

# CS 155: Real-World Security

April 14, 2016

Alex Stamos  
CSO, Facebook

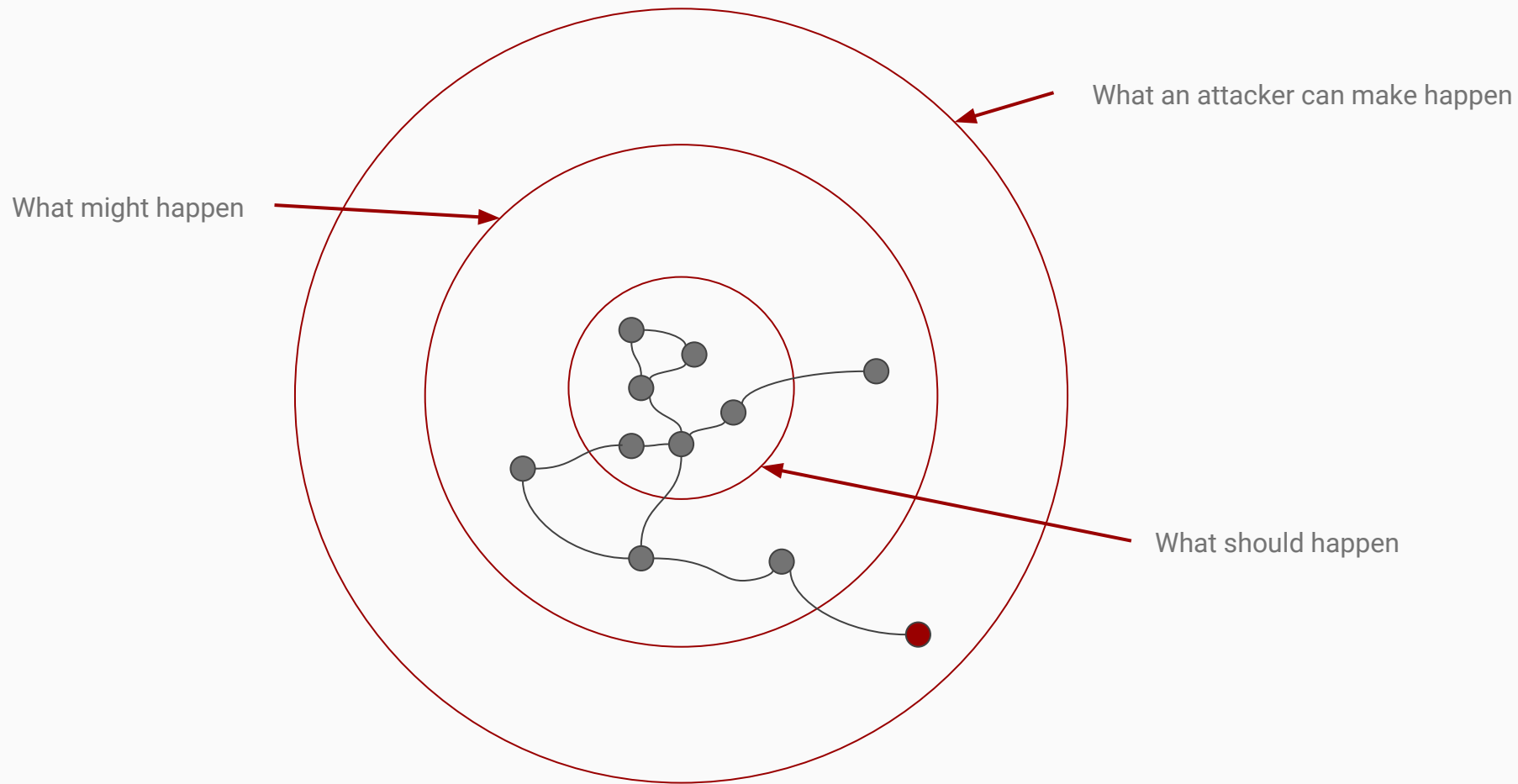


Why are you here?

# Agenda

- How bugs are found
- How defense works in the real world
- Real bugs
- Real impacts
- Interesting problems for you to solve
- Five basic tips for career success

How are bugs found?



# Vulnerability Discovery is the art of...

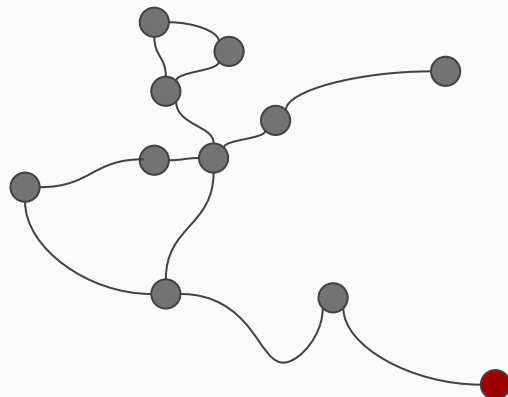
- Pushing software into exploitable states
- Predicting the kinds of mistakes engineers will make and QA/security teams will miss
- Making the impossible possible

# Fuzzing

Using automation to mutate input into a system and look for exploitable states

Enhanced by:

- Intelligently unpacking, mutating, and re-packing formats
- Instrumenting the binary to accelerate input and look for caught exceptions
- Studying control-flow and intentionally hitting corner cases



# Fuzzing

## american fuzzy lop 1.74b (readelf)

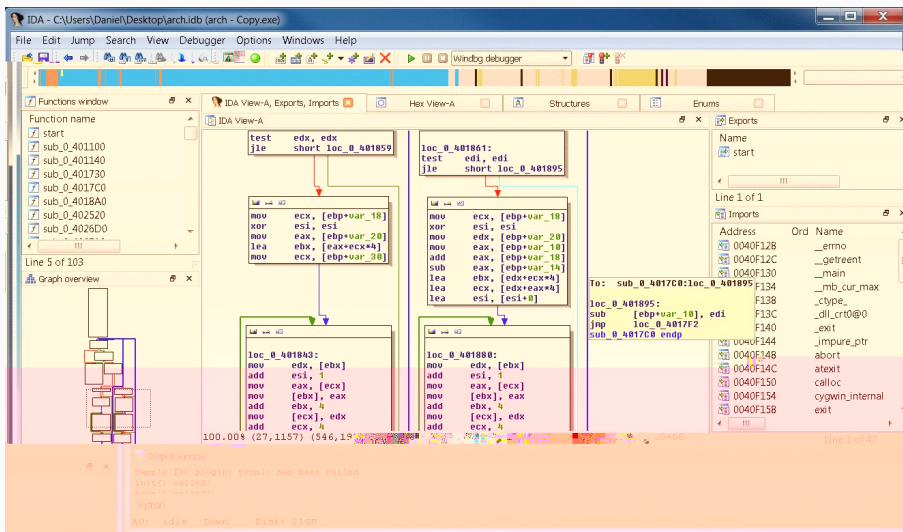
process timing		overall results	
run time : 0 days, 0 hrs, 8 min, 24 sec		cycles done : 0	
last new path : 0 days, 0 hrs, 1 min, 59 sec		total paths : 812	
last uniq crash : 0 days, 0 hrs, 3 min, 17 sec		uniq crashes : 8	
last uniq hang : 0 days, 0 hrs, 3 min, 23 sec		uniq hangs : 10	
cycle progress		map coverage	
now processing : 0 (0.00%)		map density : 3158 (4/82%)	
paths timed out : 0 (0.00%)		count coverage : 2.56 bits/tuple	
stage progress		findings in depth	
now trying : arith 8/8		favored paths : 1 (0.12%)	
stage execs : 295k/326k (90.31%)		new edges on : 318 (39.16%)	
total execs : 552k		total crashes : 63 (8 unique)	
exec speed : 1114/sec		total hangs : 191 (10 unique)	
fuzzing strategy yields		path geometry	
bit flips : 447/75.5k, 59/75.5k, 59/75.5k		levels : 2	
byte flips : 7/9436, 0/5858, 6/5950		pending : 812	
arithmetics : 0/0, 0/0, 0/0		pend fav : 1	
known ints : 0/0, 0/0, 0/0		own finds : 811	
dictionary : 0/0, 0/0, 0/0		imported : n/a	
havoc : 0/0, 0/0		variable : 0	
trim : 0.00%/1166, 38.39%			
[cpu: 15%]			



# Reverse Engineering

Reverse engineering allows the researcher to:

- Find exploitable states and work backward
- Look for common antipatterns
- Understand and bypass sanity checks and protections

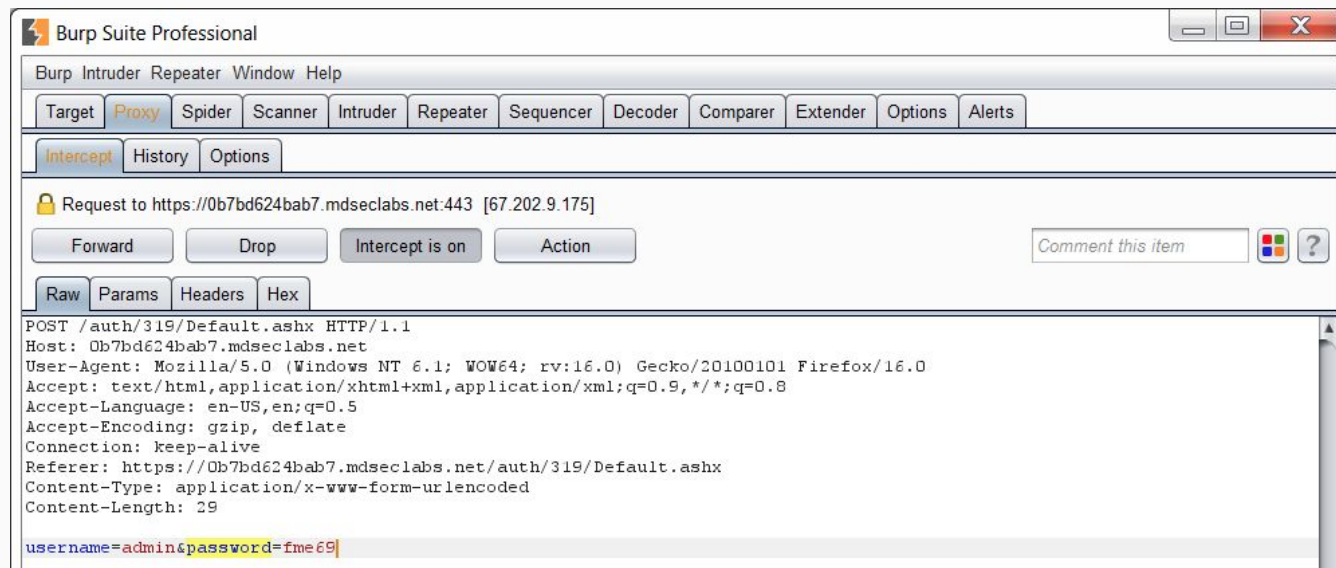


Includes:

- Debugging
- Disassembly
- Binary diffing
- Decompilation

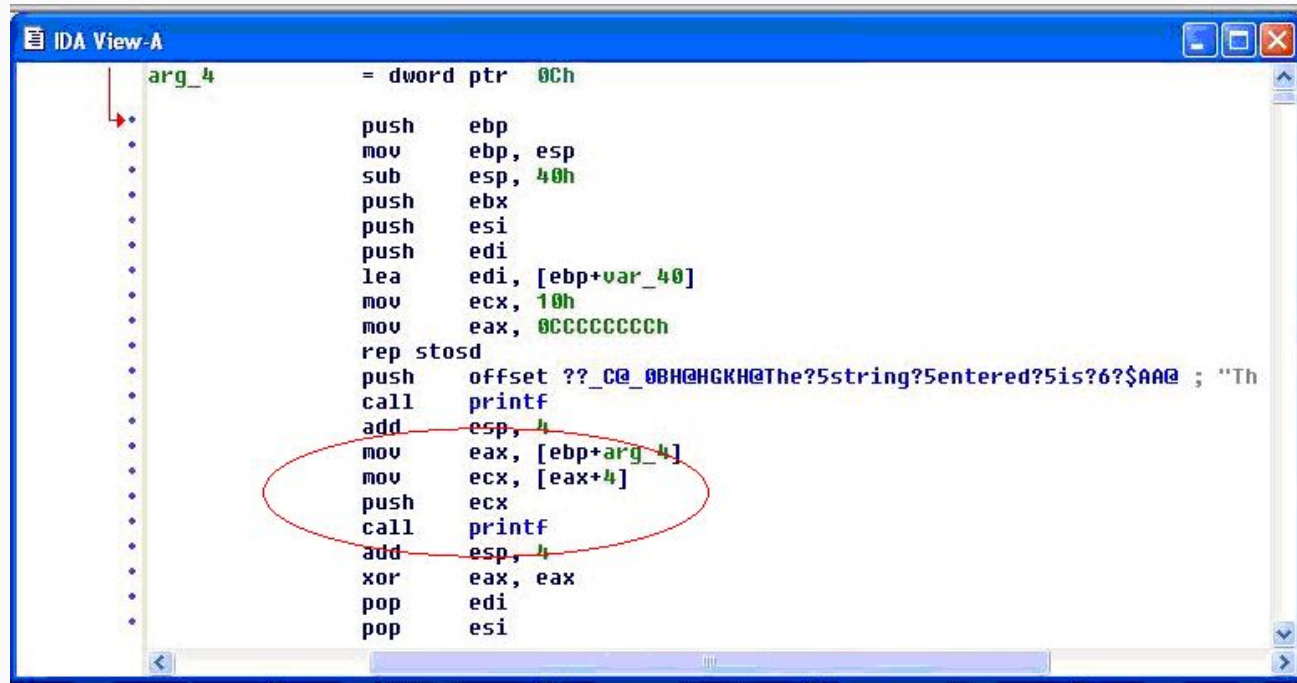
# Manual Manipulation

- Many interesting flaws boil down to asking the software to do something
- Due to:
  - Confused deputy problems
  - Missing access control checks
  - Lack of data consistency checks
- Often using tools to intercept and manipulate inputs



# Pulling it Together

1. Disassemble a binary to discover:



```
IDA View-A
arg_4 = dword ptr 0Ch

push ebp
mov ebp, esp
sub esp, 40h
push ebx
push esi
push edi
lea edi, [ebp+var_40]
mov ecx, 10h
mov eax, 0CCCCCCCCh
rep stosd
push offset ??_C@_0BH@HGKH@The?5string?5entered?5is?6?$AA@ ; "Th
call printf
add esp, 4
mov eax, [ebp+arg_4]
mov ecx, [eax+4]
push ecx
call printf
add esp, 4
xor eax, eax
pop edi
pop esi
```

# Pulling it Together

2. Use format-aware fuzzing to try to find entry points that lead to format string

```
[lcamtuf@raccoon afl]$ ./afl-analyze -e -i testcases/images/png/not_kitty.png ~/readpng
afl-analyze 2.00b by <lcamtuf@google.com>

[+] Read 218 bytes from 'testcases/images/png/not_kitty.png'.
[*] Performing dry run (mem limit = 25 MB, timeout = 1000 ms, edges only)...
[*] Analyzing input file (this may take a while)...

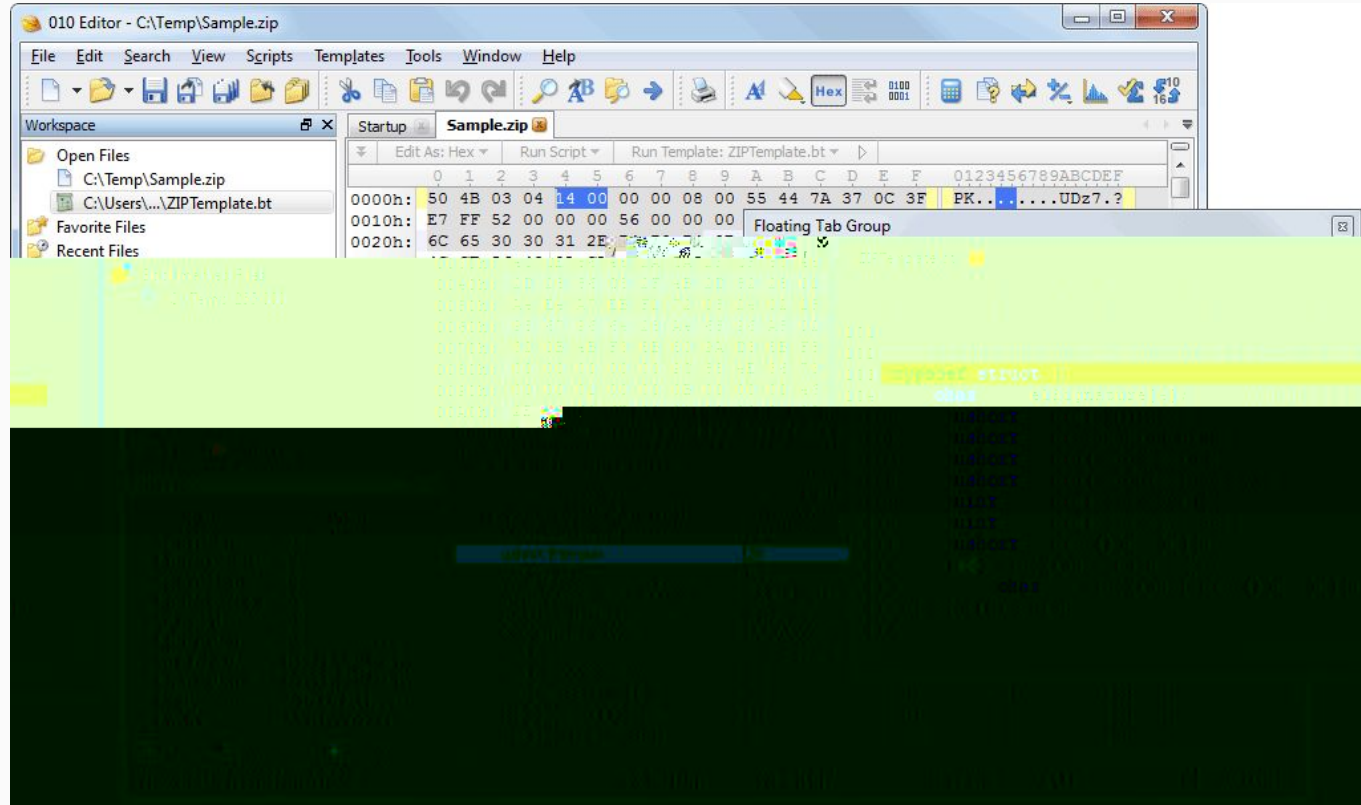
01 - no-op block
01 - superficial content
01 - critical stream
01 - "magic value" section
01 - suspected length field
01 - suspected cksum or magic int
01 - suspected checksummed block

[000000] #89 P N G #0d #0a #1a #0a #00 #00 #00 #0d I H D R
[000016] #00 #00 #00 #20 #00 #00 #00 #20 #08 #03 #00 #00 #00 D #a4 #8a >
[000032] #c6 #00 #00 #00 #19 t E X t S o f t w a r e >
[000048] e #00 A d o b e #20 I m a g e R e a >
[000064] d y q #c9 e < #00 #00 #00 #0f P L T E f #cc >
[000080] #cc #ff #ff #ff #00 #00 #00 3 #99 f #99 #ff #cc > L #af >
```

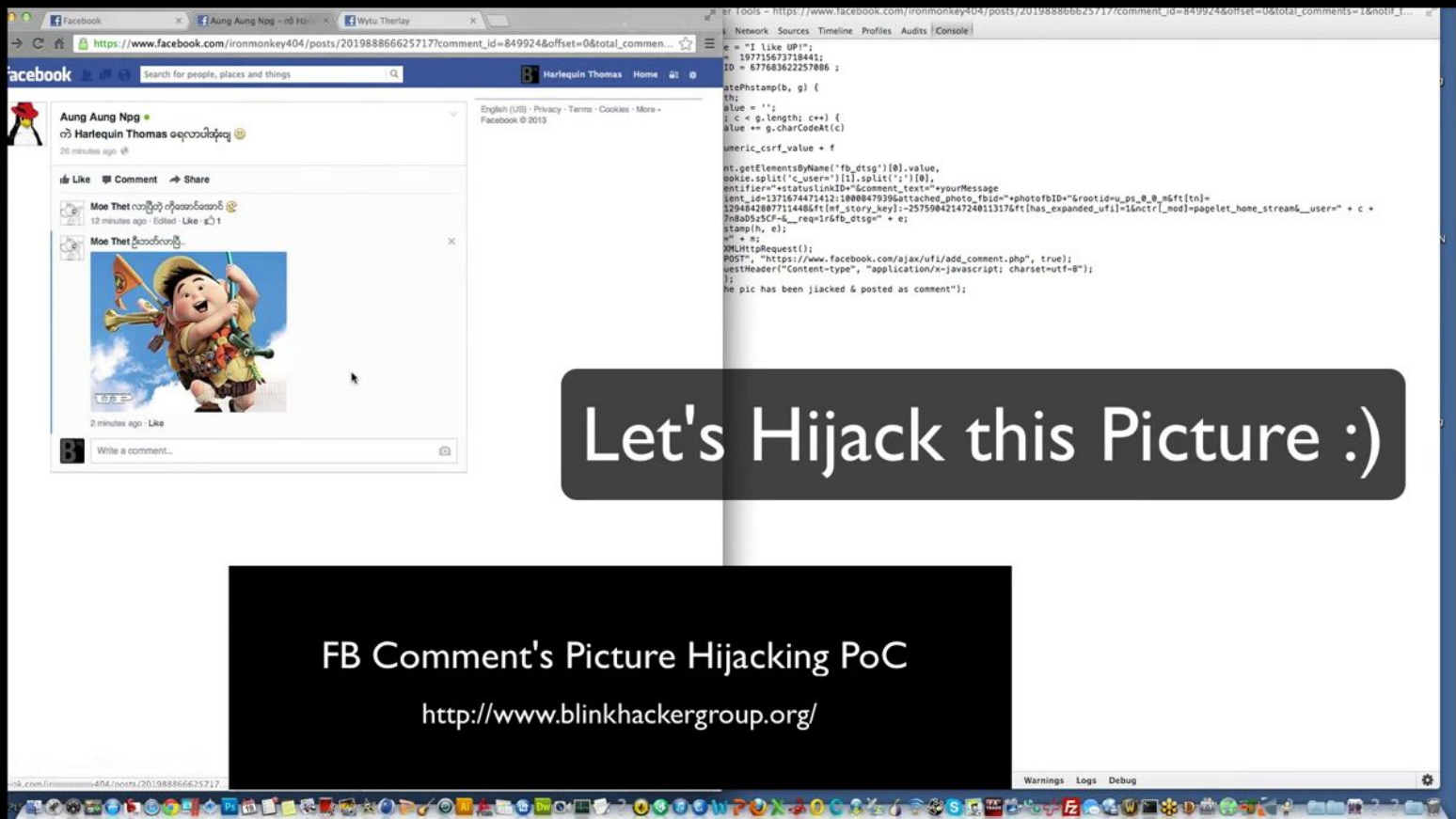
<https://lcamtuf.blogspot.com/2016/02/say-hello-to-afl-analyze.html>

# Pulling it Together

3. Researcher carefully modifies crash-creating documents by the fuzzer to obtain execution



# Real World Bugs



Let's Hijack this Picture :)

FB Comment's Picture Hijacking PoC

<http://www.blinkhackergroup.org/>

# Facebook Picture Sharing on Comment Exploit

# Apple's TLS Code

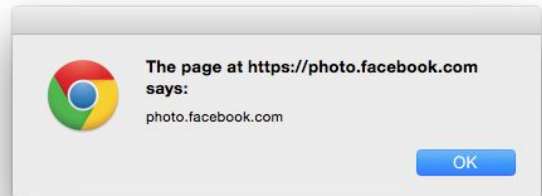
```
hashOut.data = hashes + SSL_MD5_DIGEST_LEN;
hashOut.length = SSL_SHA1_DIGEST_LEN;
if ((err = SSLFreeBuffer(&hashCtx)) != 0)
    goto fail;
if ((err = ReadyHash(&SSLHashSHA1, &hashCtx)) != 0)
    goto fail;
if ((err = SSLHashSHA1.update(&hashCtx, &clientRandom)) != 0)
    goto fail;
if ((err = SSLHashSHA1.update(&hashCtx, &serverRandom)) != 0)
    goto fail;
if ((err = SSLHashSHA1.update(&hashCtx, &signedParams)) != 0)
    goto fail;
goto fail;
if ((err = SSLHashSHA1.final(&hashCtx, &hashOut)) != 0)
    goto fail;
err = sslRawVerify(...);
```



# Embedding Script in Images

```
finl1te@mbp /tmp » hexdump -C xss-fnt-pe-png.png
```

```
00000000  89 50 4e 47 0d 0a 1a 0a 00 00 00 0d 49 48 44 52 |.PNG.....IHDR|
00000010  00 00 00 20 00 00 00 20 08 02 00 00 00 fc 18 ed |... ..|
00000020  a3 00 00 00 09 70 48 59 73 00 00 0e c4 00 00 0e |.....pHYs.....|
00000030  c4 01 95 2b 0e 1b 00 00 00 65 49 44 41 54 48 80 |... ..|
00000040  63 ac ff 3c 53 43 52 49 50 54 20 53 52 43 3d 2 |... ..|
00000050  2f 46 4e 54 2e 50 45 3e 3c 2f 73 63 72 69 70 7 |... ..|
00000060  3e c3 ea c0 46 8d 17 f3 af de 3d 73 d3 fd 15 c |... ..|
00000070  43 2f 0f b5 ab a7 af ca 7e 7d 2d ea e2 90 22 a |... ..|
00000080  73 85 45 60 7a 90 d1 8c 3f 0c a3 60 14 8c 82 5 |... ..|
00000090  30 0a 46 c1 28 18 05 a3 60 14 8c 82 61 00 00 7 |... ..|
000000a0  32 1c 02 78 65 1f 48 00 00 00 00 49 45 4e 44 a |... ..|
000000b0  42 60 82
```



# Bug or feature?

## FFmpeg Protocols Documentation

### 3.4 concat

#### Table of Contents

- 1 Description
- 2 Protocol Options
- 3 Protocols
  - 3.1 async
  - 3.2 bluray
  - 3.3 cache
  - 3.4 concat
  - 3.5 crypto
  - 3.6 data
  - 3.7 file
  - 3.8 ftp
  - 3.9 gopher
  - 3.10 hls
  - 3.11 http
    - 3.11.1 HTTP
  - 3.12 Icecast
  - 3.13 mmst
  - 3.14 mmsh
  - 3.15 md5
  - 3.16 pipe

Physical concatenation protocol.

Read and seek from many resources in sequence as if they were a unique resource.

A URL accepted by this protocol has the syntax:

```
concat:URL1|URL2|...|URLN
```

where *URL1*, *URL2*, ..., *URLN* are the urls of the resource to be concatenated, each one possibly specifying a distinct protocol.

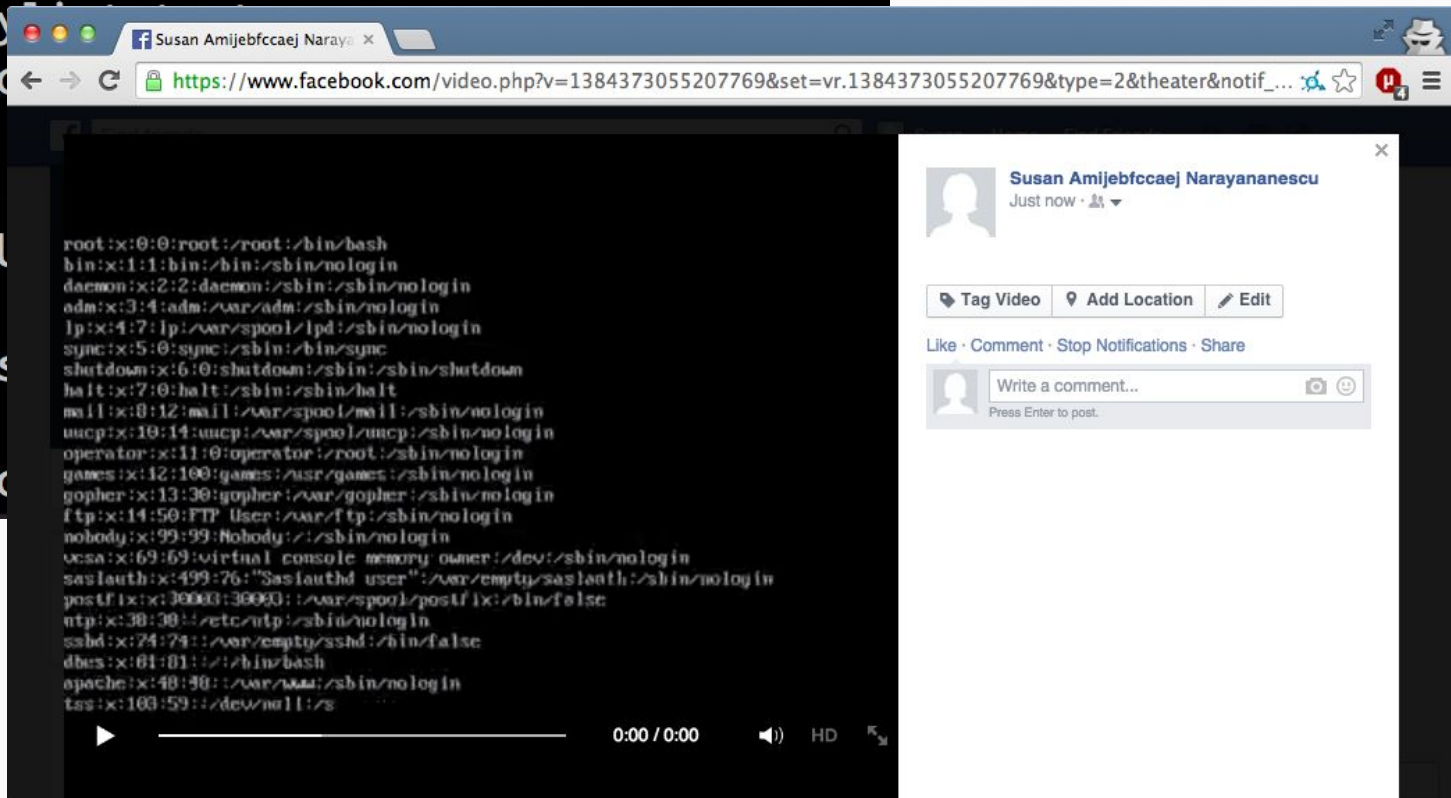
For example to read a sequence of files `split1.mpeg`, `split2.mpeg`, `split3.mpeg` with `ffplay` use the command:

```
ffplay concat:split1.mpeg\|split2.mpeg\|split3.mpeg
```

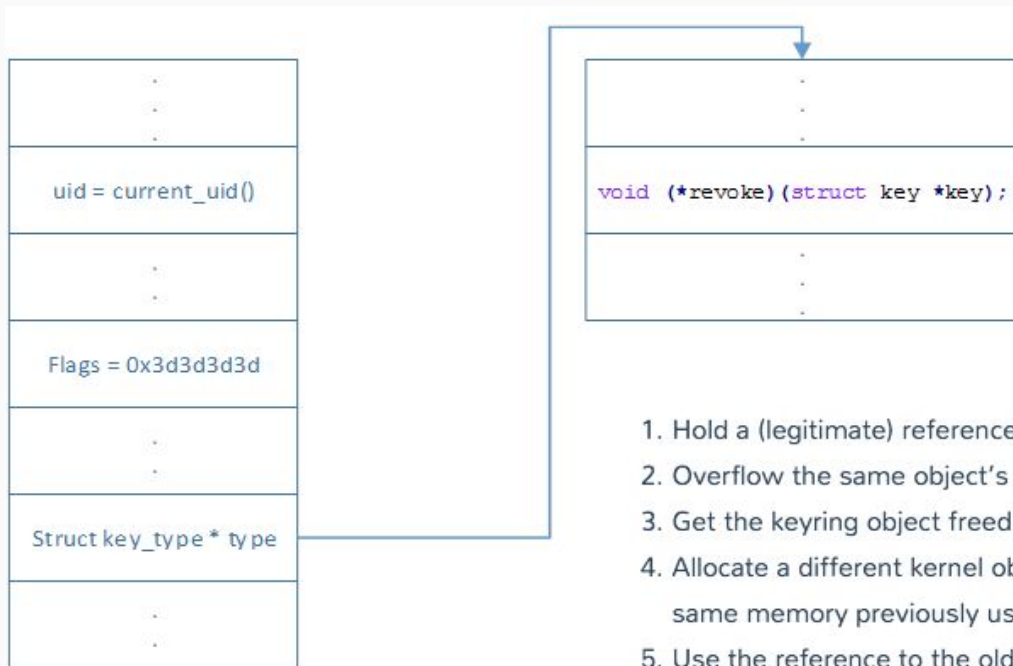
Note that you may need to escape the character "|" which is special for many shells.

# Bug or feature?

```
alexstamos-mbp:Downloads alexstamos$ file hax.mp4
hax.mp4: M3U playli...
alexstamos-mbp:Downloads alexstamos$ cat hax.mp4
#EXTM3U
#EXT-X-VERSION:4
#EXT-X-MEDIA-SEQUENCE:0
#EXTINF:10,
concat://etc/passwd
#EXT-X-ENDLIST
alexstamos-mbp:Downloads alexstamos$
```



# Memory Management



1. Hold a (legitimate) reference to a key object
2. Overflow the same object's *usage*
3. Get the keyring object freed
4. Allocate a different kernel object from user-space, with a user-controlled content, over the same memory previously used by the freed keyring object
5. Use the reference to the old key object and trigger code execution

Who Finds Bugs?

# Who Looks for Bugs?



- Have benefit of source code, access to engineers
- Target 100% coverage, so broad-and-shallow testing is common
- Generally need automation to assist



- Have less information, not a huge problem with shipped code
- Only need a handful of flaws to chain them together
- Need to find and explore issues without alerting defenders



- Various motivations. Money? Fame?
- Lots of ethical reporting options via bug bounties
- Generally want to stay on right side of the law

# Real World Defense

# Real World Defense Should Focus on...

... (for securing people)

Three biggest problems for most people:

1. Compromised reused passwords
2. Phishing credentials
3. Common, n-day malware

... (for securing enterprises)

Namely, capabilities, tools, techniques and procedures for the intrusion kill chain:

1. Reconnaissance
2. Weaponization
3. Delivery
4. Exploitation
5. Installation
6. Command and Control
7. Actions on Objectives



# Security research often misses the point

The incentives for private and academic research point the wrong way:



## The World's Address: An App That's Worn

**Anatomization and Protection of Mobile Apps' Location Privacy Threats** .....  
Kassem Fawaz, Huan Feng, and Kang G. Shin, *University of Michigan*

**LinkDroid: Reducing Unregulated Aggregation of App Usage Behaviors** .....  
Huan Feng, Kassem Fawaz, and Kang G. Shin, *University of Michigan*

**PowerSpy: Location Tracking using Mobile Device Power Analysis** .....  
Yan Michalevsky, Aaron Schulman, Gunaa Arumugam Veerapandian, and Dan Boneh, *Gabi Nakib, National Research and Simulation Center/Rafael Ltd.*

## ADDioS!

**In the Compression Hornet's Nest: A Security Study of Data Compression in Network** .....  
Giancarlo Pellegrino, *Saarland University*; Davide Balzarotti, *Eurecom*; Stefan Winter, *Technische Universität Darmstadt*

**Bohatei: Flexible and Elastic DDoS Defense** .....  
Seyed K. Fayaz, Yoshiaki Tobioka, and Vyas Sekar, *Carnegie Mellon University*; Michael J. B. S. *University of Illinois at Urbana-Champaign*

**Boxed Out: Blocking Cellular Interconnect Bypass Fraud at the Network Edge** .....  
Bradley Reaves, *University of Florida*; Ethan Sherman, *Georgia Institute of Technology*; Henry Carter, *Georgia Institute of Technology*; Patrick Traynor, *University of Florida*

## Attacks: I Won't Let You Down

**GSMem: Data Exfiltration from Air-Gapped Computers over GSM Frequencies** ..  
Mordechai Guri, Assaf Kachlon, Ofer Hasson, Gabi Kedma, Yisroel Mirsky, and Yuval Elovitz, *University of the Negev*

**Thermal Covert Channels on Multi-core Platforms** .....  
Ramya Jayaram Masti, Devendra Rai, Aanjan Ranganathan, Christian Müller, Lothar Thiele, and Srđjan Capkun, *ETH Zürich*

**Rocking Drones with Intentional Sound Noise on Gyroscopic Sensors** .....  
Yunmok Son, Hocheol Shin, Dongkwan Kim, Youngseok Park, Juhwan Noh, Kibum Chon, and Yongdae Kim, *Korea Advanced Institute of Science and Technology (KAIST)*

## How Do You Secure a Cloud and Pin it Down?

**Cache Template Attacks: Automating Attacks on Inclusive Last-Level Caches** ....  
Daniel Gruss, Raphael Spreitzer, and Stefan Mangard, *Graz University of Technology*

**A Placement Vulnerability Study in Multi-Tenant Public Clouds** .....  
Venkatanathan Varadarajan, *University of Wisconsin—Madison*; Yinqian Zhang, *The Ohio State University*; Thomas Ristenpart, *Cornell Tech*; Michael Swift, *University of Wisconsin—Madison*

**A Measurement Study on Co-residence Threat inside the Cloud** .....  
Zhang Xu, *College of William and Mary*; Haining Wang, *University of Delaware*; Zhengyu Wang, *NEC Laboratories America*

# What causes the most problems for normal users?



The screenshot shows a web browser window with the Facebook logo and a "Sign Up" button. The address bar displays a URL from Dropboxusercontent.com. The main content area is titled "Facebook Security" and contains a red-bordered box labeled "Account Recovery Center". Inside this box, a message states that the account has been reported and will be closed permanently, listing reasons such as using fake names, identity fraud, inappropriate content, and pornographic photos. Below the warning box, there are input fields for "Email or Phone:", "Password:", and "Birthday:" (with dropdowns for Day, Month, and Year). A checkbox for "Keep me logged in" and a "Log In" button are also present. At the bottom, a language selection bar includes options like English (US), Español, Português (Brasil), Français (France), Deutsch, Italiano, العربية, हिन्दी, and 中文(简体).

Facebook

https://dl.dropboxusercontent.com/u/316431807/system\_recovery/account

facebook Sign Up

### Facebook Security

#### Account Recovery Center

Your account has been reported by other users, and your account will be closed permanently, with reasons in Facebook :

- ✓ Using fake names
- ✓ Identity fraud on your account
- ✓ You write content that is not fun (rough).
- ✓ Using pornographic photos and photos of other users.

Please restore your account, to prevent account disable, thank you.

Email or Phone:

Password:

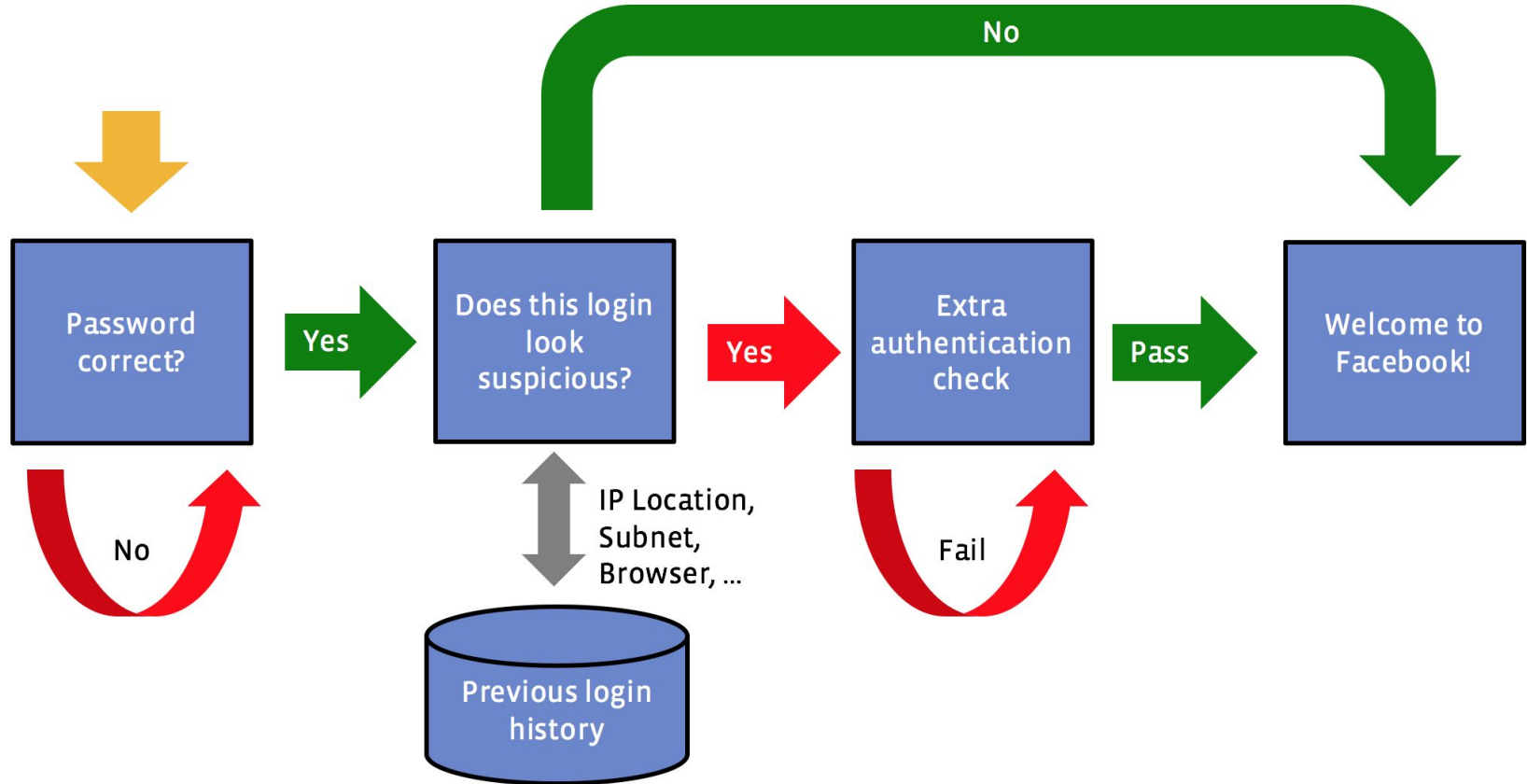
Birthday:  Day  Month  Year

☐ Keep me logged in

Log In

English (US) Español Português (Brasil) Français (France) Deutsch Italiano العربية हिन्दी 中文(简体)

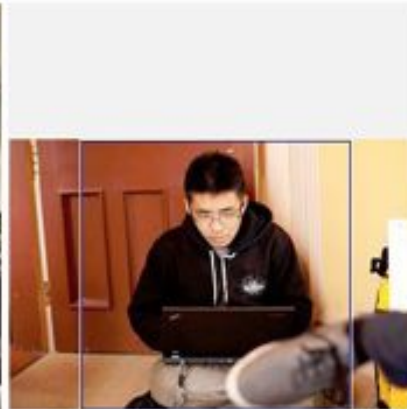
# Real World Defense - Logins



# Supplemental Authentication

Please confirm your identity

Challenge 1 of 5



This appears to be:

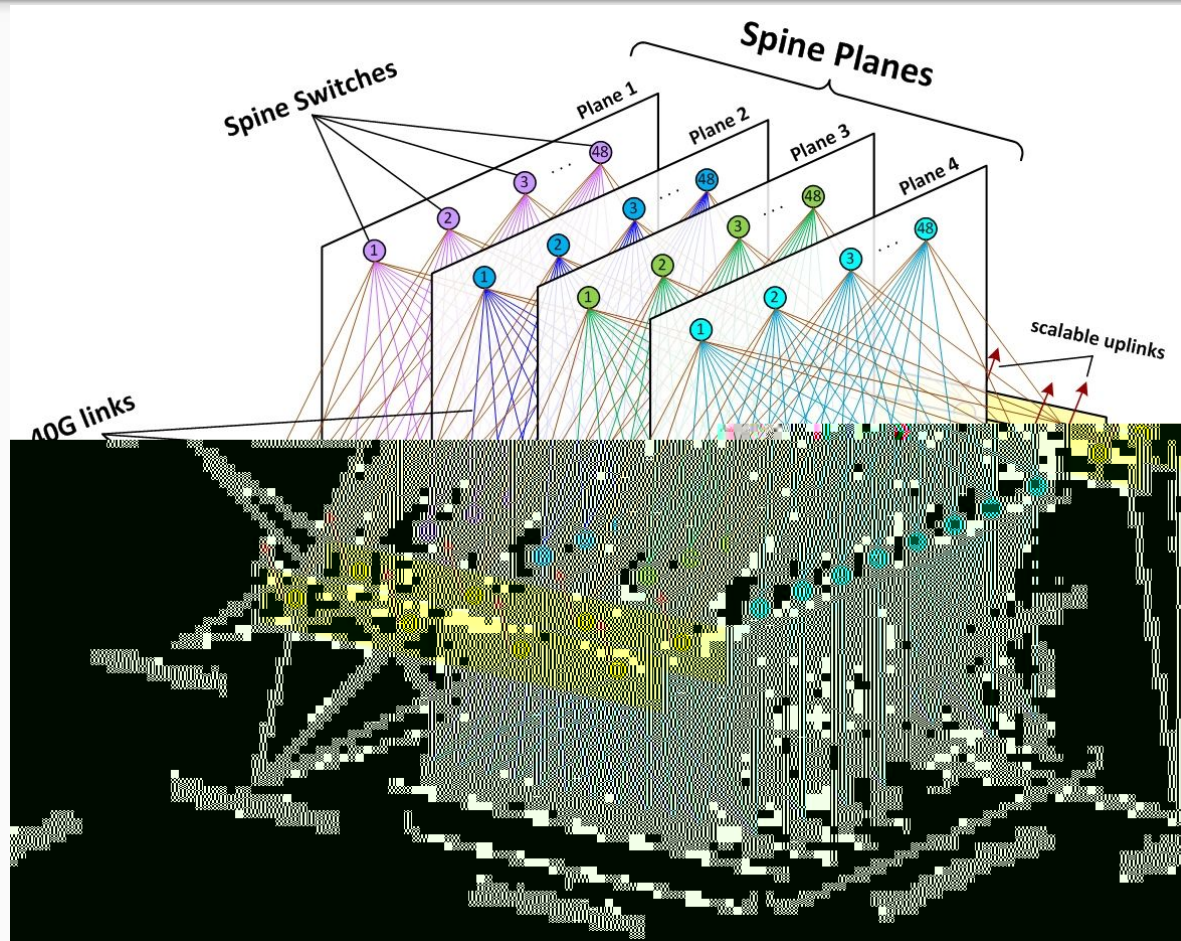
- ☐ Ej Raymond ☐ Edward Speyer ☐ Neville Bowers  
☒ Zejia Chen ☐ Paul Saab ☐ Chad Greene

Submit

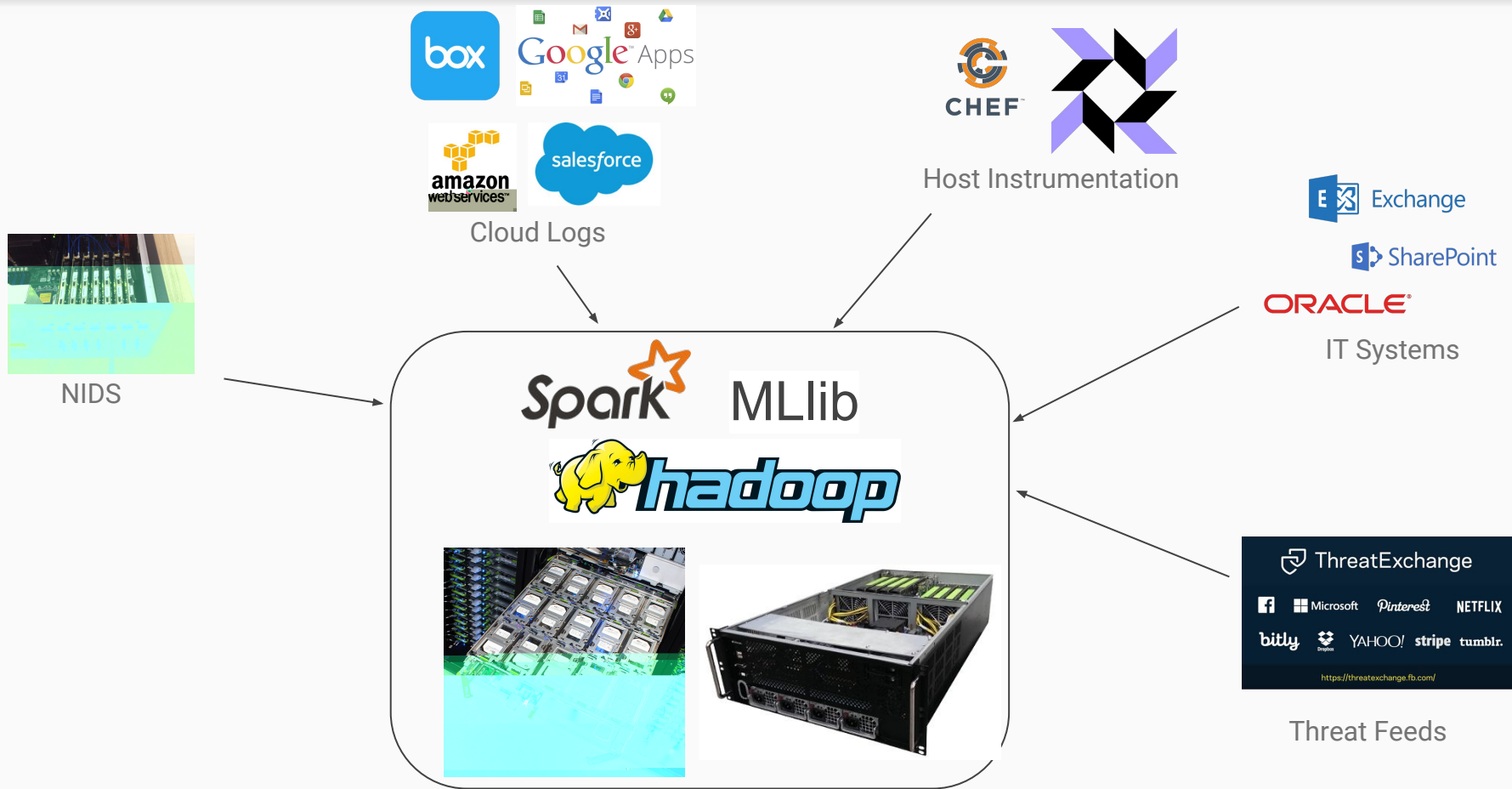
[Confirm identity another way](#)

# Open Real World Problems

# Network Security at Scale



# Dumb sensors, smart (delayed) decisions



# Careers in Security



# What impact do you want to have on the world?

InfoSec might be the most impactful engineering discipline of the 21st century.

You can choose to:

- Protect those who cannot protect themselves
- Bring voice to those who have never had it
- Secure the technologies that billions depend upon
- Stop those who wish to use technology to control and oppress millions

Participating in this industry makes you a moral actor.

Shape your career around your ethical choices, not vice versa.

# Six Tips for a Successful Career

1. Always put yourself in a position to learn and grow. Comfort == decay
2. Be part of the product, not the plumbing
3. Your point of maximum leverage comes right after you get a job offer
4. Understand the Cap Table for any private company
5. Always go into a meeting knowing what you want the outcome to be
6. It's a small industry. Be nice

Thank you and good luck!

[alex@stamos.org](mailto:alex@stamos.org)