



# IT Security Education Program @BME

Levente Buttyán and Tamás Holczer

CrySyS Lab

Budapest University of Technology and Economics

{buttyan, holczer}@crysys.hu

# Outline

---

- some background
- the official program
  - with some examples from the subdomain of network security
- talent management
- lessons learned

# 15 years of evolution

---

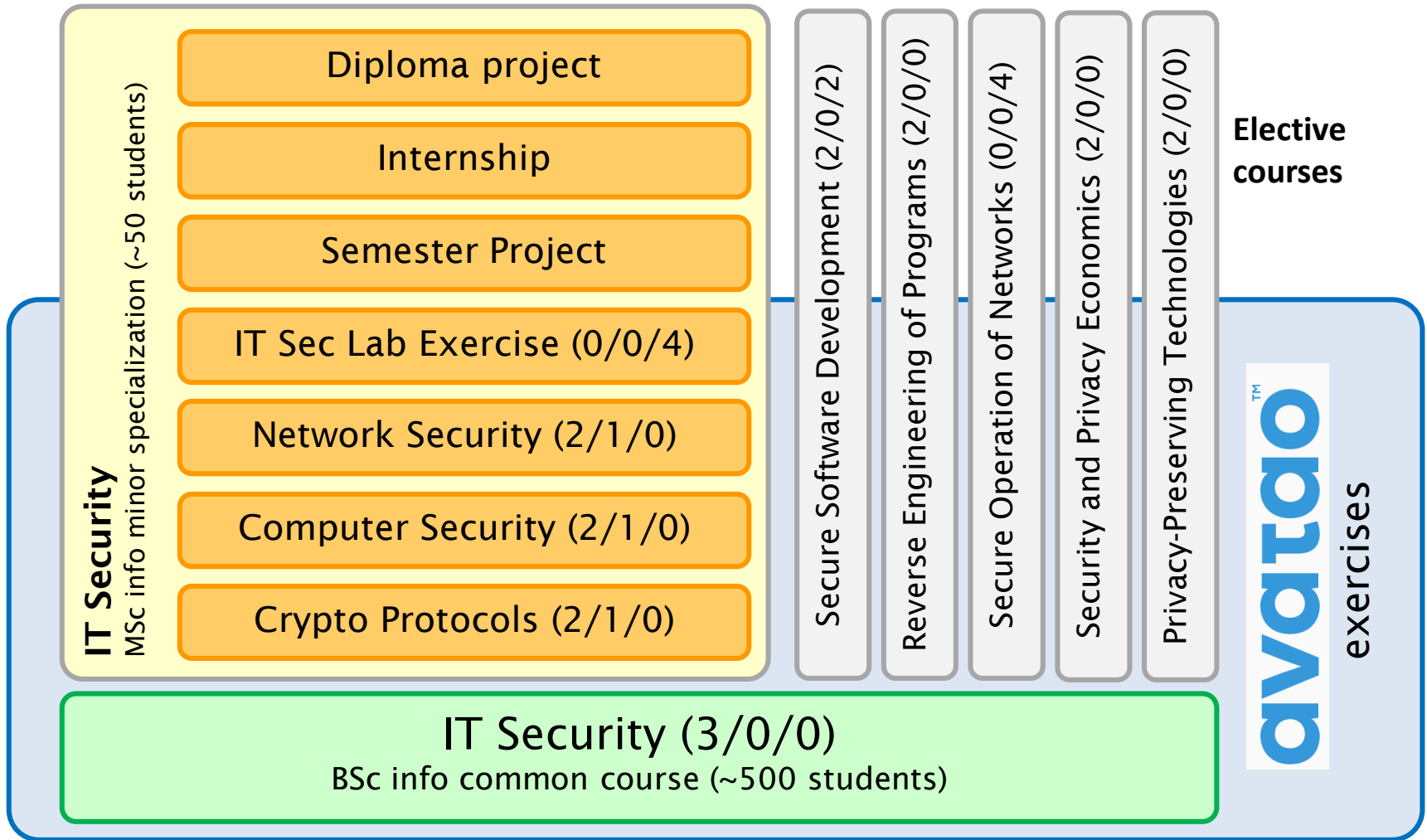
- **2003 – 2008**
  - still the old, 5-year long university program
  - common course for all CS students on *Data Security*
  - specialization on *Security of Infocommunication Systems*
    - » 5 courses + lab course + semester and diploma projects
    - » multiple departments involved
- **2009 – 2014**
  - two-level (BSc, MSc) education system introduced
  - MSc common course for all CS students on *Data Security*
  - major specialization on *Security and Telecommunication Systems*
    - » 5 courses (security and telecommunications) + labs + projects
  - many problems identified
    - » no IT Security at all at the BSc level ☹️
    - » mixing security with telecom in a single specialization was a mistake
    - » drastic decrease in number of students
- **2015 –**
  - entire curriculum has been redesigned
  - problems fixed + program expanded

# (Re)design considerations

---

- BSc level CS students must be exposed to IT security
- IT Security should not be a major
  - the world needs only a few security experts
  - what we need is a large number of engineers (software, network, embedded, ...) with strong IT security awareness
- the outcome of the program (fresh engineers) must be "usable" by industry
  - we organized a round table discussion and asked about the needs and expectations of industry partners
  - two important aspects identified that shaped our program later:
    - » more emphasis on software security
    - » capability of learning new things is more important than knowing specific technologies

# The official program



more info: <http://www.crysys.hu/education/>

# IT Security (common course)

---

- goals:
  - cover a wide spectrum and don't go deep into details
  - appetizer for the MSc *IT Security* minor
- expected learning outcome:
  - awareness of security problems in information and communication systems (including software)
  - understanding of basic security concepts, services, and mechanisms
  - limited application skills, mainly selection of fitting existing solutions
  - no design and analysis capabilities
- main challenges:
  - large number of students
    - » practice sessions are not feasible
    - » difficulties with exams, administrative burden
  - diverse background and level of engagement
- our approach: performance at stage
  - multiple lecturers (each one is expert on a given topic)
  - interesting highlights, examples, case studies (e.g., how we discovered Duqu)
  - on-line homework exercises (with solution hints)

# IT Security (MSc minor specialization)

---

- goals:
  - go somewhat deeper into major subdomains of IT security
    - » Computer Security (including software)
    - » Network Security
    - » Cryptographic Protocols (applied crypto)
  - supplement major specializations on Software Engineering, Networking, and Embedded Systems
- expected learning outcome:
  - deep understanding of security problems in information and communication systems (including software)
  - understanding contemporary approaches, tools, and mechanisms for addressing security problems
  - practical skills in identifying fitting existing solutions, and in deploying, configuring, and operating them
  - practical skills in designing new security solutions in certain application domains (e.g., developing a secure protocol or API)

# IT Security (MSc minor specialization)

---

- leverage diverse forms of learning
  - lectures
  - classroom exercises (often include demos)
  - lab exercises
  - semester projects
    - » related to our research projects or proposed by some industry partner
    - » projects can be done in teams (collaboration, team work)
    - » project plan, project report, project presentation (soft skills)
  - mandatory internship
    - » 6 weeks at industry partners
  - diploma project
    - » 1 year individual engineering work

# Cryptographic Protocols

---

- topics:
  - symmetric key ciphers
    - » stream ciphers
    - » block ciphers and block encryption modes
    - » attacks on CBC mode
  - hash functions and MAC functions
  - asymmetric key ciphers and digital signature schemes
  - random number generation
  - key exchange protocols
  - PKI
  - examples for secure channel protocols (WiFi WPA2, TLS)
  - anonymous communications
  
- + classroom exercises
- + homework assignments

# Computer Security

---

- topics:
  - user authentication and access control in operating systems
  - memory corruption attacks (e.g., buffer overflow)
  - secure coding methods, security testing of software
  - web security (attacks and defenses at client and server side)
  - browser security
  - mobile platform security
  - cloud security
  - trusted computing and tamper resistant devices
  - incident response and digital forensics
  
- + classroom exercises
  
- + homework assignments

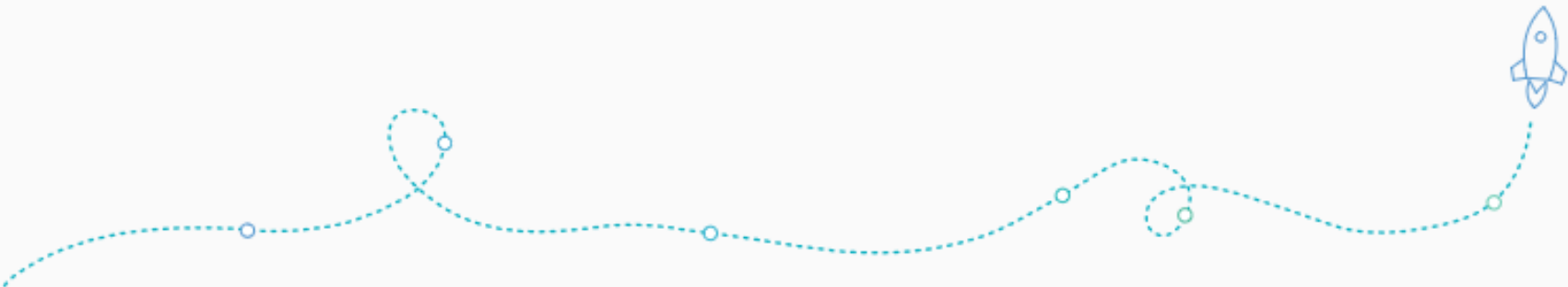
---

**avatao offers hands-on IT security exercises  
for people to sharpen their skills**







**„I hear and I forget. I see and I remember. I do and I understand.”**  
– Confucius

# avatao – on-line IT security practice platform



Secure Coding      Ethical Hacking      Security Tools      Incident Response

 <h3>C/C++</h3> <p>Writing secure code in C is challenging. As C is not a memory safe language you are responsible entirely to manage the memory. Learn how C really works and what issues you have to take care of.</p> <p><a href="#">Preview</a></p>	 <h3>Embedded Pwn</h3> <p>This path guides you through the realm of various architectures with a dedicated focus on ARM. Sharpen your skills by exploiting exotic architectures.</p> <p><a href="#">Preview</a></p>	 <h3>sqlmap</h3> <p>sqlmap is an open source penetration testing tool that automates the process of detecting and exploiting SQL injection flaws and taking over of database servers.</p> <p><a href="#">Preview</a></p>	 <h3>Malware Analysis</h3> <p>Learn the basics of malware analysis. Complete basic memory and disk forensics, report analysis and malware reversing tasks on a well-known APT which stayed under the radare for ages.</p> <p><a href="#">Preview</a></p>
--	--	--	---



# Web Security

by CrySyS Student Core

## Path

Details

Statistics

Challenges

Cookie Monster

BetterManager

Better Status

Let the Files be Included

Company Homepage

Company Homepage (Secure Version)

PHP Sadness

Sadness 1

## Sadness 1

by Gábor Szarka

Easy 25 Hints 539

365

Tweet Share

You have already solved this challenge. You don't need to solve it again, but you are free to do so.

### Skill tags

[Web Security](#) [PHP](#) [Offensive](#) [OWASP Top 10](#)

### Description

Brainy and his smart friends wrote an "intranet portal",... and then deployed it on the Internet. They did not trust any professional web developers, but developed the whole thing themselves. Well, the result is questionable at best. Although it is a "secure" portal (with password authentication, encrypted passwords, encrypted data storage) it contains a number of amateur mistakes.

In this challenge, you will meet the first version of the portal.

#### Goal:

Hack the login screen! There is an SQLite database behind the website and the login function is vulnerable to SQL Injection - that means, you can login without password. Check out the recommended readings, it will help you solve the challenge!

### Hints

[Guideline](#)

-10%

#### Start your environment

Hey there! To start your environment, please click on the button below.

[Start now!](#)

### Recommended readings

[OWASP top10](#)

[PHP Sadness](#)

[SQLi](#)

...

exercises for a night, or two.

challenges you will  
led to be ...

[Show more](#)

n this path.

# Advantages of avatao

---

- for students
  - **no need for installation**, a web browser is sufficient
  - **fast feedback**, submitted solutions are evaluated immediately
  - if something goes wrong, challenges **can be restarted at any time**
  - for most challenges, there's a **step-by-step solution guide**
- for teachers
  - **no need for building and mainting an infrastructure**
  - **500+ challenges** (and growing continuously)
  - **creating a new path of challenges takes only a few minutes**
  - usable for **homework assignments, lab exercises, exams**
  - **free** (under some conditions)



# Network Security

---

- topics:
  - network penetration testing and ethical hacking
  - network perimeter defense (firewalls)
  - network intrusion detection (IDS, SIEM)
  - network log collection and analysis
  - honeypots and sandboxes
  - spam filtering and DDoS protection
  - network infrastructure security (DNS, routing)
  - layer 2 security
  - security of special networks (industrial, vehicular, embedded)
  - botnets
  
- + classroom exercises
- + homework assignments

# Classroom exercises for Network Security

---

- updated in each year
- examples (from 2018):
  - ethical hacking tools, demos
    - » introduction to basic tools
    - » some advanced case studies
    - » help and support for lab exercises
  - Tor
    - » how to setup a Tor node?
    - » monitoring and analysis of forwarded traffic
  - network device setup
    - » based on Packet Tracer
    - » routing, filtering
    - » DHCP, NAT
  - ...

# Classroom exercises for Network Security

---

- examples (from 2018) cont'd:
  - honeypots and sandboxes
    - » Sandboxie
    - » VirusTotal
    - » JoeSandbox
    - » Modern Honeynetwork
    - » task: detection of Cowrie HP
  - smart home testbed
    - » demo of security of air conditioning, shades, central heating etc.
  - network forensics
    - » given a PCAP with some attack
    - » task: who? when? what? how?

# Homework assignment for Network Security

---

- software development
  - ~1 day work
  - Python, C#, Go
- task: implement client to connect to our server
  - steps:
    - » port knocking opens a TCP port
    - » user name request
      - challenge: solve some equations
    - » ask for hash of last solution+USER
      - challenge: create a hash with given prefix
    - » download a newly generated short-term X.509 certificate
    - » open an https connection using the certificate to get the FLAG
  - submit documentation, software, FLAG

# Homework assignment for Network Security

---

- system configuration
  - ~2-3 days work
  - issues with versions (e.g.: GnuTLS vs OpenSSL)
- task: install and configure a system to communicate with our server
  - main parts
    - » install OS on VM (recommended, Debian or Ubuntu)
    - » install and configure OpenVPN client to connect to server
    - » install and configure DNS server
      - DNSSEC, DMARC, DKIM, SPF record
    - » install and configure SMTP server (we send email to it)
      - DKIM signature, TLS
  - submit documentation, configuration files

# IT Security Lab

---

- 9 lab exercises in 1 semester
- largely based on virtualized infrastructures
- done in teams of two
  
- lab topics:
  - **Introduction to computer network security**
  - **Penetration testing**
  - Malware analysis
  - Public key cryptography and PKI
  - OS level access control
  - **Firewalls**
  - **Shell Control Box**
  - Memory corruption attacks
  - **Cyber-Physical Systems**

# Lab exercises related to network security

- Introduction to computer network security
  - sniffing (tcpdump, tshark, wireshark)
  - ARP, IP, MAC (attacks)
  - password capturing
  - scanning (nmap)
- Penetration testing
  - network scanning
  - vulnerability analysis
  - exploitation
- Firewalls
  - IPTables
  - Zorp (application layer filtering)
- Shell Control Box
  - admin auditing
- Cyber-Physical Systems
  - attacks with physical consequences



# Elective courses

---

- goal:
  - go even deeper into some selected topics
- courses:
  - Secure Software Development (lecture and lab exercise)
  - Reverse Engineering Programs
  - Secure Operation of Networks (labs based on CISCO material)
  - Foundations of Cryptography (theory oriented course)
  - Privacy Preserving Technologies
  - Economics of Security and Privacy (incentives and game theory)

# Secure Software Development

---

- alternating offensive and defensive topics
  - crypto (breaking crypto, using crypto libraries)
  - web apps (attacks and defenses on both the client and the server side)
  - managed languages (attacks and defenses in Java and C#)
  - Android application security (attacks and defenses, code signing)
  - native languages (attacks and defenses in C and C++)
  - API attacks

# Experiences so far

---

- the program became popular among students
  - IT Security common course got very positive student feedback
  - IT Security minor is the second most popular minor (after Mobile Software Development) → 50+ students
  - Secure Software Development is among the most popular elective courses (35 available places filled up on the first day of course selection)
- industry partners are happy too
  - many semester projects are related to partners' topics
  - many students are interested in internships at partners
  - some industry partners offer financial support to the lab in the form of scholarships to students and to faculty staff
- avatao is highly appreciated by students
  - they often use it for solving extra challenges (not mandatory)

# Talent management

---

- IT security courses in the university curriculum are designed for the average students
- special attention is needed to identify outstanding students, make them interested in IT security, and help them growing their talent



# About talent

---

"I am willing to guarantee that you will not read a more important and useful book in this or any other year."

—TOM PETERS, *coauthor of The Search of Excellence*

THE

GREATNESS ISN'T BORN.  
IT'S GROWN.

CODE

GREATNESS ISN'T BORN.  
IT'S GROWN. HERE'S HOW.

DANIEL COYLE

*author of the New York Times bestseller Lance Armstrong's War*

# The CrySyS Student Core

---

- an invite-only group of students who are enthusiasts and who have already proved their aptitude for IT security
- how to get invited?
  - score among the best students at our annual CrySyS Security Challenge
  - provide an impressive performance during a student semester project



# Operation of the Core

---

- weekly meetings (including the holiday seasons)
  - a member presents work he has done recently
  - invited talks from outside
  - visiting other hacker communities
  - joint practicing and preparation for CTF games
    - » discuss tutorials and write-ups
    - » solve challenges from previous years
- participation at hacking contests (CTF games)
  - usually remote participation
  - sometime travelling (needs some funding)
- creating avatao challenges
  - for the CrySyS Security Challenge and IT Security Bootcamp
- supervising bootcamp sessions

# Operation of the Core

---

- members really enjoy to be part of the Core
  - develop unique knowledge and skills
  - feel good in a social sense
  - have independence and responsibility



# Operation of the Core

---

- faculty members minimize their control on the Core
  - attract and prepare interested students
  - advise the selection of new Core members
  - acquire financial support for the operation of the group



# The Core is a *community of practice*

---

“a group of people who share a concern or a passion for something they do and learn how to do it better as they interact regularly”

-- Etienne Wenger,1991

1. a shared domain of interest
2. joint activities and information sharing
3. development of a shared “repertoire of resources”



# Efficiency by *situated learning*

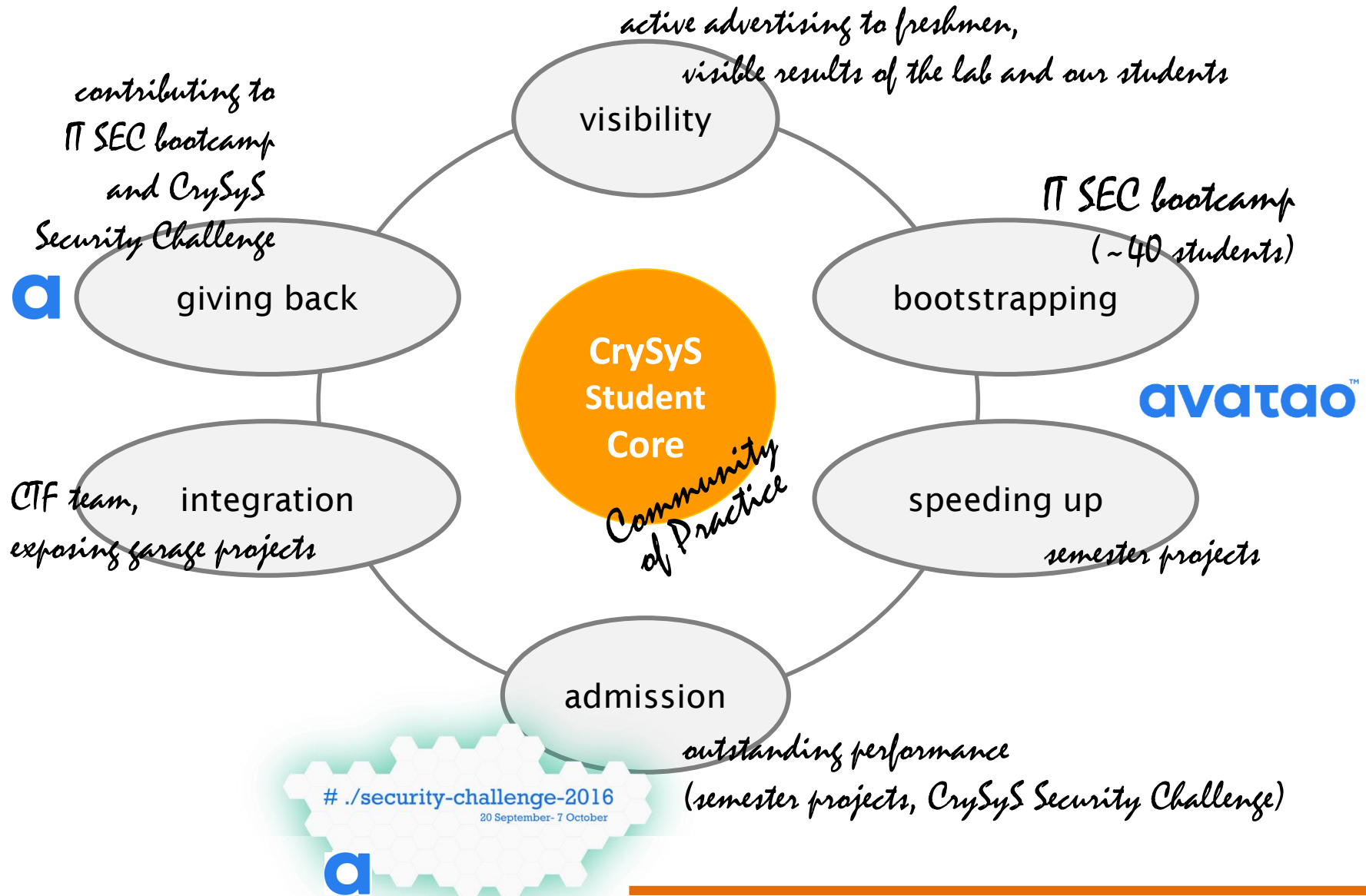
---

“learning that takes place in the same context in which it is applied”



- learning through the relationships between people (in a community of practice)
- learning by doing (under some supervision)
- better understanding
- more efficient for hands-on skills (than lectures)

# Sustainability



# Success is measurable



DefCon CTF finalist (2015, 2016, 2017)

2 UCSB iCTF 2013

2 EKOPARTY CTF 2015

4 ASIS CTF Quals 2015

7 RuCTFE 2015

1 UCSB iCTF 2014

7 HITCON CTF 2015 Final

1 ASIS CTF Finals 2015

3 RuCTF Finals 2016

3 Nuit du Hack CTF Finals 2014

3 CONFidence CTF 2015

6 Hack.lu CTF 2015

6 Belluminar 2016

+ former core members were hired by



tresorit



# More information...

---

This paper was published at the 2016 Usenix Workshop on Advances in Security Education.

## Mentoring talent in IT security – A case study

Levente Buttyán  
*CrySyS Lab, BME*  
*Budapest, Hungary*  
*buttyan@crysys.hu*

Márk Félegyházi  
*CrySyS Lab, BME*  
*Budapest, Hungary*  
*mfelegyhazi@crysys.hu*

Gábor Pék  
*CrySyS Lab, BME*  
*Budapest, Hungary*  
*pek@crysys.hu*

### Abstract

Talent management is usually not well-supported by traditional curricula, because university courses are typically designed for a large number of average students and not for the few outstanding ones. In this paper, we share our experiences on running a talent mentoring program in IT security at our university. We describe the whole process from increasing awareness of IT security among students, via maintaining a community of practice where they can improve their skills, to finally connect them to well-established IT companies. We also introduce avatao, a platform to support hands-on IT security practice. Our methods could serve as a blueprint to establish a successful talent management program in IT security in a typical academic environment.

and Network Security, as well as an IT Security Lab and individual semester projects, and (3) a number of elective courses on specific topics (such as reverse engineering programs, secure software development, secure operation of computer networks, and the mathematical foundations of cryptography).

The first issue is that our introductory level course is taught in the 6th semester, and this is the first time students are exposed to the domain of IT security. Unfortunately, by that time, most of the good students are already involved in other fields of computer science that they encountered earlier in their study. Another problem is that our courses are designed to satisfy the needs of the average students, and not the outstanding ones. Furthermore, our introductory level course is given to around 500 students; hence, it is difficult to identify the interested ones

more info: <http://core.crysys.hu/>

# Lessons learned

---

- fixing mistakes made in the design of an education program is difficult and takes a long time
- build a live relationship with industry
- make your program scalable by using on-line platforms
- talent management is important, but requires extra work