Backup gem

Jonathan Groll

Installation

```
# gem install backupgem

Bulk updating Gem source index for: http://gems.rubyforge.org

Install required dependency runt? [Yn]  Y

Install required dependency net-ssh? [Yn]  Y

Install required dependency needle? [Yn]  Y

Install required dependency madeleine? [Yn]  Y
```

or can be checked out from rubyforge:

```
svn checkout svn+ssh://rubyforge.org/var/svn/backupgem
```

A Typical Backup

A typical backup has the following sequence:

- content
- compress
- encrypt
- deliver
- rotate
- cleanup

Backup Recipe File Format

Recipes Explained

- The Backup Recipe format is pure ruby code. Anything that is valid ruby is valid in the recipe file. There are, however, a number of shortcuts that will make your life easier.

- Each of the steps are specified as an action. (An action is really nothing more than a method that becomes defined in the Actor instance.)

- You may create “hook” actions for any of the actions. So if you define a method before_content it will be called immediately before #content is called.

- In each action, the variable last_result is set. This is the return value of the previously called method in the chain. Note that this includes the output of the “hook” methods.

- All configuration variables are available to actions via the hash c[]. For example, the backup path is available to your actions as c[:backup_path].

.

Variables

Name	Desc.	Example
:backup_path	The path to place backup archives.	set :backup_path, "/var/local/backups/mediawiki"
:tmp_dir	Specify a directory that backup can use as a temporary directory. Default /tmp.	set :tmp_dir, File.dirname(__FILE__) + "/../tmp"

must have write permissions for each of these directories.

Content

The content that is to be backed up. Backup provides a couple of shortcuts for common ways to locate content and allows you to arbitrarily define your own. Some typical types of content are:

- a particular file
- a particular folder
- the contents of a particular folder

specified like so:

```
action :content, :is_file   => "/path/to/file"
action :content, :is_folder => "/path/to/folder"
action :content, :is_contents_of => "/path/to/other/folder"
(^ content is folder/* , recursive option)
```

:content as a series of shell commands just pass #action a block:

```
action(:content) do
  sh "echo \"hello $HOSTNAME\""
  sh "mysqldump -uroot database > /path/to/db.sql"
  "/path/to/db.sql"
end
```

Compress

Again, there are a few one-liners for common cases and you can create your own.

```
action :compress, :method => :tar_bz2
action :compress, :method => :tar_gzip
```

Again, you can create your own.

```
action(:compress)
  sh "my_tar  #{last_result} #{last_result}.tar"
  sh "my_bzip #{last_result}.tar #{last_result}.tar.bz2"
  last_result + ".tar.bz2"
end
```

Encrypt

Encryption is available to you, if you wish to use it.

```
  set :encrypt, true                              # default is +false+
  set :gpg_encrypt_options, "--default-recipient" # default is "
string
  action :encrypt, :method => :gpg # default, none
```

or your own:

```
  action(:encrypt)
    sh "gpg #{c[:gpg_encrypt_options]} --encrypt #{last_result}"
    last_result + ".gpg"
  end
```

Delivery

Action

Delivery is supported via scp, ftp, and mv

Full FTP support is unfinished.

Currently, only scp and mv fully support rotation

```
  action :deliver, :method => :scp
  action :deliver, :method => :ftp
  action :deliver, :method => :mv
```

Variables - :servers (ONLY 1) :ssh_user :identity_key

Rotate

Rotation of your backups is a way to keep snapshot copies of your backups in time while not keeping every single backup for every single day.

Backup supports grandfather-father-son.

```
  set :rotation_method,  :gfs (default)
```

You can specify when a son is promoted to a father by the following variable:

```
  set :son_promoted_on,    :fri
```

You specify when fathers are promoted to grandfathers by something like the following

```
  set :father_promoted_on, :last_fri_of_the_month
```

Valid arguments for specifying these promotions are as follows:

- :mon-:sun – A symbol of the abbreviation of any day of the week
- :last_[mon-sun]_of_the_month – A symbol, replacing [mon-sun] with the abbreviation for the day of the weeks. Example: :last_thu_of_the_month.
- ***Any valid Runt object***.

Example

Known Bugs /Limitations

- Unfinished FTP support
- Add support for multiple backup servers
- Add in better logging
- Remove all the tmp_dir references. use and test the standard library tmpdir