IN CYBER FORUM

EUROPE

26–28 MAR. 2024

LILLE GRAND PALAIS

@FIC_eu

@forumincybereurope

Communications sécurisées par réseaux virtuels

Damien Magoni - *Université de Bordeaux*

Réseau virtuel

Définition (approximative)

- un réseau virtuel est un ensemble de processus logiciels interconnectés par des connexions logiques et s'exécutant sur du matériel standard (COTS)

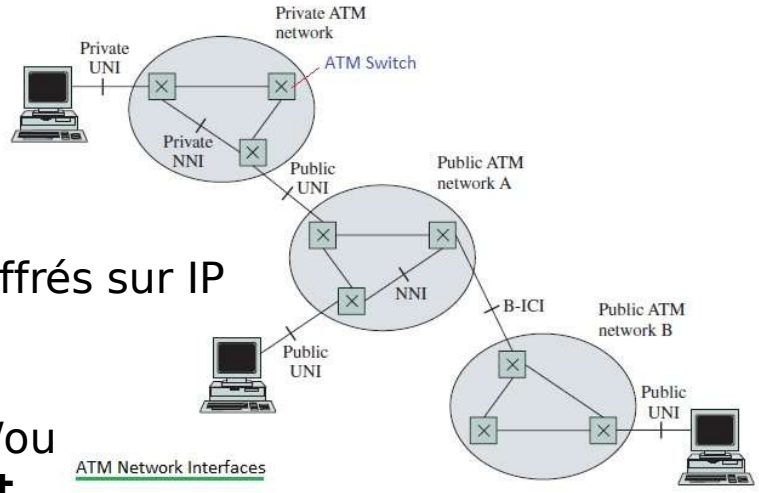
Propriétés

- pas de matériel spécifique à utiliser et à maintenir (commutateur, routeur, etc)
- administration centralisable et/ou automatisable
- configuration optimisée dynamiquement
- un réseau virtuel n'est pas nécessairement sécurisé par cryptographie

Moyen-Age (90s)

ATM (’91-)

- réseau physique partagé par les clients
- réseau logique par client basé sur des **circuits virtuels**
- commutateurs de **cellules**
- sécurité par isolation
- coût élevé



GRE (’94-) / PPTP (’99-’12)

- tunnels point-à-point (PtP) non chiffrés sur IP

IPsec (’95-) → VPN

- connections PtP entre réseaux et/ou équipements au dessus d’**Internet**
- sécurité par **authentification** et **chiffrement**, topologie en étoile
- coût moyen (eg, Cisco ASA)

Moyen-Age (90s)

L2TP (’99-, v3 : ’05-) / IPsec

- tunnel point-à-point pour VPN
- peut transporter des protocoles L2 (PPP, Ethernet) sur IP
- chiffre son trafic de contrôle uniquement
- IPsec utilisé pour sécuriser les données transmises
- coût faible

SSL (‘95-) → Stunnel (’98-)

- proxy fournissant des connexions TLS

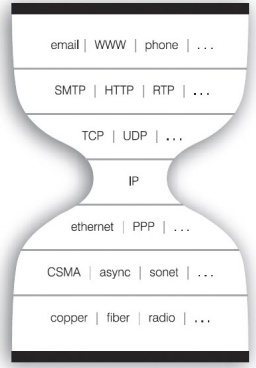
VLAN (’98-)

- réseaux virtuels sur un réseau local Ethernet partagé
- sécurité par isolation
- coût faible

Temps Modernes (’00)

MPLS (’01-)

- réseau **logique** partagé par les clients
- réseau logique par client basé sur des **chemins virtuels**
- s’exécute sur de nombreuses technos L2 et peut transporter des flux IP et L2
- VPN L3 (BGP) VPN L2 (Directed LDP)
- sécurité par isolation
- coût élevé



TLS (’99-) → OpenVPN (’01-)

- VPN logiciel basé sur TLS
- coût faible

Temps Modernes (’00)

Réseaux pair-à-pair (P2P) (’99-)
- réseaux P2P avec diffusion de requête par inondation

Réseaux anonymes (’00-)
- Freenet, The Onion Router (TOR), Invisible Internet Project (I2P)

Table de hachage distribuées (’01-)
- réseaux recouvrants (*overlay*) avec routage

Secure Socket Tunneling Protocol (’07-)
- tunnel TLS qui transporte du trafic PPP

OpenConnect (’09-)
- client VPN avec tunnels DTLS en PtP

Contemporain (’10)

SDN (’08-) / NFV (’12-) / SD-WAN (’14-)

- virtualisation des réseaux (après celle des machines)

Blockchains (’09-)

- réseaux recouvrants de diffusion

Wireguard (’15-)

- tunnels chiffrés sur UDP

ZTNA (’18-) / SASE (’19-)

- accès sécurisés des terminaux (*endpoints*)

Contemporain (’10)

Metro/Carrier Ethernet

- VLANs empilés, *provider bridge* (PB), Q-in-Q (’11-)
- *provider backbone bridge* (PBB) MAC-in-MAC (’11-)
- interconnection opérateurs au niveau Ethernet
- faible coût

❑ IEEE 802.1ad PB

C-DA	C-SA	Type 88A8	S-VID	Type 8100	C-VID	Type	Payload

❑ IEEE 802.1ah PBB or 802.1Qay PBB-TE

B-DA	B-SA	Type 88A8	B-VID	Type 88E7	I-SID	C-DA	C-SA	Type 88A8	S-VID	Type 8100	C-VID	Type	Payload

VXLAN (’14-) / NVGRE (’15-)

- trames L2 dans des paquets UDP (VXLAN) ou IP (NVGRE)
- 24-bit VID
- pour gérer les réseaux Ethernet virtuels dans les DC
- implémenté dans Open vSwitch (’09-)

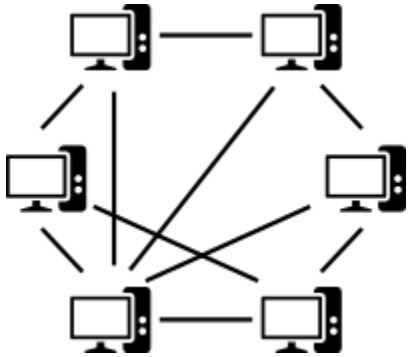
Peer-to-peer

Réseaux pair-à-pair (P2P) (1999-)

- réseaux décentralisés non-structurés de partage de fichiers (Gnutella, eDonkey, etc)

- chaque nœud est client et serveur

- les requêtes sont diffusées par *flooding* à tous les nœuds connectés

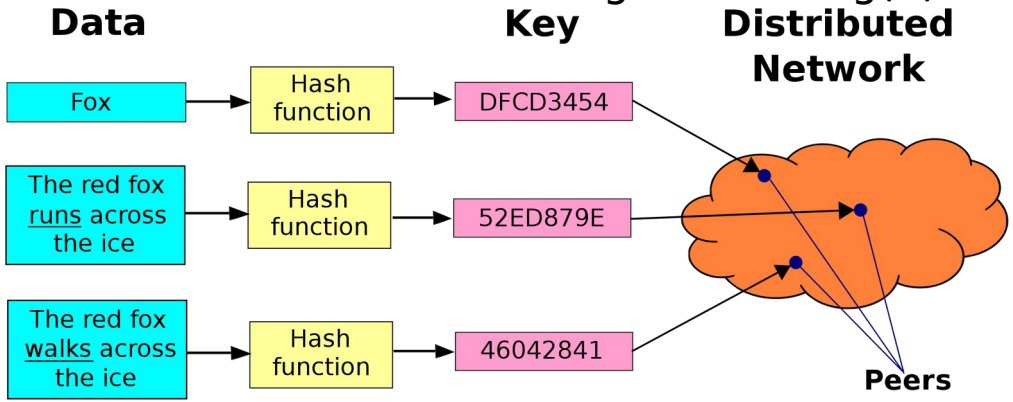


DHT

XLattice

Tables de hachage distribuées (2001-)

- réseaux P2P structurés (CAN, Chord, Kademlia, Pastry, etc) nommés réseaux recouvrants (*overlay*)
- le réseau est construit avec une topologie specifique
- permet de trouver efficacement une ressource
- les nœuds sont aussi des relais et le routage est en log(n)



Blockchains

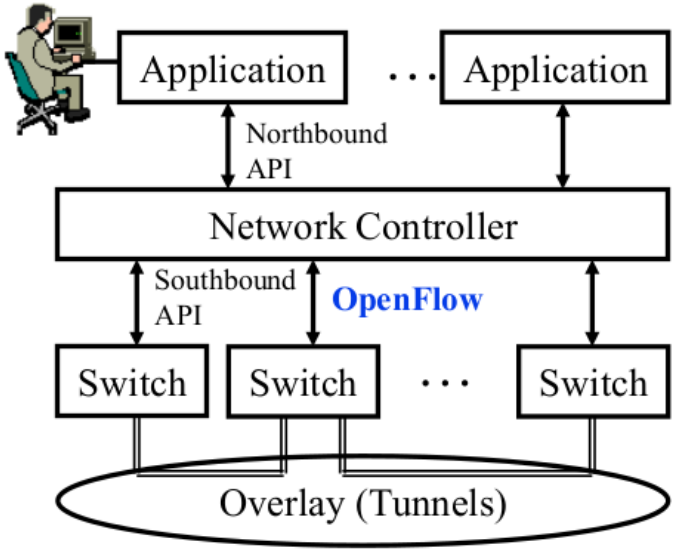
Réseaux P2P de diffusion pour *ledgers* (2009-)

- ledgers décentralisés s'appuyant sur des réseaux P2P pour des cryptomonnaies ou autre
- les communications entre les nœuds sont chiffrées
- elles utilisent un *gossip protocol*
- diffusion optimisée pour éviter les envois redondants
- les transactions doivent être diffusées à tous les nœuds connectés
- réseau bitcoin : 80k *full nodes*
- messages transportés par ZeroMQ

SDN

Software-Defined Networking (2008-)

- séparation du plan de contrôle et du plan de données
- centralisation du contrôle
- automatisation → orchestration
- optimisation des performances (élasticité)



SDN

Implémentation

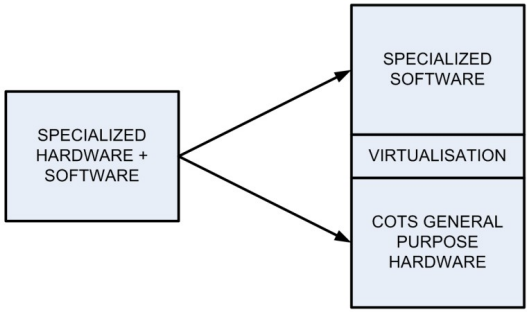
- contrôleur logiciel sur machine standard (x86)
- commutateurs programmables matériels ou logiciels
- protocole entre le contrôleur et les commutateurs (Openflow, etc)
- tables de flux (matching puis action), granularité flux

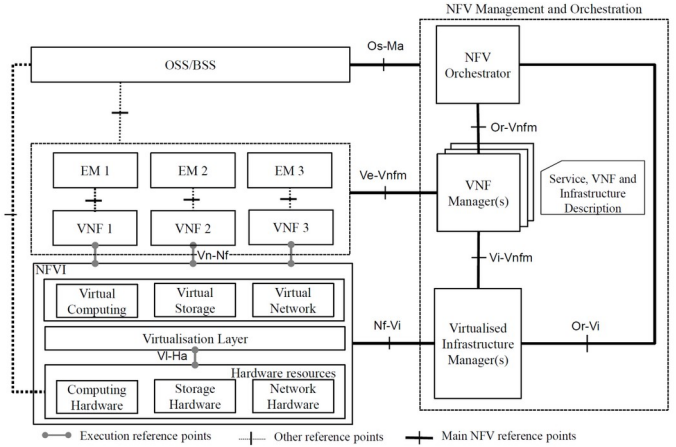
Port	Src MAC	Dst MAC	VLAN ID	Priority	EtherType	Src IP	Dst IP	IP Proto	IP ToS	Src L4 Port ICMP Type	Dst L4 Port ICMP Code	Action	Counter
*	*	0A:C8:*	*	*	*	*	*	*	*	*	*	Port 1	102
*	*	*	*	*	*	*	192.168.*.*	*	*	*	*	Port 2	202
*	*	*	*	*	*	*	*	*	*	21	21	Drop	420
*	*	*	*	*	*	*	*	0x806	*	*	*	Local	444
*	*	*	*	*	*	*	*	0x1*	*	*	*	Controller	1

NFV

Network Function Virtualisation (2012-)

- séparation des fonctions réseaux du matériel sous-jacent
- standardisation des interfaces entre fonctions et infra
- automatisation, (ré)assignation rapide des ressources
- élasticité, optimisation des performances
- capex/opex réduits





@FIC_eu

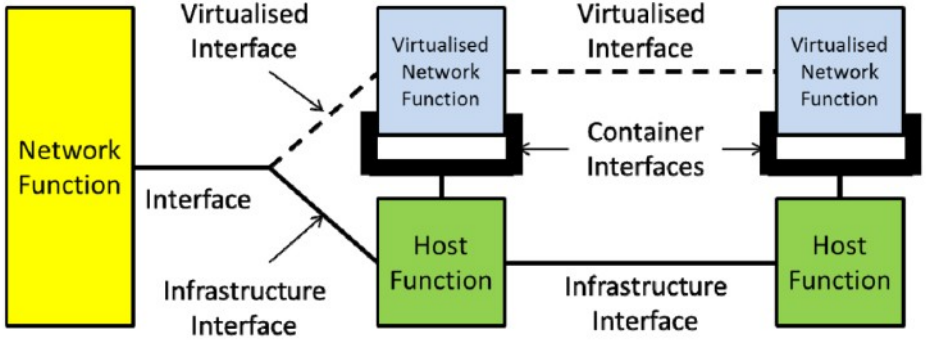
@forumincybereurope

NFV

Implémentation

- fonctions réseaux (FW, IDS, LB, CASB, etc) / réseaux mobiles (MME, S/P-GW, etc) dans des *containers* sur machines standards (x86) → *virtual appliances*

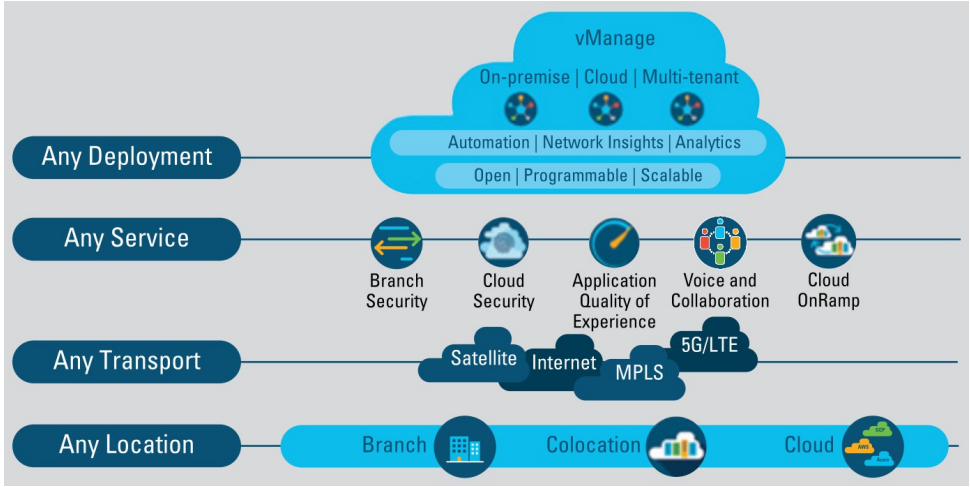
- machines physiques placées dans des PoP situés *on-premises* ou dans le *cloud*



SD-WAN

Software-Defined Wide Area Network (2014-)

- router sur plusieurs technos selon leurs coûts
- supporter la sélection dynamique de chemins
- monter/démonter rapidement des liens
- aggréger des liens
- rerouter rapidement si congestion ou panne
- prioriser selon les types de trafic (QoS)



@FIC_eu

@forumincybereurope

ZTNA

Zero-Trust Network Access (2018-)

- source unique et fiable de l’identité de l’utilisateur
- authentification de l’utilisateur
- authentification de l’équipement
- politique de conformité et santé de l’équipement
- politique d’autorisation d’accès aux applications
- politique de contrôle d’accès dans l’application

(définition du NCSC, UK)

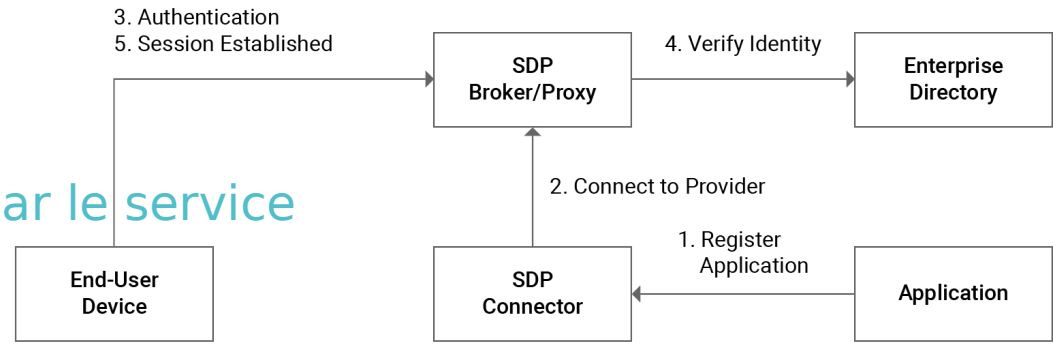
ZTNA Vendor	Service Name	SASE
Akamai	Enterprise Application Access	⏏
Broadcom	Secure Access Cloud	
Cato Networks	Cato Cloud	⏏
CheckPoint	Quantum, Cloudguard	⏏
Cisco	Duo	⏏
Crowdstrike	Workspace Essentials	
Cloudflare	Cloudflare Access	
Fortinet	FortiSASE	⏏
Google	BeyondCorp Remote Access	
McAfee	Ultimate	
Netskope	Netskope Private Access	⏏
Okta	Okta Identity Cloud	
OPAQ	Secure Access Service Edge	⏏
Palo Alto Networks	Prisma Access	⏏
Perimeter 81	Software-Defined Perimeter	⏏
SAIFE	Continuum	
Zscaler	Private Access	⏏

ZTNA

ZTNA initié par le service

Conceptual Model of Service-Initiated ZTNA

SDP : software-defined perimeter



Source: Gartner (April 2019)
ID: 386774

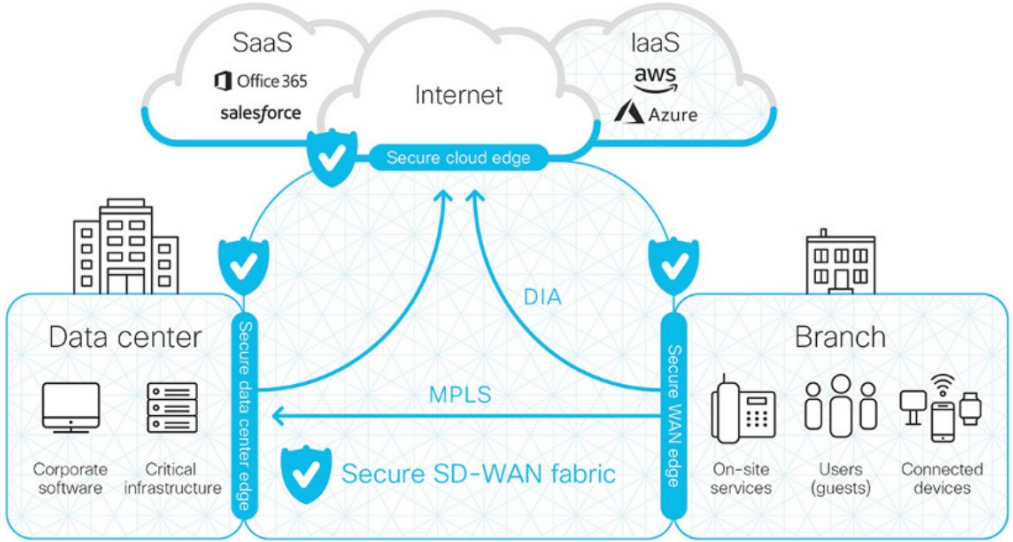
- un connecteur est installé sur le même réseau que l'application serveur et établi une cnx sortante vers un *proxy* dans le cloud
- les utilisateurs s'authentifient au SI via le *broker* pour accéder aux applis
- les *firewalls* de l'entreprise n'ouvrent pas de ports entrants
- pas d'agent sur le *endpoint*
- les protocoles doivent passer sur HTTP(S)

SASE

"To protect anywhere, anytime access to digital capabilities, security must become software-defined and cloud-delivered"
Gartner

Secure Acces Service Edge (2019-)

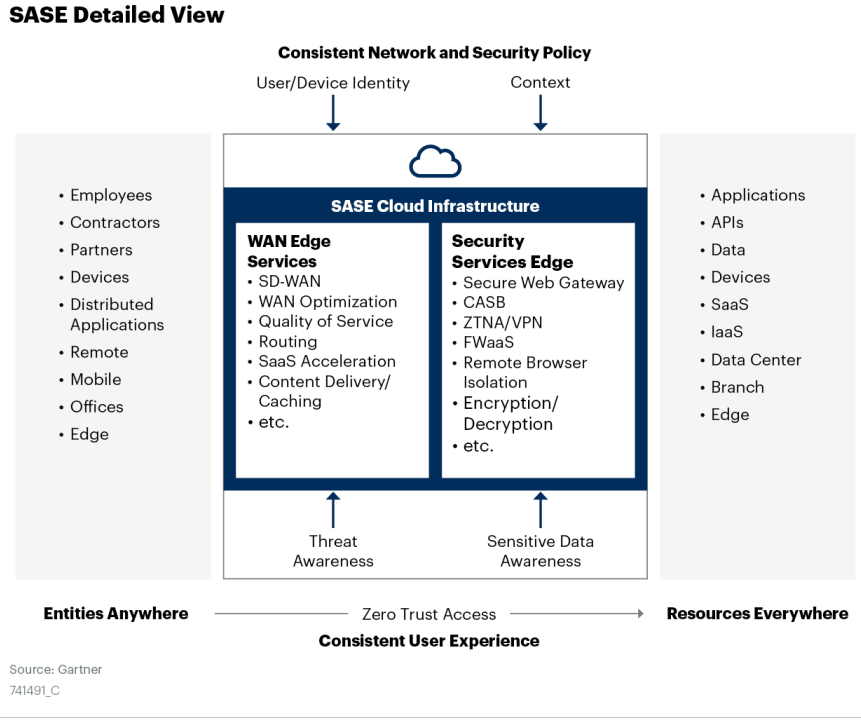
- convergence réseau (SD-WAN) et sécurité (Secure Service Edge)
- SSE = ZTNA + SWG + CASB + FWaaS + DNS + ...



SASE

Propriétés

- accès par politique (rôle, etc) aux applications *cloud* et entreprise
- sécurise les accès distants aux applis *cloud* sans passer par le LAN de l'entreprise
- points de présence multiples et dispersés pour répondre aux contraintes SLAs (disponibilité, latence)
- entité et équipement surveillé et bloqué dynamiquement en cas de comportement suspect



@FIC_eu

@forumincybereurope

Wireguard

Protocole de tunnel chiffré (2015-)

- une seule suite cryptographique (Curve25519 pour le DHE, ChaCha20 pour le chiffrement, etc)
- pas de certificats ni de PKI
- associe les clés publiques avec des adresses IP autorisées
- créé une interface virtuelle sur l'hôte
- paquets encapsulés dans UDP
- implémentation concise (4kloc) prouvée fiable par méthodes formelles
- multiplateforme (Linux, Windows, Android, etc)

Tailscale

tailscale

« People often ask us for an overview of how Tailscale works. We've been putting off answering that, because we kept changing it! »

Software-defined mesh VPN (2019-)

- maillage direct entre les nœuds → chiffrement E2E
- basé sur des tunnels wireguard (UDP)
- n'utilise pas de certificats ni PKI
- utilise un *coordination server* pour collecter et distribuer les clés publiques ainsi que les infos de connexion
- le CS communique avec un serveur d'auth. externe afin de donner les bonnes clés aux bons utilisateurs (ACL)
- utilise STUN et ICE pour gérer les pbs de NAT
- propose un réseau de relais TCP lorsque UDP n'est pas utilisable
- voir aussi Enclave.io (2020-)

0-tier

Smart programmable Ethernet switch for planet Earth (2015-)

- tous les équipements communiquent comme si ils étaient sur le même LAN
- hyperviseur réseau distribué construit sur un réseau P2P global (couche VL1) fait de liens dynamiques chiffrés entre pairs authentifiés
- l’entrée dans VL1 se fait via des *root servers*
- les trames Ethernet sont transportées par la couche VL2 qui fonctionne de manière similaire à VXLAN
- un contrôleur réseau se trouve dans chaque réseau virtuel
- il sert de CA et signe les certificats des équipements

Mysterium

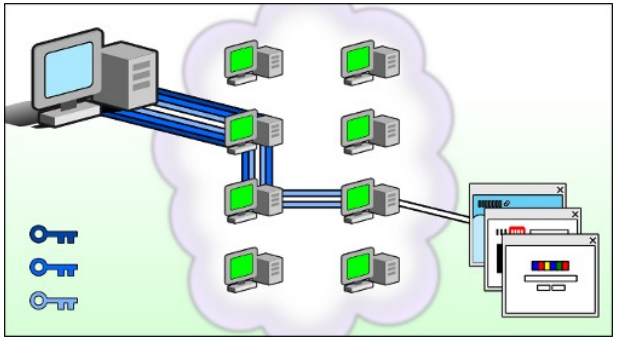
People powered privacy (2024-)

- anonymat de l’adresse IP du client en utilisant l’adresse IP d’un utilisateur résidentiel participant au VPN distribué (dVPN)
- pas de serveurs
- réseau P2P s’appuyant sur les machines des autres utilisateurs
- 20k+ nœuds dans 135 pays
- utilise wireguard

TOR

The Onion Router (2002-)

- anonymat par masquage de l'@ IP du client
- réseau recouvrant (*overlay*) de ~8k ordinateurs (Tor *relays*) appartenant à des volontaires
- utilise trois tunnels chiffrés imbriqués vers 3 relais pris au hasard
- ne transporte que TCP (utilise SOCKS)



Usage

- accessible par des applis custom (Tor Browser, Tor Messenger, Tor chat, SecureDrop, OnionShare, etc)
- boites dédiées pour router tout le trafic vers Tor (InvizBox, anonabox)
- Tails : OS qui route tout le trafic sur Tor

TOR

Fiabilité des relais

- chaque relai possède une clé publique *identity key*

- chaque répertoire d’autorité (Directory Authority) possède une *directory signing key*

- ces DA fournissent une liste signée de tous les relais connus, précisant leurs clés, localisations, politique de sortie, status, etc.

- il faut contrôler la majorité des DA (9 répertoires en 2024) pour leurrer les clients vers des relais corrompus

- logiciel Tor livré avec une liste des addresses et des clés publiques des DA

TOR

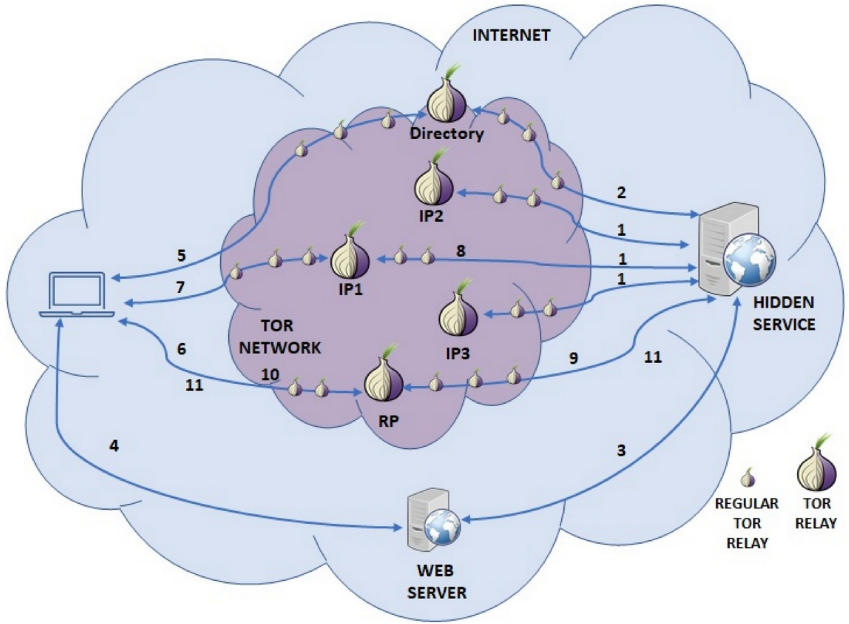
Hidden Services (HS)

- anonymat par masquage de l'@ IP et du port du serveur
- le serveur construit des circuits vers plusieurs *introduction points* (relais Tor)
- la liste de ces *ip* est stockée dans un fichier *descriptor* placé sur des nœuds HSDir (relais Tor)
- le client récupère le fichier *descriptor*

TOR

Hidden Services (HS)

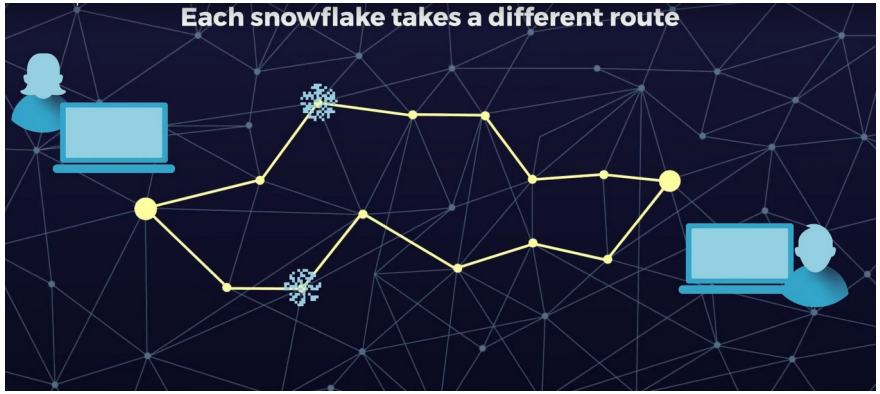
- le client construit un circuit vers un *rendezvous point*
- il créé un circuit vers un des *ip* et lui envoie le *rp*
- le serveur construit un cicruit vers le *rp*
- les deux circuits sont reliés
- les clients peuvent être authentifiés par des *preshared secrets* échangés OOB entre eux et le HS
- utilisé par Chimere (2019-)



Snowpack

Virtual & Invisible Private Network (2021-)

- anonymat par masquage des IP client et serveur (mode tunnel)
- réseau recouvrant (*Snowpack Network Overlay*) partagé de nœuds opérés par Snowpack, des partenaires et certains clients
- données fragmentées en flocons échangés via des routes séparées
- aucun nœud du SNO n'est en mesure de connaître l'origine, la destination et le contenu d'un flocon



Merci de votre attention !

Des questions ?