



# MALLOC Implementation

## MAAS Manual

**Håkan Byström, Maria Isberg, Pia Lindström,  
Fredrik Pettersson, Linus Svensson, Åke Östmark**

**MAAS Manual  
SMD113 – Networking Project**

Department of Computer Science and Electrical Engineering  
Division of Computer Communications  
2001-05-17

# Contents

<b>1. Introduction</b>	<b>1</b>
<b>2. Installation</b>	<b>1</b>
<b>3. Startup</b>	<b>1</b>
3.1 Normal mode.....	1
3.2 Verbose mode.....	1
3.3 MADCAP configuration.....	2
3.3.1 Handling of configuration file.....	2
3.3.2 Editing the configuration file.....	2
3.4 AAP configuration.....	4
3.4.1 Handling of configuration.....	4
3.4.2 Editing the configuration file.....	4
3.5 MADCAP stable storage.....	5
<b>4. Features not included</b>	<b>5</b>
4.1 Options.....	5
4.1.1 Requested Language.....	5
4.1.2 Multicast Scope List.....	5
4.1.3 List of Address Ranges.....	5
4.1.4 Feature List.....	5
4.1.5 Retry Time.....	5
4.2 Messages.....	6
4.2.1 DISCOVER.....	6
4.2.2 RENEW.....	6
<b>5. Bugs and characteristics</b>	<b>6</b>

## 1. Introduction

This manual describes how to install the MAAS v0.1 server properly and how to start up the server with different configurations and output features. The most convenient installation requires root access but the server can be installed and started in user environment. Note that this is not a complete implementation of all functionality possible for a Multicast Address Allocation Server. For further details on features not supported see section 4, and for details about known bugs and characteristics see section 5.

## 2. Installation

The MAAS is distributed as a file named `maas-<ver>.tar.gz`, where `<ver>` is the version of the library. The current version is 0.1. In order to extract the files from the archive, use the command

```
tar -xzf maas-0.1.tar.gz
```

You now have a directory called `maas-0.1`. The rest of the installation procedure is described in the file `INSTALLATION` in the newly created directory and in section 3.3 and 3.4 below, but if you run as root you should only need to do the following commands.

```
./configure
make
make install
```

The file installed is:

- `/usr/local/bin/maasd` -the MAAS main program.

For instructions on how to edit the configuration of the MAAS see section 3.3 and 3.4.

## 3. Startup

### 3.1 Normal mode

The synopsis of the server is

```
maasd [-v verbose_options] [-a aap_config_file] [-m madcap_config_file]
aap_config_name1 [aap_config_name2 ...] [madcap_config_name].
```

The verbose mode and possible flags are described in section 3.2. The configuration files are described in sections 3.3 and 3.4.

Normally the server is started with arguments selecting which MADCAP and AAP configurations are to be used. If no MADCAP configuration is chosen a default configuration is used, but this is not recommended.

Any number of AAP configurations can be started, but only one MADCAP configuration is allowed.

### 3.2 Verbose mode

The verbose mode enables the user to turn on selected messages and printouts from the server.

The pre-defined options are:

- '+' turn on all messages
- 'i' initiation
- 'g' handling of generic messages in the MADCAP part

'm'	madcap messages (getinfo, request, release, ack, nak, discover, renew, offer and unknown)
'f'	formatting outgoing options
'a'	allocations connected to the client
's'	all bytes sent and received by the server
'n'	networking information
'd'	aap messages
'e'	internal error messages in MADCAP

For example, the command

```
maasd -v in brutus
```

starts the MAAS with initiation and networking information printouts.

In the AAP part of the MAAS this feature is not fully implemented and some messages are still printed even when verbose mode is not selected. This should be changed in a future version.

This feature uses some unnecessary processing power even when verbose mode is not selected, and may therefore be removed when ready for released on a large scale internetwork if optimization is desired.

## 3.3 MADCAP configuration

### 3.3.1 Handling of configuration file

In the directory `maas-0.1/src` the file `madcap.conf` can be found. This file is to be considered as an example of a configuration file for the MADCAP part of the MAAS server. This file should be placed in the directory `/etc` or `/usr/local/etc` through the command

```
cp madcap.conf /etc
```

or

```
cp madcap.conf /usr/local/etc.
```

Edit the copied configuration file to reflect the local configuration of the MADCAP part of the MAAS in accordance to the AAP configuration made in the file `aap.conf` as explained in section 3.4. Note that these two configurations have to conform to each other if the MAAS is to work properly, and that no cross evaluation of the files is made in the MAAS.

The handling of the configuration of the MAAS is not done in an optimal way and should be improved in a future version. The current implementation states the following rules for the configuration files:

- The name of the configurations must be unique between MADCAP and AAP.
- For a set of desired AAP configurations there must be one MADCAP configuration stating the scope of all AAP configurations in the set. An example of this is shown in section 3.3.2.

The names of the configurations *can* be the same, but this only works for special cases and reduces flexibility in the choice of configurations at start-up. Following the rules above is therefore strongly recommended.

### 3.3.2 Editing the configuration file

The configuration file, `madcap.conf`, can contain any number of configurations but only one can and will be actually used. The one used is the one with the name that matches the name given as an argument at start-up. An example configuration with the name "adam\_brutus" is:

```
madcap "adam_brutus" {
```

```

/* Scope of AAP adam. */
scope {
    address_range {
        start = 224.0.0.0;
        end   = 224.255.255.255;
    };

    name_set {
        ("En", "Adam's Scope");
    };

    ttl = 16;
    min_free_addr = 10; /* unit percent */
    start_no_addr = 30;
};

/* Scope of AAP brutus. */
scope {
    address_range {
        start = 239.192.0.0;
        end   = 239.192.63.255;
    };

    name_set {
        ("En", "Brutus' Scope");
    };

    ttl = 16;
    min_free_addr = 10; /* unit percent */
    start_no_addr = 30;
};

/* MADCAP specific configuration. */
port = 2535;
CSA = 1800;
DELAY = 4 ;
ALLOC_TIME = 3600;
RESPONSE_DELAY = 60;
OFFER_HOLD = 70;
RCI = 60;
XRI = 600;
cachefilename = "/tmp/madcap.cache";

};

```

By starting `maasd` with `adam_brutus` as option you will get a configuration that states the two AAP scopes of `adam` and `brutus`, and some MADCAP specific configurations.

A configuration can contain up to 20 scopes (this is set in `madcap_cfg.c`). The two scopes of this configuration will have an address range from 224.0.0.0 to 224.255.255.255 and from 239.192.0.0 to 239.192.63.255. The name set contains a pair; a language tag and a name for each scope. A name set can contain several pairs of different language tags and names. The `ttl` is meant to be a default value and in these scopes the `ttl` are set to 16. The minimum amount of free addresses before new addresses are allocated is 10% of the total amount of already allocated addresses for both scopes. The number of addresses allocated at start in this configuration is 30 addresses for both scopes.

The reserved port number dedicated for MADCAP server is 2535. CSA (clock-skew-allowance), DELAY (discover-delay), ALLOC\_TIME, RESPONSE\_DELAY, OFFER\_HOLD, RCI and XRI are time constants and are set to recommended values in seconds. This configuration will use the file `/tmp/madcap.cache` as cache file for the address record.

Comments are indicated by the C `/*`-construct.

The order of the items in the configuration file is crucial. If the items are reordered or missing, `maasd` will promptly exit with an error message. This should be fixed in future versions.

## 3.4 AAP configuration

### 3.4.1 Handling of configuration

In the directory `maas-0.1/src` the file `aap.conf` can be found. This file is to be considered as an example of a configuration file for the AAP part of the MAAS server. This file should be placed in the directory `/etc` or `/usr/local/etc` through the command

```
cp aap.conf /etc
or
cp aap.conf /usr/local/etc.
```

Edit the copied configuration file to reflect the local configuration of the AAP's of the MAAS in accordance to the MADCAP configuration made in the file `madcap.conf` as explained in section 3.3. Note that these two configurations have to conform to each other if the MAAS is to work properly, and that no cross evaluation of the files is made in the MAAS.

The handling of the configuration of the MAAS is not done in an optimal way and should be improved in a future version. The current implementation states the following rules for the configuration files:

- The name of the configurations must be unique between MADCAP and AAP.
- For a set of desired AAP configurations there must be one MADCAP configuration stating the scope of all AAP configurations in the set. An example of this is shown in section 3.3.2.

The names of the configurations *can* be the same, but this only works for special cases and reduces flexibility in the choice of configurations at start-up. Following the rules above is therefore strongly recommended.

### 3.4.2 Editing the configuration file

The configuration file, `aapd.conf`, can contain any number of configurations. An example configuration with the name "brutus" is:

```
maas "brutus" {
    scope = (239.192.0.0, 239.192.63.255);
    // Organizational scope

    address = "239.192.0.1";
    port     = 20202;

    cache    = "/var/aapd/brutus.cache";

    STARTUP_WAIT = 150;
    ANNOUNCE_WAIT = 10;
    RESEND_WAIT = 1;
    REPEAT_INTERVAL = 30;
    ASA_INTERVAL = 30;
    ASRP_INTERVAL = 30;
};
```

By starting `maasd` with `brutus` as option, an AAP server serving the organizational scope 239.192.0.0/14 will be started. It will listen and send packets addressed to 239.192.0.1, port 20202, and will use the file `/var/aapd/brutus.cache` as cache file for the Allocation Record. The definitions following are the time-out constants of AAP in seconds.

The order of the items in the configuration file is crucial. If the items are reordered or missing, `maasd` will promptly exit with an error message. This should be fixed in future versions.

The configuration file has a syntax that has been deliberately chosen to be somewhat like that of the

C programming language or the Berkley named configuration file. Comments are indicated either by the C `"/*"`-construct, by the C++ `"//"`-construct, or by the sh(1) `"#"`-construct.

Each configuration can either be run as a MAAS instance of AAP, or as a Prefix Coordinator. The behavior is indicated by the keywords `maas` or `pc` before the configuration's name. This feature is however only used for testing a MAAS towards a PC. For a more appropriate PC there exist better implementations.

### 3.5 MADCAP stable storage

The MADCAP cache file can be unreadable for two reasons; the server went down during writing or a write error occurred while writing. For this reason a backup of the cache file, with the extension `.bak` to the filename, is made each time before the cache file is updated. If the `*.cache` file is reported to be not readable the backup file `*.cache.bak` should be copied and replace the erroneous `*.cache`.

## 4. Features not included

### 4.1 Options

This section describes options *not* fully supported in the MADCAP part of the MAAS.

#### 4.1.1 Requested Language

This option specifies the language in which the MADCAP client would like strings such as zone names to be returned. It is only included in an GETINFO message sent by the client and MAY be processed by a MADCAP server.

#### 4.1.2 Multicast Scope List

This option MAY be sent by the server in an ACK message in response to a GETINFO message sent by the client. It includes a list of the multicast scopes supported by the server.

#### 4.1.3 List of Address Ranges

This option is not supported by the server when processing a REQUEST or DISCOVER message from a client. It is however included in the ACK responds to those messages since it MUST be.

The option is used by the client when requesting a lease for a specific set of addresses. This feature should be needed only rarely, such as when a lease is accidentally allowed to expire and it needs to be reallocated.

#### 4.1.4 Feature List

This option lists optional MADCAP features supported, requested, or required by the sender. This option MAY be included in any message sent by a MADCAP server or client.

#### 4.1.5 Retry Time

The Retry Time option specifies the time at which a client should retry a REQUEST or RENEW message when using the Retry After feature.

This option should only be sent by a MADCAP server in an ACK when responding to a REQUEST or RENEW message that includes the Retry After feature in the Feature List. The option Feature List is not necessary in any type of message.

## 4.2 Messages

This section describes what part is missing in the implementation of the messages. The following messages is not usable in the way that could be expected.

### 4.2.1 DISCOVER

The message is recognized and handled by the server but the part handling confirmation or release of preallocated addresses is not implemented, i.e. the handling of the following REQUEST message is not fully implemented. Further, the addresses requested in the DISCOVER message is permanent allocated, but should be removed after [DISCOVER-DELAY] seconds (normally if the client does not respond with a REQUEST within that time). In short, the DISCOVER message does the same thing as a normal REQUEST message does.

### 4.2.2 RENEW

The message is recognized and handled by the server but the part handling the renewal is not implemented. Any RENEW message received by the server is responded with a NAK stating that the renewal cannot be granted.

## 5. Bugs and characteristics

The following list specifies known bugs in the implementation that effects the running of MAAS.

- The AAP implementation delivers multicast addresses without concern for reserved addresses or those relative within multicast address scopes.
- At few occasions, not exactly identified, the AAP starts to defend allocated addresses against itself. It has occurred for example when addresses such as \*.\*.\*.0 and \*.\*.\*.255 have been claimed by the AAP.
- The AAP has no backup of the cache file.

The following list describes conscious characteristics of the implementation that should be improved in future version of the MAAS.

- The order of the items in the configuration files is crucial. If the items are reordered or missing, `maasd` will promptly exit with an error message.
- The configuration files of AAP and MADCAP is handled separately, but must conform to each other for the MAAS to work properly.
- All configuration names must be unique.
- The messages DISCOVER and RENEW cannot be processed by the server to full extent.
- The options Requested Language, Multicast Scope List, Feature List and Retry Time is not supported by the server.
- The MAAS does not locate the backup of the MADCAP cache file in case of a read error.