

Go Web Examples Courses Easy to follow Video Courses
about Go. Now in early access!

Learn more → ×

Password Hashing

Hey, Philipp here!

I'd like to tell you, that my platform Go Web Examples Courses just launched. Enjoy easy to follow video courses about web development in Go. Make sure to check out the special offer I have for early supporters.

We'll see us over there! :)

Learn more



Password Hashing (bcrypt)

This example will show how to hash passwords using bcrypt. For this we have to `go get` the golang bcrypt library like so:

```
$ go get golang.org/x/crypto/bcrypt
```

From now on, every application we write will be able to make use of this library.

```
// passwords.go
package main

import (
    "fmt"

    "golang.org/x/crypto/bcrypt"
)
```

```
func HashPassword(password string) (string, error) {
    bytes, err := bcrypt.GenerateFromPassword([]byte(password), 14)
    return string(bytes), err
}

func CheckPasswordHash(password, hash string) bool {
    err := bcrypt.CompareHashAndPassword([]byte(hash), []byte(password))
    return err == nil
}

func main() {
    password := "secret"
    hash, _ := HashPassword(password) // ignore error for the sake of simplicity

    fmt.Println("Password:", password)
    fmt.Println("Hash:    ", hash)

    match := CheckPasswordHash(password, hash)
    fmt.Println("Match:   ", match)
}
```

```
$ go run passwords.go
Password: secret
Hash:     $2a$14$ajq8Q7fbtFRQvXpdCq7Jcuy.Rx1h/L4J600tx.gyNLbAYctGMJ9tK
Match:    true
```

[Legal Disclosure](#) [Privacy Statement](#)