

# コンピュータネットワークに関する実験

4 年 情報工学科 2 番

提出者： 1 班 池内 隆一郎

提出締め切り： 平成 28 年 7 月 8 日（金）

提出日： 平成 28 年 7 月 8 日（金）

共同実験者

1 番 安西崇

3 番 石田豊美

4 番 荻野春樹

5 番 茅野哲也

6 番 黒田滉平

7 番 鴻上遼河

# 第1章 アブストラクト

## 第2章 実験結果

### 2.1 実験1 Unixの基本コマンドの習得

#### 2.1.1 課題1-1

ls コマンドの引数に正規表現を指定して、各自に与えられたログインユーザーのホームディレクトリにある環境設定ファイルのみを表示する。図1に正規表現を使用した環境設定ファイルのコマンドと出力を示す。

```

mint02<~>[1]% pwd
/home/staff/mint02
mint02<~>[2]% ls -ldg .[a-zA-Z]*
total 50
-rw----- 1 mint02  mint    0  4月 11 14:38 .Xauthority(*)
-rw-r--r-- 1 mint02  mint 1482  3月 25 2011 .Xdefaults
-rw-r--r-- 1 mint02  mint 2463  3月 25 2011 .Xresources
drwx----- 2 mint02  mint  512  4月 11 17:33 .autosave
-rw-r--r-- 1 mint02  mint  165  3月 25 2011 .bashrc
-rw-r--r-- 1 mint02  mint 4528  3月 25 2011 .canna
-rw-r--r-- 1 mint02  mint 7704  3月 25 2011 .canna-default
-rw-r--r-- 1 mint02  mint  164  3月 25 2011 .cshrc
-rw-r--r-- 1 mint02  mint  276  3月 25 2011 .emacs
-rw-r--r-- 1 mint02  mint   40  3月 25 2011 .emacs-mule
-rw-r--r-- 1 mint02  mint  222  3月 25 2011 .emacs-xemacs
-rw-r--r-- 1 mint02  mint  505  3月 25 2011 .emacs.mew
-rw-r--r-- 1 mint02  mint   36  3月 25 2011 .fvwmrc
-rw-r--r-- 1 mint02  mint   36  3月 25 2011 .fvwmrc
-rw----- 1 mint02  mint 1159  4月 11 17:42 .history
-rw-r--r-- 1 mint02  mint  164  3月 25 2011 .login
-rw-r--r-- 1 mint02  mint    0  3月 25 2011 .nosxsession
-rw-r--r-- 1 mint02  mint 4047  3月  8 2014 .vimrc
drwxr-xr-x 2 mint02  mint  512  3月 25 2011 .xemacs
-rw-r--r-- 1 mint02  mint  340  3月 25 2011 .xmodmap
-rwxr-xr-x 1 mint02  mint 1119  3月 25 2011 .xsession
-rw----- 1 mint02  mint  830  4月 11 17:45 .xsession-errors
-rw-r--r-- 1 mint02  mint  165  3月 25 2011 .zshenv
-rw-r--r-- 1 mint02  mint  162  3月 25 2011 .zshrc

```

図 1: 正規表現を使用した環境設定ファイルの表示

ls コマンドはディレクトリを表示させるコマンドである。オプションは-l で、ファイル名だけでなくタイムスタンプやパーミッション、オーナー情報も表示させる。-d でディレクトリの中身ではなく、そのディレクトリ自体を表示させることができる。-g は-l のファイルオーナーのグループ名を表示しないというオプションだが、-l をつけている場合は付けても付けなくても同じである。正規表現は、"[]" を用いて"[]" 内の文字一文字に一致しているかどうか判定、"\*" でそれを繰り返すことにより"." から始まって小文字、大文字のアルファベットが続くファイルのみを表示させる。

図 2 に簡便化した環境設定ファイルのコマンドと出力を示す。(FreeBSD10.3) また、図 2 のコマンドで環境設定ファイルとカレントディレクトリと親ディレクトリを表示させる。図 1 の

コマンドは長いので、通常普通に使いたいだけの場合は図2のコマンドを使うほうが良い。

```
mint02:~ % ls -ldA .*
drwxr-xr-x  2 i9270  i9270   512 Jul  5 05:36 .
drwxr-xr-x  3 root   wheel   512 Jul  5 05:25 ..
-rw-r--r--  1 i9270  i9270  1066 Jul  5 05:25 .cshrc
-rw-----  1 i9270  i9270   165 Jul  5 05:40 .history
-rw-r--r--  1 i9270  i9270   252 Jul  5 05:25 .login
-rw-r--r--  1 i9270  i9270   163 Jul  5 05:25 .login_conf
-rw-----  1 i9270  i9270   379 Jul  5 05:25 .mail_aliases
-rw-r--r--  1 i9270  i9270   336 Jul  5 05:25 .mailrc
-rw-r--r--  1 i9270  i9270   817 Jul  5 05:25 .profile
-rw-----  1 i9270  i9270   281 Jul  5 05:25 .rhosts
-rw-r--r--  1 i9270  i9270   978 Jul  5 05:25 .shrc
```

図 2: 簡単化した環境設定ファイルの表示

### 2.1.2 課題 1-2

各自与えられたログインユーザーのホームディレクトリに newmember ディレクトリを作成する。図3に newmember ディレクトリを作成するコマンドを示す。

```
mint02<~>[1]% mkdir newmember
mint02<~>[2]%
```

図 3: ディレクトリ作成コマンド

図3の mkdir コマンドによりディレクトリを作成することができる。また、属性は 755 でオーナーは mkdir を実行したユーザーになる。

### 2.1.3 課題 1-3

課題 1-2 で作成した newmember ディレクトリの許可属性を以下に示すように、所有者のみに、「Read 属性」、「Write 属性」、「eXecute 属性」を与え、グループやその他のユーザーからすべての属性を削除する。

```
mint02<~>[1]% chmod 700 newmember  
mint02<~>[2]
```

図 4: 属性変更コマンド

図 4 の `chmod` コマンドで属性を変更する。700 は左から、所有者権限、グループ権限、その他ユーザー権限を表す。また、7 は「Read 属性」、「Write 属性」、「eXecute 属性」のすべてを与えるという意味であり、0 はすべてを与えないという意味である。他にも、`u+rwX` など使い権限変更をできるが、手間であるため、こちらの数字を用いるのが一般的である。

#### 2.1.4 課題 1-4

ログインユーザ名 `nishino` のホームディレクトリの中にある `2016-4I.pwd` ファイルを `newmember` ディレクトリにコピーする。図 5 にファイルをコピーするコマンドを示す。

```
mint02<~>[1]% pwd  
/home/stuff/mint02  
mint02<~>[2]% cp ../nishino/2016-4I.pwd newmember/
```

図 5: ファイルをコピーするコマンド

ファイルやディレクトリをコピーするには、図 5 に示すように `cp` コマンドを用いる。第 1 引数でコピーするファイルやディレクトリを指定し、第 2 引数でコピー先を指定する。図 5 のコマンドの場合、ファイル (`../nishino/2016-4I.pwd`) をディレクトリ (`newmember/`) に移動する。また、ディレクトリをコピーする場合は `-r` オプションを付けることによりディレクトリ内のファイルやディレクトリを再帰的にコピーすることができる。

#### 2.1.5 課題 1-5

課題 1-2 で作成した `newmember` ディレクトリに移動 (カレントディレクトリの変更)。図 6 にディレクトリを変更するコマンドを示す。

```
mint02<~>[1]% pwd
/home/stuff/mint02
mint02<~>[2]% cd newmember/
mint02<~>[3]% pwd
/home/stuff/mint02/newmember
```

図 6: ディレクトリを変更するコマンド

ディレクトリの変更には図6のようにcdコマンドを使用する。引数にディレクトリを指定することにより、指定したディレクトリに移動することができることが確認できる。

### 2.1.6 課題 1-6

grep コマンドを使用して、2016-4I.pwd ファイルから自分を含む実験メンバーを抽出し、その内容をユーザーごとに、その内容をユーザー毎に「innnn.pwd」ファイル (nnnn は、各ユーザの4桁の学籍番号) に保存する。図7にファイルから抜き出し別のファイルに保存するコマンドを示す。

```
mint02<~>[1]% cat 2016-4I.pwd | grep i9269 > i9269.pwd
mint02<~>[2]% cat 2016-4I.pwd | grep i9270 > i9270.pwd
mint02<~>[3]% cat 2016-4I.pwd | grep i9271 > i9271.pwd
mint02<~>[4]% cat 2016-4I.pwd | grep i9274 > i9274.pwd
mint02<~>[5]% cat 2016-4I.pwd | grep i9275 > i9275.pwd
mint02<~>[6]% cat 2016-4I.pwd | grep i9276 > i9276.pwd
mint02<~>[7]% cat 2016-4I.pwd | grep i9277 > i9277.pwd
```

図 7: ファイルから抜き出し別のファイルに保存するコマンド

cat コマンドを用いることにより、標準出力 (コンソール) にファイルを出力することができる。引数はファイル名である。”|”はパイプ機能である。左のコマンドの出力を右に受け渡すことにより、cat コマンドの出力を grep コマンドを用いることによって検索を行い一致した行だけを出力することができる。”>”は指定することができる。今回の場合引数に”>”の後にファイル名がきているのでそのファイルに出力する。また,”>>”を使えば、ファイルに追記することも可能である。ちなみに”<”を使って標準入力を変えることも可能である。図7のコマンドでは,”cat 2016-4I.pwd”で2016-4I.pwdの中身をgrepに渡し、それをgrepがinnnnという文字列で検索を行い、引っかかった行をinnnn.pwdファイルに出力している。

### 2.1.7 課題 1-7

自分を含む各実験メンバーが抽出された「innnn.pwd」ファイルの内容を結合し、group\_1.pwd ファイルに保存する。図 8 にファイルを結合するコマンドを示す。

```
mint02<~>[1]% cat i9269.pwd >> group_1.pwd
mint02<~>[2]% cat i9270.pwd >> group_1.pwd
mint02<~>[3]% cat i9271.pwd >> group_1.pwd
mint02<~>[4]% cat i9274.pwd >> group_1.pwd
mint02<~>[5]% cat i9275.pwd >> group_1.pwd
mint02<~>[6]% cat i9276.pwd >> group_1.pwd
mint02<~>[7]% cat i9277.pwd >> group_1.pwd
```

図 8: ファイル結合のコマンド

”>>” を用いることにファイルに追記することができる。図 8 では、innnn.pwd ファイルの中身を group\_1.pwd に追記していくことによって、group\_1.pwd へファイルを結合していく。

### 2.1.8 課題 1-8

課題 1-7 で作成した group.g.pwd ファイルの名前を members\_master.pwd に変更する。

```
mint02<~>[1]% ls
2016-4I.pwd
mint02<~>[2]% mv group_1.pwd members_master.pwd
mint02<~>[3]% ls
members_master.pwd
```

図 9: ファイル名変更コマンド

名前の変更には mv コマンドを用いる第 1 引数に元のファイル名を指定し、第 2 引数に新しいファイルの名前を指定する。図 9 では group\_1.pwd という名前のファイルを members\_master.pwd という名前に変更していることを確認できる。

### 2.1.9 課題 1-9

members\_master.pwd ファイルの許可属性を所有者のみに「Read 属性」、「Write 属性」を与え、グループやその他のユーザーからすべての属性を削除する。図 10 に許可属性変更のコマンドを示す。



```
mint02<~>[2]% chmod 600 members_master.pwd
```

図 10: 許可属性変更コマンド

2.1.3 と同じく `chmod` を使い許可属性を変更する。「Read 属性」、「Write 属性」だけを与える場合は「6」を使う。図 10 のように「600」を指定することによって、所有者のみに「Read 属性」、「Write 属性」を与えることができる。

### 2.1.10 課題 1-10

`vi` エディタや `ee`, `xemacs` などを利用して、`members_master.pwd` ファイルを完成させる。その際の「ユーザー ID」、「グループ ID」、「ユーザーについての一般的な情報」、「ホームディレクトリ領域」について以下の通りにする。図 11 に `members_master.pwd` ファイルを示す。

- ユーザー ID: 「学籍番号」
- グループ ID: 「yyyyg」(yyyy は西暦、g はグループ番号)
- ユーザーについての一般的な情報: 「4I-nn 氏名(ローマ字)」(nn は 2 桁の出席番号)
- ユーザーディレクトリ: `/home/student/innnn`

```
i9277:*:9277:20161::0:0:4I-07 Kougami Ryouga:/home/student/i9277:/bin/csh
i9275:*:9275:20161::0:0:4I-05 Kayano Tetsuya:/home/student/i9275:/bin/csh
i9271:*:9271:20161::0:0:4I-03 Ishida Toyomi:/home/student/i9271:/bin/csh
i9270:*:9270:20161::0:0:4I-02 Ikeuchi Ryuichirou:/home/student/i9270:
/usr/local/bin/bash
i9269:*:9269:20161::0:0:4I-01 Anzai Shuu:/home/student/i9269:/bin/csh
i9274:*:9274:20161::0:0:4I-04 Ogino Haruki:/home/student/i9274:/bin/csh
i9276:*:9276:20161::0:0:4I-06 Kuroda Kouhei:/home/student/i9276:/bin/csh
```

図 11: `members_master.pwd`

図 11 のようにユーザ ID やグループ ID などのを追加した。ログインシェルに関しては、自分 (i9270) だけ `bash` を用いるため、でふあるとの `/bin/csh` から `/usr/local/bin/bash` へ変更した。

### 2.1.11 課題 1-11

members\_master.pwd ファイル以外の不要になった中間作業ファイルを削除する。図 12 に不要なファイルを削除するコマンドを示す。

```
mint02<~>[1]% ls | grep -v 2016-4I.pwd | xargs rm
```

図 12: 不要なファイル削除コマンド

ファイルを削除するコマンドには rm を使う。xargs は引数にコマンドを指定し、そのコマンドに引数として渡すというコマンドである。これにより、図 12 に示すように ls コマンドでカレントディレクトリのファイルを grep に渡し、grep は”2016-4I.pwd” に一致しない文字列を xargs に渡す。それを xargs は rm に引数として渡すことによって、”2016-4I.pwd” 以外の不要なファイルをすべて削除することができた。ちなみに、grep のオプションの -v はマッチしない行の検索を行う。

### 2.1.12 初級課題

課題 2, 3 を一度に行う。図 13 にディレクトリを許可属性を指定して作成するコマンドを示す。

```
mint02<~>[1]% mkdir -m 700 newmember
```

図 13: ディレクトリを許可属性を指定して作成するコマンド

mkdir コマンドのオプションに -m を用いれば、許可属性を変更することができた。

### 2.1.13 中級課題

課題 6, 8 を一度に行う。図 14 に

```
mint02<~>[1]% cat 2016-4I.pwd | grep -E '9269*|927[0-7]*' >  
members_master_adv_1.pwd
```

図 14: 文字列を検索してファイルに保存するコマンド

図 14 に示すように cat2016-4I を grep に渡して、9269 か 9270 から 9277 に一致する 1 行を members\_master\_adv\_1.pwd に保存する。これにより、課題 6, 7 を同時に行うことができた。

### 2.1.14 上級課題

newmember ディレクトリの中に、ログインユーザー名 nishino のホームディレクトリの中にある 2016-4I.pwd ファイルへのシンボリックリンクを張る。ただし、シンボリックリンクのファイル名は、2016\_4I\_ln.pwd とする。図 15 にシンボリックリンクを作成するコマンドを示す。

```
mint02<~>[1]% ln -s ../nishino/2016-4I.pwd newmember/2016_4I_ln.pwd
```

図 15: シンボリックリンクを作成するコマンド

シンボリックリンクを張るには ln コマンドを使う。ハードリンクではなくシンボリックリンクにするために -s オプションを用いる。第 1 引数にファイル名を指定し、第 2 引数にリンクを指定する。図 15 に示すようにコマンドを実行することによって ../nishino/2016-4I.pwd への newmember/2016\_4I\_ln.pwd を張ることができた。

## 2.2 課題 2 ユーザー登録

### 2.2.1 課題 2-1

vipw コマンドを使用して /etc/master.passwd ファイルに作成した members\_master.pwd ファイルの内容を追加挿入する。図 16 に編集後の /etc/master.passwd ファイルを示す。

```

#
# $FreeBSD: src/etc/master.passwd, v 01.01.02 2016/04/18 20:12:24 kensmith
Exp $
#
root:*:0:0:Charlie &:/root:/bin/csh
toor:*:0:0:Bourne-again Superuser:/root:
daemon:*:1:1:Owner of many system processes:/root:/usr/sbin/nologin
operator:*:2:5:System &:/usr/sbin/nologin
bin:*:3:7:Binaries Commands and Source:/usr/sbin/nologin
tty:*:4:65533:Tty Sandbox:/usr/sbin/nologin
kmem:*:5:65533:KMem Sandbox:/usr/sbin/nologin
games:*:7:13:Games pseudo-user:/usr/games:/usr/sbin/nologin
news:*:8:8:News Subsystem:/usr/sbin/nologin
man:*:9:9:Mister Man Pages:/usr/share/man:/usr/sbin/nologin
ftp:*:14:5:Anonymous Ftp:/var/ftp:/nonexistent
sshd:*:22:22:Secure Shell Daemon:/var/empty:/usr/sbin/nologin
smmisp:*:25:25:Sendmail Submission User:/var/spool/clientmqueue:
    /usr/sbin/nologin
mailnull:*:26:26:Sendmail Default User:/var/spool/mqueue:/usr/sbin/nologin
bind:*:53:53:Bind Sandbox:/usr/sbin/nologin
proxy:*:62:62:Packet Filter pseudo-user:/nonexistent:/usr/sbin/nologin
_pflogd:*:64:64:pflogd privsep user:/var/empty:/usr/sbin/nologin
_dhcp:*:65:65:dhcp programs:/var/empty:/usr/sbin/nologin
uucp:*:66:66:UUCP pseudo-user:/var/spool/uucppublic:
    /usr/local/libexec/uucp/uucico
pop:*:68:6:Post Office Owner:/nonexistent:/usr/sbin/nologin
wnn:*:69:7:Wnn:/nonexistent:/nonexistent
www:*:80:80:World Wide Web Owner:/nonexistent:/usr/sbin/nologin
cups:*:193:193:CUPS Owner:/nonexistent:/usr/sbin/nologin
messagebus:*:556:556:D-BUS Daemon User:/nonexistent:/usr/sbin/nologin
avahi:*:558:558:Avahi Daemon User:/nonexistent:/sbin/nologin
haldaemon:*:560:560:HAL Daemon User:/nonexistent:/usr/sbin/nologin
polkit:*:562:562:PolicyKit User:/nonexistent:/usr/sbin/nologin
pulse:*:563:563:PulseAudio System User:/nonexistent:/usr/sbin/nologin
moodle:*:900:900:Web Course Management User:/nonexistent:/usr/sbin/nologin
nishino:*:1001:1001:NISHINO Takayuki:/home/staff/nishino:/usr/local/bin/zsh
info:*:2001:2000:Enviroment Model User:/home/staff/info:/usr/sbin/nologin
exp:*:3001:3000:Network Experiment User:/home/staff/exp:/bin/csh

```

```
mint01:*:4001:4000:Network Experiment User 01:/home/staff/mint01:/bin/csh
mint02:*:4002:4000:Network Experiment User 02:/home/staff/mint02:/bin/csh
mint03:*:4003:4000:Network Experiment User 03:/home/staff/mint03:/bin/csh
mint04:*:4004:4000:Network Experiment User 04:/home/staff/mint04:/bin/csh
mint05:*:4005:4000:Network Experiment User 05:/home/staff/mint05:/bin/csh
mint06:*:4006:4000:Network Experiment User 06:/home/staff/mint06:/bin/csh
mint07:*:4007:4000:Network Experiment User 07:/home/staff/mint07:/bin/csh
mint08:*:4008:4000:Network Experiment User 08:/home/staff/mint08:/bin/csh
mint09:*:4009:4000:Network Experiment User 09:/home/staff/mint09:/bin/csh
mint10:*:4010:4000:Network Experiment User 10:/home/staff/mint10:/bin/csh
i9277:*:9277:20161:4I-07 Kougami Ryouga:/home/students/i9277:/bin/csh
i9275:*:9275:20161:4I-05 Kayano Tetsuya:/home/students/i9275:/bin/csh
i9271:*:9271:20161:4I-03 Ishida Toyomi:/home/students/i9271:/bin/csh
i9270:*:9270:20161:4I-02 Ikeuchi Ryuichirou:/home/students/i9270:
    /usr/local/bin/bash
i9269:*:9269:20161:4I-01 Anzai Shuu:/home/students/i9269:/bin/csh
i9274:*:9274:20161:4I-04 Ogino Haruki:/home/students/i9274:/bin/csh
i9276:*:9276:20161:4I-06 Kuroda Kouhei:/home/students/i9276:/bin/csh
nobody:*:65534:65534:Unprivileged user:/nonexistent:/usr/sbin/nologin
```

図 16: /etc/master.passwd

vipw コマンドはエディタは vi で /etc/master.passwd ファイルを編集するコマンドである。また、このときにパスワードファイルには適切なロックがかかるようになっている。図 16 のようにユーザー ID の順番になるように members\_master.pwd を挿入した。ちなみに、adduser コマンドでユーザーを追加すれば、対話的にユーザーを追加することができる。

## 2.2.2 課題 2-2

/etc/group ファイルを完成させる。wheel に各自のログイン名を追加し、また、実験メンバーのグループ名を追加する。図 17 に編集後の /etc/group ファイルを示す。

```
#
# $FreeBSD: src/etc/group, v 01.01.02 2016/04/18 20:12:24 kensmith Exp $
#
wheel:*:0:root,nishino,i9270
daemon:*:1:
kmem:*:2:nishino
sys:*:3:
tty:*:4:
operator:*:5:root,nishino
mail:*:6:
bin:*:7:
news:*:8:
man:*:9:
games:*:13:
ftp:*:14:nishino
staff:*:20:nishino
sshd:*:22:
smmsp:*:25:
mailnull:*:26:
guest:*:31:
bind:*:53:
proxy:*:62:
authpf:*:63:
_pflogd:*:64:
_dhcp:*:65:
uucp:*:66:
dialer:*:68:nishino
network:*:69:
audit:*:77:
www:*:80:
cups:*:193:
messagebus:*:556:
pulse-rt:*:557:
avahi:*:558:
haldaemon:*:560:
polkit:*:562:
pulse:*:563:
pulse-access:*:564:
cms:*:900:moodle,nishino
```

```
master*:1000:
info*:2000:
exp*:3000:
mint*:4000:
Falcon_Panch*:20161:i9270,i9269,i9271,i9274,i9275,i9276,i9277
nogroup*:65533:
nobody*:65534:
```

図 17: /etc/group

図 17 のように wheel グループに自分のユーザー (i9270) を追加した。これにより、su コマンドで root 権限を取得することが可能になる。Falcon\_Panch に班の人のユーザーをすべて入れた。今回は vi で直接書き込んだが、vipw 同様、vigr というコマンドを用いてグループ変更を行うことも可能であった。adduser コマンドでユーザーを追加するときにグループに追加することも可能である。

### 2.2.3 課題 2-3

各メンバーのホームディレクトリを作成する。図 18 にホームディレクトリを作成するコマンドを作成する。

```
mint02</home/students>[1]% mkdir i9269 i9270 i9271 i9274 i9275 i9276 i9277
```

図 18: ホームディレクトリを作成するコマンド

図 18 のように mkdir コマンドを用いて引数にディレクトリ名を複数入れることによって一度にフォルダを作成することができた。

### 2.2.4 課題 2-4

各メンバーの所有者と所属グループを変更する。図 19 に所属者と所属グループを変更するコマンドを示す。

```
mint02</home/students>[1]% chown i9269:i9269 i9269
mint02</home/students>[2]% chown i9270:i9270 i9270
mint02</home/students>[3]% chown i9271:i9271 i9271
mint02</home/students>[4]% chown i9274:i9274 i9274
mint02</home/students>[5]% chown i9275:i9275 i9275
mint02</home/students>[6]% chown i9276:i9276 i9276
mint02</home/students>[7]% chown i9277:i9277 i9277
```

図 19: ホームディレクトリを作成するコマンド

chmod コマンドは第 1 引数に所有者と所属グループをコロン (:) で分けて指定し、第 2 引数にフォルダ名を指定する。図 19 のように chown コマンドを使うことによって所有者と所属グループを変更することができた。

### 2.2.5 課題 2-5

ログインユーザー名 info のホームディレクトリにある環境設定ファイル群を課題 1-1 で使用した正規表現を用いて各実験メンバーのホームディレクトリにコピーする。図 20 に環境設定ファイルをコピーするコマンドを示す。

```
mint02</home/students>[1]% cd ../staff/info
mint02</home/staff/info>[2]% cp .[a-zA-Z]* ../../students/i9269
mint02</home/staff/info>[3]% cp .[a-zA-Z]* ../../students/i9270
mint02</home/staff/info>[4]% cp .[a-zA-Z]* ../../students/i9271
mint02</home/staff/info>[5]% cp .[a-zA-Z]* ../../students/i9274
mint02</home/staff/info>[6]% cp .[a-zA-Z]* ../../students/i9275
mint02</home/staff/info>[7]% cp .[a-zA-Z]* ../../students/i9276
mint02</home/staff/info>[8]% cp .[a-zA-Z]* ../../students/i9277
```

図 20: 環境設定ファイルをコピーするコマンド

図 20 のように課題 1-1 で使用した正規表現とコピーコマンドを用いることによって各メンバーのホームディレクトリに環境設定ファイルをコピーすることができた。

### 2.2.6 課題 2-6

課題 2-5 でコピーした環境設定ファイルの所有者と所属グループを変更する。図 21 に所有者と所属グループを変更するコマンドを示す。



```
mint02</home/students/i9269>[1]% chown i9269:i9269 .[a-zA-Z]*
mint02</home/students/i9270>[1]% chown i9270:i9270 .[a-zA-Z]*
mint02</home/students/i9271>[1]% chown i9271:i9271 .[a-zA-Z]*
mint02</home/students/i9274>[1]% chown i9274:i9274 .[a-zA-Z]*
mint02</home/students/i9275>[1]% chown i9275:i9275 .[a-zA-Z]*
mint02</home/students/i9276>[1]% chown i9276:i9276 .[a-zA-Z]*
mint02</home/students/i9277>[1]% chown i9277:i9277 .[a-zA-Z]*
```

図 21: 所有者と所有グループを変更するコマンド

図 21 のように環境設定ファイルを正規表現と `chown` コマンドを用いて環境設定ファイルの所有者と所有グループを変更できた。

### 2.2.7 課題 2-7

各一般ユーザーに仮パスワード「4I-innnn.pwd」(nnnn は各ユーザーの学生番号 4 桁)を設定する。図 22 にユーザーパスワード変更コマンドを示す。

```
mint02<~>[1]% passwd i9269
New Password:
Retype New Password:
mint02<~>[1]% passwd i9270
New Password:
Retype New Password:
mint02<~>[1]% passwd i9271
New Password:
Retype New Password:
mint02<~>[1]% passwd i9274
New Password:
Retype New Password:
mint02<~>[1]% passwd i9275
New Password:
Retype New Password:
mint02<~>[1]% passwd i9276
New Password:
Retype New Password:
mint02<~>[1]% passwd i9277
New Password:
Retype New Password:
```

図 22: ユーザーパスワード変更コマンド

passwd コマンドを用いることによってパスワードを変更することができる。引数にはユーザー名を指定する。図 22 のように各ユーザー名を引数に指定することによってパスワードの変更を行うことができた。また、パスワードはキーボードから入力しても画面にはでないという仕様になっていることが確認できた。

### 2.2.8 課題 2-8

管理するコンピュータから「ログアウト」を行い、登録が完了した自分の一般ユーザーアカウントにて、ログインを行い仮パスワードの変更を行った。ログアウトとログインは GUI で行った。仮パスワードの変更は図 22 と同じように passwd コマンドを用いてパスワードを変更した。

### 2.2.9 課題 2-9

ネットワークケーブル (EIA/TIA-568B ストレートケーブル) を作成した。

## 2.2.10 上級課題

Shell の繰り返しコマンド (foreach や for など) を使用して課題 2-3 から 2-6 を一度に行う。図 23 にシェルのソースコードを示す。

```
#!/usr/bin/bash

user_name=(i9269 i9270 i9271 i9274 i9275 i9276 i9277)

for name in ${user_name[@]}; do
    mkdir -p /home/2nd_week/$name
    chown $name:$name /home/2nd_week/$name
    cp /home/staff/info/. [a-zA-Z]* /home/2nd_week/$name/
    chown $name:$name /home/2nd_week/$name/. [a-zA-Z]*
done
```

図 23: 課題 2-3 から 2-6 までを一度に行うシェル

シェルは bash を用いた。これにより配列を簡単に作成することができた。図 23 のように for 文を回し、変数 name にユーザー名を 1 つずつ入れていくところにより、各ユーザーの設定を行うことができた。mkdir のオプションの -p はサブディレクトリごとディレクトリを作成するコマンドであり、2nd\_week ディレクトリをあらかじめ作っていなくても作成できる。シェル内のディレクトリやファイルの指定は、どこから図 23 のシェルが実行されても使えるように絶対的に指定した。

## ネットワークの基礎知識

### 3.1.1 課題 3-1

ネットワークを組むにあたって、ネットワーク設定情報を表にまとめる。表にネットワーク設定情報を示す。

表1 ネットワーク設定情報(1)

ホスト名	管理者	IPアドレス	ネットマスク	ネットワーク アドレス	デフォルトゲート ウェイ
Falcon_Pantz1	i9269	10.6.1.1	255.0.0.0	10.0.0.0	10.0.0.1
Falcon_Pantz2	i9270	10.6.1.2	255.0.0.0	10.0.0.0	10.0.0.1
Falcon_Pantz3	i9271	10.6.1.3	255.0.0.0	10.0.0.0	10.0.0.1
Falcon_Pantz4	i9274	10.6.1.4	255.0.0.0	10.0.0.0	10.0.0.1
Falcon_Pantz5	i9275	10.6.1.5	255.0.0.0	10.0.0.0	10.0.0.1
Falcon_Pantz6	i9276	10.6.1.6	255.0.0.0	10.0.0.0	10.0.0.1
Falcon_Pantz7	i9277	10.6.1.7	255.0.0.0	10.0.0.0	10.0.0.1

ホスト名は Falcon\_pantz で IP アドレスは 10.6.1.n(n は班のメンバーの番号) を用いた。ネットマスクは上位 8bit で、ネットワークアドレスは 10.0.0.0 を用いた。デフォルトゲートウェイはルーター (c2611r) へつなぐのでその ip アドレス 10.0.0.1 を用いた。

### 3.1.2 課題 3-2

図 24 にネットワーク図を示す。図 24 のように表すネットワークを構築する。また、図 25, 26, 27 に編集後の設定ファイル、rc.conf, networks, hosts ファイルを示す。

```

#
# -- sysinstall generated deltas -- # Mon Apr 25 01:01:02 2016
# Created: Mon Apr 25 01:01:02 2016
# Enable network daemons for user convenience.
# Please make all changes to this file, not to /etc/defaults/rc.conf.
# This file now contains just the overrides from /etc/defaults/rc.conf.
#

check_quotas="NO"
defaultrouter="10.0.0.1"
hostname="falcon_pantz2.exp.info.nara-k.ac.jp"
ifconfig_xl0="inet 10.6.1.2 netmask 255.0.0.0"
sendmail_enable="NONE"
ntpd_enable="YES"
ntpd_flags="-p /var/run/ntpd.pid -f /var/db/ntpd.drift"
inetd_enable="YES"
sshd_enable="YES"

nfs_client_enable="YES"
linux_enable="YES"
sendmail_enable="NONE"

keymap="jp.106x"
moused_enable="YES"
hald_enable="YES"
dbus_enable="YES"
polkitd_enable="YES"
canna_enable="YES"
canna_flags="-u bin -inet"

```

図 24: rc.conf

```
#  
# $FreeBSD: src/etc/networks, v 01.01.02 2016/04/25 20:12:24 kensmith Exp $  
#  @(#)networks      5.1 (Berkeley) 6/30/90  
#  
# Your Local Networks Database  
#  
pantznet      10.0.0.0
```

图 25: networks

```

#
# $FreeBSD: src/etc/hosts, v 01.01.02 2016/04/25 20:12:24 kensmith Exp $
#      @(#)hosts      5.1 (Berkeley) 6/30/90
#      /etc/hosts (Host Database)
#
# This file should contain the addresses and aliases for local hosts that
# share this file.  Replace 'my.domain' below with the domainname of your
# machine.
#
# In the presence of the domain name service or NIS, this file may
# not be consulted at all; see /etc/nsswitch.conf for the resolution order.
#
#
# IP_address      FQDN  Hostname  [alias1]  [alias2 ...]
#
127.0.0.1    localhost.exp.info.nara-k.ac.jp localhost  # IPv4 loopback
::1         localhost.exp.info.nara-k.ac.jp localhost  # IPv6 loopback
#
#      PC
#
10.6.1.1     falcon_pantz1.exp.info.nara-k.ac.jp falcon_pantz1
10.6.1.2     falcon_pantz2.exp.info.nara-k.ac.jp falcon_pantz2
10.6.1.3     falcon_pantz3.exp.info.nara-k.ac.jp falcon_pantz3
10.6.1.4     falcon_pantz4.exp.info.nara-k.ac.jp falcon_pantz4
10.6.1.5     falcon_pantz5.exp.info.nara-k.ac.jp falcon_pantz5
10.6.1.6     falcon_pantz6.exp.info.nara-k.ac.jp falcon_pantz6
10.6.1.7     falcon_pantz7.exp.info.nara-k.ac.jp falcon_pantz7
#
#      Network Printer
#
172.16.255.1  colorps.exp.info.nara-k.ac.jp colorps      # Color Printer
#
#      Server / Gateway
#
10.255.255.250 pegasus.exp.info.nara-k.ac.jp pegasus      # FreeBSD
10.255.255.251 sleipnir.exp.info.nara-k.ac.jp sleipnir   # FreeBSD
10.255.255.252 phoenix.exp.info.nara-k.ac.jp phoenix     # FreeBSD
10.255.255.253 unicorn.exp.info.nara-k.ac.jp unicorn      # FreeBSD
#
10.0.0.1      c2611r.exp.info.nara-k.ac.jp c2611r gateway # CISCO 2611R

```

rc.confは図 26 に示すように IP アドレス、ホスト名、サブネットマスクを設定した。networks では図 27 に示すように Pantznet という名前とネットワークアドレスを設定した。hosts では図 28 に示すように falcon\_pantzN(N は班のメンバーの番号) を設定した。これにより、ネットワークの構築を完成させることができた。

### 3.1.3 課題 3-3

ログインホストから localhost に対して、以下に示す 2 通りの条件下における ping コマンドの実行結果を示す。条件 (1) LAN ケーブルを接続していない状態条件 (2) LAN ケーブルを接続している状態図 29、図 30 に条件 (1) と条件 (2) の localhost への ping コマンドを示す。

```
i9270@falcon_pantz2<~>$ ping -c 5 localhost
PING localhost (127.0.0.1): 56 data bytes
64 bytes from 127.0.0.1: icmp_seq=0 ttl=64 time=0.012 ms
64 bytes from 127.0.0.1: icmp_seq=1 ttl=64 time=0.004 ms
64 bytes from 127.0.0.1: icmp_seq=2 ttl=64 time=0.007 ms
64 bytes from 127.0.0.1: icmp_seq=3 ttl=64 time=0.004 ms
64 bytes from 127.0.0.1: icmp_seq=4 ttl=64 time=0.006 ms

--- localhost ping statistics ---
5 packets transmitted, 5 packets received, 0.0% packet loss
round-trip min/avg/max/stddev = 0.004/0.007/0.012/0.003 ms
```

図 27: localhost への ping コマンド (条件 (1))



```
i9270@falcon_pantz2<~>$ ping -c 5 localhost
PING localhost (127.0.0.1): 56 data bytes
64 bytes from 127.0.0.1: icmp_seq=0 ttl=64 time=0.012 ms
64 bytes from 127.0.0.1: icmp_seq=1 ttl=64 time=0.004 ms
64 bytes from 127.0.0.1: icmp_seq=2 ttl=64 time=0.007 ms
64 bytes from 127.0.0.1: icmp_seq=3 ttl=64 time=0.004 ms
64 bytes from 127.0.0.1: icmp_seq=4 ttl=64 time=0.006 ms

--- localhost ping statistics ---
5 packets transmitted, 5 packets received, 0.0% packet loss
round-trip min/avg/max/stddev = 0.004/0.007/0.012/0.003 ms
```

図 28: localhost への ping コマンド (条件 (2))

図 29、図 30 から、localhost への ping は LAN ケーブルをつないでいてもつないでいなくても速度は変わらないことが分かった。

### 3.1.4 課題 3-4

ログインホストからファイルサーバー (phoenix) に対して、以下に示す 3 通りの条件下における ping コマンドの実行結果を示す。条件 (1) LAN ケーブルを接続していない状態条件 (2) LAN ケーブルを接続している状態条件 (3) LAN ケーブルを接続し、ネットワークインターフェイス x10 を無効化している状態図 31、図 32、図 33 に条件 (1) と条件 (2)、条件 (3) の phoenix への ping コマンドを示す。

```
i9270@falcon_pantz2<~>$ ping -c 5 phoenix
PING phoenix (10.255.255.252): 56 data bytes

--- phoenix ping statistics ---
5 packets transmitted, 0 packets received, 100.0% packet loss
```

図 29: phoenix への ping コマンド (条件 (1))

```
i9270@falcon_pantz2<~>$ ping -c 5 phoenix
PING phoenix.exp.info.nara-k.ac.jp (10.255.255.252): 56 data bytes
64 bytes from 10.255.255.252: icmp_seq=0 ttl=64 time=0.282 ms
64 bytes from 10.255.255.252: icmp_seq=1 ttl=64 time=0.411 ms
64 bytes from 10.255.255.252: icmp_seq=2 ttl=64 time=0.288 ms
64 bytes from 10.255.255.252: icmp_seq=3 ttl=64 time=0.353 ms
64 bytes from 10.255.255.252: icmp_seq=4 ttl=64 time=0.400 ms

--- phoenix.exp.info.nara-k.ac.jp ping statistics ---
5 packets transmitted, 5 packets received, 0.0% packet loss
round-trip min/avg/max/stddev = 0.282/0.3468/0.411/0.060 ms
```

図 30: phoenix への ping コマンド (条件 (2))

```
i9270@falcon_pantz2<~>$ ping -c 5 phoenix
PING phoenix.exp.info.nara-k.ac.jp (10.255.255.252): 56 data bytes
ping: sendto:Network is down
ping: sendto:Network is down
ping: sendto:Network is down
ping: sendto:Network is down
ping: sendto:Network is down

--- phoenix.exp.info.nara-k.ac.jp ping statistics ---
5 packets transmitted, 0 packets received, 100.0% packet loss
```

図 31: phoenix への ping コマンド (条件 (3))

図 31, 図 33 より、LAN ケーブルにつながっていないときとネットワークインターフェイスを無効化しているときは phoenix にはつながることはできなかった。図 32 の LAN ケーブルが繋がっているときのみ phoenix につながることができた。

### 3.1.5 課題 3-5

各ホストの MAC アドレスを調べ、表 2 に示す。

### 3.1.6 課題 3-6

実験メンバー全員の端末接続を確認後ログインホストからブロードキャストアドレスに対して ping コマンドの実行結果を図 34 に示す。

```
i9270@falcon_pantz2<~>$ ping -c 1 10.255.255.255
PING 10.255.255.255 (10.255.255.255): 56 data bytes
64 bytes from 10.255.255.255: icmp_seq=0 ttl=64 time=0.154 ms

--- 10.255.255.255 ping statistics ---
1 packets transmitted, 1 packets received, 0.0% packet loss
round-trip min/avg/max/stddev = 0.154/0.154/0.154/0.000 ms
```

図 32: ブロードキャストアドレスへの ping

ブロードキャストアドレスはサブネットマスクが上位 8bit でネットワークアドレスが 10.0.0.0 などで 10.255.255.255 となる。図 34 のようにコマンドを打つことによって結果が得られた。

### 3.1.7 課題 3-7

情報工学科の Web サーバー (www.info.nara-k.ac.jp) に対して、ping コマンドの実行結果を図 35 に示す。

```
i9270@falcon_pantz2<~>$ ping -c 1 www.info.nara-k.ac.jp
PING www.info.nara-k.ac.jp (202.24.246.1): 56 data bytes
64 bytes from 202.24.246.1: icmp_seq=0 ttl=63 time=1.733 ms

--- 10.255.255.255 ping statistics ---
1 packets transmitted, 1 packets received, 0.0% packet loss
round-trip min/avg/max/stddev = 1.733/1.733/1.733/0.000 ms
```

図 33: www.info.nara-k.ac.jp への ping

図 35 より、外部のネットワークであるため、帰ってくるまでの時間が 1.733ms と遅いことが分かった。

### 3.1.8 課題 3-8

ログインホストからファイルサーバーに対して、以下に示す 2 とおりの条件下における ping コマンドの実行結果を図 36, 37 に示す。条件 (1) ファイルサーバー (phoenix) の AC アドレスが ARP テーブルに登録されている状態条件 (2) ファイルサーバー (phoenix) の AC アドレスが ARP テーブルに登録されていない状態

```
i9270@falcon_pantz2<~>$ ping -c 1 phoenix
PING phoenix.exp.info.nara-k.ac.jp (10.255.255.252): 56 data bytes
64 bytes from 10.255.255.252: icmp_seq=0 ttl=64 time=0.282 ms

--- phoenix.exp.info.nara-k.ac.jp ping statistics ---
1 packets transmitted, 1 packets received, 0.0% packet loss
round-trip min/avg/max/stddev = 0.282/0.282/0.282/0.000 ms
```

図 34: phoenix への ping(条件 (1))

```
i9270@falcon_pantz2<~>$ ping -c 1 phoenix
PING phoenix.exp.info.nara-k.ac.jp (10.255.255.252): 56 data bytes
64 bytes from 10.255.255.252: icmp_seq=0 ttl=64 time=0.243 ms

--- phoenix.exp.info.nara-k.ac.jp ping statistics ---
1 packets transmitted, 1 packets received, 0.0% packet loss
round-trip min/avg/max/stddev = 0.243/0.243/0.243/0.000 ms
```

図 35: phoenix への ping(条件 (2))

図 36、図 37 より若干 ARP テーブルに登録されている状態のほうが速度が速いことが確認できた。

### 3.1.9 課題 3-9

ログインホストにおけるルーティングテーブルを図 38 に示す。

```
i9270@falcon_pantz2<~>$ netstat -r
Routing tables

Internet:
Destination      Gateway          Flags           Refs      Use     Netif Expire
default          c2611r          UGS             0         10      x10
10.0.0.0         link#1          U               0         178     x10
falcon_pantz2    link#1          UHS             0          5      lo0
localhost        link#3          UH              0          1      lo0

Internet6:
Destination      Gateway          Flags           Netif Expire
localhost.exp.info localhost.exp.info UH              lo0
fe80::%lo0       link#3          U               lo0
fe80::1%lo0      link#3          UHS             lo0
ff01::3::        fe80::1%lo0     U               lo0
ff02::%lo0       fe80::1%lo0     U               lo0
```

図 36: ルーティングテーブル

図 38 のように netstat -r というコマンドでルーティングテーブルを表示することができた。

### 3.1.10 課題 3-10

traceroute コマンドを使用して、ログインホストから以下に示すホストまでの通信経路を示す。ルーター (c2611r) ファイルサーバー (phoenix) 情報工学科の Web サーバー (www.info.nara-k.ac.jp) 図 39 に traceroute コマンドを使用した結果を示す。

```
i9270@falcon_pantz2<~>$ traceroute c2611r
traceroute to c2611r.exp.info.nara-k.ac.jp (10.0.0.1), 64 hops max, 40 byte packets
 1 c2611r (10.0.0.1)  2.034 ms *  1.202 ms

i9270@falcon_pantz2<~>$ traceroute phoenix
traceroute to phoenix.exp.info.nara-k.ac.jp (10.255.255.252), 64 hops max, 40 byte packets
 1 phoenix (10.255.255.252)  0.334 ms  0.302 ms  0.237 ms

i9270@falcon_pantz2<~>$ traceroute www.info.nara-k.ac.jp
traceroute to zeus.info.nara-k.ac.jp (202.24.246.1), 64 hops max, 40 byte packets
 1 c2611r (10.0.0.1)  1.903 ms  1.892 ms  1.001 ms
 2 zeus.info.nara-k.ac.jp (202.24.246.1)  2.454 ms  2.455ms  2.390 ms
```

図 37: traceroute コマンド

traceroute のコマンドの引数にホスト名を入れることでそれぞれのホストへの通信経路を表示することができた。

## 第3章 実験結果の考察および検討

## 第4章 参考文献および参考URL



## 第5章 実験における意見や感想

## 第6章 テキストの訂正