WHITEPAPER

4CEE CLOUD

4CEE

Table of contents

Version	Date	Author(s)	Comments
1.4	17-10-2025	Sjoerd van den Top	Reviewed and incorporated feedback: - Added missing TradeInterop name in several places - 5.2.1 Removed vulnerability scanning ECR - 6.1.2 set Kubernetes backup to the correct 35 days - Accepted proposed grammar corrections in several places Added 5.3 on GitOps
1.3	14-10-2025	Sjoerd van den Top	Updated 7.6 Personal Screening to reflect the VOG requirement is due to Stiply ISO.
1.2	08-10-2025	Sjoerd van den Top	Document review and small grammatical, textual and layout changes
1.1	01-10-2025	Cloud team	Document review
1.0	30-9-2025	Cloud team	Initial document

1 Introduction & objectives

This document defines the role, responsibilities, and value of the central cloud team within the 4CEE organization. As the backbone of a multi-platform cloud ecosystem supporting Easy Systems, Diesis, Stiply, ICreative, TradeInterop and Coforce the central cloud team ensures that all cloud environments are secure, scalable, compliant, and efficiently managed.

The objective of this document is to:

- Clarify the purpose and function of the central cloud team.
- Outline its key responsibilities, including infrastructure, security, governance, and cost optimization.
- Describe how the team collaborates with product, development, and operations teams.
- Highlight the benefits of centralized cloud management across diverse platforms.
- Provide technical transparency to customers by sharing architectural principles, operational practices, and security measures.

This whitepaper consolidates insights from multiple platform-specific documents into a unified and consistent overview, offering both internal stakeholders and external customers a clear understanding of how centralized cloud operations contribute to reliability, agility, and trust.

2 Cloud architecture & technology

The 4CEE cloud ecosystem supports a diverse set of SaaS platforms including Easy Systems, Diesis, Stiply, ICreative, TradeInterop and Coforce all hosted on Amazon Web Services (AWS). While each platform serves different functional domains, they are unified under a centrally managed infrastructure operated by the 4CEE cloud team.

2.1 Role of the central 4CEE cloud team

The central cloud team is responsible for the design, provisioning, security, and operational management of the shared cloud infrastructure. Their work ensures consistency, scalability, and compliance across all platforms, while enabling product teams to focus on application development.

2.1.1 Infrastructure setup & automation

- The team provisions and maintains cloud environments using Infrastructure as Code (IaC), ensuring repeatability, traceability, and version control.
- Environments are segmented by Virtual Private Clouds (VPCs), with isolated development, test, and production stages.
- Network architecture includes load balancers, NAT gateways, private/public subnets, and security groups, tailored per platform.
- Time Synchronization (amazon time sync service)
  All compute resources synchronize time using the Amazon Time Sync Service (NTP) available at 169.254.169.123 within every VPC. This service uses regionally distributed, satellite-connected and atomic clocks and smooths leap seconds to keep systems in sync.
  - **Scope**: EC2 instances (chrony/ntpd), EKS worker nodes, and container workloads inherit the same monotonic time source.
  - **Rationale**: Accurate, consistent time underpins TLS, token validity, forensic correlation (logs/metrics/traces), and distributed systems behavior.
  - **Operations**: Time source configuration is part of golden images and node bootstrapping; drift alerts can be surfaced via node agents.

2.1.2 Security & access control

- Security is enforced through role-based access, multi-factor authentication, and least-privilege IAM roles.
- Data is encrypted both at rest and in transit, using industry-standard protocols and key management services.

- The team manages firewalls, threat detection, and vulnerability scanning using native AWS services.
- Privileged Access Paths Direct access to virtual servers is restricted to AWS Systems Manager Session Manager or a corporate VPN and always protected with MFA. Guest or anonymous access is prohibited. Detailed network configurations are not publicly exposed. Access is immediately revoked upon role change or termination.

2.1.3 Monitoring & observability

- Centralized monitoring is implemented across all layer's infrastructure, application, and network using tools like CloudWatch, Network Flow Logs, and CloudTrail.
- Alerts are routed through multiple channels to ensure rapid response, including integration with team communication platforms.

2.1.4 Deployment & CI/CD

- Development teams rely on standardized CI/CD pipelines for automated testing, building, and deployment.
- Platforms use a mix of containerized, serverless, and traditional EC2-based architectures, depending on their technical requirements.
- Deployment processes include automated rollbacks, quality gates, and peer reviews to ensure stability and security.

2.1.5 Backup & recovery

- Automated backups are scheduled across all environments, with cross-region replication for resilience.
- Recovery procedures are in place to restore services and data within defined timeframes, supported by escalation and communication protocols.

2.1.6 Capacity & performance management

The platform balances cost, performance, and reliability by combining AWS Auto Scaling Groups (ASG) for EC2 capacity and karpenter for Kubernetes node provisioning:

- **EC2 & Control Plane Capacity**
  Auto Scaling Groups provide elastic EC2 capacity for control-plane adjacent services and non-Kubernetes workloads. Scaling policies use CloudWatch metrics (CPU, memory via agents, queue depth, request rate) and can be adjusted through code-reviewed IaC.
- **Kubernetes Node Autoscaling (Karpenter)**
  Karpenter provisions right-sized nodes on-demand based on pod scheduling needs (CPU/Memory/Topology). It consolidates and reclaims nodes to reduce waste and supports multiple instance families for resilience and cost optimization.

- **Workload Autoscaling**
  Kubernetes **Horizontal Pod Autoscaler (HPA)** is used for application replicas based on metrics (e.g., CPU, custom Prometheus metrics). For batch/queue workloads, scaling reacts to backlog and latency SLOs.
- **Event-Driven Capacity**
  Monitoring events (CloudWatch alarms / Prometheus Alertmanager) can trigger scale-out/scale-in, or open an ops workflow for human approval (e.g., large traffic spikes, incident response).
- **Guardrails & Cost Controls**
  Resource quotas and budgets safeguard against runaway scaling. All changes are traceable via IaC and CI/CD gates.

This centralized model enables 4CEE to maintain a secure, scalable, and compliant cloud foundation, while supporting the autonomy and agility of individual product teams. The cloud team acts as the backbone of the ecosystem ensuring operational excellence and enabling innovation across platforms.

3 Security & compliance framework

The central cloud team plays a critical role in ensuring the security and regulatory compliance of all cloud-based platforms within the 4CEE organization. This includes Easy Systems, Diesis, Stiply, ICreative, TradeInterop and Coforce each operating within a shared AWS infrastructure but with distinct application architectures and data handling requirements.

3.1 Security management

Security is enforced across all layers of the cloud environment infrastructure, platform, and application using a combination of AWS-native services and centralized policies. Key practices include:

3.1.1 Identity & access management (IAM)

- **Least Privilege & RBAC –** Access is granted via IAM roles with the minimum required permissions.
- **MFA –** Mandatory for all privileged accounts.
- **Credential Lifecycle –** User passwords and API keys are rotated every 180 days. Secrets (API keys, tokens, connection strings) must be stored in AWS Secrets Manager (or an approved vault) and must not reside in code, images, or plain configuration.
- **Credential Sharing –** When sharing credentials is unavoidable, use PasswordState with self-destruct messages for external recipients, or AWS Secrets Manager (managed by the cloud team) for service-to-service sharing.
- **Joiner/Mover/Leaver –** Access grants require managerial approval and are time-bound; access is automatically revoked on role changes and terminations.

3.1.2 Data protection

- **Encryption at rest –** All data is encrypted using AES-256, including backups, snapshots, and storage buckets.
- **Encryption in transit –** Internal service-to-service traffic within private subnets may be left unencrypted only when a documented risk assessment approves it; all external communications use TLS 1.2+.
- **Key Management –** Encryption keys are managed via AWS Key Management Service (KMS), with automated rotation and restricted access.
- **Data Residency** – All customer data and backups are stored exclusively in data centers located within the European Union (EU). No data is processed or stored

outside the EU, ensuring compliance with GDPR and regional data sovereignty requirements.

3.1.3 Threat detection & vulnerability management

- AWS GuardDuty monitors network activity and account behavior for anomalies.
- AWS Inspector performs automated vulnerability assessments.
- AWS WAF protects web applications against common exploits such as SQL injection and cross-site scripting.
- Security Hub aggregates findings across services for centralized visibility and response.

3.1.4 Penetration testing

- Annual external penetration test by a certified third-party firm.
- Scope: infrastructure, network, and application layers.
- Remediation process for findings.
- Collaboration with external auditors under strict controls.

3.1.5 Logging & auditing

- **Auditability –** All API activity is logged via AWS CloudTrail and retained for at least 1 year in a centrally managed, access controlled log archive.
- **Log Retention –** Application and infrastructure logs are retained for a minimum of 7 days.
- **Metrics Retention –** Monitoring metrics (e.g., CloudWatch, Prometheus) are retained for a minimum of 14 days to support trend analysis and incident forensics.
- **Tamper Resistance –** Security and audit logs are replicated to a separate audit account to prevent loss or tampering if a source account is compromised.
- **Anonymization & Data Masking –** Logs must not contain personally identifiable information (PII) or confidential business information in plain text. Where logging of identifiers is unavoidable, data masking techniques (e.g., hashing, tokenization) are applied. Sensitive fields such as names, email addresses, and financial details are anonymized before storage. This ensures compliance with GDPR and internal data protection policies.

3.1.6 AWS Shared Responsibility Model

The 4CEE cloud team adheres to the AWS Shared Responsibility Model, which defines the division of security responsibilities between AWS and the customer:

- **AWS Responsibilities**: Security of the cloud, including physical infrastructure, hardware, networking, and managed services.
- **Customer Responsibilities (4CEE)**: Security in the cloud, including application security, data protection, identity and access management, and compliance with regulatory requirements.
- **Our Approach**:
  - We implement strong IAM, encryption, monitoring, and compliance controls for all workloads.
  - AWS provides certifications (ISO 27001, SOC, etc.) and ensures physical and environmental security.
  - Customers retain responsibility for application-level security and data governance within their SaaS usage.

3.2 Compliance

The central cloud team ensures that all platforms operate in accordance with relevant data protection regulations and industry standards. This includes:

- ISO/IEC 27001 & 27017 certification support.
- Data classification based on the CIP (Centrum Informatiebeveiliging en Privacybescherming) baseline, ensuring appropriate handling of sensitive business and personal data.
- Customer-initiated audits are supported under strict conditions, including scope definition, confidentiality safeguards, and use of accredited auditors. All costs involved are to be paid by the customer.

3.3 Platform-specific controls

While the security framework is centrally managed, each platform may implement additional controls tailored to its architecture:

- Multi-tenant isolation at database and application level.
- Custom authentication mechanisms (e.g., Cognito, SSO, hashed credentials).
- Application-level encryption and access restrictions.

3.4 Security governance

Security policies are reviewed annually and updated in response to emerging threats, platform changes, or regulatory updates. The central cloud team works closely with platform development teams to embed security into the software lifecycle from design to deployment.

- **Multi-account model –** AWS accounts are segmented by purpose (development, staging, production, backup, audit, security) and governed via AWS Organizations and Service Control Policies (SCPs).
- **Resource tagging standard –** All resources must carry standardized tags (e.g., Owner, Costcenter, Environment, Application, Data classification) for cost management, accountability, and policy enforcement.
- **Security replication –** Security findings and logs are replicated to a central audit account for independent visibility and durability.

3.4.1 Security awareness & phishing training

- Yearly awareness training by Security Officer.
- Random phishing simulations.
- Mandatory participation.

3.4.2 Cloud team training & certification

To maintain a high level of operational excellence and security expertise, all Cloud Platform Engineers are encouraged and supported to obtain AWS certifications relevant to their roles. Training is part of the team's continuous development program and ensures alignment with AWS best practices and evolving cloud technologies.

3.4.3 Multi-account strategy (AWS Organizations)

To enforce strong isolation and governance, workloads are segregated using AWS Organizations:

- **Organizational Units (OUs)**
  Separate OUs for **Production**, **Staging**, **Development**, **Security**, **Audit**, and **Backup**.
  Production accounts are **hardened** with stricter Service Control Policies (SCPs).
- **Service Control Policies (SCPs)**
  Deny-by-default for prohibited services, restricted regions, and sensitive APIs; allowlists for approved runtimes and data services. Break-glass is controlled and fully audited.

- **Delegated Administration**
  Central services (e.g., Security Hub, GuardDuty, CloudTrail, Backup) are centrally managed with delegated admin and **cross-account aggregation** for visibility.

4 Monitoring & incident management

The central cloud team ensures the stability, performance, and security of the cloud environment through comprehensive monitoring and a structured incident management process. Monitoring is implemented across multiple layers of the infrastructure and applications to guarantee business continuity and rapid response to potential issues.

4.1 Monitoring approach

Monitoring is applied at the following levels:

- **Application Layer –** All critical services and applications are continuously monitored for availability and performance. If a service becomes unavailable or exhibits abnormal behaviour, alerts are automatically generated for immediate investigation.
- **Infrastructure Layer –** Each server and containerized workload is monitored for key performance indicators, including:
  - **CPU Utilization** – Alerts are triggered when thresholds are exceeded for a defined period.
  - **Memory Usage** – Continuous tracking to prevent resource exhaustion.
  - **Disk Space** – Automatic alerts for low disk space conditions.
  - **Network Traffic** – Monitoring of inbound and outbound traffic to detect anomalies.
- **Security Layer –** Security monitoring is integrated using AWS-native services such as:
  - **AWS GuardDuty** – Continuous threat detection and anomaly analysis.
  - **AWS Inspector** – Automated vulnerability assessments.
  - **AWS Security Hub** – Aggregation and prioritization of security alerts across accounts.
  - **AWS WAF** – Protection against common web exploits.
- **Observability Tools –**
  - **Prometheus & Grafana** – Metrics collection and visualization for containerized environments.
  - **Opensearch** – Centralized log analysis for troubleshooting and root cause identification.

4.2 Incident management

Incidents are managed according to a structured process to minimize impact and restore services quickly:

1. **Detection & Alerting**
   Alerts from monitoring systems are sent via multiple channels (e-mail, Teams, and mobile notifications outside business hours) to ensure timely response.
2. **Initial Assessment**
   The Cloud Team evaluates the severity and scope of the incident, including potential business impact.
3. **Escalation**
   If required, incidents are escalated to the Cloud Team Lead and Delivery Manager for additional resources and communication planning.
4. **Customer Communication**
   In case of business impact, customers are informed promptly via phone and e-mail, with regular updates until resolution.
5. **Resolution & Recovery**
   The team applies predefined remediation steps or executes a disaster recovery plan if necessary. Recovery Point Objective (RPO) and Recovery Time Objective (RTO) targets are adhered to as per SLA.
6. **Post-Incident Review**
   After resolution, a root cause analysis is performed, and corrective actions are documented to prevent recurrence.

4.3 Security Incident

In case an incident occurs in which the availability, confidentiality or integrity of your data is presumably or verifiably affected and this can be related to our services or products, you must report this, preferably within 24 hours of detection.

During office hours: report to our Service Center by phone. Outside office hours (only critical incidents): call our emergency number +31 85 792 92 99

It is important that any evidence (e.g. log files, screenshots, etc.) is secured. These can be requested during the investigation. For other questions about information security, please contact our Security Officer at the email address: security@4cee.com

5 CI/CD & deployment strategy

The Central cloud team implements a robust Continuous Integration and Continuous Deployment (CI/CD) strategy to ensure rapid, reliable, and secure delivery of cloud services and applications. This approach minimizes downtime, reduces risk, and accelerates innovation.

5.1 Core principles

- **Automation first** – All build, test, and deployment processes are automated to eliminate manual errors and ensure consistency.
- **Security by design** – Security checks, vulnerability scans, and compliance validations are integrated into every stage of the pipeline.
- **Zero-downtime deployments** – Rolling updates and blue-green deployments are used to maintain service availability during releases.

5.2 Pipeline overview

Our CI/CD pipeline is built on GitLab CI/CD and integrates with containerization and orchestration technologies for seamless deployments:

5.2.1 Source control & build

- All code is managed in Git repositories.
- Commits trigger automated builds using standardized container images.
- Images are stored in a secure container registry.

5.2.2 Automated testing & quality gates

- Unit, integration, and functional tests are executed automatically.
- SonarQube enforces quality gates for code reliability, maintainability, and security.
- Container and dependency scans detect vulnerabilities early in the process.

5.2.3 Artifact management

- Built images are stored in a software container registry with role-based access control and image signing for integrity.

5.2.4 Deployment orchestration

- Deployments are managed via Kubernetes (EKS) for scalability and resilience.
- Automated deployment jobs handle rolling updates and rollback strategies.
- Infrastructure changes are applied using Infrastructure as Code (IaC) tools such as Terraform and Helm.

5.3 GitOps

GitOps is used for maintaining all software running on the Kubernetes clusters. Gitops is a set of practices that use Git as the single source of truth for both application code and its operational state (infrastructure, configuration, and deployment manifests).

- **Declarative desired state:** All desired configurations (Helm templates) are stored as code in a Git repository.
- **Automated reconciliation:** An operator continuously watches the repo and applies any changes to the target environment, ensuring the live system matches the declared state.
- **Version-controlled operations:** Every change is tracked, reviewed, and can be rolled back through normal Git workflows (pull requests, commits, tags).

By treating infrastructure like any other software artifact, GitOps improves auditability, reproducibility, and collaboration while reducing manual, error-prone steps in deployments.

5.4 Deployment environments

- **Development –** Used for feature testing and integration.
- **Staging –** Mirrors production for final validation.
- **Production –** Highly available, multi-zone AWS environment with automated failover.

5.5 Observability & feedback

- Deployment health is monitored in real-time using dashboards.
- Logs are centralized in Opensearch for troubleshooting and audit purposes.
- Alerts are integrated with collaboration tools (e.g., Microsoft Teams) for rapid response.

5.6 Third-party software governance

Installation and configuration of third-party software is restricted to authorized roles**.** Products must be sourced from trusted repositories, and patching processes (including emergency patching) must be defined and executed by the responsible team. Deployments must use standardized build and deployment pipelines; manual installation on production systems is prohibited.

6 Backup & disaster recovery

The central cloud team ensures data integrity and service continuity through a comprehensive Backup and Disaster Recovery (BDR) strategy. This strategy combines AWS-native services, Kubernetes-level backups, and cross-region replication to minimize data loss and downtime in the event of failures or disasters.

6.1 Backup strategy

6.1.1 AWS backup service

- Centralized management: All backups are managed through AWS Backup, a fully managed service that automates backup scheduling, retention, and cross-region replication.
- Frequency: Backups are performed every 24 hours for all tagged resources.
- Retention: Standard retention is 35 days (with exceptions for specific platforms).
- Cross-Region Redundancy: Backups are copied to a dedicated backup account and replicated to a secondary AWS region for disaster resilience.

6.1.2 Kubernetes cluster backups

- Scope: All Kubernetes-based workloads are protected with backup software, which creates point-in-time snapshots of cluster resources and persistent volumes.
- Frequency: Daily backups at midnight.
- Retention: 35 days.
- Recovery: Snapshots can be restored to the same or a different cluster, ensuring rapid recovery of containerized applications.

6.1.3 Platform-specific enhancements

- Continuous Database Replication: For critical workloads, continuous replication to a failover region ensures near-zero data loss.
- Custom Backup Processes: Certain platforms implement additional database and media backups for enhanced protection.

6.1.4 Backup validation & DR drills

To continuously validate recoverability, the cloud team runs weekly backup tests and DR drills:

- **Weekly Restore Tests**
  Sampled backups (databases, object stores, and kubernetes snapshots) are restored to isolated test accounts/namespaces. Integrity checks verify consistency, decryptability, and application startup.

- **RPO/RTO Verification**
  Each test records achieved **RPO/RTO** vs. targets (e.g., RPO 24h, RTO 48h). Deviations create actions in the backlog and feed into capacity/performance improvements.
- **Evidence & Traceability**
  Every test outputs an evidence bundle (runbook ID, backup IDs, timestamps, logs, screenshots) stored in JIRA for inspection.
- **Yearly DR test**
  Periodic scenario-based DR exercises validate failover runbooks, escalations, and communication plans (customer updates, incident channels).

6.2 Disaster recovery objectives

- Recovery Point Objective (RPO)
  - Standard: 24 hours (data loss limited to the last backup).
  - Enhanced: 12 hours for platforms with continuous replication.
- Recovery Time Objective (RTO)
  - Standard: 48 hours to restore full service availability.
  - Regular updates are provided every 4 hours during recovery.

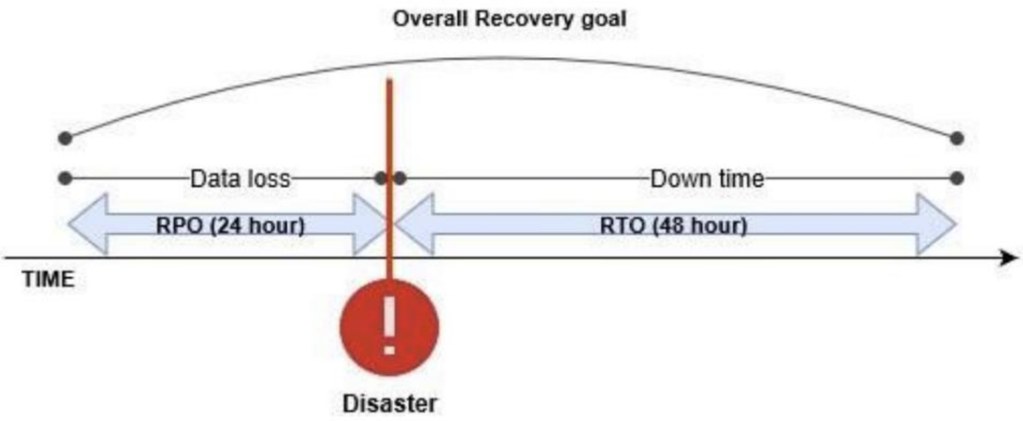

Figure 1 Overall Recovery Goal showing RPO (24h) and RTO (48h) relative to the disaster event.

6.3 Disaster recovery process

6.3.1 Incident detection & assessment

- Automated alerts from monitoring systems trigger the DR process.
- Impact analysis determines the scope and severity of the incident.

6.3.2 Data restoration

- Full backups of databases and server instances are restored from AWS Backup or kubernetes snapshots.
- Verification checks ensure data integrity before services are brought online.

6.3.3 Failover & communication

- If primary region is unavailable, workloads are restored in the secondary region.
- Customers are informed promptly, with continuous updates until resolution.

6.3.4 Post-recovery review

- Root cause analysis and corrective actions are documented to prevent recurrence.

7 Access management & roles

The Central cloud team enforces strict Identity and Access Management (IAM) policies to ensure security, compliance, and operational integrity across all cloud environments. Access is granted on the principle of least privilege, ensuring that users and systems only have the permissions necessary to perform their tasks.

7.1 Access control principles

- **Least Privilege –** Users receive only the permissions required for their role.
- **Role-Based Access Control (RBAC) –** Access is managed through predefined roles aligned with job responsibilities.
- **Segregation of Duties –** Administrative and operational roles are separated to reduce risk.
- **Multi-Factor Authentication (MFA) –** Mandatory for all privileged accounts.

7.2 Role structure

Access is organized into the following role categories:

- **Cloud Platform Engineer**
  - Full administrative access to AWS accounts and Kubernetes clusters for infrastructure provisioning, configuration, and lifecycle management.
  - Responsible for:
    1. Designing and maintaining cloud infrastructure using Infrastructure as Code (IaC) (e.g., Terraform, Helm).
    2. Managing CI/CD pipelines and deployment automation.
    3. Implementing security controls and compliance measures.
    4. Performing incident response and disaster recovery operations.
  - Access to production environments is controlled and audited, with changes executed primarily through automated pipelines.
- **Developers**
  - Access restricted to development and staging environments.
  - Limited access for deployment to production systems.
  - Deployments are handled through CI/CD pipelines.
- **Security Officers**
  - Read-only access to logs, monitoring dashboards, and security alerts.
  - Responsible for compliance audits and incident investigations.

- **Support & Operations**
    - Limited access for troubleshooting use be specific support personnel.

7.3 Access management tools & practices

- **AWS IAM –** Centralized identity and policy management.
- **IAM Roles & Policies –** Fine-grained permissions for services and users.
- **AWS Organizations & SCPs –** Governance across multiple accounts.
- **Kubernetes RBAC –** Namespace-level access control for containerized workloads.
- **Audit & Logging –** All access events are logged in AWS CloudTrail and monitored for anomalies.

7.4 Approval & revocation process

- Access requests require managerial approval.
- Elevated privileges are granted temporarily and revoked after the approved period.
- Access is immediately revoked upon role change or termination.

7.5 Security monitoring

- Continuous monitoring of login attempts and privilege escalations.
- Automated alerts for suspicious activity via AWS GuardDuty and Security Hub.

7.6 Personal screening

- Includes VOG(Certificate of Good Conduct) screening for cloud engineers due to ISO requirements for Stiply.
- NDA requirements embedded in labour and external contracts.
- Access granted only after screening and NDA confirmation.

7.7 Password & key hygiene

User passwords and API keys are rotated every 180 days. Secrets are stored in AWS Secrets Manager or an approved vault. Guest/anonymous access is prohibited. Credential sharing must use the company provided password-vault(self-destruct) or AWS Secrets Manager (Cloud team managed).

8 Data management & contract termination

The central cloud team ensures that customer data is managed securely throughout its lifecycle, from creation and storage to deletion upon contract termination. All processes comply with ISO 27001, ISO 27017, and applicable data protection regulations (e.g., GDPR).

8.1 Data management principles

- **Data Isolation –** Each customer environment is logically and physically isolated at the application and database level.
- **Encryption –**
  - Data at Rest: Encrypted using AES-256 with keys managed by AWS Key Management Service (KMS).
  - Data in Transit: Secured using TLS 1.2 or higher.
- **Access Control –** Only authorized personnel with approved roles can access customer data, following the least privilege principle.
- **Audit & Logging –** All access and changes to data are logged and monitored for compliance and security.

8.2 Data labeling & classification

To ensure proper handling of sensitive information, all data is classified and labelled according to its confidentiality and integrity requirements, following CIP guidelines:

Label	Description	Examples
Public	No restrictions on disclosure	Marketing materials
Internal	Limited to internal use	Internal process docs
Confidential	Sensitive business or personal data	Invoices, contracts
Restricted	Highly sensitive, critical for operations	Financial data, PII

Key Rules:

- Labels must be applied at the data source (e.g., database, document storage).
- Applications must enforce access controls based on classification.
- Confidential and Restricted data must always be encrypted and monitored.

8.3 Data transfer controls

Data exchange to/from customers is limited to Amazon S3 and SFTP (SSH) endpoints. Where MFA is not feasible for automated transfers, credentials must use minimum 12-character passwords and are rotated every 180 days. Encryption in transit is required (TLS 1.2+).

8.4 Data retention & backup

- Daily backups of databases and application data are performed using AWS Backup and software for Kubernetes workloads.
- Backups are retained for 35 days (or as per contractual agreements) and stored in a separate AWS account with cross-region replication for disaster recovery.
- Backup data is encrypted and access-controlled.

8.5 Contract termination process

When a customer terminates their contract, the following steps are executed:

- **Deactivation of Services**
  - Customer environments (tenants, instances, or namespaces) are deactivated and removed from production systems.
- **Data Deletion**
  - All customer data, including databases, file storage (e.g., S3 buckets), and backups, is securely deleted in accordance with retention policies.
  - Deletion is verified and logged for audit purposes.
- **Access Revocation**
  - All user accounts and API keys associated with the customer are revoked immediately.
- **Customer Notification**
  - A confirmation report is provided to the customer, detailing the deletion process and compliance with contractual and regulatory requirements.

8.6 Security & compliance

- No exceptions are made to security standards during termination.
- All processes follow GDPR and internal data classification guidelines to ensure confidentiality and integrity.

9 Glossary of Terms

Amazon Time Sync Service: A time synchronization service using regionally distributed, satellite-connected atomic clocks, available at 169.254.169.123 within every VPC.

AWS Backup: A fully managed service that automates backup scheduling, retention, and cross-region replication.

AWS GuardDuty: A threat detection service that monitors network activity and account behavior for anomalies.

AWS Inspector: A service that performs automated vulnerability assessments.

AWS Key Management Service (KMS): Manages encryption keys with automated rotation and restricted access.

AWS Organizations: A service for managing multiple AWS accounts with governance controls like Service Control Policies (SCPs).

AWS Security Hub: Aggregates security findings across AWS services for centralized visibility and response.

AWS WAF: Web Application Firewall that protects against common exploits like SQL injection and cross-site scripting.

CI/CD (Continuous Integration/Continuous Deployment): A strategy for automated testing, building, and deployment of applications.

Credential Lifecycle: Policy requiring rotation of passwords and API keys every 180 days, with secure storage in approved vaults.

Credential Sharing: Secure sharing of credentials using a corporate password vault solution or AWS Secrets Manager.

Data Residency: Policy ensuring all customer data is stored within the European Union (EU) to comply with GDPR.

Encryption at Rest: Data is encrypted using AES-256, including backups and storage buckets.

Encryption in Transit: TLS 1.2+ is used for external communications; internal traffic may be unencrypted with risk assessment.

Infrastructure as Code (IaC): Managing cloud infrastructure through code (e.g., Terraform, Helm) for repeatability and traceability.

Karpenter: Kubernetes autoscaler that provisions right-sized nodes on demand based on pod scheduling needs.

Least Privilege & RBAC: Access control model granting minimum required permissions via IAM roles.

MFA (Multi-Factor Authentication): Mandatory for all privileged accounts and break-glass procedures.

PII: Personally Identifiable Information

Prometheus & Grafana: Tools for metrics collection and visualization in containerized environments.

Recovery Point Objective (RPO): Maximum acceptable amount of data loss measured in time (e.g., 24 hours).

Recovery Time Objective (RTO): Maximum acceptable time to restore services after a disaster (e.g., 48 hours).

Secrets Management: Secure storage of sensitive credentials in AWS Secrets Manager or approved vaults.

Service Control Policies (SCPs): Governance rules applied to AWS accounts to restrict services, regions, and APIs.

Zero-Downtime Deployments: Deployment strategy using rolling updates or blue-green deployments to maintain availability.