

Re: Automatic login to a web page

Source: <http://coding.derkeiler.com/Archive/Perl/comp.lang.perl.misc/2006-10/msg01710.html>

- *From:* "Brian Wilkins" <bwilkins@xxxxxxxx>
 - *Date:* 24 Oct 2006 12:07:43 -0700
-

zentara wrote:

On 23 Oct 2006 23:32:29 -0700, "Jagjeet_Singh" <jagjeet.malhi@xxxxxxxx> wrote:

I am new to perl and do not have any idea about this language.
I need your help to solve my problem.

I have internet access on my desktop but each time my machine got rebooted I need to log-in on my ISP web's page by providing username,password and submit the button.

But I do not know perl, If this code is really small, can anyone post it ..

Or if it takes time to write this then please provide me some from where I can learn something to write this code.

like -- I use this url "http://my_isp_ip_address/indexmail.php and there are only 3 fields.

1 -- text box for username

2 -- text box for password

3 -- "Submit" button

I'll give you some tips. First get something like tcpick or ethereal, so you can watch in realtime what is going on during your login. Do it from their webpage a few times and see what is actually happening.

Re: Automatic login to a web page

Then you can use WWW::Mechanize to simulate it. There are tons of examples on the net, and WWW::Mech has a lot of examples, look in the man3 subdir of the module distribution.

A simple example

```
#!/usr/bin/perl
use warnings;
use strict;
use WWW::Mechanize;

my $username = 'user';
my $password = 'pass';
my $url = 'http://www.xyz.com/Login.aspx';
my $mech = WWW::Mechanize->new(autocheck => 1);

$mech->get( $url );
$mech->form_name("login form name");
$mech->field("username", $username);
$mech->field("password", $password);
$mech->click("login button/submit name");

$mech->content() =~ /sm.th. from Track.aspx/ and print "logged in\n"
__END__
```

As an alternative to WWW::Mechanize, you can also use a combination of LWP::UserAgent and HTTP::Request::Common

```
#!/usr/bin/perl
use warnings;
use strict;
use HTTP::Request::Common qw(GET POST);
use LWP::UserAgent;

my $url = "http://my_isp/login.cgi";

my $ua = new LWP::UserAgent(timeout=> 5);

#now actually post the form
my $request = POST $url, [
  username => 'foobar',
  password => 'foobaz',
];

my $response = $ua->request($request);
my $content = $response->as_string();

print "$content\n";
__END__
```

Re: Automatic login to a web page

#####

But like I said, you need to be able to watch your transaction with
tcpick or ethereal, to see exactly what you need to do. There are
many complications, like it may set a cookie.

==
I'm not really a human, but I play one on earth.
<http://zentara.net/japh.html>

Or you can try Curl.

```
sub do_curl {

my $curl = Curl::easy::init();

if(!$curl) {
die "curl init failed!\n";
}

my $url = "enter your url here";
my $rawHTML = ""; # stores the return HTML

$::errbuf = "";
Curl::easy::setopt($curl, CURLOPT_ERRORBUFFER, "::$errbuf");

Curl::easy::setopt($curl, CURLOPT_URL, $url);
Curl::easy::setopt($curl, CURLOPT_NOPROGRESS, 1);
Curl::easy::setopt($curl, CURLOPT_TIMEOUT, 30);
Curl::easy::setopt($curl, CURLOPT_HEADERFUNCTION, \&header_callb);
Curl::easy::setopt($curl, CURLOPT_WRITEFUNCTION, \&body_callb);
Curl::easy::setopt($curl, CURLOPT_POST, 1);
Curl::easy::setopt($curl,
CURLOPT_POSTFIELDS,"username=xyz&password=xyz");
Curl::easy::setopt($curl, CURLOPT_SSL_VERIFYPEER, 0);
Curl::easy::setopt($curl, CURLOPT_SSL_VERIFYHOST, 1);
Curl::easy::setopt($curl, CURLOPT_USERAGENT, "Mozilla/4.0");
Curl::easy::perform($curl);
Curl::easy::cleanup($curl);

}

# Used with cURL; stores the raw HTML retrieved

sub body_callb {
my($chunk,$handle)=@_;
${handle} .= $chunk;
$rawHTML .= $chunk;
```

Re: Automatic login to a web page

```
return length($chunk);
```

```
}
```

```
# Used with cURL; gets header for debugging purposes.
```

```
sub header_callb {  
return length($_[0]);
```

```
}
```

If it sets a cookie, Curl can handle that too, but it gets more complicated. It's hard for us to give you an accurate script without having a valid logon and a way to test.

.