

Web Hosting for Fame and Fortune



A Guide to using Apache
as your web-server solution



Why use Apache?



- Extremely portable
- Completely Open Source
- Proven track-record
- Most popular web server on the planet
- Support available from informal and formal channels



Picking the Server Hardware



- Fast Pentium class server
- Load up on ECC RAM (at least 128M)
- Fast Disks
- Choose a 10/100Mbps Ethernet card
- Tape Backup
- UPS



Picking the OS



- Best Choice is an Open Source OS
- East of Installation
- Hardware support
- Robustness and Reliability
- Personal "comfort level"
- *We prefer FreeBSD*



Downloading Apache



- Grab the tarball from www.apache.org or the closest mirror
 - \$ lynx <http://www.apache.org/dist>
- Decompress the tarball
 - \$ gunzip apache_1.3.x.tar.gz
- De-tar the tarball
 - \$ tar xvf apache_1.3.x.tar




Building Apache





- Determine the modules you want
 - Basic Apache modules
 - 3rd party module (mod_php)
- Configure or configure?
 - First time? Use configure
 - Need *suEXEC*? Use configure
 - Like command line? Use Configure







Useful Modules

- **mod_php**
 - Adds very powerful server-side scripting language (`<? echo "Hi World"; ?>`)
 - Fast performer and easy to learn
 - Provides access to various SQL databases
 - Most popular module for Apache
 - <http://www.php.net/>



Useful Modules

- **mod_macro**
 - Streamlines complex conf files
 - `<Macro MyVirtualHost Shost Sport Sdir>`
 - `Listen Sport`
 - `<VirtualHost Shost:Sport>`
 - `DocumentRoot Sdir`
 - `</VirtualHost >`
 - `</Macro>`
 - Use `MyVirtualHost www.apache.org 80 /projects/apache/web`
 - Use `MyVirtualHost www.perl.com 8080 /projects/perlWeb`
 - http://www.cri.ensmp.fr/~coelho/mod_macro/



Useful Modules

- **mod_vhost_alias**
 - Perfect when using hundreds/thousands of vhosts
 - Allows for real-time addition of new vhosts without server restarts
 - Smaller memory footprint
 - Cons:
 - No individual log files
 - Not as comprehensive as true `<VirtualHost>`
 - Somewhat slower



Useful Modules

- **mod_perl**
 - Embeds a true Perl interpreter to Apache
 - Most Perl scripts work with little modification
 - MUCH faster performance for "CGI"
 - Can write Apache modules completely in Perl
 - <http://perl.apache.org/>


Performance Issues - Platform

- Have plenty of RAM
- Use SCSI if you can
- Use separate SCSI buses
- Set aside swap space
- Tune the Operating System (kernel)
- Dedicate server to Apache

Performance Issues - Platform

- Don't allow shell access
- Don't use for development
- Adjust number or size of:
 - File descriptors (`rstat`)
 - Mbufs (`netstat`)
 - Process slots (`maxusers` & `CHILD_MAX`)
 - Listen queue (`SOMAXCONN`)
 - Available RAM (`vmstat`)





Performance Issues - Apache

- Tune basic directives
 - MinSpareServers
 - MaxSpareServers
 - StartServers
 - MaxClients / HARD_SERVER_LIMIT
 - MaxRequestsPerChild
 - ThreadsPerChild



Performance Issues - Apache

- AllowOverride / htaccess
 - Causes expensive "stat" for each directory
 - Set AllowOverride None at top directory
- Disable DNS lookups
 - Latency effects perceived speed of site
- Mutex locking (optimal)



Performance Issues - Apache

- Trim memory usage
 - Modules
 - mod_status / ExtendedStatus Off
 - mod_info
 - DSO
- Trim cycle usage
 - mod_status / ExtendedStatus Off
 - mod_rewrite



Performance Issues - Apache

- Avoid unneeded I/O
 - Logging (LogLevel)
 - Content (mod_mmap_static)
 - Logs on separate drive/bus
- Ensure KeepAlives are active
 - KeepAlive On
 - KeepAliveTimeout
 - KeepAliveRequests



Security Issues - Platform

- Also effect performance: Win Win!
- Keep up to date
- No shell / no cleartext passwords
- FTP setup
- Disable unneeded daemons
 - sendmail / smail / qmail
 - The "r" family
 - tftpd



Security Issues - Apache

- Run server as unprivileged user
 - Use a dedicated account
- Log files and PID file locations
 - Avoid file overwrites
- Protecting file access
 - Symbolic links and DocumentRoot
- Monitor the server and Apache





Security Issues - Apache

- Protect sensitive information

- mod_status and mod_info
- <Location /status>
SetHandler server-status
order deny,allow
deny from all
allow from 192.168.103.10
</Location>
- <Location /status>
<Limit GET>
SetHandler server-status
require valid-user
</Limit>
</Location>



Security Issues - Apache

- Protect about the risks of multiple users and CGI scripts

- cgiwrap
 - CGI scripts are run as the actual "user"
 - Prevents against users over-writing others files
 - "Limits" location of cgi-scripts
 - <http://www.u.m.r.edu/~cgiwrap/>
- suEXEC
 - Allows for per-vhost user/group



Thank you !

- Q & A
- That's all folks!

