Advanced usage

- Installing a timelord on Debian / Ubuntu
- Setting up a private RPC server on Debian / Ubuntu

Installing a timelord on Debian / Ubuntu

1. To run a timelord, you need a fully synchronized full node. If you don’t have one, follow all the steps of this tutorial.

2. The timelord software is not included in the official binary **.deb** or **.rpm** releases. You need to build it from the source code. Log in as root and install the **Git** client first.

```
apt-get install git
```

3. The timelord installer script requires root access to install certain dependencies from APT repositories. Add the BPX Chain services user to the **sudoers** with the following command:

```
usermod -a -G sudo bpxchain
```

4. Switch to the BPX Chain services user account:

```
su - bpxchain
```

4. Clone the BPX Beacon Client git repository:

```
git clone https://github.com/bpx-network/bpx-beacon-client
```

5. Install the Beacon Client from the source. This process may take several minutes.

```
cd bpx-beacon-client
. install.sh
```

6. Install the timelord module:

```
. install-timelord.sh
```

7. Press **Control + D** to log out and return to the root console.

8. Create a configuration file for the timelord **systemd** service:

```
nano /etc/systemd/system/bpx-timelord.service
```

Insert the following file content:

```
[Unit]
Description=BPX Timelord

[Service]
Type=forking
User=bpxchain
WorkingDirectory=/home/bpxchain/bpx-beacon-client
ExecStart=bash -c ". activate && bpx start timelord-launcher-only timelord-only"
ExecStop=bash -c ". activate && bpx stop timelord-only timelord-launcher-only"
Restart=always

[Install]
WantedBy=multi-user.target
```

9. Reload the system services configuration.

```
systemctl daemon-reload
```

10. Enable automatic startup of the new service.

```
systemctl enable bpx-timelord
```

Regular Timelord

If you want to run a regular timelord used by BPX consensus algorithm for generating new blocks, simply start the timelord with the default configuration:

```
systemctl start bpx-timelord
```

Bluebox Timelord

To run the bluebox timelord, which compresses old blocks in the chain, you need to edit the configuration file:

```
nano /home/bpxchain/.bpxchain/beacon/config/config.yaml
```

- In the timelord section, set `bluebox_mode` to `True`
- In the beacon section, set `send_uncompact_interval` to recommended value of `300`

Save the file and exit. Then start the timelord service:

```
systemctl start bpx-timelord
```

Setting up a private RPC server on Debian / Ubuntu

1. Set up and sync your BPX Chain full node following this guide. When creating the execution client systemd service, add some new parameters to the **bpx-geth** command line:

```
ExecStart=bpx-geth --syncmode snap --http --http.api web3,eth,net --http.corsdomain "*"
```

- `--http.api web3,eth,net` makes only safe APIs available to the public: `web3`, `eth` and `net`, while blocking access to potentially dangerous RPC methods, such as `admin` or `personal`
- `--http.corsdomain "*"` allows your RPC endpoint to be used in dApps across all domains

If you want your RPC endpoint to provide archive data as well, replace the mentioned line with the following one:

```
ExecStart=bpx-geth --syncmode full --gcmode archive --http --http.api web3,eth,net --
http.corsdomain "*"
```

2. Create an **A** record in your domain's DNS zone and point it to the IP address of your BPX full node.

> **BPX developers recommendation:** All private RPC endpoints should be named `bpx-dataseed`, so if your domain is `yourdomain.com`, please name your RPC server `bpx-dataseed.yourdomain.com`.

3. Install the **nginx** HTTP server and **certbot**:

```
apt-get install nginx python3-certbot-nginx
```

4. Configure nginx as a reverse proxy for **bpx-geth**:

```
nano /etc/nginx/sites-available/bpx-dataseed.yourdomain.com
```

```
upstream geth {
        least_conn;
        server 127.0.0.1:8545;
```

```
}

server {
        server_name bpx-dataseed.yourdomain.com;

        location / {
                proxy_pass http://geth;
        }
}
```

Save file and exit.

5. Enable the new site and restart nginx:

```
ln -s /etc/nginx/sites-available/bpx-dataseed.yourdomain.com /etc/nginx/sites-enabled/bpx-dataseed.yourdomain.com
systemctl restart nginx
```

6. Obtain a free SSL certificate from **Letsencrypt** to enable HTTPS:

```
certbot --nginx -d bpx-dataseed.yourdomain.com
```

During the Certbot wizard, provide your email address for notifications related to the certificate, then select the option to redirect all HTTP traffic to HTTPS.

7. It's done. Test your RPC endpoint by adding `https://bpx-dataseed.yourdomain.com` instead of the public RPC address in your wallet, such as MetaMask.

> Let the BPX developers know that you are hosting your own RPC endpoint. If it operates reliably for a certain period, we will promote it on our website and in other places.