Saltstack 应用指南

介绍

SaltStack 是一个服务器基础架构集中化管理平台，具备配置管理、远程执行、监控等功能，一般可以理解为简化版的 puppet 和加强版的 func。SaltStack 基于 Python 语言实现，结合轻量级消息队列（ZeroMQ）与 Python 第三方模块（Pyzmq、PyCrypto、Pyjinjia2、python-msgpack 和 PyYAML 等）构建。

通过部署 SaltStack 环境，我们可以在成千上万台服务器上做到批量执行命令，根据不同业务特性进行配置集中化管理、分发文件、采集服务器数据、操作系统基础及软件包管理等，SaltStack 是运维人员提高工作效率、规范业务配置与操作的利器。

1.Saltstack 快速入门

1.1 运行方式

- Local
- Master/Minion
- Salt SSH

1.2 三大功能

- 远程执行
- 配置管理
- 云管理(不成熟)

1.3 支持的操作系统

- Centos
- Redhat
- Debian
- Ubuntu
- FreeBSD
- Solaris
- Fedora
- Gentoo

- Mac OS X
- Archlinux
- Windows
- Suse

1.4 Staltstack   部署

准备两台机器：

    apache-server 192.168.2.125
    server1   192.168.2.200

Doc:

https://docs.saltstack.com/en/latest/topics/installation/fedora.html

EPEL for centos:

https://dl.fedoraproject.org/pub/epel/epel-release-latest-7.noarch.rpm
https://dl.fedoraproject.org/pub/epel/epel-release-latest-6.noarch.rpm
https://dl.fedoraproject.org/pub/epel/epel-release-latest-5.noarch.rpm

---

Install EPEL

```
$ sudo rpm -ivh https://dl.fedoraproject.org/pub/epel/epel-release-latest-6.noarch.rpm
```

Salt Master&&Minion install

```
sudo yum install -y salt-master
sudo yum install -y salt-minion
```

Update config minion

```
[whoami@server1 salt]$ sudo cat /etc/salt/minion|grep 'master: 192'
master: 192.168.2.125

[whoami@apache-server salt]$ sudo cat minion|grep 'master: 192.'
master: 192.168.2.125
```

Salt Master&&Minion start

```
$ sudo /etc/init.d/salt-master start
$ sudo chkconfig --level 3 salt-master on
$ sudo chkconfig --list|grep 3:on|grep salt


$ sudo /etc/init.d/salt-minion start
$ sudo chkconfig --level 3 salt-minion on
$ sudo chkconfig --list|grep 3:on|grep salt
```

在其他机器安装一个客户端 minion

```
$ sudo rpm -ivh https://dl.fedoraproject.org/pub/epel/epel-release-latest-6.noarch.rpm
$ sudo yum install -y salt-minion
$ sudo /etc/init.d/salt-minion start
```

master 监控端口

```
[whoami@apache-server ~]$  netstat -ltl|grep 45
tcp        0        0 *:4505                        *:*              LISTEN
tcp        0        0 *:4506                        *:*              LISTEN
```

配置文件路径

```
[whoami@apache-server ~]$ cd /etc/salt/
[whoami@apache-server salt]$ ls
master   minion   minion.d   minion_id   pki
```

salt-key 认证

发现两台 minion 机器，默认 key 是 hostname，也可以更改为自定义在 minion 文件修改 id 值即可！

```
[whoami@server1 salt]$ sudo cat /etc/salt/minion|grep 'id: minion'
id: minion.server1

[whoami@server1 salt]$ sudo mv minion_id /tmp/
[whoami@server1 salt]$ sudo /etc/init.d/salt-minion restart
Stopping salt-minion daemon:                                    [  OK  ]
Starting salt-minion daemon:                                    [  OK  ]
[whoami@apache-server ~]$ sudo mv /etc/salt/minion_id /tmp
[whoami@apache-server ~]$ sudo /etc/init.d/salt-minion restart
Stopping salt-minion daemon:                                    [  OK  ]
Starting salt-minion daemon:                                    [  OK  ]

#验证
```


认证 **salt-key,**操作/etc/salt/pki/master/minions 文件

```
[whoami@apache-server ~]$ sudo salt-key -a minion.server1
The following keys are going to be accepted:
Unaccepted Keys:
minion.server1
Proceed? [n/Y] Y
Key for minion minion.server1 accepted.
```

```
[whoami@apache-server ~]$ sudo salt-key -a minion.apache-server
The following keys are going to be accepted:
Unaccepted Keys:
minion.apache-server
Proceed? [n/Y] Y
Key for minion minion.apache-server accepted.
```

验证

```
[whoami@apache-server ~]$ sudo salt-key -L
Accepted Keys:
minion.apache-server
minion.server1
Denied Keys:
Unaccepted Keys:
apache-server
server1
Rejected Keys:
```

1.5 快速入门使用

#1.5.1 test 模块

```
[whoami@apache-server ~]$ sudo salt '*' test.ping
minion.apache-server:
    True
minion.server1:
    True
```

```
[whoami@apache-server ~]$ sudo salt 'minion.server1' test.ping
minion.server1:
    True
```

1.5.2 cmd 模块

```
[whoami@apache-server ~]$ sudo salt '*' cmd.run 'df -h'
minion.server1:
    Filesystem        Size  Used Avail Use% Mounted on
    /dev/sda1          20G  6.5G   13G  35% /
    tmpfs             935M   12K  935M   1% /dev/shm
    /dev/sda3          72G  500M   68G   1% /data
minion.apache-server:
    Filesystem        Size  Used Avail Use% Mounted on
    /dev/sda5         3.3G  3.0G  174M  95% /
    tmpfs             935M   16K  935M   1% /dev/shm
    /dev/sda1         190M   32M  149M  18% /boot
    /dev/sda2          24G  205M   23G   1% /data
```

```
[whoami@apache-server ~]$ sudo salt '*' cmd.run 'uptime'
minion.server1:
     19:52:55 up 48 min,  1 user,  load average: 0.00, 0.00, 0.00
minion.apache-server:
     19:36:38 up  5:54,  1 user,  load average: 0.10, 0.04, 0.01
```

1.5.3 不同的机器配置不同的状态文件

#修改 master 配置文件

```
[whoami@apache-server ~]$ sudo vim /etc/salt/master
# Per default, the master will automatically include all config files
# from master.d/*.conf (master.d is a directory in the same directory
# as the main master config file).
default_include: master.d/*.conf

# The address of the interface to bind to:
#interface: 0.0.0.0

# The tcp port used by the publisher:
#publish_port: 450

# The user under which the salt master will run. Salt will update all
# permissions to allow the specified user to run the master. The exception is
# the job cache, which must be deleted if this user is changed. If the
# modified files cause conflicts, set verify_env to False.
#user: root #默认是 root，很多权限需要 root 才能执行


#####         File Server settings         #####
```

```
###########################################
# Salt runs a lightweight file server written in zeromq to deliver files to
# minions. This file server is built into the master daemon and does not
# require a dedicated port.

# The file server works on environments passed to the master, each environment
# can have multiple root directories, the subdirectories in the multiple file
# roots cannot match, otherwise the downloaded files will not be able to be
# reliably ensured. A base environment is required to house the top file.
# Example:
# file_roots:
#     base:
#         - /srv/salt/
#     dev:
#         - /srv/salt/dev/services
#         - /srv/salt/dev/states
#     prod:
#         - /srv/salt/prod/services
#         - /srv/salt/prod/states
#
#file_roots:
#    base:
#        - /srv/salt
file_roots:
    base:
        - /etc/salt/states
    prod:
        - /etc/salt/states/prod


#####         State System settings,状态文件        #####
###########################################
# The state system uses a "top" file to tell the minions what environment to
# use and what modules to use. The state_top file is defined relative to the
# root of the base environment as defined in "File Server settings" below.
state_top: top.sls #默认，不打开也无影响

[whoami@apache-server ~]$ sudo mkdir -p /etc/salt/states/prod
```

重启 master 服务

```
[whoami@apache-server ~]$ sudo /etc/init.d/salt-master restart
Stopping salt-master daemon:                           [  OK   ]
Starting salt-master daemon:                           [  OK   ]
```

salt 日志文件

```
[whoami@apache-server ~]$ cat /var/log/salt/master

[whoami@apache-server ~]$ sudo tail /var/log/salt/minion
```

状态文件，首先需要配置对应关系即 master 配置中的 base 目录下面存储状态文件

```
[whoami@apache-server ~]$ cd /etc/salt/states/
[whoami@apache-server states]$ ls
prod

[whoami@apache-server states]$ pwd
/etc/salt/states

#状态文件写法
#####          State System settings          #####
##########################################
# The state system uses a "top" file to tell the minions what environment to
# use and what modules to use. The state_top file is defined relative to the
# root of the base environment as defined in "File Server settings" below.
state_top: top.sls    #默认，不打开也无影响

[whoami@apache-server states]$ cat /etc/salt/states/top.sls
base:
   'minion.server1':
      - init.pkg

# 初始化模块，管理初始化文件
[whoami@apache-server states]$ sudo mkdir /etc/salt/states/init
[whoami@apache-server states]$ cat init/pkg.sls
pkg.init:
   pkg.installed:
     - names:
        - lrzsz
        - mtr
        - nmap
```

通过状态文件管理主机测试

```
[whoami@apache-server ~]$ sudo salt 'minion.server1' state.sls init.pkg # '*' 表示所有主机安装
minion.server1:
----------
            ID: pkg.init
      Function: pkg.installed
          Name: mtr
        Result: True
       Comment: Package mtr is already installed.
```

```
       Started: 21:54:15.269893
      Duration: 1332.443 ms
       Changes:
----------
            ID: pkg.init
      Function: pkg.installed
          Name: nmap
        Result: True
       Comment: Package nmap is already installed.
       Started: 21:54:16.602503
      Duration: 0.42 ms
       Changes:
----------
            ID: pkg.init
      Function: pkg.installed
          Name: lrzsz
        Result: True
       Comment: Package lrzsz is already installed.
       Started: 21:54:16.602983
      Duration: 0.294 ms
       Changes:

Summary
------------
Succeeded: 3
Failed:      0
------------
Total states run:       3
```

1.5.4 salt 管理配置文件

```
cat /etc/security/limits.conf

# salt 管理配置文件
[whoami@apache-server states]$ sudo cat init/limit.sls
limit-conf-config:
  file.managed:
    - name: /etc/security/limits.conf
    - source: salt://init/files/limits.conf
    - user: root
    - group: root
    - mode: 644
```

```
[whoami@apache-server states]$ sudo mkdir init/files
[whoami@apache-server states]$ sudo cp /etc/security/limits.conf init/files/
[whoami@apache-server states]$ pwd
/etc/salt/states
[whoami@apache-server states]$ ls /etc/salt/states/init/files/
limits.conf
```

验证

```
[whoami@apache-server states]$ tail -2 /etc/security/limits.conf
#@student          -          maxlogins          4
# End of file
[whoami@server1 states]$ tail -2 /etc/security/limits.conf
#@student          -          maxlogins          4
# End of file

[whoami@apache-server states]$ tail -3 /etc/salt/states/init/files/limits.conf
*                  -          nofile             65535
```

执行命令并没有生效
```
[whoami@apache-server states]$ sudo salt 'minion.server1' state.highstate |grep Succeeded
Succeeded: 3
```

需要加入 top.sls,配置文件才会生效
```
[whoami@apache-server states]$ cat top.sls
base:
  'minion.server1':
    - init.pkg
    - init.limit
```

再次测试
```
[whoami@apache-server states]$ sudo salt 'minion.server1' state.highstate
minion.server1:
----------
          ID: pkg.init
    Function: pkg.installed
        Name: mtr
      Result: True
     Comment: Package mtr is already installed.
     Started: 22:18:21.187078
    Duration: 700.226 ms
     Changes:
----------
          ID: pkg.init
    Function: pkg.installed
```

```
               Name: nmap
             Result: True
            Comment: Package nmap is already installed.
            Started: 22:18:21.887479
           Duration: 0.483 ms
            Changes:
----------
                 ID: pkg.init
           Function: pkg.installed
               Name: lrzsz
             Result: True
            Comment: Package lrzsz is already installed.
            Started: 22:18:21.888030
           Duration: 0.323 ms
            Changes:
----------
                 ID: limit-conf-config
           Function: file.managed
               Name: /etc/security/limits.conf
             Result: True
            Comment: File /etc/security/limits.conf updated
            Started: 22:18:21.890602
           Duration: 7.702 ms
            Changes:
                     ----------
                     diff:
                         ---
                         +++
                         @@ -47,4 +47,6 @@
                          #ftp              hard     nproc           0
                          #@student         -        maxlogins       4

                         +*                 -        nofile          65535
                         +
                          # End of file


Summary
------------
Succeeded: 4 (changed=1)
Failed:      0
------------
Total states run:     4

[whoami@server1 ~]$ tail -3 /etc/security/limits.conf
```

```
*                    -        nofile          65535


# End of file
```

1.5.5 salt-key 命令使用

```
[root@apache-server ~]# salt-key
Accepted Keys:
minion.apache-server
minion.server1
Denied Keys:
Unaccepted Keys:
apache-server
server1
Rejected Keys:
```

```
[root@apache-server ~]# salt-key -d server1
The following keys are going to be deleted:
Unaccepted Keys:
server1
Proceed? [N/y] y
Key for minion server1 deleted.
[root@apache-server ~]# salt-key -d apache-server
The following keys are going to be deleted:
Unaccepted Keys:
apache-server
Proceed? [N/y] y
Key for minion apache-server deleted.
[root@apache-server ~]# salt-key
Accepted Keys:
minion.apache-server
minion.server1
Denied Keys:
Unaccepted Keys:
Rejected Keys:
```

```
[root@apache-server ~]#     salt-key –help
Actions:
     -l ARG, --list=ARG    List the public keys. The args "pre", "un", and
                           "unaccepted" will list unaccepted/unsigned keys. "acc"
                           or "accepted" will list accepted/signed keys. "rej" or
                           "rejected" will list rejected keys. "den" or "denied"
                           will list denied keys. Finally, "all" will list all
```
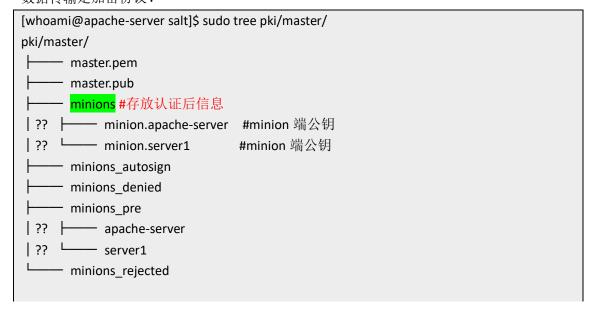
```
                                keys.
    -L, --list-all        List all public keys. (Deprecated: use "--list all")
    -a ACCEPT, --accept=ACCEPT
                                Accept the specified public key (use --include-all to
                                match rejected keys in addition to pending keys).
                                Globs are supported.
    -A, --accept-all      Accept all pending keys
    -r REJECT, --reject=REJECT
                                Reject the specified public key (use --include-all to
                                match accepted keys in addition to pending keys).
                                Globs are supported.
    -R, --reject-all      Reject all pending keys
    --include-all         Include non-pending keys when accepting/rejecting
    -p PRINT, --print=PRINT
                                Print the specified public key
    -P, --print-all       Print all public keys
    -d DELETE, --delete=DELETE
                                Delete the specified key. Globs are supported.
    -D, --delete-all      Delete all keys
    -f FINGER, --finger=FINGER
                                Print the specified key's fingerprint
    -F, --finger-all      Print all keys' fingerprints
```

2.Saltstack 数据系统

2.1 salt 认证流程

数据传输是加密协议！

```
[whoami@apache-server salt]$ sudo tree pki/master/
pki/master/
├──── master.pem
├──── master.pub
├──── minions #存放认证后信息
│ ??  ├──── minion.apache-server   #minion 端公钥
│ ??  └──── minion.server1         #minion 端公钥
├──── minions_autosign
├──── minions_denied
├──── minions_pre
│ ??  ├──── apache-server
│ ??  └──── server1
└──── minions_rejected
```

```
[whoami@server1 pki]$ sudo tree
.
└── minion
    ├── minion_master.pub
    ├── minion.pem          #私钥
    └── minion.pub          #公钥
```

```
[whoami@apache-server salt]$ sudo cat pki/master/minions/minion.server1
-----BEGIN PUBLIC KEY-----
MIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIBCgKCAQEAsg2bZbILphwZKot9wVPG
UUFNixDReAzmLqJXmXyo+zCc/M2X9/Oqed956hvmmqfeGM93qNR63h8n9UUZ9Ir+
Oy2kAPTqqZM4xab4dsYnsi8cOBUIMLCrTOYDLRCHf786Rd6g2EOE8ZaF8vvJ7Woy
I1IrvEOhiE1YGkIyFwxTW7XzQoOMTMxLrpW5aDPD9Mod5bho3bAtrItrT7K2ax5f
dIUI2azpS6in1bwspGNFCsxoai00eExsgQwXd3gZoOUCOm108A3K4vsHiu35a1cT
EDvX4ySLPotzokEci8reHzqtspZyN42Zd5fncRQjim+781Ixy5AuWMObp54M4kwh
hwIDAQAB
-----END PUBLIC KEY-----
```

```
[whoami@apache-server salt]$ sudo cat pki/minion/minion_master.pub
-----BEGIN PUBLIC KEY-----
MIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIBCgKCAQEAOpfcKt17PBNAyfyT42Yu
oxJiKpfkNjtpfXyOlDCbuZjFQtXz63ZEGTCJjr0azzyJB1+AQkc6yflIXZCBnhYj
dXsk3ImjY6Xc0ySXg6C9xmv6JrEFfKoJHcCKKrWrn7XaXfS9OrVBHjgiqrocUMPQ
amtJEnEQIi+7eYjIVHYdSvlo7/MMOLijudcFbMkr7eSLY8Jek8N1k9XVOgOr4pmR
DiFc/Lbf433gkXUUOB+adoNdrmeHtetv7rfkiaYbwKsQscZYFSZkju0fqd4vCJHr
mamUB4IBLdK8rloOnSe9/cwn1wIqJ26/vzNOVWI3j97HBD+ocf1xkeJzA95nnhCz
QQIDAQAB
-----END PUBLIC KEY-----
```

```
[root@server1 minion]# cat minion.pub
-----BEGIN PUBLIC KEY-----
MIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIBCgKCAQEAsg2bZbILphwZKot9wVPG
UUFNixDReAzmLqJXmXyo+zCc/M2X9/Oqed956hvmmqfeGM93qNR63h8n9UUZ9Ir+
Oy2kAPTqqZM4xab4dsYnsi8cOBUIMLCrTOYDLRCHf786Rd6g2EOE8ZaF8vvJ7Woy
I1IrvEOhiE1YGkIyFwxTW7XzQoOMTMxLrpW5aDPD9Mod5bho3bAtrItrT7K2ax5f
dIUI2azpS6in1bwspGNFCsxoai00eExsgQwXd3gZoOUCOm108A3K4vsHiu35a1cT
EDvX4ySLPotzokEci8reHzqtspZyN42Zd5fncRQjim+781Ixy5AuWMObp54M4kwh
hwIDAQAB
-----END PUBLIC KEY-----
```

2.2 salt 长连接,断开自动连接

```
[whoami@apache-server salt]$ sudo lsof -i:4505
COMMAND    PID USER   FD   TYPE DEVICE SIZE/OFF NODE NAME
salt-mini 3790 root   24u  IPv4  35929      0t0  TCP apache-server:44314->apache-server:4505 (ESTABLISHED)
salt-mast 5731 root   12u  IPv4  34640      0t0  TCP *:4505 (LISTEN)
salt-mast 5731 root   14u  IPv4  35930      0t0  TCP apache-server:4505->apache-server:44314 (ESTABLISHED)
salt-mast 5731 root   15u  IPv4  36357      0t0  TCP apache-server:4505->server1:43927 (ESTABLISHED)
```

2.3 数据系统

Globbing and regex

匹配 minions 通过通配符和正则表达式

Grains

http://docs.saltstack.cn/zh_CN/latest/topics/targeting/grains.html#targeting-grains

用来匹配 minion 的 grains，是指那些关于 minion 主机的静态信息，比如 OS，软件版本，虚拟化，CPU，内存等等。

```
[root@apache-server ~]# salt 'minion.server1' grains.items

minion.server1:

    ----------

    SSDs:

    biosreleasedate:

        07/31/2013

    biosversion:

        6.00

    cpu_flags:

        - fpu

        - vme

        - de

        - pse

        - tsc
```

- msr

- pae

- mce

- cx8

- apic

- sep

- mtrr

- pge

- mca

- cmov

- pat

- pse36

- clflush

- dts

- mmx

- fxsr

- sse

- sse2

- ss

- syscall

- nx

- pdpe1gb

- rdtscp

- lm

- constant_tsc

- up

- arch_perfmon

- pebs

- bts

- xtopology

- tsc_reliable

- nonstop_tsc

- aperfmperf

- unfair_spinlock

- pni

- pclmulqdq

- ssse3

- fma

- cx16

- pcid

- sse4_1

- sse4_2

- x2apic

- movbe

- popcnt

- aes

- xsave

- avx

- f16c

- rdrand

- hypervisor

- lahf_lm

- ida

- arat

- epb

- xsaveopt

- pln

- pts

- dts

- fsgsbase

- smep

cpu_model:

  Intel(R) Core(TM) i7-4710HQ CPU @ 2.50GHz

cpuarch:

  x86_64

domain:

```
fqdn:

    server1

fqdn_ip4:

    - 192.168.2.200

fqdn_ip6:

gpus:

    |_

      ----------

        model:

            SVGA II Adapter

        vendor:

            unknown

host:

    server1

hwaddr_interfaces:

    ----------

    eth0:

        00:0c:29:f8:24:c9

    lo:

        00:00:00:00:00:00

id:

    minion.server1
```

```
init:

    upstart

ip4_interfaces:

    ----------

    eth0:

        - 192.168.2.200

    lo:

        - 127.0.0.1

ip6_interfaces:

    ----------

    eth0:

        - fe80::20c:29ff:fef8:24c9

    lo:

        - ::1

ip_interfaces:

    ----------

    eth0:

        - 192.168.2.200

        - fe80::20c:29ff:fef8:24c9

    lo:

        - 127.0.0.1

        - ::1
```

```
ipv4:

    - 127.0.0.1

    - 192.168.2.200

ipv6:

    - ::1

    - fe80::20c:29ff:fef8:24c9

kernel:

    Linux

kernelrelease:

    2.6.32-573.el6.x86_64

locale_info:

    ----------

    defaultencoding:

        UTF8

    defaultlanguage:

        en_US

    detectedencoding:

        UTF-8

localhost:

    server1

machine_id:

    15239d6be81a3d34619ee9c10000000d
```

manufacturer:

    VMware, Inc.

master:

    192.168.2.125

mdadm:

mem_total:

    1869

nodename:

    server1

num_cpus:

    1

num_gpus:

    1

os:

    RedHat

os_family:

    RedHat

osarch:

    x86_64

oscodename:

    Santiago

osfinger:

```
        Red Hat Enterprise Linux Server-6

osfullname:

        Red Hat Enterprise Linux Server

osmajorrelease:

        6

osrelease:

        6.7

osrelease_info:

        - 6

        - 7

path:

        /sbin:/usr/sbin:/bin:/usr/bin

productname:

        VMware Virtual Platform

ps:

        ps -efH

pythonexecutable:

        /usr/bin/python2.6

pythonpath:

        - /usr/bin

        - /usr/lib64/python26.zip

        - /usr/lib64/python2.6
```

```
        - /usr/lib64/python2.6/plat-linux2

        - /usr/lib64/python2.6/lib-tk

        - /usr/lib64/python2.6/lib-old

        - /usr/lib64/python2.6/lib-dynload

        - /usr/lib64/python2.6/site-packages

        - /usr/lib64/python2.6/site-packages/gtk-2.0

        - /usr/lib/python2.6/site-packages

        - /usr/lib/python2.6/site-packages/setuptools-0.6c11-py2.6.egg-info

    pythonversion:

        - 2

        - 6

        - 6

        - final

        - 0

    saltpath:

        /usr/lib/python2.6/site-packages/salt

    saltversion:

        2015.5.5

    saltversioninfo:

        - 2015

        - 5

        - 5
```

```
          - 0

      selinux:

          ----------

          enabled:

              False

          enforced:

              Disabled

      serialnumber:

          VMware-56 4d 33 68 27 0f 71 17-a8 48 4b e4 89 f8 24 c9

      server_id:

          691254496

      shell:

          /bin/sh

      virtual:

          VMware

      zmqversion:

          3.2.5
[root@apache-server ~]# salt 'minion.server1' grains.ls

minion.server1:

    - SSDs

    - biosreleasedate

    - biosversion
```

- cpu_flags

- cpu_model

- cpuarch

- domain

- fqdn

- fqdn_ip4

- fqdn_ip6

- gpus

- host

- hwaddr_interfaces

- id

- init

- ip4_interfaces

- ip6_interfaces

- ip_interfaces

- ipv4

- ipv6

- kernel

- kernelrelease

- locale_info

- localhost

- machine_id

- manufacturer

- master

- mdadm

- mem_total

- nodename

- num_cpus

- num_gpus

- os

- os_family

- osarch

- oscodename

- osfinger

- osfullname

- osmajorrelease

- osrelease

- osrelease_info

- path

- productname

- ps

- pythonexecutable

- pythonpath

- pythonversion

- saltpath

- saltversion

- saltversioninfo

- selinux

- serialnumber

- server_id

- shell

- virtual

- zmqversion

```
[root@apache-server ~]# salt 'minion.server1' grains.item os
minion.server1:
    ----------
    os:
        RedHat
[root@apache-server ~]# salt 'minion.server1' grains.item selinux
minion.server1:
    ----------
    selinux:
        ----------
        enabled:
            False
        enforced:
            Disabled
```

```
[root@apache-server ~]# salt 'minion.server1' grains.get os
minion.server1:
    RedHat
[root@apache-server ~]# salt 'minion.server1' grains.item os
minion.server1:
    ----------
    os:
        RedHat
```

grains 定位

```
[root@apache-server ~]# salt -G 'os:CentOS'  test.ping
minion.apache-server:
    True
[root@apache-server ~]# salt -G 'os:redhat'  test.ping
minion.server1:
    True
```

```
[root@apache-server ~]# salt -G 'fqdn:server1'  test.ping
minion.server1:
    True
[root@apache-server ~]# salt 'minion.server1' grains.get fqdn
minion.server1:
    server1
```

自定义 grains 属性

[root@server1 ~]# tail -3 /etc/salt/minion

grains:

  roles: ambari

  env: prod

[root@server1 ~]# /etc/init.d/salt-minion restart

Stopping salt-minion daemon: [OK]

Starting salt-minion daemon: [OK]

```
[root@apache-server ~]# salt -G 'env:prod'  test.ping
minion.server1:
    True
```

```
[root@apache-server ~]# salt -G 'roles:ambari'  test.ping
minion.server1:
    True
```

[root@server1 ~]# cat /etc/salt/grains

cloud: openstack

[root@server1 ~]# /etc/init.d/salt-minion restart

Stopping salt-minion daemon: [OK   ]

Starting salt-minion daemon: [OK   ]

```
[root@apache-server ~]# salt -G 'cloud:openstack'  test.ping
minion.server1:
    True

[root@apache-server ~]# salt -G 'cloud:openstack'  service.restart openstack
```

刷新 grains

```
[root@server1 ~]# cat /etc/salt/grains
cloud: openstack
test: salt
```

```
[root@apache-server ~]# salt -G 'test:salt'  test.ping
No minions matched the target. No command was sent, no jid was assigned.
ERROR: No return received
[root@apache-server ~]# salt '*' saltutil.sync_grains
minion.apache-server:
minion.server1:
[root@apache-server ~]# salt -G 'test:salt'  test.ping
minion.server1:
    True
```

top.sls 中 grains 的写法

```
[root@apache-server ~]# tail -3 /etc/salt/states/top.sls
   'roles:nginx':
     - match: grain
     - init.pkg
```

Pillar

通过用户定义的变量匹配 minion 主机. 版本 0.9.8+支持

```
[root@apache-server ~]# salt '*' pillar.items

# 配置 pillar
[root@apache-server ~]# vim /etc/salt/master
pillar_roots:
   base:
     - /etc/salt/pillar

[root@apache-server ~]# mkdir /etc/salt/pillar
```

```
[root@apache-server ~]# cd /etc/salt/pillar/
[root@apache-server pillar]# cat top.sls
base:
  '*':
    - init.rsyslog
[root@apache-server pillar]# mkdir init
[root@apache-server pillar]# cd init/
[root@apache-server init]# pwd
/etc/salt/pillar/init

[root@apache-server init]# cat rsyslog.sls
{% if grains['os'] == 'CentOS' %}
syslog: rsyslog
{% elif grains['os'] == 'RedHat' %}
syslog: syslog
{% endif %}
[root@apache-server init]# /etc/init.d/salt-master restart
Stopping salt-master daemon:                              [  OK  ]
Starting salt-master daemon:                              [  OK  ]
```

刷新 pillar
```
[root@apache-server init]# salt '*' saltutil.refresh_pillar
minion.server1:
    True
minion.apache-server:
    True
```

测试

#开启 pillar 默认选项：
MASTER CONFIG IN PILLAR
```
[root@apache-server init]# vim /etc/salt/master
# The pillar_opts option adds the master configuration file data to a dict in
# the pillar called "master". This is used to set simple configurations in the
# master config file that can then be used on minions.
```

```
pillar_opts: True
[root@apache-server init]# salt '*' pillar.items|wc -l
710

# ls 命令，在 New in version 2015.8.0.版本才拥有
[root@apache-server init]# salt '*' pillar.ls
minion.server1:
    'pillar.ls' is not available.
minion.apache-server:
    'pillar.ls' is not available.
ERROR: Minions returned with non-zero exit code
[root@apache-server init]# salt --version
salt 2015.5.5 (Lithium)
```

3.Saltstack 远程执行

http://docs.saltstack.cn/zh_CN/latest/

3.1 目标

Targeting 指明那些将执行命令或管理服务器配置的 minion 主机。接下来的链接提供了关于如何指定和匹配目标 minion 的更多信息。


```
Target Options:
    Target Selection Options

    -E, --pcre          Instead of using shell globs to evaluate the target
                        servers, use pcre regular expressions
    -L, --list          Instead of using shell globs to evaluate the target
                        servers, take a comma or space delimited list of
                        servers.
    -G, --grain         Instead of using shell globs to evaluate the target
                        use a grain value to identify targets, the syntax for
                        the target is the grain key followed by a
                        globexpression: "os:Arch*"
    --grain-pcre        Instead of using shell globs to evaluate the target
```

use a grain value to identify targets, the syntax for
the target is the grain key followed by a pcre regular
expression: "os:Arch.*"

-N, --nodegroup   Instead of using shell globs to evaluate the target
use one of the predefined nodegroups to identify a
list of targets.

-R, --range   Instead of using shell globs to evaluate the target
use a range expression to identify targets. Range
expressions look like %cluster

-C, --compound   The compound target option allows for multiple target
types to be evaluated, allowing for greater
granularity in target matching. The compound target is
space delimited, targets other than globs are preceded
with an identifier matching the specific targets
argument type: salt 'G@os:RedHat and webser* or
E@database.*'

-I, --pillar   Instead of using shell globs to evaluate the target
use a pillar value to identify targets, the syntax for
the target is the pillar key followed by a glob
expression: "role:production*"

-J, --pillar-pcre   Instead of using shell globs to evaluate the target
use a pillar value to identify targets, the syntax for
the target is the pillar key followed by a pcre
regular expression: "role:prod.*"

-S, --ipcidr   Match based on Subnet (CIDR notation) or IPv4 address.

Match all minions:

```bash
salt '*' test.ping
```

Match all minions in the example.net domain or any of the example domains:

```bash
salt '*.example.net' test.ping
salt '*.example.*' test.ping
```

Match all the `webN` minions in the example.net domain (`web1.example.net`, `web2.example.net` ... `webN.example.net`):

```bash
salt 'web?.example.net' test.ping
```

Match the `web1` through `web5` minions:

```bash
salt 'web[1-5]' test.ping
```

Match the `web1` and `web3` minions:

```bash
salt 'web[1,3]' test.ping
```

Match the `web-x`, `web-y`, and `web-z` minions:

```bash
salt 'web-[x-z]' test.ping
```

> **注解**
>
> For additional targeting methods please review the *compound matchers* documentation.

```
[root@apache-server ~]# salt 'minio?.server1' test.ping
minion.server1:
    True
```

正则表达式

4.1.2. 正则表达式

Minions can be matched using Perl-compatible **regular expressions** (which is globbing on steroids and a ton of caffeine).

Match both `web1-prod` and `web1-devel` minions:

```bash
salt -E 'web1-(prod|devel)' test.ping
```

When using regular expressions in a State's *top file*, you must specify the matcher as the first option. The following example executes the contents of `webserver.sls` on the above-mentioned minions.

```yaml
base:
  'web1-(prod|devel)':
  - match: pcre
  - webserver
```

4.1.3. LISTS

At the most basic level, you can specify a flat list of minion IDs:

```bash
salt -L 'web1,web2,web3' test.ping
```

```
[root@apache-server ~]#  salt -E '(minion|minion-).server1' test.ping
minion.server1:
    True
```

配置文件正则匹配

```
[root@apache-server ~]# cat /etc/salt/states/top.sls
base:
   '(minion.apache-|minion.)server?':
     - init.pkg
     - init.limit
     - match: pcre

# 测试
[root@apache-server ~]# salt '*' state.highstate
minion.server1:
----------
          ID: pkg.init
    Function: pkg.installed
        Name: mtr
      Result: True
     Comment: Package mtr is already installed.
     Started: 04:47:42.751148
    Duration: 756.073 ms
     Changes:
----------
```

```
              ID: pkg.init
        Function: pkg.installed
            Name: nmap
          Result: True
         Comment: Package nmap is already installed.
         Started: 04:47:43.507518
        Duration: 0.727 ms
         Changes:
----------
              ID: pkg.init
        Function: pkg.installed
            Name: lrzsz
          Result: True
         Comment: Package lrzsz is already installed.
         Started: 04:47:43.508364
        Duration: 0.537 ms
         Changes:
----------
              ID: limit-conf-config
        Function: file.managed
            Name: /etc/security/limits.conf
          Result: True
         Comment: File /etc/security/limits.conf is in the correct state
         Started: 04:47:43.511970
        Duration: 2.745 ms
         Changes:

Summary
------------
Succeeded: 4
Failed:      0
------------
Total states run:      4
minion.apache-server:
----------
              ID: pkg.init
        Function: pkg.installed
            Name: mtr
          Result: True
         Comment: Package mtr is already installed.
         Started: 04:31:26.303740
        Duration: 778.925 ms
         Changes:
----------
```

```
          ID: pkg.init
    Function: pkg.installed
        Name: nmap
      Result: True
     Comment: Package nmap is already installed.
     Started: 04:31:27.082790
    Duration: 0.34 ms
     Changes:
----------
          ID: pkg.init
    Function: pkg.installed
        Name: lrzsz
      Result: True
     Comment: Package lrzsz is already installed.
     Started: 04:31:27.083192
    Duration: 0.275 ms
     Changes:
----------
          ID: limit-conf-config
    Function: file.managed
        Name: /etc/security/limits.conf
      Result: True
     Comment: File /etc/security/limits.conf updated
     Started: 04:31:27.085021
    Duration: 6.269 ms
     Changes:
              ----------
              diff:
                  ---
                  +++
                  @@ -47,4 +47,6 @@
                   #ftp            hard    nproc           0
                   #@student        -        maxlogins       4

                  +*               -        nofile          65535
                  +
                   # End of file

Summary
-----------
Succeeded: 4 (changed=1)
Failed:      0
-----------
Total states run:      4
```

-S command

3.2 模块

Salt 模块是远程执行的基础。它提供了一系列的功能，比如安装包，重启一个服务，运行名称命令，传输文件等等。

列表所有模块

```
[root@apache-server ~]#   salt -d|wc -l
17423
```

远程执行模块

```
[root@apache-server ~]# salt '*' disk.usage
minion.server1:
    ----------
    /:
        ----------
        1K-blocks:
            20511356
        available:
            12742716
        capacity:
            35%
        filesystem:
            /dev/sda1
        used:
            6720064
    /data:
        ----------
        1K-blocks:
            75212936
        available:
            70874216
        capacity:
```

```
                1%
        filesystem:
                /dev/sda3
        used:
                511472
    /dev/shm:
        ----------
        1K-blocks:
                957296
        available:
                957284
        capacity:
                1%
        filesystem:
                tmpfs
        used:
                12
```

#cmd.run

```
[root@apache-server ~]# salt '*' cmd.run 'df -h'
minion.server1:
    Filesystem      Size  Used Avail Use% Mounted on
    /dev/sda1        20G  6.5G   13G  35% /
    tmpfs           935M   12K  935M   1% /dev/shm
    /dev/sda3        72G  500M   68G   1% /data
minion.apache-server:
    Filesystem      Size  Used Avail Use% Mounted on
    /dev/sda5       3.3G  3.0G  171M  95% /
    tmpfs           935M   16K  935M   1% /dev/shm
    /dev/sda1       190M   32M  149M  18% /boot
    /dev/sda2        24G  205M   23G   1% /data
```

```
[root@apache-server ~]# salt '*' hosts.get_ip apache-server
minion.server1:
    192.168.2.125
minion.apache-server:
    192.168.2.125
```

获取启动那些服务
[root@apache-server ~]# salt '*' service.get_all|wc -l
148

```
[root@apache-server ~]# salt '*' service.status rsyslog
minion.server1:
    True
minion.apache-server:
    True
```

file 模块

```
[root@apache-server ~]# salt '*' file.check_hash /etc/hosts md5:127
minion.server1:
    False
minion.apache-server:
    False
```

```
[root@apache-server ~]# salt '*' file.file_exists /etc/passwd
minion.server1:
    True
minion.apache-server:
    True
```

salt-cp 命令

```
[root@apache-server ~]# salt-cp '*' /etc/hosts /tmp/hosts
{'minion.apache-server': {'/tmp/hosts': True},
 'minion.server1': {'/tmp/hosts': True}}
```

```
[root@apache-server ~]# ls /tmp/hosts
/tmp/hosts


[root@apache-server ~]# salt '*' cmd.run 'ls /tmp/hosts'
minion.server1:
    /tmp/hosts
minion.apache-server:
    /tmp/hosts
```

3.3 返回

Salt 返回接收器（returner）允许把 minion 的响应保存在各种数据存储或不同的位置，甚至把响应内容显示在命令行。Returner 可以用来扩展 Salt，和新的，定制的接口和支持新的数据库进行通信。

http://docs.saltstack.cn/zh_CN/latest/ref/returners/all/index.html

```
[root@apache-server ~]#  salt 'minion.server1' cmd.run 'uptime'
minion.server1:
    04:17:55 up  9:13,  1 user,  load average: 0.00, 0.00, 0.00
```

 # salt.returners.mysql 模块

http://docs.saltstack.cn/zh_CN/latest/ref/returners/all/salt.returners.mysql.html#module-salt.returners.mysql

```
[root@apache-server ~]# salt "*" service.restart mysqld
minion.apache-server:
    True
minion.server1:
True

#mysql 端安装
[root@apache-server ~]# yum install MySQL-python
```

设置 salt 返回结果写入 mysql 中：

1、创建表结构

```
CREATE DATABASE  `salt`

  DEFAULT CHARACTER SET utf8

  DEFAULT COLLATE utf8_general_ci;


USE `salt`;


--

-- Table structure for table `jids`

--


DROP TABLE IF EXISTS `jids`;

CREATE TABLE `jids` (

  `jid` varchar(255) NOT NULL,

  `load` mediumtext NOT NULL,

  UNIQUE KEY `jid` (`jid`)

) ENGINE=InnoDB DEFAULT CHARSET=utf8;


--
```

```
-- Table structure for table `salt_returns`

--


DROP TABLE IF EXISTS `salt_returns`;

CREATE TABLE `salt_returns` (

  `fun` varchar(50) NOT NULL,

  `jid` varchar(255) NOT NULL,

  `return` mediumtext NOT NULL,

  `id` varchar(255) NOT NULL,

  `success` varchar(10) NOT NULL,

  `full_ret` mediumtext NOT NULL,

  `alter_time` TIMESTAMP DEFAULT CURRENT_TIMESTAMP,

  KEY `id` (`id`),

  KEY `jid` (`jid`),

  KEY `fun` (`fun`)

) ENGINE=InnoDB DEFAULT CHARSET=utf8;


--

-- Table structure for table `salt_events`

--


DROP TABLE IF EXISTS `salt_events`;

CREATE TABLE `salt_events` (

`id` BIGINT NOT NULL AUTO_INCREMENT,

`tag` varchar(255) NOT NULL,

`data` varchar(1024) NOT NULL,

`alter_time` TIMESTAMP DEFAULT CURRENT_TIMESTAMP,

PRIMARY KEY (`id`),

KEY `tag` (`tag`)

) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

```
GRANT ALL PRIVILEGES ON salt.* TO 'salt'@'%'IDENTIFIED BY 'salt' WITH GRANT
OPTION;

GRANT ALL PRIVILEGES ON salt.* TO 'salt'@'localhost'IDENTIFIED BY 'salt' WIT
H GRANT OPTION;

GRANT ALL PRIVILEGES ON salt.* TO 'salt'@'apache-server'IDENTIFIED BY 'salt'
 WITH GRANT OPTION;

FLUSH PRIVILEGES ;
```

2、修改 salt master 配置文件

```
[root@apache-server ~]# tail -9 /etc/salt/master

#####      Returner settings        ######

############################################

# Which returner(s) will be used for minion's result:

#return: mysql

mysql.host: '192.168.2.125'

mysql.user: 'salt'

mysql.pass: 'salt'

mysql.db: 'salt'

mysql.port: 3306

master_job_cache: mysql
```

3、重启 master

```
[root@apache-server ~]# /etc/init.d/salt-master restart

Stopping salt-master daemon:                         [  OK  ]

Starting salt-master daemon:                         [  OK  ]
```

4、测试

```
[root@apache-server ~]# salt '*' test.ping

minion.server1:
```

```
    True

minion.apache-server:

    True

[root@apache-server ~]# salt '*' test.ping --return mysql

minion.server1:

    True

minion.apache-server:

    True
```

#返回结果

4.Saltstack 配置管理

SLS（YAML，Jinja）


```
/etc/resolv.conf:
  file.managed:
    - source: salt://zabbix/files/resolv.conf
    - user: root
    - group: root
    - mode: 755
```

```
zabbix-agent:
  pkg.installed:
    - name: zabbix20-agent
  file.managed:
    - name: /etc/zabbix_agentd.conf
    - source: salt://zabbix/files/zabbix_agentd.conf
    - template: jinja
    - defaults:
      Server: {{ pillar['zabbix-agent']['Zabbix_Server'] }}
    - require:
      - pkg: zabbix-agent
  service.running:
    - enable: True
    - watch:
      - pkg: zabbix-agent
      - file: zabbix-agent
zabbix_agentd_conf-symlink:
  file.symlink:
    - name: /etc/zabbix/zabbix_agentd.conf
    - target: /etc/zabbix_agentd.conf
    - require_in:
      - service: zabbix-agent
    - require:
      - pkg: zabbix-agent
      - file: zabbix-agent
zabbix_agentd.conf.d:
```

#案例

```
[root@apache-server ~]# cd /etc/salt/states/init/

[root@apache-server init]# pwd

/etc/salt/states/init

[root@apache-server init]# touch zabbix_agent.sls

[root@apache-server init]# cat zabbix_agent.sls

zabbix_agent:

  pkg.installed:

    - name: zabbix22-agent


  file.managed:

    - name: /etc/zabbix_agentd.conf

    - source: salt://init/files/zabbix_agentd.conf

    - user: root
```

```
    - group: root

    - mode: 644


  service.running:

    - name: zabbix-agentd

    - enable: True

    - reload: True
```

[root@apache-server files]# pwd

/etc/salt/states/init/files

[root@apache-server files]# cp /etc/zabbix_agentd.conf .

master zabbix ip address...

Server=192.168.2.125

[root@apache-server states]# pwd

/etc/salt/states

[root@apache-server states]# cat top.sls

base:

  '(minion.apache-|minion.)server?':

    - init.pkg

    - init.limit

    - match: pcre

    - init.zabbix_agent

'roles:nginx':

- match: grain

- init.pkg

+++每次修改配置之前,先测试一下

[root@apache-server states]# salt '*' state.highstate test=True

执行修改

```
[root@apache-server states]# salt '*' state.highstate
```

```
[root@apache-server states]# service zabbix-agentd status
zabbix_agentd (pid  3620) is running...
```

```
[root@server1 ~]# service zabbix-agentd status
zabbix_agentd (pid  2880) is running...
```

配置文件变动自动 reload

```
[root@apache-server init]# cat zabbix_agent.sls

zabbix_agent:

  pkg.installed:

    - name: zabbix22-agent


  file.managed:

    - name: /etc/zabbix_agentd.conf

    - source: salt://init/files/zabbix_agentd.conf

    - user: root

    - group: root

    - mode: 644


  service.running:

    - name: zabbix-agentd

    - enable: True

    - reload: True

    - watch:

      - file: zabbix_agent
```

测试成功后，执行修改操作，验证结果

```
[root@apache-server states]# vim init/files/zabbix_agentd.conf  #修改内容

[root@apache-server states]# cat init/files/zabbix_agentd.conf|grep hehe

# hehe file change
```

```
Summary
-----------
Succeeded: 7 (unchanged=2, changed=1)
Failed:    0
-----------
Total states run:     7
[root@apache-server states]# salt '*' state.highstate test=True
```

```
[root@apache-server states]# salt '*' state.highstate
```

```
Summary
-----------
Succeeded: 6 (changed=2)
Failed:    1
-----------
Total states run:     7
```

重载失败，由于此软件不支持重载...

```
[root@apache-server states]# /etc/init.d/zabbix-agentd reload
Service zabbix-agentd does not support the reload action:   [FAILED]
```

修改 zabbix_agent 代码支持 reload

```
[root@server1 ~]# vim /etc/init.d/zabbix-agentd

    reload)

        # action $"Service ${0##*/} does not support the reload action: " /bin/false

        restart

        # exit 3

        ;;

[root@apache-server states]# vim init/files/zabbix_agentd.conf

[root@apache-server states]# salt '*' state.highstate test=True

[root@apache-server states]# salt '*' state.highstate
```

```
    Function: file.managed
        Name: /etc/zabbix_agentd.conf
      Result: True
     Comment: File /etc/zabbix_agentd.conf updated
     Started: 07:05:19.442397
    Duration: 11.189 ms
     Changes:
              ----------
              diff:
                  ---
                  +++
                  @@ -5,7 +5,7 @@

                   ### Option: PidFile
                   #       Name of PID file.
                  -#  hehe file change
                  +#  delete file hah
                   # Mandatory: no
                   # Default:
                   PidFile=/var/run/zabbix/zabbix_agentd.pid
----------
          ID: zabbix_agent
    Function: service.running
        Name: zabbix-agentd
      Result: True
     Comment: Service reloaded
     Started: 07:05:19.486952
    Duration: 162.108 ms
     Changes:
              ----------
              zabbix-agentd:
```

```
Summary
----------
Succeeded: 7 (changed=2)
Failed:    0
----------
Total states run:      7
```

注意：可以不加 reload 参数，仅仅加入 watch 参数，如果文件发生变化会自动同步配置…

```
zabbix_agent:
  pkg.installed:
    - name: zabbix22-agent

  file.managed:
    - name: /etc/zabbix_agentd.conf
    - source: salt://init/files/zabbix_agentd.conf
    - user: root
    - group: root
    - mode: 644

  service.running:
    - name: zabbix-agentd
    - enable: True
    - watch:
      - file: zabbix_agent
~
```

########pillar+配置管理，实现参数自动化修改，首先需要把配置文件模板化，sls 修改

```
[root@apache-server states]# vim init/files/zabbix_agentd.conf

# master zabbix ip address...

Server={{Zabbix_Server}}


[root@apache-server states]# cat init/zabbix_agent.sls

zabbix_agent:

  pkg.installed:

    - name: zabbix22-agent


  file.managed:

    - name: /etc/zabbix_agentd.conf

    - source: salt://init/files/zabbix_agentd.conf

    - user: root

    - group: root

    - mode: 644
```

```
      - template: jinja
      - defaults:
         Zabbix_Server: {{ pillar['zabbix-agent']['Zabbix_Server'] }}


  service.running:
    - name: zabbix-agentd
    - enable: True
    - reload: True
    - watch:
       - file: zabbix_agent
```

pillar 增加，模板配置默认参数

```
[root@apache-server init]# pwd

/etc/salt/pillar/init

[root@apache-server init]# cat zabbix_agent.sls

zabbix-agent:

  Zabbix_Server: 192.168.2.125

[root@apache-server pillar]# pwd

/etc/salt/pillar

[root@apache-server pillar]# cat top.sls

base:

  '*':

    - init.rsyslog

    - init.zabbix_agent
```

测试 pillar+配置管理功能

```
[root@apache-server pillar]# salt '*' state.highstate test=True
```

```
                # master zabbix ip address...
               -Server=127.0.0.1
               +Server=192.168.2.125

               ### Option: ListenPort
               #    Agent will listen on this port for connections from the server.
----------
          ID: zabbix_agent
    Function: service.running
        Name: zabbix-agentd
      Result: True
     Comment: Service reloaded
     Started: 07:25:53.603001
    Duration: 168.697 ms
     Changes:
              ----------
              zabbix-agentd:
                  True

Summary
------------
Succeeded: 7 (changed=2)
Failed:    0
------------
Total states run:     7
[root@apache-server files]# salt 'minion.apache-server' state.highstate
```

salt 实现源码编译安装 php

[root@apache-server init]# pwd

/etc/salt/states/init

[root@apache-server init]# cat pkg.sls

pkg.init:

  pkg.installed:

    - names:

      - lrzsz

      - mtr

      - nmap

[root@apache-server init]# wget http://cn2.php.net/distributions/php-5.6.17.tar.gz

[root@server1 ~]# mv php-5.6.17.tar.gz /usr/local/src/

 [root@server1 ~]# ls /usr/local/src/

php-5.6.17.tar.gz

[root@apache-server init]# mv pkg.sls pkg.sls.old

[root@apache-server states]# cat init/pkg.sls

```
lamp-configure:

  pkg.installed:

    - names:

      - glibc

      - make

      - gcc

      - gcc-c++

      - bison

      - bison-devel

      - zlib-devel

      - libmcrypt-devel

      - mcrypt

      - mhash-devel

      - openssl-devel

      - libxml2-devel

      - libcurl-devel

      - bzip2-devel

      - readline-devel

      - libedit-devel

      - sqlite-devel

[root@apache-server states]# mkdir php

[root@apache-server states]# cd php

[root@apache-server php]# pwd

/etc/salt/states/php

[root@apache-server php]# mkdir files

[root@apache-server php]# cat php-fastcgi.sls

include:

  - init.pkg


php-install:
```

```
  file.managed:

    - name: /usr/local/src/php-5.6.17.tar.gz

    - source: salt://php/files/php-5.6.17.tar.gz

    - user: root

    - group: root

    - mode: 644


  cmd.run:

    - name: cd /usr/local/src && tar zxf php-5.6.17.tar.gz && cd php-5.6.17
&& ./configure --prefix=/usr/local/php-fastcgi --with-mysql --with-jpeg-dir
 --with-png-dir --with-zlib --enable-xml --with-libxml-dir --with-curl --ena
ble-bcmath --enable-shmop --enable-sysvsem --enable-inline-optimization --e
nable-mbregex --with-openssl --enable-mbstring --with-gd --enable-gd-native
-ttf --enable-sockets --with-xmlrpc --enable-zip --enable-soap --disable-de
bug --enable-zip --with-config-file-path=/usr/local/php-fastcgi/etc --enabl
e-fpm --with-fpm-user=root --with-fpm-group=root && make && make install

    - unless: test -d /usr/local/php-fastcgi
```

[root@apache-server states]# ls /usr/local/src/php-5.6.17.tar.gz

/usr/local/src/php-5.6.17.tar.gz

[root@apache-server states]# cp /usr/local/src/php-5.6.17.tar.gz /etc/salt/
states/php/files/

[root@apache-server states]# ls /etc/salt/states/php/files/

php-5.6.17.tar.gz

#上面的路径有误这里纠正

[root@apache-server states]# pwd

/etc/salt/states

[root@apache-server states]# tree php/

php/

├── files

│ ?? └── php-5.6.17.tar.gz

```
└── php-fastcgi.sls
[root@apache-server states]# cat top.sls
base:
  '(minion.apache-|minion.)server?':
    - init.pkg
    - init.limit
    - match: pcre
    - init.zabbix_agent
    - php.php-fastcgi


[root@apache-server states]# tree
|-- init
|   |-- files
|   |   |-- limits.conf
|   |   `-- zabbix_agentd.conf
|   |-- limit.sls
|   |-- pkg.sls
|   |-- pkg.sls.old
|   `-- zabbix_agent.sls
|-- php
|   |-- files
|   |   `-- php-5.6.17.tar.gz
|   `-- php-fastcgi.sls
|-- prod
`-- top.sls
```

```
[root@apache-server states]# salt 'minion.server1' state.highstate
```

```
[whoami@server1 src]$ ps aux |grep salt
root      1977  0.1  1.5 464324 28808 ?      SI   03:47   0:03 /usr/bin/python2.6 /usr/bin/salt-minion -d
root      2361  4.9  3.1 674964 60500 ?      SI   04:23   0:02 /usr/bin/python2.6 /usr/bin/salt-minion -d
root      2449  5.0  1.3 464324 26544 ?      R    04:24   0:00 /usr/bin/python2.6 /usr/bin/salt-minion -d
root      2450  4.0  1.3 464324 25888 ?      R    04:24   0:00 /usr/bin/python2.6 /usr/bin/salt-minion -d

[whoami@server1 src]$ ps aux |grep yum
root      3133  5.0  3.3 305760 63696 ?      S    04:27   0:01 /usr/bin/python /usr/bin/yum -y install libcurl-devel
```

```
----------
     ID: php-install
Function: cmd.run
    Name: cd /usr/local/src && tar zxf php-5.6.17.tar.gz && cd php-5.6.17 && ./configure --prefix=/usr/local/php-fastcgi --with-mysql --with-jpeg-d
ir --with-png-dir --with-zlib --enable-xml --with-libxml-dir --with-curl --enable-bcmath --enable-shmop --enable-sysvsem --enable-inline-optimization -
-enable-mbregex --with-openssl --enable-mbstring --with-gd --enable-gd-native-ttf --enable-sockets --with-xmlrpc --enable-zip --enable-soap --disable-d
ebug --enable-zip --with-config-file-path=/usr/local/php-fastcgi/etc --enable-fpm --with-fpm-user=root --with-fpm-group=root && make && make install
  Result: True
 Comment: unless execution succeeded
 Started: 05:06:40.775367
Duration: 14.145 ms
 Changes:

Summary
------------
Succeeded: 23 (changed=8)
Failed:     0
------------
Total states run:     23
```

```
[root@server1 src]# tree -L 1
.
├── php-5.6.17
└── php-5.6.17.tar.gz

1 directory, 1 file
[root@server1 src]# ls /usr/local/php-fastcgi/
bin  etc  include  lib  php  sbin  var
[root@server1 src]# /usr/local/php-fastcgi/bin/php --version
PHP 5.6.17 (cli) (built: Jan 20 2016 04:54:31)
Copyright (c) 1997-2015 The PHP Group
Zend Engine v2.6.0, Copyright (c) 1998-2015 Zend Technologies
```

已经执行成功！

Highstate

```
[root@apache-server states]# salt 'minion.server1' state.highstate
```

States Module

```
[root@apache-server ~]#  salt-run manage.up

- minion.apache-server

- minion.server1

[root@apache-server ~]#  salt-run manage.status

down:

up:

    - minion.apache-server

    - minion.server1
```

```
[root@apache-server ~]#  salt-run manage.versions
Master:
    2015.5.5
Up to date:
    ----------
    minion.apache-server:
        2015.5.5
    minion.server1:
        2015.5.5
```

```
[root@apache-server ~]# salt '*' test.ping -v
Executing job with jid 20160121053442513225
-------------------------------------------

minion.server1:
    True
minion.apache-server:
    True
```

job 管理

```
[root@apache-server ~]# salt '*' saltutil.running
minion.apache-server:
minion.server1:
```

[root@apache-server ~]# salt '*' saltutil.kill_job $jid

```
[root@apache-server ~]# salt '*' saltutil.kill_job 20160121053442513225
```

salt ssh master 端安装

[root@apache-server ~]# yum install salt-ssh –y

```
Installed:
  salt-ssh.noarch 0:2015.5.8-1.el6
Dependency Updated:
  salt.noarch 0:2015.5.8-1.el6          salt-master.noarch 0:2015.5.8-1.el6          salt-minion.noarch 0:2015.5.8-1.el6
Complete!
```

编辑配置文件：

[root@apache-server ~]# vim /etc/salt/roster

[root@apache-server ~]# tail -4 /etc/salt/roster

minion.server1:

  host: 192.168.2.200

  user: root

  passwd: hadoop

[root@apache-server ~]# /etc/init.d/salt-master restart

Stopping salt-master daemon: [OK]

Starting salt-master daemon: [OK]

```
[root@apache-server ~]# salt-ssh '*' cmd.run 'w'
minion.server1:
    06:15:02 up 38 min,  1 user,  load average: 0.00, 0.00, 0.00
    USER     TTY      FROM             LOGIN@   IDLE   JCPU   PCPU WHAT
    whoami   pts/1    192.168.2.104    05:37    2:09   0.03s  0.09s sshd: whoami [p
```

#依赖

```
include:
  - init.pkg
  - init.www
          I

php-install:
  file.managed:
    - name: /usr/local/src/php-5.5.13.tar.gz
    - source: salt://php/files/php-5.5.13.tar.gz
    - user: root
    - group: root
    - mode: 755
  cmd.run:
    - name: cd /usr/local/src && tar zxf php-5.5.13.tar.gz && cd php-5.5.13 && ./config
    - unless: test -d /usr/local/php-fastcgi
  require:
    - file: php-install
    - pkg.installed: php-install

pdo-plugin:
  cmd.run:
    - name: cd /usr/local/src/php-5.5.13/ext/pdo_mysql/ && /usr/local/php-fastcgi/bin/p
    - unless: test -f /usr/local/php-fastcgi/lib/php/extensions/no-debug-non-zts-201212
  require:
    - cmd: php-install
```

5.Saltstack 管理实践

6.Saltstack 实践案例

7.Saltstack 实现 openstack 自动化部署