SECURELOGIX®

CORPORATION

ETM® System Cryptographic Modules Security Policy

Table of Contents

REVISION SUMMARY

Document Version	Description	Date	Editor
1.0	Created document	5/4/2005	CHF
2.0	Updated document based on EWA comments dated 5/30/2005 and 6/23/2005	10/3/2005	CHF
3.0	Updated document based on EWA comments dated 10/27/2005	10/31/2005	CHF
4.0	Updated document based on EWA comments dated 12/19/2005	02/06/2006	CHF
5.0	Updated document based on EWA comments dated 6/13/2006	06/28/2006	CHF
6.0	Updated document based on NIST and CSE review comments	11/20/2006	CHF
7.0	Added procedural description of how to zeroize HMAC keys	3/7/2007	CHF

1. Introduction

1.1. Purpose

This document defines the non-proprietary security policy for the ETM System Cryptographic Modules. The ETM System contains three different cryptographic modules:

- ETM System Firmware Appliance C Comm Crypto Module, Version 5.0
- ETM System Software Application C Comm Crypto Module, Version 5.0
- ETM System Software Application Java Comm Crypto Module, Version 5.0

This security policy describes how the three cryptographic modules meet the requirements of FIPS 140-2 Level 1 for their specified target environments. This document also describes how an authorized user can operate an ETM System component in a FIPS compliant mode.

The security policy provides information regarding the identification and authentication policies, access control, security rules, roles and services, and physical security for the cryptographic modules.

The FIPS 140-2 standard details the U.S. Government requirements for cryptographic modules. Additional information about the standard and validation program is available on the NIST website: http://csrc.nist.gov/cryptval.

1.2. ETM System Overview

The ETM System is a PBX/soft switch-independent, easy-to-use platform that supports security and management applications for real-time visibility, security, and control of telecommunications resources across the enterprise.

ETM System Appliances are installed directly inline on an enterprise’s telecommunications circuits. These custom-designed embedded devices monitor and control VoIP, PRI, CAS, SS7 and analog voice traffic. The Appliances capture and report call data, health & status information, and other real-time telecommunications network and diagnostic data to the ETM System Management Server.

The figure below shows the ETM Model 3200 Appliance, the target platform for the ETM System Firmware Appliance C Comm Crypto Module, Version 5.0.

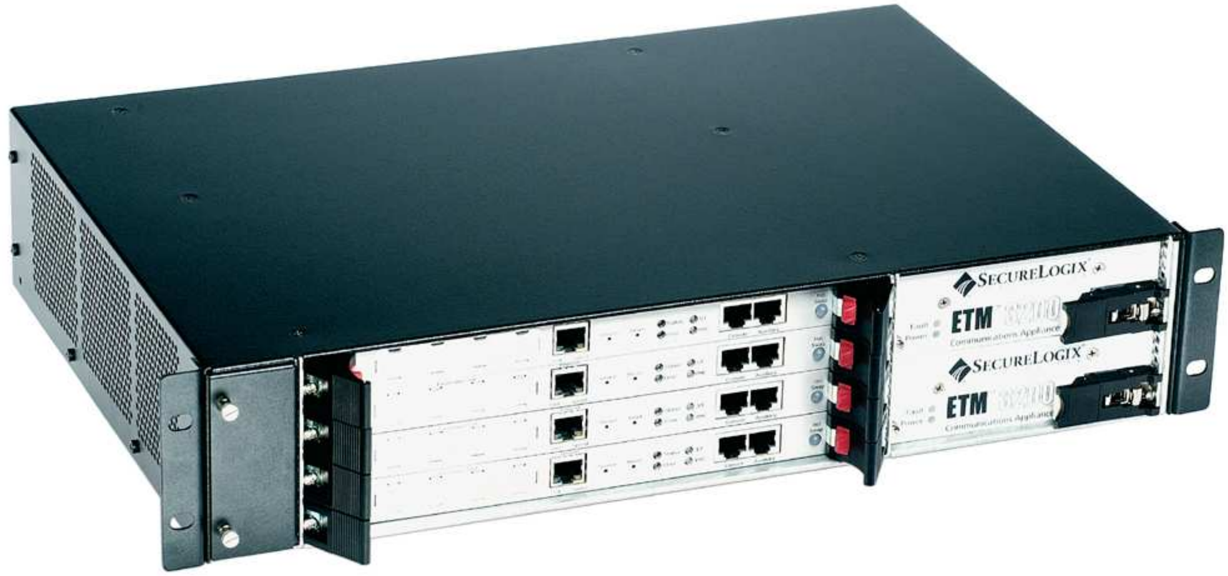

Figure 1: ETM 3200 Appliance

The Management Server, along with the Performance Manager, ETM Report Server, Usage Manager, and Directory Manager, compose the ETM System management infrastructure. These components allow users to manage Appliances, install firewall and other policy types to secure their telecommunications networks, collect call and network health & status information into an Oracle database, run reports, and perform other customized network management functions. ETM System management applications are typically deployed in a distributed architecture across the enterprise LAN or WAN.

Along with the feature set described above, the ETM System also supports a call recording capability. The call recording feature allows a user to capture audio, fax, modem, and other data transmissions from the bearer path of the monitored telecommunications circuit using an ETM System Call Recorder Appliance. The Appliance streams the call recording data to a Model 1060 Cache Appliance for temporary storage. The Cache Appliance then uploads the call data files to a Windows PC-based ETM Collection Server for final storage.

All ETM System applications support encryption for secure intra-component network communication. With the introduction of ETM v5.0, three separate FIPS 140-2 Level 1 validated cryptographic modules, as defined Section 2, will provide encryption services to ETM System applications for secure data exchange.

2. Cryptographic Module Definitions

The three ETM System cryptographic modules are defined in detail below.

- *Appliance Crypto Module*

The ETM System Firmware Appliance C Comm Crypto Module, Version 5.0 (abbreviated FirmwareCM) performs Triple DES encryption and decryption cryptographic operations. The FirmwareCM is implemented entirely in software using the C programming language. The module is embedded as part of the ETM v5.0 Appliance software to secure Appliance-to-Appliance and Appliance-to-Management Server network communication. All ETM v5.0 Appliance models utilize the FirmwareCM, as shown in Table 1.

The FirmwareCM is deployed as a Linux platform shared library that is dynamically linked into the Appliance span processes. Each Appliance executes 1-4 span processes. Each span process loads the FirmwareCM into its own process space. In addition to the FirmwareCM, a separate file containing the computed HMAC-SHA-1 hash for the cryptographic module contents is included in the ETM v5.0 software deployment to allow the cryptographic module to perform a software integrity check on itself during the power up self-tests.

The FirmwareCM is classified as a firmware cryptographic module since the customized Linux O/S that runs on the Appliance is locked down such that only software provided by SecureLogix Corporation can be loaded and executed on the Appliance.

- *Software Application Java Crypto Module*

  The ETM System Software Application Java Comm Crypto Module, Version 5.0 (abbreviated Java-AppCM) performs the same cryptographic operations as the FirmwareCM. The Java-AppCM is implemented entirely in software using the Java programming language and is deployed as a separate jar file that is dynamically linked into each of the ETM System management components, including the ETM Management Server, Performance Manager, ETM Report Server, Usage Manager, and Directory Manager.

  The Java-AppCM encrypts the data that will be exchanged between management applications. The module also handles encryption for secure communication between the Management Server and an Appliance.

  A separate file containing the HMAC-SHA-1 hash for the deployed Java-AppCM is included in the ETM v5.0 software installation. This allows the Java-AppCM to perform a software integrity check on itself during the power up self-tests.

- *Software Application C Crypto Module*

  The ETM Collection Server application utilizes the Software Application C Comm Crypto Module, Version 5.0 (abbreviated C-AppCM) for secure network communication with the Model 1060 Call Recorder Cache Appliance. The C-AppCM provides the same encryption services as the FirmwareCM and Java-AppCM.

  The C-AppCM is deployed as a standalone, dynamically linked export library (DLL) on the Windows 2003 Server.

As with the other two cryptographic modules, the ETM v5.0 Collection Server software installation includes an HMAC-SHA-1 hash file for the FIPS-required software integrity check during power up.

2.1. Cryptographic Module Boundaries

From a hardware perspective, the cryptographic module boundaries for the FirmwareCM, Java-AppCM and C-AppCM are defined by the target platforms on which each is deployed. The FirmwareCM is installed on the custom-built MPC824x-based Controller Card on the ETM System Appliance hardware platform. The Java-AppCM and C-AppCM are installed on commercially available, general purpose computing platforms. See Section 2.3 for the list of target computing platforms.

Figure 2 below shows the physical cryptographic module boundary for the FirmwareCM.

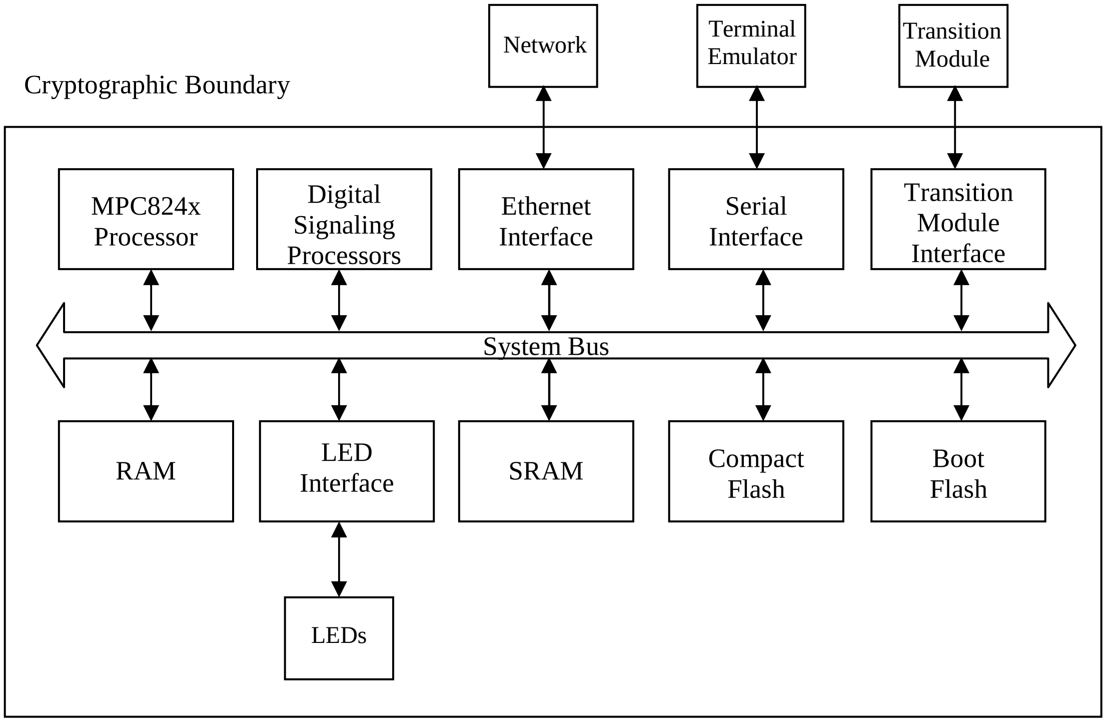

Figure 2: FirmwareCM Physical Cryptographic Module Boundary

Figure 3 depicts the hardware cryptographic module boundary for the Java-AppCM and C-AppCM.

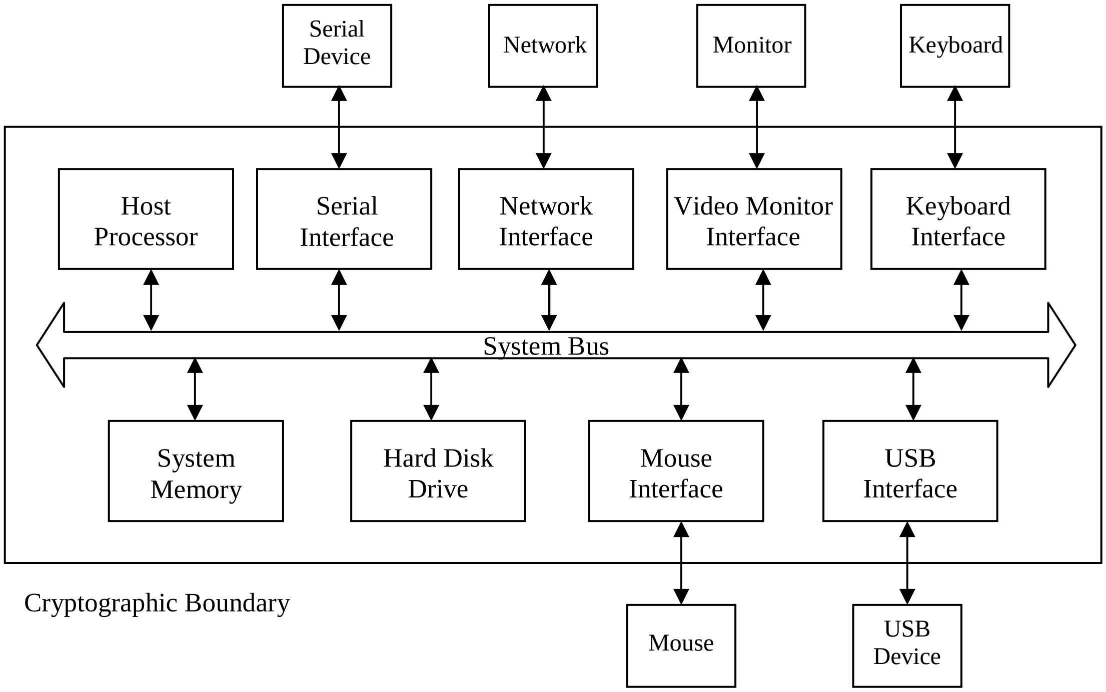

Figure 3: Physical Cryptographic Module Boundaries for the Java-AppCM and C-AppCM

The logical cryptographic boundary for each module is the dynamically linked library itself. Each library provides a set of Application Programming Interface (API) functions to the application that binds it in to its process space. The API functions provide the cryptographic services to the calling application.

Figures 4-6 show the logical cryptographic boundaries for the FirmwareCM, Java-AppCM and C-AppCM.

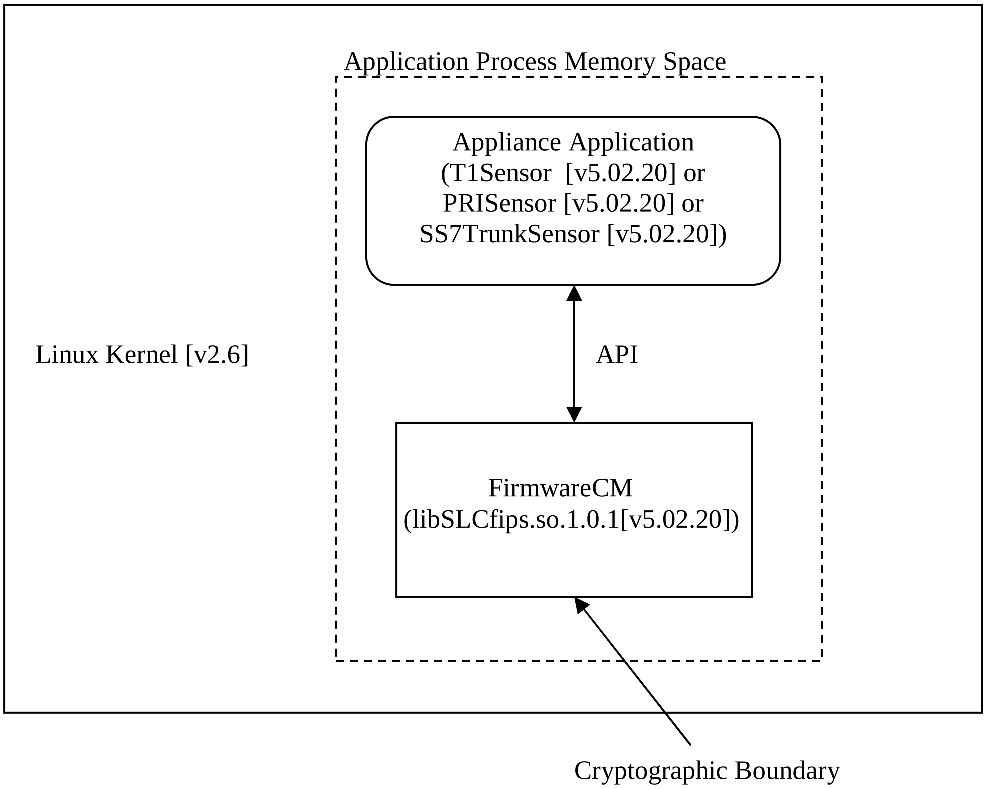

Figure 4: Logical Cryptographic Module Boundary for the FirmwareCM

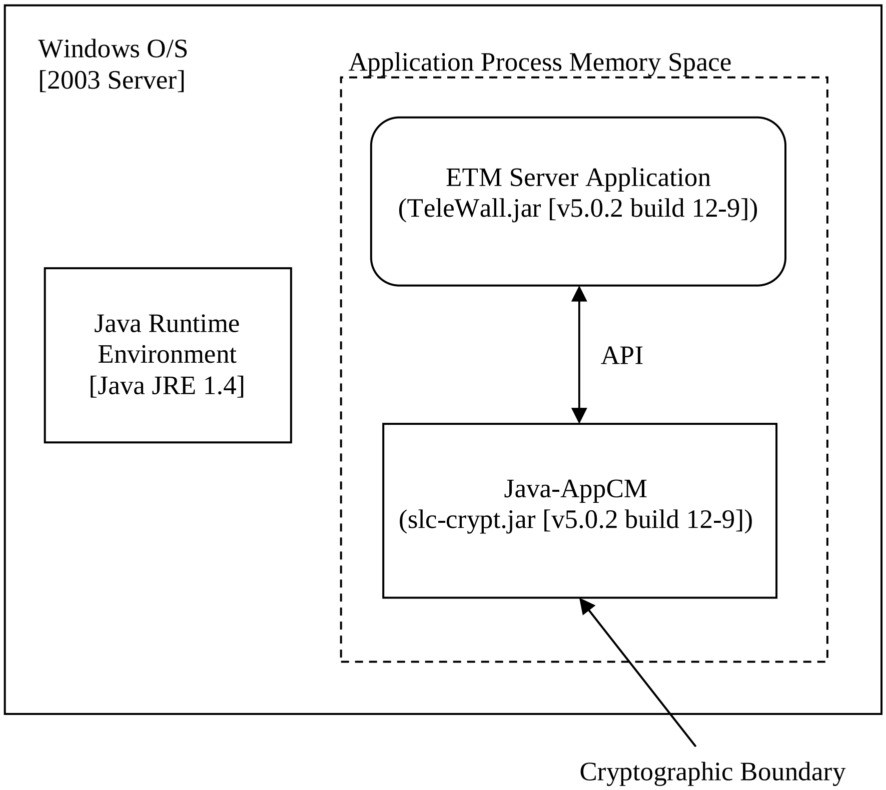

Figure 5: Logical Cryptographic Module Boundary for the Java-AppCM

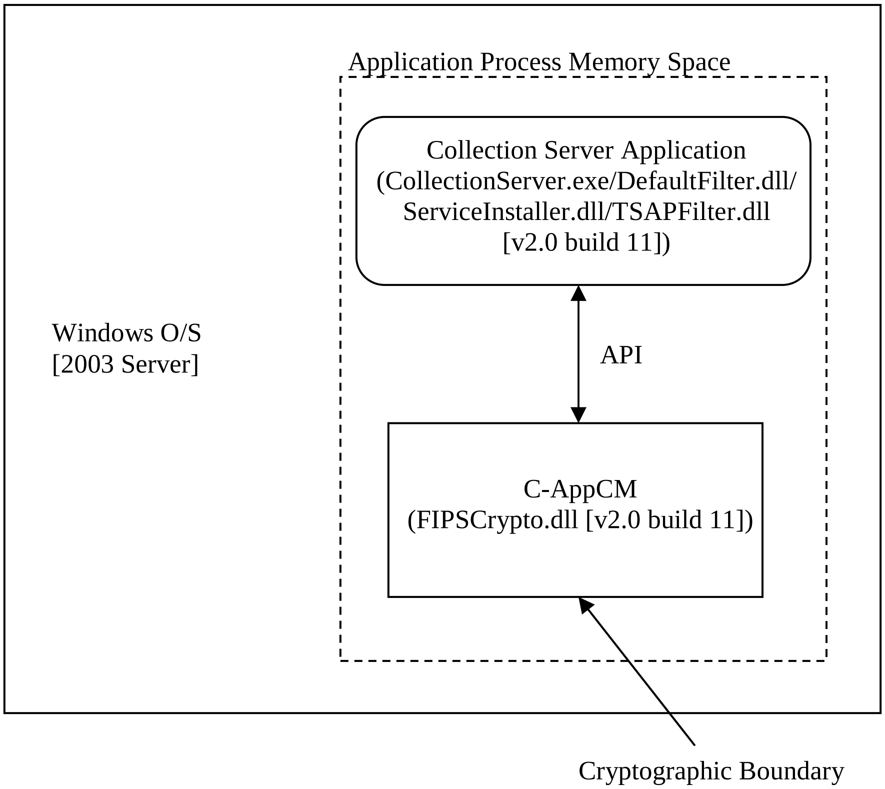
Application Process Memory Space
Collection Server Application (CollectionServer.exe/DefaultFilter.dll/ ServiceInstaller.dll/TSAPFilter.dll [v2.0 build 11])
Windows O/S [2003 Server]
API
C-AppCM (FIPSCrypto.dll [v2.0 build 11])
Cryptographic Boundary

Figure 6: Logical Cryptographic Module Boundary for the C-AppCM

2.2. Cryptographic Module Deployment

The following table maps each of the three cryptographic modules to their respective ETM System components.

Table 1 Cryptographic Module Component Applicability

ETM System Component	Cryptographic Module		
	FirmwareCM	Java-AppCM	C-AppCM
ETM Management Server		X	
Performance Manager		X	
ETM Report Server		X	
Usage Manager		X	
Directory Manager		X	

ETM System Component	Cryptographic Module		
	FirmwareCM	Java-AppCM	C-AppCM
ETM Collection Server			X
Oracle Database	N/A – no encryption used		
ETM MPC860-based Appliances, including models 1000, 1010, 1020, 1030, 1040	X		
ETM MPC824x (8240 and 8245)-based Appliances, including 2100, 3040, 3070, and 3200	X		
ETM MPC8241-based Appliances, including 1012, 1024, 1060, and 1090	X		

2.3. Target Environments

The table below lists the target environments on which the three cryptographic modules have been tested for FIPS 140-2 Level 1 compliance.

Table 2 FIPS 140-2 Level 1 Target Environments

Cryptographic Module	Target Platform	Processor	Operating System	Java Runtime Environment
FirmwareCM	ETM 3200 Appliance	MPC8245	Linux 2.6	N/A
Java-AppCM	IBM Compatible PC	Intel Pentium 4	Windows 2003 Server	Java JRE 1.4
C-AppCM	IBM Compatible PC	Intel Pentium 4	Windows 2003 Server	N/A

In addition, the next table lists the complete set of operating environments on which ETM System components are deployed. The three cryptographic modules are FIPS 140-2 Level 1 compliant for the operating environments shown in Table 3, as they can be executed in the operating environments listed without any modification.

For the FirmwareCM, equivalence is achieved by the use of the same Linux kernel and identical cryptographic module software, both compiled for the different MPC-based processors. The same Java-AppCM and C-AppCM code is used for the two Windows O/S versions.

Table 3 ETM System Components Operating Environments

ETM System Component	Processor Families				O/S Families		Language	
	MPC860	MPC8240/5	MPC8241	x86 (Intel/AMD)	Windows	Linux	C	Java
ETM Management Server				X	Win2K, Win2K3 Server			X
Performance Manager				X	Win2K, Win2K3 Server			X
ETM Report Server				X	Win2K, Win2K3 Server			X
Usage Manager				X	Win2K, Win2K3 Server			X
Directory Manager				X	Win2K, Win2K3 Server			X
ETM Collection Server				X	Win2K, Win2K3 Server		X	
Oracle Database	N/A – no encryption used							
ETM MPC860-based Appliances, including models 1000, 1010, 1020, 1030, 1040	X					X	X	

ETM MPC824x (8240 and 8245)-based Appliances, including 2100, 3040, 3070, and 3200		X				X	X	
ETM MPC8241-based Appliances, including 1012, 1024, 1060, and 1090			X			X	X	

2.4. FIPS 140-2 Security Level

The FirmwareCM meets the overall security requirements applicable to FIPS 140-2 Level 1 as shown in the table below.

Table 4 FirmwareCM Security Level Specification

Security Requirements Section	Level
Cryptographic Module Specification	1
Cryptographic Module Ports and Interfaces	1
Roles, Services, and Authentication	1
Finite State Model	1
Physical Security	1
Operational Environment	N/A
Cryptographic Key Management	1
EMI/EMC	1
Self-Tests	1
Design Assurance	1
Mitigation of Other Attacks	N/A

The Java-AppCM and C-AppCM meet the overall security requirements applicable to Level 1 for all FIPS 140-2 security requirements sections except EMI/EMC. Since the host environments for the Java-AppCM and C-AppCM are PCs rated as Class B devices, the two cryptographic modules meet Level 3 security requirements for EMI/EMC. See Table 5 below.

Table 5 Java-AppCM and C-AppCM Security Level Specifications

Security Requirements Section	Level
Cryptographic Module Specification	1
Cryptographic Module Ports and Interfaces	1
Roles, Services, and Authentication	1
Finite State Model	1
Physical Security	N/A
Operational Environment	1
Cryptographic Key Management	1
EMI/EMC	3
Self-Tests	1
Design Assurance	1
Mitigation of Other Attacks	N/A

3. Cryptographic Modules Ports and Interfaces

Since the three cryptographic modules are implemented entirely in software, the physical ports of the modules map to the ports and connectors on the targeted host-computing environment.

The FirmwareCM operates within the confines of the proprietary ETM System Appliance Controller Card, the MPC824x-based embedded controller board found in the Appliance. The physical interfaces to the Controller Card include the serial port, Ethernet port(s), and Transition Module interface. The Transition Module connects to the analog lines and digital telecommunications spans.

The Java-AppCM and C-AppCM operate on general-purpose desktop computers. The physical interfaces to a PC consist of a mouse, keyboard, monitor, network interface, serial port, and USB port.

Logical interfaces for all three cryptographic modules are defined by their Application Programming Interfaces (API). Specifically,

- The Data Input Interface is the set of parameters passed into the API function calls.
- The Data Output Interface is the subset of parameters updated by a call to an API function.
- Control Input is provided via the calls to the API functions.
- The Status Output is the return value of each API function.

4. Finite State Model

Finite State Models for each of three ETM System cryptographic modules are defined in Ref[1].

5. Physical Security

5.1. FirmwareCM

The FirmwareCM software is loaded onto and executed on an ETM System Appliance Controller Card, comprised of standard integrated circuits with standard passivation and connectors. Each Controller Card is enclosed by a 1U-4U rack-mounted rigid metallic removable cover. The Appliance must be removed from the rack and several screws must be removed from the cover in order to access the Controller Card.

The physical security of the FirmwareCM meets FIPS 140-2 Level 1 compliance.

5.2. C-AppCM and Java-AppCM

Since the C-AppCM and Java-AppCM are implemented entirely in software and executed on IBM compatible PCs, the physical security is provided by the targeted host platforms. Physical security specific to these cryptographic modules themselves is therefore not applicable for FIPS 140-2 Level 1 compliance.

6. Security Rules

This section defines how to use and configure the ETM System components to ensure compliance with the FIPS 140-2 standard.

6.1. Appliance

To operate an Appliance in FIPS mode, the list of rules below must be adhered to:

- The Appliance must be running an ETM System software release containing the ETM System Firmware Appliance C Comm Crypto Module, Version 5.0.
- The Appliance must be running with the span processes active. If the Appliance is placed in Failsafe mode, no span processes are executed and encrypted communication with the ETM Management Server is not available. Thus, an Appliance running in Failsafe mode does not operate in FIPS-approved mode and cannot access any FIPS-approved services provided by the FirmwareCM. To enter Failsafe mode when the span processes are running, the operator must issue the “restart failsafe” command from an ETM command interface. The operator must select option 6 from the Failsafe mode menu to restart the span processes, causing the Appliance to re-enter FIPS-approved mode once the power up self-tests are executed and passed.
- The Crypto Officer must configure the Appliance to use the FIPS-approved Triple DES encryption for Appliance-to-Appliance and Appliance-to-Management Server network communication.

6.2. ETM System Application

To operate the ETM Management Server, Performance Manager, ETM Report Server, Usage Manager, and Directory Manager in FIPS mode, the list of rules below must be adhered to:

- The application must be running an ETM System software release containing the ETM System Software Application Java Comm Crypto Module, Version 5.0.
- The application must use FIPS-approved Triple DES encryption for intra-component network communication.
- All ETM System Applications will operate by default in FIPS mode in ETM v5.0 by utilizing Triple DES encryption for network communication. The DES level is no longer configurable with the ETM v5.0 release. However, for Appliance-to-Management Server communication, the Appliance will dictate the DES level between the platforms. Thus, to operate the Management Server in FIPS mode, the Appliance must first enter FIPS mode according to the rules in Section 6.1. Only after the Appliance is configured to use Triple DES for communication to the Management Server will the server operate in FIPS mode.
- The application must invoke only FIPS-approved APIs when accessing cryptographic services provided by Java-AppCM. For a list of approved APIs, refer to Section 7.2. Refer to Section 7.3 for a list of non-approved APIs.

6.3. Collection Server

For the Collection Server to operate in FIPS mode, the set of rules listed below must be followed:

- The server must be running an ETM System software release containing the ETM System Software Application C Comm Crypto Module, Version 5.0.
- The Crypto Officer must configure each allowed Cache Appliance connection to use FIPS-approved Triple DES encryption for Appliance-to-Collection Server network communication.

7. Roles, Services, and Authentication

7.1. Supported Roles

Each of the three ETM System cryptographic modules supports two roles – User and Crypto Officer. A maintenance role is not supported by any of the modules.

The User and Crypto Officer roles are selected implicitly by the service that is invoked. An operator performs the User role when powering up and using an Appliance, ETM Server component, and/or Collection Server running ETM v5.0 software configured for Triple DES encrypted network communication.

An operator assumes the Crypto Officer role by configuring an Appliance, ETM Server component and/or Collection Server for Triple DES encrypted network communication. The Crypto Officer can also execute the cryptographic services allocated to the User role.

Please refer to the next section for a list of FIPS-approved services and the roles authorized to use each service.

7.2. Approved Services

Services are provided by the cryptographic modules via API functions. The table below lists the services supported by the FirmwareCM and C-AppCM and the role authorized to access each service. See Section 7.3 for a description of each supported service.

Table 5 FirmwareCM and C-AppCM Cryptographic Services

Service	Service API	FIPS-approved Security Functions Used	Cryptographic Keys/CSPs Accessed by Service	Module Type of Access to Keys/CSPs (Read, Write, Delete)	Role Authorized to Use Service
Self-Test	FIPSinit	SHA-1, HMAC-SHA-1, TDES	TDES keys, HMAC key	Read	Crypto Officer
Triple DES Encryption	FIPS_tdes_cfb64_encrypt	Triple DES (64-bit TCFB)	TDES keys	Read	Crypto Officer, User
Triple DES Decryption	FIPS_tdes_cfb64_encrypt	Triple DES (64-bit TCFB)	TDES keys	Read	Crypto Officer, User

The following services listed in the table below are applicable to the Java-AppCM.

Table 6 Java-AppCM Cryptographic Services

Service	Service API	FIPS-approved Security Functions Used	Cryptographic Keys/CSPs Accessed by Service	Module Type of Access to Keys/CSPs (Read, Write, Delete)	Role Authorized to Use Service
Self-Test	FIPSValidator()[1]	SHA-1, HMAC-SHA-1, TDES	TDES keys, HMAC key	Read	Crypto Officer
Key Entry	TripleDES()[1]	None	TDES keys	Write	Crypto

[1] Java class constructor

Service	Service API	FIPS-approved Security Functions Used	Cryptographic Keys/CSPs Accessed by Service	Module Type of Access to Keys/CSPs (Read, Write, Delete)	Role Authorized to Use Service
					Officer
Triple DES Encryption	TripleDES::cfb64_encrypt()	Triple DES (64-bit TCFB)	TDES keys	Read, Write	Crypto Officer, User
Triple DES Decryption	TripleDES::cfb64_encrypt()	Triple DES (64-bit TCFB)	TDES keys	Read, Write	Crypto Officer, User
Key Zeroize	TripleDES::setKey()	None	TDES keys	Delete	Crypto Officer, User

7.3. Non-Approved Services

The Java-AppCM software contains additional APIs that have not been validated for use in FIPS-approved mode.

- `public DesKey( String pKeyString, boolean pCheckKey )`
- `public void TripleDes::ecb_encrypt( … )`
- `public void TripleDes::cbc_encrypt( … )`
- `public void TripleDes::ofb64_encrypt( … )`

7.4. Service Descriptions

All three ETM System cryptographic modules perform the following services:

1. Self-Test – This service performs three self-tests on an ETM System cryptographic module:
   - HMAC-SHA-1 software integrity check on the module library file
   - Known Answer Test (KAT) for the Triple DES encryption algorithm
   - KAT for the Triple DES decryption algorithm

   No other cryptographic services can be accessed until all three self-tests are executed and passed. The cryptographic module returns a status value and result message to the application that indicates the success or failure of the Self-Test operation. See Section 11 for more information.
2. Triple DES Encryption – This service performs FIPS-approved Triple DES CFB64 encryption.
3. Triple DES Decryption – This service performs FIPS-approved Triple DES CFB64 decryption.

In addition to these services, the Java-AppCM performs two other services:

1. Key Entry – This service allows the three Triple DES keys to be imported into the cryptographic module boundary.
2. Key Zeroize – This service allows the three Triple DES keys that were imported into the cryptographic module boundary to be cleared. Please refer to Section 9 for a description of cryptographic key management for the Java-AppCM.

7.5. Authentication

No separate authentication mechanism is provided for each ETM System cryptographic module. Users authorized on the ETM System are implicitly authorized to access any cryptographic module services according to their defined roles.

8. Operational Environment

8.1. Appliance

The FirmwareCM operates on an ETM System Appliance Controller Board and is classified as multiple-chip embedded based on the hardware/firmware design of the board.

8.2. PC

Both the Java-AppCM and C-AppCM operate on Windows-based general purpose computing platforms. Thus, the two modules are classified as multiple-chip standalone.

Each PC must be configured in single-user mode in order for the Java-AppCM and C-AppCM to operate in FIPS-approved modes. This is accomplished by performing the following steps:

- Configure the Windows O/S to only allow a single authenticated user to login.
- Disable all remote logins to the O/S.

9. Cryptographic Key Management

9.1. Key Generation

None of the ETM System cryptographic modules supports any key generation services. Triple DES keys are referenced by address in the FirmwareCM and C-AppCM. For the Java-AppCM, Triple DES keys are directly entered into the cryptographic module in plaintext by its calling application via API calls.

9.2. Key Distribution and Storage

For the FirmwareCM and C-AppCM, cryptographic keys are not stored inside their cryptographic module boundaries. The cryptographic keys are imported into the cryptographic modules from their applications as parameters in the Triple DES encryption and decryption API function calls. The FirmwareCM and C-AppCM do not retain the Triple DES keys passed to them via API calls. The cryptographic keys only exist within the cryptographic boundaries for the duration of the API function calls.

The Java-AppCM supports the import of the three cryptographic keys required for Triple DES encryption and decryption services via API. The ETM Server component imports the cryptographic keys separately from the Triple DES encryption and decryption service API calls. The cryptographic keys are stored inside the Java-AppCM until its application calls the Triple DES key zeroize API, detailed in the next section.

None of the three ETM System cryptographic modules supports the export of cryptographic keys outside of their physical or logical cryptographic boundaries.

9.3. Key Destruction

Only the Java-AppCM stores Triple DES keys within its cryptographic module boundary. As a result, the Java-AppCM supports an API to zeroize the three cryptographic keys.

To zeroize the HMAC key accessed by the Self-Test service in the FirmwareCM, the operator must invoke the Last Resort application to reformat the compact flash on the Appliance. Please refer to section 11.1 for a description of the Last Resort application.

The operator must reformat the hard drive of the host machine that stores the Java-AppCM or C-AppCM and its associated ETM System application executables in order to zeroize the HMAC key for the two PC-based cryptographic modules.

10. Electromagnetic Interference & Compatibility

The Java-AppCM and C-AppCM meet FCC requirements for EMI/EMC when installed on their PC host platforms. All PC manufacturers selling platforms in the U.S. are required to comply with EMI/EMC standards.

All ETM System Appliances are Class A devices that comply with the EMI/EMC requirements necessary for FirmwareCM installation. FCC Part 15 test results for the ETM Model 3200 that verifies compliance with the standards are in Ref[2] – Ref[5].

11. Self-Tests

11.1. Power up Self-Tests

Upon power up, each ETM System component invokes its cryptographic module to perform a software integrity check and cryptographic algorithm checks on itself prior to allowing its application to access its other services.

When each of the three cryptographic module images is built, the computed HMAC-SHA-1 hash of the module contents is written to a separate HMAC-SHA-1 hash file. After the ETM System application loads the module into its process space, the application calls the module’s self-tests API. The API calculates the 20-byte HMAC for the module library file and compares the result to the value in the HMAC-SHA-1 hash file. The values must match for the module's software integrity check to pass.

The cryptographic algorithm tests for each of three modules execute two Known Answer Tests – one for Triple DES encryption and one for Triple DES decryption.

Both categories of power up self-tests are required to be performed and passed before any cryptographic service can be used by an ETM System component. If the self-tests have not been executed and passed, the Triple DES encryption and decryption APIs will return an error to the calling application.

When an ETM System cryptographic module passes the self-tests, the calling application will log the successful result message returned from the self-tests API.

If a cryptographic module fails a self-test, the following occurs:

- An error state is entered, disabling the cryptographic module encryption/decryption services
- An error indicator is returned from the call to the FIPS initialization API to the application program

Each application program type handles the error indicator as follows:

- Appliance span – the span logs the result message to its diagnostic logs. Additionally, the span panics, causing the Appliance card to enter Failsafe mode. In this mode, encryption is not supported and none of the services provided by the FirmwareCM can be accessed. The Appliance can still connect to the Management Server in plaintext mode so that the operator can have remote access to the Appliance from the server. This allows the operator to diagnose and remedy the problem with the cryptographic module, which might require reloading a different software package onto the Appliance.
- Management component – the application logs the error and causes the ETM Server to enter standby mode. In this mode, the server is not operational and no cryptographic services can be performed.
- Collection Server – the server logs the error indicator and exits the application.

Each Appliance span process binds the FirmwareCM into its address space and invokes the power up self-tests upon starting. Span processes are started when the Appliance powers up, after the Appliance reboots, and after the span or Appliance restarts.

Reloading a new software package on an Appliance will cause the Appliance to immediately reboot, effectively restarting the span processes. The power up self-tests will be invoked by the starting span processes as described above.

In addition, ETM v5.0 software supports an Appliance recovery capability known as Last Resort. This capability allows recovery from Appliance installation mistakes or missing software/firmware. Last Resort prompts the operator, interfacing with the Appliance via console port, to enter basic network parameters so that the Appliance can communicate with the Appliance Firmware Maintenance (AFM) Server in order for the server to download the current version of Appliance software. The Last Resort application running on the Appliance then loads the downloaded software, restoring the Appliance to a working operational state.

If the version of software downloaded by the AFM Server is ETM v5.0, then the FirmwareCM is loaded onto the Appliance from outside the existing cryptographic boundary. At the conclusion of the Last Resort process, the Appliance is rebooted and the span processes are restarted. At this point, each span process invokes the power up self-tests for the newly loaded FirmwareCM. No cryptographic services are accessible until the power up self-tests execute and pass.

For the Java-AppCM and C-AppCM, the power up self-tests are invoked when their parent applications are started.

11.2. On Demand Self-Tests

While an ETM System cryptographic module is running, there is no method to invoke on demand self-tests. The operator must restart the parent application in order to trigger the self-tests service for the ETM System cryptographic module.

11.3. Conditional Self Tests

The Software/Firmware Load Test is applicable for the FirmwareCM. This test performs the HMAC-SHA-1 integrity check.

No operations are supported by the C-AppCM or Java-AppCM that require conditional self-tests to be performed.

12. Mitigation of Other Attacks

The FirmwareCM, Java-AppCM, and C-AppCM are not designed for the mitigation of known specific attacks.

13. Cryptographic Algorithms

13.1. FIPS Approved

The FirmwareCM, Java-AppCM, and the C-AppCM support the following FIPS-approved cryptographic algorithms:

Encryption Algorithms

- Triple DES (64-bit Cipher Feedback (TCFB) mode)

Hashing Algorithms

- SHA-1

Authentication Algorithms

- HMAC-SHA-1 (20 byte MAC size)

13.2. Non-FIPS Approved

The FirmwareCM, Java-AppCM, and the C-AppCM also support the DES (64-bit Cipher Feedback (CFB64) mode) encryption algorithm. The ETM System cryptographic modules will not be in a FIPS-approved mode of operation when using the DES algorithm. The algorithm implementation has not been validated in the specific operational environments in which the cryptographic modules have been tested.

In addition, the ECB, CBC, and 64-bit OFB Triple DES modes are supported by the Java-AppCM. The Triple DES algorithm has not been validated for these modes of operation. Therefore, the Java-AppCM will not function in a FIPS-approved mode of operation when one of these Triple DES modes is executed.

14. References

[1] ETM System Cryptographic Modules Finite State Machine

[2] Intertek EMC Report for SecureLogix, Inc. on the ETM3200, File 3047025.012

[3] Intertek EMC Report for SecureLogix, Inc. on the ETM3200, File 3047025.013

[4] Intertek EMC Report for SecureLogix, Inc. on the ETM3200, File 3047025.014

[5] Intertek EMC Report for SecureLogix, Inc. on the ETM3200, File 3047025.015