

Salting things up in the sysadmin's world

Juan Manuel Santos

Who? (boring part)

- Juan Manuel Santos
- godlike64 on github / gmail / twitter
- godlike on freenode
- Information Systems Engineer / RHCA
- TL - STSE @ Red Hat
- Need beer to survive, HALP!

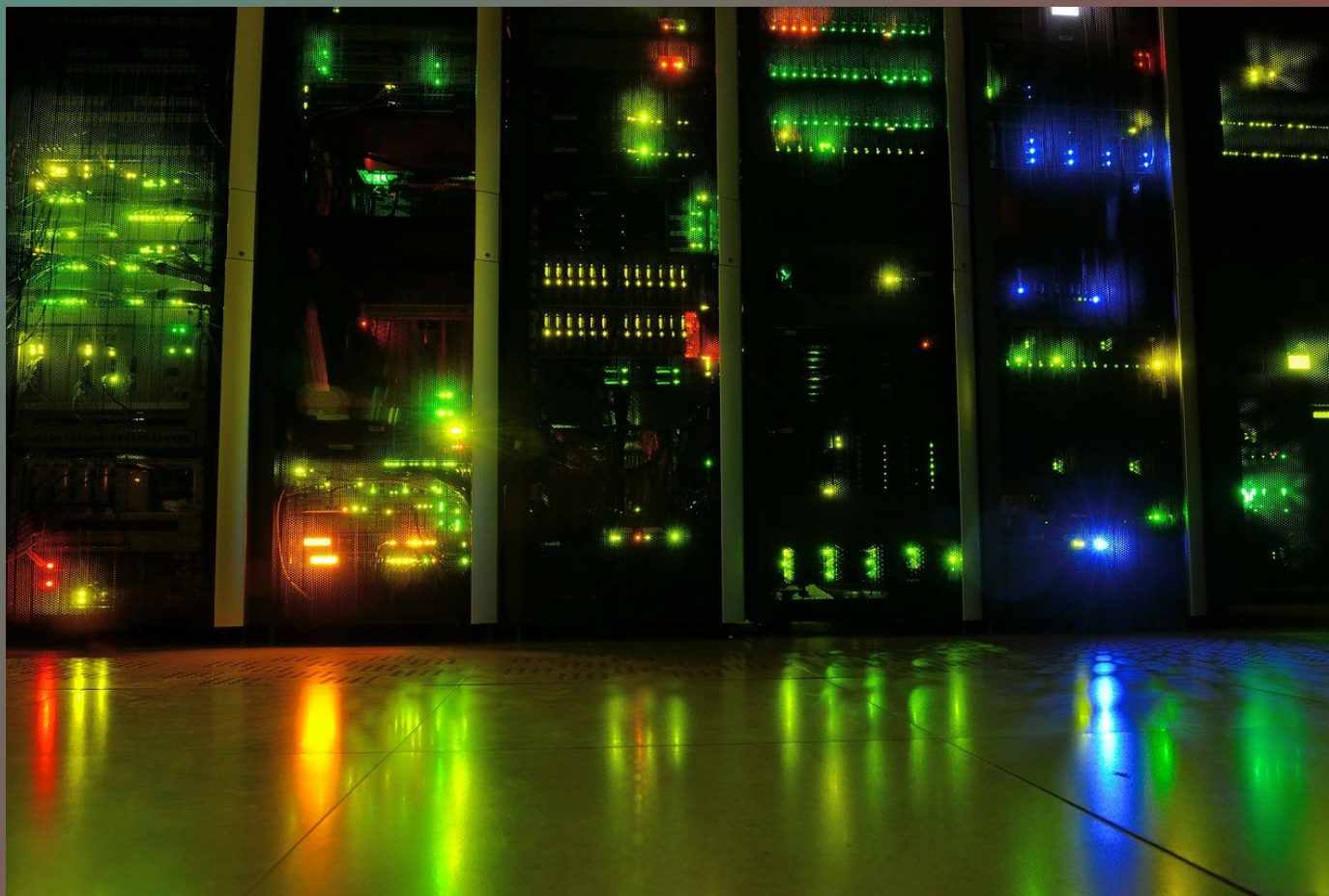
WHY?

2015-07-21





2015-07-21



2015-07-21





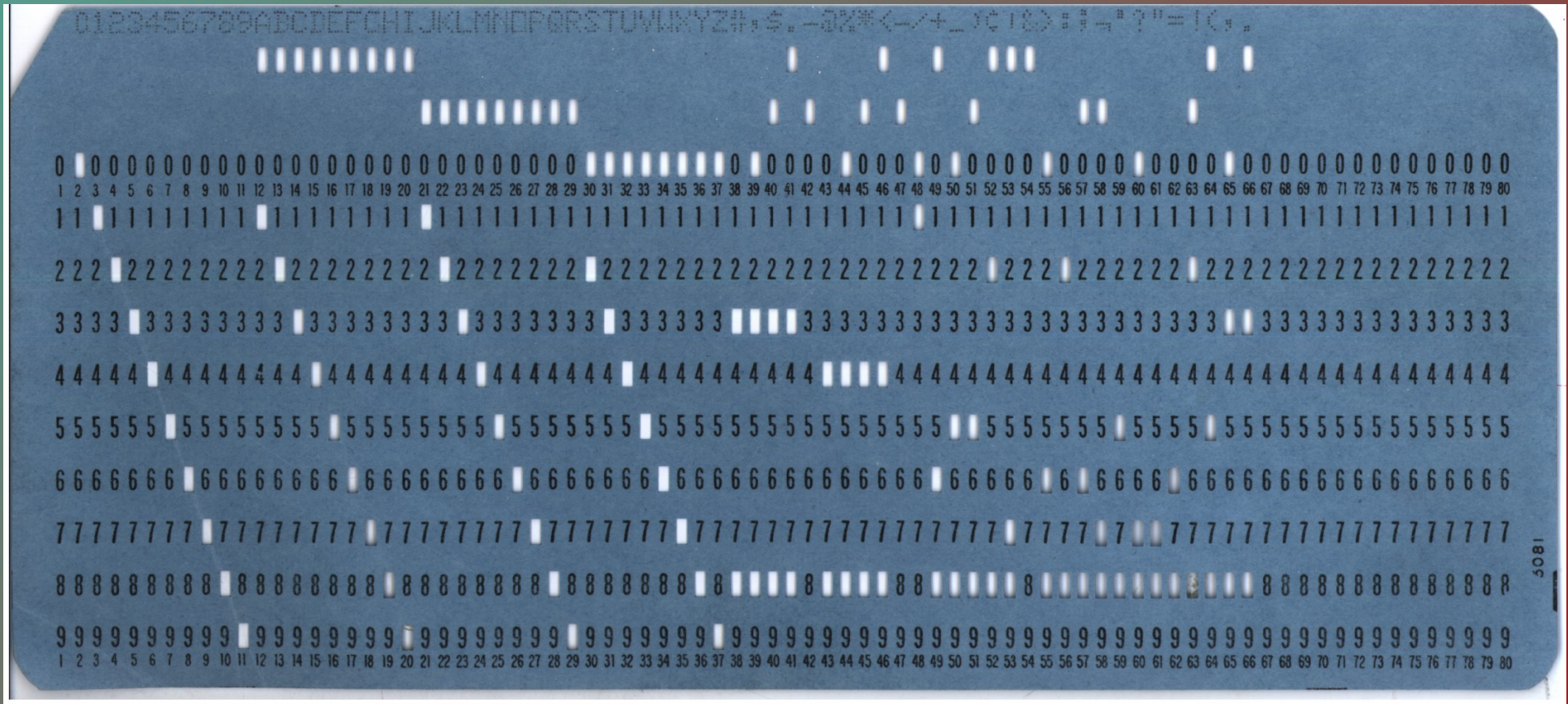
2015-07-21



2015-07-21

WHY?

1)Quantity



2015-07-21



12712414

- SLOC count for Linux kernel 3.19

WHY?

1)Quantity

2)Complexity

Sysadmins are lazy

2015-07-21



How it all began

2015-07-21





2015-07-21



#plzhelp

2015-07-21





2015-07-21

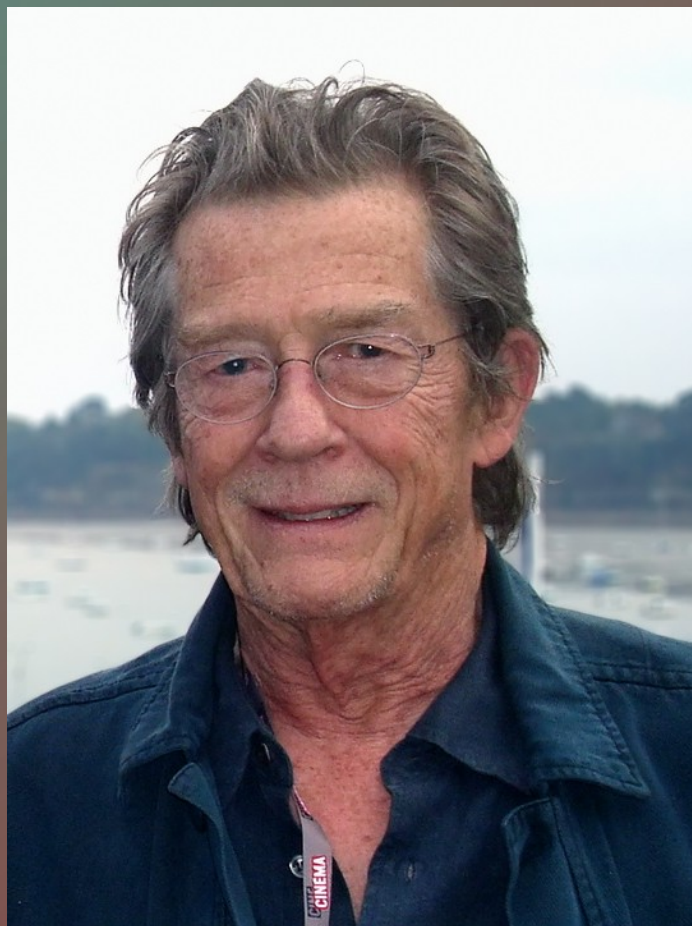


2015-07-21



2015-07-21





2015-07-21





2015-07-21



2015-07-21

?

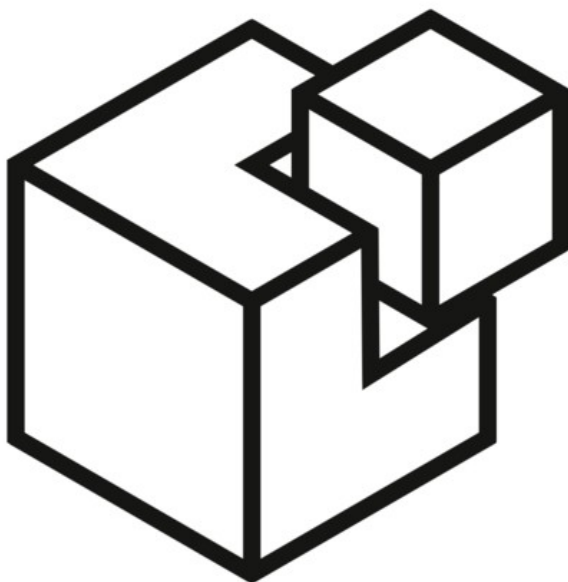
:)

2015-07-21





2015-07-21



SALTSTACK

2015-07-21



Basics



2015-07-21

Basics



Terminology

- Master - Minion
- State file --> State
- Highstate
- Grains / Pillar
- Matching (targeting)
- ... and stuff

Terminology

- Master: the one who rules them all
- Minions:
 - ID
 - [Nodegroup]
 - Grains

Example setup

- Master: gantrithor
- Minions:
 - everybody else

States & Highstates

2015-07-21



States

- Defined by SLS (SaLt State) files
- Reside in the filesystem
- In /etc/salt/master:

file_roots:

<environment>:

- <path>

Top file (top.sls)

- Entry point for environment
- Host matching

<environment>:

<match>:

- match: <match type>
- [state trees]

```
root@gantrithor ~ # cat /srv/salt/base/top.sls
base:
  gentoo:
    - match: nodegroup
    - tmux
    - bash
    - eix
    - makeconf
    - cpuflags
  'salt-master':
    - match: nodegroup
    - nodegroups
root@gantrithor ~ # █
```

States

- Where the magic happens
- Similar syntax to top.sls

<identifier>:

<type>:

- parameters

States

```
root@gantrithor ~ # ls /srv/salt/base/tmux/  
init.sls  tmux.conf  
root@gantrithor ~ # cat /srv/salt/base/tmux/init.sls  
tmux:  
  pkg.installed:  
    - refresh: False  
    - name: app-misc/tmux  
  
/etc/tmux.conf:  
  file.managed:  
    - source: salt://tmux/tmux.conf  
    - user: root  
    - group: root  
    - mode: 644  
root@gantrithor ~ # █
```

States

```
{% if grains['multicore'] == True %}
opts:
  file.managed:
    - name: /etc/portage/make.conf.d/opts
    - source: salt://makeconf/opts
    - template: jinja
    - require:
      - file: make.conf.d

remove-opts:
  file.replace:
    - name: /etc/portage/make.conf
    - pattern: '^[MAKE|EMERGE].*OPTS=.*$'
    - repl: ''
    - flags: ['DOTALL']
    - require:
      - file: source

{% endif %}
```

States

```
root@gantrithor base # cat makeconf/opts
EMERGE_DEFAULT_OPTS="--quiet-build=n
                    --jobs={{ grains['num_cpus'] }}
                    --load-average={{ grains['num_cpus'] }}"
MAKEOPTS="-j{{ grains['num_cpus'] + 1 }}
          -l{{ grains['num_cpus'] }}"
root@gantrithor base # █
```

States

```
source:
  file.blockreplace:
    - name: /etc/portage/make.conf
    - marker_start: "# START salt managed zone -- DO NOT EDIT "
    - marker_end: "# END salt managed zone"
    - content: |
        {% if grains['multicore'] == True %}source /etc/portage/make.conf.d/opts
        {% endif %}source /etc/portage/make.conf.d/use
        source /etc/portage/make.conf.d/mirrors
    - append_if_not_found: True
    - require:
        {% if grains['multicore'] == True %}
        - file: opts
        {% endif %}
        - file: use
        - cmd: mirror-update
```

Highstate

- salt '*' state.highstate
- 1) Minions download top.sls
 - 2) Minions attempt to match
 - 3) If a match is found, modules are downloaded, compiled & executed

Highstate

```
root@gantrithor ~ # salt antioch state.highstate test=True
antioch:
```

```
-----
      ID: tmux
  Function: pkg.installed
     Name: app-misc/tmux
    Result: True
   Comment: Package app-misc/tmux is already installed.
  Started: 11:43:38.588469
 Duration: 1653.323 ms
  Changes:
-----
      ID: /etc/tmux.conf
  Function: file.managed
     Result: True
   Comment: The file /etc/tmux.conf is in the correct state
  Started: 11:43:40.244239
 Duration: 265.436 ms
  Changes:
-----
```


Matching & Nodegroups

2015-07-21



Matching

- ID
- Nodegroup
- Subnet/IP
- Grains
- Compound

Matching

```
root@gantrithor ~ # salt antioch test.ping
antioch:
    True
root@gantrithor ~ # █
```

```
root@gantrithor ~ # salt -L 'antioch or panther' test.ping
panther:
    True
antioch:
    True
root@gantrithor ~ # █
```

Nodegroups

```
root@gantrithor ~ # grep i7 /etc/salt/master.d/nodegroups.conf
i7: 'antioch or panther'
root@gantrithor ~ # █
```

```
root@gantrithor ~ # salt -N i7 test.ping
panther:
    True
antioch:
    True
root@gantrithor ~ # █
```

Nodegroups

```
root@gantrithor ~ # cat /etc/salt/master.d/nodegroups.conf
nodegroups:
  salt-master: gantrithor
  portage-masters: 'gantrithor tanya'
  portage-slaves: 'antioch romina'
  i7: 'antioch or panther'
  gentoo: 'G@os:Gentoo'
root@gantrithor ~ # █
```

Nodegroups

- Flexibility!

nodegroups:

group1: 'L@foo.domain.com,bar.domain.com'

group2: 'G@os:Debian and foo.domain.com'

group3: 'G@os:Debian and N@group1'

Grains & Pillars

2015-07-21



Grains

- Bits & pieces of information
- Generated/loaded on minion startup
- salt [...] grains.ls
- salt [...] grains.items
- 50+ grains shipped
- Moar flexibility: write your own!

Grains

```
root@gantrithor ~ # cat /srv/salt/base/_grains/multicore.py
#!/usr/bin/python

import multiprocessing

def multicore():
    grains = {}
    grains['multicore'] = multiprocessing.cpu_count() > 1
    return grains
root@gantrithor ~ # █
```

Grains

```
root@gantrithor pillar # salt panther grains.get multicore
panther:
  True
root@gantrithor pillar # █
```

Pillar

- Also data
- Defined on the master
- In /etc/salt/master:

pillar_roots:

<environment>:

- <path>

Pillar

- Grains --> data retrieved from minions
- Pillar --> data sent to minions

Pillar

```
root@gantrithor pillar # salt antioch pillar.items
antioch:
-----
root@gantrithor pillar # █
```

Pillar

```
root@gantrithor pillar # pwd
/srv/salt/pillar
root@gantrithor pillar # cat top.sls
base:
  '*':
    - pkg
```

Pillar

```
root@gantrithor pillar # cat pkg/init.sls
pkgs:
  tmux: app-misc/tmux
  tmuxrc: salt://tmux/tmux.conf
root@gantrithor pillar # salt antioch pillar.items
antioch:
  -----
  pkgs:
    -----
    tmux:
      app-misc/tmux
    tmuxrc:
      salt://tmux/tmux.conf
root@gantrithor pillar #
```

Pillar

```
root@gantrithor pillar # cat /srv/salt/base/tmux/init.sls
tmux:
  pkg.installed:
    - refresh: False
    - name: {{ pillar['pkgs']['tmux'] }}

/etc/tmux.conf:
  file.managed:
    - source: {{ pillar['pkgs']['tmuxrc'] }}
    - user: root
    - group: root
    - mode: 644
root@gantrithor pillar # █
```

Docs!

- <http://docs.saltstack.com/>

Thank you!

- Questions anyone?