ECE 443/518 Fall 2022 - Graduate Project 2

Smart Contract Challenge

For example, if your CWID is A12345678, you will call 'register' function only take an interger as the parameter so you don't call it with a string 'A12345678'). The code to open the treasure box for A12345678 is 77472471525 so you will call

• You may need to use part of your program from Project 1. Depending how efficient your program is and whether it can utilize all computational power available on modern multi-threading and multi-processing, it may take from less than a minutes to over an hour

• While not absolutely necessary, testing the smart contract in an Ethereum simulator will be very helpful for you to understand how it works. The most convenient Ethereum simulator is Remix that runs directly inside your browser. I have included a short tutorial on how to use it at the end

Resources

Documentation

Report Due: 12/07 (Wed.), by the end of the day (Chicago time) Late submissions will NOT be graded

I. Objective A virual treasure box is created as a smart contract on the Ethereum blockchain. Open it to claim the prize utilizing the knowledge you have learned from this course! We don't actually require you to deploy smart contracts and to interact with them on the actual Ethereum blockchain as that could cost you real money and is slow for development and testing. As a common practice discussed later, an Ethereum simulator should be used instead.

II. The TreasureBox Smart Contract Here is the smart contract TreasureBox.sol. There are two functions 'register' and 'unlock'. You will need to first call the function 'register' to register your CWID with the treasure box and then call the function 'unlock' to open the box with a code. Once the box is opened, you can call 'register'

again, which will generate a message showing you have already unlocked the box.

The smart contract allows you to call the two functions in any order for any number of times, though it will generate error messages if you don't follow the above protocol. III. Finding the Code for Your CWID

• Read the source code of the smart contract. It tells you what condition the code should satisfy to open the box for a particular CWID.

• Formulate the condition as an equation and write a program to solve it. Similar to proof-of-work, a brute-force algorithm should be used.

'unlock(77472471525)' to open it. Call 'register(12345678)' again to verify the box is open.

Remix - Ethereum IDE

CO

of this instruction.

Submit the following to Blackboard for this project.

1. The code to open the treasure box for your CWID.

IV. Deliverables

Since your CWID is not A12345678, the code to open your box is not 77472471525. What is your code? Clearly you should not modify the smart contract source code to use any code you would like. You will need to find the code for your CWID for this project. Here are the hints.

2. A 1-page project report, describing your approach to find your code. The project should be done individually. You can discuss the project with other students but all the source code and writings should be your OWN. PLAGIARISM and called for DISCIPLINARY ACTION. NEVER share your source code and reports with others. V. Remix Tutorial (Optional) Use your browser to open Remix. By default it should open "File Explorers" on the left. Use the upload button to upload the smart contract source file.

 O
Home × FILE EXPLORERS Quicklinks Workspaces 🛨 🗹 📋 default_workspace Migration tools: → 🗅 🗀 😯 🕹 🖛 Basic migration contracts artifacts Restore files from backup zip

\$ 1_Storage.sol Help: \$ 2_Owner.sol <u>Gitter channel</u> 5 TreasureBox.sol Report on Github scripts tests README.txt Featured Plugins SOLIDITY OPTIMISM LEARNETH SOLHINT LINTER

> III Gitter channel Copen Files Q Search with transaction hash or address listen on network You can use this terminal to: . Check transactions details and start debugging. • Execute JavaScript scripts: - Select a Javascript file in the file explorer and then run \`remix.execute()\` or \`remix.exeCurrent()\` in the command line interface - Right click on a JavaScript file in the file explorer and then click \`Run\ The following libraries are accessible: web3 version 1.5.2 • ethers.js remix (run remix.help() for more info)

> > mapping(address => uint32) public registered;

registered[msg.sender] = id;

"You have already unlocked the box.");

Search with transaction hash or address

require(!unlocked[msg.sender],

listen on network

address public owner;

owner = msg.sender;

constructor() {

else {

mapping(address => uint32) public registered;

event Log(uint32 id, uint64 code, bytes text, bytes32 hash);

"You can only register one id per account.");

Q Search with transaction hash or address

mapping(address => bool) public unlocked;

function register(uint32 id) public { require(id != 0, "Invalid id.");

if (registered[msg.sender] != 0) {

registered[msg.sender] = id;

require(!unlocked[msg.sender],

listen on network

Q Q D Home 💲 TreasureBox.sol 💥

5 r contract TreasureBox {

constructor() {

* O 0

// SPDX-License-Identifier: GPL-3.0

owner = msg.sender;

The following libraries are accessible:

• remix (run remix.help() for more info)

transact to TreasureBox.register pending ...

creation of TreasureBox pending...

Q Q D Home TreasureBox.sol 🗶

5 r contract TreasureBox {

decoded output

logs

If you call "unlock" with the correct code "77472471525", the log message will show "unlock" returning true. Our smart contract also generates additional log messages to help you understand it.

* O 0

input

val

decoded input

Q Q D Home S TreasureBox.sol 🗶

5 v contract TreasureBox {

1 // SPDX-License-Identifier: GPL-3.0

pragma solidity >=0.7.0 <0.9.0;

// SPDX-License-Identifier: GPL-3.0

listen on network

pragma solidity >=0.7.0 <0.9.0;

• web3 version 1.5.2

• ethers.js

mapping(address => uint32) public registered;

event Log(uint32 id, uint64 code, bytes text, bytes32 hash);

Q Search with transaction hash or address

[vm] from: 0x5B3...eddC4 to: TreasureBox.(constructor) value: 0 wei data: 0x608...70033 logs: 0 hash: 0x68d...105f1

Right click on a JavaScript file in the file explorer and then click \ Kun\

https://remix.ethereum.org/#optimize=false&runs=200&evmVersion=null&version=soljson-v0.8.7+commit.e28d00a7.js

Q Search with transaction hash or address

800000000 gas 🕻

29127 gas 🗓

29127 gas 🚨

0xef4...0006f 🗓

) ()

) ()

https://remix.ethereum.org/#optimize=false&runs=200&evmVersion=null&version=soljson-v0.8.7+commit.e28d00a7.js

0x5c5ffc056cba8bdafc6b4646faf88455487d98c09f50886f9ea13d007d294499

TreasureBox.unlock(uint64) 0xd9145CCE52D386f254917e481eB44e9943F39138

0x5c5ffc056cba8bdafc6b4646faf88455487d98c09f50886f9ea13d007d294499

0x5B38Da6a701c568545dCfcB03FcB875f56beddC4 (

" isBigNumber": true

Search with transaction hash or address

"_isBigNumber": true

"1": "77472471525",

"id": 12345678,

"2": "0x00bc614e0000001209b819e5",

"3": "0x00bc614edbd85f99e4d197914fde2b170abf072123eeb5fee24d9fce51a69d24",

0xef4...819e5 🗘

O wei 🕻

mapping(address => bool) public unlocked;

function register(uint32 id) public { require(id != 0, "Invalid id.");

pragma solidity >=0.7.0 <0.9.0;

address public owner;

require(registered[msg.sender] == id,

"You have already unlocked the box.");

function unlock(uint64 code) public returns (bool) {

mapping(address => bool) public unlocked;

File

New File

Open "TreasureBox.sol" from "File Explorers". Move to "Solidity Compiler" on the left. Click "Compile TreasureBox.sol" and then the contract TreasureBox should appear. Remix - Ethereum IDE Ch https://remix.ethereum.org/#optimize=false&runs=200&evmVersion=null&version=soljson-v0.8.7+commit.e28d00a7.js Home TreasureBox.sol **SOLIDITY COMPILER** // SPDX-License-Identifier: GPL-3.0 pragma solidity >=0.7.0 <0.9.0; COMPILER ± 5 v contract TreasureBox { 0.8.7+commit.e28d00a7 address public owner; Include nightly builds

Solidity 12 + constructor() { owner = msg.sender; EVM VERSION event Log(uint32 id, uint64 code, bytes text, bytes32 hash); compiler default function register(uint32 id) public { COMPILER CONFIGURATION require(id != 0, "Invalid id."); Auto compile if (registered[msg.sender] != 0) { require(registered[msg.sender] == id, "You can only register one id per account."); Enable optimization

else {

Hide warnings Compile TreasureBox.sol function unlock(uint64 code) public returns (bool) { CONTRACT TreasureBox (TreasureBox.sol) * O 0 Publish on lpfs 📠 · Check transactions details and start debugging. • Execute JavaScript scripts: - Input a script directly in the command line interface Compilation Details - Right click on a JavaScript file in the file explorer and then click \`Run\ ⚠ ABI Bytecode The following libraries are accessible: • web3 version 1.5.2 • ethers.js • remix (run remix.help() for more info)

LANGUAGE

Move to "Deploy & Run Transactions" on the left. Make sure the environment is "Remix VM" or "Javascript VM", which creates an Ethereum simulator in your browser. Click "Deploy" and the the contract TreasureBox is deployed to the Ethereum simulator. Remix - Ethereum IDE Ch https://remix.ethereum.org/#optimize=false&runs=200&evmVersion=null&version=soljson-v0.8.7+commit.e28d00a7.js Q Q D Home 5 TreasureBox.sol 🗶 DEPLOY & RUN TRANSACTIONS // SPDX-License-Identifier: GPL-3.0 ENVIRONMENT pragma solidity >=0.7.0 <0.9.0;

JavaScript VM (London) 5 r contract TreasureBox { ACCOUNT 3 0x5B3...eddC4 (99.999999! \$ 1 2 GAS LIMIT 3000000 VALUE 21 •

CONTRACT TreasureBox - contracts/TreasureBox.: 25 + Publish to IPFS OR At Address Transactions recorded 1 • ethers.js • remix (run remix.help() for more info) Deployed Contracts > TREASUREBOX AT 0XD91...39138 (MEN (L) × creation of TreasureBox pending...

* Expand the deployed TreasureBox smart contract and you will see the two functions "register" and "unlock". Put "12345678" as the parameter to "register" and click it to call it. A new log messge will appear to confirm that the transaction is completes correctly. Remix - Ethereum IDE Ch https://remix.ethereum.org/#optimize=false&runs=200&evmVersion=null&version=soljson-v0.8.7+commit.e28d00a7.js DEPLOY & RUN TRANSACTIONS ■ CONTRACT TreasureBox - contracts/TreasureBc Deploy **>>** Publish to IPFS OR At Address Load contract from Address Transactions recorded 2

Deployed Contracts ➤ TREASUREBOX AT 0XD91...39138 (I 🖺 🗶 12345678 ~ v Low level interactions CALLDATA

Remix - Ethereum IDE

•

>>

Ch

CONTRACT

CALLDATA

Remix - Ethereum IDE

>>

Ch

CONTRACT

Publish to IPFS

Transactions recorded 4

unlocked

Low level interactions

Transactions recorded 5

12345678

77472471525

CALLDATA

*

DEPLOY & RUN TRANSACTIONS

TreasureBox - contracts/TreasureBc

OR

At Address Load contract from Address

Deploy

Publish to IPFS

DEPLOY & RUN TRANSACTIONS

TreasureBox - contracts/TreasureBc

OR

Load contract from Address status Transactions recorded 3 transaction hash **Deployed Contracts** ➤ TREASUREBOX AT 0XD91...39138 (I 🖺 🗶 12345678 transaction cost 111 hash input decoded input Low level interactions

10 Deployed Contracts decoded output "0": "bool: true" ➤ TREASUREBOX AT 0XD91...39138 (I 🖺 🗶) (D logs 12345678 "topic": "0x804d04e64cd8811d697278cae0e00a2a11737b259e6f0bc3f7714ce6c1c2c389", "event": "Log", 77472471525 "args": owner "3": "0x00bc614edbd85f99e4d197914fde2b170abf072123eeb5fee24d9fce51a69d24", "id": 12345678, registered

Remix - Ethereum IDE Ch https://remix.ethereum.org/#optimize=false&runs=200&evmVersion=null&version=soljson-v0.8.7+commit.e28d00a7.js Q Q ♠ Home 🕻 TreasureBox.sol 💥 DEPLOY & RUN TRANSACTIONS ■ 1 // SPDX-License-Identifier: GPL-3.0 CONTRACT pragma solidity >=0.7.0 <0.9.0; TreasureBox - contracts/TreasureBc 5 r contract TreasureBox { Deploy Q Search with transaction hash or address listen on network **>>** Publish to IPFS "topic": "0x804d04e64cd8811d697278cae0e00a2a11737b259e6f0bc3f7714ce6c1c2c389", OR "args": { "0": 12345678, At Address Load contract from Address

Now if you call "register" again with "12345678", you will get the message "You have already unlocked the box." showing that you have successfully open the box.

"code": "77472471525", **Deployed Contracts** "hash": "0x00bc614edbd85f99e4d197914fde2b170abf072123eeb5fee24d9fce51a69d24" ➤ TREASUREBOX AT 0XD91...39138 (I 🖺 🗶] O O 12345678 O wei 🕻 val 77472471525 transact to TreasureBox.register pending ... ~ v

[vm] from: 0x5B3...eddC4 to: TreasureBox.register(uint32) 0xd91...39138 value: 0 wei data: 0x130...c614e logs: 0 hash: 0xa62...f0637 Low level interactions transact to TreasureBox.register errored: VM error: revert. CALLDATA The transaction has been reverted to the initial state. Reason provided by the contract: "You have already unlocked the box.". Debug the transaction to get more information. The TreasureBox smart contract allows one Ethereum account to register and unlock for a single CWID only. You should change to a different account to test with your own CWID. There is no need to deploy the smart contract again. Remix - Ethereum IDE

Ch https://remix.ethereum.org/#optimize=false&runs=200&evmVersion=null&version=soljson-v0.8.7+commit.e28d00a7.js Q Q ♠ Home 5 TreasureBox.sol 💥 DEPLOY & RUN TRANSACTIONS // SPDX-License-Identifier: GPL-3.0 ENVIRONMENT pragma solidity >=0.7.0 <0.9.0; JavaScript VM (London) 5 r contract TreasureBox {

revert

ACCOUNT 🙃 * Ø 0 Q Search with transaction hash or address listen on network 0xAb8...35cb2 (100 ethe 💠 🕒 📝 "topic": "0x804d04e64cd8811d697278cae0e00a2a11737b259e6f0bc3f7714ce6c1c2c389", "event": "Log", GAS LIMIT "args": { 3000000 "1": "77472471525", "2": "0x00bc614e0000001209b819e5", VALUE "3": "0x00bc614edbd85f99e4d197914fde2b170abf072123eeb5fee24d9fce51a69d24", "id": 12345678, 0 "code": "77472471525", "hash": "0x00bc614edbd85f99e4d197914fde2b170abf072123eeb5fee24d9fce51a69d24" CONTRACT TreasureBox - contracts/TreasureBc 1 0 0 O wei 🕻 Deploy Publish to IPFS OR Load contract from Address transact to TreasureBox.register pending ... Transactions recorded 5 Deployed Contracts [vm] from: 0x5B3...eddC4 to: TreasureBox.register(uint32) 0xd91...39138 value: 0 wei data: 0x130...c614e logs: 0 hash: 0xa62...f0637 ➤ TREASUREBOX AT 0XD91...39138 (I 🖺 🗙

transact to TreasureBox.register errored: VM error: revert.

Debug the transaction to get more information.

The transaction has been reverted to the initial state. Reason provided by the contract: "You have already unlocked the box.".

Ð https://remix.ethereum.org/#optimize=false&runs=200&evmVersion=null&version=soljson-v0.8.7+commit.e28d00a7.js €3 Guide for migrating the old File System Download all Files as a backup zip

(1)

SOURCIFY

••• MORE Ð 2

- Select a Javascript file in the file explorer and then run \`remix.execute()\` or \`remix.exeCurrent()\` in the command line interface Ð X

[vm] from: 0x5B3...eddC4 to: TreasureBox.(constructor) value: 0 wei data: 0x608...70033 logs: 0 hash: 0x68d...105f1 Debug 💙

Debug 💙

[vm] from: 0x5B3...eddC4 to: TreasureBox.register(uint32) 0xd91...39138 value: 0 wei data: 0x130...c614e logs: 0 hash: 0x8c9...167b4 You can then call "unlock" with a code. Using a wrong code like "111" will not cause the transaction to fail but if you check the details of the log message you will see "unlock" returns false, indicating the code is wrong. [vm] from: 0x5B3...eddC4 to: TreasureBox.unlock(uint64) 0xd91...39138 value: 0 wei data: 0xef4...0006f logs: 0 hash: 0x5c5...94499 Debug 🔥

> "topic": "0x804d04e64cd8811d697278cae0e00a2a11737b259e6f0bc3f7714ce6c1c2c389", "hash": "0x00bc614edbd85f99e4d197914fde2b170abf072123eeb5fee24d9fce51a69d24"

Ð