Infrastructure : Réseau 2

DNS, URL, Proxy, VPN, Firewall, NAT, Quézako ?

Sommaire

Objectif

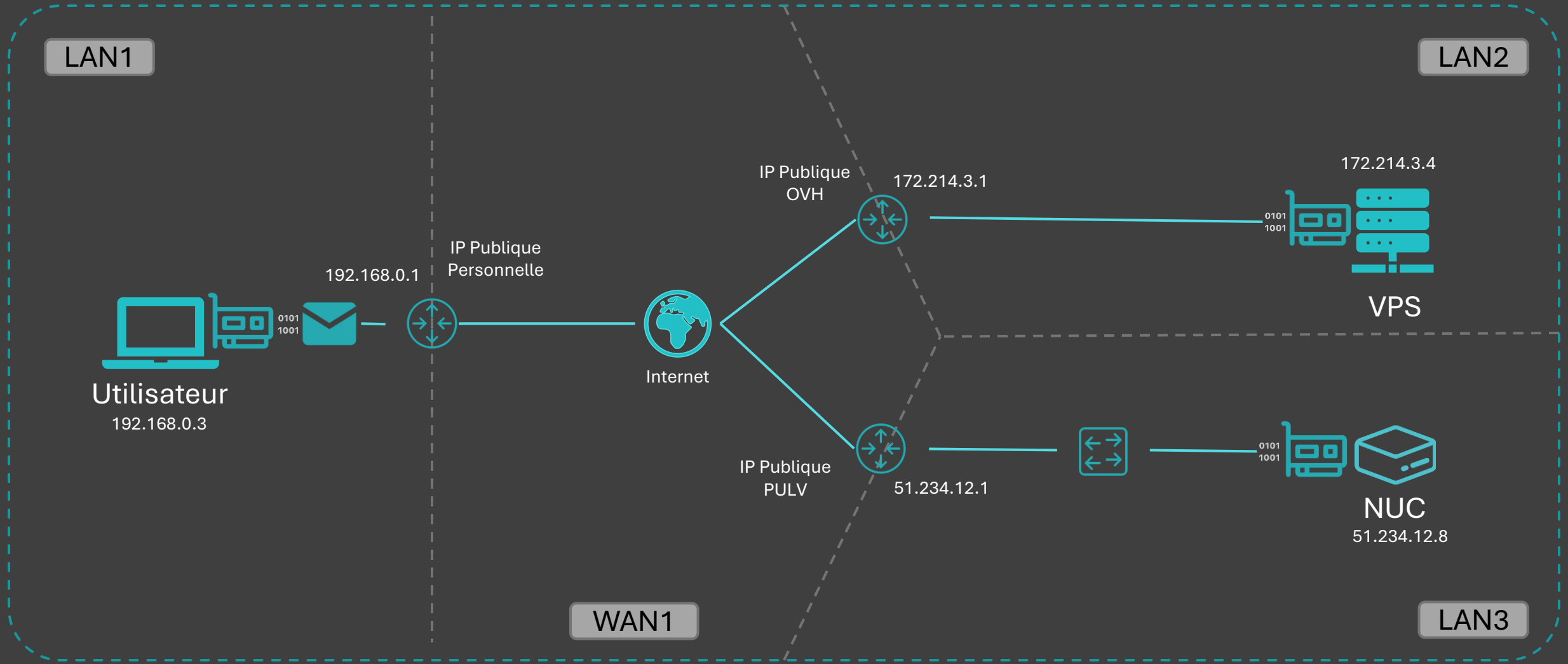
Zoomer sur des éléments plus spécifiques d’un réseau.

On reprend ce qu’on a vu la dernière fois !

I. Rappels

Quelques concepts vus la dernière fois (~5 min)

El - diagramme



DaVinciCode

Rappels & Mind Map

Typologie

	Nom	Etendue	Connexion
WAN	Wide Area Network	Pays / Planétaire	Ethernet / Fibre / Satellites
MAN	Metropolitan Area Network	Ville	Ethernet / Fibre
LAN	Local Area Network	Bâtiment	Ethernet / Wifi
VLAN	Virtual Local Area Network	Bâtiment	Ethernet / Wifi
PAN	Personal Area Network	Personne	WiFi / Bluetooth

NIC : Network Interface Controller

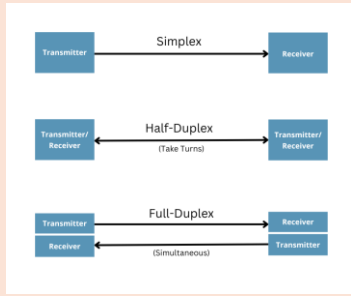
- Transmettre par Ethernet / Wifi / Bluetooth
- Connecté par :
  - PCIe : Peripheral Component Interconnect Express
  - USB

MAC : Media Access Control

- 2^48 possibilités (fixe)

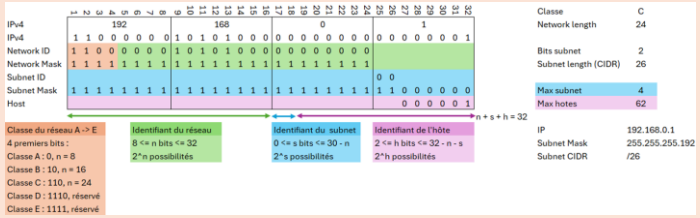
Transmission

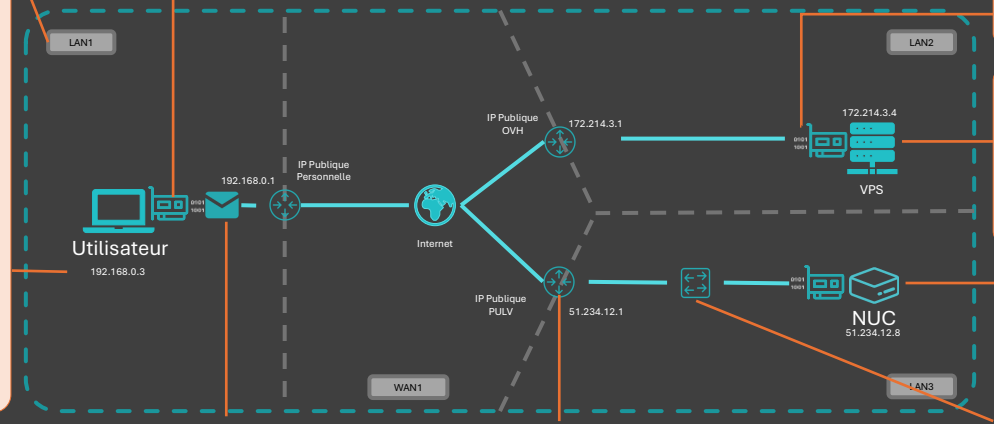
- Electromagnétique
  - Modulation
  - Phase : PSK, QPSK, QAM
  - Fréquence : FSK, OFDM
  - Amplitude : ASK, QAM
- Filaire
  - Lumineux
  - Electrique : 0V ou 5V



IP : Internet Protocol

- IPv4 : 32 bits / 4 octets (192.168.1.22)
- IPv6 :128 bits / 16 octets (fe80::cb9b:3a3:b3f0:8bb4)
- Publique (routeur) ou privée (locale)
- Manuellement ou automatiquement avec DHCP
- Subnet :





VPS : Virtual Private Server

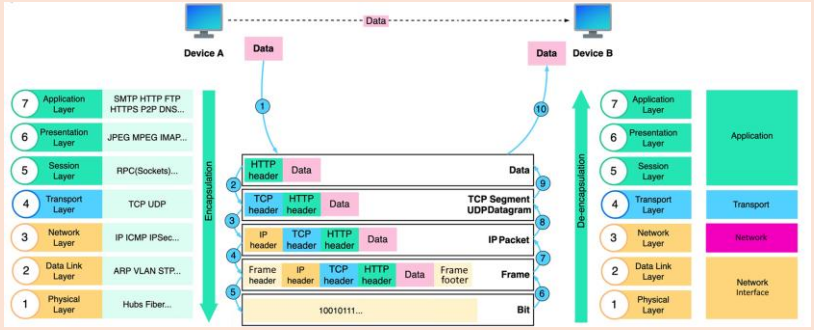
- Fragment d'un serveur plus puissant
- Machine virtuelle
- Contrôlée par une entreprise (Ex : OVH)

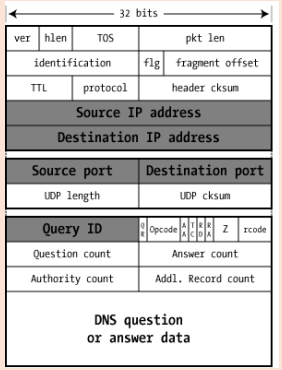
NUC : Next Unit of Computing

- Ordinateur compact
- Haute performance

OSI : Open Systems Interconnection / TCP : Transmission Control Protocol

PDU : Protocol Data Unit





Routeur

- OSI niveau 3
- WLAN / LAN / VLAN
- Redirection selon les adresses MAC / IP
- Utilise une table / protocole ARP

- 3 types : Core / Edge / Virtuel

Switch

- OSI niveau 2 / 3
- LAN / VLAN
- Utilise une table d'adressage MAC

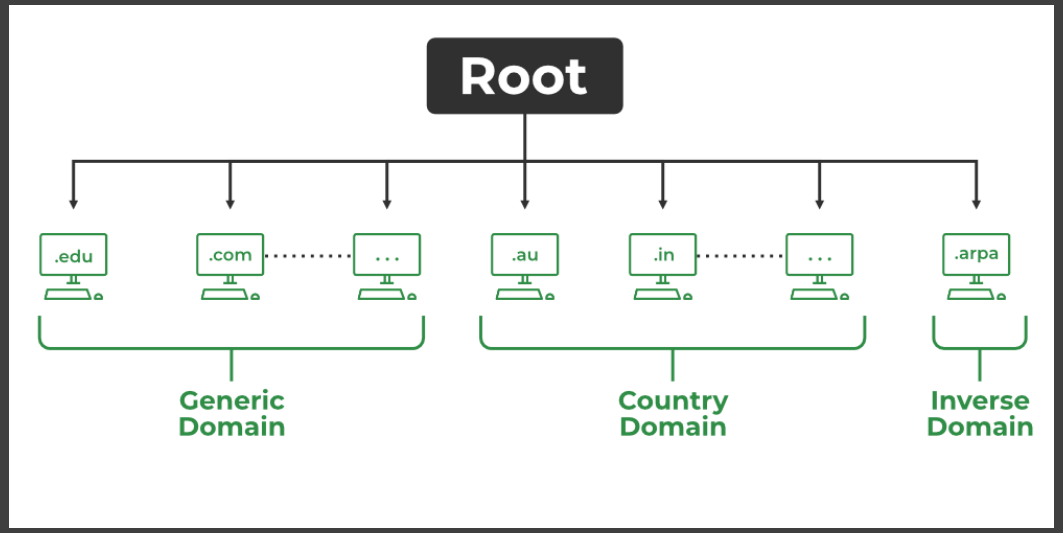
- 2 types :
  - Unmanaged (Layer 2) : MAC
  - Managed (Layer 3) : MAC + IP / VLAN

II. DNS

C’est plus simple de retenir 200 IP avec des mots.

Nom de domaine

- Qu’est ce que c’est ?
  - Remplacer une IP par une chaine de caractères
  - Identifier une machine sur un réseau
  - Exemple : google.com

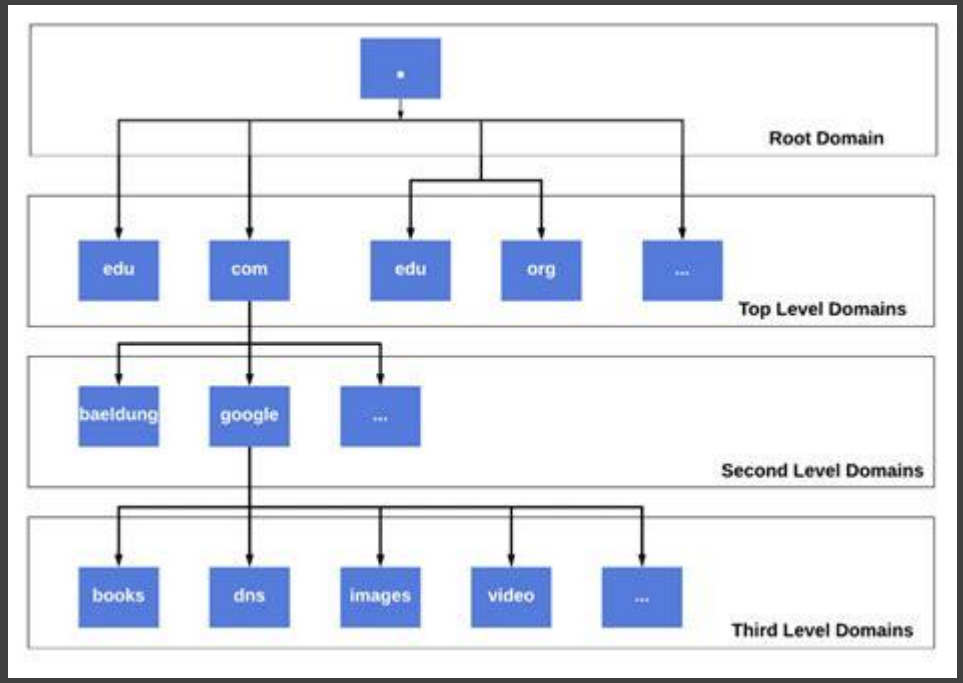


- 2 parties :
  - Top Level Domain (TLD) : .com / .fr
    - Peuvent être spécifiques à des pays
    - Spécifique à des utilisations : .local / .test
  - Second Level Domain (SLD) : davincicode / google

- Un nom de domaine est acheté pour garantir :
  - Son unicité
  - Sa sécurité : certificats SSL, authenticité

Labels & sous domaines

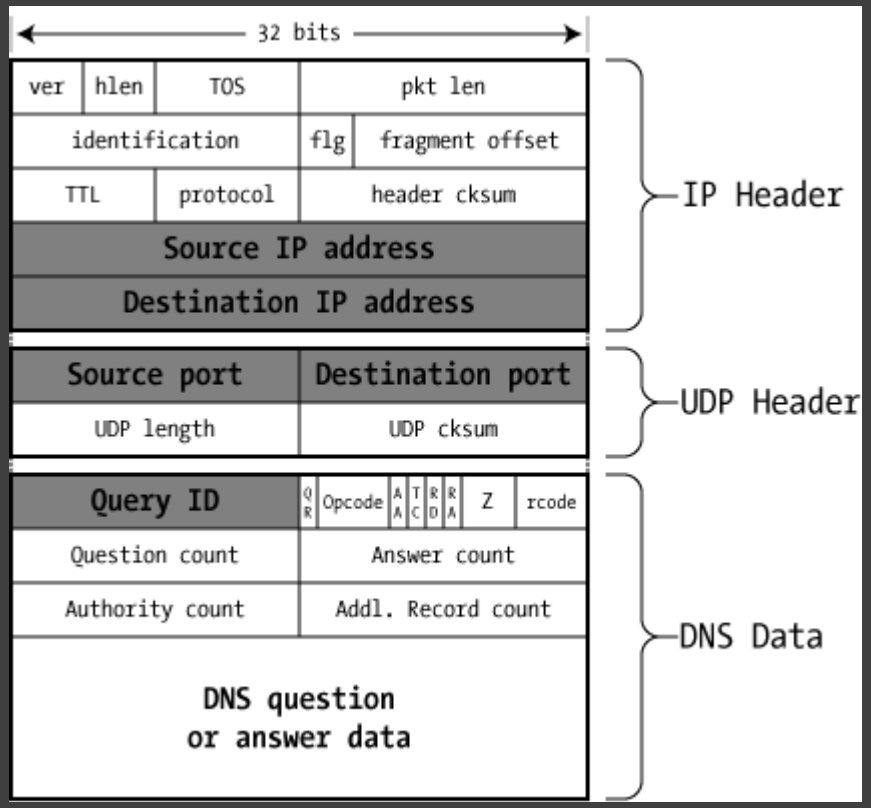
- Qu’est ce que c’est ?
  - Extension des noms de domaines
  - abc.davincicode.fr
  - Séparation des utilisations :
    - mail.davincicode.fr
    - ctf.davincicode.fr
    - …



DNS : Domain Name System

- Qu’est ce que c’est :
  - Serveur + Protocole
  - Annulaire des noms de domaines
  - Remplacer : davincicode.fr > 123.145.167.189

- Exemple de serveurs DNS
  - Publique :
    - Google : 8.8.8.8
    - Cloudflare : 1.1.1.1
  - Privé :
    - Votre box / routeur
    - Votre propre ordinateur :
      - Windows : C:\Windows\System32\drivers\etc\hosts
      - Linux : /etc/hosts



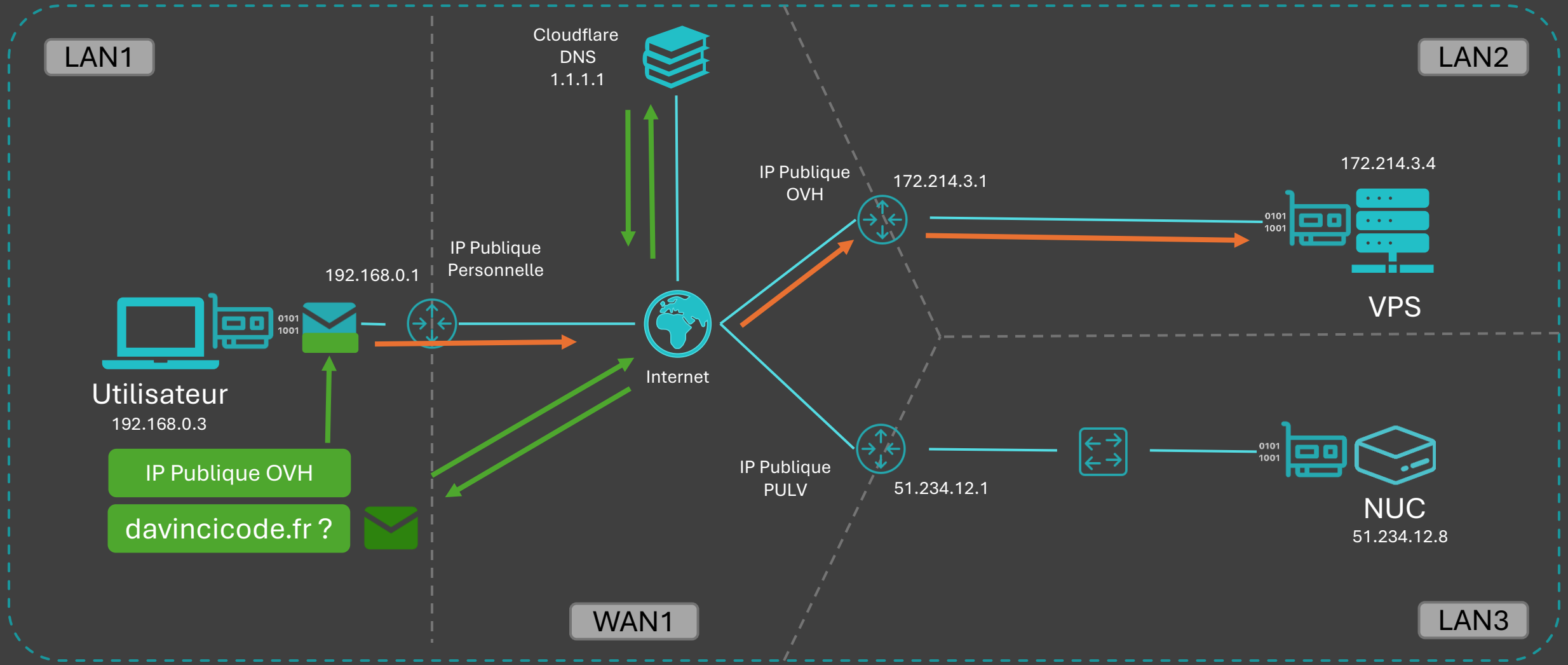
DNS Records

- Qu’est ce que c’est :
  - Ligne dans un serveur DNS : code + domaine + remplacement
  - Définis comment et par quoi un nom de domaine doit être remplacé
    - Ex : IP, nom de domaine, données brutes

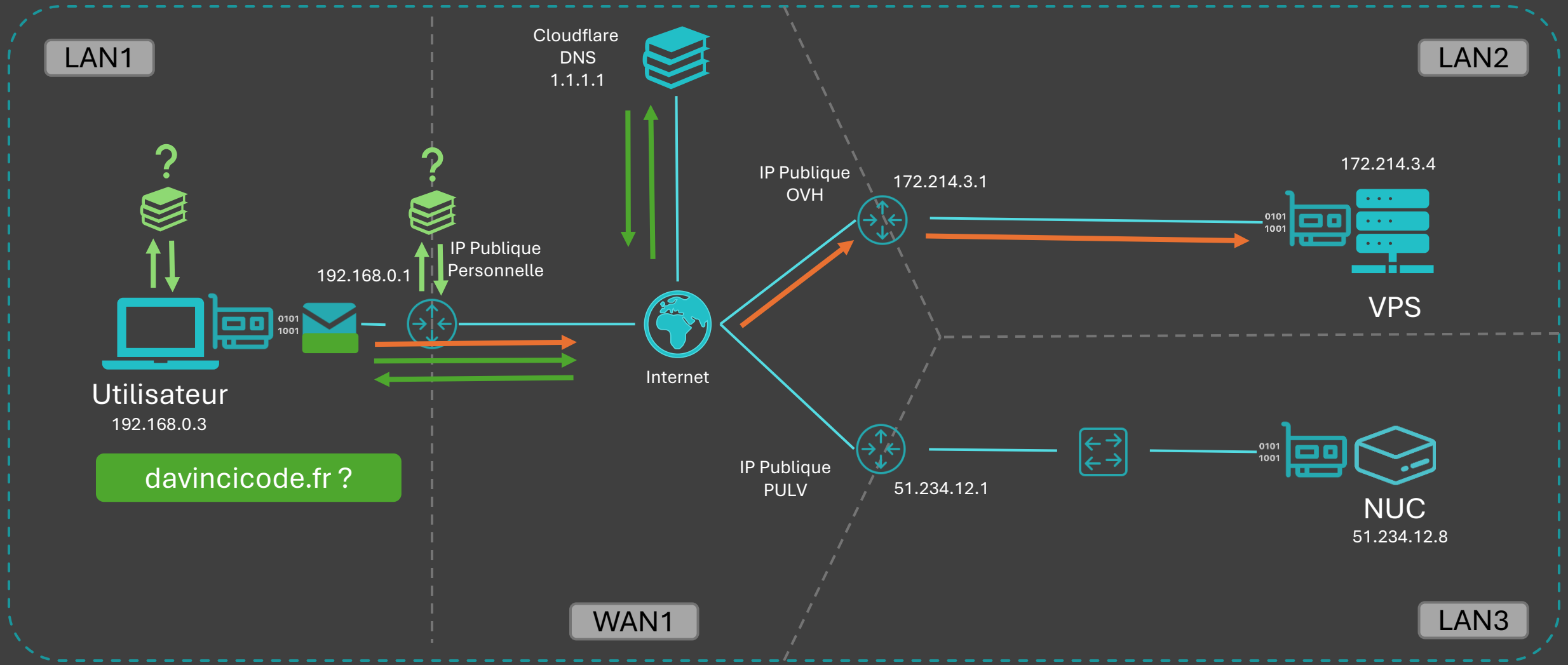
- Code :

Code	Entrée	Sortie
A	davincicode.fr	IPv4 123.145.167.189
AAAA	davincicode.fr	IPv6 ::ffff:7b91:a7bd
CNAME	test.davincicode.fr	davincicode.fr ou IPv4
MX	mail.davincicode.fr	Liste de serveurs mails
TXT		Texte prédéfini

Exemple avec serveur DNS externe



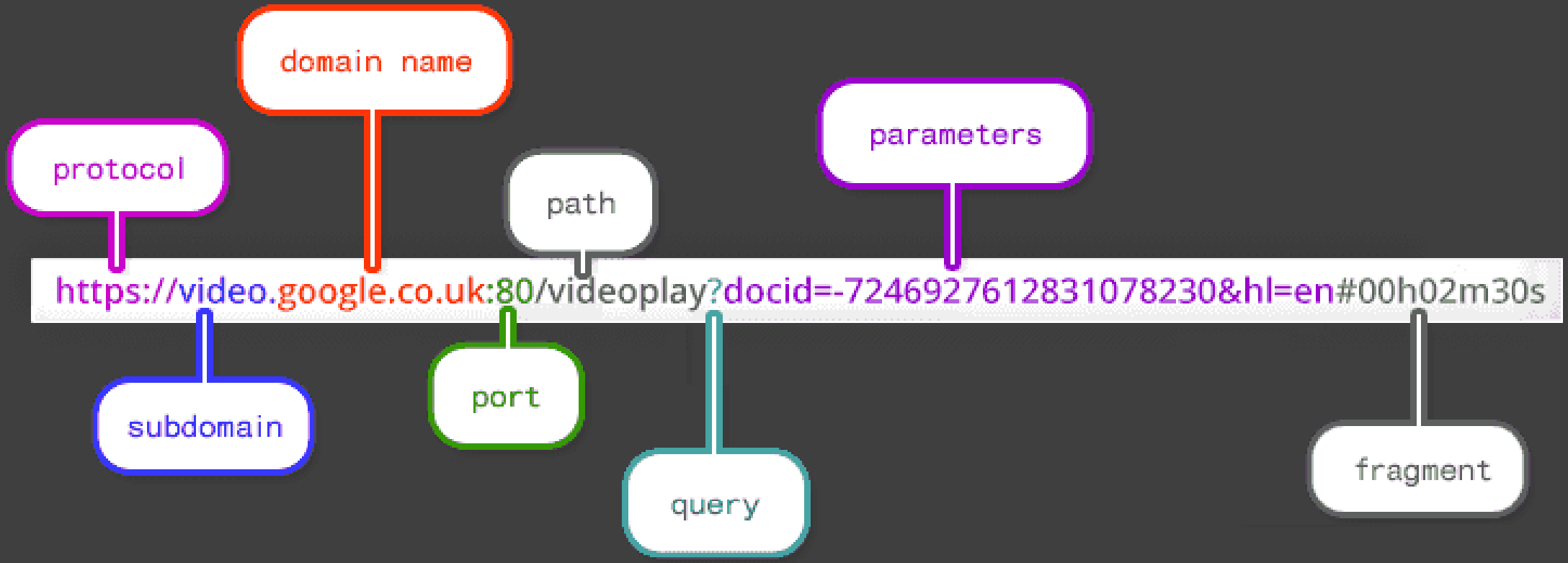
Exemple avec serveur DNS interne



III. URL

386,9 millions de noms de domaine enregistrés fin 2025

URL : Uniform Resource Locator



Pas seulement pour les sites internet (http/https), aussi pour d'autres protocoles !

URI / URC / URN

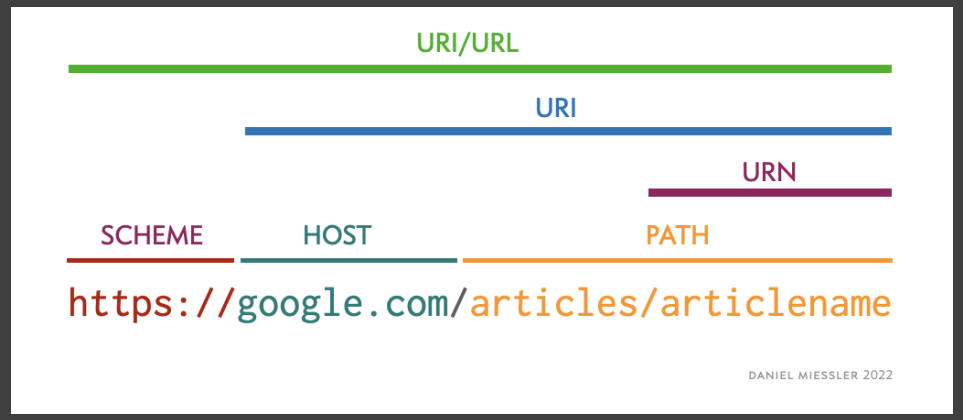
- URI : Uniform Resource Identifier
  - Séquence de charactères pour accéder à une ressource
  - URI = scheme ":" ["//" authority] path ["?" query] ["#" fragment]

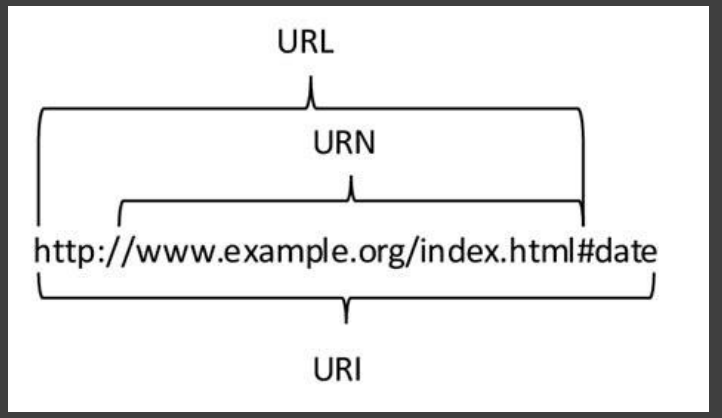
- URL : Uniform Resource Locator
  - URI avec une information sur le réseau (proto://user@host:port/…)

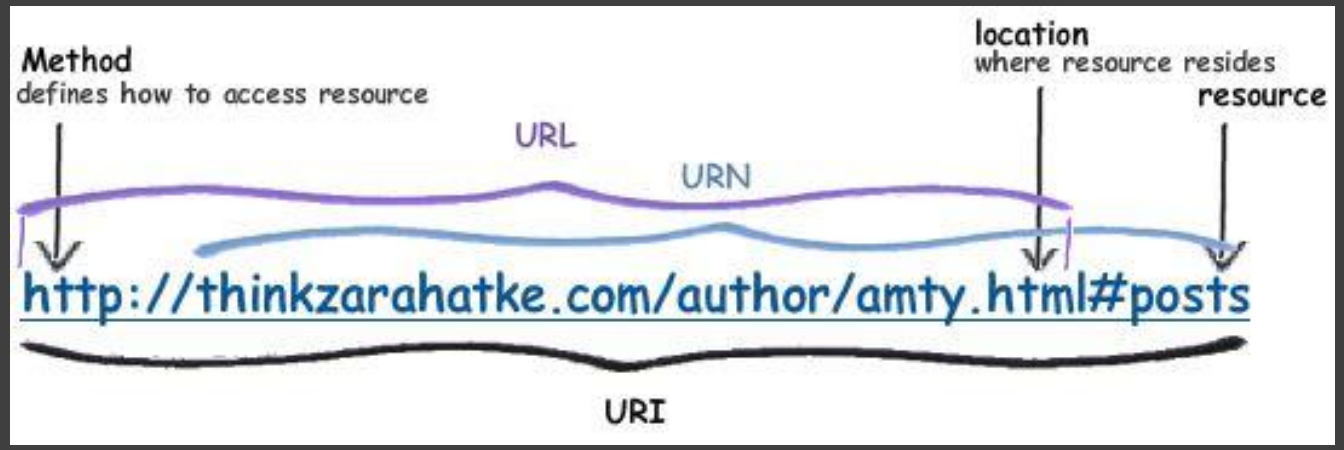
- URN : Uniform Resource Name
  - URI sans information de localisation

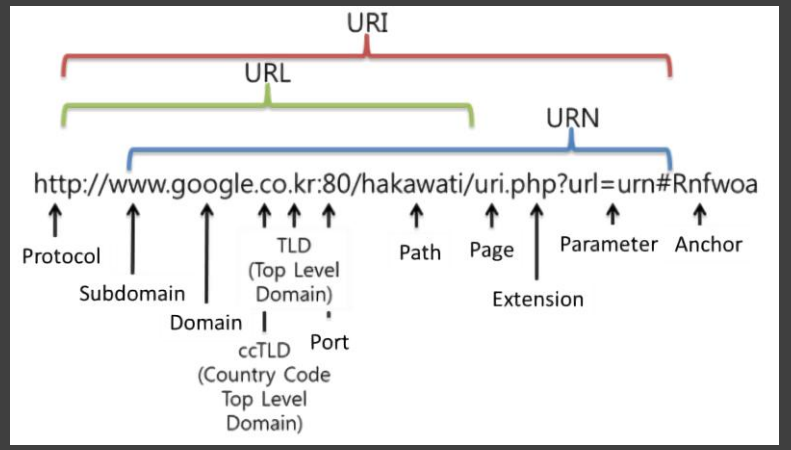
- URC : Uniform Resource Citation
  - Métadonnées d'un URI
  - Exemple, auteur, date de publication, …

Quel diagramme est correct ?

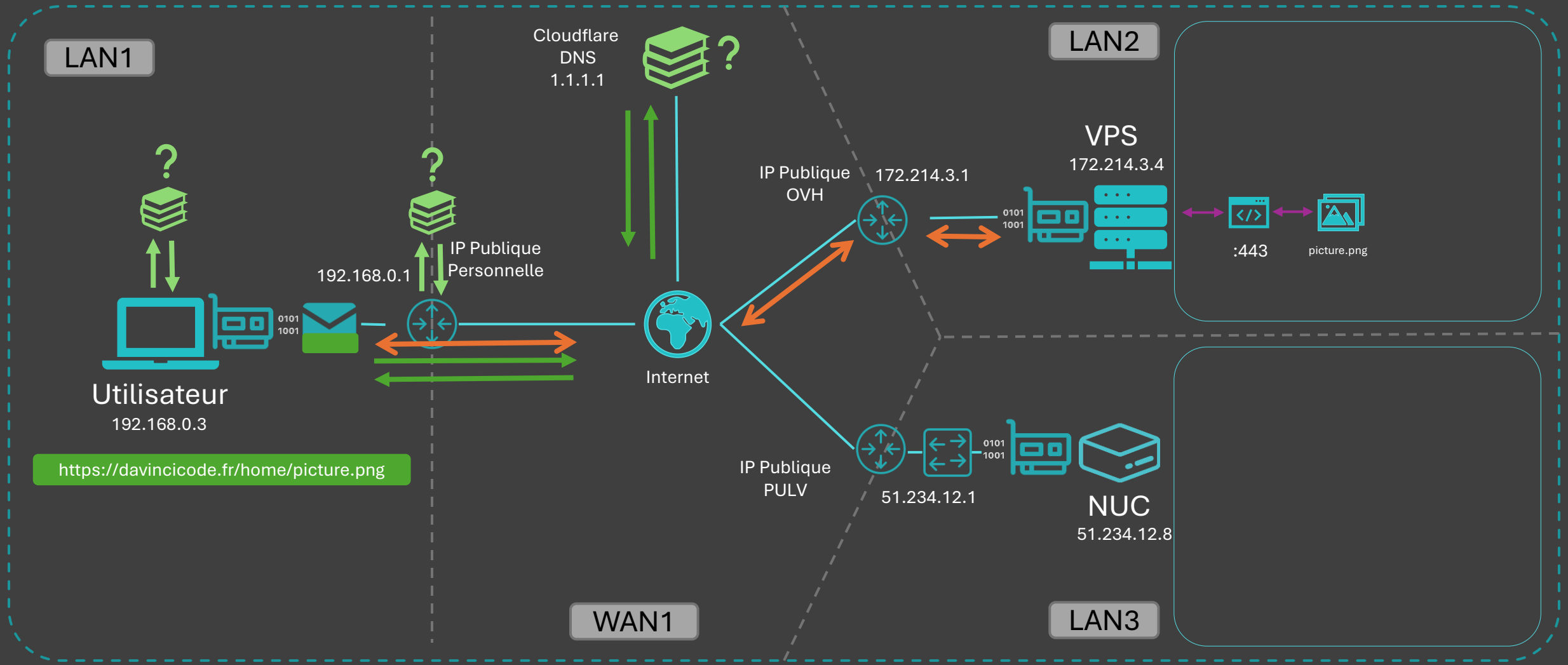








Résolution d'une URL



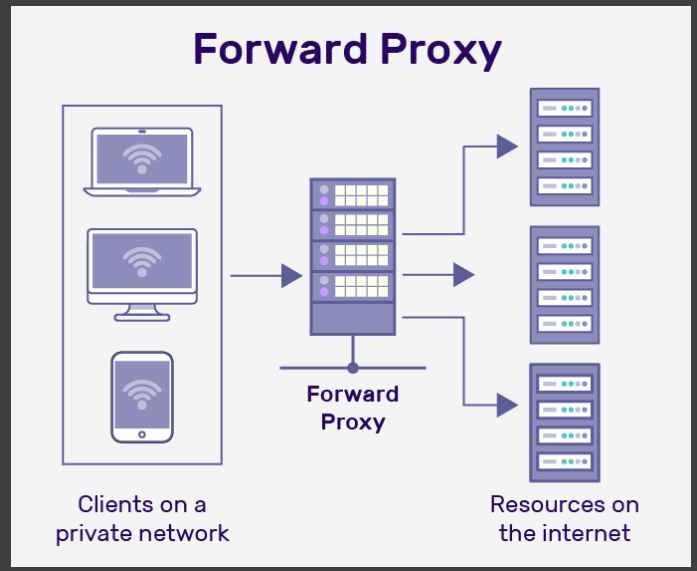
IV. Proxy

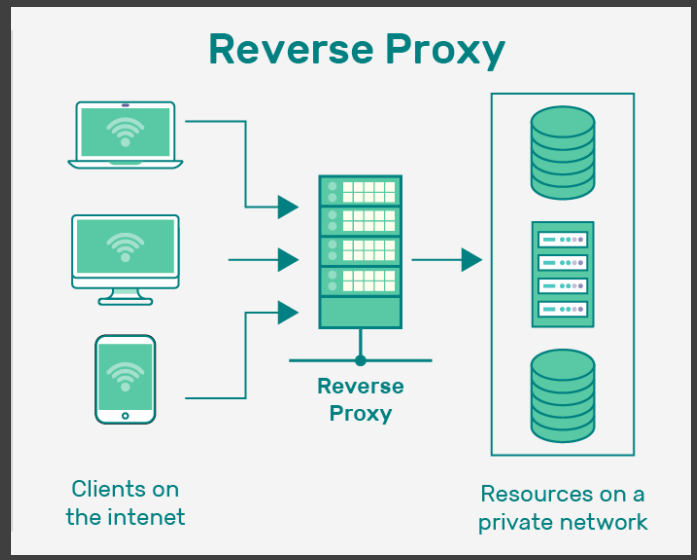
Man In The Middle en moins stylé.

Proxy

- Intermédiaire entre le client et le serveur
- Exemple : Foxy Proxy pour BurpSuite
- 3 types principaux :

Type	Fonctions
Forward	- Contrôle d'accès - Anonymisation - Log - Filtrage / Transformation
Reverse	- Load balancing - Utilisation de cache - Terminer le SSL/TLS
Transparent	- Log - Monitoring





Source : https://learn.microsoft.com/en-us/microsoft-cloud/dev/dev-proxy/concepts/what-is-proxy

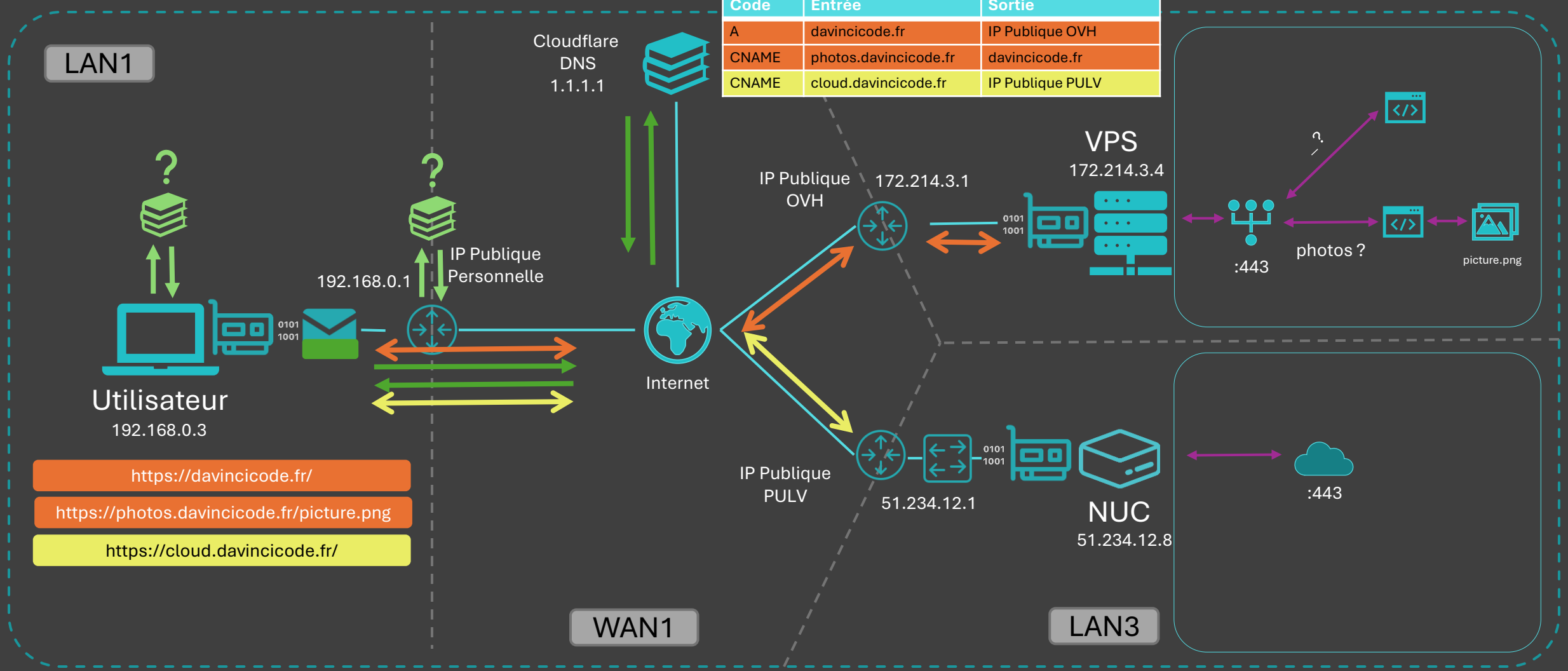
Reverse Proxy

- Un point d’entrée pour plusieurs sous-services
- Exemple : NGINX, HAProxy, Apache

- Utilisation concrète :
  - Un ordinateur avec plusieurs outils ou pages web
  - Tous les clients peuvent accéder à l’ordinateur
  - L’ordinateur redirige les clients vers les bonnes ressources
    - Avec un reverse proxy
    - Selon les paramètres utilisés par les clients

Résolution d'une URL

Code	Entrée	Sortie
A	davincicode.fr	IP Publique OVH
CNAME	photos.davincicode.fr	davincicode.fr
CNAME	cloud.davincicode.fr	IP Publique PULV



NGINX : Configurations

- Configuration globale
  - /etc/nginx/nginx.conf

- Configuration des routes
  - Bonne pratique : un fichier par route

  - Dossier de configuration disponibles : /etc/nginx/sites-available/*
  - Dossier de configurations actives : /ect/nginx/sites-enabled/*

  - Pour activer une route :
    - On créer une référence (symlink) de la configuration dans sites-enabled
    - Commande : ln –s /etc/nginx/sites-available/nom_config /ect/nginx/sites-enabled

NGINX : Config HTTP

```
map $host $target_port {
  ~^(?<port>\d+)\.box\.davincicode\.fr$ $port;
  default "";
}

server {
  listen 80;
  listen 443 ssl;

  # Autoriser les ports 49000 à 65999 "^((49[1-9]\d{2})|(5\d{4})|6[0-5]\d{3})$"
  if ($target_port !~ "^\d{5}$") {
    return 403;
  }

  server_name ~^\d+\.box\.davincicode\.fr$ *.box.davincicode.fr;

  ssl_certificate /etc/nginx/certificates/dvc_certificate.pem;
  ssl_certificate_key /etc/nginx/certificates/dvc_certificate.key;

  ssl_protocols TLSv1.2 TLSv1.3;
  ssl_ciphers HIGH:!aNULL:!MD5;

  location / {
    proxy_pass http://192.168.30.101:$target_port;
    proxy_set_header Host $host;
    proxy_set_header X-Real-IP $remote_addr;
    proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;
    proxy_set_header X-Forwarded-Proto $scheme;
  }
}
```

NGINX permet de :

- Définir des variables : map
- Définir une route http : server

Ensuite on peut :

- Extraire des parties de header
- Ecouter sur plusieurs ports à la fois, ex : 80 et 443
- Utiliser un chiffrement SSL / TLS
- Utiliser des regex
- Définir les protocoles & algorithmes de chiffrement
- Définir la redirection à appliquer
- Ajouter / modifier les headers du payload HTTP avant de le forward

⚠ Ne pas oublier les « ; » à la fin des lignes !

NGINX : Config HTTP

```
stream {

    # Prevent slow-loris style attacks on TCP
    tcp_nodelay on;
    tcp_nopush on;
    timeout 15s;

    # Optional: limit connections per IP
    limit_conn_zone $remote_addr zone=conn_limit:10m;

    upstream backend_tcp {
        server 10.0.0.10:5432 max_fails=3 fail_timeout=10s;
        server 10.0.0.11:5432 max_fails=3 fail_timeout=10s;
    }

    server {
        listen 9000 default_server;
        proxy_pass backend_tcp;
        limit_conn conn_limit 20;

        # Optional: allowlist
        allow 192.168.1.0/24;
        deny all;

        # Log format for auditing
        access_log /var/log/nginx/stream-access.log basic;
        error_log  /var/log/nginx/stream-error.log warn;
    }
}

log_format basic '$remote_addr [$time_local]‘ ……;
```

NGINX permet de :

- Définir une route tcp : stream

Ensuite on peut :

- Définir des sessions et ses paramètres
  - Timeout
  - Conn_limit.
- Rediriger les flux / ports TCP utilisés
- Filtrer les IP
- Définir plusieurs destination & backups
- Définir un format de log (pout http aussi)
- Peut aussi utiliser le chiffrement avec SSL/TLS

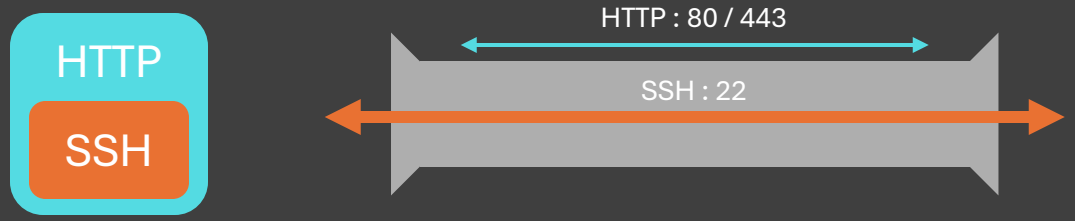
Ne pas oublier les « ; » à la fin des lignes !

V. VPN

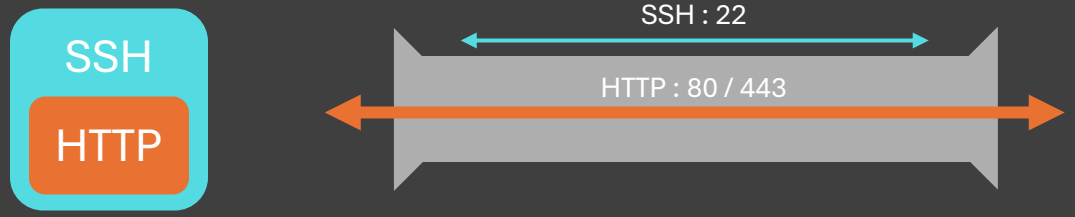
Tunneling = VPN = Tor ?

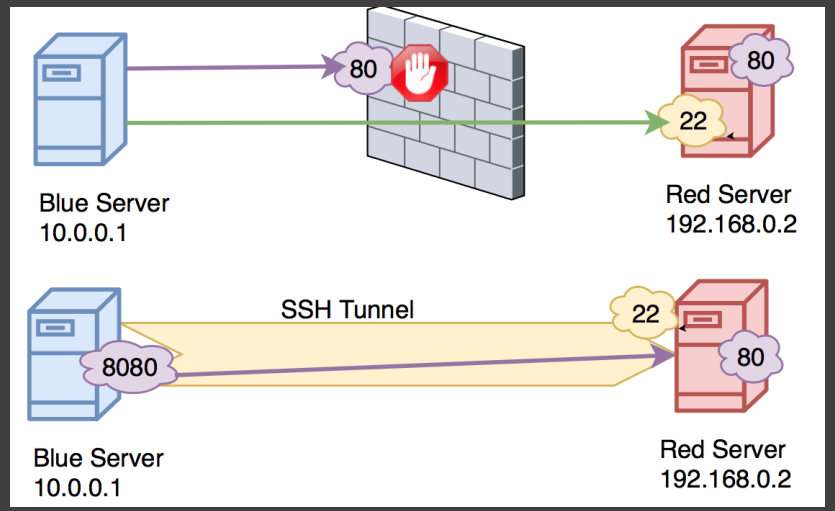
Tunneling

- Echanger des données avec des protocoles non supportés par un réseau
- Comment faire ? En utilisant l'encapsulation !
  - On prend un paquet supporté par un protocole
  - Dans ce paquet, on insert un paquet d'un autre protocole
  - Exemple : HTTP Tunneling



  - Exemple : SSH Tunneling





Source : https://www.cloudflare.com/en-gb/learning/network-layer/what-is-tunneling/

VPN

- Tunneling avec un chiffrement et une sécurité supplémentaire
- Utilisent des protocoles dédiés :

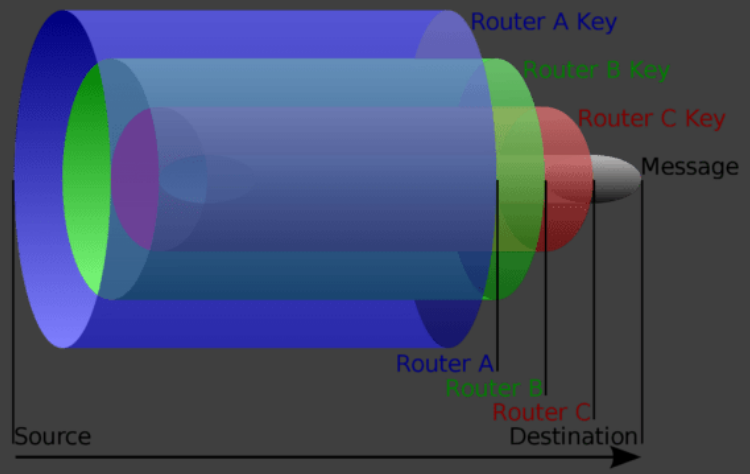
VPN Protocol	Encryption	Handshake	Mobile Support	Speed
OpenVPN	AES-256	TLS/SSL, ECC	Yes	Good
WireGuard	ChaCha20	Curve25519	Yes, native on Android	Excellent
IKEv2/IPsec	AES-256, 3DES	IKEv2	Yes	Very good
L2TP/IPsec	AES-256, 3DES	L2TP, IPsec (separate)	Yes	Good
SSTP	TLS/SSL	TLS/SSL	No	Poor
PPTP	MPPE (RC4)	MS-CHAPv2	No	Poor

- Split tunneling : Utilisation simultané d'un VPN et d'un autre réseau

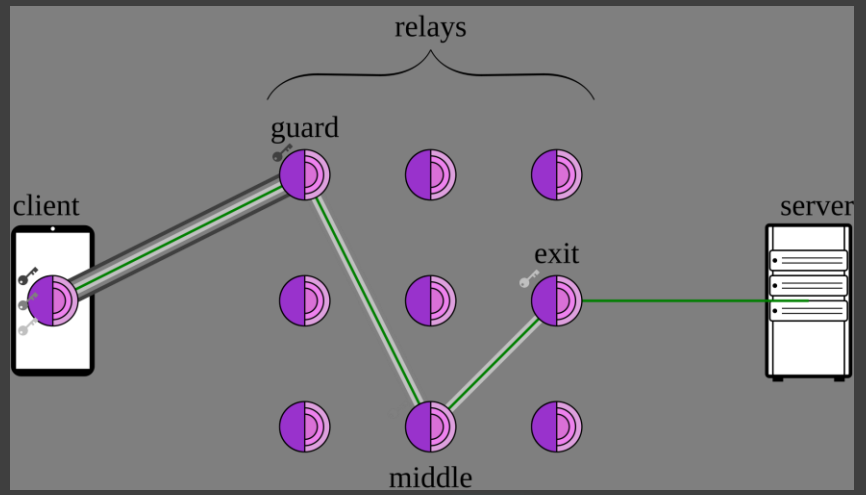
Source : https://www.cloudwards.net/vpn-protocol-breakdown/

Et Tor ?

- Chaine de chiffrement symétrique entre plusieurs nœuds

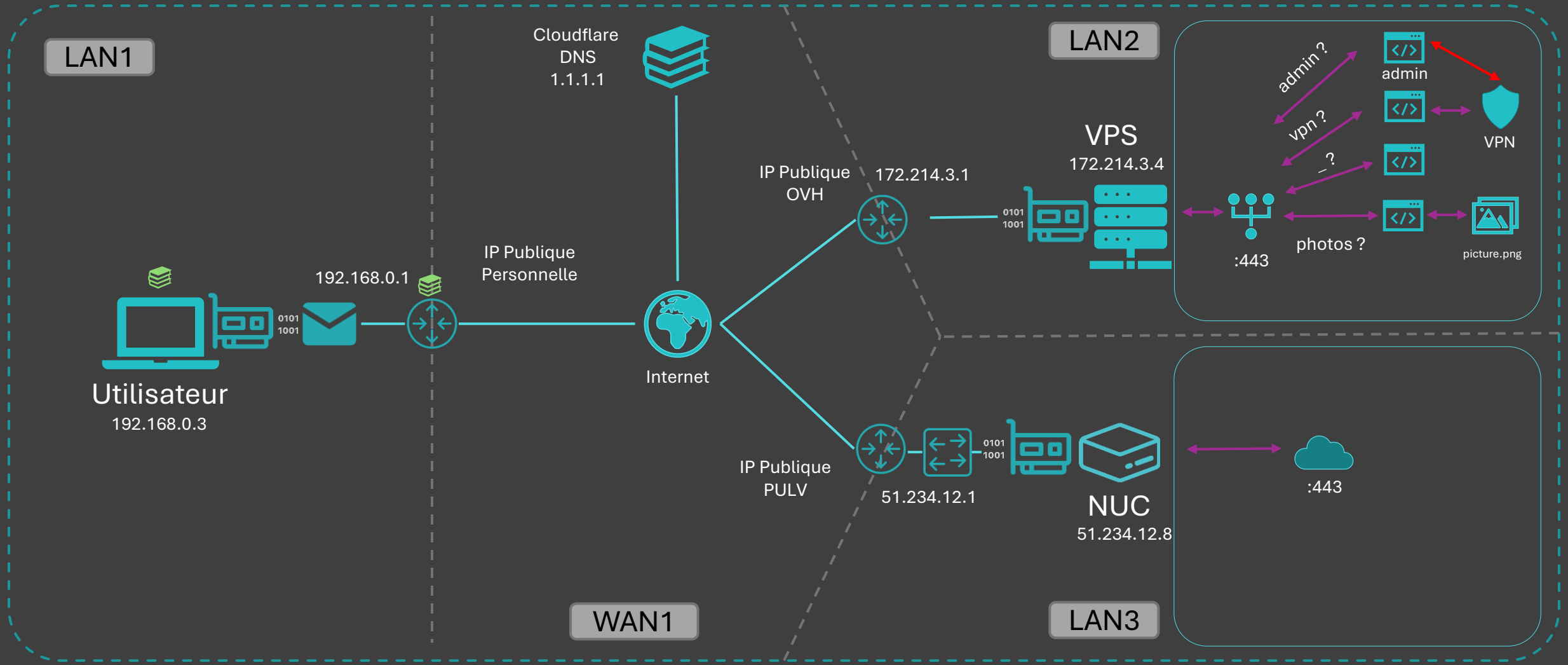


- Les nœuds sont définis au début
- Chaque nœud défini une clé avec le client
- Pour chaque message
  - Le client chiffre le contenu avec toutes les clés
    - De la dernière jusqu'à la première clé
  - Le client envoie le message au nœud suivant
  - Chaque nœud retire une étape du chiffrement



- Le handshake utilise un protocole dédié : NTor
- Ports 9001 et 9030

Ajout d'un VPN



VI. Firewall

Knock knock, who's there ?

Firewall

- Filtrer, bloquer et visualiser les données transitant sur un réseau.
- Layer 4 à 7

- Support :
  - Hardware : Machine dédiée, entre le routeur et le reste du réseau
  - Software : Installé sur un serveur, doit être installé sur chaque machine

- Fonctionnement :
  - Stateful : Filtrage des connexions actives : src, dst, contenu, connexions, comportement
  - Stateless : Filtrage des packets (Transport Layer) en servant des headers : IP, ports, protocoles

- Autres types :
  - Proxy FW : Gateway entre des réseaux interne et externe
  - Web Application FW : Protège les web apps contre les attaques XSS, SQLI, ...
  - Hybrid Mesh FW : Gestion centralisée de FW sur plusieurs supports : machines, serveurs,
  - Next Generation FW : Filtres poussés avec prévention contre intrusion, CTI, géolocalisation, ...

ACL : Access Control List

- Pour les Firewall (Stateless) et les Routeurs / Switchs (L3)
- Layer 4

- Règles de filtrage sur :
  - Les sources & destinations
  - Les ports
  - Les URLs
  - Les adresses IP / MAC
  - L'accès à des VLANs

Inbound

Rule #	Type	Protocol	Port range	Source	Allow/Deny
100	All IPv4 traffic	All	All	0.0.0.0/0	ALLOW
101	All IPv6 traffic	All	All	::/0	ALLOW
*	All traffic	All	All	0.0.0.0/0	DENY
*	All IPv6 traffic	All	All	::/0	DENY

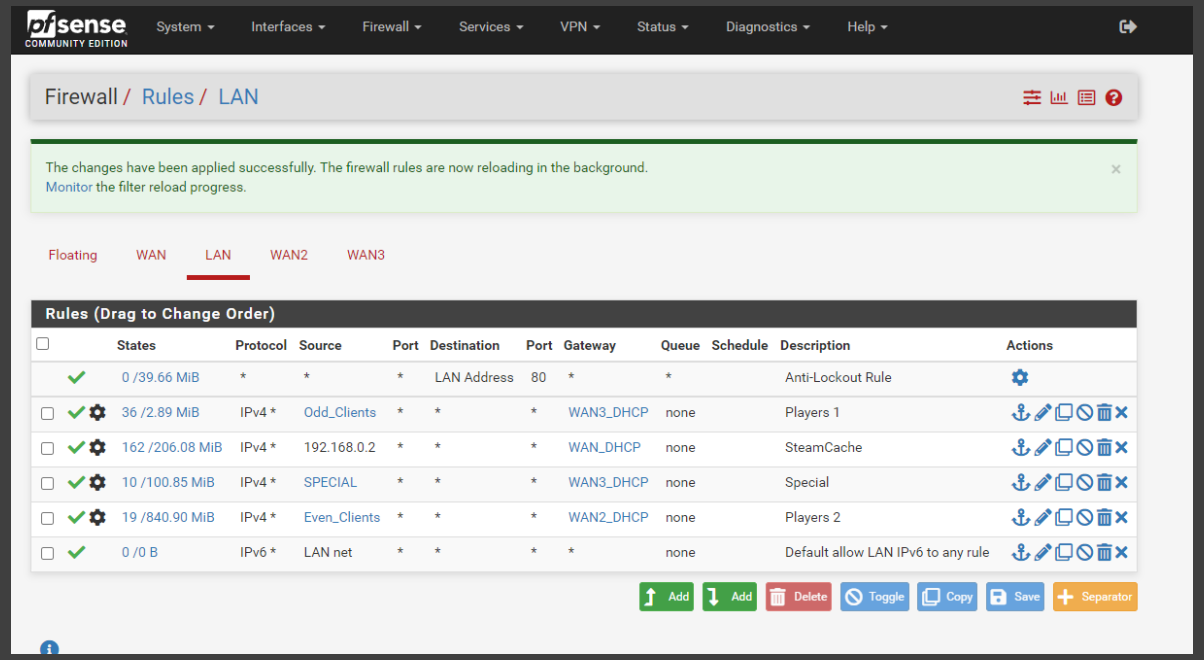
Outbound

Rule #	Type	Protocol	Port range	Destination	Allow/Deny
100	All traffic	All	All	0.0.0.0/0	ALLOW
101	All IPv6 traffic	All	All	::/0	ALLOW
*	All traffic	All	All	0.0.0.0/0	DENY
*	All IPv6 traffic	All	All	::/0	DENY

pfSense

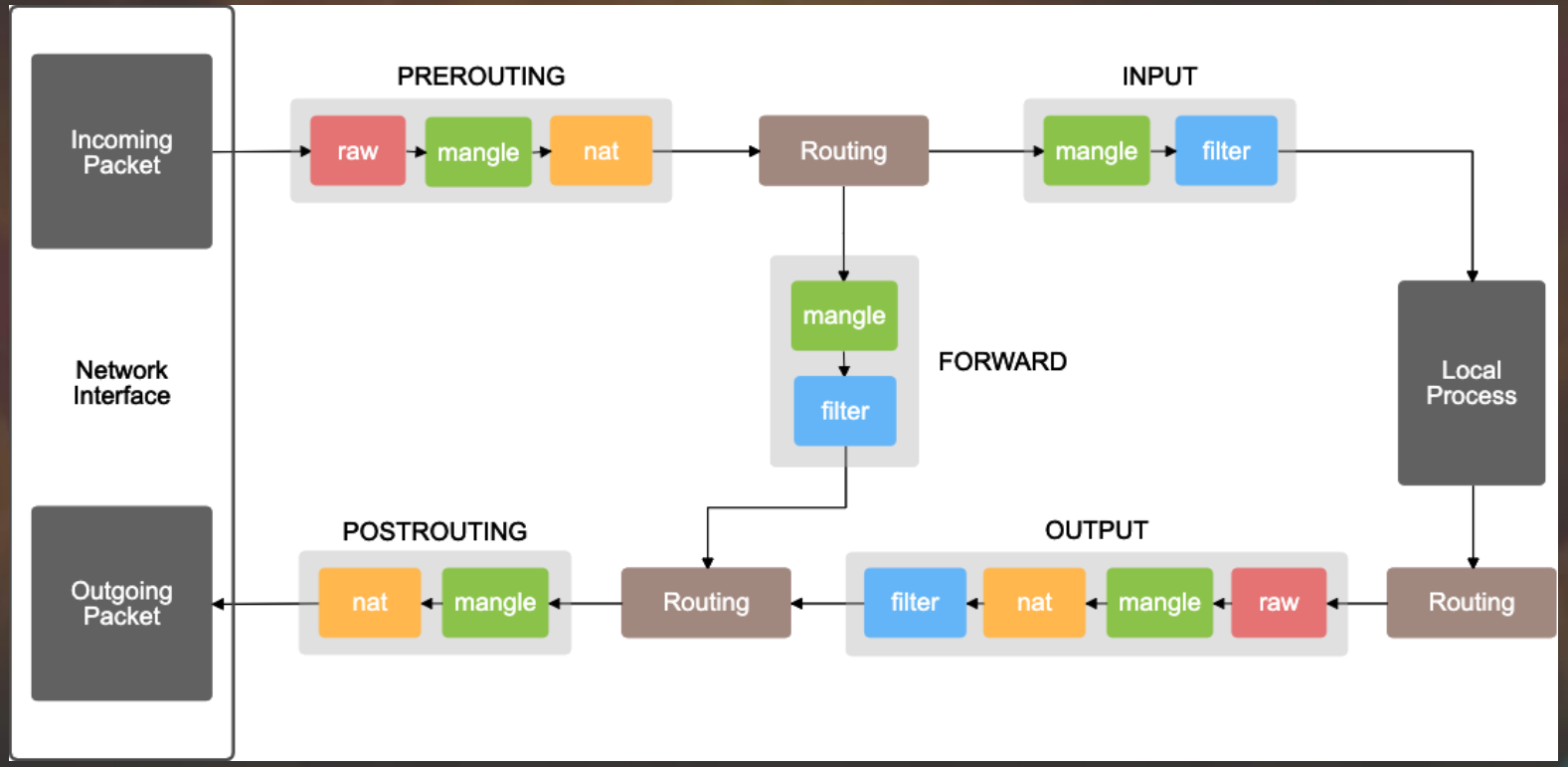
- Firewall logiciel stateful
- Open-source
- Installé sur Linux : FreeBSD

- Permet de :
  - Filtrer les IP & des ports
  - Filtrer les paquets entre les VLANs
  - Rediriger les ports et les IP
  - Et bien plus (dont serveur DHCP 🥶)



Iptables

- Firewall par défaut pour le Kernel Linux
- Configuration sur CLI à effet direct (si vous vous plantez de port, vous êtes dehors)
- Version plus récente : nftables

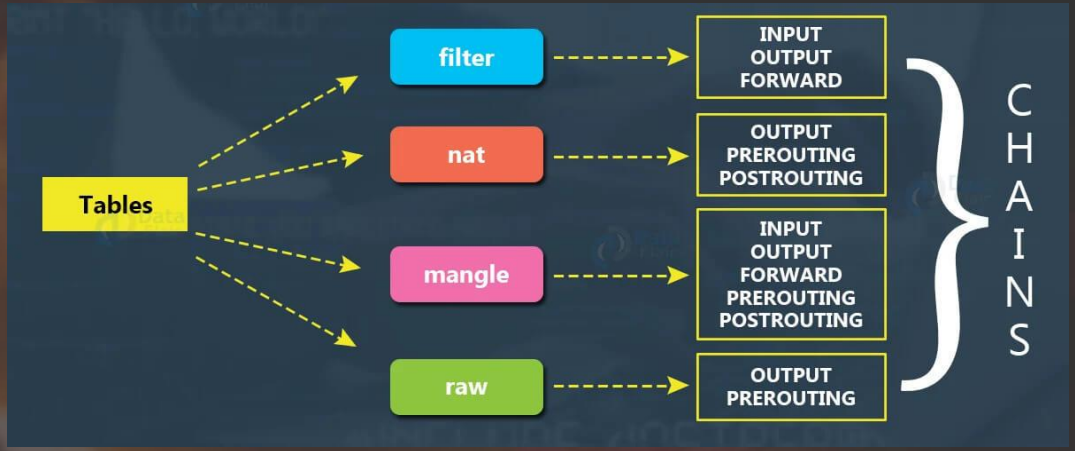


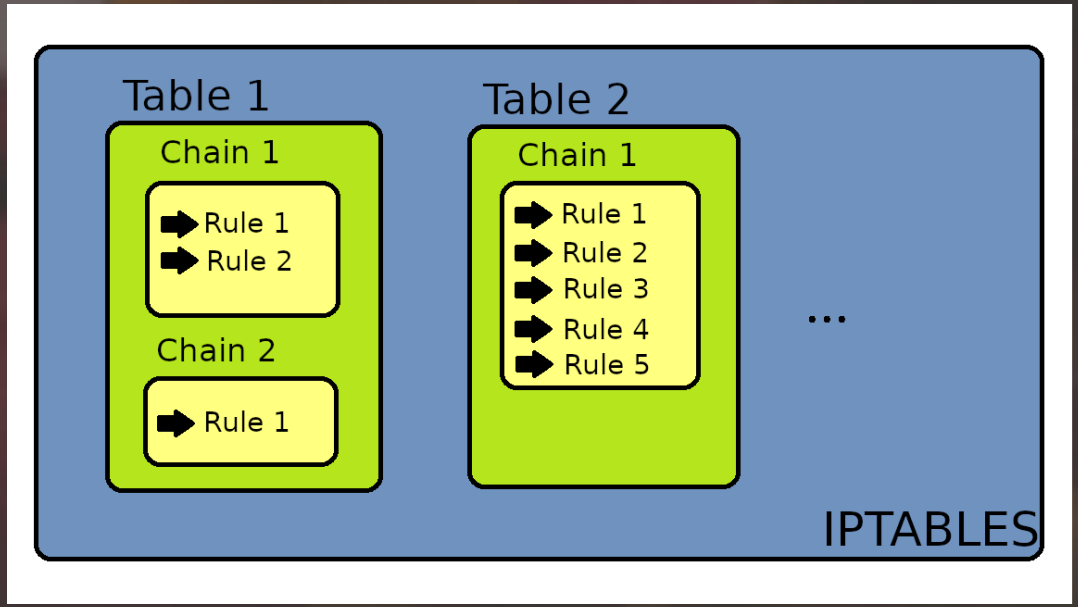
Iptables

- Table : Liste de chaines
  - FILTER : Défaut, entrée / sortie / local
  - NAT : Routage, avant / après / local
  - MANGLE : Altération de packets (TCP)
  - RAW : Exclusions

- Chaine : Liste de règles

- Règle :
  - ACCEPT : Passe
  - DROP : Bloque
  - QUEUE : Décision par une autre application 🧐
  - RETURN : Skip les règles suivantes de la chaine





Iptables

- Filtrage sur :
  - Ports
  - Source : IP & Subnet
  - Destination : IP & Subnet
  - Protocole : ICMP, TCP, UDP, ...
  - Target : Autres variables 🐑

- Ajouter des règles :
  - Ports : iptables -A INPUT -p tcp --dport 22 -j DROP
  - IP : iptables -A INPUT -s 192.168.1.0/24 -j ACCEPT
  - NAT : iptables -t nat -A POSTROUTING -o eth0 -j SNAT --to-source 203.0.113.5

Iptables

- Lister des règles :
  - IPV4 : iptables –S
  - ALL : iptables –L –v –n –line-numbers
  - TABLE : iptables -L INPUT -v -n

```
Chain INPUT (policy ACCEPT)
num  target     prot opt source               destination
1    ACCEPT     all  --  anywhere             anywhere             state RELATED,ESTABLISHED
2    ACCEPT     icmp --  anywhere             anywhere
3    ACCEPT     all  --  anywhere             anywhere
4    ACCEPT     tcp  --  anywhere             anywhere             state NEW tcp dpt:ssh
5    REJECT     all  --  anywhere             anywhere             reject-with icmp-host-prohibited
6    ACCEPT     tcp  --  anywhere             anywhere             tcp dpt:https
7    REJECT     tcp  --  anywhere             anywhere             tcp dpt:http reject-with icmp-port-unreachable
8    ACCEPT     all  --  69.63.176.13         anywhere
9    DROP       all  --  192.168.0.27         anywhere
10   REJECT     all  --  anywhere             anywhere             source IP range 192.168.0.1-192.168.0.255 reject-with

Chain FORWARD (policy ACCEPT)
num  target     prot opt source               destination
1    REJECT     all  --  anywhere             anywhere             reject-with icmp-host-prohibited

Chain OUTPUT (policy ACCEPT)
num  target     prot opt source               destination
```

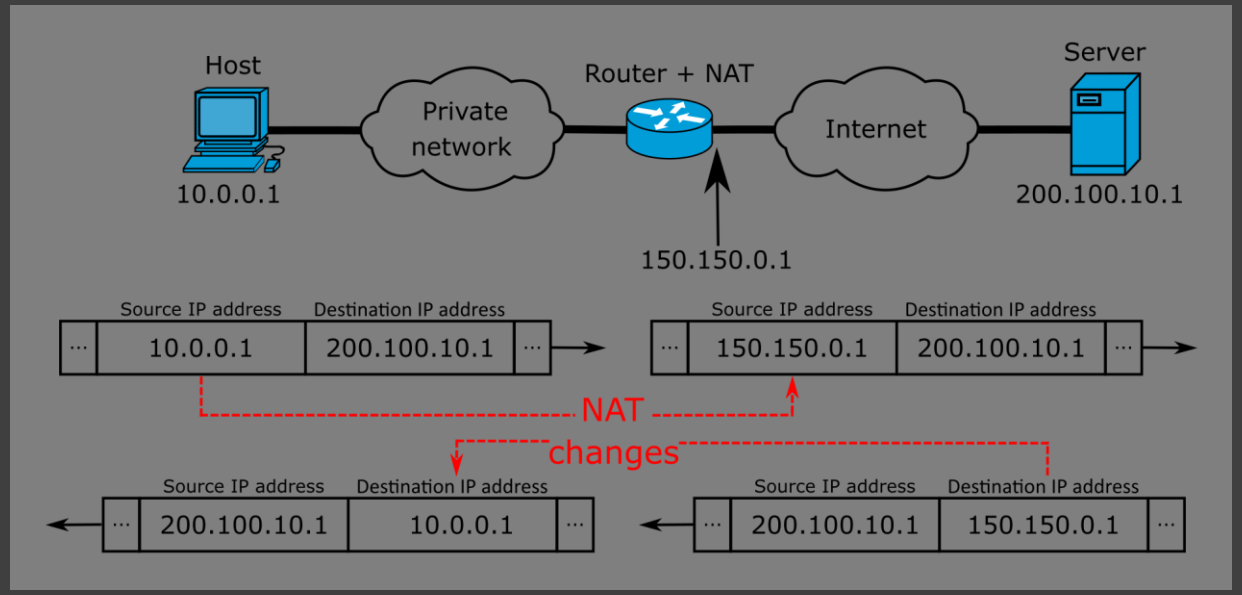
Iptables

- Exemple de modules :
  - Filtrage des authentifications non légitimes ou IP douteuses : fail2ban

- Pas de système de sauvegarde automatique (relancez et vous recommencerez)
  - Iptables-save > /etc/iptables/rules.v4
  - Iptables-restore < /etc/iptables/rules.v4
  - Pour que ce soit persistant, soit :
    - vous configurez le chargement de Linux
    - vous installer iptables-persistent

NAT : Network Address Translation

- Change les IP d'un réseau privé



- Avantages :
  - Une IP publique pour tous les hôtes
  - Chaque hôte à sa propre IP interne

- Types :
  - Statique : 1 IP publique & 1 IP privée par machine
  - Dynamique : n IP publiques & 1 IP privée par machine
  - Port Address Translation : 1 IP publique pour tous,1 port & IP privée par machine

MERCI !