

ShowRSS Downloader on QNAP

- [Overview](#)
- [Variables](#)
- [Scripts](#)
 - [showRss.php](#)

Overview

Variables

Variable	Description	Details
pass	admin password	base64encoded password
EMAIL_TO/EMAIL_FROM		
SHOWRSS_FEED		
TEMP_FOLDER		
DEST_FOLDER		

Scripts

In order to make it easier to run your php scripts, we can wrap them in a shell script like the following:

```
#!/bin/sh

PHP=/mnt/ext/opt/apache/bin/php
SCRIPT_PATH=/share/homes/xxx/scripts/
cd $SCRIPT_PATH
$PHP $SCRIPT_PATH/showRss.php <ACTION>
```

showRss.php

Usage:

> php showRss.php <download/removeCompleted/listDownloads/removeAll>

```
showrss.php

<?php

//custom variables
$pass=urlencode("xxx_BASE64ENCODEDPASSWORD_xxx");
$EMAIL_TO="xxx@xxx.com";
$EMAIL_FROM="xxxx@xxx.com";
$SHOWRSS_FEED="http://showrss.info/user/xxx.rss?magnets=true&namespaces=true&name=clean&quality=hd&re=null";
$TEMP_FOLDER="tv/tmp";
$DEST_FOLDER="tv/incoming";

//default variables
$baseURL="http://localhost:8080";
$db_FILE_NAME="db_showrss.json";
```

```

$EMAIL_SUBJECT_DOWNLOADED="DOWNLOADED - ";
$EMAIL_SUBJECT_STARTED="DOWNLOAD STARTED - ";
$EMAIL_SUBJECT_FAILED="FAILED TO DOWNLOAD - ";
$EMAIL_SUBJECT_REMOVED="CANCELLED DOWNLOAD - ";

//login
$sid=loginToQnap();

// Process Command Line
$cmd="download";
if ($argc > 1) {
    $cmd=$argv[1];
};

switch ($cmd) {
    case "download":
        echo "DOWNLOAD:\n";
        download();
        break;
    case "removeCompleted":
        echo "REMOVE COMPLETED:\n";
        removeCompleted();
        break;
    case "listDownloads":
        echo "LIST DOWNLOADS:\n";
        listDownloads();
        break;
    case "removeAll":
        echo "REMOVE ALL:\n";
        removeAll();
        break;
    default:
        showUsage();
        exit(1);
}

//done
exit(0);

//*****
// download
//*****
function download(){
    global $SHOWRSS_FEED, $DB_FILE_NAME, $baseUrl, $sid, $EMAIL_TO, $EMAIL_FROM, $EMAIL_SUBJECT_DOWNLOADED;

    //get showRSS Feed
    $xml = file_get_contents($SHOWRSS_FEED);
    //echo $xml;

    //process xml
    $xml = simplexml_load_file($SHOWRSS_FEED);

    $showArray = array();

    $showdb = json_decode(file_get_contents($DB_FILE_NAME), true);
    if($showdb==null){
        $showdb = array();
    }

    // loop through
    foreach ($xml->channel->item as $item) {
        $title = $item->title;
        $link = $item->link;
        //echo "title=" . $title ."\n";

        $pattern = "/(\d{1,2})x(\d{1,2})/";
        preg_match($pattern, $title, $matches);
        if(sizeof($matches)>=3){
            $season=intval($matches[1]);
            $episode=intval($matches[2]);
            $show = trim(substr($title,0,strrpos($title,$matches[0])));

```

```

    $showAndEpisode=sprintf("%s %02dx%02d", $show,$season,$episode);
}else{
    $show=(string)$title;
    $showAndEpisode=(string)$title;
    $season=0;
    $episode=0;
}

//echo "SHOW: " . $show . "\n";
//echo "SEASON: " . $season . "\n";
//echo "EPISODE: " . $episode . "\n";

$showInfo=array($showAndEpisode, $show, $season, $episode, $link);
array_push($showArray, $showInfo);
} //for

sort($showArray);
//print_r($showArray);

foreach ($showArray as $show) {

    //echo "show: " . $show[0] . "\n";
    //add torrent

    $found =false;
    foreach($showdb as &$showEntry){
        $showName = $showEntry[0];

        if($showName==$show[1]){
            $found=true;
            //echo "Found " . $show[1] . " in showdb \n";
            $newSeason=$show[2];
            $newEpisode=$show[3];
            $lastSeason=$showEntry[1];
            $lastEpisode=$showEntry[2];
            //echo "--- " . $lastSeason . "x" . $lastEpisode . " -> " . $newSeason . "x" . $newEpisode ."\n";

            if($lastSeason == $newSeason && $newEpisode > $lastEpisode){
                if(downloadShow($show)){
                    $showEntry[2]= $newEpisode;
                    echo "U " . $showEntry[0] . " " . $showEntry[1] . "x" . $showEntry[2] . "\n";
                }else{
                    echo "F " . $showEntry[0] . " " . $showEntry[1] . "x" . $showEntry[2] . "\n";
                }
            }elseif ($newSeason > $lastSeason ){
                if(downloadShow($show)){
                    $showEntry[1]= $newSeason;
                    $showEntry[2]= $newEpisode;
                    echo "U+ " . $showEntry[0] . " " . $showEntry[1] . "x" . $showEntry[2] . "\n";
                }else{
                    echo "F+ " . $showEntry[0] . " " . $showEntry[1] . "x" . $showEntry[2] . "\n";
                }
            }else{
                // echo "WTF " . $lastSeason . "x" . $lastEpisode . " -> " . $newSeason . "x" . $newEpisode ."\n";
            }
            break;
        }
    }

    //set latest season/episode
    if ($found==false) {

        if(downloadShow($show)){
            echo "A " . $show[1] . " " . $show[2] . "x" . $show[3] . "\n" ;
            $showDetails=array($show[1], $show[2], $show[3]);
            array_push($showdb, $showDetails);
        }else{
            echo "F " . $show[1] . " " . $show[2] . "x" . $show[3] . "\n" ;
        }
    }
}

```

```

}

//save showdb to file
//print_r($showdb);

file_put_contents($DB_FILE_NAME,json_encode($showdb));
}

//*****
// downloadShow
//*****
function downloadShow($show){
    global $baseUrl,$sid, $TEMP_FOLDER, $DEST_FOLDER;

    $showAndEpisode = $show[1] . " " . $show[2] . "x" . $show[3];
    //echo "D: " . $showAndEpisode . "\n";
    //echo "L: " . $show[4] . "\n\n";

    $temp=$TEMP_FOLDER;
    $dest=$DEST_FOLDER . "/" . $show[1];

    $realDest = "/share/" . $dest;
    if (!file_exists($realDest)) {
        mkdir($realDest, 0777, true);
    }

    if($show[2]>0){
        $dest=$dest . "/Season " . $show[2];
    }
    //echo "Dest: " . $dest . "\n";

    $realDest = "/share/" . $dest;
    if (!file_exists($realDest)) {
        mkdir($realDest, 0777, true);
    }

    $url = $baseUrl . '/downloadstation/V4/Task/AddUrl?sid=' . $sid . '&temp=' . urlencode($temp) . '&move=' .
    urlencode($dest) . '&url=' . urlencode($show[4]);
    //echo "URL: " . $url . "\n";
    $resp=file_get_contents($url);
    //echo $resp;
    if($resp=='{"error":0}'){
        //sendNotification($showAndEpisode,"started");
        return true;
    }else{
        sendNotification($showAndEpisode,"failed");
        return false;
    }
}

//*****
// sendNotification
//*****
function sendNotification($showAndEpisode,$status){
    global $EMAIL_SUBJECT_DOWNLOADED, $EMAIL_SUBJECT_STARTED,$EMAIL_SUBJECT_REMOVED, $EMAIL_SUBJECT_FAILED,
    $EMAIL_FROM, $EMAIL_TO;

    if($status == "started"){
        $subject= "" . $EMAIL_SUBJECT_STARTED . $showAndEpisode . "";
    }else if($status == "downloaded"){
        $subject= "" . $EMAIL_SUBJECT_DOWNLOADED . $showAndEpisode . "";
    }else if($status == "failed"){
        $subject= "" . $EMAIL_SUBJECT_FAILED . $showAndEpisode . "";
    }else if($status == "removed"){
        $subject= "" . $EMAIL_SUBJECT_REMOVED . $showAndEpisode . "";
    }else{
        $subject= "" . "ERROR " . $showAndEpisode . "";
    }
}

$cmd= getCwd() . "/sendEmail.sh " . $subject . " " . $EMAIL_FROM . " " . $EMAIL_TO . " " . $subject;

```

```

    shell_exec($cmd);
}

//*****
// removeCompleted
//*****
function removeCompleted(){
    global $baseUrl, $sid, $EMAIL_TO, $EMAIL_FROM, $EMAIL_SUBJECT_DOWNLOADED;

    $resp=file_get_contents($baseUrl . '/downloadstation/V4/Task/Query?
start=0&limit=25&status=completed&from=0&to=25&sid=' . $sid);
    //echo $resp . "\n";

    $json = json_decode($resp);
    $data = $json->{'data'};
    //var_dump($data);
    file_put_contents("/var/log/download.log",$json,FILE_APPEND | LOCK_EX);
    //echo "CWD: " . getCwd() . "\n";

    if(count($data)==0){
        echo "- no completed downloads\n";
    }else{

        foreach($data as $torrent){
            $name= $torrent->{'source_name'};
            $hash=$torrent->{'hash'};
            echo "removing torrent: " . $name . "\n";
            //echo "hash: " . $hash . "\n";

            if(endsWith($name,".torrent")){
                // echo "torrent file - just delete\n";
            }else{
                sendNotification($name,"downloaded");
            }

            //remove torrent
            $resp=file_get_contents($baseUrl . '/downloadstation/V4/Task/Remove?clean=0&hash=' . $hash . '&sid=' .
$sid);
        }
    }
}
//*****
// removeAll
//*****
function removeAll(){
    global $baseUrl, $sid, $EMAIL_TO, $EMAIL_FROM, $EMAIL_SUBJECT_DOWNLOADED;

    $resp=file_get_contents($baseUrl . '/downloadstation/V4/Task/Query?
start=0&limit=100&status=all&from=0&to=100&sid=' . $sid);

    $json = json_decode($resp);
    $data = $json->{'data'};

    if(count($data)==0){
        echo "- no downloads\n";
    }else{
        foreach($data as $torrent){
            $name= $torrent->{'source_name'};
            $hash=$torrent->{'hash'};

            if(endsWith($name,".torrent")){
                // echo "torrent file - just delete\n";
            }else{
                echo "- " . $name . "\n";
                //echo "file - notify completion\n";
                //sendNotification($name,"removed");
            }

            //remove torrent
            $resp=file_get_contents($baseUrl . '/downloadstation/V4/Task/Remove?clean=0&hash=' . $hash . '&sid=' .

```

```

$cid);
    }
}
}

//*****
// listDownloads
//*****
function listDownloads(){
    global $baseUrl, $cid;

    $resp=file_get_contents($baseUrl . '/downloadstation/V4/Task/Query?
start=0&limit=100&status=all&from=0&to=100&cid=' . $cid);
    //echo $resp . "\n";

    $json = json_decode($resp);
    $data = $json->{'data'};

    if(count($data)==0){
        echo "- no downloads\n";
    }else{
        foreach($data as $torrent){
            $name= $torrent->{'source_name'};
            echo "torrent: " . $name . "\n";
        }
    }
}

//*****
// loginToQnap
//*****
function loginToQnap(){
    global $baseUrl, $pass;

    //login
    $vars = 'user=admin&pass=' . $pass;
    $ch = curl_init( $baseUrl . '/downloadstation/V4/Misc/Login' );
    curl_setopt( $ch, CURLOPT_POST, 1);
    curl_setopt( $ch, CURLOPT_POSTFIELDS, $vars);
    curl_setopt( $ch, CURLOPT_FOLLOWLOCATION, 1);
    curl_setopt( $ch, CURLOPT_HEADER, 0);
    curl_setopt( $ch, CURLOPT_RETURNTRANSFER, 1);
    $resp = curl_exec( $ch );
    //echo $resp;

    $cid = json_decode($resp)->{'cid'};
    //echo "cid='" . $cid . "'\n";

    return $cid;
}

//*****
// endsWith
//*****
function endsWith($haystack, $needle)
{
    $length = strlen($needle);
    if ($length == 0) {
        return true;
    }
    return (substr($haystack, -$length) === $needle);
}

//*****
// showUsage
//*****
function showUsage(){
    global $argv;
    echo "Usage: php " . $argv[0] . " download/removeCompleted/listDownloads/removeAll\n";
}

```

?>

sendEmail.sh

sendEmail.sh

```
#!/bin/sh

SUBJECT=$1
TO=$2
FROM=$3
CONTENT=$4

echo "Subject: $SUBJECT"
echo "From: $FROM"
echo "To: $TO"
echo "CONTENT $CONTENT"

echo "Subject: $SUBJECT" > mail.txt
echo "From: $FROM" >> mail.txt
echo "To: $TO" >> mail.txt
echo "" >> mail.txt
echo $CONTENT >> mail.txt

cat mail.txt |sendmail -t
#rm mail.txt
```