

> what is ansible* ?

PUBLIC



ansible / ansible

★ Star

5,276

Fork

1,571

Ansible is a radically simple IT automation platform that makes your applications and systems easier to deploy. Avoid writing scripts or custom code to deploy and update your applications— automate in a language that approaches plain English, using SSH, with no agents to install on remote systems. <http://ansible.com/>

"infrastructure as code"

Ansible is a Top 10 Python project on GitHub with over 5,000 ★s and thousands of great users like...

synthesis of;

- configuration management, automation tools (Puppet, Chef, cfengine)
- deployment tools (Capistrano, Fabric)
- ad-hoc task execution tools (Func, pssh, etc)

*“Ansible is a **configuration management**, **application deployment** and **IT Orchestration tool**”*

> me

@serkancapkan

nl.linkedin.com/in/serkancapkan

twitter.com/serkancapkan

system administrator @Liberty Global (NL)

~~ipara.com, sahibinden.com, TMMOB EMO, ITU~~

> summary

- why we need configuration management?
- what is ansible?
- howto: installation & configuration & runtime
- discussions
- questions

> why we need configuration management?

- system installation & provisioning
- system configuration
- service installation
- application deployment
- configuration management
 - system based
 - service based
 - application based

> why we need configuration management?

- change management
 - system based
 - service based
 - application based
- automate repetitive tasks
- configuration integrity
- documentation (system based)

> why we need configuration management?

- continuous deployment (ansible&jenkins)
- packaging software product (software | software+system)
- remote command execution
- auto scalability (system base, auto)
- be able to create same systems in case of emergency
- create different environments (dev, staging, load test)

> what is ansible* ?

PUBLIC  **ansible / ansible**

★ Star

5,276

Fork

1,571

Ansible is a radically simple IT automation platform that makes your applications and systems easier to deploy. Avoid writing scripts or custom code to deploy and update your applications— automate in a language that approaches plain English, using SSH, with no agents to install on remote systems. <http://ansible.com/>

"infrastructure as code"

Ansible is a Top 10 Python project on GitHub with over 5,000 ★s and thousands of great users like...

synthesis of;

- configuration management, automation tools (Puppet, Chef, cfengine)
- deployment tools (Capistrano, Fabric)
- ad-hoc task execution tools (Func, pssh, etc)

*“Ansible is a **configuration management, application deployment** and **IT Orchestration tool**”*

> what is ansible* ?

- Michael DeHaan (@laserllama)
- python
- ssh authenticate
- ansible works, company
- free (speech & beer)
- feb 2012
- 643+ contributor

*Rocannon's World, Dispossessed

> what is ansible* ?

- radically simple
- fast, low resource usage
- no requirement (except Python 2.4 or later)
- push based (pull also possible)
- agentless
- no specific port
- no new authentication integration

> what is ansible* ?

- basic tasks basic
- complicated projects possible, easy to implement
 - playbook
 - roles
- playbooks use yaml format (like ad hoc)(:/)
 - task executed in order
- highly scalable (+6000 parallel)

> what is ansible* ?

- integratable (json output)
- robust module architecture
 - part of core code, 1 module for 1 work
 - write your own module in any language (json)
 - ansible galaxy
- ansible tower (formerly 'AWX') (web ui) (\$)

> howto: installation & configuration & runtime

- installation

- source

```
~# git clone git://github.com/ansible/ansible.git
~# cd ./ansible
~# source ./hacking/env-setup
~# sudo easy_install pip
~# sudo pip install paramiko PyYAML jinja2 httpplib2
~# export ANSIBLE_HOSTS=~/.ansible_hosts
```

- pip

```
~# sudo easy_install pip
~# sudo pip install ansible
```

- package manager

yum - EPEL

apt- PPA

> howto: installation & configuration & runtime

- installation
 - ssh key authentication
 - inventory

```
[webservers]
w1.example.com
w2.example.com

[mysql]
db1.example.com
db2.example.com
db3.example.com


[mongodb]
mg1.example.com
mg2.example.com ansible_ssh_port=22 ansible_ssh_host=10.0.2.7
mg3.example.com
```

- go!

> howto: installation & configuration & runtime

- runtime (ad-hoc)

ansible [host] -m [module] -a [arguments]



```
root@sys01:~# ansible all -m shell -a 'uptime'
etrn | success | rc=0 >>
  02:43:31 up 31 days,  4:53,  3 users,  load average: 0.00, 0.01, 0.05

gui | success | rc=0 >>
  02:43:14 up 55 days, 12:27,  3 users,  load average: 0.25, 0.12, 0.08

zimbra | success | rc=0 >>
  02:43:49 up 2 days, 10:17,  2 users,  load average: 0.05, 0.13, 0.14

pmta2 | success | rc=0 >>
  02:44:57 up 54 days, 16:08,  2 users,  load average: 0.04, 0.01, 0.00
```

> howto: installation & configuration & runtime

- runtime (ad-hoc)

```
root@sys01:~# ansible gui -a "uptime" -vvvvvvv
<10.0.2.6> ESTABLISH CONNECTION FOR USER: root
<10.0.2.6> EXEC ['ssh', '-tt', '-vvv', '-o', 'ControlMaster=auto', '-o', 'ControlPersist=60s', '-o', 'ControlPath=/root/.ansible/cp/ansible-ssh-%h-%p-%r', '-o', 'Port=22', '-o', 'KbdInteractiveAuthentication=no', '-o', 'PreferredAuthentications=gssapi-with-mic,gssapi-keyex,hostbased,publickey', '-o', 'PasswordAuthentication=no', '-o', 'ConnectTimeout=10', '10.0.2.6', '/bin/sh -c 'mkdir -p $HOME/.ansible/tmp/ansible-1395954103.75-229998591962740 && echo $HOME/.ansible/tmp/ansible-1395954103.75-229998591962740'"']
<10.0.2.6> REMOTE_MODULE command uptime
<10.0.2.6> PUT /tmp/tmpbxLGmF TO /root/.ansible/tmp/ansible-1395954103.75-229998591962740/command
<10.0.2.6> EXEC ['ssh', '-tt', '-vvv', '-o', 'ControlMaster=auto', '-o', 'ControlPersist=60s', '-o', 'ControlPath=/root/.ansible/cp/ansible-ssh-%h-%p-%r', '-o', 'Port=22', '-o', 'KbdInteractiveAuthentication=no', '-o', 'PreferredAuthentications=gssapi-with-mic,gssapi-keyex,hostbased,publickey', '-o', 'PasswordAuthentication=no', '-o', 'ConnectTimeout=10', '10.0.2.6', '/bin/sh -c '/usr/bin/python /root/.ansible/tmp/ansible-1395954103.75-229998591962740/command; rm -rf /root/.ansible/tmp/ansible-1395954103.75-229998591962740/ >/dev/null 2>&1'"']
gui | success | rc=0 >>
23:02:02 up 55 days, 8:45, 3 users, load average: 0.00, 0.02, 0.05
```

> howto: installation & configuration & runtime

```
root@sys01:~# ansible etrn -m apt -a "name=nginx state=installed"
etrn | success >> {
  "changed": true,
  "stderr": "Committing to: /etc/\nadded nginx\nadded default/nginx\nadded init.d/nginx\nadded logrotate.d/nginx\nadded nginx/conf.d\nadded nginx/fastcgi_
params\nadded nginx/koi-utf\nadded nginx/koi-win\nadded nginx/mime.types\nadded nginx/nginx.conf\nadded nginx/proxy_params\nadded nginx/scgi_params\nadded n
ginx/sites-available\nadded nginx/sites-enabled\nadded nginx/uwsgi_params\nadded nginx/win-utf\nadded nginx/sites-available/default\nadded nginx/sites-enabl
ed/default\nadded rc0.d/K20nginx\nadded rc1.d/K20nginx\nadded rc2.d/S20nginx\nadded rc3.d/S20nginx\nadded rc4.d/S20nginx\nadded rc5.d/S20nginx\nadded rc6.d/
K20nginx\nadded ufw/applications.d/nginx\nadded ufw/applications.d/nginx/ufw.profile\nCommitted revision 4.\n",
  "stdout": "Reading package lists...\nBuilding dependency tree...\nReading state information...\nThe following extra packages will be installed:\n libgd
2-noxpm libjpeg62 libxslt1.1 nginx-common nginx-full\nSuggested packages:\n libgd-tools\nThe following NEW packages will be installed:\n libgd2-noxpm libj
peg62 libxslt1.1 nginx nginx-common nginx-full\n0 upgraded, 6 newly installed, 0 to remove and 3 not upgraded.\nNeed to get 830 kB of archives.\nAfter this
operation, 2,601 kB of additional disk space will be used.\nGet:1 http://us.archive.ubuntu.com/ubuntu/ oneiric/main libjpeg62 i386 6b1-1ubuntu2 [87.3 kB]\nG
et:2 http://us.archive.ubuntu.com/ubuntu/ oneiric/main libxslt1.1 i386 1.1.26-7 [164 kB]\nGet:3 http://us.archive.ubuntu.com/ubuntu/ oneiric/main libgd2-nox
pm i386 2.0.36-rc1-dfsg-5.1ubuntu1 [200 kB]\nGet:4 http://us.archive.ubuntu.com/ubuntu/ oneiric-updates/universe nginx-common all 1.0.5-1ubuntu0.1 [14.6 kB]
\nGet:5 http://us.archive.ubuntu.com/ubuntu/ oneiric-updates/universe nginx-full i386 1.0.5-1ubuntu0.1 [359 kB]\nGet:6 http://us.archive.ubuntu.com/ubuntu/
oneiric-updates/universe nginx all 1.0.5-1ubuntu0.1 [5,602 B]\nFetched 830 kB in 7s (113 kB/s)\nSelecting previously deselected package libjpeg62.\nReading
ng database ... \n(Reading database ... 5%\n(Reading database ... 10%\n(Reading database ... 15%\n(Reading database ... 20%\n(Reading database ... 25%\n(Rea
ding database ... 30%\n(Reading database ... 35%\n(Reading database ... 40%\n(Reading database ... 45%\n(Reading database ... 50%\n(Reading database ... 55%
\n(Reading database ... 60%\n(Reading database ... 65%\n(Reading database ... 70%\n(Reading database ... 75%\n(Reading database ... 80%\n(Reading database .
.. 85%\n(Reading database ... 90%\n(Reading database ... 95%\n(Reading database ... 100%\n(Reading database ... 52137 files and directories currently instal
led.)\n\nUnpacking libjpeg62 (from ../libjpeg62_6b1-1ubuntu2_i386.deb) ...\n\nSelecting previously deselected package libxslt1.1.\n\nUnpacking libxslt1.1 (
from ../libxslt1.1_1.1.26-7_i386.deb) ...\n\nSelecting previously deselected package libgd2-noxpm.\n\nUnpacking libgd2-noxpm (from ../libgd2-noxpm_2.0.36~
rc1-dfsg-5.1ubuntu1_i386.deb) ...\n\nSelecting previously deselected package nginx-common.\n\nUnpacking nginx-common (from ../nginx-common_1.0.5-1ubuntu0.1
_all.deb) ...\n\nSelecting previously deselected package nginx-full.\n\nUnpacking nginx-full (from ../nginx-full_1.0.5-1ubuntu0.1_i386.deb) ...\n\nSelectin
g previously deselected package nginx.\n\nUnpacking nginx (from ../nginx_1.0.5-1ubuntu0.1_all.deb) ...\n\nProcessing triggers for ufw ...\n\nProcessing tri
ggers for ureadahead ...\n\nProcessing triggers for man-db ...\n\nSetting up libjpeg62 (6b1-1ubuntu2) ...\n\nSetting up libxslt1.1 (1.1.26-7) ...\n\nSetting
up libgd2-noxpm (2.0.36-rc1-dfsg-5.1ubuntu1) ...\n\nSetting up nginx-common (1.0.5-1ubuntu0.1) ...\n\nSetting up nginx-full (1.0.5-1ubuntu0.1) ...\n\nSetti
ng up nginx (1.0.5-1ubuntu0.1) ...\n\nProcessing triggers for libc-bin ... \n\nldconfig deferred processing now taking place\n\nroot <root@etrn.etrn.com>\n"
}
```


> howto: installation & configuration & runtime

- runtime (ad-hoc)

```
root@sys01:~# ansible all -m copy -a "src=/tmp/test dest=/tmp/test mode=600 owner=root group=root"
etrn | success >> {
  "changed": false,
  "gid": 0,
  "group": "root",
  "mode": "0600",
  "owner": "root",
  "path": "/tmp/test",
  "size": 0,
  "state": "file",
  "uid": 0
}

gui | success >> {
  "changed": false,
  "gid": 0,
  "group": "root",
  "mode": "0600",
  "owner": "root",
  "path": "/tmp/test",
  "size": 0,
  "state": "file",
  "uid": 0
}
```

> howto: installation & configuration & runtime

- runtime (ad-hoc)

```
root@sys01:~# ansible etrn -m service -a "name=nginx state=stopped"
etrn | success >> {
  "changed": false,
  "name": "nginx",
  "state": "stopped"
}
```

> howto: installation & configuration & runtime

- runtime (playbook)

```
---
- hosts: all
  tasks:
    - name: basic packages installations
      apt: name={{ item }} state=present
      with_items:
        - iptraf
        - zabbix-agent
      when: ansible_os_family == "Debian"
      notify: start_services
    - name: basic packages installations
      yum: name={{ item }} state=present
      with_items:
        - iptraf
        - zabbix-agent
      when: ansible_os_family == "RedHat"
      notify: start_services
  handlers:
    - name: start_services
      service: name={{ item }} state=started
      with_items:
        - zabbix-agent
```

> howto: installation & configuration & runtime

- runtime (playbook)

```
root@sys01:~# ansible-playbook /etc/ansible/playbooks/00-installation.yml

PLAY [all] *****

TASK: [basic packages installations] *****
skipping: [pmta2]
changed: [etrn] => (item=iptraf,zabbix-agent)
changed: [gui] => (item=iptraf,zabbix-agent)
changed: [zimbra] => (item=iptraf,zabbix-agent)

TASK: [basic packages installations] *****
skipping: [gui]
skipping: [zimbra]
skipping: [etrn]
ok: [pmta2] => (item=iptraf,zabbix-agent)

NOTIFIED: [start_services] *****
ok: [gui] => (item=zabbix-agent)
ok: [zimbra] => (item=zabbix-agent)
ok: [etrn] => (item=zabbix-agent)

PLAY RECAP *****
etrn      : ok=3    changed=1    unreachable=0    failed=0
gui       : ok=3    changed=1    unreachable=0    failed=0
pmta2     : ok=2    changed=0    unreachable=0    failed=0
zimbra    : ok=3    changed=1    unreachable=0    failed=0
```

> howto: installation & configuration & runtime

- runtime (playbook adhoc)

```
---
- hosts: all
  tasks:
    - name: apt installation
      apt: name={{ package }} state=present
      when: ansible_os_family == "Debian"
      notify: start_services
    - name: yum installation
      yum: name={{ package }} state=present
      when: ansible_os_family == "RedHat"
      notify: start_services
  handlers:
    - name: start_services
      service: name={{ package }} state=started
```

> howto: installation & configuration & runtime

- runtime (playbook adhoc)

```
root@sys01:~# ansible-playbook /etc/ansible/playbooks/00-adhoc-installation.yml --extra-vars "package=memcached"
```

```
PLAY [all] *****
```

```
GATHERING FACTS *****
```

```
ok: [gui]
```

```
ok: [pmta2]
```

```
TASK: [apt installation] *****
```

```
skipping: [pmta2]
```

```
changed: [gui]
```

```
TASK: [yum installation] *****
```

```
skipping: [gui]
```

```
failed: [pmta2] => {"changed": false, "failed": true, "rc": 0, "results": []}
```

```
msg: No Package matching 'memcached' found available, installed or updated
```

```
NOTIFIED: [start_services] *****
```

```
ok: [gui]
```

```
PLAY RECAP *****
```

```
to retry, use: --limit @/root/00-adhoc-installation.retry
```

```
gui          : ok=3    changed=1    unreachable=0    failed=0
```

```
pmta2        : ok=1    changed=0    unreachable=0    failed=1
```

> howto: installation & configuration & runtime

- runtime (playbook - role)

```
.
├─ group_vars
├─ roles
│   ├── common
│   │   ├── files
│   │   ├── handlers
│   │   └── tasks
│   ├── mysql
│   │   ├── handlers
│   │   ├── tasks
│   │   └── templates
│   ├── nginx
│   │   ├── handlers
│   │   ├── tasks
│   │   └── templates
│   ├── php-fpm
│   │   ├── handlers
│   │   ├── tasks
│   │   └── templates
│   └── wordpress
│       ├── tasks
│       └── templates
```


```
├─ group_vars
│   └─ all
├─ hosts
├─ roles
│   ├── common
│   │   ├── files
│   │   │   ├── epel.repo
│   │   │   ├── iptables-save
│   │   │   └── RPM-GPG-KEY-EPEL-6
│   │   ├── handlers
│   │   │   └─ main.yml
│   │   ├── tasks
│   │   │   └─ main.yml
│   ├── mysql
│   │   ├── handlers
│   │   │   └─ main.yml
│   │   ├── tasks
│   │   │   └─ main.yml
│   │   ├── templates
│   │   │   └─ my.cnf.j2
│   ├── nginx
│   │   ├── handlers
│   │   │   └─ main.yml
│   │   ├── tasks
│   │   │   └─ main.yml
│   │   ├── templates
│   │   │   └─ default.conf
│   ├── php-fpm
│   │   ├── handlers
│   │   │   └─ main.yml
│   │   ├── tasks
│   │   │   └─ main.yml
│   │   ├── templates
│   │   │   └─ wordpress.conf
│   └── wordpress
│       ├── tasks
│       │   └─ main.yml
│       └── templates
│           └─ wp-config.php
└─ site.yml
```

site.yml

```
---
- name: Install Wordpress, MySQL, Nginx, and PHP-FPM
  hosts: wordpress-server
  user: root

  roles:
    - common
    - mysql
    - nginx
    - php-fpm
    - wordpress
```

> howto: ansible-galaxy

AboutExploreBrowse RolesBrowse UsersAdd a Rolesrknc

Explore

Top Categories		Top Roles			New Roles	
Category	Total Roles	Role	# Ratings	Score	Role	Added On
system	165	gujo.collectd	1	5.00	gregswift.plight	26 Mar, 2014
development	108	rtuin.zendserver	1	5.00	adham.helal.zabbix_server	26 Mar, 2014
web	97	bennojoy.openldap_server	2	4.90	pbyule.php-cli	25 Mar, 2014
packaging	37	nicolai86.rails-deployment	2	4.90	CajuCLC.ansible-rax-mag...	25 Mar, 2014
monitoring	35	Mayeu.RabbitMQ	2	4.90	geerlingguy.svn2git	25 Mar, 2014
database	28	gdamjan.uwsgi	2	4.80	hudecof.ntp	25 Mar, 2014
networking	33	laggyluke.nodejs	1	4.80	geerlingguy.svn	24 Mar, 2014
database:sql	15	dareko.oracle_client	1	4.80	kylejohnson.QuickBooks	24 Mar, 2014
database:nosql	17	Ansibles.mysql	1	4.80	kylejohnson.plex	24 Mar, 2014
cloud	16	joshualund.ruby-common	1	4.50	altermn.rvm	23 Mar, 2014
View more...		View more...			View more...	

View All Roles

> howto: ansible-galaxy

ansible-galaxy install username.rolename

ANSIBLE GALAXY

[About](#) [Explore](#) [Browse Roles](#) [Browse Users](#) [Add a Role](#) [srknc](#)

Explore

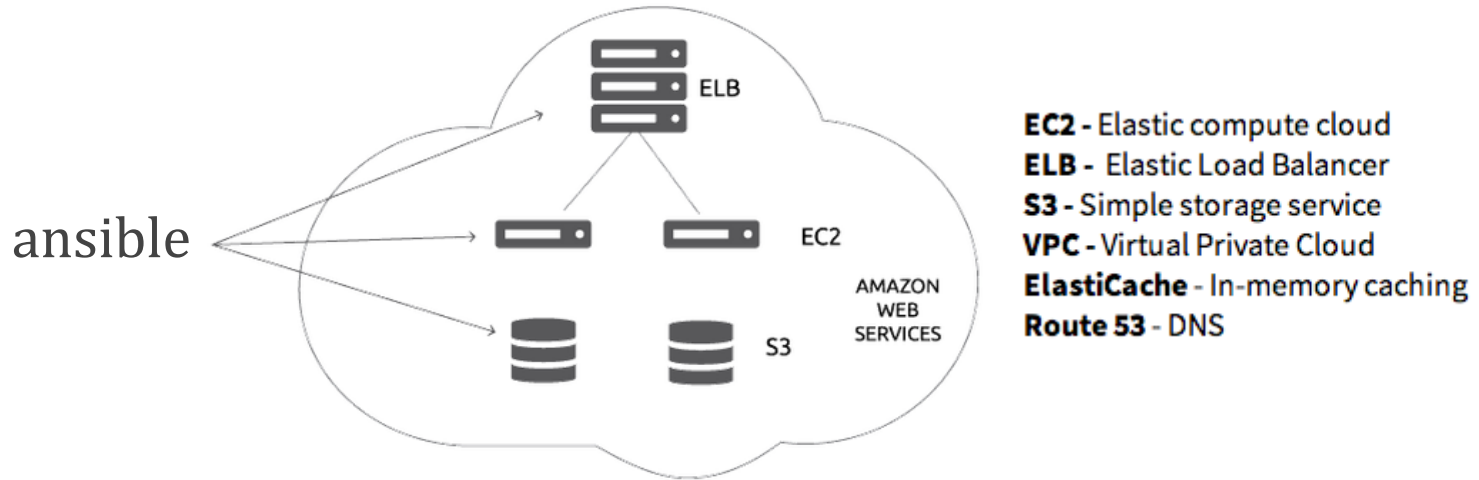
Top Categories		Top Roles			New Roles	
Category	Total Roles	Role	# Ratings	Score	Role	Added On
system	165	gujo.collectd	1	5.00	gregswift.plight	26 Mar, 2014
development	108	rtuin.zendserver	1	5.00	adham.helal.zabbix_server	26 Mar, 2014
web	97	bennojoy.openldap_server	2	4.90	pbyule.php-cli	25 Mar, 2014
packaging	37	nicolai86.rails-deployment	2	4.90	CajuCLC.ansible-rax-mag...	25 Mar, 2014
monitoring	35	Mayeu.RabbitMQ	2	4.90	geerlingguy.svn2git	25 Mar, 2014
database	28	gdamjan.uwsgi	2	4.80	hudecof.ntp	25 Mar, 2014
networking	33	laggyluke.nodejs	1	4.80	geerlingguy.svn	24 Mar, 2014
database:sql	15	dareko.oracle_client	1	4.80	kylejohnson.QuickBooks	24 Mar, 2014
database:nosql	17	Ansibles.mysql	1	4.80	kylejohnson.plex	24 Mar, 2014
cloud	16	joshualund.ruby-common	1	4.50	altermn.rvm	23 Mar, 2014
View more...		View more...			View more...	

View All Roles

> howto: installation & configuration & runtime

- cloud

- amazon, openstack, digital ocean, netScaler, google compute engine
- [ec2 workshop]



> discussions

- system administration

- system development

- devops

> extra

- <http://www.ansible.com/blog/2013/12/08/the-origins-of-ansible>
- <http://www.ansible.com/blog/2013/11/29/ansibles-architecture-beyond-configuration-management>
- <http://docs.ansible.com/>
- <http://highscalability.com/blog/2012/4/18/ansible-a-simple-model-driven-configuration-management-and-c.html>