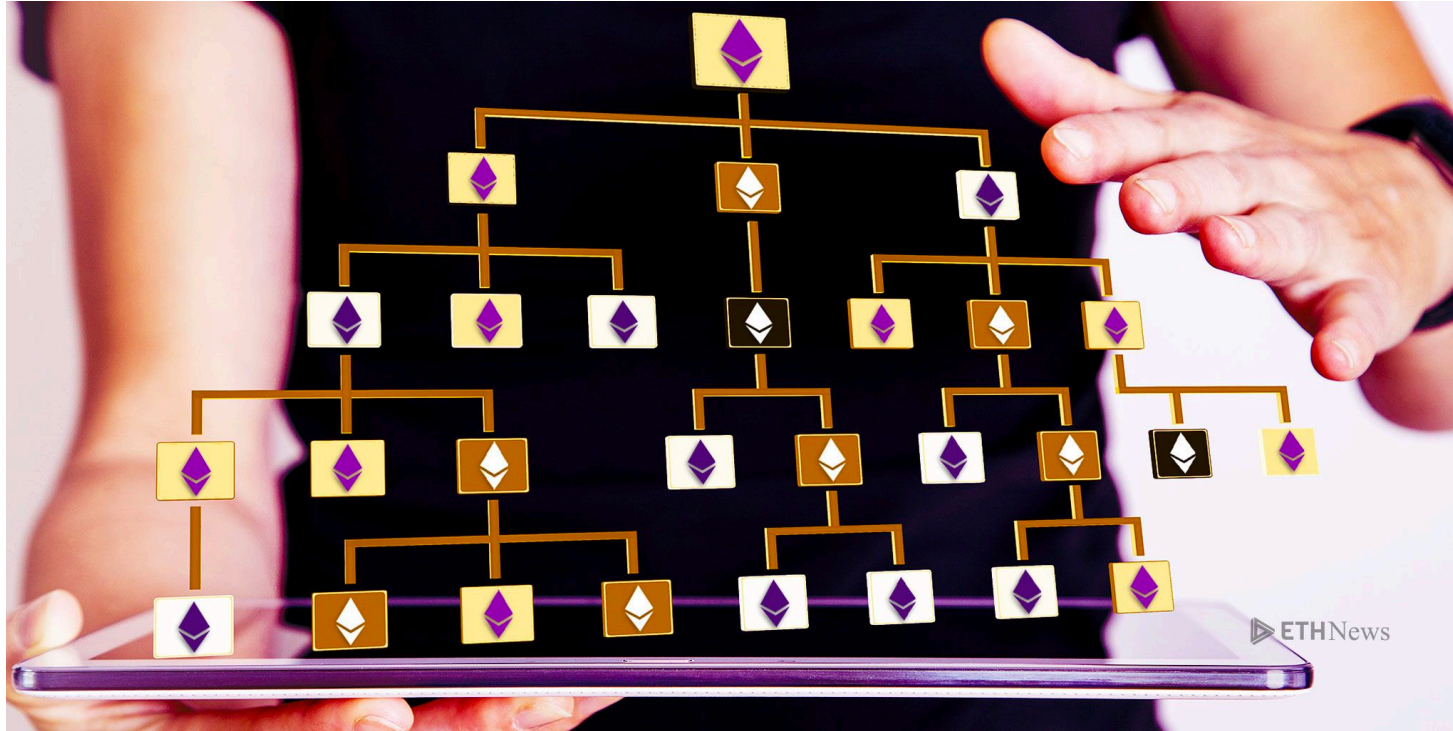


Ethereum Tokens and ICOs

Module 4

Ethereum Improvement Proposals (EIPs)



<https://eips.ethereum.org/>

Token economy

Blockchain-Based Token Economy



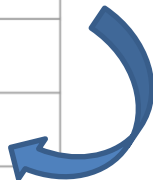
Tokens?

Ethereum tokens are simply digital assets that are being built on top of the Ethereum blockchain. ... They also strengthen the Ethereum ecosystem by driving demand for **ether**...



	ethereum	ethereum tokens
concept	smart contracts platform	digital assets on top of ethereum
market cap (as of may 2017)	~\$17 billion	~\$1.5 billion
native currency	ether	augur (rep), golem (gnt), aragon (ant), & many more
founder	vitalik buterin and team	varies by project
release method	presale raised \$18M in bitcoin	typically through crowd sales

ICO



Simply put

- Dapps can issue tokens
 - Ethereum is a platform that can be used to create any arbitrary smart contract including *smart contracts that represent digital assets called Ethereum tokens*.
- People buy certain tokens of that DAPP in exchange of their ether

Tokens are usually of 2 varieties

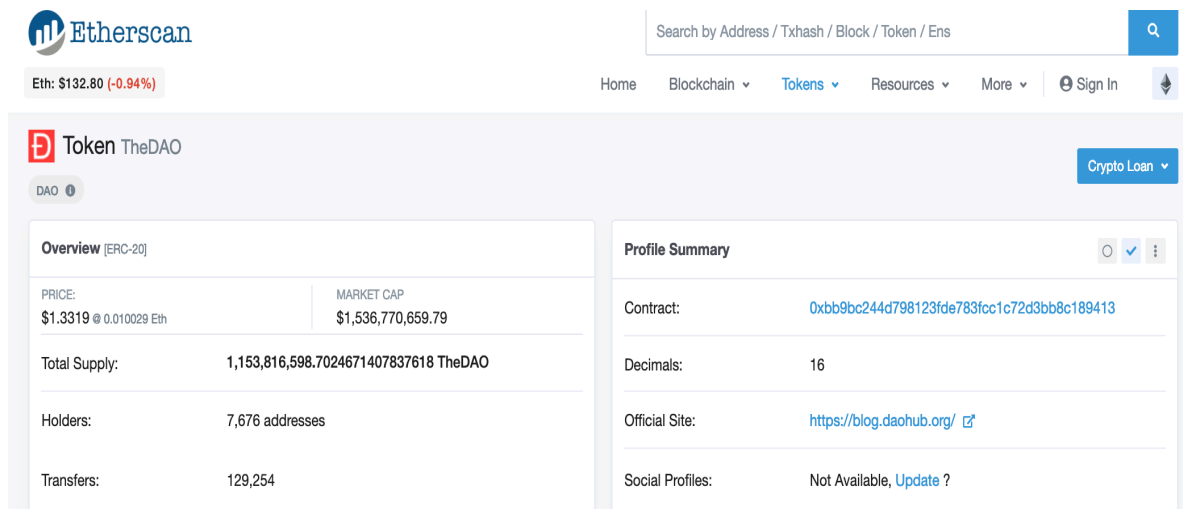
Usage Tokens:

- These are the tokens that act like native currency in their respective DAPPS.
- Golem is a pretty good example of this. If you want to use the services in Golem then you will need to pay with Golem Network Token (GNT)

<https://golem.network/>



Work Tokens: These are the tokens that identify you as a sort of shareholder in the DAPP.
A perfect example of this is the DAO tokens.



Etherscan

Eth: \$132.80 (-0.94%)

Home Blockchain Tokens Resources More Sign In

Token TheDAO DAO ⓘ

Overview [ERC-20]

PRICE:	MARKET CAP
\$1.3319 @ 0.010029 Eth	\$1,536,770,659.79
Total Supply:	1,153,816,598.7024671407837618 TheDAO
Holders:	7,676 addresses
Transfers:	129,254

Profile Summary

Contract:	0xbb9bc244d798123fde783fcc1c72d3bb8c189413
Decimals:	16
Official Site:	https://blog.daohub.org/
Social Profiles:	Not Available, Update ?

Crypto Loan

Token Tracker

Ethereum Token Market Capitalization



A total of 171,649 Token Contracts found

First

<

Page 1 of 18

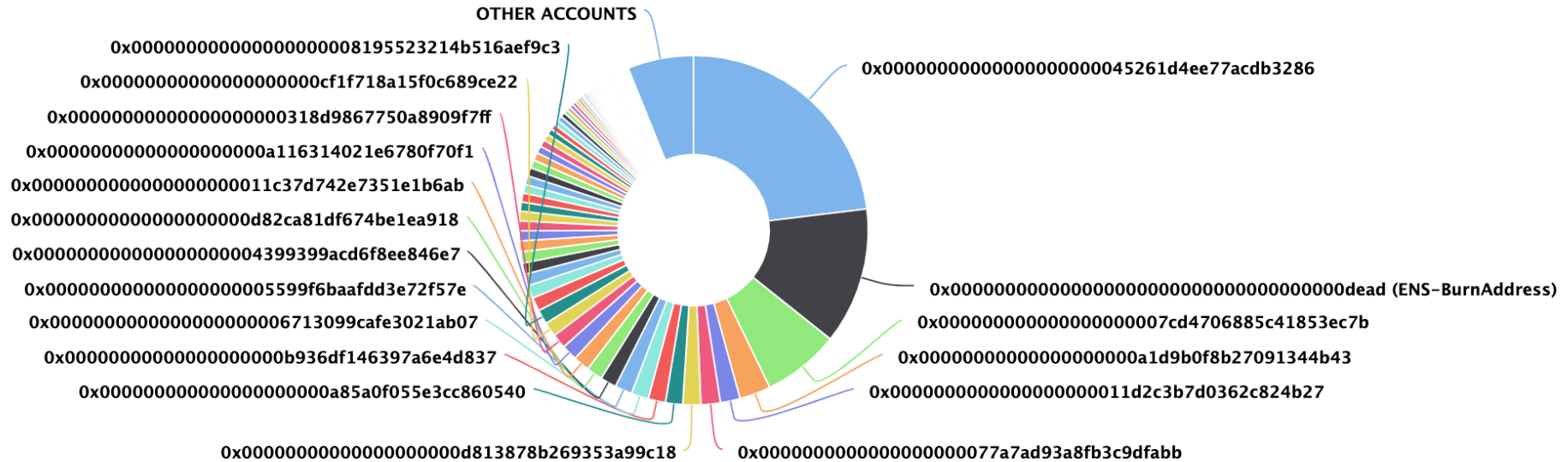
>

Last

#	Token	Price	Change (%)	Volume (24H)	Market Cap
1	 BNB (BNB) Binance aims to build a world-class crypto exchange, powering the future of crypto finance.	\$14.4247 0.0037084695 Btc 0.108808 Eth	 1.55%	\$144,064,494	\$2,036,413,141
2	 Maker (MKR) Maker is a Decentralized Autonomous Organization that creates and insures the dai stablecoin on the Ethereum blockchain	\$651.5525 0.1675087618 Btc 4.914781 Eth	 0.27%	\$917,245	\$651,552,488
3	 VeChain (VEN) VeChain aims to connect blockchain technology to the real world by providing a comprehensive governance	\$0.0046 0.0000011732 Btc	 -3.94%	\$10,190,749	\$253,064,429

VeChain Top 100 Token Holders

Source: Etherscan.io



Token contracts can be very complicated but this is what a basic token contract looks like:

```
contract MyToken {
    /* This creates an array with all balances */
    mapping (address => uint256) public balanceOf;

    /* Initializes contract with initial supply tokens to the creator of the contract */
    function MyToken(
        uint256 initialSupply
    ) {
        balanceOf[msg.sender] = initialSupply;           // Give the creator all initial tokens
    }

    /* Send coins */
    function transfer(address _to, uint256 _value) {
        if (balanceOf[msg.sender] < _value) throw;       // Check if the sender has enough
        if (balanceOf[_to] + _value < balanceOf[_to]) throw; // Check for overflows
        balanceOf[msg.sender] -= _value;                 // Subtract from the sender
        balanceOf[_to] += _value;                       // Add the same to the recipient
    }
}
```

Note, this is not an ERC20 compliant token (more later!)

How a token get its value?

There are two factors:

Supply & Demand (economics 101)

Trust

How do Ethereum tokens take care of **supply** and **demand**?

- Supply: controlled through code
 - Demand: a lot of factors
 - What is the quality of DAPP in itself? Are people excited about the DAPP? Has that DAPP been marketed properly? Is that DAPP going to solve problems?
- If the demand of the DAPP is sufficiently high, and with the supply remaining constant, the value of the token is going to be pretty high.

ERC-20, ERC-721 tokens...

What are those?

Before ERC20, there were many compatibility issues, each token had a completely unique smart contract.

As a solution, the industry came up with a standard protocol for all tokens to follow, which is now known as ERC-20.

Allow for the implementation of a standard API for smart contract tokens within Ethereum ecosystem → For easy trading, exchange and spending.....

The ERC20 standard is basically a specific set of **6 functions** which developers must use in their tokens to make them ERC20 compliant

ERC-20 tokens are tokens designed and used solely on the Ethereum platform.

They follow a list of standards so that they can be shared, exchanged for other tokens, or transferred to a crypto-wallet.

The Ethereum community created these standards (through EIP) with three optional rules, and six mandatory functions.

Optional

Token Name

Symbol

Decimal (up to 18)

Mandatory

totalSupply

balanceOf

transfer

transferFrom

approve

allowance

ERC20 defines interfaces for basic token behavior

Basic functionality:

<https://github.com/ethereum/EIPs/blob/master/EIPS/eip-20.md>

```
function totalSupply() constant returns (uint256 totalSupply)
```

```
function balanceOf(address _owner) constant returns (uint256 balance)
```

Delegating control:

```
function transfer(address _to, uint256 _value) returns (bool success)
```

```
function transferFrom(address _from, address _to, uint256 _value) returns (bool success)
```

Delegating control:

```
function approve(address _spender, uint256 _value) returns (bool success)
```

```
function allowance(address _owner, address _spender) constant returns (uint256 remaining)
```

Notes

`Transfer()` function is used for the initial distribution of tokens to user wallets. This function is the biggest reason that ERC-20 tokens have become so popular **for ICOs**, as it makes it incredibly easy to send tokens to investors when the ICO has been completed.

`transferFrom()` function is what then enables token holders to exchange tokens with one another after the initial distribution.

Why follow ERC-20 standard?

Using the ERC-20 standard ensures that a token is compliant for the following use cases (and more):

- Wallet transfers - sending tokens from one account to another
- Buying and selling on cryptocurrency exchanges
- Purchasing tokens in an crowd sale (ICO)

There are a glut of ERC20 templates on the internet

This is a widely adopted standard, and so tons of tools/service will “just work” if you adhere to ERC20 standard

<http://imgtfy.com/?q=erc20+token+template>

Example implementations are available at:

[OpenZeppelin implementation](#)

[ConsenSys implementation](#)

Are there any problems with ERC-20?

Will talk more later!

Recap

What tokens are

What exactly they do

What rules they follow (ERC20...)

How to create them (soon)

Token development

How to create an ethereum token?

= create a smart contract token.

Let's build our own ERC20 Token:

Refer to [Exercise \(KSU Token\)](#)

ICO

Initial Coin Offering



What are ICOs?

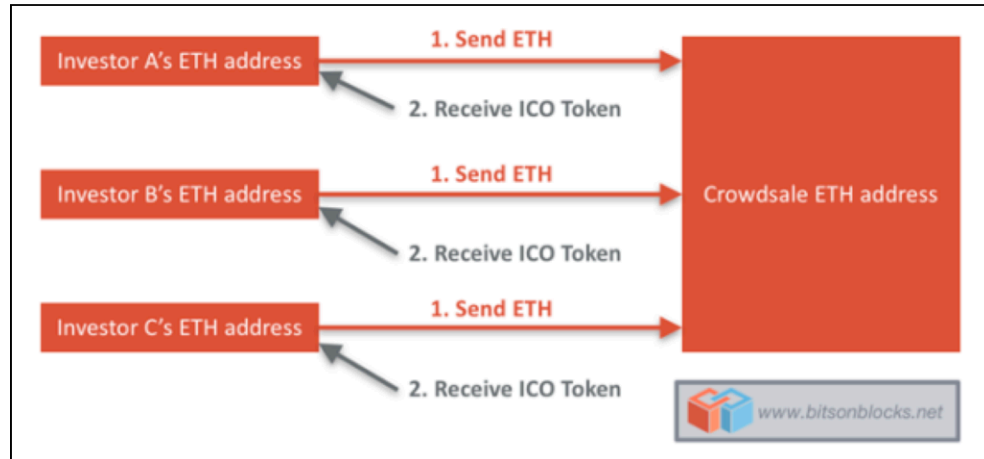
- Tokens can be distributed in a variety of ways. One popular method is holding a **crowd sale**, or an initial coin offering (ICO)— a “trustless” crowd sale

What are ICOs?

- A means of raising capital for projects by creating their own token that can be purchased by investors with Ether (initiated by dapp owners)....
- Anyone can become invested in a project they are interested in by purchasing the tokens of that particular DAPP and become a part of the project themselves (We are talking about **Work Tokens** here).

So, how does an ICO work?

- Firstly, the developer issues a limited amount of **tokens** (tokens can either have a static pre-determined price or it may increase or decrease depending on how the crowd sale is going)
- If someone wants to buy the tokens they send a particular amount of ether to the crowd sale address.



So, how does an ICO work?

- smart contract governs all of the rules for how the crowd sale works



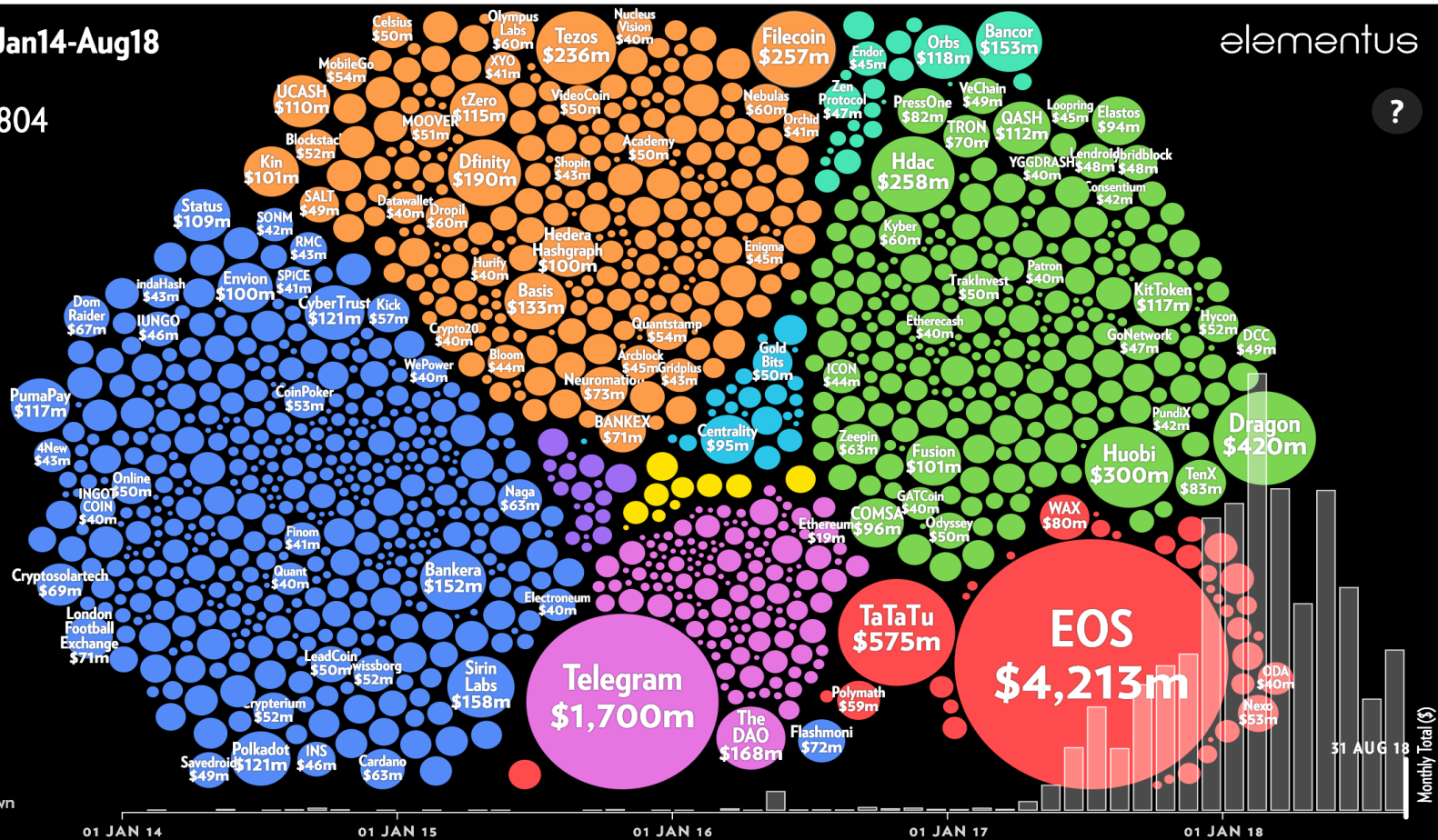
Basically two contracts involved

- Smart contract token (ERC-20 token)
- Smart contract ICO

[Typically put in a dapp with a front-end serving as ICO website]

Token Sales, Jan14-Aug18

\$28,366,035,804



Ponzi scheme

Developers have been raising even before the creation of an Alpha version of their product.

→ certain aspects need to be checked before proceeding with future ICOs (**regulation by SEC**)

Statement on Cryptocurrencies and Initial Coin Offerings

SEC Chairman Jay Clayton

Dec. 11, 2017

The world's social media platforms and financial markets are abuzz about cryptocurrencies and "initial coin offerings" (ICOs). There are tales of fortunes made and dreamed to be made. We are hearing the familiar refrain, "this time is different."

SEC Halts Fraudulent Scheme Involving Unregistered ICO

FOR IMMEDIATE RELEASE

2018-53

Washington D.C., April 2, 2018 — The Securities and Exchange Commission today charged two co-founders of a purported financial services start-up with orchestrating a fraudulent initial coin offering (ICO) that raised more than \$32 million from thousands of investors last year. Criminal authorities separately charged and arrested both defendants.

Lifecycle of an ICO Pre-launch Token

Presale / private investment

Whitepaper released

Public crowdsale

Optionally: tokens can be traded on exchanges

Development continues...

Product launches, you can use your tokens

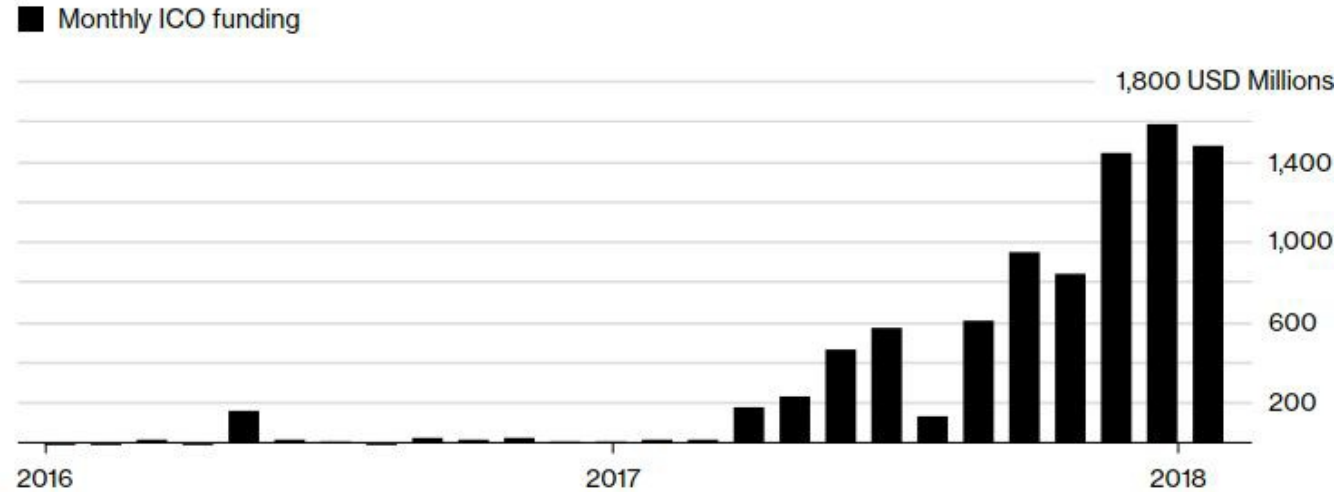
Most likely need to
register with SEC!



Typically implemented
with ERC20

What Clampdown?

Initial coin offering funding is soaring even as regulators tighten grasp



Source: CoinDesk

In 2017, 46 percent of token startups either failed after their ICOs or weren't able to complete funding, according to Bitcoin.com, which drew on statistics gathered by website TokenData. So far this year, 50 of 340 completed ICOs have failed, according to TokenData.

<https://www.bloomberg.com/news/articles/2018-04-02/crypto-hedge-fund-bubble-begins-to-deflate-as-returns-tumble>
<https://news.bitcoin.com/46-last-years-icos-failed-already/>

Let's dissect the Useless Ethereum Token



Useless Ethereum Token

<https://uetoken.com/>

<https://etherscan.io/address/0x27f706edde3aD952EF647Dd67E24e38CD0803DD6#code>

The world's first 100% honest Ethereum ICO.

You're going to give some random person on the internet money, and they're going to take it and go buy stuff with it. Probably electronics, to be honest.

Maybe even a big-screen television.

Seriously, don't buy these tokens.

ICO development (full-stack dapp)

Refer to **Tutorial 6**

Are there any problems with ERC-20?

ERC-20 is not perfect.

There are situations that tokens might be unintentionally destroyed when they are used as payment for a smart contract rather than using Ether.

To fix this bug, the Ethereum community is currently working on a new standard named [ERC-223](#).

In April 2018, a number of exchanges [suspended](#) token deposits and withdrawals of Ethereum-based tokens due to the [batchOverflow](#) bug. It is described as a 'classic integer overflow issue' and might potentially allow an attacker to 'possess a huge amount of tokens'.

Let's dig into the transfer method

Function: transfer(address _to, uint256 _value)

MethodID: 0xa9059cbb

[0]: 000000000000000000000000038716c48535f9035d8578fd35500cc5e8582ba8

{ [1]: 00172

"action": {

"callType": "call",

"from": "0x1b326ad348e19ecfd1406c43d3bf7a95547ac55c",

"gas": "0x6e25",

"input": "0xa9059cbb

000000000000000000000000038716c48535f9035d8578fd35500cc5e8582ba8

00172",

"to": "0xefeaef27c453eb96aea340d03e1724b81973cd61",

"value": "0x0"

},

Failure in an ERC20 token exchanges

Several online exchanges allow you to transfer tokens from a web form

Assumption: All interactions with contracts respects function interface

```
function transfer(address _to, uint256 _value) returns (bool success)
```

Reality: compiled EVM interprets all data as array of 32-byte blocks

Attack: attacker finds a key for an address of the form `0xAAAAAAAA...AAAAA000`
and types the 17-byte prefix into the `#amt` field

`0000...0000 AAAAAAAAA...AAAAA000      00000...0000000055000`

Example: `0000...0000 AAAAAAAAA...AAAAA000      00000000...0000000055`

```
var callData = "0"*12 + ($(#userAddr).val()) + toHex($(#amt).val(), 32)  
           padding      20 byte address      value (little endian, 32 bytes)
```

Many ERC20 Tokens are stuck in unspendable contracts

TOP 5 tokens that got stuck inside contracts:

1. 257 221 GNT^[4] inside Golem contract. (\$109 149 lost)
2. 439 GNO^[5] inside Gnosis contract. (\$105 029 lost)
3. 782 DGD^[6] inside DigixDAO contract. (\$62500 lost)
4. 259 335 SNGLS^[7] inside SingularDTV contract. (\$49 008 lost)
5. 222 REP^[8] inside Augur contract. (\$4900 lost)

To be continued...