

Install SimpleRisk on Ubuntu 20.04

Introduction

SimpleRisk is a simple and free tool to perform risk management activities. Based entirely on open source technologies and sporting a Mozilla Public License 2.0, a SimpleRisk instance can be stood up in minutes and instantly provides the security professional with the ability to submit risks, plan mitigations, facilitate management reviews, prioritize for project planning, and track regular reviews. It is highly configurable and includes dynamic reporting and the ability to tweak risk formulas on the fly. It is under active development with new features being added all the time and can be downloaded for free or demoed at <https://www.simplerisk.it/>.

Disclaimer

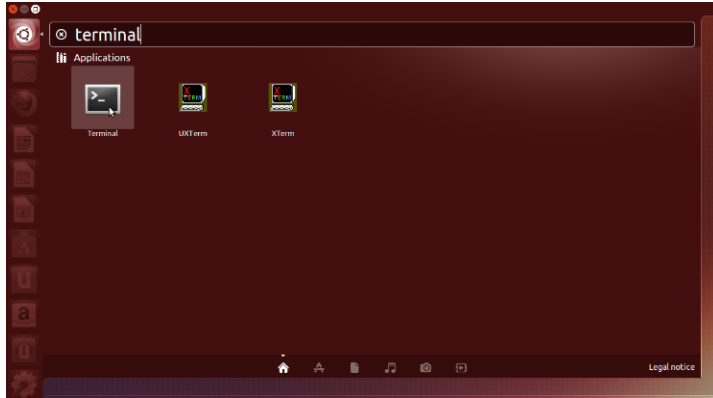
The lucky security professionals work for companies who can afford expensive GRC tools to aide in managing risk. The unlucky majority out there usually end up spending countless hours managing risk via spreadsheets. It's cumbersome, time consuming, and just plain sucks. When [Josh Sokol](#) started writing SimpleRisk, it was out of pure frustration with the other options out there. What he's put together is undoubtedly better than spreadsheets and gets you most of the way towards the "R" in GRC without breaking the bank. That said, humans can make mistakes, and therefore the SimpleRisk software is provided to you with no warranties expressed or implied. If you get stuck, you can always try sending an e-mail to support@simplerisk.it and we'll do our best to help you out. Also, while SimpleRisk was written by a security practitioner with security in mind, there is no way to promise that it is 100% secure. You accept that as a risk when using the software, but if you do find any issues, please report them to us so that we can fix them ASAP.

Install Ubuntu

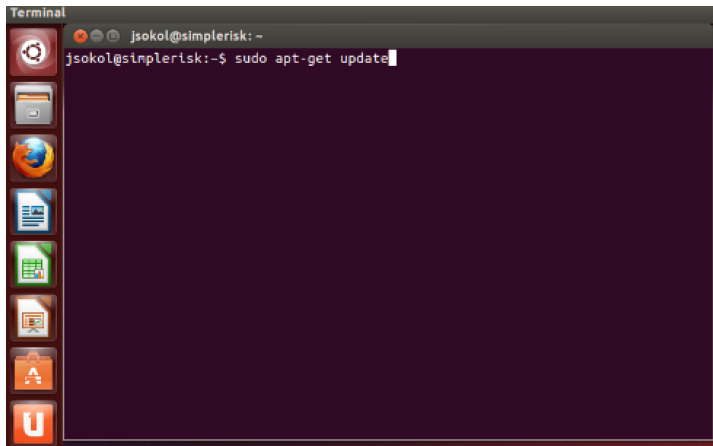
SimpleRisk should be able to work on just about any operating system that is capable of running PHP and MySQL. Since the purpose of this guide is to get you up and running with SimpleRisk as quickly as possible, we assume that you are using Ubuntu, a FREE and easy to use Linux-based operating system. Download the latest version of Ubuntu 20.04 and install it. See the Ubuntu documentation if you are having any issues there. Once you have a working installation, you can move on to the next installation steps.

Get the Latest Ubuntu Updates

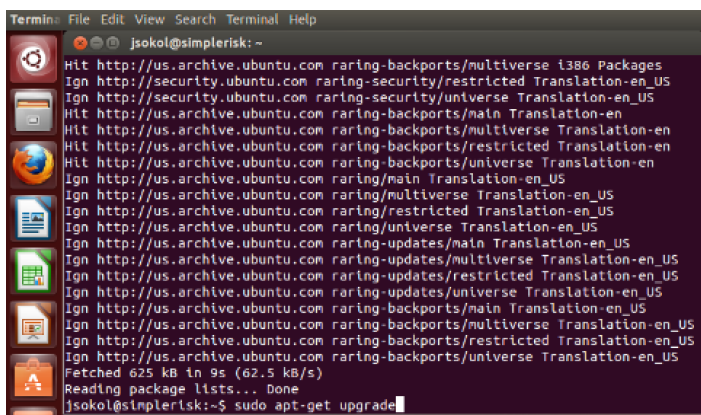
Log in to your Ubuntu installation using the username and password you defined at setup. Select the Unity menu (the one at the very top of the bar on the left) and type "terminal" in the field that pops up. This should show you a shortcut to the terminal application. You can click it to launch the terminal, but it may be a good idea to drag it to the Unity bar on the left first so that you can easily start it in the future.



Once the terminal is launched, you will want to update the OS to the latest software versions available. To do this run “sudo apt-get update” and enter your password when prompted.



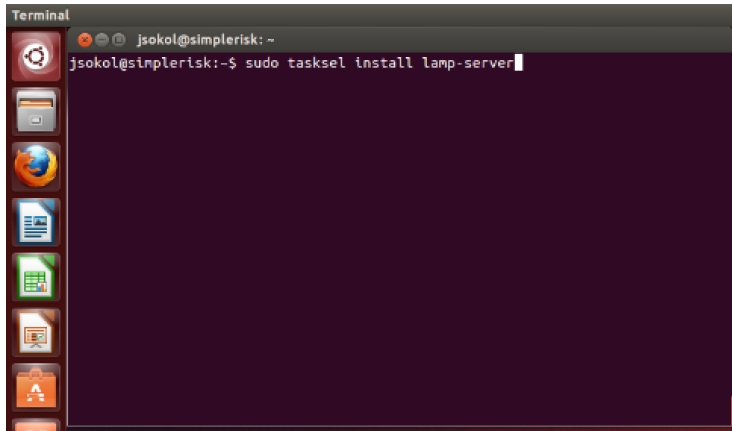
This will pull down the latest version information for all of the installed operating system files. Now run “sudo apt-get dist-upgrade” and answer “y” when it asks if you would like to continue.



Installing Apache, PHP, and MySQL

The next step is to install the necessary files in order to run Apache with PHP and MySQL on this system. To do this, first run the command “sudo apt-get install tasksel”.

Next, tell the server to install a LAMP stack by running the command “sudo tasksel install lamp-server”.



You should now see the terminal change into a package configuration application that downloads and installs the applications necessary in order to run a LAMP stack on the server. You will know that this installation process is complete when the package configuration screen goes away and you are back at the terminal shell.

Next we will need to install a few extensions to ensure SimpleRisk will run properly using the following:

```
sudo apt-get install php-mbstring php-dev php-pear php-ldap php-curl php-xml
```

```
sudo phpenmod ldap
```

Now to go ahead and setup the memory_limit for PHP we need to open the php.ini with the following:

```
vi /etc/php/php7.X/apache2/php.ini (update X for your version of apache or “ls” the /etc/php/ directory)
```

With the php.ini open search for “memory_limit” in vi “/memory_limit” should show you where the variable is stored. Update the value to 256MB save and close the file using “/wq” in vi.

Now we need to do a little setup of MySQL for the install to go smoothly, this will change the ROOT MySQL user password and we suggest making this password strong and recording it elsewhere just in case.

First login to the MySQL console using “mysql -u root -p”. The terminal will then ask for a password which should be blank on a fresh install, so just hit enter/return.

```
root@dorian-VirtualBox:/#  
root@dorian-VirtualBox:/#  
root@dorian-VirtualBox:/#  
root@dorian-VirtualBox:/#  
root@dorian-VirtualBox:/#  
root@dorian-VirtualBox:/#  
root@dorian-VirtualBox:/#  
root@dorian-VirtualBox:/#  
root@dorian-VirtualBox:/#  
root@dorian-VirtualBox:/#  
root@dorian-VirtualBox:/# mysql -u root -p  
Enter password:
```

Now in the console use the following to set the root password and confirm the plugin / change the plugin used for password authentication.

```
use mysql;
```

```
ALTER USER 'root'@'localhost' IDENTIFIED WITH mysql_native_password BY 'MyNewMySQLPassword';
```

flush privileges;

```
mysql> use mysql;
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Database changed
mysql> UPDATE user SET authentication_string=PASSWORD("simplerisk") WHERE user='root';
Query OK, 0 rows affected, 1 warning (0.00 sec)
Rows matched: 1 Changed: 0 Warnings: 1

mysql> UPDATE user SET plugin='mysql_native_password' WHERE user='root';
Query OK, 0 rows affected (0.00 sec)
Rows matched: 1 Changed: 0 Warnings: 0

mysql> flush privileges;
Query OK, 0 rows affected (0.01 sec)

mysql>
```

note: in this screenshot this has already been done hence 0 rows affected your should show 1 row affected.

The next step of setting up MySQL for a SimpleRisk install will be to set the sql-mode. To do this use the following steps:

- 1) use “vi /etc/mysql/mysql.conf.d/mysqld.cnf”
- 2) At the bottom of the config file add the following to set the sql-mode.

sql-mode="NO_ENGINE_SUBSTITUTION"

```
#
# Error log - should be very few entries.
#
log_error = /var/log/mysql/error.log
#
# Here you can see queries with especially long duration
#log_slow_queries = /var/log/mysql/mysql-slow.log
#long_query_time = 2
#log-queries-not-using-indexes
#
# The following can be used as easy to replay backup logs or for replication.
# note: if you are setting up a replication slave, see README.Debian about
# other settings you may need to change.
#server-id = 1
#log_bin = /var/log/mysql/mysql-bin.log
expire_logs_days = 10
max_binlog_size = 100M
#binlog_do_db = include_database_name
#binlog_ignore_db = include_database_name
#
# * InnoDB
#
# InnoDB is enabled by default with a 10MB datafile in /var/lib/mysql/.
# Read the manual for more InnoDB related options. There are many!
#
# * Security Features
#
# Read the manual, too, if you want chroot!
#chroot = /var/lib/mysql/
#
# For generating SSL certificates I recommend the OpenSSL GUI "tinyca".
#
#ssl-ca=/etc/mysql/cacert.pem
#ssl-cert=/etc/mysql/server-cert.pem
#ssl-key=/etc/mysql/server-key.pem
sql-mode="STRICT_TRANS_TABLES,NO_ENGINE_SUBSTITUTION"
```

3)
Now
write
the
file
out

(esc, :wq, enter).

Now we will configure Apache for the SimpleRisk API

- 1) Run the command “a2enmod rewrite” to enable mod_rewrite for Apache.
- 2) Now we will create the virtualhost configuration for SimpleRisk create new text file by using:
vi /etc/apache2/sites-enabled/simplerisk.conf.
- 3) Now using the following as an example you can copy edit the virtualhost configuration like so:

```
<VirtualHost *:80>

    ServerName simplrisk

    DocumentRoot "/var/www/simplrisk"

    <Directory "/var/www/simplrisk">

        Options -Indexes

        AllowOverride All

        allow from all

    </Directory>

</VirtualHost>
```

3b) If you would like to go ahead and setup your SSL certificate at this time please see: [How to Install a SSL Certificate for SimpleRisk](#)

4) Save the file and close your text editor.

5) Restart Apache by running the command “service apache2 restart”.

Obtaining the SimpleRisk Files

Click on the FireFox logo in the Unity bar on the left. Once FireFox loads, enter <https://www.simplrisk.com/> into the URL bar to go to the SimpleRisk site. Click on the “Download” link at the top.

Step 1

Choose Your Download Type:

Install the SimpleRisk Web Files and Database Myself

Step 2

Download the Web Bundle:

[SIMPLERISK 20160612-001 WEB BUNDLE](#)

Validate the Checksum:

MD5 Checksum = 5a22619c0248a32985e3e8d45c21456f

Step 3

Download the Installer Script:

[SIMPLERISK 20160612-001 INSTALLER](#)

Validate the Checksum:

MD5 Checksum = 0a8881bd6ed29039988a3f3cc148f0fe

Step 4

Follow the Instructions:

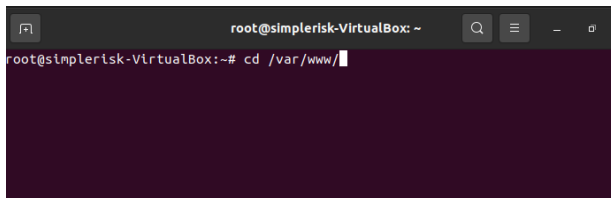
[INSTALL SIMPLERISK ON UBUNTU 13.04 \(APACHE/MYSQL/PHP\)](#)

[View All Releases and Release Notes](#)

Click to download and save both the Web Bundle and the Installer Script. Once you have the files downloaded, you can close the browser.

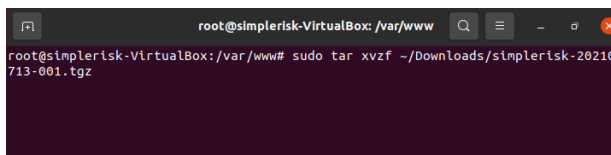
Installing the Web Files

Change to the new Apache web root by running the command “cd /var/www/”.



```
root@simplerisk-VirtualBox: ~  
root@simplerisk-VirtualBox:~# cd /var/www/
```

Remove the default index page using the command “sudo rm index.html”. Extract the web bundle into the web directory using the command “sudo tar xvf ~/Downloads/simplerisk-20210713-001.tgz” (or whatever the most current version available is).



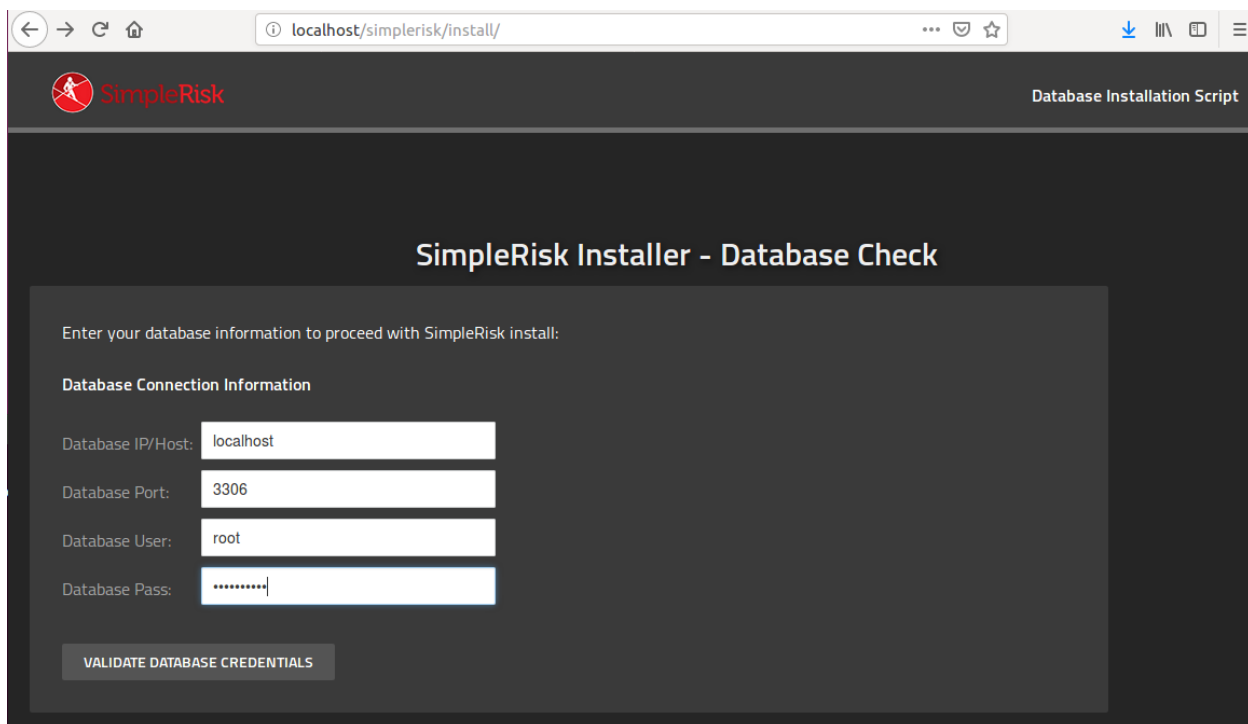
```
root@simplerisk-VirtualBox: /var/www  
root@simplerisk-VirtualBox:/var/www# sudo tar xvf ~/Downloads/simplerisk-20210713-001.tgz
```

This will extract the files into a directory into the directory /var/www/.

Now we change the ownership permissions of the “simplerisk” directory and all its sub-directories to be owned by the www-data user (or whatever user Apache is running as) using the command “sudo chown -R www-data: /var/www/*”.

Installing the Database

Now change directory into the simplerisk directory using “cd /var/www/html/simplerisk/” and extract the matching SimpleRisk installer to the “simplerisk” directory using the command “sudo tar xvf ~/Downloads/simplerisk-installer-20160612-001.tgz” (or whatever the most current version available is). This will create a new “install” directory. Next, in your web browser, navigate to <http://localhost/install> on your SimpleRisk instance. If everything works as expected, you will see an installer page designed to configure the database for you.

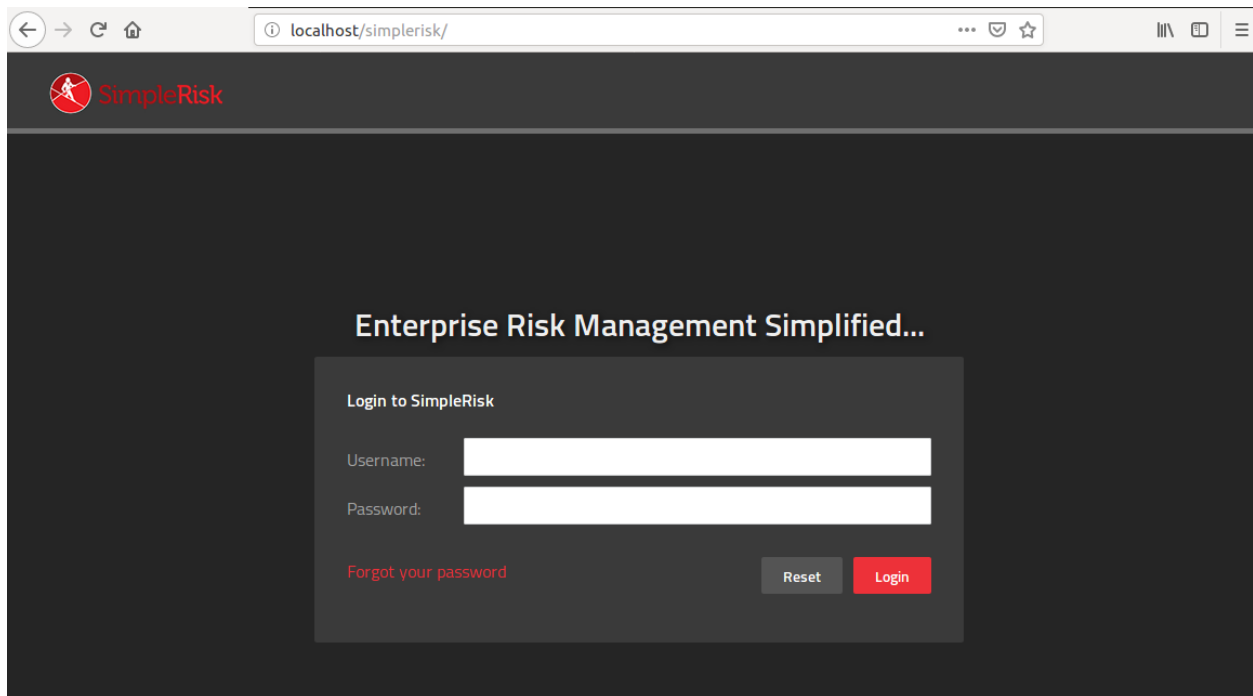


The screenshot shows a web browser window with the address bar displaying `localhost/simplerisk/install/`. The page has a dark theme with a red SimpleRisk logo in the top left and the text "Database Installation Script" in the top right. The main heading is "SimpleRisk Installer - Database Check". Below this, a message says "Enter your database information to proceed with SimpleRisk install:". Under the "Database Connection Information" section, there are four input fields: "Database IP/Host" with the value "localhost", "Database Port" with "3306", "Database User" with "root", and "Database Pass" with masked characters. A "VALIDATE DATABASE CREDENTIALS" button is at the bottom of the form.

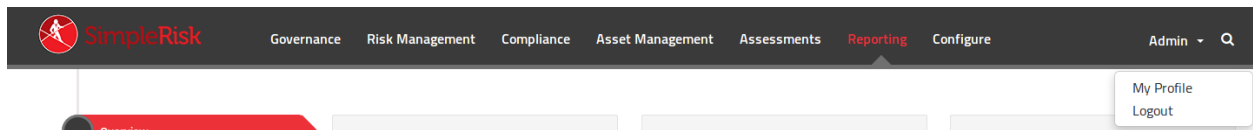
Under the Database Connection Information, provide it with your database hostname, port, username, and password (We set the root password to “simplerisk” in the previous step) Under the SimpleRisk Installation Information, provide it with the SimpleRisk database hostname, database name, and username that you would like SimpleRisk to use. A random password will be generated for the by default, it will generate a strong, randomly generated database password and we recommend that you keep that value. Under the SimpleRisk Configuration Information, you have the ability to choose the default language, session timeouts, default timezone, and other options. With the exception of the database schema language and timezone, we recommend that you keep the default values. When the script completes, it will ask if you would like to install a new `/includes/config.php` page. Select “Update” to have it automatically updated with the installer information. If it does not have permission to write to the file, or cannot find the existing `config.php` file, then it will provide you with the contents to place in it instead. It is always a good idea to delete the “install” directory once it is no longer needed using the command “sudo rm -r install”.

Logging in to SimpleRisk

You should now have performed all of the steps you need to for SimpleRisk to be up and running. Now is the moment of truth where we hopefully get to see if all of your hard work paid off. You now need to point your web browser to the URL where SimpleRisk would be installed. If you followed the optional instructions, then it should be located at <http://localhost/>. If you did not, then it is probably located at <http://localhost/simplerisk/>. You will know that you've got the right page when you see something like this:



Enter username “admin” and password “admin” to get started. Then, select the “Admin” dropdown at the top right and click on “My Profile”.



Enter your current password as “admin” and place a new long and randomly generated password into the “New Password” and “Confirm Password” fields. Then click “Submit”.

Change Password

Password should have the following requirements.

- Password should contain the minimum of 8 characters.
- Password should contain an alpha character.
- Password should contain an uppercase character.
- Password should contain a lowercase character.
- Password should contain a digit.
- Password should contain a special character.

Current Password:

New Password:

Confirm Password:

Submit

Reset

You should receive a message saying that your password was updated successfully. If so, then this is your new “admin” password for SimpleRisk. If you received a message saying that “The password entered does not adhere to the password policy”, you can change the policy by selecting “Configure” from the menu at the top followed by “User Management” on the left side. You will see a “Password Policy” section at the bottom of the page where you can change the policy and try changing your password again.

Registering SimpleRisk

This step is completely optional, but without it upgrades of SimpleRisk will require manual downloads of the new version, backing up your configuration file, extracting the new files, restoring the configuration file, and a database upgrade. It sounds like more effort than it really is, but we’ve made the process far simpler if you’re willing to tell us who you are. To register your SimpleRisk instance, select “Configure” from the menu at the top followed by “Register & Upgrade” from the menu at the left.

Enter your information and select the “Register” button. This will create a unique Instance ID for your SimpleRisk instance and download the Upgrade Extra which enables functionality for one-click backups and upgrades. If you run into issues with the registration process, we recommend that you check to ensure that the “simplerisk” directory and its sub-directories are writeable by the www-data user (or whatever user Apache is running as).

**** This completes your installation of SimpleRisk ****

SimpleRisk Paid Support and Extras

Everything that you've seen up to this point is completely free for you to install and use, forever. That said, we offer a number of ways for you to enhance your SimpleRisk instance with even more functionality. If you like what you see, and find it useful, please consider purchasing one of our inexpensive Paid Support plans or Extra functionality so that we can continue to offer you the best open source risk management tool available. Thank you!