fail0verflow

when success just isn't an option

Blog
About

Recent Posts

- PS4 Aux Hax 5: Flawed Instructions Get Optimized
- PS4 Aux Hax 4: Belize via CEC
- PS4 Aux Hax 3: Dualshock4
- PS4 Aux Hax 2: Syscon
- PS4 Aux Hax 1: Intro & Aeolia
- ShofEL2, a Tegra X1 and Nintendo Switch exploit
- Dumping a PS4 Kernel in "Only" 6 Days
- The First PS4 Kernel Exploit: Adieu
- Console Hacking 2016: Postscript
- In Memoriam: Ben "bushing" Byer
- Console Hacking 2015: Liner Notes
- 31c3 CTF - pong (pwn30)
- 31c3 CTF - safelock (signals20)
- HubCap: pwning the ChromeCast pt. 2
- HubCap: pwning the ChromeCast pt. 1
- OpenVizsla OV3 - Hardware
- RFID hacking preamble: a new peak detection for the proxmark3
- RFID hacking preamble: designing an FPGA IIR filter for the proxmark3
- Enhancing the AVIC-5000NEX - part 2
- Enhancing the AVIC-5000NEX
- plaidCTF 2014 - reeekeeeeee (web200)
- plaidCTF 2014 - rsa (for450)
- plaidCTF 2014 - wheeeee (crypto375)
- plaidCTF 2014 - bbos (for350)
- plaidCTF 2014 - graphs (crypto200)

In memory of Ben "bushing" Byer

Blogs

2022-05-17

PS4 Aux Hax 5: Flawed Instructions Get Optimized

By ps5_enthusiast

Filed under ps4 vulnerability exploit

Aaaand we're back, after an extended delay, to ... continue talking about hacking PS4 peripherals 😅.

This time, the DUT is the PS4 Virtual Reality peripheral: PSVR. We managed to find some major flaws - breaking secure boot and extracting all key material; let's go!

read more

2018-11-03

PS4 Aux Hax 4: Belize via CEC

By ps4_enthusiast

Filed under ps4 vulnerability exploit

This post describes another way to attain code execution on Aeolia (actually, the southbridge revision on PS4 Pro which was used in this case is named "Belize").

This exploit differs from the previously documented method as it does not have the prerequisite of gaining control of the APU. Additionally it is fairly generic and therefor workable on all currently released hardware and software versions of PS4.

read more

2018-07-30

PS4 Aux Hax 3: Dualshock4

By ps4_enthusiast

Filed under ps4 vulnerability exploit

In the PS4 Aux Hax series of posts, we'll talk about hacking parts of the PS4 besides the main x86 cores of the APU.

In this entry, we'll step outside of the PS4 itself, and take a look at pwning the main handheld controller used by the system.

read more

2018-07-30

PS4 Aux Hax 2: Syscon

By ps4_enthusiast

Filed under ps4 vulnerability exploit

In the PS4 Aux Hax series of posts, we'll talk about hacking parts of the PS4 besides the main x86 cores of the APU.

In this entry, we'll recount some parts of the path taken to get permanent arbitrary code exec on syscon.

read more

2018-07-30

PS4 Aux Hax 1: Intro & Aeolia

By ps4_enthusiast

Filed under ps4 vulnerability exploit

In the PS4 Aux Hax series of posts, we'll talk about hacking parts of the PS4 besides the main x86 cores of the APU.

In this first entry, we'll give some background for context and describe how we managed to run arbitrary code persistently on Aeolia, the PS4 southbridge.

read more

2018-04-24

ShofEL2, a Tegra X1 and Nintendo Switch exploit

By switch_enthusiast

Filed under switch vulnerability exploit linux

Welcome to ShofEL2 and Switch Linux, fail0verflow's boot stack for no-modification, universal code execution and Linux on the Nintendo Switch (and potentially any Tegra X1 platform).
Choosing whether to release an exploit or not is a difficult choice. Given our experiences with past consoles, we've been wary of releasing vulnerability details or exploits for fear of them being used primarily for piracy rather than homebrew. That said, the[1] Tegra bootrom bug is so obvious that multiple people have independently discovered it by now; at best, a release by other homebrew teams is inevitable, while at worst, a certain piracy modchip team might make the first move.

read more

2017-12-27

Dumping a PS4 Kernel in "Only" 6 Days

By ps4_enthusiast

Filed under ps4 vulnerability exploit

What if a secure device had an attacker-viewable crashdump format?
What if that same device allowed putting arbitrary memory into the crashdump?
Amazingly, the ps4 tempted fate by supporting both of these features!
Let's see how that turned out...

read more

2017-10-19

The First PS4 Kernel Exploit: Adieu

By ps4_enthusiast

Filed under ps4 vulnerability exploit

Plenty of time has passed since we first demonstrated Linux running on the PS4.
Now we will step back a bit and explain how we managed to jump from the browser process into the kernel such that ps4-kexec et al. are usable.

read more

2016-12-31

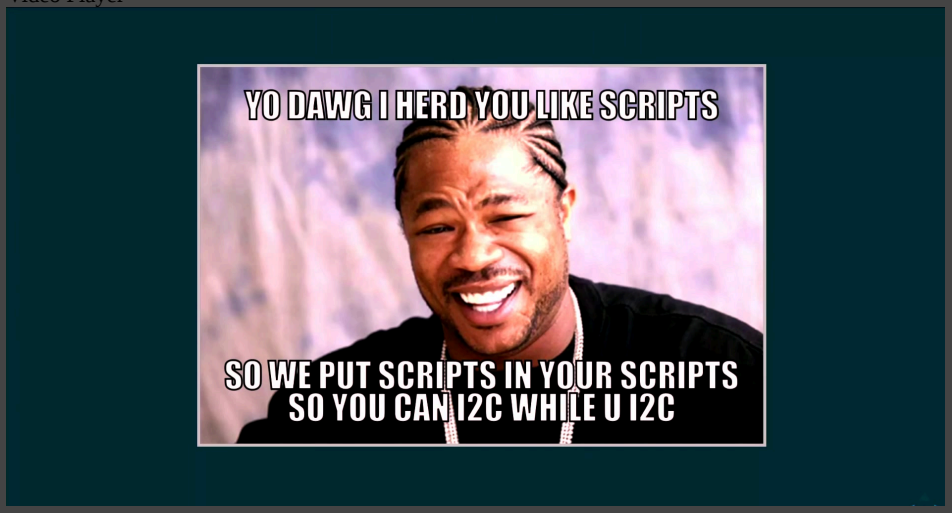
Console Hacking 2016: Postscript

By marcan

Filed under ps4

Another year, another console hacking talk! This talk picks off where our lighting talk last year left off, and goes into detail of how we ported Linux to the PS4.
If you haven't watched it, take a look before reading the rest of this post:

Video Player



Slides: Online · Download / source code

read more

2016-02-10

In Memoriam: Ben “bushing” Byer

By fail0verflow

Ben Byer
1980 – 2016

We are deeply saddened by the news that our member, colleague, and friend **Ben “bushing” Byer** passed away of natural causes on Monday, February 8th.

Many of you knew him as one of the public faces of our group, fail0verflow, and before that, Team Twiizers and the iPhone Dev Team.

Outspoken but never confrontational, he was proof that even in the competitive and oftentimes aggressive hacking scene, there is a place for both a sharp mind and a kind heart.

To us he was, of course, much more. He brought us together, as a group and in spirit. Without him, we as a team would not exist. He was a mentor to many, and an inspiration to us all.

Yet above anything, he was our friend. He will be dearly missed.

Our thoughts go out to his wife and family.

Keep hacking. It’s what bushing would have wanted.

read more

2015-12-30

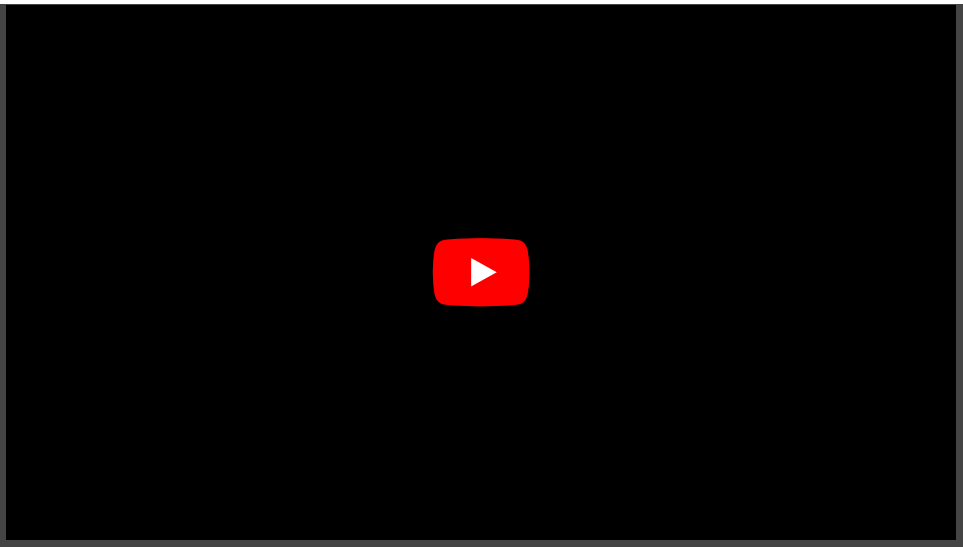
Console Hacking 2015: Liner Notes

By marcan

Filed under wiiu

If you’re here, you’ve probably heard about our lightning talk at the 32nd Chaos Communication Congress demoing Linux on a PS4. This post continues where the talk left off and clarifies a few aspects of what we’re doing, and why.

If you haven’t yet, please watch the talk before reading the rest of this post:



Slides: Online · Download / source code

read more

2014-12-31

31c3 CTF - pong (pwn30)

By marcan

Filed under 31c3CTF pwnable format string vulnerability

```
pong
pwn (30 pts)
------------------
To play it, connect to our server via:

socat -,raw,echo=0 TCP:188.40.18.92:2001
Have fun!
```

read more

2014-12-30

31c3 CTF - safelock (signals20)

By marcan

Filed under 31c3CTF electronics power analysis

```
safelock
Signals (20 pts)
------------------
This is the circuit of a safe lock. Get the key to open it!
http://188.40.18.86/safelock/

It's neither about webtronics nor ngspice. Disregard bugs in both.

If you want to write spice code directly, use something like this

cat test.cir | curl --data-binary '@-' http://188.40.18.86/safelock/contest_
```

read more

2014-09-04

HubCap: pwning the ChromeCast pt. 2

By axoltl

Filed under chromecast root USB hubcap

The chain In the last post, I explained the bug that we used to get a foothold into the system, but we're far from achieving what we want. We left off being able to overwrite anything before a particular buffer but because of caching behavior that didn't get us all that far. Additionally, we don't know exactly where we are in memory. I mentioned in the previous post that there's a debug port on the Chromecast that prints out messages from the boot loader.

read more

2014-08-29

HubCap: pwning the ChromeCast pt. 1

By axoltl

Filed under chromecast root USB hubcap

In case you're looking for the root, it was released a little while back: here The foothold I'd tell you all about what the Chromecast is, but I think Wikipedia has that part covered for me. For our purposes, all you need to know is that it's an ARMv7 based device, has WiFi, an HDMI connector and a maintenance port in the form of a micro USB port.

read more

2014-08-19

OpenVizsla OV3 - Hardware

By tmbinc (via debugmo.de)

Filed under OpenVizsla fail

(This is a guest post by tmbinc. You can read the original post here on debugmo.de.)

Fail.

That's probably the first word you think of when hearing the word "OpenVizsla". It all started good in - WTF - 2010 when bushing and pytey thought it would be a good idea to build an open-source USB sniffer.

Scam. That's what people called the project after unable to provide a working prototype after one year, two years, three years. But let me assure you: this project is not a scam. We just failed. A lot.

I don't want to swirl up the past - that was done before, but rather present the current state of affairs. TL;DR: It looks good, and OV3 actually a working USB analyzer these days, and it shipped to (almost) all of the original backers. Once all of them shipped, more of them will be sold to the public.

Let's start with a hardware overview of ov3, the third attempt to get it right. By the way, you can find everything in the openvizsla github repository. Just in case you want to build your own USB analyzer.

read more

2014-06-28

RFID hacking preamble: a new peak detection for the proxmark3

By iZsh

Filed under RFID hacking proxmark3 FPGA DSP peak detection verilog

Introduction Building upon the IIR filter from the last post we're now going to improve the edge/peak detection fpga module. Why The current algorithm is really simple, maybe too simple: whenever the ADC value is above (resp. below) a hardcoded value, it outputs 1 (resp. 0) with hysteresis. There are two problems with that: well first, the hardcoded values... For some reason no one has really complained about it so far (?

read more

2014-06-20

RFID hacking preamble: designing an FPGA IIR filter for the proxmark3

By iZsh

Filed under RFID hacking proxmark3 FPGA DSP IIR filter mkfilter butterworth verilog

Introduction At work, they recently replaced the coffee vending machine for a new one. One detail quickly piqued my interest though: you could now ask the front desk for an RFID token, pay your coffee with it instead of regular coins, and also deposit cash into the token at the vending machine. How does it work? Which RFID IC does it use? Can we play with it? The next few posts will narrate this journey.

read more

2014-05-19

Enhancing the AVIC-5000NEX - part 2

By bushing

Filed under avic avic-5000nex

img { max-width: 100%; height: auto; } Introduction This is the second part of a series of articles on reverse-engineering the Pioneer AVIC-5000NEX. to disable a nag screen; reading the previous post will give some helpful context. This post will focus on understanding the software and actually modifying it to change its behavior; a future post will cover the crafting of an update for other people to use.

read more

2014-05-12

Enhancing the AVIC-5000NEX

By bushing

Filed under avic avic-5000nex

Introduction This is the first of a several? part article on my adventures with my aftermarket in-dash navigation unit for my car, a Pioneer AVIC-5000NEX. I want to modify it to remove a nag screen. I will present five different ways of hacking it, explain how to analyze the system to modify its functionality, and hopefully end up constructing an update that other users can use to disable it on their cars.

read more

2014-04-27

plaidCTF 2014 - reeekeeeeee (web200)

By w3nk4w

Filed under plaidCTF2014 web python

```
reeekeeeeee
Web (200 pts)
------------
The Plague seems obsessed with internet memes, though we don't
yet know why. Perhaps there is a clue to what he's up to on this
server (epilepsy warning). If only you could break in....
Here is some of the source.
```

read more

2014-04-27

plaidCTF 2014 - rsa (for450)

By segher

Filed under plaidCTF2014 forensics cryptography rsa

```
rsa
Forensics (450 pts)
-------------
Our archaeologists recovered a dusty and corrupted old hard drive used by
The Plague in his trips into the past. It contains a private key, but this
has long since been lost to bitrot. Can you recover the full key from the
little information we have recovered?
```

You can download the recovered information here.

read more

2014-04-27

plaidCTF 2014 - wheeeee (crypto375)

By w3nk4w

Filed under plaidCTF2014 crypto python slide attack

```
wheeeee
Crypto (375 pts)
---------------
Although it seems like The Plague's messaging service is secure,
there are bound to be bugs in any 20th century crypto system.
We've recovered a version of the block cipher The Plague implemented.
Use their online encryptor tool, at 54.82.75.29:8193, to break the
cipher and figure out Plague's secret plans. NOTE: When the service
sends you a hex-encoded string, respond with a hex-encoded string.
```

read more

2014-04-25

plaidCTF 2014 - bbos (for350)

By aDR4eA

Filed under plaidCTF2014 Forensics strings BlackBerry

```
bbos
Forensics (350 pts)
-------------------

You have traveled back in time, but look, hunting  The Plague is tough.
You're really just going back to relax for a while  without having to
worry about all that nonsense. As you walk in the park  you stumble
across someone's BlackBerry. Wow, people still use  BlackBerry phones
(time travel gets so confusing)? You figure you should  return it to the
owner, but you have a hard time getting inside. Figure  out what's on
the phone, and maybe we'll be able to return it to the rightful owner.
```

BlackBerry was this fancy pager thing, right?

read more

2014-04-24

plaidCTF 2014 - graphs (crypto200)

By jix

Filed under plaidCTF2014 cryptography graphs sat

This challenge was about breaking a custom public key encryption system.

```
graphs
Cryptography (200 pts)
--------------
In this era, block ciphers hadn't even been invented. The Plague created thi
system based on problems he knew to be NP hard, but there must be something
can do to decode his messages.
```

We were given a python implementation of the system, the Plague's public key and an encrypted message. The implementation includes encryption, decryption (given a private key) and key generation.

read more

2014-04-23

plaidCTF 2014 - g++ (re200)

By aDR4eA

Filed under plaidCTF2014 Crypto templates preprocessor horror

```
Although it seems like The Plague's projects are  open source, it's not
quite so simple to figure out what the source code  does. We believe
this project is supposed to print out secret information, but the KEY
variable in the Makefile has been lost. Find the key, build the
project, get us the information.
```

Oh noes, the key is gone!

read more

2014-04-23

plaidCTF 2014 - parlor (crypto250)

By aDR4eA

Filed under plaidCTF2014 Crypto length extension

```
The Plague is running a betting service to build up funds for his massive
empire. Can you figure out a way to beat the house? The service is running
at 54.197.195.247:4321.
```

read more

2014-04-23

plaidCTF 2014 - zfs (for400)

By aDR4eA

Filed under plaidCTF2014 Forensics strings

```
zfs
Forensics (400 pts)
-------------------
The Plague is using state of the art systems for storing his data.
Our operatives managed to steal a drive from one of his servers,
but it seems like our haste may have led to some uber-corruption.
Can you get the data off the drive to track down The Plague?
```

Sure we can. But where do we start?

read more

2014-04-23

plaidCTF 2014 - bronies (web800)

By blasty comex

Filed under plaidCTF2014 pwning pony XSS CSRF fortify buffer overflow web

```
bronies
Web (800 pts)
------------------
We are trying to break into eXtreme Secure  Solutions, where The
Plague works as a system adminstrator.  We have found that their
internal company login page is at
http://portal.essolutions.largestctf.com/. Recon has also revealed
that  The Plague likes to browse this site during work hours:
```

```
http://54.196.225.30/ using the username ponyboy2004.  Remember, our
main target is to break into the company portal, *not* the pony site.
```

read more

2014-04-22

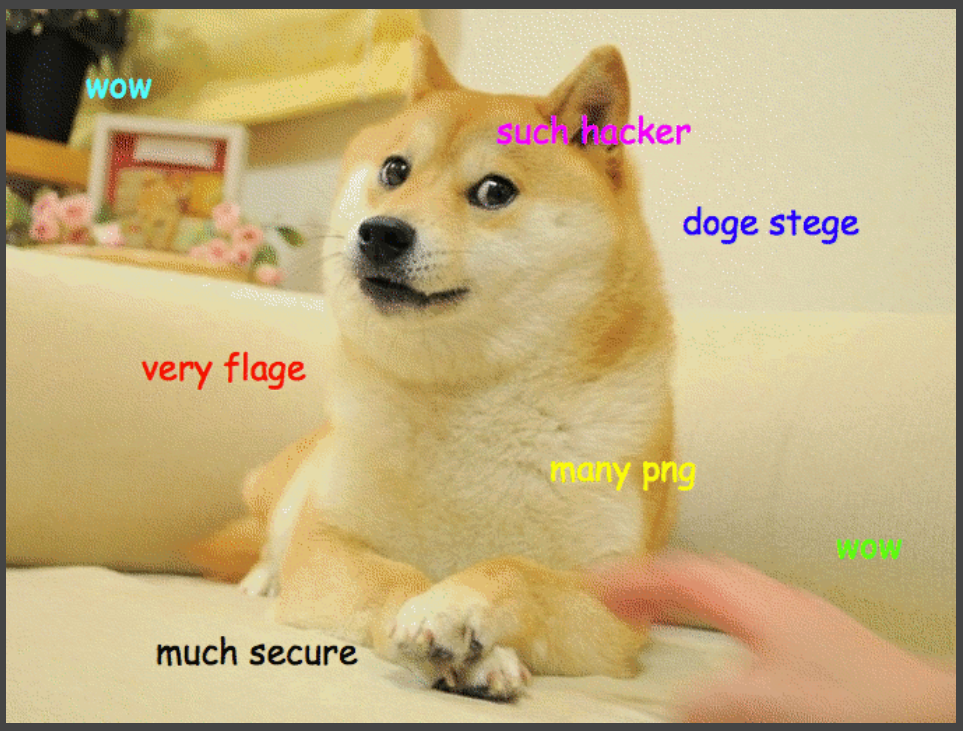
plaidCTF 2014 - doge_stege (for100)

By jix

Filed under plaidCTF2014 forensics steganography

This challenge was about extracting a (not very well) hidden message out of an image file:

```
doge_stege
Forensics (100 pts)
--------------
You were startled to learn the The Plague has been behind many of the
most popular internet memes. We believe he hides information in these
funny pictures with steganography in order to broadcast his messages
through time without detection. Find the hidden message, stop the
signal.
```



Obvious Stego is Obvious

The first thing to do with every file you get from a CTF challenge is to run the `file` command on it:

```
% file doge_stege.png
doge_stege.png: PNG image data, 680 x 510, 8-bit colormap, non-interlaced
```

read more

2014-04-22

plaidCTF 2014 - rendezvous (misc250)

By jix

Filed under plaidCTF2014 tor

This challenge was about establishing a connection to a hidden tor service which is rather picky in accepting connections. We were given the following description:

```
rendezvous
Misc (250 pts)
--------------
The Plague has a friend called Alice who has some secrets on a tor
service (http://6c4dm56aer6xn2h2.onion/). We think if we can talk to
her, we can learn some useful things about The Plague. Unfortunately
she will only rendezvous with "chandler" when he brings a cookie with
"beef" baked into it. Can you help us find her secret?
```

Getting Started

The first thing we did was of course trying to connect to the service. Whether using a tor to web gateway as for example onion.to or a local tor instance, the result was the same: no connection could be established. Using `curl -v –socks5-hostname localhost:9050 http://6c4dm56aer6xn2h2.onion/` showed that curl didn't even send the request, confirming that the problem is at the tor layer and not at the HTTP layer. Thus getting a tor connection to the hidden service is actually part of the challenge.

read more

2014-04-22

plaidCTF 2014 - ezhp (pwn200)

By iZsh

Filed under plaidCTF2014 reversing pwning linux python buffer overflow heap overflow moneyshot

```
ezhp
Pwnables (200 pts)
------------------
Luckily when you travel back in time, you still get to use all your
knowledge from the present. With that knowledge in hand, breaking
into this service (at 54.81.149.239:9174) owned by The Plague
shouldn't be hard at all.
```

To set the picture, let's identify the binary

```
izsh@box:~$ file ezhp
ezhp: ELF 32-bit LSB executable, Intel 80386, version 1 (SYSV),
dynamically linked (uses shared libs), for GNU/Linux 2.6.24,
BuildID[sha1]=0x5fa5bd76db306497b549ea3b0466cd9e9afa2705, stripped

izsh@box:~$ readelf -l ezhp | grep STACK
    GNU_STACK      0x000000 0x00000000 0x00000000 0x00000 0x00000 RWE 0x4
```

read more

2014-04-21

plaidCTF 2014 - tiffany (re300)

By marcan

Filed under plaidCTF2014 reversing linux ptrace

```
tiffany
Reversing (300 pts)
-------------------
We want to get access to a server used by The Plague. Maybe if you
can find out what key is accepted by this binary you can find out
where or when The Plague is...
```

Yay, a Linux x86_64 executable! Let's run it and see what happens, because what could possibly go wrong when running a random binary off the internet?

```
$ ./tiffany
This may take a while...
.......
Please enter a string: TEST
....
Sorry, wrong.
```

Well, that took 3 seconds to initialize and 5 seconds per input string character. Sure seems to be doing a lot of stuff. Let's load it into IDA to get a general idea.
read more

2014-04-21

plaidCTF 2014 - paris (re300)

By marcan

Filed under plaidCTF2014 reversing seh windows vm

```
paris
Reversing (300 pts)
------------------
This binary was found on some of our Windows machines. It's got The
Plague written all over it. What secrets are contained inside?
```

We are greeted by a Windows executable. Since I hate Windows and I can't be arsed to pull up a Windows VM and debugger, I decided to solve this one statically. Time to load it into IDA.
read more

2014-04-20

plaidCTF 2014 - __nightmares__ (pwn375)

By marcan

Filed under plaidCTF2014 pwning python sandbox

```
__nightmares__
Pwning (375 pts)
------------------
The Plague is building an army of evil hackers, and they are starting
off by teaching them python with this simple service. Maybe if you
could get full access to this system, at 54.196.37.47:9990, you would
be able to find out more about The Plague's evil plans.
```

This server simply evaluates any Python expression provided - with an attempt at sandboxing it.
read more

2014-04-20

plaidCTF 2014 - freya (misc250)

By sven

Filed under plaidCTF2014 wireshark https SSL Kerberos SSH

This challenge is part of the misc category:

```
freya
Misc (200 pts)
------------------
We've traveled back far, but this protocol looks familiar...
Our reconnaissance team did a great job, they got us a data capture
from the currently running systems and a private key
```

```
from the server (shell.woo.pctf which resolves to 54.226.73.167).
Take  a look at the traffic our reconnaissance team picked up, and see if yo
can get access to The Plague's server, at 54.226.73.167.
```

with the following four files:

- freya.pcapng
- freya_cert.pem
- fre[illegible]
- password

The task is pretty simple - somehow get access to shell.woo.pctf, probably by using ssh.

read more

2014-04-19

plaidCTF 2014 - curlcore (for250)

By iZsh

Filed under plaidCTF2014 Forensics wireshark https SSL

Last week we played plaidCTF with Eindbazen under the name 0xffa (can you figure out why that name?). Write-ups are mandatory in the rules, so let's start with an easy one :-)

```
curlcore
Forensics (250 pts)
-------------------
We managed to grab a memory dump off of The Plague's computer while
he was making a secure download. We think he may have been looking
for new places to hide the Prime Factorizer. Can you figure out what
messages were sent through his computer?
```

For this challenge, you get 3 files:

- capture (a network capture)
- corefile (a memory dump)
- coremaps (the process's memory map)

and the shell script which helped generating those files

```
#/bin/sh

sudo rm /tmp/capture 2>/dev/null
sudo dumpcap -i eth0 -w /tmp/capture &
DUMPCAPPID=$!

sleep 1
OUTPUT="`/usr/bin/env -i /bin/dash -c 'ulimit -c unlimited; curl -k https://c
sleep 1

sudo kill -INT $DUMPCAPPID
wait

sudo chown `whoami` /tmp/capture

echo "$OUTPUT"

sudo mv "`echo "$OUTPUT" | grep -o 'Saved corefile .*$' | cut -c 16-`" /tmp/c
sudo chown `whoami` /tmp/corefile


echo "$OUTPUT" | awk '/Mapped address spaces/,/(gdb)/' | grep -v '(gdb)' > /t

rm /tmp/curlcore.tgz 2>/dev/null
tar czf /tmp/curlcore.tgz `grep -o ' /.*$' /tmp/coremaps | sort -us | tr '\n'
```

Since we have a network capture of the https download, we need to find a way to decrypt the SSL communication...

read more

2014-01-02

Console Hacking 2013: Omake

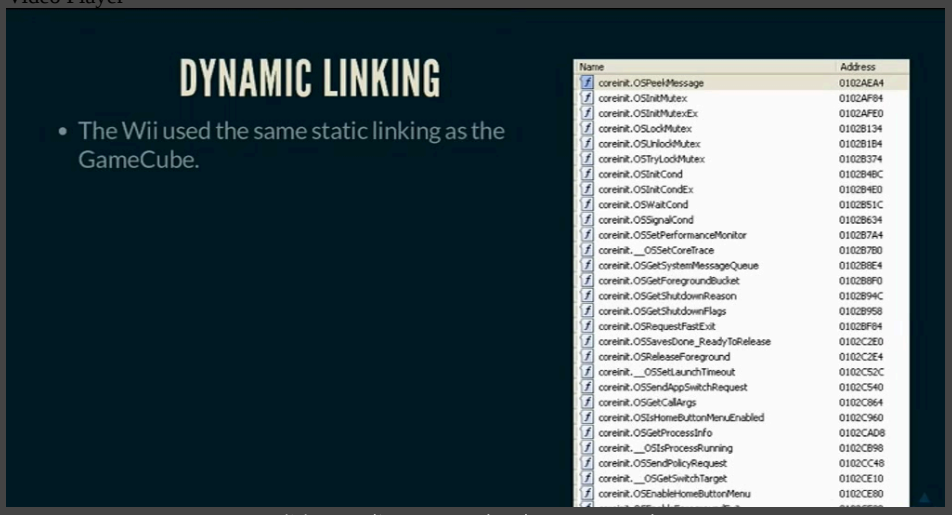
By marcan

Filed under wiiu

As you're likely aware, our team gave a lecture at the 30th Chaos Communication Congress on hacking the Wii U. This blog post is a follow-up to the talk and contains clarifications, corrections, and material that we couldn't fit in the one-hour time slot.

If you haven't yet, please watch the talk before reading the rest of this post:

Video Player



Slides: Online · Download / source code

read more

2013-05-06

The future of console homebrew (and a shot of Espresso)

By marcan

Filed under wiiu

It's been almost 7 years since the Wii was released. Back in 2006, not many owned a living room PC. PCs were still relatively bulky, and the Chinese offerings were limited to horrible media players. At the time, the prospect of having a game console double as a HTPC and being able to browse the web, play games for older platforms with emulation, and run homebrew games on a device which you already had in the living room was rather appealing.

Fast forward to today. Mobile SoCs have made huge advances - you can get a quad-core chip in a phone these days - and have made the jump to the living room. Spend $25 and you can get a Raspberry Pi, which is about on par with the Wii at $^1/_{10}$ of the launch price and 1/7th of the power consumption (with HD video). Spend $100 and you can get an Ouya, which beats the Wii U's CPU and doesn't have too shabby graphics at one third the cost. These mobile-derived devices aren't quite a replacement for game consoles yet, but they're catching up fast. They're cheap enough that they're almost disposable. The software ecosystem is much larger and wider than any console has ever had. More importantly, they're open, and the development tools and environments are way better for open development than any game console ever was.

read more

2013-01-23

Megafail

By marcan

Let's take a break from Wii U hacking to take a quick look at Mega's security.

In case you've been living under a rock the past few days, Kim Dotcom (of Megaupload infamy) has launched his new cloud storage site, Mega. Mega has an impressive sales pitch, promising secure cloud storage where only the user has the key to decrypt his or her files, and the encryption and decryption happens securely in the browser.
Today we aren't going to take a look at their encryption or their key generation, which have already been the subject of several articles. Instead, we're going to look at the security of the Mega website itself. As Mega themselves admit, if you use their web interface (and not a third-party client), the security of the entire ordeal depends on whether you trust them. After all, anyone with the ability to modify the site could just replace the JavaScript code with one that sends them (or anyone else) your password or master key. There's no way around having to trust Mega for this, but you also have to trust that Mega's site is delivered securely to you.
read more

2013-01-02

Clarification

By fail0verflow

It has come to our attention that nobody seems to have any idea what the past 4 posts have been about. In an attempt to clarify things, we have prepared a handy diagram:
read more

2012-12-30

30 Days and a Congress

By fail0verflow

Brought to you by 30 hackers and 3 tables:

2b30b703c6676c8124c7347b30c7972ffeae2b39

6a0b87fc98b306ae3366f0e0a88d0b06a2813313

read more

2012-12-14

14 Days

By fail0verflow

ee28d0be718055423ee79d89889ebe386e5b0c2d

This one didn't even require impairing drugs, all it took was asking nicely.
read more

2012-12-11

11 Days

By fail0verflow

d6356c408f36a4bf4b48abbee5bfffe91d196ee6

Devices can get drunk too. Fewer bytes, twice the fun!
read more

2012-12-08

8 Days

By fail0verflow

3d331b3165f9638c6cd6221702b2f736f7fcf931

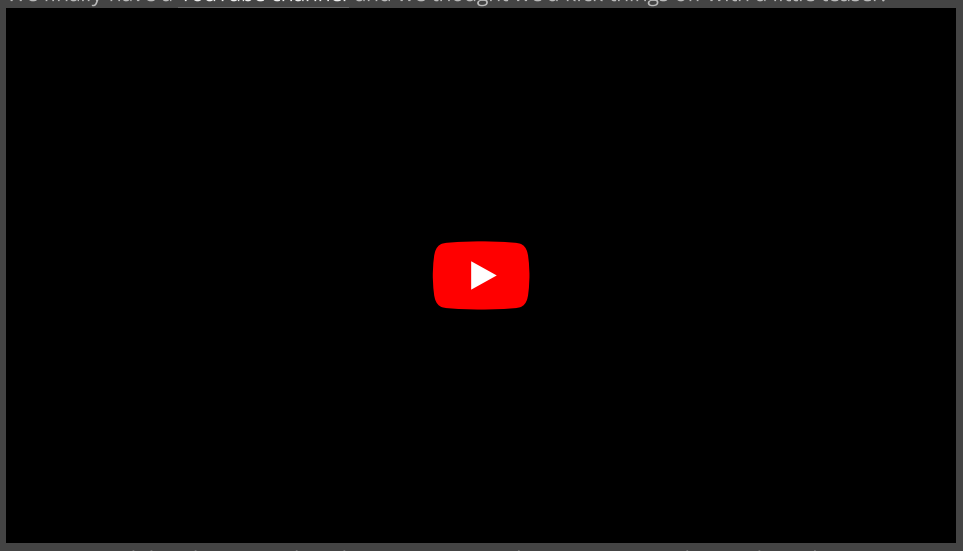
read more

2012-11-30

Wii U Teaser

By fail0verflow

We finally have a YouTube channel and we thought we'd kick things off with a little teaser:



Keep in mind that this is purely a demonstration at this stage. Depending on how things progress and what direction development takes, we may or may not release something like this in this form. Please don't ask for release dates. We'd rather spend time investigating the new system than putting together a release that may or may not end up being the Right Way to do things in the future ;).

read more

2012-07-05

CVE-2012-0217: Intel's sysret Kernel Privilege Escalation (on FreeBSD)

By iZsh

Filed under vulnerability exploit FreeBSD

CVE-2012-0217 was reported by Rafal Wojtczuk but ironically, it was fixed for Linux in 2006 as shown by CVE-2006-0744 without receiving much attention.

It is quite an interesting vulnerability on many aspects. Among them, and thanks to its hardware basis, it impacts many operating systems. For instance, as long as they run on a Intel processor in long mode (obviously), FreeBSD, NetBSD, Solaris, Xen and Microsoft Windows have been reported to be vulnerable. This therefore gives us quite an incentive to develop an exploit ;).

If you haven't yet read Xen's blog post The Intel SYSRET privilege escalation please do because we won't go again into too much details about the vulnerability itself.

Without further delay, let's dig right into the FreeBSD exploitation!

read more

2012-04-09

DCPU-16 Review

By marcan

I've always liked the idea of building complex logic systems out of a simple primitive that is just powerful enough to construct all logic - particularly in videogames. For example, in LittleBigPlanet, you can build a Tic-Tac-Toe AI out of physical elements like pistons and magnetic switches. In Minecraft, you can build an ALU out of a primitive construction element that behaves, essentially, as a NOT gate. And, if games aren't your thing, you can build CMOS logic out of UNIX pipes with a simple "mosfet" program.

Just a few days ago, Notch, the creator of Minecraft, revealed a new game, $0x10^c$. Instead of giving players a simple logic element, it will include a full-blown 16-bit CPU that can be programmed. I find this intriguing, because it allows for much more complex development yet it

doesn't step right into the boring world of "let's just throw in a lua scripting engine and call it a day". You're still limited by emulation speed and by memory constraints.

read more

2012-03-21

AT&T Microcell FAIL

By c1de0x

Filed under fail femto

One of the things we've been playing with recently is the AT&T Microcell. This device is intended to provide a cheap way for AT&T to increase their network coverage at the expense of their customers. The device is essentially a small cell-tower in a box, which shuttles your calls and data back to the AT&T mothership over your home broadband connection.

This kind of device is becoming more and more popular with the various mobile providers. They are commonly known as residential femtocells.

We're curious. We love gadgets. We love to take gadgets apart and see what makes them tick. So naturally, we've taken a look at a number of different femtocells.

We finally got around to looking at this AT&T variant this week, and discovered that it is totally full of fail.

read more

2012-03-02

Unprogramming: Intro

By fail0verflow

Filed under unprogramming reversing challenge

On Friday, the 13th of January 2012, the ACM Queue published an article by Poul-Henning Kamp entitled 'The CRYPO-CS-SETI challenge: An Un-programmng challenge'. In this post, Kamp challenged his readers to attempt to disassemble a program for an unknown computer. In what we assume was an attempt at increased dramatic impact, he described a scenario where part of an extra-terrestrial computer is discovered, with only a memory storage device intact.

We first heard of the challenge on the morning of Saturday the 14th, and thought it sounded like fun. Within five days we had completely disassembled the program. In addition, we had accidentally identified the oh-so-terrestrial source of the code.

This is the first in a series of posts in which we'll describe how we went about reverse-engineering the machine architecture using nothing but the binary blob and our wits.

read more