

What is HTTP method?

問題概要

Challenge

×

What is HTTP method ?

100

Web

FLAG = POST + DELETE + PUT + PATCH →
<http://192.168.1.3:8001>

Flag

Submit

ねらい

- HTTPメソッドについて知ってもらう
- curlコマンドから実際に扱ってもらう

アプローチ

- まず, 問題文のパズルを解く

FLAG = POST + DELETE + PUT + PATCH →
<http://192.168.1.3:8001>

→ 対象のIP:Portに対して, HTTPメソッドを用いる.

Flagの一部が帰ってくるので, それぞれの結果を並び替える

アプローチ

- curlコマンド

指定なし = GET

```
$ curl http://192.168.1.3:8001
```

----> This is GET Method

-X : メソッド指定

```
$ curl -X POST http://192.168.1.3:8001
```

----> http-ctf{THIS_

```
$ curl -X DELETE http://192.168.1.3:8001
```

----> IS_

```
$ curl -X PUT http://192.168.1.3:8001
```

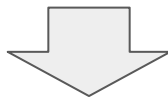
----> HTTP_

```
$ curl -X PATCH http://192.168.1.3:8001
```

----> METHOD}

答え

POST + DELETE + PUT + PATCH
↓ ↓ ↓ ↓
http-ctf{THIS_ + IS_ + HTTP_ + METHOD}



http-ctf{THIS_IS_HTTP_METHOD}

その他

- サーバーサイドプログラム
 - Go
- コンテナ化して問題サーバ上で実行
 - Dockerfile

```
1 FROM golang
2
3 COPY http-method.go .
4
5 CMD go run http-method.go
6
7 EXPOSE 8080
```

```
1 package main
2
3 import (
4     "fmt"
5     "net/http"
6 )
7
8 func handler(w http.ResponseWriter, r *http.Request) {
9     switch r.Method {
10     case "GET":
11         fmt.Fprintf(w, "This is GET Method")
12     case "POST":
13         fmt.Fprintf(w, "http-ctf{THIS_}")
14     case "DELETE":
15         fmt.Fprintf(w, "IS_")
16     case "PUT":
17         fmt.Fprintf(w, "HTTP_")
18     case "PATCH":
19         fmt.Fprintf(w, "METHOD}")
20     default:
21         fmt.Fprintf(w, "Other HTTP Method")
22     }
23 }
24
25 func main() {
26     http.HandleFunc("/", handler)
27     http.ListenAndServe(":8080", nil)
28 }
```