MAVIS Daemon

Marc Huber

COLLABORATORS

	TITLE : MAVIS Daemon	

ACTION	*NAME*	*DATE*	*SIGNATURE*
WRITTEN BY	Marc Huber	November 17, 2026	

REVISION HISTORY

NUMBER	DATE	DESCRIPTION	NAME

Contents

1 Introduction **1**

1.1 Download 1

2 Command line syntax **1**

3 Event mechanism selection **1**

4 Configuration file syntax **1**

4.1 Railroad Diagrams 4

5 Sample configuration **4**

6 Copyrights and Acknowledgements **5**

1 Introduction

In conjunction with the **remote** module, this daemon allows extending MAVIS (modular attribute-value interchange system) functionality over the network, using UDP packets for communication with clients and/or other MAVIS daemons.

1.1 Download

You can download the source code from the GitHub repository at https://github.com/tacplus-ng/event-driven-servers/. On-line documentation is available via https://tacplus-ng.github.io/event-driven-servers/doc/index.html, too.

2 Command line syntax

Command line syntax is:

```
mavisd [ -P ] [ -d level ] configuration-file [ id ]
```

The path to the configuration file is the only command line argument mandatory. *id* defaults to `tac_plus` and may be used to select a non-default section of the configuration file.

The *-P* option enables *config parse mode*. Keep this one in mind; it is imperative that the configuration file supplied is syntactically correct, as the daemon won't start if there are any parsing errors at start-up.

The `-d` switch enables debugging. You most likely don't want to use this. Read the source if you need to.

3 Event mechanism selection

Several level-triggered event mechanisms are supported. By default, the one best suited for your operating system will be used. However, you may set the environment variable `IO_POLL_MECHANISM` to select a specific one.

The following event mechanisms are supported (in order of preference):

- port (Sun Solaris 10 and higher only, `IO_POLL_MECHANISM=32`)
- kqueue (*BSD and Darwin only, `IO_POLL_MECHANISM=1`)
- /dev/poll (Sun Solaris only, `IO_POLL_MECHANISM=2`)
- epoll (Linux only, `IO_POLL_MECHANISM=4`)
- poll (`IO_POLL_MECHANISM=8`)
- select (`IO_POLL_MECHANISM=16`)

4 Configuration file syntax

A typical **mavisd** configuration file consists of a single `id` section:

```
section = mavisd { ... }
```

Default section ID is `mavisd`, but that a different ID may be selected via the command line.

Configuration directives are:

- (permit | deny) [not] CIDR

  Accept or reject requests from specific IP address ranges. This directive may appear multiple times. Matches are tried in order. IPv6 ACLs are supported. Default is to accept everything.

  Example:

```
permit 127.0.0.1/8
permit accept ::1
deny not 192.168.0.0/16
permit 192.168.0.5/32
```

- background = (yes | no)

  If set, the daemon will release its controlling terminal on startup and fork itself to the background (default: no).

- listen = { ... }

  The listen directive specifies where to listen for incoming queries. Acceptable connection endpoints are both IPv4/IPv6 based UDP and UNIX datagram sockets.

  For IP sockets, the following options are available:

  – port = UDPPort
    Specifies an UDP port to listen for incoming queries.
  – address = IPAddress
    Specifiy an IP address to bind to (optional, default: all local IP addresses).

  UNIX sockets support these directives:

  – path = UnixPath
    Path to an UNIX domain socket.
  – userid = UserID
    User ID for socket creation.
  – groupid = GroupID
    Group ID for socket creation
  – mode = Mode
    Permissions for socket creation.

  Communication via PF_UNIX sockets may only work if the host system supports anonymous binds for that protocol family. This works for Linux, which supports an abstract namespace which is independent of the file system, but, e.g., not for Sun Solaris.

  Options common to both variants are:

  – blowfish key = Key
    Specifies a key for over-the-wire encryption (optional).
  – blowfish keyfile = KeyFile
    Specifies a file to read the a key from (optional).

  The listen directive may be used multiple times and is mandatory.

- mavis module =*module* { ... }

  Load MAVIS module *module*. See the MAVIS documentation for configuration guidance.

- mavis path = *path*

  Add *path* to the search-path for MAVIS modules.

  Magic cookie substitution applies. The available conversions are:

  – %o - run-time OS type
  – %O - compile-time OS type

- `pidfile =` *file*

  The daemons process id will be written to *file* (default: unset).

- `stat period =` *seconds*

  This enables periodic statistics logging to *syslogd* (default: disabled). The logged line starts with `STAT:` and is followed by a series of *key*=*value* pairs:

  - `Q=`*count* - number of queries since start-up
  - `A=`*count* - number of queries answered since start-up
  - `R=`*count* - number of queries rejected since start-up
  - `X=`*count* - number of queries expired since start-up
  - `E=`*count* - number of queries failed since start-up
  - `T=`*seconds* - number of seconds since start-up
  - `q=`*count* - number of queries in last period
  - `a=`*count* - number of queries answered in last period
  - `r=`*count* - number of queries rejected in last period
  - `x=`*count* - number of queries expired in last period
  - `e=`*count* - number of queries failed in last period
  - `t=`*seconds* - length of last period
  - `B=`*count* - maximum number of queries in backlog since start-up
  - `b=`*count* - maximum number of queries in backlog in last period

  Modules may log additional information, e.g.:

  - `F=`*count* - number of childs forked since start-up
  - `I=`*count* - number of answers received since start-up
  - `O=`*count* - number of queries sent since start-up
  - `f=`*count* - number of childs forked in last period
  - `i=`*count* - number of answers received in last period
  - `o=`*count* - number of queries sent in last period

- `syslog (( ident =` Ident) | (`level =` Level) | (`facility =` Facility))

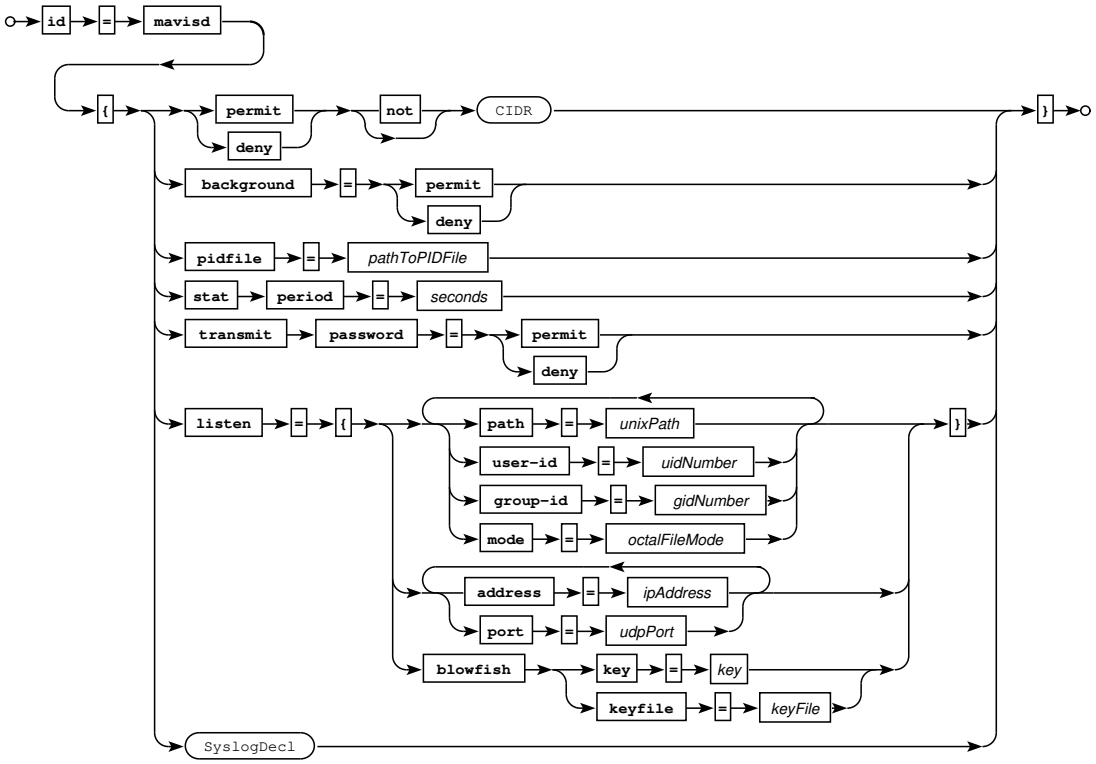
  Selects syslog *ident*, *level* and *facility*. Defaults to:

```
syslog ident = program-name
syslog facility = UUCP
syslog level = INFO
```

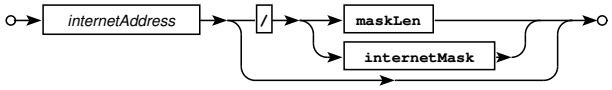
- `transmit password = ( yes | no )`

  Allow transmission of cleartext password in responses (default: `no`).

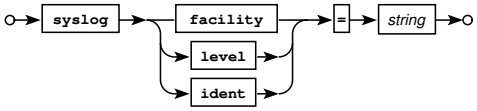
4.1 Railroad Diagrams



Railroad diagram: TcprelayConfig



Railroad diagram: CIDR



Railroad diagram: SyslogDecl

5 Sample configuration

```
module-path /some/where/lib/mavis

module-add log

module-add limit
module-conf limit blacklist count 5 time 60
module-conf limit ipreg time 60

module-add auth

module-add cache
module-conf cache expire * 60

module-add external id whatever
```

```
module-conf ext1 program /where/ever/script.pl

bind address 127.0.0.1 port 9001
```

6 Copyrights and Acknowledgements

Please see the source for copyright and licensing information of individual files.

- **The Blowfish algorithm:**

  This software uses Bruce Schneier's Blowfish algorithm.

- **Portions of the parsing code are taken from Cisco's tac_plus developers kit which is distributed under the following license:**

  Copyright (c) 1995-1998 by Cisco systems, Inc.

  Permission to use, copy, modify, and distribute this software for any purpose and without fee is hereby granted, provided that this copyright and permission notice appear on all copies of the software and supporting documentation, the name of Cisco Systems, Inc. not be used in advertising or publicity pertaining to distribution of the program without specific prior permission, and notice be given in supporting documentation that modification, copying and distribution is by permission of Cisco Systems, Inc.

  Cisco Systems, Inc. makes no representations about the suitability of this software for any purpose. THIS SOFTWARE IS PROVIDED ``AS IS'' AND WITHOUT ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, WITHOUT LIMITATION, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE.

- **The code written by Marc Huber is distributed under the following license:**

  Copyright (C) 1999-2022 Marc Huber (Marc.Huber@web.de). All rights reserved.

  Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

  1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
  2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
  3. The end-user documentation included with the redistribution, if any, must include the following acknowledgment:

     This product includes software developed by Marc Huber (Marc.Huber@web.de).

     Alternately, this acknowledgment may appear in the software itself, if and wherever such third-party acknowledgments normally appear.

  THIS SOFTWARE IS PROVIDED ``AS IS'' AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL ITS AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.