RouterOS API PHP Class

PRACTICAL APPLICATIONS

About Me

- Nick Perkins, MTCNA, MTCWE
- Systems Engineer, Technology Solutions, Wairarapa
- Reasonably new to Mikrotik, First RoS v5.16
- Have been coding PHP since I was 15

This Presentation:

- RouterOS API PHP Class – What is it
- The Basic Framework for use
- Examples of Production Use
- Security suggestions

The RouterOS API PHP Class

- http://wiki.mikrotik.com/wiki/API_PHP_class
- Author: **Denis Basta**
- Contributors:

  Nick Barnes
  Ben Menking
  Jeremy Jefferson
  Cristian Deluxe

- Provides a library to use PHP to interact with the RouterOS API

The Connection

The connection phase of the API is very simple and uses the API service on the Mikrotik device on port 8728:

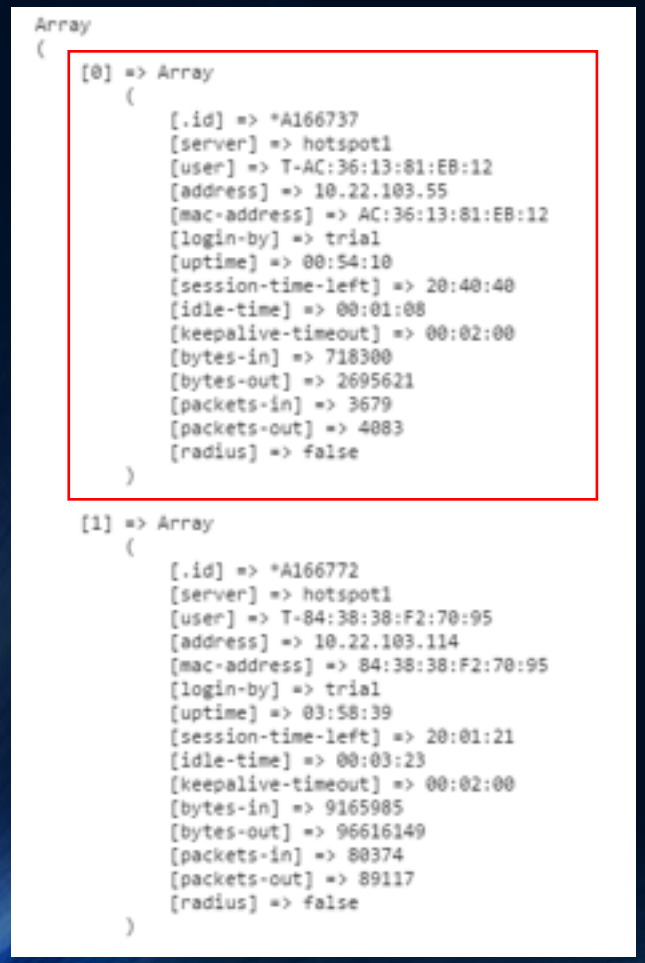
```php
<?php
require('routeros_api.class.php');                          /* <-- This is available at http://wiki.mikrotik.com/wiki/API_PHP_class */
$API = new routeros_api();                                  /* <-- Initialise the PHP API */
    if ($API->connect($routerip, $routeruser, $routerpass)) {   /* <-- Connect to our device */

    /*  Commands we want to perform, Example below: */
    $ARRAY2 = $API->comm("/ip/hotspot/active/print", false);
    $x = 0;
    foreach($ARRAY2 as $key => $value){
        $user = substr($ARRAY2[$x]['user'], 0, 2);
        if($user != "T-"){
        echo '<tr>';
        echo '<td>'.$ARRAY2[$x]['user'].'</td><td>'.((($ARRAY2[$x]['bytes-in']+$ARRAY2[$x]['bytes-out'])/1024)/1024).
        ' / '.(($ARRAY2[$x]['limit-bytes-total']/1024)/1024).'</td><td>'.$ARRAY2[$x]['session-time-left'].'</td>';
        echo '</tr>';}
        $x = $x + 1;
    }}
    else{
    echo 'Unable To Connect';                               /* <-- Graceful Failure if Connection Fails */
    }
$API->disconnect();                                         /* <-- Disconnect the session */
?>
```

Example Results

Raw Data:

From the command

$ARRAY2 = $API->comm("/ip/hotspot/active/print", false);



```
Array
(
    [0] => Array
        (
            [.id] => *A166737
            [server] => hotspot1
            [user] => T-AC:36:13:81:EB:12
            [address] => 10.22.103.55
            [mac-address] => AC:36:13:81:EB:12
            [login-by] => trial
            [uptime] => 00:54:10
            [session-time-left] => 20:40:40
            [idle-time] => 00:01:08
            [keepalive-timeout] => 00:02:00
            [bytes-in] => 718300
            [bytes-out] => 2695621
            [packets-in] => 3679
            [packets-out] => 4083
            [radius] => false
        )

    [1] => Array
        (
            [.id] => *A166772
            [server] => hotspot1
            [user] => T-84:38:38:F2:70:95
            [address] => 10.22.103.114
            [mac-address] => 84:38:38:F2:70:95
            [login-by] => trial
            [uptime] => 03:58:39
            [session-time-left] => 20:01:21
            [idle-time] => 00:03:23
            [keepalive-timeout] => 00:02:00
            [bytes-in] => 9165985
            [bytes-out] => 96616149
            [packets-in] => 80374
            [packets-out] => 89117
            [radius] => false
        )
```

Formatted in Human:

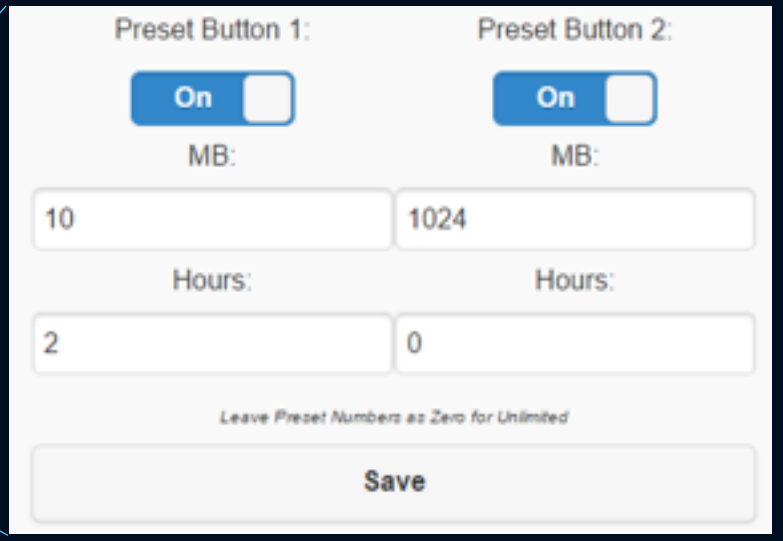
Active Trial Users

User Name	Used MB	Time Remaining
T-50:EA:D6:BF:EE:3A	12.0825767517	23h1m18s
T-C4:05:28:F3:6E:35	7.00699901581	22h59m
T-84:38:38:F2:70:95	45.1351299286	20h9m31s
T-14:A3:64:CB:9F:1F	12.0200033188	10h19m4s
T-5C:0A:5B:3E:4C:EA	8.31134128571	22h34m35s

The above info is extracted from the mikrotik to display the users that are currently actively using the properties hotspot.

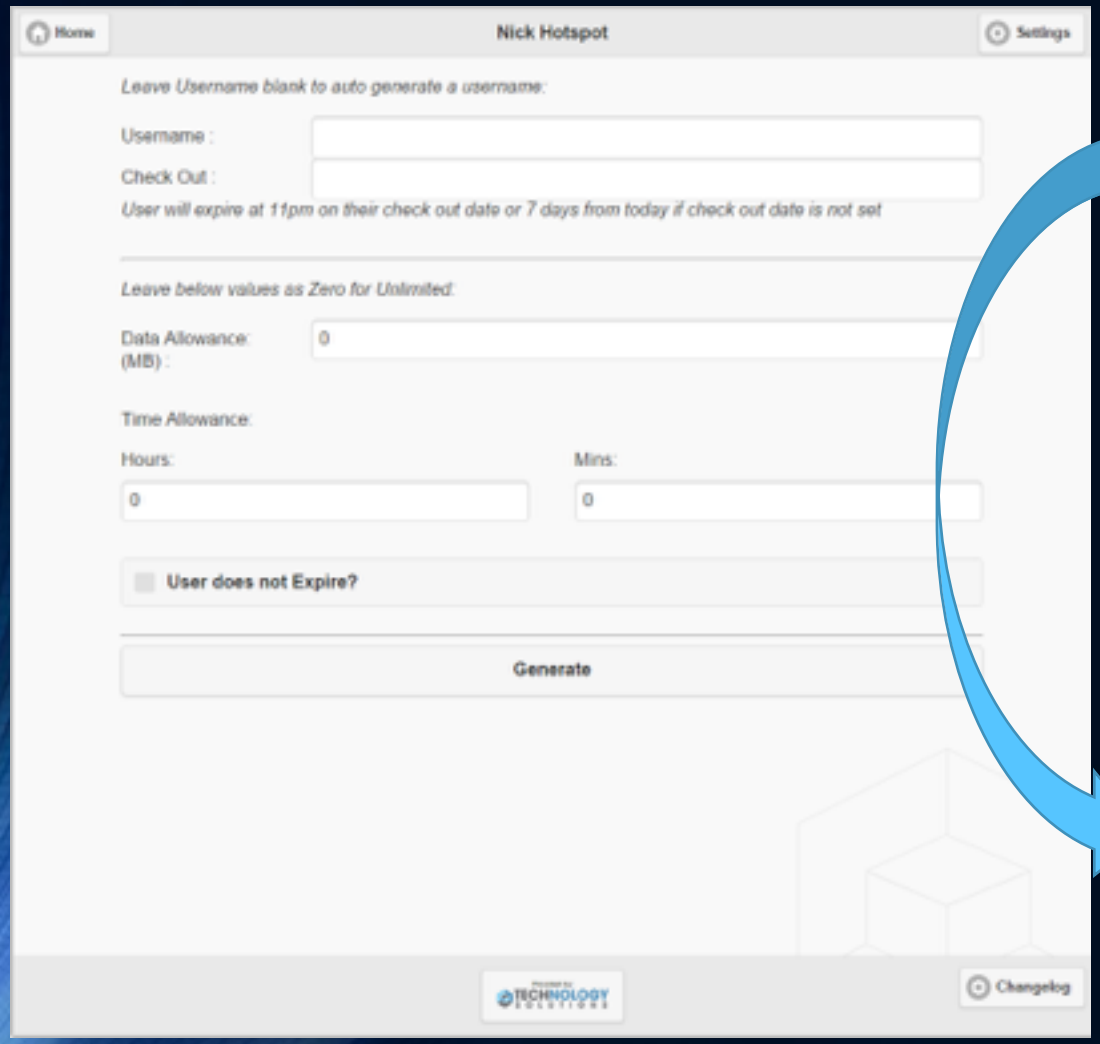
The API can run most commands that you can run in a terminal session so is very powerful and also very dangerous, We'll talk about security later

Production Example



The most useful application I have had for the PHP API thus far is to allow hotel/motel clients to create users for their user hotspot on their premises. We can use the API to generate users, remove users, view active users. Anything that a terminal command can do presented in a very user friendly, idiot-proof format.

Generate a Hotspot User:

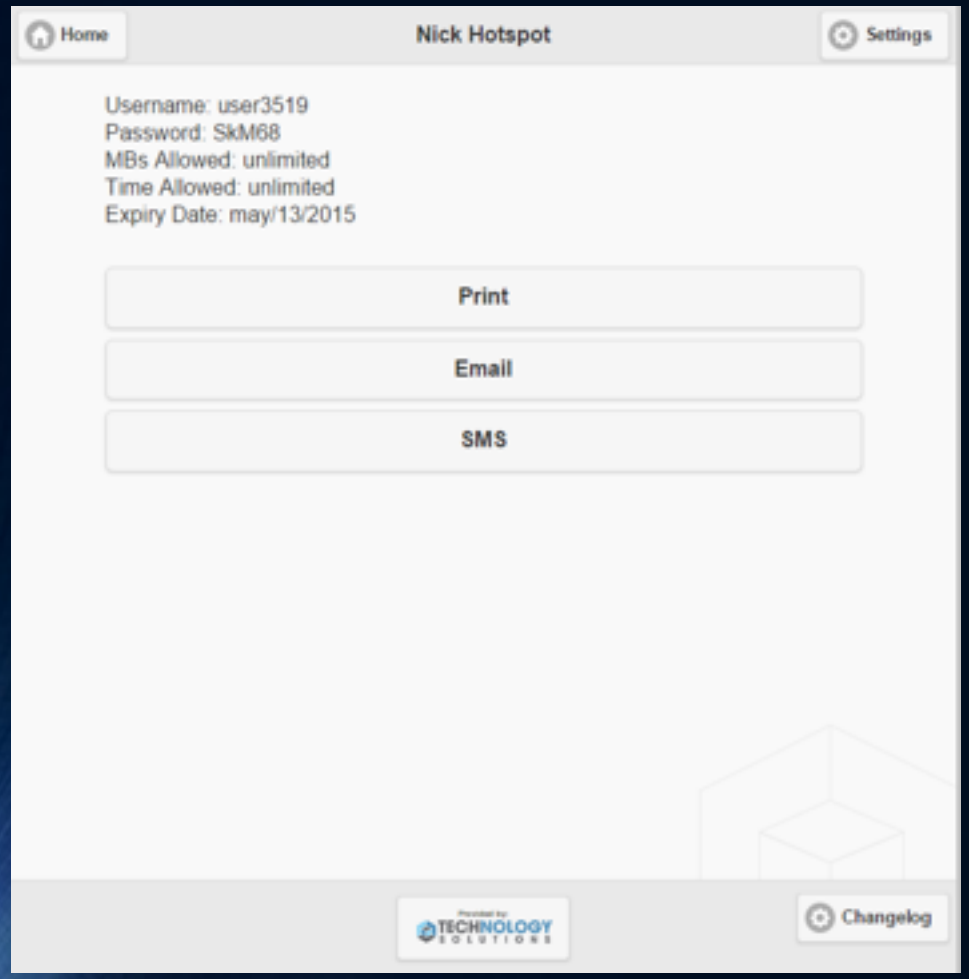



```
if ($API->connect($routerip, $routeruser, $routerpass)) {
    $ARRAY2 = $API->comm("/ip/hotspot/user/add", array(
            "name"      => $user,
            "password" => $password,
            "limit-uptime"  => $time,
            "limit-bytes-total"  => $mb,
            "comment" => $comment,
    ));
```

The above is the final command we run once we have the variables from the form on the left and have adjusted them as needed. Most Mikrotik users should be able to see the basic terminal command structure in this script

/ip hotspot users add name="<name>" password="<password>" limit-uptime="<uptime>" limit-bytes-total="<bytes>" comment="<comment>"

In this case a comment is added with a checkout date so we can remove them on the day and keep the system tidy



The resulting form for our clients is simple concise and easy to pass onto customers.

Another application that could be useful for a web interface is firewall interaction, once a set of standard firewall rules are laid out to block/unblock certain things a web interface for users to do so easily could be a valuable tool.

Example:

```
/ip firewall layer7-protocol add name=facebook regexp="^.+(facebook.com).*$"
```

We can then do what's needed with the layer 7 protocol, Mark connections, mark packets and then drop marked packets

Security

- API Must be enabled on external facing interfaces to allow access
- Recommend only allowing it from your web servers IP Address

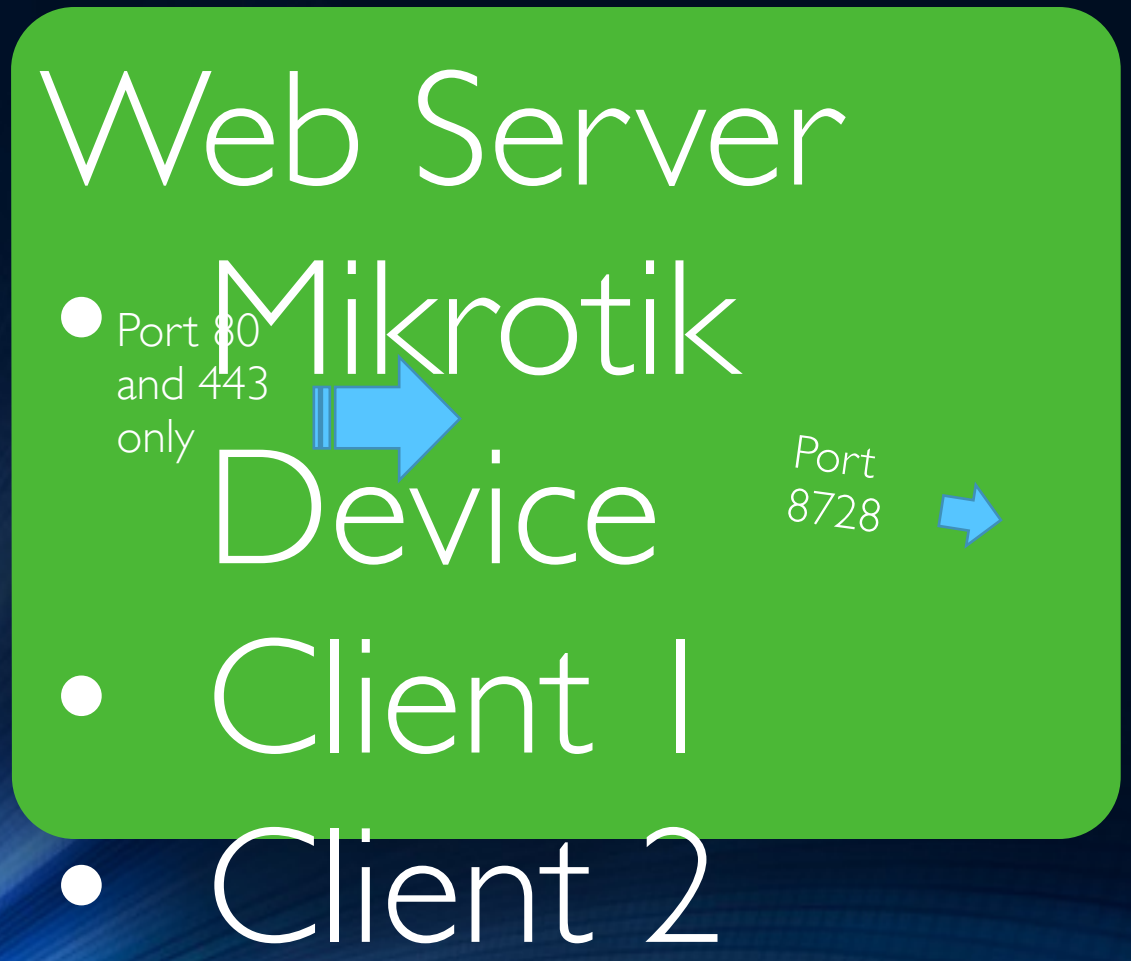
  /ip services set address=<Web Server IP> api

  However, I prefer to drop the packets at the firewall for better visibility:

  /ip firewall filter add protocol=tcp chain=input dst-port=8728,8729 action=drop src-address=!<Web Server IP>
  in-interface=<WAN Port>

- Store all user details in variables, Safest bet is to encrypt them, with a salt, in an SQL database and just grab as needed
- Provide SSL Security for all web applications that will be interacting with your Mikrotik
- Recommend creating a user group with the API, Read, Write privileges only, and create a separate user for your commands

Connection Advantage



One of the big advantages to working via the API is that users do not require special ports open to access the router, All their traffic is done on port 80 and 443, the only port opening required is 8728 coming back into the mikrotik device. They also don't need to be on the same network, they can be managing their device from anywhere.

Questions?

- Any questions, please feel free to ask?

- If anyone wants some help in the future with API commands please feel free to get in touch at nick@techs.co.nz