Target of Evaluation (TOE)

1. Introduction

   The security audit’s Target of Evaluation (TOE) is the CybSec web-based application, which uses MySQL as its database management system (DBMS). The system allows users to store, retrieve, and update data. Administrators manage access, integrity and security. This Security Target document is developed as part of the Common Criteria (CC) certification process. It defines the architecture, threats, security objectives, and assurance requirements of the TOE.

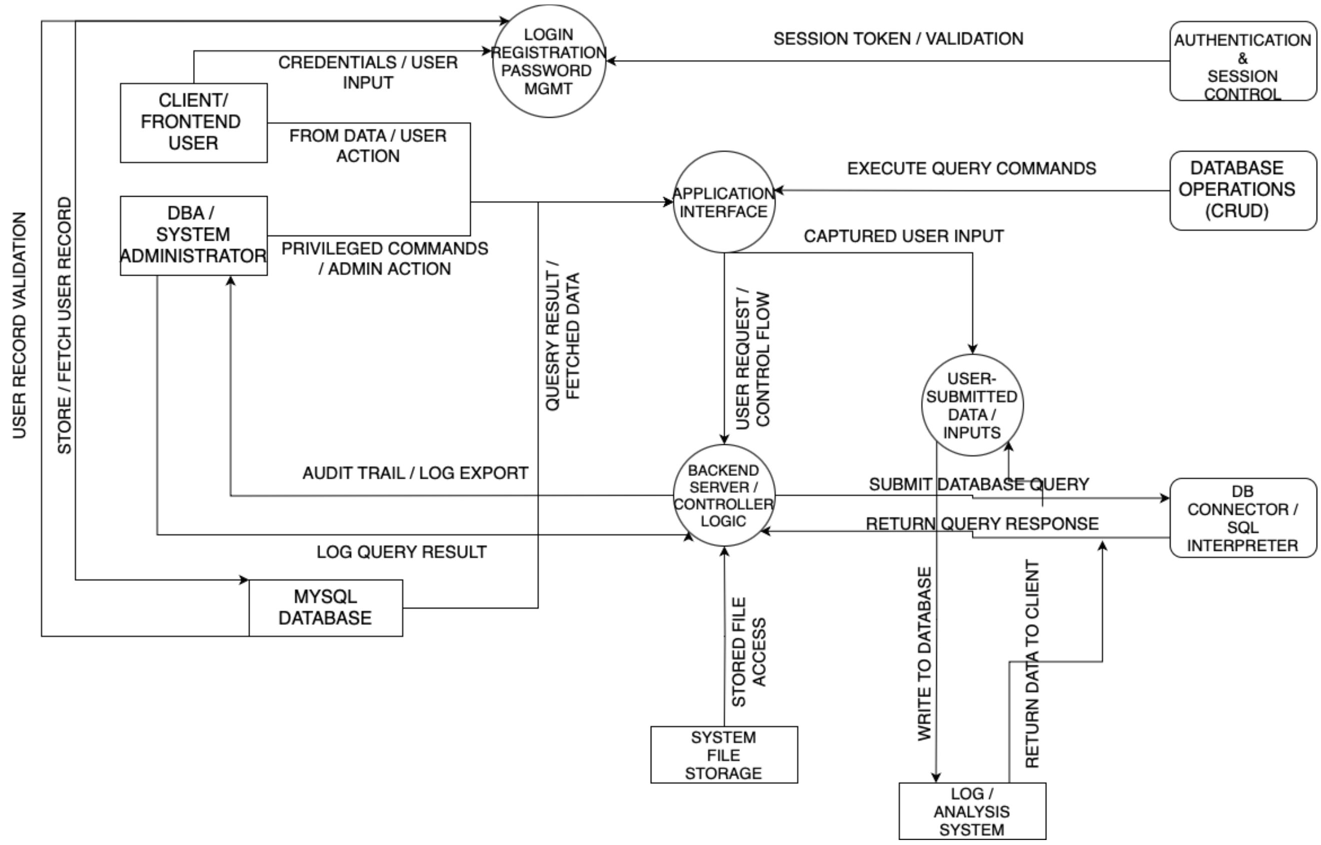
LOGIN REGISTRATION PASSWORD MGMT
SESSION TOKEN / VALIDATION
AUTHENTICATION & SESSION CONTROL
CREDENTIALS / USER INPUT
CLIENT/ FRONTEND USER
FROM DATA / USER ACTION
DBA / SYSTEM ADMINISTRATOR
PRIVILEGED COMMANDS / ADMIN ACTION
USER RECORD VALIDATION
STORE / FETCH USER RECORD
QUESRY RESULT / FETCHED DATA
APPLICATION INTERFACE
EXECUTE QUERY COMMANDS
DATABASE OPERATIONS (CRUD)
CAPTURED USER INPUT
USER REQUEST / CONTROL FLOW
USER-SUBMITTED DATA / INPUTS
AUDIT TRAIL / LOG EXPORT
BACKEND SERVER / CONTROLLER LOGIC
SUBMIT DATABASE QUERY
RETURN QUERY RESPONSE
DB CONNECTOR / SQL INTERPRETER
LOG QUERY RESULT
MYSQL DATABASE
STORED FILE ACCESS
SYSTEM FILE STORAGE
WRITE TO DATABASE
RETURN DATA TO CLIENT
LOG / ANALYSIS SYSTEM

Figure 1

Figure 1 illustrates how the system components interact. Users access the system through the Login/Registration/Password Management and interact via Application Interface. The Backend Server processes the inputs from the users, who interact with the interface. The Backend Server works with the SQL server. The diagram illustrates users getting data back. The diagram also shows that before accessing anything sensitive, a user has to get past the user.

2. System Description

The CybSec system has three tiers:

- Frontend Layer (User Interface) – runs on a user computer
- Server Layer (Data Repositories)
  - MySQL for user and operational data that is structured.
  - MongoDB for unstructured data (e.g., logs, feedback).
  - File system for document and reports.

The system supports two user roles:

- Regular User – interact with their personal data.
- Administrators – configure access policies and maintain system security.

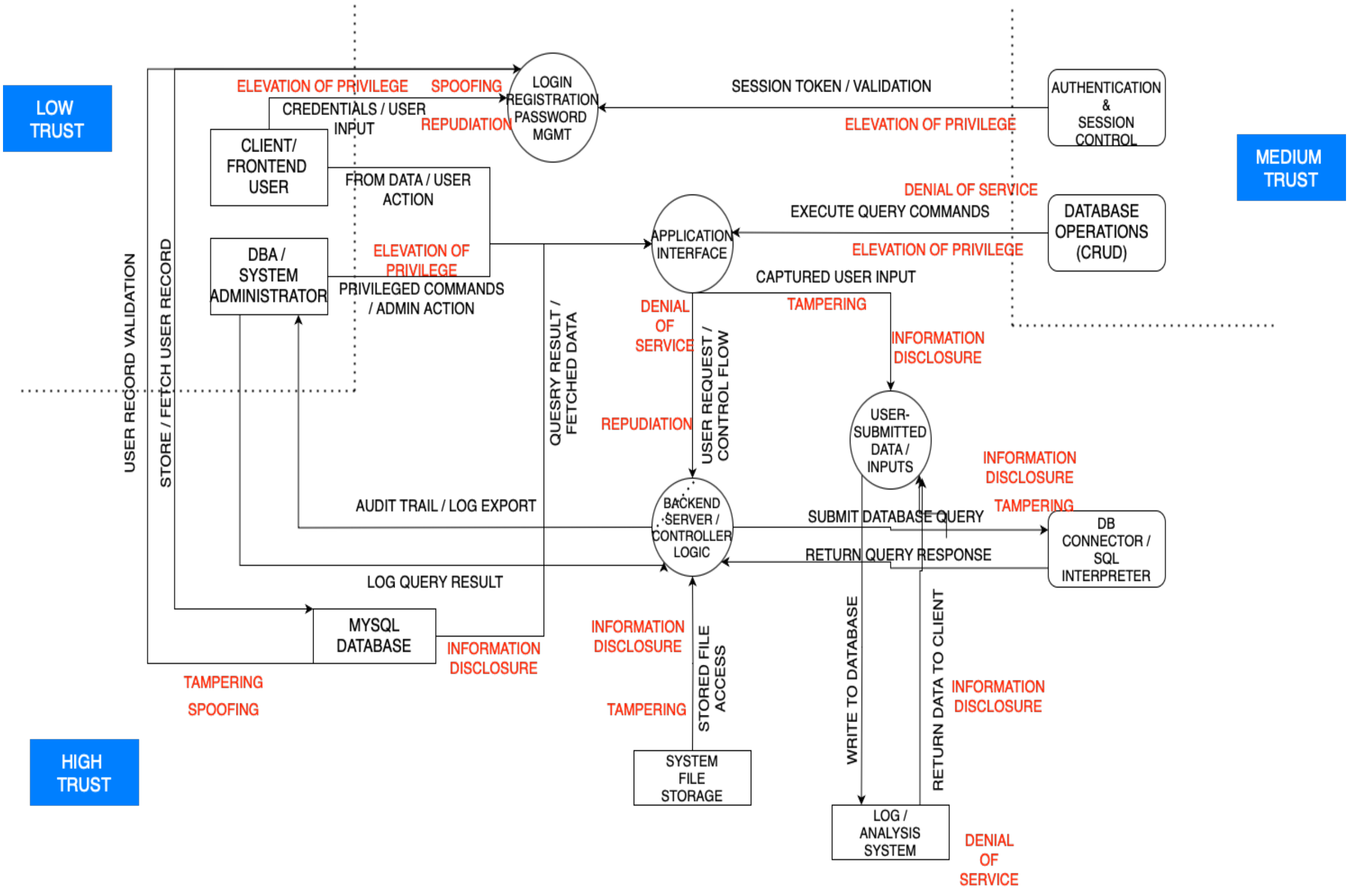

Figure 2

Figure 2 was created using draw.io as well to illustrate the architecture of the CybSec system. The front end, back end, and storage layers interact across distinct trust boundaries in the diagram. It spotlights key components and potential threat points.

Threat Analysis

The TOE may face threats affecting confidentiality, integrity, or availability. Using the STRIDE model, the following categories apply:

- Spoofing - Impersonating user to gain unauthorized access.
- Tampering – altering stored or transmitted data.
- Repudiation - Denying performed actions without accountability
- Information Disclosure - Unauthorized access to sensitive information
- Denial of Service (DoS) – disrupting availability of the system
- Elevation of Privilege - Gaining admin privileges through exploitation

Threats to the system and its environment

THREATS	DESCRIPTION
T. SPOOFING	An attacker attempts to impersonate a legitimate user by exploiting weak authentication mechanisms, such as weak passwords or unprotected login endpoints.
T. DATA_TAMPER	The adversary can alter any data within the storage layer they choose to, compromising its integrity.
T. REPUDIATION	Users deny performing certain (e.g., transactions or modifications) due to missing or insufficient logging and audit

	mechanisms. Leads to the data being baseless.
T. INFO_DISCLOSURE	Sensitive data, such as user details or transaction records, is leaked due to improper API responses, lack of encryption, or misconfigured access controls.
T. DoS	Adversary floods the website with high traffic, overwhelming the system, making the service unavailable or slow.
T. ELEVATION_PRIVILEGE	The threat actor exploits vulnerabilities in the system's design or implementation to bypass authorisations and gains access to higher level privileges which includes access to the databases and their content.
T. SESSION_HIJACK	Attackers intercept authentication sessions via MITM (Man-in-the-Middle) attacks or session fixation techniques, gaining unauthorized control over a user's session.
T.SQL_INJ	SQL injection is performed as the threat actor injects malicious code into the user input and ends up affecting the SQL database ending in the data being compromised.
T.XSS	Malicious scripts are injected into web pages viewed by other users, allowing attackers to steal session cookies or manipulate webpage content.
T. MALWARE_INJ	Malware can be injected into the IT environment and spread further into the backend and frontend. Compromises the security and functionality of the TOE.
T. BRUTE_FORCE	Automated scripts attempt to guess user passwords by

	repeatedly trying different combinations, compromising weakly protected accounts.
T. INSIDER_THREAT	A malicious or negligent insider (employee or administrator) abuses legitimate access to leak, modify, or delete sensitive data.

Assumptions regarding the TOE

ASSUMPTIONS	DESCRIPTION
A. ADMIN1	The administrators/root users of the TOE operate on well-secure systems and are trained on all aspects of the TOE.
A. ADMIN2	The administrators follow secure guidance given by the head in charge and are careful to notice any changes within the system or its' operations. They have excellent cyber security training.
A.USER	Expected that not every user will be trained on cyber security measures and may be susceptible to social engineering attacks.
A. UPDATES	Any updates on the TOE are properly evaluated to ensure their safety before being installed. Every instalment and update of the TOE is verified to ensure it is malware-free.
A. BACKEND	The backend servers are installed on separate physical machines that allow access to administrators only and are located in a secure room. They are not stored on the cloud.

A. NETWORK	The TOE operates in an environment with secure network configurations, including firewall protections, encrypted communications, and VPN access for administrators.
A. AUTHENTICATION	Users and administrators are required to use secure authentication methods such as multi-factor authentication (MFA) to access the system.
A. PHYSICAL	It is assumed that only authorized personnel have physical access to critical infrastructure, such as servers and networking equipment.
A. MONITORING	The TOE environment us assumed to have logging and monitoring mechanism in place to detect anomalies, unauthorized access attempts, and potential breaches.
A.DATA	All sensitive data stored within the system is encrypted data and protected from unauthorized modifications.
A. EXTERNAL	The TOE is assumed to be protected from direct external internet exposure, with traffic being filtered through firewalls and access control mechanism.

Security Objectives

OBJECTIVES	DESCRIPTION
O. AUTH_SEC	Some security measures that can be implemented to protect the user are two/multi-factor authentication, strong password requirements upon register, one-time password upon authentication, regular email requesting users to change their passwords, email notification every time the user logs in their account, email notification every time there is a change to their details, etc.
O.DATA_ENC	The data needs to be encrypted during its transmission between the web application user and the server.
O. ACCESS_CTRL	The databases should only be accessed by the root users/administrators, who have the minimum necessary privileges.
O. VULN_MGMT	Set in place continuous monitoring to identify vulnerabilities present in the system. Establish a patching and management cycle to address and prevent the exploitation of the discovered vulnerabilities.
O.LOG_AUDIT	Implement mechanism to log and monitor user actions or detecting suspicious activity over the network.
O.NET_SEC	Constantly analyse the traffic coming over the network. Set up a firewall with rules that block

	access to known malicious domains.
O.DATA_INT	Implement integrity controls such as checksums, hashing, and database validation to prevent data tampering and ensure information accuracy.
O. SESSION_SEC	Implement secure session management techniques to prevent session hijacking, replay attacks, and unauthorized access.
O. BACKUP_REC	Data is regularly backed up, encrypted, and protected within the external environment it is located on.
O.DIG_VERIF	Use digital signatures to verify the authenticity of critical documents and transactions within the system.

The connection between threats and objectives and the rationale behind it.

THREATS	OBJECTIVES	RATIONALE
T. SPOOFING	O. AUTH_SEC	Multi-factor authentication (MFA) and strong authentication policies mitigate impersonation attacks and unauthorized access.
T. TAMPERING	O.DATA_INT	To prevent data tampering. The data is encrypted both during transmission and while stored in the database.
T. REPUDIATION	O.LOG_AUDIT	Detailed logs of user

		activities within the store will create an audit trail.
T.INFO_DISCLOSURE	O.DATA_ENC	Encrypting sensitive data ensures that even if intercepted, it remains unreadable to unauthorized parties.
T.DOS	O.NET_SEC	Suspicious or heavy traffic is detected and mitigated early to prevent successful attacks.
T. ELEVATION_PRIVILEGE	O. ACCESS_CTRL	Strict access controls and role-based permissions limit unauthorized privilege escalation attempts.
T. SESSION_HIJACK	O. SESSION_SEC	Secure session management practices such as token expiration and IP validation prevent session hijacking attacks.
T.SQL_INJ	O. VULN_MGMT	Input validation and parameterized queries prevent SQL Injection vulnerabilities.
T.XSS	O. VULN_MGMT	Implementing secure coding practices (e.g., escaping inputs, CSP) prevents Cross-Site Scripting attacks.
T. MALWARE_INJ	O. VULN_MGMT	Patching vulnerabilities within the system and its code, will ensure the TOE is regularly updated and will

		prevent malware from exploiting weakness within the system.
T. BRUTE_FORCE	O. AUTH_SEC	Enforcing account lockout policies and CAPTCHA mitigate brute-force attacks.
T. INSIDER_THREAT	O. ACCESS_CTRL	Digital signatures validate transaction authenticity, preventing fraudulent modifications.

Security functional requirements (SFR)

The security functional requirements (SFR) for the TOE that are focused on user data protection, identification, and authentication are as follows.

O. AUTH_SEC – Authentication Security

Clas: FIA – Identification and Authentication

Family: FIA_UAU – User Authentication

Components:

- FIA_UAU.1 Timing of authentication
- FIA_UAU.2 User authentication before any action
- FIA_UAU.3 Unforgeable authentication

Rationale: Ensures that only authenticated users can access the system, preventing spoofing attacks and unauthorized access.

O.DATA_ENC – Data Encryption

Class: FDP – User Data Protection

Family: FDP_UTC – Data Confidentiality

Components:

- FDP_UTC.1 Encryption of data in transit
- FDP_UCT.2 Encryption of stored data

Rationale: Protects sensitive information from unauthorized access or interception during transmission and storage.

O_ACCESS_CTRL – Access Control

Class: FMT -Security Management

Family: FMT_MSA – Security Attributes Management

Components:

- FMT_MSA.1 Management of security attributes
- FMT_MSA.3 Statistic attribute initialization

Rationale: Enforces role-based access control to restrict unauthorized users from accessing certain resources.

O. VULN_MGMT – Vulnerability Management

Class: FPT – Protection of the TOE Security Functions

Family: FPT_TST – Testing of TOE Security Functions

Components:

- FPT_TST.1 Testing of security functionality
- FPT_TST.2 Regular security patches and vulnerability scans

Rationale: Ensures continuous assessment of security weaknesses, preventing exploitation by attackers.

O.LOG_AUDIT – Loging and Auditing

Class: FAU – Security Audit

Family: FAU_GEN – Audit Data Generation

Components:

- FAU_GEN.1 Audit data generation
- FAU_GEN.2 User identity association
- FAU_SAR.1 Security audit review

Rationale: Enables tracking and forensic analysis of user activities, helping to detect and respond to security incidents.

O.NET_SEC – Network Security

Class: FCO - Communications

Family: FCO_NRO – Non-repudiation and Secure Communications

Components:

- FCO_NRO.1 Secure session establishment
- FCO_NRO.2 non-repudiation of origin

Rationale: Protects against man-in-the-middle (MITM) attacks and ensures integrity in communication between system components.

O.DATA_INT – Data Integrity

Class: FDP – User Data Protection

Family: FDP_SDI – Stored Data Integrity

Components:

- FDP_SDI.1 Ensuring data integrity in storage
- FDP_SDI.2 Tamper-detection mechanisms

Rationale: Prevents unauthorized modification of critical data, ensuring accuracy and trustworthiness.

O. SESSION_SEC – Secure Session Management

Class: FTP – Trusted Path/Channels

Family: FTP_ITC – Trusted Channel Communication

Components:

- FTP_ITC.1 Establishment of trusted communication paths
- FTP_ITC.2 Session token validation and expiration

Rationale: Prevents session hijacking, replay attacks, and unauthorized access due to session mismanagement.

O. BACKUP_REC – Backup and Recovery

Class: FPT – Protection of the TOE Security Functions

Family: FPT_RCV – Recovery Functionality

Components:

- FPT_RCV.1 Secure backup procedures
- FPT_RCV.2 Data recovery mechanism

Rationale: Ensures resilience against data loss by providing secure backup and recovery capabilities.

O.DIG_VERIF – Digital Verification

Class: FDP – User Data Protection

Family: FDP_DAU – Data Authentication

Components:

- FDP_DAU.1 Data authentication with identity of guarantor
- FDP_DAU.2 Digital signature validation

Rationale: Protects against data repudiation and ensures authenticity in transactions using cryptographic signatures.

Security Assurance Requirements (SARs) for TOE to enable EAL2 Certification

Assurance Class	Assurance Components
ADV: Development	**ADV.ARC.1 Security architecture description** Dependencies: ADV_FSP.1 Basic functional specification ADV_TDS.1 Basic design Developer action elements: ADV_ARC.1.1D The developer shall design and implement TOE so that the security features of the TSF cannot be bypassed. ADV_ARC.1.2D The developer shall design and implement TSF so that it is able to protect itself from tampering by untrusted active entities. ADV_ARC.1.3D The developer shall provide a security architecture description of the TSF. Content and presentation elements: ADV_ARC.1.1C The security architecture description shall be at a level of detail commensurate with the description of the SFR-enforcing abstractions described in the TOE design document. ADV_ARC.1.2C The security architecture description shall describe the security domains maintained by the TSF consistently with the SFRs. ADV_ARC.1.3C The security architecture description shall describe how the TFS initialization process is secure. ADV_ARC.1.4C The security architecture description shall demonstrate that the TSF protects itself from tampering.

	ADV_ARC.1.5C The security architecture description shall demonstrate that the TSF prevents bypass of the SFR-enforcing functionality. Evaluator action elements: ADV_ARC.1.1E The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.
	ADV_FSP.2 Security-enforcing functional specification Dependencies: AVD_TDS.1 Basic design Developer action elements: ADV_FSP.2.1D The developer shall provide a functional specification. ADV_FSP.2.2D The developer shall provide a tracing from the functional specification to the SFRs Content presentation elements ADV_FSP.2.1C The functional specification shall completely represent the TFS. ADV_FSP.2.2C The functional specification shall describe the purpose and method of use for all TSFI. ADV_FSP.2.3C The functional specification shall identify and describe all parameters associated with each TSFI. ADV_FSP.2.4C For each SFR-enforcing TSFI, the functional specification shall describe the SFR-enforcing actions associated with the TSFI. ADV_FSP.2.5C For each SFR-enforcing TSFI, the functional specification shall describe direct error messages resulting from processing associated with the SFR-enforcing actions. ADV_FSP.2.6C The tracing shall demonstrate that the SFRs trace to TSFIs in the functional specification. Evaluator action elements: ADV_FSP.2.1E The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

	ADV_FSP.2.2E The evaluator shall determine that the functional specification is an accurate and complete instantiation of the SFRs.
	ADV_TDS.1 Basic design Dependencies: ADV_FSP.2 Security-enforcing functional specification Developer action elements: ADV_TDS.1.1D The developer shall provide the design of the TOE. AVD_TDS.1.2D The developer shall provide a mapping from the TSFI of the functional specification to the lowest level of decomposition available in the TOE design. Content and presentation elements: ADV_TDS.1.1C The design shall describe the structure of the TOE in terms of subsystems. ADV.TDS.1.2C The design shall identify all subsystems of the TSF. ADV.TDS.1.3C The design shall describe the behaviour of each SFR-supporting or SFR-non-interfering TSF subsystem in sufficient detail to determine that it is not SFR-enforcing. ADV_TDS.1.4C The design shall summarise the SFR-enforcing behaviour of the SFR-enforcing subsystems. ADV_TDS.1.5C The design shall provide a description of the interactions among SFR-enforcing subsystems of the TSF, and between the SFR-enforcing subsystems of the TSF and other subsystems of the TSF. ADV_TDS.1.6C The mapping shall demonstrate that all TSFIs trace to the behaviour described in the TOE design that they invoke. Evaluator action elements: ADV_TSD.1.1E The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence. ADV_TDS.1.2E The evaluator shall determine that the design is an accurate and complete instantiation of all security functional requirements.

AGD: Guidance Documents	**AGD_OPE.1 Operational user guidance** Dependencies: ADV_FSP.1 Basic functional specification Developer action elements: AGD_OPE.1.1D The developer shall provide operational user guidance. Content and presentation elements: AGD_OPE.1.1C The operational user guidance shall describe, for each user role, the user-accessible functions and privileges that should be controlled in a secure processing environment, including appropriate warnings. AGD_OPE.1.2C The operational user guidance shall describe, for each user role, how to use the available interfaces provided by the TOE in a secure manner. AGD_OPE.1.3C The operational user guidance shall describe, for each user role, the available functions, and interfaces, in particular all security parameters under the control of the user, indicating secure values as appropriate. AGD_OPE.1.4C The operational user guidance shall, for each user role, clearly present each type of security-relevant event relative to the user-accessible functions that need to be performed, including changing the security characteristics of entities under the control of the TSF. AGD_OPE.1.5C The operational user guidance shall identify all possible modes of operation of the TOE (including operation following failure or operational error), their consequences and implications for maintaining secure operation. AGD_OPE.1.6C The operational user guidance shall, for each user role, describe the security measures to be followed in order to fulfil the security objectives for the operational environment as described in the ST. AGD_OPE.1.7C The operational user guidance shall be clear and reasonable. Evaluator action elements:

	AGD_OPE.1.1E The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.
	AGD_PRE.1 Preparative procedures Dependencies: No dependencies. Developer action elements: AGD_PRE.1.1D The developer shall provide the TOE including its preparative procedures. Content and presentation elements: AGD_PRE.1.1C The preparative procedures shall describe all the steps necessary for secure acceptance of the delivered TOE in accordance with the developer's delivery procedures. AGD_PRE.1.2C The preparative procedures shall describe all the steps necessary for secure installation of the TOE and for the secure preparation of the operational environment in accordance with the security objectives for the operational environment as described in the ST. Evaluator action elements: AGD_PRE.1.1E The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence. AGD_PRE.1.2E The evaluator shall apply the preparative procedures to confirm that the TOE can be prepared securely for operation.
ALC: Life-cycle support	**ALC_CMC.2 Use of a CM system** Dependencies: ALC_CMS.1 TOE CM Developer action elements: ALC_CMC.2.1D The developer shall provide the TOE and a reference for the TOE. ALC_CMC.2.2D The developer shall provide the CM documentation. ALC_CMC.2.3D The developer shall use a CM system. Content and presentation elements: ALC_CMC.2.1C The TOE shall be labelled with its unique reference.

	ALC_CMC.2.2C The CM documentation shall describe the method used to uniquely identify the configuration items. ALC_CMC.2.3C The CM system shall uniquely identify all configuration items. Evaluator action elements: ALC_CMC.2.1E The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.
	ALC_CMS.2 Parts of the TOE CM coverage Dependencies: No dependencies. Objectives 352 A CM system can control changes only to those items that have been placed under CM (i.e., the configuration items identified in the configuration list). Placing the TOE itself, the parts that comprise the TOE, and the evaluation evidence required by the other SARs under CM provides assurance that they have been modified in a controlled manner with proper authorisations. Application notes 353 ALC_CMS.2.1C introduces the requirement that the parts that comprise the TOE (all parts that are delivered to the consumer, for example hardware parts or executable files) be included in the configuration list and hence be subject to the CM requirements of CM capabilities (ALC_CMC). 354 ALC_CMS.2.3C introduces the requirement that the configuration lists indicate the developer of each TSF relevant configuration item. “Developer” here does not refer to a person, but to the organisation responsible for the development of the item. Developer action elements: ALC_CMS.2.1D The developer shall provide a configuration list for the TOE. Content and presentation elements:

	ALC_CMS.2.1C The configuration list shall include the following: the TOE itself; the evaluation evidence required by the SARs; and the parts that comprise the TOE. ALC_CMS.2.2C The configuration list shall uniquely identify the configuration items. ALC_CMS.2.3C For each TSF relevant configuration item, the configuration list shall indicate the developer of the item. Evaluator action elements: ALC_CMS.2.1E The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.
	ALC_DEL.1 Delivery procedures Dependencies: No dependencies. Developer action elements: ALC_DEL.1.1D The developer shall document and provide procedures for delivery of the TOE or parts of it to the consumer. ALC_DEL.1.2D The developer shall use the delivery procedures. Content and presentation elements: ALC_DEL.1.1C The delivery documentation shall describe all procedures that are necessary to maintain security when distributing versions of the TOE to the consumer. Evaluator action elements: ALC_DEL.1.1E The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.
ASE: Security Target Evaluation	**ASE_CCL.1 Conformance claims** Dependencies: ASE_INT.1 ST introduction ASE_ECD.1 Extended components definition ASE_REQ.1 Stated security requirements Developer action elements: ASE_CCL.1.1D The developer shall provide a conformance claim.

	ASE_CCL.1.2D The developer shall provide a conformance claim rationale. Content and presentation elements: ASE_CCL.1.1C The conformance claim shall contain a CC conformance claim that identifies the version of the CC to which the ST and the TOE claim conformance. ASE_CCL.1.2C The CC conformance claim shall describe the conformance of the ST to CC Part 2 as either CC Part 2 conformant or CC Part 2 extended. ASE_CCL.1.3C The CC conformance claim shall describe the conformance of the ST to CC Part 3 as either CC Part 3 conformant or CC Part 3 extended. ASE_CCL.1.4C The CC conformance claim shall be consistent with the extended components definition. ASE_CCL.1.5C The conformance claim shall identify all PPs and security requirement packages to which the ST claims conformance. ASE_CCL.1.6C The conformance claim shall describe any conformance of the ST to a package as either package-conformant or package augmented. ASE_CCL.1.7C The conformance claim rationale shall demonstrate that the TOE type is consistent with the TOE type in the PPs for which conformance is being claimed. ASE_CCL.1.8C The conformance claim rationale shall demonstrate that the statement of the security problem definition is consistent with the statement of the security problem definition in the PPs for which conformance is being claimed. AS_CCL.1.9C The conformance claim rationale shall demonstrate that the statement of security objectives is consistent with the statement of security objectives in the PPsfor which conformance is being claimed. ASE_CCL.1.10C The conformance claim rationale shall demonstrate that the statement of security requirements is consistent with the statement of security requirements in the PPs for which conformance is being claimed. Evaluator action elements:

	ASE_CCL.1.1E The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.
	ASE_ECD.1 Extended components definition Dependencies: No dependencies. Developer action elements: ASE_ECD.1.1D The developer shall provide a statement of security requirements. ASE_ECD.1.2D The developer shall provide an extended components definition. Content and presentation elements: ASE_ECD.1.1C The statement of security requirements shall identify all extended security requirements. ASE_ECD.1.2C The extended components definition shall define an extended component for each extended security requirement. ASE_ECD.1.3C The extended components definition shall describe how each extended component is related to the existing CC components, families, and classes. ASE_ECD.1.4C The extended components definition shall use the existing CC components, families, classes, and methodology as a model for presentation. ASE_ECD.1.5C The extended components shall consist of measurable and objective elements such that conformance or nonconformance to these elements can be demonstrated. Evaluator action elements: ASE_ECD.1.1E The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence. ASE_ECD.1.2E The evaluator shall confirm that no extended component can be clearly expressed using existing components.
	ASE_INT.1 ST Introduction Dependencies: No dependencies. Developer action elements:

	ASE_INT.1.1D The developer shall provide an ST introduction. Content and presentation elements: ASE_INT.1.1C The ST introduction shall contain an ST reference, a TOE reference, a TOE overview and a TOE description. ASE_INT.1.2C The ST reference shall uniquely identify the ST. ASE_INT.1.3C The TOE reference shall identify the TOE. ASE_INT.1.4C The TOE overview shall summarise the usage and major security features of the TOE.ASE_INT.1.5C The TOE overview shall identify the TOE type. ASE_INT.1.6C The TOE overview shall identify any non-TOE hardware/software/firmware required by the TOE. ASE_INT.1.7C The TOE description shall describe the physical scope of the TOE. ASE_INT.1.8C The TOE description shall describe the logical scope of the TOE. Evaluator action elements: ASE_INT.1.1E The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence. ASE_INT.1.2E The evaluator shall confirm that the TOE reference, the TOE overview, and the TOE description are consistent with each other.
	ASE_OBJ.2 Security objectives Dependencies: ASE_SPD.1 Security problem definition Developer action elements: ASE_OBJ.2.1D The developer shall provide a statement of security objectives. ASE_OBJ.2.2D The developer shall provide a security objectives rationale. Content and presentation elements: ASE_OBJ.2.1C The statement of security objectives shall

	describe the security objectives for the TOE and the security objectives for the operational environment. ASE_OBJ.2.2C The security objectives rationale shall trace each security objective for the TOE back to threats countered by that security objective and OSPs enforced by that security objective. ASE_OBJ.2.3C The security objectives rationale shall trace each security objective for the operational environment back to threats countered by that security objective, OSPs enforced by that security objective, and assumptions upheld by that security objective. ASE_OBJ.2.4C The security objectives rationale shall demonstrate that the security objectives counter all threats. ASE_OBJ.2.5C The security objectives rationale shall demonstrate that the security objectives enforce all OSPs. ASE_OBJ.2.6C The security objectives rationale shall demonstrate that the security objectives for the operational environments uphold all assumptions. Evaluator action elements: ASE_OBJ.2.1E The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.
	AES_REQ.2 Derived security requirements Dependencies: ASE_OBJ.2 Security objectives ASE_ECD.1 Extended components definition Developer action elements: ASE_REQ.2.1D The developer shall provide a statement of security requirements. ASE_REQ.2.2D The developer shall provide a security requirements rationale. Content and presentation elements: ASE_REQ.2.1C The statement of security requirements shall describe the SFRs and the SARs. ASE_REQ.2.2C All subjects, objects, operations, security

	attributes, external entities and other terms that are used in the SFRs and the SARs shall be defined. ASE_REQ.2.3C The statement of security requirements shall identify all operations on the security requirements. ASE_REQ.2.4C All operations shall be performed correctly. ASE_REQ.2.5C Each dependency of the security requirements shall either be satisfied, or the security requirements rationale shall justify the dependency not being satisfied. ASE_REQ.2.6C The security requirements rationale shall trace each SFR back to the security objectives for the TOE. ASE_REQ.2.7C The security requirements rationale shall demonstrate that the SFRs meet all security objectives for the TOE. ASE_REQ.2.8C The security requirements rationale shall explain why the SARs were chosen. ASE_REQ.2.9C The statement of security requirements shall be internally consistent. Evaluator action elements: ASE_REQ.2.1E The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.
	ASE_SPD.1 Security problem definition Dependencies: No dependencies. Developer action elements: ASE_SPD.1.1D The developer shall provide a security problem definition. Content and presentation elements: ASE_SPD.1.1C The security problem definition shall describe the threats. ASE_SPD.1.2C All threats shall be described in terms of a threat agent, an asset, and an adverse action. ASE_SPD.1.3C The security problem definition shall describe the OSPs.

	ASE_SPD.1.4C The security problem definition shall describe the assumptions about the operational environment of the TOE. Evaluator action elements: ASE_SPD.1.1E The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.
	ASE_TSS.1 TOE summary specification Dependencies: ASE_INT.1 ST introduction ASE_REQ.1 Stated security requirements ADV_FSP.1 Basic functional specification Developer action elements: ASE_TSS.1.1D The developer shall provide a TOE summary specification. Content and presentation elements: ASE_TSS.1.1C The TOE summary specification shall describe how the TOE meets each SFR. Evaluator action elements: ASE_TSS.1.1E The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence. ASE_TSS.1.2E The evaluator shall confirm that the TOE summary specification is consistent with the TOE overview and the TOE description.
ATE: Tests	**ATE_COV.1 Evidence of coverage** Dependencies: ADV_FSP.2 Security-enforcing functional specification ATE_FUN.1 Functional testing Objectives 407 The objective of this component is to establish that some of the TSFIs have been tested. Application notes 408 In this component the developer shows how tests in the test documentation corresponds to TSFIs in the

	functional specification. This can be achieved by a statement of correspondence, perhaps using a table. Developer action elements: ATE_COV.1.1D The developer shall provide evidence of the test coverage. Content and presentation elements: ATE_COV.1.1C The evidence of the test coverage shall show the correspondence between the tests in the test documentation and the TSFIs in the functional specification. Evaluator action elements: ATE_COV.1.1E The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.
	ATE_FUN.1 Functional testing Dependencies: ATE_COV.1 Evidence of coverage Objectives 430 The objective is for the developer to demonstrate that the tests in the test documentation are performed and documented correctly. Developer action elements: ATE_FUN.1.1D The developer shall test the TSF and document the results. ATE_FUN.1.2D The developer shall provide test documentation. Content and presentation elements: ATE_FUN.1.1C The test documentation shall consist of test plans, expected test results and actual test results. ATE_FUN.1.2C The test plans shall identify the tests to be performed and describe the scenarios for performing each test. These scenarios shall include any ordering dependencies on the results of other tests. ATE_FUN.1.3C The expected test results shall show the

	anticipated outputs from a successful execution of the tests. ATE_FUN.1.4C The actual test results shall be consistent with the expected test results. Evaluator action elements: ATE_FUN.1.1E The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.
	ATE_IND.2 Independent testing – sample Dependencies: ADV_FSP.2 Security-enforcing functional specification AGD_OPE.1 Operational user guidance AGD_PRE.1 Preparative procedures ATE_COV.1 Evidence of coverage ATE_FUN.1 Functional testing Objectives 444 In this component, the objective is to demonstrate that the TOE operates in accordance with its design representations and guidance documents. Evaluator testing confirms that the developer performed some tests of some interfaces in the functional specification. Application notes 445 The intent is that the developer should provide the evaluator with materials necessary for the efficient reproduction of developer tests. This may include such things as machine-readable test documentation, test programs, etc. 446 This component contains a requirement that the evaluator has available test results from the developer to supplement the programme of testing. The evaluator will repeat a sample of the developer's tests to gain confidence in the results obtained. Having established such confidence the evaluator will build upon the developer's testing by conducting additional tests that exercise the TOE in a different manner. By using a platform of validated developer test results the evaluator is able to gain confidence that the

	TOE operates correctly in a wider range of conditions than would be possible purely using the developer's own efforts, given a fixed level of resource. Having gained confidence that the developer has tested the TOE, the evaluator will also have more freedom, where appropriate, to concentrate testing in areas where examination of documentation or specialist knowledge has raised particular concerns. Developer action elements: ATE_IND.2.1D The developer shall provide the TOE for testing. Content and presentation elements: ATE_IND.2.1C The TOE shall be suitable for testing. ATE_IND.2.2C The developer shall provide an equivalent set of resources to those that were used in the developer's functional testing of the TSF. Evaluator action elements: ATE_IND.2.1E The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence. ATE_IND.2.2E The evaluator shall execute a sample of tests in the test documentation to verify the developer test results. ATE_IND.2.3E The evaluator shall test a subset of the TSF to confirm that the TSF operates as specified
AVA: Vulnerability Assessment	**AVA_VAN.2 Vulnerability analysis** Dependencies: ADV_ARC.1 Security architecture description ADV_FSP.2 Security-enforcing functional specification ADV_TDS.1 Basic design AGD_OPE.1 Operational user guidance AGD_PRE.1 Preparative procedures Objectives 460 A vulnerability analysis is performed by the evaluator to ascertain the presence of potential vulnerabilities.

	461 The evaluator performs penetration testing, to confirm that the potential vulnerabilities cannot be exploited in the operational environment for the TOE. Penetration testing is performed by the evaluator assuming an attack potential of Basic. Developer action elements: AVA_VAN.2.1D The developer shall provide the TOE for testing. Content and presentation elements: AVA_VAN.2.1C The TOE shall be suitable for testing. Evaluator action elements: AVA_VAN.2.1E The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence. AVA_VAN.2.2E The evaluator shall perform a search of public domain sources to identify potential vulnerabilities in the TOE. AVA_VAN.2.3E The evaluator shall perform an independent vulnerability analysis of the TOE using the guidance documentation, functional specification, TOE design and security architecture description to identify potential vulnerabilities in the TOE. AVA_VAN.2.4E The evaluator shall conduct penetration testing, based on the identified potential vulnerabilities, to determine that the TOE is resistant to attacks performed by an attacker possessing Basic attack potential.

References:

Common Criteria (2012a) *Common Criteria for Information Technology Security Evaluation, Part 1: Introduction and General Model*. Version 3.1, Revision 4. [PDF] Common Criteria Recognition Arrangement (CCRA).

- Common Criteria (2012b) *Common Criteria for Information Technology Security Evaluation, Part 2: Security Functional Components*. Version 3.1, Revision 4. [PDF] Common Criteria Recognition Arrangement (CCRA).

- Common Criteria (2012c) *Common Criteria for Information Technology Security Evaluation, Part 3: Security Assurance Components*. Version 3.1, Revision 4. [PDF] Common Criteria Recognition Arrangement (CCRA).

Ncsc.gov.uk. (2024). *Denial of Service (DoS) guidance*. [online] Available at: https://www.ncsc.gov.uk/collection/denial-service-dos-guidance collection#:~:text="Denial of service" or ",frequently reported by the media. [Accessed 20 Mar. 2025].

MySQL / System Architecture:

Oracle (2024) *MySQL 8.0 Reference Manual*. [online] Available at: https://dev.mysql.com/doc/refman/8.0/en/ [Accessed 20 Mar. 2025].

Mozilla Developer Network (MDN) (2024) *Web security overview*. [online] Available at: https://developer.mozilla.org/en-US/docs/Web/Security [Accessed 15 Mar. 2025].

Stride Framework:

Microsoft (2005) *The STRIDE Threat Model*. [online] Available at: https://docs.microsoft.com/en-us/security/engineering/stride-threat-modeling [Accessed 20 Mar. 2025].

General:

Stallings, W. (2017) *Computer Security: Principles and Practice*. 4th ed. Pearson.

Anderson, R. (2020) *Security Engineering: A Guide to Building Dependable Distributed Systems*. 3rd ed. Wiley.