

Introduction to Ansible

System Orchestration Tool

Toshaan Bharvani - VanTosh bvba

<toshaan@vantosh.com>



RMLL/LSM 2013



11 July 2013

Toshaan Bharvani

- From Antwerp, Belgium
- Currently self-employed : VanTosh
<http://www.vantosh.com>
- Involved with Enterprise Linux : RHEL, CentOS, SLES, ...
- Likes to keep everything secure : SELinux, Web, ...
- Lives in a virtual world : KVM, Xen, LXC, ...
- Likes automation : Ansible
- Works on both hardware and software side
- Wants to take over the world
- Twitter : [@toshywoshy](#)
- Blog : <http://www.toshaan.com>

1 What is Ansible?

2 Modus operandi

3 Getting started

4 Conclusion

1

What is Ansible?

Introducing



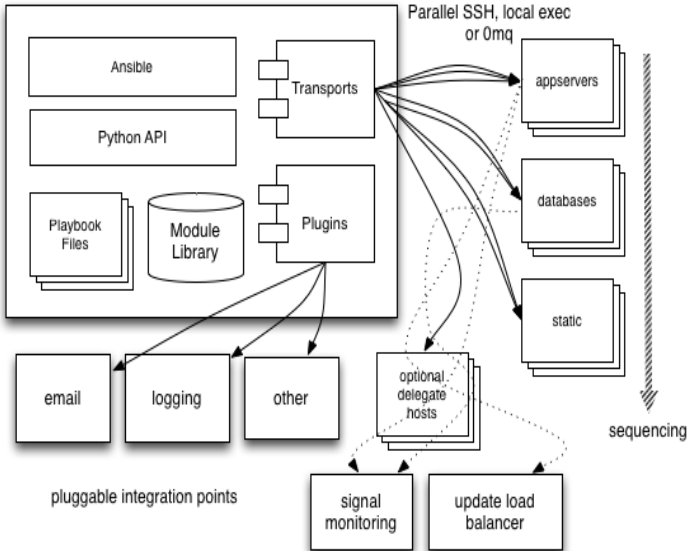
- Configuration Management Tool
- System Orchestration Tool
- Remote Execution/Deployment Tool

- Python based
- Server based, agentless¹
- uses SSH (can use ZeroMQ aka fireball)
- host information in flat files, CMDB, scripts, ...
- executes a task on the host side
- Playbook : combination of tasks with meta information
 - YAML
 - JSON
- Templates : Jinja2
- works where Python works²

¹ However in certain setup an agent might be required, but by default it is not necessary

² Ansible can actually run without Python on the remote host, however it is not fully supported

Internal Design



2

Modus operandi

Adhoc Mode

Toshaan
Bharvani -
VanTosh
bvba

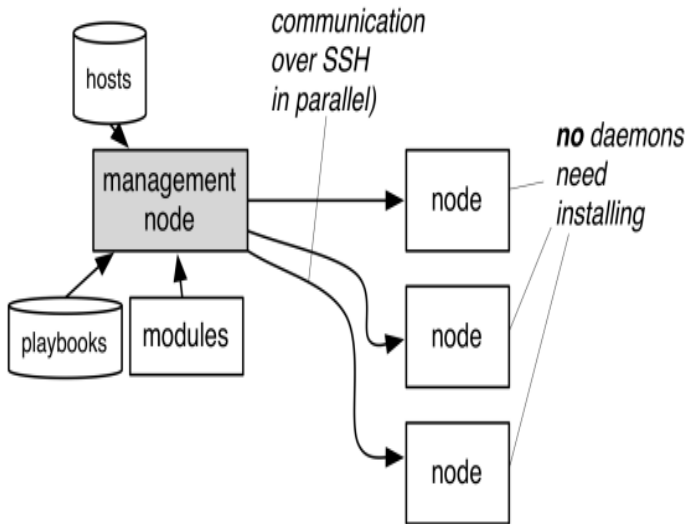
What is
Ansible?

Modus
operandi

Getting
started

Conclusion

The End



Delegation Mode

Toshaan
Bharvani -
VanTosh
bvba

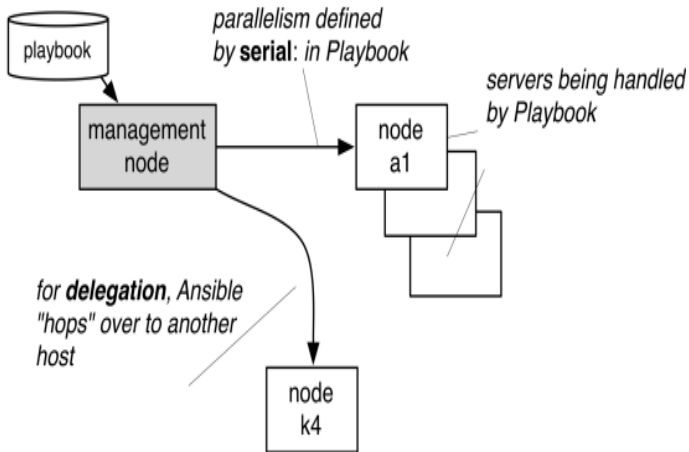
What is
Ansible?

Modus
operandi

Getting
started

Conclusion

The End



Fireball Mode

Toshaan
Bharvani -
VanTosh
bvba

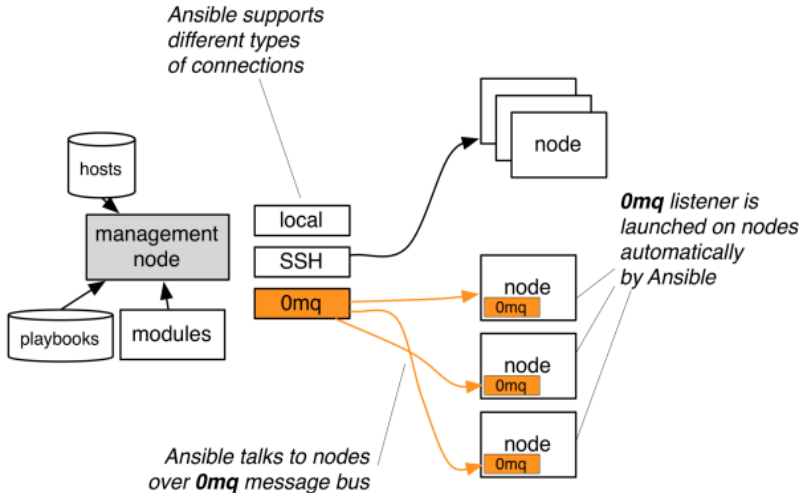
What is
Ansible?

Modus
operandi

Getting
started

Conclusion

The End



Pull Mode

Toshaan
Bharvani -
VanTosh
bvba

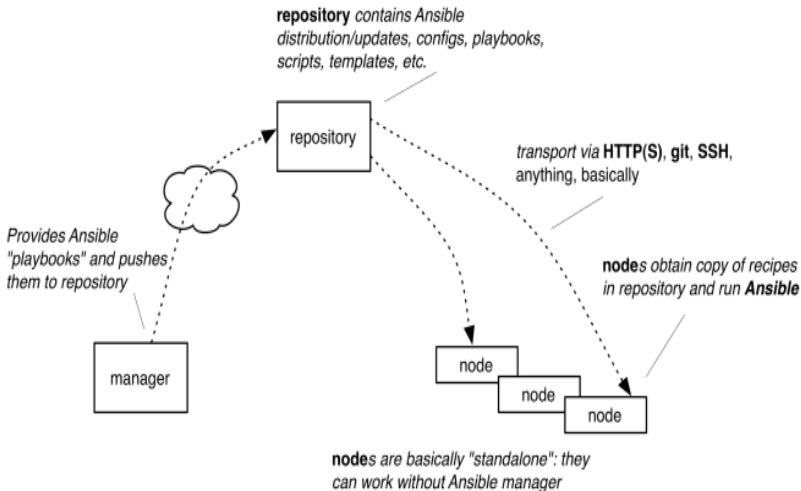
What is
Ansible?

Modus
operandi

Getting
started

Conclusion

The End



3

Getting started

- hosts file is by default in ini format

```
1 [vantosh]
2     main ansible_ssh_host=main.vantosh.com ansible_ssh_port
3         =22222 ansible_ssh_user=ansible
4         ansible_ssh_private_key_file=/path/to/my/private/key
5
6 [belgium]
7     bier.belgium.int
8     chocolates.belgium.be
9     diamonds.antwerp.com
10
11 [beer]
12     bier.belgium.int
13     pivo.russia.int
14     beer.uk.co.int
```

- hosts vars and group vars folder and files

- can be an executable script that outputs json

```
1  #!/usr/bin/env python
2
3  import sqlite3
4  import sys
5  import json
6
7  dbname = 'hosts.db3'
8
9  vars = {}
10
11 con = sqlite3.connect(dbname)
12 con.row_factory=sqlite3.Row
13 cur = con.cursor()
14
15 cur.execute("SELECT COUNT(*) FROM hosts WHERE name=?", (name
16 , ))
17 row = cur.fetchone()
18 print json.dumps(row, indent=4)
19 con.close()
```

```
1 {
2   "ansible_facts": {
3     "ansible_architecture": "x86_64",
4     "ansible_distribution": "CentOS",
5     "ansible_distribution_release": "Final",
6     "ansible_distribution_version": "6.4",
7     "ansible_form_factor": "Mini Tower",
8     "ansible_fqdn": "d54C610ED.access.telenet.be",
9     "ansible_hostname": "sani",
10    "ansible_kernel": "2.6.32-358.6.1.el6.x86_64",
11    "ansible_machine": "x86_64",
12    "ansible_os_family": "RedHat",
13    "ansible_pkg_mgr": "yum",
14    "ansible_processor": [
15      "Intel(R) Xeon(R) CPU E31220 @ 3.10GHz"
16    ],
17    "ansible_processor_cores": 4,
18    "ansible_processor_count": 1,
19    "ansible_processor_threads_per_core": 1,
20    "ansible_processor_vcpus": 4,
21    "ansible_python_version": "2.6.6",
22    "ansible_selinux": {
23      "config_mode": "enforcing",
24      "mode": "enforcing",
25      "policyvers": 24,
26      "status": "enabled",
27      "type": "targeted"
28    },
29    "ansible_virtualization_role": "host",
30    "ansible_virtualization_type": "kvm",
31  }
```


- cloud : cloudformation, digital_ocean, ec2, linode, rax, rds, s3, **virt**, openstack
- commands : **command**, raw, script, **shell**
- database : mongodb, mysql, postgresql, riak
- files : assemble, **copy**, fetch, file, ini_file, **lineinfile**, stat, **template**
- inventory : add_host, group_by
- messaging : rabbitmq
- monitoring : airbrake_deployment, datadog_event, monit, nagios, newrelic_deployment, pagerduty, pingdom
- net_infrastructure : arista, bigip_pool, netscaler
- network : **get_url**, slurp, uri
- notification : campfire, flowdock, hipchat, irc, jabber, mail, mqtt, osx_say
- packaging : **apt**, easy_install, gem, homebrew, npm, openbsd_pkg, opkg, pacman, pip, pkgin, pkgng, redhat, svr4pkg, **yum**, zypper
- source_control : bsr, **git**, hg, subversion
- system : **authorized_key**, cron, facter, filesystem, group, lvg, lvvol, mount, ohai, ping, **seboolean**, **selinux**, **service**, **setup**, sysctl, user, zfs
- utilities : debug, fail, fireball, pause, set_fact, wait_for
- web_infrastructure : django_manage, httpasswd, supervisorctl

```
1
2 # ping all hosts
3 ansible all -m ping
4
5 # run setup/facts module
6 ansible all -m setup
7
8 # run the command dd
9 ansible all -m command -a 'dd if=/dev/zero of=/tmp/ddfile bs=1024
   count=1000'
10
11 # install latest tmux package
12 ansible all -m yum -a name=tmux state=latest
```

Playbooks

Toshaan
Bharvani -
VanTosh
bvba

What is
Ansible?

Modus
operandi

Getting
started

Conclusion

The End

```
1  ---
2  - hosts: webservers
3    vars:
4      http_port: 80
5      max_clients: 200
6      http_virtual_host: {{ virtual_host_name }}
7    user: root
8    sudo: yes
9    sudo_user: postgres
10   serial: 5
11   tasks:
12     - name: take out of load balancer pool
13       action: command /usr/bin/take_out_of_pool {{
14         inventory_hostname }}
15       delegate_to: 127.0.0.1
16     - name: ensure apache is at the latest version
17       action: yum pkg=httpd state=latest
18     - name: write the apache config file
19       action: template src=/srv/httpd.j2 dest=/etc/httpd.conf
20       notify:
21         - restart apache
22     - name: ensure apache is running
23       action: service name=httpd state=started
24     - name: add back to load balancer pool
25       action: command /usr/bin/add_back_to_pool {{
26         inventory_hostname }}
27       delegate_to: 127.0.0.1
28   handlers:
29     - name: restart apache
30       action: service name=httpd state=restarted
```

```
1 # {{ ansible_managed }}
2
3 {# editmode is either "vi" or "emacs" #}
4
5 set -g prefix C-a
6 set -g status-utf8 on
7 setw -g mode-keys {{ editmode }}
```

```
1  ---
2  # Initialize fireball
3  - hosts: 0mq
4    gather_facts: no
5    connection: ssh
6    user: fb
7    sudo: yes
8    tasks:
9      - action: fireball
10
11 # fireball now!
12 - hosts: nameservers
13   connection: fireball
14   tasks:
15     - action: shell echo "Hello {{ item }}"
16       with_items:
17         - one
18         - two
```

```
1  #!/usr/bin/env python
2
3  import ansible.runner
4  import sys
5
6  res = ansible.runner.Runner(
7      pattern='a1*',
8      module_name='command',
9      module_args='/usr/bin/uname -a'
10     ).run()
11
12  print res
```

4

Conclusion

Automation **MUST** [RFC2119] be easy





Thank You



Toshaan Bharvani - VanTosh bvba <toshaan@vantosh.com>

<http://www.vantosh.com/publications>

Made with Beamer \LaTeX
a \TeX based Presentation program