Développement d'un serveur HTTPS en Go et mise en place du certificat.

Création de l'environnement et mise en place du projet.

```
# Initialiser un nouveau module Go
go mod init go-https-server

# Créer le fichier principal main.go
vim main.go
```

N'oubliez pas que vous devez appliquer ces commandes dans ~/go-https-serve

```
import (
    "crypto/tls"
    "log"
    "net/http"
)

func main() {
    // Gestionnaire simple pour la route principale
    http.HandleFunc("/", func(w http.ResponseWriter, r *http.Request) {
        w.Write([]byte("Bienvenue sur mon serveur sécurisé en HTTPS!"))
    })

// \_ Charger le certificat et la clé
    cert, err := tls.LoadX509KeyPair("server.crt", "server.key")
    if err != nil {
        log.Fatal(err)
```

```
}
  // Configurer TLS
  tlsConfig := &tls.Config{
    Certificates: []tls.Certificate{cert},
  }
  // Configuration du serveur HTTP/HTTPS
  server := &http.Server{
              ":443",
                         // Écoute sur le port HTTPS 443
    Addr:
    TLSConfig: tlsConfig, // Config TLS
    Handler: nil, // Utilise le gestionnaire par défaut
  }
  log.Println("Serveur HTTPS démarré sur https://localhost:443")
  // Démarrer le serveur HTTPS
  err = server.ListenAndServeTLS("", "") // Les certificats sont déjà dans tlsConfig
  if err != nil {
    log.Fatal(err)
  }
}
```

Avant d'exécuter le serveur, assurez-vous que les fichiers de certificat sont dans le même répertoire que votre fichier Go (server.crt et server.key).

Compilez et exécutez votre programme Go :

```
# Exécuter le serveur Go
go run main.go
```

Revision #1 Created 23 September 2024 15:13:17 by Admin Updated 23 September 2024 15:22:15 by Admin