Mesh Remote Provisioning

|  |
| --- |
| *Bluetooth®* Change Request |
| * Revision: r16 * Revision Date: 2018-Oct-15 * Group Prepared By: Mesh Working Group * Feedback Email: mesh[-main@bluetooth.org](mailto:-main@bluetooth.org) * Impacted Specifications: Mesh Profile 1.0   This Change Request proposes changes to the following specification:   * Mesh Profile v1.0 (“Source Specification”)   **Abstract:**  The proposed Remote Provisioning feature allows adding unprovisioned devices to a Bluetooth mesh network when the Provisioner is beyond immediate radio range of the unprovisioned device. The existing mesh network is used to transport the Provisioning PDU. Additionally, the Provisioner may change the device key of the mesh node using Provisioning PDUs that are transported over the mesh network. |

Revision History

| Revision Number | Date | Comments |
| --- | --- | --- |
| r00 | 2017-Jul-5 | Initial Draft |
| r04 | 2017-Aug-28 | Integration of the Bob’s and Thomas comments |
| r05 | 2017-Aug-30 | Removing Remote Provisioning Close Reason State. Renaming Remote Provisioning Server Composite state to Remote Provisioning Server Status state and Remote Provisioning Server State to Remote Provisioning Server Phase state. Allowing restarting of scan. |
| r06 | 2017-Sep-12 | Added Extended Scanning. Added message summary tables. The functionality of the Remote Provisioning PDU Report divided into two messages. |
| R07 | 2017-Sep-25 | Added state machine figure. Added or changed some behaviors to assure idempotence. Added one missing client procedure. Fixing names of the Servers and Clients in behavior sections. Added PB-Remote procedures. Fixed some references. |
| R08 | 2017-Oct-23 | Added Remote Provisioning Outbound PDU Number State and Remote Provisioning Inbound PDU Number State and appropriate fields in messages. Added OOB and Hash fields in Remote Provisioning Scan Report message. Complete separation between scanning and provisioning models. Renamed Extended Scanning to Detailed Scanning. Further work on idempotence and robustness- added Outbound Report Delivery phase. |
| R09 | 2017-Nov-10 | Many edits with comments resolution. |
| R10 | 2018-Jan-12 | TE comments resolution |
| R11 | 2018-Mar-03 | All Reports type messages are using publish mechanism. Changed and renamed Remote Provisioning Scanning Reports Destination and Remote Provisioning Reports Destination states. Some additional edits to fix missing descriptions in models. First step to move to the CR template.  Rename the Remote Provisioning Outbound PDU Number state to Remote Provisioning Outbound PDU Count state. Rename the Remote Provisioning Inbound PDU Number state to Remote Provisioning Inbound PDU Count state.  Renamed states:  Remote Provisioning Server Status -> Remote Provisioning Server Information  Remote Provisioning Scanning Server Status -> Remote Provisioning Scanning Server Information  Renamed messages:  Remote Provisioning State Get -> Remote Provisioning Information Get  Remote Provisioning State Status -> Remote Provisioning Information Status  Added two new messages:  Remote Provisioning Scan Information Get  Remote Provisioning Scan Information Status  Simplified behavior of Remote Provisioning Scan Get and Remote Provisioning Scan Status messages. |
| R12 | 2018-Apr-13 | Mandating models to use device key and removing usage of the publishing mechanism.  Removing protections against accidental Link Open/Close and Scan Start/Stop messages (handling multiple provisioners [we use device key for messages] are out of scope of the specification).  Adding Report Address field to Remote Provisioning Scan Start and Remote Provisioning Link Open.  Removing Scanning Cannot Cancel and Link Cannot Close status codes.  Detailed scanning is now mandatory.  Removed not needed requirements in behavior of remote Provisioning Scan Report, Remote Provisioning Extended Scan Report, Remote Provisioning Link Report, Remote Provisioning PDU Outbound Report, Remote Provisioning PDU Report.  Remote Provisioning state machine picture updated.  Added handling of Remote Provisioning Link Open message in Outbound Report Delivery phase.  Added 4.2.23.2.1 Remote Provisioning Timeout behavior and adjusted other texts to match this change. Added clear description of mandatory passive and optional active scanning. Renamed Link Idle to Link Active for better clarity. Added description when to start using the new Device Key in case of using PB-Local. |
| R13 | 2018-Jul-18 | Remote Provisioning Scan Server model merged into Remote Provisioning Server and Remote Provisioning Scan Client merged into Remote Provisioning Client. Renamed Remote Provisioning Server Phase state to Remote Provisioning state and renamed Remote Provisioning Scanning Server Phase state to Remote Provisioning Scanning state. Renamed Phase field to RPScanningState field in some messages and renamed Phase field to RPState field in other messages. Added Device UUID state to Remote Provisioning Server Information state. Added DeviceUUID field to Remote Provisioning Information Status message. Renamed the PB-Local to the Device Key Refresh procedure and located it in 3.10.8. Applying the New Device Key is independent of Device Key Refresh procedure. During the Device Key Refresh procedure, the IV Index and IV update flags are ignored. The Remote Provisioning Server provisioning has new drawing of the state machine. Renamed PB-Local to Device Key Refresh procedure. Defined New Device Key that can be set OOB or via Device Key Refresh procedure. Added behavior to apply the New Device Key as node’s device key. The Client is not selecting provisioning bearer. Scanning does not report same UUID unless the URI Hash becomes available via different provisioning bearer. Scanning node when Device Key Refresh procedure is available is supported, since OOB Information and URI Hash are needed by the Provisioner. Added states: Device UUID, OOB Information, Link Close Reason and Link Close Status. Removed states: Remote Provisioning Bearer Type Filter, Supported Scanning Bearer Types, Remote Provisioning Open Bearer Type. Removed fields: SupportedBearerTypes from Remote Provisioning Scan Information Status messages, FilterBearerTypes from Remote Provisioning Scan Start message, FilterBearerTypes from Remote Provisioning Scan Status message, BearerTypes from Remote Provisioning Scan Report message, OpenBearerType from Remote Provisioning Link Open message. Removed status codes: Link Opening, Link Opened, Link Already Open, Provisioning PDU Delivered. Added status codes: Link Closed by Client, Link Closed as Cannot Receive PDU, Link Closed as Cannot Send PDU, Link Closed as Cannot Deliver PDU Report, Link Closed as Cannot Deliver PDU Outbound Report. Renamed status codes: Provisioning PDU Cannot Transfer to Provisioning PDU Temporary Cannot Transfer, Wrong Bearer to Cannot Select Bearer. Moved sending of the Link Report after the finish of the PB-Remote Link Close procedure – thus the Client will always know when Server is ready. Renamed the Remote Provisioning UUID state to the Remote Provisioning Scanning UUID for better clarity. The Remove Remote Provisioning Scan Information Get and Remote Provisioning Scan Information Status messages are removed. The Remote Provisioning Link Get message is added. The Remote Provisioning Scanning Server Information state removed. States are rearranged so there are three composite states. Added OOBInformation and URIHash fields in Remote Provisioning Information Status message. |
| R14 | 2018-Aug-18 | Added Remote Provisioning Extended Scan Start. Renamed Local Provisioning Network Interface to Device Key Refresh interface. The Device Key Refresh procedure is now between the Remote Provisioning Client and Remote Provisioning Server. The Remote Provisioning Server now only reports UUIDs of devices that it can provisioned. The bearers information is not reported and Client cannot select which bearer to use. Added Remote Provisioning Procedure state to identify if remote provisioning or device key refresh is active. The NetKeyIndex parameter added to some messages. |
| R15 | 2018-Oct-1 | Version ready for TE review. |
| R16 | 2018-Oct-15 | TE comments resolved. |

Contributors

| Name | Company |
| --- | --- |
| Piotr Winiarczyk | Silvair Inc. |
| Thomas Stenersen | Nordic Semiconductor ASA |
| Robert D. Hughes | Intel Corporation |
| Brian Gix | Intel Corporation |
| Omkar Kulkarni | Nordic Semiconductor ASA |
| Victor Zhodzishsky | Cypress Semiconductor Corporation |
| Yao Wang | Barrot |

Use of this specification is your acknowledgement that you agree to and will comply with the following notices and disclaimers. You are advised to seek appropriate legal, engineering, and other professional advice regarding the use, interpretation, and effect of this specification.

Use of Bluetooth specifications by members of Bluetooth SIG is governed by the membership and other related agreements between Bluetooth SIG and its members, including those agreements posted on Bluetooth SIG’s website located at www.bluetooth.com. Any use of this specification by a member that is not in compliance with the applicable membership and other related agreements is prohibited and, among other things, may result in (i) termination of the applicable agreements and (ii) liability for infringement of the intellectual property rights of Bluetooth SIG and its members.

Use of this specification by anyone who is not a member of Bluetooth SIG is prohibited and is an infringement of the intellectual property rights of Bluetooth SIG and its members. The furnishing of this specification does not grant any license to any intellectual property of Bluetooth SIG or its members. THIS SPECIFICATION IS PROVIDED “AS IS” AND BLUETOOTH SIG, ITS MEMBERS AND THEIR AFFILIATES MAKE NO REPRESENTATIONS OR WARRANTIES AND DISCLAIM ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING ANY WARRANTIES OF MERCHANTABILITY, TITLE, NON-INFRINGEMENT, FITNESS FOR ANY PARTICULAR PURPOSE, OR THAT THE CONTENT OF THIS SPECIFICATION IS FREE OF ERRORS. For the avoidance of doubt, Bluetooth SIG has not made any search or investigation as to third parties that may claim rights in or to any specifications or any intellectual property that may be required to implement any specifications and it disclaims any obligation or duty to do so.

TO THE MAXIMUM EXTENT PERMITTED BY APPLICABLE LAW, BLUETOOTH SIG, ITS MEMBERS AND THEIR AFFILIATES DISCLAIM ALL LIABILITY ARISING OUT OF OR RELATING TO USE OF THIS SPECIFICATION AND ANY INFORMATION CONTAINED IN THIS SPECIFICATION, INCLUDING LOST REVENUE, PROFITS, DATA OR PROGRAMS, OR BUSINESS INTERRUPTION, OR FOR SPECIAL, INDIRECT, CONSEQUENTIAL, INCIDENTAL OR PUNITIVE DAMAGES, HOWEVER CAUSED AND REGARDLESS OF THE THEORY OF LIABILITY, AND EVEN IF BLUETOOTH SIG, ITS MEMBERS OR THEIR AFFILIATES HAVE BEEN ADVISED OF THE POSSIBILITY OF THE DAMAGES.

If this specification is a prototyping specification, it is solely for the purpose of developing and using prototypes to verify the prototyping specifications at Bluetooth SIG sponsored IOP events. Prototyping Specifications cannot be used to develop products for sale or distribution and prototypes cannot be qualified for distribution.

Products equipped with Bluetooth wireless technology ("Bluetooth Products") and their combination, operation, use, implementation, and distribution may be subject to regulatory controls under the laws and regulations of numerous countries that regulate products that use wireless non-licensed spectrum. Examples include airline regulations, telecommunications regulations, technology transfer controls and health and safety regulations. You are solely responsible for complying with all applicable laws and regulations and for obtaining any and all required authorizations, permits, or licenses in connection with your use of this specification and development, manufacture, and distribution of Bluetooth Products. Nothing in this specification provides any information or assistance in connection with complying with applicable laws or regulations or obtaining required authorizations, permits, or licenses.

Bluetooth SIG is not required to adopt any specification or portion thereof. If this specification is not the final version adopted by Bluetooth SIG’s Board of Directors, it may not be adopted. Any specification adopted by Bluetooth SIG’s Board of Directors may be withdrawn, replaced, or modified at any time. Bluetooth SIG reserves the right to change or alter final specifications in accordance with its membership and operating agreements.

Copyright © 2015–2017. All copyrights in the Bluetooth Specifications themselves are owned by Apple Inc., Ericsson AB, Intel Corporation, Lenovo (Singapore) Pte. Ltd., Microsoft Corporation, Nokia Corporation, and Toshiba Corporation. The Bluetooth word mark and logos are owned by Bluetooth SIG, Inc. Other third-party brands and names are the property of their respective owners.

Contents

[1 Language 9](#_Toc527039167)

[**1.1** **Language conventions** 9](#_Toc527039168)

[2 Conventions used in this Change Request 10](#_Toc527039169)

[3 Changes to Source Specification 12](#_Toc527039170)

[3.1 Changes to Section 3: Mesh networking 12](#_Toc527039171)

[4 13](#_Toc527039172)

[3.6.4.2 Receiving an Upper Transport PDU 13](#_Toc527039173)

[3.10.8 Device Key Refresh procedure 13](#_Toc527039174)

[3.10.8.1 Device Key Candidate 14](#_Toc527039175)

[3.10.8.2 Device Key Refresh Interface behavior 14](#_Toc527039176)

[4.1 15](#_Toc527039179)

[4.2 State definitions 15](#_Toc527039180)

[4.2.1 15](#_Toc527039181)

[4.2.2 15](#_Toc527039182)

[4.2.3 15](#_Toc527039183)

[4.2.4 15](#_Toc527039184)

[4.2.5 15](#_Toc527039185)

[4.2.6 15](#_Toc527039186)

[4.2.7 15](#_Toc527039187)

[4.2.8 15](#_Toc527039188)

[4.2.9 15](#_Toc527039189)

[4.2.10 15](#_Toc527039190)

[4.2.11 15](#_Toc527039191)

[4.2.12 15](#_Toc527039192)

[4.2.13 15](#_Toc527039193)

[4.2.14 15](#_Toc527039194)

[4.2.15 15](#_Toc527039195)

[4.2.16 15](#_Toc527039196)

[4.2.17 15](#_Toc527039197)

[4.2.18 15](#_Toc527039198)

[4.2.19 15](#_Toc527039199)

[4.2.20 15](#_Toc527039200)

[4.2.21 15](#_Toc527039201)

[4.2.22 Remote Provisioning Max Scanned Items 15](#_Toc527039202)

[4.2.23 Remote Provisioning Scan Parameters 15](#_Toc527039203)

[4.2.23.1 Remote Provisioning Scanning 15](#_Toc527039204)

[4.2.23.2 Remote Provisioning Scan Items Limit 16](#_Toc527039205)

[4.2.23.3 Remote Provisioning Timeout 16](#_Toc527039206)

[4.2.24 Remote Provisioning Link 16](#_Toc527039207)

[4.2.24.1 Remote Provisioning 16](#_Toc527039208)

[4.2.24.2 Remote Provisioning Procedure 16](#_Toc527039209)

[4.2.24.3 Remote Provisioning Device UUID 17](#_Toc527039210)

[4.2.24.4 Remote Provisioning Outbound PDU 17](#_Toc527039211)

[4.2.24.5 Remote Provisioning Outbound PDU Count 17](#_Toc527039212)

[4.2.24.6 Remote Provisioning Inbound PDU Count 17](#_Toc527039213)

[4.2.24.7 Link Close Reason 17](#_Toc527039214)

[4.2.24.8 Link Close Status 17](#_Toc527039215)

[4.3 Message definitions 17](#_Toc527039216)

[4.3.1 17](#_Toc527039217)

[4.3.2 17](#_Toc527039218)

[4.3.3 17](#_Toc527039219)

[4.3.4 Remote Provisioning messages 17](#_Toc527039220)

[4.3.4.1 Remote Provisioning Scan Capabilities Get 18](#_Toc527039221)

[4.3.4.2 Remote Provisioning Scan Capabilities Status 18](#_Toc527039222)

[4.3.4.3 Remote Provisioning Scan Get 18](#_Toc527039223)

[4.3.4.4 Remote Provisioning Scan Start 18](#_Toc527039224)

[4.3.4.5 Remote Provisioning Scan Stop 19](#_Toc527039225)

[4.3.4.6 Remote Provisioning Scan Status 19](#_Toc527039226)

[4.3.4.7 Remote Provisioning Scan Report 20](#_Toc527039227)

[4.3.4.8 Remote Provisioning Extended Scan Start 20](#_Toc527039228)

[4.3.4.9 Remote Provisioning Extended Scan Report 21](#_Toc527039229)

[4.3.4.10 Remote Provisioning Link Get 22](#_Toc527039230)

[4.3.4.11 Remote Provisioning Link Open 22](#_Toc527039231)

[4.3.4.12 Remote Provisioning Link Close 22](#_Toc527039232)

[4.3.4.13 Remote Provisioning Link Status 23](#_Toc527039233)

[4.3.4.14 Remote Provisioning Link Report 23](#_Toc527039234)

[4.3.4.15 Remote Provisioning PDU Send 24](#_Toc527039235)

[4.3.4.16 Remote Provisioning PDU Outbound Report 25](#_Toc527039236)

[4.3.4.17 Remote Provisioning PDU Report 25](#_Toc527039237)

[4.3.5 Messages summary 25](#_Toc527039238)

[4.3.5.1 Alphabetical summary of opcodes 25](#_Toc527039239)

[4.3.5.2 Numerical summary of opcodes 26](#_Toc527039240)

[4.3.6 Summary of status codes 26](#_Toc527039241)

[4.3.7 27](#_Toc527039242)

[4.4 Model definitions 27](#_Toc527039243)

[4.4.1 27](#_Toc527039244)

[4.4.2 27](#_Toc527039245)

[4.4.3 27](#_Toc527039246)

[4.4.4 27](#_Toc527039247)

[4.4.5 Remote Provisioning Server model 27](#_Toc527039248)

[4.4.5.1 Description 27](#_Toc527039249)

[4.4.5.2 Scan procedure 29](#_Toc527039250)

[4.4.5.3 Extended Scan procedure 30](#_Toc527039251)

[4.4.5.4 Provisioning procedure 31](#_Toc527039253)

[4.4.5.5 Behavior 34](#_Toc527039254)

[4.4.6 Remote Provisioning Client model 42](#_Toc527039255)

[4.4.6.1 Description 42](#_Toc527039256)

[4.4.6.2 Behavior 43](#_Toc527039258)

[4.4.7 Summary of SIG Model IDs 45](#_Toc527039259)

[5 Provisioning 47](#_Toc527039260)

[5.2 Provisioning bearer layer 49](#_Toc527039262)

[5.2.1 49](#_Toc527039263)

[5.2.2 49](#_Toc527039264)

[5.2.3 PB-Remote 49](#_Toc527039265)

[5.2.3.1 PB-Remote Provisioning Client 50](#_Toc527039266)

[5.2.3.2 PB-Remote Provisioning Server 50](#_Toc527039267)

[5.2.3.3 PB-Remote procedures 50](#_Toc527039268)

[5.4.1.10 Provisioning Failed 52](#_Toc527039269)

[6 References 54](#_Toc527039270)

1. **Language**
   1. **Language conventions**

Please refer to and follow any terminology, language conventions, and interpretation sections of the Source Specification.

# Conventions used in this Change Request

The formatting and color conventions described in Table 2.1 below are used in the change request to describe the specific changes and additions that are proposed to the Source Specification(s) identified on the cover page.

| Text Color | Description |
| --- | --- |
| Black | Text that is unmodified from the Source Specification.  Note: The text of the Source Specification may not be black and may contain tracked changes or other colored text that are reflected as black text in this document. |
| Red | Text that is added to the Source Specification. |
| red strikethrough | Text that is deleted from the Source Specification. |
| [green bracketed text] | Comments that are intended to aid the reader. |
| blue | Default color used for section numbers and headings of this document. |

Table 2.1: Color key for headings, captions, and body text

# 

# Changes to Source Specification

This Section sets forth the specific changes and additions, using the formatting and color conventions described in Section 2, that are proposed to the Source Specification.

## Changes to Section 3: Mesh networking

# 

[Modify section 3.6.4.2

#### 3.6.4.2 Receiving an Upper Transport PDU

Upon receiving an Upper Transport Access PDU, the access payload shall be decrypted, and the TransMIC shall be authenticated against all known application keys or the device key or the Device Key Candidate (see Section 3.10.8.2) for which the AKF and AID fields match. If the Upper Transport Access PDU authenticates and it has been checked for replay attacks (see Section 3.8.8) then it is delivered to the access layer with the contextual information of this message such as the source address, destination addresses, and the keys used for decryption and authentication.

When the Device Key Candidate is available and an access message is decrypted using the Device Key Candidate that was delivered to the access layer, then the node shall revoke the device key, the Device Key Candidate shall become the device key, and the Device Key Candidate shall become unavailable.

Upon receiving an Upper Transport Control PDU, the destination address of the PDU shall be checked against the unicast address of the elements of this node and if it matches then the message shall be processed (see Section 3.6.6).

[Insert the new sections]

### 3.10.8 Device Key Refresh procedure

This procedure is used to change the device key (DevKey) without re-provisioning a node and without a need to reconfigure the node. The Device Key Refresh procedure does not transfer a device key to the device over the air; instead, it uses the provisioning protocol to compute the Device Key Candidate (see Section 3.10.8.1). The device key value change using this procedure is thus performed at the same security level as is provisioning of the unprovisioned device. The Address, NetKey, NetKey Index, and IV Index that are provided using the provisioning protocol must match the values stored on the node; the value of the Flags field is ignored.

While the Device Key Refresh procedure is active, the Provisioning PDUs between the Provisioner and the node are routed locally to the layer executing the provisioning protocol over the Device Key Refresh Interface (see Section 3.10.8.2).

Figure 3.X illustrates how the Provisioner executes the Device Key Refresh procedure over PB-Remote provisioning bearer (see Section 5.2.3) to change the Device Key Candidate of the node.

**

Figure 3.X: Devices participating in changing the Device Key Candidate using the Device Key Refresh procedure over PB-Remote

The Device Key Refresh procedure shall be supported if PB-Remote is supported on the node. The Provisioner shall use the Device Key Refresh procedure over the PB-Remote provisioning bearer.

No more than one Device Key Refresh procedure shall be active on the node at any time.

The Device Key Refresh procedure starts with opening the Device Key Refresh Interface. Then Provisioning PDUs are exchanged, and the provisioning protocol is executed. Finally, the Device Key Refresh Interface is closed. The result of the Device Key Refresh procedure is the generation of a Device Key Candidate.

#### 3.10.8.1 Device Key Candidate

The Device Key Candidate is a candidate to become a new value of the device key. The Device Key Candidate may be delivered to the node by using an out-of-band (OOB) mechanism or by successfully executing the Device Key Refresh procedure. When the Device Key Candidate is available, it can replace the device key, as described in Section 3.6.4.2.

#### 3.10.8.2 Device Key Refresh Interface behavior

At power-up, the Device Key Refresh Interface shall be closed. When the Device Key Refresh Interface is closed, it shall not pass Provisioning PDUs.

The node opens the Device Key Refresh Interface when it receives a Remote Provisioning Link Open message (see Section 4.3.4.11) from the Provisioner indicating the Remote Provisioning Server itself as a destination. When the Device Key Refresh Interface opens, the Provisioning PDUs received over PB-Remote are delivered over the Device Key Refresh Interface to the layer executing the provisioning protocol on the node. The provisioning protocol processes and generates Provisioning PDUs as defined in Section 5.4.2.

The Device Key Refresh Interface can be closed by the Provisioner or by the layer executing the provisioning protocol on the node. The Provisioner can close the opened Device Key Refresh Interface at any time by sending a Remote Provisioning Link Close message (see Section 4.3.4.12). The Reason Code received in the Remote Provisioning Link Close message shall be passed over the Device Key Refresh Interface. When the layer that is executing the provisioning protocol encounters a protocol timeout error, it shall close the Device Key Refresh Interface, and the node shall delete the Device Key Candidate.

When the Provisioner closes the Device Key Refresh Interface with the Reason Code equal to Success after delivering a Provisioning Data PDU (see Section 5.4.1.8) that can be accepted (see Table 3.XX) over the Device Key Refresh Interface, then the Device Key Refresh procedure succeeds, and the node shall assume that new value of the device key is known to the Provisioner, and shall store the key value as the Device Key Candidate.

When the Provisioner closes the Device Key Refresh Interface with the Reason Code not equal to Success after delivering a Provisioning Data PDU (see Section 5.4.1.8) that can be accepted (see Table 3.XX) over the Device Key Refresh Interface, then the node shall delete the Device Key Candidate.

When the Device Key Refresh Interface receives a Provisioning Data PDU with provisioning data that cannot be accepted, then the node shall respond with Provisioning Failed PDU (see Section 5.4.1.10) with the Error Code parameter set to Invalid Data.

Table 3.XX defines the values of the Provisioning Data PDU that are required in order for the PDU to be accepted by the Device Key Refresh Interface. The Device Key Refresh procedure ignores the values of the Flags field.

| Provisioning Data PDU field | Device Key Refresh Interface acceptance criteria |
| --- | --- |
| Network Key | The Network Key field value is equal to the stored value of a Network Key identified by the Key Index field. |
| Key Index | The key identified by Key Index field is valid for this device. |
| IV Index | The IV Index field value is equal to the current value of the IV Index. |
| Unicast Address | The Unicast Address field value is equal to the unicast address of the primary element. |

Table 3.XX: Device Key Refresh Interface acceptance criteria for Provisioning Data PDU field values

## 

## State definitions

[Insert the following sections after the 4.2.21 section,]



### Remote Provisioning Scan Capabilities

The Remote Provisioning Scan Capabilities state is a composite state that indicates various capabilities of scanning in the Remote Provisioning Server model. The state includes a Remote Provisioning Max Scanned Items state, and a Remote Provisioning Active Scan state.

#### Remote Provisioning Max Scanned Items

The Remote Provisioning Max Scanned Items state indicates the maximum number of UUIDs that the Remote Provisioning Server can report during scanning. Table 4.1 defines the possible values. This state is read-only, and the state’s value is implementation specific. The minimum value of the state is 4. The maximum value of the state is 255.

| Value | Description |
| --- | --- |
| 0x00–0x03 | Prohibited |
| 0x04–0xFF | Maximum number of unprovisioned devices the Remote Provisioning Server can report |

Table 4.1: Remote Provisioning Max Scanned Items state values

#### Remote Provisioning Active Scan

The Remote Provisioning Active Scan state indicates if the Remote Provisioning Server supports active scanning (see [2] Vol 6, Part B, Section 4.4.3.2). Table 4.2 defines possible value of the state. This state is read-only, and the state’s value is implementation specific.

| Value | Description |
| --- | --- |
| 0x00 | The Remote Provisioning Server does not support active scanning |
| 0x01 | The Remote Provisioning Server supports active scanning |
| 0x02–0xFF | Prohibited |

Table 4.2: Remote Provisioning Active Scan state values

### Remote Provisioning Scan Parameters

The Remote Provisioning Scan Parameters state is a composite state that indicates various parameters of scanning in the Remote Provisioning Server model. The state includes a Remote Provisioning Scan state, a Remote Provisioning Scan Items Limit state, and a Remote Provisioning Timeout state.

#### Remote Provisioning Scan

The Remote Provisioning Scan state enumerates the values defined in Table 4.2, which describe the state of the Remote Provisioning Scan procedure in the Remote Provisioning Server model (see Section 4.4.5.2).

| Value | Description |
| --- | --- |
| 0x00 | Idle |
| 0x01 | Remote Provisioning Multiple Devices Scan (not limited to one device) |
| 0x02 | Remote Provisioning Single Device Scan (limited to one device) |
| 0x03–0xFF | Reserved for Future Use |

Table 4.3: Remote Provisioning Scan state values

#### Remote Provisioning Scan Items Limit

The Remote Provisioning Scan Items Limit state identifies the maximum number of items the Remote Provisioning Server may report while performing the Remote Provisioning Scan procedure.

#### Remote Provisioning Timeout

The Remote Provisioning Timeout state indicates time left until the end of the Remote Provisioning Scan procedure (see Section 4.4.5.2). Table 4.3 defines the values for the Remote Provisioning Timeout state.

| Value | Description |
| --- | --- |
| 0x00 | The Remote Provisioning Scan procedure is not in progress. |
| 0x01–0xFF | The Remote Provisioning Scan procedure is in progress. The value indicates the number of seconds remaining before the scan will stop. |

Table 4.4: Remote Provisioning Timeout state values

### Remote Provisioning Link Parameters

The Remote Provisioning Link state is a composite state that indicates various parameters of a provisioning link. This state includes a Remote Provisioning Link state, a Remote Provisioning Device UUID state, a Remote Provisioning Outbound PDU Count state, a Remote Provisioning Inbound PDU Count state, a Link Close Reason state, and a Link Close Status state.

#### Remote Provisioning Link

The Remote Provisioning Link state enumerates the values defined in Table 4.4, which describe the state of the Remote Provisioning Server model. During the Device Key Refresh procedure, the Link Opening, Outbound Packet Transfer, and Link Closing values are not used.

| Value | Description |
| --- | --- |
| 0x00 | Idle |
| 0x01 | Link Opening |
| 0x02 | Link Active |
| 0x03 | Outbound Packet Transfer |
| 0x04 | Link Closing |
| 0x05–0xFF | Prohibited |

Table 4.5: Remote Provisioning Link state values



#### Remote Provisioning Device UUID

The Remote Provisioning Device UUID state indicates either the Device UUID that the provisioning bearer is open to or, if a Device Key Refresh Procedure is in progress, the Device UUID of the Remote Provisioning Server.

#### Remote Provisioning Outbound PDU Count

The Remote Provisioning Outbound PDU Count state indicates the number of unique Provisioning PDUs delivered to the Remote Provisioning Server from the Remote Provisioning Client during the provisioning process or a Device Key Refresh procedure that is in progress.

#### Remote Provisioning Inbound PDU Count

The Remote Provisioning Inbound PDU Count state indicates the number of unique Provisioning PDUs sent to the Remote Provisioning Client during the provisioning process or a Device Key Refresh procedure that is in progress.

#### Link Close Reason

The Link Close Reason state contains the provisioning bearer link close Reason. Table 5.12 defines the Reason field values used by the PB-ADV bearer for a link closing. For the bearers that do not define the link close reason, the value of the state is not defined and the state is not used.

#### Link Close Status

The Link Close Status state contains a status code that indicates the reason why the Remote Provisioning Server started the PB-Remote Close Link procedure (see Section 5.2.3.3.2).

[End of insertion]

## Message definitions

[Insert the following section after the 4.3.3 section (“Health messages”).]



### Remote Provisioning messages

Remote Provisioning messages are used by a Remote Provisioning Client to communicate with a Remote Provisioning Server over a mesh network to find the UUID of unprovisioned devices within immediate radio range of the Remote Provisioning Server and to provision a remote device. Remote Provisioning messages also can be used to obtain extended information about an unprovisioned device or to execute a Device Key Refresh procedure.

#### Remote Provisioning Scan Capabilities Get

The Remote Provisioning Scan Capabilities Get message is an acknowledged message used by the Remote Provisioning Client to get the value of the Remote Provisioning Scan Capabilities state (see Section 4.2.22.

The response to a Remote Provisioning Scan Capabilities Get message is a Remote Provisioning Scan Capabilities Status message (see Section 4.3.4.6).

There are no parameters for this message.

#### Remote Provisioning Scan Capabilities Status

The Remote Provisioning Scan Capabilities Status message is an unacknowledged message used by the Remote Provisioning Server to report the current value of the Remote Provisioning Scan Capabilities state of a Remote Provisioning Server.

The structure of the message is defined in Table 4.6.

| Field | Size (octets) | Notes |
| --- | --- | --- |
| MaxScannedItems | 1 | The maximum number of UUIDs that can be reported during scanning |
| ActiveScan | 1 | Indication if active scan is supported |

Table 4.6: Remote Provisioning Scan Capabilities Status message fields

The MaxScannedItems field identifies the value of the Remote Provisioning Max Scanned Items state (see Section 4.2.22.1).

The ActiveScan field identifies the value of the Remote Provisioning Active Scan state (see Section 4.2.22.2).

#### Remote Provisioning Scan Get

The Remote Provisioning Scan Get message is an acknowledged message that is used by the Remote Provisioning Client to get the various scanning states of a Remote Provisioning Server model.

The response to a Remote Provisioning Scan Get message is a Remote Provisioning Scan Status message (see Section 4.3.4.6).

There are no parameters for this message.

#### Remote Provisioning Scan Start

The Remote Provisioning Scan Start message is an acknowledged message that is used by the Remote Provisioning Client to start the Remote Provisioning Scan procedure, which finds unprovisioned devices within immediate radio range of the Remote Provisioning Server (see Section 4.4.5.2).

The response to a Remote Provisioning Scan Start message is a Remote Provisioning Scan Status message (see Section 4.3.4.6).

The structure of the Remote Provisioning Scan Start message is defined in Table 4.7.

| Field | Size (octets) | Notes |
| --- | --- | --- |
| ScannedItemsLimit | 1 | Maximum number of scanned items to be reported |
| Timeout | 1 | Time limit for a scan (in seconds) |
| UUID | 16 | Device UUID (Optional) |

Table 4.7: Remote Provisioning Scan Start message fields

The ScannedItemsLimit field identifies the maximum number of unprovisioned devices the Remote Provisioning Server can report while executing the Remote Provisioning Scan procedure. Value 0 indicates that the Remote Provisioning Client does not set a limit on the number of unprovisioned devices that the Remote Provisioning Server can report.

The Timeout field identifies the new value of the Remote Provisioning Timeout state (see Section 4.2.23.2). The value of the Timeout field shall not be 0.

If the UUID field is present, the Remote Provisioning Client is requesting a Single Device Scanning procedure, i.e., a scan for a specific device identified by the value of the UUID field. If the UUID field is absent, the Remote Provisioning Client is requesting a scan for all unprovisioned devices within immediate radio range (a Multiple Devices Scanning).

#### Remote Provisioning Scan Stop

The Remote Provisioning Scan Stop message is an acknowledged message that is used by the Remote Provisioning Client to terminate the Remote Provisioning Scan procedure (see Section 4.4.5.2).

The response to a Remote Provisioning Scan Stop message is a Remote Provisioning Scan Status message (see Section 4.3.4.6).

There are no parameters for this message.

#### Remote Provisioning Scan Status

The Remote Provisioning Scan Status message is an unacknowledged message used by the Remote Provisioning Server to report the current value of the Remote Provisioning Scan Parameters state and the Remote Provisioning Scan state of a Remote Provisioning Server model.

The structure of the message is defined in Table 4.8.

| Field | Size (octets) | Notes |
| --- | --- | --- |
| Status | 1 | Status for the requesting message |
| RPScanningState | 1 | The Remote Provisioning Scan state value |
| ScannedItemsLimit | 1 | Maximum number of scanned items to be reported |
| Timeout | 1 | Time limit for a scan (in seconds) |

Table 4.8: Remote Provisioning Scan Status message fields

The Status field identifies the status of the most recent operation on Remote Provisioning Scan state, as defined in Table 4.22.

The RPScanningState field identifies the value of the Remote Provisioning Scan state (see Section 4.2.23.1).

The ScannedItemsLimit field identifies the maximum number of unprovisioned devices as requested by the Remote Provisioning Client in the Remote Provisioning Scan Start message.

The Timeout field identifies the current value of the Remote Provisioning Timeout state (see Section 4.2.23.2).

#### Remote Provisioning Scan Report

The Remote Provisioning Scan Report message is an unacknowledged message used by the Remote Provisioning Server to report the scanned Device UUID of an unprovisioned device. Based on the Remote Provisioning Scan Reports received from multiple Remote Provisioning Servers, the Remote Provisioning Client can select the most suitable Remote Provisioning Server to execute the Extended Scan procedure and/or to provision the unprovisioned device.

The structure of the message is defined in Table 4.9.

| Field | Size (octets) | Notes |
| --- | --- | --- |
| RSSI | 1 | Signed integer that is interpreted as an indication of received signal strength measured in dBm. |
| UUID | 16 | Device UUID |
| OOB | 2 | OOB information |

Table 4.9: Remote Provisioning Scan Report message fields

The RSSI field contains a signed 8-bit value and is interpreted as an indication of received signal strength measured in dBm. The Remote Provisioning Server measures the RSSI value on packets sent by the unprovisioned device. If the RSSI cannot be read, the RSSI field shall be set to 127.

The UUID field identifies the Device UUID of the unprovisioned device.

The OOB field identifies the OOB Information of the unprovisioned device (see Table 3.54).

#### Remote Provisioning Extended Scan Start

The Remote Provisioning Extended Scan Start message is an unacknowledged message that is used by the Remote Provisioning Client to request additional information about a specific unprovisioned device or about the Remote Provisioning Server itself.

As a result of processing a Remote Provisioning Extended Scan Start message, the Remote Provisioning Server sends a Remote Provisioning Extended Scan Report message (see Section 4.3.4.9).

The structure of the Remote Provisioning Extended Scan Start message is defined in Table 4.10.

| Field | Size (octets) | Notes |
| --- | --- | --- |
| ADTypeFilterCount | 1 | Number of AD Types in the ADTypeFilter field |
| ADTypeFilter | variable | List of AD Types to be reported (C.1) |
| UUID | 16 | Device UUID (Optional) |
| Timeout | 1 | Time limit for a scan (in seconds) (C.2) |

C.1: If ADTypeFilterCount field value is not zero, the ADTypeFilter shall be present; otherwise the ADTypeFilter shall not be present

C.2: If UUID field is present, the Timeout field shall also be present; otherwise Timeout field shall not be present.

Table 4.10: Remote Provisioning Extended Scan Start message fields



The ADTypeFilterCount field identifies number of AD types listed in the ADTypeFilter field.

If present, the ADTypeFilter is a list of AD types that the client is requesting. The ADTypeFilter shall not contain the Shortened Local Name AD Type, the Incomplete List of 16-bit Service UUIDs AD Type, the Incomplete List of 32-bit Service UUIDs AD Type, or the Incomplete List of 128-bit Service UUIDs AD Type.

Note: If the ADTypeFilter field contains the Complete Local Name AD Type, the client is requesting either the Complete Local Name or the Shortened Local Name.

If present, the UUID field identifies the Device UUID of the unprovisioned device for which additional information is requested (see Section 4.4.5.3). If the UUID field is absent, the request retrieves information about the Remote Provisioning Server (see Section 4.4.5.5.2.1). In the latter case, the Remote Provisioning Server ignores the Timeout field value.

The Timeout field indicates how long the Remote Provisioning Client requests the Remote Provisioning Server to collect information about the unprovisioned device identified by the UUID. Table 4.11 defines the values for the Timeout field.

| **Value** | **Description** |
| --- | --- |
| 0x00 | Prohibited |
| 0x01−0x05 | Length of time (in seconds) to collect information about the unprovisioned device |
| 0x06-0xFF | Prohibited |

*4.11*



#### Remote Provisioning Extended Scan Report

The Remote Provisioning Extended Scan Report message is an unacknowledged message used by the Remote Provisioning Server to report the advertising data requested by the client in a Remote Provisioning Extended Scan Start message (see Section 4.3.4.8).

The structure of the message is defined in Table 4.12.

| Field | Size (octets) | Notes |
| --- | --- | --- |
| Status | 1 | Status for the requesting message |
| UUID | 16 | Device UUID |
| OOBInformation | 2 | OOB Information (Optional) |
| AdvStructures | Variable | Concatenated list of AD Structures that match the AD Types requested by the client in the ADTypeFilter field of the Remote Provisioning Extended Scan Start message. (C.1). |

C.1: If OOBInformation field is present, the AdvStructures field is optional; otherwise AdvStructures field shall not be present.

Table 4.12: Remote Provisioning Extended Scan Report message fields

The Status field identifies the status of the Remote Provisioning Extended Scan Start processing, as defined in Table 4.22.

The UUID field identifies the Device UUID of either the unprovisioned device (see Section 3.10.3) or the Remote Provisioning Server.

The OOBInformation field contains the OOB Information of either the unprovisioned device (see Section 3.9.2) or the Remote Provisioning Server.

If present, the AdvStructures field contains a concatenated list of AD Structures with information requested by the Remote Provisioning Client. The value has the same format as advertising data or scan response data, as defined in [1] Section 11, Vol 3, Part C.

#### Remote Provisioning Link Get

The Remote Provisioning Link Get message is an acknowledged message used by the Remote Provisioning Client to get the Remote Provisioning Link state of a Remote Provisioning Server model.

The response to a Remote Provisioning Link Get message is a Remote Provisioning Link Status message (see Section 4.3.4.13).

There are no parameters for this message.

#### Remote Provisioning Link Open

The Remote Provisioning Link Open message is an acknowledged message used by the Remote Provisioning Client to establish the provisioning bearer between a node supporting the Remote Provisioning Server model and an unprovisioned device, or to open the Device Key Refresh Interface.

The response to a Remote Provisioning Link Open message is a Remote Provisioning Link Status message (see Section 4.3.4.13).

The structure of the Remote Provisioning Link Open message is defined in Table 4.13.

| Field | Size (octets) | Notes |
| --- | --- | --- |
| UUID | 16 | Device UUID (Optional) |

Table 4.13: Remote Provisioning Link Open message fields

If present, the UUID field identifies the Device UUID of an unprovisioned device that the link will be open to. If the UUID field is absent, the Remote Provisioning Server will open the Device Key Refresh Interface.

#### Remote Provisioning Link Close

The Remote Provisioning Link Close message is an acknowledged message used by the Remote Provisioning Client to close a provisioning bearer or the Device Key Refresh Interface.

The response to a Remote Provisioning Link Close message is a Remote Provisioning Link Status message (see Section 4.3.4.13).

The structure of the Remote Provisioning Link Close message is defined in Table 4.14.

| Field | Size (octets) | Notes |
| --- | --- | --- |
| Reason | 1 | Link close reason code |

Table 4.14: Remote Provisioning Link Close message fields

The Reason field identifies the reason for the provisioning link close. The Reason field values for the Remote Provisioning Link Close message are defined in Table 4.15.

| Reason Code | Reason Code Name | Notes |
| --- | --- | --- |
| 0x00 | Success | The provisioning or Device Key Refresh procedure completed successfully. |
| 0x01 | Prohibited | Prohibited |
| 0x02 | Fail | The provisioning or Device Key Refresh procedure failed. |
| 0x03–0xFF | RFU | Reserved for Future Use |

Table 4.15: Reason field values for a Remote Provisioning Link Close message

#### Remote Provisioning Link Status

The Remote Provisioning Link Status message is an unacknowledged message used by the Remote Provisioning Server to acknowledge a Remote Provisioning Link Get message, a Remote Provisioning Link Open message, or a Remote Provisioning Link Close message.

The structure of the message is defined in Table 4.16.

| Field | Size (octets) | Notes |
| --- | --- | --- |
| Status | 1 | Status for the requesting message |
| RPState | 1 | Remote Provisioning Link state |

Table 4.16: Remote Provisioning Link Status message fields

The Status field identifies the status of the processing of the message from the client, as defined in Table 4.22.

The RPState field identifies the Remote Provisioning Link state (see Table 4.4).

#### Remote Provisioning Link Report

The Remote Provisioning Link Report message is an unacknowledged message used by the Remote Provisioning Server to report the state change of a provisioning bearer link or the Device Key Refresh Interface.

The structure of the message is defined in Table 4.17.

| Field | Size (octets) | Notes |
| --- | --- | --- |
| Status | 1 | Status of the provisioning bearer or the Device Key Refresh Interface |
| RPState | 1 | Remote Provisioning Link state |
| Reason | 1 | Link close Reason code (Optional) |

Table 4.17: Remote Provisioning Link Report message fields

The Status field identifies the provisioning bearer status as defined in Table 4.22.

The RPState field identifies the Remote Provisioning Link state.

If present, the Reason field identifies the reason for the provisioning link close as defined in Table 5.12. The field may be present only when Status is either Link Closed by Device or Link Closed by Server and the provisioning bearer provides a Reason code.



#### Remote Provisioning PDU Send

The Remote Provisioning PDU Send message is an unacknowledged message used by the Remote Provisioning Client to deliver the Provisioning PDU to an unprovisioned device or to the Device Key Refresh Interface.

The Remote Provisioning PDU Send message should be sent as a Segmented Access message. Alternatively, the Remote Provisioning Client needs to keep track of Remote Provisioning PDU Outbound Report messages and may need to retry sending Remote Provisioning PDU Send message.

When the Remote Provisioning server receives a Remote Provisioning PDU Send message, the server attempts to deliver a Provisioning PDU. If the Provisioning PDU is delivered, the Remote Provisioning Server sends a Remote Provisioning PDU Outbound Report message (see Section 4.3.4.16). If the Remote Provisioning Server fails to deliver the Provisioning PDU, the Remote Provisioning Link Report message is sent (see Section 4.3.4.14).

The structure of the Remote Provisioning PDU Send message is defined in Table 4.19.

| Field | Size (octets) | Notes |
| --- | --- | --- |
| OutboundPDUNumber | 1 | Provisioning PDU identification number |
| ProvisioningPDU | variable | Provisioning PDU |

Table 4.18: Remote Provisioning PDU Send message fields

The OutboundPDUNumber field identifies the identification number of the Provisioning PDU set in the ProvisioningPDU field.

The ProvisioningPDU field identifies the Provisioning PDU that either will be sent to an unprovisioned device or will be processed locally if the Device Key Refresh procedure is in progress.

#### Remote Provisioning PDU Outbound Report

The Remote Provisioning PDU Outbound Report message is an unacknowledged message used by the Remote Provisioning Server to report completion of the delivery of the Provisioning PDUs that the Remote Provisioning Server either sends to a device that is being provisioned or processes locally during the Device Key Refresh procedure.

The structure of the message is defined in Table 4.20.

| Field | Size (octets) | Notes |
| --- | --- | --- |
| OutboundPDUNumber | 1 | Remote Provisioning Outbound PDU Count state |

Table 4.19: Remote Provisioning PDU Outbound Report message field

The OutboundPDUNumber field contains the value of the Remote Provisioning Outbound PDU Count state.

#### Remote Provisioning PDU Report

The Remote Provisioning PDU Report message is an unacknowledged message used by the Remote Provisioning Server to report the Provisioning PDU that either was received from the device being provisioned or was generated locally during the Device Key Refresh procedure.

The structure of the message is defined in Table 4.21.

| Field | Size (octets) | Notes |
| --- | --- | --- |
| InboundPDUNumber | 1 | Number of received Provisioning PDUs |
| ProvisioningPDU | variable | Provisioning PDU |

Table 4.20: Remote Provisioning PDU Report message fields

The InboundPDUNumber field identifies the value of the Remote Provisioning Inbound PDU Count state (see Section 4.2.24.6).

The ProvisioningPDU field identifies the Provisioning PDU that was send by an unprovisioned device or generated locally during the Device Key Refresh procedure.

[End of insertion]

[The 4.3.4 Messages summary section will become 4.3.6 Messages summary section.]

### Messages summary

#### Alphabetical summary of opcodes

[Append the following rows to the bottom of the Table 4.106]

| Message Name | Opcode |
| --- | --- |
| Remote Provisioning Extended Scan Report | 0x80 0x57 |
| Remote Provisioning Extended Scan Start | 0x80 0x56 |
| Remote Provisioning Link Close | 0x80 0x5A |
| Remote Provisioning Link Get | 0x80 0x58 |
| Remote Provisioning Link Open | 0x80 0x59 |
| Remote Provisioning Link Report | 0x80 0x5C |
| Remote Provisioning Link Status | 0x80 0x5B |
| Remote Provisioning PDU Outbound Report | 0x80 0x5E |
| Remote Provisioning PDU Report | 0x80 0x5F |
| Remote Provisioning PDU Send | 0x80 0x5D |
| Remote Provisioning Scan Capabilities Get | 0x80 0x4F |
| Remote Provisioning Scan Capabilities Status | 0x80 0x50 |
| Remote Provisioning Scan Get | 0x80 0x51 |
| Remote Provisioning Scan Report | 0x80 0x55 |
| Remote Provisioning Scan Start | 0x80 0x52 |
| Remote Provisioning Scan Status | 0x80 0x54 |
| Remote Provisioning Scan Stop | 0x80 0x53 |

Table 4.106: Alphabetical summary of opcodes

#### Numerical summary of opcodes

[Append the following rows to the bottom of the Table 4.107]

| Message Name | Opcode |
| --- | --- |
| Remote Provisioning Scan Capabilities Get | 0x80 0x4F |
| Remote Provisioning Scan Capabilities Status | 0x80 0x50 |
| Remote Provisioning Scan Get | 0x80 0x51 |
| Remote Provisioning Scan Start | 0x80 0x52 |
| Remote Provisioning Scan Stop | 0x80 0x53 |
| Remote Provisioning Scan Status | 0x80 0x54 |
| Remote Provisioning Scan Report | 0x80 0x55 |
| Remote Provisioning Extended Scan Start | 0x80 0x56 |
| Remote Provisioning Extended Scan Report | 0x80 0x57 |
| Remote Provisioning Link Get | 0x80 0x58 |
| Remote Provisioning Link Open | 0x80 0x59 |
| Remote Provisioning Link Close | 0x80 0x5A |
| Remote Provisioning Link Status | 0x80 0x5B |
| Remote Provisioning Link Report | 0x80 0x5C |
| Remote Provisioning PDU Send | 0x80 0x5D |
| Remote Provisioning PDU Outbound Report | 0x80 0x5E |
| Remote Provisioning PDU Report | 0x80 0x5F |

Table 4.107: Numerical summary of opcodes

### Summary of status codes

Table 4.108 defines status codes for configuration messages (see Section 4.3.2) and health messages (see Section 4.33) that contain a Status parameter. Status messages are sent only in response to properly formatted messages (see Section 3.7.4.4).

[Table not changed]

| Status Code | Status Code Name |
| --- | --- |
| … | … |

*Table 4.108: Summary of status codes*

Table 4.22 defines status codes for Remote Provisioning Server messages that contain a status code.

| Status Code | Status Code Name |
| --- | --- |
| 0x00 | Success |
| 0x01 | Scanning Cannot Start |
| 0x02 | Invalid State |
| 0x03 | Limited Resources |
| 0x04 | Link Cannot Open |
| 0x05 | Link Open Failed |
| 0x06 | Link Closed by Device |
| 0x07 | Link Closed by Server |
| 0x08 | Link Closed by Client |
| 0x09 | Link Closed as Cannot Receive PDU |
| 0x0A | Link Closed as Cannot Send PDU |
| 0x0B | Link Closed as Cannot Deliver PDU Report |
| 0x0C | Link Closed as Cannot Deliver PDU Outbound Report |
| 0x0D –0xFF | Reserved for Future Use |

Table 4.109: Summary of Remote Provisioning Server model status codes



## Model definitions

[Insert the following sections after the 4.4.4 section]



### Remote Provisioning Server model

#### Description

The Remote Provisioning Server model is a root model (i.e., it does not extend any other models).

The model is used to scan for unprovisioned devices, to manage the provisioning bearer link while provisioning a new device, and to perform the Device Key Refresh procedure.

If supported, the Remote Provisioning Server shall be supported by a primary element and may be supported by any secondary element. The application-layer security on the model shall use the device key (see Section 2.3.9.1).

The model defines the state instances listed in Table 4.23.

| Remote Provisioning Server States | | Bound States | | |
| --- | --- | --- | --- | --- |
| State | Instance | Model | State | Instance |
| Remote Provisioning Scan Capabilities | Primary | - | - | - |
| Remote Provisioning Scan Parameters | Primary | - | - | - |
| Remote Provisioning Link Parameters | Primary | - | - | - |

Table 4.21: Remote Provisioning Server states and bindings

The structure of elements, states, and messages defined by this model is defined in Table 4.24.

| Element | SIG Model ID | States | Messages | **Rx** | **Tx** |
| --- | --- | --- | --- | --- | --- |
| Primary | 0x0004 | Remote Provisioning Scan Capabilities | Remote Provisioning Scan Capabilities Get | M |  |
| Remote Provisioning Scan Capabilities Status |  | M |
| Remote Provisioning Scan Parameters | Remote Provisioning Scan Get | M |  |
| Remote Provisioning Scan Start | M |  |
| Remote Provisioning Scan Stop | M |  |
| Remote Provisioning Scan Status |  | M |
| Remote Provisioning Scan Report |  | M |
| Remote Provisioning Extended Scan Start | M |  |
| Remote Provisioning Extended Scan Report |  | M |
| Remote Provisioning Link Parameters | Remote Provisioning Link Get | M |  |
| Remote Provisioning Link Open | M |  |
| Remote Provisioning Link Close | M |  |
| Remote Provisioning Link Status |  | M |
| Remote Provisioning Link Report |  | M |
| Remote Provisioning PDU Send | M |  |
| Remote Provisioning PDU Outbound Report |  | M |
| Remote Provisioning PDU Report |  | M |

Table 4.22: Remote Provisioning Server elements, states, and messages

M = Mandatory

The Remote Provisioning Server supports two scan procedures: the Remote Provisioning Scan procedure (see Section 4.4.5.2) and the Remote Provisioning Extended Scan procedure (see Section 4.4.5.3).

Note: The Remote Provisioning Scan procedure and the Remote Provisioning Extended Scan procedure are independent. For example, the Remote Provisioning Client can start the Remote Provisioning Scan procedure and, while that procedure is being executed, can perform one or more Remote Provisioning Extended Scan procedures. Termination of the Remote Provisioning Scan procedure does not affect any Remote Provisioning Extended Scan procedures that are in progress.

#### Remote Provisioning Scan procedure

The Remote Provisioning Client may put the Remote Provisioning Server into the Remote Provisioning Multiple Devices Scan state to search for unprovisioned devices within immediate radio range of the Remote Provisioning Server. The Remote Provisioning Client may put the Remote Provisioning Server into the Remote Provisioning Single Device Scan state to detect if a specific unprovisioned device is present within immediate radio range of the Remote Provisioning Server.

While executing the Remote Provisioning Scan procedure, the Remote Provisioning Server collects the Device UUIDs of unprovisioned devices and passes them to the Remote Provisioning Client via Remote Provisioning Scan Report messages (see Section 4.4.5.5.1.7).

The Remote Provisioning Server shall only report the devices that it is capable of provisioning. That is, the Remote Provisioning Server shall only send a Remote Provisioning Scan Report message for a device under either of the following circumstances:

* The server receives an Unprovisioned Device beacon (see Section 3.9.2), and it supports provisioning over the PB-ADV provisioning bearer.
* The server receives a connectable advertising packet with the Service Data for the «Mesh Provisioning Service» (see Section 7.1.2.2.1), and it supports provisioning over the PB-GATT bearer.

While it is in the Remote Provisioning Multiple Devices Scan or in the Remote Provisioning Single Device Scan state, the Remote Provisioning Server shall maintain a list of unprovisioned devices that it reported to the Remote Provisioning Client to filter out duplicates (discussed later in this section). The list shall be cleared when the Remote Provisioning Scan Start message is received, and it may be cleared when the Remote Provisioning Server exits the state.

**Starting a scan.** When the Remote Provisioning Server receives a Remote Provisioning Scan Start message with parameters that can be accepted (see Section 4.4.5.5.1.4), and the Remote Provisioning Scan state is Idle, the Server shall enter the Remote Provisioning Multiple Devices Scan or the Remote Provisioning Single Device Scan state.

When entering the state, the Remote Provisioning Server shall perform the following behaviors:

* Shall save the source address and the security material (NetKey Index) of the Remote Provisioning Scan Start message, and shall use them when sending Remote Provisioning Scan Report messages. If the saved security material becomes unavailable (for example, a NetKey is deleted), the Remote Provisioning Server shall exit the Scanning state.
* Shall set the Remote Provisioning Timeout state to the value of the Timeout field of the Remote Provisioning Scan Start message, and shall start the scanning timer using that value.
* Shall set the Remote Provisioning Server Scanning state to Remote Provisioning Multiple Devices Scan (0x01) if the Remote Provisioning Scan Start message does not contain the UUID field, or shall set it to Remote Provisioning Single Device Scan (0x02) if the UUID field is present.
* Shall set the Remote Provisioning Scan Items Limit state to the value of the ScannedItemsLimit field of the Remote Provisioning Scan Start message if the value is not zero, or shall set the Remote Provisioning Scan Item Limit state to the value of the Remote Provisioning Max Scanned Items state if the ScannedItemsLimit field value is equal to zero.

**Restarting the scan.** When the Remote Provisioning Server receives a Remote Provisioning Scan Start message with parameters that can be accepted (see Section 4.4.5.5.1.4), and the Remote Provisioning Scan state is not Idle, and the source address and the security material match the saved values, the scanning timer shall be restarted, and the values of the Remote Provisioning Timeout state, the Remote Provisioning Items Limit state, and the Remote Provisioning Scan state shall be updated according to the values specified in the received Remote Provisioning Scan Start message.

**Single Device vs. Multiple Devices scanning.** If the Remote Provisioning Scan Start message specifies a single device scan (the UUID field is present), then the Remote Provisioning Server shall only send information from advertising reports received from the device associated with the Device UUID that is specified in the message.

If the Remote Provisioning Scan Start message specifies Multiple Devices scanning, then the Remote Provisioning Server shall send information about the first N unprovisioned devices, where N is the value of the Remote Provisioning Scan Items Limit state.

**Completing the scan.** A Single Device scan is considered complete when the Remote Provisioning Server sends a Remote Provisioning Scan Report for the device with the UUID specified in the Remote Provisioning Scan Start message.

A Multiple Devices scan is considered complete when the Remote Provisioning Server sends Remote Provisioning Scan Reports for the value of the Remote Provisioning Scan Items Limit state devices.

**Stopping the scan.** The Remote Provisioning Server shall exit the Remote Provisioning Multiple Devices Scan or the Remote Provisioning Single Device Scan state (i.e., shall stop sending Remote Provisioning Scan Report messages) when it receives a Remote Provisioning Scan Stop message, or when the Scanning timer expires, or when scanning is completed as specified above. If the Scanning timer is running when the Remote Provisioning Server exits Scanning state, the scanning timer shall be canceled, and the Remote Provisioning Timeout state shall be set to zero. When the Remote Provisioning Server exits the Scanning state, the Remote Provisioning Server Scanning state shall be set to Idle and the Remote Provisioning Scan Items Limit state shall be set to zero.

#### Remote Provisioning Extended Scan procedure

The Remote Provisioning Client may request additional information about an unprovisioned device that is not available in the Unprovisioned Device beacon or in the advertising packets with the Service Data for the «Mesh Provisioning Service» but may be available in the scan response data or additional advertising reports from the same device. For example, the client may request a Device Name or a URI for a device.

Remote Provisioning Server shall support the Remote Provisioning Extended Scan procedure collecting information about a single device and may support executing multiple Remote Provisioning Extended Scan procedures in parallel collecting information about multiple devices at the same time.

**Starting a scan.** The Remote Provisioning server starts executing the Remote Provisioning Extended Scan procedure when it receives a Remote Provisioning Extended Scan Start message with the UUID field present, as specified in Section 4.4.5.5.2.1.

**Collecting information.** While the Remote Provisioning Server executes the Remote Provisioning Extended Scan procedure, it collects information received in a scan response or in advertising reports from the device associated with the specified UUID. To receive scan responses from the unprovisioned devices, the Remote Provisioning Server should perform an active scan (see [2] Vol 6, Part B, Section 4.4.3.2).

The Remote Provisioning Server shall include AD Structures received in the scan response data that match the AD Type in the ADTypeFilter field of the Remote Provisioning Extended Scan Start message.

If the ADTypeFilter field received in the Remote Provisioning Extended Scan Start message contains the Uniform Resource Identifier (URI) AD Type, and URI Hash information is available in the Unprovisioned Device beacon (see Section 3.9.2), then the Remote Provisioning Server shall include ADStructures with URI data that matches the URI Hash information.

If the ADTypeFilter received in the Remote Provisioning Extended Scan Start message contains the Complete Local Name AD Type, the Remote Provisioning Server shall include AD Structure with either the Complete Local Name or the Shortened Local Name if one is available in the scan response data of the unprovisioned device.

**Scan completion.** The Remote Provisioning Extended Scan procedure is considered complete when one of the following conditions is satisfied:

* The Remote Provisioning Server collects AD structures corresponding to all AD Types specified in the ADTypeFilter field of the Remote Provisioning Extended Scan Start message.
* The timeout specified in the Timeout field of the Remote Provisioning Extended Scan Start message expires.
* The ADTypeFilter field of the Remote Provisioning Extended Scan Start message does not contain the URI AD Type and the Remote Provisioning Server receives and processes the scan response data from the device with Device UUID requested in the Remote Provisioning Extended Scan Start message.
* The ADTypeFilter field of the Remote Provisioning Extended Scan Start message contains only the URI AD Type and the Remote Provisioning Server has received advertising report or scan response with the URI corresponding to the URI Hash of the device with Device UUID requested in the Remote Provisioning Extended Scan Start message.
* The ADTypeFilter field of the Remote Provisioning Extended Scan Start message contains the URI AD Type, and at least one different AD Type in the ADTypeFilter field and the Remote Provisioning Server has received advertising report or scan response with the URI corresponding to the URI Hash of the device with Device UUID requested in the Remote Provisioning Extended Scan Start message and the Remote Provisioning Server has received the scan response from the same device.

The Remote Provisioning Server shall save the source address and the security material of the Remote Provisioning Extended Scan Start message and shall use them when sending the Remote Provisioning Extended Scan Report message. When the saved security material is no longer available, the Remote Provisioning Server shall complete the Extended Scanning procedure.

When the Extended Remote Provisioning Scan procedure completes, the Remote Provisioning Server shall send the Remote Provisioning Extended Scan Report message (see Section 4.4.5.5.2.1), which contains obtained data. When OOB Information is unavailable the OOBInformation and AdvStructures fields shall be skipped. When OOB Information is available and the obtained data is empty, the AdvStructures field shall be skipped. The Status field shall be set to Success.

#### Provisioning procedure

The Provisioning procedure is used to provision a device within immediate radio range of the Remote Provisioning Server or to change the Device Key of the Remote Provisioning Server by using the Device Key Refresh procedure.

When entering the Link Opening state, the Remote Provisioning Server shall save the source address and the security material (NetKey Index) of the Remote Provisioning Link Start message and shall use them when sending the Remote Provisioning Link Report, the Remote Provisioning PDU Outbound Report message and the Remote Provisioning PDU Report message.

When the saved security material is no longer available, the Remote Provisioning Server shall do one of the following:

* If a Device Key Refresh procedure is active, the Remote Provisioning Server shall close the Device Key Refresh Interface and then set the Remote Provisioning Link state to Idle.
* If an unprovisioned device is being provisioned, the Remote Provisioning Server shall start the PB-Remote Close Link procedure and then set the Remote Provisioning Link state to Idle.

When the Remote Provisioning Client sends the first Remote Provisioning PDU Send message after the link is opened, it shall set the OutboundPDUNumber field to 1. When the Remote Provisioning Client sends consecutive Remote Provisioning PDU Send messages with a new Provisioning PDU, it shall set the OutboundPDUNumber field to the previous value incremented by 1.

To recover from a transmission error, the Remote Provisioning Client may send the Provisioning PDU Send message again with the OutboundPDUNumber and ProvisioningPDU fields set to the same values as in the previously sent Provisioning PDU Send message.

Figure 4.X illustrates the provisioning behavior of the Remote Provisioning Server model, showing all relevant states of the model, messages processed, procedures, and state transitions that occur based on the procedure outcomes.



Figure 4.X: Remote Provisioning Link state values, message processing, and procedure execution results processing for the Provisioning procedure of the Remote Provisioning Server model

##### Example: Provisioning PDU exchange between a Remote Provisioning Client and Server

The message sequence chart in Figure 4.X illustrates the beginning of the Provisioning PDU exchange between a Remote Provisioning Client and a Remote Provisioning Server. The figure also illustrates how peers recover communication after a transmission error.



Figure 4.2: Example message sequence for the exchange of Provisioning PDUs

#### Behavior

This section describes behaviors for states and messages for the Remote Provisioning Server model.

##### Remote Provisioning Scan behavior

This section describes behaviors of the Remote Provisioning Server model associated with the Remote Provisioning Scan procedure (see Section 4.4.5.2).

*Receiving a Remote Provisioning Scan Capabilities Get message*

When a Remote Provisioning Server receives a Remote Provisioning Scan Capabilities Get message, the Remote Provisioning Server shall respond with a Remote Provisioning Scan Capabilities Status message (see Section 4.3.4.2).

*Sending a Remote Provisioning Scan Capabilities Status message*

A Remote Provisioning Server shall send a Remote Provisioning Scan Capabilities Status message as a response to a Remote Provisioning Scan Capabilities Get message (see Section 4.3.4.1).

When sending a Remote Provisioning Scan Capabilities Status message, the Remote Provisioning Server shall set the message field values as defined in Table 4.23.

| Message Field | State |
| --- | --- |
| MaxScannedItems | Remote Provisioning Max Scanned Items |
| ActiveScan | Remote Provisioning Active Scan |

*Table 4.23 Remote Provisioning Scan* *Capabilities state mapping to fields of the Remote Provisioning Scan Capabilities Status message*

###### Receiving a Remote Provisioning Scan Get message

When a Remote Provisioning Server receives a Remote Provisioning Scan Get message, the Remote Provisioning Server shall respond with a Remote Provisioning Scan Status message (see Section 4.4.5.5.1.6) with the Status field set to Success.

###### Receiving a Remote Provisioning Scan Start message

The Remote Provisioning Client sends a Remote Provisioning Scan Start message to start or restart the Remote Provisioning Scan procedure.

For a Remote Provisioning Scan Start message to be accepted, the values of the message shall meet all conditions defined in Table 4.26.

| Name | Condition |
| --- | --- |
| Items Limit | The value of the ScannedItemsLimit field shall be less than or equal to the value of the Remote Provisioning Max Scanned Items state. |

Table 4.24: Remote Provisioning Scan Start message acceptance condition

When a Remote Provisioning Server receives a Remote Provisioning Scan Start message with values that cannot be accepted, the server shall respond with a Remote Provisioning Scan Status message with the Status field set to Invalid Value.

When the Remote Provisioning Server receives a Remote Provisioning Scan Start message with values that can be accepted (see Table 4.26), and the Remote Provisioning Scan state is Idle, then the Remote Provisioning Server shall start the Remote Provisioning Scan Procedure as defined in Section 4.4.5.2, and shall respond with a Remote Provisioning Scan Status message (see Section 4.4.5.5.1.6) with the Status field set to Success.

When a Remote Provisioning Server receives a Remote Provisioning Scan Start message that can be accepted, and the Remote Provisioning Scan state is equal to either Remote Provisioning Single Device Scan or Remote Provisioning Multiple Devices Scan , and the source address or the security material of the message does not match values saved when the server entered the Scanning state, then the Remote Provisioning Server shall respond with a Remote Provisioning Scan Status message with the Status field set to Invalid State.

When a Remote Provisioning Server receives a Remote Provisioning Scan Start message that can be accepted (see Table 4.26), and the Remote Provisioning Scan state is Scanning, and the source address and the security material of the message match values saved when the server entered the Scanning state, then the Remote Provisioning Server shall restart the Remote Provisioning Scan procedure as defined in Section 4.4.5.2, and shall respond with a Remote Provisioning Scan Status message with the Status field set to Success.

###### Receiving a Remote Provisioning Scan Stop message

When a Remote Provisioning Server receives a Remote Provisioning Scan Stop message, it shall respond with a Remote Provisioning Scan Status message (see Section 4.4.5.5.1.6) with the Status field set to Success.

If the Remote Provisioning Scan procedure was started by the same Remote Provisioning Client that sent the Remote Provisioning Start message, the procedure shall be stopped (see Section 4.4.5.2).

###### Sending a Remote Provisioning Scan Status message

A Remote Provisioning Server shall send the Remote Provisioning Scan Status message in response to a Remote Provisioning Scan Get message, a Remote Provisioning Scan Start message, or a Remote Provisioning Scan Stop message.

The Remote Provisioning Scan Parameters state shall be mapped to the fields of the Remote Provisioning Scan Status message as defined in Table 4.27.

| Field | State |
| --- | --- |
| RPScanningState | Remote Provisioning Scan |
| ScannedItemsLimit | Remote Provisioning Scan Items Limit |
| Timeout | Remote Provisioning Timeout |

Table 4.25 Mapping of Remote Provisioning Scan Parameters state to the fields of a Remote Provisioning Scan Status message.

When a Remote Provisioning Server sends a Remote Provisioning Scan Status message in response to a Remote Provisioning Scan Get message, then the Status field shall be set as defined in Section 4.4.5.5.1.3.

When a Remote Provisioning Server sends a Remote Provisioning Scan Status message in response to a Remote Provisioning Scan Start message, then the Remote Provisioning Server shall set as the message field values as defined in Section 4.4.5.5.1.4.

When a Remote Provisioning Server sends a Remote Provisioning Scan Status message in response to a Remote Provisioning Scan Stop message, then the Remote Provisioning Server shall set the message field values as defined Section 4.4.5.5.1.5.

###### Sending a Remote Provisioning Scan Report message

The Remote Provisioning Scan Report messages are sent as a result of the execution of the Remote Provisioning Scan procedure, defined in Section 4.4.5.2.

The Remote Provisioning Scan Report message shall be sent as a Segmented Access message. When sending the Remote Provisioning Scan Report message the Remote Provisioning Server shall set message fields as following:

* UUID field shall be set to the Device UUID value obtained from the unprovisioned device beacon or from the connectable advertising packet with the Service Data for the «Mesh Provisioning Service» (see Section 7.1.2.2.1).
* OOB field shall be set to the OOB information value.
* RSSI field shall be set to the measured value (see Section 4.3.4.7).

##### Remote Provisioning Extended Scan behavior

This section describes behaviors of the Remote Provisioning Server model associated with the Remote Provisioning Extended Scan procedure (see Section 4.4.5.3).

###### Receiving a Remote Provisioning Extended Scan Start message

The Remote Provisioning Client sends the Remote Provisioning Extended Scan Start message to start the Remote Provisioning Extended Scan procedure for a device with a specific UUID, or to obtain information about the Remote Provisioning Server itself.

When a Remote Provisioning Server receives a Remote Provisioning Extended Scan Start message that does not contain the UUID field (i.e., the request is to obtain the information about the Remote Provisioning Server), it shall respond with a Remote Provisioning Extended Scan Report message with the Status field set to Success (see Section 4.4.5.5.2.2).

When a Remote Provisioning Server receives a Remote Provisioning Extended Scan Start message with the UUID field present (i.e., the request is to execute the Remote Provisioning Extended Scan procedure as described in Section 4.4.5.3), the server shall start a new Remote Provisioning Extended Scan procedure , and it shall send the Remote Provisioning Extended Scan Report at the time when the procedure completes. If the Remote Provisioning Server cannot start a new Remote Provisioning Extended Scan procedure, the server shall respond with a Remote Provisioning Extended Scan Report message with the Status field set to Limited Resources and skipping OOBInformation and AdvStructures fields.

###### Sending a Remote Provisioning Extended Scan Report message

The Remote Provisioning Extended Scan Report message is sent as a result of execution of the Remote Provisioning Extended Scan procedure (see Section 4.4.5.3) or in response to a Remote Provisioning Extended Scan Start message if the Remote Provisioning Server cannot start the Remote Provisioning Extended Scan procedure or if the Remote Provisioning Extended Scan Start message requests the information about the Remote Provisioning Server itself.

The Remote Provisioning Extended Scan Report message shall be sent as a Segmented Access message.

When the Remote Provisioning Server sends a Remote Provisioning Extended Scan Report message in response to a Remote Provisioning Extended Scan Start message, then the Status field shall be set as defined in Section 4.4.5.5.2.1.

When the Remote Provisioning Server sends a Remote Provisioning Extended Scan Report message after executing the Remote Provisioning Extended Scan procedure, then the Status field shall be set as defined in Section 4.4.5.3.

When the Remote Provisioning Extended Scan Report message in sent in response to a Remote Provisioning Extended Scan Start message requesting information about the Remote Provisioning Server itself, the Remote Provisioning Server shall construct the ADStructuresfield based on available local information. The UUID field shall be set to the Device UUID of the Remote Provisioning Server, and the OOBInformation field shall be set to the OOB Information of the Remote Provisioning Server.

When the Remote Provisioning Server sends a Remote Provisioning Extended Scan Report message as a result of executing the Remote Provisioning Extended Scan procedure, then the UUID field shall match the UUID in the corresponding Remote Provisioning Extended Scan Start message; the OOBInformation field shall be set to the OOB Information of the unprovisioned device, if available, and the ADStructures field shall contain AD structures obtained during the Remote Provisioning Extended Scan procedure as defined in Section 4.4.5.3, if available.

##### Provisioning link management behavior

This section describes link management behaviors for the Remote Provisioning Server model.

###### Receiving a Remote Provisioning Link Get message

When a Remote Provisioning Server receives a Remote Provisioning Link Get message, the server shall respond with a Remote Provisioning Link Status message (see Section 4.3.4.13) with the Status field set to Success.

###### Receiving a Remote Provisioning Link Open message

The response to a Remote Provisioning Link Open message is determined by the Remote Provisioning Link state when the message is received.

**When the Remote Provisioning Link state is Idle:** When a Remote Provisioning Server receives a Remote Provisioning Link Open message, and the Remote Provisioning Link state is Idle, then the Remote Provisioning Server shall set the Remote Provisioning Link state to Link Opening, shall set the Remote Provisioning Inbound PDU state to zero, and shall respond with a Remote Provisioning Link Status message (see Section 4.4.5.5.3.4) with the Status field set to Success.

In addition, the server shall execute one of the following behavior sequences based on the presence or absence of the UUID field in the message:

* If the UUID field is present in the Remote Provisioning Link Open message, the Remote Provisioning Server shall set the Remote Provisioning Device UUID state to the value of the UUID field, shall select the PB-GATT or PB-ADV provisioning bearer, and shall start the corresponding PB-Remote Open Link procedure using the value of the UUID field as the Device UUID of the chosen unprovisioned device.
* If the UUID field is absent from the Remote Provisioning Link Open message, the Remote Provisioning Server shall open the Device Key Refresh Interface, shall set the Remote Provisioning Device UUID state to its own Device UUID, shall set the Remote Provisioning Link state to Link Active, and shall send the Remote Provisioning Link Report message (see Section 4.4.5.5.3.5) with the Status field set to Success and without the Reason field.

**When the Remote Provisioning Link state is Link Opening or Link Active:** When a Remote Provisioning Server receives a Remote Provisioning Link Open message, and the Remote Provisioning Link state is either Link Opening or Link Active, the server shall execute one of the following behavior sequences depending on the conditions defined in Table 4.28:

* If all conditions defined in Table 4.28 are met, the Remote Provisioning Server shall respond with a Remote Provisioning Link Status message with the Status field set to Success.
* If one or more conditions defined in Table 4.28 are not met, the Remote Provisioning Server shall respond with a Remote Provisioning Link Status message with the Status field set to Link Cannot Open.

| Condition Name | Condition |
| --- | --- |
| Same Client | ReportAddress field is equal to the saved source address of the Remote Provisioning Link Start message. |
| Same NetKey | NetKeyIndex field is equal to the saved security material from Remote Provisioning Link Start message. |
| Same UUID | UUID field is present and is equal to the Remote Provisioning Device UUID state; or the UUID field is absent, and the Device UUID of the Remote Provisioning Server is equal to the Remote Provisioning Device UUID state. |

Table 4.26. Additional Remote Provisioning Link Open message validation conditions

**When the Remote Provisioning Link state is Link Closing or Outbound Packet Transfer:** When a Remote Provisioning Server receives a Remote Provisioning Link Open message, and the Remote Provisioning Link state is either Link Closing or Outbound Packet Transfer, then the Remote Provisioning Server shall respond with a Remote Provisioning Link Status message with the Status field set to Invalid State.

###### Receiving a Remote Provisioning Link Close message

**When the Remote Provisioning Link state is Idle:** When a Remote Provisioning Server receives a Remote Provisioning Link Close message, and the Remote Provisioning Link state is Idle, then the Remote Provisioning Server shall respond with a Remote Provisioning Link Status message with the Status field set to Success.

**When a Device Key Refresh is active:** When a Remote Provisioning Server receives a Remote Provisioning Link Close message, and the Device Key Refresh procedure is active, and the Remote Provisioning Link state is Link Active, then the Remote Provisioning Server shall close the Device Key Refresh Interface, passing the Reason Code to the layer executing the provisioning protocol; shall respond with a Remote Provisioning Link Status message (see Section 4.4.6.2.3.4) with the Status field set to Success; shall set the Remote Provisioning Link state to Idle; and shall send a Remote Provisioning Link Report message (see Section 4.4.5.5.3.5) with the Status field set to Link Closed by Client.

**When an unprovisioned device is being provisioned:** When the Remote Provisioner Server receives a Remote Provisioning Link Close message, and an unprovisioned device is being provisioned, the server’s response shall be determined by the Remote Provisioning Link state:

* If the Remote Provisioning Link state is Link Active, then the Remote Provisioning Server shall start the PB-Remote Close Link procedure using the value of the Reason field as the Reason, shall set the Link Close Reason state to the Reason field if reason is required for the PB-Remote Close Link procedure, shall set the Link Close Status state to Link Closed by Client, shall set the Remote Provisioning Link state to Link Closing, and shall respond with a Remote Provisioning Link Status message (see Section 4.4.6.2.3.4) with the Status field set to Success.
* If the Remote Provisioning Link state is Link Opening, then the Remote Provisioning Server shall stop the PB-Remote Open Link procedure, shall start the PB-Remote Close Link procedure using the value of the Reason field state as the Reason, shall set the Link Close Reason state to the Reason field if reason is required for the PB-Remote Close Link procedure, shall set the Link Close Status state to Link Closed by Client, shall set the Remote Provisioning Link state to Link Closing, and shall respond with a Remote Provisioning Link Status message with the Status field set to Success.
* If the Remote Provisioning Link state is Outbound Packet Transfer, then the Remote Provisioning Server shall abort all Provisioning Bearer PDU transfers, shall start the PB-Remote Close Link procedure using the value of the Reason field state as the Reason, shall set the Link Close Reason state to the Reason field if reason is required for the PB-Remote Close Link procedure, shall set the Link Close Status state to Link Closed by Client, shall set the Remote Provisioning Link state to Link Closing, and shall respond with a Remote Provisioning Link Status message with the Status field set to Success.

**Remote Provisioning Link state is Link Closing:** When a Remote Provisioning Server receives a Remote Provisioning Link Close message, and the Remote Provisioning Link state is Link Closing, then the Remote Provisioning Server shall respond with a Remote Provisioning Link Status message with the Status field set to Success.

Note. Starting the PB-Remote Close Link procedure initiates additional behavior described in Section 4.4.5.5.3.5.

###### Sending a Remote Provisioning Link Status message

A Remote Provisioning Server shall send the Remote Provisioning Link Status message as a response to a Remote Provisioning Link Open message or a Remote Provisioning Link Close message.

When sending a Remote Provisioning Link Status message in response to a Remote Provisioning Link Open message, the Remote Provisioning Server shall set the RPState field to the current value of the Remote Provisioning Link state and shall set the Status field as defined in Section 4.4.5.5.3.2.

When sending a Remote Provisioning Link Status message in response to a Remote Provisioning Link Close message, the Remote Provisioning Server shall set the RPState field to the current value of the Remote Provisioning Link state and shall set the Status field as defined in Section 4.4.5.5.3.3.

###### Sending a Remote Provisioning Link Report message

When sending a Remote Provisioning Link Report message, the Remote Provisioning Server shall set the RPState field to the current value of the Remote Provisioning Link state. The Remote Provisioning Link Report message shall be sent as a Segmented Access message.

**When the Remote Provisioning** **Link state is Link Opening:** When the Remote Provisioning Link state is Link Opening, and the PB-Remote Open Link procedure succeeds, then the Remote Provisioning Server shall set the Remote Provisioning Link state to Link Active and shall send a Remote Provisioning Link Report message with the Status field set to Success and without the Reason field. If the PB-Remote Open Link procedure fails, then the Remote Provisioning Server shall set the Remote Provisioning Link state to Idle and shall send a Remote Provisioning Link Report message with the Status field set to Link Open Failed and without the Reason field.

**When the Remote Provisioning Link state is Link Active or Outbound Packet:** When the Remote Provisioning Link state is either Link Active or Outbound Packet Transfer, and the unprovisioned device closes the Provisioning Bearer link or the local layer executing the provisioning protocol closes the Device Key Refresh Interface, then the Remote Provisioning Server shall set the Remote Provisioning Link state to Idle, shall abort all Provisioning Bearer PDU transfers, and shall send a Remote Provisioning Link Report message with the Status field set to Link Closed by Device and the Reason field set to the Reason provided by the Provisioning Bearer if applicable.

When the Remote Provisioning Link state is either Link Active or Outbound Packet, and the Remote Provisioning Server encounters a condition resulting in the start of the PB-Remote Close Link procedure, then the Remote Provisioning Server shall set the Remote Provisioning Link state to Link Closing, shall start the PB-Remote Close Link procedure with an appropriate Reason if applicable, shall set the Link Close Reason state to Reason if reason is required for the PB-Remote Close Link procedure, shall set the Link Close Status to Link Closed by Server, and shall abort all Provisioning Bearer PDU transfers.

**When the Remote Provisioning Link state is Link Closing:** When the Remote Provisioning Link state is Link Closing, and the PB-Remote Close Link procedure finishes, then the Remote Provisioning Server shall set the Remote Provisioning Link state to Idle and shall send a Remote Provisioning Link Report message with the Status field set to the Link Close Status state and the Reason field set to the Link Close Reason state if available.

##### Provisioning PDU transfer behavior

This section describes behaviors related to the transfer of Provisioning PDUs for the Remote Provisioning Server model.

###### Receiving a Remote Provisioning PDU Send message

The Provisioner sends provisioning PDUs to the Remote Provisioning Server via a Remote Provisioning PDU Send message to be forwarded to the unprovisioned device or to the Device Key Refresh Interface if the Device Key Refresh procedure is active.

When a Remote Provisioning Server receives a Remote Provisioning PDU Send message, the server’s response shall be determined by the Remote Provisioning Link state:

* If the Remote Provisioning Link state is Link Active, and the value of the OutboundPDUNumber field is equal to the Remote Provisioning Outbound PDU Count state incremented by 1, then the Remote Provisioning Server shall set the Remote Provisioning Link state to Outbound Packet Transfer, and shall start the PB-Remote Send PDU procedure (see Section 5.2.3.3.3) using the value of the ProvisioningPDU field as input parameter. Additionally, if the Device Key Refresh procedure is active, then the Remote Provisioning Server shall send a Remote Provisioning Outbound PDU Report (see Section 4.4.6.2.4.2) as defined in Section 4.4.5.5.4.2.
* If the Remote Provisioning Link state is Link Active, and the value of the OutboundPDUNumber field is not equal to the Remote Provisioning Outbound PDU Count state incremented by 1, then Remote Provisioning Server shall send Remote Provisioning Outbound PDU Report with the OutboundPDUNumber field set to the value of the Remote Provisioning Outbound PDU Count state.
* If the Remote Provisioning Link state is Outbound Packet Transfer, then Remote Provisioning Server shall send Remote Provisioning Outbound PDU Report with the OutboundPDUNumber field set to the value of the Remote Provisioning Outbound PDU Count state.
* If the Remote Provisioning Link state is not Link Active and is not Outbound Packet Transfer, then the Remote Provisioning Server shall ignore the received Remote Provisioning PDU Send message.

###### Sending a Remote Provisioning PDU Outbound Report message

The Remote Provisioning PDU Outbound Report message is sent to report successful transmission of a Provisioning PDU from the Remote Provisioning Server.

When the Remote Provisioning Link state is Outbound Packet Transfer, and the PB-Remote Send PDU procedure succeeds (see Section 5.2.3.3.3), then the Remote Provisioning Server shall increment the value of Remote Provisioning Outbound PDU Count state value by 1, and shall send a Remote Provisioning PDU Outbound Report with the OutboundPDUNumber field set to the value of the Remote Provisioning Outbound PDU Count state.

When the Remote Provisioning Link state is Outbound Packet Transfer, and the PB-Remote Send PDU procedure fails, then the Remote Provisioning Server shall start the PB-Remote Close Link procedure with an appropriate Reason if applicable, shall set the Link Close Reason state to the Reason if reason is required for the PB-Remote Close Link procedure, and shall set the Link Close Status state to Link Closed as Cannot Send PDU.

When the Remote Provisioning Link state is Outbound Packet Transfer, and the delivery of the Remote Provisioning PDU Outbound Report completes successfully, then the Remote Provisioning Server shall set the Remote Provisioning Link state to Link Active.

When the Remote Provisioning Link state is Outbound Packet Transfer, and the delivery of the Remote Provisioning PDU Outbound Report does not complete successfully, then the Remote Provisioning Server shall start the PB-Remote Close Link procedure with an appropriate Reason if applicable, shall set the Link Close Reason state to the Reason if reason is required for the PB-Remote Close Link procedure, and shall set the Link Close Status state to Link Closed as Cannot Deliver PDU Outbound Report.

Note: Starting of the PB-Remote Close Link procedure initiates additional behavior described in Section 4.4.5.2.4.4.

###### Sending a Remote Provisioning PDU Report message

The Remote Provisioning PDU Report message shall be sent as a Segmented Access message.

When the Remote Provisioning Link state is either Link Active or Outbound Packet Transfer, and a new inbound Provisioning PDU has been successfully transferred using the PB-Remote Receive PDU procedure (see Section 5.2.3.3.4), then the Remote Provisioning Server shall increment the Remote Provisioning Inbound PDU Count state value by 1, and shall send a Remote Provisioning PDU Report message with the ProvisioningPDU field set to new Provisioning PDU and the InboundPDUNumber field set to the value of the Remote Provisioning Inbound PDU state.

When the Remote Provisioning Link state is either Link Active or Outbound Packet Transfer, and the PB-Remote Receive PDU procedure fails, then the Remote Provisioning Server shall start the PB-Remote Close Link procedure with an appropriate Reason code if available, shall set the Link Close Reason state to the Reason if reason is required for the PB-Remote Close Link procedure, and shall set the Link Close Status state to Link Closed as Cannot Receive PDU.

When the delivery of the Remote Provisioning PDU Report does not complete successfully, then the Remote Provisioning Server shall start the PB-Remote Close Link procedure with an appropriate Reason code if applicable, shall set the Link Close Reason state to the Reason if reason is required for the PB-Remote Close Link procedure, and shall set the Link Close Status state to Link Closed as Cannot Deliver PDU Report.

Note: Because the provisioning protocol allows two consecutive Provisioning PDUs originating from the new device, but the SAR mechanism does not allow more than one transfer of the Upper Transport PDUs between two nodes (see Section 3.5.3.1), it might be necessary to cache the inbound Provisioning PDU. The behavior of the Remote Provisioning Server model cannot guarantee that only one message at a time will be scheduled to be sent to the client. Therefore, queuing of the messages must be implemented.

### Remote Provisioning Client model

This section describes behaviors relating to the Remote Provisioning Scan Capabilities state, the Remote Provisioning Scan Parameters state, the Remote Provisioning Link state and messages for the Remote Provisioning Server model.

#### Description

The Remote Provisioning Client is a root model (i.e., it does not extend any other models).

This model may be used to represent an element that can control an element of a peer device that exposes a Remote Provisioning Server model (see Section 4.4.5) by sending Remote Provisioning messages (see Section 4.3.4).

If supported, the Remote Provisioning Client model shall be supported by a primary element and may be supported by any secondary elements. The application-layer security on the model shall use the device key.

This model defines the elements and procedures listed in Table 4.29.

| Element | SIG Model ID | Procedure | Messages | **Rx** | **Tx** |
| --- | --- | --- | --- | --- | --- |
| Primary | 0x0005 | Remote Provisioning Scan Capabilities | Remote Provisioning Scan Capabilities Get |  | M |
| Remote Provisioning Scan Capabilities Status | M |  |
| Remote Provisioning Scan Parameters | Remote Provisioning Scan Get |  | M |
| Remote Provisioning Scan Start |  | M |
| Remote Provisioning Scan Stop |  | M |
| Remote Provisioning Scan Status | M |  |
| Remote Provisioning Scan Report | M |  |
| Remote Provisioning Extended Scan Start |  | M |
| Remote Provisioning Scan Extended Report | M |  |
| Remote Provisioning Link Parameters | Remote Provisioning Link Get |  | M |
| Remote Provisioning Link Open |  | M |
| Remote Provisioning Link Close |  | M |
| Remote Provisioning Link Status | M |  |
| Remote Provisioning Link Report | M |  |
| Remote Provisioning PDU Send |  | M |
| Remote Provisioning PDU Outbound Report | M |  |
| Remote Provisioning PDU Report | M |  |

Table 4.27: Remote Provisioning Client elements and procedures

M = Mandatory

#### Behavior

This section describes behaviors for procedures and messages for the Remote Provisioning Client model.

An element can send any Remote Provisioning Client message at any time to query or change a state of a peer element. The element may receive some unsolicited Remote Provisioning Server messages and shall process them and deliver them to the upper layer.

##### Remote Provisioning Scan procedure

This section describes behaviors for managing scans, and getting information about scan capabilities, and state for a Remote Provisioning Scan Server, and about unprovisioned devices eligible for provisioning by the Remote Provisioning Client model.

###### Sending a Remote Provisioning Scan Capabilities Get message

To determine the Remote Provisioning Scan Capabilities state (see Section 4.2.22) of a Remote Provisioning Server, a Remote Provisioning Client shall send a Remote Provisioning Scan Capabilities Get message. The response is a Remote Provisioning Scan Capabilities Status message (see Section 4.3.4.2).

###### Receiving a Remote Provisioning Scan Capabilities Status message

Upon receiving a Remote Provisioning Scan Capabilities Status message, a Remote Provisioning Client can determine the Remote Provisioning Scan Capabilities state (see Section 4.2.22) of a Remote Provisioning Server.

###### Sending a Remote Provisioning Scan Get message

To determine the Remote Provisioning Scan Parameters state (see Section 4.2.23) of a Remote Provisioning Server, a Remote Provisioning Client shall send a Remote Provisioning Scan Get message. The response is a Remote Provisioning Scan Status message (see Section 4.3.4.6).

###### Sending a Remote Provisioning Scan Start message

To start the Remote Provisioning Scan procedure, a Remote Provisioning Client shall send a Remote Provisioning Scan Start message. The response is a Remote Provisioning Scan Status message (see Section 4.3.4.6).

###### Sending a Remote Provisioning Scan Stop message

To stop the Remote Provisioning Scan procedure, a Remote Provisioning Client shall send a Remote Provisioning Scan Stop message. The response is a Remote Provisioning Scan Status message (see Section 4.3.4.6).

###### Receiving a Remote Provisioning Scan Status message

Upon receiving a Remote Provisioning Scan Status message, a Remote Provisioning Client can determine the Remote Provisioning Scan Parameters state (see Section 4.2.23) of a Remote Provisioning Server.

###### Receiving a Remote Provisioning Scan Report message

Upon receiving a Remote Provisioning Scan Report message, a Remote Provisioning Client can determine information about an unprovisioned device within immediate radio range of a Remote Provisioning Server.

##### Remote Provisioning Extended Scan procedure

This section describes behaviors for starting an extended scan and getting Advertisement Data for a scanned device for the Remote Provisioning Client model.

###### Sending a Remote Provisioning Extended Scan Start message

To execute the scan for specific Advertisement Data (see Section 7.1.2.2.1) for a specific device, or to receive Advertisement Data from the Remote Provisioning Server itself, a Remote Provisioning Client shall send a Remote Provisioning Extended Scan Start message. The response is a Remote Provisioning Extended Scan Report message (see Section 4.3.4.9).

###### Receiving a Remote Provisioning Extended Scan Report message

Upon receiving a Remote Provisioning Extended Scan Report message, a Remote Provisioning Client can determine OOB Information and AD values received by a Remote Provisioning Server, originating from the identified device, and matching the AD Types specified in the ADTypeFilter field of the Remote Provisioning Extended Scan Start message.

##### Provisioning link management procedure

This section describes behaviors for the management of Remote Provisioning link for the Remote Provisioning Client model.

###### Sending a Remote Provisioning Link Get message

To determine the Remote Provisioning Link state of a Remote Provisioning Server, a Remote Provisioning Client shall send a Remote Provisioning Link Get message. The response is a Remote Provisioning Link Status message (see Section 4.3.4.13).

###### Sending a Remote Provisioning Link Open message

To start the Device Key Refresh Procedure, the Remote Provisioning Client shall send a Remote Provisioning Link Open message with no UUID field.

To initiate the opening of a Provisioning Bearer link, Remote Provisioning Client shall send a Remote Provisioning Link Open message. The response is a Remote Provisioning Link Status message (see Section 4.3.4.13).

###### Sending a Remote Provisioning Link Close message

To close a Provisioning Bearer link, a Remote Provisioning Client shall send a Remote Provisioning Link Close message. The response is a Remote Provisioning Link Status message (see Section 4.3.4.13).

###### Receiving a Remote Provisioning Link Status message

Upon receiving a Remote Provisioning Link Status message, a Remote Provisioning Client can determine the Remote Provisioning Link state of a Remote Provisioning Server and can determine the result of the associated requesting message, which is indicated in the Status field of the message (see Table 4.22).

###### Receiving a Remote Provisioning Link Report message

Upon receiving a Remote Provisioning Link Report message, a Remote Provisioning Client can determine the Remote Provisioning Link state of a Remote Provisioning Server and can determine the Link Close Status. When the Reason field is present, the Remote Provisioning Client can determine the reason why the provisioning link was closed.

##### Provisioning PDU transfer procedures

This section describes behaviors related to the transfer of Provisioning PDUs for the Remote Provisioning Client model.

###### Sending a Remote Provisioning PDU Send message

To send a Provisioning PDU, a Remote Provisioning Client shall send a Remote Provisioning PDU Send message.

###### Receiving a Remote Provisioning PDU Outbound Report message

Upon receiving a Remote Provisioning PDU Outbound Report message, a Remote Provisioning Client can determine the result of the PB-Remote Send PDU procedure.

###### Receiving a Remote Provisioning PDU Report message

Upon receiving a Remote Provisioning PDU Report message, a Remote Provisioning Client can receive an output of the PB-Remote Receive PDU procedure.

[End of sections insert]

[Section 4.4.5, Summary of SIG Model IDs, becomes Section 4.4.7, Summary of SIG Model IDs.]

### Summary of SIG Model IDs

[In Table 4.127, add table entries for SIG Model IDs: 0x0004, 0x0005, 0x0006, and 0x0007.]

| Model Name | SIG Model ID |
| --- | --- |
| Configuration Server | 0x0000 |
| Configuration Client | 0x0001 |
| Health Server | 0x0002 |
| Health Client | 0x0003 |
| Remote Provisioning Server | 0x0004 |
| Remote Provisioning Client | 0x0005 |

# Provisioning

[New version of the 5 Provisioning section, first four paragraphs are unchanged]

[Move this paragraph above figure 5.1, modified text]

The generic provisioning layer, in Figure 5.2, illustrates how the provisioning PDUs are transmitted as transactions that can be segmented and reassembled. These transactions are sent over a provisioning bearer. The provisioning bearer defines how a session is established for the delivery of transactions from the generic provisioning layer to a single device. Finally, in the bottom layer of the provisioning architecture are the bearers.

[Moved and updated figure 5.1]

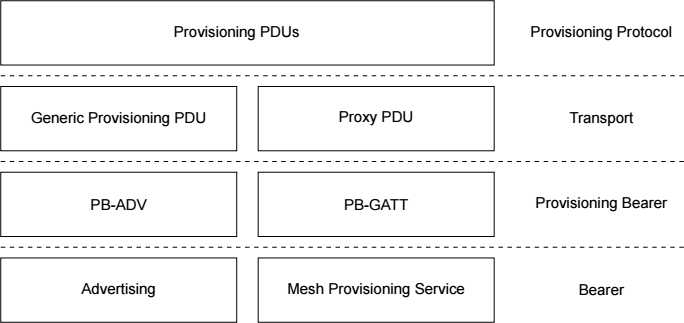


Figure 5.1: Provisioning protocol stack for the PB-ADV and PB-GATT bearers

[Insert new text]

The PB-ADV bearer allows provisioning of a device by using an advertising bearer. The PB-GATT bearer allows provisioning of a device by using the Mesh Provisioning Service.

The PB-Remote bearer allows a Provisioner that is outside immediate radio range of an unprovisioned device to communicate with a node supporting the Remote Provisioning Servermodel that is within immediate radio range of the unprovisioned device and to use that node as a re-transmitter to communicate with the unprovisioned device using PB-ADV or PB-GATT. This allows a Provisioner to provision new devices using nodes of the mesh network. Figure 5.2 illustrates this process when the Remote Provisioning Server uses PB-ADV to provision a new device.

**

Figure 5.2: Devices participating in provisioning using PB-Remote and PB-ADV

The Provisioner may use the Remote Provisioning Server (see Section 4.4.5) to identify unprovisioned devices within immediate radio range of the server (see Section 4.4.6).

[End of new text]

[New version of the 5.2 Provisioning bearer layer section]



## Provisioning bearer layer

A provisioning bearer layer enables the transport of Provisioning PDUs between a Provisioner and an unprovisioned device. Three provisioning bearers are defined:

* PB-ADV (see Section 5.2.1)
* PB-GATT (see Section 5.2.2)
* PB-Remote (see Section 5.2.3)

An unprovisioned device may support PB-ADV and may support PB-GATT. It is strongly recommended to support both PB-ADV and PB-GATT.

A Provisioner shall support either PB-ADV or PB-GATT, or both PB-ADV and PB-GATT. A Provisioner should support PB-ADV. A Provisioner may support PB-Remote.

A node may support PB-Remote.

[Insert the 5.2.3 and 5.2.4 sections after the 5.2.2 section]



### PB-Remote

The PB-Remote provisioning bearer uses the existing mesh network to provision a new device that is not within immediate radio range of the Provisioner. PB-Remote uses the PB-ADV bearer (see Section 5.2.1) or the PB-GATT bearer (see Section 5.2.2) for the last hop to the device. PB-Remote uses one of the mesh nodes as a PB-Remote Provisioning Server to manage PB-ADV or PB-GATT bearer link on behalf of the Provisioner.

PB-Remote may also be used to execute the Device Key Refresh procedure between the Provisioner and the PB-Remote Provisioning Server in order to change the device key of the PB-Remote Provisioning Server node.

Multiple instances of the PB-Remote Provisioning Client model can be used by the Provisioner to communicate with many nodes implementing the PB-Remote Provisioning Server model, thus providing the capability to provision many unprovisioned devices at the same time. The PB-Remote Provisioning Server can only communicate with one PB-Remote Provisioning Client and can only open one supported provisioning bearer at a time.

**

Figure 5.4: A Provisioner using multiple instances of the PB-Remote Provisioning Client

#### 5.2.3.1 PB-Remote Provisioning Client

The PB-Remote Provisioning Client is a provisioning device that controls the provisioning process. The PB-Remote Provisioning Client supports the Remote Provisioning Client model. The PB-Remote Provisioning Client can choose which PB-Remote Provisioning Server device to communicate with, instruct the PB-Remote Provisioning Server to start scanning for unprovisioned devices, and instruct the PB-Remote Provisioning Server to establish a Provisioning Bearer link with the chosen device. After the Provisioning Bearer link is established, the PB-Remote Provisioning Client will run the provisioning protocol by executing the PB-Remote’s Provisioning PDU transfer procedures (see Section 5.5.3.3 and Section 5.5.3.4).

#### 5.2.3.2 PB-Remote Provisioning Server

The PB-Remote Provisioning Server is a mesh node that supports the Remote Provisioning Server model. The PB-Remote Provisioning Server uses the Bluetooth Low Energy scanning mechanism and the Provisioning Bearer protocol to establish a Provisioning Bearer link with an unprovisioned device or a node. After a Provisioning Bearer link is established, the PB-Remote Provisioning Server transports Provisioning PDUs between the PB-Remote Provisioning Client and the connected Provisioning Bearer protocol.

#### 5.2.3.3 PB-Remote procedures

The PB-Remote Provisioning Server may support multiple provisioning bearers, but the server shall use either only one provisioning bearer at a time or execute Device Key Refresh procedure. Each provisioning bearer defines different steps that are needed to open or close the bearer connection and to send or receive the Provisioning PDU over the bearer. The Device Key Refresh procedure also defines it steps (see Section 3.10.8). The subsections below define common names for these steps, thus defining the PB-Remote procedures.

##### 5.2.3.3.1 PB-Remote Open Link procedure

The PB-Remote Link Open procedure is used to establish a connection between the PB-Remote Provisioning Server and the unprovisioned device or start the Device Key Refresh procedure. The PB-Remote Provisioning Server initializes the connection. The procedure accepts the UUID of the device that the provisioning bearer will be open to as a parameter. The procedure can either succeed or fail depending on the outcome of the opening of a provisioning bearer.

Table 5.6 defines Success and Fail results for the PB-Remote Open Link procedure.

| **Provisioning Bearer or procedure** | **Procedure Success** | **Procedure Fail** |
| --- | --- | --- |
| Device Key Refresh procedure | The Device Key Refresh Interface opens successfully (see Section 3.10.8.2). | The Device Key Refresh Interface open fails or the second Device Key Refresh procedure is attempted to start. |
| PB-ADV | The Link Establishment procedure (see Section 5.3.2) establishes a session. | The Link Establishment procedure does not establish a session. |
| PB-GATT | The Provisioning Client successfully opens a connection to the Provisioning Server (see Section 5.2.2). | The Provisioning Client is unable to open a connection to the Provisioning Server. |

Table 5.6: PB-Remote Open Link procedure results

##### 5.2.3.3.2 PB-Remote Close Link procedure

The PB-Remote Link Close procedure is used to close a connection between a PB-Remote Provisioning Server and the unprovisioned device or to stop the Device Key Refresh procedure. The procedure accepts one parameter: Reason. Some provisioning bearers require the Reason parameter to close the connection. This procedure always succeeds. Table 5.7 defines results for the PB-Remote Close Link procedure for each provisioning bearer.

|  |  |
| --- | --- |
| **Provisioning bearer or procedure** | **Procedure** |
| Device Key Refresh procedure | Closes the Device Key Refresh Interface (see Section 3.10.8.2). |
| PB-ADV | Link close step of the Link Establishment procedure (see Section 5.3.2). |
| PB-GATT | Closing a connection between a Provisioning Client and a Provisioning Server (see Section 5.2.2). |

Table 5.7. PB-Remote Close Link procedure results

##### 5.2.3.3.3 PB-Remote Send PDU procedure

The PB-Remote Send PDU procedure is used to send a Provisioning PDU from the PB-Remote Provisioning Server over an open provisioning bearer to the unprovisioned device or to the Device Key Refresh Interface. The procedure accepts one parameter: Provisioning PDU. The procedure can either succeed or fail depending on the outcome of the Provisioning PDU delivery.

Table 5.8 defines the results for the PB-Remote Send PDU procedure.

| **Provisioning Bearer or procedure** | **Procedure Success** | **Procedure Fail** |
| --- | --- | --- |
| Device Key Refresh procedure | The Provisioning PDU is successfully delivered to the Device Key Refresh Interface (see Section 3.10.8.2). | The Provisioning PDU delivery to the Device Key Refresh Interface fails. |
| PB-ADV | The Provisioning PDU is delivered successfully from the Provisioning Server to the unprovisioned device using the PB-ADV provisioning bearer (see Section 5.3.3) | The Provisioning PDU delivery from the Provisioning Server to the unprovisioned device using the PB-ADV bearer fails. |
| PB-GATT | The Provisioning PDU is delivered successfully from the PB-Remote Provisioning Server acting as Provisioning Client to the unprovisioned device (Provisioning Server) using the PB-GATT bearer (see Section 5.2.2). | The Provisioning PDU delivery from PB-Remote Provisioning Server to the unprovisioned device using the PB-GATT bearer fails. |

Table 5.8. PB-Remote Send PDU procedure

##### 5.2.3.3.4 PB-Remote Receive PDU procedure

The PB-Remote Receive PDU procedure is used to receive a Provisioning PDU sent by the PB-Remote Provisioning Server over an open provisioning bearer from the unprovisioned device or from the Device Key Refresh Interface. The procedure can either succeed or fail depending on the outcome of the PDU delivery. The output of the procedure is the transferred Provisioning PDU.

Table 5.9 defines the results for the PB-Remote Receive PDU procedure.

| **Provisioning Bearer or procedure** | **Procedure Success** | **Procedure Fail** |
| --- | --- | --- |
| Device Key Refresh procedure | The Provisioning PDU is successfully delivered from the Device Key Refresh Interface (see Section 3.10.8.2). | The Provisioning PDU delivery via the Device Key Refresh Interface fails. |
| PB-ADV | The Provisioning PDU is delivered successfully from the unprovisioned device to the PB-Remote Provisioning Server using the PB-ADV bearer (see Section 5.3.3) | The Provisioning PDU delivery from the unprovisioned device to the PB-Remote Provisioning Server using the PB-ADV bearer fails. |
| PB-GATT | The Provisioning PDU is delivered successfully from the PB-Remote Provisioning Server acting as Provisioning Client to the unprovisioned device (Provisioning Server) using the PB-GATT bearer (see Section 5.2.2). | The Provisioning PDU delivery from unprovisioned device to the PB-Remote Provisioning Server using the PB-GATT bearer fails. |

Table 5.9. PB-Remote Receive PDU procedure

[End of new text]

[Modify section 5.4.1.10 by adding Invalid Data value to the Error Code]

#### 5.4.1.10 Provisioning Failed

The device sends this PDU if it fails to process a received provisioning protocol PDU. The format of the parameters for this PDU is defined in Table 5.37.

|  |  |  |
| --- | --- | --- |
| Field | Size  (octets) | Notes |
| Error Code | 1 | This represents a specific error in the provisioning protocol encountered by a device. |

Table 5.37: Provisioning Failed PDU parameters format

The Provisioning Error Codes are defined in Table 5.38.

| Value | Name | Description |
| --- | --- | --- |
| 0x00 | Prohibited | Prohibited |
| 0x01 | Invalid PDU | The provisioning protocol PDU is not recognized by the device. |
| 0x02 | Invalid Format | The arguments of the protocol PDUs are outside expected values or the length of the PDU is different than expected. |
| 0x03 | Unexpected PDU | The PDU received was not expected at this moment of the procedure. |
| 0x04 | Confirmation Failed | The computed confirmation value was not successfully verified. |
| 0x05 | Out of Resources | The provisioning protocol cannot be continued due to insufficient resources in the device. |
| 0x06 | Decryption Failed | The Data block was not successfully decrypted. |
| 0x07 | Unexpected Error | An unexpected error occurred that may not be recoverable. |
| 0x08 | Cannot Assign Addresses | The device cannot assign consecutive unicast addresses to all elements. |
| 0x09 | Invalid Data | The Data block contains values that cannot be accepted because of general constraints. |
| 0x0A–0xFF | RFU | Reserved for Future Use |

Table 5.38: Provisioning error codes

# References

Source Specification(s):

1. Bluetooth Mesh Profile Specification Version 1.0
2. Bluetooth Core Specification, Version 5.0 or later