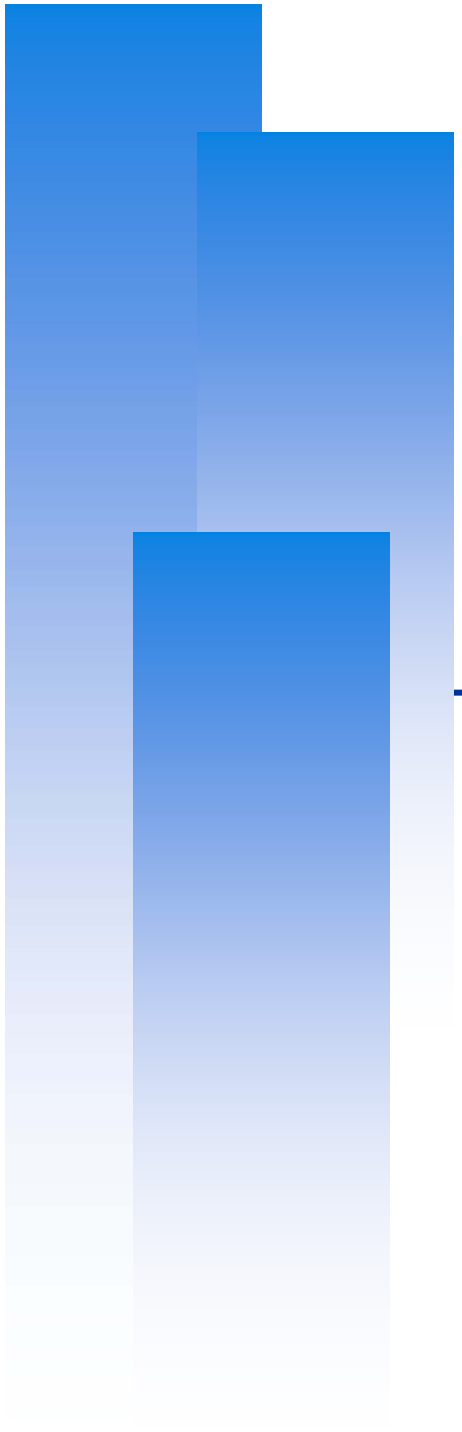


Chapter 18
Sharing System Files

Why share?

- One functioning host depends on hundreds of configuration files
  - But groups of hosts in your network needs more !!
  - Think about you have linux1 ~ linux20, and each year, there are about 250 new students in csie.

What to share?

- Good candidates to share

Filename	Function
/etc/passwd	User account information
/etc/group	UNIX group definitions
/etc/hosts	Maps between IP and hostname
/etc/services	Well-known network service port
/etc/protocols	Maps text names to protocol numbers
/etc/mail/aliases	E-mail alias
/etc/rpc	Lists ID numbers for RPC services
/etc/printcap	Printer information
/etc/termcap	Terminal type information

How to share?

- Keep a master copy of each configuration file in one place and distribute it
  - Push vs. Pull model
  - Copy files around
    - rdist
    - rsync
    - expect
- Let each machine obtain its configuration file from a center server
  - NIS

rdist – push files (1)

- Advantage
  - Simple
  - Preserve owner, group, mode, and modification time of files
- Control file
  - makefile like
  - distfile
  - How to distribute the files
    - [Usage] % rdist [-f distfile] [label]
    - [Format] label: pathnames -> destinations commands

Command	Description
notify *namelist*	Sends email to namelist
except *pathlist*	Do not distribute files in pathlist
except_pat *patternlist*	Do not distribute files that matches patternlist
Special [*pathlist*] *"string"*	Execute an sh "string" command

rdist – push files (2)

- Example

```
SYS_FILES = (/etc/passwd /etc/group /etc/mail/aliases)
GET_ALL = (bsd1 bsd2 linux1)
GET_SOME = (csduty alumni)

all: ${SYS_FILES} -> ${GET_ALL}
        notify chwong@cs.nctu.edu.tw;
        special /etc/mail/aliases "/usr/bin/newaliases";
some: ${SYS_FILES} -> ${GET_SOME}
        except /etc/mail/aliases;
        except_pat /etc/passwd*;
        notify root@cs.nctu.edu.tw;
```

- % rdist
- % rdist –f distfile
- % rdist –f distfile all

rdist – push files (3)

- Disadvantage
  - Based on rsh
    - /.rhosts or /etc/hosts.equiv permit root access
- rdist in FreeBSD
  - /usr/ports/net/rdist6
  - Use more secure “ssh” to replace rsh
    - Use public-key cryptography to do identification
    - Encrypt entire rdist conversation
    - % rdist –P /usr/local/bin/ssh –f myDistfile

expect – pull files (1)

- Write control scripts for interactive programs
- Fundamental expect commands
  - spawn
    - Start up a subprocess to control
  - send
    - Feed input to subprocess
  - expect
    - Take action depending on a subprocess's output
    - expect "pattern" {action}
      - timeout and eof are special patterns
- Our tactic
  - Connect to server using ftp and pull down what we want

expect – pull files (2)

- example

```
spawn /usr/bin/ftp netserver
while 1 { expect {
        "Name*:"        {send "netclient\r"}
        "Password:"     {send "netclientpassword\r"}
        "ftp> "         {break}
        "failed"        {send_user "Can't login.\r"; exit 1}
        timeout         {send_user "Timeout problem.\r"; exit 2}
}}
send "lcd /etc\r"
expect "ftp> " {send "cd pub/sysfiles\r"}
expect "ftp> " {send "get passwd\r"}
expect "ftp> " {send "quit\r"; send_user "\r"}
exit 0
```

NIS –

The Network Information Service (1)

- NIS (YP – Yellow Page)
  - Release by SUN in 1980s
  - For master server
    - System files are kept in original locations and edited as before
    - There will be a server process takes care of availability of these files over the network
  - Data files are hashed and formed a database for lookup efficiency
    - ypmake
    - gdbm hashing library
    - Make + Makefile
  - NIS domain
    - The NIS server and it's clients
  - Multiple NIS server
    - One master NIS server and multiple NIS slave servers

NIS –

The Network Information Service (2)

- /etc/netgroup
  - Group users, machines, nets for easy reference in other system files
  - Can be used in such as /etc/{passwd,group,exports}, /etc/exports
  - [format]

    groupname list-of-members
  - [member-format]

    (hostname, username, nisdomainname)
  - Example of /etc/netgroup

```
adm_user        (,chwong,) (,chiahung,)
adm_cc_cs       (cshome,,) (csduty,,) (csmailgate,,)
sun_cc_cs       (sun1,,) (sun2,,) (sun3,,)
bsd_cc_cs       (bsd1,,) (bsd2,,) (bsd3,,)
linux_cc_cs     (linux1,,) (linux2,,) (linux3,,)
all_cc_cs       adm_cc_cs sun_cc_cs bsd_cc_cs linux_cc_cs
```

NIS –

The Network Information Service (3)

- Prioritizing sources
  - System information can come from many resource
    - Local, NIS, ...
  - Specify the sources that we are going to use and the order of them
- /etc/{passwd, group}
  - +
    - Entire NIS map is included
  - +@
    - Include only certain netgroup
  - +name
    - Include only a single
- /etc/nsswitch.conf
  - FreeBSD has no /etc/nsswitch.conf (/etc/host.conf for hostname lookup)

```
...
passwd:  files nisplus nis
shadow:  files nisplus nis
group:   files nisplus nis
hosts:   files nisplus nis dns
...
```

NIS –

The Network Information Service (4)

- Use netgroup in other system files
  - Example for used in /etc/passwd

```
...
pop:*:68:6:Post Office Owner:/nonexistent:/sbin/nologin
www:*:80:80:World Wide Web Owner:/nonexistent:/sbin/nologin
nobody:*:65534:65534:Unprivileged user:/nonexistent:/sbin/nologin
+@admin-user:*:::::
+:*:::::/usr/local/bin/cs.nologin
```

  - Example for used in /etc/exports

```
/raid     -alldirs –maproot=root mailgate ccserv backup
/raid     -alldirs –maproot=65534 –network 140.113.209 –mask 255.255.255.0
/home     -ro –mapall=nobody –network 140.113.235.0 –mask 255.255.255.0
/usr/src  /usr/obj –maproot=0 bsd_cc_csie
```

NIS –
The Network Information Service (5)

- Advantages of NIS
  - Not necessary for administrator to be aware of NIS internal data format
  - Cross-platform
- Disadvantages of NIS
  - If a slave NIS server is down, the slave's copy may not be updated
    - Periodically poll data
  - Not secure
    - Any host on a network can claim to be NIS Server
    - Any one can read your NIS maps
  - Consume network bandwidth

How NIS works (1)

- NIS directory
  - /var/yp
- NIS Server Map directory
  - In a subdirectory of the NIS directory named for the NIS domain
    - /var/yp/+csie.nis
  - Example:

```
csduty [/var/yp] -chwong- sudo ls +cs.nis/
auto.home              group.byname          netgroup.byuser       publickey.byname
auto.master            hosts.byaddr          netid.byname          rpc.byname
auto.net               hosts.byname          networks.byaddr       rpc.bynumber
auto.user              mail.aliases          networks.byname       services.byname
bootparams             master.passwd.byname  passwd.byname         shadow.byname
ethers.byaddr          master.passwd.byuid   passwd.byuid          sudoers.pwd.byname
ethers.byname          netgroup              protocols.byname      ypservers
group.bygid            netgroup.byhost       protocols.bynumber
```

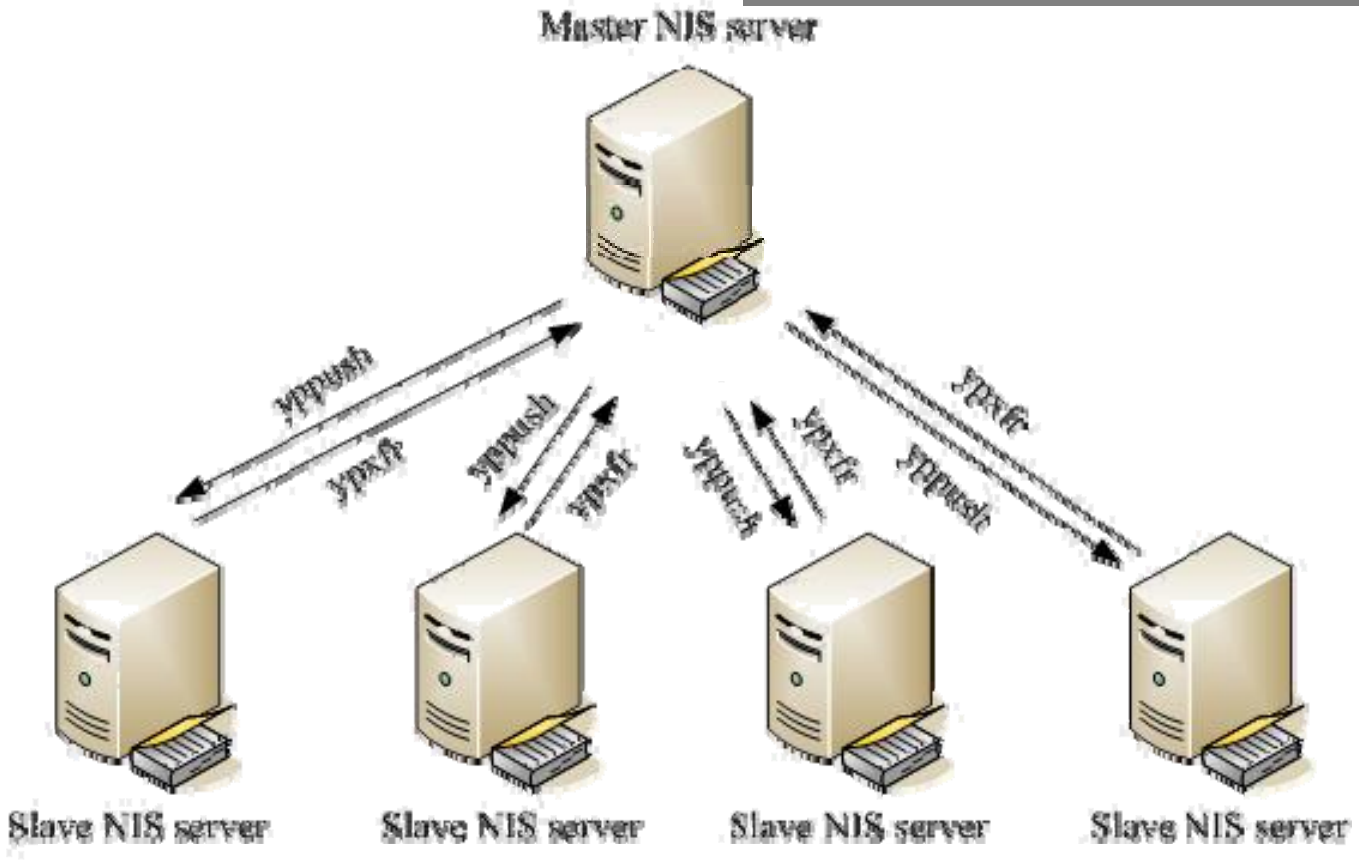
How NIS works (2)

- NIS master server → NIS slave servers
  - "ypxfr" pull command
    - Every NIS slave server runs ypxfr periodically
  - "yppush" push command
    - NIS master server use yppush to instruct each slave to execute ypxfr
  - ypservers special map
    - A list of all NIS slave servers in that NIS domain

How NIS works (3)

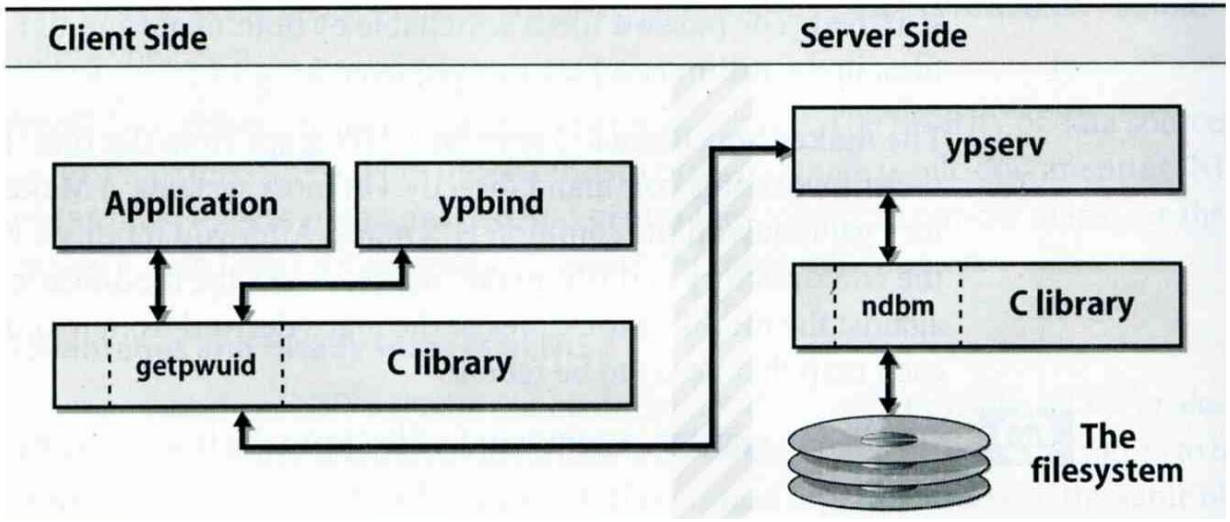
- Example of cs

```
cshome [/var/yp] -chwong- sudo cat ypservers
csduty.cs.nctu.edu.tw
csmailgate.cs.nctu.edu.tw
```



How NIS works (4)

- After all maps are ready
  - Request and response
  - ypserv daemons
    - **Run on NIS servers**
    - **Waiting for NIS requests and answering them by looking up information in maps**
  - ypbind daemons
    - **Run on every machine in NIS domain**
    - **Locate a ypserv and return the identity to the C library, which then contact the server directly**



How NIS works (5)

- NIS commands and daemons

Program	Description
domainname	Set or print name of current NIS domain
makedbm yp_mkdb (FreeBSD)	Build hashed map
ypinit	Configure a host as master or slave
ypset	Let ypbind to bind a particular NIS server
ypwhich	Find out which yp server is using
ypcat	Print the value contained in an NIS map
yppasswd	Change password on the NIS server
ypchfn	Change GECOS information on NIS server
ypchsh	Change login shell on NIS server
yppasswdd	Server daemon for yppasswd,ypchsh,ypchfn

Configuring NIS Servers

- Steps
  - Sequence: Master Server → Slave Servers → each client
- Master Server
  - Set nis domain name
  - Use ypinit to construct a list of slave servers
  - Run ypserv and rpc.yppasswdd daemons
- Slave Servers
  - Set nis domain name
  - Use ypinit to set master NIS server
  - Get NIS maps
- NIS client
  - Set nis domain name
  - Modify /etc/passwd, /etc/group
  - Run ypbind daemons

Configuring NIS Servers – FreeBSD (1)

- Edit /etc/rc.conf
  - If your host does not want to be a NIS client, remove nis_client related entries
  - It is a good idea to force NIS master server to ypbind itself
    - % man ypbind

```
…
# NIS
nisdomainname=“sabsd.nis"
nis_server_enable="YES"
nis_server_flags=""“
nis_client_enable=“YES”
nis_client_flags=“-s –m –S sabsd.nis,sabsd”
nis_yppasswdd_enable="YES"
nis_yppasswdd_flags=“”
…
```

Configuring NIS Servers – FreeBSD (2)

- Initializing the NIS Maps
  - NIS maps are generated from configuration files in /etc with exceptions : /etc/master.passwd, /etc/netgroup, /etc/passwd
  - % cp /etc/master.passwd /var/yp/master.passwd
  - % cp /etc/netgroup /var/yp/netgroup
  - Edit /var/yp/master.passwd , removing all system accounts
  - % cd /var/yp
  - % ypinit –m sabsd.nis
  - % reboot
- Rebuild yp maps whenever the configuration files are changed
- Example
  - When you change /var/yp/master.passwd
  - % cd /var/yp
  - % make

Configuring NIS Servers – FreeBSD (3)

- Makefile of NIS

```
...
YPSRCDIR = /etc
YPDIR = /var/yp
YPMAPDIR = $(YPDIR)/$(DOMAIN)
ETHERS    = $(YPSRCDIR)/ethers     # ethernet addresses (for rarpd)
BOOTPARAMS= $(YPSRCDIR)/bootparams # for booting Sun boxes (bootparamd)
HOSTS     = $(YPSRCDIR)/hosts
NETWORKS  = $(YPSRCDIR)/networks
PROTOCOLS = $(YPSRCDIR)/protocols
RPC       = $(YPSRCDIR)/rpc
SERVICES  = $(YPSRCDIR)/services
SHELLS    = $(YPSRCDIR)/shells
GROUP     = $(YPSRCDIR)/group
ALIASES   = $(YPSRCDIR)/mail/aliases
NETGROUP  = $(YPDIR)/netgroup
PASSWD    = $(YPDIR)/passwd
MASTER    = $(YPDIR)/master.passwd
YPSERVERS = $(YPDIR)/ypservers  # List of all NIS servers for a domain
PUBLICKEY = $(YPSRCDIR)/publickey
NETID     = $(YPSRCDIR)/netid
AMDHOST   = $(YPSRCDIR)/amd.map
...
```

Configuring NIS Servers – FreeBSD (4)

```
sabsd [/home/chwong] -chwong- ps auxww | grep yp
root  367  0.0  0.2  1384 1096  ??  Is  2:57PM  0:00.01 /usr/sbin/ypserv
root  381  0.0  0.2  1400 1152  ??  Is  2:57PM  0:00.00 /usr/sbin/ypbind -s -m -S sabsd.nis,sabsd
root  396  0.0  0.2  1616 1236  ??  Ss  2:57PM  0:00.00 /usr/sbin/rpc.yppasswdd
sabsd [/home/chwong] -chwong- ypwhich
sabsd.cs.nctu.edu.tw
sabsd [/home/chwong] -chwong- ypcat -x
Use "passwd" for "passwd.byname"
Use "master.passwd" for "master.passwd.byname"
Use "group" for "group.byname"
Use "networks" for "networks.byaddr"
Use "hosts" for "hosts.byaddr"
Use "protocols" for "protocols.bynumber"
Use "services" for "services.byname"
Use "aliases" for "mail.aliases"
Use "ethers" for "ethers.byname"
sabsd [/home/chwong] -chwong- ypcat passwd
chiahung:*:1000:1000:chiahung:/home/chiahung:/bin/tcsh
chwong:*:1001:1000:chwong:/home/chwong:/bin/tcsh
sabsd [/home/chwong] -chwong- ypcat hosts
140.113.17.215  sabsd.cs.nctu.edu.tw  sabsd
140.113.17.221  tphp.csie.nctu.edu.tw tphp
```

Configuring NIS Servers – FreeBSD (5)

- NIS client configuration
  - Edit /etc/rc.conf

```
...
# NIS
nisdomainname="sabsd.nis"
nis_client_enable="YES"
nis_client_flags="-s"
...
```

  - Edit /etc/master.passwd (using vipw) and /etc/group

```
...
nobody:*:65534:65534::0:0:Unprivileged user:/nonexistent:/usr/sbin/nologin
+:*:::::::::
```

```
nobody:*:65534:
+:*::
```

  - reboot