

DDNS Through Webhost

The purpose of this project was to provide a way to update customized sub-domain ip addresses that are dynamically assigned and subject to change. The update process can be achieved through a router, DD-WRT is the only router firmware I know of that provides the needed configuration options in the HTTP-GUI interface, or an update program running on a PC.

My research lead me to discover that dynamic DNS services (like dnsdynamic.org, changeip.com, [No IP](http://noip.com), afraid.org, dyndns.com) use HTTP as the update protocol. Information is sent via HTTP containing the user, password, domain and IP address. Authentication is provided by HTTP basic authentication. This could be a vulnerability and would preferably be done over an SSL connection.

Most web hosts provide configuration options through a web portal (CPL), including the ability to add and modify custom DNS entries. To complete this project I needed to figure out a way to script these changes. Curl was the first and most obvious choice. I used Firebug to monitor the post and response information transactions when logging into the web control panel and editing the DNS entries. From this I found out what information was being sent where. I used curl to successfully log into the control panel and save the session in a cookie. Then I used the cookie to post the proper information to the page that edits the DNS entries. BAM, I had my way to script the DNS changes.

CURL login with cookie info:

```
curl -c cookies.txt -d 'post information' 'destination address'
```

CURL form completion using cookie from login:

```
curl -b cookies.txt -d 'post information' 'destination address'
```

My script deletes the cookies after use.

The next step was to be able to listen for update requests, what better way than to use the web server running on the host. The test I did was on a site not running SSL, but it is only a test site and actually serving or dealing with sensitive data. In order to get a client to successfully update I had to implement the authentication mechanism. The standard file come into play, `.htaccess` and `.htpasswd`. I created a couple of test users and sub-domains and tested the functionality of authentication on the directory I chose for the test. Success!!!

So I had a way to script the changes and a server which provided authentication but still needed to link the server authentication to the update process. PHP was available on the host as was `php-curl`. After researching the DDNS client used by DD-WRT (`inadyn`) I had a basic understanding of the information exchanging between the client and the server and was ready to write a PHP script to listen for update requests in the password protected directory.

The following php script is the result. The `curl.php` script contains the update function. The update function contains sensitive information about my web host so it is not included. The update request sends the variables for IP and host name in the URL and they can be retrieved via the GET method.

```

<?php
include('path/to/curl.php');
$hostname=$_GET['hostname'];
$myip=$_GET['myip'];
$user=$_SERVER['PHP_AUTH_USER'];
$pwd=$_SERVER['PHP_AUTH_PW'];
$newline="";
$change="";
$file_handle=fopen('path/to/info.txt','rb');
while(!feof($file_handle)) {
    $line_of_text=fgets($file_handle);
    $parts=explode(' ', $line_of_text);
    if($parts[0]==$user) {
        if($parts[1]==$hostname) {
            if(trim($parts[2])==trim($myip)) {
                echo 'nochg ' . $myip;
            } else {
                update($hostname, $myip);
                echo 'good ' . $myip;
                $newline.=$parts[0] . " " . $parts[1] . " " . $myip . "\n";
                $change='yes';
            }
        } else { echo 'notfqdn'; }
    } else
        $newline.=$parts[0] . " " . $parts[1] . " " . $parts[2] . "\n";
}
fclose($file_handle);
if ($change=='yes') {
    $file_handle=fopen('path/to/info.txt','w');
    $newline=trim($newline);
    fwrite($file_handle, $newline);
    fclose($file_handle);
}
?>

```

A text file is used to check the submitted IP address against the current IP address stored in the web hosts DNS server. The text file stores the username, domain name and IP address for each subdomain. If there is no change in the IP address the script returns 'nochg'. If the domain name sent does not match with the domain name on a line of the text file it returns 'notfqdn'. If the IP has changed the script executes the update function, sends a 'good' response back to the client and writes the new address to the text file. The update function uses curl to update the IP address through the web hosts control panel.

The format for the text file looks like this

```

username fqdn current-ip-address
username fqdn current-ip-address
username fqdn current-ip-address

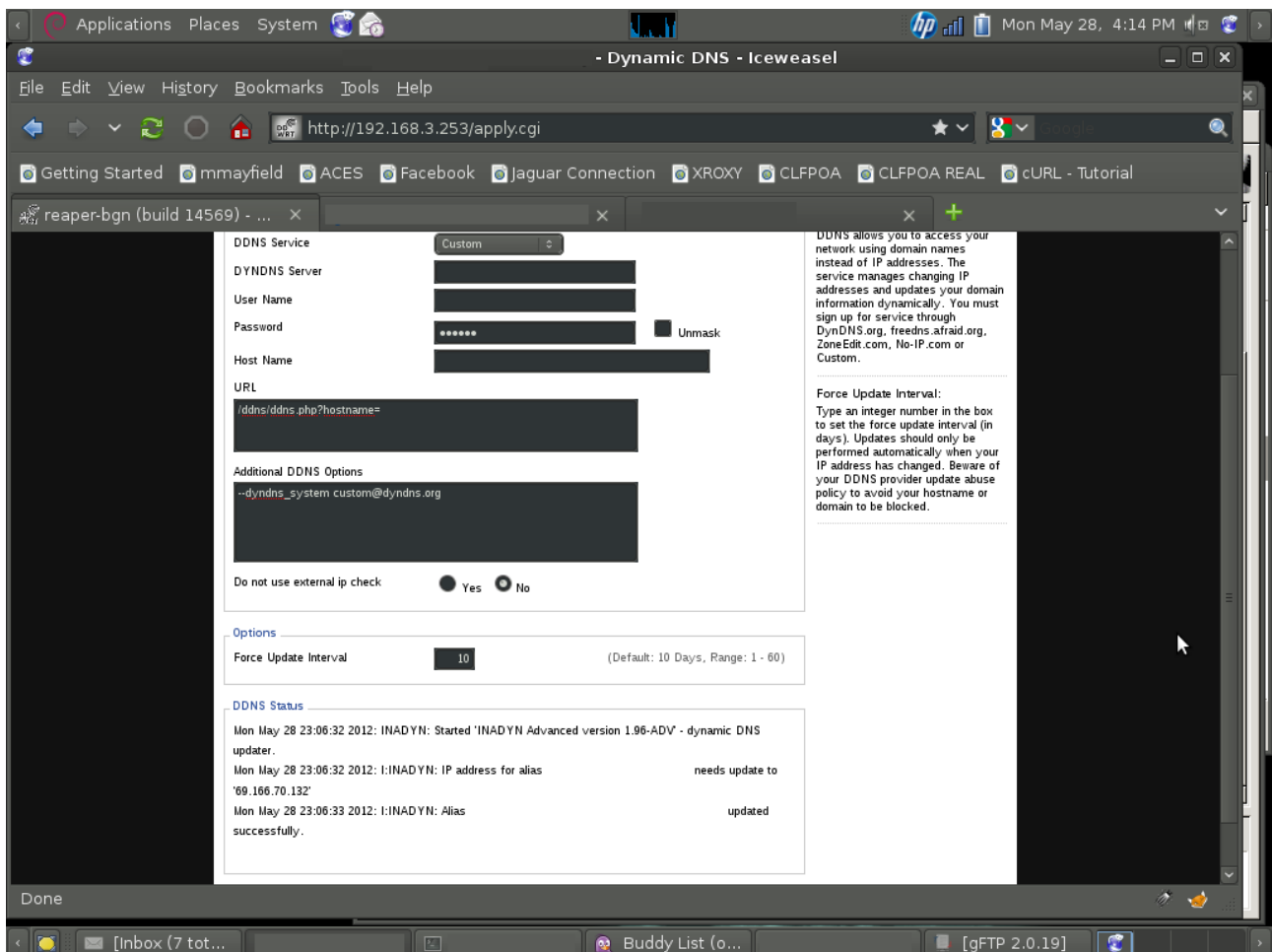
```

The curl-php portion logs in and obtains a cookie from the web host and then is basically just a conditional execution depending on the host name of the sub-domain.

The INADYN client needs three extra options to function with this setup:

```
--dyndns_system custom@dyndns.org
--dyndns_server_url /path/to/scripts?hostname=
--dyndns_server_name mywebserver.com
```

Below is a screen shot of an actual successful test using the DDNS settings in DD-WRT. It has been photo shopped to remove sensitive info. You should see how the DD-WRT GUI reflects the three required options listed above. It took me a while to get the options correct. The DD-WRT GUI appends the host name you enter to the value you put in the URL field. If you do not add the ?hostname= to the end of the URL field it does not work. To test the update I entered the current IP address in the text file and got an update not necessary response. I entered a bogus address into the text file and got an update successful response. Of course incorrect parameters like user, password, Host Name, URL or DYNDNS Server return errors.



References:

DD-WRT: <http://www.dd-wrt.com/>

FIREBUG: <http://getfirebug.com/>

INADYN: <http://www.inatech.eu/inadyn/>

CURL: <http://curl.haxx.se/>

PHP-CURL: <http://php.net/manual/en/book.curl.php>