package main

import (

"net/http"

"strings"

"github.com/dgrijalva/jwt-go"

)

var jwtSecretKey = []byte("your-secret-key")

func validateJWT(tokenString string) (bool, error) {

token, err := jwt.Parse(tokenString, func(token \*jwt.Token) (interface{}, error) {

return jwtSecretKey, nil

})

if err != nil {

return false, err

}

return token.Valid, nil

}

func authenticationMiddleware(next http.Handler) http.Handler {

return http.HandlerFunc(func(w http.ResponseWriter, r \*http.Request) {

authHeader := r.Header.Get("Authorization")

if authHeader == "" {

http.Error(w, "Missing Authorization Header", http.StatusUnauthorized)

return

}

tokenString := strings.TrimPrefix(authHeader, "Bearer ")

isValid, err := validateJWT(tokenString)

if err != nil || !isValid {

http.Error(w, "Invalid Token", http.StatusUnauthorized)

return

}

next.ServeHTTP(w, r)

})

}

func main() {

http.Handle("/secure-endpoint", authenticationMiddleware(http.HandlerFunc(func(w http.ResponseWriter, r \*http.Request) {

w.Write([]byte("Welcome to the secure endpoint!"))

})))

http.ListenAndServe(":8080", nil)

}