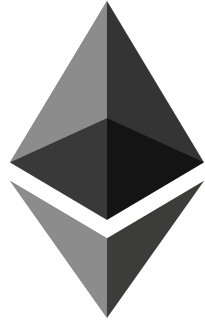


BLOCKCHAINBH



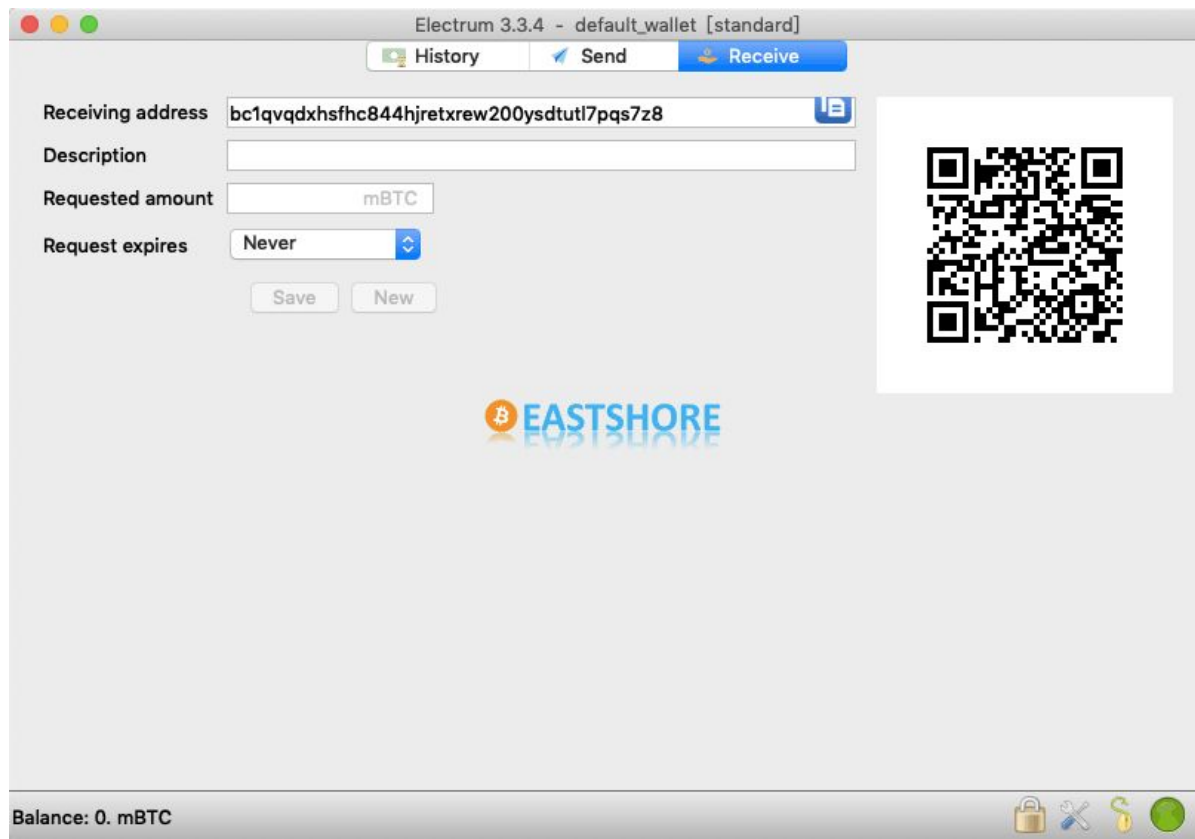
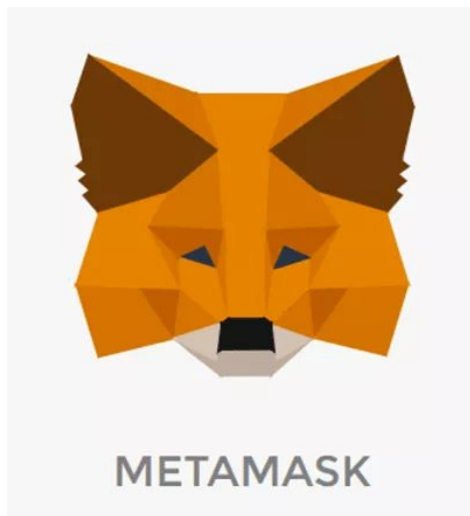
Segurança
em aplicações
Blockchain

CRIPTOMOEDAS



APLICAÇÕES BLOCKCHAIN

```
/* @dev Based on code by FirstBlood: https://github.com/Firstbloodio/token/blob/master/smart\_contract/FirstBloodToken.sol */  
contract StandardToken is ERC20, BasicToken {  
    mapping (address => mapping (address => uint256)) internal allowed;  
  
    /**  
     * @dev Transfer tokens from one address to another  
     * @param _from address The address which you want to send tokens from  
     * @param _to address The address which you want to transfer to  
     * @param _value uint256 the amount of tokens to be transferred  
     */  
    function transferFrom(address _from, address _to, uint256 _value) public returns (bool) {  
        require(_to != address(0));  
        require(_value <= balances[_from]);  
        require(_value <= allowed[_from][msg.sender]);  
        balances[_from] = balances[_from].sub(_value);  
        balances[_to] = balances[_to].add(_value);  
        balances[_from][msg.sender] = allowed[_from][msg.sender].sub(_value);  
        balances[_to][msg.sender] = allowed[_to][msg.sender].add(_value);  
    }  
}
```



PAPER WALLET



HARDWARE WALLET



"NOT YOUR KEYS, NOT YOUR BITCOIN"



RECOMENDAÇÕES

Escolher **muito bem** a carteira a usar

Pequenas quantidades para uso diário

Backup da carteira:

- Recuperação
- Mnemonic Seed
- Distribuir Backups
- Fazer Backups Regularmente



Your wallet generation seed is:

body decision painful space bloom sunlight grown vein son
third mirror dance



Please write down or memorize these 12 words (order is important). This seed will allow you to recover your wallet in case of computer failure. Your seed is also displayed as QR code, in case you want to transfer it to a mobile phone.

WARNING: Never disclose your seed. Never type it on a website.

Cancel

Next

RECOMENDAÇÕES

Criptografar sua carteira

EXCHANGE NÃO É CARTEIRA: Mtgox, Coingather, QuadrigaCX, Bitgrail...

Manter seu software atualizado

Usar multi-assinaturas

Autenticação de 2 fatores (2FA): Google Authenticator, Authy

Plano de backup para familiares/amigos

ATAQUES PASSADOS

The DAO: 3.6 milhões de Ether roubados (na época = **US\$ 70,000,000**)

Parity: bloqueio de **513,774.16 Ether** em carteiras multi assinatura

Parity: roubo de **150,000 Ether**, explorando vulnerabilidade do código

Diversas exchanges hackeadas e fundos roubados

ATAQUES COMUNS

Race Conditions (e.g. **Reentrancy**)

Integer Overflow and Underflow

Transaction-Ordering Dependence (TOD) / Front Running

Timestamp Dependence

DoS with (Unexpected) revert

DoS with Block Gas Limit

Forcibly Sending Ether to a Contract

Deprecated/historical attacks

fallback function

```
function () {  
    ...  
}
```

function with no name



- it's executed only when someone executes a function not existed in the contract or someone simply sends some ether to this contract
- not necessary
- pitfall:
 - person who owns this contract gets to decide what to do in a fallback function
 - if you simply sends some ether to this contract, you also have to pay for the execution of fallback function if there is one

BEST PRACTICES

[Ethereum Smart Contract Security Best Practices](#)

Tipos de dados apropriados

Checar se tem fundo suficiente, **subtrair**, transferir

Visibilidade apropriada para variáveis e funções

<code>public</code>	Acesso irrestrito
<code>private</code>	Contrato corrente
<code>external</code>	Outros contratos. Não pode ser chamada internamente
<code>internal</code>	Contrato corrente e contros filhos

BEST PRACTICES

Modificadores para implementar regras

Eventos para notificações / log de milestones - e.g. **withdrawFunds()**

Versão fixa do compilador Solidity. Evitar `pragma solidity ^0.5.0;`

Checar por overflows / underflows

Frameworks e bibliotecas bem testadas e auditadas

Testar! (e.g. testes unitários no Truffle framework)

DESIGN PATTERNS

[Solidity Common Patterns](#) , [Solidity Patterns](#) , [Smart Contracts Patterns](#)

Behavioral patterns

Security patterns

Upgradeability patterns

Economic patterns

FRAMEWORKS E BIBLIOTECAS

Implementações **bem testadas** e **reusáveis**

Exs: *OpenZeppelin*, *ZeppelinOS*

- . SafeMath
- . Ownable
- . Pausable
- . RBAC (Role Based Access Control)
- . ERC20

SECURITY / AUDIT TOOLS

[Ferramentas de auditoria de Smart Contracts](#)

Mythril

Solhydra

Slither

OBRIGADO!

BLOCKCHAINBH



educação e soluções em blockchain

