

SaltStack on Linux

Mike Place | Sr. Core Engineer

#LinuxCON15



Welcome!

Do you have enough Salt in your diet?

- Goals

- Architecture
- Modules
- Grains
- Pillars
- States

- Live Demos

- Because staring at slides is the ABSOLUTE WORST.

What is SaltStack

Components

- Configuration Management
- Remote Execution
- Open Event System
- Layered Architecture
- Simplicity

SaltStack Architecture

Master - Minion

SALTSTACK ARCHITECTURE



Salt Components

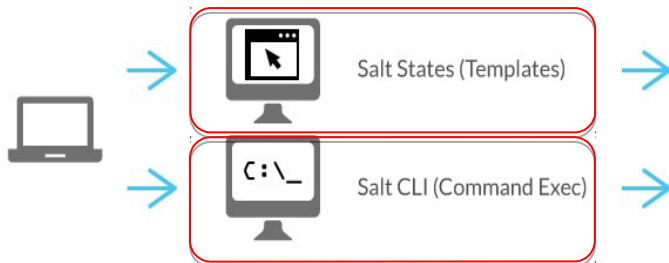
Interactions

SALTSTACK USER EXPERIENCE

SYS ADMIN



SALTSTACK TOOLS



SALT MASTER

Api
Reactor
File Server
Cmd Exec



rsa / curve / aes

SSH
ØMQ
UDP/TCP

SALT MINIONS (n)



OS / App / Config

Grains

Concepts

- A Salt grain is data sourced from a minion.
 - OS
 - Architecture
 - Path
 - IP information

Salt Pillar

Concepts

- Static data sourced from a Salt master that is shared to targeted minions.
- Can be used for standardized configurations across multiple hosts.
 - Web Farms all running the same configuration

Salt Services

Split Packages

- salt /salt-common
- salt-master
- salt-minion
- salt-cloud
- salt-ssh
- salt-api
- salt-doc
- salt-syndic

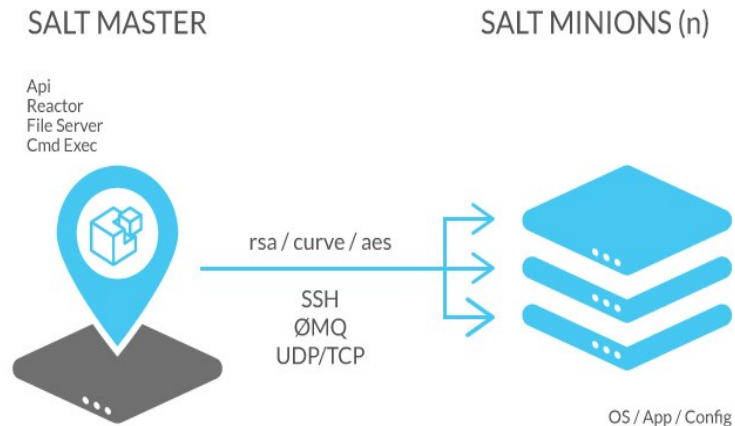
Salt Security

salt-key

- Salt uses public/private key
 - Authentication
 - Identification

Accepted Keys:
minion

Unaccepted Keys:
Rejected Keys:



Salt Files

Configuration

- Files and Directories

- `/etc/salt`

- `/var/cache/salt`

- `/var/log/salt`

- `/run/salt`

- `/etc/init.d`

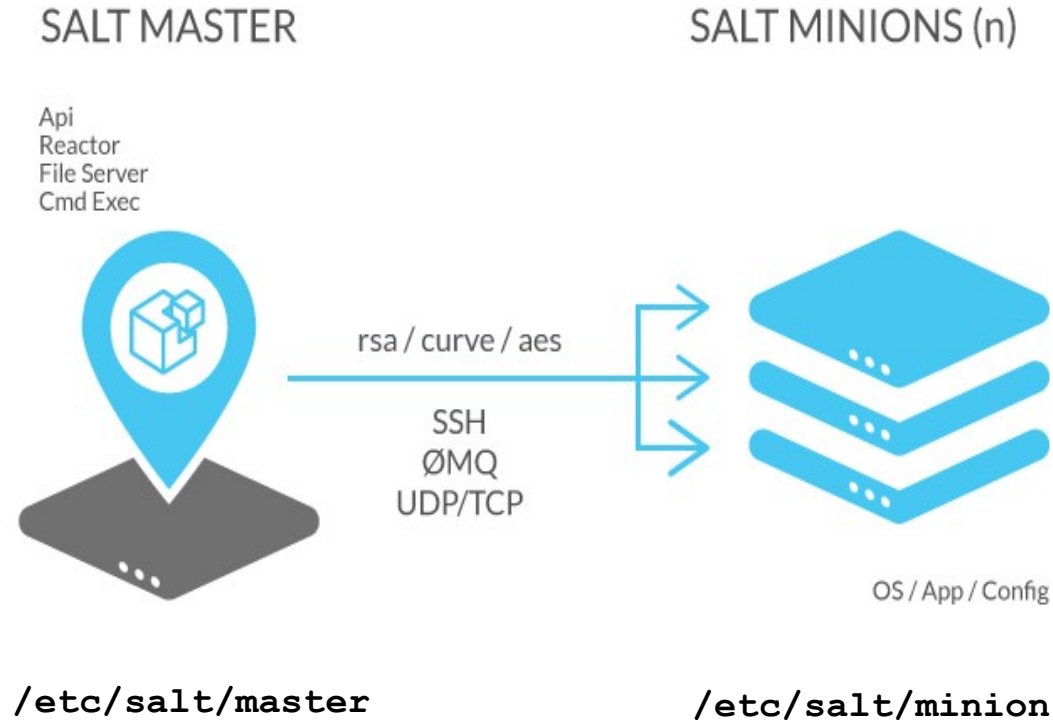
- Additional Defaults

- `/srv/salt`

- `/srv/pillar`

Configuration

Configuration Files



Using Salt

CLI

```
salt \* pkg.install vim
```

Invoke Salt

Specify Target

Call Function

Pass Arguments

Targets

Salt CLI

- Globbing
 - RegEx
 - Lists
 - Grains
 - Subnet/IP
 - Compound
- ```
salt 'web[1-5]' test.ping
salt -E 'web1-(prod|devel)' test.ping
salt -L 'web1,web2,web3' test.ping
salt -G 'os:CentOS' test.ping
salt -S 192.168.40.20 test.ping
salt -S 10.0.0.0/24 test.ping
salt -C '* and not G@kernel:Darwin' test.ping
```

# Salt Executables

## Salt CLI

- salt
- salt-call
- salt-cp
- salt-run
- salt-ssh

```
salt minion user.info daemon
```

```
salt * user.add jborg
```

```
salt minion network.interfaces
```

```
salt-call sys.doc saltutil
```

```
salt-run jobs.active
```

```
salt-cp -G 'os:Arch.amd64' httpd_conf)_tmpl /etc/http/httpd.conf
```

# State Files

SLS files in YAML, Jinja, and JSON format

## YAML state file

---

```
File: /srv/salt/redis/redis.sls:
redis-server:
 pkg.installed:
 - name: redis-server
 service:
 - running
```

## YAML with Jinja state file

---

```
{% if grains['os'] == 'Ubuntu' %}
setup_apache:
 pkg.installed:
 - name: apache2
{% endif %}
```

**Thank you**

**#LinuxCon15**

