

Puppet zadania

M. Kosobucki M. Siejak

29 maja 2012

1 Instalacja

Starsza wersja Puppeta (2.6) jest dostępna w repozytoriach stabilnego Debiana (squeeze).

1.1 Instalacja serwera

Serwera puppeta znajduje się w paczce `puppetmaster`. Instalujemy ją w standardowy sposób:

```
1 root@as10:~# apt-get install puppetmaster
```

Nie należy martwić się tym, że zainstalowanych będzie dużo zależności. Puppet jest napisany w języku Ruby i korzysta z różnych dodatkowych bibliotek.

Po instalacji serwer plików domyślnie niczego nie przepuszcza. Musimy zatem wyedytować plik `/etc/puppet/fileserver.conf` i wpisać do niego adresy hostów, które będą mogły pobierać pliki z mastera. W moim przypadku kluczowa część pliku wygląda tak:

```
[files]
  path /etc/puppet/files
  allow 150.254.78.0/24
```

Parametr `allow` przyjmuje adresy takie jak `rsyncd.conf`. W przypadku maszyn wirtualnych trzeba ustawić adresy zgodnie z adresacją VMWare Playera. Można też użyć nazw domenowych:

```
[files]
  path /etc/puppet/files
  allow *.as124.pl
```

Należy też utworzyć katalog wskazany w pliku `fileserver.conf` ponieważ domyślnie nie istnieje:

```
root@as10:/etc/puppet# mkdir -p /etc/puppet/files
```

Na koniec startujemy demon mastera:

```
root@as10:/etc/puppet# service puppetmaster start
Starting puppet master.
```

1.2 Instalacja klientów

Na maszynach klienckich instalujemy paczkę `puppet`:

```
root@as11:~# apt-get install puppet
```

Następnie należy wyedytować plik `/etc/default/puppet` i ustawić w nim parametr `START=` następująco:

```
# Start puppet on boot?
START=yes
```

Demon klienta w domyślnej konfiguracji próbuje połączyć się z hostem o nazwie `puppet`. Niestety nasz serwer zapewne wygenerował certyfikaty dla innej nazwy domenowej, więc nie możemy po prostu dodać adresu do `/etc/hosts`. W takim razie, trzeba ustawić serwer w pliku konfiguracyjnym agenta (`/etc/puppet/puppet.conf`):

```
[main]
server=as10.vm.wmi.amu.edu.pl
```

Należy pamiętać o zamienieniu nazwy na nazwę swojego mastera, a także o tym, że trzeba wszystko wykonać na obu hostach, którymi chcemy zarządzać.

Następnie uruchamiamy klienta, jeśli jeszcze nie jest uruchomiony, lub restartujemy:

```
root@as10:/etc/puppet# service puppet start
Starting puppet agent.
```

2 Konfiguracja

2.1 Autoryzacja klientów

Przy pierwszym uruchomieniu, klienci generują parę kluczy, którymi następnie próbują zautoryzować się u mastera.

Listę klientów oczekujących na autoryzację możemy podejrzeć wydając polecenie:

```
root@as10:/etc/puppet# puppetca list
as11.vm.wmi.amu.edu.pl (6F:F9:C3:4D:79:38:B1:39:46:88:B4:8A:98:\
F7:94:DF)
```

Aby zaakceptować klienta wykonujemy polecenie:

```
root@as10:/etc/puppet# puppetca --sign as11.vm.wmi.amu.edu.pl
notice: Signed certificate request for as11.vm.wmi.amu.edu.pl
3 notice: Removing file Puppet::SSL::CertificateRequest as11.vm.wmi.\
amu.edu.pl at '/var/lib/puppet/ssl/ca/requests/as11.vm.wmi.amu\
.edu.pl.pem'
```

Od tego momentu, klient będzie sprawdzał, czy na serwerze nastąpiły jakieś zmiany i ew. je aplikował.

2.2 Stworzenie zasobu

W tym przykładzie, stworzymy klasę, która będzie zapewniała zainstalowanie demona ssh i ustawienie mu odpowiedniej konfiguracji, a w razie zmiany tej konfiguracji, zrestartuje demona.

Dla uproszczenia nie będziemy korzystać z modułów i wszystkie zasoby zdefiniujemy w głównym pliku konfiguracyjnym mastera.

Aby to zrobić, tworzymy plik `/etc/puppet/manifests/site.pp` z następującą zawartością:

```
2 class ssh {
  package { 'openssh-server':
    ensure => present,
    before => file['/etc/ssh/sshd_config'],
```

```

    }
    file{ '/etc/ssh/sshd_config':
7      ensure => present,
        source => 'puppet://as10.vm.wmi.amu.edu.pl/files/etc/ssh/\
            sshd_config',
    }
    service{ 'sshd':
12      ensure => running,
        subscribe=>file['/etc/ssh/sshdconfig'],
    }
}
node default {
17   include ssh
}

```

UWAGA: \LaTeX domyślnie generuje złe znaczki w napisach, sprawdzić przy przeklejeniu, czy wkleja znaczek spod cudzysłowu.

Następnie, tworzymy katalog serwera plików, z którego będziemy serwować plik z konfiguracją:

```
root@as10:/etc# mkdir -p /etc/puppet/files/etc/ssh/
```

Dla uproszczenia, zakładam, że będziemy ustawiali klientom taki sam plik `sshd_config` jak na masterze, dlatego skopiujemy konfigurację mastera do utworzonego przed chwilą katalogu:

```
root@as10:/etc# cp /etc/ssh/sshd_config /etc/puppet/files/etc/ssh/
```

Aby zapamiętać w serwerze nowe ustawienia, należy go przeładować:

```
root@as10:~# service puppetmaster reload
```

Klient domyślnie sprawdza czy master ma do wysłania jakiejś zmiany co pół godziny. Aby wymusić sprawdzenie, wywołujemy polecenie:

```
root@as12:~# puppet agent --test
```

3 Zadania dodatkowe

Skonfigurować puppeta tak, aby na klientach zainstalowany był pakiet `sudo`, a jego plik konfiguracyjny `/etc/sudoers` kopiował się z mastera.