UpdateDD Version 2.4
Documentation

1 Table of Contents

1 Description

UpdateDD is a program for Unix like operating systems to update a dynamic DNS record. It supports multiple services, which can be listed by typing `updatedd -L'. Each service is represented by a plugin. Therefore it is very simple to add support for additional services.

Furthermore the package includes a wrapper for updatedd called `UpdateDD-Wrapper'. It provides support for configuration files and IP-Address-Caching.

Configuration-File

     ↓

UpdateDD-Wrapper   (execute) →   UpdateDD   ←→   Plugins

     ↕

Cache-File
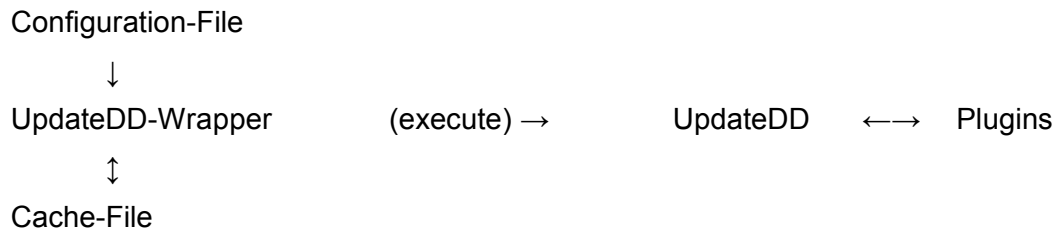
2 Usage

There are two kinds of options which are seperated by two dashes. Upper case options belongs to UpdateDD itself and lower case ones to the selected plugin. You can get more information about options and argumens by typing the following commands:

updatedd --help   about updatedd

updatedd <service> -- --help   about the selected service plugin


```
cd81:~$ updatedd --help

Usage: updatedd [OPTION]... SERVICE -- ...

Options:
  -L          list installed plugins (services) and exit
  -Y          use syslog
  --help      print help and exit
  --version   print version information and exit


cd81:~$ updatedd dyndns -- --help

Usage: updatedd [...] dyndns -- [OPTION]... [USERNAME:PASSWORD] HOSTNAME

For security reasons use the environment variable LOGIN instead of
passing the login information directly.

Options:
  -4    --ipv4 <address>     ip address version 4
  -b    --with-backmx        enable backmx
  -m    --mx <host>          hostname of your mail exchange
  -o    --offline            set host to offline mode
  -s    --system <system>    specify the system
                             (dyndns [default], statdns, custom)
  -w    --wildcard <on|off>  switch wildcard on or off
        --help               print help and exit
        --version            display version information and exit
```

Every plugin offers the option `-4' resp. `--ipv4' which can be used to specify an IP address to update the dynamic dns record with. Without this option the service tries to get your IP address automatically. If the service supports IPv6 the specific plugin also offers the option `-6' resp. `--ipv6'. All other options depends on the service and may vary.

The following examples can be used to update an dynamic dns record:

```
updatedd <service> USER:PASS HOSTNAME
updatedd <service> -- -4 IPADDR USER:PASS HOSTNAME

updatedd -Y <service> USER:PASS HOSTNAME
updatedd -Y <service> -- -4 IPADDR USER:PASS HOSTNAME
```

The option `-Y' causes UpdateDD to print all status information to the system log file which is normally `/var/log/syslog'.

You should never pass login information like usernames and passwords as agruments to a program, because of security reasons. Apart from that it would be possible for other users to get your passwords with the program `ps', because update procedures often take some minutes due to slow connections. Therefore you can use the environment variable $LOGIN to commit your login information to UpdateDD.

```
export LOGIN=user:pass
updatedd <service> HOSTNAME
updatedd <service> -- -4 IPADDR HOSTNAME
```

3 UpdateDD-Wrapper

UpdateDD-Wrapper is used to execute UpdateDD. It provides support for configuration files and IP address caching.

UpdateDD-Wrapper options:

-c \<path>	Specify another configuration file. Defaults are:
	/etc/updatedd-wrapper.conf
	/usr/etc/updatedd-wrapper.conf
	/usr/local/etc/updatedd-wrapper.conf
-d	Show debug information.
-f	Force update of dynamic dns record. Cached IP addresses will be ignored.
-i \<sec>	Set another idle time between failed updates. Default is 60 seconds.
-r \<num>	Set the number of retries.
-s	With this option UpdateDD-Wrapper will not execute UpdateDD. It just prints the commands to stdout. This option may be used to debug UpdateDD-Wrapper and to check whether it parses the configuration file correctly.
-t	Secify another cache file for IP addresses. Default is:
	/tmp/updatedd-wrapper_\<uid>.cache
-y	Print all output to syslog. Settings in the configuration file will be ignored.

3.1 Configuration File

By default, the configuration file is located at `/etc', `/usr/etc' or `/usr/local/etc'. The location depends on the way you installed UpdateDD.

The syntax is similar to the programming language C, but with one exception: comments starts with a sharp sign („#"). The configuration file includes two kinds of blocks. The first one is for specifying username and password for a record each associated with an alias name.

Example:

```
login {

        dyndns_1        = "user:pass";
        regfish_1       = "TOKEN=token";
        regfish_2       = "STD=user:pass";
        tzo_1           = "user:pass";

};
```

As for this example the alias name called `dyndns_1' will be associated with the login `user:pass'. An alias can consist of letters, numbers and underlines. The way you have to specify the login information depends on the service plugin. See `updatedd <service> -- --help' for help. Notice that each line has to end with an semicolon.
The second kind of blocks is used to set the service, what hostnames to update and how they should be updated.

Example:

```
regfish(active) {

        login           = regfish_1;
        hostnames       = "host1.org host2.org";
        ip-addr         = ipv4:`/usr/local/share/updatedd/ipserv.pl regfish`;
        use-syslog      = yes;
        options         = "--ismx";
#       force;

}
```

As for this example the selected service plugin is called `regfish' which is used for accounts at regfish.com. After the service name you have to set the following block to `active' or `disabled'. Disabled means, that the block will be ignored by UpdateDD-Wrapper.

Each block has to include the following arguments:

   login   Set the alias name for the login.

   hostnames   Set the hostname(s) seperated with a space.

Options:

    ip-addr   Specify an command which prints an ipv4 or ipv6 address to stdout.
                            Normally this should be an script which detects your IP address. UpdateDD
                            comes with some example scripts located at /usr[/local]/share/updatedd.
                            Notice that this is optional but highly recommended, because without this
                            option UpdateDD-Wrapper won't be able to cache your IP addresses.

    use-syslog   Print output to syslog. Default is `yes'.

    options   Additional options for the plugin.

    force   Ignore cached IP addresses.

4 Scripting

UpdateDD returns different values so you can easyly build up loops for your own scripts.

Status Codes:

0 =>   successfully updated.

1 =>   warning: an connection error occured or you've got no more memory left.

        You can try again here.

2 =>   error: the update was not successful because of wrong username/password or the

        hostname does not exist.

        You should not try again here, because otherwise your account may be locked up.

3 =>   Wrong arguments or options passed to UpdateDD.

5 System integration

There are two ways you can get UpdateDD to update your Dyn DNS entries automatically. If your machine is connected to the internet directly, you can use pppd to do the job. A small script is needed. Save it as '/etc/ppp/ip-up.d/updatedd'.

```
#!/bin/sh
sleep 30s  # wait until the connection established
updatedd-wrapper -y [additional options]
```

If you use PPPD it is often not necessary to Cache the IP-Addresses, because your Address changes every time you connect to the internet. In this case you can go easy on system resources if you don't use the option `ip-addr' within the UpdateDD-Wrapper configuration file or if your execute UpdateDD directly:

```
#!/usr/bin/perl -w
#!/usr/local/bin/perl -w

use strict;

###################### Configuration ######################

my $service        = "dyndns";
my $login          = "test:test";
```

```
my $hostnames        = "test.dyndns.org test2.dyndns.org";
my $ip_addr          = $ARGV[0];
my $retries          = 5;

#################################################################

$ENV{LOGIN} = $login;
my $ret;
foreach my $hostname (split /[\s,]/, $hostnames) {
   my $n = $retries;
   while($n != 0) {
      $ret = system("updatedd", "-Y", $service, "--", "-4", $ip_addr, $hostname);
      if($ret != 1) {
         $n = 0;
      } elsif($ret == 1) {
         $n--;
      }
   }
}

exit($ret);
```

You can create your own scripts to invoke UpdateDD using the information in chapter 4. Additional examples can be found at '/usr/[local/]share/updatedd'.
If no ppp daemon is installed on your machine, you can use cron instead to invoke UpdateDD at regular intervals. Therefore add the following line to '/etc/crontab':

```
*/10 * * * * root updatedd-wrapper -y &
```

In this example the UpdateDD-Wrapper is invoked every 10 minutes by the superuser root. For security reasons you are advised to use another user instead of root.

6 Bugreports

If you find a bug in this software write a bugreport to <updatedd@philipp-benner.de>. Please note that mailing me that something does work erroneously only or not at all does not help to solve the problem. Additional information is needed to correct the program:

- The complete commandline UpadteDD resp. UpdateDD-Wrapper is to be invoked with to show up the error.

- UpdateDD should be configured using `./configure -enable-debug' before compiled. I need the full terminal output. Ensure that UpdateDD's output is not written to syslog.

- If the UpdateDD-Wrapper causes the error, the output of `updatedd-wrapper -ds' and it's configuration file is needed.

7 Download

You can download UpdateDD at:

http://updatedd.philipp-benner.de

http://savannah.nongnu.org/download/updatedd/

CVS Repository

export CVS_RSH="ssh"
cvs -z3 -d:ext:anoncvs@savannah.nongnu.org:/cvsroot/updatedd co updatedd