

Foreman - Bug #11771

Saving host without particular status fails

09/10/2015 10:57 AM - Stephen Benjamin

| | |
|---|---------------------------------|
| Status: Closed | |
| Priority: Normal | |
| Assignee: Dominic Cleal | |
| Category: Reporting | |
| Target version: 1.10.0 | |
| Difficulty: | Fixed in Releases: |
| Triaged: | Found in Releases: |
| Bugzilla link: | Red Hat JIRA: |
| Pull request: https://github.com/foreman/foreman/pull/2895 | |
| Description | |
| When I save a host that has never had a configuration report: | |
| <pre>PG::NotNullViolation: ERROR: null value in column "reported_at" violates not-null constraint DETAIL: Failing row contains (18, HostStatus::ConfigurationStatus, 0, 22, null). : INSERT INTO "host_status" ("host_id", "reported_at", "status", "type") VALUES (\$1, \$2, \$3, \$4) R ETURNING "id"</pre> | |
| I get a similar error for a host that was reported through configuration management, and never had a BuildStatus. | |
| Related issues: | |
| Related to Foreman - Feature #10782: Add global status for hosts | Closed 06/11/2015 |
| Related to Foreman - Bug #12954: Foreman puppet report status does not work w... | Closed 12/29/2015 |

Associated revisions

Revision 3fc47abd - 11/11/2015 12:05 PM - Dominic Cleal

fixes #11771 - ensure created HostStatuses are saved correctly

The HostStatus reported_at timestamp is now validated to match the DB schema and populated if it's missing. HostStatus subclasses should validate correctly without needing to call #refresh.

Since creation of a ConfigurationStatus etc. can be triggered by an API render (e.g. through foreman_hooks), the #relevant? method has been implemented to ensure it remains hidden when not actually in use.

Host::Managed#get_status will now only create a new status object if it doesn't already exist. It would only check those already in the DB for an existing host, so multiple #get_status calls would create a new status object each time, causing duplicates. Extra validation and an index have been added to ensure uniqueness.

Revision 6a979ca3 - 11/16/2015 09:30 AM - Dominic Cleal

fixes #11771 - ensure created HostStatuses are saved correctly

The HostStatus reported_at timestamp is now validated to match the DB schema and populated if it's missing. HostStatus subclasses should validate correctly without needing to call #refresh.

Since creation of a ConfigurationStatus etc. can be triggered by an API render (e.g. through foreman_hooks), the #relevant? method has been implemented to ensure it remains hidden when not actually in use.

Host::Managed#get_status will now only create a new status object if it doesn't already exist. It would only check those already in the DB for an existing host, so multiple #get_status calls would create a new status object each time, causing duplicates. Extra validation and an index have been added to ensure uniqueness.

History

#1 - 09/10/2015 11:00 AM - Dominic Cleal

- Related to Feature #10782: Add global status for hosts added

#2 - 09/10/2015 11:00 AM - Dominic Cleal

- translation missing: en.field_release set to 63

Do you have a stack trace or any more context to the error? I'm not sure I understand where it's originating from.

#3 - 09/10/2015 11:01 AM - Stephen Benjamin

It appears to be from this creating a status without a reported_at

<https://github.com/theforeman/foreman/blob/develop/app/models/host/managed.rb#L872>

#4 - 09/10/2015 11:05 AM - Stephen Benjamin

Stack trace wasn't really helpful, but:

```
app/models/concerns/foreman/sti.rb:29:in `save_with_type'  
app/models/concerns/foreman/sti.rb:29:in `save_with_type'  
app/controllers/hosts_controller.rb:103:in `block in update'  
app/models/taxonomy.rb:59:in `block in no_taxonomy_scope'  
app/models/taxonomy.rb:66:in `block (2 levels) in as_taxonomy'  
app/models/concerns/foreman/thread_session.rb:147:in `as_location'  
app/models/taxonomy.rb:65:in `block in as_taxonomy'  
app/models/concerns/foreman/thread_session.rb:112:in `as_org'  
app/models/taxonomy.rb:64:in `as_taxonomy'  
app/models/taxonomy.rb:58:in `no_taxonomy_scope'  
app/controllers/hosts_controller.rb:100:in `update'  
app/controllers/concerns/application_shared.rb:13:in `set_timezone'  
app/models/concerns/foreman/thread_session.rb:32:in `clear_thread'  
lib/middleware/catch_json_parse_errors.rb:9:in `call'
```

#5 - 09/10/2015 11:07 AM - Stephen Benjamin

Sorry, had wrong line selected, this: <https://github.com/theforeman/foreman/blob/develop/app/models/host/managed.rb#L882>

Edit (domcleal) with SHA:

<https://github.com/theforeman/foreman/blob/a265235eab649fa76e942205ef4964579532ce72/app/models/host/managed.rb#L882>

#6 - 09/17/2015 10:21 AM - Dominic Cleal

If you can provide any help in reproducing it, it'd be appreciated, I haven't been able to. I suspect there's a specific order in how the host is being created, whether it has or hasn't got reports, and how get_status is being called that triggers it.

My only guess is that this is happening on a report import (though you say no reports), as that calls refresh_statuses, which calls get_status. As far as I can tell, even if a ConfigurationStatus didn't exist, it'd be created and update_timestamp would set reported_at to Time.now. I don't see how it could remain nil.

#7 - 09/18/2015 07:59 AM - Dominic Cleal

- Status changed from New to Need more information

#8 - 10/13/2015 11:00 AM - Stephen Benjamin

I continue to see these errors constantly.

For example, provisioning a new machine and then clicking Cancel Build results in:

```
PG::NotNullViolation: ERROR: null value in column "reported_at" violates not-null constraint  
DETAIL: Failing row contains (42, HostStatus::ConfigurationStatus, 0, 30, null).  
: INSERT INTO "host_status" ("host_id", "reported_at", "status", "type") VALUES ($1, $2, $3, $4) RETURNING "id"  
"
```

#9 - 10/13/2015 11:02 AM - Stephen Benjamin

Do note I'm on postgresql not SQLite in my dev environment, if that makes a difference.

#10 - 10/13/2015 11:07 AM - Stephen Benjamin

What I don't understand is this is creating the missing host statuses:

<https://github.com/foreman/foreman/blob/a265235eab649fa76e942205ef4964579532ce72/app/models/host/managed.rb#L882>
but the reported_at is never set, so how does no one else see this?

The only thing that ever sets reported_at is when the status is refreshed, which doesn't happen for this code.

When I change it to:

```
new_status = host_statuses.new(:host => self, :type => type.to_s)
new_status.refresh
```

My problems seem to go away.

#11 - 10/13/2015 11:39 AM - Dominic Cleal

Are you able to trace how #get_status is called? Is it from #refresh_statuses?

If I call refresh_statuses manually on the console, I get a new ConfigurationStatus, and it gets refreshed because it's relevant? (= true).

#12 - 10/13/2015 05:58 PM - Stephen Benjamin

Yes:

```
/home/vagrant/foreman/app/models/host/managed.rb:903:in `get_status'
/home/vagrant/foreman/app/models/host/managed.rb:921:in `configuration_status'
/home/vagrant/.rvm/gems/ruby-2.1.5/gems/activerecord-3.2.21/lib/active_record/callbacks.rb:418:in `__run__1205136284132149582__validation__1215673669422551250__callbacks'
/home/vagrant/.rvm/gems/ruby-2.1.5/gems/activerecord-3.2.21/lib/active_record/callbacks.rb:405:in `__run_callback'
/home/vagrant/.rvm/gems/ruby-2.1.5/gems/activerecord-3.2.21/lib/active_record/callbacks.rb:385:in `__run_validation_callbacks'
/home/vagrant/.rvm/gems/ruby-2.1.5/gems/activerecord-3.2.21/lib/active_record/callbacks.rb:81:in `run_callbacks'
/home/vagrant/.rvm/gems/ruby-2.1.5/gems/activemodel-3.2.21/lib/active_model/validations/callbacks.rb:53:in `run_validations!'
/home/vagrant/.rvm/gems/ruby-2.1.5/gems/activemodel-3.2.21/lib/active_model/validations.rb:195:in `valid?'
/home/vagrant/.rvm/gems/ruby-2.1.5/gems/activerecord-3.2.21/lib/active_record/validations.rb:69:in `valid?'
/home/vagrant/foreman/app/models/concerns/orchestration.rb:75:in `valid?'
[..snip..]
/home/vagrant/foreman/app/models/host/managed.rb:273:in `built'
/home/vagrant/foreman/app/controllers/hosts_controller.rb:223:in `cancelBuild'
```

I created a new host without any configuration management (confirmed host.configuration? is false - so it shouldn't be relevant), then cancelled the build.

Does this work for you? Seems pretty reproducible for me.

#13 - 10/14/2015 03:14 AM - Dominic Cleal

I can't reproduce it at all unfortunately, I tried creating a new host with provisioning yesterday, then successfully cancelled the build before it built.

The stack trace is interesting, so it appears to be validating configuration_status. Do you have any plugins installed? Does removing them make any difference (if possible)?

#14 - 10/29/2015 11:38 AM - Alexandre Barth

I'm facing the same issue. Can't create any host with foreman 1.10 RC1

#15 - 11/05/2015 08:17 AM - Nat Backand

For me it occurs when the foreman_hooks plugin is used (more specifically: when a /host/managed/create hook is called). At that point I receive the exact same error with 1.10 RC1.

If I move the create hook out of the way adding a new host works fine. As soon as I re-add the create hook it bugs out again with the previously mentioned error.

#16 - 11/09/2015 03:28 AM - Alexandre Barth

I can confirm this, thank you Nat. When hooks are deactivated, it is working fine.

#17 - 11/09/2015 06:21 AM - Dominic Cleal

- Status changed from Need more information to Assigned

- Assignee set to Dominic Cleal

Nat Backand wrote:

For me it occurs when the foreman_hooks plugin is used (more specifically: when a /host/managed/create hook is called). At that point I receive the exact same error with 1.10 RC1.

Thanks, I can reproduce it from that. I suspect what's happening is that the API render used in foreman_hooks is calling one of the status methods. Stephen's issue might be similar if not using hooks, it could be a plugin checking the status?

#18 - 11/09/2015 08:15 AM - The Foreman Bot

- Status changed from Assigned to Ready For Testing
- Pull request <https://github.com/theforeman/foreman/pull/2895> added

#19 - 11/11/2015 01:01 PM - Dominic Cleal

- Status changed from Ready For Testing to Closed
- % Done changed from 0 to 100

Applied in changeset [3fc47abd9b52acb82189028540ec13e6289e8a03](#).

#20 - 01/04/2016 03:27 AM - Dominic Cleal

- Related to Bug #12954: Foreman puppet report status does not work without a puppet proxy added