sat-nms Software User Manual

Installation Manual

Version 3.80.6-a -- 2025-12-03

© Copyright
SatService Gesellschaft für Kommunikationssysteme mbH
Hardstrasse 9
D-78256 Steisslingen

satnms-support@satservicegmbh.de

www.satnms.com
www.satservicegmbh.de
Tel +49 7738 99791-10

Table of Contents

1 Installation

This section of the manual provides installation instruction for the different modules of the***sat-nms*** Monitoring & Control software.

> ***Remark:*** On a preconfigured system delivered by SatService these step as usually **not** required. So this manual is aimed especially at people who want to deploy the system in virtual machines or customer provided bare metal server or for deployments in private cloud environment. And it is intended for a better understanding of your installation.

Quick access links to the main sections of the document:

- Base Debian Linux setup
  - packages: satnms4-base, satnms4-utils
- MNC service
  - packages: satnms4-mnc, satnms4-images, satnms4-documentation, satnms4-jars
- EventDB service
  - packages: satnms4-mnc, satnms4-images, satnms4-documentation, satnms4-jars
- Backend service
  - packages: satnms4-backend
- WebClient and nginx
  - packages: satnms4-webclient, satnms4-images, satnms4-documentation
- Database SatDB
  - packages: satnms4-satdb

There are different way where to installed all these components. The simplest solution is to install all components on one host (bare metal server or virtual machine). On the other hand all components can be run on different hosts for several reasons like redundancy or load distribution.

Please refer to the chapter Architecture for the different options. TODO!!!

The following chapters consider all components as a stand-alone installation. Therefore some steps may be repeated and can be skipped in different scenarios. But the installation of all services (MNC, backend, EventDB, WebClient and SatDB) expect the base Debian Linux installation. So whatever you want to install, make sure you followed first the instruction in the chapter Base Debian Linux setup

1.1 Installation Base Debian Linux

This section provides installation instructions for the ***sat-nms*** base and utils packages for the ***sat-nms*** Monitoring & Control System.

These packages are used to add settings and tools to a standard minimal Debian 12 installation. The following chapter descripes the installation on Debian Linux 12 but should be similar for other Debian-based distributions.

1.1.1 Requirements

- Minimal Debian 12 Installation and console or SSH access
- headless installation is sufficient
- root permission in Linux either via `su` or `sudo`
    - all examples in this manual assume that you get root shell with the `su` command
    - for installation with no direct `root` login, you have to add `sudo` in front of all commands which requires `root` permission
- access to Debian 12 repositories either via Internet or from a local Debian archive mirror
    - check with `apt-get update` access to the configured repository is available.
- for base setup: `satnms4-base_[version].deb` package (in the following referred to as `satnms4-base`)
- for utilities setup: `satnms4-utils_[version].deb` package (in the following referred to as `satnms4-utils`)

1.1.2 For the impatient

This section provides a quick installation guide for the impatient and experienced Debian Linux users. Detailed information are in the following chapters.

Install Debian Linux 12 on your host with your favorite installation method.

We recommend the following settings:

- Language: `English`
- Disk: 20.0 GB, ext4 filesystem
- TaskSel: not a full installation, just select
    - SSH Server
    - Standard system utilities
- Time: `UTC`
- Default user: `satnms`

Install ***sat-nms*** base components

```
apt-get update
apt-get upgrade
apt install /home/public/satnms4-base_[version].deb
apt install /home/public/satnms4-utils_[version].deb
reboot
```

1.1.3 Installation

For the following instructions we assume that you are using Debian 12. Similar Linux distributions like Ubuntu may also work but are not covered by this manual.

Debian Linux 12

Please follow the standard Debian 12 instruction for a minimal Debian installation. We recommend the following setting for the Debian installer:

- Language: `English`
- locale: `en_GB.UTF-8`
- Network: fixed IP address
- Clock: use NTP
- Root user: set password and allow login
- Normal user: `satnms`
- Timezone: `UTC`
- Disk: 20GB partion for all with ext4
- TaskSel: only `SSH Server` and `Standard system utilities`

Additional packages

For convenience we recommend the following additional packages:

- curl
- debsums
- dialog
- htop
- mc
- rsync
- snmp
- sudo
- tcpdump
- tnftp
- vim

```
root@satnms7-dev:~# apt-get install curl debsums dialog htop mc rsync snmp sudo tcpdump
```

satnms4-base package

The `satnms4-base` package contains files and directories required by the other satnms4 components. It should be installed on each host which run ***sat-nms*** MNC components.

```
root@satnms7-dev:~# apt-get install ./satnms4-base_1.2.2-1_all.deb
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
Note, selecting 'satnms4-base' instead of './satnms4-base_1.2.2-1_all.deb'
The following NEW packages will be installed:
  satnms4-base
0 upgraded, 1 newly installed, 0 to remove and 0 not upgraded.
Need to get 0 B/462 kB of archives.
After this operation, 0 B of additional disk space will be used.
Get:1 /root/satnms4-base_1.2.2-1_all.deb satnms4-base all 1.2.2-1 [462 kB]
Selecting previously unselected package satnms4-base.
(Reading database ... 36810 files and directories currently installed.)
Preparing to unpack .../satnms4-base_1.2.2-1_all.deb ...
Unpacking satnms4-base (1.2.2-1) ...
Setting up satnms4-base (1.2.2-1) ...
Created symlink /etc/systemd/system/multi-user.target.wants/satnms.service → /etc/systemd/
Processing triggers for mailcap (3.70+nmu1) ...
```

Beside installing different files it will also setup the following:

- create a user `satnms` if not already exists
- create a directory `/home/public` which should be used later to place satnms4 packages and backups
- enable `sudo` for a list of commands
- add additional MOTD (message of the day) scripts to show information on login
- and some more

> ***Remark:*** All satnms4 packages must be installed from local source. All following instructions assume that you have these packages in the folder `/home/public` on the host where you want to install them.

If the user `satnms` does not exists before installing this package, a new user will be created. This user will be used to run the different ***sat-nms*** services. So it is required. By default it is not possible to login to this user, because there is no password set. Login as `root` and change the password for the user `satnms` :

```
root@satnms7-dev:~# passwd satnms
New password:
Retype new password:
passwd: password updated successfully
```

satnms4-utils package

The `satnms4-utils` package contains a collection of utilities for the***sat-nms*** Monitoring & Control System like backup and restore scripts.

```
root@satnms7-dev:/home/public# apt install ./satnms4-utils_1.0.6-1_all.deb
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
Note, selecting 'satnms4-utils' instead of './satnms4-utils_1.0.6-1_all.deb'
The following NEW packages will be installed:
  satnms4-utils
0 upgraded, 1 newly installed, 0 to remove and 0 not upgraded.
Need to get 0 B/30.7 kB of archives.
After this operation, 0 B of additional disk space will be used.
Get:1 /home/public/satnms4-utils_1.0.6-1_all.deb satnms4-utils all 1.0.6-1 [30.7 kB]
Selecting previously unselected package satnms4-utils.
(Reading database ... 36932 files and directories currently installed.)
Preparing to unpack .../satnms4-utils_1.0.6-1_all.deb ...
Unpacking satnms4-utils (1.0.6-1) ...
Setting up satnms4-utils (1.0.6-1) ...
Processing triggers for mailcap (3.70+nmu1) ...
```

Common Linux settings

In preinstalled systems delivered by SatService a dedicated file will display the version of the ***sat-nms*** Linux Operation System installation: `/etc/satnms_version` .

Its format is defined as: `[linux-distribution]-satnms-[version] [date]` . For customer build OS version it is required to create this file by your own.

- `linux-distribution` is usually `debian` , but other names are allowed like `ubuntu`
- the second word needs to be `-satnms-`
  - required because several tools parsing this file to get the OS version
  - the Sysinfo device uses this information to display the matching server information
- version format is `[major-version].[minor-version][suffix] [date]` e.g. `7.01`
  - major-number must be 7
  - minor-number will be increased if SatService creates new OS images
  - for customer builds minor numbers needs to be managed by customer
  - optional suffix can be added to mark special editions: e.g. `7.01idu` shows that it is a special OS image for the ***sat-nms*** ACU Indoor Unit
  - the date should be in format `YYYY-MM-DD`

```
satnms@satnms7-dev:~$ cat /etc/satnms_version
debian-satnms-7.01 2023-10-05
```

The follwing additional settings are recommended for convenience.

- `/home/satnms/.bashrc` : uncomment line `force_color_prompt=yes` to enable colored prompt
- `/root/.bashrc` : configure prompt colors, add the following line:

```
PS1='${debian_chroot:+($debian_chroot)}\[\033[01;31m\]\u@\[\033[01;33m\]\h\[\033[00m\]:
```

1.2 Installation MNC Service

This section provides installation instructions for the MNC service used by the***sat-nms*** Monitoring & Control System.

The MNC service runs device drivers and logical device and is the interface to the equipment to be monitored and controlled.

The following topics will be covered by the Administration Manual

- Update ***sat-nms*** MNC service TODO LINK !!!
- Backup and restore ***sat-nms*** MNC service settings TODO LINK !!!

1.2.1 Requirements

- Base Debian 12 Installation with installed packages
  - `satnms4-base`
  - `satnms4-utils`
- console or SSH access
- headless installation is sufficient
- root permission in Linux either via `su` or `sudo`
  - all examples in this manual assume that you get root shell with the `su` command
  - for installation with no direct `root` login, you have to add `sudo` in front of all commands which requires `root` permission
- for MNC installation the following packages:
  - `satnms4-documentation_[version].deb` package (in the following referred to as `satnms4-documentation`)
  - `satnms4-images_[version].deb` package (in the following referred to as `satnms4-images`)
  - `satnms4-jars_[version].deb` package (in the following referred to as `satnms4-jars`)
  - `satnms4-mnc_[version].deb` package (in the following referred to as `satnms4-mnc`)

1.2.2 For the impatient

This section provides a quick installation guide for the impatient and experienced Debian Linux users. Detailed information are in the following chapters.

```
apt-get update
apt-get upgrade
apt-get install openjdk-17-jre-headless
apt install /home/public/satnms4-documentation_[version].deb
apt install /home/public/satnms4-images_[version].deb
apt install /home/public/satnms4-jars_[version].deb
apt install /home/public/satnms4-mnc_[version].deb
systemctl enable satnms
systemctl start satnms
```

1.2.3 Installation

One external tools must be installed:

1. OpenJDK Java Runtime Environment (JRE)
2. JQ Pretty print of JSON data on CLI (optional)

Debian provided installation packages (`.deb`) for OpenJDK 17.

- login as `root`
- update all installed Debian packages: `apt-get update; apt-get upgrade`
- install JRE from Debian repository: `apt-get install openjdk-17-jre-headless`

example output

```
satnms@satnms7-dev:~$ su -
Password:
root@satnms7-dev:~# apt-get install openjdk-17-jre-headless
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  ... ca-certificates-java java-common ...
Suggested packages:
  ...
The following NEW packages will be installed:
  ... ca-certificates-java java-common openjdk-17-jre-headless ...
0 upgraded, 21 newly installed, 0 to remove and 1 not upgraded.
Need to get 50.3 MB of archives.
After this operation, 210 MB of additional disk space will be used.
Do you want to continue? [Y/n] y
... downloading and installing packages ...
```

To test if Java Runtime was installed successfully, run `java -version` as user `satnms` :

example output

```
satnms@satnms7-dev:/home/public$ java -version
openjdk version "17.0.14" 2025-01-21
OpenJDK Runtime Environment (build 17.0.14+7-Debian-1deb12u1)
OpenJDK 64-Bit Server VM (build 17.0.14+7-Debian-1deb12u1, mixed mode, sharing)
```

Now install the ***sat-nms*** backend software package:

- upload `satnms4-documentation` , `satnms4-images` , `satnms4-jars` and `satnms4-mnc` package to `/home/public on the target system
- login as `root`
- install the local satnms4-* packages
    - `apt install /home/public/satnms4-documentation_[version].deb`
    - `apt install /home/public/satnms4-images_[version].deb`
    - `apt install /home/public/satnms4-jars_[version].deb`
    - `apt install /home/public/satnms4-mnc_[version].deb`

example output for package satnms4-mnc

```
root@satnms7-dev:/home/public# apt-get install ./satnms4-mnc_3.68.9-1_all.deb
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
Note, selecting 'satnms4-mnc' instead of './satnms4-mnc_3.68.9-1_all.deb'
The following packages will be upgraded:
  satnms4-mnc
1 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.
Need to get 0 B/10.2 MB of archives.
After this operation, 0 B of additional disk space will be used.
Get:1 /home/public/satnms4-mnc_3.68.9-1_all.deb satnms4-mnc all 3.68.9-1 [10.2 MB]
Reading changelogs... Done
(Reading database ... 48696 files and directories currently installed.)
Preparing to unpack .../satnms4-mnc_3.68.9-1_all.deb ...
Unpacking satnms4-mnc (3.68.9-1) over (3.68.2-1) ...
Setting up satnms4-mnc (3.68.9-1) ...
```

On success you will find a multiple new files and directories `/home/satnms/` like `satnms.jar` and directories `drivers` , `protocols` etc.

1.2.4 Configuration

This section describes only the intital configuration step for the***sat-nms*** MNC service. Because the configuration of the M&C itself can be done via Web interface. Please refere to the following chapters:

- web client: Device-Setup
- web client: Screen-Editor

The system ships a default configuration which is "ready-to-run" but you can modify some configuration to adapt the system to your environment. For a all-in-one MNC System no additional configuration is required.

Several other options require your attention only for complex setups which includes server redundancy. A detailed description of all options from is available in the Administration Section of this Manual.

To ensure the MNC service will automatically start on reboot, login as root and enable this service.

- login as `root`
- enable autostart with `systemctl enable satnms"

Now you can start the backend service as user `satnms` and check if its running.

- login as `satnms`
- run the `satnms-watch`

```
┌──────────────────────┤ sat-nms state ├─────────────────────┐
│                                  │
│  hostname / MNC name : satnms7-dev / MNC               │
│  satnms OS version   : debian-satnms-7.01 2023-10-05         │
│  ────────────────────────────────────────────────── │
│                                  │
│  MNC service         : STOPPED                     │
│  MNC service version : not installed                │
│                                  │
 ...
│                                  │
│  last updated       : 2025-14-02 16:48:48 UTC              │
│  ────────────────────────────────────────────────── │
│                                  │
│         refresh      refresh service states              │
│         satnms-start start MNC Service                    │
│         backend-stop stop Backend Service                   │
│         nginx-start  start Nginx Webserver                 │
│                                  │
│         <Select service>          <Exit>              │
└──────────────────────────────────────────────────────────────────┘
```

The output depends on the installed services. In the example above, only the***sat-nms*** MNC service is installed and running.

You can also control the MNC service directly with the follwing commands:

- start MNC service `sudo systemctl start satnms`
- stop MNC service `sudo systemctl stop satnms`

Check if the MNC service start successfully:

- status of MNC service `systemctl status satnms`

```
satnms@satnms7-dev:~/backend$ systemctl status backend
TODO
```

You can test if the MNC started successfully by checking the output of the log file `/home/satnms/debug.log` . The first line shows an early stage of start process and is an indication that the MNC service begins to start. The second line shows that all sub processes are now initialized and the MNC service is ready to serve client requests. Starting from this the backend service or clients are able to connect to the MNC service.

```
satnms@satnms7-dev:~$ tail /home/satnms/debug.log
...
000000 2025-03-05 12:32:18 2025-03-05 12:32:18   I SYSTEM Loading setup data (sat-nms MI
...
000000 2025-03-05 12:32:18 2025-03-05 12:32:18   I SYSTEM VLC 3.68.9 2025-03-05 07:22:0
...
```

Some more details and extended error messages if something failed are available in `/home/satnms/.panic.log`

A typical output of the default installation should looks like this:

```
satnms@satnms:~$ cat .panic.log
satnms version 3.68.9 2025-03-05 07:22:02
TRAPREC: File traprcv.json not found
TRAPREC: Cannot parse traprcv.json
TRAPREC: Trap receiver not started.
Universal SNMP agent listening at port 2261
TCP remote control server (r/w) listening at port 2016
TCP remote control server (r/o) listening at port 2017
SSL remote control server (r/w) listening at port 2015
```

A detailed description how to run the ***sat-nms*** MNC service is available in the administration manual.

Troubleshooting

If you have trouble to bring up the MNC service check first if the MNC service is still running with:

- `systemctl status satnms` or
- `/home/satnms/satnms-stat`

Additionally you can lookup the process ID (PID) of the backend service process:

```
satnms@satnms7-dev:~$  ps aux |grep satnms.jar
satnms   8204 ?   Sl   25:18 java -cp satnms.jar:/home/satnms/jars/* -mx512m -Djava.security.
```

Now check if the MNC process with this PID (here 8204 from the second column) is associated to a process listening on TCP ports from 2000 and above. The real number of listing ports depends on the actual configuration, but at least 2000-2007 should be visible.

```
satnms@satnms7-dev:~$ ss -tanp |grep 8204
LISTEN 0      50                 *:2000             *:*    users:(("java",pid=8204,fd=11))
LISTEN 0      50                 *:2001             *:*    users:(("java",pid=8204,fd=35))
LISTEN 0      50                 *:2002             *:*    users:(("java",pid=8204,fd=42))
LISTEN 0      50                 *:2003             *:*    users:(("java",pid=8204,fd=40))
LISTEN 0      50                 *:2004             *:*    users:(("java",pid=8204,fd=12))
LISTEN 0      50                 *:2005             *:*    users:(("java",pid=8204,fd=13))
LISTEN 0      50                 *:2006             *:*    users:(("java",pid=8204,fd=19))
LISTEN 0      50                 *:2007             *:*    users:(("java",pid=8204,fd=17))
...
```

!!!! The backend internal web server runs on Port 8443 and serves only HTTPS requests. ???? You should that this process is `LISTEN` on all interfaces (`*:8443`)

1.3 Installation Backend Service

This section provides installation instructions for the backend service used by the***sat-nms*** Monitoring & Control System.

The backend provides an unified REST-API to control all parts of an sat-nms Monitoring & Control System. It is used by the sat-nms WebClient but also available for third party applications. Please refere to the [API documentation](API Reference) for details.

The following topics will be covered by the Administration Manual

- Update ***sat-nms*** backend TODO LINK !!!
- Backup and restore ***sat-nms*** backend settings TODO LINK !!!

1.3.1 Requirements

- Base Debian 12 Installation with installed packages
    - `satnms4-base`
    - `satnms4-utils`
- console or SSH access
- headless installation is sufficient
- root permission in Linux either via `su` or `sudo`
    - all examples in this manual assume that you get root shell with the `su` command
    - for installation with no direct `root` login, you have to add `sudo` in front of all commands which requires `root` permission

- for backend installation: `satnms4-backend_[version].deb` package (in the following referred to as `satnms4-backend`)

1.3.2 For the impatient

This section provides a quick installation guide for the impatient and experienced Debian Linux users. Detailed information are in the following chapters.

```
apt-get update
apt-get upgrade
apt-get install openjdk-17-jre-headless
apt install /home/public/satnms4-backend_[version].deb
cp /home/satnms/backend/backend.properties.default /home/satnms/backend/backend.proper
systemctl enable backend
systemctl start backend
```

1.3.3 Installation

One external tools must be installed:

1. OpenJDK Java Runtime Environment (JRE)
2. JQ Pretty print of JSON data on CLI (optional)

Debian provided installation packages (`.deb`) for OpenJDK 17.

- login as `root`
- update all installed Debian packages: `apt-get update; apt-get upgrade`
- install JRE from Debian repository: `apt-get install openjdk-17-jre-headless`

example output

```
satnms@satnms7-dev:~$ su -
Password:
root@satnms7-dev:~# apt-get install openjdk-17-jre-headless
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  ... ca-certificates-java java-common ...
Suggested packages:
  ...
The following NEW packages will be installed:
  ... ca-certificates-java java-common openjdk-17-jre-headless ...
0 upgraded, 21 newly installed, 0 to remove and 1 not upgraded.
Need to get 50.3 MB of archives.
After this operation, 210 MB of additional disk space will be used.
Do you want to continue? [Y/n] y
... downloading and installing packages ...
```

To test if Java Runtime was installed successfully, run `java -version` as user `satnms` :

example output

```
satnms@satnms7-dev:/home/public$ java -version
openjdk version "17.0.14" 2025-01-21
OpenJDK Runtime Environment (build 17.0.14+7-Debian-1deb12u1)
OpenJDK 64-Bit Server VM (build 17.0.14+7-Debian-1deb12u1, mixed mode, sharing)
```

Optionally install JQ an lightweight and flexible command-line JSON processor for test the backend API.

- login as `root`
- update all installed Debian packages: `apt-get update; apt-get upgrade`
- install JRE from Debian repository: `apt-get install openjdk-17-jre-headless`

example output

```
root@satnms7-dev:/home/satnms# apt-get install jq
...
The following NEW packages will be installed:
  jq libjq1 libonig5
...
```

Now install the ***sat-nms*** backend software package:

- upload `satnms4-backend` package to `/home/public on the target system
- login as `root`
- install the local satnms4-backend package

`apt install /home/public/satnms4-backend_[version].deb` .

example output

```
root@satnms7-dev:/home/public# apt install ./satnms4-backend_1.17.10-1_all.deb
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
Note, selecting 'satnms4-backend' instead of './satnms4-backend_1.17.10-1_all.deb'
The following NEW packages will be installed:
  satnms4-backend
0 upgraded, 1 newly installed, 0 to remove and 1 not upgraded.
Need to get 0 B/24.7 MB of archives.
After this operation, 0 B of additional disk space will be used.
Get:1 /home/public/satnms4-backend_1.17.10-1_all.deb satnms4-backend all 1.17.10-1 [24.7
Selecting previously unselected package satnms4-backend.
(Reading database ... 38210 files and directories currently installed.)
Preparing to unpack .../satnms4-backend_1.17.10-1_all.deb ...
Unpacking satnms4-backend (1.17.10-1) ...
Setting up satnms4-backend (1.17.10-1) ...
```

On success you will find a new directory `/home/satnms/backend/` which contains the software and configuration files.

1.3.4 Configuration

A ***sat-nms*** backend service provides a gate to all ***sat-nms*** MNC servers in your overall system. This could be just a single MNC service running on the same host as the backend (all-in-one server) or any number of MNC service distributed over your network.

There are three steps to configure the ***sat-nms*** backend service:

1. create configuration file `/home/satnms/backend/backend.properties`
2. modify configuration file
3. enable autostart of backend service

By default the `satnms4-backend` package delivers a sample configuration for an all-in-one server installation. This template can be found in `/home/satnms/backend/backend.properties.default` . You can use this template by copy it to real backend configuration file: `/home/satnms/backend/backend.properties` and then editing it regarding you requirements.

To enable a minimal backend setup with a MNC service running on localhost just use the provide template:

```
satnms@satnms7-dev:~/backend$ cp backend.properties.default backend.properties
```

A detailed description of all options from `backend.properties` is available in the administration manual.

To ensure the backend service will automatically start on reboot, login as root and enable this service.

- login as `root`
- enable autostart with `systemctl enable backend"

Now you can start the backend service as user `satnms` and check if its running.

- login as `satnms`
- run the `satnms-watch`

```
┌────────────────────┤ sat-nms state ├──────────────────────┐
│                                    │
│  hostname / MNC name : satnms7-dev / MNC              │
│  satnms OS version   : debian-satnms-7.01 2023-10-05     │
│  ____________________________________________________  │
│                                    │
 ...
│                                    │
│  backend service     : RUNNING                  │
│  backend version     : 1.17.10                 │

...
│                                    │
│  last updated        : 2025-14-02 16:48:48 UTC          │
│  ____________________________________________________  │
│                                    │
│       refresh      refresh service states          │
│       satnms-start start MNC Service               │
│       backend-stop stop Backend Service             │
│       nginx-start  start Nginx Webserver           │
│                                    │
│       <Select service>        <Exit>         │
└─────────────────────────────────────────────────────────────┘
```

The output depends on the installed services. In the example above, only the***sat-nms*** backend service is installed and running.

You can also control the backend service directly with the follwing commands:

- start backend service `sudo systemctl start backend`
- stop backend service `sudo systemctl stop backend`

Check if the backend service start successfully:

- status of backend service `systemctl status backend`

```
satnms@satnms7-dev:~/backend$ systemctl status backend
● backend.service - sat-nms MNC Backend
     Loaded: loaded (/etc/systemd/system/backend.service; disabled; preset: enabled)
     Active: active (running) since Fri 2025-02-14 16:06:10 UTC; 5min ago
   Main PID: 18136 (java)
      Tasks: 22 (limit: 2307)
     Memory: 136.9M
        CPU: 8.788s
     CGroup: /system.slice/backend.service
             └─18136 java -Djava.security.egd=file:/dev/./urandom -jar backend.jar
```

You can test if the backend is ready to reply on REST-API requests use the heathcheck API-Endpoint which does not require authentication. It requires `jq` and `curl` installed. Run: `curl -k https://localhost:8443/healthcheck | jq` and if should report some information about the running backend software.

```
satnms@satnms7-dev:~$ curl -k -s  https://localhost:8443/healthcheck | jq
{
  "dict": [
    {
      "key": "PRODUCT",
      "val": "satnms4 backend"
    },
    {
      "key": "WEBSITE",
      "val": "www.satnms.com"
    },
    {
      "key": "VERSION",
      "val": "1.17.10 2024-12-17 12:22:54"
    },
    {
      "key": "SERIALNO",
      "val": "000001"
    }
  ]
}
```

Logfiles are available:

- `cat /home/satnms/backend/backend.log`
- `journalctl -u backend`

A detailed description how to run the ***sat-nms*** backend service is available in the administration manual.

Troubleshooting

If you have trouble to connect to the backend server's API endpoints check first if the backend service is still running with:

- `systemctl status backend` or
- `/home/satnms/backend-stat`

Additionally you can lookup the process ID (PID) of the backend service process:

```
satnms@satnms7-dev:~$ ps axu | grep backend.jar
satnms   18136  0.3  8.4 2606224 170268 ?    Ssl  16:06   0:25 java -Djava.security.egd=file:/d
```

Now check if the backend process with this PID (here 18136 from the second column) is associated to a process listening on the TCP port 8443:

```
satnms@satnms7-dev:~$ ss -taunp |grep 8443
tcp   LISTEN 0      100    *:8443     *:*    users:(("java",pid=18136,fd=10))
```

The backend internal web server runs on Port 8443 and serves only HTTPS requests. You should that this process is `LISTEN` on all interfaces (`*:8443`)

1.4 Installation SatDB database

This section provides installation instructions for the PostgreSQL database used by the***sat-nms*** Monitoring & Control System.

PostgreSQL is a free and open-source relational database management system. Installation packages are available for different operating systems and Linux distributions. The following chapter descripes the installation on Debian Linux 12 with the PostgreSQL version 15.

Details about PostgreSQL: https://www.postgresql.org/

If you have your own PostgreSQL installation or PostgreSQL cluster running on a different host or as a cloud service, you can skip the chapter Installation and move directly to chapter Configuration.

To update the database schema, we are using the Golang-Migrate tool. Details can be found here: https://github.com/golang-migrate/migrate.

Quick access links to the main sections of this installation guide:

- Database Installation
- Add database user
- Create SatDB database

The following topics will be covered by the Administration Manual

- Update SatDB database TODO LINK !!!
- Backup and restore SatDB database TODO LINK !!!

1.4.1 Requirements

- Minimal Debian 12 Installation and console or SSH access
- headless installation is sufficient
- root permission in Linux either via `su` or `sudo`
  - all examples in this manual assume that you get root shell with the `su` command
  - for installation with no direct `root` login, you have to add `sudo` in front of all commands which requires `root` permission
- access to Debian 12 repositories either via Internet or from a local Debian archive mirror
  - check with `apt-get update` access to the configured repository is available.
- for SatDB database setup: `satnms4-satdb_[version].deb` package (in the following referred to as `satnms4-satdb`)

1.4.2 For the impatient

This section provides a quick installation guide for the impatient and experienced Debian Linux users. Detailed information are in the following chapters.

```
apt-get update
apt-get upgrade
apt-get install postgresql
apt install /home/public/migrate.linux-amd64.deb
apt install /home/public/satnms4-satdb_[version].deb
```

1.4.3 Installation

Two external tools must be installed:

1. PostgreSQL
2. Golang-Migrate

Debian provided installation packages (`.deb`) for PostgreSQL.

- login as `root`
- update all installed Debian packages: `apt-get update; apt-get upgrade`
- install PostgreSQL from Debian repository: `apt-get install postgresql`

example output

```
satnms@satnms7-dev:~$ su -
Password:
root@satnms7-dev:~# apt-get update
Hit:1 http://security.debian.org/debian-security bookworm-security InRelease
Hit:2 http://deb.debian.org/debian bookworm InRelease
Hit:3 http://deb.debian.org/debian bookworm-updates InRelease
Reading package lists... Done

root@satnms7-dev:~# apt-get upgrade
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
Calculating upgrade... Done
0 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.

root@satnms7-dev:~# apt-get install postgresql
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  ... postgresql-client-15 postgresql-client-common postgresql-common ...
The following NEW packages will be installed:
  ... postgresql-client-15 postgresql-client-common postgresql-common ...
0 upgraded, 17 newly installed, 0 to remove and 0 not upgraded.
Need to get 49.1 MB/49.1 MB of archives.
After this operation, 200 MB of additional disk space will be used.
Do you want to continue? [Y/n]
... downloading and installing packages ...
```

By default Linux user `postgres` has access to the database from `localhost` only without password.

To test if the PostgreSQL database was installed and started successfully, you can connect with command line tool `psql` (in short CLI):

- login as `root`
- switch to the PostgreSQL user: `su - postgres`
- run the PostgreSQL interactive terminal: `psql`
    - installed version will be show
    - an interactive prompt is available
- leave PostgreSQL terminal with `\q`
- switch back to user root: `exit`

example output

```
root@satnms7-dev:~# su - postgres
postgres@satnms7-dev:~$ psql
psql (15.10 (Debian 15.10-0+deb12u1))
Type "help" for help.

postgres=# \q
postgres@satnms7-dev:~$
```

To install Golang-migrate you need to manually download the Debian AMD64 package

- download debian package `migrate.linux-amd64.deb` from https://github.com/golang-migrate/migrate/releases/
- upload to target system in `/home/public`
- install the package with `apt-get install /home/public/migrate.linux-amd64.deb`

example output

```
root@satnms7-dev:~# apt-get install /home/public/migrate.linux-amd64.deb
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
Note, selecting 'migrate' instead of '/home/public/migrate.linux-amd64.deb'
The following NEW packages will be installed:
  migrate
0 upgraded, 1 newly installed, 0 to remove and 0 not upgraded.
Need to get 0 B/16.2 MB of archives.
After this operation, 53.2 MB of additional disk space will be used.
Get:1 /home/public/migrate.linux-amd64.deb migrate amd64 4.18.1 [16.2 MB]
Selecting previously unselected package migrate.
(Reading database ... 55809 files and directories currently installed.)
Preparing to unpack .../public/migrate.linux-amd64.deb ...
Unpacking migrate (4.18.1) ...
Setting up migrate (4.18.1) ...
```

You can test the successfull installation with `migrate --version` which should display the installed version number, e.g. 4.18.1

1.4.4 Configuration

If PostgreSQL is installed you need to setup an user and a database for the***sat-nms*** MNC system. These setups will be automatically performed if you install the `satnms4-satdb` package . Installation instructions for system where you are not able to install our Debian package, manual instructions are included in the following chapters.

The ***sat-nms*** MNC system will use the user `satnms` to connect to the database. This user will also be the owner of the SatDB. This prevents the application from requiring administrative rights for the entire database and the MNC system can use the same PostgreSQL database engine together with other software.

The ***sat-nms*** MNC system will use a database with the name `satnms`. If you are using you own PostgreSQL installation, make sure that this database name is not already used by any other application.

- upload the `satnms4-satdb` to /home/public on the target system running PostgreSQL.
- login as `root`
- install the local satnms4-satdb package `apt install /home/public/satnms4-satdb_[version].deb`. During installation you will be prompted for a password. The default password in preinstalled systems delivered by SatService is `satnms`. Please change it to secure your system.

example output

```
root@satnms7-dev:~# apt install /home/public/satnms4-satdb_0.2.0-1_all.deb
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
Note, selecting 'satnms4-satdb' instead of '/home/public/satnms4-satdb_0.2.0-1_all.deb'
The following packages were automatically installed and are no longer required:
  libwpe-1.0-1 libwpebackend-fdo-1.0-1
Use 'apt autoremove' to remove them.
The following NEW packages will be installed:
  satnms4-satdb
0 upgraded, 1 newly installed, 0 to remove and 2 not upgraded.
Need to get 0 B/5,304 B of archives.
After this operation, 0 B of additional disk space will be used.
Get:1 /home/public/satnms4-satdb_0.2.0-1_all.deb satnms4-satdb all 0.2.0-1 [5,304 B]
Selecting previously unselected package satnms4-satdb.
(Reading database ... 159218 files and directories currently installed.)
Preparing to unpack .../satnms4-satdb_0.2.0-1_all.deb ...
checking user 'satnms' ...
checking postgresql ...
/var/run/postgresql:5432 - accepting connections
checking migrate ...
check database user 'satnms' ...
create PostgreSQL user 'satnms' ...
Enter password for new role:
Enter it again:
check database 'satnms' ...
Unpacking satnms4-satdb (0.2.0-1) ...
Setting up satnms4-satdb (0.2.0-1) ...
installing SatDB database schema ...
1/u init_satdb (26.36938ms)
2/u init_satdb (29.867137ms)
```

Remark: the default password in preinstalled systems delivered by SatService is `satnms`. Please change it to secure your system.

1.4.5 PostgreSQL network configuration

By default, access to the PostgreSQL database engine is restricted to the local system (the system where the database is installed) and no remote access via network is possible. If the ***sat-nms*** MNC system is a all-in-one installation this default configuration is sufficient and you can skip this chapter.

For a distributed MNC installation network access from a different host must be possible. Usually the database runs on the central server and the connected MNC servers needs to access the database via network.

- login as `root`
- edit the file `/etc/postgresql/15/main/pg_hba.conf`
  - if a different version of PostgreSQL is installed replace `15` with the version number
  - add a line at the end of the file with `host all all [local-net] md5`
    - where [local-net] is the network from where you want to connect e.g `192.168.2.0/24`
- edit file `/etc/postgresql/15/main/postgresql.conf`
  - uncomment and change line to: `listen_addresses = '*'`
- restart PostgreSQL: `systemctl restart postgresql`

example from pg_hba.conf

```
# allow access for two additional networks
host    all             all             172.17.1.0/16           md5
host    all             all             192.168.1.0/24           md5
```

example from postgresql.conf

```
# - Connection Settings -
listen_addresses = '*'   # what IP address(es) to listen on;
port = 5432              # (change requires restart)
```

You can test the network configuration with from any other system with an installed `psql` tool, or even on the local system if you specify the IP address of your PostgreSQL server:

- login as user `satnms`
- test connection: `psql --host [server-ip-address]`

example output

```
satnms@satnms7-dev:/etc/postgresql/15/main$ psql --host 192.168.2.231
Password for user satnms:
psql (15.10 (Debian 15.10-0+deb12u1))
SSL connection (protocol: TLSv1.3, cipher: TLS_AES_256_GCM_SHA384, compression: off)
Type "help" for help.
satnms=>
```

1.4.5.1 Change users password

To avoid that the clear text password appears in the CLI history, you can use the `\password` command in `psql`.

- login as `root`
- switch to user the PostgreSQL user: `su - postgres`
- run `psql`
- enter command `\password satnms` which prompts you for a new password

example output

```
postgres@satnms7-dev:~$ psql
psql (15.10 (Debian 15.10-0+deb12u1))
Type "help" for help.

postgres=# \password satnms
Enter new password for user "satnms":
Enter it again:
postgres=# \q
```

1.4.6 Manual configuration

You can skip this section if you successfully installed the `satnms4-satdb` package.

1.4.6.1 Manual add database user

New users can be created via CLI or with SQL statements. This manual uses the CLI tool. There is no difference between database actions or CLI command, `createuser` is just a wrapper around the SQL command `CREATE ROLE`.

- Roles subsumes users and groups, so to be precises we are create a new role and not a user

- CLI https://www.postgresql.org/docs/current/app-createuser.html

- login as `root`

- switch to user the PostgreSQL user: `su - postgres`

- run `createuser --interactive --pwprompt` to create a new user

    - name: `satnms`
    - password: [set your own password here]
    - be superuser: `no`
    - allow to create databases: `no`
    - allow to create more new roles : `no`

example output

```
root@satnms7-dev:~# su - postgres
postgres@satnms7-dev:~$ pwd
/var/lib/postgresql
postgres@satnms7-dev:~$ createuser --interactive --pwprompt
Enter name of role to add: satnms
Enter password for new role:
Enter it again:
Shall the new role be a superuser? (y/n) n
Shall the new role be allowed to create databases? (y/n) n
Shall the new role be allowed to create more new roles? (y/n) n
```

1.4.6.2 Create SatDB database

New databases can be created via CLI or with SQL statements. This manual uses the CLI tool. There is no difference between database actions or CLI command, createdb is just a wrapper around CREATE DATABASE

- Database must be named `satnms` and the owner of this database must be the role `satnms` as created above
- before you try to to create the database, make sure there exist no database with the same name.
- CLI: https://www.postgresql.org/docs/current/app-createdb.html
- create database: `createdb satnms "sat-nms MNC database"`

example output

```
TODO
```

1.4.6.3 Load SatDB database schema

To initialize the newly create database with all the required object definitions (schema) the installed `migrate` tool will be used. This tool needs the schema definition files `*.sql`) which are installed by the `satnms4-satdb` package to `/home/satnms/satdb/` . Usually this package automatically runs the database migration steps, but if this fails you will find below how to run the step manually.

You can check the current database version with:

`migrate -database 'postgres://satnms:satnms@localhost:5432/satnms?sslmode=disable' -path`

- copy migration files to `/home/satnms/satdb/`
  - for each migration (aka database version) two files exists
    - `[number]_init_satdb.up.sql`
    - `[number]_init_satdb.dow.sql`
- run migration script to apply the changes.
  - `up` command migrates to the latests version
  - execute this as user `satnms`, its not necessary to switch to user `postgres` because the database and role `satnms` are already created in the previous steps.
  - `migrate -database 'postgres://satnms:satnms@localhost:5432/satnms?sslmode=dis`

Remark: If the database is already initialized it will only run the migration for the new version(s). E.g. if already database version 2 is installed and there are migration files for version 3 and 4 it will upgrade the database to version 4.

example output

```
satnms@satnms7-dev:~$ migrate -database 'postgres://satnms:satnms@localhost:5432/satnr
1/u init_satdb (81.381471ms)
2/u init_satdb (96.040331ms)
satnms@satnms7-dev:~$
```

1.4.6.4 Force database version

In case a migration step fails, migrate will not let you run other migrations on the same database and will mark the database as dirty. You have to fix the migration and then force the corresponding version.

- run `migrate -database 'postgres://satnms:satnms@localhost:5432/satnms?sslmode=disable'`
- replace [number] by the database version you want to force. e.g. `1`

1.4.6.5 Delete database user

If you created a user by mistake you can remove it. For details refer to the PostgreSQL manual:

- CLI: https://www.postgresql.org/docs/current/app-dropuser.html
- `dropuser` is a wrapper around `DROP ROLE`

example output

```
postgres@satnms7-dev:~$ dropuser --interactive satnms
Role "satnms" will be permanently removed.
Are you sure? (y/n) y
```

1.4.6.6 Remove database

If you created a database by mistake you can remove it. For details refer to the PostgreSQL manual:

- CLI https://www.postgresql.org/docs/current/app-dropdb.html
- `dropdb` is a wrapper around `DROP DATABASE`