Sensitive On Chain Data: Exercise 1 - Mastering CAST

Intro

In this exercise you will learn how to work with `cast` tool.

Installations & Guidelines

Download and install Foundry

```
curl -L https://foundry.paradigm.xyz | bash
foundryup
```

```
For the next tasks, use this Binance RPC url:
https://rpc.ankr.com/bsc

Important: Use ONLY "cast" CLI to solve the following exercises, in your
answer provide the command that you used.
```

Useful Links

Foundy Installation Guide

Cast Overview

Cast Commands

Tasks

Task 1 - Retreiving General Information

1. Retreive the chainID.

2. Retreive the last validated block number.

Task 2 - Retreiving Transaction and Block Information

1. Get the transaction info for this tx hash
   `0x3f6da406747a55797a7f84173cbb243f4fd929d57326fdcfcf8d7ca55b75fe99`.

2. Get the block timestamp and the miner address who validated the block of the transaction from the previous question.

3. For the same transaction, get the transaction input data and contract address that was called.

Task 3 - Transaction Analysis

1. Using the data that you got from the previous question, find the function name and parameters types that was called.

2. Decode the input data.

Task 4 - Smart Contract Storage Analysis

1. Get the previous task's smart contract's bytecodes.

2. Get the contract storage slots 0, 1, and 2.

3. Get the smart contract source-code FROM HERE, and copy it to a new file in VSCODE.

Explain which are the state variables that match storage values that you found in the previous question?