Domain Module for WHMCS

Installation, Configuration & Management

Module Version 2.0.1

Released April 2017

Table of Contents

About

Openprovider For WHMCS is a registrar module which allows you to offer Openprovider domains to your customers. Key features:

- Allows WHMCS to automatically perform domain and contact management tasks on your Openprovider domains, including:
    - Domain registration, renewal, and deletion
    - Domain updates (contact changes, nameserver changes, toggle client transfer prohibited status)
    - Contact creation
    - Domain transfer requests
    - Authcode retrieval
- Allow you to use the Openprovider API to check domain availability rather then using a standard WHOIS query, offering better performance for your customers.
- Allow you to use DNS templates from your Openprovider account when registering domains through WHMCS.
- Synchronize the expiration dates and statuses of domain records in your WHMCS database and your Openprovider portfolio.
- Auto renewal settings are now synchronized between Openprovider and WHMCS. If a domain is set to "Autorenew off" on WHMCS, then autorenew will also be turned off in Openprovider. If a domain in WHMCS has the setting "Auto renew on" then in your Openprovider account it will be set to your default global autorenew setting.

Installation

1. Download **Openprovider for WHMCS**
2. Upload **includes/hooks/openprovider.js** to your server at **<your WHMCS root Directory>/includes/hooks**
3. Upload the folder **modules/registrars/openprovider** to your server at **<your WHMCS root Directory>/modules/registrars**

API Configuration

1. From the admin area of your WHMCS installation, open the registrars configuration page at **<your WHMCS domain>/admin/configregistrars.php** or through the menu **Setup>Products/Services>Domain Registrars**
2. Find Openprovider in the list of registrars and click **Activate**

3. Fill in your Openprovider API credentials and save changes.

4. If the connection to your account is successful, a dropdown box will appear with a list of the DNS templates in your Openprovider account

5. Choose your preferred DNS template preference and press save again.

Cron Configuration

In order for the Openprovider plugin for WHMCS to function properly, a cron job needs to be scheduled for the script **<your WHMCS root Directory>/modules/registrars/openprovider/cron/DomainSync.php**

Frequency: X times every 6 hours, where X = number of Domains / 200

This cron script synchronizes domain statuses, expiration dates, and autorenew settings between Openprovider and WHMCS. Domain statuses and expiration dates are led by domain settings on Openprovider, and auto renew settings are led by WHMCS. By default, this script updates 200 domains per execution, and should be scheduled so that all domains are synchronized within 6 hours. For example, if you have 7000 domains in WHMCS with Openprovider, the cron job needs to be run 7000 / 200 = **35** times every six hours, i.e. at least once every 10 minutes.

Note that WHMCS has a different script, cron.php, which it uses to perform many of it's main functions. For your installation to work properly, both of these scripts need to be configured to run regularly.

To change default time for sync of all domains:

The variable "private static $interval_hours" in the file **modules/registrars/openprovider/OpenProvider/WhmcsHelpers/DomainSync.php** can be edited and set to any integer number of hours, in order to define the period of time in which all domains should be updated.

To change default number of domains run each time script is run:

The default for value for the variable **$limit** in the construct function "public function __construct($limit = 200)" in the file **modules/registrars/openprovider/OpenProvider/DomainSync.php** can be edited and changed to an integer value.

Note that if the number of domains being updated each time the script runs is increased, or the sync time is decreased, steps must be taken to ensure that each instance finishes before the next one begins.

TLD Management and Check domain availability

Navigate to the "config domains" page, located at **<your WHMCS domain>/admin/configdomains.php** or through the menu at **Setup>Products/Services>Domain Pricing** where you will see (1) a place to select the lookup provider details, (2) a place to choose a TLD, and (3) a dropdown to choose which domain provider to choose as the default registrar for the domain.

Choose Lookup Provider

1. Under lookup provider (1) choose "change" and from the popup menu select "Openprovider"
2. The lookup provider should be displayed as such:

3. All domain availability checks by your customers will now be performed using the Openprovider API.

Select a TLD and Auto Registration

1. Insert the desired TLD (2) and from the Auto Registration dropdown (3) select Openprovider
2. Click Save, and an option "open pricing" will appear

3. Click "open pricing" and define prices for register, transfer, and renewal in the popup box which appears
4. That's it! your WHMCS installation is ready to offer domains for registration for your clients

Admin Area

Accepting Orders

Once a domain has been ordered by a client, the order appears as pending in the admin area **<your WHMCS domain>/admin/orders.php** or through the menu at **Orders>Pending Orders**. WHMCS logic is that all orders must be approved by an administrator before being ordered at the registrar. Below the client has ordered and paid for the domain "register-a-new-domain.nl" but it hasn't yet been approved.

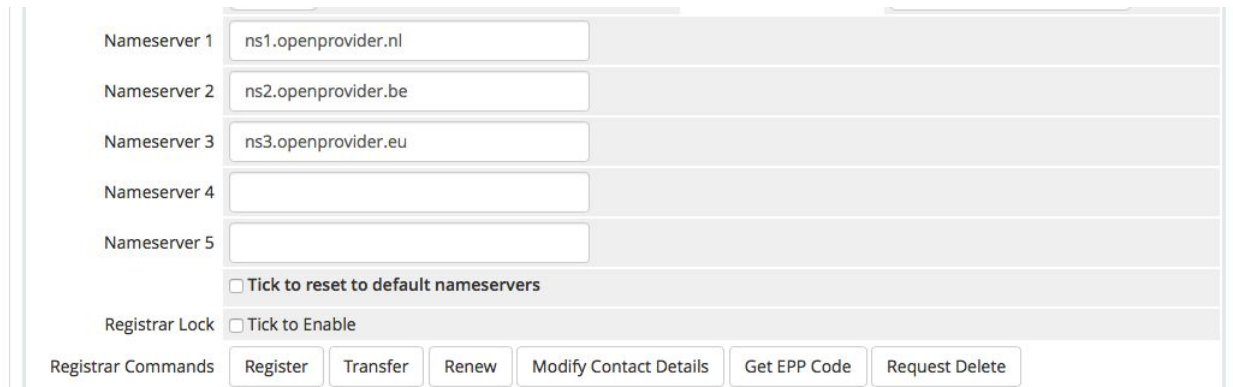Note from the previous section that Openprovider has been set as the registrar for auto registration for ".nl" domains. When you click accept order, the order is automatically placed at Openprovider. If there are other registrar options, there is a dropdown menu that can be chosen.

Managing Domains

Once a domain has been registered and activated, the domain can be managed from the admin area. You can navigate to **Clients>Domain Registrations** and select the desired domain. From here, there are options to send commands to Openprovider, including register, transfer, renew, change contact details, edit name servers, toggle registrar lock, and delete the domain.

Auto Renew Configurations

Recommended by WHMCS:
WHMCS recommends that auto-renew setting be turned off at the registrar, that way automated renew operations come only from WHMCS, reducing the chance of unwanted renewals being performed. Openprovider also suggests that admins use this configuration.

Alternative approach: Openprovider auto-renew "on"

Some admins want to give their clients extra time to pay for renewals by setting global auto-renew in Openprovider to "on". That way, domains will be renewed on their expiration date, and clients can pay late without losing their domains. This module synchronizes auto-renew settings from clients' domains into the Openprovider. So if a client marks a domain to not auto-renew in WHMCS, this will be reflected in Openprovider, overriding the global default, and ensuring you don't pay for a domain that a customer explicitly didn't want to renew.

Troubleshooting

If there are any issues with connection with Openprovider, or for some reason API commands are not working, the first troubleshooting step should be to look at the API logs.

Navigate to **Utilities>Logs>Module Logs** or **<your WHMCS domain>/admin/systemmodulelog.php** and you can find the raw API commands being sent and received by your WHMCS modules. The responses should contain some information about how the problem can be solved.

System Module Debug Log

🔆 Help

This module debugging tool can be used to identify and resolve problems communicating with remote API systems. It will record and display the raw API data being sent to, and received back from the remote system. Logging should only ever be enabled for testing, and never left enabled all the time.

Disable Debug Logging   Reset Module Debug Log

1880 Records Found, Page 1 of 38   Jump to Page: 1 ▼ Go

Date	Module	Action	Request	Response
21/03/2017 11:58	openprovider nl	checkdomainrequest	Array ([postValues] => <?xml version="1.0" encoding="UTF-8"?> <openXML><credentials>	Array ([curlResponse] => <?xml version="1.0" encoding="UTF-8"?> <openXML><reply><code>0</code>

Please contact our support staff with any questions
Support@Openprovider.nl