

EXPLAIN ETHEREUM SMART CONTRACT HACKING LIKE I AM FIVE

Zoltan Balazs – MRG Effitas
2018 March

MRG ffitas
Efficacy Assessment & Assurance



Whoami?

Zombie Browser Toolkit

<https://github.com/Z6543/ZombieBrowserPack>

HWFw Bypass tool

Similar stuff was used in PacketRedirect in Danderspritz FlewAvenue by EQGRP

<https://github.com/MRGEffitas/hwfwbypass>

Malware Analysis Sandbox Tester tool

https://github.com/MRGEffitas/Sandbox_tester

Played with crappy IoT devices – my RCE exploit code running on ~600 000 IP cameras via Persirai

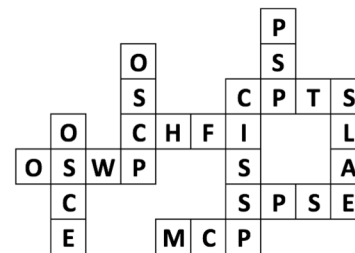
<https://jumpespijump.blogspot.hu/2015/09/how-i-hacked-my-ip-camera-and-found.html>

<https://jumpespijump.blogspot.hu/2015/08/how-to-secure-your-home-against.html>

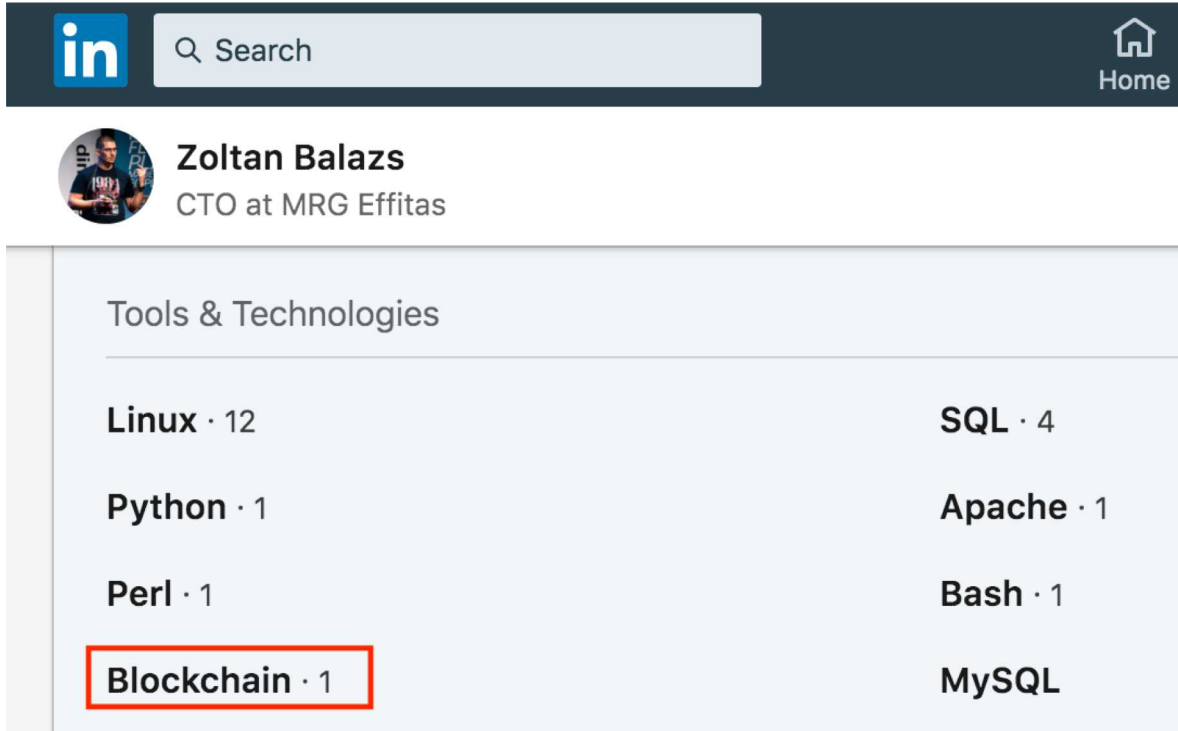
Invented the idea of encrypted exploit delivery via Diffie-Hellman key exchange, to bypass exploit detection appliances

Implemented by Angler and Nuclear exploit kit developers


<https://www.mrg-effitas.com/generic-bypass-of-next-gen-intrusion-threat-breach-detection-systems/>



So who am I to talk about this topic?



in Search Home

 **Zoltan Balazs**
CTO at MRG Effitas

Tools & Technologies

Linux · 12	SQL · 4
Python · 1	Apache · 1
Perl · 1	Bash · 1
Blockchain · 1	MySQL





Questions

Hands up if you know something about blockchain

Hands up if you ever tried to explain Bitcoin to your parents/colleagues/kids

Hands up if it ended: “it is complicated”

Hands up if you ever interacted with a Smart Contract



Everything is oversimplified

The events depicted in this presentation are fictitious. Any similarity with anything you know is merely coincidental.

This presentation will help you understand the big picture about smart contracts



I will never give you investment tips

In case you believe this presentation wants to convince you to sell/buy/HODL, you are wrong and I would never do that

The only investment tip I can give you:

Only play with money today that you can afford to lose tomorrow



Main idea of cryptocurrencies

Let's go with metaphors on this topic

Math is hard (at least for me it is true)

Let's form a group where we solve mathematical challenges **MINING**

Everyone can easily check if someone solved the hard math challenge

When someone solves a math challenge, they receive moneZ (my imaginary cryptocurrency)

$$\begin{array}{l} \text{🍗} + \text{🍗} + \text{🍗} = 45 \\ \text{🌻} \times \text{🌻} \times \text{🌻} = 125 \\ \text{🎃} + \text{🎃} + \text{🎃} = 3 \\ \text{🍗} - \text{🌻} + \text{🎃} = ? \end{array}$$

Soviet miners line up to get their highly rationed GPUs



Transactions

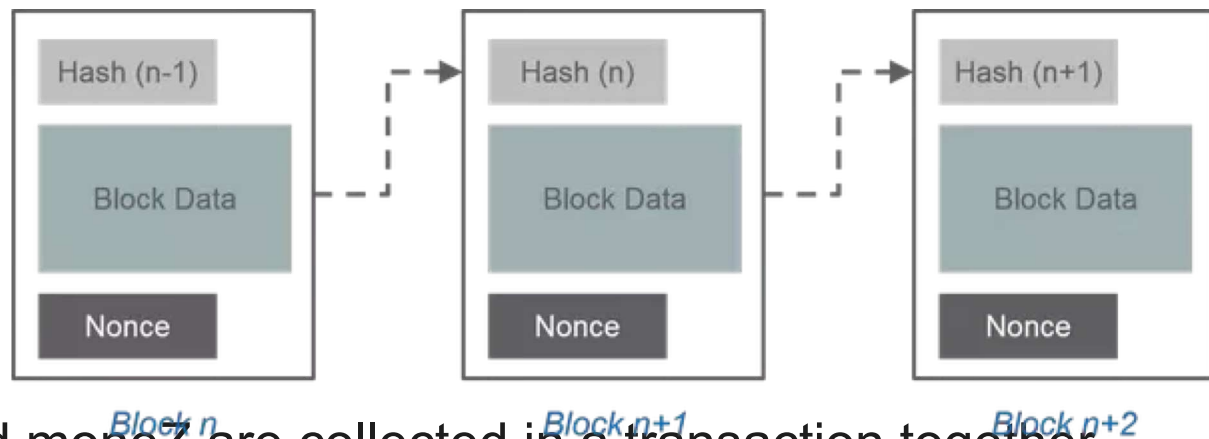
If Bunny wants to send moneZ to Piggie, everyone will know and has to know about all details of the transaction

In fact, everyone knows how much moneZ everyone has because it is public knowledge

Sidenote: ETHEREUM different



Blockchain

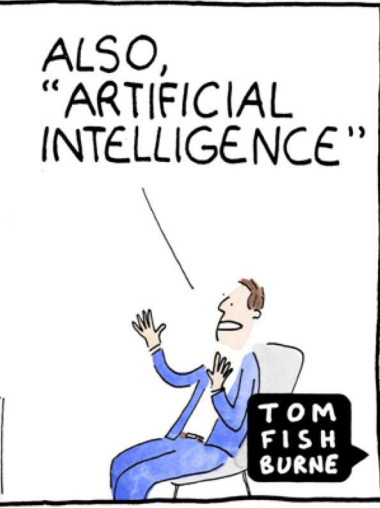
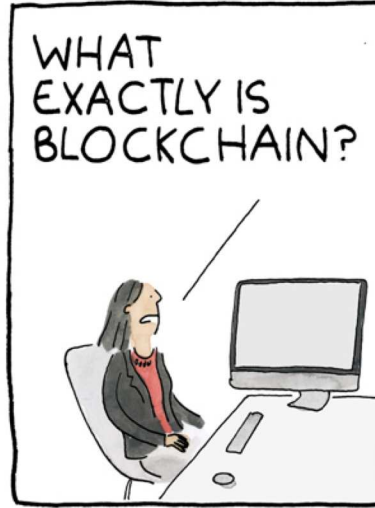
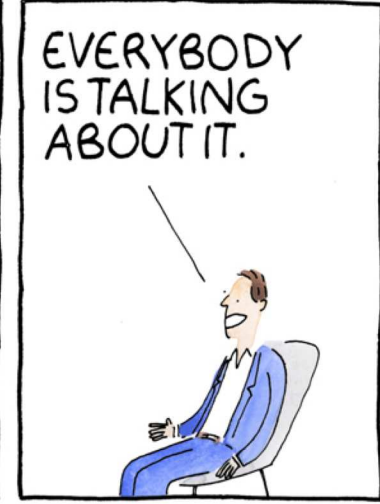
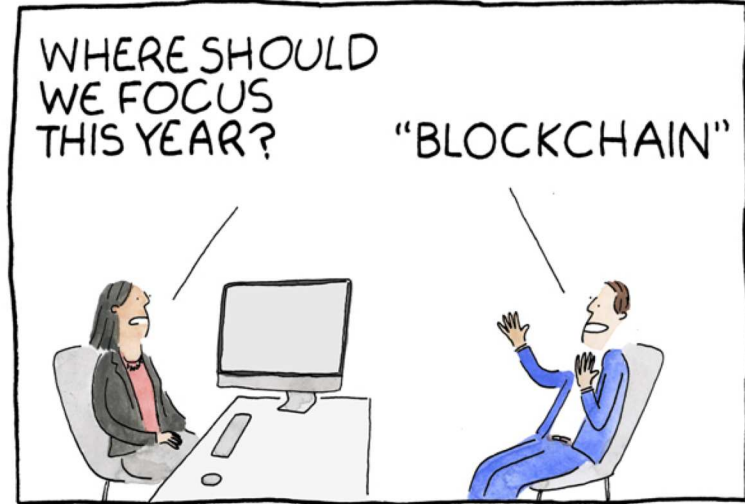


The last not yet processed moneyZ are collected in a transaction together

The bundle of transactions is included as additional parts to the math challenge to be solved **blocks**

Including the transactions in the math challenges will cost moneyZ for the initiator **transaction fee**

A long paper trail is created where every blocks are recorded **blockchain**



How can newcomers get moneZ - Bitcoin, etc.?

They can start to solve new math challenges

Before that they should get a copy of the long paper trail -
blockchain

By solving math challenges and including the transactions, they
get the transaction fee

Or they can ask someone to send them moneZ

In exchange they can give something - real money, Alpaca
socks

What is a wallet

The **wallet** holds all the moneZ you previously received

Whenever you send moneZ from your wallet to someone else's wallet, you sign the transaction with your signature, which is impossible to counterfeit

Transactions are irreversible and final*

Everyone will know you transferred the money, you can't draw back

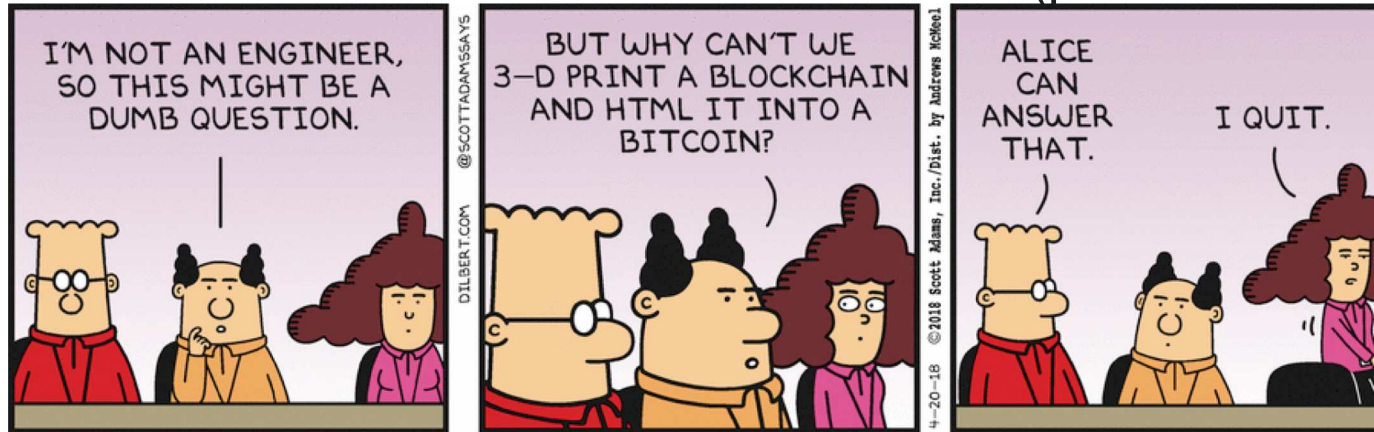


Why is this interesting?

You don't have to be in the same room to do the math challenges

You can do all the stuff through the Internet

You don't even have to know who the others are (pseudo-anonymity)



Everything you don't understand about money combined



with everything you don't understand about computers

OK, now what

Now you have a basic understanding of cryptocurrencies in general

And how Bitcoin works

Let's take a deep dive into **Smart Contracts**

Bitcoin is also capable of doing Smart Contracts

Ethereum YAC (Yet Another Cryptocurrency) was designed for Smart Contracts



What is the problem Smart Contracts try to solve?

You can transfer money without any trusted third party

But how can you sign contracts if you don't know the others and you don't want to involve any trusted third party?

Crowdfunding without Kickstarter?

Trusting online gambling sites? Why should you trust them?

Paying upfront to buy something and get delivered later?

There is a smart contract existing today to finance the defacement of a website ...

Smart Contracts

You trust the vending machine that if you put money into it, you will get a Diet Coke

If you give it \$1, and press this button, you will get a Diet Coke.

But what if the vending machine doesn't look to be trusted? What if you have to pay \$1000 to get 1000 Diet Cokes? What if the vending machine is 1000 miles away? You need a trusted third party ...

So you order 1000 Diet Cokes from Amazon ...



Smart Contracts

Sign and get a countersign of the contract
carve the contract into stone
contracts carved into the stone cannot be
modified

In the smart contract world, the stone is the blockchain
it is powered by the time and energy spent on
solved math challenges



Which language do we use?

International contracts are hard -
language

Code is universal

Contracts embedded in blockchain -
cannot be disputed

Code contracts “carved” into blockchain
(*stone*) is a Smart Contract



What is gas?

Smart contract is code which can be executed by anyone who solves the math challenges for moneZ - mines the cryptocurrencies

Similarly to moneZ transaction fees, you have to pay moneZ to get the smart contract code executed by everyone

The more complex the smart contract code is, the more moneZ you have to pay

This is called **gas** in Ethereum



VESPENE GAS

You need more of it

Are we there yet?





Ethereum Virtual Machine

Bytecode: it is not a machine code, thus you need a VM to execute it

Solidity: compile JavaScript-like code into **EVM** bytecode

Source code can be published - creates trust

Solidity source code compiles into the same bytecode (reproducible)

At least with the same parameters and same compiler version





Ethereum Virtual Machine

EVM is executing the bytecode on every node!!!

Mining node + validating nodes

EVM is Turing complete (as are most programming languages)

Turing complete means it has RAM, ROM, can calculate, goto, if-then-else, arrays

Smart contracts and individual wallets can both hold Ether

Ether --- the thing you mined when solving math problems



JavaScript developers today

Solidity (smart contract language) looks similar to JavaScript

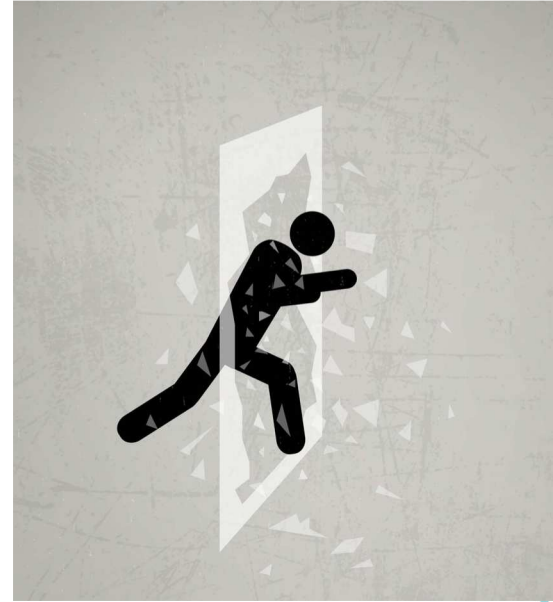
You need web3.js based frontend - this is JS


Many smart contract coders have JavaScript background

JavaScript: **You must move fast and break things**

With Ethereum Smart Contracts, this approach is not “profitable” ...



Solidity: Deploy once, be hacked anytime







Etherization
by Vedran Kajic
Strategy Game

LIVE





Life Lottery
by FreeGeeks
100% fair lottery

LIVE





Gnosis
by Consensys / Gnosis Team
Prediction market platform

LIVE





Acebusters
by Johann Barbie +1
Poker platform

LIVE




GotCHa
by Blockchain Manic
A simple and fair game to win Ethereum

LIVE





CryptoKitties
by Axiom Zen +7
Collect and breed digital cats

LIVE





EthRPS
by EthRPS Team
Play Rock, Paper, Scissors Game

LIVE




The Pyramid Game
by Jesse Busman
Place blocks to build a pyramid and profit

LIVE





IPFS
by Juan Benet
A peer-to-peer hypermedia protocol

LIVE





Leeroy ^{NSFW}
by Trie
Social media platform



LIVE




GhostKat
by GhostKat Team
An experimental streaming service that doesn't use a server



Realms of Ether
by Ethergames
A strategy game where you can own fortresses



The Million Ether Homepa...
by Peter Porobov
Advertising platform



Lotterium
by Emerson Estrella
Open source lottery

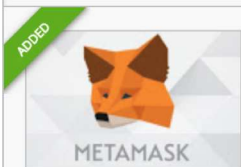


CryptoZombies
by LoomNetwork Team
Learn to code Ethereum DApps by building your own game



Extensions

More Extension Results



MetaMask

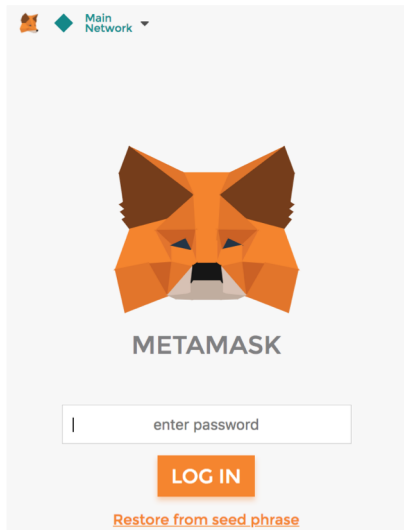
offered by <https://metamask.io>

Ethereum Browser Extension

★ RATE IT

Productivity

★★★★★ (952)



Wallet address

Ox

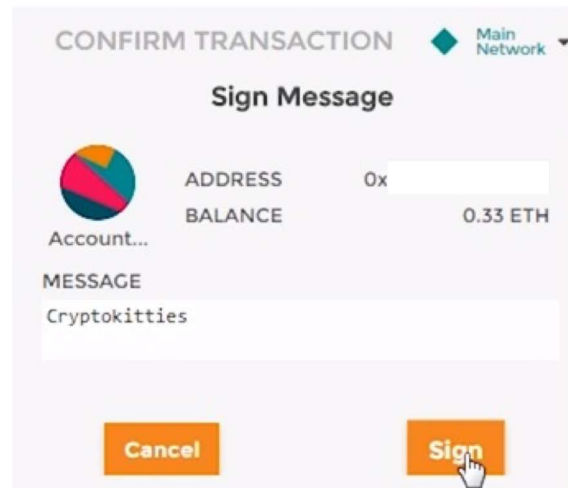
Email *

Nickname


If you have trouble signing in or editing your account, send us a message at meow@cryptokitties.co.

Make sure to save your MetaMask login information and account recovery details! We can't help you regain access if you lose it.

Save account info



Meet the latest CryptoKitties exclusive: [Golden Dragon Cat - Kitty #888](#)

 For sale \equiv 440.90



Latest Metamask since August 2018 shows the hex data to be sent

FUNCTION TYPE: Bid

```
Parameters:  
[  
  {  
    "type": "uint256"  
  }  
]
```

HEX DATA:

```
0x454a2ab300000000000000000000000000000000000000000000000000000000  
000000000000000000dd9c2
```

Identifier for the Bid function (MethodID) –
`keccak256("bid(uint256)")[:4]`

59 zeroes because the
world needs 2^{256} Kitties

Kittie ID in hex



WTF???

Promise was decentralized Amazon and all I am showing is cryptokitties ...

Well, theory and practice does not go well together

Smart contracts are great when everything is in the blockchain (Ether, kitties)
and nothing is materialized

When you have to interact with the real world, stuff gets complicated

<https://openbazaar.org/> -- it is not based on Ethereum smart contract





OpenBazaar
Discover



Transactions

My Page

 **FREE SHIPPING**



Princess Twilight Sparkle
and the Forbidden Book of Power
by G. M. Berrow



My Little Pony Twilight...

★ 0.0 (0) **\$24.99**

 **FREE SHIPPING**



Princess Twilight Sparkle vs. Changeling
Guardians of Harmony



My Little Pony Guardians...

★ 0.0 (0) **\$39.99**

 **FREE SHIPPING**




One Piece Monkey D Luff...

★ 0.0 (0) **\$29.99**

 **FREE SHIPPING**




MINI Air Hockey Table To...

★ 0.0 (0) **\$59.99**

 **INCHES**
FREE SHIPPING




My Little Pony Apple...

★ 0.0 (0) **\$41.99**

 **FREE SHIPPING**




My Little Pony Rarity...

★ 0.0 (0) **\$49.99**



I WANT HACK!!!!

Where is the hacking?

You promised hacking





Smart contracts are code. Code can be hacked

Contract written in human language can mean multiple things, it can be interpreted differently

Smart contract written in code can be interpreted only one way

Issue is, this does not mean code will be interpreted the way the Smart Contract developer thought





DAO Makes History, Raises \$130 Million, Breaking All Records

197 Total views

164 Total shares



The millennial generation is experiencing history in the making as they flock in amazing numbers - almost 5,000 members on the DAO slack channel - to fund one of the most promising decentralized autonomous organizations.

More Ethereum Attacks: Race-To-Empty is the Real Deal

09 JUNE 2016 on [ethereum](#), [smart contracts](#), [security](#), [solidity](#)

[Chriseth](#) at github casually pointed out a terrible, terrible attack on wallet contracts that I had not considered. If there were a responsible disclosure avenue for ethereum contract developers, I would use it, but there doesn't seem to be. Not only that, this code has been out and published on github for long enough that I wanted to get the news out there quickly.

In Brief: Your smart contract is probably vulnerable to being emptied if you keep track of any sort of user balances and were not very, very careful.

As always, I'm available for smart contract review and audit, [email me](#). You can read about other security considerations on my blog [here](#).



Stephan Tual

Follow

Slock.it Founder, Blockchain and Smart Contract Expert, Former CCO Ethereum

Jun 12, 2016 · 3 min read · Unlisted

No DAO funds at risk following the Ethereum smart contract 'recursive call' bug discovery

Our team is blessed to have Dr. Christian Reitwießner, Father of Solidity, as its Advisor. During the early development of the [DAO Framework 1.1](#) and thanks to his guidance we were made aware of a generic vulnerability common to all Ethereum smart contracts. We promptly circumvented this so-called “recursive call vulnerability” or “race to empty” from the DAO Framework 1.1 as can be seen on line [580](#):

The important takeaway from this is: as there is no ether whatsoever in the DAO's rewards account—this is NOT an issue that is putting any DAO funds at risk today.

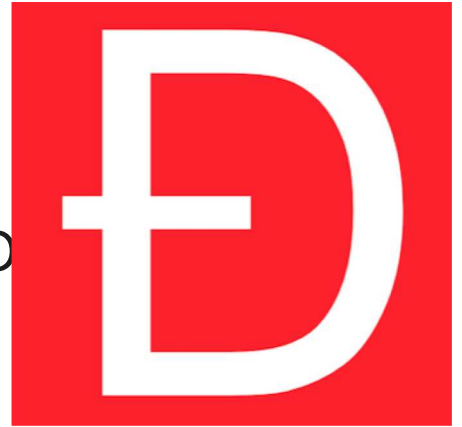
The DAO: Recursive call + race condition

DAO == Decentralised Autonomous Organization

June 18th, 2016

Attacker transfers Ether worth \$250 million from DAO

Reentrancy at the splitDAO function



<http://hackingdistributed.com/2016/06/18/analysis-of-the-dao-exploit/>

The DAO hack

It's like the bank teller doesn't change your balance until she has given you all the money you requested.

“Can I withdraw \$500? Wait, before that, can I withdraw \$500?”

The smart contract was designed only to check you have \$500 at the beginning, once, and allow themselves to be interrupted.



The DAO hack

// INSECURE --- this is not DAO code, but similar so it is easy to understand

```
function withdrawBalance() public {                                // 1st line
    uint amountToWithdraw = userBalances[msg.sender];           // 2nd line
    require(msg.sender.call.value(amountToWithdraw)()); // 3rd line. At this point, the caller's code is
    userBalances[msg.sender] = 0;                                // 4th line
}
```

Look how much I have.



Can I hold it?



The solution?

Rewrite the past and pretend it didn't happen

Attacker got away with his ETH Classic

worth \$67.4 million

Attacker had to wait 34 days

due to code in DAO Smart Contract

Someone also shorted ETH minutes before the hack started



An Open Letter To the DAO and the Ethereum community



chris4210 (66) ▾ in ethereum • 2 years ago

===== BEGIN SIGNED MESSAGE =====

To the DAO and the Ethereum community,

I have carefully examined the code of The DAO and decided to participate after finding the feature where splitting is rewarded with additional ether.

I have made use of this feature and have rightfully claimed 3,641,694 ether, and would like to thank the DAO for this reward. It is my understanding that the DAO code contains this feature to promote decentralization and encourage the creation of "child DAOs".

I am disappointed by those who are characterizing the use of this intentional feature as "theft". I am making use of this explicitly coded feature as per the smart contract terms and my law firm has advised me that my action is fully compliant with United States criminal and tort law. For reference please review the terms of the DAO:

"The terms of The DAO Creation are set forth in the smart contract code existing on the Ethereum blockchain at 0xbb9bc244d798123fde783fcc1c72d3bb8c189413. Nothing in this

Multi-signature wallets



“Captain planet, the world’s first multi-factor authentication” © dnet

Shared vulnerable library + reinit - 2017 July 20

USD 31M stolen

A lot more was in danger, but good guys were faster

Lot of shared libraries exists in the blockchain

Save gas

Contracts now share the same vulnerabilities

Parity multi-signature wallets

<https://medium.freecodecamp.org/a-hacker-stole-31m-of-ether-how-it-happened-and-what-it-means-for-ethereum-9e5dc29e33ce>



Teh code

NON LIBRARY CODE

```
function() payable { // someone called a function we don't have?
  if (msg.value > 0)    // some ether is sent
    ...
  else if (msg.data.length > 0) //ether is not sent, but some data is
    _walletLibrary.delegatecall(msg.data); //let's check if we can execute this code via shared
library
}
```

- If the method name is not defined on this contract...
- And there's no ether being sent in the transaction...
- And there is some data in the message payload...

for whatever method that calls DELEGATECALL, it will call the same method on the contract you're delegating to, but using the context of the current contract

Teh library codez

```
function initWallet(address[] _owners, uint _required, uint _daylimit) {  
    //the shared library has initWallet and it is public !  
  
    initDaylimit(_daylimit);  
    initMultiowned(_owners, _required);  
}
```


initWallet is not in the non-library code, but is called in the shared library



So some random guys don't know how to code Smart Contracts ...

paritytech / parity

<> Code | Issues 165 | Pull requests 26 | Projects 5 | Insights

Tree: e06a1e8dd9 ▾ parity / js / src / contracts / snippets / enhanced-wallet.sol

 **gavofyork** Fix initialisation bug.

2 contributors  

465 lines (390 sloc) | 15.9 KB

```
1 //sol Wallet
2 // Multi-sig, daily-limited account proxy/wallet.
3 // @authors:
4 // Gav Wood <g@ethdev.com>
```

Article [Talk](#) [Read](#) [Edit](#)

Solidity

From Wikipedia, the free encyclopedia

This article is about the programming language. For the state

Solidity is a contract-oriented programming language for writing [smart contracts](#).^[1] It is used for implementing smart contracts^[2] on various [blockchain](#) platforms.^{[3][4][5]} It was developed by [Gavin Wood](#), Christian Reitwiessner, Alex Beregszaszi, Liana Husikyan, Yoichi Hirai and several former [Ethereum](#) core contributors to enable writing smart contracts on blockchain platforms such as Ethereum.^{[6][7][8]}

Fixing the Parity bug

Parity fixed previous bug
and introduced a new one



3esmit commented on Aug 3, 2017

Contributor



BTW, when you deploy WalletLibrary, the init function will be open in that contract. I recommend you calling initWallet on WalletLibrary right after its deploy, just to ensure no one will use it.



7



5

Library contract was not initialized properly. That allowed anyone to turn the library contract into a multi-sig wallet

The next Parity hack

November 2017 - 300M USD lost

@devops199 “accidentally” called `initWallet()` method to own the library

@devops199 “accidentally” called `kill()` method to self-destruct it

It is still planned to be fixed – forking EIP-999





The Parity hacks

First hack was calling the smart contracts relying on the shared library

Lot of calls needed to steal money from all the contracts

Second hack directly called the shared library

One call to rule them all

The shared library did not have ETH, but now that it is killed, libraries relying on it are not usable anymore

The issue was “known” but risk was misdiagnosed

There was a plan to fix it



Intro to integer underflow

Underflow

If there are (unsigned integer 8) 3 people on the bus, and four of them took of the bus, how many people are still on the bus?

255



Intro to integer overflow

Overflow

If there are (unsigned integer 8) 255 people on the bus, and the bus is totally full, and one guy hops on the bus, how many people are on the bus?



Proof of Weak Hands

<https://medium.com/@optimumregret/the-surreal-madness-of-ethereums-pyramid-schemes-da705fe7d92e>

USD 2M lost
unsigned integer underflow withdrawal

Use Safemath!

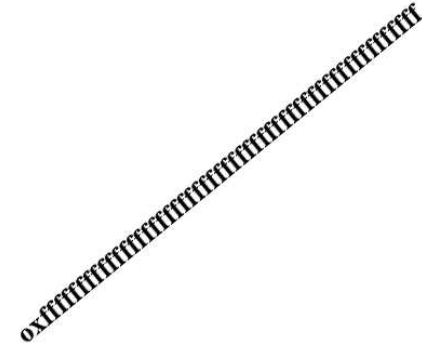
<https://etherscan.io/tx/0x233107922bed72a4ea7c75a83ecf58dae4b744384e2b3feacd28903a17b864e0>

WHO WOULD WIN?

\$2,000,000 of crypto in a program coded by pajet



the letter f





Conclusion

Writing secure Smart Contracts is hard

Ethereum is still in beta

Hacking Smart Contracts is possible, fun, but probably illegal

Hacking your own smart contract is probably not illegal

Hacking in test blockchain is not illegal



Where to learn to code? cryptozombies.io



Where to learn to hack?

← → ↻ 🏠  Secure | <https://ethernaut.zepplin.solutions>



Ethernaut

[Home](#)

[Help](#)

[About](#)

Levels

- 0. Hello Ethernaut ✓
- 1. Fallback ✓
- 2. Fallout ✓
- 3. Token ✓
- 4. Delegation ✓
- 5. Force ✓
- 6. King
- 7. Re-entrancy
- 8. Elevator

The Ethernaut by zepplin

The ethernaut is a Web3/Solidity based wargame inspired on overthewire.org and the [El Eternauta](#) comic, played in the Ethereum Virtual Machine. Each level is a smart contract that needs to be 'hacked' in order to advance.

If you are looking for the CTF version released for Devcon3, please visit ethernaut-devcon3.zepplin.solutions. This version will be maintained for some time and is still 100% playable.

Are you interested in smart contract development or security? Does securing the world's blockchain infrastructure sound exciting to you? **We are hiring!**

[Play now!](#)

References

Nick Szabo: The idea of smart contracts 1997 <https://perma.cc/V6AZ-7V8W>

https://www.reddit.com/r/explainlikeimfive/comments/12knie/eli5_bitcoins/?st=IZW0ENOG&sh=d566a3ee

<https://medium.freecodecamp.org/smart-contracts-for-dummies-a1ba1e0b9575>

https://www.reddit.com/r/explainlikeimfive/comments/4lz9t4/eli5_ethereum/

<https://github.com/b-mueller/smashing-smart-contracts/blob/master/smashing-smart-contracts-1of1.pdf>

A Decompiler for Blockchain Based Smart Contracts Bytecode by Matt Suiche

<https://www.youtube.com/watch?v=fUzKA-nap20>

<https://www.stateofthedapps.com/>

Cryptozombies.io - best tutorial

Latest hype and scams: <https://boards.4chan.org/biz/>

Hack the planet!



zoltan.balazs@mrg-effitas.com

<https://hu.linkedin.com/in/zbalazs>

Twitter – @zh4ck

www.slideshare.net/bz98

HACKERSULI !!!1!

Greetz to @VitalikButerin, Satoshi Nakamoto

<https://JumpESPJump.blogspot.com>