

SHELLPHISH

DEF CON 27

Capture the Flag *Finals*

Shortman

# The CTF

Live Attack/Defense CTF

16 Teams from all over the world

Must qualify by either winning a qualifier or finishing in the top X in the Defcon  
qualifier CTF

# Pre-qualified Teams

[DEF CON 2018 CTF](#) - 12 August 2018 - prequalified: DEFKOR00T

[HITCON CTF 2018](#) - 21 October 2018 - prequalified: Dragon Sector

[RuCTFE 2018](#) - 10 November 2018 - prequalified: saarsec

[C3CTF 2018](#) - 27 December 2018 - prequalified: mhackeroni

[PlaidCTF 2019](#) - 12 April 2019 - prequalified: HITCON

# Defcon Qualifiers

#	Team	Completed	Speedrun Individual	Speedrun Overall	Points
1	PPP	👾👾👾👾👉🚀🚀🚀👉👉👉👉👉👉👉👉👉🚀🚀👉👉	90	0	3681
2	HITCON ✕ BFKinesis	👾👾👾👾👉👉🚀🚀🚀👉👉👉👉👉👉👉👉👉🚀🚀👉👉	100	100	3571
3	SeoulPlusBadAss	👾👾👾🚀👾👉🚀🚀🚀👉👉👉👉👉👉👉👉👉🚀🚀👉👉	235	300	3230
4	A*0*E	👾👾👾👾👉🚀🚀🚀👉🚀👉👉👉👉👉👉👉🚀🚀👉👉	135	200	3195
5	Shellphish	👾👾👾👾👉🚀🚀👉👉👉👉🚀👉👉👉👉👉👉🚀🚀👉👉	65	0	3100
6	Sauercloud	👾👾👾🚀👾🚀👉👉👉👉👉👉👉👉👉🚀🚀👉👉👉🚀🚀	55	0	2932
7	Samurai	👾👾👾🚀👉🚀👾👉🚀👉👉👉👉👉👉👉👉🚀🚀👉👉	55	0	2918
8	Tea Deliverers	👾👾👾👉👾🚀👉👉👉👉👉👉👉👉👉👉👉🚀🚀👉👉	75	0	2734
9	CGC	👾👾👾👾👉👉🚀🚀🚀👉👉👉👉👉👉👉👉👉🚀🚀👉👉	100	0	2647
10	r00timentary	👾👾👾👾👉👉🚀🚀🚀👉🚀👉👉👉👉👉👉👉🚀🚀	150	0	2628
11	hxp	👾👾👾🚀🚀👾👉👉👉👉👉👉👉👉👉🚀🚀👉👉👉👉	55	0	2602
12	KaisHack GoN	👾👾👾👾🚀🚀👉👉👉👉👉👉👉👉👉🚀🚀👉👉👉👉	60	0	2465
13	TokyoWesterns	👾👾👾👉🚀🚀👾👉👉👉👉👉👉👉👉👉👉👉🚀🚀	60	0	2401
14	r3k4pig	👾👾👾👾👉🚀🚀🚀👉👉👉👉👉👉👉👉👉🚀👉👉	60	0	2337
15	RPISEC	👾👾👾👾🚀👉👉👉👉👉👉👉👉👉👉👉🚀🚀👉👉	65	0	2283

# Thursday (Day -1)

We get an information “leak” from the Order of the Overflow, that instructed us to bring the following tools:

- Microsoft Windows + Visual Studio
- MacOS + XCode + iOS SDK
- Any GNU/Linux distribution with proper toolchain + Android SDK
- FreeBSD (comes with toolchain)
  
- An extra monitor that supports HDMI...



# Friday (Day 1)

Game started at 10am (after ~5 hours of sleep)

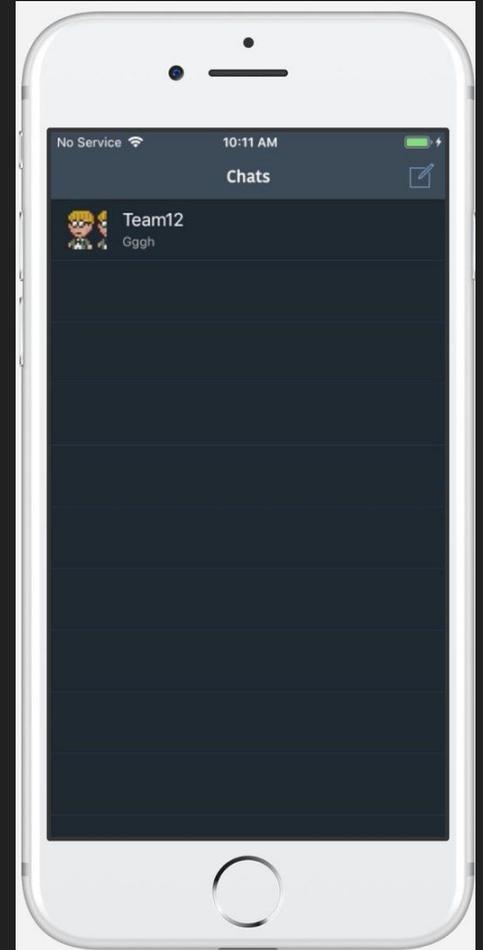
First challenges released:

- TelOoOgram: iOS messaging app similar to telegram (Objective C)
- AoOoL: Webserver, written in ??
- ROPShip: King of the Hill challenge

# Hackers Don't Use Macs....

But I actually brought my UCSB Macbook Pro

Hello TeloOogram!



# TeloOogram

- First bug identified
  - Unused “VoIP” server with a trivial buffer overflow
  - Appeared to be unexploitable
  - Easily patched (patch deployed)

# TeloOogram

- Second bug identified
  - The app requests avatar.png from contacts
  - Let's try requesting other files...
  - Success. Stole other teams creds.txt (username/password)
  - Oh yeah, and their flags
  - Easily patched (patch deployed)
- Sarsec getting more flags than us, but not exploiting us...
  - Hours pass...
  - Turns out other teams aren't great at patching
    - Try **./flag** instead of **flag**

# TeloOogram

- Third bug identified
  - Objective C parser used that was deprecated for security reasons
  - This is a nasty one...
  - Goes unexploited by any team, despite our best efforts
  -

# TeloOogram

- Removed from the game at the end of Day 1
  - We rejoice

# AoOol

Some webserver written in C/C++

- Responds to GET, UPLOAD, and CONFIG commands

Looks like there are some funky bits with parsing of a config file

I start getting spun up... then fall asleep.

# Saturday (Day 2)

Game starts at 10am (again)

- Actually a little bit late, but that's normal
- I start working on AoOol again, until...

**fish** we are getting a team XBox

be ready!!!!

**rhelmut** 🏳️‍🌈 A fu<sup>n</sup>ing what

Okay I guess I'm coming to the floor



2

**fish** we are getting a team XBox

be ready!!!!

**rhelmot** 🏳️‍🌈 A fu<sup>n</sup>ing what

Okay I guess I'm coming to the floor



2

**shortman**



I used to mod xboxes as a side business

**fish** we are getting a team XBox

be read **salls** @shortman you should come here

**rhelmat**

to work on the xbox stuff

to the floor

Okay I

**shortman** 🤘 Is there a seat?



2

**salls** yeah

one of us will switch

guys we have an issue with the xbox, anyone expert at networking?

**zanardi** xbox experts should come to the floor now

however many

**degrigis** @shortman is coming

# DoOom on an original XBOX



# DoOom on an original XBOX



# First, The Good

The XBOX had been modded to download a .xbe file over the network

It was downloading a version of Chocolate Doom

Multiplayer game against other teams!

Scoring:

- Find OOO tiles and stand on them (1 point per second)

# The hard stuff

We are told that the XBOX must be “pingable” (turns out to be a lie...)

The original .xbe has shooting disable and username “sheeple”

You can only score with the username of your team id

E.g., [14]shellphish

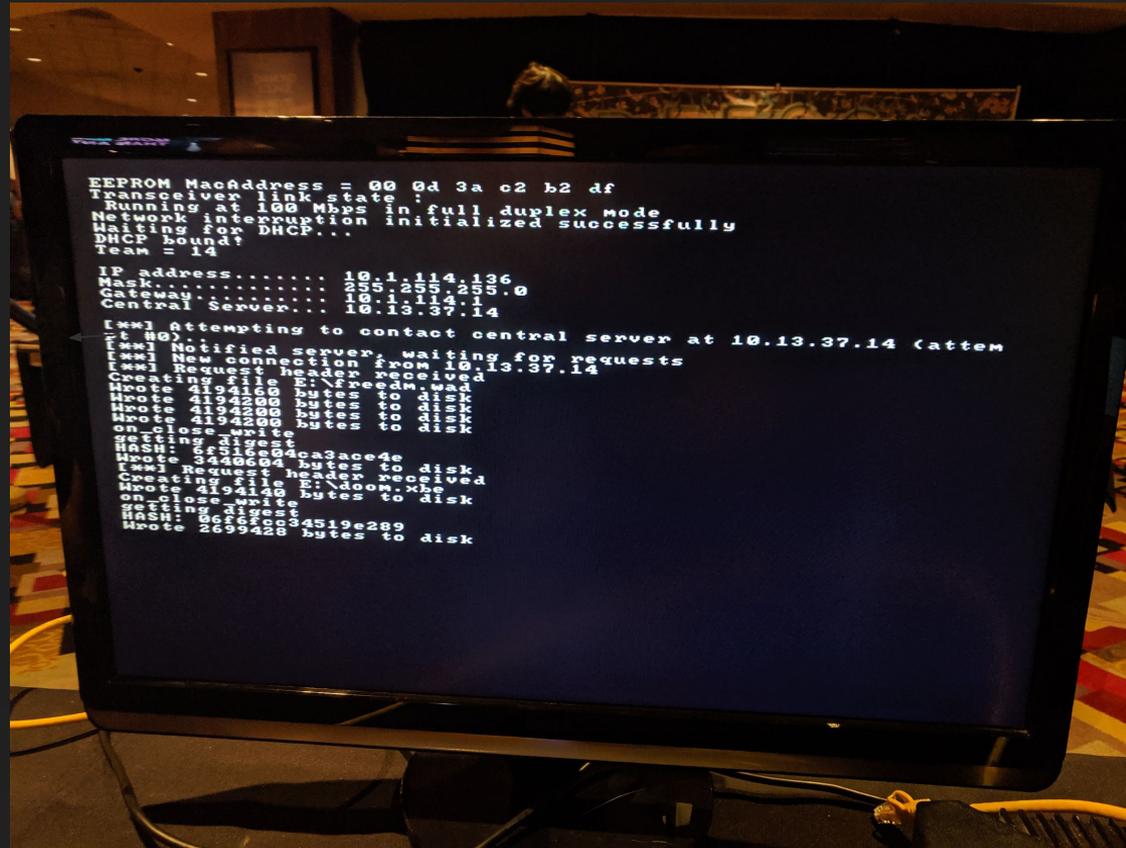
# Let the pwning begin!



No.	Time Source	Destination	Protocol	Length	Info
11	L	192.168.3.2	188.65.196.235	TCP	78 54163 -> 443 [SN] Seq=8 Win=6535 Len=0 MSS=1468 Win=32 TSval=2422636
22	L	188.65.196.235	192.168.3.2	TCP	82 443 -> 54163 [RST] ACK! Seq=8 Win=6535 Len=0 MSS=1468 Win=32 TSval=2422636
24	L	192.168.3.2	188.65.196.235	TCP	54 54163 -> 443 [ACK] Seq=1 Ack=1 Win=2144 Len=0
25	L	192.168.3.2	188.65.196.235	TLSv1.2	286 Client Hello
26	L	188.65.196.235	192.168.3.2	TCP	54 443 -> 54163 [ACK] Seq=1 Ack=233 Win=15744 Len=0
27	L	188.65.196.235	192.168.3.2	TLSv1.2	1434 Server Hello
28	L	188.65.196.235	192.168.3.2	TCP	1434 [TCP segment of a reassembled PDU]
29	L	188.65.196.235	192.168.3.2	TCP	1308 [TCP segment of a reassembled PDU]
30	L	192.168.3.2	188.65.196.235	TCP	54 54163 -> 443 [ACK] Seq=233 Ack=2761 Win=209736 Len=0
31	L	192.168.3.2	188.65.196.235	TCP	54 54163 -> 443 [ACK] Seq=233 Ack=4897 Win=206808 Len=0
32	L	188.65.196.235	192.168.3.2	TCP	1434 [TCP segment of a reassembled PDU]
33	L	188.65.196.235	192.168.3.2	TLSv1.2	529 Certificate
34	L	192.168.3.2	188.65.196.235	TCP	54 54163 -> 443 [ACK] Seq=233 Ack=5952 Win=261664 Len=0
35	L	192.168.3.2	188.65.196.235	TLSv1.2	129 Client Key Exchange
36	L	192.168.3.2	188.65.196.235	TLSv1.2	68 Change Cipher Spec
37	L	192.168.3.2	188.65.196.235	TLSv1.2	99 Encrypted Handshake Message
38	L	188.65.196.235	192.168.3.2	TCP	54 443 -> 54163 [ACK] Seq=5952 Ack=359 Win=13744 Len=0
39	L	188.65.196.235	192.168.3.2	TLSv1.2	185 Change Cipher Spec, Encrypted Handshake Message
40	L	192.168.3.2	188.65.196.235	TCP	54 54163 -> 443 [ACK] Seq=359 Ack=6083 Win=262808 Len=0
41	L	192.168.3.2	188.65.196.235	TLSv1.2	347 Application Data
53	L	188.65.196.235	192.168.3.2	TCP	54 443 -> 54163 [ACK] Seq=6083 Ack=652 Win=16768 Len=0
54	L	188.65.196.235	192.168.3.2	TLSv1.2	466 Application Data
59	L	192.168.3.2	188.65.196.235	TCP	54 54163 -> 443 [ACK] Seq=652 Ack=6415 Win=261728 Len=0
79	L	192.168.3.2	188.65.196.235	TCP	54 54163 -> 443 [FIN, ACK] Seq=652 Ack=6415 Win=262184 Len=0
80	L	188.65.196.235	192.168.3.2	TCP	54 443 -> 54163 [FIN, ACK] Seq=6415 Ack=653 Win=10708 Len=0

Frame 28: 1434 bytes on wire (11472 bits), 1434 bytes captured (11472 bits) on interface 0  
Ethernet II, Src: Realtek-PhysServer-32, 8c:8e:68:ff:64, 192:168:0a:00:ff:64, Dst: Apple\_ML3x28 (cc:25:ef:6d:3d:26)  
Internet Protocol Version 4, Src: 188.65.196.235, Dst: 192.168.3.2  
Transmission Control Protocol, Src Port: 443 (443), Dst Port: 54163 (54163), Seq: 1381, Ack: 233, Len: 1380  
0000 cc 2f 6f 3e 6f 26 4a 00 ff 68 00 69 00 \*...6.2...f.  
0010 85 8c cc 85 00 00 39 06 6a df bc 41 c4 eb 08 .....9...A...  
0020 83 82 0b 05 c4 30 2b 0d 4a 4f 50 58 .....1...M...  
0030 80 7b 35 14 08 00 3c 5d 56 83 55 fe 06 84 20 {5...}...U...  
0040 6c 6f 30 58 50 64 4c 53 fe 19 73 0d 6d 92 ...W...P...S...  
0050 fd 0b f1 28 0b 54 23 4f c8 0b 80 3c 08 a8 55 .....T90...\*...U  
0060 61 01 68 02 13 fe c8 89 bb 5e 8c 38 54 18 65 67 .....2...f...e.

# Let the pwning begin!



# Let the pwning begin!

Shooting enabled, points being scored... but... there's more..

WE FIND A HIDDEN ROOM THAT IS COVERED IN OOO TILES

The catch: you need to clip through walls to get there

# Becoming a God

We patch the binary to enable no clipping

IT WORKS!



We freak!

# Becoming a God

No points are being scored...

- Actually we can't tell if points are being scored

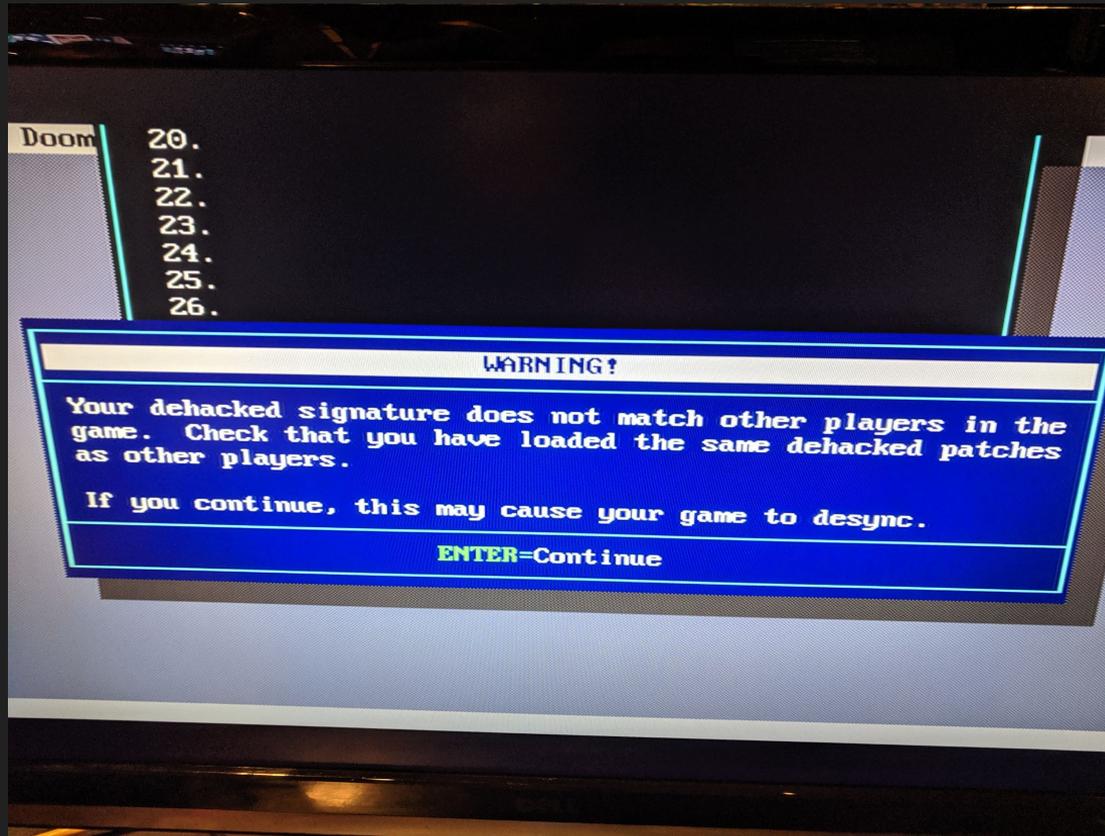
OOO tells us everything is fine

We fight for hours..

We don't know if it's working, or if we are scoring,

but we are Gods.

# We were DoOomed



# We were DoOomed

We needed to send our commands to the server as well, not just locally patch...

Also, the XBOX didn't need to be pingable...

Lack of feedback killed us.

We complained to the organizers, they promised to fix it next year.

# End of Friday

Finally, some rest...

What are the other challenges?

# The Bitflip Conjecture

---

## Definition:

A snippet of assembly code is `N-Flip Resistant` if its output remains constant (i.e., it produces the same output and exits with the same return value) even if ANY combination of N bits are flipped.

## One-flip Conjecture:

The x86 architecture is such that it is possible to write any arbitrary program (of any length) in a way that is 1-flip resistant.

- Balzaroth (Vegas 2019)

# The Bitflip Conjecture

Points are assigned based on how close you are from a complete proof

(i.e., based on how many bit flip your code was able to withstand)

---

But first, how do you want the registers initialized before executing the code?

1. I like all my registers set to zero
2. I want them pointing to the middle of a 64KB R/W region of memory)
3. Dont bother. Leave them as they are

# The Bitflip Conjecture

We are allotted 200 bytes of shellcode

This happens to be closely related to my research here...

Game on!

# The Bitflip Conjecture

Actually, the CTF is paused so we can't score

But we can still get our shellcode ready for morning

# The Bitflip Conjecture: Idea 1

Replicate shellcode, and do a checksum

```
BITS 64

_start:
    lea rax, [rel copy2]
    lea rbx, [rax-(copy2 - copy1)]
loop_start:
    dec al
    add cl, byte [rax]      ; add cl, [rax]
    cmp eax, ebx
    jnz loop_start

decide:
    cmp cl, 34
    jnz copy2

copy1:
    db SHELLCODE

copy2:
    db SHELLCODE
```



# The Bitflip Conjecture: Idea 2

Transactional Memory!

If the transaction fails, it will reset everything

PROBLEM 1: The xbegin instruction will always fail bitflips

PROBLEM 2: We need to flush the instruction cache... cpuid fails too

Still... Pretty good (~12 bits)

# The Bitflip Conjecture: Idea 3

What if we just fix the flipped bit...?

RAX = ptr to shellcode

RCX = offset to byte that was flipped

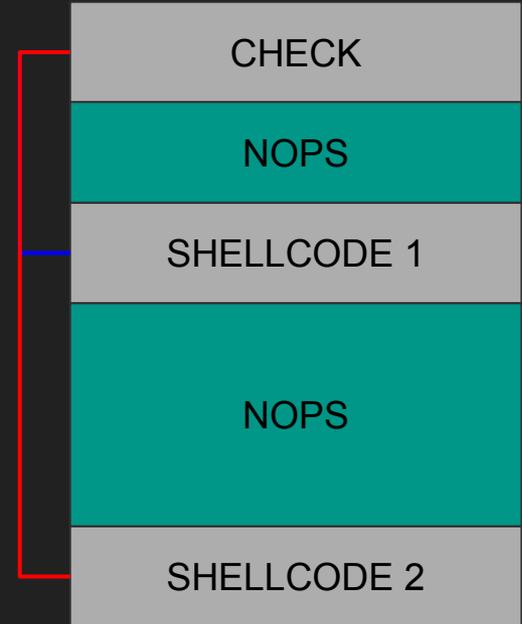
The bit that was flipped is on the stack somewhere

# The Bitflip Conjecture: Idea 3 (Improved)

Check offset

Jump to uncorrupted portion of the code

Now only our check needs to survive bit flips...



# The Bitflip Conjecture: Idea 3 (Improved)

4 Bits!!!

```
BITS 64

_start:
    sbb cl, (0x22 + copy2)
    jbe $+0x67
post_jump:

copy1:
    db SHELLCODE

buf:
    times (64 - (buf - post_jump)) db 0x90

copy2:
    db SHELLCODE
```

# Good, but not good enough

0 points scored

**subwire** ok folks, 996, we are not the highest tho

hxp next to us got 997

Untitled ▾

```
1 Bitflipping.....
2 [-x---x--] [-----xx] [-----] [-----] [-----] [-----]
  -] [-----] [-----]
3 [-----] [-----] [-----] [-----] [-----] [-----]
  -] [-----] [-----]
4 [-----] [-----] [-----] [-----] [-----] [-----]
  -] [-----] [-----]
5 [-----] [-----] [-----] [-----] [-----] [-----]
  -] [-----] [-----]
```

# Good, but not good enough

**fish** announcement: if you want to score points for bitflip, you need to score more than or equal to 999...

**@channel** ^^^

**saagarjha** So someone has a perfect?!

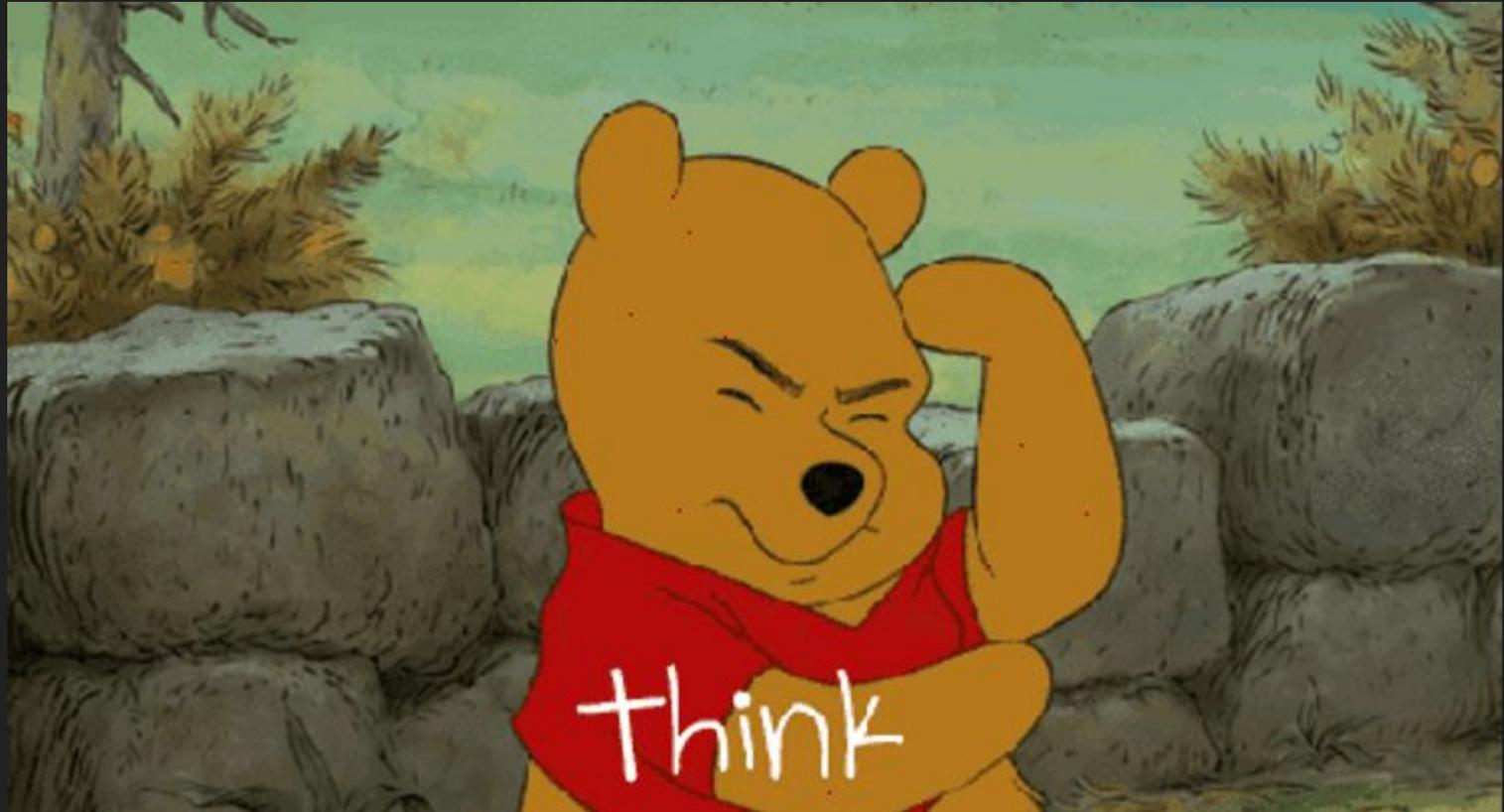
**GHOS1** How much are we getting now?

**saagarjha** 996

**salls** fu nnnn!

**fish** I bet 999 is 1-bit flipping

Good, but not good enough



# We can do better

**paul** 🌲 ???

that's 2 locally

**rhelmtot** 🏳️‍🌈 then there's a problem locally

**paul** 🌲 no, **chad** and I get the same thing

there's a problem remotely 😊

we're working on it

some register must be different

**zwimer** 🍷 Wait, I fucked up

**rhelmtot** 🏳️‍🌈 got 2 remotely



**subwire** !!

nice!

n

# Let's just fuzz offsets

**paul**  I'm fuzzing jump offsets in salls' 3 bit payload, should be able to get to 2

**zwimer**  We got 2

With lots of options

We got 1 !!!!

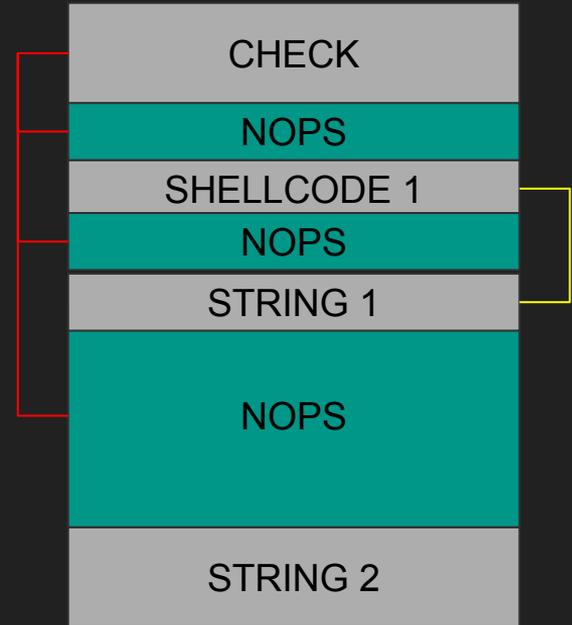
**paul**  HOLY SHI P!

[@subwire](#)

**shortman**  boom!!!!

# 1 Bit!!!

```
        BITS 64
_start:
        add al, cl
        jns $+0x60
copy1:
        NOPS
        SHELLCODE
        NOPS
        jmp copy1
the_string1:
        db "I am Invincible!"
buf:
        NOPS
Copy2:
        NOPS
        SHELLCODE
        STRING
```





# How to get 0



US



Tea Deliverers

# Final Scores

TOTAL	Attack	Defense	KoH
973 Plaid Parliament of Pwning	1442 Plaid Parliament of Pwning	213 Plaid Parliament of Pwning	769 HITCON <del>X</del> , BFKinesiS
▲ 772 HITCON <del>X</del> , BFKinesiS	1006 HITCON <del>X</del> , BFKinesiS	159 A*0*E	664 Plaid Parliament of Pwning
▲ 590 Tea Deliverers	815 Tea Deliverers	156 HITCON <del>X</del> , BFKinesiS	477 A*0*E
▼ 564 A*0*E	656 mhackeroni	147 mhackeroni	459 Tea Deliverers
▲ 556 mhackeroni	646 Samurai	132 r3kapig	443 KaisHack GoN
▼ 399 Samurai	510 A*0*E	130 Tea Deliverers	405 Sauercloud
▼ 375 Sauercloud	499 r00timentary	127 Sauercloud	377 mhackeroni
▲ 359 r00timentary	339 SeoulPlusBadAss	111 r00timentary	370 SeoulPlusBadAss
▼ 331 SeoulPlusBadAss	292 saarsec	100 Samurai	333 TokyoWesterns
▼ 284 Shellphish	131 r3kapig	98 Shellphish	269 Shellphish
284 r3kapig	114 Sauercloud	88 KaisHack GoN	173 saarsec
▼ 281 KaisHack GoN	109 Shellphish	75 SeoulPlusBadAss	123 Samurai
▼ 235 saarsec	106 CGC	58 saarsec	59 CGC
▼ 215 TokyoWesterns	96 TokyoWesterns	54 TokyoWesterns	46 r00timentary
▲ 110 CGC	8 hxp	35 CGC	5 hxp
67 hxp	2 KaisHack GoN	34 hxp	0 r3kapig
$400^{a/M_a} + 400^{d/M_d} + 200^{k/M_k}$	$\sum_{tick} (1 \text{ for each stolen flag})$	$\sum_{tick} (1 \text{ if non-exploited AND there were exploits})$	$\sum_{tick} (\text{per-service point logic})$
	$M_a = \max(\uparrow, 100) = 1442$	$M_d = \max(\uparrow, 100) = 213$	$M_k = \max(\uparrow, 100) = 769$

10th Place!

