

# RSA®Conference2018

San Francisco | April 16 – 20 | Moscone Center

SESSION ID: HUM-F02

## HOW THE BEST HACKERS LEARN THEIR CRAFT

**David Brumley**

CEO, ForAllSecure  
Professor, Carnegie Mellon University  
@thedavidbrumley



#RSAC

# George Hotz



- First iPhone JailBreak
- Playstation 3
- Zero-days in Adobe, Firefox, ...

# Richard Zhu



- Mozilla Firefox ('18)
- Microsoft Edge ('17 & '18)
- iOS Safari ('17)



#1 US Team since 2011  
#1 Overall 3 of past 7 years  
4 DEFCON wins – most wins in DEFCON history



# Learning Objectives



1. Understand how top experts use capture the flag competitions for deliberate practice.
2. See how hacking competitions gamify learning computer security.
3. Learn how to set up a system for building a top-ranked hacker culture.

# Basic Knowledge

After opening the robot's front panel and looking inside, you discover a small red button behind a tangle of wires. Pressing the button lights up the robot's primary screen. It glows black and quickly flashes blue. A line of small text types out:

```
ERROR: 0x00000023
```

The text refreshes and displays the prompt:

```
FILE SYSTEM RECOVERY INITIATED...  
FILE SYSTEM COULD NOT BE IDENTIFIED...  
PLEASE ENTER FILE SYSTEM FORMAT:
```

Submit!

Question

Flag

# Basic Knowledge

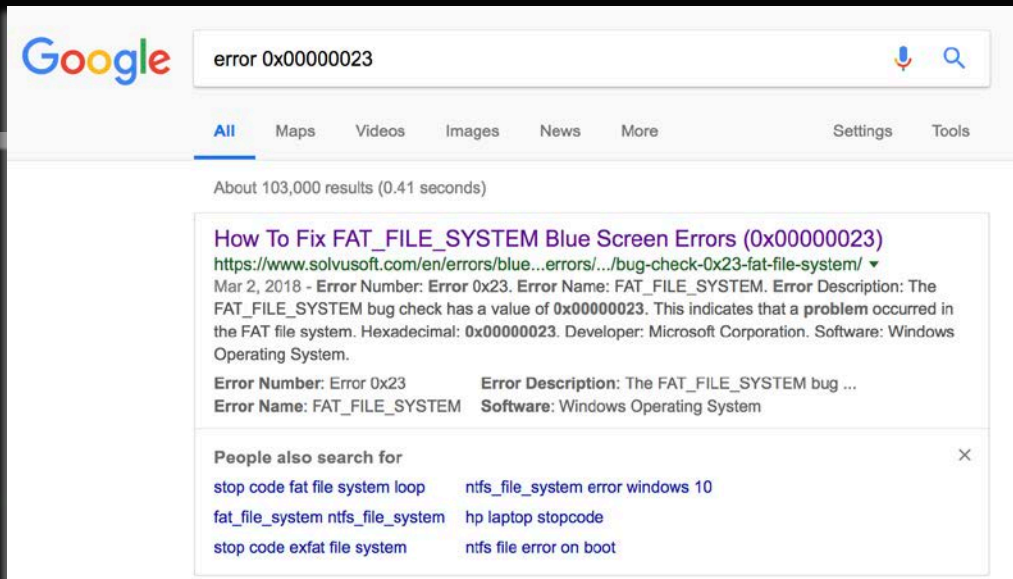
After opening the robot's front panel and looking inside, you discover a small red button behind a tangle of wires. Pressing the button lights up the robot's primary screen. It glows black and quickly flashes blue. A line of small text types out:

ERROR: 0x00000023

The text refreshes and displays the prompt:

FILE SYSTEM RECOVERY INITIATED...  
FILE SYSTEM COULD NOT BE IDENTIFIED...  
PLEASE ENTER FILE SYSTEM FORMAT:

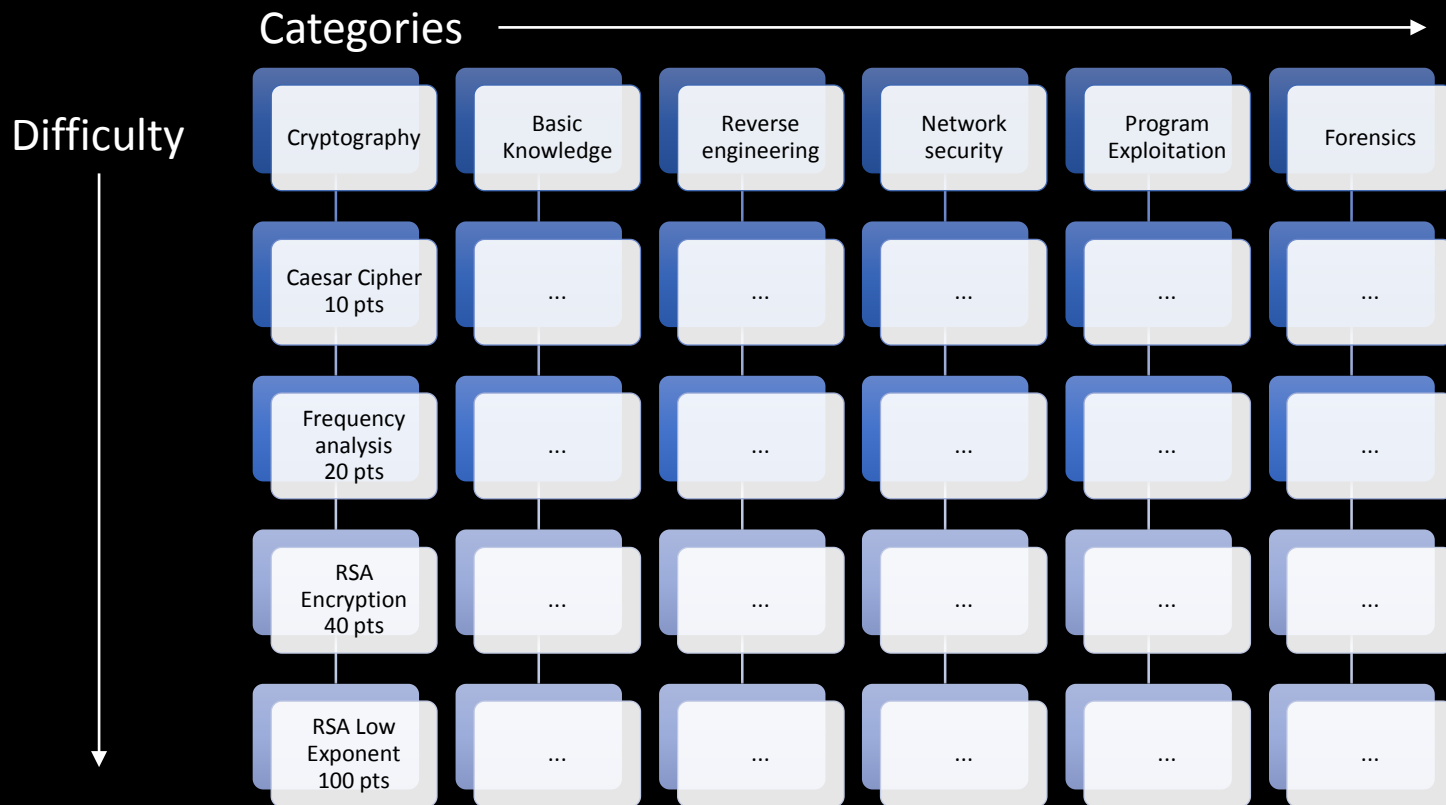
Submit!



Google search results for "error 0x00000023". The search bar shows the query. Below the search bar, the "All" tab is selected. The results show "About 103,000 results (0.41 seconds)". The first result is titled "How To Fix FAT\_FILE\_SYSTEM Blue Screen Errors (0x00000023)" with a link to <https://www.solvusoft.com/en/errors/blue...errors/.../bug-check-0x23-fat-file-system/>. The description states: "Mar 2, 2018 - Error Number: Error 0x23. Error Name: FAT\_FILE\_SYSTEM. Error Description: The FAT\_FILE\_SYSTEM bug check has a value of 0x00000023. This indicates that a problem occurred in the FAT file system. Hexadecimal: 0x00000023. Developer: Microsoft Corporation. Software: Windows Operating System." Below the description, it lists "Error Number: Error 0x23", "Error Name: FAT\_FILE\_SYSTEM", and "Error Description: The FAT\_FILE\_SYSTEM bug ...". A section titled "People also search for" lists related queries: "stop code fat file system loop", "ntfs\_file\_system error windows 10", "fat\_file\_system ntfs\_file\_system", "hp laptop stopcode", "stop code exfat file system", and "ntfs file error on boot".

Answer: FAT

# Jeopardy-Style CTF





# PRESENTING TOASTER WARS



## HIGH SCHOOL HACKING COMPETITION

APRIL 26TH 2013 - MAY 6TH 2013

Sponsorship provided by the NSA.

### TOASTER WARS

When a robot from space crash lands in your backyard it's up to your hacking skills to fix him and uncover the secret he carries...

The Adventure Will Run: April 26th 2013 - May 6th 2013. Registration is open now!

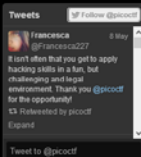


### CREATED BY

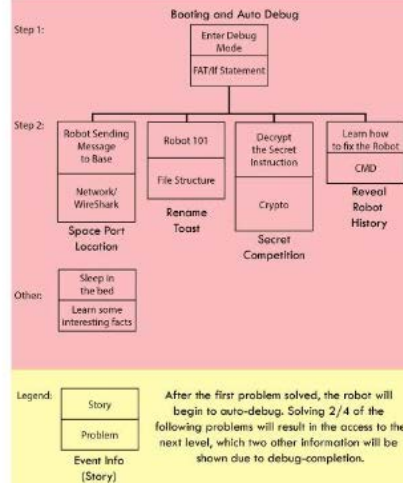
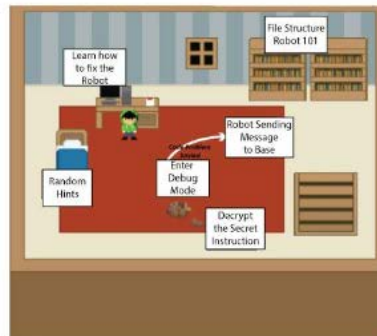
Toaster Wars is a collaboration of the [Pied Parliament of Pining](#) (PPP) of Cytan and [Osmis](#) of the [Entertainment Technology Center](#).

Both teams are student-run and based at Carnegie Mellon University.

OSIRIS



## House Story/Problem Design



# Gamify Learning

# CTF Principles



Applied, deliberate practice



Autodidactic learning



Creative problem solving



## Applied, deliberate practice

Cryptography

Reverse  
engineering

Network  
security

Program  
Exploitation

Forensics

# “Buffer Overflow”

User input size programmed

10 bytes  
long

50 bytes long

User input given

**Class: 90 minutes lecture**

1. Sophomore course
2. Students understand concept

**Challenge: Apply knowledge**

1. Real program buffer size?
2. Create long user input?
3. Create specific attack input?
4. ...

# CTF Problem: Show You Can Do It

readasm (615 solves)

5 POINTS

At the end of this sequence of instructions, how many bytes separate esp and the stored return address? Assume that we called this function using standard 32-bit x86 calling conventions.

```
804847c: functionname:
804847c: push %ebp
804847d: mov %esp,%ebp
804847f: sub $0x70,%esp
8048482: movl $0x0,0x4(%esp)
804848a: movl $0x8048580,(%esp)
```

Answer in **decimal**

🔍 HINTS

- ? You may find [this reference](#) informative.
- ? Put your answer in decimal.
- ? Not sure about something? Google for it.

SUBMIT



2

# Autodidactic Learning

Auto: self

didactic: learn



Romantic, but not real



Image: <http://www.starwars.com/news/6-great-quotes-about-the-force>

## 2

## Auto-didactic Learning

readasm (615 solves)

5 POINTS

At the end of this sequence of instructions, how many bytes separate esp and the stored return address on the program's stack?  
Assume that we called this function using standard 32-bit x86 calling conventions.

```
804847c functionname:
804847c: push %ebp
804847d: mov %esp,%ebp
804847f: sub $0x70,%esp
8048482: movl $0x0,0x4(%esp)
804848a: movl $0x8048580,(%esp)
```

Answer in **decimal**

▼ ? HINTS

- ? You may find [this reference](#) informative.
- ? Put your answer in decimal.
- ? Not sure about something? Google for it.

Richard didn't  
know either.  
He read up.



## 2

## Auto-didactic Learning

readasm (615 solves)

5 POINTS

At the end of this sequence of instructions, how many bytes separate esp and the stored return address on the program's stack?  
Assume that we called this function using standard 32-bit x86 calling conventions.

```
804847c functionname:
804847c: push %ebp
804847d: mov %esp,%ebp
804847f: sub $0x70,%esp
8048482: movl $0x0,0x4(%esp)
804848a: movl $0x8048580,(%esp)
```

4 byte ret address

4 byte ebp

0x70 byte sub  
= 112 bytesAnswer in **decimal**

## HINTS

- ? You may find [this reference](#) informative.
- ? Put your answer in decimal.
- ? Not sure about something? Google for it.

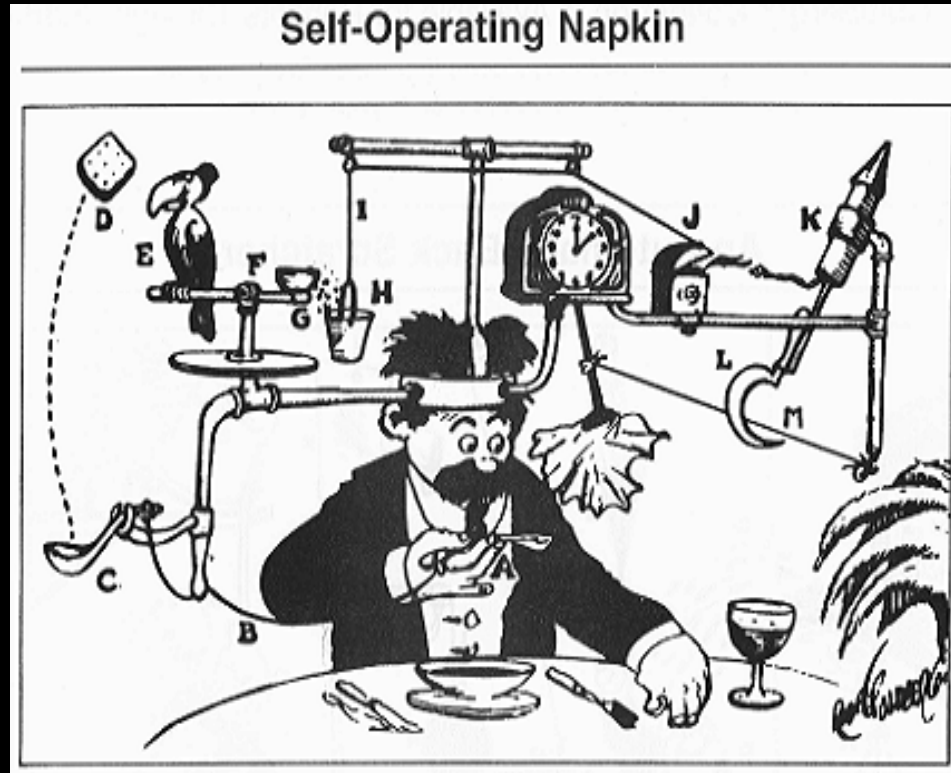
Enter flag...

SUBMIT

Answer: 116

# 3

## Creative Problem Solving



# Solution vs Result

readasm (615 solves) 5 POINTS

At the end of this sequence of instructions, how many bytes separate esp and the stored return address on the program's stack? Assume that we called this function using standard 32-bit x86 calling conventions.

```
804847c: functionname:
804847c: push %ebp
804847d: mov %esp,%ebp
804847f: sub $0x70,%esp
8048482: movl $0x0,0x4(%esp)
804848a: movl $0x8048580,(%esp)
```

Answer in **decimal**

HINTS

- ? You may find [this reference](#) informative.
- ? Put your answer in decimal.
- ? Not sure about something? Google for it.

Enter flag... SUBMIT

## Problem in CTFs: “find the flag”

- Solution is flag submitted, like 116 here.
- Wrong! flag  $\neq$  solution
- Flag = result of the solution

## This simplicity is fundamental to creativity

- Check only the results (i.e., the flag)
- Place few constraints on the solution

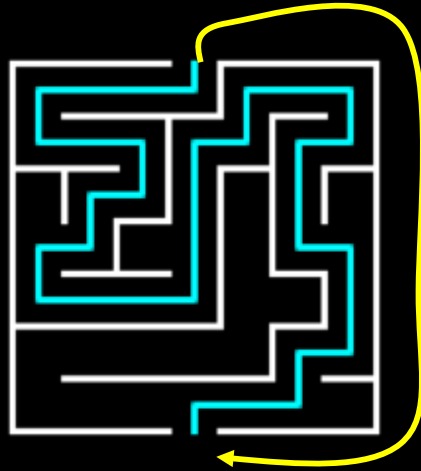
# Creativity in problem solving

$$\begin{aligned} &2 \times 2 \\ &= \left(\frac{3}{2} + \frac{1}{2}\right)^2 \\ &= \frac{3^2}{2^2} + \frac{3}{2} \frac{1}{2} + \frac{1}{2} \frac{3}{2} + \frac{1}{2^2} \\ &= \frac{9}{4} + 2 \times \frac{3}{4} + \frac{1}{4} \\ &= 1.75 + 1.5 + 0.25 \\ &= 4 \end{aligned} \quad \left. \vphantom{\begin{aligned} &2 \times 2 \\ &= \left(\frac{3}{2} + \frac{1}{2}\right)^2 \\ &= \frac{3^2}{2^2} + \frac{3}{2} \frac{1}{2} + \frac{1}{2} \frac{3}{2} + \frac{1}{2^2} \\ &= \frac{9}{4} + 2 \times \frac{3}{4} + \frac{1}{4} \\ &= 1.75 + 1.5 + 0.25 \\ &= 4 \end{aligned}} \right\} \text{All valid approaches}$$



# Hack.IM CTF 2012 Example

- Break into PHP-powered website made by organizers
- Reference solution used XPath injection vulnerability



Dutch solution found flaw in PHP, a major programming language



Ron Rivest

Adi Shamir

Len Adleman

RSA is considered mathematically  
secure.

The diagram consists of two stacked rounded rectangular boxes on the left. The top box is orange and contains the text 'Software Security'. The bottom box is blue and contains the text 'Cryptography'. To the right of these boxes are two yellow arrows pointing left towards the boxes. The top arrow points to the orange box, and the bottom arrow points to the blue box. To the right of the arrows is a block of text.

Software Security

Cryptography

The crypto (math)  
doesn't talk about  
code (app)

# Timing attacks against crypto

Suppose my wife asks:  
“Do I look fat in this outfit”

Any hesitation reveals information.

## Crypto Software

If key = 1, 1 sec. to decrypt

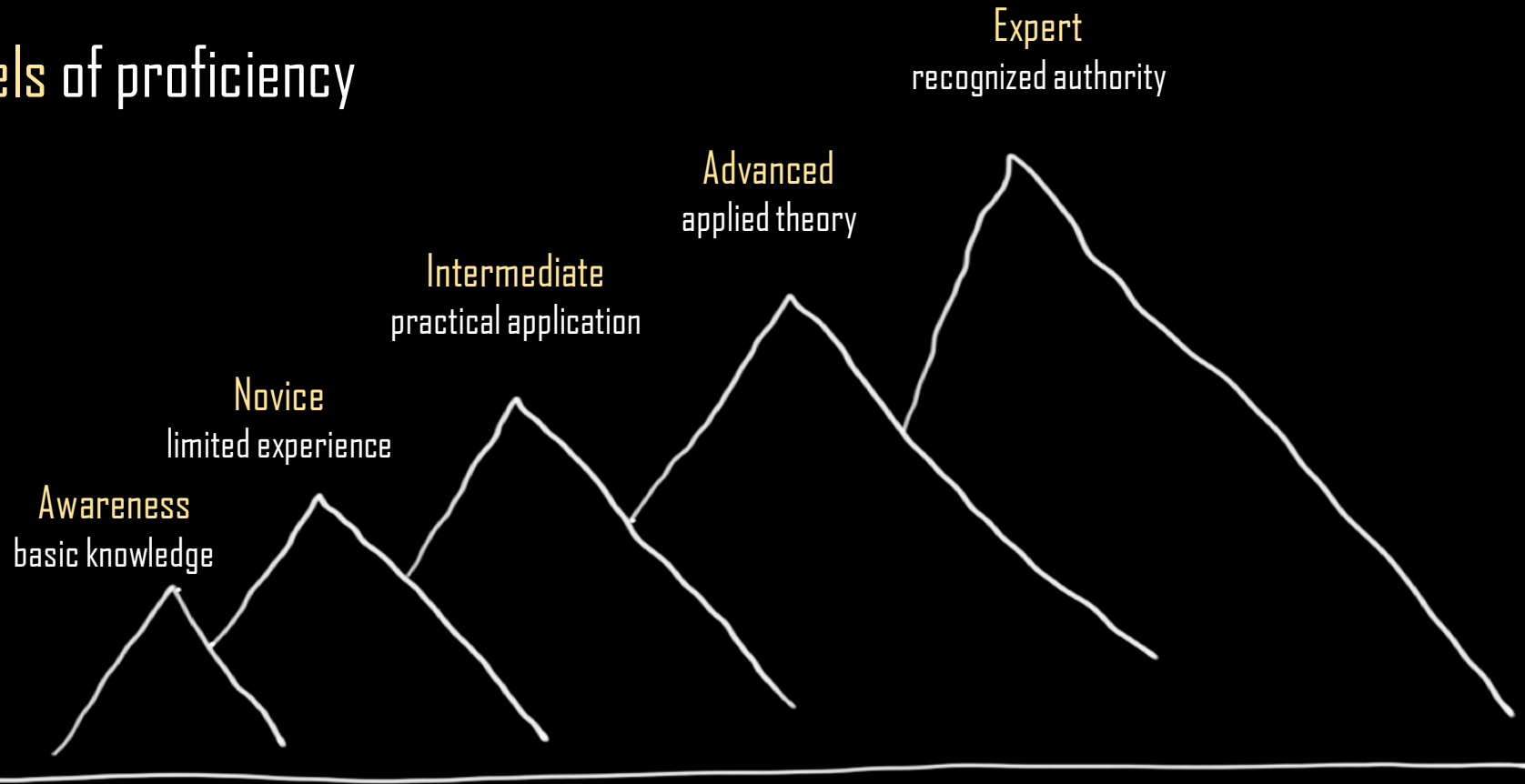
If key = 2, 2 sec. to decrypt

If key = 3, 3 sec. to decrypt

...

**Broke RSA in 2003**

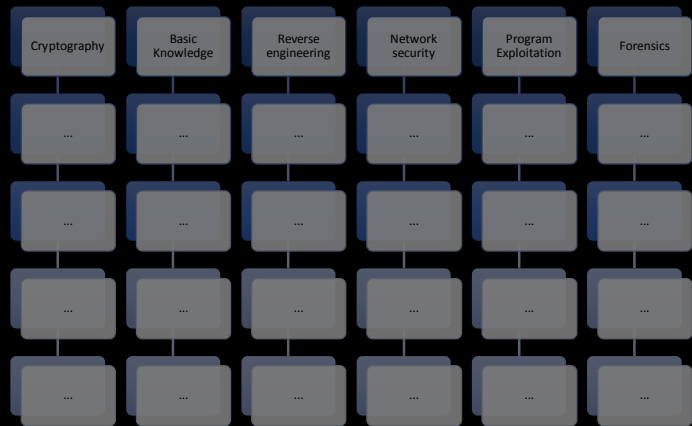
# Levels of proficiency





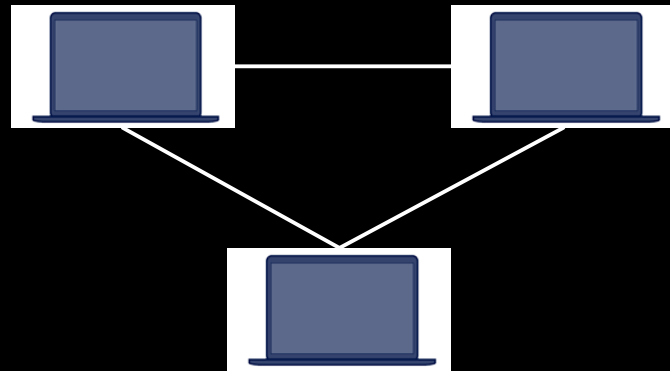
# Jeopardy-Style CTF

## Awareness - Intermediate



# Attack-Defense CTF

## Advanced - Expert



- Everyone runs same software
- Exploit others = gain points
- Be exploited = loose points



# CMU Goals

1

Grow cybersecurity field

2

Identify and attract most promising high school students

3

Systematize the above

## HIGH SCHOOL HACKING COMPETITION

APRIL 26TH 2013 – MAY 6TH 2013

### TOASTED YOGURT

When a robot from space crash lands in your backyard it's up to your hacking skills to fix him and uncover the secret he carries...

The Adventure Will Run: April 26th 2013 - May 6th 2013 [Registration](#)

Both teams are student-run and based at Canning-Mellon University.

051919

Through

**Iriscecca** @iriscecca2017 · 1 day

I just after that you get to apply  
harding skills in a fun, but  
challenging and legal  
environment. Thank you @jace-01  
for the opportunity!

13 Retweeted by jace-01

Thank you @davidm

Year 3: ~18,000

Level	Challenge	Team Completions	Acquired Skills
1	First Contact	1,368	Network Traffic Analysis
2	CFG to C	1,321	x86 Assembly and Control-Flow Graphs
2	Try Them All!	1,279	Password Hashes, Salts, and Dictionary Attacks
3	DDoS Detection	615	Defensive Traffic Analysis
3	Byte Code	1,146	Program Representation
3	SQL Injection	571	Command Injection Attacks
3	RSA	228	RSA Implementation
4	Overflow 1	216	Buffer Overflow
4	ROP 1	96	Return-to-libc Attack

18,000  
High School Students

1. Run PicoCTF.com

2. Top 50  
get recommendation

3. Coursework + CTF

Run next picoctf

Bell-curve of ability



## System recruits

1

Auto-didactic

2

Demonstrable ability

3

Top talent

# Two Themes

1. CTF problems are a proven, effective way to teach hacking skills
2. You can systematize CTF's to build your pipeline





## Next Actions

- Incorporate CTF's into your training



## Next Actions

- Incorporate CTF's into your training
- Develop system for identifying talent
  - Build CTF problems representative of skills you care about
  - Use CTF applications to recruit and/or interview

**RSA**®Conference2018



#RSAC

**THANK YOU**