

Node Auth Tutorial (JWT)

#7 - Hashing Passwords



In this Node auth tutorial we'll see how to hash our passwords before saving them to the database using bcrypt.



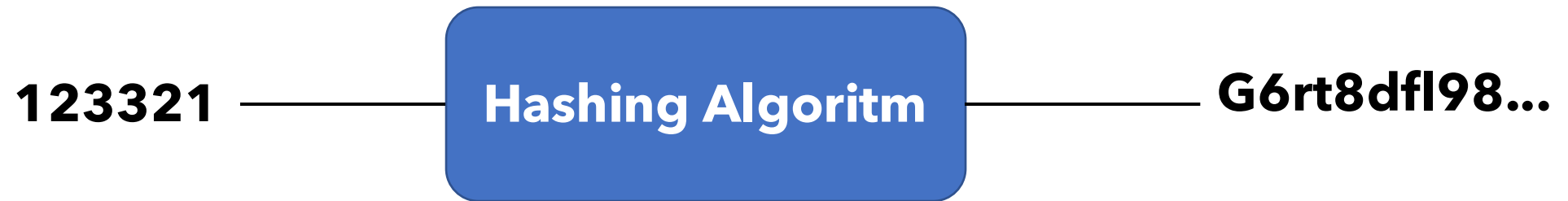
express

- Terminal

```
npm install bcrypt
```



Hashing Passwords



Hashing Passwords



```
const mongoose = require('mongoose');
const { isEmail } = require('validator');
const bcrypt = require('bcrypt');

const userSchema = new mongoose.Schema({
  email: {
    type: String,
    required: [true, 'Please enter an email'],
    unique: true,
    lowercase: true,
    validate: [isEmail, 'Please enter a valid email']
  },
  password: {
    type: String,
    required: [true, 'Please enter a password'],
    minlength: [6, 'Minimum password length is 6 characters'],
  }
});

// fire a function before doc saved to db
userSchema.pre('save', async function(next) {
  const salt = await bcrypt.genSalt();
  this.password = await bcrypt.hash(this.password, salt);
  next();
});

const User = mongoose.model('user', userSchema);
module exports = User;
```

A diagram consisting of two black arrows pointing from a central text label to two specific lines of code in the code block above. The top arrow points to the line `const bcrypt = require('bcrypt');`, and the bottom arrow points to the line `const salt = await bcrypt.genSalt();`.

Insert code

POST

http://localhost:3000/signup

Send

Params

Authorization

Headers (8)

Body

Pre-request Script

Tests

Settings

Cookies

none

form-data

x-www-form-urlencoded

raw

binary

GraphQL

JSON

Beautify

```
1 {
2   ... "email": "toad@gmail.com",
3   ... "password": "321123"
4 }
```

Body

Cookies

Headers (7)

Test Results



Status: 201 Created

Time: 250 ms

Size: 382 B

Save Response

Pretty

Raw

Preview

Visualize

JSON



```
1 {
2   "email": "toad@gmail.com",
3   "password": "$2b$10$JdAGP12QnWW7TemR7xYTBuLyKEzTG8AjEM3s4KFXCGasf/iCY1TU2",
4   "_id": "612cfc6bf61a40027b9eacb9",
5   "__v": 0
6 }
```



POSTMAN



node-auth.users

COLLECTION SIZE: 369B TOTAL DOCUMENTS: 4 INDEXES TOTAL SIZE: 72KB

Find

Indexes

Schema Anti-Patterns 0

Aggregation

Search Indexes ●

INSERT DOCUMENT

FILTER

{ field: 'value' }

► OPTIONS

Apply

Reset

QUERY RESULTS 1-1 OF 1

```
_id: ObjectId("612cfc6bf61a40027b9eacb9")
email: "toad@gmail.com"
password: "$2b$10$JdAGP12QnW7TemR7xYTBuLyKEzTG8AjEM3s4KFXCGasf/iCY1TU2"
__v: 0
```