System Administration

NFS & Web Servers

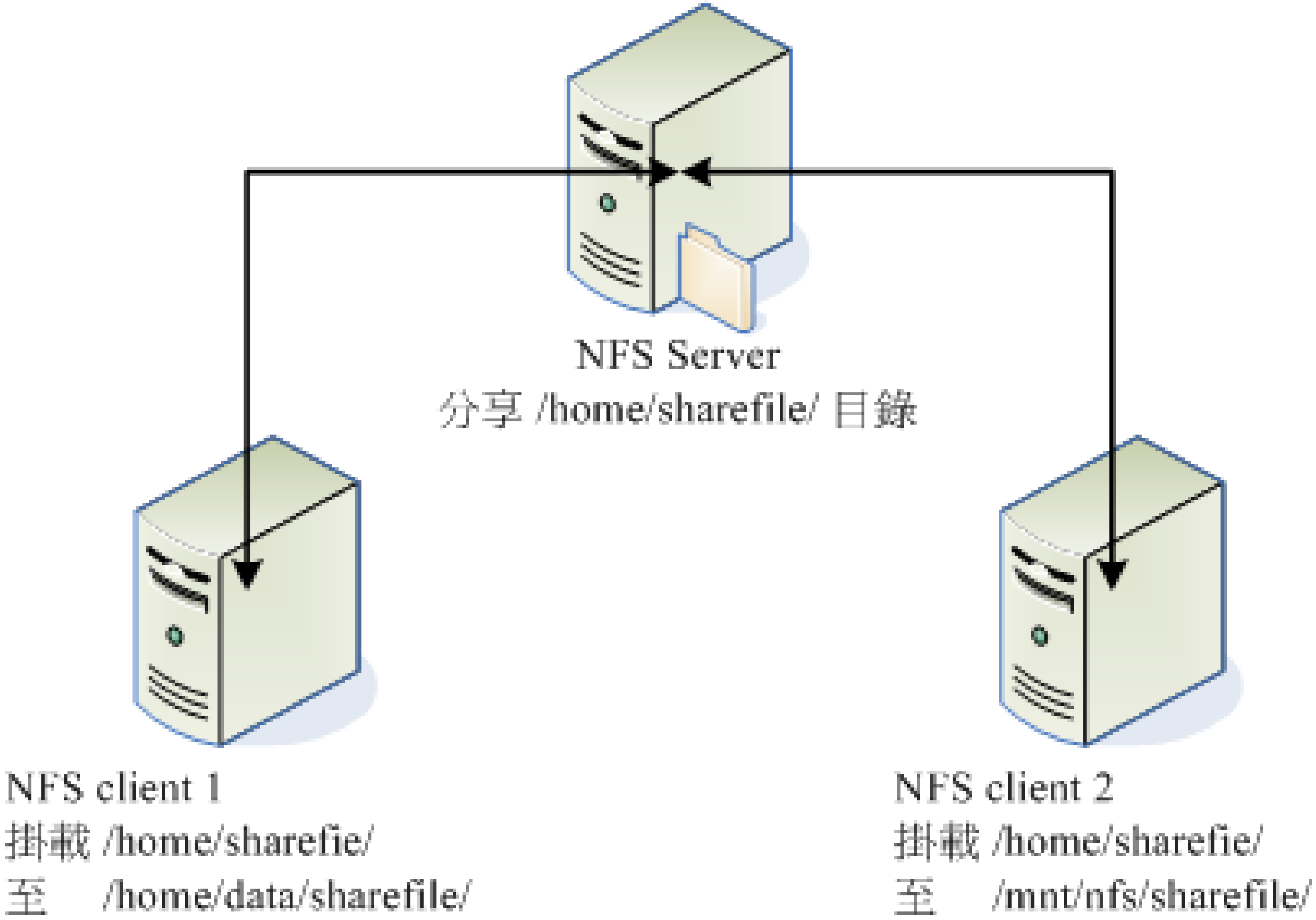
NFS SERVER

File System Operations

- Create file / directory
- Remove file / directory
- List directory
- Open file
- Read from file
- Write to file
- …

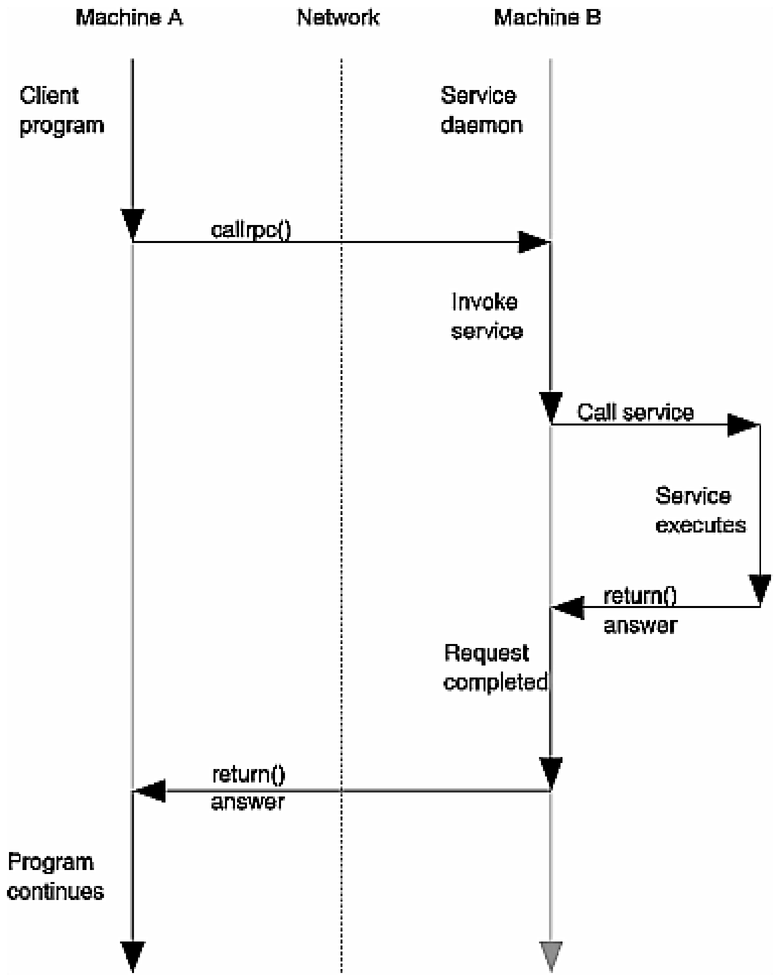
NFS

- Network file system
- File system ops over network
- RPC-based
- IP-based authorization
- Traffic not encrypted



From 鳥哥的Linux私房菜

Remote Procedure Call



From SGI IRIX Network Programming Guide

RPC – Port mapper

- List which port has what service
- “portmap” or “rpcbind”
- List services: rpcinfo -p

NFS Server

- Debian Package: nfs-kernel-server
  - NFS server is implemented in kernel. The package is for support utilities.
- Configuration: /etc/exports
  - See exports(5) manpage
- Show exported paths
  - exportfs
  - showmount

/etc/exports

```
/home   192.168.1.0/24(ro)
Path    Client IP (modifier)
目錄     分享對象     (權限)
```

Client IPs

- IPs (192.168.1.1)
- IP networks (192.168.1.0/24)
- Hostnames (www.csie.ntu.edu.tw)
- Wildcards (*.csie.ntu.edu.tw)

Modifiers

- rw / ro
- sync / async
- root_squash / no_root_squash
  - all_squash

NFS Client

- Debian Package: nfs-common
  - NFS client is implemented in kernel. The package is for support utilities.
- Configuration: /etc/fstab

/etc/fstab

```
# local
/dev/sda1  /   ext4    rw
# nfs
nfs:/home  /home   nfs rw
```

NFS mount options

- fg / bg
- hard / soft
- intr / nointr (No use after 2.6.25)
- rsize= & wsize=
- See nfs(5) manpage

Automount

- Automatically mount filesystem when accessed
  - Unmount after some time unused
- Implemented in kernel
- Package: autofs , autofs5

WEB SERVERS

Apache HTTP Server

- Oldest(?) open source web server
- Most popular according to Netcraft
- Very versatile
  - CGI/FastCGI/WSGI/PSGI/Rack/…
  - mod_perl / mod_python / mod_ruby
  - Many 3rd party modules

Lighttpd

- Lightweight HTTP(S) server
- Single process event driven
  - Early solution to C10k problem
- CGI, FastCGI, SCGI support
- Little new development

Nginx

- Web server
- Reverse proxy
- Load balancing
- Single process event driven
- FastCGI / SCGI / uWSGI
- No CGI

Multi-Processing Module

- prefork
  - 1 process per request
- worker
  - worker thread pool
  - 1 thread per connection
- Event
  - event driven with worker thread pool
  - 1 thread per request
- More info see
  - http://serverfault.com/questions/383526/how-do-i-select-which-apache-mpm-to-use

Apache Packages

- Debian meta-package
  - apache2
- MPM
  - apache2-mpm-*
- 3rd party modules
  - libapache2-mod-*

Basic Configuration

```
# What port to use
Listen 80
# My name
ServerName nasa.csie.ntu.edu.tw
# Run as
User www-data
Group www-data
# PID
PidFile /var/run/apache2.pid
# log
ErrorLog /var/log/apache2/error.log
```

Serving Configuration

```
# Where is /
DocumentRoot /var/www/base
# Permissions for /var/www/base
<Directory /var/www/base>
    Options None
    Order allow,deny
    Allow from all
</Directory>
```

Virtual Hosts

- Serving many sites with 1 server
- IP-based virtual hosts
  - 1 website per IP
- Port-based virtual hosts
  - 1 website per port
- Name-based virtual hosts
  - Many websites per IP/port
  - Differentiate with “Host” header

Name-based Virtual Host

```
NameVirtualHost *
<VirtualHost *>
    DocumentRoot /var/www/www
    ServerName www.csie.ntu.edu.tw
    <Directory /var/www/www>
        Options None
        Order allow,deny
        Allow from all
    </Directory>
</VirtualHost>
```

HTTP Authentication

- 401 Unauthorized
  - Basic
    - Password sent in plaintext
  - Digest
    - Challenge / Response
- mod_auth
  - mod_auth*
  - Many backends
- htpasswd
  - Manage Apache basic password files

HTTP Authentication

```
<Location /locked>
     # Use basic authentication
     AuthType Basic
     # Name to show in dialog
     AuthName “Restricted”
     # Use htpasswd file based
     AuthBasicProvider file
     # Path to password file
     AuthUserFile /etc/apache/users.pw
     # Any user is good
     Require valid-user
</Location>
```

URL Rewrite

- Rewrite a URL internally
  - Make pretty URLs to user
  - Map old URL to new
- Redirect
- Regex
- Conditional

- Enable mod_rewrite

URL Rewrite

```
# Load mod_rewrite
LoadModule rewrite_module
modules/mod_rewrite.so
# Enable rewrite
RewriteEngine On
# rewrite rule
# Redirect /blog?p=N to /new/blog/N
RewriteRule ^/blog?p=(\d+) /new/blog/$1
[R]
```

FastCGI

- 2.2: mod_fastcgi or mod_fcgid
- 2.4: mod_proxy, mod_proxy_fcgi
- Run PHP with FastCGI if you can
    - php-fpm – FastCGI Process Manager

PHP FastCGI for Apache 2.2

```
# Load modules
LoadModule fastcgi_module modules/mod_fastcgi.so

# Associate an alias for the 'fake' fcgi call.
Alias /php5.fcgi /var/www/php5.fcgi
# Assign the 'fake' fcgi to an 'external'
FastCGI Server
FastCGIExternalServer /var/www/php5.fcgi -flush
-host 127.0.0.1:9000

# Create the handler mappings to associate PHP
files with a call to '/php5.fcgi'
AddType application/x-httpd-fastphp5 .php
Action application/x-httpd-fastphp5 /php5.fcgi
```

PHP FastCGI for Apache 2.4

```
# Load modules
LoadModule proxy_module modules/mod_proxy.so
LoadModule proxy_fcgi_module modules/mod_proxy_fcgi.so

# Pass PHP file to FastCGI handler
ProxyPassMatch ^/(.*\.php)$ fcgi://127.0.0.1:9000/var/www/$1
```