

# This is the official FAQ of [up.rdsl.ga](https://up.rdsl.ga)

## Table of content

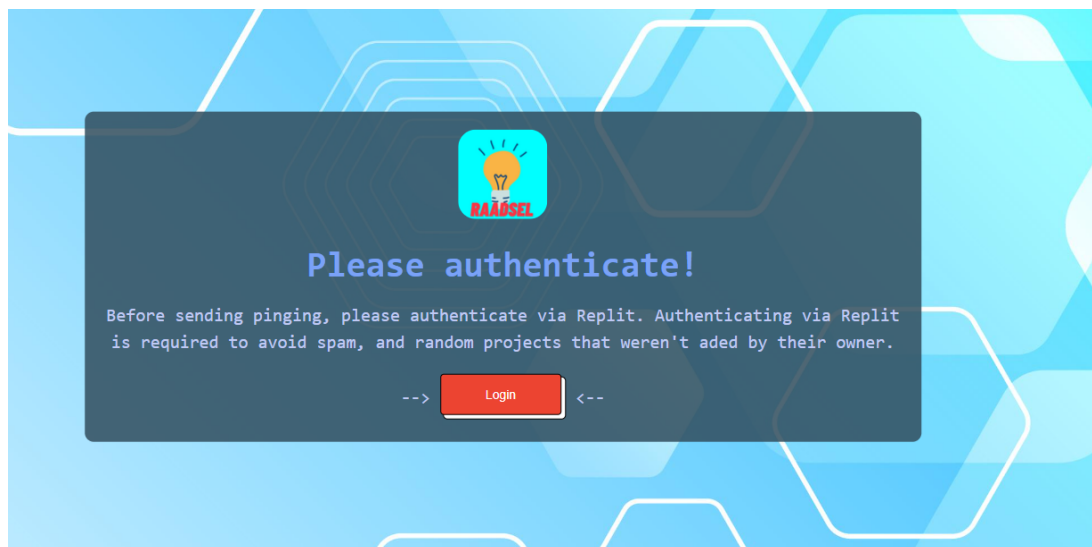
- [How do I add my webserver](#)
- [How do I create a webserver](#) (Python & NodeJS tutorial)
- [How does this work?](#)

Alright heres the FAQ:

### How do I add my webserver to RDSL Pinger?

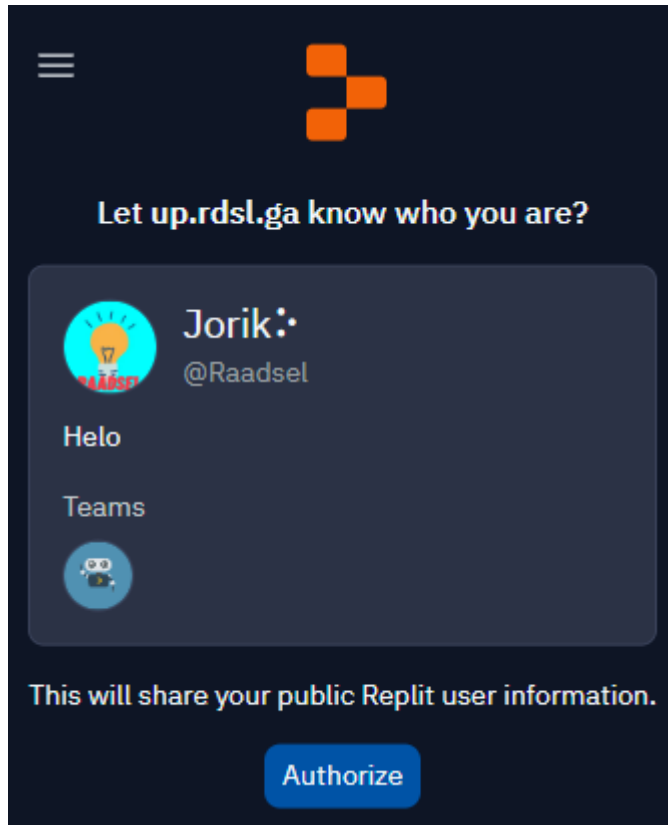
Well its pretty simple. Just follow this easy step-by-step tutorial  
**step 1**

Go to [up.rdsl.ga](https://up.rdsl.ga). If you visit this site for the first time you will see this:



**step 2**

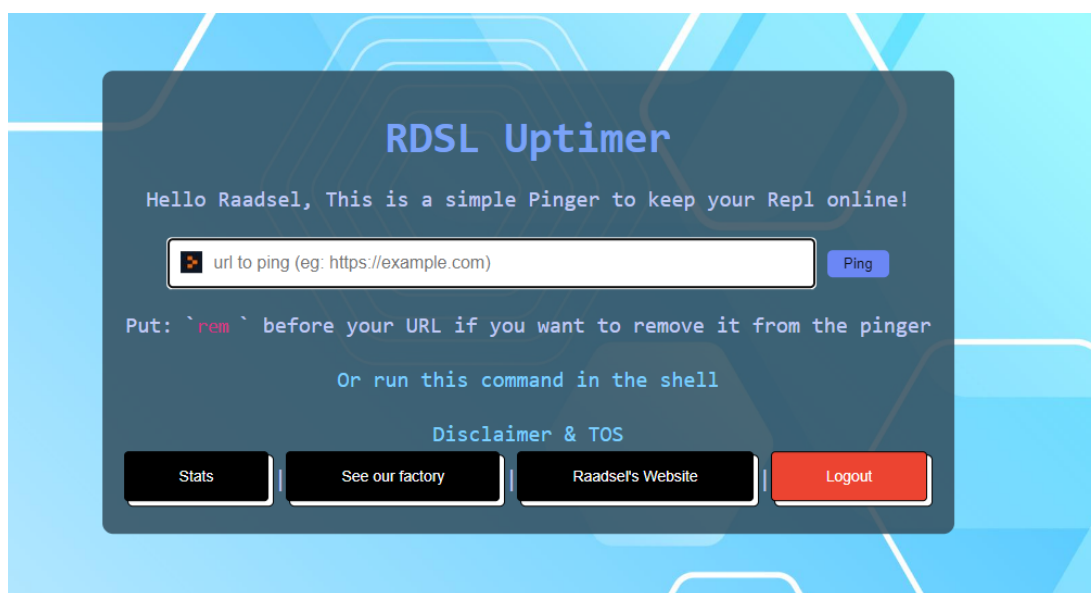
You now need to press login and a popup should show up.



Press Authorize. This is completely safe and just verifies that you are you!  
If instead of Authorize popup a Replit login page pops up. You have to login to Replit first to be able to use Replit Auth.

### step 3

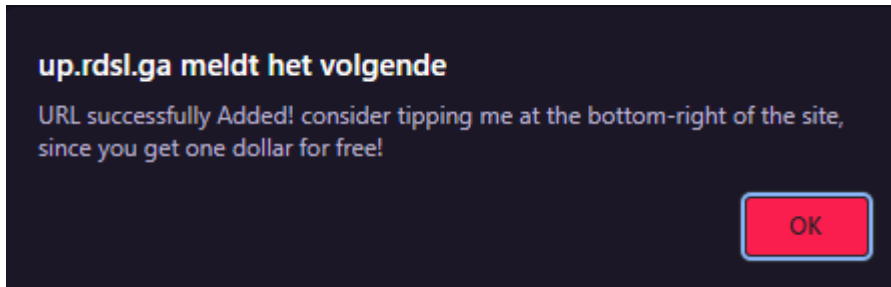
Now you should be at the home page.



In the input-bar you have to input a URL to your webserver. You can get this by going to the webview tab at your Replit and copy the URL



Now press "Ping". It should load a bit (this can take a bit because it has to verify that it is a Repl, and that you are the owner.) Now you should get a alert. If it has been added to the Pinger you should see something like this



. If something went wrong you should get a message with the error and how to fix it. Just follow those instructions!

## How do I create a webserver?

To keep your Repl online you have to create a webserver to ping. The code to create a webserver is different with each language so I will show a nodeJS example, and a Python example. If you are using Golang or another language you can just search for a webserver for that language [online](#).

### NodeJS example:

There are 2 ways to create a webserver The 1st (and the easiest) way: Go to the shell and type `npm install replit-alive-server` Now go to your main file and paste this somewhere: `require('replit-alive-server')()`; . It should start the webserver!

The 2nd way: Just paste this in the main js file you use.

```
var http = require('http');
http.createServer(function (req, res) {
  res.write("I'm alive"); //You can change this text
  res.end();
}).listen(8080);
// Keep alive using up.rdsl.ga!
```

Now it should create a webserver!

---

### Python example

For python there are also 2 ways:

The 1st (and easy way): go to the shell and run `pip install replit-keep-alive==1.1.0` . Wait untill its done loading. Now go to your main file and paste this code somewhere

```
from replit_keep_alive import WaitressStart
WaitressStart()
```

Now you should have a webserver.

The 2nd way (and harder way): You need to create another file called `keep_alive.py` (name doesn't have to be like this specific)  
In that file paste this code:

```
from flask import Flask
from threading import Thread

app = Flask('')

@app.route('/')
def home():
    return "I'm alive!" #You can change this text

def run():
    app.run(host='0.0.0.0', port=8080) #localhost doesn't work, thats why it is 0.0.0.0

def keep_alive():
    t = Thread(target=run)
    t.start()
# Keep alive using up.rdsl.ga!
```

Now go to your main py file and input this code somewhere

```
from keep_alive import keep_alive
keep_alive()
```

Now it should create a webserver!

---

## How does this work?

By default Repls aren't always online, but go offline after some time of no activity. There are two ways to fix this. Either buy Hacker plan and use [Always On](#), or create a webserver and use a pinger. (example for pinger: <https://up.rdsl.ga>). I've shown you 2 tutorials for Python and NodeJS, but the concept works for anything in any language.!