Administrivia

• Did everyone get their webserver (*OS X or Ubuntu or both*) up and running?
Today's Topic

• Running a program on the webserver
  – cgi-bin
• Homework 8
• Next time ...
  – Sending form data to the webserver
  • GET method
  • POST method
Apache Webserver on OSX

Webpage storage locations:
• cgi-bin directory:
  – /Library/WebServer/CGI-Executables/

• usage:
  – http://localhost/cgi-bin/test.cgi
  • permissions for *.cgi should be readable and executable
    – ls -l /Library/WebServer/CGI-Executables/
      – -rwxr-xr-x 1 root wheel 161 Oct 16 2014 get.cgi
      – -rwxr-xr-x 1 root wheel 125 Oct 21 2014 post.cgi
      – -rw-r--r-- 1 root wheel 113 Oct 27 16:06 test.cgi

  – to change permissions
    • sudo chmod 755 test.cgi
Apache Webserver on OSX

• By default, Apple ships Apache2 with the cgi module turned off. Assuming the webserver is running, possible symptoms:

  prints program.cgi as plain text instead of executing it

  get.cgi doesn't exist!
Apache Webserver on OSX

- Enabling cgi-bin:
  1. uncomment cgi_module line in file /etc/apache2/httpd.conf
     • sudo nano /etc/apache2/httpd.conf
  2. restart apache2
Apache Webserver on OSX

• Simple bash script, let's call it test:
  1. `#!/bin/bash`
  2. `echo "Content-Type: text/plain"`
  3. `echo`
  4. `echo "Running Bash script from cgi-bin successfully!"`
  5. `echo -n "Now: "`
  6. `date`
  7. `echo -n "User: "`
  8. `whoami`
  9. `exit 0`

```bash
$pwd
/Library/WebServer/CGI-Executables
$ls -l test
-rwxr-xr-x 1 root wheel 158 Oct 27 16:55 test
$bash test
Content-type: text/plain

Running Bash script from cgi-bin successfully!
Now: Tue Oct 27 16:58:24 MST 2015
User: sandiway
$
```
Apache Webserver on OSX

Compare:

```
bash test
Content-type: text/plain

Running Bash script from cgi-bin successfully!
User: sandiway
$```

```
http://localhost/cgi-bin/test

Running Bash script from cgi-bin successfully!
Now: Tue Oct 27 18:41:37 MST 2015
User: _www
```

- User:
- Preamble
Apache Webserver on Ubuntu

- [http://localhost/cgi-bin/](http://localhost/cgi-bin/)
- CGI binaries directory: /usr/lib/cgi-bin/
  - files must be made executable!
Apache Webserver on Ubuntu

- Test:

```
sandiway@sandiway-VirtualBox:/usr/lib/cgi-bin$ bash test
Content-Type: text/plain

Running Bash script from cgi-bin successfully!
Now: Tue Oct 27 19:07:49 MST 2015
User: sandiway
sandiway@sandiway-VirtualBox:/usr/lib/cgi-bin$
```
Apache Webserver on Ubuntu

- Enabling cgi-bin:
  - `sudo a2enmod cgi`  \(\text{ (enables cgid instead of cgi) }\)
  - `/etc/apache2/mods-enabled/`

```bash
sandiway@sandiway-VirtualBox:/etc/apache2$ ls
apache2.conf conf-enabled magic mods-enabled sites-available
conf-available envvars mods-available ports.conf sites-enabled
sandiway@sandiway-VirtualBox:/etc/apache2$ ls mods-enabled/
access_compat.load authz_user.load filter.load setenvif.load
alias.conf authindex.conf mine.conf status.conf
alias.load autoindex.load mine.load status.load
auth_basic.load deflate.conf mpm_event.conf
authn_core.load deflate.load mpm_event.load
authn_file.load dir.conf negotiation.conf
authz_core.load dir.load negotiation.load
authz_host.load env.load setenvif.conf

Before

sandiway@sandiway-VirtualBox:/etc/apache2$ ls mods-enabled/
access_compat.load authz_user.load dir.load negotiation.load
alias.conf authindex.conf mine.conf status.conf
alias.load autoindex.load mine.load status.load
auth_basic.load deflate.conf mpm_event.conf
authn_core.load deflate.load mpm_event.load
cgid.load
authn_file.load dir.conf negotiation.conf
authz_core.load dir.load negotiation.load
authz_host.load dir.conf setenvif.conf

After
```
Apache Webserver on Ubuntu

• Compare running test directly:

```bash
sandiway@sandiway-VirtualBox:/usr/lib/cgi-bin$ bash test
Content-Type: text/plain

Running Bash script from cgi-bin successfully!
Now: Tue Oct 27 19:07:49 MST 2015
User: sandiway
sandiway@sandiway-VirtualBox:/usr/lib/cgi-bin$  
```

• http://localhost/cgi-bin/test:

User: www-data

On OS X: _www
Apache Webserver on OSX

- To run programs in ~sandiway/Sites, i.e. outside of
  - /Library/WebServer/CGI-Executables
- modify the Apache httpd configuration file:
  - sudo nano /etc/apache2/httpd.conf

```bash
# AddHandler allows you to map certain file extensions to "handlers":
# actions unrelated to filetype. These can be either built into the server
# or added with the Action directive (see below)
#
# To use CGI scripts outside of ScriptAliased directories:
# (You will also need to add "ExecCGI" to the "Options" directive.)
#
#AddHandler cgi-script .cgi
```

invokes the *cgi-script* handler for all files of type `.cgi`
Apache Webserver on OSX

• Also modify /etc/apache2/users/sandiway.conf (created last lecture) to add the ExecCGI option as follows:
  1. <Directory "/Users/sandiway/Sites/">
  2. AllowOverride All
  3. Options Indexes Multiviews FollowSymLinks ExecCGI
  4. Require all granted
  5. </Directory>

File ~sandiway/Sites/test.cgi

#!/bin/bash
echo "Content-Type: text/plain"
echo
echo "Running Bash script from ~/Sites successfully!"
echo -n "Now: "
date
echo -n "User: "
whoami
exit 0
Apache Webserver on OSX

- http://localhost/~/sandiway/test.cgi

```
Running Bash script from ~/Sites successfully!
Now: Tue Oct 27 22:44:05 MST 2015
User: _www
```
Apache Webserver on Ubuntu

- By default cgi-bin is not enabled for ~/public_html, just displays program instead of running it
Apache Webserver on Ubuntu

• From
  https://httpd.apache.org/docs/current/howto/cgi.html
  – add these lines to /etc/apache2/apache.conf
    • <Directory "/home/*/public_html"/>
      • Options +ExecCGI
      • AddHandler cgi-script .cgi
    • </Directory>
  – and restart apache2
Apache Webserver on Ubuntu

- `/etc/apache2/apache2.conf`

I put the declarations here (I'm not sure it's the best place though...)

```xml
<Directory /usr/share>
  AllowOverride None
  Require all granted
</Directory>

<Directory /var/www/>
  Options Indexes FollowSymLinks
  AllowOverride None
  Require all granted
</Directory>

<Directory /home/*/public_html>
  Options +ExecCGI
  AddHandler cgi-script .cgi
</Directory>

#<Directory /srv/>
#  Options Indexes FollowSymLinks
#  AllowOverride None
#  Require all granted
#</Directory>
```
Apache Webserver on Ubuntu

- /etc/apache2/mods-available/userdir.conf

```html
<IfModule mod_userdir.c>
    UserDir public_html
    UserDir disabled root

    <Directory /home/*/public_html>
        AllowOverride FileInfo AuthConfig Limit Indexes
        Options MultiViews Indexes SymLinksIfOwnerMatch IncludesNoExec
        <Limit GET POST OPTIONS>
            Require all granted
        </Limit>
        <LimitExcept GET POST OPTIONS>
            Require all denied
        </LimitExcept>
    </Directory>
</IfModule>

# vim: syntax=apache ts=4 sw=4 sts=4 sr noet
```
Apache Webserver on Ubuntu

- `~sandiway/public_html/test.cgi`

```bash
#!/bin/bash
echo "Content-Type: text/plain"
echo
echo "Running Bash script from ~sandiway/public_html successfully!"
echo -n "Now: "
date
echo -n "User: "
whoami
exit 0
```

```
Running Bash script from ~sandiway/public_html successfully!
Now: Tue Oct 27 23:45:40 MST 2015
User: www-data
```
Homework 8

• Either on OS X or Ubuntu, enable cgi-bin to work from both the default location:
  – Ubuntu: /usr/lib/cgi-bin/
  – OS X: /Library/WebServer/CGI-Executables
  – and from your user directory:
    – Ubuntu: ¯yourusername/public_html/
    – OS X: ¯yourusername/Sites/

• Write two different bash shell scripts, one for the default location and one for the user directory

• Show your system works (screenshots)
Homework 8

• Make sure you get Homework 7 working before attempting Homework 8
• Both are due Saturday night (by midnight)
Apache Webserver

• Reference documentation:

• Warning message:
  – "Could not reliably determine the server’s fully qualified domain name"
  – Solution: define:
    • ServerName localhost