TIQR API WSDL

The specifications and information in this document are subject to change without notice. Companies, names, and data used in examples herein are fictitious unless otherwise noted. This document may not be copied or distributed by any means, in whole or in part, for any reason, without the express written permission of RCDevs.

Copyright (c) 2010-2017 RCDevs SA. All rights reserved.
http://www.rcdevs.com

WebADM and OpenOTP are trademarks of RCDevs. All further trademarks are the property of their respective owners.

Limited Warranty

No guarantee is given for the correctness of the information contained in this document. Please send any comments or corrections to info@rcdevs.com.

TiQR API WSDL

TiQR

TiQR API Description

TiQR provides 10 methods for authentication and RSA cryptography:

1. tiqrStart

Initializes a TiQR authentication or cryptographic session and returns a QRCode and a session ID.

The request contains the following attributes:

- client: Client identifier (NAS) to be used in service logs (defaults to the client IP address).
- source: IP address of the end user system (optional).
- settings (optional): List of TiQR settings which will override the user/group/application server-side settings (ex. *QRSize=5,SessionTimeout=200*).
- options: Reserved for restricted usage by some plugins.

The following attributes are for RSA cryptographic transactions (data signature and decryption).

- operation: The cryptographic operation to be applied to the inputData on TiQR:
  - auth or authentication: No RSA operation is expected. This is the default not set.
  - sign or signature: TiQR will sign the inputData with its RSA private key.
  - pgpsign or pgpsignature: TiQR will PGP sign the inputData with its RSA private key.
  - decr or decrypt: TiQR will decrypt the inputData with its RSA private key.
- inputText: A text or HTML description of the transaction to be displayed on TiQR display.
- inputData: An array of binary data to be signed or decrypted with RSA on TiQR.
  - With RSA signature, the data must be a hash (SHA1/SHA2) of the document to be signed.
  - With RSA decryption, the data should have been encrypted using the tiqrEncrypt API below.

The response contains the following attributes:

- code:
  - 1 means session initialization success.
  - 0 means session initialization failure.
- error: The error ID if code 0 was returned. The ID corresponds to the error message template names in tiqr.xml (ex. AuthFailed).
- message: The server reply message (success message or error description).

- session: This is the session ID to be passed in the other TiQR methods.
- QR: This attribute contains the QRCode GIF image in binary format.
- URI: This attribute contains the URI string corresponding to the QRCode. The client application can generate itself a QRCode with the URI instead of using the pre-generated GIF image.
- timeout: This is the expiration timeout for the opened session.

2. tiqrCheck

Queries the status of an opened session intitialized with tiqrStart.

The request contains the following attribute:

- session (mandatory): The session ID of the opened session.
- ldapPassword (optional): The LDAP password (mandatory when TiQR LoginMode is set to LDAPTIQR). RSA cryptographic operation may require LDAP password too if CryptoMode is set to LDAPTIQR.

The response contains the following attributes:

- code:
  - 0 means operation failed.
  - 1 means operation succeeded.
  - 2 means operation is still pending (no communication occurred yet with the mobile phone).
  - 3 means operation is not complete (the TiQR transaction succeeded but the user LDAP password is expected).
- error: The error ID if code 0 was returned.
- message: The server reply message (success message or error description or auth pending message).
- username: If the code is 1, this attribute returns the authenticated user login name.
- domain: If the code is 1, this attribute returns the authenticated user domain name.
- timeout: The remaining time before the session expires. Note: After a success or failure, the session is not dropped. You can optionally destroy it with the tiqrCancel method.
- data: If the code is 1, this attribute contains the ReplyData set in the LDAP user settings.

The following attributes are for RSA cryptographic transactions (data signature and decryption).

- outputData: An array of RSA signatures, PGP signed documents or decrypted data.
- publickey: The public key or certificate corresponding to the TiQR user identity.
- format: RSA (RSA public key), PGP (OpenPGP public key) or CRT (X.509 certificate).

3. tiqrOfflineCheck

Performs the user authentication from the API and not through a direct communication between the mobile phone and the TiQR server's mobile endpoint. This method should be used if the TiQR application is offline because the user mobile phone is not connected to the Internet.

The request contains the following attributes:

- username (mandatory): This is the user login name.
- domain (optional): This is the user login domain.
- session (mandatory): The session ID of the authentication session.
- tiqrPassword (mandatory): This is the TiQR authentication response (displayed by the TiQR).
- ldapPassword (optional): The LDAP password (mandatory when TiQR LoginMode is set to LDAPTIQR).

The response contains the following attributes:

- code:
  - 0 means authentication failed.
  - 1 means authentication succeeded.
- error: The error ID if code 0 was returned.
- message: The server reply message (success message or error description).
- data: This attribute contains the ReplyData set in the LDAP user settings.

4. tiqrAssign

Assigns an opened session to a specific user and optionally sends a mobile Push notification to the TiQR of this user (no need to scan the QRCode). Once assigned, a login or crypto transaction can only be performed by the provided user.

The request contains the following attribute:

- username (mandatory): This is the user login name.
- domain (optional): This is the user login domain.
- session (mandatory): The session ID of the opened session.
- push (optional): If true then a mobile push notification is sent to TiQR.

The response contains the following attributes:

- code:
  - 0 means push notification failed.
  - 1 means push notification succeeded.
- error: The error ID if code 0 was returned.

- message: The server reply message (success message or error description).

5. tiqrCancel

Destroys an initialized session before it has expired.

The request contains the following attribute:

- session (mandatory): The session ID of the opened session.

The response contains the following attributes:

- code:
  - 0 means session cancel failed.
  - 1 means session cancel succeeded.
- error: The error ID if code 0 was returned.
- message: The server reply message (success message or error description).

6. tiqrSessionQR

Retrieves the QRCode of an opened session initialized with tiqrStart.

The request contains the following attribute:

- session (mandatory): The session ID of the opened session.

The response contains the following attributes:

- code:
  - 0 means QR retrieval failed.
  - 1 means QR retrieval succeeded.
- error: The error ID if code 0 was returned.
  - message: The server reply message.
  - QR: This attribute contains the QRCode GIF image in binary format.
  - URI: This attribute contains the URI string corresponding to the QRCode.
  - timeout: This is the expiration timeout for the opened session.

7. tiqrStatus

Used to query a server status.

The request does not contain any attribute.

The response contains the following attributes:

- status:
  - 1 if the server is willing to accept requests.
  - 0 if the server cannot accept new requests.
- message: The server status details.

8. tiqrVerify

Validates an RSA signature done with TiQR.

The request contains the following attributes:

- username (mandatory): This is the user login name.
- domain (optional): This is the user login domain.
- client: Client identifier (NAS) to be used in service logs (defaults to the client IP address).
- source: IP address of the end user system (optional).
- inputData (mandatory): The binary data to be signed.
- outputData (mandatory): The signature value to be checked.

The response contains the following attributes:

- code:
  - 0 means data signature verification success.
  - 1 means data signature verification failed.
- error: The error ID if code 0 was returned.
- message: The server reply message (success message or error description).

9. tiqrEncrypt

Encrypts a binary data with the user RSA public key.

The request contains the following attributes:

- username (mandatory): This is the user login name.
- domain (optional): This is the user login domain.
- client: Client identifier (NAS) to be used in service logs (defaults to the client IP address).
- source: IP address of the end user system (optional).

- inputData (mandatory): The binary data to be encrypted.

The response contains the following attributes:

- code:
  - 0 means data encryption success.
  - 1 means data encryption failed.
- error: The error ID if code 0 was returned.
- message: The server reply message (success message or error description).

10. tiqrPubkey

Retrieves a user public key.

The request contains the following attributes:

- username (mandatory): This is the user login name.
- domain (optional): This is the user login domain.
- client: Client identifier (NAS) to be used in service logs (defaults to the client IP address).
- source: IP address of the end user system (optional).
- format: The output format (DER or PEM). Binary output (DER) is the default.

The response contains the following attributes:

- code:
  - 0 means key retrieval success.
  - 1 means key retrieval failed.
- error: The error ID if code 0 was returned.
- message: The server reply message (success message or error description).
- publicKey: The user RSA public key in DER format.

This manual was prepared with great care. However, RCDevs S.A. and the author cannot assume any legal or other liability for possible errors and their consequences. No responsibility is taken for the details contained in this manual. Subject to alternation without notice. RCDevs S.A. does not enter into any responsibility in this respect. The hardware and software described in this manual is provided on the basis of a license agreement. This manual is protected by copyright law. RCDevs S.A. reserves all rights, especially for translation into foreign languages. No part of this manual may be reproduced in any way (photocopies, microfilm or other methods) or transformed into machine-readable language without the prior written permission of RCDevs S.A. The latter especially applies for data processing systems. RCDevs S.A. also reserves all communication rights (lectures, radio and television). The hardware and software names mentioned in this manual are most often the registered trademarks of the respective manufacturers and as such are subject to the statutory regulations. Product and brand names are the property of RCDevs S.A. © 2018 RCDevs SA, All Rights Reserved