

Gaspacho

C'est bon, mangez-en

GARETTE Emmanuel

COAGUL

14 octobre 2010



Plan

Présentation

Les règles

Les groupes

Les choix

Pratique

API

Futur

Conclusion



Moi

- ▶ GARETTE Emmanuel
- ▶ membre du (ou d'un des) GULL de Dijon COAGUL ;
- ▶ initiateur du projet Gaspacho ;
- ▶ développeur principal de Gaspacho ;
- ▶ ingénieur en logiciel libre ;
- ▶ société : Edenwall Technologie ;
- ▶ projet libre de l'Éducation Nationale EOLE.



La génèse du projet

- ▶ Identification du besoin dans le milieu professionnel : EOLE ;
- ▶ projet de développement à COAGUL ;
- ▶ développement de façon autonome.



Gaspacho

- ▶ Configuration centralisée des postes clients ;
- ▶ une interface de configuration des règles ;
- ▶ un agent de configuration



Les autres projets

- ▶ Propriétaire : GPO de Microsoft ;
- ▶ Libre :
 - ▶ cfengine ;
 - ▶ puppet ;
 - ▶ ESU



Les limites de ces projets

- ▶ Centré sur un type de configuration (registre ou fichier);
- ▶ mais pas sur l'utilisateur;
- ▶ gère des fichiers et non de règle;
- ▶ doit connaître la configuration du programme.



Les avantages de Gaspacho

- ▶ Supporte un ensemble des formats de fichiers de configuration ;
- ▶ gestion des versions des logiciels ;
- ▶ gestion des distributions/systèmes d'exploitation ;
- ▶ une règle configurer différents éléments.



Gaspacho n'est pas

- ▶ Gaspacho n'est pas un logiciel de sécurisation des postes !



Technologies

- ▶ Python
- ▶ Twisted
- ▶ Elixir
- ▶ Ext-JS



Principes de base de Gaspacho

- ▶ Les règles ;
- ▶ les groupes ;
- ▶ les choix.



Plan

Présentation

Les règles

Les groupes

Les choix

Pratique

API

Futur

Conclusion



Les règles : exemple avec le proxy

Windows : clefs de registre

activation :

```
HKEY_CURRENT_USER\Software\Microsoft\Windows\
CurrentVersion\Internet Settings\ProxyEnable : 1
```

configuration :

```
HKEY_CURRENT_USER\Software\Microsoft\Windows\
CurrentVersion\Internet Settings\ProxyServer : 192.168.1.1 :3128
```

restriction :

```
HKEY_CURRENT_USER\Software\Policies\Microsoft\ Internet
Explorer\Control Panel\Proxy : 1
```



Les règles : exemple avec le proxy

Mozilla Firefox : fichier js

```
activation : lockPerf("network.proxy.type", "1");  
configuration IP : lockPerf("network.proxy.http", "192.168.1.1");  
configuration du port : lockPerf("network.proxy.http_port",  
"3128");
```



Les règles : exemple avec le proxy

GNOME : clev gconf

activation : /system/http_proxy/use_http_proxy True

configuration de l'IP : /system/http_proxy/host "192.168.1.1"

configuration du port : /system/http_proxy/port 3128



Les règles : exemple avec le proxy

OpenOffice.org : fichier XML

```
<?xml version="1.0" encoding="UTF-8"?>
<oor:component-data xmlns:oor="http://openoffice.org/2001/ro
  <node oor:name="Settings">
    <prop oor:name="ooInetProxyType" oor:type="xs:int">
      <value>2</value>
    </prop>
    <prop oor:name="ooInetHTTPProxyName" oor:type="xs:string">
      <value>192.168.10.1</value>
    </prop>
    <prop oor:name="ooInetHTTPProxyPort" oor:type="xs:int">
      <value>8080</value>
    </prop>
  </node>
```



Les règles : présentation

- ▶ La règle = un libellé ;
- ▶ classée dans des "tags" / "categories" ;
- ▶ par exemple "Configuration du serveur proxy manuelle" est une règle ;
- ▶ la variable : clef/valeur ;
- ▶ la plateforme : le chemin - le logiciel - le système.



Les règles : propriétés

- ▶ Niveau :
 - ▶ les règles machines ;
 - ▶ les règles utilisateurs.
- ▶ Type :
 - ▶ sans valeur :
 - ▶ "boolean" : vrai/faux ;
 - ▶ avec valeur :
 - ▶ "unicode" : un texte libre ;
 - ▶ "integer" : un nombre ;
 - ▶ "enum" : une liste de choix prédéfinies ;
 - ▶ "list" : une liste de choix libres ;
 - ▶ "multi" : permet de mêler plusieurs types.



Les règles : propriétés des variables

- ▶ Type de règle \neq type de variable ;
- ▶ "boolean" / "unicode" / "integer".



Les règles : les valeurs

- ▶ une valeur si activées ;
- ▶ une valeur si désactivées ;
- ▶ une valeur par défaut si activées "avec valeur".



Les règles : exemple

```
{"name": [{"lang": "fr",  
  "label": "Page de démarrage"}]  
"defaultvalue": "http://www.gaspacho-project.net",  
"type": "unicode",  
"variables": [  
  {"value_off": "IGNORE",  
    "type": "unicode",  
    "name": "browser.startup.homepage",  
    "path": "js://%AppData\\Mozilla\\Firefox\\Profiles\\Defa  
{"value_on": "1",  
  "value_off": "0",  
  "type": "integer",  
  "name": "browser.startup.page",  
  "path": "js://%AppData\\Mozilla\\Firefox\\Profiles\\Defa  
]}
```



Plan

Présentation

Les règles

Les groupes

Les choix

Pratique

API

Futur

Conclusion



Les groupes

- ▶ Administrateur n'a pas a toucher aux règles mais doit faire des groupes ;
- ▶ ensemble de machines avec même configuration ;
- ▶ possibilité de différencier suivant l'utilisateur ;
- ▶ peut limiter le nombre de règle en associant un OS/logiciel.



Les groupes de machine

- ▶ Ensemble de machines;
- ▶ IP/DNS ;
- ▶ motif inspiré du shell Unix :
 - * remplace tout
 - ? remplace un seul caractère
 - [seq] remplace tous caractères dans "seq"
 - [!seq] remplace tous caractères en dehors de "seq"



Les utilisateurs ou groupes d'utilisateurs

- ▶ Nom de l'utilisateur ;
- ▶ groupe primaire ;
- ▶ sinon règle du groupe.



Les templates

- ▶ Template = Groupe sans les machines ;
- ▶ utile dans le cadre de l'héritage.



L'héritage des groupes

- ▶ Mettre en commun des réglages ;
- ▶ Template : associé plusieurs groupes.
- ▶ Exemple :

```
default (None, user1, user2)
|
|--group1 (None, user1, user2)
|
|   |--group2 (None, user1, user2)
|   |
|   |   template1 (None, user1, user2)
```



Plan

Présentation

Les règles

Les groupes

Les choix

Pratique

API

Futur

Conclusion



Les choix

- ▶ Élément central ;
- ▶ configuration particulière d'une règle pour un groupe/template
- ▶ "règle utilisateur" peut être lié à un utilisateur ;
- ▶ si règle avec valeur, peut définir un valeur.



Les choix : type

- ▶ Choix libre ;
- ▶ choix imposé désactivé ;
- ▶ choix imposé activé ;
- ▶ valeur que si choix imposé activé.



Les choix et héritages

- ▶ L'héritage des groupes n'est utile que pour les choix ;
- ▶ le premier choix détecté s'applique ;
- ▶ si pas de choix/choix hérité : choix libre.



Plan

Présentation

Les règles

Les groupes

Les choix

Pratique

API

Futur

Conclusion



Plan

Présentation

Les règles

Les groupes

Les choix

Pratique

API

Futur

Conclusion



API

- ▶ initialisation de la base :

```
from gaspacho.group import *
from gaspacho.rule import *
from gaspacho.category import *
from gaspacho.choice import *
from gaspacho.platform import *
from gaspacho.initialize import *
from gaspacho.importfile import *
initialize_database()
```



API

- ▶ ajouter des groupes :

```
gnunux = add_user(name=u'gnunux')
worker = add_user(name=u'worker', type=u'usergroup')
dev_computer = add_computer(name=u'dev*', type=u'dns')
dev = add_group(name=u'dev')
dev.add_user(gnunux)
dev.add_computer(dev_computer)
```

- ▶ ajouter un template :

```
tplfirefox = add_template(name=u'tplfirefox')
```



API

- ▶ faire un choix :

```
rule = get_rules()[0]
set_choice(rule=rule, group=dev, user=gnunux,
           state=u'on', value="80")
```

- ▶ enregistrer/rollback :

```
commit_database()
rollback_database()
```



Plan

Présentation

Les règles

Les groupes

Les choix

Pratique

API

Futur

Conclusion



Futur

- ▶ Stabiliser le code actuel pour sortir un 0.1 (https, twisted, bugs connus, ...);
- ▶ ajout des managers;
- ▶ possibilité de définir un choix différent par application pour une règle;
- ▶ développement des agents (GNU/linux, Microsoft Windows, MacOS, ...);
- ▶ ...



Plan

Présentation

Les règles

Les groupes

Les choix

Pratique

API

Futur

Conclusion



Gaspacho c'est bon

Ingrédients pour 4 personnes :

3 tranches de pain

3 gousses d'ail

sel et poivre

4 c.soupe d'huile d'olive

500 g de tomates

1 poivron

1 concombre

1 oignon

1/2 litre d'eau

2 c.soupe de vinaigre de Xérès (ou de sherry)

<http://www.cuisine-libre.fr/gazpacho>

