

RD Client Integration Document

CURRENT VERSION DETAIL

Project Name	Windows RD service Solution
Project Version	2.0.1.34

Prepared By	Date Prepared
Manish Pandey	10 th Jan 2020
Approved By	Date Approved
Udita Singh	10 th Jan 2020
Comments	

References

RD Client Integration Document

DOC Ver.	Description of Changes	Prepared By	Date Prepared	Reviewed By	Date Reviewed	Approved By	Date Approved
1.0	1 st Beta release	Tanuj Joshi	23 May 2017	Udita Singh			
1.1	1 st Beta release	Tanuj Joshi	24 May 2017	Udita Singh	24-May-17	Udita Singh	24-May-17
1.2	Input and output data details added	Udita Singh	30-May-17	Udita Singh	30-May-17	Udita Singh	30-May-17
1.3	Integration with Mozilla Firefox and Google Chrome	Abhishek Gupta	12 June 2017	Udita Singh	12 June 2017	Udita Singh	12 June 2017
1.5	This release is tested successfully on Chrome Only	Manish Pandey	4 th July 2017	Udita Singh	4 th July 2017	Udita Singh	4 th July 2017
1.6	This release is tested successfully on Chrome Only	Manish Pandey	5 th July 2017	Udita Singh	5 th July 2017	Udita Singh	5 th July 2017
1.7	This release is tested successfully on Chrome Only	Abhishek Gupta	13 th July 2017	Udita Singh	13 th July 2017	Udita Singh	13 th July 2017
1.8	This release is tested successfully on Chrome Only	Abhishek Gupta	22 nd July 2017	Udita Singh	22 nd July 2017	Udita Singh	22 nd July 2017
1.9	This release is tested successfully on IE	Manish Pandey	31 st July 2017	Udita Singh	31 st July 2017	Udita Singh	31 st July 2017
1.10	White listed functionality has been added. Increase Response Timeout	Manish Pandey	4 th Aug 2017	Udita Singh	4 th Aug 2017	Udita Singh	4 th Aug 2017
1.11	Added Trouble shoot	Manish Pandey	17-Aug-17	Udita Singh	17-Aug-17	Udita Singh	17-Aug-17
1.12	<ol style="list-style-type: none"> 1. Proxy Configuration has been added. 2. Device Serial Number field has been added in Device Info Response. 3. Qscore field has been added in Capture Response. 4. Functionalities have been added includes – Otp+Bio, Otp+Demo, Otp(Proto and XML). 5. More logs have been added in Rd Service. 6. nmpoints field has been added for the possible combination of Biometric capture. 	Manish Pandey	23 rd Aug 2017	Udita Singh	23 rd Aug 2017	Udita Singh	23 rd Aug 2017
1.13	1. Fixed Issues for IE for Http Communication between Test Page and RD Service.	Manish Pandey	5 th Sep 2017	Udita Singh	5 th Sep 2017	Udita Singh	5 th Sep 2017
1.14	<ol style="list-style-type: none"> 1. Correction in the RD code includes storing of “ms” field(part of pfa) passing in XML as an input in the right structure. 2. “Validate Keystore function” is being called inside the thread. 3. Delay has been added during receiving/sending data to/from RD Service. 	Manish Pandey	18 th Sep 2017	Udita Singh	8 th Sep 2017	Udita Singh	8 th Sep 2017
1.15	<ol style="list-style-type: none"> 1. Morpho RD service has been listed in windows startup 2. Minor change in Main Service Code includes more logs have been added, API calling. 3. Added logic to capture fingerprint from Secured Morpho Devices. 4. Added logic to handle Device Details of Repaired Morpho Devices. 5. Changed the low level API of SDK and removed the static memory allocation in threads to avoid excess Memory consumption. 6. Changed in handling of pre-flight request. 	Manish Pandey	28 th Dec 2017	Udita Singh	28 th Dec 2017	Udita Singh	28 th Dec 2017
1.16	<ol style="list-style-type: none"> 1. Message boxes have been removed in case user intervention is not required and Instead Logs have been created at “C:\MORPHORDLOG\yearfolder\month_log.txt”. 2. New field “SrNoTag:1” has been added in 	Manish Pandey	11 th Jan 2018	Udita Singh	11 th Jan 2018	Udita Singh	11 th Jan 2018

RD Client Integration Document

	<p>ConfigSettings.ini only for NIC.</p> <p>3. In case of any issue during service upgrade while going for First time registration, it will bypass the issue and user will be able to register the device. This change has been made on a temporarily basis only for NIC. Once the testing is done successfully by NIC, the bypass action will be removed in case of first time registration.</p> <p>4. Dpsid, rdsid, rdsver, MI values are set as same in RD service across all the environment (Staging, Preprod, Prod).</p>						
1.17	<p>1. Rd service start issue has been fixed while system shutdown by changing the install – uninstall script in RD drivers to fix the Rd service start issue while system Shut down.</p> <p>2. Rd service Crash Issue has been fixed while calling capture.</p>	Manish Pandey	25 th Jan 2018	Udita Singh	25 th Jan 2018	Udita Singh	25 th Jan 2018
1.18	<p>1. Calculate checksum of UIDAI cert received from MS and store the same in file securely.</p> <p>2. Validate checksum stored in file with the checksum of pinned cert before every capture, in case of success allow capture otherwise throw certificate error to the calling app.</p> <p>3. Replaced old uidai cert pinned with latest one in Production</p>	Manish Pandey	19 th Dec 2019	Udita Singh	19 th Dec 2019	Udita Singh	19 th Dec 2019
1.19	Changed Low Level library	Manish Pandey	5th Jan 2020	Udita Singh	5th Jan 2020	Udita Singh	5th Jan 2020
1.20	Restrict Infinite Request sending attempt to 3 only to the sever in case of failure during fresh device registration and status update	Manish Pandey	10th Jan 2020	Udita Singh	10th Jan 2020	Udita Singh	10th Jan 2020

RD Client Integration Document

Table of Contents

1. Introduction.....	5
2. Scope	5
3. Installation Steps of RD service	5
4. Un-Installation Steps	5
5. Registration steps of Biometric Device	5
6. Pre-Requisites for Https Communication from Web Browser after Installation	6
7. Configuration settings according to environment	6
7.1 Staging :	6
7.2 Preproduction :	7
7.3 Production :	7
8. RD Services API Calling	8
8.1 RDSERVICE.....	8
8.2 DEVICEINFO	9
8.3 CAPTURE	10
9. Error Codes from RD service	13
10. Troubleshooting	14
11. S/W and H/W Requirements for new release	17

1. Introduction

Purpose of this document is to help the developers to integrate the Windows RD service in their application.

2. Scope

Scope of this document is limited to the Windows RD, its installation, integration with the Client application.

3. Installation Steps of RD service

1. Run 'MorphoRdServiceLOSoftSetup.exe' as administrator to install the RD Service
2. Follow the instruction in setup wizard to complete the installation.

4. Un-Installation Steps

1. Run 'C:\MorphoRdServiceLOSoftSetup\unins000.exe' to uninstall the RD Service OR Uninstall "Morpho RD Service Version Driver" from Control Panel.
2. Follow the instruction in setup wizard to complete the installation.

5. Registration steps of Biometric Device

1. Plug-in the Morpho Biometric Device.
2. If the Morpho Biometric Device is white listed in Management Server, it will be registered without giving Activation Code manually, but if it is not then user will be prompted an Activation Code Window to enter Activation Code for registration.

[Activation Code will be shared by Smart Chip Pvt Ltd]

RD Client Integration Document

6. Pre-Requisites for Https Communication from Web Browser after Installation

- 1) Set value 'CommunicationMode:0' in 'C:\MorphoRdServiceLOSoft\ConfigSettings.ini' file to enable Https communication in Morpho RD Service.
- 2) Keep the Bank/Merchant's server certificate file at 'C:\MorphoRdServiceLOSoft\' named as '**server.crt**'. Certificate must be in pem format.
- 3) Keep the Bank/Merchant's server private key file at 'C:\MorphoRdServiceLOSoft\' named as '**server.key**'. Private key must be in pem format.
- 4) Rename '127.0.0.1' as Bank/Merchant's URL to which certificate is issued, in the host file of the windows present at 'C:\Windows\System32\drivers\etc'. Update this URL in the calling JavaScript functions given in the MorphoRdServiceTestPage.html.
- 5) Restart the Morpho RD Service.

Please note:

*The **server.crt** is a CA signed certificate, in case of integration it can be a self signed certificate. But to avoid any browser issue please use signed certificate. To generate a self signed certificate please follow the below link:*

https://www.ibm.com/support/knowledgecenter/en/SSWHYP_4.0.0/com.ibm.apimgmt.cmc.doc/task_apionprem_generate_self_signed_openssl.html

or

<https://jamielinux.com/docs/openssl-certificate-authority/>

7. Configuration settings according to environment

7.1 Staging :

1. Change the RDEnviroment variable in ConfigSetting.ini file (file PATH : C:\MorphoRdServiceLOSoft\).
RDEnviroment:0

RD Client Integration Document

2. Change the URLs:

Registration : <https://Stage-rdm.smartbioplus.com/rdm-device-app/registration>

Keyrotation : <https://Stage-rdm.smartbioplus.com/rdm-key-management-app/keyRotation>

Telemetry : <https://Stage-rdm.smartbioplus.com/rdm-telemetry-app/telemetry>

3. Change URL_IP : Stage-rdm.smartbioplus.com

4. Change URL_Port: 443

5. Save the changes in config file.

6. Restart the service to effect the changes.

7.2 Preproduction :

1. Change the RDEnviroment variable ConfigSetting.ini file (file PATH : C:\MorphoRdServiceLOSoft\).

RDEnviroment:1

2. Change the URLs:

Registration : <https://pre-rdm.smartbioplus.com/rdm-device-app/registration>

Keyrotation : <https://pre-rdm.smartbioplus.com/rdm-key-management-app/keyRotation>

Telemetry : <https://pre-rdm.smartbioplus.com/rdm-telemetry-app/telemetry>

3. Change URL_IP : pre-rdm.smartbioplus.com

4. Change URL_Port: 443

5. Save the changes in config file.

6. Restart the service to effect the changes.

7.3 Production :

1. Change the RDEnviroment variable ConfigSetting.ini file (file PATH : C:\MorphoRdServiceLOSoft\).

RDEnviroment:2

2. Change the URLs:

Registration : <https://rdm.smartbioplus.com/rdm-device-app/registration>

RD Client Integration Document

Keyrotation : <https://rdm.smartbioplus.com/rdm-key-management-app/keyRotation>

Telemetry : <https://rdm.smartbioplus.com/rdm-telemetry-app/telemetry>

3. Change URL_IP : rdm.smartbioplus.com

4. Change URL_Port: 443

5. Save the changes in config file.

6. Restart the service to effect the changes.

8. RD Services API Calling

There are 3 actions in MorphoRDServiceTestPage.html, these actions are listed below:

- a. RDSERVICE
- b. DeviceInfo
- c. Capture

The sequence of calling the API is as follows:

1. RDSERVICE->DeviceInfo
2. RDSERVICE->Capture

8.1 RDSERVICE

To discover RD Service, please use the below Javascript code snippet:

Function

RDService()

```
{
var url = "http://127.0.0.1:11100";

var xhr;
var ua = window.navigator.userAgent;
var msie = ua.indexOf("MSIE ");

if (msie > 0 || !navigator.userAgent.match(/Trident.*rv\:11\./)) // If
Internet Explorer, return version number
{
//IE browser
xhr = new ActiveXObject("Microsoft.XMLHTTP");
} else {
```

RD Client Integration Document

```
//other browser
xhr = new XMLHttpRequest();
}
xhr.open('RDSERVICE', url, true);
xhr.onreadystatechange = function () {

if (xhr.readyState == 4){
var status = xhr.status;
if (status == 200) {
alert(xhr.responseText);
console.log(xhr.response);
} else {
console.log(xhr.response);
}
}
};
/*setTimeout(function(){
xhr.send();},1000);*/
xhr.send();
}
```

8.2 DEVICEINFO

To get device info, please use the below Javascript code snippet:

```
function
DeviceInfo()

{
var url = "http://127.0.0.1:11100/getDeviceInfo";

var xhr;
var ua = window.navigator.userAgent;
var msie = ua.indexOf("MSIE ");

if (msie > 0 || !!navigator.userAgent.match(/Trident.*rv\:11\./)) // If
Internet Explorer, return version number
{
//IE browser
```

RD Client Integration Document

```
xhr = new XMLHttpRequest("Microsoft.XMLHTTP");
} else {
//other browser
xhr = new XMLHttpRequest();
}

xhr.open('DEVICEINFO', url, true);
xhr.onreadystatechange = function () {

if (xhr.readyState == 4){
var status = xhr.status;
if (status == 200) {
alert(xhr.responseText);
console.log(xhr.response);
} else {
console.log(xhr.response);
}
}
};

xhr.send();
}
```

Response data:

```
<DeviceInfo dpld="" rdsId="" rdsVer="" dc="" mi="" mc="" >
<additional_info><Param name="serial_number" value=""/></additional_info></DeviceInfo>
```

- (mandatory) Unique code assigned to registered device provider. **dpId**
 - (mandatory) Unique ID of the certified registered device service. **rdsId**
 - rdsVer** - (mandatory) Registered devices service version.
 - (mandatory) Unique Registered device code. **dc**
 - (mandatory) Registered device model ID. **mi**
 - mc** - (mandatory) This attribute holds registered device public key certificate. This is signed with device provider key.
- In additional info tag, value field is the device serial number connected.

8.3 CAPTURE

To use CAPTURE command, please use the below Javascript code snippet:

RD Client Integration Document

```
function
Capture() {
    var url = "http://127.0.0.1:11100/capture";

    var PIDOPTS='<PidOptions ver=\"1.0\">'+<Opts fCount=\"1\" fType=\"0\"
iCount=\"\" iType=\"\" pCount=\"\" pType=\"\" format=\"0\" pidVer=\"2.0\"
timeout=\"10000\" otp=\"\" wadh=\"\" posh=\"\"/>'+</PidOptions>';

    /*
    format=\"0\" --> XML
    format=\"1\" --> Protobuf
    */
    var xhr;
    var ua = window.navigator.userAgent;
    var msie = ua.indexOf("MSIE ");

    if (msie > 0 || !navigator.userAgent.match(/Trident.*rv\:11\./)) // If
Internet Explorer, return version number
    {
        //IE browser
        xhr = new ActiveXObject("Microsoft.XMLHTTP");
    } else {
        //other browser
        xhr = new XMLHttpRequest();
    }

    xhr.open('CAPTURE', url, true);
    xhr.setRequestHeader("Content-Type","text/xml");
    xhr.setRequestHeader("Accept","text/xml");

    xhr.onreadystatechange = function () {

        if (xhr.readyState == 4){
            var status = xhr.status;
            if (status == 200) {
                alert(xhr.responseText);

            } else {
                console.log(xhr.response);
            }
        }
    }
}
```

RD Client Integration Document

```

    }
  };
  xhr.send(PIDOPTS);
}

```

Input data detail need to send in the above request:

```

<PidOptions ver="">
<Opts fCount="" fType="" iCount="" iType="" pCount="" pType="" format="" pidVer="" timeout="" otp=""
wadh="" posh="" />
<Demo></Demo>
<CustOpts>
<!-- no application should hard code these and should be configured on app or AUA servers. These
parameters can be used for any custom application authentication or for other configuration
parameters. Device providers can differentiate their service in the market by enabling advanced
algorithms that applications can take advantage of. -->
<Param name="" value="" />
</CustOpts>
</PidOptions>

```

It should send this input data in this key “PID_OPTIONS” using intent to RD Service
 Where:

PidOptions:

ver: Version of the PidOptions spec. Currently it is “1.0”. This is necessary to allow applications to gracefully upgrade even when RD service may be been upgraded. **RD Service must support current version and one previous version** to allow apps to upgrade at different points in time.

Opts:

Int **fCount** (optional) number of finger records to be captured (0 to 10)

Int **fType** (optional) ISO format (0 for FMR or 1 for FIR), 0 (FMR) is default

Int **iCount** (optional) number of iris records to be captured (0 to 2)

Int **iType** (optional) ISO format (0 for IIR), 0 (IIR) is default

Int **pCount** (optional) number of face photo records to be captured (0 to 1).

Currently face matching is not supported.

Int **pType** (optional) face format. Currently face matching is not supported.

Int **format** (mandatory) 0 for XML, 1 for Protobuf

String **pidVer** (mandatory) PID version

Int **timeout** capture timeout in milliseconds

String **otp** (optional) OTP value captured from user in case of 2-factor auth

String **wadh** (optional) If passed, RD Service should use this within PID block root element “as-is”. String

posh (optional) if specific positions need to be captured, applications can pass a comma delimited position attributes. See “posh” attribute definition in Authentication Specification for valid values. RD Service (if showing preview) can indicate the finger using this. If passed, this should be passed back within PID block. Default is “UNKNOWN”, meaning “any” finger/iris can be captured.

Demo:

Element allows demographic data to be passed to form PID block as per authentication specification

Response Data Format:

When it request to capture finger data using RD Service, It returns some xml data as output that would be further used to Authentication as well as eKYC.

```
<PidData>
<Resp errCode="" errInfo="" fCount="" fType="" iCount="" iType="" pCount="" pType="" nmPoints=""
qScore=""/>
<DeviceInfo />
<Skey ci="">encrypted and encoded session key</Skey> <Hmac>SHA-256
Hash of Pid block, encrypted and then encoded</Hmac> <Data
type="X|P"> base-64 encoded encrypted pid block </pid> </PidData>
```

Where:

Resp:

Int **errCode** (mandatory) 0 if no error, else standard error codes

String **errInfo** (optional) additional info message in case of error/warning

Int **fCount** (mandatory for FP) number of finger records actually captured

Int **fType** (mandatory for FP) actual format type – 0 (FMR) or 1 (FIR) Int

iCount (mandatory for Iris) number of iris records actually captured Int


iType (mandatory for Iris) actual Iris format (0 for IIR)

Int **pCount** (mandatory for Photo) number of face photo records actually captured. Currently face matching is not supported.

Int **pType** (mandatory for Photo) face format. Currently face matching is not supported.

Int **nmPoints** (mandatory for FMR capture) Number of minutiae points when FMR is captured.

Applications may use this for accepting or retrying the capture. If multiple fingers are captured, send comma delimited numbers.

Int **qScore** (optional) If quality check is done, send a normalized score that is between 0 and 100. Device providers may allow configuration within RD service to use specific quality check algorithms to be enabled. Either it can be configured within RD service or applications can pass those under PidOptions CustOpts Param. 

Skey:

String **skey** (mandatory) encrypted session key as per auth spec

String **ci** (mandatory) UIDAI public key identifier as per auth spec

Hmac:

String **hmac** (mandatory) hmac value as per auth spec.

9. Error Codes from RD service

100 "Invalid PidOptions input. XML should strictly adhere to spec."

110 "Invalid value for fType"

120 "Invalid value for fCount"

130 "Invalid value for iType"

140 "Invalid value for iCount"

RD Client Integration Document

150 "Invalid value for pidVer"
 160 "Invalid value for timeout"
 170 "Invalid value for posh"
 180 "Face matching is not supported"
 190 "Invalid value for format"
 200 "Invalid Demo structure"
 210 "Protobuf format not supported"
 700 "Capture timed out"
 710 "Being used by another application"
 720 "Device not ready"
 730 "Capture Failed"
 740 "Device needs to be re-initialized"
 750 "RD Service does not support fingerprints"
 760 "RD Service does not support Iris"
 770 "Invalid URL"
 999 "Internal error"
 240 "UIDAI Certificate from management server is invalid"

10. Troubleshooting

S.No.	Error Code	Error Info	Occurrence	Solution
1.	100	Invalid PidOptions input. XML should strictly adhere to spec.	When RD Service calling application sends corrupt pidoption xml or may be incomplete pid option xml.	Before calling capture function check pidoption xml format properly.
2.	110	Invalid value for fType	When RD Service calling application sends wrong value for finger type according to UIDAI registered device document.	Before calling capture function check fType attribute value properly. It should be according to UIDAI registered device document.
3.	120	Invalid value for fCount	When RD Service calling application sends wrong value for finger count according to UIDAI registered device document.	Before calling capture function check fCount attribute value properly. It should be according to UIDAI registered device document.
4.	130	Invalid value for iType	When RD Service calling application sends wrong value for iris type according to UIDAI registered device document.	Before calling capture function check iType attribute value properly. It should be according to UIDAI registered device document.
5.	140	Invalid value for iCount	When RD Service calling application sends wrong	Before calling capture function check iCount

RD Client Integration Document

			value for iris count according to UIDAI registered device document.	attribute value properly. It should be according to UIDAI registered device document.
6.	150	Invalid value for pidVer	When RD Service calling application sends wrong value for pidblock version according to UIDAI registered device document.	Before calling capture function check pidVer attribute value properly. It should be according to UIDAI registered device document.
7.	160	Invalid value for timeout	When RD Service calling application sends wrong value for timeout according to UIDAI registered device document.	Before calling capture function check timeout attribute value properly. It should be according to UIDAI registered device document.
8.	170	Invalid value for posh	When RD Service calling application sends wrong value for posh according to UIDAI registered device document.	Before calling capture function check posh attribute value properly. It should be according to UIDAI registered device document.
9.	180	Face matching is not supported	When RD Service calling application sends value for pCount and pType.	Morpho RD Service not supported face matching. So ignore/remove pCount and pType attributes.
10.	190	Invalid value for format	When RD Service calling application sends wrong value for format according to UIDAI registered device document.	Before calling capture function check format attribute value properly. It should be according to UIDAI registered device document.
11.	200	Invalid Demo structure	When RD Service calling application sends wrong Demo xml format according to UIDAI Aadhaar authentication document.	Before calling capture function check Demo xml format attribute value properly. It should be according to UIDAI Aadhaar authentication document.
11.	700	Capture timed out.	If Customer not putting finger on sensor within giving timeout.	Make sure customer put their finger on sensor within giving timeout.
12.	710	Being used by another application.	If Fingerprint sensor busy by another application	Make sure fingerprint sensor should be in ready state. So call device info and check rd service status before calling

RD Client Integration Document

				capture. If RD Service status is ready than capture should be perform.
13.	710	Being used by another application.	If Fingerprint sensor busy by another application	Make sure fingerprint sensor should be in ready state. So call device info and check rd service status before calling capture. If RD Service status is ready than capture should be perform.
14.	720	Device not ready.	If Fingerprint device haven't permission. During capture usb connection loose. Backward compatible issue	Make sure fingerprint sensor has permission and USB cable connection should be perfect.
15.	730	Capture Failed	Some unknown issue	Retry process
16.	740	Device needs to be re-initialized	When RD Service environment changed	Do Registration
17.	760	RD Service does not support Iris	When RD Service calling application sends value for iCount and iType.	Morpho RD Service not supported eye matching. So ignore/remove iCount and iType attributes.
18.	999	Internal error	<ul style="list-style-type: none"> • Problem Occur during PID generation • During Finger Capture • RD Service in different Environment • Device date time is not set to automatic. • Internal error 	<ul style="list-style-type: none"> • Retry Capture • Retry Capture • Launch RD Service and click refresh button at right top corner. • Please ensure that value of env attribute in PID Option xml is correct(according to RD Service environment). • Host Machine date & time should be auto sync. • Please ensure that value of env attribute in PID Option xml is

				correct(according to RD Service environment).
--	--	--	--	---

11. S/W and H/W Requirements for new release

- **Prerequisites for S/W**

OS - Windows 7

Web Browser - Chrome version 60 and IE 11

- **Prerequisites for H/W**

Morpho MSO1300 E, MSO1300 E2, MSO1300 E3 Biometric Sensor

- **Any other tool**

NA