

## Email Updates - Defect #120

### Cron output: This system is receiving updates from Red Hat Subscription Management

10/17/2013 09:52 AM - Deoren Moor

<b>Status:</b> Closed	<b>Start date:</b> 10/17/2013
<b>Priority:</b> Normal	<b>Due date:</b>
<b>Assignee:</b> Deoren Moor	<b>% Done:</b> 0%
<b>Category:</b> RHEL - All	<b>Estimated time:</b> 0.00 hour
<b>Target version:</b> 0.2	<b>GitHub Issue:</b>
<b>Affected Version:</b> devel	
<b>Resolution:</b> Fixed	

**Description**

In [r79](#) the `sync_packages_list()` function tosses stdout for the yum check-update command, but not stderr. That makes sense as we would probably want to see real errors, but we really don't care to see that status message every time yum is called.

Reference: [source:/trunk/email\\_updates.sh@79#L367](source:/trunk/email_updates.sh@79#L367)

#### Associated revisions

##### Revision 80 - 10/17/2013 09:57 AM - Deoren Moor

[issues #120]

Tossing stdout from yum check-update like we have already been doing, but now grepping stderr to toss a verbose status message that we don't care to receive every time email\_updates.sh runs.

##### Revision 81 - 10/17/2013 10:01 AM - Deoren Moor

[issues #120]

Oops; I missed one.

##### Revision 82 - 10/17/2013 10:03 AM - Deoren Moor

[issues #120]

Fixed syntax error.

##### Revision 83 - 10/17/2013 10:06 AM - Deoren Moor

[issues #120]

sigh

#### History

##### #1 - 10/17/2013 10:09 AM - Deoren Moor

I found [a tip on StackOverflow](#) that showed how to toss stdout and grep stderr and have implemented that. Once I got past my introducing syntax errors, the changes appear to work great on a RHEL 5 box.

Index: email\_updates.sh

```
=====  
--- email_updates.sh      (revision 79)  
+++ email_updates.sh      (revision 83)  
@@ -364,7 +364,15 @@  
     yum )  
         # Skip upstream sync unless running in production mode  
         if [[ "${SKIP_UPSTREAM_SYNC}" -eq 0 ]]; then  
-         yum check-update > /dev/null  
+  
+         # Fixes #120  
+         #  
+         # Toss stdout, but only toss the one RHEL status message from
```

```

+         # stderr that just mentions the system is receiving updates
+         # from Red Hat Subscription Management
+         yum check-update 2> >(grep -v 'This system is receiving') \
+         > /dev/null
+
+         fi
+     ;;
+ esac
@@ -416,8 +424,12 @@
+     # Capturing output in array so we can more easily filter out what we're not
+     # interested in considering an "update". Don't toss lines without a number
+     # yet; sanitize_string() handles that. We need "Obsoleting Packages"
-     # in place as a cut-off marker
-     YUM_CHECKUPDATE_OUTPUT=$(yum check-update -C)
+     # in place as a cut-off marker. We're also tossing (see #120)
+     # the one RHEL status message from stderr that just mentions the system
+     # is receiving updates from Red Hat Subscription Management
+     YUM_CHECKUPDATE_OUTPUT=(
+         $(yum check-update 2> >(grep -v 'This system is receiving'))
+     )

+     for line in "${YUM_CHECKUPDATE_OUTPUT[@]}"
+     do

```

## #2 - 10/17/2013 10:12 AM - Deoren Moor

I'll run this latest revision for a while before I consider this fixed.

## #3 - 01/15/2014 12:33 PM - Deoren Moor

- Status changed from Assigned to Closed

- Resolution set to Fixed

No problems related to this have been encountered, so I'm considering this fixed.