




Security Whitepaper

Last Updated: Sep 25, 2023

 *Metaphor only gathers metadata and does not access customers' data.*

Summary

We've architected Metaphor from the ground up to be secure and followed best-in-class security practices designed to meet or exceed several industry security compliance standards. This document highlights several guiding principles of our security architecture, including the least privilege principle, defense in depth, minimized attack surface, multi-factor authentication, network segmentation, and zero-trust security model.

While Metaphor only collects and processes metadata, such as schema definition, lineage/provenance, pipeline status, usage statistics, and quality metrics, from the customer's data ecosystem, we give it the same level of security treatment as if it were the customer's data.

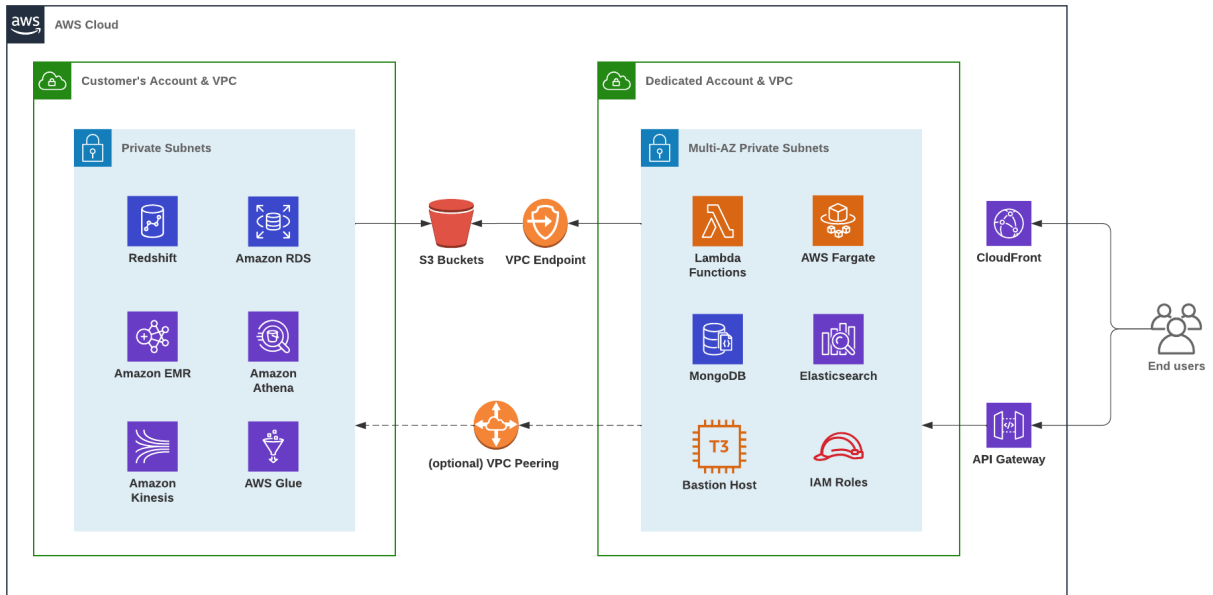
This document provides an overview of the Metaphor platform architecture, design choices, and security features that ensure the customer's metadata is exchanged, stored, processed, and accessed securely in a single cloud or cross-cloud environment.

Content

Architecture & Tenancy Model.....	4
Metadata Crawlers.....	4
Network Security.....	5
Server Security.....	5
Identity & Access.....	6
Data Security.....	6
Security Operations.....	7
Compliance.....	7
Cross-Cloud Security.....	8

Architecture & Tenancy Model

Metaphor has chosen the full-stack [account silo isolation strategy](#) by creating a **dedicated AWS account for each customer**. All the compute and storage¹ for the customer’s metadata reside in the account’s VPC private subnets, as shown below.



The dedicated AWS account is created under Metaphor’s AWS Organization by default. Customers can request to transfer the account to their AWS Organization² (In-VPC Deployment).

Metadata Crawlers

Metaphor uses metadata from multiple sources to power its platform. The metadata is gathered using system-specific “crawlers” that run regularly to keep the metadata up to date. The extracted metadata is written to a secure S3 bucket with access control & logging. We also [open-sourced](#) the crawler code for full auditability.

The customer can choose to operate the crawlers in two different configurations:

¹ With the exceptions of AWS services that can’t be placed inside a VPC, e.g. S3, SQS, API Gateway, and CloudFront. For these services, each customer still maintains its own dedicated resource, e.g. customer-specific S3 bucket, with strict IAM policies to limit access.

² Once transferred, all costs incurred in the account will roll up to the customer's consolidated billing. The existing IAM role trust relationships must be retained in order for Metaphor to deploy, manage, and monitor the resources.

1. Fully Managed: Metaphor schedules, executes, upgrades, and monitors the crawlers for the customer. The crawlers run inside the dedicated VPC and may require a [VPC peering connection](#) (detailed in [Network Security](#)). Credentials or API tokens are provided to Metaphor and stored in a secure S3 bucket (detailed in [Identity & Access](#)).
2. Self-Managed: The customer manages and runs their own crawlers inside their VPCs (or even on-prem networks) and only shares the output with Metaphor through a secure S3 bucket. No credentials sharing with Metaphor is needed in this configuration.

The configurations are not mutually exclusive—it is possible to mix and match based on the specific system or security requirements.

Network Security

As mentioned in the previous section, the compute and storage resources are placed inside private subnets. These subnets have no direct route to/from the Internet and can only initiate outbound connections through a NAT gateway. The only way to gain access to these resources inside the subnet is via a [bastion host](#), which itself is also placed in the private subnet and can only be connected through [AWS Session Manager](#).

If the customer chooses fully managed crawlers, a VPC peering connection can be set up so that the crawlers can directly connect to the customer's systems without routing through the public Internet.

We also created [VPC endpoints](#) to link the private subnets directly to all supported AWS services, such as S3 & SQS. This ensures that network traffic is routed directly to these services without going through the public Internet.

Server Security

Metaphor relies heavily on serverless ([Lambda](#) and [Fargate](#)) for computation and request serving. We use only the [default Lambda runtimes](#) or [official Node images](#), which receive regular security patches. The container images are updated to the latest version automatically during weekly deployment.

The bastion host uses the latest hardened [Amazon Linux 2 AMI](#) with SSH access completely disabled. AWS releases a new version of AMI every 2-4 weeks to include the latest security patches. We typically rebuild the bastion host using the new AMI within a few days of the release.

Identity & Access

The dedicated AWS account contains only a handful of IAM roles and no actual IAM users. In fact, the password for the account's root user is discarded immediately after creation so that the only way to gain admin access (when needed) is by assuming the admin IAM role. As a result, no AWS access key will ever be created, minimizing the risk of leaked credentials.

Following the principle of least privilege, each IAM role is designed to perform a specific task (e.g., app deployment, infrastructure provisioning/configuration, debugging, and monitoring) and is given the minimum set of permissions. The admin IAM role is only used to bootstrap the environment and update IAM policies.

If the customer chooses fully managed crawlers, data system credentials and API tokens are stored in a secure S3 bucket rather than in a database. This way, we can directly take advantage of the many security features offered by S3 (e.g., IAM integration, ACL, versioning, and logging) instead of devising custom security mechanisms. Rotating the password also becomes as simple as updating the corresponding S3 object and can be fully self-served by the customer.

The Metaphor app doesn't support password-based authentication. Users are required to authenticate using OIDC & SAML-based SSO. We strongly recommend customers enforce multi-factor authentication on their side.

Internally, Metaphor employees use SSO + MFA to log into AWS and other third-party services. For systems that do not support SSO, we use 1Password to generate strong passwords and prevent password reuses.

Data Security

Data is fully encrypted both at rest and in transit:

- All network traffic, both within the VPC and entering/leaving the VPC, is protected using TLS 1.2 (or later versions) with AES-256 encryption.
- At-rest encryption is enabled for all transactional data systems (MongoDB & Elasticsearch) using the account's default AWS KMS encrypt key.
- All data stored in S3, including the backup data, is encrypted using the account's default AWS KMS encrypt key³.

MongoDB is also backed up continuously and can be restored quickly in the event of catastrophic failure. While the backup data is stored in S3, it is not accessible by any IAM roles (including the admins) and can only be used for disaster recovery purposes.

³ We can also use customer-provided KMS encryption keys upon request.

Security Operations

Metaphor has adopted a secure process to allow developers to deploy secure code and infrastructure while providing guardrails to ensure security best practices, as well as operationalizing and hardening security throughout the software lifecycle. These include

- Software Development Life Cycle (SDLC) process development
- CI/CD toolchain hardening
- Static analysis and code review
- Infrastructure as code (IaC)
- Container security and immutability
- Continuous vulnerability monitoring
- Regular penetration testing
- Cloud configuration audit and compliance audit

All new Metaphor employees undergo background checks before starting. Only authorized staff have access to the Metaphor infrastructure and codebase.

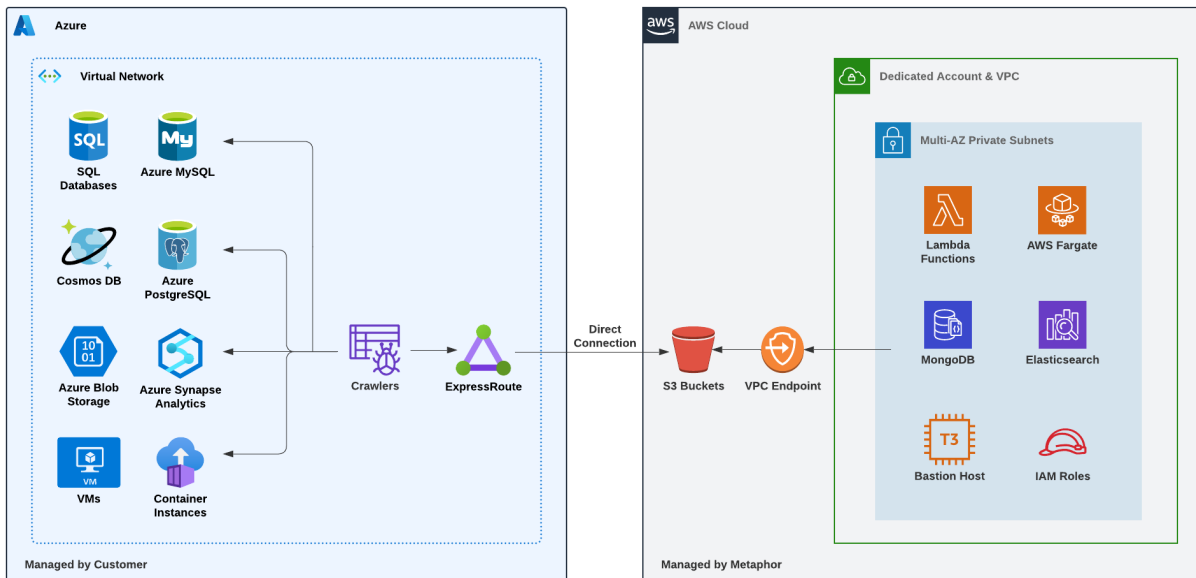
Compliance

Metaphor is SOC 2 Type 2 compliant and can provide a report upon request. We're also actively working towards other certifications, such as ISO 27001:2022, HITRUST, and FedRAMP.

Cross-Cloud Security

If your organization has data infrastructure in Azure or GCP, there are ways to ensure that your metadata is extracted, transmitted, and stored in a secure manner.

We use Azure as an example below to demonstrate how Metaphor can securely move metadata across cloud vendors. The remainder of this section assumes that the customer has all of its data infrastructure in Azure. Metaphor will still host its applications & services in a customer-specific AWS account, which can be owned by Metaphor or the customer.



To ingest metadata from various Azure data systems, customers will run [Self-Managed Crawlers](#) in their Azure environment to extract and upload the metadata to a secure S3 bucket in AWS. The bucket has a strict ACL that only allows writes from specific [IAM accounts](#).

While the connection to S3 is encrypted, the customer can further enhance the security by using an Azure [ExpressRoute](#) to form a direct connection from its VNet to AWS. This ensures that the network packets never get routed through the public Internet.