



# Sending vC Ops logs to Log Insight

Prepared by

**Tomas Baublys**

[tbaublys@vmware.com](mailto:tbaublys@vmware.com)

Version 0.1

## Version History

Date	Author	Description	Reviewers
16. Juni, 2013	Tomas Baublys	First draft	

© 2013 VMware, Inc. All rights reserved. This product is protected by U.S. and international copyright and intellectual property laws. This product is covered by one or more patents listed at <http://www.vmware.com/download/patents.html>.

VMware is a registered trademark or trademark of VMware, Inc. in the United States and/or other jurisdictions. All other marks and names mentioned herein may be trademarks of their respective companies.

VMware, Inc.  
3401 Hillview Ave  
Palo Alto, CA 94304  
[www.vmware.com](http://www.vmware.com)

## Summary

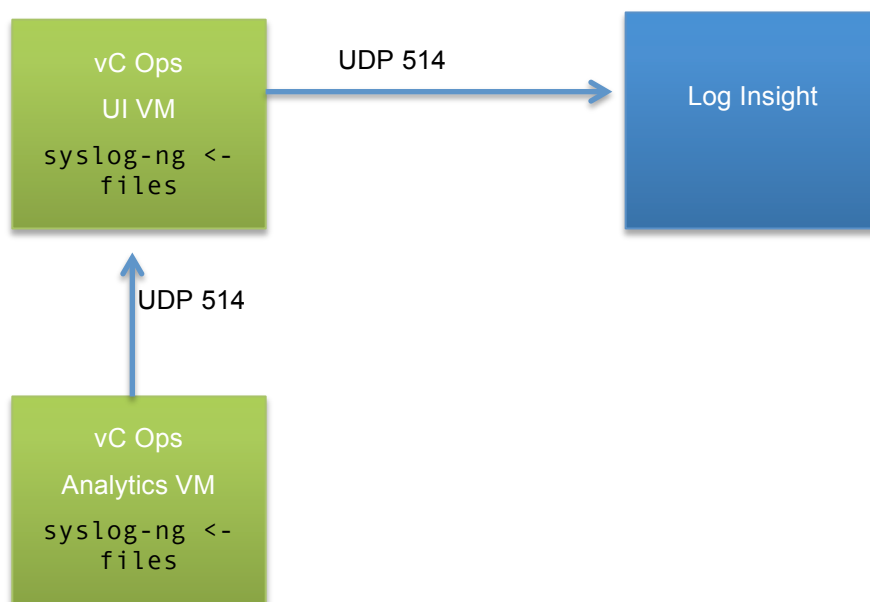
This document describes how to integrate relevant logs from vCenter Operations UI VM and Analytics VM to the Log Insight server for further analysis.

### 1.1 Prerequisites

1. SSH access to both vC Ops VMs, root credentials
2. No firewall or open connection from UI VM and Analytics VM to port 514 UDP on Log Insight server
3. Or at least a connection between both VMs (port 514) and an open connection from UI VM to the Log Insight server

### 1.2 Communication flow

There are at least two ways to get logs from both vC Ops VMs to Log Insight: either we prefer to send those directly or we consolidate it on one of VMs and use only one connection to Log Insight. If you prefer to send both directly the configuration steps will be identical on both VMs. We are going to use UI VM as a relay, so in addition it has to be configured to receive logs from the network (just one more line).



## 1.3 Configure sending logs to Log Insight (UI VM -> Log Insight)

1) On vC Ops UI VM edit `/etc/syslog-ng/syslog-ng.conf` (add your destination)

```
# Enable this and adopt IP to send log messages to a log server.
#
destination logserver { udp("loginsight01.pctp.local" port(514)); };
log { source(src); destination(logserver); };
```

2) Restart syslog

```
/etc/init.d/syslog restart
```

(if you don't like the error about missing news user, use `"useradd news"` command and restart again)

3) Test it with logger

```
# logger "Find a needle in a haystack"
```

4) Enjoy the message in the log insight UI

## 1.4 Configure UI VM to receive logs from Analytics VM

1) On vC Ops UI VM edit `/etc/syslog-ng/syslog-ng.conf` (allow receiving from network)

```
# uncomment to process log messages from network
udp (ip ("0.0.0.0") port(514));
```

2) On Analytics VM edit `/etc/syslog-ng/syslog-ng.conf` (configure forwarding to UI VM)

```
# Enable this and adopt IP to send log messages to a log server.
#
destination logserver { udp("172.29.20.1" port(514)); };
log { source(src); destination(logserver); };
```

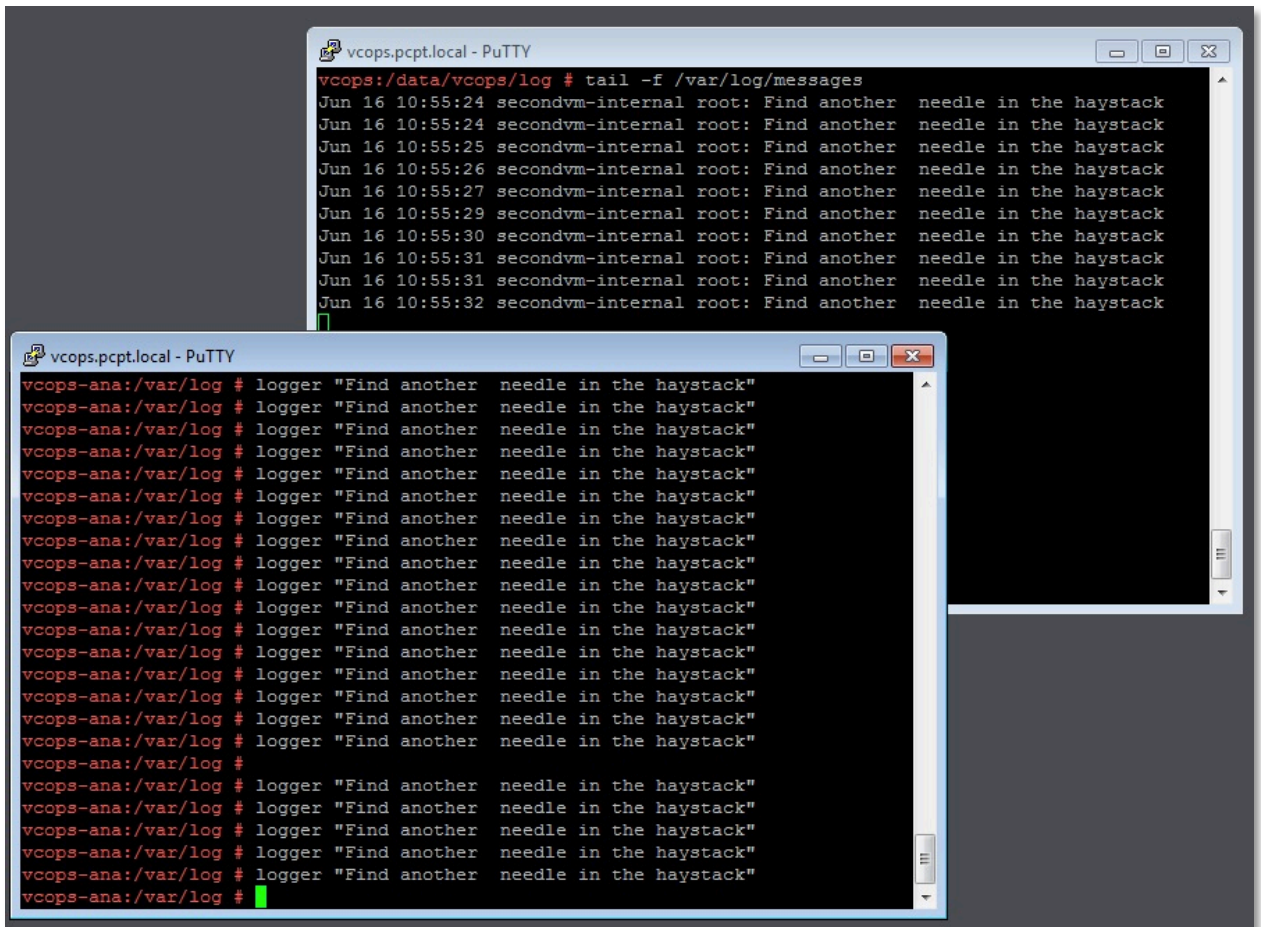
3) Test it with logger:

On Analytics VM:

```
# logger "Find another needle in a haystack"
```

On UI VM:

```
# tail -f /var/log/messages
```



**Figure 1: messages on UI VM showing message from Analytics VM**

As we already configured the UI VM to forward all messages to Log Insight, we should see the result in Log Insight UI:

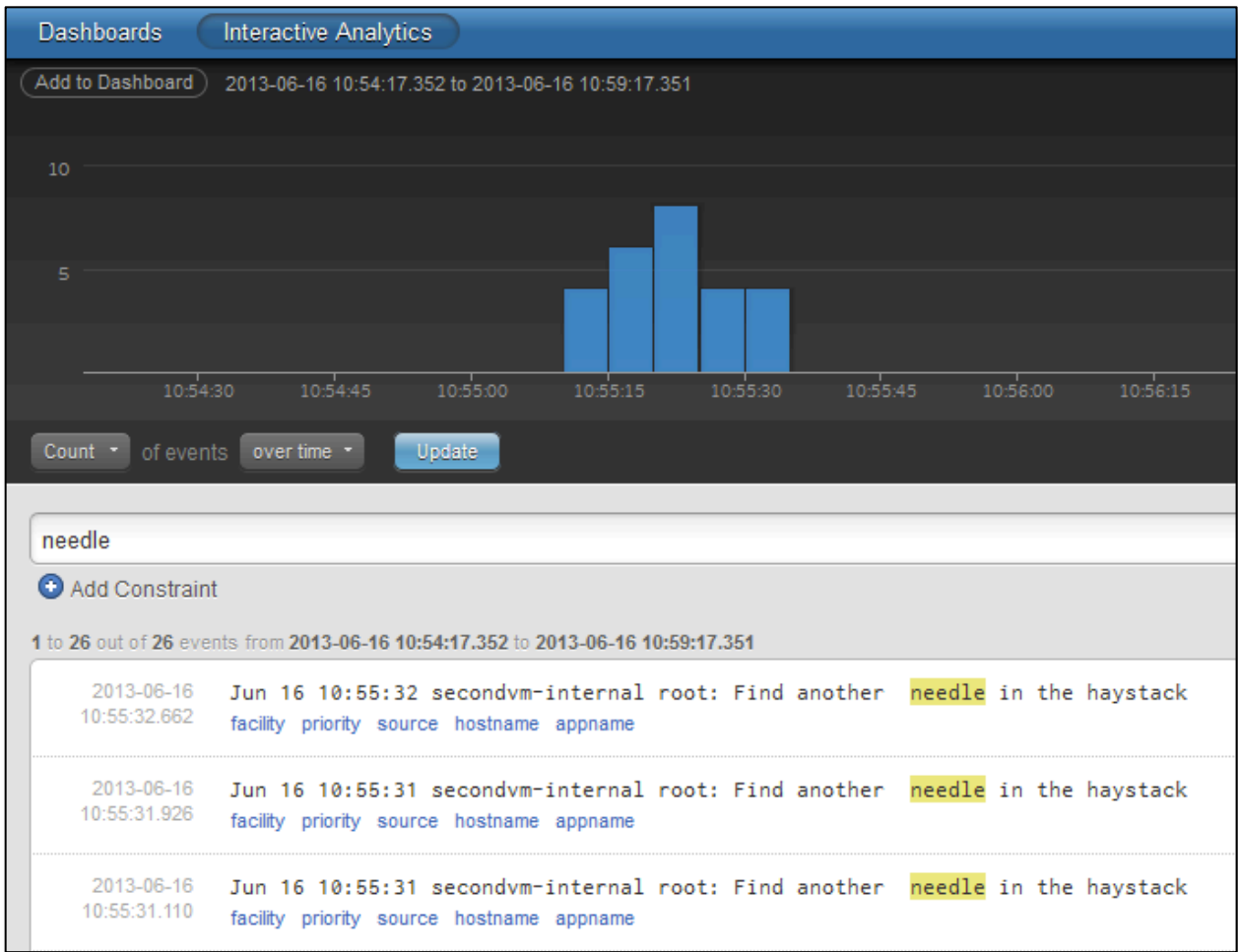


Figure 2: Same message from Analytics VM in the Log Insight

## 1.5 Configure additional logs to be included

Not all relevant vC Ops log messages are sent to the central logging facility (`/var/log/messages`) by default. Some of them are kept in different places to have them in specific application folders. So now we need to instruct the syslog-ng daemon to collect the info from those files. <sup>1</sup>

On UI VM we may add following entries below the line "`unix-dgram("/dev/log")`"

```
file("/var/log/apache2/access_log" follow_freq(10) flags(no-parse));
file("/var/log/apache2/error_log" follow_freq(10) flags(no-parse));
file("/var/log/vmware/admin.cmd.log" follow_freq(10) flags(no-parse));
file("/var/log/vmware/admin.log" follow_freq(10) flags(no-parse));
file("/var/log/vmware/ciq-firstboot.log" follow_freq(10) flags(no-parse));
file("/var/log/vmware/ciq.log" follow_freq(10) flags(no-parse));
file("/var/log/vmware/diskadd.log" follow_freq(10) flags(no-parse));
file("/var/log/vmware/lastupdate.log" follow_freq(10) flags(no-parse));
file("/var/log/vmware/mod_jk.log" follow_freq(10) flags(no-parse));
file("/var/log/vmware/vcops-admin.cmd.log" follow_freq(10) flags(no-parse));
file("/var/log/vmware/vcops-admin.log" follow_freq(10) flags(no-parse));
file("/var/log/vmware/vcops-firstboot.log" follow_freq(10) flags(no-parse));
file("/var/log/vmware/vcops-subsequentboot.log" follow_freq(10) flags(no-parse));
file("/var/log/vmware/vcops-watch.log" follow_freq(10) flags(no-parse));
file("/data/vcops/log/catalina-admin.log" follow_freq(10) flags(no-parse));
file("/data/vcops/log/catalina-web.log" follow_freq(10) flags(no-parse));
file("/data/vcops/log/catalina-web_std.log" follow_freq(10) flags(no-parse));
file("/data/vcops/log/host-manager-admin.log" follow_freq(10) flags(no-parse));
file("/data/vcops/log/host-manager-web.log" follow_freq(10) flags(no-parse));
file("/data/vcops/log/host-manager-web_std.log" follow_freq(10) flags(no-parse));
file("/data/vcops/log/HTTPPostAdapter.log" follow_freq(10) flags(no-parse));
file("/data/vcops/log/localhost-admin.log" follow_freq(10) flags(no-parse));
file("/data/vcops/log/localhost-web.log" follow_freq(10) flags(no-parse));
file("/data/vcops/log/localhost-web_std.log" follow_freq(10) flags(no-parse));
file("/data/vcops/log/analytics.log" follow_freq(10) flags(no-parse));
file("/data/vcops/log/collector.log" follow_freq(10) flags(no-parse));
```

On Analytics VM we should select different log files to be included:

```
file("/var/log/vmware/dbupgrade-alive.log" follow_freq(10) flags(no-parse));
```

---

<sup>1</sup> Excellent list of other logs we should monitor on different VMware products written by Michael White <https://blog.eng.vmware.com/logs/2013/06/what-other-log-files-should-we-monitor/>

```
file("/var/log/vmware/vcops-admin.log" follow_freq(10) flags(no-parse));
file("/var/log/vmware/vcops-watch.log" follow_freq(10) flags(no-parse));
file("/data/vcops/log/activemq.log" follow_freq(10) flags(no-parse));
file("/data/vcops/log/analytics.log" follow_freq(10) flags(no-parse));
file("/data/vcops/log/collector.log" follow_freq(10) flags(no-parse));
```

We can test the collection through adding a line to any of the logs in the list and checking to see it in the /var/log/messages:

```
# echo "This needle is hidden in the apache2 access log" >>
/var/log/apache2/access_log
```

and running grep to /var/log/messages (after 10 seconds)

```
# grep hidden /var/log/messages
```

If everything is working fine, we will also see the message in the log insight:

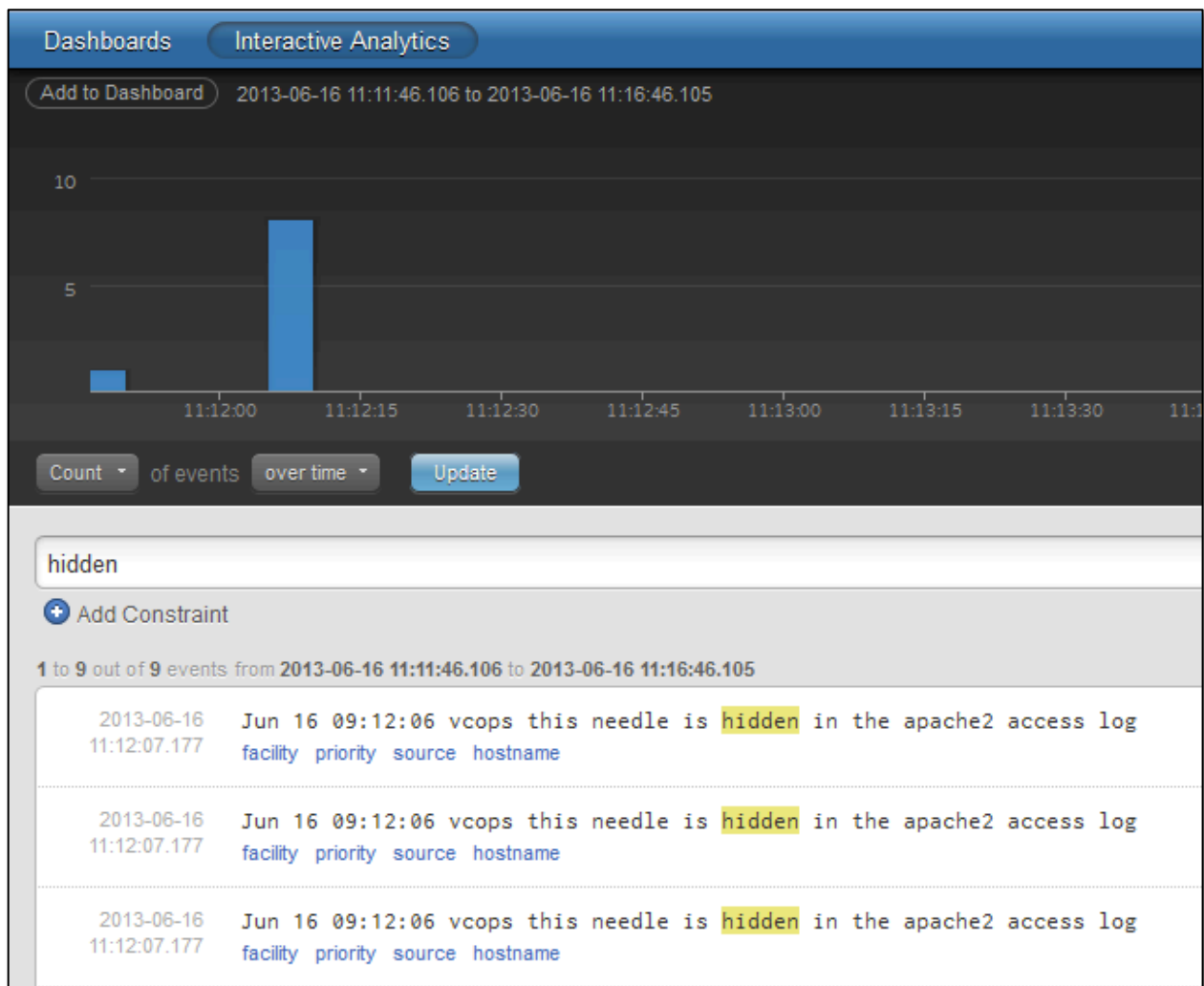


Figure 3: Log entry from apache2 collected by syslog and forwarded to Log Insight



Collecting all the application logs to /var/log/messages will make the central file a bit overcrowded. But this does not matter at all, as we are going to use Log Insight for analysis.