DEKRA Testing and Certification, S.A.U.
C/Severo Ochoa 55, 29590, Málaga

C.I.F. A29 507 456

Test report No:
NIE: 80676CRCS.017

Security Evaluation Report
DEKRA Evaluation Framework

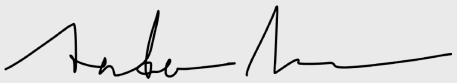
(*) Identification of item tested	Nest Thermostat 4th gen
(*) Trademark	Google LLC
(*) Model and/or type reference tested	GJQ8U
(*) Derived model not tested	Google Nest Learning Thermostat
Other identification of the product	HW: GJQ8U. SW: 2.1-25
(*) Features	WiFi, BLE
Manufacturer	Google LLC
Test method requested, standard	EN18031-1:2024
Summary	IN COMPLIANCE
Approved by (name/position & signature)	Anders Olof Möller, R&D Manager 
Date of issue	2025-11-17
Report template No	FCS270_03 (*) "Data provided by the client"

Table of Contents

Competences and Guarantees

In order to assure the traceability to other national and international laboratories, DEKRA Testing and Certification S.A.U. has a calibration and maintenance program for its measurement equipment.

DEKRA Testing and Certification S.A.U. guarantees the reliability of the data presented in this report, which is the result of the measurements and the tests performed to the item under test on the date and under the conditions stated on the report and, it is based on the knowledge and technical facilities available at DEKRA Testing and Certification at the time of performance of the test.

DEKRA Testing and Certification S.A.U. is liable to the client for the maintenance of the confidentiality of all information related to the item under test and the results of the test.

The results presented in this Test Report apply only to the particular item under test established in this document.

General Conditions

1. This report is only referred to the item that has undergone the test.
2. This report does not constitute or imply on its own an approval of the product by the Certification Bodies or competent Authorities.
3. This document is only valid if complete; no partial reproduction can be made without previous written permission of DEKRA Testing and Certification S.A.U.
4. This test report cannot be used partially or in full for publicity and/or promotional purposes without previous written permission of DEKRA Testing and Certification S.A.U. and the Accreditation Bodies.

Data Provided by the Client

The following data has been provided by the client:

1. Information relating to the description of the sample ("Identification of the item tested", "Trademark", "Model and/or type reference tested".

DEKRA Testing and Certification S.A.U. declines any responsibility with respect to the information provided by the client and that may affect the validity of results. The laboratory is not responsible for such information and it is not covered by accreditation.

Usage of Samples

The following samples have been provided by the customer to be evaluated:

The following elements do not belong to any sample:

Test Sample Description

	Description	File Name
Documents as Provided by the Applicant .:	Description of the way the mechanisms to change the authentication values are documented for the user, including all information to access the documentation. NOTE 2: Possible ways of documentation are the website of the manufacturer and the corresponding URL, the user manual or built-in help.	https://support.google.com/googlenest/answer/9248184?hl=en#Nest-Thermostat
	Description of the way the information about external sensing capabilities is documented for the user, including all information to access the documentation. NOTE 4: Possible ways of documentation are the website of the manufacturer and the corresponding URL, the user manual or built-in help.	https://store.google.com/product/nest_learning_thermostat_4th_gen_specs?hl=en-US
	Description of the way the information about processing personal data is documented for the user, including all information to access the documentation. NOTE 7: Possible ways of documentation are the website of the manufacturer and the corresponding URL, the user manual or built-in help.	https://support.google.com/googlenest/answer/9415830?hl=en&co=GENIE.Platform%3DAndroid
	Description of the way the methods for deletion of personal data documented to the user, including all information to access the documentation. NOTE 9: Possible ways of documentation are the website of the manufacturer and the corresponding URL, the user manual or built-in help.	https://support.google.com/googlenest/answer/9247296?hl=en
	The manufacturer needs to provide a Hardware Bill of Materials for the TL to verify there are no exploitable vulnerabilities in any hardware components. Proprietary ICs, and components that do not handle any data must not be included in the documentation.	Google Nest Learning Thermostat HBOM
	The manufacturer needs to provide a Software Bill of Materials for the TL to verify there are no exploitable	Google Nest Learning Thermostat SBOM

	vulnerabilities in any software components. Proprietary software components, as well as those licensed by 3rd parties, must not be included. On the other hand, every Open Source component must be included. Alternatively, an unencrypted version of the firmware may be provided instead of the SBOM.	

Identification of the Client

Company	Google LLC
Address	1600 Amphitheatre Parkway, Mountain View, California , U.S.

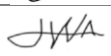
Testing Period and Place

Name	DEKRA Edificio ALEI
Test Location	C/Severo Ochoa 55, 29590, Málaga -
Date (Start)	2025-10-21
Date (Finish)	2025-11-17

Document History

Report Number	Date	Description
80676CRCS.017	2025-11-17	Emitted Evaluation Report 001

Remarks and Comments

Name	Position	Signature
Alvaro Noblejas	Evaluator	
Jorge Wallace	Project Manager	

The following table shows the tools used in this evaluation:

Name	Type	Code	Software Version	Hardware Version
xGecu TL866II Universal Programmer	Hardware	10853		
ANTSDR E200	Hardware	10556		

BinWalk	Software	09240	v3.1.0	
Ellisys Bluetooth Analyzer	Software	10898	5.0.9356	
JTAGulator	Hardware	09234	Versión software 1.11	
Synopsys Defensics	Software	09738	10.3.0	
Wireshark	Software	08455	4.6.0rc0	

Authorizations

NDA signed with Google LLC

Testing Verdicts

PASS	P
FAIL	F
NA	NA
INCONCLUSIVE	INC

EN18031-1

Access control mechanism				
	P	F	NA	INC
ACM-1 Applicability of access control mechanisms	X			
ACM-2 Appropriate access control mechanisms	X			

Authentication mechanism				
	P	F	NA	INC
AUM-1 Applicability of authentication mechanisms	X			
AUM-2 Appropriate authentication mechanisms	X			
AUM-3 Authenticator validation	X			
AUM-4 Changing authenticators			X	
AUM-5 Password strength			X	
AUM-6 Brute force protection			X	

Secure update mechanism				
	P	F	NA	INC
SUM-1 Applicability of update mechanisms	X			
SUM-2 Secure updates	X			
SUM-3 Automated updates	X			

Secure storage mechanism				
	P	F	NA	INC
SSM-1 Applicability of secure storage mechanisms	X			
SSM-2 Appropriate integrity protection for secure storage mechanisms	X			
SSM-3 Appropriate confidentiality protection for secure storage mechanisms	X			

Secure communication mechanism				
	P	F	NA	INC
SCM-1 Applicability of secure communication mechanisms	X			
SCM-2 Appropriate integrity and authenticity protection for secure communication mechanisms	X			
SCM-3 Appropriate confidentiality protection for secure communication mechanisms	X			
SCM-4 Appropriate replay protection for secure communication mechanisms	X			

Resilience mechanism				
	P	F	NA	INC
RLM-1 Applicability and appropriateness of resilience mechanisms			X	

Network monitoring mechanism				
	P	F	NA	INC
NMM-1 Applicability and appropriateness of network monitoring mechanisms			X	

Traffic control mechanism				
	P	F	NA	INC
TCM-1 Applicability of and appropriate traffic control mechanisms			X	

Confidential cryptographic keys				
	P	F	NA	INC
CCK-1 Appropriate CCKs	X			
CCK-2 CCK generation mechanisms	X			
CCK-3 Preventing static default values for preinstalled CCKs	X			

General equipment capabilities				
	P	F	NA	INC
GEC-1 Up-to-date software and hardware with no publicly known exploitable vulnerabilities	X			
GEC-2 Limit exposure of services via related network interfaces	X			
GEC-3 Configuration of optional services and the related exposed network interfaces			X	

GEC-4 Documentation of exposed network interfaces and exposed services via network interfaces	X			
GEC-5 No unnecessary external interfaces	X			
GEC-6 Input validation	X			

Cryptography				
	P	F	NA	INC
CRY-1 Best practice cryptography	X			

Appendix A: Evaluation Results

1. Categories, Security Features and Categories Summary

Security Evaluation of the ToE has been divided into different categories.

Security Analysis of each category is structured in different security features. In the same way, each security feature can be composed of several tests.

The following table shows the security features defined per each category and the number of tests of each security feature.

Category	Security Features	Nº Tests
1. EN18031-1	1.1 Access control mechanism	2
	1.2 Authentication mechanism	6
	1.3 Secure update mechanism	3
	1.4 Secure storage mechanism	3
	1.5 Secure communication mechanism	4
	1.6 Resilience mechanism	1
	1.7 Network monitoring mechanism	1
	1.8 Traffic control mechanism	1
	1.9 Confidential cryptographic keys	3
	1.10 General equipment capabilities	6
	1.11 Cryptography	1

Appendix B: Photographs

Figure 1: Nest Thermostat Unit

Appendix C : Google Nest Thermostat 4th Evaluation Results

Google Nest Thermostat 4th Test Analysis

1. Applicable Evaluation Cases

Pentesting procedures for Google Nest Thermostat 4th Evaluation Results cover 1 different areas with a total of 31 evaluation cases.

For this particular evaluation and according with the technical criteria of the evaluator the following evaluation cases has been considered applicable to this specific ToE:

Area	Evaluation Area	Identifier	Evaluation Case Title	Apply	Relevant results
EN18031-1	Access control mechanism	ACM-1	Applicability of access control mechanisms	PASS	
EN18031-1	Access control mechanism	ACM-2	Appropriate access control mechanisms	PASS	
EN18031-1	Authentication mechanism	AUM-1	Applicability of authentication mechanisms	PASS	
EN18031-1	Authentication mechanism	AUM-2	Appropriate authentication mechanisms	PASS	
EN18031-1	Authentication mechanism	AUM-3	Authenticator validation	PASS	
EN18031-1	Authentication mechanism	AUM-4	Changing authenticators	NA	
EN18031-1	Authentication mechanism	AUM-5	Password strength	NA	
EN18031-1	Authentication mechanism	AUM-6	Brute force protection	NA	
EN18031-1	Secure update mechanism	SUM-1	Applicability of update mechanisms	PASS	
EN18031-1	Secure update mechanism	SUM-2	Secure updates	PASS	
EN18031-1	Secure update mechanism	SUM-3	Automated updates	PASS	
EN18031-1	Secure storage mechanism	SSM-1	Applicability of secure storage mechanisms	PASS	
EN18031-1	Secure storage mechanism	SSM-2	Appropriate integrity protection for secure storage mechanisms	PASS	

Area	Evaluation Area	Identifier	Evaluation Case Title	Apply	Relevant results
EN18031-1	Secure storage mechanism	SSM-3	Appropriate confidentiality protection for secure storage mechanisms	PASS	
EN18031-1	Secure communication mechanism	SCM-1	Applicability of secure communication mechanisms	PASS	
EN18031-1	Secure communication mechanism	SCM-2	Appropriate integrity and authenticity protection for secure communication mechanisms	PASS	
EN18031-1	Secure communication mechanism	SCM-3	Appropriate confidentiality protection for secure communication mechanisms	PASS	
EN18031-1	Secure communication mechanism	SCM-4	Appropriate replay protection for secure communication mechanisms	PASS	
EN18031-1	Resilience mechanism	RLM-1	Applicability and appropriateness of resilience mechanisms	NA	
EN18031-1	Network monitoring mechanism	NMM-1	Applicability and appropriateness of network monitoring mechanisms	NA	
EN18031-1	Traffic control mechanism	TCM-1	Applicability of and appropriate traffic control mechanisms	NA	
EN18031-1	Confidential cryptographic keys	CCK-1	Appropriate CCKs	PASS	
EN18031-1	Confidential cryptographic keys	CCK-2	CCK generation mechanisms	PASS	
EN18031-1	Confidential cryptographic keys	CCK-3	Preventing static default values for preinstalled CCKs	PASS	
EN18031-1	General equipment capabilities	GEC-1	Up-to-date software and hardware with no publicly known exploitable vulnerabilities	PASS	
EN18031-1	General equipment capabilities	GEC-2	Limit exposure of services via related network interfaces	PASS	
EN18031-1	General equipment capabilities	GEC-3	Configuration of optional services and the related exposed network interfaces	NA	
EN18031-1	General equipment capabilities	GEC-4	Documentation of exposed network interfaces and exposed services via network interfaces	PASS	

Area	Evaluation Area	Identifier	Evaluation Case Title	Apply	Relevant results
EN18031-1	General equipment capabilities	GEC-5	No unnecessary external interfaces	PASS	
EN18031-1	General equipment capabilities	GEC-6	Input validation	PASS	
EN18031-1	Cryptography	CRY-1	Best practice cryptography	PASS	

2. Results of Evaluation Procedure

The set of evaluation procedures has been split into 1 sub-categories or areas for a total of 31 evaluation cases.

Areas

1. EN18031-1

Following is detailed the results obtained for the 1 areas in each evaluation case:

EN18031-1

Access control mechanism

ACM-1: Applicability of access control mechanisms

Results

According to **IXIT 02-Asst**, the identified access control mechanisms that manage entities' access to security/network assets are as follows:

- **Asst-MatterCert1:** Matter certificate. This asset is stored in StoreMech-Flash and this asset is managed entities' access by **AccCtrl-Flash**: The Arm Cortex cores in the i.MX RT1170 include a Memory Protection Unit (MPU).
- **Asst-MatterNetworkCredentials:** Matter network credentials. This asset is stored in StoreMech-Flash and this asset is managed entities' access by **AccCtrl-Flash**: The Arm Cortex cores in the i.MX RT1170 include a Memory Protection Unit (MPU).
- **Asst-FwKey:** Firmware signature verification key stored in OTP. This asset is stored in StoreMech-OTP and this asset is managed entities' access by **AccCtrl-OTP**: Once a value is programmed into OTP, it cannot be modified or erased, providing a strong level of protection against tampering.
- **Asst-GoogleCert2:** Google Services certificate. This asset is stored in StoreMech-OTP and this asset is managed entities' access by **AccCtrl-OTP**: Once a value is programmed into OTP, it cannot be modified or erased, providing a strong level of protection against tampering.
- **Asst-BLEKey2:** BLE link key. This asset is stored in StoreMech-Flash and this asset is managed entities' access by **AccCtrl-Flash**: The Arm Cortex cores in the i.MX RT1170 include a Memory Protection Unit (MPU).

- **Asst-DeviceModelKey1:** A hard-coded unique key per device type for OTA, ML models and other data encryption per device type. It is stored on the OTP in the TEE. This asset is stored in StoreMech-OTP and this asset is managed entities' access by **AccCtrl-OTP**: Once a value is programmed into OTP, it cannot be modified or erased, providing a strong level of protection against tampering.
- **Asst-JWT1:** Device JWT. This asset is stored in StoreMech-Flash and this asset is managed entities' access by **AccCtrl-Flash**: The Arm Cortex cores in the i.MX RT1170 include a Memory Protection Unit (MPU).
- **Asst-Logging1:** Logging and crash dumps. This asset is stored in StoreMech-Flash and this asset is managed entities' access by **AccCtrl-Flash**: The Arm Cortex cores in the i.MX RT1170 include a Memory Protection Unit (MPU).
- **Asst-DeviceCert:** The device includes a device certificate which is programmed at the factory and used to prove the authenticity of the device. This asset is stored in StoreMech-OTP and this asset is managed entities' access by **AccCtrl-OTP**: Once a value is programmed into OTP, it cannot be modified or erased, providing a strong level of protection against tampering.
- **Asst-SessionKey1:** There are several session keys which are established from the TLS connection between the device and the cloud, along with local session created on the Weave network between the device and other local devices. This asset is not persistent in the device and this asset is managed entities' access by **AccCtrl-AT**: Valid access tokens.

Furthermore, physical interfaces that provide direct access to the security/network assets are not accessible. No debug interfaces (UART/JTAG/SWD) have been identified by using the JTAGulator tool, as shown below:

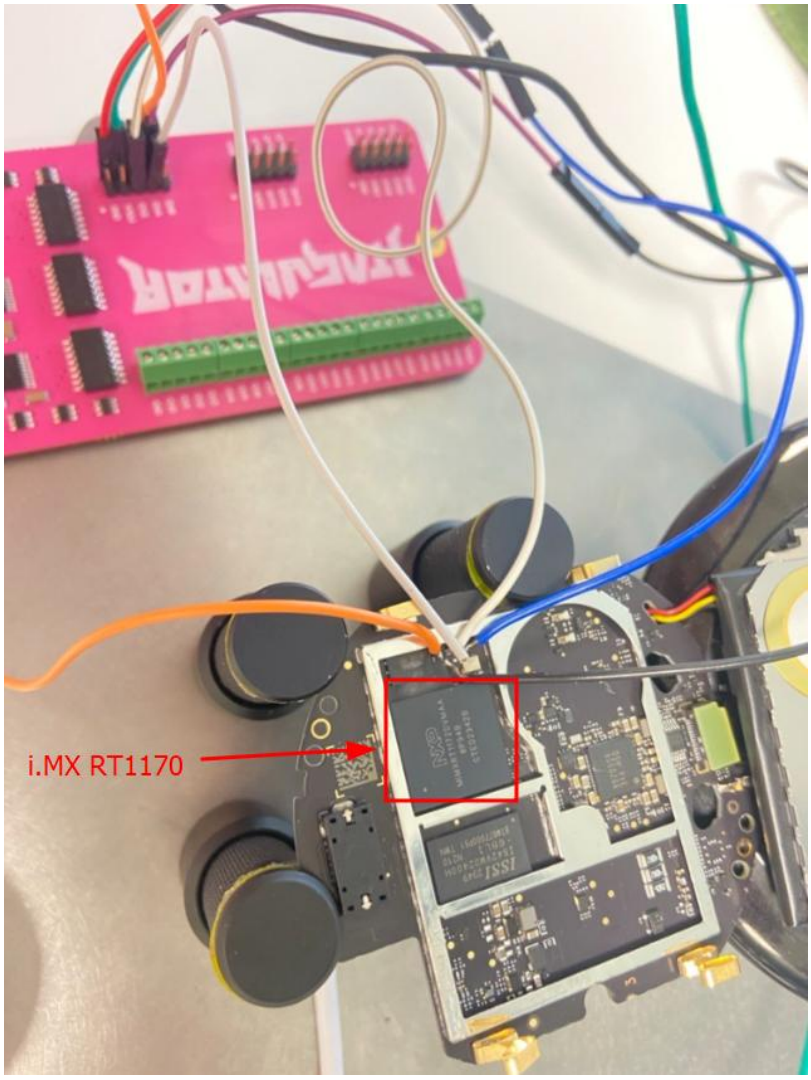


Figure 2: Set Up

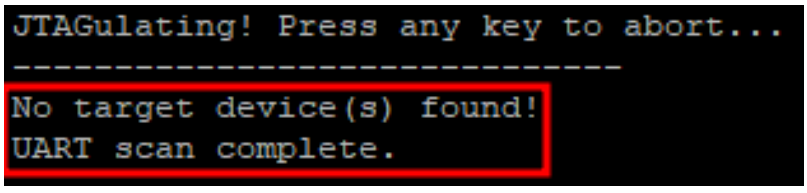


Figure 3: No UART found

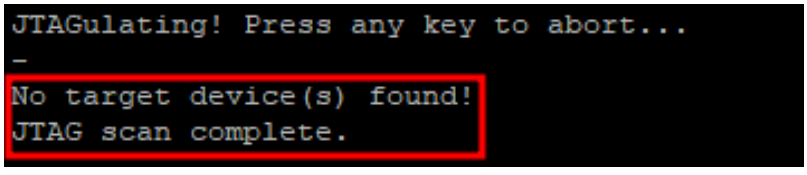


Figure 4: No JTAG found

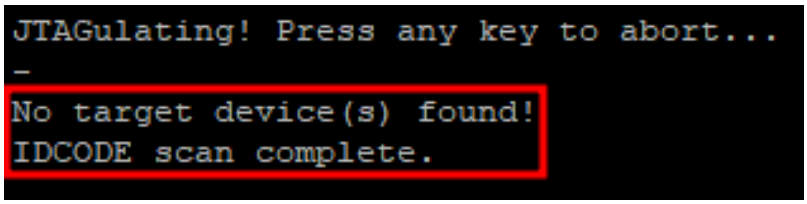


Figure 5: No SWD found

Related to i.MX RT1170 MCU, there are three JTAG security modes available: No Debug mode, Secure JTAG mode and JTAG Enabled mode. The TL can assume that the JTAG is in No Debug mode, all security-sensitive JTAG features are permanently blocked, preventing any debug.

RESULT:	PASS

ACM-2: Appropriate access control mechanisms

Results

As seen in the **Evaluation Case ACM-1**, there are some security/network assets that are protected by different access control mechanism in order to ensure these assets are protected from unauthorized access:

- **AccCtrl-AT**: Devices and users must present valid access tokens to make changes to security settings. Tokens grant the bearer access based on the principle of least privilege, ensuring they have only the access necessary to perform their current tasks or roles.

  Tokens grant the bearer access based on the principle of least privilege, ensuring they have only the access necessary to perform their current tasks or roles. To capture this tokens *Burp suite* and *Rooted phone* were used:

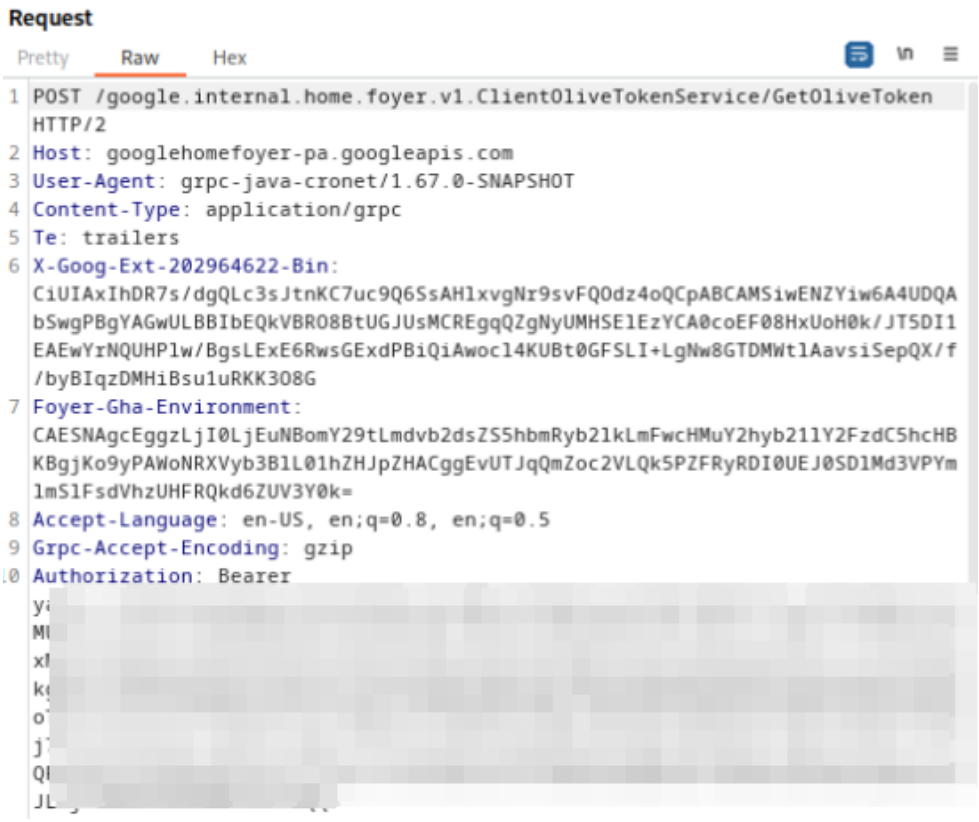


Figure 6: Bearer Token used.

- **AccCtrl-Flash**: The Arm Cortex cores in the i.MX RT1170 include a Memory Protection Unit (MPU). The MPU can be configured by the firmware to define access permissions (read, write, execute) for different memory regions, including the external flash. This helps isolate processes and prevent unauthorized access to sensitive areas of the flash. User-facing UIs or network protocols do not provide raw read/write access to the flash. Direct access to the flash contents via standard interfaces is heavily restricted.

  According to the datasheet of the MX RT1170 processor system, the block diagram shows that there is a Integrated Memory Protection Unit (MPU):

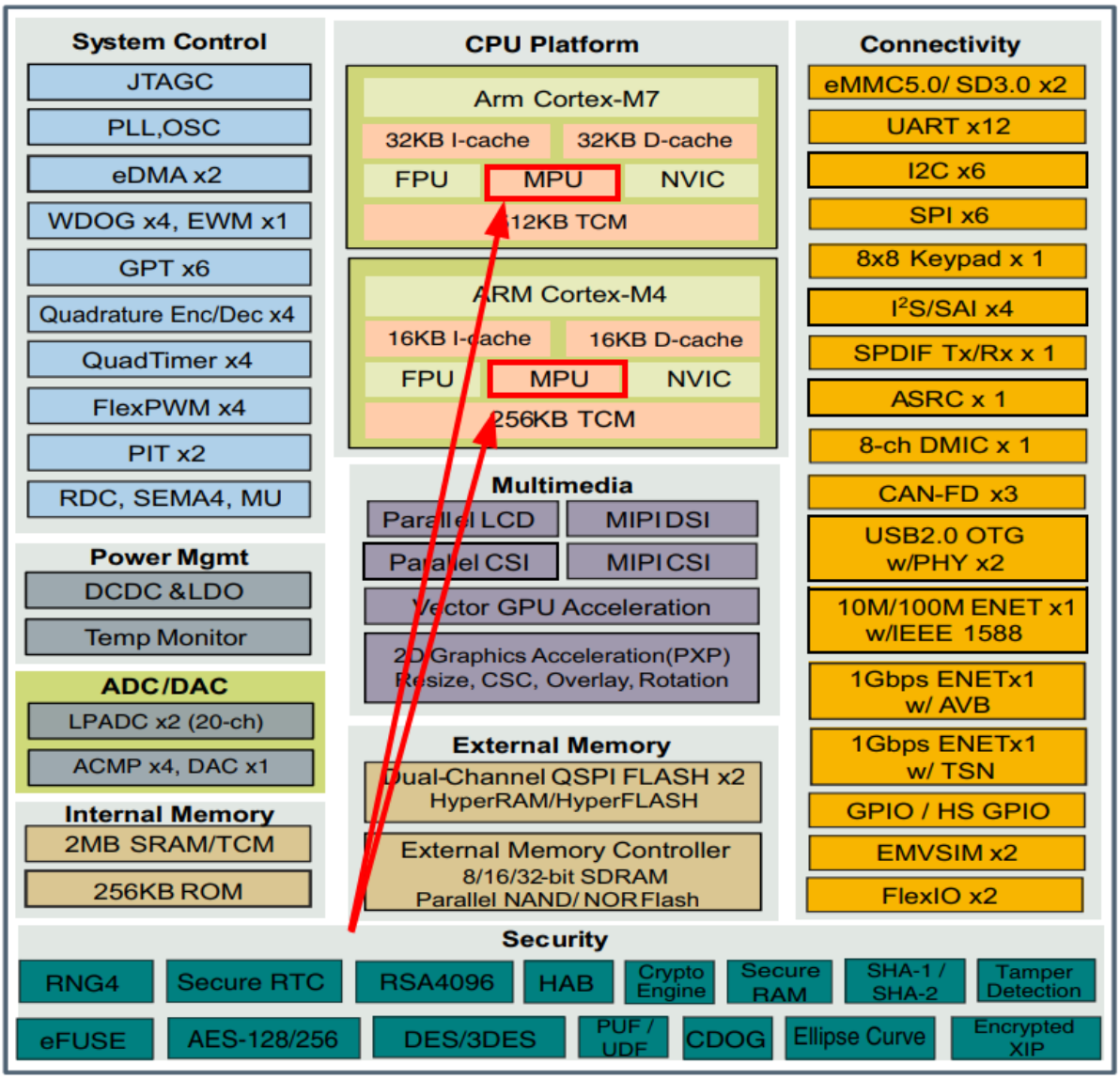


Figure 7: i.MX RT1170 system block diagram

Furthermore, as seen in **Evaluation Case ACM-1,** no debug interface provided access to the Flash memory.

- **AccCtrl-OTP**: OTP memory is embedded within the device's hardware. Once a value is programmed into OTP, it cannot be modified or erased, providing a strong level of protection against tampering. Programming the device key into OTP is restricted an.d controlled during a secure provisioning process. (write-once, read-many nature and physical inaccessibility). Device key remains confidential and protected from unauthorized access or modification

RESULT:	PASS

Authentication mechanism

AUM-1: Applicability of authentication mechanisms

Results

As can be seen on **IXIT 02-Asset** and according to the Information present on**IXIT 07-AuthMech** there are some authentication mechanism that apply to some security/network assets.

- **AuthMech-GHA**: Google Home App (GHA, iOS or Android) uses Google user account service to authenticate a user. After user is authenticated, GHA uses OAuth token to authenticate the user and associate the DUT.

  This authentication mechanism is employed to manage the access to the following assets: *Asst-MatterCert1, Asst-GoogleCert2, Asst-JWT1, Asst-SessionKey1*.

- **AuthMech-Cloud:** Device communicates with multiple end points in the cloud for configuration, upgrade, etc. The DUT authenticates a service by Service Certificates over HTTPS/TLS and also the DUT authenticates itself by device certificate over HTTPS/TLS. Some endpoints only require server authentication if the service does not need device identity or additional security is used at device side.

  This authentication mechanism is employed to manage the access to the following assets: *Asst-MatterCert1, Asst-GoogleCert2, Asst-JWT1, Asst-Logging1, Asst-SessionKey1.*

- **AuthMech-Matter:** Matter is an IPv6-based technology for Internetof things (IoT) products.

  This authentication mechanism is employed to manage the access to the following Assets: *Asst-MatterNetworkCredentials.*

- **AuthMech-GAIA:** GAIA is Google's user authentication system. It assigns a unique ID to each user and verifies their identity during login. Google services then use this ID to authorize access to resources based on user roles or permissions. Gaia is not directly accessible over a network but is used by Google services through APIs. From a user perspective, the GAIA appears as a user name and password.

  This authentication mechanism does not manage access to any assets, so is **not applicable**.

RESULT:	PASS

AUM-2: Appropriate authentication mechanisms

Results

As were seen on **Evaluation Case AUM-1** and according to the information provided on **IXIT 07-AuthMech** the following authentication mechanism are applicable:

- **AuthMech-GHA**: Google Home App (GHA, iOS or Android) uses Google user account service to authenticate a user. GHA uses Google account service to authenticate a user for login. User can use user and password or MFA (Multi Factor Authentication) supported by Google account security. As can be seen below this authentication is performed through services that are not provided by the DUT as there are provided by Google Accout Services, so is **not applicable**:

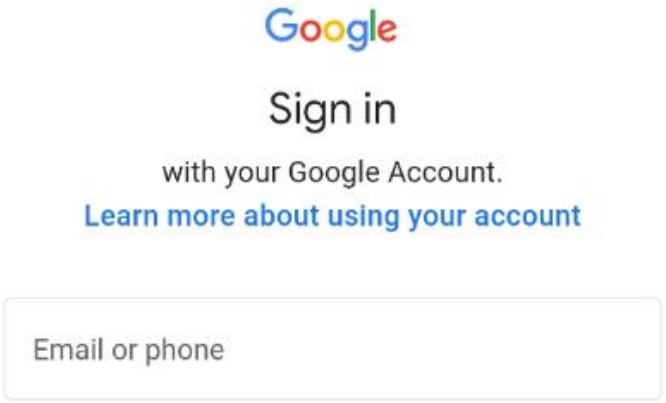


Figure 8: Authentication of user on Google Home Nest App

However, when the user afterwards wants to authenticate the DUT with their Google account, the Google Home app sends a token to the DUT as can be seen on **Evaluation Case SCM-2**, and it is the DUT that verifies this token against the Google Cloud Services. The communication mechanism used for this connection with the cloud make use of TLS v1.2 with device and server authentication through certificates as can be seen below on AuthMech-Cloud.

- **AuthMech-Matter:** Credentials (Pin or Scan Code) are generated when the Matter network is created or when the device joines a Matter network. These credentials are only valid at the moment they are generated by the user and remain valid for a short time window, after which they expire.

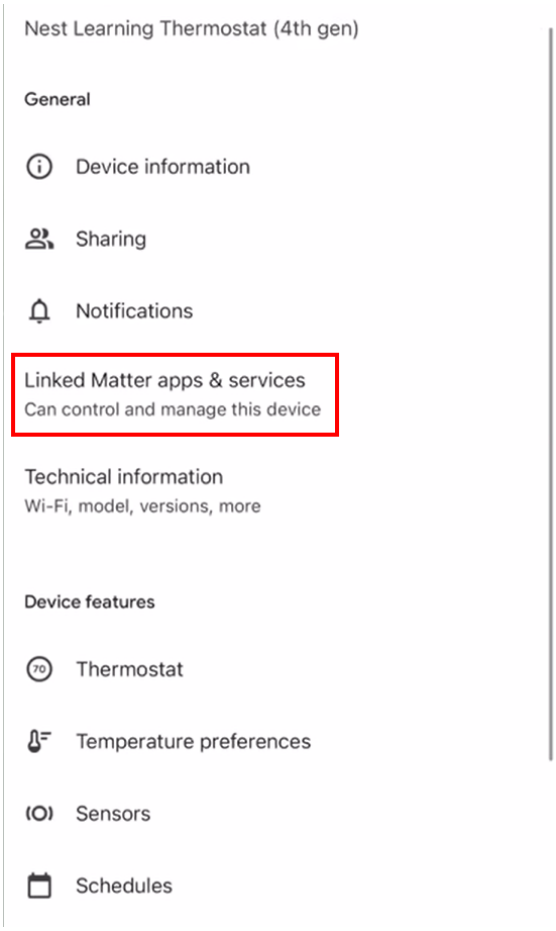


Figure 9: Manage matters in the GHA

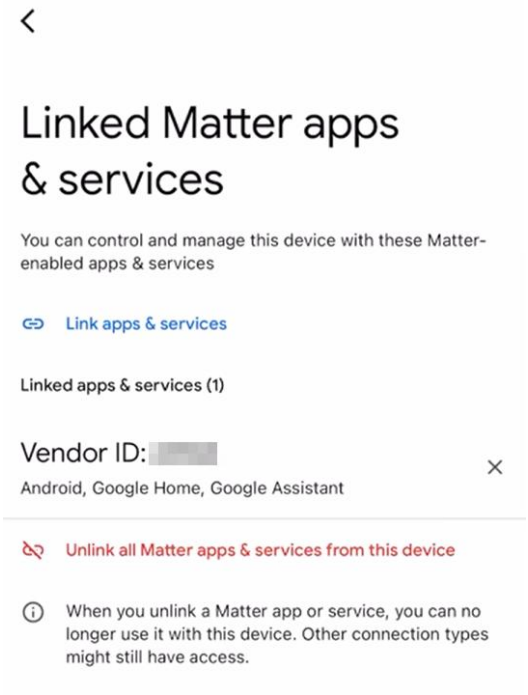


Figure 10: Linked Matter apps & services

Using GHA, a pairing code can be used to link it on a Matter-enable app.

Figure 11: Use pairing code to link

Using the DUT, a pairing code or a Scan Code cab be used:

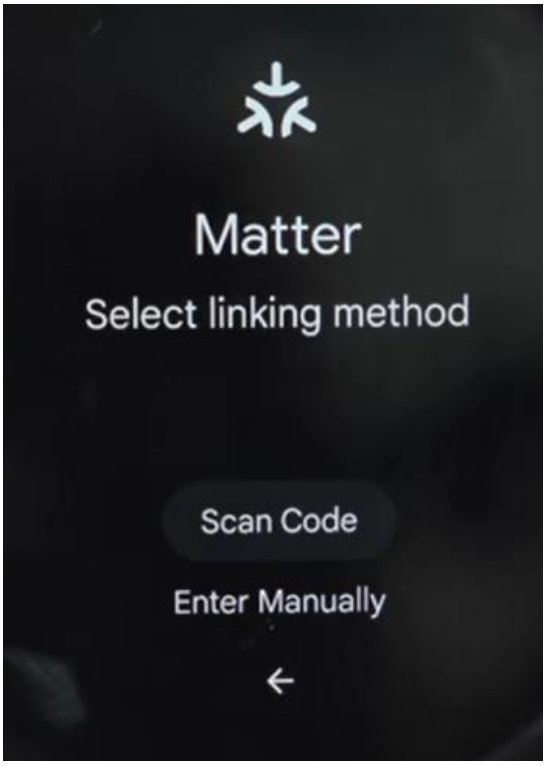


Figure 12: Linking method available in the DUT

RESULT:	PASS

AUM-3: Authenticator validation

Results

Each authentication mechanism required per **Evaluation Case AUM-1** shall validate all relevant properties of the used authenticators, dependent on the available information in the operational environment of use:

- **AuthMech-GHA**: As were seen on Evaluation Case AUM -2 Google Home App (GHA, iOS or Android) uses Google account service to authenticate a user for login. User can use user and password or MFA (Multi Factor Authentication) supported by Google account security to authenticate on the Google Home App. As were seen the DUT Authentication works using a Cloud service that verifies that a token sent by a GHA is legitimate and active, and will rejects and/or revoke expired or compromised tokens. As this verification is not performed in the DUT, this Authentication Mechanism is considered **Not Applicable** to this evaluation case.

- **AuthMech-Cloud:** Device communicates with multiple end points in the cloud for configuration, upgrade, etc. Device authenticates a service by service certificates over HTTPS/TLS. Also device authenticates itself by device certificate over HTTPS/TLS. Some endpoints only require server authentication if the service does not need device identity or additional security is used at device side.

As seen in **Evaluation Case AUM-2**, this authentication mechanism make use of TLS with device or server certificates to perform authentication.

- **AuthMech-Matter:** Credentials (Pin or Scan Code) are generated when the Matter network is created or when the device joines a Matter network. These credentials are only valid at the moment they are generated by the user and remain valid for a short time window, after which they expire.

Figure 13: Incorrect credentials cause a pairing error.

RESULT:	PASS

AUM-4: Changing authenticators

Results

According to **IXIT 07-AuthMech**, the following authenticators can be changed by an authorized entity:

- **AuthMech-GHA**: User can change Google account password, as 2-step verification and authenticator in the Google Account center and FDR to get a new device certificate. As the change of this authenticator is not performed on the DUT the test laboratory determine that this authentication mechanism is **Not Applicable** to this evaluation case.

However, as can be seen below the test laboratory have reviewed the process of changing the authenticator on the app, in order to check the 2-step verification and authenticator is enabled and working properly:

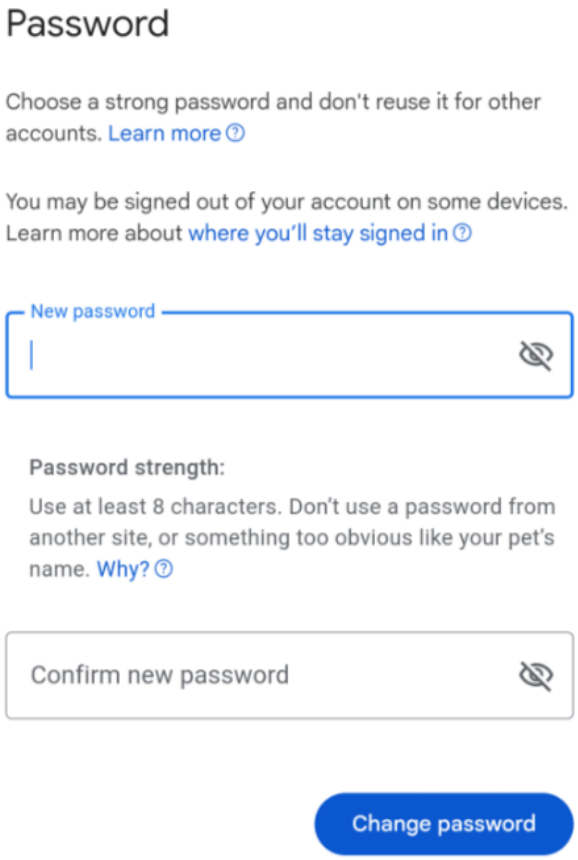


Figure 14: Change Google account password

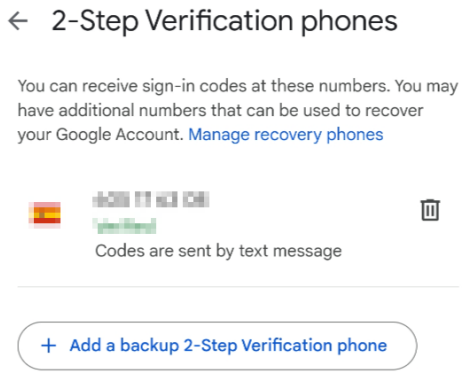


Figure 15: Change 2-step verification

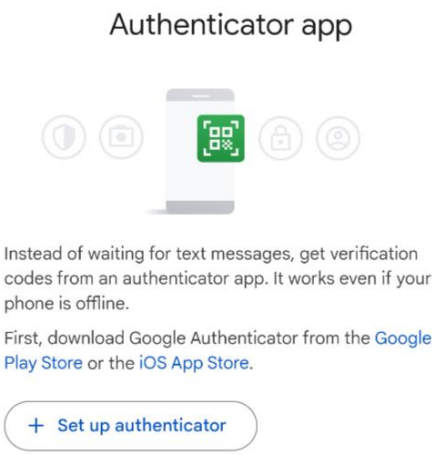


Figure 16: Change authenticator

- **AuthMech-Cloud:** The certificates used to Authenticate the DUT against the Cloud services shall be permanent. Making it possible to modify them could conflict security goals, since it would create a new attack vector that would not exist otherwise. The Test Laboratory determine that this authentication mechanism is **Not Applicable** to this evaluation case.

- **AuthMech-Matter**: Factory Device Reset to get a new device certificates, DID.

Figure 17: Factory reset

RESULT:	NA

AUM-5: Password strength

Results

According to**IXIT 07-AuthMech**, the only authentication mechanism using passwords is **AuthMech-GHA**, which relies on Google Account Services.

- Password management (strength, uniqueness, enforcement) is handled by Google in the **Google Account Center**, outside the DUT.
- The DUT does not store, process, or validate user passwords, relying instead on OAuth 2.0 authentication tokens.
- Google enforces strong password policies and multi-factor authentication (MFA) per NIST SP 800-63B, as seen in **Evaluation Case AUM-4**.

As shown in the image below, setting up the device is done through the Google Home App. This process requires signing in with a Google Account, which manages authentication separately from the DUT itself. This information can be found in the Set Up Manual https://support.google.com/googlenest/answer/7029485?hl=en&co=GENIE.Platform%3DAndroid.

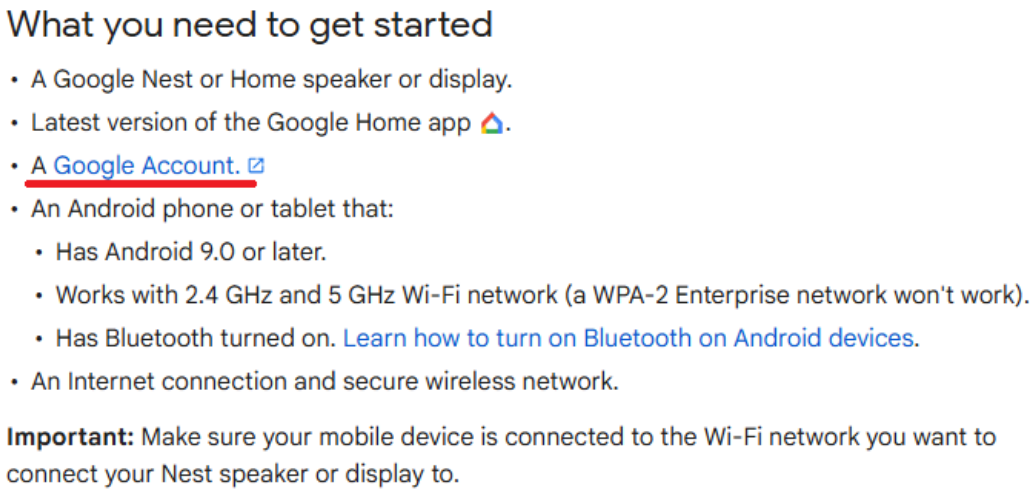


Figure 18: Setting up a Google Nest family device via the Google Home App

Since password security is externalized to Google Account Services, **AUM-5 is Not Applicable** to the DUT. Additionally, **AuthMech-Cloud** and **AuthMech-Matter** does not use passwords, making this evaluation case **Not Applicable**.

RESULT:	NA

AUM-6: Brute force protection

Results

According to **IXIT 07-AuthMech**, **AuthMech-GHA** and **AuthMech-Cloud** authenticate users via Google Account Services.

- Password verification and brute force protection are handled by Google in the **Google Account Center**, not the DUT, as seen in **Evaluation Case AUM-5.**
- Google applies login attempt throttling, account lockout, and MFA, as seen in **Evaluation Case AUM-4**, to mitigate brute force attacks.
- The DUT does not process authentication credentials.

Since brute force mitigation is implemented at the Google Service level, these Authentication mechanisms are **Not Applicable**.

Furthermore, **AuthMech-Matter** has a cryptographic key establishment protocol called SPAKE2+ for secure pairing of devices. This protocol is resistant to brute-force attacks because it involves a complex exchange of cryptographic information that makes it computationally difficult for an attacker to guess the correct keys.

RESULT:	NA

Secure update mechanism

SUM-1: Applicability of update mechanisms

Results

According to **IXIT 10-SoftComp**, the identified software components are as follows:

- **SoftComp-SBL:** Secondary boot loader and system softwares (u-boot, kernel, driver). These component is upgradable via **UpdMech-1**.
- **SoftComp-FW:** System and application firmware, this component is upgradable via **UpdMech-1**.

According to the information provided on**IXIT 09- UpdMech** the device is upgradable tvia the following update mechanism:

- **UpdMech-1:** The device performs updates without the need of user interaction. During the initialization process, the device will perform any needed update prior to entering the operation state. Once operational, the device maintains communications with the Google cloud service which will automatically push an update when needed. The updates are signed by a Google certificate and verified by the device prior to applying the image.

RESULT:	PASS

SUM-2: Secure updates

Results

Each update mechanism required per **Evaluation Case SUM-1** shall only install software whose integrity and authenticity are valid at the time of the installation:

- **UpdMech-1:** The device requests and downloads updates over an HTTP or HTTPS connection (using the implementations and algorithms provided by the Chrome Cryptographic libraries, listed below).

In addition, the update package is signed using ECDSA with SHA-256. The device includes within its firmware an embedded ECDSA public key, as evidenced by the presence of the symbol *SrpEcdsaKey* in the analyzed image. During the update process, the downloaded manifest and its contents are verified by computing their SHA-256 hash and validating the ECDSA signature against this hash using the stored public key.

```
/media/sf_Compartida_Kali_VM/Thermostat readelf -S Zirconium_Builds_2.1-25_Production_b1_release_Nest-Bismuth-2.1-25-b1-app_no_bl-en_US-US.elf
There are 11 section headers, starting at offset 0x26d934:

Section Headers:
  [Nr] Name              Type            Addr     Off    Size   ES Flg Lk Inf Al
  [ 0]                   NULL            00000000 000000 000000 00      0   0  0
  [ 1]   AUPD            PROGBITS        202e0000 000034 0134c8 00  WA  0   0  1
  [ 2] __SIG__           PROGBITS        00000000 0134fc 000229 00      0   0  1
  [ 3] __MFEST__         PROGBITS        00000000 013725 000305 00      0   0  1
  [ 4] __BCM_WIFI_FW__   PROGBITS        00000000 013a2a 08482e 00      0   0  1
  [ 5] __BCM_WIFI_CLM__  PROGBITS        00000000 098258 0004a4 00      0   0  1
  [ 6] __THREAD_FW__     PROGBITS        00000000 0986fc 0423ac 00      0   0  1
  [ 7] __BCM_WIFI_N[...] PROGBITS        00000000 0daaa8 0007e4 00      0   0  1
  [ 8] __APP__           PROGBITS        00000000 0db28c 18134b 00      0   0  1
  [ 9] __BCM_BLE__       PROGBITS        00000000 25c5d7 0112e0 00      0   0  1
  [10] .shstrtab         STRTAB          00000000 26d8b7 00007c 00      0   0  1
Key to Flags:
  W (write), A (alloc), X (execute), M (merge), S (strings), I (info),
  L (link order), O (extra OS processing required), G (group), T (TLS),
  C (compressed), x (unknown), o (OS specific), E (exclude),
  p (processor specific)
```

Figure 19: Signature available in the update package

```
strings Zirconium_Builds_2.1-25_Production_b1_release_Nest-Bismuth-2.1-25-b1-app_no
_bl-en_US-US.elf | grep -i ecdsa

SrpEcdsaKey
```

Figure 20: Embedded ECDSA key

```
binwalk Zirconium_Builds_2.1-25_Production_b1_release_Nest-Bismuth-2.1-25-b1-app_no
_bl-en_US-US.elf          127|1 err

DECIMAL       HEXADECIMAL     DESCRIPTION
--------------------------------------------------------------------------------
0             0x0             ELF, 32-bit LSB relocatable, no machine, (ARM)
72506         0x11B3A         XML document, version: "1.0"
88846         0x15B0E         ESP Image segment count: 1, flash mode: QUOUT, flash speed: 40MHz, flash size: 1MB, entry addre
ss: 0xf0a1, hash: none
814101        0xC6C15         JBOOT STAG header, image id: 0, timestamp 0xFFF103B1, image size: 199885627 bytes, image JBOOT
checksum: 0x103, header JBOOT checksum: 0xF834
871946        0xD4E0A         Neighborly text, "neighbor_table.cpp.cpp"
873254        0xD5326         Neighborly text, "Neighbornd"
874192        0xD56D0         AES S-Box
878584        0xD67F8         AES Inverse S-Box
882968        0xD7918         SHA256 hash constants, little endian
2487450       0x25F49A        CRC32 polynomial table, little endian
```

Figure 21: SHA-256 hash

RESULT:	PASS

SUM-3: Automated updates

Results

According to **IXIT 09-UpdMech**, the device is updatable via:

- **UpdMech-1**: The device performs updates without the need of user interaction. During the initialization process, the device will perform any needed update prior to entering the operation state. Once operational, the device maintains communications with the Google cloud service which will automatically push an update when needed. The updates are signed by a Google certificate and verified by the device prior to applying the image. Updates are automatic and do not require user interaction. The device will check for new available updates about every 45 minutes. The user can not disable this.

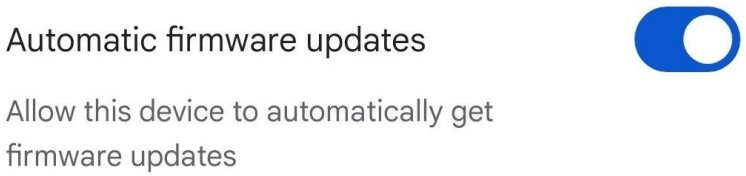


Figure 22: Notification update

RESULT:	PASS

Secure storage mechanism

SSM-1: Applicability of secure storage mechanisms

Results

According to **IXIT 02-Asst**, the following security/network assets are persistently stored on the equipment:

- The assets *Asst-MatterCert1, Asst-MatterNetworkCredentials, Asst-BLEKey2, Asst-JWT1, Asst-Logging1* are persistently stored on **StorMech-Flash**.
- The assets *Asst-FwKey, Asst-GoogleCert2, Asst-DeviceModelKey1,Asst-DeviceCert* are persistently stored on **StorMech-OTP**.
- The assets *Asst-SessionKey1* are not persistently stored, so is **not applicable**.

As seen in **Evaluation Case ACM-1**, physical interfaces that provide direct access to the storage mechanisms are not accessible.

Therefore, the storage mechanisms applicability has been verified and conforms with the requirement.

RESULT:	PASS

SSM-2: Appropriate integrity protection for secure storage mechanisms

Results

According to **IXIT 06-StorMech**, the security/network assets are protected by storing them in the following storage mechanisms:

- **StoreMech-Flash**: Flash Memory. There are 2 Flash (NOR) memories:

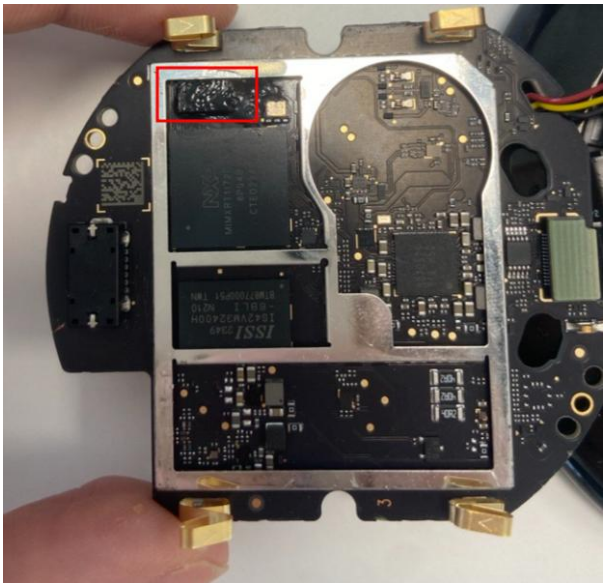


Figure 23: W25Q64JWZPIM

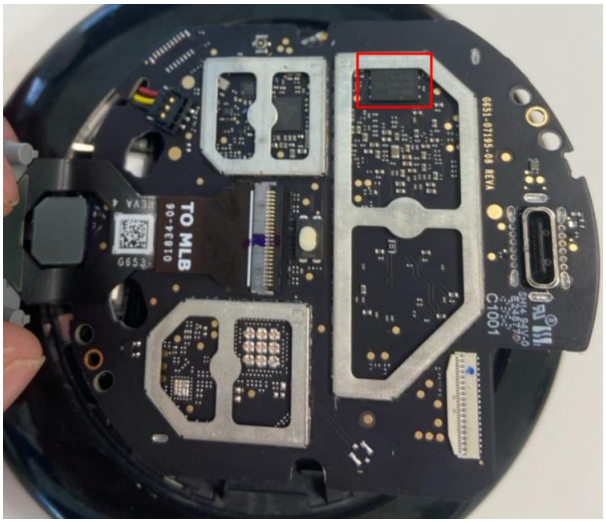


Figure 24: GD25LQ64EWIGR

Both storage mechanisms implement physical protection measures. In addition, the Flash-based storage is confidentiality protected through AES-128-CTR on-the-fly decryption, and the firmware/RTOS stored in Flash is cryptographically signed.

Furthermore, as seen in **Evaluation Case ACM-1**, the SWD/JTAG of the i.MX RT1170 MCU is locke, AHB access to the FlexSPI memory space is also inaccessible, making it impossible to read the contents of the external NOR via the MCU bus.

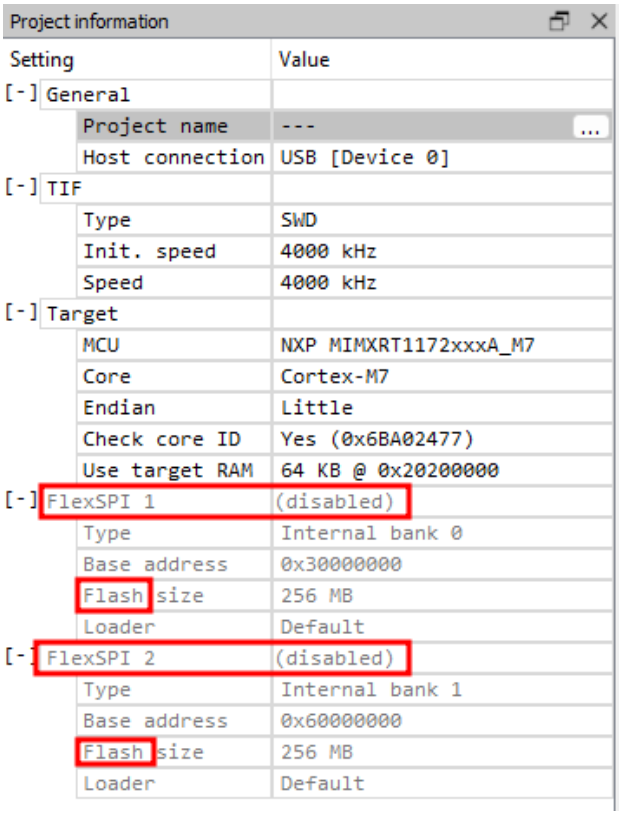


Figure 25: Flash inaccesible

- **StoreMech-OTP**: OTP memory is a special type of non-volatile memory (NVM) that permits data to be written to memory only once. Once the memory has been programmed, it retains its value upon loss of power (i.e., is non-volatile). OTP memory is used in applications where reliable and repeatable reading of data is required. Protected by system privilege and OTP read / write protection mechanism.

RESULT:	PASS

SSM-3: Appropriate confidentiality protection for secure storage mechanisms

Results

According to **IXIT 06-StorMech**, the security/network assets are protected by storing them in the following storage mechanisms:

- **StoreMech-Flash**: Flash Memory. There are 2 Flash (NOR) memories:

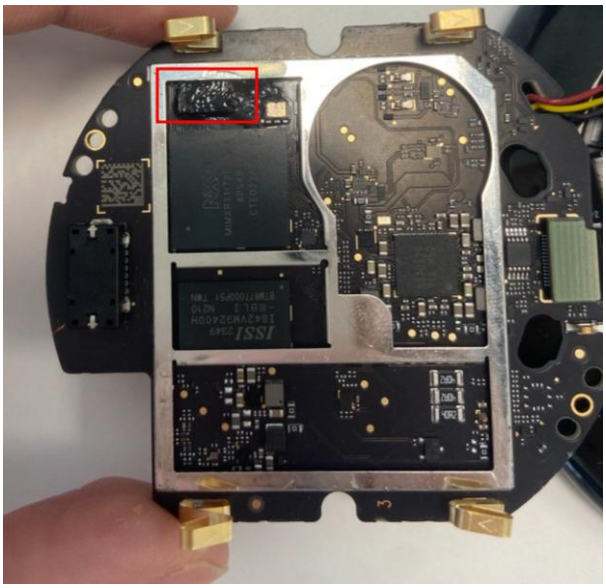


Figure 26: W25Q64JWZPIM

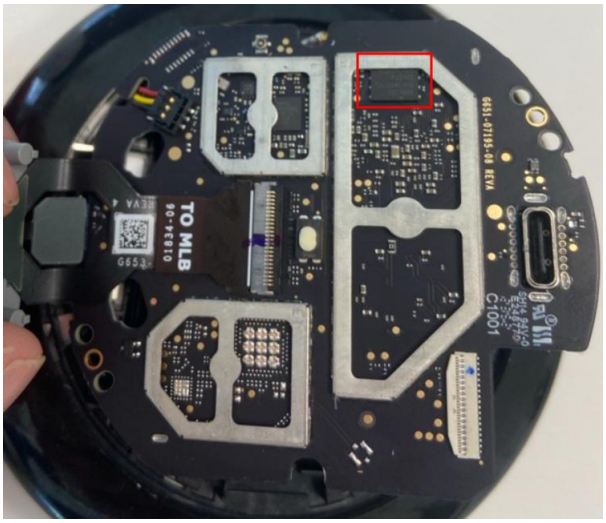


Figure 27: GD25LQ64EWIGR

Both storage mechanisms implement physical protection measures. In addition, the Flash-based storage is confidentiality protected through AES-128-CTR on-the-fly decryption, and the firmware/RTOS stored in Flash is cryptographically signed.

Furthermore, as seen in **Evaluation Case ACM-1**, the SWD/JTAG of the i.MX RT1170 MCU is locke, AHB access to the FlexSPI memory space is also inaccessible, making it impossible to read the contents of the external NOR via the MCU bus.

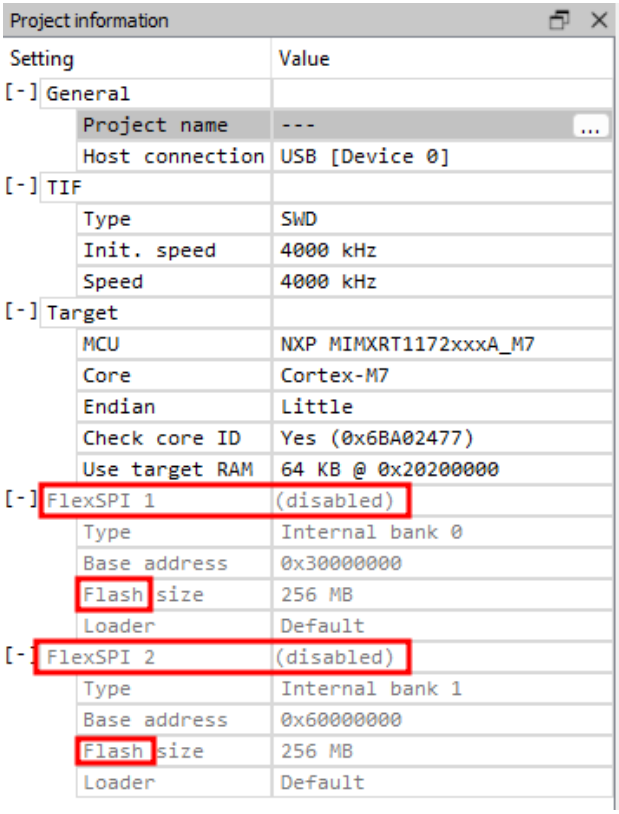


Figure 28: Flash inaccesible

- **StoreMech-OTP**: OTP memory is a special type of non-volatile memory (NVM) that permits data to be written to memory only once. Once the memory has been programmed, it retains its value upon loss of power (i.e., is non-volatile). OTP memory is used in applications where reliable and repeatable reading of data is required. Protected by system privilege and OTP read / write protection mechanism.

RESULT:	PASS

Secure communication mechanism

SCM-1: Applicability of secure communication mechanisms

Results

According to **IXIT 02-Asst**, the following security/network assets communicate with other entities via network interfaces:

- The assets *Asst-SessionKey, Asst-BLEKey, Asst-GoogleCert, Asst-JWT* are transmitted securely via **ComMech-1-WE.**
- The assets *Asst-DeviceModelKey, AsstMatterCert, Asst-MatterNetworkCredentials* are transmitted securely via **ComMech-2-WE.**
- The asset *Asst-Logging* are transmitted securely via **CommMech-4-WE.**

- The asset *Asst-GoogleCert, Asst-DeviceModelKey, Asst-JWT, Asst-Logging, Asst-SessionKey* are transmitted securely via **CommMech-BLE.**

Therefore, the communication mechanisms applicability has been verified and conforms with the requirement.

RESULT:	PASS

SCM-2: Appropriate integrity and authenticity protection for secure communication mechanisms

Results

Each secure communication mechanism required per **Evaluation Case SCM-1** shall apply best practices to protect the integrity and authenticity of the security/network assets communicated:

- **ComMech-1-WE** and **ComMech-BLE**: On this DUT this communication mechanism make use of Bluetooth Low Energy. During the initial setup process some commands from the mobile app are tranfered to the DUT trought Buetooh Low Energy in order to attach the device to the user account. To analyze that, the test laboratory have used the Ellisys Bluetooth Analyzer in order to analyze this bluetooth communication during this initial setup process. As can be seen below no Encryption at transport layer is employed:

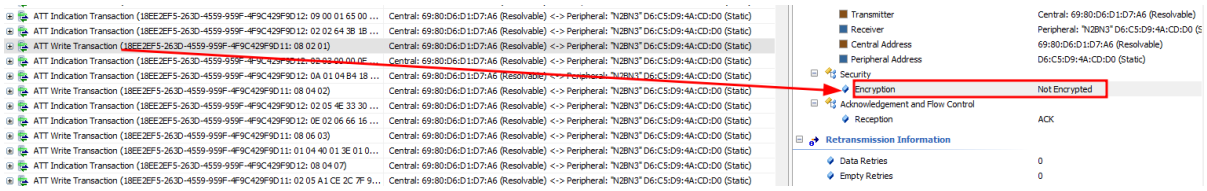


Figure 29: Bluetooth Communication not encrypted at transport layer level

However, after retrieving all the data transferred to the DUT and analysing it with binwalk, it appears that the information is encrypted at the application level as the entropy analysis is near to 1, as shown below:

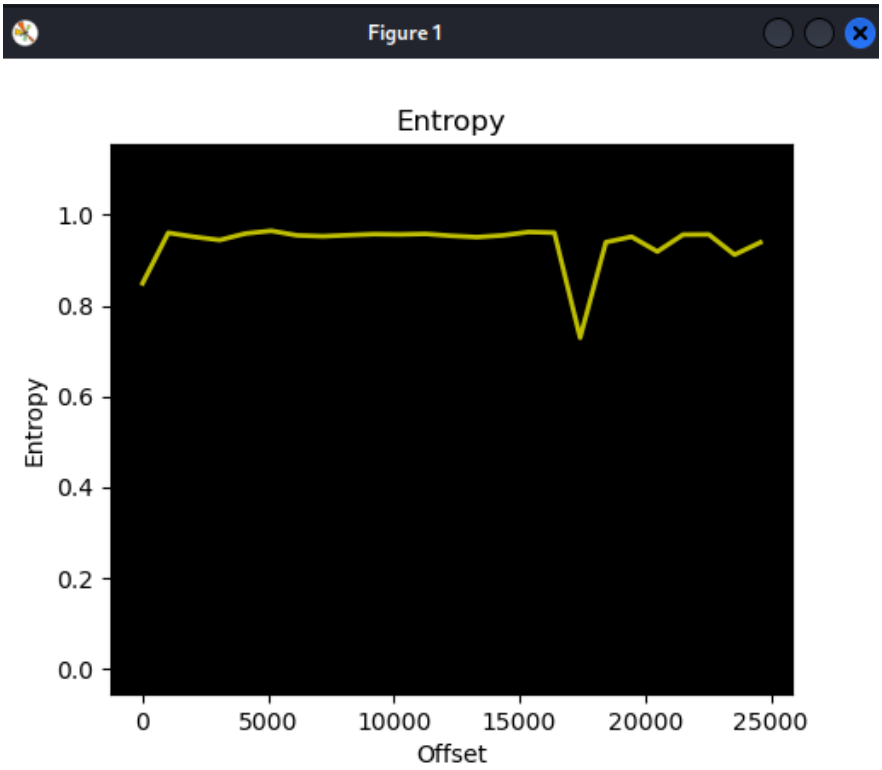


Figure 30: Binwalk analisys of transfered data

After transfering this information trought Bluetooth Low Energy then the DUT complete the addition to the user account connecting with the cloud by wifi. Traffic between the DUT and the cloud is secured with TLS v1.2 as were seen on **Evaluation Case AUM-2**.

  - Integrity: In this case, **TLSv1.2** ensures that data has not been altered during transmission through the use of a keyed Message Authentication Code (**MAC**). Each message includes an authentication code generated with a secret key shared between the sender and the receiver. In this way, if the message is modified in transit, the MAC will not match, allowing for the detection of alterations. **BLE** on the other hand uses HMAC-SHA256, **HMAC** (Hash-Based Message Authentication Code) ensures that data is not modified during transmission. The shared key established through the Diffie-Hellman (**DH**) exchange ensures that only authorised parties can validate the data.
  - Authenticity: In this case, **TLSv1.2** guarantees that the parties involved in the communication are who they claim to be. This is done through the use of digital **certificates**, which ensure that only authorized devices or services can establish secure connections with the device (DUT). These certificates are backed by a trusted Certification Authority (CA). On the other hand for **BLE, DH** exchange with pre-configured keys ensures that authenticated devices are the only authorised participants.
- **ComMech-2-WE**: The DUT communicates with different end points for device configuration, device monitoring, metrics and debug info upload (with explicit user consent), OTA. Here the traffic between the DUT and the cloud is secured with TLS v1.2.

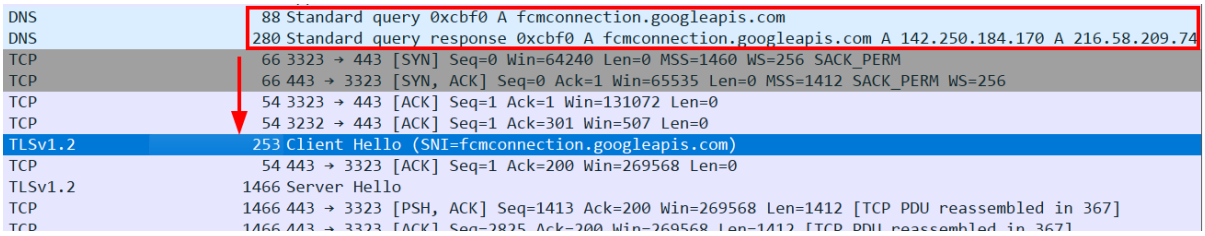


Figure 31: Communication encrypted with TLSv1.2

- Integrity: As in the previous case, **TLSv1.2** ensures that data has not been altered during transmission through the use of a keyed Message Authentication Code (**MAC**). Each message includes an authentication code generated with a secret key shared between the sender and the receiver. In this way, if the message is modified in transit, the MAC will not match, allowing for the detection of alterations.
- Authenticity: As in the previous case, **TLSv1.2** guarantees that the parties involved in the communication are who they claim to be. This is done through the use of digital **certificates**, which ensure that only authorized devices or services can establish secure connections with the device (DUT). These certificates are backed by a trusted Certification Authority (CA).

- **ComMech-4-WE:** The DUT communicates with different endpoints for device configuration and management functions, including crash upload, metrics clientsN.google.com (N is a number e.g. 1, 2, 3) tools.google.com. Here the traffic between the DUT and the cloud is secured with TLS v1.2.

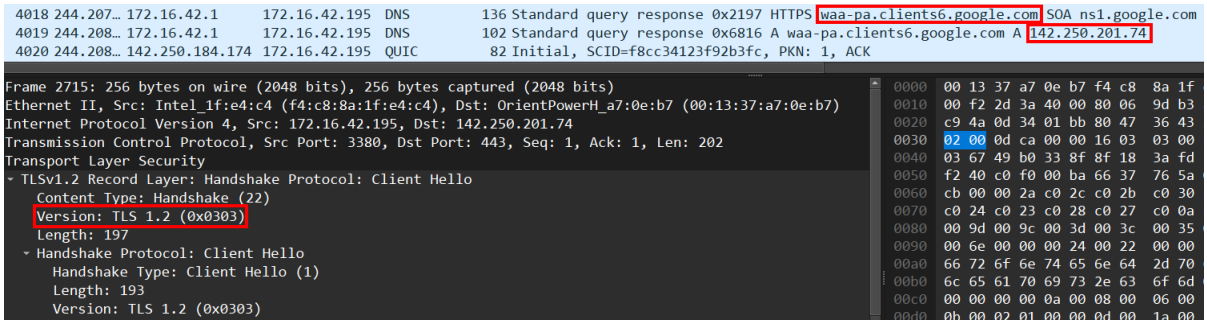


Figure 32: Communication with clients6.google.com encrypted with TLSv1.2

- Integrity: As in the previous case, **TLSv1.2** ensures that data has not been altered during transmission through the use of a keyed Message Authentication Code (**MAC**). Each message includes an authentication code generated with a secret key shared between the sender and the receiver. In this way, if the message is modified in transit, the MAC will not match, allowing for the detection of alterations.
- Authenticity: As in the previous case, **TLSv1.2** guarantees that the parties involved in the communication are who they claim to be. This is done through the use of digital **certificates**, which ensure that only authorized devices or services can establish secure connections with the device (DUT). These certificates are backed by a trusted Certification Authority (CA).

RESULT:	PASS

SCM-3: Appropriate confidentiality protection for secure communication mechanisms

Results

Each secure communication mechanism required per **Evaluation Case SCM-1** shall apply best practices to protect the confidentiality of communicated security/network assets where confidentiality protection of those is needed:

- **ComMech-1-W** and **ComMech-2-W**: Mechanisms implements the well-defined **TLSv1.2** protocol with the TLS cipher suites shown in **Evaluation Case SCM-2**.
    - Confidentiality: As all ciphersuites that are compatible are recommended by SOGIS and are not deprecated this is considered as a correct best practiques.

- **ComMech-1-W**and **ComMech-BLE**: On this DUT this communication mechanism make use of Bluetooth Low Energy. This communication is encrypted at application level as were seen on **Evaluation Case SCM-2** and it is only exposed to perform the initial set up it is considered as a correct best practices.
    - Confidentiality: As seen also in the **Evaluation Case SCM-1**, the information transmitted is encrypted at application level. The use of AES (Advanced Encryption Standard) ensures that the transmitted data is encrypted and cannot be read by an attacker who intercepts the communication. DH, on the other hand, allows secure key exchange without transmitting the key directly.

RESULT:	PASS

SCM-4: Appropriate replay protection for secure communication mechanisms

Results

Each secure communication mechanism required per **Evaluation Case SCM-1** shall apply best practices to protect the security assets and the network assets communicated against replay attacks:

- **ComMech-1-W** and **ComMech-2-W:** implements the well-defined**TLSv1.2** protocol. Because TLS is a session-based protocol with unique session keys, and because each session has unique keys, attackers cannot reuse a captured message from one session in another session, meaning that is not vulnerable to replay attacks.

- **ComMech-1-W** and **ComMech-BLE**: After pairing is authenticated by numeric comparison, an STK and BLE traffic are established. AES in BLE natively incorporates nonces.

RESULT:	PASS

Resilience mechanism

RLM-1: Applicability and appropriateness of resilience mechanisms

Results

According to **IXIT 4-Intf**, there are 3 networks interfaces in the DUT:

- **Intf-BLE:** BLE is used during the setup process and used to connect to other devices. As the intended use of this interface is on a local network that does not interoperate with others, this interface is considered **not applicable**.
- **Intf-WiFi**: The DUT needs it to connect to the Google cloud services. It also provides the functionality of the DUT for the user. Regarding to this interface, other devices in the network provide sufficient protection against DoS attacks and loss of essential functions for network operations such as WiFi router. Thus, this interface is considered **not applicable**.

- **Int-Matter:** IP-based connectivity protocol. Regarding to this interface, other devices in the network provide sufficient protection against DoS attacks and loss of essential functions for network operations, this interface is considered **not applicable**.

RESULT:	NA

Network monitoring mechanism

NMM-1: Applicability and appropriateness of network monitoring mechanisms

Results

The device is not intended to process the communication between networks which also involves public networks.

RESULT:	NA

Traffic control mechanism

TCM-1: Applicability of and appropriate traffic control mechanisms

Results

The device is not intended to process the communication between networks which also involves public networks.

RESULT:	NA

Confidential cryptographic keys

CCK-1: Appropriate CCKs

Results

According to **IXIT 03-ConKeys**, the following confidential cryptographic keys are processed by the DUT:

- **ConKey-Symkey**: AES-256 Symmetric Key for Data Encryption used to encrypt data at rest. Generated by the equipment.
  - Length: **256 bits** (AES).
  - Justification: In the Table 2, an **AES-256** key offers a security strength of **256 bits**, much higher than the required minimum of 112 bits. Therefore, it **complies** with the standard.

- **ConKey-4**: Cast auth device key - Widevine-based provisioning - Used to generate signatures in Cast authentication. Preinstalled
  - Length: **2048 bits** (RSA).
  - Justification: The Table 1 shows that an **RSA-2048** key has a security strength of **112 bits**, which meets the minimum required. Therefore, it **complies** with the standard.

- **ConKey-WeaveKey**: Weave key - legacy - Used to generate signatures in Weave authentication. Preinstalled
  - Length: **224 bits** (ECC).
  - Justification: In the Table 1, it is indicated that an **ECC key with f=224 bits** provides a security strength of between **112-128 bits**, thus meeting the 112-bit minimum standard. It **complies**.
- **ConKey-6**: Weave key - home - Used to generate signatures in Weave authentication. Preinstalled
  - Length: **224 bits** (ECC).
  - Justification: In the Table 1, it is indicated that an **ECC key with f=224 bits** provides a security strength of between **112-128 bits**, thus meeting the 112-bit minimum standard. It **complies**.

- **ConKey-13**: Widevine keybox - Used to identify and authenticate the device to Widevine servers. Preinstalled
  - Length: **128 bits** (AES).
  - Justification: According to the Table 2, an **AES-128** key offers a security strength of **128 bits**, which exceeds the 112-bit minimum. It **complies**.

- **ConKey-OTPKey**: Device-specific OTP key - A device-specific key generated by Google and used to protect various other keys. Preinstalled
  - Length: **128 or 256 bits** (AES).
  - Justification: Depending on the selected length, an **AES-128** key provides **128 bits** of security strength, and an **AES-256** key provides **256 bits** of security. In both cases, it **meets** the minimum requirement.

- **ConKey-ModelOTPKey**: Device model-specific OTP key - A device model-specific key used to protect the device-specific key. Preinstalled
  - Length: **128 or 256 bits** (AES).
  - Justification: Similar to ConKey-14, both **AES-128** and **AES-256** keys exceed the minimum 112 bits of security strength. It **complies**.

- **ConKey-BLEkey**: Used for protect BLE communication. Generated by the equipment.
  - Length: **128 bits** (AES).
  - Justification: According to the Table 2, an **AES-128** key offers a security strength of **128 bits**, which exceeds the 112-bit minimum. It **complies**.

- **ConKey-WifiKey**: Used for protect Wifi communication. Generated by the equipment.

- Length: **256 bits**.
- Justification: In the Table 2, an **AES-256** key offers a security strength of **256 bits**, much higher than the required minimum of 112 bits. Therefore, it **complies** with the standard.

All the ConKeys mentioned comply with the 112-bit minimum security strength requirement defined in the **Evaluation Case CCK-1**, as they use strong cryptographic algorithms such as **RSA-2048**, **224-bit ECC**, and **AES-128/256**.

As a sidenote, the following NIST references are to be used to determine the security level of each of the CCKs listed above.

Table 2: Comparable security strengths of symmetric block cipher and asymmetric-key algorithms

Security Strength	Symmetric Key Algorithms	FFC (DSA, DH, MQV)	IFC* (RSA)	ECC* (ECDSA, EdDSA, DH, MQV)
≤ 80	2TDEA	$L = 1024$ $N = 160$	$k = 1024$	$f = 160\text{-}223$
112	3TDEA[68]	$L = 2048$ $N = 224$	$k = 2048$	$f = 224\text{-}255$

Figure 33: NIST security strength for different algorithms 1 (Table 1)

Security Strength	Symmetric Key Algorithms	FFC (DSA, DH, MQV)	IFC* (RSA)	ECC* (ECDSA, EdDSA, DH, MQV)
128	AES-128	$L = 3072$ $N = 256$	$k = 3072$	$f = 256\text{-}383$
192	AES-192	$L = 7680$ $N = 384$	$k = 7680$	$f = 384\text{-}511$
256	AES-256	$L = 15360$ $N = 512$	$k = 15360$	$f = 512+$

Figure 34: NIST security strength for different algorithms 2 (Table 2)

RESULT:	PASS

CCK-2: CCK generation mechanisms

Results

As seen in **Evaluation Case CCK-1** and **IXIT 03-ConKeys**, the following confidential cryptographic key are used the DUT:

- **ConKey-Symkey**: AES-256 Symmetric Key for Data Encryption used to encrypt data at rest. Generated by the equipment.
    - Generation mechanism: The AES-256 key is generated by a hardware True Random Number Generator (TRNG) implemented in the NXP MIMXRT1172 (CAAM RNG4 module). This TRNG is based on physical entropy sources and complies with recognized standards for random number generation (ISO/IEC 18031 and NIST SP 800-90A/B/C). The generated key is 256 bits in length and provides sufficient security strength for the intended use.

- **ConKey-BLEkey**: Used for protect BLE communication. Generated by the equipment.
    - Generation mechanism: The BLE pairing process is handled by the Nordic nRF52832 Bluetooth SoC, which integrates a hardware True Random Number Generator (TRNG) and an AES cryptographic engine. During LE Secure Connections pairing, the device executes an Elliptic Curve Diffie-Hellman (ECDH) key exchange on curve P-256 to derive the Diffie-Hellman Key (DHKey). From this DHKey, the Short-Term Key (STK), Long-Term Key (LTK) and Identity Resolver Key (IRK) are derived using an AES-CMAC-based Key Derivation Function (KDF) as specified in the Bluetooth Core Specification v5.x. All random nonces and private keys are sourced from the on-chip TRNG, ensuring compliance with ISO/IEC 18031 entropy requirements and recognized best practices for key generation (NIST SP 800-56A and SP 800-108).

- **ConKey-WifiKey**: Used for protect Wifi communication. Generated by the equipment.
    - Generation mechanism: The WPA2 4-way handshake is processed by the Wi-Fi radio SoC SYN430132HKUBG / Broadcom BCM43013. During association with the Access Point, the device and AP exchange nonces (ANonce and SNonce) and derive the Pairwise Transient Key (PTK) and Group Temporal Key (GTK) from the Pre-Shared Key (PSK) using the standardized HMAC-SHA1 / SHA256 Key Derivation Function defined in IEEE 802.11i. The Wi-Fi SoC handles the generation of random nonces internally and executes AES-CCMP for link encryption. When the PSK is user-provided, the equipment enforces a strong-password policy (minimum 20 ASCII characters or 64 hex digits) to maintain adequate entropy. The PSK is device-generated, the hardware TRNG from the MIMXRT1172 is used to create the PSK. This mechanism complies with recognized best practices for key derivation (NIST SP 800-132 and ISO/IEC 11770).

RESULT:	PASS

CCK-3: Preventing static default values for preinstalled CCKs

Results

According to **IXIT 03-ConKeys**, the following confidential cryptographic keys are pre-installed by the DUT:

- **ConKey-4**: Cast auth device key - Widevine-based provisioning - Used to generate signatures in Cast authentication. Preinstalled and **unique** per device.
- **ConKey-WeaveKey**: Weave key - legacy - Used to generate signatures in Weave authentication. Preinstalled and **unique** per device.
- **ConKey-6**: Weave key - home - Used to generate signatures in Weave authentication. Preinstalled and **unique** per device.
- **ConKey-13**: Widevine keybox - Used to identify and authenticate the device to Widevine servers. Preinstalled
- **ConKey-OTPKey**: Device-specific OTP key - A device-specific key generated by Google and used to protect various other keys. Preinstalled and **unique** per device, is generated usaing a TRNG complies with the FIPS 140-3 standard, which is the latest version of the Federal Information Processing Standards (FIPS) Security Requirements for Cryptographic Modules.
- **ConKey-ModelOTPKey**: Device model-specific OTP key - A device model-specific key used to protect the device-specific key. Preinstalled. A device model-specific key (common to all devices of that model) used to protect the device-specific key prior to it being provisioned to a device. The key is **shared** between **Google** and the chip **vendor**. The key is used only for decryption. This CCKS key are shared parameters required for the equipment's intended functionality, so is **not applicable**.

RESULT:	PASS

General equipment capabilities

GEC-1: Up-to-date software and hardware with no publicly known exploitable vulnerabilities

Results

In terms of **Hardware Components** the test laboratory have analysed the Hardware BOM in order to check if any vulnerability is applicable to the DUT. The test laboratory has checked the NIST database, CVEdetails and INCIBE in order to search applicable vulnerabilities applicable to the main hardware components of the unit.

- SoC MIMXRT1172DVMAA: No public vulnerabilities/Has a public vulnerability but is not exploitable

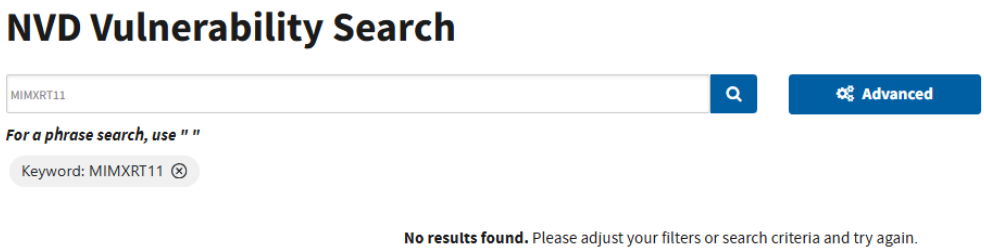


Figure 35: MIMXRT1172DVMAA Hardware search on NIST Database

- SoC SYN430132HKUBG: No public vulnerabilities/Has a public vulnerability but is not exploitable

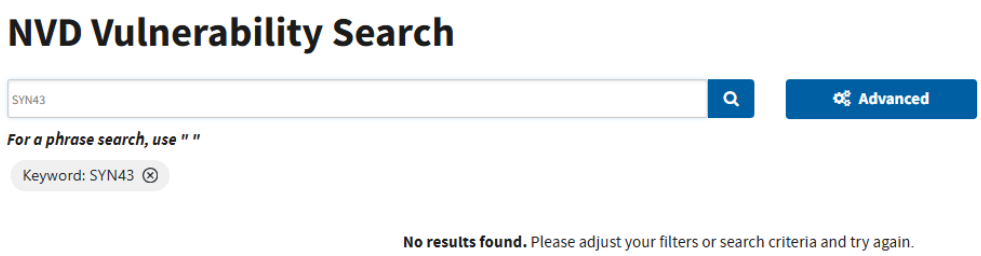


Figure 36: SYN430132HKUBG Hardware search on NIST Database

- SoC BGT60UTR13DAiP: No public vulnerabilities/Has a public vulnerability but is not exploitable

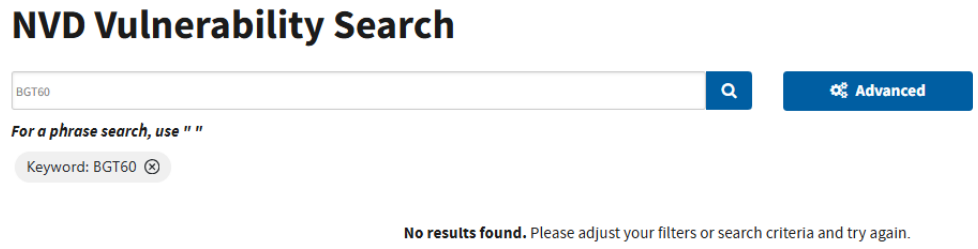


Figure 37: BGT60UTR13DAiP Hardware search on NIST Database

- SoC nRF52832-QFAA-R: No public vulnerabilities/Has a public vulnerability but is not exploitable

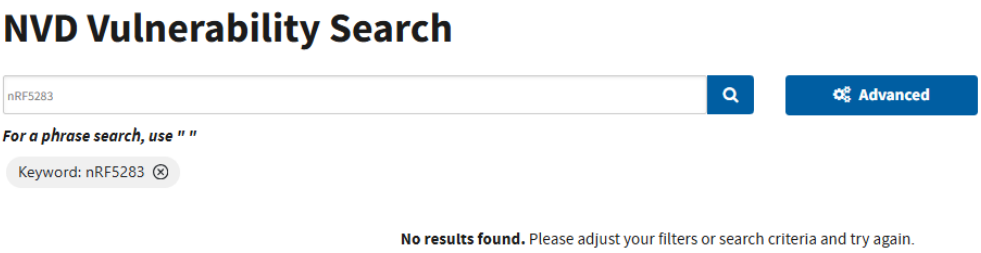


Figure 38: Hardware search on NIST Database

No vulnerable components were found. The equipment does not include exploitable vulnerabilities in the publicly known hardware that, if exploited, would affect security assets and network assets.

In terms of **Software Components** the test laboratory have analysed the Software BOM of the DUT in order to check if any vulnerability is applicable to the DUT. The test laboratory have analysed the SBOM provided by the customer and analysed with Synopsics BlackDuck.

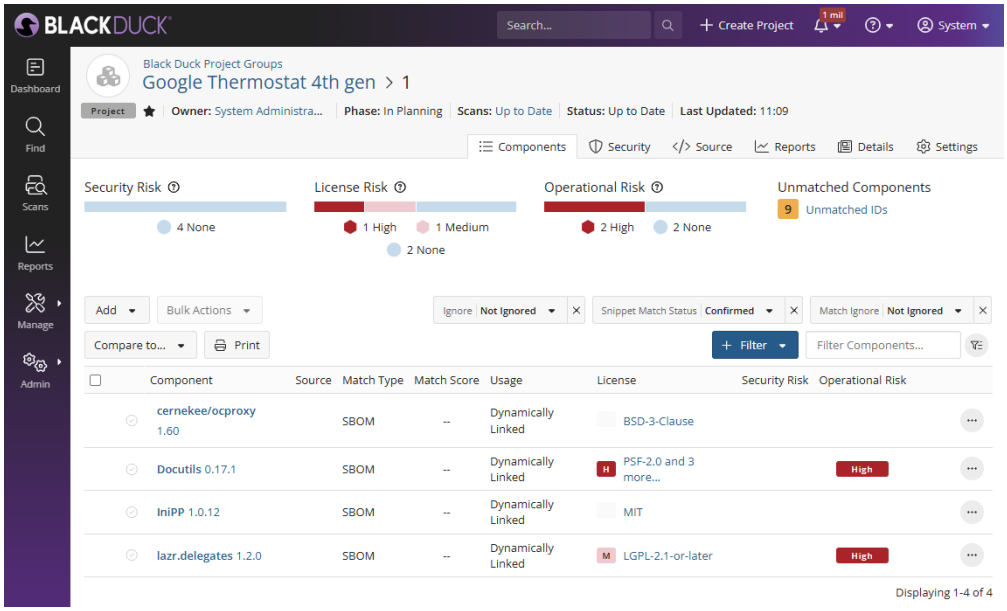


Figure 39: Synopsys Black Duck Analysis

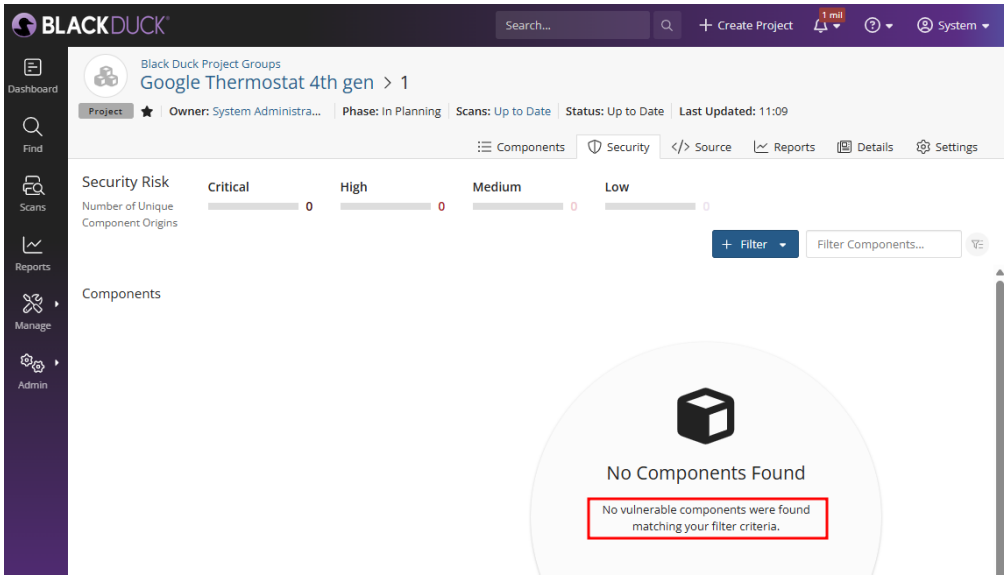


Figure 40: Synopsys Black Duck Vulnerability Report

No vulnerable components were found. The equipment does not include exploitable vulnerabilities in the publicly known software that, if exploited, would affect security assets and network assets.

RESULT:	PASS

GEC-2: Limit exposure of services via related network interfaces

Results

According to **IXIT 04-Intf** and **IXIT 18-SoftServ**, the following networks interfaces and services via network interfaces are enabled on the DUT on the default state:

- **Intf-WiFi**: The DUT needs it to connect to the Google cloud services. It also provides the functionallity of the DUT for the user. This interface is enabled by default on the device and is required for the device basic operation. The Test Laboratory have checked all the exposed services and characteristics via this interface scanning the ports and exploring all the services published.
    - Weave (TCP/UDP 11095, TCP 11096): Used for communication with Nest cloud services, so these operate over Thermostat's Wi-Fi connection to your home network and the internet.
    - Usonia (TCP 8012, 10002, 10006, 10101; UDP 10006, 10101): These ports support local network discovery and communication between Google Home ecosystem devices. This communication primarily happens over the local IP network, so these ports are open on the Wi-Fi interface.
    - Matter (TCP/UDP 5540): Thermostat acts as a Matter hub. Matter devices on your LAN (including those connected via Wi-Fi) will communicate with Thermostat over the Wi-Fi network using this por

  No characteristics or services has been found that is not required for normal operation of the device.

- **Intf-BLE**: Bluetooth is used during the setup process. This interface is enabled by default on the device and is required for the device basic operation.
  The Test Laboratory have checked all the exposed services and characteristics via this interface and exploring all the services published.

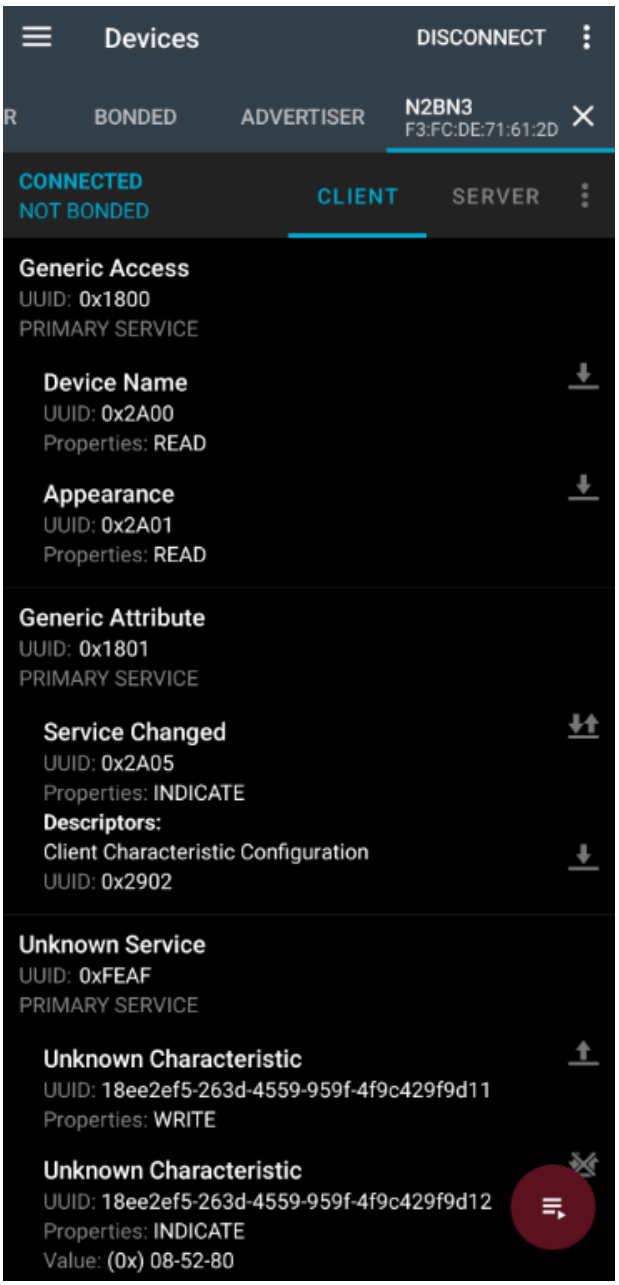


Figure 41: nRF connect GATT (Generic Attribute Profile)

No characteristics or services has been found that is not required for normal operation of the device.

RESULT:	PASS

GEC-3: Configuration of optional services and the related exposed network interfaces

Results

According to **IXIT 04-Intf**, no optional network interfaces that are manually deactivable were found.

Also, according to **IXIT 18-SoftServ**, no accessible optional software services were found.

RESULT:	NA

GEC-4: Documentation of exposed network interfaces and exposed services via network interfaces

Results

According to the information provided on **IXIT 01-Docs** the online manuals can be found on Google Nest: https://store.google.com/product/nest_learning_thermostat_4th_gen?hl=en-US

This documentation contain a description of all exposed network interfaces:

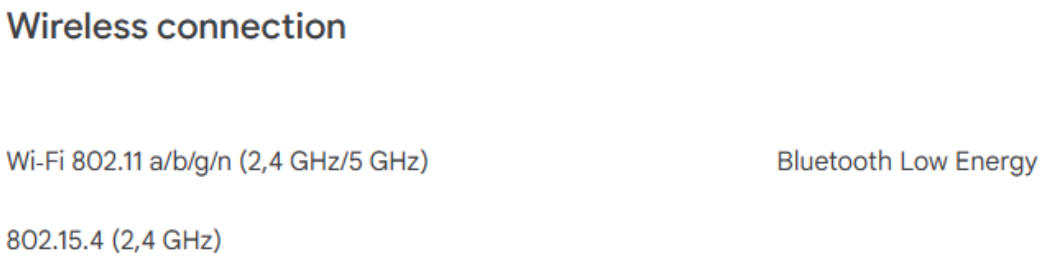


Figure 42: Reference on the Manual regarding all exposed network interfaces

RESULT:	PASS

GEC-5: No unnecessary external interfaces

Results

According to **IXIT 04-Intf**, the identified physical external interfaces are as follows:

- **Intf-MicroUSB**: Micro USB inside of the device not intended for customer use. It is a non-customer facing micro usb for connecting between device and debugging port enclosed in the device. The mirco USB port is disable for proudction devices.

Figure 43: MicroUSB

The TL has verified that all physical external interfaces found are documented.

RESULT:	PASS

GEC-6: Input validation

Results

According to **IXIT 04-Intf** and **IXIT 17-InpVal**, the identified external interfaces inputs that have potential impact on assets are as follows:

- **Intf-WiFi**: The DUT needs it to connect to the Google cloud services. It also provides the functionality of the DUT for the user. The inputs of this external interface has potential impact on security/network assets.

To test the input validation mechanism, fuzzing against this interface is performed with *OwFuzz tool*.

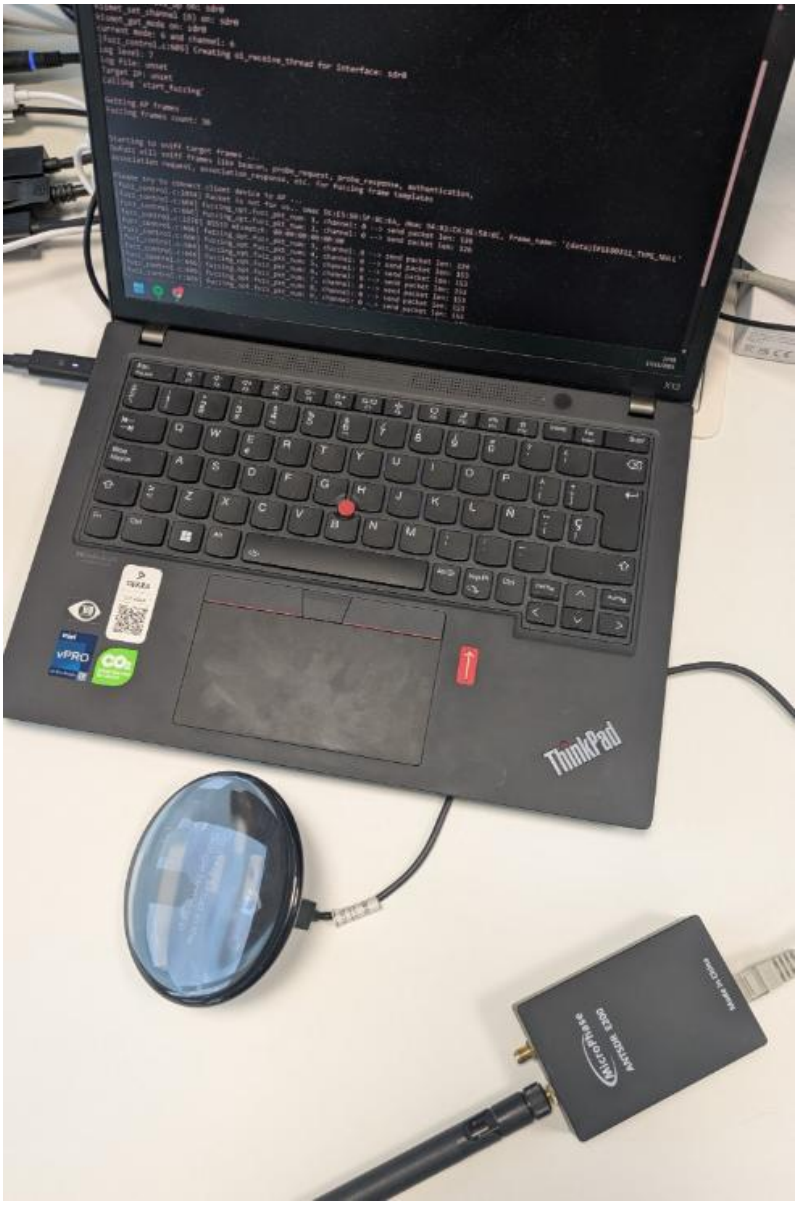


Figure 44: Test Setup Employed

```
root@analog:~/owfuzz/src# ./owfuzz -i sdr0 -m ap -c 6 -t CC:A7:C1:1C:4F:96 -b 94:83:C4:0E:58:6C -s 94:83:C4:0E:58:6C -T 2 -A WPA2_PSK_TKIP_AES
No seed value provided, using time(NULL)...
[fuzz_control.c:3079] Interface: sdr0, channel: 6
Fuzzing mode: ap
Target MAC: CC:A7:C1:1C:4F:96
Source MAC: 94:83:C4:0E:58:6C
Bssid: 94:83:C4:0E:58:6C
Fuzzing target's SSID: [owfuzz]
auth_type: 8 (WPA2_PSK_TKIP_AES)
test_type: 2 (TEST_FRAME)
sniff_frames: 0
Seed: srandom(NULL)...
[fuzz_control.c:578] Configuring interface: sdr0, channel: 6
kismet_interface_down on: sdr0
kismet_set_mode (6) on: sdr0
kismet_interface_up on: sdr0
kismet_set_channel (6) on: sdr0
kismet_get_mode on: sdr0
current mode: 6 and channel: 6
[fuzz_control.c:605] Creating oi_receive_thread for interface: sdr0
Log level: 7
Log file: unset
Target IP: unset
Calling 'start_fuzzing'

Getting AP frames
Fuzzing frames count: 36


Starting to sniff target frames ...
Owfuzz will sniff frames like beacon, probe_request, probe_response, authentication,
association request, association_response, etc. for fuzzing frame templates

Please try to connect client device to AP ...
[fuzz_control.c:2454] Packet is not for us.. smac DC:E5:5B:5F:8C:EA, dmac 94:83:C4:0E:58:6C, frame_name: '(data)IEEE80211_TYPE_NULL'
[fuzz_control.c:666] fuzzing_opt.fuzz_pkt_num: 1, channel: 0 --> send packet len: 126
[fuzz_control.c:666] fuzzing_opt.fuzz_pkt_num: 2, channel: 0 --> send packet len: 126
[fuzz_control.c:2276] BSSID mismatch: 00:00:00:00:00:00
[fuzz_control.c:666] fuzzing_opt.fuzz_pkt_num: 3, channel: 0 --> send packet len: 126
[fuzz_control.c:666] fuzzing_opt.fuzz_pkt_num: 4, channel: 0 --> send packet len: 153
[fuzz_control.c:666] fuzzing_opt.fuzz_pkt_num: 5, channel: 0 --> send packet len: 153
[fuzz_control.c:666] fuzzing_opt.fuzz_pkt_num: 6, channel: 0 --> send packet len: 153
[fuzz_control.c:666] fuzzing_opt.fuzz_pkt_num: 7, channel: 0 --> send packet len: 153
[fuzz_control.c:666] fuzzing_opt.fuzz_pkt_num: 8, channel: 0 --> send packet len: 153
[fuzz_control.c:666] fuzzing_opt.fuzz_pkt_num: 9, channel: 0 --> send packet len: 153
[fuzz_control.c:666] fuzzing_opt.fuzz_pkt_num: 10, channel: 0 --> send packet len: 153
[fuzz_control.c:666] fuzzing_opt.fuzz_pkt_num: 11, channel: 0 --> send packet len: 153
[fuzz_control.c:666] fuzzing_opt.fuzz_pkt_num: 12, channel: 0 --> send packet len: 153
[fuzz_control.c:666] fuzzing_opt.fuzz_pkt_num: 13, channel: 0 --> send packet len: 153
[fuzz_control.c:666] fuzzing_opt.fuzz_pkt_num: 14, channel: 0 --> send packet len: 153
[fuzz_control.c:666] fuzzing_opt.fuzz_pkt_num: 15, channel: 0 --> send packet len: 153
[fuzz_control.c:666] fuzzing_opt.fuzz_pkt_num: 16, channel: 0 --> send packet len: 153
[fuzz_control.c:666] fuzzing_opt.fuzz_pkt_num: 17, channel: 0 --> send packet len: 153
[fuzz_control.c:666] fuzzing_opt.fuzz_pkt_num: 18, channel: 0 --> send packet len: 153
[fuzz_control.c:666] fuzzing_opt.fuzz_pkt_num: 19, channel: 0 --> send packet len: 153
[fuzz_control.c:666] fuzzing_opt.fuzz_pkt_num: 20, channel: 0 --> send packet len: 153
[fuzz_control.c:666] fuzzing_opt.fuzz_pkt_num: 21, channel: 0 --> send packet len: 153
[fuzz_control.c:666] fuzzing_opt.fuzz_pkt_num: 22, channel: 0 --> send packet len: 153
[fuzz_control.c:666] fuzzing_opt.fuzz_pkt_num: 23, channel: 0 --> send packet len: 153
[fuzz_control.c:666] fuzzing_opt.fuzz_pkt_num: 24, channel: 0 --> send packet len: 153
[fuzz_control.c:666] fuzzing_opt.fuzz_pkt_num: 25, channel: 0 --> send packet len: 153
[fuzz_control.c:666] fuzzing_opt.fuzz_pkt_num: 26, channel: 0 --> send packet len: 153
```

Figure 45: OwFuzz Fuzzing tool execution

After some hours of execution, no strange behaviour is seen on the DUT.

- **Intf-BLE**: Bluetooth is used during the setup process. To test the input validation mechanism, fuzzing against this interface is performed with *Synopsys Defensics tool*.

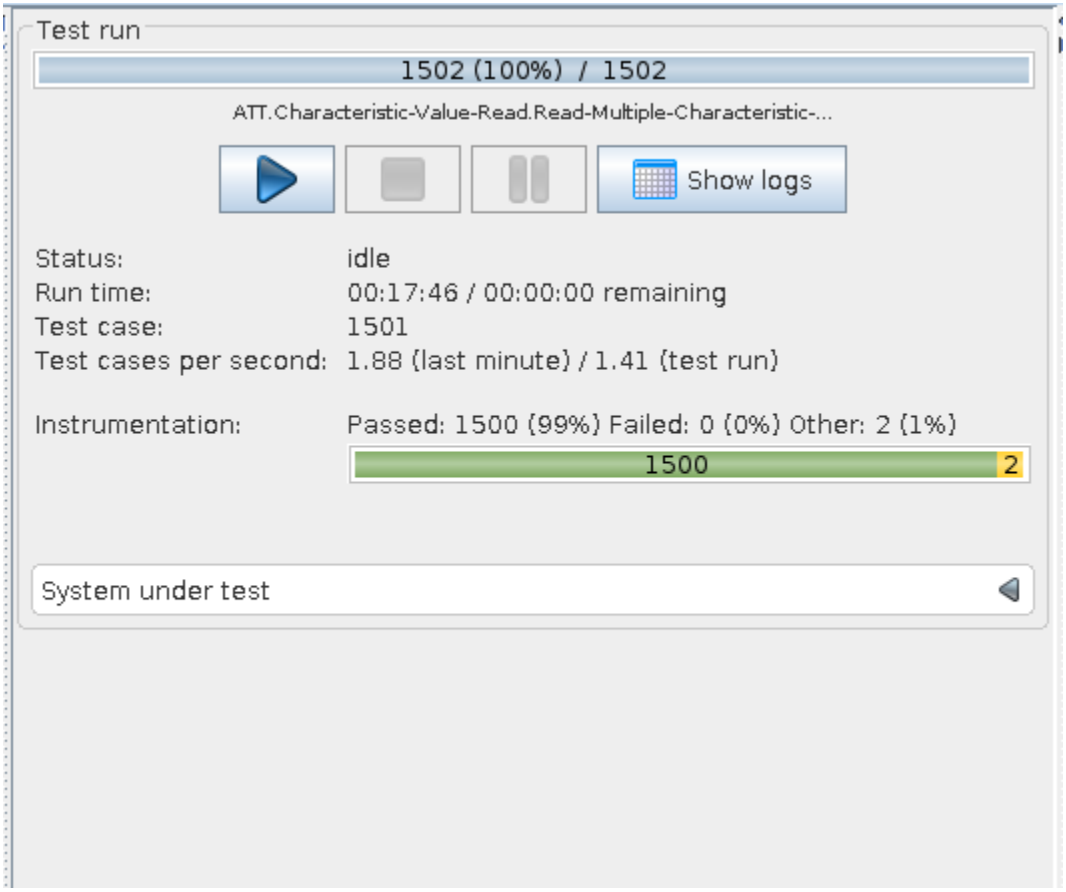


Figure 46: Synopsics Defecsics Fuzzing tool execution

Figure 47: Monitoring Fuzzing with Ellisys

After some hours of execution, no strange behaviour is seen on the DUT.

The inputs of this external interface has potential impact on security/network assets.

RESULT:	PASS

Cryptography

CRY-1: Best practice cryptography

Results

As seen in previous Evaluation Cases, when cryptography is used, it is agreed according to SOGIS agreed cryptographic mechanisms.

- **RSA 2048**

Agreed RSA primitive sizes.

Primitive	Parameters' sizes	R/L	Notes
RSA	$n \geq 3000$, $\log_2(e) > 16$	R	
	$n \geq 1900$, $\log_2(e) > 16$	L [2025]	27-LegacyRSA

Figure 48: RSA 2048 accepted by sogis

- **AES 256** and ECC with a length of 256 bits. As seen in **Evaluation Case CCK.1**, these cryptographic algorithms have more than 112 bits of security strength. Therefore, they comply with best practices.
- **AES 128** and ECC with a length of 128 bits. As seen in **Evaluation Case CCK.1**, these cryptographic algorithms have more than 112 bits of security strength. Therefore, they comply with best practices.

Agreed Block Ciphers.

Primitive	Parameters' sizes	R/L	Notes
AES [FIPS197, ISO18033-3]	k = 128 bits	R	
	k = 192 bits	R	
	k = 256 bits	R	
Triple-DES [FIPS46-3, ISO18033-3]	k = 168 bits	L[2027]	2-SmallBlocksize
	k = 112 bits	L [2024]	2-SmallBlocksize 3-TripleDES2key

Figure 49: AES 128 & AES 256 accepted by sogis

- **ECC** key with **f=224** bits. As seen in **Evaluation Case CCK.1**, these cryptographic algorithms have more than 112 bits of security strength. Therefore, they comply with best practices.
- **TLS_AES_128-GCM_SHA256** (0x1301)

TLS code	Cipher Suite	R/L	Notes
TLS v1.3 Cipher Suite			
0x1302	TLS_AES_256_GCM_SHA384	R	
0x1301	TLS_AES_128_GCM_SHA256	R	
0x1304	TLS_AES_128_CCM_SHA256	R	
TLS v1.2 Cipher Suite			
0xC02C	TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384	R	
0xC02B	TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256	R	
0xC0AD	TLS_ECDHE_ECDSA_WITH_AES_256_CCM	R	
0xC0AC	TLS_ECDHE_ECDSA_WITH_AES_128_CCM	R	

Figure 50: Cipher suite agreed by sogis.eu

RESULT:	PASS