Cybersecurity Integration Guide/DA14535

This guide provides suggestions for the integration of DA14535 into products that must address cybersecurity concerns. The target audience is the security development team of manufacturers of such products.

The EN 18031-1:2024 and EN 18031-2:2024 standards have been used as a baseline, but the mentioned approach and security integration principles can apply for other similar specifications.

Disclaimer: Please be aware that Renesas provides no guarantee that following this guide will result in conformity with these standards or any other security specification. It is the responsibility of the product manufacturer to ensure the product conforms.

Contents

Figures

Tables

1. Terms and Definitions

RED	Radio Equipment Directive
LE	Low Energy
UART	Universal Asynchronous Receiver Transmitter
OTP	One Time Programmable. DA14535 integrates an OTP memory.
Etc.	

2. References

[1] DA14535 Ultra Low Power Bluetooth 5.3 SoC Datasheet, Revision 3.0, Renesas Electronics.

[2] SW-B-002 - Release Notes SDK v.6.0.24.1464, Revision 1.7, Renesas Electronics.

[3] [UM-B-119: DA1453x/DA1458x SW Platform Reference Manual], Revision 3.6, Renesas Electronics

[4] [UM-B-118 - SDK6 System Features and Peripheral Overview], Revision 3.1, Renesas Electronics.

[5] [DA1453x Tutorial BLE Security], Revision 3.2, Renesas Electronics

[6] [SmartBoot], Revision 1.3, Renesas Electronics

[7] [https://lpccs-docs.renesas.com/sdk6_kll/index.html]

[8] DA1453x/DA1458x Tutorial Software Update Over the Air (SUOTA), Revision 4.3, Renesas Electronics

Note 1 References are for the latest published version, unless otherwise indicated.

3. Introduction

This guide is intended for the equipment manufacturer that wants to launch products that comply with cybersecurity regulations. Specifically, it describes the steps to be followed for assessing the security of products based on the DA14535 or DA14535MOD products of Renesas.

Before proceeding, we expect the reader to understand and acknowledge the following three statements:

- Using DA14535 and following this guide provides no guarantee of conformity. Each end-product has specific security requirements that must be fully assessed on their own.

- Conforming to a specific standard or regulation does not guarantee the product is secure, only that it integrates protection mechanisms that at the time of the assessment were considered adequate for the specific product and its intended operating environment.

- There are other ways to protect a product which may not be described in this guide, or even in the referenced standards. Not following this guide or not adhering to the referenced standards does not mean a product does not comply with cybersecurity regulation.

3.1 Cybersecurity Regulation and the EN 18031-x Standards

There are numerous standards and regulations relating to cybersecurity. In this guide we use the EN 18031-x family of standards and specifically:

- EN 18031-1 which applies to network connected radio equipment
- EN 18031-2 which applies to network connected radio equipment, Childcare devices, Toys and Wearables

These standards have been harmonized to serve as evidence of conformity with the RED (Radio Equipment Directive) Clauses 3.3(d), and 3.3(e). A product in scope of 3.3(d) and/or 3.3(e) must conform to bear the CE mark. Please consult your CE marking consultant or internal experts for more information.

Renesas has tested the conformity of DA14535 Module and the DA14535 Development Kit against these clauses. Please note that a product manufacturer cannot leverage Renesas' declaration of conformity. We believe - to the best of our knowledge - that following this guide will make it easier to achieve a conforming product. But it is always the responsibility of the product manufacturer to run your own analysis and testing, and to prepare your own declaration of conformity.

3.2 How to Read this Guide

This guide uses a fictional product example and takes the reader briefly through the security analysis steps. Run such an exercise at the definition phase of a new product, so that the product architecture considers cybersecurity from the start. It is good practice to follow a secure-by-design development process. However, this document does not deal with process but only with Product security.

At various points, we refer to **Recommendations & Best Practices. These are marked in bold fonts with the corresponding "Recommendation" or "Best Practice" wording, so that they can be identified faster.**

3.3 Example Product to be Used in this Guide - A Toy Doll with Sound

We will use as an example a fictional "Toy Doll with Sound" product. This toy is intended for young kids who can play with it as a standalone toy or interact with it as they grow up via a tablet or smartphone.

The toy communicates with the tablet or smartphone via a Bluetooth Low Energy (LE) link. An authorized adult individual (a parent) can control and configure the toy via the Bluetooth LE link, over a smartphone or tablet.

The doll has a microphone and a speaker. These enable it to capture and play sounds. Sounds captured can be stored in the Flash memory or can be transferred asynchronously and at slow speed via the Bluetooth interface. Sounds from flash (or sent asynchronously over the Bluetooth interface) can be played on the speaker.

There is no real-time bidirectional sound communication between the toy and the smartphone.

The figure gives a block diagram of the Toy and illustrates its connection over Bluetooth LE to a Smartphone or Tablet peer device.

Please note that the diagram illustrates the DA14535 device included in the module (DA14535MOD). If the end-product uses the bare device on the main board, similar analysis applies.

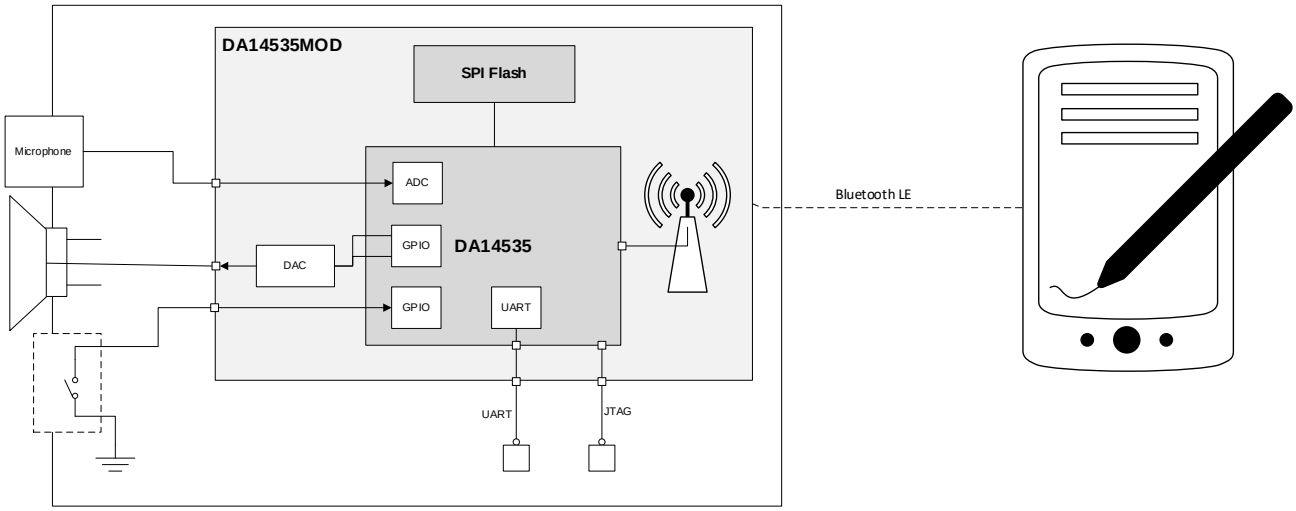

Figure 1. Example Product - A Toy Doll with Sound

4. Cybersecurity Analysis of the Product

When analyzing the product, we start by identifying the interfaces that it exposes, the assets that need protection and the mechanisms that are used for protecting them. The analysis is not complete until we assess all security risks that remain in the product.

4.1 Interfaces

The interfaces can be classified in four categories:

- Network interfaces
  - In the example product, there is only one network interface; the wireless interface which uses the Bluetooth LE protocol for communication to a peer device (like a smartphone or tablet).
- User interfaces
  - In the example product, there are three user interfaces on the product itself and another user interface that connects to the system via the Bluetooth LE Network Interface
    - Microphone - This is an input interface that feeds audio data in the product for storage or for transfer over the Bluetooth LE network interface. The stored audio data may contain private information and should therefore be protected from unauthorized access. For example, a parent alone may read aloud a password or a PIN without noticing that what he/she says is recorded on the device. **It is good practice to treat all data that is stored on the device as private.**
    - Speaker - This is an output interface that can play back sounds stored in the product or transferred over the Bluetooth LE network interface.
    - Button - This is a Yes/No input interface. It is not used for normal operation of the product but only at configuration by the authorized user (parent). To facilitate this, it is not readily available on the Toy but can only be accessed by unscrewing and opening the battery department. **It is good practice to mechanically protect user interfaces that can be used for configuring the device by authorized users, especially if the product is intended to be normally used by non-authorized users.**
    - Smartphone Application UI - This is an elaborate user interface that allows a kid to control and interact with the Toy. It also provides advanced features for configuration of the toy from the authorized user (parent). **It is strongly recommended that applications on peer devices implement different user privileges for kids and parents in Toys and Childcare devices, that can be accessed only by using strong passwords.**
- Machine interfaces
  - A machine interface is used when another device directly connects to the product. In the example product we might consider the connected smartphone as another device, and the Bluetooth LE connection as a machine interface. However, we are already considering this for our security analysis as a "network interface".

- Physical interfaces, including the ones used for debugging
  - The above-mentioned user interfaces rely on physical connections and are considered there for our security analysis.
  - In our example, we have two more physical interfaces:
    - The JTAG interface. This is a debug interface and is exclusively used during product development and testing. Before the product reaches the market, the JTAG interface is disabled. **It is strongly recommended that the JTAG interface is always disabled by setting the corresponding OTP flag of DA14535 at the end of the production cycle and before the product exists the factory floor.**
    - The UART Interface. This is used on the factory floor during production. DA14535 has a feature called UART-boot which allows you to download special firmware in RAM. We program such firmware so that UART behaves as a debug interface allowing program/read Flash and OTP. Downloading this special firmware is not allowed after the product leaves the factory floor, by disabling UART-boot in the OTP settings. **It is strongly recommended to disable UART-boot at the end of the production cycle and before the product exists the factory floor.**

4.2 Assets

The interfaces may provide access to various keys, data, code or other information which should be protected. These assets can be classified as:

- Network Assets
  - The Bluetooth Stack, including the information and data that it communicates. In the example product, the Bluetooth stack is used in two configurations:
    - Paired, when the authorized individual (parent) needs access to the toy for configuration or update.
    - Public, when a non-authorized user (child) plays with the toy. Please note that in this analysis we assume that while playing, no privacy or security related information can be accessed. This is a matter of how the toy is designed. If it is not true, we must also secure this configuration.
- Privacy Assets
  - User data that is collected and stored or communicated. In the example product, such data might be sound recordings, usage patterns, location, etc.
    - When the child is playing, such data may be stored but cannot be retrieved or communicated. One must be an authorized user (parent) to access it.
    - **It is good practice to treat all data that is stored on the device as private.**
- Security Assets
  - The application firmware. Being able to read the application firmware is not a security concern, although we still want to protect our IP. But protecting it from manipulation is critical, as via the application firmware someone can access all other interfaces and assets.
  - The user application. Although not part of the product itself, the user application that runs on the smartphone is considered an associated service that runs on a peer device. If our example product, the user application supports two modes:
    - Parent: Allows configuration of the toy, SW updates, Readout of stored data, Factory reset etc.
    - Child: Allows only temporary interaction with the toy, e.g. selection of play mode, simple yes/no responses etc.
  - Stored keys that are used by the various protection mechanisms. In the example product these are:
    - Bluetooth bonding parameters, including encryption keys
    - Keys to encrypt the application image
    - Keys used to sign and check the signature of the application image
    - Keys to encrypt storage of private data

4.3 Protection Mechanisms

Each interface and each asset that is accessible via the interfaces may need to be protected from manipulation or information leakage. The product must provide mechanisms for implementing security-related system functionality. Such mechanisms may include:

4.3.1 Mechanisms to control access to assets and interfaces

Access to the assets and the interfaces must be controlled, to ensure that an attacker cannot modify or read them if that impacts the security of the system. The physical or logical mechanisms to control access must be documented. In our example product:

- The wireless interface of the toy can only be accessed via the Bluetooth Stack. The Bluetooth stack resides in the DA14535 ROM and is only accessible via function calls from the product's firmware and debug interfaces before production.
- The user interfaces on the toy (microphone, speaker) connect to input/output ports of the DA14535 device. They are accessible via function calls from the product's firmware. The user (child) can access them physically when using the toy, e.g. by talking to the microphone or by listening to the speaker.
- The button user interface is not accessible by the child but only by the parent. Access is physically controlled because the button is located within the battery compartment. This means that someone must physically get hold of the toy to unscrew the cover of the battery compartment and access the button. **It is good practice to mechanically protect user interfaces that can be used for configuring the device by authorized users, especially if the product is intended to be normally used by non-authorized users.**
  The user application interface is accessible from the child and the parent. Access to individual functions is controlled by the configuration of the user application. Security and privacy related functions are only available to the authorized user.
- The DA14535 module has two debug interfaces (JTAG, UART). Access to these interfaces would compromise the security of the product, therefore it is controlled by two mechanisms:
  - Physical restriction - these interfaces are not exposed (they are not on the product surface).
  - Logical disabling - these interfaces are blocked by OTP flags in the production line.
- User data is stored on flash memory, inside the toy and cannot be externally read or modified. It can only be accessed by the application firmware via function calls. As an extra mechanism of protection, such data is also stored in an encrypted format. See section "Mechanisms to securely store and delete security-related and private information", for more information.
- The application firmware is stored in the flash and copied to RAM for execution by the boot sequence. There are two mechanisms to control access to it. Physically, someone would have to open the product to get access to the flash and the firmware. But even if one did, the firmware is stored with a signature that confirms its integrity. The DA14535 provides a secondary secure bootloader that must be stored in OTP. This bootloader checks the firmware image signature and only uses it if the signature is valid. **It is strongly recommended to write the secondary secure bootloader in the DA14535 OTP during production.**
  - The firmware image can also be accessed outside the device during SW update. Besides the mechanisms provided by the Bluetooth stack, having the firmware image signed prevents someone from altering it during transfer.
  - As an extra method of protection, the application firmware is always stored and transferred in an encrypted format.
- The user application on the smartphone or tablet is accessible to the child and the parent. As explained before, only parental access needs to be controlled. This can be achieved with two mechanisms that may coexist:
  - Use a password to get access to the privileged functions of the application. In this guide we focus on the product itself (the toy), so we will not elaborate more on this mechanism. It is up to the user application designer to provide an implementation of this mechanism.
  - Enable the privileged functions only if the specific peer device (and its user) has paired (in Bluetooth terms) with the toy. For more info see section: "Mechanisms of secure Communication".
  - **It is recommended that any application which provides functionality that influences product security, is designed, developed and tested so that it controls access and allows only authorized users to perform such functionality.**
- Access to the stored keys must be restricted only to the functions of the application firmware that use them. As an extra mechanism to ensure the keys cannot be altered or retrieved, the keys are stored in an encrypted

format. See section "Mechanisms to securely store and delete security-related and private information", for more information.

A summary of access control mechanisms is given in the table:

Table 1. Access control overview

Asset or Interface	Type of access control mechanism
Wireless Interface	Logical - from the stack functions
Bluetooth Stack	Physical - resides in ROM Logical - from application firmware
Microphone & Speaker User Interfaces	Logical - from application firmware
Button interface	Physical - in the battery compartment. Logical - from application firmware
Debug interfaces (JTAG, UART)	Physical - not exposed on the product Logical - disabled at production
Stored User Data	Physical - not exposed on the product Logical - from application firmware Secured - stored in encrypted format
Application Firmware	Physical - not exposed on the product Secured - encrypted during transfer and storage Secured - signed during transfer and storage
Functions of the user application	Logical - password based privileged access Secured - only over paired Bluetooth connection
Stored Keys	Physical - not exposed on the product Logical - from application firmware Secured - stored in encrypted format

4.3.2 Mechanisms to support and enforce authentication of users and/or peer devices

The user of the device needs in some cases to have access to assets and interfaces. To ensure that this access does not pose a security threat, a mechanism to authenticate the user is required and must be documented.

In connected devices, users also interact via a peer device like a smartphone or tablet. Connecting to such peer devices - when used to access security or privacy related assets and interfaces - needs an authentication mechanism that must be documented.

In our example product, we have two users:

- Parents: They need to have access to the toy's security and privacy assets.
- Children: They only play with the toy and their access is limited to non-private and non-secure assets and interfaces.

Both users interact over the Wireless interface through functions of the user application. The application uses the mechanisms of the Bluetooth stack to get read and/or modify access to individual characteristics in Toy.

When the accessed characteristics are related to privacy or security, the Bluetooth Stack is configured by the Application Firmware to block access if the peer device is not paired.

The Bluetooth pairing mechanism ensures that only authorized users (in this case, the parent) can connect.

DA14531 supports Bluetooth pairing and more specifically the "Secure Connections" standards feature which provides enhanced protection from security threats. In our example product, we use Secure Connections with the "Numeric Comparison" pairing mechanism. As described in [5], to support this method, the device should be equipped with a "display" that presents a 6-digit number to the user. The same number should match the digits

shown on the smartphone application, and the user must confirm the match to complete pairing. Since the Toy doll does not have a display, this is accomplished by utilizing the speaker, which reads out the 6 digits.

This procedure is quite tedious to repeat every time the parent wants to access the device. This is why we use another standard Bluetooth mechanism: Bonding. Bonding allows to store the connection parameters after pairing, so that the two devices (Toy and Parent's phone) are reconnected securely every time.

It is strongly recommended to use Bluetooth Secure Connections for Pairing and Bonding.

If the device has some means of providing the 6-digit number to the user, it is recommended to use the Numeric Comparison pairing method.

4.3.3 Mechanisms to securely update the product's SW

The application firmware that runs within the device must be updatable. Via SW updates, the manufacturer can deploy new versions of the application firmware, in the case that some security vulnerability is found and fixed. But making the device SW updatable is also a risk, because an attacker might exploit this to load malicious code. Therefore, it is critical that any SW update mechanism is secure and does not allow running other firmware besides the one intentionally pushed by the device manufacturer.

The DA14535 Device can support secure updates by using a secondary bootloader. This secondary bootloader is available with the SDK (version 6.0.24 and later). **It is strongly recommended to write the secondary secure bootloader in the DA14535 OTP during production.**

If the secure bootloader cannot be used for any reason, it is strongly recommended that the security risks are analyzed and other mechanisms to protect the user and the network are employed.

4.3.4 Mechanisms to securely store and delete security-related and private information

Some information that is stored on the device must be protected from readout and from manipulation. This includes at least:

- Security-related information (e.g. connection keys)

- Sensitive personal information of the end user

When the device is initialized and used on the field, such information cannot be modified or read, unless by the authorized user. Other entities or unauthorized users should not have any access.

When the device is returned to the manufacturer, handed over to a different end-user or disposed, this information should be deleted, to protect them from more elaborate (e.g physical) attacks.

It is strongly recommended to employ secure storage and deletion for all network and privacy assets of the product.

The DA14535 SDK 6.0.24 comes with functions that support encrypted storage. Using this set of functions, the application can also delete the security-related and the private information when needed. The keys used by this mechanism are created at runtime by the application image and are unique per device. **It is strongly recommended to use unique per device keys for encrypted storage, to prevent scalable attacks.**

In our toy example, the application firmware will utilize these secure storage and deletion mechanisms for network keys (bonding parameters) and for any user data.

4.3.5 Mechanisms of secure Communication

The communication channels may carry information that is critical for device security or the user privacy. When such information is transferred, the protocol must provide mechanisms for encryption. The Bluetooth Low Energy protocol provides a variety of security mechanisms, depending on the user interface capabilities of the devices that communicate.

When possible, **it is strongly recommended to use the Bluetooth Secure Connections mechanism with Numeric Comparison.** This mechanism provides very strong keys for encrypting the channel and are immune to a variety of attacks - including MITM.

4.3.6 Key Management and Cryptography

Many of the protection mechanisms explained here utilize cryptography functions. When a cryptography function is used - it must use best practice implementations that correspond to the protection needed for each of the supported assets and interfaces. Where these functions use keys, these keys must be generated, managed and stored securely.

If DA14535 and SDK 6.0.24, Renesas provides cryptography functions and key generation tools. Specifically:

- To generate the asymmetric keys for firmware image signing

- To generate the symmetric key for image encryption

- To generate keys for Bluetooth communication

- To derive a unique per device key for secure storage

It is strongly recommended to use the methods provided (or equivalent ones) for key generation & management.

4.3.7 Generic Provisions

There are some generic provisions that the end-product must support. Most of these have been described in previous sections, but we provide here a table that organizes them as per the 18031-x section 5.10.

Provision	Recommendation
Up to date SW, without known vulnerabilities	Before product launch, check public vulnerability databases of any SW library used in SDK 6.0.24 or used in additional application code. Also check specifically for any mention of DA14535. Periodically check, and - if something is found - roll out a SW update on the field with the appropriate mitigation.
Limit Exposure of services	In the Toy example, the only services that are exposed in the factory default state are the BLE services needed to connect and configure the device so that the authorized user (parent) can access any assets that are not available to the child when playing with the toy.
Configure Optional Services	
Document exposed interfaces	The user documentation (use guide) of the example Toy product describes that the toy must be configured by the parent and provide step by step easy to follow instructions of how to configure the toy, how to access personal data and how to perform SW updates via the companion smartphone application.
No unnecessary interfaces	The example Toy product expose**s** the minimal set of interfaces needed for operation. Any other interfaces are internally disabled and not exposed on the device enclosure.
Document sensing capabilities	The user documentation (use guide) of the example Toy product describes that the product has a build in microphone and that any recorded sound is stored in a way that only the parent can access it
Input Validation	Input validation via the Bluetooth interface is provided by the implementation of the protocol in DA14585.

4.3.8 Application-specific mechanisms - 18031-2

The mechanisms described here are application specific. DA14535 and SDK 6.0.24 APIs can be utilized to implement such mechanisms but there is no generic implementation.

Logging

The equipment shall provide a mechanism to log internal activities that are relevant for the protection of events, unless the logging of Events is done by other means outside the equipment. In the example Toy product, the events that can be logged are:

- SW updates and whether they are successful or not.

- Pairing attempts and whether they are successful or not

This information can be logged in the flash. If the product has flash size restrictions, the designer may consider logging by the smartphone application, but a risk analysis should be performed against the specific use case.

User Notification

The device should notify the user when a change that affects security or privacy occurs. In the example Toy product, such notifications are delivered over the user interface of the smart**p**hone application and relate to:

- Software Updated

- Pairing and Bonding

- Secure Data Deletion ("Factory Reset")

It is strongly recommended that the application firmware implements the logging of security events and generates appropriate user notifications.

4.3.9 Application-specific mechanisms - 18031-1

The mechanisms described here are application specific. DA14535 and SDK 6.0.24 APIs can be utilized to implement such mechanisms but there is no generic implementation.

Resilience

The Bluetooth LE protocol includes mechanisms that are inherently resilient to DoS attacks. However, as with all wireless protocols, there is always the chance that an attacker fully blocks the available RF spectrum. The application designer must assess if such an attack would compromise the security of the product and - if yes - develop countermeasures. In the example toy project such an attack would not allow any security or privacy compromise - although it could impact user experience.

Network Monitoring

If the device is **used** to connect other devices to the network, it must implement network monitoring mechanisms, including at least monitoring for DOS attacks. In the example of the Toy product, this is not needed.

Traffic Control

If the device is used to connect other devices to the network, it must implement traffic control mechanisms. In the example of the Toy product, this is not needed.

Depending on the actual product, the manufacturer must consider implementing (in the application firmware) network monitoring and traffic control mechanisms.

4.4 Risks

Even with all the implemented security mechanisms, a product may not be 100% secure. E.g., if an attacker gets physical access to the product or finds out the value of shared secrets, he/she may be able to device elaborate attacks that compromise the system. An analysis of these risks must be carried out and documented. A simple table like the one below should be enough.

Table 2. Cybersecurity Risks

Risk ID	Affected Asset or Interface	Severity of the attack	Method of the attack	Risk assessment
R01				

Note: At the time of preparing this guide, there was no vulnerability reported for DA14535 on the public vulnerability databases.

Please note that **any published vulnerabilities at the time of product launch must be added to a Risk Table and the risk must be adequately assessed and accepted or mitigated.**

4.5 Mapping to EN 18031-x Provisions

The following table provides a proposed mapping of the above sections with the provisions of the EN 18031-1 and 18031-2 standards.

Table 3. Mapping to EN 18031-x Provisions

Section	18031-1	18031-2	Comment
4.3.1 - Mechanisms to control access to assets and interfaces	5.1 [ACM]	5.1 [ACM]	18031-2 has more [ACM] provisions than 18031-1, specifically targeting Toys and Childcare devices.
4.3.2 - Mechanisms to support and enforce authentication of users and/or peer devices	5.2 [AUM]	5.2 [AUM]	Similar provisions on both standards
4.3.3 - Mechanisms to securely update the product's SW	5.3 [SUM]	5.3 [SUM]	Similar provisions on both standards
4.3.4 - Mechanisms to securely store and delete security-related and private information	5.4 [SSM]	5.4 [SSM] 5.7 [DLM]	Similar provisions on both standards. 18031-2 also needs a deletion mechanism.
4.3.5 - Mechanisms of secure Communication	5.5 [SCM]	5.5 [SCM]	Similar provisions on both standards
4.3.6 - Key management and cryptography	5.9 [CCK] 5.11 [CRY]	5.9 [CCK] 5.11 [CRY]	Similar provisions on both standards
4.3.7 - Generic provisions	5.10 [GEC]	5.10 [GEC]	Similar provisions on both standards
4.3.8 - Application-specific mechanisms - 18031-2	-	5.6 [LGM] 5.8 [UNM]	Applies only to 18031-2
4.3.9 - Application-specific mechanisms - 18031-1	5.6 [RLM] 5.7 [NNM] 5.8 [TCM]	-	Applies only to 18031-1

Note 1 Please note that this guide does not reference EN 18031-3. Although the provisions are very similar to those of 18031-1 and 18031-2, we do not target the use cases of EN 18031-3 with DA14585.

5. Documenting Conformity

Conformity assessment and declaration are the responsibility of the end-product manufacturer. Renesas recommends working with a specialized expert. Working with a notified body and a competent test house is also beneficial, especially if the provisions of the EN 18031-1 and 18031-2 standards are not fully met, or if one of the restrictions mentioned in the implementing act applies to the product.

Below is a simple list of work products that should be prepared and maintained by the manufacturer. The list is non-exhaustive and Renesas cannot provide any guarantee that something else will not be required.

Conformity Dossier:

- Documentation of Conformity
  - A list of all assets, interfaces and protection mechanisms.
  - Description of how each provision of the 18031-x standards is fulfilled by the corresponding mechanisms.
  - Path through all applicable decision flows in the 18031-x standards
  - List of risks and vulnerabilities (if any, see section 4.4).
- Test Report
  - Either from test house if externally performed or from internal testing
  - Include at least pass/fail and test description
  - Including test evidence records is optional but recommended
- Device information
  - Model, Commercial Name, Origin
  - HW Design, Bill of Materials
  - SW Version used for the testing
  - Photos of the product showing all interfaces
- Physical storage of the HW and the SW used for the tests
- Test instructions so that the tests can be repeated if needed

6. Conclusions

This guide used an example fictional product (a Toy) to provide suggestions on how a product can address cybersecurity requirements coming from RED articles 3.3 d, e. The target audience is the security development team of manufacturers of such products.
Disclaimer: Please be aware that Renesas provides no guarantee that following this guide will result in conformity with these standards or any other security specification. It is the responsibility of the product manufacturer to ensure the product conforms.

Appendix A - Recommendations and Best Practices

In this appendix, we list all recommendations and best practices mentioned above and (if applicable) we point to corresponding locations in the referenced user documentation.

Recommendation/Best practice	Pointer to DA14535 Documentation
It is good practice to treat all data that is stored on the device as private.	
It is good practice to mechanically protect user interfaces that can be used for configuring the device by authorized users, especially if the product is intended to be normally used by non-authorized users.	
It is strongly recommended that applications on peer devices implement different user privileges for kids and parents in Toys and Childcare devices, that can be accessed only by using strong passwords.	
It is strongly recommended that the JTAG interface is always disabled by setting the corresponding OTP flag of DA14535 at the end of the production cycle and before the product exits the factory floor.	[1] section 4.4.2, Table 34, Entry #4 (Disable SWD)
It is strongly recommended to disable UART-boot at the end of the production cycle and before the product exits the factory floor.	[1] section 4.4.1, Table 33, Address 7F82FC0 and 7F82FC4 (boot from OTP).
It is recommended that any application which provides functionality that influences product security is designed, developed and tested so that it controls access and allows only authorized users to perform such functionality.	
It is strongly recommended to use Bluetooth Secure Connections for Pairing and Bonding.	[5]
If the device has some means of providing the 6-digit number to the user, it is recommended to use the Numeric Comparison pairing method.	[5]
It is strongly recommended to write the secondary secure bootloader in DA14535 OTP during production.	[8] Section 24.4
If the secure bootloader cannot be used for any reason, it is strongly recommended that the security risks are analyzed and other mechanisms to protect the user and the network are employed.	
It is strongly recommended to employ secure storage and deletion for all network and privacy assets of the product.	[5]
It is strongly recommended to use unique per device keys for encrypted storage, to prevent scalable attacks.	[5]
it is strongly recommended to use the Bluetooth Secure Connections mechanism with Numeric Comparison	[5]
It is strongly recommended to use the methods provided by DA14535 and SDK 6.0.24 (or equivalent ones) for key generation and management: **- To generate the asymmetric keys for firmware image signing** **- To generate the symmetric key for image encryption** **- To generate keys for Bluetooth communication** **- To derive a unique per device key for secure storage**	[8] Section 24.4 [5]
Any published vulnerabilities at the time of product launch must be added in the Risk Table and the risk must be adequately assessed and accepted or mitigated.	
It is strongly recommended that the application firmware implements the logging of security events and generates appropriate user notifications.	
Depending on the actual product, the manufacturer must consider implementing network monitoring and traffic control mechanisms in the application firmware network monitoring and traffic control mechanisms.	

7. Revision History

Revision	Date	Description
1.0	May 9, 2025	Added References to documentation
00.02	May 5, 2025	Reviewed draft
00.01	April 25, 2025	Internal draft for review

STATUS DEFINITIONS

Status	Definition
DRAFT	The content of this document is under review and subject to formal approval, which may result in modifications or additions.
APPROVED or unmarked	The content of this document has been approved for publication.

ROHS COMPLIANCE

Renesas Electronics' suppliers certify that its products are in compliance with the requirements of Directive 2011/65/EU of the European Parliament on the restriction of the use of certain hazardous substances in electrical and electronic equipment. RoHS certificates from our suppliers are available on request.

IMPORTANT NOTICE AND DISCLAIMER

RENESAS ELECTRONICS CORPORATION AND ITS SUBSIDIARIES (“RENESAS”) PROVIDES TECHNICAL SPECIFICATIONS AND RELIABILITY DATA (INCLUDING DATASHEETS), DESIGN RESOURCES (INCLUDING REFERENCE DESIGNS), APPLICATION OR OTHER DESIGN ADVICE, WEB TOOLS, SAFETY INFORMATION, AND OTHER RESOURCES “AS IS” AND WITH ALL FAULTS, AND DISCLAIMS ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING, WITHOUT LIMITATION, ANY IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR NON-INFRINGEMENT OF THIRD-PARTY INTELLECTUAL PROPERTY RIGHTS.

These resources are intended for developers who are designing with Renesas products. You are solely responsible for (1) selecting the appropriate products for your application, (2) designing, validating, and testing your application, and (3) ensuring your application meets applicable standards, and any other safety, security, or other requirements. These resources are subject to change without notice. Renesas grants you permission to use these resources only to develop an application that uses Renesas products. Other reproduction or use of these resources is strictly prohibited. No license is granted to any other Renesas intellectual property or to any third-party intellectual property. Renesas disclaims responsibility for, and you will fully indemnify Renesas and its representatives against, any claims, damages, costs, losses, or liabilities arising from your use of these resources. Renesas' products are provided only subject to Renesas' Terms and Conditions of Sale or other applicable terms agreed to in writing. No use of any Renesas resources expands or otherwise alters any applicable warranties or warranty disclaimers for these products.

(Disclaimer Rev.1.1 Jan 2024)

Corporate Headquarters

TOYOSU FORESIA, 3-2-24 Toyosu
Koto-ku, Tokyo 135-0061, Japan

www.renesas.com

Trademarks

Renesas and the Renesas logo are trademarks of Renesas Electronics Corporation. All trademarks and registered trademarks are the property of their respective owners.

Contact Information

For further information on a product, technology, the most up-to-date version of a document, or your nearest sales office, please visit www.renesas.com/contact-us/