Zilliqa Schnorr Test

***] Code:**

```
--------------------------------
scilla_version 0

contract Schnorr()

transition verify(msg : ByStr, sig : ByStr64, pubk : ByStr33)
  res = builtin schnorr_verify pubk msg sig;
  e = {_eventname: "Signature Verification Result"; result : res };
  event e
end
--------------------------------
```

(* The code will simple emit event saying True or False for signature verification result. *)

***] Testnet contract address: (Load this in Savant IDE)**

```
--------------------------------
0x2c5d0bde39f7c4dcac6785ebeb9aa18d0dda554c
--------------------------------
```

***] Follow the steps to generate signature and get public key for message using zilpay wallet:**

1. On any webpage(can use ide page also) loaded from internet and open browser console.
2. Connect we the website using

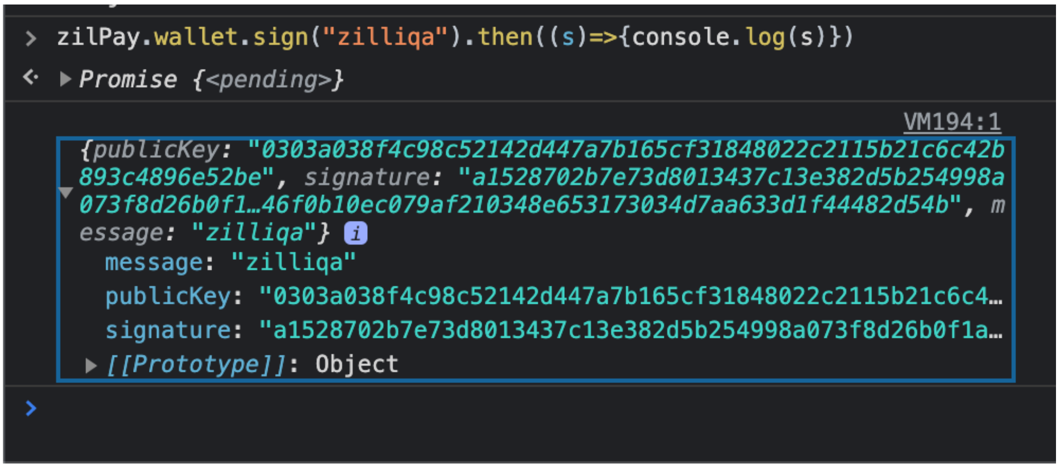
```
zilPay.wallet.connect()
```

3. Get Signature and Pub key typing following code in console

```
zilPay.wallet.sign("zilliqa").then((s)=>{console.log(s)})
```

->Wallet will ask for signature, click on **sign** also check hash by clicking toggle hash button. (Signature may be different every time you sign)

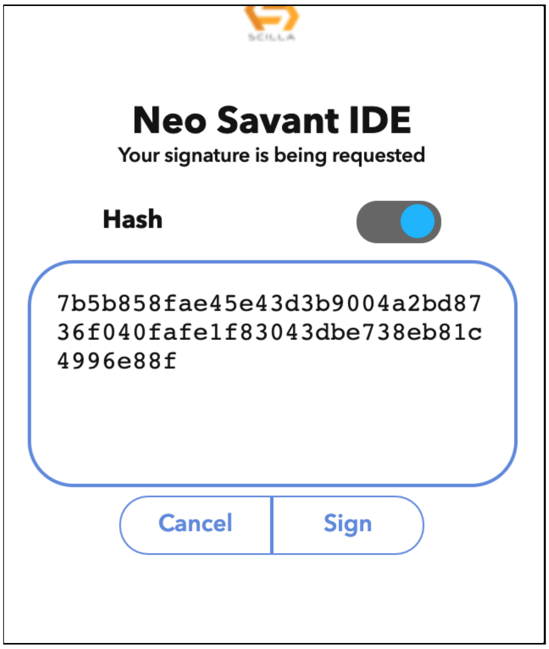
5. It will log/print javascript object with like this



```
> zilPay.wallet.sign("zilliqa").then((s)=>{console.log(s)})
<· ▶ Promise {<pending>}
                                                                    VM194:1
  ▼ {publicKey: "0303a038f4c98c52142d447a7b165cf31848022c2115b21c6c42b
    893c4896e52be", signature: "a1528702b7e73d8013437c13e382d5b254998a
    073f8d26b0f1…46f0b10ec079af210348e653173034d7aa633d1f44482d54b", m
    essage: "zilliqa"} i
      message: "zilliqa"
      publicKey: "0303a038f4c98c52142d447a7b165cf31848022c2115b21c6c4…
      signature: "a1528702b7e73d8013437c13e382d5b254998a073f8d26b0f1a…
    ▶ [[Prototype]]: Object
>
```

***] Verification of signature on chain**

Take the **publicKey** and **signature shown above;** and **messageHash (not message)** which you can get by clicking hash toggle button on wallet popup as shown below



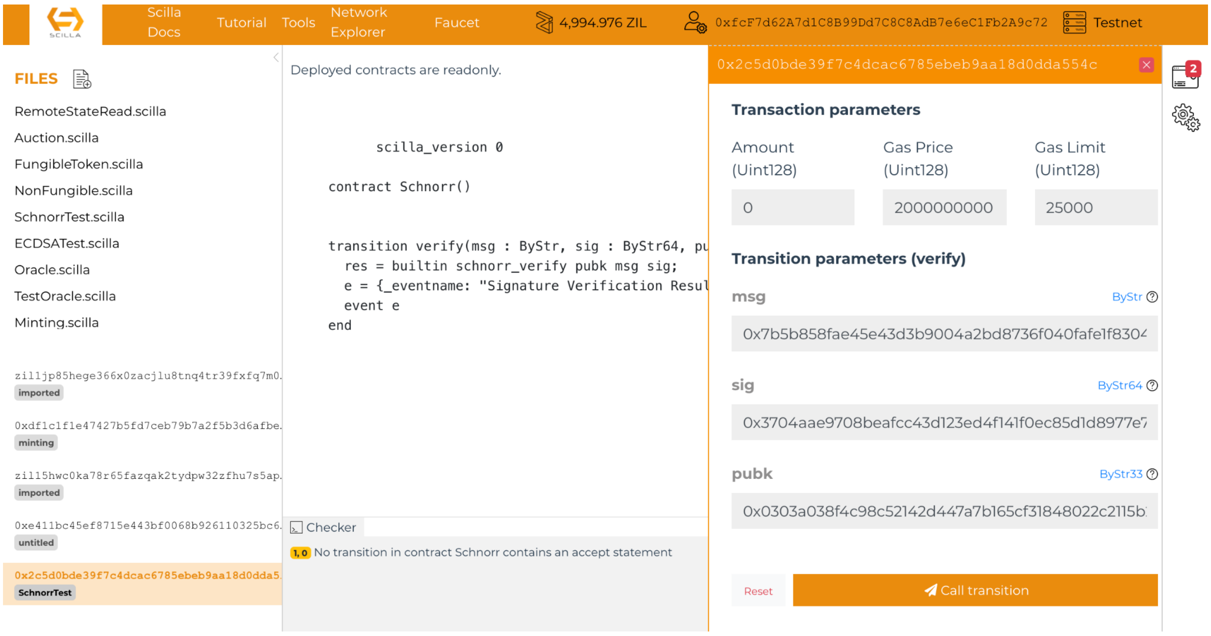
Neo Savant IDE

Your signature is being requested

Hash

```
7b5b858fae45e43d3b9004a2bd87
36f040fafe1f83043dbe738eb81c
4996e88f
```

Now in the ide load the contract above and call **verify transition** as shown below



Deployed contracts are readonly.

```
      scilla_version 0

contract Schnorr()


transition verify(msg : ByStr, sig : ByStr64, pu
  res = builtin schnorr_verify pubk msg sig;
  e = {_eventname: "Signature Verification Resul
  event e
end
```

1, 0 No transition in contract Schnorr contains an accept statement

Transaction parameters

Amount (Uint128)	Gas Price (Uint128)	Gas Limit (Uint128)
0	2000000000	25000

Transition parameters (verify)

msg ByStr
0x7b5b858fae45e43d3b9004a2bd8736f040fafe1f8304

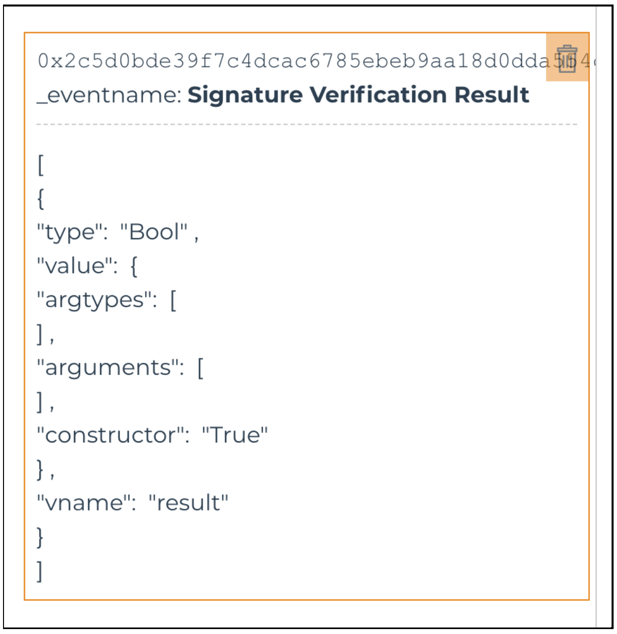
sig ByStr64
0x3704aae9708beafcc43d123ed4f141f0ec85d1d8977e7

pubk ByStr33
0x0303a038f4c98c52142d447a7b165cf31848022c2115b

Pass the value here, also don't forget to put **0x** before to specify byte data.

Now click **Call transition**

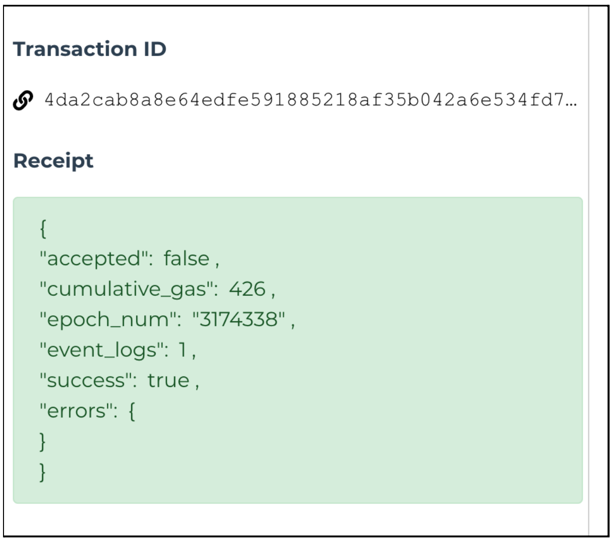
It will give you the receipt and event as shown below



0x2c5d0bde39f7c4dcac6785ebeb9aa18d0dda554

_eventname: **Signature Verification Result**

```
[
{
"type": "Bool" ,
"value": {
"argtypes": [
] ,
"arguments": [
] ,
"constructor": "True"
} ,
"vname": "result"
}
]
```



Transaction ID

4da2cab8a8e64edfe591885218af35b042a6e534fd7…

Receipt

```
{
"accepted": false ,
"cumulative_gas": 426 ,
"epoch_num": "3174338" ,
"event_logs": 1 ,
"success": true ,
"errors": {
}
}
```