Whitepaper

Introduction to Z-Wave SmartStart

TABLE OF CONTENTS

Summary

This whitepaper provides an overview of the Z-Wave SmartStart feature and the key user scenarios. When deploying smart end devices user interaction is often restricted to extremely rudimentary interfaces, such as buttons or switches. The gateway typically presents a more user-friendly interface, through a web browser or a smartphone app.

Z-Wave SmartStart aims to shift the tasks related to inclusion of an end device into a Z-Wave network away from the end device itself, and towards the more user-friendly interface of the gateway.

Z-Wave SmartStart removes the need for initiating the end device to start inclusion. Inclusion is initiated automatically on power-ON, and repeated at dynamic intervals for as long as the device is not included into a Z-Wave network. As the new device announces itself on power-ON, the protocol will provide notifications, and the gateway can initiate the inclusion process in the background, without the need for user interaction or any interruption of normal operation. This improvement also removes the possibility of other devices being included, as the SmartStart inclusion process only includes authenticated devices.

By moving the device authentication process into the manufacturing and distribution phase or service provider domain, the end user is no longer required to do anything but to power on the devices. This enables a simplified user experience where the device is genuinely ready to use, right out of the box . The device manufacturer or service provider can now prepare inclusion prior to the devices ending up at the end user's house.

Building on the elements introduced by S2 security, the Z-Wave SmartStart is not only easy for the end user, but also secure. Z-Wave SmartStart uses the same device specific keys (DSK) that form the foundation of the secure inclusion process of S2. Only authorized and intended devices are included in the Z-Wave network. Z-Wave SmartStart is based on the embedded SDK 6.8x and related gateway software components.

This whitepaper is limited in scope to the exchange of the DSK between end device and gateway, and how this process relates to Z-Wave SmartStart. For more information regarding the Z-Wave Security Ecosystem, more details on the key exchange during the inclusion process, and implementation requirements of S2 security, refer to the whitepapers on that topic [1] and ZTS for additional technical information.

Abbreviations and Terminology

Controller – Z-Wave terminology for a device that can manage the Z-Wave mesh network. A Z-Wave network can have one or more controllers. A controller can be a simple device such as a remote control capable only of specific dedicated commands. A controller connecting to other network types or services is defined as a gateway.

Gateway – A controller that has one or more additional network interfaces. A gateway allows Z-Wave networks to be operated from other networks, such as a smartphone at home, or through the internet.

DSK – Device Specific Key used in the Z-Wave S2 security scheme.

Inclusion – Z-Wave networks consist of nodes connected in a mesh topology. Inclusion refers to the process of adding a new node to an existing network. The process consists of a sequence of required steps that integrate the new node in the mesh, and not simply creating a link between the node and the controller. Several processes exist to cover multiple scenarios, such as classic inclusion, S2 inclusion, and SmartStart inclusion. Unless otherwise noted, this document uses inclusion to refer to the entire process in a SmartStart context.

Mesh Network – Network topology where each node can communicate with all other nodes in range directly. Z-Wave mesh network allows a device to send a command from one end device to another with up to four routing hops through the mesh, if the destination is not in direct range.

OOB – Out Of Band, such as visual identification and manual entry of a key compared to sending all data through the same radio channel.

OTA – Over The Air, used for wireless transfer of firmware images for updating a device.

QR Code – 2D barcode format that can contain large amounts of information in a small square of encoded blocks resembling a random checkerboard pattern. In Z-Wave, it is used to represent the S2 public part of the DSK on a device, as well as additional information needed for the inclusion process.

SDK – Software Development Kit. In the Z-Wave context, it is available either as embedded/end device SDK or controller/gateway SDK for the various product types respectively.

S2 – Security 2, Z-Wave's unique security model ensuring secure inclusion and secure communication in the Z-Wave network.

ZIPGW – Z-Wave for IP Gateway. Middleware component that maps a Z-Wave network to an IP context and make nodes available through an IP interface.

ZTS – Z-Wave Technical Support portal. The portal contains documentation and other resources available to Z-Wave developers. https://zts.sigmadesigns.com.

Building on Z-Wave S2 Security

The Z-Wave Security Ecosystem consists of several elements to secure the Z-Wave network. Out Of Band (OOB) device authentication is used to remove vulnerabilities in the network integrity while including new devices, ensuring that a new device is authenticated by providing the gateway with the unique DSK matching the new device.

The DSK of the device is used only during inclusion, where the device is granted one or more of network keys by the gateway. These keys are used to encrypt the communication, and only shared with authenticated devices.
This allows for segmentation of safety critical devices in the "S2 Access Control" class and sensors in the "S2 Authenticated" class, while the most constrained devices without authentication support are only allowed access to the "S2 Unauthenticated" class. The Access Control group is reserved for devices directly related to

access control, such as door locks. The extra group-level security used here ensures that devices not involved in the control of access control devices can never acquire the keys required to send commands to open a door lock. Additional groups exist for legacy devices, ensuring that the S2 group keys are only used by S2 included devices.

Z-Wave SmartStart utilizes the DSK of an end device to enhance the inclusion process, making it more user friendly.

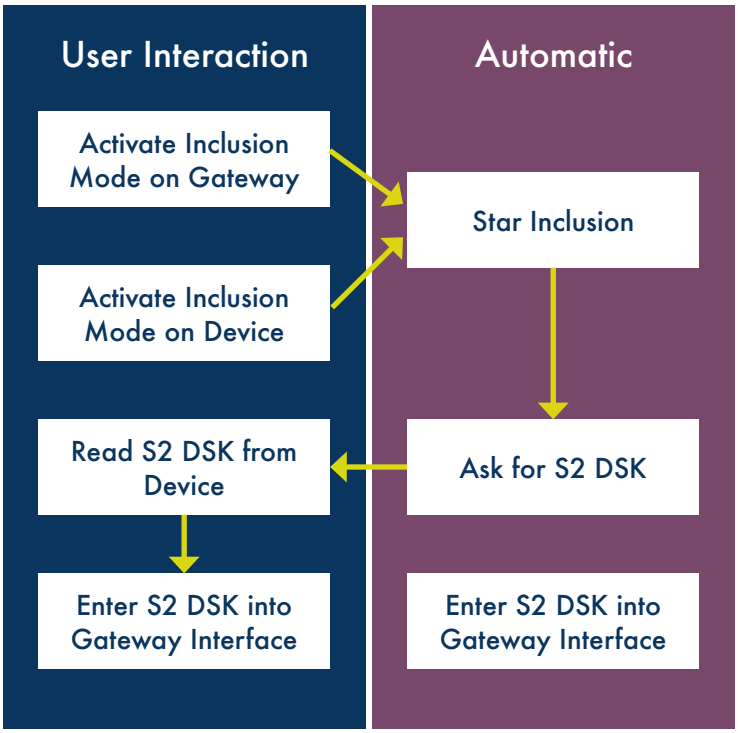

Figure 1: S2 Inclusion Flow without SmartStart

Z-Wave SmartStart under the Hood

This chapter further explores the technical elements of how Z-Wave SmartStart works. It will introduce the main concepts and describe examples of how elements can be used together, to create improvements to the user experience.

Improved Inclusion Process

Z-Wave SmartStart improves the inclusion process introduced in the S2 security ecosystem for Z-Wave [Figure 1]. S2 security already provides the building blocks for a gateway that can skip the processes of manual reading and entry of the DSK in the gateway. That process can now be achieved by scanning the label on

the end device that contains the DSK, and transfer that data to the gateway during the process of adding a new device to the mesh. Prior to Z-Wave SmartStart, this process required the end user to initiate the inclusion process on both the end device and gateway.

A SmartStart inclusion request is automatically sent by the SmartStart device on power ON. This allows the gateway to confirm the identity of the end device by matching the inclusion request with the DSK obtained OOB from the QR code. Upon this recognition, the gateway then automatically includes the authenticated devices.

The SmartStart inclusion process [Figure 2] enables DSK provisioning at any time prior to the inclusion of the new device. The end device interaction is removed from the process. The gateway typically features a much more usable interface, so the user experience of the inclusion process is significantly improved.

The use of a QR code for obtaining the DSK from the device also makes the inclusion process more flexible. The DSK can be read without the device powering ON, so it is possible to prepare the gateway to include a device prior to installing and powering up the end device.

If an end device is factory reset, it is reverted back to the Z-Wave SmartStart inclusion state. If the gateway still has the DSK, it can automatically include the device again. Depending on the implementation in the gateway, the end device may even maintain the configuration and any setup done by the end user.

If an end device or gateway is to be used with older Z-Wave devices, they will work as well. A SmartStart capable end device still supports the classic inclusion method, and a SmartStart gateway can include classic devices as well.

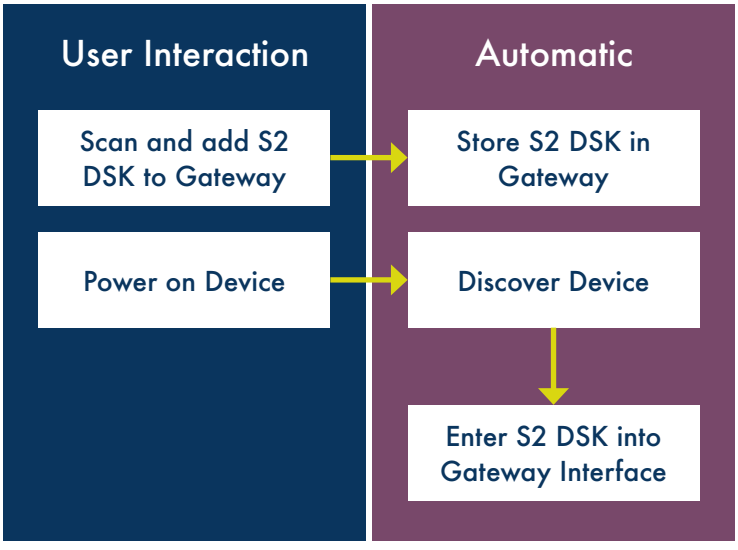

Figure 2: SmartStart Inclusion Flow

QR Data Structure

The QR format introduced with S2 features additional data, which enables the optimal user experience of SmartStart in gateways. This data allows gateways to provide the user with information prior to inclusion. In addition to the unique DSK used for authentication, the QR code format also contains product information, including manufacturer and product ID. The extra data also contains information that let the gateway determine if the device supports SmartStart.

Introducing the Provisioning List

Z-Wave SmartStart is designed to facilitate the inclusion of multiple devices simultaneously, enabled by the Provisioning List, and by storing unique end device DSKs in the gateway. By simply scanning the end device QR code before shipment to the end user, the Provisioning List is created in the gateway. This ensures that the gateway is ready to include the end device out of the box as soon as it is powered on, eliminating the need for any further user interaction on the end device. The order in which the devices are powered is irrelevant; the gateway will include them as long as they are on the Provisioning List.

Service providers can enhance their backend by allowing remote provisioning of devices at a customers' gateway as it is shipped off from the distribution center. It is also possible to store the provisioning list directly into the memory of the gateway, during production or packaging. In this way, the devices are automatically included when the end user powers them ON, and the service provider saves a technician visit or a potential support call.

Example – Ad-Hoc End User Network Expansion

In this example, an end user has purchased a light bulb end device from a retail or e-tail outlet. The scenario assumes that the user has a smartphone app serving as the gateway user interface, allowing both daily operation and administrative setup.

When unpacking the light bulb, the user accesses the gateway through the smartphone app, and finds the administrative menu to add a new device. With Z-Wave SmartStart, the first suggestion is to look for a QR code on the end device and scan it using the smartphone's camera. Upon obtaining the DSK of the light bulb, the smartphone app informs the user that the device is ready for inclusion, and that the next step is to power it ON. The smartphone app then notifies the user that the device has been included in the network and is ready for use [Figure 2].

Example – Service Provider

In this example, a service provider is offering their customers new end devices for purchase through their web site. The customers have a Z-Wave gateway in their home, and the service provider has remote access to the gateway. Call center technicians can assist customers with configuration and setup. The web shop sells end devices from several different manufacturers, and they all support Z-Wave SmartStart.

When the customer orders end devices, the order is queued at the service provider's distribution center. Each end device in the order is picked from the storage shelves and then scanned, so that the device's DSK is associated with the order. When the order is registered as fulfilled and shipped to the customer, the registered DSKs are transferred to the customer's gateway, and added to the provisioning list.

When the customer receives the package, and installs the devices, the devices automatically appear in the gateway user interface and are ready to be used. Users can even receive assistance from the gateway user interface that helps them quickly add the new devices to existing scenes, if relevant.

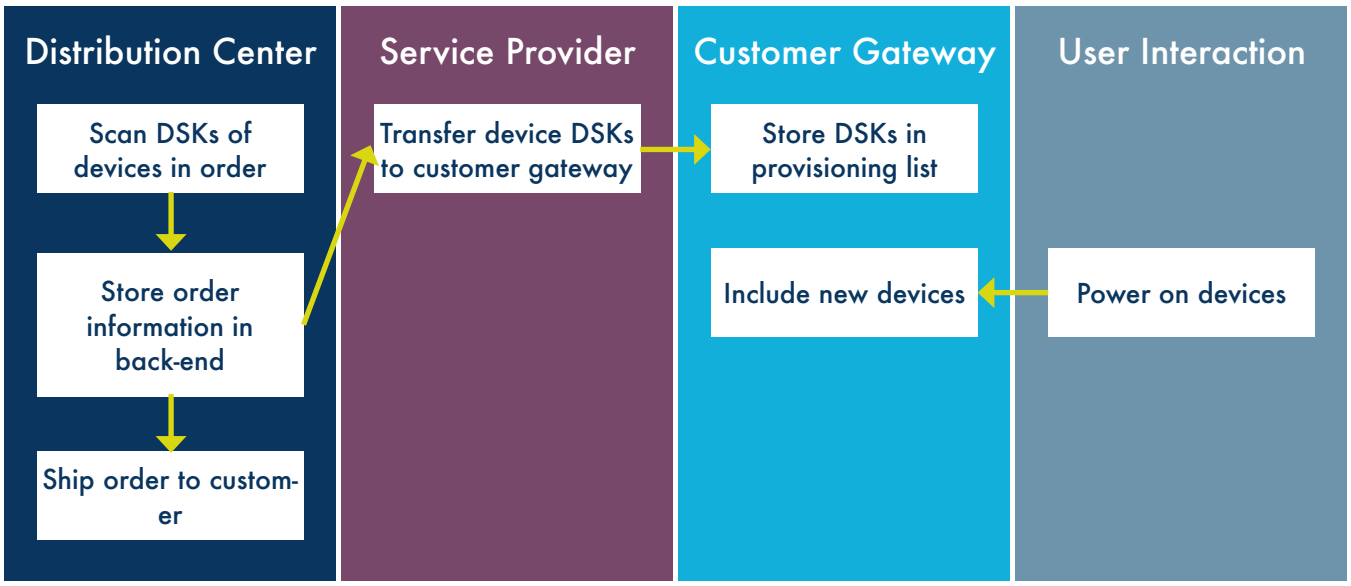

Figure 3: Service provider S2 DSK flow to provisioning list

Example – Removing SmartStart devices from a Z-Wave network

Because SmartStart always includes end devices securely, it is possible to do remote reset of an end device when it is removed from the provisioning list. The gateway can automatically perform these steps. If a device is reset without being removed from the gateway's provisioning list, it will automatically be included again the next time it is powered on.

High Level Overview of Z-Wave SmartStart Implementation

This chapter gives a high level overview for developers that can be used for initial implementation planning, and describes the requirements to implement Z-Wave SmartStart for different types of products. The document focus on new device development, although the same guidelines apply for porting devices to the latest application framework to enable SmartStart.

Developers that are creating a portfolio of devices may require the implementation of different Z-Wave devices types; for example, both end devices and a gateway. This chapter is not intended as an implementation guide or for certification requirements. Those are covered in the relevant documentation on ZTS.

End Device

For new end devices, the application framework already provides most of the S2 and SmartStart work. This is done by implementing end devices as S2 Authenticated or S2 Access Control devices, complete with DSK, and manufactured with the QR code printed on the device. Best practices also recommend that the QR code is also placed on the outside of the product box, allowing distribution centers to provision the devices to a customer without anyone having to open the boxes. This ensures that products arrive at the end user without any box content missing, or accidental drainage of battery.

Migrating to SDK 6.8x (the embedded SDK version supporting Z-Wave SmartStart) from an S2 enabled device is a simple process. There are no new command classes to implement. Some minor modifications to the inclusion process make the device transmit SmartStart inclusion requests when it is not yet part of a network. It is important that product developers maintain the functionality to support classic inclusion. This ensures that devices can still be included by controllers that do not support SmartStart. The protocol changes build on top of classic inclusion, and require no additional work from the developer side.

To test an end device's SmartStart functionality without access to a commercial gateway supporting the feature, developers can use the tools provided in the Z-Wave development kit.

Only existing OTA devices manufactured as S2 Authenticated or S2 Access Control devices can be OTA updated to support SmartStart. Other devices will either not have the unique S2 DSK, or will not have the means for the user to read out the DSK. Without a DSK to read out, a device cannot be added to a SmartStart provisioning list.

Gateway

The most important part of implementing SmartStart in a gateway is management of the provisioning list, which is covered by the following functionality:

- Include device: Functionality to include a device on the provisioning list into the network.
- Manage devices on the provisioning list: Functionality to add/remove/query devices on the provisioning list.
- Obtain DSK from a device: Functionality to scan, enter, or otherwise transport the DSK from a device to the provisioning list.
- Classic inclusion: A method for a user to enter classic inclusion mode on the gateway, for devices not supporting SmartStart.

Best practices recommend the implementation of each component independently, for easier planning, testing, and future maintenance with updated functionality.

Include Devices

This component handles inclusion requests from SmartStart capable devices. The device's DSK is cross-referenced against the provisioning list and the device is included if authenticated.

For gateways based on the ZIPGW middleware, this process is handled automatically. For gateways not based on ZIPGW middleware, functionality that handles inclusion requests, and initiates the appropriate inclusion protocol commands, must be implemented.

This component also initiates update of the gateway UI to make the new end device available to the end user.

Manage Devices on the Provisioning List

This component allows manipulation of the provisioning list, making it possible to add and remove devices. Best practices recommend that the gateway user interface should show the list and status of devices. The user can use this to confirm if a specific device is on the list and if it is currently included in the network.

For gateways based on the ZIPGW the provisioning list is contained in ZIPGW, and API calls for manipulating the list are available. An implementation of a user interface must allow the user to add or remove devices. For gateways not based on ZIPGW the provisioning list storage as well as functions to manipulate the list must be included in the implementation.

An important thing to remember is to remove devices from the provisioning list if they are excluded manually by classic exclusion. Removing the device from the provisioning list is important if the device is to be used in another nearby network, as it may be included again automatically in the first network if it is still in the provisioning list of the first gateway.

Obtain DSK from Device

This component can have a lot of variance, depending on the architecture of the gateway product and the end devices. All products will have QR codes with the DSK available, while some products may offer alternative means of obtaining the key. For simplicity, this document is focused on generic products with QR codes.

There are two main ways to obtain the DSK of a device:

- Recommended: Utilize a smartphone camera to scan and decode the QR code, particularly if the gateway already has a smartphone app that can be used as an administrative interface, showing the included end device on the gateway provisioning list
- Alternative: The user can enter the DSK in an input field using a keyboard or keypad.
  Best practices recommend implementation of both if possible, to give end users an alternative if a QR code will not scan, e.g. because it was damaged.

Classic Inclusion

For interoperability, the gateway must maintain the capability to include devices using classic inclusion. Best practices recommend implementing user assistance in the user interface if a device does not have a QR code to scan or a DSK to enter. When using classic inclusion the user interface should inform the user to initiate inclusion on the end device.

Service Provider Backend

This section assumes a gateway implemented with Z-Wave SmartStart, as well as connection to the service provider backend with means to control the Z-Wave gateways remotely.

In order to provide a good service to end users, the service provider backend must have access to the gateway provisioning list, and information regarding the devices shipped to an end user.
Once the backend process is in place for obtaining the DSKs of end devices shipped to a customer, they need to be transferred to the gateway. When using ZIPGW, API calls for adding DSKs to the provisioning list are provided for the backend to use. When using a custom middleware implementation, the method will depend on the middleware implementation.

It is important to ensure that the gateway user interface has the means to manually import DSKs to the provisioning list. This ensures an alternative method for adding devices, in the event that something has gone wrong in the backend process. It will also support scenarios where a technician has brought a replacement device along or the user has purchased a new device from a local home improvement store.

Starter Kit Controller

The purpose of the starter kit controller is to provide network management functionality for a Z-Wave starter kit with end devices. It is capable of setting up a network of the devices in the kit, and configuring basic operating functions, such as turning devices ON/OFF, by pressing buttons.

A starter kit controller can come in a number of different architectural designs that are treated differently in SmartStart implementation. If the controller is based on ZIPGW it mainly follows the guidelines for the gateway. It is still a best practice to maintain as much generic functionality as possible to make the device more useful. Please refer to certification guidelines for controllers for more information [2].

For starter kit controllers on embedded platforms that cannot use the ZIPGW middleware, the implementation requires a few more steps. A general prerequisite is that the developer has already implemented and certified the controller as S2 compliant. Programming the provisioning information during production ensures that the simple controller is not dependent on an online service to obtain the DSKs of the included devices before it can construct the network. In the future this implementation will be supported in the SDK allowing for fast an easy product creation based on sample code.

For Existing Products - Start with implementing S2 security

SmartStart requires S2 security. For all existing devices, the first step is to port the application to the latest application framework and implement support for S2 Authenticated and/or S2 Access Control. Because Z-Wave SmartStart relies heavily on the availability of the unique device specific keys (DSKs) introduced with S2 security, the unauthenticated class is not supporting SmartStart. Best practice is to certify products on the respective S2 SDK releases prior to starting SmartStart implementation. This ensures that any implementation related to S2 is already completed, and implementation of SmartStart will be faster.

More Information

This white paper covers the high level scenarios of Z-Wave SmartStart. For more information please refer to the Z-Wave developer portal ZTS https://zts.sigmadesigns.com.
The developer portal also contains more information regarding the implementation of Z-Wave SmartStart in new and existing products, as well as sample implementations that can be used for reference.

References

[1] Introduction to the Z-Wave Security Ecosystem http://z-wave.sigmadesigns.com/wp-content/uploads/2016/08/Z-Wave-Security-White-Paper.pdf

[2] Certification guidelines on the certification portal: http://z-wavecertification.sigmadesigns.com, for registered partners.