Lecture 4: Threats

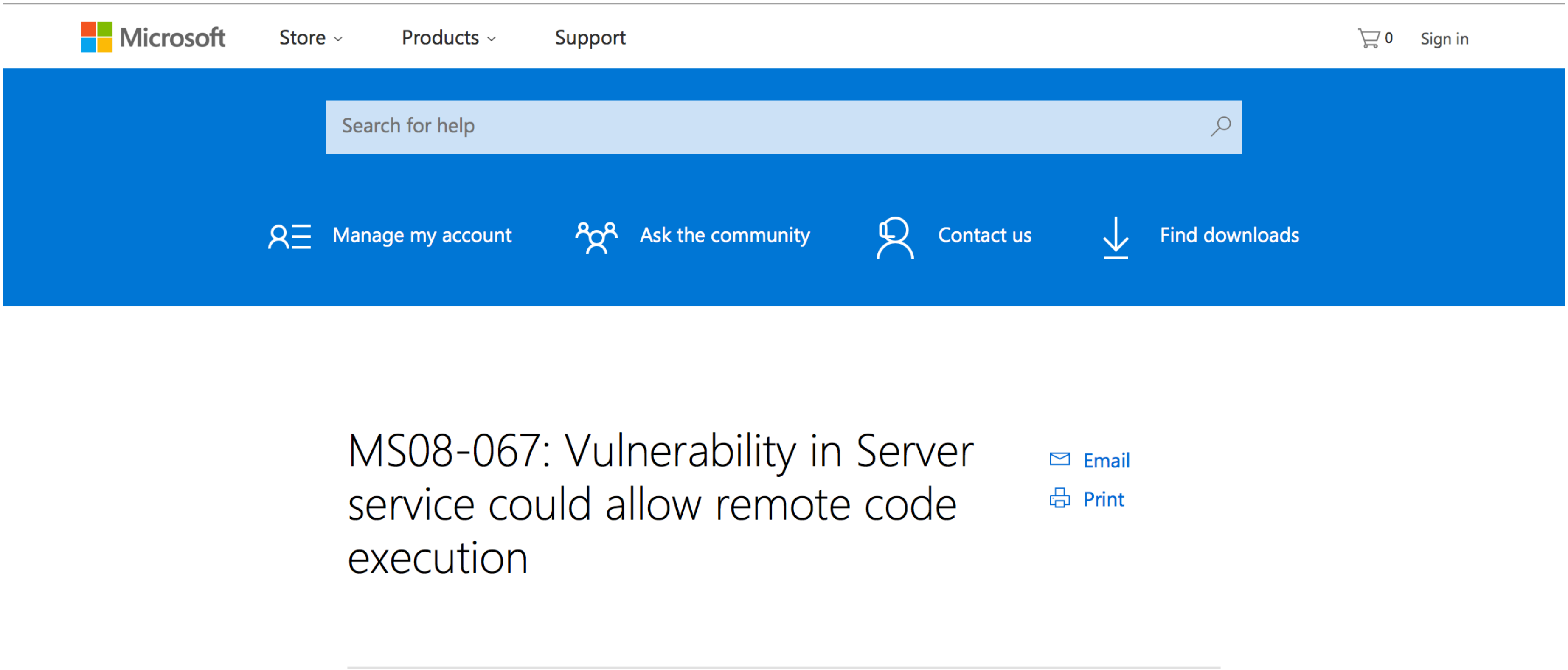
CS 5430

2/5/2018

The Big Picture

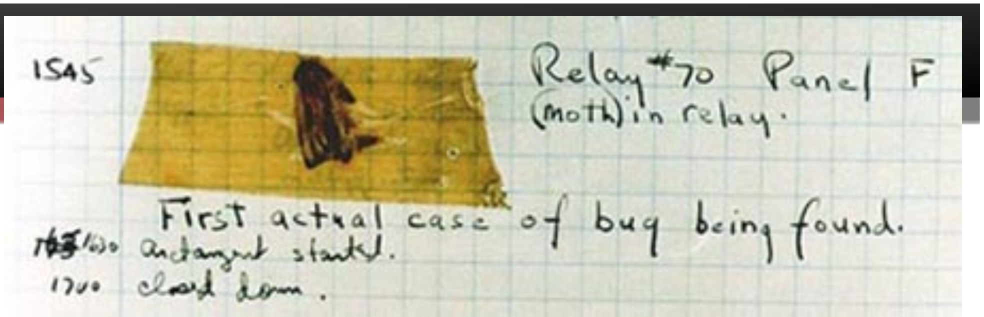
Attacks
are perpetrated by
threats
that inflict
harm
by exploiting
vulnerabilities
which are controlled by
countermeasures.

Once Upon a Time…



MS08-067: Vulnerability in Server service could allow remote code execution

Bugs



1545 Relay #70 Panel F (moth) in relay.

First actual case of bug being found.

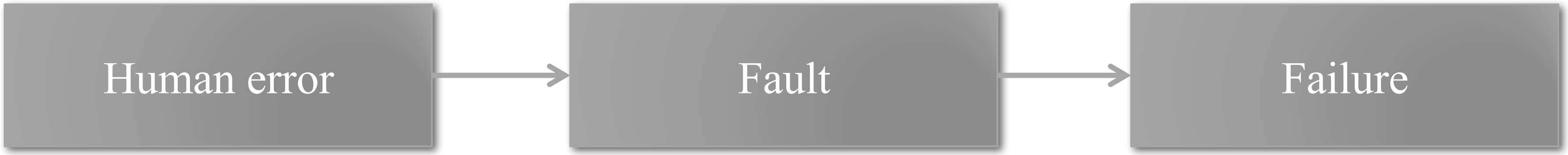
1630 Antangent started.

1700 closed down.

"bug": suggests something just wandered in

[IEEE 729]

- Fault: result of human error in software system
  - E.g., implementation doesn't match design, or design doesn't match requirements
  - Might never appear to end user
- Failure: violation of requirement
  - Something goes wrong for end user



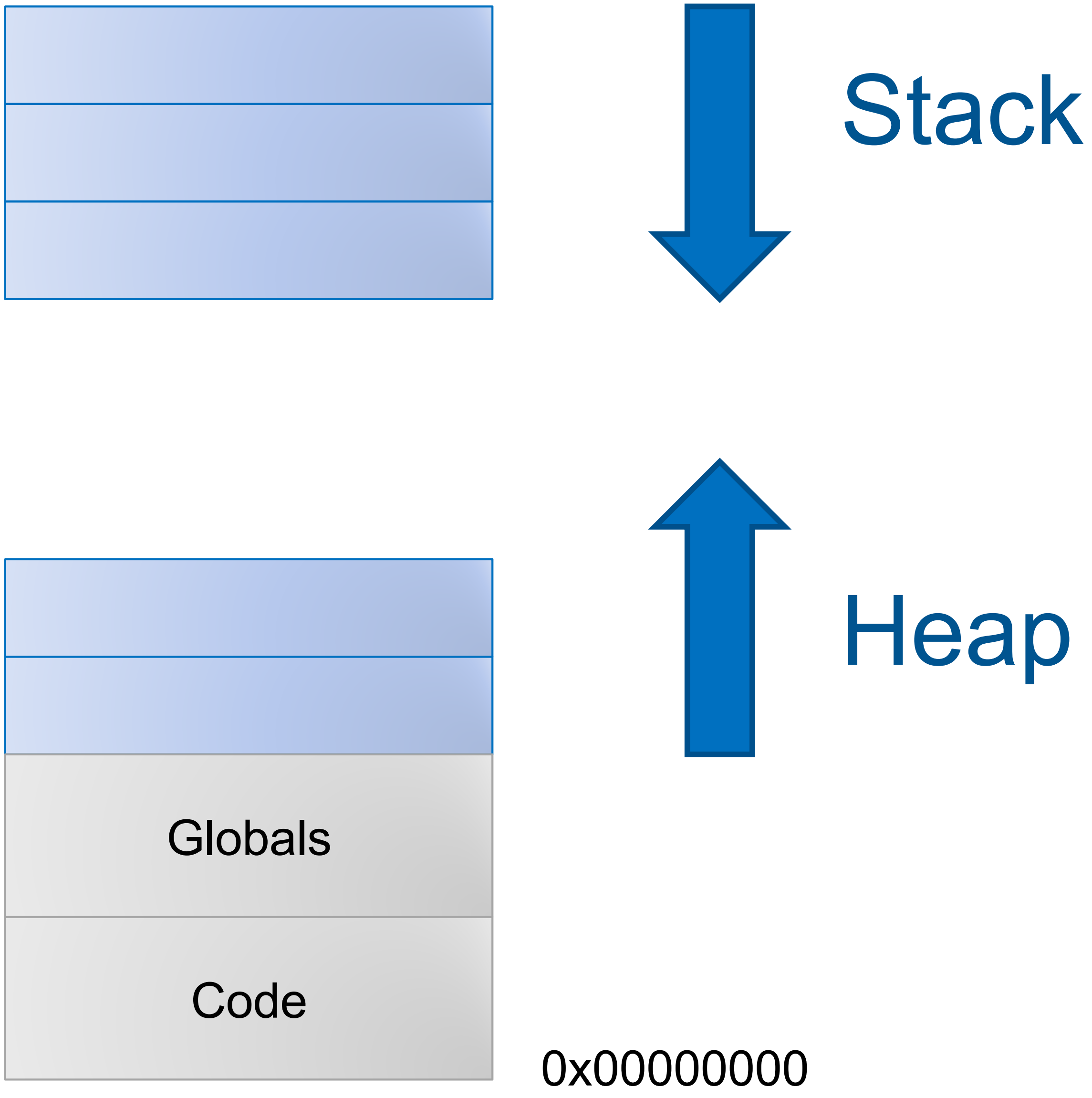
Human error → Fault → Failure

Vulnerability

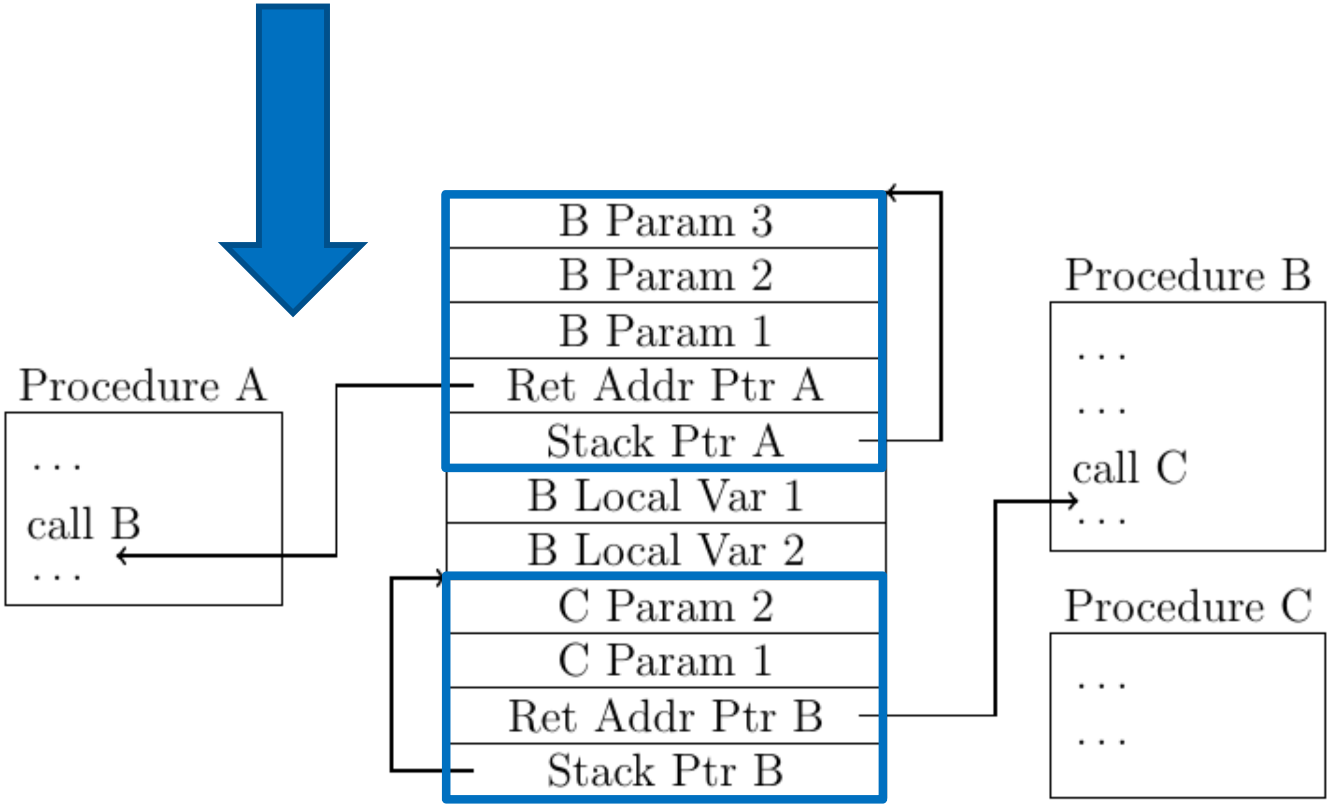
An unintended aspect of a system (design, implementation, or configuration) that can cause the system to do something it shouldn't, or fail to do something it should

- E.g., buffer overflows, code injection, cross-site scripting, missing authentication or access control, misconfiguration
- National databases: CVE, NVD
- Ignoring vulnerabilities is risky
  - Too often: "no one would/could ever exploit that"
  - *Weakest link* phenomenon
  - Timing, failure modes, message delivery, input format, etc.

Memory: A Quick Review



The Stack



Procedure A

. . .

call B

. . .

B Param 3

B Param 2

B Param 1

Ret Addr Ptr A

Stack Ptr A

B Local Var 1

B Local Var 2

C Param 2

C Param 1

Ret Addr Ptr B

Stack Ptr B

Procedure B

. . .

. . .

call C

. . .

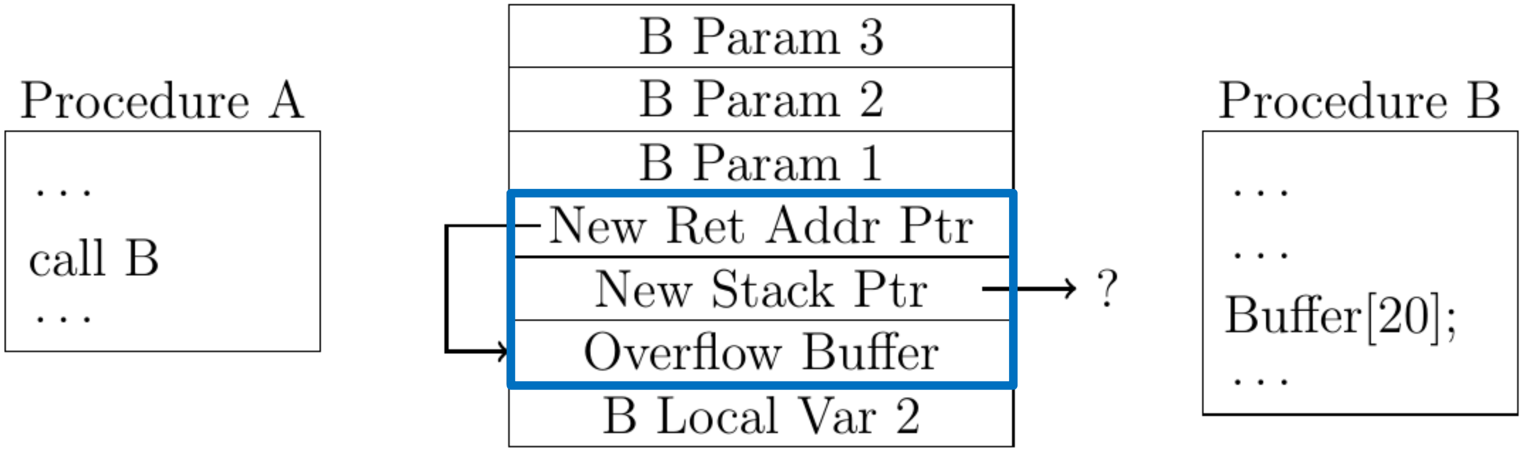
Procedure C

. . .

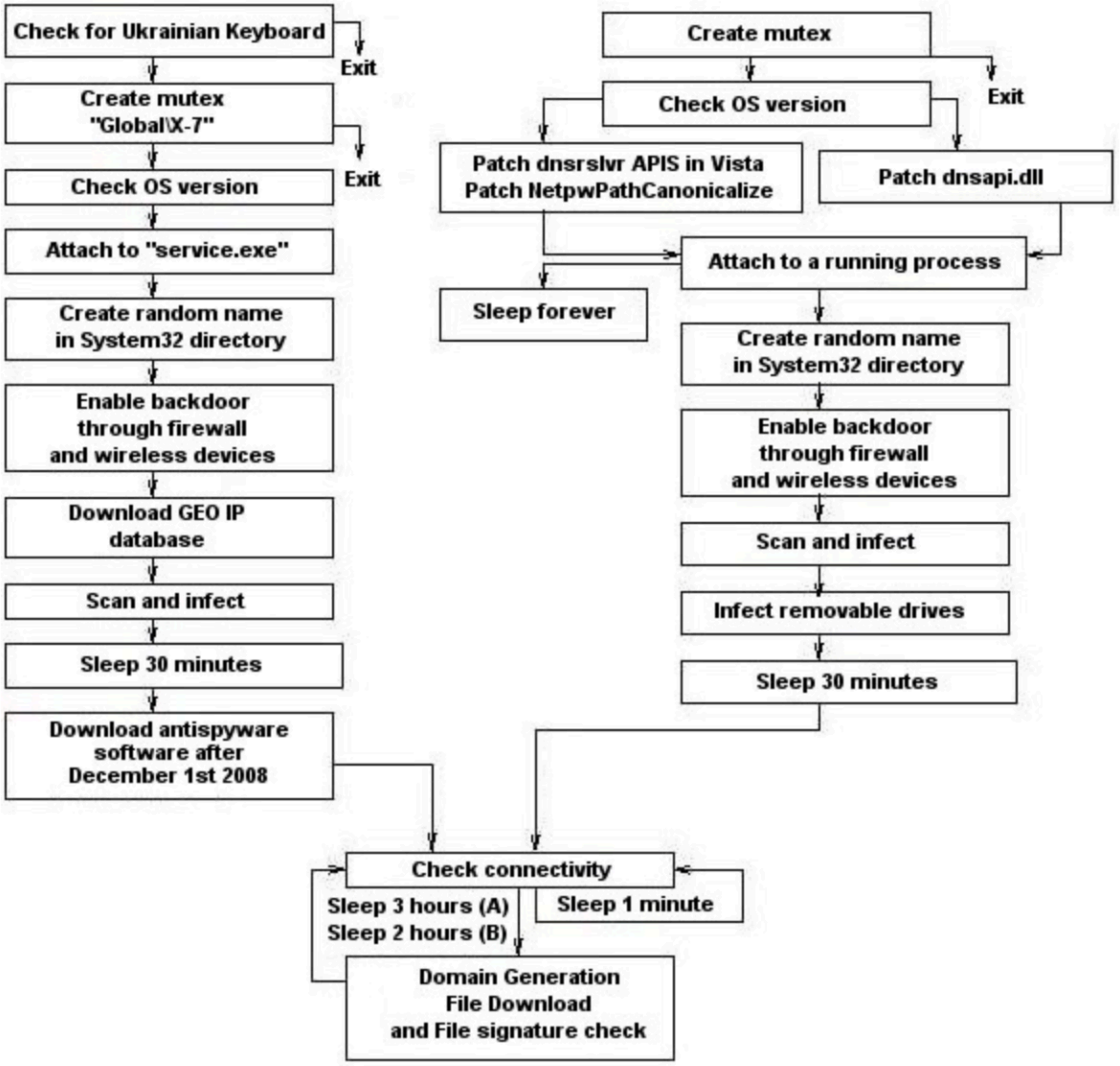
. . .

Buffer Overflows

Stack Smashing

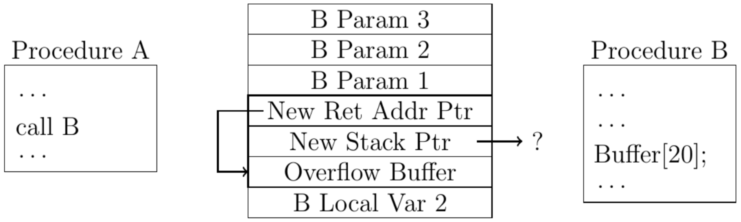


Conficker


Check for Ukrainian Keyboard
Exit
Create mutex "Global\X-7"
Exit
Check OS version
Attach to "service.exe"
Create random name in System32 directory
Enable backdoor through firewall and wireless devices
Download GEO IP database
Scan and infect
Sleep 30 minutes
Download antispyware software after December 1st 2008
Create mutex
Exit
Check OS version
Patch dnsrslvr APIS in Vista Patch NetpwPathCanonicalize
Patch dnsapi.dll
Attach to a running process
Sleep forever
Create random name in System32 directory
Enable backdoor through firewall and wireless devices
Scan and infect
Infect removable drives
Sleep 30 minutes
Check connectivity
Sleep 3 hours (A)
Sleep 2 hours (B)
Sleep 1 minute
Domain Generation File Download and File signature check

Standard Countermeasures

Attacks


Procedure A
...
call B
...

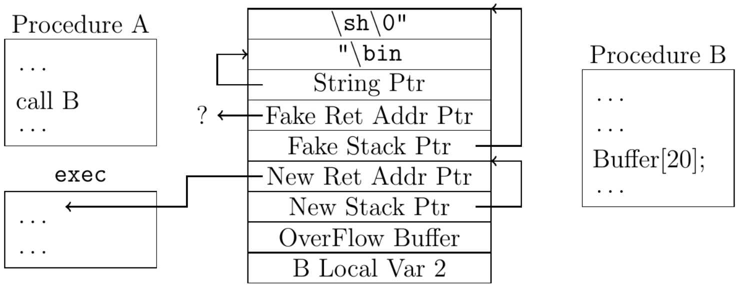
B Param 3
B Param 2
B Param 1
New Ret Addr Ptr
New Stack Ptr
Overflow Buffer
B Local Var 2

?

Procedure B
...
...
Buffer[20];
...

```
INTERNAL_SIZE_T prev_size;    /* size of prev chunk (if free) */
INTERNAL_SIZE_T size;         /* size of chunk  */

struct chunk * fd;            /* double links -- used only if free */
struct chunk * bw;
```


Procedure A
...
call B
...

exec
...
...

\sh\0"
"\bin
String Ptr
Fake Ret Addr Ptr
Fake Stack Ptr
New Ret Addr Ptr
New Stack Ptr
OverFlow Buffer
B Local Var 2

?

Procedure B
...
...
Buffer[20];
...

Defenses

x86

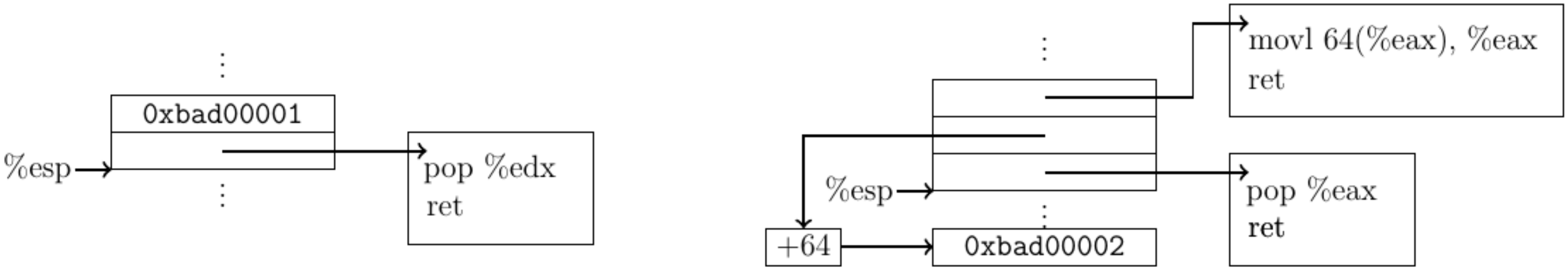
- Intel Instruction Set Architecture (ISA)
- Introduced 1978, still supported
- As of 2018, most common architecture on servers, PCs, and laptops
- dense instruction set
- variable length instructions
- not word aligned

Gadgets

```
f7 c7 07 00 00 00      test $0x00000007, %edi
0f 95 45 c3            setnzb -61 (%ebp)
```

```
c7 07 00 00 00 0f      movl $0x0f000000, (%edi)
95                     xchg %ebp, %eax
45                     inc %ebp
c3                     ret
```

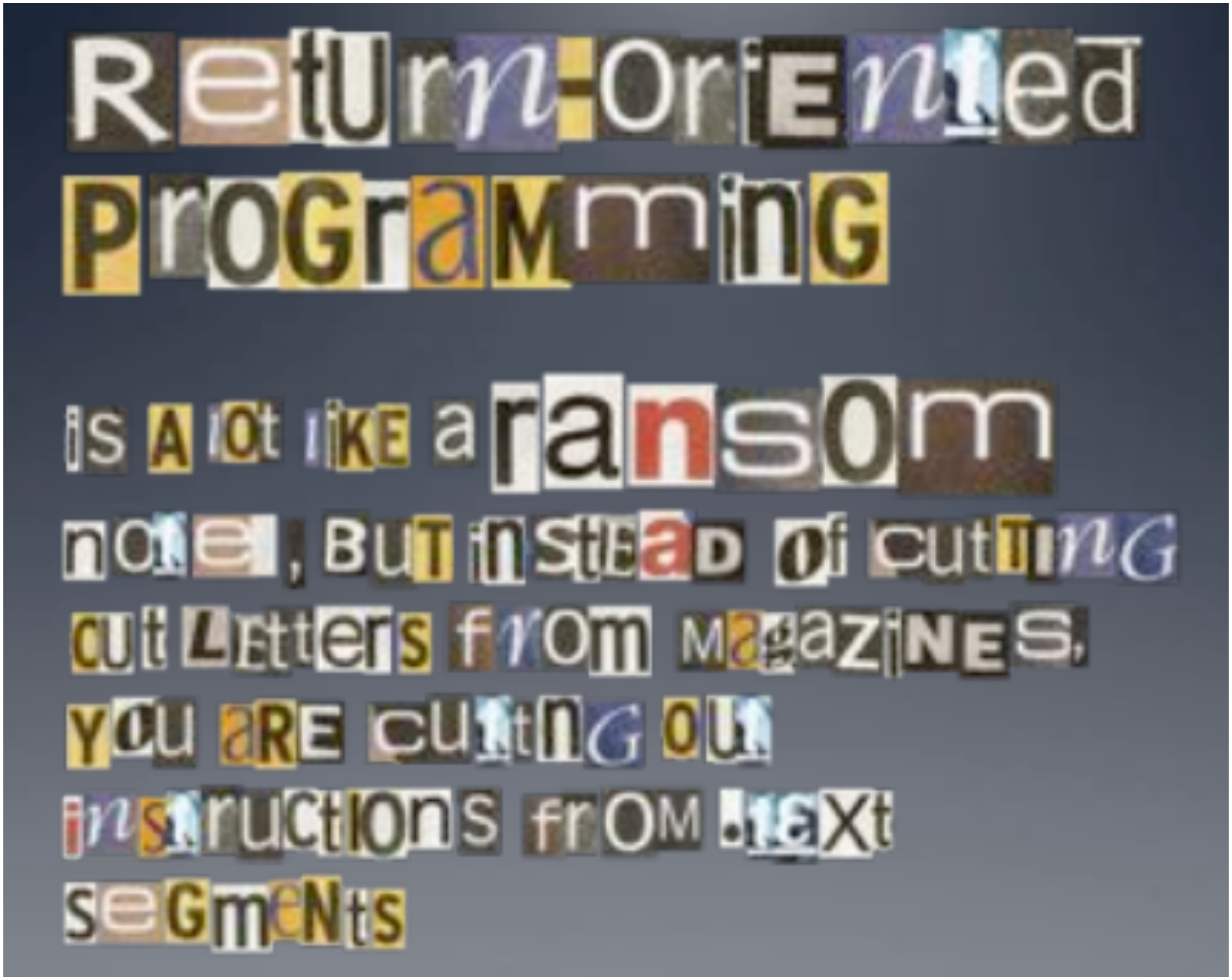
Gadgets



(a) Load constant gadget

(b) Load from memory gadget

Return Oriented Programming

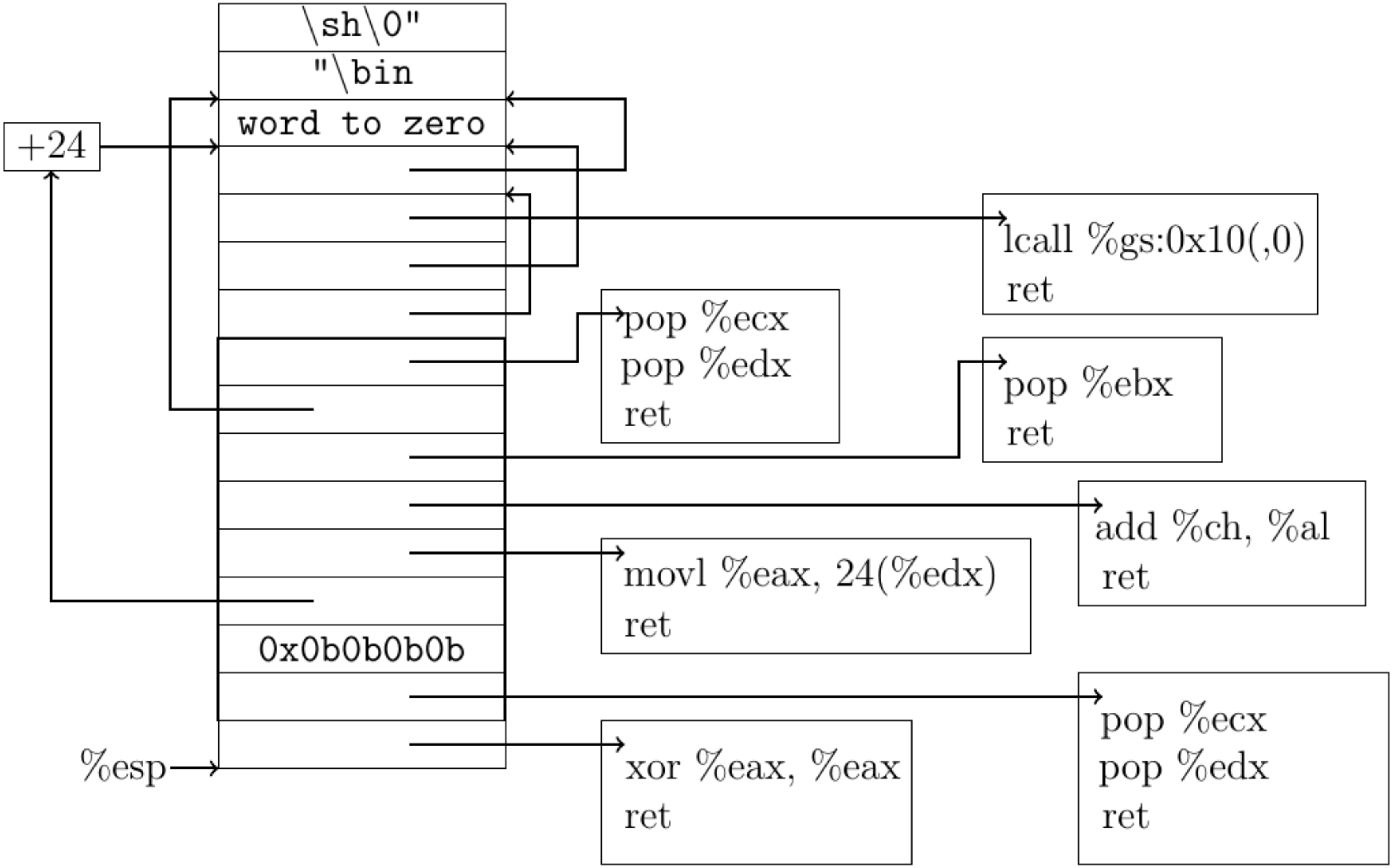


Return-Oriented Programming

is a lot like a ransom note, but instead of cutting cut letters from magazines, you are cutting out instructions from text segments

Image By: Dino Dai Zovi

Return-Oriented Shellcode



Testing

- Goal is to expose existence of faults, so that they can be fixed
- **Unit testing:** isolated components
- **Integration testing:** combined components
- **System testing:** functionality, performance, acceptance

Testing

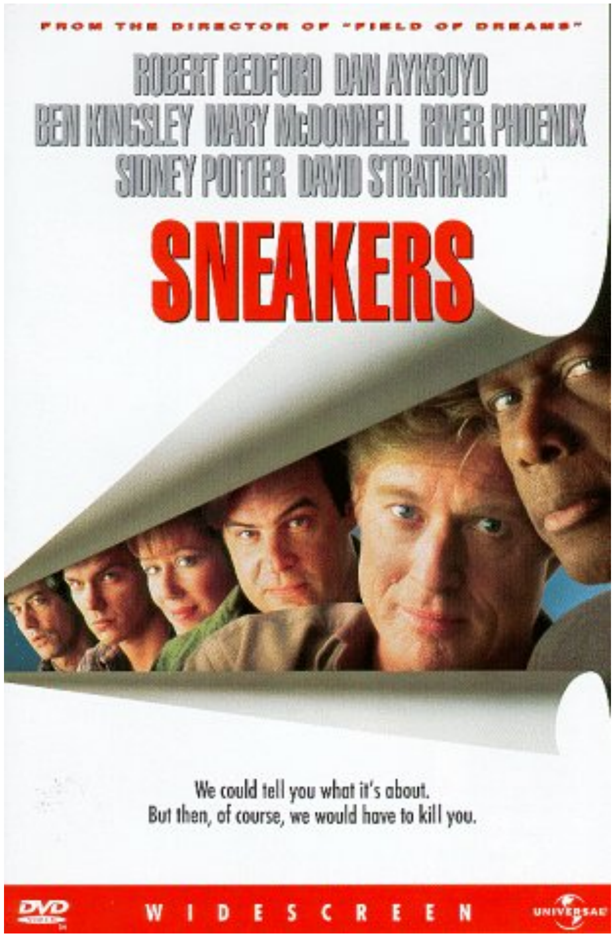
When do you stop testing?

- **Bad answer:** when time is up
- **Bad answer:** what all tests pass
- **Fun fact:** Pr[undetected faults] increases with # detected faults [Myers 1979, 2004]
- **Better answer:** when methodology is complete (code coverage, paths, boundary cases, etc.)
- **Future answer:** statistical estimation says Pr[undetected faults] is low enough (active research)

Testing for security?

Penetration testing

- Experts attempt to attack
  - Internal vs. external
  - Overt vs. covert
- Typical vulnerabilities exploited:
  - Passwords (cracking)
  - Buffer overflows
  - Bad input validation
  - Race conditions / TOCTOU
  - Filesystem misconfiguration
  - Kernel flaws


FROM THE DIRECTOR OF "FIELD OF DREAMS"
ROBERT REDFORD DAN AYKROYD
BEN KINGSLEY MARY McDONNELL RIVER PHOENIX
SIDNEY POITIER DAVID STRATHAIRN
SNEAKERS
We could tell you what it's about.
But then, of course, we would have to kill you.
DVD
WIDESCREEN
UNIVERSAL

Fuzz testing

[Barton Miller, 1989, 2000, 2006]

- Generate random inputs and feed them to programs:
  - Crash? hang? terminate normally?
  - Of ~90 utilities in '89, crashed about 25-33% in various Unixes
  - Crash implies buffer overflow potential
- Since then, "fuzzing" has become a standard practice for security testing
- Results have been repeated for X-windows system, Windows NT, Mac OS X
  - Results keep getting **worse** in GUIs but better on command line

Fuzz testing

Testing strategy:

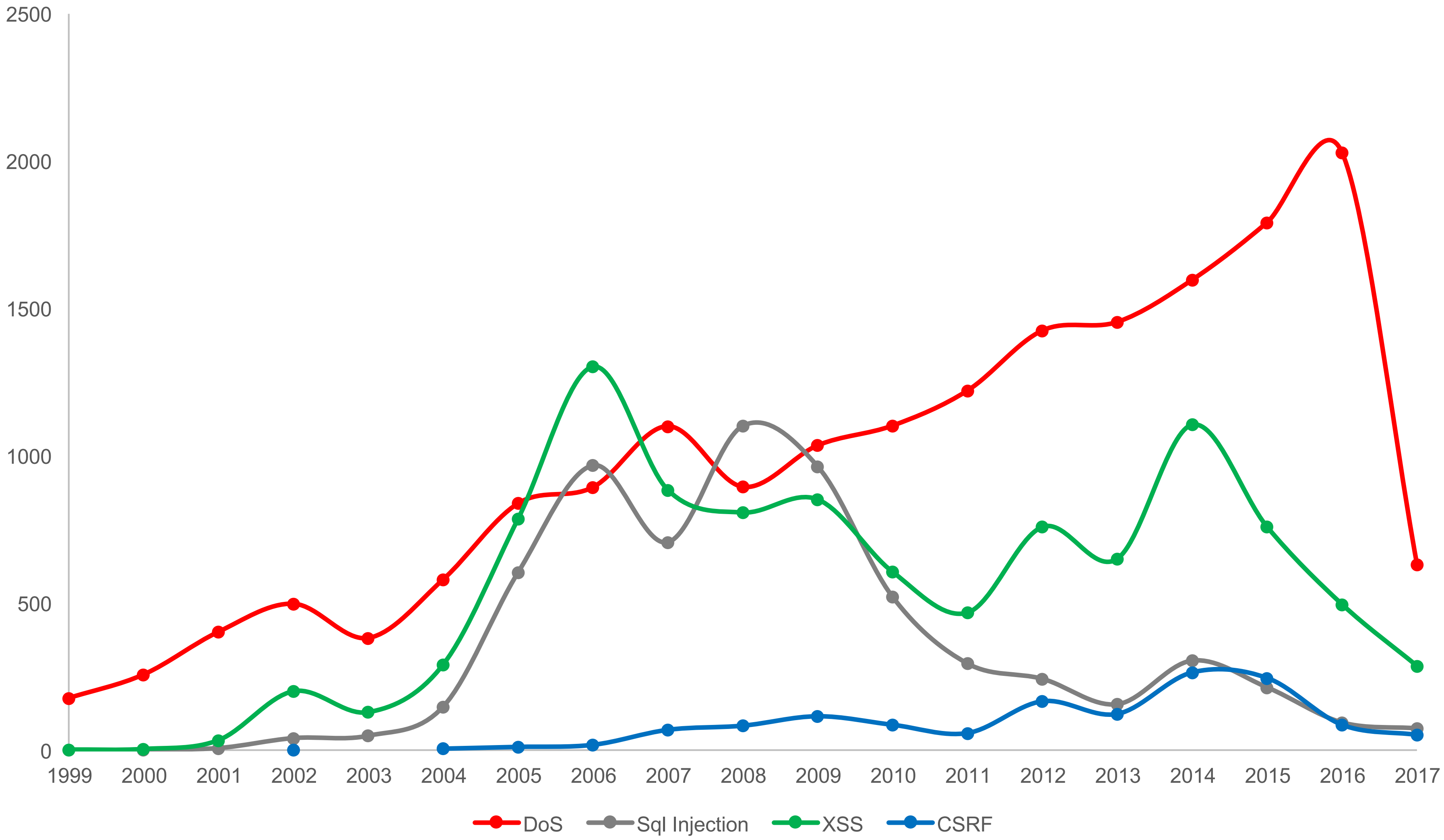
- Purely random no longer so good, just gets low-hanging fruit
- Better:
  - Use grammar to generate inputs
  - Or randomly mutate good inputs in small ways
    - especially for testing of network protocols
  - Research: use analysis of source code to guide mutation of inputs

FindBugs

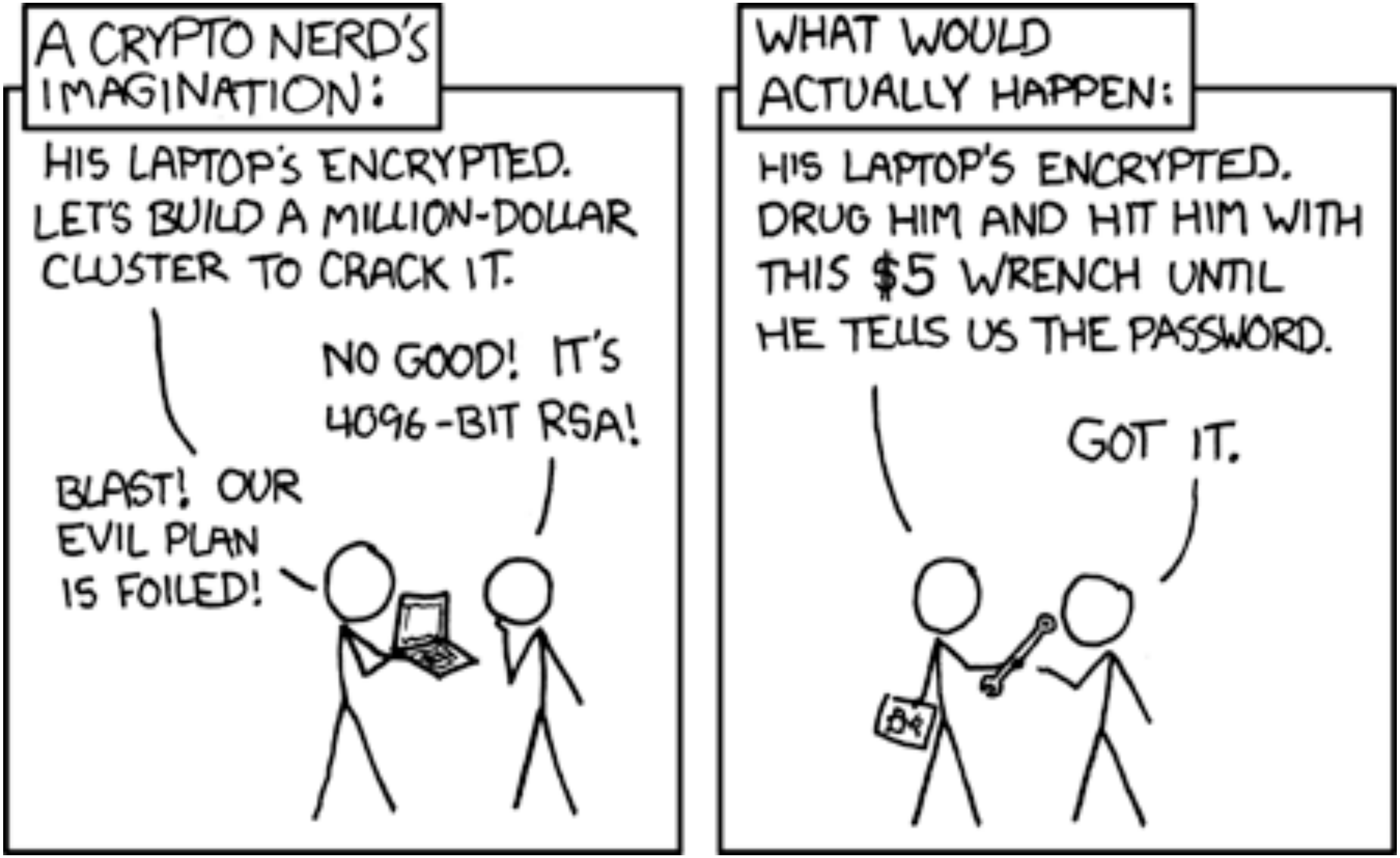
- Looks for *patterns* in code that are likely faults and that are likely to cause failures
- Categorizes and prioritizes bugs for presentation to developer
- Watch video of Prof. Bill Pugh, developer of FindBugs, present it to a Google audience:

  https://www.youtube.com/watch?v=8eZ8YWVl-2s

Web Vulnerabilities by Year



Threat Models


A CRYPTO NERD'S IMAGINATION:
HIS LAPTOP'S ENCRYPTED. LET'S BUILD A MILLION-DOLLAR CLUSTER TO CRACK IT.
NO GOOD! IT'S 4096-BIT RSA!
BLAST! OUR EVIL PLAN IS FOILED!
WHAT WOULD ACTUALLY HAPPEN:
HIS LAPTOP'S ENCRYPTED. DRUG HIM AND HIT HIM WITH THIS $5 WRENCH UNTIL HE TELLS US THE PASSWORD.
GOT IT.

Threats

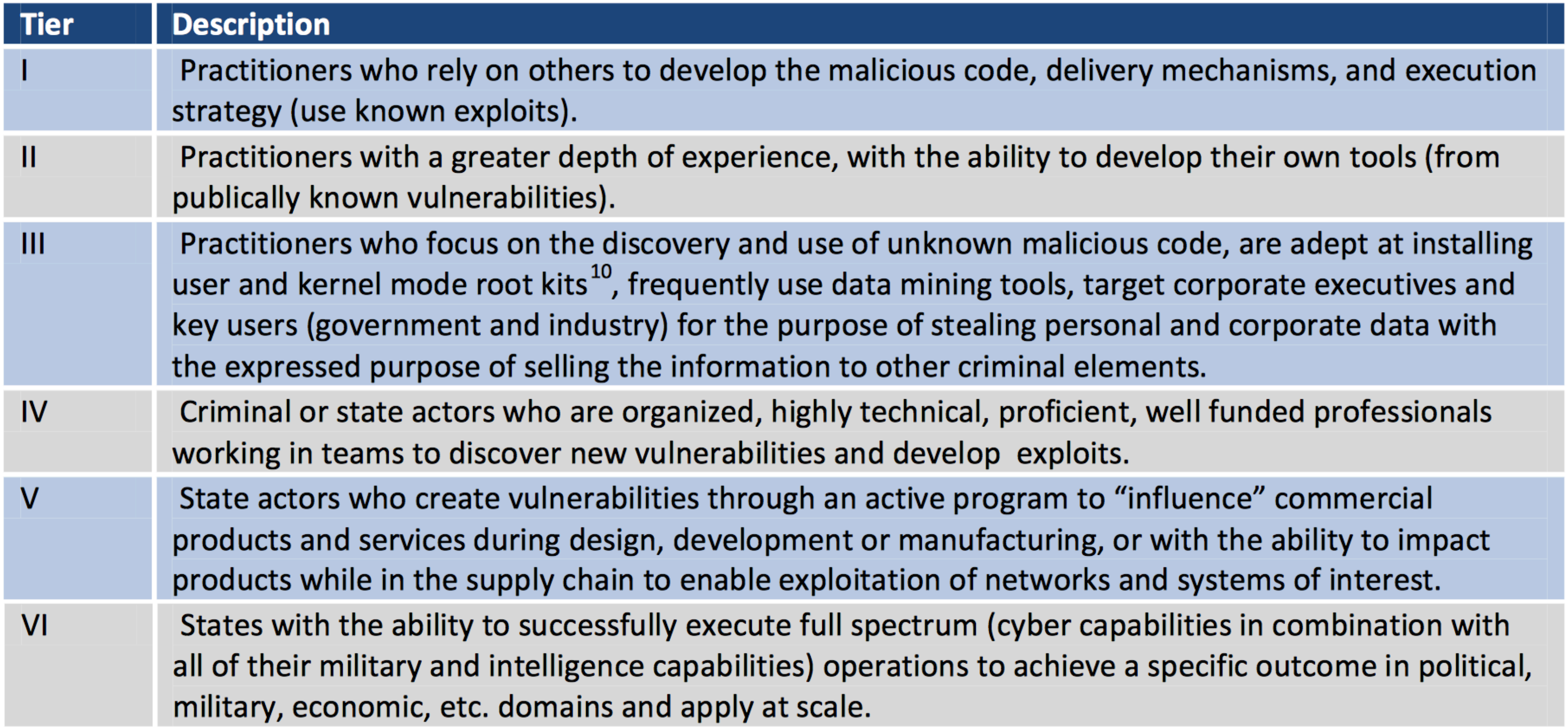
A principal that has potential to cause harm to assets

- Adversary or attacker: a human threat, motivated and capable
- Sometimes humans aren't malicious: accidents happen
- Sometimes non-humans cause harm: floods, earthquakes, power outage, hardware failure

Threat Models

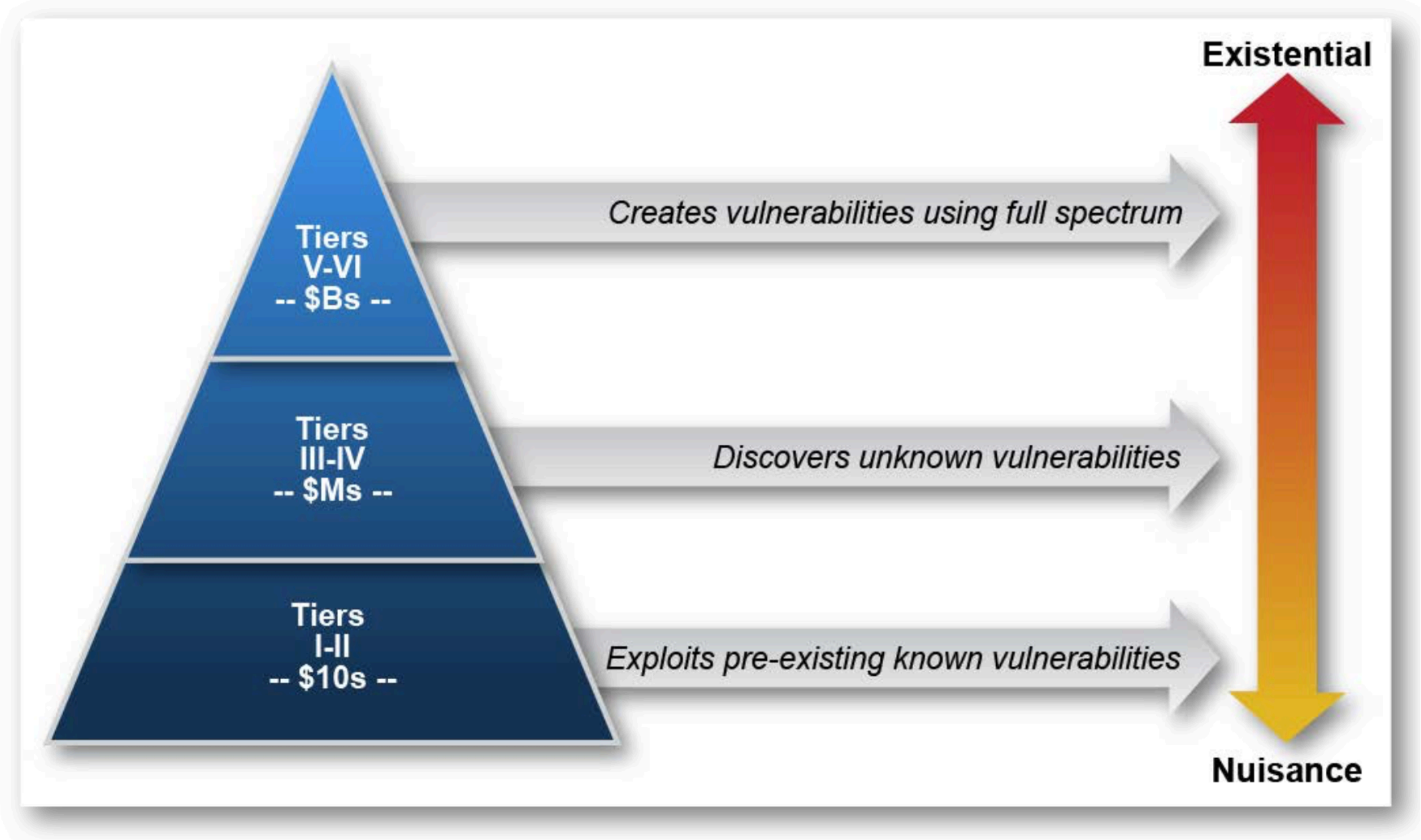
- Identify threats of concern to system
  - Especially malicious, human threats
  - What kinds of attackers will system resist?
  - What are their **motivations**, **resources**, and **capabilities**?
- Best if analysis is specific to system and its functionality
- Non threats?
  - Trusted hardware
  - Trusted environment
  - e.g., physically secured machine room reachable only by trustworthy system operators

Threats (DoD)



Tier	Description
I	Practitioners who rely on others to develop the malicious code, delivery mechanisms, and execution strategy (use known exploits).
II	Practitioners with a greater depth of experience, with the ability to develop their own tools (from publically known vulnerabilities).
III	Practitioners who focus on the discovery and use of unknown malicious code, are adept at installing user and kernel mode root kits[10], frequently use data mining tools, target corporate executives and key users (government and industry) for the purpose of stealing personal and corporate data with the expressed purpose of selling the information to other criminal elements.
IV	Criminal or state actors who are organized, highly technical, proficient, well funded professionals working in teams to discover new vulnerabilities and develop exploits.
V	State actors who create vulnerabilities through an active program to “influence” commercial products and services during design, development or manufacturing, or with the ability to impact products while in the supply chain to enable exploitation of networks and systems of interest.
VI	States with the ability to successfully execute full spectrum (cyber capabilities in combination with all of their military and intelligence capabilities) operations to achieve a specific outcome in political, military, economic, etc. domains and apply at scale.

Threats (DoD)


Existential
Tiers
V-VI
-- $Bs --
Creates vulnerabilities using full spectrum
Tiers
III-IV
-- $Ms --
Discovers unknown vulnerabilities
Tiers
I-II
-- $10s --
Exploits pre-existing known vulnerabilities
Nuisance

Classifying Threats

[S1, based on U.S. Defense Science Board]

- Inquisitive people, unintentional blunders
- Hackers driven by technical challenges
- Disgruntled employees or customers seeking revenge
- Criminals interested in personal financial gain, stealing services, or industrial espionage
- Organized crime with the intent of hiding something or financial gain
- Organized terrorist groups attempting to influence policy by isolated attacks
- Foreign espionage agents seeking to exploit information for economic, political, or military purposes
- Tactical countermeasures intended to disrupt specific weapons or command structures
- Multifaceted tactical information warfare applied in a broad orchestrated manner to disrupt a major military missions
- Large organized groups or nation-states intent on overthrowing a government

Threat Model = Capabilities

- privilege levels

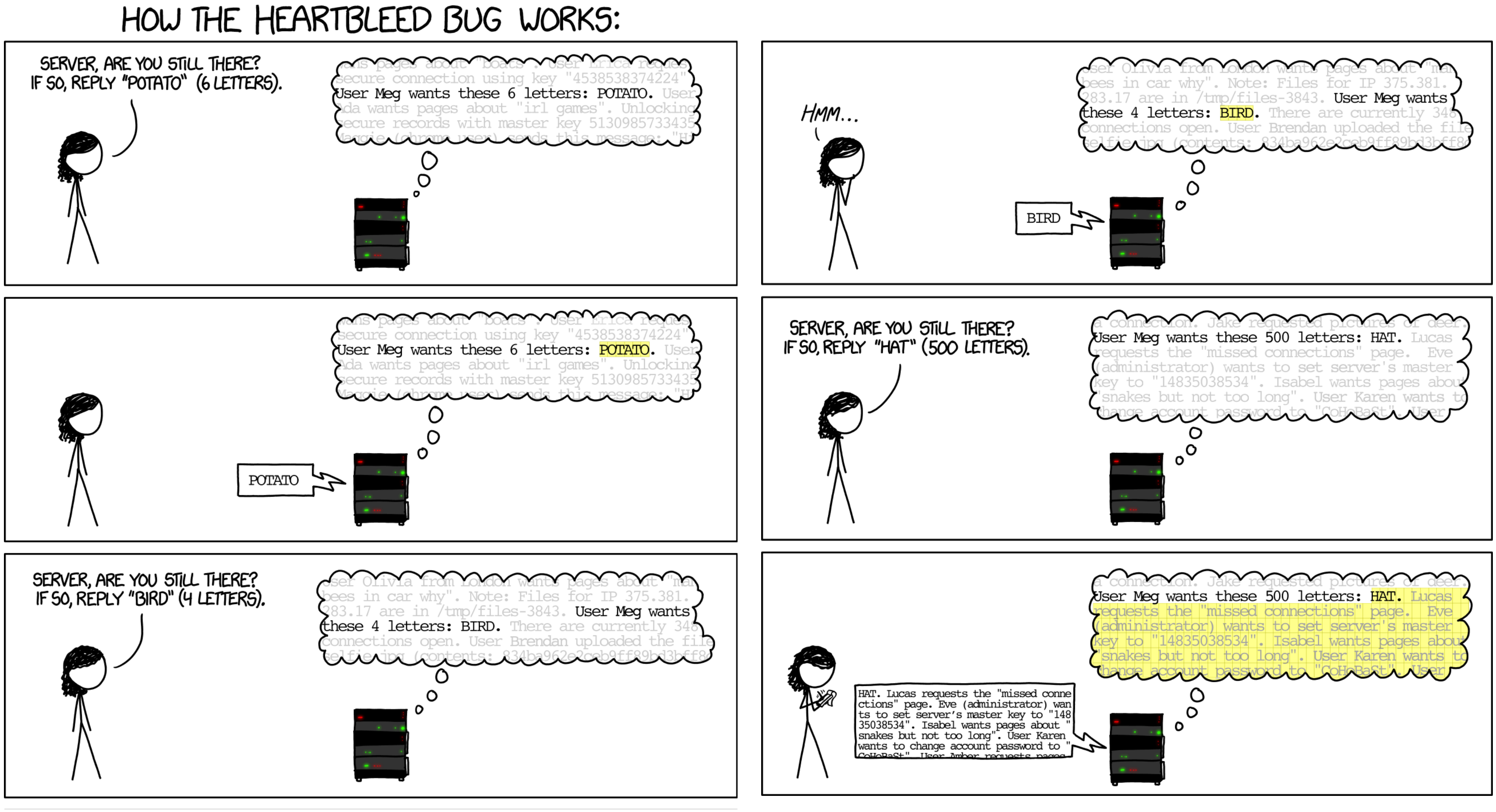
DIRTY COW

Threat Model = Capabilities

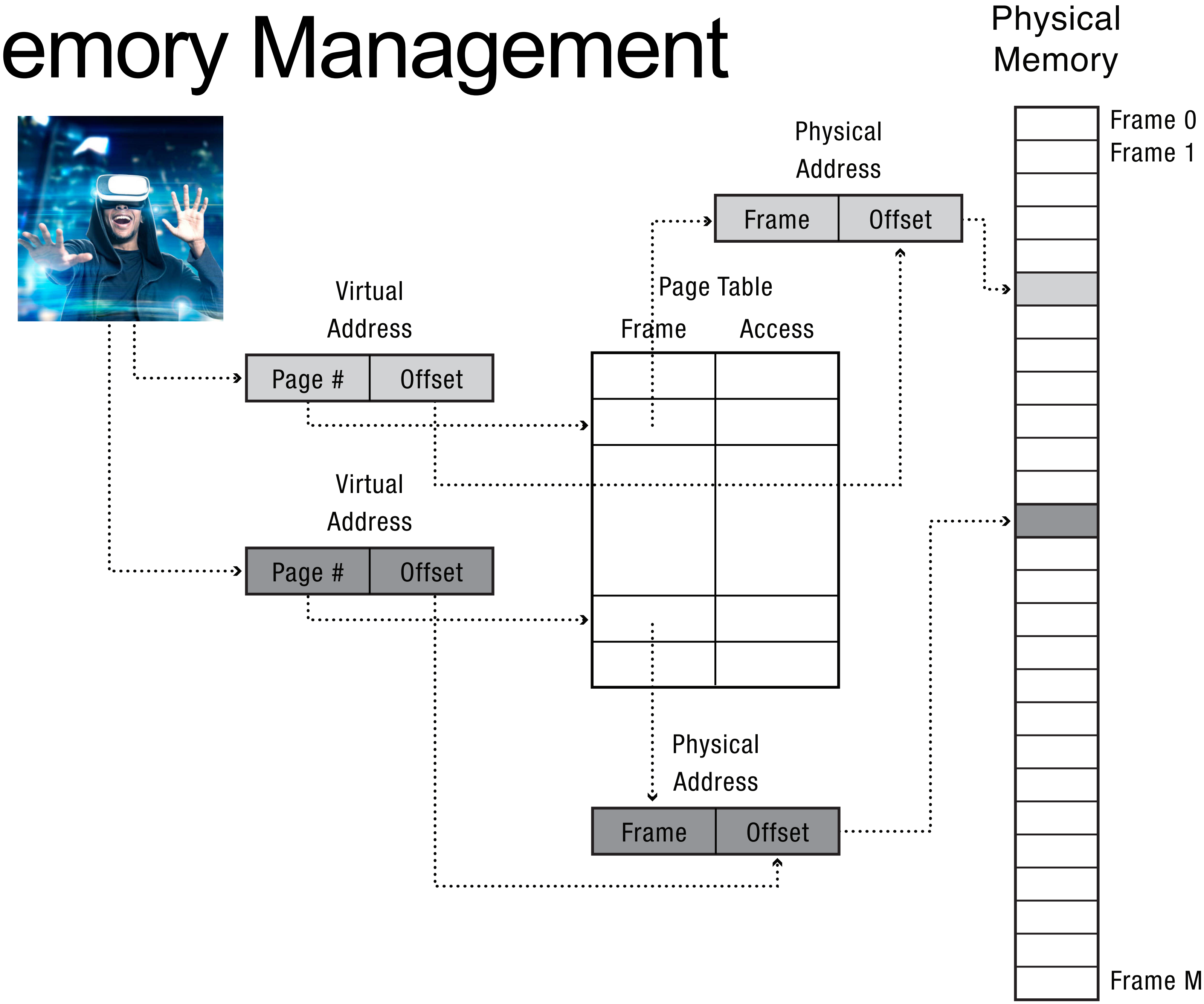
- privilege levels
- memory access

Heartbleed

Heartbleed


HOW THE HEARTBLEED BUG WORKS:
SERVER, ARE YOU STILL THERE? IF SO, REPLY "POTATO" (6 LETTERS).
wants pages about "boats". User Erica requests secure connection using key "4538538374224". User Meg wants these 6 letters: POTATO. User Ada wants pages about "irl games". Unlocking secure records with master key 5130985733435 Maggie (chrome user) sends this message: "H
POTATO
SERVER, ARE YOU STILL THERE? IF SO, REPLY "BIRD" (4 LETTERS).
User Olivia from London wants pages about "bees in car why". Note: Files for IP 375.381.283.17 are in /tmp/files-3843. User Meg wants these 4 letters: BIRD. There are currently 346 connections open. User Brendan uploaded the file selfie.jpg (contents: 834ba962e2ceb9ff89bd3bff84
HMM...
BIRD
SERVER, ARE YOU STILL THERE? IF SO, REPLY "HAT" (500 LETTERS).
a connection. Jake requested pictures of deer. User Meg wants these 500 letters: HAT. Lucas requests the "missed connections" page. Eve (administrator) wants to set server's master key to "14835038534". Isabel wants pages about "snakes but not too long". User Karen wants to change account password to "CoHoBaSt". User
HAT. Lucas requests the "missed connections" page. Eve (administrator) wants to set server's master key to "14835038534". Isabel wants pages about "snakes but not too long". User Karen wants to change account password to "CoHoBaSt". User Amber requests pages

Memory Management



Speculative Execution

```
int i1, i2;
boolean b1,b2;
boolean[] a1,a2;


if (i1 < a1.length()) {
    boolean bval= a1[i1];
    if(bval){i2= 1;} else{i2= 0;}
    if(i2 < a2.length()){
        b2 = a2[i2];
    }
}
```

Timing

MODE
SPLIT RESET
START STOP
ACCUSPLIT
STOPWATCH
10:5323
MIN.
SEC.
1/100
PRO SURVIVOR
601X
3V.1

Threat Model = Capabilities

- privilege levels
- memory access
- physical access

Stuxnet

Threat Model = Capabilities

- privilege levels
- memory access
- physical access
- key access

FileVault

The iPhone Case

Threat Model = Capabilities

- privilege levels
- memory access
- physical access
- key access
- network access

Network Adversaries

Attacker Properties			
Membership	insider		outsider
Method	active		passive
Adaptability	dynamic		static
Organization	cooperative		individual
Scope	global	extended	local
Motivation	malicious	rational	opportunistic

Dyn DDoS

Threat Models

"Security is lax on this side."