of_carrying_out_active_audit_information_security_company/ (Accessed 20.10.2016).

Teaching Penetration Testing: Focusing on Hacking Techniques or Methodology

*Alexander Dorofeev, Training center Echelon Russia, Moscow, Elektrozavodskaya str, 24.
E-mail: adorofeev@uc-echelon.ru*

*Andrian Piskozub, Lviv Politechnik National University, Ukraine.
E-mail: azpiskozub@gmail.com*

INTRODUCTION

Today, to protect corporate IT infrastructure from modern cyber threats, it is necessary to constantly test the security of its own systems. In this test we should include the methods used by real intruders, as purely speculative assessments of security are no longer relevant. Accordingly, modern students specializing in information security must own methods for penetration testing or so-called ethical hacking.

What is the best way to train specialists in these methods, if the single testing methodology accepted by the majority of experts in security does not exist yet? Basically we can see only specific hacking techniques that can be included in the test plan and no more. Many courses devoted to this subject are aimed at studying certain types of attacks and the corresponding tools used for their implementation. There is no common understanding among specialists and how the results of penetration testing should be presented in the report to the customer.

In the developed course, we tried to take into account all these aspects and using practical experience in the implementation of numerous penetration testing projects to provide students with various approaches to security testing and their combination in the form of an integrated methodology for testing security.

The approaches considered include:

Classic penetration test. In the course in testing for penetration by experts, vulnerabilities are searched and exploited using tools and techniques used by attackers. The main advantage of the approach is the ability to identify real attack vectors; the main drawback is the possibility of disrupting the availability of services and systems.

Vulnerability assessment. While performing vulnerability assessment, special software is used — vulnerability scanners. The scanner determines the versions of network services and checks the availability of information in its updated database of known vulnerabilities, which are typical for these versions of the vulnerability services. The main advantage: high speed of audit, the main drawback — the “linear” logic of the scanner and the inability to detect non-trivial attack vectors.

Configuration analysis. During the analysis of the system configuration, various security settings of the system are checked, for example, the password policy, the access rights to the system resources, the availability of installed updates, etc. The main advantage: the ability to find vulnerabilities, associated not only with errors in the development of software, but also because of errors made during administration.

Complex security testing. It is a combination of all the approaches discussed above and consists of the following phases:

1. Inventory of resources / search of targets for testing.
2. Vulnerability assessment.
3. Exploitation.
4. Privileges escalation.

The advantages of this approach are:

- identify the maximum number of vulnerabilities and non-trivial attack vectors;
- controlled risks of system failure;
- vulnerability verification.

Trying the complex approach to security, we developed the structure of our course, including the following 8 sections:

- Chapter 1. Introduction to the Penetration Testing.
- Chapter 2. Intelligence Gathering.
- Chapter 3. Vulnerability Analysis.
- Chapter 4. Exploitation.
- Chapter 5. Exploitation of Web-applications.
- Chapter 6. Social Engineering.
- Chapter 7. Exploitation Using Client-Side Attacks.
- Chapter 8. Maintaining Access.

In the first section of the course, students are introduced to modern approaches to documenting the results of penetration testing. It is assumed that, by performing laboratory tests throughout the course, students will constantly improve in the reports writing on the results of testing conducted, preparing new sections of their report.

Labs are conducted using open source solutions that are available to any university or student.

STUDYING ETHICAL HACKING

A. Basic Terms

The study of hacking techniques for understanding the approaches of ethical hacking.

The first section of the course introduces terminology from the field of ethical hacking, which is useful for assimilating the material of the whole course. Some of the terms will be considered in this article.

“White Hat” — an expert in information security, has the skills of hacking information systems and uses them to identify the vulnerabilities of systems to address them. Thus, the “white hat” is an expert-auditor who provides services for technical audit of systems, such experts are also called ethical hackers.

“Grey hat” — a specialist who works for those who pay. He can act both in the role of an auditor, and in the role of a real attacker, who decided to break the law.

“Black hat” — an intruder who earns a living by criminal craft for hacking systems.

Synonyms for the word “hacker” may also be terms such as “cracker” and “phreaker” (when hacker is hacking telephone systems).

B. Objectives of Hackers

The main purposes of the attacker are usually to gain access to certain information (for example, electronic correspondence of interested persons) or gain access (preferably administrative) to any system (for example, a client-bank).

This approach is also used to conducting penetration tests when experts are faced with similar tasks.

C. Tools

As for the tools used by the attackers, at present there is a quite wide variety: from freely distributed open-source utilities to full-scale hacking frameworks.

In general, it can be noted that the requirements for the skill level of a specialist in the use of hacker tools are constantly decreasing. If in the middle of the 2000s, in order to exploit the vulnerability, it was necessary to find the source code of the exploit, fix the specially introduced error (protection from “script kiddies”) and compile and get the executable file, now there are convenient GUI frameworks that require the user to only understand general information of the stages of the attack: the choice of purpose, exploit, payload, and so on.

Among the security testing distributives, the world standard de facto is the Kali Linux collection, which is a bootable medium or image from the Linux OS, containing hundreds of pre-installed tools for hacking. In Russia, Echelon Company is developing a similar security testing suite, Scanner-VS [5]. The complex has successfully passed the certification of FSTEC of Russia and the Ministry of Defense of Russia and is used for “combat” monitoring the state of security of information systems.

We will refer to a number of tools below and consider the relevant stages of the hacking methodology.

D. Review of the Hacking Methodology

In order to understand the techniques used by attackers to hack systems, it is enough to use the analogy with physical penetration into someone's territory in order to steal material assets.

How does a thief act? First of all, he conducts an investigation, the purpose of which is to collect information about the interesting assets to him and the steps that the rightful owner has taken to protect his assets. By collecting information and identifying gaps in the perimeter protection, the thief penetrates into the territory to pick up the prey. The attacker does not want to be caught and makes every possible attempt to hide his tracks.

A similar sequence is maintained even in the case of computer hacking, but sometimes with one peculiarity: since we are talking about information assets, theft does not lead to the disappearance of them from the owner, and accordingly, an attacker may have the desire to repeatedly steal something valuable, but to establish a permanent and comfortable access to the system, which he can use at any time.

So, let's outline the stages of penetration:

1. Information gathering (intelligence).
2. Identify vulnerabilities.
3. Operation and conduct of attacks.
4. Providing comfortable access.
5. Expansion of the zone of influence.

E. Collection of Information

What information does the hacker collect to be inside a corporate network of a particular organization?

First, information about the IT infrastructure: IP addresses, domain names, software and hardware used. Secondly, information about employees: names, contact information (both personal and work), hobbies, etc.

The sources of this information can be:

- website of the organization;
- database of registrars of domain names;
- sites of suppliers working with this organization (interested in press releases about completed projects);
- social networking sites, especially professional ones, where people, in fact, post their resumes;
- sites for finding work / employees (interesting job descriptions).

As a result, an attacker gets an idea of the purpose, has a list of interesting IT resources, and the names of employees of the organization.

Separately, it is necessary to mention the so-called social engineering. In fact, this is the use of such a science as psychology to the detriment of man. Hackers try to provoke a person to the necessary action with the help of various psychological techniques, for example, directly or indirectly disclose information.

Classics of the genre is the access to the IT administrator through a professional forum. So, for example, knowing the name of a person, his "nickname" on the Internet (as we know, most IT professionals have their own unique nicknames), an attacker enters into discussion with a victim at a technical forum. The administrator can search the forum for a solution to any technical problem. Our attacker can help solve it, but his help will lead to gaining access to the desired computer network. For example, an administrator can be given a self-contained program that fixes the problem, and at the same time performs the actions "necessary" for the attacker.

Nmap

Nmap (Network Mapper) is a security scanner used to discover hosts and services on a computer network, thus creating a "map" of the network. The software provides a number of features for probing computer networks, including host discovery and service and operating system detection. These features are extensible by scripts that provide more advanced service detection [10], vulnerability detection [10] and other features. Nmap is also capable of adapting to network conditions including latency and congestion during a scan. Nmap was originally a Linux-only utility [11], but it was ported to Windows, Solaris, HP-UX, BSD variants. Linux is the most popular platform, followed closely by Windows. While Nmap is commonly used for security audits, many systems and network administrators find it useful for routine tasks such as network inventory, managing service upgrade schedules, and monitoring host or service uptime.

The output from Nmap is a list of scanned targets, with supplemental information on each depending on the options used. Key among that information is the "interesting ports table". That table lists the port number and protocol, service name, and state. The state is either open, filtered, closed, or unfiltered. Open means that an application on the target machine is listening for

connections/packets on that port. Filtered means that a firewall, filter, or other network obstacle is blocking the port so that Nmap cannot tell whether it is open or closed. Closed ports have no application listening on them, though they could open up at any time. Ports are classified as unfiltered when they are responsive to Nmap's probes, but Nmap cannot determine whether they are open or closed. Nmap reports the state combinations open|filtered and closed|filtered when it cannot determine which of the two states describes a port. The port table may also include software version details when version detection has been requested. When an IP protocol scan is requested (-sO), Nmap provides information on supported IP protocols rather than listening ports.

In addition to the interesting ports table, Nmap can provide further information on targets, including reverse DNS names, operating system guesses, device types, and MAC addresses.

The Nmap Scripting Engine (NSE) is one of Nmap's most powerful and flexible features. It allows users to write simple scripts to automate a wide variety of networking tasks. Those scripts are then executed at the same time and have the speed and efficiency you expect from Nmap. Users can rely on the growing and diverse set of scripts distributed with Nmap, or write their own to meet custom needs.

NSE was designed to be versatile, with the following tasks in mind:

- network discovery;
- more sophisticated version detection;
- vulnerability detection;
- backdoor detection;
- vulnerability exploitation.

Maltego

Maltego [9] can be used for the information gathering phase of all security related work.

What Maltego offers to both network and resource based entities is the aggregation of information posted all over the internet — whether it's the current configuration of a router poised on the edge of your network.

Maltego allows you to enumerate personal information, linking a particular person with a company, e-mail addresses, websites, social networking groups, and phone numbers. It also facilitates passive and active reconnaissance of whois information, domain names, DNS information, IP addresses, and netblocks.

After gathering the information, Maltego allows you to identify key relationship between the information gathered.

Maltego is a tool that can graphically display the links between data, so it will make it easier to see the common aspects between pieces of information.

F. Identify Vulnerabilities

A vulnerability is a certain flaw that allows the threat to be realized. In the case of network security, good examples of vulnerabilities are:

- use of default passwords for system administration (for example, admin: admin);
- buffer overflow errors committed by programmers during development, leading to the fact that arbitrary code can be executed on a remote machine;
- no filtering of user input; etc.

There are different classifications and vulnerability registries. A good review of this topic is given in the article by Markov A. and Fadin A. [7].

It is also necessary to consider the vulnerability of the "zero day". In fact, this is a vulnerability that the vendor does not know yet and for which there is no corresponding patch. Information on the vulnerability of the "zero day" and the implementation of the program that exploits this vulnerability are a commodity in the hacker black market. The price of such goods varies from several thousand dollars to several tens of thousands.

Among experts, one can come across the view that a real hacking is possible only with the use of vulnerabilities known to a small number of persons. In fact, in a real IT infrastructure of a modern enterprise, vulnerabilities cannot be closed for years. Why? Because the installation of the patch can cause a malfunction in the operation of the systems and adversely affect the activities of the organization. In order to reduce this risk, respectable companies are implementing a rather expensive process of testing updates before they are distributed.

In the author's practice, there was a case where in a well-known large company a number of serious vulnerabilities, discovered during the penetration testing, "lived" in the system for a year after acquaintance of responsible or irresponsible IT specialists.

For hacking systems, you need information about software versions, using which you can find information about product-specific vulnerabilities. Typically, this

information can be found in two places: specialized vulnerability databases and vendor sites.

You can find out the version of the software in various ways. For example, many network services, when accessing them, show a so-called banner containing version information. This approach is used in the well-known NMAP port scanner. Sometimes the version can be determined analytically. For example, you can find a press release of the developer company or integrator who created the Internet portal, which must be cracked by an attacker. The press release often has all the necessary information about the technologies used, and comparing the date of this news release with information about the release dates of the corresponding product, it is easy to determine which versions were used. Knowing the version of the product, the attacker can find relevant information about the vulnerabilities and exploits available.

The real hacker performs these actions manually, and uses various techniques to ensure the secrecy of his activities: the use of proxy servers, cache of search engines, etc. This process can be automated with the help of special security scanners, the bulk of the checks are to identify the version and search information in a constantly updated database of its own.

The use of security scanners is associated with a large amount of very specific network traffic, which always causes suspicion among intrusion detection systems. Therefore, ethical hackers like these scanners, and unethical do not like.

The most famous vulnerability scanners are Nessus, Nexpose, OpenVAS.

1) Nessus:

Nessus is a proprietary vulnerability scanner which is developed by Tenable Network Security. It is free of charge for personal use in a non-enterprise environment.

According to surveys done by sectools.org, Nessus is the world's most popular vulnerability scanner, taking first place in the 2000, 2003, and 2006 security tools survey [12]. Tenable Network Security estimates that it is used by over 75,000 organizations worldwide.

Nessus allows scans for the following types of vulnerabilities:

- vulnerabilities that allow a remote hacker to control or access sensitive data on a system;
- misconfiguration (e.g. open mail relay, missing patches, etc.);

- default passwords, a few common passwords, and blank/absent passwords on some system accounts. Nessus can also call Hydra (an external tool) to launch a dictionary attack;
- denials of service against the TCP/IP stack by using malformed packets;
- preparation for PCI DSS audits.

Nessus provides additional functionality beyond testing for known network vulnerabilities. For instance, it can use Windows credentials to examine patch levels on computers running the Windows operating system, and can perform password auditing using dictionary and brute force methods. Nessus 3 and later versions can also audit systems to make sure they have been configured for a specific policy, such as the NSA's guide for hardening Windows servers.

The tools available in Kali Linux complement a Nessus installation, allowing everything to be in one place, easing maintenance and portability. Nessus reports on host discovery, vulnerability identification, and exploitability of a given vulnerability which are all important findings in the penetration testing process.

2) Nexpose:

Rapid7 Nexpose [13] is a vulnerability scanner which aims to support the entire vulnerability management lifecycle, including discovery, detection, verification, risk classification, impact analysis, reporting and mitigation. It integrates with Rapid7's Metasploit for vulnerability exploitation. It is sold as a standalone software, an appliance, virtual machine, or as a managed service or private cloud deployment. User interaction is conducted through a web browser.

Nexpose Vulnerability Scanner Community Edition (Nexpose CE) is a free vulnerability scanner from Rapid7 that scans devices for vulnerabilities. It can be integrated with the Metasploit exploit framework.

There are several of the Nexpose Community Edition features:

- vulnerability scanning for up to 32 IP addresses;
- regular vulnerability database updates;
- ability to prioritize the risk assessment;
- guide to remediation process;
- integration with Metasploit;
- no cost start-up security solution.

The commercial edition of Nexpose includes additional features, such as distributed scanning, more flexible reporting, web and database server scanning, and technical support.

3) OpenVAS:

OpenVAS is a vulnerability scanner that was forked from the last free version of Nessus after that tool went proprietary in 2005 [14]. OpenVAS plugins are still written in the Nessus NASL language. OpenVAS (Open Vulnerability Assessment System, the name of the fork originally known as GNessus) is a framework of several services and tools offering a vulnerability scanning and vulnerability management solution.

G. Exploitation of Vulnerabilities

Knowing for sure about the existing vulnerability or having only an assumption, the hacker will try to exploit it. First of all, of course, the attacker will look for ready-made and proven exploits in the case, if necessary he will develop it himself or ask a qualified colleague to do it.

One of the components of the exploit is the so-called “payload”. This is the functionality that runs on the remote machine to give an attacker what he needs. Typically, this is:

- obtaining the command line on the remote machine with administrator rights;
- introducing a keylogger (keylogger) for recording data entered by users of a remote machine;
- creating an account with a given password; etc.

One of the most popular frameworks for developing and using ready exploits is now the Metasploit Framework.

In addition to exploiting the various software vulnerabilities of one of the most effective attacks for several decades, there remains an attack on weak passwords, as passwords are selected by the weakest link — the person.

An IT-security specialist should be well versed in the various variations of such attacks. First of all, it should be noted that there are interactive attacks on passwords (online password cracking) and so-called offline attacks.

Interactive attacks consist in the fact that an attacker has access to the authorization functionality and consistently enters password login pairs and analyzes the

success of authorization. One of the most high-quality tools for carrying out such an attack is THC-HYDRA. In the case of offline attacks, an attacker has hash values of passwords, and the selection is performed without interacting with the system from which they were extracted.

The method of forming passwords distinguishes between the “brute force” and “dictionary attack” methods. In the first case, passwords are formed based on the specified rules. For example, an attacker from afar saw how the user entered the password and remembered that the password contains only Latin letters and numbers, and its length is 7–9 characters. Setting the selection for these parameters, the probability of successful selection increases dramatically. Attacks “by dictionary” allow you to try your luck and check if the user is using a common password. Recent password leaks clearly show that many users prefer to choose keyboard shortcuts (such as “qwerty”, “qazwsxedc”), phone numbers, dates, names, etc. A well-known utility of such a plan is Hashcat.

It is necessary to mention a few points about hashes of passwords. If a hash of the password is known, then it is not always necessary to crack it, sometimes it is quite easy to “authorize” it (“pass the hash” attack). If an attacker has access to hashes and the possibility of spoofing (for example, access to a web application at the database level), then you can simply replace the hash with one that is calculated for a known password. Password hashes do not have to be considered “on the go”; you can first generate them on specified character sets, and during the attack only perform a search, this type of attack is called an attack using rainbow tables.

Another interesting attack is the “man-in-the-middle” attack. As the most common illustration, consider the ARP-spoofing attack. As we know, the ARP protocol allows you to map the physical MAC addresses of devices to their IP addresses using the appropriate ARP tables stored by the participants in the network interaction.

The meaning of the attack is that changes are made to the tables of the victim machine and the resource with which the network exchange is interesting for the attacker (for example, the default gateway, domain controller), which makes it necessary to include an intermediary in the communication process: the intruder’s computer. The hacker, by using the sniffer, extracts the necessary data from the traffic: account names, passwords, password hashes, etc.

Separately, it is worth discussing attacks related to web-applications: Cross Site Scripting (CSS) and SQL injection (SQL injection).

In the case of CSS, a script is introduced into the web-site pages, which is executed in the user's browser when viewing this page. This can happen, for example, due to a programmer error that did not implement correctly data filtering.

For discovering the vulnerabilities in Web-applications IT-security specialists often use Web vulnerability scanners. Web vulnerability scanners suffer the common shortcomings of all scanners (a scanner can only detect the signature of a known vulnerability; they cannot determine if the vulnerability can actually be exploited; there is a high incidence of false-positive reports). Furthermore, web vulnerability scanners cannot identify complex errors in business logic, and they do not accurately simulate the complex chained attacks used by hackers.

In an effort to increase reliability, most penetration testers use multiple tools to scan web services; when multiple tools report that a particular vulnerability may exist, this consensus will direct the tester to areas that may require to manually verify the findings.

Kali Linux comes with an extensive number of vulnerability scanners for web services, and provides a stable platform for installing new scanners and extending their capabilities. This allows penetration testers to increase the effectiveness of testing by selecting scanning tools that:

- maximize the completeness (the total number of vulnerabilities that are identified) and accuracy (the vulnerabilities that are real and not false-positive results) of testing;
- minimize the time required to obtain usable results;
- minimize the negative impacts on the web services being tested. This can include slowing down the system due to an increase of traffic throughput. For example, one of the most common negative effects is a result of testing forms that input data to a database and then e-mail an individual providing an update of the change that has been made — uncontrolled testing of such forms can result in more than 30,000 e-mails being sent!

There is a significant complexity in choosing the most effective tool. In addition to the factors already listed, some vulnerability scanners will also launch the appropriate exploit and support the post-exploit activities. For our purposes, we will consider all tools that scan for exploitable weaknesses to be "vulnerability scanners." Kali provides

access to several different vulnerability scanners, including the following:

- scanners that extend the functionality of traditional vulnerability scanners to include websites and associated services (Metasploit Framework and Websploit);
- scanners that extend the functionality of non-traditional applications, such as web browsers, to support web service vulnerability scanning (OWASP Mantra);
- scanners that are specifically developed to support reconnaissance and exploit detection in websites and web services (Arachni, Nikto, Skipfish, Vega, w3af, and so on).

Vulnerability scanners are automated tools that crawl an application to identify the signatures of known vulnerabilities.

Kali comes with several different preinstalled web vulnerability scanners. Penetration testers will typically use two or three comprehensive scanners against the same target to ensure valid results. Note that some of the vulnerability scanners also include an attack functionality.

Vulnerability scanners are quite "noisy", and are usually detected by the victim. However, scans frequently get ignored as part of regular background probing across the Internet.

Kali comes with several client-side proxies, including Burp Suite, OWASP ZAP, Paros, WebScarab. They have many powerful features among their vulnerability scanners and are considered the best in this class of products. Since these complex products are described later in client-side proxies chapter, here we will analyze only the most important vulnerability scanners and here is there quick survey.

1) Arachni Web Application Scanner:

The Arachni web application scanner [15] is an intensive tool that runs from a web interface much akin to that of Tenable's Nessus. However, unlike Nessus, Arachni can only perform a scan against one host on one port at a time. If there are multiple web services running on a host and not serviced from the port, then repeated scans will have to be launched separately. For example, <http://www.random-company.com/> is hosting a web service on port 80 and phpMyAdmin on port 443 (HTTPS), the Arachni scanner will have to be run twice. It's not a fire-and-forget type of system. Arachni also has a highly configurable structure. The plugins and settings for Arachni allow for

precision scanning, and all plugins are enabled by default. Reporting is a snap and can be formatted in many different types of output.

2) Nikto:

Nikto [16] is a simple and straightforward scanner that looks for vulnerabilities on the web server and in web applications. Hosts must be scanned one at a time; however, with the output command it is easy to keep track of the scan summaries. Reports can be output to HTML, XML, CVS, NBE, and MSF to be exported to Metasploit. Many of vulnerabilities that are found with Nikto directly reference the Open Source Vulnerability Database (OSVDB).

3) Web Application Attack and Audit Framework — w3af:

Another scanner worth using is the Web Application Attack and Audit Framework (w3af) [17], a Python-based open-source web application security scanner. w3af is another lightweight intensive vulnerability scanner brought to the security community from the fine developers of OWASP. Reporting is limited and not as pretty as Arachni, but will provide a good basis for vulnerability reporting. The big advantage, or downfall depending on how a pentester is engaged on an assignment, is that w3af has a plethora of customizable vulnerability plugins that require updates from the Internet at the time the plugin is launched.

It provides preconfigured vulnerability scans in support of standards such as OWASP. The breadth of the scanner's options comes at a price — it takes significantly longer than other scanners to review a target, and it is prone to failure over long testing periods.

4) Vega:

The next step is to use more advanced scanners like Vega [18] that scan a larger number of vulnerabilities; in turn, they can take significantly longer to run to completion. It is not uncommon for complex vulnerability scans (as determined by the number of pages to be scanned as well as the site's complexity, which can include multiple pages that permit user input such as search functions or forms that gather data from the user for a back-end database) to take several days to be completed.

5) Websploit:

Websploit [19] is a ruby-based modular application that has the look and feel of Metasploit but it is designed specifically for direct attacks against web servers and social engineering. Websploit also has integration with Metasploit for payloads, exploits, and use of the Meterpreter handler. The application can scan and crawl websites then attack the web server through an automated exploitation module or cause DoS on demand. The interesting feature is using Websploit with the autopwn module.

6) WafW00f:

WafW00f [20] is a very useful python script, capable of detecting the web application firewall (WAF). This tool is particularly useful when the penetration tester wants to inspect the target application server and might get a fallback with certain vulnerability assessment techniques, for which the web application is actively protected by firewall. Thus, detecting the firewall sitting in between application server and Internet traffic not only improves the testing strategy, but also presents exceptional challenges for the penetration tester to develop the advanced evasion techniques.

7) Social Engineering and Social Engineering Toolkit (SET):

Social engineering is an act of manipulating people to perform actions that they don't intend to do. A cyber-based socially engineered scenario is designed to trap a user into performing activities that can lead to the theft of confidential information or some malicious activity. The reason for the rapid growth of social engineering amongst hackers is that it is difficult to break the security of a platform, but it is far easier to trick the user of that platform into performing unintentional malicious activity. For example, it is difficult to break the security of Gmail in order to steal someone's password, but it is easy to create a social engineered scenario where the victim can be tricked to reveal his/her login information by sending a fake login/phishing page.

The Social Engineer Toolkit [21] is designed to perform such tricking activities. Just like we have exploits and vulnerabilities for existing software and operating systems, SET is a generic exploit of humans in order to break their own conscious security.

Social Engineering Toolkit is a Python-based automation tool that creates a menu-driven application for us. Faster execution and the versatility of Python makes it the preferred language for developing modular tools like SET. It also makes it easy to integrate the toolkit with web servers. Any open source HTTP server can be used to access the browser version of SET. Apache is considered as the preferable server while working with SET.

SET was included in BackTrack 5 and the Kali Linux distribution.

One of the many great features within SET is its ability to clone a site for various Social Engineering attacks. SET can clone a site of your choosing or use one of its built-in templates.

From this point you can choose what attack vector you would like to use.

SET is a menu driven based attack system, which comprises:

- Spear-Phishing Attack Vectors.
- Website Attack Vectors.
- Infectious Media Generator.
- Create a Payload and Listener.
- Mass Mailer Attack.
- Arduino-Based Attack Vector.
- SMS Spoofing Attack Vector.
- Wireless Access Point Attack Vector.
- QRCode Generator Attack Vector.
- Powershell Attack Vectors.
- Third Party Modules.

H. Client-Side Exploitation

The greatest challenge for an attacker or an effective penetration tester is to bypass a target's security controls to achieve a compromise. This can be difficult when targeting systems located on a network because the attacker usually needs to bypass firewalls, proxies, intrusion detection systems, and other elements of a defense-in-depth architecture.

A successful workaround strategy is to directly target the client-side applications. The user initiates the interaction with the client application, allowing attackers to take advantage of the existing trust that exists between the user and the application. The use of social engineering methodologies will enhance the success of client-side attacks.

Client-side attacks target systems that typically lack the security controls (especially, firewalls and intrusion detection systems) are found on enterprise systems. If these attacks are successful and persistent communication is established, the client device can be used to launch attacks if it is reattached to the target's network.

The most known ways how to attack client-side applications are:

- Metasploit Framework attacks [22].
- The Cross-Site Scripting Framework [23].
- The Browser Exploitation Framework [24].

I. Gaining comfortable access

After the system is hacked and the hacker has access to it (for example, it can send commands to the operating system), its task is to ensure comfortable stable access. He can provide it himself, having a legitimate administrator's password or installing a so-called rootkit. A rootkit is a specially designed application or set of applications that allow an attacker to have secret administrative access to the system. Thus, the activity of the attacker remains hidden for standard OS monitoring tools.

J. Expansion of the zone of influence

Expansion of the zone of influence can be demonstrated on the following two examples.

The attacker got access to the web server at the level of the operating system. Such access allows him to find the configuration files of web applications running on this server and extract passwords from them to access databases located on other servers. With access at the database level, you can often get access to the OS, since the corresponding functionality can be left on.

The second example is related to obtaining a database of password hashes for domain controller users. The hacker selects passwords for user accounts (offline attack). The attacker checks the opened passwords on other systems and finds that a number of users use the same passwords to access various corporate resources.

Exploiting a computer, networking device or web service is great; however, the goal of most penetration tests is to maintain access to the compromised system. There are a number of methodologies for maintaining access to exploited victim systems; however, the overarching conclusion of every methodology is not to steal information but to reduce the time-consuming and exhaustive efforts

required to keep attacking the same machine over and over after it's already been compromised. If a security tester is working with a team, remote collocated servers or is in need of a secondary access point for a later access to the computer system, then efforts and expectation can be easily managed and further attacks can be more precise. We can consider here backdoors and rootkits [25].

Not to be confused with Trojan horses, a backdoor is a program that is left running on the compromised system to facilitate later entry without having to exploit the vulnerability again and again. While most Trojan horses contain a backdoor, a backdoor does not necessarily have to be part of a Trojan horse. Backdoors are applications or scripts that run like a Trojan horse but do not provide any functionality to the user of the compromised system. Backdoors can be created in several ways. Either by using root-kits, by opening a listening port on the target system, by letting the target system connect to your server, by setting up a listener for a certain packet sequence which in turn will open up a port.

Rootkits will allow you to have even more power than the system administrator has of the system. You will be able to control the remote system completely. Often rootkits also allow file, process and/or network socket concealment, while still allowing the individual in control of the rootkit to detect and use those resources. Rootkits should be customized to be able to completely cover the assessor's activities. In most cases if there is an antivirus patrolling, root-kits (usually on win32) will be detected before installation. So, modifying the root-kits is required in most situations. It's also important to notice that some rootkits won't work on different system setups.

There are at least five types of rootkits, ranging from those at the lowest level in firmware (with the highest privileges), through to the least privileged user-based variants that operate in Ring 3. Let us more deeply analyse user-mode and kernel-mode rootkits.

User-mode rootkits run in Ring 3, along with other applications as user, rather than low-level system processes. They have a number of possible installation vectors to intercept and modify the standard behavior of application programming interfaces (APIs). Some inject a dynamically linked library (such as a .DLL file on Windows) into other processes, and are thereby able to execute inside any target process to spoof it; others with sufficient privileges simply overwrite the memory of a target application.

Injection mechanisms among others include: exploitation of security vulnerabilities, interception of messages.

Kernel-mode rootkits run with the highest operating system privileges (Ring 0) by adding code or replacing portions of the core operating system, including both the kernel and associated device drivers. Many kernel-mode rootkits are developed as device drivers or loadable modules, such as loadable kernel modules in Linux or device drivers in Microsoft Windows. This class of rootkit has unrestricted security access.

Kernel rootkits can be especially difficult to detect and remove because they operate at the same security level as the operating system itself, and are thus able to intercept or subvert the most trusted operating system operations. Any software, such as antivirus software, running on the compromised system is equally vulnerable. In this situation, no part of the system can be trusted.

K. Hiding Traces

Hiding tracks can range from removing logs on hacked machines to hacking control consoles and databases of firewall, intrusion detection and SIEM systems.

L. DoS and DDoS attacks

Apart from the attacks, Denial of Service (DoS) attacks and their most widespread type Distributed Denial of Service (DDoS) stand apart.

It should be noted that DDoS attacks are associated with sending a large volume of traffic or requests from various sources to the target node. For example, an attacker instructs infected computers that are in the botnet they control to constantly open a specific web page. The victim's web server can not cope with this number of requests and stops responding.

A simple DoS attack can only be initiated by one node. Some time ago, the "slow HTTP request" attack was popular (Slow HTTP POST). The attack is implemented as follows: an attacker sends a request to the web server, indicating in a special header a large amount of data that the web server must accept. The web server prepares to process the data and allocates corresponding resources. Then the data transfer begins, but it is very slow, which allows you to use the server resources much longer than necessary, and as a result, interfere with the processing of other requests. Several thousands of such connections could easily make the web server inaccessible.

TESTING STUDENTS' SKILLS

The successful mastery of students' penetration testing skills, in our opinion, depends on the following factors:

1. Understanding the methodology for security testing. Because by knowing why one or another step is being carried out, a specialist can competently perform his work.
2. Practical skills of using hacking tools. Without skills in the use of vulnerability scanners, password brute-forcers, exploitation frameworks, it is practically impossible to test security.
3. Ability to structure information. This is necessary both for effective planning of security testing and for presenting test results in a report.

During the development of the course, we took into account all three factors. The course harmoniously combines an extensive theoretical part, which describes the methodology for testing security in general and specific testing techniques, as well as the practical part — laboratory work, within which a virtual network is created, in which practical skills of ethical hacking are developed. The method of conducting the course assumes that during the passage of laboratory work students create and constantly supplement the audit report.

SUMMARY

We examined what security testing is, what approaches exist, and what testing tools apply. Successful training of students in the testing of security is possible only in the case of a combination of studying the methodology of testing in general and gaining practical skills in the use of various tools. During the development of our course, all key success factors were taken into account. We are happy to present our students a practical and interesting course in testing the security of information systems.

SHORT DESCRIPTION OF THE DEVELOPED COURSE CONTENT

To ensure that students have mastered the methodology of integrated testing of security, we have developed course materials and laboratory works.

REFERENCES

- [1] **Dorofeyev A. V.** Status CISSP: kak poluchit i ne poteryat? Voprosy kiberbezopasnosti, 2013, No 1(1), pp. 65–68.
- [2] **Dorofeyev A. V., Markov A. S.** Menedzhment informacionnoj bezopasnosti: osnovnye koncepcii, Voprosy kiberbezopasnosti, 2014, No 1(2), pp. 67–73.
- [3] **Dorofeyev A. V.** Menedzhment informacionnoj bezopasnosti: upravlenie riskami, Voprosy kiberbezopasnosti, 2014, No 2(3), pp. 66–73.
- [4] **Dorofeyev A. V.** Menedzhment informacionnoj bezopasnosti: perehod na ISO 27001:2013, Voprosy kiberbezopasnosti, 2014, No 2(3), pp. 69–73.
- [5] <http://scanner-vs.ru>
- [6] **Andrew S. Tanenbaum, David J. Wetherall.** Computer Networks Prentice Hall; 5 edition (October 7, 2010), 960 p.
- [7] **Markov A. S., Fadin A. A.** Sistematika ujazvimoj i defektov bezopasnosti programmnyh resursov, Zashhita informacii. Inside, 2013, No 3, pp. 56–61.
- [8] **Shakhalov I. Yu., Dorofeev A. V.** Osnovy upravleniya informatsionnoj bezopasnost'yu sovremennoj organizatsii, Pravovaya informatika, 2013, No 3, pp. 4–14.
- [9] <https://www.paterva.com/>
- [10] Nmap Scripting Engine. Nmap.org. Retrieved on 2013-02-01.
- [11] The History and Future of Nmap. Nmap.org. Retrieved on 2013-02-01.
- [12] "sectools.org". Retrieved 2009-10-21.
- [13] <https://www.rapid7.com/products/nexpose/>
- [14] <http://www.openvas.org/>
- [15] <http://www.arachni-scanner.com/>
- [16] <http://www.cirt.net/nikto2/>
- [17] <http://w3af.org/>
- [18] <https://subgraph.com/vega/>
- [19] <http://sourceforge.net/projects/websploit/>
- [21] <https://github.com/EnableSecurity/wafw00f>
- [22] www.social-engineer.org/
- [23] <https://www.metasploit.com/>
- [24] https://www.owasp.org/index.php/OWASP_Xenotix_XSS_Exploit_Framework/
- [25] <http://beefproject.com/>
- [26] **Lee Allen, Tedi Heriyanto, Shakeel Ali.** Kali Linux — Assuring Security by Penetration Testing. Packt Publishing Ltd., 2014.

Digital Forensics Cases Overview

Anders Carlsson (course module leader), BTH, Sweden.
Andrey Chechulin (course module leader), SPIIRAS, Russia.
Benjamin Dorsch, BKA, Germany.
Andreas Kuehn, BKA, Germany.
Cemil Yesil, PP Westhessen, Germany.
Victor Völzow, Policeacademie Hessen, Germany.
Dr. Martin Mink, Policeacademie Hessen, Germany.

I. INTRODUCTION

Today digital devices play an important role in many areas. The increasing complexity of devices, networks and information systems lead to the growth of complexity of their forensics analysis. Possible financial, political and other benefits, which can be gained by cyber attacks, lead to substantial increase of the number and experience of potential malefactors. Moreover, nowadays almost every crime creates some traces in the digital world even if the crime is not directly connected to the information security. Despite these facts, the existing forensics analysis is a process which still depends not only on forensics experts but also on the experience of security administrators and even regular users. This fact defines the importance of the education in the area of digital forensics analysis.

The article is targeted at students, so the cases are described in an informal way. Together with Detective Smith students will investigate fictive criminal cases related to Linux, Mac OS X, Windows (7) operation systems and artifacts, file systems and memory. The description of the cases contains single steps of "how to" with illustrations. All necessary tools and the way how to use them are named and explained.

The article consists of the following sections: Section 2 contains main terms, definitions and standards that are used in the field of digital forensics; Section 3 presents four cases with digital forensics investigation (3.1 — Linux and OS X forensics, 3.2 — Windows artifacts forensics, 3.3 — FAT File system forensics, 3.4 — live forensics); Section 4 finalizes the article, highlighting the main outcomes; Sections 5-6 contain a list of tools, images and references.

II. MAIN TERMS, DEFINITIONS AND STANDARDS

A. Standards

There are several standards that specify the definitions and best practice

- ISO/IEC 27037:2012 — Information technology — Security techniques — Guidelines for identification, collection, acquisition and preservation of digital evidence;
- ISO/IEC 27042:2015 Information technology — Security techniques — Guidelines for the analysis and interpretation of digital evidence;
- ISO/IEC 27043:2015 Information technology — Security techniques — Incident investigation principles and processes;
- RFC 3227 — Guidelines for Evidence Collection and Archiving;
- ACPO Good Practice Guide for Digital Evidence.

B. Definitions and Terms

Digital evidence is the data that is used in judicial proceedings. Here "digital" means that it emanates from electronic devices such as computers and their peripheral devices, computer networks, mobile telephones, digital cameras, other mobile devices including data storage devices such as the Internet are all forms of electronic evidence.

Digital forensics is the discipline that focuses on identifying, preserving, extracting and analyzing of electronic evidence, and presenting it in court in an admissible way.

Forensics laws are the general forensics and procedural principles that should be used for legal gathering and analysis of forensics data. Specialists in each state should take local legal documents and regulations into consideration when making the forensics investigation.

Expert Witness Format (EWF) is a common forensics imaging format that supports compression, multiple hash sums and storage of meta data; file extension E01 (E02, E03, ...).

The Sleuth kit (TSK) is a collection of tools for forensics purposes that works on different image formats (Raw, EWF and other).

Regular Expression is a pattern search supported by many — but not only — Linux tools.

File slack [8] is the bytes after the logical end of a file and the end of the cluster wherein that file resides.

Master Boot Record (MBR) is the first data block on a disk. It contains the partition table and the boot loader. The MBR is limited to four primary partitions, and a single primary partition can hold an extended partition which then can be divided into so called logical partitions.

GUID Partitioning Scheme (GPT) is the part of the Unified Extensible Firmware Interface (UEFI) specification and the modern replacement for the Master Boot Record. The maximum number of partitions is 128.

More definitions and terms can be found here: <http://burgessforensic.com/computer-forensic-glossary/>

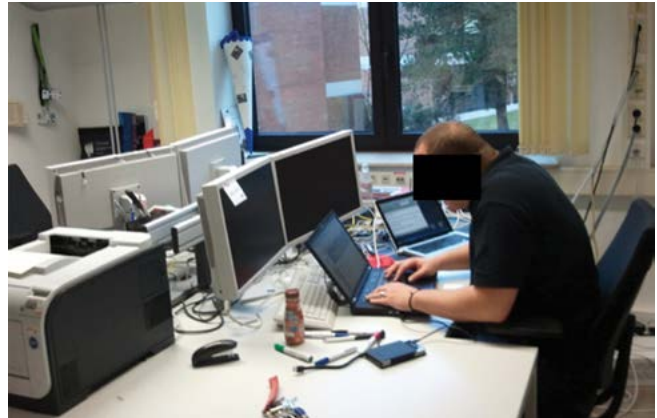


Fig. 1. Detective Miller

III. CASE STUDIES

A. Live and memory forensics case

1) Scenario

Detective Smith from the state security section of the police informs your unit about a confidential alert from the intelligence department. Mr. John DOE is suspected of being a high-ranking ☺ leader from the terrorist organization calling themselves “New World Union”. The recent alert indicates that this organization is planning a terrorist attack against a government building in your city to take revenge on the arrest of one of their “spiritual brothers” by your police force. A serious and trustable source supports this alert.

After obtaining a court order for a house search by Judge Miller, Detective Smith officially requests support from your unit. The experience of your unit is needed in order to seize forensically all electronic devices that can potentially serve as an evidence in front of court. You have been chosen to carry out this task.

Just before the Special Forces break open (было break in это незаконно взламывать) the door of Mr. DOE’s house Detective Miller tells you that further intelligence about Mr. DOE has indicated that he is an experienced system administrator working for an international company from his home office. Before you can even think about the implications of this information the special forces break the door leaving you with the following impression.

2) Benefits of the Live Data Forensics module

This scenario can only be solved by applying Live Data Forensics techniques. There are several computer systems on scene, some of them are powered on. In contrast to dead-box forensics, the danger of losing and changing data is extremely high when acquiring running computers. Since this is true, all principles for handling electronic evidence need to be obeyed, in particular, the first principle of the Electronic evidence guide (Council of Europe) which defines:

Principle 1 — Data Integrity

- No action taken should materially change any data, electronic device or media which may subsequently be used as an evidence in court.
- Electronic devices and data must not be changed, either in relation to hardware or software. The person, who is in charge of a crime scene or collecting the evidence, is responsible for maintaining the integrity of the material recovered and for ensuring the forensics chain of custody. Subsequent custodians of the devices and/or data must assume that responsibility.

When a data is accessed on a “live” computer system this must be done in such a way that causes the least impact on the data and by a person qualified to do so. Principles 2 to 5 are applied if this course of action is found to be necessary.

This means that in the scenario, special methods need to be applied that guarantee to cause the least impact on

the data. On the other hand, the scenario can only be solved when the volatile data are acquired while the systems are turned on. The screens of the suspect show the following information.



 *Fig. 2. Workplace*

Volatile data are defined as the data that are digitally stored in such a way that the probability is very high that they will be deleted, overwritten or altered within a short period of time due to human or automated interaction.

The following graphic shows the order of volatility:

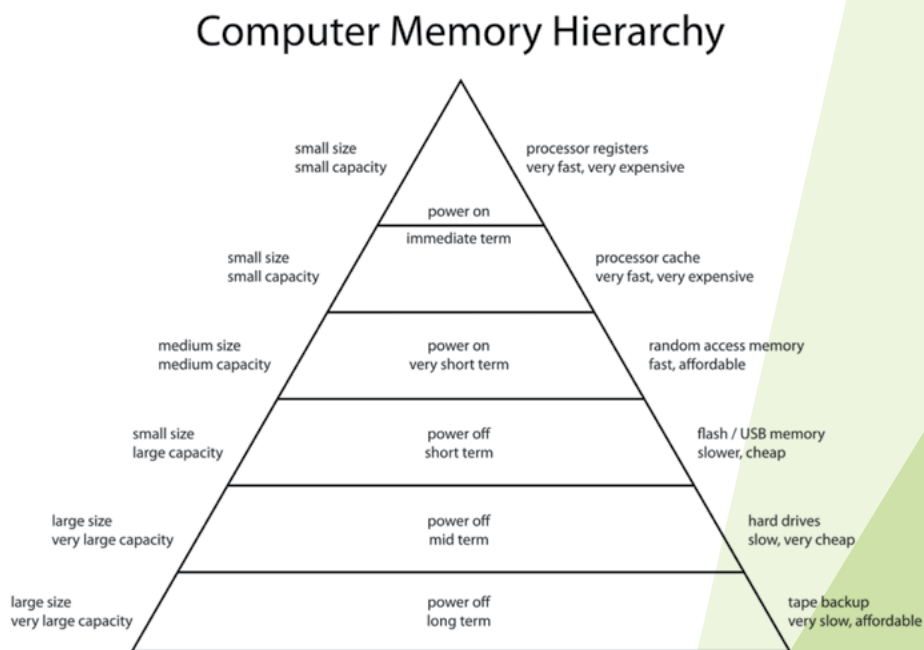


 *Fig. 3. The order of volatility*

Examples of volatile data that can be acquired from Mr. DOE's running computers are:

- Running processes;
- Running services;
- System information;
- Logged on users;
- Listening and open ports;
- ARP (address resolution protocol) cache;
- DNS cache;
- Auto-start information;
- Registry information not yet written to the disk;
- Unsaved documents;
- Binaries of process, services including those of Malware only residing in memory;
- Passwords, encryption keys, etc.

In this case, there were some opened, yet unsaved viewable documents, and BitLocker and FileVault full disk encryption was used by Mr. DOE to protect his hard drives. All of this information could be saved by acquiring forensically the memory of the computer system.

The acquired RAM dumps were analyzed in the forensics lab.

3) Benefits of the Memory Forensics Module

Memory forensics is all about analyzing previously acquired images of a memory dump of a computer. This memory dump has typically been acquired during the live forensics phase on scene.

In order to extract information from a computer memory it is vital to understand the structure of it. The management of the data of memory is the most important aspect that needs to be understood. The following graphics show a schematic representation of memory allocation:

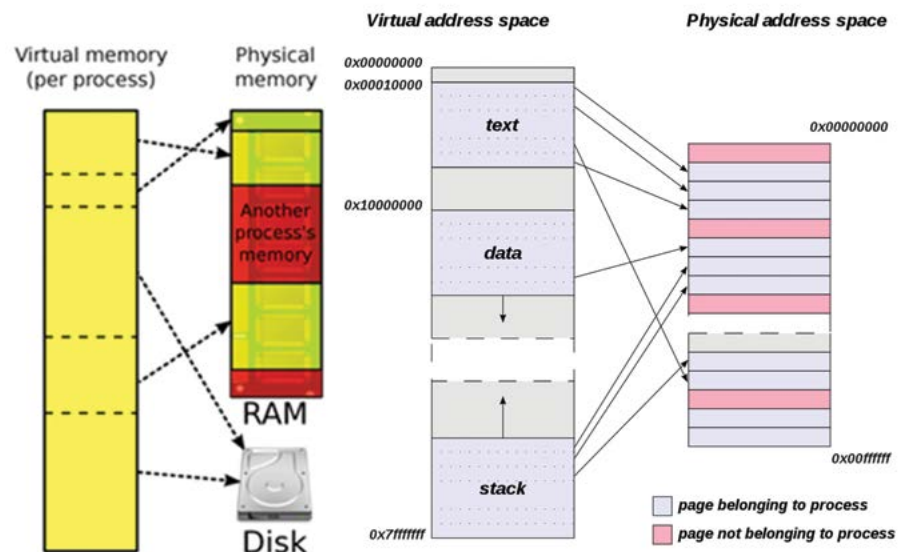


Fig. 4. Schematic representation of memory allocation

It is possible to extract process information from the RAM dump of Mr. DOE's computer using specialized tools for memory forensics. The following screenshot shows an extract of the output of such a specialized tool.

Offset(U)	Name	PID	PPID	Thds	Hnds	Sees	Mo
w64 Start	Exit						
0xfffffe001a5eb0900	System	4	0	174	0	-----	
0 2014-05-19 16:35:40 UTC+0000							
0xfffffe001ab1f0900	smss.exe	412	4	2	0	-----	
0 2014-05-19 16:35:40 UTC+0000							
0xfffffe001ab161900	csrss.exe	624	616	10	0	0	
0 2014-05-19 16:35:45 UTC+0000							
0xfffffe001abfa7900	wininit.exe	688	616	1	0	0	
0 2014-05-19 16:35:47 UTC+0000							
0xfffffe001ac557900	services.exe	764	688	4	0	0	
0 2014-05-19 16:35:48 UTC+0000							
0xfffffe001abf4d900	lsass.exe	812	688	6	0	0	
0 2014-05-19 16:35:48 UTC+0000							
0xfffffe001ab6c1900	svchost.exe	896	764	10	0	0	
0 2014-05-19 16:35:51 UTC+0000							
0xfffffe001ab71b900	svchost.exe	924	764	9	0	0	
0 2014-05-19 16:35:51 UTC+0000							
0xfffffe001ac166080	svchost.exe	360	764	16	0	0	
0 2014-05-19 16:35:52 UTC+0000							
0xfffffe001ac842900	svchost.exe	652	764	28	0	0	
0 2014-05-19 16:35:53 UTC+0000							
0xfffffe001ac55f900	svchost.exe	760	764	41	0	0	
0 2014-05-19 16:35:54 UTC+0000							
0xfffffe001ac83e900	svchost.exe	440	764	23	0	0	
0 2014-05-19 16:35:54 UTC+0000							
0xfffffe001ac83a900	svchost.exe	1088	764	32	0	0	
0 2014-05-19 16:35:56 UTC+0000							
0xfffffe001acc5a900	spoolsv.exe	1416	764	14	0	0	
0 2014-05-19 16:35:58 UTC+0000							
0xfffffe001acc42900	svchost.exe	1536	764	22	0	0	
0 2014-05-19 16:35:58 UTC+0000							
0xfffffe001acd12900	armsvc.exe	1644	764	2	0	0	
1 2014-05-19 16:35:58 UTC+0000							
0xfffffe001acd14900	AdminService.e	1684	764	6	0	0	
0 2014-05-19 16:35:59 UTC+0000							
0xfffffe001acd0c900	cmdagent.exe	1708	764	106	0	0	
0 2014-05-19 16:36:00 UTC+0000							
0xfffffe001acd09900	dragon_updater	1748	764	4	0	0	
1 2014-05-19 16:36:00 UTC+0000							

 Fig. 5. RAM dump analysis

For each of those processes an additional information, such as a process memory dump, can be extracted and analyzed. In addition to that, encryption keys from the full disk encryption can be extracted. The following screenshot shows the result of a scan for FileVault encryption keys on an OS X memory dump of Mr. DOE's Mac computer.

Volatility Foundation Volatility Framework 2.5	
Address	Volume Master Key
0xffffffff802da40000	8556f88b5308c3857ef4a0e0d53e70c7
0xffffffff8069746000	4d8266b26268909ca823581557b8bdb5

 Fig. 6. Encryption keys extraction

Those keys can then be used to decrypt or unlock the encrypted disk.

```
root@kali:~# fvdemount -e EncryptedRoot.plist.wipekey \
> -k 4d8266b26268909ca823581557b8bdb5 /dev/sdb2 /mnt/fvde
fvdemount 20151013

root@kali:~# mount -o loop,ro /mnt/fvde/fvde1 /mnt/MacHD/
root@kali:~# ls -l /mnt/MacHD
total 16
drwxrwxr-x 1 root  80  36 Jan 14 09:56 Applications
drwxr-xr-x 1 root root 39 Jan 14 10:05 bin
drwxrwxr-t 1 root  80  2 Sep 16 11:00 cores
dr-xr-xr-x 1 root root  2 Sep 16 11:00 dev
lrwxr-xr-x 1 root root 11 Feb  7 08:55 etc -> private/etc
dr-xr-xr-x 1 root root  2 Feb  7 09:06 home
-rw-r--r-- 1 root root 313 Aug  1 2015 installer.failurerequests
drwxr-xr-x 1 root root 59 Feb  7 09:03 Library
dr-xr-xr-x 1 root root  2 Feb  7 09:06 net
drwxr-xr-x 1 root root  2 Sep 16 11:00 Network
drwxr-xr-x 1 root root  6 Feb  7 08:59 private
drwxr-xr-x 1 root root 59 Feb  7 08:56/sbin
drwxr-xr-x 1 root root  4 Feb  7 08:48 System
drwxr-xr-x 1 501 root  2 Feb 13 13:23 [this is a test folder. Test.]
lrwxr-xr-x 1 root root 11 Feb  7 08:55 tmp -> private/tmp
drwxr-xr-x 1 root  80  5 Feb  7 09:48 Users
drwxr-xr-x 1 root root 12 Feb  7 08:56 usr
lrwxr-xr-x 1 root root 11 Feb  7 08:56 var -> private/var
drwxrwxrwt 1 root  80  5 Feb 13 10:33 Volumes
```

Fig. 7. Example of decryption

4) Benefits of the SSD forensics module

In the forensics lab, a solid state drive has been found in one of Mr. DOE's computers. This SSD needs special treatment and knowledge when being analyzed. The controller of the SSD will most likely change data on the flash memory as soon as it has power — even if it is connected to a write-blocker.

A challenge in the analysis of an SSD lies in the recovery of deleted data. On a combination of an SSD and an operating system which supports the TRIM command, the clusters that were occupied by a deleted file can be erased immediately as shown in the following representation.

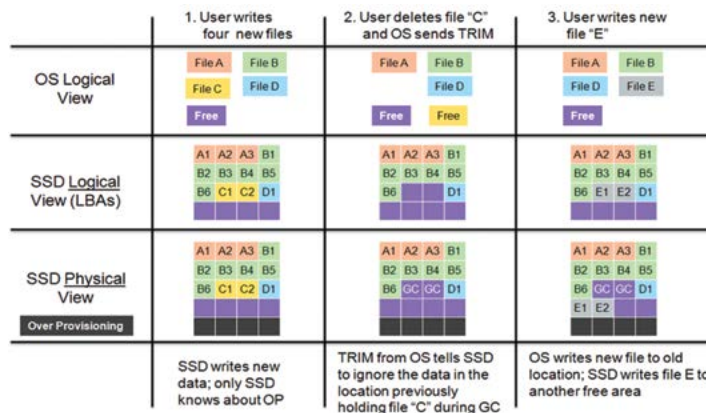


Fig. 8. SSD disk structure

Even though SSDs typically offer less possibility to recover deleted data, the SSD controller offers some additional features which can have beneficial effects for the forensics examination. An example for this is a write amplification feature as shown in the following schematic graphic.

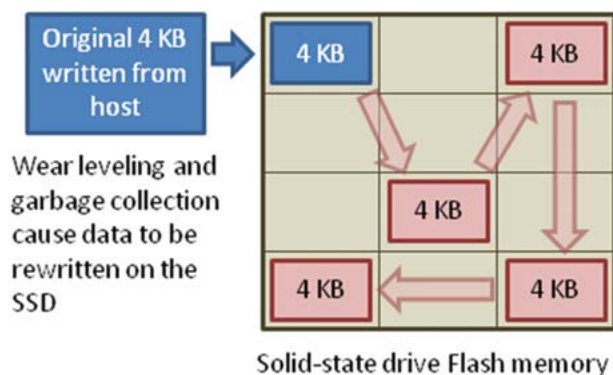


Fig. 9. Schematic graph of a write amplification feature

As shown in the graphic, there could be multiple instances of the same data which were only intended to be written once. Even if one instance of that data is overwritten, the other instances might still be on the chip.

In the case of Mr. DOE, the suspect deleted an evidential file from the SSD containing a list of equipment needed to perform a terrorist act against a government building. By disordering and reassembling the NAND chips and extracting the raw data, traces as well as fragments of the contents of this file could be recovered.

5) Summary

The terrorist case of Mr. DOE could be solved. To accomplish it the investigator had to have special knowledge in the areas of live forensics, memory forensics and SSD forensics. The steps taken can be summarized as follows:



Fig. 10. General steps of the investigation

After the volatile data like the RAM contents were taken while the suspect's computers were powered on during the house search, these data were analyzed. The particular attention was given to the analysis of the memory dump which contained not only pieces of an unsaved document but also encryption keys for a disk that had been fully encrypted. These keys were extracted and used to decrypt the disk. On that disk, the evidential data was found which pointed to an SSD drive which was also seized in the search. The analysis of the SSD drive was challenging because the recovery of the data deleted from an SSD is complicated due to the TRIM command. However, SSDs also offer positive effects for digital forensics. Those positive effects include the feature of write amplification which was the reason why in this case the investigator was able to recover traces of an evidential file which has been deleted by the suspect.

B. Linux and OS X Forensics Case

1) Scenario

During the live analysis of the computers of Mr. John Doe, the investigator discovers that on one of them there is an open terminal window with a command running:

```
# dd if=/dev/zero of=/dev/sda
```

As they note that it is a wiping command overwriting the internal hard disk, they press Ctrl-C to cancel the command. It turns out that the terminal is connected to a remote Linux machine, meaning that not the hard disk of the local machine was wiped but the one of the remote machines. Mr. Doe probably had been able to start the command before he was arrested. There must have been some interesting data on that machine! Using their Linux command line skills, the investigators identify the IP address of the remote machine. Opening of the IP address in a web browser shows the page of a web proxy¹ that one needs to register for to use; shortly afterwards the page does not react anymore. But using the IP address it is not difficult to determine the location of the remote machine: it is in a data center in a nearby city.

Next Detective Smith goes to the data center — after he got a search warrant from Judge Miller — to search the server that has the IP address that is investigated. The server is still running but has halted, probably because

¹ an intermediary server is the one — in this case — that can be used to surf web pages anonymously.

part of its file system was deleted. Since a live analysis is not possible he cuts the power and seizes the server.

In his office, Smith creates a forensics image in EWF format using a hardware write blocker. When he opens the image in his standard forensics software on Windows, no partitions or file systems are found. Looking at the image with a hex viewer on a Linux command line he discovers that the start of the image contains only zeroes:

```
# xxd /dev/sda | more.
00000000: 0000 0000 0000 0000 0000 0000 0000 0000 .....
00000010: 0000 0000 0000 0000 0000 0000 0000 0000 .....
00000020: 0000 0000 0000 0000 0000 0000 0000 0000 .....
00000030: 0000 0000 0000 0000 0000 0000 0000 0000 .....
00000040: 0000 0000 0000 0000 0000 0000 0000 0000 .....
00000050: 0000 0000 0000 0000 0000 0000 0000 0000 .....
00000060: 0000 0000 0000 0000 0000 0000 0000 0000 .....
00000070: 0000 0000 0000 0000 0000 0000 0000 0000 .....
00000080: 0000 0000 0000 0000 0000 0000 0000 0000 .....
:.
```

The wiping command deleted approx. the first 30 Megabytes of the disk — this means the partition table was overwritten (MBR resp. GPT) and the file system header of the first partition at least. The partition table could be reconstructed by information still present in the image (using file carving or specialized recovery tools such as TestDisk2 [3] to find the start of partitions by identifying file system headers). But reconstruction of the missing information of the file system header — needs to access the file system — is hard. Therefore, before he goes on with the examination, Detective Smith asks Mr. Doe about the web proxy. The only information he learns from him is the structure of two log files used on the proxy. The line of each of the log files has the following format:

Table I. Log file structure

Subscriber log file	Access log file
User ID (8 characters)	User ID (8 characters)
Name	Date of request
First name	Time of request
Email address	IP address of the user at the time of the request
Credit card number	URL that was requested

Smith also gets the information that the alert that originally led to the search of Mr. Doe's house was an e-mail sent via a web mailer. He also provides you with the content of the email (that includes the date when it was sent and the text which was sent).

It's not possible to access the file system but since both the log files and the e-mail sent are stored in text format, Smith decides to search for text strings in the image.

2) Benefits of the Linux Forensics Module

Linux comes out of the box with a variety of tools suitable for forensics analysis and can be upgraded with specialized forensics tools. Smith uses a standard tool, the grep command, to search for text in the image that is a part of the log files. The access log file contains the URLs accessed over the web proxy — he may find the URL of the web mailer here — and the User ID connects to the subscriber log file where personally identifiable information, such as the credit card number, is stored (see Table ENTER_REFERENCE_TO_TABLE_HERE).

Since the image contains a lot of binary (i.e. non printable) data, Smith first applies the strings command to extract readable strings. He decides to search for IP addresses — since there is an IP address entry in both log files — using a regular expression. A regular expression is a pattern search supported by a lot of Linux commands including the grep command that Smith is going to use. The command

```
# strings case214-proxy.E01 | egrep '([0-9]{1,3}\.){3}[0-9]{1,3}' > logs
```

first extracts the printable strings (“strings” command) and then searches each line of output for the pattern of an IP address (“grep” command; pattern: 1 to 3 digits followed by a dot, repeated three times, followed by 1 to 3 digits) and saves the output in the file logs (“> logs”).¹ Looking inside the logs file, Detective Smith notices that each line is either from the subscriber log file or from the access log file.

What is next? Using grep Smith searches the log file for entries of the access log file that contain both the date when the e-mail was sent and the URL of the web mailer which was used to send the e-mail. He gets three matches, all of them with the same User ID 94603814. Now he has to search the subscriber log file for the entry with that User ID. Since the number that represents the User ID could be a part of other text strings he makes grep search for full words only (-w option):

```
# grep -w 94603814 logs
```

The search yields one match which is the entry of the subscriber log file for that User ID. The entry contains a credit card number which Smith uses for a request to the credit card company for information on that customer. In some days he gets the answer to the request. It turns out that the customer is Mr. Doe! He might have sent the e-mail himself with one of his computers. Smith will be able to prove that if he finds traces of the e-mail on Doe’s computers. Since the Macintosh computer seems to be the one he mostly used he’ll start searching on that one.

3) Benefits of the OS X Forensics Module

The e-mail was sent via a web mailer using an Internet browser. Safari is the web browser preinstalled on OS X, the operating systems used on Macintosh computers. Most users of OS X use Safari, so Smith starts by searching the Internet history of Safari on the forensics image he acquired from Mr. Doe’s Mac-

¹ Note: the regular expression used here will also find non-valid IP addresses such as 501.23.256.999, but it is not a pattern too complex; “grep -E” is the grep command supporting extended regular expressions.

intosh laptop. To access the files on the image he first has to mount the file systems on the partitions of the image. On Linux or OS X it is possible by using the command line tool `ewfmount` [6] to convert the image from EWF to raw format and then mount a file system using the `mount` command. An alternative way would be to use the tools of The Sleuth kit (TSK) which can access the file system — and much more — of an image without the need to mount the file system.

After mounting, the first thing Detective Smith is able to collect, is the information of the operating system: the version of OS X is stored in `/System/Library/CoreServices/SystemVersion.plist`, the list of users of the system can be taken out of the directory `/private/var/db/dslocal/nodes/Default/users/` (each user has a plist file there, named after the user's name with information on that user including the password hash). In this case, Smith has an OS X 10.10 (Yosemite) and only one regular — i.e. non-system — user. This must be the user used by Mr. Doe.

Starting with Safari Version 8 (on OS X 10.10) the Internet history of a Safari user is stored in the file `History.db` in the directory `Library/Safari/` of a user's home directory (it was in the file `History.plist` before). Smith opens the `History.db` file of Mr. Doe's user with a SQLite viewer and search through the tables and entries in the database. But neither there nor in the other places — such as the directory `Library/Caches/com.apple.Safari/` — Smith can find any traces of the web mailer URL. And there is no other web browser installed that could have been used.

To make sure he did not overlook anything he is going to try a different approach: as with the proxy server he is going to search for text strings in the raw image. He uses the `grep` command and some suitable key words from the e-mail that was sent over the web mailer:

```
# strings -td case214-macbook.E01 | grep -f keywords.txt
```

The option “-td” of strings prints the offset (in bytes) where the hit was found in the image at the beginning of each output line, the option “-f keywords.txt” of grep reads the key words to search from the file `keywords.txt` (one per line) and “> hits.txt” saves the results in file `hits.txt`.

The search for key words yields a byte offset for each hit. To find out to which file in the file system the offset belongs or whether it is unallocated Smith decides to go the following way:

1. Byte offset → Sector offset
2. Sector offset → Cluster offset
3. Cluster offset → Inode (MFT entry)
4. Inode → file name and content

Details for each step:

1. A sector is an address on a disk (first sector on the disk has address 0, second sector address 1, and so on). It holds a number of bytes (normally 512). Looking at the partition header Smith sees that a sector actually has 512 bytes. The sector number that contains the given byte offset can be computed by dividing the byte offset by the sector size => `byte offset / 512`.

2. A file system uses clusters to address the content of files. A cluster contains a number of sector (e.g. 4), the first cluster of a file system, i.e. partition, has address 0, the second cluster address 1, and so on. Looking at the file system header Smith sees that a cluster consists of 8 sectors. The cluster that contains the sector offset can be computed by subtracting the starting sector from the given sector offset and dividing the resulting number by the cluster size (in sectors) => $(\text{sector offset} - \text{first sector of partition}) / 8$
3. An inode (Unix file system) resp. MFT entry (NTFS) contains all information on a file, including the list of clusters that contain the file content. Given the clusters from step 2 the command "ifind" from The Sleuth kit (TSK), a collection of forensics tools that operate on forensics images, can be used to search for the corresponding inodes.
4. The inode contains information on a file, including file name and path and the list of data clusters. Given the inodes from step 3 the TSK tool "ffind" can be used to display the file name (including path) and the command "icat" to extract the data (i.e. the file content).

Since Smith has already had such a case once, he gladly remembers that at that time he created a script file. The shell script contains all the necessary commands and will do the conversions automatically — he just needs to supply the file name of the image, the file that contains the search words and the offset of the file system (i.e. partition).

The script outputs five file names. Looking at four of them Smith finds the key words he searched for, but none contains the text of the e-mail. The fifth extracted file doesn't contain the key word at all. First he wonders why this file was found, but then he has an idea: he looks at the slack of this file, i.e. the bytes in between the end of the logical file and the end of the last cluster the file resides in (so called "file slack"). The file slack can be extracted with the TSK "icat" command using the option "-s". And, bingo, inside the file slack he finds part of an HTML page of the web mailer that was used to send the e-mail and the text of the e-mail. The file was saved in the cache of the Safari web browser, but its clusters at some later point were overwritten with another file (the one he found with his script). But since the new file didn't use up all the space in the last cluster, the part of the original file was still present in this last cluster.

4) Summary

The investigator Detective Smith used his special knowledge in Linux, Mac OS X and file systems to solve this case.

He extracted information from a forensics image using standard Linux command line tools as strings and grep, and regular expressions to search for IP addresses. Because the start of the original disk had been wiped the employment of tools that are based on file system metadata wouldn't have been possible. Later in the course he used his knowledge in file systems to determine the file name belonging to certain byte positions of another image, and in the end succeeded in extracting crucial data from file slack space. On the Mac Operating System he was able to search the files containing the Internet browsing history.

C. Windows artifacts case

1) Scenario

One day the CEO of a big German company receives an e-mail from a direct competitor with an attached Windows-Word-Document "Secret_Document.doc". The document contains all critical data of his own company. Since the CEO has seen this document, he decides to start an investigation in his own company, to find the employee, who has written the document.

In spite of the fact that only three people in his company has access to that kind of data, the CEO decided to do an examination of these three computers in a digital forensics matter. Therefore, he asks Mr. Schulz from his IT-Security Unit for support.

Mr. Schulz gets a forensics image of a person who is saying that he/she could have written this secret document. Before Mr. Schulz starts the digital forensics examination, he asks for all necessary information he can get from this case. All names from the suspects, the secret document itself and the administrative rules of each employee in the IT infrastructure.

The secret document is written in a Word Document and it is signed with a nickname „Gandalf“. One of the employee says that he thinks that he has a Trojan on his computer, because the mouse is moving by itself and the computer would do strange things sometimes.

2) Analysis

Mr. Schulz now focuses on the computer of the employee, who has said that he thinks that his computer is compromised by others. Now, it could also be possible, that it is not an employee who has stolen the secret data and that the suspect is somewhere else, stealing all secret information of the company network with the help of a Trojan.

First of all, Mr. Schulz is mounting the forensics image, so that he is able to have a look at all logical data on the hard drive. He recognizes that Operating System Windows 7 is running on the PC. He is trying to find evidences in the logical system, so he is not looking into the deleted files. First of all, he is looking manually over the data, but he can't not find the document, he is looking for. Therefore, he needs to go deeper.

The employee says that he thinks that his computer was controlled remotely. Therefore, he is checking the Windows Event Logs.

Since Windows 3.1, the Operating System Windows provides so called Event Logs. Windows Event Logs store information handling with the Operating System, for example Event Logs of applications, system logs and security logs. The security Event Logs are activated by default in Windows Vista/7. All allowed or unauthorized logins are registered in these Event Logs. So Mr. Schulz concentrates on the Security Event Logs. One of the employees said that he thinks that someone is logged in his computer without his permission. The location for the Event Logs in Windows 7 is `%SystemRoot%\System32\winevt\Logs\*.evtx`.

The Windows Event Logs are constructed in these categories.

- protocol name;
- source;
- event ID;
- user;
- task category;
- computer;
- and so on.

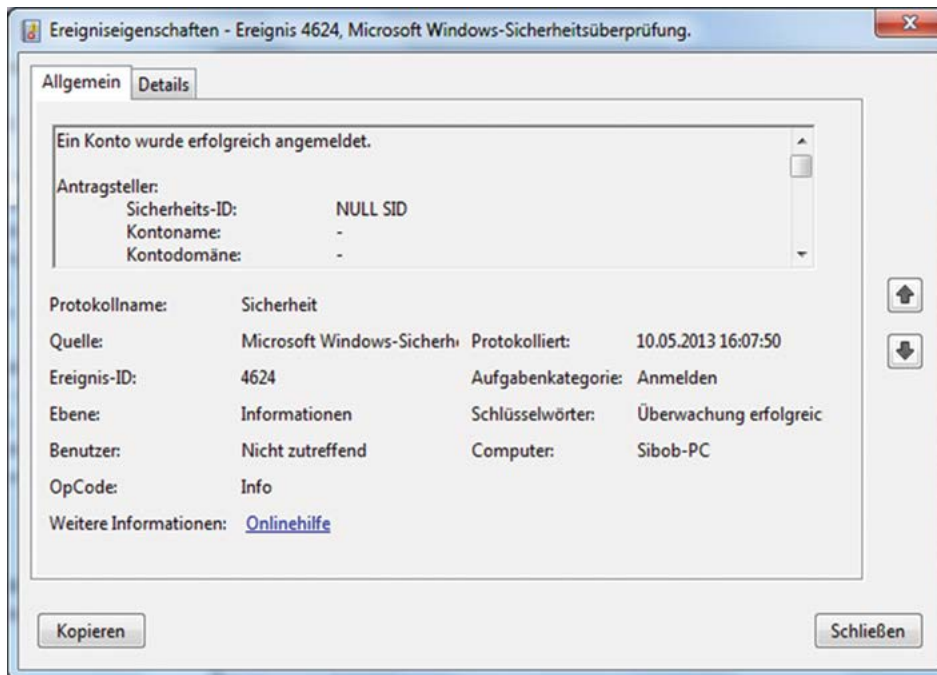


Fig. 11. Event Logs viewer

In these Event Logs, Mr. Schulz is looking for the Event ID 4624. This ID represents a successful login event.

There he couldn't find any useful information which proves that anybody else was logged in on the computer. After that he needs to check if someone tried to login on the computer without success. Therefore, he is using the command line program Log Parser with the following command:

```
LogParser "SELECT distinct SID, RESOLVE_SID(SID) as Username  
FROM Security WHERE EventID IN (529; 530; 531; 532; 533; 534;  
535; 537; 539)"
```

...without any result. There were also no unauthorized tries of logging in. The statement of the employee that someone else has logged in the computer remotely or is trying to use the computer for their purposes, could not be proved with the Windows Event Logs.

After looking into the Event Log files Mr. Schulz tries to find more indications. So he starts looking into the Browser Artifacts of the employee. There are two browsers installed on the computer, Internet Explorer and Mozilla Firefox.

First of all, he checks Internet Explorer artifacts. In Windows 7 these artifacts are stored in: `C:\Users\user\AppDataLocal\Microsoft\Windows\Temporary Internet Files\`

He recognizes very quickly that this file is empty. So he focuses on Mozilla artifacts. These artifacts are stored in Windows 7 in the folder:

```
%\Users\[Nutzer]\AppData\Roaming\Mozilla\Firefox\Profiles\[Profil-ID]\
```

There he can find a lot of different artifacts:

- addons.sqlite;
- Bookmarks.html;
- places.sqlite;
- cookies.sqlite;
- formhistory.sqlite;
- signons.sqlite.

For example, all visited websites are stored in places.sqlite. This information is stored in SQL-lite-format. Therefore, he needs a SQL-lite browser to view these data. To have an overview about all visited websites he uses the SQL-command:

SELECT url, last_visit_date, visit_count FROM moz_places

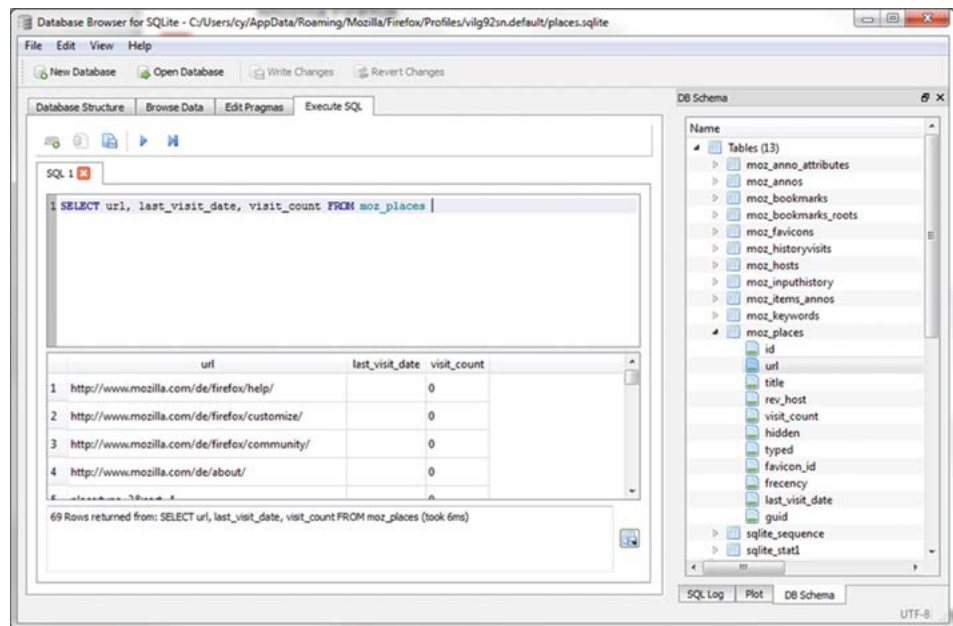


Fig. 12. Database browser

The table “visited websites” shows, which websites have been visited with this computer. Mr. Schulz notices that all websites from other competitors of his company are stored in the web history. Even the website from the company, which gave the hint about the secret document, is stored in the web history. Because of that, he is now looking deeper into the Browser Artifacts.

The next command Mr. Schulz uses should show up all data which is stored in the table “form”. There Mr. Schulz can find all web searches, which were typed with the computer of the suspect.

Therefore, he is using the SQL command on the SQL-Lite database formhistory.sqlite in the same directory:

```
SELECT fieldname, value, timesUsed, firstUsed, lastUsed FROM moz_
formhistory
```

The result shows what the employee was searching for:

- "competitors";
- "selling information";
- "competitors selling information";
- "how to become rich".

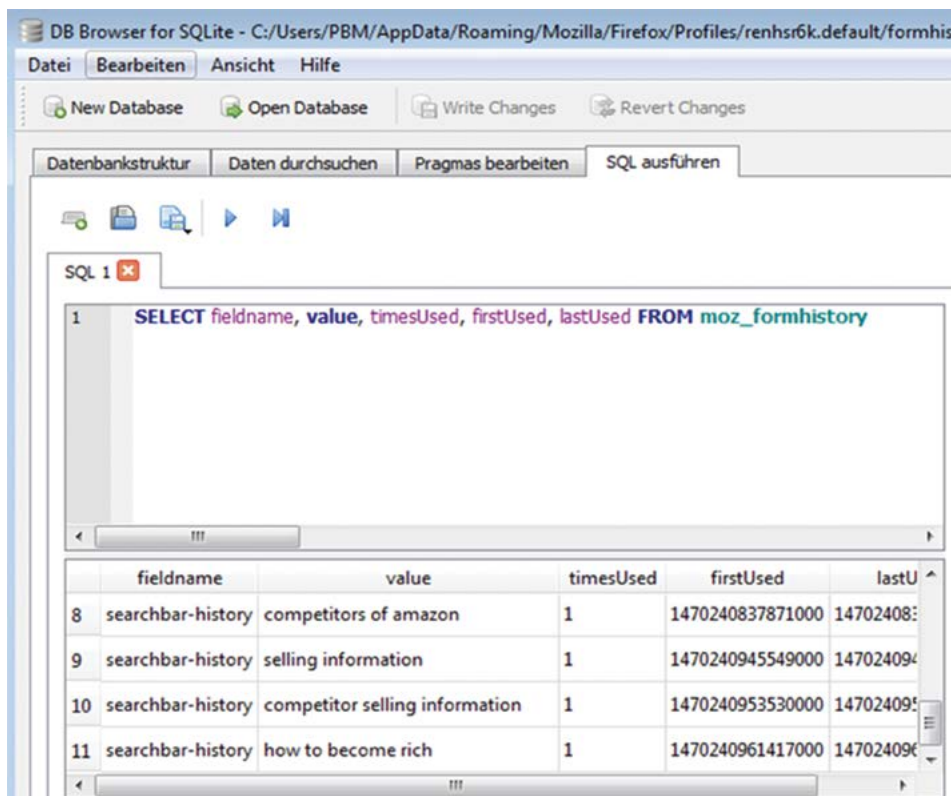
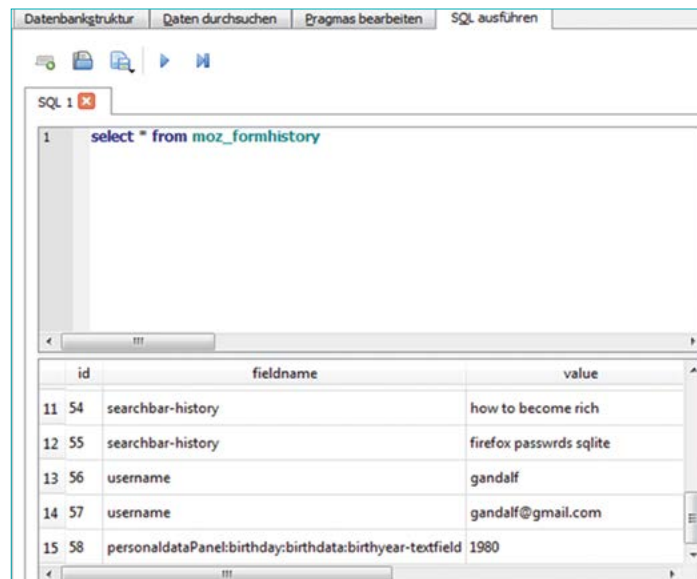


 Fig. 13. Database browser

In the same SQLite Database Mr. Schulz can find out that the user of this computer uses the e-mail address "Gandalf@gmail.com" and that he uses the nickname "Gandalf" for a forum.

```
Select * from moz_formhistory
```



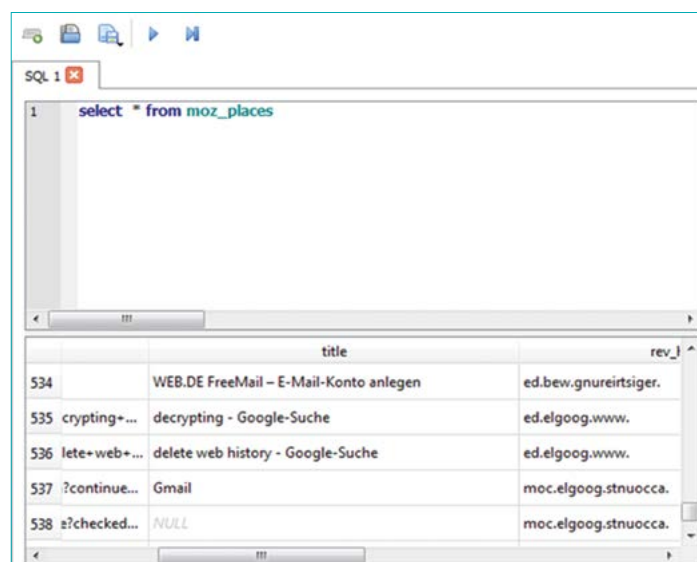
The screenshot shows a database browser window with the following tabs: Datenbankstruktur, Daten durchsuchen, Pragmas bearbeiten, and SQL ausführen. The SQL 1 tab is active, displaying the query: `select * from moz_formhistory`. Below the query, a table of results is shown with columns: id, fieldname, and value.

	id	fieldname	value
11	54	searchbar-history	how to become rich
12	55	searchbar-history	firefox passwrds sqlite
13	56	username	gandalf
14	57	username	gandalf@gmail.com
15	58	personaldataPanel:birthday:birthdata:birthyear-textfield	1980

Fig. 14. Database browser

In the web history Mr. Schulz can also see that the user was searching for how to hide artifacts on a computer. With this SQL-command he can see that the user was searching for how to hide or delete windows artifacts on the computer.

Select * from moz_places



The screenshot shows a database browser window with the following tabs: Datenbankstruktur, Daten durchsuchen, Pragmas bearbeiten, and SQL ausführen. The SQL 1 tab is active, displaying the query: `select * from moz_places`. Below the query, a table of results is shown with columns: id, title, and rev_j.

	id	title	rev_j
534		WEB.DE FreeMail - E-Mail-Konto anlegen	ed.bew.gnureirtsiger.
535	crypting+...	decrypting - Google-Suche	ed.elgoog.www.
536	lete+web+...	delete web history - Google-Suche	ed.elgoog.www.
537	?continue...	Gmail	moc.elgoog.stnuocca.
538	a?checked...	NULL	moc.elgoog.stnuocca.

Fig. 15. Database browser

It can be seen in the history that the user was logged in a Gmail account and attached a file called "Secret_Document.txt".

With the SQL-command

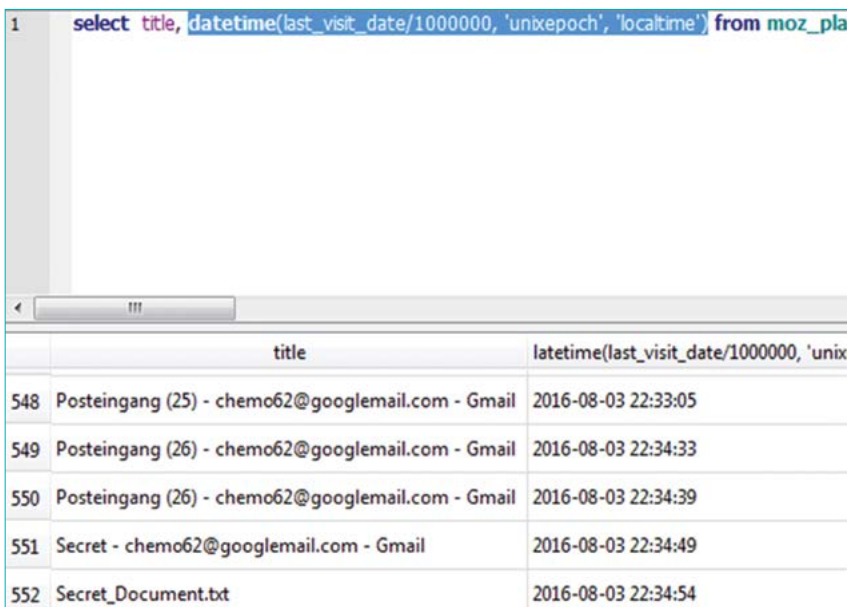
```
SELECT url, title, visit_count, last_visit_date FROM moz_places
```

it is possible to see all URLs, all titles from the website, the count of how often the website has been visited and the last time the website was visited.

Here, Mr. Schulz can see that the document was attached to a Gmail webmail e-mail.

To convert Firefox Timestamp in a human readable form, the SQL command should look like this:

```
SELECT url, title, visit_count, datetime(last_visit_date/1000000, 'unixepoch', 'localtime') FROM moz_places
```



	title	latetime(last_visit_date/1000000, 'unix
548	Posteingang (25) - chemo62@googlemail.com - Gmail	2016-08-03 22:33:05
549	Posteingang (26) - chemo62@googlemail.com - Gmail	2016-08-03 22:34:33
550	Posteingang (26) - chemo62@googlemail.com - Gmail	2016-08-03 22:34:39
551	Secret - chemo62@googlemail.com - Gmail	2016-08-03 22:34:49
552	Secret_Document.txt	2016-08-03 22:34:54

 **Fig. 16.** Database browser

All this information is very suspicious. That's why Mr. Schulz is now trying to find more evidence for the existence of the secret document. In the data, which is visible for the user, the secret document cannot be found.

So he is looking now into some locations which are not very common for usual users.

There are Restore Points and Volume Shadow Copies.

Restore Points is a feature in the Windows operating system which allows the user to restore the system to a previous version of the operating system. This func-

tion was set in Windows to recover the system after problems. It was introduced with Windows ME. In prior Windows versions it was based on a file filter that watched changes for a certain set of file extensions and copied files before they were overwritten. Restore Points themselves can be of forensics importance because they represent snapshots of a computer Registry and system files.

In Windows 7 you will find the Restore Points on this location:

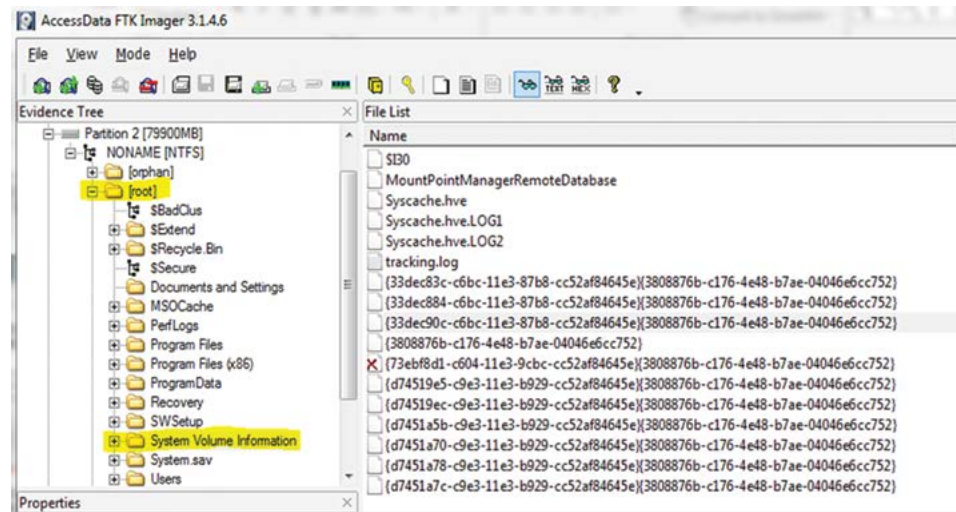


 **Fig. 17.** AccessData FTK imager tool

Mr. Schulz is now looking into the Restore Points. Maybe he is able to find some hints. Restore Point snapshots do capture the registry hive files.

In the registry, he is able to find a list of applications and filenames of the most recent files opened in windows.

HKEY_CURRENT_USER\SOFTWARE\Microsoft\Windows\CurrentVersion\Explorer\RunMRU

There he can find the document named "*Secret_Document.doc*". So, a document with that name was stored or opened on this computer.

Volume Shadow Copies:

The Volume Shadow Copy Service (VSS) was introduced with the Windows XP operating system and then provided in an extended version with Windows Server 2003, Windows Vista, Windows 7, Windows Server 2008 and Windows Server 2012 and Windows 10.

It is used for easy creation and deployment of version objects (so-called snapshots). As part of the hard disk capacity previously provided this service saves modifications to files and folders as part of up to 64 shadow copies, so older version levels.

The shadow copies themselves are read-only. The advantage of this technology is, primarily, the fact that users can recover overwritten older versions of a file independently without having to rely on an intervention by an administrator.

Additionally, the VSS can be used to create backups of consistent states of file systems or metadata that can otherwise be blocked by write operations (e.g. as the registry or the system databases).

The VSS automatically creates volume shadow copies at regular intervals, but they can also be created by installation of third party software or manually by the user. By examining a Shadow Copy, it is possible to view previous versions of a file, a directory or a volume.

Mr. Schulz virtualizes the machine and checks in the live system if there are any Volume Shadow Copies available.

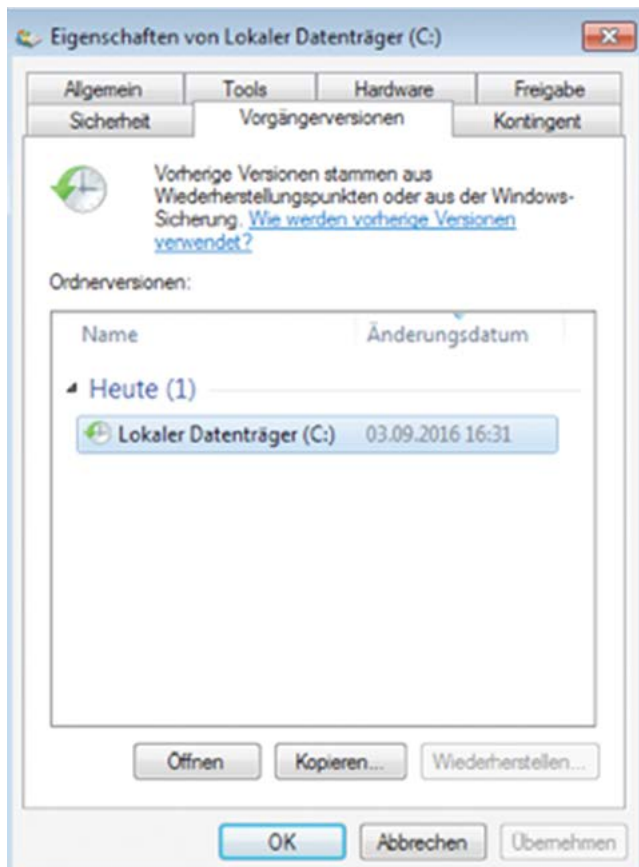


Fig. 18. List of the Volume Shadow Copies

There is also a possibility to have a look in the command line if there are any Volume Shadow Copies available. With this command Mr. Schulz can list all Volume Shadow Copies.

`C:\>vssadmin list shadows /for=C:`

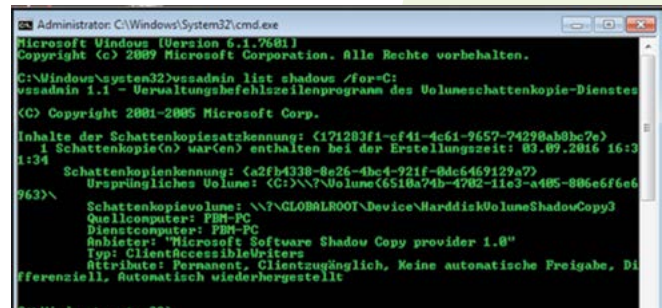


Fig. 19. List of the Volume Shadow Copies

Mr. Schulz opens the Volume Shadow Copy in the GUI and we can see all documents.

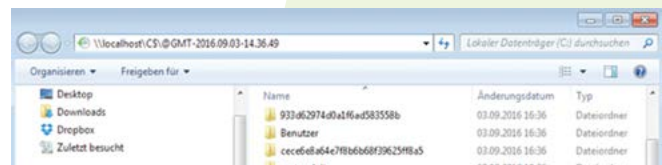


Fig. 20 Volume shadow copy files

In that Volume Shadow he is looking for the secret document. He can find the Word document "Secret_Document.doc" stored on the Desktop. The content and the metadata of this document match with the case.

3) Benefits of the Windows artifacts module

Think out of the box! All operating systems store information about the user's behavior. In some cases, it's easy to find the data you are searching for, for example, if the document you are looking for is stored on the Desktop. But in some cases, you need to think out of the box. Some data which can be interesting, even parts of it, can be found in artifacts. There are a lot of Application- and Operating-System artifacts which are produced during a user-session. The user himself doesn't recognize that his system produced artifacts but for a digital forensics point of view they could give the main reference for the case.

In our scenario, we saw that our searched files were not stored directly visible on the Desktop. At first glance it looked like there was no data which gave a hint to the case. However, after looking at the Windows artifacts, we found sufficient evidence for the case.

First of all, we used the Windows Event Logs to see if there were any abnormal behaviors on the computer. We couldn't find any clue that a stranger logged in the computer remotely.

After analyzing the Windows Events, we focused on the Browser artifacts. We were able to find some hints in the Browser Artifacts. For example, the search terms for deleting evidence, the nickname which was used in the document in a form and the document name itself.

After finding important hints in relation to the scenario we went deeper searching for more artifacts.

In the Volume Shadows, we could find the document we were searching for.

With the help of the Windows artifacts, we were able to establish without doubt that the document we were looking for had been sent from the computer we examined.

4) Summary

One of the main benefits of Windows artifacts is that you know that there are some that these could be a part of a puzzle which brings you to your aim. Some of these small artifacts may not show directly the data you are searching for, but all small different artifacts together can give you hints for the data you are searching for. All programs and the operating system by itself produces artifacts which can be important. These artifacts are mostly produced because of usability and bug fixing the programs. But these benefits can be also used to analyze the data. First of all, it needs to be clear that there are many artifacts produced on the system. Search in all Windows artifacts is not reasonable. You need to be clear about some facts before analyzing:

- which data you are searching;
- usage of the system;
- behavior of the user;
- communication platforms;
- storage behavior (cloud, external devices, etc.).

All these facts give us hints in which artifacts we should search for our data. As we showed above in our scanner, we were searching for a document which was sent via e-mail to another person. Therefore, we searched in the Browser artifacts, Windows Event Logs and in the Volume Shadow Copies for our data. The Browser artifacts and the Windows Event Logs gave us some hints about the document we were searching for. After we were able to find enough evidences which gave hints on the act, we went deeper. But the artifacts themselves can also give you some hints that the data you are searching for was or is not stored on the system you are analyzing.

D. FAT File system case

1) Scenario

A German company recognized that some of its employees found a way to betray the installed time registration system. Investigations were initiated to check if employees are registered correctly in the time-registration system.

On the spot, the IT-manager of the company informed the investigators, that the time recognition system of the company is located in a database on a server of the company. On a running system, the IT-manager demonstrated and explained, how the time registration system works, how it stores data and how this data can be revealed.

Afterwards, the investigators decided, to shut down the server, to seize the hard drive and to examine it later in the laboratory.

After the secure copy was put into the system of the laboratory and the image of the hard drive was uploaded into an analyzing program, the investigators noticed that the expected data structure of the server didn't occur. The mistake during seizing the hard drive can be excluded, because the hashes of both systems are equal

Name	Description	Size	Created	Modified	Record changed	Attr.	1st sector
(Stammverzeichnis)	existing	6,1 GB					19
Boot-Sektor	virtual (for examination purposes)	0,5 KB					0
Brachliegender Speicher	virtual (for examination purposes)						
Dateisystemschlupf	virtual (for examination purposes)	6,1 GB					2.880
FAT 1	virtual (for examination purposes)	4,5 KB					1
FAT 2	virtual (for examination purposes)	4,5 KB					10
Freier Speicher (net)	virtual (for examination purposes)	1,4 MB					

Fig. 21. Hard disk structure

That is why the shown data structure has to be analyzed for finding noticeable differences.

2) Analysis

Instead of the expected Master Boot Record, Sector 0 contains a Volume Boot Record of a Floppy Disc (FAT12).

Offset	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	ANSI ASCII
00000000	EB	3C	90	4D	53	44	4F	53	35	2E	30	00	02	01	01	00	e< MSDOS5.0
00000001	02	E0	00	40	0B	F0	09	00	12	00	02	00	00	00	00	00	à 0 0
00000002	00	00	00	00	00	00	29	F9	17	28	31	4E	4F	20	4E	41	)ù (1NO NA
00000003	4D	45	20	20	20	20	46	41	54	31	32	20	20	20	FA	0E	ME FAT12 ú
00000004	1F	A1	4C	00	A3	D3	7C	A1	4E	00	A3	D5	7C	A0	6E	04	;L ÉÓI;N ÉÓI n
00000005	A2	BD	7D	A1	13	04	48	A3	13	04	B1	06	D3	E0	2D	C0	om); HE ± Óà-À
00000006	07	A3	4E	00	A3	26	00	C7	06	4C	00	82	7C	C7	06	24	EN Éé Ç L ,lÇ \$
00000007	00	62	7D	BE	00	7C	8B	FE	8E	C0	B9	00	01	FC	F3	A5	b)M <pZÀ³ uóY
00000008	CD	19	80	FC	AA	75	01	CF	80	FC	02	75	45	83	F9	01	í eu*u Ieu uEfu
00000009	75	40	80	FE	00	75	3B	50	53	56	57	9C	2E	FF	1E	D3	u0Ep u;PSVWœ.y Ó
0000000A	7C	73	02	EB	36	26	81	BF	FE	01	55	AA	74	02	EB	27	la e64 ðp U*t e'
0000000B	26	80	BF	BC	01	C9	74	7A	E8	8B	00	E8	31	00	8B	F3	æz¼ Êtzè< èl <ó
0000000C	80	FA	79	77	1E	83	C6	02	BF	02	7C	B9	1E	00	32	F6	Eúyw fÆ z ³ 20
0000000D	EB	30	EA	F4	97	00	F0	B8	01	00	F8	5F	5E	5B	44	44	e0è0- 0, æ_^(DD
0000000E	CA	02	00	81	C6	BE	01	BF	BE	7D	B9	20	00	EB	13	B8	Ê æ¼ (¼)³ e .
0000000F	01	03	9C	2E	FF	1E	D3	7C	73	07	5B	B1	01	32	F6	EB	æ.y Ó s [± 20è
00000010	D6	C3	1E	06	1F	0E	07	FC	F3	A5	B9	01	00	BB	00	7C	ÖÄ uóY³ »
00000011	B8	01	03	9C	2E	FF	1E	D3	7C	72	EF	1E	07	2E	FE	06	, æ.y Ó xí .p
00000012	BD	7D	1F	EB	B2	05	14	07	1C	01	0C	75	16	1D	10	16	¼) e³ u
00000013	1E	FF	E8	11	00	5F	5E	5B	58	9C	2E	FF	1E	D3	7C	32	yè _^(Xœ.y Ó 2
00000014	F6	B1	01	CA	02	00	80	FA	79	77	14	26	8B	47	16	B6	0± Ê Eúyw æ<G ¶
00000015	01	3C	03	73	03	B1	03	C3	3C	07	73	03	B1	05	C3	B1	< s ± Å< s ± Å±
00000016	0E	C3	50	1E	33	C0	8E	D8	A0	17	04	24	0C	3C	0C	75	ÄP 3A20 \$ < u
00000017	1B	E4	60	3C	53	75	15	E4	61	50	0C	80	E6	61	58	E6	a'<Su æaP EæaXæ
00000018	61	B8	02	00	CD	10	B0	20	E6	20	CD	19	A0	6E	04	A3	a, í * æ í n É
00000019	06	00	A3	0E	00	0E	1F	3A	06	BD	7D	76	18	33	C0	CD	É : ¼)v 3Aí
0000001A	10	BE	25	7D	B4	0E	33	DB	FC	AC	3C	FF	74	06	34	55	¼)³ 30ü-<yé 4U
0000001B	CD	10	EB	F0	F4	1F	58	EA	87	E9	00	F0	C9	0E	00	00	í e00 Xæ±é 0É
0000001C	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
0000001D	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
0000001E	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
0000001F	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	5 AA

Fig. 22. Hard disk data

Sectors 1 till 12 contain only the characters „/x00“.

Sector 13 contains a Master Boot Record. This is expected only in Sector 1 and has no function in Sector 13.

Sectors 14 till 62 contain only the characters „/x00“.

The content of the data of Sector 63 is similar to a Volume Boot Record of a FAT32-System.

Offset	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	ANSI ASCII
000008A00	B	58	90	4D	53	57	49	4E	34	2E	31	00	02	08	20	00	EX MSWIN4.1
000008A10	02	00	00	00	00	F8	00	00	3F	00	FF	00	3F	00	00	00	ø ? ý ?
000008A20	18	E6	C1	00	6E	30	00	00	00	00	00	00	02	00	00	00	æÀ n0
000008A30	01	00	06	00	00	00	00	00	00	00	00	00	00	00	00	00	
000008A40	80	00	29	F7	0F	20	40	4E	4F	20	4E	41	4D	45	20	20	€)+ @NO NAME
000008A50	20	20	46	41	54	33	32	20	20	20	FA	33	C9	8E	D1	BC	FAT32 ú3ÉŽŃ*
000008A60	F8	7B	8E	C1	BD	78	00	C5	76	00	1E	56	16	55	BF	22	ø(ŽÁ×× Åv V U¿"
000008A70	05	89	7E	00	89	4E	02	B1	0B	FC	F3	A4	8E	D9	BD	00	Ń~ ŃN ± úó=ŽŮ×
000008A80	7C	C6	45	FE	0F	8B	46	18	88	45	F9	38	4E	40	7D	25	ÆP <F `EúSŃŃ)*
000008A90	8B	C1	99	BB	00	07	E8	97	00	72	1A	83	EB	3A	66	A1	<Á×× è- r fè:f;
000008AA0	1C	7C	66	3B	07	8A	57	FC	75	06	80	CA	02	88	56	02	f; ŠWuu €È `V
000008AB0	80	C3	10	73	ED	BF	02	00	83	7E	16	00	75	45	8B	46	€Å sí¿ f~ u€<F
000008AC0	1C	8B	56	1E	B9	03	00	49	40	75	01	42	BB	00	7E	E8	<V ` Iëu B» ~è
000008AD0	5F	00	73	26	B0	F8	4F	74	1D	8B	46	32	33	D2	B9	03	- s&°øOt <F23Ó¹
000008AE0	00	3B	C8	77	1E	8B	76	0E	3B	CE	73	17	2B	F1	03	46	;Èw <v ;İs +ñ F
000008AF0	1C	13	56	1E	EB	D1	73	0B	EB	27	83	7E	2A	00	77	03	V eŃs è'f~* w
000008B00	E9	FD	02	BE	7E	7D	AC	98	03	F0	AC	84	C0	74	17	3C	éý Ń~)-`ð-„Ät <
000008B10	FF	74	09	B4	0E	BB	07	00	CD	10	EB	EE	BE	81	7D	EB	ýt ' » İ eİ× }è
000008B20	E5	BE	7F	7D	EB	E0	98	CD	16	5E	1F	66	8F	04	CD	19	ä× }eäİ ^ f İ
000008B30	41	56	66	6A	00	52	50	06	53	6A	01	6A	10	8B	F4	60	AVİj RP S¿ j <ð`
000008B40	80	7E	02	0E	75	04	B4	42	EB	1D	91	92	33	D2	F7	76	€~ u `Bè ``30÷v
000008B50	18	91	F7	76	18	42	87	CA	F7	76	1A	8A	F2	8A	E8	C0	`÷v B*È÷v ŠòŠèÄ
000008B60	CC	02	0A	CC	B8	01	02	8A	56	40	CD	13	61	8D	64	10	İ İ, ŠVŃİ a d
000008B70	5E	72	0A	40	75	01	42	03	5E	0B	49	75	B4	C3	03	18	^r @u B ^ İu`Ä
000008B80	01	27	0D	0A	55	6E	67	75	65	6C	74	69	67	65	73	20	' Ungueltiges
000008B90	53	79	73	74	65	6D	20	FF	0D	0A	45	2F	41	2D	46	65	System ý E/A-Fe
000008BA0	68	6C	65	72	20	20	20	20	FF	0D	0A	44	61	74	65	6E	hler ý Daten
000008BB0	74	72	61	65	67	65	72	20	77	65	63	68	73	65	6C	6E	traeger wechseln
000008BC0	20	75	6E	64	20	54	61	73	74	65	20	64	72	75	65	63	und Taste druec
000008BD0	6B	65	6E	0D	0A	00	00	00	49	4F	20	20	20	20	20	20	ken IO
000008BE0	53	59	53	4D	53	44	4F	53	20	20	20	53	59	53	7E	01	SYSMSDOS SYS~
000008BF0	00	57	49	4E	42	4F	4F	54	20	53	59	53	00	00	55	AA	WINBOOT SYS U*

Fig. 23. Hard disk data

Offset	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	ANSI ASCII
000007E00	B	58	90	4D	53	57	49	4E	34	2E	31	00	02	08	20	00	EX MSWIN4.1
000007E10	02	00	00	00	00	F8	00	00	3F	00	FF	00	3F	00	00	00	ø ? ý ?
000007E20	18	E6	C1	00	6E	30	00	00	00	00	00	00	02	00	00	00	æÀ n0
000007E30	01	00	06	00	00	00	00	00	00	00	00	00	00	00	00	00	
000007E40	80	00	29	F7	0F	20	40	4E	4F	20	4E	41	4D	45	20	20	€)+ @NO NAME
000007E50	20	20	46	41	54	33	32	20	20	20	FA	33	C9	8E	D1	BC	FAT32 ú3ÉŽŃ*
000007E60	F8	7B	8E	C1	BD	78	00	C5	76	00	1E	56	16	55	BF	22	ø(ŽÁ×× Åv V U¿"
000007E70	05	89	7E	00	89	4E	02	B1	0B	FC	F3	A4	8E	D9	BD	00	Ń~ ŃN ± úó=ŽŮ×
000007E80	7C	C6	45	FE	0F	8B	46	18	88	45	F9	38	4E	40	7D	25	ÆP <F `EúSŃŃ)*
000007E90	8B	C1	99	BB	00	07	E8	97	00	72	1A	83	EB	3A	66	A1	<Á×× è- r fè:f;
000007EA0	1C	7C	66	3B	07	8A	57	FC	75	06	80	CA	02	88	56	02	f; ŠWuu €È `V
000007EB0	80	C3	10	73	ED	BF	02	00	83	7E	16	00	75	45	8B	46	€Å sí¿ f~ u€<F
000007EC0	1C	8B	56	1E	B9	03	00	49	40	75	01	42	BB	00	7E	E8	<V ` Iëu B» ~è
000007ED0	5F	00	73	26	B0	F8	4F	74	1D	8B	46	32	33	D2	B9	03	- s&°øOt <F23Ó¹
000007EE0	00	3B	C8	77	1E	8B	76	0E	3B	CE	73	17	2B	F1	03	46	;Èw <v ;İs +ñ F
000007EF0	1C	13	56	1E	EB	D1	73	0B	EB	27	83	7E	2A	00	77	03	V eŃs è'f~* w
000007F00	E9	FD	02	BE	7E	7D	AC	98	03	F0	AC	84	C0	74	17	3C	éý Ń~)-`ð-„Ät <
000007F10	FF	74	09	B4	0E	BB	07	00	CD	10	EB	EE	BE	81	7D	EB	ýt ' » İ eİ× }è
000007F20	E5	BE	7F	7D	EB	E0	98	CD	16	5E	1F	66	8F	04	CD	19	ä× }eäİ ^ f İ
000007F30	41	56	66	6A	00	52	50	06	53	6A	01	6A	10	8B	F4	60	AVİj RP S¿ j <ð`
000007F40	80	7E	02	0E	75	04	B4	42	EB	1D	91	92	33	D2	F7	76	€~ u `Bè ``30÷v
000007F50	18	91	F7	76	18	42	87	CA	F7	76	1A	8A	F2	8A	E8	C0	`÷v B*È÷v ŠòŠèÄ
000007F60	CC	02	0A	CC	B8	01	02	8A	56	40	CD	13	61	8D	64	10	İ İ, ŠVŃİ a d
000007F70	5E	72	0A	40	75	01	42	03	5E	0B	49	75	B4	C3	03	18	^r @u B ^ İu`Ä
000007F80	01	27	0D	0A	55	6E	67	75	65	6C	74	69	67	65	73	20	' Ungueltiges
000007F90	53	79	73	74	65	6D	20	FF	0D	0A	45	2F	41	2D	46	65	System ý E/A-Fe
000007FA0	68	6C	65	72	20	20	20	20	FF	0D	0A	44	61	74	65	6E	hler ý Daten
000007FB0	74	72	61	65	67	65	72	20	77	65	63	68	73	65	6C	6E	traeger wechseln
000007FC0	20	75	6E	64	20	54	61	73	74	65	20	64	72	75	65	63	und Taste druec
000007FD0	6B	65	6E	0D	0A	00	00	00	49	4F	20	20	20	20	20	20	ken IO
000007FE0	53	59	53	4D	53	44	4F	53	20	20	20	53	59	53	7E	01	SYSMSDOS SYS~
000007FF0	00	57	49	4E	42	4F	4F	54	20	53	59	53	00	00	55	AA	WINBOOT SYS U*

Fig. 24. Hard disk data

Sector 69 contains the similar data of Sector 63. It is a copy of Sector 63.

3) Summary

Starting with Sector 63, a correct file system (FAT32) is stored on the hard drive.

The missing Master Boot Record, including the jump to the start of the Partition at Sector 63, avoids the interpretation of a physical hard drive and the correct exposure of the file system with the included data.

The check of Sector 0 shows that there is a boot virus included.

The original Master Boot Record in Sector 13 has to be taken to its position of origin (Sector 0).

This could be done by virtual recreation of the hard drive.

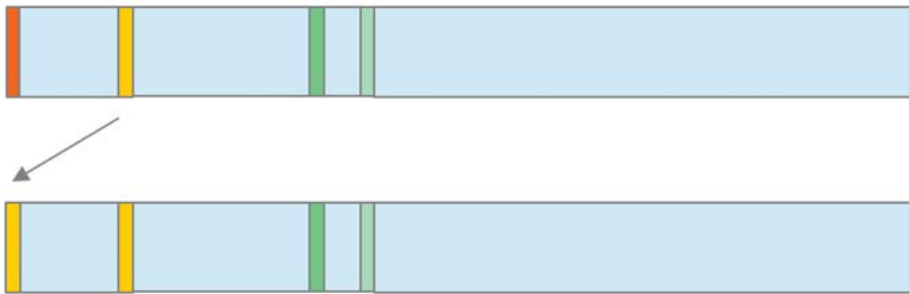


Fig. 25. Master Boot Record structure

Name	Beschreibung	Größe	Erzeugung	Änderung	Record-Änderung	Attr.	1. Sektor
(Stammverzeichnis)	existierend	6,1 GB					24.828
BONUS (2)	existierend	459 KB		02.12.2003 09:51:16			32.028
DRIVER (759)	existierend	41,6 MB		02.12.2003 09:51:16			32.964
GRAFIX0 (10)	existierend	23,1 KB		02.12.2003 09:52:54			121.564
PROGS (130)	existierend	32,3 MB		02.12.2003 09:52:54			121.660
RECYCLED (1)	existierend	631 KB	03.12.2003 07:34:16	03.12.2003 07:34:18			196.356
RUNTIME (8)	existierend	3,8 MB		02.12.2003 09:53:32			188.460
System Volume Information	existierend	0,6 KB	03.12.2003 07:34:16	03.12.2003 07:34:18		SH	196.364
2.HTM	existierend	0,6 KB		06.05.1998 16:04:42		A	27.332
3.HTM	existierend	1,3 KB		06.05.1998 15:57:38		A	27.340
4.HTM	existierend	2,9 KB		06.05.1998 16:00:14		A	27.348
5.HTM	existierend	0,7 KB		06.05.1998 15:12:02		A	27.356
ABOUT.HTM	existierend	1,4 KB		06.05.1998 15:12:10		A	27.364
ASPI2DOS.SYS	existierend	34,5 KB		05.05.1999 22:22:00		A	25.204
ASPI4DOS.SYS	existierend	14,0 KB		05.05.1999 22:22:00		A	24.996
ASPI8DOS.SYS	existierend	36,7 KB		05.05.1999 22:22:00		A	25.276
ASPI8U2.SYS	existierend	39,8 KB		05.05.1999 22:22:00		A	25.356
ASPICD.SYS	existierend	28,9 KB		05.05.1999 22:22:00		A	25.076
AUTOEXEC.BAT	existierend	1,1 KB		05.05.1999 22:22:00		A	24.836
AUTORUN.INF	existierend	49 B		22.03.1997 09:08:46		A	27.372
BONUS.HTM	existierend	4,9 KB		06.05.1998 15:23:06		A	27.380
BROWSER.EXE	existierend	740 KB		08.10.1997 17:12:00		A	27.396
BROWSER.INI	existierend	168 B		06.05.1998 16:03:34		A	28.876
BROWSR95.EXE	existierend	765 KB		25.09.1997 12:44:10		A	28.884

Fig. 26. Recovered disk structure

IV. CONCLUSION

The Article shows that all three common Operating Systems offer several possibilities to solve different forensics problems. Data which isn't available at first sight could mostly be recovered or detected if you have a deeper look into the system.

The aim of the article is to acquaint readers with different aspects and possibilities of "Digital forensics" in general. Also it provides readers with the examples of real-world case investigation by various aspects of digital forensics, including operation systems, file system and live forensics analysis. The cases are presented to provide readers with the knowledge of real-world case investigation by various aspects of digital forensics. Nowadays an analysis of such cases requires a special knowledge in both theoretical and practical areas of operation system architecture, security software, network services, hardware analysis, etc. The presence of such kind of experience will help it-specialists in their future life not only if they work in digital forensics area, but for preventing the traces from unpremeditated damage for their further analysis by forensics specialists.

V. LIST OF TOOLS AND IMAGES FOR EDUCATION

A. List of tools

The list of tools can be separated by the phases of forensics investigation.

1) Data collection phase

Tcpdump [2] is an open source command-line tool for monitoring (sniffing) network traffic. Tcpdump works by capturing and displaying packet headers and matching them against a set of criteria. It understands boolean search operators and can use host names, IP addresses, network names and protocols as arguments.

Dd [1] is a command-line utility for Unix and Unix-like operating systems primary purpose of which is to convert and copy files. On Unix, device drivers for hardware (such as hard disk drives) and special device files (such as /dev/zero and /dev/random) appear in the file system just like normal files; dd can also read and/or write from/to these files, provided that function is implemented in their respective driver. As a result, dd can be used for tasks such as backing up the boot sector of a hard drive and obtaining a fixed amount of random

data. The dd program can also perform conversions on the data as it is copied, including byte order swapping and conversion to and from the ASCII and EBCDIC text encodings [1].

2) Evidence extraction phase

Wireshark [4] is the software for network traffic analysis. It has a feature set which includes the following: deep inspection of hundreds of protocols; live capture and offline analysis; standard three-pane packet browser; multi-platform: runs on Windows, Linux, mac OS, Solaris, Free BSD, Net BSD and many others; captured network data can be browsed via a GUI, or via the TTY-mode TShark utility; read/write many different capture file formats: tcpdump (libpcap), Pcap NG, Catapult DCT2000, Cisco Secure IDS iplog, Microsoft Network Monitor, Network General Sniffer® (compressed and uncompressed), Sniffer® Pro, and Net X-Ray®, Network Instruments Observer, Net Screen snoop, Novell LANalyzer, RADCOM WAN/LAN Analyzer, Shomiti/Finisar Surveyor, Tektronix K12xx, Visual Networks Visual Up Time, WildPackets EtherPeek/TokenPeek/AiroPeek, and many others; capture files compressed with gzip can be decompressed on the fly; live data can be read from Ethernet, IEEE 802.11, PPP/HDLC, ATM, Bluetooth, USB, Token Ring, Frame Relay, FDDI, and others; decryption support for many protocols, including IPsec, ISAKMP, Kerberos, SNMPv3, SSL/TLS, WEP, and WPA/WPA2; output can be exported to XML, PostScript®, CSV or plain text.

Autopsy [5] is a digital forensics platform and graphical interface to The Sleuth Kit and other digital forensics tools. It has a feature set which includes the following:

- **Timeline Analysis:** displays system events in a graphical interface to help identify activity.
- **Keyword Search:** text extraction and index searched modules enable one to find files that mention specific terms and find regular expression patterns.
- **Web Artifacts:** extracts web activity from common browsers to help identify user activity.
- **Registry Analysis:** uses Reg Ripper to identify recently accessed documents and USB devices.
- **LNK File Analysis:** identifies short cuts and accessed documents.
- **Email Analysis:** parses MBOX format messages, such as Thunderbird.
- **EXIF:** extracts geo location and camera information from JPEG files.

- File Type Sorting: groups files by their type to find all images or documents.
- Thumbnail viewer: displays thumbnail of images to help view pictures quickly.
- Robust File System Analysis: support for common file systems, including NTFS, FAT12/FAT16/FAT32/ExFAT, HFS+, ISO9660 (CD-ROM), Ext2/Ext3/Ext4, Yaffs2 and UFS from The Sleuth Kit.
- Hash Set Filtering: filters out known good files using NSRL and flags known bad files using custom hashsets in HashKeeper, md5sum, and EnCase formats.
- Tags: tag files with arbitrary tag names such as 'bookmark' or 'suspicious' and add comments.
- Unicode Strings Extraction: extracts strings from unallocated space and unknown file types in many languages (Arabic, Chinese, Japanese, etc.).
- File Type Detection is based on signatures and extension mismatch detection.
- Interesting Files Module will flag files and folders based on name and path.
- Android Support: extracts data from SMS, call logs, contacts, Tango, Words with Friends and more.

3) Evidence analysis phase

NetworkMiner [7] is a Network Forensics Analysis Tool. NetworkMiner can be used as a passive network sniffer/packet capturing tool in order to detect operating systems, sessions, hostnames, open ports etc. NetworkMiner can also parse PCAP files for offline analysis and to regenerate/reassemble transmitted files and certificates from PCAP files. NetworkMiner makes it easy to perform advanced Network Traffic Analysis by providing extracted artifacts in an intuitive user interface. NetworkMiner can extract files and certificates transferred over the network by parsing a PCAP file or by sniffing traffic directly from the network. This functionality can be used to extract and save media files (such as audio or video files) which are streamed across a network from websites such as YouTube. Supported protocols for file extraction are FTP, TFTP, HTTP, SMB, SMB2 and SMTP. User credentials (usernames and passwords) for supported protocols also can be extracted by NetworkMiner. The credentials tab sometimes also shows information, such as user accounts for popular online services like Gmail or Facebook, that can be used to identify a particular person.

TestDisk [3] is a powerful free data recovery software. It is designed to help recover lost partitions and/or make non-booting disks bootable again when these symptoms are caused by faulty software: certain types of viruses or human error (such as accidental deleting of a Partition Table). It provides the following possibilities:

- fix partition table, recover deleted partition;
- recover FAT32 boot sector from its backup;
- rebuild FAT12/FAT16/FAT32 boot sector;
- fix FAT tables;
- rebuild NTFS boot sector;
- recover NTFS boot sector from its backup;
- fix MFT using MFT mirror;
- locate ext2/ext3/ext4 Backup SuperBlock;
- undelete files from FAT, exFAT, NTFS and ext2 filesystem;
- copy files from deleted FAT, exFAT, NTFS and ext2/ext3/ext4 partitions.

SANS Investigative Forensics Toolkit ("SIFT") [9] is a computer forensics VMware appliance that is pre-configured with all the necessary tools to perform a detailed digital forensics examination. It is compatible with expert witness format (E01), advanced forensics format (AFF) and raw (dd) evidence formats. The new version has been completely rebuilt on an Ubuntu base with many additional tools and capabilities that can match any modern forensics tool suite. It includes the following software:

- MantaRay (Automated Forensics Processing), MantaRay's GitHub;
- Sleuth Kit (file system analysis tools);
- log2timeline (timeline generation tool);
- ssdeep & md5deep (hashing tools);
- Foremost/Scalpel (file carving);
- Wireshark (Network Forensics);
- Vinetto (thumbs.db examination);
- Pasco (IE web history examination);
- Rifiuti (Recycle Bin examination);
- Volatility Framework (memory analysis);
- DFLabs PTK (GUI front-end for Sleuth kit);
- Autopsy (GUI front-end for Sleuth kit);
- PyFLAG (GUI Log/Disk examination).

4) Evidence presentation phase

Microsoft Power Point, Word and Open Office are tools for report writing.

B. Images + Challenges

For the self-study one can find the tasks, challenges, etc. on the following web-sites:

- <http://digitalcorpora.org/> — *Images & Challenges*
- <http://digital-forensics.sans.org/community/challenges>
- <http://www.honeynet.org/challenges> — *Challenges (including disk images, memory dumps and pcap files)*
- <http://old.honeynet.org/challenge/>
- <http://old.honeynet.org/scans/>
- <http://forensicscontest.com/puzzles> — *Network Forensics challenges*
- http://www.cfreds.nist.gov/Hacking_Case.html
- <http://www.dc3.mil/challenge/>

Most of the disk images, memory dumps and network packet captures available on these websites are free and can be used without additional authorization.

VI. ACKNOWLEDGEMENTS

This research was financially supported by Research Project of the European Community program TEMPUS № 544455-TEMPUS-1-2013-1-SE-TEMPUS-JPCR.

I would like to thank all Engensec Digital forensics team, which consists of several German Police Officers, who do not want to be named, for their inestimable contribution to this paper. I would also like to show my special gratitude to Benjamin Dorsch for sharing his pearls of wisdom during the course of the article writing.



REFERENCES

- [1] dd tool. Available: [https://en.wikipedia.org/wiki/Dd_\(Unix\)](https://en.wikipedia.org/wiki/Dd_(Unix))
- [2] tcpdump tool. Available: <http://www.tcpdump.org/>
- [3] TestDisk tool. Available: <https://www.cgsecurity.org/wiki/TestDisk>
- [4] Wireshark tool. Available: <https://www.wireshark.org/>
- [5] Autopsy tool. Available: <http://www.sleuthkit.org/autopsy/>
- [6] ewfmount tool. Available: <https://github.com/libyal/libewf/wiki/Mounting>
- [7] NetworkMiner tool. Available: <http://www.netresec.com/?page=networkminer>
- [8] Slack description. Available: <http://www.forensicwiki.org/wiki/Slack>
- [9] SANS Investigative Forensics Toolkit (SIFT) Workstation. Available: <https://digital-forensic.sans.org/community/downloads>

The Cloud Threat Landscape

*Anders Carlsson, PhD, Blekinge Institute of Technology, Sweden.
Email: anders.carlsson@bth.se*

*Alexander Adamov, Kharkiv National University of Radio Electronics, Ukraine.
Email: oleksandr.adamov@nure.ua*

Abstract

With ever growing adoption of virtualization and migration of IT infrastructure to the cloud, security concerns are raised about protecting cloud environments against new generations of cyber attacks.

The article describes the cloud threat landscape including insider, co-tenant, and cloud provider attacks as well as threats related to Network Functions Virtualization (NFV) adoption that may be a result of exploiting vulnerabilities and exposures of a cloud environment or human behavior.

Understanding the cloud threat landscape is an essential part of securing the cloud environment and the starting point for mitigation techniques and models to protect a cloud environment. The cloud attack scenarios are categorized, explained, and illustrated with the references to the real-world examples.

1. Introduction

Nowadays, we are witnessing a global trend of moving IT infrastructure into the cloud. As a result, clouds need to be protected against cyber attacks similarly to on-premises network environments. This applies to all types of cloud environments: private, public, and hybrid.

The term Man-in-the-Cloud (MitC) attack has been coined in [1] describing the possible scenarios of the MitC attack against services such as OneDrive, Dropbox, Google Drive, and Box. However, such a general service-centric classification is not enough to understand specifics of cloud-related cyber attacks and as a result to come up with and implement efficient mitigation steps.

The academic publications have also, in general, lack of practical details and real-world threats examples when describing possible classes of cyber attacks to the cloud as well as how to use the cloud resources for malicious purposes. In short, they are only loosely applicable for mitigation of real-world threats.

The authors of Threat Models, and Solutions for Virtualized Systems [2] try to show the evolution of cloud cyber attacks. However, there is a lack of references to modern types of cloud cyber attacks. Due to, theoretical threats overview based on available CVE dated as far back as 2015.

M. Berrezzouq, A. Elghazi in [3] only discuss the security issues connected with storing data in the cloud.

The work Computing Security Threats and Attacks with Their Mitigation Techniques [4] provides a shallow overview of cloud cyber threats without connecting them to real-world attacks.

Sen-Tarng Lai, Fang-Yie Leu made an attempt to measure a cloud threat in their work [5]. The authors proposed to use a Security Threats Measurement Model for Reducing Cloud Computing Security Risk where they split all cloud security factors into system, management, and techniques levels that can be used as a primitive metric, then combined into a security measurement, and transformed to a threat indicator. The efficiency of the proposed model has not been proved by the authors, no experiments results were provided.

2. Classification of cloud threats

The cloud threat landscape includes the following main classes of threats and attack strategies:

- Threats due to cloud provider vulnerabilities and exposures.
- Co-tenant threats.
- Threats to a cloud provider.
- Network Functions Virtualization (NFV) threats.

They were described using the STRIDE threat model: Spoofing, Tampering, Repudiation, Information Disclosure, Denial of service, Elevation of privilege.

2.1. Threats due to cloud provider vulnerabilities and exposures

Improper configuration of cloud infrastructure or failure in maintenance by a cloud provider can be leveraged by an attacker to compromise a tenant's environment.

- Misconfiguration of services in the cloud may result in elevation of attacker's privilege.
- A maintenance failure, for instance, not wiping disks on nodes between re-allocations, may lead to information disclosure.
- Disablement of security services when high load. For instance, turning off rules and taking protocols out of IDPS scan may lead to passing a cyber attack.
- VMs connected to the management network may lead to information disclosure and, as a result, elevation of privilege.
- Misconfiguration of administrator privileges. For example, Xenaccess [6] enables an administrator to directly access the VM memory at run time by running a user level process in zero domain.
- Disclosure of data stored by tenant in a cloud environment due to misconfiguration in data access control or legislation.
- Insecure interfaces and application programming interfaces (APIs) may lead to penetration to a cloud environment resulting in tampering, repudiation, information disclosure, and elevation of privileges.

2.2. Co-tenant threats

A tenant may attack another tenant within the same cloud infrastructure.

- The side-channel attacks that may lead to confidentiality violation.
 - The existence of Cross-VM attacks in an Amazon EC2 platform has been shown in [7].
 - Timing side-channels can be an insidious threat to cloud environment as they pervasively exist and are hard to control because of parallelism and shared infrastructure. This allow a malicious tenant to steal information from other tenants without leaving traces or raising alarms [8].
 - L2 cache covert channel can be used to obtain small secrets such as a private key [9].
 - CPU load can also be used as a covert channel to encode information [10].

- Memory disclosure attack [11, 12] is capable of detecting the existence of an application or file on a co-residing VM by measuring the write access time that differs between deduplicated pages and regular ones.
- Meltdown attack discovered by Daniel Gruss, Moritz Lipp, Stefan Mangard, Michael Schwarz [13], Jann Horn, Werner Haas [14], and Thomas Prescher [15]
- Spectre attack discovered by Jann Horn [16] and Paul Kocher et al [17].
- Escaping a VM due to a vulnerability in hypervisor may lead to taking over the host by a malicious tenant.
- Getting access to shared resources such as storage, network, and so on may lead to elevation of privileges, information disclosure, and other types of attacks.
- A tenant can be compromised and used to run a DDoS attack.
- Brute-force and dictionary attack to gain access to cloud nodes.
- Shared cloud provider's infrastructure may lead to spoofing, tampering, and elevation of privilege attacks. For instance, a shared mail service can be used to run a spear-phishing attack from one tenant to another, a shared DNS service may lead to the DNS poisoning attack, a shared Web service such as cloud admin web interfaces may be a source of XSS, CSRF, SQL injection attacks.

2.3. Threats to a cloud provider

Cloud provider may experience unsolicited tenants behavior threats cover the following categories of attacks:

- A tenant may try to hack a cloud provider.
- A tenant may try to run out of a VM or container through security breaches and connect to a management network elevating privileges.
- An attacker can sign up as a legitimate tenant using stolen credentials and performs malicious activities such as:
 - Running a botnet.
 - Cryptomining.
 - Hosting a Command-and-Control server for the expenses to be paid by a hacked tenant.
- Brute-force and dictionary attacks (EoP).
- Resource exhaustion (DoS).
- Tenant's misbehaviour that leads to IP address black-listing or loss of reputation of a cloud provider that include.

- Outgoing DDoS attacks.
- Sending Spam.
- Distributing malware, pirated, or other illegal content.
- Storing exfiltrated data.
- Running a Command & Control server.
- Outsider threats.
- Targeted attacks.
- DDoS to a cloud provider.
- Human-related threats: insider access and social engineering such as spoofing and phishing.
- Third-party access through a hijacked account or service leading to information disclosure.
- Man-in-the-Middle attack and DDoS attacks using BGP routers exposed to the Internet.
- Vulnerabilities in network devices.

2.4. Network Functions Virtualization (NFV) threats

NFV threats cover both virtualization and networking threats. To mitigate these threats, it is needed to harden every of these areas [18].

Adoption of NFV brings the following threats to cloud environment threat landscape:

- Unsolicited modification of VNF images raises an integrity threat, which can be mitigated by verifying integrity of VNF images.
- DoS amplification attacks can be caused by network traffic loops, when a VNF's output goes to directly or through intermediaries to its input.
- Overconsumption of the virtualization infrastructure resources by an attacker that may lead to DoS.
- Intellectual Property related threats. The proprietary code in VNFs can be a subject for information disclosure.
- One VM can access another VM's memory through shared IOMMU (Intel VT-d) getting direct access from a network adapter to a VM's memory that may lead to information disclosure, DoS, or tampering.
- The cloned VNF image may contain security keys, certificates, passwords, and tokens.
- An attacker can exploit debugging and monitoring interfaces enabled in a VNF for remote support.

3. Examples of cloud-related cyber attacks

To complete the picture of the cloud threat landscape, let us consider the following examples and strategies of cloud cyber attacks.

1. Operation Cloud Hopper (2009-2017) [19] — a cyber espionage campaign that targeted managed IT service providers (MSPs), allowing an attacker, APT10 group, gain access to the intellectual property and sensitive data of MSPs' clients via the PlugX, Poison Ivy, ChChes, RedLeaves, Quasar, and Graftor backdoors used at different time.
2. Using public cloud storage service.
Operation Cloud Atlas (2014) — attackers used the CloudMe cloud provider to store exfiltrated data during the cyber espionage campaign run with the help of the malware for the Windows, Linux, Android, and iOS platforms: Trojan.Win32.CloudAtlas, Trojan.Linux.Cloudatlas.a, Trojan.AndroidOS.Cloudatlas.a, Trojan.IphoneOS.Cloudatlas.a [20]. Many malware campaigns used the Dropbox accounts for malware distribution in 2014 [21].
3. Using public cloud resources for inappropriate activities such as mining cryptocurrency, hosting Command and Control servers, sending SPAM, or organizing a proxy botnet.
In January 2018, attackers compromised the Tesla's AWS cloud via the Kubernetes administration consoles exposed to the Internet and started cryptomining. To avoid fast detection, the attackers used another cloud service by CloudFlare to hide addresses of mining servers behind a content distribution network (CDN) [22].
4. Cloud services exposed to the Internet can be attacked due to misconfiguration and unpatched vulnerabilities.
In October 2017, attackers managed to compromised more than 26 000 of MongoDB and MySQL databases exposed to the Internet using disabled access control or brute forcing user/password combinations [23, 24].
5. During the last decade, huge amount of companies and organisations has outsource the corporate data to cloud operators. Only after external review, it was observed that the data could be accessed by third party.
During 2015 Swedish government «Road Administration» outsourced near every thing to IBM, that included email, staff document folder and data. This included database which handles:
 - car registration and owner information;
 - driving license and personal data.

IBM had uncontrolled support staff in India with full access to the sensitive data such as unmarked police cars, military and intelligence service cars and huge amount of citizens' data with protected identity including police and military.

Moreover, the US legislation may violate privacy of the stored data in the cloud. On March 23, 2018, the President of the United States, Donald Trump, signed the CLOUD (Clarifying Lawful Overseas Use of Data) Act [25]. The intended purpose of the CLOUD Act is to clarify how and when the United States can gain access to data stored in cloud data centers in other legal jurisdictions.

The Act consists of two parts:

- Part I: ECPA (Electronic Communications Privacy Act)
- Part II: Framework for Bilateral Agreements on Cross-Border Data Requests.

Both parts of the Cloud Act remove the barriers that might otherwise prohibit a U.S. provider from denying a government's order to access data stored in the cloud even if the data is stored in another country. Practically, this is similar to the FISA Act (Foreign Intelligence Surveillance Act [26]) that allows NSA to wiretapping everything everywhere.

4. Conclusions

The cloud vulnerabilities and enabled threats have been studied in this paper to identify the cloud threat landscape providing an essential base for developing and implementing mitigation techniques as a part of best practices for cloud security hardening.

The following types of threats were reviewed: insider, co-tenant, cloud provider, NFV, and privacy. Each of the class has numerous attack vectors described above and complemented with references to real-world attack examples.

The given examples of cloud cyber attacks fits the proposed cloud threat classification model.

The cloud threat landscape topic has been also covered by the Advanced Network Security course developed within the new EU-recognized Master's program within the Educating the Next generation experts in Cyber Security (ENGENSEC) project [27].

5. References

[1] https://www.imperva.com/docs/HII_Man_In_The_Cloud_Attacks.pdf

- [2] **D. Sgandurra, E. Lupu**, Evolution of Attacks, Threat Models, and Solutions for Virtualized Systems, February 2016 ACM Computing Surveys (CSUR): Volume 48 Issue 3, February 2016.
- [3] **M. Berrezzouq, A. Elghazi**, Issues and threats of data stored in cloud, 2017 3rd International Conference of Cloud Computing Technologies and Applications (CloudTech), 2017, pp. 1–6.
- [4] **N. Amara, H. Zhiqui, A. Ali**, Cloud Computing Security Threats and Attacks with Their Mitigation Techniques, 2017 International Conference on Cyber-Enabled Distributed Computing and Knowledge Discovery (CyberC), 2017, pp. 244–251.
- [5] **Sen-Tarng Lai, Fang-Yie Leu**, A Security Threats Measurement Model for Reducing Cloud Computing Security Risk, 2015 9th International Conference on Innovative Mobile and Internet Services in Ubiquitous Computing, 2015, pp. 414–419.
- [6] **B. D. Payne, M. Carbone, W. Lee**, Secure and Flexible Monitoring of Virtual Machines, In Proc. ACSAC'07, 2007.
- [7] **T. Ristenpart, E. Tromer, H. Shacham, S. Savage**, Hey, you, get off of my cloud: exploring information leakage in third-party compute clouds, Proc. 16th ACM conference on Computer and communications security, 2009, pp. 199–212.
- [8] **A. Aviram, S. Hu, B. Ford, R. Gummadi**, Determinating timing channels in compute clouds, In Proc. 2010 ACM workshop on Cloud computing security workshop (CCSW '10). ACM, New York, NY, USA, 103–108.
- [9] **Y. Xu, M. Bailey, F. Jahanian, K. Joshi, M. Hiltunen, R. Schlichting**, An exploration of L2 cache covert channels in virtualized environments, in Proc. 3rd ACM workshop on Cloud computing security workshop, New York, NY, USA, 2011, pp. 29–40.
- [10] **K. Okamura and Y. Oyama**, Load-based covert channels between Xen virtual machines, in Proc. 2010 ACM Symposium on Applied Computing, New York, NY, USA, 2010, pp. 173–180.
- [11] **K. Suzaki, K. Iijima, T. Yagi, C. Artho**, Memory deduplication as a threat to the guest OS, in Proc. Fourth European Workshop on System Security, New York, NY, USA, 2011, p. 1:1–1:6.
- [12] **K. Suzaki, K. Iijima, T. Yagi, C. Artho**, Software Side Channel Attack on Memory Deduplication, in 23rd ACM Symposium on Operating Systems Principles, poster, 2011.

- [13] **Daniel Gruss, Moritz Lipp, Stefan Mangard, Michael Schwarz**, Graz University of Technology, available at <https://meltdownattack.com/meltdown.pdf>
- [14] **Jann Horn**, Google Project Zero, available at <https://googleprojectzero.blogspot.com/2018/01/reading-privileged-memory-with-side.html>
- [15] **Werner Haas, Thomas Prescher**, Cyberus Technology, available at <https://blog.cyberus-technology.de/posts/2018-01-03-meltdown.html>
- [16] **Jann Horn** (Google Project Zero) available at <https://googleprojectzero.blogspot.com/2018/01/reading-privileged-memory-with-side.html>
- [17] **Paul Kocher** et al. Spectre Attacks: Exploiting Speculative Execution, available at <https://spectreattack.com/spectre.pdf>
- [18] http://www.etsi.org/deliver/etsi_gs/NFV-SEC/001_099/001/01.01.01_60/gs_NFV-SEC001v010101p.pdf
- [19] https://www.schneier.com/blog/archives/2017/04/apt10_and_cloud.html
- [20] <https://securelist.com/cloud-atlas-redoctober-apt-is-back-in-style/68083/>
- [21] <https://www.adaware.com/myadaware/security-center/whitepapers/lavasoftware-security-bulletin-february-2014-under-the-dropbox-umbrella>
- [22] <https://blog.redlock.io/cryptojacking-tesla>
- [23] <https://www.imperva.com/blog/2017/10/ransomware-attacks-on-mysql-and-mongodb/>
- [24] <https://www.bleepingcomputer.com/news/security/massive-wave-of-mongodb-ransom-attacks-makes-26-000-new-victims/>
- [25] <https://www.congress.gov/bill/115th-congress/house-bill/4943>
- [26] https://sv.wikipedia.org/wiki/Foreign_Intelligence_Surveillance_Act
- [27] <http://engensec.eu/>

Validation of Results

Експертний висновок про впровадження курсів з кібербезпеки у процес навчання

Комісія Державного підприємства «Український науково-технічний і впроваджувальний центр комплексного захисту інформації» (Український центр «Безпека»), що спеціалізується на комплексному захисті інформації, у складі заступника директора з технічних питань Кordonets K. B. дійшла до висновку, що навчальні курси:

- *Advanced Network & Cloud Security*
- *Wireless & Mobile Security*
- *Software Secure Development*
- *Malware*
- *Web Security*
- *Pentest & Ethical Hacking*

які розроблені в рамках Темпус проекту №544455-TEMPUS-1-2013-1-SE-TEMPUS-JPCR «Підготовка наступного покоління експертів з кібербезпеки: нова визнана ЄС магістерська програма» (ENGENSEC), яка фінансується Європейським Союзом, дозволяють створити сучасну методичну базу для обміну знаннями.

Перелік курсів покриває основні вимоги до спеціалістів з безпеки і відповідає сучасним викликам, з якими стикаються спеціалісти з безпеки в умовах державних і комерційних структур. Також курси побудовані таким чином, що дозволяють швидко пристосовуватися до змін у галузі безпеки.

Як висновок, комісія відмічає, що вищевказані навчальні курси матимуть позитивний вплив на освітній процес підготовки фахівців з кібербезпеки, дозволять суттєво підвищити рівень компетенцій студентів і аспірантів кафедри Інформаційної і кібернетичної безпеки Державного університету телекомунікацій.

Заступник директора
з технічних питань

Кордонет К. В.



Expert Opinion on Introducing Cyber Security Courses into the Learning Process

The commission of the State enterprise "Ukrainian scientific-technical and promotional center of complex information defense" (Ukrainian center "Safety") specializing in complex information defense, deputy director for technical issues Kordonets K.V. came to the conclusion that the training courses:

- *Advanced Network & Cloud Security*
- *Wireless & Mobile Security*
- *Software Secure Development*
- *Malware*
- *Web Security*
- *Pentest & Ethical Hacking*

developed within the framework of Tempus Project No. 544455-TEMPUS-1-2013-1-SE-TEMPUS-JPCR "Educating the Next Generation Experts in Cyber Security: the New EU-recognized Master's program" (ENGENSEC) that is financed by European Union, allow to create a modern methodological basis for the exchange of knowledge.

The list of courses covers the basic requirements of the security specialists and meets the current challenges faced by the security specialist in the state and commercial structures. Courses are also designed to quickly adapt to changes in the field of security.

As a conclusion, the commission notes that the above-mentioned training courses will have a positive impact on the educational process of training cyber security specialists, will significantly increase the level of competencies of students and post-graduate students of the Information and Cybernetic Security State University of Telecommunications.

Deputy director
for technical issues

Kordonets K.V.



Висновок про впровадження курсів програми ENGENSEC

Комісія Навчально-науково-виробничого комплексу «Інформаційно-комунікаційні системи», що спеціалізується на захисті інформації у інформаційно-комунікаційних системах, у складі голови правління Фененкова Анатолія Костянтиновича та заступника директора з наукової роботи Браїловського Миколи Миколайовича дійшла до висновку, що навчальні курси:

- *Penetration Testing and Ethical Hacking*
- *Wireless & Mobile Security*
- *Advanced Network and Cloud Security*
- *Malware*
- *Web Security*
- *Software Secure Development*

які розроблені в рамках Темпус проекту №544455-TEMPUS-1-2013-1-SE-TEMPUS-JPCR «Підготовка наступного покоління експертів з кібербезпеки: нова визнана ЄС магістерська програма» (ENGENSEC), яка фінансується Європейським Союзом, дозволили розгорнути віддалену лабораторію.

Поставлене в рамках проекту обладнання дозволяє розгорнути кіберполігон, на якому в повному обсязі можна проводити емуляцію атак на рівні операційної системи, створювати захищене середовище, аналізувати зловмисне програмне забезпечення, аналізувати структуру мережі і проводити атаки на інфраструктуру.

Через те, що система має можливість запускати один і той самий образ кілька разів, то при збої роботи не потрібно кожен раз перевстановлювати середовища, а достатньо лише перезапустити систему. Можливості систем обмежені лише можливостями середовища віртуалізації OpenStack.

Комісія відмічає, що вищевказаний кіберполігон має позитивний вплив на освітній процес підготовки фахівців з кібербезпеки, дозволили суттєво підвищити рівень компетенцій студентів на прямому навчання «Кібербезпека».

Голова правління
«ІНФОРМАЦІЙНО-КОМУНІКАЦІЙНІ СИСТЕМИ»
Заступник директора
з наукової роботи

Фененков А. К.

Браїловський М. М.

Opinion on Introducing ENGENSEC Courses

The commission of the Educational and Scientific Production Complex "Information and communication systems" specializing in the protection of information in information and communication systems, consisting of chairman of the board Feynnikov Anatoly Konstantinovich and deputy director for scientific work Brailovsky Mykolay Mykolayovych came to the conclusion that the training courses:

- *Penetration Testing and Ethical Hacking*
- *Wireless & Mobile Security*
- *Advanced Network and Cloud Security*
- *Malware*
- *Web Security*
- *Software Secure Development*

developed within the framework of Tempus Project No. 544455-TEMPUS-1-2013-1-SE-TEMPUS-JPCR "Educating the Next Generation Experts in Cyber Security: the New EU-recognized Master's program" (ENGENSEC) that is financed by European Union, allowed to deploy a remote laboratory.

The equipment provided by the project enables the deployment of a cyberpolygon, where it is possible to carry out emulation attacks at the operating system level, create a secure environment, analyze malware, analyze the structure of the network and carry out attacks on the infrastructure.

Due to the fact that the system has the ability to run the same image several times, it is not necessary to reinstall the environment every time you fail, but just restart the system. System capabilities are limited to the capabilities of the OpenStack virtualization environment.

The commission notes that the aforementioned cyberpolygon has a positive impact on the educational process of training cyber security specialists, which allowed to substantially increase the "Cybersecurity".

Chairman of the board
Deputy director
for scientific work

Feynnikov A. K.

Brailovsky M. M.



ЗАТВЕРДЖУЮ
Проректор з науково-педагогічної роботи
Державного університету телекомунікацій
Беркман Л. Н.
« » 2017 р.

ВИСНОВОК

про результати впровадження навчальних курсів в освітній процес університету

Комісія Державного університету телекомунікацій у складі директора Навчально-наукового інституту Захисту інформації Довбешка Станіслава Володимировича, завідуючого кафедрою Інформаційної і кібернетичної безпеки Вишнівського Віктора Вікторовича, начальника навчально-методичного відділу освітньої діяльності Навчально-наукового інституту Захисту інформації Навчально-наукового центру Гуменюка Валентина Валентиновича, начальника навчально-наукового центру Явтушенко Анатолія Миколайовича дійшла до висновку, що навчальні курси:

1. Технології розробки і тестування програмного забезпечення систем ІКБ (Security Software Development).
2. Методи та засоби забезпечення безпеки мережевої інфраструктури (Advanced Networks and Cloud Security).
3. Технології виявлення уразливостей та протидії зловмисному програмному забезпеченню (Malware).
4. Технології виявлення уразливостей мережевих ресурсів (Penetration Testing and Ethical Hacking).
5. Методи та засоби забезпечення безпеки безпроводових і мобільних мереж (Wireless and Mobile Security).
6. Технології виявлення уразливостей та забезпечення безпеки Web-ресурсів (Web-security),

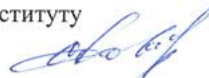
які були розроблені в рамках Темпус-проекту №544455-TEMPUS-1-2013-1-SE-TEMPUS-JPCR «Підготовка наступного покоління експертів з кібербезпеки: нова визнана ЄС магістерська програма» (ENGENSEC), яка фінансується Європейським Союзом, суттєво підвищили якість підготовки студентів університету освітнього рівня «магістр» за спеціальністю 125 «Кібербезпека».

Дані навчальні курси дають можливість впровадити програму подвійних дипломів підготовки магістрів між Державним університетом телекомунікацій (Київ, Україна) та Технологічним інститутом Блекінге (Карлскруна, Швеція).

На отриманому в рамках проекту обладнанні розгорнуто і проведено тестування кіберполігону, який дозволить проводити лабораторні та практичні заняття магістрантам як в межах лабораторії з п'ятнадцяти робочих місць, так і дистанційно (лабораторія доступна за зовнішнім посиланням <http://dut.resela.eu>).

Комісія відмічає, що впровадження в освітній процес в університеті вищевказаних навчальних курсів покращило в цілому лабораторну базу кафедри, що дозволило суттєво підвищити рівень компетенції викладачів, а також надано можливість студентам спеціальності 125 «Кібербезпека» Державного університету телекомунікацій отримувати актуальні знання на сучасному апаратному забезпеченні.

Директор Навчально-наукового інституту
Захисту інформації



Довбешко С. В.

Завідуючий кафедрою
Інформаційної і кібернетичної безпеки



Вишнівський В. В.

Начальник навчально-методичного відділу
освітньої діяльності
Навчально-наукового інституту
Захисту інформації
Навчально-наукового центру



Гуменюк В. В.

Начальник Навчально-наукового центру



Явтушенко А. М.

Validation of Results



ENGENSEC

Educating the Next generation
experts in Cyber Security



01001, м. Київ, пр. Перемоги, 56
Телефон: 458-3036
Факс: 458-3036
E-mail: info@elan-ua.net
www.elan-ua.net

№ 134-1/34

від "30" 10 2017 року

Експертний висновок
про впровадження курсів
програми ENGENSEC
у процес навчання

Комісія Товариства з обмеженого відповідальності «Елан», що спеціалізується на криптозахисті, технічному захисті інформації, розробці спеціальних технічних засобів для негласного знімання інформації тощо, у складі директора Семка В. В. і заступника директора з науки Михайловського О. О. дійшла до висновку, що навчальні курси:

- *Advanced Network & Cloud Security;*
- *Malware;*
- *Web Security;*
- *Penetration Testing & Ethical Hacking;*
- *Wireless & Mobile Security;*
- *Software Secure Development;*

які розроблені в рамках Темпус проекту №544455-TEMPUS-1-2013-1-SE-TEMPUS-JPCR «Підготовка наступного покоління експертів з кібербезпеки: нова визнана ЄС магістерська програма» (ENGENSEC), яка фінансується Європейським Союзом, суттєво підвищили якість підготовки магістрантів і їх конкурентоспроможність на ринку захисту інформації.

Кіберполігон розгорнутий на високому професійному рівні і відповідає сучасним потребам бізнесу і забезпеченню безпеки держави в цілому.

Як загальний висновок, комісія відмічає, що вищевказані навчальні курси здійснили позитивний вплив на освітній процес підготовки фахівців з кібербезпеки, дозволили суттєво підвищити рівень компетенцій студентів та випускників кафедри Інформаційної і кібернетичної безпеки Державного університету телекомунікацій.

Директор  В. Семко

Заступник
директора
з науки  О. Михайловський

Expert Opinion
on Introducing ENGENSEC Courses
into the Learning Process

Company "Elan" specializing in cryptographic protection, technical protection of information, the development of special technical means for the tacit removal of information, etc., consisting of director Semko V. V. and deputy director of science Mikhailovskiy O. O. came to the conclusion that the training courses:

- *Advanced Network & Cloud Security;*
- *Malware;*
- *Web Security;*
- *Penetration Testing & Ethical Hacking;*
- *Wireless & Mobile Security;*
- *Software Secure Development;*

developed within the framework of Tempus Project No. 544455-TEMPUS-1-2013-1-SE-TEMPUS-JPCR "Educating the Next Generation Experts in Cyber Security: the New EU-recognized Master's program" (ENGENSEC) that is financed by European Union, significantly improved the quality of preparation for graduate students and their competitive ability in the information protection market.

Cyberpolygon is deployed on a high professional level and meets the modern business needs and ensuring the security of the state as a whole.

As a general conclusion, the commission notes that the aforementioned training courses have had a positive impact on the educational process of training cyber security specialists, have allowed to significantly increasing the level of competence of students and graduates of the Information and Cybernetic Security Department of the State University of Telecommunications.

Director  V. Semko

Deputy
director
of science  O. Mikhailovskiy



001133

ВИСНОВОК

**про результати впровадження навчальних курсів в освітній процес
Львівського навчально-наукового інституту Державного вищого
навчального закладу "Університет банківської справи"**

Інтегрований підхід до підготовки магістрів спеціалізації «Економічна кібернетика» у Львівському інституті ДВНЗ "Університет банківської справи" потребує ґрунтовного вивчення дисциплін з кібербезпеки. Це дає випускникам змогу успішно працювати: у підрозділах Національного банку України та банківських установ, в інвестиційних компаніях та акціонерних товариствах, в органах державного управління будь-якого рівня, у фірмах, що займаються розробкою, впровадженням та супроводженням інформаційних систем, а також на підприємствах, що застосовують комп'ютерні технології.

Вважаю, що навчальні курси:

- Розробка програмного забезпечення в сфері інформаційної безпеки;
- Інформаційна безпека телекомунікаційних та хмарних технологій;
- Менеджмент обробки інцидентів інформаційної безпеки;
- Методи виявлення та аналізу шкідливого програмного забезпечення;
- Системи аналізу вразливостей та етичний хакінг;
- Проектування, експлуатація та захист бездротових мереж;
- Адміністрування, аудит та безпека інформаційних служб Internet,

які розроблені в рамках Темпус Проекту No. 544455-TEMPUS-1-2013-1-SE-TEMPUS-JPCR "Підготовка наступного покоління експертів з кібербезпеки: нова визнана ЄС магістерська програма" (ENGENSEC), яка фінансується Європейським Союзом, дозволяє суттєво підвищити якість підготовки студентів освітнього рівня «магістр» спеціалізацією «Економічна кібернетика» у Львівському інституті ДВНЗ «Університет банківської справи».

Впровадження в освітній процес даних навчальних курсів позитивно впливає на освітній процес підготовки фахівців з економічної кібернетики, а також суттєво підвищує рівень компетенцій магістрів спеціалізації «Економічна кібернетика» Львівського навчально-наукового інституту Державного вищого навчального закладу "Університет банківської справи".

Гарант освітньої магістерської програми «Інформаційний менеджмент та бізнес-аналітика у банках і фінансових установах», професор,
д.т.н., професор кафедри Економічної кібернетики
4vyach@gmail.com

Чаплиця В.М.

Підпис *Чаплиця В.М.* засвідчує
Провідний фахівець
відділу кадрів *В.М.*



Validation of Results



ENGENSEC

Educating the Next generation
experts in Cyber Security

Certified translation from Ukrainian into English

CONCLUSION

on the results of the implementation of some courses in the educational process at the Lviv Educational and Scientific Institute of the Banking University, State Higher Educational Institution.

An integrated approach to the economic cybernetics master's degree programme at the Lviv Educational and Scientific Institute of the Banking University, State Higher Educational Institution requires a detailed study of cybersecurity disciplines. Therefore, the graduates have an opportunity to work successfully: at offices of the National Bank of Ukraine and other banking institutions, in investment companies and joint-stock companies, in state authorities of any level, with companies involved in the development, implementation and maintenance of information systems, as well as in enterprises, using computer technology.

I believe that the following courses:

- Security Software Development
- Advanced Networks and Cloud Security
- Digital Forensic
- Malware
- Penetration testing and ethical hacking
- Wireless & Mobile Security
- Web-security)

developed within Tempus Project No 544455-TEMPUS-1-2013-1-SET-TEMPUS-JPCR entitled "Training of the Next Generation Cyber-Security Experts a New EU Recognized Master's Degree Program" (ENGENSEC), funded by the European Union, can significantly improve the quality of education for master's degree students majoring in economic cybernetics at the Lviv Educational and Scientific Institute of the Banking University.

The implementation of these courses will enrich the economic cybernetics programme educational process. It will also significantly increase the competence of economic cybernetics master's programme graduates at the Lviv Educational and Scientific Institute of the Banking University.

Guarantor of the Information Management and Business Analytics at Banks and Financial Institutions Master's Degree Program
Professor, D.Eng., Professor at the Department of Economic Cybernetics
4vyach@gmail.com

/signature/ Chaplyha V.M.

Seal: "Ministry of Education and Science of Ukraine * Banking University * State Higher Educational Institution * Lviv Educational and Scientific Institute * Human Resources No. 1 EDRPOU code 34716922"

Stamp: "I hereby certify the signature of Chaplyha V.M.
Lead specialist of the HR /signature/

Certification by Translator

I, TARAS MELNYK, certify that I am fluent (conversant) in the English and Ukrainian languages and that the above document is, to the best of my knowledge and ability, an accurate translation of the document attached.

Signature _____ Date November 6, 2014

Typed name TARAS MELNYK

Telephone: +38(067)8673107 +38(066)6235780 Email: express.pereklad@gmail.com





November 9, 2017

Dear Anders Carlson,

We at UnderDefense LLC highly appreciate security events and workshops held by Engensec. During the last several years our professional security experts have been sharing their experience, knowledge. Holding additional lectures and trainings in Lviv Universities for students' competence, capabilities and skills improvement, and we see a great and positive impact of such events on the education and growth of the future analysts and engineers of Information Security.

First of all, events in different universities led by Engensec give students a chance to meet Information Security leading analysts and allow them to feel themselves as a part of the enormous IT world. Second of all - organizations of these kind of events are a great opportunity to meet students from different countries, share experience of the studying processes and get familiar with other cultures and traditions of the world, in particular - country-organizer.

Another amazing side of those events is that during 2 weeks students gain practical and much needed skills of ethical hacking, reverse engineering, network hacking and practically cover other summer school topics and just study new material, the universities usually don't cover.

It is a significant experience for team building inside foreign groups.

Two weeks workshops can give a good start of new fields of Information Security as an industry.

We are actively involved in the education of young professionals and highly appreciate the work of Information Security community in an evolving field.

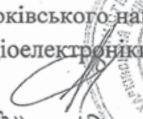
Best regards,
Nazar Tymoshyk





ЗАТВЕРДЖУЮ

Проректор з науково-методичної роботи
Харківського національного університету
радіоелектроніки


І.В. Рубан
« 26 » 2017 року

ВИСНОВОК

про результати впровадження навчальних курсів в освітній процес
університету

Комісія Харківського національного університету радіоелектроніки у складі декана факультету інфокомунікацій Снігурова А.В., начальника навчального відділу Міхнової А.В., начальника навчально-методичного відділу Милотченко І.О. дійшла до висновку, що навчальні курси:

Розробка програмного забезпечення в сфері інформаційної безпеки (Security Software Development)

Інформаційна безпека телекомунікаційних та хмарних технологій (Advanced Networks and Cloud Security)

Менеджмент обробки інцидентів інформаційної безпеки (Digital Forensic)

Методи виявлення та аналізу шкідливого програмного забезпечення (Malware)

Системи аналізу вразливостей та етичний хакінг (Penetration testing and ethical hacking)

Проектування, експлуатація та захист бездротових мереж (Wireless & Mobile Security)

Адміністрування, аудит та безпека інформаційних служб Internet (Web-security),

які розроблені в рамках Темпус Проекту No. 544455-TEMPUS-1-2013-1-SE-TEMPUS-JPCR "Підготовка наступного покоління експертів з кібербезпеки: нова визнана ЄС магістерська програма" (ENGENSEC), яка фінансується Європейським Союзом, суттєво підвищили якість підготовки студентів університету освітнього рівня «магістр» за спеціальністю «Кібербезпека». Впровадження в освітній процес даних навчальних курсів дозволило розробити програму подвійних дипломів підготовки магістрів між Харківським національним університетом радіоелектроніки та Блекінге

технологічним інститутом (Карлскруна, Швеція). Обладнання, яке отримане університетом в рамках даного проекту на базі якого розгортається хмарна лабораторія, дозволить суттєво підвищити практичну спрямованість підготовки фахівців з кібербезпеки.

Як загальний висновок, комісія відмічає, що впровадження в освітній процес в університеті вищевказаних навчальних курсів здійснило позитивний вплив на освітній процес підготовки фахівців з кібербезпеки, дозволило суттєво підвищити рівень компетенцій студентів та випускників спеціальності «Кібербезпека» Харківського національного університету радіоелектроніки.

Декан факультету інфокомунікацій



А.В.Снігуров

Начальник навчального відділу



А.В.Міхнова

Начальник навчально-
методичного відділу



І.О. Милютченко

Validation of Results



ENGENSEC

Educating the Next generation
experts in Cyber Security



CONCLUSION

about possibility of using courses developed at the project 544455-TEMPUS-1-2013-1-SE-TEMPUS-JPCR: “Educating the Next generation experts in Cyber Security: the new EU-recognized Master’s program” (ENGENSEC) for training highly qualified cyber security practical experts

Western Regional Educational Scientific Informational Security Centre (WRESISC) has been performing a wide range of supporting activities and works in the information security sphere as well as taking part in preparation and training cyber security specialists in Ukraine. After analyzing by WRESISC’s leading specialists during dissemination of the ENGENSEC course modules: Advanced Network & Cloud Security; Wireless & Mobile Security; Secure Software Development; Malware; Web Security; Penetration Testing and Ethical Hacking; Digital Forensic; Remote Security Laboratory (ReSeLa) our Centre began implementation most of them for training security specialists as well as cyber security master students during their practical training while writing Master Theses.

Together with Information Security department and Information Technologies Security department at Lviv Polytechnic National University we are implementing the developed ENGENSEC laboratory works using the supplied by EU equipment during this project with the special Open Stack developed module ReSeLa. The students during practical training are highly motivated while performing scientific and research activities especially those six who spent one semester at Blekinge Institute of Technology in Sweden and two of them even defended theses there.

The ENGENSEC project has really created a new Master’s program in the area of IT Security as a response on current cybersecurity threats. In the nearest future we are planning to continue developing and improving Open Stack Remote Security Laboratory together with our Ukrainian colleagues from Kharkiv National University of Radioelectronics and State University of Telecommunications (Kyiv); Swedish Blekinge Institute of Technology as well as Polish Wroclaw Institute of Technology.

Director of WRESISC
of Lviv Polytechnic National University



V. Dudykevych

Cyber Risk Research Institute

Taylor Rodriguez

Andrew Vance

Executive Directors

409 East Caramillo St
Colorado Springs, Co 80907
+1 (800) 282 - 1659
andrew.vance@cyber-institute.org
taylor.rodriguez@cyber-institute.org

Anders Carlson

Blekinge Tekniska Högskola
37179 Karlskrona
0455 – 38 50 00
anders.carlson@bth.se

November 2, 2017

Dear Anders Carlson,

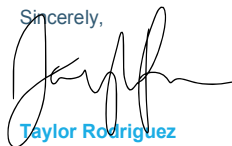
This letter serves as our confirmation of participation as an ENGENSEC partner that participated in the ENGENSEC IT Security 2017 summer school held at Lviv Polytechnic University.

Cyber Risk Research Institute (CRRI) served as an external evaluator and analyzed the courses developed by the ENGENSEC team. CRRI is currently developing a strategy to promote and endorse that these courses be implemented as part of a NATO Multinational Cyber Defence Education and Training (MNCDE&T) curriculum which CRRI serves as a United States NGO Delegate. Our organization also intends to endorse and demonstrate the ENGENSEC curriculum to our United States academic and industry partners.

We were impressed by the course curriculum that the ENGENSEC team had developed and we believe that it has enormous potential to educate the next generation of security professionals. We wish to continue to participate as an ENGENSEC partner and recommend current and future curriculum to our NATO, academic, and industry colleagues.

Cyber Risk Research Institute is a 501c 3 nonprofit as recognized by United States Internal Revenue Service (IRS). Reference the IRS tax identification number 81-3818108.

Sincerely,



Taylor Rodriguez



Andrew Vance

<https://www.crri.us>



Validation of Results



ENGENSEC

Educating the Next generation
experts in Cyber Security



JOINT STOCK COMPANY
«ECHELON UNION FOR SCIENCE AND DEVELOPMENT»

107023, Moscow, Electrozavodskaya str., 24, tel./fax: +7 (495) 223 23 92, 645 38 09, mail@npa-echelon.ru, www.npa-echelon.com

Blekinge Tekniska Högskola
37179 Karlskrona
0455 - 38 50 00
anders.carlson@bth.se

January 22, 2018

Dear Anders Carlson!

Training center "Echelon" participated in development of penetration testing and ethical hacking course. As a part of IT-security consulting firm we specialize in IT-security training for post-graduate students and we tried to put most practical things in the course.

We were impressed by the content of other courses developed by our colleagues in ENGENSEC team. We would like to recommend the ENGENSEC courses for including in curriculum of higher education institutions for developing real professionals in our field.

Our team was glad to participate in ENGENSEC project and we would appreciate to be informed about similar education projects.

Sincerely,

Alexander Dorofeev



DIN EN ISO 9001
GOST RV 0015-002
ISO/IEC 27001

information security services and solutions

St. Petersburg Institute
for Informatics and Automation
of the Russian Academy of Sciences
(SPIIRAS)
199178, St. Petersburg, 14th line, 39
Phone: (812)328-33-11
Fax: (812)328-44-50
E-mail: spiiran@iiias.spb.su
<http://www.spiiras.nw.ru>
«23» January 2018
№ 073-10-01/36

SUMMARY

of the ENGENSEC project results' introduction into SPIIRAS academic activity

The SPIIRAS scholars have participated in the project № 544455-TEMPUS-1-2013-1-SE-TEMPUS-JPCR, particularly in the development of the following training courses: "Advanced Network and Cloud Security" and "Digital Forensics" within this project within this project.

In a result the above training courses have been developed to be used at the masters' training in computer security, information and telecommunications systems and network technologies. At that the SPIIRAS team members have acquired a considerable experience while preparing the courses "Advanced Network and Cloud Security" and "Digital Forensics. Undoubtedly, a highly professional level of all participants engaged in this project is worth noting as well as an efficient management ensured by the project coordinator Professor Anders Carlsson.

Acting Director



Andrey Ronzhin

Chief researcher, Head of laboratory

Igor Kotenko



►► Contractor

Blekinge Institute of Technology (Karlskrona, Sweden)

►► Consortium

The ENGENSEC consortium consists of 22 partners from 7 countries. The project has been funded with support from the European Commission

►► Objective

The main objective of the ENGENSEC is to create new Master's program in area of IT Security as a response on Cyber Security threats by educating the next generation experts.

►► Proposal

The ENGENSEC proposes the way to address the educational challenges in area of cyber security and privacy in the e-oriented world.

►► We are on the Internet



ISBN: 978-91-7295-963-7



<http://engensec.eu/>

<https://www.facebook.com/engensec>