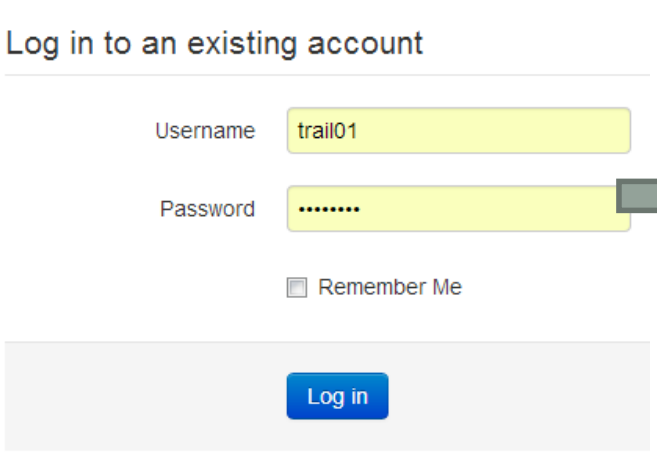
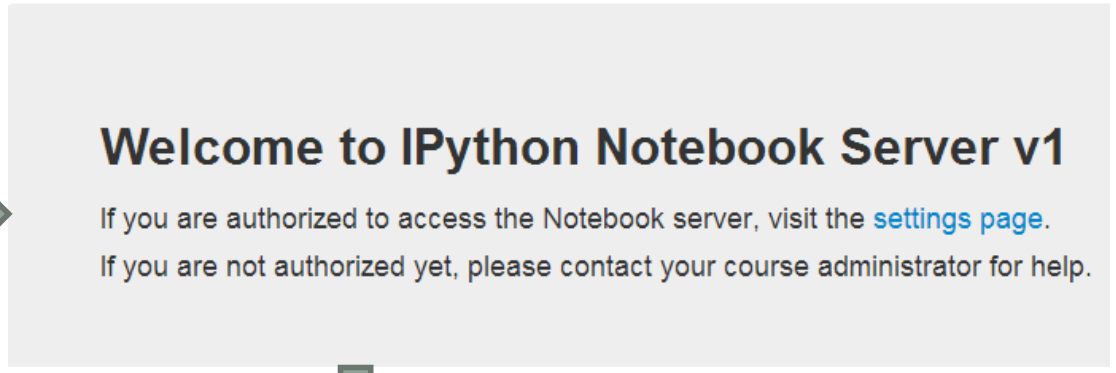
IPYTHON SERVICE OVERVIEW

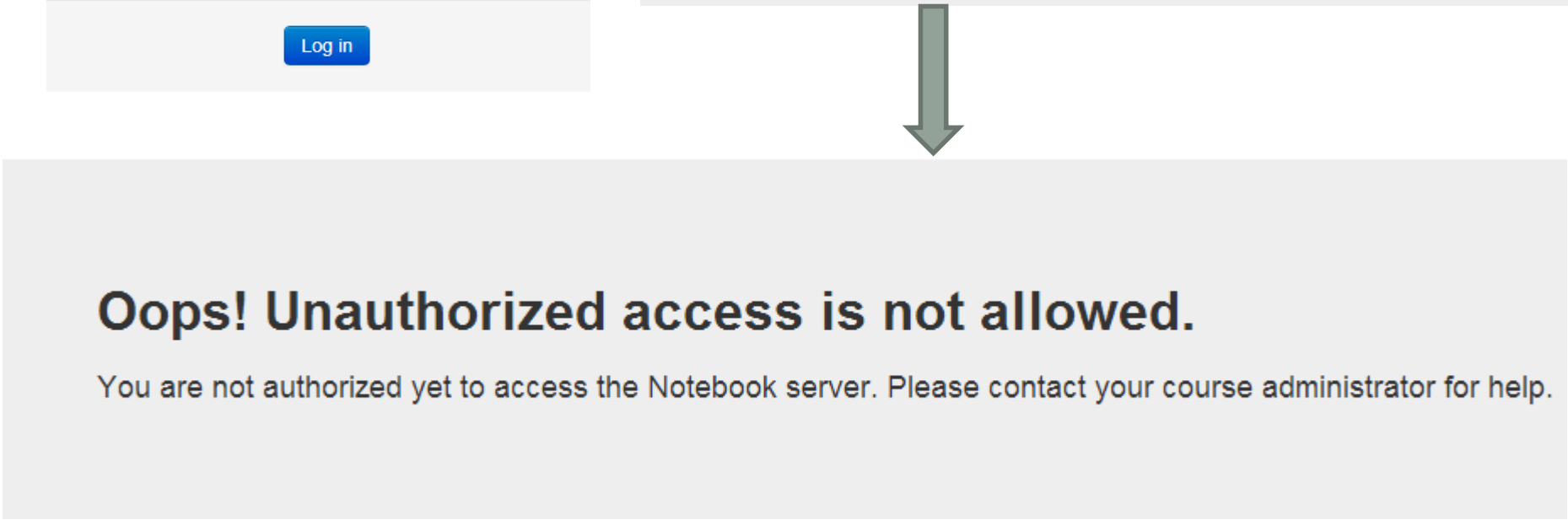
M O Faruque Sarker
f.sarker@ucl.ac.uk
23/10/2013

Service Overview: End-user's Perspective

- IPython service facilitates accessing IPython notebook servers from an Internet browser
  - Users can login using their UCL Login on https://ipython.ucl.ac.uk/
  - After the first time login, Site/Course Admin needs to approve access to every user for accessing a particular Notebook server
  - Site/Course Admin also ensures that the particular Notebook servers are running
  - All approved users can see their Notebook server access information if they are approved AND their individual Notebook servers are running


Log in to an existing account
Username
trail01
Password
Remember Me
Log in


Welcome to IPython Notebook Server v1
If you are authorized to access the Notebook server, visit the settings page.
If you are not authorized yet, please contact your course administrator for help.


Oops! Unauthorized access is not allowed.
You are not authorized yet to access the Notebook server. Please contact your course administrator for help.

Follow these steps to run your IPython Notebook server

1. Copy your Notebook server password:

Click the button below to copy the password into your sytem clipboard (if your browser doesn't have the Flash plug-in installed, use your mouse).

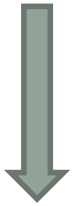
z\czxxzvxcvbxcbx

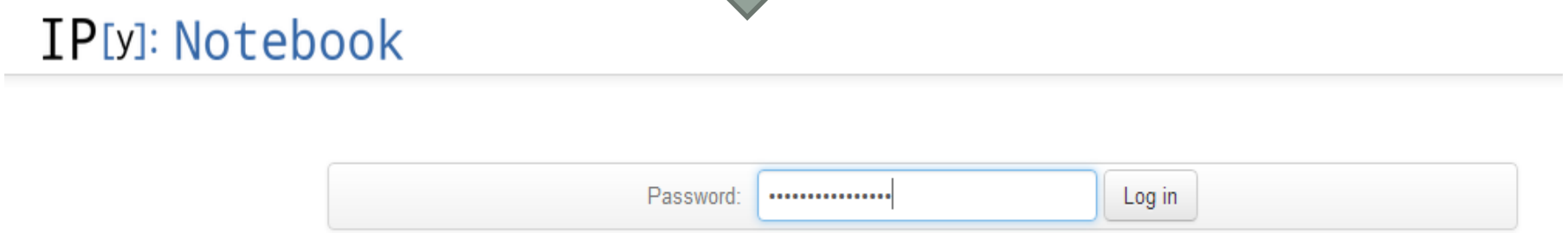
2. Vist your Notebook server URL: http://ipython.ucl.ac.uk:9004

On that page, paste the password in the input box and click login. Then enjoy working on your IPython notebooks!

3. Remember to log out!

When you finish working, always log out from your Notebook server by clicking the logout button on the top of your Notebook server's landing page. Also don't forget to log out from the homepage.





Service Overview: End-user's Perspective (contd.)

- After the first time login and approval of access by Site Admin end-users can always find their personal Notebook server access information from the settings page: https://ipython.ucl.ac.uk/account/settings/
  - Copy the password of the Notebook server
  - Click the provided Notebook server URL, e.g.:

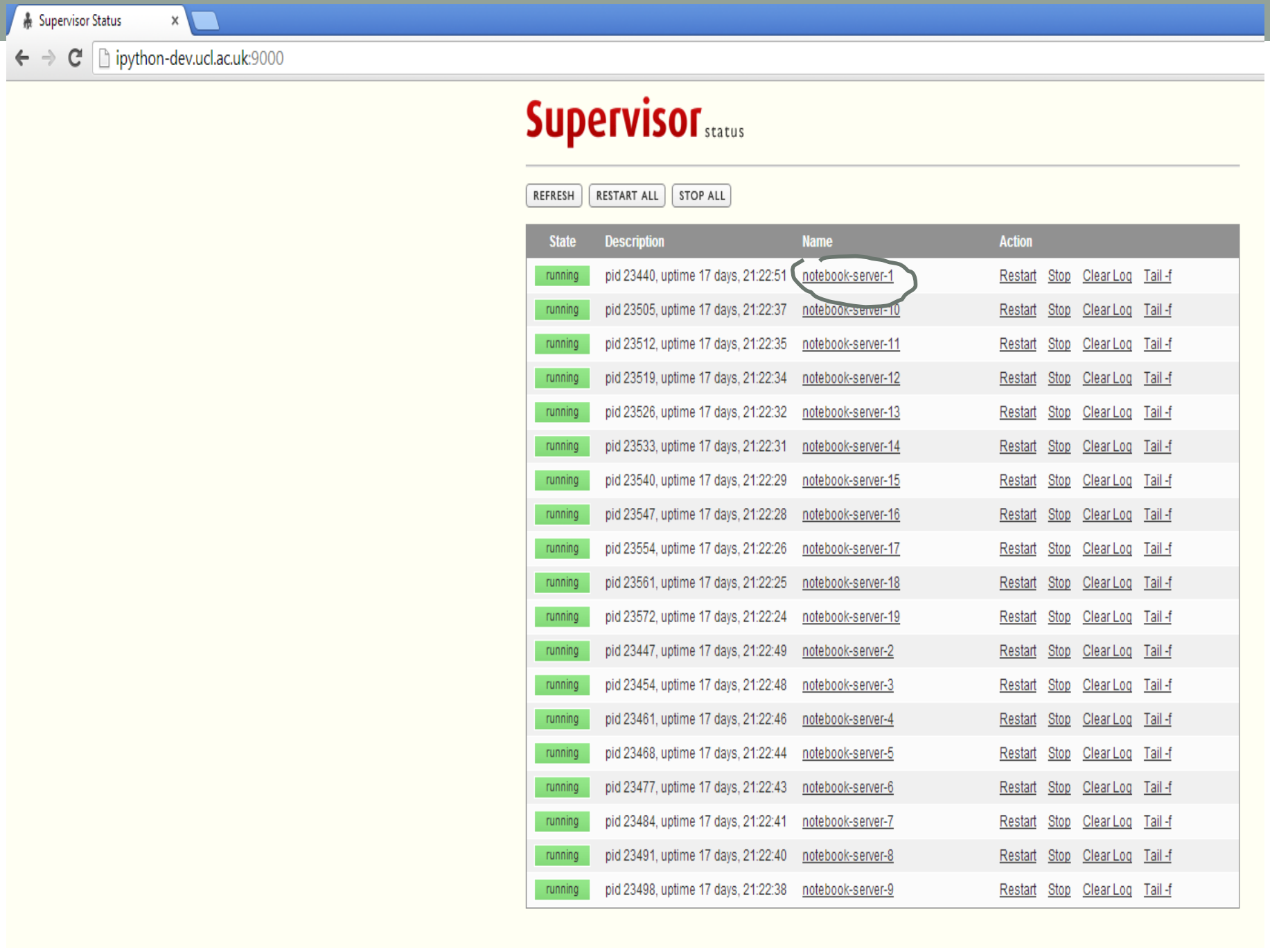
    http://ipython.ucl.ac.uk:9004
  - The Notebook server's front-end web interface can be accessed in the browser's another tab/window.
- End-users are responsible for protecting their Notebook server's password, use it fairly by launching limited number of Notebooks and shut-down Notebooks when they are not-in-use to minimize the overall memory usage.

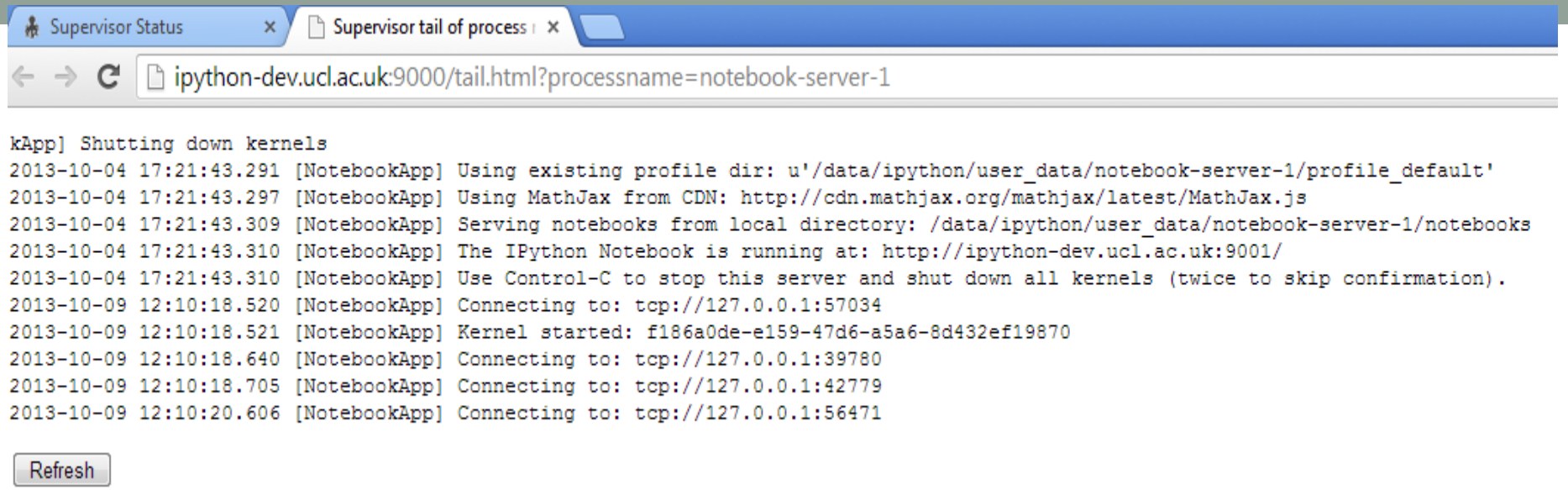
Service overview: Site/Course Admin's Perspective

- Site admins need to check that all notebook servers are up and running from a web interface of Supervisor at:

  http://ipython.ucl.ac.uk:9000

  - Notebook server's start-up log message should be checked
  - Problematic Notebook servers should be restarted
  - Unused/spare notebook servers should be stopped to get more working memory


Supervisor Status
ipython-dev.ucl.ac.uk:9000
Supervisor status
REFRESH
RESTART ALL
STOP ALL
State
Description
Name
Action
running
pid 23440, uptime 17 days, 21:22:51
notebook-server-1
Restart Stop Clear Log Tail -f
running
pid 23505, uptime 17 days, 21:22:37
notebook-server-10
Restart Stop Clear Log Tail -f
running
pid 23512, uptime 17 days, 21:22:35
notebook-server-11
Restart Stop Clear Log Tail -f
running
pid 23519, uptime 17 days, 21:22:34
notebook-server-12
Restart Stop Clear Log Tail -f
running
pid 23526, uptime 17 days, 21:22:32
notebook-server-13
Restart Stop Clear Log Tail -f
running
pid 23533, uptime 17 days, 21:22:31
notebook-server-14
Restart Stop Clear Log Tail -f
running
pid 23540, uptime 17 days, 21:22:29
notebook-server-15
Restart Stop Clear Log Tail -f
running
pid 23547, uptime 17 days, 21:22:28
notebook-server-16
Restart Stop Clear Log Tail -f
running
pid 23554, uptime 17 days, 21:22:26
notebook-server-17
Restart Stop Clear Log Tail -f
running
pid 23561, uptime 17 days, 21:22:25
notebook-server-18
Restart Stop Clear Log Tail -f
running
pid 23572, uptime 17 days, 21:22:24
notebook-server-19
Restart Stop Clear Log Tail -f
running
pid 23447, uptime 17 days, 21:22:49
notebook-server-2
Restart Stop Clear Log Tail -f
running
pid 23454, uptime 17 days, 21:22:48
notebook-server-3
Restart Stop Clear Log Tail -f
running
pid 23461, uptime 17 days, 21:22:46
notebook-server-4
Restart Stop Clear Log Tail -f
running
pid 23468, uptime 17 days, 21:22:44
notebook-server-5
Restart Stop Clear Log Tail -f
running
pid 23477, uptime 17 days, 21:22:43
notebook-server-6
Restart Stop Clear Log Tail -f
running
pid 23484, uptime 17 days, 21:22:41
notebook-server-7
Restart Stop Clear Log Tail -f
running
pid 23491, uptime 17 days, 21:22:40
notebook-server-8
Restart Stop Clear Log Tail -f
running
pid 23498, uptime 17 days, 21:22:38
notebook-server-9
Restart Stop Clear Log Tail -f


Supervisor Status
Supervisor tail of process
ipython-dev.ucl.ac.uk:9000/tail.html?processname=notebook-server-1
kApp] Shutting down kernels
2013-10-04 17:21:43.291 [NotebookApp] Using existing profile dir: u'/data/ipython/user_data/notebook-server-1/profile_default'
2013-10-04 17:21:43.297 [NotebookApp] Using MathJax from CDN: http://cdn.mathjax.org/mathjax/latest/MathJax.js
2013-10-04 17:21:43.309 [NotebookApp] Serving notebooks from local directory: /data/ipython/user_data/notebook-server-1/notebooks
2013-10-04 17:21:43.310 [NotebookApp] The IPython Notebook is running at: http://ipython-dev.ucl.ac.uk:9001/
2013-10-04 17:21:43.310 [NotebookApp] Use Control-C to stop this server and shut down all kernels (twice to skip confirmation).
2013-10-09 12:10:18.520 [NotebookApp] Connecting to: tcp://127.0.0.1:57034
2013-10-09 12:10:18.521 [NotebookApp] Kernel started: f186a0de-e159-47d6-a5a6-8d432ef19870
2013-10-09 12:10:18.640 [NotebookApp] Connecting to: tcp://127.0.0.1:39780
2013-10-09 12:10:18.705 [NotebookApp] Connecting to: tcp://127.0.0.1:42779
2013-10-09 12:10:20.606 [NotebookApp] Connecting to: tcp://127.0.0.1:56471
Refresh

Service Overview: Site/Course Admin's Perspective (contd.)

- Site admins also need to approve/enable users to access Notebook servers from the Admin panel: https://ipython.ucl.ac.uk/admin/ipysite/userprofile/
  - By selecting All or individual users and applying right action from the Action dropdown box
  - Inactive users need to be detected and corresponding Notebook servers should be stopped

https://ipython-dev.ucl.ac.uk/admin/ipysite/userprofile/

Django administration

Home › Ipysite › Notebook Server User Settings

Select Notebook Server User Setting to change

Action: --------- Go 0 of 7 selected

Username	First name	Last name	Email
ucapdrb	David R	Bowler	david.bowler@ucl.ac.uk
zcapg97	Kathryn R	Newbould	kathryn.newbould.10@ucl.ac.uk
zcaph23	Martin	Buettner	martin.buettner.11@ucl.ac.uk
trail01			trail01@ucl.ac.uk
cceahpa	Harry	Panayiotou	h.panayiotou@ucl.ac.uk
nbadmin			lta-team@ucl.ac.uk
cceamof	M O Faruque	Sarker	f.sarker@ucl.ac.uk

7 Notebook Server User Settings

Add Notebook Server User Setting

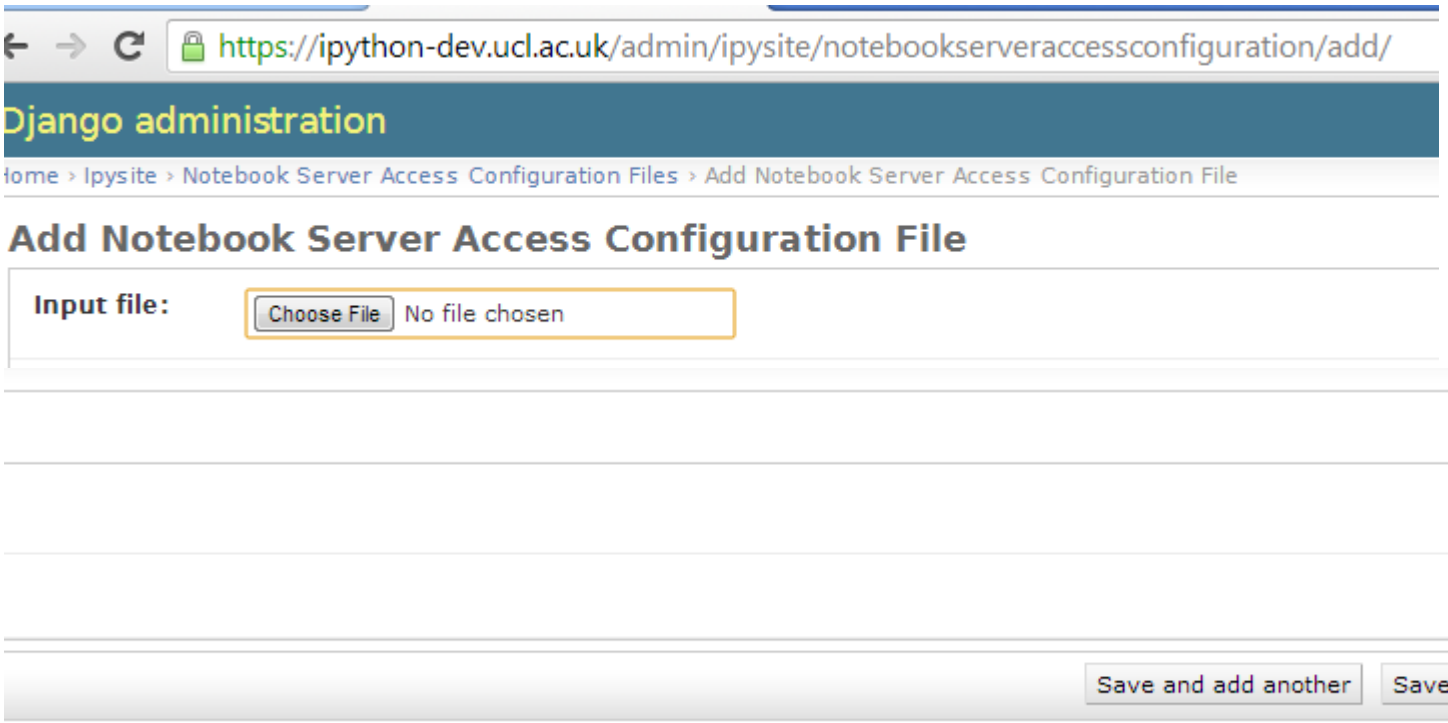
Notebook server port	Notebook server password	Can access Notebook server
9006	76ALQVKK3U270BB0	✓
9008	ERVT3Y3ROK2OD50H	✓
9007	KNITWCOU9YABWXLD	✓
9005	67VBL7I043RH6U7Z	✓
9003	0GLRBQBYHTJIQ2DT	✓
9001	UXM0EP0N8RGUBTQC	✓
9002	AV7A7HYQVBQ1796D	✓

Service Overview: Site/Course Admin's Perspective (contd.)

- For enabling Notebook server access to many users at once, CSV/TXT files (containing one username per line) can be uploaded from Admin interface:

  https://ipython.ucl.ac.uk/admin/ipysite/notebookserveraccessconfiguration/

  - Add a 'Notebook Server Access Configuration File'
  - Select a file and type of action: enable/disable
  - Some more admin actions are available: disable all users, show list of enabled users etc.
  - List of users can be extracted from a Moodle course's Grade report (export as CSV) and deleting unnecessary fields


https://ipython-dev.ucl.ac.uk/admin/ipysite/notebookserveraccessconfiguration/add/
Django administration
Home › Ipysite › Notebook Server Access Configuration Files › Add Notebook Server Access Configuration File
Add Notebook Server Access Configuration File
Input file:
Choose File
No file chosen
Save and add another
Save and continue editing
Save

Application Administration Overview

- Managing Supervisord
  - Config and log files are inside /data/ipython/supervisord
  - All Notebook servers process configuration files are inside /data/ipython/supervisord/conf.d/ (one file per server)
  - Run-time configuration of each Notebook server is kept on it's process configuration file
- Finding Logs
  - Supervisord: /data/ipython/supervisord/supervisord.log
  - Each Notebook server's log can be viewed either from Supervisord web interface or /data/ipython/user_data/notebook-server-X/logs

Application Administration Overview…

- Adding more Notebook servers
  - Use supplied scripts under utils/* to generate Notebook server process configuration files . The generate_entry() function on generate_all_nbserver_config.py should be used to generate a config file. Alternately use Fabric script setup_notebook_configs() from deploy/fabfile.py
  - Create Notebook server data directories under /data/ipython/user_data/
  - Add more Notebook server process configuration files under /data/ipython/supervisord/conf.d and then restart Supervisord (make sure all Notebook servers are properly stopped before the restart)
  - Ensure that data on each Notebook server configuration process matches exactly as listed on /data/ipython/shared_config_files/all_nbserver_config.csv. This file will be read when users are logged in to load the specific settings (Notebook server password and port) on the Application Admin interface
  - Test the configuration by running the Notebook server and logon to it using the generated password
  - After the first time run of a Notebook server, create a Notebook server profile file to it's profile path

    e.g. /data/ipython/user_data/notebook-server-XXX/profile_default/ipython_config.py

  And edit the ipython_dir variable accordingly.

Application Administration Overview…

- Installing a new Python package
  - Activate virtualenv and run pip
    - source /data/ipython/virtualenvs/supervised-ipython-nbserver/bin/activate
    - pip install <package_name>
  - Often Linux OS dependant packages needs to be installed separately
- Application source code: https://github.com/writefaruq/supervised-ipython-nbserver

Some Assumptions/Limitations

- Supervisord has launched all Notebook servers and they are running on the designated ports (Port No = 9000 + Notebook server ID)
- Server has enough memory to run all Notebook servers and notebook processes
- Notebook servers settings are controlled via the command line arguments and profile configurations
- Python virtual environment has necessary packages installed
- While using CSV file uploads, usernames present in that CSV file have already logged into the site (and their user profile data is populated)

Possible Common Issues

- User is not authorized to access the Notebook server
  - Site/Course admin hasn't enable access yet
- Notebook server(s) is not running
  - Supervisord isn't running (Check Supervisord web interface and Nagios alerts). This can happen after the machine reboot if Linux initialization scripts failed to run.
  - Notebook server is running on different port: If the designated port is already used by some other application, Notebook server will be launched on a random port. In order to fix this, all notebook servers need to be stopped and started again.
- Notebook server is crashing
  - Too many Notebooks are open and machine has no more free memory
  - IPython Issue: Report to IPython.org

Open Issues and Q & A

- Limiting the memory usage by Notebook servers
  - Possible by using Supervisord plugin
- Disk quota
  - No restriction is setup

Links and Resources

- Application Source code: https://github.com/writefaruq/supervised-ipython-nbserver
- IPython: http://www.ipython.org
- Supervisord: http://supervisord.org/
- Django: https://www.djangoproject.com/
- Pinax: http://pinaxproject.com/