



Discover Zigbee Nodes with Digi Remote Manager®

October 2017

90002254

Contents

Overview	3
Discover Zigbee nodes using the UI.....	3
Discover Zigbee nodes programmatically.....	4
Example: Zigbee discovery	4
Sample Zigbee Discover POST content	4
Sample response from Zigbee Discover command.....	5
Where to find more examples	7

Overview

Digi Remote Manager® provides access to Zigbee networks by means of Zigbee-to-IP gateways. Each gateway has its own "local" PAN (Private Area Network) of Zigbee nodes. Those nodes can join and leave the network at random; for this reason, it is important to know the state of the network when you want to interact with nodes on those networks.

Since Remote Manager has access to many thousands of gateways, it also has access to all of the PANs associated with those gateways. Programmers can fully interact with the networks and individual nodes on those networks using the Remote Manager web services APIs.

Gateways do not automatically update their local PAN status to Remote Manager; therefore, the list of Zigbee nodes maintained by Remote Manager is a cached list. In order to get a view that represents the current state of the remote network(s), you must perform a discovery on those networks. You may do that either from the Remote Manager UI or programmatically via the Remote Manager web services APIs.

NOTE: This tutorial assumes that you have a Remote Manager account and a network that includes Zigbee devices. You can log into your account or sign up for a free trial at <https://remotemanager.digi.com/login.do>.

Discover Zigbee nodes using the UI

Log into Remote Manager and navigate to "Device Management > XBee Networks." You will see a list of all of the Zigbee devices. These include the XBee coordinators in the gateways. (See the "Product Type" column to see the device types.)

The "Device ID" of the gateway appears in the device ID column for any Zigbee node that is associated with that gateway. Note that this is a *cached* view of the network. If you want to see for sure which nodes are associated at that moment, highlight the line that represents the gateway and click "Discover" and check the "Clear the cache on the device..." checkbox. That's the only way to see which nodes might have become un-associated with the network:

The screenshot shows the Digi Remote Manager interface with a table of Zigbee devices. The table has columns for Extended Address, Gateway Device ID, Node ID, Module Type, Product Type, Node Type, Service, and Role. One row is highlighted in green, representing a gateway. A 'Discover' dialog box is open over this row, with the checkbox 'Clear the cache on the device and perform a full rediscover' checked.

Extended Address	Gateway Device ID	Node ID	Module Type	Product Type	Node Type	Service	Role
00:13:A2:00:40:3E:26:1F	00000000-00000000-00409DFF-FF5C525A	Router0	XBee ZB	Wall Router	0x30008		Router
00:13:A2:00:40:48:59:BD	00000000-00000000-00409DFF-FF5C525A	rpm0	XBee ZB	Smart Plug	0x3000f		Router
00:13:A2:00:40:48:5A:43	00000000-00000000-00409DFF-FF5C525A	rpm1	XBee ZB	Smart Plug	0x3000f		Router
00:13:A2:00:40:60:F3:57	00000000-00000000-00409DFF-FF5C525A	North-Wall	XBee ZB	Smart Plug	0x3000f		Router
00:13:A2:00:40:AB:ES:11	00000000-00000000-00409DFF-FF5C525A	Gateway	XBee ZB on SZC	X2e Gateway	0xa001a		Coordinator
00:13:A2:00:40:92:4B:ES	00000000-00000000-00409DFF-FF6A1511		XBee ZB	X4 Gateway	0x30002		Coordinator

NOTE: Performing a discovery without clearing the cache will show you nodes that have joined the network. However, it won't remove nodes that may not be reachable.

Discover Zigbee nodes programmatically

You can use the following web service APIs to discover Zigbee nodes and interact with Zigbee networks:

- /ws/XbeeCore
- /ws/sci
 - RCI

/ws/XbeeCore lets you list all of the Zigbee nodes within your account. This shows which nodes are associated with each gateway. You can specify a gateway ID if you want to narrow the search to nodes associated with just that gateway, or you can return the full list of nodes. For more information on the XbeeCore interface, see the [Digi Remote Manager Programmer Guide](#).

/ws/sci is a general-purpose interface designed for interacting with one or more gateways. SCI (Server Command Interface) is a very powerful interface that allows you to send commands to gateways to have them perform specific functions.

The /ws/sci interface has several operators. The one we will focus on for this example is the **send_message** operator, which lets you interact with gateways using the RCI (Remote Command Interface) API. The SCI request specifies the target(s) and allows you to specify certain options for the request. For example, you can specify the following:

- Whether the request should be synchronous or asynchronous
- Whether it should use the Remote Manager cache or be sent to the gateway(s)

For more information on the SCI interface, see the [Digi Remote Manager Programmer Guide](#).

RCI (Remote Command Interface) requests are wrapped within SCI requests. RCI requests use several commands that allow you to interact directly with the gateway (or list of gateways, simultaneously). These commands let you query settings, set settings, query status and interact with custom Python programs running on the gateway by use of the **do_command**. For more information on the RCI, see the [Digi Remote Manager Programmer Guide](#).

Example: Zigbee discovery

In this example, we will perform the HTTP POST to <https://remotemanager.digi.com/ws/sci>.

Sample Zigbee Discover POST content

```
<sci_request version="1.0">
  <send_message cache="false">
    <targets>
      <device id="00000000-00000000-00409DFF-FF5C525A"/>
    </targets>
    <rci_request version="1.1">
      <do_command target="zigbee">
        <discover option="clear"/>
      </do_command>
    </rci_request>
  </send_message>
</sci_request>
```

Note the following:

1. The **cache="false"** parameter specified as part of the SCI "send_message" command causes Remote Manager to bypass the server cache and send the "send_message" request all the way down to the gateway.
2. The **option="clear"** parameter specified as part of the RCI do_command "discover" causes the gateway to first clear its local cache before performing the Zigbee discovery.
3. As a result, when the response is returned from the gateway, both the gateway's local cache and server-side cache in Remote Manager are updated with the latest list of Zigbee nodes.
4. As with any other SCI command, this action can be submitted against a single gateway or multiple gateways using the "target" options specified under the SCI section of the [Digi Remote Manager Programmer Guide](#).

The response to the discovery request will be a list of all Zigbee nodes discovered and returned by the gateway.

Sample response from Zigbee Discover command

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<result>
  <resultTotalRows>6</resultTotalRows>
  <requestedStartRow>0</requestedStartRow>
  <resultSize>6</resultSize>
  <requestedSize>1000</requestedSize>
  <remainingSize>0</remainingSize>
  <XbeeCore>
    <xpExtAddr>00:13:A2:00:40:3E:26:1F</xpExtAddr>
    <devConnectwareId>00000000-00000000-00409DFF-FF5C525A</devConnectwareId>
    <cstId>2899</cstId>
    <grpId>3560</grpId>
    <xpNetAddr>36098</xpNetAddr>
    <xpNodeType>1</xpNodeType>
    <xpParentAddr>65534</xpParentAddr>
    <xpProfileId>49413</xpProfileId>
    <xpMfgId>4126</xpMfgId>
    <xpDeviceType>196616</xpDeviceType>
    <xpNodeId>Router0</xpNodeId>
    <xpDiscoveryIndex>1</xpDiscoveryIndex>
    <xmtModuleTypeDesc>XBee ZB</xmtModuleTypeDesc>
    <xptProductTypeDesc>Wall Router</xptProductTypeDesc>
    <xpUpdateTime>2017-09-04T15:36:00.000Z</xpUpdateTime>
    <grpPath/>
  </XbeeCore>
  <XbeeCore>
    <xpExtAddr>00:13:A2:00:40:48:59:BD</xpExtAddr>
    <devConnectwareId>00000000-00000000-00409DFF-FF5C525A</devConnectwareId>
    <cstId>2899</cstId>
    <grpId>3560</grpId>
    <xpNetAddr>25346</xpNetAddr>
    <xpNodeType>1</xpNodeType>
    <xpParentAddr>65534</xpParentAddr>
    <xpProfileId>49413</xpProfileId>
    <xpMfgId>4126</xpMfgId>
    <xpDeviceType>196623</xpDeviceType>
    <xpNodeId>rpm0</xpNodeId>
```

```
<xpDiscoveryIndex>4</xpDiscoveryIndex>
<xmtModuleTypeDesc>XBee ZB</xmtModuleTypeDesc>
<xptProductTypeDesc>Smart Plug</xptProductTypeDesc>
<xpUpdateTime>2017-09-04T15:36:00.000Z</xpUpdateTime>
<grpPath/>
</XbeeCore>
<XbeeCore>
  <xpExtAddr>00:13:A2:00:40:48:5A:43</xpExtAddr>
  <devConnectwareId>00000000-00000000-00409DFF-FF5C525A</devConnectwareId>
  <cstId>2899</cstId>
  <grpId>3560</grpId>
  <xpNetAddr>8341</xpNetAddr>
  <xpNodeType>1</xpNodeType>
  <xpParentAddr>65534</xpParentAddr>
  <xpProfileId>49413</xpProfileId>
  <xpMfgId>4126</xpMfgId>
  <xpDeviceType>196623</xpDeviceType>
  <xpNodeId>rpm1</xpNodeId>
  <xpDiscoveryIndex>2</xpDiscoveryIndex>
  <xmtModuleTypeDesc>XBee ZB</xmtModuleTypeDesc>
  <xptProductTypeDesc>Smart Plug</xptProductTypeDesc>
  <xpUpdateTime>2017-09-04T15:36:00.000Z</xpUpdateTime>
  <grpPath/>
</XbeeCore>
<XbeeCore>
  <xpExtAddr>00:13:A2:00:40:60:F3:57</xpExtAddr>
  <devConnectwareId>00000000-00000000-00409DFF-FF5C525A</devConnectwareId>
  <cstId>2899</cstId>
  <grpId>3560</grpId>
  <xpNetAddr>57990</xpNetAddr>
  <xpNodeType>1</xpNodeType>
  <xpParentAddr>65534</xpParentAddr>
  <xpProfileId>49413</xpProfileId>
  <xpMfgId>4126</xpMfgId>
  <xpDeviceType>196623</xpDeviceType>
  <xpNodeId>North-Wall</xpNodeId>
  <xpDiscoveryIndex>5</xpDiscoveryIndex>
  <xmtModuleTypeDesc>XBee ZB</xmtModuleTypeDesc>
  <xptProductTypeDesc>Smart Plug</xptProductTypeDesc>
  <xpUpdateTime>2017-09-04T15:36:00.000Z</xpUpdateTime>
  <grpPath/>
</XbeeCore>
<XbeeCore>
  <xpExtAddr>00:13:A2:00:40:92:4B:E5</xpExtAddr>
  <devConnectwareId>00000000-00000000-00409DFF-FF6A1511</devConnectwareId>
  <cstId>2899</cstId>
  <grpId>3560</grpId>
  <xpNetAddr>0</xpNetAddr>
  <xpNodeType>0</xpNodeType>
  <xpParentAddr>65534</xpParentAddr>
  <xpProfileId>49413</xpProfileId>
  <xpMfgId>4126</xpMfgId>
  <xpDeviceType>196610</xpDeviceType>
  <xpDiscoveryIndex>1</xpDiscoveryIndex>
  <xmtModuleTypeDesc>XBee ZB</xmtModuleTypeDesc>
  <xptProductTypeDesc>X4 Gateway</xptProductTypeDesc>
  <xpUpdateTime>2017-09-04T14:58:00.000Z</xpUpdateTime>
```

```
<grpPath/>
</XbeeCore>
<XbeeCore>
  <xpExtAddr>00:13:A2:00:40:AB:E5:11</xpExtAddr>
  <devConnectwareId>00000000-00000000-00409DFF-FF5C525A</devConnectwareId>
  <cstId>2899</cstId>
  <grpId>3560</grpId>
  <xpNetAddr>0</xpNetAddr>
  <xpNodeType>0</xpNodeType>
  <xpParentAddr>65534</xpParentAddr>
  <xpProfileId>49413</xpProfileId>
  <xpMfgId>4126</xpMfgId>
  <xpDeviceType>655386</xpDeviceType>
  <xpNodeId>Gateway</xpNodeId>
  <xpDiscoveryIndex>3</xpDiscoveryIndex>
  <xmtModuleTypeDesc>XBee ZB on S2C</xmtModuleTypeDesc>
  <xptProductTypeDesc>X2e Gateway</xptProductTypeDesc>
  <xpUpdateTime>2017-09-04T15:36:00.000Z</xpUpdateTime>
  <grpPath/>
</XbeeCore>
</result>
```

Where to find more examples

For more information and examples, see the **API Explorer** tab under the **Documentation** section of the Remote Manager UI:

SCI Targets Examples Export Send Clear

Path: /ws/sci

HTTP Method: GET

```
1 <sci_request>
2   <send_message>
3     <target>
4       <device>
5     </target>
6   </send_message>
7 </sci_request>
```

- Alarm
- Carrier
- DeviceCore
- FileData
- SCI
 - v1/devices
 - Scheduled Operation
 - v1/health_configs
 - v1/streams
 - v1/monitors
 - XbeeCore
 - Smart Energy
 - Data Streams
 - Push and Query Monitor
 - Embedded
- Asynchronous
- Cached request
- Data Service
- Device Tag Target
- Disconnect device
- File System Service
- Firmware
- Iridium
- Python Callback
- RCI
 - Descriptors
 - DIA
 - File System
 - Query device settings
 - Query device state
 - Set device setting
- Reboot device
- Run command on device
- SM/UDP
- SMS
- Uncached request
- XBee
 - Discover nodes
 - Query node settings
 - Query node state
 - Set node setting