

# How To Access Data With PHP

The following PHP code example demonstrates how to configure a connection to download data from an Earthdata Login enabled server. The first example takes advantage of the .netrc file for configuring usernames and passwords (see [this page](#) for details on how to set up a .netrc file). If this is not an option, the second code example demonstrates how to provide credentials directly (you will need to find a secure way to configure the credentials).

```
$url = "http://e4ftl01.cr.usgs.gov/ASTT/AST_L1T.003/2016.08.22
/AST_L1T_00308222016003413_20160823093712_24613.hdf.xml";
$COOKIE_FILE = '<path to a file for storing cookies (optional)>';

/* Initialize the CURL command */
$ch = curl_init();
curl_setopt($ch, CURLOPT_URL, $url);
curl_setopt($ch, CURLOPT_FOLLOWLOCATION, true);
curl_setopt($ch, CURLOPT_NETRC, true);
curl_setopt($ch, CURLOPT_RETURNTRANSFER, true);
//curl_setopt($ch, CURLOPT_VERBOSE, true); /* Useful for debugging */

/*
 * For more efficient operation, preserve session cookies across
 * executions. If cookie files cannot be used, comment out the next
 * two lines and uncomment the third.
 */
curl_setopt($ch, CURLOPT_COOKIEFILE, $COOKIE_FILE);
curl_setopt($ch, CURLOPT_COOKIEJAR, $COOKIE_FILE);
//curl_setopt($ch, CURLOPT_COOKIEFILE, '');

/* Execute the request */
$result = curl_exec($ch);
$status = curl_getinfo($ch, CURLINFO_HTTP_CODE);
curl_close($ch);

if( $status != 200 ) {
    /* ERROR! */
}
else {
    /* This contains the returned data for the URL */

    echo "RESULT: $result\n";
}
}
```

The following example provides credentials rather than relying upon the .netrc file.

```

$url = 'http://e4ftl01.cr.usgs.gov/ASTT/AST_L1T.003/2016.08.22
/AST_L1T_00308222016003413_20160823093712_24613.hdf.xml';
$username = '<Earthdata Login username>';
$password = '<Earthdata Login password>'; /* Must be secured */
$COOKIE_FILE = '<path to a file for storing cookies (optional)>';

/* Initialize the CURL command */

$ch = curl_init();
curl_setopt($ch, CURLOPT_URL, $url);
curl_setopt($ch, CURLOPT_FOLLOWLOCATION, true);
curl_setopt($ch, CURLOPT_NETRC, false);
curl_setopt($ch, CURLOPT_RETURNTRANSFER, true);
//curl_setopt($ch, CURLOPT_VERBOSE, true); /* Useful for debugging */

/*
 * For more efficient operation, preserve session cookies across
 * executions. If cookie files cannot be used, comment out the next
 * two lines and uncomment the third.
 */
curl_setopt($ch, CURLOPT_COOKIEFILE, $COOKIE_FILE);
curl_setopt($ch, CURLOPT_COOKIEJAR, $COOKIE_FILE);
//curl_setopt($ch, CURLOPT_COOKIEFILE, '');

/* Execute the request */

$result = curl_exec($ch);
$status = curl_getinfo($ch, CURLINFO_HTTP_CODE);

if ( $status == '401' ) {
    /*
     * We are required to authenticate, so add in the authorization header
     * and continue the request.
     */
    $auth = base64_encode("$username:$password");
    $headers = array("Authorization: BASIC $auth");
    curl_setopt($ch, CURLOPT_HTTPHEADER, $headers);
    $result = curl_exec($ch);
    $status = curl_getinfo($ch, CURLINFO_HTTP_CODE);
}
curl_close($ch);

if( $status != 200 )
{
    // ERROR!
}
else {
    echo "RESULT: $result\n";
}

```