

# SciOffice deploy

## 1. Description

Replace examples

<server ip> -> 5.227.18.5  
<ssh port> -> 22  
<server hostname> -> ngtudocs.com  
<company name> -> ngtu  
<company timezone> -> *Europe/Moscow*  
<user name> -> zena  
<user email> -> zenkov@scioffice.com  
<db root password> -> Fx48usd1  
<mysql scioffice password> -> Fx48usd2  
<company noreply email password> -> Fx48usd3  
<supervisor password> -> Fx48usd4  
<django secret key> -> generate on <https://djecrety.ir/>

*Button names printed in italic text format*

*# server bash command on grey background*

*# user pc bash command on blue background*

## 2. Requirements

Процессор: 2.4ГГц, 4 ядра;

Оперативная память: 16 Гб;

Диск: 500 Гб;

ОС: Debian Linux 10 Jessie;

Выделенный внешний IP с настроенным DNS.

Noreply email (логин, пароль), для отправки сообщений с сервера. Желательно gmail.

## 3. Putty

Set:

*Session->Host Name = <server ip>*

*Session->Port = <ssh port>*

*Window->Translation->Remote character set = UTF-8*

*Connection->Data->Auto-login username = root*

*Terminal->Features->Disable application keypad = True*

*Session->Saved Sessions = <company name> Production Server*

Click Save

Click Open

```
if message == 'Server sent disconnect message
    type 2 (protocol error):
    "Too many authentication failures for root from
    ... ssh2"
```

Use terminal before chapter "6. Create user" will be done

```
# ssh -o PreferredAuthentications=password -o PubkeyAuthentication=no
root@<server ip> -p <ssh port>
```

## 4. Change language

```
# export LANG=en_US.UTF-8
# dpkg-reconfigure locales
Select en_US.UTF-8 UTF-8
Select Default locale for the system environment = en_US.UTF-8
# apt-cache search kde-l10n
Find kde-l10n-engb
# apt-get install kde-l10n-engb
# reboot
```

## 5. Update OS

```
# apt-get update
# apt-get -V upgrade
# apt-get install locate mlocate
# updatedb
```

## 6. Create user

```
# adduser <user name>
Create password
# apt-get install sudo
# usermod -a -G sudo <user name>
user-pc # cd ~/.ssh
user-pc # ssh-keygen -t rsa -b 4096 -C "<user email>"
Create the same password
Name file <company name>_rsa
user-pc # ssh-copy-id -o PubkeyAuthentication=no -i ~/.ssh/<company
name>_rsa.pub <server ip> -p <ssh port>
user-pc # sudo apt-get install putty-tools
user-pc # puttygen <company name>_rsa -o <company name>_rsa.ppk -O private
Open Putty
Select Session->Saved Sessions = <company name>
Click Load
Set Connection->Data->Auto-login username = <user name>
Set Connection->SSH->Auth->Private key file for authentication = /home/<user
name>/.ssh/<company name>_rsa.ppk
Click Save
```

Click *Open*

## 7. Color Bash Prompt

```
# mcedit ~/.bashrc
Set force_color_prompt=yes
# sudo mcedit /root/.bashrc
Set
```

```
PS1='${debian_chroot:+($debian_chroot)}\[\033[01;31m\]u@\h\[\033[00m\]:\[\033[01;34m\]\w\[\033[00m\]\$ '
```

Find # *You may uncomment the following lines if you want `ls` to be colorized:* and uncomment the following lines

Reconnect

## 8. Set default editor

```
# update-alternatives --config editor
```

Select mcedit number

In mc select *Options->Configuration->Use internal edit* and then *Options->Save*

*setup*

## 9. Install ntp (Network Time Protocol)

```
# apt-get install ntp
# dpkg-reconfigure tzdata
Select Europe
Select <company timezone>
# reboot
```

## 10. Install webmin

```
# mcedit /etc/apt/sources.list.d/webmin.list
Add deb http://download.webmin.com/download/repository sarge contrib
Add deb http://webmin.mirror.somersettechsolutions.co.uk/repository sarge contrib
Save
# cd /var/tmp
# wget http://www.webmin.com/jcameron-key.asc
# apt-key add jcameron-key.asc
# apt-get update
# apt-get install webmin
```

## 11. Harden kernel using /etc/sysctl.conf

```
# mcedit /etc/sysctl.d/local.conf
```

Paste:

```
### Hetzner Online GmbH installimage
# sysctl config
#net.ipv4.ip_forward=1
net.ipv4.conf.all.rp_filter=1
net.ipv4.icmp_echo_ignore_broadcasts=1
net.ipv4.tcp_syncookies = 1
# ipv6 settings (no autoconfiguration)
net.ipv6.conf.default.autoconf=0
net.ipv6.conf.default.accept_dad=0
net.ipv6.conf.default.accept_ra=0
net.ipv6.conf.default.accept_ra_defrtr=0
net.ipv6.conf.default.accept_ra_rtr_pref=0
net.ipv6.conf.default.accept_ra_pinfo=0
net.ipv6.conf.default.accept_source_route=0
net.ipv6.conf.default.accept_redirects=0
net.ipv6.conf.default.forwarding=0
net.ipv6.conf.all.autoconf=0
net.ipv6.conf.all.accept_dad=0
net.ipv6.conf.all.accept_ra=0
net.ipv6.conf.all.accept_ra_defrtr=0
net.ipv6.conf.all.accept_ra_rtr_pref=0
net.ipv6.conf.all.accept_ra_pinfo=0
net.ipv6.conf.all.accept_source_route=0
net.ipv6.conf.all.accept_redirects=0
net.ipv6.conf.all.forwarding=0
# Disable IPv6 (optional)
net.ipv6.conf.all.disable_ipv6 = 1
net.ipv6.conf.default.disable_ipv6 = 1
net.ipv6.conf.lo.disable_ipv6 = 1
net.ipv6.conf.eth0.disable_ipv6 = 1
# reboot
```

## 12. Config email

```
# apt-get install exim4
# dpkg-reconfigure exim4-config
Change default option from "local delivery only" to "internet site ..."
Accept the default settings in the remaining steps.
# mcedit /etc/exim4/update-exim4.conf.conf
Replace dc_local_interfaces='127.0.0.1 ; ::1' with dc_local_interfaces='127.0.0.1'
# mcedit /etc/exim4/exim4.conf.template
Add to begging of file disable_ipv6 = true
# reboot
# echo "This is the body for direct email" | mail -s "Subject for direct email " "<user
email>"
# mcedit /etc/aliases
Add root: <user email>, root
# echo "This is the body for root" | mail -s "Subject for root" root
```

## 13. Iptables firewall

Open *Webmin->Networking->Linux Firewall*  
Click *Reset Firewall*  
Select the last option *Block all except ports used for virtual hosting, on interface eth0*  
Check *Enable firewall at boot time?*  
Click *Setup Firewall*  
Select *Module config->Display comment in rules list? = Yes*  
Remove DNS, FTP, POP3, IMAP rules:  
    *If protocol is TCP and destination port is 53 (DNS)*  
    *If protocol is UDP and destination port is 53 (DNS)*  
    *If protocol is TCP and destination port is 20:21 (FTP)*  
    *If protocol is TCP and destination ports are 110,995 (POP3)*  
    *If protocol is TCP and destination ports are 143,220,993 (IMAP)*  
Add  
    *If protocol is TCP and destination port is 8000 (Daphne server)*  
    *If protocol is TCP and destination port is 8090 (Supervisor) and "Action to  
take = Drop"*  
# reboot

## 14. Git

```
# apt-get install git
# git config --global user.name '<user name>'
```

```
# git config --global user.email <user email>
# git config --global core.editor "mcedit"
# git config --global core.autocrlf input
# git config --list
```

```
# mkdir /root/.ssh
# cd /root/.ssh
# ssh-keygen -t rsa -C "<user email>"
No password
Name file git_deploy_rsa
Copy data from /root/.ssh/git_deploy_rsa.pub
Click Add Key on https://git.scioffice.com/zenkov/git\_deploy/settings/repository
Paste public key
Add to /home/<user name>/.profile:
agent_is_running() {
    if [ "$SSH_AUTH_SOCK" ]; then
        # ssh-add returns:
        # 0 = agent running, has keys
        # 1 = agent running, no keys
        # 2 = agent not running
        ssh-add -l >/dev/null 2>&1 || [ $? -eq 1 ]
    else
        false
    fi
}
if ! agent_is_running; then
    eval "$(ssh-agent -s)"
    ssh-add /home/<user name>/.ssh/git_deploy_rsa
fi
```

```
# mcedit ~/.ssh/config
Paste:
Host git.scioffice.com
    Hostname git.scioffice.com
    IdentityFile ~/.ssh/git_deploy_rsa
    IdentitiesOnly yes
# reboot
```

## 15. Tools

```
# apt-get install p7zip-full
# apt-get install zip
# apt-get install htop
# apt-get install apticron
# apt-get install debian-goodies
# checkrestart
```

```
# apt-get install needrestart
# apt-get install redis-server
# apt-get install apache2-dev
# apt-get install rabbitmq-server
```

## 16. Antivirus

```
# cd /var/tmp
# wget
https://netix.dl.sourceforge.net/project/rkhunter/rkhunter/1.4.6/rkhunter-1.4.6.tar.gz
# tar xvf rkhunter-1.4.6.tar.gz
# /var/tmp/rkhunter-1.4.6/installer.sh --install
# ln -s /usr/local/bin/rkhunter /usr/bin/rkhunter
# rkhunter --update
# rkhunter --propupd
# mcedit /etc/rkhunter.conf
Change MAIL-ON-WARNING=<user email>
Uncomment MAIL_CMD=mail -s "[rkhunter] Warnings found for ${HOST_NAME}"
```

## 17. Monitor resources

```
# mkdir -p /root/scripts/monitor
Copy cpu-alert.sh and memory-alert.sh to created folder.
# chmod 700 /root/scripts/monitor/cpu-alert.sh
# chmod 700 /root/scripts/monitor/memory-alert.sh
```

## 18. Node.js

```
# cd /var/tmp
# apt-get install curl software-properties-common
# curl -sL https://deb.nodesource.com/setup_11.x | sudo bash -
# apt-get install nodejs
```

## 19. MySQL

```
# cd /var/tmp
# wget https://dev.mysql.com/get/mysql-apt-config_0.8.12-1_all.deb
# dpkg -i mysql-apt-config_0.8.12-1_all.deb
# apt-get update
```

```
# apt-get install mysql-community-server
Choose MySQL 5.7
Set <db root password>
# mysql_secure_installation
# systemctl restart mysql.service
# apt-get install libmysqlclient-dev
# reboot
Create user lecsus with <mysql scioffice password>
Create DB scioffice and grant all permissions for user lecsus in Webmin
```

## 20. MongoDB

```
# apt-key adv --keyserver hkp://keyserver.ubuntu.com:80 --recv
2930ADAE8CAF5059EE73BB4B58712A22-91FA4AD5
# echo "deb http://repo.mongodb.org/apt/debian jessie/mongodb-org/3.6 main" | tee
/etc/apt/sources.list.d/mongodb-org-3.6.list
# apt-get update
# apt-get install -y mongodb-org
# service mongod start
# mcedit /var/log/mongodb/mongod.log
Find waiting for connections on port 27017
# mongo --port 27017
Paste:
use admin
db.createUser(
  {
    user: "root",
    pwd: "<db root password>",
    roles: [ { role: "userAdminAnyDatabase", db: "admin" }, "readWriteAnyDatabase" ]
  }
)

db.createUser(
  {
    user: "lecsus",
    pwd: "<mysql scioffice password>",
    roles: [
      { role: "readWrite", db: "scioffice" }
    ]
  }
)

db.createRole(
  {
    role: "executeEval",
    privileges: [ {
      resource: { anyResource: true },
      actions: [ "anyAction" ] } ],
```



```
roles: []
})
```

```
db.grantRolesToUser("lecsus", [{ role: "executeEval", db: "admin" } ])
```

Type *exit*

```
# mcedit /etc/mongod.conf
```

Replace *#security:* with:

```
security:
```

```
  authorization: enabled
```

```
# service mongod start
```

```
# mongo --port 27017
```

Paste:

```
use admin
```

```
db.auth("root", "<db root password>")
```

Check grant access

Type *exit*

```
# systemctl enable mongod.service
```

```
# reboot
```

## 21. Python

Don't use *deb* <http://ftp.de.debian.org/debian> testing main

```
# apt-get update
```

```
# apt-get upgrade
```

```
# apt-get dist-upgrade
```

```
# apt-get install build-essential
```

```
# cd /var/tmp
```

```
# wget https://www.openssl.org/source/openssl-1.1.1a.tar.gz
```

```
# tar xvf openssl-1.1.1a.tar.gz
```

```
# cd /var/tmp/openssl-1.1.1a
```

```
# ./config
```

```
# make
```

```
# make install
```

```
# ln -sf /usr/local/ssl/bin/openssl `which openssl`
```

```
# openssl version -v
```

Check version 1.1.1a

if `version_check == False` and no directory `/usr/local/ssl/bin`:

```
# cd /var/tmp
```

```
# rm -rf /var/tmp/openssl-1.1.1a
```

```
# tar xvf openssl-1.1.1a.tar.gz
```

```
# cd /var/tmp/openssl-1.1.1a
```

```
# ./config --=/opt
```

```
# make
```

```
# make install
```

```
# ln -sf /opt/bin/openssl `which openssl`
```

```
# reboot
```

```
# openssl version -v
```

Check version 1.1.1a

```

# apt-get install libreadline-gplv2-dev libncursesw5-dev libssl-dev libsqlite3-dev
tk-dev libgdbm-dev libc6-dev libbz2-dev
# apt-get install python-dev python-setuptools python-pip python-smbus checkinstall
# apt-get install libffi-dev
# cd /var/tmp
# wget https://www.python.org/ftp/python/3.7.1/Python-3.7.1.tar.xz
# tar xvf Python-3.7.1.tar.xz
# cd /var/tmp/Python-3.7.1
# ./configure --enable-shared --with-ensurepip=install
# make -j4
# make altinstall
# python3.7
if error == 'error while loading shared libraries ...'
# mcedit /etc/ld.so.conf.d/python3_7.conf
    Paste /usr/local/lib
    # ldconfig

```

## 22. Apache

```

# apt-get install apache2 apache2-mpm-prefork
# apachectl configtest
if output.startswith('AH00558 ... Could not reliably determine the server's fully
qualified domain name ...'):
    # mcedit /etc/apache2/apache2.conf
    Paste ServerName <server ip> after #ServerRoot "/etc/apache2"
# apachectl configtest
Check Syntax OK
# service apache2 reload
Check in browser <server ip>
# cd /etc/apache2/sites-available
# a2dissite 000-default
# mcedit /etc/apache2/apache2.conf
Change:
<Directory /var/www/>
    Options -Indexes +FollowSymlinks
    ...
</Directory>
# service apache2 reload

# mcedit /etc/supervisor/conf.d/apache.conf
Paste:
[program:apache]
command=apache2ctl -DFOREGROUND
autostart=true
autorestart=true
startretries=1
startsecs=1
redirect_stderr=true

```

```
stderr_logfile=/var/log/apache2/apache_error.log
stdout_logfile=/var/log/apache2/apache_out.log
user=root
killasgroup=true
stopasgroup=true
```

```
# update-rc.d apache2 disable
# reboot
```

## 23. SciOffice

Create mail box *<company name>.noreply@gmail.com*

Click user avatar icon, click "Аккаунт Google"

Navigate to *Безопасность -> Ненадежные приложения, у которых есть доступ к аккаунту* and set *Вкл*

```
# cd /var/www/html
# git clone git@git.scioffice.com:zenkov/scioffice.git
# cd /var/www/html/scioffice
# git fetch
# git merge origin/master
```

```
# mcedit /var/www/html/scioffice/lecsus/settings.py
```

Change:

```
DEBUG = False
```

```
PRODUCTION = True
```

```
SERVER_EMAIL = '<company name>.noreply@gmail.com'
```

```
DEFAULT_FROM_EMAIL = '<company name>.noreply@gmail.com'
```

```
EMAIL_HOST_USER = '<company name>.noreply@gmail.com'
```

```
EMAIL_HOST_PASSWORD = '<company noreply email password>'
```

```
ALLOWED_HOSTS = [
    '<server ip>',
    '<server hostname>',
    '<server hostname>.',
]
```

```
DATABASES = {
    'default': {
        'ENGINE': 'django.db.backends.mysql',
        'NAME': 'scioffice',
        'USER': 'lecsus',
```

```

    'PASSWORD': '<mysql scioffice password>',
    'HOST': 'localhost',
    ...
  },
}
DOCUMENTS_DATABASE = {
    'NAME': 'scioffice',
    'HOST': 'localhost',
    'USER': 'lecsus',
    'PASSWORD': '<mysql scioffice password>',
}

SECRET_KEY = '<django secret key>'

TIME_ZONE = '<company timezone>'

# mcedit /var/www/html/scioffice/lecsus/wsgi.py
Change:
site.addsitedir('/var/www/virtual_env/scioffice/local/lib/python3.7/site-packages')
sys.path.append('/var/www/html/scioffice')
activate_env = '/var/www/virtual_env/scioffice/bin/activate_this.py'

# mkdir /var/www/virtual_env
# python3.7 -m venv /var/www/virtual_env/scioffice
# cd /var/www/virtual_env/scioffice/bin
# ./pip install --upgrade pip
# ./pip install -r /var/www/html/scioffice/conf/requirements.txt
# ./pip install mod_wsgi
# cd /var/www/html/scioffice
# npm install
# npm run build-production
# /var/www/virtual_env/scioffice/bin/python3.7 manage.py collectstatic
# /var/www/virtual_env/scioffice/bin/python3.7 manage.py migrate
# cd /var/www/html
# chown -R www-data:www-data scioffice

# mcedit /etc/apache2/sites-available/scioffice.conf
Paste:
<VirtualHost <server ip>:80>
    ServerName <server hostname>
    ServerAdmin <user email>
    DocumentRoot /var/www/html/scioffice
    LogLevel warn
    ErrorLog ${APACHE_LOG_DIR}/scioffice.error.log
    CustomLog ${APACHE_LOG_DIR}/scioffice.access.log combined
    ErrorDocument 404 /404/

    ProxyPreserveHost On

```

```
ProxyPass "/ws/" "ws://127.0.0.1:8000/ws/"
ProxyPassReverse "/ws/" "ws://127.0.0.1:8000/ws/"
```

```
<Directory /var/www/html/scioffice/>
  Options -Includes -Indexes -ExecCGI +FollowSymLinks +MultiViews
  AllowOverride None
  Require all granted
</Directory>
```

```
Alias /media/ /var/www/html/scioffice/lecsus/media/
Alias /static/ /var/www/html/scioffice/static_collected/
Alias /robots.txt /var/www/html/scioffice/static_collected/robots.txt
Alias /favicon.ico /var/www/html/scioffice/static_collected/favicon.ico
```

```
<Directory /static/>
  Require all granted
</Directory>
<Directory /media/>
  Require all granted
</Directory>
```

```
<Directory /var/www/html/scioffice/lecsus/>
  <Files wsgi.py>
    Require all granted
  </Files>
</Directory>
```

```
WSGIDaemonProcess scioffice processes=2 threads=15
WSGIProcessGroup scioffice
WSGIScriptAlias / /var/www/html/scioffice/lecsus/wsgi.py
```

```
SetOutputFilter DEFLATE
SetEnvIfNoCase Request_URI \.(?:gif|jpe?g|ico|png)$ \ no-gzip dont-vary
SetEnvIfNoCase Request_URI \.(?:exe|t?gz|zip|bz2|sit|rar)$ \ no-gzip dont-vary
SetEnvIfNoCase Request_URI \.pdf$ no-gzip dont-vary
```

```
BrowserMatch ^Mozilla/4 gzip-only-text/html
BrowserMatch ^Mozilla/4\.0[678] no-gzip
BrowserMatch \bMSIE !no-gzip !gzip-only-text/html
```

```
</VirtualHost>
```

```
# cd /var/www/virtual_env/scioffice/bin
# mcedit /etc/apache2/mods-available/wsgi.conf
```

Paste:

```
<IfModule mod_wsgi.c>
```

```
#This config file is provided to give an overview of the directives,
#which are only allowed in the 'server config' context.
```

*#For a detailed description of all available directives please read  
#<http://code.google.com/p/modwsgi/wiki/ConfigurationDirectives>*

*#WSGISocketPrefix: Configure directory to use for daemon sockets.  
#  
#Apache's DEFAULT\_REL\_RUNTIMEDIR should be the proper place for WSGI's  
#Socket. In case you want to mess with the permissions of the directory,  
#you need to define WSGISocketPrefix to an alternative directory.  
#See <http://code.google.com/p/modwsgi/wiki/ConfigurationIssues> for more  
#information*

*#WSGISocketPrefix /var/run/apache2/wsgi*

*#WSGIPythonOptimize: Enables basic Python optimisation features.  
#  
#Sets the level of Python compiler optimisations. The default is '0'  
#which means no optimisations are applied.  
#Setting the optimisation level to '1' or above will have the effect  
#of enabling basic Python optimisations and changes the filename  
#extension for compiled (bytecode) files from .pyc to .pyo.  
#When the optimisation level is set to '2', doc strings will not be  
#generated and retained. This will result in a smaller memory footprint,  
#but may cause some Python packages which interrogate doc strings in some  
#way to fail.*

*WSGIPythonOptimize 1*

*#WSGIPythonPath: Additional directories to search for Python modules,  
#                  overriding the PYTHONPATH environment variable.  
#  
#Used to specify additional directories to search for Python modules.  
#If multiple directories are specified they should be separated by a ':'.  
#*

*#WSGIPythonPath directory|directory-1:directory-2:...*

*#WSGIPythonEggs: Directory to use for Python eggs cache.  
#  
#Used to specify the directory to be used as the Python eggs cache  
#directory for all sub interpreters created within embedded mode.  
#This directive achieves the same affect as having set the  
#PYTHON\_EGG\_CACHE environment variable.  
#Note that the directory specified must exist and be writable by the user  
#that the Apache child processes run as. The directive only applies to  
#mod\_wsgi embedded mode. To set the Python eggs cache directory for*

*#mod\_wsgi daemon processes, use the 'python-eggs' option to the #WSGIDaemonProcess directive instead.*

*#WSGI PythonEggs directory*

*#WSGIRestrictEmbedded: Enable restrictions on use of embedded mode.*

*#*

*#The WSGIRestrictEmbedded directive determines whether mod\_wsgi embedded #mode is enabled or not. If set to 'On' and the restriction on embedded #mode is therefore enabled, any attempt to make a request against a #WSGI application which hasn't been properly configured so as to be #delegated to a daemon mode process will fail with a HTTP internal server #error response.*

*#WSGIRestrictEmbedded On|Off*

*#WSGIRestrictStdin: Enable restrictions on use of STDIN.*

*#WSGIRestrictStdout: Enable restrictions on use of STDOUT.*

*#WSGIRestrictSignal: Enable restrictions on use of signal().*

*#*

*#Well behaved WSGI applications neither should try to read/write from/to #STDIN/STDOUT, nor should they try to register signal handlers. If your #application needs an exception from this rule, you can disable the #restrictions here.*

*#WSGIRestrictStdin On*

*#WSGIRestrictStdout On*

*#WSGIRestrictSignal On*

*#WSGIAcceptMutex: Specify type of accept mutex used by daemon processes.*

*#*

*#The WSGIAcceptMutex directive sets the method that mod\_wsgi will use to #serialize multiple daemon processes in a process group accepting requests #on a socket connection from the Apache child processes. If this directive #is not defined then the same type of mutex mechanism as used by Apache for #the main Apache child processes when accepting connections from a client #will be used. If set the method types are the same as for the Apache #AcceptMutex directive.*

*#WSGIAcceptMutex default*

```
#WSGIImportScript: Specify a script file to be loaded on process start.  
#  
#The WSGIImportScript directive can be used to specify a script file to be  
#loaded when a process starts. Options must be provided to indicate the  
#name of the process group and the application group into which the script  
#will be loaded.
```

```
#WSGIImportScript process-group=name application-group=name
```

```
#WSGILazyInitialization: Enable/disable lazy initialisation of Python.  
#  
#The WSGILazyInitialization directives sets whether or not the Python  
#interpreter is preinitialised within the Apache parent process or whether  
#lazy initialisation is performed, and the Python interpreter only  
#initialised in the Apache server processes or mod_wsgi daemon processes  
#after they have forked from the Apache parent process.
```

```
#WSGILazyInitialization On|Off
```

```
</IfModule>
```

```
# mcedit /etc/apache2/mods-available/wsgi.load
```

```
Paste:
```

```
LoadModule wsgi_module
```

```
"/var/www/virtual_env/scioffice/lib/python3.7/site-packages/mod_wsgi/server/mod_ws  
gi-py37.cpython-37m-x86_64-linux-gnu.so"
```

```
WSGI PythonHome "/var/www/virtual_env/scioffice"
```

```
# a2enmod proxy
```

```
# a2enmod proxy_wstunnel
```

```
# a2enmod wsgi
```

```
# a2ensite scioffice
```

```
# service apache2 reload
```

```
# apt-get install supervisor
```

```
# mkdir /var/log/daphne
```

```
# chown www-data:www-data /var/log/daphne
```

```
unzip
```

```
https://drive.google.com/file/d/15F2Ybe6jKwkQt98N9YgYKYbtXv9UCpHp/view?usp=sharing to /etc/supervisor/conf.d
```

```
copy
```

```
https://drive.google.com/file/d/1kVdb6o81WTSZUwbjRehI9hGJm5B7fKPm/view?usp=sharing to /etc/logrotate.d
```

```
# systemctl disable mongod
```

```
# update-rc.d apache2 disable
```



```
# mkdir /var/log/supervisord
# mkdir /var/log/celery
# reboot
```

## 24. Git Deploy

```
# hostnamectl set-hostname <server hostname>
# pip install virtualenv
# virtualenv /var/www/virtual_env/git_deploy
# cd /var/www
# git clone git@git.scioffice.com:zenkov/git_deploy.git
# cd /var/www/git_deploy
# git fetch
# git merge origin/master

# mcedit /var/www/git_deploy/git_deploy/wsgi.py
Change:
site.addsitedir('/var/www/virtual_env/git_deploy/local/lib/python2.7/site-packages')
sys.path.append('/var/www/html/git_deploy')
sys.path.append('/var/www/html/git_deploy/git_deploy')
activate_env = '/var/www/virtual_env/git_deploy/bin/activate_this.py'

# mcedit /var/www/git_deploy/configs/default_config.json
Change:
"email_host_password": "12345678"
"secret_key": "12345678"
# mcedit /var/www/git_deploy/configs/host_settings.py
Change:
EMAIL_HOST_USER = '<company name>.noreply@gmail.com'
EMAIL_HOST_PASSWORD = '<company noreply email password>'
SECRET_KEY = '<django secret key>'
SITE_PATH = '/var/www/html/scioffice'
SITE_PYTHON_ENV_PATH = '/var/www/virtual_env/scioffice/bin/python'
# mcedit /var/www/git_deploy/scripts/deploy.py
Change:
#!/var/www/virtual_env/git_deploy/bin/python

# cd /var/www/virtual_env/git_deploy/bin
# ./pip install -r /var/www/git_deploy/requirements.txt
# cd /var/www/git_deploy
# /var/www/virtual_env/git_deploy/bin/python2.7 manage.py migrate
# /var/www/virtual_env/git_deploy/bin/python2.7 manage.py collectstatic

# cp -r /var/www/git_deploy/scripts /var/www/git_deploy_updater
# mcedit /var/www/git_deploy_updater/settings.py
Change:
```

```
site_path = '/var/www/git_deploy'  
python_env_path = '/var/www/virtual_env/git_deploy/bin/python'
```

Check:

```
# /var/www/git_deploy_updater/deploy.py
```

Success if no errors in console and there is message "Already up to date"

```
# cd /var/www/git_deploy
```

```
# /var/www/virtual_env/git_deploy/bin/python2.7 manage.py production
```

Success if no errors in console and ends up with "\*\*\*\*\* END \*\*\*\*\*"

```
# /var/www/virtual_env/git_deploy/bin/python2.7 manage.py production
```

Зайдите на сайт <server hostname> под учетной записью "admin" и паролем "admin". Создайте папку, см.рис.1

SciOffice.com

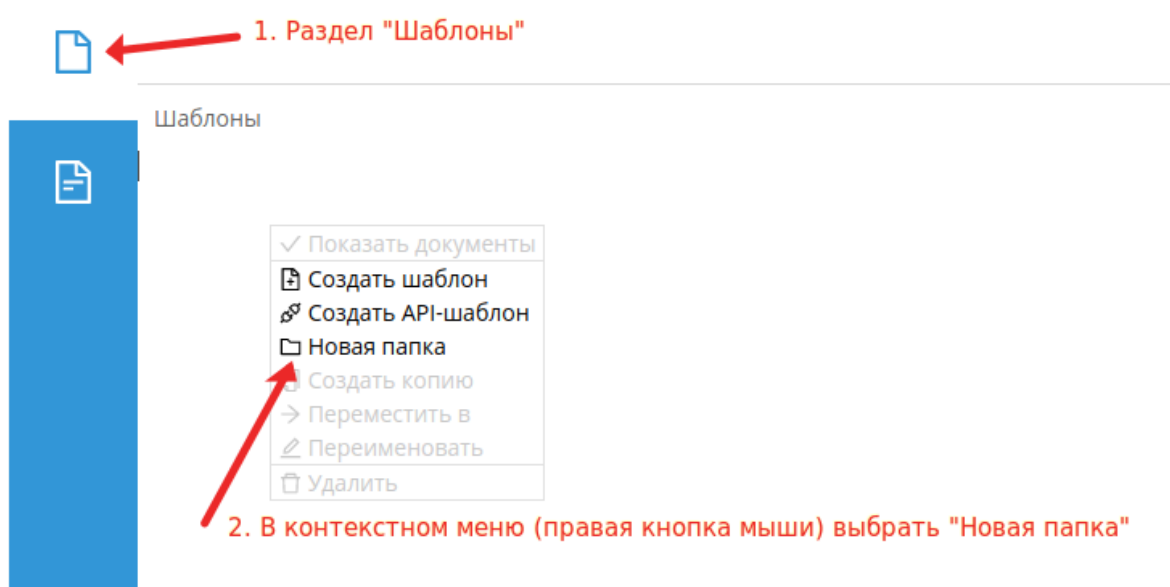


Рисунок 1 - Создание папки

```
#cd /var/www/git_deploy/deploy/temp/backups/refs/heads/master
```

Зайди в папку с именем, совпадающем с текущей датой и запомни имя файла с максимальной цифрой

```
# /var/www/virtual_env/git_deploy/bin/python2.7 manage.py rollback
```

```
refs/heads/master/<current_date_folder>/<last_number_file_name>
```

Обновите страницу <server hostname>, папка должна пропасть - бэкап во время обновления кода работает.

```
# /var/www/virtual_env/git_deploy/bin/python2.7 manage.py backup
```

```
/var/www/html/scioffice
```

В консоли должно появиться "Backup from "/var/www/html/scioffice" to "/var/www/git\_deploy/deploy/temp/backups/scioffice/<current\_date\_folder>/<last\_number\_file\_name>"

Создайте ещё раз папку, как на рис.1

```
# /var/www/virtual_env/git_deploy/bin/python2.7 manage.py rollback
```

```
scioffice/<current_date_folder>/<last_number_file_name>
```

Обновите страницу <server hostname>, папка должна пропасть - бэкап по команде работает.

## 25. SSL

```
# hostnamectl set-hostname <server hostname>
```

```
# dpkg-reconfigure exim4-config
```

Click OK

Set system mail name: <server hostname>

```
# cd var/tmp
```

```
# wget https://dl.eff.org/certbot-auto
```

```
# chmod a+x certbot-auto
```

```
# ./certbot-auto --apache certonly
```

```
# mcedit /etc/apache2/sites-available/scioffice.conf
```

Change port to 443

Add to VirtualHost

```
SSLCertificateFile /etc/letsencrypt/live/<server hostname>/fullchain.pem
```

```
SSLCertificateKeyFile /etc/letsencrypt/live/<server hostname>/privkey.pem
```

```
Include /etc/letsencrypt/options-ssl-apache.conf
```

```
# a2enmod ssl
```

```
# service apache2 reload
```

```
# mcedit /etc/apache2/sites-available/scioffice_aliases.conf
```

Paste:

```
<VirtualHost <server ip>:80>
```

```
    ServerName www.<server hostname>
```

```
    Redirect / https://<server hostname>/
```

```
</VirtualHost>
```

```
<VirtualHost <server ip>:443>
```

```
    ServerName www.<server hostname>
```

```
    Redirect / https://<server hostname>/
```

```
</VirtualHost>
```

```
# a2ensite scioffice_aliases
```

```
# service apache2 reload
```

## 26. Cron tasks

Open *Webmin*->*System*->*Scheduled Cron Jobs*

Click *Create a new scheduled cron job*

Create jobs:

1. RkHunter

*Execute cron job as: root*

*Command: /usr/local/bin/rkhunter --cronjob --update --quiet*

*Description: Rootkit hunter (<server hostname>)*

*When to execute: Daily (at midnight)*

## 2. Backup

*Execute cron job as: root*

*Command: /var/www/virtual\_env/git\_deploy/bin/python*

*/var/www/git\_deploy/manage.py backup /var/www/html/scioffice*

*Description: Backup of scioffice (<server hostname>)*

*When to execute: Daily (at midnight)*

## 3. Check data integrity

*Execute cron job as: www-data*

*Command: /var/www/virtual\_env/scioffice/bin/python*

*/var/www/html/scioffice/manage.py check\_data\_integrity*

*Description: Check data integrity of SciOffice Documents and Templates (<server hostname>)*

*When to execute: Daily (at midnight)*

## 4. Check copy transaction

*Execute cron job as: www-data*

*Command: /var/www/virtual\_env/scioffice/bin/python*

*/var/www/html/scioffice/manage.py check\_copy\_transaction\_is\_ok*

*Description: Check transaction of SciOffice copy master is ok (<server hostname>)*

*When to execute: Daily at 01:00*

## 5. Check export transaction

*Execute cron job as: www-data*

*Command: /var/www/virtual\_env/scioffice/bin/python*

*/var/www/html/scioffice/manage.py check\_batch\_export\_transaction\_is\_ok*

*Description: Check transaction of SciOffice batch exports is ok (<server hostname>)*

*When to execute: Daily at 02:00*

## 6. Check import transaction

*Execute cron job as: www-data*

*Command: /var/www/virtual\_env/scioffice/bin/python*

*/var/www/html/scioffice/manage.py check\_batch\_import\_transaction\_is\_ok*

*Description: Check transaction of SciOffice batch imports is ok (<server hostname>)*

*When to execute: Daily at 03:00*

## 7. Clean export temporary files

*Execute cron job as: www-data*

*Command: /var/www/virtual\_env/scioffice/bin/python*

*/var/www/html/scioffice/manage.py clear\_tmp\_of\_batch\_export*

*Description: Remove \*.sci files created when there was an export (<server hostname>)*

*When to execute: Daily at 04:00*

## 8. SSL certificate update

*Execute cron job as: root*

*Command: certbot-auto renew --pre-hook "service apache2 stop" --post-hook "service apache2 start"*

*Description: Update ssl certificate letsencrypt.org (<server hostname>)*

*When to execute: Monthly (on the 1st)*

#### 9. CPU monitor

*Execute cron job as = root*

*Command = /root/scripts/monitor/cpu-alert.sh*

*Description = CPU monitor*

*Config run every 5 minutes*

#### 10. RAM monitor

*Execute cron job as = root*

*Command = /root/scripts/monitor/memory-alert.sh*

*Description = RAM monitor*

*Config run every 5 minutes*

## 27. Update OS

```
# apt-get update
```

```
# apt-get -V upgrade
```

```
# updatedb
```

```
# reboot
```

Config <https://uptimerobot.com/>

## 28. Security

```
# update-rc.d webmin disable
```

```
# mcedit /etc/ssh/sshd_config
```

if all server admins was created:

```
Set PermitRootLogin no
```

```
Set AllowUsers <user name> <another user name>
```

```
Set X11Forwarding no
```

```
# systemctl restart ssh.service
```

```
# reboot
```

## Links

<https://www.pontikis.net/blog/debian-jessie-web-server-setup>

<https://docs.mongodb.com/v3.6/tutorial/install-mongodb-on-debian/>

<https://docs.mongodb.com/v3.6/tutorial/enable-authentication/>