SYSTEMS & LANGUAGES FOR INFORMATICS (12 CFU)

Instructors: Prof. Giuseppe Anastasi, Prof. Fabio Gadducci, Prof. Federica Paganelli

OBJECTIVES. This course is intended for students graduated in disciplines different from computer science/engineering. It aims at improving the preparation in areas such as computer programming, languages, and algorithms (module on *Languages for Informatics*) as well as computer systems and networks (module on *Systems for Informatics*). In addition to basic concepts, the student will learn how to develop a distributed application based on sockets.

MODULE 1: LANGUAGES FOR INFORMATICS - TENTATIVE PROGRAM

Instructors: Fabio Gadducci, Federica Paganelli

PROGRAMMING AND PROBLEM SOLVING:

- Introducing the principles of programming languages (programming constructs, static and run-time mechanisms) using the OCAML programming language (or Fsharp) [10 hours]
- Introducing the concepts of problem solving in order to provide a grounding design methodology of algorithms over basic data structures (sequences, sets, trees and maps) [10 hours]
- Introduction to program complexity [4 hours]

C LANGUAGE:

- Constructs, functions, data structures and pointers in C [12 hours]
- Posix library and PThread [12 hours]

MODULE 2: SYSTEMS FOR INFORMATICS - TENTATIVE PROGRAM

Instructors: Giuseppe Anastasi

COMPUTER SYSTEMS. Computer Organization. Interrupt Mechanism. Assembly Language. Operating System Organization. Processes and Process Management. Memory Management, Virtual Memory. I/O Management. File System. Databases. DBMS. SQL Language. [24 hours]
COMPUTER NETWORKS. Preliminary Concepts. Point-to Point links. PPP Protocol. Local Area Networks (LANs). Ethernet. Switched Networks. Virtual LANs. Internet. IP Protocol, IP Addresses, Datagram Forwarding. Transport Protocols (UDP, TCP). Wireless Networks and Mobile Computing. [18 hours].
DISTRIBUTED APPLICATIONS. Socket-based Interface. Client/Server and P2P Applications. Popular Internet Applications (Web, E-mail, P2P Content Sharing, P2P) [6 hours]

REFERENCES

- Y. Minsky, A. Madhavapeddy, J. Hickey, Real World OCAML: functional programming for the masses, O'Reilly, 2015
- M. Gabrielli, S. Martini, Programming Languages: principles and paradigms, Springer Verlag
- Kernighan, Brian W.; Ritchie, Dennis M. (February 1978). The C Programming Language (1st ed.). Englewood Cliffs, NJ: Prentice Hall. ISBN 0-13-110163-3.
- Marc J. Rochkind. Advanced UNIX Programming 2nd Edition, Addison-Wesley Professional Computing Series, 2004.
- A. Silberschatz, P. Galvin, G. Gagne, *Operating Systems Concepts*, IX Edition.
- J. Kurose, K. Ross, *Computer Networking*. *A Top-Down Approach* - VII Edition. Pearson Education

ASSESSMENT

- oral examination + project discussion (design and implementation of a client/server or P2P application based on socket, using the C programming language)

ELECTRONIC SYSTEMS (6 CFU)
Module of ELECTRONICS AND COMMUNICATION TECHNOLOGIES
Instructors: Prof. Roberto, Saletti; Prof. Federico, Baronti

OBJECTIVES. The main objective of the Electronic Systems module is to provide perspective students from different bachelor studies with a common Electronics background, and to let them acquire a common and shared vocabulary on the Electronic Systems domain. The students will acquire competences on and knowledge about the main electronic platforms used in cybersecurity applications.

TENTATIVE PROGRAM
PRELIMINARY CONCEPTS. Course overview. Architecture of digital signal processing systems. Information conversion between the analog and digital domains. AD and DA converters description and characteristics. Circuit architectures and performance of the most common AD and DA converters. [8 hours]

DIGITAL ELECTRONIC CIRCUITS. Recalls about logic networks (combinational and sequential networks). Digital Electronics. Structure, performance and characteristics of digital circuits. Brief overview of memory technologies. Complex digital networks: basic computer architecture. Overview of hardware description languages (HDL) for digital system design. [16 hours]

ELECTRONIC PLATFORMS. HDL-based design flow. Platforms for implementation of digital functions. Software programmable (MCU) and hardware programmable platforms (FPGA). Application specific integrated circuits. Definition and performance comparison of the most common programmable architectures such as MCU, DSP, GPU, FPGA and ASIC.
Characteristics and performance of an MCU platform (MicroController Unit). Computer peripherals aimed at cybersecurity applications. [18 hours]

HARDWARE/SOFTWARE DESIGN METHODOLOGIES. Examples of a computer design based on system integration tools and HDL design methodologies on programmable logic. [6 hours]

REFERENCES
- J.Wakerly – Digital Design: Principles and Practices (Pearson)
- Papers provided by instructors

ASSESSMENT
Oral examination

COMMUNICATION TECHNOLOGIES (6 CFU)
Module of ELECTRONICS AND COMMUNICATION TECHNOLOGIES
Instructor: Prof. Marco Luise

OBJECTIVES. The course of Communication Technologies has the purpose of integrating the competence about digital communication systems and technologies for those students who have never been exposed to such topics before. The learning objective is to provide general knowledge of the architectural features and basic technologies of the main communication systems for the transport and access network (wired and wireless), also presenting specific examples. The course will provide students with i) a general knowledge of the basic technologies that allow the design of wired (copper, fiber) and wireless communication systems, ii) a specific knowledge of the main communication standards for transport and access networks, and iii) in-depth knowledge of robust spread spectrum wireless communication techniques..

TENTATIVE PROGRAM

1. Basics of Signals, Spectra, Digital Communications, and Information Theory (4 hours)
2. Digital Data Signals for wired and wireless media – Wireless Communications (6 hours)
3. Generation of cellular networks (2G to 5G), multiplexing and multiple access, including CDMA and OFDM(A) (12 hours)
4. Fiber-Optic communications for Internet backbones (12 hours)
5. Technologies for the access network: the families of xDSL and FTTx (8 hours)
6. Physical-secure communications: Spread spectrum signaling and anti-jamming/spoofing (6 hours)

REFERENCES

- J. Proakis, M Salehi, "Digital Communications (5th Ed.)", McGraw-Hill
- Learning material provided by the instructor

ASSESSMENT
Oral examination

ORGANIZATIONAL SCIENCES AND INFORMATION AND TECHNOLOGY LAW (12 CFU)

Instructors: Prof. Dianora Poletti, Prof. Federico Niccolini, Dr. Federica Casarosa

OBJECTIVES

The course aims to improve students' knowledge about principles, theories and methodologies of organizational science and technology law. At the end of the Course students will acquire skills to interpret structural, strategic, cultural, human related and knowledge dynamics for the correct analysis of a cyber-resilient information systems in the organizational context. Students will also acquire knowledge of the legal rules applicable to computer technologies and their implementation in cybersecurity systems.

MODULE 1: ORGANIZATIONAL SCIENCES

Instructor: Federico Niccolini

FUNDAMENT OF ORGANIZATIONAL SCIENCE:

- Structural dimensions, structures for cybersecurity, Interorganizational networks, strategic process
- Organizational culture and related typologies, Knowledge management and organizational learning

APPLICATIONS OF ORGANIZATIONAL SCIENCE:

- Organizational cybersecurity risk management, Organizational resilience

MODULE 2: INFORMATION AND TECHNOLOGY LAW

Instructors: Dianora Poletti

FUNDAMENT OF CYBERSECURITY LAW:

- Scope of cybersecurity regulation, the legal framework applicable, analysis of cybersecurity regulation and it impact over the regulation at national level. Data Protection and cybersecurity.

CERTIFICATION AUTHORITIES AND LIABILITY REGIME:

- The role and specificities of the European Cybersecurity Certification, Standardization, Liability in case of data breach

CASE STUDY :

- Cybersecurity and heath care system.

REFERENCES

MODULE 1:

- Daft, R. L. (2020). *Organization theory and design*. Cengage learning (13th edition) (Chaps. 1-3, 9, 11).
- Nonaka I. (2007, July-August), *The knowledge creating company*, Harvard Business Review, pp. 162-166
- Nonaka, I., & Toyama, R. (2003). *The knowledge-creating theory revisited: knowledge creation as a synthesizing process*. Knowledge Management Research & Practice, *1*(1), 2-10
- Carlton, M., Levy, Y., & Ramim, M. M. (2019). Mitigating cyber-attacks through the measurement of non-IT professionals' cybersecurity skills. *Information and Computer Security, 27*(1), 101-121.
- National Institute of Standards and Technology (2018a). *Cybersecurity framework.*
- Paananena, H., Lapkeb, M., & Siponena, M. (2020). State of the art in information security policy development. *Computer & Security*, 101608. https://doi.org/10.1016/j.cose.2019.101608
- Radichel, T. (2014). Case study: Critical controls that could have prevented target breach. *SANS Institute Information Security Reading Room*, 1-30.

MODULE 2:

- Papers provided by instructors

ASSESSMENT

MODULE 1: the exam consists of a final written test followed by an oral text. Optionally, students can also participate in preliminary group projects. Discussion of group projects will be part of the exam for the students that voluntary participated

MODULE 2: the exam consists of a final written test followed by an oral text.

DATA AND SYSTEM SECURITY (9 CFU)

Instructor: Prof. Stefano Chessa

OBJECTIVES

The course provides an up-to-date view of the latest developments of cybersecurity in data and system management, with the main reference to operating systems, distributed systems, and mobile systems. The covered topics are the definition of threats to computer systems and the discussion of the countermeasures that can be taken. For each covered topic, the course presents its foundations, the design aspects of secure systems and provides examples from the real world of standards and applications.

PROGRAM

- ELEMENTS OF COMPUTER SECURITY: threats, attacks, security requirements and defense strategies, elements of cryptography. [6 hours]
- USER AUTHENTICATION AND ACCESS CONTROL: Discretionary Access Control (DAC), Role Based Access Control (RBAC), Mandatory Access Control (MAC), Attribute Based Access Control (ABAC) [10 hours]
- SECURITY IN DATABASES AND DATACENTERS: SQL injection, inferential and out of band attacks, countermeasures, DBMS access control, database encryption [4 hours]
- ATTACKS: malware, denial of service [8 hours]
- INTRUSION PREVENTION AND DETECTION [2 hours]
- SOFTWARE SECURITY: buffer and heap overflow, handling program input, CGI/PHP code injection, vulnerable shell scripts [6 hours]
- OPERATING SYSTEM SECURITY: virtualization, Linux and Windows security, trusted computing and multilevel security [22 hours]
- SECURITY IN IOT AND CYBER-PHYSICAL SYSTEMS: aspects of security in IEEE 802.15.4, ZigBee, Bluetooth, MQTT and other standards [10 hours]
- BLOCKCHAINS [4 hours]

REFERENCES

- Computer Security – Principles and Practice (Pearson, fourth edition), W. Stallings, L. Brown
- Scientific papers provided by the instructor

ASSESSMENT

- Written test + oral examination

APPLIED CRYPTOGRAPHY (9 CFU)

Instructor: Prof. Gianluca Dini

OBJECTIVES. The Applied Cryptography course provides an updated overview of the most recent developments in applied cryptography and its applications in the field of computer engineering for the design and implementation of products, protocols, services and secure systems. For each covered topic, the course presents the fundamental aspects in terms of security and performance properties. The course will make extensive use of examples taken from real world, standards and applications. The course is organized in two parts: teaching, for about 60 hours, and hands-on lab sessions, for about 12 hours. During the course many case studies will be illustrated and discussed (e.g., LFSR, WEP, IpSec, Secure Socket Layer, Kerberos, and so on) as well as analytical and side-channel attacks.

TENTATIVE PROGRAM

Symmetric Ciphers: one-time pad, stream-ciphers, and block-ciphers. The DES and AES ciphers. Encryption modes. (10 hours)

Hash functions: message digest codes and message authentication codes. Black box attacks: the birthday attack. How to build message digest codes and message authentication codes. Authenticated encryption. (10 hours)

Public key cryptography. Diffie-Hellman key establishment. The RSA and the ElGamal cryptosystems. The Elliptic Curves Cryptography. The (Generalized) Discrete Logarithm Problem. Factorization. (10 hours)

Digital signatures, certificates, certification authorities, and public key infrastructures. The X.509v3 certificate format. (10 hours)

Key establishment. Design and analysis of authentication and key establishment protocols. Perfect forward security. (10 hours)

Secure Random Number Generators. True Random Number Generators. Pseudo-random Number Generators. Cryptographically Secure Pseudo-Random Generators. Ah-hoc generators. (10 hours)

hands-on lab sessions (12 hours)

ASSESSMENT

- Written test + oral examination

LANGUAGE-BASED TECHNIQUES FOR SECURITY (9 CFU)

Instructor: Prof. Gian Luigi Ferrari, Prof. Chiara Bodei

OBJECTIVES: Traditionally, computer security has been largely enforced at the level of operating systems. However, operating-system security policies are low-level (such as access control policies, protecting particular files), while many attacks are high-level, or application-level (such as email worms that pass by access controls pretending to be executed on behalf of a mailer application). The key to defending against application-level attacks is application-level security. Because applications are typically specified and implemented in programming languages, this area is generally known as language-based security. A direct benefit of language-based security is the ability to naturally express security policies and enforcement mechanisms using the developed techniques of programming languages.

The aim of the course is to allow each student to develop a solid understanding of application level security, along with a more general familiarity with the range of research in the field. In-course discussion will highlight opportunities for cutting-edge research in each area. The course intends to provide a variety of powerful tools for addressing software security issues:

- To obtain a deeper understanding of programming language-based concepts for computer security.
- To understand the design and implementation of security mechanisms.
- To understand and move inside the research in the area of programming languages and security.

This course combines practical and cutting-edge research material. For the practical part, the dual perspective of attack vs. protection is threaded through the lectures, laboratory assignments, and projects. For the cutting-edge research part, the course's particular emphasis is on the use of formal models of program behaviour for specifying and enforcing security properties.

Learning Goals - After the course, students should be able to apply practical knowledge of security for modern programming languages. This includes the ability to identify application- and language-level security threats, design and argue for application- and language-level security policies, and design and argue for the security, clarity, usability, and efficiency of solutions, as well as implement such solutions in expressive programming languages. Student should be able to demonstrate the critical knowledge of principles behind such application-level attacks as race conditions, buffer overruns, and code injections. You should be able to master the principles behind such language-based protection mechanisms as static security analysis, program transformation, and reference monitoring.

TENTATIVE PROGRAM

- Memory corruption flaws: buffer overflow, stack overflow, code injection [4]
- Language-based Security: Java stack inspection and access control, certified compilation, typed assembler languace, code obfuscation [12]
- Language-based Security Lab [8]
- Information flow security: integrity, confidentiality, hidden channels. Type-based information flow [8]
- Information flow security Lab: JSflow [8 hours]
- Security in Web Application [8 hours]
- Formal methods for Security: type systems and static analysis [8]
- Formal Method for security Lab [8]

ASSESSMENT

There are lab assignments. The lab assignments are experimental activities about specific problems. To pass the course, students must pass the labs by making a presentation of the assignments in class and pass the requirements on a written report that documents the activities done.

HW AND EMBEDDED SECURITY (9 CFU)

Instructors: Prof. Sergio Saponara, Prof. Daniele Rossi

OBJECTIVES.

The course Hardware and Embedded Security (9 CFU, 72 hours of frontal lessons) aims at providing the required skills to analyze, design and verify dedicated HW solutions or HW/SW embedded systems (e.g. Hardware security modules integrated in general purpose processors) for several cryptographic functions for encryption/decryption, signature and anomaly/intrusion detection. The course will also present application examples of HW security and embedded security to IoT, Automotive or Industry4.0 case studies.

TENTATIVE PROGRAM

More in details the program of the course will cover the following subjects with two parts: the first part more related to Embedded Security at digital Hardware and low-level SW levels, while the second more focused on Hardware security considering technological aspects

Embedded Security at digital Hardware and low-level SW levels [40 hours]:

- Introduction to the course, teachers, type of exam, basics required to follow the course [2 hours]
- HW/SW co-design for cybersecurity and comparison of SW-based solutions vs HW-based ones in terms of energy efficiency, real-time operating capability, flexibility, cost and size [5 hours].
- Analysis of cryptographic accelerators embedded in General Purpose processors (e.g. HSM-Hardware Security Modules in Intel and/or ARM and/or Aurix platforms) [6 hours]
- Examples of HW accelerators for cybersecurity for asymmetric and symmetric cryptography and for signature (e.g. coprocessors for AES, SHA, ECC) and evolution towards post-quantum cryptography [16 hours]
- Embedded solutions for anomaly/intrusion detection [8 hours]
- Examples of application of embedded security (digital Hardware and low-level SW levels) to IoT and Automotive case studies [3 hours]

Hardware security considering technological aspects [32 hours]:

- Correlations among security and safety issues [5 hours].
- Technologies and architectures for secure storage of data and keys and Smart cards [8 hours]
- Technology trends for on-chip generation of random data, Physically Unclonable Functions (PUF), HW Random Number Generation (e.g. TRNG/CSPNRG) [8 hours]
- Physical levels “side-channel” cybersecurity attacks (by analyzing thermal, power and electrical signals) [8 hours]
- Examples of application of HW security considering technological aspects to IoT, Automotive or Industry4.0 case studies [3 hours]

REFERENCES

J. Szefer, “Principles of Secure Processor Architecture Design”, Morgan & Claypool publisher, 2018
Teaching material (slides, notes,..) provided by the teachers

ASSESSMENT

Oral exam (with at least one question on each of the 2 main parts) plus the discussion of the report of a practical (hand-on) project assigned to group of students (e.g. 2 students for each group) by the teachers during the course

NETWORK SECURITY (9 CFU)

Instructor: Prof. Rosario Giuseppe Garroppo, Prof. Michele Pagano

OBJECTIVES

The explosive growth of network connections has increased the dependence of both individuals and organization on the information stored and communicated using computer systems and their interconnections. The purpose of the course is to provide a practical survey of network security applications and standards in wired and wireless networks. The emphasis is on applications that are widely used in Internet and for corporate networks, and on standards, especially in Internet, WLAN and mobile networks standards.

TENTATIVE PROGRAM

Introduction to the course: Fundamental Security Design principles, Network Security Model [2 hours]

Network Access Control and Cloud Security, Key Distribution and User Authentication, Extensible Authentication Protocol: Authentication Methods, EAP Exchanges, IEEE 802.1X Port-Based Network Access Control [10 hours]

IP and Transport-Level Security: IPsec Services and protocol, Transport and Tunnel Modes, IP Security Policy, Encapsulating Security Payload (ESP), Internet Key Exchange (IKE), TLS Architecture, Heartbeat Protocol, SSL/TLS ATTACKS [9 hours]

Application security: Web security, HTTPS, Secure Shell, Electronic Mail Security, S/MIME, Domain Name System (DNS) and security extensions (DNSSEC) [12 hours]

Intrusion Detection and Firewalls: Rule-Based Intrusion Detection, Distributed Intrusion Detection, Firewall Characteristics and Access Policy, Firewall Location and Configurations: DMZ Networks, Virtual Private Networks, Distributed Firewalls, [6 hours]

Wireless Network Security: Threats and Measures [6 hours]

IEEE 802.11 Wireless LAN: IEEE 802.11i Wireless LAN Security, IEEE 802.11i Services, WPA2 and WPA3 certification program. [9 hours]

Security in Mobile Networks: GSM/GPRS, UMTS and LTE threats, security services and procedures, Overview of 5G security architecture in 3GPP, 5G Threats [12 hours]

Security and privacy in the IoT: Traditional versus Lightweight security, Secure data aggregation, Security in the wireless technologies for the IoT [6 hours]

REFERENCES

- W. Stallings, Network security essentials: applications and standards, 6th ed., Ed. Pearson, 2017
- Penttinen, Jyrki T. J., 5G explained: security and deployment of advanced mobile communications, ed. Wiley, 2019
- William A. Arbaugh, Jon Edney, Real 802.11 Security: Wi-Fi Protected Access and 802.11i, Ed. Addison-Wesley Professional, 2003
- Cirani, Simone, Ferrari, Gianluigi, Picone, Marco, Veltri, Luca, Internet of things: architectures, protocols and standards, Ed. Wiley, 2019

ASSESSMENT

- Oral examination

SECURE SOFTWARE ENGINEERING (9 CFU)

Instructor: Prof. Antonio Brogi

OBJECTIVES.

The aim of the course is to introduce security-aware, advanced software engineering techniques. The course includes a 3 ECTS hands-on lab for active learning, and continuous assessment activities during the term.

TENTATIVE PROGRAM

- Agile software development (Agile principles, user stories) [6 hours]
- Microservices (motivations, definition, properties, case studies) [8 hours]
- Security in application design (confidentiality, integrity, availability) [6 hours]
- Static analysis of software security (vulnerability analyses) [6 hours]
- Secure software deployment (cloud- and container-based) [6 hours]
- Dynamic analysis of software security (development/release/user testing, monitoring) [8 hours]
- Security in Edge and Fog computing [8 hours]
- Hands-on lab [24 hours]

REFERENCES

The teaching material will include scientific articles, slides and teaching notes.
All the teaching material for the course will be made available by the Instructor on the net.

EXAM

Continuous assessment and oral exam

DEPENDABILITY (6 CFU)

Instructor: Cinzia Bernardeschi

OBJECTIVES. The Dependability course will give the theoretical foundations of computer-based systems dependability and a view on methodologies for the design of dependable systems in industrial contexts.

TENTATIVE PROGRAM

PRELIMINARY CONCEPTS. Course overview, introduction to dependability, examples of safety-critical systems failures. Dependability attributes: Reliability, Availability, Safety, Maintainability. Threats to dependability: faults, errors, failures. Taxonomy of faults. Malicious faults. Fault activation and error propagation. Means to attain dependability: fault prevention, fault tolerance, fault removal and fault forecasting. [8 hours]

DESIGN PRINCIPLES. Error detection and recovery. Fault masking. Redundant architectures. Hardware redundancy: Passive, Active and Hybrid techniques. Information redundancy: Coding. Error detection and correction codes. Self-checking circuits. Software redundancy: design diversity. Byzantine faults. Consensus problem. [10 hours]

QUANTITATIVE EVALUATION. Mathematical models of reliability (R(t)): fault probability density function and failure rate. Exponential failure law of the hardware
COMBINATORIAL MODELS: Series, Parallel, MofN. Fault trees (FT). Failure model and effect analysis (FMEA). [8 hours]

STATE-BASED MODELS. Discrete/continuos time Markov chains: transient analysis, steady-state analysis. Mathematical models of availability (A(t)): repair rate. Steady-state availability. Measurement of Safety (S(t)). Fail-safe and fail-unsafe states. [6 hours]

SOFTWARE RELIABILITY. Software faults. Failure data collection. Data pre-processing for reliability analysis. Software reliability growth models. [4 hours]

RISK ANALYSIS. Hazard analysis, risk assessment, tolerable risk, risk mitigation. [4 hours]

SECURITY ISSUES. Vulnerability and attacks. Security threat modeling. Security-aware risk assessment methods. Attack trees. Adversary view security evaluation. [4 hours]

INTERNATIONAL STANDARDS. Standards for safety-critical systems. Introduction to ISO 26262 "Road vehicles–Functional safety" and ISO/SAE DIS 21434 “Road vehicles-Cybersecurity engineering”. [4 hours]

REFERENCES

- John Knight. Fundamentals of Dependable Computing for Software Engineers, Chapman & Hall, 2012
- D.M. Nicol, W.H. Sanders, K.S. Trivedi. Model-Based Evaluation: From Dependability to Security. In: IEEE Transactions on Dependable and Secure Computing, vol. 1 (1), 2004
- Papers provided by instructors

ASSESSMENT

- Written test + oral examination

ARTIFICIAL INTELLIGENCE FOR SECURITY (6 CFU)

Instructors: Prof. Francesco Marcelloni, Prof. Gianluca Dini

OBJECTIVES. The course aims to introduce the main methods and techniques of artificial intelligence used in information security applications. Further, it intends to discuss the main attacks against artificial intelligence systems and the related defensive techniques.

TENTATIVE PROGRAM

PRELIMINARY CONCEPTS. Introduction to the Machine Learning Process: data collection, data pre-processing, classification, clustering, anomaly detection [3 CFU]

ARTIFICIAL INTELLIGENCE UNDER ATTACK: evasion attacks and data poisoning, defensive techniques [1 CFU]

SECURITY WITH ARTIFICIAL INTELLIGENCE: detection of spam/phishing, detection of intrusions and malware, detection of online frauds, analysis of the cyber threat intelligence [2 CFU]

REFERENCES

- Clarence Chio, David Freeman, “Machine Learning and Security”, O'Reilly, 2018.
- Jiawei Han, Micheline Kamber, Jian Pei, “Data Mining: Concepts and Techniques”, 3rd Edition, Morgan Kaufmann 2012.
- Papers provided by the instructors
- Slides provided by the instructors

ASSESSMENT

- Written test + oral examination + project

PENETRATION AND DEFENCE LABORATORY (6 CFU)

Instructors: Prof. Giuseppe Lettieri, Prof. Pericle Perazzo

OBJECTIVES. The course will give hands-on experience on the most important techniques used in the exploitation of software and hardware vulnerabilities, and the countermeasures adopted to mitigate such attacks.

TENTATIVE PROGRAM

OPERATING SYSTEM: discretionary access control; suid/sgid binaries; special characters; attacks through environment variables (PATH, IFS, ...); symlink attacks; Time-Of-Check-to-Time-Of-Use (TOCTOU) errors; sandboxing via containers (namespaces, control groups); secure monitors (AppArmor); [1 CFU]

PROGRAMMING: secure coding concepts and practices in the C and C++ languages; processes and their address space: the stack and the heap, dlmalloc, mmap()/mprotect(), dynamic libraries, the Global Offset Table (GOT) and the Procedure Linkage Table (PLT); integer overflows; buffer and heap overflows; format string vulnerabilities; use-after-free and double-free errors; code injection and code reuse: return-to-libc, Return Oriented Programming (ROP); Address Space Layout Randomization (ASLR) and Position Independent Executables (PIE); control flow integrity. [3 CFU]

NETWORKING: network scanning, service scanning, fuzzing, OS fingerprinting, DNS bruteforcing, packet sniffing; web applications: mapping, authentication vulnerabilities, login bruteforcing, session management vulnerabilities, session hijacking, SQL injection, LDAP injection, cross-site scripting, web spidering, access control vulnerabilities, XML external entities
[2 CFU]

REFERENCES

- Robert Seacord. Secure Coding in C and C++ (2nd edition). Addison-Wesley Professional, 2013.
- Dafydd Stuttard and Marcus Pinto. The Web Application Hacker's Handbook: Finding and Exploiting Security Flaws. John Wiley & Sons, 2011.
- Stuart McClure, Joel Scambray, George Kurtz.Hacking Exposed (7th edition). McGraw-Hill, 2012
- Chris Anley, The Shellcoder's Handbook: Discovering and Exploiting Security Holes, (2nd Edition), John Wiley & Sons, 2007
- Papers provided by the instructors

ASSESSMENT

- Written test + oral examination + project

BIOMETRIC SYSTEMS (6 CFU)

Instructor: Prof. Enzo Pasquale Scilingo

OBJECTIVES. This course provides fundamentals about techniques to verify or recognize the identity of a living person based on the analysis of biological/physiological traits and/or behavioural features.

TENTATIVE PROGRAM

Biometrics overview (history of biometrics, applications) (4 hours)
Recognition, identification and verification (4 hours)
Privacy, security and ethics (4 hours)
Overview of image processing (4 hours)
Physiological biometric systems: fingerprint recognition, face recognition, iris recognition, retina recognition, hand recognition, vein patterns (16 hours)
Behavioral biometric systems: keystroke dynamics, signature recognition, voice recognition, gait recognition (8 hours)
Multi-modal biometric systems (4 hours)
Biometric applications (4 hours)

REFERENCES

- To be still identified

ASSESSMENT

- Oral examination

ELECTROMAGNETIC SECURITY (6 CFU)

Instructor: Prof. Agostino Monorchio

OBJECTIVES. The main objective of the course is to introduce the problem of security issues due to intentional and unintentional electromagnetic signals as well as countermeasure methods.
Specifically, students will acquire the following competences: *i.* how vulnerable information and communication systems are to radiated and conducted electromagnetic fields; *ii.* how to design electromagnetic shielding and secure rooms for data protection from electromagnetic threats; *iii.* which NATO standards and procedures are currently in use for limiting the information leakage through radiated and conducted electromagnetic signals.

TENTATIVE PROGRAM

- Electromagnetic threats - Vulnerability of information systems to electromagnetic threats [2 hours]
- Undesired emissions from non-intentional sources and E.M. signals interception: E.M. propagation fundamentals [8 hours]
- Spectrum sensing and monitoring - Radiogonometry systems - Signal demodulations - Omnidirectional and directional antennas. [10 hours]
- E.M. Shielding and secure rooms: effect of materials, effect of apertures and cable connections. Coupling mechanisms of e.m. signals. Zoning of infrastructures. [12 hours]
- Active security and intentional interferences: Radio Jamming - Friendly jamming for secure wireless communications. [10 hours]
- Standards and measurement procedures: COMSEC and TEMPEST (Transient Electromagnetic Pulse Emanation Standard) - National and international (NATO) standards - TEMPEST equipment and devices [6 hours]

REFERENCES

- Notes and papers provided by instructors

ASSESSMENT

- Oral examination