[Line CTF] gotm

문제에 접속하면 아무 것도 뜨질 않아서 소스 코드를 확인해봤다
Go 언어로 작성된 파일이었다.

```
func main() {
    admin := Account{admin_id, admin_pw, true, secret_key}
    acc = append(acc, admin)

    http.HandleFunc("/", root_handler)
    http.HandleFunc("/auth", auth_handler)
    http.HandleFunc("/flag", flag_handler)
    http.HandleFunc("/regist", regist_handler)
    log.Fatal(http.ListenAndServe("0.0.0.0:11000", nil))
}
```

/, /auth, /flag, /regist 모두 접속해봤는데 아무런 변화가 없어서 각 url 별로 해당하는 함수를 확인해보니 /, /flag 는 jwt 토큰이 있어야 했고, /auth, /regist 는 파라미터로 id, pw 가 있어야 했다.

```
func root_handler(w http.ResponseWriter, r *http.Request) {
    token := r.Header.Get("X-Token")
    if token != "" {
        id, _ := jwt_decode(token)
        acc := get_account(id)
        tpl, err := template.New("").Parse("Logged in as " + acc.id)
        if err != nil {
        }
        tpl.Execute(w, &acc)
    } else {

        return
    }
}
```

Jwt 토큰에서 유저 정보를 가져와서 어떤 계정으로 로그인 했는지 확인해준다

```go
func flag_handler(w http.ResponseWriter, r *http.Request) {
    token := r.Header.Get("X-Token")
    if token != "" {
        id, is_admin := jwt_decode(token)
        if is_admin == true {
            p := Resp{true, "Hi " + id + ", flag is " + flag}
            res, err := json.Marshal(p)
            if err != nil {
            }
            w.Write(res)
            return
        } else {
            w.WriteHeader(http.StatusForbidden)
            return
        }
    }
}
```

Admin 권한일 경우 flag 를 출력해준다

```go
func regist_handler(w http.ResponseWriter, r *http.Request) {
    uid := r.FormValue("id")
    upw := r.FormValue("pw")

    if uid == "" || upw == "" {
        return
    }

    if get_account(uid).id != "" {
        w.WriteHeader(http.StatusForbidden)
        return
    }
    if len(acc) > 4 {
        clear_account()
    }
    new_acc := Account{uid, upw, false, secret_key}
    acc = append(acc, new_acc)

    p := Resp{true, ""}
    res, err := json.Marshal(p)
    if err != nil {
    }
    w.Write(res)
    return
}
```

Id, pw 를 받아와서 계정을 새로 만들어서 acc 라는 배열에 추가한다.

```go
func auth_handler(w http.ResponseWriter, r *http.Request) {
    uid := r.FormValue("id")
    upw := r.FormValue("pw")
    if uid == "" || upw == "" {
        return
    }
    if len(acc) > 1024 {
        clear_account()
    }
    user_acc := get_account(uid)
    if user_acc.id != "" && user_acc.pw == upw {
        token, err := jwt_encode(user_acc.id, user_acc.is_admin)
        if err != nil {
            return
        }
        p := TokenResp{true, token}
        res, err := json.Marshal(p)
        if err != nil {
        }
        w.Write(res)
        return
    }
    w.WriteHeader(http.StatusForbidden)
    return
}
```

Id, pw 를 읽어서 이에 맞는 게정을 찾고 그 정보로 jwt 토큰을 만들어준다

나머지 함수들은 jwt 토큰을 만들거나 계정 관리하는 배열을 읽어오거나 청소하는 기능을 하는 함수였다

Admin 아이디로 계정을 만들어서 시도해봤는데 계정 생성할 때 is_admin 을 false 로 해서 만들기 때문에 admin 으로 위장할 수는 없었다. Password 게싱도 해봤는데 되질 않았고 결국 해결 방법을 못 찾고 못 풀었다.