Application Report
SWRA513B–February 2017–Revised January 2020

CC3x20, CC3x35 SimpleLink™ Wi-Fi® Internet-on-a chip™ Solution Device Provisioning

ABSTRACT

The CC3120/CC3135 and CC3220/CC3235x devices are part of the SimpleLink™ microcontroller (MCU) platform, which consists of Wi-Fi®, *Bluetooth*® low energy, Sub-1 GHz and host MCUs, which all share a common, easy-to-use development environment with a single core software development kit (SDK) and rich tool set. A one-time integration of the SimpleLink™ platform enables you to add any combination of the portfolio's devices into your design, allowing 100 percent code reuse when your design requirements change. For more information, visit www.ti.com/simplelink.

This document describes the provisioning process, which provides the SimpleLink™ Wi-Fi® device with the information (network name, password, and so forth) needed to connect to a wireless network.

Contents

Trademarks

1 Introduction

The provisioning process provides the SimpleLink™ Wi-Fi® device with the information (network name, password, and more) needed to connect to a wireless network. The process usually occurs only once, letting end users connect their devices to their local networks for the first time. Providing this information may become challenging because all Internet of Things (IoT) devices are not equipped with conventional input peripherals such as keyboards or touch screens.

The CC3120/CC3135 and CC3220/CC3235x SimpleLink™ Wi-Fi® Internet-on-a chip™ solution from Texas Instruments™ offers smart and fast built-in Wi-Fi® provisioning capabilities with security options, which allows end users to wirelessly configure their IoT devices, using a smartphone or tablet running a dedicated provisioning app. The provisioning capabilities are provided as a comprehensive end-to-end solution, and easily embedded by developers on their own wireless applications.

The solution offers several built-in provisioning modes as follows:

- Access point provisioning, a provisioning method in which the device creates a wireless network of its own, allowing a PC or smartphone to connect to it directly, and provide its initial configuration.
- SmartConfig™ technology provisioning, a proprietary provisioning method from TI, that uses a smart phone or tablet to broadcast the network credentials to the unprovisioned device.
- WPS provisioning

Because the provisioning logic is fully integrated inside the SimpleLink™ Wi-Fi® device, developers can easily use it in their embedded wireless applications without any prior provisioning knowledge.

The CC3135 and CC3235x devices are dual-band, Wi-Fi® 2.4 GHz and 5 GHz. The provisioning process in these devices supports both 2.4-GHz and 5-GHz channels.

NOTE: The provisioning process may take longer when using the 5-GHz configuration, due to longer scan cycles.

1.1 Terminology

Table 1 lists the acronyms used in this document.

Table 1. Acronyms

Acronym	Description
AES	Advanced Encryption Standard
AP	Access point
APSC	Access point SmartConfig
HTTP	Hyper Text Transfer Protocol
HTTPS	Hyper Text Transfer Protocol Secure
REST	Representational State Transfer
SC	SmartConfig
SSID	Service set identifier
STA	Station
UUID	Universally unique identifier
WAC	Wireless accessory configuration
WLAN	Wireless local area network

2 Overview

2.1 General Description

The SimpleLink™ Wi-Fi® provisioning process is divided into the Configuration stage and the Confirmation stage. The process begins with the configuration stage. During this stage, the SimpleLink™ Wi-Fi® device waits for the end user to provide the information needed to connect to the wireless network. The user can use an external device such as a smartphone or tablet running a dedicated provisioning app provided by TI to configure the following parameters:

- Network name (SSID)
- Password
- Device name (optional)
- UUID (optional)

The device saves the provided network information into its serial flash memory as a new profile. The profile is kept encrypted as a system file and only the networking subsystem can access the password to this network. Once a profile is successfully configured, the device moves to the confirmation stage. The confirmation stage tests the profile that was configured during the configuration stage. During the confirmation stage, the device tries to connect to the wireless network found in the new configured profile. If the connection is successful, the device also tries to provide feedback about the successful connection to the user's smart phone provisioning app, which configured the profile. A connection is defined as successful if the WLAN connection is established, and an IP address is acquired.

If the connection is successful, and the feedback is delivered to the user, the confirmation stage is successful, and the provisioning process successfully ends. If the connection attempt fails, or if it is successful but the feedback is not delivered to the user, the confirmation stage fails, and the device moves back to the configuration stage. At this point, the user's smart phone provisioning app can ask the device to send the fail reason of the previous confirmation attempt, and configure a new profile. In this case, the configured profile is not deleted.

Possible reasons for confirmation failure:

- SSID is not found during the scan.
- SSID is found, but the WLAN connection is not successfully established.
- WLAN connection is successfully established, but the IP address is not acquired.

If feedback about a successful connection is not delivered to the user during the confirmation stage, but the user asked for the confirmation result during the following configuration stage, a successful result is sent, and the provisioning process successfully ends.

Figure 1 shows the provisioning process flow.

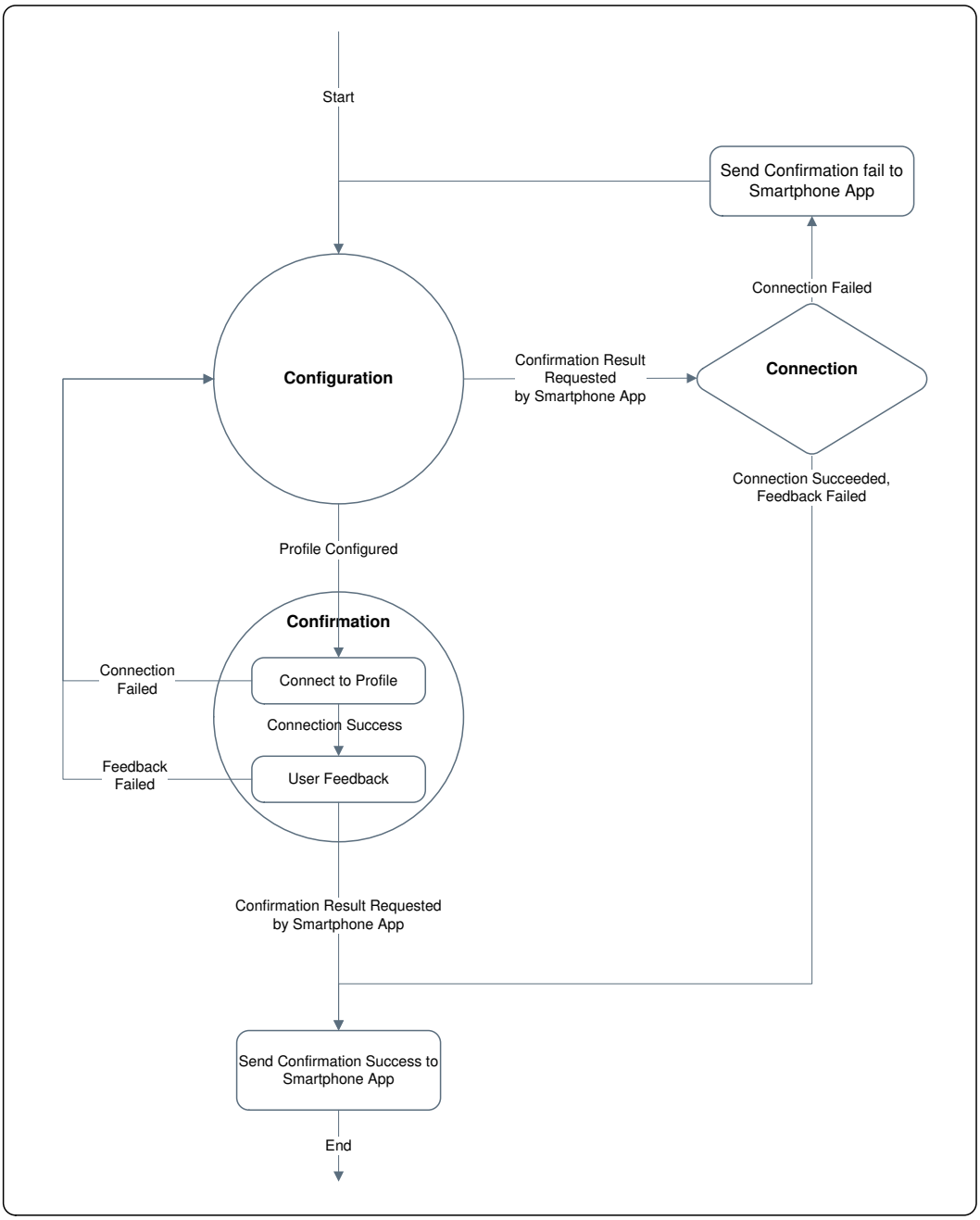

Figure 1. Provisioning Process Flow Chart

The networking subsystem internally executes all provisioning activities (adding new profiles, testing new profiles, reporting results to the user, among others). Switching between the provisioning stages and device roles (AP or STA) is also done internally. The networking subsystem constantly sends the host updates regarding the progress of the provisioning process. The host only starts the provisioning process; once started, no further actions are needed.

Figure 2 shows the host application provisioning flow.

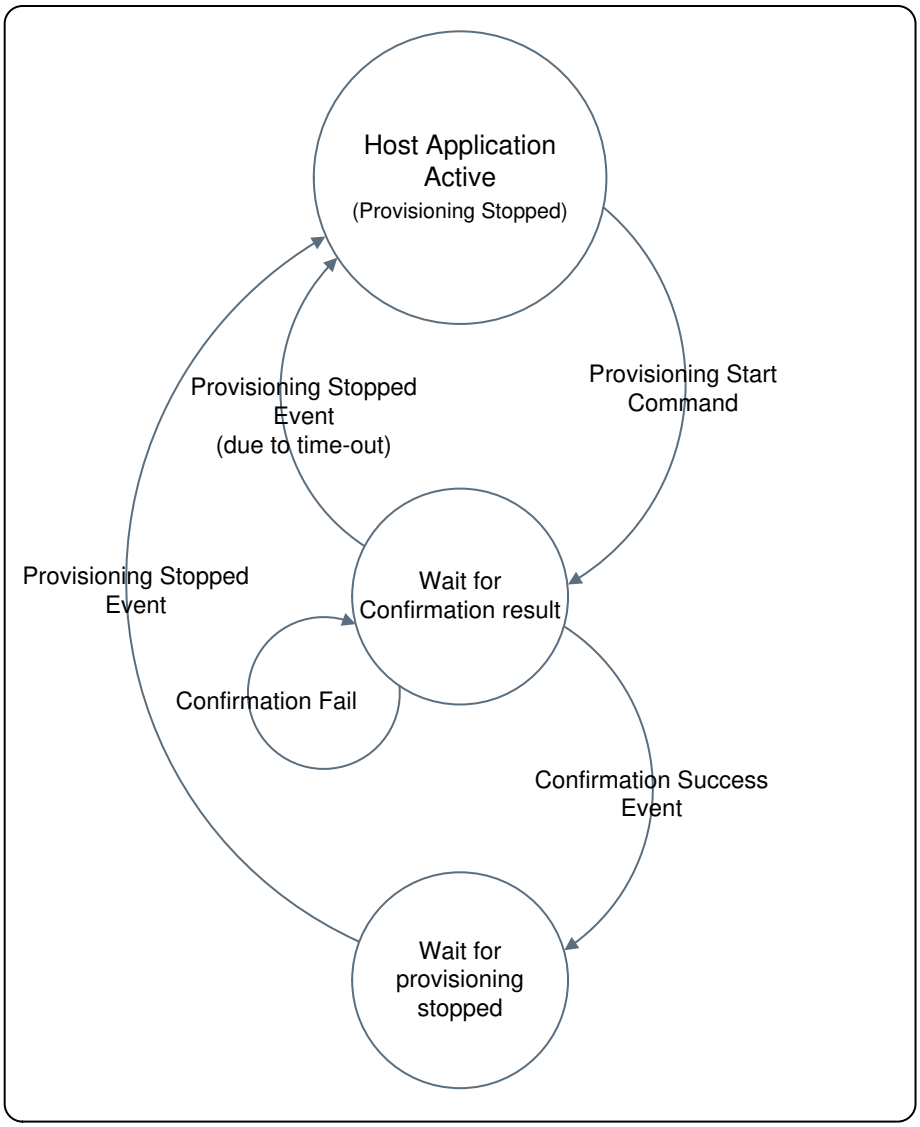

Figure 2. Host Application Provisioning Flow

Once a provisioning process starts, the host should wait for the networking subsystem to send it the confirmation result. During this time, the host cannot perform any networking activities that may interrupt the ongoing provisioning process. If the confirmation result received is success, the provisioning process automatically stops, and the host should wait for the provisioning stopped event. If the confirmation failed, the host is notified of the fail reason, and the provisioning process continues for another configuration attempt.

3 Provisioning Configuration Modes

The provisioning configuration stage can start in several different modes:

- AP provisioning
- SmartConfig™ provisioning
- AP plus SmartConfig™ provisioning
- AP plus SmartConfig™ plus External configuration provisioning

3.1 AP Provisioning

The SimpleLink™ Wi-Fi® device is in AP role, using a predefined network name and security parameters. Users can connect with an external device (such as a smartphone) to the SimpleLink™ AP, and add a profile through the HTTP/HTTPS server of the device. This configuration mode can also deliver the confirmation result to users if it was not successfully delivered during the confirmation stage.

NOTE: If the AP is set to be secured, the password must be unique per device. A possible security breach can be exposed if all new devices are configured by default with the same password.

3.2 SmartConfig™ Provisioning

SmartConfig™ is a TI proprietary provisioning method that uses a smartphone or tablet to broadcast network credentials to an unprovisioned device. In this mode, the SimpleLink™ Wi-Fi® device is in STA role, scanning for the SmartConfig™ data broadcasts, enabling users to add a profile through a SmartConfig™-capable smart phone application. The SmartConfig™ process supports the delivery of the encrypted network credentials, by using a symmetric key encryption (AES128). The key is applied both in the mobile application and in the SimpleLink™ device.

3.3 AP Plus SmartConfig™ Provisioning

The SimpleLink™ Wi-Fi® device is in AP role, simultaneously scanning for SmartConfig™ broadcasts. Users can either connect with an external device (such as a smartphone) to the SimpleLink™ AP, and add a profile through the HTTP/HTTPS server of the device, or add a profile using a SmartConfig™-capable smartphone application. This configuration mode can also deliver the confirmation result to users if it was not successfully delivered during the confirmation stage.

3.4 AP Plus SmartConfig™ Plus External Configuration Provisioning

In this mode the SimpleLink™ Wi-Fi® device is in AP role, enabling the user to use AP provisioning or SmartConfig™ provisioning (same as APSC mode), or in addition, to use an external configuration method that is managed in the host level (such as WAC).

4 Initiating the Provisioning Process

The provisioning process starts after receiving an explicit request from the host application. Provisioning can also start automatically under certain conditions.

4.1 Host-Initiated Provisioning

When the host initiates the provisioning process, it should provide: the desired configuration mode, the role (AP or STA) the device should switch to if a successful provisioning, and an inactivity time-out value which defines the period of time (in seconds) the system waits before it automatically stops the provisioning process because no user activity is detected.

4.2 Auto-Provisioning

When the auto-provisioning connection policy is enabled, the provisioning process automatically starts in the following cases:

- No saved profiles exist, and 2 seconds have passed since the device started, without receiving any command from the host.
- The device is in STA role, the auto-start connection policy is enabled, the profile list is not empty, and the device was disconnected from the WLAN network for more than 2 minutes.

If the provisioning process is auto-started while in STA role, SC-only configuration mode is used. If the provisioning process is started while in AP role, APSC configuration mode is used. Whenever a provisioning process is auto-started, a proper event notifies the host.

4.3 Ending the Provisioning Process

Once the provisioning process starts, it repeatedly switches between the configuration and confirmation stages until one of the following occurs.

- A configured profile is successfully confirmed.
- The host sends a stop provisioning command.
- There is no user activity for some time (defined by the inactivity timeout parameter).
- The device is reset during the provisioning configuration stage.

When the provisioning process stops due to host request or after inactivity time-out expires, the device switches back to the role that was active when the provisioning process started. If the process stops because a profile was successfully confirmed, the device switches to the role defined by the host during the provisioning start command.

After provisioning successfully stops, the host receives a PROVISIONING_STOPPED status event. The event is sent after switching to the desired role is complete. When the host issues a provisioning stop command, it should wait for the PROVISIONING_STOPPED event before issuing additional commands.

5 Profile Confirmation

Once a new profile is configured to the device, it must be confirmed. A profile is successfully confirmed only if: the WLAN connection was successful, an IP address was successfully acquired, and the user's smartphone provisioning app (that configured the profile) received the successful confirmation result from the device (feedback). After a successful confirmation, the device can either stay connected to the new network (STA role), or switch back to the AP role.

5.1 Feedback

Delivering feedback to the user's smartphone provisioning app during the confirmation stage (if the connection was successful) can be done either over the local wireless network configured to the device, or over the Internet through a cloud-based server.

5.1.1 Local Feedback

When feedback is delivered locally, the user's smartphone provisioning app must connect to the same wireless network to which the SimpleLink™ Wi-Fi® device is connected, and ask the HTTP server of the device for the confirmation result. To do so, the smartphone provisioning app must discover the new IP address the device acquired. After the SimpleLink™ Wi-Fi® device successfully acquires an IP address, it advertises itself using broadcast and multicast packets, to enable the smartphone provisioning app to discover its address.

5.1.2 External Feedback (Cloud)

Delivering feedback to the user's smartphone provisioning app can also be done using an external cloud-based server. When the SimpleLink™ Wi-Fi® device connects to the configured network and acquires an IP address, it tries to contact a cloud-based server. The user's smartphone provisioning app, instead of connecting the HTTP server of the SimpleLink™ Wi-Fi® device over the local wireless network, connects the cloud-based server over the Internet, and asks whether or not the SimpleLink™ Wi-Fi® device is connected to the cloud. In this mode, the smartphone provisioning app does not need to discover the IP address that the device acquired.

The networking subsystem does not internally connect the cloud-based server, the host application does. When the device successfully connects, and acquires an IP address, it notifies the host through an event that it may start trying to connect to the cloud server (for example sending host sockets commands). If confirmation is successful, the host should manually stop the provisioning process, and order the networking subsystem to stay in STA role. If confirmation fails, the host should notify the networking subsystem about the failure, and the networking subsystem should return to the configuration stage as usual (the networking subsystem is unaware of the confirmation results coming from the cloud).

To use cloud-based feedback, the external confirmation bit should be set in the provisioning host command flags parameter.

5.2 Confirmation Fail

If the SimpleLink™ Wi-Fi® device fails to connect to the configured profile, the device switches back to the configuration stage, and the smartphone provisioning app will not find the device on the local wireless network or the cloud. If the current configuration mode requires that the SimpleLink™ Wi-Fi® device is in the AP role, the smartphone provisioning app may try to directly connect to the device, and then ask for the confirmation result (SSID not found, WLAN connection failed, or IP not acquired).

If the SimpleLink™ Wi-Fi® device successfully connects to the configured profile, but the user's smartphone provisioning app failed to collect the confirmation result over the local wireless network or the cloud, the device switches back to the configuration stage. If the current configuration mode requires that the SimpleLink™ Wi-Fi® device is in the AP role, the smartphone app may try to directly connect to the AP of the device, and then ask for the confirmation result. If the smartphone app can connect the device and asks for the confirmation result (which is *connection successful, feedback failed*) the device answers with the *confirmation success* result, because the feedback was eventually successfully delivered to the user (although it was in the configuration stage instead of the confirmation stage), and the provisioning process will successfully end. In addition, the device switches to the role that it was requested to switch to if successful in provisioning. If the device is in STA role, it also automatically connects to the new added profile.

Profiles that were configured during the configuration stage are not deleted in case of a failed confirmation.

6 External Configuration

When the provisioning process starts in APSC plus external configuration mode, the device is ready to serve stations that are:

- Trying to connect it (AP provisioning)
- Ready to handle SmartConfig™ transmissions (SC provisioning)
- Allow the host to manage an external provisioning method (such as WAC)

Unlike other configuration modes, this mode lets the host send commands and receive events from the networking subsystem while provisioning is running. The APIs are unblocked when the EXTERNAL_CONFIGURATION_READY event is sent to the host. The event is sent immediately after the networking subsystem successfully starts the provisioning process.

When the host identifies that the user chooses to use the external configuration method, it should stop the internal running provisioning process, and continue carrying out the external provisioning process.

If the user chooses to use one of the internal provisioning methods (AP or SC provisioning), the device must be restarted before it can continue with the internal provisioning process. At this point, the networking subsystem sends a RESET_REQUEST event to the host. The host should stop its external provisioning process (close opened sockets and so forth), restart the SimpleLink™ Wi-Fi® device, and wait for the internal provisioning process to end as usual.

7 Host APIs

Starting and stopping the provisioning process is done with one host command. When the process starts, it is managed internally by the networking subsystem until it ends. No host application intervention is needed during the process. Information regarding the progress of the provisioning process is reported to the host through the provisioning status event.

7.1 Provisioning Command

The host controls the provisioning process using one command: **sl_WlanProvisioning**

_i16 **sl_WlanProvisioning**(_u8 ProvisioningCmd, _u8 RequestedRoleAfterSuccess, _u16 InactivityTimeoutSec, char *pSmartConfigKey, _u32 Flags);

- **ProvisioningCmd:** Specifies the provisioning configuration method.

  Table 2 lists the values the **ProvisioningCmd** command can have.

Table 2. ProvisioningCmd Values

Command	Value	Action
SL_WLAN_PROVISIONING_CMD_START_MODE_AP	0	Start provisioning in AP configuration mode.
SL_WLAN_PROVISIONING_CMD_START_MODE_SC	1	Start provisioning in SmartConfig configuration mode.
SL_WLAN_PROVISIONING_CMD_START_MODE_APSC	2	Start provisioning in AP plus SmartConfig configuration mode.
SL_WLAN_PROVISIONING_CMD_START_MODE_APSC_EXTERNAL_CONFIGURATION	3	Start provisioning in AP plus SmartConfig configuration mode, and enable the use of external configuration methods.
SL_WLAN_PROVISIONING_CMD_STOP	4	Stop currently running provisioning process.
SL_WLAN_PROVISIONING_CMD_ABORT_EXTERNAL_CONFIRMATION	5	Stop currently running confirmation stage, if external confirmation is used. Device returns to configuration stage.

- **RequestedRoleAfterSucess:** The desired role (AP or STA) to which the device should switch if provisioning is successful (relevant only if the value of the *ProvisioningCmd* is 0, 1, 2, or 3). If the value of the *ProvisioningCmd* command is SL_WLAN_PROVISIONING_CMD_STOP (4), this parameter can be used (by using a value of 0 × FF) to order the device to stay in its current role (instead of switching back to the role that was active when provisioning started, as usually occurs when provisioning is stopped).
- **InactivityTimeoutSec:** Defines the period of time (in seconds) the system waits before it automatically stops the provisioning process when no user activity is detected. Relevant only if the value of the *ProvisioningCmd* command is 0, 1, 2, or 3.
- **Flags:** Optional configuration conducted by a bitmap.

Table 3. Flags

Command	Value	Action
BIT_0	ENABLE_EXTERNAL_CONFIRMATION	Defines whether to use external confirmation or not. Relevant only if the value of the *ProvisioningCmd* command is 0, 1, 2, or 3.

- **pSmartConfigKey:** Symmetric key which is used to decrypt the credentials of the configured network transferred from the mobile application. Use the same key in the mobile application and in the SimpleLink™ device. The key length must be set to 16 characters. The security key must be unique for each product (recommendation: add a label on the product with the unique security key).
- **Return values:** (see Table 4)

Table 4. Return Values

Command	Value	Action
STATUS_OK	0	Command was successfully executed.
SL_ERROR_WLAN_PROVISIONING_ABORT_PROVISIONING_ALREADY_STARTED	–2169	Start provisioning command failed because provisioning process is already running.
SL_ERROR_WLAN_PROVISIONING_ABORT_HTTP_SERVER_DISABLED	–2170	Start provisioning command failed because HTTP server is disabled.
SL_ERROR_WLAN_PROVISIONING_ABORT_PROFILE_LIST_FULL	–2171	Start provisioning command failed because profile list is full.
SL_ERROR_WLAN_PROVISIONING_ABORT_INVALID_PARAM	–2172	Start provisioning command failed because one of the parameters is invalid.
SL_ERROR_WLAN_PROVISIONING_ABORT_GENERAL_ABORT	–2173	Start provisioning command failed because of an unknown reason.
SL_ERROR_WLAN_PROVISIONING_CMD_NOT_EXPECTED	–2177	Provisioning command failed because it was not expected.

7.2 Provisioning Status Event

The provisioning status event contains the following parameters.

- Status
- Role
- WlanStatus
- SsidLen
- Ssid

Table 5 lists the values the Status parameter can have.

Table 5. Status Values

Command	Value	Action
SL_WLAN_PROVISIONING_GENERAL_ERROR	0	The provisioning process encountered an unknown error.
SL_WLAN_PROVISIONING_CONFIRMATION_STATUS_FAIL_NETWORK_NOT_FOUND	1	The profile confirmation failed because the SSID was not found.
SL_WLAN_PROVISIONING_CONFIRMATION_STATUS_FAIL_CONNECTION_FAILED	2	The SSID was found, but the profile confirmation failed because WLAN connection was not successful.
SL_WLAN_PROVISIONING_CONFIRMATION_STATUS_FAIL_CONNECTION_SUCCESS_IP_NOT_ACQUIRED	3	The SSID was found, the WLAN connection was successful, but the profile confirmation failed because an IP address was not successfully acquired.

Table 5. Status Values (continued)

Command	Value	Action
SL_WLAN_PROVISIONING_CONFIRMATION_STATUS_SUCCESS_FEEDBACK_FAILED	4	The SSID was found, the WLAN connection was successful, an IP address was successfully acquired, but the feedback to the user about the successful connection was not successfully delivered.
SL_WLAN_PROVISIONING_CONFIRMATION_STATUS_SUCCESS	5	The SSID was found, the WLAN connection was successful, an IP address was successfully acquired and the feedback to the user about the successful connection was successfully delivered. Confirmation stage ended successfully.
SL_WLAN_PROVISIONING_ERROR_ABORT	6	The provisioning process was not started due to an unknown error.
SL_WLAN_PROVISIONING_ERROR_ABORT_INVALID_PARAM	7	The provisioning process was not started due to an invalid parameter.
SL_WLAN_PROVISIONING_ERROR_ABORT_HTTP_SERVER_DISABLED	8	The provisioning process was not started because the HTTP server is disabled.
SL_WLAN_PROVISIONING_ERROR_ABORT_PROFILE_LIST_FULL	9	The provisioning process was not started because the profile list is full.
SL_WLAN_PROVISIONING_ERROR_ABORT_PROVISIONING_ALREADY_STARTED	10	The provisioning process was not started because it is already running.
SL_WLAN_PROVISIONING_AUTO_STARTED	11	The provisioning process was automatically started by the device.
SL_WLAN_PROVISIONING_STOPPED	12	The provisioning process ended.
SL_WLAN_PROVISIONING_SMART_CONFIG_SYNCED	13	SmartConfig configuration data transmission was discovered by the device. The device starts listening and collecting the profile data.
SL_WLAN_PROVISIONING_SMART_CONFIG_SYNC_TIMEOUT	14	SmartConfig configuration data transmission was discovered by the device, but the device cannot extract the profile data out of it.
SL_WLAN_PROVISIONING_CONFIRMATION_WLAN_CONNECT	15	A WLAN connection was established during a confirmation stage.
SL_WLAN_PROVISIONING_CONFIRMATION_IP_ACQUIRED	16	An IP address was acquired during a confirmation stage.
SL_WLAN_PROVISIONING_EXTERNAL_CONFIGURATION_READY	17	User may start configuring the device using an external confirmation method (relevant only when APSC plus External Configuration mode is used).

If the value of the status parameter is SL_WLAN_PROVISIONING_STOPPED (12), additional information is provided with the following parameters.

- **Role:** The active role (AP/STA) after the provisioning process ends.
- **WlanStatus:** If the active role is STA, this parameter shows the WLAN connection status of the device (0-Disconnected, 1-Scanning, 2-Connecting, 3-Connected) when the provisioning process ends.
- **Ssid, SsidLen:** If WlanStatus is connected, these parameters provide the SSID to which we are connected.

During the provisioning process the device might change its active role and connection status without informing the host application; when the process ends this information is sent to the host to inform it of the current status of the device. These parameters are not relevant in other provisioning status values.

7.3 Provisioning Profile-Added Event

When a profile is added during the configuration stage of provisioning, the SL_WLAN_EVENT_PROVISIONING_PROFILE_ADDED event is sent to the host.

7.4 Reset Request Event

During provisioning, the device might restart itself as part of the process. If reset is required while the host is busy (for example when sockets are open during host external configuration provisioning), instead of performing the restart, the device asks the host to do it. When this event arrives, the host should stop all activities (close all opened sockets), and restart the device.

7.5 Blocking APIs During Provisioning

Because during the provisioning process the device switches between different roles, connects to different APs, and changes its IP addresses, host commands may not be properly served. As a result, when the host issues a command during an active provisioning process, the SL_RET_CODE_PROVISIONING_IN_PROGRESS (–2014) error is returned. Only the *sl_WlanProvisioning* and *sl_stop* commands are allowed. If the host is interested in executing a different command, it must either wait for the provisioning process to end, or manually stop it (using the SL_WLAN_PROVISIONING_CMD_STOP command). In addition, events that may be sent to the host during the provisioning connection attempts (such as NETAPP_IPACQUIRED) are blocked and will not reach the user (except for the provisioning dedicated events, such as the provisioning status event).

In some cases, after provisioning starts the APIs are unblocked to let the host perform some actions that are necessary for completing the provisioning process. These actions follow:

- External Confirmation: when the confirmation result is sent to the user over the Internet, the host must open a socket to the cloud server. To enable it, the APIs are unblocked immediately after the PROVISIONING_CONFIRMATION_IP_ACQUIRED status event is sent to the host.
- External Configuration: when APSC plus External Configuration mode is used, the host must open sockets during the configuration stage. To enable it, the APIs are unblocked immediately after the PROVISIONING_EXTERNAL_CONFIGURATION_READY status event is sent to the host.
- Auto-Provisioning: when provisioning is auto-started, the APIs are still allowed (unlike host-initiated provisioning where the APIs are blocked immediately after the provisioning process has started). APIs are blocked only after user activity is detected (for example, a profile is added).

8 Provisioning Use Examples

8.1 Successful SmartConfig™ Provisioning

Figure 3 shows a sequence diagram describing a successful provisioning process using the SmartConfig™ method.

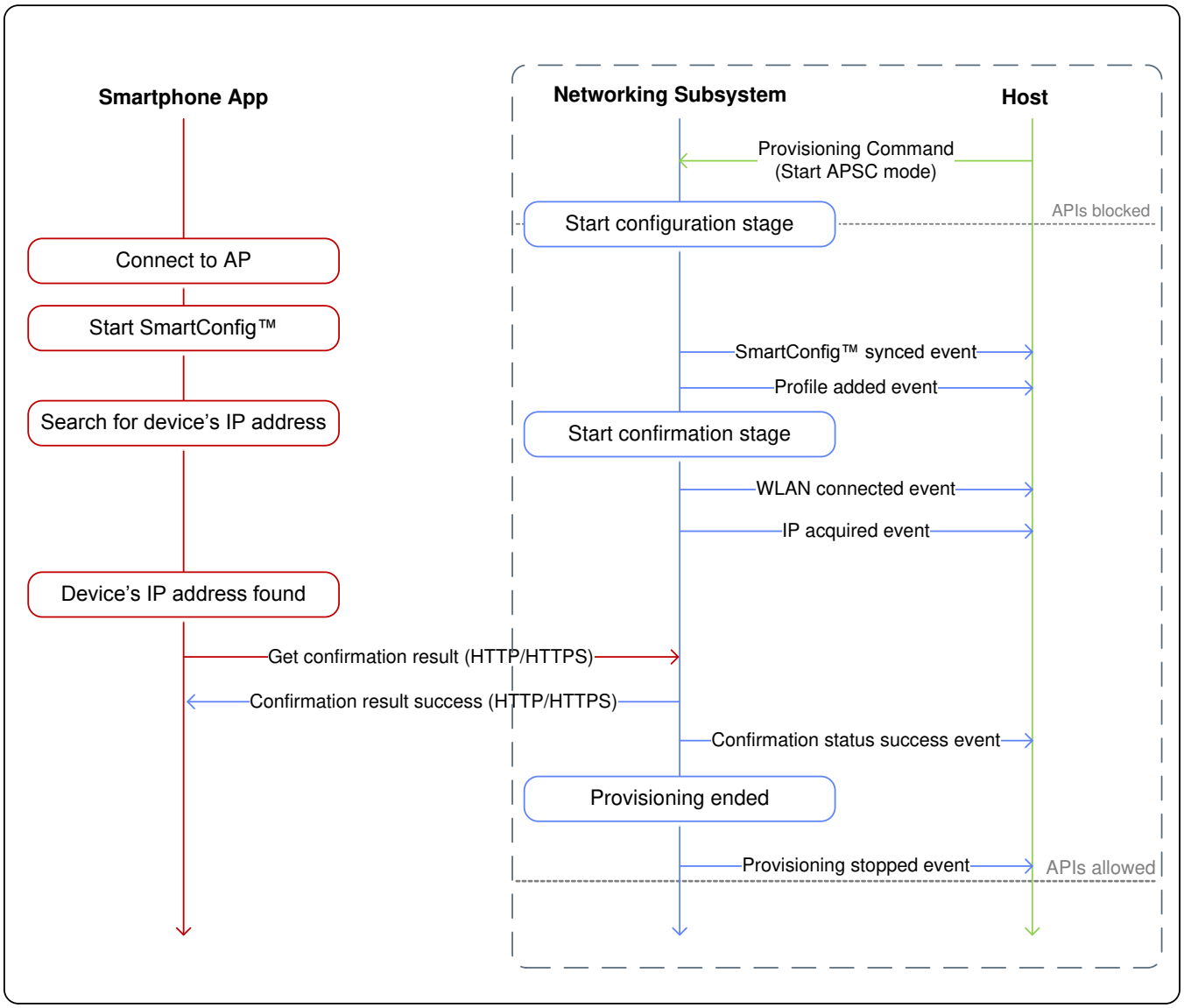

Figure 3. Successful SC Provisioning Example

NOTE: The APIs are blocked during the entire provisioning process.

8.2 Unsuccessful SmartConfig™ Provisioning

Figure 4 shows a sequence diagram describing an unsuccessful provisioning process using the SmartConfig™ method due to IP acquire failure.

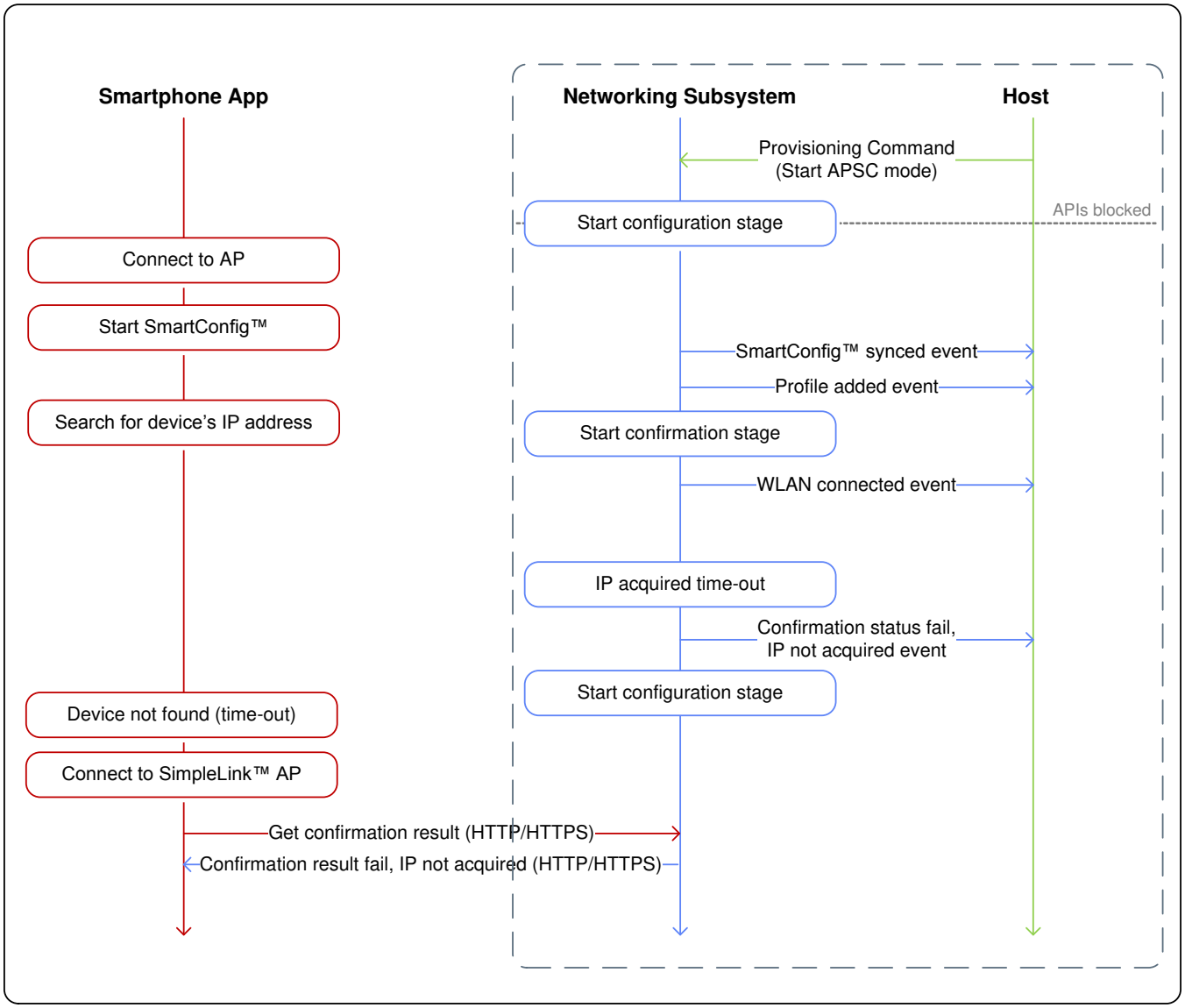

Figure 4. Unsuccessful SC Provisioning Example

8.3 Successful SmartConfig™ Provisioning With AP Fallback

Figure 5 shows a sequence diagram describing a successful provisioning process using the SmartConfig™ method that was completed after fallback to AP.

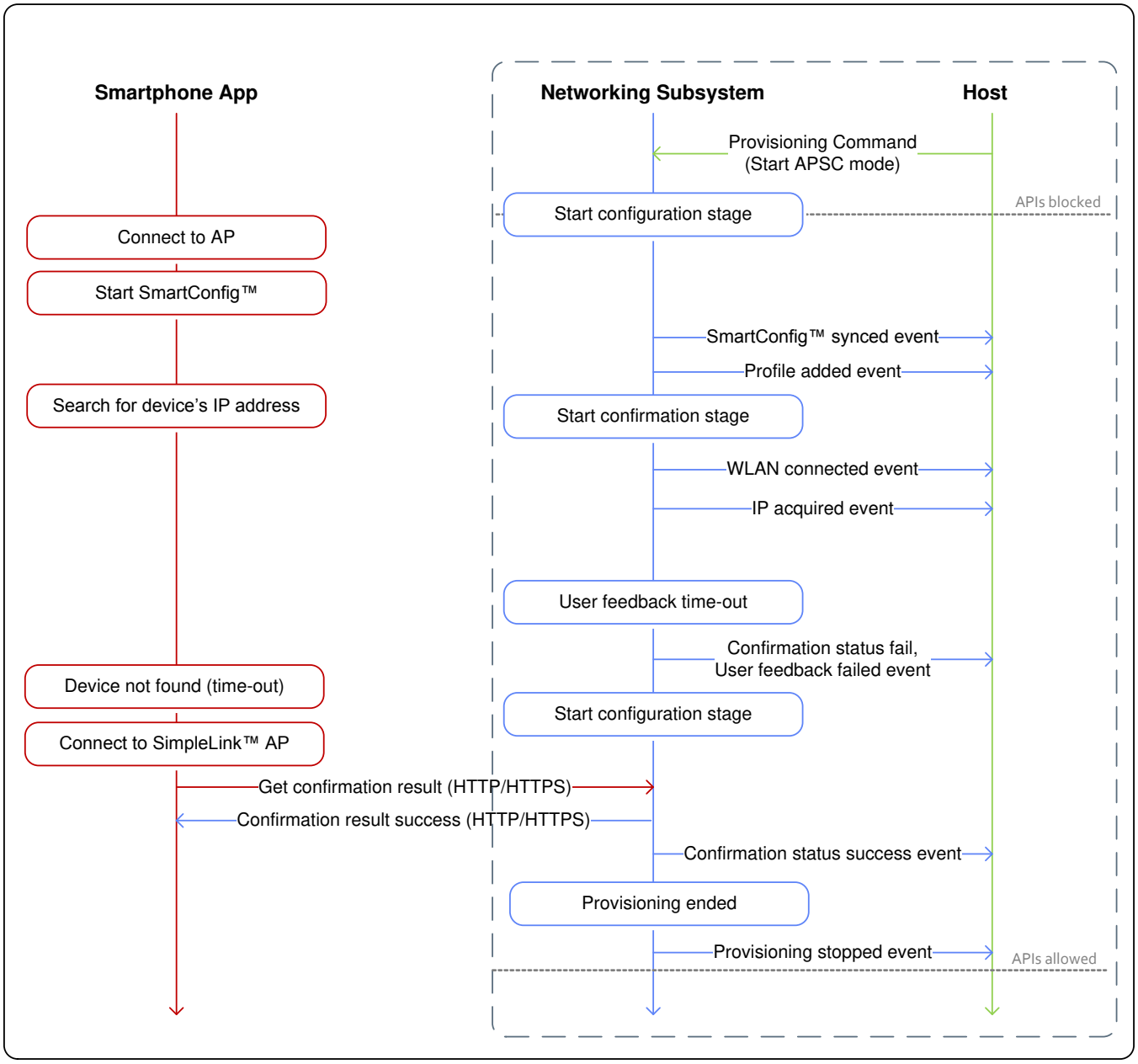

Figure 5. Successful SC Provisioning with AP Fallback Example

8.4 Successful AP Provisioning

Figure 6 shows a sequence diagram describing a successful provisioning process using the AP provisioning method.

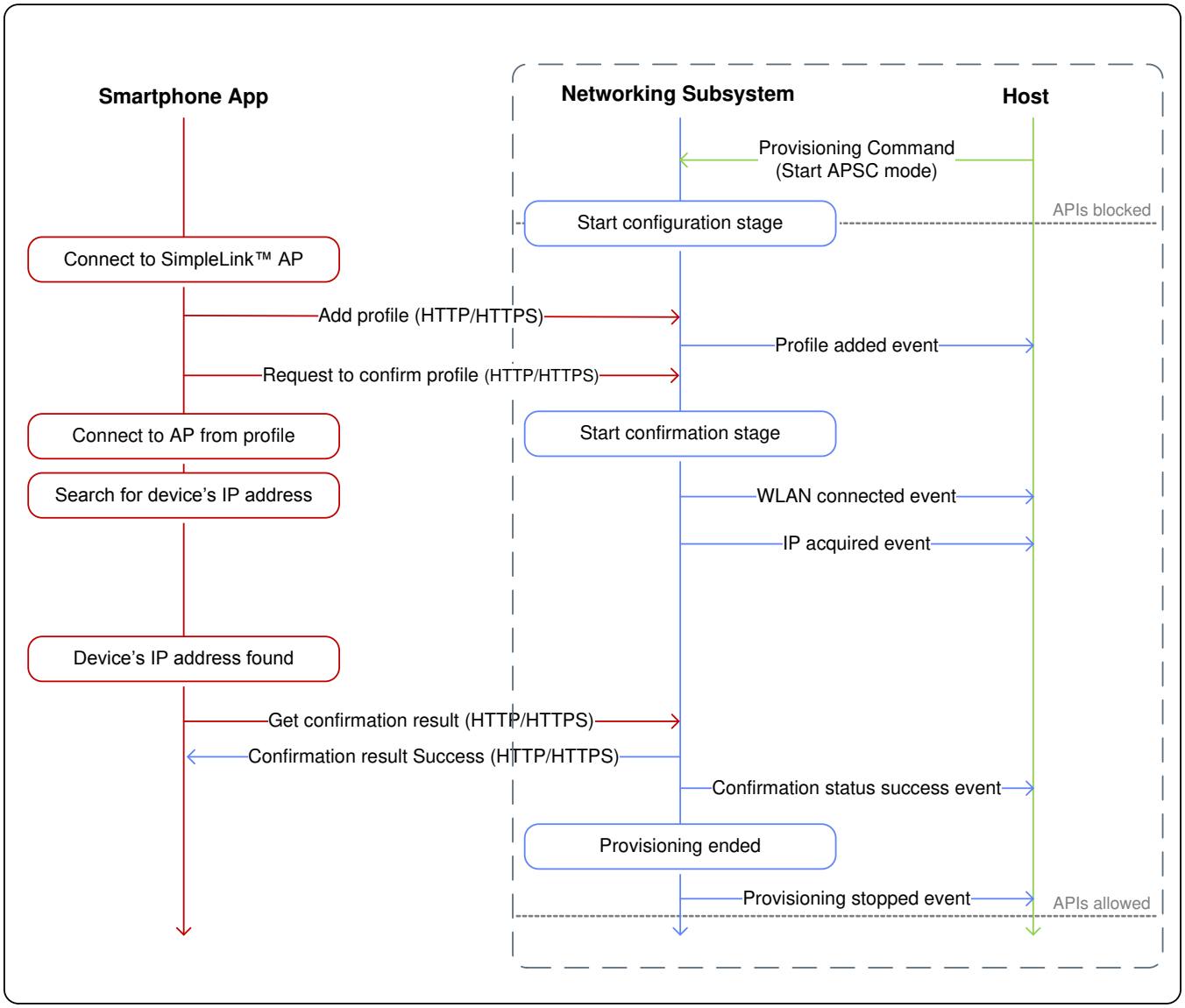

Figure 6. Successful AP Provisioning Example

8.5 Successful AP Provisioning With Cloud Confirmation

Figure 7 shows a sequence diagram describing a successful provisioning process using cloud confirmation AP provisioning.

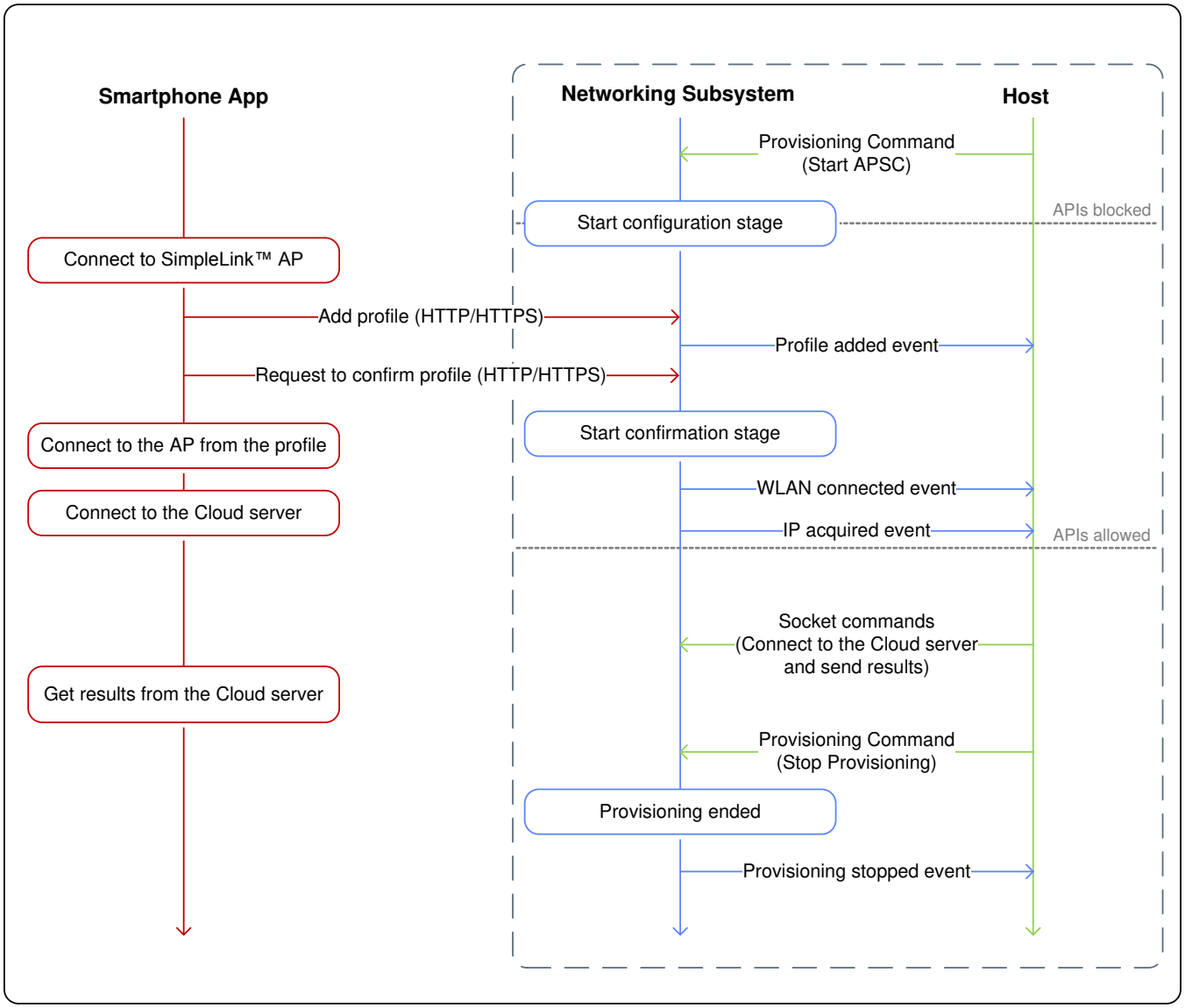

Figure 7. Successful AP Provisioning Cloud Example

NOTE: When user feedback is provided through a cloud server (external confirmation), the APIs are unblocked immediately after the CONFIRMATION_IP_ACQUIRED provisioning status event is sent to the host. At this point the host is able to send to the networking subsystem the socket commands needed to connect to the cloud server.

Because the networking subsystem is unaware of the confirmation result coming from the cloud server, the host is responsible for stopping the provisioning process (and ordering the networking subsystem to stay in its active role, STA) if the confirmation is successful. For the same reason the host must order the networking subsystem to switch back to the configuration stage (by sending the ABORT_EXTERNAL_CONFIRMATION command) if the confirmation failed.

8.6 Using External Configuration Method: WAC

Figure 8 shows a sequence diagram describing a successful provisioning process using an external provisioning method WAC.

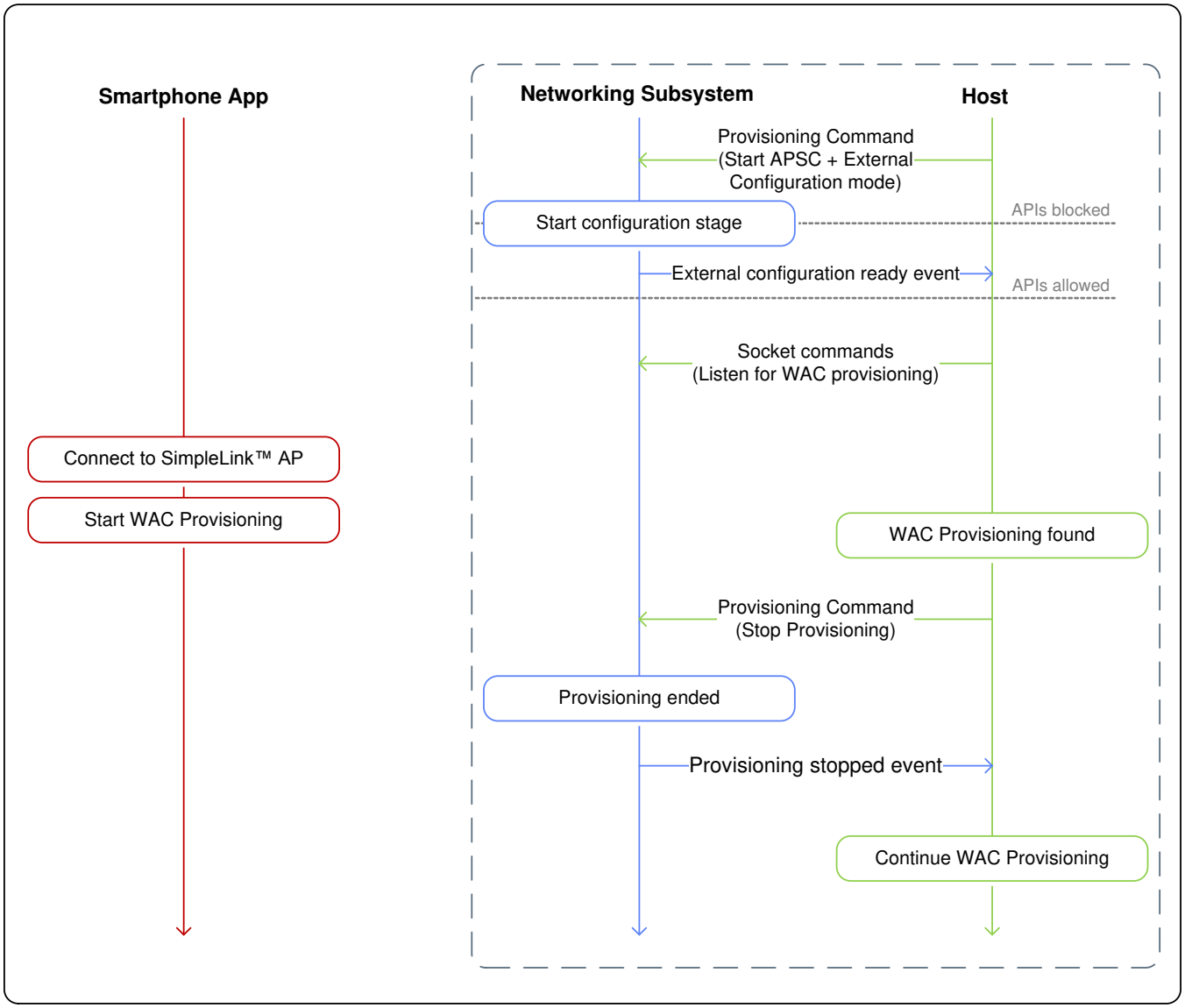

Figure 8. WAC Provisioning Example

NOTE: When Provisioning is started in APSC plus External configuration mode, the host can start sending commands to the networking subsystem only after the *external configuration ready* provisioning status event is received. When the host identifies that a user has started a provisioning process using the external configuration method, it should order the networking subsystem to stop the internal provisioning process. When the networking subsystem is stopped, the host can continue with its provisioning process.

8.7 Successful SmartConfig™ Provisioning While External Configuration Enabled

Figure 9 shows a sequence diagram describing a successful provisioning process using the SmartConfig™ method while external configuration is enabled.

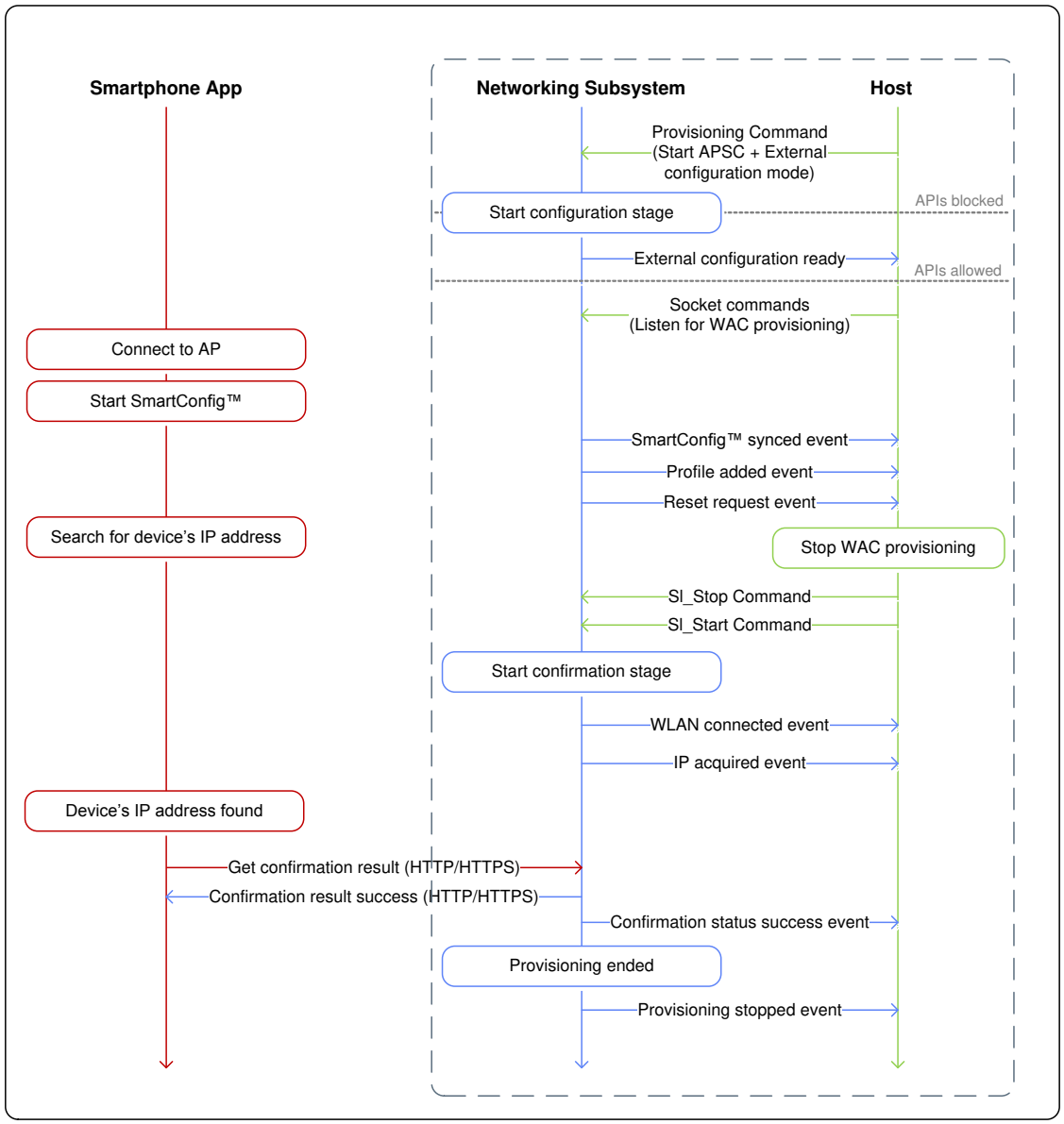

Figure 9. Successful SC Provisioning While External Configuration Enabled Example

NOTE: When Provisioning is started in APSC plus External configuration mode, and the user is using one of the internal provisioning methods (AP or SC), the device sends a reset request event to the host. The host should close all the sockets and activities of its external provisioning process, and restart the device. When the device is restarted, it continues to the confirmation stage and the internal provisioning process continues as usual.

A.1 Provisioning HTTP/HTTPS Server APIs

Table 6 shows the provisioning HTTP/HTTPS server APIs.

Table 6. Provisioning HTTP/HTTPS Server APIs

Function	Method	URI	Parameters and Return Values
Add Profile	POST	/api/1/wlan/profile_add	__SL_P_P.A = SSID __SL_P_P.B = Security type __SL_P_P.C = Security key __SL_P_P.D = Priority
Set Device Name	POST	/api/1/netapp/set_urn	__SL_P_S.B = Device name
Get Device Name	GET	/param_device_name.txt	Returns device name string
Confirmation Request	POST	/api/1/wlan/confirm_req	None
Get Confirmation Result	GET	/param_cfg_result.txt	Return values: 0 - Confirmation not started 1 - SSID not found 2 - Connection failed 3 - IP not acquired 4 - Feedback failed 5 - Confirmation success
Get Device Version	GET	/param_product_version.txt	Returns R2.0 (for CC3120/CC3135 and CC3220/CC3235x devices)
Start AP Scan	POST	/api/1/wlan/en_ap_scan	__SL_P_SC1 = time between scan cycles __SL_P_SC2 = number of scan cycles
Get Scan Results	GET	/netlist.txt	Returns a List of SSIDs and their security types (0-Open, 1-WEP, 3-WPA/WPA2, 5-WPA3)

Revision History

NOTE: Page numbers for previous revisions may differ from page numbers in the current version.

Changes from A Revision (January 2019) to B Revision	Page
• Updated Get Scan Results function in the Provisioning HTTP/HTTPS Server APIs table.	20

IMPORTANT NOTICE AND DISCLAIMER

TI PROVIDES TECHNICAL 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 AND 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 skilled developers designing with TI products. You are solely responsible for (1) selecting the appropriate TI 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. TI grants you permission to use these resources only for development of an application that uses the TI products described in the resource. Other reproduction and display of these resources is prohibited. No license is granted to any other TI intellectual property right or to any third party intellectual property right. TI disclaims responsibility for, and you will fully indemnify TI and its representatives against, any claims, damages, costs, losses, and liabilities arising out of your use of these resources.

TI’s products are provided subject to TI’s Terms of Sale (www.ti.com/legal/termsofsale.html) or other applicable terms available either on ti.com or provided in conjunction with such TI products. TI’s provision of these resources does not expand or otherwise alter TI’s applicable warranties or warranty disclaimers for TI products.

Mailing Address: Texas Instruments, Post Office Box 655303, Dallas, Texas 75265