blackhať USA 2019

AUGUST 3-8, 2019

MINimum Failure Stealing Bitcoins with Electromagnetic Fault Injection Colin O'Flynn C.T.O., NewAE Technology Inc. Assistant Professor, Dalhousie University

BHUSA YBLACKHATEVENTS

My Dual Life

- C.T.O @ NewAE Technology Inc.
 - NewAE produces embedded tooling for hardware security validation
 - 700+ customers in 44 countries.
 - Started the open-source ChipWhisperer project.
- Assistant Professor @ Dalhousie University
 - Part of Electrical & Computer Engineering Dept.
 - Working on cybersecurity research + with local embedded startups.



The Talk With it All!! In no particular order:

- A true history of block chain!
- Me gambling with 0.3 BTC!
- Physical give-aways!
- Attacking bitcoin wallets & stealing funds!
- Stealing authentication credentials!
- Crass commercialization!
- Open-Source tool release!
- Fixing your crappy code!

Blockchain Background (1/2)

tome > Building Materials > Concrete, Cement & Masonry > Concrete Materials, Tools & Accessories > Concrete Blocks & Bricks > 1000150633



Oldcastle 8-inch Wall Block Model # MPS6 | Store SKU # 1000150633

\$3.58 / each

Sold In-Store Only

246 In Stock at KINGSTON

FREE PICK UP IN-STORE

Pick Up: Today

Blockchain Background (2/2)



Example: Blockchain Secures Voting Booth

E

History



- May 11, 2008: First concrete implementation. (H. Simpson)
- Oct 31, 2008: More wellknown computer implementation. (S. Nakamoto)

• A true history of block chain!

- Me gambling with 0.3 BTC!
- Physical give-aways!
- Attacking bitcoin wallets & stealing funds!
- Stealing authentication credentials!
- Crass commercialization!
- Open-Source Tool Release!
- Fixing your crappy code!

Let's Embed Those Blockchains



Why a Hardware Wallet?



Trezor Features

Security measures of Trezor.

Ο

Firmware verification.

The bootloader verifies the firmware signature. The device only runs if the firmware is correctly signed by SatoshiLabs. Otherwise, the Trezor warns you.

Ст

Protected key operations.

Operations with private and public keys are only allowed after user authentication via PIN.

${igsidentsizeticontrolseptilestyle{\controlseptilestyle{\controlseptilestyle{\controlseptilestyle{\controlseptile{\controls$

Additional passphrase support.

Trezor supports **BIP39 passphrases**, which are never stored or remembered on the device.

Ю

Reliable backup & recovery.

Your recovery seed protects you against theft, loss or destruction of your device. Simply restore the recovery seed, and your wallet is back.

₫

Ultrasound hardware seal.

Trezor hardware case is ultrasonically welded, making it difficult to be restored after breakage.

۵

Secure update procedure.

The bootloader erases memory on firmware update and resonly if the firmware signature is value of the firmware signature.

¢

Write-protected bootloader.

The bootloader is write protected, safeguarded by the Memory Prote

Learn more about security

Jour Security





Example: Ultrasonic Seal

<Hopefully my camera works>

Why Pick on Trezor?

- Trezor is Open Source
 - Anyone can validate the source code, modify it, etc.
 - Lessons learned on Trezor can be applied outside of bitcoin wallets.
 - Embedded systems, IoT, automotive, etc.

• This problem I'm disclosing has been fixed with issued firmware patch.

About Bitcoin Recovery Seeds



From Trezor documentation:

Understanding the recovery seed.

The recovery seed is a crucial element for the security of your Trezor hardware wallet. If your device is lost, damaged or stolen, you can use your **recovery seed to restore access to your entire wallet, passwords and other data** associated with it. The process is simple; you only have to enter the words of your seed into your new Trezor device. (You may also use any other wallets or applications that use the same standard as the Trezor.)

Important Caveat: You can also password-protect this seed, but it's not done by default. If password protected, the attacks I'm going to describe don't work!

What's inside the Trezor?



How to Get Recovery Seeds?

35C3 PRESENTATION





Dmitry Nedospasov

Dmitry Nedospasov is a hardware design and security engineer, security researcher, trainer, speaker and reverse-engineerer. In 2014 Dmitry received his PhD (Dr-Ing.) in IC Security at TU Berlin.

y in



Josh Datko

Josh Datko is an embedded systems engineer, security researcher and former submarine officer. Josh is best known for his <u>2017 presentation</u> on insecurities in cryptocurrency hardware wallets.





Thomas Roth

Thomas Roth was named as one of the <u>30 under 30 in Technology</u> by the Forbes Magazine. His main focus is on mobile and embedded systems with published research on topics like TrustZone, payment terminals, and embedded security.

🤊 in

Wallet.Fail

- Vulnerabilities on several Bitcoin Wallets.
- For Trezor specifically:
 - How to copy recovering seed out of a backup stored in SRAM.
 - Required physical access to PCB (open enclosure), but can be performed with high reliability.

My Inspiration – Wallet.Fail Talk



How else to get recovery seed?



27 flash memory layout:

29	name	I	range	size	function
30 31	Sector	0 0x08000	0000 - 0x08003FF	+ F 16 KiB	bootloader code
32	Sector	1 0x08004 What's g	000 - 0x08007FF	F 16 KiB	bootloader code
33					
34	Sector	2 0x08008	3000 - 0x0800BFF	F 16 KiB	metadata area
35	Sector	3 0x08000	000 - 0x0800FFF	F 16 KiB	metadata area
36		Recovery	seed, device PIN sa	ved here!	-+
37	Sector	4 0x08010	000 - 0x0801FFF	F 64 KiB	application code
38	Sector	5 0x08020	0000 - 0x0803FFF	F 128 KiB	application code
39	Sector	6 0x08040	0000 - 0x0805FFF	F 128 KiB	application code
40	Sector	7 0x08060	0000 - 0x0807FFF	F 128 KiB	application code
41		===+=======		==+========	
42	Sector	8 0x08080	0000 - 0x0809FFF	F 128 KiB	application code
43	Sector	9 0x080A0	0000 - 0x080BFFF	F 128 KiB	application code
44	Sector	10 0x080C0	0000 - 0x080DFFF	F 128 KiB	application code
45	Sector	11 0x080E0	0000 - 0x080FFFF	F 128 KiB	application code

USB Descriptors

File Options Help Image: Control of the system	USB View	
Image: Book State	File Options Help	
Image: Second state of the second s	File Options Help Image: Constraint of the system of the syst	External Hub: USB#VID_2109&PID_2812#7&3932ea2a&0&1#{f18: A Hub Power: Self Power Number of Ports: 4 Power switching: Individual Compound device: No Over-current Protection: Individual Device Descriptor: bcdUSB: 0x0210 bDeviceClass: 0x00 bDeviceClass: 0x00 bDeviceProtocol: 0x01 bMaxPacketSize0: 0x40 (64) idVendor: 0x2812 bcdDevice: 0x9090 bcdDevice: 0x9090
Imanufacturer: 0x01 Imanufacturer: 0x02 Imanufacturer: 0x00 Imanufacturer: 0x00 Imanufacturer: 0x01 Imanufacturer: 0x01 Imanufacturer: 0x01 Imanufacturer: 0x01 Imanufacturer: 0x00 Imanufacturer: 0x00 Imanufacturer: 0x01 Imanufacturer: 0x00 Imanufacturer:	E W [Loca1] [PIP 6&1921ddad&0] DeviceConnected : U: Control [PIP 7&3932ea2a&0] DeviceConnected Control [PIP 7&3932ea2a&0] DeviceConnected Control [PIP 7&3757626&0] DeviceConnected 	<pre>iManufacturer: 0x01 iFroduct: 0x02 0x0409: "USB2.0 Hub " iSerialNumber: 0x00 bNumConfigurations: 0x01 ConnectionStatus: DeviceConnected Current Config Value: 0x01 Device Bus Speed: High Device Address: 0x08 Open Pipes: 1 Endpoint Descriptor: bEndpoint Address: 0x81 IN Transfer Type: Interrupt wMaxPacketSize: 0x001 (1) bInterval: 0x0C</pre>
Configuration Descriptor: TotalLength: 0x00 Ox00 Substrate State	Locad [IPIP 68125e606580] DeviceConnected : U: Locad [IPIP 68125e606580] DeviceConnected : USB Input Device Locad 2] DeviceConnected : USB Composite D	Configuration Descriptor: WTotalLength: 0x0019 bNumInterfaces: 0x01 bConfigurationValue: 0x01 iConfiguration: 0x00 bmAttributes: 0xD0 (Bus Powered Self Powered Reg MaxPower: 0x00 (0 Ma) Interface Descriptor: bInterfaceNumber: 0x00 bAlternateSetting: 0x00 bAlternateSetting: 0x00 bMumEndpoints: 0x01 bInterfaceSubClass: 0x00 bInterfaceProtocol: 0x00 iInterface: 0x00

wLength → Host Provided Max Request Size

9.4.3 Get Descriptor

• This request returns the current device configuration value.

	bmRequest Type	bRequest	wValue		windex		wLength		
ytes	0	1	2	3	4	5	6	7	

Direction 0x80	GET DESCRIPTOR	Descriptor Type (HI) and Descriptor Index (LO)	Zero or Language ID (9.6.7)	Descriptor Length
1: D-to-H 0: Device	6	High byte: Descriptor Types 1: DEVICE 2: CONFIGURATION 3: STRING 4: INTERFACE 5: ENDPOINT 6: DEVICE_QUALIFIER 7: OTHER_SPEED_CONFIG 8: INTERFACE_POWER	String Descriptors: Language ID Others: Zero	The number of bytes to return.
Data Descriptor		Low Byte: Descriptor Index		

Send MIN() of wLength & Struct Length

} else if (((req->bmRequestType & USB_REQ_TYPE_RECIPIENT) == USB_REQ_TYPE_INTERFACE) &&
 (req->wIndex == WINUSB_REQ_GET_EXTENDED_PROPERTIES_OS_FEATURE_DESCRIPTOR) &&
 (usb_descriptor_index(req->wValue) == winusb_wcid.functions[0].bInterfaceNumber)) {

```
*buf = (uint8_t*)(&guid);
```

*len = MIN(*len, guid.header.dwLength);

status = USBD_REQ_HANDLED;

Checking Implementation Details



Open Source FTW :)

- I can be lazy since firmware is fully known & I can modify it even.
- We can 'simulate' the glitch to ensure things will work as we expect.



Validating This Will Work (1/2)

Index	m:s.ms.us.ns	Len E	Err De	v Ep	R	Record	Summary
0	0:00.000.000.000					Capture started (Aggregate)	[02/06/19 00:45:55]
1	0:00.000.000.000					Most connected>	
2	0:00.000.633.500					P <full-speed></full-speed>	
3	0:23.658.183.950	146 B	22	2 00	Þ	Ontrol Transfer	92 00 00 00 00 01 05 00 01 00 8
24	0:06.791.576.583	146 B	22	2 00	Þ	Ontrol Transfer	92 00 00 00 00 01 05 00 01 00 8
45	0:03.879.450.166	146 B	22	2 00	Þ	Ontrol Transfer	92 00 00 00 00 01 05 00 01 00 8
66	1:58.972.722.583	65535 B	22	2 00	Þ	Ontrol Transfer	92 00 00 00 00 01 05 00 01 00 8
4171	0:11.333.695.616					Capture stopped	[02/06/19 00:48:40]

Expected response (146 bytes)

 Use debugger to skip MIN() check.

Validating This Will Work (2/2)

File Disk E	dit View Struct	ures Crypto Mac	ro Help			
		1 🔑 📞	🔏 [+]	0 + +		
	FF FF FF FF	FF FF FF FF	FF FF FF FF	FF FF FF FF	FF FF FF FF	<u> </u>
00001584	FF FF FF FF	FF FF FF FF	FF FF FF FF	FF FF FF FF	FF FF	<u> </u>
	FF FF FF FF	FF FF FF FF	FF FF FF FF	FF FF FF FF	FF FF FF FF	<u> </u>
00015BA	FF FF FF FF	FF FF FF FF	FF FF FF FF	FF FF FF FF	FF FF	999999999999999999999999
0001500	FF FF FF FF	FF FF FF FF	FF FF FF FF	FF FF FF FF	FF FF	<u> </u>
000015DE 000015F0	FF FF FF FF	FF FF FF FF	FF FF FF FF	FF FF FF FF	FF FF	<u> </u>
00001602	FF FF FF FF	FF FF FF FF	FF FF FF FF	FF FF FF FF	FF FF	\$
00001614	FF FF FF FF	FF FF FF FF FF FF FF FF	FF FF FF FF FF FF FF FF	FF FF FF FF FF FF FF FF	FF FF FF FF	<u> </u>
00001638	FF FF FF FF	FF FF FF FF	FF FF FF FF	FF FF FF FF	FF FF	999999999999999999999999
0000164A	FF FF FF FF	FF FF FF FF	FF FF FF FF	FF FF FF FF	FF FF	<u> </u>
0000165C	FF FF FF FF	FF FF FF FF	FF FF FF FF	FF FF FF FF	FF FF	<u> </u>
00001680	FF FF FF FF	FF FF FF FF	FF FF FF FF	FF FF FF FF	FF FF	<u> </u>
00001692 000016A4	FF FF FF FF	FF FF FF FF	FF FF FF FF	FF FF FF FF	FF FF	999999999999999999999999
000016B6	FF FF FF FF	FF FF FF FF	FF FF FF FF	FF FF FF FF	FF FF	\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$
000016C8	FF FF FF FF	FF FF FF FF	FF FF FF FF	FF FF FF FF FF FF FF FF	FF FF FF FF	99999999999999999999999
000016EC	FF FF FF FF	FF FF FF FF	FF FF FF FF	FF FF FF FF	FF FF	9999999999999999999999999
000016FE	FF FF FF FF	FF FF FF FF	FF FF FF FF	FF FF FF FF	FF FF	999999999999999999999
00001710	00 00 00 00	00 00 00 00				TRZROO-
00001734						
00001746						
0000176A						
0000177C						
0000170L						
000017B2						
000017C4 000017D6						
000017E8						
000017FA 0000180C		00 00 00 00 73 74 6F 72	00 00 00 00 00 E5 BC 7E CE	00 00 00 00 42 ED BE 58	00 00 CO 52	storő ½/LÎBÍ/XÅB
0000181E	44 AE 0D 0A		00 00 00 00		00 00	D®
00001842						
00001866						
UUUU1878 0000188A						
0000189C						
000018AE	65 78 65 72 6F 65 20 73	63 69 73 65 68 61 74 65	20 6D 75 73 20 60 69 74	63 6C 65 20 61 72 64 20	74 GF	exercise muscle to
000018D2	69 67 67 65	72 20 68 6F	73 70 69 74	61 6C 20 77	65 61	igger hospital wea
000018E4	70 6F 6E 20	76 6F 6C 63	61 6E 6F 20	72 69 67 69	64 20	pon volcano rigid
	6B 20 6F 75	74 65 72 20	70 6C 61 63	65 20 73 70 65 20 6C 6F	67 69	k outer place logi
0000191A	63 20 6F 6C	64 20 61 62	61 6E 64 6F	6E 20 61 73		c old abandon aspe
0000192C 0000193E	63 74 20 73 72 79 20 62	6B 69 20 73 6C 61 73 74	70 61 72 65 20 6C 61 6F	20 76 69 63 67 75 61 67	74 6F 65 00	ct ski spare victo rv blast language
00001950						
00001986						
00001998						
000019AA 000019BC				79 20 54 72		My Trez
000019CE	6F 72 00 00					
000019E0 000019E2						
00001A04						
00001A16						
00001A28 00001A3A						
00001A4C						
UUUUTA5E	100 00 00 00					

STICY H

27

Generated and Induced Magnetic Field



Example of Coils





Triggering EMFI

	0040021	0.00.024.202.000	140.0	4		· 🚽 Oonaor Hansion	JE 00 00 TF 00 01 00 00 00 00 00 00 00 00 00 00 ZE 00 TF 00 00 00
FS 🏶	6546642	0:00.056.668.166	146 B	2	28 00	Control Transfer	92 00 00 00 00 01 05 00 01 00 88 00 00 07 00 00 02 A 00 44 00 65 00
FS 🏶	6546643	0:00.000.000.000	8 B	2	28 00	SETUP txn	C1 21 00 00 05 00 FF 1A
FS 🏶	6546647	0:00.000.025.333	64 B	2	28 00	🖻 🗐 IN txn	92 00 00 00 00 01 05 00 01 00 88 00 00 07 00 00 00 2A 00 44 00 65 00
FS 🏶	6546651	0:00.000.070.750	64 B	2	28 00	🖻 🗐 IN txn	00 00 7B 00 30 00 32 00 36 00 33 00 62 00 35 00 31 00 32 00 2D 00 38 00
FS 🏶	6546655	0:00.000.071.083	18 B	2	28 00	🖻 🗐 IN txn	39 00 64 00 38 00 65 00 66 00 35 00 7D 00 00 00 00 00
FS 🏶	6546659	0:00.000.026.333	0 B	2	28 00	OUT txn	
FS 💲	6546663	0:00.025.202.500	8 B	T 2	28 00	4 🧊 SETUP txn	C1 21 00 00 05 00 FF 1A
FS 🏶	6546664	0:00.000.000.000	3 B	2	28 00	SETUP packet	2D 1C B8 Packet Match
FS 🏶	6546665	0:00.000.003.416	11 B	2	28 00	1010 DATAO packet	C3 C1 21 00 00 05 00 FF 1A 83:d when the selected Match will be asserted when the selected
FS 🏶 🚺	6546666	0:00.000.008.666	1 B	2	28 00	 ACK packet 	D2 and endpoint PID, device address, endpoint, and data
FS 💲	6546667	0:00.000.013.166	1.99 s	2	28 00	🥩 [41215 IN-NAK]	[Periodic Timeout] pattern match.
FS 💲	6546668	0:02.000.005.333	1.99 s	2	28 00	[41201 IN-NAK]	[Periodic Timeout] Output Pin 4: Active High
FO	0540000		1.00			Charles and and a	Trioper on Matchy

- TotalPhase Beagle 480
- Hardware trigger on USB physical-layer packets!





-

St.

-

.

ChipWhisperer -

Target

PULS

·JUL VO

6.0

SCW520

Serial Ho. D43854F00154

0

.

0

0

USB Switch (hard reset required due to hard fault vectors)

BR

PhyWhisperer-USB

- Cheapish (\$250 USD) triggering device for USB physical-layer packets.
- Works as a sniffer too.
- Open-source HW/SW.
- Needs separate fault injection driver -ChipSHOUTER (EMFI), ChipWhisperer (Voltage glitching), or Mux (Voltage glitching)



https://github.com/newaetech/phywhispererusb 33

PhyWhisperer-USB

- USB 2.0 LS/FS/HS Phy
- Can switch power on/off to target (critical for glitching).
- Interpose with a real (external) USB host.

Stuff that works also:

 Sniffing USB packets (Wireshark as front-end).

Stuff you *could* do (but I'm way too lazy for):

• Generate USB packets.



- A true history of block chain!
- Me gambling with 0.3 BTC!
- Physical give-aways!
- Attacking bitcoin wallets & stealing funds!
- Stealing authentication credentials!
- Crass commercialization!
- Open-Source Tool Release!
- Fixing your crappy code!

PhyWhisperer on CrowdSupply





- Everyone loves OSHW.
- Nobody wants to actually build it.
- Solution: Search "PhyWhisperer" to see crowd funding campaign!

- A true history of block chain!
- Me gambling with 0.3 BTC!
- Physical give-aways!
- Attacking bitcoin wallets & stealing funds!
- Stealing authentication credentials!
- Crass commercialization!
- Open-Source Tool Release!
- Fixing your crappy code!

EMFI Demo

<Hopefully my camera works>

- A true history of block chain!
- Me gambling with 0.3 BTC!
- Physical give-aways!
- Attacking bitcoin wallets & stealing funds!
- Stealing authentication credentials!
- Crass commercialization!
- Open-Source Tool Release!
- Fixing your crappy code!

Wait – What About Authentication Tokens?

 Since submitting & preparing this I realized several other interesting targets, and rather than a second talk I give you...





Who's using these tokens?

Microsoft

BONUS CONTENT

Password-less protection

Reduce your risk exposure with password alternatives



Example: Solo Keys Authentication Token BONUS CONTENT Address Description Variable Name else if($(req \rightarrow Walue \gg 8) ==$ 0x200000bc HID Descriptor. USBD_HID_Desc HID_DESCRIPTOR_TYPE){ Pointer to key. 0x20001b0c signing key pbuf = USBD HID Desc; 0x20001b10 HMAC secret. master_secret

len = MIN(USB_HID_DESC_SIZ, req ->wLength);

TIP: You can find ECC private keys in memory if you have public key to compare with. If memory layout is unknown, we can register a new service (to get public key from device), perform attack, and figure out if any private keys in our memory dump.

ECC Private Key.

SHA256 context.

0x20001b50

0x20001b70

privkey.8369

sha256 ctx



https://fidoalliance.org/certification/authenticator-certification-levels/

- A true history of block chain!
- Me gambling with 0.3 BTC!
- Physical give-aways!
- Attacking bitcoin wallets & stealing funds!
- Stealing authentication credentials!
- Crass commercialization!
- Open-Source Tool Release!
- Fixing your crappy code!

How Concerned Should You Be?



How to Fix It?

- Why can you send back 64K of memory? No descriptors are that big!
- Devices have memory protection.
 - We can armour the sensitive data with invalid memory segments, or "disable" memory segments when read-out shouldn't be needed.
- Move memory layout around so we can't read into sensitive data.
 - Less useful as another fault might let us corrupt a pointer instead.
- Encrypt data in-place.
 - Don't allow a "dumb" dump to figure out this critical data!



- The disclosed problem has been fixed in latest firmware patch.
- The disclosed problem did not affect people using passphrases.

https://blog.trezor.io/details-of-security-updates-for-trezor-onefirmware-1-8-0-and-trezor-model-t-firmware-2-1-0-408e59dc012

Solo Key Fixes

• The disclosed problem has been fixed in GIT!

- A true history of block chain!
- Me gambling with 0.3 BTC!
- Physical give-aways!
- Attacking bitcoin wallets & stealing funds!
- Stealing authentication credentials!
- Crass commercialization!
- Open-Source Tool Release!
- Fixing your crappy code!



- Tamper-resistant enclosures <u>aren't enough</u> when discussing "nearphysical" attacks. Commercially available EMFI tools exist (see -ChipSHOUTER) and can be purchased sometimes...
- If implementing a USB device, validate your response size makes sense to avoid many attacks!
- Testing against EMFI is useful to understand vulnerability but you can do some testing with simulation/emulation and through code review.

Let's Do It!

- I've got PhyWhisperer-USB PCBs here for you ⁽²⁾ Join the CrowdSupply and make the full thing happen!
- See my WOOT'19 paper (linked from oflynn.com) & White Paper!

Blogoflynn.comTwitter@colinoflynnEmailcolin@oflynn.comDal Emailcoflynn@dal.caCompanynewae.com

I'm terrible on response times right now, please don't take it personally if it takes weeks++ :/

For More, See:

• WOOT 2019 Paper:

https://www.usenix.org/conference/woot19/presentation/oflynn

- My blog post: <u>http://colinoflynn.com/2019/03/glitching-trezor-using-emfi-through-the-enclosure/</u>
- PhyWhisperer-USB repo:
 https://github.com/powpotoch/ph

https://github.com/newaetech/phywhispererusb