

الجمهورية الجزائرية الديمقراطية الشعبية  
République Algérienne démocratique et populaire

وزارة التعليم العالي والبحث العلمي  
Ministère de l'enseignement supérieur et de la recherche scientifique

جامعة سعد دحلب البلدية  
Université SAAD DAHLAB de BLIDA

كلية التكنولوجيا  
Faculté de Technologie

قسم الإلكترونيك  
Département d'Électronique



## Mémoire de Master

Filière : **Électronique**

Spécialité : **Systèmes de télécommunications**

Présenté par :

AGUEMOUN Salah

&

MAALEM Med Ettayeb

**THEME:**

---

**Déploiement d'une solution stockage**

**Objet dans le cloud**

---

Proposé par : **AGUEMOUN Med Amine**

&

**AIT SAADI Houcine**

Année Universitaire 2023-2024

## ***REMERCIEMENTS***

---

Nous tenons à remercier toutes les personnes qui ont contribué au succès de notre stage et qui nous ont aidés lors de la rédaction de ce rapport.

Tout d'abord, nous adressons nos remerciements à notre maitre de stage, Mr Mohamed Amine AGUEMOUN pour son accueil, le temps passé ensemble et le partage de son expertise au quotidien.

Grâce aussi à sa confiance nous avons pu nous accomplir totalement dans nos missions avec son aide précieuse dans les moments les plus délicats.

Nous tenons à remercier Dr. Hocine AIT SAADI Chef du département de l'électronique pour nous avoir permis d'accéder à ce stage de fin d'étude.

Nous gratifions également toute l'équipe de télécommunications et système : Madame BOUTALEB, Mr DAHMANI, pour leurs remarques et leurs directives.

## **Dédicaces**

A mes très chers parents

Je dédie ce PFE à mes parents, pour l'amour qu'ils m'ont toujours donné, leurs encouragements et toute l'aide qu'ils m'ont apportée durant mes études

Aucun mot, aucune dédicace ne pourrait exprimer mon respect, ma considération, et mon amour pour les sacrifices qu'ils ont consentis pour mon instruction. « Rabi yetawel fe amarkom »

A mes chers frères

Mes grands frères Mustapha, Bilel, Sofiane et mon frangin adoré Mohamed

Amine

Que Dieu vous apporte le bonheur, vous aide à réaliser tous vos vœux et vous offre un avenir plein de succès

A mon binôme « MAALEM MED ETTAYEB »

A tous mes amis de l'USDB et spécialement

« GHEZLI Rania »

Et tous mes professeurs de la spécialité système de télécommunication

**AGUEMOUN SALAH**

## **Dédicaces**

A ma chère mère et À mon cher père décédé,

Pour l'éducation qu'ils m'ont prodigué avec tous les moyens et au prix  
de tous les sacrifices qu'ils ont consentis à mon égard, pour le sens du  
devoir qu'ils m'ont enseigné depuis mon enfance

A mes chères sœurs

A tous mes amis et collègues que j'ai pu connaître depuis mon enfance

Surtout à mon partenaire Aguemoun Lyasse

A tous les professeurs qui m'ont enseigné tout au long de mon parcours

A tous ceux qui m'aiment, A tous ceux que j'aime

Je vous dédie ce modeste travail.

**MAALEM Med Ettayeb**

## **Abstract:**

In the world of digitalization, it is important to preserve the data, this latter is a container of useful information critical more over crucial, for this Several approaches are already known such as backup, archiving, lifting users' needs is the main objective of this work.

A new approach was made which is the deployment of an object storage solution published on Algerian cloud, a very intuitive and easy to operate with the security standards adding to this a consistent with user needs, it essential to keep data in reliable space, to finish perspectives are presented and improvements can be done.

Keywords: data, cloud, backup, object storage, security

## **Résumé :**

dans le monde de numérisation , il est important de conserver la données , cette dernière et un conteneur de l'information utile critique de plus cruciale et minimiser les pertes pour cela plusieurs approches sont déjà connues tels que le backup et l'archivage et soulever les besoins des utilisateurs , dans cette thèse une nouvelle approche a été faite qui était le déploiement d'une solution de stockage objet publiée dans le web avec une console très intuitive et facile à exploiter avec les normes de sécurité et conforme aux besoins des utilisateurs , il primordiale de conserver les données dans un espace fiable , pour terminer les perspectives sont présentés et des améliorations peuvent prouver notre solution.

Mots clés : donnée, cloud, restauration, stockage objet, sécurité.

## **الملخص :**

في عالم الرقمنة، من المهم الحفاظ على البيانات، فهذه الأخيرة حاوية من المعلومات الهامة المفيدة أكثر أهمية وتقليل الخسائر لهذا عدة أساليب معروفة بالفعل مثل النسخ الاحتياطي والأرشفة ورفع احتياجات المستخدم، في هذه الأطروحة تم اتباع نهج جديد يتمثل في نشر حل تخزين الكائنات المنشور على الويب باستخدام وحدة تحكم سهلة الاستخدام للغاية وسهلة التشغيل مع معايير الأمان ومتوافقة مع احتياجات المستخدم، ومن الضروري الاحتفاظ بالبيانات في مساحة موثوق لإنهاء المنظورات يتم تقديمها والتحسينات يمكن أن تثبت الحل الذي نقدمه

كلمات المفاتيح:البيانات, الأمان, تخزين الكائنات, النسخ الاحتياطي .

## Listes des acronymes et abréviations

TIC : Technologies de l'Information et de la Communication

ANPDP : Autorité Nationale de Protection des Données à caractère Personnel.

POP : Post Office Protocol

VOIP : Voice Over Internet Protocol

FAI : Fournisseur d'accès à Internet

PME : Petites et les Moyennes Entreprises

TPE : Très Petites Entreprises

IP : Internet Protocol

ALPAL 2: Alger-Palma 2

SeMeWe 4: South East Asia–Middle East–Western Europe 4

VM: Virtual Machine

VSAN: Virtual Storage Area Network

VDC: Virtual data center

CPU: Central Processing Unit

RAM: Random Access Memory

VPS: Virtual Private Server

DDoS: Distributed Denial-of-Service

SSL: Secure Sockets Layer

SGBDR : Système de Gestion de Base de Données Relationnelle

IA: Intelligence Artificielle

SQL: Structured Query Language

IoT: Internet of Things

NAS: Network Attached Storage

SMB: Server Message Block

NFS: Network File System

SDK: Software Development Kit

API: Application Programming Interface

GCE: Google Compute Engine

AWS: Amazon Web Services

CRM: Customer Relationship Management

WAN: Wide area network

CDN: Content Delivery Network

SDN: Software Defined Network

KVM: Kernel-based Virtual Machine

MacOS: Macintosh Operating System

ESXi: Elastic Sky X integrated

SaaS: Software as a Service

PaaS: Platform as a Service

IaaS: Infrastructure as a Service

AWS EC2: Amazon Web Services Elastic Compute Cloud

iOS: iPhone Operating System

SSH: Secure Socket Shell

S3: Simple Storage Service

SNSD: Single-Node Single-Drive

RPM: Red Hat Package Manager

DNS: Domain Name System

FQDN: Fully Qualified Domain Name

RDP: Remote Desktop Protocol

CA: Certificate Authority

URL: Uniform Resource Locator

MC: Minio Client

PATH: Program for Appropriate Technology in Health

FTTP : Fiber to the premises (Fibre jusqu'aux locaux - entreprises)

HTTP: Hypertext Transfer Protocol

HTTPS: Hyper Text Transfer Protocol Secure

# Table des matières

---

Introduction générale.....	1
Chapitre I	
I.1 Introduction .....	4
I.2 Historique d'Icosnet .....	4
I.3 Présentation d'icosnet .....	4
I.3.1 Adresse de la société Icosnet .....	5
I.4 Infrastructure d'icosnet .....	5
I.4.1 Datacenter .....	5
I.4.2 Infrastructure réseau .....	6
I.4.3 Infrastructure virtuelle.....	7
I.4.3.1 Site principal.....	7
I.4.3.2 Site Secours .....	7
I.5 Service de la société .....	7
I.6 Conclusion .....	7
Chapitre II	
II.1 Introduction .....	9
II.2 Historique .....	9
II.3 Définition de Cloud computing .....	10
II.4 Comment fonctionne le Cloud computing .....	11
II.4.1 Machine virtuelle .....	12
II.4.2 Hyperviseur .....	13
II.5 Composants de Cloud computing .....	14
II.5.1 Data center (Centres des données) .....	14
II.5.2 Capacités de mise en réseau .....	14
II.5.3 Virtualisation .....	15
II.5.3.1 Définition de la virtualisation .....	15
II.6 Avantages de Cloud computing .....	16
II.7. Services de Cloud Computing .....	17
II.7.1 Infrastructure as a service (IaaS) .....	17
II.7.2 Platform as a service (PaaS) .....	18
II.7.3 Software as a service (SaaS) .....	18
II.8 Modèles de déploiement du Cloud computing .....	19
II.8.1 Cloud public .....	19

II.8.2 Cloud privé .....	20
II.8.3 Cloud hybride .....	20
II.8.4 Multicloud .....	20
II.9 Exemples d'utilisation de cloud .....	21
II.10 Conclusion .....	23
Chapitre III	
III.1 Introduction .....	25
III.2 Types de donnée .....	25
III.2.1 Données structurées .....	25
III.2.2 Données non structurées.....	25
III.2.3 Comparaison entre les données non structurées et données structurées .....	26
III.3 Types de stockages.....	27
III.3.1 Stockage fichier .....	27
III.3.2 Stockage blocs.....	28
III.3.3 Le stockage Objet .....	29
III.3.3.1 l'importance de stockage objet.....	30
III.3.3.2 Les avantages de stockage objet.....	32
III.3.3.3 Cas d'utilisation du stockage d'objets .....	33
III.4 Comparaison entre le stockage objet et les autres types .....	34
III.4.1 Comparaison entre le stockage d'objets et le stockage fichier .....	34
III.4.2 Comparaison entre le stockage d'objets et le stockage par bloc .....	35
III.5 Solutions open source.....	37
III.6 Conclusion .....	38
Chapitre IV	
IV.1 Introduction .....	40
IV.2 Création de la machine virtuelle .....	40
IV.2.1 Accéder à cette machine via le protocole SSH .....	42
IV.3 Configuration préliminaire un serveur linux .....	43
IV.3.1 Caractéristique de la machine .....	43
IV.3.2 Changements de mot de passe .....	44
IV.3.3 Changement de nom de la machine .....	44
IV.3.4 Editer le fichier /etc/hostname avec l'utilitaire vi .....	44
IV.4 Déploiement du serveur MinIO .....	45
IV.4.1 Téléchargement de MinIO server .....	45
IV.4.2 Création du fichier systemd pour le service minio.service dans le système .....	46

IV.4.2.1 Systemd (system daemon) .....	46
IV.4.3 Créer le fichier de variables d'environnement.....	47
IV.4.4 Pointage DNS .....	50
IV.4.4.1 Authentification .....	52
IV.4.5 Installation certificat SSL .....	52
IV.5 Création des buckets .....	54
IV.5.1 Gestion des versions du compartiment.....	54
IV.5.2 Gestion des versions et capacité de stockage.....	55
IV.5.3 Tester une bucket sur le serveur MINIO .....	56
IV.6 MinIO Client (mc).....	58
IV.6.1 Installer MinIO Client (MC) sur Linux.....	59
IV.6.2 Créez un alias pour le service compatible S3.....	61
IV.6.2.1 Sous-commandes de « mc alias ».....	61
IV.6.2.2 Command « mc mb » .....	65
IV.7 Application d'intégration de la solution stockage objet .....	66
IV.8 Conclusion .....	68
Conclusion générale.....	69

# Table des figures

---

## Chapitre I :

I.1 : Icosnet .....	5
I.2 : Zone datacenter .....	6
I.3 : Le réseau d'icosnet.....	6
I.4 : Infrastructure virtuelle.....	7

## Chapitre II :

II.1: Cloud computing .....	11
II.2 Le Cloud computing migre les ressources au sein d'Internet .....	12
II.3 : Hyperviseur .....	13
II.4 : Datacenter .....	14
II.5: Les services de Cloud .....	17
II.6 : Modèles de déploiement du Cloud .....	19
II.7: Google drive .....	21
II.8: iCloud .....	21
II.9: Amazon Cloud drive .....	22
II.10: Ibox .....	22

## Chapitre III

III.1 : Types de données .....	27
III.2: Stockage Fichier .....	28
III.3: Stockage Bloc .....	29
III.4: Stockage Objet .....	30

## Chapitre IV :

IV.1 Login page .....	40
IV.2 Service VPS-linux .....	41
IV.3 Les caractéristiques de la machine virtuel .....	42
IV.4 Protocole SSH .....	43
IV.5 Saisir le mot de passe root .....	43
IV.6 Les Caractéristique de la machine .....	43
IV.7 Changement de mot de passe avec succès .....	44
IV.8 Changement de nom de la machine .....	44
IV.9 Le fichier /etc/hostname .....	44
IV.10 Redémarrer la machine .....	44
IV.11 Mis à jour du système d'exploitation .....	45
IV.12 Ping google.com .....	45
IV.13 Les nouveaux caractéristique de la machine .....	45
IV.14 Téléchargement de minio server .....	46
IV.15 Le fichier minio.rpm est présent dans le système .....	46
IV.16 L'exécution de minio server .....	46
IV.17 Systemd .....	46
IV.18 L'utilisateur et le groupe .....	47
IV.19 créer un dossier /data/minio .....	47
IV.20 Fichier de variables d'environnement .....	47
IV.21 Minio en mode active.....	48
IV.22 Les ports minio sont à l'écoute en interne .....	48
IV.23 Le site est inaccessible .....	48

IV.24 Autorisations au niveau de firewalld .....	49
IV.25 La console minio est accessible via le net .....	49
IV.26 Serveur DNS qui gère le domaine icosnethosting.com .....	50
IV.27 localisations DNS .....	50
IV.28 Pointage DNS .....	51
IV.29 Propagation DNS .....	51
IV.30 Minio consol avec le lien FQDN .....	51
IV.31 Minio consol apres l'authentification .....	52
IV.32 Créer un dossier minio-user .....	52
IV.33 Le propriétaire des fichiers + dossier .....	53
IV.34 Le chemin de la certificat SSL .....	53
IV.35 Le certificat est bien installé .....	53
IV.36 Création des buckets .....	54
IV.37 Les buckets .....	56
IV.38 Téléchargement d'un fichier .....	56
IV.39 Le fichier (ch3) est bien stocké dans la bucket .....	57
IV.40 Le code URL pour le partage de fichier .....	57
IV.41 Le fichier a été bien partager .....	58
IV.42 La commande « cp » .....	58
IV.43 Téléchargements des packages .....	59
IV.44 La modification des permissions et rendre le « mc » exécutable .....	59
IV.45 Les diverses commandes de mc .....	60
IV.46 Déplacement vers le path /usr/local/bin/mc .....	60
IV.47 Le path /usr/local/bin/mc .....	61

IV.48 La version de mc installée .....	62
IV.49 La machine client .....	62
IV.50 Les alias par défaut .....	62
IV.51 Ajouter le serveur via le port 9000 .....	63
IV.52 La création d'une clé d'accès .....	63
IV.53 L'insertion de la clé d'accès .....	64
IV.54 La résolution dns elle est intacte .....	64
IV.55 Le mc alias est bel et bien installé .....	64
IV.56 Le serveur minio est ajouté avec succès .....	64
IV.57 Test de la connexion depuis mc client vers le serveur minio .....	65
IV.58 Créer un bucket .....	65
IV.59 Vérification au niveau de la console minio .....	65
IV.60 Générer une clé secrète .....	66
IV.61 Donner un nom à la repository de stockage objet .....	66
IV.62 Les informations du serveur minio .....	67
IV.63 Les bucket de minio sur le veeam .....	67
IV.64 Le résultat final depuis le serveur veeam .....	68
IV.65 La bucket vbrsuite sur le serveur minio .....	68

**Liste des tableaux :**

Tableau II.1 : Les avantages de cloud computing.....	16
Tableau III.1 : Comparaison entre le stockage objet et le bloc.....	37
Tableau IV.1 Sous-commandes de « mc alias » .....	61

# Introduction générale

---

Au cours de ces dernières années, les technologies de l'information et la communication TIC sont devenues un vecteur de changement de plus en plus important de la croissance des entreprises, de nos jours fournir des solutions d'hébergement des données accessibles et fiables par le biais du réseau mondiale internet présente aujourd'hui un défi dans le marché algérien avec la présence de plusieurs fournisseurs de services accès internet, solution de communication par voie IP mais aussi des solution de télécommunications et hébergement.

Autant d'exemples pour montrer que les TIC et la numérisation sont une partie intégrante de notre quotidien nous pouvons citer les systèmes de paiement en ligne, les services hébergement messagerie professionnel et site web, les applications, sauvegardes des données, la reprise après sinistre (desaster recovery), Par exemple, les entreprises du secteur de la santé utilisent le Cloud pour développer des traitements plus personnalisés pour leurs patients

La mise en place d'une infrastructure Cloud qui répond au besoin en termes d'hébergement est cruciale tout en garantissant l'accessibilité, la sécurité mais surtout la fiabilité en point de vue de stockage de ces données et les diverses applications exploitées afin de fournir un service de succès aux utilisateurs.

En général, le Cloud computing est la mise en disposition de ressources informatiques matérielles et logicielles à la demande selon le service demandé accessible via internet en exploitant des protocoles et méthodes d'accès conformement aux critères de la sécurité Les serveurs situés dans le Cloud sont hébergés au sein de datacenter.

Le Cloud présente aujourd'hui le fondement de plus l'utilisation de l'internet on cite alors : les sites web avec une réservation de nom de domaine, la messagerie électronique professionnelle, les moteurs de recherches, les bases de données, les applications mobiles et métiers, les médias en streaming, et plein d'autre services hébergés dans le Cloud toute en conformité avec les lois des pays qui visent à la protection des données.

Suivant la nouvelle qui est la 18-06 qui est entrée en vigueur le 10 Aout 2023, un organisme national ANPDP (Autorité Nationale de Protection des Données A Caractère Personnel) a été créé afin de veiller lors de traitement des données à caractères personnel au respect de la dignité humaine et de la vie privée.

Notre projet vise à faire la conception et déploiement d'une solution de stockage objet dans un Cloud Algérien qui répond aux divers besoins des entreprises et clients afin d'intégrer et gérer en toute flexibilité en administration le stockage, nous pouvons donner des cas utiles : données backups, bases de données, contenu site web ....

Dans un premier lieu nous allons présenter le fournisseur d'accueil icosnet le premier fournisseur services internet, explication des composants de Cloud, notions sur les divers types de stockages avec un benchmark, en finalisant par la partie pratique.

# Chapitre I

## Présentation de l'entreprise d'accueil

## **I.1 Introduction :**

Nous présentons ici l'historique du projet ainsi qu'une description de la société Icosnet, ses activités et un aperçu de sa structure générale. Nous fournissons également un bref aperçu du travail effectué dans cette étude, en soulignant son importance et ses objectifs.

## **I.2 Historique d'Icosnet :**

ICOSNET est une entreprise spécialisée dans les solutions numériques et les technologies de l'information. Elle a franchi plusieurs étapes depuis sa création jusqu'à aujourd'hui, on peut citer les étapes suivantes :

**De 1999 à 2001 :** la fondation de la société Icosnet et le lancement services accès internet.

**2005 :** Approbation VOIP.

**2007 :** Approbation Wimax.

**2008 :** Acquisition de numérotation VOIP.

**2009 :** Installation POP Londres.

**2013 :** Installation POP Madrid.

**2015 :** Installation POP Paris.

**2017 :** Lancement solution Vasii.

**2019 :** Lancement data Center Alger.

**2020 :** Autorisation cloud.

**2021 :** Approbation Center Oran [1].

## **I.3 Présentation d'icosnet :**

Depuis 1999, Icosnet est un opérateur Télécom établi en Algérie. En tant qu'unique FAI privé du pays, Icosnet offre plusieurs services comprenant des communications unifiées, de la cyber sécurité et des solutions Cloud via de Data Center situé en Algérie, elle dispose de sa propre infrastructure donc elle est en mesure de proposer des solutions de pointe à des clients locaux et internationaux, s'adressant aux grands comptes locaux.



**Figure I.1 : Logo Icosnet**

**I.3.1 Adresse de la société Icosnet :**

Centre des Affaires El Qods, 6ème niveau de la tour Centrale, 16002, Chéraga, Algérie

**I.4 Infrastructure d'icosnet :**

**I.4.1 Datacenter :**

- Une salle d'énergie qui se dispose de toutes les armoires électriques du Datacenter (armoire réseau, armoire GE, armoires inverseur, TGBT, TGHQ...)
- Une salle de câblage télécom (fibre optique et cuivre), pour la connexion des liens clients et nos liens infrastructure
- Une salle serveur 02 avec 14 racks, deux sont dédiés pour la partie télécom et 12 pour la partie serveurs.
- Un local qui héberge les deux groupes électrogènes.
- Une salle opération pour la supervision et les interventions des clients
- Une salle fermée, sécurisée pour les bouteilles d'extinction

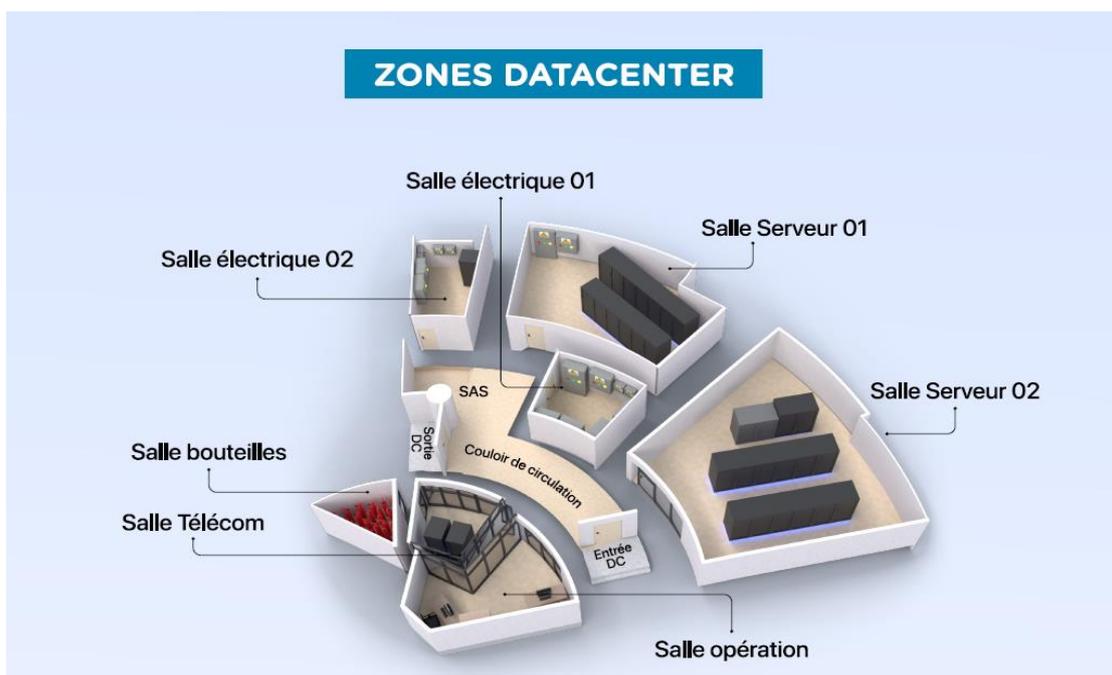


Figure I.2 : Zone datacenter

**I.4.2 Infrastructure réseau :**

Le Datacenter est connecté avec le backbone national avec un lien de 1 Gbps et un autre backbone international via deux liens 2.5 Gbps (ALPAL 2) et 1.2 Gbps (SeMeWe 4) sur nos deux Pops Londres et Madrid respectivement. Et pour se connecter à internet, ils sont connectés sur plusieurs services de transit IP avec ces partenaires tels que (Orange, Vodafone, Telefonica, telecomitalia... ETC...)

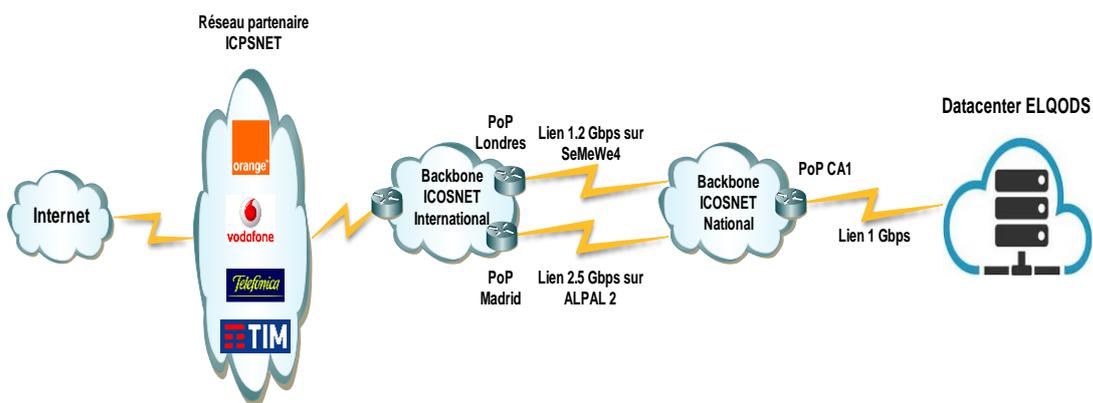


Figure I.3 : Le réseau d'icosnet

En 2021, icosnet a déployé son second Data center pour assurer le disaster recovery sur l'ouest du pays basé à Oran, à 400Km d'Alger.

### I.4.3 Infrastructure virtuelle

La société dispose des clusters VMware convergés avec les caractéristiques ci-dessous :

#### I.4.3.1 Site principal :

- Un cluster management basé sur VMware
- Un cluster de production basé sur VMware
- Gestion réseau virtuel « SDN » via NSX
- Gestion de stockage virtuel via VSAN
- Plateforme de sauvegarde sous VEEAM

#### I.4.3.2 Site Secours :

Cluster de production sous VMware

Dépôt pour la réplication des sauvegardes du site principal

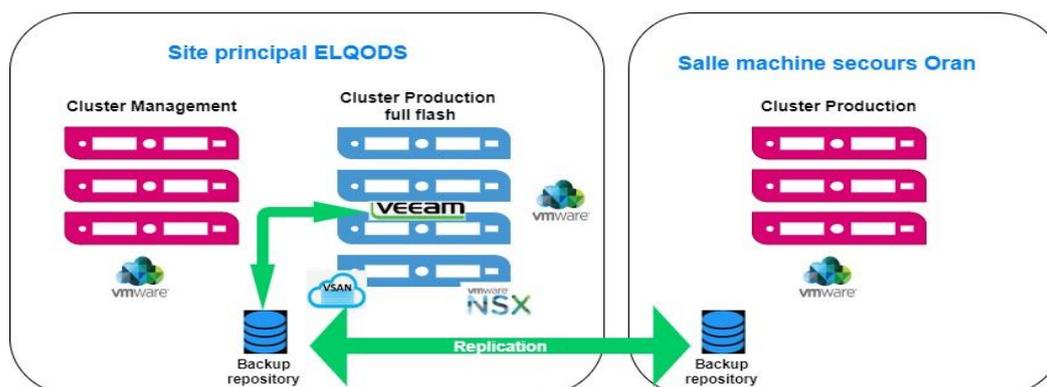


Figure I.4 : Infrastructure virtuelle

### I.5 Service de la société :

Les opérations de la société sont diversifiées dans quatre catégories de produits distinctes, notamment les services cloud, la cyber sécurité, la communication collaborative et l'accès Internet.

### I.6 Conclusion :

Dans ce premier chapitre nous avons présenté la société et l'environnement dans lequel nous avons développé ce travail et fait un rappel des différentes étapes suivies afin de mieux comprendre l'intérêt des différents chapitres qui vont suivre. Dans le chapitre qui suit nous présenterons le plan d'action et expliquerons l'intérêt du choix de la solution adoptée.

# Chapitre II

## Cloud computing

## II.1 Introduction :

Le paysage numérique actuel est transformé par une révolution majeure : l'avènement de l'informatique dématérialisée, couramment désignée sous le terme de Cloud computing, cette évolution technologique occupe une place essentielle dans la transformation numérique de nombreux industries et entreprises.

Avec l'essor du travail à distance, de l'apprentissage en ligne et de la collaboration virtuelle, le Cloud fournit une infrastructure flexible et accessible pour satisfaire la demande croissante de capacité de stockage, de puissance de traitement et de facilité d'accès aux données. En permettant une évolutivité rapide, une résilience améliorée et une flexibilité incomparable, Le Cloud computing est désormais indispensable pour soutenir les activités commerciales et les interactions en ligne dans cet environnement en perpétuelle évolution.

## II.2 Historique :

Bien que le Cloud computing remonte aux débuts de l'informatique, sa forme moderne a surtout émergé au cours des dernières décennies. Un résumé de son évolution est le suivant :

- **Origine du concept (années 1950-1960) :**

Dans les années 1950 et 1960, les chercheurs ont d'abord proposé l'idée de partager des ressources informatiques et d'accéder aux ordinateurs à distance, ont exploré les concepts de partage de temps et d'informatique mainframe et ont jeté les bases conceptuelles du cloud computing.

- **Développement des réseaux et d'Internet (années 1970 à 1990) :**

Avec l'avènement des réseaux informatiques et d'Internet dans les années 1970 et 1980, de nouvelles possibilités d'allocation des ressources informatiques sont apparues. Les premières formes de services en ligne étaient disponibles, mais avec une infrastructure et une connectivité limitée.

- **L'apparition des services de cloud computing dans les années 2000 :**

Dans les années 2000, des entreprises comme Amazon, Google et Salesforce ont commencé à proposer des services commercialisables de cloud computing. En 2002, Amazon Web Services (AWS) a été introduit pour offrir des services d'infrastructure à la demande. Plus tard, Google App Engine a été lancé en 2008, fournissant une plateforme robuste dédiée au développement d'applications. Salesforce a aussi introduit sa plateforme Cloud en 1999, afin de proposer des services CRM en ligne.

- **Essor fulgurant et pénétration massive dans tous les secteurs entre 2010 et 2020 :**

Au cours de la décennie suivante, Le Cloud computing a atteint une pénétration ubiquitaire. Dans le domaine des technologies de l'information L'engouement des entreprises pour le cloud a été motivé par la recherche d'agilité, d'économies, de mise à l'échelle dynamique et de sécurité renforcée. De nouveaux participants ont fait leur apparition, offrant une variété de services cloud, en comptant des solutions de calcul, de stockage, de bases de données, de machine Learning, etc.

- **Évolution versle multcloud et l'Edge computing (années 2020 et au-delà) :**

La tendance vers le multcloud, où les entreprises utilisent plusieurs fournisseurs de Cloud pour diversifier leurs services, ainsi que vers l'Edge computing, où le traitement des données est effectué plus près des utilisateurs finaux, sont devenues des thèmes importants dans le domaine du Cloud computing. Ces développements visent à améliorer la performance, la résilience et la sécurité des applications Cloud. Aujourd'hui, le Cloud computing est devenu un élément essentiel de l'infrastructure informatique pour de nombreuses entreprises et organisations à travers le monde, et il continue d'évoluer avec l'émergence de nouvelles technologies et de nouveaux besoins.

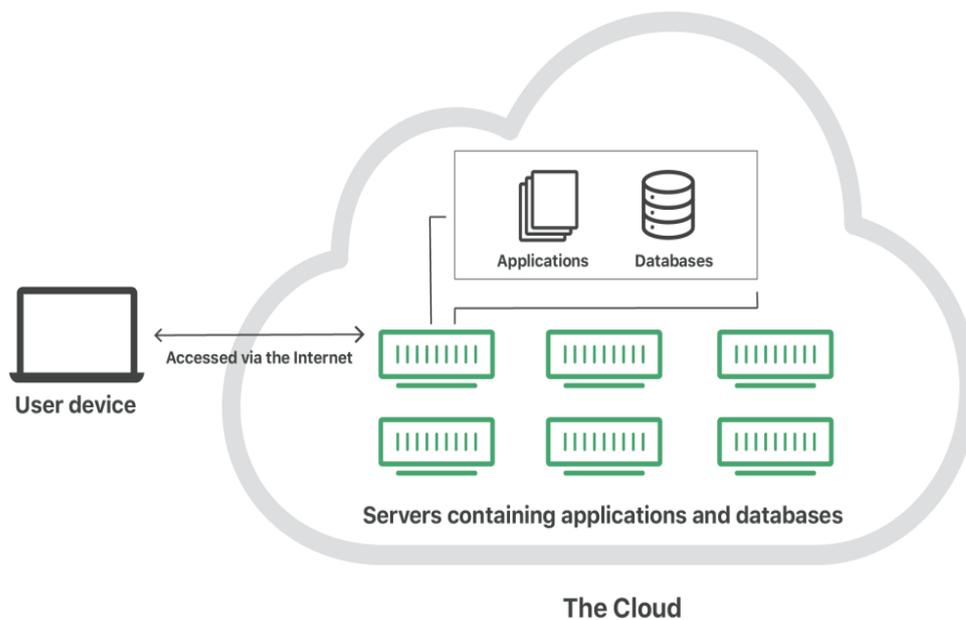
### **II.3 Définition de Cloud computing :**

NIST : (National Institute of Standards and Technology) en français " Institut national des normes et de la technologie " définit le Cloud Computing comme suit : « Le Cloud Computing est l'ensemble des disciplines, pratiques, technologies et modèles commerciaux utilisés pour délivrer comme un service à la demande et par le réseau des capacités informatiques,(logiciels, plateformes, matériels) ».

Dans la littérature il y a d'autres définitions de Cloud Computing nous choisisons les suivants :

- Un modèle qui offre aux utilisateurs du réseau un accès à la demande, à un ensemble de ressources informatiques partagées et configurables, aussi avec le,Cloud, n'importe qui peut entrer et à partir de n'importe quel endroit et à tout moment.
- C'est un concept abstrait, son but est de pousser les entreprises à externaliser les ressources numériques qu'elles stockent. Ces ressources offrant des capacités de stockage et de calcul, des logiciels de gestion de messagerie, et d'autres services sont mises à disposition par des sociétés tierces et accessibles, grâce à un système d'identification, via un PC et une connexion à Internet [2].

Elles n'ont plus besoin de mise à jour et de maintenance de leurs propres serveurs, cette tâche est garantie par le fournisseur de cloud ceci est particulièrement avantageux pour les petites entreprises qui ne disposent pas toujours des ressources nécessaires pour leur propre infrastructure interne, mais qui peuvent externaliser leurs besoins en infrastructure à moindre coût grâce au cloud, de plus, le cloud facilite les opérations internationales des entreprises, car les employés et les clients ont la possibilité d'accéder aux mêmes fichiers et applications depuis n'importe quel endroit sur la planète [3].



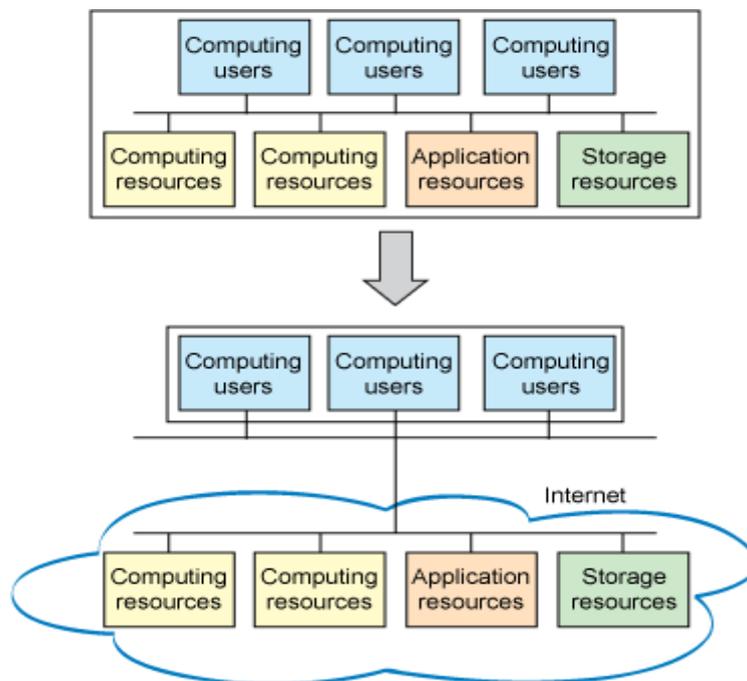
**Figure II.1 : Cloud computing [3].**

#### **II.4 Comment fonctionne le Cloud computing :**

L'informatique dans le cloud est composée d'une technologie appelée virtualisation, cela permet de créer une simulation d'ordinateur entièrement numérique, appelée machine virtuelle, qui fonctionne comme un ordinateur physique complet avec ses propres équipements matériels. Les machines virtuelles qui sont connectées à un même hôte sont isolées dans des environnements appelés "sandbox", ce qui signifie qu'elles n'interagissent pas les unes avec les autres, ainsi, les fichiers et applications d'une machine virtuelle ne peuvent être accessibles par les autres, même si elles sont hébergées sur le même matériel physique.

Les machines virtuelles permettent une meilleure utilisation plus efficace du matériel qui les héberge. En effectuant diverses machines virtuelles simultanément, un serveur peut être divisé en plusieurs serveurs virtuels, et un centre de données peut être segmenté en une multitude de centres de données virtuels, capables de servir de nombreuses organisations, cela

permet aux fournisseurs de cloud d'utiliser leurs serveurs à un nombre beaucoup plus important de clients, et ce qui a un coût élevé. [3].



**FigureII.2 Le cloud computing migre les ressources au sein d'internet [4]**

#### II.4.1 Machine virtuelle :

Une machine virtuelle représente un environnement qui opère comme un système informatique autonome, avec ses propres ressources telles que processeur, mémoire, interface réseau et espace de stockage, cependant, elle est créée au sein d'un système matériel physique, que ce soit sur site ou à distance.

L'hyperviseur, un logiciel dédié, assure la séparation des ressources de la machine virtuelle du matériel sous-jacent et les provisionne de manière appropriée pour leur utilisation par la machine virtuelle. Les machines physiques dotées d'un hyperviseur tel que KVM (Kernel-based Virtual Machine) sont communément appelées hôtes, ordinateurs hôtes, systèmes d'exploitation hôtes ou simplement hôtes. Les multiples machines virtuelles qui utilisent ces ressources sont désignées comme des invités, ordinateurs invités, systèmes d'exploitation invités ou simplement invités. L'hyperviseur gère les ressources de calcul (processeur, mémoire, stockage) comme un pool de ressources, permettant leur allocation sans difficulté entre les invités existants ou pour de nouvelles machines virtuelles [5].

### II.4.2 Hyperviseur :

C'est un logiciel qui a pour but de créer et de gérer des machines virtuelles. Aussi appelés moniteurs de machines virtuelles, il s'occupe des systèmes exploitation et des ressources des machines virtuelles, ce qui les permet de créer et d'administrer, et quand il est utilisé comme hyperviseur, le matériel physique est désigné comme l'hôte, tandis que toutes les machines virtuelles qui utilisent ses ressources désignées comme invitées, l'hyperviseur permet de gérer les ressources (processeur, mémoire, stockage) sous forme d'un pool, ce qui permet de les réallocations entre les invites existants et les nouvelles machines virtuelles.

Afin d'exécuter des machines virtuelles, tous les hyperviseurs nécessitent des composants nécessaires au niveau du système d'exploitation : gestionnaire de mémoire, ordonnanceur, pile d'entrées/sorties (E/S), pilotes de périphériques, gestionnaire de la sécurité, pile réseau, etc. L'hyperviseur gère les ressources allouées à chaque machine virtuelle et gère la planification des ressources en fonction des capacités physiques disponibles, le matériel est toujours en mesure d'exécuter les instructions demandées par les machines virtuelles, tandis que l'hyperviseur est en charge de la planification. La virtualisation est un atout majeur car elle permet d'exécuter différents systèmes d'exploitation en parallèle et de partager les mêmes ressources matérielles virtuelles, sans cette technologie, un seul système d'exploitation pourrait être effectué sur le matériel.

Il existe une variété d'hyperviseurs, qu'ils proviennent de la communauté Open Source ou qu'ils sont proposés par des fournisseurs traditionnels. VMware est l'un des principaux fournisseurs de solutions de virtualisation, mettant à disposition des produits tels que l'hyperviseur ESXi et la plateforme de virtualisation vSphere [5].

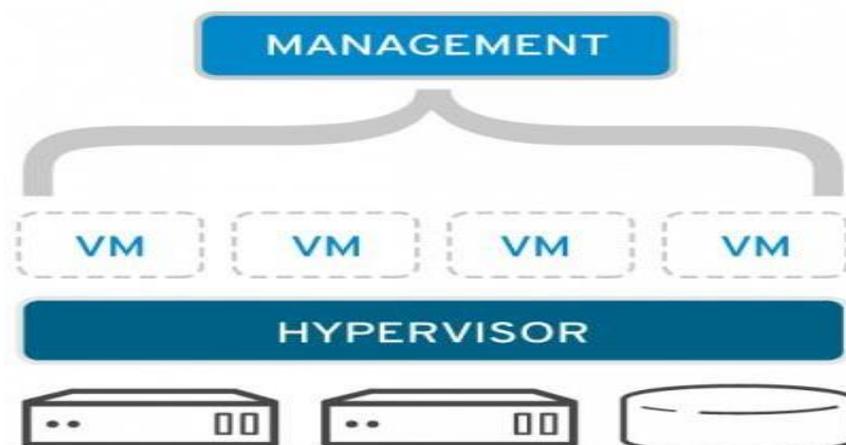


Figure II.3 : Hyperviseur [5]

## II.5 Composants de Cloud computing :

Voici quelques éléments importants de l'architecture moderne du cloud computing d'aujourd'hui :

### II.5.1 Data center (Centres des données) :

Cet espace de données est géré et opéré par un prestataire de services cloud, les utilisateurs peuvent ensuite louer des serveurs et des infrastructures auprès du fournisseur, sans se soucier de l'approvisionnement, de l'achat ou de la maintenance des composants matériels et de l'espace requis, les data centres cloud ont également une meilleure capacité d'utilisation des ressources et de modalités de paiement. [6].



Figure II.4 : Datacenter [1]

### II.5.2 Capacités de mise en réseau :

Dans le domaine du cloud computing, il est important d'avoir une connectivité réseau haut débit. Un réseau étendu (WAN) relie les utilisateurs frontaux (comme les interfaces côté client accessibles via des dispositifs compatibles Web) aux fonctions back-end (comme les centres de données et les services applicatifs basés sur cette technologie). Les technologies avancées pour utiliser le cloud computing comme des équilibres de charge, des réseaux de diffusion de contenu (CDN) et des réseaux définis par logiciel (SDN) permettent aux utilisateurs de partager rapidement des informations entre les utilisateurs frontaux et les ressources back-end [4].

### **II.5.3 Virtualisation :**

Le cloud computing consiste à virtualiser l'infrastructure informatique, y compris les serveurs, les logiciels de système d'exploitation, le réseau et d'autres infrastructures abstraites, grâce à un logiciel spécial pour pouvoir être regroupées et séparées indépendamment des limites matérielles physiques, donc il est possible de diviser un seul serveur matériel en plusieurs serveurs virtuels [4].

#### **II.5.3.1 Définition de la virtualisation :**

La virtualisation est une technologie qui vous permet de créer différents environnements simulés ou ressources dédiées à partir d'un seul système physique. Son logiciel, appelé hyperviseur, est directement relié au matériel et permet de fragmenter ce système unique.

En plusieurs environnements de sécurité distincts, il s'agit de machines virtuelles, ces dernières ont pour capacité de séparer les ressources du matériel et de les distribuer de manière adéquate.

La virtualisation permet de tirer le meilleur parti de vos anciens investissements, le matériel physique équipé d'un hyperviseur est appelé « hôte », tandis que toutes les machines virtuelles utilisant ses ressources sont appelées « invités », ces invités s'occupent des ressources de calcul (processeur, mémoire, stockage) de manière à ce qu'ils puissent les déplacer sans difficulté.

Les opérateurs peuvent contrôler les instances virtuelles du processeur, de la mémoire, du stockage et d'autres ressources, de façon à ce que les invités reçoivent les ressources dont ils ont besoin, lorsqu'ils en ont besoin [5].

### **II.6 Avantages de Cloud computing :**

Le Cloud computing présente plusieurs avantages par rapport à l'informatique traditionnelle sur site, où une entreprise doit posséder et entretenir des centres de données physiques et des serveurs pour accéder à la puissance de calcul, au stockage de données et à d'autres ressources. Parmi eux, mentionnons :

<b>Flexibilité</b>	<b>Efficacité</b>	<b>Valeur stratégique</b>
<p><b>Évolutivité :</b> L'infrastructure Cloud est extensible à la demande afin de s'adapter aux charges de travail changeantes.</p>	<p><b>Accessibilité :</b> Les applications et les données hébergées dans le cloud sont accessibles depuis quasiment tout appareil connecté à Internet.</p>	<p><b>Mises à jour régulières :</b> Les fournisseurs de services mettent régulièrement à jour les offres pour offrir aux utilisateurs la technologie la plus récente.</p>
<p><b>Options de stockage :</b> Les utilisateurs ont la possibilité de sélectionner entre des offres de stockage public, privé ou hybride, selon leurs exigences en matière de sécurité et d'autres critères à prendre en compte.</p>	<p><b>Rapidité de mise sur le marché :</b> Le développement dans le Cloud permet aux utilisateurs de commercialiser rapidement leurs applications.</p>	<p><b>Collaboration :</b> L'accès partout dans le monde permet aux équipes de collaborer à partir d'emplacements très éloignés les uns des autres.</p>
<p><b>Choix du degré de contrôle :</b> Les organisations peuvent déterminer le degré de contrôle souhaité grâce aux options sous forme de service. Il peut s'agir de logiciels sous forme de service SaaS ou PaaS ou IaaS,</p>	<p><b>Économies sur les équipements :</b> Le Cloud computing utilise des ressources distantes, permettant aux entreprises d'économiser sur le coût des serveurs et autres équipements.</p>	<p><b>Rationalisation du travail :</b> Les fournisseurs de services Cloud gèrent l'infrastructure sous-jacente, permettant aux entreprises de se concentrer sur le développement d'applications et sur d'autres priorités.</p>
<p><b>Sélection des outils :</b> Les utilisateurs peuvent sélectionner des outils et des fonctionnalités préconfigurés dans un menu pour créer une solution adaptée à leurs besoins.</p>	<p><b>Structure de paiement à l'utilisation :</b> Une structure de paiement à l'utilisation signifie que les utilisateurs paient uniquement pour les ressources qu'ils utilisent.</p>	<p><b>Avantage concurrentiel :</b> Les organisations peuvent évoluer avec plus de souplesse que leurs concurrents qui doivent consacrer des ressources informatiques à la gestion de l'infrastructure.</p>
<p><b>Fonctions de sécurité :</b> Le Cloud privé virtuel, le chiffrement et les Clés API permettent de sécuriser les données.</p>	<p><b>Sécurité des données :</b> Les pannes matérielles n'entraînent pas de perte de données grâce aux sauvegardes réseau</p>	

**Tableau II.1 : Les avantages de cloud computing [5].**

### II.7. Services de Cloud Computing :

Il y a trois façons d'utiliser Cloud Computing : les services d'infrastructure, des plateformes et des logiciels, chaque modèle de service est différent du cloud computing et partage ses propres responsabilités avec le fournisseur de services [7].

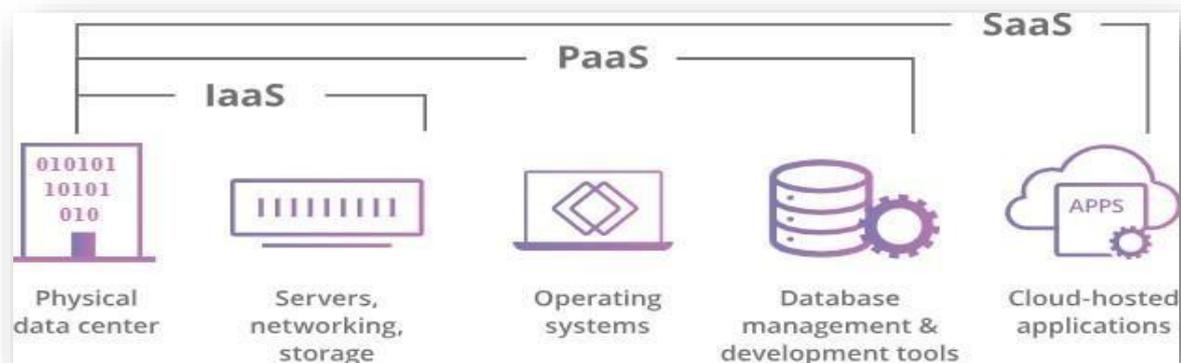


Figure II.5 : Les services de Cloud [3].

#### II.7.1 Infrastructure as a service (IaaS) :

L'IaaS couche du Cloud Computing, la plus complexe en termes de gestion est l'infrastructure comme un service (IaaS). L'infrastructure fournit des capacités de calcul, de stockage et une bande passante suffisante. Cette infrastructure est mise à disposition de façon à gérer automatiquement la charge de travail requise par les applications.

Il y a très peu de limitation pour le client si ce n'est la partie matérielle qui peut être contournée grâce aux systèmes de virtualisation. Les applications vont dès lors pouvoir être déployées sans être liées à un serveur spécifique. La virtualisation répond de manière dynamique là où les serveurs physiques fournissent un ensemble de ressources allouées selon les besoins, et où la relation entre les applications et les ressources de calcul, de stockage et de réseau pourront s'adapter de manière automatique pour répondre à la charge de travail et aux exigences demandées [8].

#### Parmi les fournisseurs de services IaaS :

- OpenStack.
- AWS EC2
- DigitalOcean
- Google Compute Engine (GCE)
- Microsoft Azure

### II.7.2 Platform as a service (PaaS):

La plateforme comme un service (PaaS), est la plateforme d'exécution, de déploiement et de développement des applications. Le client maintient ses applications, le fournisseur maintient : les runtimes, l'intégration SOA4, les bases de données, le logiciel serveur, la virtualisation, le matériel serveur, le stockage et les réseaux.

Un service PaaS met à disposition des environnements de développement prêts à l'emploi, fonctionnels et performants. Parmi les solutions : Windows Azure de Microsoft, AppEngine de Google, Force.com de Salesforce. Chaque fournisseur de PaaS propose des environnements de développement différents, Google AppEngine se limite à Java et Python, tandis Windows Azure permet de travailler avec les langages .NET, PHP, Python, Ruby et Java [8].

### II.7.3 Software as a service (SaaS):

Ce type de service, des applications sont mises à la disposition des consommateurs. Les applications peuvent être manipulées à l'aide d'un navigateur web ou installées de façon locale sur un PC, et le consommateur n'a pas à se soucier d'effectuer des mises à jour, d'ajouter des patches de sécurité et d'assurer la disponibilité du service.

Gmail est un exemple de tel service. Il offre au consommateur un service de courrier électronique et le consommateur n'a pas à se soucier de la manière dont le service est fourni.

#### **Avantage :**

Plus d'installation, plus de mise à jour (elles sont continuées chez le fournisseur), plus de migration de données etc. Paiement à l'usage. Test de nouveaux logiciels avec facilité

#### **Inconvénient :**

Limitation par définition au logiciel proposé. Pas de contrôle sur le stockage et la sécurisation des données associées au logiciel. Réactivité des applications Web pas toujours idéale [7], **Parmi les exemples d'applications SaaS, on citera :**

- Salesforce,
- Cisco WebEx
- Dropbox
- Google Workspace
- MailChimp
- Slack
- Trello
- Veeam
- Zoom

## II.8 Modèles de déploiement du Cloud computing :

Après avoir sélectionné le ou les services de Cloud Computing que vous avez choisis, vous pouvez choisir entre trois principaux modèles de déploiement de Cloud Computing, le Public Cloud, le Privé Cloud, le Hybrid Cloud et le Multicloud.

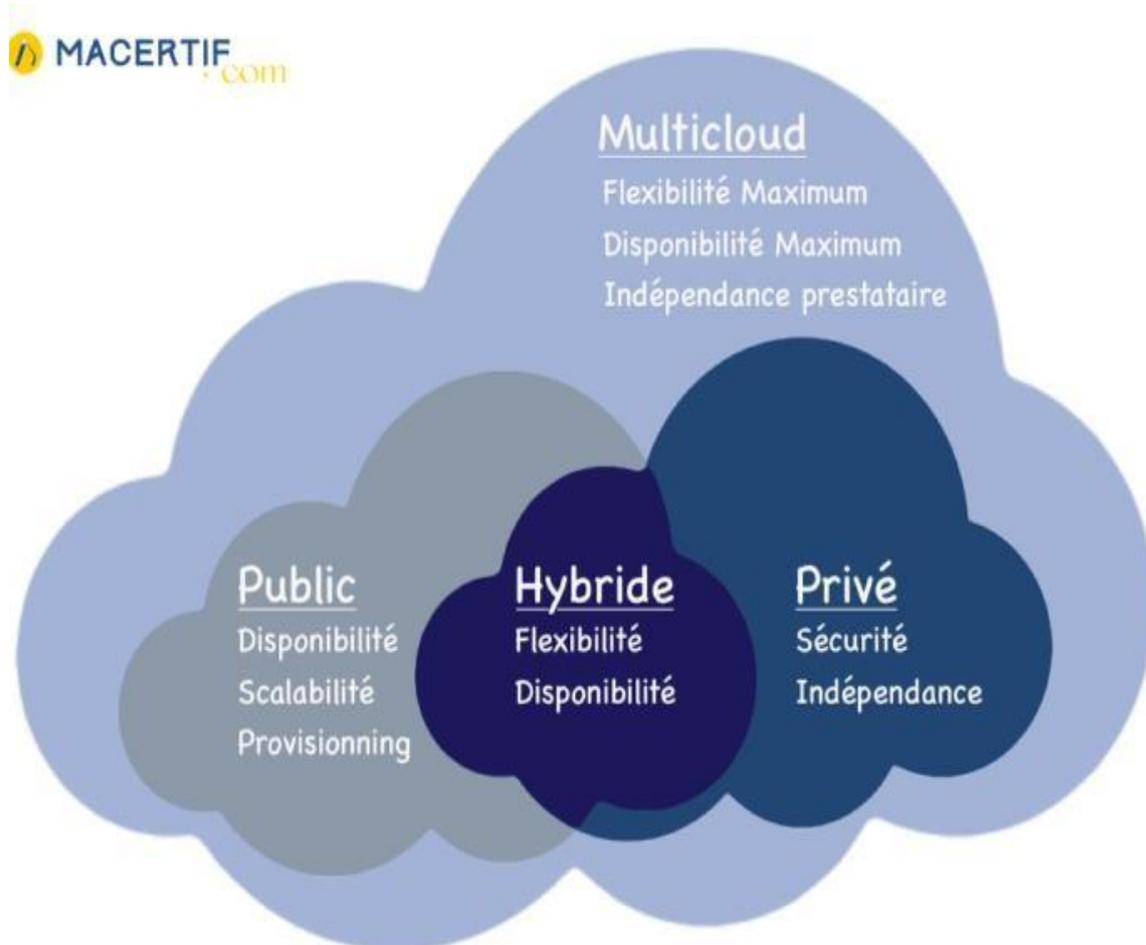


Figure II.6 : Modèles de déploiement du Cloud [9].

### II.8.1 Cloud public :

Un Cloud public repose sur le modèle standard de Cloud computing, dans lequel un fournisseur de services rend des ressources, telles que des applications et du stockage, accessibles au grand public via Internet. Les services de Cloud public disponibles pour quiconque souhaite les utiliser ou les acheter, C'est une structure souple et ouverte, géré par un fournisseur tiers.

#### Exemple :

Microsoft Azure, Amazon Web Services (AWS), ...etc. [10].

### II.8.2 Cloud privé :

Un Cloud privé est un ensemble des services et des ressources disponibles à un seul client par exemple une entreprise.

Il peut être géré par l'entreprise elle-même, ou bien avec ses branches, dans ce cas il s'appelle « Le Cloud privé Interne ».

Il peut être géré par un prestataire externe loué par l'entreprise, dans ce cas on l'appelle « Le Cloud privé Externe », il est accessible via des réseaux sécurisés de type VPN (Virtual Private Network). L'avantage de ce type de Cloud par rapport au Cloud public réside dans l'aspect de la sécurité et la protection des données. Ce type de cloud est :

- Cher pour le client.
- Dédié et sécurisé.
- Moins flexible comparé au cloud public [10].

### II.8.3 Cloud hybride :

Avec le modèle hybride, les organisations peuvent associer le Cloud privé avec le public, de cette façon, ces organisations peuvent déployer leurs principales et sensibles applications sur leur Cloud privé et déployer le reste sur le Cloud public, ce qui peut réduire considérablement les coûts de construction et de management des ressources sur le Cloud.

#### Exemples:

- Google Apps, Amazon s3 [10].

### II.8.4 Multicloud :

Le multicloud est une stratégie de cloud computing qui combine les meilleurs services de plusieurs fournisseurs de cloud pour créer une solution. Cette stratégie se base sur le travail (workload), à l'entreprise et à la gouvernance des données. Une solution multicloud qui comprend l'IaaS, le PaaS et le SaaS dans une architecture étroite et faible. Une solution multicloud bien conçue doit considérer le réseau, la performance, la sécurité, la gestion des opérations et le coût total [12].

**II.9 Exemples d'utilisation de cloud :**

- **Google Drive** : Google Drive est un pur service Cloud Computing. Il peut stocker des documents en ligne en utilisant des applications Cloud Google Docs, Google Sheets et Google Slides. Ce service est disponible sur un ordinateur, depuis une tablette, ou même depuis un smartphone, Il peut être utilisé comme des applications mobiles appelées Docs et Sheets. La plupart des services Google sont classés dans le domaine du Cloud Computing. C'est le cas de Gmail, Google Calendar, et Google Maps par exemple [12].

**Figure II.7: Google drive**

- **Apple iCloud** : est principalement utilisé pour stocker en ligne et le backup. Il est également utile pour synchroniser des emails, des contacts et le calendrier. Vous pouvez utiliser toutes les informations sur iPhone, Mac OS ou Windows depuis le panneau de contrôle iCloud. Apple propose aussi des versions Cloud de son traitement de texte Pages, de sa feuille de calcul Spreadsheet, et de son logiciel de présentation Keynote. La plateforme aide les utilisateurs d'iPhone à utiliser la fonctionnalité Localiser mon iPhone.[13]

**Figure II.8 : Cloud**

- **Amazon Cloud Drive** : Amazon propose essentiellement le stockage de musique au format MP3 et de photos. Les abonnés Amazon Prime bénéficie d'une capacité de stockage illimitée. Amazon Cloud Drive sert également à stocker le contenu acheté pour Kindle. En résumé, cette plateforme sert à stocker tout le contenu numérique acheté auprès d'Amazon [13].



Figure II.9: Amazon Cloud drive

- **Ibox de icosnet** : Un espace cloud entièrement sécurisé, situé en Algérie, offre de nombreux avantages pour collaborer, stocker et partager des données avec une grande flexibilité. L'interface graphique ibox offre une interface graphique plaisante, une prise en charge facile et des menus clairs. Il est possible de sauvegarder vos fichiers sans restriction grâce à ces offres qui ne sont pas limitées [1].



Figure II.10: Ibox [1].

**II.10 Conclusion :**

Il est évident que le cloud computing pose des défis majeurs en matière de confidentialité et de sécurité. Pour préserver la confidentialité, le chiffrement est une option, mais il est important de faire preuve de rigueur dans le choix d'un service de cloud computing.

Le commerce en ligne était également confronté à des controverses à l'époque du développement du Web, cependant, chaque année, des milliards de dollars sont consacrés aux transactions de commerce électronique à l'échelle mondiale. Ainsi, Les technologies classiques telles que Secure Sockets Layer (SSL) seront mises à profit pour assurer la sécurité du Web dans le cloud computing.

La course au cloud computing est ouverte avec le développement open source sur Linux. Il est évident que nous contribuons à un retour aux centres de données centralisés en raison des investissements considérables dans le cloud computing, il sera très intéressant d'observer les nouvelles technologies et les nouvelles architectures.

# Chapitre III

## Stockage objet

### III.1 Introduction

Le stockage cloud est une méthode innovante pour la gestion, la conservation et la récupération à distance des données. Contrairement au stockage local, il repose sur des centres de données distants gérés par des tiers. Cette solution flexible et fiable présente un avantage économique indéniable. Les données sont accessibles depuis presque tous les appareils connectés au cloud. Les services de stockage cloud mettent l'accent sur la sécurité, intégrant souvent le chiffrement des données, des sauvegardes automatiques et une authentification multi-facteurs. Dans un monde de plus en plus numérique, le stockage cloud simplifie la gestion des données et offre flexibilité et accessibilité, tant pour les entreprises que pour les particuliers. Dans ce chapitre, nous explorerons un type de stockage en particulier, le stockage objet, qui révolutionne la gestion des données par les entreprises.

### III.2 Types de donnée :

On peut distinguer deux types de données les plus connus et ce sont :

#### III.2.1 Données structurées :

Les données structurées suivent un format standardisé pour une accessibilité efficace par les logiciels et les individus, typiquement organisées sous forme de tableau avec des rangées et des colonnes, elles définissent clairement les attributs des données, en raison de leur caractère quantitatif, les ordinateurs peuvent les traiter efficacement pour en extraire des informations significatives. Par exemple, un tableau structuré de données clients, avec des colonnes telles que nom, adresse et numéro de téléphone, permet de fournir des informations telles que le nombre total de clients ou la localité avec le plus grand nombre de clients [14].

#### III.2.2 Données non structurées

Les données non structurées se réfèrent à des informations qui ne suivent pas de modèle de données préétabli et ne peuvent donc pas être stockées dans une base de données relationnelle traditionnelle ou un système de gestion de base de données relationnelle (SGBDR). Les textes et le multimédia sont deux exemples courants de contenus non structurés.

De nombreux documents professionnels, ainsi que les e-mails, les vidéos, les photos, les pages Web et les fichiers audio, entrent dans cette catégorie.

Entre 80 % et 90 % des données générées et collectées par les organisations sont non structurées, et leur volume augmente rapidement, bien plus rapidement que celui des bases de données structurées [15].

Les données non structurées sont plus cruciales, étant donné que plus de 80 % des données commerciales sont disponibles sous cette forme. Cette importance devrait encore croître rapidement en 2023 et au-delà. Outre le volume croissant, ces données non structurées contiennent des informations précieuses. Par exemple, les factures d'achat peuvent aider un fournisseur de télécommunications à segmenter ses clients en fonction de leurs données.

Malgré leur importance, de nombreuses entreprises sont confrontées à des difficultés pour accéder et utiliser les données non structurées. Certains de ces défis sont :

- Incapacité à traiter des volumes de données croissants
- Accéder aux données cloisonnées
- Non-conformité réglementaire
- Convivialité réduite des données
- Vulnérabilité accrue aux cyberattaques [16].

### **III.2.3 Comparaison entre les données non structurées et données structurées :**

Commençons par les données structurées : elles sont habituellement stockées dans une base de données relationnelle, parfois appelée données relationnelles. Ces données peuvent être facilement organisées dans des champs désignés, tels que des champs pour les codes postaux, les numéros de téléphone et les numéros de cartes de crédit. Les données conformes à la structure du SGBDR peuvent être facilement recherchées à l'aide de requêtes définies par l'homme ou de logiciels comme SQL.

En revanche, les données non structurées ne correspondent pas à ces modèles de données prédéfinis et ne peuvent pas être stockées dans un SGBDR. Du fait qu'elles sont disponibles dans de nombreux formats, leur assimilation, leur traitement et leur analyse posent un défi majeur pour les logiciels conventionnels. Des outils appropriés peuvent être utilisés pour entreprendre des recherches de contenu simples sur des données textuelles non structurées [15].

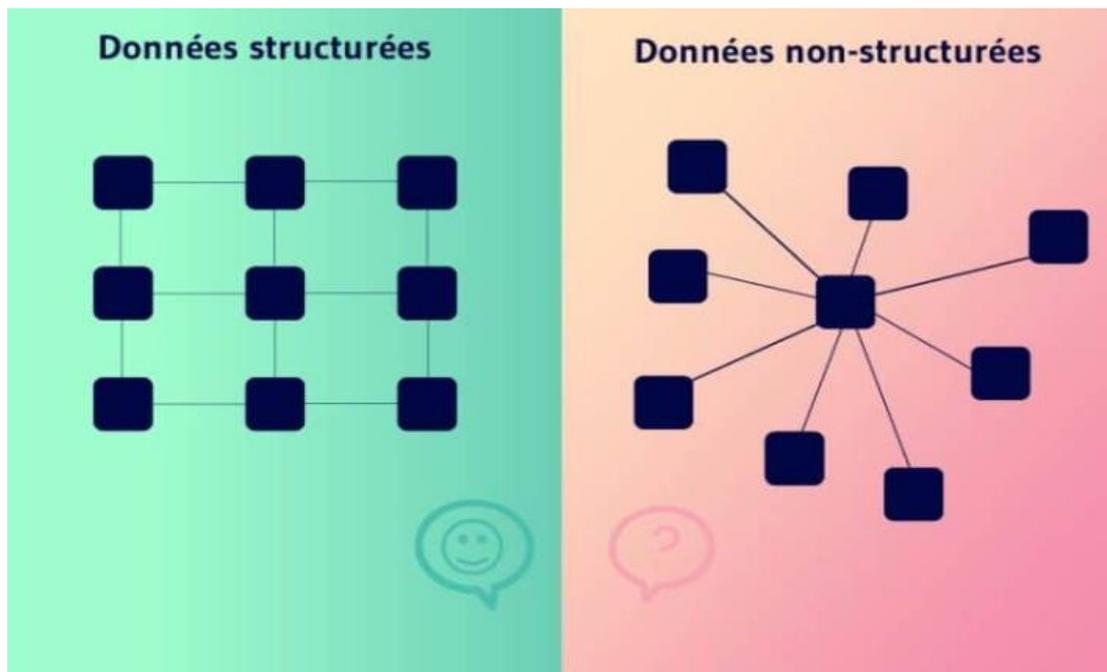


Figure III.1 : types de données [16].

### III.3 Types de stockages

Il existe trois types de stockage connus dans le cloud :

#### III.3.1 Stockage fichier :

Le stockage de fichiers fonctionne en organisant les données dans des dossiers, semblable à la façon dont vous stockez des fichiers physiques dans un système de classement papier dans un bureau. Pour accéder à des informations spécifiques dans un fichier, vous devez savoir où ce document est situé - dans quelle pièce, quel placard, quel tiroir et quel classeur. Cette même structure de stockage hiérarchique est utilisée pour le stockage de fichiers : les fichiers sont nommés, marqués avec des métadonnées, puis rangés dans des dossiers.

Pour retrouver une donnée, vous devez connaître son chemin d'accès. Avec le temps, la recherche et la récupération de fichiers peuvent prendre du temps à mesure que le nombre de fichiers augmente. Bien que l'évolutivité soit limitée, le stockage de fichiers reste un moyen simple de stocker de petites quantités de données de tout type et de les rendre accessibles à plusieurs utilisateurs simultanément [18].

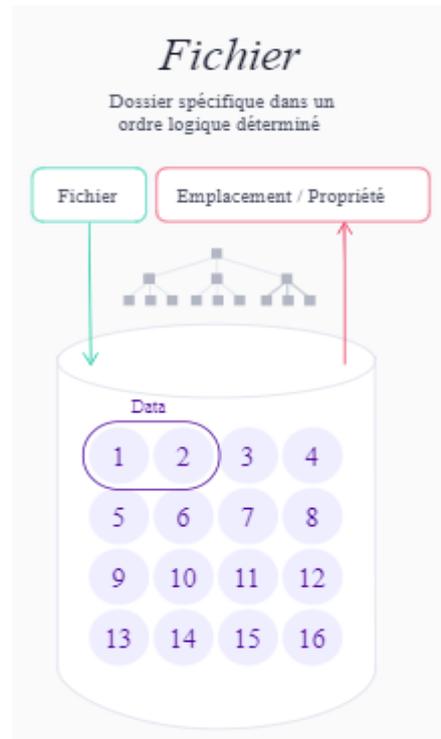


Figure III.2: Stockage Fichier [21].

### III.3.2 Stockage blocs:

Le stockage de blocs améliore les performances du stockage de fichiers en fragmentant les fichiers en blocs distincts et en les stockant séparément. Dans un système de stockage de blocs, chaque fragment de données brute se voit attribuer un identifiant unique, ce qui permet de les assembler dans le fichier complet lorsque cela est nécessaire. Contrairement au stockage de fichiers, le stockage de blocs ne nécessite pas un seul chemin d'accès aux données. Ainsi, vous pouvez le stocker à l'emplacement le plus pratique et le récupérer rapidement en cas de besoin.

Le stockage de blocs est efficace pour les organisations manipulant de grandes quantités de données transactionnelles ou des applications critiques nécessitant des performances constantes avec un délai minimal. Cependant, ce type de stockage peut être onéreux, ne propose pas de fonctionnalités de métadonnées et nécessite un système d'exploitation pour accéder aux blocs [18].

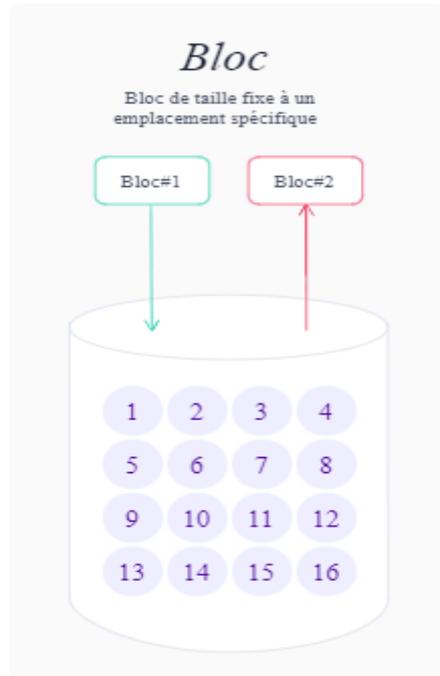


Figure III.3: Stockage Bloc [21].

### III.3.3 Le stockage Objet :

La technologie de stockage d'objets concerne le stockage et la gestion de données dans un format non structuré appelé objets, dans le contexte actuel, les organisations manipulent d'importantes quantités de données non structurées, comprenant des photos, des vidéos, des e-mails, des pages Web, des données de capteurs et des fichiers audio. Les systèmes de stockage d'objets dans le cloud répartissent ces données sur plusieurs appareils physiques tout en permettant aux utilisateurs d'y accéder efficacement à partir d'un référentiel de stockage unique et virtuel, cette approche est particulièrement adaptée aux applications cloud natives nécessitant évolutivité et flexibilité, ainsi qu'à l'intégration de magasins de données existants pour des besoins d'analyse, de sauvegarde ou d'archivage.

Un élément clé de la technologie de stockage d'objets réside dans l'importance des métadonnées, contrairement au stockage traditionnel qui organise les fichiers dans une hiérarchie de dossiers, le stockage d'objets regroupe les éléments de données constituant un fichier, y ajoute les métadonnées utilisateur et lui attribue un identifiant personnalisé, cette approche crée une structure plate, appelée compartiment, permettant la récupération et l'analyse de n'importe quel objet du compartiment, indépendamment de son type ou de ses caractéristiques fonctionnelles.

Le stockage d'objets est particulièrement adapté pour les lacs de données en raison de son architecture adaptée aux vastes volumes de données. Chaque élément de données est stocké individuellement en tant qu'objet, avec les métadonnées de l'objet fournissant un identifiant unique pour un accès plus facile. Cette architecture élimine les limitations de mise à l'échelle du stockage traditionnel, faisant du stockage d'objets le choix privilégié pour le stockage en cloud [12].

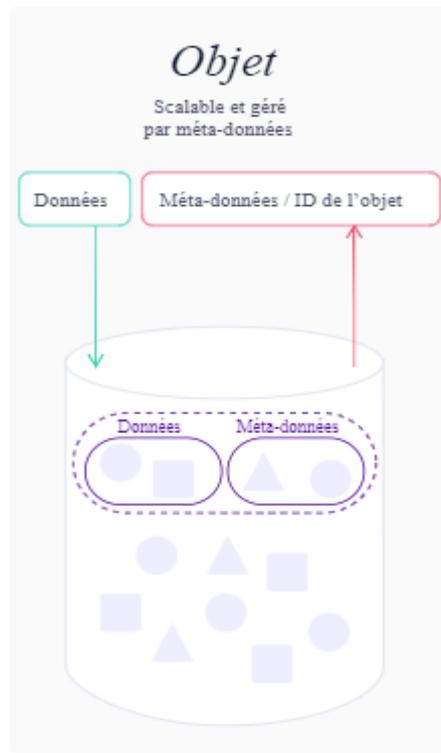


Figure III.4 : Stockage Objet [21]

### III.3.3.1 l'importance de stockage objet

À mesure que les entreprises se développent, elles doivent gérer des ensembles de données en constante expansion, provenant de diverses sources et utilisées par de multiples applications, processus métier et utilisateurs finaux. Une grande partie de ces données sont non structurées, se présentant sous différents formats et étant stockées sur divers supports, ce qui complique leur intégration dans un référentiel central. Cette complexité entrave l'innovation, car les données ne sont pas facilement accessibles pour l'analyse, l'apprentissage automatique ou le développement de nouvelles applications cloud natives. Le stockage d'objets contribue à surmonter ces silos en offrant une solution de stockage massivement évolutive et rentable pour conserver tout type de données dans leur format d'origine.

Cette approche supprime la complexité, les limitations de capacité et les barrières financières associées aux systèmes de stockage traditionnels, car le stockage d'objets permet une mise à l'échelle illimitée à des coûts par gigaoctet bas.

La gestion des données non structurées est facilitée grâce à une interface d'application conviviale. Il est possible d'optimiser les coûts de stockage des données en utilisant des politiques et de changer automatiquement de niveau de stockage si nécessaire. Le stockage d'objets dans le cloud simplifie l'analyse et la compréhension des données, ce qui accélère le processus décisionnel.

Alors que les objets peuvent être stockés localement, le stockage d'objets est spécialement conçu pour le cloud, offrant une capacité de mise à l'échelle pratiquement illimitée, une grande durabilité et une rentabilité optimale. Avec le stockage d'objets dans le cloud, les données sont facilement accessibles depuis n'importe quel endroit [14].

### III.3.3.2 Les avantages de stockage objet

Dans l'ère d'Internet et des communications numériques, où d'importantes quantités de données multimédias basées sur le Web sont générées à un rythme croissant, il est judicieux de considérer une solution de stockage objet pour conserver votre donnée

- **Stockage et gestion des données non structurées :** Le stockage objet est largement adopté à l'ère du cloud computing et pour la gestion des données non structurées, qui, selon les analystes, représenteront bientôt la majeure partie des données dans le monde. Le volume de contenu généré sur le Web, comprenant des e-mails, des vidéos, des publications sur les réseaux sociaux, des documents, des données de capteurs issues des appareils Internet des objets (IoT) et bien plus encore, est énorme et en constante croissance. Les données non structurées sont généralement statiques (immuables) mais peuvent être nécessaires à tout moment et en tout lieu, comme les images et les fichiers vidéo, ou les sauvegardes de données archivées, par exemple. Le stockage objet basé sur le cloud est une solution idéale pour la conservation des données à long terme. Il permet de remplacer les archives traditionnelles telles que le stockage en réseau (NAS), ce qui réduit votre infrastructure informatique. Il facilite également l'archivage et le stockage des données réglementaires obligatoires devant être conservées sur de longues périodes. De plus, il offre une solution rentable pour conserver de grandes quantités de contenu multimédia riche, telles que des images et des vidéos, qui sont rarement consultées.

- **Évolutivité** : L'évolutivité illimitée représente peut-être l'avantage le plus significatif du stockage de données basé sur les objets. Les objets, qui sont des unités discrètes de données pouvant être stockées en n'importe quelle quantité, sont conservés dans un environnement de données à structure plate, tel qu'un serveur de stockage. Il suffit d'ajouter simplement davantage de périphériques ou de serveurs en parallèle à un cluster de stockage d'objets pour permettre un traitement supplémentaire et répondre aux besoins de débits plus élevés requis par des fichiers volumineux tels que des vidéos ou des images.
- **Complexité réduite** : Le stockage d'objets simplifie le processus en éliminant la complexité associée à un système de fichiers hiérarchique comprenant des dossiers et des répertoires. Cette approche réduit le risque de ralentissement des performances et améliore l'efficacité lors de la récupération des données, car il n'y a pas de structure de dossiers, de répertoires ou de hiérarchies complexes à parcourir. Cela se traduit par une amélioration des performances, en particulier lors de la gestion de grandes quantités de données.
- **Reprise après sinistre/disponibilité** : Vous avez la possibilité de paramétrer les systèmes de stockage d'objets pour qu'ils dupliquent leur contenu. Ainsi, en cas de défaillance d'un disque au sein d'un cluster, une copie de sauvegarde est disponible, assurant la continuité et préservant les performances du système. Les données peuvent être dupliquées à l'intérieur des nœuds et des clusters, ainsi qu'entre des centres de données répartis géographiquement, offrant ainsi une sauvegarde supplémentaire hors site et même entre différentes régions géographiques. Le stockage objet représente une alternative plus efficace aux solutions de sauvegarde sur bande, qui nécessitent le chargement et le retrait physique des bandes des lecteurs et leur déplacement hors site pour assurer une redondance géographique. Vous pouvez utiliser le stockage objet pour sauvegarder automatiquement les bases de données sur site dans le cloud et dupliquer de manière rentable les données entre des centres de données répartis. Il est possible d'ajouter une sauvegarde supplémentaire hors site et même de la répliquer dans toutes les régions géographiques afin d'assurer une reprise après sinistre.
- **Métadonnées personnalisables** : Chaque objet constitue un conteneur autonome qui comporte des métadonnées ou des informations descriptives qui lui sont propres. Ces métadonnées sont utilisées pour des fonctions cruciales telles que les politiques de rétention, de suppression et de routage, les stratégies de reprise après sinistre (protection des données), ainsi que pour vérifier l'authenticité du contenu.

- **Abordabilité** : Les services de stockage d'objets adoptent un modèle de tarification à l'utilisation, ce qui élimine tout coût initial ou investissement en capital. Vous réglez simplement des frais d'abonnement mensuels pour une quantité définie de capacité de stockage, de récupération de données, d'utilisation de la bande passante et de transactions API. Cette tarification est généralement basée sur le niveau ou le volume, ce qui signifie que vous payez moins pour des volumes de données plus importants.
- **Compatibilité cloud** : Le stockage d'objets est souvent associé aux environnements cloud ou hébergés, où il est proposé en tant que service de stockage partagé. Cette solution permet à plusieurs entreprises ou services au sein d'une entreprise de partager le même référentiel de stockage, tout en ayant chacun accès à une portion distincte de l'espace de stockage. Cette approche de stockage partagé offre une optimisation intrinsèque en termes d'évolutivité et de coûts [4].

### III.3.3.3 Cas d'utilisation du stockage d'objets :

Il existe plusieurs cas d'utilisation du stockage d'objets. Par exemple, il peut vous aider des manières suivantes :

- **Proposez des médias riches** : Définissez des flux de travail en tirant parti de solutions de pointe pour la gestion des données non structurées. Réduisez vos coûts pour les médias riches distribués à l'échelle mondiale.
- **Gérer le contenu distribué** : Optimisez la valeur de vos données tout au long de leur cycle de vie et proposez des services de stockage compétitifs.
- **Adoptez l'Internet des objets (IoT)** : Gérez efficacement les données de machine à machine, prenez en charge l'intelligence artificielle et l'analyse, et réduisez le coût et le temps du processus de conception [19].

### III.4 Comparaison entre le stockage objet et les autres types :

#### III.4.1 Comparaison entre le stockage d'objets et le stockage fichier :

Les principales distinctions entre le stockage d'objets et le stockage de fichiers résident dans la structure des données et leur capacité à s'adapter à des volumes croissants, le stockage de fichiers est organisé de manière hiérarchique, utilisant des répertoires et des dossiers, et suit des protocoles de fichiers spécifiques tels que SMB, NFS ou Lustre. En revanche, le stockage d'objets adopte une structure plate, associée à des métadonnées et à un identifiant unique pour chaque objet, facilitant ainsi la recherche parmi un grand nombre d'autres objets.

Du fait de ces différences structurelles, le stockage de fichiers et le stockage d'objets présentent des capacités d'expansion distinctes.

Le stockage d'objets offre une évolutivité quasiment illimitée, atteignant des pétaoctets et des milliards d'objets. En revanche, en raison de sa nature hiérarchique et des chemins d'accès associés, le stockage de fichiers se heurte à des limitations en termes de capacité d'évolution [14].

#### III.4.2 Comparaison entre le stockage d'objets et le stockage par bloc :

Le stockage d'objets se révèle optimal pour la gestion de vastes ensembles de données non structurées, surtout lorsque la résilience, l'évolutivité infinie, la capacité d'adaptation et la gestion sophistiquée des métadonnées sont des critères déterminants pour des performances globales satisfaisantes.

D'un autre côté, le stockage par blocs offre une latence minimale et des performances élevées dans divers scénarios d'utilisation. Ses fonctionnalités se révèlent particulièrement bénéfiques pour le stockage de données structurées, les volumes de systèmes de fichiers de machines virtuelles et les charges de lecture et d'écriture élevées [14].

Voilà un tableau qui résume la comparaison entre le stockage objet et le bloc :

<b>Base de comparaison</b>	<b>Stockage d'objets</b>	<b>Stockage bloc</b>
<b>Unités de stockage</b>	<p>Le stockage d'objets est stocké sous forme de stockage personnalisable et volumineux de données principalement sous forme d'objets. Il stocke les fichiers et les systèmes de fichiers stockés sous forme de métadonnées utilisées aujourd'hui dans les plates formes de cloud computing.</p>	<p>Le stockage par blocs est stocké sous forme d'énormes morceaux de données sous la forme de blocs utilisés dans l'ancienne version de stockage de fichiers. Il n'a pas le concept de stockage de métadonnées.</p>
<b>Mises à jour des services</b>	<p>Les services de stockage d'objets ne disposent pas d'un support de mise à jour sur place qui pourrait produire des versions de plus en plus récentes d'objets, ce qui complexifierait l'architecture globale et nuirait à l'efficacité globale des performances.</p>	<p>Le stockage de verrouillage prend en charge les mises à jour sur place qui peuvent créer des versions de plus en plus récentes de blocs à des fins de stockage.</p>
<b>Gestion des performances</b>	<p>Le stockage objet fonctionne mieux avec une efficacité élevée pour les gros systèmes de fichiers et les grandes organisations où les données sont régulièrement requises et où les ressources sont collectées.</p>	<p>Le stockage par blocs a un niveau de performance optimisé pour le stockage des données sur le serveur de base de données et l'endroit où les transactions ont lieu pour stocker des blocs de données.</p>

<p><b>Flexibilité de stockage</b></p>	<p>Dans le stockage objet, les données sont accessibles et récupérées à partir de n'importe quelle région géographique, et non délimitée par une région. Par conséquent, les utilisateurs peuvent accéder au stockage n'importe où et n'importe où, quand ils le souhaitent.</p>	<p>Dans le stockage par blocs, le stockage de données et l'accès à une région éloignée de celle-ci peuvent augmenter la latence et les problèmes de stockage dans une base de données, ce qui peut entraîner une modification substantielle des performances globales.</p>
<p><b>Évolutivité dans le stockage de dataObject</b></p>	<p>Le stockage objet offre la meilleure fonctionnalité d'évolutivité car il peut stocker et faire évoluer un énorme volume de données depuis des pétaoctets jusqu'à l'infini sans aucun problème.</p>	<p>Étant donné que l'adresse constitue la partie information des blocs dans le stockage par blocs, l'évolutivité est un problème et présente une plus grande limitation par rapport au stockage d'objets</p>
<p><b>Le meilleur cas d'utilisation pour</b></p>	<p>Le stockage objet est mieux utilisé pour les données de fichiers statiques qui servent de stockage cloud</p>	<p>Les blocs de stockage par blocs sont stockés sous forme de données transactionnelles et de données changeantes qui sont fréquemment modifiées</p>
<p><b>Charge de travail</b></p>	<p>Le stockage d'objets peut stocker des métadonnées riches en fonctionnalités, ainsi que des analyses à grande échelle à stocker et à récupérer.</p>	<p>Le stockage par blocs a un accès limité aux données, mais en termes de sortie et de performances, il produit un résultat cohérent.</p>

<p><b>La plus grande force d'utilisation</b></p>	<p>Les services de stockage objet sont moins chers et cohérents, mais la dernière version objet ne contient pas de données immédiates ; la modification prévoit le transfert d'une nouvelle version de l'objet.</p>	<p>Le stockage par blocs est coûteux et fiable et les données peuvent être produites immédiatement.</p>
<p><b>Limites</b></p>	<p>Il ne fournit pas de protocole de partage et est inefficace pour les données transactionnelles.</p>	<p>Le stockage pose des difficultés lors de l'expansion de la région</p>

**Tableau III.1 : comparaison entre le stockage objet et le bloc [20].**

**III.5 Solutions open source**

L'open source désigne généralement un modèle de développement logiciel universel et non propriétaire. Dans un tel environnement, la collaboration est encouragée. Le public a accès gratuitement à l'ensemble des codes sources, documentations, kits de développement logiciel (SDK) et interfaces de programmation d'applications (API).

Les développeurs peuvent modifier et améliorer le code source, puis partager, distribuer ou publier ces efforts au sein de la communauté des développeurs. D'autres développeurs peuvent ensuite télécharger ce code ou le modifier davantage.

Les technologies open source offrent une flexibilité et un contrôle maximal sur vos options de gestion et de stockage de données. Grâce aux outils open source et à l'accès aux API ouvertes, vous pouvez personnaliser le code en fonction des besoins spécifiques de votre organisation. Vous n'êtes pas contraint par des technologies propriétaires au fur et à mesure de votre développement, mais vous avez la liberté d'utiliser le matériel existant que vous possédez peut-être (ou une combinaison de matériel de différents fournisseurs). Vous bénéficiez également des efforts d'autres développeurs au sein de la communauté.

En ce qui concerne les systèmes de stockage basés sur les objets, plusieurs solutions open source sont disponibles, telles que Ceph, MinIO, Openio.io et SwiftStack ou OpenStack Swift. Bien que ces solutions présentent des caractéristiques, des options politiques et des méthodologies différentes, elles poursuivent toutes le même objectif : permettre le stockage à grande échelle de données numériques non structurées [4].

### **III.6 Conclusion :**

Le stockage objet a transformé notre façon de gérer et de traiter les données. En utilisant une approche basée sur des objets plutôt que sur des blocs ou des fichiers traditionnels, il offre une gestion plus flexible et efficace des vastes volumes de données.

Grâce à des fonctionnalités telles que la scalabilité, la durabilité, la sécurité renforcée et la facilité d'accès, le stockage objet répond aux besoins croissants des entreprises et des organisations pour une gestion des données fiable et rentable. Son adoption croissante témoigne de son importance dans le paysage technologique actuel, façonnant ainsi l'avenir du stockage et de la gestion des données à l'ère numérique.

# Chapitre IV

## Conception et déploiement de la solution MINIO

## IV.1 Introduction

Les services de stockage en ligne comme Amazon S3 sont pratiques, mais peuvent s'avérer coûteux. Leur dépendance peut poser problème, surtout lors du développement d'une application. Heureusement, il existe des outils open source, tels que MinIO, qui est une solution de stockage d'objets offrant une API compatible avec Amazon Web Services S3 et supportant toutes les fonctionnalités principales de S3 (Simple Storage Service), MinIO est conçu pour être déployé partout : dans le cloud public ou privé, sur des infrastructures bare-metal, dans des environnements orchestrés, ainsi que sur des infrastructures de périphérie.

Dans ce chapitre on va développer le travail mis en œuvre dans cette étude et qui consiste à déployer la solution MinIO sur les plates-formes linux dans une configuration SNSD (Single-Node Single-Drive) permet un développement et une évaluation préliminaires.

Les déploiements SNSD utilisent un backend avec un codage d'effacement sans parité, ce qui ne garantit aucune fiabilité ou disponibilité supplémentaire au-delà de ce que fournit le volume de stockage sous-jacent, ces déploiements sont idéaux pour des tests et évaluations locaux, ou pour des charges de travail de données de petite échelle sans exigences de haute disponibilité.

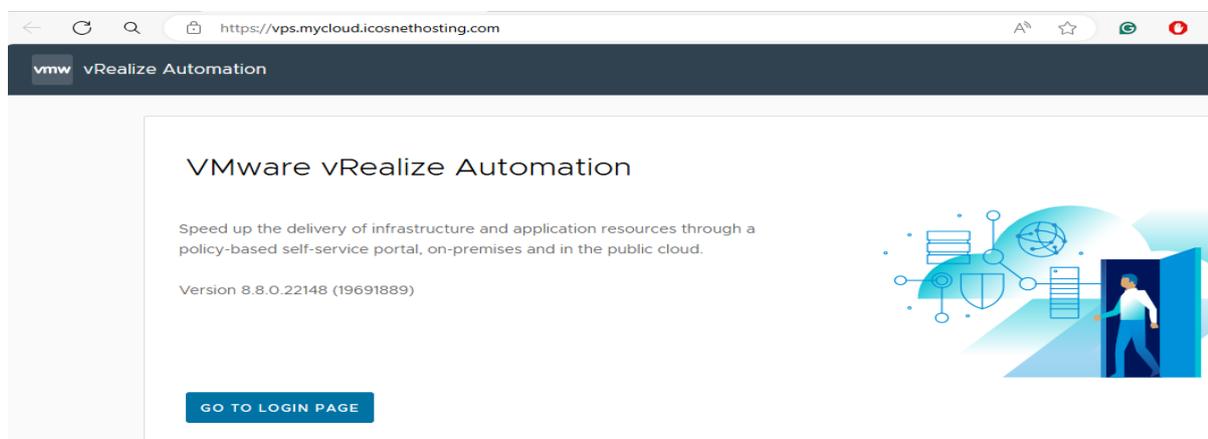
L'implémentation de la solution MinIO se fera dans un environnement virtuel basé sur l'application VMware.

## IV.2 Création de la machine virtuelle

Déploiement de la machine virtuelle dans un environnement VMware :

Suivant le lien d'approvisionnement : <https://vps.mycloud.icosnethosting.com/>.

Qui est le portail de déploiement, donc nous allons déployer notre machine sous linux.



**Figure IV.1 Login page.**

Cette figure nous montre le portail de déploiement des machines virtuelle de ICOSNET.

Après avoir insérer le nom d'utilisateur et le mot de passe on clique sur le service requête sur le VPS :

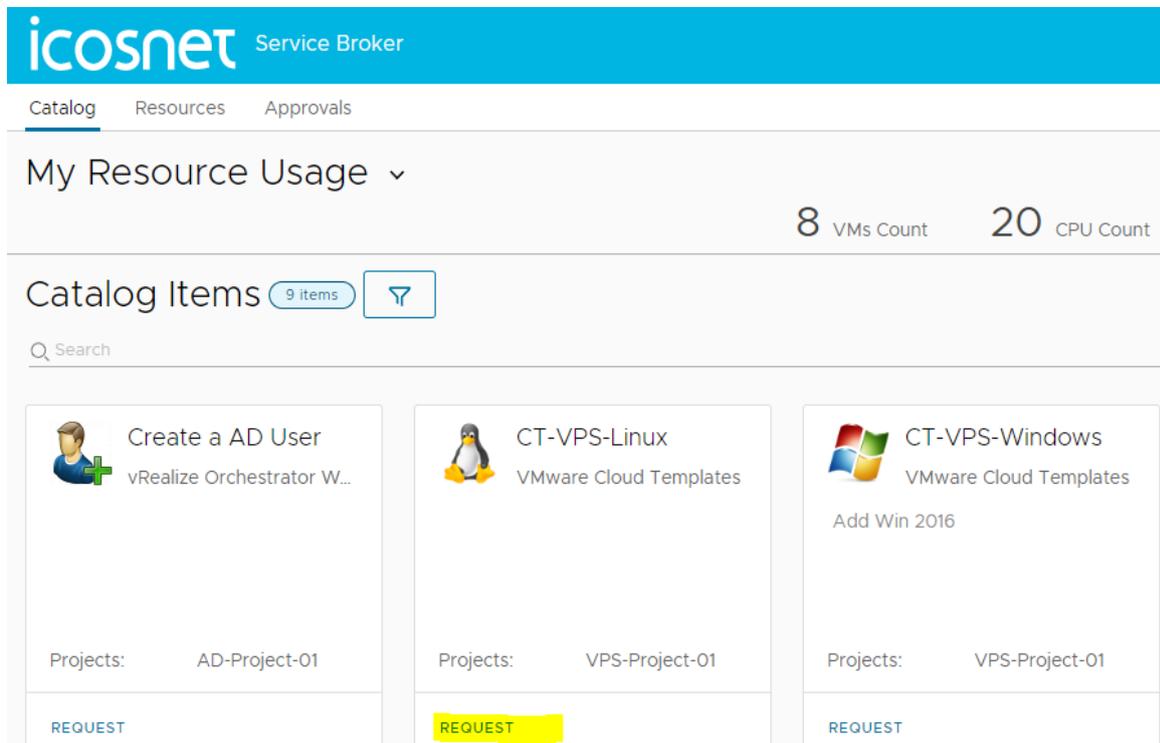
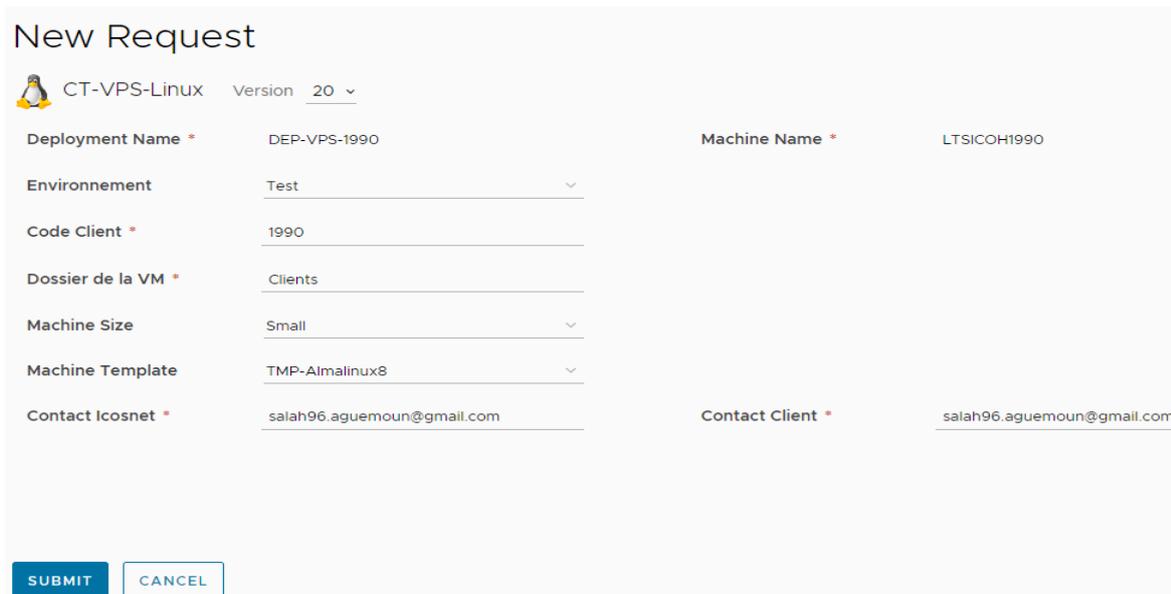


Figure IV.2 Service VPS-linux.

Cette figure nous montre les choix des ressources pour le **déploiement (linux ou bien windows)** A partir de formulaire comme montrer dans la figure ci-dessous est de saisir les informations nécessaires pour démarrer la machine :

- **Choix environnement** : Test.
- **Code client** : C'est numéro à attribuer à la machine par exemple 1999.
- **Dossier de la VM** : dans clients.
- **Machine size** : on va prendre taille S, cette machine sera provisionnée avec :
- **Disque 80 GB.**
- **RAM 4 GB.**
- **1 VCPU (Virtual CPU).**
- **Machine Template** : On va choisir la Template almalinux8.



New Request

CT-VPS-Linux Version 20

Deployment Name *	DEP-VPS-1990	Machine Name *	LTSICOH1990
Environnement	Test		
Code Client *	1990		
Dossier de la VM *	Clients		
Machine Size	Small		
Machine Template	TMP-Almalinux8		
Contact Icosnet *	salah96.aguemoun@gmail.com	Contact Client *	salah96.aguemoun@gmail.com

SUBMIT CANCEL

**Figure IV.3 Les caractéristiques de la machine virtuelle.**

Puis cliquer sur submit.

Une fois que la progression sera achevée, la machine virtuelle sera prête pour l'exploitation, elle aura donc la configuration suivante :

- Couple adresse IP privée / publique.
- Ses caractéristiques physiques : taille S (80GB disque, 4 GB RAM, 1VCPU).
- Almalinux comme système d'exploitation.
- Accessible via le protocole SSH port 22 en public avec des sources IP algérienne.

### IV.2.1 Accéder à cette machine via le protocole SSH :

Alors pour se faire, il faut installer un outil de terminal pour Windows qui sera un objet d'un client SSH, dans notre cas de figure ça sera **Mobaxterm**, ci-dessous le lien de téléchargement.

<https://mobaxterm.mobatek.net/download.html>

**Afin de se connecter à notre machine via cet outil avec le protocole SSH, il faut s'assurer tout d'abord que :**

La machine virtuelle est publiée dans l'internet avec une adresse IP publique et fixe.

Le service SSH est en marche dans cette machine et avoir un numéro de port en cas de changement pour des raisons de sécurité.

Un mot de passe et un utilisateur pour l'authentification.

Aller dans terminal --- sessions --- ssh :

- **Remote host** : c'est l'adresse IP publique de notre machine **197.140.18.88**
- **User name** : root.
- **Port** : 22

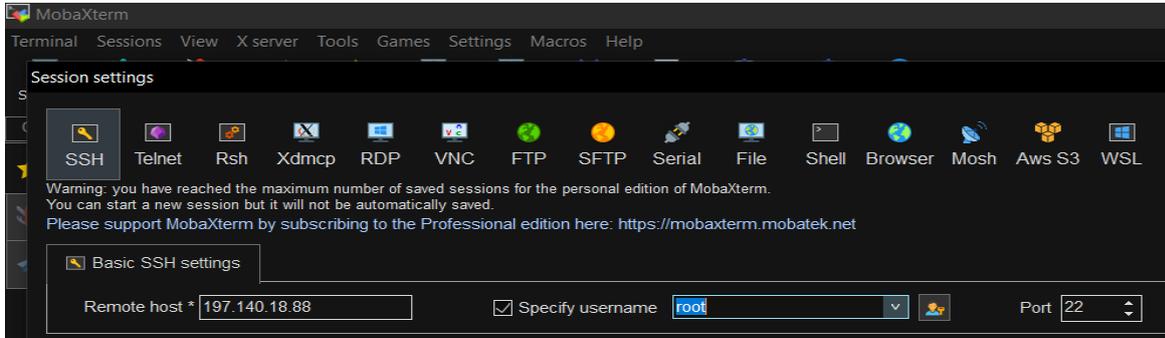


Figure IV.4 Protocole SSH.

Cette figure nous montre comment accéder au protocole SSH

Puis saisir le mot de passe root :

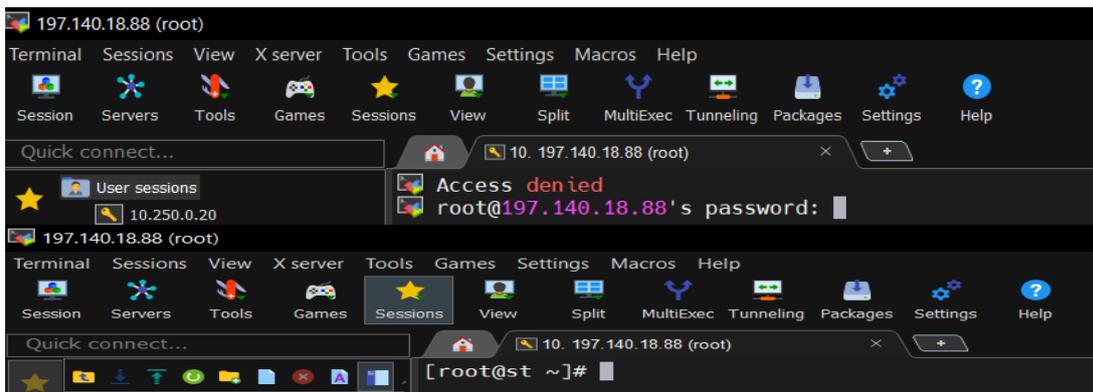


Figure IV.5 L'accéder au root avec succès.

Cette figure nous montre comment accéder à notre machine virtuelle

## IV.3 Configuration préliminaire un serveur linux :

### IV.3.1 Caractéristique de la machine :

```
[root@localhost /]# hostnamectl
Static hostname: localhost.localdomain
Icon name: computer-vm
Chassis: vm
Machine ID: 576612e0c05f48dcbc69cdcaf8734218
Boot ID: 324568d83bc143d686a3cd1ce3b66f00
Virtualization: vmware
Operating System: AlmaLinux 8.8 (Sapphire Caracal)
CPE OS Name: cpe:/o:almalinux:almalinux:8::baseos
Kernel: Linux 4.18.0-477.27.2.el8_8.x86_64
Architecture: x86-64
```

Figure IV.6 Les caractéristique de la machine.

La commande « hostnamectl » comme le montre la figure ci-dessus permet de gérer le nom d'hôte et afficher les caractéristiques de votre machine.

L'accès de la machine se fait via **SSH** (le port **22**) vers l'adresse publique **197.140.18.88**

### IV.3.2 Changements de mot de passe :

Changement de mot de passe root et mettre un autre encore plus compliqué par mesure de sécurité à l'aide de la commande **passwd** (cette commande sert à modifier le mot de passe d'un utilisateur), comme montre la figure ci-dessous

```
[root@localhost ~]# passwd
Changing password for user root.
New password:
Retype new password:
passwd: all authentication tokens updated successfully.
```

Figure IV.7 Changement de mot de passe avec succès.

Vous pouvez utiliser le générateur de mot de passe aléatoire en ligne :

<https://www.lastpass.com/fr/features/password-generator>

### IV.3.3 Changement de nom de la machine :

On va opter pour : **st.icosnethosting**

```
[root@st ~]# hostname st.icosnethosting.com
[st.icosnethosting ~]# hostname
st.icosnethosting.com
```

Figure IV.8 Changement de nom de la machine.

Cette figure montre comment nous avons changer le nom de notre machine.

### IV.3.4 Editer le fichier /etc/hostname avec l'utilitaire vi

```
host.com hostname hosts
[root@localhost ~]# vi /etc/hostname
[st.icosnethosting ~]# cat /etc/hostname
st.icosnethosting.com
```

Figure IV.9 Le fichier /etc/hostname.

Dans cette figure l'éditeur de texte « **vi** » permet de modifier les fichiers de configuration de divers programmes et fonctionnalités. Il fonctionne en mode texte.

La commande (**cat**) c'est pour afficher le contenu d'un fichier en sortie standard.

Après on redémarre la machine pour prendre effet comme montre la figure ci-dessous

```
[root@localhost ~]# reboot
```

Figure IV.10 Redémarrer la machine.

## IV.4 Déploiement du serveur MinIO :

### IV.4.1 Téléchargement de MinIO server :

Avant de télécharger MinIO server il faut assurer que le OS est mis à jour

```
[root@st /]# dnf update && upgrade
Last metadata expiration check: 2:45:02 ago on Mon 13 May 2024 03:44:34 AM EDT.
Dependencies resolved.
```

```
Transaction Summary
=====
Install      7 Packages
Upgrade    288 Packages

Total download size: 638 M
Is this ok [y/N]: █
```

Figure IV.11 Mis à jour du système d'exploitation.

Cette figure montre comment faire le mis à jour dans un système linux.

Dans la figure ci-dessous nous avons fait un ping vers google.com pour vérifier que la connexion internet + résolution dns fonctionnent sans aucun problème :

```
[root@st /]# ping google.com
PING google.com (142.251.37.206) 56(84) bytes of data.
64 bytes from mrs09s15-in-f14.1e100.net (142.251.37.206): icmp_seq=1 ttl=115 time=20.6 ms
64 bytes from mrs09s15-in-f14.1e100.net (142.251.37.206): icmp_seq=2 ttl=115 time=20.2 ms
64 bytes from mrs09s15-in-f14.1e100.net (142.251.37.206): icmp_seq=3 ttl=115 time=20.4 ms
64 bytes from mrs09s15-in-f14.1e100.net (142.251.37.206): icmp_seq=4 ttl=115 time=20.6 ms
^C
--- google.com ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3004ms
rtt min/avg/max/mdev = 20.238/20.454/20.601/0.201 ms
```

Figure IV.12 Ping google.com.

```
[root@st /]# hostnamectl
Static hostname: st.icosnethosting.com
Icon name: computer-vm
Chassis: vm
Machine ID: 576612e0c05f48dcbc69cdcaf8734218
Boot ID: 406740cdddcc4ef397b09f6468b19aa9
Virtualization: vmware
Operating System: AlmaLinux 8.9 (Midnight Oncilla)
CPE OS Name: cpe:/o:almalinux:almalinux:8::baseos
Kernel: Linux 4.18.0-513.24.1.el8_9.x86_64
Architecture: x86-64
```

Figure IV.13 Les nouvelles caractéristiques de la machine.

Dan cette figure on peut visualiser les nouvelles caractéristiques de la machine

Utilisez les commandes suivantes pour télécharger le dernier RPM MinIO stable et l'installer.

Nous utiliserons la commande « wget » (est un utilitaire de ligne de commande gratuit utilisé pour télécharger des fichiers depuis Internet), comme montre la figure ci-dessous

wget [https://dl.min.io/server/minio/release/linux-amd64/archive/minio-20240510014138.0.0-1.x86\\_64.rpm](https://dl.min.io/server/minio/release/linux-amd64/archive/minio-20240510014138.0.0-1.x86_64.rpm) -O minio.rpm

```
[root@st /]# wget https://dl.min.io/server/minio/release/linux-amd64/archive/minio-20240510014138.0.0-1.x86_64.rpm -O minio.rpm
--2024-05-13 06:46:15-- https://dl.min.io/server/minio/release/linux-amd64/archive/minio-20240510014138.0.0-1.x86_64.rpm
Resolving dl.min.io (dl.min.io)... 178.128.69.202, 138.68.11.125
Connecting to dl.min.io (dl.min.io)|178.128.69.202|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 38023685 (36M) [application/x-rpm]
Saving to: 'minio.rpm'

minio.rpm          1%[>] 664.00K 133KB/s  eta 4m 35s

minio.rpm          100%[=====] 36.26M 1.45MB/s  in 86s

2024-05-13 06:47:41 (433 KB/s) - 'minio.rpm' saved [38023685/38023685]
```

Figure IV.14 Téléchargement de MinIO server.

Dans la figure ci-dessous nous pouvons vérifier que le fichier minio.rpm est présent dans le système :

```
[root@st /]# ls | grep minio
minio.rpm
```

Figure IV.15 Le fichier minio.rpm est présent dans le système.

Minio est bien présent dans le système mais il n'est pas exécutable (il est en rouge) alors en va l'exécuter avec la commande (dnf install minio.rpm), comme montre la figure ci-dessous

```
[root@st /]# dnf install minio.rpm
Last metadata expiration check: 2:56:22 ago on Mon 13 May 2024 07:33:59 AM EDT.
Dependencies resolved.

=====
Package                Architecture      Version           Repository        Size
=====
Installing:
minio                   x86_64           20240510014138.0.0-1  @commandline    36 M
```

Figure IV.16 L'exécution de MinIO server.

## IV.4.2 Création du fichier systemd pour le service minio.service dans le système :

### IV.4.2.1 Systemd (system daemon) :

Le démon du système est un ensemble de programmes destiné à la gestion système, conçu pour le noyau Linux. Il permet le chargement en parallèle des services au démarrage.

Les packages .rpm installent le fichier de service **systemd** suivant dans le répertoire : **/usr/lib/systemd/system/minio.service**, comme montre la figure ci-dessous

Pour les installations binaires, créez ce fichier manuellement sur tous les hôtes MinIO.

Nous pouvons vérifier cela en exploitant le chemin **/usr/lib/systemd/system/minio.service**

```
[root@st system]# cat minio.service
[Unit]
Description=MinIO
Documentation=https://docs.min.io
Wants=network-online.target
After=network-online.target
AssertFileIsExecutable=/usr/local/bin/minio
```

Figure IV.17 Systemd.

Le fichier minio.service s'exécute par défaut en tant qu'utilisateur et groupe minio-user.

Vous pouvez créer l'utilisateur et le groupe à l'aide des commandes groupadd et useradd. Dans la figure suivante nous avons créé l'utilisateur, le groupe et définit les autorisations pour accéder aux chemins de dossiers destinés à être utilisés par MinIO, ces commandes nécessitent généralement des autorisations root (sudo).

```
[root@st /]# groupadd -r minio-user
[root@st /]# useradd -M -r -g minio-user minio-user
[root@st /]# chown minio-user:minio-user /dev/mapper/almalinux-root
```

Figure IV.18 L'utilisateur et le groupe.

Dans notre cas nous exploitant le système de fichier `/dev/mapper/almalinux-root` qui sera utilisé par MinIO

### IV.4.3 Créer le fichier de variables d'environnement

Créez un fichier de variables d'environnement dans `/etc/default/minio`. Le conteneur MinIO Server peut utiliser ce fichier comme source de toutes les variables d'environnement.

Mais tout d'abord on doit créer un dossier `/data/minio` et déclarer l'utilisateur + groupe minio-user comme propriétaire de ce dossier comme montre la figure ci-dessous :

```
[root@st /]# mkdir -p /data/minio
[root@st /]# chown -R minio-user:minio-user /data/minio
```

Figure IV.19 Créer un dossier `/data/minio`.

Puis coller la configuration au niveau `/etc/default/minio` comme montre la figure ci-dessous :

```
[root@st /]# vi etc/default/minio
# MINIO_ROOT_USER and MINIO_ROOT_PASSWORD sets the root account for the MinIO server.
# This user has unrestricted permissions to perform S3 and administrative API operations on any resource in the deployment.
# Omit to use the default values 'minioadmin:minioadmin'.
# MinIO recommends setting non-default values as a best practice, regardless of environment

MINIO_ROOT_USER=myminioadmin
MINIO_ROOT_PASSWORD=minio-secret-key-change-me

# MINIO_VOLUMES sets the storage volume or path to use for the MinIO server.
MINIO_VOLUMES="/data/minio"

# MINIO_OPTS sets any additional commandline options to pass to the MinIO server.
# For example, '--console-address :9001' sets the MinIO Console listen port
MINIO_OPTS="--console-address :9001"

# MINIO_SERVER_URL sets the hostname of the local machine for use with the MinIO Server
# MinIO assumes your network control plane can correctly resolve this hostname to the local machine

# Uncomment the following line and replace the value with the correct hostname for the local machine and port for the MinIO server (9000 by default).
#MINIO_SERVER_URL="http://minio.example.net:9000"
~
```

Figure IV.20 Fichier de variables d'environnement.

## Vérification de service MinIO :

```
[root@st /]# systemctl restart minio

[root@st ~]# systemctl status mi
microcode.service minio.service
[root@st ~]# systemctl status minio.service
● minio.service - MinIO
   Loaded: loaded (/usr/lib/systemd/system/minio.service; enabled; vendor preset: disabled)
   Active: active (running) since Mon 2024-06-10 11:29:18 EDT; 2 weeks 0 days ago
     Docs: https://docs.min.io
  Process: 36999 ExecStartPre=/bin/bash -c if [ -z "${MINIO_VOLUMES}" ]; then echo "Variable MINIO_VOLUMES not set in /etc/default/minio"
 Main PID: 37001 (minio)
    Tasks: 10
   CGroup: /system.slice/minio.service
           └─37001 /usr/local/bin/minio server --certs-dir /home/minio-user/.minio/certs/CAs/CAs --console-address icosnethosting.com:9000

Jun 10 11:29:18 st.icosnethosting.com minio[37001]: License: GNU AGPLv3 - https://www.gnu.org/licenses/agpl-3.0.html
Jun 10 11:29:18 st.icosnethosting.com minio[37001]: Version: RELEASE.2024-05-10T01-41-38Z (go1.22.3 linux/amd64)
```

Figure IV.21 MinIO en mode active.

Dans cette figure nous avons lancer un service au démarrage, et pour cela on a utilisé la commande enable et on voit bien que le service **MinIO** et en mode active.

**Nous pouvons même voir que les ports MinIO sont à l'écoute en interne à l'aide de la commande « netstat » comme montre la figure ci-dessous :**

```
[root@st /]# netstat -tulpn
Active Internet connections (only servers)
Proto Recv-Q Send-Q Local Address           Foreign Address         State       PID/Program name
tcp        0      0 127.0.0.1:9000          0.0.0.0:*               LISTEN      88817/minio
tcp        0      0 0.0.0.0:22            0.0.0.0:*               LISTEN      26799/sshd
tcp6       0      0 :::9000                :::*                    LISTEN      88817/minio
tcp6       0      0 :::1:9000              :::*                    LISTEN      88817/minio
tcp6       0      0 :::9001                :::*                    LISTEN      88817/minio
tcp6       0      0 :::22                  :::*                    LISTEN      26799/sshd
udp        0      0 127.0.0.1:323         0.0.0.0:*               LISTEN      895/chronyd
udp6       0      0 :::1:323               :::*                    LISTEN      895/chronyd
```

Figure IV.22 Les ports MinIO sont à l'écoute en interne.

Nous avons noté que malgré que le port 9000 / 9001 sont ouverts au niveau système, le navigateur montre que la connexion est impossible comme montre la figure ci-dessous :

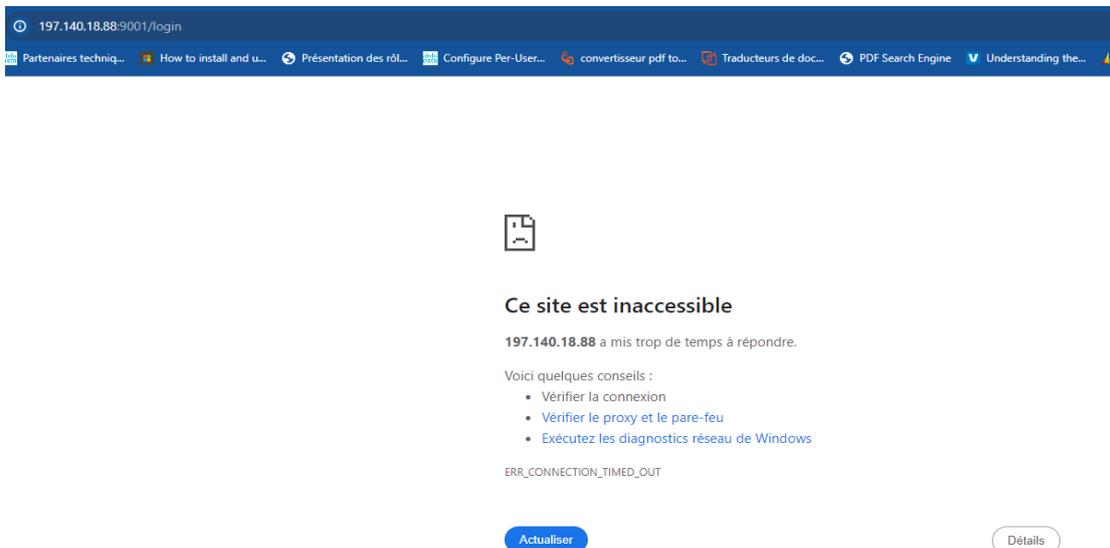


Figure IV.23 Le site est inaccessible.

## Chapitre IV conception et déploiement de la solution MINIO

Pour cela nous devons donc autoriser les ports en interne, cela veut dire créer des autorisations au niveau de firewalld système puis on redémarre ce dernier et vérifier que les ports sont bien ouverts.

```
[root@st /]# firewall-cmd --permanent --add-port=9000-9001/tcp
success

[root@st /]# firewall-cmd --reload
success

[root@st /]# firewall-cmd --list-ports
9000-9001/tcp
```

Figure IV.24 Autorisations au niveau de firewalld.

Dans cette figure nous avons ajouté les port 9000-9001 au niveau de firewall.

Si on lance maintenant le test, la console MINIO est maintenant accessible via le net <http://197.140.18.88:9001/login> comme montre la figure ci-dessous :

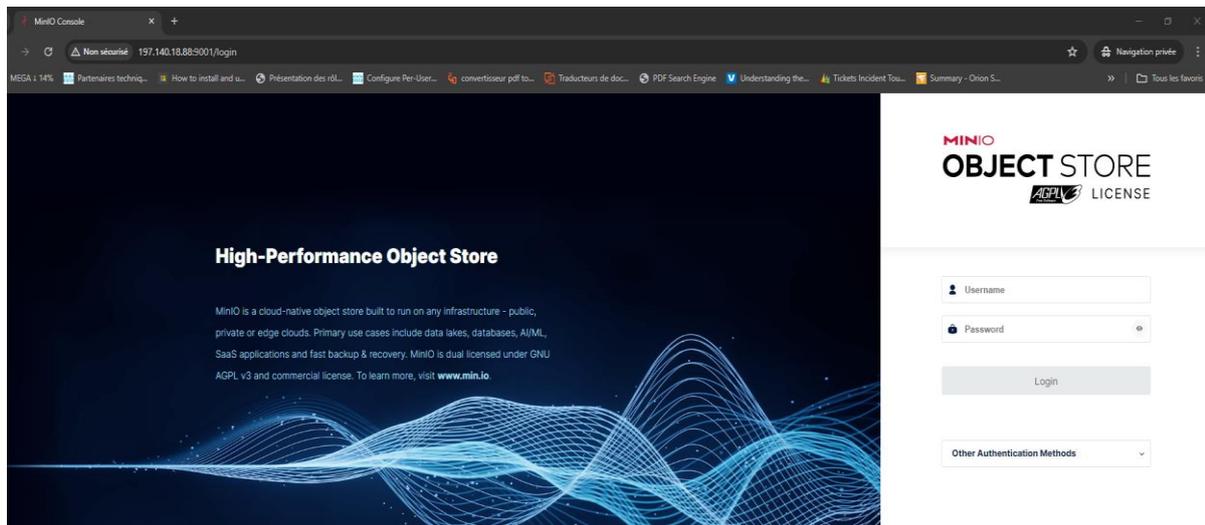


Figure IV.25 La console MinIO est accessible via le net.

### IV.4.4 Pointage DNS :

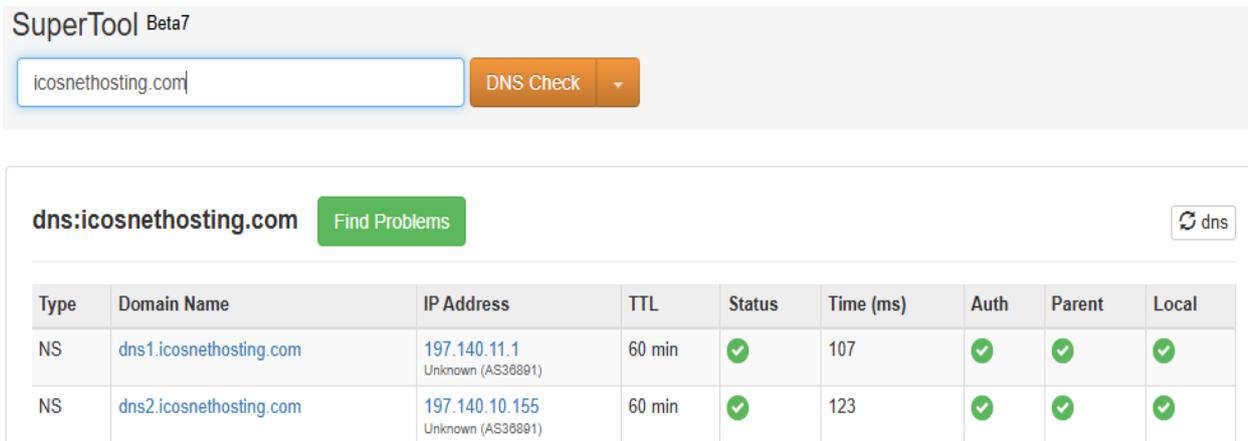
Nous avons déclaré un nom pour notre application : **st.icosnethosting.com**

Or que nous devons donc pointer l'adresse publique du serveur **197.140.18.88** vers le nom FQDN **st.icosnethosting.com**

Le pointage sera fait au niveau de serveur DNS qui gère le domaine **icosnethosting.com**

Nous pouvons visualiser le serveur DNS qui le gère via le lien suivant :

<https://mxtoolbox.com/SuperTool.aspx>



SuperTool Beta7

icosnethosting.com DNS Check

dns:icosnethosting.com Find Problems dns

Type	Domain Name	IP Address	TTL	Status	Time (ms)	Auth	Parent	Local
NS	dns1.icosnethosting.com	197.140.11.1 Unknown (AS36891)	60 min	✓	107	✓	✓	✓
NS	dns2.icosnethosting.com	197.140.10.155 Unknown (AS36891)	60 min	✓	123	✓	✓	✓

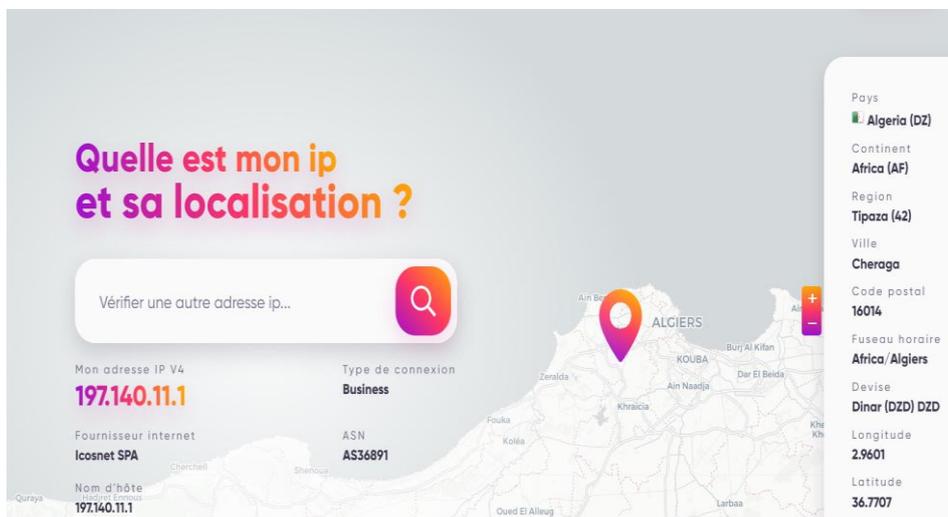
**Figure IV.26** Serveur DNS qui gère le domaine icosnethosting.com.

Dans cette figure nous pouvons visualiser que les deux serveurs DNS publics sont :

- dns1.icosnethosting.com dont l'IP **197.140.11.1**
- dns2.icosnethosting.com dont l'IP **197.140.10.155**

Ces DNS sont hébergés au niveau de cloud Icosnet en Algérie.

Vous pouvez voir la localisation par le biais des outils en ligne : <https://ipcost.com/fr>



**Figure IV.27** Localisation DNS.

Cette figure montre comment voir la localisation des serveurs DNS.

Nous accédant donc au serveur **197.140.11.1** via **RDP** et pointer l'adresse serveur **197.140.18.88** vers **st.icosnethosting.com**, comme montre la figure ci-dessous :

New Host

Name (uses parent domain name if blank):  
st

Fully qualified domain name (FQDN):  
st.icosnethosting.com.

IP address:  
197.140.18.88

Create associated pointer (PTR) record

Add Host Cancel

Figure IV.28 Pointage DNS.

Vu les informations confidentielles nous n'avons pas pu afficher les records au niveau de ce serveur DNS, vérification de propagation DNS dans la figure ci-dessous :

<https://www.whatsmydns.net/#A/st.icosnethosting.com>

Location	IP Address	Status
Canoga Park CA, United States Sprint	197.140.18.88	✓
Dallas TX, United States Speakeasy	197.140.18.88	✓
Dothan AL, United States Comodo	197.140.18.88	✓
Reston VA, United States Sprint	197.140.18.88	✓
Providence RI, United States Verizon	197.140.18.88	✓

Figure IV.29 Propagation DNS

Dans cette figure ci-dessous montre que nous pouvons maintenant accéder via le navigateur avec le lien FQDN au lieu d'adresse IP :

<http://st.icosnethosting.com:9001/login>

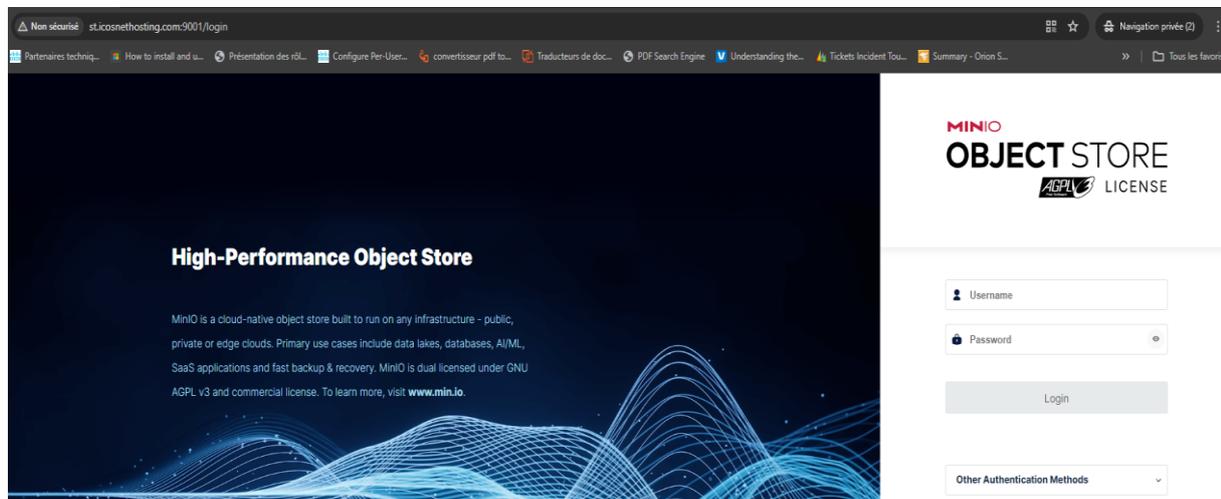


Figure IV.30 Minio consol avec le lien FQDN

## IV.4.4.1 Authentification :

En ce qui concerne l'authentification, les identifiants sont spécifiés dans le fichier `/etc/default/minio`.

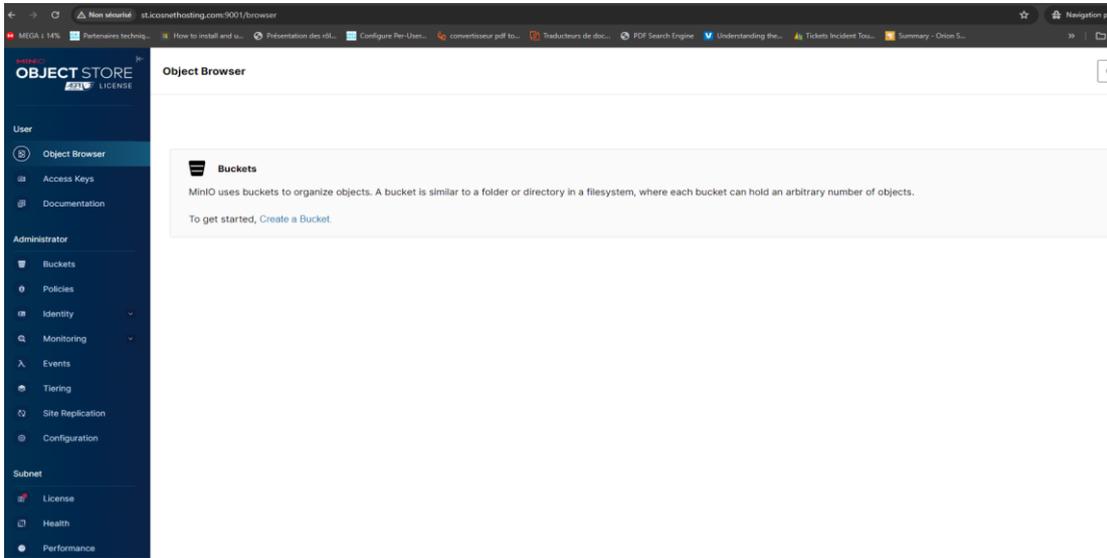


Figure IV.31 Minio console après l'authentification.

Cette figure montre notre console Minio après l'authentification au niveau de fichier de variable d'environnement.

## IV.4.5 Installation certificat SSL :

Nous exploitons le certificat SSL pour le domaine `*.icosnethosting.com`

« \*. » : désigne le wildcard, le certificat couvrira le domaine `icosnethosting.com` et les premiers sous-domaines.

Les fichiers d'installation de certificat SSL « `*.icosnethosting` » sont issues par l'autorité de certificat (CA : certificate authority) après une procédure de réservation de ce dernier via le portail de registrar `icosnet`.

Le certificat Wild card peut être installé dans plusieurs serveurs.

Dans notre cas nous allons l'installer pour le FQDN : `st.icosnethosting.com`.

Tout d'abord on va créer un dossier `minio-user` dans lequel on va créer le chemin d'installation de notre certificat, comme montré dans la figure ci-dessous :

```
[root@st home]# mkdir minio-user
[root@st home]# ls
minio-user
[root@st home]# ll
total 0
drwxr-xr-x. 2 root root 6 May 20 09:46 minio-user
[root@st home]#
[root@st home]# chown -R minio-user:minio-user minio-user/
[root@st home]# ll
total 0
drwxr-xr-x. 2 minio-user minio-user 6 May 20 09:46 minio-user
```

Figure IV.32 Créer un dossier minio-user.

Nous pouvons lister par la suite que le minio-user est le propriétaire des fichiers + dossier avec les droits adéquats :

```
[root@st CAs]# ll
total 16
drwx-----. 2 minio-user minio-user   6 May 20 16:49 CAs
-rwxr--r--. 1 minio-user minio-user 3272 May 20 14:20 private.key
-rwxr--r--. 1 minio-user minio-user 8234 May 20 14:19 public.crt
[root@st CAs]# pwd
/home/minio-user/.minio/certs/CAs/CAs
```

Figure IV.33 Le propriétaire des fichiers + dossier.

Dans cette figure nous avons utilisé la commande « ll » pour lister le minio-user.

Spécifier le chemin de notre certificat SSL dans le fichier de configuration de l'application MINIO dans la variable MINIO-OPTS comme montrer dans la figure ci-dessous :

```
[root@st CAs]# cat /etc/default/minio
# MINIO_VOLUMES sets the storage volume or path to use for the MinIO server.
MINIO_VOLUMES="/data/minio"

# MINIO_OPTS sets any additional commandline options to pass to the MinIO server.
# For example, `--console-address :9001` sets the MinIO Console listen port
MINIO_TLS="on"
MINIO_OPTS="--certs-dir /home/minio-user/.minio/certs/CAs/CAs --console-address icosnethosting.com:9001"
```

Figure IV.34 Le chemin de la certificat SSL.

Vous pouvez vérifier que le certificat est bel et bien installé via le lien qui est accessible via internet : <https://st.icosnethosting.com:9001/login> comme montrer dans la figure ci-dessous :

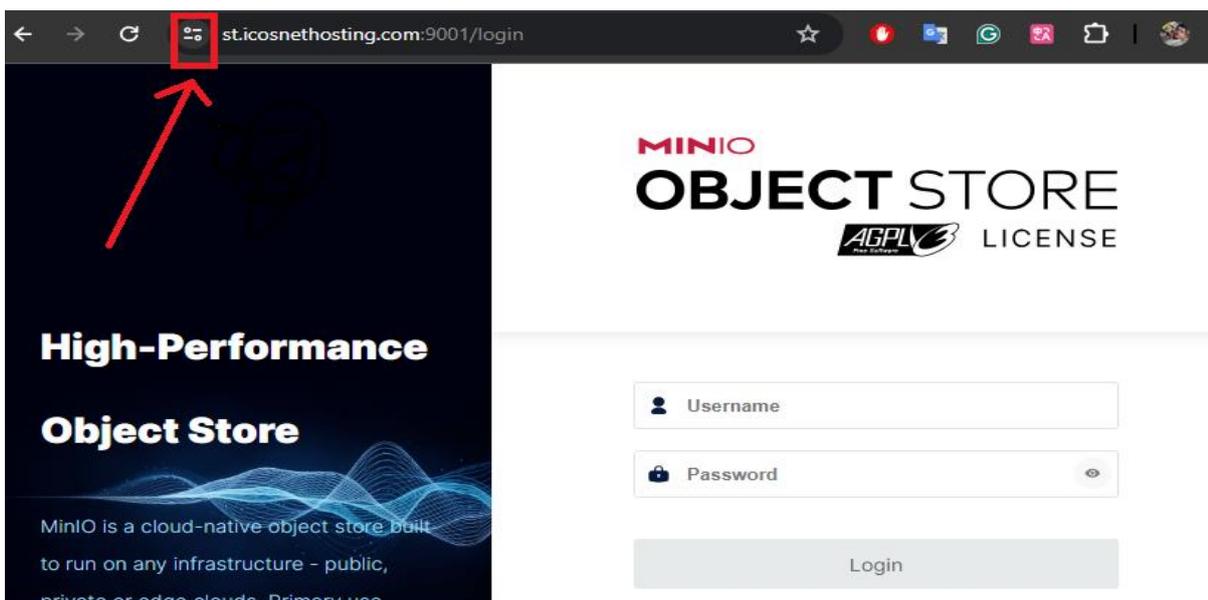


Figure IV.35 Le certificat est bien installé.

## IV.5 Création des buckets :

Les buckets ou les compartiments sont des conteneurs où les objets sont stockés, qui peuvent être des fichiers ou des données. Chaque objet dans un bucket est identifié par une clé unique, agissant comme son adresse, facilitant l'accès et la gestion des données stockées.

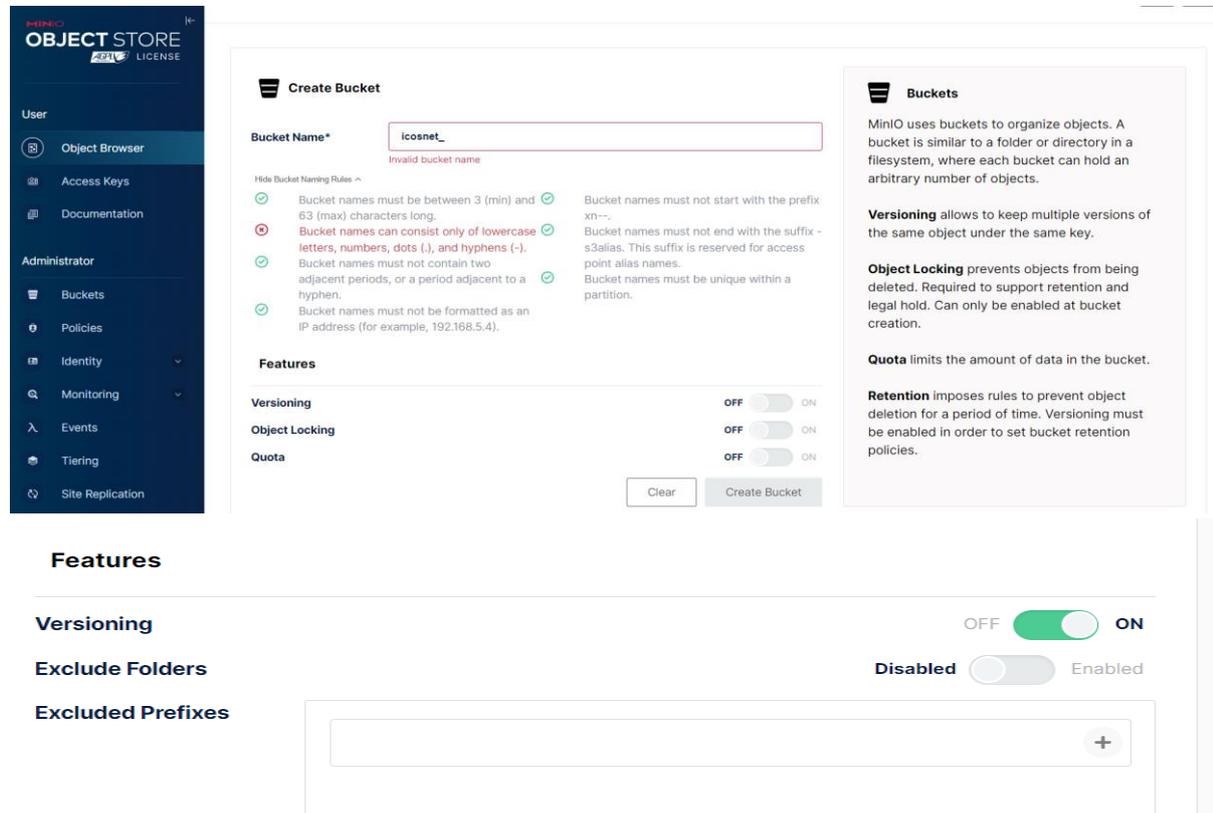


Figure IV.36 Création des buckets.

Cette figure montre comment nous avons créé une bucket dans la minio console.

### IV.5.1 Gestion des versions du compartiment

MinIO prend en charge l'enregistrement de plusieurs « versions » d'un objet dans un seul compartiment.

Lorsqu'il est activé, le contrôle de version permet à MinIO de conserver plusieurs itérations du même objet. Les opérations d'écriture qui écraseraient normalement les objets existants entraîneront la création de nouveaux objets versionnés. La gestion des versions MinIO empêche les écrasements et les suppressions accidentelles tout en permettant les opérations d'écriture « annuler ». La gestion des versions du bucket est une condition préalable à la configuration des règles de verrouillage et de conservation des objets.

Pour un compartiment versionné, une opération d'écriture qui modifie un objet génère une nouvelle version de l'objet avec un ID de version unique. MinIO marque la dernière version

de l'objet que les clients récupèrent par défaut. Les clients peuvent alors explicitement choisir de répertorier, récupérer ou supprimer une version d'objet spécifique.

MinIO limite la gestion des versions d'objet à 10 000 versions maximum de chaque objet.

Si une opération d'écriture dépasse la limite de 10 000 versions d'objet, MinIO bloque l'opération et renvoie une erreur.

MinIO ne crée pas de versions pour la création, la mutation ou la suppression d'objets de répertoire explicites (« préfixes »).

Les objets créés dans cet objet de répertoire explicite conservent un comportement de version normal.

### **IV.5.2 Gestion des versions et capacité de stockage**

MinIO n'effectue pas de versionnage de type incrémentiel ou différentiel. Pour les charges de travail impliquant de nombreuses mutations, cela peut entraîner une utilisation importante du lecteur par des versions d'objet plus anciennes ou obsolètes.

Par exemple, considérons un objet de 1 Go contenant des données de journal. Une application ajoute 100 Mo de données au journal et les télécharge sur MinIO. Ce dernier contiendrait alors à la fois les versions 1 Go et 1,1 Go de l'objet. Si l'application répétait ce processus chaque jour pendant 10 jours, le compartiment contiendrait finalement plus de 14 Go de données associées à un seul objet.

MinIO prend en charge la configuration des règles de gestion du cycle de vie des objets pour expirer automatiquement ou faire la transition des versions d'objets obsolètes et libérer de la capacité de stockage.

Par exemple, vous pouvez configurer une règle pour faire expirer automatiquement les versions d'objet 90 jours après qu'elles soient devenues obsolètes (c'est-à-dire qu'elles ne soient plus la « dernière » version de cet objet).

Vous pouvez également effectuer une suppression manuelle des versions d'objet à l'aide des commandes suivantes :

- `mc rm --versions` - Removes all versions of an object.
- `mc rm --versions --older-than`

## IV.5.3 Tester une bucket sur le serveur MINIO :

La figure ci-dessous montre l'ensemble de tous les buckets que nous avons créé :

Nom	Objets	Taille	Accéder
icosnet-26-05-2024	72	83,8 Mo	R/É
minio1-icosnetbucket	1	329,6 Ko	R/É
test	0	0,0 B	R/É
vbrsuite	4 083	2,3 Gio	R/É
sauvegarde vbrsuite	0	0,0 B	R/É
vbrsuite-sauvegarde3	0	0,0 B	R/É
vbrsuite-sauvegarde4	3	0,0 B	R/É
vbrsuite-sauvegarde6	0	0,0 B	R/É
vbrsuite-sauvegarde9	0	0,0 B	R/É

Figure IV.37 L'ensemble de tous les buckets.

On va opter vers la bucket « minio1-icosnetbucket » et faire un téléchargement d'un fichier quelconque comme montrer dans la figure ci-dessous :

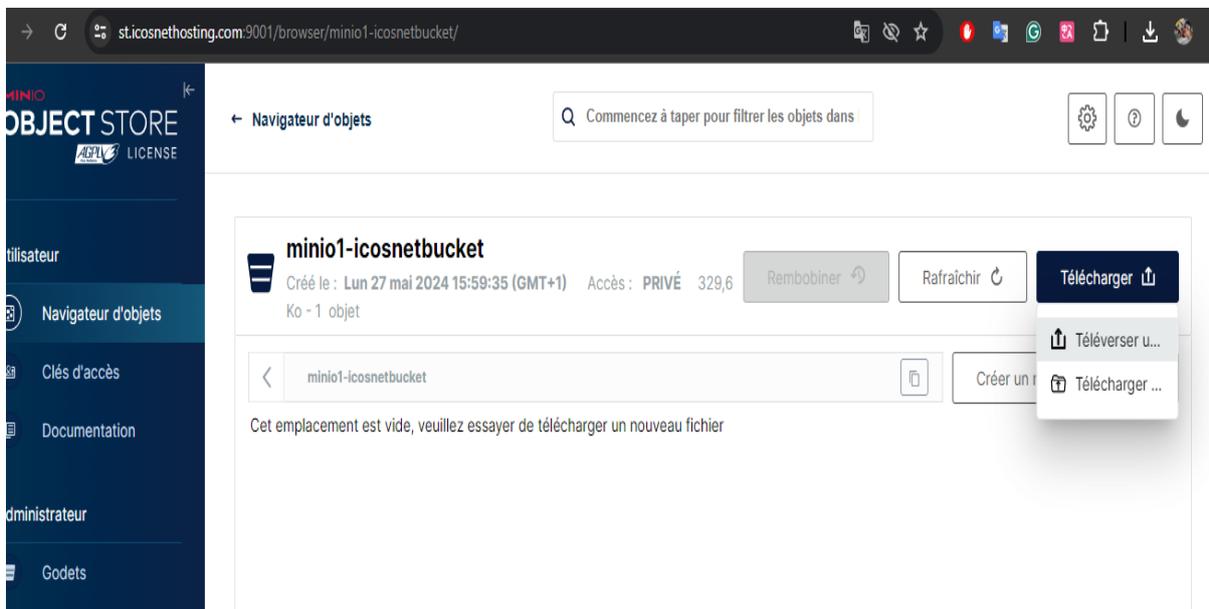


Figure IV.38 Téléchargement d'un fichier.

On voit bien que le fichier a été bien stocké dans la bucket « minio1-icosnetbucket » comme le montre la figure ci-dessous :

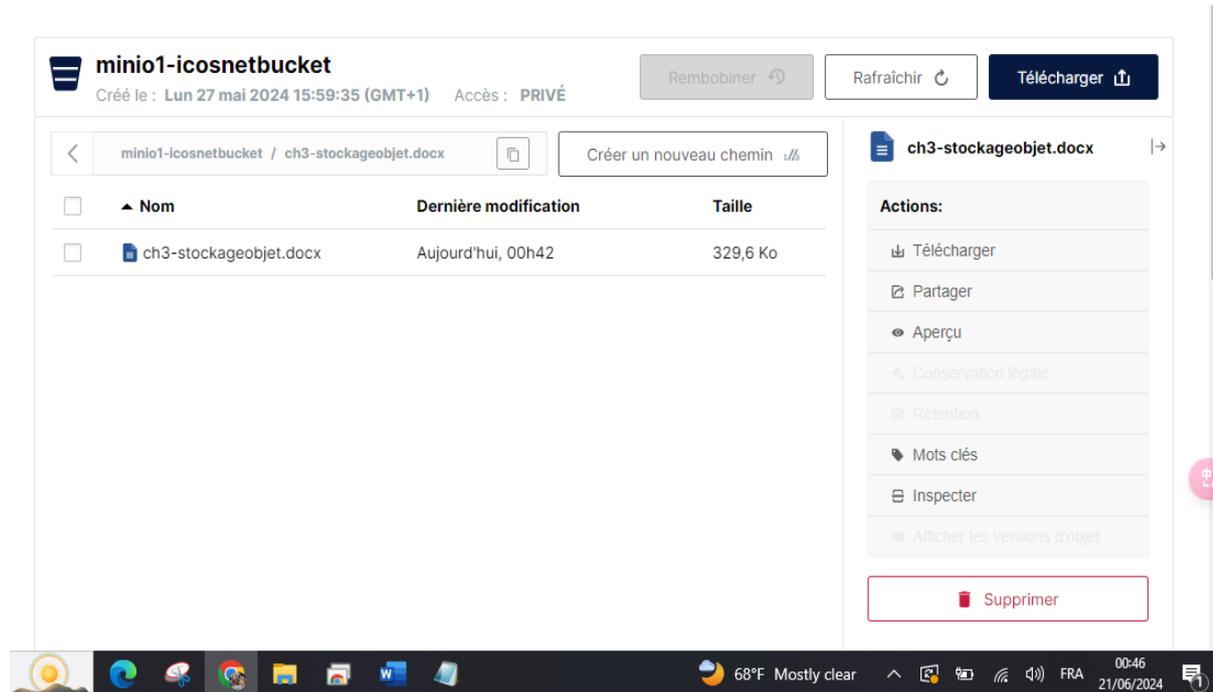


Figure IV.39 Le fichier (ch3) est bien stocké dans la bucket.

Maintenant nous pouvons voir le plan d'action qui permet de télécharger, de partager, aperçu ou supprimer le fichier, nous avons choisi de tester le partage de ce fichier via le code URL comme montrer dans la figure ci-dessous :

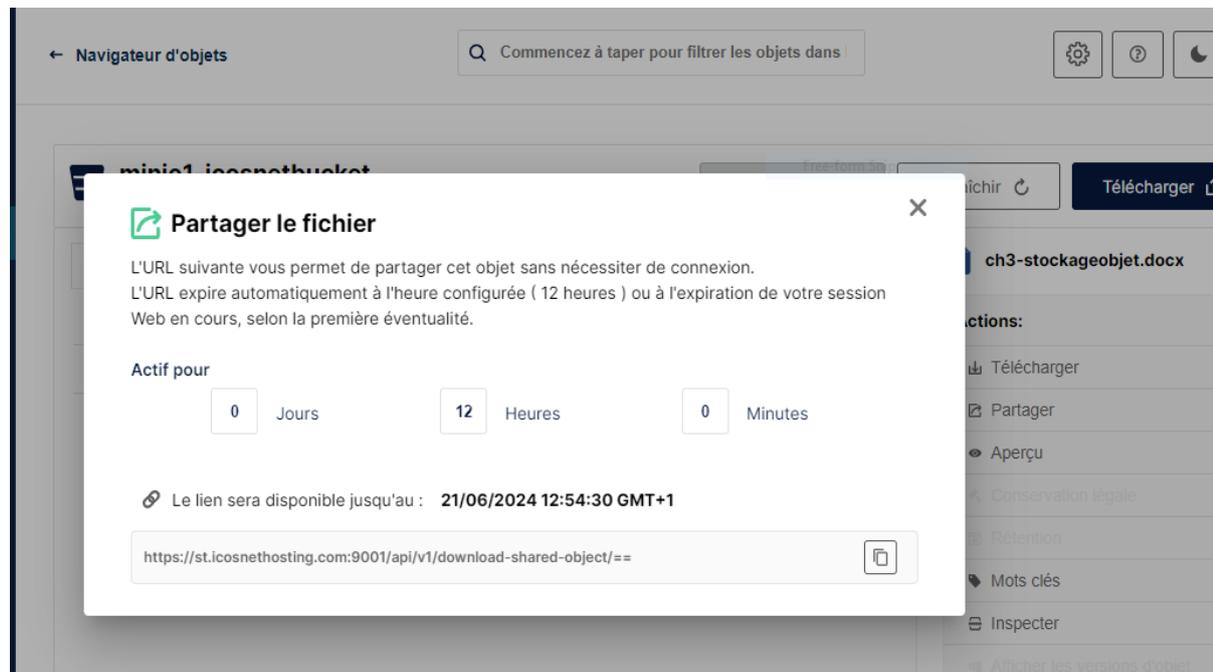


Figure IV.40 Le code URL pour le partage de fichier

**On peut partager l'URL à n'importe quelle destination et on va voir est ce que le fichier est bien envoyer comme montrer la figure ci-dessous**



**Figure IV.41 Le fichier a été bien partager.**

## IV.6 MinIO Client (mc)

L'outil de ligne de commande MinIO Client « mc client » fournit une alternative moderne aux commandes UNIX telles que **ls**, **cat**, **cp**, **mirror** avec prise en charge des systèmes de fichiers et des services de stockage cloud compatibles Amazon S3.

L'intention ici était de faire fonctionner le client d'une manière familière pour ceux d'entre vous qui utilisent Linux régulièrement tout en présentant un outil plus moderne et plus performant qui peut être utilisé à la fois localement et avec des systèmes de stockage d'objets comme MinIO.

### Exemple :

```
'cp file1.txt file2.txt' #Copies files locally from 'file1.txt' to 'file2.txt'
'mc cp file1.txt file2.txt' #Copies files locally from 'file1.txt' to 'file2.txt'
'mc cp ~/file1.txt minio1/testbucket1/file1.txt' #Copies the local 'file1.txt' to the 'minio1' cluster
(defined by an alias) into the bucket 'testbucket1'
'mc cp minio1/testbucket1/file1.txt minio2/testbucket2/file1.txt' #Copies the 'file1.txt' on 'minio1' in
'testbucket1' into the bucket named 'testbucket2' on another cluster aliased 'minio2'
```

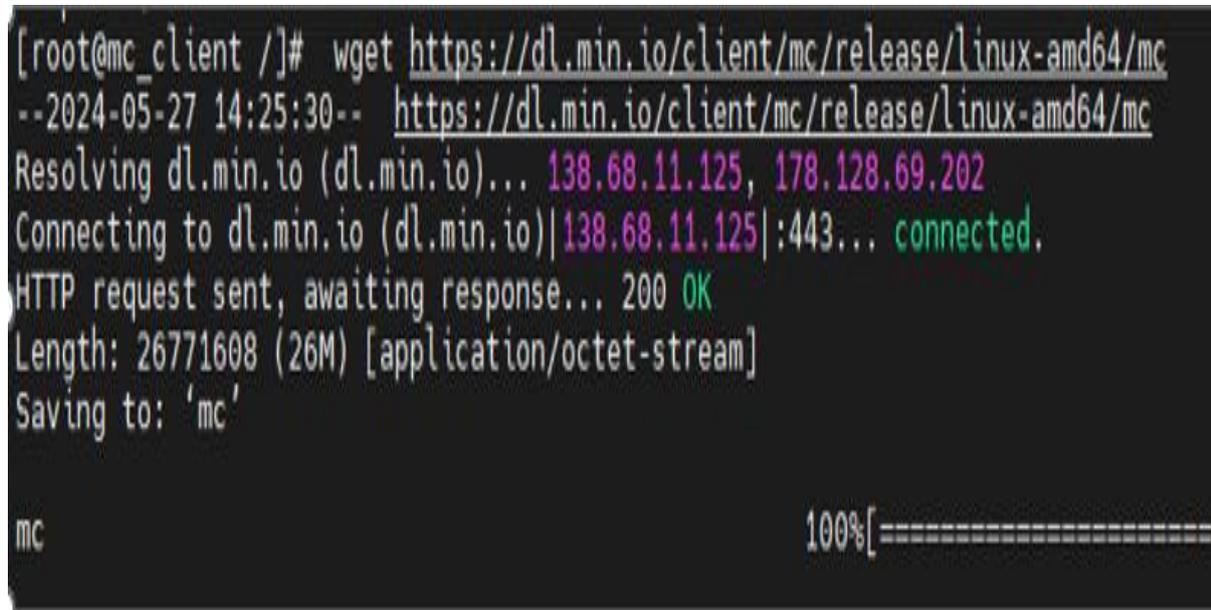
**Figure IV.42 La commande « cp ».**

Cette figure illustre la commande `cp` sous Linux qui copie des fichiers ou des répertoires d'un emplacement à un autre. Le client MinIO peut être utilisé pour faire la même chose avec `mc cp`, mais fonctionne à la fois avec les fichiers Linux locaux ainsi qu'avec la copie de fichiers du local vers S3 et de S3 vers S3.

### IV.6.1 Installer MinIO Client (MC) sur Linux

Les commandes suivantes ajoutent une extension temporaire à votre PATH système pour exécuter le mc utilitaire.

```
[root@mc_client /]# wget https://dl.min.io/client/mc/release/linux-amd64/mc
```

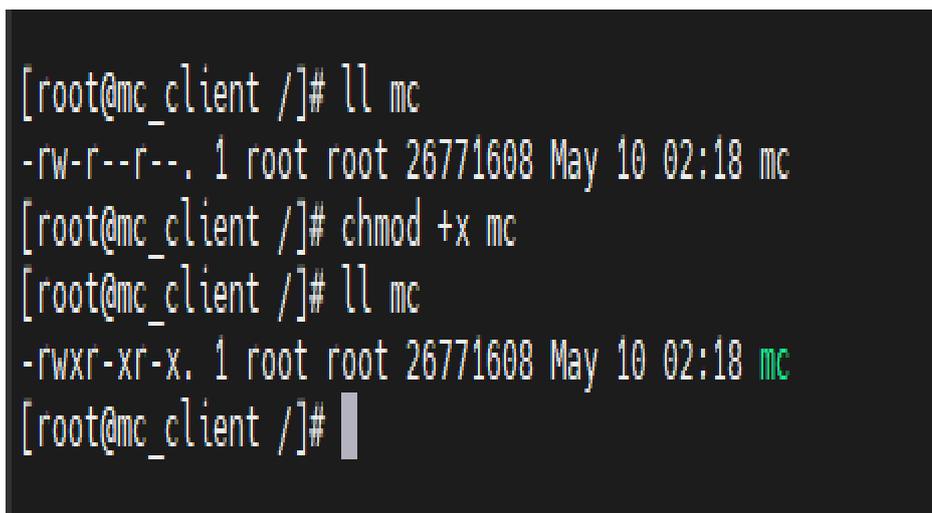


```
[root@mc_client /]# wget https://dl.min.io/client/mc/release/linux-amd64/mc
--2024-05-27 14:25:30-- https://dl.min.io/client/mc/release/linux-amd64/mc
Resolving dl.min.io (dl.min.io)... 138.68.11.125, 178.128.69.202
Connecting to dl.min.io (dl.min.io)|138.68.11.125|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 26771608 (26M) [application/octet-stream]
Saving to: 'mc'

mc                               100%[=====]
```

Figure IV.43 Téléchargements des packages.

Cette figure illustre la commande « wget » qui sert à télécharger les paquets de **MC client**. Après on passe à la modification des permissions et le rendre exécutable à l'aide de la commande « **chmod+x mc** » :



```
[root@mc_client /]# ll mc
-rw-r--r--. 1 root root 26771608 May 10 02:18 mc
[root@mc_client /]# chmod +x mc
[root@mc_client /]# ll mc
-rwxr-xr-x. 1 root root 26771608 May 10 02:18 mc
[root@mc_client /]# █
```

Figure IV.44 La modification des permissions et rendre le « mc » exécutable.

Cette figure montre comment nous avons modifier les permissions et rendre la machine client exécutable.

On pourra voir par la suite les diverses commandes de mc :

```
[root@mc_client /]# ./mc --help
NAME:
  mc - MinIO Client for object storage and filesystems.

USAGE:
  mc [FLAGS] COMMAND [COMMAND FLAGS | -h] [ARGUMENTS...]

COMMANDS:
  alias      manage server credentials in configuration file
  admin      manage MinIO servers
  anonymous  manage anonymous access to buckets and objects
  batch      manage batch jobs
  cp         copy objects
  cat        display object contents
  diff       list differences in object name, size, and date between two buckets
  du         summarize disk usage recursively
  encrypt    manage bucket encryption config
  event      manage object notifications
  find       search for objects
  get        get s3 object to local
  head       display first 'n' lines of an object
  ilm        manage bucket lifecycle
  idp        manage MinIO IDentity Provider server configuration
  license    license related commands
  legalhold  manage legal hold for object(s)
  ls         list buckets and objects
  mb         make a bucket
  mv         move objects
  mirror     synchronize object(s) to a remote site
  od         measure single stream upload and download
  ping       perform liveness check
  pipe       stream STDIN to an object
  put        upload an object to a bucket
```

Figure IV.45 Les diverses commandes de mc.

Cette figure illustre les diverses commandes de « mc » a l'aide de la commande « help »

Puis le déplacer dans le path `/usr/local/bin/mc` :

```
[root@mc_client /]# mv mc /usr/local/bin/mc
```

Figure IV.46 Déplacement vers le path `/usr/local/bin/mc`

Cette figure illustre comment nous avons déplacé le fichier de « mc » vers un autre chemin spécifique.

## Vérification :

```
[root@mc_client /]# ./usr/local/bin/mc --help
NAME:
  mc - MinIO Client for object storage and filesystems.

USAGE:
  mc [FLAGS] COMMAND [COMMAND FLAGS | -h] [ARGUMENTS...]

COMMANDS:
  alias      manage server credentials in configuration file
  admin      manage MinIO servers
  anonymous  manage anonymous access to buckets and objects
  batch      manage batch jobs
  cp         copy objects
  cat        display object contents
  diff       list differences in object name, size, and date between two buckets
  du         summarize disk usage recursively
  encrypt    manage bucket encryption config
  event      manage object notifications
  find       search for objects
  get        get s3 object to local
  head       display first 'n' lines of an object
  ilm        manage bucket lifecycle
  idp        manage MinIO IDentity Provider server configuration
  license    license related commands
  legalhold  manage legal hold for object(s)
  ls         list buckets and objects
  mb         make a bucket
  mv         move objects
  mirror     synchronize object(s) to a remote site
  od         measure single stream upload and download
  ping       perform liveness check
  pipe       stream STDIN to an object
  put        upload an object to a bucket
  quota     manage bucket quota
  rm         remove object(s)
  retention  set retention for object(s)
  rb         remove a bucket
  replicate  configure server side bucket replication
  ready      checks if the cluster is ready or not
  sql        run sql queries on objects
  stat       show object metadata
```

Figure IV.47 Le path /usr/local/bin/mc

Le **mc** est bien placé dans le path (chemain) /usr/local/bin/mc

### IV.6.2 Créez un alias pour le service compatible S3

Les commandes « mc alias » fournissent une interface pratique pour gérer la liste des hôtes compatibles S3 qui mc peuvent se connecter et exécuter des opérations.

Les commandes qui fonctionnent sur des services compatibles S3 nécessitent la spécification d'un alias pour ce service.

#### IV.6.2.1 Sous-commandes de « mc alias »

Sous-commandes	Description
<b>list</b>	La commande « <b>mc alias list</b> » répertorie tous les alias dans la configuration <b>mc</b> locale.
<b>remove</b>	« <b>mc alias remove</b> » Cela supprime un alias existant de la configuration <b>mc</b> locale.
<b>set</b>	La commande « <b>mc alias set</b> » ajoute ou met à jour un alias à la configuration <b>mc</b> locale.
<b>import</b>	La commande « <b>mc alias import</b> » importe une configuration d'alias à partir d'un document JSON.
<b>export</b>	La commande « <b>mc alias export</b> » exporte une configuration d'alias à partir de la configuration existante .

Tableau IV.1 Sous-commandes de « mc alias » [20].

Nous pouvons voir la version de mc installée a l'aide de la commande « version » comme montrer la figure ci-dessous :

```
[root@mc_client /]# mc --version
mc version RELEASE.2024-05-09T17-04-24Z (commit-id=fdb36acbb1d793b6cca622a55e6292f0d52309f0)
Runtime: go1.22.3 linux/amd64
Copyright (c) 2015-2024 MinIO, Inc.
License GNU AGPLv3 <https://www.gnu.org/licenses/agpl-3.0.html>
```

Figure IV.48 La version de mc installée.

**Ajouter un alias dans mc pour notre serveur qui est déployé :**

**st.icosnethosting.com** depuis la machine cliente **197.140.18.214** via la commande **mc** :

Il faut juste veiller que cette machine pourra connecter vers le serveur **MinIO 197.140.18.88** via le port **9000**.

```
[root@mc_client /]# telnet 197.140.18.88 9000
Trying 197.140.18.88...
Connected to 197.140.18.88.
Escape character is '^]'.
^CConnection closed by foreign host.
```

Figure IV.49 La machine client.

Cette figure illustre le test de connexion entre la machine client et serveur de minio.

**Vérification des alias par défaut :**

```
[root@mc_client /]# mc alias list
mc: Configuration written to `/root/.mc/config.json`. Please update your access credentials.
mc: Successfully created `/root/.mc/share`.
mc: Initialized share uploads `/root/.mc/share/uploads.json` file.
mc: Initialized share downloads `/root/.mc/share/downloads.json` file.
gcs
  URL      : https://storage.googleapis.com
  AccessKey : YOUR-ACCESS-KEY-HERE
  SecretKey : YOUR-SECRET-KEY-HERE
  API      : S3v2
  Path     : dns
local
  URL      : http://localhost:9000
  AccessKey :
  SecretKey :
  API      :
  Path     : auto
play
  URL      : https://play.min.io
  AccessKey : Q3AM3UQ867SPQQA43P2F
  SecretKey : zufftfteslswRu7BJ86wekitnifILbZam1KYY3TG
  API      : S3v4
  Path     : auto
s3
  URL      : https://s3.amazonaws.com
  AccessKey : YOUR-ACCESS-KEY-HERE
  SecretKey : YOUR-SECRET-KEY-HERE
  API      : S3v4
  Path     : dns
```

Figure IV.50 Les alias par défaut.

D'après cette figure Nous pouvons voir qu'il y'a des alias qui se trouvent par défaut, à titre d'exemple :

URL: <https://storage.googleapis.com>

URL: <http://localhost:9000>

URL: <https://play.min.io>

URL: <https://s3.amazonaws.com>

Nous avons donc essayé d'ajouter le serveur via le port 9000, comme montre la figure ci-dessous :

```
[root@mc_client /]# mc alias set https://197.140.18.88:9000 minio-user 1234567890
Enter Secret Key: █
```

Figure IV.51 Ajouter le serveur via le port 9000.

Ce dernier revendique une clé secrète pour pouvoir l'ajouter comme alias et avoir la main pour le manipuler via mc client, nous pouvons donc fournir cette clé depuis le panneau :

<https://st.icosnethosting.com:9001/access-keys/new-account>, comme montre la figure ci-dessous :

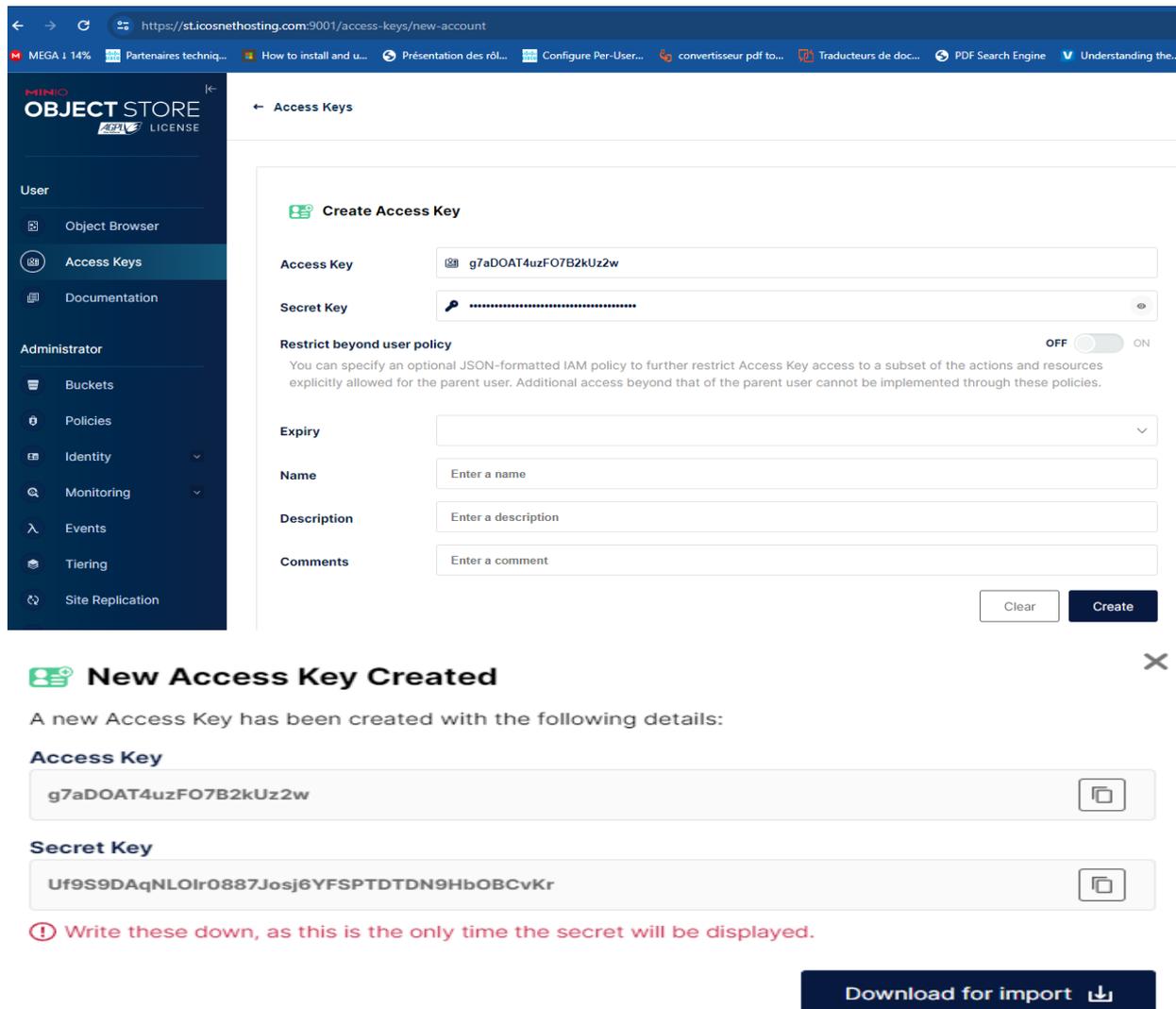


Figure IV.52 La création d'une clé d'accès.

Puis On Coller la clé d'accès ici :

```
[root@mc_client /]# mc alias set https://197.140.18.88:9000 minio-user 1234567890
Enter Secret Key: █
```

Figure IV.53 L'insertion de la clé d'accès.

Juste il faut s'assurer que la résolution DNS soit faite depuis la machine **mc client**

```
[root@mc_client /]# mc alias set minio1 https://st.icosnethosting.com:9000 minio-user 1234567890
Added `minio1` successfully.
```

Figure IV.54 La résolution DNS elle est intacte.

Pour vérifier cela, nous pouvons lister les alias :

```
[root@mc_client /]# mc alias list
gcs
  URL      : https://storage.googleapis.com
  AccessKey : YOUR-ACCESS-KEY-HERE
  SecretKey : YOUR-SECRET-KEY-HERE
  API      : S3v2
  Path     : dns

local
  URL      : http://localhost:9000
  AccessKey :
  SecretKey :
  API      :
  Path     : auto

minio1
  URL      : https://st.icosnethosting.com:9000
  AccessKey : minio-user
  SecretKey : 1234567890
  API      : s3v4
  Path     : auto

play
  URL      : https://play.min.io
  AccessKey : Q3AM3U0867SP0QA43P2F
  SecretKey : zuf+tfteSlwRu7Bj86wekitnifILbZam1KYY3TG
  API      : S3v4
  Path     : auto

s3
  URL      : https://s3.amazonaws.com
  AccessKey : YOUR-ACCESS-KEY-HERE
  SecretKey : YOUR-SECRET-KEY-HERE
  API      : S3v4
  Path     : dns
```

Figure IV.55 Le mc alias est bel et bien installé.

Cette figure montre que le serveur **st.icosnethosting.com** a été ajouté avec succès, si veut lister un alias spécifique qui est dans notre cas le serveur :

```
[root@mc_client /]# mc alias list minio1
minio1
  URL      : https://st.icosnethosting.com:9000
  AccessKey : minio-user
  SecretKey : 1234567890
  API      : s3v4
  Path     : auto
```

Figure IV.56 Le serveur MinIO est ajouté avec succès.

## Test de la connexion depuis mc client vers le serveur MINIO :

```
[root@mc_client /]# mc admin info minio1
● st.icosnethosting.com:9000
  Uptime: 6 days
  Version: 2024-05-10T01:41:38Z
  Network: 1/1 OK
  Drives: 1/1 OK
  Pool: 1

Pools:
  1st, Erasure sets: 1, Drives per erasure set: 1

1.1 MiB Used, 3 Buckets, 4 Objects, 4 Versions
1 drive online, 0 drives offline, EC:0
```

Figure IV.57 Test de la connexion depuis mc client vers le serveur MinIO.

Cette figure montre que la connexion depuis mc client vers le serveur MinIO est en marche.

### IV.6.2.2 Command « mc mb » :

La commande « **mc mb** » pour crée un compartiment (bucket) ou un répertoire au chemin spécifié, donc nous pouvons créer un bucket via la commande « **mc mb** », comme montre la figure ci-dessous :

```
[root@mc_client /]# mc mb minio1/minio1-icosnetbucket
Bucket created successfully `minio1/minio1-icosnetbucket`.
```

Figure IV.58 Créer un bucket.

Vérification au niveau de la console MinIO : <https://st.icosnethosting.com:9001/login>

<b>icosnet-26-05-2024</b> Created: Sun May 26 2024 15:04:05 GMT+0200 (heure d'été d'Europe centrale) Access: R/W
 Usage <b>390.6</b> <sub>KiB</sub> / <b>200</b> <sub>MiB</sub>
 Objects <b>3</b>
<b>minio1-icosnetbucket</b> Created: Mon May 27 2024 16:59:35 GMT+0200 (heure d'été d'Europe centrale) Access: R/W
 Usage <b>0.0</b> <sub>B</sub>
 Objects <b>0</b>
<b>test</b> Created: Sun May 26 2024 17:32:19 GMT+0200 (heure d'été d'Europe centrale) Access: R/W
 Usage <b>0.0</b> <sub>B</sub>
 Objects <b>0</b>

Figure IV.59 Vérification au niveau de la console MinIO.

Cette figure montre que la bucket (minio1-icosnetbucket) a été bien ajoutée dans la console MinIO.

## IV.7 Application d'intégration de la solution stockage objet :

Dans cette petite manipulation nous allons déclarer notre solution de stockage objet comme un répertoire pour transférer les fichiers backups de la machine cliente à titre d'exemple.

La solution de backup **Veeam** qui est une solution de sauvegarde et restauration en cloud plus précisément.

L'application Veeam est installée sous un serveur Windows server hébergé au niveau de cloud Icosnet avec une connexion via le protocole **SMB** avec des baies de stockages externes pour le Upload des backups.

Vu que la solution MinIO est prête, nous allons donc exploiter la rubrique stockage objet, on doit tout d'abord générer une clé secrète depuis la console MinIO, il faut noter qu'une fois cette dernière sera transmise à l'administrateur des services backups, le serveur Veeam backup aura le privilège de manipuler les buckets de MinIO via l'API en interaction avec le serveur via le port **9000** <https://st.icosnethosting.com:9000>

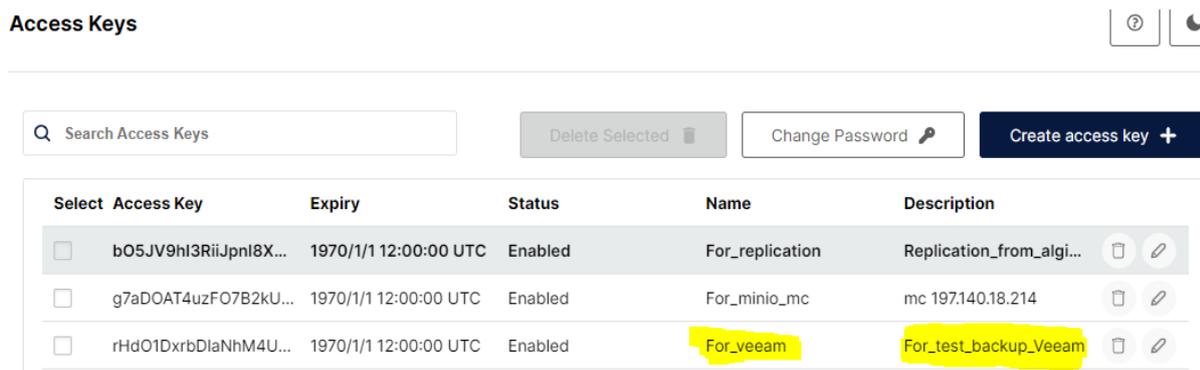


Figure IV.60 Générer une clé secrète.

Cette figure illustre la création d'une clé secrète au niveau de la console MinIO.

**La deuxième étape consiste à configurer le repository là où le Veeam va stocker ses backups (MINIO) en fournissant les informations nécessaires (donner un nom) comme montre la figure ci-dessous :**

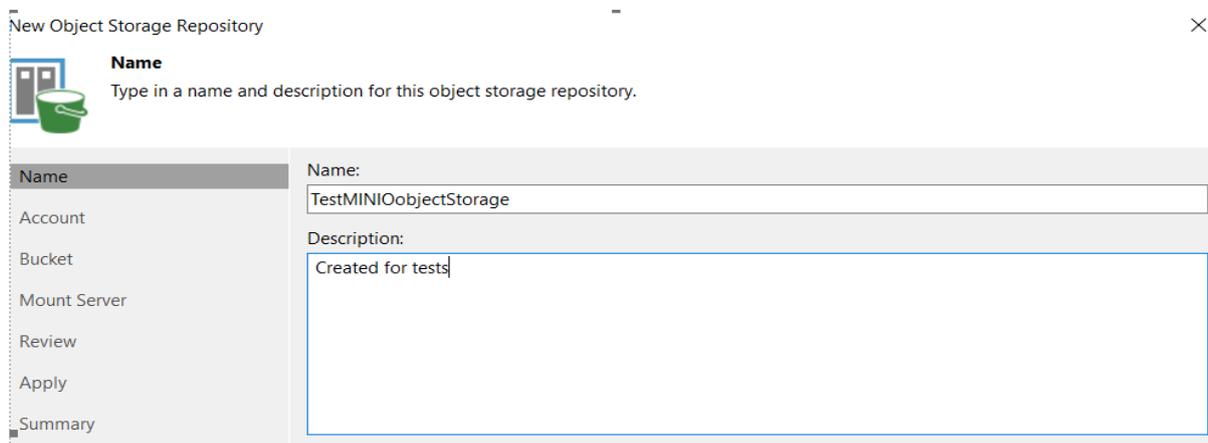
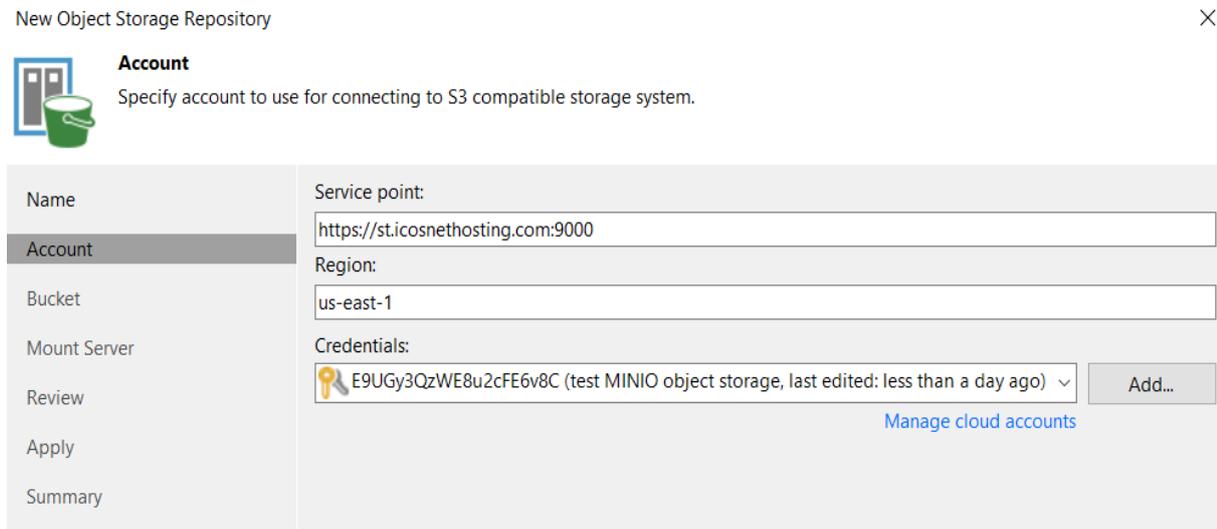


Figure IV.61 Donner un nom à la repository de stockage objet.

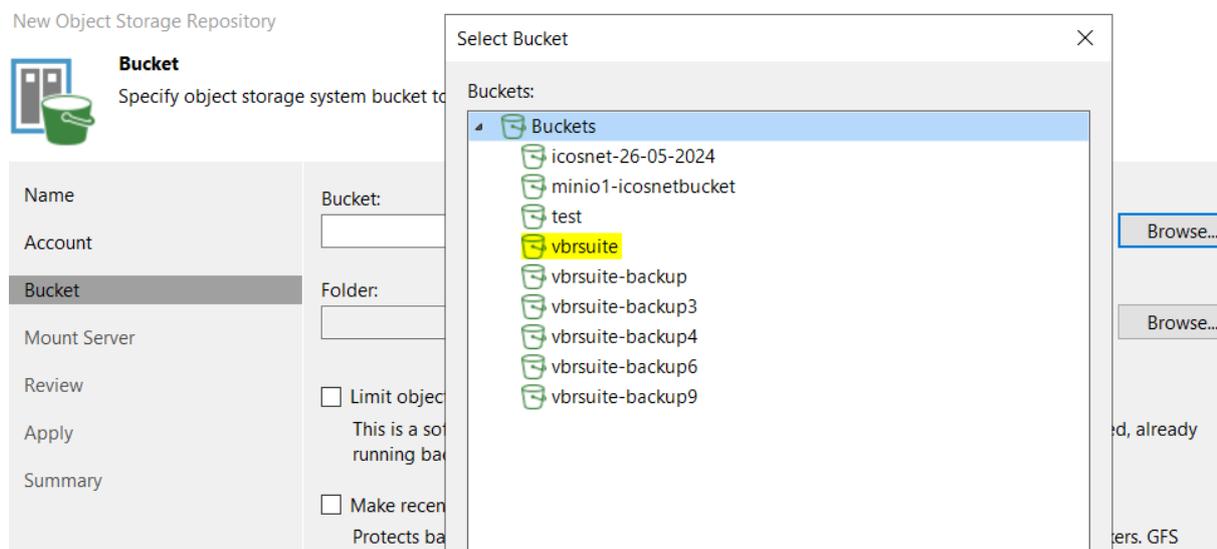
**Spécifier les informations du serveur MINIO, aussi la clé secrète déjà partagée avec ce dernier comme montre la figure ci-dessous :**



**Figure IV.62 Les informations du serveur MinIO.**

Pour la région c'est par défaut choisie par Veeam, on va la laisser telle qu'elle est.

**Après avoir introduit toutes les informations nécessaires, il est temps de voir le résultat suivant :**



**Figure IV.63 Les bucket de MinIO sur le Veeam.**

D'après cette figure, on pourra voir que le serveur backup Veeam aura désormais la vue sur les divers buckets créés dans le serveur **MinIO**.

Dans notre exemple d'application nous avons choisi le bucket comme choix de dépôt pour stocker les fichiers backups de la machine virtuelle MC client (mc).

Ci-dessous le résultat final depuis le serveur Veeam :

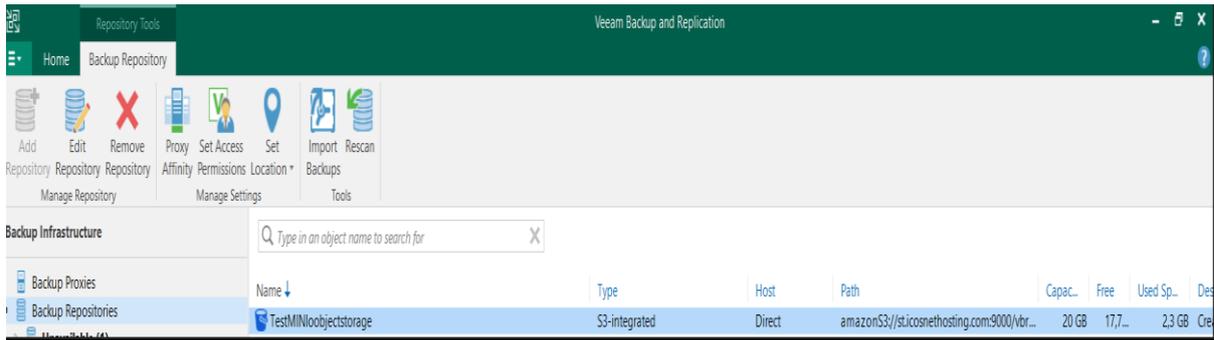


Figure IV.64 Le résultat final depuis le serveur Veeam.

Cette figure illustre que le serveur Veeam a bien fait le backup de la machine client (mc) dans une bucket (vbrsuite) dans MinIO server.

Au niveau de la console MinIO :



Figure IV.65 La bucket vbrsuite sur le serveur MinIO.

Cette figure montre que les fichiers backup de la machine client sont bien stockés dans la bucket vbrsuite au niveau de la console.

**En résumé :** MinIO compatible S3 pourra être utilisé et exploité avec les solutions de sauvegardes via son API en toute flexibilité.

## IV.8 Conclusion :

Dans ce chapitre nous avons réussi à publier une application via le web dans cloud algérien, les divers résultats montrent que l'application MinIO est très fiable et facile à gérer suivant les rebriques, aussi un test en production via une solution qui le prouve.

## Conclusion Générale

---

La rédaction de ce mémoire reflète un travail pratique qui répond au besoin d'entreprise en termes d'un service de stockage fiable accessible via l'internet.

Les étapes de conception était basée sur les bonnes pratiques des conceptions et publication des applications dans le web, vue les résultats nous pouvons confirmer que la solution de stockage objet répond parfaitement aux besoins clients, au terme de services et rubriques, depuis la console web mais aussi par rapport aux administrations via les outils qui facilitent le dépannage et l'administration soit à distance, soit via la console shell.

Entre outre, plusieurs autres rubriques peuvent être améliorés notamment la partie journalisations, réplication d'un site distant, déploiement de la solution dans un cluster de plusieurs nœuds, cela dépend de la croissance du besoin.

## 1. DNS

Le DNS (Domain Name System, ou système de noms de domaine) fonctionne comme l'annuaire téléphonique de l'Internet. Alors que les internautes accèdent aux informations en ligne à l'aide de noms de domaine (tels que nytimes.com ou espn.com), les navigateurs web utilisent des adresses IP (Internet Protocol, ou protocole Internet). Le DNS convertit ces noms de domaine en adresses IP, permettant ainsi aux navigateurs de charger les ressources web.

## 2. Nom de Domaine

Un nom de domaine est une chaîne de texte associée à une adresse IP alphanumérique, permettant d'accéder à un site Web via un logiciel client. C'est le texte que les utilisateurs saisissent dans la barre d'adresse de leur navigateur pour visiter un site web spécifique. Par exemple, le nom de domaine de Google est "google.com".

## 3. Un port

Les ports sont des points virtuels où commencent et se terminent les connexions réseau. Les ports sont basés sur des logiciels et gérés par le système d'exploitation de l'ordinateur. Chaque port est associé à un processus ou un service spécifique.

## 4. Numéro de port

Les ports de tous les appareils connectés au réseau sont standardisés et chaque port se voit attribuer un numéro. La plupart des ports sont réservés à certains protocoles ; par exemple, tous les messages HTTP (Hypertext Transfer Protocol) vont au port 80.

## 5. Le protocole RDP

Le protocole RDP (Remote Desktop Protocol) est conçu pour permettre aux utilisateurs du monde entier d'accéder et de contrôler les ordinateurs via un canal fiable et sécurisé. Cet outil pratique et sécurisé contribue à rendre les employés plus productifs et leur donne la flexibilité dont ils ont besoin pour accomplir leur travail dans un monde en constante évolution.

## 6. Firewall NAT (pare-feu NAT)

Les pare-feu NAT sont utilisés pour assurer la sécurité du réseau local et protéger les données sur les appareils locaux. Le pare-feu NAT traduit cette adresse IP privée en adresse IP publique lorsque le paquet traverse le réseau interne. Le pare-feu NAT bloque les données indésirables ou malveillantes lorsque des paquets provenant de l'extérieur du réseau sont renvoyés pour empêcher les pirates de s'infiltrer dans votre réseau local via des connexions non autorisées.

## 7. API

Une API (interface de programmation d'application) est un ensemble de définitions et de protocoles qui simplifie la création et l'intégration des applications.

## 8. Mxtoolbox

Mxtoolbox est un outil tiers permettant de vérifier les enregistrements DNS de votre domaine. En tant que webmaster, cet outil est très utile pour établir ou rechercher des enregistrements DNS et obtenir d'autres rapports d'analyse sur la santé de votre domaine.

## 9. Private key

La clé privée, ou Private key, est un code alphanumérique complexe utiliser pour sécuriser les communications numériques sur le web. Elle est stockée dans un fichier .key sur le serveur web et protège les certificats SSL.

## 10. Public.crt

Un fichier **crt**, ou fichier de certificat, contient uniquement la clé publique ainsi que d'autres informations sur le certificat numérique. Ces fichiers peuvent être utilisés pour des connexions sécurisées.

## 11. la commande « Wget »

« **Wget** » est un programme de téléchargement de fichiers depuis le Web, utilisable en ligne de commande et sans interaction. Il prend en charge les protocoles HTTP, HTTPS et FTP.

## RÉFÉRENCES BIBLIOGRAPHIQUES

---

- [1] : <https://icosnet.com.dz/> Consulté 2/2024.
- [2] : AOUAMEUR, TAHRINE « Comparaison et mise en place des plateformes de Cloud Computing : OpenStack et Eucalyptus », mémoire de master présentée l'université Kasdi Merbah de Ouargla, 2013.
- [3] : <https://www.cloudflare.com/en-gb/learning/cloud/what-is-the-cloud/> Consulté 2/2024.
- [4] : [https://www.ibm.com/topics/cloud-computing?mhsrc=ibmsearch\\_a&mhq=cloud%20computing#Cloud+use+cases](https://www.ibm.com/topics/cloud-computing?mhsrc=ibmsearch_a&mhq=cloud%20computing#Cloud+use+cases) Consulté 2/2024.
- [5] : <https://www.redhat.com/fr/topics/virtualization> Consulté 2/2024.
- [6] : <https://www.ovhcloud.com/fr/what-data-center/> Consulté 2/2024.
- [7] : <https://kinsta.com/fr/blog/types-de-cloud-computing/#iass> Consulté 2/2024.
- [8] : MIMOUNE, « Etude sur la sécurité du Cloud Computing », mémoire de master présentée l'université de M'SILA, 2015.
- [9] : <https://management-digital.macertif.com/Cloud-Computing/A2-difference-hybride-prive-public-iaas-pass-saas/#> Consulté 3/2024.
- [10] : HAMDANI, KERROUM, OUALLOUCHE « Etude et comparaison des failles de sécurité D'Open Stack et Open Nebula », mémoire de master présentée l'université MOULOUD MAMMERI DE TIZI-OUZOU, 2019.
- [11] : <https://www.telehouse.fr/actualites/les-9-avantages-du-cloud-computing-pour-les-entreprises/> Consulté 3/2024.
- [12] : <https://www.oracle.com/fr/cloud/multicloud/what-is-multicloud/> Consulté 3/2024.
- [13] : <https://www.lebigdata.fr/pourquoi-adopter-cloud-computing> Consulté 3/2024.
- [14] : <https://aws.amazon.com/fr/what-is/object-storage/> Consulté 3/2024.
- [15] : <https://www.mongodb.com/resources/basics/unstructured-data/unstructured-data> Consulté 3/2024.
- [16] : <https://www.astera.com/type/blog/unstructured-data-challenges/> Consulté 3/2024.
- [17] : <https://www.twaino.com/definition/b/balises-de-donnees-structurees/> Consulté 3/2024.
- [18] : <https://cloud.google.com/learn/what-is-object-storage?hl=fr#section-3> Consulté 3/2024.
- [19] : <https://www.netapp.com/data-storage/storagegrid/what-is-object-storage/> Consulté 3/2024.
- [20] : <https://www.educba.com/object-storage-vs-block-storage/> Consulté 3/2024.

## RÉFÉRENCES BIBLIOGRAPHIQUES

---

[21] : <https://www.scaleway.com/fr/blog/comprendre-les-differents-types-de-stockage/>

Consulté 3/2024.

[22] : <https://min.io/docs/minio/linux/reference/minio-mc/mc-alias.html> Consulté 3/2024.

