

FreeNAS - Bug #34843

Fix "Set of coroutines/Futures is empty" error when querying for invalid service

06/12/2018 01:06 PM - Eric Turgeon

Status: Done	
Priority: No priority	
Assignee: Vladimir Vinogradenko	
Category: Middleware	
Target version: 11.2-BETA1	
Seen in: Master - FreeNAS Nightlies	Needs Merging: No
Severity: Medium	Needs Automation: No
Reason for Closing:	Support Suite Ticket: n/a
Reason for Blocked:	Hardware Configuration:
Needs QA: No	ChangeLog Required: No
Needs Doc: No	

Description

I have found that smartd service does not start returning 200 and just for fun I did try to start smart and return 200 even if the service does not start.

See my lpython log below.

```
In [1]: from functions import POST, DELETE, GET, PUT
```

```
In [2]: GET('/service/?service=smartd')
```

```
Out[2]: <Response [200]>
```

```
In [3]: GET('/service/?service=smartd').text
```

```
Out[3]: '[\n {\n  "id": 18,\n  "service": "smartd",\n  "enable": true,\n  "state": "STOPPED",\n  "pids": []\n }]\n'
```

```
In [4]: POST("/service/start/", {"service": "smartd", "service-control": {"onetime": True}}).text
```

```
Out[4]: 'false'
```

```
In [5]: POST("/service/start/", {"service": "smartd", "service-control": {"onetime": True}})
```

```
Out[5]: <Response [200]>
```

```
In [6]: GET('/service/?service=smartd').text
```

```
Out[6]: '[\n {\n  "id": 18,\n  "service": "smartd",\n  "enable": true,\n  "state": "STOPPED",\n  "pids": []\n }]\n'
```

```
In [7]: POST("/service/start/", {"service": "smart", "service-control": {"onetime": True}})
```

```
Out[7]: <Response [200]>
```

```
In [8]: GET('/service/?service=smartd').text
```

```
Out[8]: '[\n {\n  "id": 18,\n  "service": "smartd",\n  "enable": true,\n  "state": "STOPPED",\n  "pids": []\n }]\n'
```

```
In [9]: GET('/service/?service=smart').text
```

```
Out[9]: '<h1>500 Internal Server Error</h1><br><h2>Traceback:</h2>\n
```

```
&lt;pre>Traceback (most recent call last):\n
```

```
File &quot;/usr/local/lib/python3.6/site-packages/aiohttp/web_protocol.py&quot;;, line 38
```

```
1, in start\n    resp = await self._request_handler(request)\n
```

```
File &quot;/usr/local/lib/python3.6/asyncio/coroutines.py&quot;;, line 110, in __next__\n
```

```
    return self.gen.send(None)\n
```

```
File &quot;/usr/local/lib/python3.6/site-packages/aiohttp/web_app.py&quot;;, line 322, in
```

```
_handle\n    resp = await handler(request)\n
```

```
File &quot;/usr/local/lib/python3.6/asyncio/coroutines.py&quot;;, line 110, in __next__\n
```

```
return self.gen.send(None)\n
File &quot;/usr/local/lib/python3.6/site-packages/aiohttp/web_middlewares.py&quot;;, line
88, in impl\n    return await handler(request)\n
File &quot;/usr/local/lib/python3.6/asyncio/coroutines.py&quot;;, line 110, in __next__\n
    return self.gen.send(None)\n
File &quot;/usr/local/lib/python3.6/site-packages/aiohttp/web_middlewares.py&quot;;, line
78, in impl\n    return await handler(request)\n
File &quot;/usr/local/lib/python3.6/asyncio/coroutines.py&quot;;, line 110, in __next__\n
    return self.gen.send(None)\n
File &quot;/usr/local/lib/python3.6/site-packages/middlewared/restful.py&quot;;, line 365
, in on_method\n    return await do(method, req, resp, *args, **kwargs)\n
File &quot;/usr/local/lib/python3.6/asyncio/coroutines.py&quot;;, line 110, in __next__\n
    return self.gen.send(None)\n
File &quot;/usr/local/lib/python3.6/site-packages/middlewared/restful.py&quot;;, line 497
, in do\n    result = await self.middleware.call(methodname, *method_args)\n
File &quot;/usr/local/lib/python3.6/asyncio/coroutines.py&quot;;, line 110, in __next__\n
    return self.gen.send(None)\n
File &quot;/usr/local/lib/python3.6/site-packages/middlewared/main.py&quot;;, line 1018,
in call\n    return await self._call(name, serviceobj, methodobj, params, pipes=pipes)\n
File &quot;/usr/local/lib/python3.6/asyncio/coroutines.py&quot;;, line 110, in __next__\n
    return self.gen.send(None)\n
File &quot;/usr/local/lib/python3.6/site-packages/middlewared/main.py&quot;;, line 953, i
n _call\n    return await methodobj(*args)\n
File &quot;/usr/local/lib/python3.6/asyncio/coroutines.py&quot;;, line 110, in __next__\n
    return self.gen.send(None)\n
File &quot;/usr/local/lib/python3.6/site-packages/middlewared/schema.py&quot;;, line 646,
in nf\n    return await f(*args, **kwargs)\n
File &quot;/usr/local/lib/python3.6/asyncio/coroutines.py&quot;;, line 110, in __next__\n
    return self.gen.send(None)\n
File &quot;/usr/local/lib/python3.6/site-packages/middlewared/plugins/service.py&quot;;,
line 107, in query\n    await asyncio.wait(list(jobs.keys()), timeout=15)\n
File &quot;/usr/local/lib/python3.6/asyncio/tasks.py&quot;;, line 304, in wait\n
    raise ValueError(&#x27;Set of coroutines/Futures is empty.&#x27;)\n
ValueError: Set of coroutines/Futures is empty.\n&lt;/pre&gt;
```

Associated revisions

Revision 50eebdbc - 06/13/2018 11:50 AM - Vladimir Vinogradenko

fix(service): Fix "Set of coroutines/Futures is empty" when querying for invalid service

Ticket: #34843

Revision 05159630 - 06/22/2018 12:27 PM - Vladimir Vinogradenko

fix(service): Fix "Set of coroutines/Futures is empty" when querying for invalid service

Ticket: #34843

Revision a37ccee3 - 08/22/2018 01:02 PM - Vladimir Vinogradenko

fix(service): Fix "Set of coroutines/Futures is empty" when querying for invalid service

Ticket: #34843

(cherry picked from commit 50eebdbc06ba26a1c3d88e9f4586d01b1660559e)

History

#1 - 06/12/2018 06:51 PM - Dru Lavigne

- Assignee changed from Release Council to William Grzybowski

#2 - 06/13/2018 05:04 AM - William Grzybowski

- Assignee changed from William Grzybowski to Vladimir Vinogradenko
- Severity changed from New to Medium
- Needs Merging changed from Yes to No

#3 - 06/13/2018 05:17 AM - William Grzybowski

- Target version changed from Backlog to 11.2-RC2

#4 - 06/13/2018 11:59 AM - Vladimir Vinogradenko

```
In [4]: POST("/service/start/", {"service": "smartd", "service-control": {"onetime": True}}).text
Out[4]: 'false'
```

```
In [5]: POST("/service/start/", {"service": "smartd", "service-control": {"onetime": True}})
Out[5]: <Response [200]>
```

It is supposed to be this way.

Exception was fixed.

#5 - 06/13/2018 11:59 AM - Vladimir Vinogradenko

- Status changed from Unscreened to Ready for Testing

#6 - 06/13/2018 12:02 PM - Dru Lavigne

- Subject changed from REST API 2.0 post /service/start/ failed to start smartd with out error to Fix "Set of coroutines/Futures is empty" error when querying for invalid service
- Target version changed from 11.2-RC2 to 11.2-BETA1
- Seen in changed from 11.2-RC2 to Master - FreeNAS Nightlies

Master PR: <https://github.com/freenas/freenas/pull/1359>

#7 - 06/15/2018 05:44 AM - Eric Turgeon

On the latest build this morning it is not fixed.

```
===== test session starts =====
=====
platform freebsd12 -- Python 3.6.5, pytest-3.4.2, py-1.5.3, pluggy-0.6.0 -- /usr/local/bin/python3.6
cachedir: .pytest_cache
rootdir: /usr/home/ericbsd/projects/freenas/freenas/tests, inifile:
collected 8 items

api2/smart.py::test_01_create_a_new_smartttest PASSED
    [ 12%]
api2/smart.py::test_02_check_that_API_reports_new_smartttest PASSED
    [ 25%]
api2/smart.py::test_03_update_smartttest PASSED
    [ 37%]
api2/smart.py::test_04_delete_smartttest PASSED
    [ 50%]
api2/smart.py::test_05_enable_smartd_service_at_boot PASSED
    [ 62%]
```

```
api2/smart.py::test_06_lock_smartd_service_at_boot PASSED
    [ 75%]
api2/smart.py::test_07_starting_smartd_service PASSED
    [ 87%]
api2/smart.py::test_08_checking_to_see_if_smartd_service_is_running FAILED
    [100%]
```

```
----- generated xml file: /usr/home/ericbsd/projects/freenas/freenas/tests/results/smart_tests_result.xml -----
```

```
===== FAILURES =====
_____ test_08_checking_to_see_if_smartd_service_is_running _____
```

```
def test_08_checking_to_see_if_smartd_service_is_running():
    results = GET('/service/?service=smartd')
    assert results.json()[0]["state"] == "RUNNING", results.text
E     AssertionError: [
E         {
E           "id": 18,
E           "service": "smartd",
E           "enable": true,
E           "state": "STOPPED",
E           "pids": []
E         }
E     ]
E     assert 'STOPPED' == 'RUNNING'
E     - STOPPED
E     + RUNNING
```

```
api2/smart.py:82: AssertionError
===== 1 failed, 7 passed in 22.46 seconds =====
```

Trying to start the service return 200 but failed to run, the state still STOPPED.

```
In [3]: GET('/service/?service=smartd').text
Out[3]: '[\n {\n   "id": 18,\n   "service": "smartd",\n   "enable": true,\n   "state": "STOPPED",\n   "pids": []\n }]'

In [4]: POST("/service/start/", {"service": "smartd", "service-control": {"onetime": True}})
Out[4]: <Response [200]>

In [5]: GET('/service/?service=smartd').text
Out[5]: '[\n {\n   "id": 18,\n   "service": "smartd",\n   "enable": true,\n   "state": "STOPPED",\n   "pids": []\n }]'
```

Service that does not exist should return an error code.

```
In [6]: POST("/service/start/", {"service": "smart", "service-control": {"onetime": True}})
Out[6]: <Response [200]>

In [7]: GET('/service/?service=smart').text
Out[7]: '[]'

POST("/service/start/", {"service": "dead", "service-control": {"onetime": True}})
Out[8]: <Response [200]>
```

#8 - 06/15/2018 05:45 AM - Dru Lavigne

- Status changed from Ready for Testing to Failed Testing

#9 - 06/15/2018 06:44 AM - Eric Turgeon

link to the latest Incremental build

<https://builds.ixsystems.com/jenkins/view/Builds/view/All/job/FreeNAS%20-%20Master%20-%20Incremental%20Build/3206/artifact/artifacts/iso/FreeNAS-11.2-MASTER-201806150906-91108b28e.iso>

#10 - 06/15/2018 07:48 AM - Eric Turgeon

Step to reproduce the above with the iso linked above.

The iso is placed into the freenas/test/iso folder
It is run under a BHYVE build by ixautomation.

```
sudo ixautomation --run api2-tests --systype freenas --keep-alive
```

When it is done the result of IPython are done in the freenas/tests folder.

```
cd path/to/freenas/tests  
sudo ipython-3.6
```

IT is done on a TrueOS machine.

#11 - 06/15/2018 12:46 PM - Eric Turgeon

So as I was told the service does not start on real ATA hardware, so only a CallError remains on this issue.

#12 - 06/15/2018 01:02 PM - William Grzybowski

We can't add a CallError, we use it for arbitrary services that are not defined in the database.

I am thinking we could use DEVICESCAN in smartd.conf if no disks support SMART.

#13 - 06/17/2018 08:12 AM - Vladimir Vinogradenko

We can't add a CallError, we use it for arbitrary services that are not defined in the database.

So POST("/service/start", {"service": "dead", "service-control": {"onetime": True}}) should just issue service dead start and return false (which it does)?

I am thinking we could use DEVICESTAN in smartd.conf if no disks support SMART.

```
Jun 17 08:05:59 freenas smartd[34183]: smartd 6.6 2017-11-05 r4594 [FreeBSD 11.1-STABLE amd64] (local build)
Jun 17 08:05:59 freenas smartd[34183]: Copyright (C) 2002-17, Bruce Allen, Christian Franke, www.smartmontools.org
Jun 17 08:05:59 freenas smartd[34183]: Opened configuration file /usr/local/etc/smartd.conf
Jun 17 08:05:59 freenas smartd[34183]: Drive: DEVICESTAN, implied '-a' Directive on line 1 of file /usr/local/etc/smartd.conf
Jun 17 08:05:59 freenas smartd[34183]: Configuration file /usr/local/etc/smartd.conf was parsed, found DEVICESTAN, scanning devices
Jun 17 08:05:59 freenas smartd[34183]: Device: /dev/ada0, opened
Jun 17 08:05:59 freenas smartd[34183]: Device: /dev/ada0, VBOX HARDDISK, S/N:VB574cb89c-65bf0f89, FW:1.0, 4.29 GB
Jun 17 08:05:59 freenas smartd[34183]: Device: /dev/ada0, not found in smartd database.
Jun 17 08:05:59 freenas smartd[34183]: Device: /dev/ada0, lacks SMART capability
Jun 17 08:05:59 freenas smartd[34183]: Device: /dev/ada0, to proceed anyway, use '-T permissive' Directive.
Jun 17 08:05:59 freenas smartd[34183]: Device: /dev/ada1, opened
Jun 17 08:05:59 freenas smartd[34183]: Device: /dev/ada1, VBOX HARDDISK, S/N:VB9f605184-8f6d0e32, FW:1.0, 4.29 GB
Jun 17 08:05:59 freenas smartd[34183]: Device: /dev/ada1, not found in smartd database.
Jun 17 08:05:59 freenas smartd[34183]: Device: /dev/ada1, lacks SMART capability
Jun 17 08:05:59 freenas smartd[34183]: Device: /dev/ada1, to proceed anyway, use '-T permissive' Directive.
Jun 17 08:05:59 freenas smartd[34183]: Device: /dev/cd0, opened
Jun 17 08:05:59 freenas smartd[34183]: Device: /dev/cd0, packet devices [this device CD/DVD] not SMART capable
Jun 17 08:05:59 freenas smartd[34183]: Unable to monitor any SMART enabled devices. Try debug (-d) option. Exiting...
```

It will still exit. The only way to prevent it from exiting is passing -q never to daemon which will require updating rc.conf depending on if we have found any SMART-enabled devices. In my opinion, it is not worth it.

#14 - 06/18/2018 06:08 AM - Dru Lavigne

- Status changed from *Failed Testing* to *Ready for Testing*

#15 - 06/26/2018 07:41 AM - Vladimir Vinogradenko

It's been decided that we won't fix SMART on VM issue.

#16 - 06/26/2018 07:42 AM - Vladimir Vinogradenko

This bug is only visible when doing manual API calls. No QA is needed here.

#17 - 06/26/2018 08:30 AM - Eric Turgeon

- Needs QA changed from *Yes* to *No*

#18 - 06/26/2018 09:01 AM - Bonnie Follweiler

- Status changed from *Ready for Testing* to *Passed Testing*

#19 - 06/26/2018 11:48 AM - Dru Lavigne

- Status changed from *Passed Testing* to *Done*