Fiche de revision Generale

C'est avec plaisir que je prépare une fiche de révision exhaustive, organisée par module ou catégorie, incluant des exemples simples pour faciliter la compréhension, tout en conservant tous les détails pertinents issus de vos sources.

FICHE DE RÉVISION DÉTAILLÉE : CONCEPTS CLÉS DU CLOUD ET DE L'IT

(Tous les éléments des sources sont inclus avec des exemples simples.)

I. Les Fondamentaux de la Sécurité (DICP & IAM)

1. Le Modèle DICP (Piliers de la Sécurité de l'Information)

Ce modèle représente les piliers essentiels de la sécurité.

Pilier	Définition	Exemple Simple pour Comprendre
Disponibilité	Les systèmes et les données sont accessibles aux utilisateurs autorisés lorsque nécessaire.	Assurer que le site web de la compagnie est toujours en ligne.
Intégrité	Les données ne doivent pas être modifiées ou altérées de manière non autorisée.	S'assurer qu'un virement de 100 € dans une base de données reste 100 € pendant la transaction.
Confidentialité	Les informations restent accessibles uniquement aux personnes autorisées (protection contre les fuites).	Seul l'administrateur peut voir le mot de passe d'un utilisateur.
Preuve et Traçabilité	Nécessité de conserver une trace (journaux, logs) des actions pour prouver qui a fait quoi et quand.	Les logs AWS montrent que l'utilisateur "Alice" a arrêté le serveur EC2 à 14h00.

2. Gestion des Identités et des Accès (IAM)

IAM est le service utilisé dans le cloud pour gérer **qui** peut faire **quoi** sur les ressources.

- **Identité :** Utilisateurs (individuels ou applications) qui possèdent un compte (ex: un administrateur, un développeur).
- **Gestion des accès :** Définition des permissions précises (accès à un serveur, à une base de données, etc.).

- **Stratégies IAM (Policies) :** Documents définissant les permissions **autorisées** ou **refusées** sur des ressources spécifiques. Par défaut, tout est refusé.
- **Rôles IAM :** Identités avec des autorisations temporaires, souvent utilisées par des applications ou services AWS pour accéder à d'autres ressources.
    - *Exemple :* Un **rôle** "Lecteur S3" est attribué à une application pour qu'elle puisse lire, mais pas supprimer, des fichiers dans un compartiment S3.
- **Principe de Sécurité :** Appliquer le **principe du moindre privilège** (n'accorder que ce qui est absolument nécessaire).

3. Contrôle d'Accès et Authentification (ACL, MFA)

- **ACL (Access Control List / Liste de Contrôle d'Accès) :** Liste qui définit qui peut accéder à une ressource et ce qu'ils peuvent faire (lire, écrire, exécuter, refuser, etc.).
    - *Exemple :* Une ACL dans un pare-feu autorise uniquement le trafic Web (port 80) à entrer sur le serveur, bloquant tout le reste.
- **MFA (Multi-Factor Authentication) :** Ajoute une couche de sécurité en demandant plusieurs preuves d'identité (mot de passe + code SMS/token).
    - *Recommandation :* L'activation de la MFA est **hautement recommandée** pour l'utilisateur racine du compte AWS.

II. Le Cloud Computing et les Modèles

1. Définition et Avantages du Cloud

Le Cloud Computing est la mise à disposition de ressources informatiques **à la demande** via Internet, avec une tarification basée sur l'utilisation (**paiement à l'usage**).

- **Avantages Financiers (Coûts) :**
    - Remplacement des dépenses d'investissement (**CapEx**) par des dépenses d'exploitation variables (**OpEx**).
        - *Exemple CapEx vs OpEx :* **CapEx** = Achat initial d'un serveur physique. **OpEx** = Payer une instance EC2 uniquement pour les heures où elle est utilisée.
    - Bénéficier d'économies d'échelle massives.
    - Élimination de la spéculation de capacité (ne plus avoir à deviner la capacité nécessaire).
- **Avantages Opérationnels :** Augmentation de l'agilité et de la vitesse, déploiement mondial en quelques minutes.

2. Modèles de Service (SaaS, PaaS, IaaS)

Modèle	Définition	Fournisseur gère	Client gère / se concentre sur	Exemple Simple
IaaS (Infrastructure as a Service)	Location d'infrastructures brutes (serveurs virtuels).	Serveurs, stockage, réseaux, machines virtuelles.	**OS**, applications, données.	Amazon EC2 (Location d'un serveur virtuel).
PaaS (Platform as a Service)	Plateforme de développement fournie.	Infrastructure + OS + middleware + bases de données.	Le code et les applications déployées.	Google App Engine ou Heroku (Déploiement simple d'un code Python).
SaaS (Software as a Service)	Logiciel prêt à l'emploi, accessible en ligne.	Tout (infrastructure, plateforme, application, maintenance).	Utilisation de l'application via navigateur/appli.	Gmail, Microsoft 365, Dropbox.

3. Modèles de Déploiement

- **Cloud Public :** Services fournis par un fournisseur tiers sur Internet (ex: AWS, Azure, GCP), accessibles au grand public.
- **Cloud Hybride :** Combinaison d'un environnement sur site (privé) et d'un cloud public, permettant l'échange de données.
- **Cloud Privé / Sur Site (On-Premises) :** Ressources informatiques hébergées dans le centre de données interne d'une organisation.

4. Modèle de Responsabilité Partagée AWS

Ce modèle répartit clairement les obligations de sécurité entre le fournisseur et le client.

Responsabilité AWS : Sécurité DU Cloud	Responsabilité Client : Sécurité DANS le Cloud
Protection de l'infrastructure globale (Régions, AZs, matériel, logiciels).	Sécurité des données client (chiffrement, gestion des accès).

Services entièrement gérés (ex: DynamoDB).	Système d'exploitation invité (mises à jour, correctifs). Configuration réseau (Groupes de sécurité, listes ACL).

- *Observation :* La responsabilité du client est maximale pour l'**IaaS** (où il gère l'OS) et minimale pour le **SaaS** (où AWS gère la majorité des composants).

III. Infrastructure Mondiale AWS

L'infrastructure est conçue pour la haute disponibilité et la tolérance aux pannes.

- **Régions :** Zone géographique physique isolée regroupant des centres de données. Elles assurent l'isolement géographique et répondent aux exigences de souveraineté des données.
- **Zones de Disponibilité (AZ) :** Emplacements distincts et physiquement séparés au sein d'une Région. L'isolement des défaillances garantit la tolérance aux pannes et la haute disponibilité.
    - *Exemple :* Déployer un serveur Web dans **deux AZs différentes** garantit que si une catastrophe frappe une zone, l'application reste en ligne dans l'autre.
- **Points de Présence (PoPs) / Localités de Bordure (Edge Locations) :** Utilisés par les services de périphérie pour réduire la latence.
    - *Exemple :* Un utilisateur en Asie accède au contenu du site Web stocké en cache dans un PoP local, réduisant ainsi le temps de chargement.

IV. Calcul (Compute) et Évolutivité

1. Amazon EC2 (Elastic Compute Cloud)

Service web qui fournit une capacité de calcul redimensionnable (machines virtuelles).

- **AMI (Amazon Machine Image) :** Modèle pré-configuré (OS, serveur d'application) utilisé pour créer une instance EC2. C'est un "snapshot" réutilisable.
    - *Exemple :* Choisir une AMI avec Linux et le logiciel de base de données déjà installé pour lancer immédiatement un nouveau serveur.
- **Types d'instances :** Différentes combinaisons de CPU, mémoire, stockage (ex: Usage général, Optimisé pour le calcul, Optimisé pour la mémoire).
- **Tarification EC2 :**

- **À la demande (On-Demand) :** Facturation à la seconde, sans engagement.
- **Instances Réservées (RI) :** Réductions significatives pour un engagement d'utilisation sur 1 ou 3 ans.
- **Instances Spot :** Capacité inutilisée à des remises importantes, idéales pour les charges de travail tolérantes aux pannes ou qui peuvent être interrompues.

2. Évolutivité et Auto Scaling

- **Mise à l'échelle horizontale (Scaling Out / In) :** Ajout ou suppression d'instances (meilleur pour l'élasticité du cloud).
- **Mise à l'échelle verticale (Scaling Up / Down) :** Augmentation ou diminution de la puissance des instances existantes.
- **Amazon EC2 Auto Scaling (ASG) :** Garantit qu'un nombre suffisant d'instances est disponible et gère l'ajout/retrait d'instances.
  - **Politique de Suivi de Cible :** Ajuste la capacité pour maintenir une métrique spécifique à un niveau cible (ex: maintenir l'utilisation moyenne du CPU à 60%).
  - **Politique Planifiée :** Augmente ou diminue la capacité à des heures spécifiques et prévisibles (ex: monter en charge tous les matins à 8h00).

3. Services Serverless et Conteneurs

- **AWS Lambda :** Service de calcul **sans serveur** qui exécute du code en réponse à des événements.
  - *Exemple :* Exécuter une fonction pour redimensionner automatiquement une image chaque fois qu'un utilisateur la télécharge dans S3.
- **AWS Fargate :** Moteur de calcul **sans serveur** pour les conteneurs (utilisé avec ECS ou EKS), éliminant la nécessité de gérer les serveurs EC2 sous-jacents.

V. Stockage et Bases de Données

1. Services de Stockage (Types)

Type de Stockage	Service AWS Principal	Caractéristique Clé	Exemple Simple

Objet	Amazon S3 (Simple Storage Service)	Stockage en *buckets*, durabilité de 99,999999999 % (onze "neufs").	Stocker des milliards de fichiers, des sauvegardes, ou des photos.
Bloc	Amazon EBS (Elastic Block Store)	Similaire à un disque dur, attaché à **une seule** instance EC2 dans la même AZ.	Le disque principal où le système d'exploitation de votre serveur EC2 est installé.
Fichier	Amazon EFS (Elastic File System)	Système de fichiers évolutif, **partageable simultanément** par plusieurs instances EC2.	Permettre à 5 serveurs Web d'accéder au même répertoire de fichiers de configuration.

- **S3 Classes de Stockage :** Permettent d'optimiser les coûts selon la fréquence d'accès (Standard, Intelligent-Tiering, Standard-IA, Glacier pour l'archivage).
    - **Politiques de Cycle de Vie :** Transitionnent automatiquement les objets entre ces classes pour réduire les coûts.
- **EBS Instantanés (Snapshots) :** Sauvegardes ponctuelles des volumes EBS, stockées dans S3.
- **S3 Glacier :** Optimisé pour l'archivage très rentable et très durable, tolérant des temps de récupération de plusieurs heures.

2. Bases de Données

Type de Base de Données	Propriétés Clés	Service AWS Principal	Exemple Simple
Relationnelles (RDBMS)	Tables, SQL, Adhère aux propriétés **ACID** (Atomicité, Cohérence, Isolation, Durabilité).	**Amazon RDS** (service géré) ou **Amazon Aurora** (moteur haute performance AWS).	Gérer des transactions bancaires qui nécessitent une cohérence stricte (ACID).
Non Relationnelles (NoSQL)	Schéma flexible, Évolutivité horizontale, Suit souvent le modèle	**Amazon DynamoDB** (Clé-Valeur/Document).	Gérer des paniers d'achat ou des sessions utilisateurs à haute vitesse et grand volume.

	BASE (Finalement Cohérent).		
Entrepôt de Données (OLAP)	Optimisé pour l'analyse de données à grande échelle, utilise un stockage orienté colonne.	**Amazon Redshift**.	Analyser des millions de ventes historiques pour créer des rapports d'entreprise.

- **Haute Disponibilité RDS :** Assurée par le déploiement **Multi-AZ** (réplication synchrone).
- **Performance RDS :** Améliorée par les **Réplicas en lecture** (réplication asynchrone).
- **Aurora :** Moteur RDBMS compatible MySQL/PostgreSQL, qui réplique les données sur six copies dans trois AZs.

VI. Mise en Réseau et Diffusion de Contenu

1. Amazon VPC (Virtual Private Cloud)

Un réseau virtuel isolé, défini par l'utilisateur, dans le cloud AWS.

- **Sous-réseaux :** Divisions du VPC, soit **Publics** (avec accès Internet via IGW) soit **Privés** (sans accès Internet direct).
- **Passerelle Internet (IGW) :** Permet la communication entre le VPC et Internet.
- **Passerelle NAT (NAT Gateway) :** Permet aux instances des sous-réseaux **privés** d'initier des connexions **sortantes** vers Internet (ex: pour télécharger des mises à jour, sans être accessibles de l'extérieur).
- **Appairage de VPC (VPC Peering) :** Connexion réseau entre deux VPC pour la communication privée.
- **AWS Direct Connect :** Établit une connexion réseau physique et dédiée entre votre centre de données sur site et AWS.

2. Sécurité de VPC (Filtrage)

- **Groupes de sécurité (Security Groups) :** Agissent au niveau de l'**instance** (ENI/EC2). Ils sont **Stateful** (si le trafic sort, la réponse est autorisée à entrer).
    - *Exemple :* Autoriser uniquement votre adresse IP de bureau à se connecter à votre instance EC2 via SSH.

- **ACL réseau (NACL) :** Agissent au niveau du **sous-réseau**. Ils sont **Stateless** (le trafic d'entrée et de sortie est traité indépendamment, nécessitant deux règles explicites).

3. Services de Périphérie (Edge Services)

- **Amazon Route 53 :** Service Web **DNS** (Domain Name System) hautement disponible et évolutif. Utilisé pour enregistrer des noms de domaine et router le trafic.
- **Amazon CloudFront :** Réseau de diffusion de contenu (CDN) rapide qui distribue le contenu (statique et dynamique) aux utilisateurs en utilisant les **Points de Présence (PoPs)** pour réduire la latence.

VII. Surveillance, Architecture et Coûts

1. Surveillance et Équilibrage de Charge

- **AWS CloudWatch :** Service fondamental de monitoring et d'observabilité. Il collecte, visualise et analyse métriques, logs et événements en temps réel.
    - **Métriques :** Variables numériques représentant des données (ex: utilisation du CPU, trafic réseau).
    - **Alarmes :** Déclenchent des actions lorsqu'une métrique atteint un seuil (ex: l'alarme déclenche la mise à l'échelle automatique de l'ASG).
- **Elastic Load Balancing (ELB) :** Répartit le trafic entrant entre plusieurs cibles (instances EC2) dans différentes AZs pour augmenter la tolérance aux pannes et l'évolutivité.
    - **Types d'ELB :** Application Load Balancer (**ALB**, Couche 7/HTTP) et Network Load Balancer (**NLB**, Couche 4/faible latence).

2. Cadre AWS Well-Architected (Cinq Piliers)

Ce cadre aide à concevoir des architectures sécurisées, performantes, résilientes et efficaces.

1. **Excellence Opérationnelle :** Se concentre sur l'exécution, la surveillance, l'automatisation (Infrastructure as Code) et l'apprentissage des défaillances.
2. **Sécurité :** Protéger les systèmes, appliquer le principe de **défense en profondeur** (sécurité à toutes les couches), et protéger les données en transit et au repos.
3. **Fiabilité :** Assurer qu'un système fonctionne comme prévu, même après une défaillance.

    - **RTO (Recovery Time Objective) :** Temps maximum acceptable pour que le service soit restauré après un sinistre.
    - **RPO (Recovery Point Objective) :** Quantité maximale de données qui peut être perdue lors d'un sinistre.
4. **Efficacité des Performances :** Utiliser efficacement les ressources, adopter les architectures **serverless**, et optimiser en continu.
5. **Optimisation des Coûts :** Adopter un modèle de consommation basée sur l'usage (FinOps), éviter les dépenses inutiles et attribuer les dépenses.

3. Outils de Conformité et d'Audit

- **AWS Trusted Advisor :** Inspecte votre environnement AWS et fournit des recommandations en temps réel dans cinq catégories : Coûts, Performance, Sécurité, Tolérance aux pannes et Limites de service.
- **AWS Config :** Surveille et évalue en permanence les configurations des ressources pour vérifier la conformité aux réglementations.
- **AWS Artifact :** Fournit un accès instantané aux documents de conformité (SOC, ISO, HIPAA BAA).

4. Gestion des Coûts

- **Coût Total de Possession (TCO) :** Utilisé pour comparer les coûts sur site (CapEx) avec les coûts du cloud (OpEx).
- **AWS Organizations :** Gère centralement plusieurs comptes AWS. Permet la **Facturation Consolidée** (un seul paiement, bénéfice des paliers dégressifs et des remises d'instances réservées).
- **AWS Budgets :** Définit des budgets personnalisés et envoie des alertes si les coûts dépassent un seuil.
- **Balises d'allocation des coûts (Cost Allocation Tags) :** Permettent d'attribuer les coûts à des projets ou des départements spécifiques.
    - *Exemple :* Attribuer le "Tag: Projet=Marketing" à tous les serveurs et buckets utilisés par l'équipe Marketing pour connaître leur dépense exacte.

VIII. Concepts Divers

- **SLA (Service Level Agreement) :** Contrat formel définissant le niveau de service attendu (Disponibilité, Performance, Délai de réponse du support) et les pénalités en cas de non-respect.

- **SDK (Software Development Kit) :** Kit de développement logiciel contenant APIs, librairies, documentation et outils pour coder et intégrer directement les services (ex: AWS SDK, Android SDK).
- **PoC (Proof of Concept / Preuve de concept) :** Démonstration limitée qui montre qu'une idée ou une technologie est réalisable avant d'investir davantage.
- **VPC Endpoints :** Permettent aux instances d'accéder aux services AWS (ex: S3) sans passer par l'Internet public.
- **CDM (Customer Data Management) :** Gestion des données clients dans le cloud, impliquant souvent Amazon S3, AWS Glue et DynamoDB.
- **AMI (Alternate Mark Inversion ou Advanced Metering Infrastructure) :** Acronyme à distinguer de l'Amazon Machine Image (AMI) d'EC2.