FireEye X-Agent Application Security Target

Acumen Security, LLC.

Document Version: 1.8

Table Of Contents

Revision History

Version	Date	Description
1.0	June 2018	Publication
1.1	September 2018	Updated based on internal review and vendor feedback
1.2	September 2018	Added detail
1.3	October 2018	Updated based on internal review
1.4	December 2018	Updated based on ECR comments
1.5	February 2019	Updated based on functional testing
1.6	May 2019	Updates for project check-out
1.7	July 2019	Updated based on ECR comments
1.8	July 2019	Updated to include TD0427 and TD0434

1 Security Target Introduction

1.1 Security Target and TOE Reference

This section provides information needed to identify and control this ST and its TOE.

Category	Identifier
ST Title	FireEye X-Agent Security Target
ST Version	1.8
ST Date	July 2019
ST Author	Acumen Security, LLC.
TOE Identifier	FireEye X-Agent
TOE Software Version	28.8.3
TOE Developer	FireEye, Inc.
Key Words	Software

Table 1 TOE/ST Identification

1.2 TOE Overview

The TOE is a software agent that resides on a host platform. The software exclusively interacts with the NIAP validated FireEye HX Series Appliances (NIAP VID 10892). This interaction consists of the TOE receiving policies from an external HX series appliance (validated separately) and sending any alerts that are found as a result of these scans. This is done via polling. The TOE is an enterprise managed agent that runs in the background of an endpoint platform. It is intended that the user will have no interaction with the software and will not be alerted of communications with the external HX appliance.

The frequency at which the agent communicates with the HX appliance is set by the enterprise. By default, each agent polls the HX appliance every 600 seconds (10 minutes) to obtain information and task requests and polls the appliance every 30 minutes to obtain the latest indicators. When new policies are received, they are used to identify potential intrusions on the host platform.

1.3 TOE Architecture

1.3.1 Physical Boundaries

The TOE boundary is the application software which runs on the host platform. The software is pushed to the host platform from a FireEye HX series and installs natively as a kernel and user space application. The software runs on Microsoft Operating Systems. The following Operating Systems are included in this evaluation,

- Windows Server 2012R2 x64 running on an Intel Xeon E5 processor
- Windows Server 2008R2 (SP1) x64 running on an Intel Xeon E5 processor
- Windows 10 Version 1507 x86 and x64 running on an Intel Xeon E5 processor
- Windows 10 Version 1511 x86 and x64 running on an Intel Xeon E5 processor
- Windows 10 Version 1607 x86 and x64 running on an Intel Xeon E5 processor
- Windows 10 Version 1703 x86 and x64 running on an Intel Xeon E5 processor
- Windows 10 Version 1709 x86 and x64 running on an Intel Xeon E5 processor
- Windows 10 Version 1803 x86 and x64 running on an Intel Xeon E5 processor
- Windows 10 Version 1809 x86 and x64 running on an Intel Xeon E5 processor
- Windows Server 2016 Version 1607 on an Intel Xeon E5 processor

1.3.2 Security Functions provided by the TOE

The TOE provides the security functionality required by [SWAPP].

1.3.2.1 Cryptographic Support

The TOE provides cryptographic support for the following features,

- TLS connectivity with the following entities:
  - HX Series Appliance (NIAP VID 10892)
- Digital certificate validation

Each of these cryptographic algorithms have been validated for conformance to the requirements specified in their respective standards, as identified below. Each of these algorithms are implemented as part of the FireEye OpenSSL Cryptographic Library version 2.0.10-fe1 which is part of the TOE.

Algorithm	Standard	Mode/Keysize	CAVP Cert. #
AES	FIPS 197 SP 800-38A	CBC 128, CBC 256	C779 (AES)
SHA	FIPS 180-4	SHA-1, SHA-256	C779 (SHS)
RSA	FIPS 186-4	n = 2048 SHA-256	C779 (RSA)
HMAC	FIPS 198-1	HMAC-SHA-1, HMAC-SHA-256	C779 (HMAC)
DRBG	SP 800-90A	CTR_DRBG(AES-256)	C779 (DRBG)

Table 2 CAVP Certificate References

1.3.2.2 Identification and Authentication

The TOE uses X.509v3 certificates as defined by RFC 5280 to authentication the TLS connection to the HX Series appliance. The TOE validates the X.509 certificates using the certificate path validation algorithm defined in RFC 5280.

1.3.2.3 Secure Software Update

The TOE is distributed as a Microsoft .MSI file providing a consistent and reliable versioning. After initial installation, all updates to the X-Agent are distributed as .MSI. Each TOE installation and update is signed by FireEye and can only come from the HX Series appliance associated with the TOE.

1.3.2.4 Privacy

The TOE does not transmit Personally Identifiable Information (PII) over the network. This aids in protecting the privacy of users of the host platform.

1.3.2.5 Protection of the TSF

The TOE employs several mechanisms to ensure that it is secure on the host platform. The TOE never allocates memory with both write and execute permission. The TOE is designed to operate in an environment in which the following security techniques are in effect, Data execution prevention, Mandatory address space layout randomization (no memory map to an explicit address), Structured exception handler overwrite protection, Export address table access filtering, and Anti-Return Oriented Programming. This allows the TOE to operate in an environment in which the Enhanced Mitigation Experience Toolkit is also running. During compilation the TOE is built with several flags enabled that check for engineering flaws. The TOE is built with the /GS flag enabled. This reduces the possibilities of stack-based buffer overflows in the product. The compiler enables ASLR by default. The TOE is not built with the /DYNAMICBASE:NO which would disable ASLR.

1.3.2.6 Trusted Path/Channels

The TOE receives scanning policies from the associated HX Series appliance over the network which it uses on the host platform. This connection is always secured using TLS.

1.3.3 TOE Documentation

- [ST] FireEye X-Agent Application Security Target, version 1.8
- [AGD] Common Criteria FireEye Endpoint Agent Addendum, Release 28 Revision 2

1.3.4 Other References

Protection Profile for Application Software, version 1.2, dated, 22 April 2016 [SWAPP].

2 Conformance Claims

2.1 CC Conformance

This TOE is conformant to:

- Common Criteria for Information Technology Security Evaluations Part 1, Version 3.1, Revision 4, September 2012
- Common Criteria for Information Technology Security Evaluations Part 2, Version 3.1, Revision 4, September 2012: Part 2 extended
- Common Criteria for Information Technology Security Evaluations Part 2, Version 3.1, Revision 4, September 2012: Part 3 extended

2.2 Protection Profile Conformance

This TOE is conformant to:

- Protection Profile for Application Software, version 1.2, dated, 22 April 2016 [SWAPP].

2.3 Conformance Rationale

This Security Target provides exact conformance to Version 1.2 of the Protection Profile for Application Software, version 1.2. The security problem definition, security objectives and security requirements in this Security Target are all taken from the Protection Profile performing only operations defined there.

2.3.1 Technical Decisions

All NIAP Technical Decisions (TDs) issued to date that are applicable to [SWAPP] have been addressed. The following table identifies all applicable TD:

Identifier	Applicable	Exclusion Rationale (if applicable)
0434 – Windows Desktop Applications Test	Yes	
0427 – Reliable Time Source	Yes	
0392 – FCS_TLSC_EXT.1.2 Wildcard Checking	Yes	
0390 – Cryptographically Secure RNG	Yes	
0389 – Handling of SSH EP claim for platform	Yes	
0385 – FTP_DIT_EXT.1 Assurance Activity Clarification	No	This TD applies to the VPN Client Module.
0382 – Configuration Storage Options for Apps	Yes	
0380 – Linux Keyring Requirement in FCS_STO_EXT.1	No	This TD applies to the Linux platform. The TOE operations on Windows platforms.
0364 – Android mmap testing for FPT_AEX_EXT.1.1	No	This TD applies to the Android platform. The TOE operations on Windows platforms.
0359 – Buffer Protection	Yes	
0358 – Cipher Suites for TLS in SWApp v1.2	Yes	
0327 – Default file permissions for FMT_CFG_EXT.1.2	Yes	
0326 – RSA-based key establishment schemes	Yes	
0305 – Handling of TLS connections with and without mutual authentication	No	The TOE does not claim conformance to FCS_TLSC_EXT.2
0304 – Update to FCS_TLSC_EXT.1.2	Yes	
0300 – Sensitive Data in FDP_DAR_EXT.1	Yes	
0296 – Update to FCS_HTTPS_EXT.1.3	No	The TOE does not claim conformance to FCS_HTTPS_EXT.1
0295 – Update to FPT_AEX_EXT.1.3 Assurance Activities	Yes	
0293 – Update to FCS_CKM.1(1)	No	This TD has been archived
0283 – Cipher Suites for TLS in SWApp v1.2	No	This TD has been archived

Identifier	Applicable	Exclusion Rationale (if applicable)
0269 – Update to FPT_AEX_EXT.1.3 Assurance Activity	No	This TD has been archived
0268 – FMT_MEC_EXT.1 Clarification	Yes	
0267 – TLSS testing - Empty Certificate Authorities list	No	The TOE does not claim conformance to FCS_TLSS_EXT.1
0244 – FCS_TLSC_EXT - TLS Client Curves Allowed	No	The TOE does not claim conformance to FCS_TLSC_EXT.4
0241 – Removal of Test 4.1 in FCS_TLSS_EXT.1.1	No	The TOE does not claim conformance to FCS_TLSS_EXT.1
0238 – User-modifiable files FPT_AEX_EXT.1.4	Yes	
0221 – FMT_SMF.1.1 - Assignments moved to Selections	No	This TD only applies to the SWFE EP.
0218 – Update to FPT_AEX_EXT.1.3 Assurance Activity	No	This TD has been archived
0217 – Compliance to RFC5759 and RFC5280 for using CRLs	Yes	
0215 – Update to FCS_HTTPS_EXT.1.2	No	The TOE does not claim conformance to FCS_HTTPS_EXT.1
0192 – Update to FCS_STO_EXT.1 Application Note	No	This TD has been archived
0178 – Integrity for installation tests in AppSW PP	Yes	
0177 – FCS_TLSS_EXT.1 Application Note Update	No	Superseded by TD0389
0174 – Optional Ciphersuites for TLS	Yes	
0172 – Additional APIs added to FCS_RBG_EXT.1.1	No	Replaced by TD0390
0163 – Update to FCS_TLSC_EXT.1.1 Test 5.4 and FCS_TLSS_EXT.1.1 Test	Yes	
0131 – Update to FCS_TLSS_EXT.1.1 Test 4.5	No	The TOE does not claim conformance to FCS_TLSS_EXT.1
0122 – FMT_SMF.1.1 Assignments moved to Selections	No	This TD has been archived
0121 – FMT_MEC_EXT.1.1 Configuration Options	No	This TD only applies to the SWFE EP.
0119 – FCS_STO_EXT.1.1 in PP_APP_v1.2	Yes	
0107 – FCS_CKM - ANSI X9.31-1998, Section 4.1.for Cryptographic Key Generation	No	Superseded by TD0326

Table 3 Technical Decisions

3 Security Problem Definition

The security problem definition has been taken from [SWAPP] and is reproduced here for the convenience of the reader. The security problem is described in terms of the threats that the TOE is expected to address, assumptions about the operational environment, and any organizational security policies that the TOE is expected to enforce.

3.1 Threats

The following threats are drawn directly from the [SWAPP].

ID	Threat
T.NETWORK_ATTACK	An attacker is positioned on a communications channel or elsewhere on the network infrastructure. Attackers may engage in communications with the application software or alter communications between the application software and other endpoints in order to compromise it.
T.NETWORK_EAVESDROP	An attacker is positioned on a communications channel or elsewhere on the network infrastructure. Attackers may monitor and gain access to data exchanged between the application and other endpoints.
T.LOCAL_ATTACK	An attacker can act through unprivileged software on the same computing platform on which the application executes. Attackers may provide maliciously formatted input to the application in the form of files or other local communications.
T.PHYSICAL_ACCESS	An attacker may try to access sensitive data at rest.

Table 4 Threats

3.2 Assumptions

The following assumptions are drawn directly from the [SWAPP].

ID	Assumption
A.PLATFORM	The TOE relies upon a trustworthy computing platform with a reliable time clock for its execution. This includes the underlying platform and whatever runtime environment it provides to the TOE.
A.PROPER_USER	The user of the application software is not willfully negligent or hostile, and uses the software in compliance with the applied enterprise security policy
A.PROPER_ADMIN	The administrator of the application software is not careless, willfully negligent or hostile, and administers the software within compliance of the applied enterprise security policy

Table 5 OSPs

3.3 Organizational Security Policies

There are no OSPs for the application.

4 Security Objectives

The security objectives have been taken from [SWAPP] and are reproduced here for the convenience of the reader.

4.1 Security Objectives for the TOE

The following security objectives for the TOE were drawn directly from the [SWAPP].

ID	TOE Objective
O.INTEGRITY	Conformant TOEs ensure the integrity of their installation and update packages, and also leverage execution environment-based mitigations. Software is seldom if ever shipped without errors, and the ability to deploy patches and updates to fielded software with integrity is critical to enterprise network security. Processor manufacturers, compiler developers, execution environment vendors, and operating system vendors have developed execution environment-based mitigations that increase the cost to attackers by adding complexity to the task of compromising systems. Application software can often take advantage of these mechanisms by using APIs provided by the runtime environment or by enabling the mechanism through compiler or linker options. Addressed by: FDP_DEC_EXT.1, FMT_CFG_EXT.1, FPT_AEX_EXT.1, FPT_TUD_EXT.1
O.QUALITY	To ensure quality of implementation, conformant TOEs leverage services and APIs provided by the runtime environment rather than implementing their own versions of these services and APIs. This is especially important for cryptographic services and other complex operations such as file and media parsing. Leveraging this platform behavior relies upon using only documented and supported APIs Addressed by: FMT_MEC_EXT.1, FPT_API_EXT.1, FPT_LIB_EXT.1
O.MANAGEMENT	To facilitate management by users and the enterprise, conformant TOEs provide consistent and supported interfaces for their security-relevant configuration and maintenance. This includes the deployment of applications and application updates through the use of platform-supported deployment mechanisms and formats, as well as providing mechanisms for configuration. This also includes providing control to the user regarding disclosure of any PII. Addressed by: FMT_SMF.1, FPT_IDV_EXT.1, FPT_TUD_EXT.1.5, FPR_ANO_EXT.1
O.PROTECTED_STORAGE	To address the issue of loss of confidentiality of user data in the event of loss of physical control of the storage medium, conformant TOEs will use data-at-rest protection. This involves encrypting data and keys stored by the TOE in order to prevent unauthorized access to this data. This also includes unnecessary network communications whose consequence may be the loss of data Addressed by: FDP_DAR_EXT.1, FCS_STO_EXT.1, FCS_RBG_EXT.1
O.PROTECTED_COMMS	To address both passive (eavesdropping) and active (packet modification) network attack threats, conformant TOEs will use a trusted channel for sensitive data. Sensitive data includes cryptographic keys, passwords, and any other data specific to the application that should not be exposed outside of the application Addressed by: FTP_DIT_EXT.1, FCS_TLSC_EXT.1, FCS_DTLS_EXT.1, FCS_RBG_EXT.1

Table 6 Objectives for the TOE

4.2 Security Objectives for the Operational Environment

The following security objectives for the operational environment assist the TOE in correctly providing its security functionality. These track with the assumptions about the environment.

ID	Objective for the Operational Environment
OE.PLATFORM	The TOE relies upon a trustworthy computing platform for its execution. This includes the underlying operating system and any discrete execution environment provided to the TOE.

OE.PROPER_USER	The user of the application software is not willfully negligent or hostile, and uses the software within compliance of the applied enterprise security policy
OE.PROPER_ADMIN	The administrator of the application software is not careless, willfully negligent or hostile, and administers the software within compliance of the applied enterprise security policy

Table 7 Objectives for the environment

5 Security Requirements

This section identifies the Security Functional Requirements for the TOE and/or Platform. The Security Functional Requirements included in this section are derived from Part 2 of the Common Criteria for Information Technology Security Evaluation, Version 3.1, Revision 4, dated: September 2012 and all international interpretations.

Requirement	Auditable Event
FCS_CKM_EXT.1	Cryptographic Key Generation Services
FCS_CKM.2	Cryptographic Key Establishment
FCS_COP.1(1)	Cryptographic Operation - Encryption/Decryption
FCS_COP.1(2)	Cryptographic Operation - Hashing
FCS_COP.1(3)	Cryptographic Operation - Signing
FCS_COP.1(4)	Cryptographic Operation - Keyed-Hash Message Authentication
FCS_RBG_EXT.1	Random Bit Generation Services
FCS_RBG_EXT.2	Random Bit Generation from Application
FCS_STO_EXT.1	Storage of Credentials
FCS_TLSC_EXT.1	TLS Client Protocol
FDP_DEC_EXT.1	Access to Platform Resources
FDP_NET_EXT.1	Network Communications
FDP_DAR_EXT.1	Encryption Of Sensitive Application Data
FIA_X509_EXT.1	X.509 Certificate Validation
FIA_X509_EXT.2	X.509 Certificate Authentication
FMT_MEC_EXT.1	Supported Configuration Mechanism
FMT_CFG_EXT.1	Secure by Default Configuration
FMT_SMF.1	Specification of Management Functions
FPR_ANO_EXT.1	User Consent for Transmission of Personally Identifiable Information
FPT_API_EXT.1	Use of Supported Services and APIs
FPT_AEX_EXT.1	Anti-Exploitation Capabilities
FPT_TUD_EXT.1	Integrity for Installation and Update
FPT_LIB_EXT.1	Use of Third Party Libraries
FTP_DIT_EXT.1	Protection of Data in Transit

Table 8 SFRs

5.1 Conventions

The CC defines operations on Security Functional Requirements: assignments, selections, assignments within selections and refinements. This document uses the following font conventions to identify the operations defined by the CC:

- Assignment: Indicated with *italicized* text;
- Refinement: Indicated with **bold** text;
- Selection: Indicated with underlined text;
- Iteration: Indicated by appending the iteration number in parenthesis, e.g., (1), (2), (3).
- Where operations were completed in the PP itself, the formatting used in the PP has been retained.

Explicitly stated SFRs are identified by having a label 'EXT' after the requirement name for TOE SFRs. Formatting conventions outside of operations matches the formatting specified within the PP.

5.2 Security Functional requirements

5.2.1 Cryptographic Support (FCS)

FCS_CKM_EXT.1 Cryptographic Key Generation Services

FCS_CKM_EXT.1.1

The application shall [generate no asymmetric cryptographic keys].

FCS_CKM.2 Cryptographic Key Establishment

FCS_CKM.2.1

The application shall [implement functionality] to perform cryptographic key establishment in accordance with a specified cryptographic key establishment method: [RSA-based key establishment schemes] that meets the following: [NIST Special Publication 800-56B, "Recommendation for Pair-Wise Key Establishment Schemes Using Integer Factorization Cryptography"].

FCS_COP.1(1) Cryptographic Operation - Encryption/Decryption

FCS_COP.1.1(1)

The application shall perform encryption/decryption in accordance with a specified cryptographic algorithm

- AES-CBC (as defined in NIST SP 800-38A) mode

and [no other modes] and cryptographic key sizes 256-bit key sizes and [128-bit key sizes].

FCS_COP.1(2) Cryptographic Operation - Hashing

FCS_COP.1.1(2) The application shall perform cryptographic hashing services in accordance with a specified cryptographic algorithm [SHA-1, SHA-256] and message digest sizes [160, 256] bits that meet the following: FIPS Pub 180-4.

FCS_COP.1(3) Cryptographic Operation - Signing

FCS_COP.1.1(3)

The application shall perform cryptographic signature services (generation and verification) in accordance with a specified cryptographic algorithm [RSA schemes using cryptographic key sizes of 2048-bit or greater that meet the following: FIPS PUB 186-4, "Digital Signature Standard (DSS)", Section 4].

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

FCS_COP.1.1(4)

The application shall perform keyed-hash message authentication in accordance with a specified cryptographic algorithm

- HMAC-SHA-256 and [SHA-1]

with key sizes [*256 bits, 160 bits*] and message digest sizes 256 and [160] bits that meet the following: FIPS Pub 198-1 The Keyed-Hash Message Authentication Code and FIPS Pub 180-4 Secure Hash Standard.

FCS_RBG_EXT.1 Random Bit Generation Services

FCS_RBG_EXT.1.1

The application shall [implement DRBG functionality] for its cryptographic operations.

FCS_RBG_EXT.2 Random Bit Generation Services

FCS_RBG_EXT.2.1

The application shall perform all deterministic random bit (DRBG) services in accordance with NIST Special Publication 800-90A using [CTR_DRBG(AES)].

FCS_RBG_EXT.2.2

The deterministic RBG shall be seeded by an entropy source that accumulates entropy from a platform-based DRBG and [

no other noise source

] with a minimum of [

256 bits

] of entropy at least equal to the greatest security strength (according to NIST SP 800-57) of the keys and hashes that it will generate.

FCS_STO_EXT.1 Storage of Credentials

FCS_STO_EXT.1.1

The application shall [implement functionality to securely store [*digital certificates*]] to non-volatile memory.

FCS_TLSC_EXT.1 TLS Client Protocol

FCS_TLSC_EXT.1.1

The application shall [implement TLS 1.2 (RFC 5246)] supporting the following ciphersuites:

- [TLS_RSA_WITH_AES_128_CBC_SHA as defined in RFC 5246].

FCS_TLSC_EXT.1.2

The application shall verify that the presented identifier matches the reference identifier according to RFC 6125.

FCS_TLSC_EXT.1.3

The application shall establish a trusted channel only if the peer certificate is valid.

5.2.2 User Data Protection (FDP)

FDP_DEC_EXT.1 Access to Platform Resources

FDP_DEC_EXT.1.1

The application shall restrict its access to [network connectivity].

FDP_DEC_EXT.1.2

The application shall restrict its access to [system logs, *[RAM, filesystem]*].

FDP_NET_EXT.1 Network Communications

FDP_NET_EXT.1.1

The application shall restrict network communication to [*polling and downloading new scanning policies to be used to identify potential intrusions on the host OS from the associated FireEye HX appliance, sending information to the associated FireEye HX appliance as defined in the downloaded scanning policies*].

FDP_DAR_EXT.1 Encryption Of Sensitive Application Data

FDP_DAR_EXT.1.1

The application shall [leverage platform provided functionality to encrypt sensitive data] in non-volatile memory.

5.2.3 Identification and Authentication (FIA)

FIA_X509_EXT.1 X.509 Certificate Validation

FIA_X509_EXT.1.1

The application shall [implement functionality] to validate certificates in accordance with the following rules:

- RFC 5280 certificate validation and certificate path validation.
- The certificate path must terminate with a trusted CA certificate.
- The application shall validate a certificate path by ensuring the presence of the basicConstraints extension and that the CA flag is set to TRUE for all CA certificates.
- The application shall validate the revocation status of the certificate using [a Certificate Revocation List (CRL) as specified in RFC 5280 Section 6.3].
- The application shall validate the extendedKeyUsage field according to the following rules:
    - Certificates used for trusted updates and executable code integrity verification shall have the Code Signing purpose (id-kp 3 with OID 1.3.6.1.5.5.7.3.3) in the extendedKeyUsage field.
    - Server certificates presented for TLS shall have the Server Authentication purpose (id-kp 1 with OID 1.3.6.1.5.5.7.3.1) in the extendedKeyUsage field.
    - Client certificates presented for TLS shall have the Client Authentication purpose (id-kp 2 with OID 1.3.6.1.5.5.7.3.2) in the extendedKeyUsage field.
    - S/MIME certificates presented for email encryption and signature shall have the Email Protection purpose (id-kp 4 with OID 1.3.6.1.5.5.7.3.4) in the extendedKeyUsage field.
    - OCSP certificates presented for OCSP responses shall have the OCSP Signing purpose (id-kp 9 with OID 1.3.6.1.5.5.7.3.9) in the extendedKeyUsage field.
    - Server certificates presented for EST shall have the CMC Registration Authority (RA) purpose (id-kp-cmcRA with OID 1.3.6.1.5.5.7.3.28) in the extendedKeyUsage field.

FIA_X509_EXT.1.2

The application shall only treat a certificate as a CA certificate if the basicConstraints extension is present and the CA flag is set to TRUE.

FIA_X509_EXT.2 X.509 Certificate Authentication

FIA_X509_EXT.2.1

The application shall use X.509v3 certificates as defined by RFC 5280 to support authentication for [TLS].

FIA_X509_EXT.2.2

When the application cannot establish a connection to determine the validity of a certificate, the application shall [not accept the certificate].

5.2.4 Security Management (FMT)

FMT_MEC_EXT.1 Supported Configuration Mechanism

FMT_MEC_EXT.1.1

The application shall [invoke the mechanisms recommended by the platform] vendor for storing and setting configuration options.

FMT_CFG_EXT.1 Secure by Default Configuration

FMT_CFG_EXT.1.1

The application shall only provide enough functionality to set new credentials when configured with default credentials or no credentials.

FMT_CFG_EXT.1.2

The application shall be configured by default with file permissions which protect the application's binaries and data files from modification by normal unprivileged user.

FMT_SMF.1 Specification of Management Functions

FMT_SMF.1.1

The TSF shall be capable of performing the following management functions [*no management functions*].

5.2.5 Privacy (FPR)

FPR_ANO_EXT.1 User Consent for Transmission of Personally Identifiable Information

FPR_ANO_EXT.1.1

The application shall [not transmit PII over a network].

5.2.6 Protection of TSF (FPT)

FPT_API_EXT.1 Use of Supported Services and APIs

FPT_API_EXT.1.1

The application shall use only documented platform APIs.

FPT_AEX_EXT.1 Anti-Exploitation Capabilities

FPT_AEX_EXT.1.1

The application shall not request to map memory at an explicit address except for [*no exceptions*].

FPT_AEX_EXT.1.2

The application shall [not allocate any memory region with both write and execute permissions].

FPT_AEX_EXT.1.3

The application shall be compatible with security features provided by the platform vendor.

FPT_AEX_EXT.1.4

The application shall not write user-modifiable files to directories that contain executable files unless explicitly directed by the user to do so.

FPT_AEX_EXT.1.5

The application shall be compiled with stack-based buffer overflow protection enabled.

FPT_TUD_EXT.1 Integrity for Installation and Update

FPT_TUD_EXT.1.1

The application shall [leverage the platform] to check for updates and patches to the application software.

FPT_TUD_EXT.1.2

The application shall be distributed using the format of the platform-supported package manager.

FPT_TUD_EXT.1.3

The application shall be packaged such that its removal results in the deletion of all traces of the application, with the exception of configuration settings, output files, and audit/log events.

FPT_TUD_EXT.1.4

The application shall not download, modify, replace or update its own binary code.

FPT_TUD_EXT.1.5

The application shall [provide the ability] to query the current version of the application software.

FPT_TUD_EXT.1.6

The application installation package and its updates shall be digitally signed such that its platform can cryptographically verify them prior to installation.

FPT_LIB_EXT.1 Use of Third Party Libraries

FPT_LIB_EXT.1.1

The application shall be packaged with only [*api-ms-win-core-console-l1-1-0.dll, api-ms-win-core-datetime-l1-1-0.dll, api-ms-win-core-debug-l1-1-0.dll, api-ms-win-core-errorhandling-l1-1-0.dll, api-ms-win-core-file-l1-1-0.dll, api-ms-win-core-file-l1-2-0.dll, api-ms-win-core-file-l2-1-0.dll, api-ms-win-core-handle-l1-1-0.dll, api-ms-win-core-heap-l1-1-0.dll, api-ms-win-core-interlocked-l1-1-0.dll, api-ms-win-core-libraryloader-l1-1-0.dll, api-ms-win-core-localization-l1-2-0.dll, api-ms-win-core-memory-l1-1-0.dll, api-ms-win-core-namedpipe-l1-1-0.dll, api-ms-win-core-processenvironment-l1-1-0.dll, api-ms-win-core-processthreads-l1-1-0.dll, api-ms-win-core-processthreads-l1-1-1.dll, api-ms-win-core-profile-l1-1-0.dll, api-ms-win-core-rtlsupport-l1-1-0.dll, api-ms-win-core-string-l1-1-0.dll, api-ms-win-core-synch-l1-1-0.dll, api-ms-win-core-synch-l1-2-0.dll, api-ms-win-core-sysinfo-l1-1-0.dll, api-ms-win-core-timezone-l1-1-0.dll, api-ms-win-core-util-l1-1-0.dll, api-ms-win-crt-conio-l1-1-0.dll, api-ms-win-crt-convert-l1-1-0.dll, api-ms-win-crt-environment-l1-1-0.dll, api-ms-win-crt-filesystem-l1-1-0.dll, api-ms-win-crt-heap-l1-1-0.dll, api-ms-win-crt-locale-l1-1-0.dll, api-ms-win-crt-math-l1-1-0.dll, api-ms-win-crt-multibyte-l1-1-0.dll, api-ms-win-crt-private-l1-1-0.dll, api-ms-win-crt-process-l1-1-0.dll, api-ms-win-crt-runtime-l1-1-0.dll, api-ms-win-crt-stdio-l1-1-0.dll, api-ms-win-crt-string-l1-1-0.dll, api-ms-win-crt-time-l1-1-0.dll, api-ms-win-crt-utility-l1-1-0.dll, audits.dll, concrt140.dll, libeay32.dll, libuv.dll, msvcp140.dll, mxcore.dll, ssleay32.dll, ucrtbase.dll, vcruntime140.dll, zlib1.dll*].

5.2.7 Trusted Path/Channel (FTP)

FTP_DIT_EXT.1 Protection of Data in Transit

FTP_DIT_EXT.1.1

The application shall [

encrypt all transmitted data with [TLS]

] between itself and another trusted IT product.

5.3 TOE SFR Dependencies Rationale for SFRs

The Protection Profile for Application Software contains all the requirements claimed in this Security Target. As such, the dependencies are not applicable since the PP has been approved.

5.4 Security Assurance Requirements

The TOE assurance requirements for this ST are taken directly from the Protection Profile for Application Software which are derived from Common Criteria Version 3.1, Revision 4. The assurance requirements are summarized in the table below.

Assurance Class	Components	Components Description
Development	ADV_FSP.1	Basic Functional Specification
Guidance Documents	AGD_OPE.1	Operational User Guidance
	AGD_PRE.1	Preparative User Guidance
Life Cycle Support	ALC_CMC.1	Labeling of the TOE
	ALC_CMS.1	TOE CM Coverage
	ALC_TSU_EXT.1	Timely Security Updates
Tests	ATE_IND.1	Independent Testing – Conformance
Vulnerability Assessment	AVA_VAN.1	Vulnerability Analysis

Table 9 Security Assurance Requirements

5.5 Rationale for Security Assurance Requirements

The functional specification describes the external interfaces of the TOE; such as the means for a user to invoke a service and the corresponding response of those services. The description includes the interface(s) that enforces a security functional requirement, the interface(s) that supports the enforcement of a security functional requirement, and the interface(s) that does not enforce any security functional requirements. The interfaces are described in terms of their purpose (general goal of the interface), method of use (how the interface is to be used), parameters (explicit inputs to and outputs from an interface that control the behavior of that interface), parameter descriptions (tells what the parameter is in some meaningful way), and error messages (identifies the condition that generated it, what the message is, and the meaning of any error codes). The development evidence also contains a tracing of the interfaces to the SFRs described in this ST.

5.6 Assurance Measures

The TOE satisfies the identified assurance requirements. This section identifies the Assurance Measures applied by FireEye to satisfy the assurance requirements. The table below lists the details.

SAR Component	How the SAR will be met
ADV_FSP.1	The functional specification describes the external interfaces of the TOE; such as the means for a user to invoke a service and the corresponding response of those services. The description includes the interface(s) that enforces a security functional requirement, the interface(s) that supports the enforcement of a security functional requirement, and the interface(s) that does not enforce any security functional requirements. The interfaces are described in terms of their purpose (general goal of the interface), method of use (how the interface is to be used), parameters (explicit inputs to and outputs from an interface that control the behavior of that interface), parameter descriptions (tells what the parameter is in some meaningful way), and error messages (identifies the condition that generated it, what the message is, and the meaning of any error codes).
AGD_OPE.1	The Administrative Guide provides the descriptions of the processes and procedures of how the administrative users of the TOE can securely administer the TOE using the interfaces that provide the features and functions detailed in the guidance.
AGD_PRE.1	The Installation Guide describes the installation, generation, and startup procedures so that the users of the TOE can put the components of the TOE in the evaluated configuration.
ALC_CMC.1	The Configuration Management (CM) documents describe how the consumer identifies the evaluated TOE. The CM documents identify the configuration items, how those configuration items are uniquely identified, and the adequacy of the procedures that are used to control and track changes that are made to the TOE. This includes details on what changes are tracked and
ALC_CMS.1	

SAR Component	How the SAR will be met
	how potential changes are incorporated.
ALC_TSU_EXT.1	Users of the FireEye X-Agent should report any security related issues via the FireEye webpage (https://www.fireeye.com/support.html), which provides a secure channel. Software updates/fixes are also provided via the FireEye webpage. Public availability of an update for a publicly disclosed vulnerability is typically 90 days or less and a maximum of 180 days.
ATE_IND.1	FireEye will provide the TOE for testing.
AVA_VAN.1	FireEye will provide the TOE for testing.

Table 10 TOE Security Assurance Measures

6 TOE Summary Specification

This chapter identifies and describes how the Security Functional Requirements identified above are met by the TOE.

TOE SFR	Rationale
FCS_CKM_EXT.1	The TOE does not generate RSA keys as it is not required for TLS sessions as the TOE acts as a receiver of keys from the server.
FCS_CKM.2	For RSA Key Establishment, the TOE implements sections 7.1 and 7.2.1 of SP 800-56B. The TOE does not perform any operation marked as “Shall Not” or “Should not” in SP 800-56B. Additionally, the TOE does not omit any operation marked as “Shall.” The TOE is a TLS client (i.e. sender), so it does not perform RSA decryption.
FCS_COP.1(1)	The TOE provides symmetric encryption and decryption capabilities using AES in CBC mode (128 and 256 bits) as described in NIST SP 800-38A. See Table 2 for validation details.
FCS_COP.1(2)	The TOE provides cryptographic hashing services using SHA-1 and SHA-256 with message digest sizes 160 and 256 bits respectively, as specified in FIPS Pub 180-4 “Secure Hash Standard.” These hashes are used as part of TLS session negotiation and with HMACs used to verify the integrity of TLS traffic. The hash functions are also used in conjunction with RSA as part of the HX server certificate verification. See Table 2 for validation details.
FCS_COP.1(3)	The TOE provides cryptographic signature services using RSA Digital Signature Algorithm with key size of 2048 and greater as specified in FIPS PUB 186-4, “Digital Signature Standard”. See Table 2 for validation details.
FCS_COP.1(4)	The TOE provides keyed-hashing message authentication services using HMAC-SHA-1 and SHA-256 with key size and message digests sizes of 160 and 256 bits respectively, as specified in FIPS Pub 198-1, "The Keyed-Hash Message Authentication Code,” and FIPS 180-4, “Secure Hash Standard.” See Table 2 for validation details.
FCS_RBG_EXT.1	The TOE implements DRBG functionality.
FCS_RBG_EXT.2	The TOE provides random bit generation services using an SP 800-90A CTR_DRBG using AES-256. The DRBG is seeded with at least 256 bits of entropy from the DRBG provided by the Windows platform through the CryptGenRandom API. See Table 2 for validation details.
FCS_STO_EXT.1	The TOE stores an X.509 CA certificate (and embedded public key) in a JSON structure stored in non-volatile memory. The JSON structure is encrypted using the TOE provided AES algorithm and is not accessible to any external entity. The CA certificate is used to validate the HX server certificate. The TOE does not store other secret keys, PKI private keys, passwords or other public keys.
FCS_TLSC_EXT.1	In support of secure communication with external entities, the TOE supports the TLS protocol. TLS is used to facilitate communication with the following entities, • HX Series Appliances • The TOE only communicates with the HX appliance using TLS_RSA_WITH_AES_128_CBC_SHA X.509 certificates used for this connection are validated using the certificate path validation algorithm defined in RFC 5280. The TOE supports CN-ID (with a DNS name) and DNS-ID (i.e. DNS name in the SAN) reference identifiers. For both types of identifiers, the TOE supports exact matching and wildcard when the wildcard is the entire left-most label. The TOE does not support IP address referenced identifiers or support certificate pinning.
FDP_DEC_EXT.1 FDP_NET_EXT.1	The TOE never processes or sends PII data outside the boundary of the host platform. The only external communication that is supported by the TOE is with the associated HX appliance. The communication consists of a fast-polling channel on port 80 which is used to receive a Boolean value about whether there are further instructions to receive. If there are, a TLS-protected channel on Port 443 is initiated with the HX and instructions or

TOE SFR	Rationale
	updates are transferred via the TLS session. This channel is used to download new scanning policies. The TOE then acts on these policies (e.g., performing scans on the platform). These downloaded policies may also include instructions to send the results of the scanning to the associated HX. In these cases, the TOE again initiates a TLS-protected channel on Port 443 as before. The TOE never accesses any other host platform hardware functionality besides network connectivity. Depending upon the contents of the policies the TOE receives from the associated HX appliance, the TOE may access the host OS syslog. The contents of memory and the filesystem are scanned as well, leveraging the functionality provided by a kernel driver (fekern.sys) which is installed with the TOE.
FDP_DAR_EXT.1	The only information that is stored by the TOE are the policies, the TOE identity, and associated HX identity which are downloaded from the associated HX series appliance. This data is protected by the Windows platform using the OS provided services. The provided guidance documentation provides instructions to ensure that BitLocker is enabled in the evaluated configuration. No other data is stored by the X-Agent.
FIA_X509_EXT.1	The TOE uses X.509v3 certificates as defined by RFC 5280 to support authentication for TLS connections. The X.509 certificates are validated by the FireEye OpenSSL Cryptographic Library during the TLS handshake when it received the TLS server certificate from the HX server. The X.509 certificates are validated using the certificate path validation algorithm defined in RFC 5280, which can be summarized as follows: • the public key algorithm and parameters are checked • the current date/time is checked against the validity period • revocation status is checked • issuer name of X matches the subject name of X+1 • name constraints are checked • policy OIDs are checked • policy constraints are checked; issuers are ensured to have CA signing bits • path length is checked • critical extensions are processed The TOE does not support intermediate CAs. The HX server certificate must be issued by the CA trusted by the TOE. The TOE accepts only CRL files managed by the PKI service on the HX Management Console to determine whether HX certificates have been revoked. If the PKI service is down and the CRL is unavailable, the TOE rejects the connection.
FIA_X509_EXT.2	
FMT_MEC_EXT.1	The X-Agent software does not provide any Security Relevant configuration options for the software. The software is an agent that installs on the host systems OS. Once installed, the product only allows very limited interaction with the host OS user. The TOE stores the settings configured during installation in the C:\ProgramData\FireEye\xagt directory.
FMT_CFG_EXT.1	The TOE does not require any credentials to be configured. The TOE does not authenticate the users of the host OS and the HX server certificate is provided in each TLS connection. The TOE utilizes a CA certificate that is part of the TOE installation. No other functionality is available until after the TOE is installed on the host platform. No modifications may be made to the X-Agent or its associated data by any unprivileged user of the host platform.
FMT_SMF.1	The TOE is pushed to the host platform by the HX appliance completely configured. At no time does the TOE user perform any management of the software.
FPR_ANO_EXT.1	The TOE does not transmit PII over the network.
FPT_API_EXT.1	The TOE leverages the following platform provided Application Programing Interfaces (APIs): ADVAPI32.dll, apphelp.dll, bdcore.dll, CFGMGR32.dll, CLBCatQ.dll, comctl32.dll,

TOE SFR	Rationale
	credssp.dll, CRYPT32.dll, CRYPTBASE.dll, CRYPTSP.dll, cscapi.dll, dbghelp.dll, DEVOBJ.dll, dhcpcsvc.dll, DNSAPI.dll, fastprox.dll, FLTLIB.dll, fwpuclnt.dll, GDI32.dll, imagehlp.dll, IMM32.dll, IPHLPAPI.dll, kernel32.dll, KERNELBASE.dll, LPK.dll, MSASN1.dll, MSCTF.dll, msi.dll, mssprxy.dll, msvcrt.dll, mswsock.dll, NETAPI32.dll, netutils.dll, NSI.dll, ntdll.dll, NTDSAPI.dll, ntmarta.dll, ntshrui.dll, ODBC32.dll, odbcint.dll, ole32.dll, OLEAUT32.dll, pcwum.dll, pdh.dll, perfdisk.dll, profapi.dll, PROPSYS.dll, PSAPI.dll, rasadhlp.dll, RPCRT4.dll, RpcRtRemote.dll, rsaenh.dll, sechost.dll, Secur32.dll, SETUPAPI.dll, SHELL32.dll, SHLWAPI.dll, slc.dll, srvcli.dll, SspiCli.dll, tdh.dll, USER32.dll, USERENV.dll, USP10.dll, VERSION.dll, wbemcomn.dll, wbemprox.dll, wbemsvc.dll, webio.dll, winhttp.dll, WINNSI.dll, WINSTA.dll, WINTRUST.dll, wkscli.dll, WLDAP32.dll, WS2-32.dll, wship6.dll, wshtcpip.dll, and WTSAPI32.dll.
FPT_AEX_EXT.1	The TOE never allocates memory with both write and execute permission. Write execution is always separate from execute. The TOE is designed to operate in an environment in which the following security techniques are in effect, Data execution prevention, Mandatory address space layout randomization (no memory map to an explicit address), Structured exception handler overwrite protection, Export address table access filtering, and Anti-Return Oriented Programming. This allows the TOE to operate in an environment in which the Enhanced Mitigation Experience Toolkit is also running. TOE executables are written to “C:\Program Files (x86)\FireEye\xagt”, in which no other files are written. In particular, no executable files are co-located in the directory in which the software is installed. During compilation the TOE is built with several flags enabled that check for engineering flaws. The TOE is built with the /GS flag enabled. This reduces the possibilities of stack-based buffer overflows in the product. The compiler enables ASLR by default. The TOE is not compiled with the /DYNAMICBASE:NO which would disable ASLR.
FPT_TUD_EXT.1	The TOE, initial installation as well as updates, is distributed as a Microsoft .MSI file. The TOE software version can be queried via the Microsoft command prompt by invoking X-Agent with the -v parameter. TOE updates are signed using digital certificates. The MSI packages are signed using certificates with a public trust chain which leads to Entrust. Some components of the installation package (for instance, containment driver), are signed using the Mandiant/FireEye internal CA. Updates are distributed as .MSI files provided by the associated HX appliances. The TOE provides the ability to completely remove all application files when uninstalled. The user can use the platform provided web browser to query the HX series appliance to determine if an update is available. Users of the FireEye X-Agent should report any security related issues via the FireEye webpage (https://www.fireeye.com/support.html), which provides a secure channel. Software updates/fixes are also provided via the FireEye webpage. Public availability of an update for a publicly disclosed vulnerability is typically 90 days or less and a maximum of 180 days.
FPT_LIB_EXT.1	The following supporting DLLs are brought with the TOE into the installation directory: • api-ms-win-core-console-l1-1-0.dll • api-ms-win-core-datetime-l1-1-0.dll • api-ms-win-core-debug-l1-1-0.dll • api-ms-win-core-errorhandling-l1-1-0.dll • api-ms-win-core-file-l1-1-0.dll • api-ms-win-core-file-l1-2-0.dll • api-ms-win-core-file-l2-1-0.dll • api-ms-win-core-handle-l1-1-0.dll • api-ms-win-core-heap-l1-1-0.dll • api-ms-win-core-interlocked-l1-1-0.dll • api-ms-win-core-libraryloader-l1-1-0.dll • api-ms-win-core-localization-l1-2-0.dll

TOE SFR	Rationale
	• api-ms-win-core-memory-l1-1-0.dll • api-ms-win-core-namedpipe-l1-1-0.dll • api-ms-win-core-processenvironment-l1-1-0.dll • api-ms-win-core-processthreads-l1-1-0.dll • api-ms-win-core-processthreads-l1-1-1.dll • api-ms-win-core-profile-l1-1-0.dll • api-ms-win-core-rtlsupport-l1-1-0.dll • api-ms-win-core-string-l1-1-0.dll • api-ms-win-core-synch-l1-1-0.dll • api-ms-win-core-synch-l1-2-0.dll • api-ms-win-core-sysinfo-l1-1-0.dll • api-ms-win-core-timezone-l1-1-0.dll • api-ms-win-core-util-l1-1-0.dll • api-ms-win-crt-conio-l1-1-0.dll • api-ms-win-crt-convert-l1-1-0.dll • api-ms-win-crt-environment-l1-1-0.dll • api-ms-win-crt-filesystem-l1-1-0.dll • api-ms-win-crt-heap-l1-1-0.dll • api-ms-win-crt-locale-l1-1-0.dll • api-ms-win-crt-math-l1-1-0.dll • api-ms-win-crt-multibyte-l1-1-0.dll • api-ms-win-crt-private-l1-1-0.dll • api-ms-win-crt-process-l1-1-0.dll • api-ms-win-crt-runtime-l1-1-0.dll • api-ms-win-crt-stdio-l1-1-0.dll • api-ms-win-crt-string-l1-1-0.dll • api-ms-win-crt-time-l1-1-0.dll • api-ms-win-crt-utility-l1-1-0.dll • audits.dll • concrt140.dll • libeay32.dll • libuv.dll • msvcp140.dll • mxcore.dll • ssleay32.dll • ucrtbase.dll • vcruntime140.dll • zlib1.dll
FTP_DIT_EXT.1	The TOE communicates externally with one trust IT entity, the FireEye HX Series appliances. The X-Agent periodically polls the HX appliance for policy updates. To do this the TOE initiates a TLS 1.2 secured tunnel using the TOE cryptographic implementation. Updates to the scanning policies are sent through this TLS 1.2 tunnel. No additional information is sent from the TOE.

Table 11 TOE Summary Specification SFR Description

End of Document