

mgr - Bug #38047

Bug # 38094 (Resolved): mgr: crash list

mgr: segfault, dashboard, PyModule::get_typed_option_value > PyErr_Restore

01/25/2019 12:49 PM - Volker Theile

Status:	Need Review	Start date:	01/25/2019
Priority:	Normal	Due date:	
Assignee:		% Done:	0%
Category:	ceph-mgr	Estimated time:	0.00 hour
Target version:	v14.0.0	Reviewed:	
Source:	Development	Affected Versions:	
Tags:		ceph-qa-suite:	
Backport:		Pull request ID:	
Regression:	No		
Severity:	3 - minor		

Description

See source <https://github.com/ceph/ceph/blob/master/src/mgr/PyModule.cc#L622>.

```
-20> 2019-01-25 12:34:05.484 7fbcc3e51700 20 mgr Gil Switched to new thread state 0x56017ba60d0
0
-19> 2019-01-25 12:34:05.484 7fbcc3e51700 20 mgr ~Gil Destroying new thread state 0x56017ba60d0
0
-18> 2019-01-25 12:34:05.484 7fbcc3e51700 20 mgr Gil Switched to new thread state 0x56017ba60d0
0
-17> 2019-01-25 12:34:05.484 7fbcc3e51700 20 mgr ~Gil Destroying new thread state 0x56017ba60d0
0
-16> 2019-01-25 12:34:05.716 7fbcc3e51700 20 mgr[dashboard] [::ffff:127.0.0.1:58398] [GET] [admin] /api/mgr/module/telemetry
-15> 2019-01-25 12:34:05.716 7fbcc3e51700 4 mgr get_config get_config key: mgr/dashboard/AUDIT_API_ENABLED
-14> 2019-01-25 12:34:05.716 7fbcc3e51700 4 mgr get_typed_config get_typed_config AUDIT_API_ENABLED not found
-13> 2019-01-25 12:34:05.716 7fbcc3e51700 20 mgr[dashboard] AMT: token: eyJ0eXAI0iJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJpc3MiOiJjZXB0LWRhc2hib2FyZCIsImp0aSI6IjJhYmQwYjYxLThiYjgtNDM0MiliMzY2LWFjMTFlNzEzN2U3YSIsImV4cCI6MTU0ODQ0MTY4NCwiaWF0IjoxNTQ4NDYyODg0LCJ1c2VybmFtZSI6ImFkbWwLRoFaNxCsgU_9mRpA76kE
-12> 2019-01-25 12:34:05.716 7fbcc3e51700 4 mgr get_store get_store key: mgr/dashboard/jwt_token_black_list
-11> 2019-01-25 12:34:05.716 7fbcc3e51700 4 ceph_store_get jwt_token_black_list not found
-10> 2019-01-25 12:34:05.716 7fbcc3e51700 20 mgr[dashboard] AMT: checking authorization...
-9> 2019-01-25 12:34:05.720 7fbcc3e51700 20 mgr[dashboard] AMT: checking '['read']' access to 'config-opt' scope
-8> 2019-01-25 12:34:05.724 7fbcc3e51700 4 mgr get_config get_config key: mgr/telemetry/contact
-7> 2019-01-25 12:34:05.728 7fbcc3e51700 10 mgr get_typed_config get_typed_config contact found: test@test.com
-6> 2019-01-25 12:34:05.728 7fbcc3e51700 4 mgr get_config get_config key: mgr/telemetry/description
-5> 2019-01-25 12:34:05.728 7fbcc3e51700 10 mgr get_typed_config get_typed_config description found:
-4> 2019-01-25 12:34:05.728 7fbcc3e51700 4 mgr get_config get_config key: mgr/telemetry/enabled
-3> 2019-01-25 12:34:05.728 7fbcc3e51700 10 mgr get_typed_config get_typed_config enabled found: True
-2> 2019-01-25 12:34:05.728 7fbcc3e51700 4 mgr get_config get_config key: mgr/telemetry/interval
-1> 2019-01-25 12:34:05.728 7fbcc3e51700 10 mgr get_typed_config get_typed_config interval found: None
0> 2019-01-25 12:34:05.764 7fbcc3e51700 -1 *** Caught signal (Segmentation fault) **
```

in thread 7fbcc3e51700 thread_name:dashboard

```
ceph version 14.0.1-2876-gad3cf4810f (ad3cf4810f969c167f5763ac9025dbbb2045d5ef) nautilus (dev)
1: (()+0x9f5024) [0x5601767bb024]
2: (()+0x11f70) [0x7fbce43d0f70]
3: (PyErr_Restore()+0x24) [0x7fbce5113ef4]
4: (PyErr_FormatV()+0x12) [0x7fbce5132892]
5: (PyErr_Format()+0x94) [0x7fbce5132964]
6: (PyLong_FromString()+0xbe2) [0x7fbce5159392]
7: (PyInt_FromString(char const*, char**, int)+0x29) [0x560176613547]
8: (PyModule::get_typed_option_value(std::__cxx11::basic_string<char, std::char_traits<char>, std::allocator<char> > const&, std::__cxx11::basic_string<char, std::char_traits<char>, std::allocator<char> > const&)+0xce) [0x560176611c10]
9: (ActivePyModules::get_typed_config(std::__cxx11::basic_string<char, std::char_traits<char>, std::allocator<char> > const&, std::__cxx11::basic_string<char, std::char_traits<char>, std::allocator<char> > const&)+0x33d) [0x5601764e8fd9]
10: (()+0x743826) [0x560176509826]
11: (_PyCFFunction_FastCallDict()+0x92) [0x7fbce519fc72]
12: (()+0x184acd) [0x7fbce51adacd]
13: (_PyEval_EvalFrameDefault()+0x30a) [0x7fbce51d949a]
14: (()+0x1079af) [0x7fbce51309af]
15: (()+0x13dc70) [0x7fbce5166c70]
16: (()+0x184bb2) [0x7fbce51adbb2]
17: (_PyEval_EvalFrameDefault()+0x30a) [0x7fbce51d949a]
18: (PyEval_EvalCodeEx()+0x317) [0x7fbce51aebd7]
19: (()+0x186bde) [0x7fbce51afbde]
20: (PyObject_Call()+0x43) [0x7fbce5132563]
21: (_PyEval_EvalFrameDefault()+0x1a5d) [0x7fbce51dabed]
22: (PyEval_EvalCodeEx()+0x317) [0x7fbce51aebd7]
23: (()+0x186bde) [0x7fbce51afbde]
24: (PyObject_Call()+0x43) [0x7fbce5132563]
25: (_PyEval_EvalFrameDefault()+0x1a5d) [0x7fbce51dabed]
26: (()+0x1079af) [0x7fbce51309af]
27: (_PyFunction_FastCallDict()+0x53d) [0x7fbce5131bed]
28: (_PyObject_FastCallDict()+0x31e) [0x7fbce513218e]
29: (_PyObject_Call_Prepend()+0x60) [0x7fbce513d600]
30: (PyObject_Call()+0x43) [0x7fbce5132563]
31: (_PyEval_EvalFrameDefault()+0x1a5d) [0x7fbce51dabed]
32: (_PyFunction_FastCallDict()+0x12a) [0x7fbce51317da]
33: (_PyObject_FastCallDict()+0x31e) [0x7fbce513218e]
34: (_PyObject_Call_Prepend()+0x60) [0x7fbce513d600]
35: (PyObject_Call()+0x43) [0x7fbce5132563]
36: (()+0x17ce25) [0x7fbce51a5e25]
37: (PyObject_Call()+0x43) [0x7fbce5132563]
38: (_PyEval_EvalFrameDefault()+0x1a5d) [0x7fbce51dabed]
39: (PyEval_EvalCodeEx()+0x317) [0x7fbce51aebd7]
40: (()+0x186bde) [0x7fbce51afbde]
41: (PyObject_Call()+0x43) [0x7fbce5132563]
42: (_PyEval_EvalFrameDefault()+0x1a5d) [0x7fbce51dabed]
43: (PyEval_EvalCodeEx()+0x317) [0x7fbce51aebd7]
44: (()+0x186bde) [0x7fbce51afbde]
45: (PyObject_Call()+0x43) [0x7fbce5132563]
46: (_PyEval_EvalFrameDefault()+0x1a5d) [0x7fbce51dabed]
47: (()+0x1079af) [0x7fbce51309af]
48: (_PyFunction_FastCallDict()+0x318) [0x7fbce51319c8]
49: (_PyObject_FastCallDict()+0x31e) [0x7fbce513218e]
50: (_PyObject_Call_Prepend()+0x60) [0x7fbce513d600]
51: (PyObject_Call()+0x43) [0x7fbce5132563]
52: (()+0x17ce25) [0x7fbce51a5e25]
53: (_PyObject_FastCallDict()+0x8b) [0x7fbce5131efb]
54: (()+0x184c31) [0x7fbce51adc31]
55: (_PyEval_EvalFrameDefault()+0x30a) [0x7fbce51d949a]
56: (()+0x13daba) [0x7fbce5166aba]
57: (()+0x184bb2) [0x7fbce51adbb2]
58: (_PyEval_EvalFrameDefault()+0x30a) [0x7fbce51d949a]
59: (()+0x13daba) [0x7fbce5166aba]
60: (()+0x184bb2) [0x7fbce51adbb2]
```

```
61: (_PyEval_EvalFrameDefault()+0x30a) [0x7fbce51d949a]
62: (()+0x13daba) [0x7fbce5166aba]
63: (()+0x184bb2) [0x7fbce51adbb2]
64: (_PyEval_EvalFrameDefault()+0x30a) [0x7fbce51d949a]
65: (_PyFunction_FastCallDict()+0x12a) [0x7fbce51317da]
66: (_PyObject_FastCallDict()+0x31e) [0x7fbce513218e]
67: (_PyObject_Call_Prepend()+0x60) [0x7fbce513d600]
68: (PyObject_Call()+0x43) [0x7fbce5132563]
69: (()+0x17b2a6) [0x7fbce51a42a6]
70: (()+0x177172) [0x7fbce51a0172]
71: (_PyObject_FastCallDict()+0x8b) [0x7fbce5131efb]
72: (()+0x184c31) [0x7fbce51adc31]
73: (_PyEval_EvalFrameDefault()+0x30a) [0x7fbce51d949a]
74: (()+0x13daba) [0x7fbce5166aba]
75: (()+0x184bb2) [0x7fbce51adbb2]
76: (_PyEval_EvalFrameDefault()+0x30a) [0x7fbce51d949a]
77: (_PyFunction_FastCallDict()+0x12a) [0x7fbce51317da]
78: (_PyObject_FastCallDict()+0x31e) [0x7fbce513218e]
79: (_PyObject_Call_Prepend()+0x60) [0x7fbce513d600]
80: (PyObject_Call()+0x43) [0x7fbce5132563]
81: (()+0x17ce25) [0x7fbce51a5e25]
82: (PyObject_Call()+0x43) [0x7fbce5132563]
83: (_PyEval_EvalFrameDefault()+0x1a5d) [0x7fbce51dabed]
84: (()+0x1079af) [0x7fbce51309af]
85: (()+0x13dc70) [0x7fbce5166c70]
86: (()+0x184bb2) [0x7fbce51adbb2]
87: (_PyEval_EvalFrameDefault()+0x30a) [0x7fbce51d949a]
88: (_PyFunction_FastCallDict()+0x12a) [0x7fbce51317da]
89: (_PyObject_FastCallDict()+0x31e) [0x7fbce513218e]
90: (_PyObject_Call_Prepend()+0x60) [0x7fbce513d600]
91: (PyObject_Call()+0x43) [0x7fbce5132563]
92: (()+0x17b2a6) [0x7fbce51a42a6]
93: (()+0x177172) [0x7fbce51a0172]
94: (_PyObject_FastCallDict()+0x8b) [0x7fbce5131efb]
95: (()+0x184c31) [0x7fbce51adc31]
96: (_PyEval_EvalFrameDefault()+0x30a) [0x7fbce51d949a]
97: (_PyFunction_FastCallDict()+0x12a) [0x7fbce51317da]
98: (_PyObject_FastCallDict()+0x31e) [0x7fbce513218e]
99: (_PyObject_Call_Prepend()+0x60) [0x7fbce513d600]
NOTE: a copy of the executable, or `objdump -rDS <executable>` is needed to interpret this.
```

--- logging levels ---

```
0/ 5 none
0/ 1 lockdep
0/ 1 context
1/ 1 crush
1/ 5 mds
1/ 5 mds_balancer
1/ 5 mds_locker
1/ 5 mds_log
1/ 5 mds_log_expire
1/ 5 mds_migrator
0/ 1 buffer
0/ 1 timer
0/ 1 filer
0/ 1 striper
0/ 1 objecter
0/ 5 rados
0/ 5 rbd
0/ 5 rbd_mirror
0/ 5 rbd_replay
0/ 5 journaler
0/ 5 objectcacher
0/ 5 client
1/ 5 osd
0/ 5 optracker
0/ 5 objclass
```

```
1/ 3 filestore
1/ 3 journal
1/ 1 ms
20/20 mon
20/20 monc
1/ 5 paxos
0/ 5 tp
1/ 5 auth
1/ 5 crypto
1/ 1 finisher
1/ 1 reserver
1/ 5 heartbeatmap
1/ 5 perfcounter
1/ 5 rgw
1/ 5 rgw_sync
1/10 civetweb
1/ 5 javaclient
1/ 5 asok
1/ 1 throttle
0/ 0 refs
1/ 5 xio
1/ 5 compressor
1/ 5 bluestore
1/ 5 bluefs
1/ 3 bdev
1/ 5 kstore
4/ 5 rocksdb
4/ 5 leveldb
4/ 5 memdb
1/ 5 kinetic
1/ 5 fuse
20/20 mgr
1/ 5 mgrc
1/ 5 dpdk
1/ 5 eventtrace
-2/-2 (syslog threshold)
-1/-1 (stderr threshold)
max_recent      10000
max_new         1000
log_file /ceph/build/out/mgr.x.log
--- end dump of recent events ---
```

Related issues:

Related to mgr - Bug #38026: mgr: segfault, restful, PyModuleRunner::serve() ...

Resolved

01/23/2019

History

#1 - 01/25/2019 12:50 PM - Volker Theile

- Related to Bug #38026: mgr: segfault, restful, PyModuleRunner::serve() > PyString_FromString() > PyObject_Malloc() added

#2 - 01/25/2019 02:05 PM - Volker Theile

- Description updated

#3 - 01/25/2019 02:10 PM - Volker Theile

- Description updated

#4 - 01/30/2019 12:54 PM - Ernesto Puerta

- Parent task set to #38094

#5 - 02/12/2019 05:30 PM - Ernesto Puerta

- Subject changed from mgr: segv somewhere in PyModule::get_typed_option_value to mgr: segfault, dashboard, PyModule::get_typed_option_value

> *PyErr_Restore*

#6 - 02/15/2019 01:28 PM - Ernesto Puerta

- *Status changed from New to Need Review*

Potentially fixed by <https://github.com/ceph/ceph/pull/26445>