

qrLogin
Developer's Guide
Version 1.2

Table of contents

qrLogin. System description	2
How to Embed System on Your Web Source	2
Main Functions	2
Strategy to Embed qrLogin System on Web Source	2
Mode of Operations	2
Authentication on the Web Source	3
New User Registration on Web Source or in the Mobile Application	4
Authentication Methods	4
Data Transmission Methods	4
Levels of Credentials Transmission Security	5
New User with Static Password	5
New User with OTP Password.	6
New User with Static Password Generated in Mobile Application	6
New User with OTP Password and Key Generated in Mobile Application	7
How to link the scheme of qrlogin:// to authentication QR-code image.	7
Annexes	9
Types and Methods of Authentication Used in qrLogin Application	9
OTP Password Parameters when Registering New Account	10
QR Code Examples	10
Recommended Error Codes	11

qrLogin. System description

QrLogin is an authentication system based on the reading of qr code by mobile phone and transfer of authentication data via the http / https protocol to application or to web resource.

The mobile application qrLogin by reading of a specially generated qr-code allows:

- To authenticate on a web resource or in an application;
- To subtract and save account data;
- To subtract the credentials of the new account, generate a password or key and send these data to the server to complete the registration of this account.

How to Embed System on Your Web Source

Main Functions

For the correct work of the authentication system qrLogin the following functions must be supported:

1. Authentication function that includes:
 - a. Generation of the authentication qr code on the login page of the web resource;
 - b. Receiving and verifying an authentication request from a mobile device that will arrive at the URL specified in the authentication qr code;
 - c. Checking of data received and logging in auto mode onto a web resource with user credentials in case if this data is correct.
2. Registration function for a new user with qrLogin support that includes:
 - a. Generation of qr-code with credentials of user account on the registration page and / or on the credential settings page. This code is required to update the mobile application account database.

Strategy to Embed qrLogin System on Web Source

Before embedding the qrLogin system in a web resource, you need to answer 3 questions that will help you uniquely choose which authentication and registration methods do you need to support.

1. What type of http requests is more convenient to handle, POST with parameters or POST with JSON?
2. What type of passwords should the system support, static password or the OTP one?
3. What level of security during the transferring of passwords from a web resource to a mobile application is required? Is it needed to generate passwords or keys in the mobile application, or will this data be transmitted in the qr-registration code.

Once these questions are answered it becomes clear what type of qr-code from those that are described in [Format of QR Codes while New User Registering](#) will need to be supported in the web resource, what data should be prepared to generate this code and what requests from the mobile application need to be processed.

Mode of Operations

The general approach in executing of qrLogin system consists in:

- Mobile application reads the qr-code of authentication of the web resource and finds the corresponding record in the local database of accounts;
- Generates authentication data that includes the user name, password or one-time password and the session ID obtained from the qr authorization code;
- This data is transmitted to the web resource. Methods and format of data transfer is described in the section [Data transmission methods](#);
- This data is transmitted to the web resource. More information about methods and format of data transferring can be found at [Data Transmission Methods](#);
- The web resource checks the relevance of the session identifier, as well as the validity of the credentials, and in the case of a positive check, automatically logs onto the web resource from the authentication page of the corresponding transmitted session identifier;
- In response to a request from the mobile device in case of positive authentication, the web resource should return HTTP status “OK-200”;
- If the request was processed with an error, the web resource should return the corresponding error status. The recommended error codes are listed in the table [Recommended Error Codes](#);
- QrLogin is recommended to be used as an additional authentication system. If there is no Internet connection on the mobile device or in case of inability to use it (roaming mode), the user can enter both passwords - the static password and the OTP one - manually through the web source authentication form.

Authentication on the Web Source

For authentication on the web resource please place on the login form a QR-code that will contain the following text information:

Line	Description	Value
1	qrLogin identifier. This is the constant value for all QR codes	QRLOGIN
2	Function identifier. This is the constant value for this function type.	L:V1
3	Source URL	https://www.resources.com/sites/yousite
4	Unique session identifier	b7e3d7a5707806c17a380c31a56c

The first and the second lines are constant values. The first line - “QRLOGIN” is a system identifier, that is represented in all qr-codes at the same place (first line). The second line is an identifier which informs that this code is intended for authentication on the source.

IMPORTANT: All constant values should be entered without any changes and with capital letters.

The third line contains determination of source URL. URL is an identifier of your web source in the database of mobile application. URL should not contain any link on the pages of your source. It can be just web domain name (for example: www.yourname.com) or the name of domain with path name to your application (for example: www.resources.com/sites/yousite). URL of source can include also a protocol identifier (for example:

<https://www.resources.com/sites/yoursite>). If the URL of source does not contain information about protocol the mobile application automatically will add “http://”

The fourth line is the unique identifier of user session on the web resource. For this parameter, you can use both the http-session identifier or any other identifier generated on the web resource. The main restriction is that this identifier should be the unique one and should uniquely identify the session of user on the web source.

The lines are separated by a symbol “line feed” (\n) - code 0x0A

The examples of qr-codes can be found in [QR Code Samples](#)

New User Registration on Web Source or in the Mobile Application

For new user registration the web source should place a specially generated qr-code on the page of registered user properties. By reading of this qr-code with the scanner of mobile application the user will add or update the account in the database of mobile application.

The QR-code of new user registration has a number of settings that depend on the authentication method, the method of transferring data to the server and the security level of transferring credentials to the mobile application.

Authentication Methods

Nowadays qrLogin system supports 2 authentication methods:

- with static password;
- with OTP password. System supports the newest algorithm of one-time passwords generating - TOTP (Time-based One Time Password Algorithm, RFC 6238). In realization of this algorithm the SHA1, SHA256, SHA512 algorithms and the possibility to generate the password with up to 8 symbols are supported.

Data Transmission Methods

Two methods of transferring data to the authentication page of web source can be used:

- HTTP POST request where the authentication data is transferred as the parameters of request. In this case the structure of request is as follows: URL: <https://yoursite.com/yourApp/loginpage.php>
PARAMETERS:

objectName=qrLogin&login=*username*&sessionId=*xxxxSESSIONIDxxxx*&password=*userpassword*

- HTTP POST with JSON. In this case the authentication parameters are transferred as JSON object in POST request body. In this case the structure of request is as follows:

```
{
  "objectName": "qrLogin",
  "sessionId": "xxxxSESSIONIDxxxx",
  "password": "userpassword",
  "login": "username"
}
```

In the same way a mobile device transmits both the authentication data and the data for user registering on the web resource when the password generation method is performed on the mobile application side.

Levels of Credentials Transmission Security

When transferring credentials from a web resource to a mobile application to realize a new account registration there are two levels of security:

- standard security level - in this case all account data among them the password and the key to generate the OTP password are included into registration QR-code.
- upgraded security level - in this case QR code contains all data except the password or OTP password key. When this occurs the qr-code must also contain the identifier of the current user session. The mobile application by scanning this code will propose to user to generate a password or generate a key for the OTP password and to send this data to a web resource. Please see the details at [Data Transmission Methods](#).

IMPORTANT: To ensure an acceptable level of security for credentials we recommend you to use only the HTTPS protocol while transferring data between the web resource and the mobile application. The mobile application has as an option of self-signed SSL certificate. Also the mobile application can realize correctly the redirection of HTTP requests to HTTPS if such option is configured on your web server.

Format of QR Codes while New User Registering

Each qr code in its first line contains the identifier “QRLOGIN”.

In the second line it is determined the identifier of new user adding operation “NU:V1” or “NU:V2”.

The third line contains the information about URL web page that is the identifier of your source. URL can include also protocol identifier but it should not contain any link on the pages of your source.

For example: <https://youdomain.com>, <https://hostersite.com/youapplication>, <yoursite.com/application1>

The fourth line contains information about the path name and the name of page or part of URL that will receive requests from the mobile application. In the mobile application the complete URL that will be used to send the request will be formed as the third line of qr code + the fourth line. In case if the URL of source does not contain information about protocol the mobile application automatically will add “HTTP://”

The lines are separated by a symbol “line feed” (n) - code 0x0A

The examples of qr-codes can be found in [QR Code Samples](#)

New User with Static Password

Line	Description	Value
1	qrLogin identifier. This is the constant value for all QR codes	QRLOGIN
2	Function identifier. This is the constant value for this function type.	NU:V1
3	Source URL	https://www.resources.com/sites/yosite
4	Page of user authorization from the mobile application	/qrlogin.xhtml
5	User name	admin
6	Password	admin

7	Type of request to the server. To see the types and the reference codes please see the table below and visit the link Types and Methods of Authorization used in qrLogin Application	1
---	--	---

New User with OTP Password.

Line	Description	Value
1	qrLogin identifier. This is the constant value for all QR codes	QRLOGIN
2	Function identifier. This is the constant value for this function type.	NU:V1
3	Source URL	https://www.resources.com/sites/yousite
4	Page of user authorization in the mobile application	/loginsecret.shtml
5	User name	<i>admin</i>
6	OTP password parameters	Please see the link OTP Password Parameters when Registering New Account
7	Type of request to the server. Please see the link Types and Methods of Authorization Used in qrLogin Application	4

New User with Static Password Generated in Mobile Application

Line	Description	Value
1	qrLogin identifier. This is the constant value for all QR codes	QRLOGIN
2	Function identifier. This is the constant value for this function type.	NU:V2
3	Source URL	https://www.yousite.com
4	Page for user authorization and account credentials transfer when registration is done in mobile application	/qrlogin.php
5	User name	<i>admin</i>
6	Parameters of password generation separated by the symbol “;”: 1 - minimum password length 2 - if 1 - password should contain small and uppercase letters 3 - if 1 - password should contain numbers	<i>8;1;1;0</i>

	4 - if 1 - password should contain special characters	
7	Type of request to the server. Please see the link Types and Methods of Authorization Used in qrLogin Application	2
8	The unique session identifier. It is needed to identify the current user while receiving the password data from the mobile application.	c08596ad07f529e123e0f8e3223b

New User with OTP Password and Key Generated in Mobile Application

Line	Description	Value
1	qrLogin identifier. This is the constant value for all QR codes	QRLOGIN
2	Function identifier. This is the constant value for this function type.	NU:V2
3	Source URL	https://www.yousite.com
4	Page for user authorization and account credentials transfer when registration is done in mobile application	/qrlogin.php
5	User name	<i>admin</i>
6	OTP password key parameters. In this code the parameters should start with the symbol “;” “30” - time of password validity in seconds “SHA1” - key generation algorithm, the algorithms SHA1, SHA256, SHA512 are supported “6” - password length	<i>;30;SHA1;6</i>
7	Type of request to the server. Please see the link Types and Methods of Authorization Used in qrLogin Application	4
8	The unique session identifier. It is needed to identify the current user while receiving the password data from the mobile application.	c08596ad07f529e123e0f8e3223b

How to link the scheme of qrlogin:// to authentication QR-code image.

To support authentication with qrLogin application on the mobile device please put the picture of QR-code in HTML code of web source as <a> tag, href-parameter of which one is formatted as follows:

“qrlogin://”+”text of QR-code for authentication”+”qrlogin://”+”URL to return to web browser for iOS application”.

The text content of QR-code should be transferred to qrLogin-URL without any changes. For correct displaying of information in URL new line character “\n” should be corrected to “%0A”. No line advance action should be realized at the end of QR-code text content.

Example of URL:

*qrlogin://QRLOGIN%0AL:V1%0Aqrlogin.info/forum%0A2b6f237da373b639448a788ea18cf12qrlogin
://http%3A%2F%2Fqrlogin.info%2Fforum%2Findex.php*

Example of HTML code:

```
<a  
href="qrlogin://QRLOGIN%0AL:V1%0Aqrlogin.info/forum%0A28724a2b649681934471740b8c399ae  
8qrlogin://http%3A%2F%2Fqrlogin.info%2Fforum%2Findex.php" title="Scan in qrLogin"><?xml  
version="1.0" encoding="utf-8"?>  </a>
```

Once entered on web source by using the mobile device with preinstalled qrLogin application user should click on QR-code image. Upon doing that qrLogin application is launched. In this application user can pass authorization by parameters specified in URL on web source and after that qrLogin will return control back to browser.

The same procedure should be applied to obtain the image of QR-code for new user registration so that please use `<a>` tag href-parameter of which one is formatted as follows:

“qrlogin://”+” text of QR-code for new user registration”

There is no necessity in URL adding to return to web browser for iOS.

Example of URL:

qrlogin://QRLOGIN%0ANU:V1%0Aqrlogin.info/forum%0A/qrlogin%0Auser%0A%0A2

Example of HTML code:

```
<a href="qrlogin://QRLOGIN%0ANU:V1%0Aqrlogin.info/forum%0A/qrlogin%0Amorry%0A%0A2"  
title="Scan for save account qrLogin">  </a>
```

Annexes

Types and Methods of Authentication Used in qrLogin Application

Title	Description	Code
Static Password, HTTP(s) request POST with parameters	Mobile application performs POST request that contents authentication data as for example: objectName=qrLogin&login=admin&sessionId=bd54390368c4000b6dfae69bdbf&password=admin The order of fields in request parameters can be arbitrary. In case of successful authorization server returns the standard HTTP status "OK" - 200	1
Static Password, HTTP(s) request POST with JSON	Mobile application opens URL of web source with the authentication page (for example: https://qrlogin.tcon.com.ua/qrLogin/loginsecret.xhtml) and using POST method sends JSON object with the next content: {"objectName":"qrLogin","sessionId":"bdfa017358ee07403f77c564c2b0","password":"admin","login":"admin"} The order of fields in JSON object can be arbitrary, the format will match the example.	2
OTP Password, HTTP(s) request POST with parameters	Mobile application performs POST request that contents authentication data as for example: objectName=qrLogin&login=test&sessionId=be3ee6cf88573b767bccef0af0f4&password=737018 As the password, the OTP password calculated by the device according to the OTP password parameters is transmitted when registering. The order of fields in request parameters can be arbitrary. In case of successful authorization server returns the standard HTTP status "OK" - 200	3
OTP Password, HTTP(s) request POST with JSON	Mobile application opens URL of web source with the authentication page (for example: https://qrlogin.tcon.com.ua/qrLogin/loginsecret.xhtml) and using POST method sends JSON object with the next content: {"sessionId":"be711d16882a90be0a5cba6cc3eb","password":"495655","objectName":"qrLogin","login":"test"} The order of fields in JSON object can be arbitrary, the format will match the example. As the password, the OTP password calculated by the device according to the OTP password parameters is transmitted when registering.	4
No show password when error.	If there is no need to display the user's password on the screen of the mobile device with an authentication error, add 100 to the code. For example: 101, 102, 103, ...	1XX

OTP Password Parameters when Registering New Account

The OTP password string consists of the following lines separated by a semicolon (;):

OTPKey;ValidTime;Algorithm;PasswordLen

OTPKey - secret key that is used as basis for OTP password calculation. It should be compliant with determined algorithm and is transmitted as the string of key's hex-value ;

ValidTime - time of OTP password validity in seconds;




Algorithm - OTP password calculation algorithm. SHA1, SHA256, SHA512 are supported



PasswordLen - length of password in symbols, allowed values 1 - 8.

Example:

43ACCF A77B3620735D6D6E0068AFE2BA1059EE0A8D366A2150ED47B167BE32A393613F02A089F496C0
CE8FA5FFA106436D2FDDA72E0684D4A80E4F58520FCB7E;300;SHA1;6

QR Code Examples

Function	Example
Authorization QR code contents: <i>QRLOGIN</i> <i>L:V1</i> <i>qrlogin.tcon.com.ua/qrLogin</i> <i>b7e3d7a5707806c17a380c31a56c</i>	
New user registration with static password in mobile application QR code contents: <i>QRLOGIN</i> <i>NU:V1</i> <i>qrlogin.tcon.com.ua/qrLogin</i> <i>/loginsecret.xhtml</i> <i>admin</i> <i>admin</i> <i>1</i>	
New user registration with OTP password. The key is located in QR code. QR code contents: <i>QRLOGIN</i> <i>NU:V1</i> <i>qrlogin.tcon.com.ua/qrLogin</i> <i>/loginsecret.xhtml</i> <i>admin</i> <i>43ACCF A77B3620735D6D6E0068AFE2BA1059EE0A8D366A2</i> <i>150ED47B167BE32A393613F02A089F496C0CE8FA5FFA1064</i> <i>36D2FDDA72E0684D4A80E4F58520FCB7E;30;SHA1;6</i>	

4	
<p>New user registration with static password that is generated in the mobile application and is transmitted to the server. QR code contents: <i>QRLOGIN</i> <i>NU:V2</i> <i>qrlogin.tcon.com.ua/qrLogin</i> <i>/loginsecret.xhtml</i> <i>admin</i> <i>8;1;1;0</i> <i>2</i> <i>c08596ad07f529e123e0f8e3223b</i></p>	
<p>New user registration with OTP password the key for which one is is generated in the mobile application and is transmitted to the server. QR code contents: <i>QRLOGIN</i> <i>NU:V2</i> <i>qrlogin.tcon.com.ua/qrLogin</i> <i>/loginsecret.xhtml</i> <i>admin</i> <i>;30;SHA1;6</i> <i>4</i> <i>c08596ad07f529e123e0f8e3223b</i></p>	

Recommended Error Codes

qrLogin mobile application determines command result by HTTP status that the web resource returns to the http request.

Code	Identification	Objectives
200	OK	Requested action has successfully completed
400	BAD_REQUEST	Request is formed with error
403	FORBIDDEN	Access is denied. Authentication failed.
406	NOT_ACCEPTABLE	Request to update the password or key when registering new user with the password / key generation on the mobile device is not performed
408	REQUEST_TIMEOUT	Timeout for the internal operations of web resource.