Aide-mémoire RXI - Réseaux et Internet

CONCEPTS FONDAMENTAUX

Topologies de Réseau

Bus : Tous les nœuds connectés à un câble commun
Étoile : Tous les nœuds connectés à un concentrateur central
Anneau : Nœuds connectés en boucle fermée
Maillée : Interconnexion complète ou partielle entre nœuds
Arbre : Hiérarchie de nœuds avec un nœud racine

Types de Réseaux

LAN (Local Area Network) : Réseau local (bâtiment)
MAN (Metropolitan Area Network) : Réseau métropolitain
WAN (Wide Area Network) : Réseau étendu (pays/monde)
PAN (Personal Area Network) : Réseau personnel (Bluetooth, etc.)
VLAN : Réseau local virtuel

Paramètres de Performance

Débit : Quantité de données transmises par seconde (bit/s)
Latence : Temps de transmission d'un point à un autre (ms)
Gigue : Variation de la latence dans le temps
Perte de paquets : Pourcentage de paquets perdus
Disponibilité : Pourcentage de temps opérationnel (99.9% = 8.76h d'indisponibilité/an)

Types de Délais

Transmission : Temps pour envoyer un paquet (Taille/Débit)
Propagation : Temps de traversée du support (Distance/Vitesse)
File d'attente : Temps d'attente dans les routeurs
Traitement : Temps de traitement par les équipements
Total = Transmission + Propagation + File d'attente + Traitement

Exemples de Calculs

Délai de transmission : $T_{trans} = \frac{L}{R}$
L : Taille du paquet (bits)
R : Débit du lien (bits/s)
Délai de propagation : $T_{prop} = \frac{d}{v}$
d : Distance (m)
v : Vitesse de propagation (m/s)
Bande passante-délai produit : $BDP = R \times RTT$
Efficacité : $\eta = \frac{T_{trans}}{T_{trans}+2\times T_{prop}}$

INTRODUCTION AUX RÉSEAUX

Concepts Fondamentaux

Réseau : Ensemble d'équipements interconnectés échangeant des informations
Protocole : Ensemble de règles définissant comment les données sont transmises
Topologies :
Bus : Tous les nœuds partagent le même support
Étoile : Tous les nœuds connectés à un point central
Anneau : Les nœuds forment une boucle
Maillage : Interconnexions multiples entre les nœuds
Modèle OSI (7 couches) :
Physique : Transmission brute des bits
Liaison : Accès au média, détection d'erreurs (Ethernet, PPP)
Réseau : Routage des paquets (IP, ICMP)
Transport : Communication de bout en bout (TCP, UDP)
Session : Gestion des sessions
Présentation : Format des données
Application : Interface utilisateur (HTTP, FTP, SMTP)
Modèle TCP/IP (4 couches) :
Accès réseau (Liaison + Physique)
Internet (Réseau)
Transport
Application (Session + Présentation + Application)

Types de Réseaux

LAN (Local Area Network) : Réseau local (bâtiment)
MAN (Metropolitan Area Network) : Réseau métropolitain
WAN (Wide Area Network) : Réseau étendu (pays, monde)
PAN (Personal Area Network) : Réseau personnel (Bluetooth, USB)
VLAN (Virtual LAN) : Segmentation logique d'un réseau physique

Concepts Clés

Bande passante : Débit maximum théorique (en bps)
Débit : Quantité de données transmises par unité de temps
Latence : Temps de transmission d'un point à un autre
Jitter : Variation de la latence dans le temps
Tolérance aux pannes : Capacité à continuer à fonctionner en cas de défaillance
Qualité de Service (QoS) : Gestion des priorités de trafic

COUCHE PHYSIQUE

1. Codage des Données

NRZ (Non-Return to Zero) :
1 = niveau haut, 0 = niveau bas
Problème : pas d'horloge intégrée
NRZI : Changement de niveau = 1, pas de changement = 0
Manchester :
1 = front montant, 0 = front descendant
Auto-horlogé mais nécessite le double de bande passante
4B/5B :
4 bits encodés sur 5 bits
Permet d'éviter les longues séquences de 0

2. Supports de Transmission

Paire torsadée :
UTP (non blindé) et STP (blindé)
Cat 5e : 1 Gbps, 100 MHz, 100m
Cat 6 : 10 Gbps, 250 MHz, 55m
Fibre optique :
Multimode : courtes distances, LED
Monomode : longues distances, laser
Avantages : immunité aux interférences, débit élevé
Sans fil :
2.4 GHz (WiFi b/g/n), 5 GHz (WiFi a/n/ac)
Problèmes : interférences, atténuation

3. Atténuation et Bruit

Atténuation : Perte de puissance avec la distance
Bruit thermique : Dû à l'agitation thermique
Diaphonie : Interférence entre paires torsadées
RSL (Received Signal Level) : Puissance du signal reçu
Facteur d'affaiblissement : dB = 10 log10(P2/P1)

4. Délais de Propagation

Vitesse de propagation :
Cuivre : 2/3 de c (vitesse de la lumière)
Fibre : 2/3 de c
Délai de propagation : Distance / Vitesse
Délai de transmission : Taille / Débit
BDP (Bande Passante-Délai) : Débit × RTT

STRUCTURE D'INTERNET

Hiérarchie des Réseaux

AS (Autonomous System) : Réseau sous contrôle administratif unique
Identifié par un numéro (ASN) sur 16 ou 32 bits
Exemples : Opérateurs télécoms, grands fournisseurs de contenu
Tiers d'AS :
Tier 1 : Réseaux majeurs interconnectés (peering gratuit)
Tier 2 : Réseaux régionaux, achètent du transit aux Tier 1
Tier 3 : Réseaux locaux, achètent du transit aux Tier 2

Interconnexion des Réseaux

Peering : Échange de trafic direct entre AS
Settlement-free : Aucun paiement (mutuellement bénéfique)
Paid peering : Un AS paie l'autre pour le trafic
Transit : Un AS paie un autre AS pour accéder au reste d'Internet
IXP (Internet eXchange Point) : Point d'échange de trafic neutre
Réduit la latence entre réseaux
Diminue les coûts de transit

Protocoles d'Interconnexion

BGP (Border Gateway Protocol) :
Protocole de routage inter-domaine (EGP)
Échange des préfixes IP entre AS
Basé sur des politiques (pas uniquement sur la métrique)
Communautés BGP :
Étiquettes pour appliquer des politiques de routage
Exemple : "ne pas annoncer à tel AS"

Architecture Typique d'un FAI

Backbone : Réseau cœur à haut débit
Points de Présence (PoP) : Points d'accès répartis géographiquement
Points d'Échange (IXP) : Interconnexion avec d'autres réseaux
Réseau d'Accès : Connexion des abonnés (ADSL, fibre, 4G/5G)

Sécurité et Gestion

ROA (Route Origin Authorization) : Vérifie qu'un AS peut annoncer un préfixe
RPKI (Resource Public Key Infrastructure) : Infrastructure pour sécuriser les annonces BGP
Looking Glass : Outil pour visualiser les routes depuis différents points du réseau

COUCHE PHYSIQUE & LIAISON

Couche Physique

Fonctions :
Transmission brute des bits sur le support physique
Définition des caractéristiques électriques/optiques
Spécification des connecteurs et câbles
Support de transmission :
Câble coaxial (10BASE5, 10BASE2)
Paire torsadée (UTP/STP) - Cat5e, Cat6, Cat7
Fibre optique (mono/multimode)
Sans fil (Wi-Fi, Bluetooth, 4G/5G)
Techniques de transmission :
Bande de base vs bande large
Transmission série/parallèle
Synchronisation (asynchrone/synchrone)
Codage (NRZ, Manchester, 4B/5B, 8B/10B)

Couche Liaison de Données

Sous-couches :
LLC (Logical Link Control) - 802.2
MAC (Media Access Control) - 802.3 (Ethernet), 802.11 (Wi-Fi)
Fonctions :
Découpage en trames (framing)
Contrôle d'erreur (CRC, checksum)
Contrôle de flux
Gestion des accès multiples (CSMA/CD, CSMA/CA, jeton)
Adressage MAC :
48 bits (6 octets) en notation hexadécimale (ex: 00:1A:2B:3C:4D:5E)
3 premiers octets : OUI (Organizational Unique Identifier)
3 derniers octets : Numéro de série
Adresses spéciales : broadcast (FF:FF:FF:FF:FF:FF), multicast
Protocoles :
Ethernet (IEEE 802.3)
Wi-Fi (IEEE 802.11)
PPP (Point-to-Point Protocol)
VLAN (IEEE 802.1Q)
STP (Spanning Tree Protocol - 802.1D)

Ethernet (IEEE 802.3)

Caractéristiques :
Débits : 10 Mbps, 100 Mbps, 1 Gbps, 10 Gbps, 40 Gbps, 100 Gbps
Topologie logique en bus, physique en étoile
Accès multiple avec détection de collision (CSMA/CD)
Types de trames :
Ethernet II (le plus courant)
IEEE 802.3 (avec 802.2 LLC)
IEEE 802.3 avec SNAP
Champs d'une trame Ethernet :
Préambule (7 octets) + SFD (1 octet)
Adresse MAC destination (6 octets)
Adresse MAC source (6 octets)
Type/Longueur (2 octets)
Données (46-1500 octets)
FCS (Frame Check Sequence) - 4 octets (CRC-32)

COMMUTATION ET SWITCHES

Types de Commutation

Commutation de circuits :
Établit un circuit dédié avant la communication
Garantit une bande passante fixe
Peu efficace pour les données en rafales
Exemple : Réseau téléphonique commuté (RTC)
Commutation de paquets :
Découpe les données en paquets indépendants
Meilleure utilisation de la bande passante
Plus tolérant aux pannes
Deux approches :
Datagramme (sans connexion, ex: IP)
Circuits virtuels (avec connexion, ex: MPLS)

Fonctionnement des Switches

Table d'adresses MAC :
Stocke les associations adresse MAC/port
Apprentissage automatique (flooding, filtrage, apprentissage)
Durée de vie des entrées (généralement 5 minutes)
Processus de commutation :
Réception d'une trame
Vérification de l'adresse MAC de destination
Si connue, envoi uniquement sur le port concerné
Si inconnue, diffusion sur tous les ports (sauf source)
Mise à jour de la table d'adresses MAC
Avantages :
Segmentation des domaines de collision
Augmentation de la bande passante disponible
Sécurité améliorée (isolation des segments)

Protocoles et Mécanismes

STP (Spanning Tree Protocol) :
Évite les boucles dans les réseaux commutés
Désactive les chemins redondants
Temps de convergence : 30-50 secondes
VLAN (Virtual LAN) :
Segmentation logique d'un réseau physique
Réduction du domaine de diffusion
Séparation des flux de trafic
802.1Q :
Standard pour le marquage des trames VLAN
Ajoute 4 octets à l'en-tête Ethernet
Permet jusqu'à 4094 VLANs

Sécurité des Switches

Port Security :
Limite les adresses MAC par port
Actions en cas de violation (protect/restrict/shutdown)
Protection contre les attaques :
BPDU Guard : Protège contre les attaques STP
DHCP Snooping : Préserve l'intégrité du DHCP
DAI (Dynamic ARP Inspection) : Protège contre l'empoisonnement ARP

ROUTAGE IP

Acheminement vs Routage

Acheminement (Forwarding):
Niveau 3 (IP), table de routage
Décision par paquet
Utilise adresse IP destination
Routage (Routing):
Protocoles (RIP/OSPF/BGP)
Découverte et partage des routes
Construction des tables de routage

Processus de remise

Directe: Même sous-réseau
ARP pour IP->MAC
Trame avec MAC source/destination
Indirecte: Autre réseau
Envoi à la passerelle par défaut
Trame avec MAC source et MAC passerelle

Protocoles de routage

RIP (IGP, Vecteur distance):
Métrique: nombre de sauts (max 15)
Mises à jour: 30s (UDP 520)
Split Horizon, Poison Reverse
Convergence lente (180s timeout)
Exemple de mise à jour RIP :

```
R1 reçoit de R2:
Destination  Métrique  Next Hop
192.168.3.0  2         R2

R1 met à jour sa table :
- Ajoute 1 au métrique (2+1=3)
- Met à jour si meilleure route
- Ignore si pire route
```

OSPF (IGP, État de liens):
Métrique: coût basé sur le débit
Algorithme Dijkstra (plus court chemin)
Zones et backbone (area 0)
Convergence rapide
BGP (EGP):

Routage inter-domaine (AS)
Politiques de routage
TCP 179, mises à jour incrémentielles

Algorithme de Dijkstra

Initialisation des coûts (0 pour le nœud source, ∞ autres)
Sélection du nœud avec le plus petit coût non finalisé
Mise à jour des coûts des voisins
Répéter jusqu'à ce que tous les nœuds soient finalisés

Exemple :

```
[A]--2--[B]--3--[D]
 |     /        /
 4   1       2
  \ /       /
   [C]-----/
```

Étapes de calcul :
1. Coûts initiaux: A=0, B=C=D=∞
2. Depuis A: B=2, C=4, D=∞
3. Depuis B: D=5 (via B)
4. Depuis C: D=6 (via C) $\rightarrow$ on garde D=5 (via B)

Tables de routage

Analyse d'une table de routage:

```
Destination  Gateway     Genmask       Flags Metric Ref  Use Iface
0.0.0.0      192.168.1.1 0.0.0.0       UG    100    0    0 eth0
192.168.1.0  0.0.0.0     255.255.255.0 U     100    0    0 eth0
10.0.0.0     192.168.1.2 255.0.0.0     UG    0      0    0 eth0
```

Ligne 1: Route par défaut via 192.168.1.1
Ligne 2: Réseau local 192.168.1.0/24
Ligne 3: Route statique vers 10.0.0.0/8

Types de routes:
Directes (C): interfaces connectées
Statiques (S): configurées manuellement
Dynamiques (R-RIP, O-OSPF, B-BGP)
Défaut (0.0.0.0/0): route par défaut

Configuration de routes statiques:

```
# Topologie :
# [PC1]---[R1]---[R2]---[PC2]
# 192.168.1.0/24   10.0.0.0/30   192.168.2.0/24

# Sur R1 :
# Interface vers PC1
ip addr add 192.168.1.1/24 dev eth0

# Interface vers R2
ip addr add 10.0.0.1/30 dev eth1

# Route vers le réseau de PC2
ip route add 192.168.2.0/24 via 10.0.0.2
```

Commandes utiles:

```
# Afficher la table de routage
ip route  # Linux
route print  # Windows

# Tracer la route
traceroute 8.8.8.8  # Linux
tracert 8.8.8.8    # Windows
```

ADRESSAGE IP & SOUS-RÉSEAUX

Adresses IPv4

Format : 32 bits (4 octets) en notation décimale pointée (ex: 192.168.1.1)
Classes d'adresses :
Classe A : 1.0.0.0 - 126.255.255.255 (masque par défaut: 255.0.0.0)
Classe B : 128.0.0.0 - 191.255.255.255 (masque: 255.255.0.0)
Classe C : 192.0.0.0 - 223.255.255.255 (masque: 255.255.255.0)
Classe D : 224.0.0.0 - 239.255.255.255 (multicast)
Classe E : 240.0.0.0 - 255.255.255.255 (expérimental)
Adresses spéciales :
0.0.0.0 : Hôte actuel (this network)
127.0.0.1 : Bouclage local (localhost)
169.254.0.0/16 : Lien-local (APIPA)
255.255.255.255 : Broadcast limité (local)
Adresses privées (RFC 1918) :
10.0.0.0/8 (10.0.0.0 - 10.255.255.255)
172.16.0.0/12 (172.16.0.0 - 172.31.255.255)
192.168.0.0/16 (192.168.0.0 - 192.168.255.255)

Masques et CIDR

Masque de sous-réseau : Sépare l'adresse IP en partie réseau et hôte
Notation CIDR (Classless Inter-Domain Routing) : /n (où n est le nombre de bits à 1 dans le masque)

Masque	CIDR	Nb adresses
255.255.255.255	/32	1
255.255.255.254	/31	2
255.255.255.252	/30	4
255.255.255.248	/29	8
255.255.255.240	/28	16
255.255.255.224	/27	32
255.255.255.192	/26	64
255.255.255.128	/25	128
255.255.255.0	/24	256
255.255.254.0	/23	512
255.255.252.0	/22	1024

Calcul du sous-réseau :
Déterminer le nombre de bits d'hôte nécessaires : $2^n \geq$ nb d'hôtes + 2
Calculer le masque : $32 - n$ bits à 1 à gauche
Déterminer le pas : $2^{(32-masque)}$
Les adresses de sous-réseau sont des multiples du pas

Exemple de sous-réseautage

Exercice : Découper 192.168.1.0/24 en 4 sous-réseaux
Solution :
4 sous-réseaux = 2 bits (car $2^2 = 4$)
Nouveau masque : /26 (24+2)
Pas = $2^{(32-26)} = 2^6 = 64$
Sous-réseaux :
192.168.1.0/26 (0-63)
192.168.1.64/26 (64-127)
192.168.1.128/26 (128-191)
192.168.1.192/26 (192-255)

VLSM (Variable Length Subnet Mask)

Permet d'utiliser des masques de longueur variable dans un même réseau
Avantage : Meilleure utilisation des adresses IP
Exemple :
Réseau principal : 192.168.1.0/24
Sous-réseau A : 50 hôtes $\rightarrow$ /26 (62 hôtes)
Sous-réseau B : 25 hôtes $\rightarrow$ /27 (30 hôtes)
Sous-réseau C : 10 hôtes $\rightarrow$ /28 (14 hôtes)

NAT (Network Address Translation)

Fonction : Permet à plusieurs hôtes d'un réseau privé de partager une adresse IP publique
Types :
NAT statique : Mappage 1:1 entre adresses privées et publiques
NAT dynamique : Pool d'adresses publiques partagées
PAT/Overload : Partage d'une seule IP publique avec des ports différents
Avantages :
Économie d'adresses IPv4
Sécurité accrue (masquage de la topologie interne)

FRAGMENTATION IPV4

Concepts de Base

MTU (Maximum Transmission Unit) : Taille maximale d'une trame sur un lien
Ethernet : 1500 octets (sans en-tête)
PPPoE : 1492 octets (à cause de l'encapsulation)
Réseaux sans fil : 2304 octets (parfois plus)
Path MTU : MTU la plus petite sur le chemin source-destination
DF (Don't Fragment) : Bit dans l'en-tête IP pour interdire la fragmentation

Champs d'En-tête IP pour la Fragmentation

Identification (16 bits) : Identifiant unique du datagramme original
Flags (3 bits) :
Bit 0 : Réservé (0)
Bit 1 (DF) : Don't Fragment
Bit 2 (MF) : More Fragments (1 si d'autres fragments suivent)
Fragment Offset (13 bits) : Position du fragment dans le datagramme original (en blocs de 8 octets)

Processus de Fragmentation

Un routeur reçoit un datagramme plus grand que le MTU du lien sortant
Si DF=1 : Le routeur supprime le paquet et envoie une ICMP "Fragmentation Needed"
Si DF=0 : Le routeur fragmente le datagramme en respectant :
Tous les fragments sauf le dernier doivent avoir une taille multiple de 8 octets
L'en-tête IP est copié dans chaque fragment (avec modifications)
Le champ Offset est mis à jour pour chaque fragment
Le champ MF est mis à 1 pour tous les fragments sauf le dernier
Le destinataire réassemble les fragments en utilisant les champs Identification, MF et Fragment Offset

Exemple de Fragmentation

Datagramme original : 4000 octets (en-tête IP : 20 octets, données : 3980 octets)
MTU : 1500 octets
Fragments :
1480 octets de données + 20 octets en-tête (MF=1, Offset=0)
1480 octets de données + 20 octets en-tête (MF=1, Offset=185)
1020 octets de données + 20 octets en-tête (MF=0, Offset=370)

Problèmes et Solutions

Perte de fragments : Si un fragment est perdu, tout le datagramme doit être retransmis
Reconstruction : Nécessite de la mémoire et du temps CPU
Path MTU Discovery : Mécanisme pour déterminer le Path MTU avant l'envoi
L'émetteur envoie avec DF=1
Si ICMP "Fragmentation Needed" est reçu, diminuer la taille et réessayer
Répéter jusqu'à ce que la taille optimale soit trouvée

Comparaison IPv4 vs IPv6

IPv4 : Fragmentation possible par les routeurs intermédiaires
IPv6 : Fragmentation uniquement par l'émetteur (extension header)
Utilise l'en-tête d'extension "Fragment"
Les routeurs intermédiaires ne fragmentent pas (envoient ICMPv6 "Packet Too Big")

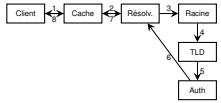
DNS

Port : 53 (UDP/TCP), 853 (DoT), 443 (DoH)
Hiérarchie : `www.sub.ex.com.` (point final = racine)
Enregistrements :
`A`: IPv4, `AAAA`: IPv6, `MX`: Mail, `CNAME`: Alias
`TXT`: SPF/DKIM, `NS`: Serveur, `PTR`: Inverse
`SOA`: Début d'autorité, `SRV`: Services
Exemple de zone DNS :
Requêtes :
Récursive: Résolveur fait tout
Itérative: Client fait les requêtes
Inverse: IP $\rightarrow$ nom (PTR)
Non-récursive: Cache uniquement
Cache :
TTL: 300s (5min) à 24h
Niveaux: Navigateur $\rightarrow$ OS $\rightarrow$ Résolveur
Commandes: `systemd-resolve -flush-caches`, `ipconfig /flushdns`
Sécurité :
DNSSEC :

```
# Vérifier la signature DNSSEC
dig +dnssec example.com

# Afficher la chaîne de confiance
delv +vtrace example.com
```

Chiffrement :
`DoT` (DNS over TLS) : Port 853
`DoH` (DNS over HTTPS) : Port 443
`DoQ` (DNS over QUIC) : Port 853 (UDP)
Exemple : `curl -H 'accept: application/dns-json' 'https://dns.google/resolve?name=example.com&type=A'`
Attaques :
`Empoisonnement` : Falsification des réponses DNS
`Amplification` : Utilisation de requêtes DNS pour du DDoS
`Hijacking` : Redirection du trafic DNS
`NXDOMAIN` : Attaque par déni de service
Contremesures :
Activer DNSSEC
Utiliser DoH/DoT
Configurer rate limiting
Mettre à jour régulièrement BIND/autres serveurs DNS



Tips

`+short` : Sortie minimale
`+trace` : Debug résolution
`@srv` : Serveur spécif.

PROTOCOLES DE ROUTAGE

Concepts de Base

Routage : Processus de sélection du meilleur chemin pour acheminer des paquets
Métrique : Valeur utilisée pour déterminer le meilleur chemin (saut, coût, bande passante, etc.)
Distance administrative : Priorité d'un protocole de routage (plus le nombre est bas, plus la priorité est élevée)
Table de routage : Base de données contenant les routes vers différents réseaux

Types de Routage

Routage direct : Pour les réseaux directement connectés
Routage statique : Routes configurées manuellement
Avantages : Faible surcharge, prévisible, sécurisé
Inconvénients : Non adapté aux grands réseaux, maintenance manuelle
Routage dynamique : Routes apprises automatiquement
Avantages : Adaptation aux changements, évolutif
Inconvénients : Surcharge réseau, complexité
Par défaut : Route de dernier recours (0.0.0.0/0)

Protocoles de Routage Intérieur (IGP)

RIP (Routing Information Protocol)
Distance administrative : 120
Métrique : Nombre de sauts (max 15)
Mises à jour : Toutes les 30 secondes (port 520)
Algorithme : Distance-vector (Bellman-Ford)
Version 2 prend en charge VLSM et authentification
OSPF (Open Shortest Path First)
Distance administrative : 110
Métrique : Coût basé sur la bande passante
Algorithme : Link-state (Dijkstra)
Hiérarchie : Zones (area) avec backbone (area 0)
Avantages : Évolutif, convergence rapide, support VLSM
EIGRP (Enhanced IGRP)
Distance administrative : 90 (interne), 170 (externe)
Métrique composite (bande passante, délai, charge, fiabilité, MTU)
Algorithme : DUAL (Diffusing Update Algorithm)
Propriétaire Cisco, maintenant partiellement ouvert (RFC 7868)

Protocoles de Routage Extérieur (EGP)

BGP (Border Gateway Protocol)
Distance administrative : 20 (eBGP), 200 (iBGP)
Métrique : Politiques (AS Path, Local Pref, MED, etc.)
Port TCP 179
Utilisé entre AS (Systèmes Autonomes) différents
Version 4 est la version actuelle (BGP-4)
Comparaison IGP vs EGP :
IGP : Au sein d'un même AS (RIP, OSPF, EIGRP)
EGP : Entre différents AS (BGP)

Commandes de Routage sous Linux

```
# Afficher la table de routage
ip route show

# Ajouter une route statique
ip route add 192.168.1.0/24 via 192.168.0.1

# Supprimer une route
ip route del 192.168.1.0/24

# Vider le cache de routage
ip route flush cache

# Afficher les voisins (ARP)
ip neigh

# Tracer la route
traceroute 8.8.8.8
mtr 8.8.8.8  # Alternative plus complète

# Afficher les statistiques de routage
ip -s route

# Configurer le forwarding IP
echo 1 > /proc/sys/net/ipv4/ip_forward  # IPv4
echo 1 > /proc/sys/net/ipv6/conf/all/forwarding  #
↪  IPv6

# Configurer NAT (exemple simple)
iptables -t nat -A POSTROUTING -o eth0 -j MASQUERADE
```

Bonnes Pratiques

Utiliser le routage statique pour les petits réseaux stables
Privilégier OSPF pour les grands réseaux
Toujours avoir une route par défaut configurée
Mettre en place des mécanismes de redondance
Sécuriser les mises à jour de routage (authentification)
Surveiller les métriques et les changements de topologie

FIREWALL

Objectifs

Confidentialité : Accès non autorisé
Intégrité : Modification non autorisée
Disponibilité : DDoS

Fonctions

Centre de décision (goulet d'étranglement)
Application politique de sécurité
Surveillance trafic
Limitation exposition
Limites :

Pas contre utilisateurs internes
Si trafic contourne (WLAN)
Protection limitée virus/vers

Stratégies

Refus par défaut : `"Deny all, permit by exception"`
Recommandé
Tout interdit sauf autorisé
Permission par défaut : `"Permit all, deny by exception"`
Déconseillé
Tout permis sauf interdit
Liste blanche : Connexions autorisées

Types de firewalls

Filtre de paquets (L3/4) :
Filtrage basé sur les en-têtes (IP/Ports/Protocole)
Ex: iptables, pfSense
Peut bloquer des services en filtrant les ports
Exemple iptables :

```
# Politique par défaut
iptables -P INPUT DROP
iptables -P FORWARD DROP
iptables -P OUTPUT ACCEPT

# Autoriser le trafic local
iptables -A INPUT -i lo -j ACCEPT
iptables -A OUTPUT -o lo -j ACCEPT

# Autoriser le trafic établi
iptables -A INPUT -m conntrack \
  --ctstate ESTABLISHED,RELATED -j ACCEPT

# Autoriser SSH (port 22)
iptables -A INPUT -p tcp --dport 22 -j ACCEPT

# Autoriser HTTP/HTTPS
iptables -A INPUT -p tcp --dport 80 -j ACCEPT
iptables -A INPUT -p tcp --dport 443 -j ACCEPT
```

Stateful (L4) :
Suit l'état des connexions (SYN/ACK/FIN)
Protection anti-spoofing
Meilleure sécurité que le simple filtrage
Application (L7) :
Analyse le contenu des paquets
Détection d'attaques applicatives
Ex: FortiGate, Palo Alto, Check Point
Proxy :
Intermédiaire pour les connexions
Peut faire du filtrage de contenu
Cache les adresses IP internes

Architectures courantes

Routeur écran simple :
Niveau de sécurité basique
Pas de défense en profondeur
Solution économique mais peu sécurisée
DMZ (Zone démilitarisée) :
Zone tampon entre Internet et réseau interne
Architecture à deux pare-feux (interne et externe)
Solution standard en entreprise
Serveur bastion :
Hôte durci exposé à Internet
Souvent utilisé comme point d'entrée unique
Doit être renforcé au maximum
Zero Trust :
Principe : `Never trust, always verify`
Vérification continue des utilisateurs et appareils
Accès minimal nécessaire (principe du moindre privilège)

Règles de filtrage

```
# Politiques par défaut
iptables -P INPUT DROP
iptables -P FORWARD DROP
iptables -P OUTPUT ACCEPT

# Autoriser le trafic local
iptables -A INPUT -i lo -j ACCEPT

# Connexions établies
iptables -A INPUT -m conntrack --ctstate
↪  ESTABLISHED,RELATED -j ACCEPT

# Exemple : Autoriser SSH depuis une IP spécifique
iptables -A INPUT -p tcp --dport 22 -s 192.168.1.100
↪  -j ACCEPT

# Exemple : Autoriser le trafic HTTP/HTTPS
iptables -A INPUT -p tcp --dport 80 -j ACCEPT
iptables -A INPUT -p tcp --dport 443 -j ACCEPT

# Protection contre les scans de ports
iptables -N PORTSCAN
iptables -A PORTSCAN -m limit --limit 1/s -j LOG
↪  --log-prefix "Portscan: "
iptables -A PORTSCAN -j DROP
```

Règles bidirectionnelles

Pour un service comme SSH, deux règles sont nécessaires :

Entrante : autoriser le trafic vers le port 22
Sortante : autoriser les réponses (généralement couvert par ESTABLISHED)

Protections avancées

Brute force :
Outils : fail2ban, sshguard
Limite des tentatives de connexion
Spoofing :
Reverse Path Forwarding (RPF)
Filtrage entrée/sortie
Vérification des adresses IP sources
DoS/DDoS :
Limite du nombre de connexions
SYN cookies
Détection des flux anormaux
Injection :
Web Application Firewall (WAF)
Filtrage couche application (L7)
Détection des signatures d'attaques

Bonnes pratiques

Mises à jour régulières des règles et du système
Journalisation et surveillance des événements
Tests de pénétration réguliers
Documentation des règles et procédures
Formation des utilisateurs aux bonnes pratiques
Plan de réponse aux incidents

SÉCURITÉ RÉSEAU AVANCÉE

Concepts de Sécurité

Disponibilité : Assurer l'accès aux ressources
Intégrité : Garantir l'exactitude des données
Confidentialité : Limiter l'accès aux seuls utilisateurs autorisés
Authentification : Vérifier l'identité des entités
Non-répudiation : Empêcher le déni des actions

Menaces Courantes

Attaques par déni de service (DoS/DDoS)
SYN Flood, Ping of Death, Smurf, UDP Flood
Attaques par réflexion/amplification
Attaques par l'homme du milieu (MitM)
ARP Spoofing, DNS Spoofing, SSL Stripping
Écoute clandestine (Sniffing)
Usurpation d'identité (Spoofing)
Attaques par rejeu (Replay)
Ingénierie sociale
Exploitation des vulnérabilités logicielles

Techniques de Sécurité

Chiffrement :
Symétrique (AES, DES, 3DES, RC4)
Asymétrique (RSA, ECC, ElGamal)
Hachage (SHA, MD5 - à éviter, RIPEMD)
Protocoles sécurisés :
SSL/TLS (HTTPS, SMTPS, IMAPS, etc.)
IPsec (AH, ESP, IKE)
SSH (au lieu de Telnet)
SFTP/SCP (au lieu de FTP)
Authentification :
802.1X (EAP, PEAP, EAP-TLS)
RADIUS, TACACS+
Mots de passe forts + 2FA

Pare-feu (Firewall)

Types :
Pare-feu sans état (stateless)
Pare-feu avec état (stateful)
Pare-feu applicatif (proxy)
Pare-feu de nouvelle génération (NGFW)
Technologies :
Filtrage de paquets
Inspection avec état (stateful inspection)
Détection/prévention d'intrusion (IDS/IPS)
Prévention des fuites de données (DLP)
Implémentation :
iptables/nftables (Linux)
Windows Firewall
Pare-feu matériels (Cisco ASA, Fortinet, etc.)

Réseaux Privés Virtuels (VPN)

Types :
Accès à distance (client-à-site)
Site-à-site (intranet/extranet)
SSL/TLS vs IPsec
Protocoles :
IPsec (IKEv1/IKEv2, ESP/AH)
OpenVPN
WireGuard
L2TP/IPsec
PPTP (obsolète, non sécurisé)
Implémentation :
Serveur VPN (OpenVPN, StrongSwan, WireGuard)
Clients VPN natifs ou logiciels dédiés
Authentification par certificats ou clés pré-partagées

Surveillance et Détection

SIEM (Security Information and Event Management)
Agrégation des logs
Corrélation d'événements
Détection des menaces
IDS/IPS :
Basé sur signature (Snort, Suricata)
Basé sur le comportement/anomalies
Honeypots/honeynets
Analyse de vulnérabilités :
Scanners de vulnérabilités (Nessus, OpenVAS)
Tests d'intrusion
Audits de sécurité

Bonnes Pratiques

Appliquer le principe du moindre privilège
Maintenir les systèmes à jour
Mettre en œuvre une défense en profondeur
Former les utilisateurs aux bonnes pratiques
Sauvegarder régulièrement les données
Documenter les politiques et procédures
Tester régulièrement la sécurité
Avoir un plan de réponse aux incidents

PROTOCOLES TRANSPORT

TCP/UDP

TCP (Taille en-tête: 20-60 octets):
Orienté connexion, fiable, livraison ordonnée des segments.
Numéros de séquence (SEQ): Utilisés pour ordonner les segments et détecter les pertes.
Numéros d'acquittement (ACK): Indiquent le prochain octet attendu, confirmant la réception des données.
Flags TCP:
`SYN` (Synchronize): Initie une connexion.
`ACK` (Acknowledge): Acquitte la réception de données/flags.
`FIN` (Finish): Termine une connexion (fermeture propre).
`RST` (Reset): Réinitialise une connexion (fermeture abrupte, erreur).
`PSH` (Push): Demande à la couche application de traiter les données immédiatement.
`URG` (Urgent): Indique des données urgentes (pointeur urgent).
Établissement de connexion (3-way handshake):
Client → Serveur: `SYN` (seq=x)
Serveur → Client: `SYN, ACK` (seq=y, ack=x+1)
Client → Serveur: `ACK` (seq=x+1, ack=y+1)
Fermeture de connexion (4-way handshake) (chaque côté ferme indépendamment):
Initiateur → Destinataire: `FIN, ACK` (seq=u, ack=v)
Destinataire → Initiateur: `ACK` (seq=v, ack=u+1)
Destinataire → Initiateur: `FIN, ACK` (seq=v, ack=u+1) (après avoir fini d'envoyer ses propres données)
Initiateur → Destinataire: `ACK` (seq=u+1, ack=v+1)
Fenêtre glissante (Sliding Window): Mécanisme de contrôle de flux. L'émetteur ne peut envoyer que la quantité de données spécifiée par la fenêtre de réception du destinataire. La taille de la fenêtre est annoncée dans l'en-tête TCP.
Contrôle de congestion: Mécanismes (ex: Slow Start, Congestion Avoidance, Fast Retransmit/Recovery) pour éviter de surcharger le réseau. Adapte la quantité de données envoyées en fonction des conditions du réseau (pertes, délais).
UDP (Taille en-tête: 8 octets):
Sans connexion, non fiable (pas de garantie de livraison, d'ordre ou d'unicité).
Faible latence, faible surcharge (overhead).
Utilisé pour : DNS (port 53), DHCP (ports 67, 68), SNMP (ports 161, 162), VoIP, streaming vidéo, jeux en ligne.
En-tête simple: Ports source/destination, longueur, checksum (optionnel en IPv4, obligatoire en IPv6 si utilisé par la couche sup).

Multiplexage et Sockets

Multiplexage: Permet à plusieurs applications sur un hôte de communiquer sur le réseau simultanément en utilisant différents numéros de port.
Démultiplexage: Le récepteur utilise les numéros de port pour diriger les segments entrants vers la bonne application/processus.
Socket: Point de terminaison d'une communication. Identifié par une paire (adresse IP, numéro de port). Ex: `192.168.1.10:80`.

Ports courants

20/21: FTP	161/2: SNMP
22: SSH	389: LDAP
25: SMTP	443: HTTPS
53: DNS	465: SMTPS
67/68: DHCP	993: IMAPS
80: HTTP	995: POP3S
110: POP3	3389: RDP
123: NTP	3306: MySQL
143: IMAP	5432: PostgreSQL

Commandes

```
# Connexions
ss -tulnp  # écoute
ss -tunp   # actif

# Test et analyse
nc -zv hote port
tcpdump -ni eth0 'port 80'
```

OUTILS DE DIAGNOSTIC

Commandes essentielles

`ping -c 4 google.com`: Test de connectivité
`traceroute -n google.com`: Tracer la route
`ss -tulnp`: Ports en écoute
`ss -tunp`: Connexions actives
`dig google.com +short`: Résolution DNS
`tcpdump -ni eth0 'port 80'`: Analyse trafic
`nc -zv hote port`: Tester un port

Options clés

`ping`: `-c` (compteur), `-i` (intervalle), `-s` (taille)
`ss`: `-t` (TCP), `-u` (UDP), `-l` (écoute), `-n` (pas de DNS)
`dig`: `+short` (court), `+trace` (trace complète)
`tcpdump`:
`-n`: Ne pas résoudre les noms d'hôtes (affiche les IPs).
`-nn`: Ne pas résoudre les noms d'hôtes ni les noms de ports.
`-i interface`: Spécifie l'interface d'écoute (ex: `eth0`, `any`).
`-c count`: Capture `count` paquets puis s'arrête.
`-w fichier.pcap`: Écrit les paquets bruts dans un fichier (format pcap).
`-r fichier.pcap`: Lit les paquets depuis un fichier.
`-X`: Affiche le contenu du paquet en hexa et ASCII (utile pour l'analyse de protocole).
`-v, -vv, -vvv`: Augmente la verbosité.
Filtres BPF (Berkeley Packet Filter):
`host 1.2.3.4`: Trafic impliquant l'hôte 1.2.3.4 (source ou destination).
`src host 1.2.3.4`: Trafic depuis l'hôte 1.2.3.4.
`dst host 1.2.3.4`: Trafic vers l'hôte 1.2.3.4.
`net 192.168.1.0/24`: Trafic vers ou depuis le réseau spécifié.
`port 80`: Trafic sur le port 80 (source ou destination).
`tcp port 22`: Trafic TCP sur le port 22.
`udp port 53`: Trafic UDP sur le port 53.
`icmp`: Paquets ICMP.
Combinaisons: `host 1.2.3.4 and port 80`, `src 1.2.3.4 and not dst port 22`.

Autres outils et précisions

`ip addr` (ou `ifconfig` - déprécié): Affiche/manipule les adresses IP et les interfaces réseau.
`ip route` (ou `route` - déprécié): Affiche/manipule la table de routage IP.
`ip neigh` (ou `arp -a` pour IPv4): Affiche/manipule la table des voisins (cache ARP pour IPv4, cache NDP pour IPv6).
`ss` (remplace `netstat`): Outil d'investigation des sockets.
`netstat -tulnp` est souvent remplacé par `ss -tulnp`.
`curl -I url` ou `curl -v url`: Récupère les en-têtes HTTP/S ou des informations détaillées sur la connexion.
`whois domaine.com`: Obtient des informations sur l'enregistrement d'un nom de domaine.
`iptables -L -n -v` / `ip6tables -L -n -v`: Liste les règles du pare-feu Netfilter (pour IPv4 et IPv6 respectivement).
`nc -lvp port` (Netcat): Outil polyvalent pour lire/écrire des données sur le réseau. Utile pour tester des ports, transférer des fichiers, créer des backdoors simples.
`mtr hote` (My Traceroute): Combine `ping` et `traceroute` pour un diagnostic continu de la route.
`iperf3 -s` (serveur) / `iperf3 -c serveur`: Mesure la bande passante TCP/UDP entre deux hôtes.
`nmap [options] cible`: Scanner de ports et de réseau puissant.
Ex: `nmap -sS -p 1-1000 hote` (Scan SYN des 1000 premiers ports).
Attention : Utiliser avec précaution et uniquement sur des réseaux autorisés.

Exemples

```
# Vérifier la connectivité
ping -c 4 -i 0.5 google.com

# Ports ouverts sur un hôte
nc -zv hote 22 80 443
```

```
# Requête DNS complète
dig +trace google.com

# Statistiques réseau
ss -s

# Surveillance trafic en direct
tcpdump -n 'tcp port 80' -c 10
```

IPV6

Objectifs et caractéristiques

Résoudre la pénurie d'adresses IPv4
Simplification de l'en-tête pour un routage plus efficace
Meilleure prise en charge de la mobilité et de la sécurité (IPsec)
Configuration automatique des adresses (SLAAC)
Meilleure gestion du multicast

Adressage IPv6

Structure : 128 bits (vs 32 en IPv4)
Notation :
8 groupes de 4 chiffres hexadécimaux séparés par ':'
Exemple : `2001:0db8:85a3:0000:0000:8a2e:0370:7334`
Règles de simplification :
Suppression des zéros non significatifs :
`2001:db8:85a3:0:0:8a2e:370:7334`
Remplacement d'une suite de zéros par '::' (une seule fois) : `2001:db8:85a3::8a2e:370:7334`
Types d'adresses :
`2000::/3` : Unicast Global (adresses publiques)
Préfixe global : 48b (fourni par le FAI)
ID de sous-réseau : 16b (pour segmentation interne)
ID d'interface : 64b (EUI-64 ou aléatoire)
`fe80::/10` : Link-Local (valide uniquement sur le lien local)
Générées automatiquement par chaque interface
Utilisées pour la découverte des voisins et la configuration automatique
Ne sont jamais routées
`fc00::/7` : Unique Local (adresses privées)
`fc00::/8` : Réservé (usage futur)
`fd00::/8` : Adresses locales uniques (générées aléatoirement)
Équivalent aux adresses privées IPv4 (10.0.0.0/8, 172.16.0.0/12, 192.168.0.0/16)
Non routables sur Internet
`ff00::/8` : Multicast
`ff02::1` : Tous les nœuds du lien local
`ff02::2` : Tous les routeurs du lien local
`ff02::5` : Tous les routeurs OSPFv3
`ff02::1:2` : Tous les agents DHCPv6
`ff02::1:ff00:0/104` : Adresses de multicast sollicitées (utilisées pour DAD)
Adresses spéciales :
`::1/128` : Adresse de bouclage (loopback)
`::/128` : Adresse non spécifiée
`::/0` : Route par défaut
`2001:db8::/32` : Documentation et exemples (RFC 3849)
`2002::/16` : 6to4 (obsolète, RFC 7526)

En-tête IPv6

Simplifié par rapport à IPv4 pour un traitement plus rapide par les routeurs.
Longueur fixe de 40 octets (contre 20-60 octets pour IPv4).
Champs moins utilisés ou optionnels déplacés vers des **en-têtes d'extension** :
Permet de garder l'en-tête de base concis et efficace.
Traités uniquement par le nœud de destination (sauf Hop-by-Hop).
Exemples : Hop-by-Hop Options, Routing, Fragment, Authentication (AH), Encapsulating Security Payload (ESP), Destination Options.
Somme de contrôle (checksum) de l'en-tête a été **éliminée** :
Les couches supérieures (ex: TCP, UDP) et la couche liaison (ex: Ethernet) ont leurs propres mécanismes de contrôle d'erreur robustes.
Réduit le temps de traitement dans les routeurs (pas besoin de recalculer le checksum à chaque saut si le TTL est modifié, ce qui n'est plus le cas avec Hop Limit).

Mécanismes de configuration

SLAAC (Stateless Address Autoconfiguration)
Génération d'une adresse link-local (fe80::/64 + ID d'interface)
Vérification d'unicité avec DAD (Duplicate Address Detection)
Envoi d'une Sollicitation de Routeur (RS, ICMPv6 type 133)
Réception d'une Annonce de Routeur (RA, ICMPv6 type 134) contenant :
Préfixe réseau global
Longueur du préfixe (généralement /64)
Durée de vie (valid lifetime)
Drapeaux :
A (Autonomous flag) : Utiliser le préfixe pour SLAAC
M (Managed flag) : Utiliser DHCPv6 pour l'adressage
O (Other flag) : Utiliser DHCPv6 pour d'autres paramètres (DNS, etc.)
Génération d'une adresse globale avec le préfixe reçu
Vérification d'unicité avec DAD pour la nouvelle adresse
DHCPv6
Deux modes de fonctionnement :
`M=1, O=0` : Configuration d'adresse uniquement via DHCPv6
`M=0, O=1` : Autres paramètres (DNS, NTP, etc.) via DHCPv6
Messages principaux :
Solicit (1) : Client recherche des serveurs DHCPv6
Advertise (2) : Réponse des serveurs disponibles
Request (3) : Demande d'adresse ou de paramètres
Reply (7) : Réponse du serveur avec configuration
EUI-64 modifié
Méthode pour générer un identifiant d'interface unique à partir d'une adresse MAC
Étapes :
Séparer l'adresse MAC en deux parties (24 bits + 24 bits)
Insérer `fffe` au milieu
Inverser le bit U/L (Universal/Local) du premier octet (7ème bit)
Exemple : MAC `00:11:22:33:44:55` ⟶ `0211:22ff:fe33:4455`
Alternative : Utilisation d'identifiants d'interface aléatoires pour la confidentialité
Messages principaux DHCPv6 :
`Solicit` : Client recherche des serveurs DHCPv6 (multicast ff02::1:2).
`Advertise` : Serveur(s) répond(ent) au client (unicast).
`Request` : Client choisit un serveur et demande une configuration (unicast).
`Reply` : Serveur confirme et fournit la configuration (unicast).

NDP (Neighbor Discovery Protocol)

Remplace ARP (résolution d'adresse) et ICMP Redirect d'IPv4
Fonctionnalités principales :
Découverte des routeurs (RA/RS)
Résolution d'adresses (NS/NA)
Détection d'adresses dupliquées (DAD)
Détection de voisins non accessibles (NUD)
Découverte des préfixes et paramètres de liaison
Messages ICMPv6 :
`133` : Router Solicitation (RS)
`134` : Router Advertisement (RA)
`135` : Neighbor Solicitation (NS)
`136` : Neighbor Advertisement (NA)
`137` : Redirect (redirection de route)
Protection contre le DAD :
Avant d'utiliser une adresse, un nœud envoie une NS à l'adresse de multicast sollicitée
Si aucune NA n'est reçue, l'adresse est considérée comme unique

Fragmentation en IPv6

Contrairement à IPv4, seuls les nœuds sources peuvent fragmenter les paquets
Si un routeur reçoit un paquet trop grand pour le MTU du lien suivant :
Il supprime le paquet
Envoie un message ICMPv6 "Paquet trop grand" (Type 2, Code 0)
Le message contient le MTU du lien suivant
La source utilise la découverte de MTU pour déterminer la taille maximale des paquets
L'en-tête de base IPv6 ne contient pas de champ pour la fragmentation
La fragmentation utilise un en-tête d'extension "Fragment" (si nécessaire)
MTU minimum pour IPv6 est de `1280 octets`.
Tous les liens IPv6 doivent être capables de transmettre des paquets de cette taille sans nécessiter de fragmentation par les routeurs intermédiaires.
Les hôtes peuvent utiliser la Découverte de MTU de Chemin (Path MTU Discovery - PMTUD) pour trouver le MTU optimal pour une destination donnée.

Mécanismes de transition

Dual Stack : Exécution simultanée d'IPv4 et IPv6
Solution la plus simple et recommandée
Nécessite que tous les équipements prennent en charge les deux protocoles
Tunneling : Encapsulation d'IPv6 dans IPv4
6in4 : Tunnels manuels (RFC 4213)
6to4 : Tunnels automatiques utilisant l'adresse IPv4 (obsolète, RFC 7526)
Teredo : Tunnels à travers NAT (RFC 4380)
Translation : Traduction entre IPv4 et IPv6
NAT64 : Permet aux hôtes IPv6 d'accéder aux ressources IPv4
DNS64 : Génère des adresses IPv6 synthétiques pour les noms IPv4
464XLAT : Permet aux applications IPv4 d'accéder à IPv6 via NAT64
Proxying : Utilisation de serveurs mandataires
SOCKS64 : Proxy applicatif pour la traduction
Bump-in-the-Host (BIH) : Traduction au niveau hôte

Commandes utiles

```
# Afficher les adresses IPv6
ip -6 addr show

# Afficher les routes IPv6
ip -6 route show

# Tester la connectivité
ping6 ipv6.google.com
ping6 -c 4 2001:4860:4860::8888  # Google DNS IPv6

# Traceroute IPv6
traceroute6 ipv6.google.com
tracepath6 2001:4860:4860::8888

# Vérifier les voisins IPv6 (équivalent de arp -a)
ip -6 neigh show

# Désactiver temporairement IPv6 (si nécessaire)
echo 1 > /proc/sys/net/ipv6/conf/all/disable_ipv6

# Activer le forwarding IPv6 (pour les routeurs)
echo 1 > /proc/sys/net/ipv6/conf/all/forwarding

# Afficher les groupes multicast
netstat -gn

# Dépanner avec tcpdump (filtre pour IPv6)
tcpdump -i eth0 ip6
tcpdump -i eth0 'icmp6 && (ip6[40] == 128 || ip6[40]
↪  == 129)'  # Ping6
```

POINTS IMPORTANTS MANQUANTS

1. Modèles de Référence (OSI et TCP/IP)

Modèle OSI : 7 couches (Physique, Liaison, Réseau, Transport, Session, Présentation, Application)
Modèle TCP/IP : 4 couches (Accès réseau, Internet, Transport, Application)
Entités paires : Communication logique entre couches équivalentes
Couches traitées par :
Nœuds terminaux : Toutes les couches
Routeurs : Jusqu'à la couche Réseau
Switches : Jusqu'à la couche Liaison

2. Paramètres de Performance

Durée d'un bit = $\frac{1}{R}$ (où R est le débit en bits/s)
Délai total = $T_{trans} + T_{prop} + T_{file} + T_{trait}$
T_{trans} : Transmission
T_{prop} : Propagation
T_{file} : File d'attente
T_{trait} : Traitement
Sur plusieurs tronçons : Sommation des délais de chaque tronçon

3. Commutation

Paquets vs Circuits :
Avantages paquets : Meilleure utilisation BP, tolérance aux pannes
Avantages circuits : Garantie de débit, délai constant
Mode **Cut-Through** : Commutation immédiate après lecture de l'adresse

4. Réseaux Locaux

Switch :
Apprentissage : Source MAC + port d'entrée
Inondation si MAC inconnue ou broadcast
Ethernet :
Padding si trame < 64 octets
Type : Protocole de couche supérieure (0x0800=IPv4, 0x86DD=IPv6)
ARP : Résolution IP vers MAC, cache ARP

5. Réseaux IP (v4/v6)

Sous-réseaux :
ID Réseau = IP ET Masque
Broadcast = IP OU (NOT Masque)
Plage valide : ID Réseau +1 à Broadcast -1
TTL : Décrémenté par chaque routeur, paquet supprimé si 0
Fragmentation :
IPv4 : Possible sur les routeurs
IPv6 : Uniquement par la source

6. IPv6 Spécifique

Adresses spéciales :
::1 = localhost
FF02::1 = Tous les nœuds
EUI-64 : Inversion bit U/L + insertion FFFE
DAD : Vérification unicité adresse
NDP : Remplace ARP (RS/RA/NS/NA)
Pas de NAT nécessaire

7. Routage

RIP :
Mise à jour : Si meilleur chemin (métrique + 1 < coût actuel)
Limites : Lente convergence, count-to-infinity
Améliorations : Split Horizon, Poison Reverse
OSPF :
État des liens (LSA)
Dijkstra pour calculer les plus courts chemins

8. DNS/DHCP

DNS :
MX = Serveur mail
Non-authoritative = Réponse mise en cache
DHCP : Découvre, Offre, Demande, Accusé (DORA)

9. Sécurité

DMZ : Zone démilitarisée pour services exposés
Whitelist : Tout interdit sauf ce qui est explicitement autorisé

10. Transport (TCP/UDP)

UDP :
Pour temps réel, tolérance aux pertes
Moins d'overhead que TCP
TCP :
Retransmission sur : Timeout ou 3 ACK dupliqués
Contrôle de congestion : Slow Start, Congestion Avoidance (AIMD)
Contrôle de flux : Fenêtre glissante adaptée au récepteur

ANNEXES

Table des codes ICMP

Type	Code	Description
0	0	Echo Reply (réponse ping)
3	0	Destination Network Unreachable
	1	Destination Host Unreachable
	2	Destination Protocol Unreachable
	3	Destination Port Unreachable
	4	Fragmentation Needed and DF set
	5	Source Route Failed
	6	Destination Network Unknown
	7	Destination Host Unknown
5	0	Redirect Datagram for the Network
	1	Redirect Datagram for the Host
8	0	Echo Request (ping)
11	0	TTL expired in transit
	1	Fragment reassembly time exceeded

Comparaison OSPF vs RIP

OSPF	RIP
Algorithme de type état de liens	Algorithme de type vecteur de distance
Métrique basée sur le coût (bande passante)	Métrique basée sur le nombre de sauts
Pas de limite de sauts	Limite de 15 sauts
Convergence rapide	Convergence lente
Consomme plus de CPU/mémoire	Léger en ressources
Configuration complexe	Configuration simple
Convient aux grands réseaux	Pour petits réseaux simples

Commandes réseau rapides

Test de connectivité:
`ping <cible>` : Test ICMP de base
`ping6 <cible>` : Test ICMP IPv6
`traceroute <cible>` : Affiche le chemin réseau
`mtr <cible>` : Combinaison ping + traceroute
Informations réseau:
`ip addr show` : Affiche les adresses IP
`ip route show` : Affiche la table de routage
`ss -tuln` : Affiche les ports en écoute
`ip -6 addr` : Affiche les adresses IPv6
Dépannage DNS:
`dig <domaine>` : Requête DNS détaillée
`nslookup <domaine>` : Outil d'interrogation DNS
`host <domaine>` : Résolution de nom simple
Analyse réseau:
`tcpdump -i eth0 -n` : Capture de paquets
`netstat -tuln` : Statistiques réseau
`iftop` : Bande passante en temps réel