Personal Server on Kali Linux

Follow the steps to setup your own server on the any debian based linux. Here I has use the **kali linux** machine

Install the FocalBoard

Focalboard in Linux is like having a digital whiteboard where you can jot down ideas, make lists, and organize tasks. It's a software tool that helps you keep track of what needs to be done, who's responsible for what, and when tasks are due. It's flexible and customizable, letting you create different boards for different projects or areas of your life. Think of it as your personal or team's command center for staying organized and productive.

```
┌──(root㉿kali)-[~]
└─# wget https://github.com/mattermost/focalboard/releases/download/v0.15.0/focalboard-server-linux-amd64.tar.gz
```

Click here to get above link

```
┌──(root㉿kali)-[~]
└─# tar -xvzf focalboard-server-linux-amd64.tar.gz
```

```
┌──(root㉿kali)-[~]
└─# sudo mv focalboard /opt
```

Install the NGINX

Nginx is a popular open-source web server software. It's widely used for serving web pages, handling reverse proxying, load balancing, and more. In simple terms, Nginx is like the traffic cop of the internet. When you visit a website, Nginx is often the software that receives your request, forwards it to the appropriate destination (like a web application or another server), and then returns the response to your browser. It's known for its high performance, stability, and scalability, making it a go-to choice for many websites and web applications.

```
┌──(root㉿kali)-[~]
└─# apt update
```

```
┌──(root㉿kali)-[~]
└─# apt install nginx
```

Configure NGINX

Create new site :-

```
┌──(root㉿kali)-[~]
└─# nano /etc/nginx/sites-available/focalboard
```

Copy paste this configuration into the above file.

```
upstream focalboard {
  server localhost:8000;
  keepalive 32;
}

server {
  listen 80 default_server;

  server_name focalboard.example.com;

  location ~ /ws/* {
    proxy_set_header Upgrade $http_upgrade;
    proxy_set_header Connection "upgrade";
    client_max_body_size 50M;
    proxy_set_header Host $http_host;
    proxy_set_header X-Real-IP $remote_addr;
    proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;
    proxy_set_header X-Forwarded-Proto $scheme;
    proxy_set_header X-Frame-Options SAMEORIGIN;
    proxy_buffers 256 16k;
    proxy_buffer_size 16k;
    client_body_timeout 60;
    send_timeout 300;
    lingering_timeout 5;
    proxy_connect_timeout 1d;
```

```
    proxy_send_timeout 1d;
    proxy_read_timeout 1d;
    proxy_pass http://focalboard;
  }

  location / {
    client_max_body_size 50M;
    proxy_set_header Connection "";
    proxy_set_header Host $http_host;
    proxy_set_header X-Real-IP $remote_addr;
    proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;
    proxy_set_header X-Forwarded-Proto $scheme;
    proxy_set_header X-Frame-Options SAMEORIGIN;
    proxy_buffers 256 16k;
    proxy_buffer_size 16k;
    proxy_read_timeout 600s;
    proxy_cache_revalidate on;
    proxy_cache_min_uses 2;
    proxy_cache_use_stale timeout;
    proxy_cache_lock on;
    proxy_http_version 1.1;
    proxy_pass http://focalboard;
  }
}
```

(Remember :- insert your own static ip in front of server_name)

If there is a default site, you may need to delete it

```
┌──(root㉿kali)-[~]
└─# rm /etc/nginx/sites-enabled/default
```

Enable the Focalboard site, test the config, and reload NGINX:

```
┌──(root㉿kali)-[~]
└─#  ln -s /etc/nginx/sites-available/focalboard /etc/nginx/sites-enabled/foc
alboard
```

```
┌──(root㉿kali)-[~]
└─# nginx -t
nginx: the configuration file /etc/nginx/nginx.conf syntax is ok
nginx: configuration file /etc/nginx/nginx.conf test is successful
```

```
┌──(root㉿kali)-[~]
└─# systemctl start nginx.service
```

Install Postgresql

Focalboard stores data in a SQLite database by default, but we recommend running against Postgres in production (we've tested against Postgres 10.15). To install, run:

```
┌──(root㉿kali)-[~]
└─# apt install postgresql postgresql-contrib
```

Then run as the postgres user to create a new database:

```
┌──(root㉿kali)-[~]
└─# sudo --login --user postgres
postgres@kali:~$
```

```
postgres@kali:~$ psql
psql (16.1 (Debian 16.1-1+b1))
Type "help" for help.
```

On the psql prompt, run the following commands (**change the user/password** to your own values):

```
postgres=#
postgres=# CREATE DATABASE boards;
CREATE DATABASE
```

```
postgres=# CREATE USER boardsuser WITH PASSWORD 'boardsuser-password';
CREATE ROLE
postgres=#
```

Quite and exit from the prompt

```
postgres=# \q
postgres@kali:~$ exit
logout

┌──(root㉿kali)-[~]
└─#
```

Edit the Focalboard config.json:

```
┌──(root㉿kali)-[~]
└─# nano /opt/focalboard/config.json
```

Change the dbconfig setting to use the postgres database you created:

```
    "dbtype": "postgres",
"dbconfig": "postgres://boardsuser:boardsuser-password@localhost/boards?sslmode=disable&connect_timeout=10",
```

Configure Focalboard to run as a service

This will keep the server running across reboots. First, create a new service config file:

```
┌──(root㉿kali)-[~]
└─# nano /lib/systemd/system/focalboard.service
```

Copy paste the following into the above file:-

```
[Unit]
Description=Focalboard server

[Service]
Type=simple
Restart=always
RestartSec=5s
ExecStart=/opt/focalboard/bin/focalboard-server
WorkingDirectory=/opt/focalboard

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

Make systemd reload the new unit, and start it on machine reboot:

```
┌──(root㉿kali)-[~]
└─# systemctl daemon-reload

┌──(root㉿kali)-[~]
└─# systemctl start focalboard.service

┌──(root㉿kali)-[~]
└─# systemctl enable focalboard.service
```

Test the server

Type your Ip address into the browser.

You will get this page after login to the page