In various restricted environments, individuals often need to use VPN servers. While there are both free and paid VPN providers available, many users remain skeptical about trusting these providers. This guide will walk you through a method to create your personal OpenVPN server using Docker. We'll aim to simplify the process so it can be completed in 5 minutes or less.

Our OpenVPN server will handle multiple user accounts and offer different port options, thanks to Docker's port exporting features. We'll start with UDP port 3000, distinct from the default UDP 1194. We won't use a pre-built image but will create our custom image using a Dockerfile and scripts by kylemanna, licensed under MIT. Let's clone the repository and build our image.

```
# Clone the Git repository onto your server and navigate to the cloned
repository's directory
git clone https://github.com/kylemanna/docker-openvpn.git
cd docker-openvpn/

# Build a new Docker image from these files, naming it 'myownvpn'
docker build -t myownvpn .

# Create a volume or directory to store configuration files and keys
cd ..
mkdir vpn-data && touch vpn-data/vars

# Generate an OpenVPN config file, initialize the PKI, and start the VPN
server
docker run -v $PWD/vpn-data:/etc/openvpn --rm myownvpn ovpn_genconfig -u
udp://IP_ADDRESS:3000
docker run -v $PWD/vpn-data:/etc/openvpn --rm -it myownvpn ovpn_initpki
docker run -v $PWD/vpn-data:/etc/openvpn -d -p 3000:1194/udp --cap-
add=NET_ADMIN myownvpn

# Create a user for connecting to the OpenVPN server and generate a
configuration file
docker run -v $PWD/vpn-data:/etc/openvpn --rm -it myownvpn easyrsa build-
client-full user1 nopass
docker run -v $PWD/vpn-data:/etc/openvpn --rm myownvpn ovpn_getclient user1
> user1.ovpn
```