Testing SaltOpen’s Features

Author: saltstack community

Scope of This Document

Testing document on some saltstack topics.

Glossary

Name	Definition
Special Name	And its meaning here.

Orchestration

- Docs
    - https://docs.saltproject.io/salt/user-guide/en/latest/topics/runners-orchestration.html
    - end
- End

Start up state

- Startup States
- https://www.reddit.com/r/saltstack/comments/1f13nec/help_state_not_applying_at_the_minion_start/?rdt=34268

Using /opt/saltstack/salt/bin/soluble

1. DOC:
    a. https://github.com/saltstack/soluble
    b. end
2. WHAT:
    a. Auto accept salt-minion if it provide a pre authorized grain key.
3. WHY:
    a. To speedup the onboarding salt-minion without login salt-master to accept minion key.
4. HOW: Success of mater downgrade operation is vital to keep existing configuration.
    a. Install
       $ sudo /opt/saltstack/salt/bin/pip3 install soluble
       $ sudo ln -s /opt/saltstack/salt/bin/soluble /usr/bin/soluble
       $ sudo /opt/saltstack/salt/bin/soluble --versions-report
    b. /opt/saltstack/salt/bin/soluble
    c. end
5. End

Using pre-shared grain key to accept minion

1. DOC:
    a. Autoaccept minions from Grains by uuid grain
2. WHAT:
    a. Auto accept salt-minion if it provides a pre authorized grain key-
    b. Without blindly auto accept a minion connection with a controlled grain key.
3. WHY:
    a. To speedup the onboarding salt-minion without login salt-master to accept minion key.
4. HOW: Success of mater downgrade operation is vital to keep existing configuration.
    a. Salt Server:
        i. Master to have following config
        [me@salt01 ~]$ cat /etc/salt/master.d/autosign_grains.conf
        autosign_grains_dir: /etc/salt/autosign_grains
        [me@salt01 ~]$
        ii. 1st psk key for minion group 1.
        openssl rand -hex 8 >> /etc/salt/autosign_grains/psk
        60d3e6ce10350756
        iii. Generate 2nd psk key for minion group 2.
        openssl rand -hex 8 >> /etc/salt/autosign_grains/psk
        iv. Check psk keys generated.
        wc -l /etc/salt/autosign_grains/psk
    b. Minion side:
        i. Shell script to install a new LTS minion

```
# WHAT: my bootstrap script on top of https://bootstrap.saltproject.io
# As root user on newer version of OS
# make sure we can reach salt01/02.mot.com and salt repo
echo $HOSTNAME && which rpm && which sed && which ncat
nc -zv salt01.test.net 4505 && nc -zv salt01.test.net 4506 && nc
-zvbootstrap.saltproject.io 443
cd ${HOME}; curl -o bootstrap-salt.sh -L https://bootstrap.saltproject.io
# Install LTS minion which is currently 3006
bash bootstrap-salt.sh onedir 3006
# Adjust minion setting
sed -i.bak  -e 's!^#master:.*!#master: salt\nmaster: ["salt01.test.net"]!'
/etc/salt/minion
sed -i.bak  -e 's!^#hash_type.*!hash_type: sha256!' /etc/salt/minion
sed -i.bak  -e 's!^#rejected_retry: False!rejected_retry: True!'
/etc/salt/minion
egrep '^master|^hash_type|^rejected_retry' /etc/salt/minion
hostname -f > /etc/salt/minion_id



systemctl enable --now salt-minion && sleep 3 && systemctl status salt-minion
```

```
salt-call --master=salt01.test.net test.version
```

c. end

5. End

Salt-master STS2LTS downgrade

6. DOC:
    a. end
7. WHAT: Downgrade salt-master from STS(3007.x) to LTS(3006.x)
8. WHY: STS salt-masters found to be not stable from a few bug reports. Publish port on master will die over a period uptime, unless restart master.
9. HOW: Success of mater downgrade operation is vital to keep existing configuration.
    a. Switch over to use LTS repo
    b. RedHat salt-master, Do a dnf downgrade salt-* on salt-master.
    c. end
10. End

Minion STS2LTS downgrade

1. DOC:
    a. end
2. WHAT: Downgrade salt minions from STS(3007.x) to LTS(3006.x)
3. WHY: STS salt-masters found to be not stable from a few bug reports. Publish port on master will died unless restart master.
4. HOW:
    trick is to change repo pointing to 3006 for both RedHat and Debian based OS.
    RedHat minion, we can just do a pkg downgrup.
    Debian minion, I only know remove old one and install new one.
    a. Example 2LTS.sls file under salt01:/srv/salt

```
{% if salt['grains.get']('os_family') == 'RedHat' %}

Install_3006_LTS_repo:
  file.managed:
    - stateful: True
    - name: /etc/yum.repos.d/salt.repo
    - source: salt://salt/files/3006-LTS_RHEL.repo
    - user: root
    - group: root
    - mode: '0644'
#  parsing script output! Stdout must be JSON or a line of name=value pairs.
dnf_repo_refresh:
  cmd.run:
#    - stateful: True
    - name: dnf clean all
#So that minion can reconnect instead of just die.
enable_retry_not_exit:
  file.line:
      - name: /etc/salt/minion
      - mode: ensure
      - content: "rejected_retry: True"
      - before: ^#rejected_retry:.*
#
downgrade2lts_restart_minion:
  cmd.run:
    - stateful: True
    - name: echo "dnf downgrade -y salt " | at -v now
    - unless: ps -ef | grep -v grep | grep /usr/sbin/atd
{% endif %}

{% if grains['os_family'] == 'Debian' %}
{% if grains['osmajorrelease'] in [23] %}
not_supported:
  cmd.run:
    - stateful: True
    - name: echo "ubuntu 23 is not supported by saltstack"
```

```
{% endif %}
Install_3006-LTS_DEB_20.repo:
  file.managed:
    - stateful: True
    - name: /root/bootstrap-salt.sh
    - source: salt://salt/files/bootstrap-salt.sh.2024.01.04
    - user: root
    - group: root
    - mode: '0700'
enable_retry_not_exit:
  file.line:
      - name: /etc/salt/minion
      - mode: ensure
      - content: "rejected_retry: True"
      - before: ^#rejected_retry:.*
{% if grains['osmajorrelease'] in [20,22,24] %}

Install_3006-LTS_DEB_{{ grains['osmajorrelease'] }}.repo:
  file.managed:
    - stateful: True
    - name: /etc/apt/sources.list.d/salt.list
    - source: salt://salt/files/3006-LTS_DEB_{{ grains['osmajorrelease'] }}.repo
    - user: root
    - group: root
    - mode: '0644'
downgrade2lts_restart_minion:
  cmd.run:
    - stateful: True
    - name: echo "apt remove -y salt-minion salt-common && bash /root/bootstrap-salt.sh onedir
3006 && sleep 3 && systemctl restart salt-minion" | at -v now
    - unless: ps -ef | grep -v grep | grep /usr/sbin/atd
{% endif %}

{% endif %}
```

b. Repository files for LTS (3006) for different OS.

```
[me@salt01 files]$ more  *.repo
::::::::::::::
3006-LTS_DEB_20.repo
::::::::::::::
deb [signed-by=/usr/share/keyrings/salt-archive-keyring.gpg]
https://repo.saltproject.io/salt/py3/ubuntu/20.04/amd64/3006/ jammy main
::::::::::::::
3006-LTS_DEB_22.repo
::::::::::::::
deb [signed-by=/usr/share/keyrings/salt-archive-keyring.gpg]
https://repo.saltproject.io/salt/py3/ubuntu/22.04/amd64/3006/ jammy main
::::::::::::::
3006-LTS_DEB_24.repo
::::::::::::::
```

```
deb [signed-by=/usr/share/keyrings/salt-archive-keyring.gpg]
https://repo.saltproject.io/salt/py3/ubuntu/24.04/amd64/3006/ noble  main
::::::::::::::
3006-LTS_RHEL.repo
::::::::::::::
[salt-repo]
name=Salt repo for RHEL/CentOS 8 PY3
baseurl=https://repo.saltproject.io/salt/py3/redhat/8/x86_64/3006
skip_if_unavailable=True
priority=10
enabled=1
enabled_metadata=1
gpgcheck=1
gpgkey=https://repo.saltproject.io/salt/py3/redhat/8/x86_64/3006/SALT-PROJECT-GP
G-PUBKEY-2023.pub
::::::::::::::
STS_DEB_20.repo
::::::::::::::
deb [signed-by=/usr/share/keyrings/salt-archive-keyring.gpg]
https://repo.saltproject.io/salt/py3/ubuntu/20.04/amd64/latest/ jammy main
::::::::::::::
STS_DEB_22.repo
::::::::::::::
deb [signed-by=/usr/share/keyrings/salt-archive-keyring.gpg]
https://repo.saltproject.io/salt/py3/ubuntu/22.04/amd64/latest/ jammy main
::::::::::::::
STS_RHEL.repo
::::::::::::::
[salt-repo]
name=Salt repo for RHEL/CentOS 8 PY3
baseurl=https://repo.saltproject.io/salt/py3/redhat/8/x86_64/latest
skip_if_unavailable=True
priority=10
enabled=1
enabled_metadata=1
gpgcheck=1
gpgkey=https://repo.saltproject.io/salt/py3/redhat/8/x86_64/3006/SALT-PROJECT-GP
G-PUBKEY-2023.pub
[me@salt01 files]$
```

c. end

5. End

Minion LTS2STS

1. DOC:
    a. Master Cluster
    b. end
2. WHAT: Using salt's default HA and load balancing without pacemaker add-on package.
3. WHY: TBC
4. HOW: TBCo

Using saltstack's default HA and LB support

1. DOC:
    a. Master Cluster
    b. end
2. WHAT: Using salt's default HA and load balancing without pacemaker add-on package.
3. WHY: TBC
4. HOW: TBCo

Dead minions management process

1. DOC:
    a. https://www.reddit.com/r/saltstack/comments/1e1br5n/saltstack_and_dead_minions_discoverymanagement/?rdt=39247
    b. Compound matchers
2. WHAT: Find a better way to deal with minion not responding
3. WHY: TBC
4. HOW:
    a. Remove dead minion after a while
    b. Exclude Dead minions by using -C , compound RE after first run.
        i. Enable summary report.
        ii. Test.ping first to see the dead minion
        iii. Exclude the dead minions using compund and L
        ```
        salt -C "* and not L@dead01,dead02,dead02," test.ping
        ```
        iv. Change * to with -C statement.
    c. End

Using --out=json and jq for return formating

1. DOC:
    a. end
2. WHAT:
3. WHY: TBC
4. HOW:
    a. end

Salt ACL testing

5. What: Limit regular user account (me) to test.ping salt commands by saltstacks ACL system.
6. WHY: Access control for salt users needing certain commands to only certain minions
7. HOW:
    a. Prepare a rockylinux i testing environment
    b. Have salt-master running as salt user.

```
[me@rocky9t01 ~]$ sudo egrep ^user\:  /etc/salt/master
user: salt
[me@rocky9t01 ~]$
```

c. Use me test account and make sure it is in salt group.

```
[me@rocky9t01 ~]$ id
uid=1000(me) gid=1000(me) groups=1000(me),10(wheel),982(salt)
[me@rocky9t01 ~]$
```

d. Enable group rw for salt group.

```
# sudo chmod g+w  /var/log/salt/master
```

e. Configure ACL in /etc/salt/master.d/acl.conf

```
[me@rocky9t01 ~]$ cat /etc/salt/master.d/acl.conf |egrep -v '^#|^$'
publisher_acl:
  me:
    - test.ping
    - network.*
external_auth:
  pam:
    me:
      - test.ping
    saltdev:
      - test.*
[me@rocky9t01 ~]$
```

f. Test out the granted test.ping command without using sudo as root.

```
[me@rocky9t01 ~]$ salt rocky9t01 test.ping
rocky9t01:
    True
[me@rocky9t01 ~]$
```

g. Restart salt-master when acl.conf changed. *.ipc got set back to 600 from 660.

```
[me@rocky9t01 ~]$ ls -l /var/run/salt/master/*
srw-rw---- 1 salt salt 0 Jun 23 14:06 /var/run/salt/master/master_event_pub.ipc
srw------- 1 salt salt 0 Jun 23 14:06 /var/run/salt/master/master_event_pull.ipc
srw------- 1 salt salt 0 Jun 23 14:06 /var/run/salt/master/publish_pull.ipc
srw------- 1 salt salt 0 Jun 23 14:06 /var/run/salt/master/workers.ipc
[me@rocky9t01 ~]$
```

h. Testing unpermitted **test.version** module.

```
[me@rocky9t01 ~]$ salt \* test.version
Authorization error occurred.
[me@rocky9t01 ~]$
```

i. end

8. Bug
    a. Why salt-master restart reset the file permission in /var/run/salt/master/
9. End

Grains

1. DOC:
    a. How to automatically add custom grains attributes to minions using SaltStack Reactor? - Stack Overflow
    b. end
2. End

Salt-api using pepper

1. Docs:
    a. https://docs.saltproject.io/en/latest/ref/cli/salt-api.html
    b. https://github.com/saltstack/pepper
    c. GitHub - saltstack/pepper: A library and stand-alone CLI tools to access a salt-api instance
    d. pepper
    e. end
2. On onedir 3006.8, it is using its own python environ ment.

```
[me@rocky9t01 ~]$ find /opt/saltstack/salt/bin
/opt/saltstack/salt/bin
/opt/saltstack/salt/bin/calc-prorate
/opt/saltstack/salt/bin/cheroot
/opt/saltstack/salt/bin/cherryd
/opt/saltstack/salt/bin/distro
/opt/saltstack/salt/bin/jp.py
/opt/saltstack/salt/bin/normalizer
/opt/saltstack/salt/bin/pip
/opt/saltstack/salt/bin/pip3
/opt/saltstack/salt/bin/pip3.10
/opt/saltstack/salt/bin/python3
/opt/saltstack/salt/bin/python3-config
/opt/saltstack/salt/bin/python3.10
/opt/saltstack/salt/bin/python3.10-config
/opt/saltstack/salt/bin/relenv
[me@rocky9t01 ~]$
```

3. Install pepper

```
# sudo /opt/saltstack/salt/bin/pip install salt-pepper  salt-pepper CherryPy PyOpenSSL

[me@rocky9t01 ~]$ sudo /opt/saltstack/salt/bin/pip install salt-pepper  salt-pepper
CherryPy PyOpenSSL
Requirement already satisfied: salt-pepper in
/opt/saltstack/salt/lib/python3.10/site-packages (0.7.6)
```

```
Requirement already satisfied: CherryPy in
/opt/saltstack/salt/lib/python3.10/site-packages (18.6.1)
Requirement already satisfied: PyOpenSSL in
/opt/saltstack/salt/lib/python3.10/site-packages (24.0.0)
Requirement already satisfied: cheroot>=8.2.1 in
/opt/saltstack/salt/lib/python3.10/site-packages (from CherryPy) (8.5.2)
Requirement already satisfied: portend>=2.1.1 in
/opt/saltstack/salt/lib/python3.10/site-packages (from CherryPy) (2.4)
Requirement already satisfied: more-itertools in
/opt/saltstack/salt/lib/python3.10/site-packages (from CherryPy) (5.0.0)
Requirement already satisfied: zc.lockfile in
/opt/saltstack/salt/lib/python3.10/site-packages (from CherryPy) (1.4)
Requirement already satisfied: jaraco.collections in
/opt/saltstack/salt/lib/python3.10/site-packages (from CherryPy) (3.4.0)
Requirement already satisfied: cryptography<43,>=41.0.5 in
/opt/saltstack/salt/lib/python3.10/site-packages (from PyOpenSSL) (42.0.5)
Requirement already satisfied: six>=1.11.0 in
/opt/saltstack/salt/lib/python3.10/site-packages (from
cffi>=1.12->cryptography<43,>=41.0.5->PyOpenSSL) (2.21)
Requirement already satisfied: pytz in /opt/saltstack/salt/lib/python3.10/site-packages
(from tempora>=1.8->portend>=2.1.1->CherryPy) (2022.1)
WARNING: Running pip as the 'root' user can result in broken permissions and
conflicting behaviour with the system package manager. It is recommended to use a
virtual environment instead: https://pip.pypa.io/warnings/venv
<snipped>
[notice] A new release of pip is available: 23.3.2 -> 24.1
[notice] To update, run: /opt/saltstack/salt/bin/../bin/python3.10 -m pip install
--upgrade pip
[me@rocky9t01 ~]$
[me@rocky9t01 ~]$ ls -l /opt/saltstack/salt/bin/*pep*
-rwxr-xr-x 1 root root 574 Jun 23 14:30 /opt/saltstack/salt/bin/pepper
[me@rocky9t01 ~]$
```

4. home/.pepperrc

```
[me@rocky9t01 ~]$ cat .pepperrc
[main]
SALTAPI_URL=https://localhost:8000/
SALTAPI_USER=saltdev
SALTAPI_PASS=saltdev
SALTAPI_EAUTH=pam
[me@rocky9t01 ~]$
```

5. Create self-signed certificate

```
[me@rocky9t01 ~]$ sudo salt-call tls.create_self_signed_cert
local:
    Created Private Key: "/etc/pki/tls/certs/localhost.key" Created Certificate:
"/etc/pki/tls/certs/localhost.crt"
[me@rocky9t01 ~]$
```

6. Change key to salt-master running user “salt”
   sudo chown salt:salt /etc/pki/tls/certs/localhost.*

7. /etc/salt/master.d/salt-api.conf
   rest_cherrypy:
     port: 8000
     ssl_crt: /etc/pki/tls/certs/localhost.crt
     ssl_key: /etc/pki/tls/certs/localhost.key
8. Log file level
   ```
   [me@rocky9t01 ~]$ grep log /usr/lib/systemd/system/salt-api.service
   ExecStart=/usr/bin/salt-api --log-file-level=trace
   --log-file=/var/log/salt/salt-api.log
   [me@rocky9t01 ~]$
   ```
9. Log files from tail -f /var/log/salt/salt-api.log
   ```
   2024-06-23 15:16:50,240 [salt.utils.event :311 ][DEBUG   ][45830] MasterEvent PUB socket URI:
   /var/run/salt/master/master_event_pub.ipc
   2024-06-23 15:16:50,240 [salt.utils.event :312 ][DEBUG   ][45830] MasterEvent PULL socket URI:
   /var/run/salt/master/master_event_pull.ipc
   2024-06-23 15:16:50,242 [cherrypy.error   :213 ][INFO    ][45830] [23/Jun/2024:15:16:50] ENGINE
   Listening for SIGTERM.
   2024-06-23 15:16:50,242 [cherrypy.error   :213 ][INFO    ][45830] [23/Jun/2024:15:16:50] ENGINE
   Listening for SIGHUP.
   2024-06-23 15:16:50,242 [cherrypy.error   :213 ][INFO    ][45830] [23/Jun/2024:15:16:50] ENGINE
   Listening for SIGUSR1.
   2024-06-23 15:16:50,242 [cherrypy.error   :213 ][INFO    ][45830] [23/Jun/2024:15:16:50] ENGINE
   Bus STARTING
   2024-06-23 15:16:50,388 [cherrypy.error   :213 ][INFO    ][45830] [23/Jun/2024:15:16:50] ENGINE
   Serving on https://0.0.0.0:8000
   2024-06-23 15:16:50,389 [cherrypy.error   :213 ][INFO    ][45830] [23/Jun/2024:15:16:50] ENGINE
   Bus STARTED
   2024-06-23 15:16:59,861 [salt.utils.process:32  ][TRACE   ][45782] Process manager iteration
   2024-06-23 15:17:09,870 [salt.utils.process:32  ][TRACE   ][45782] Process manager iteration
   ```
10. testing
    ```
    export SALTAPI_USER=saltdev SALTAPI_PASS=saltdev SALTAPI_EAUTH=pam
    /opt/saltstack/salt/bin/pepper '*' test.ping
    /opt/saltstack/salt/bin/pepper '*' test.kwarg hello=dolly
    ```
11.
    ```
    [me@rocky9t01 ~]$ systemctl status salt-api;date
    ● salt-api.service - The Salt API
         Loaded: loaded (/usr/lib/systemd/system/salt-api.service; enabled; preset: disabled)
         Active: active (running) since Sun 2024-06-23 15:30:26 CDT; 9min ago
           Docs: man:salt-api(1)
                 file:///usr/share/doc/salt/html/contents.html
                 https://docs.saltproject.io/en/latest/contents.html
       Main PID: 48452 (/opt/saltstack/)
          Tasks: 103 (limit: 203326)
         Memory: 96.4M
            CPU: 2.740s
         CGroup: /system.slice/salt-api.service
    ```

```
                ├─48452 "/opt/saltstack/salt/bin/python3.10 /usr/bin/salt-api --log-file-level=trace
    --log-file=/var/log/salt/salt-api.log MainProc>
                └─48475 "/opt/saltstack/salt/bin/python3.10 /usr/bin/salt-api --log-file-level=trace
    --log-file=/var/log/salt/salt-api.log RunNetap>

    Jun 23 15:30:26 rocky9t01 systemd[1]: Starting The Salt API...
    Jun 23 15:30:26 rocky9t01 systemd[1]: Started The Salt API.
    Sun Jun 23 03:39:48 PM CDT 2024
    [me@rocky9t01 ~]$
```

12. Saltdev api user.

```
[saltdev@rocky9t01 ~]$ id
uid=1001(saltdev) gid=1001(saltdev) groups=1001(saltdev),982(salt)
[saltdev@rocky9t01 ~]$
```

13. Oneliner test.

/opt/saltstack/salt/bin/pepper --saltapi-url=https://localhost:8080 --username=saltdev

14. Fail test

```
[me@rocky9t01 ~]$  /opt/saltstack/salt/bin/pepper rocky9t01  test.ping
Pepper error: Authentication denied
[me@rocky9t01 ~]$
```

15. Error :`Could not authenticate using provided credentials`

```
[root@rocky9t01 ~]# curl -sSk https://localhost:8000/login  -H 'Accept: application/x-yaml' -d
username=saltdev -d password=saltdev -d eauth=pam
<!DOCTYPE html PUBLIC
"-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html>
<head>
    <meta http-equiv="Content-Type" content="text/html; charset=utf-8"></meta>
    <title>401 Unauthorized</title>
    <style type="text/css">
    #powered_by {
        margin-top: 20px;
        border-top: 2px solid black;
        font-style: italic;
    }

    #traceback {
        color: red;
    }
    </style>
</head>
    <body>
        <h2>401 Unauthorized</h2>
        <p>Could not authenticate using provided credentials</p>
        <pre id="traceback"></pre>
    <div id="powered_by">
      <span>
```

```
        Powered by <a href="http://www.cherrypy.org">CherryPy 18.6.1</a>
      </span>
    </div>
    </body>
</html>
[root@rocky9t01 ~]#
```

16. End

Salt-ssh

1. DOC:
    a. Salt SSH
    b. salt-ssh vs salt heist
    c. end
2. End

Salt gitfs

Gitfs gitlab ssh:// support on Rocky8

1.
2. HOW
    a. sudo dnf install -y python3-pygit2
    ```
    (1/3): python3-pygit2-0.26.4-1.el8.x86_64.rpm   1.3 MB/s | 159 kB     00:00
    (3/3): libgit2-0.26.8-2.el8.x86_64.rpm          433 kB/s | 453 kB     00:01
    ```
    b. Gitlab only support ssh:// and https:// syntax from git client.
    c. old
    ```
    [me@rocky9t01 ~]$ cksum /usr/lib64/libgit2.so.0.26.8
    2900388378 1031352 /usr/lib64/libgit2.so.0.26.8
    [me@rocky9t01 ~]$
    ```
    d. Install recompiled libgit2-0.26.8-3.el8.x86_64.rpm which depends on libssh2 rpm.
    e. new
    ```
    [me@rocky9t01 ~]$ cksum /usr/lib64/libgit2.so.0.26.8
    1912634117 1044000 /usr/lib64/libgit2.so.0.26.8
    [me@rocky9t01 ~]$
    ```
    f. Gitfs goal
    ```
    [me@rocky9t01 ~]$ sudo salt-run fileserver.dir_list
    - ansiblegate
    - test01-github-com-ssh
    - test01-gitlab-com-ssh
    [me@rocky9t01 ~]$
    ```
    g. Make sure manual git check out works.
    ```
    [me@rocky9t01 ~]$ git clone git@gitlab.com:tjyang/mysalt.git
    Cloning into 'mysalt'...
    remote: Enumerating objects: 3, done.
    remote: Counting objects: 100% (3/3), done.
    remote: Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
    Receiving objects: 100% (3/3), done.
    [me@rocky9t01 ~]$

    git clone  git://github.com/me/salt-formula.git
    ```

    h. Following in /etc/salt/master should works
    ```
    gitfs_remotes:
    - ssh://git@gitlab.com/tjyang/mysalt.git:
       - pubkey:  /home/me/.ssh/id_rsa.pub
       - privkey: /home/me/.ssh/id_rsa
       - mountpoint: salt://test01-gitlab-com-ssh
    ```
        i. Command line of above ssh:// syntax
        ```
        #git clone git@gitlab.com:tjyang/mysalt.git test01
          [me@salt01c ~]$ ls .ssh/id*
          .ssh/id_rsa  .ssh/id_rsa.pub
          [me@salt01c ~]$
        ```
        ii. end

i. Error message in /var/log/salt/master

```
2022-11-27 13:05:29,405 [salt.utils.gitfs ][ERROR   ] Unable to fetch SSH-based gitfs
remote 'ssh://git@gitlab.com/tjyang/mysalt.git'. You may need to add ssh:// to the repo
string or libgit2 must be compiled with libssh2 to support SSH authentication.
Traceback (most recent call last):
  File "/usr/lib/python3.6/site-packages/salt/utils/gitfs.py", line 1870, in _fetch
    fetch_results = origin.fetch(**fetch_kwargs)
  File "/usr/lib64/python3.6/site-packages/pygit2/remote.py", line 405, in fetch
    check_error(err)
  File "/usr/lib64/python3.6/site-packages/pygit2/errors.py", line 64, in check_error
    raise GitError(message)
_pygit2.GitError: unsupported URL protocol
```

j. Pygit2 bind with libgit2 which need to have SSH enabled

```
  %cmake -DTHREADSAFE=ON -DUSE_SSH=OFF ..
/usr/bin/python3 -c "import pygit2 as pg; bool(pg.features and pg.GIT_FEATURE_SSH)";echo $?
```

k. end

3. WHY
    a. RHEL8 rpm doesn’t support libssh
    b. End

4. HOW to USE_SSH in libgit2 ?
    a. Rebuild .rpm
    b. Use pip to install
    c. end
5. References:
6. end

This is Heading 1 with a Table

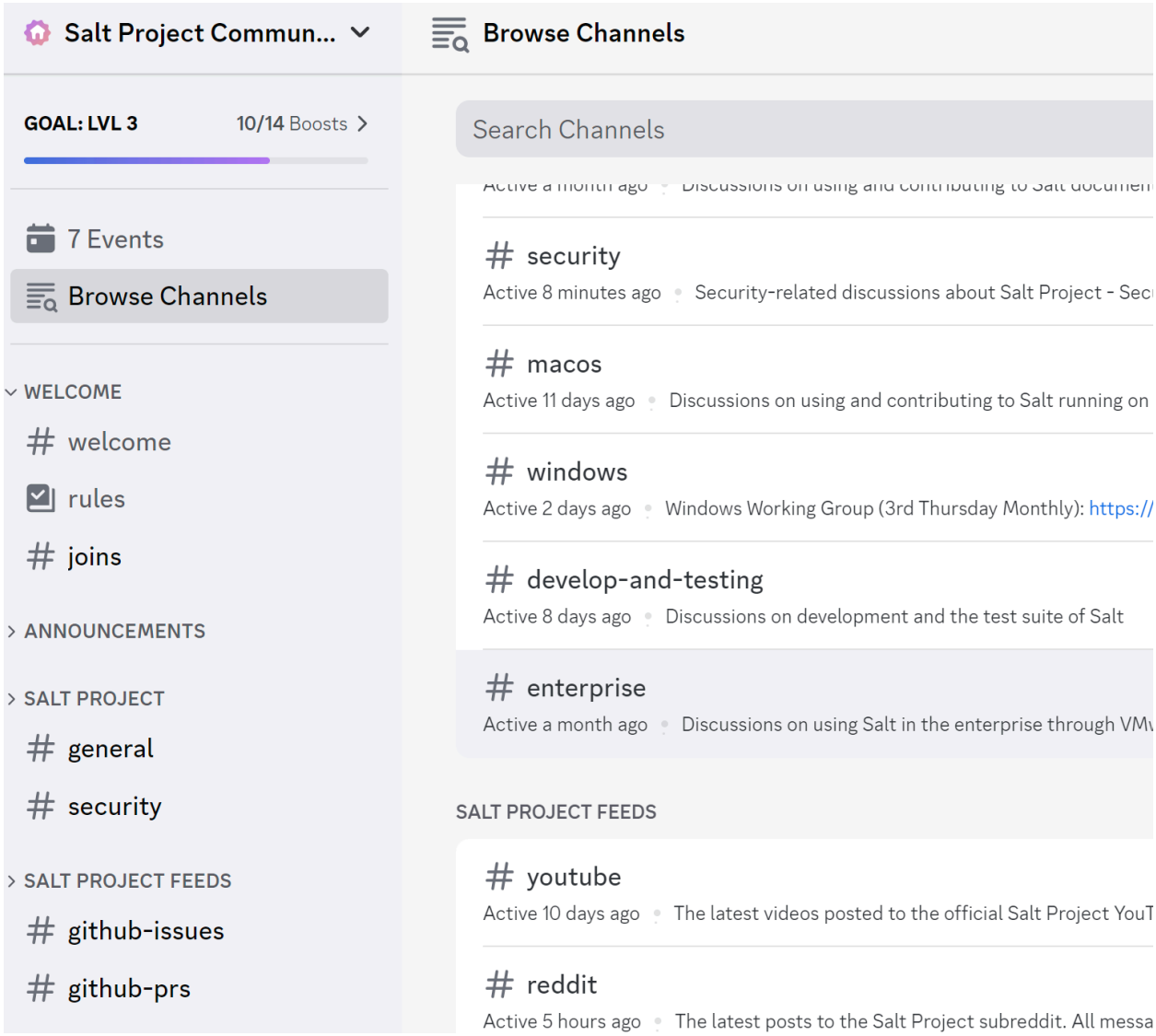
Name		

Revision History

Date	**18Name**	**Comment of changes**
06/18/2024	tjyang2001@gmail.com	**init**

References

1. Discord



2. end