DECENTRALIZED
TRAINING SERIES

Hands-on Session

Fundamentals on Ethereum Smart Contracts

Part II

Anastasia Theodouli Kostas Moschou

Information Technologies Institute (ITI), Centre for Research and Technologies Hellas (CERTH)

UNIC | Institute For the Future

CERTH CENTRE FOR RESEARCH & TECHNOLOGY HELLAS | iti Information Technologies Institute

Basic concepts [1/3]

- What is Blockchain?
  - A Blockchain is a continuously growing list of records called blocks. Each block contains a cryptographic hash of its previous block, thus forming a chain.
- What is Ethereum?
  - Ethereum is an open-source, public, blockchain-based distributed computing platform and operating system featuring smart contract (scripting) functionality [11]

Basic concepts [2/3]

- What is an Ethereum account?
  - Accounts represent *(pseudonymous) identities* of external agents (e.g., human personas, automated agents, etc).
  - Accounts are essential for users to interact with the Ethereum blockchain via transactions.
  - Accounts use public key cryptography to sign transactios so that the identity of transaction senders can be securely validated.
  - Every account is defined **by a pair of keys**, a private key and public key.
- What are Ethereum addresses?
  - Accounts are *indexed* by their addresses
  - They are calculated based on the cryptographic hash (keccak256) of the public key which in turn derives from the private key of the user using a mathematic function [14]
  - Ethereum account addresses are 40 hexadecimal digits long. The prefix '0x' means that Ethereum addresses are in hexadecimal notation. E.g.

    0x364198936b17c5c406bA47453E94FD0B3b250275
- In Ethereum, there are two types of Accounts:
  - **Externally Owned Accounts.** They are controlled by private keys and are used to transfer crypto-assets and to create and deploy Smart Contracts to the Blockchain. In their state, they have balance.
  - **Contract Accounts.** They are not controlled by any private key. They are associated with some code (i.e. a Smart Contract). In their state they have both balance and storage.[7] [15]

Basic concepts [1/3]

- What are Ethereum Wallets?
  - Ethereum Wallets are clients (software) that generate and hold for you **private keys** that gives you control over your Ether (the Ethereum token), or other Ethereum-based tokens, and provides you with **Ethereum addressses** which correspond to your public keys and people can use them to send you tokens. [7] Ethereum wallets help you create and manage *multiple* accounts.
- Example Digital Ethereum Wallets
  - Web based (MyEtherWallet, MetaMask)
  - Desktop apps (Ethereum Mist Wallet, Exodus)
  - HW wallets (Ledger, Trezor)
  - Mobile (Breadwallet, Jaxx)

Basic concepts [3/3]

- What is a Smart contract?
  - A script deployed on the Blockchain that it is automatically executed upon a predefined set of rules. It allows for a (quasi) Turing complete programmable logic in the way that Blockchain state changes.
  - A collection of code (its functions) and data (its state) that resides at a specific Contract address on the Ethereum blockchain. [15]
- What is a Decentralised application (Dapp)?
  - It is an application running on a peer-to-peer network of computers rather than a single computer.
  - It is similar to a conventional web application except that instead of an API connecting to a Database, there is a Smart contract connecting to a blockchain. [10]
  - Check a list of Dapps developed to the Ethereum network here:

  (https://www.stateofthedapps.com/)

I / Ethereum Mist Wallet

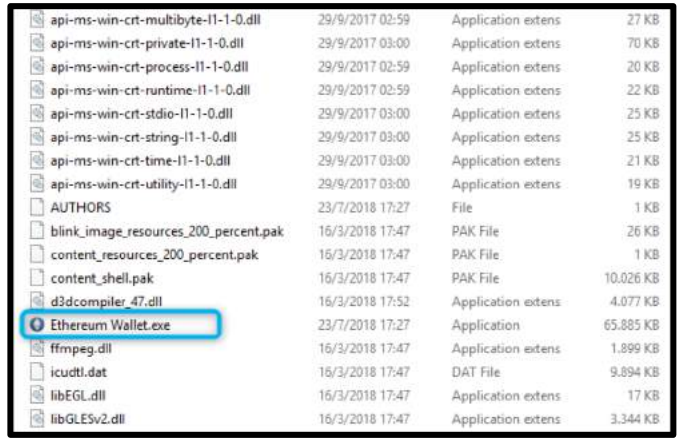
UNIC | Institute For the Future

CERTH CENTRE FOR RESEARCH & TECHNOLOGY HELLAS

iti Information Technologies Institute

Ethereum Mist Wallet / Installation

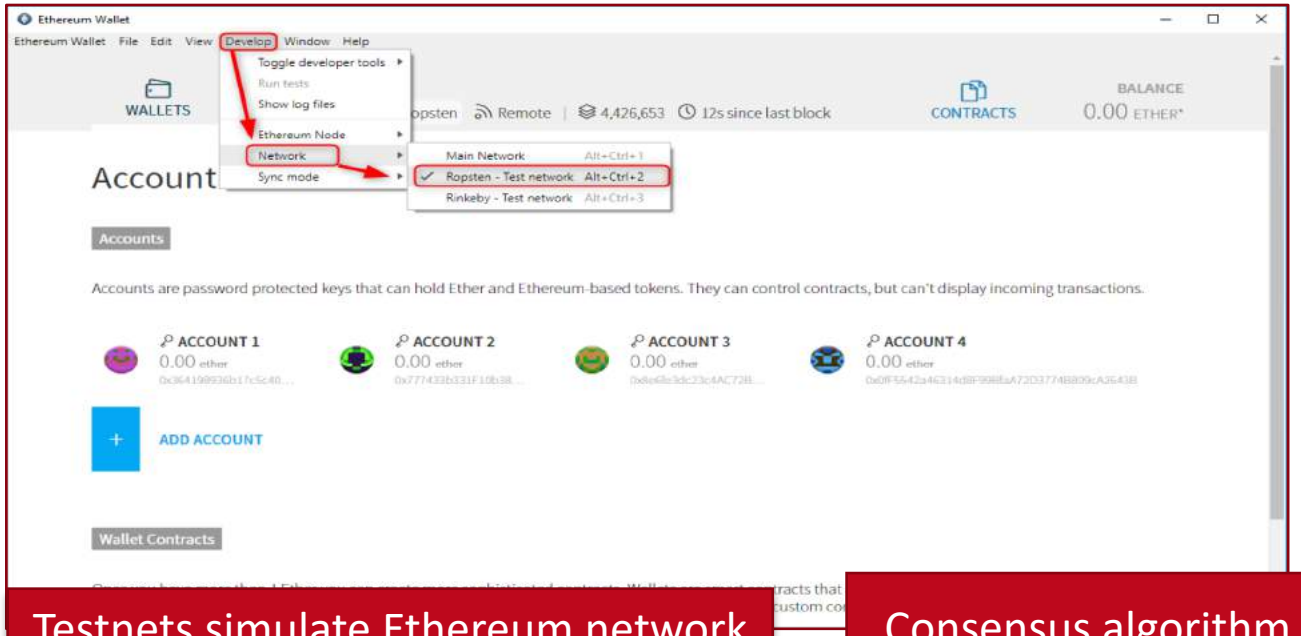
- Ethereum Mist Wallet is a Desktop application that provides a user interface that enable users
  - Create and manage multiple accounts
  - Sign transactions to send and receive Ethers and other crypto-assets built on Ethereum
  - Deploy and interact with Solidity smart contracts
- Ethereum Mist Wallet
  - Go to the page (https://github.com/ethereum/mist/releases) and download the latest Ethereum Wallet (currently, it is 0.11.1) **depending on your system**
  - Install the wallet. E.g. on windows, unzip the .zip file to a folder and run EthereumWallet.exe
- Firstly, follow the Steps to launch the app, choose network, and create a password that will secure your first account in the wallet



api-ms-win-crt-multibyte-l1-1-0.dll	29/9/2017 02:59	Application extens	27 KB
api-ms-win-crt-private-l1-1-0.dll	29/9/2017 03:00	Application extens	70 KB
api-ms-win-crt-process-l1-1-0.dll	29/9/2017 02:59	Application extens	20 KB
api-ms-win-crt-runtime-l1-1-0.dll	29/9/2017 02:59	Application extens	22 KB
api-ms-win-crt-stdio-l1-1-0.dll	29/9/2017 03:00	Application extens	25 KB
api-ms-win-crt-string-l1-1-0.dll	29/9/2017 03:00	Application extens	25 KB
api-ms-win-crt-time-l1-1-0.dll	29/9/2017 03:00	Application extens	21 KB
api-ms-win-crt-utility-l1-1-0.dll	29/9/2017 03:00	Application extens	19 KB
AUTHORS	23/7/2018 17:27	File	1 KB
blink_image_resources_200_percent.pak	16/3/2018 17:47	PAK File	26 KB
content_resources_200_percent.pak	16/3/2018 17:47	PAK File	1 KB
content_shell.pak	16/3/2018 17:47	PAK File	10.026 KB
d3dcompiler_47.dll	16/3/2018 17:52	Application extens	4.077 KB
Ethereum Wallet.exe	23/7/2018 17:27	Application	65.885 KB
ffmpeg.dll	16/3/2018 17:47	Application extens	1.899 KB
icudtl.dat	16/3/2018 17:47	DAT File	9.894 KB
libEGL.dll	16/3/2018 17:47	Application extens	17 KB
libGLESv2.dll	16/3/2018 17:47	Application extens	3.344 KB

Ethereum Mist Wallet / Connect to Network

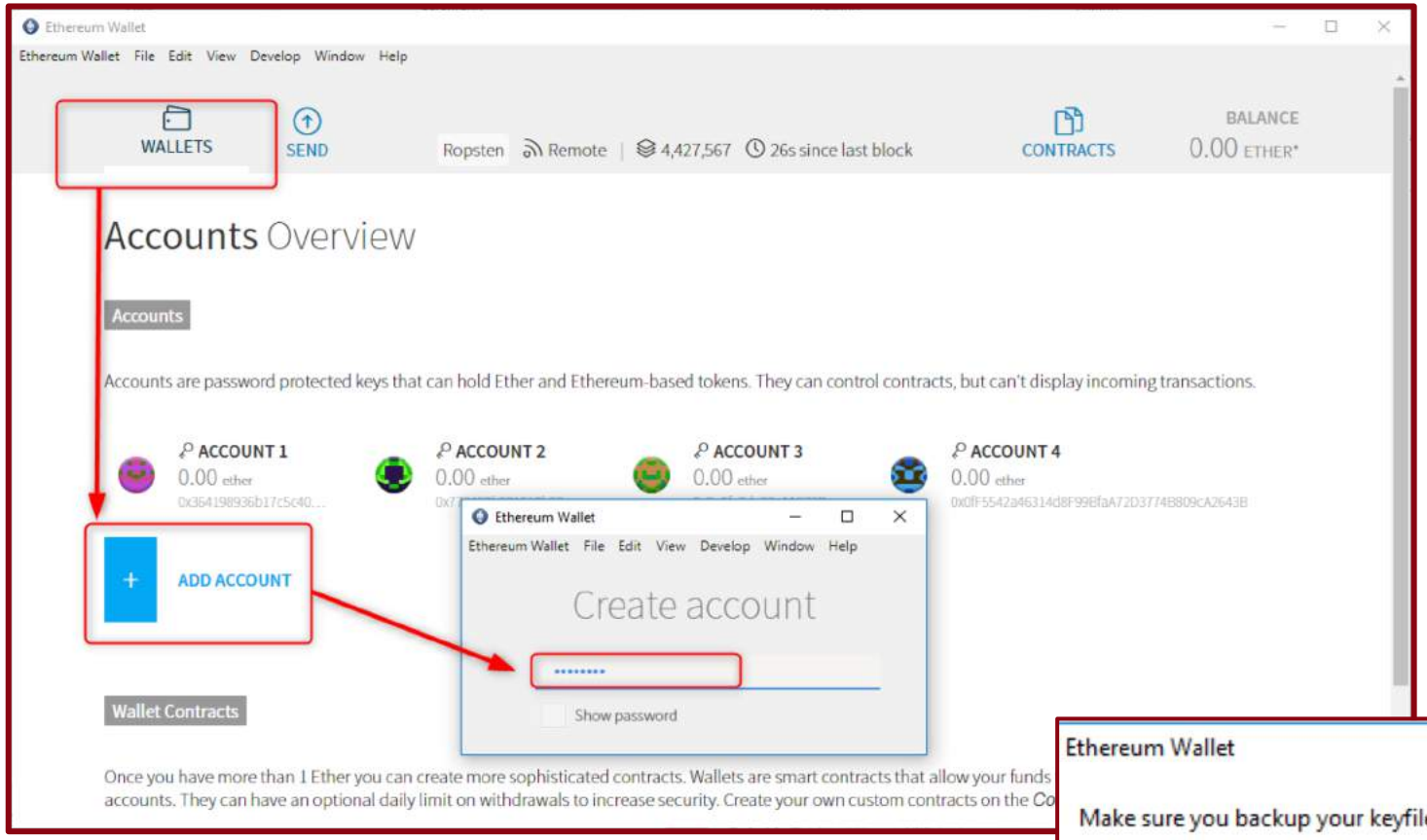
- Options to connect to network
  - **Main Network –** The main Ethereum blockchain network. A peer-to-peer network of participating nodes which *secure* and *maintain* the blockchain [4]
    - https://ethstats.net/ (Statistics page)
  - **Ropsten** - A testing network that runs the same protocol as Ethereum and is used for *later stage testing purposes*. Ropsten uses rETHS (not real Ether) to deploy contracts and pay transaction fees
  - **Rinkeby** - Test network that uses Proof-of-Authority (PoA) consensus algorithm rather than Proof-of-Work (Pow) used by main net and Ropsten



Testnets simulate Ethereum network and Ethereum Virtual Machine (EVM). They allow developers to upload and interact with Smart Contracts *without paying the cost of gas* [1]

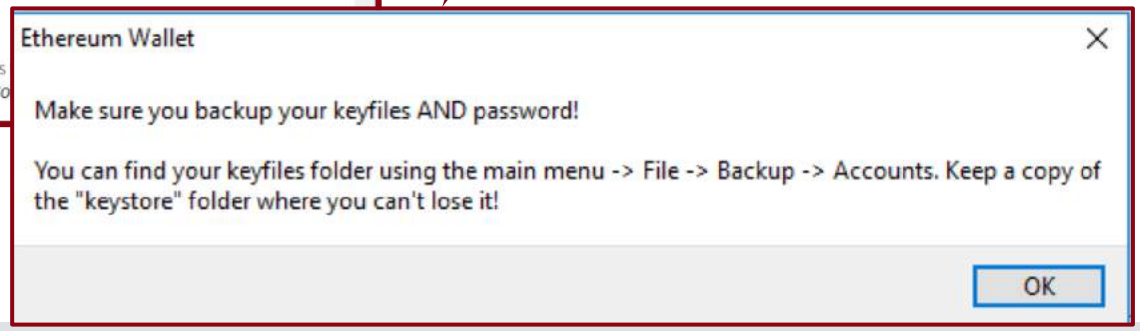
Consensus algorithm is a process used in the Blockchain network so that all nodes agree on a single state of the network

Ethereum Mist Wallet / Create new account



Password selection Tips

- Use a strong password Remember : Account Password cannot be changed!!
- Take a regular backup of the keyfiles



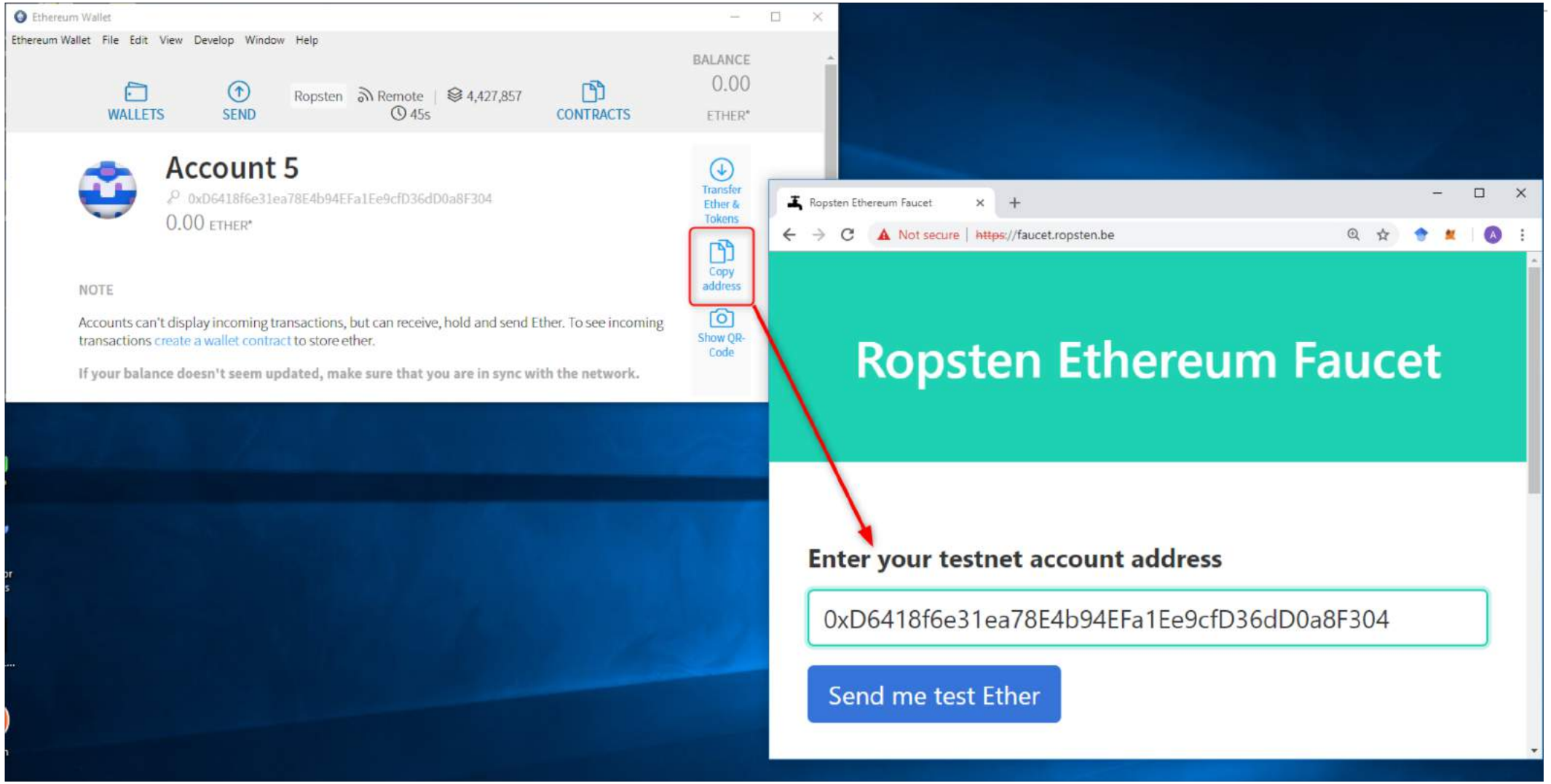
Ethereum Mist Wallet / Get rETHs

- To submit transactions to the Ropsten network, you need Ropsten ETHs (rETHs)
- There are three ways to get initial rETHs
  - Mining with your computer
  - Asking someone who mines on their computer to send you
  - Use of an Ethereum faucet [2]
    - https://faucet.ropsten.be/
    - https://faucet.metamask.io/
    - http://faucet.bitfwd.xyz/

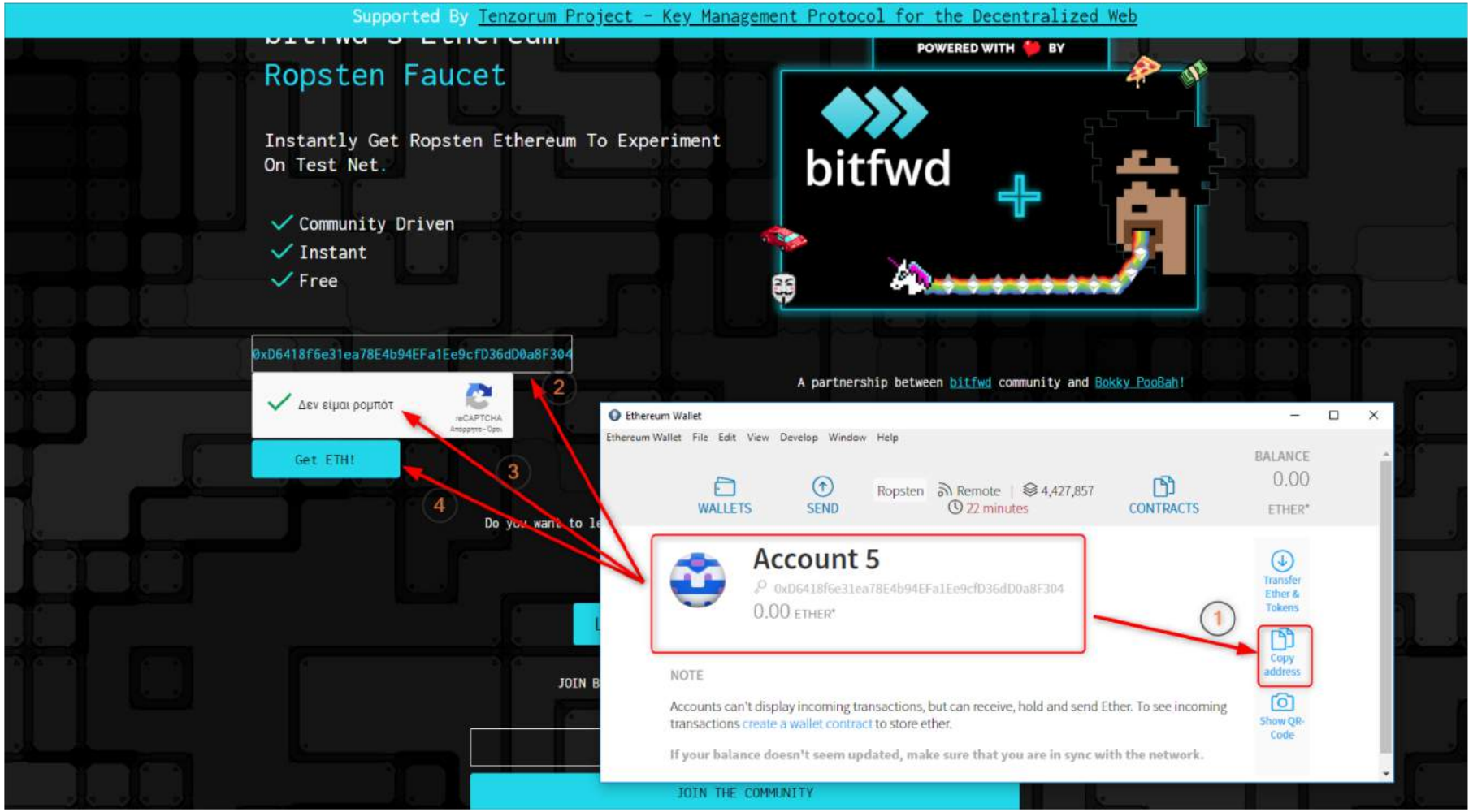
Miners are Ethereum nodes

- For each block of transactions, miners use computers to repeatedly and very quickly guess answers to a puzzle until one of them wins.
- If the miner finds an answer to the puzzle, the miner will be awarded ether and broadcast the block across the network for each node to validate and add to their own copy of the ledger

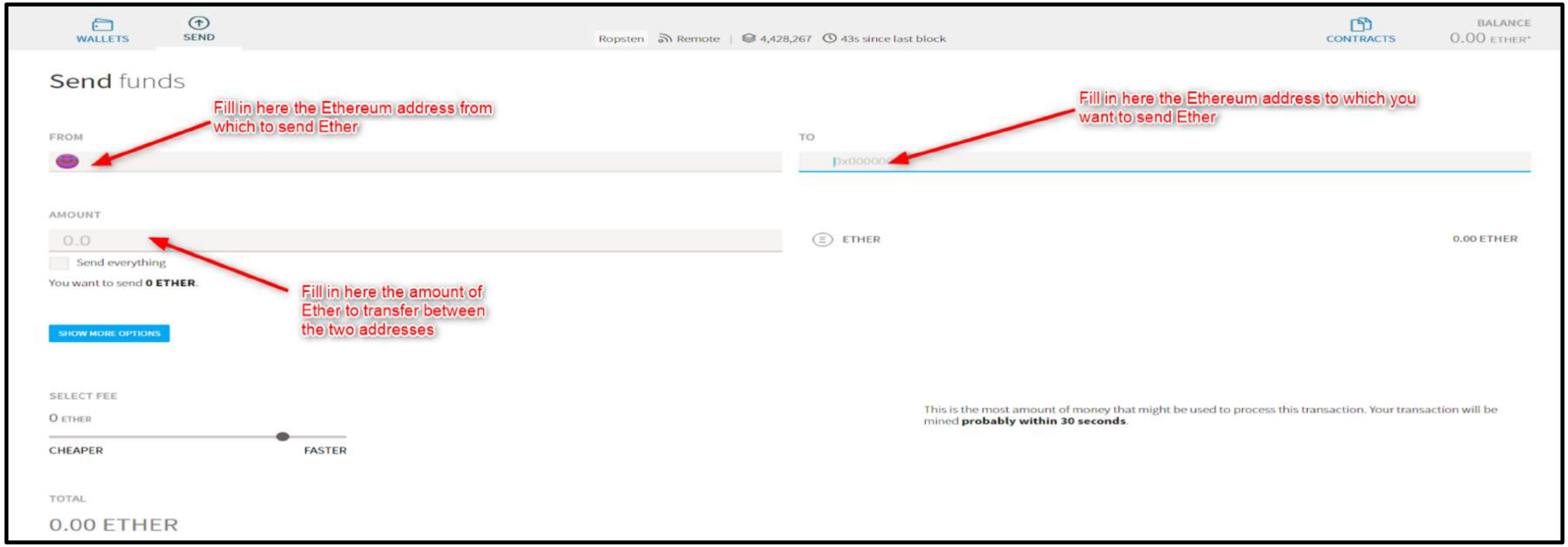
Ethereum Mist Wallet / Get rETHs / Ropsten Faucet



Ethereum Wallet / Mist Get rETHs / Bitfwd Faucet



Ethereum Mist Wallet / Send rETHs



II / My Ether Wallet

UNIC | Institute For the Future

CERTH CENTRE FOR RESEARCH & TECHNOLOGY HELLAS

iti Information Technologies Institute

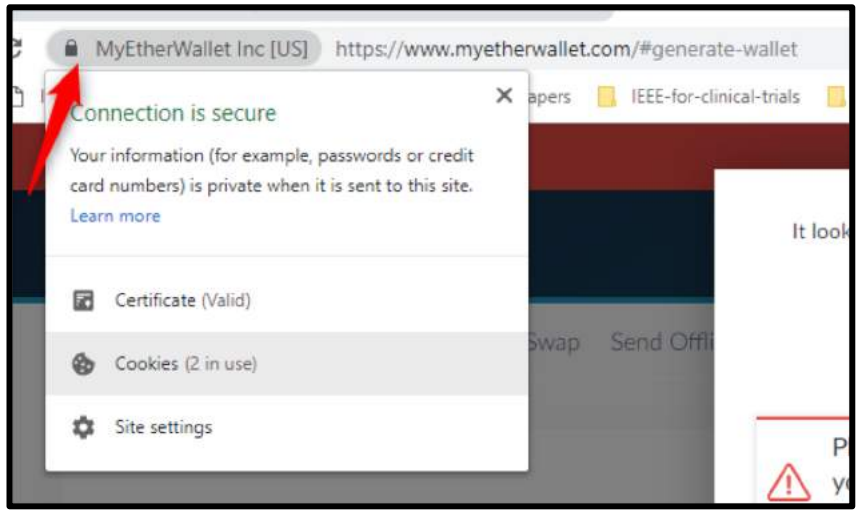
MyEtherWallet / Introduction

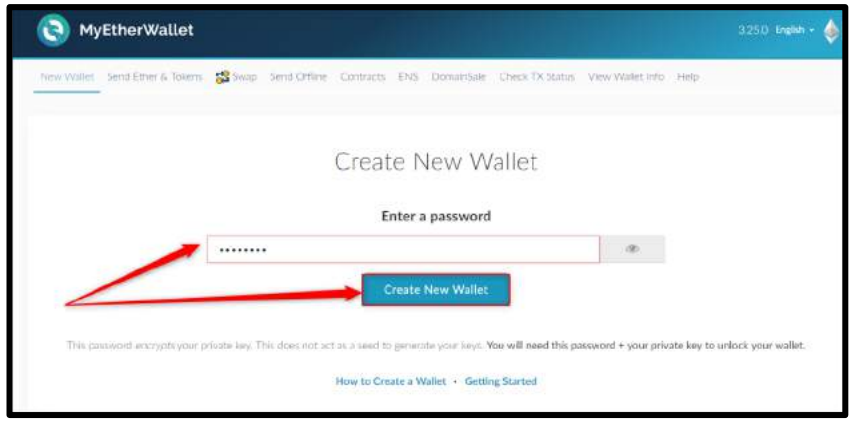
- What is MyEtherWallet (MEW)? [13]
  - It is a open source Ethereum wallet written in JavaScript. MEW source code is available here (https://github.com/MyEtherWallet/MyEtherWallet)
- It is available as
  - a web application (https://www.myetherwallet.com/)
  - a Chrome browser extension (https://chrome.google.com/webstore/detail/myetherwallet/nlbmnnijcnlegkjjpcfjclmcfggfefdm?hl=en)
  - Desktop application available to run MyEtherWallet offline and locally (https://kb.myetherwallet.com/offline/running-myetherwallet-locally.html)
- It is used to
  - store and transfer Ether and other ERC-20 tokens
  - interact with Smart Contracts

IMPORTANT
All the necessary data is generated and stored using the browser's machine and nothing is stored in MyEtherWallet servers !!!!

MyEtherWallet / Installation and usage

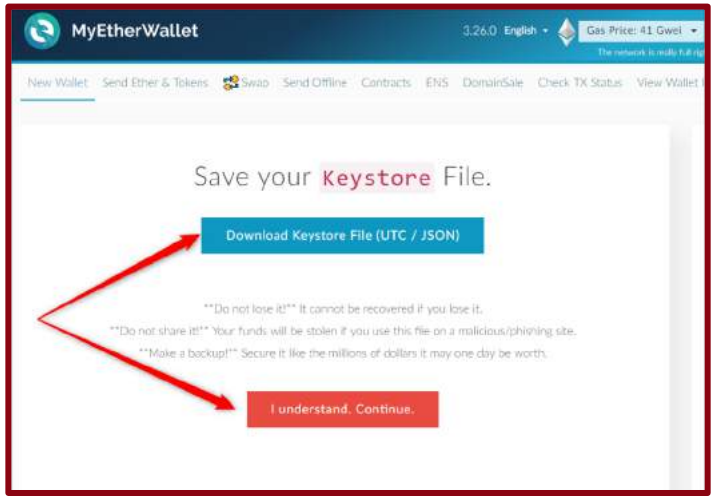
- Go to this link (https://www.myetherwallet.com/#generate-wallet)
- It is important to spend some time to read the security warnings, to bookmark the page, and to check the certificate of the website to make sure you are connected with a safe *https* connection.
- Make sure to keep your private key/password safe!!
- Enter a password (at least 9 characters long) and click on 'Create New Wallet'





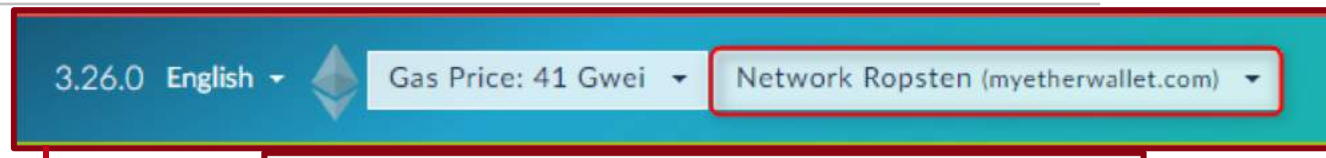
MyEtherWallet / Installation and usage

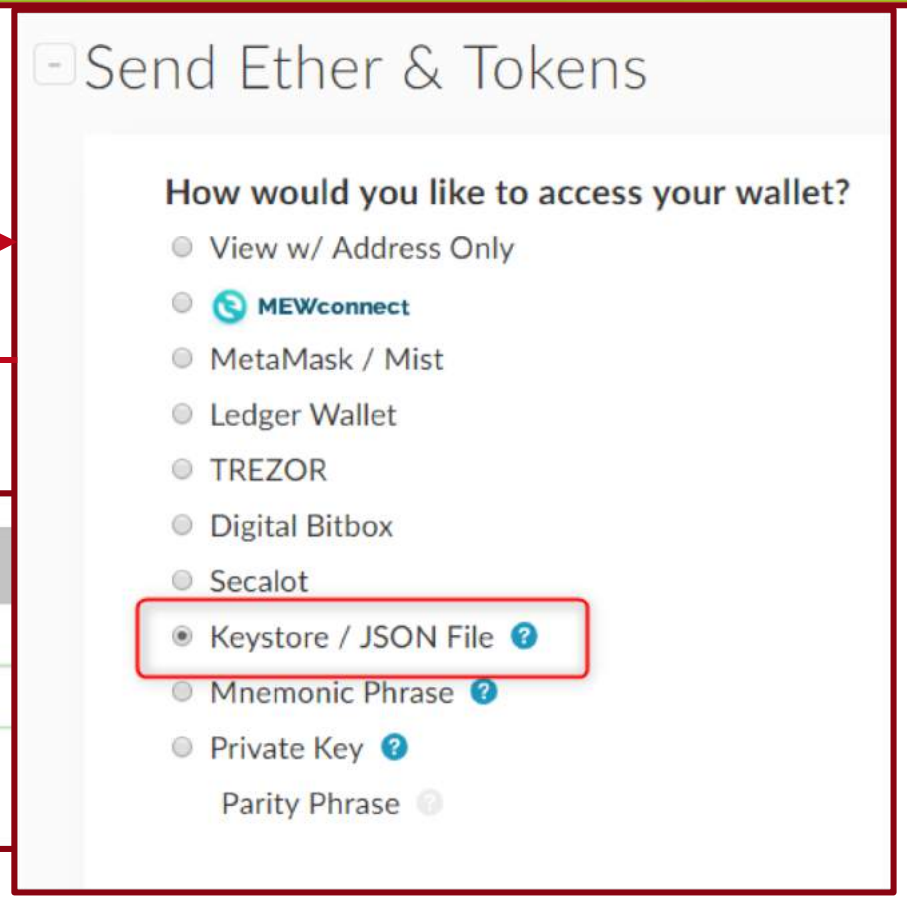
- Download and save keystore
  - Click on ‘Download Keystore File (UTC / JSON)‘
  - Click on ‘I understand. Continue.’
  - Do not share it !!
  - Take a backup of the keystore !!
- Save your private key (optional)
- You can also download and print your address and private key in Paper Wallet

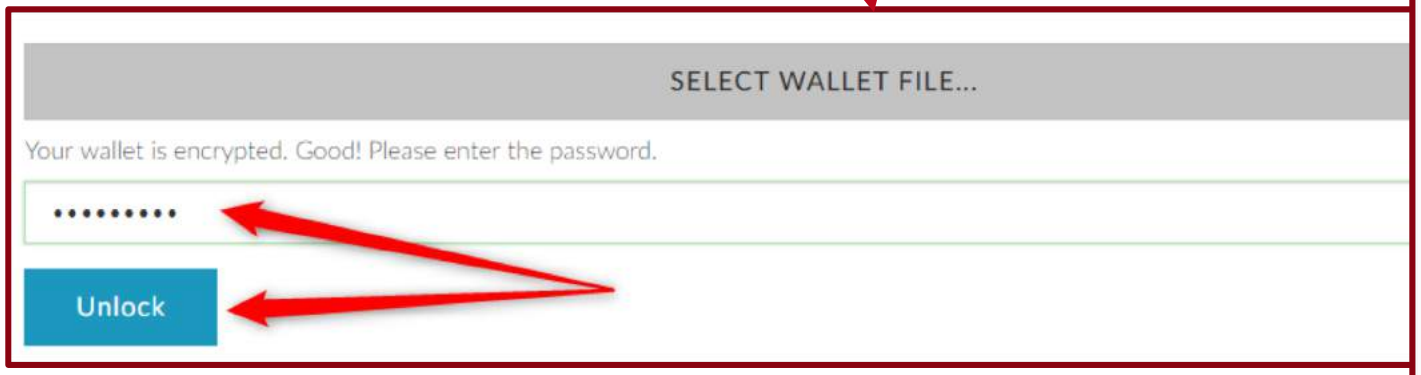


MyEtherWallet / Unlock your account to manage Ether

- From the top menu, go to ‘Send Ether & Tokens’
- From the top right dropdown, choose ‘Network Ropsten ’
- Choose an option on how to access your wallet, e.g. “Keystore / JSON file”;
- Browse to your keystore file stored before
- Enter your password
- Click on ‘Unlock’

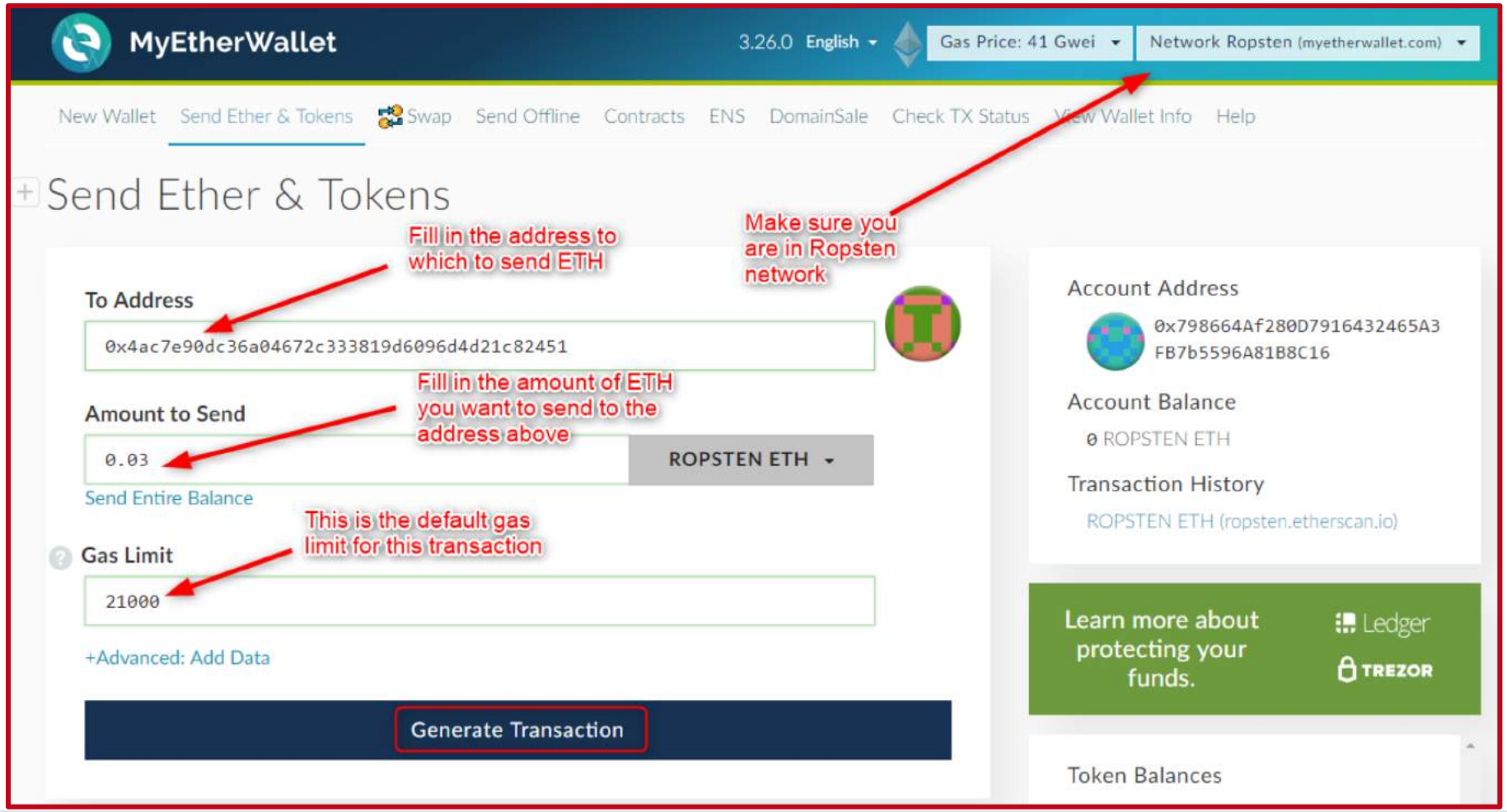






MyEtherWallet / Send Ether and Tokens

- Fill in the data and click on ‘Generate Transaction’



III / MetaMask

UNIC | Institute For the Future

CERTH CENTRE FOR RESEARCH & TECHNOLOGY HELLAS

iti Information Technologies Institute

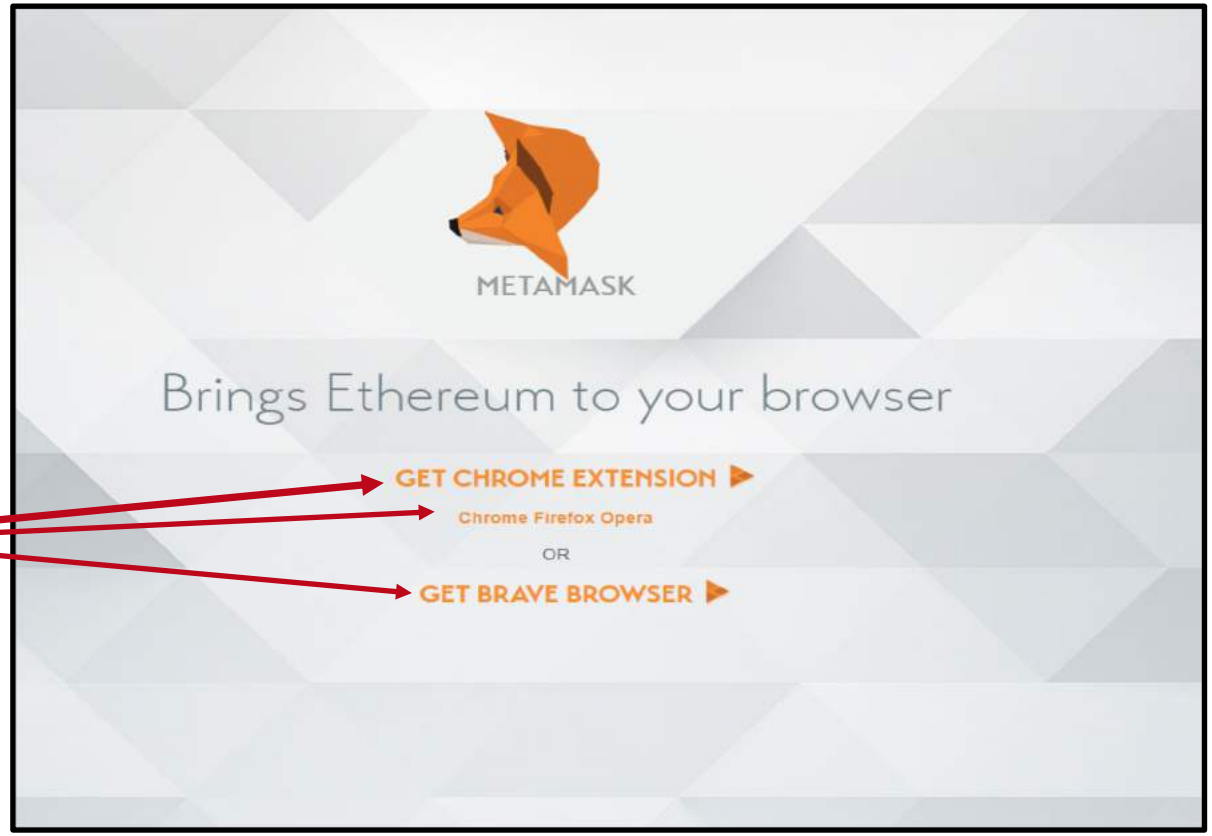
MetaMask

- MetaMask is a secure identity vault for Ethereum. It allows you to hold ether and other ethereum-based tokens in multiple accounts attached to this vault.
- It provides a user interface to manage Ethereum accounts and sign blockchain transactions.
- MetaMask turns your web browser into a Dapp browser allowing you to connect with decentralized applications (Dapps) deployed on the Blockchain network within the browser without running a full Ethereum node. [6]
- It can connect with many Ethereum Blockchain networks (main net, Ropsten testnet, Rinkeby testnet, Kovan testnet, local nodes)
- MetaMask is available as an extension for the
  - Google Chrome, Mozilla Firefox, Opera,Brave
- MetaMask source code is available here (https://github.com/MetaMask)

IMPORTANT
Your account vault is encrypted and stored locally on your browser. As such, your account data is not stored in MetaMask servers !!!!

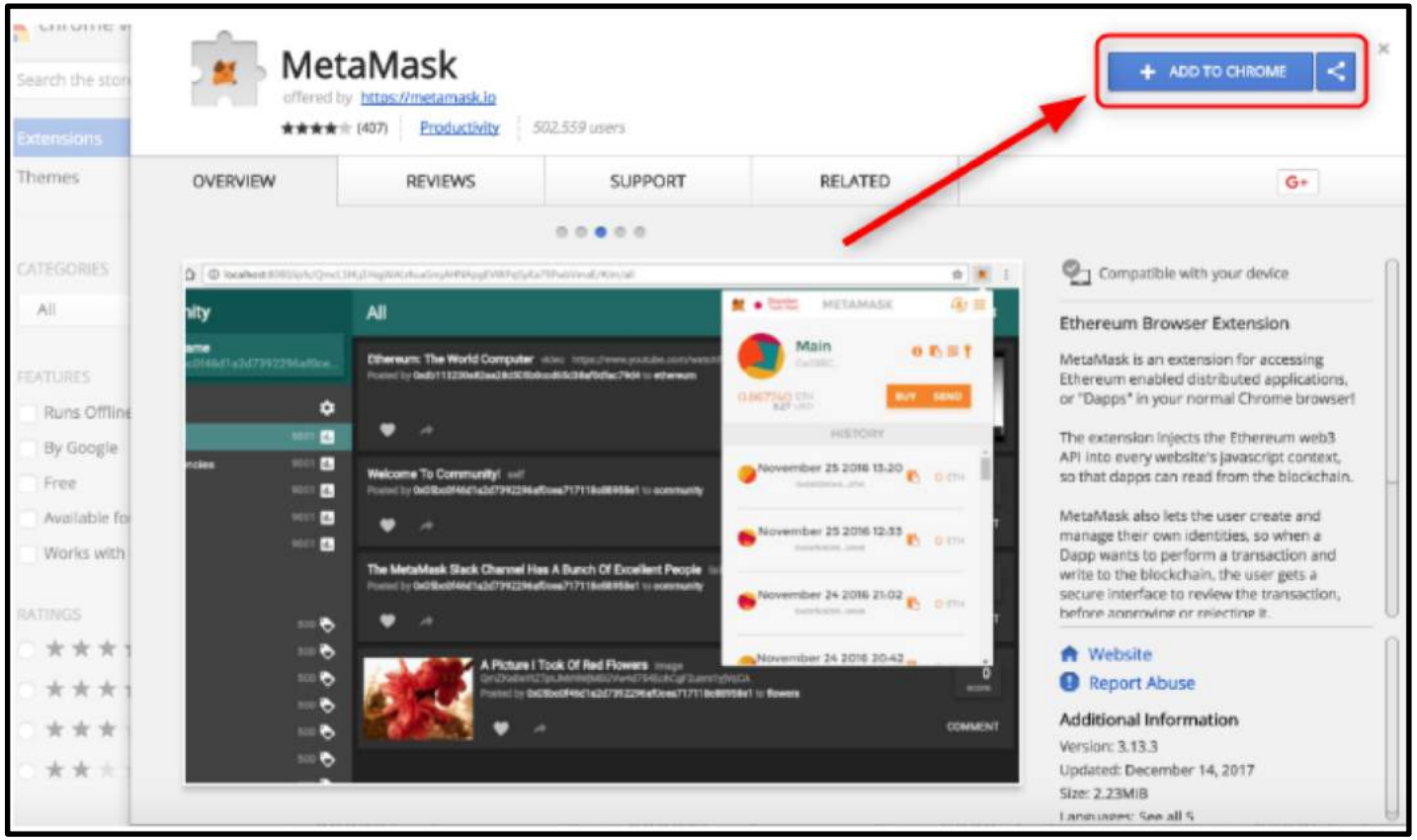
MetaMask / Installation [1/3]

- Go to this page (https://metamask.io/) to install MetaMask
- Select your preferred supported browser (Chrome, Firefox, Opera or Brave) with which to integrate MetaMask



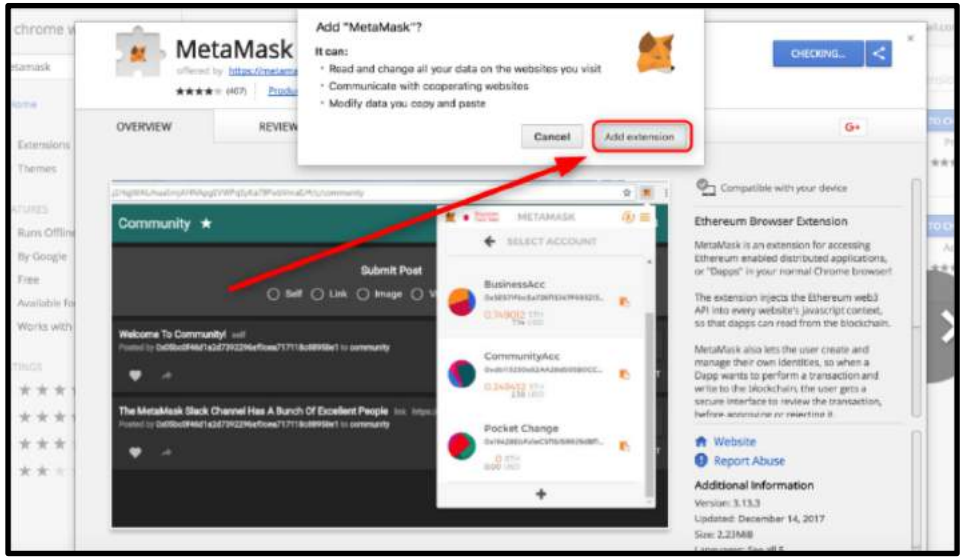
MetaMask / Installation [2/3]

- For Chrome, click ‘Add to Chrome’ to add MetaMask as a Chrome extension



MetaMask / Installation [3/3]

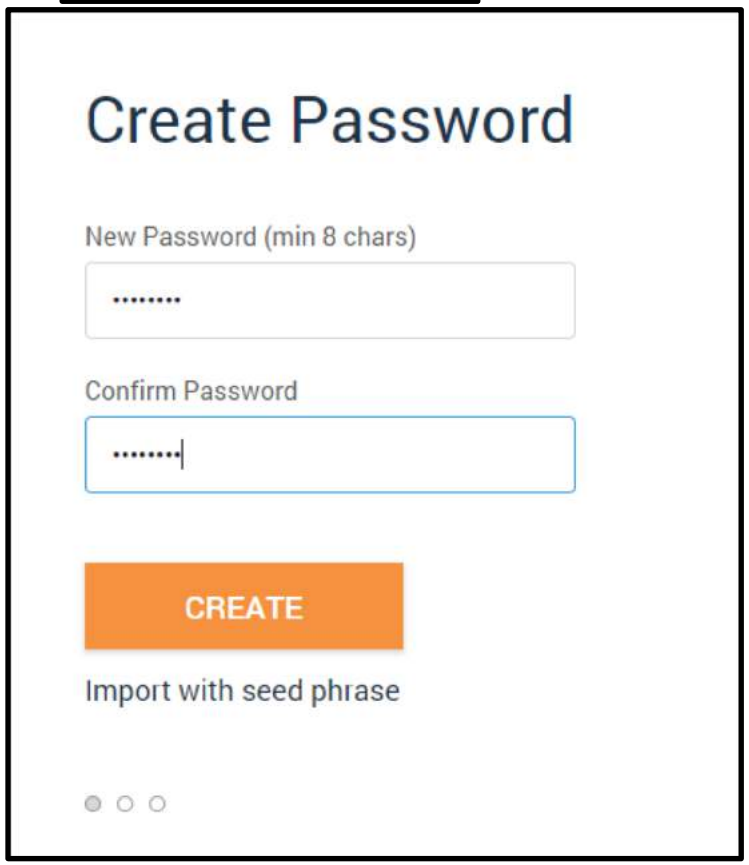
- Click ‘Add extension’



- A MetaMask button will appear at the top right in Chrome browser address bar

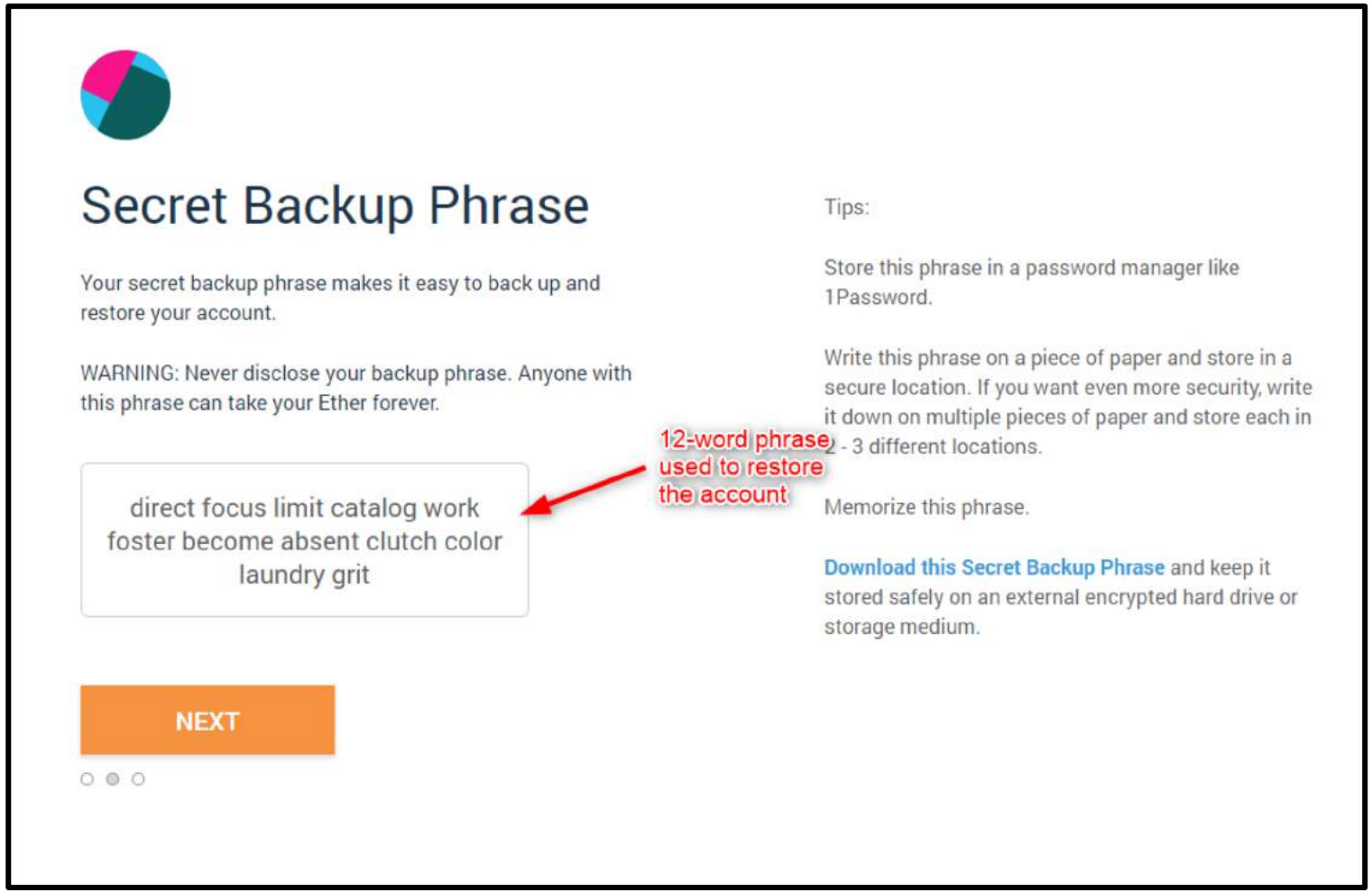
MetaMask / Create account [1/3]

- Hit on the MetaMask button located at the top right of the Chrome browser bar and follow the steps
- Create a Password. This will be used to protect your account. Your private key will be stored encrypted in the browser for security purposes. You will need this password to *unlock* your account.
- Follow the wizard steps (add image, accept, accept terms of use)



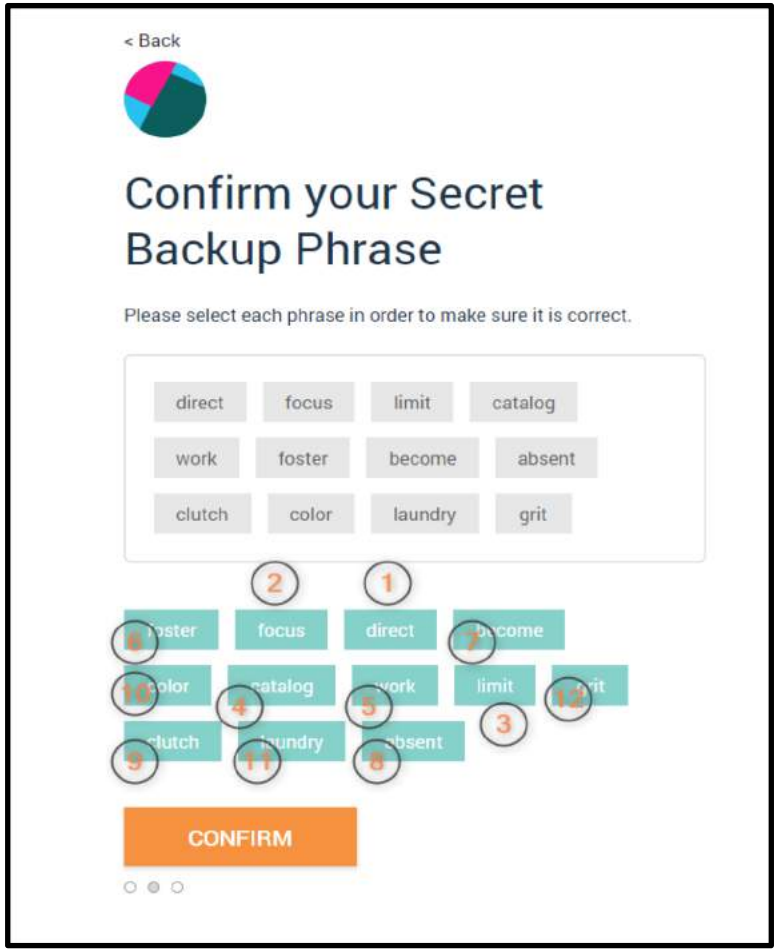
MetaMask / Create account [2/3]

- 12 word Secret phrase allows you to restore your account even from another MetaMask installation
- Make sure to properly backup the secret phrase !!
  - Memorise it
  - Write it in paper in many locations
  - Download and keep it in external storage



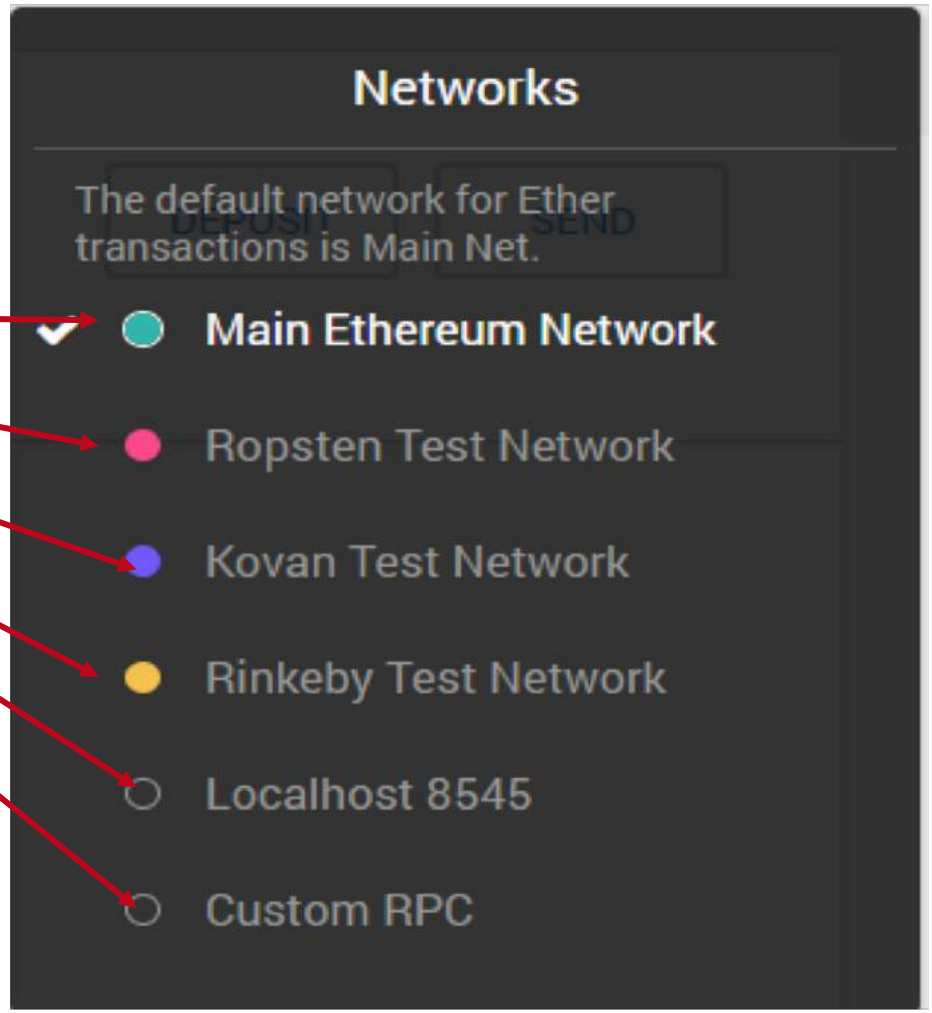
MetaMask / Create account [3/3]

- In the wizard, click on the words in the *same order in which they appear* in your saved passphrase so as to confirm your passphrase



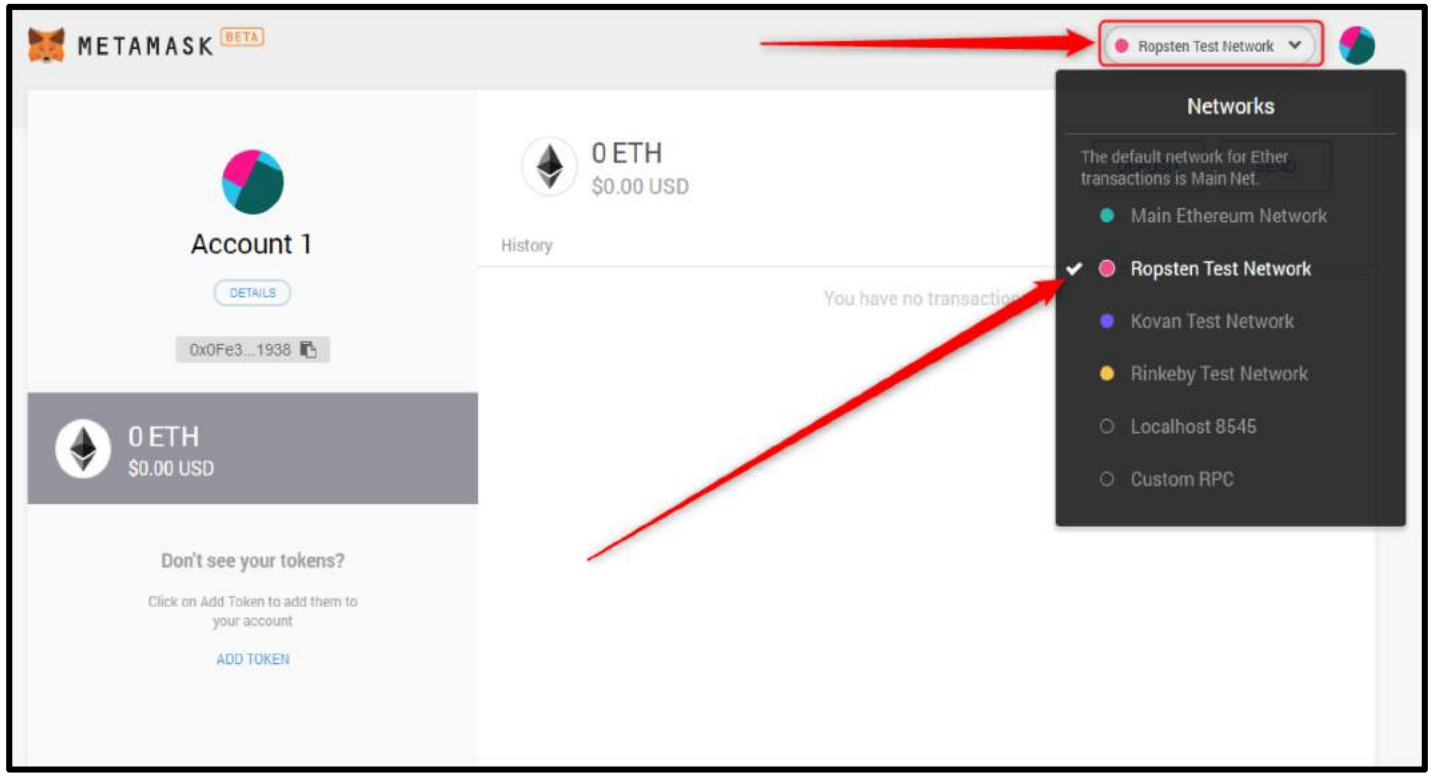
MetaMask / Network options to connect

- The alternative Blockchain networks with which you can connect



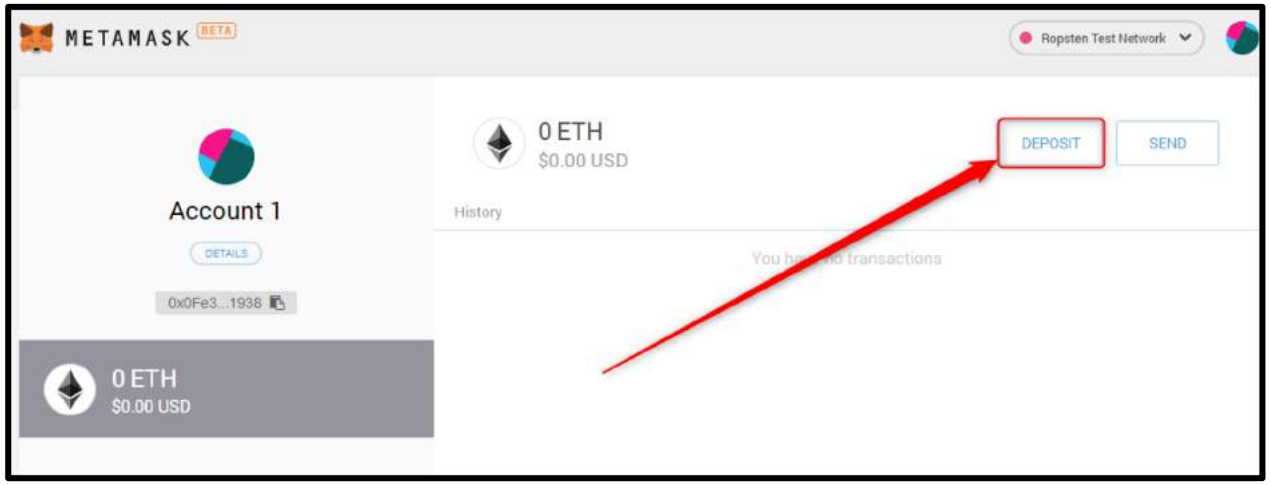
MetaMask / Connect with Ropsten Test Network

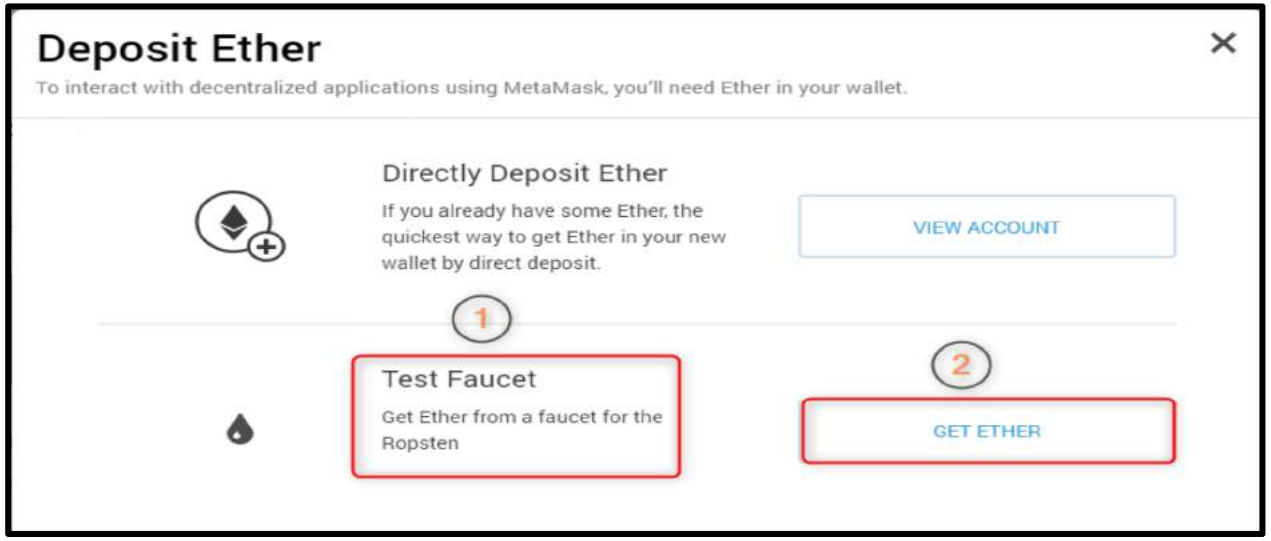
- In the main MetaMask screen, choose **Ropsten Test Network** from the drop down menu at the top right



MetaMask / Get Initial rETHs [1/4]

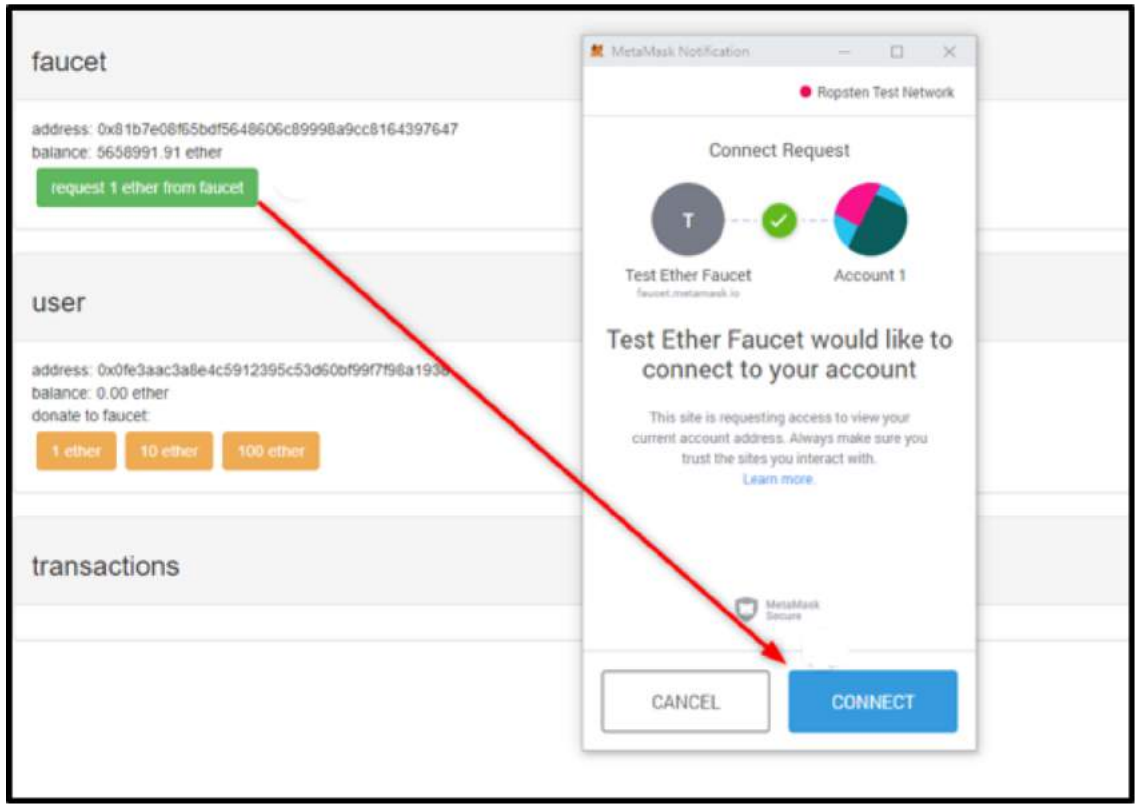
- You will need initial ETHER
  - To transfer ETHER (or any other token) from your account to another account
  - To deploy Smart Contracts
  - To interact with Smart Contracts and Dapps
- To get Initial ETHER
  - In the main MetaMask screen, click on Deposit at the top right
  - Choose 'Test Faucet' > 'GET ETHER' to connect with the MetaMask Faucet that can provide you with initial ETHs





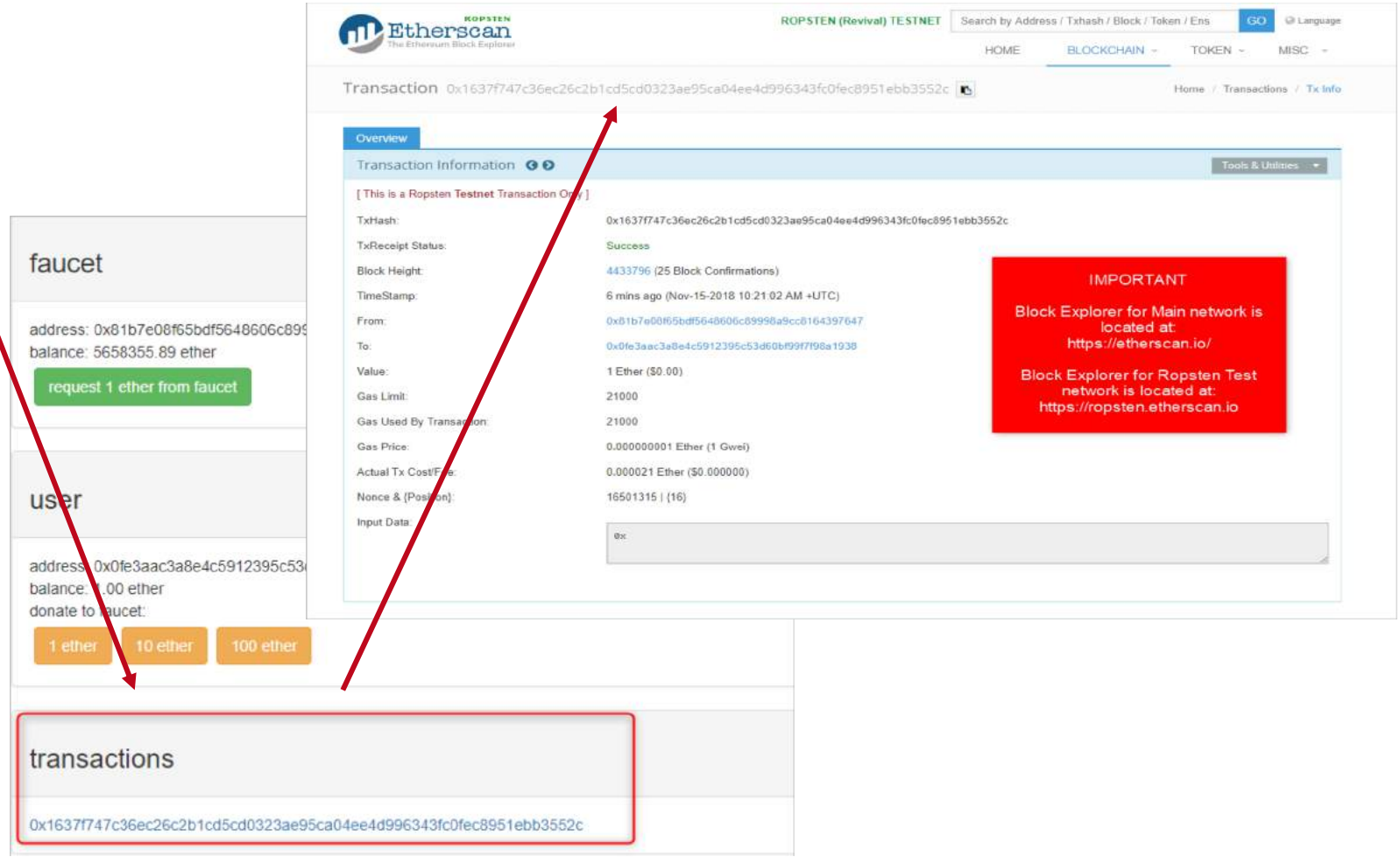
MetaMask / Get Initial rETHs [2/4]

- You will be redirected to the MetaMask Ether Faucet (https://faucet.metamask.io/)
  - Click on ‘request 1 ether from faucet’ so that the faucet address sends you 1 ETHER
  - Click ‘Connect’ on the MetaMask Notification



MetaMask / Get Initial rETHs [3/4]

- When the faucet address sends ETH to your address a Transaction is done in the Ropsten network
- The Transaction hash appears at the bottom of the MetaMask Ether Faucet screen
- You can see the details of this transaction in the Ropsten Etherscan Block Explorer by clicking on the Transaction hash

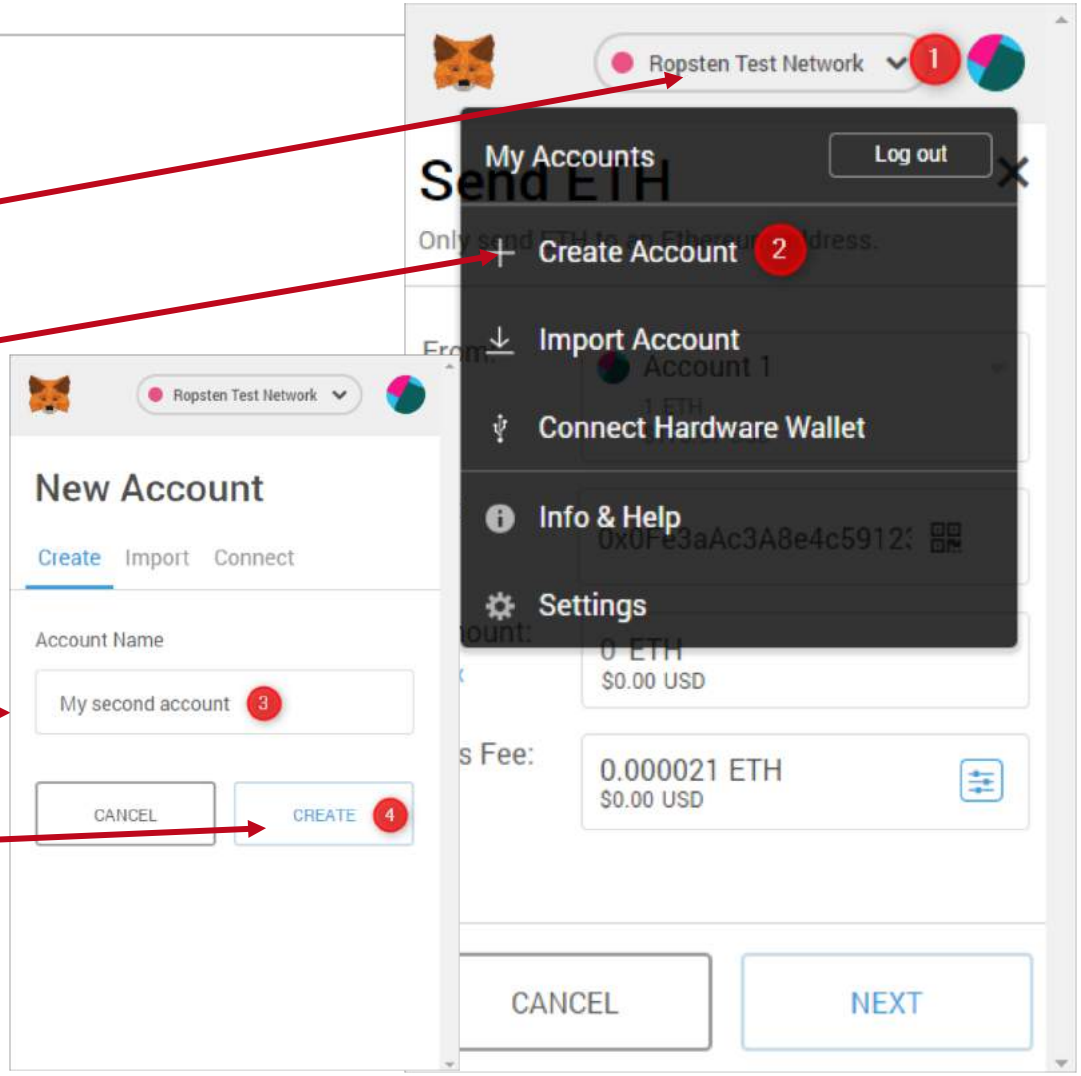


MetaMask / Get Initial rETHs [4/4]

- The transferred ETH appears now in your account

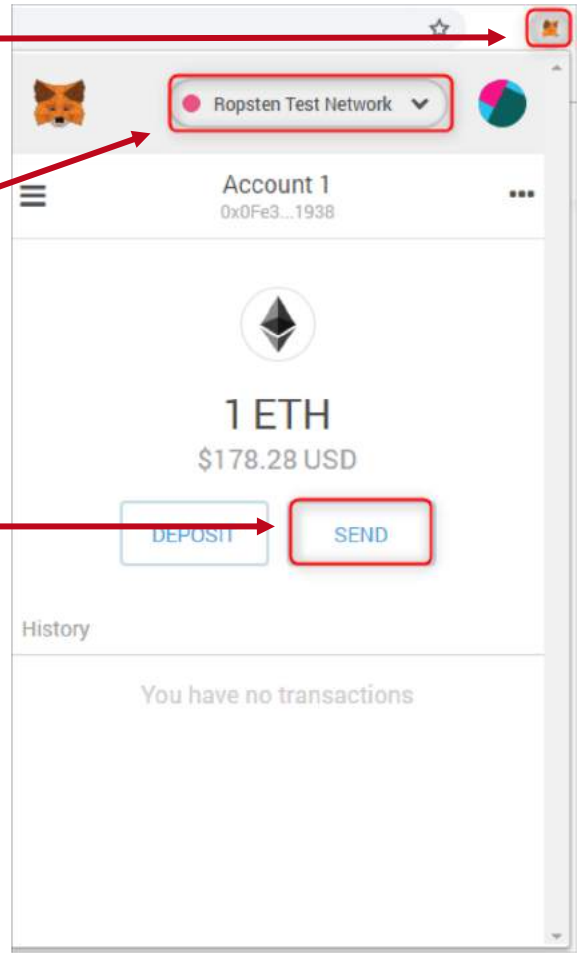
MetaMask / Send ETHERs [1/4]

- First create a new account
  - Click your icon at the top right of the MetaMask window
  - Click on ‘Create Account’ (You can also import an external account or connect with a hardware wallet)
  - Give your new account a name
  - Click on ‘Create’



MetaMask / Send ETHERs [2/4]

- To send ETH into another address, click on the MetaMask button at the top right in Chrome address bar
- Make sure that you are in the Ropsten Test Network
- Click on the button ‘SEND’



MetaMask / Send ETHERs [3/4]

- In the **From** field, choose your first address
- In the **To** field, choose your second address
- In the **Amount** field, add the amount in Ether to be transferred from your first to your second account, e.g. 0.03 ETH

IMPORTANT

- The amount of ETHER to send should not be more than the balance on your account !!
- Do not forget to take the transaction fees into account !!

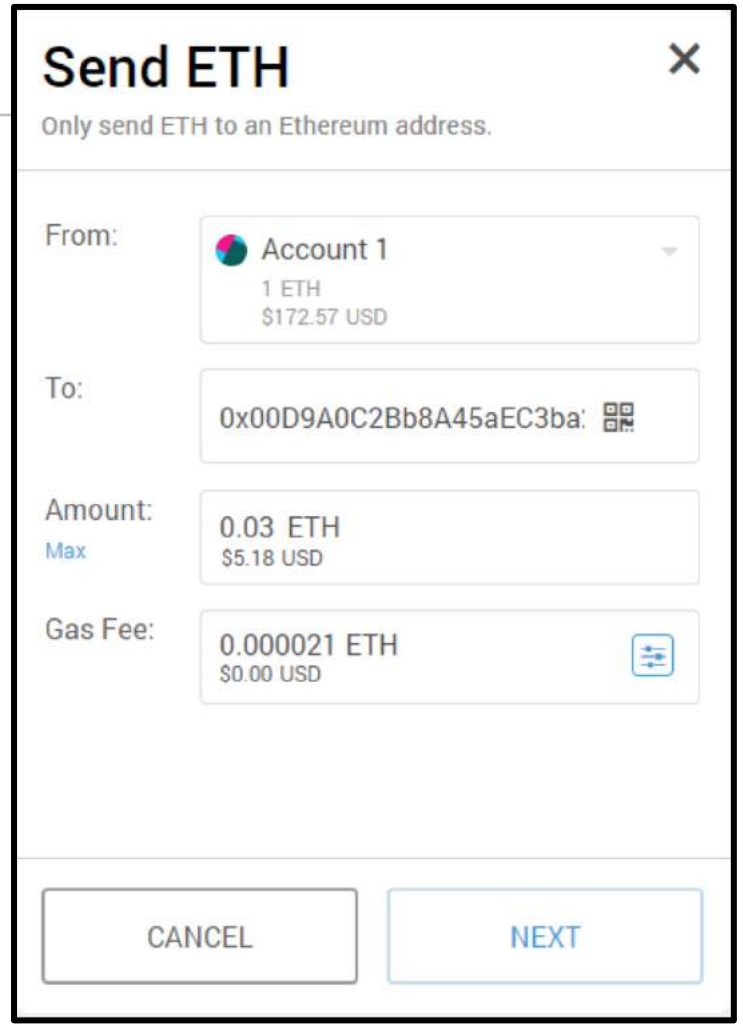
AMOUNT + Gas fee <= account balance

Gas: a unit for the processing cost for a particular operation a computer executes

Gas price: the price in ETHER to pay to the miner per computational step

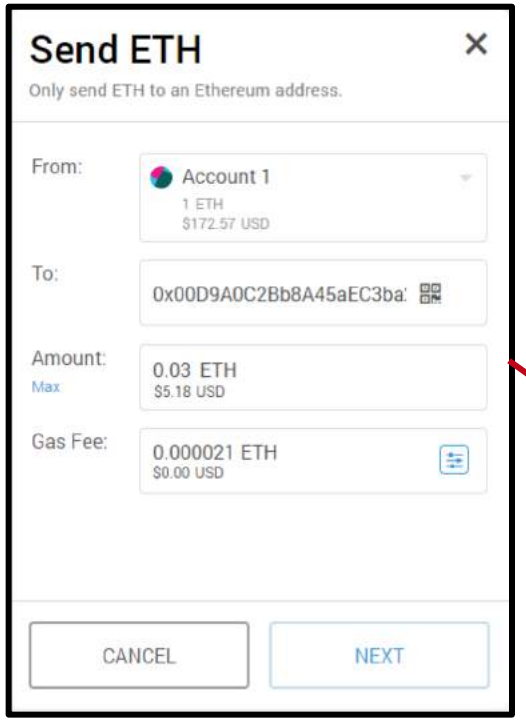
Gas limit: the maximum amount of gas we want to pay for a transaction

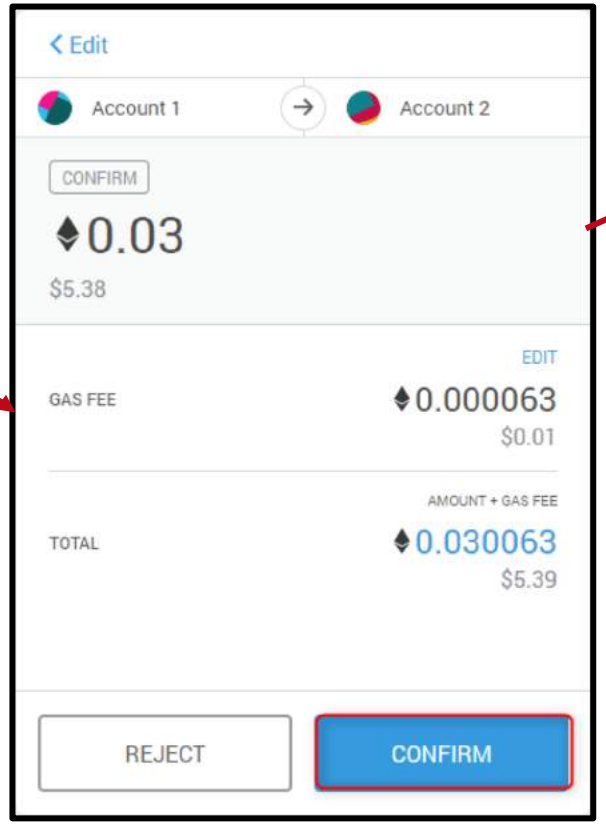
Gas fee = Gas limit * Gas price [5][9]

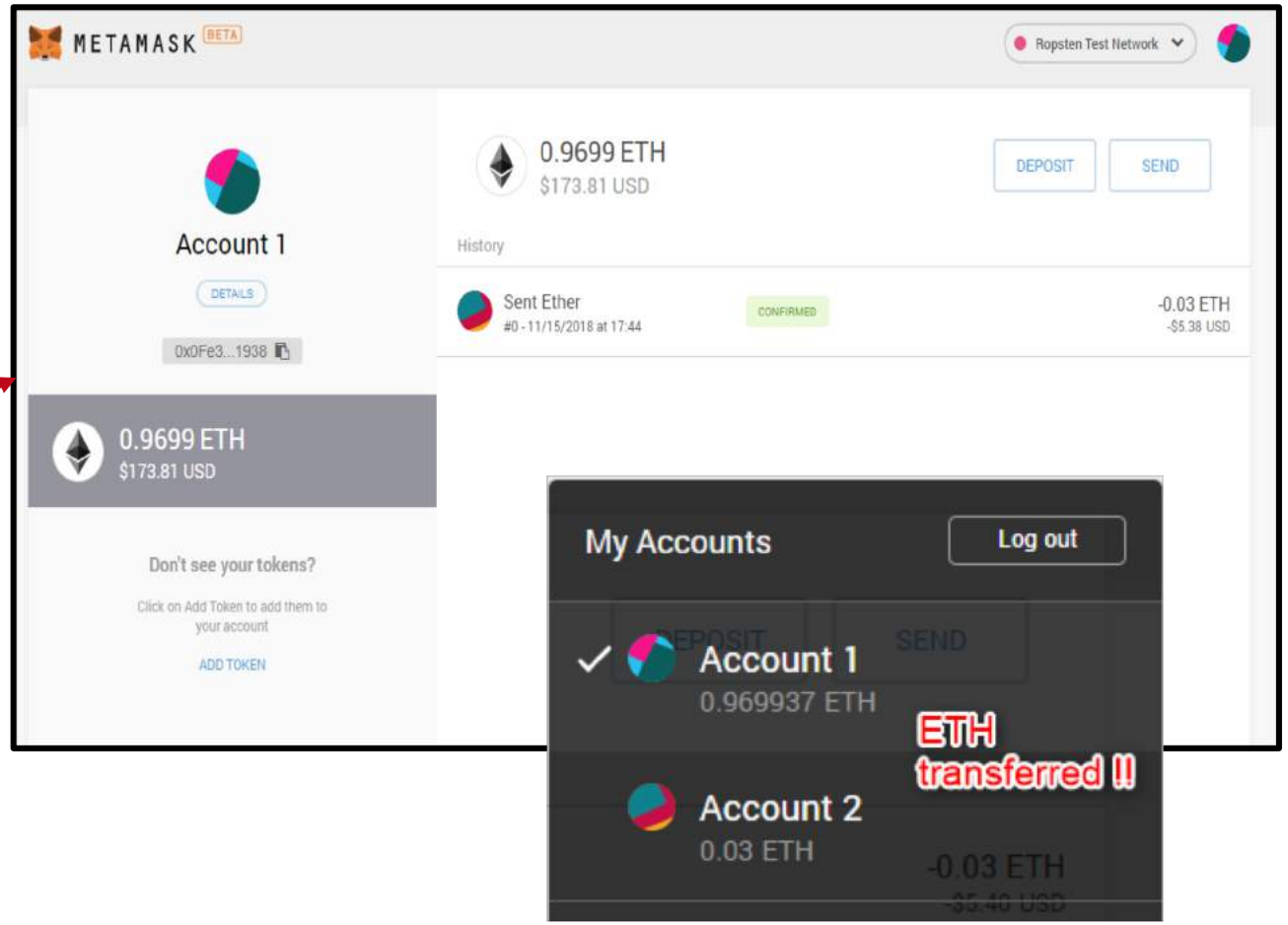


MetaMask / Send ETHERs [4/4]

- Click ‘Next’
- Click ‘Confirm’
- Wait until the transaction gets confirmed







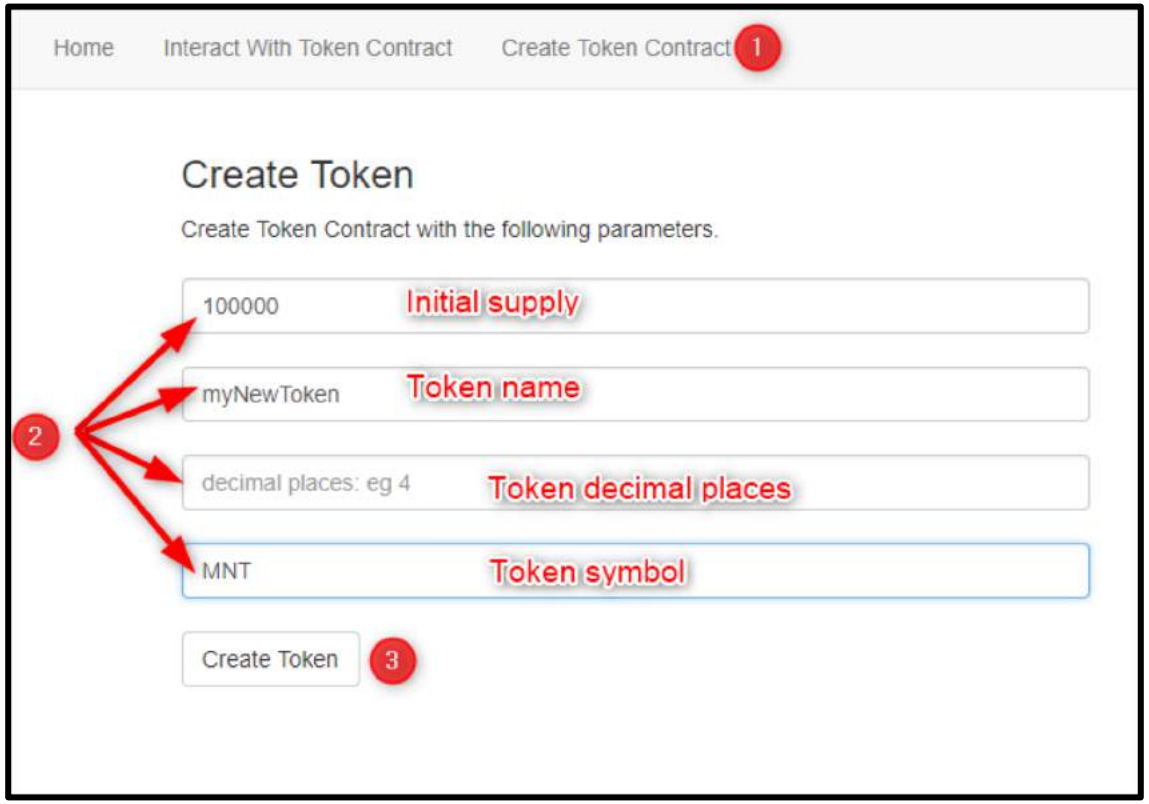
UNIC Institute For the Future

CERTH CENTRE FOR RESEARCH & TECHNOLOGY HELLAS

iti Information Technologies Institute

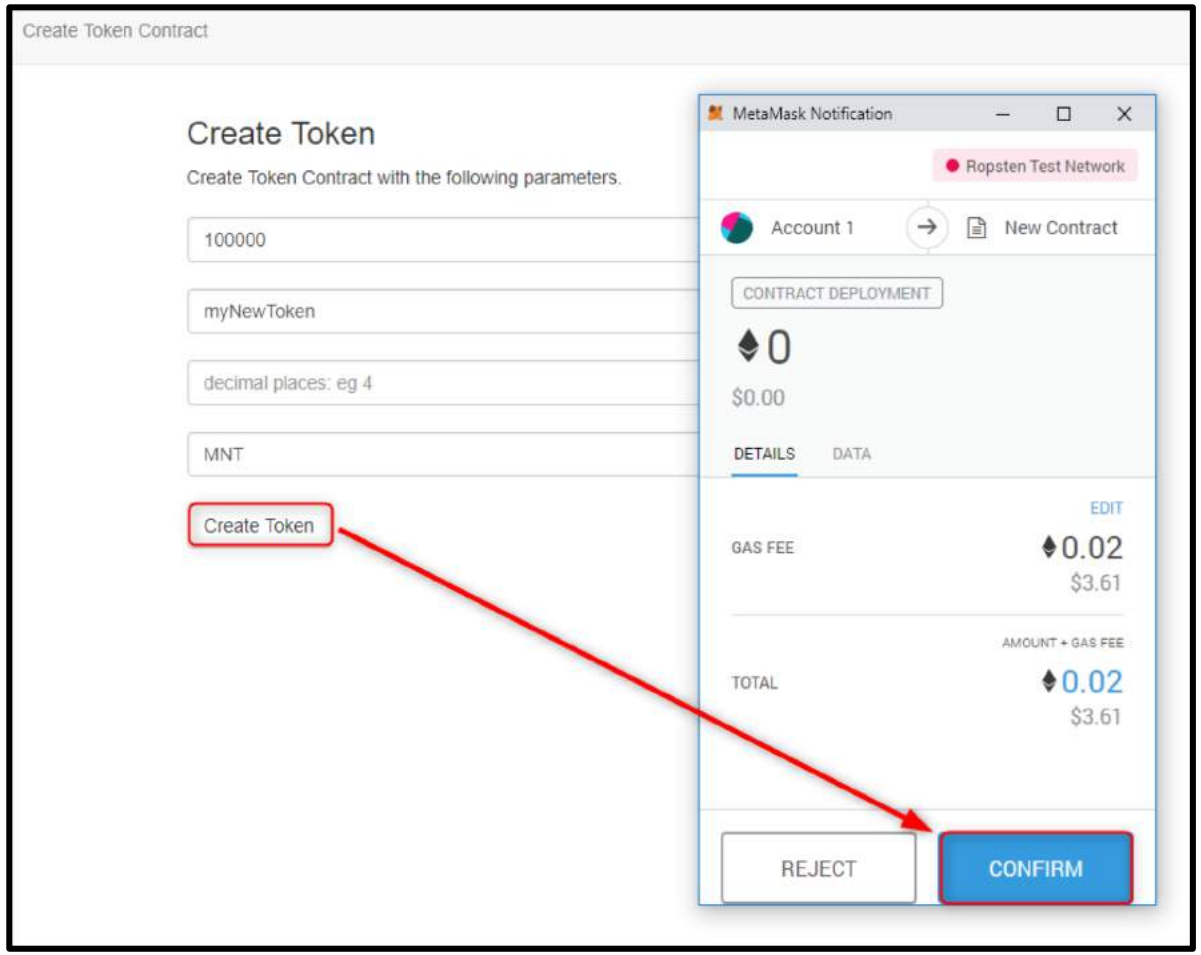
MetaMask / Integration with Dapp [1/4]

- Visit the page (https://tokenfactory.surge.sh/#/)
- This website has access to the Ethereum Blockchain network via an API called Web3 that is injected to the website by MetaMask
- From the top menu, click on 'Create Token Contract'
- Provide the input parameters for the Smart Contract
- Click on 'Create Token'



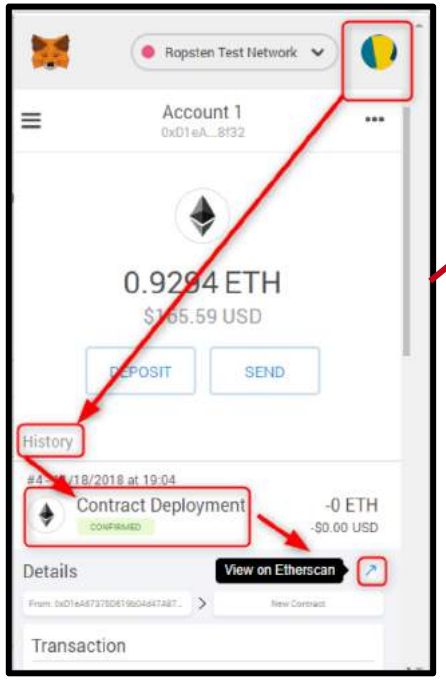
MetaMask / Integration with Dapp [2/4]

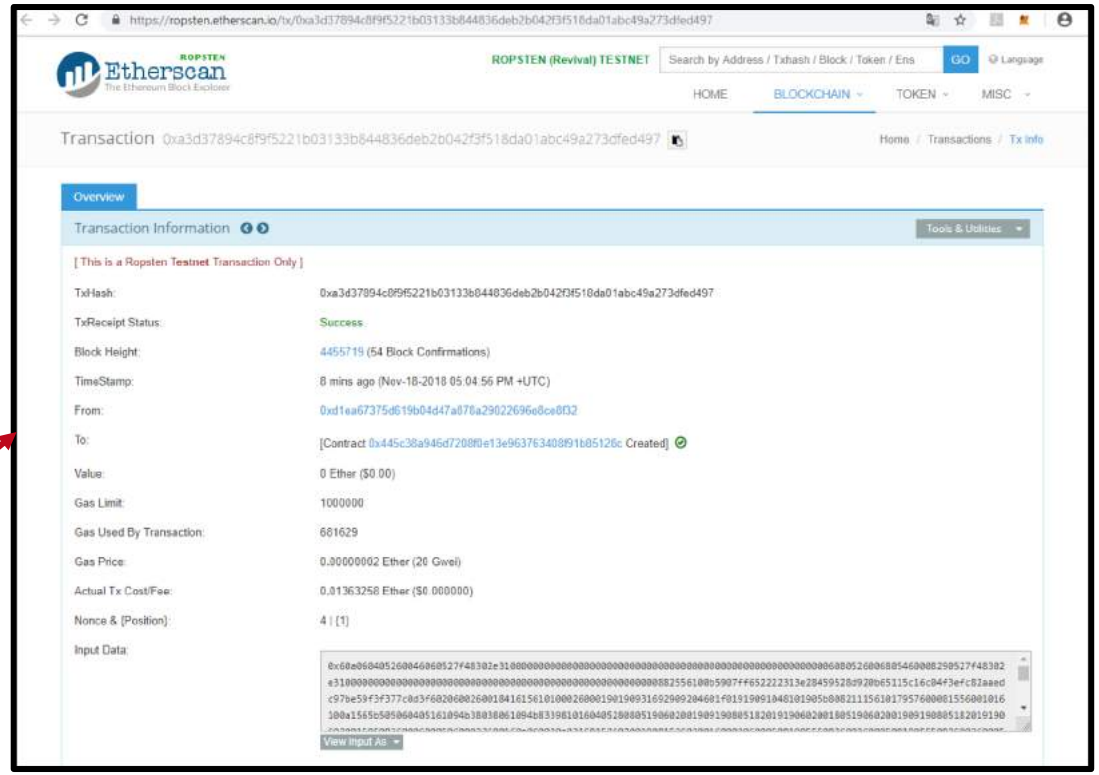
- Upon clicking on ‘Create Token’, web3 sends a transaction to the Ethereum Blockchain network
- MetaMask window is automatically launched and asks users to confirm, i.e. to *sign* the transaction
- After submitting the transaction, the transaction is added to a Block and the Block number increases by 1.
- A MetaMask notification at the bottom right will inform you that the transaction was confirmed by the network.



MetaMask / Integration with Dapp [2/4]

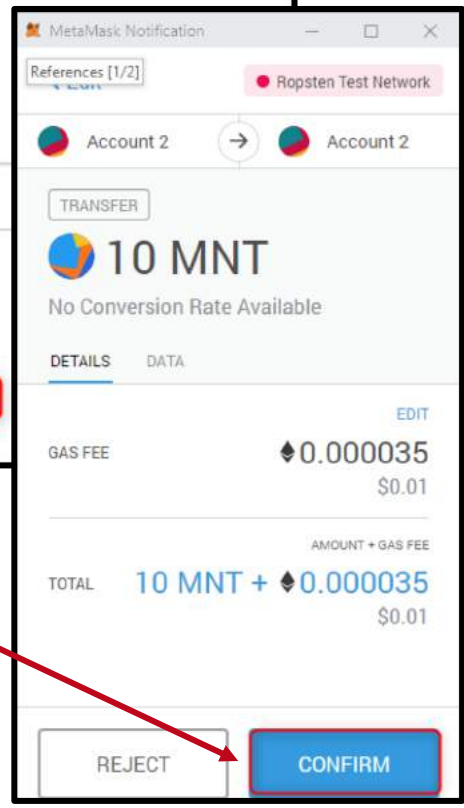
- To see the transaction on Ropsten Ethescan
  - Click on MetaMask button
  - Click on the first account icon
  - Under History, click on the last transaction (appears first in the History)
  - Click on the icon ‘View on Etherscan’

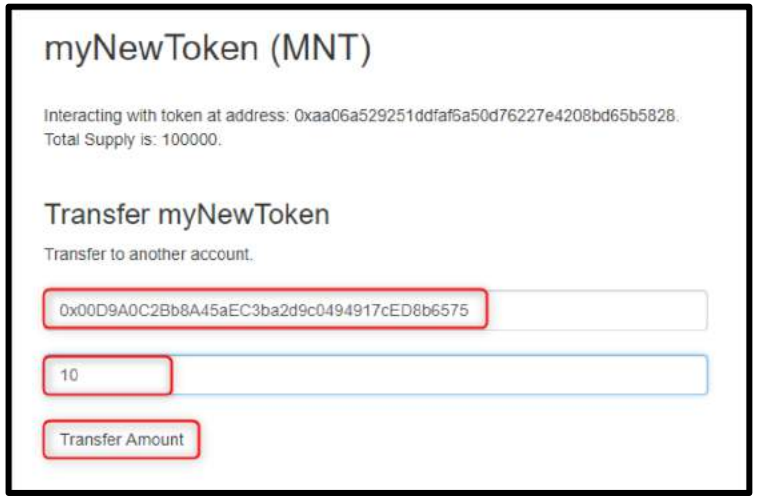




MetaMask / Integration with Dapp [3/4]

- Copy the first address in your MetaMask wallet and paste it in the 'Check Balance' field to check its balance, it should be 100000 (i.e. equal to the initial token supply).
- Transfer myNewToken to the second address in your MetaMask wallet.
  - In MetaMask window, switch to Account2
  - Copy Account2 address
  - Paste it to Field under 'Transfer myNewToken'
  - Fill in the amount e.g.10 MNTs
  - Click on 'Transfer Amount'
  - Confirm the transaction
  - Check the new balance of Account2 (for the MNT token) !!!

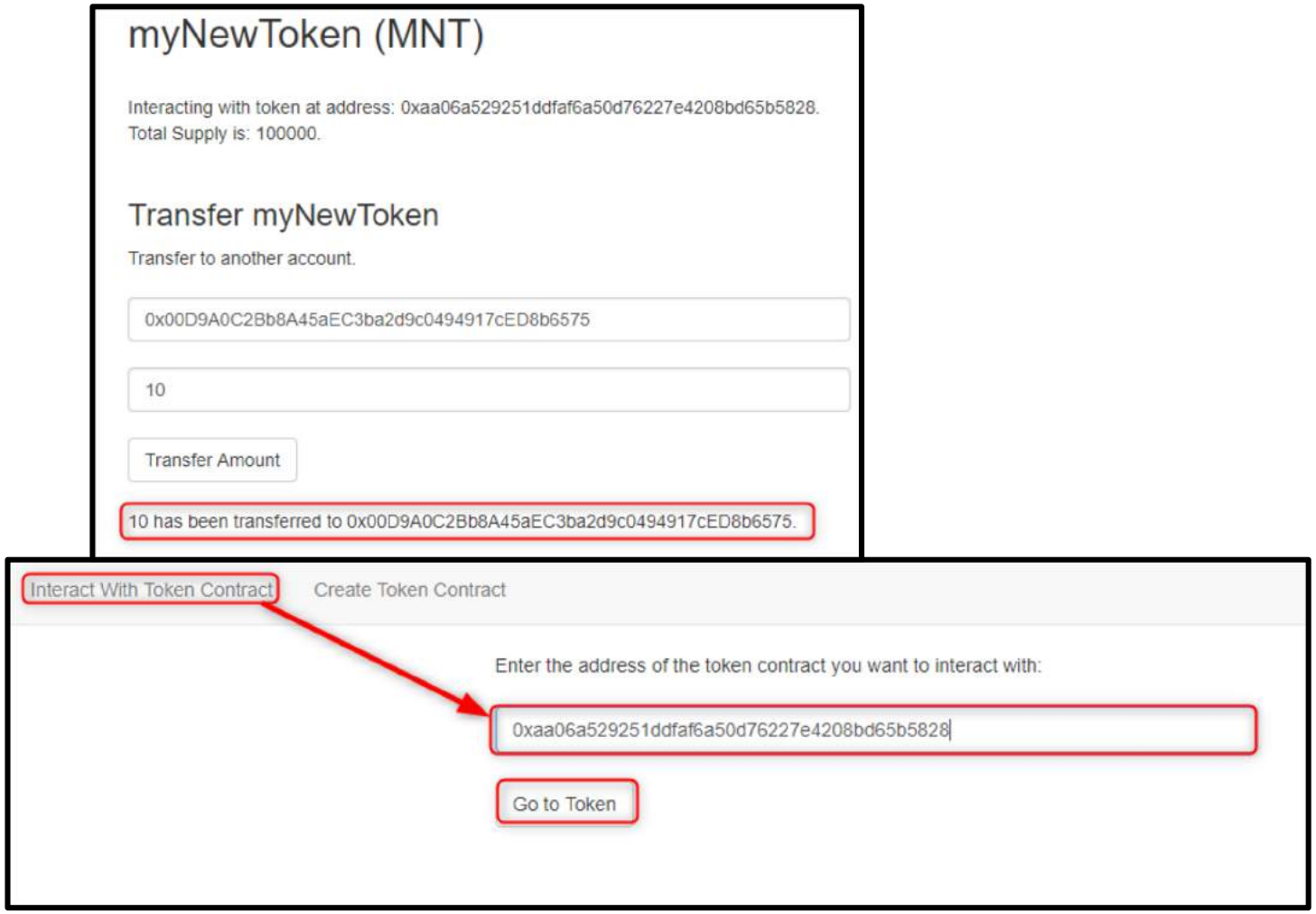




MetaMask / Integration with Dapp [4/4]

- Transaction is confirmed and its details are available on Ropsten Etherscan and on MetaMask History of the first address

- Each time you want to interact with the new Token contract
  - From the top menu, go to 'Interact With Token Contract'
  - Fill in the contract address which you can find from Ropsten Etherscan searching with your account address
  - Click 'go to contract'



IV / Introduction to Smart Contracts

UNIC | Institute For the Future

CERTH CENTRE FOR RESEARCH & TECHNOLOGY HELLAS

iti Information Technologies Institute

Contents

“To be clear, at this point I quite regret adopting the term "smart contracts". I should have called them something more boring and technical, perhaps something like “persistent scripts". ”

Vitalik Buterin

What is a smart contract?

- Code running on the blockchain
- Smart contracts can store data
- Smart Contracts are represented by:
  - Address
  - Application Binary Interface(ABI)
- Every participant to the blockchain network has access to all smart contracts
- Smart contracts on Ethereum are written in Solidity
- Solidity is a contract-oriented, high-level language for implementing smart contracts
- Solidity is similar to Javascript
- Contracts in Solidity are similar to classes in object-oriented languages.

V / Solidity and Remix IDE

UNIC | Institute For the Future

CERTH CENTRE FOR RESEARCH & TECHNOLOGY HELLAS

iti Information Technologies Institute

Introduction to Solidity(1/4)

- Value Types
  - int / uint
    - Fixed size arrays
    - Dynamic size arrays
  - Bool (true, false)
  - bytes (e.g. 0x6b6f73746173)
    - Fixed size arrays
    - Dynamic size arrays
  - String
    - Dynamic size arrays
  - Address (e.g. 0xfEc7a9042ee75C78cf59e68f215b1E69ff04bb3e)
    - Holds a 20 byte value with size of an Ethereum address

Introduction to Solidity(2/4)

- Reference Types
  - arrays
    - Arrays can have a compile-time fixed size or they can be dynamic
    - E.g. bytes32[] names
  - structs
    - Used to group several variables
    - struct Funder {
      address addr;
      uint amount;
      }
- Function Types
  - Constructors
  - Call (Getters)
  - Transaction (Setters)
  - function <name>(<parameter types>) {public | internal | external} [pure | constant | view | payable] [returns (<return types>)]{ <do some staff> }

Introduction to Solidity(3/4)

- Mappings
  - Mappings can be seen as hash tables which are virtually initialized such that every possible key exists and is mapped to a value whose byte-representation is all zeros
  - E.g. mapping(bytes32=>bool) public
- Modifiers
  - modifier <name>{ <do some staff> }
  - The function body is inserted where the special symbol `_;` in the definition of a modifier appears. This means that if the owner calls this function, the function is executed and otherwise, an exception is thrown
  - E.g.

```
modifier onlyOwner {
        require(
    msg.sender == owner,
            "Only owner can call this function."
  );
  _;
}
```

Introduction to Solidity(4/4)

- Special variables and functions
  - *msg.sender* sender of the message
  - *msg.value* number of wei sent with the message
  - *require(bool condition)* reverts if the condition is not met - to be used for errors in inputs or external components.
  - *assert(bool condition)* invalidates the transaction if the condition is not met - to be used for internal errors.
  - *Keccak256('some text')* outputs the hash value of 'some text'
- Enum Types
- Events
- Inheritance

General Structure of a Smart Contract

```
pragma solidity ^0.4.<XX>;
contract <Name> {
  <Declare the state variables>
  constructor(<args>) public {
      <do some staff>
  }
  function <name> (<args>)  {declaration} {
      <do some staff>
  }
  function <name> (<args>)  {declaration} returns
(<return types>) {
      <do some staff>
     <return something>  }}
```


UNIC | Institute For the Future
CERTH CENTRE FOR RESEARCH & TECHNOLOGY HELLAS
iti Information Technologies Institute

Example Smart Contract

```
pragma solidity ^0.4.25;
contract Voting {
  mapping (bytes32 => uint) public votesReceived;
  bytes32[] public candidateList;

  constructor(bytes32[] candidateNames) public {
    candidateList = candidateNames;
  }
 function voteForCandidate(bytes32 candidate)
public {
    require((validCandidate(candidate) == true));
    votesReceived[candidate] += 1;
  }
 function totalVotesFor(bytes32 candidate) public
view returns (uint) {
    require((validCandidate(candidate) == true));
    return votesReceived[candidate];
  }
 function validCandidate(bytes32 candidate) public
view returns (bool) {
    for(uint i = 0; i < candidateList.length; i++) {
      if (candidateList[i] == candidate) {
        return true;
      }
    }
    return false;
  }

}
```

Structure of Smart Contract

- State variables (contain persistent data)
  - bytes32[] candidateList
  - mapping(bytes32 => uint) public votesReceived;
- Functions
  - Constructor function
    - constructor(bytes32[] candidateNames) public {

```
        candidateList = candidateNames;
    }
```

  - Transaction functions (Setters)
    - function voteForCandidate(bytes32 candidate) public {

```
        require((validCandidate(candidate) == true));
        votesReceived[candidate] += 1;
    }
```

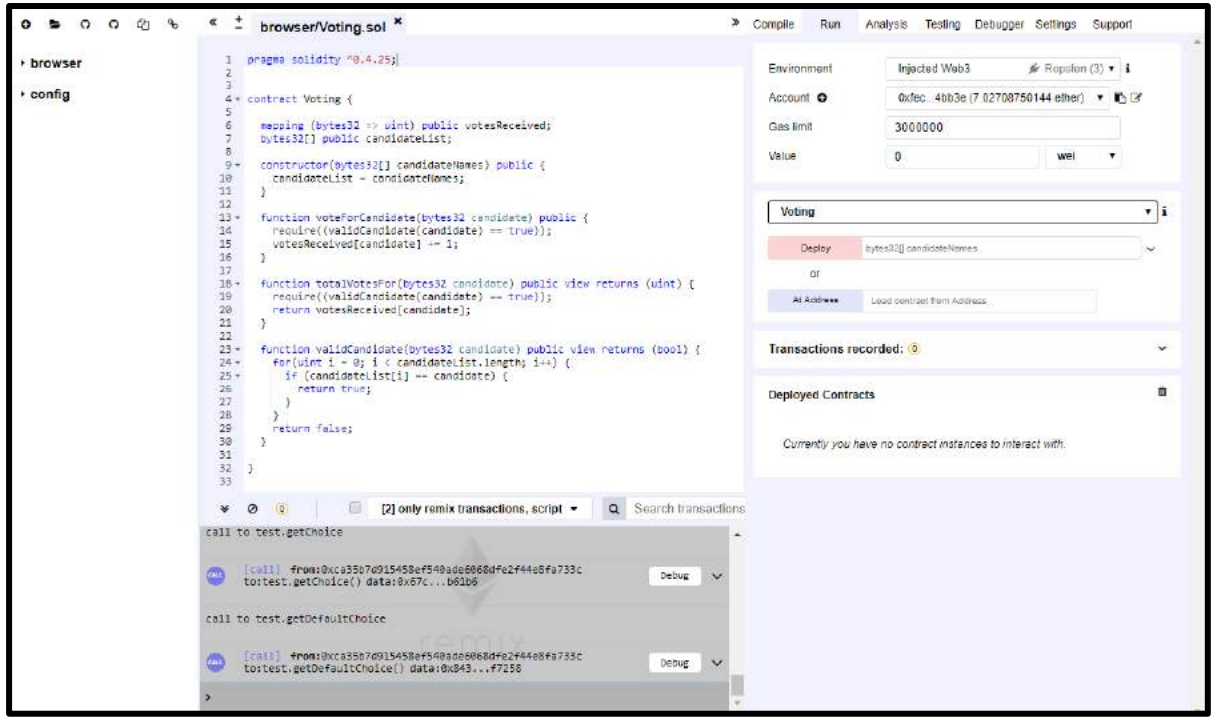
  - Call functions (Getters)
    - function totalVotesFor(bytes32 candidate) public view returns (uint) {

```
        require((validCandidate(candidate) == true));
        return votesReceived[candidate];
    }
```

Remix-IDE

- Integrated development environment that allows developers to:
  - Compile
  - Deploy
  - Interact (transact and call) with solidity smart contracts

- https://remix.ethereum.org/



References [1/2]

- [1] https://karl.tech/intro-guide-to-ethereum-testnets/
- [2] https://medium.com/bitfwd/get-ropsten-ethereum-the-easy-way-f2d6ece21763
- [3] https://medium.com/@attores/step-by-step-guide-getting-started-with-ethereum-mist-wallet-772a3cc99af4
- [4] http://ethdocs.org/en/latest/network/connecting-to-the-network.html#the-ethereum-network
- [5] http://blockchainlab.com/pdf/Ethereum_white_paper-a_next_generation_smart_contract_and_decentralized_application_platform-vitalik-buterin.pdf
- [6] https://support.ddex.io/hc/en-us/articles/115004408534-Installing-a-digital-wallet-MetaMask
- [7] https://99bitcoins.com/ethereum-wallets/

References [2/2]

- [8] https://karl.tech/intro-guide-to-ethereum-testnets/
- [9] https://kb.myetherwallet.com/gas/what-is-gas-ethereum.html
- [10] https://medium.com/crowdbotics/building-ethereum-dapps-with-meta-mask-9bd0685dfd57
- [11] https://en.wikipedia.org/wiki/Ethereum
- [12] https://medium.com/@BangBitTech/what-is-consensus-algorithm-in-blockchain-different-types-of-consensus-models-12cce443fc77
- [13] https://coinsutra.com/myetherwallet-step-step-introduction-guide-beginners/
- [14] https://etherworld.co/2017/11/17/understanding-the-concept-of-private-key-public-key-and-address-in-ethereum-blockchain/
- [15] http://ethdocs.org/en/latest/account-management.html

Basic concepts [2/4]

- What is Consensus algorithm?
  - Consensus algorithm is a process used in the Blockchain network so that all nodes to agree on a single state of the network
  - Example consensus algorithms used by Ethereum are [12]
    - *Proof-of-work (PoW).* The miners (i.e. the nodes of the network that compete to assemble new blocks) have to solve mathematically complex puzzles on the new block before approving the block to the ledger. After solving the puzzle, the solution is then forwarded to other miners and verified by them before being accepted to their respective copies of the ledger.
    - *Proof-of-Stake (PoS).* The creator of a new block is chosen in a deterministic way, depending on its wealth, also defined as stake. This algorithm is more energy efficient than PoW.

Decentralized Training Series Workshop:
Certified Ethereum Specialist

19-20 November 2018, Athens, Greece

Thank you !!! / Questions?

UNIC | Institute For the Future

CERTH CENTRE FOR RESEARCH & TECHNOLOGY HELLAS

iti Information Technologies Institute