Juniper Event Driven Infrastructure auto remediation demo

Khelil Sator
ksator@juniper.net
November 2017

ABOUT J-EDI

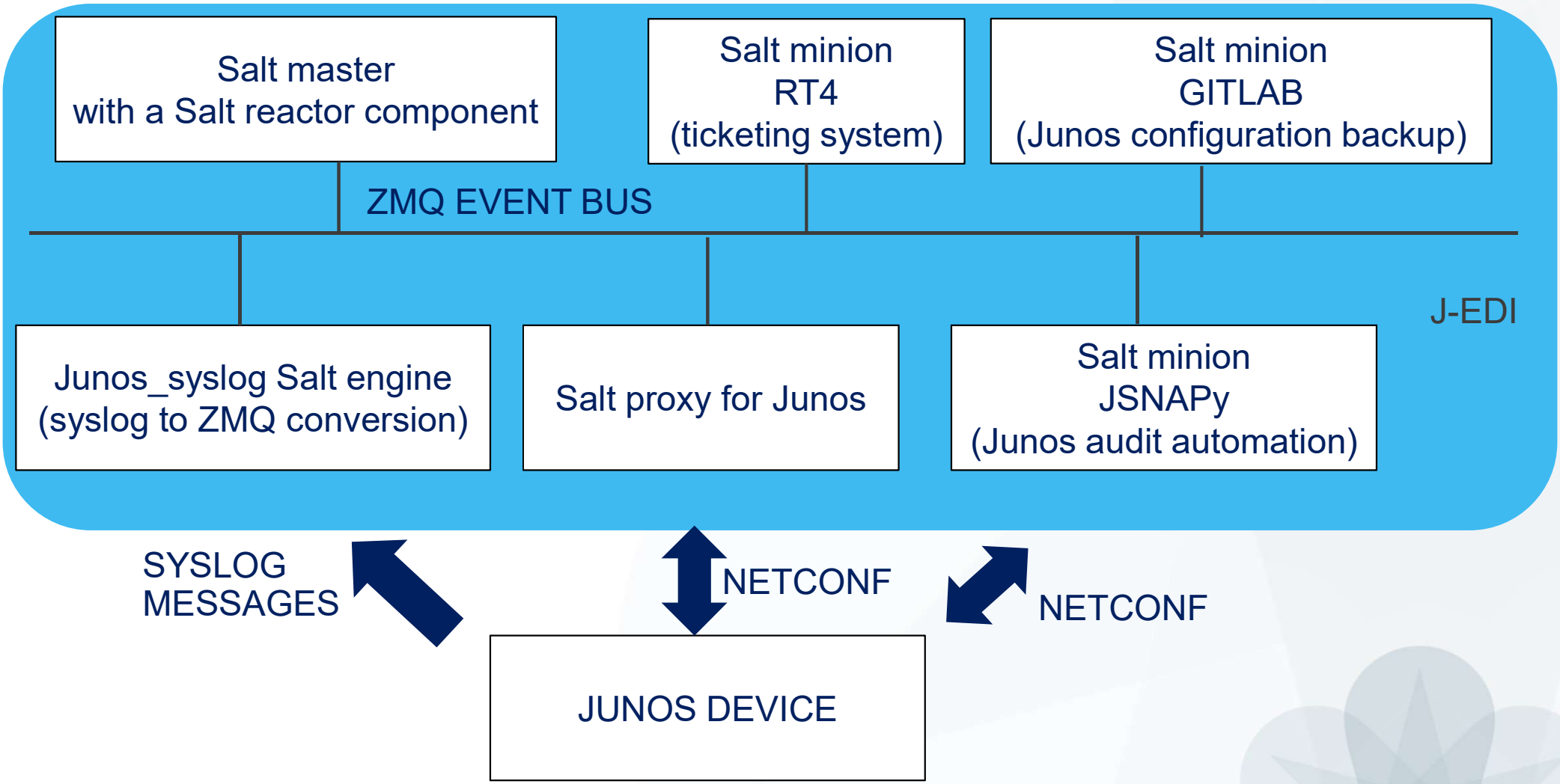
- Juniper Event Driven Infrastructure (J-EDI)
  - Uses regular/native SaltStack building blocks
    - Salt master, minions, event bus, reactor, …
    - Salt proxies for Junos (developed by Juniper and provided to SaltStack)
    - Salt modules for Junos (developed by Juniper and provided to SaltStack)
    - Salt junos_syslog engines (developed by Juniper and provided to SaltStack)
  - Loosely couples a collection of open-source and Juniper maintained tools
    - Junos Space, JSNAPy, OpenNTI, Appformix, Juniper Secure Analytics, Request Tracker 4, …
  - Is developed by Juniper.
  - Is installed, configured, and supported by Juniper Professional Services.

J-EDI PLUGINS USED FOR THIS DEMO

- Junos Backup plugin
  - To automate JUNOS configuration backup on a GITLAB server at each JUNOS commit.

- JSNAPy plugin
  - JSNAPy is a tool to automate verifications on Junos devices (operational state verifications and configuration verifications).
  - To automate JUNOS configuration compliance check, J-EDI uses JSNAPy python library to run a test at each JUNOS commit, and send a 0MQ message to the SaltStack event Bus with the JSNAPy test result details.
  - Auto remediation is done for compliance check failures using a SaltStack proxy for Junos.

- Request Tracker 4 plugin
  - RT4 is a popular, widely deployed, ticketing system. It has an API that can be used for the CRUD operations against tickets.
  - J-EDI interacts with RT4 using its API to automate tickets manipulation (tickets creation and update)



SALT PROXY for JUNOS

- For help about Junos automation with SaltStack, you can visit this repository https://github.com/ksator/junos-automation-with-saltstack

- SaltStack supports Junos automation with a Salt proxy
  - Proxy controls junos devices without installing salt on device.
  - It uses Junos API: junos-eznc python library (pyez) and NETCONF on the device.

- It provides execution modules for Junos so you can run commands on various machines in parallel with a flexible targeting system
  - https://docs.saltstack.com/en/latest/ref/modules/all/salt.modules.junos.html

- It provides state modules for Junos so you can apply sls files
  - https://docs.saltstack.com/en/latest/ref/states/all/salt.states.junos.html

- Junos facts are stored in salt grains

SALT PROXY for JUNOS

```
root@JEDI-cluster-demo:~# salt 'vsrx' test.ping
vsrx:
    True

root@JEDI-cluster-demo:~# salt 'vsrx' junos.cli 'show version'
vsrx:
    ----------
    message:

        Hostname: vsrx
        Model: vsrx
        Junos: 15.1X49-D100.6
        JUNOS Software Release [15.1X49-D100.6]
    out:
        True
```

JUNOS SYSLOG SALT ENGINE

- Listens to syslog events
- Extracts events information
- Sends information on the master/minion event bus.
- Control the type of events to be sent.
- Salt reactors has the ability to take actions according to these events (event driven automation).

- Junos_syslog engine configuration

```
root@JEDI-cluster-demo:~# more /etc/salt/master
...
engines:
  - junos_syslog:
      port: 516
...
```

JUNOS SYSLOG CONFIGURATION

- For junos_syslog engine to receive events, syslog must be set on the junos device:
  - The ip address is the one of the server running the syslog engine
  - The port is the port where the engine is listening for events.

```
ksator@vsrx> show configuration system syslog host 192.168.233.204
any any;
match UI_COMMIT_COMPLETED;
port 516;
```

JSNAPy TEST FILE

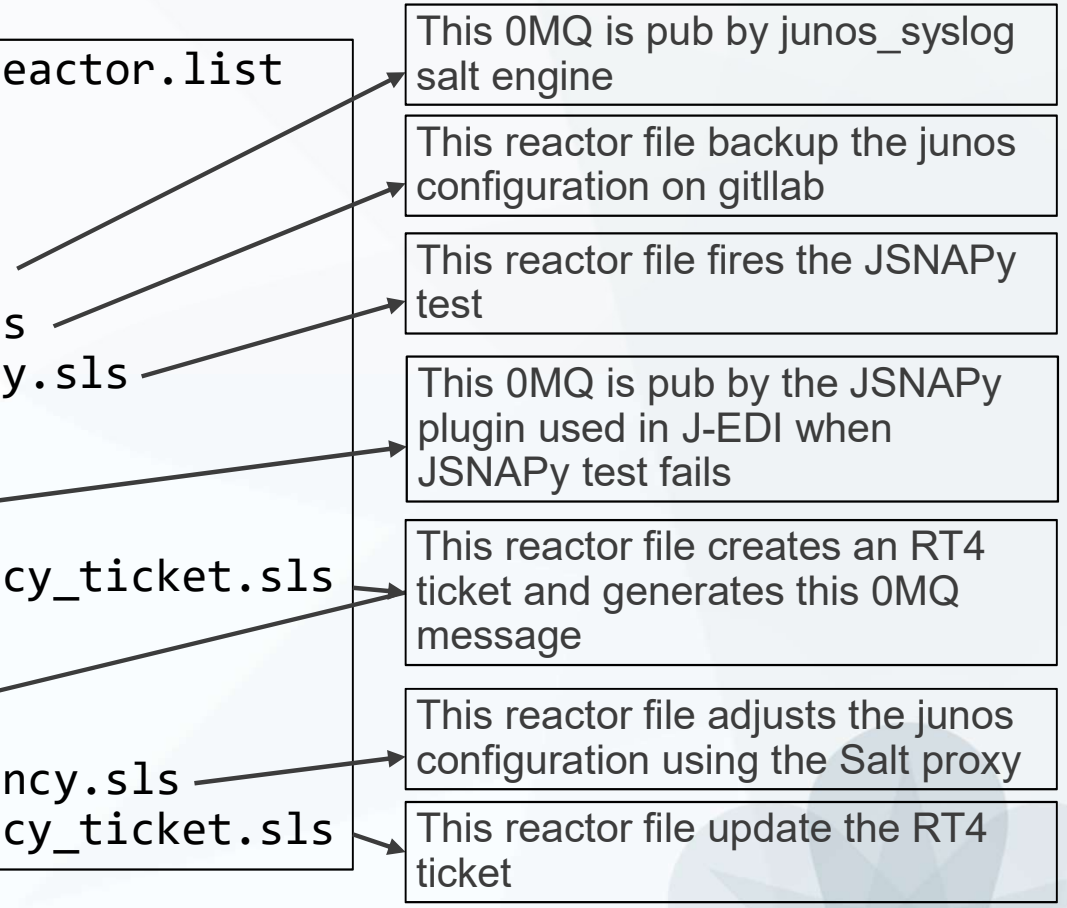
- For help about Junos verification with JSNAPy, you can visit this repository https://github.com/ksator/junos-verifications-automation-with-jsnapy
- So telnet configuration is not allowed

```
root@JEDI-cluster-demo:~# more /etc/jsnapy/testfiles/test_telnet.yml
tests_include:
  - test_telnet_config

test_telnet_config:
  - rpc: get-config
  - kwargs:
      filter_xml: configuration/system
  - item:
      xpath: system/services
      tests:
        - not-exists: telnet
```

SALTSTACK REACTOR CONFIGURATION

```
root@JEDI-cluster-demo:~# salt-run reactor.list
....
|_
  ----------
  jnpr/syslog/*/UI_COMMIT_COMPLETED:
      - /srv/reactor/junos_backup.sls
      - /srv/reactor/check_compliancy.sls
|_
  ----------
  jnpr/compliance/failed:
      - /srv/reactor/create_compliancy_ticket.sls
|_
  ----------
  jnpr/enforce_compliancy/start:
      - /srv/reactor/enforce_compliancy.sls
      - /srv/reactor/update_compliancy_ticket.sls
```



DEMO #1

Commit a permitted configuration change on a junos device

EVENT DRIVEN AUTOMATION

- A human or a process commits a configuration change on a junos device
  - The junos device sends a UI_COMMIT_COMPLETED syslog message to SaltStack
    - The SaltStack junos_syslog engine publishes a 0MQ message
- The reactor component of the master is subscribing to this 0MQ topic
  - So it executes sls files
    - To backup the new configuration file in a git repository
    - To run a configuration compliance test using JSNAPy.
      - As this configuration change is permitted, the compliance test passes
- No other action is automated

COMMIT A PERMITTED CHANGE ON JUNOS

- Commit a permitted configuration change on a junos device:

```
ksator@vsrx# set system login message "welcome to J-EDI demo"

[edit]
ksator@vsrx# show | compare
[edit system login]
+   message "welcome to J-EDI demo";

[edit]
ksator@vsrx# commit and-quit
commit complete
Exiting configuration mode

ksator@vsrx> show system commit
0   2017-11-05 21:58:59 UTC by ksator via cli
```

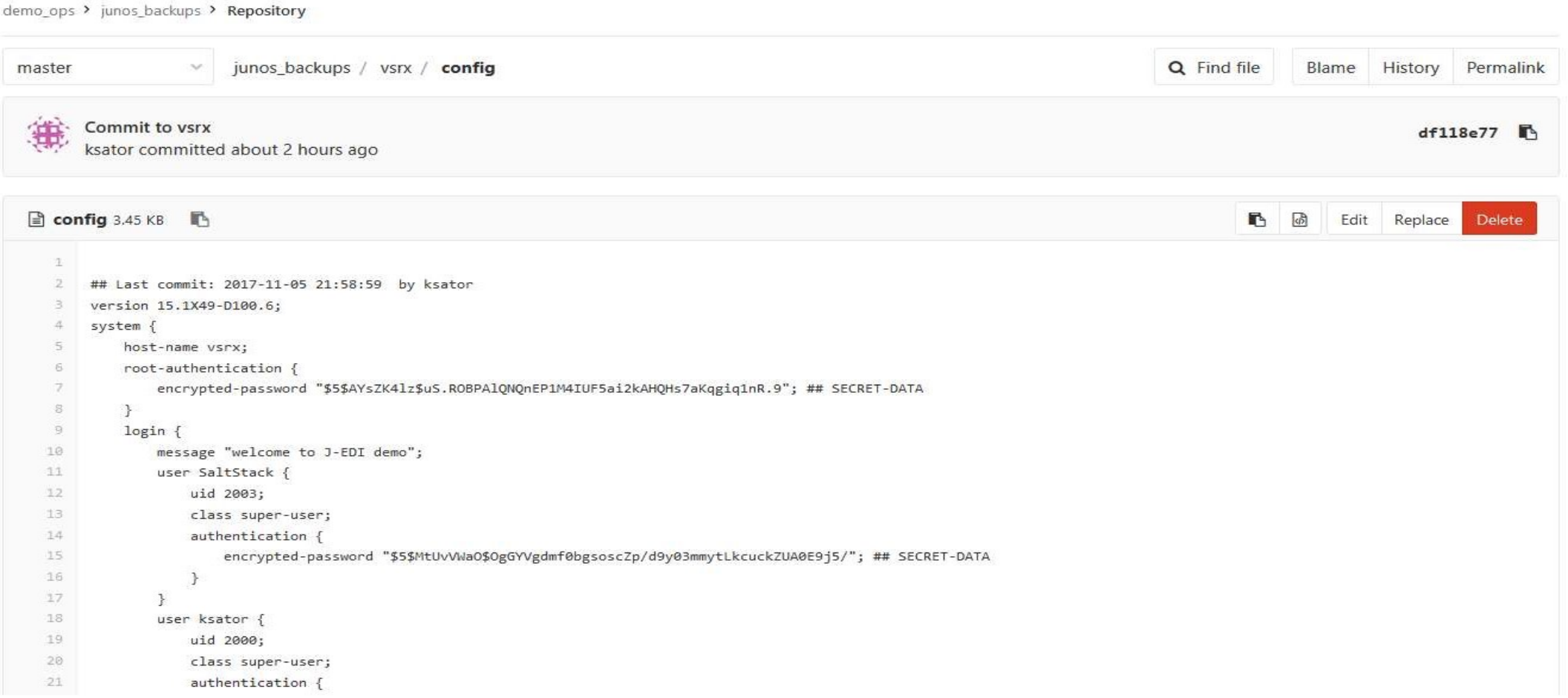
TCPDUMP OUTPUT ON JUNOS_SYSLOG ENGINE

```
root@JEDI-cluster-demo:~# tcpdump -i ens33 port 516 -XX
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on ens33, link-type EN10MB (Ethernet), capture size 262144 bytes
14:00:02.817674 IP 192.168.233.165.syslog > cluster.516: SYSLOG
local7.warning, length: 74
        0x0000:  000c 29fd f54a 000c 2973 7ad1 0800 4500  ..)..J..)sz...E.
        0x0010:  0066 8967 0000 4011 9c5c c0a8 e9a5 c0a8  .f.g..@..\......
        0x0020:  e9cc 0202 0204 0052 68a4 3c31 3838 3e4e  .......Rh.<188>N
        0x0030:  6f76 2020 3620 3030 3a31 313a 3238 2076  ov..6.00:11:28.v
        0x0040:  7372 7820 6d67 645b 3833 3731 325d 3a20  srx.mgd[83712]:.
        0x0050:  5549 5f43 4f4d 4d49 545f 434f 4d50 4c45  UI_COMMIT_COMPLE
        0x0060:  5445 443a 2063 6f6d 6d69 7420 636f 6d70  TED:.commit.comp
        0x0070:  6c65 7465                                lete
```

EVENT PUBLISHED BY JUNOS_SYSLOG SALT ENGINE


```
jnpr/syslog/vsrx/UI_COMMIT_COMPLETED    {
    "_stamp": "2017-11-15T13:00:02.849128",
    "daemon": "mgd",
    "event": "UI_COMMIT_COMPLETED",
    "facility": 23,
    "hostip": "192.168.233.165",
    "hostname": "vsrx",
    "message": "commit complete",
    "pid": "83712",
    "priority": 188,
    "raw": "<188>Nov  6 00:11:28 vsrx mgd[83712]: UI_COMMIT_COMPLETED: commit
complete",
    "severity": 4,
    "timestamp": "2017-11-15 14:00:02"
}
```

ENTIRE CONFIGURATION IS PLACED INTO GITLAB



BACKUP HISTORY INTO GITLAB

- Backup history shows the configuration change details and the timestamp and the user

DEMO #2

Commit a NOT permitted configuration change on a junos device

EVENT DRIVEN AUTOMATION

- A human or a process commits a configuration change on a junos device
  - The junos device sends a UI_COMMIT_COMPLETED syslog message to SaltStack
    - The SaltStack junos_syslog engine publishes a 0MQ message
- The reactor component of the master is subscribing to this 0MQ topic
  - So it executes sls files
    - To backup the new configuration file in a git repository
    - To run a configuration compliance test using JSNAPy.
      - This configuration change is not permitted, so the compliance test fails and a 0MQ message is published
- The reactor component of the master is subscribing to this 0MQ topic
  - So it executes an sls file to create an auto remediation ticket and to publishes a 0MQ message
- The reactor component of the master is subscribing to this 0MQ topic
  - So it executes sls files
    - To enforce the configuration compliancy (auto-remediation) on the device using a SaltStack proxy for Junos (so a new JUNOS commit happens)
    - To update the ticket to track the auto-remediation action

COMMIT A NOT PERMITTED CHANGE ON JUNOS

- Commit a non permitted configuration change on a junos device:

```
[edit]
ksator@vsrx# set system services telnet

[edit]
ksator@vsrx# commit and-quit
commit complete
Exiting configuration mode

ksator@vsrx> show system commit
0   2017-11-06 00:11:27 UTC by ksator via cli
```

BACKUP HISTORY SHOW THE CHANGE DETAILS

Showing 1 changed file ▾ with 2 additions and 1 deletions

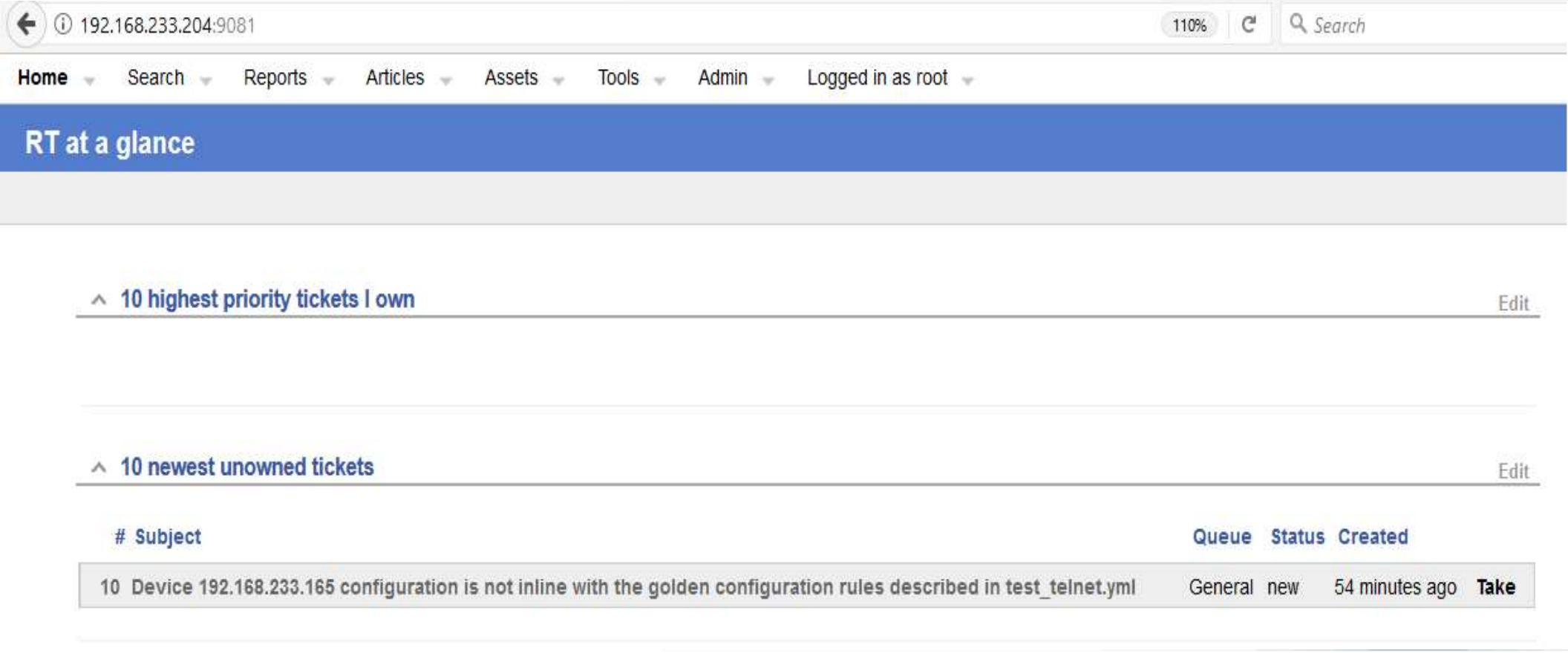
Hide whitespace changes | Inline | Side-by-side

vsrx/config

View file @ 6ae26124

```
1   1
2       - ## Last commit: 2017-11-05 21:58:59  by ksator
    2   + ## Last commit: 2017-11-06 00:11:27  by ksator
3   3     version 15.1X49-D100.6;
4   4     system {
5   5         host-name vsrx;
... ...   @@ -35,6 +35,7 @@ system {
35  35                root-login allow;
36  36                max-sessions-per-connection 32;
37  37            }
    38  +         telnet;
38  39            netconf {
39  40                ssh;
40  41            }
... ...
```

J-EDI CREATES A NEW RT4 TICKET



AUTO REMEDIATION

Showing 1 changed file with 1 additions and 2 deletions

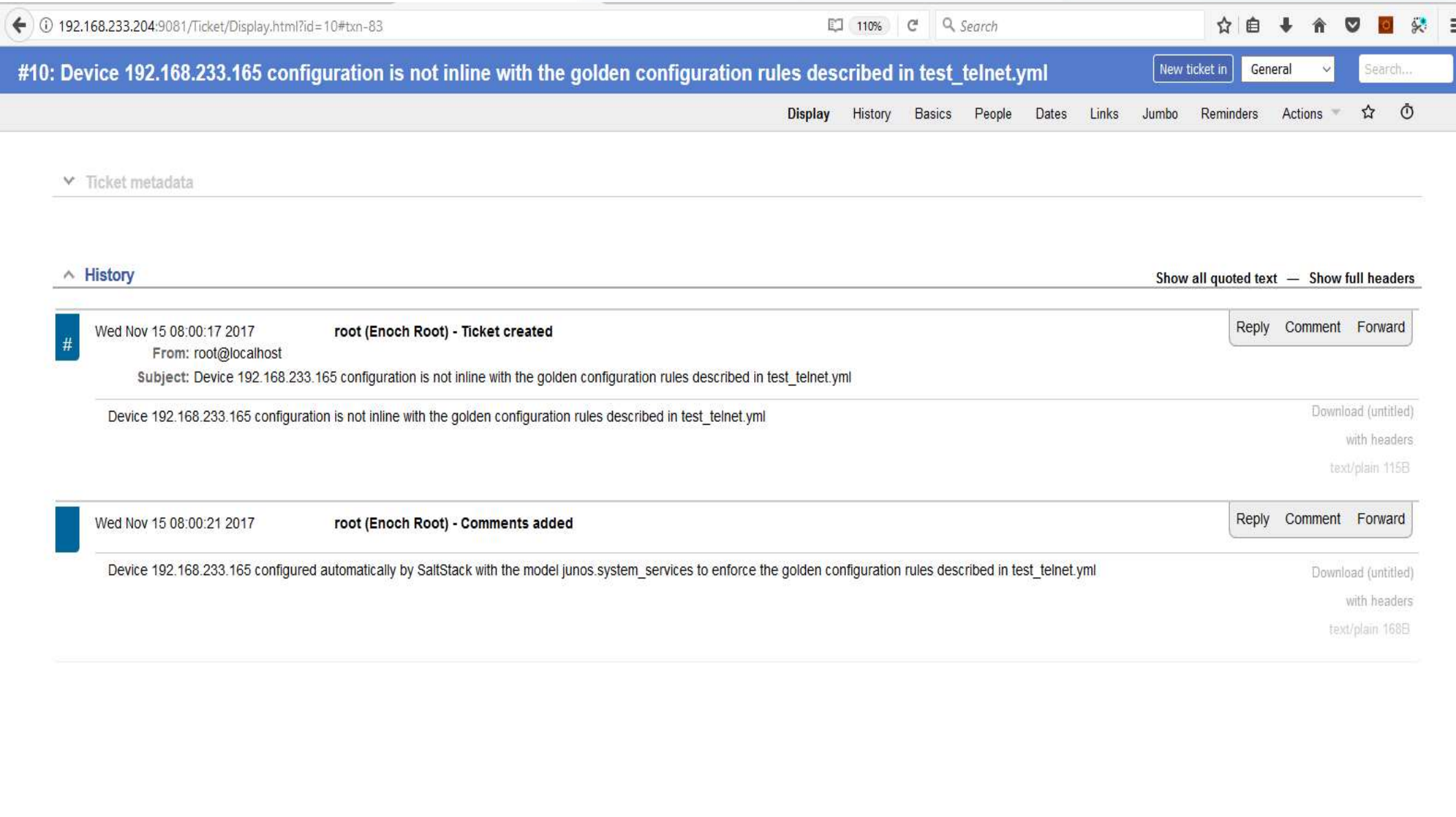
Hide whitespace changes | Inline | Side-by-side

vsrx/config

View file @ 7b097c30

```
1  1
2     - ## Last commit: 2017-11-06 00:11:27  by ksator
   2  + ## Last commit: 2017-11-06 00:11:55  by SaltStack
3  3  version 15.1X49-D100.6;
4  4  system {
5  5      host-name vsrx;
...  ...  @@ -35,7 +35,6 @@ system {
35  35              root-login allow;
36  36              max-sessions-per-connection 32;
37  37          }
38     -        telnet;
39  38          netconf {
40  39              ssh;
41  40          }
...  ...
```

TICKET UPDATE



AUTO REMEDIATION DETAILS ON JUNOS

```
ksator@vsrx> show system commit
0   2017-11-06 00:11:55 UTC by SaltStack via netconf
    configured automatically with SaltStack using the model
system_services due to ticket 10
1   2017-11-06 00:11:27 UTC by ksator via cli
2   2017-11-05 21:58:59 UTC by ksator via cli
```

```
ksator@vsrx> show configuration | compare rollback 1
[edit system services]
-    telnet;

ksator@vsrx> show configuration system services | display set
set system services ssh root-login allow
set system services ssh max-sessions-per-connection 32
set system services netconf ssh
set system services web-management http interface fxp0.0

ksator@vsrx>
```


Thank you