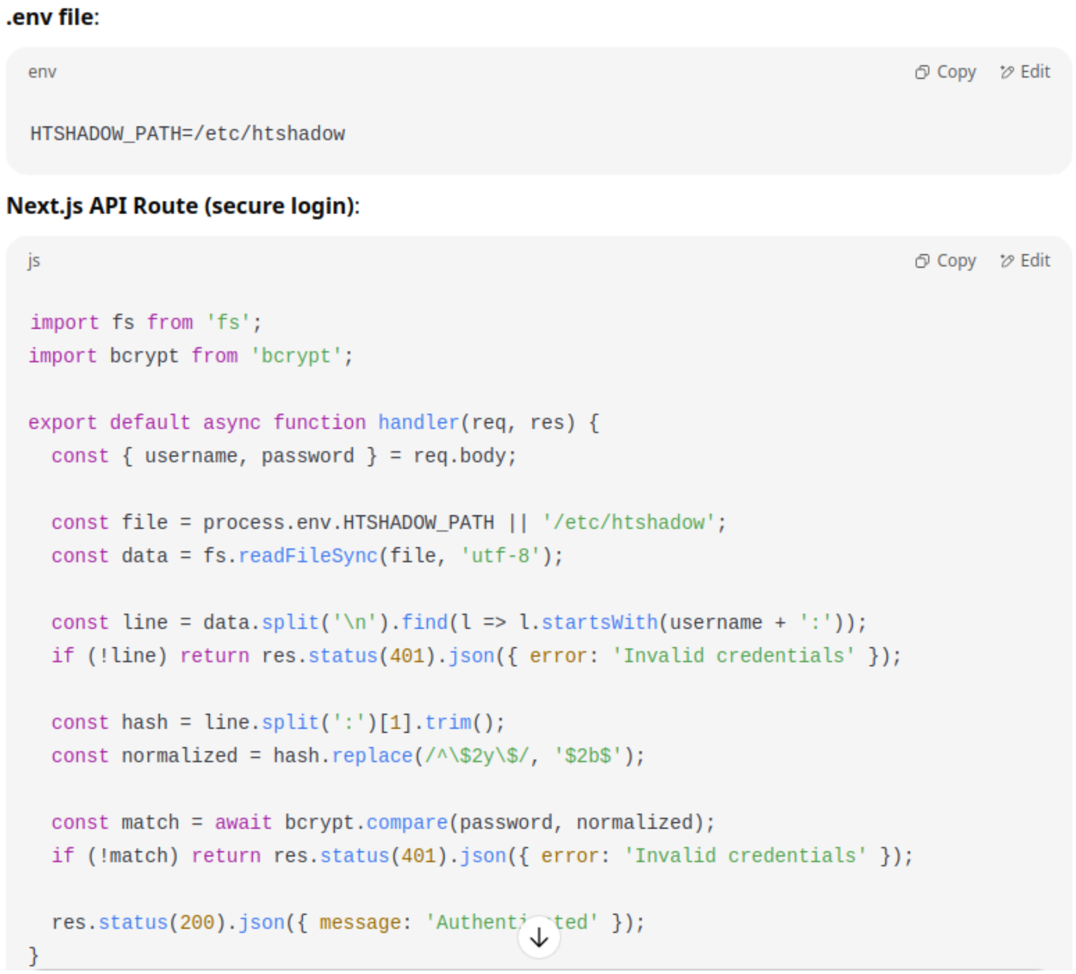
For development purpose the passwords are saved in the file /etc/example_htshadow
The example file is already there on the web server, so you can use it
It looks like this:

```
test1:$2y$05$/DOLvW/Ik.IObiHeAhCaEeHEbfZBozBvHihclOISfRAG4kKu4MuFe
test2:$2y$05$Y3x6nEsss4CNpqkajRfpPuYWvXKybRCVGuemitDxPQZSJzTdx/thC
test3:$2y$05$xyifOKtxdYLqkmLThhnjwul412EjjeuiCtvjc4fJqEb8Rqhmhc6l2
```

test1 has password `xc4uuicX`
test2 has password `YTxuiT8c`
test3 has password `JXNXkmdD`

It is **bcrypt-hashed password**

Here is an example generated by Chatgpt how to read the hash and confirm the password. Instead of /etc/htshadow use **/etc/example_htshadow** Later, when it is deployed I will change it.



.env file:

```
HTSHADOW_PATH=/etc/htshadow
```

Next.js API Route (secure login):

```js
import fs from 'fs';
import bcrypt from 'bcrypt';

export default async function handler(req, res) {
  const { username, password } = req.body;

  const file = process.env.HTSHADOW_PATH || '/etc/htshadow';
  const data = fs.readFileSync(file, 'utf-8');

  const line = data.split('\n').find(l => l.startsWith(username + ':'));
  if (!line) return res.status(401).json({ error: 'Invalid credentials' });

  const hash = line.split(':')[1].trim();
  const normalized = hash.replace(/^\$2y\$/, '$2b$');

  const match = await bcrypt.compare(password, normalized);
  if (!match) return res.status(401).json({ error: 'Invalid credentials' });

  res.status(200).json({ message: 'Authenti ted' });
}
```