

The Long Title

Self-documenting
Python
Xmlrpc server
on Debian VPS
With nginx
And SSL

VPS

Virtual Private Server.

rapidvps.com 19.99/month
Or fsockvps.com 9.99/month
Or whatever other you like.

Get image of Debian Stable

Advantage: root.

File Edit View History Bookmarks Tools Tabs Help Bookmarks

http://rapidvps.com/index.php?page=Management.Reinstall

asana D... My Account Facebook | ... How import... 1&1 1&1 Control... http://...est.py TC TORN My C

[9893677] box1.kissws.com

Reinstalling your OS is currently limited to two events per 24 hours with no overall limit.

Step 1/2 :: Confirm what you are about to do. Waive liability of RapidVPS.

You are about to reinstall your VPS operating system. This is a convenient but potentially devastating feature of our system currently have on your VPS, will be **ANNIATED**. If there is any data on this server which you need and do not have a backup your local machine (or other destination) and restart this process. The server which will be deleted has the following details:

Hostname: box1.kissws.com

IPs: 208.77.100.96 208.77.100.107

Select new Operating System:

- CentOS 4.6
- CentOS 4.6
- CentOS 5
- CentOS 5 + Jaxer
- Debian 3.1
- Debian 4
- Debian 5**
- Fedora Core 4
- Fedora Core 5

Click the button below for instructions on how to backup your current VPS data of server **box1.kissws.com 208.77.100.96** and the place.

Destroy Server and Continue

Debian

Upgrade to latest Debian stable (should be version 5, lenny)

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

The default python should be 2.5.2

```
box1:~# python  
Python 2.5.2 (r252:60911, Jan  4 2009, 17:40:26)  
[GCC 4.3.2] on linux2  
Type "help", "copyright", "credits" or "license" for more  
information.  
>>>
```

Nginx

What is nginx?

- Very small, very fast web server.
- No CGI support
- Good reverse-proxy
- Supports ssl

`apt-get install nginx`

Config:

`/etc/nginx/nginx.conf` (we'll go back to that!)

Python

```
# mkdir /scripts/
```

```
#cd /scripts/
```

Create file server.py

(the service)

Create file startserver.py

(starts/ restarts the server.py)

startserver.py

```
#!/usr/bin/python2.5

import os
import time

server_pid = os.popen("ps -C server.py -o pid=").read().strip()

print "server_pid: ", server_pid

if server_pid.isdigit():
    killed = os.popen("kill " + server_pid).read()
    print "killed server: " , killed

server_started = os.system(" nohup /scripts/server.py & ")
print "server_started: " , server_started

time.sleep(2)

reload_page = os.popen('wget -q https://kissws.com/ -O /scripts/index.html --no-check-
certificate').read()

delete_index_html = os.popen('rm /scripts/index.htm*').read()
```

DocXMLRPCServer

Module in standard library:

<http://docs.python.org/library/docxmlrpcserver.html>

Simple to use.

GET returns html description of docstrings
POST provides xmlrpc service


```
#!/usr/bin/python2.5
```

simple_server.py

```
import DocXMLRPCServer
import os
```

```
server = DocXMLRPCServer.DocXMLRPCServer(("", 18002))
server.register_introspection_functions()
print server.get_request()
```

```
server.set_server_title("title")
server.set_server_name("servername")
serverdoc = """description goes here"""
server.set_server_documentation(serverdoc)
```

```
class Allfunctions:
    def server_uptime(self):
        """Return the result of the unix uptime command"""
        return os.popen("uptime").read()

    def status(self):
        """Return 'ok' when this is working."""
        return 'ok'
```

```
server.register_instance(Allfunctions())
server.serve_forever()
```



servername

description goes here

Methods

server_uptime()

Return the result of the unix uptime command

status()

Return 'ok' when this is working.

system.listMethods()

```
system.listMethods() => ['add', 'subtract', 'multiple']
```

Returns a list of the methods supported by the server.

system.methodHelp(method_name)

```
system.methodHelp('add') => "Adds two integers together"
```

Returns a string containing documentation for the specified method.

Test from my winXP machine

```
Shortcut to python.exe
Python 2.6.3 (r263rc1:75186, Oct 2 2009, 20:40:30) [MSC v.1500 32 bit (Intel)]
on win32
Type "help", "copyright", "credits" or "license" for more information.
>>> import xmlrpclib
>>> s = xmlrpclib.ServerProxy(r'http://kissws.com:18002')
>>> print s.status()
ok
>>> print s.server_uptime()
05:23:44 up 96 days, 22:26, 2 users, load average: 0.00, 0.00, 0.00
>>>
```

server.py (1)

```
kissws.com - PuTTY
GNU nano 2.0.7 File: server.py
#!/usr/bin/python

import DocXMLRPCServer
import os
import datetime
import sqlite3
import sys
#import psyco
#psyco.full()

server = DocXMLRPCServer.DocXMLRPCServer(("127.0.0.1", 8001))
server.register_introspection_functions()
print server.get_request()
```

server.py (2)

```
class Allfunctions:
    __log_file_sub_path = r'./logfiles/'
    __queue_file_sub_path = r'./queues/'
    __list_file_sub_path = r'./lists/'

    def server_uptime(self):
        """Return the result of the unix uptime command"""
        return os.popen("uptime").read()

    def status(self):
        """Return 'ok' when this is working."""
        return 'ok'
```

server.py (3)

```
def list_create(self, list_name):
    """Create a new list"""
    list_name = self.__clean_list_name(list_name)
    list_file = self.__list_file_sub_path + list_name
    conn = sqlite3.connect(list_file)
    cur = conn.cursor()
    sql = 'create table list ([item text]);'
    cur.execute(sql)
    conn.commit()
    cur.close()
    conn.close()
    return 'ok'
```

```
def list_append(self, list_name, list_item_value):
    """Append item to list. The list must already exist."""
    list_name = self.__clean_list_name(list_name)
    list_file = self.__list_file_sub_path + list_name
    if self.__list_exists(list_name):
        conn = sqlite3.connect(list_file)
        cur = conn.cursor()
        list_data_list = (list_item_value,)
        cur.execute('insert into list values (?);', list_data_list)
        conn.commit()
        cur.close()
        conn.close()
        result = "ok"
    else:
        result = "List not found."
    return result
```

```
def __clean_list_name(self, list_name):
    """Clean the list name."""
    new_list = ""
    if not list_name.isalnum():
        for c in list_name:
            new_c = c
            if not c.isalnum():
                new_c = "_"
            new_list = new_list + new_c
    else:
        new_list = list_name
    return new_list
```

SQLite data

```
box1:/scripts/lists# ls -lah
total 36K
drwxr-xr-x  2 root root 4.0K Oct  8 05:36 .
drwxr-xr-x  6 root root 4.0K Oct  8 06:02 ..
-rw-r--r--  1 root root 2.0K Oct  8 05:36 chris_todo
-rw-r--r--  1 root root 2.0K Oct  5 04:38 mumbai_special4
-rw-r--r--  1 root root 2.0K Oct  5 04:38 mumbai_speciar14
-rw-r--r--  1 root root 2.0K Oct  5 04:23 mumbaispecial1
-rw-r--r--  1 root root 2.0K May 15 17:07 test
-rw-r--r--  1 root root 2.0K Oct  5 04:02 test2
-rw-r--r--  1 root root 2.0K Oct  5 04:02 test4
box1:/scripts/lists# █
```

```
kissws.com - PuTTY
box1:/scripts/lists# sqlite3 chris_todo
SQLite version 3.5.9
Enter ".help" for instructions
sqlite> .tables
list
sqlite> .schema list
CREATE TABLE list (item text);
sqlite> select * from list;
there is something to do
there is something else
there is something other
I want bacon!
sqlite> .quit
box1:/scripts/lists# █
```

server.py list_view function

```
def list_view(self, list_name):
    """Return the list."""
    list_name = self.__clean_list_name(list_name)
    list_file = self.__list_file_sub_path + list_name
    if os.path.exists(list_file):
        conn = sqlite3.connect(list_file)
        cur = conn.cursor()
        sql = 'select rowid, item from list;'
        cur.execute(sql)
        rows = cur.fetchall()
        result_list = []
        for item_set in rows:
            result_list.append(str(item_set[0]) + ': ' + str(item_set[1]) + "\n")
        result = ''.join(result_list)
        cur.close()
        conn.close()
    else:
        result = "List not found."
    return result
```

```
>>> print s.list_view('chris_todo')
1: there's something to do
2: there is something else
3: there is something other
4: I want bacon!
```

```
>>>
```

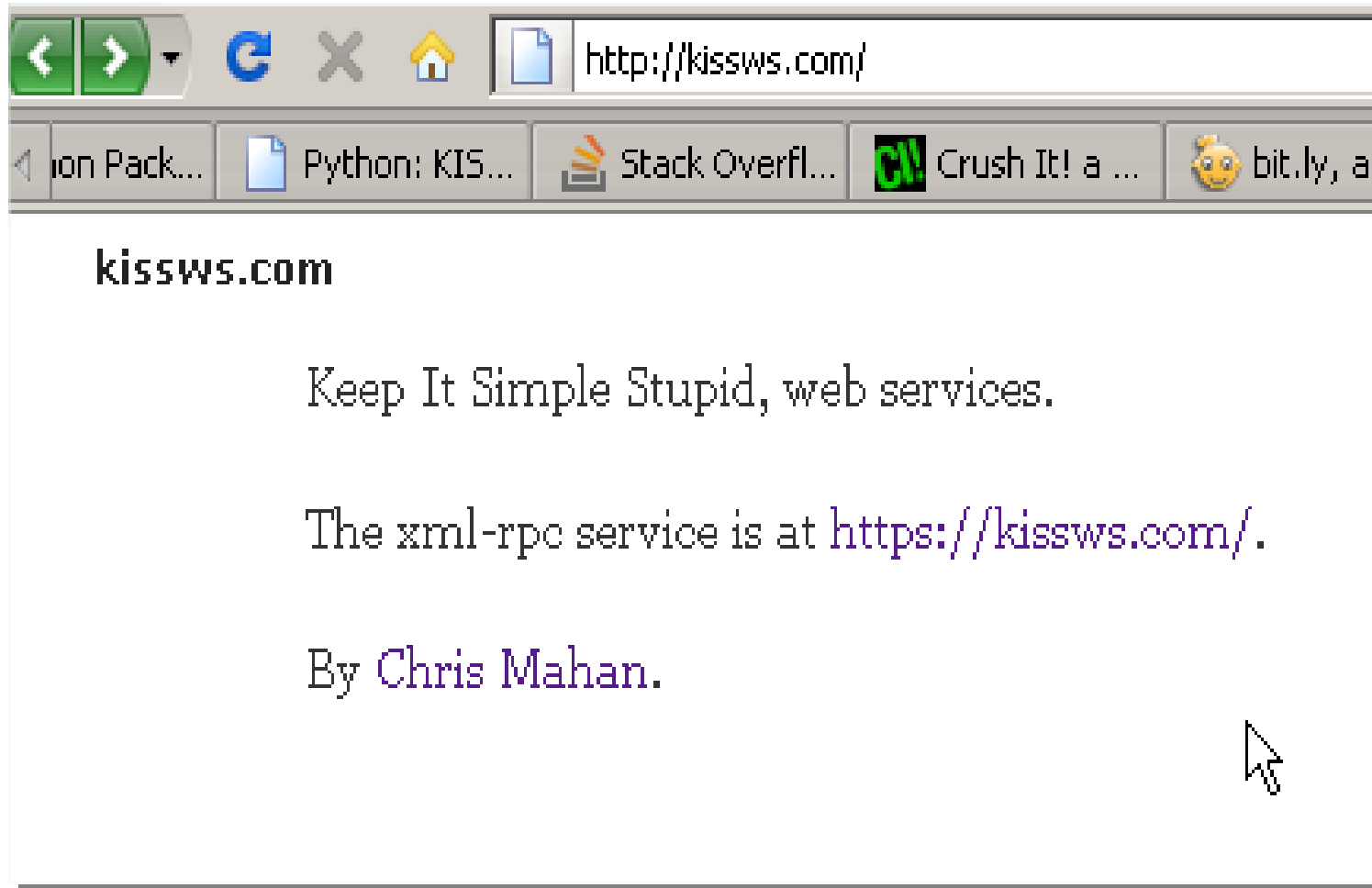

nginx.conf

```
server {
    listen 443;
    ssl on;
    ssl_certificate /etc/nginx/ssl/server.crt;
    ssl_certificate_key /etc/nginx/ssl/server.key;

    server_name kissws.com www.kissws.com;
    location /gtapi/ {
        proxy_pass http://127.0.0.1:8008/;
        proxy_redirect default;
    }

    # kissws services, proxies /scripts/server.py
    location / {
        proxy_pass http://127.0.0.1:8001/;
        proxy_redirect default;
    }
}
server {
    listen 80;
    server_name kissws.com www.kissws.com;
    location / {
        index index.xml;
        root /var/www/kissws.com/;
    }
    location /ws/ {
        proxy_pass http://127.0.0.1:8001/;
        proxy_redirect default;
    }
}
```

Standard web page at port 80



Encrypted Xmlrpc service at port 443



Browser address bar: `https://kissws.com/`

Browser tabs: sh It! a ..., bit.ly, a sim..., Jp Calderone, Python Pack..., Python Pack..., Aqu...

kissws.com: Keep It Simple, Stupid! Web Service

Some simple web services, all in xmlrpc

Notice: the stuff is no longer under development as of July 1 2009. I'm g

There is an issue with restarting the service when the VM goes down. The

Contact: `chris.mahan@gmail.com`

Using python 2.5.2 on debian stable with psyco. No other dependencies exc

Methods

- list_append**(list_name, list_item_value)
Append item to list. The list must already exist.
- list_create**(list_name)
Create a new list
- list_delete**(list_name, force=False)
Delete an empty list. Pass True to parameter force to delete
- list_drop**(list_name, list_item_number=0)
Delete item from list. The list must already exist.
- list_exists**(list_name)
Check if list exists.

DNS Settings at dnsmadeeasy.com

The screenshot shows a web browser window with the URL <https://www.dnsmadeeasy.com/servlet/domainlist>. The browser's address bar and tabs are visible. The main content area displays the DNS settings for the domain **kissws.com**. At the top right, there is a link to [Remove all Records](#).

NS Records (6)

Name	Value (nameserver)	
kissws.com.	ns10.dnsmadeeasy.com.	N / A
kissws.com.	ns11.dnsmadeeasy.com.	N / A
kissws.com.	ns12.dnsmadeeasy.com.	N / A
kissws.com.	ns13.dnsmadeeasy.com.	N / A
kissws.com.	ns14.dnsmadeeasy.com.	N / A
kissws.com.	ns15.dnsmadeeasy.com.	N / A

A Records (1)

Name	IP	FO / SM		
kissws.com.	208.77.100.96	Off / Off	N/A	Remove

CNAME Records (1)

Name	Value (alias to)	
www	kissws.com.	Remove

MX Records (0)

Mail for this host	Sent to this server	MX Level	
There are no MX records configured.			

Domain Registrar Settings

The screenshot shows the GoDaddy Domain Manager interface. At the top, there's a navigation bar with 'My Account > DOMAINMANAGER'. Below it, there are buttons for 'Domains', 'Buy/Sell', 'Tools', and 'Help'. A search bar contains 'KISSWS.COM' with a 'Go' button and navigation arrows. The main content area is titled 'Set Nameservers'. It features a grid of input fields for nameservers 1 through 13. Nameservers 1, 2, 5, and 6 are pre-filled with 'NS10.DNSMADEEASY.COM', 'NS11.DNSMADEEASY.COM', 'NS14.DNSMADEEASY.COM', and 'NS15.DNSMADEEASY.COM' respectively. Nameservers 3, 4, 7, 8, 9, 10, 11, 12, and 13 are empty. The interface also includes a sidebar with 'Organize' and 'Domain' options.

Go Daddy My Account > DOMAINMANAGER

Domains Buy/Sell Tools Help

Back KISSWS.COM Go

Set Nameservers

Nameserver 1: *	Nameserver 2: *	Nameserver 3:
<input type="text" value="NS10.DNSMADEEASY.COM"/>	<input type="text" value="NS11.DNSMADEEASY.COM"/>	<input type="text" value="NS12.DNSMADEEASY.COM"/>
Nameserver 5:	Nameserver 6:	Nameserver 7:
<input type="text" value="NS14.DNSMADEEASY.COM"/>	<input type="text" value="NS15.DNSMADEEASY.COM"/>	<input type="text"/>
Nameserver 9:	Nameserver 10:	Nameserver 11:
<input type="text"/>	<input type="text"/>	<input type="text"/>
Nameserver 13:		
<input type="text"/>		

Point your name servers to DNSMadeEasy.com

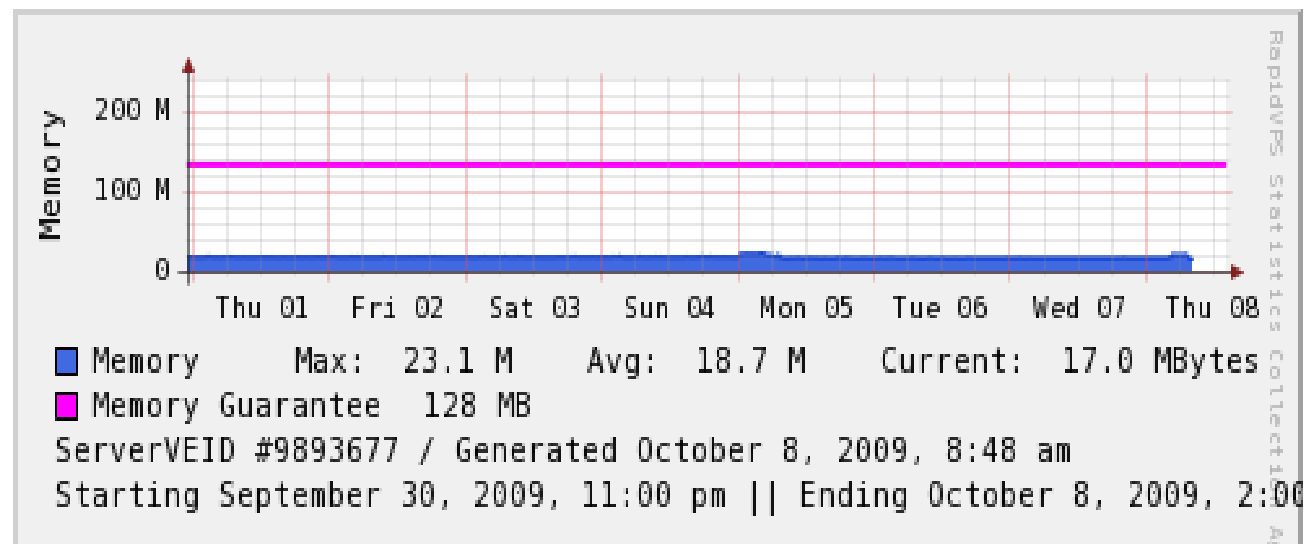
Low memory usage. Easily fits in 128 MB RAM

RapidVPS Resource Usage History

window. Sta
and are avai
using VPS tec

[9893677] box1.kissws.com Memory Usage View

From: 2009 October 01 To: 2009 October 08



Questions?

Thank you

Questions & comment:

chris.mahan@gmail.com

(818) 671-1709

<http://christophermahan.com/>

This document is at

<http://kissws.com/socal-piggies20091008.pdf>