**DOCS-02-Configure Centralized AUDIT-Logging With Audispatch.Docx**

Written By: Warron French

Last Edited By: Warron S French

Table of Contents

[Preface 2](#_Toc450055734)

[Standards 2](#_Toc450055735)

[Configuration Procedure 3](#_Toc450055736)

[Configure clear-text log-data transport over a network 3](#_Toc450055737)

[On the client side 3](#_Toc450055738)

[On the server 3](#_Toc450055739)

[Configure TCP\_Wrappers support for receipt 3](#_Toc450055740)

[Configure iptables to support receipt of client audit-data 4](#_Toc450055741)

[Reboot all machines involved. 4](#_Toc450055742)

[Test Procedure 5](#_Toc450055743)

# Preface

The point of this document is to help a System Administrator figure out how to configure centralized audit logging on CentOS-6.x system so that the conventional and native commands – ***aureport*** and ***ausearch*** can function and parse logdata in a single place; while giving the ability to separate data on a per-host basis, or review all records within a dated-timeframe as it would pertain to all hosts.

## Standards

Information in a light blue background indicates the content of a file that has been opened for edits (perhaps using – vi).

#!/bin/bash –x

<some code snippet>

…

<some code snippet>

Other text with has something like ‘gdm@yourSys:/etc/gdm>’ in a grey background implies you should be the user=gdm, on the system=yourSys and in the directory=/etc/gdm – for example, and executing the command that prompt precedes; such as change directory to /etc/skel, for example

root@test-node1:~> cd /etc/skel

Note, the text will also be backgrounded in grey, so that points of major importance can be highlighted in yellow if necessary.

# Configuration Procedure

## Configure clear-text log-data transport over a network

This procedure begins with first configuring the client end of the connection. In the case of this sample the hostname will be referred to as **client1**, and the server will be referred to as **server1**; by which actual file manipulations will complete the configuration process. Then a restart of services will become necessary.

### On the client side

1. You will require the installation of the ***audispd-plugins*** package which should be reachable via the YUM repos, otherwise use the CentOS-6.7 Installation DvD1 and copy/rpm-install the package manually.
2. Configure the audit-remote-plugin on the client1 machine first, as the root user, and set the active attribute to the value of *yes*.

root@client1:~> vim /etc/audisp/plugins.d/au-remote.conf

active = yes

1. Next, as the root user, configure the “Audispatch Remote” module; and set all of the variables to the values associated below.

root@client1:~> vim /etc/audisp/audisp-remote.conf

remote\_server = server1

mode = forward

local\_port = 60

1. Finally, on the client1 machine tell the client to report its own name in the audit records that it submits to the primary loghost – server1 by altering the “Audispatch Daemon” configuration file; also be certain to use ALL CAPS for the term HOSTNAME and use that term specifically; this is not the hostname of the server, it is the word hostname.

root@client1:~> vim /etc/audisp/audispd.conf

name\_format = hostname

### On the server

1. Configure the server to receive audit log-data messages from the client (machines in your case); by starting off with an edit of the /etc/audit/auditd.conf file.

root@server1:~> vim /etc/audit/auditd.conf

set tcp\_listen\_port = 60

set tcp\_client\_ports = 60

set use\_libwrap = yes

**Note1**: Do not set use\_libwrap=yes unless you want to enforce the use of tcp\_wrappers; set it to no, if you don’t want to use tcp\_wrappers.

1. Configure the server’s auditd.conf file to set name\_format to hostname (explicitly).

root@server1:~> vim /etc/audit/auditd.conf

name\_format = hostname

#### Configure TCP\_Wrappers support for receipt

After telling auditd to listen on port 60, to receive client submission of audit-records, and also use TCP-Wrappers through the enabling of the *libwrap* library by setting the value to *yes* you will be required to enable the configuration of the traffic flow.

1. In /etc/hosts.allow enable the auditd process to listen to the network that the client1 machine is coming from – which for this working example is 192.168.22.0.

root@server1:~> vim /etc/hosts.allow

auditd:192.168.22.0

#### Configure iptables to support receipt of client audit-data

It will be necessary to configure iptables to allow a hole for port 60/tcp through the firewall.

1. Yank and copy the original line for ssh (port 22/tcp), then alter the numerical value from 22 to 60; to look like the following as a resulting line of iptables syntax:

ACCEPT tcp -- 0.0.0.0/0 0.0.0.0/0 state NEW tcp dpt:22

ACCEPT tcp -- 0.0.0.0/0 0.0.0.0/0 state NEW tcp dpt:60

### Reboot all machines involved.

This is the suggestion from Steve Grubb, because it will otherwise set the auid to the value of the admin’s AUID; and this is not desirable.

1. **Reboot** the whole server.

root@server1:~> reboot

1. **Reboot** the all of the clients; or in our case, **client1**.

root@client1:~> reboot

# Test Procedure

Leave no stone unturned, test everything so that you are confident that your audit trails really are getting logged into the central server, **server1** (in our case) from the **client1** machine (in our case), and can actually be audited by the audit team.

To check if working, perform the following on all machines within 10 minutes:

ausearch --start recent -m DAEMON\_ACCEPT -i

**Note2**: To get an encrypted transport, you need to use kerberos and that is beyond an email for setting it up.

**Note3**: You can now use the **ausearch** and **aureport** commands appropriately all on the single consolidated-audit-logging server and parse the data on a per client basis using the switch –node client1 (or whatever) to both commads.