
# Supporting Document
# Mandatory Technical Document

Evaluation Activities for the collaborative Protection Profile for Database Management Systems

12 June 2020

Version 1.0

# Foreword

This is a supporting document, intended to complement the Common Criteria version 3 and the associated Common Evaluation Methodology for Information Technology Security Evaluation.

Supporting documents may be “Guidance Documents”, that highlight specific approaches and application of the standard to areas where no mutual recognition of its application is required, and as such, are not of normative nature, or “Mandatory Technical Documents”, whose application is mandatory for evaluations whose scope is covered by that of the supporting document. The usage of the latter class is not only mandatory, but certificates issued as a result of their application are recognized under the Common Criteria Recognition Arrangement (CCRA).

This supporting document has been developed by the Database Management System international Technical Community (DBMS-iTC) and is designed to be used to support the evaluations of products against the collaborative Protection Profiles (cPPs) identified in Section 1.1.

**Technical Editor:** Database Management System (DBMS) international Technical Community (iTC)

**Document history:**

| Version | Date | Description |
|---|---|---|
| V0.01 | July 16th, 2019 | Initial release for DBMS-iTC use |
| V0.02 | February 20th, 2019 | Updated with some Evaluation Activities |
| V0.03 | March 8th, 2019 | Updated after iTC face to face meeting |
| V0.04-0.16 | | Interim updates after iTC meetings |
| V0.17 | 7 April 2020 | Changes accepted |
| 1.0 | 16 June 2020 | Initial Release |

**General Purpose:** See Section 1.1.

**Field of special use:** This Supporting Document applies to the evaluation of TOEs claiming conformance with the collaborative Protection Profile (cPP) for Database Management Systems [DBMScPP].

**Acknowledgements:**

This Supporting Document was developed by the Database Management System international Technical Community (DBMS-iTC) with representatives from industry, Government agencies, and Common Criteria Test Laboratories.

# Contents

## Figures / Tables

# 1. Introduction

## 1.1 Technology Area and Scope of Supporting Document

This Supporting Document (SD) defines the Evaluation Activities associated with the collaborative Protection Profile for Database Management Systems [DBMScPP].

This Supporting Document is mandatory for evaluations of products that claim conformance to the following cPP:

a) collaborative Protection Profile for Database Management Systems [DBMScPP]

Although Evaluation Activities (EA) are defined for the evaluators to follow, the definitions in this Supporting Document aim to provide a common understanding for developers, evaluators and users as to what aspects of the Target of Evaluation (TOE) are tested in an evaluation against the associated cPP, and to what depth the testing is carried out.

This common understanding contributes to the goal of ensuring that evaluations against the cPP achieve comparable, transparent and repeatable results. In general, the definition of Evaluation Activities will also help Developers to prepare for evaluation by identifying specific requirements for their TOE. The specific requirements in Evaluation Activities may in some cases clarify the meaning of Security Functional Requirements (SFRs), and may identify particular requirements for the content of Security Targets (ST), especially the TOE Summary Specification (TSS), user guidance documentation and testing activities.

## 1.2 Structure of the Document

EAs can be defined for both SFRs and Security Assurance Requirements (SAR). These are defined in separate sections of this Supporting Document.

If any EA cannot be successfully completed in an evaluation, then the overall verdict for the evaluation is a ‘fail’. In rare cases there may be acceptable reasons why an EA may be modified or deemed not applicable for a particular TOE, but this must be agreed with the Certification Body (CB) for the evaluation.

In general, if all EAs (for both SFRs and Security Assurance Requirements (SARs)) are successfully completed in an evaluation then it would be expected that the overall verdict for the evaluation is a ‘pass’. To reach a ‘fail’ verdict when the EAs have been successfully completed would require a specific justification from the evaluator as to why the Evaluation Activities were not sufficient for that TOE.

Similarly, at the more granular level of Assurance Components, if the EAs for an Assurance Component and all of its related SFR EAs are successfully completed in an evaluation then it would be expected that the verdict for the Assurance Component is a ‘pass’. To reach a ‘fail’ verdict for the Assurance Component when these EAs have been successfully completed would require a specific justification from the evaluator as to why the EAs were not sufficient for that TOE.

## 1.3 Application of this Supporting Document

This Supporting Document (SD) defines three types of EAs; TSS, Guidance Documentation, and Tests are designed to be used in conjunction with cPPs. cPPs that rely on this SD will explicitly identify this document as a source for the EAs. Each security requirement (SFR or SAR) specified in the cPP could have multiple associated EAs. The security requirement naming convention is consistent between the cPP and SD ensuring a clear one to one correspondence between security requirements and EAs.

The cPP and SD are designed to be used in conjunction with each other, where the cPP lists SFRs and SARs and the SD catalogues EAs associated with each SFR and SAR. Some of the SFRs included in the cPP are optional. Therefore, an ST claiming conformance to the cPP does not necessarily have to include all possible SFRs defined in the cPP.

In an ST conformant to the cPP, several operations need to be performed (mainly selections and assignments). Some EAs define separate actions for different selected or assigned values in SFRs. The evaluator shall neither carry out EAs related to SFRs that are not claimed in the ST nor EAs related to specific selected or assigned values that are not claimed in the ST.

EAs do not necessarily have to be executed independently from each other. A description in a guidance documentation or one test case, for example, can cover multiple EAs at a time, no matter whether the EAs are related to the same or different SFRs.

# 2. Evaluation Activities for SFRs

## 2.1 Class: Security Audit (FAU)

### FAU_GEN.1 Audit data generation

**TSS**

The list of auditable events is included in FAU_GEN.1. No further TSS activities are defined.

**Guidance Documentation**

The evaluator shall check the guidance documentation to ensure that, as a minimum, the auditable events specified in FAU_GEN.1 are listed and the associated information recorded is consistent with the definition of the SFRs.

**Tests**

For the events listed in the table of audit events in the ST, the evaluator shall verify the TOE's ability to correctly generate audit records and that the associated information required by the ST is included in the audit record.

Note that the testing here may be accomplished in conjunction with the testing of the security mechanisms.

### FAU_GEN.2 User identity association

**TSS**

See FAU_GEN.1

**Guidance Documentation**

See FAU_GEN.1

**Tests**

This activity is accomplished in conjunction with the testing of FAU_GEN.1.1.

### FAU_SEL.1 Selective audit

**TSS**

The evaluator shall examine the TSS to verify that it identifies the attributes by which the TOE can be configured to selectively enable or disable the generation of auditable events.

**Guidance Documentation**

The evaluator shall examine the operational guidance to verify that it provides a list of the attributes that can be used to selectively enable or disable the generation of auditable events as well as instructions for performing this operation.

**Tests**

i. The evaluator shall generate audit records for each attribute specified in FAU_SEL.1.

ii. The evaluator shall log on to the TOE using a role that is sufficiently privileged to modify the set of events that the TOE audits, and select auditable events for each attribute specified by FAU_SEL.1 in the ST, including any attribute included in the assignment. This shall be done for each attribute separately and a combination of two or more of the attributes.

iii. The evaluator shall then:

   a. Verify that audit logs are generated for the auditable events that have been selected;

   b. Verify that audit logs are not generated for the auditable events that are not selected.

NOTE: The following testing may be done in conjunction with other assurance activities since auditable events occur as a by-product of the TOE being used to perform other security functions.

## 2.2 Class: User Data Protection (FDP)

### FDP_ACC.1 Subset access control

**TSS**

The TSS evaluation activities are included in the FDP_ACF.1.

**Guidance Documentation**

The Guidance evaluation activities are included in the FDP_ACF.1.

**Tests**

The test evaluation activities are included in the FDP_ACF.1.

### FDP_ACF.1 Security attribute based access control

**TSS**

The evaluator shall examine the TSS and verify that an explanation of the discretionary access control policy is given, and that the explanation is both clear and understandable.

**Guidance Documentation**

The evaluator shall examine the guidance to verify that it:

- Clearly states the access control rules of the TOE;
- Explains how the security and object attributes are used by the TOE in order to achieve the desired access control;
- Instructs administrators on how to allow users access to objects using any additional rules defined in FDP_ACF.1.3; and
- Instructs administrators on how to deny users access to objects using any additional rules defined in FDP_ACF.1.4.

**Tests**

The evaluator shall devise tests that exercise each of the access control rules.

NOTE: It is not necessary to test every combination of the rules, but each rule must be included at least once in the test cases.

### FDP_RIP.1 Subset residual information protection

**TSS**

The evaluator shall examine the TSS to ensure that, at a minimum, it describes how the previous information content is made unavailable.

**Guidance Documentation**

There are no AGD assurance activities for this requirement beyond what is necessary to satisfy the requirements in [CEM].

**Tests**

There are no ATE assurance activities for this requirement beyond what is necessary to satisfy the requirements in [CEM].

## 2.3 Class: Identification and authentication (FIA)

### FIA_ATD.1 User attribute definition

**TSS**

The evaluator shall check to ensure that the TSS contains a description of the user security attributes that the TOE uses to implement the SFR, which is consistent with the definition of the SFR.

**Guidance Documentation**

There are no AGD assurance activities for this requirement beyond what is necessary to satisfy the requirements in [CEM].

**Tests**

There are no ATE assurance activities for this requirement beyond what is necessary to satisfy the requirements in [CEM].

### FIA_UAU.2 User authentication before any action

**TSS**

There are no ASE assurance activities for this requirement beyond what is necessary to satisfy the requirements in [CEM].

**Guidance Documentation**

There are no AGD assurance activities for this requirement beyond what is necessary to satisfy the requirements in [CEM].

**Tests**

The evaluator shall examine the guidance documentation to ensure that no TOE Security Functionality (TSF) mediated actions are available before user identification and authentication is completed.

### FIA_UID.2 User identification before any action

**TSS**

There are no ASE assurance activities for this requirement beyond what is necessary to satisfy the requirements in [CEM].

**Guidance Documentation**

There are no AGD assurance activities for this requirement beyond what is necessary to satisfy the requirements in [CEM].

**Tests**

Testing is performed in conjunction with FIA_UAU.2.

## 2.4 Class: Security Management (FMT)

### FMT_MSA.1 Management of security attributes

**TSS**

The evaluator shall verify that the TSS contains a description of all of the security attributes in the discretionary access control policy that can be managed by authorized administrators. The evaluator shall also verify that the TSS describes how these security attributes are protected from unauthorized access.

The evaluator shall verify that the description of security attributes includes all of those given in FIA_ATD.1.

**Guidance Documentation**

The evaluator shall verify that the guidance contains a description of the management functionality associated with security attributes.

**Tests**

The evaluator shall log on as an authorized administrator and perform allowed operations on the security attributes. The evaluator shall verify that the operations are performed as expected.

The evaluator shall log on as user without the appropriate privileges and attempt to perform administrator-allowed operations on the security attributes. The evaluator shall verify that the operations are not permitted.

### FMT_MSA.3 Static attribute initialization

**TSS**

The evaluator shall verify that the TSS describes the mechanisms to generate top level security attributes and their default values.

**Guidance Documentation**

The evaluator shall examine the guidance and verify that no ability to specify alternative initial values as an override to the default values is found.

**Tests**

The evaluator shall create at least one new container object (e.g. a table) at the top-level. The evaluator shall check that the attributes of the container object has the default value(s) described in the TSS values.

The evaluator shall create new lower-level objects (e.g. rows, cells). The evaluator shall check that the attributes of the lower-level object(s) have the same default permissions as the higher-level object.

### FMT_MTD.1 Management of TSF data

**TSS**

This was performed in conjunction with FAU_SEL.1.

**Guidance Documentation**

This was performed in conjunction with FAU_SEL.1.

**Tests**

Testing is performed in conjunction with FAU_SEL.1.

### FMT_REV.1(1) Revocation

**TSS**

The evaluator shall examine the TSS to verify that it defines the revocation rules associated with user security attributes and that the revocation rules are sufficiently described in informal language.

The evaluator shall examine the TSS to verify that the timing and/or conditions of revocation is specified.

**Guidance Documentation**

The evaluator shall examine the guidance documentation to verify that the user security attribute revocation rules are adequately described to the authorized administrator.

**Tests**

i. The evaluator shall log on as a user and verify that the user is able to perform actions in accordance with the user security attributes, specified in FMT_REV.1.1(1). If revocation is effective at the next log on then the user shall log off.

ii. The evaluator shall log on as an authorized administrator and revoke user security attribute(s) in accordance with the guidance.

iii. The evaluator shall verify that the user is no longer able to perform actions in accordance with the revoked user security attributes.
NOTE: any consideration of the time for the revocation to be effective shall be considered appropriately by the evaluator before completing (iii).

NOTE: In the steps above the term “user” implies the same user throughout the test.

## FMT_REV.1(2) Revocation (DAC)

### TSS

The evaluator shall examine the TSS to verify that it defines the revocation rules associated with object security attributes and that the revocation rules are sufficiently described in informal language.

The evaluator shall examine the TSS to verify that the timing and/or conditions of revocation is specified.

### Guidance Documentation

The evaluator shall examine the guidance documentation to verify that the object security attribute revocation rules are adequately described.

### Tests

i. The evaluator shall log on as a user with sufficient privileges to objects and verify that the user is able to perform actions on objects in accordance with the object security attributes, specified in FMT_REV.1.1(2).

ii. The evaluator shall log on as a database user with sufficient privileges as allowed by the DAC policy and revoke object security attribute(s) in accordance with the guidance.

iii. The evaluator shall verify that the user is no longer able to perform actions in accordance with the revoked object security attributes.

NOTE: Any consideration of the time for the revocation to be effective shall be considered appropriately by the evaluator before completing (iii).

NOTE: In the steps above the term “user” implies the same user throughout the test.

## FMT_SMF.1 Specification of Management Functions

### TSS

The evaluator shall examine the TSS and verify that the management functions listed in FMT_SMF.1 are described in informal language.

### Guidance Documentation

The evaluator shall examine the guidance documentation to ensure that there is appropriate guidance for configuring and using all of the management functions listed in FMT_SMF.1.

### Tests

The evaluator shall devise and execute tests for each of the management functions listed in FMT_SMF.1.

NOTE: If management functions have already been tested in conjunction with other SFRs in the ST then it is not necessary to repeat the testing for this evaluation activity.

**FMT_SMR.1 Security roles**

**TSS**

The evaluator shall examine the TSS to verify that it provides a description of all of the roles listed in FMT_SMR.1.1

**Guidance Documentation**

The evaluator shall review the operational guidance in order to verify that it discusses the listed administrative role(s), the privileges associated with each role, and how users are associated with each role.

**Tests**

The evaluator shall associate a user with each of the listed roles and verify that the user privileges are consistent with the descriptions in the TSS.

TOE Access (FTA)

**FTA_MCS.1 Basic limitation on multiple concurrent sessions**

**TSS**

The evaluator shall examine the TSS and verify that it states the default number of concurrent sessions per user for the evaluated configuration. If the default number of concurrent sessions can be changed then the evaluator should verify that the TSS states that the default can be changed.

**Guidance Documentation**

The evaluator shall examine the guidance documentation and verify that it states how the default number of sessions per user is set and, if applicable, how the default can be changed.

**Tests**

The evaluator shall establish the maximum number of concurrent sessions and verify that this number of concurrent sessions is allowed. The evaluator shall attempt to establish a number of sessions greater than the maximum specified and verify that additional concurrent sessions cannot be established.

If the default number of concurrent sessions can be changed then the evaluator shall change the default value and repeat the test.

### FTA_TSE.1 TOE session establishment

**TSS**

The evaluator shall examine the TSS and verify that the attributes that can be used to deny session establishment are listed and described.

**Guidance Documentation**

The evaluator shall examine the guidance documentation and verify that a description of how denial of session establishment is configured is included.

**Tests**

For each of the listed attributes used for denial of session establishment, the evaluator shall use the guidance documentation to configure the TSF to deny session establishment using that attribute. The evaluator shall verify that session establishment is denied appropriately in each case.

# 3. Evaluation Activities for Optional SFRs

These activities are only required when the optional SFRs are claimed.

## 3.1 Class: Identification and Authentication (FIA)

### FIA_USB_EXT.2 Enhanced user-subject binding

**TSS**

The evaluator shall check to ensure that the TSS contains a description of rules for the assignment of security attributes associated with the users to the subjects, the rules for the initial association of attributes, and how the rules are enforced.

**Guidance Documentation**

There are no AGD assurance activities for this requirement beyond what is necessary to satisfy the requirements in [CEM].

**Tests**

The evaluator shall verify the association of security attributes to subjects by establishing a user with a set of security attributes, changing the attributes and verifying that the new attributes result in the expected change. If there are any additional rules in FIA_USB_EXT.2.2, FIA_USB_EXT.2.3 or FIA_USB_EXT.2.4, the evaluator must perform a test to demonstrate that each rule holds true. Where practical and appropriate for the rule, the evaluator must also perform a negative test that demonstrates the rule being enforced.

## 3.2 Class: Protection of the TSF (FPT)

### FPT_TRC.1 Internal TSF consistency

**TSS**

The evaluator shall examine the TSS and verify that it includes a description of how data is replicated between physically separated parts of the TOE and how consistency between the TOE Security Functionality (TSF) data in the parts is achieved. The description shall include how any TSF data inconsistencies are corrected without undue delay.

**Guidance Documentation.**

The evaluator shall examine the guidance documentation and verify that necessary instructions on how to properly configure the TOE for replication are included.

**Tests**

The evaluator shall configure the replication of a TOE with physically separated parts. The evaluator shall compare the TSF data in each part of the TOE and verify that they are consistent. The evaluator shall take into consideration any expected differences that are described in the TSS.

NOTE: This could be achieved through appropriate sampling of the TSF data on each part of the TOE.

## 3.3 Class: TOE access (FTA)

### FTA_TAH_EXT.1 TOE access information

**TSS**

The evaluator shall examine the guidance documentation to verify that a statement is included in regard to whether configuration of this function is needed.

**Guidance Documentation**

The evaluator shall examine the guidance documentation to verify that configuration information is included if indicated in the TSS.

The evaluator shall verify that the guidance documentation includes information in regard to how a user retrieves the information required in the FTA_TAH_EXT.1.

**Tests**

Test 1: The evaluator shall follow the guidance documentation instructions for retrieving:

a) The date and time of the session establishment attempt of the user, and
b) The incremental count of successive unsuccessful session establishment,

and verify that it can be retrieved and that the information is correct.

Test 2: The evaluator shall assume a user role and verify that the following information can be retrieved by following the instructions given in the guidance documentation.

a) The previous last successful session establishment, and
b) The last unsuccessful attempt to session establishment and the number of unsuccessful attempts since the previous last successful session establishment.

The evaluator shall verify that users can only access their own information.

# 4. Evaluation Activities for SARs

In order to meet the goals of the evaluation, some of the [CEM] work units have been refined. Otherwise, the evaluator shall perform the CEM activity as specified.

## 4.1 ADV: Development

**Security architecture description (ADV_ARC.1)**

In order to meet these goals some refinement of the ADV_ARC.1 [CEM] work units is needed. The following table indicates, for each work unit in ADV_ARC.1, whether the [CEM] work unit is to be performed as written, or if it has been clarified by an Evaluation Activity. If clarification has been provided, a reference to this clarification is provided in the table.

| [CEM] ADV_ARC.1 Work Units | Evaluation Activities |
|---|---|
| ADV_ARC.1-1 The evaluator ***shall examine*** the security architecture description to determine that the information provided in the evidence is presented at a level of detail commensurate with the descriptions of the SFR-enforcing abstractions contained in the functional specification and TOE design document. | The evaluator shall perform the [CEM] activity as specified. |
| ADV_ARC.1-2 The evaluator ***shall examine*** the security architecture description to determine that it describes the security domains maintained by the TSF. | The evaluator shall perform the [CEM] activity as specified. |
| ADV_ARC.1-3 The evaluator ***shall examine*** the security architecture description to determine that the initialisation process preserves security. | The evaluator shall perform the [CEM] activity as specified. |
| ADV_ARC.1-4 The evaluator ***shall examine*** the security architecture description to determine that it contains information sufficient to support a determination that the TSF is able to protect itself from tampering by untrusted active | The evaluator shall perform the [CEM] activity as specified. |

| [CEM] ADV_ARC.1 Work Units | Evaluation Activities |
|---|---|
| entities. | |
| ADV_ARC.1-5 The evaluator ***shall examine*** the security architecture description to determine that it presents an analysis that adequately describes how the SFR-enforcing mechanisms cannot be bypassed. | The evaluator shall verify that the evidence indicates whether or not the TOE dynamically creates Structured Query Language (SQL) code, or another query language code for databases that do not use SQL, using supplied input. If dynamic code is used, the evaluator shall verify that the evidence describes the mechanisms that have been implemented to prevent or to mitigate the possibility of SQL injection using dynamic code. (e.g. prepared statements, filtering mechanisms, privilege reduction). |

***Table 1: Mapping of ADV_ARC.1 [CEM] Work Units to Evaluation Activities***

**Security-enforcing functional specification (ADV_FSP.2)**

The evaluator shall perform the [CEM] activity as specified for ADV_FSP.2.

**Basic Design (ADV_TDS.1)**

The evaluator shall perform the [CEM] activity as specified for ADV_TDS.1.

### 4.2 AGD: Guidance Documentation

**Operational User Guidance (AGD_OPE.1)**

Specific requirements and checks on the user guidance documentation are identified (where relevant) in the individual Evaluation Activities for each SFR. Additionally, the evaluator is expected to ensure that the [CEM] requirements of AGD_OPE.1 [CEM] are met.

**Preparative Procedures (AGD_PRE.1)**

Specific requirements and checks on the user guidance documentation are identified (where relevant) in the individual Evaluation Activities for each SFR. Additionally, the evaluator is expected to ensure that the [CEM] requirements of AGD_OPE.1 [CEM] are met.

### 4.3 Class ALC: Life-cycle Support

**Use of a CM System (ALC_CMC.2)**

The evaluator shall perform the [CEM] activity as specified for ALC_CMC.2.

### Parts of the TOE CM Coverage (ALC_CMS.2)

The evaluator shall perform the [CEM] activity as specified for ALC_CMS.2.

### Delivery Procedures (ALC_DEL.1)

The evaluator shall perform the [CEM] activity as specified for ALC_DEL.2.

### Systematic Flaw Remediation (ALC_FLR.3)

A DBMS is often a key component in a larger infrastructure. Therefore, the response to potential security flaws must be clearly established, and comprehensive. There must be a means of providing information and solutions to users in a timely manner, using automated means. ALC_FLR.3 has been mandated to meet these requirements.

The following table indicates, for each work unit in ALC_FLR.3, whether the [CEM] work unit is to be performed as written, or if it has been clarified by an Evaluation Activity. If clarification has been provided, a reference to this clarification is provided in the table.

| [CEM] ALC_FLR.3 Work Units | Evaluation Activities |
|---|---|
| ALC_FLR.3-1 The evaluator ***shall examine*** the flaw remediation procedures documentation to determine that it describes the procedures used to track all reported security flaws in each release of the TOE. | The evaluator shall perform the [CEM] activity as specified. |
| ALC_FLR.3-2 The evaluator ***shall examine*** the flaw remediation procedures to determine that the application of these procedures would produce a description of each security flaw in terms of its nature and effects. | The evaluator shall perform the [CEM] activity as specified. |
| ALC_FLR.3-3 The evaluator ***shall examine*** the flaw remediation procedures to determine that the application of these procedures would identify the status of finding a correction to each security flaw. | The evaluator shall perform the [CEM] activity as specified. |

| [CEM] ALC_FLR.3 Work Units | Evaluation Activities |
|---|---|
| ALC_FLR.3-4 The evaluator ***shall check*** the flaw remediation procedures to determine that the application of these procedures would identify the corrective action for each security flaw. | The evaluator shall perform the [CEM] activity as specified. |
| ALC_FLR.3-5 The evaluator ***shall examine*** the flaw remediation procedures documentation to determine that it describes a means of providing the TOE users with the necessary information on each security flaw. | The evaluator shall perform the [CEM] activity as specified. |
| ALC_FLR.3-6 The evaluator ***shall examine*** the flaw remediation procedures to determine that the application of these procedures would result in a means for the developer to receive from TOE user reports of suspected security flaws or requests for corrections to such flaws. | The evaluator shall perform the [CEM] activity as specified. |
| ALC_FLR.3-7 The evaluator ***shall examine*** the flaw remediation procedures to determine that the application of these procedures would result in a timely means of providing the registered TOE users who might be affected with reports about, and associated corrections to, each security flaw. | The evaluator shall perform the [CEM] activity as specified. The evaluator must ensure that the vendor has a defined set of timeframes for response to vulnerabilities. The evaluator must ensure that the vendor has rationale for those timeframes. |
| ALC_FLR.3-8 The evaluator ***shall examine*** the flaw remediation procedures to determine that the application of these procedures would result in automatic distribution of the reports and associated corrections to the registered | The evaluator shall perform the [CEM] activity as specified. |

| [CEM] ALC_FLR.3 Work Units | Evaluation Activities |
|---|---|
| TOE users who might be affected. | |
| ALC_FLR.3-9 The evaluator ***shall examine*** the flaw remediation procedures to determine that the application of these procedures would help to ensure that every reported flaw is corrected. | The evaluator shall perform the [CEM] activity as specified. |
| ALC_FLR.3-10 The evaluator ***shall examine*** the flaw remediation procedures to determine that the application of these procedures would help to ensure that the TOE users are issued remediation procedures for each security flaw. | The evaluator shall perform the [CEM] activity as specified. |
| ALC_FLR.3-11 The evaluator ***shall examine*** the flaw remediation procedures to determine that the application of these procedures would result in safeguards that the potential correction contains no adverse effects. | The evaluator shall perform the [CEM] activity as specified. |
| ALC_FLR.3-12 The evaluator ***shall examine*** the flaw remediation guidance to determine that the application of these procedures would result in a means for the TOE user to provide reports of suspected security flaws or requests for corrections to such flaws. | The evaluator shall perform the [CEM] activity as specified. |
| ALC_FLR.3-13 The evaluator ***shall examine*** the flaw remediation guidance to determine that it describes a means of enabling the TOE users to register with the developer. | The evaluator shall perform the [CEM] activity as specified. |

| [CEM] ALC_FLR.3 Work Units | Evaluation Activities |
|---|---|
| ALC_FLR.3-14 The evaluator ***shall examine*** the flaw remediation guidance to determine that it identifies specific points of contact for user reports and enquiries about security issues involving the TOE. | The evaluator shall perform the [CEM] activity as specified. |

***Table 2: Mapping of ALC_FLR.3 [CEM] Work Units to Evaluation Activities***

### 4.4 Class ASE: Security Target Evaluation

When evaluating a Security Target, the evaluator performs the work units as presented in the CEM. In addition, the evaluator ensures the content of the TSS in the ST satisfies the EAs specified in Section 2 (Evaluation Activities for SFRs) and Section 3 (Evaluation Activities for Optional SFRs).

### 4.5 Class ATE: Tests

**Evidence of Coverage (ATE_COV.1)**

The developer is expected to provide evidence of functional testing of the DBMS, at a level consistent with ATE_COV.1.

**Functional Testing (ATE_FUN.1)**

The developer is expected to provide evidence of functional testing of the DBMS, at a level consistent with ATE_FUN.1. Automated testing may be used in whole or in part to satisfy the developer test requirements.

**Independent Testing (ATE_IND.2)**

Testing is performed to confirm the functionality described in the TSS, and that this functionality can be exercised in accordance with the guidance documentation. The focus of the testing is to confirm that the requirements specified in the SFRs are being met. The Evaluation Activities within this document identify the specific testing activities necessary to verify compliance with the SFRs. The evaluator must produce a test report documenting the plan for and results of testing. The test report must also ensure that all the requirements of ATE_IND.2 have been met, as noted below.

| [CEM] ATE_IND.2 Work Units | Evaluation Activities |
|---|---|
| ATE_IND.2-1 The evaluator ***shall examine*** the TOE to determine that the test configuration is consistent with the configuration under evaluation as specified in the | The evaluator shall perform the [CEM] activity as specified. |

| [CEM] ATE_IND.2 Work Units | Evaluation Activities |
|---|---|
| ST. | |
| ATE_IND.2-2 The evaluator ***shall examine*** the TOE to determine that it has been installed properly and is in a known state. | The evaluator shall perform the [CEM] activity as specified. |
| ATE_IND.2-3 The evaluator ***shall examine*** the set of resources provided by the developer to determine that they are equivalent to the set of resources used by the developer to functionally test the TSF. | The evaluator shall perform the [CEM] activity as specified. |
| ATE_IND.2-4 The evaluator ***shall conduct*** testing using a sample of tests found in the developer test plan and procedures. | The evaluator shall perform the [CEM] activity as specified. Each of the TSFIs must be exercised. |
| ATE_IND.2-5 The evaluator ***shall check*** that all the actual test results are consistent with the expected test results. | The evaluator shall perform the [CEM] activity as specified. |
| ATE_IND.2-6 The evaluator ***shall devise*** a test subset. | The test subset shall be comprised of a sample of the developer test cases plus all of the Test EAs noted within this document. This does not preclude the evaluators from adding their own tests. |
| ATE_IND.2-7 The evaluator ***shall produce*** test documentation for the test subset that is sufficiently detailed to enable the tests to be reproducible. | The evaluator shall perform the [CEM] activity as specified. |
| ATE_IND.2-8 The evaluator ***shall conduct*** testing. | The evaluator shall perform the [CEM] activity as specified. |
| ATE_IND.2-9 The evaluator ***shall record*** the following information about the tests that | The evaluator shall perform the [CEM] activity as specified. |

| [CEM] ATE_IND.2 Work Units | Evaluation Activities |
|---|---|
| compose the test subset:<br><br>a) identification of the interface behaviour to be tested;<br><br>b) instructions to connect and setup all required test equipment as required to conduct the test;<br><br>c) instructions to establish all prerequisite test conditions;<br><br>d) instructions to stimulate the interface;<br><br>e) instructions for observing the interface;<br><br>f) descriptions of all expected results and the necessary analysis to be performed on the observed behaviour for comparison against expected results;<br><br>g) instructions to conclude the test and establish the necessary post-test state for the TOE;<br><br>h) actual test results. | |
| ATE_IND.2-10 The evaluator ***shall check*** that all actual test results are consistent with the expected test results. | The evaluator shall perform the [CEM] activity as specified. |
| ATE_IND.2-11 The evaluator ***shall report*** in the ETR[1] the evaluator testing effort, outlining the testing approach, configuration, depth and results. | The evaluator shall perform the [CEM] activity as specified. |

[1] Evaluation Technical Report

***Table 3: Mapping of ATE_IND.2 [CEM] Work Units to Evaluation Activities***

## 4.6 Class AVA: Vulnerability Assessment

**Vulnerability Analysis (AVA_VAN.2)**

While vulnerability analysis is inherently a subjective activity, a minimum level of analysis can be defined and some measure of objectivity and repeatability (or at least comparability) can be imposed on the vulnerability analysis process. In order to achieve such objectivity and repeatability it is important that the evaluator follows a set of well-defined activities, and documents the findings so others can follow these arguments and come to the same conclusions as the evaluator. While this does not guarantee that different evaluation facilities will identify exactly the same type of vulnerabilities or come to exactly the same conclusions, the approach defines the minimum level of analysis and the scope of that analysis, and provides CBs a measure of assurance that the minimum level of analysis is being performed by the evaluation facilities.

In order to meet these goals some refinement of the AVA_VAN.2 [CEM] work units is needed. The following table indicates, for each work unit in AVA_VAN.2, whether the [CEM] work unit is to be performed as written, or if it has been clarified by an Evaluation Activity. If clarification has been provided, a reference to this clarification is provided in the table.

| [CEM] AVA_VAN.2 Work Units | Evaluation Activities |
|---|---|
| AVA_VAN.2-1 The evaluator ***shall examine*** the TOE to determine that the test configuration is consistent with the configuration under evaluation as specified in the ST. | The evaluator shall perform the [CEM] activity as specified. |
| AVA_VAN.2-2 The evaluator ***shall examine*** the TOE to determine that it has been installed properly and is in a known state | The evaluator shall perform the [CEM] activity as specified. |
| AVA_VAN.2-3 The evaluator ***shall examine*** sources of information publicly available to identify potential vulnerabilities in the TOE. | Replace [CEM] work unit with activities outlined in Appendix A.2. |
| AVA_VAN.2-4 The evaluator ***shall conduct*** a search of the ST, guidance documentation, functional specification, TOE | The evaluator shall perform the [CEM] activity as specified. |

| [CEM] AVA_VAN.2 Work Units | Evaluation Activities |
|---|---|
| design and security architecture description evidence to identify possible potential vulnerabilities in the TOE. | |
| AVA_VAN.2-5 The evaluator ***shall record*** in the ETR[2] the identified potential vulnerabilities that are candidates for testing and applicable to the TOE in its operational environment. | Replace the [CEM] work unit with the analysis activities on the list of potential vulnerabilities in Appendix A.1 through A.6 and documentation as specified in Appendix A.7. |
| AVA_VAN.2-6 The evaluator ***shall devise*** penetration tests, based on the independent search for potential vulnerabilities. | Replace the [CEM] work unit with the activities specified in Appendix A.6. |
| AVA_VAN.2-7 The evaluator ***shall produce*** penetration test documentation for the tests based on the list of potential vulnerabilities in sufficient detail to enable the tests to be repeatable. The test documentation shall include:<br><br>a) identification of the potential vulnerability the TOE is being tested for;<br><br>b) instructions to connect and setup all required test equipment as required to conduct the penetration test;<br><br>c) instructions to establish all penetration test prerequisite initial conditions;<br><br>d) instructions to stimulate the TSF; | The [CEM] work unit is captured in Appendix A.7; there are no substantive differences. |

[2] Evaluation Technical Report

| [CEM] AVA_VAN.2 Work Units | Evaluation Activities |
|---|---|
| e) instructions for observing the behaviour of the TSF;<br><br>f) descriptions of all expected results and the necessary analysis to be performed on the observed behaviour for comparison against expected results;<br><br>g) instructions to conclude the test and establish the necessary post-test state for the TOE. | |
| AVA_VAN.2-8 The evaluator ***shall conduct*** penetration testing. | The evaluator shall perform the [CEM] activity as specified. See Appendix A.6 for guidance related to attack potential for confirmed flaws. |
| AVA_VAN.2-9 The evaluator ***shall record*** the actual results of the penetration tests. | The evaluator shall perform the [CEM] activity as specified. |
| AVA_VAN.2-10 The evaluator ***shall report*** in the ETR the evaluator penetration testing effort, outlining the testing approach, configuration, depth and results. | Replace the [CEM] work unit with the reporting called for in Appendix A.7. |
| AVA_VAN.2-11 The evaluator ***shall examine*** the results of all penetration testing to determine that the TOE, in its operational environment, is resistant to an attacker possessing a Basic attack potential. | This work unit is replaced by the activities defined in Appendix A.6 and A.7. |
| AVA_VAN.2-12 The evaluator ***shall report*** in the ETR all exploitable vulnerabilities and residual vulnerabilities, detailing for each:<br><br>a) its source (e.g. [CEM] activity being undertaken when it was conceived, known to the | Replace the [CEM] work unit with the reporting called for in Appendix A.7. |

| [CEM] AVA_VAN.2 Work Units | Evaluation Activities |
|---|---|
| evaluator, read in a publication);<br><br>b) the SFR(s) not met;<br><br>c) a description;<br><br>d) whether it is exploitable in its operational environment or not (i.e. exploitable or residual).<br><br>e) the amount of time, level of expertise, level of knowledge of the TOE, level of opportunity and the equipment required to perform the identified vulnerabilities, and the corresponding values using the tables 3 and 4 of Annex B.4. | |

***Table 4: Mapping of AVA_VAN.2 [CEM] Work Units to Evaluation Activities***

Because of the level of detail required for the evaluation activities, the bulk of the instructions are contained in Appendix A, while an “outline” of the evaluation activity is provided below.

The evaluator formulates flaw hypotheses in accordance with process defined in A.6. The evaluator documents the flaw hypotheses generated for the TOE in the report in accordance with the guidelines in Appendix A.7. The evaluator shall perform vulnerability analysis in accordance with Appendix A.6. The results of the analysis shall be documented in the report according to Appendix A.7.

# 5. References


[CC1] Common Criteria for Information Technology Security Evaluation, Part 1: Introduction and General Model
CCMB-2017-04-001, Version 3.1 Revision 5, April 2017

[CC2] Common Criteria for Information Technology Security Evaluation,
Part 2: Security Functional Components,
CCMB-2017-049-002, Version 3.1 Revision 5, April 2017

[CC3] Common Criteria for Information Technology Security Evaluation,
Part 3: Security Assurance Components,
CCMB-2017-04-003, Version 3.1 Revision 5, April 2017

[CEM] Common Methodology for Information Technology Security Evaluation,
Evaluation Methodology,
CCMB-2017-04-004, Version 3.1, Revision 5, April 2017

[CCADD] CC and CEM Addenda,
Exact Conformance, Selection-Based SFRs, Optional SFRs
CCMB-2017-05-XXX, Version 0.5, May 2017

[DBMScPP] collaborative Protection Profile for Database Management Systems,
Version 1.0, 12 June 2020

# Appendix A. Vulnerability Analysis

## A.1 Sources of vulnerability information

[CEM] Work Unit AVA_VAN.2-3 has been supplemented in this SD to provide a better-defined set of flaws to investigate and procedures to follow based on this particular technology. Terminology used is based on the flaw hypothesis methodology, where the evaluation team hypothesizes flaws and then either proves or disproves those flaws (a flaw is equivalent to a “potential vulnerability” as used in the [CEM]). Flaws are categorized into four “types” depending on how they are formulated:

1. A list of flaw hypotheses applicable to the technology described by the cPP derived from public sources as documented in Appendix A.2 – this fixed set has been agreed to by the iTC. Additionally, this will be supplemented with entries for a set of public sources that are directly applicable to the TOE or its identified components (Type 1 flaws, as defined by the process in Appendix A.2); this is to ensure that the evaluators include in their assessment applicable entries that have been discovered since the cPP was published;

2. A list of flaw hypotheses contained in this document that are derived from lessons learned specific to that technology and other iTC input (for example, potential flaws that might be derived from other open sources and vulnerability databases) as documented in Appendix A.3. At this time, the iTC has identified one Type 2 flaw (SQL Injection). Additional Type 2 flaws may be identified for subsequent versions of this cPP.

3. A list of flaw hypotheses derived from information available to the evaluators; this includes the baseline evidence provided by the developer and described in this SD (documentation associated with EAs, documentation described in Appendix A), as well as other information (public and/or based on evaluator experience) as documented in Appendix A.3; and

4. A list of flaw hypotheses that are generated through the use of iTC-defined tool types; their application is specified in Appendix A.5.

## A.2 Type 1 Hypotheses—Public-Vulnerability-based

The following list of public sources of vulnerability information was selected by the iTC:

a) Search Common Vulnerabilities and Exposures: https://cve.mitre.org/cve/

b) Search the National Vulnerability Database: https://nvd.nist.gov/

c) Search US-CERT: https://www.kb.cert.org/vuls/search/

d) Search CVE[3] Details: https://www.cvedetails.com/

e) Search Packet Storm: https://www.packetstormsecurity.org/

At minimum, the search terms should include software identifier (e.g. name) and version and will be used by the evaluators in formulating hypotheses during their analyses. The list of sources above was searched with the following search terms:

- Product name
- If specific platform libraries are included in the evaluated configuration (as specified in the administrator guidance) then the search terms should include those items and their specified version
- Keywords associated with the TOE

The evaluator will also consider the requirements that are chosen and the appropriate guidance that is tied to each requirement.

In order to supplement this list, the evaluators shall also perform a search on the sources listed above to determine a list of potential flaw hypotheses that are more recent than the publication date of the cPP, and those that are specific to the TOE and its components as specified by the additional documentation mentioned above. Any duplicates – either in a specific entry, or in the flaw hypothesis that is generated from an entry from the same or a different source – can be noted and removed from consideration by the evaluation team.

As part of type 1 flaw hypothesis generation for the specific components of the TOE, the evaluator shall also search the developer's websites to determine if flaw hypotheses can be generated. For instance, if security patches have been released for the version of the component being evaluated, the subject of those patches may form the basis for a flaw hypothesis.

## A.3 Type 2 Hypotheses—iTC-Sourced

### A.3.1 SQL Injection

SQL Injection is a security vulnerability that allows an attacker to manipulate queries. Typically, these queries are made by an application to a database; however, if the database creates SQL code dynamically, or includes a client that creates SQL code dynamically, then this vulnerability may exist within the DBMS TOE.

The result of such a query may allow an attacker to view data that would not normally be available to that user, may allow the user to infer information about the database structure or content, or may allow the attacker to modify or delete data.

[3] Common Vulnerabilities and Exposures

If the information presented for ADV_ARC.1-5 indicates that the DBMS dynamically creates queries from user input, the evaluator must test the effectiveness of the mitigation mechanisms. The evaluator must devise and execute at least one test case to demonstrate this function. It is recommended, but not required, that the test case be based on one of the attacks described by the Open Web Application Security Project (OWASP).

The evaluator must also devise a test for SQL vulnerabilities if the public vulnerability search results indicate that recent (within two years) versions of the TOE were susceptible to an SQL Injection attack. Additional client or environmental components that may be described in public vulnerabilities only need to be tested if they are part of the DBMS TOE, or the operational environment described in the ST.

If no relevant public vulnerabilities are found, and the evaluator determines that the DBMS does not dynamically create SQL queries (or any other query language code), then the evaluator will not be required to perform SQL Injection testing.

## A.4 Type 3 Hypotheses—Evaluation-Team-Generated

The iTC has leveraged the expertise of the developers and the evaluation labs to diligently develop the appropriate search terms and vulnerability databases. They have also thoughtfully considered the iTC-sourced hypotheses the evaluators should use based upon the applicable use case and the threats to be mitigated by the SFRs. Therefore, it is the intent of the iTC, for the evaluation to focus all effort on the Type 1 and Type 2 Hypotheses.

If the evaluators discover a Type 3 potential flaw that they believe should be considered, they should work with their CB to determine the feasibility of pursuing the hypothesis. The CB may determine whether the potential flaw hypotheses are worth submitting to the iTC for consideration as Type 2 hypotheses in future drafts of the cPP/SD.

## A.5 Type 4 Hypotheses—Tool-Generated

The evaluator will determine the open Transmission Control Protocol (TCP) and User Datagram Protocol (UDP) ports (e.g. by scanning of the DBMS) and verify that there are no unknown open ports. All open ports must be associated with expected services and protocols.

The evaluator will also choose a vulnerability scanning tool to scan for potential vulnerabilities. Although the iTC does not intend to restrict the list of tools that can be used, the tool must be able to provide up to date scanning, through updated signatures, or another mechanism.

## A.6 Process for Evaluator Vulnerability Analysis

As flaw hypotheses are generated from the activities described above, the evaluation team will disposition them; that is, attempt to prove, disprove, or determine the non-applicability of the hypotheses. This process is as follows:

The evaluator will refine each flaw hypothesis for the TOE and attempt to disprove it using the information provided by the developer or through penetration testing. During this process, the evaluator is free to interact directly with the developer to determine if the flaw exists, including requests to the developer for additional evidence (e.g., detailed design information, consultation with engineering staff); the CB may be included in these discussions.

### A.6.1 Unavailable evidence

In the case that the developer objects to the information being requested as being beyond that required by the evaluation activity/cPP and cannot provide other evidence that the flaw is disproved, the evaluator prepares an appropriate set of materials as follows:

- The documents used in formulating the hypothesis, and why it represents a potential compromise against a specific TOE function;
- An argument why the flaw hypothesis could neither be proven nor disproved by the evidence provided so far; and
- The types of information required to investigate the flaw hypothesis further.

The CB will then either approve or disapprove the request for additional information. If approved, the developer provides the requested evidence to disprove the flaw hypothesis (or, of course, acknowledge the flaw).

If the CB disapproves the request for additional information, the evaluator will follow AVA_VAN.2.4E and devise suitable penetration tests to enable the flaw to be disproved or classified as a residual vulnerability.

### A.6.2 Dealing with flaws

If the evaluator finds a flaw, the evaluator must report these flaws to the developer. All reported flaws must be addressed as follows:

a) If the developer confirms that the flaw exists and that it is exploitable at Basic Attack Potential, then a change is made by the developer, and the resulting resolution is agreed by the evaluator.

b) If the developer, the evaluator, and the CB agree that the flaw is exploitable only above Basic Attack Potential and does not require resolution for any other reason, and no change is made, then the flaw is noted as a residual vulnerability in the proprietary ETR.

c) If the developer and evaluator agree that the flaw is exploitable only above Basic Attack Potential, but it is deemed critical to fix because of technology-specific or cPP-specific aspects such as typical use cases or operational environments, then a change is made by the developer, and the resulting resolution is agreed by the evaluator.

Disagreements between the evaluator and the developer regarding questions of the existence of a flaw, its attack potential, or whether it should be deemed critical to fix are resolved by the CB.

Any testing performed by the evaluator and the results of the analysis are documented as outlined in Appendix A.7 below.

As indicated in Appendix A.7, the public statement with respect to vulnerability analysis that is performed on TOEs conformant to the cPP is constrained to coverage of flaws associated with Types 1 and 2 (defined in Appendix A.1) flaw hypotheses only. The fact that the iTC generates these candidate hypotheses indicates that these must be addressed.

## A.7 Reporting

The evaluators shall produce a report on the vulnerability assessment that is delivered to the overseeing CB. This may form part of the ETR, or may be in another format if so required by the CB.

This report must contain:

- The flaw identifiers returned when the procedures for searching public sources were followed according to instructions in the SD per Appendix A.2 (cf. AVA_VAN.2-4);
- A statement that the evaluators have examined the Type 1 flaw hypotheses specified in this SD in Appendix A.2 (i.e. the flaws listed in the previous bullet) and the Type 2 flaw hypotheses specified in this SD by the iTC in Appendix A.3;
- A list of all of the flaw hypotheses generated (cf. AVA_VAN.2-4);
- The evaluator penetration testing effort, outlining the testing approach, configuration, depth and results (cf. AVA_VAN.2-10);
- All documentation used to generate the flaw hypotheses (in identifying the documentation used in coming up with the flaw hypotheses, the evaluation team must characterize the documentation so that a reader can determine whether it is strictly required by this SD, and the nature of the documentation (design information, developer engineering notebooks, etc.));
- How each flaw hypothesis was resolved (this includes whether the original flaw hypothesis was confirmed or disproved, and any analysis relating to whether a residual vulnerability is exploitable by an attacker with Basic Attack Potential) (cf. AVA_VAN.2-11);
- The evaluator shall report all exploitable vulnerabilities and residual vulnerabilities, detailing for each:

- Its source (e.g. [CEM] activity being undertaken when it was conceived, known to the evaluator, read in a publication);
- The SFR(s) not met;
- A description;
- Whether it is exploitable in its operational environment or not (i.e. exploitable or residual).
- The amount of time, level of expertise, level of knowledge of the TOE, level of opportunity and the equipment required to perform the identified vulnerabilities (cf. AVA_VAN.2-12);

- In the case that actual testing was performed in the investigation (either as part of flaw hypothesis generation using tools specified by the iTC in Appendix A.5 or in proving/disproving a particular flaw) the steps followed in setting up the TOE (and any required test equipment); executing the test; post-test procedures; and the actual results (to a level of detail that allow repetition of the test, including the following:
  - Identification of the potential vulnerability the TOE is being tested for;
  - Instructions to connect and setup all required test equipment as required to conduct the penetration test;
  - Instructions to establish all penetration test pre-requisite initial conditions;
  - Instructions to stimulate the TSF;
  - Instructions for observing the behaviour of the TSF;
  - Descriptions of all expected results and the necessary analysis to be performed on the observed behaviour for comparison against expected results;
  - Instructions to conclude the test and establish the necessary post-test state for the TOE. (cf. AVA_VAN.2-7).

# Appendix B. Glossary

The terms, definitions and abbreviations given in [CC1] and [CEM] apply to this document. Additional terms, definitions and abbreviations applicable are found in the DBMS cPP. In addition, the following are used in this document:

## B.1 Terms and Definitions

| Term | Meaning |
|---|---|
| Administrator | The term 'Administrator' refers to a user who has been specifically granted the authority to manage some portion or the entire TOE and whose actions may affect the DAC. Administrators may possess special privileges that provide capabilities to override portions of the access control policy. |
| Application | An executable program. |
| Database Management System (DBMS) | A suite of programs that typically manage large structured sets of persistent data, offering ad hoc query facilities to many users. They are widely used in business applications. |
| Discretionary Access Control (DAC) | A means of restricting access to objects based on the identity of subjects and/or groups to which they belong. Those controls are discretionary in the sense that a subject with certain access permission is capable of passing that permission (perhaps indirectly) on to any other subject. |

## B.2 Acronyms used in this SD

| Acronym | Meaning |
|---|---|
| **CB** | Certification Body |
| **CC** | Common Criteria |
| **CCDB** | Common Criteria Development Board |
| **CCRA** | Common Criteria Recognition Arrangement |
| **CEM** | Common Evaluation Methodology |
| **cPP** | collaborative Protection Profile |
| **CVE** | Common Vulnerabilities and Exposures |
| **DBMS** | Database Management System |
| **EA** | Evaluation Activities |
| **ETR** | Evaluation Technical Report |
| **iTC** | International Technical Community |
| **OWASP** | Open Web Application Security Project |
| **SAR** | Security Assurance Requirement |
| **SD** | Supporting Document |
| **SFR** | Security Functional Requirement |
| **SQL** | Structured Query Language |

| | |
|---|---|
| **ST** | Security Target |
| **TCP** | Transmission Control Protocol |
| **TOE** | Target of Evaluation |
| **TSF** | TOE Security Functionality |
| **TSS** | TOE Summary Specification |
| **UDP** | User Datagram Protocol |