www.GossamerSec.com

Assurance Activity Report for BlackBerry UEM Server and Android Client v12

Version 0.3
05/29/24

Prepared by:
Gossamer Security Solutions
Accredited Security Testing Laboratory – Common Criteria Testing
Columbia, MD 21045

Prepared for:
National Information Assurance Partnership
Common Criteria Evaluation and Validation Scheme

Document: AAR-11427

Revision History

Revision	Date	Authors	Summary
Version 0.1	05/03/24	Gossamer	Initial draft
Version 0.2	05/22/24	Gossamer	Addressed ECR Comments
Version 0.3	05/29/24	Gossamer	Addressed ECR Comment

The TOE Evaluation was Sponsored by:

BlackBerry
2240 University Ave. E
Waterloo, ON N2K 0A9
Canada

Evaluation Personnel:

- Douglas Kalmus

Common Criteria Versions:

- Common Criteria for Information Technology Security Evaluation Part 1: Introduction, Version 3.1, Revision 5, April 2017
- Common Criteria for Information Technology Security Evaluation Part 2: Security functional components, Version 3.1, Revision 5, April 2017
- Common Criteria for Information Technology Security Evaluation Part 3: Security assurance components, Version 3.1, Revision 5, April 2017

Common Evaluation Methodology Versions:

- Common Methodology for Information Technology Security Evaluation, Evaluation Methodology, Version 3.1, Revision 5, April 2017

Table of Contents

1. Introduction

This document presents evaluations results of the BlackBerry UEM Server and Android Client MDMPP40/MDMA10/PKGTLS11 evaluation. This document contains a description of the assurance activities and associated results as performed by the evaluators.

1.1 References

The following guidance documentation included material used to satisfy the Guidance assurance activities.

[ST] BlackBerry UEM Server and Android Client v12 Security Target

[Admin] BlackBerry UEM Administrative Guidance Document, UEM Version 12.19, May 2024

Other Documentation References

[MDMPP40] Protection Profile for Mobile Device Management, Version 4.0, 25 April 2019

[MDMA10] PP-Module for MDM Agents, Version 1.0, 25 April 2019

[PKGTLS11] Functional Package for Transport Layer Security (TLS), Version 1.1, 11 February 2019

1.2 CAVP Certificate Justification

The following pertains to the applicability of the certificates referenced throughout Section **Error! Reference source not found.**, "**Error! Reference source not found.**."

The TOE is a software application made up of the BlackBerry Unified Endpoint Management (UEM) Server and UEM Android Client. The UEM Android Client utilizes cryptographic functions provided by its platform. Refer to the evaluation material for platforms for details. The UEM Server uses its Certicom Security Builder GSE-J Crypto Core (version 2.9.2) library to perform all cryptographic operations.

The BlackBerry UEM server, including the Core and UI security enforcing components, is implemented with a combination of Java and native code running on Windows Server 2016 or Windows Server 2019 with Java JRE 17. Ideally, the scope of supported platforms for the evaluation would be Windows Server 2016 or 2019 wherever it is deployable, however, it will be limited due to NIAP policy about CAVP algorithm certificates – the allowed environments would be expected to conform to the environments of the CAVP certificates (e.g., using the processors used for CAVP algorithm testing). In this case, the CAVP testing for Certicom was done on Windows Server 2016 and 2019 running in a virtual environment (VMWare ESXi 7.0) on an Intel Xeon E5-2670.

The TOE uses the Certicom Security Builder GSE-J Crypto Core (version 2.9.2) library for all cryptographic operations.

Functions	Requirement	Certificate
Encryption/Decryption		
AES CBC (128 and 256 bits) AES-GCM (128 and 256 bits)	MDMPP40:FCS_COP.1(1)	A5201
Cryptographic hashing		
SHA-256, SHA-384, SHA-512, (digest sizes 256, 384, 512)	MDMPP40:FCS_COP.1(2)	A5201
Cryptographic signature services		
RSA Digital Signature Algorithm (rDSA) (modulus 2048, 3072, 4096)	MDMPP40:FCS_COP.1(3)	A5201
ECDSA schemes using 'NIST curves' P-256/384/521	MDMPP40:FCS_COP.1(3)	A5201
Keyed-hash message authentication		
HMAC-SHA-256, HMAC-SHA-384, HMAC-SHA-512 (key size = block size)	MDMPP40:FCS_COP.1(4)	A5201
Random bit generation		
Hash_DRBG (any), HMAC_DRBG (any), CTR_DRBG (AES); all 256-bits	MDMPP40:FCS_RBG_EXT.1	A5201
Key Generation		
RSA Key Generation MDM Server – 2048 bits, 3072 bits, 4096 bits	MDMPP40:FCS_CKM.1	A5201
ECC Schemes MDM Server SigGen/SigVer P-256, P384, P-521		A5201
Key Establishment		
RSA Key Establishment MDM Server	MDMPP40:FCS_CKM.2	Tested with known good implementation
EC-based Key Establishment MDM Server - P-256/384/521		A5201

2. Protection Profile SFR Assurance Activities

2.1 Security audit (FAU)

2.1.1 Server Alerts (MDMPP40:FAU_ALT_EXT.1)

2.1.1.1 MDMPP40:FAU_ALT_EXT.1.1

TSS Assurance Activities: None Defined

Guidance Assurance Activities: None Defined

Testing Assurance Activities: None Defined

Component TSS Assurance Activities: The evaluator shall examine the TSS and verify that it describes how the alert system is implemented. The evaluator shall also verify that a description of each assigned event is provided in the TSS.

Section 6.1 of [ST] indicates that the UEM Server has configurable administrator notifications, including changes in device enrollment status and failure to apply policies to mobile devices. These administrator notifications may be forwarded by email to a configured administrator.

Component Guidance Assurance Activities: The evaluator shall examine the guidance document and verify that it describes how the alerts can be configured, if configurable.

Section entitled "Create an event notification" in [Admin] provides instructions to define an event notification to alert the administrator about selected activities. The instructions describe how to configure email messages to be sent to an administrator as a notification of 'enrollment' or of 'policy or profile delivery failure'.

Component Testing Assurance Activities: For each MDM Agent/platform listed as supported in the ST:

Test 1: The evaluator shall enroll a device and ensure that the MDM server alerts the administrator of the change in enrollment status. The evaluator shall unenroll (retire) a device and ensure that the MDM server alerts the administrator of the FAU_GEN.1.1(1) change in enrollment status.

Test 2: The evaluator shall configure policies, which the MDM agent should not be able to apply. These policies shall include: a setting which is configurable on the MDM Server interface but not supported by the platform on which the MDM Agent runs, if any such settings exist a valid configuration setting with an invalid parameter, which

may require manual modification of the policy prior to transmission to the device The evaluator shall deploy such policies and verify that the MDM server alerts the administrator about the failed application of the policy.

Test 3: (Conditional) The evaluator shall trigger each of the events listed and ensure that the MDM Server alerts the administrator.

Test 1: The evaluator configured alerts and proceeded to enroll and unenroll a device. The evaluator observed that the alerts were properly generated in response to the action performed.

Test 2: The evaluator configured policies as defined below and ensured that the failure condition was detected and an alert was sent to the administrator using the configured methods.

1. Configure a policy that is not supported by the platform and verify that the MDM server alerts the administrator.

2. Manually modify a policy prior to transmission to the device such that it contains a valid configuration variable with an invalid value, then verify that the MDM server alerts the administrator.

Test 3 was not performed as there were no additional events beyond those tested by the preceding test cases.

2.1.2 Agent Alerts (MDMA10:FAU_ALT_EXT.2)

2.1.2.1 MDMA10:FAU_ALT_EXT.2.1

TSS Assurance Activities: None Defined

Guidance Assurance Activities: None Defined

Testing Assurance Activities: None Defined

2.1.2.2 MDMA10:FAU_ALT_EXT.2.2

TSS Assurance Activities: None Defined

Guidance Assurance Activities: None Defined

Testing Assurance Activities: None Defined

Component TSS Assurance Activities: The evaluator shall examine the TSS and verify that it describes how the alerts are implemented.

The evaluator shall examine the TSS and verify that it describes how the candidate policy updates are obtained and the actions that take place for successful (policy update installed) and unsuccessful (policy update not installed) cases. The software components that are performing the processing must also be identified in the TSS and verified by the evaluator.

The evaluator also ensures that the TSS describes how reachability events are implemented, and if configurable are selected in FMT_SMF_EXT.4.2. The evaluator verifies that this description clearly indicates who (MDM Agent or MDM Server) initiates reachability events.

The evaluator shall ensure that the TSS describes under what circumstances, if any, the alert may not be generated (e.g., the device is powered off or disconnected from the trusted channel), how alerts are queued, and the maximum amount of storage for queued messages.

Section 6.1 of [ST] indicates that alerts are sent across the mutually authenticated TLS channel between the Agent and the UEM Server available once the device is enrolled. If this channel is not available, alerts are queued and forwarded when the channel is re-established. The local file storage is limited only by the space available on the mobile device.

Section 6.1 of [ST] explains that the UEM Android Client can send a range of alerts to the UEM Server including: when policies are applied; reachability status (which could be triggered by the Server or Client); change in enrollment state; and failure to install or update an application from the MAS Server.

This section also indicates that when the UEM Android Client receives a candidate policy, it is checked according to MDMA10:FMT_POL_EXT.2 where an alert is sent if the policy is not accepted (i.e., fails its signature check). If the check succeeds, the UEM Android Client checks each policy setting and applies the settings that are valid for the given device using available management APIs.

Section 6.1 of [ST] indicates that an administrator can query a device to obtain its current status. If the specified device does not have network connectivity, the UEM Android Client queues the query and delivers it when the device next contacts the UEM Server. This section describes that reachability events can be initiated by the UEM Android Client when it sends any alerts or other messages to the UEM Server and alternately can be initiated by the UEM Server where it can make requests of the UEM Android Client. The reachability status on the UEM Server is based on any secure communication with the UEM Android Client.

Component Guidance Assurance Activities: None Defined

Component Testing Assurance Activities: Test 1: The evaluator shall perform a policy update from the test environmentMDM server. The evaluator shall verify the MDM Agent accepts the update, makes the configured changes, and reports the success of the policy update back to the MDM Server.

Test 2: The evaluator shall perform each of the actions listed in FAU_ALT_EXT.2.1 and verify that the alert does in fact reach the MDM Server.

Test 3: The evaluator shall configure the MDM Agent to perform a network reachability test, both with and without such connectivity and ensure that results reflect each.

Test 4: The evaluator shall remove network connectivity from the MDM Agent and generate an alert/event as defined in FAU_ALT_EXT.2.1. The evaluator shall restore network connectivity to the MDM Agent and verify that the alert generated while the TOE was disconnected is sent by the MDM Agent upon re-establishment of the connectivity.

Test 1: The evaluator used the UEM Web UI to push a device policy and observed that the device installed the policy, acknowledged the policy to the UEM Server, and reported success to the UEM Server which caused the server to send an alert email.

Test 2: The evaluator performed each of the following actions from the UEM Server and verified that the UEM Android Client generated alerts:

- successful application of policies to a mobile device
- receiving and generating periodic reachability events
- change in enrollment state
- failure to install an application from the MAS Server
- failure to update an application from the MAS Server

Test 3 & 4: The evaluator confirmed that network connectively on a target device with network connectivity functioned as expected. The evaluator observed that connectivity status was updated following a device command being sent from the UEM Server. The evaluator then enabled airplane mode and issued a device command. After several minutes the evaluator disabled airplane mode and observed that the “Last Seen” time for the mobile device updated itself without further action by the evaluator.

2.1.3 Audit Data Generation (MDMPP40:FAU_GEN.1(1))

2.1.3.1 MDMPP40:FAU_GEN.1.1(1)

TSS Assurance Activities: The evaluator shall check the TSS and ensure that it lists all of the auditable events. The evaluator shall check to make sure that every audit event type mandated by the PP is described in the TSS. The

evaluator shall verify that for every audit event described in FAU_GEN.1.2(1) the TSS, the description indicates where the audit event is generated (TSF, TOE platform).

If 'invoke platform-provided functionality' is selected, the evaluator shall examine the TSS to verify that it describes (for each supported platform) how this functionality is invoked (it should be noted that this may be through a mechanism that is not implemented by the MDM Server; nonetheless, that mechanism will be identified in the TSS as part of this evaluation activity).

Section 6.1 of [ST] references Table 5-2 within the ST (which is presented in section 5.1.1.3) which provides a complete list of audit events and their contents in table form. The events in this table match the events listed in Table 2 of the [ST].

Guidance Assurance Activities: The evaluator shall check the administrative guide and ensure that it lists all of the auditable events. The evaluator shall check to make sure that every audit event type mandated by the PP is described.

The evaluator shall also make a determination of the administrative actions that are relevant in the context of this PP including those listed in the Management section. The evaluator shall examine the administrative guide and make a determination of which administrative commands are related to the configuration (including enabling or disabling) of the mechanisms implemented in the TOE that are necessary to enforce the requirements specified in the PP. The evaluator shall document the methodology or approach taken while determining which actions in the administrative guide are security relevant with respect to this PP. The evaluator may perform this activity as part of the activities associated with ensuring the AGD_OPE guidance satisfies the requirements.

The section entitled "Auditing" in [Admin] contains information about the audit mechanism provided by the UEM Server and UEM Android Client. This information includes details about the audit records which the TOE generates including details encompassing the required content. The tables in section entitled "Auditing events in BlackBerry UEM" of [Admin] provide a complete mapping of required audit events to the TOE generated events.

Testing Assurance Activities: The evaluator shall test the TOEs ability to correctly generate audit records by having the TOE generate audit records for the events listed in the provided table and administrative actions. This should include all instances of an event. The evaluator shall test that audit records are generated for the establishment and termination of a channel for each of the cryptographic protocols contained in the ST. For administrative actions, the evaluator shall test that each action determined by the evaluator above to be security relevant in the context of this PP is auditable.

Note that the testing here can be accomplished in conjunction with the testing of the security mechanisms directly. For example, testing performed to ensure that the administrative guidance provided is correct verifies

that AGD_OPE.1 is satisfied and should address the invocation of the administrative actions that are needed to verify the audit records are generated as expected.

The evaluator constructed a list of required audit events based on the TOE Security Target [ST]. The evaluator also verified that the events from the ST could be found in the AGD documentation, and that the information in the AGD was accurate enough to locate actual audit events generated by the TOE. The evaluator then either identified events generated when performing a corresponding function in another test or performed operations directly (e.g., log in and log out) to cause each required audit event to be generated. The evaluator captured screen shots and/or textual content that is included in the proprietary detailed test report.

2.1.3.2 MDMPP40:FAU_GEN.1.2(1)

TSS Assurance Activities: The evaluator shall check the TSS and ensure that it provides a format for audit records. Each audit record format type must be covered, along with a brief description of each field.

MDMPP40: The evaluator shall check the TSS and ensure that it provides a format for audit records. Each audit record format type must be covered, along with a brief description of each field.

Section 6.1 of [ST] indicates the TOE generates audit records with the content of audit events including the additional content specified in the table 2 of [ST]. Each event in the TOE's audit log includes a date/time stamp, event type and category, user, host, success indicator, and additional information for specific events (indicated in the third column of the Table 2).

Guidance Assurance Activities: The evaluator shall check the administrative guide and ensure that it provides a format for audit records. Each audit record format type must be covered, along with a brief description of each field. The evaluator shall check to make sure that the description of the fields contains the information required in FAU_GEN.1.2(1).

MDMPP40: The evaluator shall check the administrative guide and ensure that it provides a format for audit records. Each audit record format type must be covered, along with a brief description of each field. The evaluator shall check to make sure that the description of the fields contains the information required in FAU_GEN.1.2(1).

The tables in section entitled "Auditing events in BlackBerry UEM" of [Admin] provide a complete mapping of required audit events to the TOE generated events.

Testing Assurance Activities: When verifying the test results from FAU_GEN.1.1(1), the evaluator shall ensure the audit records generated during testing match the format specified in the administrative guide, and that the fields in each audit record have the proper entries.

Note that the testing here can be accomplished in conjunction with the testing of the security mechanisms directly. For example, testing performed to ensure that the administrative guidance provided is correct verifies

that AGD_OPE.1 is satisfied and should address the invocation of the administrative actions that are needed to verify the audit records are generated as expected.

The Auditable Events table includes optional, selection-based and objective requirements. The auditing of these requirements are only required if the requirement is included in the ST.

(man) - mandatory requirement

(sel) - selection-based requirement

(obj) - objective requirement

(opt) - optional requirement

MDMPP40: When verifying the test results from FAU_GEN.1.1(1), the evaluator shall ensure the audit records generated during testing match the format specified in the administrative guide, and that the fields in each audit record have the proper entries.

Note that the testing here can be accomplished in conjunction with the testing of the administrative guidance provided is correct verifies that AGD_OPE.1 is satisfied and should address the invocation of the administrative actions that are needed to verify the audit records are generated as expected.

The Auditable Events table includes optional, selection-based and objective requirements. The auditing of these requirements are only required if the requirement is included in the ST.

(man) - mandatory requirement

(sel) - selection-based requirement

(obj) - objective requirement

(opt) - optional requirement

While performing the FAU_GEN.1(1).1 tests, the evaluators collected and confirmed that the required audit record content was present in each audit record.

Component TSS Assurance Activities: None Defined

Component Guidance Assurance Activities: None Defined

Component Testing Assurance Activities: None Defined

2.1.4 Audit Data Generation (MDMA10:FAU_GEN.1(2))

2.1.4.1 MDMA10:FAU_GEN.1.1(2)

TSS Assurance Activities: None Defined

Guidance Assurance Activities: None Defined

Testing Assurance Activities: None Defined

2.1.4.2 MDMA10:FAU_GEN.1.2(2)

TSS Assurance Activities: None Defined

Guidance Assurance Activities: None Defined

Testing Assurance Activities: None Defined

Component TSS Assurance Activities: The evaluator shall check the TSS and ensure that it provides a format for audit records. Each audit record format type must be covered, along with a brief description of each field.

If 'invoke platform-provided functionality' is selected, the evaluator shall examine the TSS to verify that it describes (for each supported platform) how this functionality is invoked (it should be noted that this may be through a mechanism that is not implemented by the MDM Agent; nonetheless, that mechanism will be identified in the TSS as part of this evaluation activity).

In the architecture of this product, the MAS server is a feature embedded within the UEM Server and is not a distinct architectural component. The audits for the MAS server were described in the table of auditable events found in Section 6.1 of [ST], in a manner identical to the audits for the UEM Server.

Component Guidance Assurance Activities: None Defined

Component Testing Assurance Activities: The evaluator shall use the TOE to perform the auditable events defined in the Auditable Events table in FAU_GEN.1.1(2) and observe that accurate audit records are generated with contents and formatting consistent with those described in the TSS. Note that this testing can be accomplished in conjunction with the testing of the security mechanisms directly.

The evaluator constructed a list of required audit events based on the TOE Security Target [ST] and MDMA10:FAU_GEN.1(2). The evaluator then either identified events generated when performing a corresponding function in another test or performed operations directly (e.g., log in and log out) to cause each required audit event to be generated. The evaluator captured screen shots and/or textual content that is included in the

proprietary detailed test report. While performing these actions, the evaluators collected and confirmed that the required audit record content was present in each audit record.

2.1.5 Audit Generation (MAS Server) (MDMPP40:FAU_GEN.1(2))

2.1.5.1 MDMPP40:FAU_GEN.1.1(2)

TSS Assurance Activities: The evaluator shall check the TSS and ensure that it provides a format for audit records.

Section 6.1 of [ST] explains that the UEM Android Client uses the mobile device audit store, so the required audit events (see Table 3) are recorded directly in the mobile device security log and subject to any filtering configured in the mobile device. This section also indicates that each event in the audit log includes a date/time stamp, event type, subject identity, success indicator, and additional information for specific events (indicated in the third column of the Table 3).

Guidance Assurance Activities: The evaluator shall check the administrative guide and ensure that it provides a format for audit records. Each audit record format type must be covered, along with a brief description of each field.

The section entitled "Auditing" in [Admin] contains information about the audit mechanism provided by the UEM Server and UEM Android Client. This information includes details about the audit records which the TOE generates including details encompassing the required content. The tables in section entitled "Auditing events in BlackBerry UEM" and "Device audit information" of [Admin] provide a complete mapping of required audit events to the TOE generated events.

Testing Assurance Activities: The evaluator shall verify that when an application or update push fails, that the audit records generated match the format specified in the guidance and that the fields in each audit record have the proper entries.

The evaluator constructed a list of required audit events based on the TOE Security Target [ST] and FAU_GEN.1(2). The evaluator then either identified events generated when performing a corresponding function in another test or performed operations directly (e.g., log in and log out) to cause each required audit event to be generated. The evaluator captured screen shots and/or textual content that is included in the proprietary detailed test report.

2.1.5.2 MDMPP40:FAU_GEN.1.2(2)

TSS Assurance Activities: The evaluator shall check the TSS and ensure that it provides a format for audit records. Each audit record format type must be covered, along with a brief description of each field.

In the architecture of this product, the MAS server is a feature embedded within the UEM Server and is not a distinct architectural component. The audits for the MAS server were described in the table of auditable events found in Section 6.1 of [ST], in a manner identical to the audits for the UEM Server.

Guidance Assurance Activities: The evaluator shall check the administrative guide and ensure that it provides a format for audit records. Each audit record format type must be covered, along with a brief description of each field. The evaluator shall check to make sure that the description of the fields contains the information required in FAU_GEN.1.2(2).

The tables in section entitled "Auditing events in BlackBerry UEM" and "Device audit information" of [Admin] provide a complete mapping of required audit events to the TOE generated events.

Testing Assurance Activities: When verifying the test results from FAU_GEN.1.1(2), the evaluator shall ensure the audit records generated during testing match the format specified in the administrative guide, and that the fields in each audit record have the proper entries.

Note that the testing here can be accomplished in conjunction with the testing of the security mechanisms directly. For example, testing performed to ensure that the administrative guidance provided is correct verifies that AGD_OPE.1 is satisfied and should address the invocation of the administrative actions that are needed to verify the audit records are generated as expected.

While performing the FAU_GEN.1(2).1 tests, the evaluators collected and confirmed that the required audit record content was present in each audit record.

Component TSS Assurance Activities: None Defined

Component Guidance Assurance Activities: None Defined

Component Testing Assurance Activities: None Defined

2.1.6 Network Reachability Review (MDMPP40:FAU_NET_EXT.1)

2.1.6.1 MDMPP40:FAU_NET_EXT.1.1

TSS Assurance Activities: None Defined

Guidance Assurance Activities: None Defined

Testing Assurance Activities: None Defined

Component TSS Assurance Activities: The evaluator ensures that the TSS describes how reachability events are implemented, for each supported mobile platform. The evaluator verifies that this description clearly indicates who (MDM Agent or MDM Server) initiates reachability events.

Section 6.1 of [ST] indicates that the UEM Server provides the ability for an administrator to determine the connectivity status of any enrolled agent. The UEM server can display the last check-in time of all of its enrolled devices and the administrator can check terminology: initiate a check-in command to refresh that information if required.

Component Guidance Assurance Activities: The evaluator shall verify that the guidance instructs administrators on the method of determining the network connectivity status of an enrolled agent.

The section entitled "Finding the last time the device contacted BlackBerry UEM" in [Admin] provides instructions to determine "Last contact time" for a given device.

Component Testing Assurance Activities: For each MDM Agent/platform listed as supported in the ST: The evaluator shall configure the MDM Agent/platform to perform a network reachability test, both with and without such connectivity and shall ensure that by following the guidance, the evaluator can determine results that reflect both.

The evaluator configured network connectivity between the mobile devices (both Android and iOS) and the UEM Server and attempted to use the server to check-in with the MD Agent. The evaluator observed on the server that the device status update was successful. The evaluator then changed the network connectivity such that the mobile device cannot reach the UEM Server. With the network between the mobile device and the UEM Server disrupted, the evaluator attempted to use the agent to do a status update to the UEM Server. The evaluator observed on the server that the device status update was NOT successful. Finally, the evaluator re-established connectivity and observed that the status update occurred.

2.1.7 Audit Review (MDMPP40:FAU_SAR.1)

2.1.7.1 MDMPP40:FAU_SAR.1.1

TSS Assurance Activities: If 'invoke platform-provided functionality' is selected, the evaluator shall examine the TSS to verify that it describes (for each supported platform) how this functionality is invoked (it should be noted that this may be through a mechanism that is not implemented by the MDM Server; nonetheless, that mechanism will be identified in the TSS as part of this evaluation activity).

The section in this SFR in [ST] is both invoke platform-provided functionality and implement functionality. Section 6.1 of [ST] explains that the UEM Server collects, protects, and can display audit messages to authorized administrators. As the UEM Server generates audit events it stores them within its SQL database. Those events

are available to be viewed via the UEM Server administrator portal. Periodically, the audit events stored in the SQL database are also forwarded to a configured SYSLOG server where they can be viewed in the operational environment.

Note that audit events collected from enrolled mobile devices are received and immediately forwarded to a configured SYSLOG server where they can be examined. The UEM Server does not provide any interface to view these audit events and thus the functions from the environment are invoked to read audit data from the SYSLOG server.

Guidance Assurance Activities: None Defined

Testing Assurance Activities: None Defined

2.1.7.2 MDMPP40:FAU_SAR.1.2

TSS Assurance Activities: If 'invoke platform-provided functionality' is selected, the evaluator shall examine the TSS to verify that it describes (for each supported platform) how this functionality is invoked (it should be noted that this may be through a mechanism that is not implemented by the MDM Server; nonetheless, that mechanism will be identified in the TSS as part of this evaluation activity).

The section in this SFR in [ST] is both invoke platform-provided functionality and implement functionality. Section 6.1 of [ST] explains that the UEM Server collects, protects, and can display audit messages to authorized administrators. As the UEM Server generates audit events it stores them within its SQL database. Those events are available to be viewed via the UEM Server administrator portal. Periodically, the audit events stored in the SQL database are also forwarded to a configured SYSLOG server where they can be viewed in the operational environment.

Note that audit events collected from enrolled mobile devices are received and immediately forwarded to a configured SYSLOG server where they can be examined. The UEM Server does not provide any interface to view these audit events and thus the functions from the environment are invoked to read audit data from the SYSLOG server.

Guidance Assurance Activities: The evaluator shall check the AGD guidance and ensure that it describes how the administrator accesses the audit data and describes the format of the audit record.

The section entitled "Auditing" in [Admin] contains a description of the contents of audit records. Specifically, the table entitled "Audit record fields for server audits" describes important fields that can be found within records of the audits generated by the UEM Server while the table "Audit record fields for devices" describes important fields that can be found within records of the audits generated by the UEM Android client.

This section also indicates that BlackBerry UEM records all actions that administrators perform in the management console and displays them in the Audit screen.

Testing Assurance Activities: The evaluator shall attempt to view the audit record as the authorized administrator and verify that the action succeeds. The evaluator shall ensure the audit records generated during testing match the format specified in the administrative guide.

As part of FAU_GEN.1 testing, the evaluator observed that the audit records generated by the TOE matched the audit record format described by the AGD. The evaluator also showed that the UEM Server Web UI offered a way to view audit records.

Component TSS Assurance Activities: None Defined

Component Guidance Assurance Activities: None Defined

Component Testing Assurance Activities: None Defined

2.1.8 Security Audit Event Selection (MDMPP40:FAU_SEL.1)

2.1.8.1 MDMPP40:FAU_SEL.1.1

TSS Assurance Activities: None Defined

Guidance Assurance Activities: None Defined

Testing Assurance Activities: None Defined

Component TSS Assurance Activities: If 'invoke platform-provided functionality' is selected, the evaluator shall examine the TSS to verify that it describes (for each supported platform) how this functionality is invoked (it should be noted that this may be through a mechanism that is not implemented by the MDM Server; nonetheless, that mechanism will be identified in the TSS as part of this evaluation activity).

Not applicable, as 'invoke platform-provided functionality' is not selected.

Component Guidance Assurance Activities: The evaluator shall review the administrative guidance to ensure that the guidance itemizes all event types, as well as describes all attributes that are to be selectable in accordance with the requirement, to include those attributes listed in the assignment. The administrative guidance shall also contain instructions on how to set the preselection as well as explain the syntax (if present) for multi-value pre-selection. The administrative guidance shall also identify those audit records that are always recorded, regardless of the selection criteria currently being enforced.

The subsection entitled "Configure audit settings" in the section entitled "Auditing events in BlackBerry UEM" in the [Admin] describes the set of audit events which are selectable by an administrator. This section contains instructions to choose the event types to audit and the success/failure conditions to audit. Later in the following sections, are statements indicating the audits which are always collected. The section entitled, "TLS auditing" states that "All TLS related failures and audit records containing Event = TLS server connection error' or 'TLS client connection error' will be generated". Also, the section entitled "Deployer audit records" states that "Each time you run the Extractor.exe file an audit record is created."

Component Testing Assurance Activities: The evaluator shall also perform the following tests:

Test 1: For each attribute listed in the requirement, the evaluator shall devise a test to show that selecting the attribute causes only audit events with that attribute (or those that are always recorded, as identified in the administrative guidance) to be recorded.

Test 2: [conditional] If the TSF supports specification of more complex audit preselection criteria (e.g., multiple attributes, logical expressions using attributes) then the evaluator shall devise tests showing that this capability is correctly implemented. The evaluator shall also, in the test plan, provide a short narrative justifying the set of tests as representative and sufficient to exercise the capability.

Test 1: The selection in the requirement indicates that the TOE allows configuration of audit events based on event type as well as the success or failure of the security event. The evaluator configured a set of event types to be collected and a set to be omitted. Upon performing events matching each set, the evaluator observed that audit data was generated for the set of events to be collected, and not for the set to be omitted.

The sets were:

* Event type audited for success only (User Logged in / User Logged out)

  - Audit events of this type were found following successful activity that would cause the event.

* Event type not audited. (User Logged in)

  - No events of this type were found following any activity that would cause the event.

* Event type audited for both success and failure (Device enrollment started)

  - Audit events of this type were found following successful activity that would cause the event as well as following failed activity that would cause the event.

Test 2: The TOE does not provide complex combinations of pre-selection criteria beyond what was addressed by Test 1.

2.1.9 Security Audit Event Selection (MDMA10:FAU_SEL.1(2))

2.1.9.1 MDMA10:FAU_SEL.1.1(2)

TSS Assurance Activities: None Defined

Guidance Assurance Activities: None Defined

Testing Assurance Activities: None Defined

Component TSS Assurance Activities: If 'invoke platform-provided functionality' is selected, the evaluator shall examine the TSS of the ST to verify that it describes (for each supported platform) how this functionality is invoked (it should be noted that this may be through a mechanism that is not implemented by the MDM Agent; nonetheless, that mechanism will be identified in the TSS as part of this evaluation activity).

Section 6.1 of [ST] indicates that the UEM Android Client uses the mobile device audit storage and leverages the audit selection capabilities of the mobile device which can filter events based on event type, severity level, user identifier, and success/failure.

Component Guidance Assurance Activities: The evaluator shall examine the operational guidance to determine that it contains instructions on how to define the set of auditable events as well as explains the syntax for multi-value selection (if applicable). The evaluator shall also verify that the operational guidance shall identify those audit records that are always recorded, regardless of the selection criteria currently being enforced.

The section entitled "Device audit information" in [Admin] explains that IT policies can be used to filter the device events which are collected. This section includes a table describing each policy setting and option that can be controlled.

Component Testing Assurance Activities: Test 1: For each attribute listed in the requirement, the evaluator shall devise a test to show that selecting the attribute causes only audit events with that attribute (or those that are always recorded, as identified in the administrative guidance) to be recorded.

Test 2: [conditional]: If the TSF supports specification of more complex audit pre-selection criteria (e.g., multiple attributes, logical expressions using attributes) then the evaluator shall devise tests showing that this capability is correctly implemented. The evaluator shall also, in the test plan, provide a short narrative justifying the set of tests as representative and sufficient to exercise the capability.

Test 1: The evaluator selected specific "Audit Events" to be collected on devices and performed actions on tested devices to cause those events to be generated. The evaluator observed the configured audit records in the device's audit data. The evaluator then disabled the same "Audit Events" and performed the same actions on tested devices that would cause those events to be generated. The evaluator observed the configured audit records were not in the device's audit data.

Test 2: The TOE does not support complex combinations of pre-selection criteria.

2.1.10 External Trail Storage (MDMPP40:FAU_STG_EXT.1)

2.1.10.1 MDMPP40:FAU_STG_EXT.1.1

TSS Assurance Activities: None Defined

Guidance Assurance Activities: None Defined

Testing Assurance Activities: None Defined

Component TSS Assurance Activities: The evaluator shall examine the TSS to ensure it describes the means by which the audit data are transferred to the external audit server, and how the trusted channel is provided.

Section 6.1 of [ST] indicates that the UEM Server stores audit data in the same SQL database where most of its configuration data is stored. As such, the UEM Server depends on SQL access restrictions to protect audit data. This audit information can be periodically exported to a secure SYSLOG server using TLS.

Component Guidance Assurance Activities: The evaluator shall also examine the operational guidance to determine that it describes the relationship between the local audit data and the audit data that are sent to the audit log server. For example, when an audit event is generated, is it simultaneously sent to the external server and the local store, or is the local store used as a buffer and 'cleared' periodically by sending the data to the audit server.

The evaluator shall also examine the operational guidance to ensure it describes how to establish the trusted channel to the audit server, as well as describe any requirements on the audit server (particular audit server protocol, version of the protocol required, etc.), as well as configuration of the TOE needed to communicate with the audit server.

The section entitled "Storing audit logs" in [Admin] explains that when an audit event is generated, it is stored in the BlackBerry UEM database. Every 15 minutes the UEM server sends these saved audit records to the audit server. The sections entitled "Set up export of server audit records to a syslog server" and "Set up export of device audit records to a syslog server" describe how to configure the UEM server to securely transmit audit records to an external syslog server. Sections entitled "Set up export of server audit records to a syslog server" indicates the TLS version, ciphersuites and x509 certificate constraints which the TOE requires the syslog server support.

Component Testing Assurance Activities: Testing of the trusted channel mechanism will be performed as specified in the associated evaluation activities for the particular trusted channel mechanism. The evaluator shall perform the following test for this requirement:

Test 1: The evaluator shall establish a session between the TOE and the audit server according to the configuration guidance provided. The evaluator shall then examine the traffic that passes between the audit server and the TOE

during several activities of the evaluator's choice designed to generate audit data to be transferred to the audit server. The evaluator shall observe that these data are not able to be viewed in the clear during this transfer, and that they are successfully received by the audit server. The evaluator shall record the particular software (name, version) used on the audit server during testing.

The evaluator performed a TLS test that caused audit data to be generated on the UEM Server and transferred to the SYSLOG server. The evaluator captured the network traffic between the UEM server and the external SYSLOG server when the UEM Server was transferring audit data. Inspection of the network traffic showed that the traffic was protected by TLS and not sent in plaintext. The evaluator also examined audit data on the UEM Web UI and was able to locate the same audit events on the SYSLOG server.

2.1.11 Audit Event Storage (MDMPP40:FAU_STG_EXT.2)

2.1.11.1 MDMPP40:FAU_STG_EXT.2.1

TSS Assurance Activities: None Defined

Guidance Assurance Activities: None Defined

Testing Assurance Activities: None Defined

Component TSS Assurance Activities: If 'invoke platform-provided functionality' is selected, the evaluator shall examine the TSS to verify that it describes (for each supported platform) how the audit record protection functionality is invoked (it should be noted that this may be through a mechanism that is not implemented by the MDM Server; nonetheless, that mechanism will be identified in the TSS as part of this evaluation activity).

If 'implement functionality' is selected, the evaluator shall ensure that the TSS describes how the audit records are protected from unauthorized modification or deletion. The evaluator shall ensure that the TOE uses audit trail specific protection mechanisms.

Section 6.1 of [ST] indicates that the UEM Server stores audit data in the same SQL database where most of its configuration data is stored. As such, the UEM Server depends on SQL access restrictions to protect audit data. This audit information can be periodically exported to a secure SYSLOG server using TLS.

Component Guidance Assurance Activities: None Defined

Component Testing Assurance Activities: The evaluator shall perform the following tests:

Test 1: The evaluator shall access the audit trail as an unauthorized user and attempt to modify and delete the audit records. The evaluator shall verify that these attempts fail.

Test 2: The evaluator shall access the audit trail as an authorized user and attempt to modify and delete the audit records. The evaluator shall verify that these attempts succeed. The evaluator shall verify that only the records intended for modification and deletion are modified and deleted.

Section 6.1 of [ST] explains that only authorized administrators can log into the UEM Server and no TOE interface to audit records exist outside of the UEM Web UI and the underlying server platform. With these restrictions in mind, the evaluator inspected the UEM Web UI and found it is not possible for an unauthorized user to modify or delete audit records through TOE interfaces.

2.2 Cryptographic support (FCS)

2.2.1 Cryptographic Key Generation (MDMPP40:FCS_CKM.1)

2.2.1.1 MDMPP40:FCS_CKM.1.1

TSS Assurance Activities: None Defined

Guidance Assurance Activities: None Defined

Testing Assurance Activities: None Defined

Component TSS Assurance Activities: If 'invoke platform-provided functionality' is selected:

The evaluator shall examine the TSS to verify that it describes (for each supported platform) how the key generation functionality is invoked (it should be noted that this may be through a mechanism that is not implemented by the MDM Server; nonetheless, that mechanism will be identified in the TSS as part of this evaluation activity).

If 'implement functionality' is selected:

The evaluator shall ensure that the TSS identifies the key sizes supported by the TOE. If the ST specifies more than one scheme, the evaluator shall examine the TSS to verify that it identifies the usage for each scheme.

Section 6.2 of [ST] indicates that the UEM Server uses its Certicom Security Builder GSE-J Crypto Core (version 2.9.2) to generate asymmetric RSA keys for authentication. The UEM Server issues its own RSA 2048-bit, 3072-bit and 4096-bit certificates and stores them into the UEM Server SQL database.

The UEM Android Client relies upon the UEM Server for issuance of its RSA certificate. During the UEM Android Client's enrollment process, the UEM Android Client uses its evaluated mobile device platform to generate a RSA 2048-bit keypair and it sends a CSR request to the UEM Server resulting in an issued certificate.

Both UEM Server and UEM Android client provide ECDSA key generation using P-256, P-384 and P-521 curves in support of ECDHE key establishment.

Component Guidance Assurance Activities: The evaluator shall verify that the AGD guidance instructs the administrator how to configure the TOE to use the selected key generation schemes and key sizes for all uses defined in this PP.

Section "Validate and install the BlackBerry UEM software" of [Admin] provides instructions to install the system using a method that will ensure that the TOE is using TLS and certificates in a manner that confirms to the description in [ST]. That is, no cryptographic configuration of key generation, key establishment, encryption, signature algorithms or hashing (e.g., choosing ciphersuites) can be performed on the UEM Server following installation.

Component Testing Assurance Activities: Key Generation for FIPS PUB 186-4 RSA Schemes:

Since this evaluation activity is satisfied by the alternate FIPS CAVP program, the evaluation activity has not been reproduced here, but it can be found in the source Protection Profile.

Key Generation for Elliptic Curve Cryptography (ECC): FIPS 186-4 ECC Key Generation Test:

For each supported NIST curve, i.e., P-256, P-384 and P-521, the evaluator shall require the implementation under test (IUT) to generate 10 private/public key pairs. The private key shall be generated using an approved random bit generator (RBG). To determine correctness, the evaluator shall submit the generated key pairs to the public key verification (PKV) function of a known good implementation.

Key Generation for Elliptic Curve Cryptography (ECC): FIPS 186-4 Public Key Verification (PKV) Test:

For each supported NIST curve, i.e., P-256, P-384 and P-521, the evaluator shall generate 10 private/public key pairs using the key generation function of a known good implementation and modify five of the public key values so that they are incorrect, leaving five values unchanged (i.e., correct). The evaluator shall obtain in response a set of 10 PASS/FAIL values.

Key Generation for Finite-Field Cryptography (FFC):

Since this evaluation activity is satisfied by the alternate FIPS CAVP program, the evaluation activity has not been reproduced here, but it can be found in the source Protection Profile.

Diffie-Hellman Group 14 and FFC Schemes using 'safe-prime' groups:

Testing for FFC Schemes using Diffie-Hellman group 14 and/or 'safe-prime' groups is done as part of testing in FCS_CKM.2.1.

The TOE has been CAVP tested. Refer to the section entitled, "CAVP Certificate Justification" earlier in this document and to the TSS assurance activity for this requirement above where relevant CAVP certificates are identified.

The TOE's RSA key exchange mechanism is used in the TLS handshake process and during both product development and evaluation testing, the TOE's implementation undergoes testing to ensure TLS compatibility. Any defect in the RSA key exchange mechanism would result in an inability to negotiate TLS_RSA_* ciphersuites with a separate, known good implementation. Gossamer's TLSS & TLSC testing results demonstrate the TOE conforms to the specifications.

2.2.2 Cryptographic Key Establishment (MDMPP40:FCS_CKM.2)

2.2.2.1 MDMPP40:FCS_CKM.2.1

TSS Assurance Activities: None Defined

Guidance Assurance Activities: None Defined

Testing Assurance Activities: None Defined

Component TSS Assurance Activities: If 'invoke platform-provided functionality' is selected:

The evaluator shall examine the TSS to verify that it describes (for each supported platform) how the key establishment functionality is invoked (it should be noted that this may be through a mechanism that is not implemented by the MDM Server; nonetheless, that mechanism will be identified in the TSS as part of this evaluation activity).

If 'implement functionality' is selected:

The evaluator shall ensure that the supported key establishment schemes correspond to the key generation schemes identified in FCS_CKM.1.1. If the ST specifies more than one scheme, the evaluator shall examine the TSS to verify that it identifies the usage for each scheme.

The evaluator shall ensure that the TSS describes how the TOE handles decryption errors. In accordance with NIST Special Publication 800-56B, the TOE must not reveal the particular error that occurred, either through the contents of any outputted or logged error message or through timing variations. If KTS-OAEP is supported, the evaluator shall create separate contrived ciphertext values that trigger each of the three decryption error checks described in NIST Special Publication 800-56B section 7.2.2.3, ensure that each decryption attempt results in an error, and ensure that any outputted or logged error message is identical for each. If KTS-KEM-KWS is supported, the evaluator shall create separate contrived ciphertext values that trigger each of the three decryption error

checks described in NIST Special Publication 800-56B section 7.2.3.3, ensure that each decryption attempt results in an error, and ensure that any outputted or logged error message is identical for each.

If Diffie-Hellman group 14 is selected from FCS_CKM.2.1, the TSS shall describe how the implementation meets RFC 3526 Section 3.

Section 6.2 of [ST] indicates that the UEM Server uses its Certicom Security Builder GSE-J Crypto Core (version 2.9.2) to generate asymmetric RSA keys for key establishment. The UEM Server issues its own RSA 2048-bit certificates and stores them into the UEM Server SQL database.

The UEM Android Client relies upon the UEM Server for issuance of its RSA certificate. During the UEM Android Client's enrollment process, the UEM Android Client uses its evaluated mobile device platform to generate an RSA 2048-bit keypair and it sends a CSR request to the UEM Server resulting in an issued certificate.

Both UEM Server and UEM Android client provide ECDSA key generation using P-256, P-384 and P-521 curves in support of ECDHE key establishment.

This section of the ST also indicates that the TOE handles decryption errors in accordance with NIST Special Publication 800-56B. The TOE does not reveal the particular error that occurred, either through the contents of any outputted or logged error message or through timing variations.

Component Guidance Assurance Activities: The evaluator shall verify that the AGD guidance instructs the administrator how to configure the TOE to use the selected key establishment scheme(s).

Section "Validate and install the BlackBerry UEM software" of [Admin] provides instructions to install the system using a method that will ensure that the TOE is using TLS and certificates in a manner that confirms to the description in [ST]. That is, no cryptographic configuration of key generation, key establishment, encryption, signature algorithms or hashing (e.g., choosing ciphersuites) can be performed on the UEM Server following installation.

Component Testing Assurance Activities: The evaluator shall verify the implementation of the key establishment schemes supported by the TOE using the applicable tests below.

SP800-56A Key Establishment Schemes:

Since this evaluation activity is satisfied by the alternate FIPS CAVP program, the evaluation activity has not been reproduced here, but it can be found in the source Protection Profile.

RSA-based key establishment:

The evaluator shall verify the correctness of the TSF's implementation of RSAESPKCS1-v1_5 by using a known good implementation for each protocol selected in FTP_TRP.1(1), FTP_TRP.1(2), FTP_TRP.1(3), FTP_ITC.1(1), FTP_ITC.1(2), FPT_ITT.1(1), and FPT_ITT.1(2) that uses RSAES-PKCS1-v1_5.

Diffie-Hellman Group 14:

The evaluator shall verify the correctness of the TSF's implementation of Diffie-Hellman group 14 by using a known good implementation for each protocol selected in FTP_TRP.1(1), FTP_TRP.1(2), FTP_TRP.1(3), FTP_ITC.1(1), FTP_ITC.1(2), FPT_ITT.1(1), and FPT_ITT.1(2) that uses Diffie-Hellman Group 14.

FFC Schemes using 'safe-prime' groups:

The evaluator shall verify the correctness of the TSF's implementation of 'safe-prime' groups by using a known good implementation for each protocol selected in FTP_TRP.1(1), FTP_TRP.1(2), FTP_TRP.1(3), FTP_ITC.1(1), FTP_ITC.1(2), FPT_ITT.1(1), and FPT_ITT.1(2) that uses 'safe-prime' groups. This test must be performed for each 'safeprime' group that each protocol uses.

ECC - The TOE has been CAVP tested for ECC. Refer to the section entitled, "CAVP Certificate Justification" earlier in this document and to the TSS assurance activity for this requirement above where relevant CAVP certificates are identified.

RSA - The TOE's RSA implementation was verified using a known good implementation. The TOE's RSA key exchange mechanism is used in the TLS handshake process and during both product development and evaluation testing, the TOE's implementation undergoes testing to ensure TLS compatibility. Any defect in the RSA key exchange mechanism would result in an inability to negotiate TLS_RSA_* ciphersuites with a separate, known good implementation. Gossamer's TLSS & TLSC testing results demonstrate the TOE conforms to the specifications.

2.2.3 Cryptographic Key Destruction (MDMPP40:FCS_CKM_EXT.4)

2.2.3.1 MDMPP40:FCS_CKM_EXT.4.1

TSS Assurance Activities: If 'invoking platform-provided functionality' is selected, the evaluator shall examine the TSS to verify that it describes (for each supported platform) how the key destruction functionality is invoked (it should be noted that this may be through a mechanism that is not implemented by the MDM Server; nonetheless, that mechanism will be identified in the TSS as part of this evaluation activity).

Section 6.2 of [ST] states that the UEM Server stores certificates (the only persistently stored keying material) in its SQL database in encrypted format.

This section also states that the UEM Sever will directly overwrite the old keys with the new in the encrypted key store in the SQL database.

Section 6.2 of [ST] states that the UEM Android Client relies upon its evaluated platform to securely clear keys (TLS and HTTPS session keys) from memory when no longer needed as the UEM Android Client utilizes platform provided TLS and key storage.

Guidance Assurance Activities: None Defined

Testing Assurance Activities: None Defined

2.2.3.2 MDMPP40:FCS_CKM_EXT.4.2

TSS Assurance Activities: The evaluation activity used is dependent on the selection made in FCS_CKM_EXT.4.1. FCS_COP.1.1(1)

The evaluator shall check to ensure the TSS lists each type of plaintext key material and CSP (authentication data, authorization data, secret/private symmetric keys, data used to derive keys, etc.) and its origin and storage location.

The evaluator shall verify that the TSS describes when each type of key material and CSP is no longer needed.

If 'invoke platform-provided functionality' is selected:

The evaluator shall examine the TSS to verify that it describes (for each supported platform) how the key releasing functionality is invoked (it should be noted that this may be through a mechanism that is not implemented by the MDM Server; nonetheless, that mechanism will be identified in the TSS as part of this evaluation activity).

If 'implement functionality' is selected:

The evaluator shall also verify that, for each type, the type of clearing procedure that is performed is listed. If different types of memory are used to store the materials to be protected, the evaluator shall check to ensure that the TSS describes the clearing procedure in terms of the memory in which the data are stored (for example, 'secret keys stored on flash are cleared by overwriting once with zeros, while secret keys stored on the internal persistent storage device are cleared by overwriting one time with a random pattern that is changed before each write'). For block erases, the evaluator shall also ensure that the block erase command used is listed and shall verify that the command used also addresses any copies of the plaintext key material that may be created in order to optimize the use of flash memory.

Section 6.2 of [ST] states that the UEM Server stores certificates (the only persistently stored keying material) in its SQL database in encrypted format. This section also explains that the UEM Android client relies upon its evaluated platform to securely store keys for HTTPS, TLS and private certificates. Therefore, neither the UEM Server nor UEM Android client store any plaintext keying material.

Guidance Assurance Activities: None Defined

Testing Assurance Activities: For each software and firmware key clearing situation the evaluator shall repeat the following tests. Note that at this time hardware-bound keys are explicitly excluded from testing.

Test 1: The evaluator shall utilize appropriate combinations of specialized operational environment and development tools (debuggers, simulators, etc.) for the TOE and instrumented TOE builds to test that keys are cleared correctly, including all intermediate copies of the key that may have been created internally by the TOE during normal cryptographic processing with that key.

Cryptographic TOE implementations in software shall be loaded and exercised under a debugger to perform such tests. The evaluator shall perform the following test for each key subject to clearing, including intermediate copies of keys that are persisted encrypted by the TOE:

1. Load the instrumented TOE build in a debugger.

2. Record the value of the key in the TOE subject to clearing.

3. Cause the TOE to perform a normal cryptographic processing with the key from #1.

4. Cause the TOE to clear the key.

5. Cause the TOE to stop the execution but not exit.

6. Cause the TOE to dump the entire memory footprint of the TOE into a binary file.

7. Search the content of the binary file created in #4 for instances of the known key value from #1.

The test succeeds if no copies of the key from #1 are found in step #7 above and fails otherwise. The evaluator shall perform this test on all keys, including those persisted in encrypted form, to ensure intermediate copies are cleared.

Test 2: In cases where the TOE is implemented in firmware and operates in a limited operating environment that does not allow the use of debuggers, the evaluator shall utilize a simulator for the TOE on a general purpose operating system. The evaluator shall provide a rationale explaining the instrumentation of the simulated test environment and justifying the obtained test results.

The evaluator installed the UEM Server software application into a Hyper-V VM. Using a test system with a customized versions of OpenSSL and stunnel (patched to log TLS Session keys) as the TLS peer, the evaluator established TLS connections with the TOE on the Syslog, LDAP, Web UI and Agent communication interfaces. After all sessions had been terminated, the evaluator saved the memory image of the running VM using Hyper-V operations into a file. The evaluator then searched the memory dump file for each TLS session key logged on the evaluator's test system. The evaluator did not find any of the keys associated with these 4 TLS channels in the memory dump file.

Component TSS Assurance Activities: None Defined

Component Guidance Assurance Activities: None Defined

Component Testing Assurance Activities: None Defined

2.2.4 Cryptographic Operation (Confidentiality Algorithms) (MDMPP40:FCS_COP.1(1))

2.2.4.1 MDMPP40:FCS_COP.1.1(1)

TSS Assurance Activities: None Defined

Guidance Assurance Activities: None Defined

Testing Assurance Activities: None Defined

Component TSS Assurance Activities: If 'invoke platform-provided functionality' is selected:

The evaluator shall examine the TSS to verify that it describes (for each supported platform) how the encryption/decryption functionality is invoked for each mode and key size selected in the MDM Server's ST (it should be noted that this may be through a mechanism that is not implemented by the MDM Server; nonetheless, that mechanism will be identified in the TSS as part of this evaluation activity).

Section 6.2 of [ST] indicates that the UEM Android Client relies upon its evaluated platform for AES encryption and decryption while the UEM Server uses the Certicom library for its AES CBC and GCM operations.

Component Guidance Assurance Activities: None Defined

Component Testing Assurance Activities: Since this evaluation activity is satisfied by the alternate FIPS CAVP program, the evaluation activity has not been reproduced here, but it can be found in the source Protection Profile.

The TOE has been CAVP tested. Refer to the section entitled, "CAVP Certificate Justification" earlier in this document and to the TSS assurance activity for this requirement above where relevant CAVP certificates are identified.

2.2.5 Cryptographic Operation (Hashing Algorithms) (MDMPP40:FCS_COP.1(2))

2.2.5.1 MDMPP40:FCS_COP.1.1(2)

TSS Assurance Activities: None Defined

Guidance Assurance Activities: None Defined

Testing Assurance Activities: None Defined

Component TSS Assurance Activities: If 'invoke platform-provided functionality' is selected:

The evaluator shall examine the TSS to verify that it describes (for each supported platform) how the hash functionality is invoked for each digest size selected in the MDM Server's ST (it should be noted that this may be through a mechanism that is not implemented by the MDM Server; nonetheless, that mechanism will be identified in the TSS as part of this evaluation activity).

If 'implement functionality' is selected:

The evaluator shall check that the association of the hash function with other TSF cryptographic functions (for example, the digital signature verification function) is documented in the TSS. The evaluator checks the AGD documents to determine that any configuration that is required to be done to configure the functionality for the required hash sizes is present. The TSF hashing functions can be implemented in one of two modes. The first mode is the byte-oriented mode. In this mode the TSF only hashes messages that are an integral number of bytes in length; i.e., the length (in bits) of the message to be hashed is divisible by 8. The second mode is the bit-oriented mode. In this mode the TSF hashes messages of arbitrary length. As there are different tests for each mode, an indication is given in the following sections for the bit-oriented vs. the byte-oriented testmacs.

Section 6.2 of [ST] indicates that the UEM Server uses its Certicom Security Builder GSE-J Crypto Core version 2.9.2 to provide SHA-256, SHA-384 and SHA-512 hashing functionality. This section also indicates that the UEM Android Client relies upon its evaluated platform for SHA-256, SHA-384 and SHA-512 hashing functionality.

Component Guidance Assurance Activities: None Defined

Component Testing Assurance Activities: Since this evaluation activity is satisfied by the alternate FIPS CAVP program, the evaluation activity has not been reproduced here, but it can be found in the source Protection Profile.

The TOE has been CAVP tested. Refer to the section entitled, "CAVP Certificate Justification" earlier in this document and to the TSS assurance activity for this requirement above where relevant CAVP certificates are identified.

2.2.6 Cryptographic Operation (Signature Algorithms) (MDMPP40:FCS_COP.1(3))

2.2.6.1 MDMPP40:FCS_COP.1.1(3)

TSS Assurance Activities: None Defined

Guidance Assurance Activities: None Defined

Testing Assurance Activities: None Defined

Component TSS Assurance Activities: If 'invoke platform-provided functionality' is selected:

The evaluator shall examine the TSS to verify that it describes (for each supported platform) how the digital signature functionality is invoked for each operation they are used for in the MDM Server (it should be noted that this may be through a mechanism that is not implemented by the MDM Server; nonetheless, that mechanism will be identified in the TSS as part of this evaluation activity).

Section 6.2 of [ST] indicates that the UEM Server uses its Certicom Security Builder GSE-J Crypto Core version 2.9.2 to generate and verify RSA and ECDSA signatures. This section also indicates that the UEM Android Client relies upon its evaluated platform for signature verification and generation. Both the UEM Server and UEM Android Client generate and verify RSA 2048-bit, 3072-bit, and 4096-bit signatures as well as ECDSA signatures using P-256, P-384 or P-521 curves.

Component Guidance Assurance Activities: None Defined

Component Testing Assurance Activities: Since this evaluation activity is satisfied by the alternate FIPS CAVP program, the evaluation activity has not been reproduced here, but it can be found in the source Protection Profile.

The TOE has been CAVP tested. Refer to the section entitled, "CAVP Certificate Justification" earlier in this document and to the TSS assurance activity for this requirement above where relevant CAVP certificates are identified.

2.2.7 Cryptographic Operation (Keyed-Hash Message Authentication) (MDMPP40:FCS_COP.1(4))

2.2.7.1 MDMPP40:FCS_COP.1.1(4)

TSS Assurance Activities: None Defined

Guidance Assurance Activities: None Defined

Testing Assurance Activities: None Defined

Component TSS Assurance Activities: If 'invoke platform-provided functionality' is selected:

The evaluator shall examine the TSS to verify that it describes (for each supported platform) how the keyed-hash functionality is invoked for each mode and key size selected in the MDM Server's ST (it should be noted that this

may be through a mechanism that is not implemented by the MDM Server; nonetheless, that mechanism will be identified in the TSS as part of this evaluation activity).

If 'implement functionality' is selected:

The evaluator shall examine the TSS to ensure that it specifies the following values used by the HMAC function: key length, hash function used, block size, and output MAC length used.

Section 6.2 of [ST] indicates that the UEM Server uses its Certicom Security Builder GSE-J Crypto Core version 2.9.2 to provide HMAC-SHA-256, HMAC-SHA-384 and HMAC-SHA-512 hashing functionality. This section also indicates that the UEM Android Client (which relies upon its evaluated platform) utilizes the platform's HMAC-SHA-256, HMAC-SHA-384 and HMAC-SHA-512 hashing functionality. When using HMAC as part of TLS, both the client and server utilize HMAC keys equal to the block size of the underlying hash algorithm. Thus, when employing HMAC-SHA-256, the TOE uses a 32-byte key to generate a 32-byte hash. Likewise, when employing HMAC-SHA-384, the TOE uses a 48-byte key when performing hashing to produce a 48-byte hash and when employing HMAC-SHA-512, the TOE uses a 48-byte key when performing hashing to produce a 64-byte hash.

Component Guidance Assurance Activities: None Defined

Component Testing Assurance Activities: Since this evaluation activity is satisfied by the alternate FIPS CAVP program, the evaluation activity has not been reproduced here, but it can be found in the source Protection Profile.

The TOE has been CAVP tested. Refer to the section entitled, "CAVP Certificate Justification" earlier in this document and to the TSS assurance activity for this requirement above where relevant CAVP certificates are identified.

2.2.8 HTTPS Protocol (MDMPP40:FCS_HTTPS_EXT.1)

2.2.8.1 MDMPP40:FCS_HTTPS_EXT.1.1

TSS Assurance Activities: None Defined

Guidance Assurance Activities: None Defined

Testing Assurance Activities: None Defined

2.2.8.2 MDMPP40:FCS_HTTPS_EXT.1.2

TSS Assurance Activities: None Defined

Guidance Assurance Activities: None Defined

Testing Assurance Activities: None Defined

Component TSS Assurance Activities: None Defined

Component Guidance Assurance Activities: None Defined

Component Testing Assurance Activities: Test 1: The evaluator shall attempt to establish an HTTPS connection with a web server, observe the traffic with a packet analyzer, and verify that the connection succeeds and that the traffic is identified as TLS or HTTPS.

Other tests are performed in conjunction with the TLS evaluation activities.

The UEM Server acts as a TLS server for remote administration. The packet captures from test case FTP_TRP.1.3(1)-t1 were analyzed and showed that traffic between a remote administrator and the UEM Server is protected by HTTPS/TLS, using TLS version 1.2. The packet captures also show that the data is not plaintext.

2.2.9 Initialization Vector Generation (MDMPP40:FCS_IV_EXT.1)

2.2.9.1 MDMPP40:FCS_IV_EXT.1.1

TSS Assurance Activities: None Defined

Guidance Assurance Activities: None Defined

Testing Assurance Activities: None Defined

Component TSS Assurance Activities: If 'invoke platform-provided functionality' is selected: The evaluator shall examine the TSS to verify that it describes (for each supported platform) how the IV generation is invoked for each mode selected in the MDM Server's ST (it should be noted that this may be through a mechanism that is not implemented by the MDM Server; nonetheless, that mechanism will be identified in the TSS as part of this evaluation activity).

If 'implement functionality' is selected: The evaluator shall examine the TSS to ensure that it details the encryption of user credentials, persistent secrets, and private keys and the generation of the IVs used for that encryption.

Section 6.2 of [ST] explains that the UEM Server generates IVs for AES CBC and AES GCM using unpredictable (random) IVs drawn from the SHA-256 Hash_DRBG (any), HMAC_DRBG (any), and CTR_DRBG (AES) (which meets the “unpredictable” requirement of SP 800-38A and SP800-38D). The UEM Server derives AES CBC IVs as part of the TLS handshake (which also meets the “unpredictable” and “non-repeating” requirements of SP 800-38A and SP800-38D).

The UEM Android Client invokes platform APIs for TLS connections where IVs are generated by its platform as required.

Component Guidance Assurance Activities: None Defined

Component Testing Assurance Activities: The evaluator shall ensure that the generation of IVs for each key encrypted by the

same KEK meets Table 4.

By reviewing the Security Target, the evaluator concluded that IVs are used as part of AES-CBC and AES-GCM used in support of the TLS handshake. These IVs are drawn from the SHA-256 Hash_DRBG (any), HMAC_DRBG (any), and CTR_DRBG (AES) (which meets the "unpredictable" requirement of SP 800-38A and SP800-38D). These DRBG meet FCS_RBG_EXT.1. The unpredictability provided by use of these DRBG satisfies SP 800-38A and 800-38D.

2.2.10 Extended: Random Bit Generation (MDMPP40:FCS_RBG_EXT.1)

2.2.10.1 MDMPP40:FCS_RBG_EXT.1.1

TSS Assurance Activities: If 'invoke platform-provided functionality' is selected:

The evaluator shall examine the TSS to verify that it describes (for each supported platform) how the RBG functionality is invoked for each operation they are used for in the MDM Server (it should be noted that this may be through a mechanism that is not implemented by the MDM Server; nonetheless, that mechanism will be identified in the TSS as part of this evaluation activity).

Section 6.2 of [ST] explains that the UEM Server's Certicom library provides a Hash_DRBG (any), HMAC_DRBG (any), and CTR_DRBG (AES) seeded by the underlying platform (Microsoft Windows Server). The Server seeds its DRBG using the Windows BCryptGenRandom() function. Because Microsoft's entropy implementation cannot be tested, an assumption of entropy is made regarding it (as required for any untestable third-party source), specifically that the output of BCryptGenRandom() contains at least 0.666 bits of entropy per bit of output. With at least 0.666 bits of entropy per bit of output, the Server appropriately seeds its DRBG with at least 256-bits of entropy.

Section 6.2 also explains that the UEM Android Client makes indirect use of the AES-256 CTR_DRBG belonging to its underlying platform for all random bit generation (indirectly using it when calling platform provided cryptographic APIs).

Guidance Assurance Activities: None Defined

Testing Assurance Activities: Since this evaluation activity is satisfied by the alternate FIPS CAVP program, the evaluation activity has not been reproduced here, but it can be found in the source Protection Profile.

The TOE has been CAVP tested. Refer to the section entitled, "CAVP Certificate Justification" earlier in this document and to the TSS assurance activity for this requirement above where relevant CAVP certificates are identified.

2.2.10.2 MDMPP40:FCS_RBG_EXT.1.2

TSS Assurance Activities: Documentation shall be produced-and the evaluator shall perform the activities-in accordance with Appendix D: Entropy Documentation and Assessment and the 'Clarification to the Entropy Documentation and Assessment Annex.'

In the future, specific statistical testing (in line with NIST SP 800-90B) will be required to verify the entropy estimates.

The Entropy description is provided in a separate (non-ST) document that has been delivered to CCEVS for approval. Note that the entropy analysis has been accepted by CCEVS/NSA.

Guidance Assurance Activities: None Defined

Testing Assurance Activities: None Defined

Component TSS Assurance Activities: None Defined

Component Guidance Assurance Activities: None Defined

Component Testing Assurance Activities: None Defined

2.2.11 Cryptographic Key Storage (MDMPP40:FCS_STG_EXT.1)

2.2.11.1 MDMPP40:FCS_STG_EXT.1.1

TSS Assurance Activities: None Defined

Guidance Assurance Activities: None Defined

Testing Assurance Activities: None Defined

Component TSS Assurance Activities: Regardless of whether this requirement is met by the TSF or the TOE platform, the evaluator will check the TSS to ensure that it lists each persistent secret (credential, secret key) and private key needed to meet the requirements in the ST. For each of these items, the evaluator will confirm that the TSS lists for what purpose it is used, and how it is stored. The evaluator then performs the following actions.

Persistent secrets and private keys manipulated by the TOE platform:

The evaluator shall examine the TSS to verify that it describes (for each supported platform) how the key storage functionality is invoked for each persistent secret and private key described in the TSS (it should be noted that this may be through a mechanism that is not implemented by the MDM Server; nonetheless, that mechanism will be identified in the TSS as part of this evaluation activity).

Persistent secrets and private keys manipulated by the TSF:

The evaluator reviews the TSS to determine that it makes a case that, for each item listed as being manipulated by the TOE, it is not written unencrypted to persistent memory, and that the item is stored by the platform.

Section 6.2 of [ST] indicates that the UEM Server uses Certificates to provide an internal PKI which supports the issuance of certificates for its own use (i.e., to identify itself to other parties) and to be sent to mobile devices during enrollment. This section also indicates that for the private keys of certificates used by the TOE the UEM Server encrypts these persistent keys by storing them encrypted in its SQL database with PBEWithHmacSHA256AndAES256 (AES-CBC mode) using a 256-bit DEK created during installation. At no time does the Server store any plaintext keys on its hard drive or its SQL database. The Server does not store any ephemeral keys (e.g., TLS/HTTPS session keys).

Section 6.2 of [ST] also explains that The UEM Android Client stores keys for the certificates it receives from the UEM Server in the platform provided key storage (Android KeyStore) and then utilizes those keys securely through the platform provided key storage API.

Component Guidance Assurance Activities: None Defined

Component Testing Assurance Activities: None Defined

2.2.12 Cryptographic Key Storage (MDMA10:FCS_STG_EXT.1(2))

2.2.12.1 MDMA10:FCS_STG_EXT.1.1(2)

TSS Assurance Activities: None Defined

Guidance Assurance Activities: None Defined

Testing Assurance Activities: None Defined

Component TSS Assurance Activities: The evaluator will verify that the TSS lists each persistent secret (credential, secret key) and private key needed to meet the requirements in the ST. For each of these items, the evaluator will confirm that the TSS lists for what purpose it is used, and, for each platform listed as supported in the ST, how it is stored. The evaluator shall verify that the Agent calls a platform-provided API to store persistent secrets and private keys.

Section 6.2 of [ST] explains that the UEM Android Client stores keys for the certificates it receives from the UEM Server in the platform provided key storage (Android KeyStore) and then utilizes those keys securely through the platform provided key storage API. This certificate is used to identify and authenticate the mobile device to the UEM server.

Component Guidance Assurance Activities: None Defined

Component Testing Assurance Activities: None Defined

2.2.13 Encrypted Cryptographic Key Storage (MDMPP40:FCS_STG_EXT.2)

2.2.13.1 MDMPP40:FCS_STG_EXT.2.1

TSS Assurance Activities: None Defined

Guidance Assurance Activities: None Defined

Testing Assurance Activities: None Defined

Component TSS Assurance Activities: The evaluator shall examine the TSS to ensure it describes in detail how user credentials, persistent secret and private keys are stored and encrypted. The evaluator shall review the TSS to determine that it makes a case that key material is not written unencrypted to persistent memory and that it identifies the mode of encryption.

If 'invoke platform-provided functionality' is selected, the evaluator shall examine the TSS to verify that it describes (for each supported platform) how the key encryption functionality is invoked (it should be noted that this may be through a mechanism that is not implemented by the MDM Server; nonetheless, that mechanism will be identified in the TSS as part of this evaluation activity).

Section 6.2 of [ST] explains that the UEM server stores the private keys for its certificate as encrypted values in its SQL database. This section explains that the UEM server uses PBEWithHmacSHA256AndAES256 (AES-CBC mode) using a 256-bit DEK created during installation. This DEK is stored and protected by the Windows key store.

Furthermore, this section explains that at no time does the Server store any plaintext keys on its hard drive or its SQL database.

Component Guidance Assurance Activities: None Defined

Component Testing Assurance Activities: None Defined

2.2.14 TLS Protocol (PKGTLS11:FCS_TLS_EXT.1)

2.2.14.1 PKGTLS11:FCS_TLS_EXT.1.1

TSS Assurance Activities: None Defined

Guidance Assurance Activities: The evaluator shall ensure that the selections indicated in the ST are consistent with selections in the dependent components.

FCS_TLS_EXT.1.1 indicates that the TOE implements TLS client and TLS server-side functionality. The evaluator verified that the ST also included the necessary FCS_TLSC_EXT and FCS_TLSS_EXT requirements to support all TLS uses as required by FPT_ITT.1(2), FTP_ITC.1(1), FTP_ITC.1(2), FTP_TRP.1(1) and FTP_TRP.1(2).

Testing Assurance Activities: None Defined

Component TSS Assurance Activities: None Defined

Component Guidance Assurance Activities: None Defined

Component Testing Assurance Activities: None Defined

2.2.15 TLS Client Protocol (PKGTLS11:FCS_TLSC_EXT.1)

2.2.15.1 PKGTLS11:FCS_TLSC_EXT.1.1

TSS Assurance Activities: The evaluator shall check the description of the implementation of this protocol in the TSS to ensure that the cipher suites supported are specified. The evaluator shall check the TSS to ensure that the cipher suites specified include those listed for this component.

Section 6.2 of [ST] indicates that the TOE supports TLSv1.2 only, using the following ciphersuites when it is acting as a TLS client. The UEM server is a TLS client to an external audit server (syslog) and to an external LDAP server.

Ciphersuites used by UEM Server:

TLS_RSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5246,

TLS_RSA_WITH_AES_256_CBC_SHA256 as defined in RFC 5246,

TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5289,

TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384 as defined in RFC 5289.

Guidance Assurance Activities: The evaluator shall also check the operational guidance to ensure that it contains instructions on configuring the product so that TLS conforms to the description in the TSS.

The sections entitled "Set up export of server audit records to a syslog server" and "Set up export of device audit records to a syslog server" describe the steps to load the trusted root for a syslog server. These sections include instructions to update and execute the configurations scripts from Appendix A and B. Appendix A and B contain additional configuration scripts that are used during TOE installation and TOE updates. This script contains values which cause the TOE to use TLS and to provide mutual authentication to external syslog services as requested. These scripts also indicate the reference identifier port being used by the syslog server.

Section "Validate and install the BlackBerry UEM software" of [Admin] provides instructions to install the system using a method that will ensure that the TOE is using TLS and certificates in a manner that confirms to the description in [ST]. That is, no cryptographic configuration of key generation, key establishment, encryption, signature algorithms or hashing (e.g., choosing ciphersuites) can be performed on the UEM Server following installation.

Testing Assurance Activities: The evaluator shall also perform the following tests:

Test 1: The evaluator shall establish a TLS connection using each of the cipher suites specified by the requirement. This connection may be established as part of the establishment of a higher-level protocol, e.g., as part of an EAP session. It is sufficient to observe the successful negotiation of a cipher suite to satisfy the intent of the test; it is not necessary to examine the characteristics of the encrypted traffic in an attempt to discern the cipher suite being used (for example, that the cryptographic algorithm is 128-bit AES and not 256-bit AES).

Test 2: The goal of the following test is to verify that the TOE accepts only certificates with appropriate values in the extendedKeyUsage extension, and implicitly that the TOE correctly parses the extendedKeyUsage extension as part of X.509v3 server certificate validation. The evaluator shall attempt to establish the connection using a server with a server certificate that contains the Server Authentication purpose in the extendedKeyUsage extension and verify that a connection is established. The evaluator shall repeat this test using a different, but otherwise valid and trusted, certificate that lacks the Server Authentication purpose in the extendedKeyUsage extension and ensure that a connection is not established. Ideally, the two certificates should be similar in structure, the types of identifiers used, and the chain of trust.

Test 3: The evaluator shall send a server certificate in the TLS connection that does not match the server-selected cipher suite (for example, send a ECDSA certificate while using the TLS_RSA_WITH_AES_128_CBC_SHA cipher suite or send a RSA certificate while using one of the ECDSA cipher suites.) The evaluator shall verify that the product disconnects after receiving the server's Certificate handshake message.

Test 4: The evaluator shall configure the server to select the TLS_NULL_WITH_NULL_NULL cipher suite and verify that the client denies the connection.

Test 5: The evaluator shall perform the following modifications to the traffic:

Test 5.1: Change the TLS version selected by the server in the Server Hello to an undefined TLS version (for example 1.5 represented by the two bytes 03 06) and verify that the client rejects the connection.

Test 5.2: Change the TLS version selected by the server in the Server Hello to the most recent unsupported TLS version (for example 1.1 represented by the two bytes 03 02) and verify that the client rejects the connection.

Test 5.3: [conditional] If DHE or ECDHE cipher suites are supported, modify at least one byte in the server's nonce in the Server Hello handshake message, and verify that the client does not complete the handshake and no application data flows.

Test 5.4: Modify the server's selected cipher suite in the Server Hello handshake message to be a cipher suite not presented in the Client Hello handshake message. The evaluator shall verify that the client does not complete the handshake and no application data flows.

Test 5.5: [conditional] If DHE or ECDHE cipher suites are supported, modify the signature block in the server's Key Exchange handshake message, and verify that the client does not complete the handshake and no application data flows. This test does not apply to cipher suites using RSA key exchange. If a TOE only supports RSA key exchange in conjunction with TLS, then this test shall be omitted.

Test 5.6: Modify a byte in the Server Finished handshake message, and verify that the client does not complete the handshake and no application data flows.

Test 5.7: Send a message consisting of random bytes from the server after the server has issued the Change Cipher Spec message and verify that the client does not complete the handshake and no application data flows. The message must still have a valid 5-byte record header in order to ensure the message will be parsed as TLS.

The evaluator performed the following tests against the UEM Server's TLS client interface (including the syslog and LDAP connections).

Test 1: The evaluator established a TLS session on each trusted server using the UEM Server's client implementations for each of the claimed ciphersuites in turn. The evaluator used a network sniffer to capture the TLS session negotiation and observed that the expected TLS cipher is negotiated.

Test 2: The evaluator configured the established a TLS session with the UEM server's client implementations for LDAP and Syslog. The evaluator configured the server to send a certificate with the Server Authentication purpose in the extendedKeyUsage field. Using a network sniffer the evaluator captured the TLS session negotiation and observed that the TLS session was successfully negotiated. The evaluator reconfigured the test server to retry the TLS session using a certificate that is missing the Server Authentication purpose in the extendedKeyUsage field. Using a network sniffer the evaluator captured the TLS session negotiation and observed that the TLS session is not successfully negotiated.

Test 3: The evaluator established a TLS session from the UEM Server's client implementations. A modified test server negotiates an RSA ciphersuite, but returns an ECDSA Certificate. Using a network sniffer to capture the TLS session negotiation and observed that the TLS session is not negotiated successfully.

Test 4: The evaluator configured a test server to accept only the TLS_NULL_WITH_NUL_NULL ciphersuite. The evaluator then attempted to establish a TLS session from the UEM server's client implementations for LDAP and Syslog to that test server. Using a network sniffer the evaluator captured the TLS session negotiation and observed that the TLS session is not successfully negotiated.

Test 5: The evaluator obtained a packet captures of the TLS session negotiation between the UEM server's client implementations for LDAP and Syslog and a test server with Mutual Authentication configured on the test server. The evaluator made connection attempts from the client to the test server. The server implementation of the TLS protocol was modified as stated in the 7 scenarios described by the Assurance Activity. The evaluator inspected each packet captures to ensure that the connections are rejected for each scenario.

2.2.15.2 PKGTLS11:FCS_TLSC_EXT.1.2

TSS Assurance Activities: The evaluator shall ensure that the TSS describes the client's method of establishing all reference identifiers from the application-configured reference identifier, including which types of reference identifiers are supported (e.g. Common Name, DNS Name, URI Name, Service Name, or other application-specific Subject Alternative Names) and whether IP addresses and wildcards are supported. The evaluator shall ensure that this description identifies whether and the manner in which certificate pinning is supported or used by the product.

Section 6.2 of [ST] explains that the UEM Server performs certificate checking in conformance with FIA_X509_EXT.1 and accepts the use of wildcards in the SAN or CN. This section also states that the UEM server performs hostname checking to ensure that the expected hostname matches the certificate Common Name or Subject Alternate Name (when the UEM Server validates the certificate from an LDAP or Syslog server). Section 6.2 also explains that neither the UEM Server nor the UEM Android Client utilize certificate pinning.

Guidance Assurance Activities: The evaluator shall verify that the AGD guidance includes instructions for setting the reference identifier to be used for the purposes of certificate validation in TLS.

The sections entitled "Set up export of server audit records to a syslog server" and "Set up export of device audit records to a syslog server" describe the steps to load the trusted root for a syslog server. These sections include instructions to update and execute the configurations scripts from Appendix A and B. Appendix A and B contain additional configuration scripts that are used during TOE installation and TOE updates. These scripts contain values which cause the TOE to use TLS and to provide mutual authentication to external syslog services as requested. These scripts also indicate the reference identifier port being used by the syslog server.

Testing Assurance Activities: The evaluator shall configure the reference identifier according to the AGD guidance and perform the following tests during a TLS connection. If the TOE supports certificate pinning, all pinned certificates must be removed before performing Tests 1 through 6. A pinned certificate must be added prior to performing Test 7. (TD0499 applied)

Test 1: The evaluator shall present a server certificate that contains a CN that does not match the reference identifier and does not contain the SAN extension. The evaluator shall verify that the connection fails. Note that some systems might require the presence of the SAN extension. In this case the connection would still fail but for the reason of the missing SAN extension instead of the mismatch of CN and reference identifier. Both reasons are acceptable to pass Test 1.

Test 2: The evaluator shall present a server certificate that contains a CN that matches the reference identifier, contains the SAN extension, but does not contain an identifier in the SAN that matches the reference identifier. The evaluator shall verify that the connection fails. The evaluator shall repeat this test for each supported SAN type.

Test 3: [conditional] If the TOE does not mandate the presence of the SAN extension, the evaluator shall present a server certificate that contains a CN that matches the reference identifier and does not contain the SAN extension. The evaluator shall verify that the connection succeeds. If the TOE does mandate the presence of the SAN extension, this Test shall be omitted.

Test 4: The evaluator shall present a server certificate that contains a CN that does not match the reference identifier but does contain an identifier in the SAN that matches. The evaluator shall verify that the connection succeeds.

Test 5: The evaluator shall perform the following wildcard tests with each supported type of reference identifier. The support for wildcards is intended to be optional. If wildcards are supported, the first, second, and third tests below shall be executed. If wildcards are not supported, then the fourth test below shall be executed.

Test 5.1: [conditional]: If wildcards are supported, the evaluator shall present a server certificate containing a wildcard that is not in the left-most label of the presented identifier (e.g. foo.*.example.com) and verify that the connection fails.

Test 5.2: [conditional]: If wildcards are supported, the evaluator shall present a server certificate containing a wildcard in the left-most label but not preceding the public suffix (e.g. *.example.com). The evaluator shall configure the reference identifier with a single left-most label (e.g. foo.example.com) and verify that the

connection succeeds. The evaluator shall configure the reference identifier without a left-most label as in the certificate (e.g. example.com) and verify that the connection fails. The evaluator shall configure the reference identifier with two left-most labels (e.g. bar.foo.example.come) and verify that the connection fails.

Test 5.3: [conditional]: If wildcards are supported, the evaluator shall present a server certificate containing a wildcard in the left-most label immediately preceding the public suffix (e.g. *.com). The evaluator shall configure the reference identifier with a single left-most label (e.g. foo.com) and verify that the connection fails. The evaluator shall configure the reference identifier with two left-most labels (e.g. bar.foo.com) and verify that the connection fails.

Test 5.4: [conditional]: If wildcards are not supported, the evaluator shall present a server certificate containing a wildcard in the left-most label (e.g. *.example.com). The evaluator shall configure the reference identifier with a single left-most label (e.g. foo.example.com) and verify that the connection fails.

Test 6: [conditional] If URI or Service name reference identifiers are supported, the evaluator shall configure the DNS name and the service identifier. The evaluator shall present a server certificate containing the correct DNS name and service identifier in the URIName or SRVName fields of the SAN and verify that the connection succeeds. The evaluator shall repeat this test with the wrong service identifier (but correct DNS name) and verify that the connection fails.

Test 7: [conditional] If pinned certificates are supported the evaluator shall present a certificate that does not match the pinned certificate and verify that the connection fails.

The evaluator performed the following tests against the UEM Server's TLS client interface (including the syslog and LDAP client implementations).

Test 1: The evaluator established a TSL session from each of the syslog and LDAP client implementations targeting a server using a valid certificate with a CN matching the domain name used by the client. Using a network sniffer to capture the TLS session negotiation the evaluator examined the traffic capture and observed a successful connection. The evaluator then established a TLS session from the UEM Server targeting a server using a server certificate that does not contain an identifier in either the Subject Alternative Name (SAN) or Common Name (CN) that matches the reference identifier. Using a network sniffer to capture the TLS session negotiation the evaluator examined the traffic capture and observed that the TLS session was not negotiated successfully.

Test 2: The evaluator established a TSL session from each of the syslog and LDAP client implementations targeting a server using a server certificate that contains a CN that matches the reference identifier, contains the SAN extension, but does not contain an identifier in the SAN that matches the reference identifier. Using a network sniffer to capture the TLS session negotiation the evaluator examined the traffic capture and observed that the TLS session was not negotiated successfully.

Test 3: The evaluator established a TSL session from each of the syslog and LDAP client implementations targeting a server using a server certificate that contains a CN that matches the reference identifier and does not contain the

SAN extension. Using a network sniffer to capture the TLS session negotiation the evaluator examined the traffic capture and observed that the TLS session was negotiated successfully.

Test 4: The evaluator established a TSL session from each of the syslog and LDAP client implementations targeting a server using a server certificate that contains a CN that does not match the reference identifier but does contain an identifier in the SAN that matches. Using a network sniffer to capture the TLS session negotiation the evaluator examined the traffic capture and observed that the TLS session was negotiated successfully.

Test 5: The evaluator configured a test server to use a server certificate containing a reference identifier as described by test 5 from the above assurance activity. The evaluator established a TLS session from each of the syslog and LDAP client implementations targeting the name shown in column 2 of the following table. Using a network sniffer to capture the TLS session negotiation the evaluator examined the traffic capture and observed that the TLS session was negotiated as shown in column 3 of the following table.

Certificate Contents	Host ID	Expected Result
CN=bar.*.example.com	bar.foo.example.com	No Connection
SAN=bar.*.example.com	bar.foo.example.com	No Connection
CN= *.example.com	bar.foo.example.com	No Connection
SAN = *.example.com	bar.foo.example.com	No Connection
CN= *.example.com	foo.example.com	Successful Connection
SAN= *.example.com	foo.example.com	Successful Connection
CN= *.com	foo.example.com	No Connection
SAN= *.com	foo.example.com	No Connection
CN= *.example.com	example.com	No Connection
SAN= *.example.com	example.com	No Connection
CN= *.com	example.com	No Connection
SAN= *.com	example.com	No Connection

Test 6: The TOE does not support the optional URI or Service Name used as reference identifiers.

Test 7: The TOE does not support certificate pinning.

2.2.15.3 PKGTLS11:FCS_TLSC_EXT.1.3

TSS Assurance Activities: If the selection for authorizing override of invalid certificates is made, then the evaluator shall ensure that the TSS includes a description of how and when user or administrator authorization is obtained. The evaluator shall also ensure that the TSS describes any mechanism for storing such authorizations, such that future presentation of such otherwise-invalid certificates permits establishment of a trusted channel without user or administrator action.

The product does not offer a mechanism to override invalid certificates.

Guidance Assurance Activities: None Defined

Testing Assurance Activities: The evaluator shall demonstrate that using an invalid certificate results in the function failing as follows, unless excepted:

Test 1a: The evaluator shall demonstrate that a server using a certificate with a valid certification path successfully connects.

Test 1b: The evaluator shall modify the certificate chain used by the server in test 1a to be invalid and demonstrate that a server using a certificate without a valid certification path to a trust store element of the TOE results in an authentication failure.

Test 1c [conditional]: If the TOE trust store can be managed, the evaluator shall modify the trust store element used in Test 1a to be untrusted and demonstrate that a connection attempt from the same server used in Test 1a results in an authentication failure.

(TD0513 applied)

Test 2: The evaluator shall demonstrate that a server using a certificate which has been revoked results in an authentication failure.

Test 3: The evaluator shall demonstrate that a server using a certificate which has passed its expiration date results in an authentication failure.

Test 4: The evaluator shall demonstrate that a server using a certificate which does not have a valid identifier results in an authentication failure.

Test 1: The evaluator configured server using a certificate with a valid certification path terminating in a certificate which was not configured in the TOE as trusted. The evaluator observed that the TOE rejected the certificate. The evaluator then loaded the trusted CA certificate(s) needed to validate the server's certificate, and demonstrated that the connection succeeded. The evaluator then deleted the CA certificate that was loaded in the previous test part, and showed that the connection again failed.

Test 2: The evaluator demonstrated that a server presenting a certificate which has been revoked to the TOE results in the TOE rejecting the certificate and the connection failing.

Test 3: The evaluator demonstrated that a server presenting a certificate which has passed its expiration date results in the TOE rejecting the certificate and the connection failing.

Test 4: The evaluator demonstrated that a server presenting a certificate which does not have a valid identifier results in the TOE rejecting the certificate and the connection failing.

Component TSS Assurance Activities: None Defined

Component Guidance Assurance Activities: None Defined

Component Testing Assurance Activities: None Defined

2.2.16 TLS Client Support for Mutual Authentication (PKGTLS11:FCS_TLSC_EXT.2)

2.2.16.1 PKGTLS11:FCS_TLSC_EXT.2.1

TSS Assurance Activities: The evaluator shall ensure that the TSS description required per FIA_X509_EXT.2.1 includes the use of client-side certificates for TLS mutual authentication. The evaluator shall also ensure that the TSS describes any factors beyond configuration that are necessary in order for the client to engage in mutual authentication using X.509v3 certificates.

Section 6.2 of [ST] explains that the UEM Server can be configured with a certificate to present during TLS negotiation with an audit (Syslog) server and LDAP server. The certificates used by the UEM server for these communication channels are independent from one another and from the TOE's internal certificates used for communication with mobile devices. The administrator must load the certificate corresponding to each client communication channel, prior to installing the UEM server's client certificate for these channels.

Guidance Assurance Activities: The evaluator shall ensure that the AGD guidance includes any instructions necessary to configure the TOE to perform mutual authentication. The evaluator also shall verify that the AGD guidance required per FIA_X509_EXT.2.1 includes instructions for configuring the client-side certificates for TLS mutual authentication.

The sections entitled "Set up export of server audit records to a syslog server" and "Set up export of device audit records to a syslog server" describe the steps to load the trusted root for a syslog server. These sections include instructions to update and execute the configurations scripts from Appendix A and B. Appendix A and B contain additional configuration scripts that are used during TOE installation and TOE updates. These scripts contain

values which cause the TOE to use TLS and to provide mutual authentication to external syslog services as requested. These scripts also indicate the reference identifier port being used by the syslog server.

Section "Validate and install the BlackBerry UEM software" of [Admin] provides instructions to install the system using a method that will ensure that the TOE is using TLS and certificates in a manner that confirms to the description in [ST]. That is, no cryptographic configuration of key generation, key establishment, encryption, signature algorithms or hashing (e.g., choosing ciphersuites) can be performed on the UEM Server following installation.

Testing Assurance Activities: The evaluator shall also perform the following tests:

Test 1: The evaluator shall establish a connection to a server that is not configured for mutual authentication (i.e. does not send Server's Certificate Request (type 13) message). The evaluator observes negotiation of a TLS channel and confirms that the TOE did not send Client's Certificate message (type 11) during handshake.

Test 2: The evaluator shall establish a connection to a server with a shared trusted root that is configured for mutual authentication (i.e. it sends Server's Certificate Request (type 13) message). The evaluator observes negotiation of a TLS channel and confirms that the TOE responds with a non-empty Client's Certificate message (type 11) and Certificate Verify (type 15) message.

Test 1: The evaluator established a TLS session from each of the syslog and LDAP client implementations and a test server that was not configured for mutual authentication. The evaluator observed that the TLS connection was successful and the TOE did not send a certificate or a certificate verify message.

Test 2: The evaluator established a TLS session from each of the syslog and LDAP client implementations and a test server that was configured for mutual authentication. The evaluator observed that the TLS connection was successful and the TOE did send both a certificate and a certificate verify message.

Component TSS Assurance Activities: None Defined

Component Guidance Assurance Activities: None Defined

Component Testing Assurance Activities: None Defined

2.2.17 TLS Client Support for Supported Groups Extension (PKGTLS11:FCS_TLSC_EXT.5)

2.2.17.1 PKGTLS11:FCS_TLSC_EXT.5.1

TSS Assurance Activities: The evaluator shall verify that TSS describes the Supported Groups Extension.

Section 6.2 of [ST] states that for key exchanges, the UEM server supports ECDHE key exchanges with EC curves secp256r1, secp384r1, and secp521r1.

Guidance Assurance Activities: None Defined

Testing Assurance Activities: The evaluator shall also perform the following test:

Test 1: The evaluator shall configure a server to perform key exchange using each of the TOE's supported curves and/or groups. The evaluator shall verify that the TOE successfully connects to the server.

The evaluator configured a server to use various certificates necessary to cause a TLS negotiation using each of the key exchange methods. The evaluator then initiated a TLS session from each of the syslog and LDAP client implementations while capturing traffic. Inspection of the traffic indicated that the TOE did in fact negotiate a successful connection using each of the claimed key exchange methods.

Component TSS Assurance Activities: None Defined

Component Guidance Assurance Activities: None Defined

Component Testing Assurance Activities: None Defined

2.2.18 TLS Server Protocol - per TD0739 (PKGTLS11:FCS_TLSS_EXT.1)

2.2.18.1 PKGTLS11:FCS_TLSS_EXT.1.1

TSS Assurance Activities: The evaluator shall check the description of the implementation of this protocol in the TSS to ensure that the cipher suites supported are specified. The evaluator shall check the TSS to ensure that the cipher suites specified include those listed for this component.

Section 6.2 of [ST] indicates that the TOE supports TLSv1.2 only, using the following ciphersuites when it is acting as a TLS Server.

- TLS_RSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5246, (Server-to-Agent communication only)
- TLS_RSA_WITH_AES_256_CBC_SHA256 as defined in RFC 5246, (Server-to-Agent communication only)
- TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5289,
- TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384 as defined in RFC 5289,
- TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5289

Guidance Assurance Activities: The evaluator shall also check the operational guidance to ensure that it contains instructions on configuring the TOE so that TLS conforms to the description in the TSS

The TOE does not offer configuration for cryptographic operations once installed. Section "Validate and install the BlackBerry UEM software" of [Admin] provides instructions to install the system using a method that will ensure that the TOE is using TLS and certificates in a manner that confirms to the description in [ST]. No cryptographic configuration of key generation, key establishment, encryption, signature algorithms or hashing (e.g., choosing ciphersuites) can be performed on the UEM Server.

Testing Assurance Activities: The evaluator shall also perform the following tests:

Test 1: The evaluator shall establish a TLS connection using each of the cipher suites specified by the requirement. This connection may be established as part of the establishment of a higher-level protocol, e.g., as part of an EAP session. It is sufficient to observe the successful negotiation of a cipher suite to satisfy the intent of the test; it is not necessary to examine the characteristics of the encrypted traffic in an attempt to discern the cipher suite being used (for example, that the cryptographic algorithm is 128-bit AES and not 256-bit AES).

Test 2: The evaluator shall send a Client Hello to the server with a list of cipher suites that does not contain any of the cipher suites in the server's ST and verify that the server denies the connection. Additionally, the evaluator shall send a Client Hello to the server containing only the TLS_NULL_WITH_NULL_NULL cipher suite and verify that the server denies the connection.

Test 3: If RSA key exchange is used in one of the selected ciphersuites, the evaluator shall use a client to send a properly constructed Key Exchange message with a modified EncryptedPreMasterSecret field during the TLS handshake. The evaluator shall verify that the handshake is not completed successfully and no application data flows.

Test 4: The evaluator shall perform the following modifications to the traffic:

Test 4.1: Removed per TD0469

Test 4.2: Modify a byte in the data of the client's Finished handshake message, and verify that the server rejects the connection and does not send any application data.

Test 4.3: Demonstrate that the TOE will not resume a session for which the client failed to complete the handshake (independent of TOE support for session resumption):

Test 4.3i [conditional]: If the TOE does not support session resumption based on session IDs according to RFC4346 (TLS1.1) or RFC5246 (TLS1.2) or session tickets according to RFC5077, the evaluator shall perform the following test:

a) The evaluator shall send a Client Hello with a zero-length session identifier and with a SessionTicket extension containing a zero-length ticket.

b) The evaluator shall verify the server does not send a NewSessionTicket handshake message (at any point in the handshake).

c) The evaluator shall verify the Server Hello message contains a zero-length session identifier or passes the following steps:

Note: The following steps are only performed if the ServerHello message contains a non-zero length SessionID.

d) The evaluator shall complete the TLS handshake and capture the SessionID from the ServerHello.

e) The evaluator shall send a ClientHello containing the SessionID captured in step d). This can be done by keeping the TLS session in step d) open or start a new TLS session using the SessionID captured in step d).

f) The evaluator shall verify the TOE (1) implicitly rejects the SessionID by sending a ServerHello containing a different SessionID and by performing a full handshake (as shown in Figure 1 of RFC 4346 or RFC 5246), or (2) terminates the connection in some way that prevents the flow of application data.

Test 4.3ii [conditional]: If the TOE supports session resumption using session IDs according to RFC4346 (TLS1.1) or RFC5246 (TLS1.2), the evaluator shall carry out the following steps (note that for each of these tests, it is not necessary to perform the test case for each supported version of TLS):

a) The evaluator shall conduct a successful handshake and capture the TOE-generated session ID in the Server Hello message. The evaluator shall then initiate a new TLS connection and send the previously captured session ID to show that the TOE resumed the previous session by responding with ServerHello containing the same SessionID immediately followed by ChangeCipherSpec and Finished messages (as shown in Figure 2 of RFC 4346 or RFC 5246).

b) The evaluator shall initiate a handshake and capture the TOE-generated session ID in the Server Hello message. The evaluator shall then, within the same handshake, generate or force an unencrypted fatal Alert message immediately before the client would otherwise send its ChangeCipherSpec message thereby disrupting the handshake. The evaluator shall then initiate a new Client Hello using the previously captured session ID, and verify that the server (1) implicitly rejects the session ID by sending a ServerHello containing a different SessionID and performing a full handshake (as shown in figure 1 of RFC 4346 or RFC 5246), or (2) terminates the connection in some way that prevents the flow of application data.

Test 4.3iii [conditional]: If the TOE supports session tickets according to RFC5077, the evaluator shall carry out the following steps (note that for each of these tests, it is not necessary to perform the test case for each supported version of TLS):

a) The evaluator shall permit a successful TLS handshake to occur in which a session ticket is exchanged with the non-TOE client. The evaluator shall then attempt to correctly reuse the previous session by sending the session ticket in the ClientHello. The evaluator shall confirm that the TOE responds with a ServerHello with an empty SessionTicket extension, NewSessionTicket, ChangeCipherSpec and Finished messages (as seen in figure 2 of RFC 5077).

b) The evaluator shall permit a successful TLS handshake to occur in which a session ticket is exchanged with the non-TOE client. The evaluator will then modify the session ticket and send it as part of a new Client Hello message. The evaluator shall confirm that the TOE either (1) implicitly rejects the session ticket by performing a full handshake (as shown in figure 3 or 4 of RFC 5077), or (2) terminates the connection in some way that prevents the flow of application data.

Test 4.4: Send a message consisting of random bytes from the client after the client has issued the ChangeCipherSpec message and verify that the server denies the connection.

(TD0588 applied)

Test 1: The UEM server acts as a TLS server when providing its Web UI and when accepting connections from Agent software on a mobile device. The evaluator established a TLS session from a test server to the UEM Server Web UI and Agent interfaces for each of the claimed ciphersuites in turn. The evaluator used a network sniffer to capture the TLS session negotiation and observed that the expected TLS cipher is negotiated.

Test 2: Using each TLS server interface offered by the UEM Server, the evaluator sent a Client Hello to the server with a list of cipher suites that does not contain any of the cipher suites in the server's ST and verified that the server denies the connection. Additionally, the evaluator sent a Client Hello to the server containing only the TLS_NULL_WITH_NULL_NULL cipher suite and observed that the session is not successfully negotiated.

Test 3: The TOE does not support RSA key exchanges on the Web UI interface; therefore, this test was performed only on the UEM Server interface for Agent communication. The evaluator used a test system to connect to the UEM Server's TLS interface offered to Agents for communication. Using an RSA key exchange, the evaluator modified the EncryptedPreMasterSecret field during the TLS handshake and found that the UEM Server rejected the connection.

Test 4.1: Removed by TD:

Test 4.2 - The evaluator attempted a connection to the TOE where the evaluator modified a byte in the Finished handshake message, verified that the TOE rejected the connection attempt after receiving the modified Finished message, and that the TOE sent no application data.

Test 4.3i - Not Applicable, as the TOE does support session resumption based on session IDs.

Test 4.3ii: The evaluator first established a TLS connection and demonstrated a successful session ID reuse. Next, the evaluator attempted to open a TLS connection to the TOE where the evaluator's client recorded the session_id from a failed TLS connection attempt (by sending the Finished message before the ChangeCipherSpec message). The evaluator observed the TOE reject the connection. Finally, the evaluator attempted to initiate a TLS connection where the session ID was reused from the previous, failed connection. The evaluator observed that the TOE rejected the session resumption with the bad session ID, and returned a new session ID.

Test 4.3iii - Not Applicable, as the TOE does not support session tickets

Test 4.4 - The evaluator garbled a message between the TOE and its TLS peer. The evaluator observed that the Client denies the connection. Due to the nature of the error, regardless of whether the TOE is the client or server, the client is always the first to recognize the error.

2.2.18.2 PKGTLS11:FCS_TLSS_EXT.1.2

TSS Assurance Activities: The evaluator shall verify that the TSS contains a description of the denial of old SSL and TLS versions consistent relative to selections in FCS_TLSS_EXT.1.2.

Section 6.2 of [ST] states that the UEM Server supports TLS versions 1.2. It further states that all versions of the SSL protocol and older versions of the TLS protocol are refused by the UEM Server.

Guidance Assurance Activities: The evaluator shall verify that the AGD guidance includes any configuration necessary to meet this requirement.

The TOE does not offer configuration for cryptographic operations once installed. Section "Validate and install the BlackBerry UEM software" of [Admin] provides instructions to install the system using a method that will ensure that the TOE is using TLS and certificates in a manner that confirms to the description in [ST]. No cryptographic configuration of key generation, key establishment, encryption, signature algorithms or hashing (e.g., choosing ciphersuites) can be performed on the UEM Server.

Testing Assurance Activities: Test 1: The evaluator shall send a Client Hello requesting a connection with version SSL 2.0 and verify that the server denies the connection. The evaluator shall repeat this test with SSL 3.0 and TLS 1.0, and TLS 1.1 if it is selected.

The evaluator attempted to establish a TLS session on from a test system to each of the TOE TLS server implementations (i.e., Web UI and Agent communication) using each of the unsupported versions of SSL and TLS. The evaluator used a network sniffer to capture the session negotiation and observed that only the expected protocol and version were offered and that all others were rejected during negotiation.

2.2.18.3 PKGTLS11:FCS_TLSS_EXT.1.3

TSS Assurance Activities: The evaluator shall verify that the TSS describes the key agreement parameters of the server's Key Exchange message

Section 6.2 of [ST] indicates that the UEM Server supports RSA key exchanges using 2048-bit, 3072-bit, or 4096-bit RSA certificates and ECDHE key exchanges with EC curves secp256r1, secp384r1, and secp521r1.

Guidance Assurance Activities: The evaluator shall verify that any configuration guidance necessary to meet the requirement must be contained in the AGD guidance.

The TOE does not offer configuration for cryptographic operations once installed. Section "Validate and install the BlackBerry UEM software" of [Admin] provides instructions to install the system using a method that will ensure that the TOE is using TLS and certificates in a manner that confirms to the description in [ST]. No cryptographic configuration of key generation, key establishment, encryption, signature algorithms or hashing (e.g., choosing ciphersuites) can be performed on the UEM Server.

Testing Assurance Activities: The evaluator shall conduct the following tests. The testing can be carried out manually with a packet analyzer or with an automated framework that similarly captures such empirical evidence. Note that this testing can be accomplished in conjunction with other testing activities. For each of the following tests, determining that the size matches the expected size is sufficient.

Test 1: [conditional] If RSA-based key establishment is selected, the evaluator shall configure the TOE with a certificate containing a supported RSA size and attempt a connection. The evaluator shall verify that the size used matches that which is configured and that the connection is successfully established. The evaluator shall repeat this test for each supported size of RSA-based key establishment. (TD0739 applied)

Test 2: [conditional] If finite-field (i.e. non-EC) Diffie-Hellman ciphers are selected, the evaluator shall attempt a connection using a Diffie-Hellman key exchange with a supported parameter size or supported group. The evaluator shall verify that the key agreement parameters in the Key Exchange message are the ones configured. The evaluator shall repeat this test for each supported parameter size or group.

Test 3: [conditional] If ECDHE ciphers are selected, the evaluator shall attempt a connection using an ECDHE ciphersuite with a supported curve. The evaluator shall verify that the key agreement parameters in the Key Exchange message are the ones configured. The evaluator shall repeat this test for each supported elliptic curve.

Test 1-3: The evaluator attempted a connection to the UEM Server agent communication interface using a 2048-bit RSA-based key establishment and found that the connection was successful. This was repeated with 3072 bit and 4096 key sizes. The evaluator also demonstrated that for both the Web UI and the Agent communication interfaces a TLS session could be negotiated using ECDHE key exchange with either secp256r1, secp384r1 or secp521r1 curves.

Component TSS Assurance Activities: None Defined

Component Guidance Assurance Activities: None Defined

Component Testing Assurance Activities: None Defined

2.2.19 TLS Server Support for Mutual Authentication (PKGTLS11:FCS_TLSS_EXT.2)

2.2.19.1 PKGTLS11:FCS_TLSS_EXT.2.1

TSS Assurance Activities: None Defined

Guidance Assurance Activities: None Defined

Testing Assurance Activities: None Defined

2.2.19.2 PKGTLS11:FCS_TLSS_EXT.2.2

TSS Assurance Activities: The evaluator shall ensure that the TSS description required per FIA_X509_EXT.2.1 includes the use of client-side certificates for TLS mutual authentication.

If error messages are provided prior to terminating a session, the evaluator shall ensure the error messages are described.

Section 6.3 of [ST] explains that the server and client use X.509 certificates for mutual authentication during the establishment of the trusted channel for UEM Server to Agent communication.

Guidance Assurance Activities: The evaluator shall verify that the AGD guidance required per FIA_X509_EXT.2.1 includes instructions for configuring the client-side certificates for TLS mutual authentication. The evaluator shall ensure that the AGD guidance includes instructions for configuring the server to require mutual authentication of clients using these certificates.

The TOE does not offer configuration for cryptographic operations once installed. Section "Validate and install the BlackBerry UEM software" of [Admin] provides instructions to install the system using a method that will ensure that the TOE is using TLS and certificates in a manner that confirms to the description in [ST]. No cryptographic configuration of key generation, key establishment, encryption, signature algorithms or hashing (e.g., choosing ciphersuites) can be performed on the UEM Server.

Testing Assurance Activities: The evaluator shall use TLS as a function to verify that the validation rules in FIA_X509_EXT.1.1 are adhered to and shall perform the following tests. The evaluator shall apply the AGD guidance to configure the server to require TLS mutual authentication of clients for the following tests, unless overridden by instructions in the test activity.

If error messages are provided prior to terminating a session, the evaluator shall configure the test client to be able to observe application data sent over the non-mutually authenticated channel, and in addition to observing that the TSF terminates the session, that the only data received under the channel is the error message as described in the TSS:

Test 1: The evaluator shall configure the server to send a certificate request to the client. The client shall send a certificate_list structure which has a length of zero. The evaluator shall verify that no sensitive application data flows prior to termination; if error messages are sent, the evaluator shall observe that an non-mutually authenticated channel is established, observe the data received by the test client to ensure only the error message indicated in the TSS is provided, and observe that the channel is then terminated.

Test 2: The evaluator shall configure the server to send a certificate request to the client. The client shall send no client certificate message, and instead send a client key exchange message in an attempt to continue the handshake. The client is required to respond to the certificate request message, even if the certificate message is empty. The evaluator shall verify that the handshake is not finished successfully and no application data flows.

Test 3: The evaluator shall configure the server to send a certificate request to the client without the supported_signature_algorithm used by the client's certificate. The evaluator shall attempt a connection using the client certificate and verify no sensitive application data flows prior to termination; if error messages are sent, the evaluator shall observe that an non-mutually authenticated channel is established, observe the data received by the test client to ensure only the error message indicated in the TSS is provided, and observe that the channel is then terminated.

Test 4: The evaluator shall demonstrate that using a certificate without a valid certification path results in the function failing. Using the administrative guidance, the evaluator shall then load a certificate or certificates needed to validate the certificate to be used in the function, and demonstrate that the function succeeds. The evaluator then shall delete one of the certificates, load the modified certificate path, and verify that no sensitive application data flows prior to termination; if error messages are sent, the evaluator shall observe that an non-mutually authenticated channel is established, observe the data received by the test client to ensure only the error message indicated in the TSS is provided, and observe that the channel is then terminated.

Test 5: The aim of this test is to check the response of the server when it receives a client identity certificate that is signed by an impostor CA (either Root CA or intermediate CA). To carry out this test the evaluator shall configure the client to send a client identity certificate with an issuer field that identifies a CA recognised by the TOE as a trusted CA, but where the key used for the signature on the client certificate does not in fact correspond to the CA certificate trusted by the TOE (meaning that the client certificate is invalid because its certification path does not in fact terminate in the claimed CA certificate). The evaluator shall verify that no sensitive application data flows prior to termination; if error messages are sent, the evaluator shall observe that an non-mutually authenticated channel is established, observe the data received by the test client to ensure only the error message indicated in the TSS is provided, and observe that the channel is then terminated.

Test 6: The evaluator shall configure the client to send a certificate with the Client Authentication purpose in the extendedKeyUsage field and verify that the server accepts the attempted connection. The evaluator shall repeat this test without the Client Authentication purpose and shall verify that no sensitive application data flows prior to termination; if error messages are sent, the evaluator shall observe that an non-mutually authenticated channel is established, observe the data received by the test client to ensure only the error message indicated in the TSS is

provided, and observe that the channel is then terminated. Ideally, the two certificates should be identical except for the Client Authentication purpose.

Test 7: The evaluator shall perform the following modifications to the traffic: a) Configure the server to require mutual authentication and then modify a byte in the client's certificate. The evaluator shall verify that the server rejects the connection. b) Configure the server to require mutual authentication and then modify a byte in the signature block of the client's Certificate Verify handshake message. The evaluator shall verify that the server rejects the connection.

The UEM Server Web UI does not support mutual authentication, so this testing was performed on the UEM Server to Agent communication interface only.

Test 1: The UEM Server always sends a certificate request to a client. The evaluator attempted to connect a test system to the UEM Server's Agent communication interfaces where the client (test system) responded to the certificate request with a certificate list structure which had a length of zero. Using a packet capture the evaluator determined that the handshake was not finished successfully and no application data was exchanged.

Test 2: The evaluator attempted to connect a test system to the UEM Server's Agent communication interfaces where the client (test system) responded to the certificate request with no client certificate message. The client instead sent a client key exchange message in an attempt to continue the handshake. The evaluator verified that the handshake is not finished successfully and no application data flows.

Test 3: The evaluator configured a client to have a certificate using a signature algorithm not supported by the UEM Server. The evaluator's test system provided this certificate to the UEM Server in response to the UEM Server's CERTIFICATE REQUEST message and observed that the UEM server did not complete the TLS session negotiation successfully.

Test 4: The UEM Server's Agent communication interface always requires mutual authentication. The evaluator attempted to connect a client this interface where the client used a certificate with a broken certificate path. The evaluator used a network sniffer to capture the TLS session negotiation and observed that the TLS session is rejected by the server. This test was performed in conjunction with the FIA_X509_EXT.1.1(1) Test 1 invalid chain test.

Test 5: The evaluator configured a test system to act as a TLS client and to send a client identity certificate to the UEM Server's Agent communication interface. The certificate had an issuer field that identifies a CA recognized by the TOE as a trusted CA, but where the key used for the signature on the client certificate does not in fact correspond to the CA certificate trusted by the TOE (i.e., the client certificate is invalid because its certification path does not in fact terminate in the claimed CA certificate). Using a packet capture the evaluator determined that the handshake was not finished successfully.

Test 6: The UEM Server's Agent communication interface always requires mutual authentication. The evaluator attempted to connect a client this interface where the client sent a certificate with the Client Authentication purpose in the extendedKeyUsage field. Using a network sniffer to capture the TLS session negotiation and

observed that the TLS session was accepted by the server. The evaluator retried the TLS session to connect a client this interface where the client sent a certificate that is missing the Client Authentication purpose in the extendedKeyUsage field. The evaluator used a network sniffer to capture the TLS session negotiation and observed that the TLS session was rejected by the server.

Test 7: Using the UEM Server's Agent communication interface, the evaluator performed the modifications to the traffic described by the assurance activity above and verified that the server rejects the TLS connection.

2.2.19.3 PKGTLS11:FCS_TLSS_EXT.2.3

TSS Assurance Activities: If the product implements mutual authentication, the evaluator shall verify that the TSS describes how the DN and SAN in the certificate is compared to the expected identifier.

Section 6.2 of [ST] states that when the UEM server is accepting TLS communications from an Agent, the UEM server verifies the UEM Android Client's certificate by ensuring that that the Distinguished Name (DN) in the presented certificate matches a DN in a database of valid, known DNs.

Guidance Assurance Activities: If the DN is not compared automatically to the domain name, IP address, username, or email address, the evaluator shall ensure that the AGD guidance includes configuration of the expected identifier or the directory server for the connection.

The TOE requires no special configuration of a reference identifier to identify enrolled devices. During enrollment, the TOE automatically generates a certificate for the device which includes a unique distinguished name. This name is automatically compared to the certificate presented by a device each time the device contacts the UEM Server.

Testing Assurance Activities: Test 1: The evaluator shall send a client certificate with an identifier that does not match any of the expected identifiers and verify that the server denies the connection. The matching itself might be performed outside the TOE (e.g. when passing the certificate on to a directory server for comparison).

Using the UEM Server's Agent communication interface, evaluator captured the network traffic for the TLS client connection attempts and verified that the TLS session connected as expected using the proper certificate with matching DN. The evaluator also determined that the connection was rejected using a certificate with a DN identifier that does not match.

Component TSS Assurance Activities: None Defined

Component Guidance Assurance Activities: None Defined

Component Testing Assurance Activities: None Defined

2.3 IDENTIFICATION AND AUTHENTICATION (FIA)

2.3.1 CLIENT AUTHORIZATION - PER TD0754 (MDMPP40:FIA_CLI_EXT.1)

2.3.1.1 MDMPP40:FIA_CLI_EXT.1.1

TSS Assurance Activities: None Defined

Guidance Assurance Activities: None Defined

Testing Assurance Activities: None Defined

Component TSS Assurance Activities: The evaluator shall ensure that the TSS describes how the client is uniquely identified.

Section 6.3 of the ST explains that the TOE issues unique X509 certificates from the UEM server during the enrollment process.

These certificates are used by the TOE to uniquely identify each client.

Component Guidance Assurance Activities: None Defined

Component Testing Assurance Activities: None Defined

2.3.2 ENROLLMENT OF MOBILE DEVICE INTO MANAGEMENT (MDMPP40:FIA_ENR_EXT.1)

2.3.2.1 MDMPP40:FIA_ENR_EXT.1.1

TSS Assurance Activities: The evaluator shall examine the TSS and verify that it describes the process of enrollment for each MDM Agent/platform listed as supported in theST. This description shall include the trusted path used for enrollment (FTP_TRP.1(2)), the method of user authentication (username/password, token, etc.), the method of authentication decision (local or remote authentication services), and the actions performed on the MDM Server upon successful authentication.

Section 6.3 of [ST] describes that during the enrollment process, a user (or administrator) can log into the UEM Server to issue an activation password. The activation password is in turn used to activate a mobile device over a secure TLS channel between the agent on the mobile device and the UEM Server. The UEM Server, having authenticated to the agent through presentation of its certificate during the TLS handshake, checks that the activation password is valid. An administrator can configure the UEM Server to limit enrollment based on device

serial numbers, number of devices, and within a specific time period. Assuming any configured conditions are met, the UEM Server will issue an X509 certificate to the agent. Once the enrollment process has completed, all subsequent connections between the agents and the UEM Server occur through a mutually authenticated TLS session (in which the agent presents its certificate to the server).

Guidance Assurance Activities: None Defined

Testing Assurance Activities: Test 1: The evaluator shall attempt to enroll a device without providing correct credentials. The evaluator shall verify that the device is not enrolled and that the described enrollment actions are not taken.

Test 2: The evaluator shall attempt to enroll the device providing correct credentials. The evaluator shall verify that the device is enrolled and that the described enrollment actions are taken.

Test 1: The evaluator attempted to enroll a device without providing correct credentials. The evaluator verified that the device is not enrolled and that the described enrollment actions are not taken.

Test 2: The evaluator enrolled the device providing correct credentials. The evaluator verified that the device is enrolled and that the described enrollment actions are taken.

2.3.2.2 MDMPP40:FIA_ENR_EXT.1.2

TSS Assurance Activities: The evaluator shall examine the TSS and verify that it implements a policy to limit the user's enrollment of devices.

Section 6.3 of [ST] explains that an administrator can configure the UEM Server to limit enrollment based on device serial numbers, number of devices, and within a specific time period.

Guidance Assurance Activities: The evaluator shall ensure that the administrative guidance describes the method(s) of restricting user enrollment and that it instructs the administrator how to configure the restrictions.

The section entitled "Create an activation profile" in [Admin] provide instructions for defining an activation profile which can be used to specify the restrictions place on a user's enrollment of devices. This indicates using the activation profile the administrator can place a restriction based on number of devices. Restrictions based on time period are set using instructions found in the section entitled, “Set an activation password and send an activation email Message”. The section entitled “Import or export a list of approved device IDs” provides instructions to set the restriction based on serial numbers.

Testing Assurance Activities: For each type of policy selected, the evaluator shall perform the following:

Test 1: The evaluator shall attempt to configure the MDM Server according to the administrative guidance in order to prevent enrollment. The evaluator shall verify that the user cannot enroll a device outside of the configured limitation. (For example, the evaluator may try to enroll a disallowed device, or may try to enroll additional devices beyond the number allowed).

The evaluator configured limits on device enrollment using device serial numbers, number of devices, and within a specific time period. The evaluator then attempted to enroll devices that violated each configured limitation and found that the UEM Server enforced these limitations.

Component TSS Assurance Activities: None Defined

Component Guidance Assurance Activities: None Defined

Component Testing Assurance Activities: None Defined

2.3.3 Agent Enrollment of Mobile Device into Management (MDMA10:FIA_ENR_EXT.2)

2.3.3.1 MDMA10:FIA_ENR_EXT.2.1

TSS Assurance Activities: None Defined

Guidance Assurance Activities: None Defined

Testing Assurance Activities: None Defined

Component TSS Assurance Activities: The evaluator shall examine the TSS to verify that it describes which types of reference identifiers are acceptable and how the identifier is specified (e.g. preconfigured in the MDM Agent, by the user, by the MDM server, in a policy).

Section 6.3 of [ST] states that during enrollment the UEM Android Client records the unique URL (FQDN or IP address) of the UEM Server for future communication purposes. This value is initially configured by the mobile device user when attempting to enroll the mobile device.

Component Guidance Assurance Activities: The evaluator shall examine the operational guidance to verify that it describes how to configure reference identifier of the MDM Server's certificate and, if different than the reference identifier, the Domain Name or IP address (for connectivity) of the MDM Server.

The TOE offers no ability to configure the reference identifier within the MDM Server's TLS certificates used at the Web UI and Agent Communication interfaces. The MDM Server's TLS certificates are created during installation and cannot be changed without re-installation of the product.

Component Testing Assurance Activities: The evaluator shall follow the operational guidance to establish the reference identifier of the MDM server on the MDM Agent and in conjunction with other evaluation activities verify that the MDM Agent can connect to the MDM Server and validate the MDM Server's certificate.

The evaluator used the UEM Server to generate a QR code that included information to allow the UEM Android client to connect to the UEM Server, verify its certificate, and proceed with enrollment. The evaluator verified that using the QR code allowed the UEM Android client to successfully validate the UEM Server certificate and successfully enroll the device.

2.3.4 Timing of Authentication (MDMPP40:FIA_UAU.1)

2.3.4.1 MDMPP40:FIA_UAU.1.1

TSS Assurance Activities: If 'invoke platform-provided functionality' is selected, the evaluator shall examine the TSS to verify that it describes (for each supported platform) how this functionality is invoked (it should be noted that this may be through a mechanism that is not implemented by the MDM Server; nonetheless, that mechanism will be identified in the TSS as part of this evaluation activity).

The selection for this SFR in [ST] was to 'implement functionality'.

Guidance Assurance Activities: None Defined

Testing Assurance Activities: None Defined

2.3.4.2 MDMPP40:FIA_UAU.1.2

TSS Assurance Activities: If 'invoke platform-provided functionality' is selected, the evaluator shall examine the TSS to verify that it describes (for each supported platform) how this functionality is invoked (it should be noted that this may be through a mechanism that is not implemented by the MDM Server; nonetheless, that mechanism will be identified in the TSS as part of this evaluation activity).

The selection for this SFR in [ST] was to 'implement functionality'.

Guidance Assurance Activities: None Defined

Testing Assurance Activities: The evaluator shall perform the following tests:

Test 1: The evaluator shall attempt to perform the prohibited actions before authentication. The evaluator shall verify the actions cannot be performed.

Test 2: The evaluator shall attempt to perform the prohibited actions after authentication. The evaluator shall verify the actions can be performed.

Test 1: The evaluator attempted the actions that are available through the UEM Server interface prior to login and found only those claimed by the SFR were available. The evaluator found no way of accessing any prohibited actions as an unauthenticated user could not make it past the login screen.

Test 2: The evaluator confirmed that only authorized administrators can login to the TOE in both the previous test case and the referenced FAU_STG_EXT.2-t1. All of the actions that were not allowed to the unauthenticated user and were otherwise tested under FMT were allowed to the correctly authenticated user in their respective tests.

Component TSS Assurance Activities: None Defined

Component Guidance Assurance Activities: None Defined

Component Testing Assurance Activities: None Defined

2.3.5 X.509 Certificate Validation - per TD0641 (MDMPP40:FIA_X509_EXT.1(1))

2.3.5.1 MDMPP40:FIA_X509_EXT.1.1(1)

TSS Assurance Activities: If invoke platform-provided functionality is selected:

The evaluator shall examine the TSS to verify that it describes (for each supported platform) how this functionality is invoked (it should be noted that this may be through a mechanism that is not implemented by the MDM server; nonetheless, that mechanism will be identified in the TSS as part of this evaluation activity.)

The TSS must describe when revocation checking is performed. It is expected that revocation checking is performed when a certificate is used in an authentication step. It is not sufficient to verify the status of an X.509 certificate only when it is loaded onto the device.

If implement funcationality is selected:

The evaluator shall ensure the TSS describes where the check of validity of the certificates takes place. The evaluator ensures the TSS also provides a description of the certificate path validation algorithm.

The TSS must describe when revocation checking is performed. It is expected that revocation checking is performed when a certificate is used in an authentication step. It is not sufficient to verify the status of an X.509 certificate only when it is loaded onto the device.

Section 6.3 of [ST] describes that the UEM Server implements the validation and handling of X509 certificates and that the UEM Android Client invokes platform services which validate and handle X.509 certificates. It also explains that the server and client use X.509 certificates for mutual authentication during the establishment of the trusted channel for UEM Server to Agent communication.

This section also describes the steps that the UEM Server implements during the validation and certificate chain checking of a certificate. Section 6.3 describes that the UEM Server validates authentication certificates (including the full path) and check their revocation station using OCSP. The TOE processes certificates presented during the TLS handshake by first checking the received certificate's validity period and appropriate key usage property. The TOE checks that it can construct a certificate path from the server's certificate through any intermediary CAs to a trusted root CA. If the TOE can successfully build the certificate path, then the TOE will next check the validity of the CA certificates (e.g., checking its validity dates and that the CA flag is present in the basic constraints section for all CA certs) in the chain. Assuming the TOE determines that all CA certificates in the chain are valid, the TOE will finally check the revocation status of the server's certificate. The TOE will not accept any certificate for which it cannot determine the validity and will reject the connection attempt.

Finally, section 6.3 explains that the UEM Android client invokes the evaluated TLS/HTTPS APIs provided by its platform to provide the TLS protocol and X509 validation whenever the client establishes a trusted channel to the UEM Server.

Guidance Assurance Activities: If 'internal lookup of TOE-managed certificate status' is selected, then the evaluator shall ensure the AGD documentation describes how issued certificates are reported as invalid.

Not applicable, as 'internal lookup of TOE-managed certificate status' is not selected.

Testing Assurance Activities: The tests described must be performed in conjunction with the other certificate services evaluation activities, including each of the functions in FIA_X509_EXT.2.1. The tests for the extendedKeyUsage rules are performed in conjunction with the uses that require those rules. The evaluator shall create a chain of at least four certificates: the node certificate to be tested, two Intermediate CAs, and the self-signed Root CA.

Test 1: The evaluator shall demonstrate that validating a certificate without a valid certification path results in the function failing, for each of the following reasons, in turn:

- by establishing a certificate path in which one of the issuing certificates is not a CA certificate,

- by omitting the basicConstraints field in one of the issuing certificates,

- by setting the basicConstraints field in an issuing certificate to have CA=False,

- by omitting the CA signing bit of the key usage field in an issuing certificate, and

- by setting the path length field of a valid CA field to a value strictly less than the certificate path.

The evaluator shall then establish a valid certificate path consisting of valid CA certificates, and demonstrate that the function succeeds. The evaluator shall then remove trust in one of the CA certificates, and show that the function fails.

Test 2: The evaluator shall demonstrate that validating an expired certificate results in the function failing.

Test 3: The evaluator shall test that the TOE can properly handle revoked certificates - conditional on whether CRL, OCSP, or OCSP stapling, or certificate status lookup is selected; if multiple methods are selected, then a test shall be performed for each method. The evaluator shall test revocation of the node certificate and revocation of the intermediate CA certificate (i.e. the intermediate CA certificate should be revoked by the root CA). If OCSP stapling per RFC 6066 is the only supported revocation method, testing revocation of the intermediate CA certificate is omitted. For FIA_X509_EXT.1.1(2) if included, if 'no revocation method' is selected, this test is omitted. The evaluator shall ensure that a valid certificate is used, and that the validation function succeeds. The evaluator then attempts the test with a certificate that has been revoked (for each method chosen in the selection) to ensure when the certificate is no longer valid that the validation function fails, if 'internal lookup of TOE-managed certificate status' is selected, then the evaluator shall follow AGD guidance to report the certificate as invalid.

Test 4: [conditional] If OCSP option is selected, the evaluator shall send the TOE an OCSP response signed by a certificate that does not have the OCSP signing purpose and verify that validation of the OCSP response fails. If CRL is selected, the evaluator shall cause a CA to sign a CRL with a certificate that has a key usage extension but does not have the cRLsign key usage bit set, and verify that validation of the CRL fails. If certificate status lookup is selected, this test is omitted. For FIA_X509_EXT.1.1(2) if included, if 'no revocation method' is selected, this test is omitted.

Test 5: The evaluator shall modify any byte in the first eight bytes of the certificate and demonstrate that the certificate fails to validate. (The certificate will fail to parse correctly).

Test 6: The evaluator shall modify any byte in the last byte of the certificate and demonstrate that the certificate fails to validate. (The signature on the certificate will not validate).

Test 7: The evaluator shall modify any byte in the public key of the certificate and demonstrate that the certificate fails to validate. (The signature on the certificate will not validate).

Test 8a: (Conditional on support for EC certificates as indicated in FCS_COP.1(3)). The evaluator shall establish a valid, trusted certificate chain consisting of an EC leaf certificate, an EC Intermediate CA certificate not designated as a trust anchor, and an EC certificate designated as a trusted anchor, where the elliptic curve parameters are specified as a named curve. The evaluator shall confirm that the TOE validates the certificate chain.

Test 8b: (Conditional on support for EC certificates as indicated in FCS_COP.1(3)). The evaluator shall replace the intermediate certificate in the certificate chain for Test 8a with a modified certificate, where the modified

intermediate CA has a public key information field where the EC parameters uses an explicit format version of the Elliptic Curve parameters in the public key information field of the intermediate CA certificate from Test 8a, and the modified Intermediate CA certificate is signed by the trusted EC root CA, but having no other changes. The evaluator shall confirm the TOE treats the certificate as invalid.

Note that since the UEM Android Client (agent) utilizes its host device for TLS and X509 functions, it is not subject to these test cases. However, this testing applies to both client and server implementations of the UEM Server where peer authentication is performed. Thus, the evaluator tested the implementations of x509 validation at the UEM Server syslog interface, the UEM Server LDAP interface, and the UEM Server's Agent communication interface.

Test 1: The evaluator attempted to successfully utilize each UEM Server x509 implementations using TLS communication with a test system, where the test system provided a good client certificate as a baseline. The evaluator used a network sniffer to capture the TLS session negotiations and observed the connection was successful. The evaluator attempted to make the same connections using certificates with a broken certificate path. The evaluator used a network sniffer to capture the TLS session negotiations and observed these connections were not successful. The evaluator then configured a server certificate with an invalid certification path by deleting the root CA so that the certificate chain was invalid because of a missing (or deleted) certificate. The connection was refused in each case. The evaluator then configured a test server to send an authentication certificate issued by a Sub CA with no BasicConstraints and with BasicConstraints but the CA Flag set to false and then with an intermediate CA with no keyCertSign purpose and then with an intermediate CA with a path length field set too low. The connection was refused in each case.

Test 2: The evaluator attempted to make a connection to each of the UEM Server x509 implementations from a test client using an expired client certificate and alternately connect from each implementation to a test server or client using an expired server certificate as a baseline. The evaluator used a network sniffer to capture the TLS session negotiations and observed the connection was not successful. The evaluator then configured a server or client certificate that had an expired subCA. The connection was refused in each case.

Test 3: The evaluator used a test client or server to attempt to connect to each of UEM Server x509 implementations. The test client/server presented a certificate during the TLS negotiation where the certificate was valid. A packet capture was obtained of this TLS negotiation which shows that the connection was successful. The evaluator revoked the test client/server certificate and attempted the same connection from the test client/server. The connection attempt performed after revoking the certificate was not successful.

Test 4: For the CRL, the evaluator configured a CRL provider to issue a CRL that was signed by a certificate that did not have the CRLsign Key usage bit set. The evaluator then attempted to connect to each of the UEM server x509 implementations from a test client/server where the certificate just described was used by the test client/server. The evaluator also attempted to connect to a test client/server for each implementation identified above where the test client/server was using the improper (an untrusted certificate) CRLsign certificate. All connection attempts failed with the TOE detecting the improper certificate being used to sign the CRL. This was repeated for OCSP and OCSP signing purpose with comparable results.

Test 5: The evaluator configured the test client/server to send an authentication certificate 1) that is valid, 2) that has one byte in the ASN1 field changed, 3) that has one byte in the certificate signature changed, and 4) that has one byte in the certificate public key changed, and attempted to make a connection from the test client to each of UEM Server x509 implementations and alternately connect from each of the implementations to a test client/server using that certificate. The evaluator used a network sniffer to capture the TLS session negotiations and observed each connection was not successful.

Test 6 - This test was performed with test 5.

Test 7 - This test was performed with test 5.

Test 8a and 8b: these conditional tests are not applicable, as the TOE does not claim support for any ECDSA ciphersuites.

2.3.5.2 MDMPP40:FIA_X509_EXT.1.2(1)

TSS Assurance Activities: If 'invoke platform-provided functionality' is selected:

The evaluator shall examine the TSS to verify that it describes (for each supported platform) how this functionality is invoked (it should be noted that this may be through a mechanism that is not implemented by the MDM Server; nonetheless, that mechanism will be identified in the TSS as part of this evaluation activity).

Section 6.3 of [ST] indicates that the UEM Server implements the validation and handling of X509 certificates and that the UEM Android Client invokes platform services which validate and handle X.509 certificates. It further explains that the UEM Android client invokes the evaluated TLS/HTTPS APIs provided by its platform to provide the TLS protocol and X509 validation whenever the client establishes a trusted channel to the UEM Server.

Guidance Assurance Activities: None Defined

Testing Assurance Activities: If 'implement functionality' is selected:

The tests described must be performed in conjunction with the other certificate services evaluation activities, including the functions in FIA_X509_EXT.2.1. The evaluator shall create a chain of at least four certificates: the node certificate to be tested, two Intermediate CAs, and the self-signed Root CA.

Test 1: The evaluator shall construct a certificate path, such that the certificate of the CA issuing the TOEs certificate does not contain the basicConstraints extension. The validation of the certificate path fails.

Test 2: The evaluator shall construct a certificate path, such that the certificate of the CA issuing the TOEs certificate has the cA flag in the basicConstraints extension not set. The validation of the certificate path fails.

Test 3: The evaluator shall construct a certificate path, such that the certificate of the CA issuing the TOEs certificate has the cA flag in the basicConstraints extension set to TRUE. The validation of the certificate path succeeds.

This testing applies to both client and server implementations in the UEM Server where peer authentication is performed. Thus, the evaluator tested the implementations of x509 validation at the UEM Server syslog interface, the UEM Server LDAP interface, and the UEM Server's Agent communication interface.

Test 1: The evaluator configured an issuing CA using a certificate without the basicConstraints extension. The evaluator then attempted to connections using a test system to each TLS implementation in the UEM Server. All connection attempts failed with the UEM Server detecting the improper certificate path.

Test 2: The evaluator configured an issuing CA using a certificate with the basicConstraints extension with the cA flag that is not set. The evaluator then attempted to connections using a test system to each TLS implementation in the UEM Server. All connection attempts failed with the UEM Server detecting the improper certificate path.

Test 3: The evaluator configured an issuing CA using a certificate with the basicConstraints extension with the cA flag that is set as TRUE. The evaluator then attempted to connections using a test system to each TLS implementation in the UEM Server. All connection attempts succeeded with the UEM Server accepting the certificate path.

Component TSS Assurance Activities: None Defined

Component Guidance Assurance Activities: None Defined

Component Testing Assurance Activities: None Defined

2.3.6 X.509 Certificate Authentication - per TD0641 (MDMPP40:FIA_X509_EXT.2)

2.3.6.1 MDMPP40:FIA_X509_EXT.2.1

TSS Assurance Activities: If 'invoke platform-provided functionality' is selected, the evaluator shall examine the TSS to verify that it describes (for each supported platform) how this functionality is invoked (it should be noted that this may be through a mechanism that is not implemented by the MDM Server; nonetheless, that mechanism will be identified in the TSS as part of this evaluation activity).

Section 6.3 of [ST] indicates that the TOE selected both "invoke platform-provided functionality" and "implement functionality" for its processing of x509 certificate validation. The UEM Server implements the validation and handling of X509 certificates and that the UEM Android Client invokes platform services which validate and handle X.509 certificates. This section explains that the UEM Android client invokes the evaluated TLS/HTTPS APIs

provided by its platform to provide the TLS protocol and X509 validation whenever the client establishes a trusted channel to the UEM Server.

Guidance Assurance Activities: None Defined

Testing Assurance Activities: None Defined

2.3.6.2 MDMPP40:FIA_X509_EXT.2.2

TSS Assurance Activities: The evaluator shall check the TSS to ensure that it describes how theTOE chooses which certificates to use, and any necessary instructions in the administrative guidance for configuring the operating environment so that the TOE can use the certificates.

If 'invoke platform-provided functionality' is selected, the evaluator shall examine the TSS to verify that it describes (for each supported platform) how this functionality is invoked (it should be noted that this may be through a mechanism that is not implemented by the MDM Server; nonetheless, that mechanism will be identified in the TSS as part of this evaluation activity).

If 'implement functionality' is selected, the evaluator shall examine the TSS to confirm that it describes the behavior of the TOE when a connection cannot be established during the validity check of a certificate used in establishing a trusted channel. The evaluator shall verify that any distinctions between trusted channels are described.

Section 6.2 of [ST] explains that the UEM Server can be configured with an X509v3 certificate to present during TLS negotiation with an audit (Syslog) server and an X509v3 certificate to present during TLS negotiation with an LDAP server. The certificates used by the UEM server for these communication channels are independent from one another and from the TOE's internal certificates used for communication with mobile devices. The administrator must load the certificate corresponding to each client communication channel, prior to installing the UEM server's client certificate for these channels.

Section 6.3 of [ST] explains that the UEM Android client invokes the evaluated TLS/HTTPS APIs provided by its platform to provide the TLS protocol and X509 validation whenever the client establishes a trusted channel to the UEM Server.

Section 6.3 also states that the UEM Server will not accept any certificate for which it cannot determine the validity and will reject the connection attempt.

Guidance Assurance Activities: If the requirement that the administrator is able to specify the default action is selected, then the evaluator shall ensure that the operational guidance contains instructions on how this configuration action is performed.

The requirement that the administrator is able to specify the default action is not selected.

Testing Assurance Activities: The evaluator shall perform the following test for each trusted channel:

Test 1: The evaluator shall demonstrate use of a valid certificate requiring certificate validation checking to be performed in at least some part by communicating with a non-TOE IT entity. The evaluator shall then manipulate the environment so that the TOE is unable to verify the validity of the certificate, and observe that the action selected in FIA_X509_EXT.2.2 is performed. If the selected action is administrator-configurable, then the evaluator shall follow the operational guidance to determine that all supported administrator-configurable options behave in their documented manner.

The evaluator performed successful connections (for LDAP, Syslog and Agent channels) using valid certificates and a reachable OCSP responders and found that connections were successful. The evaluator also attempted these connections when the OCSP responder was unreachable and found that The TOE will not accept any certificate for which it cannot determine the validity and rejected the connection attempt.

Component TSS Assurance Activities: None Defined

Component Guidance Assurance Activities: None Defined

Component Testing Assurance Activities: None Defined

2.3.7 X.509 Unique Certificate - per TD0754 (MDMPP40:FIA_X509_EXT.5)

2.3.7.1 MDMPP40:FIA_X509_EXT.5.1

TSS Assurance Activities: None Defined

Guidance Assurance Activities: None Defined

Testing Assurance Activities: None Defined

Component TSS Assurance Activities: None Defined

Component Guidance Assurance Activities: None Defined

Component Testing Assurance Activities: None Defined

2.4 Security Management (FMT)

2.4.1 Management of Functions Behavior (MDMPP40:FMT_MOF.1(1))

2.4.1.1 MDMPP40:FMT_MOF.1.1(1)

TSS Assurance Activities: None Defined

Guidance Assurance Activities: None Defined

Testing Assurance Activities: None Defined

Component TSS Assurance Activities: The evaluator shall examine the TSS and user documents to ensure that they describe what security management functions are restricted to the administrator and what actions can be taken for each management function. The evaluator shall verify that the security management functions are restricted to authorized administrators and the administrator is only able to take the actions as described in the user documents.

Section 6.4 of [ST] explains that the UEM Server component of the TOE restricts all security management functions (identified below for FMT_SMF.1(1)/FMT_SMF.1(2)/FMT_SMF.1(3)) to an authorized administrator. This section also contains a table of management functions that administrators may apply on devices.

Component Guidance Assurance Activities: None Defined

Component Testing Assurance Activities: Test 1: The evaluator shall attempt to access the functions and policies in FMT_SMF.1(1) as an unauthorized user and verify that the attempt fails.

Test 2: [conditional] The evaluator shall attempt to access the functions and policies in FMT_SMF.1(3) as an unauthorized user and verify that the attempt fails.

The evaluator attempted to access the functions and policies in FMT_SMF.1(1) as an unauthorized user and observed that the attempt failed.

The evaluator attempted to access the functions and policies in FMT_SMF.1(3) as an unauthorized user and observed that the attempt failed.

2.4.2 Management of Functions Behavior (Enrollment) (MDMPP40:FMT_MOF.1(2))

2.4.2.1 MDMPP40:FMT_MOF.1.1(2)

TSS Assurance Activities: None Defined

Guidance Assurance Activities: None Defined

Testing Assurance Activities: None Defined

Component TSS Assurance Activities: The evaluator shall examine the TSS and verify that it describes how unauthorized users are prevented from enrolling in the MDM services.

Section 6.4 of [ST] explains that the authorized administrator can enable mobile device users to enroll their mobile devices. An authorized administrator provides the mobile device user with a username and a password that will allow them to login in order to create an activation password to enroll their devices.

Component Guidance Assurance Activities: None Defined

Component Testing Assurance Activities: The test of this function is performed in conjunction with FIA_ENR_EXT.1.

The test of this function is performed in conjunction with FIA_ENR_EXT.1.

2.4.3 Management of Functions in (MAS Server Downloads) (MDMPP40:FMT_MOF.1(3))

2.4.3.1 MDMPP40:FMT_MOF.1.1(3)

TSS Assurance Activities: None Defined

Guidance Assurance Activities: None Defined

Testing Assurance Activities: None Defined

Component TSS Assurance Activities: The evaluator shall examine the TSS to determine that all methods of initiating an application download or update push are specified.

Section 6.4 bullet FMT_SMF.1(3) of [ST] explains that the UEM Server supports the configuration of application groups assigned to individual applications and devices. It also indicates that once an individual application is assigned to an app group, if it is marked as “Required”, the application is pushed to and installed on the mobile devices assigned to that app group. When assigned to the app group but not required, the users of the assigned mobile devices can use the client to download and install the application.

Component Guidance Assurance Activities: The evaluator shall confirm that the operational guidance contains how to initiate an application download or update push.

The section entitled "Assign an app to a user account" in [Admin] contains instructions to allow an administrator to control apps installed on a device, per user.

The section entitled “Assign an app to a user group” in [Admin] contains instructions to allow an administrator to control apps installed on an administrator-specified group of devices.

Component Testing Assurance Activities: The evaluator shall ensure that the MAS Server verifies that the mobile device is enrolled in the MDM Server and is in a compliant state. The evaluator shall verify that an application cannot be downloaded from the MAS Server prior to enrolling the device with the MDM. The evaluator shall partially enroll the mobile device, so the device is connected to the MDM Server, but is not compliant and verify that applications cannot be downloaded.

Using two mobile devices (one enrolled, one not enrolled) the evaluator assigned a test application to a newly created “testuser” account. Upon assigning the application to the user account, the evaluator observed that the application could be seen in the enrolled UEM client's work applications menu. The evaluator attempted to access the MAS server via the UEM application on the device that was not enrolled, but could not find any method of doing so.

The evaluator then attempted to partially enroll the mobile device to see if the client was granted access before the enrollment process completes. The evaluator found that the test application fails to open and displays a prompt, not letting the user proceed any further (including, to the MAS server) until enrollment was fully completed.

The evaluator observed that the device could have access to the applications on the MAS server only if it was enrolled and granted that permission.

2.4.4 Trusted Policy Update - per TD0754 (MDMPP40:FMT_POL_EXT.1)

2.4.4.1 MDMPP40:FMT_POL_EXT.1.1

TSS Assurance Activities: None Defined

Guidance Assurance Activities: None Defined

Testing Assurance Activities: None Defined

Component TSS Assurance Activities: The evaluator shall verify that the ST describes how policies are signed, to include whether the private key used for signing is associated with an X509 certificate or public key, the method for distributing the policy verification material (a certificate or provisioned public key) to the agent, and the method for distinguishing whether a policy is appropriate for an agent. If tokens are claimed in FMT_POL_EXT.1.3, the evaluator shall verify the ST describes how tokens are established and distributed to the agent. (TD0754 applied)

Section 6.4 of [ST] explains that each device policy is signed by the UEM server using an RSA certificate issued for that purpose. The UEM Android Client checks the signature of each policy is receives in order to ensure it is valid before application to the mobile device.

Component Guidance Assurance Activities: If applicable, the evaluator shall verify that the AGD guidance instructs administrators on configuring the Enterprise certificate to be used for signing policies or signing the policies before applying them.

Section 6.4 of [ST] indicates that each device policy is signed by the UEM server using an RSA certificate issued for that purpose. These certificates are generated by the UEM Server and are not imported. Therefore, no instructions are necessary in the administrative guidance.

Component Testing Assurance Activities: The evaluator shall perform a policy update in accordance withFMT_SMF.1(1). The evaluator shall examine the policy either at the MDM Server, in transmission, or at the MDM agent, and verify the TSF signs the update and provides it to the MDM Agent.

The evaluator performed a policy update in accordance with MDMPP40:FMT_SMF.1(1). The evaluator examined the policy at the MDM Server, and verified the UEM Server had signed the update and provided it to the MDM Agent.

2.4.5 Agent Trusted Policy Update - per TD0755 (MDMA10:FMT_POL_EXT.2)

2.4.5.1 MDMA10:FMT_POL_EXT.2.1

TSS Assurance Activities: None Defined

Guidance Assurance Activities: None Defined

Testing Assurance Activities: None Defined

2.4.5.2 MDMA10:FMT_POL_EXT.2.2

TSS Assurance Activities: None Defined

Guidance Assurance Activities: None Defined

Testing Assurance Activities: None Defined

Component TSS Assurance Activities: The evaluator ensures that the TSS describes how the candidate policies are obtained by the MDM Agent, the processing associated with verifying the digital signature of the policy updates,

and the actions that take place for successful (signature was verified) and unsuccessful (signature could not be verified) cases. The software components that are performing the processing must also be identified in the TSS and verified by the evaluators.

Section 6.4 of [ST] explains that the UEM Server utilizes TLS (with mutual/client authentication using X509 certificates) as the trusted channel to protect all data transmitted between the UEM Server and the UEM Android Clients. The UEM Android client periodically contacts the UEM Server. Once a trusted channel is established, the UEM server transfers applicable device policies and commands to the UEM Android client. Each device policy is signed by the UEM server using an RSA certificate issued for that purpose. The UEM Android Client checks the signature of each policy is receives in order to ensure it is valid before application to the mobile device. A policy with an invalid signature is not installed.

Component Guidance Assurance Activities: None Defined

Component Testing Assurance Activities: This evaluation activity is performed in conjunction with the evaluation activity for FIA_X509_EXT.1 and FIA_X509_EXT.2 as defined in the Base-PPs.

Test 1: The evaluator shall perform a policy update from an available configuration interface (such as through a test MDM Server). The evaluator shall verify the update is signed and is provided to the MDM Agent. The evaluator shall verify the MDM Agent accepts the digitally signed policy.

Test 2: The evaluator shall perform a policy update from an available configuration interface (such as through a test MDM Server). The evaluator shall provide an unsigned and an incorrectly signed policy to the MDM Agent. The evaluator shall verify the MDM Agent does not accept the digitally signed policy.

Test 1: The evaluator modified a policy that was applied to a mobile device update using the UEM Server. The evaluator verified the update was signed and was sent to the MDM Agent. The evaluator verified the MDM Agent accepted the digitally signed policy.

Test 2: The evaluator modified a policy that was applied to a mobile device update using the UEM Server. The evaluator ensured that the policy transmitted to the MDM agent had an invalid signature. The evaluator verified the MDM Agent accepted the digitally signed policy. The evaluator repeated this this activity sending an unsigned policy to the MDM agent.

2.4.6 Security Attribute Expiration (MDMPP40:FMT_SAE_EXT.1)

2.4.6.1 MDMPP40:FMT_SAE_EXT.1.1

TSS Assurance Activities: None Defined

Guidance Assurance Activities: None Defined

Testing Assurance Activities: None Defined

2.4.6.2 MDMPP40:FMT_SAE_EXT.1.2

TSS Assurance Activities: None Defined

Guidance Assurance Activities: None Defined

Testing Assurance Activities: None Defined

Component TSS Assurance Activities: The evaluator shall verify that the TSS contains a description of the process of enrollment for each MDM Agent/platform listed as supported in the ST. This description shall the method of user authentication (username/password, token, etc.).

Section 6.4 of [ST] explains that the UEM Server requires that any user attempting to enroll a mobile device authenticate to the server (through a TLS trusted channel). The UEM Android Client component of the TOE is configured with an X.509v3 certificate suitable to facilitate secure communication with the UEM Server. This certificate is provisioned (signed using a certificate issued by the UEM Server) during device enrollment. The UEM Server can issue activation passwords used to enroll mobile devices. An administrator can configure the time limit after which an activate password will expire and no longer be usable to enroll a mobile device.

Component Guidance Assurance Activities: The evaluator shall check to ensure that the operational guidance contains instructions to configure the expiration time for each method of user authentication listed in the TSS.

The section entitled "Set an activation password and send an activation email message" in [Admin] includes instructions to configure the expiration time for Activation emails.

Component Testing Assurance Activities: Test 1: The evaluator shall configure the MDM Server according to the administrative guidance to set an expiration time for the enrollment authentication data. For each method of user authentication listed in the TSS, the evaluator shall attempt to enroll using authentication data that has expired. The evaluator shall verify that enrollment was unsuccessful.

The evaluator obtained an activation email for a user and demonstrated the activation information (including the user's credentials) could be used to enroll a device. The evaluator waited for the activation email to become expired, verified it could no longer be viewed through the TOE, and attempted to use the expired email (which had been saved). The evaluator observed that the enrollment attempt using the expired email was unsuccessful.

2.4.7 Specification of Management Functions (Server configuration of Agent) (MDMPP40:FMT_SMF.1(1))

2.4.7.1 MDMPP40:FMT_SMF.1.1(1)

TSS Assurance Activities: None Defined

Guidance Assurance Activities: None Defined

Testing Assurance Activities: None Defined

Component TSS Assurance Activities: The evaluator shall examine the TSS to ensure that it describes each management function claimed. The evaluator shall examine the TSS to ensure that it identifies the management functions implemented for each supported MDM Agent/platform, which are likely to be subsets of all of the management functions available to the administrator on the MDM Server. The evaluator shall examine the TSS to verify that any differences between management functions and policies for each supported MDM Agent/platform are clearly indicated.

The evaluator shall determine if the Mobile Device has been evaluated. If so, the evaluator shall examine the TSS to verify that it clearly identifies which management functions match the selections and assignments of the evaluated Mobile Device and which management functions were not evaluated.

(TD0479 applied)

Section 6.4 of [ST] includes a Table 6-1, "Supported Device Management Commands and Policies". This table identifies the management functions which can be applied to Android or iOS devices using the TOE. Each management command and policy are identified as being applicable to a Samsung Android or iOS device. The scope of supported managed client devices for the evaluation is limited by the set of devices evaluated on the NIAP PCL. Refer to Section 1.4 of [ST] for additional information about the specific evaluated platforms).

Component Guidance Assurance Activities: None Defined

Component Testing Assurance Activities: For each MDM Agent/platform listed as supported in the ST:

Test 1: The evaluator shall verify the ability to command each MDM Agent functional capability and configure each MDM Agent policy listed above.

Test 1: The evaluator verified the UEM Server could send each command to the MDM Agent and could configure each MDM Agent policy listed in the requirement.

2.4.8 Specification of Management Functions (Server Configuration of Server) (MDMPP40:FMT_SMF.1(2))

2.4.8.1 MDMPP40:FMT_SMF.1.1(2)

TSS Assurance Activities: None Defined

Guidance Assurance Activities: None Defined

Testing Assurance Activities: None Defined

Component TSS Assurance Activities: The evaluator shall examine the TSS to ensure that it describes each management function listed. For function c.4, the evaluator shall examine the TSS to ensure that it describes the privacy-sensitive information that the TOE has the capability to collect from enrolled mobile devices.

Section 6.4 of [ST] indicates that the UEM Server component of the TOE supports the security management functions to configure and manage itself, including configuring a login banner. Among the available security management functions are the ability to configure X.509v3 certificates, manage the device registration process (enrolling specific devices by device serial numbers, limiting the number of devices a user can enroll and controlling the time available for enrollment). The UEM Server can also configure the administrator login session timeout value.

Component Guidance Assurance Activities: The evaluator shall verify the AGD guidance includes detailed instructions of what options are available and how to configure each management functional capability listed.

The sections entitled "Create an LDAP client certificate used for mutual authentication and connect to an LDAP directory" and "Set up export of server audit records to a syslog server" describe the process for installing the X.509v3 certificates used by the UEM Server.

The section entitled "Create an activation profile" provides instructions to configure limitations on device enrollment.

The section entitled "Configure management functions" describes how to configure the TOE unlock banner on devices.

The section entitled "Set an activation password and send an activation email message" indicates how to configure the length of time the enrollment authenticator is valid.

The section entitled “Set up export of server audit records to a syslog server", explains how to configure the transfer of UEM sever logs to a syslog server for storage, analysis, and reporting.

The section entitled "Set the session timeout limit" explains how to configure administrator login session timeouts.

Component Testing Assurance Activities: The tests of functions b, c.1, c.2, and c.5 are performed in conjunction with the use of the function. Test 3 also covers function c.4. The evaluator shall perform the following test:

Test 1: The evaluator shall configure the TSF authentication certificate(s) and verify that the correct certificate is used in established trusted connections (FPT_ITT.1(1), FPT_ITT.1(2), FTP_ITC.1(1), and FTP_TRP.1(2)).

Test 2: (conditional) The evaluator shall configure the periodicity for the assigned list of commands to the agent for several configured time periods and shall verify that the MDM Server performs the commands schedule.

Test 3: (conditional) The evaluator shall design and perform tests to demonstrate that the assigned function may be configured and that the intended behavior of the function is enacted by the MDM Server.

Test 1: The evaluator configured the TSF authentication certificate(s) and verified that the correct certificate was used in established trusted connections (FPT_ITT.1(2), FTP_ITC.1(1), and FTP_TRP.1(2)).

Test 2: This TOE does not allow configuration of the period for any events, so this conditional test was not applicable.

Test 3: The evaluator demonstrated that the assigned function may be configured and that the intended behavior of the function was enacted by the MDM Server.

2.4.9 Specification of Management Functions (MAS Server) (MDMPP40:FMT_SMF.1(3))

2.4.9.1 MDMPP40:FMT_SMF.1.1(3)

TSS Assurance Activities: None Defined

Guidance Assurance Activities: None Defined

Testing Assurance Activities: None Defined

Component TSS Assurance Activities: The evaluator shall examine the TSS to ensure that it describes each management function listed.

The evaluator shall examine the TSS to determine if the MAS Server creates its own groups or relies on the groups specified by the MDM Server.

The MDM and MAS server for this TOE are the same. Section 6.4 of [ST] explains that the administrator of the UEM Server can configure application groups that are assigned to individual applications and devices. The administrator also has the ability to download applications for deployment.

Component Guidance Assurance Activities: The evaluator shall confirm that the operational guidance contains how to create and define user groups and how to specify which applications are accessible by which group. The evaluator shall verify the AGD guidance includes detailed instructions of what options are available and how to configure each management functional capability listed.

The section entitled "Assign an app to a user group" in [Admin] provides instructions to create a user group and to specify an application that is accessible by that group.

Component Testing Assurance Activities: The evaluator shall ensure that the MAS client can only access the applications specified for the group they are enrolled in. The evaluator shall create a user group, making sure that the MAS client user is excluded from the group. Verify that an application accessible to that group cannot be accessed. The evaluator shall include the MAS client user in the group and assure that the application can be accessed.

The evaluator created an application group, assigned an application to that group, and assigned a user to be a member of that group. The evaluator then used a device belonging to the user and observed that the device could install the application. The evaluator attempted to install the application from a device belonging to a user that was not a member of the group. The evaluator observed the application could not be installed in this case.

2.4.10 Specification of Management Functions - per TD0755 (MDMA10:FMT_SMF_EXT.4)

2.4.10.1 MDMA10:FMT_SMF_EXT.4.1

TSS Assurance Activities: None Defined

Guidance Assurance Activities: None Defined

Testing Assurance Activities: None Defined

2.4.10.2 MDMA10:FMT_SMF_EXT.4.2

TSS Assurance Activities: None Defined

Guidance Assurance Activities: None Defined

Testing Assurance Activities: None Defined

Component TSS Assurance Activities: The evaluator shall verify that the any assigned functions are described in the TSS and that these functions are documented as supported by the platform. The evaluator shall examine the TSS to verify that any differences between management functions and policies for each supported mobile device are listed.

The evaluator shall verify that the TSS describes the methods in which the MDM Agent can be enrolled.

The TSS description shall make clear if the MDM Agent supports multiple interfaces for enrollment and configuration (for example, both remote configuration and local configuration).

The FMT_SMF_EXT.4/FMT_UNR_EXT.1 bullet in Section 6.4 of [ST] indicates that the UEM Android Client provides the ability to accept and implement the commands and policies identified as being sent by the UEM server. Since this is an MDM Agent module requirement, it applies only to the UEM Android Client.

Section 6.4 of [ST] describes that Enrollment of a device occurs requires an MDM administrator to define a device, user and password on the UEM Server. The UEM Android Client on that device is then used (by the mobile device user) to initiate an enrollment action which provides the user's ID and password, along with identification of the device.

Component Guidance Assurance Activities: The evaluator shall verify the AGD guidance includes detailed instructions for configuring each function in this requirement.

If the MDM Agent is a component of the MDM system (i.e. MDM Server is the Base-PP), the evaluator shall verify, by consulting documentation for the claimed mobile device platforms, that the configurable functions listed for this Agent are supported by the platforms.

If the MDM Agent supports multiple interfaces for configuration (for example, both remote configuration and local configuration), the AGD guidance makes clear whether some functions are restricted to certain interfaces.

The UEM Server issues certificates to the agent during enrollment, no administrative actions is required beyond the enrollment procedures.

The section entitled "Configure management functions" contains a table that explains how the UEM management functions can be used to provide the required MDMPP40:FMT_SMF.1(1) management functions.

Section entitled "Activate an Android device using a QR code" (for Android) and "Activating an iOS devices that are enrolled in DEP" (for iOS) describes the process for configuring the UEM Server to allow device enrollment.

Component Testing Assurance Activities: Test 1: In conjunction with the evaluation activities in theBase-PP, the evaluator shall attempt to configure each administrator-provided management function and shall verify that the mobile device executes the commands and enforces the policies.

Test 2: [conditional: if 'import the certificates to be used for authentication of MDM Agent communications' is selected in FMT_SMF_EXT.4.1] The evaluator shall configure the MDM Agent authentication certificate in accordance with the configuration guidance. The evaluator shall verify that the MDM Agent uses this certificate in performing the tests for FPT_ITT.1(2) (MDM as Base-PP) or FTP_ITC_EXT.1(2) (MDF as Base-PP). (TD0755 applied, supercedes TD0491)

Test 3: In conjunction with other evaluation activities, the evaluator shall attempt to enroll the MDM Agent in management with each interface identified in the TSS, and verify that the MDM Agent can manage the device and communicate with the MDM Server.

Test 4: [conditional] In conjunction with the evaluation activity for FAU_ALT_EXT.2.1, the evaluator shall configure the periodicity for reachability events for several configured time periods and shall verify that the MDM Server receives alerts on that schedule.

Test 5: [conditional] The evaluator shall design and perform tests to demonstrate that the assigned function may be configured and that the intended behavior of the function is enacted by the mobile device.

Test 1: The testing performed for FMT_SMF.1(1) address all of the required commands and management functions. Within that test case, not only are actions taken to demonstrate that the MDM server can perform the commands and management function, but steps within the test case also demonstrate that the commands and management function results in the expected change to the configuration of the mobile device.

Test 2: The evaluator enrolled a phone and collected network traffic immediately following the completion of the enrollment. Once enrolled, the UEM Android Client performed various interactions with the UEM Server to obtain policies and perform queued device commands. Inspection of this packet capture showed the certificate provided by the UEM Server to the UEM Android client during enrollment is the same certificate that is used by the Agent to authenticate to the UEM server after enrollment.

Test 3: These activities can be found as part of the testing of MDMPP40FTP_TRP.1.3(2), MDMPP40:FPT_ITT.1(2) and MDMPP40:FMT_SMF.1(1). In these tests the UEM Android Client is enrolled, obtains policies and commands and is used to perform various management function.

Test 4: This TOE does not allow configuration of the period for reachability events, so this conditional test was not applicable.

Test 5: The testing performed for FMT_SMF.1(1) address importing certificates as well as controlling the configuration and policies applied to mobile devices.

2.4.11 Security Management Roles (MDMPP40:FMT_SMR.1(1))

2.4.11.1 MDMPP40:FMT_SMR.1.1(1)

TSS Assurance Activities: None Defined

Guidance Assurance Activities: None Defined

Testing Assurance Activities: None Defined

2.4.11.2 MDMPP40:FMT_SMR.1.2(1)

TSS Assurance Activities: None Defined

Guidance Assurance Activities: None Defined

Testing Assurance Activities: None Defined

Component TSS Assurance Activities: The evaluator shall examine the TSS to verify that it describes the administrator role and the powers granted to and limitations of the role.

Section 6.4 of [ST] explains that the UEM Server provides several different roles: server primary administrators, security configuration administrators, device user administrators, auditor, and MD users. This section goes on to describe the powers available to these roles.

Component Guidance Assurance Activities: The evaluator shall review the operational guidance to ensure that it contains instructions for administering the TOE and which interfaces are supported.

The section entitled "Log in to BlackBerry UEM for the first time" in [Admin] provides instructions and the URL necessary for an administrator to access the UEM Web UI. This is the only available administrative interface.

Component Testing Assurance Activities: In the course of performing the testing activities for the evaluation, the evaluator shall use all supported interfaces, although it is not necessary to repeat each test involving an administrative action with each interface. The evaluator shall ensure, however, that each supported method of administering the TOE that conforms to the requirements of this PP be tested; for instance, if the TOE can be administered through a local hardware interface or HTTPS then both methods of administration must be exercised during the evaluation team's test activities.

The only remote interface offered by the TOE is the HTTPS protected Web UI. The evaluator connected to the UEM Server as an administrator using the Web UI for all configuration steps necessary during testing. The evaluator performed all management operations in support of testing using the Web UI (e.g., the steps in FTA_TAB.1 Test 1).

2.4.12 Security Management Roles (MAS Server) (MDMPP40:FMT_SMR.1(2))

2.4.12.1 MDMPP40:FMT_SMR.1.1(2)

TSS Assurance Activities: None Defined

Guidance Assurance Activities: None Defined

Testing Assurance Activities: None Defined

2.4.12.2 MDMPP40:FMT_SMR.1.2(2)

TSS Assurance Activities: None Defined

Guidance Assurance Activities: None Defined

Testing Assurance Activities: None Defined

Component TSS Assurance Activities: The evaluator shall examine the TSS to verify that it describes the administrator role and the powers granted to and limitations of the role.

The MAS Server functionality is integrated into the UEM Server and thus there are no additional roles.

Section 6.4 of [ST] explains that the UEM Server provides several different roles: server primary administrators, security configuration administrators, device user administrators, auditor, and MD users. This section goes on to describe the powers available to these roles.

Component Guidance Assurance Activities: The evaluator shall review the operational guidance to ensure that it contains instructions for administering the TOE and which interfaces are supported.

The UEM Server provides MAS server functionality, thus the section entitled "Log in to BlackBerry UEM for the first time" in [Admin] provides instructions and the URL necessary for an administrator to access the UEM Web UI. This is the only available administrative interface.

Component Testing Assurance Activities: In the course of performing the testing activities for the evaluation, the evaluator shall use all supported interfaces, although it is not necessary to repeat each test involving an administrative action with each interface. The evaluator shall ensure, however, that each supported method of administering the TOE that conforms to the requirements of this PP be tested; for instance, if the TOE can be administered through a local hardware interface or HTTPS then both methods of administration must be exercised during the evaluation team's test activities.

Because the MAS server and the MDM server are the same software entity, no further testing was performed beyond that described for FMT_SMR.1(1)-t1.

2.4.13 User Unenrollment Prevention (MDMA10:FMT_UNR_EXT.1)

2.4.13.1 MDMA10:FMT_UNR_EXT.1.1

TSS Assurance Activities: None Defined

Guidance Assurance Activities: None Defined

Testing Assurance Activities: None Defined

Component TSS Assurance Activities: The evaluator shall ensure that the TSS describes the mechanism used to prevent users from unenrolling or the remediation actions applied when unenrolled.

Section 6.4 of [ST] explains that the administrator can configure (using the UEM Console) the UEM Android Client to prevent the mobile phone's user from removing the client's administrative privileges, thus preventing the user from un-enrolling the client. If an administrator has not restricted the mobile phone user's ability to remove the UEM Android Client's administrative privileges, then the user can remove the UEM Android Client's administrative privileges (un-enrolling it from the UEM Server). Finally, the administrator can forcibly un-enroll the UEM Android Client from the UEM Server.

Component Guidance Assurance Activities: The evaluator shall ensure that the administrative guidance instructs administrators in configuring the unenrollment prevention in each available configuration interface. If any configuration allows users to unenroll, the guidance also describes the actions that unenroll the Agent.

The section "Configure management functions" in [Admin] includes instructions which can specify how an administrator can allow users to deactivate (unenroll) devices or prevent users from deactivating devices.

Component Testing Assurance Activities: Test 1: If 'prevent the unenrollment from occurring' is selected: The evaluator shall configure the Agent according to the administrative guidance for each available configuration interface, shall attempt to unenroll the device, and shall verify that the attempt fails.

Test 2: If 'apply remediation actions' is selected: If any configuration allows the user to unenroll, the evaluator shall configure the Agent to allow user unenrollment, attempt to unenroll, and verify that the remediation actions are applied.

Test 1: The evaluator configured a policy to prevent un-enrollment of a mobile device, ensured that the policy was applied to the MD, and attempted steps equivalent to un-enrollment (i.e., UEM Android Client uninstall). The evaluator observed that the un-enrollment was not successful.

Test 2: The ST does not make the selection for this conditional test.

2.5 Protection of the TSF (FPT)

2.5.1 Use of Supported Services and APIs (MDMPP40:FPT_API_EXT.1)

2.5.1.1 MDMPP40:FPT_API_EXT.1.1

TSS Assurance Activities: None Defined

Guidance Assurance Activities: None Defined

Testing Assurance Activities: None Defined

Component TSS Assurance Activities: The evaluator shall verify that the TSS lists the platform APIs used by the MDM software. The evaluator shall then compare the list with the supported APIs (available through e.g. developer accounts, platform developer groups) and ensure that all APIs listed in the TSS are supported.

Section 6.5 of [ST] explains that the proprietary list of platform APIs used by the TOE on the MDM server and MDM Agent is included in an Appendix A of the ST. The Appendix identifies a set of platform APIs used by the TOE and indicates while the API are all published API for the platform, the specific set of APIs used by the TOE is considered proprietary. The evaluator examined the API in the provided list and found all of the API in Appendix A to be published, and supported APIs offered by the platforms. None of the API in Appendix A were unpublished API.

Component Guidance Assurance Activities: None Defined

Component Testing Assurance Activities: None Defined

2.5.2 Internal TOE TSF Data Transfer (MDM Agent) (MDMPP40:FPT_ITT.1(2))

2.5.2.1 MDMPP40:FPT_ITT.1.1(2)

TSS Assurance Activities: None Defined

Guidance Assurance Activities: None Defined

Testing Assurance Activities: None Defined

Component TSS Assurance Activities: The evaluator shall examine the TSS to determine that the methods and protocols used to protect distributed TOE components are described. The evaluator shall also confirm that all protocols listed in the TSS in support ofTOE administration are consistent with those specified in the requirement, and are included in the requirements in the ST.

If 'invoke platform-provided functionality' is selected, the evaluator shall examine the TSS to verify that it describes (for each supported platform) how this functionality is invoked (it should be noted that this may be through a mechanism that is not implemented by the MDM Server; nonetheless, that mechanism will be identified in the TSS as part of this evaluation activity).

Section 6.5 of [ST] explains that the UEM Server utilizes TLS (with mutual/client authentication using X509 certificates) as the trusted channel to protect all data transmitted between the UEM Server and the UEM Android Clients. The UEM Server implements TLS, while the UEM Android client invokes platform APIs to utilize the evaluated TLS provided by its platform.

Component Guidance Assurance Activities: The evaluator shall confirm that the operational guidance contains instructions for establishing the communication paths for each supported method.

The TOE always uses TLS for communication between the UEM Android client and the UEM Server. No configuration is required.

Component Testing Assurance Activities: Test 1: The evaluator shall ensure that communications using each specified (in the operational guidance) communication method is tested during the course of the evaluation, setting up the connections as described in the operational guidance and ensuring that communication is successful.

Test 2: The evaluator shall ensure, for each method of communication, the channel data is not sent in plaintext.

Further evaluation activities are associated with the specific protocols.

The evaluator started a packet capture to collect traffic between the UEM Android Client on a mobile device (using TLS and X509 implemented in the platform device) and the UEM server components (using TLS directly implemented in the TOE). The evaluator then used the UEM Android Client to enroll the mobile device, push policies, download applications, and issue various commands. The evaluator reviewed the packet capture and determined that all network traffic associated with these actions was protected by TLS (test 1) and application data was not in plaintext (test 2).

2.5.3 Use of Third Party Libraries (MDMPP40:FPT_LIB_EXT.1)

2.5.3.1 MDMPP40:FPT_LIB_EXT.1.1

TSS Assurance Activities: None Defined

Guidance Assurance Activities: None Defined

Testing Assurance Activities: None Defined

Component TSS Assurance Activities: The evaluator shall verify that the TSS lists the libraries used by the MDM software. The evaluator shall verify that libraries found to be packaged with or employed by the MDM software are limited to those in the assignment.

The TOE utilizes the Certicom Security Builder® GSE-J Crypto Core 2.9.2 for secure TLS communication for each of its secure connections. The TOE also relies upon the MS SQL Driver to provide a communication pathway to the MS SQL server.

Component Guidance Assurance Activities: None Defined

Component Testing Assurance Activities: None Defined

2.5.4 Functionality Testing (MDMPP40:FPT_TST_EXT.1)

2.5.4.1 MDMPP40:FPT_TST_EXT.1.1

TSS Assurance Activities: None Defined

Guidance Assurance Activities: None Defined

Testing Assurance Activities: None Defined

2.5.4.2 MDMPP40:FPT_TST_EXT.1.2

TSS Assurance Activities: None Defined

Guidance Assurance Activities: None Defined

Testing Assurance Activities: None Defined

Component TSS Assurance Activities: If 'invoke platform-provided functionality' is selected, the evaluator shall examine the TSS to verify that it describes (for each supported platform) how this functionality is invoked (it should be noted that this may be through a mechanism that is not implemented by the MDM Server; nonetheless, that mechanism will be identified in the TSS as part of this evaluation activity).

If 'implement functionality' is selected, the evaluator shall examine the TSS to ensure that it details the self-tests that are run by the TSF on start-up; this description should include an outline of what the tests are actually doing (e.g., rather than saying 'memory is tested', a description similar to 'memory is tested by writing a value to each memory location and reading it back to ensure it is identical to what was written' shall be used). The evaluator shall ensure that the TSS makes an argument that the tests are sufficient to demonstrate that the TSF is operating correctly.

The evaluator shall examine the TSS to ensure that it describes how to verify the integrity of stored TSF executable code when it is loaded for execution. The evaluator shall ensure that the TSS makes an argument that the tests are sufficient to demonstrate that the integrity of stored TSF executable code has not been compromised. The evaluator also ensures that the TSS (or the operational guidance) describes the actions that take place for successful (e.g. hash verified) and unsuccessful e.g., hash not verified) cases.

Section 6.5 of [ST] indicates that the TOE runs the Certicom library performs power-up Known Answer Tests for each of its cryptographic algorithms (including AES, RSA, Diffie-Hellman, SHA, HMAC-SHA) to ensure correct operations. Section 6.5 also indicates that the TOE performs integrity checks for its executable code to ensure that the computed SHA-256 hash of the code during startup matches its expected value. Section 6.5 describes that should any self-test fail the cryptographic module will enter an error state and if the startup integrity check fails the UEM Server will fail to start. The combination of cryptographic tests, integrity tests and blocking startups upon failure is sufficient to prevent the TOE from executing if its software is corrupted.

Component Guidance Assurance Activities: None Defined

Component Testing Assurance Activities: The evaluator shall perform the following tests:

Test 1: The evaluator performs the integrity check on a known good TSF executable and verifies that the check is successful.

Test 2: The evaluator modifies the TSF executable, performs the integrity check on the modified TSF executable and verifies that the check fails.

Test 1: The evaluator restarted the TOE UEM Core services and found the following audit records in the audit trail. This indicates that the TOE did perform an integrity test when it started and passed the test.

Test 2: The evaluator shut-down the TOE, modified a TOE executable and started the TOE. The evaluator observed the TOE integrity check detect the modified executable and terminate the restart.

2.5.5 Trusted Update (MDMPP40:FPT_TUD_EXT.1)

2.5.5.1 MDMPP40:FPT_TUD_EXT.1.1

TSS Assurance Activities: None Defined

Guidance Assurance Activities: The evaluator shall ensure that the administrator guidance includes instructions for determining the current version of the TOE.

The section entitled "Locate the BlackBerry UEM version that you are using" in [Admin] provides instructions to determine the current version of the TOE.

Testing Assurance Activities: The evaluator shall query the TSF for the current version of the software according to the AGD guidance and shall verify that the current version matches that of the documented and installed version.

The evaluator logged in to the UEM Server as an administrator and observed on the “Help > About BlackBerrry UEM” screen the current version of the running UEM Server. The version displayed was consistent with the version being evaluated.

2.5.5.2 MDMPP40:FPT_TUD_EXT.1.2

TSS Assurance Activities: If 'invoke platform-provided functionality' is selected, the evaluator shall examine the TSS to verify that it describes (for each supported platform) how this functionality is invoked (it should be noted

that this may be through a mechanism that is not implemented by the MDM Server; nonetheless, that mechanism will be identified in the TSS as part of this evaluation activity).

Section 6.5 of [ST] indicates that an administrator must obtain updates and explicitly update the TOE using that update by using operations provided by the platform user interface. The UEM Server's updates are distributed in the form of a zip file. This zip file contains an executable that is digitally signed.

Guidance Assurance Activities: None Defined

Testing Assurance Activities: None Defined

2.5.5.3 MDMPP40:FPT_TUD_EXT.1.3

TSS Assurance Activities: If 'invoke platform-provided functionality' is selected, the evaluator shall examine the TSS to verify that it describes (for each supported platform) how this functionality is invoked (it should be noted that this may be through a mechanism that is not implemented by the MDM Server; nonetheless, that mechanism will be identified in the TSS as part of this evaluation activity).

If 'implement functionality' is selected, the evaluator shall examine the TSS and verify that it describes the standards by which the updates are digitally signed and how the signature verification process is implemented.

Section 6.5 of [ST] indicates that an administrator must obtain updates and explicitly update the TOE using that update. The UEM Server's updates are distributed in the form of a zip file. This zip file contains an executable that is digitally signed.

Guidance Assurance Activities: The evaluator shall examine the AGD guidance to verify that it describes how to query the current version of the TSF software, how to initiate updates and how to check the integrity of updates prior to installation.

The section entitled "Locate the BlackBerry UEM version that you are using" in [Admin] provides instructions to determine the current version of the TOE. The section entitled "Upgrade the BlackBerry UEM software" describes how to initiate updates. The steps provided cause the execution of the Extractor.exe program which verifies the integrity of the update.

Testing Assurance Activities: The evaluator shall perform the following tests:

Test 1: The evaluator shall attempt to initiate an update digitally signed by the vendor and verify that the update is successfully installed.

Test 2: The evaluator shall attempt to install an update not digitally signed by the vendor and verify that either the signature can be checked (allowing the update to be aborted) or the update is not installed.

Test 1: The evaluator followed guidance to initiate a TOE update referring to a properly signed update provided by the vendor. The update changed the TOE version and verified the new code.

Test 2: The evaluator removed the digital signature from a valid update and produced an unsigned update. Upon attempting to use the unsigned installation program the platform rejected the operation indicating a failure validating the Manifest Signature.

Component TSS Assurance Activities: None Defined

Component Guidance Assurance Activities: None Defined

Component Testing Assurance Activities: None Defined

2.6 TOE ACCESS (FTA)

2.6.1 DEFAULT TOE ACCESS BANNERS (MDMPP40:FTA_TAB.1)

2.6.1.1 MDMPP40:FTA_TAB.1.1

TSS Assurance Activities: None Defined

Guidance Assurance Activities: None Defined

Testing Assurance Activities: None Defined

Component TSS Assurance Activities: If 'invoke platform-provided functionality' is selected, the evaluator shall examine the TSS to verify that it describes (for each supported platform) how this functionality is invoked (it should be noted that this may be through a mechanism that is not implemented by the MDM Server; nonetheless, that mechanism will be identified in the TSS as part of this evaluation activity).

If 'implement functionality' is selected, the TSS shall describe when the banner is displayed.

Section 6.6 of [ST] explains that an administrator may configure a login notice to display whenever an administrator accesses the management console (i.e., Web UI). The notice can be configured to inform the administrator or user about any terms and conditions involved with using the interface. When configured to display, the administrator or user must click 'OK' before being allowed to log in.

Component Guidance Assurance Activities: The evaluator follows the operational guidance to configure a notice and consent warning message.

The section entitled "Create a login notice for the consoles" in [Admin] provides instructions to configure a login notice on the UEM Server.

Component Testing Assurance Activities: The evaluator shall also perform the following test: The evaluator shall start up or unlock the TSF. The evaluator shall verify that the notice and consent warning message is displayed in each instance described in the TSS.

The evaluator followed guidance to specify a TOE banner. The evaluator observed that the banner configured appeared in the login window for the UEM Server Web UI.

2.7 Trusted path/channels (FTP)

2.7.1 Inter-TSF Trusted Channel (Authorized IT Entities) (MDMPP40:FTP_ITC.1(1))

2.7.1.1 MDMPP40:FTP_ITC.1.1(1)

TSS Assurance Activities: If 'invoke platform-provided functionality' is selected, the evaluator shall examine the TSS to verify that it describes (for each supported platform) how this functionality is invoked (it should be noted that this may be through a mechanism that is not implemented by the MDM Server; nonetheless, that mechanism will be identified in the TSS as part of this evaluation activity).

Section 6.7 of [ST] indicates that the UEM Server implements TLS to secure communication with all enrolled clients or agents (including the Apple iOS agent and UEM Android Client) as well as external audit and LDAP authentication servers. This section also indicates that the UEM Server uses an IPsec channel implemented in its host Windows Server 2016 or Windows Server 2019 operating system for all communication with its SQL database server.

Guidance Assurance Activities: None Defined

Testing Assurance Activities: None Defined

2.7.1.2 MDMPP40:FTP_ITC.1.2(1)

TSS Assurance Activities: If 'invoke platform-provided functionality' is selected, the evaluator shall examine the TSS to verify that it describes (for each supported platform) how this functionality is invoked (it should be noted

that this may be through a mechanism that is not implemented by the MDM Server; nonetheless, that mechanism will be identified in the TSS as part of this evaluation activity).

Section 6.7 of [ST] indicates that the UEM Server implements TLS to secure communication with all enrolled clients or agents (including the Apple iOS agent and UEM Android Client) as well as external audit and LDAP authentication servers. Section 6.7 also indicates that the UEM Server uses an IPsec channel implemented in its host Windows Server 2016 or Windows Server 2019 operating system for all communication with its SQL database server.

The UEM Server is a TLS client when communicating with the external audit and LDAP authentication servers and an IPSec peer when communicating with the SQL database.

Guidance Assurance Activities: None Defined

Testing Assurance Activities: None Defined

2.7.1.3 MDMPP40:FTP_ITC.1.3(1)

TSS Assurance Activities: The evaluator shall examine the TSS to determine that the methods of communication with authorized IT entities are indicated, along with how those communications are protected.

If 'invoke platform-provided functionality' is selected, the evaluator shall examine the TSS to verify that it describes (for each supported platform) how this functionality is invoked (it should be noted that this may be through a mechanism that is not implemented by the MDM Server; nonetheless, that mechanism will be identified in the TSS as part of this evaluation activity).

Section 6.7 of [ST] indicates that the UEM Server implements TLS to secure communication with all enrolled clients or agents (including the Apple iOS agent and UEM Android Client) as well as external audit and LDAP authentication servers. Section 6.7 also indicates that the UEM Server uses an IPsec channel implemented in its host Windows Server 2016 or Windows Server 2019 operating system for all communication with its SQL database server.

The UEM Server is a TLS client when communicating with the external audit and LDAP authentication servers and an IPSec peer when communicating with the SQL database.

Guidance Assurance Activities: The evaluator shall confirm that the operational guidance contains instructions for configuring the communication channel between the MDM Server and authorized IT entities for each supported method.

The TOE implements trusted channels to an LDAP and syslog server, while relying upon its OE to provide IPSec protected communication with an SQL server.

The section entitled "Create an LDAP client certificate used for mutual authentication and connect to an LDAP directory" in [Admin] describes how to define and configure the LDAP communications to use TLS.

The UEM server provides connectivity to syslog for UEM Server security audits and for device audits. While these syslog servers can be the same or different, they must both be configured. The [Admin] document describes how to configure both.

The sections entitled "Set up export of server audit records to a syslog server" and "Set up export of device audit records to a syslog server" describe the steps to load the trusted root for a syslog server. These sections include instructions to update and execute the configurations scripts from Appendix A and B. Appendix A and B contain additional configuration scripts that are used during TOE installation and TOE updates. These scripts contain values which cause the TOE to use TLS and to provide mutual authentication to external syslog services as requested. These scripts also indicate the reference identifier port being used by the syslog server.

Section "Validate and install the BlackBerry UEM software" of [Admin] provides instructions to install the system using a method that will ensure that the TOE is using TLS and certificates in a manner that confirms to the description in [ST]. That is, no cryptographic configuration of key generation, key establishment, encryption, signature algorithms or hashing (e.g., choosing ciphersuites) can be performed on the UEM Server following installation.

The TOE relies upon its operating environment to provide IPsec protection for communication between the UEM Server and its SQL database. The section entitled "Validate and install the BlackBerry UEM software" explains that if the SQL server is installed on a machine other than the one running the UEM Server, then the platform IPsec should be used to protect this communication.

Testing Assurance Activities: Test 1: The evaluators shall ensure that communications using each specified (in the operational guidance) communication method is tested during the course of the evaluation, setting up the connections as described in the operational guidance and ensuring that communication is successful.

Test 2: The evaluator shall ensure, for each method of communication, the channel data is not sent in plaintext.

Test 3: The evaluator shall ensure, for each communication channel with the MDM Server, that a protocol analyzer identifies the traffic as the protocol under testing.

Further evaluation activities are associated with the specific protocols.

Test 1: A successful syslog and LDAP connect was established as part of the PKGTLS11:FCS_TLSC_EXT.1.2-t1 test. The evaluator collected network traffic demonstrating both trusted channels are successfully negotiated using TLS. The evaluator also captured network traffic showing that traffic between the UEM Sever and SQL database was protected by IPsec.

Test 2: The network traffic showed that application data for syslog, LDAP and SQL was not in plain text (MDMPP40:FTP_ITC.1(1)-t2).

Test 3: The network traffic showed that application data was protected using TLS as claimed by the [ST] for syslog and LDAP connections, while application data for communication with the SQL database was protected by IPsec.

Component TSS Assurance Activities: None Defined

Component Guidance Assurance Activities: None Defined

Component Testing Assurance Activities: None Defined

2.7.2 Inter-TSF Trusted Channel (MDM Agent) (MDMPP40:FTP_ITC.1(2))

2.7.2.1 MDMPP40:FTP_ITC.1.1(2)

TSS Assurance Activities: If 'invoke platform-provided functionality' is selected, the evaluator shall examine the TSS to verify that it describes (for each supported platform) how this functionality is invoked (it should be noted that this may be through a mechanism that is not implemented by the MDM Server; nonetheless, that mechanism will be identified in the TSS as part of this evaluation activity).

Section 6.7 of [ST] explains that the UEM Server uses implements TLS to secure communication with all enrolled clients or agents (including the Apple iOS agent and UEM Android Client).

Guidance Assurance Activities: None Defined

Testing Assurance Activities: None Defined

2.7.2.2 MDMPP40:FTP_ITC.1.2(2)

TSS Assurance Activities: If 'invoke platform-provided functionality' is selected, the evaluator shall examine the TSS to verify that it describes (for each supported platform) how this functionality is invoked (it should be noted that this may be through a mechanism that is not implemented by the MDM Server; nonetheless, that mechanism will be identified in the TSS as part of this evaluation activity).

Section 6.7 of [ST] explains that the UEM Server uses implements TLS to secure communication with all enrolled clients or agents (including the Apple iOS agent and UEM Android Client).

Guidance Assurance Activities: None Defined

Testing Assurance Activities: None Defined

2.7.2.3 MDMPP40:FTP_ITC.1.3(2)

TSS Assurance Activities: The evaluator shall examine the TSS to determine that the methods of Agent-Server communication are indicated, along with how those communications are protected. The evaluator shall also confirm that all protocols listed in the TSS in support of remote TOE administration are consistent with those specified in the requirement, and are included in the requirements in the ST.

If 'invoke platform-provided functionality' is selected, the evaluator shall examine the TSS to verify that it describes (for each supported platform) how this functionality is invoked (it should be noted that this may be through a mechanism that is not implemented by the MDM Server; nonetheless, that mechanism will be identified in the TSS as part of this evaluation activity).

Section 6.7 of [ST] explains that the UEM Server uses implements TLS to secure communication with all enrolled clients or agents (including the Apple iOS agent and UEM Android Client).

Guidance Assurance Activities: The evaluator shall confirm that the operational guidance contains instructions for configuring the communication channel between the MDM Agent and the MDM Server for each supported method.

The TOE protects communication between the UEM server and MDM agents using TLS by default. Upon successful installation, the TOE will be in a configuration that is compliant with the use of TLS and cryptographic features to protect MDM server to MDM agent communication. The section entitled, "Set up export of server audit records to a syslog server" and "Set up export of device audit records to a syslog server" in [Admin] describes the installation process that configures the TOE to protect syslog communication with TLS. The section entitled, "Create an LDAP client certificate used for mutual authentication and connect to an LDAP directory" in [Admin] describes the installation process that configures the TOE to protect LDAP communication with TLS.

Testing Assurance Activities: Test 1: The evaluators shall ensure that communications using each specified (in the operational guidance) Agent-Server communication method is tested during the course of the evaluation, setting up the connections as described in the operational guidance and ensuring that communication is successful.

Test 2: The evaluator shall ensure, for each method of Agent-Server communication, the channel data is not sent in plaintext.

Test 3: The evaluator shall ensure, for each communication channel with the MDM Server, that a protocol analyzer identifies the traffic as the protocol under testing.

Further evaluation activities are associated with the specific protocols.

For this requirement the evaluator exercised the connection between the UEM Server and the iOS agent. Communications between the UEM Server and the UEM Android Client are addressed by FPT_ITT.1(2).

Test 1: The TLS protocol thoroughly tested as part of the corresponding protocol requirements (e.g., FCS_TLSS_EXT.1). The TLS connection between the Apple iOS agent and UEM Server was demonstrated by the FIA_ENR_EXT.1 requirement. During this test activity, the evaluator captured network traffic associated with the Apple iOS agent enrolling with the UEM Server and observed that this traffic was protected by TLS.

Test 2: The test references network traffic that was collected while testing the Syslog and LDAP servers in FCS_TLSC_EXT.1. These packet captures show that the TLS application data was not sent in plaintext. The evaluator also demonstrated a connection between the MDM Server and SQL database, where the connection was protected using the Windows platform-provided IPsec. The evaluator demonstrated that, once IPsec was configured, the traffic between the TOE and SQL server were not sent in plaintext.

Test 3: The network traffic showed that application data between the Apple iOS agent and UEM Server was protected using TLS as claimed by the [ST].

Component TSS Assurance Activities: None Defined

Component Guidance Assurance Activities: None Defined

Component Testing Assurance Activities: None Defined

2.7.3 Trusted Channel (MDMPP40:FTP_ITC_EXT.1)

2.7.3.1 MDMPP40:FTP_ITC_EXT.1.1

TSS Assurance Activities: None Defined

Guidance Assurance Activities: None Defined

Testing Assurance Activities: None Defined

Component TSS Assurance Activities: The evaluator shall ensure that the TSS contains whether the MDM Server communication channel is internal or external to the TOE.

Section 6.7 of [ST] explains that the UEM Server uses TLS to secure communication with all enrolled clients or agents (including the Apple iOS agent and UEM Android Client) as well as external audit and LDAP authentication servers. Since the UEM Server provides the MAS server functionality, there is no additional communication path for the MAS audit communications.

Component Guidance Assurance Activities: None Defined

Component Testing Assurance Activities: This testing can be completed in conjunction with the testing for FPT_ITT.1(1)/FPT_ITT.1(2), FTP_ITC.1(2) or FTP_ITC.1(3).

This testing was completed in conjunction with the testing for FTP_ITC.1(1) and FTP_ITC.1(2).

2.7.4 Trusted Path (for Remote Administration) (MDMPP40:FTP_TRP.1(1))

2.7.4.1 MDMPP40:FTP_TRP.1.1(1)

TSS Assurance Activities: If 'invoke platform-provided functionality' is selected, the evaluator shall examine the TSS to verify that it describes (for each supported platform) how this functionality is invoked (it should be noted that this may be through a mechanism that is not implemented by the MDM Server; nonetheless, that mechanism will be identified in the TSS as part of this evaluation activity).

Section 6.7 of [ST] indicates that the UEM Server implements HTTPS as its trusted communication path for communications and remote Administrators must connect to the UEM Server using HTTPS (through a normal web browser) to securely administer the UEM Server. The UEM Server provides no other mechanism or method beyond HTTPS for a remote Administrator to configure or access the UEM Server.

Guidance Assurance Activities: None Defined

Testing Assurance Activities: None Defined

2.7.4.2 MDMPP40:FTP_TRP.1.2(1)

TSS Assurance Activities: If 'invoke platform-provided functionality' is selected, the evaluator shall examine the TSS to verify that it describes (for each supported platform) how this functionality is invoked (it should be noted that this may be through a mechanism that is not implemented by the MDM Server; nonetheless, that mechanism will be identified in the TSS as part of this evaluation activity).

As described by section 6.7 of [ST], the UEM Server implements a web-based user interface (Web UI) for remote administration. This Web UI is accessible through HTTPS/TLS. Because the TOE includes an HTTPS server side interface, the FCS_HTTPS_EXT.1 and FCS_TLSS_EXT.1 requirements are included in the [ST].

Guidance Assurance Activities: None Defined

Testing Assurance Activities: None Defined

2.7.4.3 MDMPP40:FTP_TRP.1.3(1)

TSS Assurance Activities: The evaluator shall examine the TSS to determine that the methods of remote TOE administration are indicated, along with how those communications are protected. The evaluator shall also confirm that all protocols listed in the TSS in support of TOE administration are consistent with those specified in the requirement, and are included in the requirements in the ST.

If 'invoke platform-provided functionality' is selected, the evaluator shall examine the TSS to verify that it describes (for each supported platform) how this functionality is invoked (it should be noted that this may be through a mechanism that is not implemented by the MDM Server; nonetheless, that mechanism will be identified in the TSS as part of this evaluation activity).

As described by section 6.7 of [ST], the UEM Server implements a web-based user interface (Web UI) for remote administration. This Web UI is accessible through HTTPS/TLS. Because the TOE includes an HTTPS server-side interface, the FCS_HTTPS_EXT.1 and FCS_TLSS_EXT.1 requirements are included in the [ST].

Guidance Assurance Activities: The evaluator shall confirm that the operational guidance contains instructions for establishing the remote administrative sessions for each supported method.

The TOE protects communication between the UEM server and remote administrators using HTTPS/TLS on its Web UI by default. Upon successful installation, the TOE will be in a configuration that is compliant with the use of TLS and cryptographic features to protect remote administration of the UEM Server Web UI. The relevant sections of [Admin] that describe installation can be found in the Guidance Assurance Activity for FCS_TLSS_EXT.1.1.

Testing Assurance Activities: The evaluator shall also perform the following tests:

Test 1: The evaluators shall ensure that communications using each specified (in the operational guidance) remote administration method is tested during the course of the evaluation, setting up the connections as described in the operational guidance and ensuring that communication is successful.

Test 2: For each method of remote administration supported, the evaluator shall follow the operational guidance to ensure that there is no available interface that can be used by a remote user to establish remote administrative sessions without invoking the trusted path.

Test 3: The evaluator shall ensure, for each method of remote administration, the channel data is not sent in plaintext.

Further evaluation activities are associated with the specific protocols.

Test 1: The evaluator connected to the UEM Server as an administrator using all claimed remote interface (i.e., HTTPS/TLS). The evaluator performed at one management activity using that interface. The evaluator collected the network traffic associated with this action that was transmitted between the UEM Server and remote administrator (using a packet capture tool). The evaluator examined the collected network traffic and observed that the traffic was protected from disclosure and modification by TLS v1.2 (and that traffic was encrypted).

Test 2: As a result of a review of guidance documentation, the evaluator identified no path for administration other than the Web UI and console access to the host servers (e.g., via RDP or direct console access).

Test 3: Review of the packet capture from test 1 above, showed that the traffic between the UEM Server and the remote administrator was not plaintext.

Component TSS Assurance Activities: None Defined

Component Guidance Assurance Activities: None Defined

Component Testing Assurance Activities: None Defined

2.7.5 Trusted Path (for Enrollment) (MDMPP40:FTP_TRP.1(2))

2.7.5.1 MDMPP40:FTP_TRP.1.1(2)

TSS Assurance Activities: If 'invoke platform-provided functionality' is selected, the evaluator shall examine the TSS to verify that it describes (for each supported platform) how this functionality is invoked (it should be noted that this may be through a mechanism that is not implemented by the MDM Server; nonetheless, that mechanism will be identified in the TSS as part of this evaluation activity).

Section 6.7 of [ST] indicates that the UEM Server implements a TLS trusted communication channel for all communications with MD users. MD users initiate the communication channel by login via the UEM Android Client (or other client) and thereafter all communications between the Client (on behalf of the MD user) and the UEM Server travel across the secure channel.

Guidance Assurance Activities: None Defined

Testing Assurance Activities: None Defined

2.7.5.2 MDMPP40:FTP_TRP.1.2(2)

TSS Assurance Activities: If 'invoke platform-provided functionality' is selected, the evaluator shall examine the TSS to verify that it describes (for each supported platform) how this functionality is invoked (it should be noted that this may be through a mechanism that is not implemented by the MDM Server; nonetheless, that mechanism will be identified in the TSS as part of this evaluation activity).

Section 6.7 of [ST] indicates that the UEM Server implements a TLS trusted communication channel for all communications with MD users. MD users initiate the communication channel by login via the UEM Android Client

(or other client) and thereafter all communications between the Client (on behalf of the MD user) and the UEM Server travel across the secure channel.

Guidance Assurance Activities: None Defined

Testing Assurance Activities: None Defined

2.7.5.3 MDMPP40:FTP_TRP.1.3(2)

TSS Assurance Activities: The evaluator shall examine the TSS to determine that the methods of remote enrollment are indicated, along with how those communications are protected. The evaluator shall also confirm that all protocols listed in the TSS in support of enrollment are consistent with those specified in the requirement, and are included in the requirements in the ST.

If 'invoke platform-provided functionality' is selected, the evaluator shall examine the TSS to verify that it describes (for each supported platform) how this functionality is invoked (it should be noted that this may be through a mechanism that is not implemented by the MDM Server; nonetheless, that mechanism will be identified in the TSS as part of this evaluation activity).

Section 6.7 of [ST] indicates that the TOE implements a TLS connection with the UEM Android client (MDM agent) for enrollment. Because the TOE includes an TLS server-side interface, the FCS_TLSS_EXT.1 and FCS_TLSS_EXT.2 requirements are included in [ST].

Guidance Assurance Activities: The evaluator shall confirm that the operational guidance contains instructions for establishing the enrollment sessions for each supported method.

The TOE protects communication between the UEM server and MDM agents using TLS by default. Upon successful installation, the TOE will be in a configuration that is compliant with the use of TLS and cryptographic features to protect MDM server to MDM agent communication. The relevant sections of [Admin] that describe installation can be found in the Guidance Assurance Activity for FCS_TLSC_EXT.1.1.

Testing Assurance Activities: For each MDM Agent/platform listed as supported in the ST:

Test 1: The evaluators shall ensure that communications using each specified (in the operational guidance) enrollment method is tested during the course of the evaluation, setting up the connections as described in the operational guidance and ensuring that communication is successful.

Test 2: For each method of enrollment supported, the evaluator shall follow the operational guidance to ensure that there is no available interface that can be used by a remote user to establish enrollment sessions without invoking the trusted path.

Test 3: The evaluator shall ensure, for each method enrollment, the channel data is not sent in plaintext.

Further evaluation activities are associated with the specific protocols.

Test 1-3: The communication actions performed between an agent (both UEM Android client and iOS agent) and the UEM Server for enrollment are demonstrated by test MDMPP40:FTP_TRP.1.3(2)-t1. In those test cases the packet captures show:

- the use of TLS for enrollment for both Android and iOS,
- that the TLS version is version 1.2, and
- that data transmitted through TLS for enrollment is not plaintext.

Test 2 & 3: The evaluator followed guidance and found no obvious ways to enroll outside of TLS. Test case MDMPP40:FTP_TRP.1.3(2)-t2 shows the enrollment interfaces presented by the client apps and concludes there was no alternate method of enrollment. The evaluator found that the packet capture of the enrollment showed the enrollment was TLS and was not plaintext.

Component TSS Assurance Activities: None Defined

Component Guidance Assurance Activities: None Defined

Component Testing Assurance Activities: None Defined

3. Protection Profile SAR Assurance Activities

3.1 Development (ADV)

3.1.1 Basic Functional Specification (ADV_FSP.1)

Assurance Activities: There are no specific evaluation activities associated with these SARs, except ensuring the information is provided. The functional specification documentation is provided to support the evaluation activities described in Section 5 and the relevant appendices, and other activities described for AGD, ATE, and AVA SARs. The requirements on the content of the functional specification information is implicitly assessed by virtue of the other evaluation activities being performed; if the evaluator is unable to perform an activity because there is insufficient interface information, then an adequate functional specification has not been provided.

Since functional specification documentation is provided to support the evaluation activities described in the claimed Protection Profiles (as well as packages and/or modules), the successful completion of assurance activities associated with those evaluation activities implies an adequate FSP has been provided.

3.2 Guidance documents (AGD)

3.2.1 Operational User Guidance (AGD_OPE.1)

Assurance Activities: Some of the contents of the operational guidance will be verified by the evaluation activities in Sections 4.2, 4.3, and 4.4 and evaluation of the TOE according to the CEM. The following additional information is also required.

If cryptographic functions are provided by the TOE, the operational guidance shall contain instructions for configuring the cryptographic engine associated with the evaluated configuration of the TOE. It shall provide a warning to the administrator that use of other cryptographic engines was not evaluated nor tested during the CC evaluation of the TOE.

The documentation must describe the process for verifying updates to the TOE by verifying a digital signature - this may be done by the TOE or the underlying platform.

The evaluator shall verify that this process includes the following steps:

Instructions for obtaining the update itself. This should include instructions for making the update accessible to the TOE (e.g., placement in a specific directory).

Instructions for initiating the update process, as well as discerning whether the process was successful or unsuccessful. This includes generation of the hash/digital signature.

The TOE will likely contain security functionality that does not fall in the scope of evaluation under this PP. The operational guidance shall make it clear to an administrator which security functionality is covered by the evaluation activities.

The [Admin] documents provide information to allow device users and administrators to operate the TOE features properly. The additional information specific to this assurance activity is identified by the following list.

a) The section entitled "Validate and install the BlackBerry UEM software" contains information about how to install the TOE, while subsequent sections describe how to configure the TOE to used TLS to protect LDAP and syslog.
b) The section entitled "Upgrade the BlackBerry UEM software" explains the steps necessary to perform an update of the TOE.
c) The section entitled "Server and device requirements" states that the configuration described throughout this guidance document explains how to configure and use the system in a manner that is compliant with the Security Target.

3.2.2 Preparative Procedures (AGD_PRE.1)

Assurance Activities: As indicated in the introduction above, there are significant expectations with respect to the documentation, especially when configuring the operational environment to support TOE functional requirements. The evaluator shall check to ensure that the guidance provided for the TOE adequately addresses all platforms claimed for theTOE in the ST.

Detailed step-by-step instructions for installing the TOE are provided by section of [Admin] referenced throughout the preceding Guidance Assurance Activities.

3.3 Life-cycle support (ALC)

3.3.1 Labelling of the TOE (ALC_CMC.1)

Assurance Activities: The evaluator shall check the ST to ensure that it contains an identifier (such as a product name/version number) that specifically identifies the version that meets the requirements of the ST. Further, the evaluator shall check the AGD guidance and TOE samples received for testing to ensure that the version number is consistent with that in the ST. If the vendor maintains a website advertising theTOE, the evaluator shall examine the information on the website to ensure that the information in the ST is sufficient to distinguish the product.

The evaluator verified that the ST, TOE and Guidance are all labelled with the same software name and version. The information is specific enough to procure the TOE and it includes hardware models and software versions. The

evaluator checked the TOE software version and hardware identifiers during testing by examining the actual machines that were used for testing.

3.3.2 TOE CM Coverage (ALC_CMS.1)

Assurance Activities: The 'evaluation evidence required by the SARs' in this PP is limited to the information in the ST coupled with the guidance provided to administrators and users under the AGD requirements. By ensuring that the TOE is specifically identified and that this identification is consistent in the ST and in the AGD guidance (as done in the evaluation activity for ALC_CMC.1), the evaluator implicitly confirms the information required by this component.

Life-cycle support is targeted aspects of the developer's life-cycle and instructions to providers of applications for the developer's devices, rather than an in-depth examination of the TSF manufacturer's development and configuration management process. This is not meant to diminish the critical role that a developer's practices play in contributing to the overall trustworthiness of a product; rather, it's a reflection on the information to be made available for evaluation.

The evaluator shall ensure that the TSF is uniquely identified (with respect to other products from the TSF vendor), and that documentation provided by the developer in association with the requirements in the ST is associated with the TSF using this unique identification.

See 3.3.1 for an explanation of how all CM items are addressed.

The TSF is uniquely identified as BlackBerry Unified Endpoint Management (UEM) Server and Android Client version 12.19.

3.4 Tests (ATE)

3.4.1 Independent Testing - Conformance (ATE_IND.1)

Assurance Activities: The evaluator shall prepare a test plan and report documenting the testing aspects of the system. The test plan covers all of the testing actions contained in the CEM and the body of this PP's Evaluation Activities. While it is not necessary to have one test case per test listed in an evaluation activity, the evaluator must document in the test plan that each applicable testing requirement in the ST is covered.

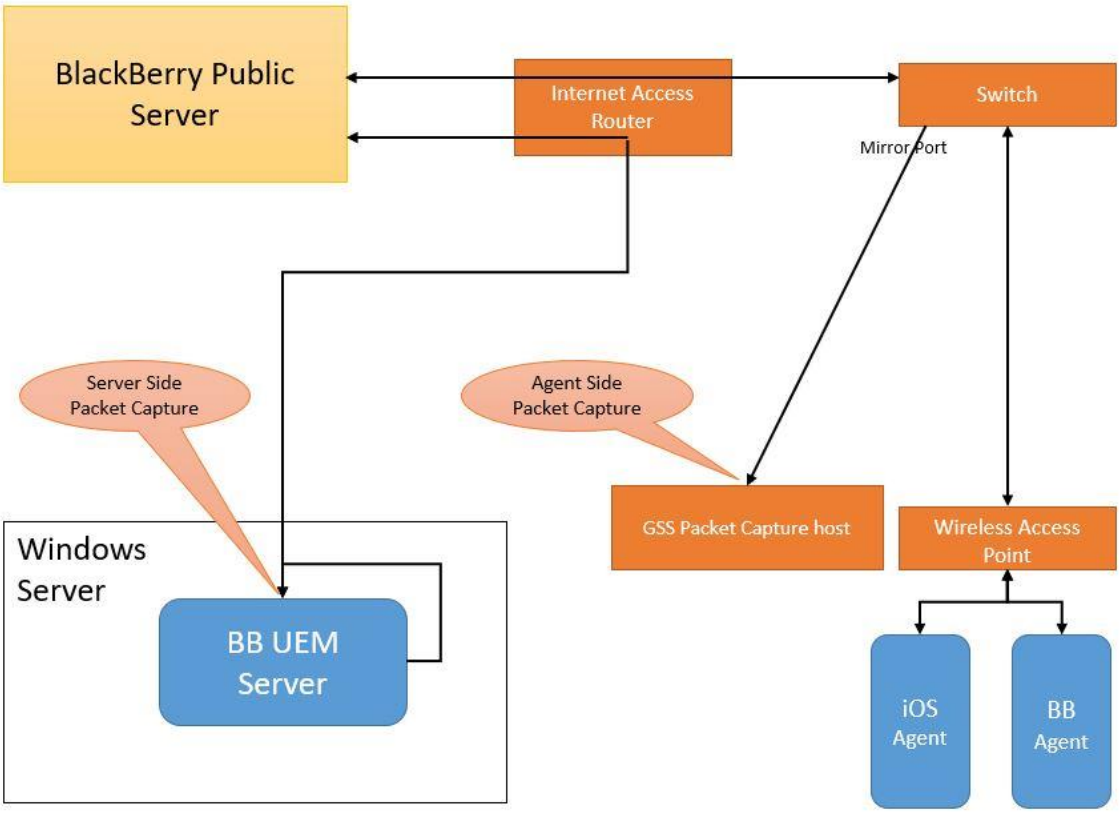
The test plan identifies the platforms to be tested, and for those platforms not included in the test plan but included in the ST, the test plan provides a justification for not testing the platforms. This justification must address the differences between the tested platforms and the untested platforms, and make an argument that the differences do not affect the testing to be performed. It is not sufficient to merely assert that the differences have no affect; rationale must be provided. If all platforms claimed in the ST are tested, then no rationale is necessary.

The test plan describes the composition of each platform to be tested, and any setup that is necessary beyond what is contained in the part of a test or as a standard pre-test condition. This may include special test drivers or

tools. For each driver or tool, an argument (not just an assertion) should be provided that the driver or tool will not adversely affect the performance of the functionality by the TOE and its platform. This also includes the configuration of the cryptographic engine to be used. The cryptographic algorithms implemented by this engine are those specified by this PP and used by the cryptographic protocols being evaluated (IPsec, TLS/HTTPS, SSH).

The test plan identifies high-level test objectives as well as the test procedures to be followed to achieve those objectives. These procedures include expected results. The test report (which could just be an annotated version of the test plan) details the activities that took place when the test procedures were executed, and includes the actual results of the tests. This shall be a cumulative account, so if there was a test run that resulted in a failure; a fix installed; and then a successful re-run of the test, the report would show a 'fail' and 'pass' result (and the supporting details), and not just the 'pass' result.

The evaluator created a proprietary Detailed Test Report (DTR) to address all aspects of this requirement. The DTR discusses the test configuration, test cases, expected results, and test results. The evaluator ran the entire test suite on Windows Server 2016 or 2019 running in a virtual environment (VMWare ESXi 7.0) on and Intel Xeon E5-2670. The testing performed includes the application of policies pushed to various devices, TLS and x509 support, installation and trusted update (as defined by the Testing Assurance Activities of the MDMPP40, MDMA10 and PKGTLS11. The following diagram indicates the test environment.



Packet captures were obtained using servers running tcpdump or wireshark as necessary to capture the test specific traffic. The following are other test tools used during testing.

- Evaluator Laptops
  - Windows 10 with Microsoft Edge / Google Chrome
    - Standard Windows utilities (e.g., notepad, snip tool)
    - Chrome (Version 113)
    - SSH Client – Putty version 6.2
    - SSH Client – SecureCRT version 5.1.2
    - Wireshark version 3.6.5
    - Nmap version 6.25
- Gossamer Test Server
  - Ubuntu Linux 16.04 (DNS: tlv0-16x.gss.com, Address: 10.0.0.30/00:15:5d:00:25:14)
    - Standard Linux commands (e.g., cat, grep, awk)
    - OpenSSL version 1.0.2g-fips (used to generate certificates)

- micro-httpd (comes with Ubuntu – used to establish trivial web for HTTP/TLS client testing
- tcpdump (comes with Ubuntu – used to generate packet capture files for network traffic)
- adb (Android Debug Bridge) version 1.0.41 (used to issue debug commands to the Android devices)
- Evaluator-developed test scripts to facilitate HTTPS/TLS cipher and certificate test cases

3.5 Vulnerability Assessment (AVA)

3.5.1 Vulnerability Survey (AVA_VAN.1)

Assurance Activities: As with ATE_IND, the evaluator shall generate a report to document their findings with respect to this requirement. This report could physically be part of the overall test report mentioned in ATE_IND, or a separate document. The evaluator performs a search of public information to determine the vulnerabilities that have been found in network infrastructure devices and the implemented communication protocols in general, as well as those that pertain to the particular TOE. The evaluator documents the sources consulted and the vulnerabilities found in the report. For each vulnerability found, the evaluator either provides a rationale with respect to its non-applicability, or the evaluator formulates a test (using the guidelines provided in ATE_IND) to confirm the vulnerability, if suitable. Suitability is determined by assessing the attack vector needed to take advantage of the vulnerability. If exploiting the vulnerability requires expert skills and an electron microscope, for instance, then a test would not be suitable and an appropriate justification would be formulated.

The vulnerability analysis is in the proprietary Detailed Test Report (DTR) prepared by the evaluator on 05/02/2024. The vulnerability analysis includes a public search for vulnerabilities. No unaddressed CVE bulletins were discovered.

The search was also updated in the Test Report on 5/22/2024.

The evaluator searched the National Vulnerability Database (https://web.nvd.nist.gov/view/vuln/search) and Vulnerability Notes Database (http://www.kb.cert.org/vuls/) with the following search terms: "BlackBerry", “Certicom Security Builder", "GSE-J", "Unified Endpoint Management", "Java runtime environment", "LDAP", and "TLS".