

Wait, I don't want to be the Linux Administrator for SAS VA

Jonathan Boase; Zencos Consulting

ABSTRACT

Whether you are a new SAS administrator or switching to a Linux environment, you have a complex mission. This job becomes even more formidable when you are working with a system like SAS Visual Analytics that requires multiple users loading data daily. Eventually a user will have data issues or create a disruption that causes the system to malfunction. When that happens, what do you do next? In this paper, we will go through the basics of a SAS Visual Analytics Linux environment and how to troubleshoot the system when issues arise.

INTRODUCTION

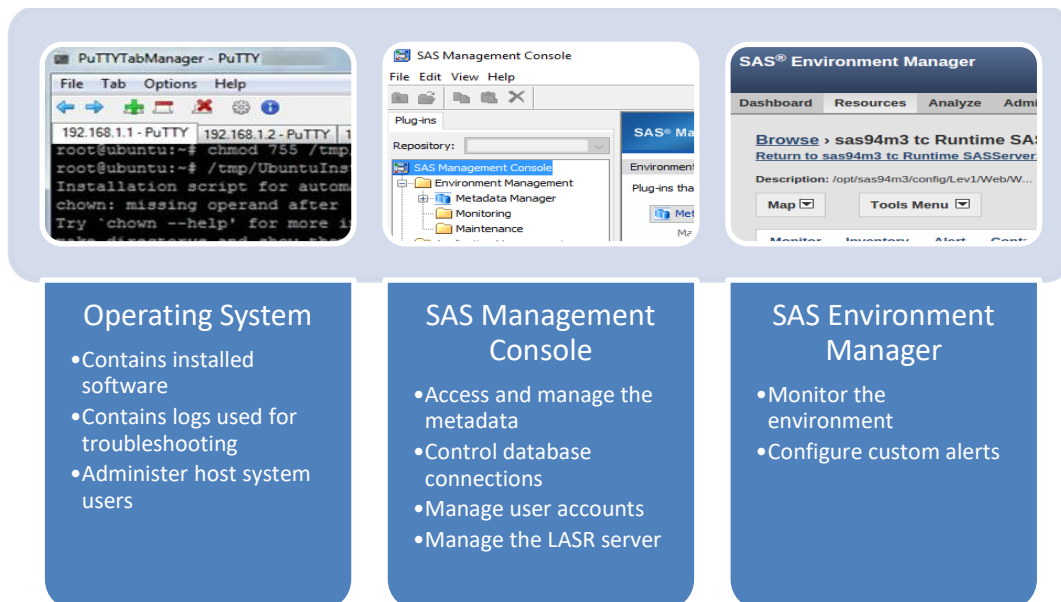
Many companies choose to implement SAS Visual Analytics in a Linux environment. With a distributed deployment, it's the only choice but many chose this operating system because it reduces operating costs. If you are the newly chosen SAS platform administrator, you might be more versed in a Windows environment and feel intimidated by Linux.

This paper introduces using basic Linux commands and methods for troubleshooting a SAS Visual Analytics environment. The paper assumes that SAS Visual Analytics is installed on a SAS 9.4 platform for Linux and that the reader has some familiarity with other operating systems, such as Windows.

PLATFORM ADMINISTRATION 101

SAS platform administrators work with three main product areas. Each area provides a different functionality based on the task the administrator needs to perform. The following figure defines each area and provides a general overview of its purpose.

Figure 1 Platform Administrator Tools



With any operating system, there is always a lot to learn. Just by knowing a few basic commands, you can navigate the system and troubleshoot issues. As your confidence grows, you can increase the skills and command knowledge.

ACCESSING THE ENVIRONMENT

The SAS software is installed on the Linux operating system. Most administrators access Linux remotely using an SSH client such as PuTTY or MobaXterm. You can download these applications to your Windows desktop. These applications allow you to securely access the Linux environment just as if you were on the Linux terminal.

Using an SSH client, you can run commands that allow you to interact with the operating system. To login, you will need an account on the host system, such as the SAS installer account (sas).

NAVIGATING LINUX

Once you have accessed the environment, you can start exploring the directory structure. There are two directories where the SAS software is installed: SASHOME and SASCONFIG. SASHOME contains supporting software like SAS Foundation and SAS Management Console. SASConfig contains the configuration and log files for SAS Visual Analytics. (SASHOME and SASCONFIG are also found in the Windows environment, but the paths are different.)

Exploring the Environment

Just like Windows, there are paths in Linux. A path is the folder or directory where files are located. A default installation path for SASCONFIG is **/opt/sas/config**. When logging into the environment, you start in the user's home directory. If the user is sas, then the current working directory is **/home/sas**. Typically, each user is assigned a sub directory in the Home directory.

The following commands are used to navigate the environment:

Command	Usage
pwd	Prints (or displays) the present working directory
cd <path>	Changes the directory to the specified path allowing you to navigate the file structure. Paths and filenames are case sensitive in Linux! Each of the following paths leads to a different folder because of the capitalization: cd /opt/SAS cd /opt/sas cd /opt/Sas
ls	Lists the directory contents.
ls -l	Add the -l option to see more file details.
ls -lart	Add -lart to see the detailed contents by date
ls *log	Use a wildcard (*) to see the listing of all files ending with log

In the following example, you can see how to use these commands. The sas user shows the working directory, navigates to the SASCONFIG Lev1 directory, and displays the directory contents.

Figure 2 Navigating the environment

```
[sas@server1 ~] pwd
/home/sas
[sas@server1 ~] cd /opt/sas/config/Lev1/
[sas@server1 Lev1] ls
AdvancedAnalyticsCommonDataServer  level_env_usermods.sh  sas.servers.pre
AppData                            Logs                   sasv9_meta.cfg
Applications                        manageservers.sh      schedulingServer
Backup                             metadataConfig_old1.xml Shared
ConfigData                         metadataConfig.xml    Utilities
ConnectSpawner                     ObjectSpawner         VASchedDist
DecisionManagerCommonDataSvr       perfStats.log         Web
DeploymentTesterServer             SASApp                WebInfrastructurePlatformDataServer
Documents                          SASMeta               WIPScheduledBatch
generate_boot_scripts.sh           sas.servers
level_env.sh                       sas.servers.mid
```

You can use absolute and relative paths with Linux. An absolute path means the location of a file or directory from the root directory (/). The **/opt/sas/config** path refers to the complete path from start to finish. Relative path is where you are related to the present working directory. If you are in the **/opt/sas**

directory and you want to go to the **/opt/sas/config** path, you can type **config** instead of the complete path. Notice that the forward slash was not used.

VIEWING FILES

Platform administrators should be familiar with the log locations and log content. Log files can help you understand system issues. SAS Visual Analytics is a collection of Web applications. When a user is having an issue with an individual application, such as the SAS Visual Analytics Designer or maybe logging into the Hub then you want to review its related log file. Use these commands to view files.

Command	Usage
cat <file>	Displays the entire file contents.
head <file>	Displays the first 10 lines of file.
tail <file>	Displays the last 10 lines of a file.

For this deployment, the SAS Visual Analytics applications are associated with SASServer12_1, so the associated logs are in that directory. In this example, we navigate to the SASServer12_1 directory, list the log files in the directory, and look at the end of the SAS LASR Authorization log file.

Figure 3 Viewing Files

```
[sas@server1 Lev1] cd Web/Logs/SASServer12_1/
[sas@server1 SASServer12_1] ls *log
SASLASRAuthorization7.3.log          SASVisualAnalyticsHub7.3.log
SASSearchService.log                SASVisualAnalyticsHyperlink7.3.log
SASVisualAnalyticsAdministrator7.3.log SASVisualAnalyticsTransport7.3.Journal.log
SASVisualAnalyticsDesigner7.3.log    SASVisualAnalyticsTransport7.3.log
SASVisualAnalyticsExplorer7.3.log    SASVisualAnalyticsViewer7.3.log
SASVisualAnalyticsGraphBuilder7.3.log SASVisualDataBuilder7.3.log
[sas@server1 SASServer12_1] tail SASLASRAuthorization7.3.log
 APP 2016-07-01 00:07:44,902 INFO [unknown unknown] com.sas.lasr.authorization.listener.ContextLi
eadedInit -
*
*
This application's 'ContextListener'.com.sas.lasr.authorization.listener.ContextListener - context
d.
context.getServerInfo(Pivotal tc Runtime 3.0.0.RELEASE/7.0.55.A.RELEASE)
context.getServletContextName(SAS LASR Authorization Service)
context.getInitParameterNames(
  application-name="LASR Authorization Service 7.3"
  log4j-config-name-prefix="SASLASRAuthorization")
```

Tip! Refer to [SAS Usage Note 55426](#) for a list of the logs in other middle tier directories.

SEARCHING FOR FILES OR FILE CONTENT

If you are having difficulty locating logs or other files, you can use the find command. If you need to search file content, you can use the grep command. The following table defines each command.

Command	Usage
find	Search directories for files. Syntax: find <location> <options> <search term>
grep	Searches files line-by-line for specific patterns. Syntax: grep <options> <search term> <location>

Searching for Files

You can search for specific logs or all logs in the environment. In the following example, the user searched for the SAS Visual Analytics Designer log. If you don't recall a full file name you can use wildcards (shown as *) to assist. Notice that the command returned other files with **design** in name. In the second example, the user was searching for the autoloading logs.

Figure 4 Searching for logs

```
[sas@server1 Lev1]$ find ./Web/Logs -name "*Desi*log"
./Web/Logs/SASServer12_1/SASVisualAnalyticsDesigner7.3.log
./Web/Logs/SASServer1_1/SASThemeDesignerForFlex4.7.log
[sas@server1 Lev1]$ find ./ -iname "*autoload*log"
./Applications/SASVisualAnalytics/VisualAnalyticsAdministrator/Logs/AutoLoad_2016.07.01_00.33.00.log
./Applications/SASVisualAnalytics/VisualAnalyticsAdministrator/Logs/AutoLoad_2016.07.01_00.50.01.log
./Applications/SASVisualAnalytics/VisualAnalyticsAdministrator/Logs/AutoLoad_2016.07.01_00.20.42.log
./Applications/SASVisualAnalytics/VisualAnalyticsAdministrator/Logs/AutoLoad_2016.07.01_00.52.00.log
./Applications/SASVisualAnalytics/VisualAnalyticsAdministrator/Logs/AutoLoad_2016.07.01_00.48.54.log
./Applications/SASVisualAnalytics/VisualAnalyticsAdministrator/Logs/AutoLoad_2016.07.01_00.45.03.log
./Applications/SASVisualAnalytics/VisualAnalyticsAdministrator/VALIBLA/Logs/AutoLoad_2016.09.08_11.16.02.log
./Applications/SASVisualAnalytics/VisualAnalyticsAdministrator/EVDMLA/Logs/AutoLoad_2016.07.01_00.02.03.log
./Applications/SASVisualAnalytics/VisualAnalyticsAdministrator/EVDMLA/Logs/AutoLoad_2016.07.01_00.06.46.log
./Applications/SASVisualAnalytics/VisualAnalyticsAdministrator/EVDMLA/Logs/AutoLoad_2016.07.01_00.14.00.log
./Applications/SASVisualAnalytics/VisualAnalyticsAdministrator/EVDMLA/Logs/AutoLoad_2016.07.01_00.09.58.log
[sas@server1 Lev1]$
```

Searching File Content

Some of the logs grow large and it is not easy to scroll through the content to find errors or warnings. In the following example, the user searches all the logs in current directory for an error using the grep command.

Figure 5 Searching file content

```
[sas@server1 SASServer11_1]$ pwd
/opt/sas/config/Lev1/Web/Logs/SASServer11_1
[sas@server1 SASServer11_1]$ grep "WARN" *.log
SASForecastServer14.1.log:WARN 2017-01-01 09:19:02,442 [localhost-startStop-2] - Forced destruction
context by the Session Service. Session Context=6alb040af300d3ec:-2ab4fc69:1584f2a97d3:-751c
SASTimeSeriesStudioMidTier14.1.log:WARN 2017-01-01 09:19:02,119 [localhost-startStop-2] - Forced d
e session context by the Session Service. Session Context=6alb040af300d3ec:-2ab4fc69:1584f2a97d3:-6
[sas@server1 SASServer11_1]$
```

It's smart to learn as many grep command tricks as possible. Perhaps you found a warning but don't recall the exact log. The command has multiple options, here's an example of searching through the subdirectories of Logs for warnings.

- **-r** Look through the subfolders starting from the `path` location
- **-i** ignores the case of the search pattern string
- **-l** (lower-case L) shows the matching file names

Figure 6 Advanced grep commands

```
[sas@server1 Web1]$ grep -rli './Logs' -e "warn"
./Logs/SASServer12_1/SASVisualAnalyticsTransport7.3.log
./Logs/SASServer12_1/SASSearchService.log
./Logs/SASServer12_1/SASVisualAnalyticsHyperlink7.3.log
./Logs/SASServer1_1/SASWIPClientAccess9.4.log
./Logs/SASServer1_1/SASIdentityServices9.4.log
./Logs/SASServer1_1/SASLogon9.4.log
./Logs/SASServer1_1/SASWIPServices9.4.log
./Logs/SASServer1_1/SASWorkflowServices9.4.log
./Logs/SASServer11_1/SASForecastServer14.1.log.2016-05-07
./Logs/SASServer11_1/SASForecastServer14.1.log.2016-07-07
./Logs/SASServer11_1/SASTimeSeriesStudioMidTier14.1.log.2016-05-04
```

TROUBLESHOOTING SKILLS

When you need to troubleshoot the system, it is useful to know some commands that help you determine the application and system status. This topic provides some helpful guidelines.

IS SAS VISUAL ANALYTICS RUNNING

There are several scripts in the Lev1 directory that are used to control SAS Visual Analytics and its supporting processes. The `sas.servers` script is the main one. It has four parameters that allow you to control what happens: *start*, *stop*, *restart*, and *status*. As you might guess, the start parameter begins all processes and likewise the stop parameter ends all processes.

When you run `sas.servers` with the status parameter it shows the server status. In the following example notice that the servers are UP, which means they are actively running. Note when running a script, you must append a “.” before the name to run it. This indicates the script is in the current working directory.

Figure 7 Example of `sas.servers` with status

```
[sas@server1 Lev1]$ ls
AdvancedAnalyticsCommonDataServer  Documents          ObjectSpawner      SchedulingServer
AppData                            generate_boot_scripts.sh  perfStats.log      Shared
Applications                        level_env.sh         SASApp             Utilities
Backup                             level_env_usermods.sh   SASMeta            VASchedDist
ConfigData                         Logs                sas.servers        Web
ConnectSpawner                    manageservers.sh       sas.servers.mid    WebInfrastructurePlatform
DecisionManagerCommonDataSvr       metadataConfig_old1.xml sas.servers.pre     WIPSSchedBatch
DeploymentTesterServer              metadataConfig.xml     sasv9_meta.cfg

[sas@server1 Lev1]$ ./sas.servers status
SAS servers status:
SAS Web Infrastructure Data Server is UP
SAS Advanced Analytics Common Data Server is UP
SAS Decision Manager Common Data Server is UP
SAS Metadata Server 1 is UP
SAS Object Spawner 1 is UP
SAS CONNECT Spawner 1 is UP
SAS DIP Job Runner 1 is UP
SAS Information Retrieval Studio Server is UP
SAS JMS Broker is UP
SAS Cache Locator Service ins_41415 is UP
SAS Web Server is UP
SAS Web App Server SASServer1_1 is UP
SAS Web App Server SASServer11_1 is UP
SAS Web App Server SASServer12_1 is UP
SAS Web App Server SASServer13_1 is UP
SAS Web App Server SASServer2_1 is UP
SAS Web App Server SASServer7_1 is UP
SAS Environment Manager is UP
SAS Environment Manager Agent is UP
[sas@server1 Lev1]$
```

Your output may look different based on what you have installed

The SAS Servers need to be started in order, which this script handles. Notice that the SAS Metadata Server is started prior to the Object Spawner and Web Server. The Metadata Server must be running so that permissions can be checked and other server connections can be made. SAS Visual Analytics is running when the SAS Web App Server `SASSERVERn` shows as “UP”. When we were looking at the logs, the logs were associated with `SASServer12_1`. Refer to [the system administration documentation](#) for a complete description of `sas.servers` script.

Note: If you are starting the `sas.server` processes you must be logged in as the SAS Installer (`sas`) account.

WHAT PROCESSES ARE RUNNING

Linux refers to running applications as processes or jobs. When a process is started, it is assigned a process ID, which is called a PID. You can see running processes and stop the process if necessary.

Command	Usage
<code>ps gf</code>	Shows information for the running processes. The <code>g</code> means for the group and <code>f</code> indicates you want a full listing. There are other options that might be useful.
<code>kill <pid></code>	Allows you to stop a running process. Use the <code>-9</code> option when you want the process to stop immediately. This is considered a nuclear option because the process does not end gracefully.

View Running Processes

In the following example, you can see the output of the `ps gf` command. The output has the following columns: PID, TTY, STAT, TIME, and COMMAND. With this command, you know everything running under your group and how long it has been running. Most of these applications should look familiar – for instance notice the Command column shows the Object Spawner and Metadata Server. This is the result of the `sas.servers` script.

Figure 8 Viewing running processes

```
[sas@server1 Lev1]$ ps gf
  PID TTY          STAT       TIME COMMAND
 7947 pts/2        Ss           0:00 -bash
16488 pts/2        T            0:00 \_ top
20936 pts/2      R+           0:00 \_ ps gf
 7772 pts/1      S+           0:00 bash
 5939 pts/0      Ss+          0:00 -bash
10649 pts/0      Sl           4:14 /opt/sas/sashome/SASPrivateJavaRuntimeEnvironment/9.4/jre/bin/java -Dnop
10233 pts/0      Sl           7:04 /opt/sas/sashome/SASPrivateJavaRuntimeEnvironment/9.4/jre/bin/java -Dnop
10014 pts/0      Sl           3:24 /opt/sas/sashome/SASPrivateJavaRuntimeEnvironment/9.4/jre/bin/java -Dnop
 9162 pts/0      Sl           40:10 /opt/sas/sashome/SASPrivateJavaRuntimeEnvironment/9.4/jre/bin/java -Dnop
 8774 pts/0      Sl           6:08 /opt/sas/sashome/SASPrivateJavaRuntimeEnvironment/9.4/jre/bin/java -Dnop
 7701 pts/0      Sl          124:40 /opt/sas/sashome/SASPrivateJavaRuntimeEnvironment/9.4/jre/bin/java -Dnop
 7368 pts/0      Sl           2:21 /opt/sas/sashome/SASPrivateJavaRuntimeEnvironment/9.4/jre/bin/java -Xms1
 7288 pts/0      Sl          19:57 /opt/sas/sashome/SASPrivateJavaRuntimeEnvironment/9.4/jre/bin/java -Xms1
 7259 pts/0      Sl           0:01 /usr/bin/python bin/information-retrieval-studio-server.py
 7342 pts/0      Z            0:00 \_ [pcrawler] <defunct>
 7203 pts/0      S            0:00 /bin/sh -p /opt/sas/config/Lev1/Web/Applications/SASWIPSSchedulingServices
 7208 pts/0      Sl           3:26 \_ /opt/sas/sashome/SASPrivateJavaRuntimeEnvironment/9.4/jre/bin/java -t
 7165 pts/0      S            0:00 /bin/sh -p /opt/sas/config/Lev1/ConnectSpawner/ConnectSpawner.sh start2
 7174 pts/0      Sl           0:00 \_ /opt/sas/sashome/SASFoundation/9.4/utilities/bin/cntspawn -sspi -xml
 7121 pts/0      S            0:00 /bin/sh -p /opt/sas/config/Lev1/ObjectSpawner/ObjectSpawner.sh start2 ta
 7130 pts/0      Sl           0:00 \_ /opt/sas/sashome/SASFoundation/9.4/utilities/bin/objspawn -dnsmatch
 7043 pts/0      S            0:00 /bin/sh -p /opt/sas/config/Lev1/SASMeta/MetadataServer/MetadataServer.sh
 7060 pts/0      Sl           3:09 \_ /opt/sas/sashome/SASFoundation/9.4/sasexe/sas
[sas@server1 Lev1]$
```

Stopping Long Running Scripts

If the output is long, you may want to narrow the list down further so you can see a specific process. This is another effective use of the `grep` command. For instance, suppose the Autoload script was having issues and needed to be stopped.

Let's use some of our commands to find the process and stop it. In the following example, the `ps` and `grep` commands are combined to search for the Autoload.sas program. Notice that when the results return, there are two PIDs. One for the autoload process and one for the `grep` process.

Figure 9 Killing a process

```
[sas@server1 ~]$ ps -ef | grep -i autoload
sas      530      1  0 00:51 pts/1    00:00:00 /opt/sas/sashome/SASFoundation/9.4/sasexe/sas
        -sysin /opt/sas/config/Lev1/Applications/SASVisualAnalytics/VisualAnalyticsAdministrator/AutoLoad
        d.sas -log /opt/sas/config/Lev1/Applications/SASVisualAnalytics/VisualAnalyticsAdministrator/Lo
        gs/AutoLoad_#Y.#m.#d_#H.#M.#S.log -print /opt/sas/config/Lev1/Applications/SASVisualAnalytics/V
        isualAnalyticsAdministrator/Logs/AutoLoad.lst -batch -noterminal -logparm rollover=session
sas      608 30255  0  0:52 pts/3    00:00:00 grep --color=auto -i autoload
[sas@server1 ~]$ kill -9 530
[sas@server1 ~]$ ps -ef | grep -i autoload
sas      658 30255  0  0:52 pts/3    00:00:00 grep --color=auto -i autoload
[sas@server1 ~]$
```

Once the PID is identified, the process is stopped. To confirm the process was stopped, repeat the initial command. Notice that the PID value changed for the `grep` command. This is a separate instance of the `grep` command so Linux assigns a new PID.

Caution!

- When using `kill`, **you must be accurate**. You can inadvertently kill a needed process by transposing a number. You might prefer the cut-and-paste method over your typing skills.
- Avoid using the `kill` command to stop the SAS services. Use the `sas.servers` script instead.

DETERMINE SYSTEM HEALTH

The following commands help you understand the status of the environment.

Command	Usage
df -h	Shows the available disk space by device name. Use the h option to make the output human readable or condense the data size.
top	Shows the currently running processes. Provides summary information about the CPU and memory usage. Tip! Some prefer htop over top. It provides a similar view but offers more functionality.

Review Disk Space

Use the df command to review the disk space. Sometimes errors occur because the server simply lacks room to process commands. In the following example, the df command shows the space on the devices. Watch the Use% column. If the main server was above 90% you might want to investigate the issue.

Figure 10 Review the disk space

```
[sas@server1 ~]$  
[sas@server1 ~]$ df -h  
Filesystem                Size  Used Avail Use% Mounted on  
/dev/mapper/vg_server1-lv_root 176G 141G 26G  85% /  
tmpfs                     14G   76K  14G   1% /dev/shm  
/dev/sda1                  477M   64M  388M  15% /boot  
/dev/mapper/vg_server1-lv_home 9.5G  190M  8.9G   3% /home  
192.168.1.159:/usbshare/_share 4.6T  487G  4.1T  11% /mnt/winshare/_share  
cm_processes               14G   2.5M   14G   1% /var/run/cloudera-scm-agent/process  
[sas@server1 ~]$
```

Find Process Hogs

When an application has an issue, it might consume all available CPU or memory. Use the top command to review the running processes along with other system statistics. Type q to return to the command line.

Figure 11 Finding memory hogs

```
top - 01:19:01 up 22:51, 1 user, load average: 0.02, 0.08, 0.14  
Tasks: 420 total, 1 running, 418 sleeping, 0 stopped, 1 zombie  
Cpu(s): 1.5%us, 1.2%sy, 0.2%ni, 97.1%id, 0.0%wa, 0.0%hi, 0.1%si, 0.0%st  
Mem: 28744856k total, 23484696k used, 5260160k free, 475672k buffers  
Swap: 12378108k total, 0k used, 12378108k free, 3365888k cached  


| PID  | USER     | PR | NI | VIRT  | RES  | SHR  | S | %CPU | %MEM | TIME+    | COMMAND  |
|------|----------|----|----|-------|------|------|---|------|------|----------|----------|
| 5781 | cloudera | 20 | 0  | 3408m | 705m | 18m  | S | 1.7  | 2.5  | 18:40.92 | java     |
| 7701 | sas      | 20 | 0  | 7695m | 2.9g | 30m  | S | 1.7  | 10.7 | 30:43.55 | java     |
| 2644 | yarn     | 20 | 0  | 1803m | 453m | 22m  | S | 1.3  | 1.6  | 13:29.80 | java     |
| 2813 | oozie    | 20 | 0  | 3408m | 530m | 22m  | S | 1.3  | 1.9  | 18:13.18 | java     |
| 2287 | cloudera | 20 | 0  | 4824m | 1.4g | 19m  | S | 1.0  | 5.0  | 16:07.07 | java     |
| 9162 | sas      | 20 | 0  | 5358m | 2.9g | 40m  | S | 0.7  | 10.6 | 43:03.48 | java     |
| 1950 | cloudera | 20 | 0  | 787m  | 5744 | 5000 | S | 0.3  | 0.0  | 0:08.67  | postgres |
| 2131 | hdfs     | 20 | 0  | 1606m | 218m | 21m  | S | 0.3  | 0.8  | 1:15.15  | jsvc     |
| 2442 | hdfs     | 20 | 0  | 1700m | 473m | 23m  | S | 0.3  | 1.7  | 5:39.64  | java     |


```

NEXT STEPS

If you are a new platform administrator, here are some things you should do:

- Read the SAS Visual Analytics administration manual so you are familiar with the tool and what customizations you can make.

- You learn some basics of the SAS Visual Analytics front-end environment. When a user contacts you with an issue, you should at least understand the differences among the web applications. Refer to [1] for a general overview of the application.
- Learn to use the SAS Management Console and SAS Environment Manager. You should also read the SAS documentation for these products.
- Seek classes or assistance from others as you learn. Many companies share tips on their blogs (Metacoda, Zencos, etc) that are useful to platform administrators.

CONCLUSION

If you are more familiar with a point-and-click environment, then transitioning to Linux will be a steeper learning curve. Tools like Mobaxterm that offer a point-and-click interface with the command line interface can ease your transition. It's smart to learn some of the shortcuts for the operating system like setting up alias commands and writing bash scripts.

REFERENCES

- [1] Aanderud, Kumpfmiller, Welke. 2016. "9 Frequently Asked Questions about Getting Started with SAS Visual Analytics" *Proceedings of the SAS Global 2016 Conference*, Las Vegas, NV: SAS Global Forum. Available at <http://support.sas.com/resources/papers/proceedings16/10360-2016.pdf>.
- [2] "Using the sas.servers Script on UNIX to Start or Stop All Servers", SAS(R) 9.4 Intelligence Platform: System Administration Guide, Fourth Edition.

ACKNOWLEDGMENTS

Thanks to David Barron, Andrew Peredery, and Nick Welke for reviewing and providing example ideas. Thanks to Tricia Aanderud and Maria Nicholson for assisting with writing and proofreading this paper. Thanks to Paul Homes for providing some artwork.

RECOMMENDED READING

- Paul Homes Platform Administrator Blog, <https://platformadmin.com/blogs/paul/>
- Zencos Blog, <http://www.zencos.com/blog>
- *Introduction to SAS Visual Analytics*, Aanderud, Collum, and Kumpfmiller, SAS Press

CONTACT INFORMATION

Your comments and questions are valued and encouraged. Contact the author at:

Jonathan Boase
SAS Platform Administration Consultant
Zencos Consulting [<http://www.zencos.com>]
jboase@zencos.com

All brand and product names are trademarks of their respective companies.