Network Camera Threat Model and Security Analysis (English language Protection Profile)

Architecture & Technology Group

Document number:	DEN0073	Version:	BET00
Date of Issue:	13/02/2018		
Author:	Arm Ltd		
Authorised by:	ATG		

Abstract

Security should start with a Threat Model and Security Analysis (TMSA) that lists the assets that need protection in a system and the threats that are considered in scope. From this starting point, a step by step process can be used to establish security objectives and Security Functional Requirements (SFRs). With the inherent diversity of IoT there will be a greater need for device manufacturers to have a reference TMSA for their product. Arm has created a series of reference English language Protection Profiles for IoT products to show how this might be done in a way that is understandable by non-security experts. These security analyses are accompanied by at a glance summary documents and useful appendices that show how Arm TrustZone and CryptoIsland technology can be used to meet some of the SFRs. We hope that you find these documents useful as a starting point for creating a TMSA for your IoT device.

Keywords

Network Camera, Platform Security Architecture, PP, Protection Profile, PSA, Threat Model Security Analysis, TMSA, TrustZone

Distribution list

Name	Function		Name	Function

Non-Confidential Proprietary Notice

Contents

1 About this document

1.1 PP Identification

Title: Network Camera Protection Profile

Authors: Arm Ltd
CC Version: 3.1 revision 5
Assurance Level: EAL 2
Reference:
Version Number:
Keywords: Network Camera

1.2 Change control

This document is tracked in SharePoint internally.

1.3 Current status and anticipated changes

Current Status: Beta

1.4 Change history

Release Date	Version	Comments
10/11/2017	0.1	First complete version
01/12/2017	0.2	Revision after Arm comments
26/12/2017	0.3	Added a TrustZone Support Appendix
16/01/2018	0.4	Fixes and template modification

1.5 References

This document refers to the following documents.

Ref	Doc No	Author(s)	Title
[CC-1]	CCMB-2017-04-001		Common Criteria for Information Technology Security Evaluation, Version 3.1, Revision 5, April 2017. Part 1: Introduction and general model.
[CC-2]	CCMB-2017-04-002		Common Criteria for Information Technology Security Evaluation, Version 3.1, Revision 5, April 2017. Part 2: Security functional components

[CC-3]	CCMB-2017-04-003		Common Criteria for Information Technology Security Evaluation, Version 3.1, Revision 5, April 2017. Part 3: Security assurance components
[CEM]	CCMB-2017-04-004		Common Methodology for Information Technology Security Evaluation (CEM), Version 3.1, Revision 5, April 2017. Evaluation methodology
[Comp]			Joint Interpretation Library, Composite product evaluation for Smart Cards and similar devices, Version 1.2, January 2012
[GPRoT]	GP_REQ_025	GlobalPlatform	Root of Trust Definitions and Requirements, March 2017, Version 1.0.1

1.6 Terms

This document uses the following terms and abbreviations.

Term	Meaning
API	Application Programming Interface
CC	Common Criteria
CPU	Central Processing Unit
EAL	Evaluation Assurance Level
LED	Light Emitting Diode
HTTPS	HyperText Transfer Protocol Secure
IPSec	Internet Protocol Security
NTP	Network Time Protocol
OS	Operating System
OSP	Organisational Security Policy
OTP	One-Time-Programmable
PP	Protection Profile
PTZ	Pan-Tilt-Zoom
RAM	Random Access Memory
REE	Rich Execution Environment
ROM	Read Only Memory
SFP	Security Function Policy
SFR	Security Functional Requirement

SoC	System-on-Chip
ST	Security Target
TEE	Trusted Execution Environment
TLS	Transport Layer Security
TOE	Target of Evaluation
TSF	TOE Security Functionality
TSFI	TSF Interface
TSS	TOE Security Service

1.7 Terminology and Definitions

1. The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [RFC 2119]:

MUST: This word, or the terms "REQUIRED" or "SHALL", mean that the definition is an absolute requirement of the specification.

MUST NOT: This phrase, or the phrase "SHALL NOT", mean that the definition is an absolute prohibition of the specification.

SHOULD: This word, or the adjective "RECOMMENDED", mean that there may exist valid reasons in particular circumstances to ignore a particular item, but the full implications must be understood and carefully weighed before choosing a different course.

SHOULD NOT: This phrase, or the phrase "NOT RECOMMENDED" mean that there may exist valid reasons in particular circumstances when the particular behaviour is acceptable or even useful, but the full implications should be understood and the case carefully weighed before implementing any behaviour described with this label.

MAY: This word, or the adjective "OPTIONAL", mean that an item is truly optional. One vendor may choose to include the item because a particular marketplace requires it or because the vendor feels that it enhances the product while another vendor may omit the same item. An implementation which does not include a particular option MUST be prepared to interoperate with another implementation which does include the option, though perhaps with reduced functionality. In the same vein an implementation which does include a particular option MUST be prepared to interoperate with another implementation which does not include the option (except, of course, for the feature the option provides.)

2 Introduction

2. This section provides an overview of the TOE.

2.1 TOE Overview

2.1.1 TOE Type

3. The TOE of this PP is a network-connected camera, such as used in homes and offices, with some processing capabilities to connect autonomously to a network. It may also include some local analysis of the pictures.

4. The TOE is a platform composed of a hardware device and a firmware implementing the network camera functionalities. The firmware itself may include a generic purpose operating system.

2.1.2 TOE Usage and Major Security Features

5. Network cameras are used to stream live video. In order to reduce required bandwidth, this usually means that the camera will have enough computing power to encode the video stream in a compressed form.

6. The cameras considered in this PP require a network connection. As live video streaming requires significant bandwidth, most network cameras are either connected through a cable or through Wi-Fi, typically on a local network.

7. There are many possible uses for network cameras, corresponding to very different security contexts, that we can abstract as follows:

Personal use, general purpose. From babyphones to security cameras, event detection and privacy protections are essential, but the achievable level of security assurance is limited by cost constraints.

Enterprise, general purpose. With a traditional security camera, in a protected environment, event detection and video flow integrity are essential. Risks are limited, so the level of security assurance does not need to be maximal.

Enterprise, high security. When a camera is highly exposed, or when it is used to protect high-value assets, the same feature are essential, but the level of security assurance must be significantly higher, even if it drives up the costs.

8. Despite these differences, the security features to be included in cameras are similar enough to be described in a single Protection Profile, as the appropriate security assurance can be personalized in a camera's Security Target.

9. The video stream and access to management and administration interfaces, network cameras include at least the following security features:

User and admin authentication. Users must be authenticated before to access the camera, and before to modify its configuration or perform maintenance operations on it. Local and network authentication may rely on different methods and credentials.

Authorization. Some functions are restricted to a limited number of users, and may only be available in some conditions (locally, for instance).

Network authentication. The establishment of a network connection requires a mutual authentication between the device and the remote server or user.

Encryption of video stream. The video stream can be encrypted with a key that is only shared with intended recipients (servers or users).

Secure communication. More generally, any network communication is performed using a protocol that includes integrity and confidentiality protections.

Log of security events. Security events are logged locally on the camera, to be made available in the forensic analysis of an attack or other suspicious event.

Software update. The software running on the camera can be updated in order to fix vulnerabilities identified after the device's deployment.

2.1.3 Required non-TOE Hardware/Software/Firmware

10. The TOE is embedded in a hardware device (the camera) that includes hardware that is not part of the TOE but that is used by the TOE, such as the sensor, the network interface or other hardware. Security functionalities shall not depend on that hardware. The ST writer shall make explicit which hardware is part of the TOE and which is not.

2.2 TOE Description

11. The figure below illustrates the main components for a network camera and the TOE for this PP.

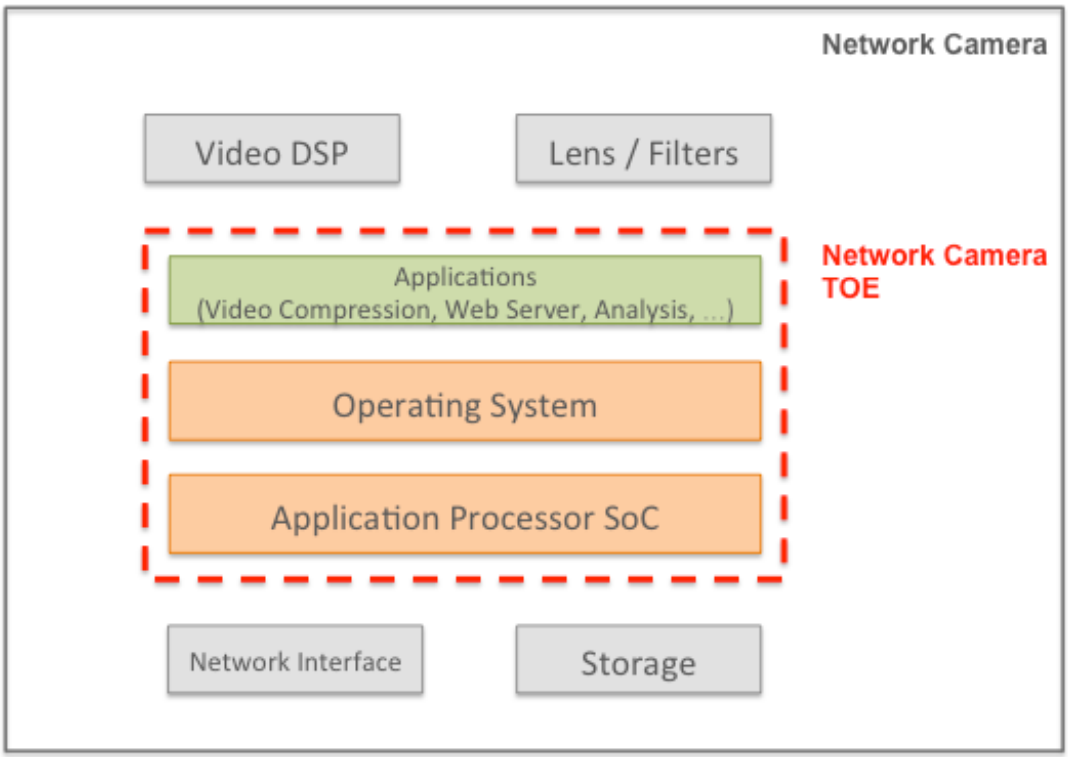

Figure 1: Network Camera TOE

2.2.1 TOE Features

2.2.1.1 Hardware

12. Hardware for a Network Camera is typically composed of a SoC with hardware video encoders, flash memory, network controller (Wi-Fi and/or Ethernet) and the camera itself that may also include infrared LEDs, motors for pan-tilt-zoom or movable filters.

13. The SoC may support OTPs to store sensitive data, such as camera ID or secrets.

2.2.1.2 Firmware

14. Firmware for a Network Camera is typically composed of a boot-loader, which is the first piece of code called by the ROM, an operating system, such as Linux, with drivers for controlling camera peripherals and applications running on top of this OS.

15. The applications may include for instance a motion sensor, a video uploader, web server for remote control of the camera.

16. Firmware is usually stored on a flash memory to support upgrade.

2.2.2 TOE Operational Environment

17. The TOE operational environment is composed of the place this camera is used (such as office or home) and the network the camera is connected to.

2.2.3 TOE Life Cycle

18. The TOE Life Cycle is as follows:

Phase	Actors
1 & 2: Firmware / Software / Hardware design	The network camera software developer is in charge of software development and testing. The device manufacturer may design additional software that will be linked with the network camera in phase 4. The network camera hardware designer is in charge of designing (part of) the processor(s) where the network camera software runs and designing (part of) the hardware security resources used by the network camera. The silicon vendor designs the ROM code and the secure portion of the network camera chipset.
3: Silicon/chip manufacturing	The silicon vendor produces the chipset for the network camera device.
4: Software manufacturing	The device manufacturer is responsible for the integration, validation, and preparation of the software to load in the product that will include the network camera.
5: Device manufacturing	The device manufacturer is responsible for the device assembly and initialization and any other operation on the device before delivery to the end user.
6: Operational phase	The end user gets a device ready for use. The end-user personalizes TOE and network credentials prior use. The network camera may be updated if it has not been designed to be immutable.

7: End-usage termination	The end user terminates their relationship to allow device resale by performing a factory reset of the network camera.

19. The TOE delivery point may occur at the end of phases 3, 4 or 5.

3 Conformance Claims

3.1 CC Conformance Claim

20. This Protection Profile is CC Part 2 [CC2] and CC Part 3 [CC3] conformant of Common Criteria version 3.1, revision 5.

3.2 Package Claim

21. The minimum assurance level for the evaluation of a Network Camera with a TOE conformant to this PP is EAL 2.

3.3 PP Claim

22. This Protection Profile does not claim conformance to any other Protection Profile.

3.4 Conformance Claim to this PP

23. The conformance to this PP, required for the Security Targets and Protection Profiles claiming conformance to it, is demonstrable, as defined in CC Part 1 [CC1].

4 Security Problem Definition

4.1 Users and External Entities

24. The external entities that are considered in this PP are the User and the Admin.

25. The User can access the video stream after authentication.

26. The Admin can modify the camera configuration, perform firmware update and access logs after authentication.

4.2 Assets

4.2.1 TSF Data

27. The following assets contain data that belong to TSF.

4.2.1.1 Camera ID

28. A unique ID to identify the device on a network, such as a MAC address.

29. Properties: Integrity

4.2.1.2 Firmware

30. The camera's firmware.

31. Properties: Integrity, Authenticity

4.2.1.3 Firmware Certificate

32. The cryptographic certificate used to authenticate firmware and firmware updates.

33. Properties: Integrity

4.2.1.4 Logs

34. The event logs, that can be used to detect suspicious activities.

35. Properties: Integrity

4.2.2 User Data

4.2.2.1 Video Stream

36. The video stream produced by the camera sent over the network.

37. Properties: Integrity, Confidentiality

4.2.2.2 Configuration

38. The camera's dynamic configuration, including network configuration such as the name of a WLAN network, or IP and DNS addresses and camera settings such as pan, tilt, and zoom, the events to be detected and notified.

39. Properties: Integrity

4.2.2.3 Credentials

40. The authentication credentials, used for local and remote authentication, such as:

    Network credentials, to authenticate if needed on the network, for instance a Wi-Fi pre-shared key or a 802.1x certificate, to be protected in integrity and confidentiality.

    Device authentication credentials to authenticate on remote servers, to be protected in integrity and confidentiality.

    Server authentication data, such as public key certificates, to be protected in integrity.

    Session keys, used after establishment of a trusted communication channel with servers, to be protected in integrity and confidentiality.

    Administration and user credentials, to authenticate to the services provided by the network camera, either for administration or for regular use, to be protected in integrity and confidentiality.

    User biometric patterns to be used in face recognition or similar algorithms, to be protected in integrity.

41. Properties: Integrity, Confidentiality

4.2.3 Others

42. Although assets of this section are not informational assets, but rather resources available to the TOE, they may be the direct targets of attackers.

4.2.3.1 Computing Power

43. The processing capabilities of the TOE, as provided by its central and possibly graphic processing units.

4.2.3.2 Network Bandwidth

44. The network resources used by the TOE to exchange data. As the TOE processes video, the volume of exchanged data may be significant.

4.2.3.3 Storage Space

45. The mass storage space used by the TOE to store data. As the TOE processes video, the volume of stored data may be significant.

4.3 Threats

46. An attacker is a threat agent (a person or a process acting on his/her behalf) trying to undermine the TOE security policy defined by the current ST and, hence, the TSF. The attacker especially tries to change properties of the assets defined in Section 4.2.

4.3.1 T.IMPERSONATION

47. An attacker impersonates a legitimate user on the camera, either a regular user that can access the video stream or an admin user.

48. The user credentials may be obtained through default admin passwords, interception, for instance in insecure communication links, or exposed through data disclosure.

49. The attacker may then access video stream, modify configuration or try to modify firmware.

50. Assets threatened directly: Credentials
Assets threatened indirectly: Video Stream, Configuration.

4.3.2 T.MITM

51. An attacker performs a Man-In-The-Middle attack or impersonates a server the camera connects to, for instance to upload the video stream or the event logs.

52. The attacker may rely on insecure communication links or prior modification of the server credentials on the camera through insecure configuration.

53. The attacker may then access and modify Video Stream, Logs, Credentials, Configuration data.

54. Assets threatened directly: Credentials (Server), Logs, Video Stream, Configuration

4.3.3 T.FIRMWARE_ABUSE

55. An attacker installs a flawed version of the firmware and obtains partial or total control of the camera. The firmware may have been modified prior to the attack to include a malware or consist of an outdated version of the original firmware.

56. The attacker may for instance modify on the device the value of the firmware certificate used to authenticate the installed firmware or firmware updates.

57. Such an attack can allow for elevation of privileges, where a regular user gains access to admin privileges.

58. This attack can also be used to take control over the TOE resources, for instance to carry a denial-of-service attack on other network devices, to store illegal files or to mine cryptocurrencies.

59. Assets threatened directly: Firmware, Firmware Certificate, Computing Power, Network Bandwidth, Storage Space.
Assets threatened indirectly: All.

4.3.4 T.TAMPER

60. An attacker tampers with the camera and tries to access or modify the media on which assets are stored.

61. This includes basic PCB attacks, after opening the camera case, such as eavesdropping buses, desoldering memory chips, use of debug interfaces.

62. Assets threatened directly: All.

4.4 Organisational Security Policies

63. The TOE and its environment shall comply with the following organizational security policies (OSP) as security rules, procedures, practices or guidelines imposed by an organization upon its operation.

4.4.1 P.CREDENTIALS_MANAGEMENT

64. The Admin shall change the default passwords of the TOE, if any, prior the operational usage of the TOE.

65. Additionally, the Admin and the User shall ensure confidentiality of their passwords.

4.5 Assumptions

66. This section describes the assumptions about the operational environment of the TOE.

4.5.1 A.TRUSTED_ADMIN

67. Admin of the TOE are assumed to follow and apply administrative guidance in a trusted manner.

5 Security Objectives

5.1 Security Objectives for the TOE

5.1.1 OT.ACCESS_CONTROL

68. The TOE shall authenticate User before granting access to the Video Stream.

69. The TOE shall authenticate Admin before granting access the camera configuration and logs and before performing firmware update.

5.1.2 OT.SECURE_STORAGE

70. The TOE shall protect integrity and confidentiality of Credentials when stored, and protect integrity of Firmware Certificate, Configuration and Logs when stored.

5.1.3 OT.FIRMWARE_AUTHENTICITY

71. The TOE shall authenticate and verify integrity of firmware image during boot and of new firmware versions prior upgrade.

72. The TOE shall also reject attempts of firmware downgrade.

5.1.4 OT.COMMUNICATION

73. The TOE shall be able to authenticate remote servers where Video Stream and Logs are uploaded and provide integrity and confidentiality protection for export outside of the TOE.

5.1.5 OT.AUDIT

74. The TOE shall maintain log of all significant events and allow access and analysis of these logs to authorized users only.

5.1.6 OT.SECURE_STATE

75. The TOE shall maintain a secure state even in case of failures, for instance failure of verification of firmware integrity.

5.2 Security Objectives for the Operational Environment

5.2.1 OE.CREDENTIALS_MANAGEMENT

76. Identical to P.CREDENTIALS_MANAGEMENT (p. 15).

5.2.2 OE.TRUSTED_ADMIN

77. The Admin of the TOE is not careless, wilfully negligent or hostile.

5.3 Security Objectives Rationale

78. The following table provides an overview for security objectives coverage (TOE and its environment) and also gives an evidence for sufficiency and necessity of the defined objectives. It shows that all threats and OSPs are addressed by the security objectives and it also shows that all assumptions are addressed by the security objectives for the TOE operational environment.

	OT.ACCESS_CONTROL	OT.SECURE_STORAGE	OT.FIRMWARE_AUTHENTICITY	OT.COMMUNICATION	OT.AUDIT	OT.SECURE_STATE	OE.CREDENTIALS_MANAGEMENT	OE.TRUSTED_ADMIN
T.IMPERSONATION	X				X		X	
T.MITM				X				
T.FIRMWARE_ABUSE	X		X			X		
T.TAMPER		X				X		
P.CREDENTIALS_MANAGEMENT	X						X	
A.TRUSTED_ADMIN								X

Table 1: Security Objectives Rationale

79. A justification required for suitability of the security objectives to cope with the security problem definition is given below.

5.3.1 Security Objective Rationales: Threats

5.3.1.1 Threat: T.IMPERSONATION

80. This threat assumes that the TOE can be attacked by impersonating of a legitimate user. This threat is countered by the security objectives OT.ACCESS_CONTROL that ensures authentication of users to access TOE functionalities and OT.AUDIT that allows for audit of TOE users activities and by the security objective on the operational environment OE.CREDENTIALS_MANAGEMENT that ensures that no default password can be used on operational usage.

5.3.1.2 Threat: T.MITM

81. This threat assumes that the TOE can be attacked by intercepting or spying communications with remote servers. This threat is countered by the security objective OT.COMMUNICATION that ensures authentication of remote servers and protection in confidentiality and integrity of exchanged data.

5.3.1.3 Threat: T.FIRMWARE_ABUSE

82. This threat assumes that the TOE can be attacked by modifying the firmware or installing and outdated flawed version. This threat is countered by the security objectives OT.ACCESS_CONTROL that ensures that only Admin can initiate firmware upgrade, OT.FIRMWARE_AUTHENTICITY that ensures verification of firmware authenticity prior use and prior upgrade and OT.SECURE_STATE that ensures that the TOE maintains a secure state even in case of failure of verification of firmware integrity.

5.3.1.4 Threat: T.TAMPER

83. This threat assumes that the TOE can be attacked by locally accessing TOE storage. This threat is countered by the security objectives OT.SECURE_STORAGE that ensures a secure storage for TOE assets and by OT.SECURE_STATE that ensures that the TOE maintains a secure state in case of failure of integrity checks.

5.3.2 Security Objective Rationales: Security Policies

84. Each identified security policy in this Security Target is addressed by at least one security objective for the TOE or security objective for the operational environment. This section provides a mapping from each security policy to the security objectives and provides a rationale how the security policy is fulfilled.

5.3.2.1 Policy: P.CREDENTIALS_MANAGEMENT

85. This security policy is directly upheld by the security objective on the operational environment OE.CREDENTIALS_MANAGEMENT.

5.3.3 Security Objective Rationales: Assumptions

86. Each security assumption in this Security Target is addressed by at least one security objective for the operational environment. This section maps assumptions to environmental security objectives and provides a rationale how the assumption is fulfilled.

5.3.3.1 Assumption: A.TRUSTED_ADMIN

87. This security policy is directly upheld by the security objective on the operational environment OE.TRUSTED_ADMIN.

6 Security Requirements

6.1 Security Functional Requirements

88. This part of the ST defines the detailed security functional requirements that are satisfied by the TOE.

89. These requirements are derived from the Security Objectives for the TOE (Section 5.1). Each sub-section is labelled with a security objective and provides the corresponding requirements.

90. As defined in Section 1.7, “shall” represent mandatory requirements, while “should” denotes requirements for which there may exists valid reasons to ignore them. However, if such a requirement is ignored, the full implications must be understood and the ST shall justify any removal of such requirements.

6.1.1 OT.ACCESS_CONTROL

91. The TOE shall maintain the roles Admin and User.

92. The TOE shall allow authentication of users according to these roles through user-initiated interactive sessions.

93. **Note 1**: The ST writer shall explicit how credentials for user authentication are managed on the TOE. This may include techniques to prevent weak passwords and also to protect them against disclosure (for instance use of salted hashes).

94. The TOE shall manage a threshold for unsuccessful authentication attempts. The ST writer shall precise the actions taken is this threshold is reached.

95. The TOE shall require each user to be successfully authenticated before allowing any other actions on behalf of that user.

96. The TOE shall allow termination of user’s own interactive session and automatically terminate a remote interactive session after session inactivity.

97. The TOE shall enforce an access control policy on TOE assets and operations based on the identity of the user requesting access. The ST writer shall define rules of this policy.

98. **Note 2**: This policy will typically include rules such as:

    Access to Configuration, Logs, Credentials assets is only allowed to authenticated users with role Admin.
    Access to Firmware upgrade operations is only allowed to authenticated users with role Admin.
    Access to video stream assets is only allowed to authenticated users with role User.

99. The TOE shall prevent unauthorized uses of all assets. In particular, the TOE shall prevent reading of all Credentials and shall not provide an interface to do so.

6.1.2 OT.SECURE_STORAGE

100. The TOE shall monitor for integrity errors assets with a security need for integrity (Camera ID, Firmware, Firmware Certificate, Logs, Configuration, Credentials).

101. **Note 3**: The TOE will typically ensure integrity either with hardware based write-once mechanisms, such as OTP, or through cryptographic hash functions. In the latter case, the ST writer shall explicit the cryptographic algorithms used for secure storage and related key characteristics and random generation methods.

102. Upon detection of a data integrity error, the TOE shall maintain a secure state. The ST writer shall specify reaction of the TOE in this case.

103. **Note 4**: For assets with a security need for confidentiality (Credentials), protection of relies on access control measures (OT.ACCESS_CONTROL). However the TOE may offer additional protection by encryption of persistent memory. The ST writer shall specify the mechanism used and related encryption techniques.

6.1.3 OT.FIRMWARE_AUTHENTICITY

104. The TOE shall rely on a secure boot mechanism to authenticate and verify integrity of firmware prior transferring control to the firmware.

105. **Note 5**: A secure boot will typically rely on a multi-stage boot process where the authenticity of the first stage is assumed from read-only memory and other stages with verification of cryptographic signatures with asymmetric keys. The ST writer shall explicit which signature schemes are used at the various stages, including the hash algorithm, and the size of the various parameters (e.g., modulus of 2048 bits and exponent of 32 bits for RSASSA-PSS with SHA-512). He shall also specify the list of standards that are met by the chosen schemes or none.

106. If the firmware is loaded from a removable media, the TOE shall use a persistent storage to store the version of the last installed firmware and compare this version to the version from the loaded firmware to prevent loading of an out-dated firmware.

107. Upon detection of a firmware authenticity error, the TOE shall maintain a secure state. The ST writer shall specify the action to be taken if the verification fails (cf. OT.SECURE_STATE).

108. **Note 6**: The TOE may enter a maintenance mode where the ability to return a secure state is provided.

109. On firmware upgrade requests, the TOE shall first authenticate the upgrade binary based on digital signature and verify its integrity. The TOE shall also check that version of the firmware for upgrade is more recent than the firmware currently installed.

110. **Note 7**: The ST writer shall explicit which signature scheme is used.

111. Upon detection of an error during upgrade, the TOE shall revert to the version of the firmware prior the upgrade request.

112. The TOE should provide the ability to check availability of firmware upgrade and notify Admin.

6.1.4 OT.COMMUNICATION

113. The TOE shall establish a trusted communication channel with remote servers prior any exchange of TSF data or User data and verify if the peer certificate is valid.

114. **Note 8**: Trusted communication channels include any of IPsec, TLS or HTTPS performed by the TOE. Validity of the peer certificate is determined by the certificate path, the expiration date, and the revocation status.

115. The TOE shall prevent the disclosure and modification of user data when exporting user data outside of the TOE.

116. **Note 9**: Protection of user data relies on the encryption techniques provided with the trusted communication channel. The ST writer shall explicit which encryption algorithms are used and related key sizes.

117. The TOE should support network authentication (e.g. Wi-Fi or IEEE 802.1X).

6.1.5 OT.AUDIT

118. The TOE shall maintain an audit trail of security events. Each record shall mention the nature of the event, date and time of the event and the user, if any, responsible for the event.

119. **Note 10**: The ST writer shall explicit which events are logged. This will include at least failed and successful authentication attempts, firmware upgrade requests and progress, integrity errors.

120. The TOE shall prevent users from deleting entries from the audit trail.

121. **Note 11**: The only audit trail operations and interfaces that should be available on the TOE are appending a line to the audit trail and export outside of the TOE.

122. The TOE should rely on a NTP server to provide reliable source for time stamps for the audit trail.

123. **Note 12**: If the TOE supports the use of a NTP server, the operational guidance for the TOE shall provide instructions for the Admin to configure the NTP client on the TOE.

6.1.6 OT.SECURE_STATE

124. The TOE shall maintain a secure state in case of failures, such as firmware integrity error, firmware upgrade error, RNG error, failure to establish a trusted communication channel.

125. **Note 13**: If the TOE should encounter a failure in the middle of a critical operation, the TOE should not just quit operating, leaving key material and user data unprotected. The ST writer shall specify

126. The TOE shall ensure residual information protection for credentials and session keys after they are being used.

6.2 Security Assurance Requirements

127. The current assurance package was chosen based on the pre-defined assurance packet EAL 2. EAL 2 is chosen because the threats that were chosen are consistent with an attacker of basic attack potential.

7 Acknowledgements

128. This document was prepared for Arm by Prove & Run
http://www.provenrun.com

Appendix A Support of SFRs by Arm CryptoIsland IP

129. This appendix explains how SFRs of this PP can be implemented using an Arm Cortex-A SoC embedding Arm CryptoIsland IP.

PP Requirement	Support from CryptoIsland IP
OT.ACCESS_CONTROL	
Authentication of User and Admin	Secure cryptographic and RNG support. This feature can be used to support cryptographic algorithms used for authentication.
Access control policy on assets	Data protection functionalities, in particular support for asset use policy. This feature can be used to implement an access control policy on TOE assets based on the identity of the requester and additionally on the lifecycle state, the intended usage, and HW interface used for the request
OT.SECURE_STORAGE	
Integrity and confidentiality protection for stored assets	Persistent trusted storage based on OTP and local storage protected by an encryption key (AES-256 key). This feature, that offers integrity and confidentiality protection, can be used to store assets. OTP will be reserved for immutable assets, such as the Camera ID, and local storage for other assets.
OT.FIRMWARE_AUTHENTICITY	
Verification of firmware authenticity prior boot	Loaded SW validation functionality that authenticates loaded images based on a hardware root of trust. This feature can be used as part of the secure boot process to verify firmware during device start-up.
Verification of firmware authenticity prior update	SW update validation. This feature can be used to verify integrity and authenticity of firmware update image. The firmware authenticate is based on a cryptographic signature with PKI. It reports failures during the update process and fails back on the last valid image.
Anti-rollback for firmware update	SW update validation. This feature can also verify freshness of firmware update image.
OT.COMMUNICATION	
Authentication of remote servers	Secure cryptographic and RNG support. This feature can be used to implement and support cryptographic protocols for communication. Related cryptographic keys can be stored in the persistent trusted storage provided by CryptoIsland IP.
Integrity and confidentiality protection for exchanged assed	
Replay protection	No direct support.
OT.AUDIT	
Audit trail of security events	No direct support.
Protection of audit trail	Persistent trusted storage functionality can be used to security store and control accesses to audit trails.

Secure timestamp	No direct support.
OT.SECURE_STATE	
Residual information protection for confidential assets	No direct support.
Protection of debug features	Authenticated debug functionality. Debug certificates can be used to protect and activate debug features of the processor.
Secure state in case of failure	Alarm signals handling. Possible reactions include for instance aborting current operation, resetting the processor, deactivating the device, zeroizing keys.
Self-tests	No direct support.

Appendix B Support of SFRs by Arm TrustZone IP

130. This appendix explains how SFRs of this PP can be implemented using an Arm Cortex-A SoC embedding Arm TrustZone IP.

131. The implementation make use of the Secure World provided by TrustZone to secure sensitive functions and assets and isolate them for the Normal World where the general purpose OS of the Network Camera is executed. In particular, a secure OS is required for the Secure World to support secure applications, for instance a micro-kernel or a Trusted Execution Environment (TEE).

PP Requirement	Support from TrustZone IP
OT.ACCESS_CONTROL	
Authentication of User and Admin	Secure application to manage authentication credentials, to support user authentication and to control access to credentials.
Access control policy on assets	
OT.SECURE_STORAGE	
Integrity and confidentiality protection for stored assets	Secure application for secure storage of assets.
OT.FIRMWARE_AUTHENTICITY	
Verification of firmware authenticity prior boot	No direct support.
Verification of firmware authenticity prior update	Secure application to manage firmware update credential and to support verification of firmware authenticity and freshness of firmware update image.
Anti-rollback for firmware update	
OT.COMMUNICATION	
Authentication of remote servers	Secure application to manage authentication and encryption credentials for communication, to support authentication and encryption.
Integrity and confidentiality protection for exchanged assed	
Replay protection	No direct support.
OT.AUDIT	
Audit trail of security events	Secure application for audit trail management, including secure storage and timestamp.
Protection of audit trail	
Secure timestamp	
OT.SECURE_STATE	
Residual information protection for confidential assets	No direct support.
Protection of debug features	No direct support.
Secure state in case of failure	Isolation of Normal World and Secure World execution environments.
Self-tests	No direct support.

Appendix C Compatibility with Root-of-Trust PP

132. The Root of Trust Protection Profile targets platforms that provide a set of trusted and basic functions or services from which an initial chain or trust can be derived. It is based on the GlobalPlatform *Root of Trust Definitions and Requirements* document [GPRoT]. The PP is a modular-PP, organized as a base-PP corresponding to the Root of Trust platform and PP-modules corresponding to optional security services based on top of this platform, such as authentication, confidentiality, authorization or update services.

133. This appendix explains how SFRs of this PP can inherit from the requirements set in the Root of Trust PP and related PP-modules.

PP Requirement	Support from a Root of Trust
OT.ACCESS_CONTROL	
Authentication of User and Admin	Root of Trust with an Authentication Service allows authenticating users.
Access control policy on assets	Root of Trust with an Authorization Service allows enforcing an access control policy on TOE assets.
OT.SECURE_STORAGE	
Integrity and confidentiality protection for stored assets	A Root of Trust with a Confidentiality and Integrity Services allows enforcing confidentiality and integrity of storage for TOE assets.
OT.FIRMWARE_AUTHENTICITY	
Verification of firmware authenticity prior boot	A Root of Trust with a Verification Service allows verifying the authenticity of firmware.
Verification of firmware authenticity prior update	A Root of Trust with an Update Service allows enforcing integrity and authenticity of firmware update.
Anti-rollback for firmware update	No direct support.
OT.COMMUNICATION	
Authentication of remote servers	Root of Trust with an Authentication Service allows authenticating remote entities.
Integrity and confidentiality protection for exchanged assed	No direct support.
Replay protection	No direct support.
OT.AUDIT	
Audit trail of security events	No direct support.
Protection of audit trail	No direct support.
Secure timestamp	No direct support.
OT.SECURE_STATE	
Residual information protection for confidential assets	No direct support.
Protection of debug features	No direct support.
Secure state in case of failure	No direct support.