

LEXIQUE SQL

v1.0.0 | 20/01/2025 | Auteur : Bauer Baptiste

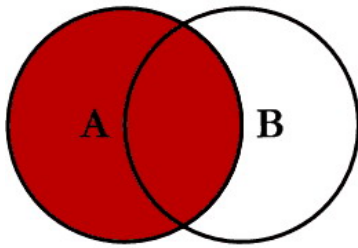
Chapitre | Durée de réalisation : 3

Table des matières

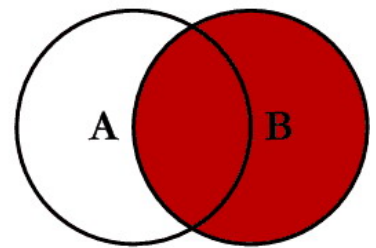
1. 3.3.a) Le coût moyen des réservations par client parisien (nom, adresse) pour l'année 2023 (trié par ordre décroissant de ce coût).....	1
2. 3.3.b) Les camions (immatriculation, date de mise en circulation), triés par ordre alphabétique des types, qui n'ont jamais été réservés et qui ne roulent pas au gazole.....	2
3. 3.3.c) Pour chaque partenaire (raison sociale, nom et adresse mail du contact), le coût total des réservations 2023 lorsque ce coût est inférieur à 40 000 €.	2
4. 3.3.d) La liste chronologique des réservations 2024 facturées mais non encore réglées, avec le nom et le numéro de téléphone des clients concernés.....	3
Résumé des fonctions d'agrégation SQL.....	4
1. Fonctions d'agrégation de base	5
2. Clauses associées	6
3. Fonctions d'agrégation avancées	7
4. Utilisation avec DISTINCT	8
Fonctions usuelles en SQL	9
5. Fonctions de chaîne de caractères.....	10
6. Fonctions numériques	11
7. Fonctions de date	12
8. Fonctions conditionnelles	13
9. Fonctions de conversion	14
Modèles de transformation numérique avec exemples détaillés.....	15
10. 1. Modèle de maturité digitale (MIT Sloan & Capgemini)	16
11. 2. Les 9 éléments de Gartner	17
12. 3. Six dimensions de Rogers	18
13. 4. Modèle RMMM (adapté).....	19
14. 5. Framework d'Altimeter	20
15. 6. Modèle 7S de McKinsey	21
16. 7. Matrice de Wade	22
17. 8. CMMI adapté au digital	23
IPv6 : Vue d'ensemble	25
18. Caractéristiques principales	26
18.1. Format d'adresse	26
18.2. Espace d'adressage	26
18.3. Types d'adresses.....	26
19. Avantages techniques.....	27
20. Défis de déploiement	28
21. Types d'adresses courants.....	29
22. Masques de sous-réseau en IPv6	30
22.1. Notation	30

22.2. Différences avec IPv4	30
22.3. Préfixes courants en IPv6	30
22.4. Exemple pratique	30
23. Principes fondamentaux de l'architecture Zero Trust	31
24. Implications pour la sécurité des SI	32

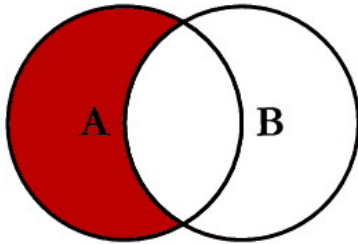
SQL JOINS



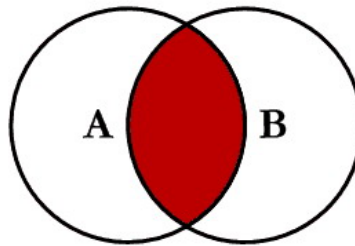
```
SELECT <select_list>
FROM TableA A
LEFT JOIN TableB B
ON A.Key = B.Key
```



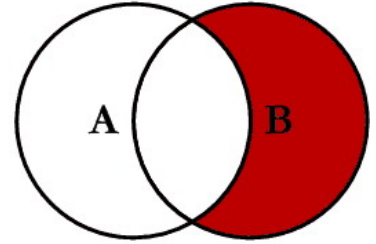
```
SELECT <select_list>
FROM TableA A
RIGHT JOIN TableB B
ON A.Key = B.Key
```



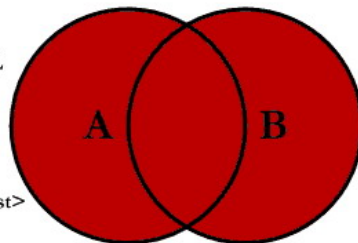
```
SELECT <select_list>
FROM TableA A
LEFT JOIN TableB B
ON A.Key = B.Key
WHERE B.Key IS NULL
```



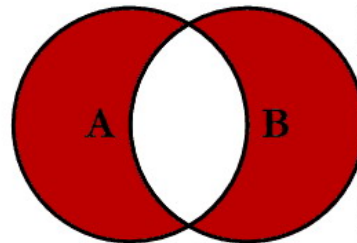
```
SELECT <select_list>
FROM TableA A
INNER JOIN TableB B
ON A.Key = B.Key
```



```
SELECT <select_list>
FROM TableA A
RIGHT JOIN TableB B
ON A.Key = B.Key
WHERE A.Key IS NULL
```



```
SELECT <select_list>
FROM TableA A
FULL OUTER JOIN TableB B
ON A.Key = B.Key
```



```
SELECT <select_list>
FROM TableA A
FULL OUTER JOIN TableB B
ON A.Key = B.Key
WHERE A.Key IS NULL
OR B.Key IS NULL
```

© C.L. Moffatt, 2008

Je vais vous proposer une correction des requêtes SQL demandées dans le dossier 3, question 3.3 du sujet TransCo.

1. 3.3.a) Le coût moyen des réservations par client parisien (nom, adresse) pour l'année 2023 (trié par ordre décroissant de ce coût).

```
SELECT c.nomCli, c.adresseCli, AVG(r.montantResa) AS coutMoyen
FROM CLIENT c
JOIN RESERVATION r ON c.idCli = r.idCli
WHERE c.villeCli = 'Paris'
AND YEAR(r.dateResa) = 2023
GROUP BY c.idCli, c.nomCli, c.adresseCli
ORDER BY coutMoyen DESC;
```

Cette requête : - Sélectionne le nom et l'adresse des clients - Joint la table CLIENT avec RESERVATION - Filtre les clients parisiens et les réservations de 2023 - Calcule le coût moyen des réservations pour chaque client - Ordonne les résultats par coût moyen décroissant

2. 3.3.b) Les camions (immatriculation, date de mise en circulation), triés par ordre alphabétique des types, qui n'ont jamais été réservés et qui ne roulent pas au gazole.

```
SELECT cp.immat, cp.dateCirculation
FROM CAMIONPARTENAIRE cp
JOIN TYPECAMION tc ON cp.idType = tc.idType
WHERE tc.carburant <> 'Gazole'
AND cp.idCamion NOT IN (
    SELECT idCamion
    FROM RESERVATION
)
ORDER BY tc.libelleType;
```

Correction : Je constate une erreur dans ma requête. Le champ `dateCirculation` n'est pas présent dans le schéma fourni pour CAMIONPARTENAIRE. Ce champ doit faire partie des informations à ajouter pour les camions de TransCo dans la question 3.4. Voici la requête corrigée :

```
SELECT cp.immat
FROM CAMIONPARTENAIRE cp
JOIN TYPECAMION tc ON cp.idType = tc.idType
WHERE tc.carburant <> 'Gazole'
AND cp.idCamion NOT IN (
    SELECT idCamion
    FROM RESERVATION
)
ORDER BY tc.libelleType;
```

Cette requête : - Sélectionne l'immatriculation des camions - Joint CAMIONPARTENAIRE avec TYPECAMION - Filtre les camions qui n'utilisent pas de gazole - Exclut les camions qui ont été réservés au moins une fois - Ordonne les résultats par le libellé du type de camion

3. 3.3.c) Pour chaque partenaire (raison sociale, nom et adresse mail du contact), le coût total des réservations 2023 lorsque ce coût est inférieur à 40 000 €.

```
SELECT p.raisonSocialePartenaire, p.nomContact, p.mailContact,
       SUM(r.montantResa) AS coutTotal
FROM PARTENAIRE p
JOIN CAMIONPARTENAIRE cp ON p.idPartenaire = cp.idPartenaire
JOIN RESERVATION r ON cp.idCamion = r.idCamion
```

```

WHERE YEAR(r.dateResa) = 2023
GROUP BY p.idPartenaire, p.raisonSocialePartenaire, p.nomContact, p.mailContact
HAVING SUM(r.montantResa) < 40000
ORDER BY p.raisonSocialePartenaire;

```

Cette requête : - Sélectionne la raison sociale, le nom et l'email du contact pour chaque partenaire - Effectue des jointures entre PARTENAIRE, CAMIONPARTENAIRE et RESERVATION - Filtre les réservations de l'année 2023 - Calcule le coût total des réservations pour chaque partenaire - Ne garde que les partenaires dont le coût total est inférieur à 40 000 € - Ordonne les résultats par raison sociale (ajout personnel pour une meilleure lisibilité)

4. 3.3.d) La liste chronologique des réservations 2024 facturées mais non encore réglées, avec le nom et le numéro de téléphone des clients concernés.

```

SELECT r.idResa, r.dateResa, c.nomCli, c.telCli
FROM RESERVATION r
JOIN CLIENT c ON r.idCli = c.idCli
JOIN FACTURE f ON r.idResa = f.idResa
WHERE YEAR(r.dateResa) = 2024
AND f.dateFacture IS NOT NULL
AND f.dateReglement IS NULL
ORDER BY r.dateResa;

```

Cette requête : - Sélectionne l'identifiant et la date de réservation, ainsi que le nom et téléphone du client - Joint les tables RESERVATION, CLIENT et FACTURE - Filtre les réservations de 2024 - Sélectionne uniquement les réservations facturées (date de facture non nulle) - Mais non encore réglées (date de règlement nulle) - Ordonne les résultats par date de réservation pour obtenir la liste chronologique

Ces requêtes permettent d'extraire efficacement les informations demandées à partir du schéma relationnel fourni pour le projet de réservation en ligne de camions.

Résumé des fonctions d'agrégation SQL

Les fonctions d'agrégation SQL vous permettent d'effectuer des calculs sur un ensemble de valeurs pour renvoyer une valeur unique. Voici les principales fonctions d'agrégation :

1. Fonctions d'agrégation de base

- **COUNT()** : Compte le nombre de lignes ou de valeurs non NULL

```
SELECT COUNT(*) FROM employees;  
SELECT COUNT(department_id) FROM employees;
```

- **SUM()** : Calcule la somme des valeurs

```
SELECT SUM(salary) FROM employees;
```

- **AVG()** : Calcule la moyenne des valeurs

```
SELECT AVG(salary) FROM employees;
```

- **MIN()** : Retourne la valeur minimale

```
SELECT MIN(salary) FROM employees;
```

- **MAX()** : Retourne la valeur maximale

```
SELECT MAX(salary) FROM employees;
```


2. Clauses associées

- **GROUP BY** : Groupe les résultats par une ou plusieurs colonnes

```
SELECT department_id, AVG(salary)
FROM employees
GROUP BY department_id;
```

- **HAVING** : Filtre les résultats d'agrégation (comme WHERE, mais pour les groupes)

```
SELECT department_id, AVG(salary)
FROM employees
GROUP BY department_id
HAVING AVG(salary) > 50000;
```

3. Fonctions d'agrégation avancées

- **STRING_AGG()** / **GROUP_CONCAT()** : Concatène des chaînes de caractères
- **ARRAY_AGG()** : Crée un tableau à partir des valeurs
- **JSONB_AGG()** / **JSON_ARRAYAGG()** : Agrège les valeurs en JSON

4. Utilisation avec DISTINCT

```
SELECT COUNT(DISTINCT department_id) FROM employees;
```

Ces fonctions sont essentielles pour l'analyse de données et les rapports en SQL.

Fonctions usuelles en SQL

5. Fonctions de chaîne de caractères

- **CONCAT()** : Concatène des chaînes

```
SELECT CONCAT('Bonjour ', nom) FROM utilisateurs;
```

- **SUBSTRING()** : Extrait une partie d'une chaîne

```
SELECT SUBSTRING(nom, 1, 3) FROM utilisateurs;
```

- **UPPER()** / **LOWER()** : Convertit en majuscules/minuscules

```
SELECT UPPER(nom), LOWER(email) FROM utilisateurs;
```

- **LENGTH()** / **LEN()** : Retourne la longueur d'une chaîne

```
SELECT nom, LENGTH(nom) FROM utilisateurs;
```

- **TRIM()**, **LTRIM()**, **RTRIM()** : Supprime les espaces

```
SELECT TRIM(' texte ') FROM dual;
```

6. Fonctions numériques

- **ROUND()** : Arrondit un nombre

```
SELECT ROUND(prix, 2) FROM produits;
```

- **FLOOR()** / **CEILING()** : Arrondit vers le bas/haut

```
SELECT FLOOR(12.7), CEILING(12.2) FROM dual;
```

- **ABS()** : Valeur absolue

```
SELECT ABS(-15) FROM dual;
```

7. Fonctions de date

- **CURRENT_DATE / GETDATE()** : Date actuelle

```
SELECT CURRENT_DATE FROM dual;
```

- **EXTRACT() / DATEPART()** : Extrait une partie d'une date

```
SELECT EXTRACT(YEAR FROM date_commande) FROM commandes;
```

- **DATE_ADD() / DATEADD()** : Ajoute un intervalle à une date

```
SELECT DATE_ADD(date_commande, INTERVAL 7 DAY) FROM commandes;
```

- **DATEDIFF()** : Différence entre deux dates

```
SELECT DATEDIFF(day, date_commande, date_livraison) FROM commandes;
```

8. Fonctions conditionnelles

- **CASE** : Expression conditionnelle

```
SELECT nom,  
CASE  
WHEN age < 18 THEN 'Mineur'  
ELSE 'Majeur'  
END AS statut  
FROM utilisateurs;
```

- **COALESCE()** : Retourne la première valeur non NULL

```
SELECT COALESCE(téléphone, email, 'Non renseigné') FROM utilisateurs;
```

- **NULLIF()** : Retourne NULL si deux expressions sont égales

```
SELECT NULLIF(ville, pays) FROM adresses;
```


9. Fonctions de conversion

- **CAST() / CONVERT()** : Convertit un type de données

```
SELECT CAST(prix AS INT) FROM produits;
```

Ces fonctions sont disponibles dans la plupart des systèmes de gestion de base de données, bien que la syntaxe puisse varier légèrement selon le SGBD utilisé (MySQL, PostgreSQL, SQL Server, Oracle, etc.).

Modèles de transformation numérique avec exemples détaillés

10. 1. Modèle de maturité digitale (MIT Sloan & Capgemini)

Exemple détaillé: Une chaîne hôtelière internationale

- **Débutants:** Hôtel local avec site web basique, réservations par téléphone et email, aucune stratégie digitale cohérente.
- **Conservateurs:** Chaîne Marriott avant 2015 - bonne gouvernance IT, processus définis, mais technologies clients conservatrices comparées aux concurrents.
- **Fashionistas:** Chaîne ayant investi dans applications mobiles, chatbots, miroirs intelligents dans les chambres, mais sans intégration cohérente ni vision unifiée.
- **Digirati:** Marriott après transformation - application mobile connectée au CRM, check-in digital, clés numériques sur smartphone, chambres connectées et personnalisation basée sur l'IA, gouvernance claire et leadership engagé.

11. 2. Les 9 éléments de Gartner

Exemple détaillé: Starbucks

- **Pilier Expérience Client:**

- Expérience omnicanale: Application Starbucks permettant commandes mobiles, paiements et programme de fidélité
- Design thinking: Conception des points de vente comme "troisième lieu" entre maison et travail
- Nouvelle génération de clients: Innovations adaptées aux millennials (commandes personnalisées, paiement sans contact)

- **Pilier Excellence Opérationnelle:**

- Transformation des processus: Réorganisation des comptoirs pour accueillir les commandes mobiles
- Intégration des données: Analyse prédictive pour anticiper les ventes et gérer les stocks
- Automatisation: Machines et processus standardisés pour garantir qualité et rapidité

- **Pilier Nouveaux Business Models:**

- Monétisation des données: Utilisation des données clients pour créer des offres personnalisées
- Expansion digitale: Développement de capsules pour machines à café domestiques
- Écosystème de partenaires: Collaboration avec Spotify, Microsoft, Uber pour enrichir l'expérience

12. 3. Six dimensions de Rogers

Exemple détaillé: Netflix

- **Clients:** Passage de la location de DVD par courrier au streaming personnalisé grâce à l'analyse comportementale
- **Concurrence:** Repositionnement face à des concurrents traditionnels (Blockbuster) puis nouveaux (Amazon, Disney+)
- **Données:** Utilisation des données de visionnage pour créer des contenus originaux comme "House of Cards"
- **Innovation:** Expérimentation continue (interface, algorithmes de recommandation, formats interactifs comme "Bandersnatch")
- **Valeur:** Évolution de la proposition de valeur: de "plus de choix que les vidéoclubs" à "contenu premium personnalisé accessible partout"
- **Organisation:** Transformation de la culture d'entreprise vers l'autonomie et la responsabilisation des équipes

13. 4. Modèle RMMM (adapté)

Exemple détaillé: Décathlon

- **Niveau Basique:** Magasins physiques, site e-commerce simple, peu d'intégration
- **Niveau Intermédiaire:** Click & Collect, stock unifié online/offline, application mobile basique
- **Niveau Avancé:** Personnalisation des recommandations, application avec scan de produits, géolocalisation en magasin
- **Niveau Optimisé:** Expérience omnicanale fluide, bornes interactives, essayage virtuel, inventaire en temps réel
- **Niveau Innovant:** Magasins laboratoires automatisés, RA/RV pour tester les produits, co-création avec les clients via plateformes communautaires

14. 5. Framework d'Altimeter

Exemple détaillé: LEGO

- **Business as Usual:** Vente traditionnelle de boîtes de construction en magasin
- **Test & Apprentissage:** Premières expériences digitales isolées (site web amélioré, quelques applications)
- **Systématisation:** Intégration cohérente entre canaux physiques et digitaux
- **Adaptation stratégique:** Lancement de LEGO Ideas (plateforme de co-crédation), LEGO Life (réseau social sécurisé pour enfants)
- **Convergence organisationnelle:** Réorganisation interne, création d'équipes cross-fonctionnelles
- **Innovation & Adaptation:** Écosystème complet: jeux vidéo, films, applications AR, kits de programmation robotique Mindstorms

15. 6. Modèle 7S de McKinsey

Exemple détaillé: Microsoft sous Satya Nadella

- **Stratégie:** Pivot du "Windows partout" vers "Mobile first, cloud first"
- **Structure:** Réorganisation abandonnant les silos produits pour adopter divisions fonctionnelles
- **Systèmes:** Adoption d'outils agiles, DevOps, intégration continue
- **Style:** Leadership ouvert à la collaboration, fin de la culture de compétition interne
- **Staff:** Recrutement de talents dans l'IA, cloud, mobile
- **Skills:** Formation massive aux nouvelles compétences, valorisation de la curiosité
- **Shared Values:** Changement culturel de "know-it-all" à "learn-it-all", valorisation de l'empathie et de la diversité

16. 7. Matrice de Wade

Exemple détaillé: Banque BNP Paribas

- **Changements tech limités + changements orga limités:** Optimisations des guichets automatiques existants
- **Changements tech importants + changements orga limités:** Lancement application mobile sans réorganisation
- **Changements tech limités + changements orga importants:** Réorganisation des équipes de front-office pour approche client centrée
- **Changements tech importants + changements orga importants:** Lancement de "Hello bank!" (banque 100% digitale) avec nouvelle organisation dédiée, processus et culture différents

17. 8. CMMI adapté au digital

Exemple détaillé: Airbus dans sa gestion de la chaîne d'approvisionnement

- **Niveau 1 (Initial):** Processus ad hoc, communications par email avec fournisseurs
- **Niveau 2 (Géré):** Processus standardisés, premier système d'échange informatisé
- **Niveau 3 (Défini):** Plateforme collaborative avec fournisseurs clés, processus optimisés
- **Niveau 4 (Géré quantitativement):** Suivi en temps réel des pièces, analyse prédictive des délais, KPIs précis
- **Niveau 5 (Optimisé):** Jumeaux numériques de la chaîne logistique, simulations avancées, amélioration continue basée sur l'IA

Chacun de ces modèles offre une perspective unique sur la façon dont les organisations peuvent évoluer et se transformer dans l'ère numérique.

Classification des initiatives numériques de TransCo selon le modèle de Venkatraman (1994) Le modèle de N. Venkatraman "IT-Enabled Business Transformation" (1994) identifie cinq niveaux de transformation numérique, avec des bénéfices potentiels et des degrés de changement organisationnel croissants: Niveau 1: Exploitation localisée

Initiatives TransCo:

Automatisation de la saisie des bons de livraison Numérisation des documents administratifs
Déploiement d'outils bureautiques modernes Mise à jour des postes de travail informatiques

Analyse: TransCo utilise la technologie pour améliorer l'efficacité opérationnelle à travers des changements mineurs et isolés, sans modifier fondamentalement ses processus. Niveau 2: Intégration interne

Initiatives TransCo:

Mise en place d'un ERP pour centraliser les informations Création d'un système d'information RH intégré
Déploiement d'un intranet collaboratif Interconnexion des systèmes comptables et logistiques

Analyse: TransCo intègre ses systèmes d'information pour assurer une meilleure cohérence interne et une circulation plus fluide de l'information. Niveau 3: Refonte des processus (BPR)

Initiatives TransCo:

Réingénierie du processus de gestion des commandes Mise en œuvre d'un système de suivi en temps réel des livraisons
Transformation de la chaîne d'approvisionnement avec identification RFID Refonte du processus de maintenance prédictive des véhicules

Analyse: TransCo redessine ses processus métiers en s'appuyant sur les capacités numériques, allant au-delà de la simple automatisation. Niveau 4: Reconfiguration du réseau d'affaires

Initiatives TransCo:

Plateforme B2B pour connecter clients et partenaires logistiques
Système d'API ouvertes pour les prestataires de transport
Portail fournisseurs pour la gestion collaborative des approvisionnements
Marketplace de services logistiques à la demande

Analyse: TransCo étend sa transformation au-delà de ses frontières pour inclure partenaires, fournisseurs et clients dans un écosystème interconnecté. Niveau 5: Redéfinition du périmètre d'activité

Initiatives TransCo:

Lancement d'une offre de "logistique as a service" basée sur l'IA
Création d'une division de conseil en optimisation de chaîne logistique
Développement de solutions de transport autonomes
Service de prévision de trafic et d'optimisation de routes basé sur l'analyse prédictive

Analyse: TransCo utilise le numérique pour redéfinir sa mission et son modèle d'affaires, transcendant son métier traditionnel de transport. Conclusion TransCo déploie des initiatives à tous les niveaux du modèle de Venkatraman, des exploitations localisées (niveau 1) jusqu'à la redéfinition de son périmètre d'activité (niveau 5). Cette approche équilibrée lui permet de:

Récolter des bénéfices à court terme (niveaux 1-2)
Transformer profondément ses opérations (niveau 3)
Repenser ses relations avec son écosystème (niveau 4)
Se réinventer stratégiquement pour l'avenir (niveau 5)

Cette stratégie multi-niveaux suggère une maturité numérique avancée et une vision transformationnelle claire, combinant améliorations évolutives et initiatives révolutionnaires.

Oui, il existe bien des masques de sous-réseau en IPv6, mais ils fonctionnent différemment de ceux d'IPv4 et sont généralement représentés d'une manière plus simple.

IPv6 : Vue d'ensemble

IPv6 (Internet Protocol version 6) est le protocole de couche réseau développé pour succéder à IPv4, principalement en réponse à l'épuisement des adresses IPv4.

18. Caractéristiques principales

18.1. Format d'adresse

- **Longueur** : 128 bits (contre 32 bits pour IPv4)
- **Notation** : Hexadécimale avec 8 groupes de 4 chiffres hexadécimaux séparés par des ":"
Exemple : 2001:0db8:85a3:0000:0000:8a2e:0370:7334
- **Simplification** : Les zéros consécutifs peuvent être omis Exemple simplifié :
2001:db8:85a3::8a2e:370:7334

18.2. Espace d'adressage

- Environ $3,4 \times 10^{38}$ adresses (340 undécillions)
- Théoriquement suffisant pour attribuer plusieurs milliards d'adresses à chaque être humain

18.3. Types d'adresses

- **Unicast** : Identifie une seule interface
- **Multicast** : Identifie un groupe d'interfaces (commence par FF)
- **Anycast** : Attribué à plusieurs interfaces, le paquet est livré à l'interface la plus proche
- **Plus de broadcast** : Remplacé par des adresses multicast spécifiques

19. Avantages techniques

- **Configuration automatique** : Auto-configuration sans état (SLAAC)
- **Simplification des en-têtes** : Structure plus efficace
- **Support natif de la sécurité** : IPsec intégré
- **Meilleur support de la QoS** : Champ Flow Label pour identifier les flux
- **Plus besoin de NAT** : Suffisamment d'adresses pour tous les appareils
- **Routage plus efficace** : Hiérarchie d'adressage améliorée

20. Défis de déploiement

- Adoption lente malgré l'épuisement des adresses IPv4
- Mécanismes de transition :
 - Double pile (IPv4/IPv6)
 - Tunnels (6to4, Teredo)
 - Traduction (NAT64, DNS64)
- Problèmes de compatibilité avec certains équipements et applications anciennes

21. Types d'adresses courants

- **Global Unicast** : Équivalent aux adresses publiques IPv4, commencent généralement par 2000::/3
- **Link-Local** : Adresses pour communication sur le lien local uniquement (fe80::/10)
- **Unique Local** : Équivalent aux adresses privées en IPv4 (fc00::/7)
- **Loopback** : ::1/128 (équivalent à 127.0.0.1 en IPv4)

La transition vers IPv6 continue progressivement à l'échelle mondiale, avec une adoption variable selon les régions et les fournisseurs de services Internet.

22. Masques de sous-réseau en IPv6

En IPv6, les masques de sous-réseau sont presque exclusivement exprimés en utilisant la notation CIDR (Classless Inter-Domain Routing), qui indique simplement le nombre de bits utilisés pour la partie réseau de l'adresse.

22.1. Notation

- En IPv6, on utilise la notation "/xx" où xx est un nombre entre 0 et 128 indiquant combien de bits (en partant de la gauche) constituent le préfixe réseau
- Exemple : `2001:db8::/64` signifie que les 64 premiers bits identifient le réseau

22.2. Différences avec IPv4

- En IPv4, les masques sont souvent écrits en notation décimale pointée (ex: 255.255.255.0)
- En IPv6, cette notation n'est presque jamais utilisée en raison de sa complexité (il faudrait 16 octets)
- Il n'existe pas de classes d'adresses (comme A, B, C) en IPv6

22.3. Préfixes courants en IPv6

- `/64` : Préfixe standard pour les réseaux finaux (les 64 premiers bits identifient le réseau, les 64 derniers identifient l'interface)
- `/48` : Typiquement attribué aux sites (entreprises, campus)
- `/32` ou `/29` : Souvent attribués aux FAI (Fournisseurs d'Accès Internet)
- `/127` : Utilisé pour les liens point à point (équivalent au `/30` ou `/31` en IPv4)
- `/128` : Identifie une seule adresse (équivalent au `/32` en IPv4)

22.4. Exemple pratique

Pour un réseau d'entreprise avec l'adresse `2001:db8:acad::/48` : - Le préfixe réseau est `2001:db8:acad`
- L'entreprise peut créer jusqu'à 65 536 sous-réseaux (16 bits disponibles) - En utilisant généralement `/64` pour chaque sous-réseau - Par exemple : `2001:db8:acad:1::/64`, `2001:db8:acad:2::/64`, etc.

La simplicité de ce système est l'un des avantages de l'IPv6, car il facilite l'allocation hiérarchique des adresses et la summarisation des routes.

L'architecture Zero Trust est un modèle de sécurité informatique qui représente un changement de paradigme fondamental dans la manière de concevoir la sécurité des systèmes d'information. Voici ses principales caractéristiques et implications :

23. Principes fondamentaux de l'architecture Zero Trust

- **"Ne jamais faire confiance, toujours vérifier"** : Contrairement au modèle traditionnel de sécurité périmétrique (modèle du "château fort"), Zero Trust part du principe qu'aucune entité, interne ou externe au réseau, ne doit être considérée comme fiable par défaut.
- **Vérification continue** : Chaque accès aux ressources nécessite une authentification et une autorisation, quel que soit l'emplacement de l'utilisateur (à l'intérieur ou à l'extérieur du réseau).
- **Principe du moindre privilège** : Les utilisateurs n'ont accès qu'aux ressources strictement nécessaires à l'accomplissement de leurs tâches.
- **Microsegmentation** : Division du réseau en zones isolées pour limiter les mouvements latéraux en cas de compromission.

24. Implications pour la sécurité des SI

1. **Transformation architecturale** : Nécessite une refonte de l'architecture réseau traditionnelle vers un modèle centré sur l'identité et les données plutôt que sur le périmètre.
2. **Authentification multifactorielle (MFA)** : Devient une composante essentielle et systématique de tous les accès aux ressources.
3. **Contrôle d'accès basé sur le contexte** : Les décisions d'autorisation prennent en compte multiples facteurs (identité, appareil, comportement, localisation, etc.).
4. **Surveillance continue** : Nécessite la mise en place de systèmes sophistiqués de détection et de réponse aux incidents (EDR, XDR) avec analyse comportementale.
5. **Gestion des identités centralisée** : Devient la pierre angulaire de la sécurité, souvent avec des solutions IAM (Identity and Access Management) avancées.
6. **Chiffrement omniprésent** : Les données sont chiffrées en transit et au repos, sans exception.
7. **Complexité accrue de gestion** : Requiert des outils d'automatisation et d'orchestration pour gérer efficacement les politiques de sécurité.
8. **Impact sur les performances** : Peut entraîner des défis en termes de latence et de performances des applications.
9. **Gouvernance et conformité** : Facilite la démonstration de la conformité aux réglementations (RGPD, etc.) grâce à une traçabilité accrue.

L'architecture Zero Trust est particulièrement pertinente dans le contexte actuel du travail à distance, du cloud computing et des menaces informatiques évoluées. C'est un sujet avancé qui démontre une compréhension des enjeux contemporains de cybersécurité dans le cadre d'un concours comme l'agrégation économie-gestion option D.