

Expectをつかったサーバ管理レシピ

長野雅広

Id:kazeburo

自己紹介

- 某SNSサイトで運用担当Perlプログラマをやっています
- Blog: <http://blog.nomadscafe.jp/>
- 花粉症

今日の話題

- **十数台のサーバのセットアップをどう行うか**
 - 同じコマンドをsshでサーバに発行していく
 - 初期状態のサーバにも使いたい

コマンド実行を自動化したい

- **手動だと間違える**
 - Config間違えてソースコード流出とか
- **単純に面倒**

自動化しにくい部分

- rootでのパスワードログイン
- sudoでパスワード入力
- ホスト公開鍵のプロンプト

それExpectでできるよ

Expectとは

- “Expect は、スクリプトの指示に従って、対話的なプログラムと“会話”するプログラム”
- 参考URL
 - <http://www.linux.or.jp/JM/html/expect/man1/expect.1.html>
 - <http://www-06.ibm.com/jp/developerworks/linux/020614/jl-sc1.html>

自作ツール”mssh”を使っています

- Expectやpasswordの処理をひとまとめにした短いPerlスクリプト

```
> mssh -f server.list -u boofy -p -- sudo sh -c "cd /tmp && id -u && uptime"
boofy password: *****
[boofy@server1] CMD: sudo sh -c 'cd /tmp && id -u && uptime'
Password:
0
00:26:46 up 27 days, 13:38, 0 users, load average: 0.00, 0.00, 0.00

[boofy@server2] CMD: sudo sh -c 'cd /tmp && id -u && uptime'
Password:
0
00:26:46 up 27 days, 9:26, 0 users, load average: 0.00, 0.00, 0.00
```

msshのレシピ

- CPANモジュール
 - IO::Prompt
 - String::ShellQuote
 - Expect.pm
- 自分で書いたコードは非常に少ない

IO::Prompt

- プロンプトでパスワードの入力してスクリプト終了まで使いまわす
- 入力したパスワードが画面にでないように*に置き換え

```
use IO::Prompt;  
my $passwd = prompt("$user password: ", -echo=>"*");
```

String::ShellQuote

- shellに適したquoteを行う

```
> mssh -- sudo sh -c "cd /tmp && pwd"  
XXX ¥@ARGV;  
---  
- sudo  
- sh  
- -c  
- cd /tmp && pwd
```

- “system ‘ssh’, @ARGV”は

```
“ssh sh -c cd /tmp” && pwd
```

String::ShellQuote

```
use String::ShellQuote;  
> mssh -- sudo sh -c "cd /tmp && pwd"  
XXX ¥@ARGV;  
---  
- sudo  
- sh  
- 'cd /tmp && pwd'
```

- 狙ったとおりに実行できる

Expect.pm

- PerlによるExpectの実装

<http://search.cpan.org/dist/Expect/>

```
my $ssh = Expect->spawn(qw/ssh -l user host/,shell_quote @ARGV);
$ssh->expect(
    180,
    [ qr/assword:¥s*$/ => sub {
        $ssh->send("$pass¥n");
        exp_continue;
    } ],
    [ qr/.+$/ => sub {
        exp_continue;
    } ]
);
```

完成したコード

- 下記エントリー
 - <http://blog.nomadscafe.jp/archives/000712.html>

それ**でもできるよ

- **いろいろな言語、フレームワークでできそう**
 - Expect
 - zsh
 - Capistrano
 - Net::SSH::Perl
 - などなど
- **Perlならではの利点**
 - CPANモジュールの活用。GlueとしてのPerl
 - アプリケーションとの連動、柔軟性、拡張性

ご清聴ありがとうございました