SURE 静岡大学学術リポジトリ
Shizuoka University REpository

工学部webサーバのためのミラーサイトの立ち上げとそのバックアップファイルの作成技術の習得

メタデータ	言語: Japanese 出版者: 静岡大学技術部 公開日: 2008-12-25 キーワード (Ja): キーワード (En): 作成者: 太田, 諭之 メールアドレス: 所属:
URL	https://doi.org/10.14945/00002920

工学部webサーバのためのミラーサイトの立ち上げとそのバックアップファイルの作成技術の習得

工学部 技術部 情報技術支援室　太田諭之

１．はじめに

現在、静岡大学工学部のホームページは一基のサーバマシンで稼動している。しかし、本家のサーバが停止してしまった場合ホームページを見ることが出来ない。このような場合に備えてミラーサーバを設置しミラーサイトを開設した。

２．ミラーリングとは

Webページの内容をそっくりそのまま他のサーバにコピーして閲覧可能にする技術である。Webページのクローンのようなもので、複製されたページをミラーサイトという。この、ミラーサイトはネットワークのトラブル等により本家のページが見られなくなった場合やハードディスクのクラッシュによりデータが破壊されたトラブルに備えて常に見られるようにしておくものである。通常、オリジナルのサーバとミラーサーバは別の場所に設置されていることが多い。

３．使用するミラーサーバ

ミラーサーバの機種の選定に当たっては価格が比較的抑えられたもの、最低限のメモリとプロセッサ、 RAID機能を備えている機種を重点的に選んだ。その結果、デル株式会社の PowerEdge SC420 を使用することにした。 OS は工学部サーバの OS と同じ Vine Linux3.2 をインストールした。ハードディスクのクラッシュによるデータ損失に備えるため RAID ０構成とした。

図１． DELL社製　PowerEdge SC440 [1]

４．研修内容

4-1．ねらい

ミラーリングは市販されているソフトを利用しても出来るが、Linuxで一般的に使用されているコマンド wget を用いて対象ページのリンク先アドレスの変換の設定を行いhttpファイルの取得を行う。そしてこの圧縮ファイルの保存を行い、作業が完了した旨をメールで配信する一連の流れを定期的に実行するスクリプトを作成し、ミラーリングと同時にバックアップも実行できるようにする。

1　http://www.dell.com/downloads/jp/products/pedge/poweredge_sc420.pdf

4-2． PHP のインストール

PHP とは、Hypertext Preprocessor の略でオープンソースの汎用スクリプト言語である。特にサーバサイドで動作するWebアプリケーションの開発に適している。工学部のホームページはこの PHP で作成されているため、上記のミラーサーバに PHP をインストールをする。

```
#apt-get update
#apt-get install php php-mysql
```

を管理者権限で実行し、

```
#vi /home/httpd/html/test.php
```

test.php をドキュメントルート上に置き、以下を打ち込み保存。

```
<?PHP
phpinfo();
 ?>
```

作成が完了したらファイルを保存。ドキュメントルートにアクセスして PHP 本体の情報、バージョン、拡張モジュール等が画面に表示されたら PHP のインストールは成功している。

4-3． wget コマンドを使用する。

Wget とは非対話式のネットワークのダウンロードコマンドで、HTTP、 HTTPS 、FTPをサポートしている。コマンドプロンプト画面から、工学部のwebサーバに接続してファイルをダウンロードする。使用法は、

wget *[option].. [URL]..*

のようにコマンドの後にオプションと URL を記述する。実際に、工学部のホームページサーバよりホームページをダウンロードした。

```
#cd /home/httpd/html -ダウンロード先へディレクトリ移動
#wget -l 5 -r -k -nH www.eng.shizuoka.ac.jp
```

各オプションの説明

-l　(--level) depth ：リンクをたどる階層を指定する。（指定しないと５階層となる）

-k　(--convert-links) ：リンク先をローカルでも参照できる様にする。（例：
www.eng.shizuoka.ac.jp → *自ホスト名* .eng.shizuoka.ac.jp に変換される）

-nH　(--no-host-directories)：ローカル保存時、ホスト名を外す。

-r　(recursive)：再帰的にファイルを取得する。

4-4． 定期的なバックアップのために

定期的にサーバ側で工学部のページをコピー、ミラーサイトへ反映させるためにスクリプトを作成した。

```
#!/bin/sh  -シェルスクリプト
cd "/backup/wget"  -カレントディレクトリはwget専用のフォルダとする
date=`date +%Y-%m-%d`  -今日の日付を返す
mkdir ${date}  -バックアップした日付のディレクトリを作成する
cd "/backup/wget/${date}/"  -バックアップした日付のディレクトリに移動する
wget -l 10 -k -r -nH www.eng.shizuoka.ac.jp -o "${date}"  -wgetを実行

                    ～略～

tar czvf ${date}.tar.gz /backup/wget/archive/   -ファイルを圧縮する
(ls -l /backup/wget;df -a) |mail -s "Kougakubu Backup Compleated!" info-tech
@gijutsubu.eng.shizuoka.ac.jp  -バックアップが完了したらPostfixで管理者にメール送信
```

図２．バックアップのためのスクリプトの一部

スクリプト（図２）は、 wget が実行された日付のファイル名を所定のフォルダ（ /backup/wget/ ）に作成し静岡大学工学部のホームページを 10 階層分たどり取得されたファイルを保存する。そして、外部からもページを見られるようににDocument Root（home/httpd/html/）に wget のファイルをコピーする。
最後に、圧縮ファイルを作成し一連の流れをメールにて管理者に送信をする。
工学部ホームページのトップで用いられているフラッシュ（.swf）や一部の画像ファイル（ .gif ）は個別にコピーしている。

```
[root@techstuff20 wget]# ls
2007-10-26/  2007-11-11/  2007-11-23/  2007-12-07/  Original/
2007-10-31/  2007-11-14/  2007-11-25/  2007-12-09/  archive/
2007-11-02/  2007-11-16/  2007-11-28/  2007-12-12/  backup.sh*
2007-11-04/  2007-11-18/  2007-11-30/  2007-12-14/  backup.sh~*
2007-11-07/  2007-11-20/  2007-12-02/  2007-12-16/  www/
2007-11-09/  2007-11-21/  2007-12-05/  2007-12-18/
[root@techstuff20 wget]# cd archive/
[root@techstuff20 archive]# ls
2007-11-20.tar.gz  2007-11-21.tar.gz  2007-12-18.tar.gz
[root@techstuff20 archive]#
```

図３．時系列に並んだバックアップファイル

図３は、ミラーサーバに時系列ごとに保存されたバックアップファイルを一覧表示させたものである。年度・月・日ごとにファイルが保存されているのが確認できる。日付の後に tar.gz とあるのは圧縮されたファイルである。 backup.sh はスクリプトファイルである。

次にスクリプトを定期的に同時刻に実行するためにLinuxには標準でcronというデーモンが用意されている。設定ファイル /etc/crontab でスクリプトを実行する曜日、時間を入力する。

```
[root@techstuff20 root]# less /etc/cron
cron.d        cron.hourly   cron.weekly   crontab~
cron.daily    cron.monthly  crontab
```

図４． /etc/cron/ ファイルの一覧

図４は/etc/cron内のファイルを一覧表示したものである。それぞれの定時刻に処理できるファイルが並んでいる。それぞれの詳細は以下の通りである。

/etc/cron.monthly　　毎月
　　/corn.weekly　毎週
　　/cron.daily　　毎日
　　/cron.hourly　　毎時

```
SHELL=/bin/bash
PATH=/sbin:/bin:/usr/sbin:/usr/bin
MAILTO=root
HOME=/

# run-parts
01 * * * * root run-parts /etc/cron.hourly
02 4 * * * root run-parts /etc/cron.daily
05 3 * * 0,3,5 root run-parts /etc/cron.weekly
42 4 1 * * root run-parts /etc/cron.monthly
```

図５．　/etc/crontab の設定

図５は図４を実行している設定ファイル /etc/crontab である。
バックアップは一週間単位で実行するので、 /etc/cron.weekly でシェルスクリプト backup.sh を実行するように設定すれば良い。図５の“05 3 * * 0,3,5”の意味は日、水、金曜日の午前3:05にroot権限で cron.weekly を実行せよ（一週間に三回実行する）という意味である。3:00ちょうどに設定しないのは他のサービスと被らない為で図５では3:05に設定した。

4-5．バックアップ及びミラーリング完了を伝えるメール送信

Kougakubu Backup Compleated!　　backup@techstuff20.eng.shizuoka.ac.jp

件名: Kougakubu Backup Compleated!
差出人: backup@techstuff20.eng.shizuoka.ac.jp
送信日時: 17:34
宛先: info-tech@gijutsubu.eng.shizuoka.ac.jp

```
drw-r--r--   31 backup   root        4096 Dec  7 03:04 2007-12-07
drw-r--r--   31 backup   root        4096 Dec  9 03:04 2007-12-09
drw-r--r--   31 backup   root        4096 Dec 12 03:04 2007-12-12
drw-r--r--   31 backup   root        4096 Dec 14 03:04 2007-12-14
drw-r--r--   31 backup   root        4096 Dec 16 03:04 2007-12-16
drw-r--r--   31 backup   root        4096 Dec 18 13:21 2007-12-18
drwxr-xr-x   31 backup   root        4096 Oct 31 11:18 Original
drwxr-xr-x    2 root     root        4096 Dec 18 13:22 archive
-rwxr-xr-x    1 backup   root         579 Dec 18 13:14 (backup.sh)
-rwxr-xr-x    1 backup   root         580 Dec 18 11:44 backup.sh~
drwxr-xr-x   75 root     root        8192 Nov 20 08:40 www
合計 414848
-rw-r--r--    1 backup   root   212185192 Nov 20 16:35 2007-11-20.tar.gz
-rw-r--r--    1 backup   root   212183463 Nov 21 10:12 2007-11-21.tar.gz
-rw-r--r--    1 backup   root         139 Dec 18 17:34 2007-12-18.tar.gz
ファイルシステム     1k-ブロック    使用中      空き 使用% マウント場所
/dev/hdb2            77735460  10074504  63712180  14% /
none                        0         0         0   -  /proc
none                        0         0         0   -  /proc/bus/usb
/dev/hdb1                46633      6702     37523  16% /boot
none                        0         0         0   -  /dev/pts
none                   252460         0    252460   0% /dev/shm
```

時系列ごとのバックアップファイル
スクリプトファイル
圧縮ファイル
ディスクのパーティション

図６．　wget が完了した旨を伝えるメール

図 6 は wget が完了し管理者に送信されたメールである。時系列ごとのバックアップファイル、圧縮されたファイル、ディスクのパーティションが一目で確認できる。管理者は常に wget でファイルが取得されているか？ディスクの容量は正常か？圧縮ファイルが作成されているか？を日々の業務でチェックをする。又ファイルの容量（使用％）が80%を超えたらファイルを別のメディアに保存をするなどして容量が圧迫されるのを防ぐ。

図７． wget によりミラーサイトを表示させる

図 7 は工学部のホームページとミラーサイトを並べたものである。ページの中身は全く変わらないがアドレスが異なる事に注目頂きたい。左が工学部のホームページ、右がミラーページである。前述のスクリプトで左のページに変更が加えられると右のページに即時に変更箇所が反映出来る様にした。

５．まとめ

今回の研修では、実際に参加者が wget コマンドを用いて工学部の工作センターのホームページ（ http://mtc17.eng.shizuoka.ac.jp/ （現在は新ホストに移行）） をそのままダウンロードしてバックアップをし、自分のメールアドレスへメール送信をした。私自身も今まで定期的に手動でWebページのコンテンツファイルのコピーを取って圧縮し、Document Rootへの保存をしていたのだが、今回の研修でこの作業をサーバで自動化させたため大幅な手間を省くことができた。

研修参加者の意見として、「 wget というコマンドを今まで知っていたがこのような利用ははじめて」や「インターネット上のページは wget で全てのページ情報を取得できてしまうため、セキュリティ面で心配」があった。

今後は、フラッシュがダウンロードできないこと、一部の gif ファイルがダウンロードできないこと、第三者による wget の使用の規制方法 [2] を課題として取り組んでいきたい。

・研修参加者
磯谷章（実験教育支援室）
大橋和義（情報技術支援室）
高木廣伸（学科系技術支援室）
水野保則（情報技術支援室）

[2] wget の規制方法は Basic 認証があるが、ユーザ名とパスワードでアクセス許可を与える為、パスワードを知らない不特定ユーザのアクセスを規制してしまう欠点がある。