

**PROJECT 10**  
**MAKING A BLOCKCHAIN SURVEY WITH**  
**MULTICHAIN**  
***SPRING TRIMESTER 2022***  
**By**  
***NIKHIL PATEL***

**OVERVIEW** :- In this lab, I will learn how to use multichain at the application level using PHP and multichain through API. Here I learn about a voting system in which voters receive voting tokens that they may use to vote for a certain candidate, blockchain is used for voting, and each voter can view the total number of votes cast for a candidate at any moment.

**PROCEDURE** :-

**1) Removing Old Multichain on the Poll Server**

Go to one of the machine and open console and execute this command:

**ps aux | grep multi**

As a result if any Multichain daemon is running you will see on right hand side with the command line argument. Note down the id number for next step.

For each and every multichain process execute this command to kill it:

**kill 19066( change process as per yours)**

Execute this command to remove the old multichain data form the server:

**rm -rf ~/. multichain**

**2) Installing MultiChain on the Poll Server**

To install multichain on the server follow the steps mentioned below:

**cd /tmp**

**wget http://www.multichain.com/download/multichain-1.0.1.tar.gz**

**tar -xvzf multichain-1.0.1.tar.gz**

**cd multichain-1.0.1**

**sudo mv multichaind multichain-cli multichain-util /usr/local/bin**

```
elzorro@ubuntu:~$ ps aux | grep multi
elzorro  1597  0.0  0.0 14224  944 pts/1    S+   07:59   0:00 grep --color=auto mult
1
elzorro@ubuntu:~$ kill 1597
bash: kill: (1597) - No such process
elzorro@ubuntu:~$ cd /tmp
elzorro@ubuntu:/tmp$
elzorro@ubuntu:/tmp$ wget http://www.multichain.com/download/multichain-1.0.1.tar.gz
--2022-07-14 08:05:52--  http://www.multichain.com/download/multichain-1.0.1.tar.gz
Resolving www.multichain.com (www.multichain.com)... 162.243.214.85
Connecting to www.multichain.com (www.multichain.com)[162.243.214.85]:80... connected.
HTTP request sent, awaiting response... 301 Moved Permanently
Location: https://www.multichain.com/download/multichain-1.0.1.tar.gz [following]
--2022-07-14 08:05:58--  https://www.multichain.com/download/multichain-1.0.1.tar.gz
Connecting to www.multichain.com (www.multichain.com)[162.243.214.85]:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 10055511 (9.6M) [application/x-gzip]
Saving to: 'multichain-1.0.1.tar.gz'

multichain-1.0.1.tar.gz  100%[=====] 9.59M  204KB/s  in 74s

2022-07-14 08:07:12 (133 KB/s) - 'multichain-1.0.1.tar.gz' saved [10055511/10055511]

elzorro@ubuntu:/tmp$
elzorro@ubuntu:/tmp$ tar -xvzf multichain-1.0.1.tar.gz
multichain-1.0.1/
multichain-1.0.1/multichain-util
multichain-1.0.1/multichain-cli
multichain-1.0.1/README.txt
multichain-1.0.1/multichainind
multichain-1.0.1/multichainind-cold
elzorro@ubuntu:/tmp$
elzorro@ubuntu:/tmp$ cd multichain-1.0.1
elzorro@ubuntu:/tmp/multichain-1.0.1$
elzorro@ubuntu:/tmp/multichain-1.0.1$ sudo mv multichainind multichain-cli multichain-util /usr/local/bin
[sudo] password for elzorro:
elzorro@ubuntu:/tmp/multichain-1.0.1$ multichain-util create survey

MultiChain 1.0.1 Utilities (protocol 10009)

Blockchain parameter set was successfully generated.
You can edit it in /home/elzorro/.multichain/survey/params.dat before running multichainind for the first time.

To generate blockchain please run "multichainind survey -daemon".
elzorro@ubuntu:/tmp/multichain-1.0.1$ nano ~/.multichain/survey/params.dat
elzorro@ubuntu:/tmp/multichain-1.0.1$
```

### 3) Creating a Blockchain on the Poll Server

On the Poll server we will create a new blockchain named “*Survey*”.

**multichain-util create survey**

### 4) Adjusting Blockchain Settings

Go to the *poll server*, run this command:

**nano ~/.multichain/survey/params.dat**

Go to “ *global permission* “ section change these three parameter:

**anyone-can-connect = true**

**anyone-can-send = true**

**anyone-can-receive = true**

Go to “ *Consensus requirements* “ section change these three parameter:

**setup-first-blocks = 10000**

## 5) Initialize the Blockchain

Again go to poll server, run this command:

**multichaind survey –daemon**

The server will start and you will see the node address that others can use to connect with this string as shown below.

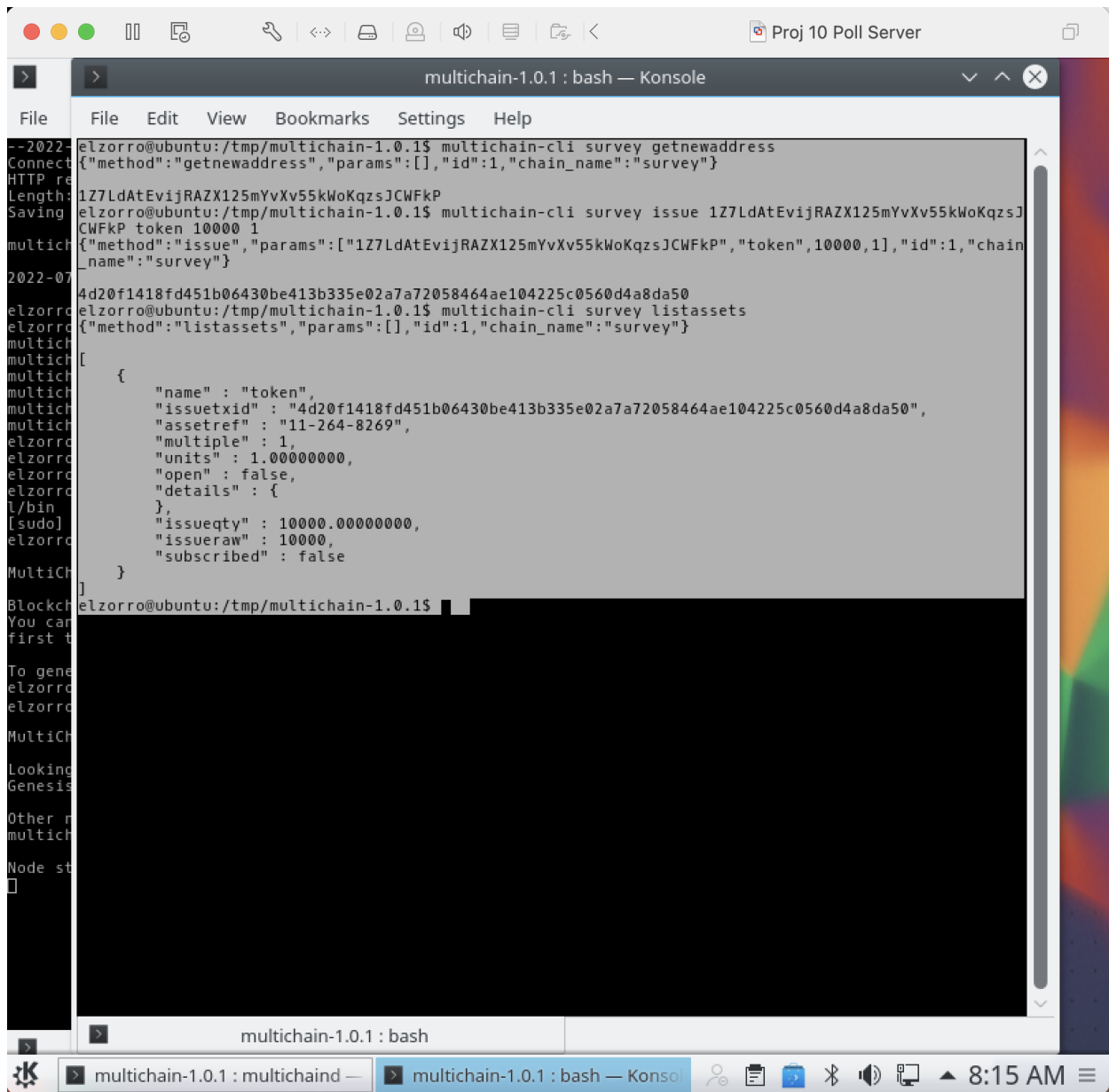
*Note: If you didn't get a prompt after launching Node as shown in the image below, please open another console to continue with the query server commands. Minimize and do not damage the original console.*

## 6) Getting an Address on the Poll Server

Go to the *poll server*, run this command:

**multichain-cli survey getnewaddress**

*Make an note of address appears as result from above command*



```
elzorro@ubuntu:/tmp/multichain-1.0.1$ multichain-cli survey getnewaddress
{"method": "getnewaddress", "params": [], "id": 1, "chain_name": "survey"}
1Z7LdAtEviJRAZX125mYvXv55kWoKqzsJCWFkP
elzorro@ubuntu:/tmp/multichain-1.0.1$ multichain-cli survey issue 1Z7LdAtEviJRAZX125mYvXv55kWoKqzsJCWFkP token 10000 1
{"method": "issue", "params": ["1Z7LdAtEviJRAZX125mYvXv55kWoKqzsJCWFkP", "token", 10000, 1], "id": 1, "chain_name": "survey"}
4d20f1418fd451b06430be413b335e02a7a72058464ae104225c0560d4a8da50
elzorro@ubuntu:/tmp/multichain-1.0.1$ multichain-cli survey listassets
{"method": "listassets", "params": [], "id": 1, "chain_name": "survey"}
[
  {
    "name": "token",
    "issuetxid": "4d20f1418fd451b06430be413b335e02a7a72058464ae104225c0560d4a8da50",
    "assetref": "11-264-8269",
    "multiple": 1,
    "units": 1.00000000,
    "open": false,
    "details": {
    },
    "issueqty": 10000.00000000,
    "issueraw": 10000,
    "subscribed": false
  }
]
```

## 7) Issue the "token" Asset

Now we'll build a non-subdivisible asset called "token" and place 10,000 of them on the Poll server.

**multichain-cli survey issue (*Address note before*) token 10000 1**

**multichain-cli survey listassets**

## 8) Install Apache on the Poll Server

Go to the *poll server*, and enter password run this command :

**sudo apt-get update**

You can skip the command in red if your poll server has apache already installed.

**dpkg --configure -a**

```
sudo rm /var/cache/apt/archives/lock
```

```
sudo rm /var/lib/dpkg/lock
```

```
sudo apt-get install apache2
```

```
elzorro : bash — Konsole <2>
File Edit View Bookmarks Settings Help

--2022-07-14 12:20:07.127Ld Saving multichain ("met _name 4d20f elzorro elzorro elzorro multichain multichain multichain multichain multichain elzorro elzorro elzorro l/bin [sudo] elzorro Multichain Blockchain You can first To gene elzorro elzorro elzorro MultiCh lib Looking Genesis Other multichain Node st [ Do yo Get:1 Get:2 Get:3 lld1 Get:4

Enabling module mime.
Enabling module negotiation.
Enabling module setenvif.
Enabling module filter.
Enabling module deflate.
Enabling module status.
Enabling conf charset.
Enabling conf localized-error-pages.
Enabling conf other-vhosts-access-log.
Enabling conf security.
Enabling conf serve-cgi-bin.
Enabling site 000-default.
Processing triggers for libc-bin (2.23-0ubuntu10) ...
Processing triggers for systemd (229-4ubuntu21.2) ...
Processing triggers for ureadahead (0.100.0-19) ...
Processing triggers for ufw (0.35-0ubuntu2) ...
elzorro@ubuntu:~$ sudo service apache2 start
elzorro@ubuntu:~$ sudo apt-get install php libapache2-mod-php php-mcrypt php-mysql -y
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following additional packages will be installed:
  libapache2-mod-php7.0 libmcrypt4 php-common php7.0 php7.0-cli php7.0-common php7.0-json
  php7.0-mcrypt php7.0-mysql php7.0-opcache php7.0-readline
Suggested packages:
  php-pear libmcrypt-dev mcrypt
The following NEW packages will be installed:
  libapache2-mod-php libapache2-mod-php7.0 libmcrypt4 php php-common php-mcrypt php-mysql php7.0
  php7.0-cli php7.0-common php7.0-json php7.0-mcrypt php7.0-mysql php7.0-opcache php7.0-readline
0 upgraded, 15 newly installed, 0 to remove and 545 not upgraded.
Need to get 3,684 kB of archives.
After this operation, 14.7 MB of additional disk space will be used.
Get:1 http://us.archive.ubuntu.com/ubuntu xenial-updates/main amd64 php-common all 1:35ubuntu6.1 [1
0.8 kB]
Get:2 http://us.archive.ubuntu.com/ubuntu xenial-updates/main amd64 php7.0-common amd64 7.0.33-0ubu
ntu0.16.04.16 [845 kB]
Get:3 http://us.archive.ubuntu.com/ubuntu xenial-updates/main amd64 php7.0-json amd64 7.0.33-0ubu
ntu0.16.04.16 [16.9 kB]
Get:4 http://us.archive.ubuntu.com/ubuntu xenial-updates/main amd64 php7.0-opcache amd64 7.0.33-0ub
untu0.16.04.16 [77.2 kB]
Get:5 http://us.archive.ubuntu.com/ubuntu xenial-updates/main amd64 php7.0-readline amd64 7.0.33-0u
buntu0.16.04.16 [12.8 kB]
Get:6 http://us.archive.ubuntu.com/ubuntu xenial-updates/main amd64 php7.0-cli amd64 7.0.33-0ubuntu
0.16.04.16 [1,283 kB]
Get:7 http://us.archive.ubuntu.com/ubuntu xenial-updates/main amd64 libapache2-mod-php7.0 amd64 7.0
.33-0ubuntu0.16.04.16 [1,225 kB]
Get:8 http://us.archive.ubuntu.com/ubuntu xenial-updates/main amd64 libapache2-mod-php all 1:7.0+35
ubuntu6.1 [2,990 B]
Get:9 http://us.archive.ubuntu.com/ubuntu xenial/universe amd64 libmcrypt4 amd64 2.5.8-3.3 [63.0 kB]
Get:10 http://us.archive.ubuntu.com/ubuntu xenial-updates/main amd64 php7.0 all 7.0.33-0ubuntu0.16.
04.16 [1,300 B]
Get:11 http://us.archive.ubuntu.com/ubuntu xenial-updates/main amd64 php all 1:7.0+35ubuntu6.1 [2,8
```

Then run the below command to run apache server.

```
sudo service apache2 start
```

## 9) Install PHP on the Poll Server

Go to the *poll server*, run this command:

```
sudo apt-get install php libapache2-mod-php php-mcrypt php-mysql -y
```

## 10) Install Curl on the Poll Server

Go to the *poll server*, run this command:

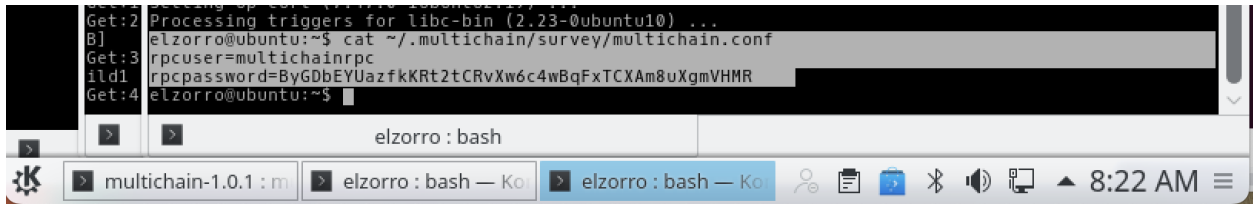
```
sudo apt install curl -y
```

A screenshot of a terminal window titled "elzorro : bash — Konsole <2>". The terminal displays the output of running `dpkg-query -f='\${Package} \${Version} \${Architecture}\n'` which lists installed packages like libapache2-mod-php, libmcrypt4, php7.0, and php7.0-mcrypt. It also shows the command `sudo apt install curl -y` being executed, followed by its progress output indicating that curl will be installed along with libcurl3-gnutls. The terminal interface includes a menu bar at the top with options like File, Edit, View, Bookmarks, Settings, and Help. At the bottom, there are tabs for different terminal sessions, including "multichain-1.0.1 : m...", "elzorro : bash — Ko...", and "elzorro : bash — Ko...". The system clock in the bottom right corner shows 8:21 AM.

### 11) Find your RPC Credentials

Go to the *poll server*, run this command:

**cat ~/.multichain/survey/multichain.conf**



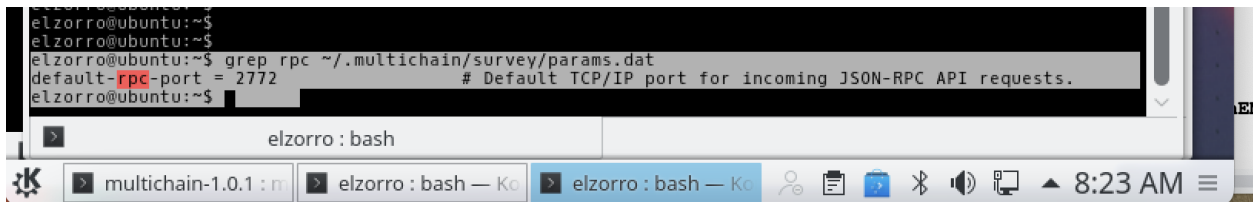
```
Get:2 Processing triggers for libc-bin (2.23-0ubuntu10) ...
B] elzorro@ubuntu:~$ cat ~/.multichain/survey/multichain.conf
Get:3 rpcuser=multichainrpc
ild1 rpcpassword=ByGDbEYUazfkKrt2tCRvXw6c4wBqFXTcXAm8uXgmVHMR
Get:4 elzorro@ubuntu:~$
```

After executing the above command you will get username & password for RPC.

### 12) Find your RPC Port

Go to the *poll server*, run this command:

**grep rpc ~/.multichain/survey/params.dat**



```
elzorro@ubuntu:~$
elzorro@ubuntu:~$
elzorro@ubuntu:~$ grep rpc ~/.multichain/survey/params.dat
default-rpc-port = 2772 # Default TCP/IP port for incoming JSON-RPC API requests.
elzorro@ubuntu:~$
```

### 13) Make the info.php Script

Go to the *poll server*, run this command:

**sudo nano /var/www/html/info.php**

Enter or paste in this code. You will need to change two items:

1. Replace the password in the line beginning with "\$a" with the correct password on your Poll Server
2. Replace the port number at the end of the line beginning with "\$c" with the actual port number on your server.

```
<?php
echo "<h1>Information About the Survey Blockchain</h1>";

$a = 'curl -s --user
multichainrpc:8EC2qNiyoyAVj6dQE6Sf5VuyVJF2zSewrsixnEMdoh7t
--data-binary \'';
$b = '{"jsonrpc": "1.0", "id": "", "method": "getinfo", "params": [';
$c = ']' }\' -H "content-type: text/plain;" http://127.0.0.1:9706/;
$cmd = $a . $b . $c;
```



```
echo "\n<h2>Raw Output</h2><pre>\n";  
$ret=system($cmd);
```

```
echo "\n<h2>Decoded Output</h2>\n";  
$rets = json_decode($ret, true);  
print_r($rets);
```

```
echo "\n<h2>Single-Item Output</h2>\n";  
echo $rets['result']['version'];  
?>
```

#### 14) Testing the info.php Script

```
php /var/www/html/info.php
```

```
Proj 10 Poll Server
elzorro : bash — Konsole <2>
File Edit View Bookmarks Settings Help
elzorro@ubuntu:~$ php /var/www/html/info.php
Could not open input file: /var/www/html/info.php
elzorro@ubuntu:~$ php info.php

<h1>Information About the Survey Blockchain</h1>
<h2>Raw Output</h2><pre>
{"result":{"version":"1.0.1","nodeversion":10001901,"protocolversion":10009,"chainname":"survey","description":
"MultiChain survey","protocol":"multichain","port":2773,"setupblocks":10000,"nodeaddress":"survey@192.168.26.13
2:2773","burnaddress":"1XXXXXXXXXh8XXXXXXXXtjXXXXXXXXcUHxaA","incomingpaused":false,"miningpaused":false,"wal
letversion":60000,"balance":0.00000000,"walletdbversion":2,"reindex":false,"blocks":72,"timeoffset":0,"connecti
ons":0,"proxy":"","difficulty":0.00000006,"testnet":false,"keypoololdest":1657811564,"keypoolsize":2,"paytxfee"
:0.00000000,"relayfee":0.00000000,"errors":""},"error":null,"id":""}

<h2>Decoded Output</h2>
Array
(
    [result] => Array
        (
            [version] => 1.0.1
            [nodeversion] => 10001901
            [protocolversion] => 10009
            [chainname] => survey
            [description] => MultiChain survey
            [protocol] => multichain
            [port] => 2773
            [setupblocks] => 10000
            [nodeaddress] => survey@192.168.26.132:2773
            [burnaddress] => 1XXXXXXXXXh8XXXXXXXXtjXXXXXXXXcUHxaA
            [incomingpaused] =>
            [miningpaused] =>
            [walletversion] => 60000
            [balance] => 0
            [walletdbversion] => 2
            [reindex] =>
            [blocks] => 72
            [timeoffset] => 0
            [connections] => 0
            [proxy] =>
            [difficulty] => 6.0E-8
            [testnet] =>
            [keypoololdest] => 1657811564
            [keypoolsize] => 2
            [paytxfee] => 0
            [relayfee] => 0
            [errors] =>
        )
    [error] =>
    [id] =>
)

<h2>Single-Item Output</h2>
1.0.1elzorro@ubuntu:~$
```

## 15) Make a pay.php Script

You may create a script that pays out a token now that you know how to operate the Multichain from PHP. The script will deliver a single token to a given address.

Go to the *poll server*, run this command:

```
sudo nano /var/www/html/pay.php
```

Enter or paste in this code. You will need to change two items:

1. Replace the password in the line beginning with "\$a" with the correct password on your Poll Server
2. Replace the port number at the end of the line beginning with "\$d" with the actual port number on your server.

```
<?php
```

```
echo "<h1>Sending you a Survey Token!</h1>";  
$addr = $_POST["address"];
```

```
$a = 'curl -s --user  
multichainrpc:8EC2qNiyoyAVj6dQE6Sf5VuyVJF2zSewrsixnEMdoh7t  
--data-binary \'  
$b = '{"jsonrpc": "1.0", "id": "", "method": "sendassettoaddress",  
"params": [  
$c = ', "token", 1';  
$d = ']' }\' -H "content-type: text/plain;" http://127.0.0.1:9706/;  
$cmd = $a . $b . $addr . $c . $d;
```

```
$ret=system($cmd);  
?>
```

```
elzorro : sudo — Konsole
GNU nano 2.5.3 File: /var/www/html/pay.php

?php
echo "<h1>Sending you a Survey Token!</h1>";
$addr = $_POST["address"];

$a = 'curl -s --user multichainrpc:ByGDbEYUazfkKRT2tCRvXw6c4wBqFxTCXAm8uXgmVHMR --data-binary \'';
$b = '{"jsonrpc": "1.0", "id": "", "method": "sendassettoaddress", "params": [';
$c = '"', "token", 1';
$d = ']' }\' -H "content-type: text/plain;" http://127.0.0.1:2772/';
$cmd = $a . $b . $addr . $c . $d;

$ret=system($cmd);
?>
```

## 16) Make a faucet.htm Page

Go to the *poll server*, run this command:

```
sudo nano /var/www/html/faucet.htm
```

Enter or paste in this code.

```
<html><head><title>Survey Token Faucet</title></head>
<body bgcolor="#cccccc">
<h1 align="center">Survey Token Faucet</h1>
```

```
<form method="post" action="pay.php">
<p align="center"><b>Enter your address</b></p>
<p align="center"><input type="text" name="address"
size="90"></textare></p>
<p align="center">
<button type="submit" name="submitButton" value="">Get Token</button>
</form>

</body></html>
```

### 17) Removing Old Multichains on the Voter

Go to the *voter server*, run this command:

```
ps aux | grep multi
```

As a result if any Multichain daemon is running you will see on the right hand side with the command line argument. Note down the id number for the next step.

For each and every multichain process execute this command to kill it:

```
kill 19066( change process as per yours)
```

Execute this command to remove the old multichain data form the server:

```
rm -rf ~/. multichain
```

### 18) Install Multichain on the Voter

Go to the *voter server*, run this command:

```
cd /tmp
wget http://www.multichain.com/download/multichain-1.0.1.tar.gz
tar -xvzf multichain-1.0.1.tar.gz
cd multichain-1.0.1
sudo mv multichaind multichain-cli multichain-util /usr/local/bin
```

### 19) Finding the IP Address of your Poll Server

Go to the *poll server*, run this command:

```
ifconfig
```

As shown below, find the IP address of your Ethernet interface.

```
[protocol] => multichain
[port] => 2773
[setupblocks] => 10000
[nodeaddress] => survey@192.168.26.132:2773
[burnaddress] => 1XXXXXXXXWzXXXXXXXXh8XXXXXXXXTjXXXXXXXXcUHxaA
[incomingpaused] =>
[miningpaused] =>
[walletversion] => 60000
[balance] => 0
[walletdbversion] => 2
[reindex] =>
[blocks] => 72
[timeoffset] => 0
[connections] => 0
[proxy] =>
[difficulty] => 6.0E-8
[testnet] =>
[keypoololdest] => 1657811564
[keypoolsize] => 2
[paytxfee] => 0
[relayfee] => 0
[errors] =>

)

[error] =>
[id] =>

<h2>Single-Item Output</h2>
1.0.1elzorro@ubuntu:~$ sudo nano /var/www/html/pay.php
elzorro@ubuntu:~$ sudo nano /var/www/html/faucet.htm
elzorro@ubuntu:~$ sudo nano /var/www/html/pay.php
elzorro@ubuntu:~$ sudo nano /var/www/html/pay.php
elzorro@ubuntu:~$ ifconfig
ens33
    Link encap:Ethernet  HWaddr 00:0c:29:5c:3c:7f
    inet addr:192.168.26.132  Bcast:192.168.26.255  Mask:255.255.255.0
    inet6 addr: fe80::836c:6c18:d614:f022/64  Scope:Link
    UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
    RX packets:20769 errors:0 dropped:0 overruns:0 frame:0
    TX packets:4601 errors:0 dropped:0 overruns:0 carrier:0
    collisions:0 txqueuelen:1000
    RX bytes:30690731 (30.6 MB)  TX bytes:305441 (305.4 KB)

    Link encap:Local Loopback
    inet addr:127.0.0.1  Mask:255.0.0.0
    inet6 addr: ::1/128  Scope:Host
    UP LOOPBACK RUNNING  MTU:65536  Metric:1
    RX packets:19087 errors:0 dropped:0 overruns:0 frame:0
    TX packets:19087 errors:0 dropped:0 overruns:0 carrier:0
    collisions:0 txqueuelen:1000
    RX bytes:1557734 (1.5 MB)  TX bytes:1557734 (1.5 MB)

elzorro@ubuntu:~$
```

## 20) Finding the Network Port

Go to the *poll server*, run this command:

```
grep network-port ~/.multichain/survey/params.dat
```

Find the port number, as shown below.

## 21) Connecting the Voter to the Blockchain

Go to the *poll server*, replacing the node address and port number with the correct values for your Poll server, which you found just one step above.

```
multichaind survey@172.16.1.135:9707 -daemon
```

## 22) Getting an Address

Go to the *poll server*, run this command:

**multichain-cli survey getnewaddress**

*Make an note of address appears as result from above command.*

## 23) Using the Faucet

Go to the *poll server or voter server*, run this command:

**172.16.1.135/faucet.htm**

Copy and paste your address from the Voter's page onto the Faucet page, as seen below.

Click the "**Get Token**" button.

The response shows "**error**": **null**, as shown below--it worked!

## 24) Checking your Balance

Go to the *voter server*, run this command:

**multichain-cli survey gettotalbalances**

```
multi-chain-1.0.1 : bash — Konsole
File Edit View Bookmarks Settings Help

MultiChain server starting
Retrieving blockchain parameters from the seed node 172.16.1.135:2773 ...
Error: Couldn't connect to the seed node 172.16.1.135 on port 2773 - please check multi-chain-1.0.1 is running at that address and that your firewall settings allow incoming connections.
elzorro@ubuntu:/tmp/multi-chain-1.0.1$
elzorro@ubuntu:/tmp/multi-chain-1.0.1$
elzorro@ubuntu:/tmp/multi-chain-1.0.1$
elzorro@ubuntu:/tmp/multi-chain-1.0.1$
elzorro@ubuntu:/tmp/multi-chain-1.0.1$ multi-chain-cli survey getnewaddress
error: No credentials found for chain "survey"

You must set rpcpassword=<password> in the configuration file:
/home/elzorro/.multi-chain/survey/multi-chain.conf
If the file does not exist, create it with owner-readable-only file permissions.
elzorro@ubuntu:/tmp/multi-chain-1.0.1$ multi-chain survey@192.168.26.132:2773 -daemon

MultiChain 1.0.1 Daemon (protocol 10009)

MultiChain server starting
Retrieving blockchain parameters from the seed node 192.168.26.132:2773 ...
Other nodes can connect to this node using:
multi-chain survey@192.168.26.131:2773

Node started
elzorro@ubuntu:/tmp/multi-chain-1.0.1$ multi-chain-cli survey getnewaddress
{"method": "getnewaddress", "params": [], "id": 1, "chain_name": "survey"}
1TSrQt5fYymLAoaz8p5KbRvMKbn9LCsDZ8U93W
elzorro@ubuntu:/tmp/multi-chain-1.0.1$ 192.168.26.132/faucet.htm
bash: 192.168.26.132/faucet.htm: No such file or directory
elzorro@ubuntu:/tmp/multi-chain-1.0.1$ 192.168.26.131/faucet.htm
bash: 192.168.26.131/faucet.htm: No such file or directory
elzorro@ubuntu:/tmp/multi-chain-1.0.1$ multi-chain-cli survey gettotalbalances
{"method": "gettotalbalances", "params": [], "id": 1, "chain_name": "survey"}
[
  {
    "name" : "token",
    "assetref" : "11-264-8269",
    "qty" : 1.00000000
  }
]
elzorro@ubuntu:/tmp/multi-chain-1.0.1$
```

## 25) Getting an Account for Candidate #1

Go to the *poll server*, run this command:

**multi-chain-cli survey getnewaddress**

**multi-chain-cli survey getnewaddress**



```
elzorro@ubuntu:~$  
Tfelzorro@ubuntu:~$  
elzorro@ubuntu:~$  
elzorro@ubuntu:~$ multichain-cli survey getnewaddress  
{"method": "getnewaddress", "params": [], "id": 1, "chain_name": "survey"}  
a  
11Q2JsiRnggESg7WfYn4YtFQdweTskKqBtXdwu  
elzorro@ubuntu:~$  
elzorro@ubuntu:~$  
elzorro@ubuntu:~$  
elzorro@ubuntu:~$  
elzorro@ubuntu:~$  
elzorro@ubuntu:~$  
elzorro@ubuntu:~$  
elzorro@ubuntu:~$  
elzorro@ubuntu:~$  
elzorro@ubuntu:~$ multichain-cli survey getnewaddress  
{"method": "getnewaddress", "params": [], "id": 1, "chain_name": "survey"}  
1FVRcHfu8hewR2ZCSUgKAubQiqVGCfPEikX7be  
elzorro@ubuntu:~$  
elzorro : bash
```

## 26) Making a Voting Page

Go to the *poll server*, run this command:

```
sudo nano /var/www/html/vote.htm
```

Paste in this code. Replace the addresses with the two addresses you just made.

```
<html>
```

```
<head><title>Candidates</title></head>
```

```
<body>
```

```
<h2>Vote for Candidate #1</h2>
```

```
multichain-cli survey sendassettoaddress
```

```
1VRxsr8G1oe9WApAXUEXT6BJ9BEjJeHJhGKTkP token 1
```

```
<h2>Vote for Candidate #2</h2>
```

```
multichain-cli survey sendassettoaddress
```

```
1M3p1VDLSfZZBERkLkERf77vZ24tf1nNhJdmYm token 1
```

```
</body>
```

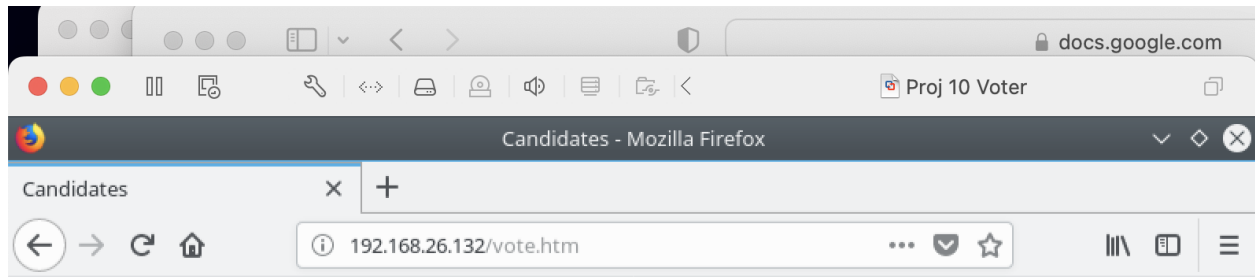
```
</html>
```

```
elzorro : sudo — Konsole
File Edit View Bookmarks Settings Help
GNU nano 2.5.3 File: /var/www/html/vote.htm
<html>
<head><title>Candidates</title></head>
<body>
<h2>Vote for Candidate #1</h2>
multichain-cli survey sendassettoaddress 1Q2Js1RnggE5g7WfYn4YtFQdweTsksKqBtXdwu token 1
<h2>Vote for Candidate #2</h2>
multichain-cli survey sendassettoaddress 1FVRcHfu8hewR2ZCSUgKAubQiqVGCfPEikX7be token 1
</body>
</html>
```

## 27) Voting

Go to the *Web Browser*, run this command:

**172.16.1.135/vote.htm**

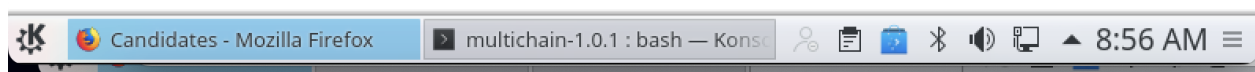


## Vote for Candidate #1

`multichain-cli survey sendassettoaddress 1Q2JsiRnggE5g7WFYn4YtFQdweTsksKqBtXdwu token 1`

## Vote for Candidate #2

`multichain-cli survey sendassettoaddress 1FVRcHfu8hewR2ZCSUgKAubQiqVGCFPEikX7be token 1`



Go to the *voter server*, run this command:

Replacing the values with your candidate 1 and 2 address values, run **ONE** of the commands.

**`multichain-cli survey sendassettoaddress  
1VRxsr8G1oe9WApAXUEXT6BJ9BEjJeHJhGKTkP token 1`**

**OR**

**`multichain-cli survey sendassettoaddress  
1M3p1VDLSfZZBERkLkERf77vZ24tf1nNhJdmYm token 1`**

```
elzorro@ubuntu:/tmp/multichain-1.0.1$
elzorro@ubuntu:/tmp/multichain-1.0.1$
elzorro@ubuntu:/tmp/multichain-1.0.1$ multichain-cli survey getnewaddress
error: No credentials found for chain "survey"

You must set rpcpassword=<password> in the configuration file:
/home/elzorro/.multichain/survey/multichain.conf
If the file does not exist, create it with owner-readable-only file permissions.
elzorro@ubuntu:/tmp/multichain-1.0.1$ multichaind survey@192.168.26.132:2773 -daemon

MultiChain 1.0.1 Daemon (protocol 10009)

MultiChain server starting
Retrieving blockchain parameters from the seed node 192.168.26.132:2773 ...
Other nodes can connect to this node using:
multichaind survey@192.168.26.131:2773

Node started
elzorro@ubuntu:/tmp/multichain-1.0.1$ multichain-cli survey getnewaddress
{"method": "getnewaddress", "params": [], "id": 1, "chain_name": "survey"}

1T5rQt5fYymLAoaz8p5KbRvMKbn9LCsDZ8U93W
elzorro@ubuntu:/tmp/multichain-1.0.1$ 192.168.26.132/faucet.htm
bash: 192.168.26.132/faucet.htm: No such file or directory
elzorro@ubuntu:/tmp/multichain-1.0.1$ 192.168.26.131/faucet.htm
bash: 192.168.26.131/faucet.htm: No such file or directory
elzorro@ubuntu:/tmp/multichain-1.0.1$ multichain-cli survey gettotalbalances
{"method": "gettotalbalances", "params": [], "id": 1, "chain_name": "survey"}

[
  {
    "name" : "token",
    "assetref" : "11-264-8269",
    "qty" : 1.00000000
  }
]
elzorro@ubuntu:/tmp/multichain-1.0.1$
elzorro@ubuntu:/tmp/multichain-1.0.1$
elzorro@ubuntu:/tmp/multichain-1.0.1$
elzorro@ubuntu:/tmp/multichain-1.0.1$
elzorro@ubuntu:/tmp/multichain-1.0.1$ multichain-cli survey sendassettoaddress 1Q2JsiRnggE5g7WFYn4YtFQdweTsksKqBtXdwu token 1
{"method": "sendassettoaddress", "params": ["1Q2JsiRnggE5g7WFYn4YtFQdweTsksKqBtXdwu", "token", 1], "id": 1, "chain_name": "survey"}

915d96bd63a1939be7c6008e86270fe9a27a18590696fb199d0e2d089870e239
elzorro@ubuntu:/tmp/multichain-1.0.1$
```

## 28) Make the results.php Script

Go to the *poll server*, run this command:

**sudo nano /var/www/html/results.php**

Enter or paste in this code. You will need to change four items:

1. Replace the \$scan1 value with the address for Candidate #1
2. Replace the \$scan2 value with the address for Candidate #2
3. Replace the password in the line beginning with "\$a" with the correct password on your Poll Server
4. Replace the port number at the end of the line beginning with "\$d" with the actual port number on your server.

<?php

**\$scan1 = "1VRxsr8G1oe9WApAXUEXT6BJ9BEjJeHJhGKTkP";**

**\$scan2 = "1M3p1VDLSfZZBERkLkERf77vZ24tf1nNhJdmYm";**

**\$a = 'curl -s --user**

**multichainrpc:8EC2qNiyoyAVj6dQE6Sf5VuyVJF2zSewrsixnEMdoh7t**  
**--data-binary \'**

**\$b = '{ "jsonrpc": "1.0", "id": "", "method": "getaddressbalances",**  
**"params": [**

**\$c = ''**

**\$d = ']' }\' -H "content-type: text/plain;" http://127.0.0.1:9706/**

**echo "\n<h2>Candidate #1 Total</h2><pre>\n";**

**\$cmd = \$a . \$b . \$scan1 . \$c . \$d;**

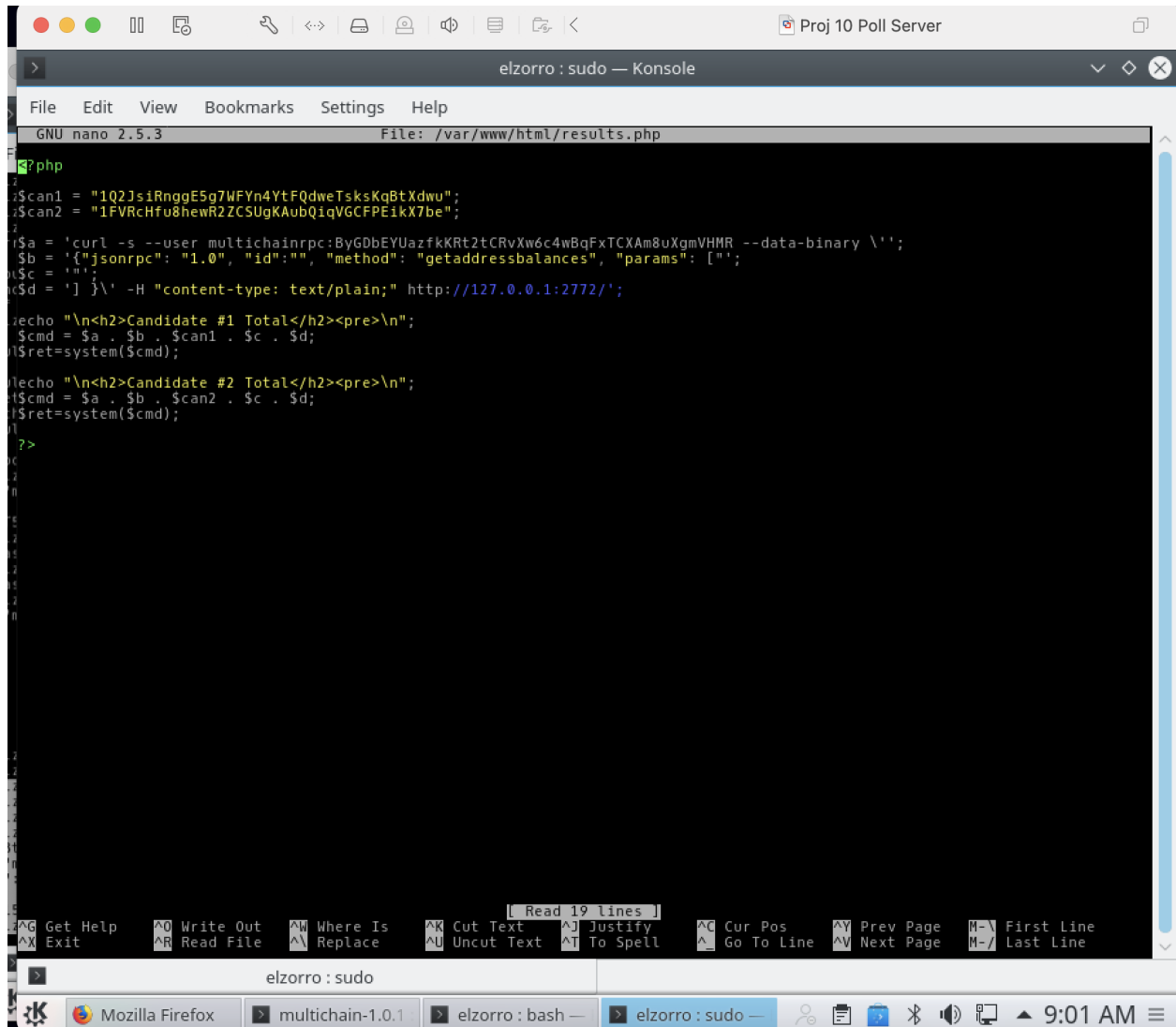
**\$ret=system(\$cmd);**

**echo "\n<h2>Candidate #2 Total</h2><pre>\n";**

**\$cmd = \$a . \$b . \$scan2 . \$c . \$d;**

**\$ret=system(\$cmd);**

**?>**



The screenshot shows a terminal window titled "elzorro : sudo — Konsole". Inside, the GNU nano 2.5.3 editor is open, editing the file /var/www/html/results.php. The script is a PHP file that uses the curl command to interact with a Multichain RPC endpoint. It defines two candidate IDs, \$scan1 and \$scan2, and constructs a JSON-RPC request to get their balances. The script then echoes the results in a formatted HTML-like output. The terminal window has a menu bar with File, Edit, View, Bookmarks, Settings, and Help. The bottom status bar shows various keyboard shortcuts for editing and navigation. The system tray at the bottom indicates the time is 9:01 AM and shows icons for Mozilla Firefox, multichain-1.0.1, and the terminal session.

```
GNU nano 2.5.3 File: /var/www/html/results.php
?php
$scan1 = "1Q2JsiRnggE5g7WfYn4YtFQdweTsksKqBtXdwu";
$scan2 = "1FVRcHfu8hewR2ZCSUgKAubQiqVGCFPEikX7be";

$a = 'curl -s --user multichainrpc:ByGDbEYUazfkRt2tCRvXw6c4wBqFXTXAm8uXgmVHMR --data-binary \'';
$b = '{"jsonrpc": "1.0", "id": "", "method": "getaddressbalances", "params": ["']';
$c = '";
$d = '}]\' -H "content-type: text/plain;" http://127.0.0.1:2772/';

echo "\n<h2>Candidate #1 Total</h2><pre>\n";
$cmd = $a . $b . $scan1 . $c . $d;
$ret=system($cmd);

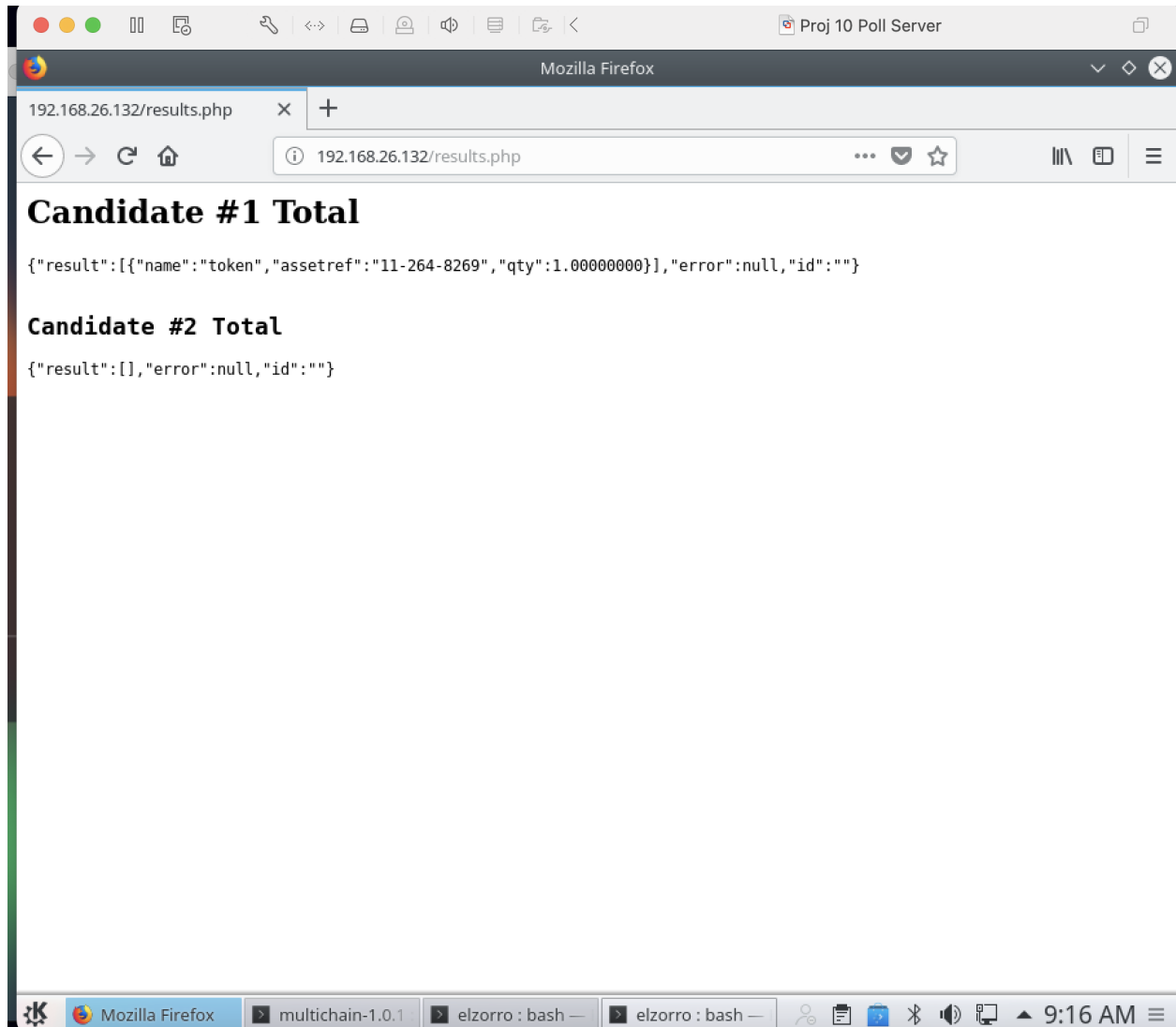
echo "\n<h2>Candidate #2 Total</h2><pre>\n";
$cmd = $a . $b . $scan2 . $c . $d;
$ret=system($cmd);

?>
```

## 29) Viewing the Results

Go to the *Web Browser*, run this command:

**172.16.1.135/results.php**



### 30) Save Screenshot of image showing qty.

**CONCLUSION** : Finally, we demonstrated how to design a voter system that is secure and allows the user to confirm that his vote was accurately delivered. We did this with the help of multichain and php code for producing tokens, which were supposed to be limited to one per user but did not work, allowing users to gain several tokens and vote multiple times.