

# Installation on SUSE Linux 10.0"

Ron Brand <ron.brand@syscp.org>  
Florian Aders <florian.aders@syscp.org>

---

## **Installation on SUSE Linux 10.0"**

by Ron Brand and Florian Aders

Copyright © 2006 SysCP

This work is licensed under the Creative Commons Attribution-Noncommercial-Share Alike 2.0 Germany License. To view a copy of this license, visit <http://creativecommons.org/licenses/by-nc-sa/2.0/de/> or send a letter to Creative Commons, 543 Howard Street, 5th Floor, San Francisco, California, 94105, USA.

---

---

---

# Table of Contents

The first words .....	vi
1. Welcome .....	vi
2. Naming conventions .....	vi
1. Requirements .....	7
1. User requirements .....	7
2. Server Requirements .....	7
2. Installing software packages .....	8
1. Online repositories .....	8
2. Required packages/modules .....	8
3. SysCP installation .....	10
1. Webfrontend .....	10
2. Backend .....	11
2.1. Apache2 .....	11
2.2. ProFTPd .....	11
2.3. Courier .....	12
2.4. Postfix .....	12
2.5. Bind Nameserver .....	13
2.6. Cron .....	13
A. The last words .....	14
1. Personal Remarks .....	14
2. Credits .....	14
3. Disclaimer .....	14

---

## List of Tables

3.1. Needed settings: .....	10
-----------------------------	----

---

# The first words

## 1. Welcome

Welcome to the installation - process for the SysCP server-management-tool. You have chosen a powerful tool with a very small need of resources. Just execute the commands we show you in this HowTo and you should be happy. If problems occur, ask on our discussion mailinglist [<http://lists.syscp.org>] or visit our forum [<http://syscp-forum.org>]. Do yourself a favour and use the powerful search function before asking any question. The results you get might be more efficient than a single question about a particular problem.

## 2. Naming conventions

In this HowTo, we used the following basic naming conventions:

Commands executed as root:

```
syscp ~ # /execute/this/command
```

Commands executed as a specific user:

```
vmail@syscp $ /execute/this/command/as/user/vmail
```

Output of various programs:

```
I am the echo of a normal command, executed in a shell
```

Content of a file:

```
# The following sets the variable PATH to a useless value  
PATH=/dev/null
```

Filenames: /etc/apache2/httpd.conf

Variable names: \$iamavariable

---

# Chapter 1. Requirements

## 1. User requirements

It is essential to be familiar with your Linux-Environment. You might encounter situations, where you definitely need a shell. If you wish to install your server according to this HowTo it is advisable to consult and understand the basics of networking, DNS and - of course - Linux itself. You don't need to know every detail of your mail system already, this will be discussed here in this HowTo. Should you, however, have no experience at all how to operate a server which is connected to the Internet, we strongly recommend the usage of a test-environment. A wrongly configured server is a target for all kind of attack and misuse.

This HowTo requires also basic-knowledge about the MySQL database server. Installation and configuration will not be discussed here. Last but not least you need to know basics about your apache webserver, without this basic knowledge you wont be able to configure your *SysCP* setup, and the software might not function properly. Every admin should be aware what kind of software is running on his/her system. Please keep in mind one basic thing:

**The best admintool can never replace a good administrator**

## 2. Server Requirements

To perform a successful installation of *SysCP* it is assumed to have a SUSE 10.0 up and running. A discussion about how to install Linux itself will not take place here. Certainly there are enough good HowTos available elsewhere and are misplaced here. In addition a MySQL database is needed too. Likewise there are good HowTos for helping you to install MySQL.

---

# Chapter 2. Installing software packages

## 1. Online repositories

Although SUSE Linux 10.0 is more a Desktop System than a server, its still possible to set up a system with fully functional *SysCP*-support. However, it is necessary to add some lately published installation-sources. Fortunately these sources include the modules that we need to built up *SysCP*. They were not shipped with the original SUSE CD/DVD. This sources are:

- [http://software.opensuse.org/download/server:/ftp/SUSE\\_Linux\\_10.0/](http://software.opensuse.org/download/server:/ftp/SUSE_Linux_10.0/)
- [http://repos.opensuse.org/home:/cboltz/SUSE\\_Linux\\_10.0/](http://repos.opensuse.org/home:/cboltz/SUSE_Linux_10.0/)

Add the above sources to your installation of SUSE

## 2. Required packages/modules

- *apache2 Webservice*
- *apache2-prefork Multi-Processing Module(mpm)similar to the process-model in Apache 1.3*
- *apache2-mod\_php5 PHP5 Module for Apache2.0*
- *mysql Database server*
- *mysql-client Standard-MySQL-CLients*
- *php5 PHP5 Core-Files*
- *php5-mysql PHP5 Extension-Module for access to MySQL database servers.*

Note: You might want to have PHP4 installed for compatibility reasons. In this case use the respective php4 - modules. This HowTo however is based and tested with PHP5.

- *postfix a powerful Mail-MTA*
- *postfix-mysql MySQL-Plugin for Postfix*
- *courier-imap lightweight server to provide IMAP and POP3 functionality*
- *courier-authlib provides authentication services*
- *courier-authlib-mysql MySQL support for the courier authentication library*
- *proftpd FTP-server*
- *proftpd-sql-mysql MySQL module for proftpd*
- *bind Domain Name Server (DNS)*
- *cyrus-sasl-sqlauxprop MySQL auxprop plugin*

- openssl *for crypted connections*
- phpMyAdmin (optional) *to manage your MySQL account over the web*
- webalizer *to manage your statistics*

Note: Keep in mind that above mentioned packages/modules will require further packages to resolve dependencies. YAST will automatically inform you upon installation which additional software is needed.

---

# Chapter 3. SysCP installation

## 1. Webfrontend

Type the following commands in a Shell, to create the directory for *SysCP*:

```
syscp ~ # mkdir /srv/www/htdocs/syscp
```

Currently there is no SUSE rpm of *SysCP*. Therefore we will get and install the \*.tar.gz:

```
syscp ~ # cd /root
```

```
syscp ~ # wget http://files.syscp.org/releases/tgz/syscp-current.tar.gz
```

In this HowTo we will install *SysCP* into the directory `/srv/www/htdocs/syscp`:

```
syscp ~ # tar xzvf /root/syscp-current.tar.gz -C /srv/www/htdocs/
```

```
syscp ~ # chown -R wwwrun:www /srv/www/htdocs/syscp/*
```

In order to provide maximum user-flexibility and let *SysCP* find its Include-files we need to change the default entry in `/etc/php5/apache2/php.ini` as follows: replace `include_path = "/usr/share/php5"` with `include_path = ".:usr/share/php5"`.

**Please note:** Due to some security-issues inside PHP, a customer can disable `open_basedir` and / or `safe_mode`, if it's disabled in your `php.ini`. The function used for this is called `ini_restore`. To prevent these attacks, you can disable this function in your `php.ini`:

```
disable_functions = ini_restore
```

To continue with your *SysCP* installation restart the needed services:

```
syscp ~ # /etc/init.d/apache2 restart
```

```
syscp ~ # /etc/init.d/mysql restart
```

Call the installer of *SysCP*: `http://<your ip>/syscp/` and hopefully you will get the message: "You have to configure SysCP first!" click on "configure" to continue the installers script.

**Important:** Should you have a MySQL root password set already, you need to give it to the installer, if you don't have set a MySQL root password already (e.g. if this is your first run of MySQL), just type in your desired root password, *SysCP* will set it for you.

After installing *SysCP* just log in with your admin account and click on **Settings** in the left-handed menu. Here we need to change some things:

**Table 3.1. Needed settings:**

Apache configuration directory:	/etc/apache2
Apache reload command:	/etc/init.d/apache2 reload
Bind config directory:	/etc/named.d
Bind reload command:	/etc/init.d/named reload

## 2. Backend

### 2.1. Apache2

See the section Configuration (in the menu) -> SUSE 10.0 -> Apache Webserver and follow the instructions there (execute the following commands in a shell.)

```
syscp ~ # echo -e "\nInclude /etc/apache2/vhosts.conf" >> /
etc/apache2/httpd.conf
```

```
syscp ~ # touch /etc/apache2/vhosts.conf
```

```
syscp ~ # mkdir -p /var/kunden/webs/
```

```
syscp ~ # mkdir -p /var/kunden/logs/
```

```
syscp ~ # /etc/init.d/apache2 restart
```

Don't forget to change this section in the `httpd.conf` (the default is shown below) according to your needs, otherwise you will get 403 error.

```
# forbid access to the entire filesystem by default
<Directory />
    Options None
    AllowOverride None
    Order deny,allow
    Deny from all
</Directory>
```

The easiest way to allow access to the customer - webs is to add the following directly after the default:

```
# Allow access to the SysCP - customer - webs
<Directory /var/kunden/webs>
    Options None
    AllowOverride None
    Order allow,deny
    Allow from all
</Directory>
```

### 2.2. ProFTPD

As mentioned earlier, the ProFTPD of SUSE 10.0 comes with MySQL support. This will make things quite easy. Just use the config files suggested by SysCP: Configuration -> SUSE 10.0 -> ProFTPD

To enable TLS-mode of ProFTPD we need some modifications. Lets create a certificate, which is needed to establish a crypted connection. It can be created easily using this command:

```
syscp ~ # openssl req -new -x509 -days 365 -nodes -out /
etc/ssl/certs/proftpd.cert.pem -keyout /etc/ssl/certs/proftpd.key.pem
```

After creating the certificate we must adjust the configuration of Proftpd. Add these lines to `/etc/proftpd/proftpd.conf`:

```
# Uncomment this if you would use TLS module:
TLSEngine                on
TLSLog                   /var/log/ftp_tls.log
TLSProtocol               SSLv23
```

```

TLSOptions                NoCertRequest
TLRSRSCertificateFile    /etc/ssl/certs/proftpd.cert.pem
TLRSRSCertificateKeyFile /etc/ssl/certs/proftpd.key.pem
TLSVerifyClient          off
# Uncomment the following line to force tls-login
#TLSRequired              on

```

Save your proftpd.conf and restart ProFTPD

```
syscp ~ # /etc/init.d/proftpd restart
```

## 2.3. Courier

Nothing much to say here: Install the suggested files from Configuration -> SUSE 10.0 -> Courier (POP3/IMAP) into the given directories and restart the services:

```
syscp ~ # /etc/init.d/courier-authdaemon restart
```

```
syscp ~ # /etc/init.d/courier-pop restart
```

Simple, isn't it? ;)

## 2.4. Postfix

We appreciate the newly added MySQL support for Postfix in SUSE 10.0. Let us start the configuration: Refer again to the Mainpage of *SysCP* and call Configuration -> SUSE 10.0 -> Postfix(MTA) you need to run the following commands in your shell:

```
syscp ~ # mkdir -p /var/spool/postfix/etc/pam.d
```

```
syscp ~ # groupadd -g 2000 vmail
```

```
syscp ~ # useradd -u 2000 -g vmail vmail
```

```
syscp ~ # mkdir -p /var/kunden/mail/
```

```
syscp ~ # chown -R vmail:vmail /var/kunden/mail/
```

Now change the following files or create them - if they do not exist - with the content shown on the "Configuration" - site.

- /etc/postfix/main.cf
- /etc/postfix/mysql-virtual\_alias\_maps.cf
- /etc/postfix/mysql-virtual\_mailbox\_domains.cf
- /etc/postfix/mysql-virtual\_mailbox\_maps.cf
- /usr/lib/sasl2/smtpd.conf

Please Note: The MySQL-password has not been replaced for security reasons. Please replace "MYSQL\_PASSWORD" on your own. If you forgot your MySQL-password you'll find it in "lib/userdata.inc.php". Then restart the Postfix MTA:

```
syscp ~ # /etc/init.d/postfix restart
```

## 2.5. Bind Nameserver

In Configuration -> SUSE 10.0 -> Bind Nameserver (DNS) you will find a short description how to configure bind, the Domain Name Server shipped with SUSE:

```
syscp ~ # echo "include \"/etc/named.d/syscp_bind.conf\";" >> /  
etc/named.conf
```

```
syscp ~ # touch /etc/named.d/syscp_bind.conf
```

After a restart, bind is ready to use:

```
syscp ~ # /etc/init.d/named restart
```

## 2.6. Cron

Configuration -> SUSE 10.0 -> Crond will guide you through the Configuration of the *SysCP* cronscript. First create the needed directory including the php.ini file for the CLI:

```
syscp ~ # mkdir -p /etc/php5/syscpcron
```

```
syscp ~ # touch /etc/php5/syscpcron/php.ini
```

Copy the suggested content into the newly created file `/etc/php5/syscpcron/php.ini`. Create the file `/etc/cron.d/syscp` with the content:

```
#  
# Set PATH, otherwise restart-scripts won't find start-stop-daemon  
#  
PATH=/usr/local/sbin:/usr/local/bin:/sbin:/bin:/usr/sbin:/usr/bin  
#  
# Regular cron jobs for the syscp package  
#  
*/5 * * * * root /usr/bin/php5 -q -C /etc/php5/syscpcron /srv/www/htdocs/syscp/scr
```

Make sure there is an empty line at the end, otherwise cron will not read the command. After a restart you should be able to have a running system.

```
syscp ~ # /etc/init.d/cron restart
```

Make sure, that your runlevel - editor is set to restart all needed services upon reboot.

**Reminder:** do NOT use any Microsoft Windows editor (e.g. Notepad), some of your files might become corrupt due to wrong line breaks.

---

# Appendix A. The last words

## 1. Personal Remarks

Finally! *SysCP* is installed and fully functional. At this point we wish you much fun with your server and *SysCP*. If you encounter any problems with this HowTo, just ask, either on our Discussion - Mailinglist (<http://lists.syscp.org>) or in the forum (<http://syscp-forum.org>), but please use the search-function first. Many question were answered in the past, your might be solved, too.

## 2. Credits

This HowTo was originally written by Ron Brand and formatted by Florian Aders.

## 3. Disclaimer

This HowTo was written to the best of our knowledge. Although it will be maintained carefully, the authors cannot guarantee a 100% error free work. Use it at your own risc. The authors can not be held responsible for damage on hard/software due to the usage of this document. Feel free to distribute this HowTo as long as the Credits and Disclaimer will remain untouched.