API for QNAP QTS Authentication

Version: 4.2

Disclaimer

INFORMATION IN THIS DOCUMENT IS PROVIDED IN CONNECTION WITH QNAP® PRODUCTS. NO LICENSE, EXPRESS OR IMPLIED, BY ESTOPPEL OR OTHERWISE, TO ANY INTELLECTUAL PROPERTY RIGHTS IS GRANTED BY THIS DOCUMENT. EXCEPT AS PROVIDED IN QNAP'S TERMS AND CONDITIONS OF SALE FOR SUCH PRODUCTS, QNAP ASSUMES NO LIABILITY WHATSOEVER, AND QNAP DISCLAIMS ANY EXPRESS OR IMPLIED WARRANTY, RELATING TO SALE AND/OR USE OF QNAP PRODUCTS INCLUDING LIABILITY OR WARRANTIES RELATING TO FITNESS FOR A PARTICULAR PURPOSE, MERCHANTABILITY, OR INFRINGEMENT OF ANY PATENT, COPYRIGHT OR OTHER INTELLECTUAL PROPERTY RIGHT.

QNAP products are not intended for use in medical, life saving, life sustaining, critical control or safety systems, or in nuclear facility applications.

QNAP may make changes to product specifications at any time, without notice. The information here is subject to change without notice. Contact your local QNAP sales office or your distributor to obtain the latest product specifications before placing your product order.

QNAP, QNAP logo, QTS, myQNAPcloud, and VioStor are trademarks or registered trademarks of QNAP Systems, Inc. or its subsidiaries. Other names and brands may be claimed as the property of others.

Table of Contents

1. Overview

This document provides API for authentication .

2. System API

2.1 Authentication

Description:

Get sid for authentication by password.

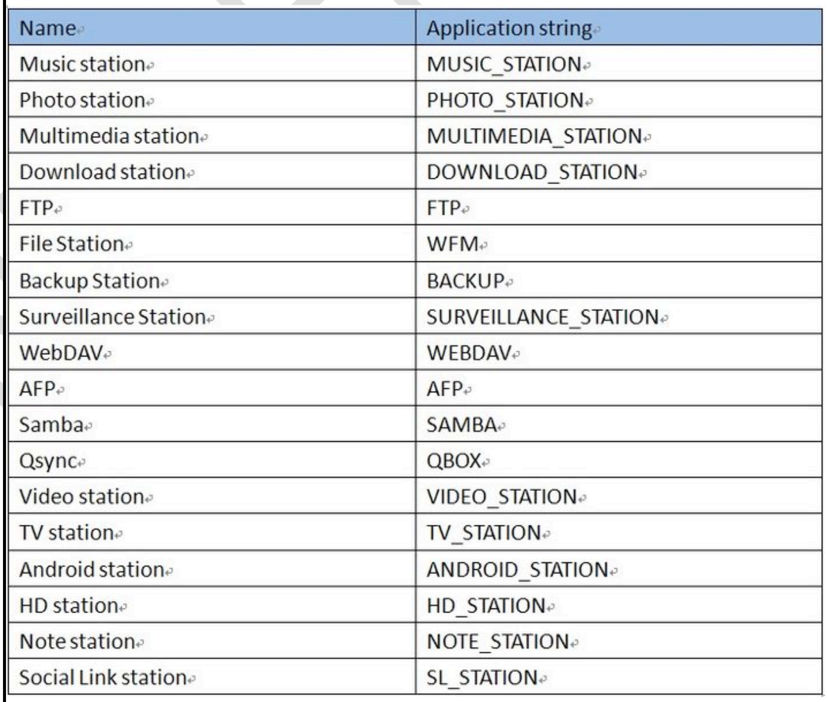
use Remember me, get qtoken for authentication.

Command:

http://IP:8080/cgi-bin/authLogin.cgi?user=${username}&pwd=${encode_string}&remme=${remme}&service=${service}&remote_ip=${remote_ip}&device=${device}&force_to_check_2sv={force_to_check_2sv}

http://IP:8080/cgi-bin/authLogin.cgi?user=${username}&plain_pwd=${pwd_in_plain_text}&remme=${remme}&service=${service}&remote_ip=${remote_ip}&device=${device}&force_to_check_2sv={force_to_check_2sv}

Variable	Description
${username}	Login user name
${encode_string}	Password (Please refer to attached 「get_sid.js」 to get ezEncode function)
${pwd_in_plain_text}	Password in plain text
${remme}	Optional: 1:return qtoken 0:clean qtoken
${force_to_check_2sv}	Optional: 1: Force to check 2-step verification including query from 127.0.0.1 Others: no force
${service}	Username/ Password authentication by specifying service name. When the value>=100, sid will NOT be generated.

<table>
<tr><td></td><td>101: Photo Station
102: Music Station
103: Video Station
100: Others
99: Force to check 2-step verification including query from 127.0.0.1</td></tr>
<tr><td>${remote_ip}</td><td>Optional; Only for localhost(127.0.0.1), specify the remote address</td></tr>
<tr><td>${device}</td><td>Optional;
Specify the client' device name</td></tr>
<tr><td>${check_privilege}</td><td>Optional;
Specify the application string to check

<table>
<tr><th>Name</th><th>Application string</th></tr>
<tr><td>Music station</td><td>MUSIC_STATION</td></tr>
<tr><td>Photo station</td><td>PHOTO_STATION</td></tr>
<tr><td>Multimedia station</td><td>MULTIMEDIA_STATION</td></tr>
<tr><td>Download station</td><td>DOWNLOAD_STATION</td></tr>
<tr><td>FTP</td><td>FTP</td></tr>
<tr><td>File Station</td><td>WFM</td></tr>
<tr><td>Backup Station</td><td>BACKUP</td></tr>
<tr><td>Surveillance Station</td><td>SURVEILLANCE_STATION</td></tr>
<tr><td>WebDAV</td><td>WEBDAV</td></tr>
<tr><td>AFP</td><td>AFP</td></tr>
<tr><td>Samba</td><td>SAMBA</td></tr>
<tr><td>Qsync</td><td>QBOX</td></tr>
<tr><td>Video station</td><td>VIDEO_STATION</td></tr>
<tr><td>TV station</td><td>TV_STATION</td></tr>
<tr><td>Android station</td><td>ANDROID_STATION</td></tr>
<tr><td>HD station</td><td>HD_STATION</td></tr>
<tr><td>Note station</td><td>NOTE_STATION</td></tr>
<tr><td>Social Link station</td><td>SL_STATION</td></tr>
</table></td></tr>
</table>

Note: encode_string = ezEncode(utf16to8('${real_password}'))

Example:

http://IP:8080/cgi-bin/authLogin.cgi?user=admin&pwd=YWRtaW4%3D&remme=1

(if real_password is "admin", it will be encoded as "YWRtaW4%3D")

Authentication by service

http://127.0.0.1:8080/cgi-bin/authLogin.cgi?plain_pwd=admin&user=admin&remote_ip=172.17.20.49&device=richardnb

Return value:

Example of successful authentication:

```
<?xml version="1.0" encoding="UTF-8" ?>
        <QDocRoot version="1.0">
        <qtoken><![CDATA[1e29b890910e8135f1692ed4030256fe]]></qtoken>  <= here is qtoken, if remme=0 hide
        <authPassed><![CDATA[1]]></authPassed>
        <authSid><![CDATA[ral08opo]]></authSid>              <= here is sid, generated when service>=100 was not
set
        <isAdmin><![CDATA[1]]></isAdmin>
        </QDocRoot>
```

Example of failed authentication:

```
         <QDocRoot version="1.0">
         <qtoken>1e29b890910e8135f1692ed4030256fe</qtoken>
        <authPassed>0</authPassed>
        <errorValue>-1</errorValue>
        </QDocRoot>
```

Authorisation by service

http://${ip}:8080/cgi-bin/authLogin.cgi?plain_pwd=admin&user=aix&remote_ip=172.17.20.49&service=104&device=aixchou&check_privilege=VIDEO_STATION

Return value:

Example of permission deny:

```
<?xml version="1.0" encoding="UTF-8" ?>
         <QDocRoot version="1.0">
         <doQuick><![CDATA[]]></doQuick>
         <is_booting><![CDATA[0]]></is_booting>
         <mediaReady><![CDATA[1]]></mediaReady>
         <SMBFW><![CDATA[0]]></SMBFW>
         <PermissionDeny><![CDATA[1]]></PermissionDeny>     <= permission deny
```

```
<authPassed><![CDATA[0]]></authPassed>
<errorValue><![CDATA[-1]]></errorValue>
<username><![CDATA[aix]]></username>
<ts><![CDATA[85022350]]></ts>
<fwNotice><![CDATA[0]]></fwNotice>
<title><![CDATA[]]></title>
<content><![CDATA[]]></content>
<psType><![CDATA[1]]></psType>
<showVersion><![CDATA[0]]></showVersion>
<show_link><![CDATA[1]]></show_link>
</QDocRoot>
```

2.2 Authentication with qtoken

Description:

Get sid for authentication by qtoken

Command:

http://IP:8080/cgi-bin/authLogin.cgi?user=${username}&qtoken=${qtoken}&remme=${remme}

Variable	Description
${username}	Login user name
${qtoken}	qtoken
${remme}	Optional: 1:when login by qtoken without return qtoken tag 0:clean qtoken

Example:

http://IP:8080/cgi-bin/authLogin.cgi?user=admin&qtoken=1e29b890910e8135f1692ed4030256fe&remme=1

Return value:

Example of successful authentication:

```
<?xml version="1.0" encoding="UTF-8" ?>
      <QDocRoot version="1.0">
      <authPassed><![CDATA[1]]></authPassed>
      <authSid><![CDATA[ral08opo]]></authSid>          <= here is sid
      <isAdmin><![CDATA[1]]></isAdmin>
      </QDocRoot>
```

Example of failed authentication:

```
 <QDocRoot version="1.0">
<authPassed>0</authPassed>
<errorValue>-1</errorValue>
</QDocRoot>
```

2.3 Authentication with 2 step verification

Description:

If 2-step verification enabled, get sid for 1st authentication by password and 2nd authentication by security code(6 digits) / emergency security code(8 digits) / emergency answer.

In 4.2.0, only support QTS system / File Station login.

HTTP Request from 127.0.0.1 can get sid without 2-step verification.

2.3.1. 1st verification

Description:

1st verification

Command:

http://172.17.20.20:8080/cgi-bin/authLogin.cgi?pwd=${encode_string}&r=0.802557202605028&remme=${remme}&serviceKey=1&user=${username}

Variable	Description
${username}	Login user name
${encode_string}	Password (Please refer to attached 「get_sid.js」 to get ezEncode function)
${remme}	Optional: 1:return qtoken 0:clean qtoken

Note: encode_string = ezEncode(utf16to8('${real_password}'))

Example:

http://172.17.20.20:8080/cgi-bin/authLogin.cgi?pwd=YWRtaW4%3D&r=0.02297725847566645&remme=1&serviceKey=1&user=admin

(if real_password is "admin", it will be encoded as "YWRtaW4%3D")

Return value:

Example of successful for 1st authentication:

```
<QDocRoot version="1.0">
        <doQuick></doQuick>
        <is_booting>0</is_booting>
        <mediaReady>1</mediaReady>
        <SMBFW>0</SMBFW>
        <authPassed>0</authPassed>
        <need_2sv>1</need_2sv>
        <lost_phone>1</lost_phone>
        <emergency_try_count>0</emergency_try_count>
        <emergency_try_limit>5</emergency_try_limit>
        <username>admin</username>
        <groupname>administrators</groupname>
        <ts>88323841</ts>
        <fwNotice>0</fwNotice>
        <title></title>
        <content></content>
        <psType>0</psType>
        <showVersion>0</showVersion>
        <show_link>1</show_link>
</QDocRoot>
```

Example of failed for 1st authentication:

```
<QDocRoot version="1.0">
        <doQuick></doQuick>
        <is_booting>0</is_booting>
        <mediaReady>1</mediaReady>
        <SMBFW>0</SMBFW>
        <authPassed>0</authPassed>
        <errorValue>-1</errorValue>
        <username>admin</username>
        <ts>88323841</ts>
        <fwNotice>0</fwNotice>
        <title></title>
        <content></content>
        <psType>0</psType>
        <showVersion>0</showVersion>
        <show_link>1</show_link>
</QDocRoot>
```

Below are the tags inside process list.

Tag name	Type	Description
authPassed	int	authentication result 0:fail 1:success
need_2sv	int	whether user need 2-step verification 1:need 2-step verification if need_2sv not return, please refer general Authentication
lost_phone	int	click "Verify another way" will lead to the following Optional: 1:send emergency e-mail that contains emergency security code(8 digits) 2:emergency question handler
emergency_try_count	int	if "lost_phone" is 1, it means the number of send emergency mail if "lost_phone" is 2, it means the number of answer incorrectly
emergency_try_limit	int	if "lost_phone" is 1, it means the maximum number of send emergency mail if "lost_phone" is 2, it means the maximum number of answer incorrectly so far, we define "emergency_try_limit" is 5

2.3.2. 2nd verification

Description:

verify security code(6 digits) or emergency security code(8 digits)

Command:

http://172.17.20.20:8080/cgi-bin/authLogin.cgi?pwd=${encode_string}&r=0.3938051044582034&remme=${remme}&security_code=${security_code}&serviceKey=1&user=${username}

Variable	Description
${username}	Login user name
${encode_string}	Password (Please refer to attached 「get_sid.js」 to get ezEncode function)
${remme}	Optional: 1:return qtoken 0:clean qtoken
${security_code}	security code(6 digits) while emergency handler is "Send emergency e-mail", it can be emergency security code(8 digits)

Note: encode_string = ezEncode(utf16to8('${real_password}'))

Example:

http://172.17.20.20:8080/cgi-bin/authLogin.cgi?pwd=YWRtaW4%3D&r=0.3938051044582034&security_code=215238&serviceKey=1&user=admin

(if real_password is "admin", it will be encoded as "YWRtaW4%3D")

Return value:

Example of successful for 2nd authentication:

```
<QDocRoot version="1.0">
        <doQuick></doQuick>
        <is_booting>0</is_booting>
        <mediaReady>1</mediaReady>
        <SMBFW>0</SMBFW>
        <authPassed>1</authPassed>
        <authSid>mxz01een</authSid>
        <need_2sv>1</need_2sv>
        <lost_phone>1</lost_phone>
        <emergency_try_count>0</emergency_try_count>
        <emergency_try_limit>5</emergency_try_limit>
        <isAdmin>1</isAdmin>
        <username>admin</username>
        <groupname>administrators</groupname>
        <ts>88323841</ts>
        <fwNotice>0</fwNotice>
        <title></title>
        <content></content>
        <psType>0</psType>
        <showVersion>0</showVersion>
        <show_link>1</show_link>
</QDocRoot>
```

Example of failed for 2nd authentication:

```
<QDocRoot version="1.0">
        <doQuick></doQuick>
        <is_booting>0</is_booting>
        <mediaReady>1</mediaReady>
        <SMBFW>0</SMBFW>
        <authPassed>0</authPassed>
        <need_2sv>1</need_2sv>
        <lost_phone>1</lost_phone>
        <emergency_try_count>0</emergency_try_count>
        <emergency_try_limit>5</emergency_try_limit>
        <date_time>
                <timezone>(GMT+08:00) Taipei</timezone>
                <timestamp>1432803710</timestamp>
                <date_format_index>1</date_format_index>
                <time_format>24</time_format>
        </date_time>
        <username>admin</username>
        <groupname>administrators</groupname>
        <ts>97469902</ts>
        <fwNotice>0</fwNotice>
        <SUID>6801bd1901459a79a9a39eb6c24da8fb</SUID>
        <title></title>
        <content></content>
        <psType>1</psType>
        <showVersion>0</showVersion>
        <show_link>1</show_link>
</QDocRoot>
```

most parameter can refer 1st verification

Below are the other tags inside process list.

Tag name	Type	Description
timezone	string	Time Zone, ex."(GMT+08:00) Taipei"
timestamp	int	seconds since Jan 01 1970
date_format_index	int	date format index

		1 : year/month/day, 2 : year.month.day, 3 : year-month-day, 4 : month/day/year, 5 : month.day.year, 6 : month-day-year, 7 : day/month/year, 8 : day.month.year, 9 : day-month-year
time_format	int	time format : 24 / 12

2.3.3. Send Emergency E-mail

Description:

send emergency e-mail request (while 1st verification or 2nd verification return value "lost_phone" is 1)

Command:

http://172.17.20.20:8080/cgi-bin/authLogin.cgi?pwd=${encode_string}&r=0.3938051044582034&remme=${remme}&send_mail=1&serviceKey=1&user=${username}

Variable	Description
${username}	Login user name
${encode_string}	Password (Please refer to attached 「get_sid.js」 to get ezEncode function)
${remme}	Optional: 1:return qtoken 0:clean qtoken

Note: encode_string = ezEncode(utf16to8('${real_password}'))

Example:

http://172.17.20.20:8080/cgi-bin/authLogin.cgi?pwd=YWRtaW4%3D&r=0.3938051044582034&send_mail=1&serviceKey=1&user=admin

(if real_password is "admin", it will be encoded as "YWRtaW4%3D")

Return value:

```
<QDocRoot version="1.0">
        <doQuick></doQuick>
        <is_booting>0</is_booting>
        <mediaReady>1</mediaReady>
        <SMBFW>0</SMBFW>
        <send_result>1</send_result>
        <emergency_try_count>3</emergency_try_count>
        <emergency_try_limit>5</emergency_try_limit>
        <username>admin</username>
        <groupname>administrators</groupname>
        <ts>88323841</ts>
        <fwNotice>0</fwNotice>
        <title></title>
        <content></content>
        <psType>0</psType>
        <showVersion>0</showVersion>
        <show_link>1</show_link>
</QDocRoot>
```

Below are the tags inside process list.

Tag name	Type	Description
send_result	int	send mail result 1:success 0:fail -1:the email notification service is not enabled
emergency_try_count	int	it means the number of send emergency mail
emergency_try_limit	int	it means the maximum number of send emergency mail so far, we define "emergency_try_limit" is 5

2.3.4. Get Security Question

Description:

get security question (while 1st verification or 2nd verification return value "lost_phone" is 2)

Command:

http://172.17.20.20:8080/cgi-bin/authLogin.cgi?pwd=${encode_string}&r=0.3938051044582034&remme=${remme}&get_question=1&serviceKey=1&user=${username}

Variable	Description
${username}	Login user name
${encode_string}	Password (Please refer to attached 「get_sid.js」 to get ezEncode function)
${remme}	Optional: 1:return qtoken 0:clean qtoken

Note: encode_string = ezEncode(utf16to8('${real_password}'))

Example:

http://172.17.20.20:8080/cgi-bin/authLogin.cgi?get_question=1&pwd=YWRtaW4%3D&r=0.3938051044582034&serviceKey=1&user=admin

(if real_password is "admin", it will be encoded as "YWRtaW4%3D")

Return value:

```
<QDocRoot version="1.0">
        <doQuick></doQuick>
        <is_booting>0</is_booting>
        <mediaReady>1</mediaReady>
        <SMBFW>0</SMBFW>
        <security_question_no>4</security_question_no>
        <security_question_text>how are you?</security_question_text>
        <username>admin</username>
        <groupname>administrators</groupname>
        <ts>88323841</ts>
        <fwNotice>0</fwNotice>
        <title></title>
        <content></content>
        <psType>0</psType>
        <showVersion>0</showVersion>
        <show_link>1</show_link>
```

</QDocRoot>

Below are the tags inside process list.

Tag name	Type	Description
security_question_no	int	question no 1: frontend will show "What is your pet's name?" 2: frontend will show "What is your favorite sport?" 3: frontend will show "What is your favorite color?" 4: Custom question
security_question_text	string	Custom question (only for "security_question_no" is 4)

2.3.5. Get Security Question for Mobile App

Description:

get security question for Mobile App (while 1st verification or 2nd verification return value "lost_phone" is 2)

Command:

http://172.17.20.20:8080/cgi-bin/authLogin.cgi?pwd=${encode_string}&r=0.3938051044582034&remme=${remme}&get_question=1&serviceKey=1&user=${username}&q_lang=${q_lang}

Variable	Description
${username}	Login user name
${encode_string}	Password (Please refer to attached 「get_sid.js」 to get ezEncode function)
${remme}	Optional: 1:return qtoken 0:clean qtoken

${q_lang}	Languages for seucruty question {CZE, DAN, DUT, ENG, ESM, FIN, FRE, GER, GRK, HUN, ITA, JPN, KOR, NOR, POL, POR, ROM, RUS, SCH, SPA, SWE, TCH, THA, TUR}

Note: encode_string = ezEncode(utf16to8('${real_password}'))

Example:

http://172.17.20.20:8080/cgi-bin/authLogin.cgi?get_question=1&pwd=YWRtaW4%3D&r=0.3938051044582034&serviceKey=1&user=admin&q_lang=ENG

(if real_password is "admin", it will be encoded as "YWRtaW4%3D")

Return value:

```
<QDocRoot version="1.0">
        <doQuick></doQuick>
        <is_booting>0</is_booting>
        <mediaReady>1</mediaReady>
        <SMBFW>0</SMBFW>
        <security_question_no>4</security_question_no>
        <system_question_text>how are you?</system_question_text>
        <security_question_text>how are you?</security_question_text>
        <username>admin</username>
        <groupname>administrators</groupname>
        <ts>88323841</ts>
        <fwNotice>0</fwNotice>
        <title></title>
        <content></content>
        <psType>0</psType>
        <showVersion>0</showVersion>
        <show_link>1</show_link>
</QDocRoot>
```

Below are the tags inside process list.

Tag name	Type	Description
security_question_no	int	question no 1: frontend will show "What is your pet's name?" 2: frontend will show "What is your favorite sport?" 3: frontend will show "What is your favorite color?" 4: Custom question

system_question_text	string	security question text (for mobile app)
security_question_text	string	Custom question (only for "security_question_no" is 4)

2.3.6. Security Question Authentication

Description:

verify security answer (while 1st verification or 2nd verification return value "lost_phone" is 2)

Command:

http://172.17.20.20:8080/cgi-bin/authLogin.cgi?pwd=${encode_string}&r=0.3938051044582034&remme=${remme}&security_answer=${security_answer}&serviceKey=1&user=${username}

Variable	Description
${username}	Login user name
${encode_string}	Password (Please refer to attached 「get_sid.js」 to get ezEncode function)
${remme}	Optional: 1:return qtoken 0:clean qtoken
${security_answer}	string

Note: encode_string = ezEncode(utf16to8('${real_password}'))

Example:

http://172.17.20.20:8080/cgi-bin/authLogin.cgi?pwd=YWRtaW4%3D&r=0.4000929836850201&security_answer=fine&serviceKey=1&user=admin

(if real_password is "admin", it will be encoded as "YWRtaW4%3D")

Return value: Example of successful for security question authentication:

```
<QDocRoot version="1.0">
        <doQuick></doQuick>
        <is_booting>0</is_booting>
        <mediaReady>1</mediaReady>
        <SMBFW>0</SMBFW>
        <authPassed>1</authPassed>
        <authSid>m9x71gxw</authSid>
        <emergency_try_count>1</emergency_try_count>
        <emergency_try_limit>5</emergency_try_limit>
        <isAdmin>1</isAdmin>
        <username>admin</username>
        <groupname>administrators</groupname>
        <ts>88323841</ts>
        <fwNotice>0</fwNotice>
        <title></title>
        <content></content>
        <psType>0</psType>
        <showVersion>0</showVersion>
        <show_link>1</show_link>
</QDocRoot>
```

Example of failed for security question authentication:

```
<QDocRoot version="1.0">
        <doQuick></doQuick>
        <is_booting>0</is_booting>
        <mediaReady>1</mediaReady>
        <SMBFW>0</SMBFW>
        <authPassed>0</authPassed>
        <emergency_try_count>1</emergency_try_count>
        <emergency_try_limit>5</emergency_try_limit>
        <username>admin</username>
        <groupname>administrators</groupname>
        <ts>88323841</ts>
        <fwNotice>0</fwNotice>
        <title></title>
        <content></content>
        <psType>0</psType>
        <showVersion>0</showVersion>
        <show_link>1</show_link>
</QDocRoot>
```

here's parameter can refer 1st verification