Operating Systems Security

General information about this course

Radboud University, Nijmegen, The Netherlands

Winter 2016/2017

About this course

- Lecture (hoorcollege): Monday, 10:30–12:30 in HG00.303
- Exercise class (werkcollege): Friday, 8:30–10:30 in HG00.303
- Exam on Monday, January 23, 12:30–15:30 in LIN 1
- Exam grade is your final grade for this course
- 3 EC points
- Website:
  http://www.cs.ru.nl/~vmoonsamy/os-security-2016.html
- Language of the lectures: English

Teachers

Veelasha Moonsamy
Office: Mercator I, 3.18
v.moonsamy@cs.ru.nl

Peter Schwabe
Office: Mercator I, 3.18
peter@cryptojedi.org

Joeri de Ruiter
Office: Mercator I, 3.07
joeri@cs.ru.nl

Tim van Dijk
t1.vandijk@student.ru.nl

Thom Wiggers
t.wiggers@ru.nl

Homework

- Homework assignments will be online (at the latest) Friday morning
- Homework assignments are due Friday (one week later) by midnight (sharp!)
- Homework submission through Blackboard
- Homework submission in groups of 2 (preferably)
- Grading of homework in **g**, **v**, **o**, and **NSI**
- Grading has no effect on final grade, but:

  More than one NSI and you're not admitted to the exam!

Homework environment

- Programming courses need a computer (with compiler etc.)
- Network security course needs a network that you can break

Homework environment

- Programming courses need a computer (with compiler etc.)
- Network security course needs a network that you can break
- Operating systems security course needs an operating system

Homework environment

- Programming courses need a computer (with compiler etc.)
- Network security course needs a network that you can break
- Operating systems security course needs an operating system
- Part of first assignment: Set up Linux in a virtual machine
- Practical Exercises will mainly use Linux

Examples of what you will learn

- How authentication and authorization works (and fails)

Examples of what you will learn

- How authentication and authorization works (and fails)
- How processes are separated

Examples of what you will learn

- How authentication and authorization works (and fails)
- How processes are separated
- How the OS helps to make memory attacks harder

Examples of what you will learn

- How authentication and authorization works (and fails)
- How processes are separated
- How the OS helps to make memory attacks harder
- Why traditional UNIX security is insufficient today
- Malware and how it hides from malware scanners

Examples of what you will learn

- How authentication and authorization works (and fails)
- How processes are separated
- How the OS helps to make memory attacks harder
- Why traditional UNIX security is insufficient today
- Malware and how it hides from malware scanners
- How operating-systems can be “hardened”:
  - Enforcing mandatory access control
  - Compartmentalization and virtualization
  - Examples: Subgraph OS, Qubes, Android

Disclaimer

- Some things taught in this course are illegal when you do it “in the wild”
- You’re grown up, use your skills responsibly
- If you want to try something out, get consent

Disclaimer

- Some things taught in this course are illegal when you do it “in the wild”
- You’re grown up, use your skills responsibly
- If you want to try something out, get consent
- In the homework, don’t break anything that others still need
- Be careful when attacking your own machine:
  - Make sure that you attack the *virtual* machine
  - Make sure that the attack only affects the virtual machine