# Sauron

## User Guide

**Timo Kokkonen**

tjko@iki.fi

**Sauron: User Guide**

by Timo Kokkonen

Published 2003-03-05
Copyright © 2003 by Timo Kokkonen

This manual describes Sauron, a free DNS & DHCP management system.

Revision History

Revision 0.6 2003-03-05 Revised by: tjko
Updated to cover new features in Sauron 0.6.x
Revision 0.5 2003-02-21 Revised by: tjko
First draft

# Table of Contents

# List of Tables

# List of Figures

# Preface

This manual is designed to address user's questions about installing and using Sauron, free DNS & DHCP management software.

Listed below are the topics covered in this User Guide.

- Chapter 1: introduction to Sauron, and pointers to additional information about Sauron.
- Chapter 2: explains how to install and configure Sauron.
- Chapter 3: provides an overview of how to start using Sauron.
- Chapter 4: explains how to manage user accounts.
- Chapter 5: reference for Sauron web-based user interface.
- Chapter 6: reference for Sauron command-line utilities.

# Chapter 1. Introduction

Sauron is a scalable system for management of DNS and DHCP services. Sauron can dynamically generate complete DNS and DHCP configurations from a central SQL-database. Automatic generation of DNS "reverse" zones (with support for smaller than C-class reverse delegations) is supported. Sauron comes with web interface and command-line interface. The web interface comes with user/group based access controls for distributed management.

## 1.1. History of Sauron

Sauron was originally designed and written by Timo Kokkonen at the Computing Center of University of Jyvaskyla. In 1999, when work on Sauron was started, it was meant to provide a simple web based interface for letting multiple users simultaneously manage DNS zone files, but it has since grown into a full featured DNS & DHCP management software.

Sauron is still under development, but it has been in production use since early 2001, managing the campus area network (CAN) with over 10,000 computers at University of Jyvaskyla, Finland.

In January 2003, Sauron was publicly released under the GNU General Public License (GPL).

## 1.2. System Overview

Sauron is built around sauron "back-end" that is used by the command-line and web interfaces to access the data stored in a SQL-database (see Figure 1-1). Sauron is able to dynamically generate complete DNS (ISC BIND) and DHCP (ISC DHCP) configurations based on the data stored in the SQL-database. Sauron is currently written almost completely in Perl using only standard perl modules and couple of additional modules from CPAN. Web interface is stand-alone CGI script and should work with almost any WWW server that supports CGI.

Database is organized as follows; a number of servers may be defined where each server may contain several zones. At server level there may be "global" definitions for DNS & DHCP settings and a Layer-2 network map (VLAN map), which is used generating necessary topology information for dhcpd. Each server can have multiple (DNS) zones defined. Hosts belong to a zone and may contain additional information (such as asset IDs) in addition to DNS/DHCP information.

**Figure 1-1. System Overview**

Sauron: General System Layout
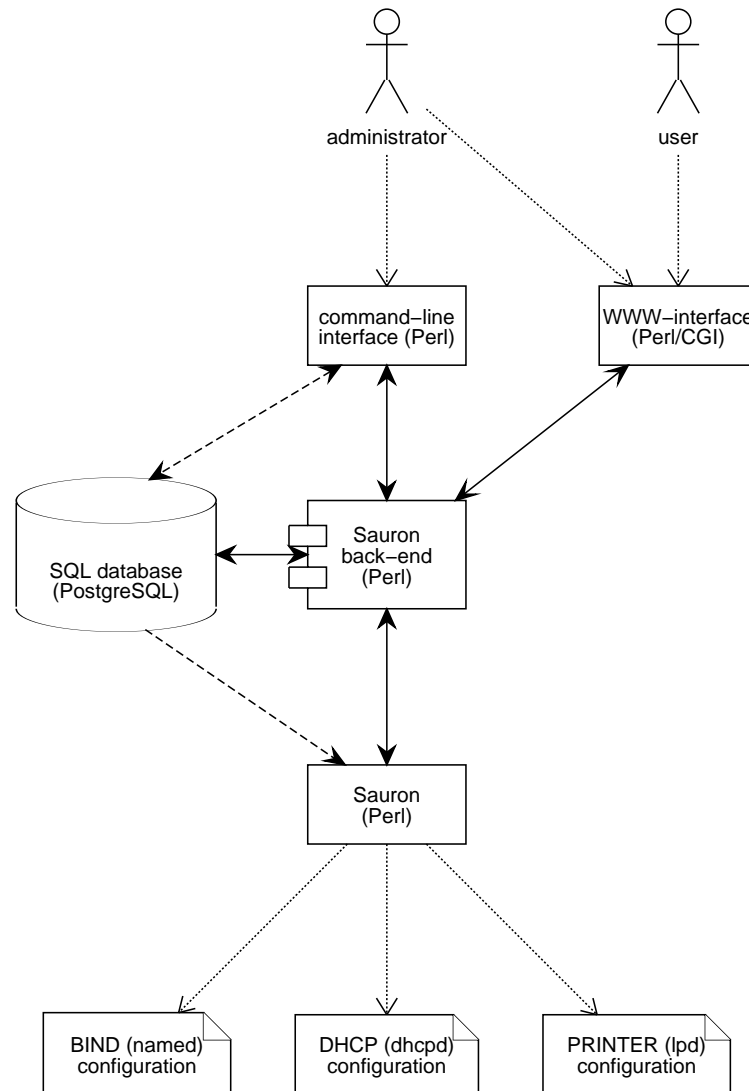


$Id: overview.gd,v 1.2 2001/03/01 21:56:58 tjko Exp $

### 1.2.1. Features

Some of the main features of Sauron include:

- Generates complete BIND (DNS) configuration file set for each server
- Generates complete dhcpd (ISC's DHCP) configuration file set for each server
- Automatically generates DNS "reverse" zones
- Supports smaller than class-C "reverse" zone delegations
- Dynamic host aliases (both CNAME and A record)
- Support for dynamic DHCP address pools with access controls (requires DHCP v3)
- DHCP fail-over support (requires DHCP v3)
- MX and WKS record templates for hosts
- Groups; hosts may be assigned into a group that defines common attributes for it
- Support for most of the standard DNS RRs (resource records)
- Support for automatic validation of generated DNS and DHCP configuration file sets

## 1.3. Sauron Home Page

Currently Sauron homepage is located at: *http://sauron.jyu.fi/* (http://sauron.jyu.fi/)

## 1.4. Software Distribution Sites

Sauron can be downloaded from following sites:

*Primary Site (HTTP)* (http://sauron.jyu.fi/pub/sauron/)
*Secondary Site (FTP)* (ftp://ftp.jyu.fi/pub/sauron/)
*Secondary Site (HTTP)* (http://ftp.jyu.fi/ftp/pub/sauron/)

## 1.5. Mailing lists

- `<sauron-announce@lists.jyu.fi>` - new release announcements and other important messages (moderated)

- `<sauron-users@lists.jyu.fi>` - general discussion and help forum

- `<sauron-devel@lists.jyu.fi>` - development and bug tracking

To subscribe any of these mailing lists, send an empty e-mail to *<list-name>*-request@lists.jyu.fi address with subject line "subscribe" (where <list-name> is the name of the list you want to subscribe).

For example, to subscribe `<sauron-users@lists.jyu.fi>` you would need to send following empty e-mail:

```
To: sauron-users-request@lists.jyu.fi
Subject: subscribe
```

# 1.6. Acknowledgements

Special thanks for following people, who have been very helpful to the development of Sauron:

Teemu Lähteenmäki `<tola@iki.fi>`
Riku Meskanen `<mesrik@iki.fi>`
Otto J. Mäkelä `<om@iki.fi>`

# 1.7. Copyright Notice

Sauron is Copyright (C) 1999-2003 Timo Kokkonen.

Sauron is Copyright (C) 1999-2003 Computing Center, University of Jyvaskyla.

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 675 Mass Ave, Cambridge, MA 02139, USA.

# Chapter 2. Installation

This chapter covers the general installation process for Sauron. This includes software installation and initial system configuration. Make sure you have all the required programs installed, before starting to install Sauron.

## 2.1. System Requirements

Sauron is developed under Linux and Solaris, but is should work pretty much on any *nix like OS (as long as you can get the required programs on your platform). Sauron is currently written entirely in Perl (with some supporting shell-scripts).

### 2.1.1. Required Programs

Following programs are needed to run Sauron.

- PostgreSQL v7.x or later (7.2.3 or newer recommended)
- Perl 5 or later (at least 5.6 recommended)
- Perl modules (these can be found from CPAN):
  - CGI v2.752 or later (older versions should work fine too)
  - Digest::MD5 v2.20 or later
  - Net::DNS v0.33 or later
  - Net::Netmask v1.9002 or later
  - Pg v1.9.0 or later (PostgreSQL interface)
- Apache or any other WWW server that supports CGI (for WWW interface)

### 2.1.2. Related Programs

To use DNS & DHCP configurations generated by Sauron you will also need *Internet Software Consortium's* (http://www.isc.org/) DNS & DHCP programs.

- BIND 8.2.x or later (9.2.x or newer recommended)
- DHCP v2 or later (3.x or newer strongly recommended)

## 2.2. Installing Sauron

Sauron is currently available as a source "tarball" as well as a RPM package (RPM package is made for RedHat Linux systems). Decide which version you want to use to install Sauron from and proceed to either Section 2.2.1 or Section 2.2.2.

### 2.2.1. Installing from the sources

First step, when installing from the source archive is to unpack the tar file into a directory and change current directory to that directory. This can be done as follows:

```
gunzip -c sauron-0.5.0.tar.gz | tar xvf -
cd sauron-0.5.0
```

Next step is to install Sauron from the sources. By default Sauron program files will be installed under `/usr/local/sauron`, and configuration files under `/usr/local/etc/sauron`. If you want to install Sauron under some other directories you need to specify *--prefix* and *--sysconfdir* arguments when invoking configure script. To install Sauron under default directories issue following commands:

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

### 2.2.2. Installing from the RPM

To install (or update) Sauron from RPM package you can simply install the package as follows:

```
rpm -Uvh sauron-0.5.0-1.noarch.rpm
```

## 2.3. Sauron Configuration

Sauron configuration settings are stored in `config` file, which is usually located in `/usr/local/etc/sauron/` or in `/etc/sauron/` (if Sauron is installed from RPM). This configuration file is parsed actually as a Perl code, so it is possible to set the configuration variables conditionally if desired.

Configuration file contains common configuration settings for all Sauron components (except browser.cgi which uses it's own configuration file `config-browser`). Here we cover only the basic

settings that you need to set before using Sauron. For more advanced settings, see the configuration file where all the advanced configuration options are documented.

## General Configuration Options

DB_CONNECT

Database connection parameters string. This defines the database to connect to, and optionally database username and password to use. See Section 2.4 how to setup a database for Sauron.

To setup connection to a database named sauron in localhost, you might want to set this variable to:

```
"dbname=sauron user=foo password=bar"
```

Or, if you're connecting to a database at remote computer:

```
"host=remotehost port=5432 dbname=sauron user=foo password=bar"
```

**Table 2-1. Database Connection String Parameters**

| Parameter | Default value | Description |
|---|---|---|
| host | localhost | Hostname of the database server. |
| port | 5432 | Port to connect on database server. |
| dbname | current user-id | Database name to connect. |
| user | current user-id | Database username. |
| password | | Password. |

PROG_DIR

Base directory for Sauron installation. This should be set to the directory where program files where installed. Usually this is either `/usr/local/sauron` or `/opt/sauron` (if installed from RPM).

LOG_DIR

Directory for log files. Currently only the web interface (`sauron.cgi`) uses this directory to write to `sauron.log` file in this directory. Remember to make sure that the `saurno.cgi` has permissions to write into this file.

SERVER_ID

Unique identifier for this Sauron installation. This identifier is used by the web interface to generate unique cookies names, so you can access several different Sauron web interfaces simultaneously from your browser. This identifier is also displayed by the web interface; to help you to identify which Sauron web interface you're currently connected to.

# 2.4. Database Configuration

To use Sauron, you need to create a database for it in PostgreSQL. Unless you plan on restoring existing Sauron database from a database dump, you also need to setup necessary tables for Sauron to use.

## 2.4.1. Creating database for Sauron

Choose name for your database (for example: `sauron`) and create empty database in PostgreSQL using createdb program.

```
createdb sauron
```

After creating the database, you may either restore existing Sauron database dump or start from "scratch" and create necessary tables for Sauron as described in following Sections.

## 2.4.2. Initializing Database

Before using Sauron, you need to initialized the database. For this there is **createtables** utility that is included with Sauron. To initialize the database, simply run following command in Sauron program directory:

```
./createtables
```

This command will create all the necessary tables needed by Sauron. You can check that everything worked by issuing the following command:

```
./status
```

This should produce output similar to following showing that the database connection works:

```
Sauron v0.5.0 status

Database connection:   OK
Database version:      1.0
CGI interface:         Enabled

No servers found in database!
```

It is also recommended to check that the Perl modules in use meet the requirements (see Section 2.1). This can be done by running **status** command with `--versions` option:

```
./status --versions
```

### 2.4.3. Initializing Global Tables

Sauron has several global tables that contain data that is used by all servers stored in a Sauron database. These tables contain a table (`ether_info`) that contains mappings of OUI's to Manufacturers that is used to display the manufacturers of NICs based on the Ethernet MAC address. There is also a table (`root_servers`) that contains global (default) root server information, and optionally also server specific root server information.

To populate OUI (`ether_info`) table with the included files, issue following commands in the Sauron top-level directory:

```
./import-ethers contrib/Ethernet.txt
./import-ethers --force contrib/additional-ether-codes.txt
```

You may also want to download IEEE's up to date public list of OUIs from *IEEE's web site* (http://standards.ieee.org/regauth/oui/index.shtml), and install it as follows:

```
./import-ethers oui.txt
```

You should also add default root servers into `root_servers` table. This can be done as follows:

```
./import-roots default contrib/named.root
```

NOTE! remember to check for newer version of this file at *ftp.rs.internic.net* (ftp://ftp.rs.internic.net/domain/).

## 2.5. WWW Server Configuration

To be able to use Sauron's web interface you need to make `sauron.cgi` and `browser.cgi` (if you want to use the Sauron DNS browser) available through your WWW server. To do this you typically would either make symbolic links to these files or copy these files in your `cgi-bin` directory. Note, if you're installing from the RPM package, these symbolic links are created automatically, so you only need to check that your web server configuration allow symlinks in your cgi-bin directory.

You should use your web server configuration and/or firewall to limit access to `sauron.cgi` and `browser.cgi` CGI scripts only from the machines you need to access Sauron from.

Sauron web interface uses several icons and images that are by default installed under the top-level Sauron directory. You need to make these available through your web server via `/sauron/icons/` directory (unless you're installing from the RPM which does this automatically for you). To do this you can create that directory under your WWW server root and then either make symbolic links or copy all the image files from `<sauron top-level directory>/icons/` into this directory.

# Chapter 3. Getting Started

This chapter is intended as a brief overview of how to build a simple DNS & DHCP server configuration using Sauron. For those in hurry, there is also small demonstration database that can be used to test Sauron quickly (see Section 3.2.3).

## 3.1. Creating Administrator Account

First step before using Sauron is to create administrator account(s). Administrator accounts are normal user accounts that have superuser flag enabled. For adding accounts there is command-line utility adduser that can be used to add user accounts either interactive or batch mode. To add administrator (superuser) account interactively, invoke adduser with option `--superuser`:

```
./adduser --superuser
```

Here is sample output from adduser program:

```
Enter username: admin
Enter group name (empty for none):
Enter user description (full name): Administrator
Enter user email address: hostmaster@middle.earth
Enter optional user info:
Enter account expiration date (dd-mm-yyyy, +<n>d, +<n>y) [none]:
Enter password [JD!JqRGg]:
          Username: admin
             Group: <none>
          Longname: Administrator
             email: hostmaster@middle.earth
           comment:
        expiration: <none>
         superuser: true
Add this user [y/n]?y
User admin added successfully.
```

Normal users accounts can be added interactively by starting adduser without any arguments. Difference between superuser accounts and normal user accounts is that normal accounts need to be granted permission to access/modify desired servers/zones/hosts. See Chapter 4 for more information about managing user accounts and privileges.

## 3.2. Creating New Server

Sauron can manage multiple servers, where each "server" is contains its own DNS & DHCP server configuration information. This allows single Sauron system to manage multiple DNS & DHCP servers. Normally each servers are completely separate from each other, but it is possible to generate "slave"

servers that use another server as a master server, and inherit it's all master zones (as slave zones), and optionally slave server can also inherit master server's access controls (for DNS server).

There are two ways to create a new server in Sauron. One is to import existing DNS & DHCP configuration into Sauron, for easier migration into use of Sauron. Another way is to start from "scratch" and create new server and zones interactively using Sauron's web interface. Following sections show example how to do this both ways (there is also section showing how to use the included demo database to take a quick look how Sauron works; Section 3.2.3).

## 3.2.1. Using Web Interface

First, login using the web interface as an administrator into Sauron. You should now be in "Select server" dialog, with empty list of servers to choose from. Now select "Add" to add a new server. See Table 5-1 for information of the fields used in this dialog.

Next step is to define global DHCP server settings, by choosing Edit from Servers menu. Now you can add desired DHCP settings in the Global DHCP Settings field, one setting per line without the semicolon (;) at the end of line. Here's some typical global DHCP settings:

```
authoritative
allow bootp
use-host-decl-names on
default-lease-time 7200
max-lease-time 38400
option domain-name-servers ns1.middle.earth,ns2.middle.earth
option ntp-servers ntp1.middle.earth,ntp2.middle.earth
```

After creating a server, it's time to add networks into Networks table in Nets menu. Generate net for each network that has been delegated for your use (and you're generating "reverse" zones for). You should also generate subnets within these networks that describe your network topology. If your network is split up into VLANs / Layer-2 Networks (or "Shared Networks" as DHCP calls them), add you necessary VLANs using Add vlan before, and assign each subnet into appropriate VLAN. It is also possible to create "virtual subnets" that are used only for access control purposes by setting "DHCP" setting for these subnets to false.

Next step is to add the default zones (localhost and reverse zones for 0.0.0.0/8, 127.0.0.0/8, and 255.0.0.0/8) into the server. This can be done from Zones menu using Add Default Zones, which should produce following output:

```
Adding zone: localhost...OK (id=1)
Adding zone: 127.in-addr.arpa...OK (id=2)
Adding zone: 0.in-addr.arpa...OK (id=3)
Adding zone: 255.in-addr.arpa...OK (id=4)
```

Now its time to add your zone(s) into the server. This can be done from Zones using the Add command. See Table 5-3 for description of the fields used in this dialog. After creating a zone, edit the zone by

selecting Edit from Zones menu, and add Name server (NS) entries and other needed entries for the zone. See Table 5-4 for description of the fields used in this dialog.

Finally after creating the zone(s) you want. You may start adding hosts into your zones. Before adding hosts make sure you've selected correct server and zone (Note, you can save your current zone and server selection as defaults from Login menu using the Save Settings command). To add a new host simply select Add host from Hosts menu. If you want to make a "copy" of existing host, first select the host and press Copy button. This will fill out the new host dialog with information from the current host. See Table 5-6 for description of the fields used in this dialog.

There are also templates and groups that you may define and use when adding/modifying hosts. To makes it easier to define or change certain settings for large groups of hosts, by just modifying the related template or group. More information about these can be found in Section 5.4.

## 3.2.2. Using Existing DNS/DHCP Configuration

For importing existing BIND configuration files and existing DHCP configuration file there is `import` and `import-dhcp` utilities. These utilities can be used to import new server into existing Sauron database.

First step is to import your existing BIND configuration into Sauron. This can be done as follows:

```
./import ns1 /etc/named.conf
```

Where import will read your BIND configuration from `named.conf`. Import will read also master zone-files from directories specified in `named.conf`. Use `--dir=<directory>` to specify alternate location for zone-file if they're not located in the directories specified in `named.conf`.

Since BIND configuration does not contain information about your network topology nor the hardware (Ethernet) addresses of hosts, you should also import your existing DHCP configuration file `dhcpd.conf` using `import-dhcp`. This can be done as follows:

```
./import-dhcp --global ns1 /etc/dhcpd.conf
```

This will update hosts in the server "ns1" with information from dhcpd configuration file. Option `--global` causes global dhcp settings to be imported into the server record.

## 3.2.3. Using Demonstration Database

There is small demonstration database included with Sauron distribution. This database is a dump of Sauron database that contains sample server, which demonstrates some features of Sauron.

To use this database dump you need to create a new database in PostgreSQL and restore this dump into it using `psql` utility. This can be done as follows (from the sauron top-level directory):

```
cd test
createdb sauron-demo
psql sauron-demo -f sauron-demo.dump
```

After creating the database you need to set DB_CONNECT variable in `config` file to connect to this database.

This demonstration database has three user accounts already setup:

**Table 3-1. Demo Database Accounts**

| account | password | description |
|---------|----------|-------------|
| admin | admin | sample administrator account |
| user1 | user1 | user with minimal rights |
| user2 | user2 | user with more rights |

# 3.3. Generating DNS & DHCP Configurations

Generation of BIND and dhcpd configurations files is done using command-line utility `sauron`. Configuration files can be generated at the server where Sauron and PostgreSQL are installed, or Sauron software can be installed also in the actual DNS & DHCP server and configured to connect to PostgreSQL database over network (if connecting PostgreSQL over the network SSL connection should be used).

To generate complete BIND configuration file set as well as dhcpd configuration file, following command could be used:

```
./sauron --bind --dhcp --updateserial ns1 /opt/ns/
```

This will create configuration files for server "ns1". Option `--updateserial` forces the update of zone SOA serial numbers. Normally Sauron tries to detect if zone serial number update is necessary and update it only when needed, but this detection routine may sometimes fail, so for now it's recommended to use `--updateserial` option.

Sauron should always generate valid configuration files for BIND and dhcpd, but to be sure and automatically validate generated files, there is `--check` option that causes `sauron` command to check generated files. For this to work you need to set specify the locations of `dhcpd`, `named-checkconf`, and `named-checkzone` programs in Sauron configuration file. Here's an example for enabling `--check` option to work in a RedHat Linux:

```
# set to enable dhcpd.conf validation from sauron
$SAURON_DHCP_CHK_PROG = '/usr/sbin/dhcpd';
$SAURON_DHCP_CHK_ARGS = '-q -t -cf';

# set to enable named.conf validation from sauron
$SAURON_NAMED_CHK_PROG = '/usr/sbin/named-checkconf';
$SAURON_NAMED_CHK_ARGS = '';

# set to enable zone file validation from sauron
$SAURON_ZONE_CHK_PROG = '/usr/sbin/named-checkzone';
```

```
$SAURON_ZONE_CHK_ARGS = '-q';
```

It is also possible to tell `sauron` to send e-mail notifications to users whose changes are in effect after new configuration files are put into use using the `--mail` option. For this option to work you need to configure a mailer program for Sauron to use in configuration file. Here's example how to do this in a RedHat Linux:

```
# mailer configuration (set to enable email notifications)
$SAURON_MAILER = '/usr/sbin/sendmail';
$SAURON_MAILER_ARGS = '-t';
$SAURON_MAIL_FROM = 'postmaster@middle.earth';
```

This configures Sauron to use Sendmail for sending e-mail notifications, and messages will appear to come from <postmaster@middle.earth>.

You can automate BIND & dhcpd configuration files generation using for example a simple shell script that is run periodically from cron. But you should use `--check` option when doing this to make sure you always end up with valid configuration files.

# Chapter 4. Account Management

Currently only the Sauron's Web Interface uses user accounts and privileges. A text-based interface (to be used via SSH or locally) is planned and will use the same user accounts and privileges as the web interface.

There are two ways to assign privileges to users either directly setting privileges to a user or assigning user to a user group and then setting privileges to that user group. It is recommended to use user groups to grant users privileges, since when there is large number of user accounts, it is much easier to modify permissions of a user group than having to modify each user's permission individually.

> There is no access control when using the command-line utilities. Thus, it may be necessary protect access to the Sauron database (in PostgreSQL) and configuration file `config` (if it contains password to access the database).

## 4.1. Managing User Accounts

For managing user accounts there are several command-line utilities: `adduser`, `moduser`, and `deluser`. Following sections show how to do basic administrative tasks using these utilities.

### 4.1.1. Creating User

User accounts are created using `adduser` utility. This utility allows adding user accounts either interactively or in batch mode by specifying all the required information on command-line. Following example shows how to add normal user account interactively (see Section 6.3 for information how to use this command in non-interactive mode). To add new user into Sauron simply run `adduser` command from Sauron top-level directory:

```
./adduser
```

When started without any arguments (or without `--user` and `--passwd` option), interactive mode is started. Below is an example session how to add an account:

```
Enter username: pippin
Enter group name (empty for none): hobbits
Enter user description (full name): Peregrin Took
Enter user email address: pippin@shire.middle.earth
Enter optional user info:
Enter account expiration date (dd-mm-yyyy, +<n>d, +<n>y) [none]:
Enter password [UtH4RCda]:
        Username: pippin
           Group: hobbits (GID=1)
        Longname: Peregrin Took
           email: pippin@shire.middle.earth
         comment:
      expiration: <none>
       superuser: false
```

```
Add this user [y/n]?y
User pippin added successfully.
```

When creating account and not setting group for that account, `adduser` will remind that some privileges should be set for the user, before account can be used. This is not the case when assigning user to a group, since it's assumed that group contains necessary privileges for the account. See Section 4.1.2 for more information about user privileges and how to modify them.

## 4.1.2. Setting Privileges

Accounts must have some privileges (except superuser/administrator accounts) for them to be useful. Privileges can be assigned directly to the user account using `moduser` utility or indirectly to a group using `modgroup` utility.

Typically user (or group) is assigned one or more "server" privileges that grant user read-only access into those server(s). Then user is assigned "zone" privileges that grant user access into all hosts inside certain zone(s). If needed, user's access into all hosts inside the zone(s) user has write access can be further limited by assigning restrictive privileges to limit users access to only a subset of hosts inside the zone(s). For this there are rules to limit access by hostname (regexp) or by IP number (either by network/subnet or by IP-mask). See Table 4-1 for complete list of available user/group privileges.

**Table 4-1. Account/Group Privileges**

| Type | Target | Description | Value |
|------|--------|-------------|-------|
| Server | server ID | Controls the access to a server and all zones inside the server. | R=read access, RW=read/write access, RWX=read/write access (with less restrictions) |
| Zone | zone ID | Controls the access to a zone. | R=read access, RW=read/write access, RWX=read/write access (with less restrictions) |
| Net | net ID | Limits users ability to use only IP addresses within "auto assign ranges" of the networks/subnets listed by the Net rules. | |
| IP mask | | Limits users ability to use only IP addresses to those matching the IP-mask | IP-mask (for example: 192.168.1-10.* |
| Hostname mask | | Limits user to be able to modify only hosts with domainname matching any of Hostname mask rules. | regular expression (for example: ^\.math$) |

| Type | Target | Description | Value |
|------|--------|-------------|-------|
| Authorization Level | | Defines user's authorization level. This can be used to control users access into templates/groups, and also to control how much information about networks is displayed to the user. | authorization level (0-999, 0=default) |
| Host Expiration Limit | | If defined, user can only create host entries that expire in n days or sooner. | expiration limit (in days) |
| Default Department | | Default Department field value for new hosts. | Department Name |

In addition to modifying the user account itself, `moduser` utility can also be used to modify user privileges. To add privileges interactively for given account, just invoke `moduser` with `--add` option. For example:

```
./moduser pippin --add
```

After adding privileges to a user, `moduser` utility can be also used to display user's current privileges by giving only a username as argument. For example:

```
./moduser pippin
```

This should produce output similar to below displaying account details and privileges.

```
           Username: pippin (id=2)
           longname: Peregrin Took
              email: pippin@shire.middle.earth
               info:
          superuser: No
              group: hobbits (gid=1)
      passwd status: valid (CRYPT)
         last login:
 account expiration: <never>
    account created: Sat Feb 15 00:18:40 2003 by tjko
      last modified: <never>

ID     Type     Ref.                     Mode
------ -------- ------------------------ ------------------------------
[User privileges]
3      hostname \.shire$                 (hostname constraint)

[Group (hobbits) privileges]
1      server   test                     R
```

```
2       zone      test:middle.earth          RW
```

### 4.1.3. Removing User

User account can be removed using `deluser` utility. This utility works interactively when started without any arguments. To remove user(s) non-interactively use `--user=<username>`

To remove user account interactively, simply start `deluser` without any arguments from Sauron top-level directory:

```
./deluser
```

### 4.1.4. Disabling User Temporarily

It is also possible to disable (lock) user accounts without actually removing them, using `moduser` utility with `--lock` option.

Locking user account "pippin", issue following command from Sauron top-level directory:

```
./moduser pippin --lock
```

### 4.1.5. Listing Users

To list all users in Sauron database, use `moduser` command with `--list` option. This will list all the users as well as user's group, name, and email information. This can be done from Sauron top-level directory as follows:

```
./moduser --list
```

## 4.2. Managing User Groups

User groups are meant to ease the user privilege management, by allowing user privileges to be set for a user group and then user can be assigned to a desired user group. User inherits all the privileges from the

group she belongs to. User can still have user specific privileges defined, even if user belongs to a group. User's actual privileges are combination of user specific and group based privileges.

## 4.2.1. Creating Group

User groups can be created using `addgroup` utility. To add a user group interactively start `addgroup` program from Sauron top-level directory as follows:

```
./addgroup
```

Here is example how to add a group named "hobbits":

```
Enter group name: hobbits
Enter group description: Shire Hobbits
        Groupname: hobbits
        Description: Shire Hobbits
Add this group [y/n]?y
Group hobbits added successfully.
```

After creating a new group, it should be assigned some privileges as described in Section 4.2.2.

## 4.2.2. Setting Group Privileges

User groups (like User accounts) must have some privileges assigned to them for them to be useful. Privileges can be assigned directly to the user group using `modgroup` utility. Privileges that can be set to a user group are same as the privileges for a user. For more information about privileges see Section 4.1.2.

To add privileges for a user group interactively, start `modgroup` utility with the `--add` option. This can be done as follows from the Sauron top-level directory:

```
./modgroup hobbits --add
```

After adding desired privileges to the group, `modgroup` utility can also be used to display group's privileges by starting it with only the group name as argument.

```
./modgroup hobbits
```

This should produce output similar to following:

```
 Groupname: hobbits (id=1)
   Comment: Shire Hobbits

ID     Type     Ref.                      Mode
------ -------- ------------------------- -------------------------------
1      server   test                      R
2      zone     test:middle.earth         RW
```

### 4.2.3. Removing Group

User group can be removed using `delgroup` utility. This utility works interactively when started without any arguments. To remove user(s) non-interactively use `--group=<groupname>`

To remove user group interactively, simply start `delgroup` without any arguments from Sauron top-level directory:

```
./delgroup
```

## 4.3. External User Authentication

Normally users are authenticated against encrypted passwords stored into `users` table in Sauron database. Sauron currently uses by default standard Unix passwords. However, it is possible to use external authentication program for user authentication. Which makes it possible to authenticate users against almost any service.

To enable external user authentication, `SAURON_AUTH_PROG` in variable in Sauron configuration file must be set to point to external authentication script or program.

A sample script (`kerberos-auth`) for authenticating against a Kerberos KDC is included in the `contrib/` directory.

If `SAURON_AUTH_PROG` is set in the configuration file, Sauron will run the authentication program every time user is authenticated. Authentication program is expected to read one line from standard input that is in following format:

```
username password
```

Authentication program should then authenticate the user using the given username and password, and return with exit code 0 (zero) if authentication was successful. Otherwise a non-zero exit code should be used.

# Chapter 5. Web Interface

Web based user interface for Sauron is implemented as a single CGI script (`sauron.cgi`). To use this script WWW server software that supports CGI programs is needed. Currently this interface is developed and tested using Apache, but it should work with any WWW server software, as long as it has support for CGIs.

It is recommended to setup your WWW server to use HTTPS and only allow access to the `sauron.cgi` CGI script using encrypted HTTPS connection. If your WWW server supports both unencrypted (HTTP) and encrypted (HTTPS) connections, it is recommended to setup a separate "cgi-bin" directory for HTTPS connections and place `sauron.cgi` (or symbolic link to it) only in that directory.

Although `sauron.cgi` has it's own access control mechanism, it is recommended to further improve security by restricting access to the web interface for only the hosts that need access to the Sauron using the WWW server configuration and/or a firewall.

Sauron's web interface doesn't use JavaScript, only standard HTML is used. Goal is to keep the web interface lightweight and usable by all browsers (graphical as well as text-based).

For session tracking `sauron.cgi` uses cookies. Each generated cookie contains only a MD5 hash of user's session identifier and has maximum life of a week. Normally the cookie is deleted when user logs out. Cookies are only sent back to the issuing server. Sauron supports the use of "secure cookies" via configuration file setting `SAURON_SECURE_COOKIES` (this is not enabled by default since some buggy browsers don't handle secure cookies correctly).

# 5.1. Servers Menu

Servers menu provides commands for creating and modifying servers for administrator. For normal user it only provides server selection command for selecting the active server.

## 5.1.1. Show Current

Show Current command displays the currently selected server information. It is default command when entering Servers menu.

## 5.1.2. Select

Select command displays list of available servers for selecting active server. For normal user's this list will include only the servers user has at least read access to.

## 5.1.3. Add

Add command is used to create a new server. This option is available only for administrator. See Table

5-1 for descriptions of the fields used in the command dialog.

**Table 5-1. New Server Dialog**

| Field | Description | Example |
|-------|-------------|---------|
| Name | Short name of the server (this handle can contain only letters, numbers, and hyphens) | ns1 |
| Hostname | Domainname of the server (FQDN) | ns1.middle.earth. |
| IP address | IP address of the server | 192.168.1.1 |
| Hostmaster | Default hostmaster email address for all zones in the server (replace "@" in the email address with ".") | hostmaster.middle.earth. |
| Configuration directory | Base directory for BIND configuration files | `/var/named` |
| Slave for | Option that allows to select a master server for this server (making it a slave). Slave server will automatically inherit all the master zones of it's master as slave zones. Slave server can also optionally inherit DNS server access controls from the master server. | None |
| Comment | Long description of the server (optional) | primary name server |

## 5.1.4. Delete

Delete command allows removal of a server. This command should be used with caution, since it will remove the entire server from the database. This command is available only for administrator.

## 5.1.5. Edit

Edit command is used to edit existing server. This option is available only for administrator. See Table 5-2 for descriptions of the fields used in the command dialog.

**Table 5-2. Edit Server Dialog**

| Field | Description | Example |
|-------|-------------|---------|

| Field | Description | Example |
|---|---|---|
| Server Name | Short name of the server (this handle can contain only letters, numbers, and hyphens) | ns1 |
| Hostname | Domainname of the server (FQDN) | ns1.middle.earth. |
| IP address | IP address of the server | 192.168.1.1 |
| Output mode | Option for selecting wheter to generate full `named.conf` or `named.zones` file that contains only zones (to be included in your static `named.conf`). | "Generate full named.conf" |
| Comments | Optional field reserved for comments | |
| Hostmaster | Default hostmaster email address for all zones in the server (replace "@" in the email address with ".") | hostmaster.middle.earth. |
| Refresh | Zone SOA record default "refresh" value | 43200 |
| Retry | Zone SOA record default "retry" value | 3600 |
| Expire | Zone SOA record default "expire" value | 2419200 |
| Minimum (negative caching TTL) | Zone SOA record default "minimum" value | 86400 |
| Default TTL | Default TTL for records in zones for this server | 86400 |
| Default zone TXT | TXT records that get automatically included for each zone in this server | |
| Configuration directory | Base directory for BIND configuration files | `/var/named` |
| Primary zone-file path | Pathname relative to configuration directory for storing master zone files | |
| Slave zone-file path | Pathname relative to configuration directory for storing slave zone files | NS2/ |
| Root-server file | filename for root server (zone) file. | named.ca |
| pid-file path | pathname for pid-file (BIND option) | |
| dump-file path | pathname for dump-file (BIND option) | |

| Field | Description | Example |
|---|---|---|
| statistics-file path | pathname for statistics-file (BIND option) | |
| memstatistics-file path | pathname for memstatistics-file (BIND option) | |
| named-xfer path | pathname for named-xfer (BIND option) | |
| Forward (mode) | BIND forward setting; default, only, first | default |
| Forwarders | BIND forwarders setting (list of server IPs to forward queries to) | |
| Transfer source IP | Source IP address for zone transfers (for multi-homed hosts) | |
| Query source IP | Source IP address for DNS queries sent by server (for multi-homed hosts) | |
| Query source port | Source port for DNS queries sent by server | |
| Listen on port | Port that the server listens for DNS queries (allows setting non-standard port) | |
| Listen-on | BIND listen-on setting; list of IPs (interfaces) server should listen for queries | |
| Allow-transfer | BIND allow-transfer setting; list of CIDRs from where to allow zone transfers. | |
| Allow-query | BIND allow-query setting; list of CIDRs from where to allow DNS queries. | |
| Allow-recursion | BIND allow-recursion setting; list of CIDRs from where to allow recusive DNS queries (causes recursion to be disabled from anywhere else). | |
| Blackhole | BIND blackhole setting; list of CIDRs from where any requests are to be ignored by the server. | |
| Do not generate HINFO records | Controls whether to omit HINFO records from the generated zone files or not. | No |
| Do not generate WKS records | Controls whether to omit WKS records from the generated zone files or not. | No |

| Field | Description | Example |
|---|---|---|
| Notify | BIND notify setting; default, yes, no | Default |
| Auth-nxdomain | BIND auth-nxdomain setting; default, yes, no | Default |
| Recursion | BIND recursion setting; default, yes, no | Default |
| Dialup mode | BIND dialup setting; default, yes, no | Default |
| Allow multiple CNAMEs | BIND multiple_cnames setting; default, yes, no | Default |
| RFC2308 Type 1 mode | BIND rfc2308_type1 setting; default, yes, no | Default |
| Check-names (Masters) | BIND check-names master setting; default, fail, ignore, fail | Default |
| Check-names (Slaves) | BIND check-names slave setting; default, fail, ignore, fail | Default |
| Check-names (Responses) | BIND check-names response setting; default, fail, ignore, fail | Default |
| Version string | If this is set then this string is that BIND will report as it's version (instead of the real version) | |
| Logging options | List of BIND logging options (these are placed inside "logging" section in `named.conf`) | |
| auto-domainnames | Controls wheter to automatically generate correct "domain-name" option for each host in `dhcpd.conf`. | No |
| Global DHCP Settings | List of global DHCP settings. These lines are include in the global scope of `dhcpd.conf`. Any valid global DHCP option can be used here (lines should not end with ";"). | |
| Enable failover protocol | Controls wheter DHCP failover protocol is enabled. | No |
| Port number | Port to be used by DHCP failover protocol | 519 |
| Max Response Delay | Max Response Delay for DHCP failover protocol | 60 |
| Max Unacked Updates | Max Unacked Updates for DHCP failover protocol | 10 |

| Field | Description | Example |
|-------|-------------|---------|
| MCLT | Maximum Client Lead Time (MCLT) for DHCP failover protocol | 3600 |
| Split | Split for DHCP failover protocol | 128 |
| Load balance max | Load balance max (seconds) for DHCP failover protocol | 3 |

# 5.2. Zones Menu

Zones menu contains commands for creating and modifying zones for administrator. For normal user it only provides zone selection command for selecting the active zone.

## 5.2.1. Show Current

Show Current command displays the currently selected zone information. It is default command when entering Zones menu. Name of the currently selected zone can be seen on the left side of the screen in "Current selections" box.

## 5.2.2. Show Pending

Show Pending command displays list of pending changes for hosts in this zone. This list currently displays all new or modified hosts (or aliases) since DNS configuration was last time generated.

Also when viewing a host record, there is word PENDING at the end of the "Record created" or "Last modified" fields if changes to this record are not yet in effect.

## 5.2.3. Select

Select command displays list of available zones in current server for selecting active zone.

## 5.2.4. Add

Add command is used to create a new zone. This command is only available for administrator. See Table 5-3 for descriptions of the fields used in the command dialog.

**Table 5-3. New Zone Dialog**

| Field | Description | Example |
|-------|-------------|---------|

| Field | Description | Example |
|---|---|---|
| Zone name | Zone name (without the dot in the end) | middle.earth |
| Type | Zone type; Master=master zone, Slave=slave zone | Master |
| Reverse | Reverse zone flag; Yes=reverse zone, No=normal zone | No |

## 5.2.5. Add Default Zones

Add Default Zones command is used to automatically add the standard zones for normal DNS server. This command will try to add following zones into current server:

```
localhost
127.in-addr.arpa
0.in-addr.arpa
255.in-arrd.arpa
```

This command is only available for administrator.

## 5.2.6. Copy

Copy command copies entire zone (including all the hosts in a master zone) under new name. This option can be used to use existing zone as a template when creating new zones. This command is available only for administrator.

## 5.2.7. Delete

Delete command allows removal of a zone. This command should be used with caution, since it will remove the entire zone from the database. This command is available only for administrator.

## 5.2.8. Edit

Edit command is used to edit existing zone. This command is only available for administrator. See Table 5-4 for descriptions of the fields used in the command dialog.

**Table 5-4. Edit Zone Dialog**

| Field | Description | Example |
|---|---|---|
| Zone name | Zone name (without the dot in the end) | middle.earth |

| Field | Description | Example |
|---|---|---|
| Comments | optional comments about the zone | company main zone |
| Info TXT record auto generation | Flag that controls the automatic generation of descriptive TXT records for each host from the host info fields (User,Dept,Location,Info) | Yes |
| "Dummy" zone | Flag when set, causes this zone to be treated as a "dummy" zone that only appears in the `named.conf` but the zone file will not be generated by Sauron. | No |
| Class | Zone class; IN (Internet), CHAOS, HESIOD, HS | IN |
| Hostmaster | Zone specific hostmaster's email address. If empty hostmaster's email address from server record is used. | |
| Check-names | BIND's "check-names" option; default, fail, ignore, warn | default |
| Notify | BIND's "notify" option; default, yes, no | default |
| Refresh | SOA record refresh value. If empy, value from the server record is used instead | |
| Retry | SOA record retry value. If empy, value from the server record is used instead | |
| Expire | SOA record expire value. If empy, value from the server record is used instead | |
| Minimum (negative caching TTL) | SOA record minimun value. If empy, value from the server record is used instead | |
| Default TTL | Default TTL for RRs in this zone. If empy, value from the server record is used instead | |
| IP addresses (A) | Zone A record(s). | 192.168.1.1 |
| Name servers (NS) | Zone NS records (there should always be at least two name servers defined for a zone) | ns1.middle.earth<br>ns2.middle.earth |

| Field | Description | Example |
|---|---|---|
| Mail exchanges (MX) | Zone MX records | 10 mail1<br>10 mail2<br><br>20 mailbackup |
| Info (TXT) | Zone TXT records. | |
| Allow dynamic updates | BIND allow-update option | |
| Allow queries from | BIND allow-query option | |
| Allow zone-transfers from | BIND allow-transfer option | |
| [Stealth] Servers to notify | BIND also-notify option | |
| Zone specific DHCP entries | dhcpd options for all hosts in this zone | |

# 5.3. Hosts Menu

Hosts menu contains commands for creating and modifying hosts. In addition, this menu provides the search commands for searching hosts within a zone.

Most of the host specific commands are not in the Hosts menu itself, but are displayed as buttons when user views a host record. Some of these buttons such as Edit or Delete are only available when user has privileges to make modifications to hosts in that zone.

## 5.3.1. Search

Search command is the default command when entering into Hosts menu. This command provides powerful search engine for hosts within current zone. See Table 5-5 for descriptions of the fields used in the command dialog.

**Table 5-5. Search Hosts Dialog**

| Field | Description | Example |
|---|---|---|
| Record type | Limits search to specific record type; <Any Type>, Host, Host reservation, Delegation, Plain MX, Alias, AREC Alias, Glue record, DHCP only | Host |
| Subnet | Search only hosts within specified network | |

| Field | Description | Example |
|---|---|---|
| CIDR (block) or IP | Lmit search to given CIDR block or IP address (NOTE! this field supersedes the subnet field) | 192.168.1.128/25 |
| Domain pattern (regexp) | Limit search scope to hosts that have domainname that matches regular expression given in this field. | \.sales$ |
| Sort order | Sort search result by IP or hostname | by IP |
| Entries per page | Number of hosts to display per page | 256 |
| Search field | Defines the field that is searched for pattern defined in next field (Pattern). | Ether |
| Pattern (regexp) | Search hosts records for this regular expression in the field defined in "Search field" field. | |

## 5.3.2. Last Search

Last Search command performs the previous search again. This provides a quick way to redo a search you did earlier.

## 5.3.3. Add host

Add host command is used to create a host. This command displays slightly different dialog for administrators or users with "RWX" access to the zone than normal users who will get more restrictive dialog. See Table 5-6 for descriptions of the fields used in the command dialog.

**Table 5-6. Add Host Dialog**

| Field | Description | Example |
|---|---|---|
| Hostname | Hostname (without the zonename part of the host's FQDN) | ws001 |
| Subnet | Select "Manual IP" for manually specifying host's IP number, or select desired network for automatically selecting next available IP number from within that networks auto assign range. | <Manual IP> |

| Field | Description | Example |
|---|---|---|
| IP | Host's IP number (only if "Manual IP" selected in subnet field). | |
| Router (priority) | This field should be zero for hosts. For router interfaces, this should be set to positive priority value (1=highest priority). Priority is used for choosing "best" router interface when generating DHCP config for each subnet. | 0 |
| Group | Group which this host belongs to. | None |
| MX template | MX template to use for this host. | None |
| WKS template | WKS template to use for this host. | None |
| User | primary user/use for this host (informational) | Gandalf the White |
| Dept. | department this host belongs to (informational) | Mages Guild |
| Location | current location of the host (informational) | White Tower |
| Info | extra information about the host (informational) | |
| HINFO hardware | Host hardware description | PC |
| HINFO software | Host software (OS) description | MS-WINDOWS-2000 |
| Ethernet address | Host NIC hardware (MAC) address (this must be unique within zone) | 00:00:01:23:45:66 |
| Asset ID | This field is reserved for organizational asset ID tag (this must be unique within server) | |
| Model | Host model information (informational) | |
| Serial no. | Host serial number (informational) | |
| Misc. | miscellaneous information about the host (informational) | |
| Expiration date | Host will automatically disappear from DNS & DHCP configurations after it has expired (sauron can automatically clean expired hosts from database when run with --clean option) | |

## 5.3.4. Edit (button)

Edit button is used to modify existing host. This button can only be used by normal user if she has necessary privileges to make modifications to the host. See Table 5-7 for description of the fields used in this dialog.

**Table 5-7. Edit Host Dialog**

| Field | Description | Example |
|---|---|---|
| Hostname | Hostname (without the zonename part of the host's FQDN) | ws001 |
| IP address | Host's IP address. | 192.168.1.100 |
| User | primary user/use for this host (informational) | Gandalf the White |
| Dept. | department this host belongs to (informational) | Mages Guild |
| Location | current location of the host (informational) | White Tower |
| [Extra] Info | extra information about the host (informational) | |
| HINFO hardware | Host hardware description | PC |
| HINFO software | Host software (OS) description | MS-WINDOWS-2000 |
| Ethernet address | Host NIC hardware (MAC) address (this must be unique within zone) | 00:00:01:23:45:66 |
| Asset ID | This field is reserved for organizational asset ID tag (this must be unique within server) | |
| Model | Host model information (informational) | |
| Serial no. | Host serial number (informational) | |
| Misc. | miscellaneous information about the host (informational) | |
| Group | Group which this host belongs to. | None |
| MX template | MX template to use for this host. | None |
| WKS template | WKS template to use for this host. | None |
| Expiration date | Host will automatically disappear from DNS & DHCP configurations after it has expired (sauron can automatically clean expired hosts from database when run with --clean option) | |

## 5.3.5. Delete (button)

Delete button is used to delete a host. This button can only be used by normal user if she has necessary privileges to make modifications to the host. If a host has dynamic aliases that point to it those aliases are also automatically removed when the host is deleted.

## 5.3.6. Copy (button)

Copy button can be used to "copy" a host easily. This button will take the information from the current host and use this information to pre-fill Add host command dialog. If hostname contains number it is incremented, otherwise number two (2) is appended to the end of the hostname.

## 5.3.7. Move (button)

Move button is used to move a host into a new subnet. After selecting new subnet for this host, next available IP address within the auto assign address range of that network is suggested as new IP address for the host. This button can only be used by normal user if she has necessary privileges to make modifications to the host.

## 5.3.8. Alias (button)

Alias button is used to add a dynamic (CNAME or A record) alias for a host. This button can only be used by normal user if she has necessary privileges to make modifications to the host.

## 5.3.9. Add alias

Add alias command is used to add a static (CNAME) alias. This command is meant for adding static "out-of-zone" aliases only. Normally aliases should be added using Alias button, which will create a dynamic alias for that host. This command is available for administrator and users with "RWX" access to the zone.

## 5.3.10. Add MX entry

Add MX entry command is used to add plain MX record (without A records or anything). This command is available for administrator and users with "RWX" access to the zone.

## 5.3.11. Add delegation

Add delegation command is used to add a zone delegation (NS records). This command is available for administrator and users with "RWX" access to the zone.

## 5.3.12. Add glue rec.

Add glue rec. command is used to add glue records that are sometimes required for delegations where one ore more delegated zone's name servers are within that zone. This command is available for administrator and users with "RWX" access to the zone.

## 5.3.13. Add DHCP entry

Add DHCP entry command is used to add a "host" entry that will only appear in DHCP configuration, but not in DNS configuration. This command is available for administrator and users with "RWX" access to the zone.

## 5.3.14. Add SRV rec.

Add SRV rec. command is used to add a SRV records. This command is available for administrator and users with "RWX" access to the zone.

## 5.3.15. Add reservation

Add reservation command is used to add a host reservation. Host reservations can be used to reserve hostnames and IP numbers for future use.

## 5.3.16. Network Settings (button)

Network Settings button displays network settings for selected host. This will include IP address, netmask, network address, broadcast address, and default gateway for the host.

## 5.3.17. History (button)

History button displays history of this host record. This will include information like when the record was created, and when it was modified and by whom. This option is available for users who by default have authorization level of 1 or higher (this can be controlled via variable ALEVEL_HISTORY in configuration file).

## 5.3.18. Ping (button)

Ping button allows user to run "ping" command against the selected host. This option is only available when it is explicitly enabled in configuration file and user has at least the required authorization level.

## 5.3.19. Traceroute (button)

Traceroute button allows user to run "traceroute" command against the selected host. This option is only available when it is explicitly enabled in configuration file and user has at least the required authorization level.

# 5.4. Templates Menu

Templates menu contains commands for creating and modifying templates. Only "show" commands under this menu are available for normal users.

Templates are used by hosts, they can be used to specify common MX, WKS, or HINFO records for groups of hosts. Normal users can only use templates that have same or lover privilege level than the user has.

## 5.4.1. Show MX

Show MX command displays list of MX templates defined for this zone. To view specific MX template click on it's name. To edit or delete a MX template use Edit or Delete buttons from the bottom of the screen when viewing a template.

## 5.4.2. Show WKS

Show WKS command displays list of WKS templates defined for this server. To view specific WKS template click on it's name. To edit or delete a WKS template use Edit or Delete buttons from the bottom of the screen when viewing a template.

## 5.4.3. Show HINFO

Show HINFO command displays list of HINFO templates defined for this Sauron installation. To view specific HINFO template click on it's name. To edit or delete a HINFO template use Edit or Delete buttons from the bottom of the screen when viewing a template.

## 5.4.4. Add MX

Add MX command is used to create a new MX template for selected zone. This command is only available for administrator.

## 5.4.5. Add WKS

Add WKS command is used to create a new WMX template for selected server. This command is only available for administrator.

## 5.4.6. Add HINFO

Add HINFO command is used to create a new HINFO template. These templates are global and available to all servers in this database (Sauron installation). This command is only available for administrator.

# 5.5. Groups Menu

Groups menu contains commands for creating and modifying (host) groups. Normal users can only list and view group details; all other commands are for administrator only.

Groups are used by hosts; they are used to specify common attributes for hosts. Normal users can only use groups that have same or lover privilege level than the user has. There are currently three different types of groups.

### Group Types

Normal

> Normal group is used to specify common DHCP and PRINTER entries for hosts.

DHCP class

> This group is like normal group, but in DHCP configuration a class named after this group will be created and all hosts in this group will be "subclassed" into that class. This class can be used for example to allow only hosts in this group to use certain dynamic address pool.

Dynamic Address Pool

> This group is used only for hosts (IPs) that belong to a certain IP address pool for a DHCP server. For each dynamic address pool for DHCP server, group of host entries that belong to this group should be created (IP addresses of these hosts don't necessarily have to be successive, but for each group a separate pool statement will be created in `dhcpd.conf`). When generating DHCP configuration, dynamic address pools are automatically generated for each group of this type. DHCP entries set for this group are put inside "pool" declaration in `dhcpd.conf` (these can be used to deny/allow certain classes of hosts to use this pool).

## 5.5.1. Show Groups

Show Groups command displays list of (host) groups defined for active server. To view details about a specific group, just click on it's name. To edit or delete a group use Edit or Delete buttons from the bottom of the screen when viewing a template.

## 5.5.2. Add

Add command is used to create new (host) group for currently active server. This command is only available for administrator.

# 5.6. Nets Menu

Nets menu contains commands for managing networks and VLANs (Layer-2 networks). Normal user can only see list of networks that have lower or same privilege level set than the user has. VLAN information is by default visible only for users with privilege level 5 or above (this can be controlled via `ALEVEL_VLANS` variable in configuration file).

Networks are divided into two types "Net" and "Subnet". Where "Net" is used for networks that are delegated for your use (and you generate reverse zones for). "Subnet" is used for networks (subnets) within these "Nets" (networks). This information is used (along with VLAN information) to generate network topology for DHCP configuration.

VLANs are used to describe the "Layer-2 network" topology of your network. Each network can be assigned to one VLAN. All networks that don't have VLAN set are part of virtual VLAN called "CHAOS" (normal VLAN names are always lower case).

## 5.6.1. Networks

Networks command lists all the networks that are defined for active server. To view details of a network, just click on the network. To edit or delete selected network use Edit or Delete buttons. For information about selected network use Net Info button that displays technical information about given network and map of IPs currently in use.

## 5.6.2. Add net

Add net command is used to create a new "Net" (network) for active server. This should be used only for networks that are delegated for your use. This command is only available for administrator.

## 5.6.3. Add subnet

Add subnet command is used to create a new "Subnet" for active server. This should be used for all subnets within your official networks. This command is only available for administrator.

## 5.6.4. Edit (button)

Edit button is used to modify existing network. See Table 5-8 for information about fields used in this dialog. This button is only available for administrator when viewing network details.

**Table 5-8. Edit Net Dialog**

| Field | Description | Example |
|-------|-------------|---------|
| Net name | Network (subnet) name (handle that can contain only letters and numbers and hyphen) | sales-net |
| Description | Long description of network | Sales Deparment Network |

| Field | Description | Example |
|---|---|---|
| Net (CIDR) | Network defined as CIDR block | 192.168.1.0/24 |
| VLAN | VLAN (Layer-2 network) this network belongs to | None |
| Authorization level | Authorization level required for user to see this network | 0 |
| Private (hide from browser) | Private network (hide from `browser.cgi`) | No |
| Comment | Optional comments aobut this network | |
| Range start | Auto assign address range start (Sauron will automatically assign new IP addresses from a network that are inside the auto assign range). | 192.168.1.1 |
| Range end | Auto assign address range end. | 192.168.1.254 |
| DHCP | DHCP enabled flag; this network is "visible" to DHCP server only if this is enabled | Enabled |
| Net specific DHCP entries | DHCP configuration options lines specific to this network (don't use semicolon at the end of these lines). | |

## 5.6.5. VLANs

VLANs command lists all VLANs (Layer-2 networks) that are defined for active server. To view details of a VLAN, just click on the network. To edit or delete selected network use Edit or Delete buttons. This command is only available for users with sufficiently high authorization level.

## 5.6.6. Add vlan

Add vlan command is used to create a new VLAN (Layer-2 network) entry for active server. This command is only available for administrator.

# 5.7. Login Menu

Login menu contains miscellaneous commands that mainly provide commands to tune user account settings, and display information about the web interface itself.

### 5.7.1. User Info

User Info command displays general information about current user's account and user privileges (permissions). See Table 4-1 for description about user privileges.

### 5.7.2. Who

Who command displays list of currently logged on users. This list includes username, user's full name, host where user coming from, when user logged in, and how long user has been idle.

### 5.7.3. News (motd)

News (motd) command displays all system news (or message of day) messages. When users login they will see latest three news messages automatically.

### 5.7.4. Login

Login command allows user to login again using different user account.

### 5.7.5. Logout

Login command allows user to logout from the system. This will also try to erase session cookie from user's browser by issuing new cookie "dummy" cookie with expiration time set to "now".

### 5.7.6. Change password

Change password command allows user to change her own password. Password change dialog will prompt for user's current password and new password twice to reduce risk of typing mistakes in new password.

### 5.7.7. Edit settings

Edit settings command allows user to change her email address and toggle automatic email notifications on or off. Automatic email notifications are sent to user's email address when user's changes come to effect (new DNS configuration is generated using `sauron` command with `--mail` option).

### 5.7.8. Save settings

Save settings command save user's current server and zone selection. When user logs on next time, server and zone selections are recalled and user will be in Hosts menu by default.

## 5.7.9. Frames ON/OFF

Frames ON/OFF command allows user to toggle between frames and non-frames operation mode of the web interface. It should be noted that the non-frames mode of the web interface is currently faster and better tested.

## 5.7.10. Lastlog

Lastlog command displays list or resent user sessions (with links to session infos). This command is only available for administrator.

## 5.7.11. Session Info

Session Info command allows administrator to browse log information about user sessions by session id (SID). Log information contains information about what changes user has made to hosts. This command is only available for administrator.

## 5.7.12. Add news msg

Add news msg command allows administrator to add news system news messages that are displayed when to users when they log in.

# Chapter 6. Command Reference

This chapter provides a reference for all the command-line utilities included with Sauron. All these utilities have built-in help, which can be displayed by invoking them with single argument `--help` (these utilities use GNU style long options).

These utilities are meant for Sauron administrators and do not implement any access control methods. To protect unauthorized access into Sauron database, restrict access to the database in PostgreSQL (and don't forget to restrict access to the Sauron configuration file if it contains sensitive information such as database password).

## 6.1. addgroup

This command is used to add a new user group. User groups can be used to simplify user privilege management by assigning privileges to a group and then assigning users to that group.

**addgroup** [--help] [--group=*name*] [--name=*longname*]

This command can be used interactively when run without any arguments, or in patch mode by specifying both `--group` and `--name` arguments on command-line.

## 6.2. addhosts

This command can be used to add host entries into existing zone in Sauron from a text file that is standard zone-file format.

**addhosts** [--help] {*server*} {*zone*} {*file*} [*OPTIONS*]

| option | description |
|---|---|
| --name=*regexp* | Host name filter. Only add hosts if domainname matches to given regular expression. |
| --newonly | Add only hosts that do not already exist in Sauron database. |
| --noreverse | Do not add reverse records. |
| --outofzone | Allow out of zone records. |
| --verbose | Produce more verbose output. |
| --group=*name* | Assign new hosts into given group. |
| --info=*user:dept:location:info* | Set info fields for all imported host to these values. |
| --force=*type* | Force host record in Sauron to be this type: dhcp-only |
| --commit | Commit changes (without this no changes are made into database). |

By default this program will not make any changes to Sauron database, it will only display what would be made. Use `--commit` to make changes into Sauron database.

## 6.3. adduser

This command is used to add a new user for Sauron web user interface.

**adduser**  [--help] [--user=*username*] [--passwd=*password*] [--group=*name*] [--name=*users-full-name*] [--email=*email-address*] [--comment=*comments*] [--superuser] [--expiration=*dd-mm-yyyy*]

This command can be used interactively when run without any arguments, or in patch mode by specifying at least `--user` and `--passwd` options on command-line.

## 6.4. check-pending

Program to be run from cron that check for pending changes in Sauron database and sends notification of pending changes to address given on command-line.

**check-pending**  [--help] {*servername*} {*emailaddress*}

To use this program SAURON_MAILER (and related settings) have to be configured in Sauron configuration file.

## 6.5. delgroup

This command removes user group permanently.

**delgroup**  [--help] [--group=*groupname*] [--newgroup=*groupname*]

When run without any arguments this program runs in interactive mode prompting for group to be deleted. Groups can also be removed non-interactively by specifying `--group` option on command-line. Normally users that were in the deleted group are left without a group. If `--newgroup` is specified then any users that were in the deleted group are assigned to this group.

## 6.6. deluser

This command removes user account permanently. Account can also be locked using **moduser** program (see Section 6.18).

**deluser**  [--help] [--user=*username*]

When run without any arguments this program runs in interactive mode prompting for user to be deleted. Users can also be removed non-interactively by specifying --user option on command-line.

# 6.7. expire-hosts

This is a utility for expiring unused (hosts that haven't been issued a DHCP lease within given time-frame) host entries.

**expire-hosts** [--help] {*server*} [*OPTIONS*]

| option | description |
|---|---|
| --cidr=*CIDR* | Select only hosts within this CIDR block. |
| --name=*regexp* | Select only hostnames matching the regular expression. |
| --netname=*regexp* | Select only hosts in the networks matching the regular expression (selects only hosts within auto assign ranges in these networks). |
| --treshold=*days* | Expire hosts with no DHCP activity in last <days> days (default: 90 days). |
| --norecord | Expire only hosts wit no recorded DHCP activity. |
| --edate=*dd-mm-yyyy* | Set hosts to expire on this date (default: now). |
| --verbose | Make program to display more verbose output. |
| --commit | Commit changes (without this no changes are made into database). |

By default this program will not make any changes to Sauron database, it will only display what would be made. Use --commit to make changes into Sauron database.

# 6.8. export-networks

This utility generates *nix /etc/networks format file of networks defined for given server.

**export-networks** [--help] [--verbose] [--comments] [--all] {*servername*}

This program sends generated file into standard output. Normally only networks that have DHCP setting set to true are included in the listing. To force listing of all networks defined for this server use --all option. Network names from Sauron are not included as comments into output unless --comments option is used.

# 6.9. generatehosts

This is a utility for generating number of similar host entries into Sauron database with one command.

**generatehosts** [--help] {*server*} {*zone*} {*hostname*} {*starting-IP*} {*count*} [*OPTIONS*]

| option | description |
|---|---|
| `--noreverse` | Do not add reverse records. |
| `--outofzone` | Allow out of zone records. |
| `--verbose` | Produce more verbose output. |
| `--seqstart=`*n* | Start sequence numbering from n. |
| `--group=`*name* | Assign new hosts to this group. |
| `--info=`*user:dept:location:info* | Set info fields for all hosts to these values. |
| `--commit` | Commit changes (without this no changes are made into database). |

By default this program will not make any changes to Sauron database, it will only display what would be made. Use `--commit` to make changes into Sauron database.

Hostname argument serves as a template for hostname of generated hosts. Following tags can be used in `hostname`:

| :N: | current sequence number |
|---|---|
| :IP1: | first octet of current IP-number |
| :IP2: | second octet of current IP-number |
| :IP3: | third octet of current IP-number |
| :IP4: | fourth octet of current IP-number |

# 6.10. import

Utility for importing complete BIND configuration file set into Sauron. This program parses BIND configuration files and generates new server and related zones into Sauron. See also Section 3.2.2.

**import** [--help] [*OPTIONS*] {*servername*} {*named.conf file*}

| option | description |
|---|---|
| `--orphans=`*zonename* | Zone for 'orphan' PTR record hosts. |
| `--notransaction` | Do not use transactions when updating database (potentially dangerous, but speeds up things). |
| `--dir=`*directory* | Directory where configuration files are located (if not in the directory specified in named.conf). |

| option | description |
|---|---|
| --verbose | Make program to display more verbose output. |

By default **import** expects to find zone-files in the locations specified in the `named.conf`, but `--dir` option can be used to explicitly specify location for zone-files. After importing server from existing BIND configuration, related DHCP configuration should be imported using **import-dhcp** command.

## 6.11. import-dhcp

This utility can be used after importing existing DNS configuration using **import** utility to add information (Ethernet addresses and network map) from DHCP configuration into a server.

**import-dhcp** [--help] [*OPTIONS*] {*servername*} {*dhcpd.conf file*}

| option | description |
|---|---|
| --dir=*directory* | Directory where included files are located (if not in the directory specified in dhcpd.conf). |
| --chaosnet=*name* | Treat this shared-network as default VLAN (default is "CHAOS"). |
| --global | Import "global" entries into server DHCP settings. |
| --verbose | Make program to display more verbose output. |

When using this program after importing existing DNS configuration using **import**, `--global` should be used to import global DHCP settings into the server DHCP settings.

## 6.12. import-ethers

Utility for assigning OUI (Ethernet card manufacturer) information into global table `ether_info`. This table enables Sauron to recognize and display Network card (NIC) manufacturer information.

**import-ethers** [--help] [--force] {*file*}

This program currently understands OUI information in old `Ethernet.txt` format and in the IEEE's OUI list format (`oui.txt`). Normally this utility only adds new OUIs into database, but using `--force` option allows updating all OUIs.

## 6.13. import-roots

Utility for importing root server hints into Sauron. Sauron has global table for root servers. But it is also possible to import server specific root servers which override global root servers table.

**import-roots** [--help] [--update] [*servername* | default] {*file*}

This program imports standard root server (hint) zone-file. To import global root servers use "default" as the server name.

## 6.14. import-zone

This a utility for importing DNS zones into existing server. Zones can be read either from a zone file or fetched directly from DNS using zone transfer.

**import-zone** [--help] [*OPTIONS*] {*servername*} {*zone*} [*zone file*]

| option | description |
|---|---|
| --nameserver=*server* | Use this nameserver for zone transder (argument must be either IP number or domainname). |
| --zone=*name* | Insert hosts into this parent zone, instead of generating new zone. |
| --group=*name* | Assign new hosts into this group. |
| --verbose | Make program to display more verbose output. |

Normally **import-zone** command generates a new zone in Sauron, but with --zone option it is possible to import only the hosts from a zone into that zone's parent zone (which must already exist in Sauron).

## 6.15. last

Utility for listing "lastlog" information about Sauron WWW-interface users. This command is similar to *nix **last** command.

**last** [-h] [-*num*] [*username*]

Option -*num* can be used to limit how many entries to display. To list only lastlog entries of a specific user enter that user name as the last argument.

## 6.16. modgroup

This is utility for modifying user groups. This utility can also be used to list all user groups. User groups can be assigned all the same privileges as a user, but assigning privileges to groups and then users make privilege management more manageable with large number of users. See also Section 4.1.2.

**modgroup** [--help] {*groupname*} [*OPTIONS*]

| option | description |
|---|---|
| --list | List all groups and exit. |

| option | description |
|---|---|
| --add | Add interactively privileges to given group. |
| --add=server,*servername*,*mode* | Add server privilege (mode=R\|RW\|RWX). |
| --add=zone,*servername:zonename*,*mode* | Add zone privilege (mode=R\|RW\|RWX). |
| --add=net,*servername:net* | Add net (IP-range) restriction. |
| --add=host,*regexp* | Add hostname mask (restriction). |
| --add=ipmask,*ipmask* | Add IP-mask restriction. |
| --add=level,*level* | Add privilege level. |
| --add=elimit,*days* | Add host expiration limit. |
| --add=def_dept,*string* | Add default department for adding hosts. |
| --del=*id* | Delete privilege by it's ID number. |
| --delall | Delete all privileges. |

This program can be used to list all available groups by specifying only --list. Privileges can be added either interactively by specifying short form of --add or non-interactively by using long form.

## 6.17. modhosts

Simple utility for modifying or deleting several hosts at a time.

**modhosts** [--help] {*server*} {*zone*} [*OPTIONS*]

| option | description |
|---|---|
| --cidr=*CIDR* | Select only hosts within this CIDR block. |
| --name=*regexp* | Select only hostnames matching the regular expression. |
| --info=*regexp* | Select only hosts with matching info (this looks for pattern in User, Dept, Location, and Info fields). |
| --type=*type* | Select only hosts of this type (available types: host, cname, arc, srv). |
| --ether=*regexp* | Select only hosts with matching Ethernet (MAC) address. |
| --etherempty | Select only hosts without Ethernet (MAC) address. |
| --delete | Delete matching hosts. |
| --move=*CIDR*,*IP* | Move host into given network (starting from IP within that network). |
| --rename=*regexp* | Rename hosts using given "substitution" regular expression. |
| --setedate=*days* | Set host expiration dates to: today+*days*. |
| --setgroup=*groupname* | Assign hosts to this group. |

| option | description |
|--------|-------------|
| --commit | Commit changes (without this no changes are made to database). |

By default this program will not make any changes to Sauron database, it will only display what would be made. Use --commit to make changes into Sauron database.

## 6.18. moduser

This is utility for modifying user accounts. This utility can also be used to list all users. This utility can also be used to modify user account privileges, but it's recommended to use user groups rather than assigning privileges directly to user accounts. For more information about user privileges see Section 4.1.2.

**moduser** [--help] {*username*} [*OPTIONS*]

| option | description |
|--------|-------------|
| --name="*user's name*" | Set user's full name. |
| --email=*address* | Set user's email address. |
| --comment=*text* | Set user comments. |
| --expiration=*dd-mm-yyyy* \| NONE | Set account expiration date. |
| --superuser=yes \| no | Toggle superuser flag for user. |
| --passwd | Set user password interactively. |
| --passwd=*password* | Set user password. |
| --group=*groupname* \| NONE | Set user's group. |
| --list | List all users and exit. |
| --lock | Lock user account. |
| --unlock | Unlock user account. |
| --add | Add interactively privileges to given group. |
| --add=server,*servername,mode* | Add server privilege (mode=R\|RW\|RWX). |
| --add=zone,*servername:zonename,mode* | Add zone privilege (mode=R\|RW\|RWX). |
| --add=net,*servername:net* | Add net (IP-range) restriction. |
| --add=host,*regexp* | Add hostname mask (restriction). |
| --add=ipmask,*ipmask* | Add IP-mask restriction. |
| --add=level,*level* | Add privilege level. |
| --add=elimit,*days* | Add host expiration limit. |
| --add=def_dept,*string* | Add default department for adding hosts. |
| --del=*id* | Delete privilege by it's ID number. |
| --delall | Delete all privileges. |

This program can be used to list all users by specifying only --list. Privileges can be added either

interactively by specifying short form of `--add` or non-interactively by using the long form.

# 6.19. runsql

Utility for sunning SQL code from a file into Sauron database. This utility is used by **createtables** to initialize Sauron database.

**runsql** [--help] [--separate] `filename`...

Normally all files given as arguments are processed inside on transaction. To process each file as a separate transaction use `--separate` option.

# 6.20. sauron

This is the program for generating DNS and DHCP configuration files from Sauron database for given server.

**sauron** [--help] [`OPTIONS`] {`servername`} [`target directory`]

| option | description |
|---|---|
| `--all` | Generate all (DNS & DHCP) configuration files. |
| `--bind` | Generate BIND (named) configuration files. |
| `--dhcp` | Generate DHCP (dhcpd) configuration files. |
| `--printer` | Generate PRINTER (lpd) configuration files (not yet implemented). |
| `--clean` | Cleanup expired host records from the server. By default host records that have been expired over 30 days are removed, but this can be controlled using configuration file option `SAURON_REMOVE_EXPIRED_DELAY`. |
| `--updateserial` | force SOA serial number update on master zones. |
| `--check` | Check validity of generated DNS & DHCP configuration files. (see Section 3.3). |
| `--dhcp2` | Enable DHCP v2 compatibility mode. |
| `--mail` | Enable email notification sending. |

Option `--updateserial` forces the update of zone SOA serial numbers for master zones. Normally Sauron tries to detect if zone serial number update is necessary and update it only when needed, but this detection routine may sometimes fail, so for now it's recommended to use `--updateserial` option.

# 6.21. status

Utility for listing database connection status and the WWW user interface status information including currently logged in users.

**status**  [--help] [*OPTIONS*]

| option | description |
|---|---|
| --pending | Display number of host changes pending for each server. |
| --pending=*servername* | Display pending host changes for given server. |
| --cgi-disable=*message* | Disable WWW interface and set message to be displayed when users try to access the WWW interface. |
| --cgi-enable | Enable WWW interface. |
| --quiet | Display less verbose output. |
| --versions | Display module version numbers. |

This command can be easily used from shell scripts to test for pending changes for a server by using --pending=*servername* (and optionally --quiet option) which returns exit code 2 if there are pending hosts (0=no pending hosts, 1=error).

# 6.22. update-dhcp-info

Utility designed to be run from cron that parses logs generated by **dhcpd** and updates "Last lease issued by DHCP server" fields for hosts.

**update-dhcp-info**  [--help] [--verbose] [--year=*yyyy*] {*servername*} *logfile*

Since normal syslog logfiles have timestamps without year information, option --year can be used to specify the year to be used if parsing logfiles from previous year. Normally **update-dhcp-info** assumes that the logfile is from current year.

# Appendix A. Technical Details

This appendix describes the Sauron database layout. Sauron database contains database version number stored in `settings` table. This database version number must match with the version Sauron "back end" was made for. If database format changes in future versions, the database version number will be incremented and a conversion script from previous version will be shipped with Sauron.

Current Sauron Database version is 1.0

**Figure A-1. Table Relations**

Sauron: Database Table Relationships
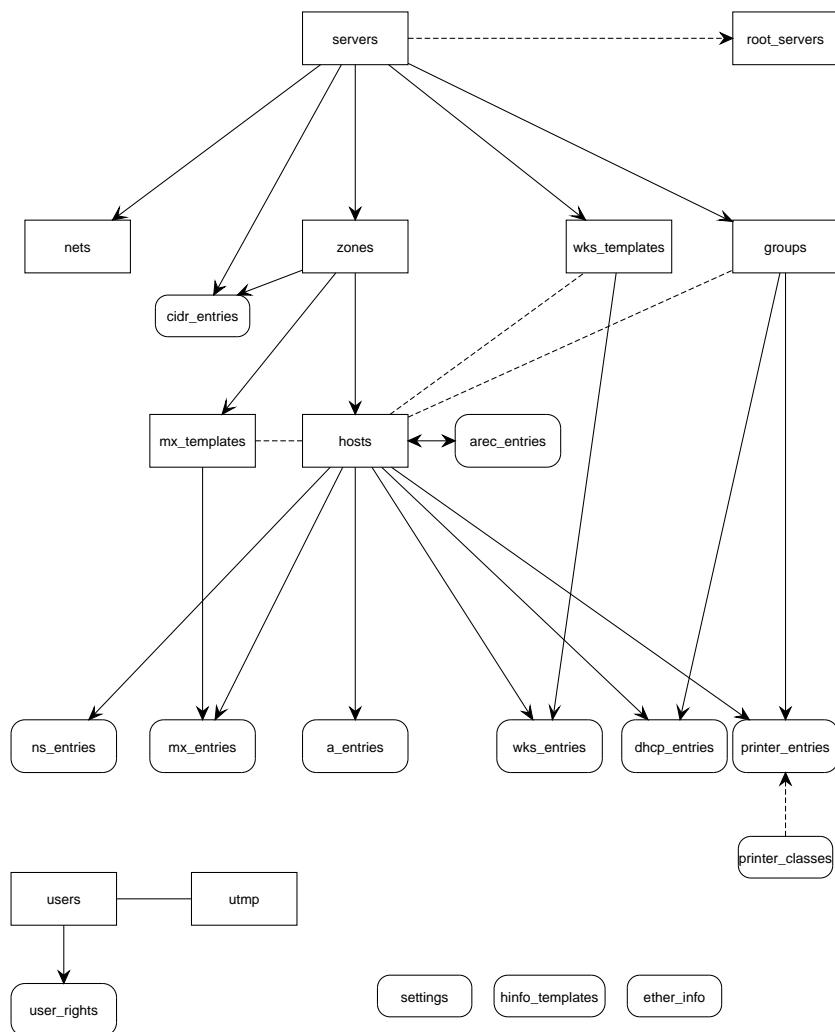


$Id: relations.gd,v 1.2 2001/03/01 21:56:59 tjko Exp $

# A.1. Sauron: SQL table descriptions

Tables:

- common_fields
- settings
- servers
  - zones(zones.server -> servers.id)
    - hosts(hosts.zone -> zones.id)
      - dhcp_entries(dhcp_entries.ref -> hosts.id)
      - mx_entries(mx_entries.ref -> hosts.id)
      - ns_entries(ns_entries.ref -> hosts.id)
      - printer_entries(printer_entries.ref -> hosts.id)
      - a_entries(a_entries.host -> hosts.id)
      - txt_entries(txt_entries.ref -> hosts.id)
      - srv_entries(srv_entries.ref -> hosts.id)
      - wks_entries(wks_entries.ref -> hosts.id)
      - arec_entries(arec_entries.host -> hosts.id)
      - arec_entries(arec_entries.arec -> hosts.id)

    - cidr_entries(cidr_entries.ref -> zones.id)
    - dhcp_entries(dhcp_entries.ref -> zones.id)
    - mx_entries(mx_entries.ref -> zones.id)
    - mx_templates(mx_templates.zone -> zones.id)
      - hosts(hosts.mx -> mx_templates.id)

    - ns_entries(ns_entries.ref -> zones.id)
    - txt_entries(txt_entries.ref -> zones.id)

  - groups(groups.server -> servers.id)
    - hosts(hosts.grp -> groups.id)

- dhcp_entries(dhcp_entries.ref -> groups.id)

- printer_entries(printer_entries.ref -> groups.id)

- nets(nets.server -> servers.id)

  - dhcp_entries(dhcp_entries.ref -> nets.id)

- cidr_entries(cidr_entries.ref -> servers.id)

- dhcp_entries(dhcp_entries.ref -> servers.id)

- txt_entries(txt_entries.ref -> servers.id)

- wks_templates(wks_templates.server -> servers.id)

  - hosts(hosts.wks -> wks_templates.id)

  - wks_entries(wks_entries.ref -> wks_templates.id)

- root_servers(root_servers.server -> servers.id)

- vlans(vlans.server -> servers.id)

  - nets(nets.vlan -> vlans.id)

- ether_info

- printer_classes

  - printer_entries(printer_entries.ref -> printer_classes.id)

- users

  - user_rights(user_rights.ref -> users.id)

  - utmp(utmp.uid -> users.id)

- user_groups

  - users(users.gid -> user_groups.id)

  - user_rights(user_rights.ref -> user_groups.id)

  - utmp(utmp.gid -> user_groups.id)

- hinfo_templates

- history

- lastlog

- news

## A.1.1. Table Descriptions

### A.1.1.1. common_fields

virtual table; generic fields for most of the tables

| field | type | comments |
|---|---|---|
| cdate | INT4 | creation date |
| cuser | CHAR(8) DEFAULT 'unknown' | creating user |
| mdate | INT4 | modification date |
| muser | CHAR(8) DEFAULT 'unknown' | last changed by this user |
| expiration | INT4 | expiration date |

### A.1.1.2. settings

global settings table

| field | type | comments |
|---|---|---|
| key | TEXT NOT NULL CHECK(key <> '') | name os setting tuple |
| value | TEXT | string value of setting |
| ivalue | INT4 | interger value of setting |
| CONSTRAINT | global_key PRIMARY KEY (key) | |

### A.1.1.3. servers

This table contains servers that are managed with this system. For each server named/dhcpd/printer configuration files can be automagically generated from the database.

| field | type | comments |
|---|---|---|
| id | SERIAL PRIMARY KEY | unique ID |
| name | TEXT NOT NULL CHECK(name <> '') | server name |
| zones_only | BOOL DEFAULT false | if true, generate named.zones file otherwise generate complete named.conf |
| no_roots | BOOL DEFAULT false | if true, no root server (hint)zone entry is generated |

| field | type | comments |
|---|---|---|
| dhcp_mode | INT DEFAULT 1 | DHCP subnet map creation mode: 0 = use VLANs,1 = use networks |
| dhcp_flags | INT DEFAULT 0 | DHCP option flags: 0x01 = auto-generate domainnames0x02 = enable failover protocol |
| named_flags | INT DEFAULT 0 | named option flags: 0x01 = access control from master (slave only) 0x02 = include also slave zones from master (slave only) 0x04 = do NOT generate HINFO records0x08 = do NOT generate WKS records |
| masterserver | INT DEFAULT -1 | dynamically add slave zonesfor all zones in master server |
|  |  | named.conf options...more to be added as needed... |
| version | TEXT | version string to display (optional) |
| directory | TEXT | base directory for named (optional) |
| pid_file | TEXT | pid-file pathname (optional) |
| dump_file | TEXT | dump-file pathname (optiona) |
| named_xfer | TEXT | named-xfer pathname (optional) |
| stats_file | TEXT | statistics-file pathname (optional) |
| memstats_file | TEXT | memstatistics-file pathname (optional) |
| named_ca | TEXT DEFAULT 'named.ca' | root servers filename |
| pzone_path | TEXT DEFAULT '' | relative path for masterzone files |
| szone_path | TEXT DEFAULT 'NS2/' | relative path for slave zone files |
| query_src_ip | TEXT | query source ip (optional) (ip \| '*') |
| query_src_port | TEXT | query source port (optional) (port '*') |
| listen_on_port | TEXT | listen on port (optional) |
| transfer_source | INET | transfer-source (optional) |
| forward | CHAR(1) DEFAULT 'D' | forward: D=defaultO=only, F=first |
|  |  | check-names: D=default, W=warn, F=fail, I=ignore |
| checknames_m | CHAR(1) DEFAULT 'D' | check-names master |
| checknames_s | CHAR(1) DEFAULT 'D' | check-names slave |

| field | type | comments |
|---|---|---|
| checknames_r | CHAR(1) DEFAULT 'D' | check-names response |
| | | boolean flags: D=default, Y=yes, N=no |
| nnotify | CHAR(1) DEFAULT 'D' | notify |
| recursion | CHAR(1) DEFAULT 'D' | recursion |
| authnxdomain | CHAR(1) DEFAULT 'D' | auth-nxdomain |
| dialup | CHAR(1) DEFAULT 'D' | dialup |
| multiple_cnames | CHAR(1) DEFAULT 'D' | multiple-cnames |
| rfc2308_type1 | CHAR(1) DEFAULT 'D' | rfc2308-type1 |
| | | default TTLs |
| ttl | INT4 DEFAULT 86400 | default TTL for RR records |
| refresh | INT4 DEFAULT 43200 | default SOA refresh |
| retry | INT4 DEFAULT 3600 | default SOA retry |
| expire | INT4 DEFAULT 2419200 | default SOA expire |
| minimum | INT4 DEFAULT 86400 | default SOA minimum (negative caching ttl) |
| | | IPv6 |
| ipv6 | TEXT | reserved |
| | | DHCP failover |
| df_port | INT DEFAULT 519 | listen port |
| df_max_delay | INT DEFAULT 60 | max-response-delay |
| df_max_uupdates | INT DEFAULT 10 | max-unacked-updates |
| df_mclt | INT DEFAULT 3600 | mlct |
| df_split | INT DEFAULT 128 | split |
| df_loadbalmax | INT DEFAULT 3 | load balance max seconds |
| | | defaults to use in zones |
| hostname | TEXT | primary servername for sibling zone SOAs |
| hostaddr | INET | primary server IP address |
| hostmaster | TEXT | hostmaster name for sibling zone SOAsunless overided in zone |
| comment | TEXT | |
| CONSTRAINT | servers_name_key UNIQUE(name) | |

## A.1.1.4. zones

This table contains zone definitions of a server.

| field | type | comments |
|---|---|---|
| id | SERIAL | unique ID |

| field | type | comments |
|---|---|---|
| server | INT4 NOT NULL | ptr to a record in servers table-->servers.id |
| active | BOOL DEFAULT true | zone active flag (only active zones are included innamed configuration) |
| dummy | BOOL DEFAULT false | dummy zone flag |
| type | CHAR(1) NOT NULL | zone type: (H)int, (M)aster, (S)lave, (F)orward |
| reverse | BOOL DEFAULT false | true for reverse (arpa) zones |
| noreverse | BOOL DEFAULT false | if true, zone not used in reversemap generation |
| flags | INT DEFAULT 0 | zone option flags: 0x01 = generate TXT records fromuser,dept,location,info fields |
| forward | CHAR(1) DEFAULT 'D' | forward: D=default, O=only, F=first |
| nnotify | CHAR(1) DEFAULT 'D' | notify: D=default, Y=yes, N=no |
| chknames | CHAR(1) DEFAULT 'D' | check-names: D=default, W=warn, F=fail,I=ignore |
| class | CHAR(2) DEFAULT 'in' | zone class (IN) |
| name | TEXT NOT NULL CHECK (name <> '') | zone name |
| hostmaster | TEXT | hostmaster (email) (optional; if not defined value from server tableis used instead) |
| serial | CHAR(10) DEFAULT '1999123001' | zone serial number(automagically updated) |
| serial_date | INT4 DEFAULT 0 | zone serial last update date |
| refresh | INT4 | zone SOA refresh time |
| retry | INT4 | zone SOA retry time |
| expire | INT4 | zone SOA expire time |
| minimum | INT4 | zone SOA minimum (negative caching) time |
| ttl | INT4 | default TTL for RRs in this zone (if not defined, value from servers record isused instead) |
| zone_ttl | INT4 | unused |
| comment | TEXT | |
| reversenet | CIDR | contains CIDR of the reverse zone(if applicaple) |
| parent | INT4 DEFAULT -1 | unused |
| CONSTRAINT | zones_key PRIMARY KEY (name,server) | |

## A.1.1.5. hosts

This table contains host entries for a zone.

| field | type | comments |
|---|---|---|
| id | SERIAL PRIMARY KEY | unique ID |
| zone | INT4 NOT NULL | ptr to a zone table record-->zones.id |
| type | INT4 DEFAULT 0 | host type: 0=misc, 1=host, 2=subdomain (delegation), 3=mx entry, 4=alias (cname), 5=printer, 6=glue record, 7=alias (arec), 8=srv entry, 9=dhcp only, 10=zone, 101=host reservation |
| domain | TEXT NOT NULL CHECK(domain <> ") | host domain name |
| ttl | INT4 | TTL for host records, default if NULL |
| class | CHAR(2) DEFAULT 'IN' | class (IN) |
| grp | INT4 DEFAULT -1 | ptr to group-->groups.id |
| alias | INT4 DEFAULT -1 | ptr to another host record(for CNAME alias) |
| cname_txt | TEXT | CNAME value for out-of-zone alias |
| hinfo_hw | TEXT | HINFO hardware |
| hinfo_sw | TEXT | HINFO software |
| loc | TEXT | LOC record value |
| wks | INT4 DEFAULT -1 | ptr to wks_templates table entry-->wks_templates.id |
| mx | INT4 DEFAULT -1 | ptr to mx_templates table entry-->mx_templates.id |
| rp_mbox | TEXT DEFAULT '.' | RP mbox |
| rp_txt | TEXT DEFAULT '.' | RP txt |
| router | INT4 DEFAULT 0 | router if > 0, also router priority(1 being highest priority) |
| prn | BOOL DEFAULT false | true for virtual printer entries |
| flags | INT4 DEFAULT 0 | reserved |
| ether | CHAR(12) | Ethernet address (MAC) |
| ether_alias | INT4 DEFAULT -1 | ptr to another host record(for ETHER address) |
| dhcp_date | INT4 | last time host requested IP |
| dhcp_info | TEXT | reserved |

| field | type | comments |
|---|---|---|
| info | TEXT | Host info (appears as TXT record) |
| location | TEXT | Host location info |
| dept | TEXT | Department name |
| huser | TEXT | User info |
| model | TEXT | host model info |
| serial | TEXT | serial number |
| misc | TEXT | misc info |
| asset_id | TEXT | asset ID |
| comment | TEXT | comment |
| CONSTRAINT | hostname_key UNIQUE (domain,zone), | |
| CONSTRAINT | ether_key UNIQUE(ether,zone), | |
| CONSTRAINT | asset_key UNIQUE(asset_id,zone) | |

## A.1.1.6. groups

Group descriptions, linked to server record. Hosts can "belong" to one group and get DHCP/printer/etc definitions from that group.

| field | type | comments |
|---|---|---|
| id | SERIAL PRIMARY KEY | unique ID |
| server | INT4 NOT NULL | ptr to a servers table record-->servers.id |
| name | TEXT NOT NULL CHECK(name <> '') | group name |
| type | INT NOT NULL | group type: 1 = normal group, 2 = dynamic address pool,3 = DHCP client class |
| alevel | INT4 DEFAULT 0 | required authorization level |
| comment | TEXT | |
| CONSTRAINT | groups_key UNIQUE(name,server) | |

## A.1.1.7. nets

Net/subnet descriptions, linked to server record. Used mainly for generating subnet map for DHCP and access control/user friendliness in front-ends.

| field | type | comments |
|---|---|---|
| id | SERIAL | unique ID |

| field | type | comments |
|---|---|---|
| server | INT4 NOT NULL | ptr to a servers table record-->servers.id |
| netname | TEXT | (sub)net name |
| name | TEXT | descriptive name of the (sub)net |
| net | CIDR NOT NULL | net CIDR |
| subnet | BOOL DEFAULT true | subnet flag |
| vlan | INT4 DEFAULT -1 | ptr to vlans table record-->vlans.id |
| alevel | INT4 DEFAULT 0 | required authorization level |
| type | INT4 DEFAULT 0 | network type/option flags:0x01 = private (hidden from browser) |
| ipv6 | TEXT | reserved |
| rp_mbox | TEXT DEFAULT '.' | RP mbox |
| rp_txt | TEXT DEFAULT '.' | RP txt |
| no_dhcp | BOOL DEFAULT false | no-DHCP flag |
| range_start | INET | auto assign address range start |
| range_end | INET | auto assign address range end |
| comment | TEXT | comment |
| CONSTRAINT | nets_key PRIMARY KEY (net,server) | |

## A.1.1.8. cidr_entries

This table contains CIDRs used in server various contexts.

| field | type | comments |
|---|---|---|
| id | SERIAL PRIMARY KEY | unique ID |
| type | INT4 NOT NULL | type: 1=server (allow-transfer) 2=zone (allow-update) 3=zone (masters) 4=zone (allow-query) 5=zone (allow-transfer) 6=zone (also-notify) 7=server (allow-query) 8=server (allow-recursion) 9=server (blackhole) 10=server (listen-on) 11=server (forwarders)12=zone (forwarders) |
| ref | INT4 NOT NULL | ptr to table speciefied by type field -->servers.id-->zones.id |
| ip | CIDR | CIDR value |
| comment | TEXT | |

### A.1.1.9. dhcp_entries

This table contains DHCP options user in various contexts.

| field | type | comments |
|---|---|---|
| id | SERIAL PRIMARY KEY | unique ID |
| type | INT4 NOT NULL | type: 1=server, 2=zone, 3=host, 4=net, 5=group6=vlan (shared-network) |
| ref | INT4 NOT NULL | ptr to table speciefied by type field -->servers.id -->zones.id -->hosts.id -->nets.id-->groups.id |
| dhcp | TEXT | DHCP entry value (without trailing ';') |
| comment | TEXT | |

### A.1.1.10. ether_info

This table contains Ethernet adapter manufacturer codes.

| field | type | comments |
|---|---|---|
| ea | CHAR(6) PRIMARY KEY | manufacturer code (6 bytes in hex) |
| info | TEXT | manufacturer name & info |

### A.1.1.11. mx_entries

This table contains MX record entries.

| field | type | comments |
|---|---|---|
| id | SERIAL PRIMARY KEY | unique ID |
| type | INT4 NOT NULL | type: 1=zone (not used anymore!), 2=host,3=mx_templates |
| ref | INT4 NOT NULL | ptr to table speciefied by type field -->zones.id -->hosts.id-->mx_templates |
| pri | INT4 NOT NULL CHECK (pri >= 0) | MX priority |
| mx | TEXT | MX domain (FQDN) |
| comment | TEXT | |

### A.1.1.12. mx_templates

MX entry templates, hosts may link to one entry in this table. Entries are zone specific.

| field | type | comments |
|---|---|---|
| id | SERIAL PRIMARY KEY | unique ID |
| zone | INT4 NOT NULL | ptr to a zone table record-->zones.id |
| alevel | INT4 DEFAULT 0 | required authorization level |
| name | TEXT | template name |
| comment | TEXT | |

### A.1.1.13. ns_entries

This table contains NS resource record definitions.

| field | type | comments |
|---|---|---|
| id | SERIAL PRIMARY KEY | unique ID |
| type | INT4 NOT NULL | type: 1=zone (not used anymore!),2=host |
| ref | INT4 NOT NULL | ptr to table speciefied by type field -->zones.id-->hosts.id |
| ns | TEXT | value of NS record (FQDN) |
| comment | TEXT | |

### A.1.1.14. printer_classes

Global table to store printer classes (printcap stuff) these classess maybe referred to in PRINTER fields in other tables.

| field | type | comments |
|---|---|---|
| id | SERIAL PRIMARY KEY | unique ID |
| name | TEXT UNIQUE NOT NULL CHECK(name <> '') | class name |
| comment | TEXT | |

### A.1.1.15. printer_entries

This table contains printer definition entries.

| field | type | comments |
|---|---|---|
| id | SERIAL PRIMARY KEY | unique ID |

| field | type | comments |
|---|---|---|
| type | INT4 NOT NULL | type: 1=group, 2=host,3=printer_class |
| ref | INT4 NOT NULL | ptr to table speciefied by type field -->groups.id -->hosts.id-->printer_classes.id |
| printer | TEXT | printcap entry |
| comment | TEXT | |

## A.1.1.16. a_entries

Addresses (A records) for hosts, linked to a host record.

| field | type | comments |
|---|---|---|
| id | SERIAL PRIMARY KEY | unique ID |
| host | INT4 NOT NULL | ptr to hosts table id-->hosts.id |
| ip | INET | IP number |
| ipv6 | TEXT | reserved |
| type | INT4 DEFAULT 0 | reserved |
| reverse | BOOL DEFAULT true | generate reverse (PTR) record flag |
| forward | BOOL DEFAULT true | generate (A) record flag |
| comment | CHAR(20) | |

## A.1.1.17. txt_entries

This table contains TXT record entries and miscellaneous text entries.

| field | type | comments |
|---|---|---|
| id | SERIAL PRIMARY KEY | unique ID |
| type | INT4 NOT NULL | type: 1=zone (not used anymore!), 2=host, 3=server10=server (BIND logging entry) |
| ref | INT4 NOT NULL | ptr to table speciefied by type field -->zones.id -->hosts.id-->servers.id |
| txt | TEXT | value of TXT record |
| comment | TEXT | comments |

## A.1.1.18. srv_entries

This table contains MX record entries.

| field | type | comments |
|---|---|---|
| id | SERIAL PRIMARY KEY | unique ID |
| type | INT4 NOT NULL | type:1=host |
| ref | INT4 NOT NULL | ptr to table speciefied by type field-->hosts.id |
| pri | INT4 NOT NULL CHECK (pri >= 0) | priority |
| weight | INT4 NOT NULL CHECK (weight >= 0) | weight |
| port | INT4 NOT NULL CHECK (port >= 0) | port |
| target | TEXT NOT NULL DEFAULT '.' | target |
| comment | TEXT | comment |

## A.1.1.19. users

This table contains (user interface) user account information.

| field | type | comments |
|---|---|---|
| id | SERIAL PRIMARY KEY | unique ID |
| gid | INT4 DEFAULT -1 | ptr to user group -->user_groups.id |
| username | TEXT NOT NULL CHECK(username <> '') | login name |
| password | TEXT | encrypted password (MD5 or Crypt) |
| name | TEXT | long user name |
| email | TEXT | user email address |
| superuser | BOOL DEFAULT false | superuser flag |
| server | INT4 DEFAULT -1 | default server id |
| zone | INT4 DEFAULT -1 | default zone id |
| last | INT4 DEFAULT 0 | last login time |
| last_pwd | INT4 DEFAULT 0 | last password change time |
| last_from | TEXT | last login host |
| search_opts | TEXT | default search options |
| flags | INT4 DEFAULT 0 | user account flasgs:0x01 = email notifications on |
| comment | TEXT | |

| field | type | comments |
|-------|------|----------|
| CONSTRAINT | username_key UNIQUE(username) | |

### A.1.1.20. user_rights

This table contains record defining user rights.

| field | type | comments |
|-------|------|----------|
| id | SERIAL PRIMARY KEY | unique ID |
| type | INT NOT NULL | type: 1=user_group2=users |
| ref | INT NOT NULL | ptr to users table specified by type -->user_groups.id-->users.id |
| rtype | INT NOT NULL | type: 1=server, 2=zone, 3=net, 4=hostnamemask 5=IP mask 6=authorization level 7=host expiration limit (days)8=default for dept |
| rref | INT NOT NULL | ptr to table specified by type field |
| rule | CHAR(40) | R,RW,RWS or regexp |

### A.1.1.21. user_groups

This table contains records defining user groups.

| field | type | comments |
|-------|------|----------|
| id | SERIAL PRIMARY KEY | unique ID |
| name | TEXT NOT NULL CHECK (name <> '') | group name |
| comment | TEXT | comments |
| CONSTRAINT | user_groups_name_key UNIQUE(name) | |

### A.1.1.22. wks_entries

This table contains WKS record entries.

| field | type | comments |
|-------|------|----------|
| id | SERIAL PRIMARY KEY | unique ID |
| type | INT4 NOT NULL | type: 1=host,2=wks_template |
| ref | INT4 NOT NULL | ptr to table speciefied by type field -->hosts.id-->wks_templates.id |
| proto | CHAR(10) | protocol (tcp,udp) |

| field | type | comments |
|---|---|---|
| services | TEXT | services (ftp,telnet,smtp,http,...) |
| comment | TEXT | |

### A.1.1.23. wks_templates

WKS entry templates, hosts may link to one entry in this table. Entries are server specific.

| field | type | comments |
|---|---|---|
| id | SERIAL PRIMARY KEY | unique ID |
| server | INT4 NOT NULL | ptr to a server table record-->servers.id |
| alevel | INT4 DEFAULT 0 | required authorization level |
| name | TEXT | template name |
| comment | TEXT | |

### A.1.1.24. utmp

This table contains "utmp" data of currently logged in www-interface users.

| field | type | comments |
|---|---|---|
| cookie | CHAR(32) PRIMARY KEY | session id cookie (MD5) |
| uid | INT4 | ptr to users table record-->users.id |
| gid | INT4 | ptr to user_groups table record-->user_groups.id |
| sid | INT4 | session ID |
| uname | TEXT | username |
| addr | CIDR | user's IP address |
| superuser | BOOL DEFAULT false | superuser flag |
| auth | BOOL DEFAULT false | user authenticated flag |
| mode | INT4 | current status of user |
| w | TEXT | last command user excecuted |
| serverid | INT4 DEFAULT -1 | current server id |
| server | TEXT | current server name |
| zoneid | INT4 DEFAULT -1 | current zone id |
| zone | TEXT | current zone name |
| login | INT4 DEFAULT 0 | login time |
| last | INT4 DEFAULT 0 | last activity time |
| searchopts | TEXT | current search options |

| field | type | comments |
|---|---|---|
| searchdomain | TEXT | current search domain |
| searchpattern | TEXT | current search pattern |

### A.1.1.25. hinfo_templates

HINFO templates table contains list of default values for HINFO records.

| field | type | comments |
|---|---|---|
| id | SERIAL PRIMARY KEY | unique ID |
| hinfo | TEXT NOT NULL CHECK(hinfo <> ") UNIQUE | HINFO value |
| type | INT4 DEFAULT 0 | type: 0=hardware, 1=software |
| pri | INT4 DEFAULT 100 | priority (defines the order in whichentries are displayed in user interfaces) |

### A.1.1.26. arec_entries

pointers to A record aliased hosts, linked to a host record.

| field | type | comments |
|---|---|---|
| id | SERIAL PRIMARY KEY | unique ID |
| host | INT4 NOT NULL | ptr to hosts table id-->hosts.id |
| arec | INT4 NOT NULL | ptr to aliased host id -->hosts.id |

### A.1.1.27. root_servers

This table contains root server definitions.

| field | type | comments |
|---|---|---|
| id | SERIAL PRIMARY KEY | unique ID |
| server | INT4 NOT NULL | ptr to server table id-->servers.id |
| ttl | INT4 DEFAULT 3600000 | |
| domain | TEXT NOT NULL | domainname |
| type | TEXT NOT NULL | A,NS,... |
| value | TEXT NOT NULL | value |

### A.1.1.28. history

history table contains "log" data of modifications done to the databse

| field | type | comments |
| --- | --- | --- |
| id | SERIAL PRIMARY KEY | unique ID |
| sid | INT NOT NULL | session ID |
| uid | INT NOT NULL | user ID |
| date | INT NOT NULL | date of record |
| type | INT NOT NULL | record type: 1=hosts table modification, 2=zones 3=servers 4=nets5=users |
| ref | INT | optional reference |
| action | CHAR(25) | operation performed |
| info | CHAR(80) | extra info |

### A.1.1.29. lastlog

lastlog table contains "lastlog" data of database users

| field | type | comments |
| --- | --- | --- |
| id | SERIAL PRIMARY KEY | unique ID |
| sid | INT NOT NULL | session ID |
| uid | INT NOT NULL | user ID |
| date | INT NOT NULL | date of record |
| state | INT NOT NULL | record type: 1=logged in 2=logged out 3=idle timeout4=reconnect |
| ldate | INT DEFAULT -1 | logout date |
| ip | INET | remote IP |
| host | CHAR(40) | remote host |

### A.1.1.30. news

This table contains motd/news to be displayed when user logs in...

| field | type | comments |
| --- | --- | --- |
| id | SERIAL PRIMARY KEY | unique ID |
| server | INT DEFAULT -1 | ptr to server or -1 for globalnews messages |
| info | TEXT NOT NULL | news/motd message |

## A.1.1.31. vlans

"VLAN" (Layer-2 networks/shared networks) descriptions, linked to server record. Used mainly for generating of shared-network map for DHCP.

| field | type | comments |
|---|---|---|
| id | SERIAL | unique ID |
| server | INT4 NOT NULL | ptr to a servers table record-->servers.id |
| name | TEXT NOT NULL CHECK(name <> '') | name of vlan |
| description | TEXT | long name |
| comment | TEXT | comments |
| CONSTRAINT | vlans_key PRIMARY KEY (name,server) | |

# Appendix B. Software License

```
        GNU GENERAL PUBLIC LICENSE
           Version 2, June 1991


 Copyright (C) 1989, 1991 Free Software Foundation, Inc.
                          675 Mass Ave, Cambridge, MA 02139, USA
 Everyone is permitted to copy and distribute verbatim copies
 of this license document, but changing it is not allowed.


     Preamble


   The licenses for most software are designed to take away your
 freedom to share and change it.  By contrast, the GNU General Public
 License is intended to guarantee your freedom to share and change free
 software--to make sure the software is free for all its users.  This
 General Public License applies to most of the Free Software
 Foundation's software and to any other program whose authors commit to
 using it.  (Some other Free Software Foundation software is covered by
 the GNU Library General Public License instead.)  You can apply it to
 your programs, too.


   When we speak of free software, we are referring to freedom, not
 price.  Our General Public Licenses are designed to make sure that you
 have the freedom to distribute copies of free software (and charge for
 this service if you wish), that you receive source code or can get it
 if you want it, that you can change the software or use pieces of it
 in new free programs; and that you know you can do these things.


   To protect your rights, we need to make restrictions that forbid
 anyone to deny you these rights or to ask you to surrender the rights.
 These restrictions translate to certain responsibilities for you if you
 distribute copies of the software, or if you modify it.


   For example, if you distribute copies of such a program, whether
 gratis or for a fee, you must give the recipients all the rights that
 you have.  You must make sure that they, too, receive or can get the
 source code.  And you must show them these terms so they know their
 rights.


   We protect your rights with two steps: (1) copyright the software, and
 (2) offer you this license which gives you legal permission to copy,
 distribute and/or modify the software.


   Also, for each author's protection and ours, we want to make certain
 that everyone understands that there is no warranty for this free
 software.  If the software is modified by someone else and passed on, we
 want its recipients to know that what they have is not the original, so
 that any problems introduced by others will not reflect on the original
 authors' reputations.
```

   Finally, any free program is threatened constantly by software
patents.  We wish to avoid the danger that redistributors of a free
program will individually obtain patent licenses, in effect making the
program proprietary.  To prevent this, we have made it clear that any
patent must be licensed for everyone's free use or not licensed at all.

   The precise terms and conditions for copying, distribution and
modification follow.

     GNU GENERAL PUBLIC LICENSE
    TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

   0. This License applies to any program or other work which contains
a notice placed by the copyright holder saying it may be distributed
under the terms of this General Public License.  The "Program", below,
refers to any such program or work, and a "work based on the Program"
means either the Program or any derivative work under copyright law:
that is to say, a work containing the Program or a portion of it,
either verbatim or with modifications and/or translated into another
language.  (Hereinafter, translation is included without limitation in
the term "modification".)  Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not
covered by this License; they are outside its scope.  The act of
running the Program is not restricted, and the output from the Program
is covered only if its contents constitute a work based on the
Program (independent of having been made by running the Program).
Whether that is true depends on what the Program does.

   1. You may copy and distribute verbatim copies of the Program's
source code as you receive it, in any medium, provided that you
conspicuously and appropriately publish on each copy an appropriate
copyright notice and disclaimer of warranty; keep intact all the
notices that refer to this License and to the absence of any warranty;
and give any other recipients of the Program a copy of this License
along with the Program.

You may charge a fee for the physical act of transferring a copy, and
you may at your option offer warranty protection in exchange for a fee.

   2. You may modify your copy or copies of the Program or any portion
of it, thus forming a work based on the Program, and copy and
distribute such modifications or work under the terms of Section 1
above, provided that you also meet all of these conditions:

    a) You must cause the modified files to carry prominent notices
    stating that you changed the files and the date of any change.

    b) You must cause any work that you distribute or publish, that in
    whole or in part contains or is derived from the Program or any
    part thereof, to be licensed as a whole at no charge to all third
    parties under the terms of this License.

c) If the modified program normally reads commands interactively
when run, you must cause it, when started running for such
interactive use in the most ordinary way, to print or display an
announcement including an appropriate copyright notice and a
notice that there is no warranty (or else, saying that you provide
a warranty) and that users may redistribute the program under
these conditions, and telling the user how to view a copy of this
License.  (Exception: if the Program itself is interactive but
does not normally print such an announcement, your work based on
the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole.  If
identifiable sections of that work are not derived from the Program,
and can be reasonably considered independent and separate works in
themselves, then this License, and its terms, do not apply to those
sections when you distribute them as separate works.  But when you
distribute the same sections as part of a whole which is a work based
on the Program, the distribution of the whole must be on the terms of
this License, whose permissions for other licensees extend to the
entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest
your rights to work written entirely by you; rather, the intent is to
exercise the right to control the distribution of derivative or
collective works based on the Program.

In addition, mere aggregation of another work not based on the Program
with the Program (or with a work based on the Program) on a volume of
a storage or distribution medium does not bring the other work under
the scope of this License.

  3. You may copy and distribute the Program (or a work based on it,
under Section 2) in object code or executable form under the terms of
Sections 1 and 2 above provided that you also do one of the following:

    a) Accompany it with the complete corresponding machine-readable
    source code, which must be distributed under the terms of Sections
    1 and 2 above on a medium customarily used for software interchange; or,

    b) Accompany it with a written offer, valid for at least three
    years, to give any third party, for a charge no more than your
    cost of physically performing source distribution, a complete
    machine-readable copy of the corresponding source code, to be
    distributed under the terms of Sections 1 and 2 above on a medium
    customarily used for software interchange; or,

    c) Accompany it with the information you received as to the offer
    to distribute corresponding source code.  (This alternative is
    allowed only for noncommercial distribution and only if you
    received the program in object code or executable form with such
    an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for

making modifications to it.  For an executable work, complete source
code means all the source code for all modules it contains, plus any
associated interface definition files, plus the scripts used to
control compilation and installation of the executable.  However, as a
special exception, the source code distributed need not include
anything that is normally distributed (in either source or binary
form) with the major components (compiler, kernel, and so on) of the
operating system on which the executable runs, unless that component
itself accompanies the executable.

If distribution of executable or object code is made by offering
access to copy from a designated place, then offering equivalent
access to copy the source code from the same place counts as
distribution of the source code, even though third parties are not
compelled to copy the source along with the object code.

   4. You may not copy, modify, sublicense, or distribute the Program
except as expressly provided under this License.  Any attempt
otherwise to copy, modify, sublicense or distribute the Program is
void, and will automatically terminate your rights under this License.
However, parties who have received copies, or rights, from you under
this License will not have their licenses terminated so long as such
parties remain in full compliance.

   5. You are not required to accept this License, since you have not
signed it.  However, nothing else grants you permission to modify or
distribute the Program or its derivative works.  These actions are
prohibited by law if you do not accept this License.  Therefore, by
modifying or distributing the Program (or any work based on the
Program), you indicate your acceptance of this License to do so, and
all its terms and conditions for copying, distributing or modifying
the Program or works based on it.

   6. Each time you redistribute the Program (or any work based on the
Program), the recipient automatically receives a license from the
original licensor to copy, distribute or modify the Program subject to
these terms and conditions.  You may not impose any further
restrictions on the recipients' exercise of the rights granted herein.
You are not responsible for enforcing compliance by third parties to
this License.

   7. If, as a consequence of a court judgment or allegation of patent
infringement or for any other reason (not limited to patent issues),
conditions are imposed on you (whether by court order, agreement or
otherwise) that contradict the conditions of this License, they do not
excuse you from the conditions of this License.  If you cannot
distribute so as to satisfy simultaneously your obligations under this
License and any other pertinent obligations, then as a consequence you
may not distribute the Program at all.  For example, if a patent
license would not permit royalty-free redistribution of the Program by
all those who receive copies directly or indirectly through you, then
the only way you could satisfy both it and this License would be to
refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under
any particular circumstance, the balance of the section is intended to
apply and the section as a whole is intended to apply in other
circumstances.

It is not the purpose of this section to induce you to infringe any
patents or other property right claims or to contest validity of any
such claims; this section has the sole purpose of protecting the
integrity of the free software distribution system, which is
implemented by public license practices.  Many people have made
generous contributions to the wide range of software distributed
through that system in reliance on consistent application of that
system; it is up to the author/donor to decide if he or she is willing
to distribute software through any other system and a licensee cannot
impose that choice.

This section is intended to make thoroughly clear what is believed to
be a consequence of the rest of this License.

  8. If the distribution and/or use of the Program is restricted in
certain countries either by patents or by copyrighted interfaces, the
original copyright holder who places the Program under this License
may add an explicit geographical distribution limitation excluding
those countries, so that distribution is permitted only in or among
countries not thus excluded.  In such case, this License incorporates
the limitation as if written in the body of this License.

  9. The Free Software Foundation may publish revised and/or new versions
of the General Public License from time to time.  Such new versions will
be similar in spirit to the present version, but may differ in detail to
address new problems or concerns.

Each version is given a distinguishing version number.  If the Program
specifies a version number of this License which applies to it and "any
later version", you have the option of following the terms and conditions
either of that version or of any later version published by the Free
Software Foundation.  If the Program does not specify a version number of
this License, you may choose any version ever published by the Free Software
Foundation.

  10. If you wish to incorporate parts of the Program into other free
programs whose distribution conditions are different, write to the author
to ask for permission.  For software which is copyrighted by the Free
Software Foundation, write to the Free Software Foundation; we sometimes
make exceptions for this.  Our decision will be guided by the two goals
of preserving the free status of all derivatives of our free software and
of promoting the sharing and reuse of software generally.

    NO WARRANTY

  11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY
FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW.  EXCEPT WHEN

OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES
PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED
OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF
MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE.  THE ENTIRE RISK AS
TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU.  SHOULD THE
PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING,
REPAIR OR CORRECTION.

  12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING
WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR
REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES,
INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING
OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED
TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY
YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER
PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE
POSSIBILITY OF SUCH DAMAGES.

     END OF TERMS AND CONDITIONS

Appendix: How to Apply These Terms to Your New Programs

  If you develop a new program, and you want it to be of the greatest
possible use to the public, the best way to achieve this is to make it
free software which everyone can redistribute and change under these terms.

  To do so, attach the following notices to the program.  It is safest
to attach them to the start of each source file to most effectively
convey the exclusion of warranty; and each file should have at least
the "copyright" line and a pointer to where the full notice is found.

    <one line to give the program's name and a brief idea of what it does.>
    Copyright (C) 19yy  <name of author>

    This program is free software; you can redistribute it and/or modify
    it under the terms of the GNU General Public License as published by
    the Free Software Foundation; either version 2 of the License, or
    (at your option) any later version.

    This program is distributed in the hope that it will be useful,
    but WITHOUT ANY WARRANTY; without even the implied warranty of
    MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.  See the
    GNU General Public License for more details.

    You should have received a copy of the GNU General Public License
    along with this program; if not, write to the Free Software
    Foundation, Inc., 675 Mass Ave, Cambridge, MA 02139, USA.

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this
when it starts in an interactive mode:

```
      Gnomovision version 69, Copyright (C) 19yy name of author
      Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type 'show w'.
      This is free software, and you are welcome to redistribute it
      under certain conditions; type 'show c' for details.
```

The hypothetical commands 'show w' and 'show c' should show the appropriate parts of the General Public License.  Of course, the commands you use may be called something other than 'show w' and 'show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary.  Here is a sample; alter the names:

```
  Yoyodyne, Inc., hereby disclaims all copyright interest in the program
  'Gnomovision' (which makes passes at compilers) written by James Hacker.

  <signature of Ty Coon>, 1 April 1989
  Ty Coon, President of Vice
```

This General Public License does not permit incorporating your program into proprietary programs.  If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library.  If this is what you want to do, use the GNU Library General Public License instead of this License.