DbProtect AppRadar 2009.1 R2

Security Target

Version 1.0
05/21/2012

Prepared for:
Application Security, Inc.

350 Madison Avenue, 6th Floor
New York, NY 10017

Prepared By:
Science Applications International Corporation

Common Criteria Testing Laboratory

6841 Benjamin Franklin Drive
Columbia, MD 21046

LIST OF TABLES

1. Security Target Introduction

This section identifies the Security Target (ST) and Target of Evaluation (TOE) identification, ST conventions, ST conformance claims, and the ST organization. The TOE is DbProtect AppRadar 2009.1 R2 hereafter called AppRadar, provided by Application Security, Inc. The TOE provides real-time database intrusion detection and security auditing. AppRadar provides database-specific, monitoring, and auditing of the popular commercially available database servers.

The Security Target contains the following additional sections:

- Section 2 – Target of Evaluation (TOE) Description
  This section gives an overview of the TOE, describes the TOE in terms of its physical and logical boundaries, and states the scope of the TOE.
- Section 3 – TOE Security Environment
  This section details the expectations of the environment and the threats that are countered by the TOE and operational environment.
- Section 4 – TOE Security Objectives
  This section details the security objectives of the TOE and operational environment.
- Section 5 – IT Security Requirements
  The section presents the security functional requirements (SFR) for the TOE and details the assurance requirements.
- Section 6 – TOE Summary Specification
  The section describes the security functions represented in the TOE that satisfy the security requirements.
- Section 7 – Protection Profile Claims
  This section presents any protection profile claims.
- Section 8 – Rationale
  This section closes the ST with the justifications of the security objectives, requirements and TOE summary specifications as to their consistency, completeness, and suitability.

1.1 Security Target, TOE and CC Identification

ST Title – DbProtect AppRadar 2009.1 R2 Security Target

ST Version – Version 1.0

ST Date – 21 May 2012

TOE Identification – DbProtect AppRadar 2009.1 R2

TOE Developer – Application Security, Inc.

Evaluation Sponsor – Application Security, Inc.

CC Identification – Common Criteria for Information Technology Security Evaluation, Version 3.1, Revision 3, July 2009

1.2 Conformance Claims

This TOE is conformant to the following CC specifications:

- Common Criteria for Information Technology Security Evaluation Part 2: Security Functional Requirements, Version 3.1, Revision 3, July 2009.
  - Part 2 Extended
- Common Criteria for Information Technology Security Evaluation Part 3: Security Assurance Requirements, Version 3.1, Revision 3, July 2009.
  - Part 3 Conformant

- Assurance Level Package:
    - EAL 2 augmented with ALC_FLR.2

1.3 Conventions

The following conventions have been applied in this document:

- Security Functional Requirements – Part 2 of the CC defines the approved set of operations that may be applied to functional requirements: iteration, assignment, selection, and refinement.
    - Iteration: allows a component to be used more than once with varying operations. In the ST, iteration is indicated by a letter placed at the end of the component. For example FDP_ACC.1a and FDP_ACC.1b indicate that the ST includes two iterations of the FDP_ACC.1 requirement, a and b.
    - Assignment: allows the specification of an identified parameter. Assignments are indicated using bold and are surrounded by brackets (e.g., [**assignment**]). Note that an assignment within a selection would be identified in italics and with embedded bold brackets (e.g., [***[selected-assignment]***]).
    - Selection: allows the specification of one or more elements from a list. Selections are indicated using bold italics and are surrounded by brackets (e.g., [***selection***]).
    - Refinement: allows the addition of details. Refinements are indicated using bold, for additions, and strike-through, for deletions (e.g., "… **all** objects …" or "… ~~some~~ **big** things …").
- Other sections of the ST – Other sections of the ST use bolding to highlight text of special interest, such as captions.
- Requirements that are Part 2 extended, i.e., are not included in the CC Part 2 but are explicitly defined by the ST author, are marked with an (EXP) after the requirement name.

2. TOE Description

The Target of Evaluation (TOE) is DbProtect AppRadar 2009.1 R2, hereafter called AppRadar.

AppRadar is an application that provides real-time database intrusion detection and security auditing. AppRadar provides database-specific, monitoring, and auditing of the commercially available database servers (see section 2.2 for the applicable databases and versions). The key features of the TOE are database event monitoring and auditing.

2.1 TOE Overview

The TOE is a software application that runs in the context of an operating system. AppRadar includes two system components: 1) a console, which serves as a data collector and provides a web-based front end and 2) a number of sensors that monitor databases on a host or on the network and send collected data back to the console. An example deployment of the TOE is depicted in figure 1, below.

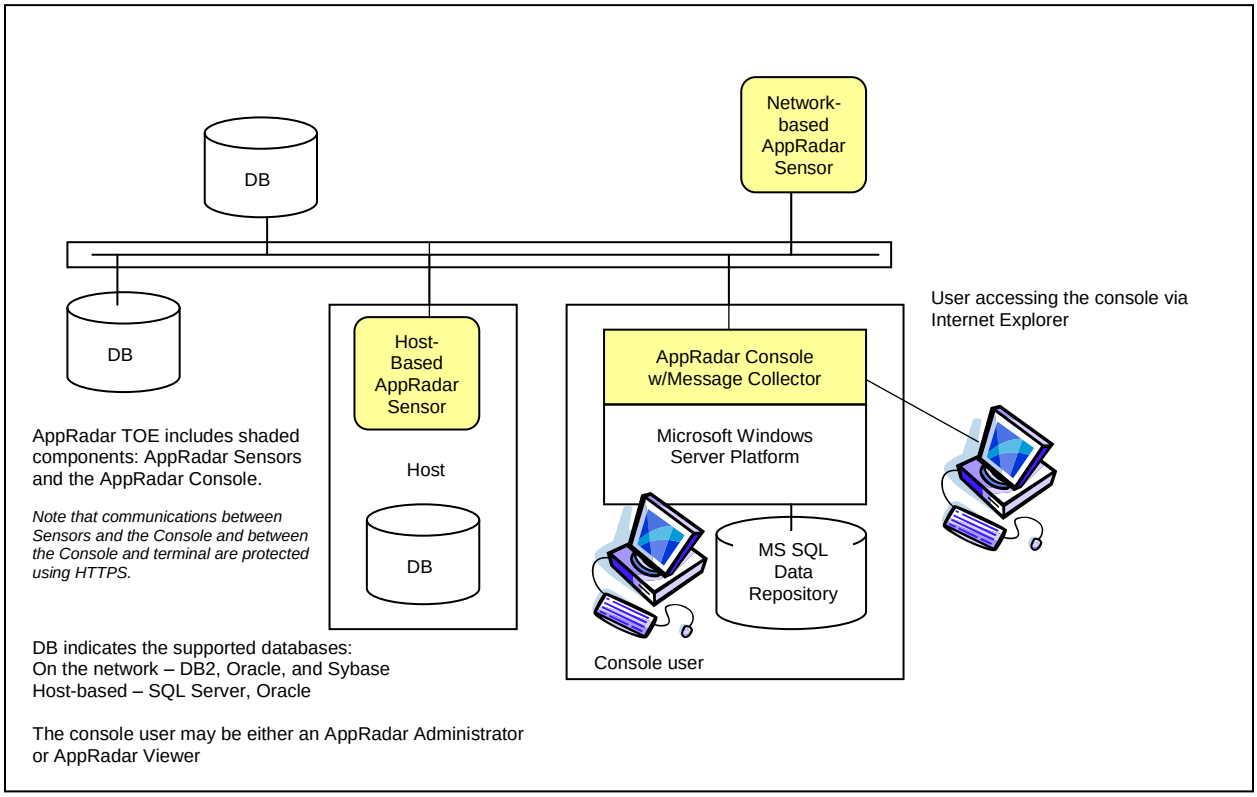

Figure 1. Diagram of TOE and TOE Environment

The AppRadar Console runs on a Microsoft Windows (see section 2.2 for specific versions) and it is accessible via Microsoft Internet Explorer.
The AppRadar Sensors run on Microsoft Windows, Sun Solaris or Red Hat Enterprise Linux. For specific information on the required operational environment, see section 2.2.

The AppRadar Console provides capabilities to initialize and manage the TOE and to view alerts and audit data collected by the Sensors. All communication between the Console and Sensors uses HyperText Transfer Protocol Secure (HTTPS) or SOAP over HTTPS.

The AppRadar Console allows users to:

- Initialize and configure sensors

- Load database identification and authentication information (i.e., login and password) required for querying Sensors during the initialization of the sensors
- View alerts on the AppRadar Console and in a Log File, an SNMP Trap Receiver, or through configurable email forwarding
- Initialize and configure policies
- Initialize and configure filters
- View reports

When a user connects to the AppRadar Console via their browser (using SSL), they are prompted for a username and password. The AppRadar Console uses the username and password to attempt to log the user into the host Microsoft Windows operating system (OS). All user definitions and authentication credentials are controlled by the OS in the operational environment. The AppRadar Console is accessed by two types of users: AppRadar Administrators and AppRadar Viewers. However, those roles are not defined by AppRadar; but rather are distinct, product-specific Windows groups that must be administered (e.g., user assigned) by the Windows administrator as part of the operational environment. Using Windows Groups (determined in the processing of logging the user into the Windows host platform), the TOE permits AppRadar Administrators to perform all AppRadar functions, including configuring the system, modifying existing policies and filters and creating new policies and filters, and assigning additional destinations for alerts in addition to the Console. AppRadar Viewers are allowed to only view the AppRadar configuration, policies, filters, and reports. AppRadar Viewers may not initialize any TOE components or configure any policies or filters. AppRadar Viewers have no access to the monitored databases' identification and authentication information, which is entered by the AppRadar Administrator at sensor install and is maintained by the administrator as required by the specific database. Note that users that are not assigned to either the AppRadar Administrator or AppRadar Viewer role will be denied access during the log in process. AppRadar Console is accessible to users via Internet Explorer.

AppRadar Sensors monitor database activity; there is a sensor associated with each database monitored. There are two types of AppRadar sensors:

- Host-based sensors monitor Microsoft SQL Server or Oracle databases. Host-based sensors are located on the same machine as the monitored database. The Sensor captures SQL commands and reports activity back to the AppRadar Console, which stores information in its backend database.
- Network-based sensors monitor Oracle, DB2, or Sybase ASE databases on the network. The network-based sensors may be located anywhere on the network where database traffic is flowing to and from the monitored database; the network-based sensor is similar to a sniffer. The database traffic is analyzed and information on the activity is reported back to the AppRadar Console, which stores the information in its backend database.

AppRadar Sensors monitor for a variety of events such as intrusion attempts or auditing of normal usage as defined by TOE policies and filters. Audit records and Alerts are created by the sensors based on database events.

An alert is a notification of a monitored event detected on the database host or network and an audit is a record of standard database activity. AppRadar Sensors generates alerts for activities defined as security events by the TOE policies. The alerts and audit records are sent via a network connection to the AppRadar Console (actually its Message Collector component) and are stored in the Microsoft SQL[1] Data Repository (i.e., backend database), which is outside the TOE boundary.

[1] Note that the AppRadar Console can be configured to utilize a Microsoft SQL server that is running either on the same host server or on another server that is continuously accessible via a network connection. Note also that the TOE can be configured to either use Windows authentication for database access or alternately to use database authentication. In the latter case, the TOE stored the applicable database credentials using Windows Data protection API (DPAPI) to protect them and recall them when needed.

2.2 TOE Architecture

The TOE architecture is depicted in Figure 1 above. The TOE consists of an AppRadar Console, which includes the AppRadar Console service and Message Collector software applications, and a number of AppRadar Sensors, which are also software applications.

The TOE is distributed with the 3rd partyTomcat Engine 5.5.20 to facilitate the web-based management interface. However, Tomcat is considered a distinct component in the operational environment.

The following lists the TOE's requirements for the supporting operational environment:

Databases monitored:

AppRadar monitors the following databases and versions:

AppRadar Sensor Type	Database Platform	Versions
Network-based Sensor	Oracle	Oracle 7.x, Oracle 8, 8i, 9i, 9iR2, 10g, 10gR2
	Sybase	Sybase ASE 11.x through 15
	IBM DB2	IBM DB2 UDB version 8 and 9 IBM DB2 for zSeries v7 and v8
Host-based Sensor	Microsoft SQL Server	Microsoft SQL Server 2008 (all editions) Microsoft SQL Server 2005 (all editions) Microsoft SQL Server 2000 (all editions)
	Oracle	Oracle 9iR2, 10g, and 10gR2
	IBM DB2	IBM DB2 UDB version 8 and 9

AppRadar Console requirements:

- Operating system: Microsoft Windows Server 2003 or 2008 Enterprise Edition, Microsoft Windows Server 2003 or 2008 Enterprise x64 each with the latest patches.

  Note that the Windows Data Protection API (DPAPI) is required to encrypt backend database credentials if Windows authentication is not used.

- Browser: Internet Explorer 7.0 or higher with JavaScript enabled. Sun Microsystems Java Runtime Environment (JRE) 1.6 is required for DbProtect Console applet to load into the web browser. Refer to the DbProtect User Guide for troubleshooting JRE security settings on Internet Explorer.

  Note the browser requirement applies to any host from which the AppRadar Console might be accessed, including the local host.

- Networking: Network connectivity is required for the AppRadar Console to communicate with AppRadar Sensors. Also, OpenSSL is required to encrypt that communication using SSL.

  Note that OpenSSL is also used to encrypt (and thereby protect) database credentials used by the host-based sensors.

  Note that Tomcat 5.5.20, distributed with the TOE, is required to enable the web-based management front end of the TOE.

- Backend Database: Microsoft SQL Server 2000 SP4, Microsoft SQL Server 2005, Microsoft SQL Server 2008, used to store data collected by the sensors and consolidated by the Console (MSDE 2000 SP3, which is bundled with AppRadar, is not included in the evaluated configuration)..

- Optional Services:

- AppRadar Console can optionally be configured to email alerts using a configured SMTP server.
- AppRadar Console can optionally be configured to send SNMP traps to a configured SNMP server (i.e., trap receiver).

AppRadar Host-based Sensor requirements:

- Operating system:
  - For Microsoft SQL Server:
    - Microsoft Windows Server 2003 or 2008 Enterprise Edition, Microsoft Windows Server 2003 or 2008 Enterprise x64,
  - For Oracle
    - Sun Solaris 8, 9, 10 (32 and 64 bit SPARC)
    - Red Hat Enterprise Linux 3, 4 and 5 (32 bit x86 and 64 bit x64)
    - AIX 5.2 Technology Level 5 and greater
    - HP-UX 11i v1 or later on the PA-RISC processor and HPUX 11i v2 or later on the Itanium (IA64) processor
    - Windows Server 2003 (including Enterprise Edition), 32-bit
  - For DB2
    - Red Hat Enterprise Linux 3, 4, or 5 (32-bit x86 and 64-bit x64)
    - Solaris 8, 9, and 10 (64-bit SPARC)
    - AIX 5.2 Technology Level 5 and greater (32-bit and 64-bit)
    - Windows Server 2003 or 2008 (including Enterprise Edition), 32-bit
- Networking: Network connection to the AppRadar Console. Also, OpenSSL is required to encrypt that communication using SSL.

AppRadar Network-based Sensor requirements:

- Operating system: Microsoft Windows Server 2003 or 2008 Enterprise Edition, Microsoft Windows Server 2003 or 2008 Enterprise x64.
- Networking: Network connection to the AppRadar Console. Also, OpenSSL is required to encrypt that communication using SSL.

2.2.1 Physical Boundaries

The physical boundaries of the TOE are depicted in Figure 1, above, and include the AppRadar Console application (a software application) one or more AppRadar sensor applications (software applications).

The DbProtect console component provides the ability to manage both the AppRadar product and the sibling AppDetective product, which was separately evaluated. While this TOE doesn't address the integration of both products, the use of the AppDetective product with the DbProtect console in addition to the AppRadar product does not invalidate the evaluated configuration of the TOE.

2.2.2 Logical Boundaries

This section identifies: the security functions provided by DbProtect AppRadar and functions provided by the operational environment in which DbProtect AppRadar operates.

2.2.2.1 Evaluated Security Functions

This section summarizes the security functions provided by the TOE:

- Security audit

- Security management
- Database data collection and monitoring

2.2.2.1.1 Security audit

The TOE generates audit records for administrator operations, including attempts to access the TOE and its data and any configuration changes made to the TOE. The audit records are stored in the backend database and can be queried by the TOE to facilitate review of those records..

2.2.2.1.2 Security management

The TOE provides security management functions to allow installation of TOE sensor components, modifications to TOE policies and filters, and loading of database login IDs and Passwords. The functions are accessible via an SSL-enabled web-browser. Each user is required to be identified an authenticated, using services of the host operating system (OS), and must be in one of the two pre-defined OS groups associated with the TOE in order to get access to the corresponding functions. Once a user is identified, authenticated and found to be associated with only one of the applicable groups, the corresponding functions are presented to the user so they can be used.

2.2.2.1.3 Database data collection and monitoring

The TOE monitors database functions based on policies and filters defined in the TOE by the AppRadar Administrator. Both normal database usage and security events are monitored and records are generated. Security events, as defined by TOE policies, cause the TOE to generate an alert that is sent to the Console, or optionally to an SNMP and/or SMTP server configured in the operational environment.

In general, the database monitoring is performed by AppRadar Sensor components that are configured to monitor associated databases in accordance with their individual configurations. The AppRadar Sensor components are centrally managed via the AppRadar Console component and report the results of monitoring to the AppRadar Console so that they can be centrally accessed.

Note that AppRadar includes both host-based and network-based sensors. Host-based sensors reside on the same host as the database they monitor while network-based sensors reside on the same network as the database they monitor. Host-based sensors use database credentials in order to monitor database activities while network-based sensors monitor network traffic to discern database activities. Both types of sensors perform similar monitoring functions, though given the differences in mechanics there are some differences in their functions.

- Network-based Sensors fire Alerts for all remote connections, e.g., from a web server communicating to its remote back-end database. However, they do not detect activity originating from the database host.
- Host-based Sensors detect both local and remote activity. However, they detect only successfully executed commands.

The TOE relies on SSL in the operational environment to protect stored host-based database credentials. The TOE also relies on the Windows Data Protection API (DPAPI) to protect credentials for its backend database when Windows authentication is not being used.

2.2.2.2 Functions Provided by the TOE Operational Environment

The TOE relies on the operational environment in which it operates for the following security and other functionality:

- Protect the TOE's stored executable image and its execution environment;
- Protect TOE stored data, including audit records;
- Provide a means to audit attempts to access the TOE stored executable image and stored data from the operational environment (i.e., not through the TOE's own interfaces);
- Provide a reliable time stamp for use in audit records and scan results;
- Identify and authenticate authorized users; and,

- Provide encryption services used to encrypt database credentials and also to encrypt communication channels between the TOE components and also between the TOE Console and web browsers used to access it.

Additionally, the TOE relies on its host to facilitate communication with target database applications and operating system products for the purposes of scanning and auditing. The TOE uses the ODBC, Oracle Instant client, DB2 client, Lotus Notes Domino C++, or TCP/IP socket APIs.

2.2.2.3 Functions not Addressed by the Evaluation

The product provides a tool, ASAP Updater, which can be used to update the TOE and its knowledge base of application problems. However, the developer's deployment methodology is to make only complete releases of the TOE software available to customers. Use of ASAP Updater would take the TOE out of its evaluated configuration, and so it is excluded from the evaluation.

Similarly, the product includes a Configuration Manager tool. The Configuration Manager tool provides a means for modifying various configuration parameters on the Console's host machine. This tool is not necessary for the normal use of the TOE and has been excluded from the evaluated configuration as a result.

Additionally, the following capabilities have been excluded from the scope of analysis during the evaluation: use of the report customization (e.g., to exclude specific data) capabilities available in the product.

2.3 TOE Documentation

Application Security, Inc. offers a series of documents that describe the installation of AppRadar as well as guidance for subsequent use and administration of the applicable security features as follows:

- DbProtect AppRadar 2009.1 R2 Evaluated Configuration, Version 0.9, April 6, 2012
- DbProtect 2009.1R2 Administrators' Guide, April 17, 2009 (updated April 6, 2012)
- DbProtect 2009.1R2 Installation Guide, April 7, 2009
- DbProtect 2009.1R2 Users' Guide, April 21, 2009 (updated April 6, 2012)

3. Security Environment

The TOE security environment describes the security aspects of the intended environment in which the TOE is to be used and the manner in which it is expected to be employed. The statement of the TOE security environment defines the following:

- Threats that the TOE and the environment of the TOE counters
- Assumptions made about the operational environment and the intended method of use for the TOE

Furthermore, the TOE is intended to be used in environments where the relative assurance that its security functions are enforced is commensurate with EAL2 as defined in the CC.

3.1 Threats

T.COMM	Data transmitted between TOE components may be captured and read by an attacker, thereby compromising sensitive TOE data.
T.IMPCON	An unauthorized user may inappropriately change the configuration of the TOE causing potential intrusions to go undetected.
T.INADVE	Inadvertent activity and access by unauthorized or authorized users may be undetected on a database the TOE monitors thereby compromising the data maintained in the database.
T.MISACT	Malicious activity by attackers may be undetected on a database the TOE monitors, thereby compromising the data maintained in the database.
T.MISUSE	Unauthorized accesses and activity indicative of misuse by unauthorized or authorized users may be undetected on a database the TOE monitors, thereby compromising the data maintained in the database.
T.PRIVIL	An unauthorized user may gain undetected access to the TOE and exploit system privileges to gain access to TOE security functions and data.
T.PROTECT	An attacker may access and modify TOE data assets and configuration data, causing the TOE functions to work incorrectly and causing loss of TOE data and database monitoring functions.

3.2 Assumptions

A.ACCESS	The TOE has access to all the IT System data it needs to perform its functions.
A.DYNMIC	The TOE will be managed in a manner that allows it to appropriately address changes in the IT System the TOE monitors.
A.LOCATE	The processing resources of the TOE, i.e., the sensors and console will be located within controlled access facilities, which will prevent unauthorized physical access. Remote access to the console component of the TOE is possible outside the controlled access facilities.
A.MANAGE	There will be one or more competent individuals assigned to manage the TOE and the security of the information it contains.
A.NOEVIL	The authorized administrators are not careless, willfully negligent, or hostile, and will follow and abide by the instructions provided by the TOE documentation.
A.NOTRST	The TOE can only be accessed by authorized users.

A.PROTCT	The TOE software critical to security policy enforcement will be protected from unauthorized physical modification.

4. Security Objectives

This section identifies the security objectives of the TOE and its supporting operational environment. The security objectives are intended to counter identified threats and address applicable assumptions.

4.1 Security Objectives for the TOE

O.AUDIT	The TOE must record audit records for use of the TOE functions.
O.AUDIT_REVIEW	The TOE must provide a means for TOE users to review all audit and database monitoring data.
O.MANAGE	The TOE must allow authorized users to manage appropriate TOE functions.
O.MONITOR	The TOE must monitor databases for defined attacks and normal activity and generate alerts for defined attacks and audit records for normal activity that can be viewed by TOE users.

4.2 Security Objectives for the TOE Operational Environment

The following security objectives for the operational environment of the TOE must be satisfied in order for the TOE to fulfill its own security objectives.

OE.CREDEN	Those responsible for the TOE must ensure that all access credentials are protected by the users in a manner which is consistent with IT security.
OE.IDENT_AUTHEN	The operational environment must be able to identify and authenticate users prior to allowing access to TOE functions and data.
OE.INSTAL	Those responsible for the TOE must ensure that the TOE is delivered, installed, managed, and operated in a manner which is consistent with IT security.
OE.INTROP	The TOE is interoperable with the IT System it monitors.
OE.PERSON	Personnel working as authorized administrators shall be carefully selected and trained for proper operation of the System.
OE.PHYCAL	Those responsible for the TOE must ensure that those parts of the TOE critical to security policy are protected from any physical attack.
OE.PROTCT	The operational environment must protect the TOE and databases in the operational environment from unauthorized modifications and access to its functions and data including protection of communication between TOE components from unauthorized modification or disclosure.
OE.ROLES	The operational environment must be associate users with roles as defined by the TOE.
OE.TIME	The operational environment must protect provide a reliable time source.

5. IT Security Requirements

The security requirements for the TOE have been drawn from Parts 2 and 3 of the Common Criteria, with the exception of some extended security functional requirements crafted to better represent the monitoring functions of the TOE. The security functional requirements have been selected to correspond to the actual security functions implemented by the TOE while the assurance requirements have been selected to offer a low to moderate level of assurance that those security functions are properly realized.

5.1 Extended Security Functional Requirements

A class of Database Collection and Monitoring (IDD) requirements was created to specifically address the database monitoring and data collected by the TOE. The database monitoring function of the TOE is not specified within the Common Criteria; therefore the requirements for this function had to be explicitly defined. The audit class of FAU was used as a model for creating these requirements.

The purpose of this class of requirements is to address the unique nature of the database monitoring function and provide for requirements about collecting and reviewing the data. These requirements embody all the necessary security functions. IDD_DDC_EX.1 has been created to address database data collection. IDD_DDR_EX.1 has been created to address applicable reactions based on collected data. IDD _PRT_EX.1 has been created explicitly to ensure that the security credentials used to access target IT systems must be protected; otherwise the targets might be more vulnerable due to the TOE. It has been added to the IDD class since it is directly related to the main database monitoring functions of the TOE. Each is defined as follows:

5.1.1 Database data collection (IDD_DDC)

Family Behavior

The family of SFRs is intended to address functions related to collecting pertinent information from monitored databases.

Management: IDD_DDC_EX.1

The following actions could be considered for the management functions in FMT:

a) Management of the collection function.

Audit: IDD_DDC_EX.1

The following actions should be auditable if IDD_DDC_EX.1 is included in the PP/ST:

- Minimal: Configuration or use of the collection function.

5.1.1.1 Database data collection (IDD_DDC_EX.1)

Hierarchical to: No other components

Dependencies: None

IDD_DDC_EX.1.1 The TSF shall be able to collect the following information from the monitored database:

a) Identification and authentication events to the database, both successful and unsuccessful.
b) Predefined database attacks including: Buffer overflows; Predefined threats based on database configuration settings; Password attacks; Privilege escalation; Database scans using security tools: Web Application attacks.

IDD_DDC_EX.1.2 The TSF shall collect and record the following information: Date and time of the event, type of event, subject identity if known, application that triggered the event if available, client host IPv4 address if known.

5.1.2 Database data react (IDD_DDR)

Family Behavior

The family of SFRs is intended to address functions related to reacting (e.g., alerts) based on pertinent findings based on collected or monitored data.

Management: IDD_DDR_EX.1

The following actions could be considered for the management functions in FMT:

a) Management of the reaction function.

Audit: IDD_DDR_EX.1

The following actions should be auditable if IDD_DDR_EX.1 is included in the PP/ST:

- Minimal: Configuration of the reaction function.

5.1.2.1 Database data react (IDD_DDR_EX.1)

Hierarchical to: No other components

Dependencies: IDD_DDC_EX.1

IDD_DDR_EX.1.1 The TSF shall send an alarm to the AppRadar console and to other destinations including log files, SNMP, and/or SMTP if defined by the AppRadar Administrator when a monitored security event is detected.

5.1.3 DB Credential Protection (IDD_PRT)

Family Behavior

The family of SFRs is intended to address functions related to storing and protecting credentials used by the TOE to access scan targets in the operational environment.

Management: IDD_PRT_EX.1

The following actions could be considered for the management functions in FMT:

a) Management of the security credentials.

Audit: IDD_PRT_EX.1

There are no auditable events foreseen.

5.1.3.1 DB Credential Protection (EXP) (IDD_PRT_EX.1)

Hierarchical to: No other components

Dependencies: None

IDD_PRT_EX.1.1 The TSF shall ensure that configured IT System security credentials are stored in a secure manner.

5.2 TOE Security Functional Requirements

The following table describes the SFRs satisfied by the TOE. Extended requirements are marked with (EXP) to indicate that they are not drawn from the Common Criteria.

Requirement Class	Requirement Component
FAU: Security audit	FAU_GEN.1: Audit data generation
	FAU_SAR.1: Audit review
FMT: Security management	FMT_MTD.1: Management of TSF data
	FMT_SMF.1: Specification of Management Functions
IDD: Database data collection and	IDD_DDC_EX.1: Database data collection

monitoring (EXP)	IDD_DDR_EX.1: Database data react
	IDD_PRT_EX.1: DB Credential Protection (EXP)

Table 1 TOE Security Functional Components

5.2.1 Security Audit (FAU)

5.2.1.1 Audit Data Generation (FAU_GEN.1)

FAU_GEN.1.1 The TSF shall be able to generate an audit record of the following auditable events:

a) Start-up and shutdown of the audit functions;

b) All auditable events for the [***not specified***] level of audit; and

c) [**Audit events defined in the table below**].

Security Functional Requirement	Audit event
FAU_SAR.1	Access to audit records through the AppRadar Console
FMT_MTD.1	All modifications to the values of TSF data: TOE configuration changes including initialize and modify database login ID and password used by the TOE for access to monitored databases; modify existing and create new filters and policies; add additional destinations for alerts
FMT_SMF.1	Use of management functions

Table 2 Audit Events

FAU_GEN.1.2 The TSF shall record within each audit record at least the following information:

a) Date and time of the event, type of event, subject identity (if applicable), and the outcome (success or failure) of the event; and

b) For each audit event type, based on the auditable event definitions of the functional components included in the PP/ST, [**no additional information]**:

5.2.1.2 Audit Review (FAU_SAR.1)

FAU_SAR.1.1 The TSF shall provide **[AppRadar Administrator, AppRadar Viewer]** with the capability to read **[all information]** from the audit records.

FAU_SAR.1.2 The TSF shall provide the audit records in a manner suitable for the user to interpret the information.

Application Note: Audit records refer to all records generated by the TOE including database monitoring events and alerts defined in IDD_DDC_EX.1 and IDD_DDR_EX.1 and TOE audit events defined in FAU_GEN.1.

5.2.2 Security management (FMT)

5.2.2.1 Management of TSF data (FMT_MTD.1)

FMT_MTD.1.1 The TSF shall restrict the ability to [***change_default, modify*** **[initialize]**] the [**database login ID and password used by the TOE for access to monitored databases, policies, filters, destination of alerts**] to [**AppRadar Administrator**].

5.2.2.2 Specification of Management Functions (FMT_SMF.1)

FMT_SMF.1.1 The TSF shall be capable of performing the following security management functions: [**initialize and modify database login ID and password used by the TOE for access to monitored**

databases; modify existing and initialize new filters and policies; initialize additional destinations for alerts].

5.2.3 Database data collection and monitoring (EXP) (IDD)

5.2.3.1 Database data collection (IDD_DDC_EX.1)

IDD_DDC_EX.1.1 The TSF shall be able to collect the following information from the monitored database:

a) Identification and authentication events to the database, both successful and unsuccessful.

b) Predefined database attacks including: Buffer overflows; Predefined threats based on database configuration settings ; Password attacks; Privilege escalation; Database scans using security tools: Web Application attacks.

IDD_DDC_EX.1.2 The TSF shall collect and record the following information: Date and time of the event, type of event, subject identity if known, application that triggered the event if available, client host IPv4 address if known.

5.2.3.2 Database data react (IDD_DDR_EX.1)

IDD_DDR_EX.1.1 The TSF shall send an alarm to the AppRadar console and to other destinations including log files, SNMP, and/or SMTP if defined by the AppRadar Administrator when a monitored security event is detected.

5.2.3.3 DB Credential Protection (EXP) (IDD_PRT_EX.1)

IDD_PRT_EX.1.1 The TSF shall ensure that configured IT System security credentials are stored in a secure manner.

5.3 TOE Security Assurance Requirements

The security assurance requirements for the TOE are the EAL 2 augmented with ALC_FLR.2 as specified in Part 3 of the Common Criteria.

Requirement Class	Requirement Component
ADV: Development	ADV_ARC.1: Security architecture description
	ADV_FSP.2: Security-enforcing functional specification
	ADV_TDS.1: Basic design
AGD: Guidance documents	AGD_OPE.1: Operational user guidance
	AGD_PRE.1: Preparative procedures
ALC: Life-cycle support	ALC_CMC.2: Use of a CM system
	ALC_CMS.2: Parts of the TOE CM coverage
	ALC_DEL.1: Delivery procedures
	ALC_FLR.2: Flaw reporting procedures
ATE: Tests	ATE_COV.1: Evidence of coverage
	ATE_FUN.1: Functional testing
	ATE_IND.2: Independent testing - sample
AVA: Vulnerability assessment	AVA_VAN.2: Vulnerability analysis

Table 3 EAL2 augmented with ALC_FLR.2 Assurance Components

5.3.1 Development (ADV)

5.3.1.1 Security architecture description (ADV_ARC.1)

ADV_ARC.1.1d
The developer shall design and implement the TOE so that the security features of the TSF cannot be bypassed.

ADV_ARC.1.2d
The developer shall design and implement the 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.

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 TSF initialisation 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.

ADV_ARC.1.1e
The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

5.3.1.2 Security-enforcing functional specification (ADV_FSP.2)

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.

ADV_FSP.2.1c
The functional specification shall completely represent the TSF.

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.

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.

5.3.1.3 Basic design (ADV_TDS.1)

ADV_TDS.1.1d
The developer shall provide the design of the TOE.

ADV_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.

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.

ADV_TDS.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.

5.3.2 Guidance documents (AGD)

5.3.2.1 Operational user guidance (AGD_OPE.1)

AGD_OPE.1.1d
The developer shall provide operational user guidance.

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.

AGD_OPE.1.1e

The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

5.3.2.2 Preparative procedures (AGD_PRE.1)

AGD_PRE.1.1d

The developer shall provide the TOE including its preparative procedures.

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.

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.

5.3.3 Life-cycle support (ALC)

5.3.3.1 Use of a CM system (ALC_CMC.2)

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.

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.

ALC_CMC.2.1e

The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

5.3.3.2 Parts of the TOE CM coverage (ALC_CMS.2)

ALC_CMS.2.1d

The developer shall provide a configuration list for the TOE.

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.

ALC_CMS.2.1e

The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

5.3.3.3 Delivery procedures (ALC_DEL.1)

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.

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.

ALC_DEL.1.1e

The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

5.3.3.4 Flaw reporting procedures (ALC_FLR.2)

ALC_FLR.2.1d

The developer shall document and provide flaw remediation procedures addressed to TOE developers.

ALC_FLR.2.2d

The developer shall establish a procedure for accepting and acting upon all reports of security flaws and requests for corrections to those flaws.

ALC_FLR.2.3d

The developer shall provide flaw remediation guidance addressed to TOE users.

ALC_FLR.2.1c

The flaw remediation procedures documentation shall describe the procedures used to track all reported security flaws in each release of the TOE.

ALC_FLR.2.2c

The flaw remediation procedures shall require that a description of the nature and effect of each security flaw be provided, as well as the status of finding a correction to that flaw.

ALC_FLR.2.3c

The flaw remediation procedures shall require that corrective actions be identified for each of the security flaws.

ALC_FLR.2.4c

The flaw remediation procedures documentation shall describe the methods used to provide flaw information, corrections and guidance on corrective actions to TOE users.

ALC_FLR.2.5c

The flaw remediation procedures shall describe a means by which the developer receives from TOE users reports and enquiries of suspected security flaws in the TOE.

ALC_FLR.2.6c

The procedures for processing reported security flaws shall ensure that any reported flaws are remediated and the remediation procedures issued to TOE users.

ALC_FLR.2.7c

The procedures for processing reported security flaws shall provide safeguards that any corrections to these security flaws do not introduce any new flaws.

ALC_FLR.2.8c

The flaw remediation guidance shall describe a means by which TOE users report to the developer any suspected security flaws in the TOE.

ALC_FLR.2.1e

The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

5.3.4 Tests (ATE)

5.3.4.1 Evidence of coverage (ATE_COV.1)

ATE_COV.1.1d

The developer shall provide evidence of the test coverage.

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.

ATE_COV.1.1e

The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

5.3.4.2 Functional testing (ATE_FUN.1)

ATE_FUN.1.1d

The developer shall test the TSF and document the results.

ATE_FUN.1.2d

The developer shall provide test documentation.

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.

ATE_FUN.1.1e

The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

5.3.4.3 Independent testing - sample (ATE_IND.2)

ATE_IND.2.1d

The developer shall provide the TOE for testing.

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.

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.

5.3.5 Vulnerability assessment (AVA)

5.3.5.1 Vulnerability analysis (AVA_VAN.2)

AVA_VAN.2.1d

The developer shall provide the TOE for testing.

AVA_VAN.2.1c

The TOE shall be suitable for testing.

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.

6. TOE Summary Specification

This chapter describes the security functions:

- Security Audit
- Security Management
- Database Data Collection and Monitoring

The TOE ensures that its mechanisms cannot be bypassed within its scope of control by requiring users to be identified and also authenticated by the operational environment. Only once a user is identified and authenticated and also found to be associated with one of the defined TOE-specific groups, access to the corresponding TOE functions will be granted.

Each AppRadar Sensor is installed and associated with an AppRadar Console. Each AppRadar Sensor component can be associated with only one console. The AppRadar Sensor component, when configured according to the evaluated guidance, will communicate only with its associated console using SSL (HTTP over SSL or SOAP over HTTPS) on ports designated during installation and configuration. The SSL implementation is provided by the OpenSSL component in the operational environment which is invoked by the web server front end (i.e., Tomcat) also provided by the operational environment to facilitate web-based TOE communications.

6.1 Security audit

The TOE generates audit records for all successful access to TOE audit data, and TOE configuration changes.

Note that the audit mechanism is always enabled and there is no capability to disable the audit. The Evaluated Configuration Guide provides instructions to configure the DbProtect Console host to create an audit record when the TOE Console starts and upon shutdown effectively showing when audit starts up and stops. However, the TOE itself always generates audit events while it is running and auditing cannot be disabled obviating any need to audit start-up and shutdown of audit which is simply not applicable.

The audit records are generated for all changes to the TOE configuration and include the date/time, type of event, the success or failure of the event if applicable, and identification of the user who caused the event, if applicable. Note that the identification and authentication function is performed by the OS and those events would be recorded in the context of the OS; however, the TOE is aware when user authentication is successful and whether the user is in one of the TOE roles and as a result the TOE logs attempts (successful and unsuccessful) to access the TOE user functions. TOE audit records are stored and protected in the Microsoft SQL Server database in the operational environment, which also contains the TOE configuration data and the database monitoring data. Audit records may be viewed by all TOE users through the Console using predefined report templates. To view audit records and alerts, users (both the AppRadar Administrator and AppRadar Viewer) use the Alerts, Dashboard, and Reports tabs in the AppRadar Console. The Alerts table allows the user to view all alerts. The Dashboard allows the user to see all types of audit records, including those records related to TOE access and configuration and to the databases being monitored by the TOE. The Reports tab allows the user to view predefined reports and to define new reports of audit records. Please see section 6.1.2 below for additional information on AppRadar Console tabs.

The TOE monitors the SQL database where the audit records are stored and detects access attempts and all modifications, just as it does with other monitored databases. In addition, the operational environment provides for protection of the database through non-bypassability, domain separation, and identification and authentication, all provided by the operating system. There is no automatic backup of the SQL database; instead, the AppRadar Administrator is instructed to periodically backup the database and to monitor its size and manage it outside of the TOE. This backup and monitoring instruction is provided in the guidance documentation for the TOE.

The Security audit function is designed to satisfy the following security functional requirements:

- FAU_GEN.1: The TOE generates audit events for data accesses and use of the TOE functions.
- FAU_SAR.1: The TOE provides the capability to review audit records.

6.2 Security management

The TOE performs security management to allow installation of TOE components, modifications to TOE policies and filters, creation of reports, and initializing and modifying the login ID and passwords of databases that are monitored by the TOE. Security management is performed by the AppRadar Administrators, who are assigned the role in the operational environment, i.e., they are members of the AppRadar Administrators Group defined in Windows. Note that roles and identification and authentication are the responsibility of the OS in the operational environment. AppRadar users include those with the AppRadar Administrator and AppRadar Viewer roles, i.e., members of the AppRadar Administrators Windows Group and the AppRadar Viewers Windows Group. AppRadar Administrators have access to all TOE functions. AppRadar Viewers can review AppRadar Sensors, Alerts, Policies, etc. (anything an AppRadar Administrator can view), but they cannot create, modify, or delete information. AppRadar Viewers can, however, run reports.

Security management functions are performed using the AppRadar Console. The AppRadar Console provides a user interface that allows access to a series of tabs and a set of workflows. The tabs include the following:

- The **Home** tab displays the AppRadar Console Home page, which includes links to workflows and general application information.
- The **Sensors** tab displays the **Sensor Manager**, which allows the AppRadar Administrator to configure and register AppRadar Sensors. This tab allows the AppRadar Administrator to initialize and modify the database login ID and password that is used by the TOE to access the monitored database, i.e., the database monitored by each sensor. A database login ID and password is stored in the TOE for each database that is monitored in order to provide needed the access needed by the sensor to perform database monitoring. The login ID and password is determined and initialized in the database by the appropriate database administrator; the TOE uses the login and password to access the database, just as any user would.
- The **Alerts** tab displays the **Alert Manager**, which allows the AppRadar Administrator to monitor, acknowledge, Filter, archive, and delete Alerts. The AppRadar Console receives Alerts from the deployed AppRadar Sensors.
- The **Policies** tab displays the **Policy Manager**, which allows the AppRadar Administrator to create, edit, import, export, delete and deploy Policies which associate Alert-triggering Rules with the AppRadar Sensors.
- The **Dashboard** tab displays the **Dashboard**, which provides a graphical, high level summary of Alerts, audits, and total audit volume from all of the sensors. Specific alert and audit information is also available in the Reports tab.
- The **Filters** tab displays the **Filter Manager**, which allows the AppRadar Administrator to create, edit, export, import and delete Filters, which customize the functionality of built-in Rules and Policies.
- The **Reports** tab displays **Report Manager**, which allows AppRadar Administrators and Viewers to generate reports for Alerts and audit events generated by the AppRadar Sensors.
- The **Email** tab displays the **Email Forwarding Rules** page, which allows the AppRadar Administrator to email Alerts, instead of/in addition to the SNMP trap or file methods specified during the configuration/deployment of an instance for a registered AppRadar Sensor.
- The **Help** tab displays the AppRadar online help.

The AppRadar Console workflows are accessible from the Home tab and provide an easy to follow guide through the various tabs and pages to perform specific functions, such as registering sensors, creating policies, and managing alerts.

The Security management functions performed by the AppRadar Console meet security functional requirements FMT_MTD.1 and FMT_SMF.1, which require that the AppRadar Administrator be able to initialize and modify database credentials for network-based databases; modify existing and create new policies and filters; and add additional destinations for alerts (i.e., specific SNMP and SMTP servers in the operational environment).

The Security management function is designed to satisfy the following security functional requirements:

- FMT_MTD.1: The TOE restricts the TOE management functions to the AppRadar Administrator.
- FMT_SMF.1: The TOE is able to perform the following security management functions: initialize and modify database credentials for network-based databases; modify existing and create new policies and filters; and add additional destinations for alerts.

6.3 Database data collection and monitoring

The TOE monitors database functions based on policies and filters predefined in the TOE and modified or enhanced by the AppRadar Administrator. The AppRadar Administrator performs modification and enhancement of policies and filters using the AppRadar Console described in section 6.1.2, above. Both normal database usage and security events are monitored and records are generated. Security events, as defined by TOE policies, cause the Sensors to generate alerts and audit records that are sent to the AppRadar Console. The Sensors are configured initially and their policies may be updated using the Sensors tab on the AppRadar Console. Sensors are configured with applicable policies by the AppRadar Administrator using the AppRadar Console. Once configured, the Sensor generates the Alerts and Audit records and the AppRadar Console "listens" for the Alerts and Audit Records on the AppRadar Console port specified for the Sensor during Sensor configuration.

For all of the databases monitored, the Sensors collect the following data and allow it to be displayed on the AppRadar Console:

- Identification and authentication events to the database, both successful and unsuccessful
- Predefined database attacks, including:
  - Buffer overflows
  - Predefined threats (based on known threat signatures related to file permissions and other database configuration data)
  - Password attacks
  - Privilege escalation
  - Database scans using security tools
  - Web Application attacks
- Other potential attacks defined by the AppRadar Administrator using the AppRadar Console Policy and Filter tabs.

Database monitoring is performed by AppRadar based on a set of predefined attacks and events, however, the AppRadar Administrator may define additional events to be monitored or modify the predefined events by modifying and creating new policies and filters using the data management capabilities of the AppRadar Console, described in section 6.1.2 above. For every event audited or alert generated, AppRadar collects the following information: Date and time of the event, type of event, subject identity, and outcome of the event. Note that the TOE uses a reliable time source in the operational environment for the date and time of the event.

The Sensors send alarms to the AppRadar console, which includes the Message Collector component that listens on HTTPS traffic port specified for that Sensor during Sensor configuration. Alerts are sent to the AppRadar Console and, if configured by the AppRadar Administrator, the AppRadar Console will forward the Alerts to a log file, to the SNMP Trap Receiver (in the operational environment). Using the AppRadar Console, the AppRadar Administrator may also specify that alerts be emailed (to an SMTP server in the operational environment).

As indicated earlier, there are two types or sensors: host-based and network-based. Both types of sensors can be configured similarly though they are each constrained by the means by which they can access/perceive database activity. Network-based sensors perceive activity from a network perspective where network traffic is analyzed to discern database activities. Host-based sensors are co-resident on the database host and are configured with database credentials for direct access to the database and can capture of SQL commands. Given they are on the same host, they can perceive local (non network) activity, but generally only perceive successful operations. Regardless, within their scope of perception both types of sensors perform common security monitoring functions.

The TOE can access some targeted IT systems using privileged accounts in order to better monitor applicable database. Applicable security credentials can be configured into the TOE and the TOE will store and protect those credentials using capabilities provided by its environment. For the Backend database, the TOE can be configured to

use database authentication or use Windows authentication. In the latter case, it doesn't store any crednetials to access the database, but rather relies on Windows to facilitate authentication. In the former case, the TOE calls upon the Windows Data Protection API (DPAPI) to encrypt and store the credentials in its registry so that they can be recalled by the TOE when needed for database access. For target monitored database systems, OpenSSL (AES-256) is used by the AppRadar Console to encrypt/decrypt applicable credentials with Sensor-specific keys which are stored in the Backend Database. The individual database credentials are stored locally (in the host filesystem) in encrypted form by each host-based Sensor and sent to the AppRadar Console to be decrypted when needed. Network-based sensors do not use database credentials.

The database data collection and monitoring function is designed to satisfy the following security functional requirements:

- IDD_DDC_EX.1: The TOE provides the means to collect and analyze pertinent data as described above.
- IDD_DDR_EX.1: The TOE provides the ability to send alerts prompted by analysis findings.
- IDD_PRT_EX.1: The TOE protects credentials used to access other IT systems in the operational environment.

7. Protection Profile Claims

There are no Protection Profile claims in this Security Target.

8. Rationale

This section provides the rationale for completeness and consistency of the Security Target. The rationale addresses the following areas:

- Security Objectives;
- Security Functional Requirements;
- Security Assurance Requirements;
- Requirement Dependencies; and
- TOE Summary Specification.

8.1 Security Objectives Rationale

This section shows that all secure usage assumptions and threats are completely covered by security objectives. In addition, each objective counters or addresses at least one assumption or threat.

8.1.1 Security Objectives Rationale for the TOE and Operational Environment

This section provides evidence demonstrating the coverage of threats and usage assumptions by the security objectives.

	T.COMM	T.IMPCON	T.INADVE	T.MISACT	T.MISUSE	T.PRIVIL	T.PROTECT	A.ACCESS	A.DYNMIC	A.LOCATE	A.MANAGE	A.NOEVIL	A.NOTRST	A.PROTCT
O.AUDIT		X				X								
O.AUDIT_REVIEW		X	X	X	X	X								
O.MANAGE		X				X								
O.MONITOR			X	X	X									
OE.CREDEN												X	X	
OE.IDENT_AUTHEN		X				X								
OE.INSTAL												X		
OE.INTROP								X	X					
OE.PERSON									X		X			
OE.PHYCAL										X		X	X	X
OE.PROTCT	X						X							
OE.ROLES		X				X								
OE.TIME		X	X	X	X	X								

Table 4 Environment to Objective Correspondence

8.1.1.1 T.COMM

Data transmitted between TOE components may be captured and read by an attacker, thereby compromising sensitive TOE data

This Threat is satisfied by ensuring that:

- OE.PROTCT.: This objective ensures that all data transmitted between TOE components is protected from unauthorized modification or disclosure.

8.1.1.2 T.IMPCON

An unauthorized user may inappropriately change the configuration of the TOE causing potential intrusions to go undetected.

This Threat is satisfied by ensuring that:

- O.AUDIT: This objective ensures that all attempts to change the configuration of the TOE are audited.
- O.AUDIT_REVIEW: This objective ensures the TOE users can view audit records of changes and attempted changes to the TOE, thereby ensuring detection of configuration changes.
- O.MANAGE: This objective ensures that the TOE provides the capability for authorized user to manage the TOE configuration.
- OE.IDENT_AUTHEN: This objective ensures that the operational environment identifies and authenticates users prior to any TOE function accesses.
- OE.ROLES: This objective ensures that the operational environment assigns roles to users, thereby limiting TOE configuration functions to only those users with the appropriate role.
- OE.TIME: This objective ensures that the operational environment provides a reliable time source for the TOE to use to accurately timestamp audit records.

8.1.1.3 T.INADVE

Inadvertent activity and access by unauthorized or authorized users may be undetected on a database the TOE monitors thereby compromising the data maintained in the database.

This Threat is satisfied by ensuring that:

- O.AUDIT_REVIEW: This objective ensures that the TOE users can view audit records of database usage, events, and alerts, thereby ensuring detection of unauthorized accesses and activity indicative of misuse.
- O.MONITOR: This objective ensures that the TOE monitors the registered databases for defined attacks and normal activity.
- OE.TIME: This objective ensures that the operational environment provides a reliable time source for the TOE to use to accurately timestamp audit records.

8.1.1.4 T.MISACT

Malicious activity by attackers may be undetected on a database the TOE monitors, thereby compromising the data maintained in the database.

This Threat is satisfied by ensuring that:

- O.AUDIT_REVIEW: This objective ensures that the TOE users can view audit records of database usage, events, and TOE generated alerts, thereby ensuring detection of unauthorized accesses and activity indicative of misuse.
- O.MONITOR: This objective ensures that the TOE monitors the registered databases for defined attacks and normal activity.
- OE.TIME: This objective ensures that the operational environment provides a reliable time source for the TOE to use to accurately timestamp audit records.

8.1.1.5 T.MISUSE

Unauthorized accesses and activity indicative of misuse by unauthorized or authorized users may be undetected on a database the TOE monitors, thereby compromising the data maintained in the database.

This Threat is satisfied by ensuring that:

- O.AUDIT_REVIEW: This objective ensures the TOE users can view audit records of database usage, events, and alerts, thereby ensuring detection of unauthorized accesses and activity indicative of misuse.
- O.MONITOR: This objective ensures that the TOE monitors the registered databases for defined attacks and normal activity.
- OE.TIME: This objective ensures that the operational environment provides a reliable time source for the TOE to use to accurately timestamp audit records.

8.1.1.6 T.PRIVIL

An unauthorized user may gain undetected access to the TOE and exploit system privileges to gain access to TOE security functions and data.

This Threat is satisfied by ensuring that:

- O.AUDIT: This objective ensures that all access to TOE data access is audited.
- O.AUDIT_REVIEW: This objective ensures the TOE users can view audit records of changes and attempted changes to the TOE, thereby ensuring detection of data access.
- O.MANAGE: This objective ensures that the TOE provides the capability for authorized user to manage the TOE configuration.
- OE.IDENT_AUTHEN: This objective ensures that the operational environment identifies and authenticates users prior to any TOE function accesses.
- OE.ROLES: This objective ensures that the operational environment assigns roles to users, thereby limiting TOE configuration functions to only those users with the appropriate role.
- OE.TIME: This objective ensures that the operational environment provides a reliable time source for the TOE to use to accurately timestamp audit records.

8.1.1.7 T.PROTECT

An attacker may access and modify TOE data assets and configuration data, causing the TOE functions to work incorrectly and causing loss of TOE data and database monitoring functions.

This Threat is satisfied by ensuring that:

- OE.PROTCT: This objective ensures that the operational Environment protects the TOE software and data and the databases that are monitored.

8.1.1.8 A.ACCESS

The TOE has access to all the IT System data it needs to perform its functions.

This Assumption is satisfied by ensuring that:

- OE.INTROP: Ensures the TOE has the needed access.

8.1.1.9 A.DYNMIC

The TOE will be managed in a manner that allows it to appropriately address changes in the IT System the TOE monitors.

This Assumption is satisfied by ensuring that:

- OE.INTROP: Ensures the TOE has the proper access to the IT System.
- OE.PERSON: Ensures that the TOE will managed appropriately.

8.1.1.10 A.LOCATE

The processing resources of the TOE, i.e., the sensors and console will be located within controlled access facilities, which will prevent unauthorized physical access. Remote access to the console component of the TOE is possible outside the controlled access facilities.

This Assumption is satisfied by ensuring that:

- OE.PHYCAL: Provides for the physical protection of the TOE.

8.1.1.11 A.MANAGE

There will be one or more competent individuals assigned to manage the TOE and the security of the information it contains.

This Assumption is satisfied by ensuring that:

- OE.PERSON: Ensures all authorized administrators are qualified and trained to manage the TOE.

8.1.1.12 A.NOEVIL

The authorized administrators are not careless, willfully negligent, or hostile, and will follow and abide by the instructions provided by the TOE documentation.

This Assumption is satisfied by ensuring that:
- OE.CREDEN: Supports this assumption by requiring protection of all authentication data.
- OE.INSTAL: Ensures that the TOE is properly installed and operated.
- OE.PHYCAL: Provides for physical protection of the TOE by authorized administrators.

8.1.1.13 A.NOTRST

The TOE can only be accessed by authorized users.

This Assumption is satisfied by ensuring that:
- OE.CREDEN: Supports this assumption by requiring protection of all authentication data.
- OE.PHYCAL: Provides for physical protection of the TOE to protect against unauthorized access.

8.1.1.14 A.PROTCT

The TOE software critical to security policy enforcement will be protected from unauthorized physical modification.

This Assumption is satisfied by ensuring that:
- OE.PHYCAL: Provides for the physical protection of the TOE hardware and software.

8.2 Security Requirements Rationale

This section provides evidence supporting the internal consistency and completeness of the components (requirements) in the Security Target. Note that **Table 5** indicates the requirements that effectively satisfy the individual objectives.

8.2.1 Security Functional Requirements Rationale

All Security Functional Requirements (SFRs) identified in this Security Target are fully addressed in this section and each SFR is mapped to the objective for which it is intended to satisfy.

	O.AUDIT	O.AUDIT_REVIEW	O. MANAGE	O.MONITOR
FAU_GEN.1	X			
FAU_SAR.1		X		
FMT_MTD.1			X	
FMT_SMF.1			X	
IDD_DDC_EX.1				X

IDD_DDR_EX.1				X
IDD_PRT_EX.1				X

Table 5 Objective to Requirement Correspondence

8.2.1.1 O.AUDIT

The TOE must record audit records for use of the TOE functions.

This TOE Security Objective is satisfied by ensuring that:

- FAU_GEN.1: Security-relevant events must be defined and auditable for the TOE.

8.2.1.2 O.AUDIT_REVIEW

The TOE must provide a means for TOE users to review all audit and database monitoring data.

This TOE Security Objective is satisfied by ensuring that:

- FAU_SAR.1: The TOE must provide users with the ability to review audit records and the records generated by sensors for the monitored databases.

8.2.1.3 O.MANAGE

The TOE must allow authorized users to manage appropriate TOE functions.

This TOE Security Objective is satisfied by ensuring that:

- FMT_MTD.1: The TOE must ensure that only Administrators can perform the management functionality associated with their role as identified in FMT_MTD.1.
- FMT_SMF.1: The TOE must perform appropriate TOE management functionality as defined in FMT_SMF.1.

8.2.1.4 O.MONITOR

The TOE must monitor databases for defined attacks and normal activity and generate alerts for defined attacks and audit records for normal activity that can be viewed by TOE users.

This TOE Security Objective is satisfied by ensuring that:

- IDD_DDC_EX.1: The TOE must collect information from monitored databases including identification and authentication events, predefined database attacks, and other policies defined by the Administrator and must collect and record the following information: Date and time of the event, type of event, subject identity if known, application that triggered the event if available, client host IPv4 address if known.
- IDD_DDR_EX.1: The TOE must send an alarm to the AppRadar console and other destinations defined by the Administrator when a monitored security event is detected.
- IDD_PRT_EX.1: The TOE must protect the credentials used when monitoring databases so as to mitigate the possibility of added exposure of those IT systems.

8.3 Security Assurance Requirements Rationale

EAL2 was selected as the assurance level because the TOE is a commercial product whose users require a low to moderate level of independently assured security. Application Security, AppRadar is targeted for an environment with good physical access security and competent administrators. Within such environment, it is assumed that attackers will have little attack potential. As such, EAL2 is appropriate to provide the assurance necessary to counter the limited potential for attack. EAL2 is augmented with ALC_FLR.2, Flaw reporting procedures to provide instructions to users for reporting flaws, to provide internal vendor procedures for identifying, tracking and correcting product flaws.

8.4 Requirement Dependency Rationale

The following table identifies the dependencies of the requirements in this ST, including the extended requirements defined in this ST. As indicated in the table, all of the dependencies are satisfied, except those identified in **[bold-red-bracketed]** text.

ST Requirement	CC Dependencies	ST Dependencies
FAU_GEN.1	FPT_STM.1	**[FPT_STM.1]**
FAU_SAR.1	FAU_GEN.1	FAU_GEN.1
FMT_MTD.1	FMT_SMR.1 and FMT_SMF.1	**[FMT_SMR.1]** and FMT_SMF.1
FMT_SMF.1	none	none
IDD_DDC_EX.1	none	none
IDD_DDR_EX.1	IDD_DDC_EX.1	IDD_DDC_EX.1
IDD_PRT_EX.1	none	none
ADV_ARC.1	ADV_FSP.1 and ADV_TDS.1	ADV_FSP.2 and ADV_TDS.1
ADV_FSP.2	ADV_TDS.1	ADV_TDS.1
ADV_TDS.1	ADV_FSP.2	ADV_FSP.2
AGD_OPE.1	ADV_FSP.1	ADV_FSP.2
AGD_PRE.1	none	none
ALC_CMC.2	ALC_CMS.1	ALC_CMS.2
ALC_CMS.2	none	none
ALC_DEL.1	none	none
ALC_FLR.2	none	none
ATE_COV.1	ADV_FSP.2 and ATE_FUN.1	ADV_FSP.2 and ATE_FUN.1
ATE_FUN.1	ATE_COV.1	ATE_COV.1
ATE_IND.2	ADV_FSP.2 and AGD_OPE.1 and AGD_PRE.1 and ATE_COV.1 and ATE_FUN.1	ADV_FSP.2 and AGD_OPE.1 and AGD_PRE.1 and ATE_COV.1 and ATE_FUN.1
AVA_VAN.2	ADV_ARC.1 and ADV_FSP.2 and ADV_TDS.1 and AGD_OPE.1 and AGD_PRE.1	ADV_ARC.1 and ADV_FSP.2 and ADV_TDS.1 and AGD_OPE.1 and AGD_PRE.1

Table 6 Requirement Dependencies

The TOE relies on the environment to provide reliable time stamps per OE.TIME. Further, the TOE relies on the operational environment to assign users to roles per OE.ROLES so that the TOE can appropriately enforce restrictions on its administrative functions. As such, both of the missing dependencies identified in the table above are satisfied with objectives for the operational environment of the TOE.

8.5 TOE Summary Specification Rationale

Each subsection in Section 6, the TOE Summary Specification, describes a security function of the TOE. Each description is followed with rationale that indicates which requirements are satisfied by aspects of the corresponding security function. The set of security functions work together to satisfy all of the security functions and assurance requirements. Furthermore, all of the security functions are necessary in order for the TSF to provide the required security functionality.

This Section in conjunction with Section 6, the TOE Summary Specification, provides evidence that the security functions are suitable to meet the TOE security requirements. The collection of security functions work together to provide all of the security requirements. The security functions described in the TOE summary specification are all necessary for the required security functionality in the TSF. **Table 7 Security Functions vs. Requirements Mapping** demonstrates the relationship between security requirements and security functions.

	Security Audit	Security Management	Database data collection and monitoring
FAU_GEN.1	X		
FAU_SAR.1	X		
FMT_MTD.1		X	
FMT_SMF.1		X	
IDD_DDC_EX.1			X
IDD_DDR_EX.1			X
IDD_PRT_EX.1			X

Table 7 Security Functions vs. Requirements Mapping