

# API GUIDELINE

---

## Pos Digicert Organization Roaming Certificate Web API (e-Invoice)



## REVISION CONTROL AND CHANGE HISTORY

Revision Number	Approval Date	Approved By
Revision 1.0	20 <sup>th</sup> May 2024	Nazril Bin Mohd Ghani
Revision 1.1	5 <sup>th</sup> August 2024	Nazril Bin Mohd Ghani
Revision 1.2	6 <sup>th</sup> August 2024	Nazril Bin Mohd Ghani

## **Disclaimer**

The information in this document is intended only for the person or entity to which it is addressed and may contain confidential and/or privileged material. Any review, retransmission, dissemination, or other use of, or taking of any action in reliance upon this information by persons or entities other than the intended recipient is prohibited. If you received this document in error, please contact the sender and delete the material from all computers or medium.

## **Ownership**

This is the intellectual property of Pos Digicert Sdn Bhd (Pos Digicert) and all its components belongs to Pos Digicert, located in Star Central, Cyberjaya, Malaysia.

## **Sample Code**

The sample code in this document is provided “AS IS” and any express or implied warranties, including the implied warranties of merchantability and fitness for a particular purpose are disclaimed. We voluntarily put forth our effort to enhance the industry by providing this sample code and Pos Digicert will not be held liable for any direct, indirect, incidental, special, exemplary, or consequential damages (including, but not limited to, procurement of substitute goods or services; loss of use, data, or profits; or business interruption) sustained by the client or a third party, however caused and on any theory of liability, whether in contract, strict liability, or tort arising in any way out of the use of this sample code, even if advised of the possibility of such damage.

## **Feedback**

To get further clarification on the API/web service or the usage of the API/web service, email can be sent to [invoice@posdigicert.com.my](mailto:invoice@posdigicert.com.my)

## **Confidentiality Statement**

© Pos Digicert Sdn Bhd

This document is the property of Pos Digicert and no part thereof shall be reproduced, stored in a retrieval system, or transmitted in any form or by any means, electronic, mechanical,

photocopying, recording or otherwise without written approval from the management of Pos Digicert.

## Contents

1	Introduction .....	5
2	Development Information .....	5
2.1	Staging Environment .....	5
2.2	Production Environment .....	5
3	API: Signing .....	6
3.1	Verify Certificate Status .....	6
3.1.1	verifyRoamingCert: Request Parameter .....	6
3.1.2	verifyRoamingCert: Response Parameter and Status Code .....	6
3.1.3	Sample Code .....	7
3.2	Signing .....	11
3.2.1	signHash: Request Parameter .....	11
3.2.2	signHash: Response Parameter and Status Code .....	11
3.2.3	Sample Code .....	12
3.3	Find Certificate .....	15
3.3.1	findCert: Request Parameter .....	15
3.3.2	findCert: Response Parameter and Status Code .....	15
3.3.3	Sample Code .....	16
4	Certificate Details .....	19

## 1 Introduction

This document is the API guide for Organization Roaming Certificate.

Organization Roaming Certificate is centralized, managed and stored in the cloud, rather than stored in local device or server located at the client's data centre. The signer certificate is associated with corresponding private key which is protected by hardware security module (HSM) with FIPS-140 Level 3 certified. Using the HSM, private key is protected from being copied and having the risk of illegal use by someone to impersonate the organization to sign on behalf of the organization.

The signer certificate is generated by Pos Digicert as licensed Certification Authority and assigned to the organization that would use the certificate for E-Invoice. The certificate follows requirement set forth by Inland Revenue Board of Malaysia (IRBM).

In order to access and use the private key to sign the documents for E-Invoice, the Service Provider system or other system such as ERP software that will utilize E-Invoice SDK, prior to calling the E-Invoice SDK, must call the designated Organization Roaming Certificate web service as described in this document.

## 2 Development Information

Please refer below for development information.

### 2.1 Staging Environment

ITEM	VALUE
MYCRS Project Code	Beta-MYCRS_IL7NQ138

### 2.2 Production Environment

ITEM	VALUE
MYCRS Project Code	MYCRS_LH4N3V01C3

Sample code including Postman Json file could be downloaded from:

URL	<a href="https://posdigicertsupport.freshdesk.com/a/solutions/articles/69000858077/edit?lang=en">https://posdigicertsupport.freshdesk.com/a/solutions/articles/69000858077/edit?lang=en</a>
ZIP password	MYCRS_TCNRYYC5!#\$

### 3 API: Signing

This section will describe API for Signing.

#### 3.1 Verify Certificate Status

<b>PROTOCOL / METHOD</b>	HTTPS / POST
<b>URL (Staging)</b>	https://demo-kit.posdigicert.com.my/DssManagerApi/rest/roaming_hsm/verifyRoamingCert
<b>URL (Production)</b>	https://dss.posdigicert.com.my/DssManagerApi/rest/roaming_hsm/verifyRoamingCert
<b>Request Content-type</b>	x-www-form-urlencoded
<b>Response Content-type</b>	application/json
<b>PURPOSE</b>	To check roaming certificate existence. <b>Revoked Certificate</b> and <b>Expired Certificate</b> are same as certificate not available.

##### 3.1.1 verifyRoamingCert: Request Parameter

REQUEST			
NAME	TYPE	LENGTH	DESCRIPTION / RULE
pCode	String	50	Mandatory. Project Code. Value for Staging and Production is different.
userID	String	15	Mandatory. Business Registration Number (BRN) when applying the roaming certificate.
orgID	String	15	Mandatory. Business Registration Number (BRN) when applying the roaming certificate.

##### 3.1.2 verifyRoamingCert: Response Parameter and Status Code

RESPONSE										
NAME	TYPE	DESCRIPTION / RULE								
statusCode	String	-Refer below-								
certificate	String	Certificate details for information only. Sample return value: <pre>{     "root":     "MIJpDCCBYygAwIBAgIQSmYDwi&lt;TRUNCATED&gt;",     "certificate": "MIIFyTCCA7GgAwIBAgI&lt;TRUNCATED&gt;",     "intermediate": "MIICjCCA/KgAwIBAg&lt;TRUNCATED&gt;",     "certificate":     "MIIFyTCCA7GgAwIBAgIQW0ysoa&lt;TRUNCATED&gt;",     "statusCode": "901"   }</pre> <table border="1"> <thead> <tr> <th>Value</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td>root</td> <td>Root CA certificate</td> </tr> <tr> <td>certificate</td> <td>Signer / User certificate</td> </tr> <tr> <td>intermediate</td> <td>Intermediate CA certificate</td> </tr> </tbody> </table>	Value	Description	root	Root CA certificate	certificate	Signer / User certificate	intermediate	Intermediate CA certificate
Value	Description									
root	Root CA certificate									
certificate	Signer / User certificate									
intermediate	Intermediate CA certificate									
STATUS CODE										

RESPONSE		
NAME	TYPE	DESCRIPTION / RULE
901		Success
800		Operation Roaming Failed
804		Certificate PIN is Blocked
805		User Not Exist (Note: If there were incorrect passing parameter values. The same 805 code will be returned)
808		Certificate is Revoked

### 3.1.3 Sample Code

```

*****
*****
* THIS SAMPLE CODE COMES WITHOUT ANY WARRANTY.
*
* YOU ARE FREE TO ALTER AT ANY TIME BASED ON YOUR SYSTEM REQUIREMENT,
* WHERE APPROPRIATE.
*
* THE SAMPLE CODE IS BASED ON JAVA 1.8 USING STANDARD JAVA LIBRARY.
* NO ANY THIRD-PARTY LIBRARY IS REQUIRED TO RUN THIS SAMPLE CODE.
*
* YOU MIGHT WANT TO USE THIRD PARTY LIBRARIES TO EASE DEVELOPMENT,
* IN ADDIITON TO THIS SAMPLE CODE.

*****
*****/

import java.io.BufferedReader;
import java.io.DataOutputStream;
import java.io.IOException;
import java.io.InputStreamReader;
import java.io.UnsupportedEncodingException;
import java.net.URL;
import java.net.URLEncoder;

import javax.net.ssl.HttpURLConnection;

public class VerifyCertificateStatus {

    public static void main(String[] args) {

        String id = "199801001482";
        String orgID = "199801001482";
        String projectID = "Beta-MYCRS_IL7NQ138";

        /**
        *****
        *****
        * prepare parameters

```

```

*
*****
****
*/

try {
    id = URLEncoder.encode(id, "UTF-8");
    orgID = URLEncoder.encode(orgID, "UTF-8");

} catch (UnsupportedEncodingException e) {
    e.printStackTrace();
}

String paramString = "pCode=" + projectID + "&" + "userID=" + id + "&" + "orgID=" + orgID;
System.out.println(paramString);
String URL = "https://demo-
kit.posdigicert.com.my/DssManagerApi/rest/roaming_hsm/verifyRoamingCert";

int responseCode = 0;
String res = "";

/**
*****
****
* open connection
*
*****
****
*/

URL url = null;
URLConnection con = null;
try {
    url = new URL(URL);
    con = (URLConnection) url.openConnection();

//set POST method
con.setRequestMethod("POST");
}
catch (IOException ioe) {
    ioe.printStackTrace();
}

/**
*****
****
* set Request header
*
*****
****
*/

```



```
con.setRequestProperty("User-Agent", "Java");
con.setRequestProperty("Accept-Language", "en-US,en;q=0.5");
con.setRequestProperty("Content-Type", "application/x-www-form-urlencoded");

con.setDoOutput(true);
DataOutputStream wr;
try {
    wr = new DataOutputStream(con.getOutputStream());
    wr.writeBytes(paramString);
    wr.flush();
    wr.close();

    //get HTTP response code
    responseCode = con.getResponseCode();

} catch (IOException e) {
    e.printStackTrace();
}

System.out.println("HTTP Response Code : " + responseCode);

/**
*****
* get response form server
*
*****
*****/
if (responseCode == 200) {
    System.out.println("Output from Server .... \n");

    BufferedReader br = null;
    try {
        br = new BufferedReader(new InputStreamReader(
            con.getInputStream()));
    } catch (IOException e) {
        e.printStackTrace();
    }

    String output;

    try {
        while ((output = br.readLine()) != null) {
            res = res + output;
        }
    } catch (IOException e) {
        e.printStackTrace();
    }
}
```

```
con.disconnect();
}

System.out.println(res);

/**
 * parse the json string using any 3rd party Json library i.e GSON, etc.
 *
 * statusCode is returned status code. 901 = success. Then get the cert chain as below.
 * "root" is root cert in base64 format.
 * "intermediate" is intermediate cert in base64 format.
 * "endUserCertificate" is user cert in base64 format.
 */

}

}
```

(this space is intentionally left blank)

### 3.2 Signing

<b>PROTOCOL / METHOD</b>	HTTPS / POST
<b>URL (Staging)</b>	https://demo-kit.posdigicert.com.my/DssManagerApi/rest/roaming_hsm/signHash256RSA
<b>URL (Production)</b>	https://dss.posdigicert.com.my/DssManagerApi/rest/roaming_hsm/signHash256RSA
<b>Request Content-type</b>	x-www-form-urlencoded
<b>Response Content-type</b>	application/json
<b>PURPOSE</b>	To get signed hash of XML document in SHA256RSA algorithm

#### 3.2.1 signHash: Request Parameter

REQUEST			
NAME	TYPE	LENGTH	DESCRIPTION / RULE
pCode	String	50	Mandatory. Project Code. Value for Staging and Production is different.
userID	String	15	Mandatory. Business Registration Number (BRN) when applying the roaming certificate.
orgID	String	15	Mandatory. Business Registration Number (BRN) when applying the roaming certificate.
data	String	N/A	Mandatory. Document hash to be signed. In Base64 string.
pin	String	16	Mandatory. The certificate PIN (password).

#### 3.2.2 signHash: Response Parameter and Status Code

RESPONSE		
NAME	TYPE	DESCRIPTION / RULE
statusCode	String	-Refer below-
signedData	String	Signed hash in Base64 format.
STATUS CODE		
901	Success	
902	Parameter incomplete	
800	Operation Roaming Failed	
802	PIN is Blocked	
804	Invalid PIN	
805	User Not Exist	
808	Cert Revoked	

### 3.2.3 Sample Code

```

*****
* THIS SAMPLE CODE COMES WITHOUT ANY WARRANTY.
*
* YOU ARE FREE TO ALTER AT ANY TIME BASED ON YOUR SYSTEM REQUIREMENT,
* WHERE APPROPRIATE.
*
* THE SAMPLE CODE IS BASED ON JAVA 1.8 USING STANDARD JAVA LIBRARY.
* NO ANY THIRD PARTY LIBRARY IS REQUIRED TO RUN THIS SAMPLE CODE.
*
* YOU MIGHT WANT TO USE THIRD PARTY LIBRARIES TO EASE DEVELOPMENT,
* IN ADDIITON TO THIS SAMPLE CODE.
*****/
import java.io.BufferedReader;
import java.io.DataOutputStream;
import java.io.IOException;
import java.io.InputStreamReader;
import java.io.UnsupportedEncodingException;
import java.net.URL;
import java.net.URLEncoder;

import javax.net.ssl.HttpURLConnection;

public class SignRoaming {

    public static void main(String[] args) {

        String id = "199801001482";
        String orgID = "199801001482";
        String pin = "UuRspACP ";
        String projectID = "Beta-MYCRS_IL7NQ138";
        String data = "<xml>data</xml>";

        /**
        *****
        * prepare parameters
        *
        *****
        */

        try {
            pin = URLEncoder.encode(pin, "UTF-8");
            data = URLEncoder.encode(data, "UTF-8");

        } catch (UnsupportedEncodingException e) {
            e.printStackTrace();
        }

        String paramString = "pCode=" + projectID + "&" + "userID=" + id + "&" + "orgID=" + orgID + "&" +
"data=" + data + "&" + "pin=" + pin;

```

```

String URL = "https://demo-
kit.posdigicert.com.my/DssManagerApi/rest/roaming_hsm/signHash256RSA";

int responseCode = 0;
String res = "";

/**
*****
* open connection
*
*****
*/

URL url = null;
URLConnection con = null;
try {
    url = new URL(URL);
    con = (URLConnection) url.openConnection();

    //set POST method
    con.setRequestMethod("POST");
}
catch (IOException ioe) {
    ioe.printStackTrace();
}

/**
*****
* set Request header
*
*****
*/

con.setRequestProperty("User-Agent", "Java");
con.setRequestProperty("Accept-Language", "en-US,en;q=0.5");
con.setRequestProperty("Content-Type", "application/x-www-form-urlencoded");

con.setDoOutput(true);
DataOutputStream wr;
try {
    wr = new DataOutputStream(con.getOutputStream());
    wr.writeBytes(paramString);
    wr.flush();
    wr.close();

    //get HTTP response code
    responseCode = con.getResponseCode();

} catch (IOException e) {
    e.printStackTrace();
}

```

```
System.out.println("HTTP Response Code : " + responseCode);

/**
*****
* get response form server
*
*****
*/
if (responseCode == 200) {
    System.out.println("Output from Server .... \n");

    BufferedReader br = null;
    try {
        br = new BufferedReader(new InputStreamReader(
            (con.getInputStream())));
    } catch (IOException e) {
        e.printStackTrace();
    }

    String output;

    try {
        while ((output = br.readLine()) != null) {
            res = res + output;
        }
    } catch (IOException e) {
        e.printStackTrace();
    }

    con.disconnect();
}

System.out.println(res);

/**
* parse the json string using any 3rd party Json library i.e GSON, etc.
*
* statusCode is returned status code. 901 = success. Then get the signedData.
* signedData is in base64 format. raw hash signature using sha256
*
*/

}

}
```

### 3.3 Find Certificate

<b>PROTOCOL / METHOD</b>	HTTPS / POST
<b>URL (Staging)</b>	<a href="https://demo-kit.posdigicert.com.my/DssManagerApi/rest/roaming_hsm/findCert">https://demo-kit.posdigicert.com.my/DssManagerApi/rest/roaming_hsm/findCert</a>
<b>URL (Production)</b>	<a href="https://dss.posdigicert.com.my/DssManagerApi/rest/roaming_hsm/findCert">https://dss.posdigicert.com.my/DssManagerApi/rest/roaming_hsm/findCert</a>
<b>Request Content-type</b>	x-www-form-urlencoded
<b>Response Content-type</b>	application/json
<b>PURPOSE</b>	To get the whole certificate chain including end user certificate.

#### 3.3.1 findCert: Request Parameter

REQUEST			
NAME	TYPE	LENGTH	DESCRIPTION / RULE
pCode	String	50	Mandatory. Project Code. Value for Staging and Production is different.
userID	String	15	Mandatory. Business Registration Number (BRN) when applying the roaming certificate.
orgID	String	15	Mandatory. Business Registration Number (BRN) when applying the roaming certificate.

#### 3.3.2 findCert: Response Parameter and Status Code

RESPONSE		
NAME	TYPE	DESCRIPTION / RULE
statusCode	String	-Refer below-
root	String	Root certificate in Base64.
intermediate	String	Intermediate certificate in Base64.
endUserCertificate	String	User certificate in Base64.
STATUS CODE		
902	Parameter incomplete.	
800	Operation Roaming Failed	
805	User Do Not Exist	

(this space is intentionally left blank)

### 3.3.3 Sample Code

```
/*
 * THIS SAMPLE CODE COMES WITHOUT ANY WARRANTY.
 *
 * YOU ARE FREE TO ALTER AT ANY TIME BASED ON YOUR SYSTEM REQUIREMENT,
 * WHERE APPROPRIATE.
 *
 * THE SAMPLE CODE IS BASED ON JAVA 1.8 USING STANDARD JAVA LIBRARY.
 * NO ANY THIRD PARTY LIBRARY IS REQUIRED TO RUN THIS SAMPLE CODE.
 *
 * YOU MIGHT WANT TO USE THIRD PARTY LIBRARIES TO EASE DEVELOPMENT,
 * IN ADDIITON TO THIS SAMPLE CODE.
 */

import java.io.BufferedReader;
import java.io.DataOutputStream;
import java.io.IOException;
import java.io.InputStreamReader;
import java.io.UnsupportedEncodingException;
import java.net.URL;
import java.net.URLEncoder;

import javax.net.ssl.HttpURLConnection;

public class FindCert {

    public static void main(String[] args) {

        String id = "199801001482";
        String orgID = "199801001482";
        String projectID = "Beta-MYCRS_IL7NQ138";

        /**
         * prepare parameters
         */

        try {
            id = URLEncoder.encode(id, "UTF-8");
            orgID = URLEncoder.encode(orgID, "UTF-8");

        } catch (UnsupportedEncodingException e) {
            e.printStackTrace();
        }

        String paramString = "pCode=" + projectID + "&" + "userID=" + id + "&" + "orgID=" + orgID;

        String URL = "https://demo-kit.posdigicert.com.my/DssManagerApi/rest/roaming_hsm/findCert";
    }
}
```



```
int responseCode = 0;
String res = "";

/**
*****
 * open connection
 *
*****
 */

URL url = null;
HttpsURLConnection con = null;
try {
    url = new URL(URL);
    con = (HttpsURLConnection) url.openConnection();

    //set POST method
    con.setRequestMethod("POST");
}
catch (IOException ioe) {
    ioe.printStackTrace();
}

/**
*****
 * set Request header
 *
*****
 */

con.setRequestProperty("User-Agent", "Java");
con.setRequestProperty("Accept-Language", "en-US,en;q=0.5");
con.setRequestProperty("Content-Type", "application/x-www-form-urlencoded");

con.setDoOutput(true);
DataOutputStream wr;
try {
    wr = new DataOutputStream(con.getOutputStream());
    wr.writeBytes(paramString);
    wr.flush();
    wr.close();

    //get HTTP response code
    responseCode = con.getResponseCode();

} catch (IOException e) {
    e.printStackTrace();
}

System.out.println("HTTP Response Code : " + responseCode);
```

```
/**
*****
* get response form server
*
*****
*/
if (responseCode == 200) {
    System.out.println("Output from Server .... \n");

    BufferedReader br = null;
    try {
        br = new BufferedReader(new InputStreamReader(
            (con.getInputStream())));
    } catch (IOException e) {
        e.printStackTrace();
    }

    String output;

    try {
        while ((output = br.readLine()) != null) {
            res = res + output;
        }
    } catch (IOException e) {
        e.printStackTrace();
    }

    con.disconnect();
}

System.out.println(res);

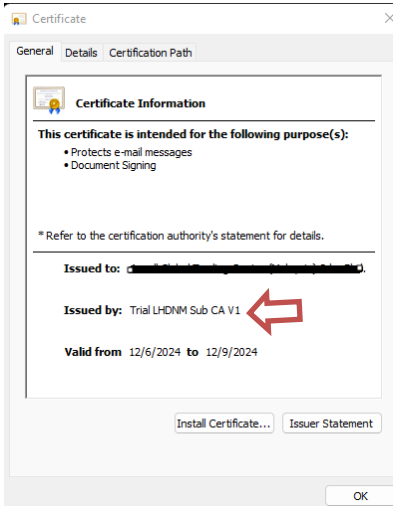
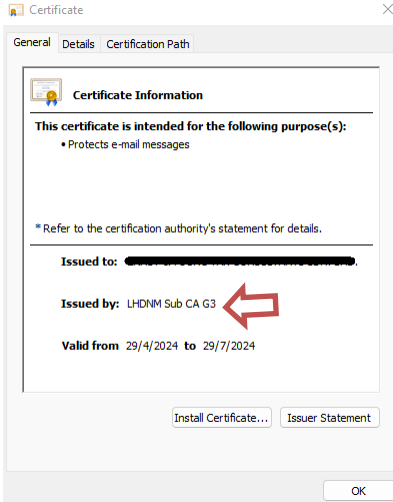
/**
* parse the json string using any 3rd party Json library i.e GSON, etc.
*
* statusCode is returned status code. 901 = success. Then get the cert chain as below.
* "root" is root cert in base64 format.
* "intermediate" is intermediate cert in base64 format.
* "endUserCertificate" is user cert in base64 format.
*/

}

}
```

## 4 Certificate Details

Below are the details to differentiate the roaming certificate for Sandbox / Preprod and Production environment:

Environment	Sandbox / Preprod	Production
<b>Pos Digicert Certificate Chain</b>	<p><b>Intermediate:</b>                      CN = Trial LHDNM Sub CA V1                      OU = Terms of use at                      http://www.posdigicert.com.my                      O = LHDNM                      C = MY</p> <p><b>Root:</b>                      CN = Trial Pos Digicert Class 2                      Root CA V1                      OU = 457608-K                      O = Pos Digicert Sdn. Bhd.                      C = MY</p> <p><b>How to identify:</b></p> 	<p><b>Intermediate:</b>                      CN = LHDNM Sub CA G3                      OU = Terms of use at                      http://www.posdigicert.com.my                      O = LHDNM                      C = MY</p> <p><b>Root:</b>                      CN = Pos Digicert Class 2 Root CA G3                      OU = 457608-K                      O = Pos Digicert Sdn. Bhd.                      C = MY</p> <p><b>How to identify:</b></p> 
<b>Roaming URL</b>	https://demo-kit.posdigicert.com.my/DssManagerApi/rest/roaming_hsm/verifyRoamingCert	https://dss.posdigicert.com.my/DssManagerApi/rest/roaming_hsm/verifyRoamingCert
<b>IRBM E-INVOICE API BASE URL</b>	preprod-api.myinvois.hasil.gov.my	api.myinvois.hasil.gov.my

(End of Document)