Installing and Configuring Apache

Dr. Patrick R. Michaud

October 16, 2004

Overview

- Review of “how the web works”
- Apache configuration files
- Setting up ~user (public_html) URIs
- Handling multiple domains from a single webserver (virtual hosts)
- Home-based web servers

How the web works - URIs 1

Uniform Resource Identifiers

method://server[:port]/path

method – the protocol used to access the resource

server – the server to contact for the resource

port – an optional IP port number to use

path – the identifier for the resource on the server

How the web works - URIs 2

Uniform Resource Identifiers

http://www.pmichaud.com/toast/toast.html

method – “http”

server – “www.pmichaud.com”

port – not specified, uses protocol default (80 for http)

path – “/toast/toast.html”

What the browser does

A browser given a URI such as
"http://www.pmichaud.com/toast/toast.html"

- determines the protocol to be used from the URI
- looks up the IP address of the server
- connects to the server at the address on the specified port
- issues a request for the resource given by the path
- receives a response from the server containing the requested resource
- disconnects
- displays the resource to the user

What the server does

A web server:

- Waits for incoming browser connections
- Obtains the URI path from the browser's request
- Locates the resource requested via the URI path
  - URI paths are not normally filesystem paths!
- Determines how the resource should be processed
- Returns the corresponding resource to the browser

Apache

- Derived from original NCSA httpd
- Named for the collection of patches made to NCSA httpd (“a patchy web server”)
- Now maintained by the Apache Software Foundation
- Market leader for web server software
- Current versions 2.0.52 and 1.3.31

Apache configuration

- Available as part of most Linux distributions
- Simply activate “httpd” or “apache” service
- Main configuration file is “httpd.conf” (held in /etc/httpd/conf on RH/FC2)
- Other configuration files can be placed in /etc/httpd/conf.d
- Log files often in /var/log/httpd or /usr/apache/log

Key configuration directives

- DocumentRoot - location of web content
- ServerName - host name of server
  - Note: This has to correspond with DNS to work properly!
- UserDir - allows requests to /~user/ to serve user's public_html directory
- Alias - change how URIs are mapped to file resources
- ScriptAlias - cause resources to be treated as scripts to be executed
- Redirect - tells browsers to re-request a resource at a different URI

UserDir directive

- Allows requests of the form /~user/somepage to come from user's public_html directory
- Add the following line to httpd.conf:

```
UserDir public_html
```

- Important Notes:
  - Under RH9/FC2 must also comment out

```
UserDir disable
```

  - Each user must create the public_html directory
  - The public_html directory and the user's home directory must have world execute permissions (e.g., chmod 755 or 711)
  - Any files to be served must have world read permissions (e.g., 644)

File permissions and access

- Apache (httpd) process starts as root, opens ports and then switches to another user with limited privileges such as “nobody”, “http”, “www”, or “apache”
- Any files or scripts to be accessed from the web must set permissions to allow access by the webserver process

Virtual domains

- Virtual domains allow a single webserver installation to serve pages from multiple domains
- Each virtual domain may have its own DocumentRoot, ServerName, log files, access controls, etc.
- Two types: IP-based and named-based virtual hosts
  - Today, almost everyone uses name-based virtual hosts

Virtual domain configuration

- In a config file, add

```
NameVirtualHost *

<VirtualHost *:80>
ServerName   dns.name.of.vhost
ServerAlias  other.vhost.dns.names
DocumentRoot /path/to/vhosts/docroot
</VirtualHost>
```

Virtual domain configuration example

- For example, here's a virtual host config I use:

```
NameVirtualHost *

<VirtualHost *:80>
ServerName   www.hri.tamucc.edu
ServerAlias  hri.tamucc.edu
ServerAdmin  hriweb@sci.tamucc.edu
DocumentRoot /a5/hriweb/public_html
</VirtualHost>
```

Important note about virtual domains

- An admin cannot create arbitrary virtual host entries and expect them to “just work”
- Server host names (including virtual hosts) must correlate with DNS or some other address lookup service in order for browsers to find the server

Home-based web servers

- Prerequisites for public web access
  - Stable (static) IP address
  - Bandwidth
  - ISP terms of service
- Good to have a domain name to use instead of publishing IP addresses
- Simply point the domain name to your server's IP address, run Apache on your server, and you're all set!

Security issues

- All standard server security issues apply
- Serving simple static web pages is generally secure
  - Beware of setting up aliases or symlinks that allow arbitrary access to the filesystem
- CGI scripts offer more vulnerabilities, especially if they access external shell programs or applications
- Scripts generally run in the limited “www” or “apache” account

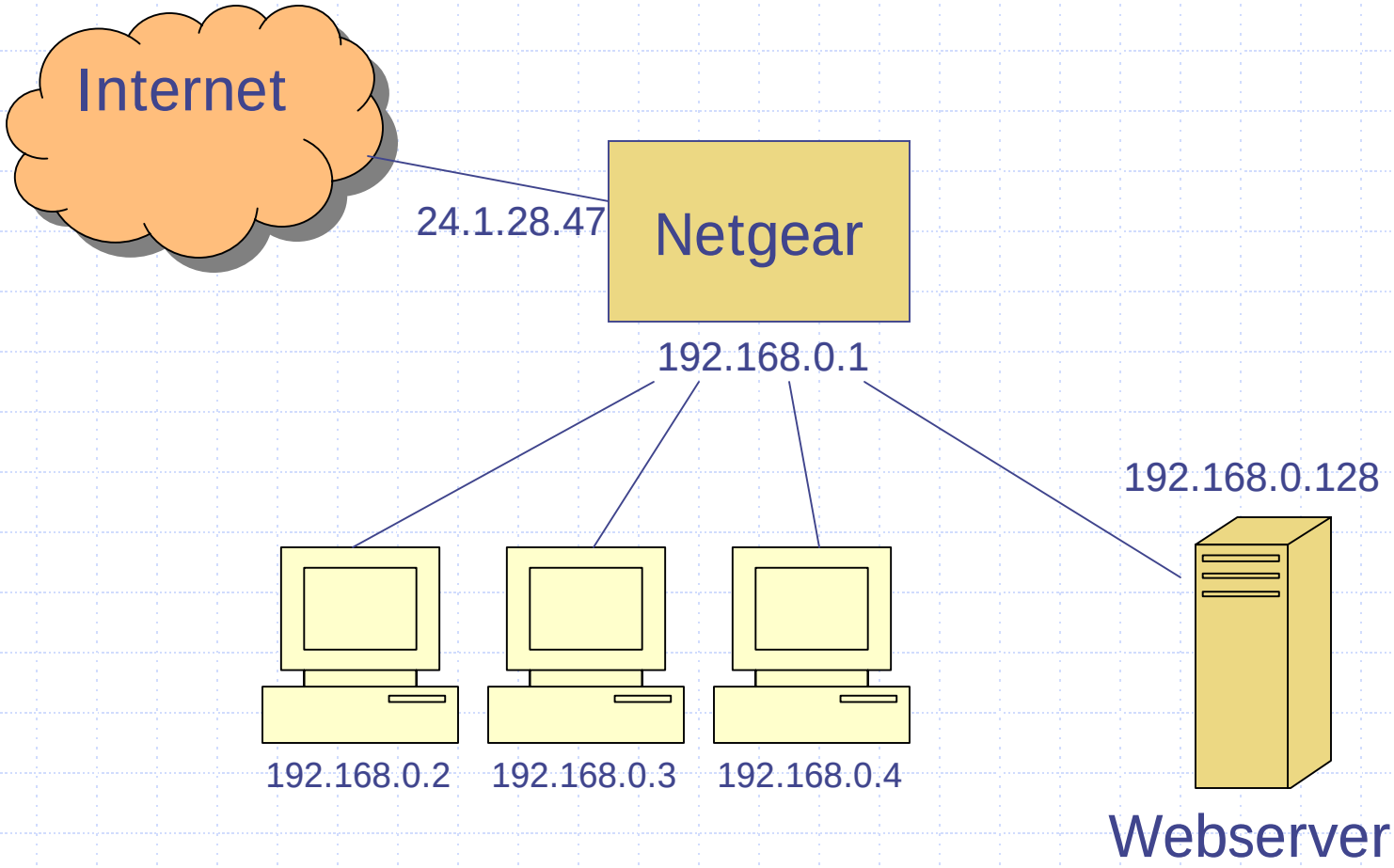
Firewalls/NATs

- If running a webserver from behind a firewall or Network Address Translation (NAT), then you have to build a conduit or forward requests through the firewall to the server
- Often called “Port Forwarding” or just “Forwarding”
- Set port 80 on your firewall to forward to port 80 on your webserver
- Relatively safe as long as the webserver itself is secure

Home-based server example - 1

- I have a Linux box behind a NetGear router/firewall
- The Linux box is configured with an IP address of 192.168.0.128 for my local network
- The NetGear router has an external IP address of 24.1.28.47 (obtained from ISP)
- I own the “patrickmichaud.com” domain name (via Network Solutions)

Home-based server example - 2



Home-based server example - 3

- Set up Apache on the linux box
- Configure the NetGear router to forward incoming requests for port 80 to my linux box at 192.168.0.128 (port 80)
- Set up my DNS service so that “www.patrickmichaud.com” points to the IP address of my router (24.1.28.47)