Foreman - Bug #14012

Errors on hosts status, configuration status is out of sync

03/02/2016 09:48 AM - Alexandre Barth

Status:	Closed		
Priority:	Normal		
Assignee:	Marek Hulán		
Category:	Reporting		
Target version:	1.11.0		
Difficulty:		**Fixed in Releases:**	
Triaged:		**Found in Releases:**	1.11.0
Bugzilla link:		**Red Hat JIRA:**	
Pull request:			

Description

I don't know why but with foreman 1.11, all hosts Configuration status become Error or Out of Sync.
This morning all my hosts had a green status icon(about 100 hosts), and now they have all became red or orange with tooltip Configuration Error or Configuration Out of Sync.
Puppet reports still incoming every 30 minutes with no errors.

Related issues:

Related to Foreman - Refactor #13883: Host::Managed.last_report_object associ...	**Closed**	**02/24/2016**

History

#1 - 03/02/2016 09:56 AM - Dominic Cleal

- Subject changed from Errors on hosts status to Errors on hosts status, configuration status is out of sync

- Category set to Reporting

- translation missing: en.field_release set to 71

Could you retrieve a bit of debug please from Foreman for one of the affected hosts? Run foreman-rake console and substitute puppet.example.com in the lines below for a hostname:

1. Host.find_by_name('puppet.example.com').get_status(HostStatus::ConfigurationStatus)
2. Host.find_by_name('puppet.example.com').last_report_object.attributes
3. Setting[:outofsync_interval]
4. Setting[:puppet_interval]

Thanks.

#2 - 03/02/2016 11:08 AM - Alexandre Barth

```
Loading production environment (Rails 4.1.5)

irb(main):001:0> Host.find_by_name('foo.com').get_status(HostStatus::ConfigurationStatus)

=> #<HostStatus::ConfigurationStatus id: 4, type: "HostStatus::ConfigurationStatus", status: 12288, host_id: 4, reported_at: "2016-03-02 11:34:16">

irb(main):002:0> Host.find_by_name('foo.com').last_report_object.attributes

=> {"id"=>6, "host_id"=>4, "reported_at"=>Wed, 02 Mar 2016 11:34:16 UTC +00:00, "created_at"=>Wed, 02 Mar 2016 11:35:27 UTC +00:00, "updated_at"=>Wed, 02 Mar 2016 11:35:27 UTC +00:00, "status"=>12288, "metrics"=>"--- !ruby/hash:ActionController::Parameters {}\n", "type"=>"ConfigReport"}

irb(main):003:0> Setting[:outofsync_interval]

=> 5

irb(main):004:0> Setting[:puppet_interval]

=> 30

irb(main):005:0>
```

#3 - 03/02/2016 11:11 AM - Dominic Cleal

Thanks. The last reported times there are today at 11:34 UTC, do you know if that's correct? It's 16:11 UTC as I write this, suggesting the host isn't reporting in correctly, or there's a bug in how we're storing the times if either the host or Foreman is running in a non-UTC timezone.

#4 - 03/02/2016 11:27 AM - Alexandre Barth

Currently it's 17:26 CET (16:26 UTC) on my foreman server

#5 - 03/02/2016 11:39 AM - Alexandre Barth

In foreman interface, for this host i have a report with 7 minutes ago in the last report column, and other reports every 30 minutes before that.

But still 11:34 UTC in foreman-rake console

#6 - 03/03/2016 03:40 AM - Dominic Cleal

Was that time when you upgraded, or just a random time? Could you try a couple more steps please:

1. Host.find_by_name('puppet.example.com').get_status(HostStatus::ConfigurationStatus)
2. Host.find_by_name('puppet.example.com').get_status(HostStatus::ConfigurationStatus).relevant?
3. Host.find_by_name('puppet.example.com').refresh_statuses
4. Host.find_by_name('puppet.example.com').get_status(HostStatus::ConfigurationStatus)

#7 - 03/03/2016 04:24 AM - Alexandre Barth

11:34 UTC is the time of the first report generated by the host, which was a report with errors. Since this time the host generated reports every 30 minutes with no errors.

```
Loading production environment (Rails 4.1.5)

irb(main):001:0> Host.find_by_name('foo.com').get_status(HostStatus::ConfigurationStatus)

=> #<HostStatus::ConfigurationStatus id: 4, type: "HostStatus::ConfigurationStatus", status: 12288, host_id: 4, reported_at: "2016-03-02 11:34:16">

irb(main):002:0> Host.find_by_name('foo.com').get_status(HostStatus::ConfigurationStatus).relevant?

=> true

irb(main):003:0> Host.find_by_name('foo.com').refresh_statuses

=> 2

irb(main):004:0> Host.find_by_name('foo.com').get_status(HostStatus::ConfigurationStatus)

=> #<HostStatus::ConfigurationStatus id: 4, type: "HostStatus::ConfigurationStatus", status: 12288, host_id: 4, reported_at: "2016-03-02 11:34:16">

irb(main):005:0>
```

#8 - 03/03/2016 06:37 AM - Dominic Cleal

Could you please try running:

1. Host.find_by_name('foo.com').get_status(HostStatus::ConfigurationStatus).refresh
2. Host.find_by_name('foo.com').get_status(HostStatus::ConfigurationStatus)
3. HostStatus.status_registry

Thanks for the debug info, hopefully we'll be able to narrow it down and reproduce it soon.

#9 - 03/03/2016 09:00 AM - Marek Hulán

It seems to me that your reports are not associated or even created. Could you please try following?

```
Report.where(:host_id => 4).limit(2).order('created_at DESC')
```

Also I'd be interested in production.log, could you capture it while running a puppet run on some client (ideally on that host 'foo.com')? Also following might be interesting

```
Report.where(:host_id => 4).order('created_at DESC').first.status
Report.where(:host_id => 4).order('created_at DESC').first.logs
```

Note that these commands might contain some sensitive data, so clean the output first.

#10 - 03/03/2016 09:03 AM - Alexandre Barth

```
Loading production environment (Rails 4.1.5)
```

```
irb(main):001:0> Host.find_by_name('foo.com').get_status(HostStatus::ConfigurationStatus).refresh

=> 12288

irb(main):002:0> Host.find_by_name('foo.com').get_status(HostStatus::ConfigurationStatus)

=> #<HostStatus::ConfigurationStatus id: 4, type: "HostStatus::ConfigurationStatus", status: 12288, host_id: 4, reported_at: "2016-03-02 11:34:16">

irb(main):003:0> HostStatus.status_registry

=> #<Set: {HostStatus::ConfigurationStatus(id: integer, type: string, status: integer, host_id: integer, reported_at: datetime),
HostStatus::BuildStatus(id: integer, type: string, status: integer, host_id: integer, reported_at: datetime)}>

irb(main):004:0>
```

#11 - 03/03/2016 09:28 AM - Alexandre Barth

```
Loading production environment (Rails 4.1.5)

irb(main):001:0> Report.where(:host_id => 4).order('created_at DESC').first.status

=> {"applied"=>0, "restarted"=>0, "failed"=>0, "failed_restarts"=>0, "skipped"=>0, "pending"=>0}

irb(main):002:0> Report.where(:host_id => 4).order('created_at DESC').first.logs

=> #<ActiveRecord::Associations::CollectionProxy [#<Log id: 2550, source_id: 1, message_id: 3, report_id: 3607, level_id: 1, created_at: "2016-03-03
14:19:40", updated_at: "2016-03-03 14:19:40">, #<Log id: 2551, source_id: 1, message_id: 4, report_id: 3607, level_id: 1, created_at: "2016-03-03
14:19:40", updated_at: "2016-03-03 14:19:40">, #<Log id: 2552, source_id: 1, message_id: 212, report_id: 3607, level_id: 1, created_at: "2016-03-03
14:19:40", updated_at: "2016-03-03 14:19:40">, #<Log id: 2553, source_id: 1, message_id: 718, report_id: 3607, level_id: 1, created_at: "2016-03-03
14:19:40", updated_at: "2016-03-03 14:19:40">, #<Log id: 2554, source_id: 1, message_id: 719, report_id: 3607, level_id: 1, created_at: "2016-03-03
14:19:40", updated_at: "2016-03-03 14:19:40">]>

irb(main):003:0> Report.where(:host_id => 4).limit(2).order('created_at DESC')

=> #<ActiveRecord::Relation [#<ConfigReport id: 3607, host_id: 4, reported_at: "2016-03-03 14:19:26", created_at: "2016-03-03 14:19:40",
updated_at: "2016-03-03 14:19:40", status: 0, metrics: "--- !ruby/hash:ActionController::Parameters\nresour...", type: "ConfigReport">,
#<ConfigReport id: 3536, host_id: 4, reported_at: "2016-03-03 13:58:07", created_at: "2016-03-03 13:59:39", updated_at: "2016-03-03 13:59:39",
status: 0, metrics: "--- !ruby/hash:ActionController::Parameters\nresour...", type: "ConfigReport">]>

irb(main):004:0>
```

And this is production.log while running puppet run on foo.com host :

```
2016-03-03 15:19:27 [app] [I] Started GET "/node/foo.com?format=yml" for 20.69.2.95 at 2016-03-03 15:19:27 +0100

2016-03-03 15:19:27 [app] [I] Processing by HostsController#externalNodes as YML

2016-03-03 15:19:27 [app] [I]   Parameters: {"name"=>"foo.com"}

2016-03-03 15:19:27 [app] [I]   Rendered text template (0.0ms)

2016-03-03 15:19:27 [app] [I] Completed 200 OK in 99ms (Views: 0.8ms | ActiveRecord: 8.5ms)

2016-03-03 15:19:31 [app] [I] Started POST "/api/hosts/facts" for 20.69.2.95 at 2016-03-03 15:19:31 +0100

2016-03-03 15:19:31 [app] [I] Processing by Api::V2::HostsController#facts as JSON

2016-03-03 15:19:31 [app] [I]   Parameters: {"facts"=>"[FILTERED]", "name"=>"foo.com", "certname"=>"foo.com", "apiv"=>"v2",
:host=>{"name"=>"foo.com", "certname"=>"foo.com"}}

2016-03-03 15:19:31 [app] [I] Import facts for 'foo.com' completed. Added: 0, Updated: 8, Deleted 0 facts

2016-03-03 15:19:32 [app] [I] Completed 201 Created in 346ms (Views: 3.8ms | ActiveRecord: 34.7ms)

2016-03-03 15:19:32 [app] [I] Started GET "/node/foo.com?format=yml" for 20.69.2.95 at 2016-03-03 15:19:32 +0100

2016-03-03 15:19:32 [app] [I] Processing by HostsController#externalNodes as YML

2016-03-03 15:19:32 [app] [I]   Parameters: {"name"=>"foo.com"}

2016-03-03 15:19:32 [app] [I]   Rendered text template (0.0ms)

2016-03-03 15:19:32 [app] [I] Completed 200 OK in 101ms (Views: 0.8ms | ActiveRecord: 8.8ms)

2016-03-03 15:19:40 [app] [I] Started POST "/api/reports" for 20.69.2.95 at 2016-03-03 15:19:40 +0100
```

```
2016-03-03 15:19:40 [app] [I] Processing by Api::V2::ReportsController#create as JSON

2016-03-03 15:19:40 [app] [I]   Parameters: {"report"=>"[FILTERED]", "apiv"=>"v2"}

2016-03-03 15:19:40 [app] [I] processing report for foo.com

2016-03-03 15:19:40 [app] [I] Imported report for foo.com in 0.06 seconds

2016-03-03 15:19:40 [app] [I]   Rendered api/v2/reports/create.json.rabl (19.1ms)

2016-03-03 15:19:40 [app] [I] Completed 201 Created in 101ms (Views: 18.1ms | ActiveRecord: 23.7ms)
```

#12 - 03/03/2016 10:40 AM - Marek Hulán

I was trying to reproduce locally but I didn't succeed. Could you please enable debug log level and provide production.log during report import once again? Hopefully it would give us some clue.

#13 - 03/07/2016 04:00 AM - Marek Hulán

- Status changed from New to Ready For Testing

- Assignee set to Marek Hulán

It seems that order specified as a key on has_one association does not work, making it lamda helps. Apparently it was fixed for 1.12, I tested that it fixes the issue on 1.11. Dominic, would you mind cherry-picking this to 1.11-stable branch or should I create PR?

#14 - 03/07/2016 04:00 AM - Marek Hulán

- Related to Refactor #13883: Host::Managed.last_report_object association uses deprecated :order parameter added

#15 - 03/07/2016 04:02 AM - Dominic Cleal

- Status changed from Ready For Testing to Closed

Sure, I've moved the ticket and will cherry pick it for RC2. Thanks for finding that Marek, I'll close this ticket now - please reopen if you see it with the fix in RC2.