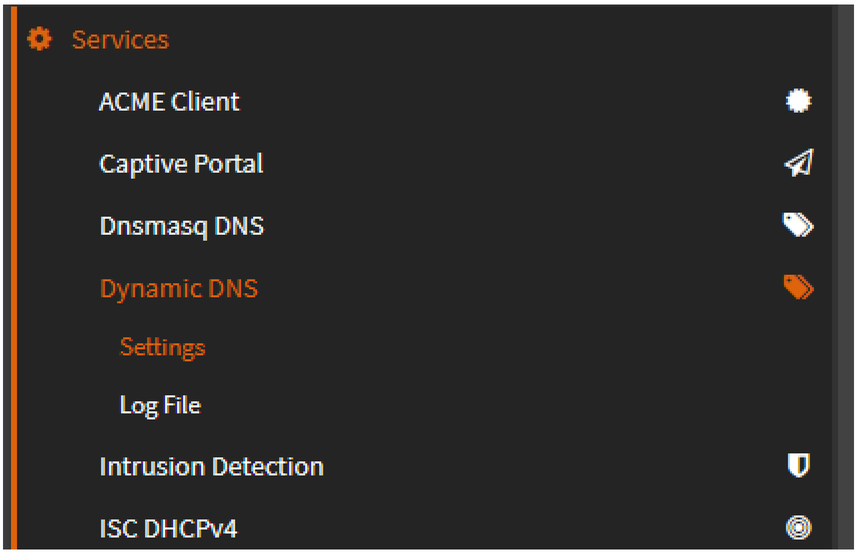
WireGuard VPN

VPN server using Cloudflare DDNS and WireGuard

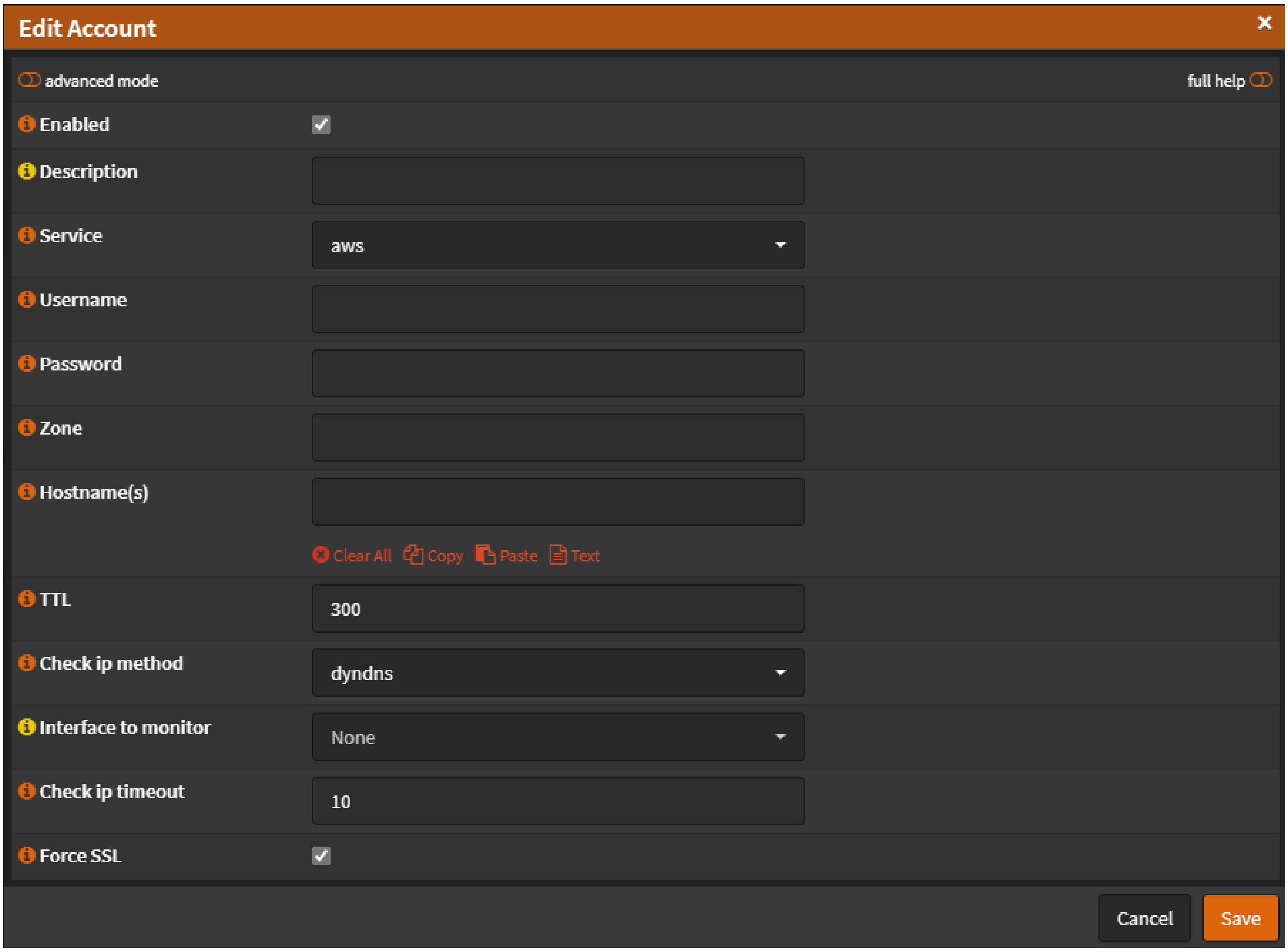
Cloudflare DDNS

If you have a Dynamic WAN IP, you'll need to set up some sort of DDNS client. Most ISPs use Dynamic IPs with residential customers, so this is pretty common and there are multiple options for working around this. I currently manage my domains with Cloudflare, so I'll be using their DDNS so I can have all my management under 1 provider. Setting this up is fairly simple!

First, make sure the os-ddclient plugin is installed on your OPNsense firewall. Once installed, navigate to Services ---> Dynamic DNS ---> Settings:



Next, select the "+" icon to add an account.

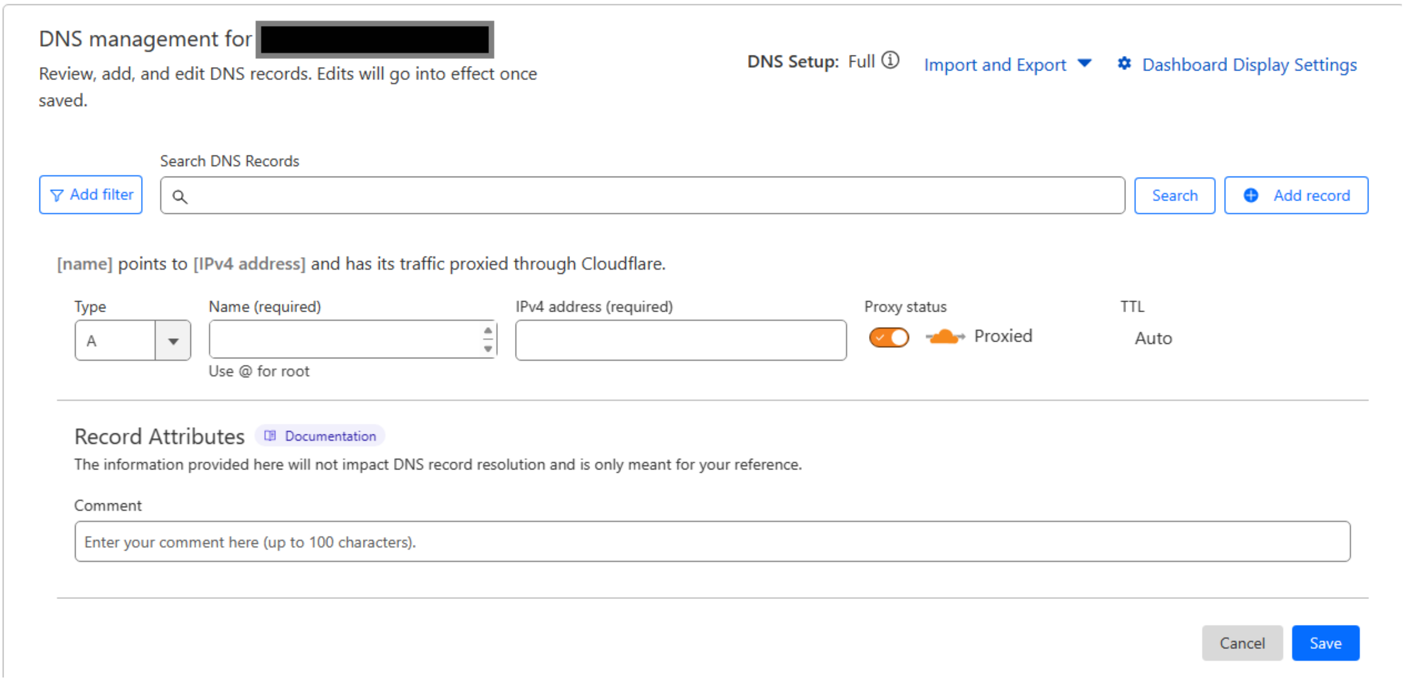


Edit Account

○ advanced mode — full help ○

Enabled	☑
Description	
Service	aws
Username	
Password	
Zone	
Hostname(s)	
TTL	300
Check ip method	dyndns
Interface to monitor	None
Check ip timeout	10
Force SSL	☑

Open up a web browser and create an A Record with your domain registrar for a subdomain. On Clouflare its fairly simple. Navigate to your DNS records, and create a new record:



DNS management for [redacted]

Review, add, and edit DNS records. Edits will go into effect once saved.

DNS Setup: Full

[name] points to [IPv4 address] and has its traffic proxied through Cloudflare.

Type	Name (required)	IPv4 address (required)	Proxy status	TTL
A			Proxied	Auto

Use @ for root

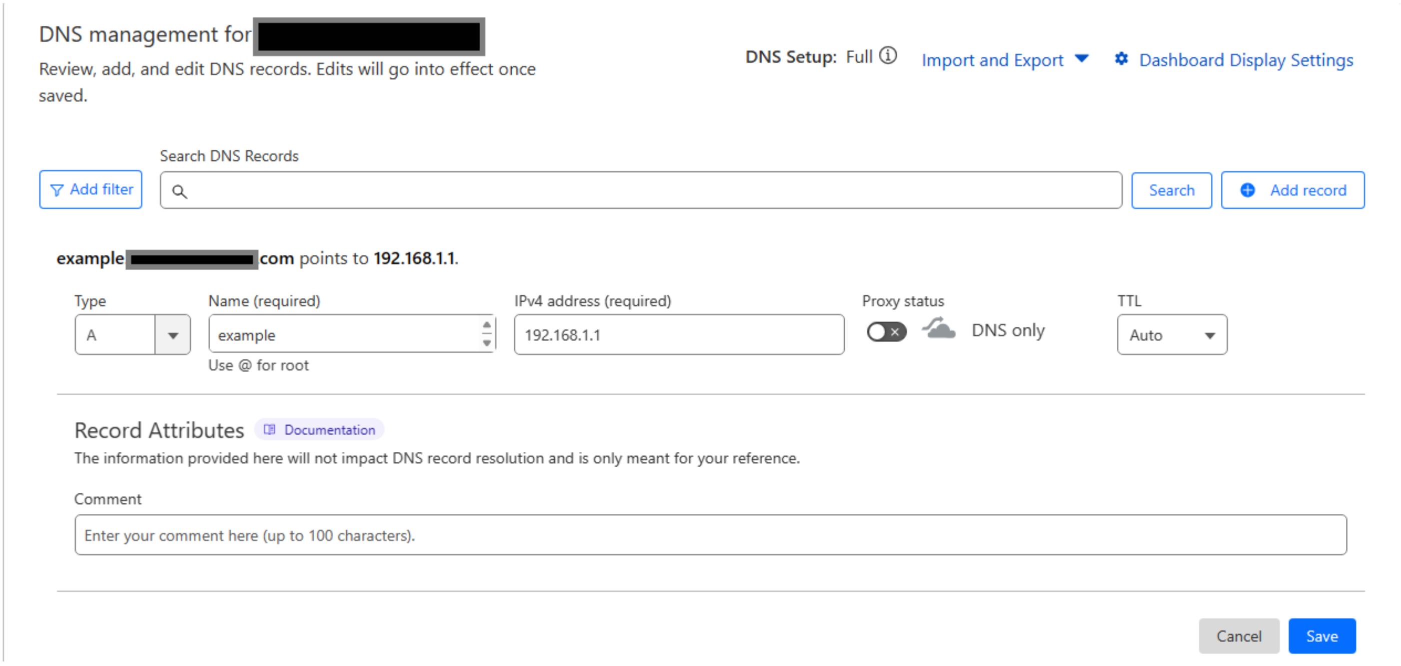
Record Attributes

The information provided here will not impact DNS record resolution and is only meant for your reference.

Comment: Enter your comment here (up to 100 characters).

- Enter a name for your subdomain, and any IP address. The IP you enter doesn't matter as this record will be updated with your WAN IP automatically.
- Make sure you turn Proxy off

Your final settings should look like this:



DNS management for [redacted]

Review, add, and edit DNS records. Edits will go into effect once saved.

DNS Setup: Full ⓘ

example[redacted]**com** points to **192.168.1.1**.

Type: A

Name (required): example

Use @ for root

IPv4 address (required): 192.168.1.1

Proxy status: DNS only

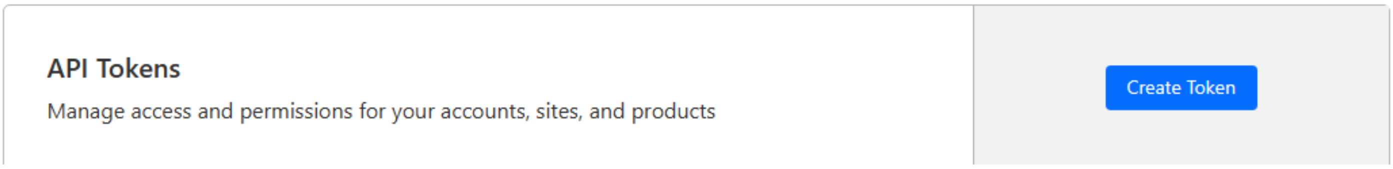
TTL: Auto

Record Attributes

The information provided here will not impact DNS record resolution and is only meant for your reference.

Comment: Enter your comment here (up to 100 characters).

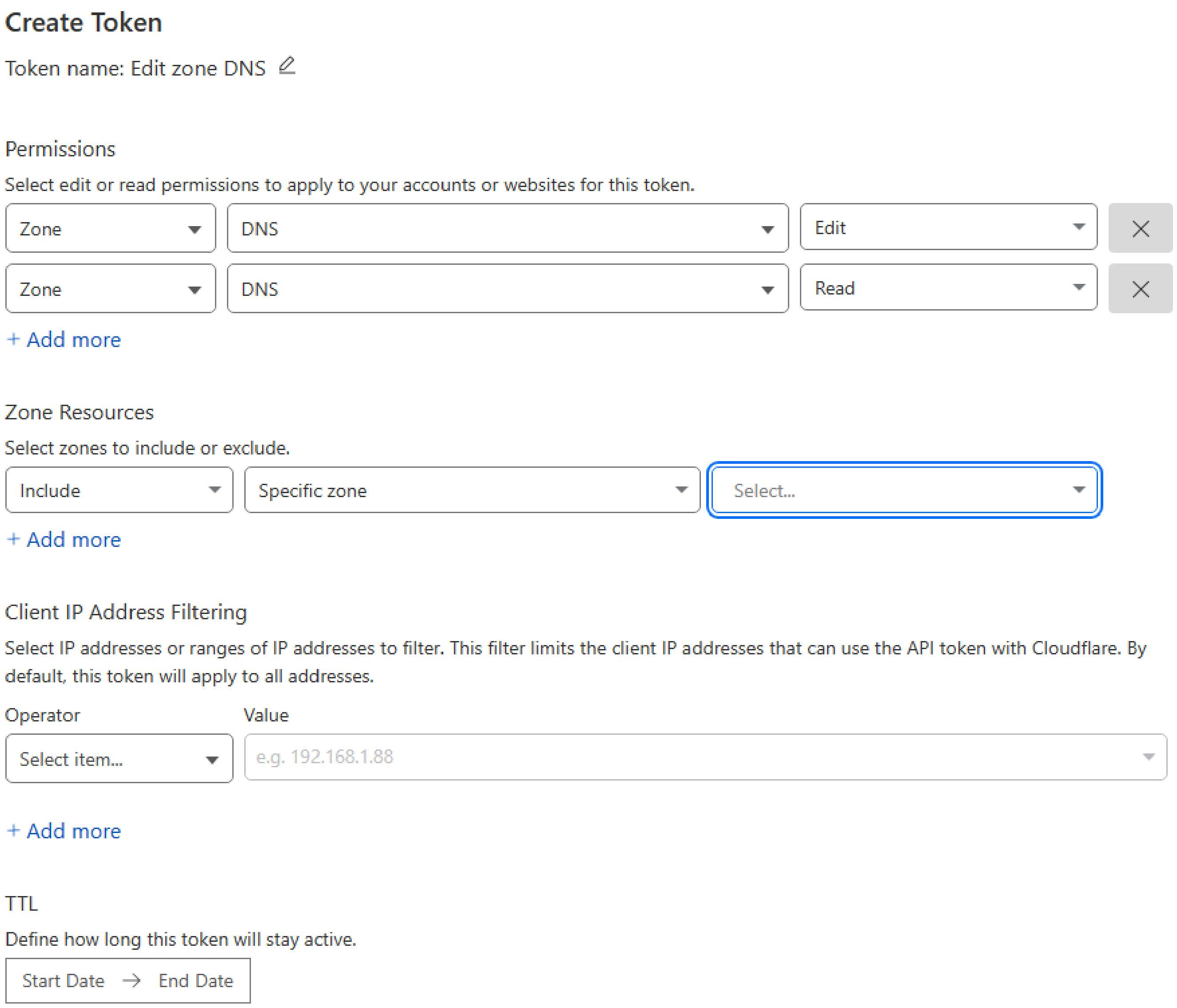
With this record saved, navigate to your API tokens and generate a new API token. Navigate to Overview in Cloudflare, then scroll down and select "Get API token". On the next page, select create token:



API Tokens

Manage access and permissions for your accounts, sites, and products

Use the "Edit zone DNS" template and configure the following:



Create Token

Token name: Edit zone DNS

Permissions

Select edit or read permissions to apply to your accounts or websites for this token.

Zone | DNS | Edit

Zone | DNS | Read

Zone Resources

Select zones to include or exclude.

Include | Specific zone | Select...

Client IP Address Filtering

Select IP addresses or ranges of IP addresses to filter. This filter limits the client IP addresses that can use the API token with Cloudflare. By default, this token will apply to all addresses.

Operator: Select item...

Value: e.g. 192.168.1.88

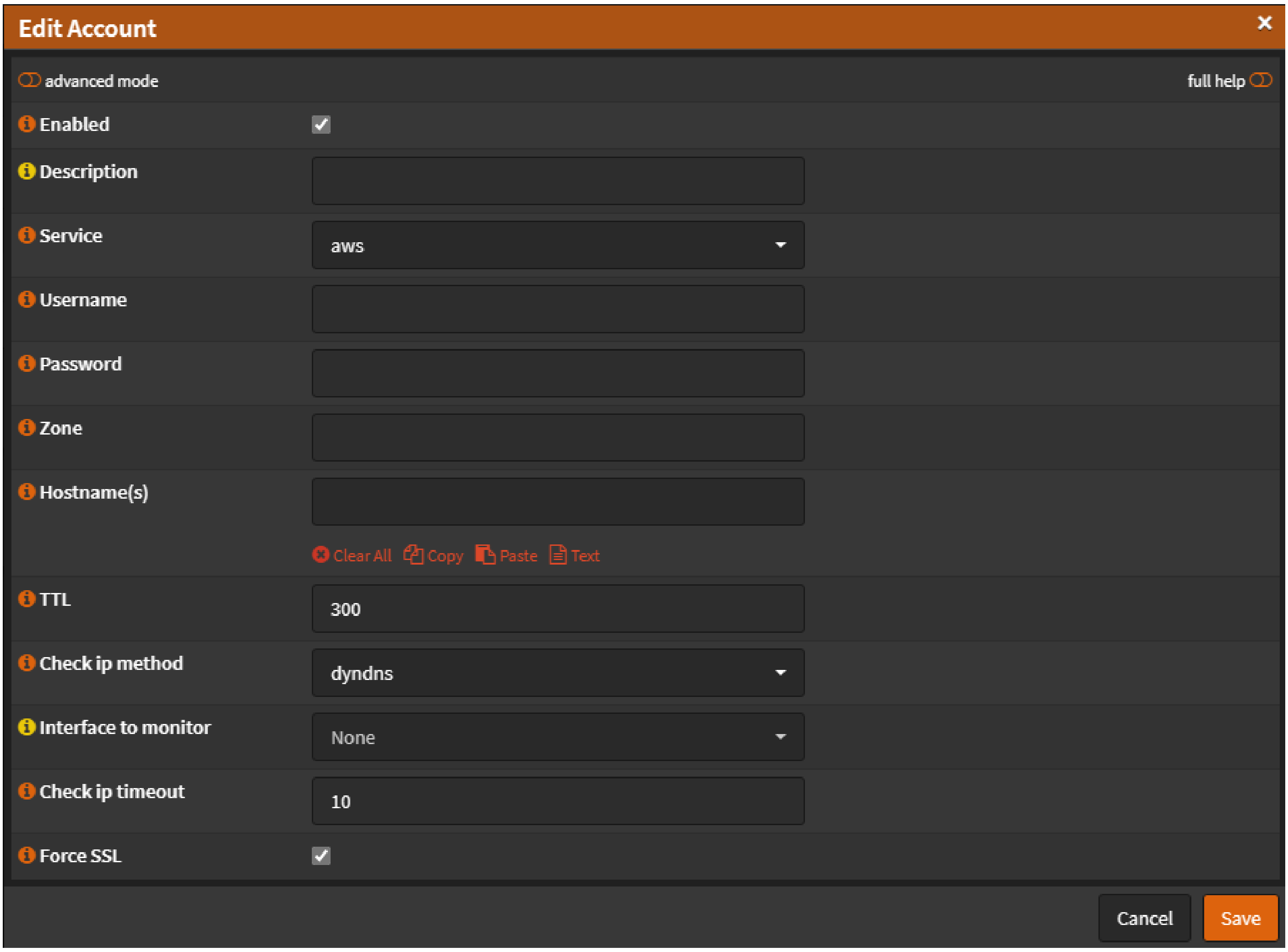
TTL

Define how long this token will stay active.

Start Date → End Date

- Enter a name for the token
- Add another permission as Zone - DNS - Read
- Under zone resrouces configure Include - Specific Zone - Select the domain you have the A Record configured with
- After creating the token, save it somewhere! You will not be able to view this token again!

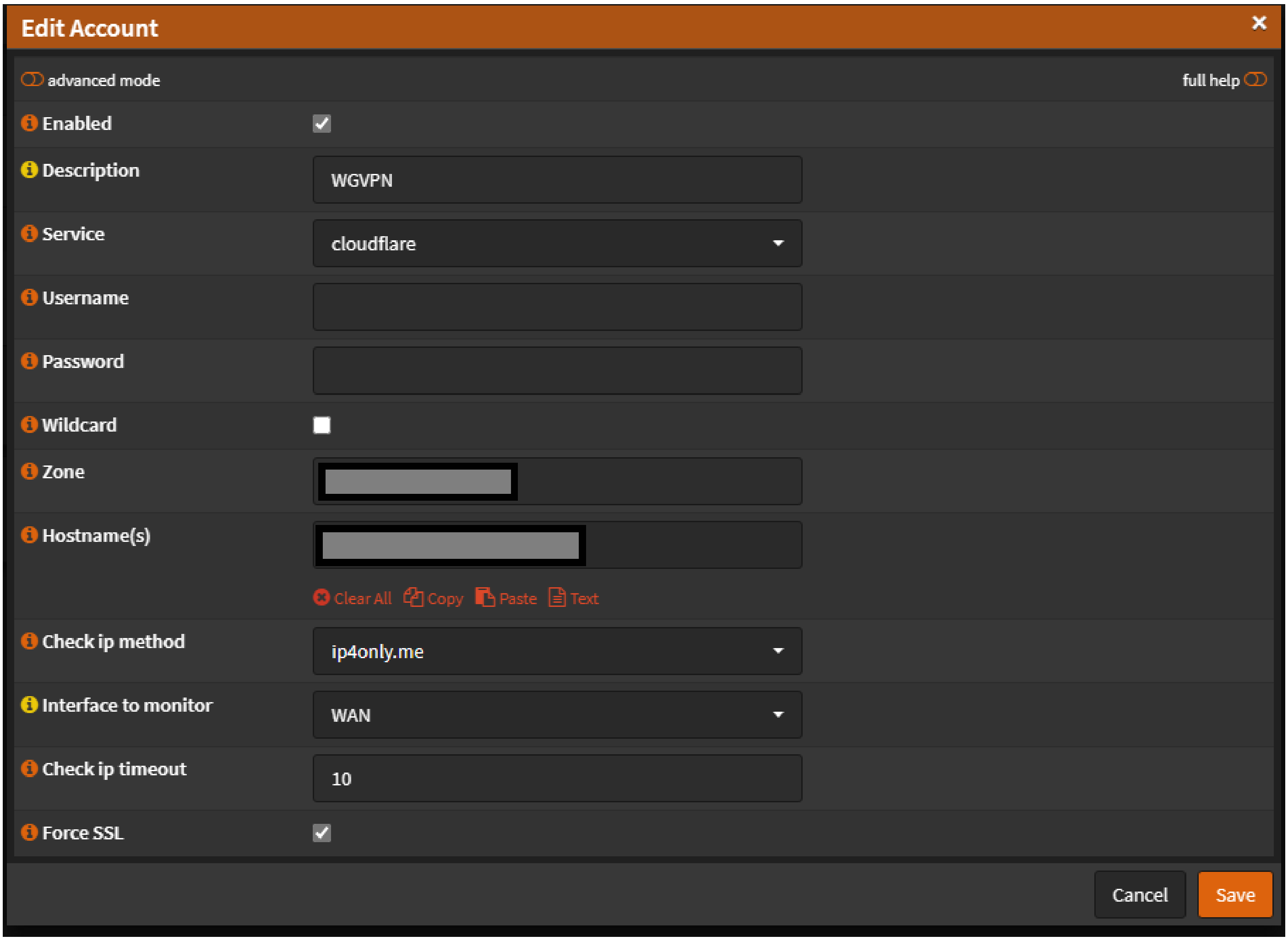
With your A Record configured, and API token in hand, you can now go back to the OPNsense Page:



Edit Account

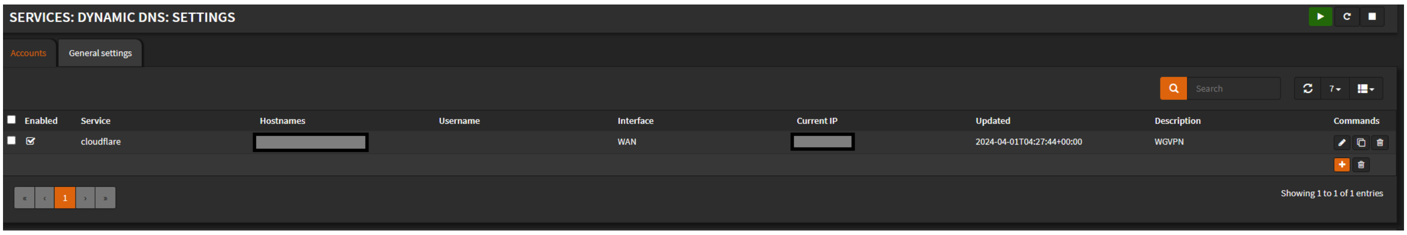
Enabled	☑
Description	
Service	aws
Username	
Password	
Zone	
Hostname(s)	
TTL	300
Check ip method	dyndns
Interface to monitor	None
Check ip timeout	10
Force SSL	☑

- Enable the account
- Give it a Description or name
- Select Cloudflare under Service
- Keep username blank
- Enter your API token as the password
- For zone, enter your domain name
  - example.com
- For Hostname, enter your FQDN
  - vpn.example.com
- For Check IP method, select ip4only.me
- Force SSL, then save configurations



Enabled	☑
Description	WGVPN
Service	cloudflare
Username	
Password	
Wildcard	☐
Zone	[redacted]
Hostname(s)	[redacted]
Check ip method	ip4only.me
Interface to monitor	WAN
Check ip timeout	10
Force SSL	☑

Save your settings and apply the new configurations. Select the refresh icon and your WAN IP should now be updated!



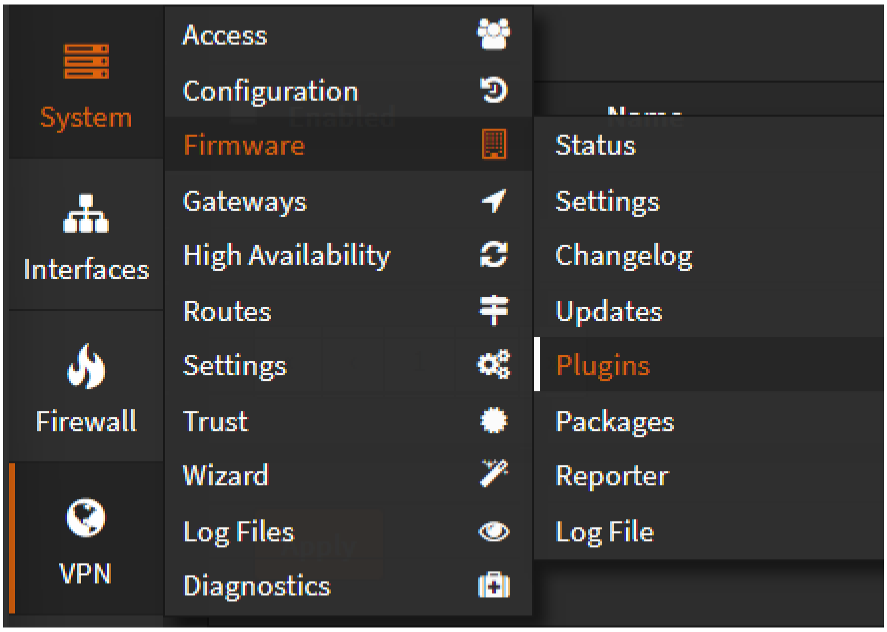
☐	Enabled	Service	Hostnames	Username	Interface	Current IP	Updated	Description	Commands
☐	☑	cloudflare	[redacted]		WAN	[redacted]	2024-04-01T04:27:44+00:00	WGVPN	

Showing 1 to 1 of 1 entries

Check your DNS A Record to see if your WAN IP has updated. It should automatically update. You can now get your WAN IP from this subdomain, as it'll automatically update. To ensure it automatically updates, I've created a cron job in my router to check for changes in my IP every 6 hours and update if necessary.

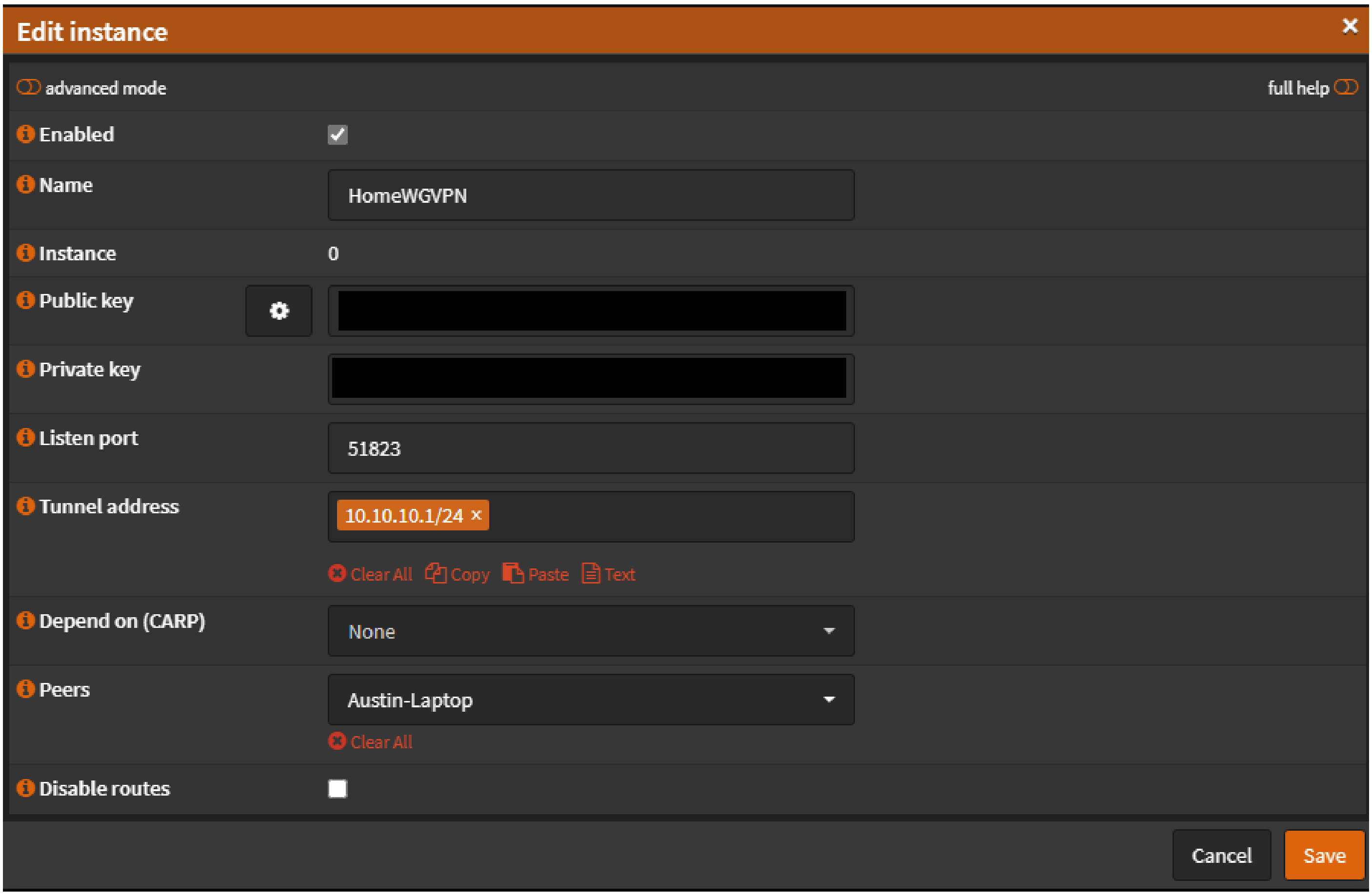
OPNsense Local Configuration

To get started with WireGuard in OPNsense, download & install the plug-in available by naviagting through the Web GUI @ System ---> Firmware ---> Plugins:



Instance/Peer

Next, find Wireguard under the VPN tab in the menu and select WireGuard. Navigate to "Instances" to create and set up an instance. Select the "+" icon and edit your instance:



Edit instance

advanced mode — full help

Enabled ☑

Name: HomeWGVPN

Instance: 0

Public key: [redacted]

Private key: [redacted]

Listen port: 51823

Tunnel address: 10.10.10.1/24

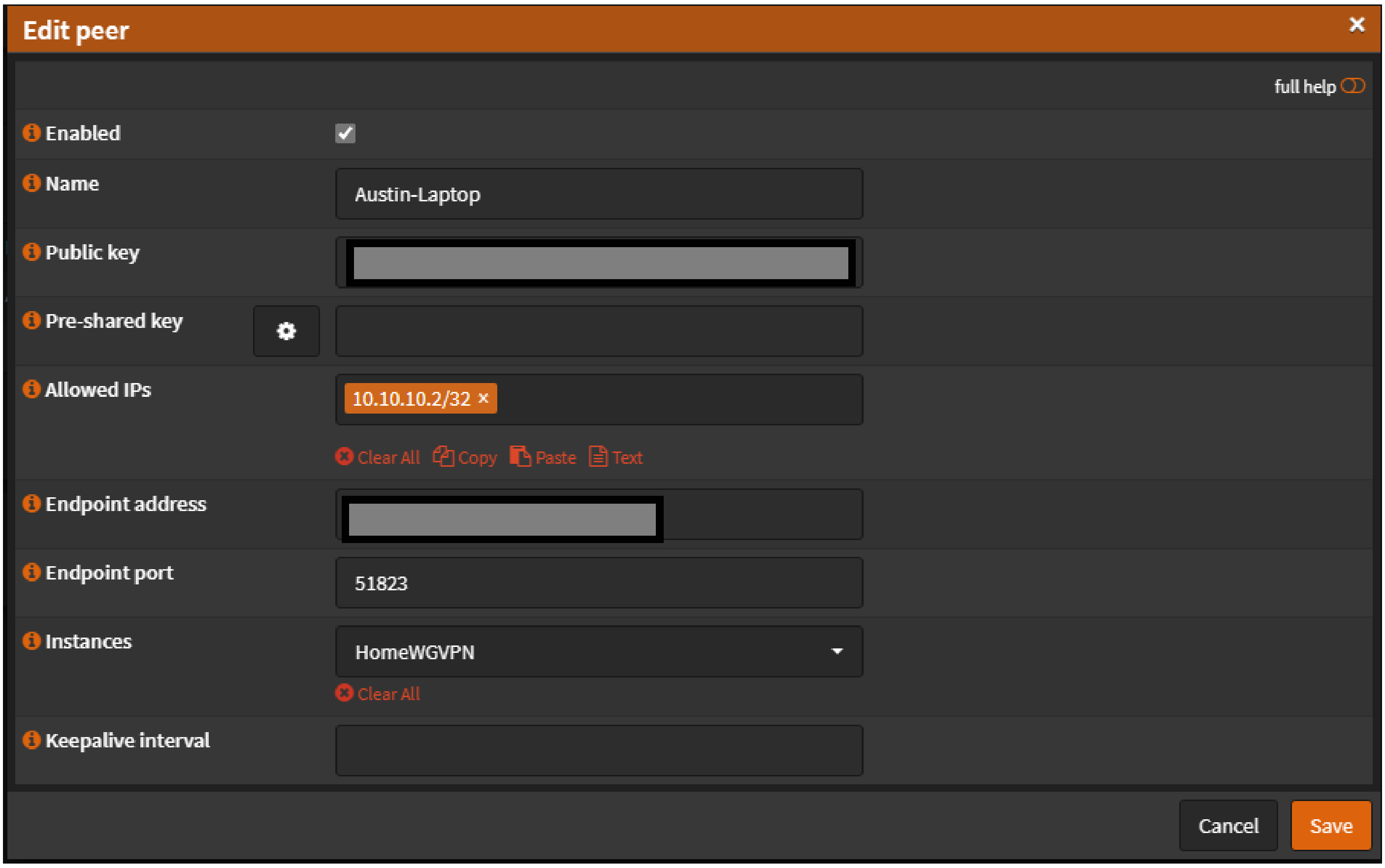
Depend on (CARP): None

Peers: Austin-Laptop

Disable routes □

- Name your instance
- WireGuard uses port 51820 by default, use this port or a higher unique port
- Identify subnets and/or IPs you want accessible through the tunnels
- Generate your public and private keys by clicking the Gear Icon. Save these as you'll need it when setting up your peer

Next, navigate to the "Peer" tab next to Instances, and select the "+" icon to add a new peer. Keep in mind, you'll need to be configuring your WireGuard Client simultaneously as you configure your peer, as you'll need you public key from your WireGuard client:



Edit peer

Enabled	☑
Name	Austin-Laptop
Public key	[redacted]
Pre-shared key	
Allowed IPs	10.10.10.2/32 ×
Endpoint address	[redacted]
Endpoint port	51823
Instances	HomeWGVPN
Keepalive interval	

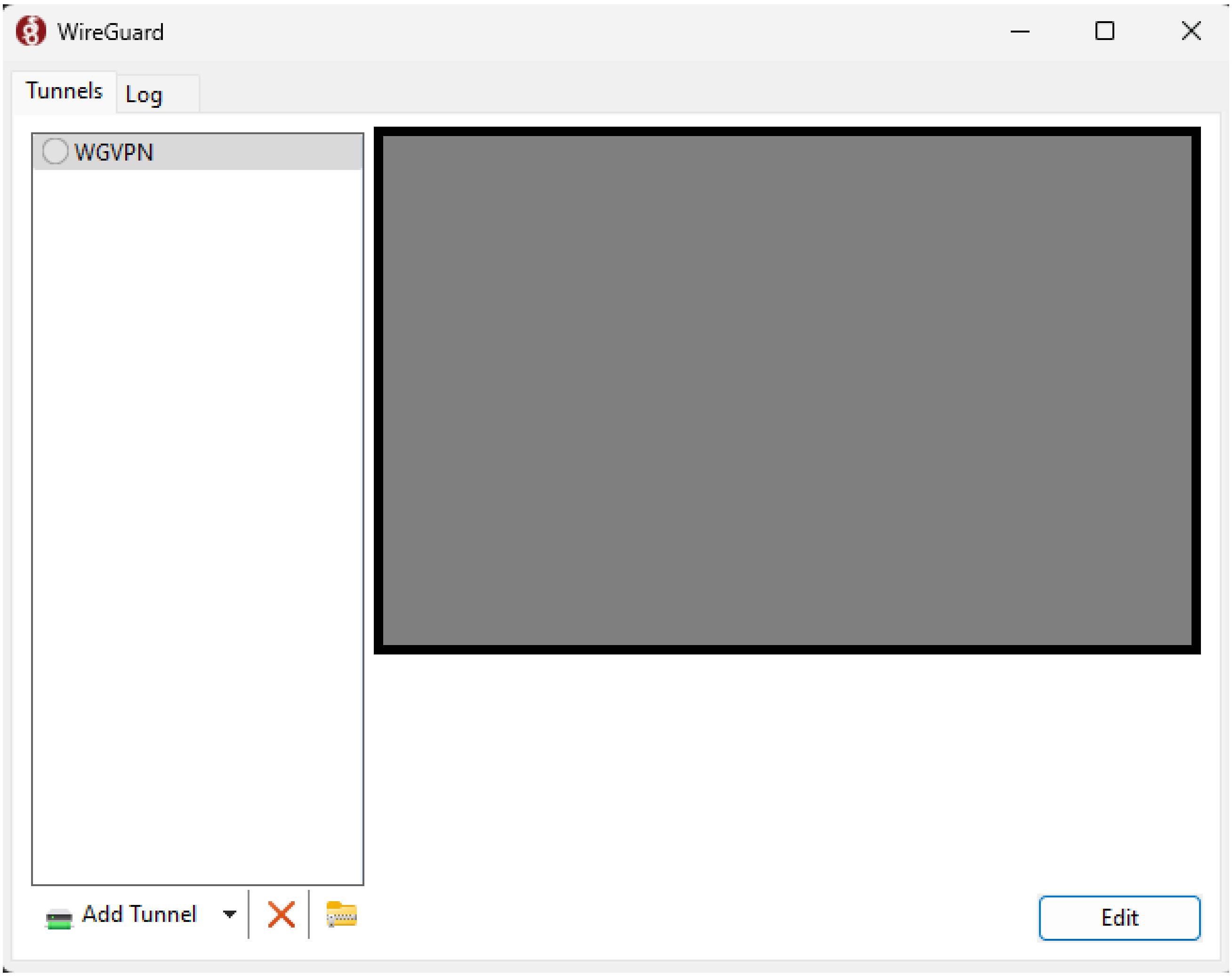
- Here, you'll want to create a name. I used Austin-laptop as I'll be configuring my laptop for connection
- Next, take the public key generated from you client tunnel configuration and input it here
- For endpoint address, input the domain or WAN IP you'll be using. If you've set up a cloudfare ddns subdomain with OPNsense client, you can do as I did:
    - I entered vpn.example.com as that was the subdomain I've configured with cloudflare to automatically be updated with my WAN IP
- For allowed IPs, give your machine a designated IP address in CIDR notation in the subnet. I've established in my Instance that the tunnel will be using the 10.10.10.1/24 subnet, so I've given the peer an address of 10.10.10.2/32

Client

On your laptop or WireGuard client that will be connecting to this network, you'll need to set up a config file.

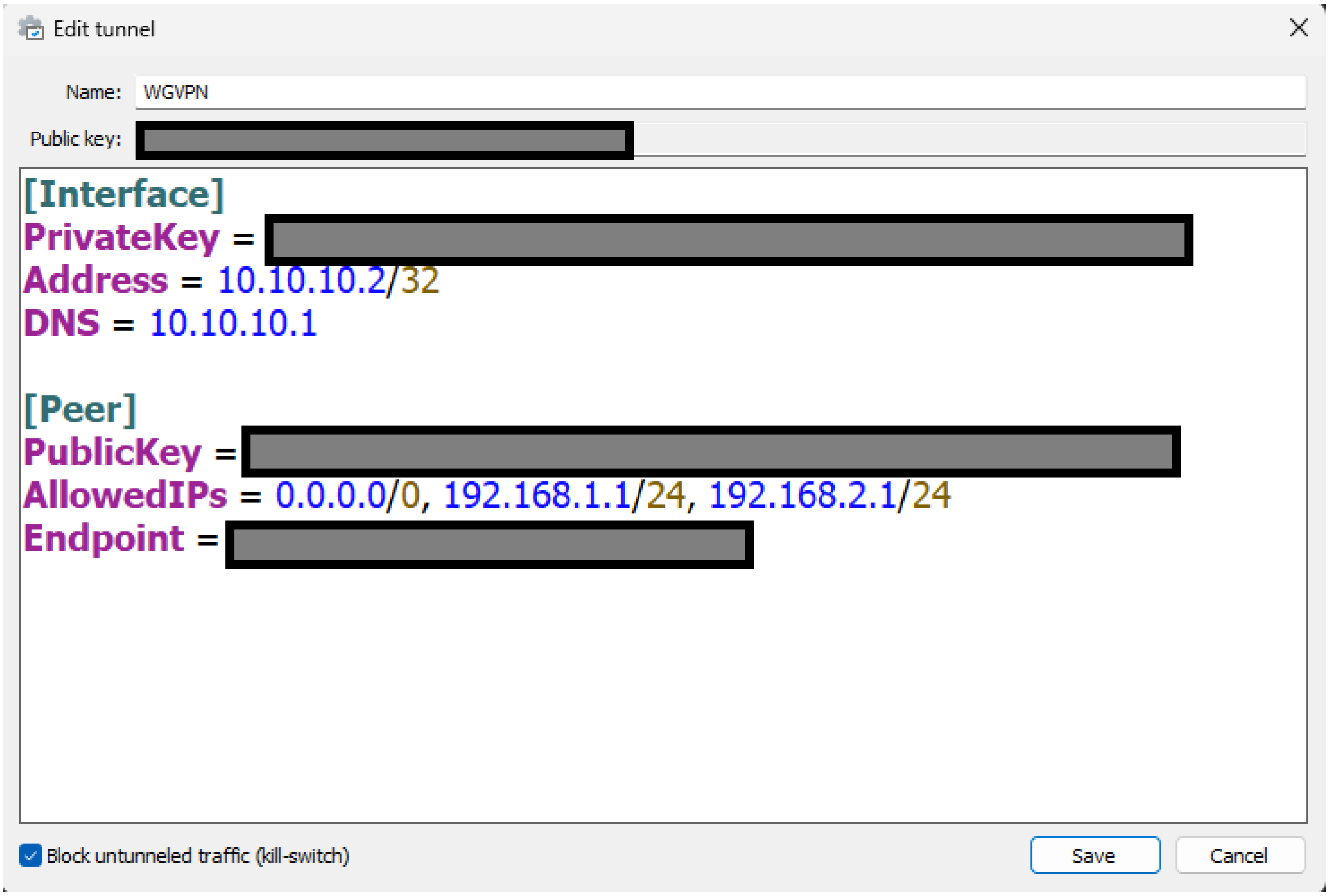
Install the WireGuard client by downloading it from their website - WireGuard

Launch the client on your laptop, and select add tunnel:



[redacted]

Next, you be able to configure your tunnel



Name: WGVPN

Public key: [redacted]

```
[Interface]
PrivateKey = [redacted]
Address = 10.10.10.2/32
DNS = 10.10.10.1

[Peer]
PublicKey = [redacted]
AllowedIPs = 0.0.0.0/0, 192.168.1.1/24, 192.168.2.1/24
Endpoint = [redacted]
```

☑ Block untunneled traffic (kill-switch)

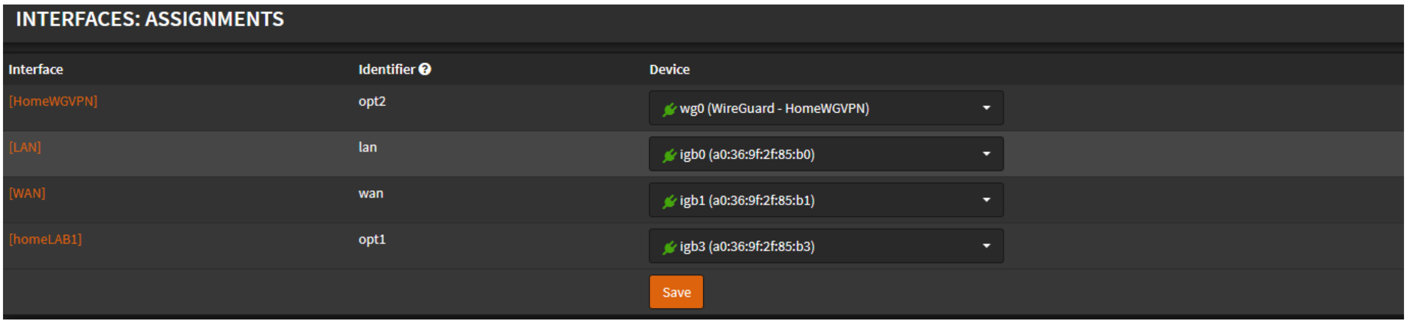
- When configuring your client, add the Address line and use your AllowedIP that you entered in your OPNsense Peer configuration
- For DNS, enter your gateway, or a specific DNS server you use
- For PublicKey, enter the Public Key of the instance that was created in OPNsense
- For AllowedIPs, enter the subnets or specific IPs you'd like to access
- For endpoint, enter your WAN IP & port#. If you're using a subdomain with Cloudflare DDNS, enter your.subdomain.com:portnumber

With OPNsense configured and your client configured, you'll just need to configure some firewall rules to let your computer access local devices. Check out the next page to see how!

Firewall Rules

The last step of your WireGuard set up involved creating 2 firewall rules. One for your WAN firewall, and one for your Tunnel.

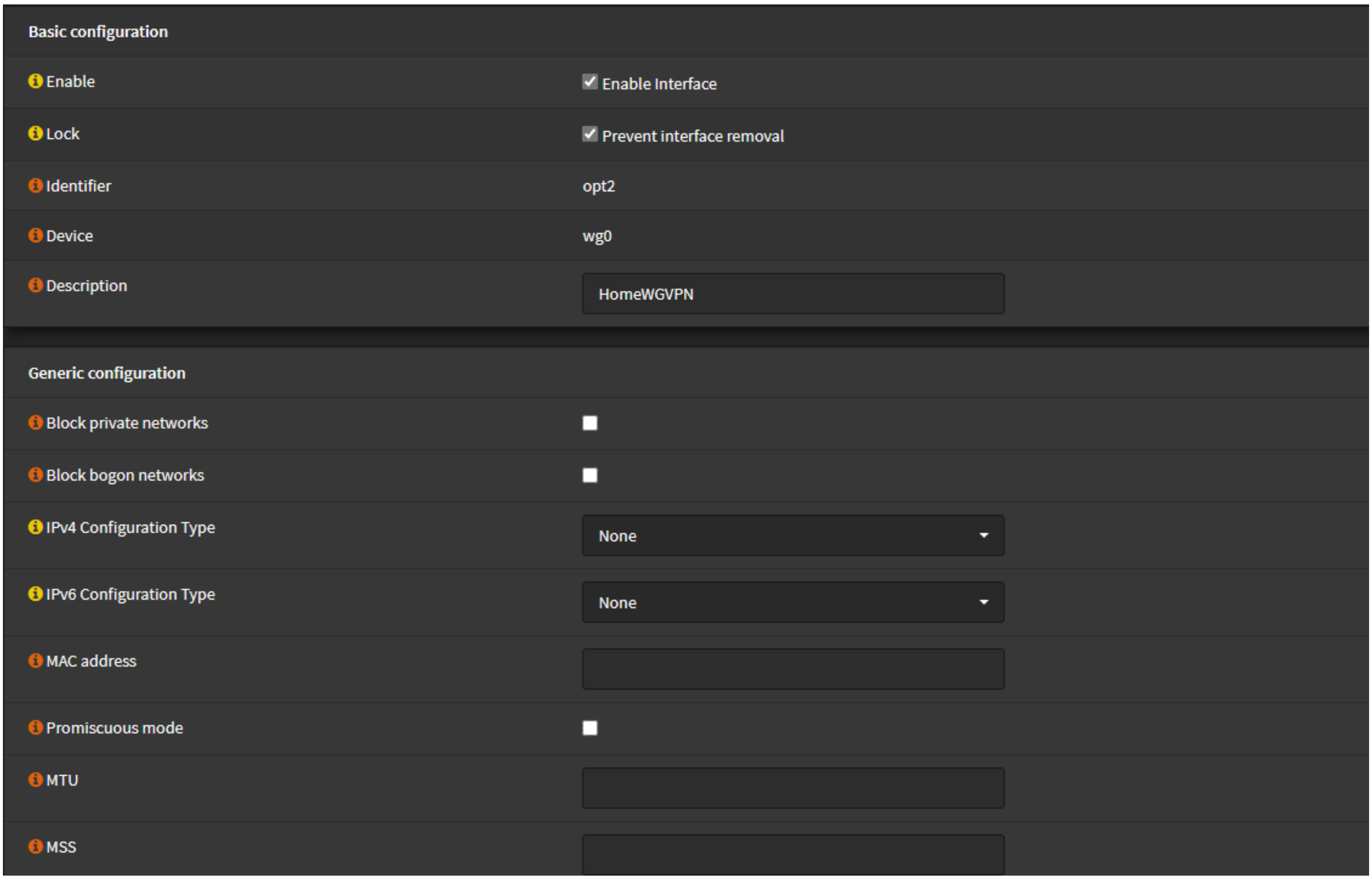
If you haven't done so already, assign your WireGuard VPN as an interface. To do so, navigate to Interfaces ---> Assignments:



INTERFACES: ASSIGNMENTS

Interface	Identifier	Device
[HomeWGVPN]	opt2	wg0 (WireGuard - HomeWGVPN)
[LAN]	lan	igb0 (a0:36:9f:2f:85:b0)
[WAN]	wan	igb1 (a0:36:9f:2f:85:b1)
[homeLAB1]	opt1	igb3 (a0:36:9f:2f:85:b3)

Select your Interface in the sidebar menu:



Basic configuration

Enable	☑ Enable Interface
Lock	☑ Prevent interface removal
Identifier	opt2
Device	wg0
Description	HomeWGVPN

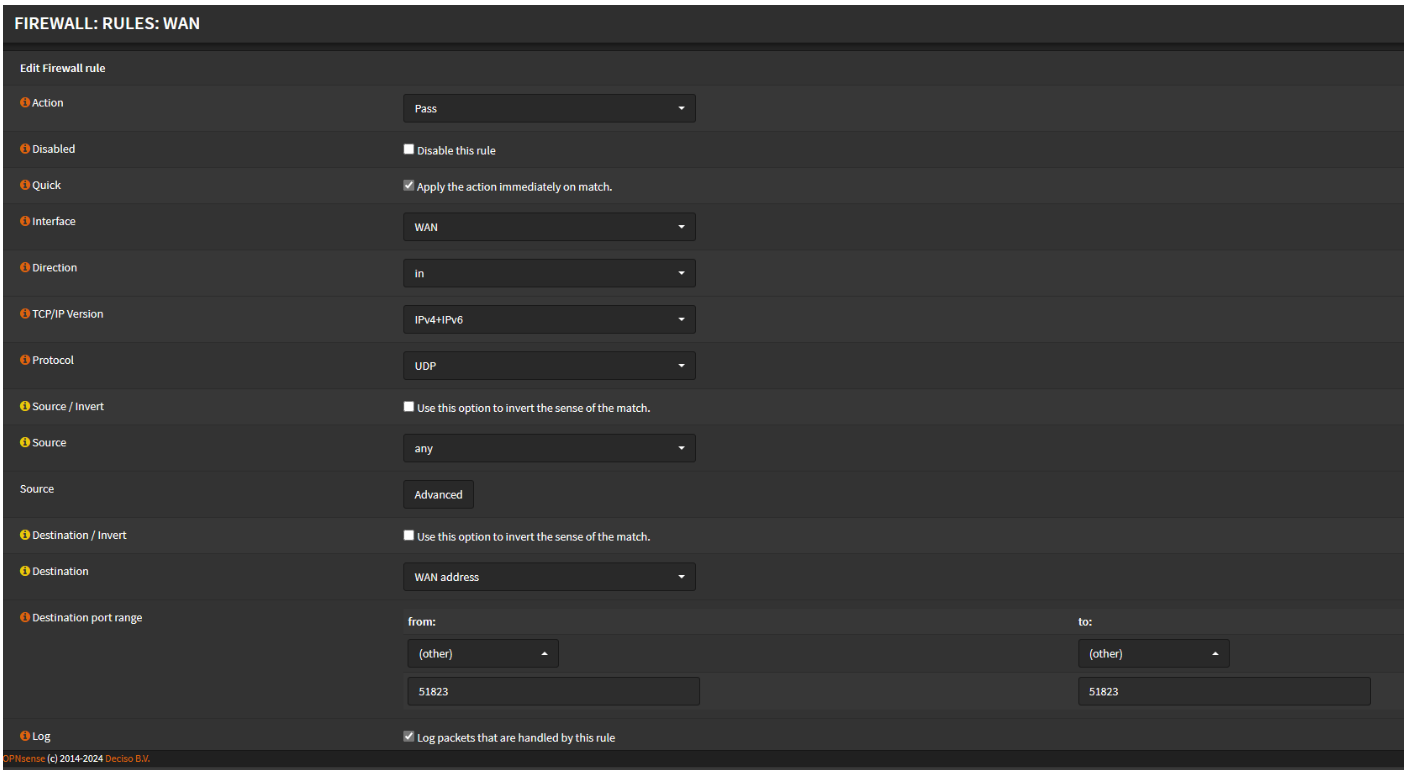
Generic configuration

Block private networks	□
Block bogon networks	□
IPv4 Configuration Type	None
IPv6 Configuration Type	None
MAC address	
Promiscuous mode	□
MTU	
MSS	

- Enable the interface
- Lock to prevent removal
- No other configs need to be done, save changes and apply settings.

WAN Rule

Navigate to Firewall ---> Rules ---> WAN and create a new rule:



FIREWALL: RULES: WAN

Edit Firewall rule

Action	Pass
Disabled	□ Disable this rule
Quick	☑ Apply the action immediately on match.
Interface	WAN
Direction	in
TCP/IP Version	IPv4+IPv6
Protocol	UDP
Source / Invert	□ Use this option to invert the sense of the match.
Source	any
Source	Advanced
Destination / Invert	□ Use this option to invert the sense of the match.
Destination	WAN address
Destination port range	from: (other) 51823 to: (other) 51823
Log	☑ Log packets that are handled by this rule

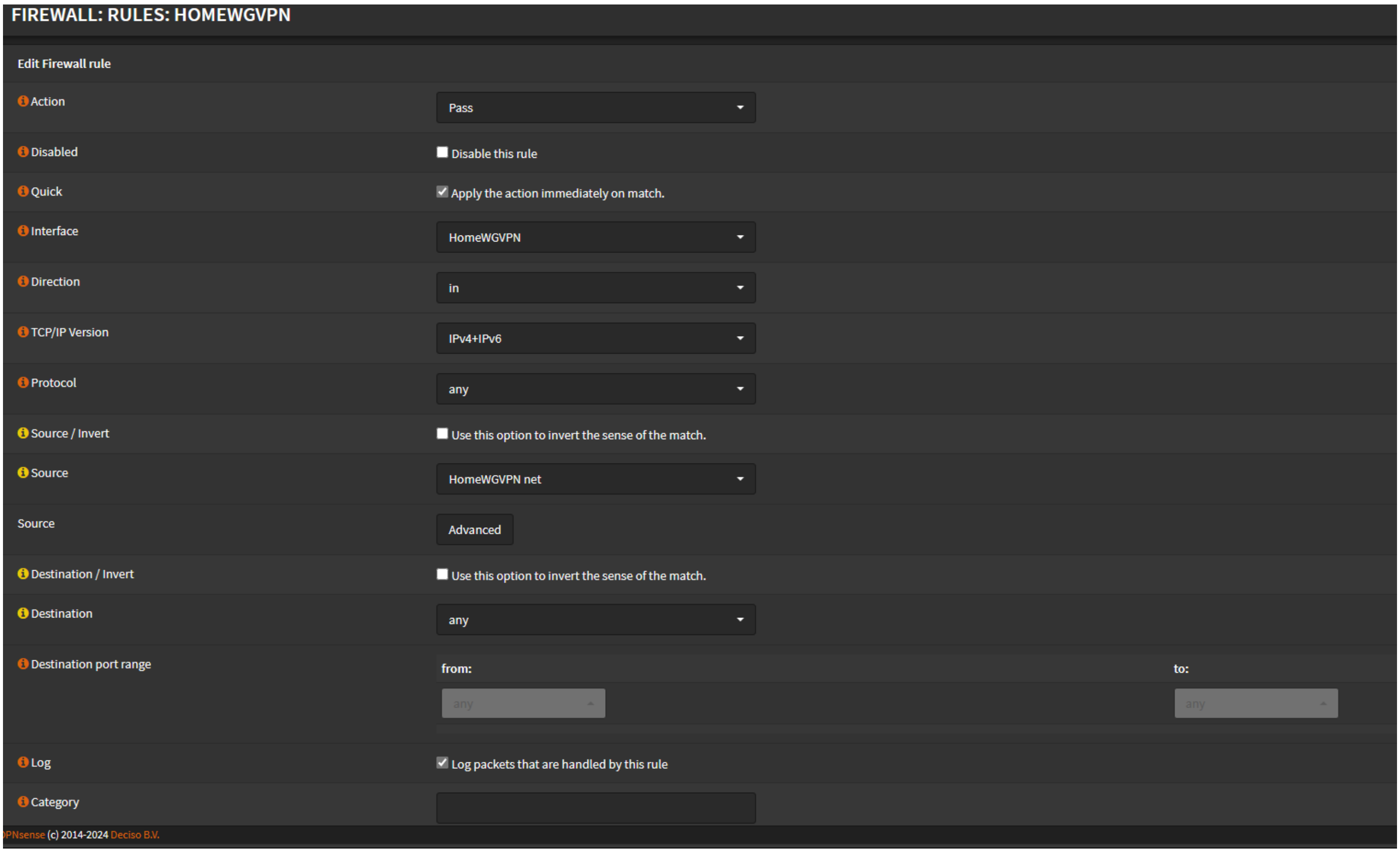
OPNsense (c) 2014-2024 Deciso B.V.

Select the following settings for the rule:

- Action = Pass
- Quick - checked
- Interface = WAN
- Direction = In
- TCP/IP Version = IPv4 + IPv6
- Protocol = UDP
- Source = ANY
- Destination = WAN Address
- Destination Port Range = Enter your Port Number you designated earlier
  - Default WireGuard port is 51820
- Save and apply rules

WireGuard Interface Rule

Navigate to your Firewall ---> Rules ---> Select your WireGuard Interface then click create:



FIREWALL: RULES: HOMEWGVPN

Edit Firewall rule

Action	Pass
Disabled	□ Disable this rule
Quick	☑ Apply the action immediately on match.
Interface	HomeWGVPN
Direction	in
TCP/IP Version	IPv4+IPv6
Protocol	any
Source / Invert	□ Use this option to invert the sense of the match.
Source	HomeWGVPN net
Source	Advanced
Destination / Invert	□ Use this option to invert the sense of the match.
Destination	any
Destination port range	from: any to: any
Log	☑ Log packets that are handled by this rule
Category	

PNsense (c) 2014-2024 Deciso B.V.

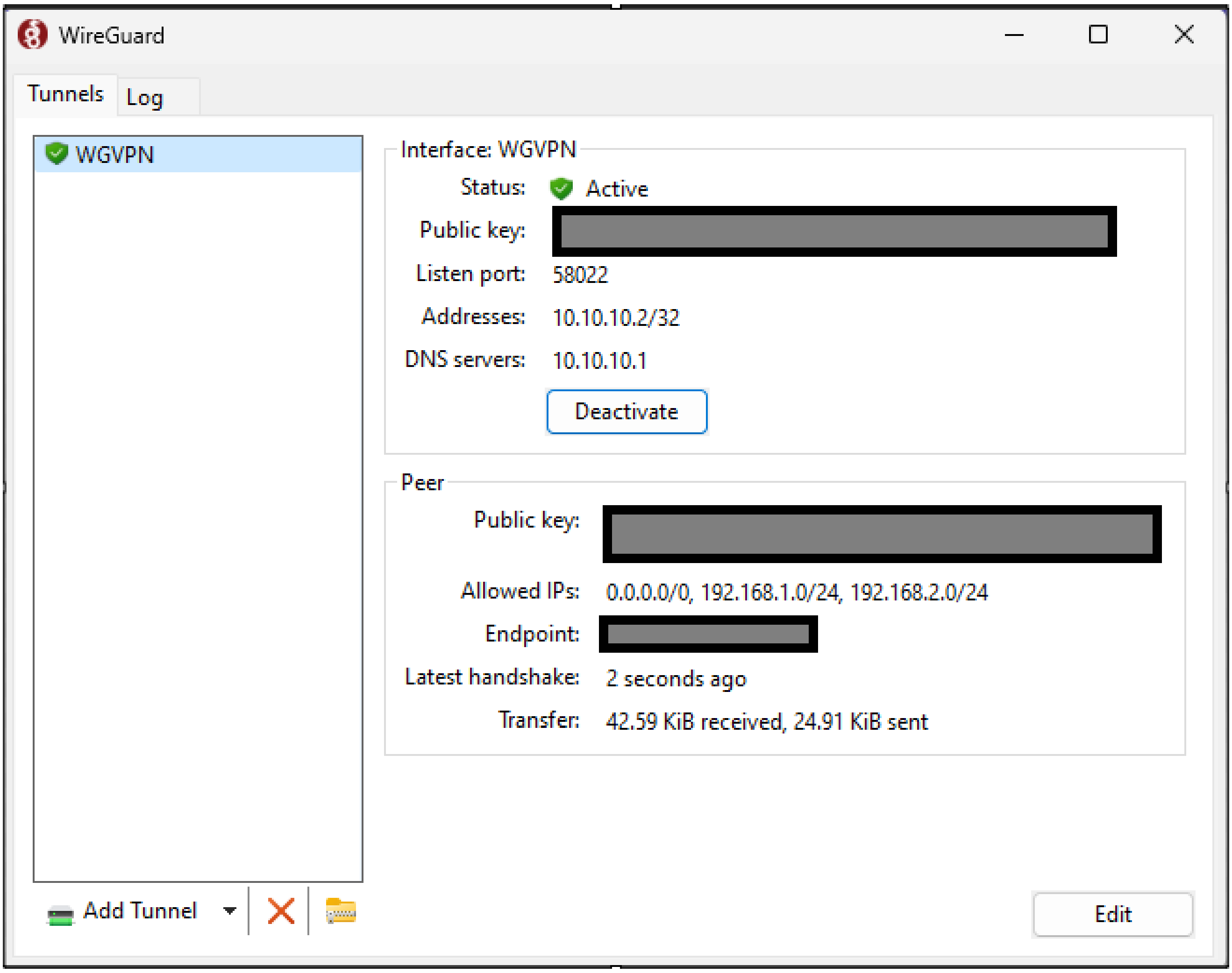
This rule will allow your client to access any device on the local network. Configure the following settings:

- Action = Pass
- Quick = Checked
- Interface = WireGuard Interface
- Direction = In
- TCP/IP Version = IPv4 + IPv6
- Protocol = any
- Source = Select your WireGuard Interface Net as source
- Destination = Any
- Save and apply rules.

Your configuration should now be all set. Check out the next page to view results!

Connection Results

To test this, I went to a library and tried connecting to my network via my WireGuard Client:



Interface: WGVPN

Status: Active

Public key: [redacted]

Listen port: 58022

Addresses: 10.10.10.2/32

DNS servers: 10.10.10.1

Peer

Public key: [redacted]

Allowed IPs: 0.0.0.0/0, 192.168.1.0/24, 192.168.2.0/24

Endpoint: [redacted]

Latest handshake: 2 seconds ago

Transfer: 42.59 KiB received, 24.91 KiB sent

- You can see my connection status is active
- My latest handshake was 2 seconds ago
- I can connect to local devices when trying to ping or through accessing their Web GUIs at their local IPs and was able to manage my services remotely.