CHAINTRACE

BY CARTER, KURSAT,ANNA AND LEO

CHAINTRACE: A NEW FRONTIER IN BLOCKCHAIN SECURITY

TRACING THE UNTRACEABLE IN THE CRYPTO ECOSYSTEM

What is...
...a Blockchain?

1

A Digital Ledger

A Blockchain is a digital ledger which keeps records of all transactions taking place on a peer to peer network.

2

Encrypted Information

All Information transferred via blockchain is encrypted and every occurrence recorded, meaning once the block is created and added to the chain, it cannot be altered.

3

Peer to Peer

Lets you interact or send transactions with a peer, without an intermediary. Removes the middle man.

4

Data Sharing

The blockchain can be used for more than the transfer of currency. It can also be used to share contracts, records and any other type of data.

5

Decentralization

The blockchain is decentralized, so there isn't a need for a central, certifying authority.

BLOCKCHAIN.WTF

"OVER A MILLION DOLLARS WAS STOLEN FROM MY FATHER"

- CARTER

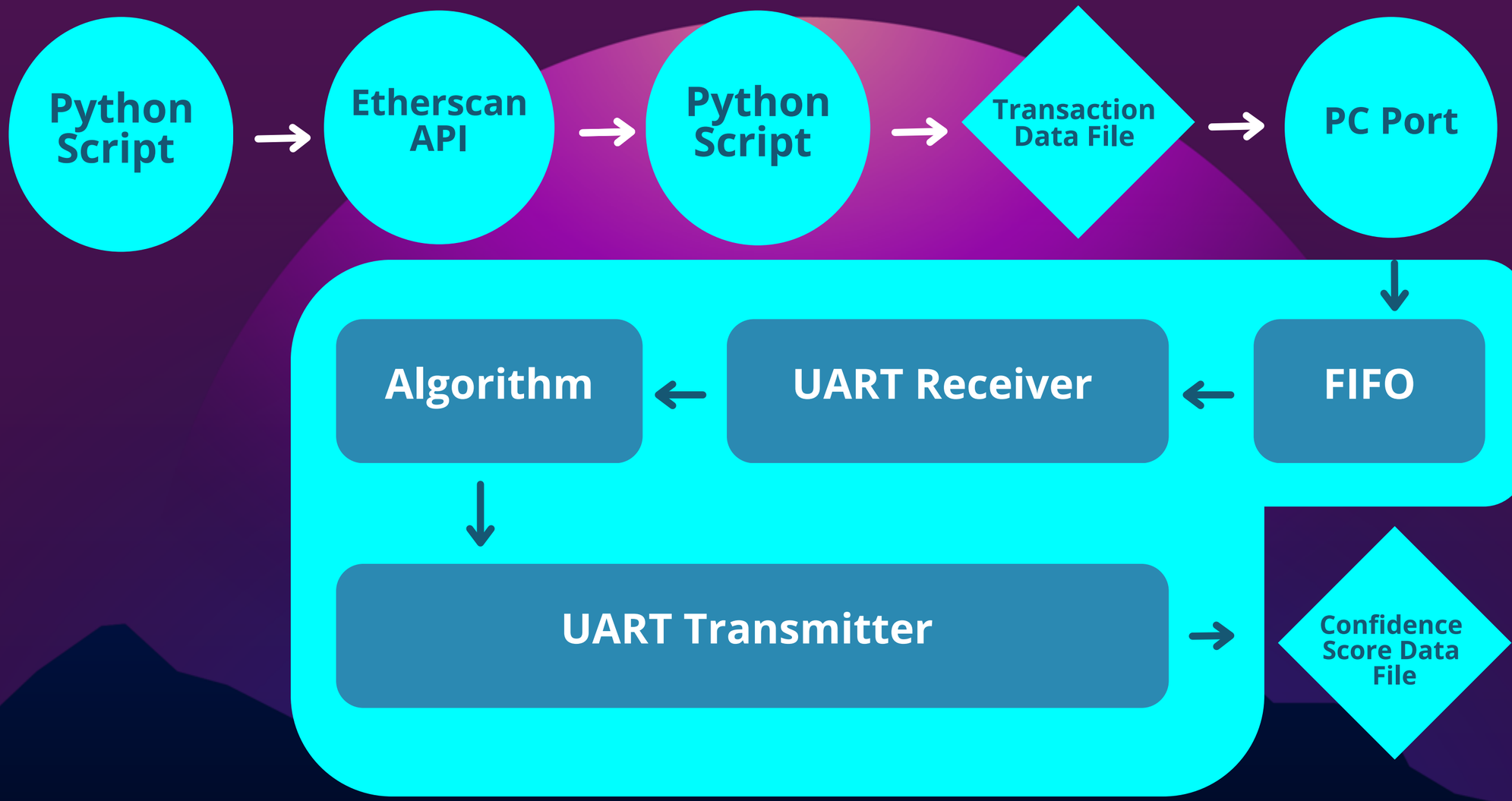
INTRODUCING CHAINTRACE:

TACKLING REAL-WORLD CRYPTO SCAMS

UNIQUE ANGLE

REAL-TIME LIGHTENING FAST
PARALLEL CRYPTO ANALYSIS

Block Diagram



UART CONNECTION FIFO

- Purpose: Stores received data temporarily, acting as a buffer between the UART receiver and the processing unit.
- Capacity Management: FIFO structure to manage data for sequential processing.
- Parameterized Design: Customizable data and address size.
- Flow Control: Incorporates signals to indicate when the FIFO is full or empty, aiding in effective flow management.

UART CONNECTION RECIEVER

- Functionality: Converts serial data received from an external source into parallel form for internal processing.
- Baud Rate Synchronization: Aligns with the transmitter's baud rate
- Start and Stop Bits: Recognizes start and stop bits to determine the boundaries of each data packet.
- Data Sampling: Employs oversampling.
- Integration with FIFO: Seamlessly interfaces with larger_fifo to transfer received data for temporary storage and subsequent processing.

CONFIDENCE SCORE ALGORITHM PART 1

Calculating confidence score based on 4 factors:

- Transfer Method 15%
  - Tether/ Monero, other methods (Token Transfer)
    - Calculating the running ratio by taking the amount of token transfer transactions dividing by the total of transactions
- Value 20%
  - Calculating the moving average value, total value of all transactions of wallet dividing by the total amount of transactions
    - Calculate confidence amount based on 5 degrees of value thresholds

CONFIDENCE SCORE ALGORITHM PART 2

Calculating confidence score based on 4 factors:

- In/Out Ratio 35%
  - Track the total number of transactions going in
  - Divide by the total number to calculate the ratio
- Period 30%
  - Track the first time stamp and last
  - Subtract and then divide by the total number of transactions to calculate the period (sec/req)

CONFIDENCE SCORE ALGORITHM PART 3

Final Confidence Score Calculation for a single wallet

- Looking at all transactions in that wallet
- 4 registers assigned for each factor (m, v, i, p)
  - m = method
  - v = value
  - i = in/out ratio
  - p = period
- Sum of all confidence score we get in each factor

CODE SNIPPET

METHOD FACTOR

```
//method
always @(posedge clk) begin
    case (method_field)
    tether: begin
        method_indicator <= 1;
    end
    monero: begin
        method_indicator <= 1;
    end
    other_method: begin
        moving_method <= moving_method+1;
    end
end
```

```
if(method_indicator == 1 || 100*(moving_method/total_running_sum) >= 15){
    m <= 15;
}else if (100*(moving_method/total_running_sum) >= 10){
    m <= 10;
}else if (100*(moving_method/total_running_sum) >= 5){
    m <= 5;
}else{
    m<=0;
}
```

VALUE FACTOR

```
if(moving_avg_value/total_running_sum >= value_degree5){
    v <= 20;
}else if (moving_avg_value/total_running_sum >= value_degree4){
    v <= 17;
}else if (moving_avg_value/total_running_sum >= value_degree3){
    v <= 14;
}else if (moving_avg_value/total_running_sum >= value_degree2){
    v <= 10;
}else if (moving_avg_value/total_running_sum >= value_degree1){
    v <= 7;
}else{
    v <=0;
}
```

IN/OUT FACTOR

```
if(100*(moving_in/total_running_sum) >= 95 || 100*(moving_in/total_running_sum) <= 5 ){
    i <= 35;
}else if (100*(moving_in/total_running_sum) >= 90 || 100*(moving_in/total_running_sum) <= 10){
    i <= 30;
}else if (100*(moving_in/total_running_sum) >= 85|| 100*(moving_in/total_running_sum) <= 15 ){
    i <= 25;
}else if (100*(moving_in/total_running_sum) >= 80 || 100*(moving_in/total_running_sum) <= 20 ){
    i <= 20;
}else if (100*(moving_in/total_running_sum) >= 75 || 100*(moving_in/total_running_sum) <= 25 ){
    i <= 15;
}else if (100*(moving_in/total_running_sum) >= 70 || 100*(moving_in/total_running_sum) <= 30){
    i <= 10;
}else{
    i<=0;
}
```

SUCCESSES

- Converting EtherScan JSON file to a Binfile that can be easily computed in Verilog
    - Since the Binfile is in a way not readable by humans we were able to comprehend it by looking at the ASCII chart
        - Character set -> Hex -> Binary
- Establishing UART connection between PC and FPGA
- Sending large amounts of data to the FPGA with a python script and being able to process that data to send it into our algorithm
- Fully implementing a Verilog Algorithm that determines the confidence score of given wallet

FAILURES

- The initial intention was to have a full-functioning tracing program with intricate depths, following an illicit transaction through multiple wallets (multiple depths) and getting suspicious endpoints with a confidence score.
- Providing a a confidence score for a single suspicious wallet by looking at all the transactions in that wallet.

CHALLENGES

- Implementing an unfamiliar field without much prior knowledge about
  - Cryptocurrency, Blockchain, Transaction tracing etc..
  - Creating the criteria for suspicious activity (confidence score)
- Understanding how to interpret the Binfile and utilize it in Verilog
- Performing mathematical operations in Verilog with a larger bitwidth

NEXT STEPS

- We have all the components of the Block Diagram, need to work on the connection portion in the next days

OUR VISION FOR THE FUTURE

THANK YOU

HTTPS://GITHUB.COM/MRRCARTER/CHAINTRACE