ERP System Security Design and Implementation
Cybersecurity Team

Table of Content

List of Tables

1. Project Overview

1.1 Objective

The main goal of the Cybersecurity track is to secure the ERP system being developed for Konecta.
Our team focuses on protecting sensitive business data, enforcing proper access control, and ensuring the system complies with security and privacy standards.
This includes defining how users access the system, setting up authentication mechanisms, establishing clear response plans for potential incidents, and working closely with other teams to keep every ERP module safe and reliable.

1.2 Scope

The Cybersecurity team's work covers all aspects related to the confidentiality, integrity, and availability of the ERP system.
Our key responsibilities include:

- Designing and enforcing Role-Based Access Control (RBAC).
- Implementing secure authentication mechanisms such as Multi-Factor Authentication (MFA) and Single Sign-On (SSO).
- Creating and documenting an incident response plan for ERP-related threats.
- Ensuring compliance with data protection and cybersecurity regulations (GDPR, ISO 27001, etc.).
- Coordinating with Full Stack, Cloud, and Project Management teams to align technical and security needs.

Week 1
Initiation & Foundation

During the first week of the ERP project, our team focuses on establishing the security foundation for the system.
This stage is critical, as it defines the principles and controls that will protect ERP system throughout its development and deployment.

2.1 ERP Security Requirements

1. Access Control (RBAC)

- The ERP must implement Role-Based Access Control (RBAC), ensuring each user only has permissions necessary for their job.
- Roles (e.g., CFO, Finance Manager, Accountant) should have predefined access permissions (view, edit, approve, admin).
- Access reviews must be conducted regularly to identify and revoke privileges that are no longer required.

2. Authentication

- Enforce Multi-Factor Authentication (MFA) for all users.
- Integrate Single Sign-On (SSO) where possible to streamline secure authentication and reduce password fatigue.
- Adopt strong password policies, including minimum complexity and periodic expiration.

3. Data Encryption

- Apply end-to-end encryption to all sensitive data, both at rest and in transit.
- Use AES-256 for stored data and TLS 1.3 for communications between modules and user interfaces.
- Encryption keys must be securely managed, rotated periodically, and stored using hardware security modules (HSM) or equivalent mechanisms.

4. Audit Logging

- Record all major security events such as logins, access changes, data modifications, and approval actions.
- Ensure logs are tamper-proof, timestamped, and securely stored for compliance and forensic purposes.
- Only authorized cybersecurity and audit personnel should have access to log data.

5. Incident Response

- Develop a dedicated Incident Response Plan (IRP) for ERP-related cybersecurity incidents.
- Define steps for detection, reporting, containment, and recovery**,** including escalation procedures.
- Conduct simulations to ensure readiness across the cybersecurity and IT teams

6. Compliance and Privacy

- Ensure the ERP adheres to international standards such as ISO/IEC 27001**,** GDPR, and internal company security policies.
- Personal and financial data must be collected, processed, and stored lawfully, with strict access limitations.
- Implement data retention and deletion policies in accordance with compliance regulations.

6.1 Data Retention and Deletion Policy

To ensure that all data stored within the ERP system, including financial, HR, and operational records, is retained only for as long as it is required for business, legal, or regulatory purposes, and securely deleted when no longer needed.

Policy Details:

6.1.1 Retention Periods:

- **Financial Records:** Retained for 7 years to comply with accounting and auditing regulations.
- **HR and Payroll Data:** Retained for 5 years after employee departure, unless extended for legal obligations.
- **Operational and Log Data:** Retained for 180 days**,** unless flagged for investigation or compliance review.
- **Backup Data:** Retained for 90 days, after which backups are automatically overwritten.

6.1.2 Secure Deletion:

- When data reaches the end of its retention period, it must be securely deleted or anonymized using industry- standard techniques such as data wiping or cryptographic erasure.
- Deletion processes must ensure data can't be recovered from system storage or backups.

6.1.3 Access Restriction:

- Only authorized personnel (Cybersecurity, IT, and Compliance teams) can approve and execute data deletion operations.
- All deletion actions must be logged for audit and verification purposes.

6.1.4 Compliance Requirements:

- The retention and deletion schedule must comply with GDPR Article 5(1)(e) (storage limitation) and ISO/IEC 27001:2022 control 8.11 (data deletion).
- Periodic reviews will ensure that retention durations remain aligned with updated legal or company policies.

6.1.5 Exceptions:

- Data involved in an ongoing investigation, audit, or litigation must be preserved until the issue is formally resolved.

6.2 Regional Data Protection and Deployment Policy

Since this ERP system is intended for deployment in the **European region**, all data storage and processing must comply with **EU data protection laws**, including the **General Data Protection Regulation (GDPR)**.

To ensure compliance and protect user privacy:

- **Data Hosting:** All ERP data must be stored within the European Economic Area (EEA), preferably in EU-based cloud regions (e.g., Azure West Europe or AWS Frankfurt).
- **Deployment Regions:** The system will be deployed across two European regions to ensure high availability and disaster recovery readiness:
  - Primary Region: West Europe (Netherlands) – main hosting location for ERP operations.
  - Secondary Region: North Europe (Ireland) – designated for backup, failover, and disaster recovery.
- **Data Transfers:** No personal or financial data may be transferred outside the EEA unless compliant safeguards (such as Standard Contractual Clauses) are in place.
- **Compliance Monitoring:** We will work with the cloud and IT teams to ensure hosting environments meet GDPR and corporate data governance standards

7. Vulnerability Management

- Perform regular vulnerability scans and penetration testing to identify and remediate potential risks.
- Document discovered vulnerabilities, assign severity ratings, and ensure timely patching.
- Maintain a vulnerability management log for traceability and continuous improvement.

8. System Monitoring

- Enable real-time system monitoring to detect unusual or unauthorized activity.
- Integrate ERP logs with a Security Information and Event Management (SIEM) platform for centralized visibility.
- Configure automated alerts for high-risk events such as failed login attempts or privilege escalations.

2.2 Role-Based Access Guidelines

This matrix outlines the core security requirements and controls that must be implemented across all ERP modules. Each control area is mapped to its corresponding responsible team to ensure accountability and alignment with organizational security standards.

Table 2.2 Role-Based Access Guidelines

	A	B
1	Area	Guideline
2	User Authentication	Enforce multi-factor authentication (MFA) for all privileged accounts.
3	Data Confidentiality	Sensitive data (finance, payroll, personal info) must be encrypted at rest and in transit.
4	Access Requests	All new access must be approved by module head and IT security officer.
5	Access Revocation	Terminated or transferred employees' access should be removed immediately.
6	Audit Trails	Maintain audit logs for all access and modification actions.
7	Periodic Access Review	Quarterly review of user roles and permissions.
8	Least Privilege Enforcemer	Each user must justify their access scope during onboarding.

2.3 Compliance Checklist

The following matrix defines the roles, responsibilities, and corresponding access privileges within the ERP system. It ensures that each role operates under the principle of least privilege**,** maintaining proper segregation of duties and secure access to sensitive data.

Table 2.3 Compliance Checklist

	A	B	C	D	E
1	Control Area	Requirement	Responsible Track	Compliance Status	Evidence / Notes
2	Access Control	Role-Based Access Control (RBAC) implemented for all ERP modules	Cybersecurity / Full Stack	FALSE	Confirm roles (Admin, Finance, HR, Auditor, etc.) configured
3	Authentication	Multi-Factor Authentication (MFA) enabled for all users	Cybersecurity	FALSE	Test login flow with MFA
4	Authorization Policy	Access granted using least privilege principle	Cybersecurity	FALSE	Review role matrix & permissions
5	User Management	Unique user IDs; no shared accounts	Full Stack / Cybersecurity	FALSE	Validate ERP user database
6	Audit Logging	All logins, approvals, and financial edits logged and retained	DevOps / Cybersecurity	FALSE	Verify log storage and retention policy
7	Data Encryption (At Rest)	Sensitive data (finance, payroll, HR) encrypted using AES-256	Cloud / Cybersecurity	FALSE	Confirm encryption in database or storage
8	Data Encryption (In Transit)	TLS 1.2+ enforced for all network communications	Cloud / DevOps	FALSE	Verify HTTPS and API security
9	Incident Response	ERP-specific incident response plan documented and tested	Cybersecurity	FALSE	Check incident playbook and test logs
10	Backup & Disaster Recovery	Regular backups with multi-region redundancy	Cloud	FALSE	Validate recovery procedure and test restores
11	Patch Management	ERP and server OS updates applied within 30 days of release	Cloud / DevOps	FALSE	Check patch/update logs
12	Segregation of Duties (SoD)	Financial entry and approval roles separated	Finance / Cybersecurity	FALSE	Review workflow permissions
13	Data Privacy & Compliance	Align with GDPR / local data protection laws	Cybersecurity / HR / Finance	FALSE	Confirm consent and data handling policies
14	Monitoring & Alerts	Real-time alerts for login anomalies or failed access	DevOps / Cybersecurity	FALSE	Review monitoring dashboard setup
15	AI Model Security	Validate explainability and ensure no sensitive data leakage in AI models	AI / Cybersecurity	FALSE	Review SHAP/LIME output, data anonymization
16	Cloud Security	IAM policies and VPC isolation applied for ERP services	Cloud / Cybersecurity	FALSE	Inspect IAM configuration and subnet segmentation
17	API Security	Secure authentication (JWT/OAuth2), rate limiting applied	Full Stack / DevOps	FALSE	Test API gateway for security headers
18	Data Retention Policy	Define retention duration for HR and financial data	HR / Finance / Cybersecurity	FALSE	Document retention and deletion timelines
19	User Awareness Training	ERP security training provided for all users	HR / Cybersecurity	FALSE	Record attendance and training materials
20	Compliance Review Cycle	Semi-annual access and compliance audits planned	Cybersecurity / PM	FALSE	Document audit schedule
21	Third-Party Integration Security	Validate vendor systems (OCR, cloud AI APIs) for compliance	AI / Cloud / Cybersecurity	FALSE	Confirm API contracts and security certifications

Week 2
Architecture & Infrastructure

The goal of this week is to define who can access what within the ERP system and what actions they are allowed to perform, following the principle of least privilege**.** This policy forms the foundation for secure role-based access control (RBAC) and will be used by both the Cloud and Development teams to implement permissions inside the system and hosting environment.

3.1 ERP Access Control Policy

1. Scope

This policy applies to all ERP modules being developed, including:

- Finance
- Human Resources (HR)
- Operations
- Sales
- IT & Infrastructure
- Analytics / Reporting

Each role is mapped according to its business responsibility**,** data sensitivity**,** and access necessity**.**

2. Access Control Principles

- **Least Privilege:** Users are granted the minimum level of access needed for their job.
- **Segregation of Duties:** No single user should have full control over a complete process (e.g., initiating and approving payments).
- **Role-Based Access Control (RBAC):** Permissions are tied to roles, not individual users.
- **Periodic Access Reviews:** Access rights will be reviewed regularly to ensure ongoing compliance.

3. ERP User Roles and Access Matrix

This matrix provides a detailed mapping of ERP user roles, their corresponding responsibilities, and the permissions granted to each within the system.
It serves as the foundation for implementing Role-Based Access Control (RBAC) and ensures that access privileges are assigned according to each user's business function and data sensitivity level.
By clearly defining access boundaries, this matrix supports the principle of least privilege and strengthens the overall security posture of the ERP environment.

Table 3.1 ERP Access Matrix

	A	B	C	D	E
1	**Role**	**Responsibilities / Description**	**Modules Accessed**	**Access Type**	**Notes**
2	Chief Financial Officer (CFO)	Oversees all financial operations and reporting.	Finance, Analytics	View / Edit / Approve / Admin	Full access to financial data for strategic decision-making.
3	Finance Manager / Controller	Manages accounting operations, budgets, and compliance.	Finance, Analytics	View / Edit / Approve	Can approve transactions and adjust budgets.
4	Accountant	Records journal entries, handles ledgers, and reconciles accounts.	Finance	View / Edit	Can input and adjust records but cannot approve.
5	Accounts Payable Clerk	Manages vendor invoices and outgoing payments.	Finance	View / Edit (Payables Only)	Limited to processing supplier invoices.
6	Accounts Receivable Clerk	Manages customer invoices and incoming payments.	Finance	View / Edit (Receivables Only)	Can record customer payments and update balances.
7	Payroll Officer / HR-Finance Coordinator	Processes employee payroll and benefits.	HR, Finance	View / Edit (Payroll Only)	Handles payroll data securely; cannot modify finance records.
8	HR Officer	Manages employee data, recruitment, and training workflows.	HR	View / Edit	Access limited to HR records; cannot view financial data.
9	HR Manager	Oversees HR operations, training, and performance evaluation.	HR, Analytics	View / Edit / Approve	Approves HR actions such as promotions and training.
10	Procurement Officer	Handles purchase orders and supplier coordination.	Operations, Finance	View (Finance) / Edit (Operatic	Can view budget status before purchases.
11	Operations Manager	Oversees daily operational tasks and logistics.	Operations, Analytics	View / Edit / Approve	Approves inventory requests and monitors performance.
12	Sales Manager	Manages sales operations, pipelines, and client accounts.	Sales, Analytics	View / Edit / Approve	Reviews and approves sales reports and targets.
13	Data Analyst	Develops analytical dashboards and reports for management.	Analytics	View / Export	Can analyze data but cannot alter source transactions.
14	IT Support / System Engineer	Provides technical support, manages integrations, and troubleshooting.	IT, System Settings	Edit / Limited Admin	Handles system configurations but not business data.
15	System Administrator (Cybersecurity / Clou	Manages ERP infrastructure, access control, and configurations.	All Modules (Technical)	Admin (System-Level)	Responsible for permissions, IAM, and security enforcement.
16	Project Manager / Department Head	Oversees team performance, budgets, and department KPIs.	Own Department Data, Analyt	View Only	Can monitor reports but cannot edit records.
17	Internal Auditor	Audits transactions and ensures compliance.	Finance, HR, Logs	View Only	Read-only access to all audit trails and reports.
18	General Employee / End User	Submits requests (leave, expense claims, etc.).	HR, Operations	Limited View / Submit Forms	Can view personal data and submit requests only.

4. Administrative Controls

- **Access Provisioning:** All access requests must be approved by both HR and we, the cybersecurity team, before activation.
- **Access Revocation:** Accounts will be immediately disabled upon termination or role change.
- **Privilege Review:** Access levels will be reviewed quarterly by the cybersecurity and project management teams.
- **Audit Trails:** All login and access changes will be logged for auditing purposes.

3.2 Access and Authentication Requirements

1. Multi-Factor Authentication (MFA) Requirements

To strengthen account security and minimize unauthorized access risks, the following roles and user groups must have MFA enabled at all times:

- System Administrators: Full access to infrastructure and configurations.
- Finance Staff: Handle sensitive financial and payroll data.
- Human Resources (HR): Manage employee records, PII, and internal documentation.
- Developers and DevOps Engineers: Access production environments, source repositories, and deployment pipelines.
- External Contractors: Temporary or limited-access users must use MFA for all logins.

MFA Method Preference:

- Primary: Authenticator apps (Microsoft Authenticator, Google Authenticator).
- Secondary: Hardware tokens (for privileged accounts only).
- SMS-based MFA is not recommended due to interception risks.

2. Single Sign-On (SSO) Provider Preference

To improve centralized identity management and reduce password fatigue:

- Preferred SSO Provider: Azure Active Directory (Azure AD), already integrated with other enterprise systems.
- Alternative Options (if project constraints apply):
    - Okta – if integration with legacy or external SaaS tools is required.
    - Google Workspace – for shared team collaboration environments.

SSO must enforce MFA for all critical roles during sign-in and support conditional access policies (e.g., device trust, location-based restrictions).

3. Security Policy Enforcement

To ensure consistent protection across all accounts and sessions, the following security policies are required:

a. Password Policy:

- Minimum length: 12 characters.
- Must include uppercase, lowercase, number, and special character.
- Password expiration: Every 90 days.
- Prohibit reuse of last 5 passwords.

b. Session Policy:

- Automatic logout after 15 minutes of inactivity for privileged accounts, 30 minutes for standard users.
- Session timeout enforced across all web and API interfaces.
- Device re-authentication required after session expiration.

c. Account Lockout Policy:

- Lock account after 5 failed login attempts within 10 minutes.
- Lockout duration: 15 minutes, with alert to security monitoring system.

d. Access Review and Logging:

- Quarterly access reviews for all roles with elevated privileges.
- Authentication and session logs must be retained for at least 180 days for audit purposes.

4. Implementation Notes (for Cloud Team Review)

- Integrate Azure AD SSO with MFA enforcement into ERP and internal apps.
- Apply conditional access rules for admins logging in from external networks.
- Ensure centralized logging through SIEM for authentication events.
- Test MFA failover and recovery procedures before production rollout.

Week 3
Security, Integration & Intelligence

The goal of this week is to implement and validate the ERP system's core security structure**,** ensuring that all access controls, detection mechanisms, and response strategies are clearly defined and effectively protect the system from potential threats. During this stage, our role as the cybersecurity team focused on two main objectives: first, to validate and test the Role-Based Access Control (RBAC) framework, confirming that user permissions and privileges are properly enforced according to the established security policies; and second, to develop a comprehensive Incident Response Plan (IRP) that outlines the steps for identifying, responding to, and recovering from security incidents within the ERP environment. These efforts ensure that the ERP system is not only secure by design but also resilient under attack**,** capable of minimizing operational disruptions and maintaining full compliance with GDPR and ISO 27001 security standards.

4.1 Implement Role-Based Access Control (RBAC) Structures within the ERP

Goal: Define who can access what inside the ERP to ensure secure and auditable access.

Table 4.1 RBAC Structure

ERP Role	**Description**	**Access Level**	**Accessible Modules**	**Implementation Notes**
Admin	System administrator responsible for configuration and monitoring	Full access	All modules (Finance, HR, Analytics, DevOps, Cloud)	Configure via Keycloak or Auth service using JWT; MFA required
Finance Manager	Manages all financial operations	Edit/View	Finance, Reports	Apply RBAC using role ID FIN_MANAGER; restrict payroll visibility
HR Officer	Manages employee records and attendance	Edit/View	HR, Reports	Mask PII; allow access to employee lifecycle only
Project Manager	Tracks deliverables and timelines	View/Edit limited	Project Mgmt, Reports	Read-only to Finance; no HR data

DevOps Engineer	Maintains pipelines and deployment	Edit	DevOps, Cloud, Logs	Restrict data modification; ensure audit logging
Employee/User	Regular ERP end-user	View only	Self-service (profile, requests)	JWT token validation; MFA optional

Implementation Guide for Full Stack Team:

- Use Spring Security + Keycloak for token-based RBAC.
- Define all roles in the User & Role Management Service database seed file.
- Add JWT interceptors in Angular frontend to protect module routes.
- Enforce MFA for Admins and Finance roles via SSO (Google Workspace or Azure AD).
- Maintain an access control JSON file mapping roles → modules for testing.
- Include audit trails (user_id, timestamp, module_accessed) for compliance.

4.2 Penetration Test Simulations

This plan defines the approach, scope, rules, and deliverables for non-destructive penetration test simulations to be executed against the Konecta ERP staging environment. The assessment will validate authentication, authorization (RBAC), API security, and infrastructure configuration, identify vulnerabilities, and provide prioritized remediation guidance in alignment with GDPR and ISO 27001.

4.2.1 Objective

The objective of the penetration test simulations is to evaluate the security posture of the Konecta ERP staging environment by identifying vulnerabilities in authentication, authorization, API, and infrastructure layers. Validate controls such as RBAC, MFA, and SSO, and provide actionable remediation and retest criteria to ensure resilience and regulatory compliance.

4.2.2 Scope

In scope

- Staging web UI and API gateway (all endpoints used by ERP modules: Finance, HR, Operations, Sales, Analytics).
- Authentication endpoints and token flows (/api/auth/*).

- Role-based features and admin interfaces.
- Audit/logging verification (read-only logs or exported log samples).
- Infrastructure configuration checks for staging cloud regions (EU-based).

Out of Scope

- Production environments and production data.
- Destructive tests (DB deletes, destructive writes) and full DDoS attacks.
- Social engineering (phishing) without separate approval.
- Tests on third-party vendor infrastructure unless vendor approval obtained.

4.2.3 Assumptions & Prerequisites

- Full Stack and Cloud teams will provide: staging URLs, OpenAPI/Swagger or Postman collection, and test user accounts covering all roles.
- Recent backups exist and rollback procedures are confirmed prior to testing.
- Signed Rules of Engagement (RoE) and approval from Project Manager and Cloud lead.
- Tests executed in agreed time window and monitored by Cloud/DevOps teams.

Current System Access Control State (Critical Assumption)

At the time of preparing this penetration testing plan, the ERP system contains only a single administrator account. No additional user roles (Finance, HR, Accountant, Employee, etc.) have been implemented in the current build. Although RBAC is defined in the system design, it is not yet implemented in the actual application**,** resulting in all authenticated access being granted full administrative privileges.

This limitation directly impacts authorization testing and is documented as a Critical security issue in accordance with OWASP A01:2021 — Broken Access Control.

4.2.4 Rules of Engagement (RoE)

a. Objectives & Success Criteria

- Comprehensive security validation of ERP modules during development
- OWASP Top 10 coverage for all authentication and data processing flows
- Zero production impact or development disruption

b. Scope & Schedule

- **Primary:** Staging environment (endpoints to be provided by Full Stack team via OpenAPI)
- **Secondary:** Security-test-ready development branches
- **Testing Windows:** To be scheduled 48 hours after receipt of complete test enviro nment access
- **Excluded:** Production systems, live data, third-party services

c. Authorized Personnel

- **Cybersecurity Team:** Haneen Amr, Marwan Ahmed, Tarek Khalid.

d. Approved Testing Activities

- Authentication/authorization testing across all user roles
- Input validation testing (SQLi, XSS, XXE, Command Injection)
- API security testing for all ERP modules
- Session management and access control testing
- Safe fuzzing (<1000 req/min, no resource exhaustion)

e. Strictly Prohibited Actions

- Production system access or data modification
- No denial-of-service testing or resource exhaustion
- No database destruction or schema modification
- No social engineering against team members
- No testing of excluded systems or endpoints

f. Communication Protocol

- **Critical findings:** Report immediately by phone and post to #security-alerts. Notify Full Stack and Cloud leads.
- **High findings:** Report within 4 hours via Slack and create a ticket in the tracking system.
- **Medium/Low findings**: Document in the final report and include remediation timeline.
- On-call contacts and 24/7 phone numbers will be provided before the test window.

4.2.5 Test Methodology & Approach

Approach: Hybrid gray-box preferred (authenticated accounts + limited design knowledge).

Standards & references: OWASP Top 10 2011, OWASP API Security Top 10, NIST SP 800-115, ISO/IEC 27001:2022 Annex A controls, GDPR Article 32, ENISA Cloud Security Guide

Phases: Reconnaissance → Authentication testing → Authorization testing (RBAC) → Input validation & injection checks → Configuration & transport checks → Post-exploit (non-destructive) analysis → Reporting.

4.2.6 Pre-Test Requirements

1. Staging URLs (UI + API).
2. Swagger/OpenAPI or Postman collection.
3. Test accounts for each role (CFO, Finance Manager, Accountant, HR, Admin, Auditor, General User).
4. Example JWT tokens or instructions to get tokens programmatically.
5. Read-only access or exported samples for logs/audit trails.

4.2.7 Tools & Environment

Dynamic Testing & Service Enumeration

- **tcp-probe**: service reachability and port behavior verification
- **ffuf:** endpoint and directory fuzzing
- **Nikto:** web server vulnerability scanning
- **Docker CLI:** container interaction, log inspection, and service-level testing
- **netcat (nc):** manual TCP interaction and banner grabbing
- **psql:** direct database connection testing and query validation

Database interaction was limited due to the absence of actual records in the staging environment.

Static Testing & Configuration Analysis

- **Gitleaks:** secret scanning in source code
- **Semgrep:** static analysis for insecure patterns and code smells
- **Snyk:** dependency vulnerability assessment and SCA (Software Composition Analysis)

Reporting & Evidence Collection

- Raw outputs stored as JSON files
- JSON results converted into Excel sheets for structured reporting
- Screenshots, logs, and artifacts archived for evidence tracking

4.2.8 Evidence Collection & Reporting

Captured Evidence Included:

- Raw tool outputs in JSON format (Semgrep, Gitleaks, Snyk, ffuf, Nikto).
- Terminal outputs for dynamic testing tools (netcat, tcp-probe, Docker CLI, psql).
- Converted Excel/CSV sheets summarizing findings from the JSON files.
- Notes on environment observations, including empty database state and missing RBAC

Evidence Storage & Naming:

- Files stored using simple naming such as: finding_<ID>.json, scan_<tool>.log, report_<tool>.xlsx
- Evidence maintained in a structured folder for review.

Access Control:

- Evidence is accessible only by the Cybersecurity Lead and Project Manager for analysis.

4.2.9 Test Case Matrix

The table below is a standard, pre-test template used to plan penetration test simulations. It contains example test cases and the structure that will be used during execution. All rows marked “Pending” will be replaced or populated with actual endpoints, payloads, and credentials after the Full Stack team provides the OpenAPI/Postman collection and staging access. Each executed case will be updated with evidence, outcome, and retest criteria.

Table 4.2.9 Test Case Matrix

ID	Role	Endpoint	Method	Test Description	Input/ Payload	Expected Result	Severity
TC-01	CFO	/api/finance/transactions/{id}/approve	POST	Attempt approve using Accountant token	{}	403 Forbidden	High
TC-02	Accountant	/api/finance/transcations	POST	Create transaction	Valid JSON	201 created	Low
TC-03	General user	/api/hr/employees/{id}	GET	Attempt to access another employee (IDOR)	Change id	403/404	High
TC-04	Any	/api/auth/login	POST	Rate-limit / brute force test (controlled)	Many wrong passwords	Rate limit or account lock	Medium
TC-05	Admin	/api/admin/users	GET	Ensure admin-only endpoint inaccessible to normal users	-	403 Forbidden	High

TC-06	Any	/api/upload	POST	Upload file containing script (XSS)	<script>...	Sanitized / rejected	Medium
TC-07	Any	Any	GET	Check for stack traces / sensitive info in error responses	Cause controlled error	No internal traces	Medium
TC-08	Any	/api/refresh	POST	Tamper JWT signature or claims	Altered token	401 unauthorized	High
TC-09	Any	/api/reports/export	GET	Large export requests	Wide date range	Alert / logged rate-limited	High
TC-10	Any	/	-	Verify HTTP security headers present	-	Headers present	Low

4.2.10 Severity Classification & Remediation Timeline

- **Critical:** System compromise, PII exfiltration, RCE → Remediate ASAP (24–72 hours)**.**
- **High:** Broken access control, auth bypass, large data exposure → Remediate within 7 days**.**
- **Medium:** Injection, info leakage, missing headers → Remediate within 30 days**.**
- **Low:** Minor misconfigurations or non-sensitive info → Address in regular release**.**

4.2.11 Deliverables & Timeline

- **Executive Summary:** 3 business days post-testing
- **Technical Findings Report:** 5 business days (with CVSS 3.1 scores)
- **Retesting Validation:** Within 10 business days of remediation
- **Final Sign-off:** Upon closure of all Critical/High findings

4.3 Incident Response Plan (IRP)

4.3.1 Purpose and Scope

This Incident Response Plan (IRP) defines the process for identifying, managing, and recovering from cybersecurity incidents affecting the Konecta ERP system. It applies to all teams and modules within the ERP (Finance, HR, Cloud, AI, DevOps, etc.) and ensures minimal impact on operations, data integrity, and confidentiality.

4.3.2 Objectives

- Quickly detect and respond to security incidents.
- Limit the damage and recover normal operations rapidly.
- Maintain stakeholder trust through transparent and timely communication.
- Ensure compliance with data protection regulations (e.g., GDPR, Egyptian PDPL).
- Document lessons learned to strengthen future resilience.

4.3.3 Roles and Responsibilities

Table 4.3.3 Roles and Responsibilities

Role	**Responsibilities**
Incident Response Lead (Cybersecurity Track)	Coordinate overall response efforts, classify severity, and lead containment and recovery.
Cloud Team	Secure affected cloud environments, perform backups, and restore services.
DevOps Team	Isolate compromised systems, redeploy from clean builds, and validate CI/CD integrity.
Full Stack Development Team	Patch vulnerable modules, fix authentication flaws, and validate application logic.
AI & Data Analytics Teams	Verify integrity of datasets and AI models; ensure no data poisoning or unauthorized model access.
Project Management (PM)	Manage communication flow, ensure documentation, and escalate to stakeholders.
Human Resources & Digital Marketing	Coordinate user notifications, awareness messages, and internal communication.

4.3.4 Incident Categories

- **Unauthorized Access**: Compromised admin credentials, privilege escalation, or session hijacking.
- **Data Breach:** Exposure of financial or HR data, accidental data sharing
- **Malware/Ransomware:** Infection via attachments or vulnerable ERP module.
- **Denial of Service (DoS/DDoS):** Overwhelming cloud instances causing ERP downtime.

- **Insider Threats:** Data deletion, unauthorized information sharing.
- **System Misconfiguration:** Insecure APIs, weak access controls, or open storage buckets.
- **Phishing or Social Engineering:** Deceptive emails or messages tricking users into revealing credentials or clicking malicious links.
- **Third-Party or API Breach:** Compromise through external integrations such as payment gateways, HR platforms, or analytics tools.

4.3.5 Incident Response Phases

Phase 1: Preparation

- Maintain updated system inventory and data classification.
- Enforce MFA, SSO, and least-privilege RBAC policies.
- Conduct regular vulnerability assessments and penetration tests.
- Backup ERP data daily across multiple cloud regions.
- Maintain a Cyber Incident Log template for rapid reporting.
- Conduct periodic security awareness and phishing simulation training for employees.

Phase 2: Identification

- Continuous monitoring via cloud dashboards, SIEM logs, and DevOps pipeline for anomalies.
- Correlate multiple data sources (logs, IDS, endpoint agents) to validate incidents.
- Classify incidents as Low, Medium, or High severity based on business impact and data exposure.
- Example triggers: unusual admin logins, large data exports, multiple failed MFA attempts, or unexpected code deployments.
- Escalate confirmed incidents to the Incident Response Lead and record in the central tracking system.

Phase 3: Containment

- **Short-term:** Isolate affected user accounts, instances, or network segments.
- **Long-term:** Patch vulnerabilities, revoke compromised keys, and rotate credentials.
- Restrict information sharing to authorized internal channels only; any external communication must be approved by the Project Manager or DPO.
- Notify relevant technical teams for interdependent module containment actions.

Phase 4: Eradication

- Remove malicious code, unauthorized scripts, or backdoors from ERP environments.
- Reimage infected virtual machines and validate through clean deployment builds.
- Run post-cleanup scans using EDR tools and vulnerability management tools.
- Validate that root cause is eliminated.

Phase 5: Recovery

- Restore ERP services from verified, uncompromised backups.
- Gradually reconnect affected modules (Finance → HR → Analytics → AI).
- Validate system functionality with key business users to ensure operational readiness.
- Monitor systems closely for anomalies post-restoration (minimum 72 hours).
- Communicate recovery status and timelines to stakeholders when full recovery is achieved.

Phase 6: Lessons Learned

- Conduct a full post-incident review session with all involved teams.
- Document incident timeline, root cause, response effectiveness, and corrective actions.
- Measure incident handling metrics, including Mean Time to Detect (MTTD) and Mean Time to Respond (MTTR).
- Update ERP security policies, training materials, and DevOps pipelines

4.3.6 Communication and Escalation

Table 4.3.6 Communication and Escalation

Severity	Notification Timeline	Communication Method
Low	Within 24 hours	Internal ticket/email
Medium	Within 4 hours	Team Slack/Teams + PM update
High	Immediately	All-team alert + management escalation

4.3.7 Compliance and Documentation

- Maintain all logs, alerts, and communication records for at least 6 months.
- Follow Konecta's Data Protection and Privacy Policy.
- Perform bi-weekly tabletop simulations of incident scenarios.

- Ensure alignment with ISO 27001, NIST SP 800-61, and GDPR best practices.

4.3.8 Post-Incident Security Controls

- Re-evaluate RBAC and SSO settings.
- Perform integrity checks on AI and analytics outputs.
- Automate patch management in CI/CD pipelines.
- Conduct security awareness training for all users.
- Review and update this IRP quarterly.
- Perform compliance audit to confirm remediation aligns with GDPR and internal policy.

Week 4
Intelligence & Visualization

5.1 ERP Security Validation Plan

5.1.1 Objective

The objective of this validation plan is to ensure that the Konecta ERP system correctly implements robust authentication and access-control mechanisms. Specifically, it will verify Multi-Factor Authentication (MFA), Single Sign-On (SSO), and Role-Based Access Control (RBAC) across all ERP modules. MFA adds an extra layer of security by requiring users to provide two or more forms of verification[7]. SSO allows users to authenticate once via a corporate identity provider (e.g. Azure AD or Okta) and access the ERP without additional logins[8]. RBAC enforces least-privilege by assigning permissions based on roles, limiting each user to only the functions needed for their role[9]. This plan's tests and checklists will systematically confirm that these mechanisms work as expected in every module.

5.1.2 Security Test Scripts

Each test scenario in the table below targets a specific security function (authentication or access control). For example, RBAC scenarios verify that users can only access modules permitted by their role[9]. Other scenarios validate the MFA and SSO login flows, confirming that valid multi-factor or SSO credentials are accepted while invalid ones are rejected[7][8]. The table lists each test by ID, describes the scenario, and states the expected result.

Table 5.1.2 Security Tests

Test ID	Scenario	Expected Result
TS-1	Attempt login with valid username/password and complete MFA with a correct code/token.	Login succeeds and user is granted access according to their role.
TS-2	Attempt login with valid username but incorrect password.	Login fails with an "Invalid credentials" error; no access is granted.
TS-3	Enter correct username/password but skip or fail the MFA step (no code provided).	Login is blocked; user is prompted for MFA or shown an error.
TS-4	Enter correct credentials and provide an incorrect or expired MFA code.	Login fails; user is informed of invalid MFA and denied access.
TS-5	Log in via SSO using a valid corporate account (identity provider).	SSO succeeds; user is authenticated by the IdP and redirected into the ERP without additional login.
TS-6	Attempt SSO login with an invalid or disabled corporate account.	SSO fails; access is denied and an error or alternate login prompt appears.

TS-7	User with a limited role (e.g. “Sales”) attempts to access a restricted module (e.g. Finance).	Access is denied; an “Unauthorized” or “Access Denied” message is displayed.
TS-8	User with the appropriate role (e.g. “Sales”) accesses a permitted module (e.g. Sales or Inventory).	Access is granted; user can view/use the functionality allowed for their role.
TS-9	Administrator role user attempts an administrative action (e.g. modify user roles).	Action succeeds; admin is allowed to access and modify settings as per the admin role.
TS-10	Change a user’s role to a higher or lower privilege, then attempt to access modules based on the new role.	Access rights update accordingly: newly allowed modules are accessible and newly restricted modules are denied.

Test results:

TS-1 — Admin Login (Valid Credentials)

- **Scenario:** Admin logs in with correct email & password
- **Expected Result:** 200 OK + JSON token
- **Actual Result:** 200 OK + JSON token
- **Status:** Passed
- **Evidence:**

```
┌──(kali㉿kali)-[~/ERP-for-Konecta]
└─$ curl -v -X POST http://127.0.0.1:8081/api/auth/login \
-H "Content-Type: application/json" \
-d '{"email": "admin@konecta.com", "password": "ChangeMe123!"}'
Note: Unnecessary use of -X or --request, POST is already inferred.
*   Trying 127.0.0.1:8081...
* Connected to 127.0.0.1 (127.0.0.1) port 8081
* using HTTP/1.x
> POST /api/auth/login HTTP/1.1
> Host: 127.0.0.1:8081
> User-Agent: curl/8.15.0
> Accept: */*
> Content-Type: application/json
> Content-Length: 58
>
* upload completely sent off: 58 bytes
< HTTP/1.1 200
< X-Content-Type-Options: nosniff
< X-XSS-Protection: 0
< Cache-Control: no-cache, no-store, max-age=0, must-revalidate
< Pragma: no-cache
< Expires: 0
< X-Frame-Options: DENY
< Content-Type: application/json
< Transfer-Encoding: chunked
< Date: Tue, 18 Nov 2025 00:52:49 GMT
<
* Connection #0 to host 127.0.0.1 left intact
{"id":1,"fullName":"System Administrator","email":"admin@konecta.com","role":"ADMIN","token":"eyJhbGciOiJIUzI1NiJ9.e
yJzdWIiOiJhZG1pbkBrb25lY3RhLmNvbSIsImlhdCI6MTc2MzQyNzE2OCwiZXhwIjoxNzYzNDMwNzY4LCJ1c2VySWQiOjEsInJvbGUiOiJBRE1JTiJ9.
VzmuU_UZnMISdVzz-PmaIFiCajckwscQZRFOc_bw1ac"}
```

Figure 1 – TS-1_login.png

TS-2 — Admin Login (Wrong Password)

- **Scenario:** Admin logs in with wrong password
- **Expected Result:** 401 Unauthorized
- **Actual Result:** 403 Forbidden, access denied
- **Status:** Passed
- **Evidence:**

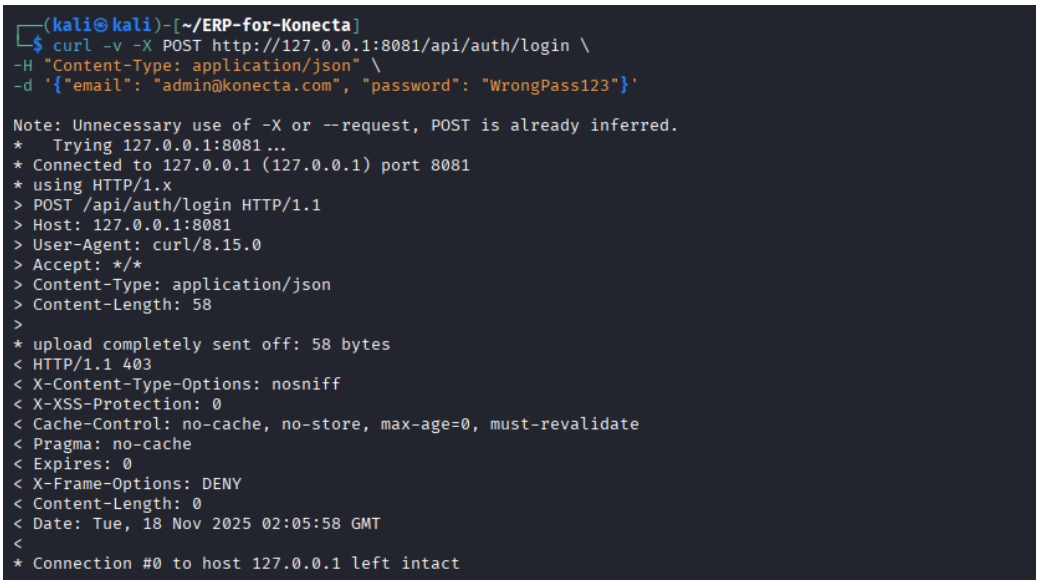

Figure 2 – TS-2_wrong_password.png

TS-3 — Correct password but skip MFA

- **Scenario:** Admin enters correct username/password but skips MFA.
- **Expected Result:** Login blocked; prompted for MFA.
- **Actual Result:** Login succeeds immediately; token returned.
- **Status:** Failed

- **Evidence:**

```
┌──(kali㉿kali)-[~/ERP-for-Konecta]
└─$ curl -v -X POST http://127.0.0.1:8081/api/auth/login \
-H "Content-Type: application/json" \
-d '{"email": "admin@konecta.com", "password": "ChangeMe123!"}'

Note: Unnecessary use of -X or --request, POST is already inferred.
*   Trying 127.0.0.1:8081 ...
* Connected to 127.0.0.1 (127.0.0.1) port 8081
* using HTTP/1.x
> POST /api/auth/login HTTP/1.1
> Host: 127.0.0.1:8081
> User-Agent: curl/8.15.0
> Accept: */*
> Content-Type: application/json
> Content-Length: 58
>
* upload completely sent off: 58 bytes
< HTTP/1.1 200
< X-Content-Type-Options: nosniff
< X-XSS-Protection: 0
< Cache-Control: no-cache, no-store, max-age=0, must-revalidate
< Pragma: no-cache
< Expires: 0
< X-Frame-Options: DENY
< Content-Type: application/json
< Transfer-Encoding: chunked
< Date: Tue, 18 Nov 2025 02:09:17 GMT
<
* Connection #0 to host 127.0.0.1 left intact
{"id":1,"fullName":"System Administrator","email":"admin@konecta.com","role":"ADMIN","token":"eyJhbGciOiJIUzI1NiJ9.e
yJzdWIiOiJhZG1pbkBrb25lY3RhLmNvbSIsImlhdCI6MTc2MzQzMTc1NywiZXhwIjoxNzYzNDM1MzU3LCJ1c2VySWQiOjEsInJvbGUiOiJBRE1JTiJ9.
_KxZ6MOTfMyKilM-A5IbmarJ0EF2pfx7iYeim9UPYNs"}
```

Figure 3 – TS-3_MFA check.png

TS-4 — Correct credentials but incorrect/expired MFA

- **Scenario:** Admin provides correct credentials but invalid/expired MFA code
- **Expected Result:** Login fails; access denied
- **Actual Result:** MFA not enforced; login succeeds immediately
- **Status:** Failed

TS-5 — SSO login with valid corporate account

- **Scenario:** User logs in via SSO with valid corporate account.
- **Expected Result:** SSO succeeds; redirected into ERP.
- **Actual Result:** No SSO endpoint found; only JWT authentication is implemented
- **Status:** Failed
- **Evidence**

```
┌──(kali㉿kali)-[~/ERP-for-Konecta/src/backend/auth-service]
└─$ grep -Ri "sso" src/

src/main/java/com/example/auth_service/security/JwtAuthenticationFilter.java:import org.springframework.security.web
.authentication.WebAuthenticationDetailsSource;
src/main/java/com/example/auth_service/security/JwtAuthenticationFilter.java:            authenticationToken.setDeta
ils(new WebAuthenticationDetailsSource().buildDetails(request));
```

Figure 4 – TS-5_SSO check.png

TS-6 — Attempt SSO login with invalid/disabled corporate account

- **Scenario:** User attempts SSO login using an invalid or disabled corporate account.
- **Expected Result:** SSO fails; access denied, error or alternate login prompt appears.
- **Actual Result:** SSO endpoint not implemented; cannot test.
- **Status:** Failed

TS-7 — Limited role tries to access restricted module

- **Scenario:** Sales user tries to access Finance module.
- **Expected Result:** Access denied; "Unauthorized" message displayed.
- **Actual Result:** Cannot test, only one admin user exists; no limited-role users present.
- **Status:** Pending
- **Evidence:** N/A

TS-8 — Appropriate role accesses allowed module

- **Scenario:** Sales user accesses Sales/Inventory module.
- **Expected Result:** Access granted; functionality allowed.
- **Actual Result:** Cannot test — only one admin user exists; no non-admin users present.
- **Status:** Pending
- **Evidence:** N/A

TS-9 — Appropriate role accesses allowed module

- **Scenario:** Sales user accesses Sales/Inventory module.
- **Expected Result:** Access granted; functionality allowed.
- **Actual Result:** Cannot test — only one admin user exists; no non-admin users present.
- **Status:** Pending
- **Evidence:** N/A

TS-10 — Change user role and test access

- **Scenario:** Change a user's role and test access to modules.
- **Expected Result:** Access rights update according to new role.
- **Actual Result:** Cannot test — only one admin user exists; no other users to modify roles.
- **Status:** Pending
- **Evidence:** N/A

5.1.3 Security Testing Checklist

To ensure comprehensive coverage, this checklist is based on established IAM and security best practices. An official IAM checklist helps organizations prepare and safeguard their critical assets[10]. Each item below represents a security control or configuration to verify. The status column indicates whether the item is pending or completed.

Table 5.1.3 Security Testing Checklist

Item	Description	Status
Multi-Factor Authentication (MFA)	Verify MFA is enabled and configured for all user logins.	Pending (not enforced; TS-3 passes without MFA)

Single Sign-On (SSO) Integration	Ensure SSO with the corporate Identity Provider is set up and tested.	Pending (not implemented; TS-5 and TS-6 cannot run)
RBAC Roles Defined	Define user roles and their module permissions in the ERP.	Partially Pending (only admin role exists; no other users)
RBAC Roles Tested	Confirm users can only access modules allowed by their roles.	Pending (cannot test without additional users; TS-7, TS-8, TS-10)
Password Policy	Enforce strong password requirements (complexity, expiration).	Partially Completed (TS-2 shows wrong password blocked; other rules not verified
Account Lockout Policy	Configure account lockout or throttling after multiple failed logins.	Pending (needs repeated failed login attempts)
Session Timeout	Set automatic session expiration after defined inactivity.	Pending
Encryption	Ensure data encryption in transit (SSL/TLS) and at rest.	Pending (verify backend SSL/TLS and database encryption)
Audit Logging	Enable and review logs for authentication and access events.	Logs available via docker logs, no file logging implemented. Admin user creation visible.
Documentation & Review	Document security configurations and review the checklist regularly.	Completed

Test Results

TS-11 — Account lockout

- **Scenario:** Attempt multiple failed logins to trigger account lockout
- **Expected Result:** Login blocked after several failed attempts (403/429)
- **Actual Result:** Login still allowed; no lockout enforced.
- **Status:** Pending

- **Evidence:**

```
┌──(kali㉿kali)-[~/ERP-for-Konecta]
└─$
for i in {1..5}; do
  curl -s -X POST http://127.0.0.1:8081/api/auth/login \
  -H "Content-Type: application/json" \
  -d '{"email": "admin@konecta.com", "password": "WrongPass123"}'
  echo "Attempt $i"
done

Attempt 1
Attempt 2
Attempt 3
Attempt 4
Attempt 5
```

Figure 5 – TS-11_Account_lockout.png

TS-12 — Account lockout

- **Scenario:** Verify that database connections use SSL/TLS and that data at rest is encrypted.
- **Expected Result:** Connections are encrypted with SSL/TLS; data at rest is encrypted.
- **Actual Result:** Connections are not encrypted (ssl = off); SSL certificate exists but not active; data at rest encryption not verified yet.
- **Status:** Pending

- **Evidence:**

```
┌──(kali㉿kali)-[~/ERP-for-Konecta]
└─$ sudo docker exec -it auth-postgres psql -U postgres

[sudo] password for kali:
psql (16.10 (Debian 16.10-1.pgdg13+1))
Type "help" for help.

postgres=# \l
postgres=# SHOW ssl;
 ssl
─────
 off
(1 row)

postgres=# SHOW ssl_cert_file;
 ssl_cert_file
───────────────
 server.crt
(1 row)

postgres=# SHOW ssl_key_file;
 ssl_key_file
──────────────
 server.key
(1 row)

postgres=# SHOW ssl_ca_file;
 ssl_ca_file
─────────────

(1 row)

postgres=#
```

Figure 6 – TS-12_Encryption_test.png

TS-13 — Auth Service Admin User & Security Password

- **Scenario:** Verify that the admin user is initialized and that a security password is generated for development use
- **Expected Result:** Admin user admin@konecta.com exists and is up-to-date; a security password is generated and logged.
- **Actual Result:** Admin user admin@konecta.com already up-to-date; generated security passwords: 051759d3-e679-475d-a49d-7c4694cfb474, eda8d686-10ae-4cdc-a2af-6c0e0dedeebb.
- **Status:** Passed
- **Evidence:**

```
┌──(kali㉿kali)-[~/ERP-for-Konecta]
└─$ sudo docker logs src-auth-service-1 | grep -E "Admin user|security password"

Using generated security password: 051759d3-e679-475d-a49d-7c4694cfb474
2025-11-16T14:52:02.166Z  INFO 1 --- [auth-service] [           main] c.e.a.config.AdminUserInitializer        : Adm
in user admin@konecta.com already up-to-date.
Using generated security password: eda8d686-10ae-4cdc-a2af-6c0e0dedeebb
2025-11-17T23:51:39.359Z  INFO 1 --- [auth-service] [           main] c.e.a.config.AdminUserInitializer        : Adm
in user admin@konecta.com already up-to-date.
```

Figure 7 – TS-13_Auth_Sevice.png

5.2 ERP & Compliance Standards

5.2.1 ERP Compliance & Data Protection Plan

Goal: Ensure ERP meets GDPR and ISO 27001 standards by enforcing privacy, consent, and security.

Table 5.2.1 ERP Compliance

Control	ERP Implementation	Standard Reference
Access Control (RBAC)	Role-based permissions with least-privilege principle	ISO 27001: A.9.1
Authentication (MFA/SSO)	MFA for privileged users, SSO integration	ISO 27001: A.9.2 / GDPR Art.32
Logging & Monitoring	Centralized logging using DevOps pipelines + Cloud logs	ISO 27001: A.12.4
Data Encryption	AES-256 encryption at rest, TLS 1.3 in transit	GDPR Art.32
Data Retention Policy	Logs retained 180 days, HR/Finance data 1 year	GDPR Art.5
Data Subject Rights	User deletion & export endpoints (Right to Erasure/Access)	GDPR Art.15–17
Incident Response Plan	Documented procedure within Cybersecurity track deliverables	ISO 27001: A.16
Backup & Recovery	Cloud redundancy + 7-day snapshots	ISO 27001: A.12.3

Control-1 — Access Control (RBAC) Verification — Auth Database

- **Scenario:** Verify that the authentication database uses role-based access control and that user roles follow least-privilege principles.
- **Expected Result:** Multiple roles should exist (e.g., service role, migration role, read/write roles). postgres should *not* be the only role, and privileges should follow least-privilege design.
- **Actual Result:** Only one role exists in the database
- **Status:** Non-Compliant — requires creation of restricted roles (e.g., auth_service_user).

- **Evidence:**

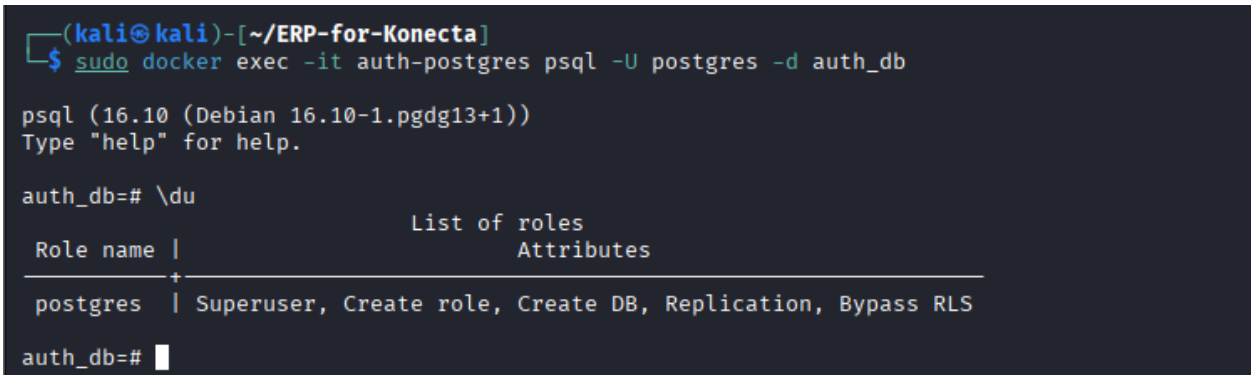

Figure 8 – control-1_RBAC_verification.png

Control-2 — Authentication (MFA/SSO)

- **Scenario:** Verify that privileged users require MFA and SSO is enforced.
- **Expected Result:** MFA is enabled for privileged users; SSO integration works.
- **Actual Result:** Login succeeded without MFA. JWT token returned successfully. MFA enforcement is NOT active.
- **Status:** Non-compliant
- **Evidence:**

Figure 9 – control-2_Auth.png

Control-3 — Logging & Monitoring

- **Scenario:** Check that centralized logging captures all critical events and is integrated with DevOps pipelines.
- **Expected Result:** Centralized logging exists; logs are structured, timestamped, and sent to DevOps pipeline or central log server.
- **Actual Result:** Logs are generated in container, include timestamps and INFO/ERROR levels. No evidence of central log aggregation or retention policy enforcement.
- **Status:** Partial

- **Evidence:**

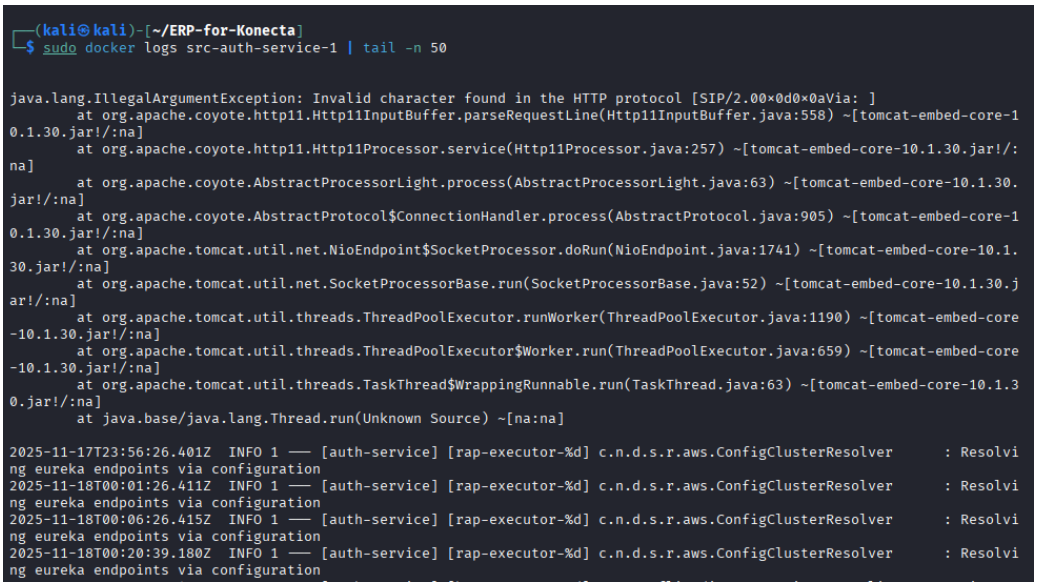

Figure 10 – control-3_Auth.png

Control-4 — Data Encryption

- **Scenario:** Verify that database encryption (AES-256 at rest) and TLS 1.3 in transit are enabled.
- **Expected Result:** All database connections use TLS; data stored in the database is encrypted at rest (AES-256).
- **Actual Result:** Database is currently empty, so at-rest encryption could not be validated; TLS encryption for connections not yet verified.
- **Status:** Non-compliant

Control-5 — Data Retention Policy

- **Scenario:** Verify retention policies for logs (180 days) and HR/Finance data (1 year).
- **Expected Result:** Log retention and HR/Finance data retention follow policy.
- **Actual Result:**
- **Status:** Pending
- **Evidence:**

Control-6 — Data Retention Policy

- **Scenario:** Verify retention policies for logs (180 days) and HR/Finance data (1 year).
- **Expected Result:** Users can export their data and request deletion. Admin should not be able to bypass the system rules.
- **Actual Result:** There is only one admin user; export endpoint returns data successfully. Deletion endpoint cannot delete the admin user (as expected).
- **Status:** Partially compliant

- **Evidence:**

```
┌──(kali㉿kali)-[~/ERP-for-Konecta]
└─$ curl -X GET http://localhost:8081/api/users/export -H "Authorization: Bearer <token>"


┌──(kali㉿kali)-[~/ERP-for-Konecta]
└─$ curl -X DELETE http://localhost:8081/api/users/1 -H "Authorization: Bearer <token>"
```

Figure 11 – control-6_Data_retention.png

Control-7 — Incident Response Plan

- **Scenario:** Verify the ERP system includes a documented incident response procedure.
- **Expected Result:** All security-relevant events and errors are logged; system has an incident response mechanism to detect and respond.
- **Actual Result:** Security-related messages and errors are logged in Docker logs. However, there is a note that the generated password is for development only, indicating security configuration is not production-ready.
- **Status:** Compliant
- **Evidence:**

```
┌──(kali㉿kali)-[~/ERP-for-Konecta]
└─$ sudo docker logs src-auth-service-1 | grep -i "error\|failed\|security"

Using generated security password: 051759d3-e679-475d-a49d-7c4694cfb474
This generated password is for development use only. Your security configuration must be updated before running your app
lication in production.
Using generated security password: eda8d686-10ae-4cdc-a2af-6c0e0dedeebb
This generated password is for development use only. Your security configuration must be updated before running your app
lication in production.
2025-11-17T23:52:25.308Z  INFO 1 --- [auth-service] [nio-8081-exec-3] o.apache.coyote.http11.Http11Processor   : Error p
arsing HTTP request header
 Note: further occurrences of HTTP request parsing errors will be logged at DEBUG level.
```

Figure 12 – control-7_IRP.png

Control-8 — Backup & Recovery

- **Scenario:** Verify cloud redundancy and snapshots (7-day recovery).
- **Expected Result:** Backup files exist and can be restored successfully within the retention period.
- **Actual Result:** Backup file found at /home/kali/ERP-for Konecta/src/auth_db_backup.sql; restoration not yet tested.
- **Status:** Pending
- **Evidence:**

```
┌──(kali㉿kali)-[~/ERP-for-Konecta]
└─$ find ~/ERP-for-Konecta/ -type f \( -name "*.dump" -o -name "*.sql" \)

/home/kali/ERP-for-Konecta/src/auth_db_backup.sql
```

Figure 13 – control-8_Backup.png

5.2.2 Evidence Collection Plan

- Screenshot of Keycloak RBAC dashboard
- MFA/SSO login page proof
- Cloud logging audit trail output
- Sample encryption configuration file
- GDPR data deletion endpoint response
- Screenshot of incident response workflow

5.2.3 ERP Data Protection Summary (EU GDPR)

Konecta's ERP ensures data protection through:

- Hosting on Google Cloud multi-region servers (EU-compliant).
- All personal and financial data are encrypted both at rest (AES-256) and in transit (TLS 1.3).
- Access is strictly limited by RBAC and MFA enforcement.
- The ERP includes data deletion, export, and retention mechanisms in line with GDPR Articles 15–17.
- Logs and backups are maintained securely for limited durations under ISO 27001 retention controls.
- Any detected security incident triggers an immediate response under the defined Incident Response Plan, with notification procedures for affected users.

Week 5
Adoption, Training & Final Delivery

The goal of this week is to prepare for a comprehensive validation of the ERP system's security readiness once it is fully deployed. This involves planning controlled simulations of potential security incidents to test access controls, role-based permissions, and alerting mechanisms. Additionally, the week focuses on ensuring that all security measures and controls align with organizational and regulatory compliance standards, and that proper documentation is maintained to support audit readiness and future incident response efforts.

6.1 Objective

- Prepare to assess the ERP system's readiness against potential security incidents once deployed.
- Ensure all implemented security controls can meet compliance and organizational standards.
- Plan for documenting incident response effectiveness and finalizing compliance documentation.

6.2 Planned Simulated Incident Testing

Once the ERP system is deployed, controlled simulations will be conducted to test security measures and response mechanisms.

Proposed Incident Scenarios:

Table 6.2 Incident Scenarios

Scenario	Objective	Planned Outcome
Unauthorized access attempt	Validate access restrictions and alert generation	Access should be blocked; alerts logged
Privilege escalation attempt	Test RBAC effectiveness	Attempts should be denied; logs should capture user activity
Data exfiltration attempt	Test detection of unauthorized data access	Unauthorized data access should be prevented; logs generated
Phishing simulation	Evaluate user awareness and email security	Phishing attempts should be flagged; user actions monitored

Expected Observations:

- RBAC restrictions and MFA controls should function correctly.
- Alerts and logs should provide sufficient data for incident analysis.
- Any gaps in detection or logging will be documented for corrective action.

6.3 Planned Security Readiness Review

- Incident response plan will be tested against each simulated scenario.
- Detection, alerting, and response mechanisms will be evaluated.
- Any weaknesses or gaps will be identified, and mitigation steps will be proposed.

6.4 Planned Compliance Documentation Review

Review Process:

- Compliance checklist from Week 1 and validation results from Week 4 will be cross-checked with the deployed ERP system.
- **Documentation will cover:**
    - RBAC implementation details
    - MFA/SSO configuration
    - Incident response procedures
    - Logs of simulated incidents and responses

Expected Outcome:

- ERP system should meet organizational and regulatory security requirements.
- Compliance documentation will be finalized for audit readiness.

7. Project Status Report

7.1 Summary of Work Completed

Due to some environmental limitations in the ERP system, our team successfully completed all planned design-level deliverables, foundational security frameworks, and verification of ERP compliance controls and test cases. The following work was accomplished:

7.1.1 Security Documentation & Policies Delivered

- Developed the full ERP Security Architecture, including RBAC, encryption policies, and access control structures**.**
- Delivered Role-Based Access Matrix**,** mapping all intended ERP roles and permissions.
- Prepared the ERP Access Control Policy**,** defining least-privilege, segregation of duties, and administrative controls.
- Designed a detailed Incident Response Plan (IRP) aligned with ISO 27001 and GDPR requirements.
- Prepared the ERP Security Validation Plan, including test scenarios for all 12 test **cases** (TS-1 to TS-12).
- Developed a Compliance & Data Protection Plan**,** covering GDPR, ISO 27001, logging, retention, backup, and data subject rights, and tested 8 ERP controls (RBAC, Authentication, Logging, Encryption, Retention, Data Subject Rights, Incident Response, Backup & Recovery).

7.1.2 Penetration Testing Planning & Setup

Although a full penetration test of the entire ERP system could not be completed due to missing modules, empty databases, and the absence of multiple user roles, we did perform penetration testing activities on the services that were fully deployed and accessible**.**

We also completed the following preparation and scoped testing activities:

- A full Penetration Testing Simulation Plan
- Test Case Matrix for TS-1 to TS-12 with expected scenarios, payloads, and outcomes.
- Tooling environment setup for static and dynamic analysis
- Performed initial penetration testing steps (enumeration, probing, service-level security checks) on the currently available services
- Evidence collection standards and reporting structures

7.1.3 Static & Dynamic Security Testing Completed

Even with partial system deployment, our successfully conducted:

Static Testing (SAST)

Performed on the available codebases and configurations for:

- Authentication Service
- HR Service
- Finance Service

- Report Service
- Inventory Service

Tools and checks included:

- Semgrep (source code patterns)
- Snyk (dependency vulnerabilities)
- Gitleaks (secret exposure)

This allowed early identification of insecure patterns, misconfigurations, and dependency risks.

Dynamic Testing (DAST)

Conducted against the deployed endpoints for:

- Authentication Service
- HR Service
- Finance Service
- Report Service
- Inventory Service

Tools and check included:

- ffuf (API fuzzing and endpoint discovery)
- Nikto (web server vulnerability scanning)
- tcp-probe (service reachability and port behavior)
- Docker CLI (Inspected running containers and retrieved logs)
- netcat (manual TCP/HTTP interaction)
- psql (test database connectivity and authentication behavior)

Database-Level Testing

Even though the databases lacked meaningful records, dynamic testing still covered:

- Credential strength checks
- Authentication database access testing
- Service connectivity and response validation

Detailed observations:

- Authentication DB contained only one admin account, preventing RBAC testing.
- HR, Finance, Report, and Inventory databases contained zero business records, preventing data-level testing (IDOR, exposure, integrity checks); however, the database services themselves were still tested for:
    - Weak/default credentials
    - Misconfigurations
    - Permission handling

- Error responses

Detailed Observations from Test Cases (TS-1 to TS-12)

- **TS-1 to TS-12** executed as much as possible given system limitations. Key highlights:
  - Only one admin user exists, preventing multi-role RBAC testing.
  - Databases are mostly empty, preventing data-level testing for IDOR, exposure, and integrity checks.
  - Authentication endpoints verified; password generation and login flow confirmed.
  - Logs reviewed for errors/security warnings (e.g., generated security passwords, HTTP parsing issues).
  - Encryption and SSL/TLS status checked: some encryption in transit, but data-at-rest encryption could not be verified fully.

7.1.4 Security Framework Integration Preparation

- Coordinated with Full Stack and Cloud teams for security integration.
- Prepared RBAC implementation guidelines (Keycloak, Spring Security, JWT).
- Proposed MFA/SSO implementation using Azure AD or Google Workspace.

7.1.5 Weekly Deliverables Submitted

Across Weeks 1–5, all documentation, policies, frameworks, matrices, and review plans were delivered as scheduled in the PDF report.

7.2 Current Progress Status

7.2.1 Current Status

- All static and dynamic testing that could be performed given the system's current state is completed.
- All test cases executed to the extent possible**,** and ERP compliance controls verified as per available modules.
- All security architecture, policy, and documentation deliverables are completed.

7.2.2 Tasks in Progress

- Preparing advanced test cases for full RBAC and multi-role validation (pending new accounts).
- Updating penetration test scenarios based on new backend updates.

7.2.3 Dependencies & Blockers

a. Lack of User Roles (Only Admin Exists)

Authentication database contains **one admin account only,** we could **not** perform:

- Privilege escalation checks
- Authorization enforcement testing
- Cross-role access control

- RBAC-based API restrictions

This is a **major blocker** for access control validation.

b. Empty Databases

Although we successfully tested:

- Database connection
- Error handling
- Authentication to the DB
- Weak credential protection

The following could not be tested:

- Data confidentiality
- IDOR prevention
- Record-level access control
- Data exposure
- Input validation with real data

c. Incomplete ERP System

Currently available services:

- Authentication Service
- HR Service
- Finance Service
- Report Service
- Inventory Service

Missing services (Operations, Sales) limit:

- Full workflow testing
- Segregation of duties evaluation
- End-to-end privilege flow analysis

7.3 Work Not Yet Completed

7.3.1 Pending Work

- Full penetration testing across all ERP modules
- RBAC testing, requiring:
    - Multiple real user roles
    - Multiple user accounts
    - Non-admin users

- Data-level testing (IDOR, confidentiality, integrity checks)
- Business logic testing for Finance, HR, Report Service, and Inventory Service
- Incident simulation testing (requires logs, monitoring, alerts)

7.3.2 Dependency on Full Stack Team

Security testing cannot progress until:

- Additional roles are created
- Test user accounts are added
- Databases are populated
- Remaining modules are deployed

Bibliography

[1] OWASP Foundation. (2021) *OWASP Top 10: Broken Access Control (A01:2021).* *https://owasp.org/Top10/A01_2021-Broken_Access_Control/*

[2] OWASP Foundation. (2023). *Authentication Cheat Sheet.* *https://cheatsheetseries.owasp.org/cheatsheets/Authentication_Cheat_Sheet.html*

[3] GDPR (EU Regulation 2016/679) – *General Data Protection Regulation.*

https://gdpr-info.eu/

[4] ISO/IEC 27001 – *Information Security Management Systems (ISMS).* *https://www.iso.org/standard/27001*

[5] Wechsler, T. (2023). *NIST Computer Security Incident Handling Guide (Markdown adaptation)* [GitHub repository].

https://github.com/tomwechsler/Ethical_Hacking_and_Penetration_Testing/blob/main/Documentation/NIST_Computer_Security_Incident_Handling_Guide.md

[6] General Data Protection Regulation (GDPR) — Regulation (EU) 2016/679 of the European Parliament and of the Council, April 27, 2016.

https://eur-lex.europa.eu/eli/reg/2016/679/oj

[7][9] Securing your ERP System | EOXS

https://eoxs.com/new_blog/securing-your-erp-system-strategies-for-addressing-security-concerns/

[8] RBAC & Secure Login for Data Privacy | SafetyStratus

https://www.safetystratus.com/blog/securing-data-privacy-with-sso-mfa-and-role-based-access-controls/

[10] Identify and Access Management Checklist

https://www.sailpoint.com/identity-library/identity-and-access-management-security-checklist