

NAME

saptune – Comprehensive system optimisation management for SAP solutions (**Version 2**)

ATTENTION: If you still use version **1** of saptune please refer to man page saptune_v1(8) instead.

SYNOPSIS

saptune daemon [start | status | stop]

saptune note [list | verify]

saptune note [apply | simulate | verify | customise | create | revert | show | delete] NoteID

saptune note rename NoteID newNoteID

saptune solution [list | verify]

saptune solution [apply | simulate | verify | revert] SolutionName

saptune revert all

saptune version

saptune help

DESCRIPTION

saptune is designed to automate the configuration recommendations from SAP and SUSE to run an SAP application on SLES for SAP. These configuration recommendations normally referred to as SAP Notes. So some dedicated SAP Notes are the base for the work of saptune. Additionally some best practice guides are added as Note definitions to optimise the system for some really special cases.

saptune can apply settings for a single Note definition or can set all settings for a predefined solution, which is a collection of several SAP Notes. Additionally it can revert all these settings back to the previous system settings.

It can verify, if the current system configuration is in sync with a specific Note definition or a defined solution and displays the differences. Additionally, it can simulate the setting of a single SAP Note definition or a defined solution and will display the changes which will be done to the system.

saptune does not only set kernel values (like sysctl does), but also values like cpu governor, energy perf bias, force latency (dma latency) and the disk io scheduler. Additionally it will check/verify, if suitable rpm versions are installed and special kernel command line values are set, according to the relevant SAP Notes. So saptune checks and applies values in various locations during runtime like

/proc/sys/, /proc/sys/vm/, /proc/sys/kernel, /proc/sys/fs, /sys/block/*/queue/scheduler and /sys/block/*/queue/nr_requests, /sys/devices/system/cpu/*/cpufreq/scaling_governor, /sys/devices/system/cpu/*/cpuidle/state*/latency, /sys/devices/system/cpu/*/cpuidle/state*/disable, /dev/shm, /etc/security/limits.d/, /etc/systemd/logind.conf.d/ and some others.

saptune fully integrates with tuned(8), the tuned-profile name associated with this utility is "saptune".

We decided to have only ONE solution applied, but multiple Notes. Each Note is applied exactly once.

DAEMON ACTIONS

- start** Start tuned(8) daemon, set tuning profile to "saptune", and apply a set of optimisations to the system, if solutions or notes were selected during a previous call of saptune. The daemon will be automatically activated upon system boot.
- status** Report the status of tuned(8) daemon and whether it is using the correct profile.
- stop** Stop tuned(8) daemon, and revert all optimisations that were previously applied by saptune. The daemon will no longer automatically activate upon boot.

NOTE ACTIONS

Note denotes either a SAP Note, a vendor specific tuning definition or SUSE recommendation article.

- apply** Apply optimisation settings specified in the Note. The Note will be automatically activated upon system boot if the daemon is enabled.

If a Note definition contains a '**[reminder]**' section, this section will be printed after the note has applied successfully. It will be highlighted with red color.

A Note can only be applied once.

ATTENTION: Please be in mind: If a Note definition to be applied contains parameter settings which are likewise set before by an already applied Note these settings get be overwritten. The last comes, the last wins, it's all about 'order'.

So be careful when applying solutions or notes or when reverting notes, especially if these notes are part of an already applied solution. You can re-apply such a note, but the order - and may be the resulting parameter settings - will be unlike before. Special attention is needed, if customer or vendor specific notes from */etc/saptune/extra* are used.

- list** List all SAP Notes, vendor specific tuning definitions and SUSE recommendation articles that saptune is capable of implementing.

Currently implemented notes are marked with '+', if manually enabled, '*', if enabled by solutions or '-', if a note belonging to an enabled solution was reverted manually. In all cases the notes are highlighted with green color.

If an **override** file exists for a NoteID, the note is marked with 'O'.

- verify** If a Note ID is specified, saptune verifies the current running system against the recommendations specified in the Note. If Note ID is not specified, saptune verifies all system parameters against all implemented Notes. As a result you will see a table containing the following columns

SAPNote, Version | Parameter | Expected | Override | Actual | Compliant

Expected shows the values read from the Note definition file

Override shows the values found in an **override** file

Actual shows the current system value

Compliant shows **yes**, if the 'Expected' and 'Actual' value matches, or **no**, if there is no match.

In some rows you can find references to **footnotes** containing additional information. They may explain, why a value does not match.

e.g.

- [1] setting is not supported by the system
- [2] setting is not available on the system
- [3] value is only checked, but NOT set
- [4] cpu idle state settings differ
- [5] expected value does not contain a supported scheduler

If a Note definition contains a '**reminder**' section, this section will be printed below the table and the footnotes. It will be highlighted with red color.

simulate

Show all changes that will be applied to the system if the specified Note is applied. As a result you will see a table containing the following columns

Parameter | Value set | Value expected | Override | Comment

Value set shows the current system value

Value expected shows the values read from the Note definition file

Override shows the values found in an **override** file

Comment shows references to **footnotes** containing additional information. They may explain, why a value will not be set by saptune.

e.g.

[1] setting is not supported by the system

[2] setting is not available on the system

[3] value is only checked, but NOT set

[4] cpu idle state settings differ

[5] expected value does not contain a supported scheduler

If a Note definition contains a '**reminder**' section, this section will be printed below the table and the footnotes. It will be highlighted with red color.

customise

This allows to customize the values of the saptune Note definitions. The Note definition file will be copied from `/usr/share/saptune/notes` or `/etc/saptune/extra` to the override location at `/etc/saptune/override`, if the file does not exist already. After that an editor will be launched to allow changing the Note definitions. The editor is defined by the **EDITOR** environment variable. If not set editor defaults to `/usr/bin/vim`.

You can only change the value from already available parameters of the note. But you are not able to add new parameters.

If you want to use new parameters to tune the system, please create your own custom Note definition file in `/etc/saptune/extra`.

You can prevent a parameter from being changed by leaving the parameter value in the override file empty. The parameter will be marked as 'untouched' in the override column of the verify table.

The values from the override files will take precedence over the values from `/usr/share/saptune/notes` or `/etc/saptune/extra`. In such case you will not lose your customized Notes between saptune or vendor updates.

The saptune options 'list', 'verify' and 'simulate' will mark the existence of an override file and the contained values.

ATTENTION: Creating or changing an override file just changes the configuration *inside* this Note definition file, but does not change the *running* configuration of the system.

That means: When creating or changing an override file for an **already applied** Note definition, please do a '`saptune note revert <NoteID>`' and then apply this Note again, to get the changes take effect.

create This allows to create own Note definition files in `/etc/saptune/extra`. The Note definition file will be created from a template file into the location `/etc/saptune/extra`, if the file does not exist already. After that an editor will be launched to allow changing the Note definitions. The editor is defined

by the **EDITOR** environment variable. If not set editor defaults to `/usr/bin/vim`. You need to choose an unique NoteID for this operation. Use `'saptune note list'` to find the already used NoteIDs.

- revert** Revert optimisation settings carried out by the Note, and the Note will no longer be activated automatically upon system boot.
- show** Print content of Note definition file to stdout
- delete** This allows to delete a customer or vendor specific Note definition file including the corresponding override file if available. A confirmation is needed to finish the action.

ATTENTION:

Note definition files shipped by the saptune package - so called *internal* saptune Note definition files - **must not** be deleted. There will be an appropriate error message.

If a corresponding override file is available, there will be the possibility to delete this file instead.

ATTENTION:

If the Note is already applied, the command will be terminated with the information, that the Note first needs to be reverted before it can be deleted.

rename

This allows to rename a customer or vendor specific Note definition file to a new name. If a corresponding override file is available, this file will be renamed too. A confirmation is needed to finish the action.

If the **new** Note definition name already exists the command will be terminated with a respective message.

ATTENTION:

Note definition files shipped by the saptune package - so called *internal* saptune Note definition files - and their corresponding override files, if available, **must not** be renamed. There will be an appropriate error message.

ATTENTION:

If the Note is already applied, the command will be terminated with the information, that the Note first needs to be reverted before it can be deleted.

SOLUTION ACTIONS

A solution is a collection of one or more Notes. Activation of a solution will activate all associated Notes.

The solution definitions can be found in the file `/usr/share/saptune/solutions`

It's not possible to combine solutions, there can only be **one** solution enabled.

- apply** Apply optimisation settings recommended by the SAP solution. These settings will be automatically activated upon system boot if the daemon is enabled.
- list** List all SAP solution names that saptune is capable of implementing.
The currently implemented solution is marked with '*' and is highlighted with green color. A deprecated solution is marked with 'D'.
If an **override** file exists for a solution, the solution is marked with 'O'.

simulate

Show all notes that are associated with the specified SAP solution, and all changes that will be applied once the solution is activated.

- verify** If a solution name is specified, saptune verifies the current running system against the recommended settings of the SAP solution. If solution name is not specified, saptune verifies all system

parameters against all implemented solutions.

revert Revert optimisation settings recommended by the SAP solution, and these settings will no longer be activated automatically upon system boot.

REVERT ACTIONS

revert all

Revert all optimisation settings recommended by the SAP solution and/or the Notes, and these settings will no longer be activated automatically upon system boot.

VERSION ACTIONS

version

Will display the currently active saptune version.

HELP ACTIONS

help Will display the syntax of saptune

VENDOR SUPPORT

To support vendor or customer specific tuning values, saptune supports 'drop-in' files residing in */etc/saptune/extra*. All files found in */etc/saptune/extra* are listed when running '**saptune note list**'. All **note options** are available for these files.

We simplified the file name syntax for these vendor files. But the old file names are still valid and supported.

Related to this we add 'header' support (see description of section [version] in saptune-note(5)) for the vendor files as already available for the Note definition files in */usr/share/saptune/notes* to get a proper description during saptune option 'list'

Syntax of the file names:

<NoteID>.conf
e.g. V4711.conf

old syntax of the file names: <NoteID>-<description>
e.g. Vendor-Recommended_OS_Settings
or SAP4711-very_aromatic_tunings

Syntax of the file:

The content of the 'drop-in' file should be written in a INI file style with sections headed by '[section_name]' keywords. See saptune-note(5) to find the supported sections and their available options.

ATTENTION: If renaming or removing an active (aka 'already applied') note definition file from the file system the **old** name of this note still remains in the configuration of saptune. This may lead to unexpected messages.

So please always revert the note **before** renaming or removing it from the file system.

Even if editing an active vendor or customer specific note definition file on the file system level, please do a revert of that note and then apply the Note again, to get the changes take effect.

FILES

/usr/share/saptune/notes

the saptune SAP Note definitions, which can be listed by '**saptune note list**'

The files are named with the number of their corresponding SAP Note (==NoteID).

A description of the syntax and the available tuning options can be found in saptune-note(5)

Please do not change the files located here. You will lose all your changes during a saptune package

update.

/etc/sysconfig/saptune

the central saptune configuration file containing the information about the currently enabled notes and solutions, the order in which these notes are applied and the version of saptune currently used.

/etc/saptune/extra

vendor or customer specific tuning definitions.

Please see **VENDOR SUPPORT** above for more information.

/etc/saptune/override

the saptune Note definition override location.

If you need to customize the Note definitions found in */usr/share/saptune/notes* or */etc/saptune/extra*, you can copy them to */etc/saptune/override* and modify them as you need. Please stay with the original name of the Note definition (the NoteID) and do **NOT** rename it.

Or use '**saptune note customize NoteID**' to do the job for you.

/usr/share/saptune/solutions

this file contains the saptune solution definitions, which can be listed by '**saptune solution list**'

At the moment saptune supports two architectures - *ArchX86* for the x86 platform and *ArchPPC64LE* for 64-bit PowerPC little endian platform - with different solution definitions.

Please do not change as maintenance updates of package saptune will overwrite this file without preserving any custom changes.

/var/lib/saptune/saved_state/ /var/lib/saptune/parameter/

saptune was designed to preserve the state of the system before starting the SAP specific tuning, so that it will be possible to restore this previous state of the system, if the SAP specific tuning is no longer needed or should be changed.

This system state is saved during the 'apply' operation of saptune in the saptune internal used files in */var/lib/saptune/saved_state* and */var/lib/saptune/parameter*. The content of these files highly depends on the previous state of the system.

If the values are applied by saptune, no further monitoring of the system parameters are done, so changes of saptune relevant parameters will not be observed. If a SAP Note or a SAP solution should be reverted, then first the values read from the */var/lib/saptune/saved_state* and */var/lib/saptune/parameter* files will be applied to the system to restore the previous system state and then the corresponding *save_state* file will be removed.

Please do not change or remove files in this directory. The knowledge about the previous system state gets lost and the revert functionality of saptune will be destructed. So you will lose the capability to revert back the tunings saptune has done.

NOTE

When the values from the saptune Note definitions are applied to the system, no further monitoring of the system parameters are done. So changes of saptune relevant parameters by using the 'sysctl' command or by editing configuration files will not be observed. If the values set by saptune should be reverted, these unrecognized changed settings will be overwritten by the previous saved system settings from saptune.

ATTENTION

Higher or lower system values set by the system, the SAP installer or by the administrator using sysctl command or sysctl configuration files will be now **overwritten** by saptune, if they are part of the applied Note definitions.

saptune now sets the values read from the Note definition files irrespective of already set higher system

values. If you need other tuning values as defined in the Note definition files, please use the possibility to create **override** files, which contain the values you need.

SEE ALSO

saptune-note(5) saptune-migrate(7) saptune(8) saptune_v1(8) tuned(8) tuned-adm(8)

AUTHOR

Soeren Schmidt <soeren.schmidt@suse.com>, Angela Briel <abriel@suse.com>