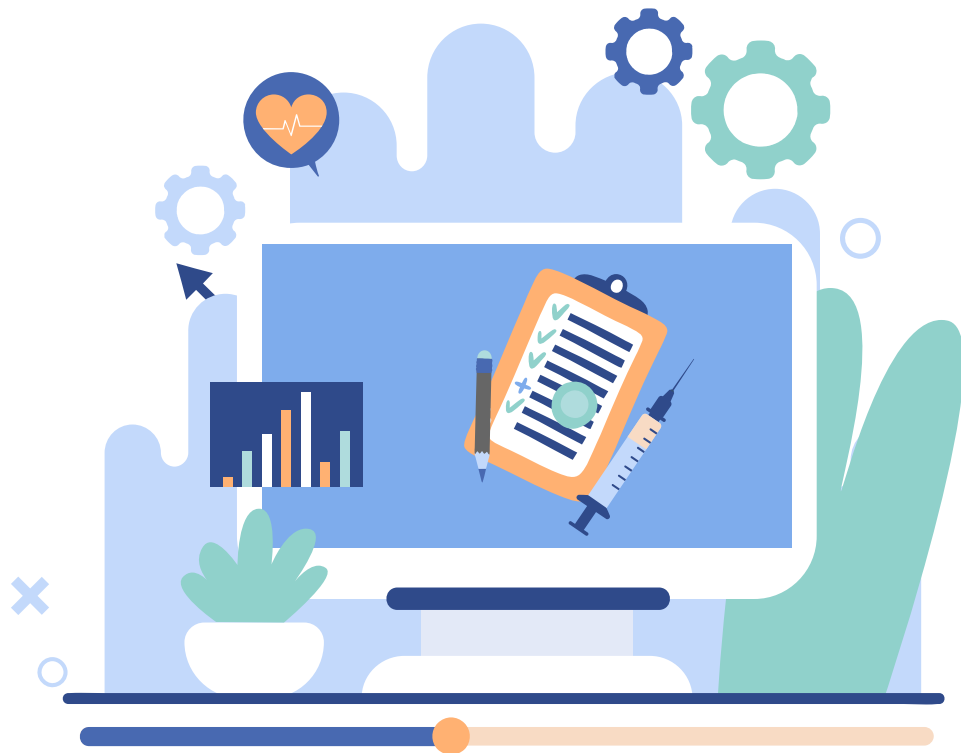




Formation HTTP

Siffrein
Seaweedbrain





SOMMAIRE

01. HTTP, pour quoi faire ?

02. HTTP, en détail

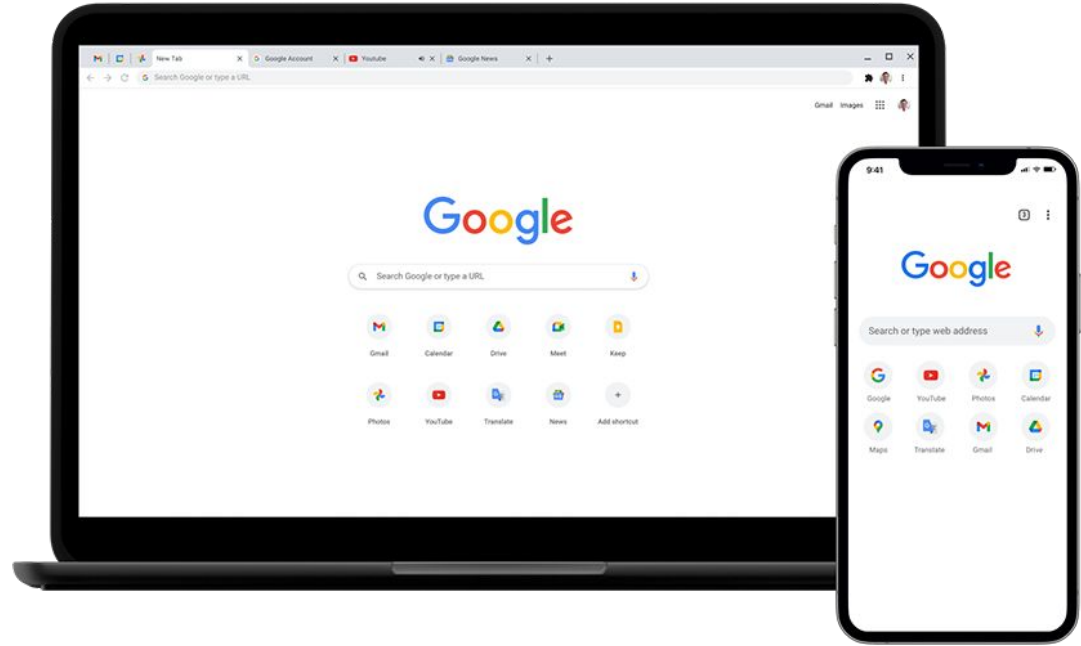
03. HTTP, les versions

04. HTTP, en pratique

01.

HTTP, Pour quoi faire ?

Le coeur du web



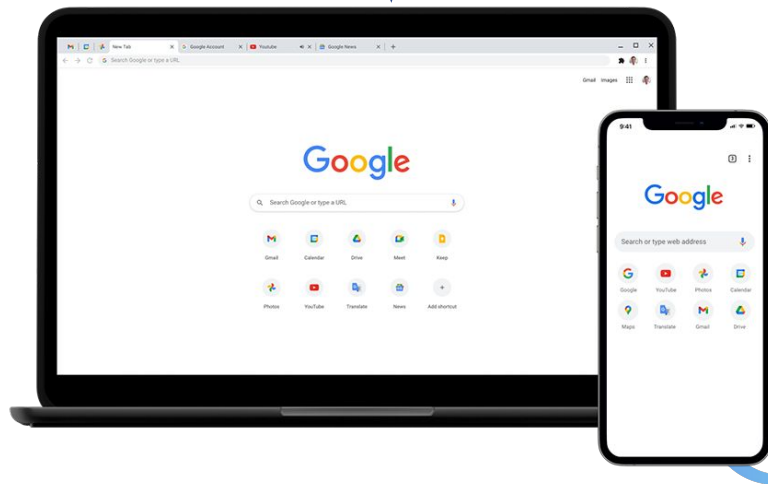
01.

HTTP, Pour quoi faire ?

Création de l'HTML

```
1 <!DOCTYPE html>
2 <html>
3 <head>
4   <title>Un nouveau langage : HTML</title>
5 </head>
6 </html>
```

?????



01.

HTTP, Pour quoi faire ?

Création des navigateurs

```
1 <!DOCTYPE html>
2 <html>
3 <head>
4   <title>Un nouveau langage : HTML</title>
5 </head>
6 </html>
```



Safari
Apple



Firefox
Mozilla



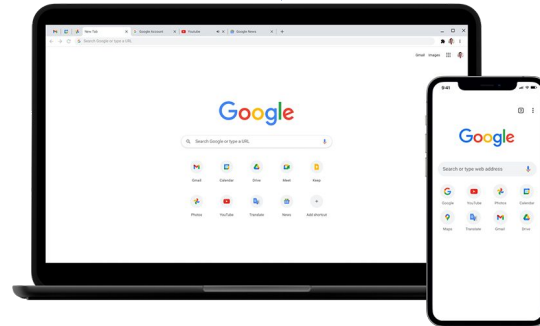
Chrome
Google



Edge new
Microsoft



Opera
Opera Software

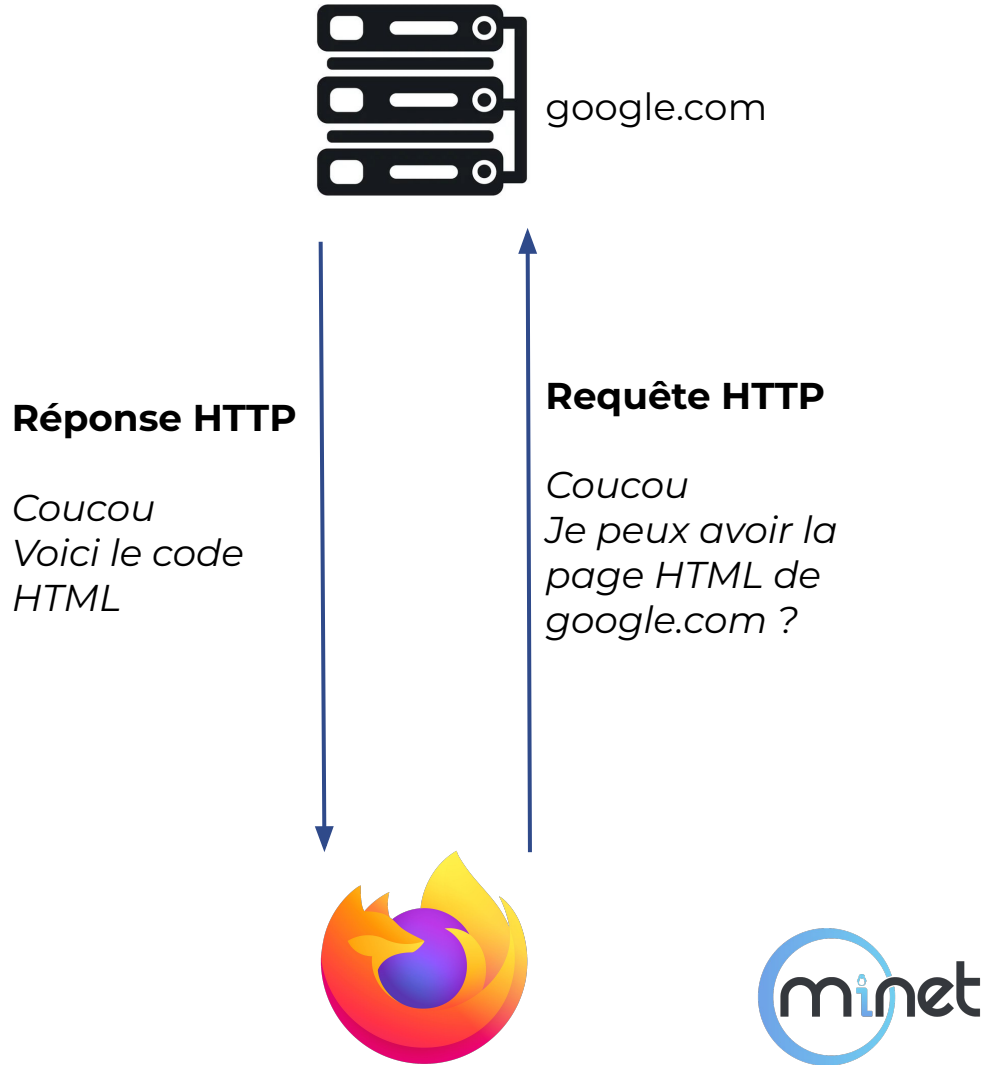


01.

HTTP,

Pour quoi faire ?

Création de l'HTTP



Bon, HTTP c koi ?

Basé sur TCP

Protocole applicatif, chiffré ou non



Communication serveur/client Stateless

Paires de requêtes-réponses entièrement indépendante



Le serveur ne fait qu'écouter

Seul le client envoie des requêtes



HTTP n'est pas,



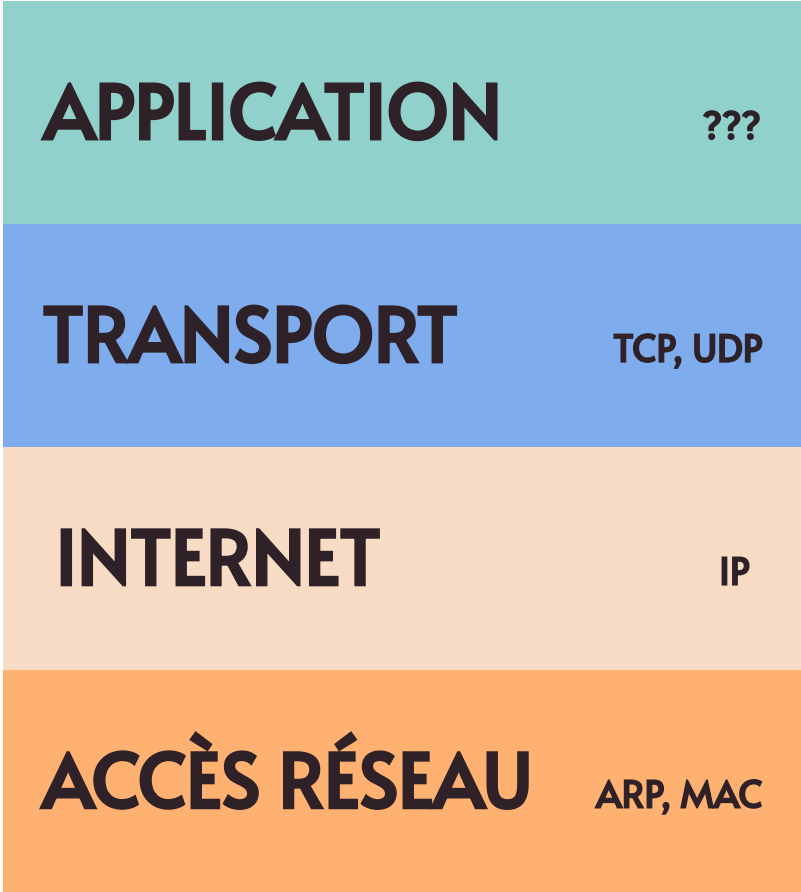
Un protocole de transport

HTTP ne s'occupe pas de transporter les données, c'est TCP qui s'en occupe



Dépendant de TCP

On verra plus tard les autres cas



APPLICATION

???

TRANSPORT

TCP, UDP

INTERNET

IP

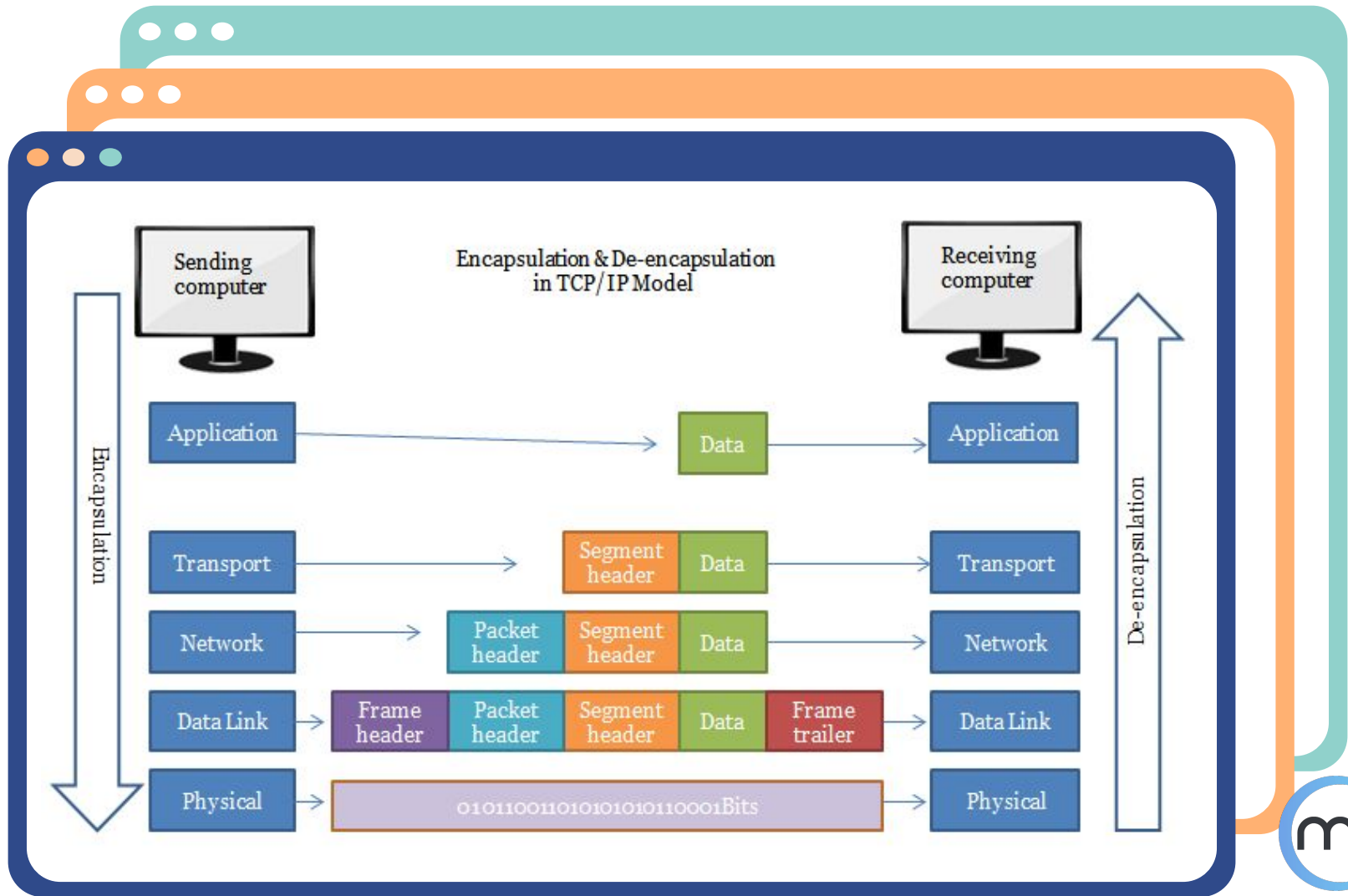
ACCÈS RÉSEAU

ARP, MAC

02.

HTTP en détail

Le modèle TCP/IP



Les requêtes HTTP

Première ligne de la requête :

[Méthode] [Chemin] HTTP/[Version]

GET adh6.minet.net/dashboard HTTP/1.1

Les méthodes :

GET

/pet/{petId} Find pet by ID

PUT

/pet Update an existing pet

DELETE

/pet/{petId} Deletes a pet

POST

/pet/{petId}/uploadImage uploads an image

Les requêtes HTTP

HEADER : Paramètres nécessaires à la requêtes (et plus si affinités)

Méthode, chemin, user agent, content-type, cookies, keys, host, referrer, et n'importe quel autre paramètres

BODY (optionnel) : N'importe quel paramètre (essentiellement pour les requêtes POST ou PUT).

Comment faire une requête GET ?

<https://youtube.com/videos?id=76>

Protocole

Nom de domaine

Chemin

Paramètres



Les réponses HTTP

Première ligne de la réponse :

HTTP/[Version] [Code] [Status]

HTTP/1.1 200 OK

HTTP/1.1 404 Not Found

HTTP/1.1 502 Internal Server Error

1XX Information		4XX Client (Continue)	
100	Continue	407	Proxy Authentication Required
101	Switching Protocols	408	Request Timeout
102	Processing	409	Conflict
103	Early Hints	410	Gone
		411	Length Required
2XX Success		412	Precondition Failed
200	OK	413	Payload Too Large
201	Created	414	URI Too Large
202	Accepted	415	Unsupported Media Type
203	Non-Authoritative Information	416	Range Not Satisfiable
205	Reset Content	417	Exception Failed
206	Partial Content	418	I'm a teapot
207	Multi-Status (WebDAV)	421	Misdirected Request
208	Already Reported (WebDAV)	422	Unprocessable Entity (WebDAV)
226	IM Used (HTTP Delta Encoding)	423	Locked (WebDAV)
		424	Failed Dependency (WebDAV)
3XX Redirection		425	Too Early
300	Multiple Choices	426	Upgrade Required
301	Moved Permanently	428	Precondition Required
302	Found	429	Too Many Requests
303	See Other	431	Request Header Fields Too Large
304	Not Modified	451	Unavailable for Legal Reasons
305	Use Proxy	499	Client Closed Request
306	Unused		
307	Temporary Redirect	5XX Server Error Responses	
308	Permanent Redirect	500	Internal Server Error
4XX Client Error		501	Not Implemented
400	Bad Request	502	Bad Gateway
401	Unauthorized	503	Service Unavailable
402	Payment Required	504	Gateway Timeout
403	Forbidden	505	HTTP Version Not Supported
404	Not Found	507	Insufficient Storage (WebDAV)
405	Method Not Allowed	508	Loop Detected (WebDAV)
406	Not Acceptable	510	Not Extended
Compiled by Ivan Tay.		511	Network Authentication Required
		599	Network Connect Timeout Error

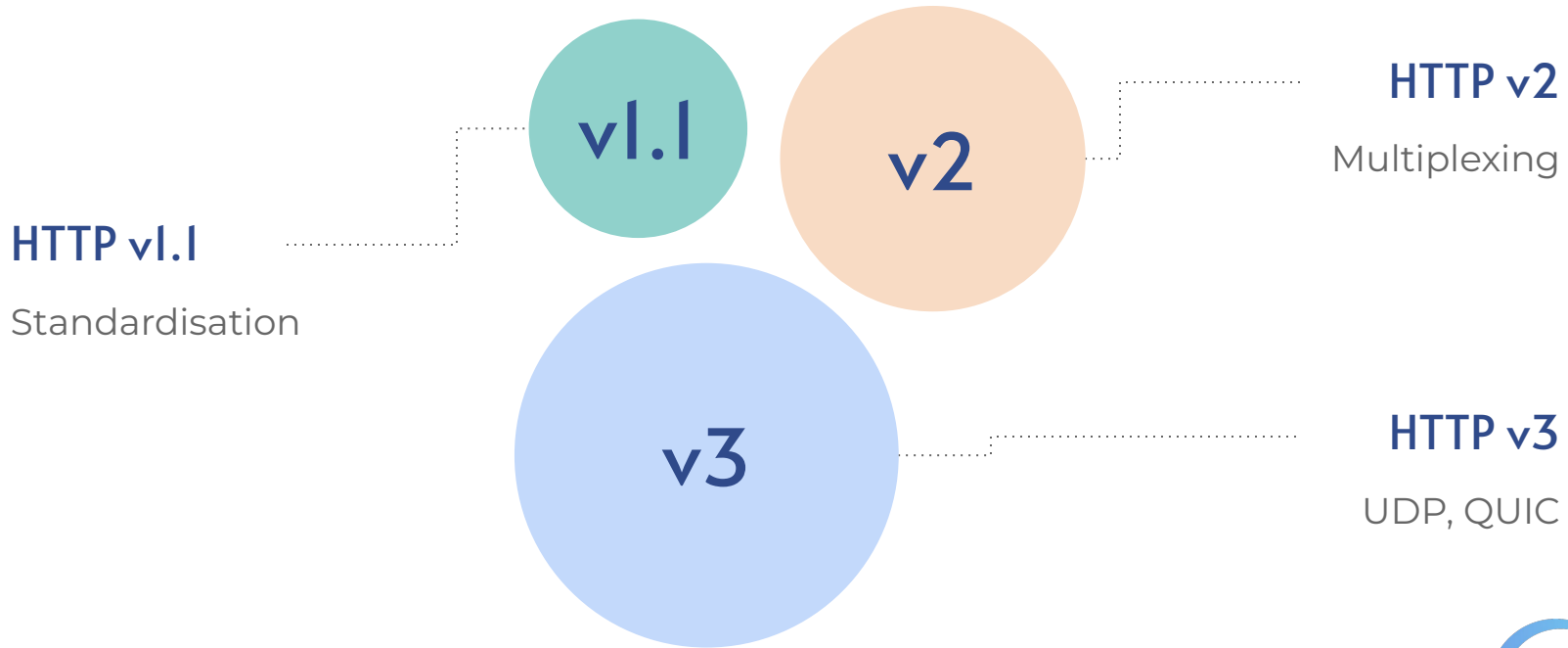
Les réponses HTTP

HEADER : Paramètres nécessaires à la requêtes (et plus si affinités)

Méthode, chemin, user agent, content-type, cookies, keys, host, referrer, et n'importe quel autre paramètres

BODY (optionnel) : contenu de la réponse

Les versions HTTP



L'HTTPS

Preuve d'authenticité du
propriétaire du site

Certificat SSL/TLS

Clé publique du site web

**Chiffrement
asymétrique**

https://pornhub.com/*****

**Chiffrement de la
communication**

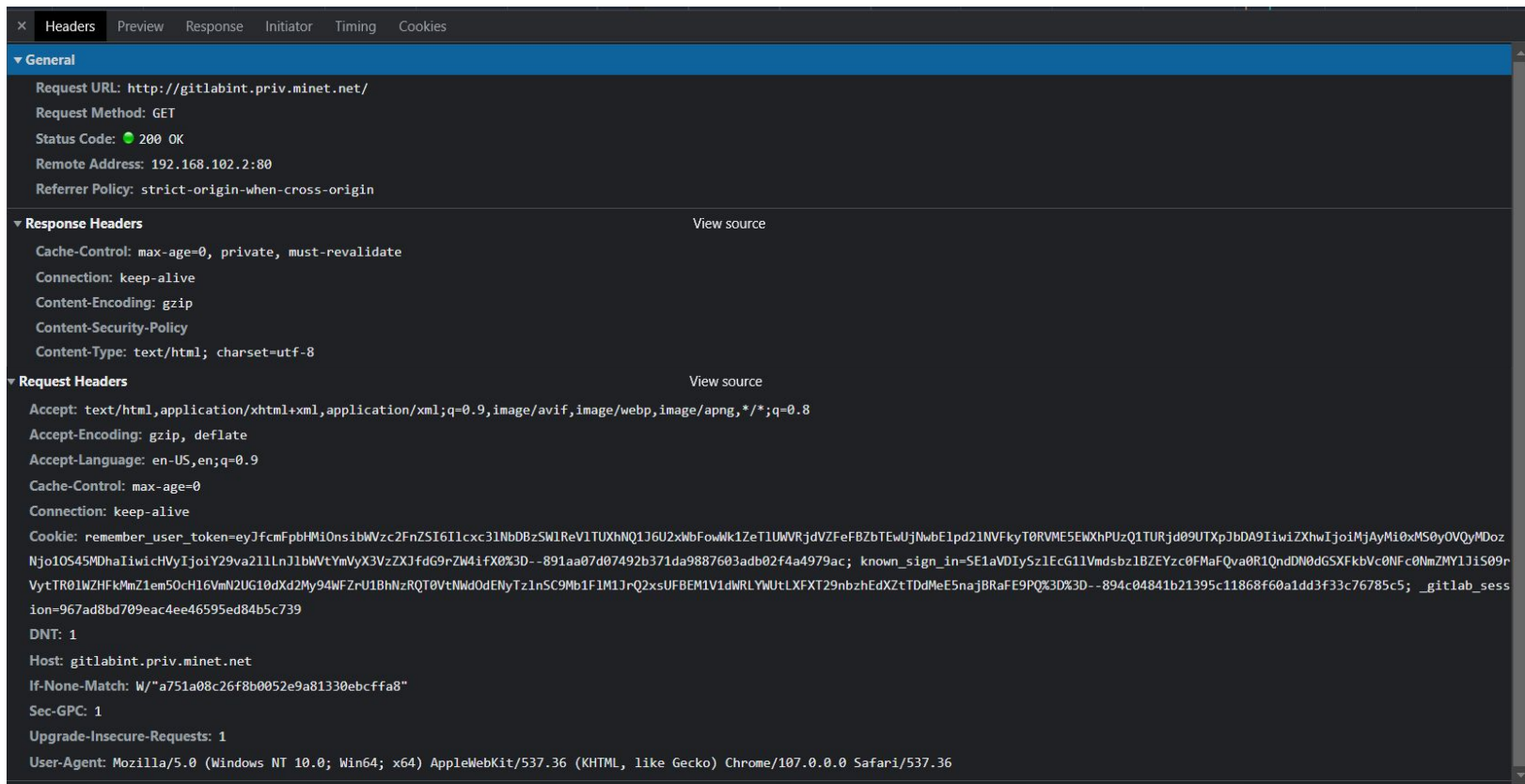
Rend caduque les attaque de
man in the middle

Empêche le MITM

Exemple: Requête HTTP [Wireshark]

ip.addr == 192.168.102.2				
No.	Source	Destination	Protocol	Info
1	10.0.1.102	192.168.102.2	TCP	50523 → 80 [SYN] Seq=0 Win=642
2	192.168.102.2	10.0.1.102	TCP	80 → 50523 [SYN, ACK] Seq=0 Ac
3	10.0.1.102	192.168.102.2	TCP	50523 → 80 [ACK] Seq=1 Ack=1 W
4	10.0.1.102	192.168.102.2	HTTP	GET / HTTP/1.1
5	192.168.102.2	10.0.1.102	TCP	80 → 50523 [ACK] Seq=1 Ack=89
6	192.168.102.2	10.0.1.102	HTTP	HTTP/1.1 302 Found (text/html
7	10.0.1.102	192.168.102.2	TCP	50523 → 80 [FIN, ACK] Seq=89 A
8	192.168.102.2	10.0.1.102	TCP	80 → 50523 [FIN, ACK] Seq=786
9	10.0.1.102	192.168.102.2	TCP	50523 → 80 [ACK] Seq=90 Ack=78

Exemple: Requête HTTP [DevTools]



The screenshot displays the Chrome DevTools Network tab with the 'Headers' sub-tab selected. It shows a successful GET request to `http://gitlabint.priv.minet.net/` with a status of 200 OK. The response headers indicate a gzipped HTML document. The request headers show standard browser defaults for a web browser, including `Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8` and a `Cookie` containing session tokens.

General

- Request URL: `http://gitlabint.priv.minet.net/`
- Request Method: GET
- Status Code: 200 OK
- Remote Address: `192.168.102.2:80`
- Referrer Policy: `strict-origin-when-cross-origin`

Response Headers [View source](#)

- Cache-Control: `max-age=0, private, must-revalidate`
- Connection: `keep-alive`
- Content-Encoding: `gzip`
- Content-Security-Policy
- Content-Type: `text/html; charset=utf-8`

Request Headers [View source](#)

- Accept: `text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8`
- Accept-Encoding: `gzip, deflate`
- Accept-Language: `en-US,en;q=0.9`
- Cache-Control: `max-age=0`
- Connection: `keep-alive`
- Cookie: `remember_user_token=eyJfcmlpbnVzc2FnZSI6I1cx31NbDBzSW1ReV1TUXhNQ1J6U2xwFowMk1ZeTlUMVRjdVZFeFBZbTEwUjNwbE1pd2lNVFkyT0RVME5ENXhPUzQ1TURjd09UTXpJbDA9IiwiaXhwIjo1MjAyMi0xMS0yOVQyMDozNjo1OS45MDhaIiwiaXhwIjoiY29va211LnJlbWVtYmVyX3VzXjFfdG9rZW4ifX0%3D--891aa07d07492b371da9887603adb02f4a4979ac; known_sign_in=SE1aVDIySzlEcG1lVmdsbz1BZEYzc0FMaFQva0R1QndDN0dGSXfbVc0Nfc0NmZMY1JiS09rVytTR0lWZHFkMmZ1em50cH16VmN2UG10dXd2My94WFZrU1BhNzRQT0VtNWdOdElyTz1nSC9Mb1F1M1JrQ2xsUFBEM1V1dWRLYWUtLXFXt29nbzhEdXZtTDdMeE5naJBraFE9PQ%3D%3D--894c04841b21395c11868f60a1dd3f33c76785c5; _gitlab_session=967ad8bd709eac4ee46595ed84b5c739`
- DNT: 1
- Host: `gitlabint.priv.minet.net`
- If-None-Match: `W/"a751a08c26f8b0052e9a81330ebcffa8"`
- Sec-GPC: 1
- Upgrade-Insecure-Requests: 1
- User-Agent: `Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/107.0.0.0 Safari/537.36`



Exemple: Réponse HTTP [cURL]

```
> curl -i --head http://gitlabint.priv.minet.net/users/sign_in
HTTP/1.1 200 OK
Cache-Control: max-age=0, private, must-revalidate
Connection: keep-alive
Content-Encoding: gzip
Content-Security-Policy:
Content-Type: text/html; charset=utf-8

<!DOCTYPE html>
<html class="devise-layout-html">
<head prefix="og: http://ogp.me/ns#">
<meta charset="utf-8">
<title>Sign in · GitLab</title>
<link rel="preload" href="/assets/application_utilities-
bc405f753d3fc57a7c9cf1aafc04c07ef4c758eced90f0d06275492f20f8fcbf.css" as="style" type="text/css">
<link rel="preload" href="/assets/application-
1caf2b894e48f649fcfd6b20de756e07ce64c1a756b9a20ff4505caeffa1a361.css" as="style" type="text/css">
<link rel="preload" href="/assets/highlight/themes/white-
69223daaf028b2b7665d7b9256b5d2f2703ce7f83cb2b7c86479acaf55a59001.css" as="style" type="text/css">
# .....
```



► GET https://backprox.minet.net/api/1.0.0/vm?search=Comment valider sa 2A

▼ GET

Scheme: https

Host: backprox.minet.net

Filename: /api/1.0.0/vm

search: Comment valider sa 2A

Adresse: 157.159.40.103:443

HTML

Brut ☒

```
1  <!DOCTYPE html><html lang="en" class="h-100"><head>
2    <meta charset="utf-8">
3    <title>{{ title }}</title>
4    <base href="/">
5    <meta name="viewport" content="width=device-width, initial-scale=1">
6    <link rel="icon" href="favicon.ico">
7    <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/js/bootstrap.bundle.min.js">
8    <script src="https://maxcdn.bootstrapcdn.com/bootstrap/4.1.0/js/bootstrap.min.js"></script>
9    <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.3.1/jquery.min.js"></script>
10   <style>:root{--blue:#007bff;--indigo:#6610f2;--purple:#6f42c1;--pink:#e83e8c;--red:#dc3545;--
11   <body class="h-100">
12   <app-root></app-root>
13   <script src="runtime.js" type="module"></script><script src="polyfills.js" type="module"></s
14
15   </body></html>
```



▼ POST

Scheme: https

Host: backprox.minet.net

Filename: /api/1.0.0/dns

Adresse: 157.159.40.103:443

▶ En-têtes Cookies Requête Réponse

🔍 Filtrer les paramètres de la requête

JSON

entry: "estebanEtNicolasMangezCeRatio"

ip: "157.159.195.70"

▶ En-têtes Cookies Requête Réponse Délais Trace de la pile Sécurité

🔍 Filtrer les propriétés

JSON Brut ☐

detail: "The server encountered an internal error and was unable to complete your request. Either the server is overloaded or there is an error in the application."

status: 500

title: "Internal Server Error"

type: "about:blank"

Un seul mot pour la fin

<https://http.cat>