Foreman - Bug #12013

Cannot provision host because MAC address is supposedly taken

09/30/2015 10:32 AM - Simon Wydooghe

Status:	Closed		
Priority:	Normal		
Assignee:	Marek Hulán		
Category:	Network		
Target version:	1.10.0		
Difficulty:		**Fixed in Releases:**	
Triaged:		**Found in Releases:**	1.9.2
Bugzilla link:		**Red Hat JIRA:**	
Pull request:	https://github.com/theforeman/foreman/pull/2774		

Description

I cannot provision a new virtual host because the MAC address is taken supposedly. I've provided two screenshots.
The first one show's the error. The second shows a search I did for that specific MAC address which turned out
empty. So I have no idea where Foreman is getting the information that this MAC exists somewhere.

The log file shows this:

```
2015-09-30T16:22:25 [app] [I] Started PUT "/discovered_hosts/mac525400e862b9" for 10.30.6.143 at 2
015-09-30 16:22:25 +0200
2015-09-30T16:22:25 [app] [I] Processing by DiscoveredHostsController#update as */*
2015-09-30T16:22:25 [app] [I]   Parameters: {"utf8"=>"✓", "authenticity_token"=>"L9Dovkmtyzt5qUEq0
H7WEw78UnIpmWUK6d33qQOzhqg=", "host"=>{"name"=>"vidyo1", "hostgroup_id"=>"10", "environment_id"=>"
1", "puppet_ca_proxy_id"=>"1", "puppet_proxy_id"=>"1", "puppetclass_ids"=>[""], "managed"=>"true",
 "progress_report_id"=>"[FILTERED]", "type"=>"Host::Managed", "interfaces_attributes"=>{"0"=>{"_de
stroy"=>"0", "mac"=>"52:54:00:e8:62:b9", "iden2015-09-30T16:22:25 [app] [I] Started PUT "/discover
ed_hosts/mac525400e862b9" for 10.30.6.143 at 2015-09-30 16:22:25 +0200
2015-09-30T16:22:25 [app] [I] Processing by DiscoveredHostsController#update as */*
2015-09-30T16:22:25 [app] [I]   Parameters: {"utf8"=>"✓", "authenticity_token"=>"L9Dovkmtyzt5qUEq0
H7WEw78UnIpmWUK6d33qQOzhqg=", "host"=>{"name"=>"vidyo1", "hostgroup_id"=>"10", "environment_id"=>"
1", "puppet_ca_proxy_id"=>"1", "puppet_proxy_id"=>"1", "puppetclass_ids"=>[""], "managed"=>"true",
 "progress_report_id"=>"[FILTERED]", "type"=>"Host::Managed", "interfaces_attributes"=>{"0"=>{"_de
stroy"=>"0", "mac"=>"52:54:00:e8:62:b9", "identifier"=>"eth0", "name"=>"vidyo1", "domain_id"=>"1",
 "subnet_id"=>"2", "ip"=>"10.30.202.49", "managed"=>"1", "primary"=>"1", "provision"=>"1", "tag"=>
"", "attached_to"=>"", "id"=>"88348"}}, "architecture_id"=>"1", "operatingsystem_id"=>"3", "build"
=>"1", "medium_id"=>"9", "ptable_id"=>"67", "disk"=>"", "root_pass"=>"[FILTERED]", "is_owned_by"=>
"3-Users", "enabled"=>"1", "model_id"=>"13252", "comment"=>"", "overwrite"=>"false"}, "id"=>"mac52
5400e862b9"}
2015-09-30T16:22:26 [app] [I] Failed to save: Mac has already been takentifier"=>"eth0", "name"=>"
vidyo1", "domain_id"=>"1", "subnet_id"=>"2", "ip"=>"10.30.202.49", "managed"=>"1", "primary"=>"1",
 "provision"=>"1", "tag"=>"", "attached_to"=>"", "id"=>"88348"}}, "architecture_id"=>"1", "operati
ngsystem_id"=>"3", "build"=>"1", "medium_id"=>"9", "ptable_id"=>"67", "disk"=>"", "root_pass"=>"[F
ILTERED]", "is_owned_by"=>"3-Users", "enabled"=>"1", "model_id"=>"13252", "comment"=>"", "overwrit
e"=>"false"}, "id"=>"mac525400e862b9"}
2015-09-30T16:22:26 [app] [I] Failed to save: Mac has already been taken
```

Anything I can do to help with this?

Related issues:

Related to Foreman - Tracker #2409: Networking		**New**	
Related to Foreman - Feature #11972: Implement regex for option ignore_puppet...		**Closed**	**09/26/2015**

Associated revisions

Revision 0085a137 - 10/02/2015 04:04 AM - Marek Hulán

Fixes #12013 - ignore all macvtap interfaces

History

#1 - 10/01/2015 03:46 AM - Marek Hulán

- Category set to Network

Try searching using "has_mac = ..." rather than mac that just searches for primary mac addresses. Any chance you're using organizations or locations? (implied by using katello e.g.)

#2 - 10/01/2015 05:35 AM - Simon Wydooghe

Aha, I found the machine in question... And I'm kicking myself in the head for not figuring this one out before. The guest connects to the local network with a macvtap interface on the host. The host (also registered in Foreman) reported a new macvtap interface with this MAC address to Foreman. When the guest wants to get autoprovisioned, Foreman says the MAC is already taken because the host reported this new interface. Is there any way around this? As it is now, this would mean macvtap adapters are a no go because both host and guest will always have some interface with identical MAC.

I deleted the interface on the host, and then I could provision the guest but on a next puppet run, the host will report the interface again.

#3 - 10/01/2015 05:36 AM - Simon Wydooghe

And no, no organizations or locations in use on my setup. Thank you for the "has_mac" tip by the way, it led me to find the culprit :)

#4 - 10/01/2015 06:25 AM - Marek Hulán

You're second user that reported this, is there some reason to use macvtap interfaces instead of other types? Unfortunately we don't have better solution right now than disabling importing interfaces based on puppet facts upload, to do it navigate to Administer -> Settings -> Provisioning and set "Ignore Puppet facts for provisioning" to true. This will disable any updates to network interfaces based on incoming data from puppet.

I think we could consider all macvtap interfaces as virtual (which do not enforce MAC address uniqueness), is it reasonable to rely on interface name being always in form of /macvtap.*/? On VM I suppose it's something like eth0 which would correctly be recognized as physical interface.

#5 - 10/01/2015 06:25 AM - Marek Hulán

- Related to Tracker #2409: Networking added

#6 - 10/01/2015 09:04 AM - Simon Wydooghe

The reason behind using direct interfaces is to not have another NAT and to be able to manage the network using the same network infrastructure used for physical machines. It also allows me to provision a VM from the Foreman managed TFTP server just like the physical machines.

This is the interface on the host:

```
4: macvtap0@eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UNKNOWN group default qlen
 500
    link/ether 52:54:00:e8:62:b9 brd ff:ff:ff:ff:ff:ff
    inet6 fe80::5054:ff:fee8:62b9/64 scope link
      valid_lft forever preferred_lft forever
```

This is the interface on the vm:

```
4: macvtap0@eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UNKNOWN group default qlen
 500
    link/ether 52:54:00:e8:62:b9 brd ff:ff:ff:ff:ff:ff
    inet6 fe80::5054:ff:fee8:62b9/64 scope link
      valid_lft forever preferred_lft forever
```

Good to know virtual interfaces do not require uniqueness. I will also try the "ignore puppet facts" setting.

#7 - 10/01/2015 09:13 AM - Simon Wydooghe

Sorry, I made a copy/paste error, this is the interface on the vm:

2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP group default qlen 1000
link/ether 52:54:00:e8:62:b9 brd ff:ff:ff:ff:ff:ff
inet 10.30.202.49/22 brd 10.30.203.255 scope global dynamic eth0
valid_lft 3506sec preferred_lft 3506sec
inet6 fe80::5054:ff:fee8:62b9/64 scope link
valid_lft forever preferred_lft forever

So a 'physical' interface just like you said.

#8 - 10/01/2015 10:54 AM - Marek Hulán

- Status changed from New to Assigned

- Assignee set to Marek Hulán

Ok, thanks for the explanation. Would you mind uploading facter output from both machines so I can easily reproduce locally and test the patch I'm working on? To gather the output, just run following on VM and hypervisor and put the files here please. You can clean any sensitive data from these files (e.g. IP can be replaced by some 129.168.0.1 stuff)

```
facter --yaml > vm.yaml
facter --yaml > hv.yaml
```

#9 - 10/01/2015 11:22 AM - Marek Hulán

Ah nevermind, it's not gonna work this way. Our virtual interfaces requires attached_to value, so the best we can do is to skip all interfaces with such identifier.

#10 - 10/01/2015 11:29 AM - The Foreman Bot

- Status changed from Assigned to Ready For Testing

- Pull request https://github.com/theforeman/foreman/pull/2774 added

- Pull request deleted ()

#11 - 10/01/2015 01:05 PM - Simon Wydooghe

Okay Marek, I have little knowledge of Foreman's internals so just let me know if you still want that facter output! Thanks for your help.

#12 - 10/02/2015 03:14 AM - Marek Hulán

Facter output is not needed, thanks. If you wish you can test the patch from https://github.com/theforeman/foreman/pull/2774 or wait until it gets merged and released, there's a chance it gets into upcoming 1.10.

#13 - 10/02/2015 05:01 AM - Marek Hulán

- Status changed from Ready For Testing to Closed

- % Done changed from 0 to 100

Applied in changeset 0085a13725e0b8a391110f2f023f15a32648e0df.

#14 - 10/02/2015 05:26 AM - Marek Hulán

- translation missing: en.field_release set to 63

#15 - 10/02/2015 07:04 AM - Lukas Zapletal

- Related to Feature #11972: Implement regex for option ignore_puppet_facts_for_provisioning added

Files

1443622817.png	73.4 KB	09/30/2015		Simon Wydooghe
1443622876.png	17.8 KB	09/30/2015		Simon Wydooghe