Web Publishing with Athena

Information Services & Technology
Some Terminology

• URI - Uniform Resource Identifier
  – Identifies a resource

• URL - Uniform Resource Locator
  – Tells you where it is and how to get it

• DNS - Domain Name Service
  – Turns something.mit.edu into a numeric address

• CGI - Common Gateway Interface
  – Allows programs to generate web content dynamically
URL


scheme  domain  path  fragment
Relative vs. Absolute

• http://web.mit.edu/ist/helpdesk
  – Absolute URL

• /ist/helpdesk/minicourses
  – Absolute path, but relative to the *scheme* and *domain* portion of whatever URL is active

• helpdesk/minicourses
  – Relative path
DNS

• All computers on the internet identified by a numerical address (e.g. 18.9.22.69)
• DNS maintains a mapping of IP addresses and hostnames (and other things)
  – Record Types
    • A record - Address for a hostname
    • CNAME - Canonical name for a hostname
Hostnames at MIT

• something.mit.edu
  – A hostname, not a "subdomain"

• The hostname can point to an IP address controlled by you (e.g. in your dorm)
  – Request at http://rcc.mit.edu

• Or can be an alias ("CNAME") for another hostname (e.g. scripts.mit.edu)
What actually happens?

GET / HTTP/1.1
Host: web.mit.edu

HTTP/1.1 200 OK
HTTP Response Codes

• 200 - OK

• 301 - Moved Permanently

• 302 - Found (Moved Temporarily)

• 403 - Forbidden

• 404 - Not Found
Generating the content

• Commercial software
  – Adobe Dreamweaver
  – Microsoft Word
• Free Software
  – OpenOffice/LibreOffice
  – Amaya
  – Mozilla/SeaMonkey Composer
• By hand
## Hosting a website

<table>
<thead>
<tr>
<th>web.mit.edu</th>
<th>scripts.mit.edu</th>
<th>external hosting provider</th>
<th>your own web server at MIT</th>
</tr>
</thead>
</table>

- **Ease of maintenance**
- **Flexibility**
web.mit.edu

• Allows static content
  – HTML files, images, PDF, Flash animations, movies, etc
  – JavaScript, Style sheets (CSS), etc

• Minimal web form support (e-mail only)
  – suitable for "guest books", "comment forms", etc.

• Some support for server directives and server-side includes
Review of AFS permissions

• Per-directory basis
• Permissions can be assigned to users or groups ("Moira lists")
• List of permissions for a directory called Access Control List (ACL)
• 7 modes (can be combined)
  – Read, List, Insert, Delete, Write, lock, Administer
  – or the aliases read, write, all, or none
Viewing Permissions

• `fs listacl -dir directory`

• Can be abbreviated
  
  – `fs la directory`
  
  – If directory omitted, defaults to current working directory

• Example:

  joeuser@mint-square:~$ fs la
  Access list for . is
  Normal rights:
    system:expunge ld
    system:anyuser l
    joeuser rlidwka
Granting Permission

- `fs setacl -dir directory -acl entity mode`
  - `directory` - a relative or absolute path
  - `entity` - a username or system:group
  - `mode` - e.g. rl or aliases like read

- Can be abbreviated:
  `fs sa directory entity mode`

- Example
  - `mkdir shared`
  - `fs sa shared jruser read`
Special AFS Entities

- `system:anyuser`
  - Anyone, anywhere
  - required for web access to a directory
    - parent directories must be listable, e.g. `system:anyuser l`

- `system:authuser`
  - Anyone with an Athena account
  - Applies to AFS access only, not web access
Your URL

- http://web.mit.edu/joeuser/www/
- http://www.mit.edu/~joeuser/
- Corresponds to /mit/joeuser/www/
index.html

- Web server looks for this file in directories
- If it's there, display it
- If not, display a listing of the directory
htaccess.mit

• Many web servers use ~/.htaccess to control how a directory is served
• web.mit.edu uses ~/.htaccess.mit and supports a subset of features supported by ~/.htaccess
• When you view a URL, the server looks for .htaccess.mit in that directory
  – And its parent directories
Restricting Access

• Create .htaccess.mit file
  
  <limit GET>
  require valid-user
  </limit>

• fs sa . system:anyuser none

• fs sa . system:htaccess.mit read
system:htaccess.mit

• A special group
• Allows the web servers access to your files
  – But if there's no .htaccess.mit file to tell them what to do, they'll just serve up your files, without restriction
• If you add system:htaccess.mit to an ACL, add an .htaccess.mit file
  – Corollary: If you delete .htaccess.mit, remove system:htaccess.mit
Limiting to users and groups

• Users
  
  <limit GET>
  require user joeuser
  </limit>

• Groups
  – Must be "group" in Moira
  
  <limit GET>
  require group myfriends
  </limit>
Caveats

• Can restrict to users or groups, not both
• **Always** test your restrictions
  – Use a browser without certificates
  – Ask a friend who is not on the list to try it

• When you create a new directory, permissions are inherited
  – Always double-check with `fs ls`
Custom Error Documents

- In .htaccess.mit

  ErrorDocument 404 /joeuser/www/error.html

  ErrorDocument 404 http://www.harvard.edu

  ErrorDocument 404 "Your princess is in another castle"

  ErrorDocument 403 /joeuser/www/notallowed.html
Custom Error Documents

• The document must exist
• If it's a URL, the URL must not redirect you somewhere else
• Best to use full AFS path so it works from all servers (web.mit.edu, www.mit.edu, stuff.mit.edu)
  – /afs/athena.mit.edu/user/j/o/joeuser/www/
Redirecting

• .htaccess.mit

  Redirect 301 /joeuser/www http://joeuser.com

  Redirect 301 /~joeuser http://joeuser.com

• Use 302 instead to prevent Google from changing the URL in listings
Other Tricks

• `.htaccess` is the standard file used by the Apache web server to control some things
• `.htaccess.mit` is our version
  – Some things work, some things don't
  – Not ever guaranteed to work for anything but certificate restrictions and ErrorDocument
• Recommended that you edit on Athena, not on your own computer
E-mail Forms

• CGIemail

• You create a web form and a corresponding text file
  – values from form filled in to special fields in text file; result e-mailed to address you specify

• Good for comment forms, reservation forms, guestbooks, etc.

• Anything more advanced requires scripts.mit.edu
scripts.mit.edu

- Dynamic content (wiki, blog, etc)
- Support for virtually any script in any scripting language
Services

• Web scripts (e.g. CGI)
• Cron scripts
• Mail scripts
• SQL database service (MySQL)
• Autoinstallers
  – Wordpress, MediaWiki, Trac, phpBB, Django, Gallery2
Getting Started

• Sign up:
  
  add scripts

scripts

• Can sign up your locker, or a group locker that you control
  
  – If you're on the AFS ACL for the root of the locker
scripts.mit.edu

• Content goes in ~/web_scripts
  – Special permissions set here to allow access by Scripts servers

• Your URL:
  – http://joeuser.scripts.mit.edu
  – Can't be accessed from http://web.mit.edu
  – Can request something.mit.edu as alias
Some more special AFS entities

• `daemon.scripts`
  – The Scripts servers (sort of)
    • Some black magic is done to ensure that even though joeuser and janeuser have web_scripts directories writable by daemon.scripts, one can't use the other
  – Can assign write permission

• `system:scripts-security-upd`
  – Allows the Scripts servers to make updates to selected software packages
Testing your scripts

• You can log in to the scripts servers to test things
  
  ssh scripts.mit.edu
  ssh -l lockername scripts.mit.edu

• Scripts is load-balanced, so don't count on getting the same server each time
Getting the most out of scripts

• If you're requesting a hostname for a group or project, get the AFS locker first, then request the hostname
  – Staging things in your locker and then moving them breaks things like Wordpress

• Use a Moira list (group) for access to your locker
  – e.g. `fs sa /mit/group system:group-www all`
  – then add/remove people to group-www
Keeping Your Site Secure

• Bad guys always on the lookout for wikis/blogs to fill with spam
• Use good passwords for administering your blog, wiki, etc.
  – And make sure your users do
• Check your site periodically
• Know how to quickly re-create it if necessary
Getting Help

• web.mit.edu
  – http://kb.mit.edu
  – helpdesk@mit.edu

• scripts.mit.edu
  – http://scripts.mit.edu/faq
  – scripts@mit.edu
More Minicourses

- Sep 7: Dotfiles and Shell Customizations

- http://web.mit.edu/ist/helpdesk/minicourses/