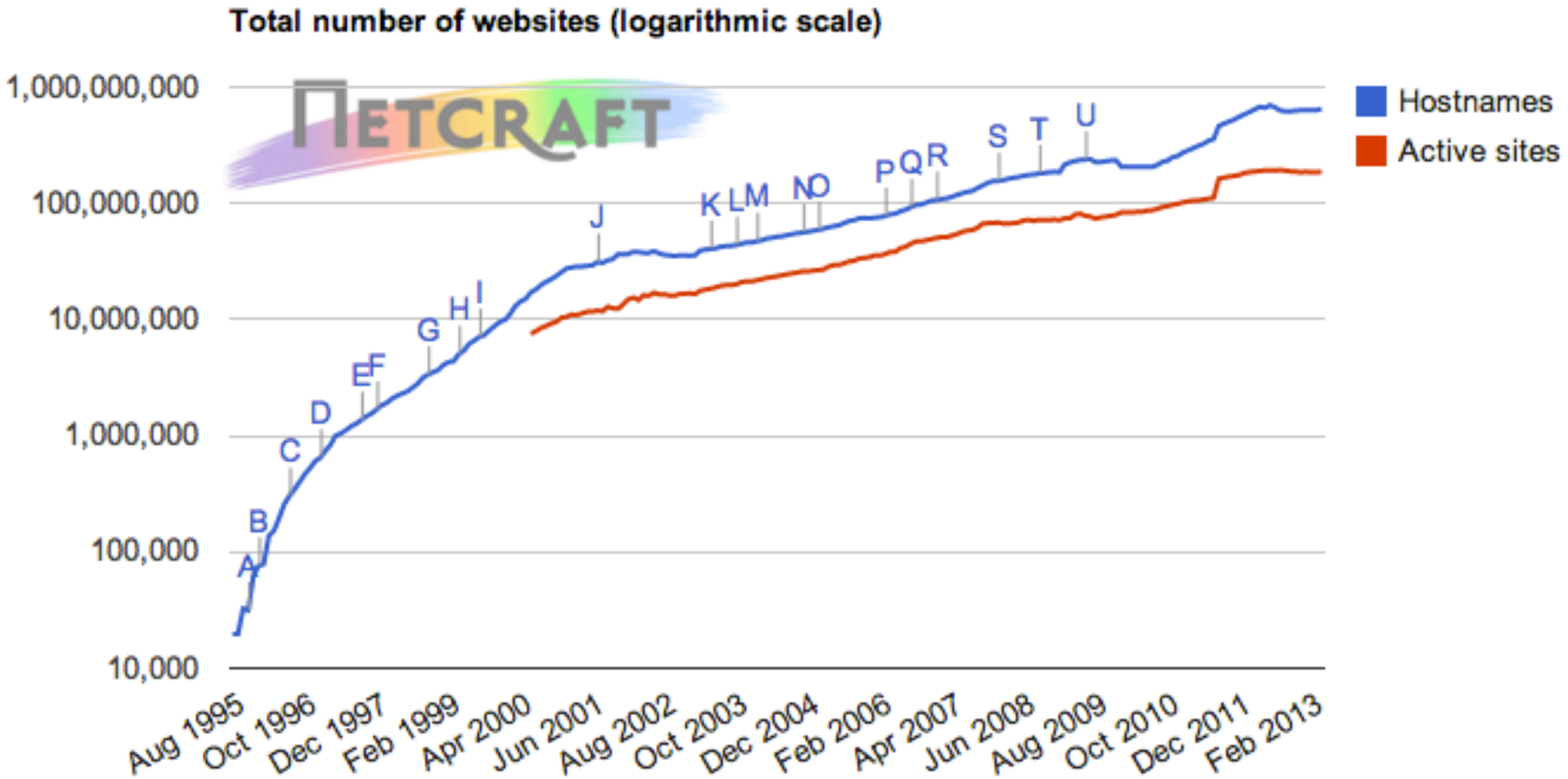
Web server and Apache

Lorenzo Bracciale

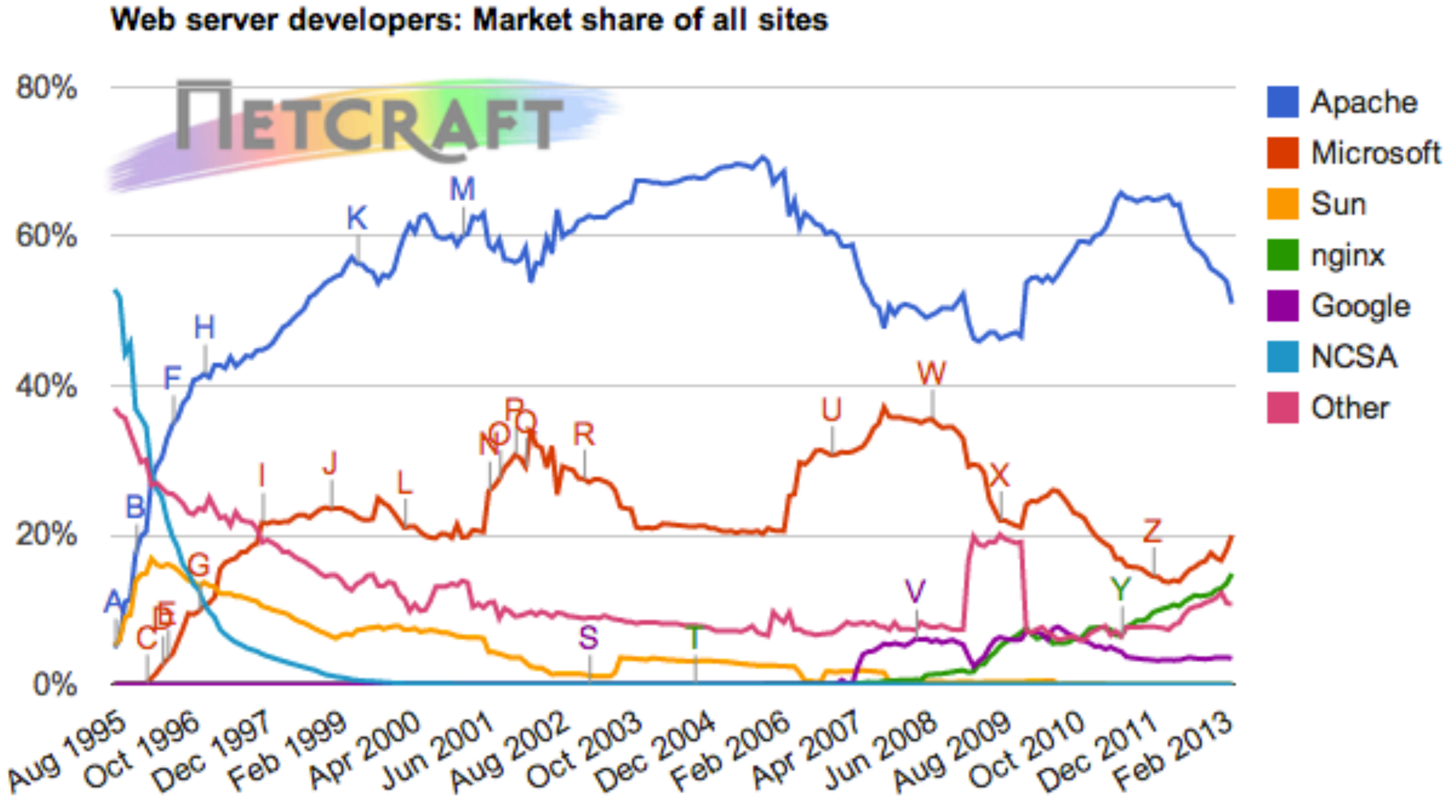
Marco Bonola

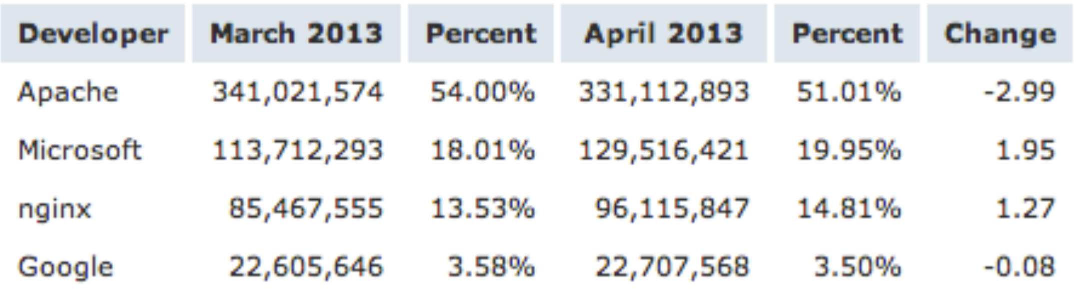
Number of Websites



src: http://news.netcraft.com/archives/category/web-server-survey/

Apache diffusion





Developer	March 2013	Percent	April 2013	Percent	Change
Apache	341,021,574	54.00%	331,112,893	51.01%	-2.99
Microsoft	113,712,293	18.01%	129,516,421	19.95%	1.95
nginx	85,467,555	13.53%	96,115,847	14.81%	1.27
Google	22,605,646	3.58%	22,707,568	3.50%	-0.08

because of
user-generated content →

Where to study

- Apache Server 2 - Mohammed J. Kabir
  - Hungry Minds

- Apache Server 2 Official Documentation
  - http://httpd.apache.org/docs/2.0/

Apache

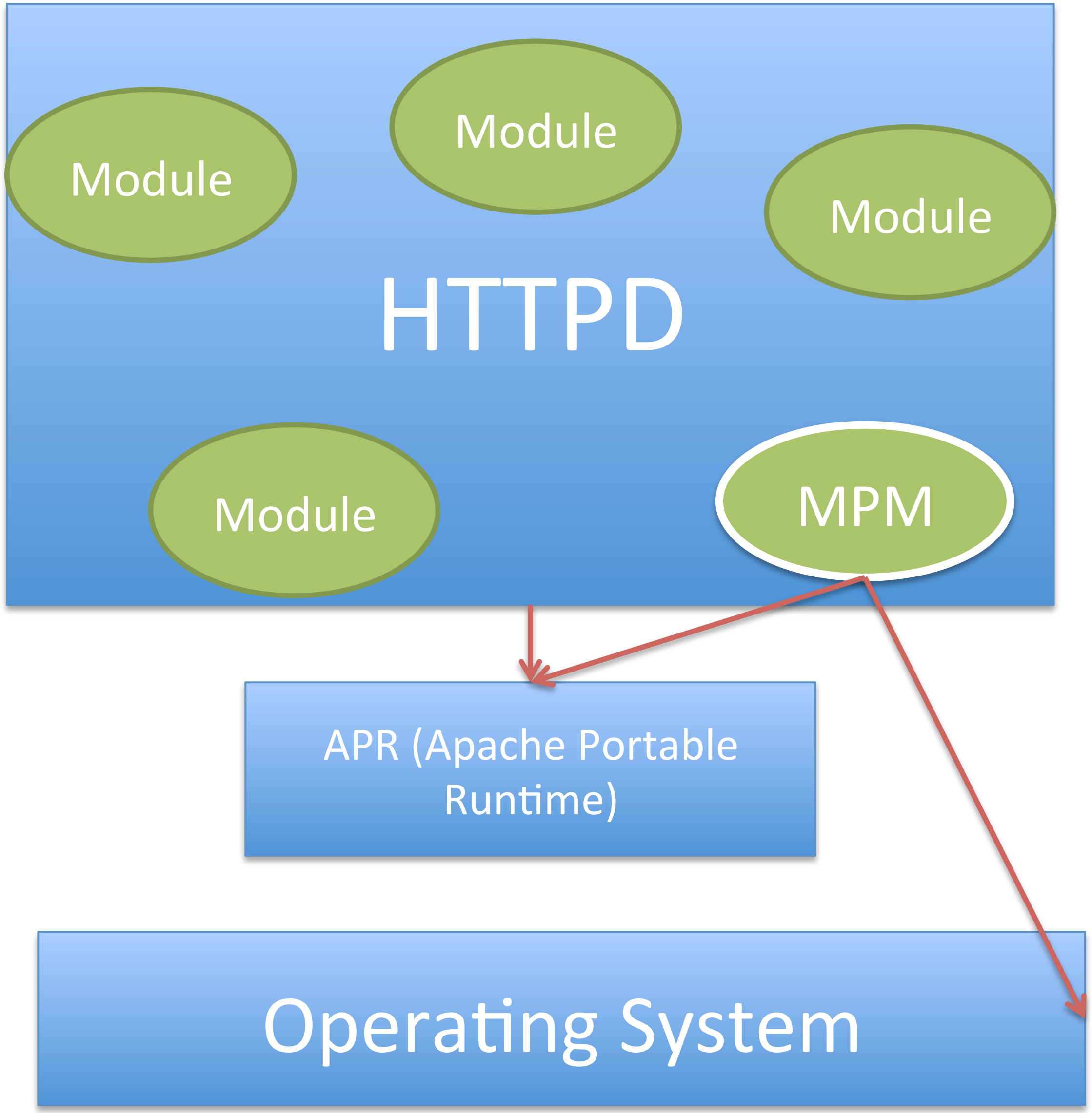
- developed by the *Apache Software Foundation* (released April 1995)
- bundled in the popular "LAMP" (**L**inux **A**pache **M**ysql **P**hp)

- apt-get install apache2

- Start / stop:
  - /etc/init.d/apache2 start
  - /etc/init.d/apache2 stop

try!

Apache Architecture



small core

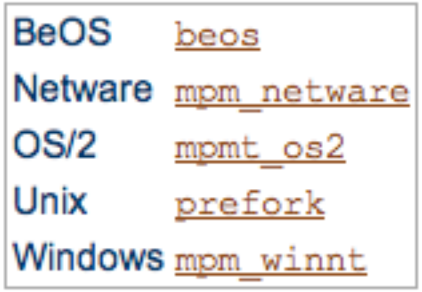
several modules

- compiled statically or loaded dynamically

Cross platform

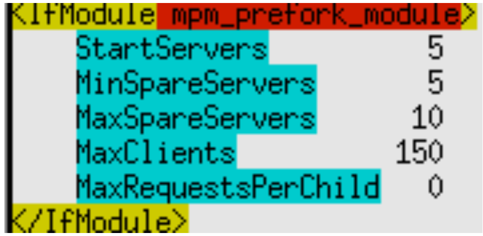
utilities (APR)

How apache works: MPM



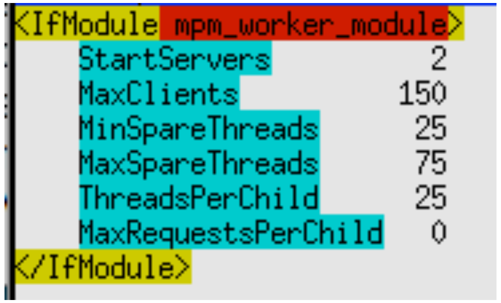
BeOS	beos
Netware	mpm_netware
OS/2	mpmt_os2
Unix	prefork
Windows	mpm_winnt

- MultiProcessing Modules (MPMs) since Apache2:
  - In apache 1.3 uses a preforking architecture
    - the parent creates/destroys children if required
    - does not work well on some platform (such Windows)
  - MPM offers several alternatives (implemented in MPM modules) :
    - prefork MPM (like Apache 1.3)
    - worker MPM (multiple child, each one with several threads)
    - winnt MPM: single process, multithread (specific for windows)



```
<IfModule mpm_prefork_module>
    StartServers          5
    MinSpareServers       5
    MaxSpareServers      10
    MaxClients          150
    MaxRequestsPerChild   0
</IfModule>
```

We can tune parameters in /etc/apache2/apache2.conf



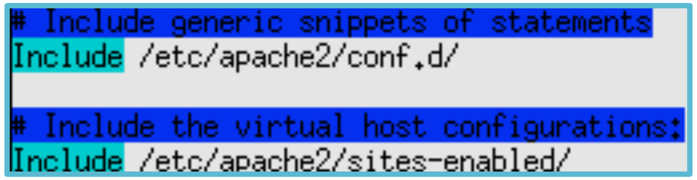
```
<IfModule mpm_worker_module>
    StartServers          2
    MaxClients          150
    MinSpareThreads      25
    MaxSpareThreads      75
    ThreadsPerChild      25
    MaxRequestsPerChild   0
</IfModule>
```

Processes vs Threads

- Both threads and processes are methods of parallelizing an application
- **Processes** are independent execution units that contain their own state information, use their own address spaces, and only interact with each other via inter-process communication (IPC) mechanisms
- **Threads** share the same state and same memory space, and can communicate with each other directly, because they share the same variables

Are your cgi library thread safe?

Apache Web server files



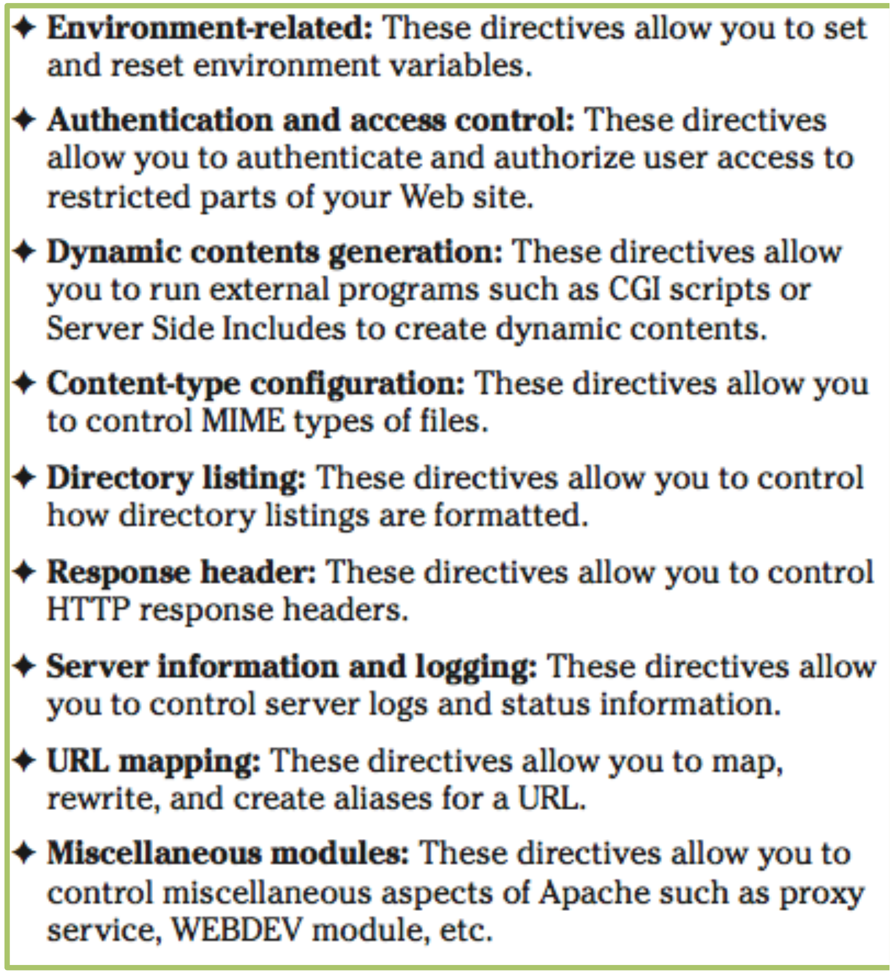
```
# Include generic snippets of statements
Include /etc/apache2/conf.d/

# Include the virtual host configurations:
Include /etc/apache2/sites-enabled/
```

/usr/sbin/apache2	Apache 2 server binary
/usr/sbin/apache2ctl	Apache2 control interface (configtest could help!)
/etc/apache2/apache2.conf	default configuration file (could be overwritten during apache upgrade)
/etc/apache2/httpd.conf	user configuration files (and files inside /etc/apache2/conf.d)
/etc/apache2/conf.d	other configuration files (included as well in apache2.conf)
/etc/apache2/sites-available	configuration files for virtual hosting
/etc/apache2/sites-enable	symbolic links to sites-available files (created with a2ensite, a2dissite)
/etc/apache2/mods-available	configuration files for modules
/etc/apache2/mods-enabled	symbolic links to mods-available files (created with a2enmod, a2dismod)
/var/log/apache2	log files

Configuring Apache

- ~ 360 directives (!!!). Few selected:
  - ServerRoot: path to configuration, error and log files
  - PidFile
  - ServerName: name and port of the server
  - DocumentRoot: where find files to serve
  - ErrorDocument: override standard error messages

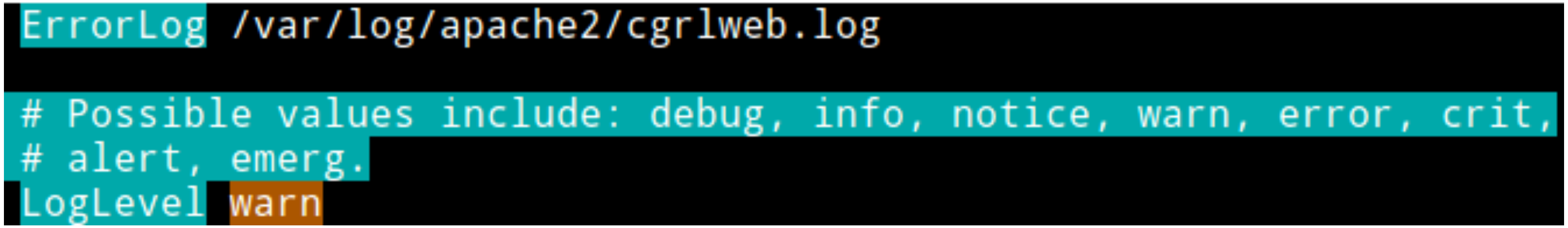


- ✦ **Environment-related:** These directives allow you to set and reset environment variables.
- ✦ **Authentication and access control:** These directives allow you to authenticate and authorize user access to restricted parts of your Web site.
- ✦ **Dynamic contents generation:** These directives allow you to run external programs such as CGI scripts or Server Side Includes to create dynamic contents.
- ✦ **Content-type configuration:** These directives allow you to control MIME types of files.
- ✦ **Directory listing:** These directives allow you to control how directory listings are formatted.
- ✦ **Response header:** These directives allow you to control HTTP response headers.
- ✦ **Server information and logging:** These directives allow you to control server logs and status information.
- ✦ **URL mapping:** These directives allow you to map, rewrite, and create aliases for a URL.
- ✦ **Miscellaneous modules:** These directives allow you to control miscellaneous aspects of Apache such as proxy service, WEBDEV module, etc.

Apache Modules

- Apache has modular architecture:
  - To enable/disable modules : a2enmod / a2dismod MODNAME
  - configurable via directives
  - apache2ctl –M #list
  - mod_so load module at runtime (Dynamic Shared Object (DSO) mechanism) LoadModule

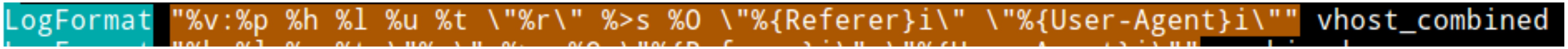
Logging



```
ErrorLog /var/log/apache2/cgrlweb.log

# Possible values include: debug, info, notice, warn, error, crit,
# alert, emerg.
LogLevel warn
```

- location and content: CustomLog directive
- Format: LogFormat
  - – specified with common logfile format*



```
LogFormat "%v:%p %h %l %u %t \"%r\" %>s %O \"%{Referer}i\" \"%{User-Agent}i\"" vhost_combined
```

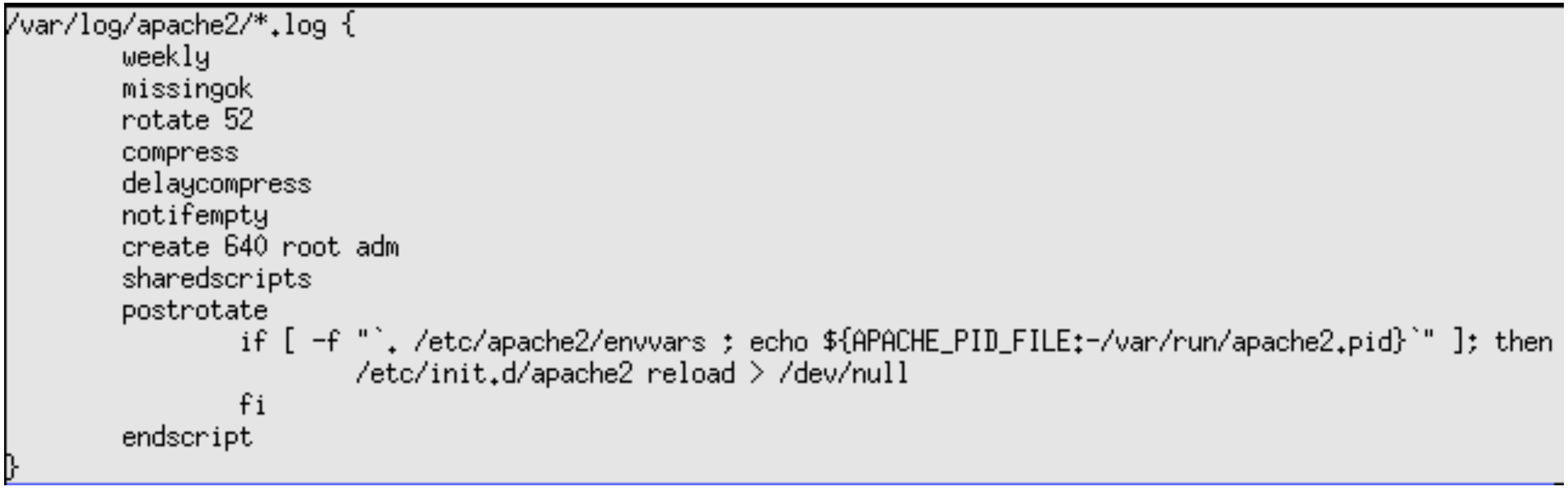
(*) http://www.w3.org/Daemon/User/Config/Logging.html#common-logfile-format

Logrotate

not only for apache!

Log Size: Typically 1 MB for 10000 requests so…

/etc/logrotate.d/apache2

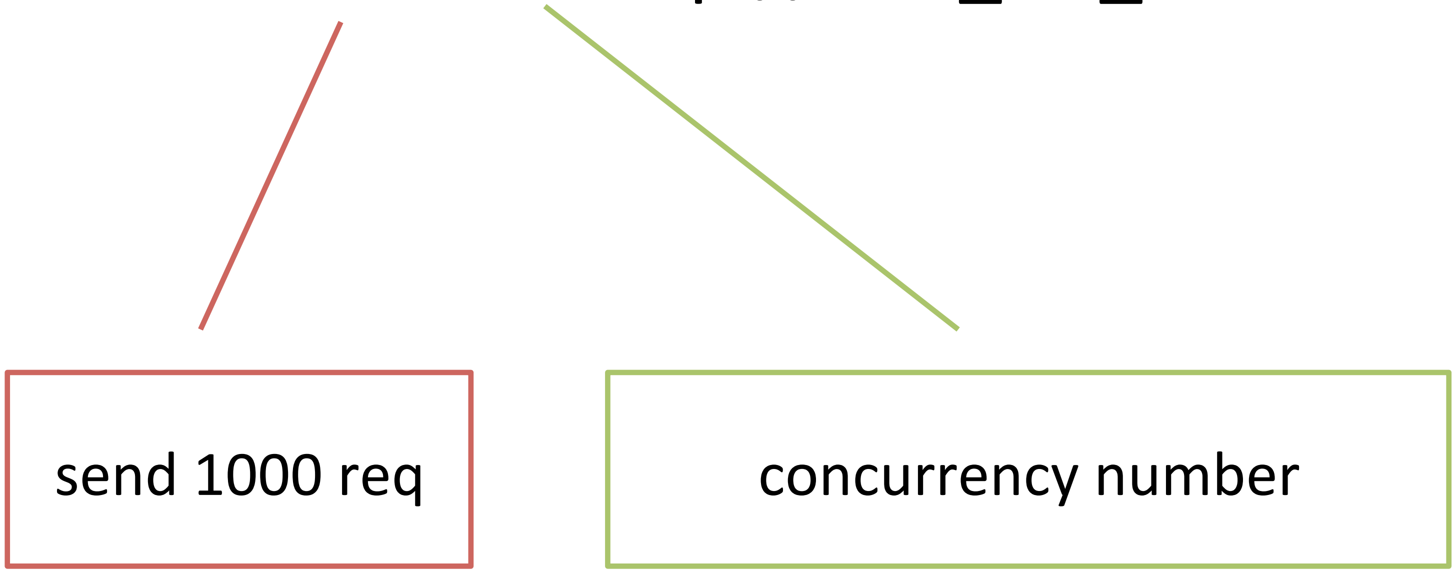


```
/var/log/apache2/*.log {
        weekly
        missingok
        rotate 52
        compress
        delaycompress
        notifempty
        create 640 root adm
        sharedscripts
        postrotate
                if [ -f "`. /etc/apache2/envvars ; echo ${APACHE_PID_FILE:-/var/run/apache2.pid}`" ]; then
                        /etc/init.d/apache2 reload > /dev/null
                fi
        endscript
}
```

- rotate at most 52 times, weekly
- compress (you can see that using zcat, zless or pipelining gzip and cat/tail)

Apache benchmarking

- ab (Apache HTTP server benchmarking tool)
- ab -n 1000 -c 5 http://URL_TO_TEST



send 1000 req

concurrency number

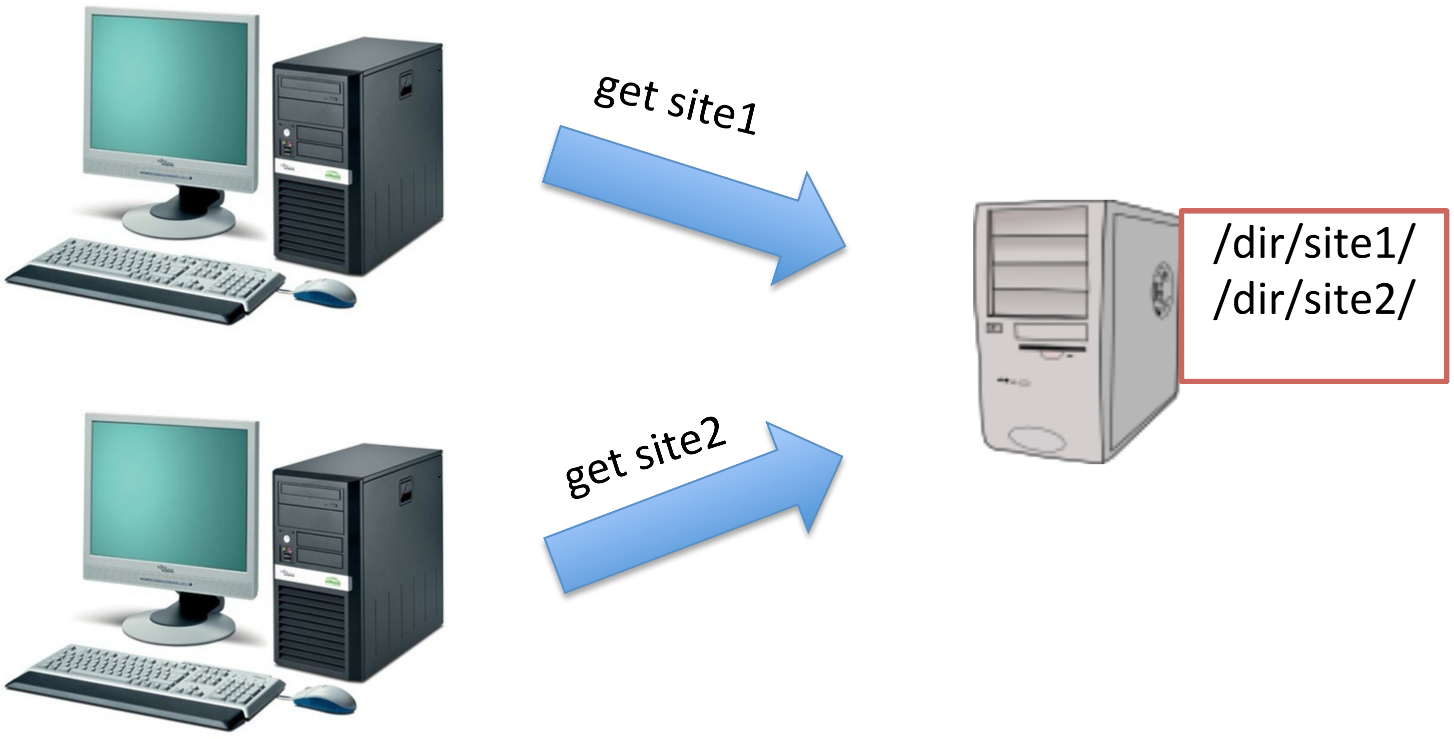
Reference documentation:
http://httpd.apache.org/docs/2.2/programs/ab.html

Exercise

- Benchmark apache changing MPM

Virtual Hosting

- Problem: *Several* websites, *one* webserver
  - Typically: *name-based* virtual host (but IP based is possible as well)

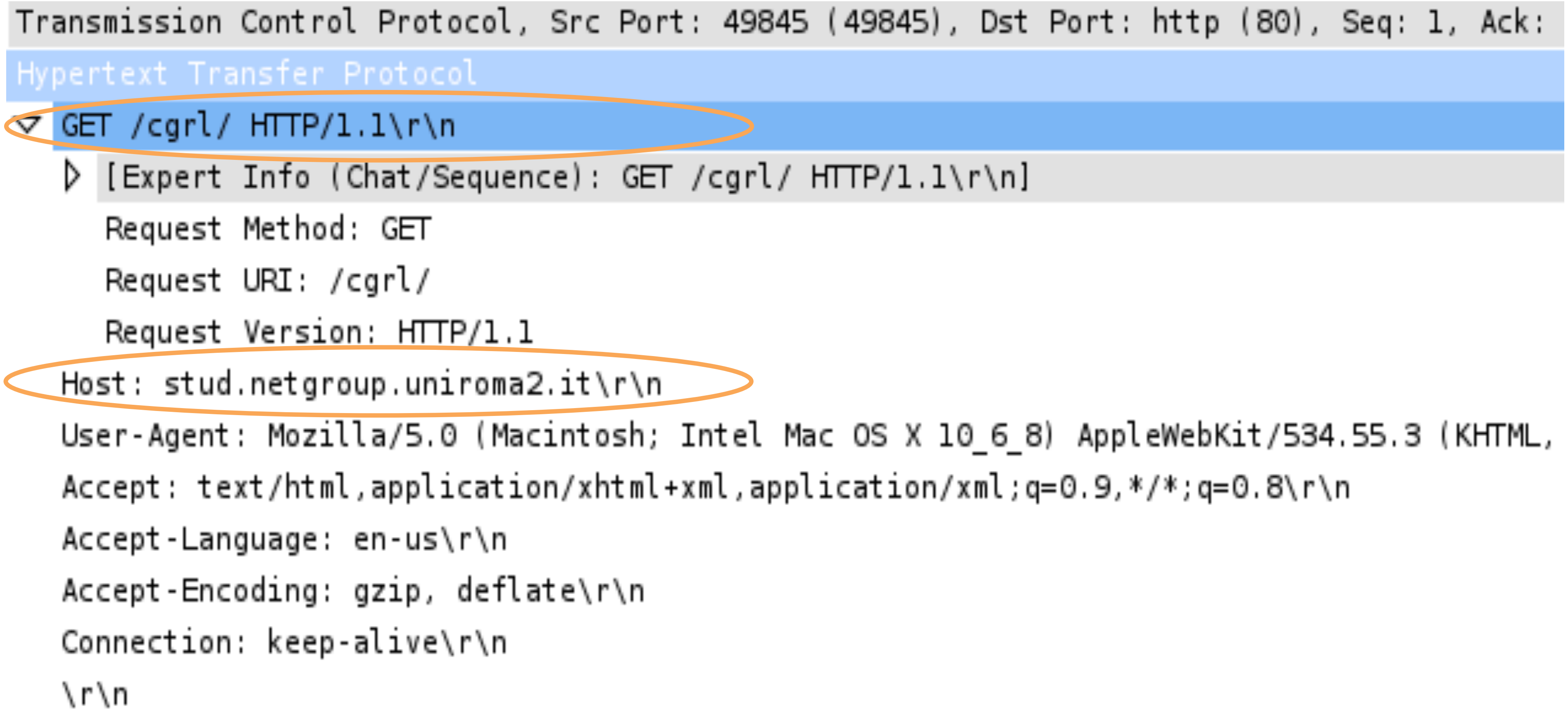


get site1

get site2

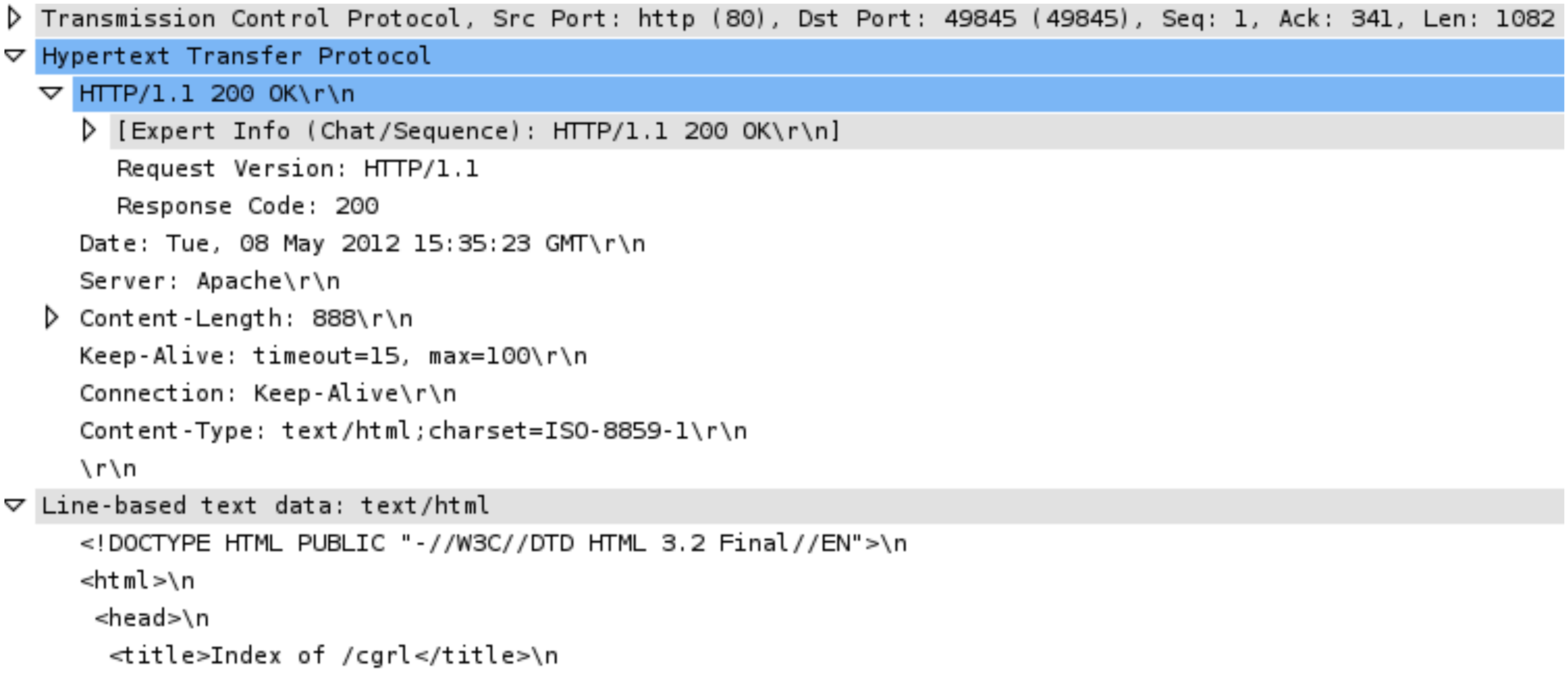
/dir/site1/
/dir/site2/

HTTP Request



```
Transmission Control Protocol, Src Port: 49845 (49845), Dst Port: http (80), Seq: 1, Ack:
Hypertext Transfer Protocol
▽ GET /cgrl/ HTTP/1.1\r\n
  ▷ [Expert Info (Chat/Sequence): GET /cgrl/ HTTP/1.1\r\n]
    Request Method: GET
    Request URI: /cgrl/
    Request Version: HTTP/1.1
  Host: stud.netgroup.uniroma2.it\r\n
  User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10_6_8) AppleWebKit/534.55.3 (KHTML,
  Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8\r\n
  Accept-Language: en-us\r\n
  Accept-Encoding: gzip, deflate\r\n
  Connection: keep-alive\r\n
  \r\n
```

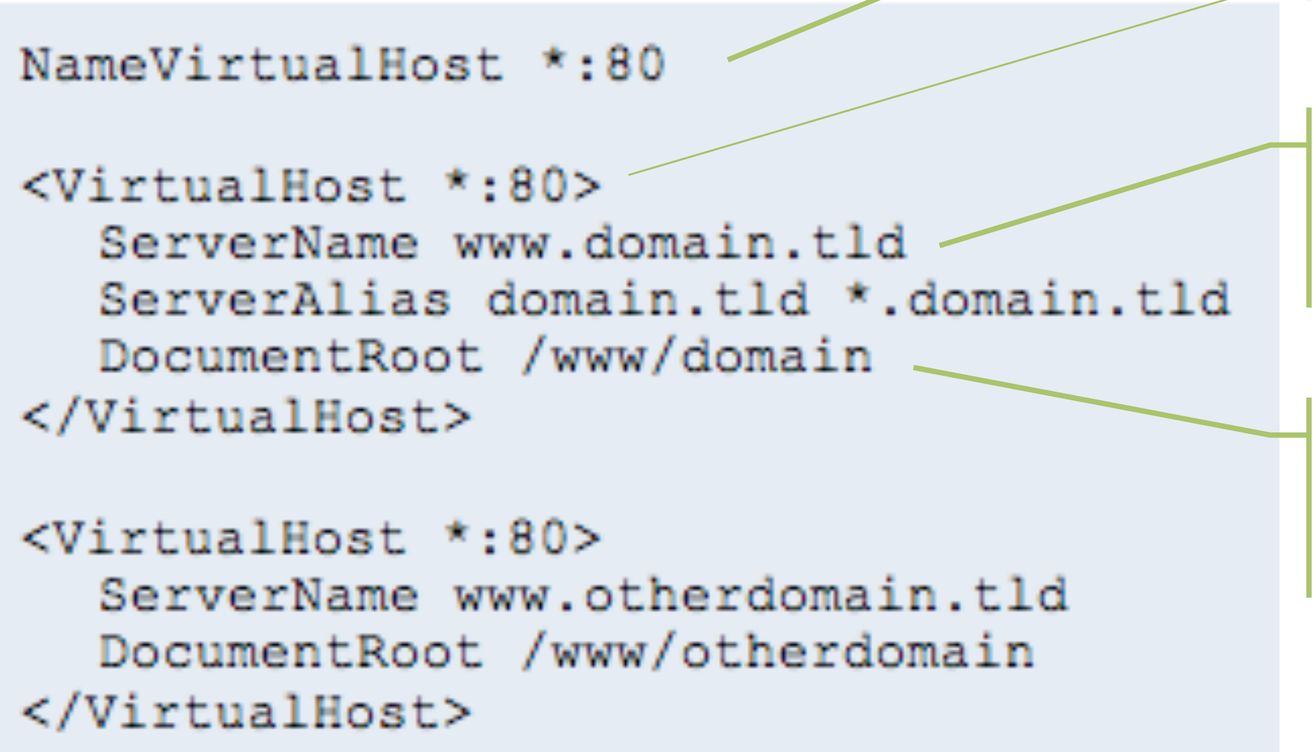
HTTP Response



```
▷ Transmission Control Protocol, Src Port: http (80), Dst Port: 49845 (49845), Seq: 1, Ack: 341, Len: 1082
▽ Hypertext Transfer Protocol
  ▽ HTTP/1.1 200 OK\r\n
    ▷ [Expert Info (Chat/Sequence): HTTP/1.1 200 OK\r\n]
      Request Version: HTTP/1.1
      Response Code: 200
    Date: Tue, 08 May 2012 15:35:23 GMT\r\n
    Server: Apache\r\n
  ▷ Content-Length: 888\r\n
    Keep-Alive: timeout=15, max=100\r\n
    Connection: Keep-Alive\r\n
    Content-Type: text/html;charset=ISO-8859-1\r\n
    \r\n
▽ Line-based text data: text/html
    <!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 3.2 Final//EN">\n
    <html>\n
     <head>\n
      <title>Index of /cgrl</title>\n
```

Virtual Hosting: example

put virtual hosts confs in sites-available dir!



```
NameVirtualHost *:80

<VirtualHost *:80>
   ServerName www.domain.tld
   ServerAlias domain.tld *.domain.tld
   DocumentRoot /www/domain
</VirtualHost>

<VirtualHost *:80>
   ServerName www.otherdomain.tld
   DocumentRoot /www/otherdomain
</VirtualHost>
```

IP/Port for listening requests

Name of the virtual host (match **host** http field)

Where the html files are (simple case)

use /etc/hosts to test virtual host without DNS modifications

DocumentRoot

- Where apache finds your documents (html files etc)
  - Typically search for: index.html index.cgi index.pl index.php index.xhtml index.htm
  - Defined from DirectoryIndex (order matters)

```
<html>
        <body>
                <h1>
                        HELLO CGRL
                </h1>
        </body>
</html>
```

index.html sample file

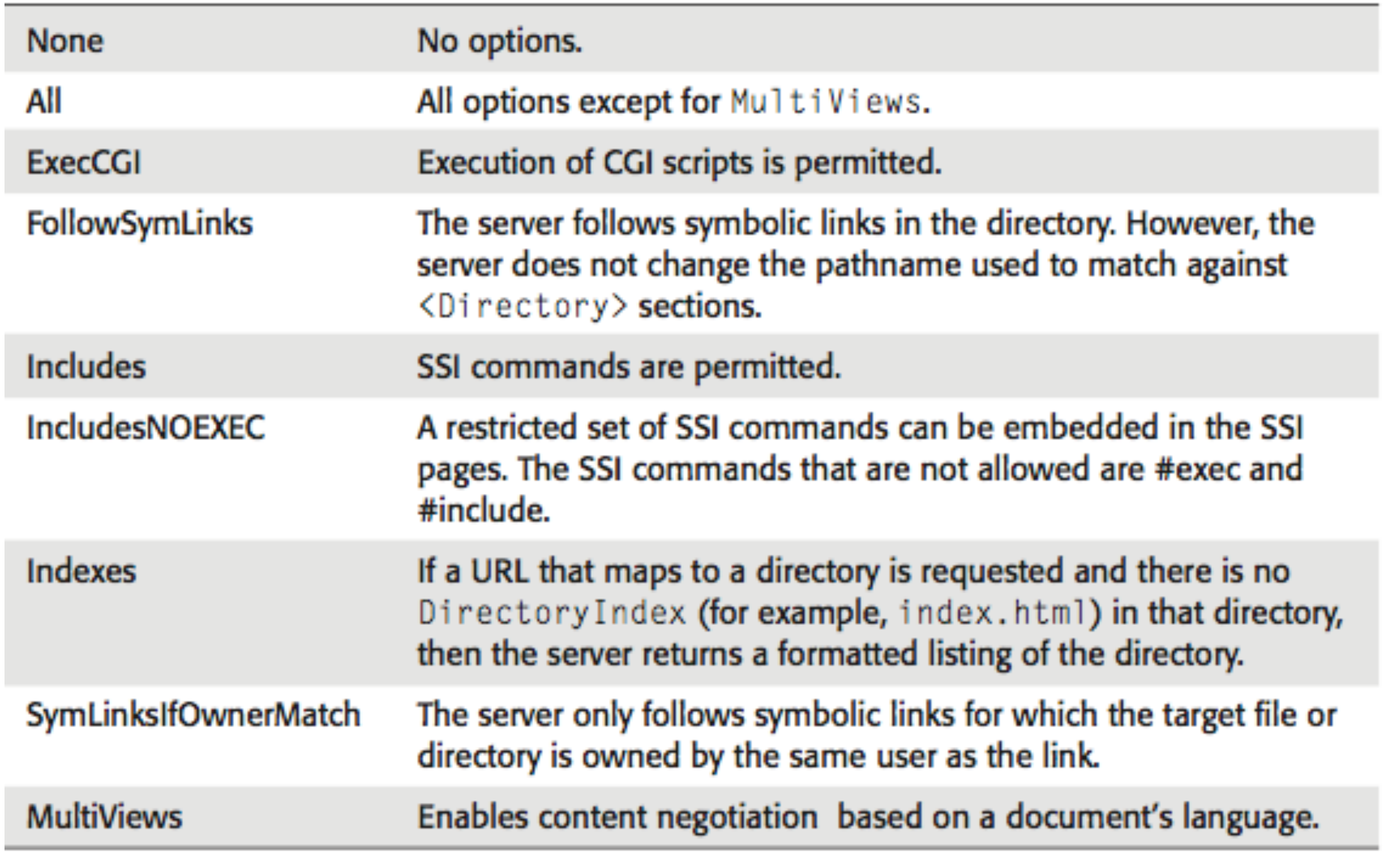
Exercise 1: virtual hosting

1. Create two dirs “cgrl-web” and “yourname-web”. Put in the directory two index.html files
2. Configure 2 virtual web hosts
   – www.cgrl.edu
   – www.yourname.com
3. Start apache
4. modify /etc/hosts
5. View your websites with a browser

Standard Container Directives

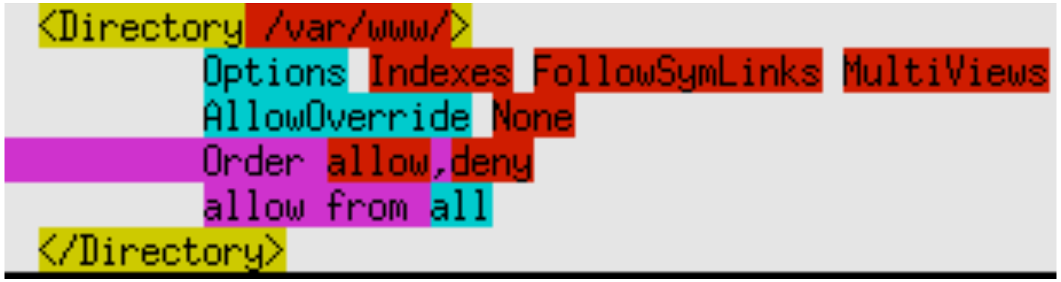
- Many Container contexts:
  - <VirtualHost ...>: already seen...
  - <Directory>: applies one or more directives to a directory
  - <Files>: applies one or more directives to a file
  - <Location>: applies one or more directives to a URL
- AllowOverride: enable/disable directories directives overriding.
- .htaccess: default filename for the per-directory configuration

Options directive



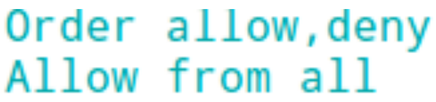
None	No options.
All	All options except for MultiViews.
ExecCGI	Execution of CGI scripts is permitted.
FollowSymLinks	The server follows symbolic links in the directory. However, the server does not change the pathname used to match against <Directory> sections.
Includes	SSI commands are permitted.
IncludesNOEXEC	A restricted set of SSI commands can be embedded in the SSI pages. The SSI commands that are not allowed are #exec and #include.
Indexes	If a URL that maps to a directory is requested and there is no DirectoryIndex (for example, index.html) in that directory, then the server returns a formatted listing of the directory.
SymLinksIfOwnerMatch	The server only follows symbolic links for which the target file or directory is owned by the same user as the link.
MultiViews	Enables content negotiation based on a document's language.

Options +Setting1 – Setting2



```
<Directory /var/www/>
        Options Indexes FollowSymLinks MultiViews
        AllowOverride None
        Order allow,deny
        allow from all
</Directory>
```

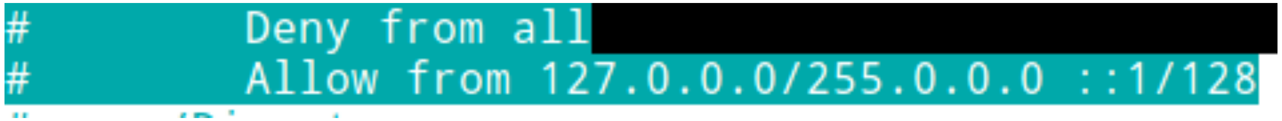
Allow-Deny



```
Order allow,deny
Allow from all
```

- First, all Allow directives are evaluated; at least one must match, or the request is rejected. Next, all Deny directives are evaluated. If any matches, the request is rejected. Last, any requests which do not match an Allow or a Deny directive are denied by default.

A domain name, IP, network/netmask (CIDR)



```
#        Deny from all
#        Allow from 127.0.0.0/255.0.0.0 ::1/128
```

Exercise 2: directory listening

1. Take the previous example
2. Create a directory in your DocumentRoot “myfiles” and put some stuff (try a symbolic link) inside that

   ```
   <Directory /your/dir/myfiles>
   Options +Indexes
   </Directory>
   ```

3. Create a directory inside “myfiles”: “mysecretfiles”:

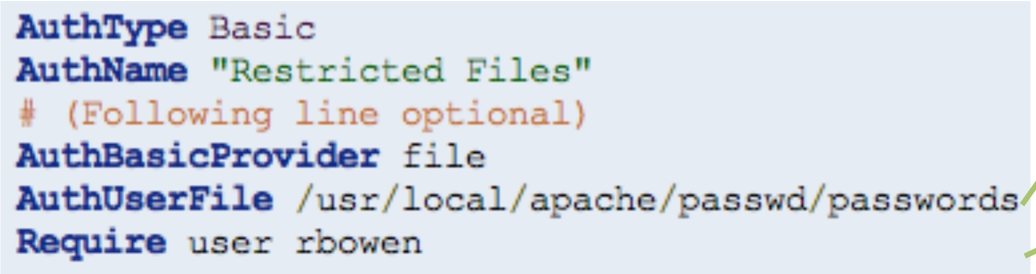
   ```
   <Directory /your/dir/myfiles/mysecrefiles>
   Options -Indexes
   </Directory>
   ```

.htaccess

- Same syntax as the main configuration files
  - so use <Directory> block instead (it's faster!)
  - Common misconception: not specifically for passwords or rewrite!
- "AllowOverride" : Types of directives that are allowed in .htaccess files (None, All, one or more directive inside these groups: {**AuthConfig, FileInfo, Indexes, Limit, Options}**)

Exercise 3: .htaccess password protection

- Let we create a new file with passwords:
  - htpasswd –c /root/apache_pass cgrl
  - Then put these directives in .htaccess (or <Directory>)



```
AuthType Basic
AuthName "Restricted Files"
# (Following line optional)
AuthBasicProvider file
AuthUserFile /usr/local/apache/passwd/passwords
Require user rbowen
```

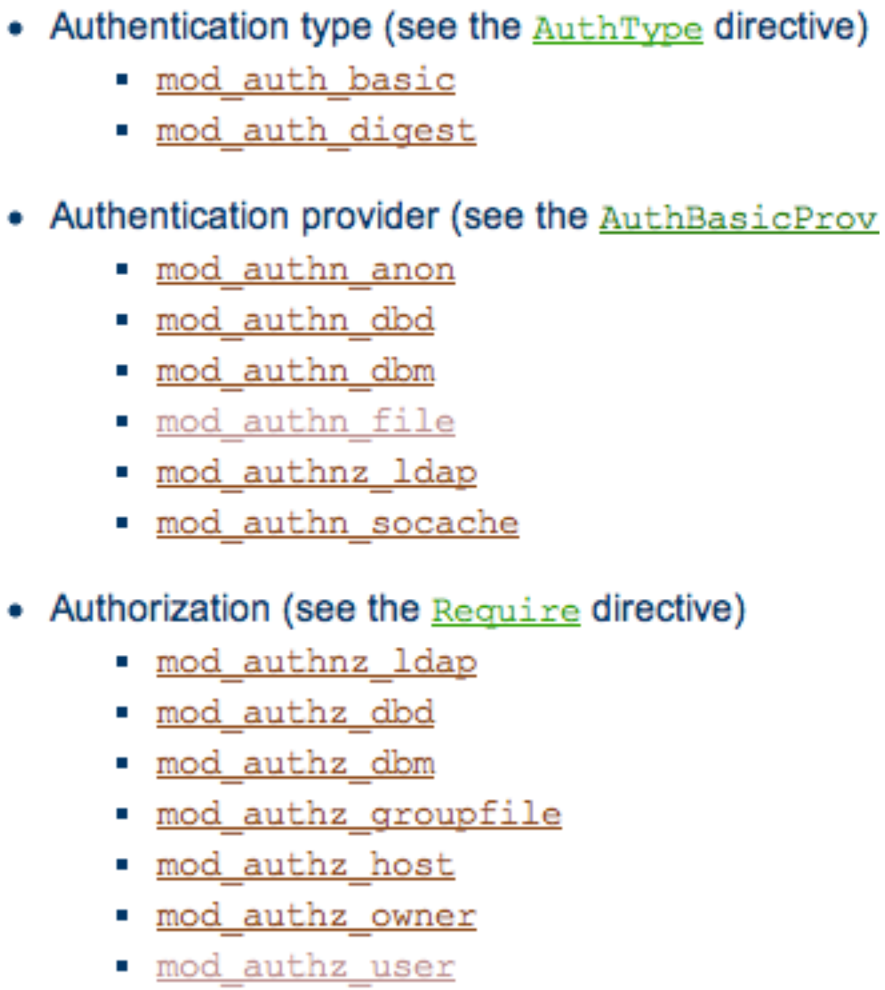
password file

users

- Now protect our secret file...

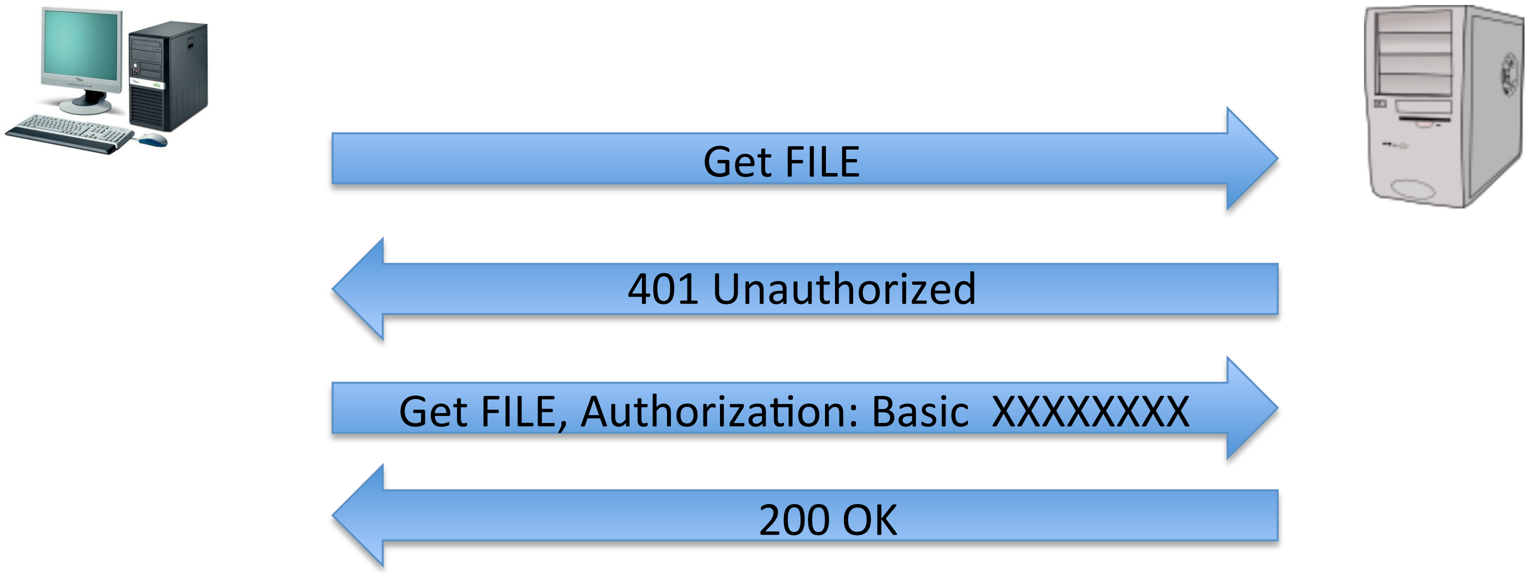
What we did?

- Authentication
  - process by which you verify that someone is who they claim he is
- Authorization
  - someone is allowed to be where they want to go, or to have information that he wants to have



- Authentication type (see the AuthType directive)
  - mod_auth_basic
  - mod_auth_digest
- Authentication provider (see the AuthBasicProv
  - mod_authn_anon
  - mod_authn_dbd
  - mod_authn_dbm
  - mod_authn_file
  - mod_authnz_ldap
  - mod_authn_socache
- Authorization (see the Require directive)
  - mod_authnz_ldap
  - mod_authz_dbd
  - mod_authz_dbm
  - mod_authz_groupfile
  - mod_authz_host
  - mod_authz_owner
  - mod_authz_user

HTTP Basic Authentication

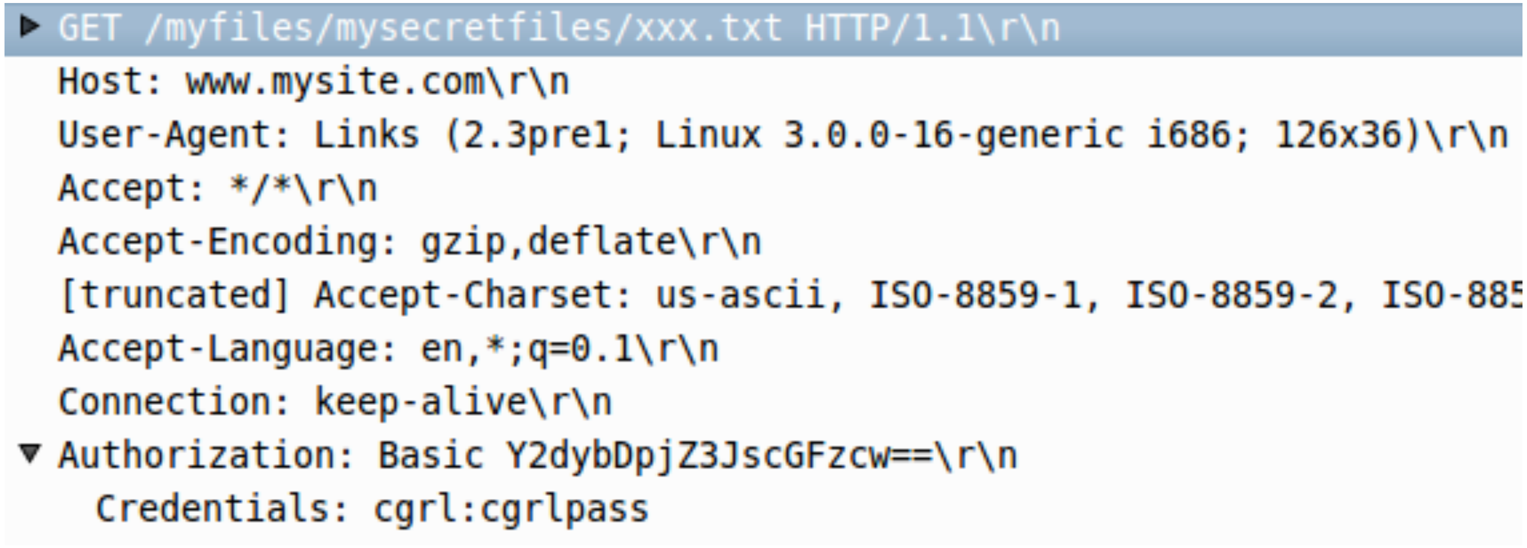


Get FILE

401 Unauthorized

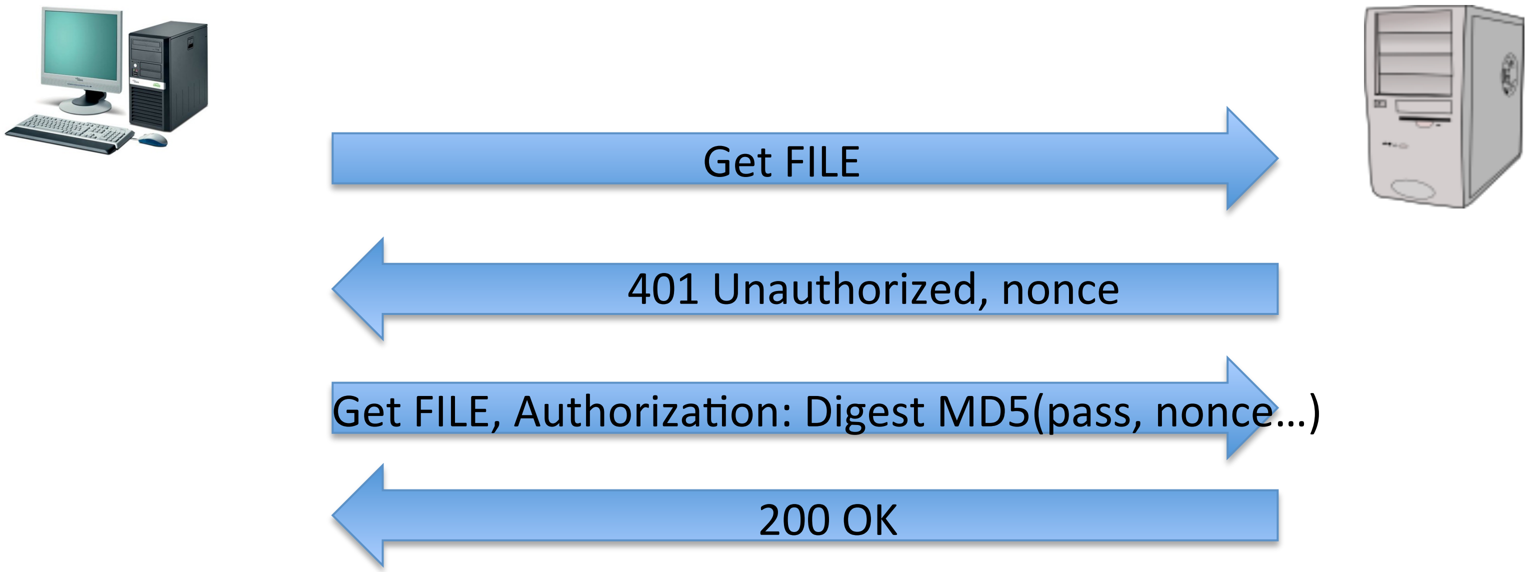
Get FILE, Authorization: Basic XXXXXXXX

200 OK



```
▶ GET /myfiles/mysecretfiles/xxx.txt HTTP/1.1\r\n
  Host: www.mysite.com\r\n
  User-Agent: Links (2.3pre1; Linux 3.0.0-16-generic i686; 126x36)\r\n
  Accept: */*\r\n
  Accept-Encoding: gzip,deflate\r\n
  [truncated] Accept-Charset: us-ascii, ISO-8859-1, ISO-8859-2, ISO-885
  Accept-Language: en,*;q=0.1\r\n
  Connection: keep-alive\r\n
▼ Authorization: Basic Y2dybDpjZ3JscGFzcw==\r\n
    Credentials: cgrl:cgrlpass
```

HTTP Digest Authentication

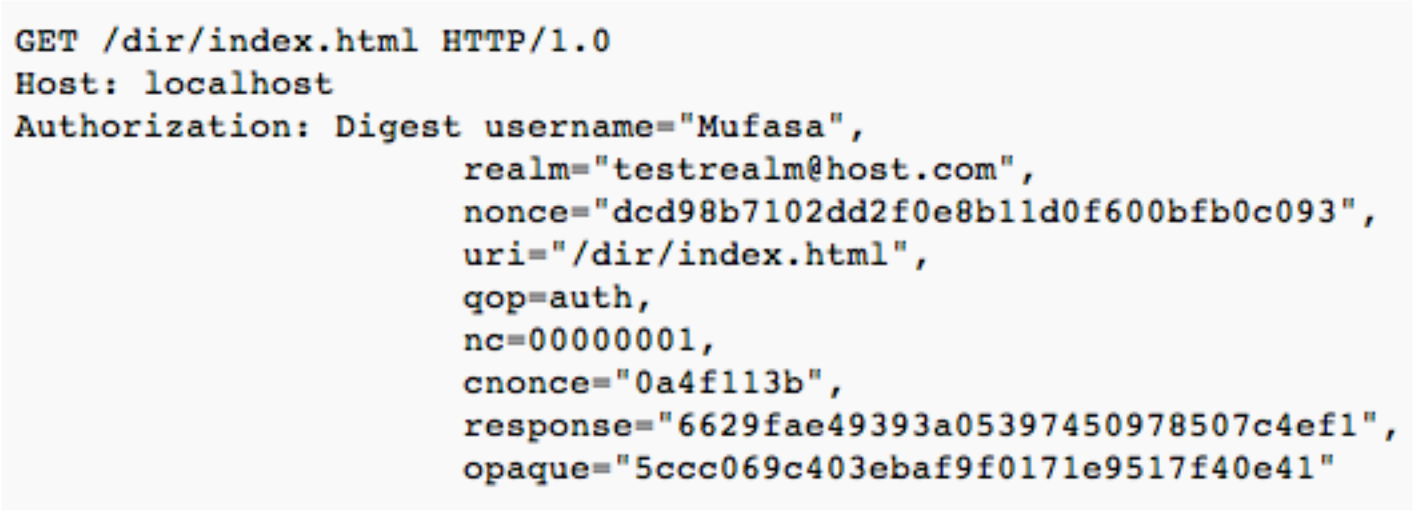


Get FILE

401 Unauthorized, nonce

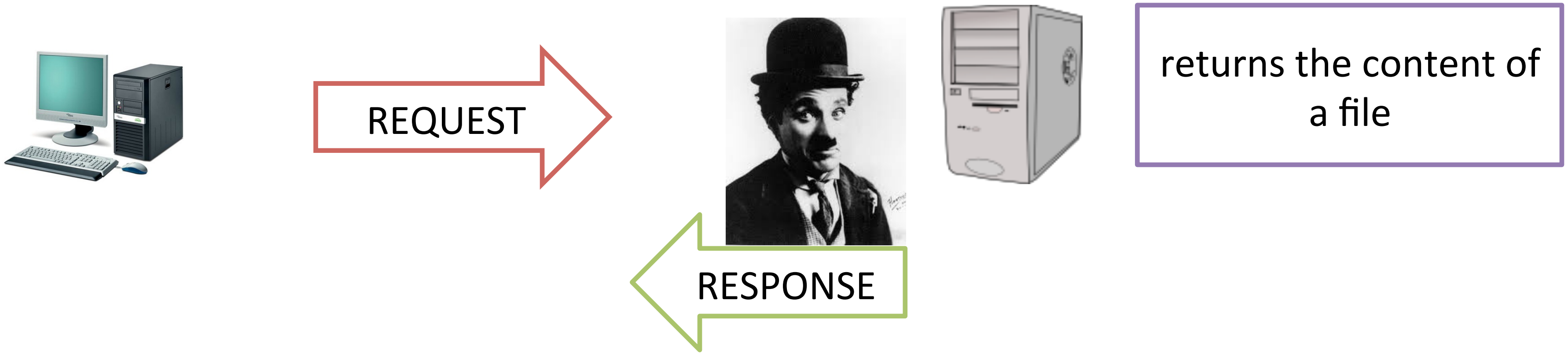
Get FILE, Authorization: Digest MD5(pass, nonce…)

200 OK



```
GET /dir/index.html HTTP/1.0
Host: localhost
Authorization: Digest username="Mufasa",
                     realm="testrealm@host.com",
                     nonce="dcd98b7102dd2f0e8b11d0f600bfb0c093",
                     uri="/dir/index.html",
                     qop=auth,
                     nc=00000001,
                     cnonce="0a4f113b",
                     response="6629fae49393a05397450978507c4ef1",
                     opaque="5ccc069c403ebaf9f0171e9517f40e41"
```

Static web pages

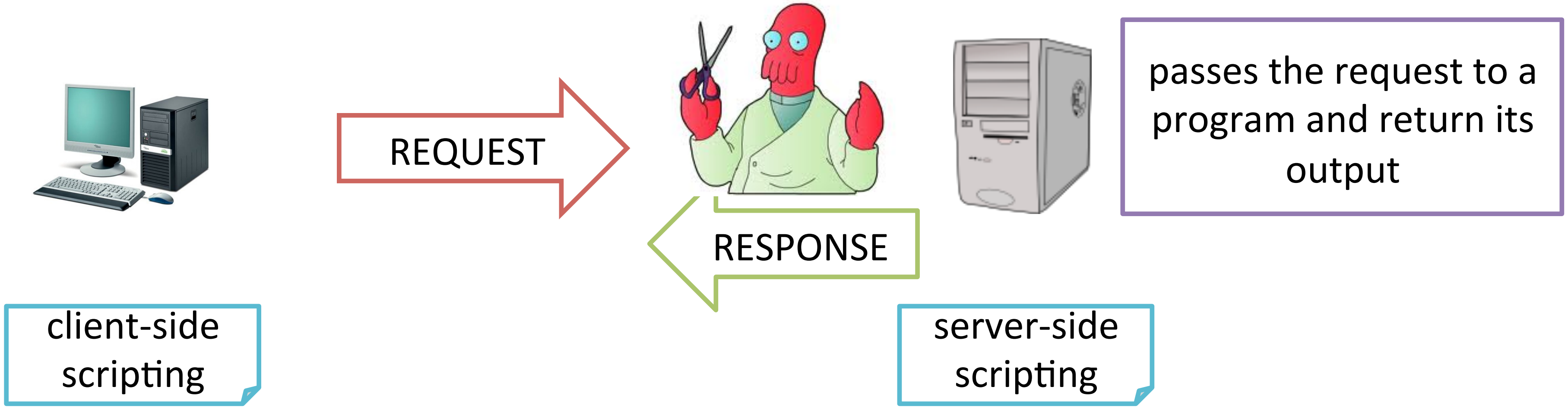


REQUEST

RESPONSE

returns the content of a file

Dynamic web pages



REQUEST

RESPONSE

passes the request to a program and return its output

client-side scripting

server-side scripting

Client-side scripting languages

- javascript
- actionscript

CAN NOT DO WHATEVER YOU WANT

Server-side scripting languagues

- C/C++
- bash (!)
- Perl
- ASP
- PHP
- Java
- Python
- Lua
- Ruby
- ColdFusion
- …

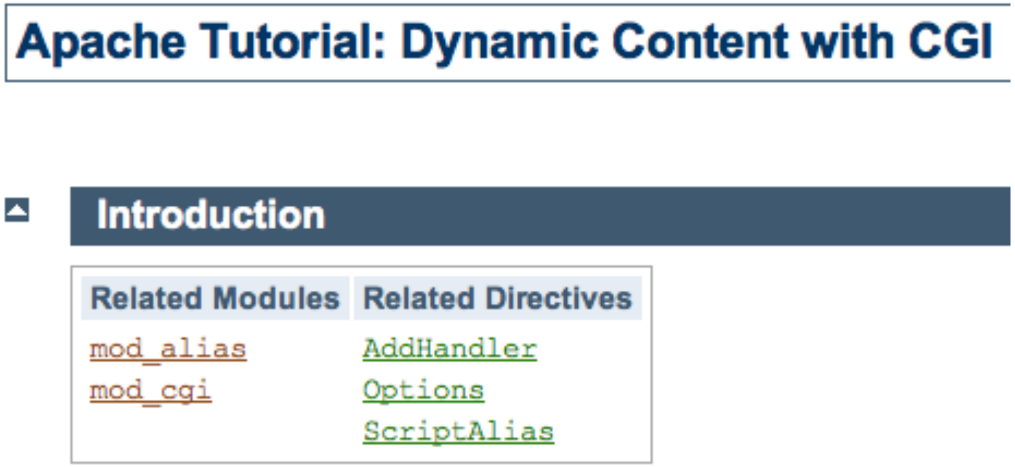
DO WHATEVER YOU WANT

rfc 3875 (2004)
used since 1991

mod_cgi

Common Gateway Interface

- Standard way to delegate the generation of web pages to executable files
- processes isolated from the core Web server



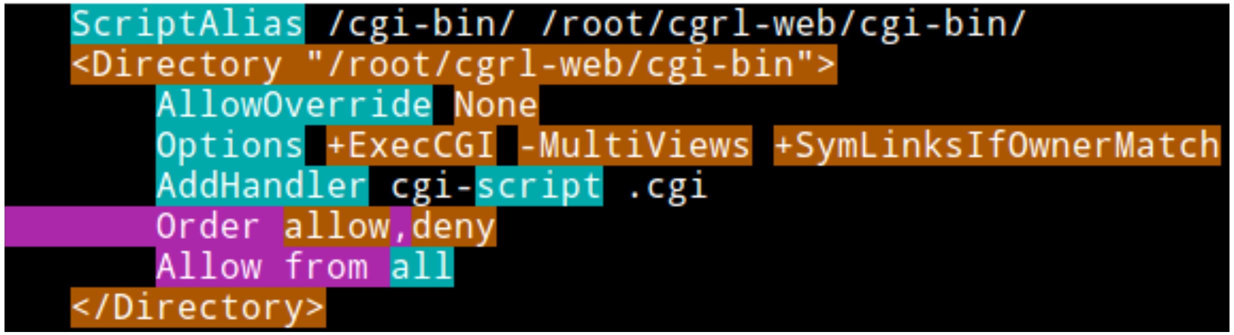
Apache Tutorial: Dynamic Content with CGI

Introduction

Related Modules	Related Directives
mod_alias	AddHandler
mod_cgi	Options
	ScriptAlias

http://httpd.apache.org/docs/2.0/howto/cgi.html

- Check the lab
  for an example!
  - www.cgrl.edu/cgi-bin/env.cgi
  - www.cgrl.edu/cgi-bin/interact.cgi



```
   ScriptAlias /cgi-bin/ /root/cgrl-web/cgi-bin/
   <Directory "/root/cgrl-web/cgi-bin">
       AllowOverride None
       Options +ExecCGI -MultiViews +SymLinksIfOwnerMatch
       AddHandler cgi-script .cgi
       Order allow,deny
       Allow from all
   </Directory>
```

mod_fcgid

FastCGI

- *CGI: every time you access to a page, you call a program whose output generate the HTTP response*
  - *Launching/Quitting one program per request could cost a lot!*
- *mod_fcgid starts a sufficient number instances of the program to handle concurrent requests, and these programs remain running to handle further incoming requests.*
  - *Significantly faster!*