

オープンソフトウェア利用促進事業
第3回OSSモデルカリキュラム導入実証

**システム管理方法論
演習支援システム利用マニュアル**

平成22年12月
公立大学法人 公立はこだて未来大学

目次

1	概要	2
2	システム利用の流れ	2
3	個別課題のチェック	2
3.1	SSH 課題	2
3.1.1	準備	2
3.1.2	SSH 接続	2
3.1.3	コマンド実行	2
3.2	メール課題	3
3.2.1	準備	3
3.2.2	学生サーバへの接続	4
3.2.3	ローカルでのメール送受信	4
3.2.4	外部サーバとのメール送受信	4
3.2.5	POP・IMAP からのメール読み込み	4
3.3	Web 課題	5
3.3.1	準備	5
3.3.2	課題確認	5
3.3.3	個別の課題確認	5
3.4	WebDAV 課題	6
3.4.1	準備	6
3.4.2	学生サーバへの接続	6
3.4.3	WebDAV にアクセス可能かどうかのチェック	6
3.4.4	パスワードロックのチェック	6
3.4.5	パスワード設定のチェック	6
3.5	NFS 課題	7
3.5.1	準備	7
3.5.2	ファイル、ディレクトリ表示	7
3.5.3	ログファイルの表示	7
3.5.4	ファイル差分表示	8
3.6	rsync 課題	8
3.6.1	準備	8
3.6.2	ファイル、ディレクトリ表示	8
3.6.3	ログファイルの表示	8
3.6.4	ファイル差分表示	8
4	その他のチェック	8
4.1	Nmap によるポートチェック	8
4.1.1	準備	8
4.1.2	学生サーバのネットワークへの参加の確認	9
4.1.3	学生サーバのポートチェック	9
4.2	パスワードクラック	9
4.2.1	準備	9
4.2.2	学生サーバへの接続	9

1 概要

本マニュアルは、演習支援システムの利用方法について記述するものである。インストールマニュアルに沿って本演習支援システムを整備した後、本マニュアルに従って、各課題のチェックを行う。各課題の詳細については、講義資料と演習資料を参照のこと。

2 システム利用の流れ

各課題のチェックのためには、config ディレクトリ内の config.php に設定を記述する必要がある。詳細な記述方法については、各課題の準備の節において記述する。また、併せてインストールマニュアルを参照する。

各課題の設定後は、演習支援システム内の index.php に Web ブラウザからアクセスする。index.php には、事前に servers.csv 内に設定したクラス毎に、図 2.1 のような class.php へのリンクが貼られている。この class.php が、各学生サーバへのポータルページとなっており、このページから各課題チェックを実行する。

システム管理方法論2010							
sysadmin05	Alive	ssh	mail	web	webdav	ntp	更新
sysadmin07	Alive	ssh	mail	web	webdav	ntp	更新
sysadmin09	Alive	ssh	mail	web	webdav	ntp	更新
sysadmin13	Dead	ssh	mail	web	webdav	ntp	更新
sysadmin15	Alive	ssh	mail	web	webdav	ntp	更新
sysadmin17	Alive	ssh	mail	web	webdav	ntp	更新
sysadmin19	Alive	ssh	mail	web	webdav	ntp	更新

図 2.1: class.php の表示例

3 個別課題のチェック

3.1 SSH 課題

3.1.1 準備

インストールマニュアルを参照し、config ディレクトリ内の servers.csv 内に学生サーバの IP アドレス、ユーザー名、パスワードを記述する。

3.1.2 SSH 接続

ページを開いた際、自動的に servers.csv を読み込み、SSH 接続モジュール (lib/ssh_lib.php) を用いて学生サーバに SSH 接続を行う。この際、servers.csv に記述されたユーザー名、パスワードを用いてログインをする事が可能であれば、ページに緑の字で「成功」と表示される。ログインができない場合は赤い字で大きく「失敗」と表示される。成功と表示されている場合は、黒線の枠とコマンド入力用のテキストフィールドが表示され、枠内に「コマンドを入力し「実行」ボタンを押してください…」と表示される (図 3.1)。また、失敗と表示されている場合は、枠が表示されない。

3.1.3 コマンド実行

2で成功と表示され、コマンド入力用のテキストフィールドが表示されている場合のみ使用できる。

テキストフィールドに1行でコマンドを入力し実行ボタン、または Enter キーを押すと、入力したコマンドを

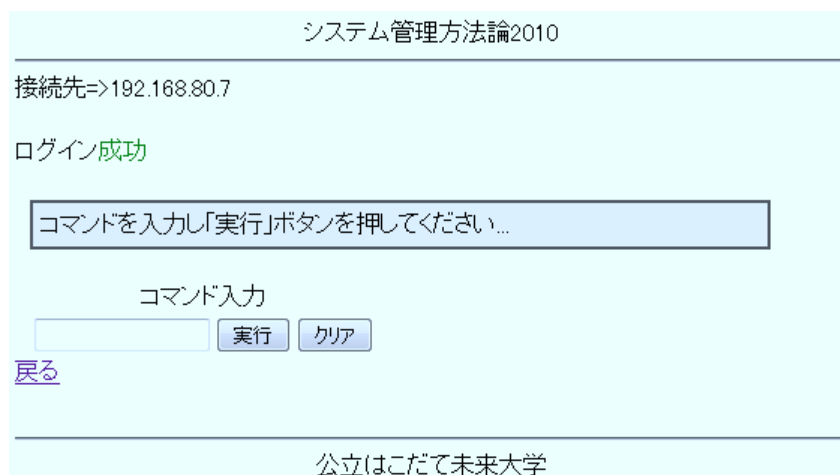


図 3.1: ログイン成功時

ログイン先で実行する。またその際、テキストフィールドに記述した内容は消去される。クリアボタンを押すと、テキストフィールドに記述した内容をすべて消去する。コマンド実行から表示までの間は、黒枠内に「wait…」と表示される。コマンドの実行結果は、黒枠内に表示される(図 3.2, 図 3.3)。ただし、対話式のコマンドは使用することができない。コマンドは連続して実行することができるが、黒枠に記述されるコマンド実行結果は、コマンド実行を行う度に書き換えられる。

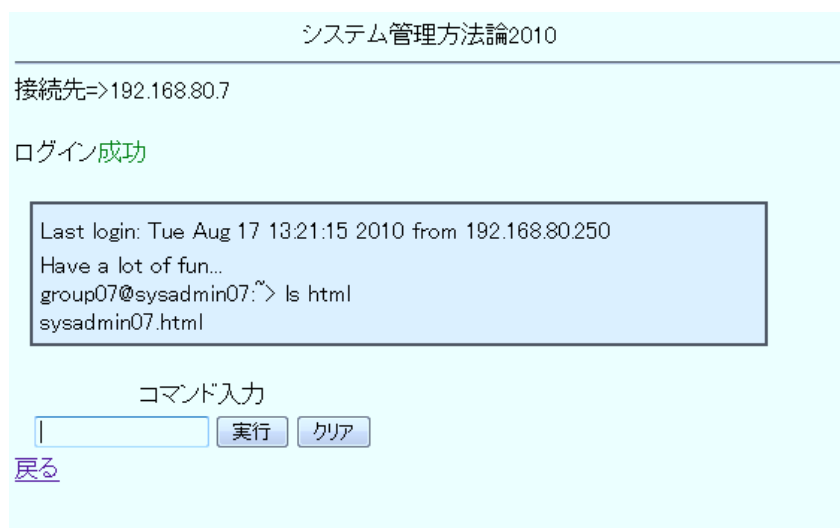


図 3.2: コマンド実行例 1

3.2 メール課題

3.2.1 準備

インストールマニュアルを参照し、config ディレクトリの servers.csv 内に学生サーバの ip アドレス、ユーザ名、パスワードを記述する。また、config.php 内の \$responderAddress に、自動応答用メールアドレスを設定する。(例: \$responderAddress="responder@sysadmin2010.per.c.fun.ac.jp";)

接続先=>192.168.80.7

ログイン成功

```
Last login: Tue Aug 17 13:21:18 2010 from 192.168.80.250
Have a lot of fun...
group07@sysadmin07:~> cat /etc/resolv.conf
### /etc/resolv.conf file autogenerated by netconfig
#
# Before you change this file manually, consider to define the
# static DNS configuration using the following variables in the
# /etc/sysconfig/network/config file:
# NETCONFIG_DNS_STATIC_SEARCHLIST
# NETCONFIG_DNS_STATIC_SERVERS
# NETCONFIG_DNS_FORWARDER
# or disable DNS configuration updates via netconfig by setting:
# NETCONFIG_DNS_POLICY=""
#
# See also the netconfig(8) manual page and other documentation.
#
# Note: Manual change of this file disables netconfig too, but
# may get lost when this file contains comments or empty lines
# only, the netconfig settings are same with settings in this
# file and in case of a "netconfig update -f" call.
```

図 3.3: コマンド実行例 2

3.2.2 学生サーバへの接続

設定した servers.csv を読み込み、ssh 接続モジュール (lib/ssh_lib.php) を用いて、学生サーバに接続を行う。接続失敗した際は、その旨が表示される。成功した場合は、次の段階に移る。図 3.4 のような、課題チェック画面が表示される。

3.2.3 ローカルでのメール送受信

mail コマンドを用いて、自身のユーザ名宛にメールを送信する。この際、成功の表示がでていれば、メールの送信が成功している。次に、環境変数 \$MAIL 内をチェックし、送信したメールが受信できていることを確認する。成功の表示がでていれば、メールの受信が成功している。

3.2.4 外部サーバとのメール送受信

config.php ファイル内に記述した自動応答アドレス宛にメールを送信する。成功の表示が出ていれば、外部サーバへのメール送信が成功している。次に、\$MAIL 内に自動応答アドレスからの応答メールがあることを確認する。成功の表示が出ていれば、外部サーバからのメール受信が成功している。

3.2.5 POP・IMAP からのメール読み込み

fetchmail を使い、POP3 と IMAP からメールの読み出しを行う。読み出し不可が表示された場合は、ポートが開いていない等の原因が考えられる。読み出し可が表示された場合は、ブラウザ上から読み出したメールの確認ができる。

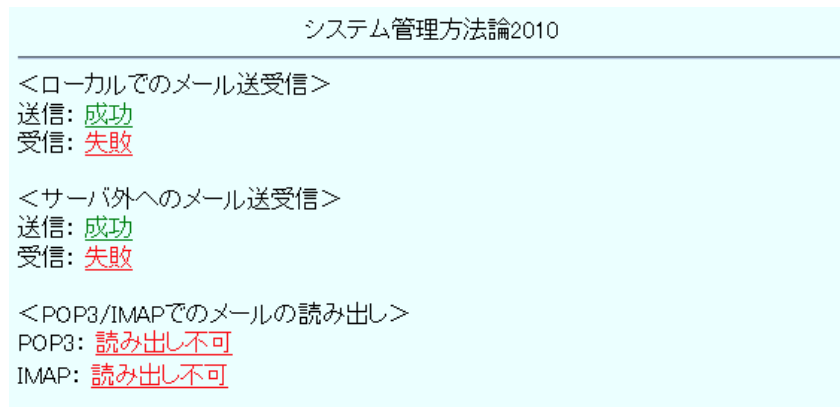


図 3.4: メール課題のチェック画面

3.3 Web 課題

3.3.1 準備

インストールマニュアルを参照し、config ディレクトリの servers.csv 内に学生サーバの ip アドレス、ユーザ名、パスワードを記述する。

3.3.2 課題確認

画面の上部には、下記の順番に学生サーバの内容が表示される。

<http://アドレス>

<http://アドレス/dav>

3.3.3 個別の課題確認

ssh ログインが成功した場合は、図 3.5 のように、上記の学生サーバのアドレスの下に各ユーザのアドレスが表示される。

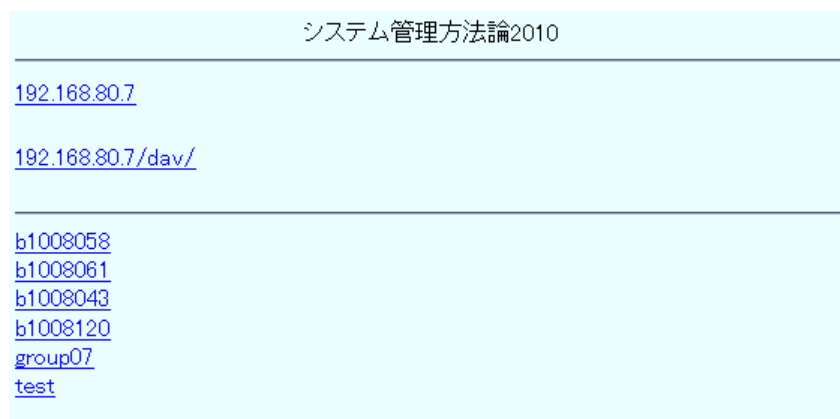


図 3.5: Web 課題のチェック画面

3.4 WebDAV 課題

3.4.1 準備

インストールマニュアルを参照し、config ディレクトリの servers.php 内に学生サーバの ip アドレス、ユーザ名、パスワードを記述する。また、config.php 内の \$WebDAVAddress に WebDAV のディレクトリパスを設定する。さらに、WebDAV チェックのために cadaver をインストールしておく。

(例: \$webdavAddress="/dav";)

3.4.2 学生サーバへの接続

servers.csv を読み込み、ssh 接続モジュール (lib/ssh_lib.php) を用い、学生サーバに接続を行う。接続失敗した際は、その旨が表示される。成功した場合は、図 3.6 のように表示され、次の段階に移る。

```
システム管理方法論2010
```

```
対象-> http://192.168.80.7/dav
apache : 起動中
```

図 3.6: 学生サーバへの接続

3.4.3 WebDAV にアクセス可能かどうかのチェック

cadaver を用いて指定した URL にアクセスできるかどうかチェックする。アクセスできなければ WebDAV が有効になっていないと判断しその旨を表示する。アクセスできれば図 3.7 のように表示され、次の段階に移る。

```
Array
(
    [0] => Authentication required for Please entry your ID and password on server `192.168.80.7':
    [1] => Username:
Authentication aborted!
    [2] => Could not open collection:
    [3] => 401 Authorization Required
    [4] => dav:/dav/?
    [5] =>
)

WebDAVアクセス:OK
```

図 3.7: WebDAV へのアクセス

3.4.4 パスワードロックのチェック

cadaver を用いて指定した WebDAV にパスワードロックがかかっているかどうかチェックする。WebDAV にそのままアクセスできたらパスワードロックがかかっていると判断し、その旨を表示する。パスワードを求められたらロックがかかっていると判断し図 3.8 のように表示され、次の段階に移る。

3.4.5 パスワード設定のチェック

servers.csv に記述されたユーザ名とパスワードが WebDAV に正しく設定されているかどうかチェックする。正しいユーザ名とパスワードでアクセスが出来れば図 3.9 のように表示する。アクセスできなければ、その旨を表示する。

```

Array
(
    [0] => Authentication required for Please entry your ID and password on server `192.168.80.7':
    [1] => Username:
Authentication aborted!
    [2] => Could not open collection:
    [3] => 401 Authorization Required
    [4] => dav:/dav/?
    [5] =>
)
パスワードロック:ON

```

図 3.8: WebDAV のパスワードロックの確認

```

Array
(
    [0] => Authentication required for Please entry your ID and password on server `192.168.80.7':
    [1] => Username: dav:/dav/>
    [2] => Connection to `192.168.80.7' closed.
    [3] =>
)
パスワード設定:OK

```

公立はこだて未来大学

図 3.9: WebDAV のパスワード設定の確認

3.5 NFS 課題

3.5.1 準備

config ディレクトリ内の config.php に記述されている \$nfsTargetFolder 変数に、課題において NFS マウント先として指定する、各グループのディレクトリが存在するディレクトリのパスを指定する。\$nfsLogFile 変数に、NFS のログが記載されるログファイルのパスをそれぞれ指定する。

(例: \$nfsTargetFolder="/exports"; \$nfsLogFile="/var/log/messages";)

3.5.2 ファイル、ディレクトリ表示

課題において、NFS マウント先として指定されたディレクトリ (例: /exports/group01) 内のファイル、サブディレクトリを一覧表示する。ファイルの場合、ファイル名、ファイルサイズ (単位はバイト)、更新日時が 1 行に表示される。サブディレクトリの場合、サブディレクトリ名、[dir]、サブディレクトリ更新日時が 1 行に表示される。ファイル名、サブディレクトリ名はそれぞれハイパーリンクになっている。ファイル名をクリックすると、そのファイルの内容を表示する。ファイルがアーカイブファイル (拡張子が .bz2 のファイル) の場合は、アーカイブファイルに含まれているファイル、ディレクトリを一覧表示する。サブディレクトリ名をクリックすると、そのサブディレクトリ内のファイル一覧を表示する。

3.5.3 ログファイルの表示

\$nfsLogFile で指定したログファイルから、NFS に関連するログのみを抽出し表示する。表示場所は、2 の表示部の下である。

3.5.4 ファイル差分表示

2で表示した一覧の中に、拡張子がそれぞれ「.old」と「.new」である同じ名前のファイルがあった場合(例: backup.old と backup.new), 拡張子が「.new」であるファイル名の横に「diff」というリンクが表示される。これをクリックすると diff コマンドを実行し、先の2つのファイルの差分を表示する。

3.6 rsync 課題

3.6.1 準備

config ディレクトリ内の config.php に記述されている \$rsyncTargetFolder 変数に、課題においてマウント先として指定する、各グループのディレクトリが存在するディレクトリのパスを指定する。\$rsyncLogFile 変数に、rsync のログが記載されるログファイルのパスをそれぞれ指定する。

(例: \$rsyncTargetFolder="/exports"; \$rsyncLogFile="/var/log/rsyncd.log";)

3.6.2 ファイル, ディレクトリ表示

課題において、マウント先として指定されたディレクトリ(例: /exports/group01)内のファイル, サブディレクトリを一覧表示する。ファイル名, サブディレクトリ名はそれぞれハイパーリンクになっている。ファイル名をクリックすると、そのファイルの内容を表示する。ファイルがアーカイブファイル(拡張子が.bz2のファイル)の場合は、アーカイブファイルに含まれているファイル, ディレクトリを一覧表示する。サブディレクトリ名をクリックすると、そのサブディレクトリ内のファイル一覧を表示する。

3.6.3 ログファイルの表示

\$rsyncLogFile で指定したログファイルから、rsyncに関連するログのみを抽出し表示する。表示場所は、2の表示部の下である。

3.6.4 ファイル差分表示

2で表示した一覧の中に、拡張子がそれぞれ「.old」と「.new」である同じ名前のファイルがあった場合(例: backup.old と backup.new), 拡張子が「.new」であるファイル名の横に「diff」というリンクが表示される。これをクリックすると diff コマンドを実行し、先の2つのファイルの差分を表示する。

4 その他のチェック

ここからは、課題チェックとは別に、演習を円滑に進める目的で学生サーバに対して行うチェックである。

4.1 Nmap によるポートチェック

4.1.1 準備

インストールマニュアルを参照し、config ディレクトリの servers.csv 内に学生サーバのクラス, グループ名, ip アドレス, ユーザ名, パスワードを記述する。また、config.php 内の \$targetProtocols に、チェックするサービスのポート番号, リンク名, リンク先スクリプト名を記述する。

(例: \$targetProtocols=array(
array("port"=>"22", "linkName"=>"ssh", "script"=>"ssh.php"),
array("port"=>"22,25,110,143", "linkName"=>"mail", "script"=>"mail.php"),
array("port"=>"80,443", "linkName"=>"web", "script"=>"web.php"),
array("port"=>"80,443", "linkName"=>"webdav", "script"=>"webdav.php"),

```
array("port"=>"123", "linkName"=>"ntp", "script"=>"ntp.php"),
array("port"=>"none", "linkName"=>"nfs", "script"=>"nfs.php"),
array("port"=>"none", "linkName"=>"rsync", "script"=>"rsync.php")
);)
```

4.1.2 学生サーバのネットワークへの参加の確認

index.php より各クラスへのリンクへアクセスすることによって、config ディレクトリ内の servers.csv に記述されたクラスごとに学生サーバへネットワークへの参加確認を行う。ネットワークへの参加が確認できた場合は Alive が、出来なかった場合は Dead が表示される。

4.1.3 学生サーバのポートチェック

index.php より各クラスへのリンクにアクセスすることによって、config ディレクトリ内の config.php に記述されたポートが開いているかのチェックを行う。記述されたポートが開いている場合は詳細動作をチェックするためのスクリプトへのリンクが表示される。開いていない場合はリンクが表示されない。

4.2 パスワードクラック

4.2.1 準備

インストールマニュアルを参照し、config ディレクトリの servers.csv 内に学生サーバの ip アドレス、ユーザ名、パスワードを記述する。また、パスワードの候補となるキーワードを、passwd.txt 内へ記述する。

4.2.2 学生サーバへの接続

server.csv を読み込み、ssh 接続モジュールを用いて、学生サーバへ passwd.txt に記述されたキーワードをパスワードとして接続を試みる。接続が成功した場合は、下記のように表示される。

```
ppCannon!! -> ユーザー パスワード
```

この著作物は、「クリエイティブ・コモンズ・ライセンス 表示 2.1 日本」により、公立はこだて未来大学から利用許諾されています。
詳しい利用許諾条項は、<http://creativecommons.org/licenses/by/2.1/jp/legalcode> を御覧ください。