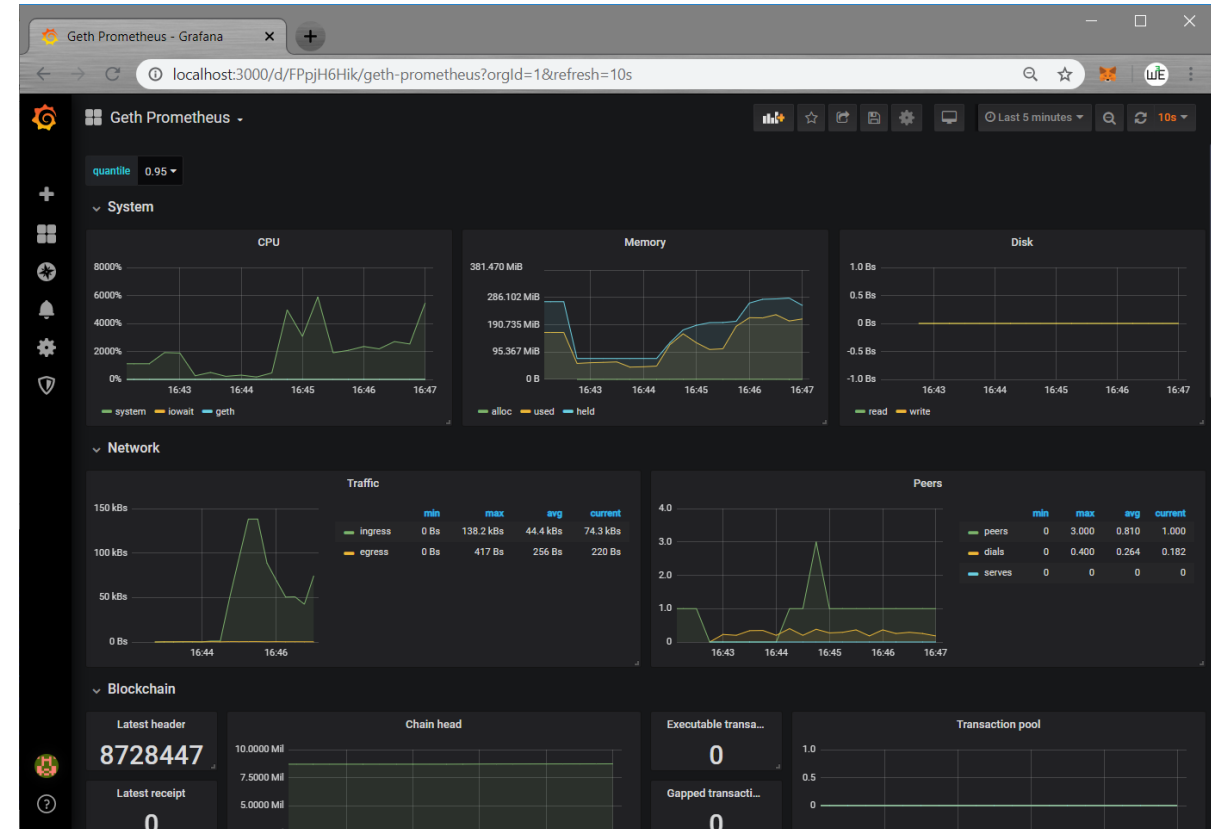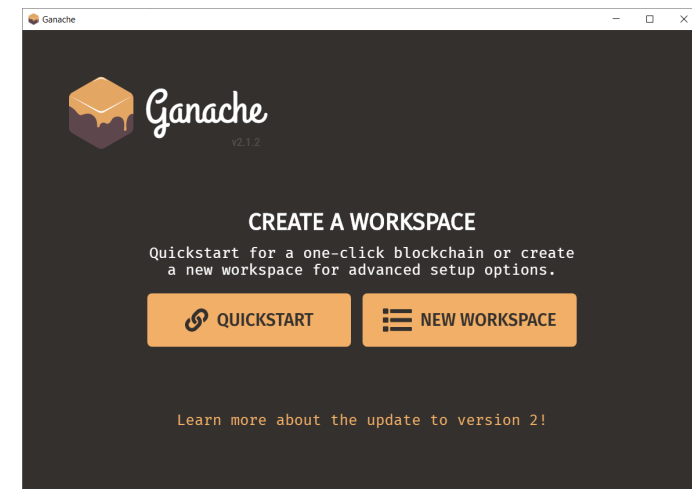# PD-4.0 Nodes & RPC interface

- PD-4.1 Run a local Node (Ganache)
- PD-4.2 Access local nodes via Metamask
- PD-4.3 Geth (via Grid)
- PD-4.4 Management of nodes
- PD-4.5 RPC functions
- PD-4.6 Tools to access RPC interface
- PD-4.7 Web interface for RPC
- PD-4.8 Test networks
- PD-4.9 Infura account

# PD-4.1.1 Install Ganache

# PD-4.1.1 Ganache – fixed seed



spend vital trip member scheme buzz liquid normal supreme soft device cereal

# PD-4.1.2 Ganache and MetaMask
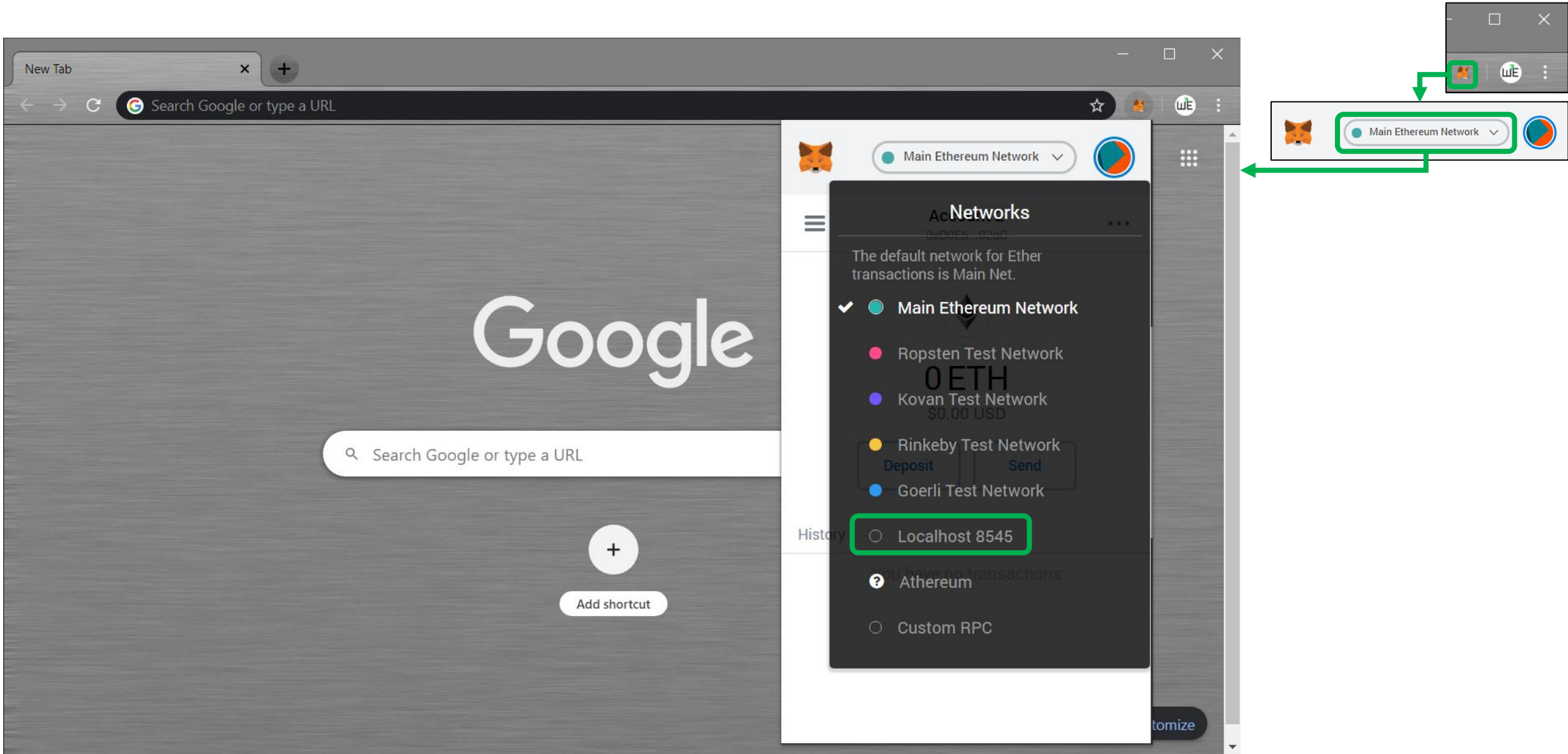
# PD-4.1.2 Ganache and MetaMask



ACCOUNT INFORMATION

ACCOUNT ADDRESS
0×6B5bB8441DD079F8Da87FF48F74F3A4F08bf417B

PRIVATE KEY
4f8801870ed9311f0f4a371c7b2d50c3fe45dfcbf0caf85dba3c48baed68be2f
Do not use this private key on a public blockchain; use it for development purposes only!

DONE

4f8801870ed9311f0f4a371c7b2d50c3fe45dfcbf0caf85dba3c48baed68be2f

# PD-4.1.2 Ganache and MetaMask
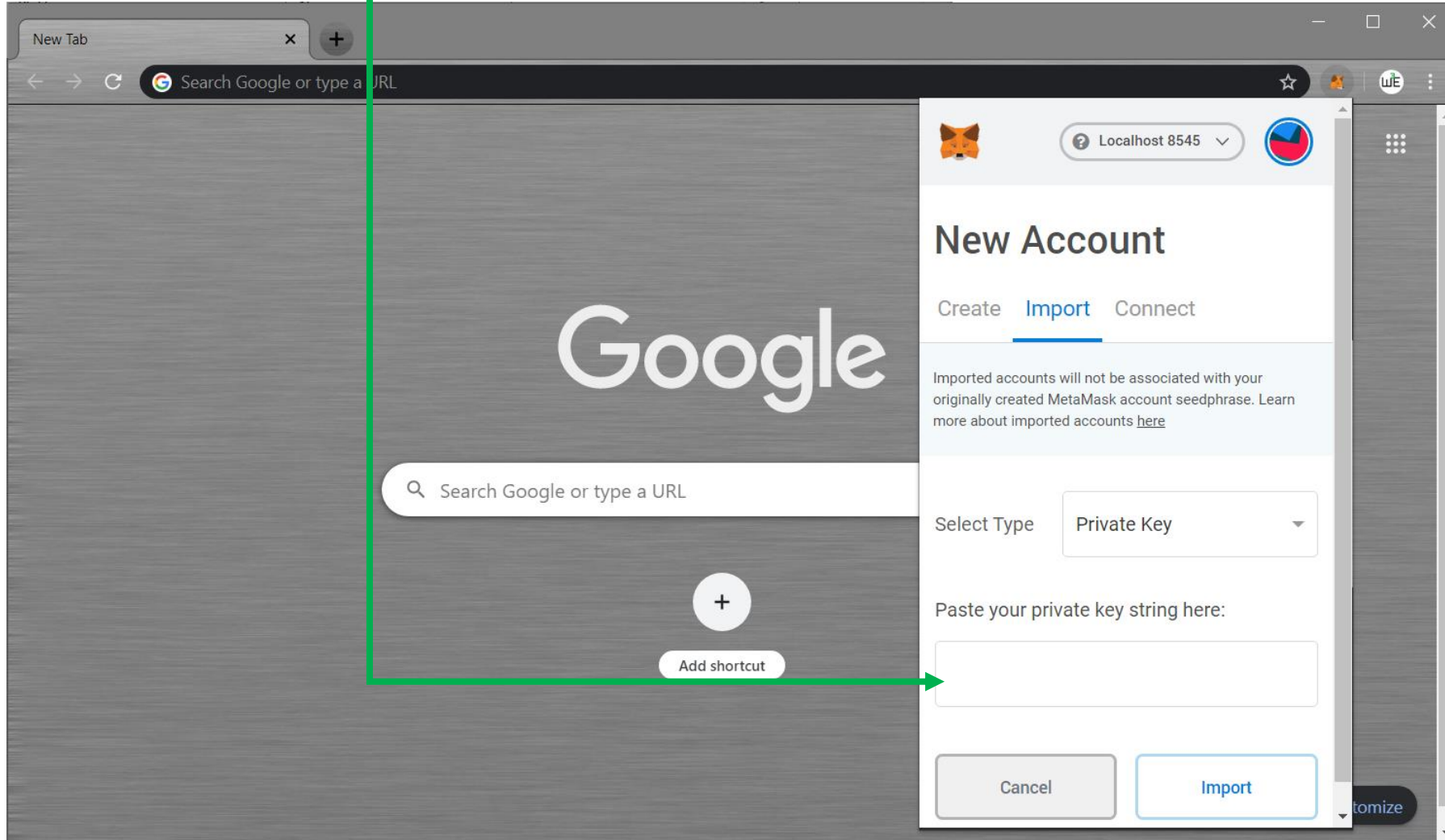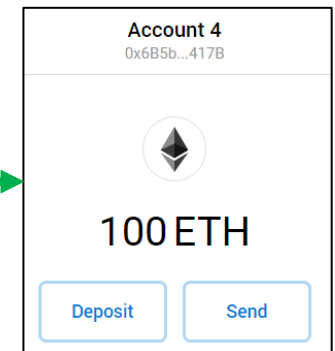
# PD-4.1.3 Ganache and Play Editor

# PD-4.1.4 Ganache and Remix

# PD-4.1.5 Troubleshoot metamask and Ganache
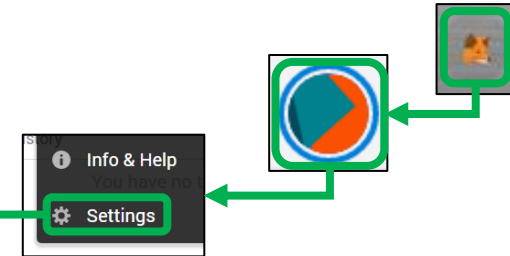
# PD-4.2.1 RPC to Ganache



Client

Ethereum Blockchain

Ganache

RPC interface (poort 7545)
- HTTP(s)
- WS(S) - WebSockets

https://github.com/ethereum/wiki/wiki/JSON-RPC

https://github.com/ethereum/wiki/wiki/JSON-RPC#json-rpc-support
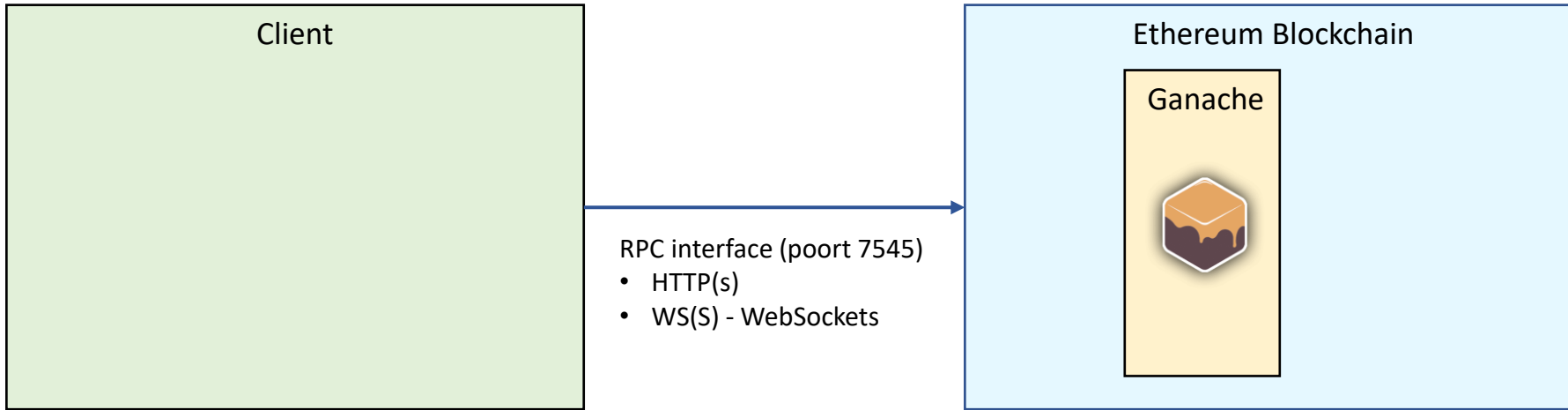
# PD-4.2.1 RPC to Ganache

| Name | Function |
|------|----------|
| web3_clientVersion | Current client version. |

**Windows**
```
set request=web3_clientVersion
set body={"jsonrpc":"2.0","method":"%request%","params":[],"id":67}
curl -H "Content-Type: application/json" -g --data "%body:"=""%" http://localhost:7545
```

**Linux**
```
export request=web3_clientVersion
export body=' {"jsonrpc":"2.0","method":"'$request'","params":[],"id":67} '
curl -H "Content-Type: application/json" -g --data $body http://localhost:7545
```
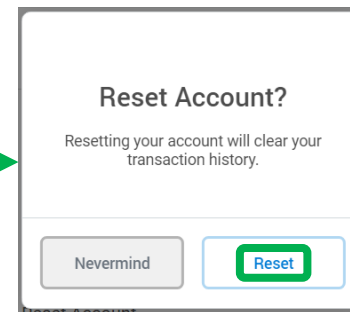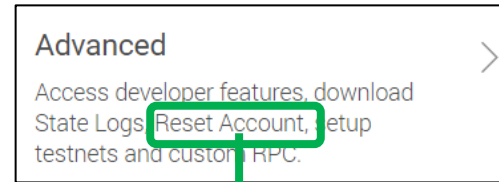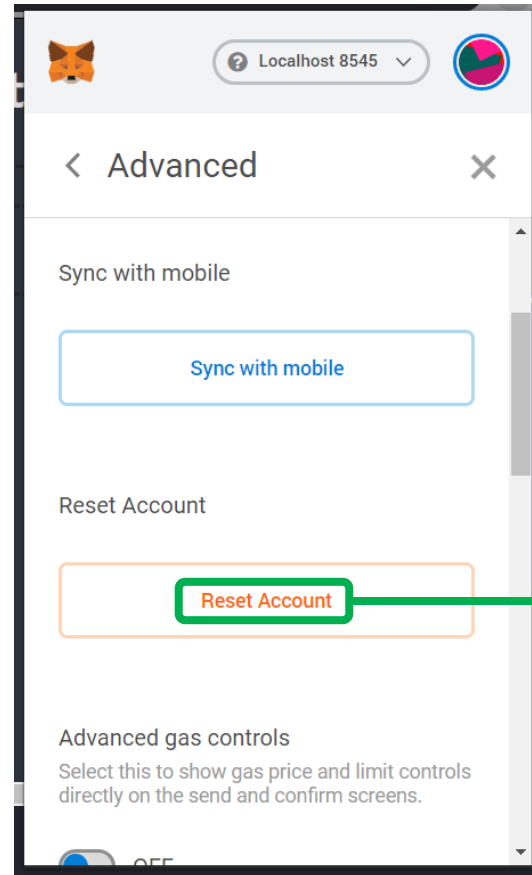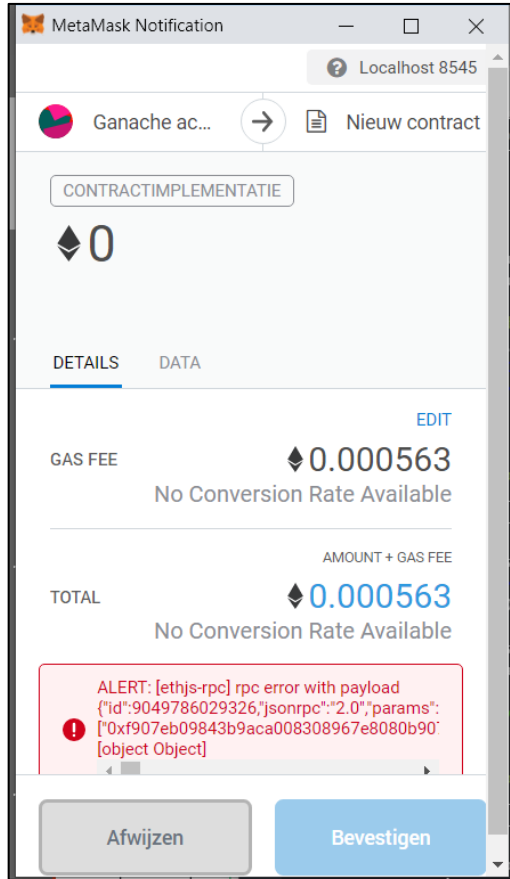
**Result**
```
{"id":67,"jsonrpc":"2.0","result":"EthereumJS TestRPC/v2.8.0/ethereum-js"}
```

https://curl.haxx.se/download.html

https://github.com/ethereum/wiki/wiki/JSON-RPC#web3_clientversion

https://github.com/web3examples/ethereum/blob/master/rpc/web3_clientVersion_port_7545.cmd

# PD-4.2.1 Ganache - RPC eth_accounts

**Windows**
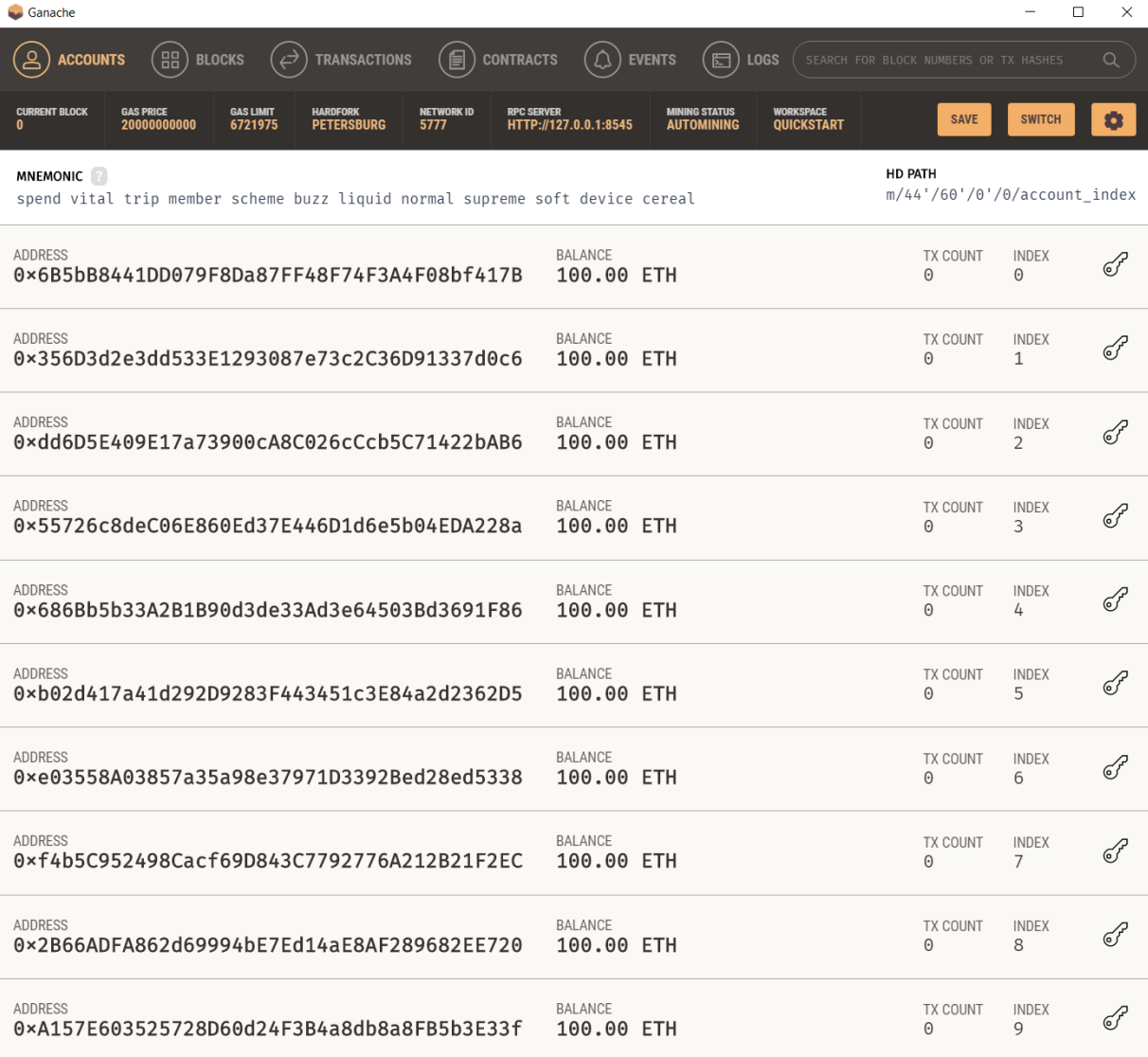
curl -d "{""method"":""eth_accounts ""}"

http://localhost:7545

**Linux**

curl -d '{"method":"eth_accounts "}' http://localhost:7545

**Result**

```
{"result":[
"0x6b5bb8441dd079f8da87ff48f74f3a4f08bf417b",
"0x356d3d2e3dd533e1293087e73c2c36d91337d0c6",
"0xdd6d5e409e17a73900ca8c026cccb5c71422bab6",
"0x55726c8dec06e860ed37e446d1d6e5b04eda228a",
"0x686bb5b33a2b1b90d3de33ad3e64503bd3691f86",
"0xb02d417a41d292d9283f443451c3e84a2d2362d5",
"0xe03558a03857a35a98e37971d3392bed28ed5338",
"0xf4b5c952498cacf69d843c7792776a212b21f2ec",
"0x2b66adfa862d69994be7ed14ae8af289682ee720",
"0xa157e603525728d60d24f3b4a8db8a8fb5b3e33f"
]}
```

https://github.com/web3examples/ethereum/blob/master/rpc/eth_accounts_ganache.cmd
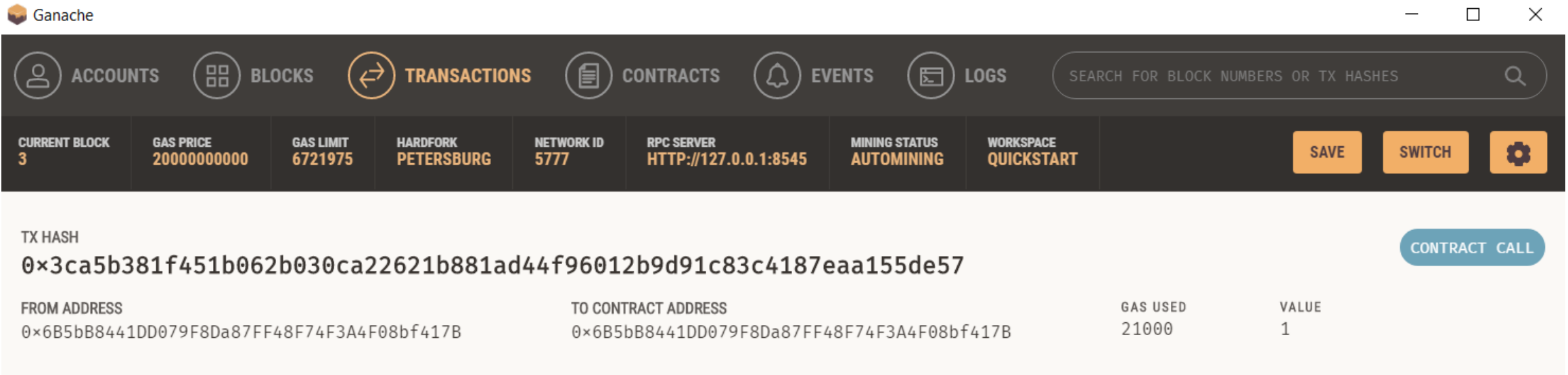
# PD-4.2.1 Ganache - eth_sendTransaction

**Windows:**

set address=0x6B5bB8441DD079F8Da87FF48F74F3A4F08bf417B

set params={ "from": "%address%", "to": "%address%", "value": "0x1" }

set body= {"jsonrpc":"2.0","method":"**eth_sendTransaction**","params":[%params%],"id":1}

curl -H "Content-Type: application/json" -g --data "%body:"=""%" http://localhost:7545

**Result:**

{"id":1,"jsonrpc":"2.0","result":"0x3ca5b381f451b062b030ca22621b881ad44f96012b9d91c83c4187eaa155de57"}

# PD-4.2.2 wscurl subscribe ganache

**Start:**

wscurl  ws://localhost:7545

**Paste:** (without <enter>)

{"id": 1, "method": "web3_clientVersion"}

**Output:**

{"id":1,"result":"**EthereumJS TestRPC/v2.8.0/ethereum-js**"}

# PD-4.2.2 wscurl subscribe ganache

**Start:**
wscurl  ws://localhost:7545

**Paste:** (without <enter>)
{"id": 1, "method": "eth_subscribe", "params": ["newHeads"]}
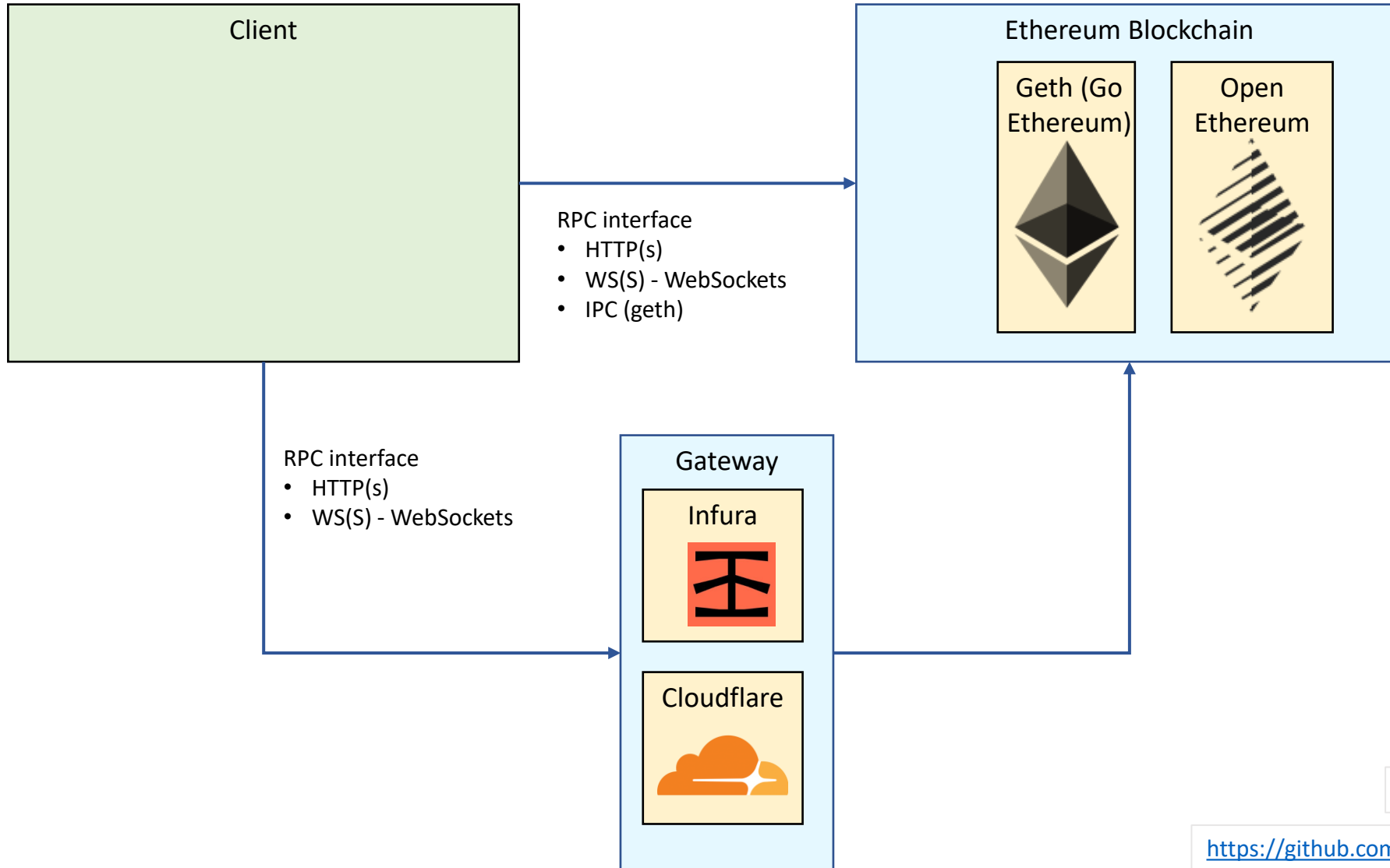
**Output:**
```
{"id":1,"result":"0x7"}
{ jsonrpc": "2.0",
 "method": "eth_subscription",
 "params":
 { "subscription": "0x7",
   "result":
   {  "hash": "0x27d8a20ec365c8a91d3f8919731a87c472c1c570e9199a4c65857281964b6938",
      "parentHash": "0x399173ef5ac8fa65c8ca7466d5830bdfc4ef92a85d76cf1072a24fd9e608069e",
     "sha3Uncles": "0x1dcc4de8dec75d7aab85b567b6ccd41ad312451b948a7413f0a142fd40d49347",
     "miner": "0x0000000000000000000000000000000000000000",
     "stateRoot": "0xf20e4654c8b521ad84a67164b7a27c699292a8e73a19bd08abc6c35a52d192bc",
     "transactionsRoot": "0x7570cbff644607a1bd7b7bb559fed522a8931cc1366c00b2640cc85df2d36173",
     "receiptsRoot": "0x056b23fbba480696b65fe5a59b8f2148a1299103c4f57df839233af2cf4ca2d2",
     "logsBloom": "0x00000…..",
     "difficulty": "0x0",
     "number": "0xa",
     "gasLimit": "0x6691b7",
     "gasUsed": "0x5208",
     "nonce": "0x0000000000000000",
     "mixHash": "0x0000000000000000000000000000000000000000000000000000000000000000",
     "timestamp": "0x5fb3b463",
     "extraData": "0x"
 }
 }
}
```

https://www.npmjs.com/package/wscurl

https://www.npmjs.com/package/wscat

http://web3examples.com/ethereum/demo/Use_ganache_cli_Windows_WS_Subscriptions.html

# PD-4.2.3 RPC Interface



Client

RPC interface
- HTTP(s)
- WS(S) - WebSockets
- IPC (geth)

RPC interface
- HTTP(s)
- WS(S) - WebSockets

Ethereum Blockchain

Geth (Go Ethereum)

Open Ethereum

Gateway

Infura

Cloudflare

https://github.com/ethereum/wiki/wiki/JSON-RPC

https://github.com/ethereum/wiki/wiki/JSON-RPC#json-rpc-support

# PD-4.2.3 Call RPC Functions via web interface



http://localhost:7545

eth_blockNumber

**Input parameters:**

POST

https://cloudflare-eth.com
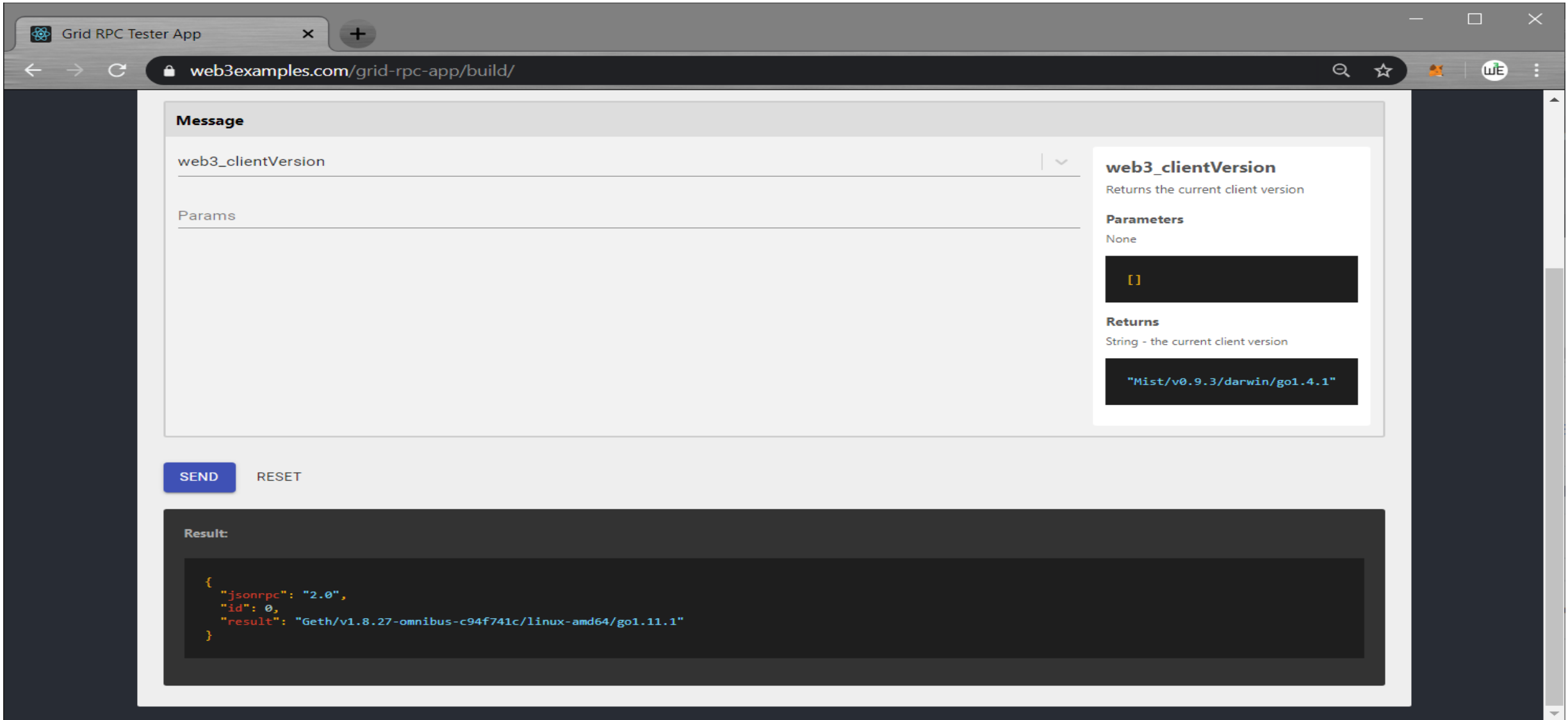
```
{"jsonrpc":"2.0",
"method":"web3_clientVersion",
"params":[],"id":67
}
```

https://www.apirequest.io/

http://web3examples.com/ethereum/demo/Check_contract_balance_via_RPC_online.html

# PD-4.2.3 Grid RPC Tester - web3_clientVersion

# PD-4.2.3 Grid RPC Tester - eth_getBalance

# PD-4.2.3 Web interface eth_blockNumber

```html
1   <body>
2   Get current Ethereum blocknumber<br>
3   Compare to <a href="https://etherscan.io/blocks" target="_blank">Etherscan</a>
4   <pre id="log" style="width:100%;height:200px"></pre>
5   <script>
6   function log(logstr) {
7       document.getElementById("log").innerHTML +=logstr+"\n";
8   }
9   async function f() {
10      const request=JSON.stringify({"jsonrpc":"2.0","method":"eth_blockNumber","params":[],"id":67});
11      log(`Sending ${request}`);
12      const rawResponse = await fetch('https://mainnet.infura.io', {
13          method: 'POST',
14          headers: {
15              'Accept': 'application/json',
16              'Content-Type': 'application/json'
17          },
18          body: request
19      });
20      const content = await rawResponse.json();
21      log(`Received ${JSON.stringify(content)}`);
22      log(`Current blocknumber = ${parseInt(content.result)}`);
23  }
24  f();
25  </script>
26  </body>
```

https://web3examples.com/ethereum/rpc/blocknr.html

# PD-4.2.5 Web interface eth_subscribe (websockets)

```
1   <body>
2   Get latest Ethereum blocknumbers via websockets<br>
3   Compare to <a href="https://etherscan.io/blocks" target="_blank">Etherscan</a>
4   <pre id="log" style="width:100%;height:200px"></pre>
5   <script>
6   function log(logstr) {
7       document.getElementById("log").innerHTML +=logstr+"\n";
8   }
9   const server="wss://mainnet.infura.io/ws";
10  log(`Opening connection to ${server}`);
11  var socket = new WebSocket(server);
12  socket.onopen = function (event) {
13      const request=JSON.stringify({"id": 1, "method": "eth_subscribe", "params": ["newHeads"]});
14      log(`Connection opened, sending: ${request}`);
15      socket.send(request);
16  };
17  socket.onmessage = async function (event) {
18      log(`Received ${event.data}`);
19      var data=JSON.parse(event.data);
20      log(`Current blocknumber = ${parseInt(data.params.result.number)}`);
21  }
22  socket.onerror = function(error) {
23      log(`[error] ${error.message}`);
24  };
25  </script>
26  </body>
```
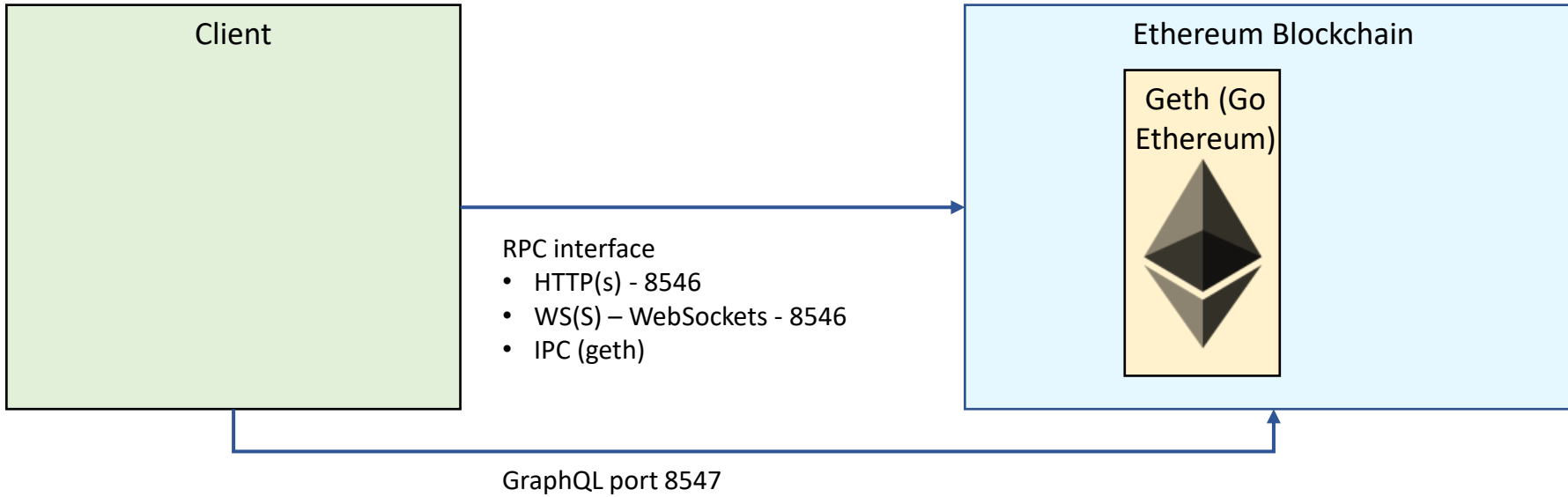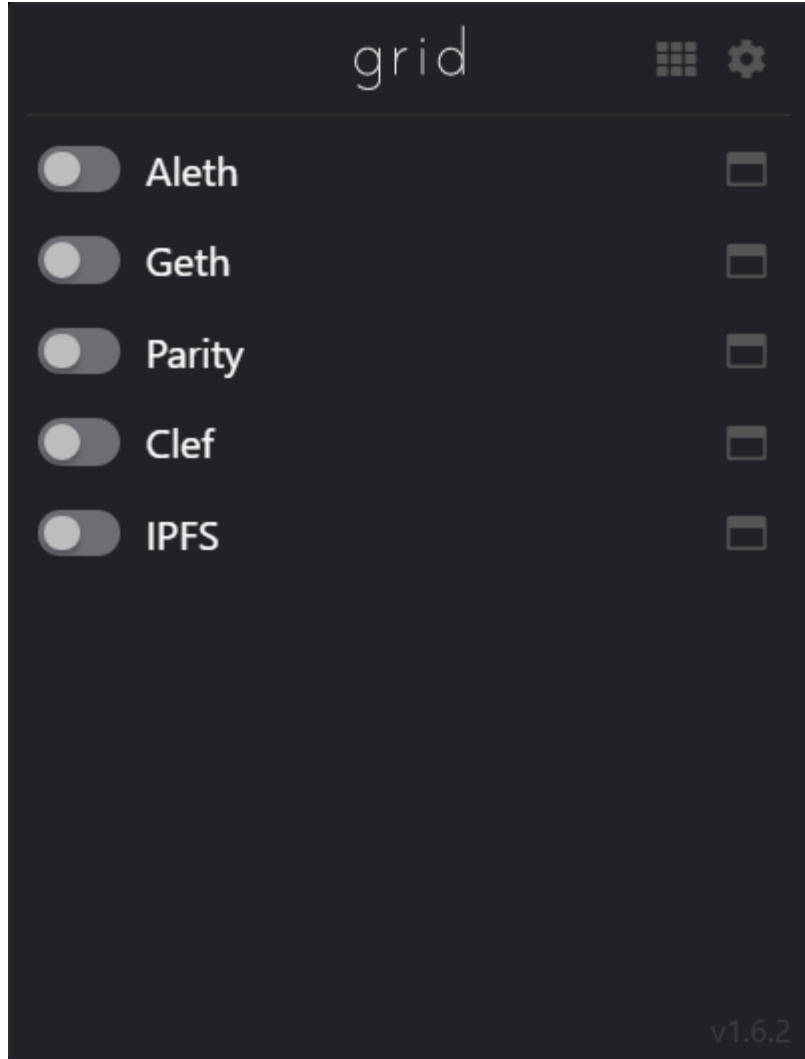
# PD-4.3 RPC Interface Geth (via Grid)



Client

Ethereum Blockchain

Geth (Go Ethereum)

RPC interface
- HTTP(s) - 8546
- WS(S) – WebSockets - 8546
- IPC (geth)

GraphQL port 8547

https://github.com/ethereum/wiki/wiki/JSON-RPC

https://github.com/ethereum/wiki/wiki/JSON-RPC#json-rpc-support

# PD-4.3 Ethereum - Grid

# PD-4.3 Grid - Geth

# PD-4.3 Geth ports

# PD-4.3 Geth - node

# PD-4.4 Management with Grafana & Prometheus

Browser

Grafana

Prometheus

Geth (Go Ethereum)

Port 3000

Port 9090

Port 6060
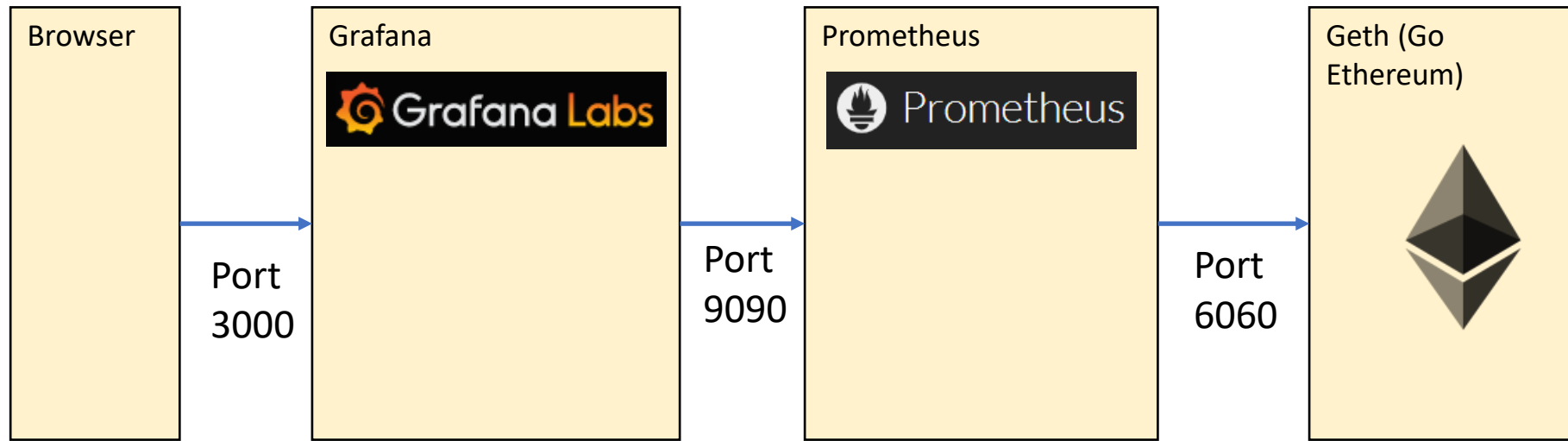
# PD-4.4 Configure Geth (via Grid)



**Add parameters:**
--pprof --metrics --metrics.expensive

# PD-4.4 Start Geth (via Grid)

# PD-4.4 Install  & configure Prometheus

**prometheus.yml**

```
global:
  scrape_interval: 5s
  scrape_timeout: 5s
  evaluation_interval: 5s
alerting:
  alertmanagers:
  - static_configs:
    - targets: []
    scheme: http
    timeout: 5s
scrape_configs:
- job_name: geth
  metrics_path: /debug/metrics/prometheus
  scheme: http
  static_configs:
  - targets: ['localhost:6060']
```

https://prometheus.io

http://web3examples.com/ethereum/grafana_prometheus/prometheus.yml

https://github.com/karalabe/geth-prometheus/blob/master/prometheus.yml

# PD-4.4 Start Prometheus

# PD-4.4 Install Grafana

# PD-4.4 Configure datasource



http://localhost:3000/
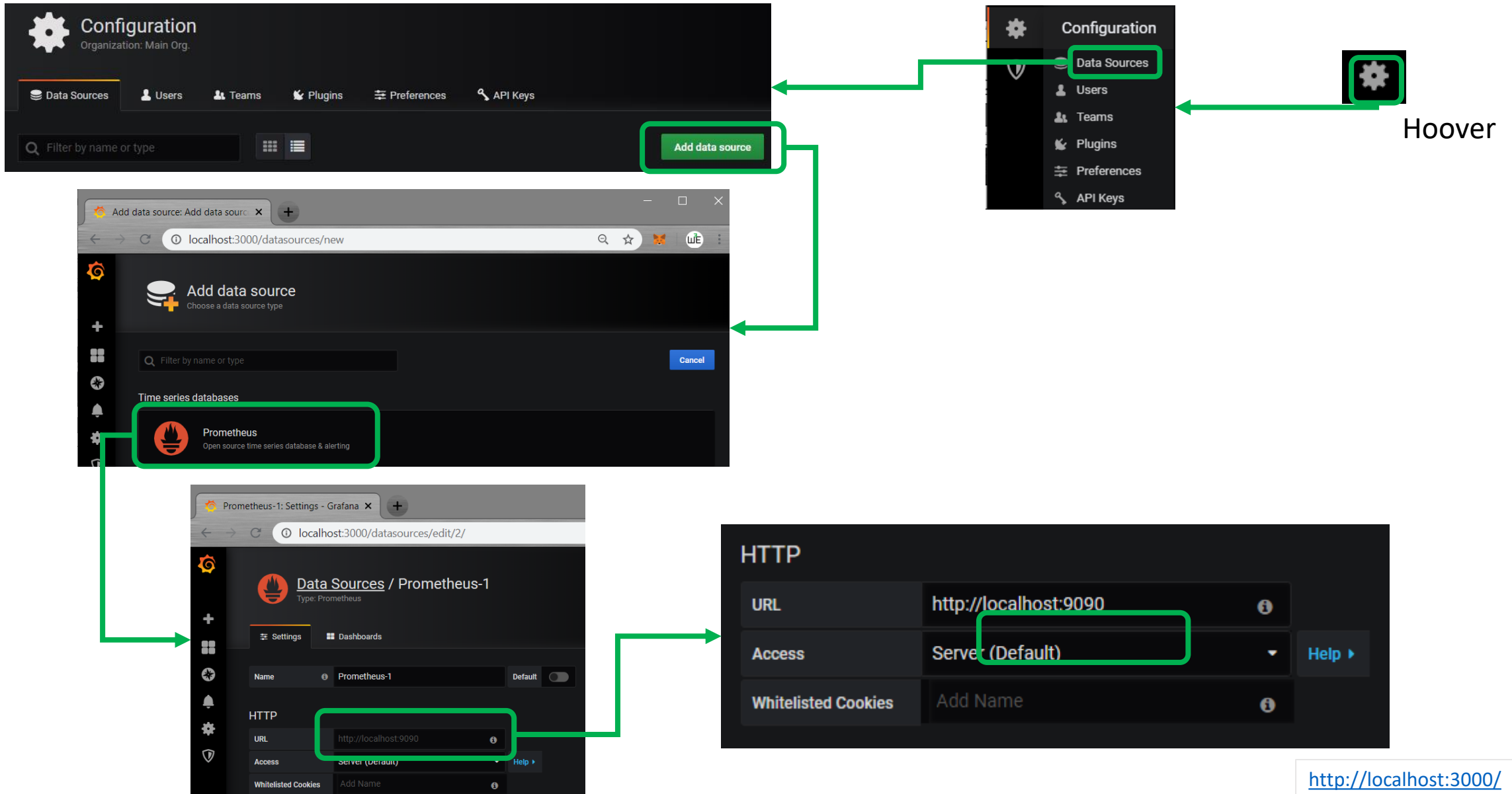
# PD-4.4 Configure Dashboard



Hoover
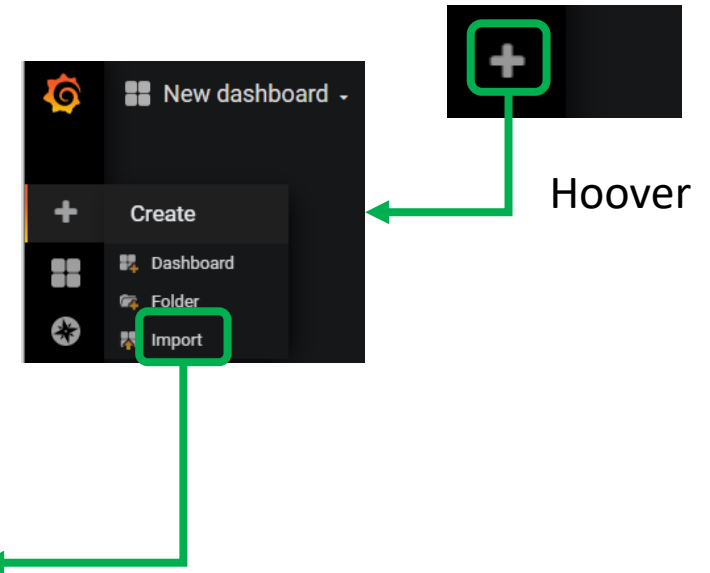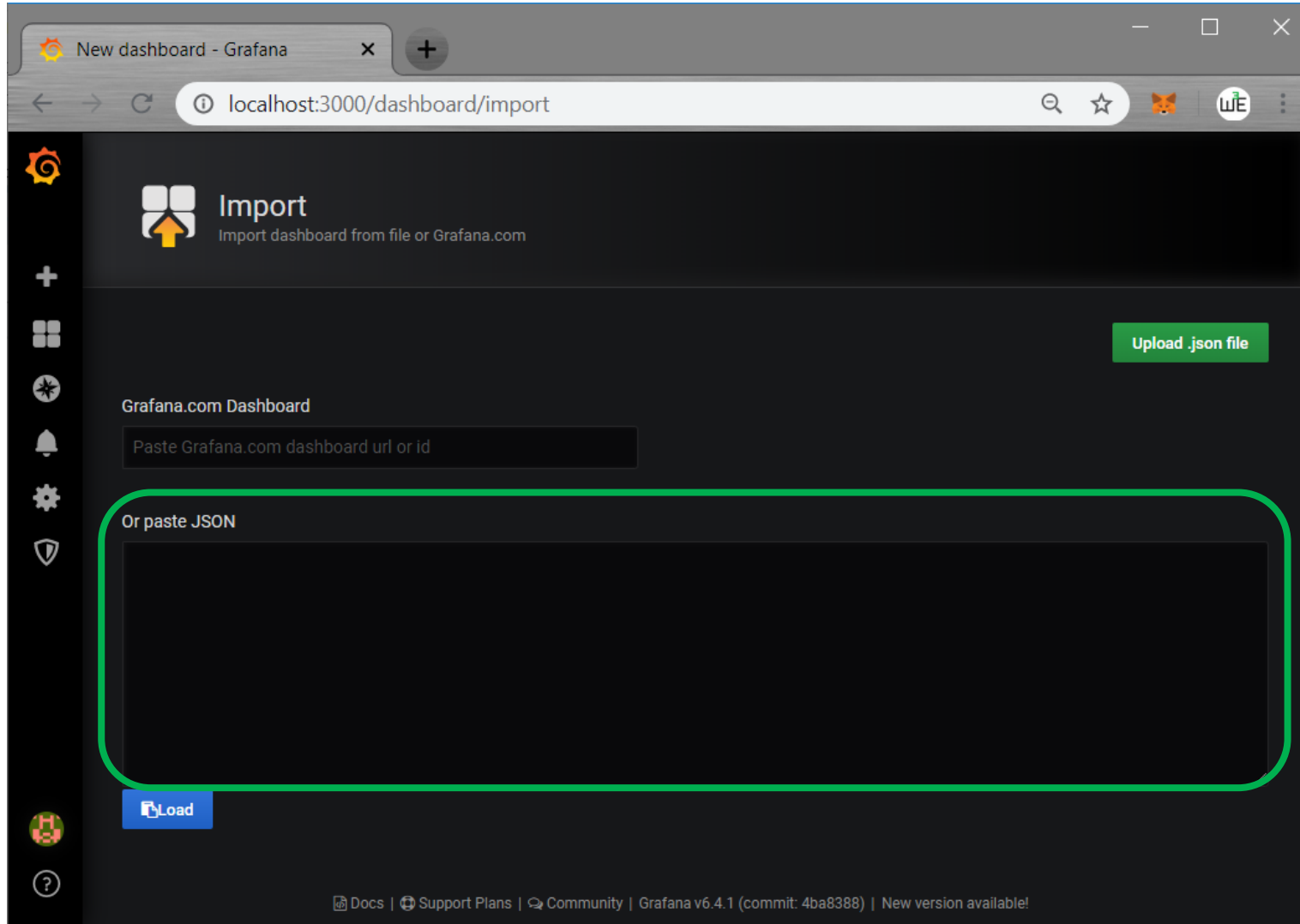
# PD-4.4 Dashboard configuration

# PD-4.4 Start Dashboard

# PD-4.5 RPC Interface



**Client**

RPC interface
- HTTP(s)
- WS(S) - WebSockets
- IPC (geth)

RPC interface
- HTTP(s)
- WS(S) - WebSockets

**Ethereum Blockchain**

Geth (Go Ethereum)

Open Ethereum

**Gateway**

Infura

Cloudflare

https://eth.wiki/json-rpc/api
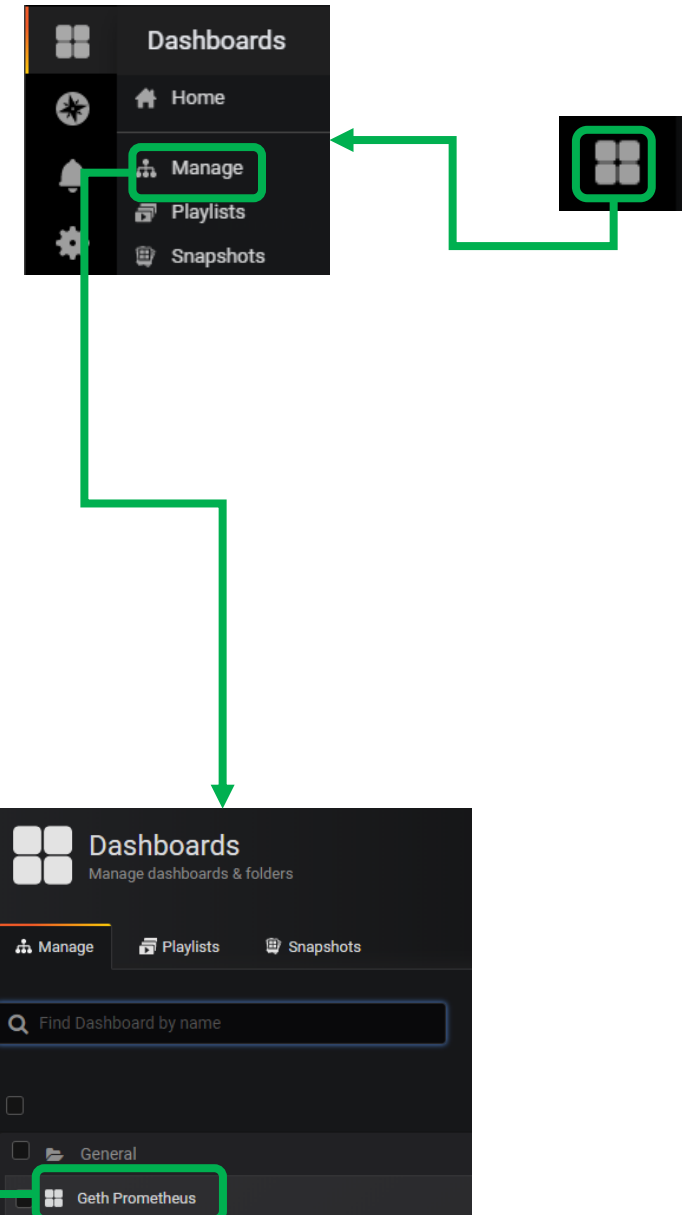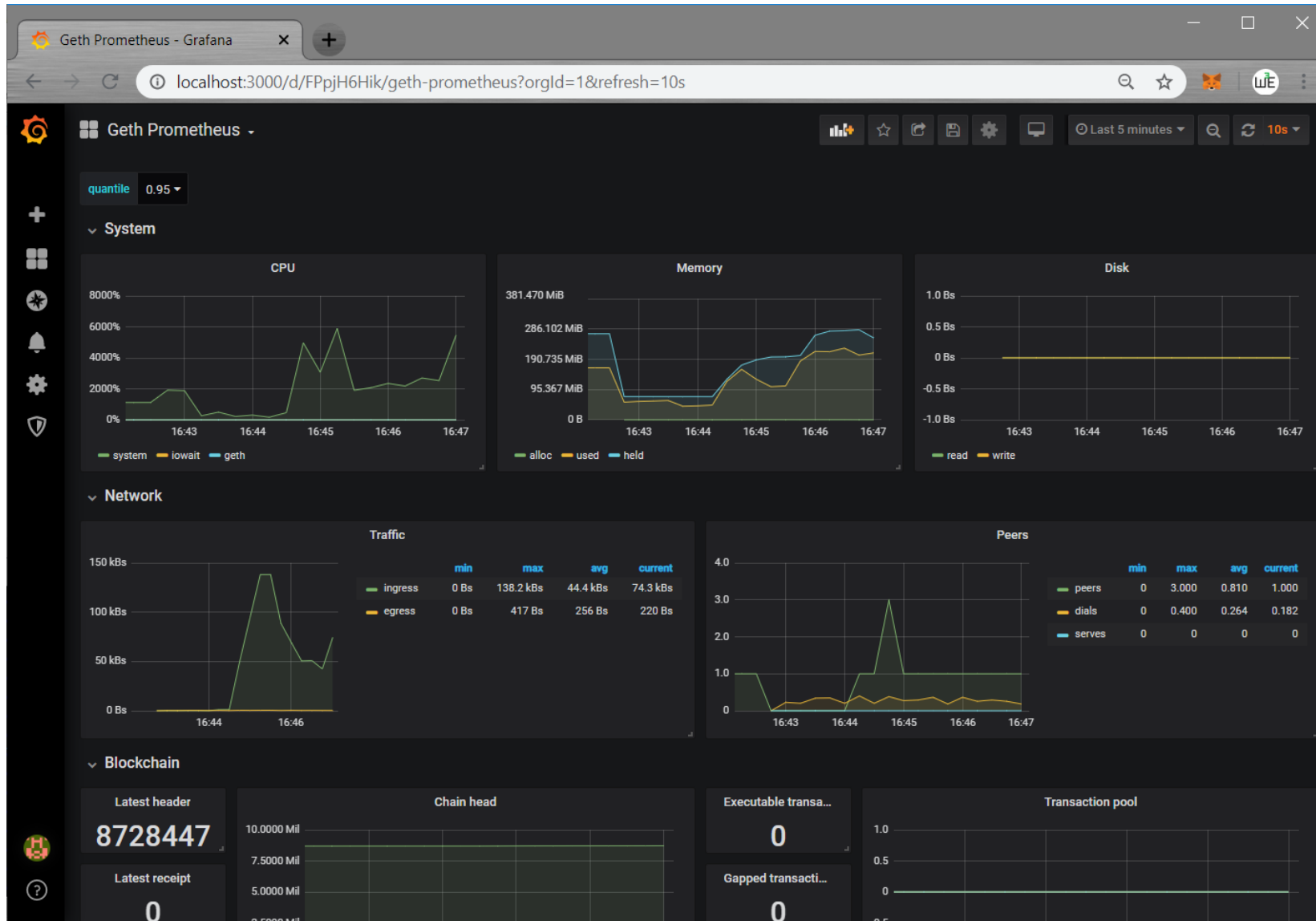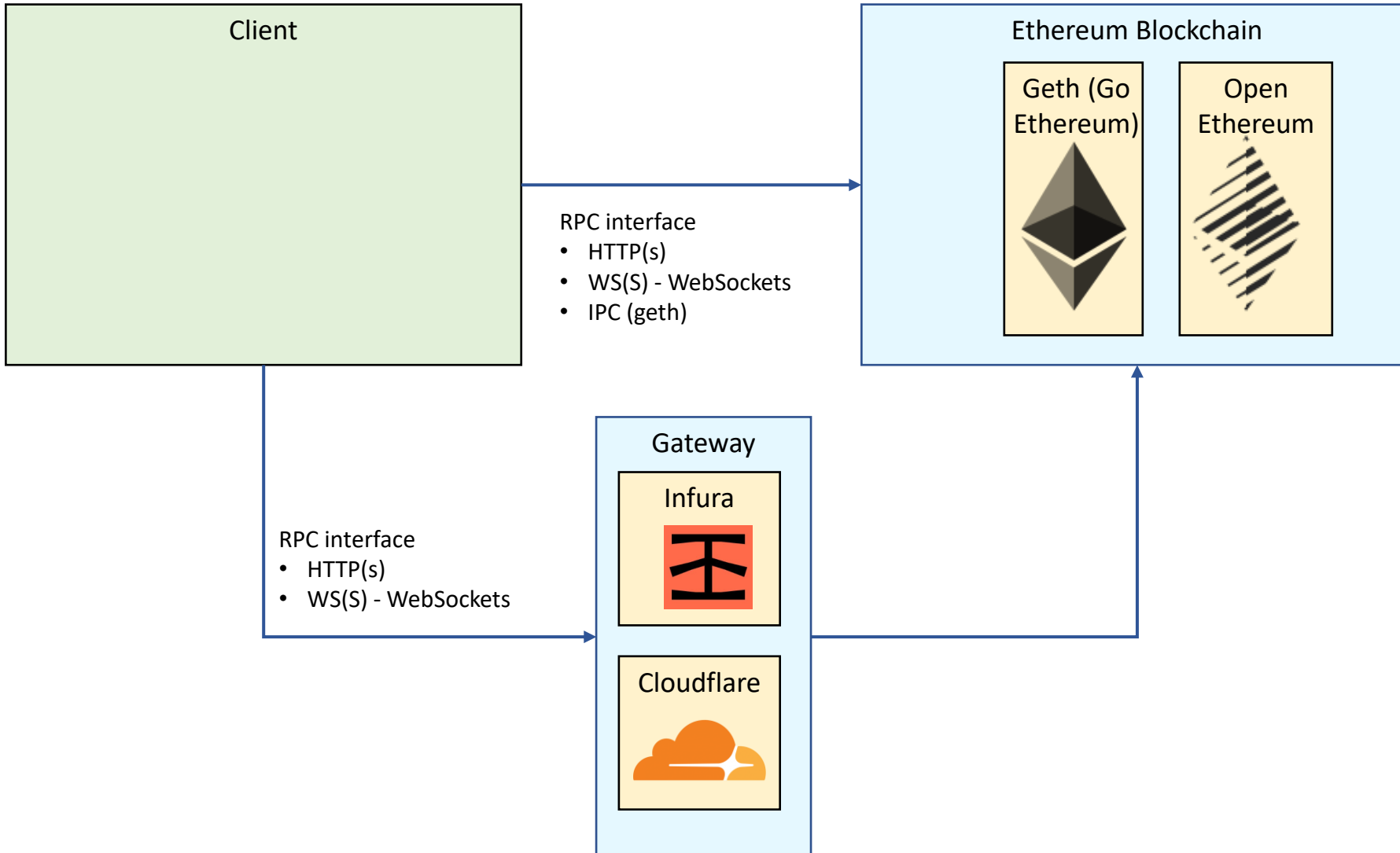
# PD-4.5 RPC Calls General information

| Name | Function |
|---|---|
| web3_clientVersion | Current client version |
| eth_gasPrice | Current price per gas in wei |
| eth_hashrate | Number of hashes per second that the node is mining with |
| eth_mining | Is client actively mining new blocks? |
| eth_protocolVersion | Current ethereum protocol version |
| eth_syncing | Is client syncing? |
| net_listening | Is client actively listening for network connections? |
| net_peerCount | Number of peers currently connected to the client |

https://github.com/web3examples/ethereum/blob/master/rpc/general_info_mainnet.cmd

https://eth.wiki/json-rpc/api

# PD-4.5 RPC Calls Network information

| Name | Function |
|------|----------|
| eth_chainId | Current chain id |
| net_version | Current network id   (mainnet usually 1 ) |

## EVM Networks

A list of EVM networks. Wallets and Web3 middleware providers should be able to use the appropriate Chain ID and Network ID to connect to the correct chain.

This table is also available as a JSON feed at chains.json.

| Chain ID | Name | Short Name | Chain | Network | Network ID |
|----------|------|------------|-------|---------|------------|
| 1 | Ethereum Mainnet | eth | ETH | mainnet | 1 |
| 3 | Ethereum Testnet Ropsten | rop | ETH | ropsten | 3 |
| 42 | Ethereum Testnet Kovan | kov | ETH | kovan | 42 |
| 43110 | Athereum | ath | ATH | athereum | 43110 |
| 5 | Ethereum Testnet Görli | gor | ETH | goerli | 5 |
| 6 | Ethereum Classic Testnet Kotti | kot | ETC | kotti | 6 |

https://chainid.network/

https://eth.wiki/json-rpc/api

# PD-4.5 RPC Calls Block & log information

| Name | Function |
| --- | --- |
| eth_blockNumber | Number of most recent block |
| eth_getBlockByHash | Information about a block |
| eth_getBlockByNumber | Information about a block |
| eth_getStorageAt | Value from a storage position at a given address |
| eth_getCode | Code at a given address |
| eth_getLogs | All logs matching a given filter object |

https://eth.wiki/json-rpc/api

# PD-4.5 RPC Calls Transaction information

| Name | Function |
| --- | --- |
| eth_getBlockTransactionCountByHash | Number of transactions in a block from a block matching the given block hash |
| eth_getBlockTransactionCountByNumber | Number of transactions in a block matching the given block number |
| eth_getTransactionByBlockHashAndIndex | Transaction by block hash and transaction index position |
| eth_getTransactionByBlockNumberAndIndex | Transaction by block number and transaction index position. |
| eth_getTransactionByHash | Transaction requested by transaction hash |
| eth_pendingTransactions | Returns the pending transactions list |

https://eth.wiki/json-rpc/api

# PD-4.5 RPC Calls Uncles

| Name | Function |
|------|----------|
| eth_getUncleCountByBlockHash | Number of uncles in this block |
| eth_getUncleCountByBlockNumber | Number of uncles in this block |
| eth_getUncleByBlockHashAndIndex | Information about a uncle |
| eth_getUncleByBlockNumberAndIndex | Information about a uncle |

**Ethereum uncle**
- orphan block
- mined just after a correct block was added to the main chain
- valid block that arrives too late
- a block can include max two uncles

**Reward for miner**
- each uncle referenced earns 1/32 of block reward

**Rewards to miner of uncle block**
- depends on distance to the main chain block it was referenced by
- max six generations
- max 7/8 block reward

https://etherscan.io/uncles

https://eth.wiki/fundamentals/mining

# PD-4.5 RPC Calls Account information

| Name | Function |
| --- | --- |
| eth_accounts | List of addresses owned by client |
| eth_coinbase | Client coinbase address |
| eth_getBalance | Balance of the account of given address |
| eth_getProof | Account- and storage-values of the specified account including the Merkle-proof |
| eth_getTransactionCount | Number of transactions sent from an address |

# PD-4.5 RPC Calls Transactions & function calls

| Name | Function |
|------|----------|
| eth_sign | The sign method calculates an Ethereum specific signature with: sign(keccak256("\x19Ethereum Signed Message:\n" + len(message) + message))).<br>**Note** the address to sign with must be unlocked. |
| eth_estimateGas | Generates and returns an estimate of how much gas is necessary to allow the transaction to complete. The transaction will not be added to the blockchain. |
| eth_call | Executes a new message call immediately without creating a transaction on the block chain (e.g. read only) |

| Name | Function |
|------|----------|
| eth_sendTransaction | Creates new message call transaction or a contract creation, if the data field contains code |
| eth_getTransactionReceipt | Get the contract address, after the transaction was mined, when you created a contract. |
| eth_sendRawTransaction | Creates new message call transaction or a contract creation for signed transactions. |

https://eth.wiki/json-rpc/api

# PD-4.5 RPC Calls Utilities

| Name | Function |
|------|----------|
| web3_sha3 | Returns Keccak-256 hash of the given data<br>Note: not the standardized SHA3-256 |

**Windows**
set **request**=web3_sha3
set **param**="0x68656c6c6f20776f726c64"set
**body**={"jsonrpc":"2.0","method":"**%request%**","params":[%param%],"id":67}
curl -H "Content-Type: application/json" -g --data "**%body:"=""%**"  https://mainnet.infura.io

**Result**
{
  "jsonrpc":"2.0",
  "id":67,
  "result":"**0x47173285a8d7341e5e972fc677286384f802f8ef42a5ec5f03bbfa254cb01fad**"
}

https://github.com/web3examples/ethereum/blob/master/rpc/web3_sha3_mainnet.infura.io.cmd

https://eth.wiki/json-rpc/api

# PD-4.5 RPC Calls Mining

| Name | Function |
|------|----------|
| eth_mining | Is client is actively mining new blocks? |
| eth_hashrate | Number of hashes per second that the node is mining with |
| eth_submitWork | Submit a proof-of-work solution |
| eth_getWork | Returns the hash of the current block, the seedHash, and the boundary condition to be met ("target") |

https://eth.wiki/json-rpc/api

# PD-4.5 RPC Calls Filters

| Name | Function |
|------|----------|
| eth_newBlockFilter | Filter to notify when a new block arrives |
| eth_newPendingTransactionFilter | Filter to notify when new pending transactions arrive. |
| eth_getFilterChanges | Polling method for a filter, which returns an array of logs which occurred since last poll. |
| eth_newFilter | Filter to notify logs changes |
| eth_getFilterLogs | Returns an array of all logs matching filter with given id. |
| eth_uninstallFilter | Uninstalls a filter with given id. Should always be called when watch is no longer needed. |

# PD-4.6 eth_call


Keccak-256 Online — Online Tools

**Keccak-256**
Keccak-256 online hash function

```
name()
```
Input type: Text

Hash ☑ Auto Update

```
06fdde0383f15d582d1a74511486c9ddf862a882fb7904b3d9fe9b8b8e58a796
```

**Function:**
name()

| Name | Function |
|------|----------|
| eth_call | Executes a new message call immediately without creating a transaction on the block chain. |


Function Signature Database — 4byte.directory/signatures/?bytes4_signature=3b668059

Ethereum Function Signature Database

Browse Signatures  Submit Signatures  Submit ABI  Submit Solidity Source File  API Docs

**Search Signatures** `3b668059` [Search]

| ID | Text Signature | Bytes Signature |
|----|----------------|-----------------|
| 101293 | ContractBalance() | 0x3b668059 |

**3b668059**d6bfbb6e1d86e2e5ec1727b55e73f334c0abbebf0cf96aba42a485ab

**Enter command:**
curl -H "Content-Type: application/json" -d
"{""jsonrpc"":""2.0"",""method"":""eth_call"",""params"":[{""from"":""0xEA9a7c7cD8d4Dc3acc6f0AaEc1506C8D6041a1c5"",""to"":""0x01848a2cFc60d5b24EB6630F12048C1825059AfC"",""data"":""0x3b668059""},""latest""],""id"":67}"
https://ropsten.infura.io

**Output:**
{"jsonrpc":"2.0","id":67,"result":"0x00000000000000000000000000000000000000000000000000002b4c77783338000"}

# PD-4.7 RPC URL's and testnetworks

| Mainnet URL | Websockets | web3_clientVersion |
|---|---|---|
| https://mainnet.infura.io/v3/.. | wss://mainnet.infura.io/ws/v3/.. | Geth/v1.8.27-omnibus-c94f741c/linux-amd64/go1.11.1 |
| https://cloudflare-eth.com | | cloudflare-geth |
| https://main-rpc.linkpool.io | wss://main-rpc.linkpool.io/ws | OpenEthereum//v3.0.1-stable-8ca8089-20200601/x86_64-unknown-linux-gnu/rustc1.43.1 |
| https://rpc.slock.it/mainnet | | Parity-Ethereum//v2.5.13-stable-253ff3f-20191231/x86_64-linux-gnu/rustc1.40.0 |

# PD-4.7 MetaMask test networks
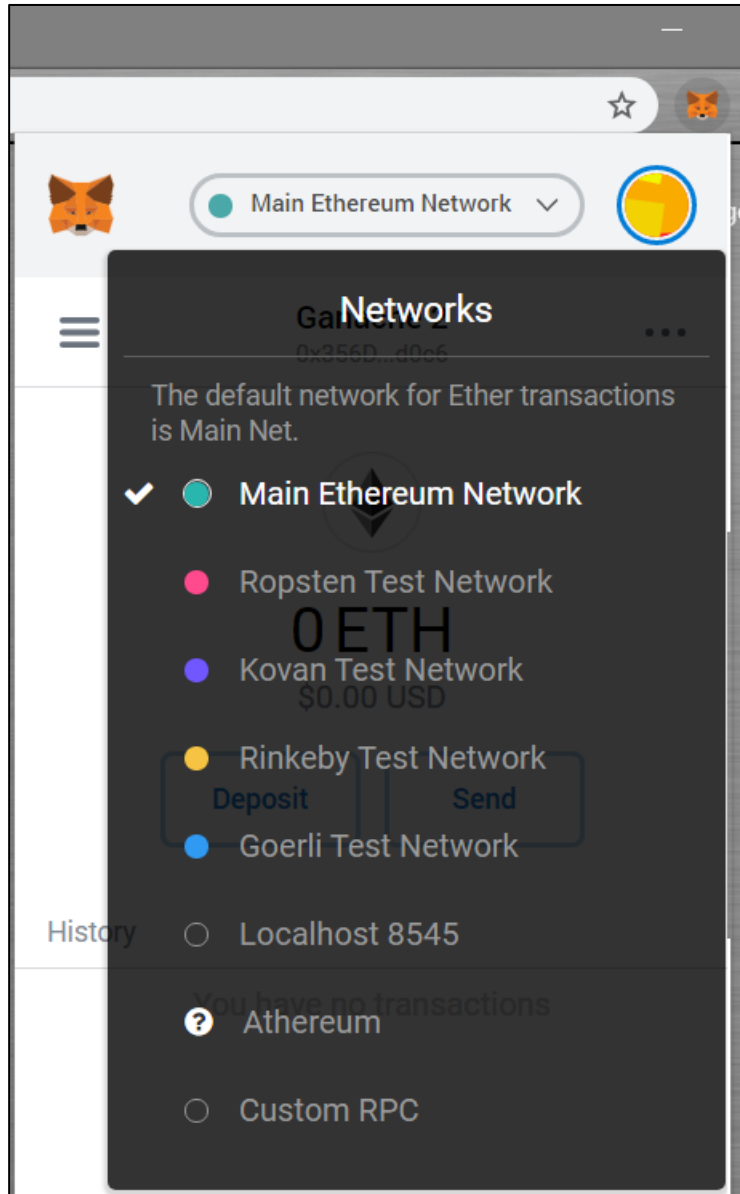
# PD-4.7 Test networks

| Network | Ropsten | Kovan | Rinkeby | Görli | Athereum |
|---|---|---|---|---|---|
| Consensus Protocol | Proof of Work (best emulates mainnet environment) | Proof of Authority | Proof of Authority (Clique) | Proof of Authority (Clique) | Avalanche |
| Supported Clients | Both Geth / Parity | Parity only | Geth only | Multi-client | AVA |
| Average Block Times | ~15 sec. | ~4 sec. | ~15 sec. | ~16 sec. | < 1 sec |

# PD-4.7 Chain registry

```
1   {
2     "name": "Ethereum Testnet Ropsten",
3     "chain": "ETH",
4     "network": "ropsten",
5     "rpc": [
6       "https://ropsten.infura.io/v3/${INFURA_API_KEY}"
7     ],
8     "faucets": [
9       "https://faucet.ropsten.be?${ADDRESS}"
10    ],
11    "nativeCurrency": {
12      "name": "Ropsten Ether",
13      "symbol": "ROP",
14      "decimals": 18
15    },
16    "infoURL": "https://github.com/ethereum/ropsten",
17    "shortName": "rop",
18    "chainId": 3,
19    "networkId": 3,
20    "ens": {
21      "registry":"0x112234455c3a32fd11230c42e7bccd4a84e02010"
22    }
23  }
```

https://github.com/ethereum-lists/chains

https://github.com/ethereum-lists/chains/blob/master/_data/chains/3.json

https://chainid.network/chains.json

https://chainid.network

# PD-4.7 RPC URL's

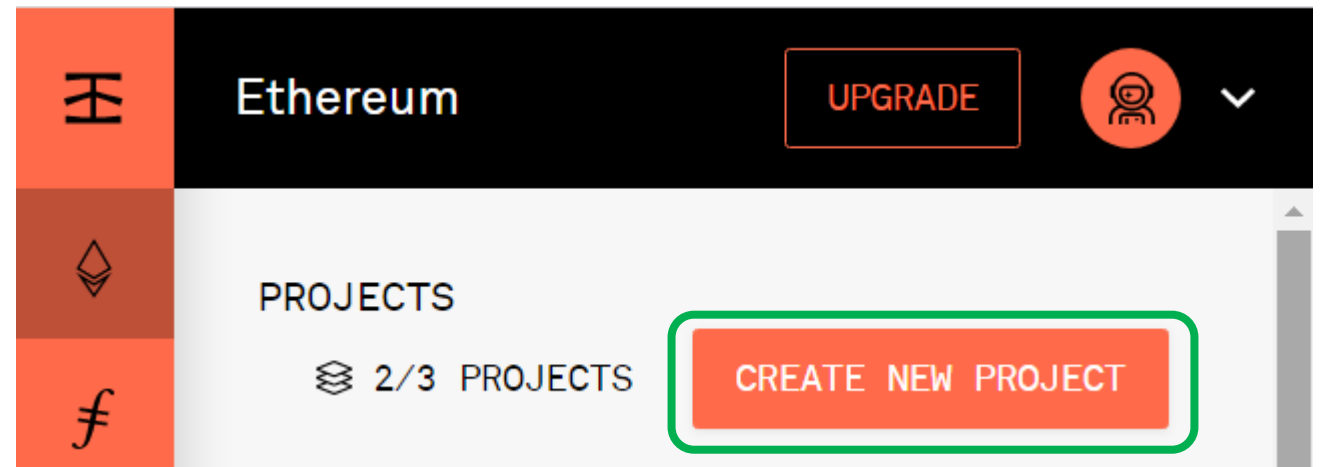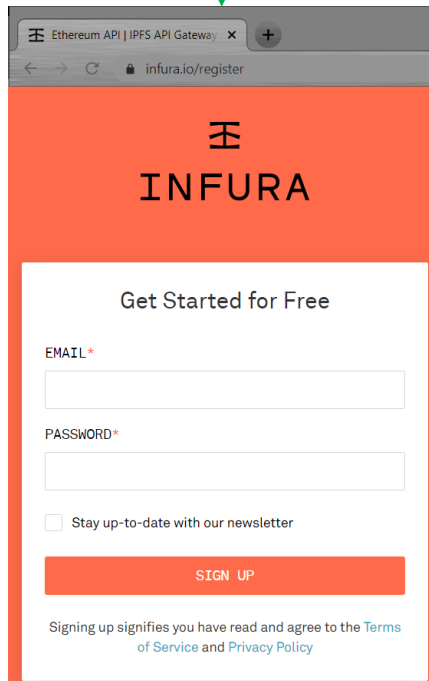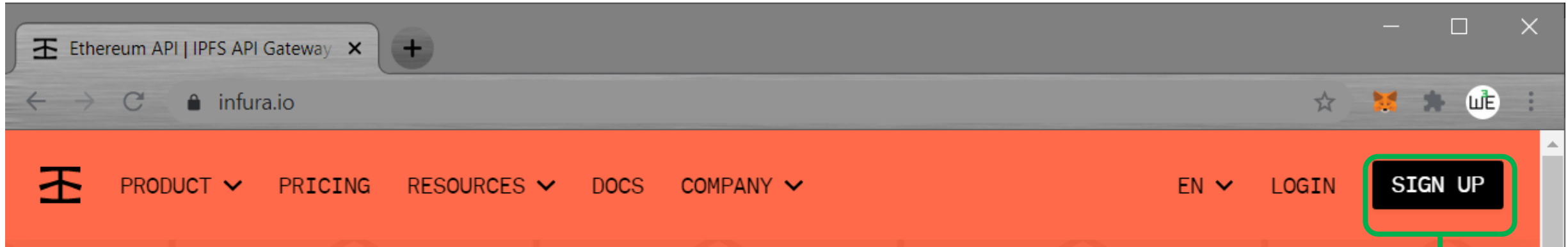| Test URL | Websockets | web3_clientVersion |
|---|---|---|
| https://ropsten.infura.io/v3/.. | wss://ropsten.infura.io/ws/v3/.. | Geth/v1.9.5-omnibus-6ef96469-20190920/linux-amd64/go1.12 |
| https://kovan.infura.io/v3/.. | wss://kovan.infura.io/ws/v3/.. | Parity-Ethereum//v2.6.4-beta-badb045-20190926/x86_64-linux-gnu/rustc1.36.0 |
| https://rinkeby.infura.io/v3/.. | wss://rinkeby.infura.io/ws/v3/.. | Geth/v1.9.6-omnibus-81a21b6f-20191004/linux-amd64/go1.12 |
| https://goerli.infura.io/v3/.. | wss://goerli.infura.io/ws/v3/.. | Geth/v1.9.6-omnibus-81a21b6f-20191004/linux-amd64/go1.12 |
| https://rpc.goerli.mudit.blog | | Geth/v1.9.6-unstable-311419c7-20190923/linux-amd64/go1.11.5 |
| https://rpc.slock.it/goerli | | Slock.it-Proxy//v1.0.0 |
| https://goerli.prylabs.net | wss://goerli.prylabs.net/websocket | Geth/v1.9.6-stable-bd059680/linux-amd64/go1.13.1 |
| https://api.avax-test.network/ext/bc/C/rpc | | |

# PD-4.7 Testnet information

| Network | More info | Chainnr | Faucet | Github |
|---------|-----------|---------|--------|--------|
| Ropsten | https://github.com/ethereum/ropsten | 3 | https://faucet.metamask.io<br>https://faucet.ropsten.be<br>https://teth.bitaps.com | https://github.com/ethereum/ropsten |
| Kovan | https://kovan-testnet.github.io/website | 42 | https://faucet.kovan.network<br>https://gitter.im/kovan-testnet/faucet | https://github.com/kovan-testnet/proposal |
| Rinkeby | https://www.rinkeby.io | 4 | https://faucet.rinkeby.io | https://github.com/ethereum/EIPs/issues/225 |
| Goerli | https://goerli.net | 5 | https://goerli-faucet.slock.it<br>https://faucet.goerli.mudit.blog | https://github.com/goerli/testnet |
| Athereum | https://athereum.avax.network | 0xa869 | https://faucet.avax-test.network | https://github.com/ava-labs |

# PD-4.7 Testnet explorers & dashboards

| Network | Dashboard | Explorer |
| --- | --- | --- |
| Ropsten | | https://ropsten.etherscan.io<br>https://blockexplorer.one/eth/ropsten<br>https://teth.bitaps.com |
| Kovan | | https://kovan.etherscan.io |
| Rinkeby | https://stats.rinkeby.io | https://rinkeby.etherscan.io<br>https://blockexplorer.one/eth/rinkeby |
| Goerli | https://stats.goerli.net | https://goerli.etherscan.io |
| Athereum | | https://cchain.explorer.avax-test.network |

# PD-4.8 Request Infura API key