

# Computer Security

## SEGC-00 - Overview

Paulo Lício de Geus

LASCA-IC-Unicamp  
Laboratory of Security and Cryptography  
Institute of Computing  
University of Campinas

February 10, 2019

# MO639/MC942

## Summary

This course is about computer security, both from the network view as well as the machine, its OS and applications' point of view. Students shall demonstrate ability in research and programming and shall acquire both overall comprehension and specific abilities.

## Lecture and Office Hours

Lectures: 3:19–21, 5:21–23; Room: 351 (IC3.5)

Office hours: 3:18, 5:20; Room: 26

full mailing list (professor + TAs + students):

[ic-segc-l@g.unicamp.br](mailto:ic-segc-l@g.unicamp.br)

instructors list (professor + TAs): [ic-segc-staff-l@g.unicamp.br](mailto:ic-segc-staff-l@g.unicamp.br)

## Topics

- Introduction
- Basic Knowledge
- Network Security
- Protocol Analysis
- Network Defenses

Planned, but probably only on a follow-up course:

- Machine Defenses
- Application Security
- Web Vulnerabilities

## References

- Nakamura-deGeus - Segurança de Redes em Ambientes Cooperativos, Novatec, 2010 (“more or less” 3rd ed)
- Garfinkel-Schwartz-Spafford - Practical Unix and Internet Security 3rd ed, 2003
- Zwicky-Cooper-Chapman - Building Internet Firewalls, 2nd ed, 2000
- Mann-Mitchell-Krell - Linux System Security 2nd ed, 2002
- Hoglund-McGraw - Exploiting Software: How to Break Code 1st ed, 2002
- Howard-LeBlanc - Writing Secure Code: Practical Strategies and Proven Techniques for Building Secure Applications in a Networked World, 1st ed, 2002
- Anley-Heasman-Lindner-Richarte - The Shellcoder’s Handbook: Discovering and Exploiting Security Holes, 1st ed, 2007
- Davis-Bodmer-LeMasters - Hacking Exposed: Malware and Rootkits, 1st ed, 2009

## Prerequisites

- Computer Networks, strong emphasis on TCP/IP
- C, Python, Assembly Programming
- Linux system level knowledge
- Autonomy to research methods/solutions

## Evaluation

$n$  written Exams: 2/4 of grade

$p$  practical Experiments: 1/4

$k$  extra Assignments: 1/4

Note: the extra assignments factor on the grade may vary in the 0–100% range.

- Exams are individual: plagiarism not accepted
- Practical experiences in teams of 1–2 students

# Introduction

- Failures and vulnerabilities, common practices and excuses, threats
- Vulnerabilities in general, risks, Internet history
- Increase of vulnerabilities and incidents, hacking history
- Classical incidents, parlance, ethics, policies (penetration testing)

# Basic, assumed knowledge

- Unix services: rc scripts, Unix and TCP/IP sockets, xinetd, users/permissions
- Filesystems, suid, PAM, OTP, ACLs, capabilities
- Cryptography: basic functions, algorithms, secure communications, man-in-the-middle  
key management, certificates, PKI, OpenPGP

# Network Security

- TCP/IP: protocols, addressing, CIDR, Ethernet, ARP, sniffing tools  
IP routing, IP spoofing, hijacking, MITM, ARP spoofing
- IP: attacks, fragmentation, Ping of Death
- ICMP: ping, attacks, smurf, redirect, dest unreachable, time exceeded, traceroute
- UDP: header, spoofing, hijacking, NIS, NFS, portscan
- TCP: sequence, windows, flags, portscan, OS fingerprinting, spoofing/Mitnick  
ISN, hijacking, hunt, ACK storm, SYN flooding/cookies, states
- IPsec: AH, ESP, modes, IKE, IPv6
- Wireless: link level, CSMC/CA, modes, associations, energy, WEP/WPA, attacks  
sniffing, crypto attacks, DoS, injection, MITM, protections, 802.1x, wardriving

# Protocol Analysis

- FTP: vulnerabilities, attacks/scan through bounce
- DNS: resolving, zone transfer, spoofing, hijacking, contamination, Kaminsky, Birthday
- Botnets: history, Drive-by downloads, architectures, evolution, uses

# Network Defenses

- IDS, architectures, classification, abuse/anomaly, precision, IDMEF, NIDS/HIDS
- Sniffing: injection, evasion, desynchronization, defragmentation
- TCP reassembly: alert and component correlation, normalization, pre-processing  
alert recombination, verification, attack sequence reconstruction
- packet filtering, secure channels, e-mail, VPN, https

# Machine Defenses

- Software installation, backup, system accounting
- Event logging and monitoring, auditing, password scan  
file integrity, TCPWrapper, SELinux, VM

# Application Security

- local and remote attacks, Unix processes, parameters, filesystems
- TOCTOU attacks, open files, assembly, memory addressing
- x86 registers, data sizes, signals, instructions, execution levels  
stack, frames, prologue/epilogue
- Object files: .COM, a.out, PE, ELF. process data structures, gdb
- Buffer overflows: stack, shell code, syscalls, execve, string, egg, overrun
- Advanced attacks: setjmp/longjmp, off-by-one, array/integer/heap overflow  
teardrop, return-to-libc, chunk management, memory allocation, unlink macro,  
double free, C++ Vtables, format string, locale
- Solutions to overflow: static/dynamic analysis, libc replacement, confining, StackGuard, StackShield, Propolice, Windows, PaX, Armor

# Web Vulnerabilities

- Architecture, http, methods, status, headers, URI/URL/URN, authentication
- State, attached info, CGI, ASP, Servlets, PHP, Java Applets, ActiveX
- Scripting languages, JavaScript, AJAX, XML, Mashups
- Authentication and authorization attacks, injection: PHP, HTML, SQL. XPath, LDAP  
XSS, reflection, solutions, CSRF/XSRF, HTTP attacks, scanners, crawling