**The easiest way to secure Home Assistant with HTTPS**

 2021.3.13  2021.9.30  4 mins

*30.09.2021*: added configuration changes required by Home Assistant version 2021.7 and later (X-Forwarded-For)

**Motivation**

If you’ve made up your mind to expose your Home Assistant instance to the wild Internet, which is full of evil hackers, you should take care of the security. Today we’ll learn how to set up your HTTPS connection really quick and without pain. Well, almost no pain.

We’ll use Nginx Proxy Manager addon which will do all the dirty work for us. Why not use [DuckDNS integration](https://www.home-assistant.io/integrations/duckdns/" \t "_blank) or [Let’s Encrypt add-on](https://github.com/home-assistant/addons/tree/master/letsencrypt) instead? Well, those two also work well, but my goal was to find a scalable solution that will support as many addons with their own domains as necessary. I personally don’t like the idea of editing certificate file location for each addon configuration again and again.

What essentially Nginx Proxy Manager addon does is called “SSL termination”. It will receive encrypted incoming traffic from the internet, decrypt it and pass it to your local Home Assistant instance so that it will not even notice that HTTPS is used. You can still use local name inside your local network like hassio.local:8123

This method works pretty well with Google Home, Amazon Alexa, and other services where HTTPS endpoint is required.

**What is required**

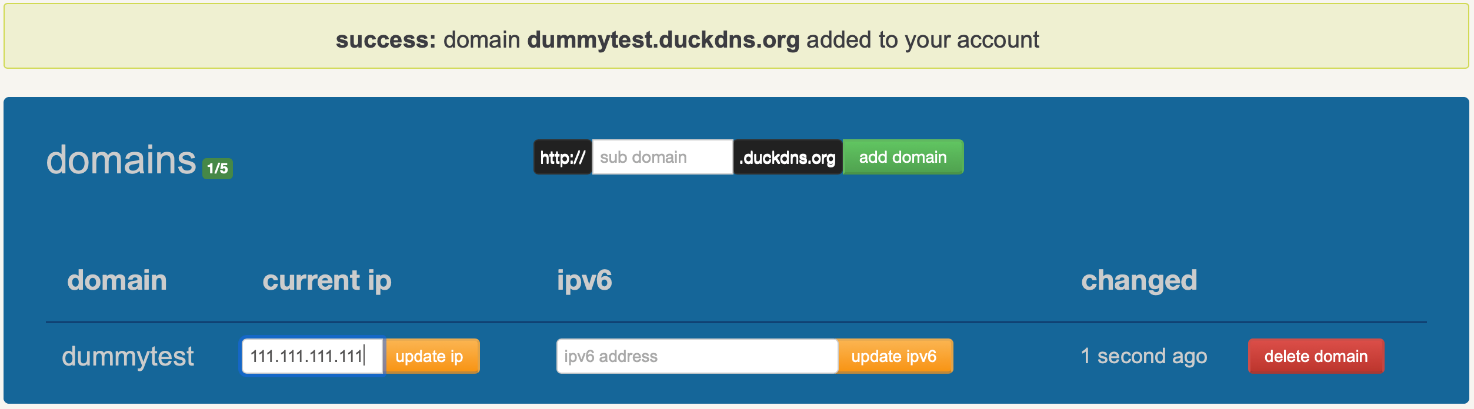
1. A static IP address that is accessible from outside. You can also use a dynamic IP address, but you should set up a Duckdns client which will send updates to Duckdns service once your IP is changed. Duckdns client set up is not covered within this article.
2. duckdns.org account or your own domain
3. Home Assistant OS or Home Assistant Supervised (we need one which supports add-ons)

**Set up port forwarding on your router**

There are plenty of resources describing how to set up port forwarding on a router, you can find one by googling port forwarding <router name>. Once you figured it out, you’ll need to forward ports 443 (for HTTPS) and 80 (for Let’s Encrypt ACME protocol) to the local IP address of the machine running Home Assistant.

**Grab a domain at duckdns.org**

Log in to [https://duckdns.org](https://duckdns.org/) and choose a domain (I will use [http://dummytest.duckdns.org](http://dummytest.duckdns.org/) as an example below):



The **current ip** field will show your IP address, if it does not match your real IP, this is the place where you can change it.

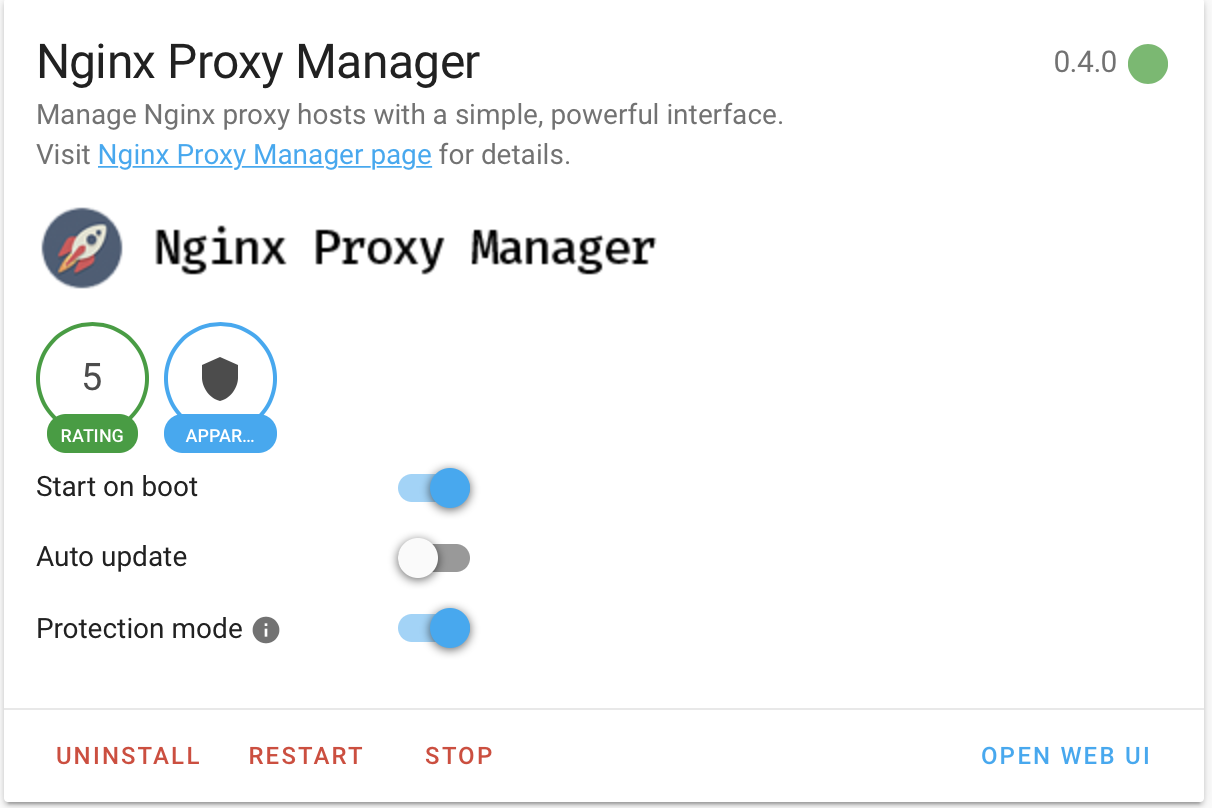
**Set up Nginx Proxy Manager addon**

Open your Home Assistant, go to **Supervisor**-> **Add On Store**. You want two addons: **Nginx Proxy Manager** and **MariaDB**.

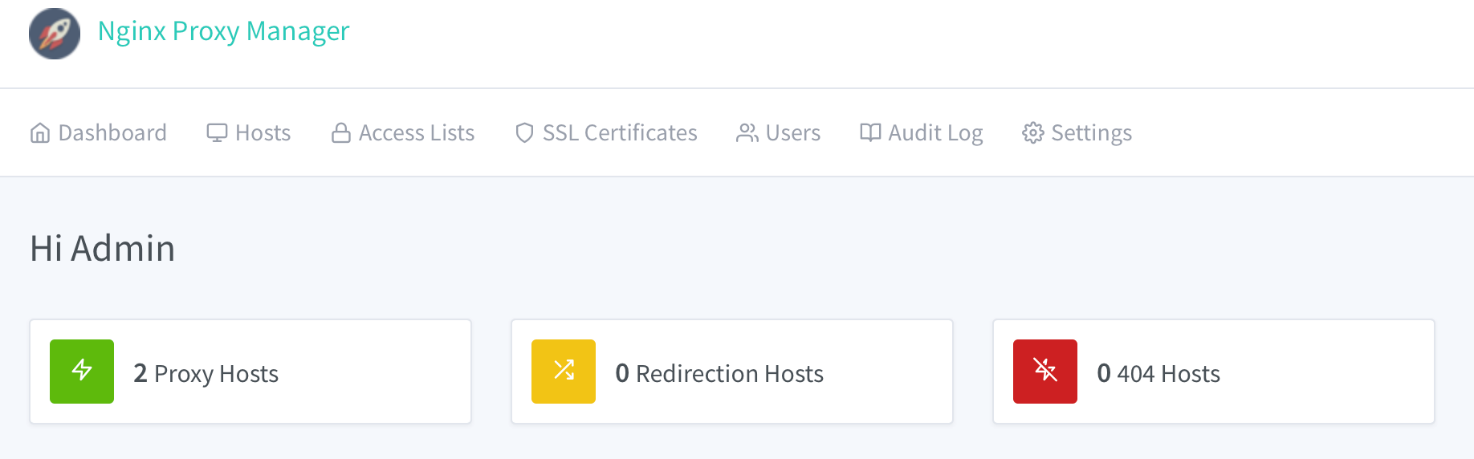
A few words about MariaDB addon dependency. Former versions of Nginx Proxy Manager did not require MariaDB for operation, instead, they had it embedded as this is an essential part of Nginx Proxy Manager software. Later on, Frenk, who is the maintainer of the majority of Home Assistant OS addons, decided to split this dependency up so that HA users who already use MariaDB addon for the Home Assistant recorder component may not host extra instances of this database.

A minimum configuration setting for MariaDB addon is a password for homeassistant user, which is not necessary for Nginx Proxy Manager but is essential if you decide to switch from SQLite database to something [more productive](https://github.com/home-assistant/hassio-addons/tree/master/mariadb#home-assistant-configuration). This is anyway is a good improvement which speed-ups Logbook and History pages in Home Assistant.

Once the password is set, MariaDB should start without errors. Now we need to install Nginx Proxy Manager. Once started, it will automatically find and connect to MariaDB addon without user intervention. No additional settings are required, we should open its page via **Open Web UI** link:



Log in using admin@example.com as username and password changeme and immediately change password.



Go to **Proxy Hosts**, add your host, enable **Websockets Support** and hit **Save**:

Et billede, der indeholder tekst

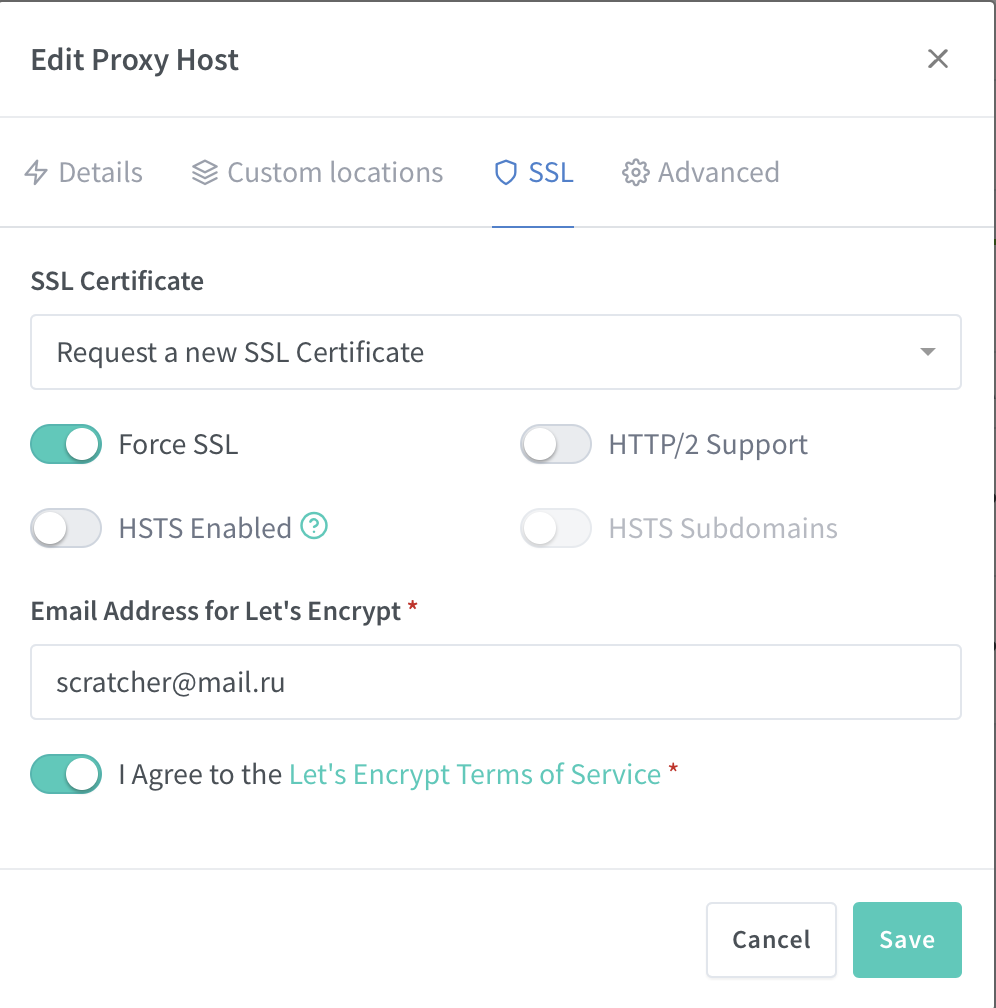
Automatisk genereret beskrivelse

For Home Assistant versions 2021.7 and later you have to add following changes to the http section of configuration.yaml and restart Home Assistant:

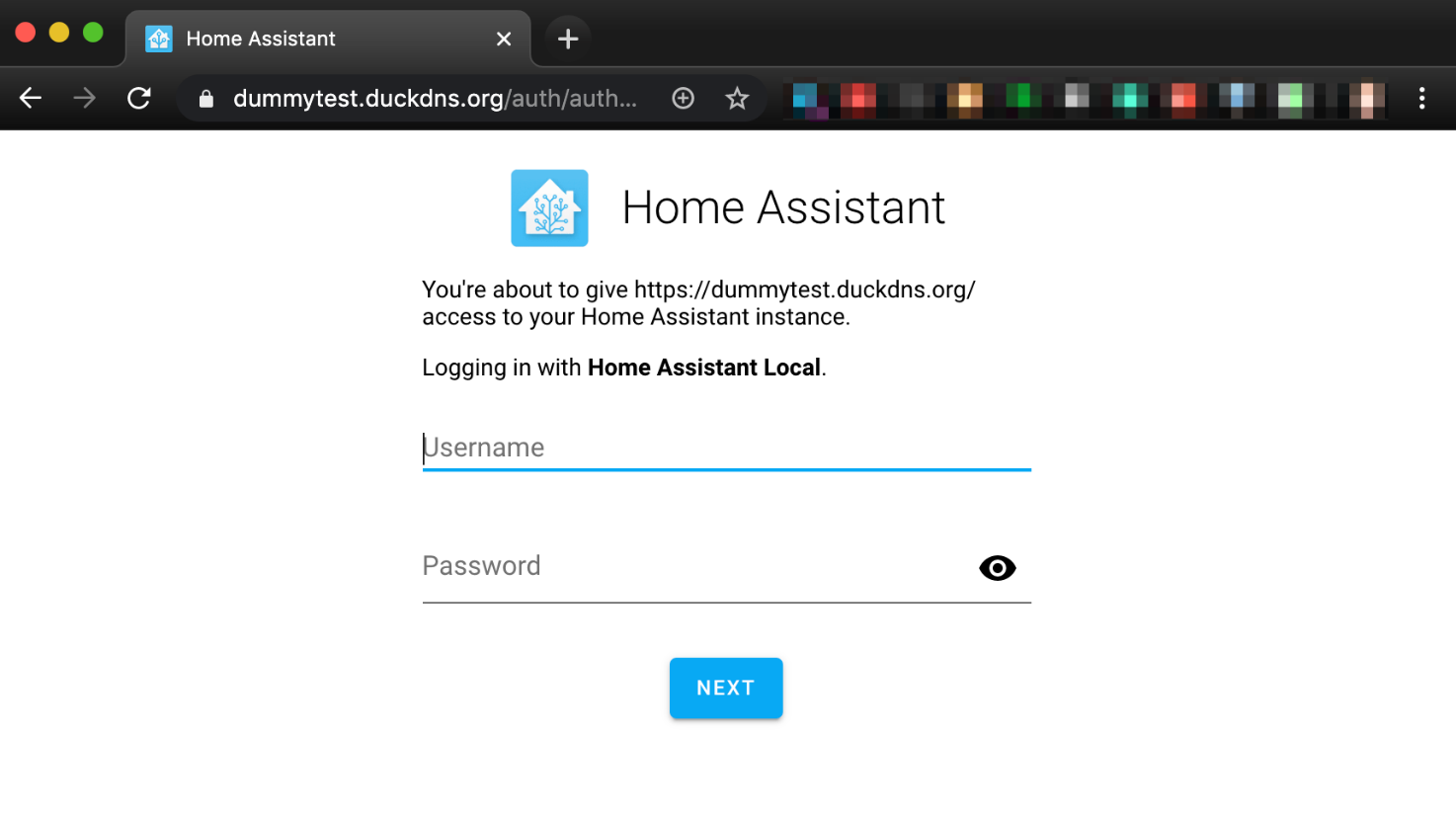
|  |  |
| --- | --- |
| 1  2  3  4 | http:  use\_x\_forwarded\_for: true  trusted\_proxies:  - 172.30.33.0/24 |

At this point it is recommended to open up Home Assistant URL [http://dummytest.duckdns.org](http://dummytest.duckdns.org/) and ensure that basic HTTP access is working. It will be used to obtain a Let’s Encrypt certificate at the next step. If you don’t see your HA prompt, most probably port forwarding was not set up properly, additionally you could check your router’s firewall settings.

If Home Assistant is accessible (via HTTP), go back to the Nginx Proxy Manager addon page and edit the previously created connection. Go to **SSL** tab and select **Request a new SSL Certificate**, the switches **Force SSL** and **I Agree to…** should also be turned on. Save your settings:



That’s it! You can now open [https://dummytest.duckdns.org](https://dummytest.duckdns.org/) in your browser and ensure that your connection is secure:



Addon will take care of automatic Let’s Encrypt certificate renewal. It is also highly recommended to set up [two-factor authentication](https://www.home-assistant.io/docs/authentication/multi-factor-auth/). As a second factor one may use one of TOTP smartphone applications like Google Authenticator or Authy.