

# SUSE Manager / Uyuni - Valid PTR Record required for SSH-Push Clients

**title:** SUSE Manager / Uyuni - Valid PTR Record required for SSH-Push Clients  
**Date:** 2021-02-23  
**modified:** 2021-02-23  
**tags:** Uyuni, SUSE Manager, SSH, Salt, DNS  
**description:** Clients connected to SUSE Manager / Uyuni need PTR DNS Record  
**category:** Linux  
**slug:** suse-manager-uyuni-valid-ptr-required-for-ssh-push  
**Author:** Dominik Wombacher  
**lang:** en  
**transid:** suse-manager-uyuni-valid-ptr-required-for-ssh-push  
**Status:** published

I'm responsible for a [SUSE Manager](#) Installation, the commercial product based on [Uyuni](#), with almost 900 connected Server.

The recommend connection method is using a Salt Minion, but SSH-Push - based on Salt-SSH - is the agentless alternative. In our Environment, something around 200 Server using this connection type, because there is already a Salt Minion running, attached to a different Salt Master.

## Problem

During the last weeks we faced some issues were the **salt-api** service is going to crash due to an unhandled exception from time to time.

Investigation together with SUSE is still ongoing but so far it seems like, that under rare conditions, a Server with connection method SSH-Push and without valid PTR DNS Record can indeed raise an Exception in the Salt Codebase.

## Background

Function `ip_to_host` located in `/usr/lib/python3.6/site-packages/salt/utls/network.py` is passing the FQDN to Python built-in Function `socket.gethostbyaddr`. This will raise an Exception, handled by `ip_to_host` which then return `None` (Normally that should be a Hostname).

Based on the return value, a new Minion Job for Server `None` will be started and end up in an **unhandled Exception** that cause salt-api to crash as shown in the following error message (Debug Mode need to be enabled for *Salt* as well *Taskomatic*):

`/var/log/salt/api_error`

```
Traceback (most recent call last):
  File "/usr/lib/python3.6/site-packages/salt/netapi/rest_cherrypy/app.py", line 860, in hypermedia_handler
    ret = cherrypy.serving.request._hypermedia_inner_handler(*args, **kwargs)
  File "/usr/lib/python3.6/site-packages/cherrypy/_cpdispatch.py", line 54, in __call__
    return self.callable(*self.args, **self.kwargs)
  File "/usr/lib/python3.6/site-packages/salt/netapi/rest_cherrypy/app.py", line 2140, in POST
    'return': list(self.exec_lowstate()),
  File "/usr/lib/python3.6/site-packages/salt/netapi/rest_cherrypy/app.py", line 1187, in exec_lowstate
    ret = self.api.run(chunk)
  File "/usr/lib/python3.6/site-packages/salt/netapi/__init__.py", line 152, in run
```

```
        return l_fun(*f_call.get('args', ()), **f_call.get('kwargs', {}))
  File "/usr/lib/python3.6/site-packages/salt/netapi/__init__.py", line 218, in ssh
```

```

        return ssh_client.cmd_sync(kwargs)
File "/usr/lib/python3.6/site-packages/salt/client/ssh/client.py", line 159, in cmd_sync
    **kwargs)
File "/usr/lib/python3.6/site-packages/salt/client/ssh/client.py", line 123, in cmd
    for ret in ssh.run_iter(jid=kwargs.get('jid', None)):
File "/usr/lib/python3.6/site-packages/salt/client/ssh/__init__.py", line 696, in run_iter
    self.cache_job(jid, host, ret[host], fun)
File "/usr/lib/python3.6/site-packages/salt/client/ssh/__init__.py", line 720, in cache_job
    'fun': fun})
File "/usr/lib/python3.6/site-packages/salt/returners/local_cache.py", line 147, in returner
    hn_dir = os.path.join(jid_dir, load['id'])
File "/usr/lib64/python3.6/posixpath.py", line 94, in join
    genericpath._check_arg_types('join', a, *p)
File "/usr/lib64/python3.6/genericpath.py", line 149, in _check_arg_types
    (funcname, s.__class__.__name__) from None
TypeError: join() argument must be str or bytes, not 'NoneType'

During handling of the above exception, another exception occurred:

Traceback (most recent call last):
File "/usr/lib/python3.6/site-packages/cherrypy/_cprequest.py", line 638, in respond
    self._do_respond(path_info)
File "/usr/lib/python3.6/site-packages/cherrypy/_cprequest.py", line 697, in _do_respond
    response.body = self.handler()
File "/usr/lib/python3.6/site-packages/cherrypy/lib/encoding.py", line 219, in __call__
    self.body = self.oldhandler(*args, **kwargs)
File "/usr/lib/python3.6/site-packages/salt/netapi/rest_cherrypy/app.py", line 894, in hypermedia_handler
    if cherrypy.config['debug']
File "/usr/lib64/python3.6/traceback.py", line 167, in format_exc
    return ".join(format_exception(*sys.exc_info(), limit=limit, chain=chain))
File "/usr/lib64/python3.6/traceback.py", line 121, in format_exception
    type(value), value, tb, limit=limit).format(chain=chain))
File "/usr/lib64/python3.6/traceback.py", line 498, in __init__
    _seen=_seen)
File "/usr/lib64/python3.6/traceback.py", line 509, in __init__
    capture_locals=capture_locals)
File "/usr/lib64/python3.6/traceback.py", line 338, in extract
    if limit >= 0:

TypeError: '>=' not supported between instances of 'TypeError' and 'int'

```

## */var/log/salt/api*

```

[...]
2021-02-08 01:08:01,229 [salt.utils.network:239 ][DEBUG ][38989] salt.utils.network.ip_to_host('server.example.com') failed: [Errno 0] Resolver Error 0 (no error)
[...]
2021-02-08 01:08:02,801 [salt.loaded.int.returner.local_cache:252 ][DEBUG ][38989] Adding minions for job 20210208000802780236: [None]
[...]
2021-02-08 01:08:02,839 [salt.utils.process:767 ][ERROR ][5340] An un-handled exception from the multiprocessing process 'Process-1:335' was caught:
[...]
2021-02-08 01:08:02,916 [salt.client.ssh:644 ][ERROR ][38989] Target 'None' did not return any data, probably due to an error.
2021-02-08 01:08:02,917 [salt.loaded.int.netapi.rest_cherrypy.app:887 ][DEBUG ][38989] Error while processing request for: /run

```

## */var/log/rhn/rhn\_taskomatic\_daemon.log*

```

2021-02-08 01:08:02,976 [Thread-70313] ERROR com.redhat.rhn.taskomatic.task.SSHPush - com.suse.salt.netapi.
exception.SaltException: Response code: 500
java.lang.RuntimeException: com.suse.salt.netapi.exception.SaltException: Response code: 500
    at com.suse.manager.webui.services.impl.SaltService.callSync(SaltService.java:235)
    at com.suse.manager.webui.services.impl.SaltService.ping(SaltService.java:244)
    at com.redhat.rhn.taskomatic.task.sshpush.SSHPushWorkerSalt.performCheckin(SSHPushWorkerSalt.java:337)
    at com.redhat.rhn.taskomatic.task.sshpush.SSHPushWorkerSalt.lambda$run$0(SSHPushWorkerSalt.java:122)
    at java.base/java.util.Optional.ifPresent(Optional.java:183)
    at com.redhat.rhn.taskomatic.task.sshpush.SSHPushWorkerSalt.run(SSHPushWorkerSalt.java:110)
    at EDU.oswego.cs.dl.util.concurrent.PooledExecutor$Worker.run(PooledExecutor.java:732)
    at java.base/java.lang.Thread.run(Thread.java:834)
Caused by: com.suse.salt.netapi.exception.SaltException: Response code: 500
    at com.suse.salt.netapi.client.impl.HttpAsyncClientImpl.createSaltException(HttpAsyncClientImpl.java:
145)
    at com.suse.salt.netapi.client.impl.HttpAsyncClientImpl.access$000(HttpAsyncClientImpl.java:27)
    at com.suse.salt.netapi.client.impl.HttpAsyncClientImpl$1.completed(HttpAsyncClientImpl.java:121)
    at com.suse.salt.netapi.client.impl.HttpAsyncClientImpl$1.completed(HttpAsyncClientImpl.java:101)
    at org.apache.http.concurrent.BasicFuture.completed(BasicFuture.java:123)

```

```

    at org.apache.http.impl.nio.client.DefaultClientExchangeHandlerImpl.responseCompleted
(DefaultClientExchangeHandlerImpl.java:181)
    at org.apache.http.nio.protocol.HttpAsyncRequestExecutor.processResponse(HttpAsyncRequestExecutor.java:
442)
    at org.apache.http.nio.protocol.HttpAsyncRequestExecutor.inputReady(HttpAsyncRequestExecutor.java:332)

```

```
at org.apache.http.impl.nio.DefaultNHttpClientConnection.consumeInput(DefaultNHttpClientConnection.java:265)
at org.apache.http.impl.nio.client.InternalIODispatch.onInputReady(InternalIODispatch.java:81)
at org.apache.http.impl.nio.client.InternalIODispatch.onInputReady(InternalIODispatch.java:39)
at org.apache.http.impl.nio.reactor.AbstractIODispatch.inputReady(AbstractIODispatch.java:121)
at org.apache.http.impl.nio.reactor.BaseIOReactor.readable(BaseIOReactor.java:162)
at org.apache.http.impl.nio.reactor.AbstractIOReactor.processEvent(AbstractIOReactor.java:337)
at org.apache.http.impl.nio.reactor.AbstractIOReactor.processEvents(AbstractIOReactor.java:315)
at org.apache.http.impl.nio.reactor.AbstractIOReactor.execute(AbstractIOReactor.java:276)
at org.apache.http.impl.nio.reactor.BaseIOReactor.execute(BaseIOReactor.java:104)
at org.apache.http.impl.nio.reactor.AbstractMultiworkerIOReactor$Worker.run
(AbstractMultiworkerIOReactor.java:588)
... 1 more
```

You can quickly check, with two Python based commands, if the reverse lookup is working for a given FQDN. In this Example we assume the FQDN is **server.example.com** and the assigned IP Address is **10.11.12.13**:

```
python3 -c 'import socket; print(socket.gethostbyaddr("server.example.com"))'
python3 -c 'import socket; print(socket.getaddrinfo("server.example.com", 0, 0, 0, 0))'
```

In case everything is fine, both should return something similar as the following:

```
('server.example.com', [], ['10.11.12.13'])
[((<AddressFamily.AF_INET: 2>, <SocketKind.SOCK_STREAM: 1>, 6, '', ('10.11.12.13', 0)),
  (<AddressFamily.AF_INET: 2>, <SocketKind.SOCK_DGRAM: 2>, 17, '', ('10.11.12.13', 0)),
  (<AddressFamily.AF_INET: 2>, <SocketKind.SOCK_RAW: 3>, 0, '', ('10.11.12.13', 0))]
```

Further reading regarding *gethostbyaddr*:  
<https://docs.python.org/3/library/socket.html#socket.gethostbyaddr>

## Solution

I expect that SUSE is going to include a Patch in future SUSE Manager and Uyuni releases were a Server will probably skipped if there is no valid PTR Record. So you should double check in your Environment, that all Systems connected via SSH-Push / Salt-SSH have the necessary DNS Records and can be resolved by their FQDN as well IP Address to avoid potential issues.