OWASP Hackademic Challenges
A practical environment for teaching application security

Dr Konstantinos Papapanagiotou
konstantinos@owasp.org
@kpapapan

Spyros Gasteratos
spyros@owasp.org
@northdpole
Kostas

• 10+ years of experience in InfoSec as a consultant and researcher
• Currently: InfoSec Services Team Lead at OTE S.A.
• Involved with OWASP since 2005 as the Greek Chapter Leader
  • Co-Started the Hackademic Challenges Project in 2011.
  • Organized the OWASP AppSec Research 2012 conference.
• Research
  • PhD in Trust in MANETs – Univ. of Athens, GR
  • 10+ publications and 50+ citations
  • Teaching InfoSec and AppSec at Greek universities

Spyros

• Freelance web developer, uni student.
• Currently: Web Developer at Telesto Technologies Ltd.
• Involved with OWASP since 2010
  • Co-Organized OWASP AppSec Research 2012
  • Joined the Hackademic development team in 2011.
• Participated as a student at GSOC in 2012 and as a mentor in 2013
• Coming soon: Internship at CERN Security Team
No students were harmed in the making of this project
What is hackademic?

- Relatively simple challenges, mainly web-based that involve JavaScript, PHP, web server mis-configuration, etc.
- The goal is to present the general idea behind certain security issues, rather than having complex, sophisticated challenges.
- Variety of topics covered, rather than going too deep into one of them.
- Some may seem simple and ‘old-fashioned’ (e.g. XSS) but websites vulnerable to them still exist!
Challenges - Motivation

• Teach security at 300+ students each semester.
• Students have varying background, skills and knowledge.
• University courses are too theoretical.
• Every student (and every teacher) wants to have a “pen-test lab”
labs are cool but...

- Hard to build/maintain (especially if students practice hacking on them!)
- Most existing vulnerable apps (e.g. WebGoat) are nice for demos or self-teaching but not designed for use in a class-lab environment.
- Need to promote discussion and interaction
- Need to introduce the “attacker’s perspective”
Vasilis and Alex have to teach more than 300 students/semester at TEI of Larissa.

Hackademic is born

Hackademic is presented at the OWASP Summit

Several universities around the world use hackademic

Hackademic becomes an OWASP project

NJIT team builds a joomla frontend

New, custom frontend as a result of GSOC 2012

Spyros starts his final year thesis on Hackademic

Hackademic gets a slot at GSOC 2012

Plugin API as a result of GSOC 2013
hackademic v0.2

- Based on a Joomla frontend
- 10 web application security challenges
  - From simple to intermediate
  - Topics: information gathering, xss, encoding, etc.
- More challenges came in later
  - Crypto
  - SQLi
  - Entire VMs
Rules for Challenges

• There must be a scenario/story/myth.
• It must target a specific topic.
• The solution should be single and deterministic.
• There should be a “timeline” and a strategy for delivering the knowledge behind the set of all these exercises
• The difficulty in solving the exercises should escalate
Student’s reactions
it works!

• Student expect typical “text-based”, theoretical lectures

• Instead, for a minute they get to ‘think like an attacker’.

• Several students, upon completion of the given challenges, attempted the next ones. Some did so at home ⇒ They liked it!

• Can lead to several discussions and input from students
Questionnaires

• 25 questions in total
• Approx. 500 students have replied up to now
  • Looking to automate this...
• Questions on the level of skills/knowledge
• Feedback on the use of challenges
Usefulness of exercises

- Significantly: 50%
- Not at all: 3%
- Little: 12%
- Very much: 35%
Why a new version?

• Lots of interest to build new challenges
• Similar interest to use hackademic in various classes/universities.

• Need to work on usability and ease of installation
• Need to facilitate importing new challenges
New features:

- An entirely new interface
- Installer
  - Facilitates/automates installation
  - Prerequisites: Apache/PHP/MySQL (XAMPP, LAMP, etc.)
Installer

- Facilitates/automates installation
- Prerequisites: Apache/PHP/MySQL (XAMPP, LAMP, etc.)
Role-based access

- Admin
- Student
  - Can view progress, his rank among his class and global rank
- Teacher
  - Can create classes and assign students to them
  - Can monitor students’ progress and score
  - Can post articles
Class/Students Management

Added usability for teachers:

• Create/Manage/Archive Class
• Assign students to classes
• Assign challenges to classes (students)
• Monitor student/class progress
• Add announcements/articles
Importing new challenges

Ability to import new challenges
• (Nearly) automated procedure
• Workflow:
  • Teacher uploads challenge as .zip file
  • Challenge is automatically placed in the correct directory
  • Admin checks challenge
  • Admin published challenge
  • Teacher can add challenge to class
Excellent

Good

Average

Poor
Scoring System

• Instead of a simple, binary system we implemented a complex way of marking:
  • Maximum attempts
  • Time for completion
  • Attempts/minute
  • Player keeps trying after being successful
  • Use of known user agents (vulnerability scanners)
  • Cheating detection: too many challenges solved with 1 attempt only.
Security Enhancements

- Use of ESAPI-PHP for:
  - Input validation
  - Escaping
  - Session management
- Access control improvements
- Quite complex (along with session management) as there are two different levels of access: CMS and challenges.
Other features

- Easy to use installer (all you need is Apache/MySQL/PHP)
- Multiple solutions per challenge
Extending Hackademic

- Plugin API
- Endless possibilities to extend Hackademic
  - Add or change functionality
  - Create themes
- Plugins work by defining actions that hook execution points and callbacks that do the work
- Plugins are manageable through the UI

More info: https://github.com/span/hackademic/wiki/Plugin-API-Overview
Demo
Hackademic Ecosystem

Hackademic/OWASP Academies

Students

Teachers

Professionals/Community
(Near) Future Work

• Documentation – user’s guide
• Release a hardened VM version
• Migrate from esapi-php to a more modern, actively developed library
• Add integrated questionnaires for students/teachers (for stats and/or exams)
• Add teaching content
• Add more challenges – engage with the community
• Implement reporting mechanism
Thanks

Dr. Vasilis Vlachos
Andreas Venieris
Anastasis Stasinopoulos
Alex Papanikolaou
Pragya Gupta
Daniel Kvist
Fotis Liatsis
Nikos Danopoulos
Petros Andreou
Thank You!

http://hackademic.eu

Next Stop:
OWASP AppSec USA Project Summit

Dr Konstantinos Papapanagiotou
konstantinos@owasp.org
@kpapapan

Spyros Gasteratos
spyrosgaster@gmail.com
@northdpole