Linux Troubleshooting

RH342: Objective 1

Understand and employ general methods for troubleshooting

- Collect system information to aid in troubleshooting
- Consult documentation resources to aid in troubleshooting
- Monitor systems for vital characteristics
- Configure systems to send log messages to a centralized host

RH342 Troubleshooting Compendium

Objective 1 | Objective 2 | Objective 3 | Objective 4 | Objective 5 | Objective 6 | Objective 7 | Objective 8

Objective 1:

Understand and employ general methods for troubleshooting

- Collect system information to aid in troubleshooting
- Consult documentation resources to aid in troubleshooting
- Monitor systems for vital characteristics
- Configure systems to send log messages to a centralized host

Collect system information to aid in troubleshooting

Traditional | PCP

Traditional Methods

Basics

System load: `uptime`

Disk usage: `df -h`

Swap usage: `free -m`

Running processes: `top`

Listening processes (TCP): `ss -ltnp`

List processes: `ps aux`

Extras

Top 10 processes by CPU: `ps aux --sort -pcpu | head`

Top 10 processes by memory: `ps aux --sort -rss | head`

Open files: `lsof -u <USERNAME> | wc -l`

Open file limits: `su - <USERNAME> -c ulimit' -Hn'`

Open network connections: `lsof -i`

Note: key (yes, in capital, not small)

1. Press **M** - to sort the process list by **memory** usage.
2. Press **'P'** - to sort the process list by **cpu/processor** usage.
3. Press **'T'** - to sort by the running time.
4. Press **'R'** - to **Reverse** the sorting order for any(M,P,T)
5. **Press 'x' (lower case x) to** Highlight the sorted column with bold text
6. Press '**b**' - Highlight sorted column **background** color
7. **Press 'd' -** Change the update delay - 'd'

   < Change delay from 3.0 to > press number in seconds
8. Press 'o' or 'O' – Filter /Search processes

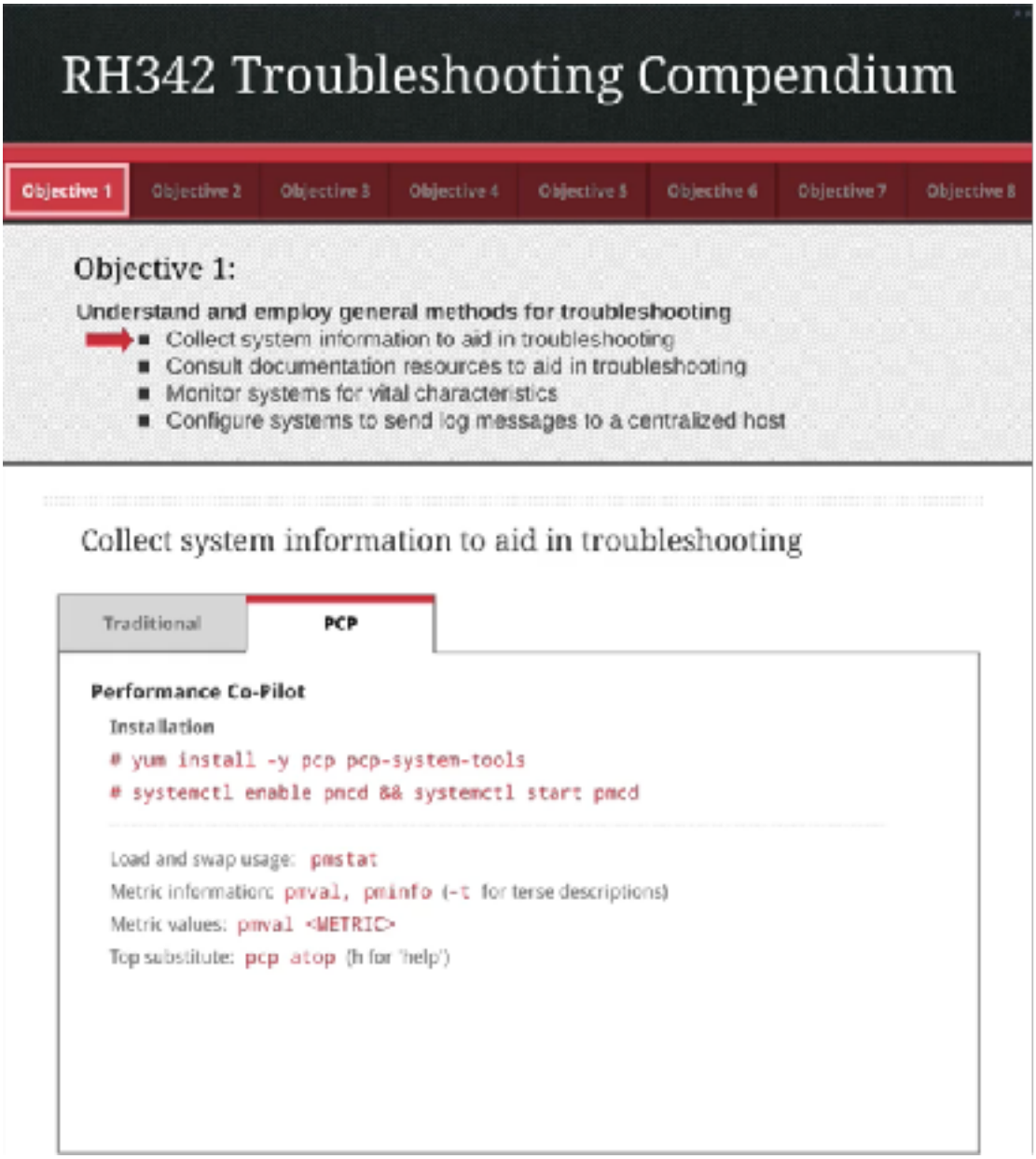
   Ex:

   add filter #3 (case sensitive) as: [!]FLD?VAL

   provide: %CPU>0.0

See active filters - Press Ctrl+o to see currently active filters
Clear filter - Press '=' key to clear any active filters

9. Press '1' - to display all CPU cores



```
[root@hdpnode14 ~]# pmstat
@ Wed Oct 16 03:38:43 2019
 loadavg                      memory      swap        io    system         cpu
   1 min   swpd   free   buff  cache   pi   po   bi   bo   in   cs  us  sy  id
    0.56  33784  4108m    884  6855m    0    0    0  244 1092 2056   2   1  97
    0.52  33784  4108m    884  6855m    0    0    0   10 1172 2152   2   1  97
    0.56  33784  4105m    884  6855m    0    0    0   24 2976 4236  30  11  58
```

RH342 Troubleshooting Compendium

Objective 1 | Objective 2 | Objective 3 | Objective 4 | **Objective 5** | Objective 6 | Objective 7 | Objective 8

Objective 5:

Troubleshoot and fix network connectivity issues

- Use standard tools to verify network connectivity
- Identify and fix network connectivity issues
- Inspect network traffic to aid troubleshooting

Use standard tools to verify network connectivity

Testing

Testing TCP connectivity with telnet

```
# telnet <HOST> <PORT>
```

Testing connectivity with nc

```
# nc -vz [add -u for UDP] <HOST> <PORT>
```

Listening

Using netcat to listen

```
# yum install nmap-ncat
# nc -l [add -u for UDP] <PORT>
```

RH342 Troubleshooting Compendium

Objective 5:

Troubleshoot and fix network connectivity issues

- Use standard tools to verify network connectivity
- Identify and fix network connectivity issues
- Inspect network traffic to aid troubleshooting

Identify and fix network connectivity issues

Network interface identification and configuration

Interface name

```
/etc/udev/rules.d/70-persistent-net.rules
```

Interface configuration

```
/etc/sysconfig/network-scripts/ifcfg-<NAME>
```

Interface information

```
# ip addr show dev <DEVICE>
# ip route
# nmcli d ( show for more detail)
# nmcli c ( show <CONNECTION> for more detail)
```

DNS configuration

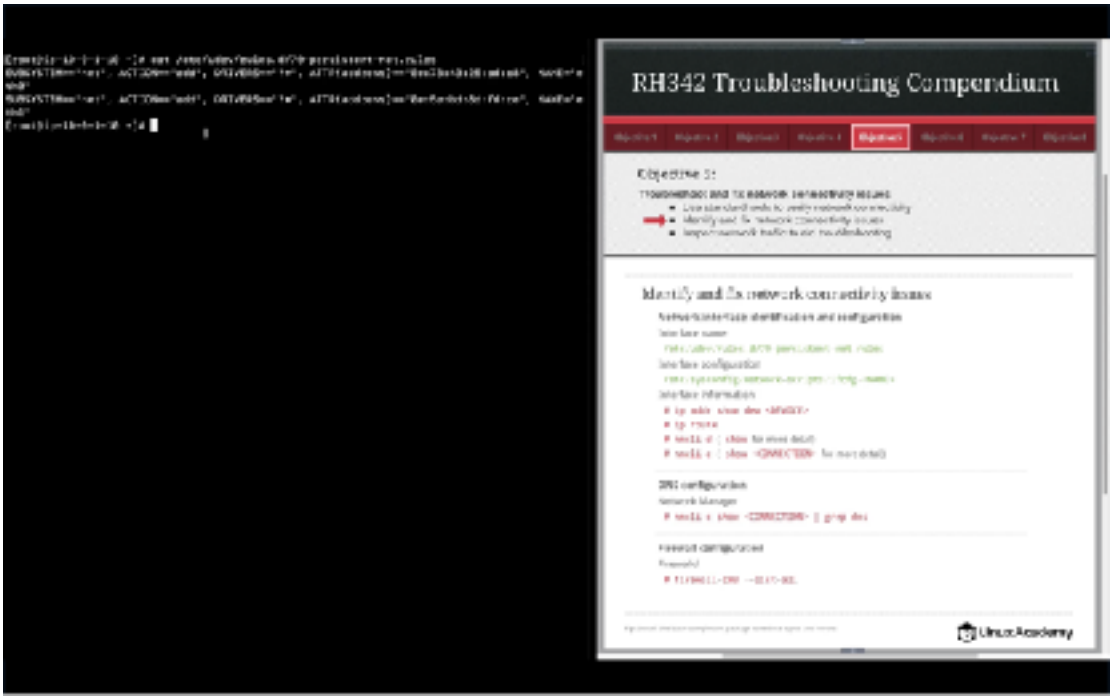
Network Manager

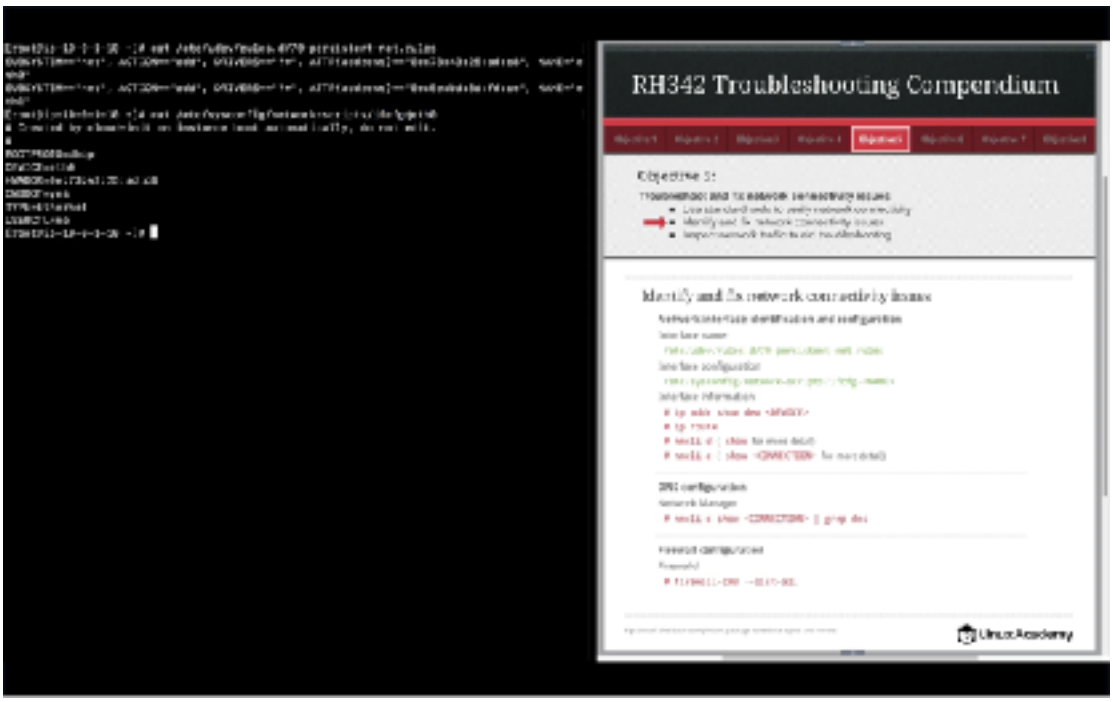
```
# nmcli c show <CONNECTION> | grep dns
```

Firewall configuration

Firewalld

```
# firewall-cmd --list-all
```


RH342 Troubleshooting Compendium
Objective 5:
Identify and fix network connectivity issues
Linux Academy


RH342 Troubleshooting Compendium
Objective 5:
Identify and fix network connectivity issues
Linux Academy

RH342 Troubleshooting Compendium

Objective 5:

Troubleshoot and fix network connectivity issues

- Use standard tools to verify network connectivity
- Identify and fix network connectivity issues
- Inspect network traffic to aid troubleshooting

Identify and fix network connectivity issues

Network interface identification and configuration

Interface name

```
/etc/udev/rules.d/70-persistent-net.rules
```

Interface configuration

```
/etc/sysconfig/network-scripts/ifcfg-<NAME>
```

Interface information

```
# ip addr show dev <DEVICE>
# ip route
# nmcli d ( show for more detail)
# nmcli c ( show <CONNECTION> for more detail)
```

DNS configuration

Network Manager

```
# nmcli c show <CONNECTION> | grep dns
```

Firewall configuration

Firewalld

```
# firewall-cmd --list-all
```

RH342 Troubleshooting Compendium

Objective 5:

Troubleshoot and fix network connectivity issues

- Use standard tools to verify network connectivity
- Identify and fix network connectivity issues
- Inspect network traffic to aid troubleshooting

Inspect network traffic to aid in troubleshooting

Using tcpdump

Create a packet capture

```
# yum install tcpdump
# tcpdump -i <INTERFACE> -w <OUTPUT FILE>
```

Read a packet capture

```
# tcpdump -r <FILE>
```

Using Wireshark

Create a packet capture

```
# yum install wireshark
# tshark -i <INTERFACE> -w <OUTPUT FILE>
```

Read a packet capture

```
# tshark -r <FILE>
```

Capture filters

Capture filter removing port 22 traffic

```
# [ tcpdump | tshark ] -i <INTERFACE> -w <OUTPUT FILE> 'port not 22'
```

RH342 Troubleshooting Compendium

Objective 1 | Objective 2 | Objective 3 | Objective 4 | Objective 5 | **Objective 6** | Objective 7 | Objective 8

Objective 6:

Diagnose application issues

- Identify library dependencies for third-party software
- Identify if an application suffers from memory leaks
- Use standard tools to debug an application
- Identify and fix issues related to SELinux

Identify library dependencies for third-party software

Resolving missing libraries

List libraries needed by an application

```
# ldd <PATH/APPLICATION>
```

Identify libraries

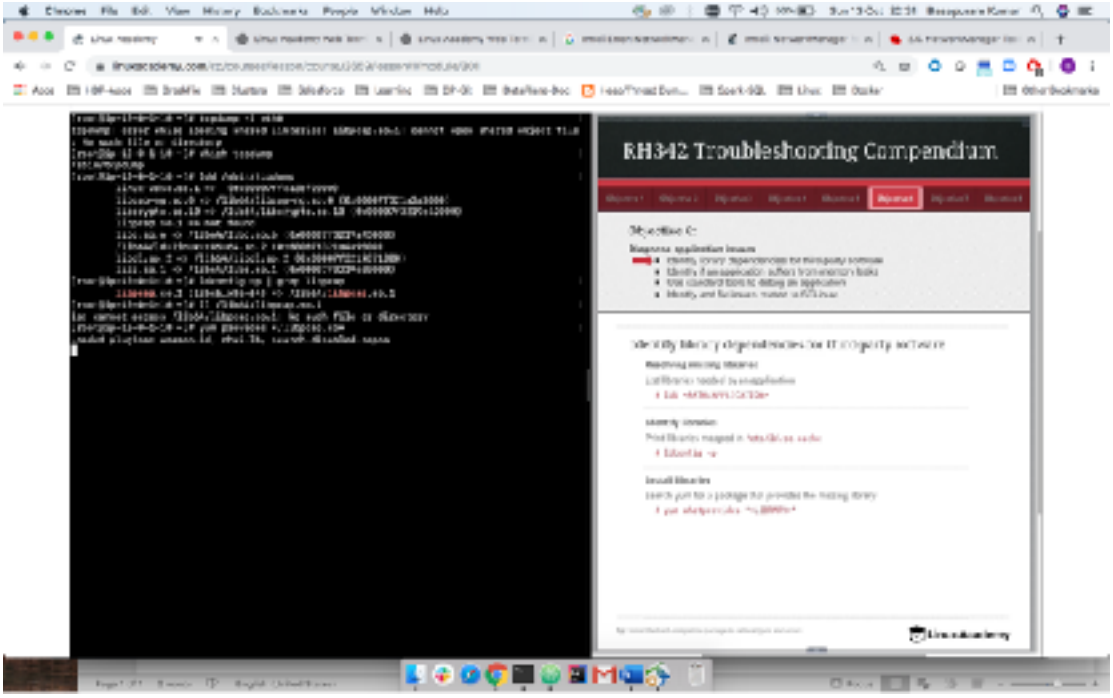
Print libraries mapped in `/etc/ld.so.cache`

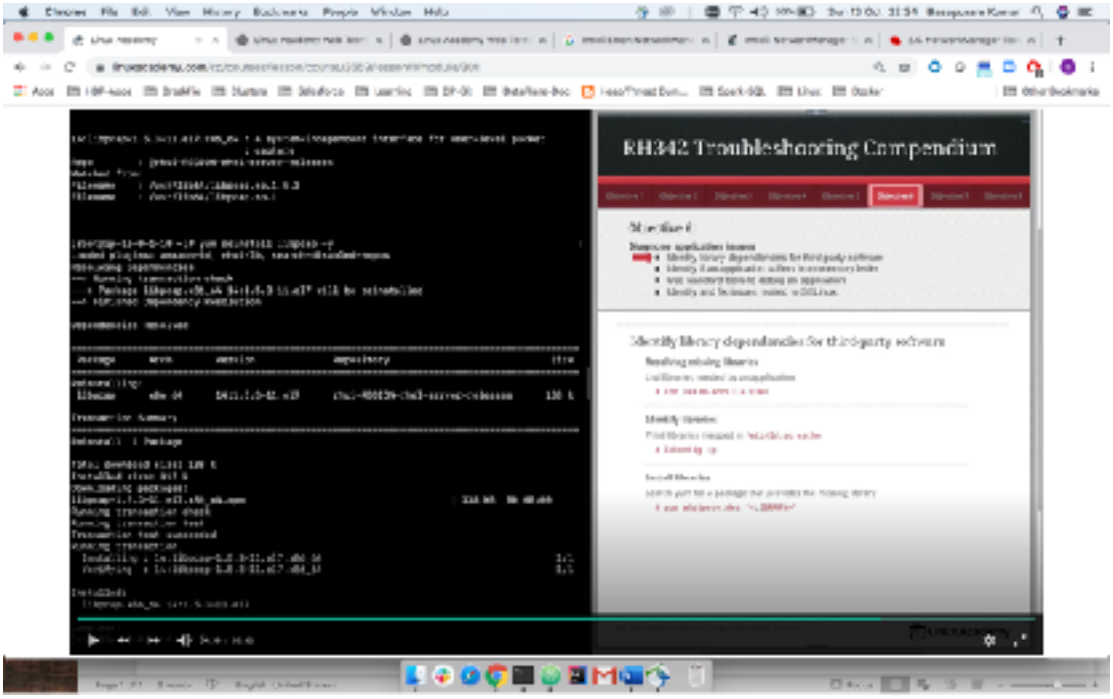
```
# ldconfig -p
```

Install libraries

Search yum for a package that provides the missing library

```
# yum whatprovides *<LIBRARY>*
```





Application Memory leak issue and detecting

Objective 6:

Diagnose application issues

- Identify library dependencies for third-party software
- Identify if an application suffers from memory leaks
- Use standard tools to debug an application
- Identify and fix issues related to SELinux

Identify if an application suffers from memory leaks

Memory consumption

Show swap usage

```
# free -m
```

List the top 10 processes by memory consumption

```
# ps aux --sort -rss | head
```

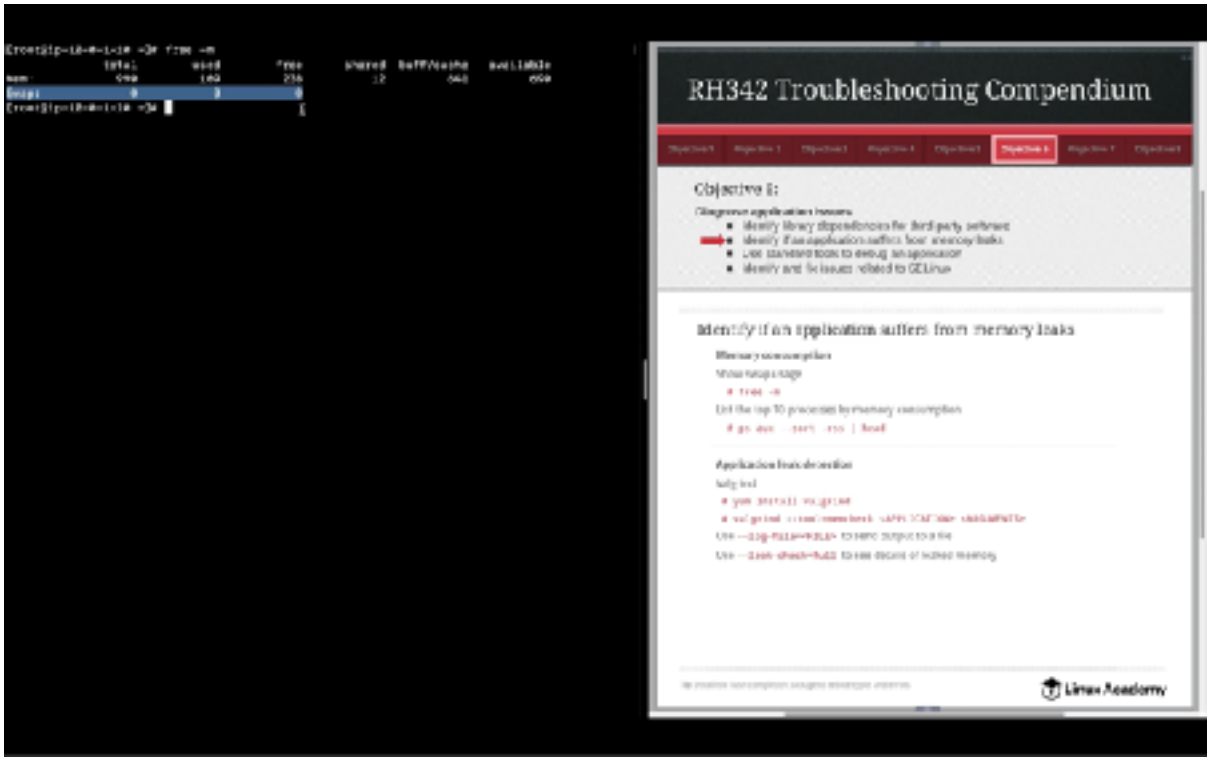
Application leak detection

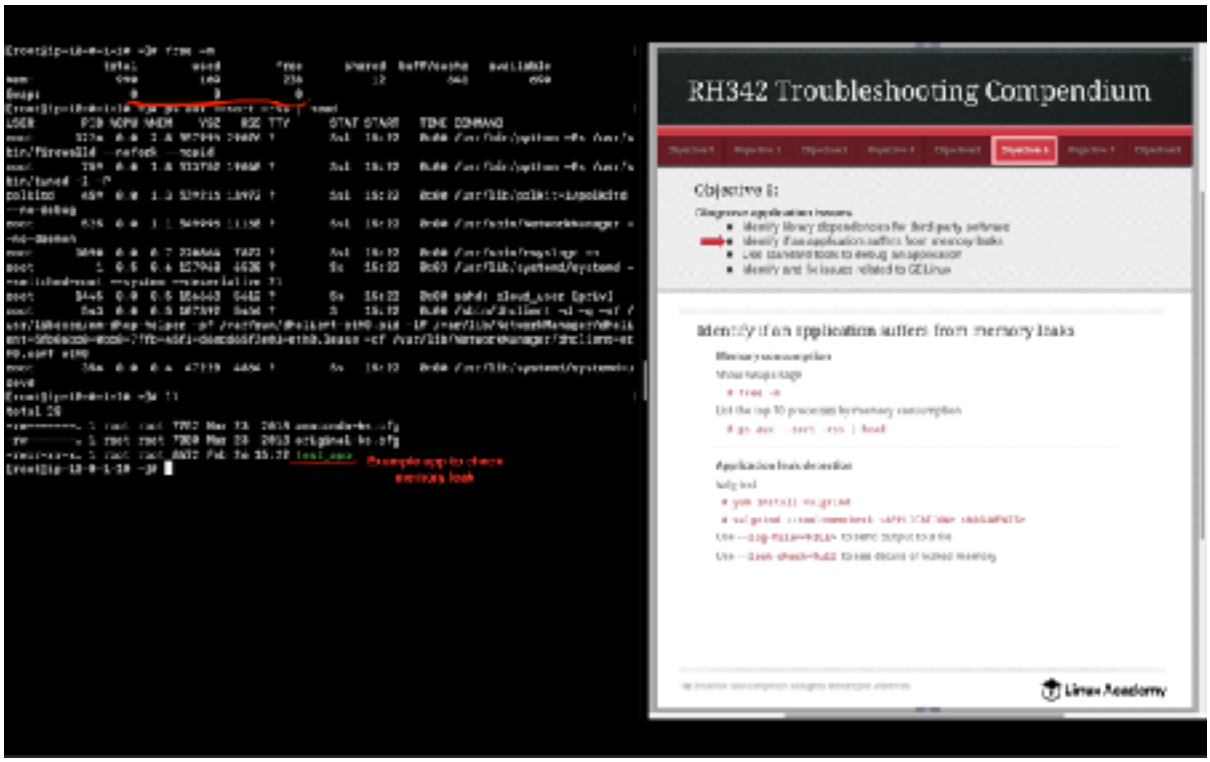
Valgrind

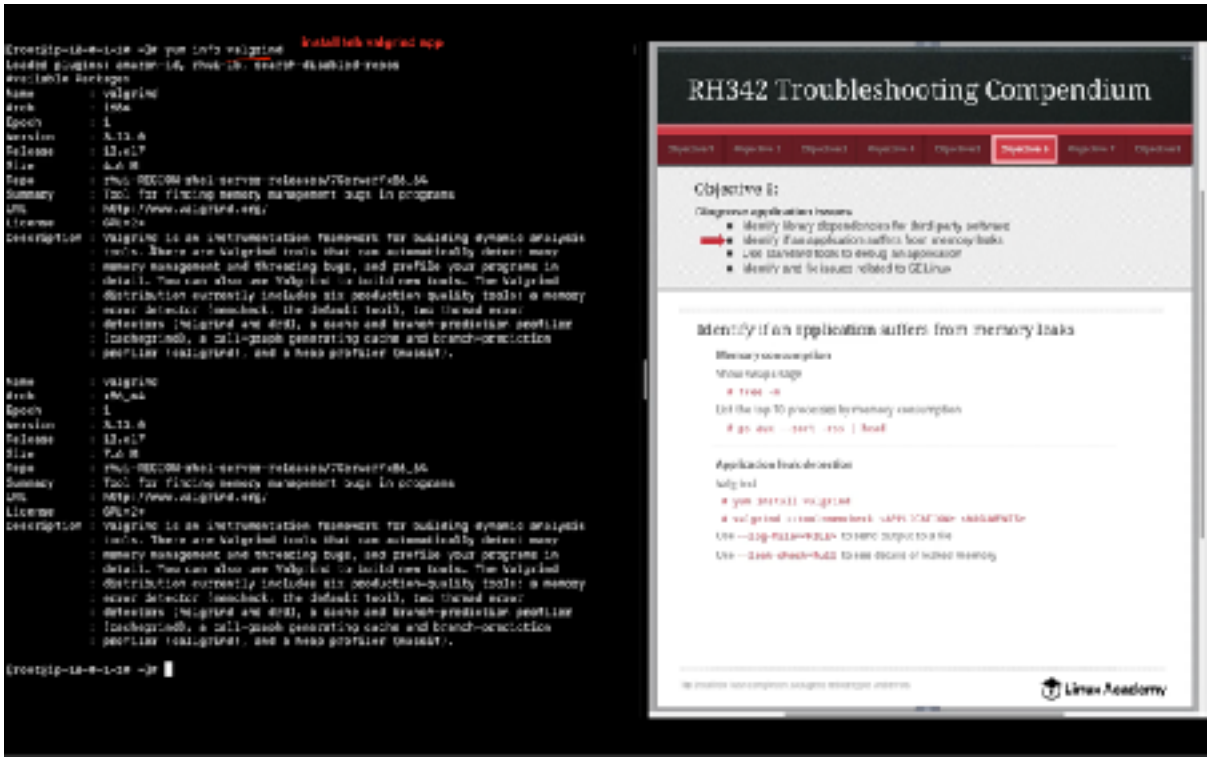
```
# yum install valgrind
# valgrind --tool=memcheck <APPLICATION> <ARGUMENTS>
```

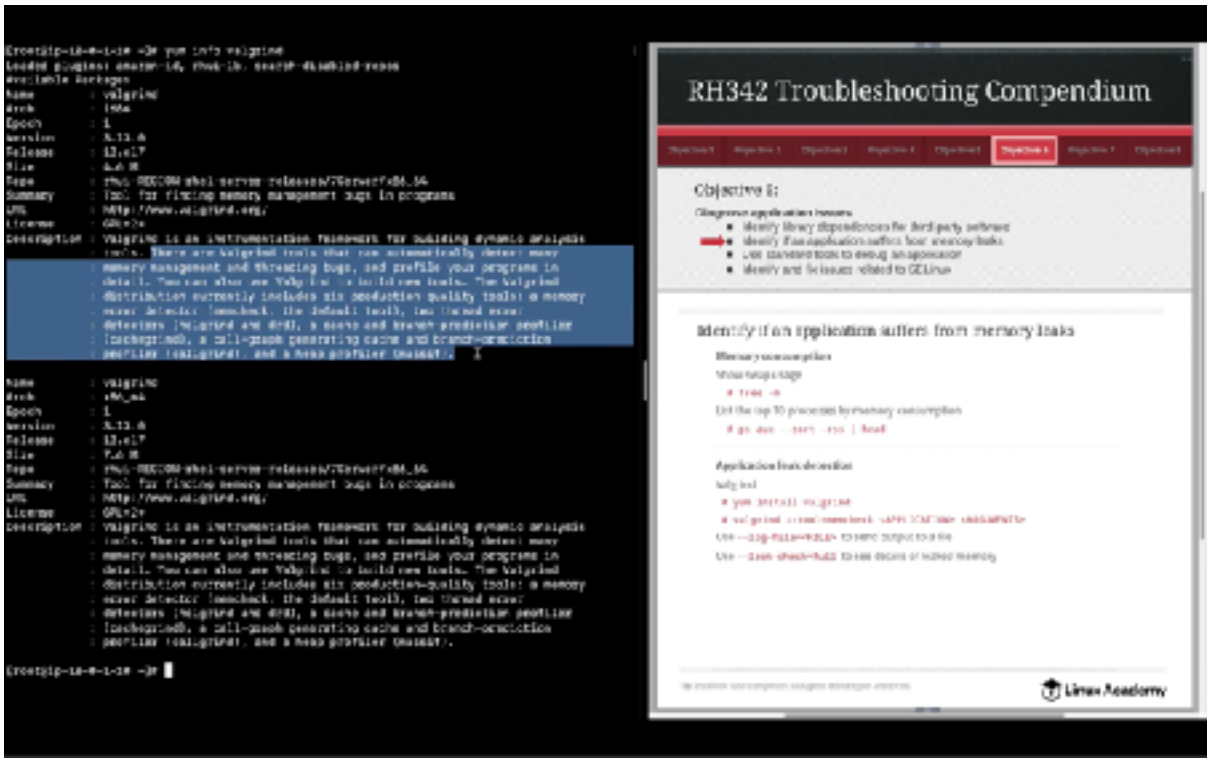
Use `--log-file=<FILE>` to send output to a file

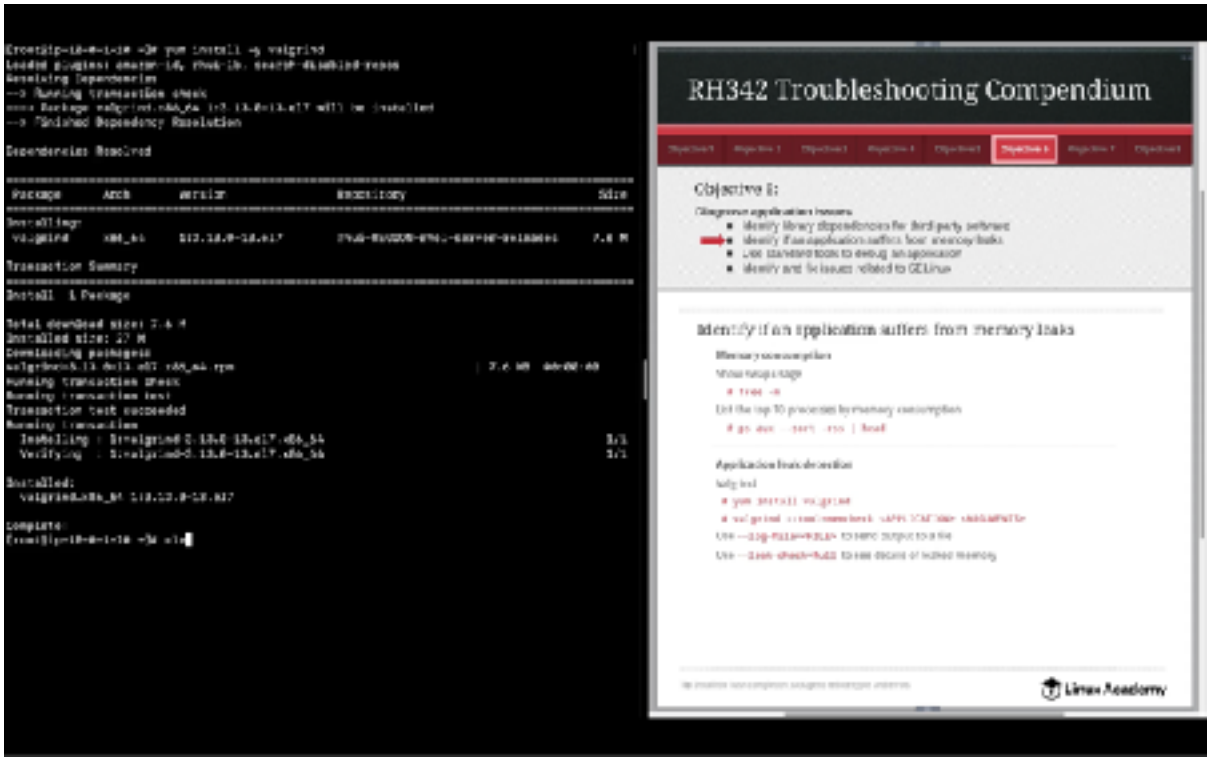
Use `--leak-check=full` to see details of leaked memory

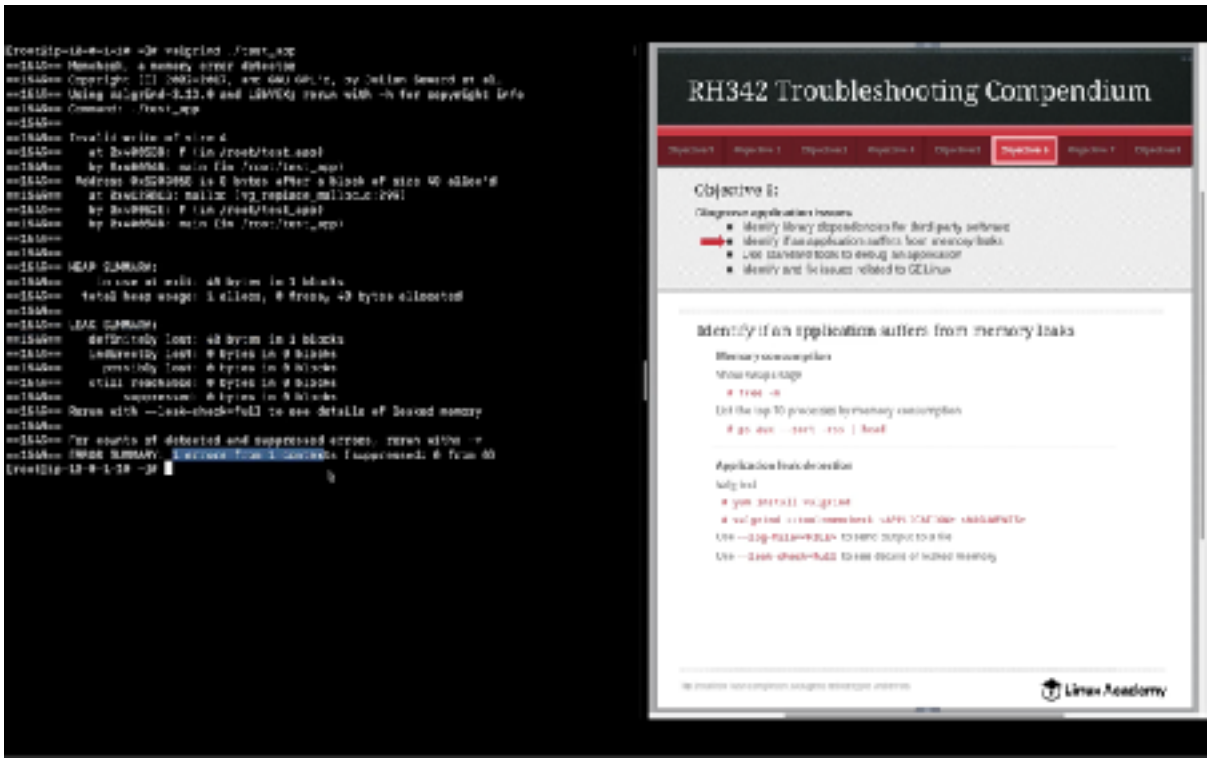

RH342 Troubleshooting Compendium
Objective I:
Identify if an application suffers from memory leaks
Linux Academy

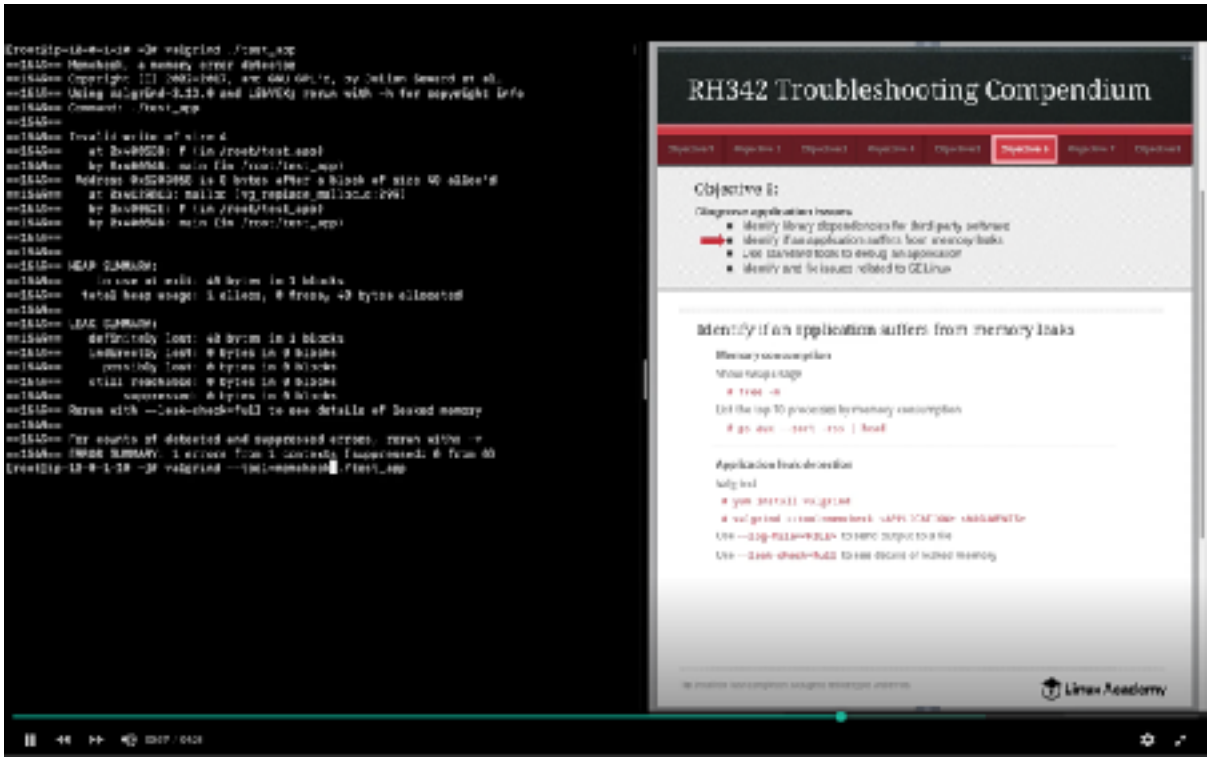

RH342 Troubleshooting Compendium
Objective I:
Identify if an application suffers from memory leaks
Example app to check memory leak
Linux Academy

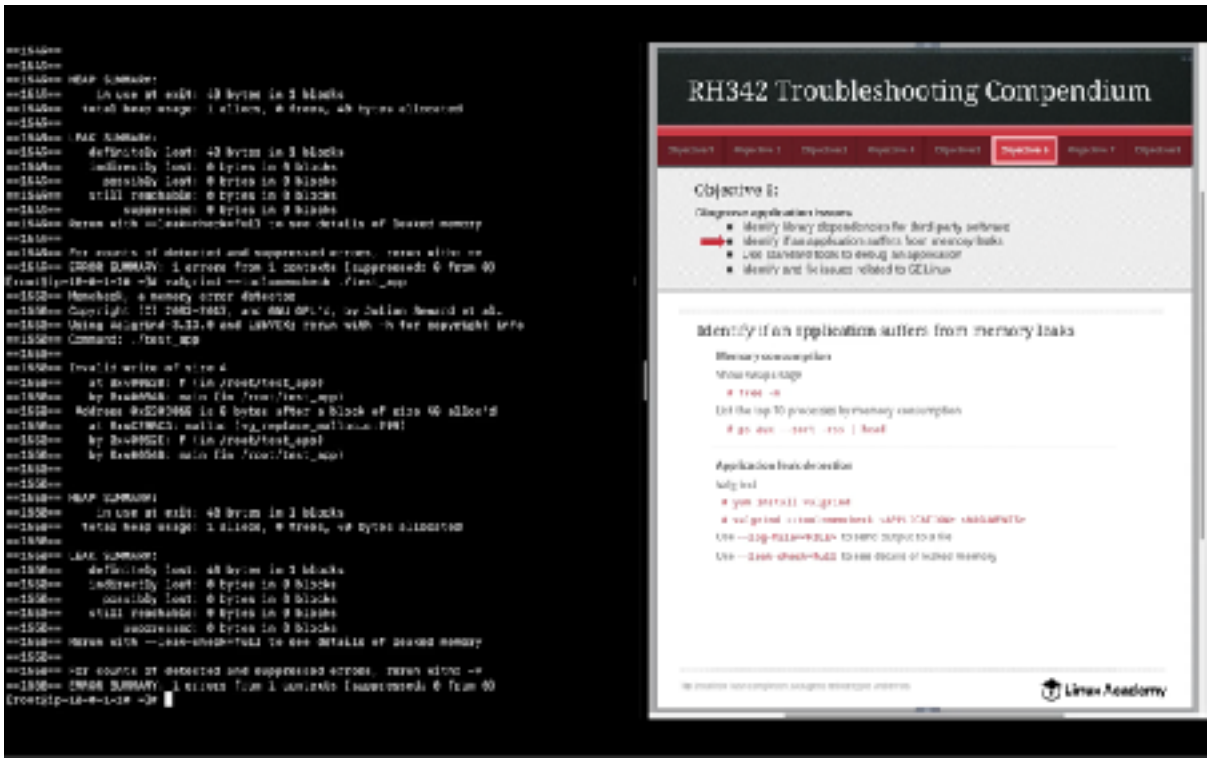

RH342 Troubleshooting Compendium
Objective I:
Identify if an application suffers from memory leaks
Linux Academy

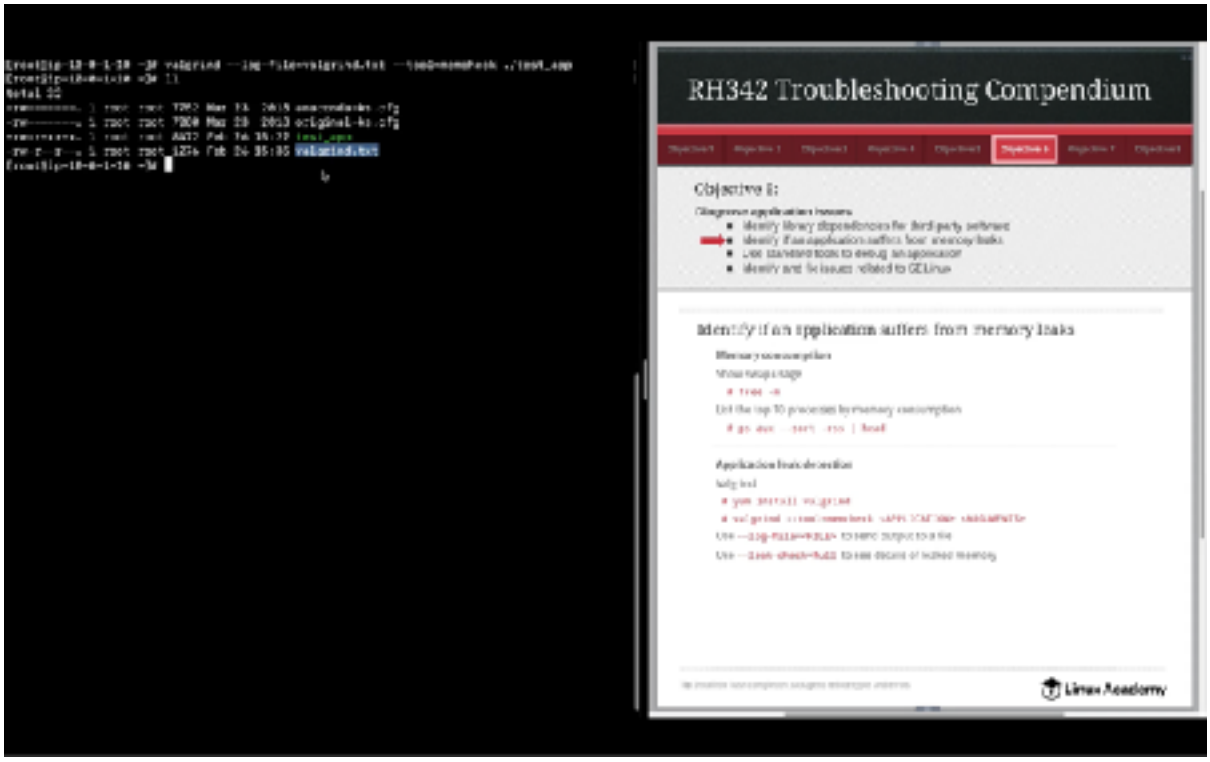

RH342 Troubleshooting Compendium
Objective I:
Identify if an application suffers from memory leaks
Linux Academy

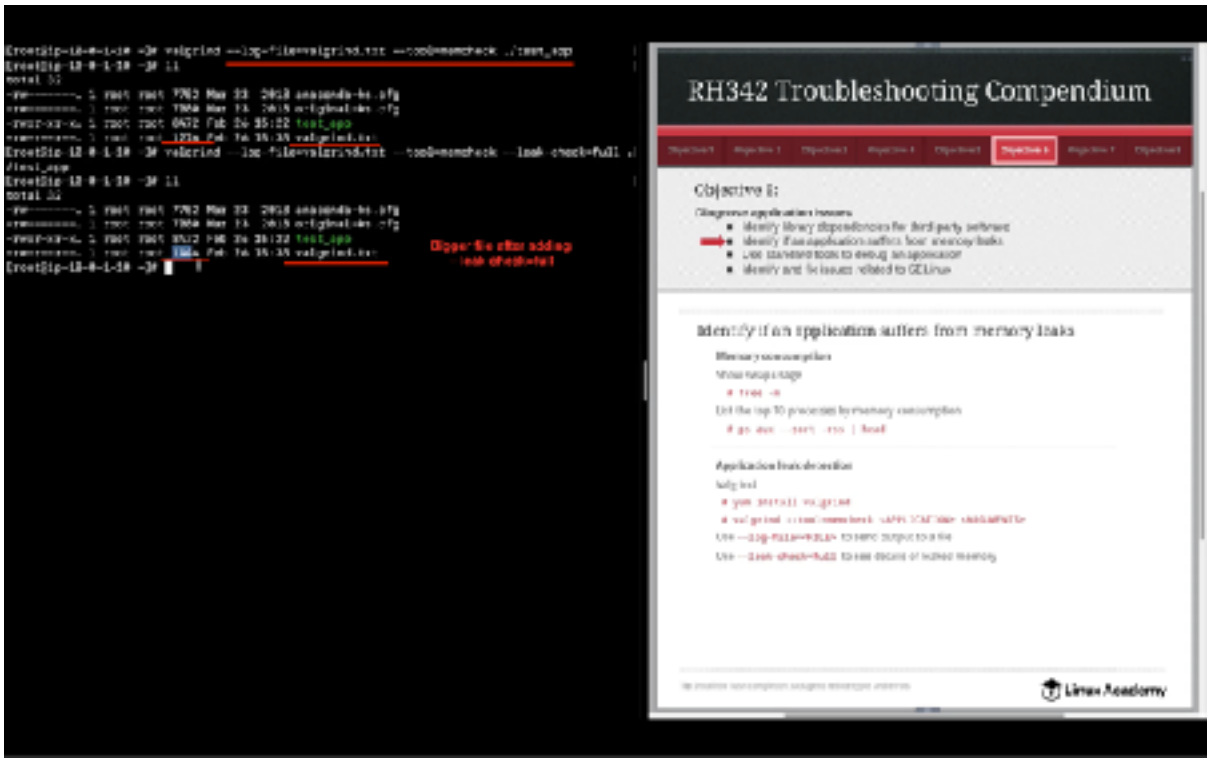

RH342 Troubleshooting Compendium
Objective 5:
Identify if an application suffers from memory leaks
Linux Academy

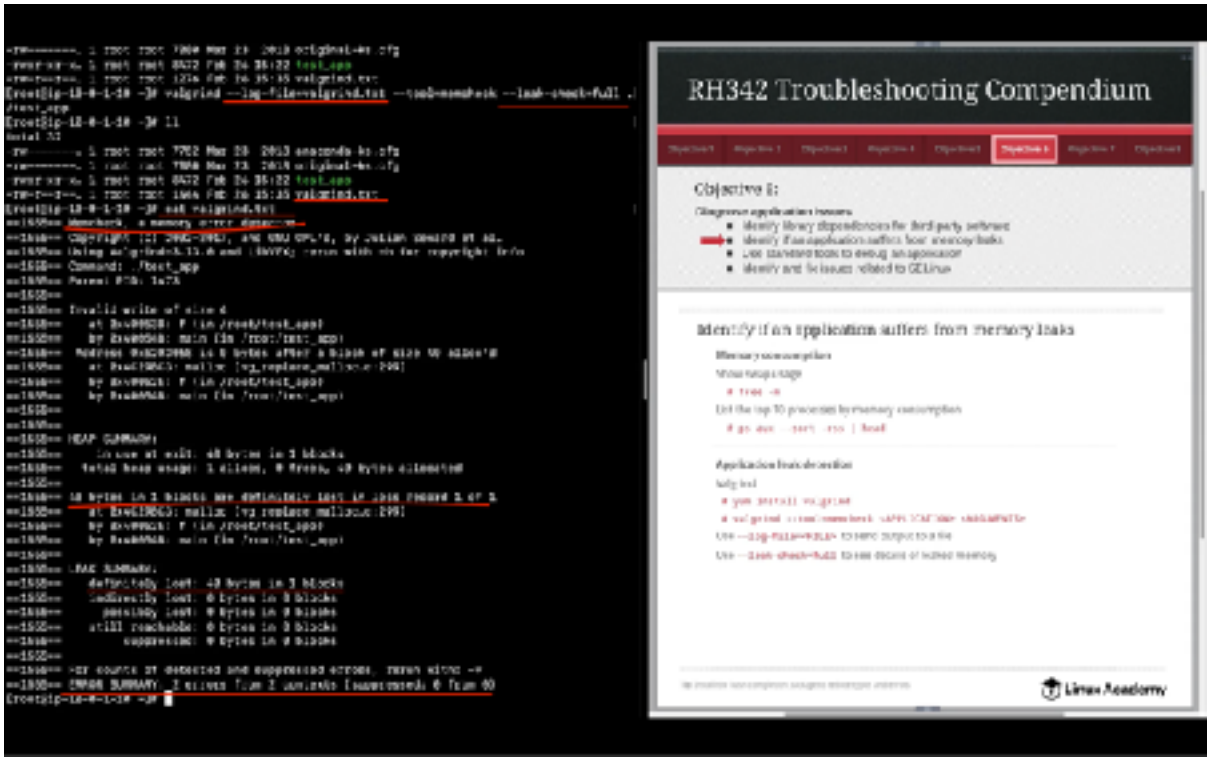

RH342 Troubleshooting Compendium
Objective 5:
Identify if an application suffers from memory leaks
Linux Academy

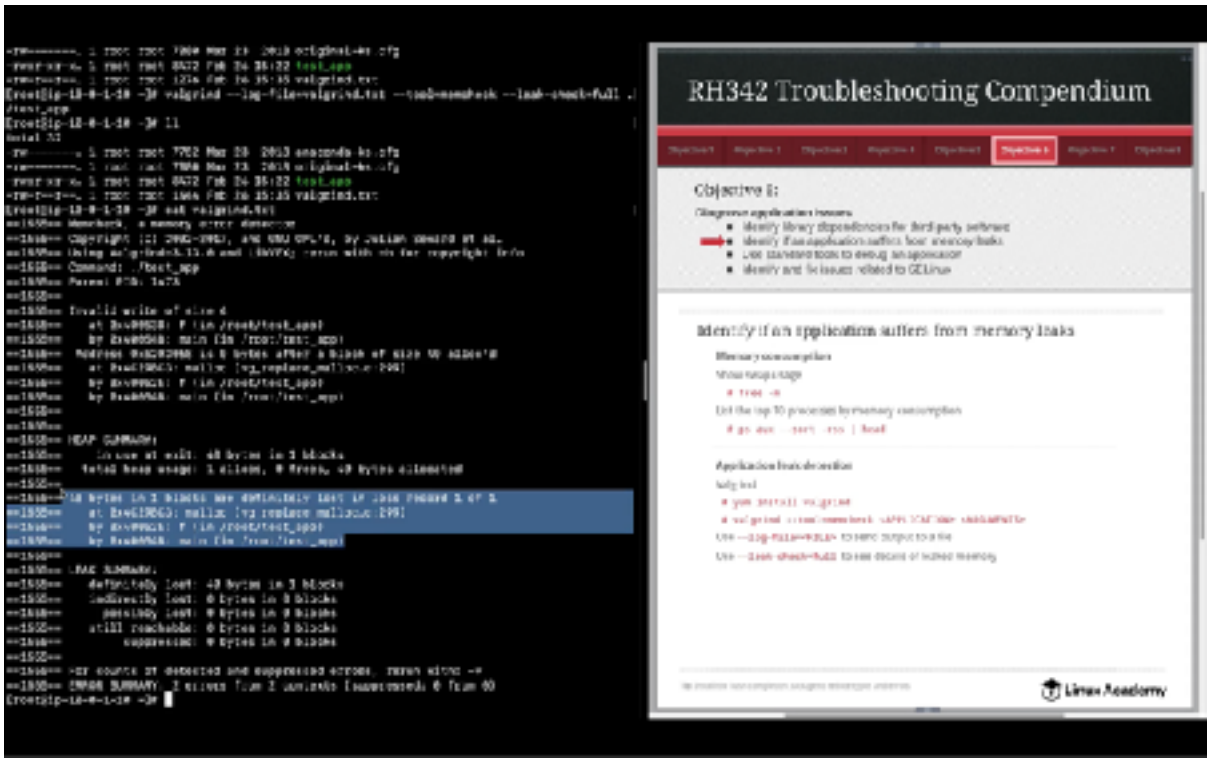

RH342 Troubleshooting Compendium
Objective I:
Identify if an application suffers from memory leaks
Linux Academy


RH342 Troubleshooting Compendium
Objective I:
Identify if an application suffers from memory leaks
Linux Academy


RH342 Troubleshooting Compendium
Objective 1:
Identify if an application suffers from memory leaks
Linux Academy


Bigger file after adding
leak-check=full
RH342 Troubleshooting Compendium
Objective 1:
Identify if an application suffers from memory leaks
Linux Academy





<u>Standard Tolls to debug an application</u>

Strace

Ltrace

Objective 6:

Diagnose application issues

- Identify library dependencies for third-party software
- Identify if an application suffers from memory leaks
- Use standard tools to debug an application
- Identify and fix issues related to SELinux

Use standard tools to debug an application

strace

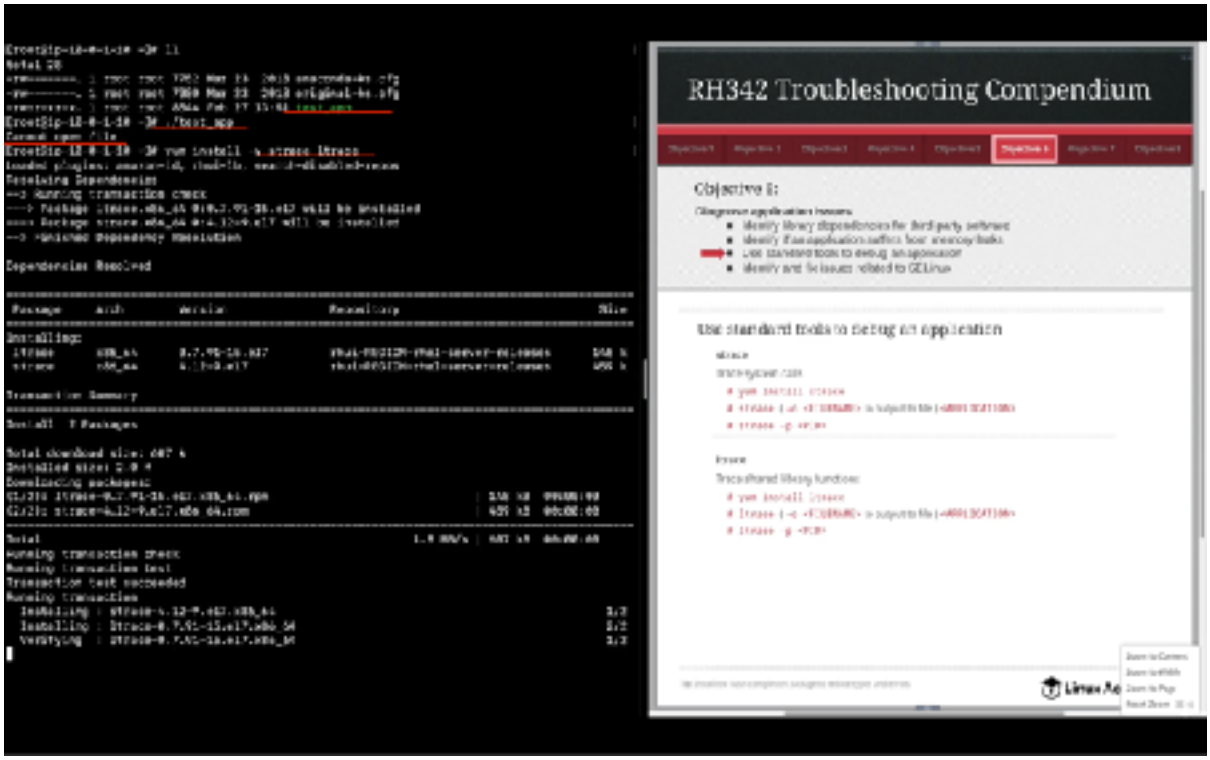
Trace system calls

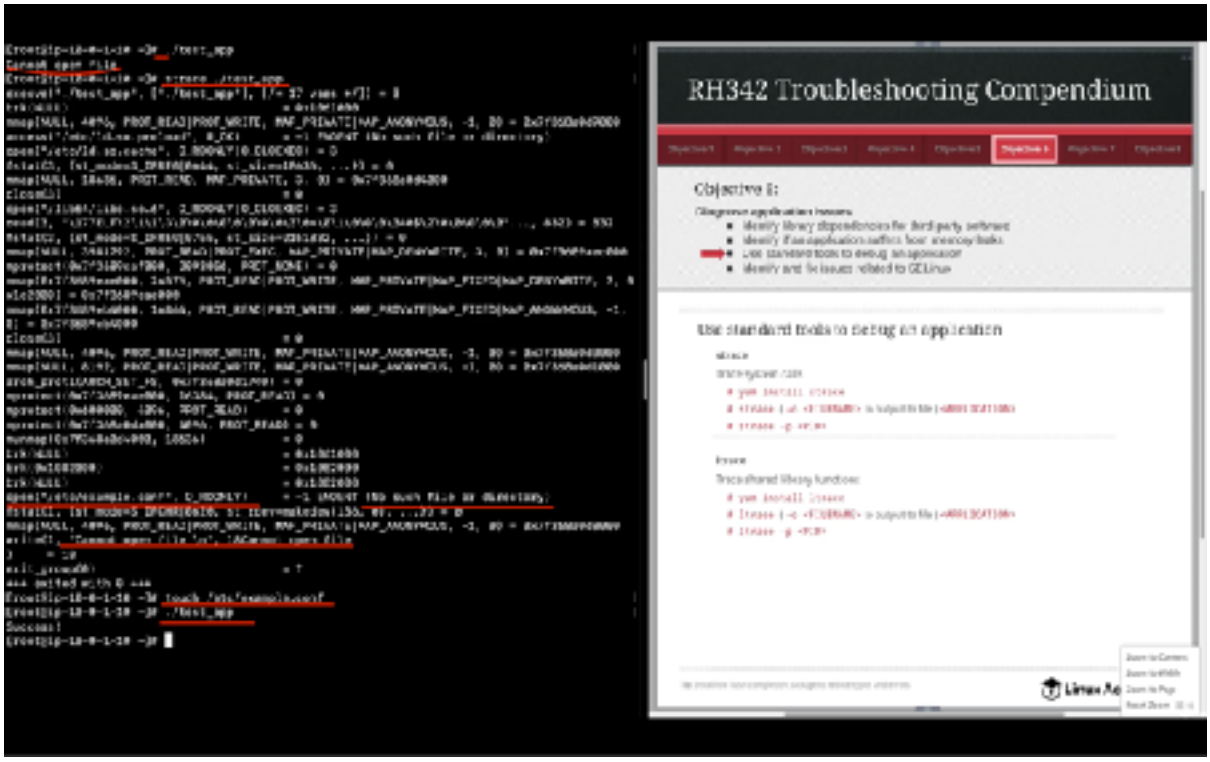
```
# yum install strace
# strace [ -o <FILENAME> to output to file ] <APPLICATION>
# strace -p <PID>
```

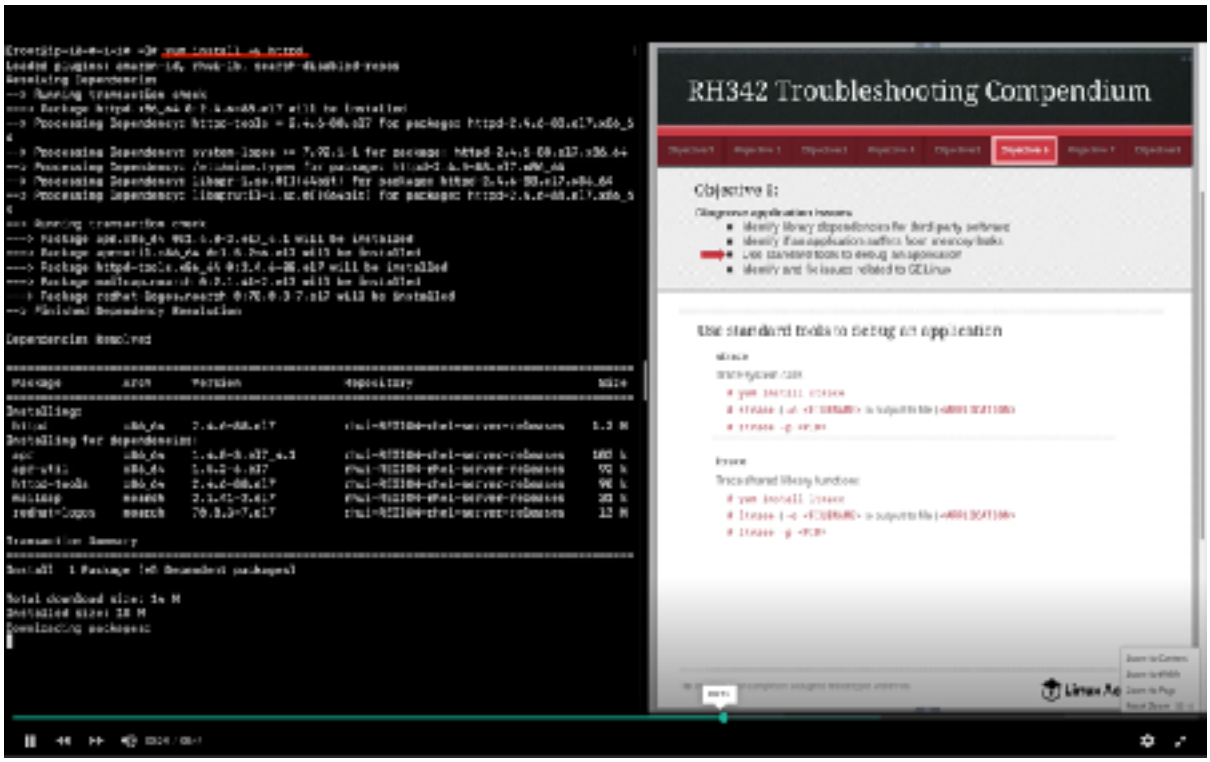
ltrace

Trace shared library functions

```
# yum install ltrace
# ltrace [ -o <FILENAME> to output to file ] <APPLICATION>
# ltrace -p <PID>
```




RH342 Troubleshooting Compendium
Objective I:
Use standard tools to debug an application


RH342 Troubleshooting Compendium
Objective I:
Use standard tools to debug an application

RH342 Troubleshooting Compendium

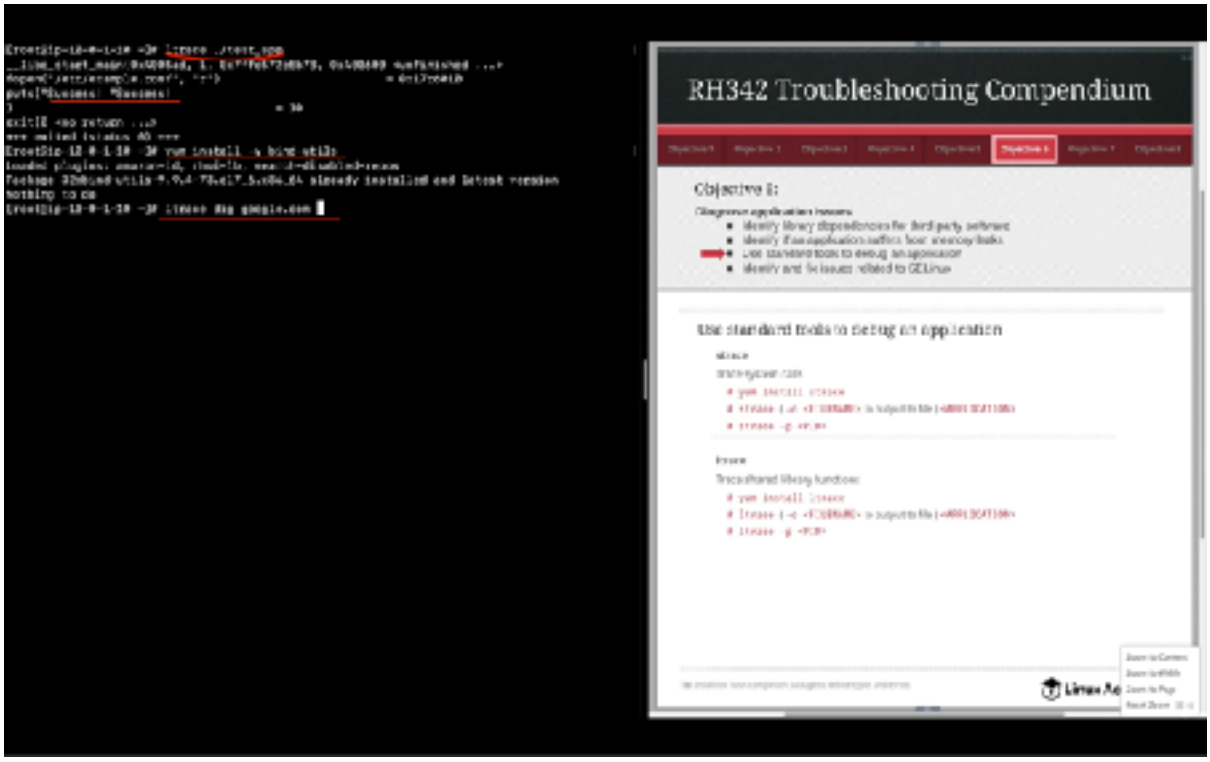
Objective 1:

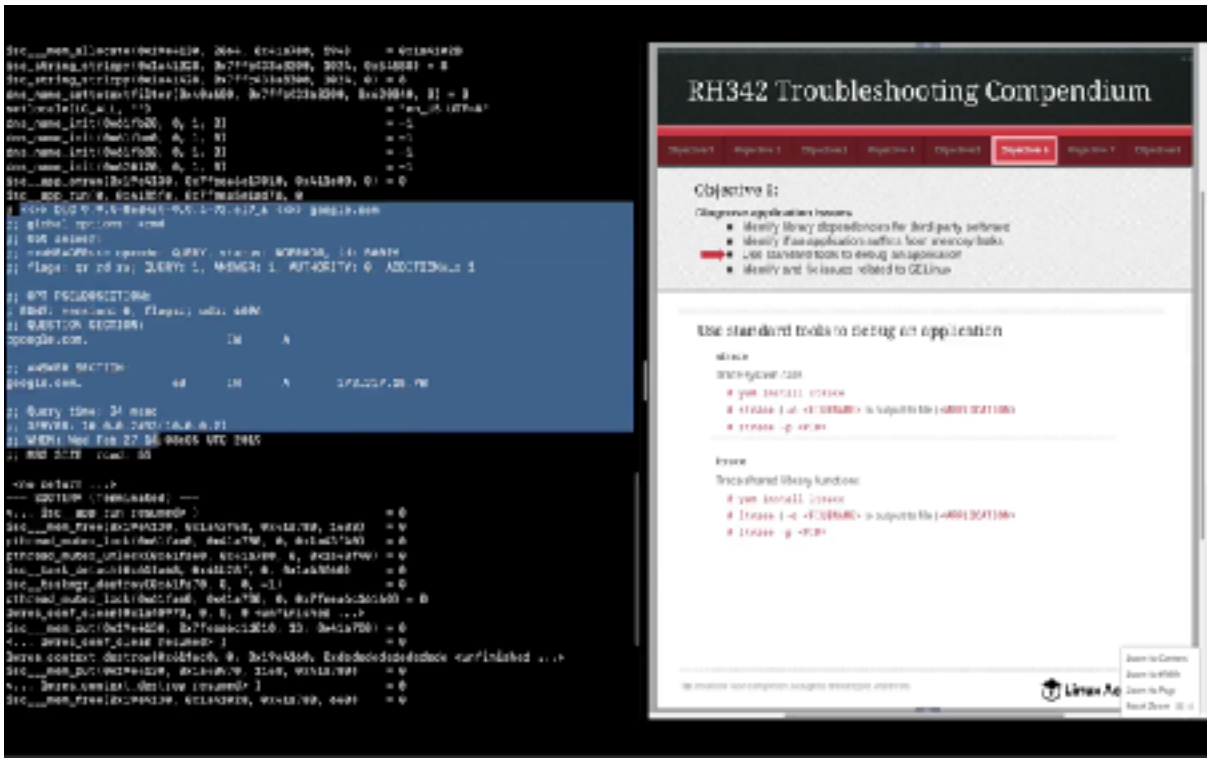
Use standard tools to debug an application

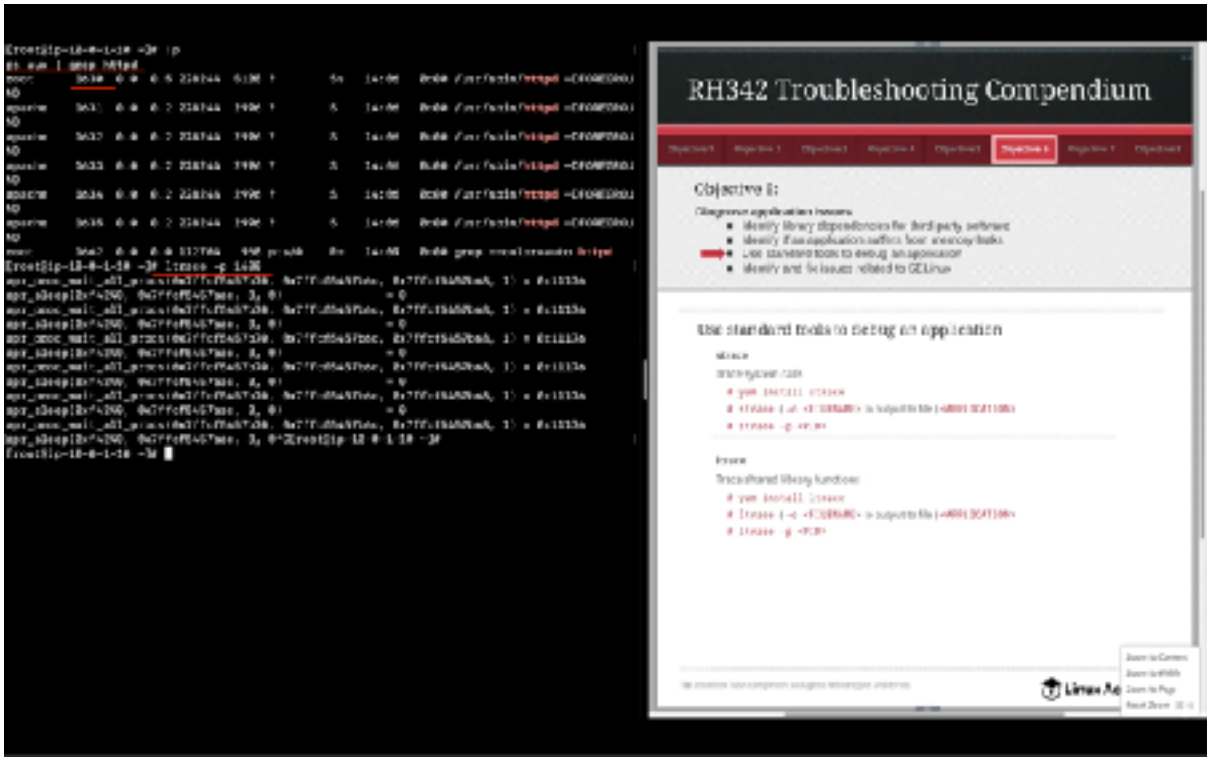
RH342 Troubleshooting Compendium

Objective 1:

Use standard tools to debug an application


RH342 Troubleshooting Compendium
Objective 5:
Use standard tools to debug an application


RH342 Troubleshooting Compendium
Objective 5:
Use standard tools to debug an application



Fixing Issues related to SELinux

Objective 6:

Diagnose application issues

- Identify library dependencies for third-party software
- Identify if an application suffers from memory leaks
- Use standard tools to debug an application
- Identify and fix issues related to SELinux

Identify and fix issues related to SELinux

Identify

SELinux log location: /var/log/audit/audit.log

Search for recent AVC errors

```
# ausearch -m avc -ts recent
```

Run the log through sealert

```
# sealert -a /var/log/audit/audit.log
```

Context lables

```
# seinfo -t
```

Booleans

```
# getsebool -a
```

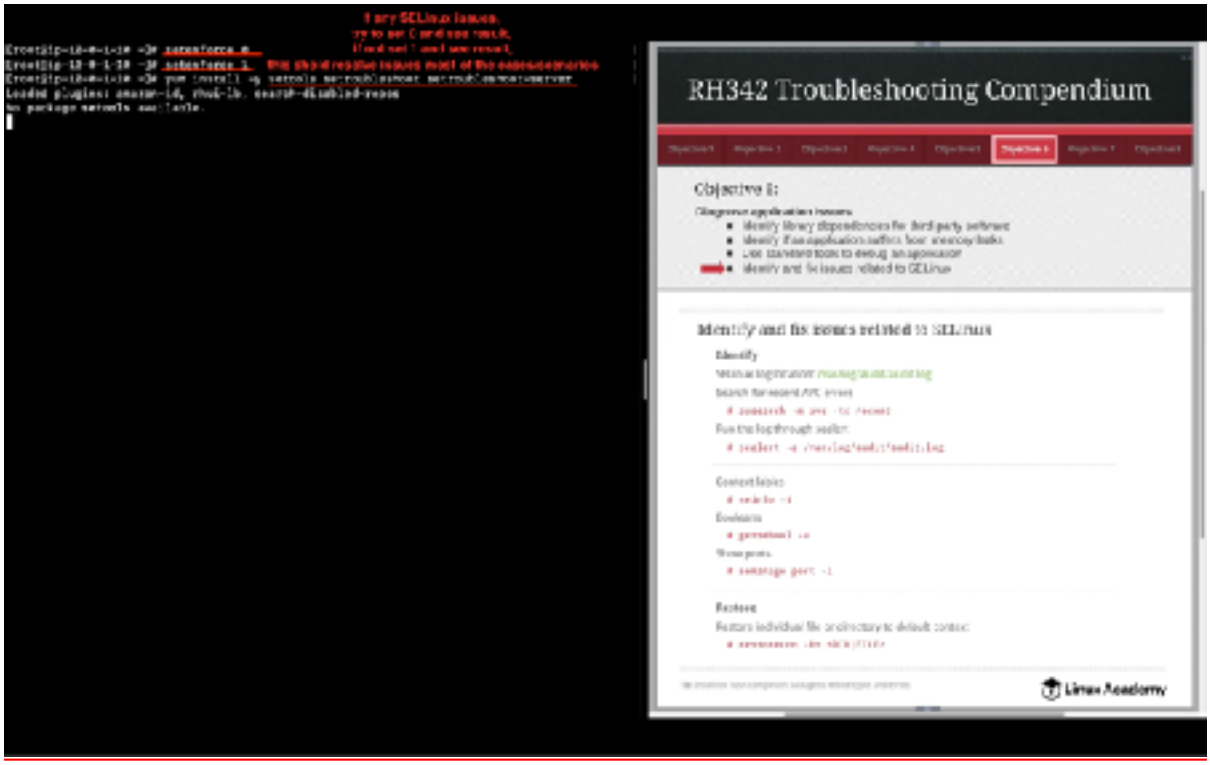
Show ports

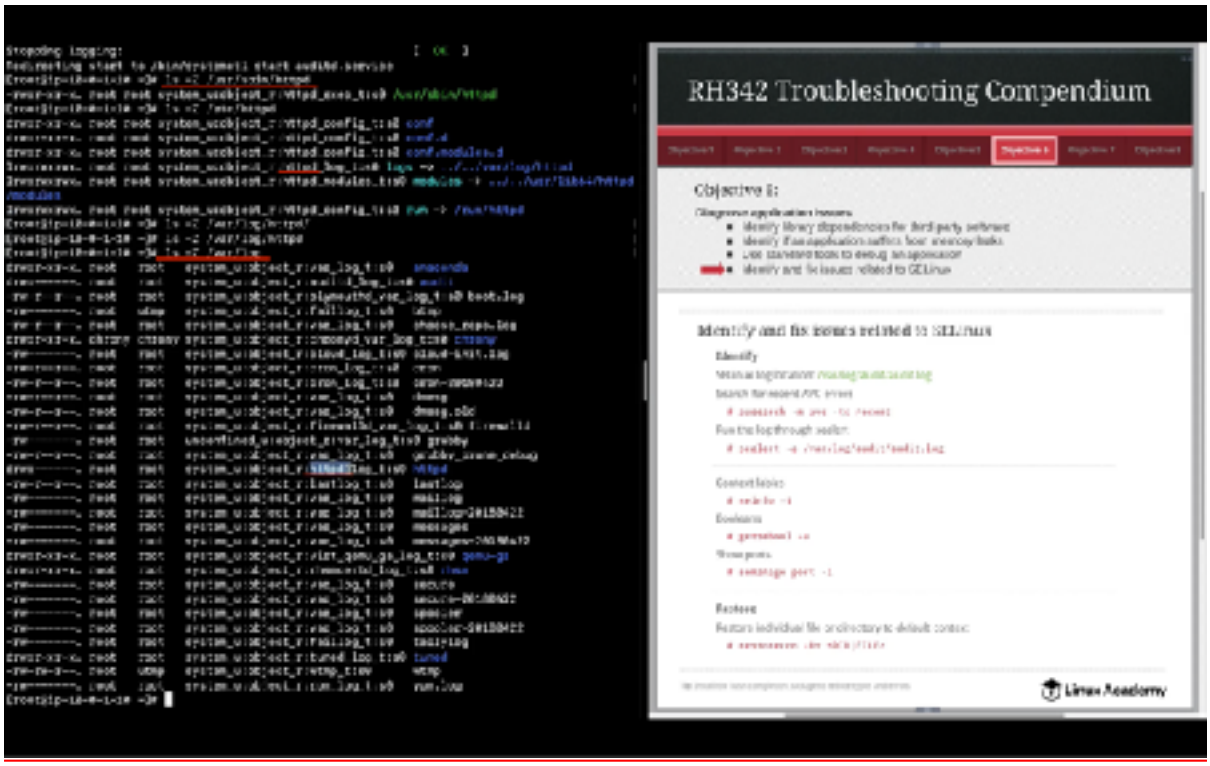
```
# semanage port -l
```

Restore

Restore individual file or directory to default context

```
# restorecon -Rv <DIR|FILE>
```


RH342 Troubleshooting Compendium
Objective I:
Identify and fix issues related to SELinux
Linux Academy


RH342 Troubleshooting Compendium
Objective I:
Identify and fix issues related to SELinux
Linux Academy

RH342 Troubleshooting Compendium

Objective 1:

Identify and fix issues related to SELinux

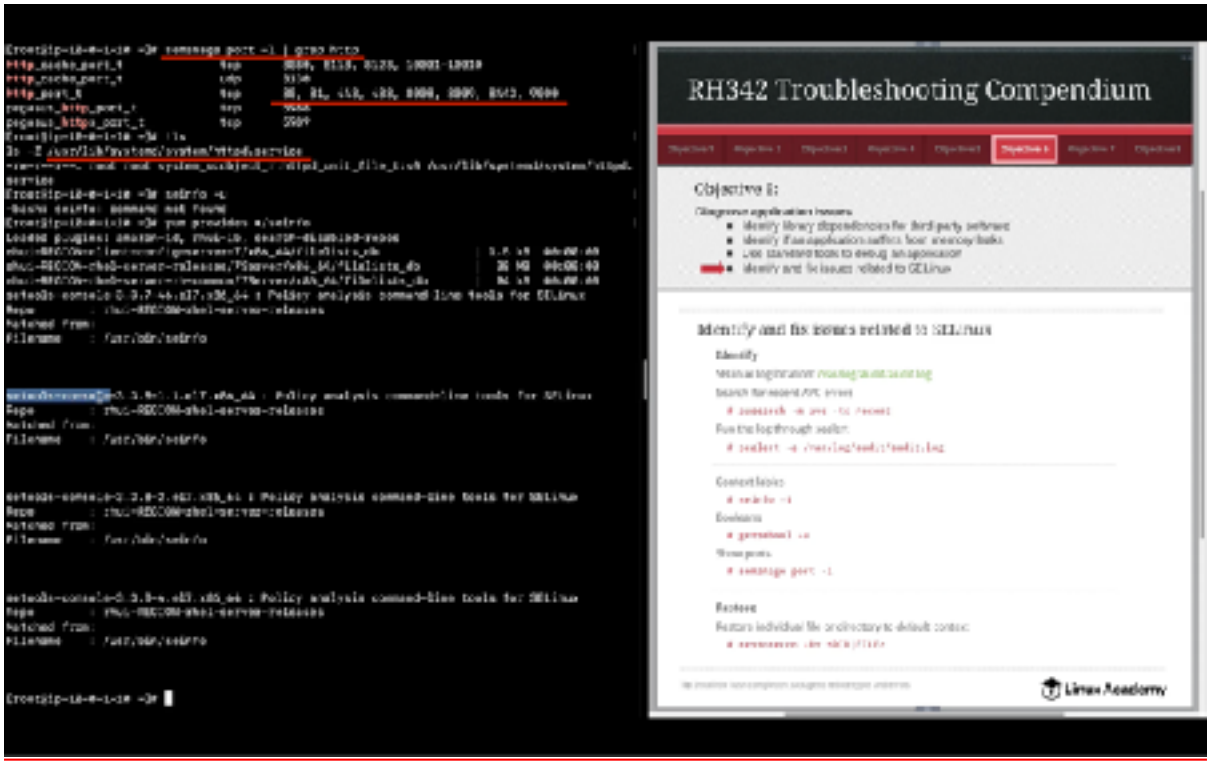
Linux Academy

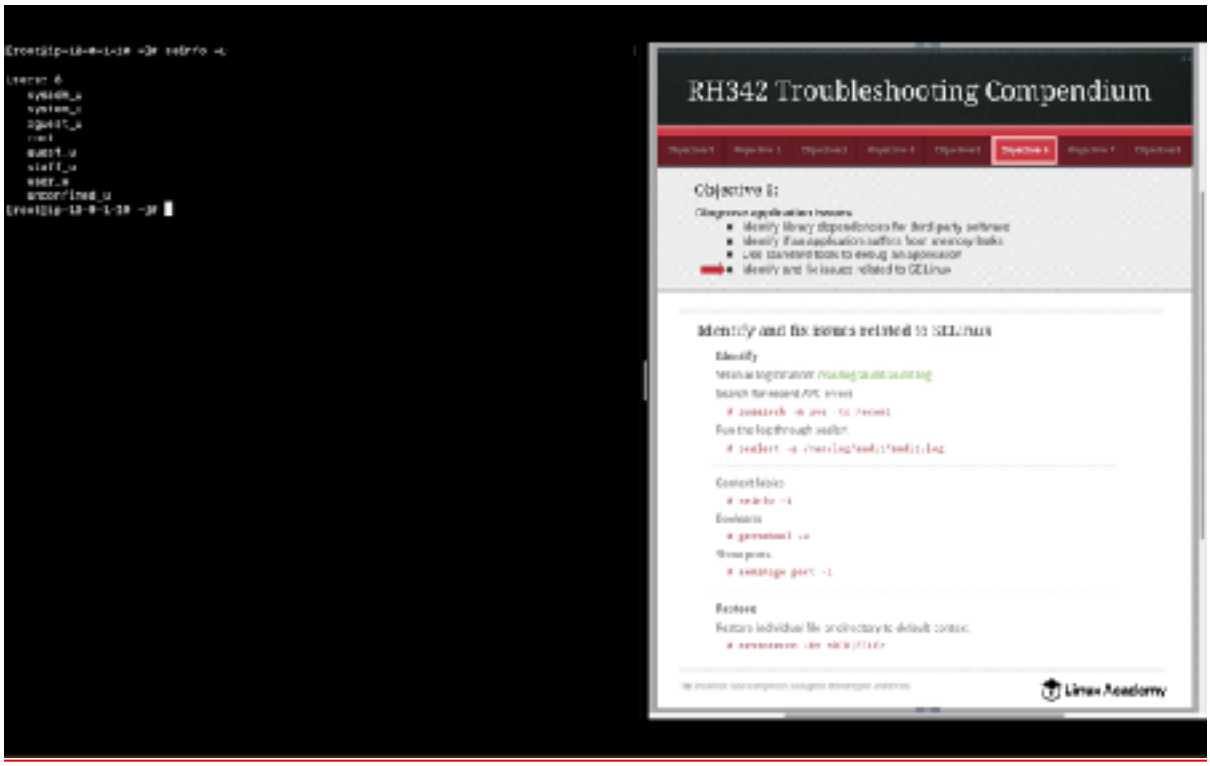
RH342 Troubleshooting Compendium

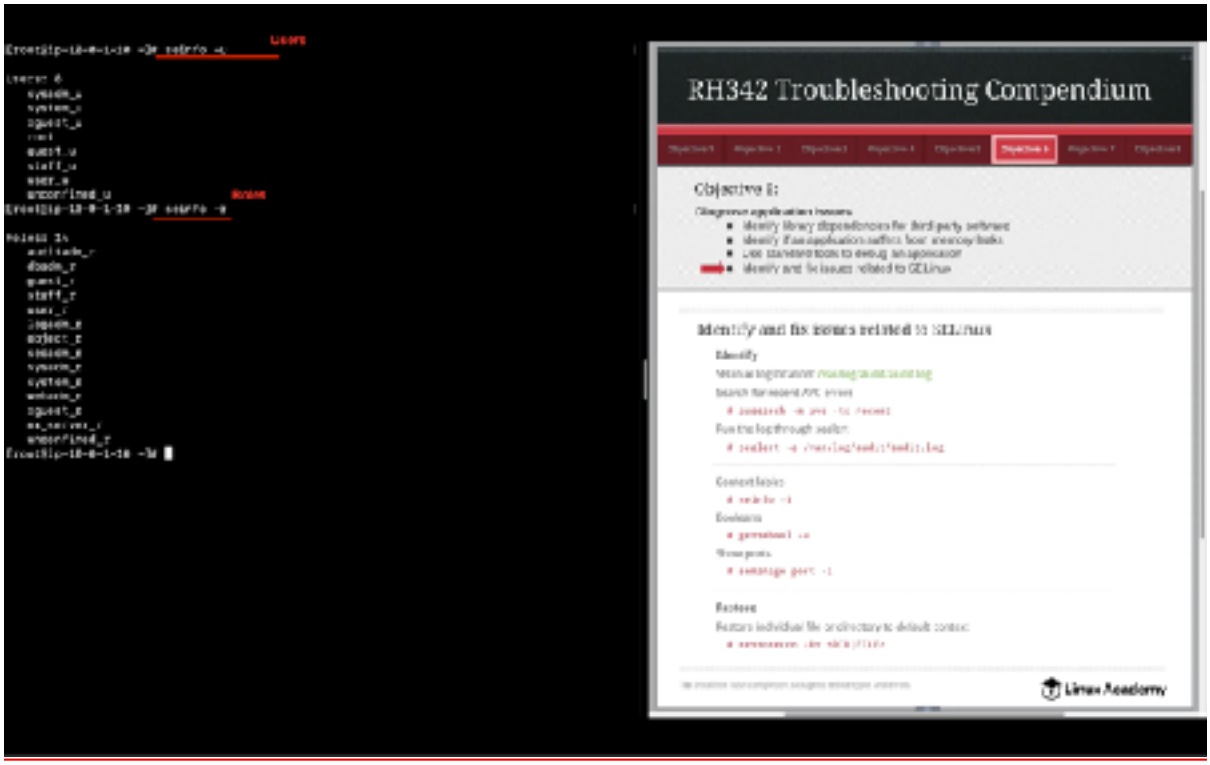
Objective 1:

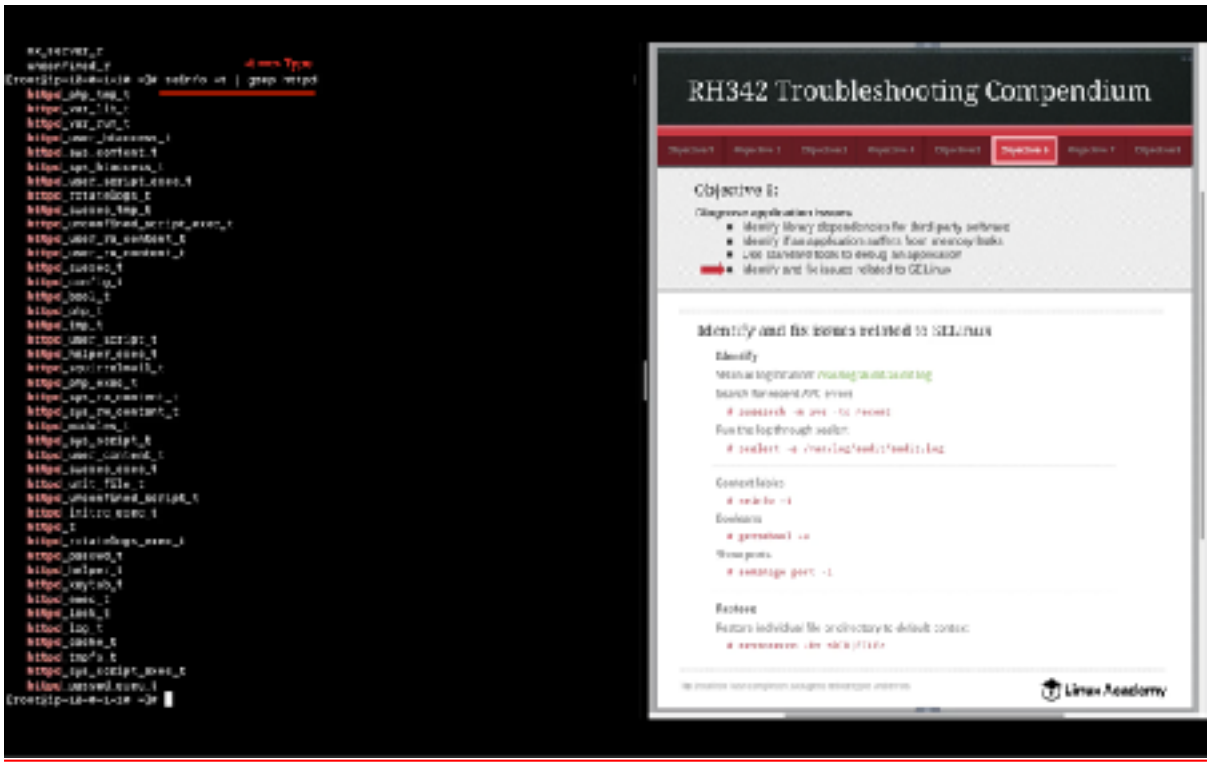
Identify and fix issues related to SELinux

Linux Academy


RH342 Troubleshooting Compendium
Objective 5:
Identify and fix issues related to SELinux
Linux Academy


RH342 Troubleshooting Compendium
Objective 5:
Identify and fix issues related to SELinux
Linux Academy


Users
Roles
RH342 Troubleshooting Compendium
Objective 1:
Identify and fix issues related to SELinux
Linux Academy


Access Type
RH342 Troubleshooting Compendium
Objective 1:
Identify and fix issues related to SELinux
Linux Academy

RH342 Troubleshooting Compendium

Objective 5:

Identify and fix issues related to SELinux

Identify

Restore

Linux Academy

RH342 Troubleshooting Compendium

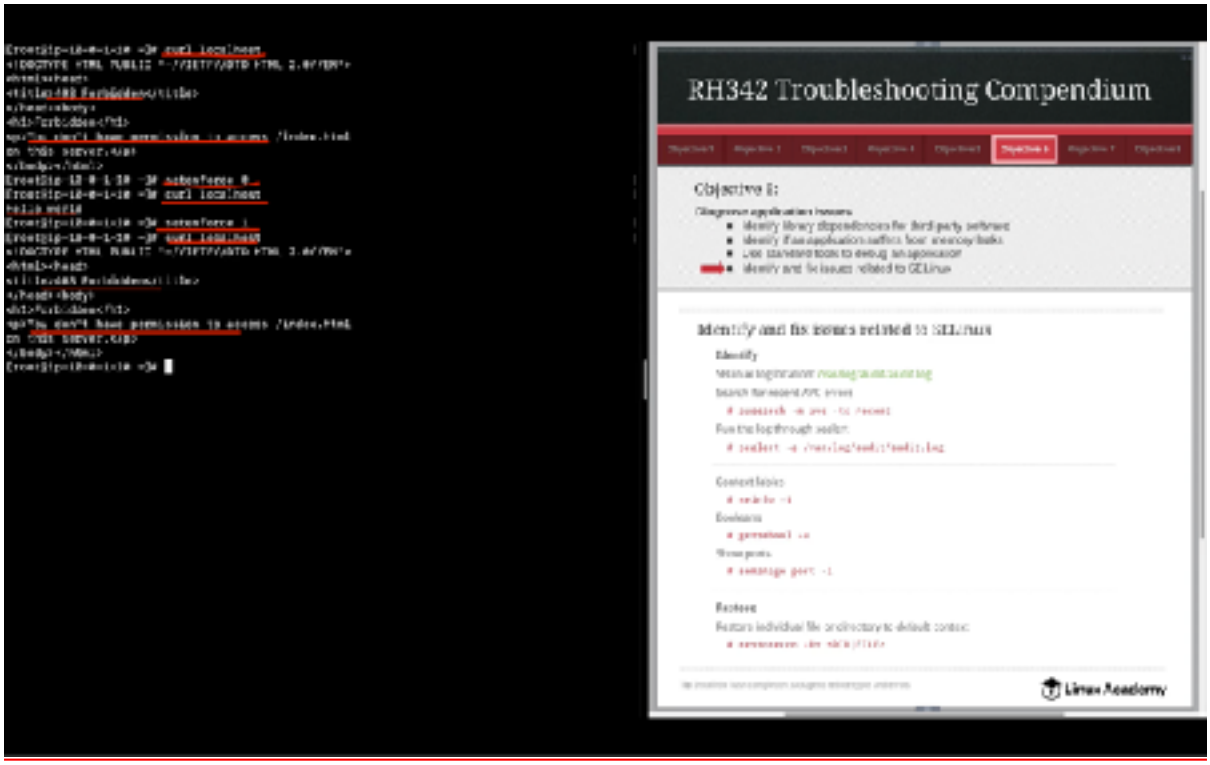
Objective 5:

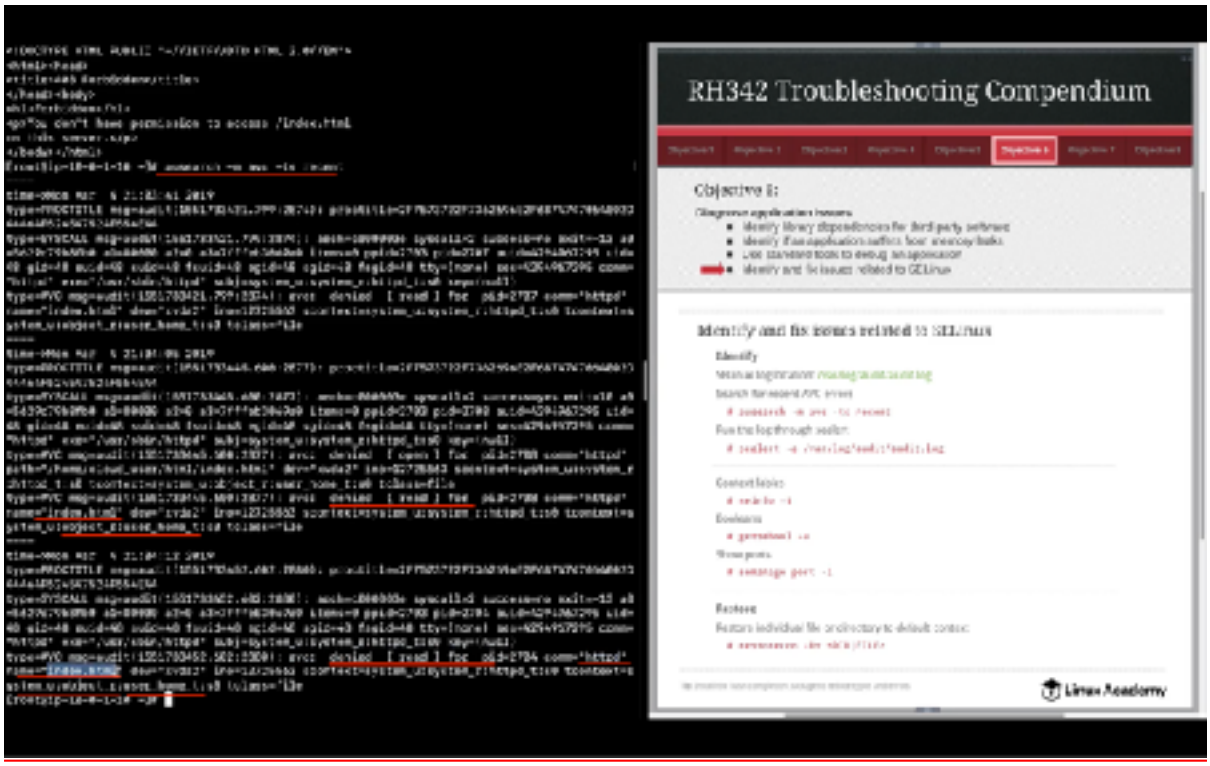
Identify and fix issues related to SELinux

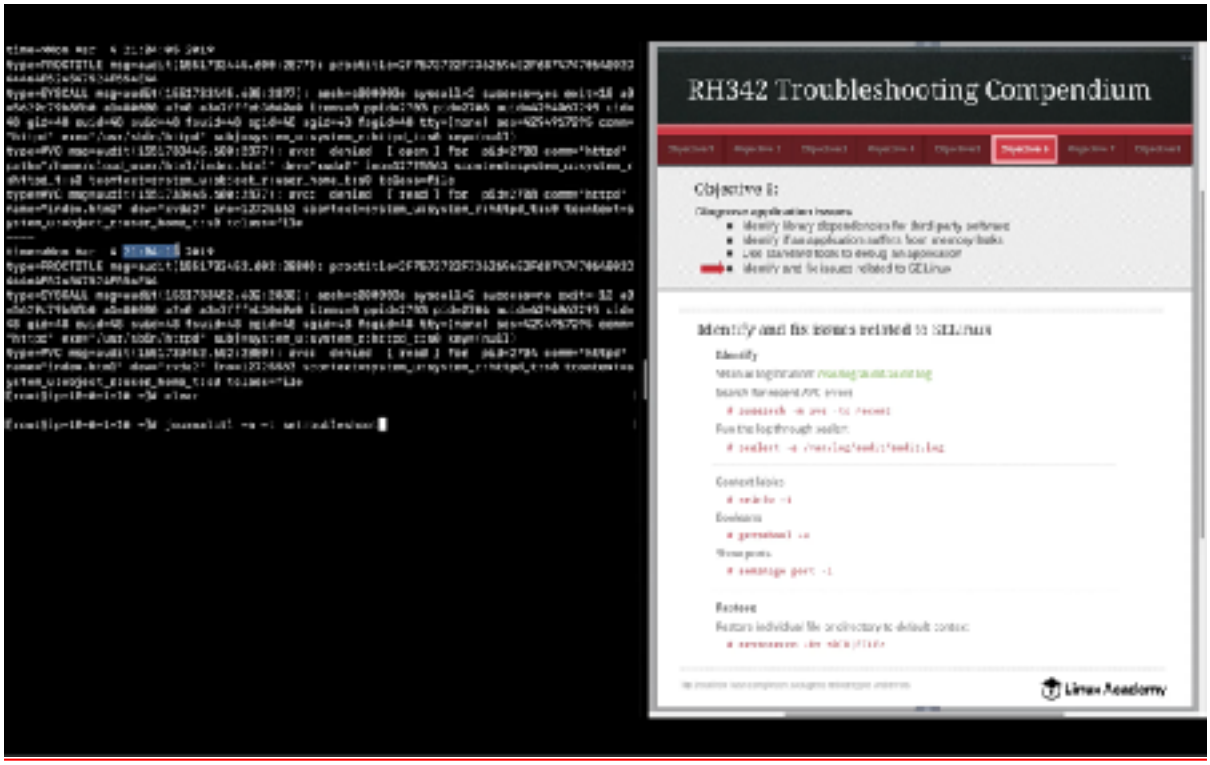
Identify

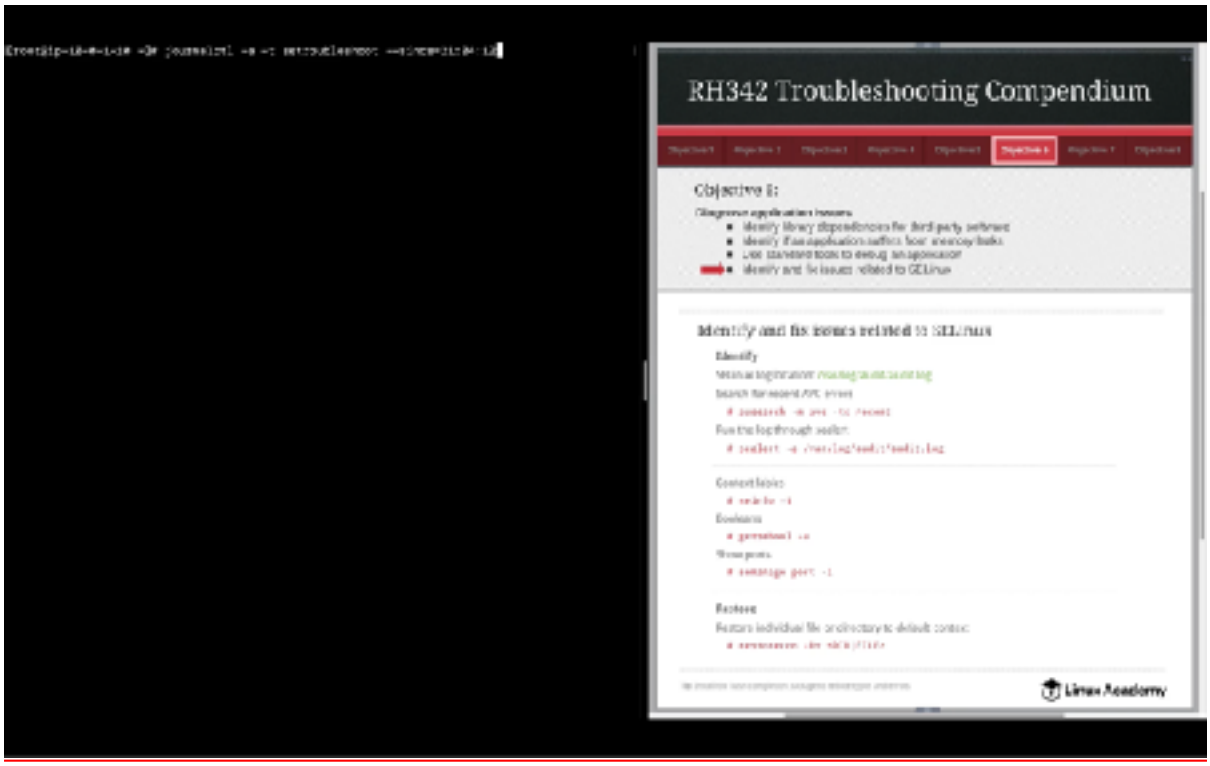
Restore

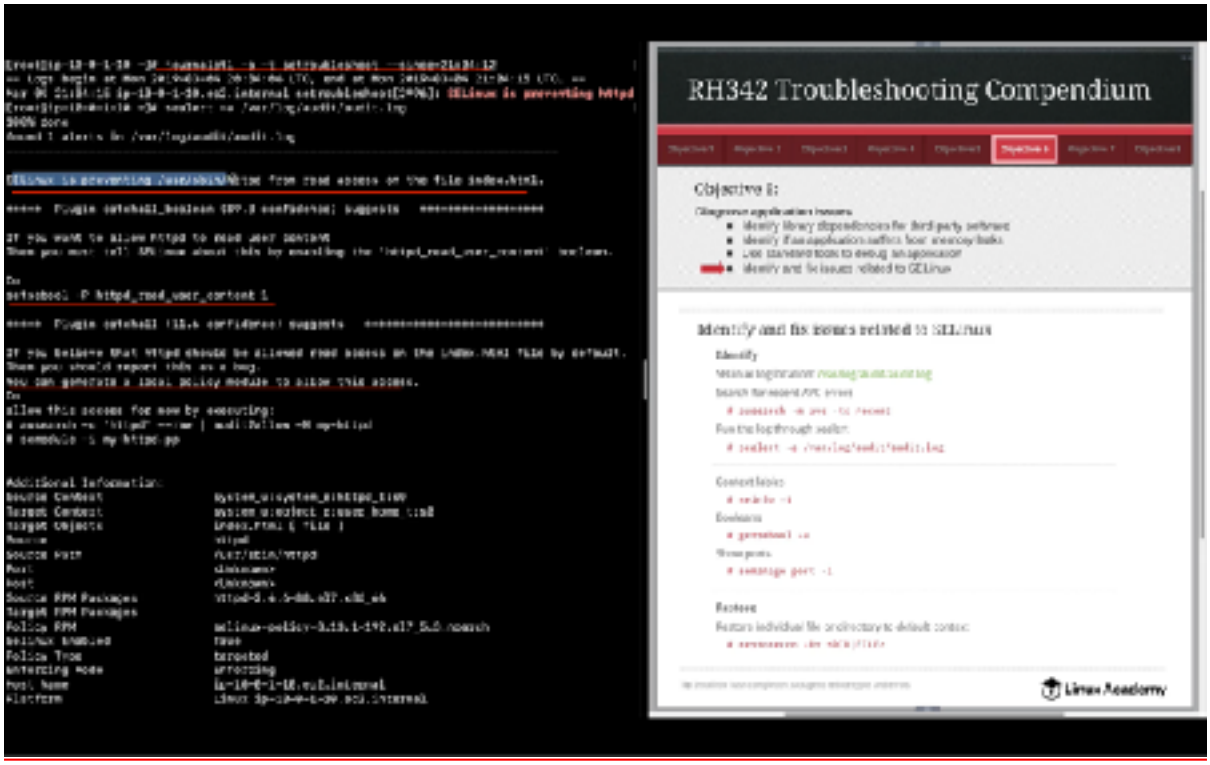
Linux Academy

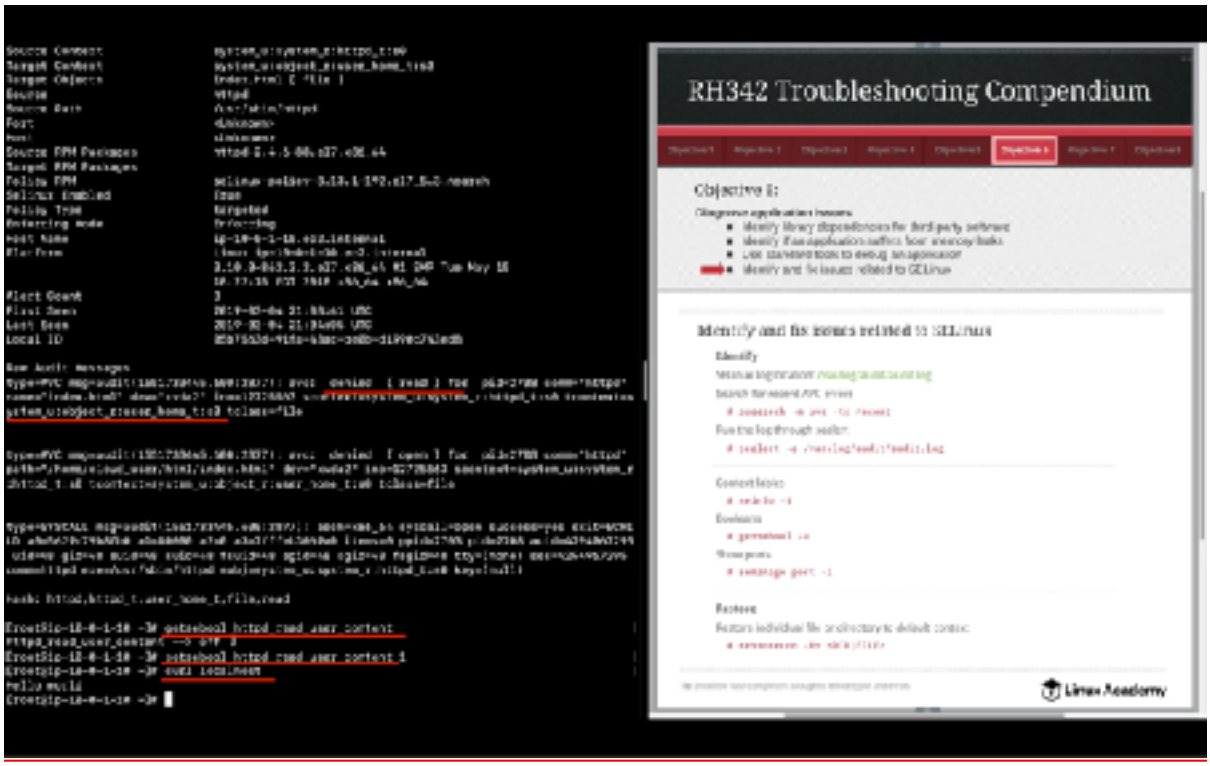

RH342 Troubleshooting Compendium
Objective I:
Identify and fix issues related to SELinux
Identify
Restore
Linux Academy

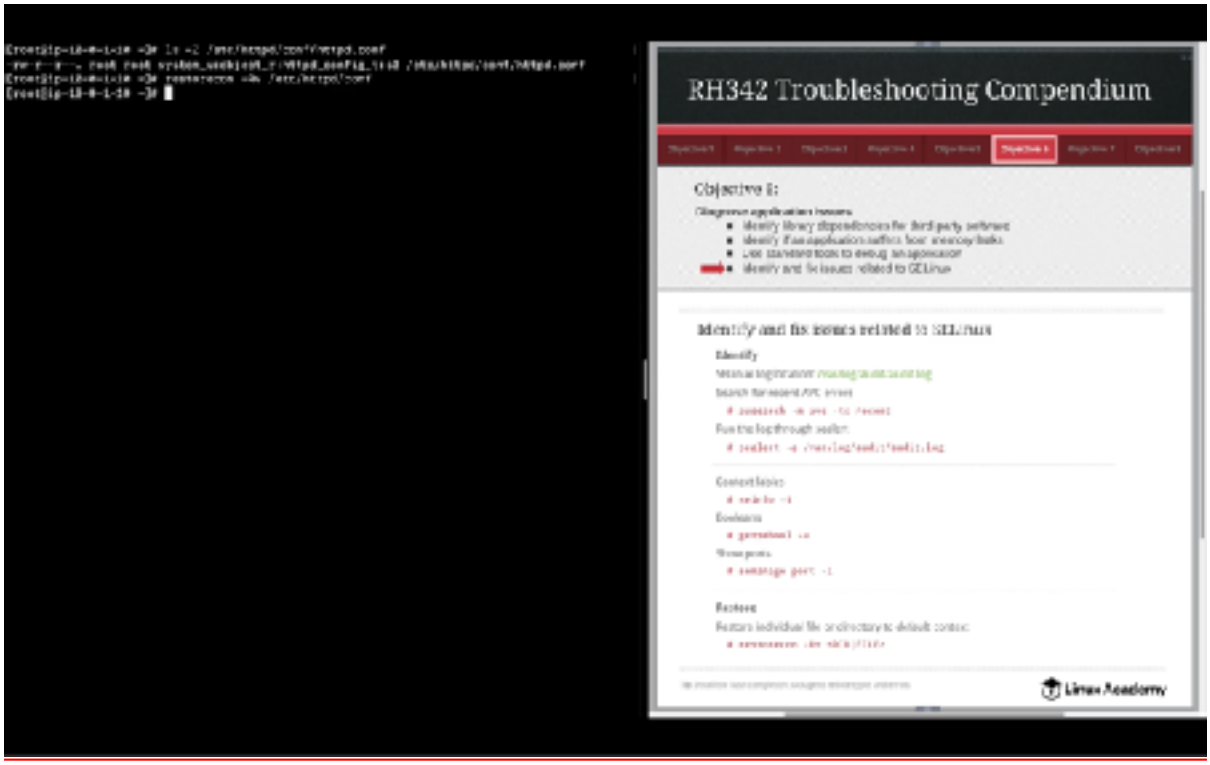

RH342 Troubleshooting Compendium
Objective I:
Identify and fix issues related to SELinux
Identify
Restore
Linux Academy


RH342 Troubleshooting Compendium
Objective 5:
Identify and fix issues related to SELinux
Linux Academy

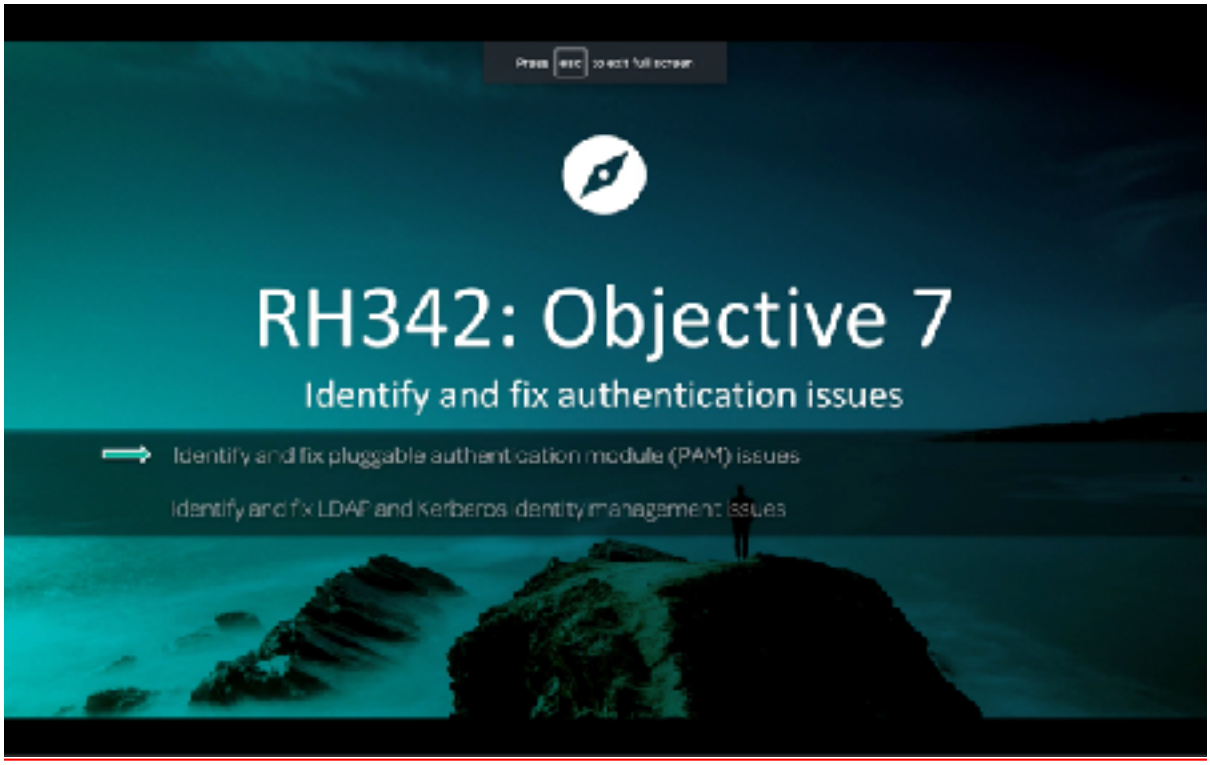

RH342 Troubleshooting Compendium
Objective 5:
Identify and fix issues related to SELinux
Linux Academy


RH342 Troubleshooting Compendium
Objective I:
Identify and fix issues related to SELinux
Linux Academy


RH342 Troubleshooting Compendium
Objective I:
Identify and fix issues related to SELinux
Linux Academy


RH342 Troubleshooting Compendium
Objective 1:
Identify and fix issues related to SELinux
Linux Academy

PAM


Press esc to exit full screen
RH342: Objective 7
Identify and fix authentication issues
Identify and fix pluggable authentication module (PAM) issues
Identify and fix LDAP and Kerberos identity management issues

Objective 7:

Identify and fix authentication issues

- Identify and fix pluggable authentication module (PAM) issues
- Identify and fix LDAP and Kerberos identity management issues

Identify and fix pluggable authentication module issues

Identify

authconfig settings: /etc/sysconfig/authconfig

Log file: /var/log/secure

PAM documentation: /usr/share/doc/pam-*/txts/

Service config files: /etc/pam.d /<FILE>

Troubleshoot

PAM file format

```
type control module-path [module-arguments]
```

Type: account, auth, password, session

Control: required, requisite, sufficient, optional, include, substack, [value=action]

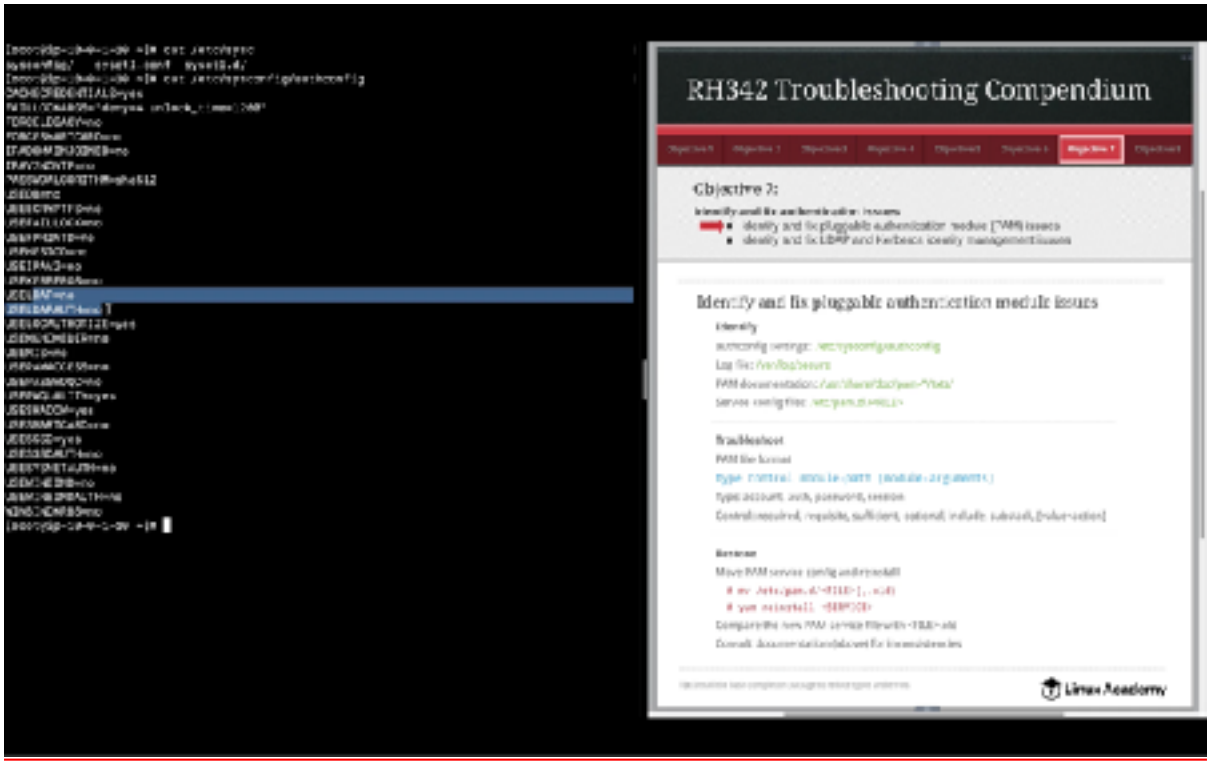
Restore

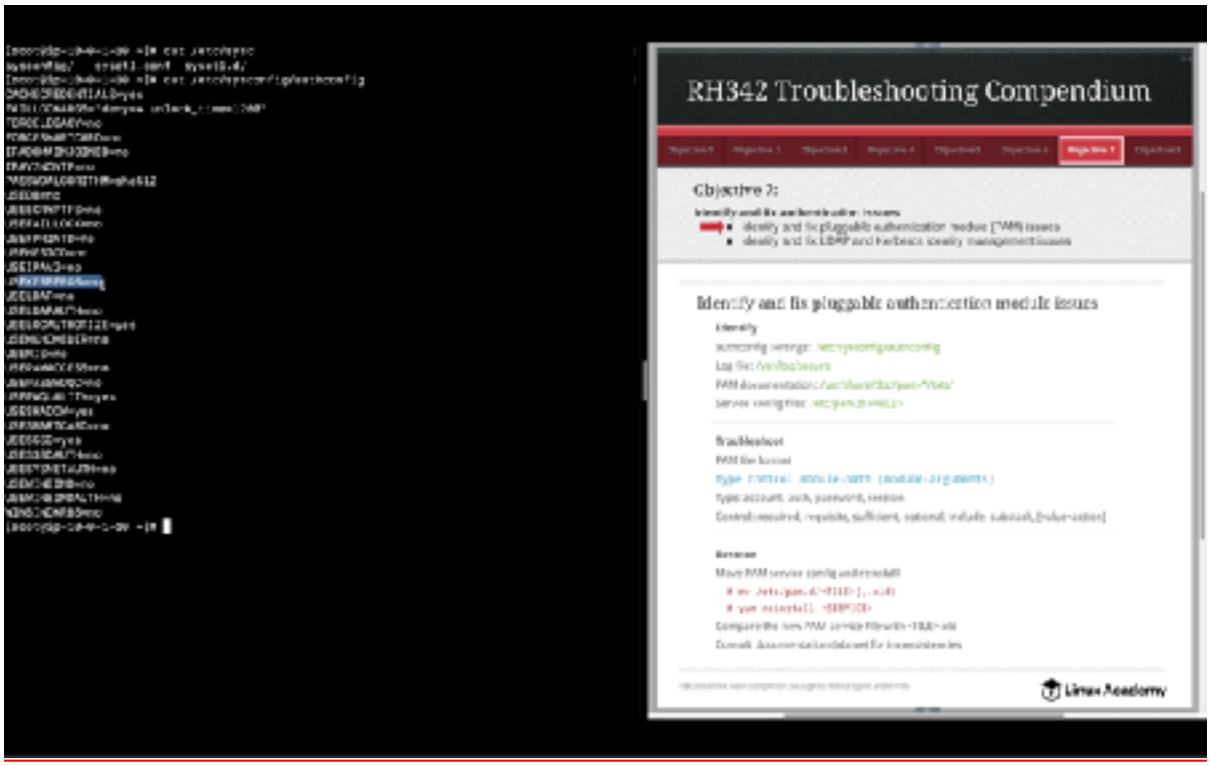
Move PAM service config and reinstall

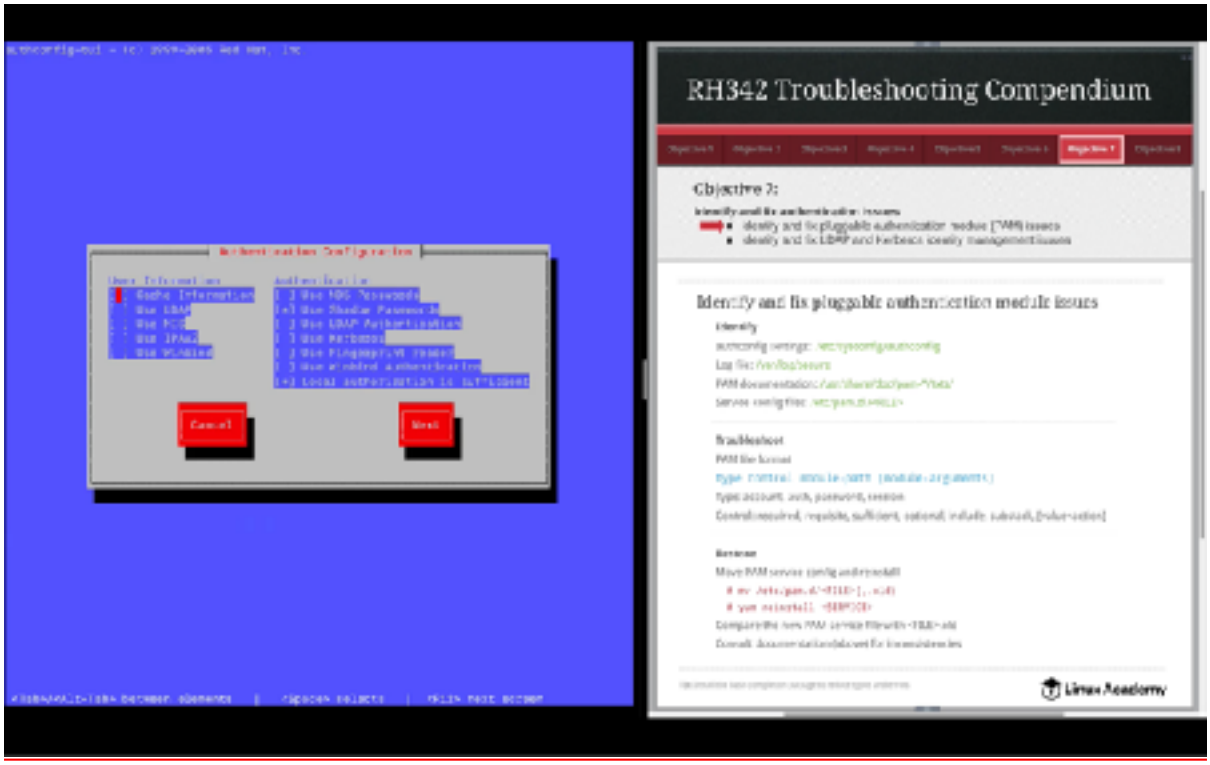
```
# mv /etc/pam.d/<FILE>{,.old}
# yum reinstall <SERVICE>
```

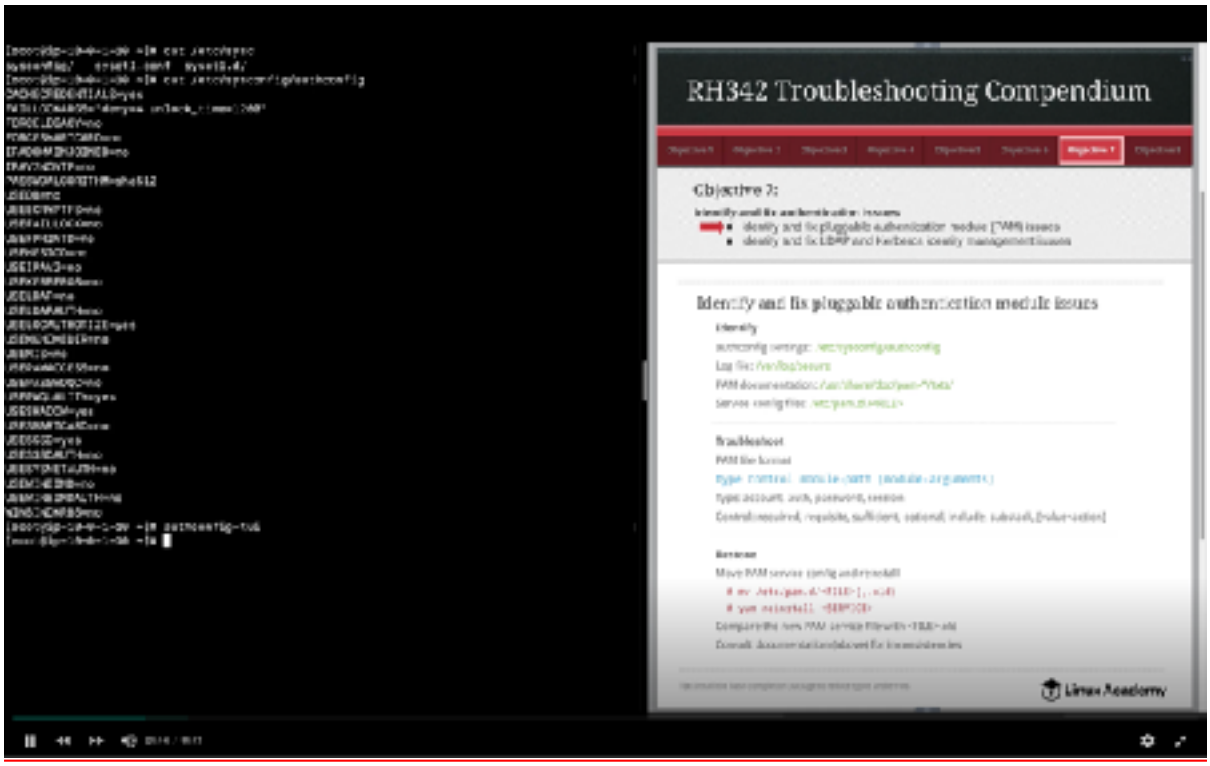
Compare the new PAM service file with <FILE>.old

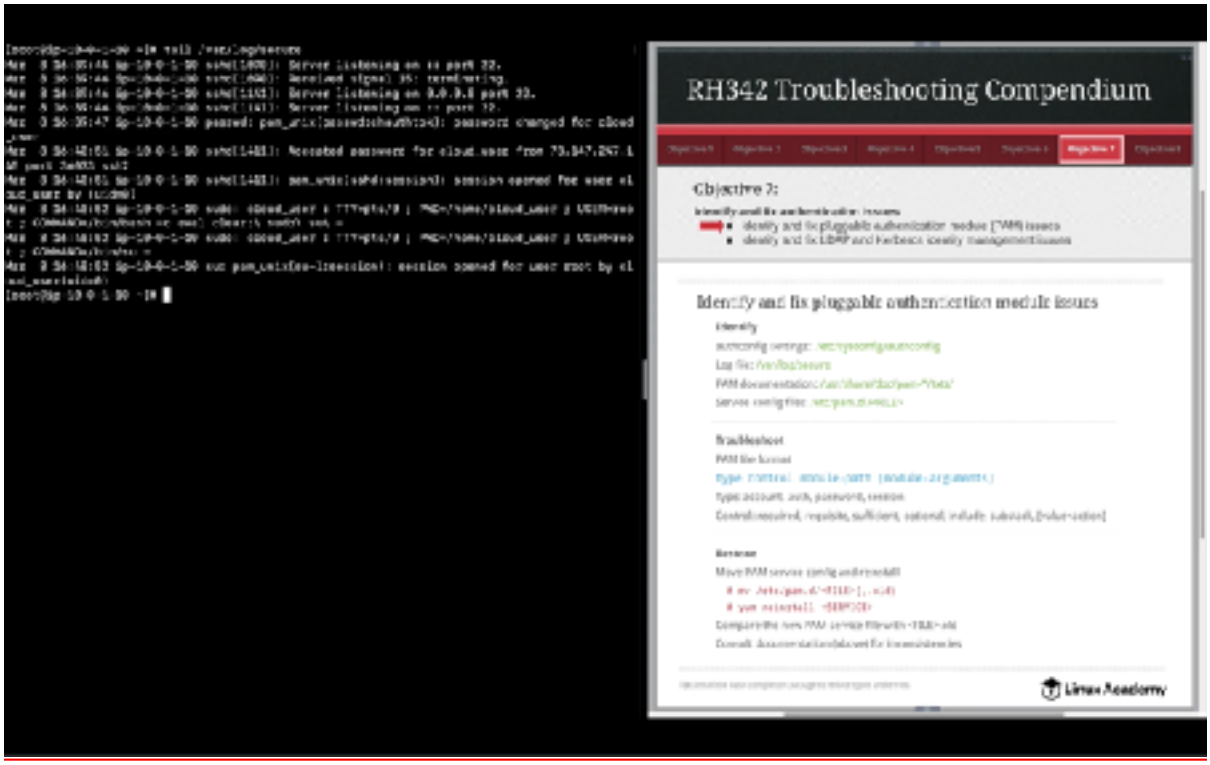
Consult documentation (above) for inconsistencies

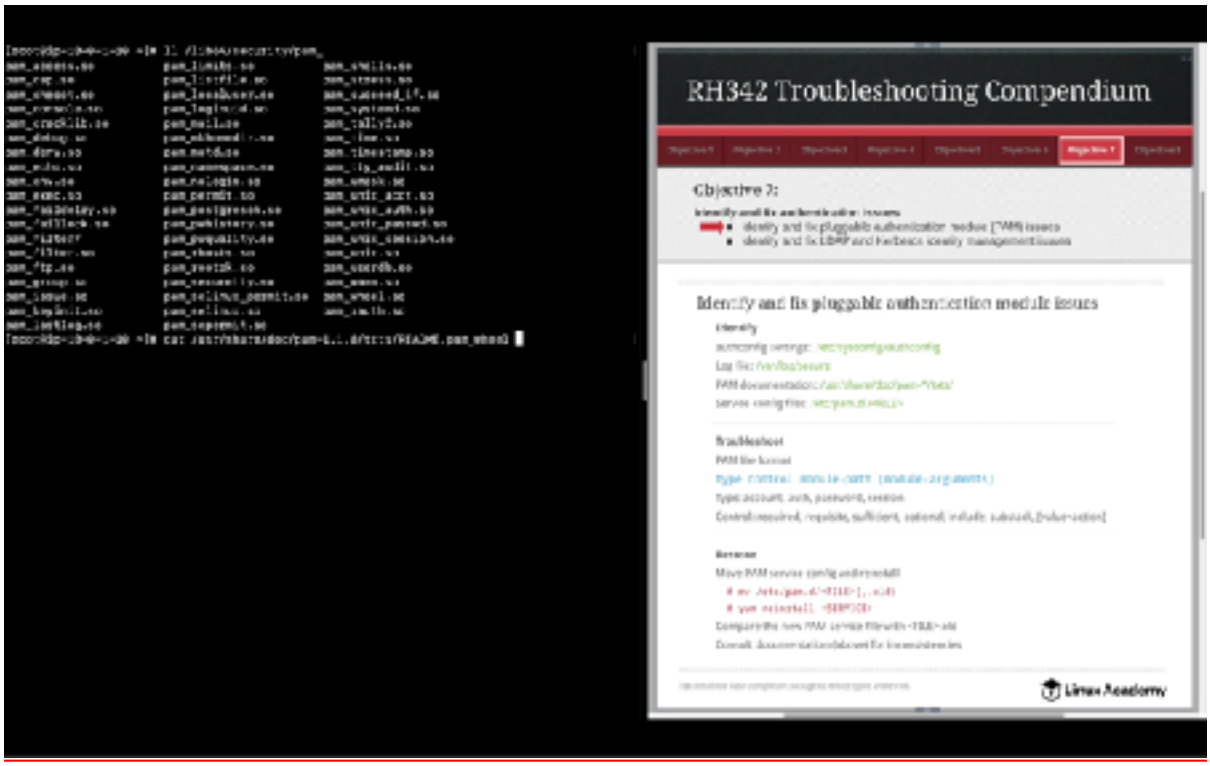

RH342 Troubleshooting Compendium
Objective 7:
Identify and fix pluggable authentication module issues
Troubleshoot
Linux Academy

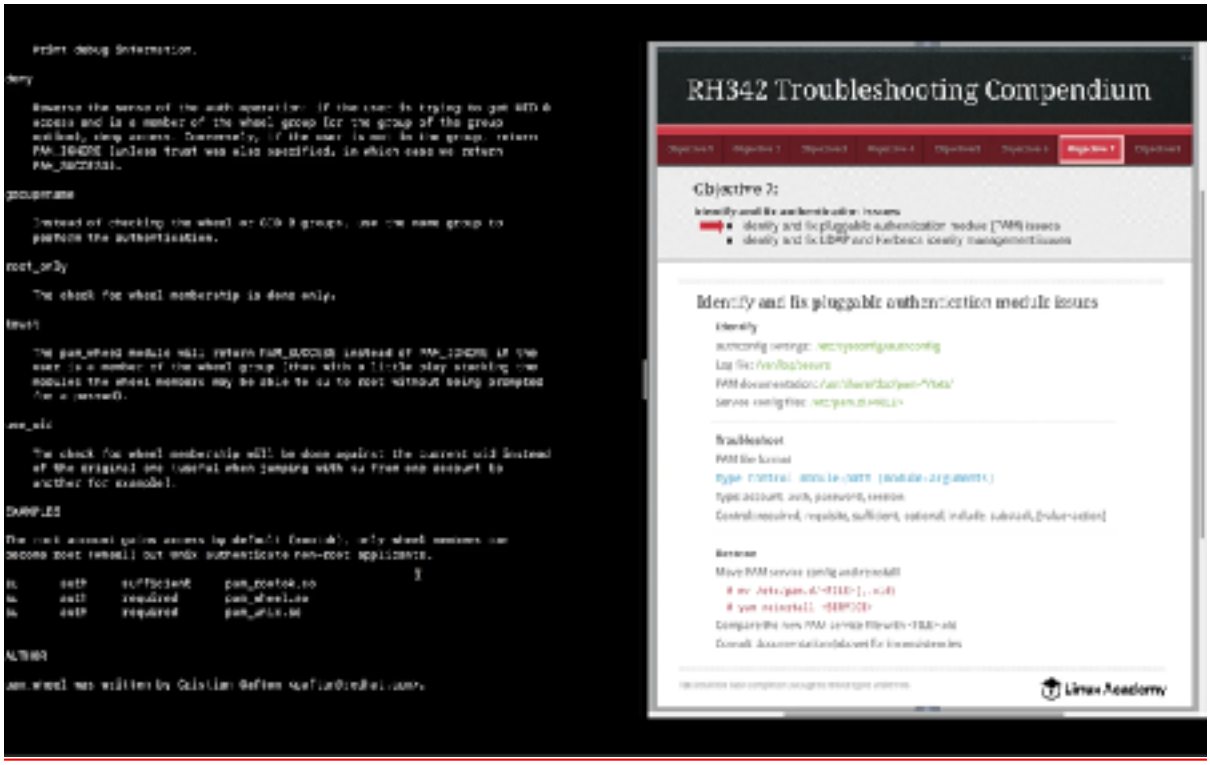

RH342 Troubleshooting Compendium
Objective 7:
Identify and fix pluggable authentication module issues
Troubleshoot
Linux Academy

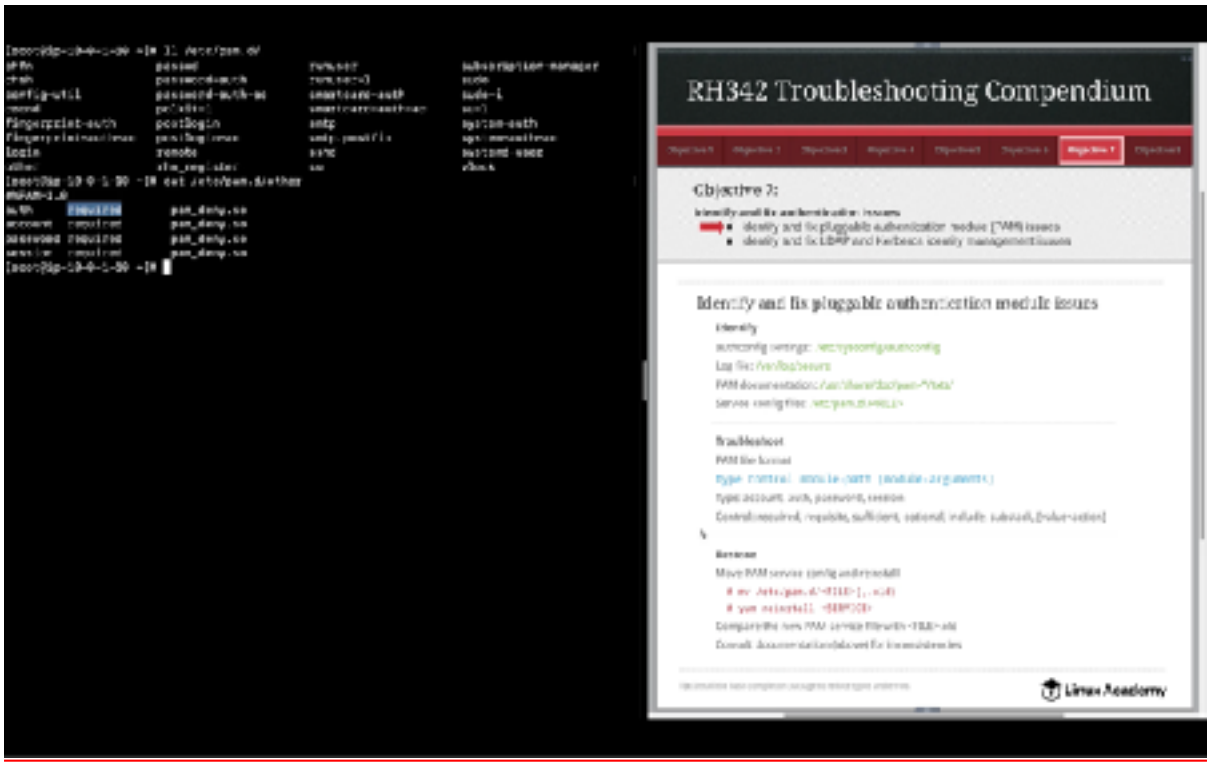

RH342 Troubleshooting Compendium
Objective 7:
Identify and fix pluggable authentication module issues
Troubleshoot
Linux Academy

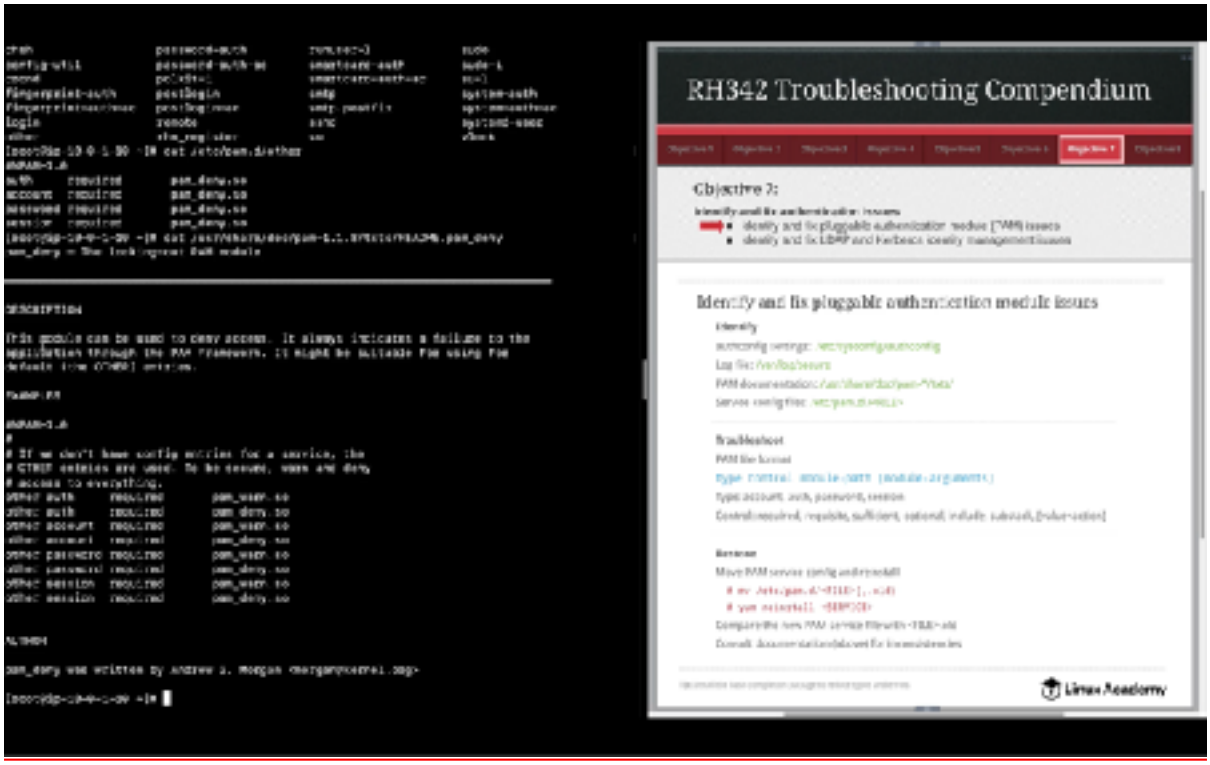

RH342 Troubleshooting Compendium
Objective 7:
Identify and fix pluggable authentication module issues
Troubleshoot
Linux Academy

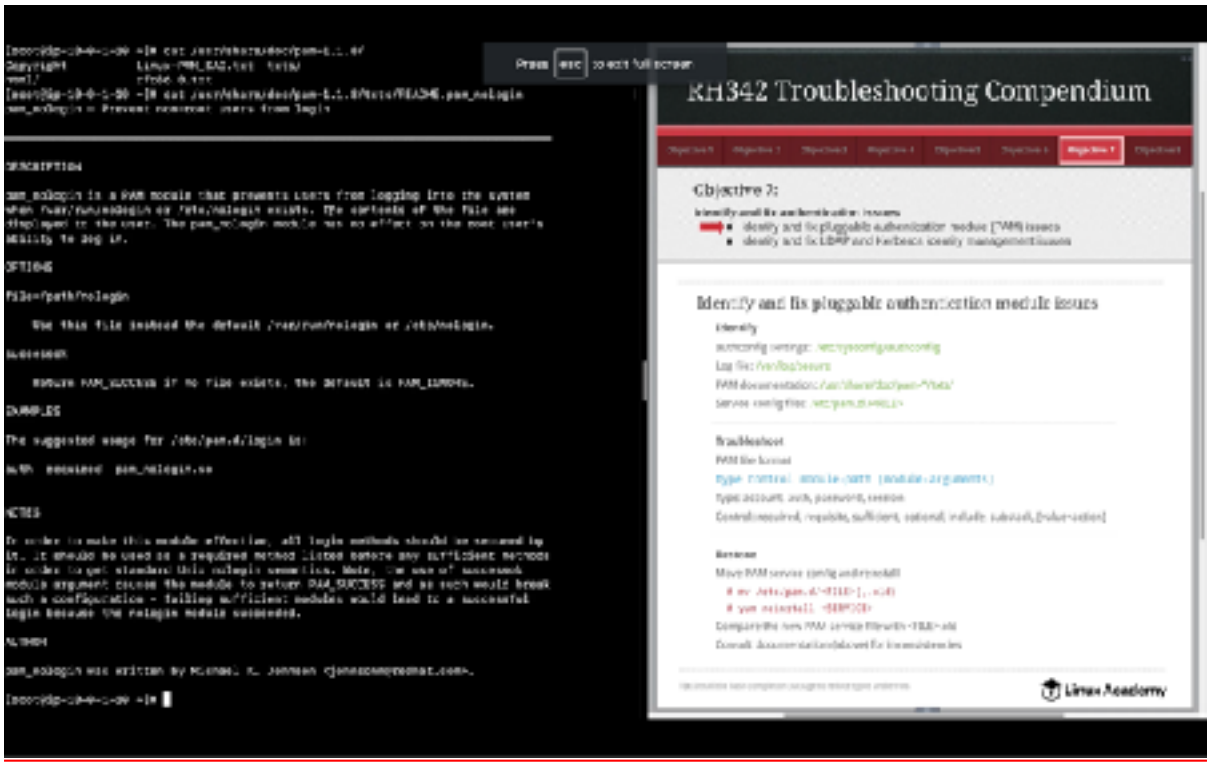

RH342 Troubleshooting Compendium
Objective 7:
Identify and fix pluggable authentication module issues
Linux Academy


RH342 Troubleshooting Compendium
Objective 7:
Identify and fix pluggable authentication module issues
Linux Academy


RH342 Troubleshooting Compendium
Objective 7:
Identify and fix pluggable authentication module issues
Troubleshoot
Linux Academy


RH342 Troubleshooting Compendium
Objective 7:
Identify and fix pluggable authentication module issues
Troubleshoot
Linux Academy





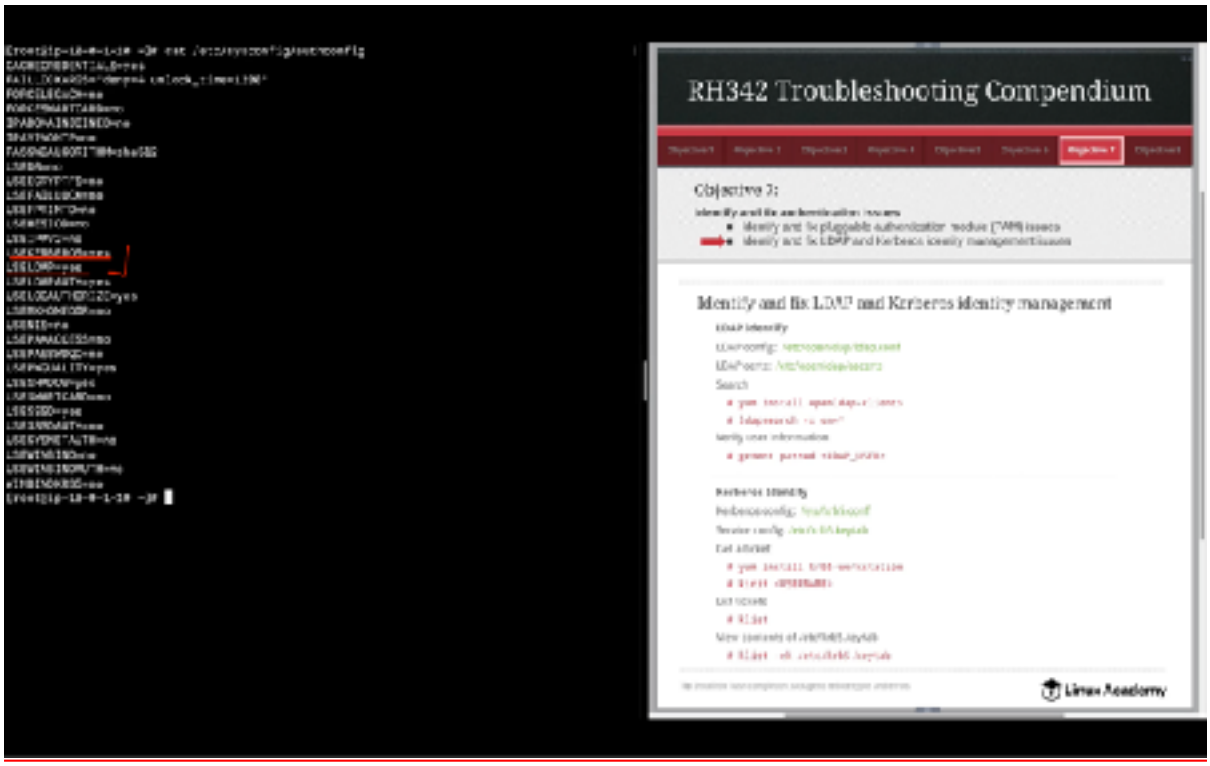
Ldap and Kerberos Troubleshooting

RH342: Objective 7

Identify and fix authentication issues

Identify and fix pluggable authentication module (PAM) issues

→ Identify and fix LDAP and Kerberos identity management issues



Objective 7:

Identify and fix authentication issues

- Identify and fix pluggable authentication module (PAM) issues
- Identify and fix LDAP and Kerberos identity management issues

Identify and fix LDAP and Kerberos identity management

LDAP Identify

LDAP config: /etc/openldap/ldap.conf

LDAP certs: /etc/openldap/cacerts

Search

```
# yum install openldap-clients
# ldapsearch -x cn=*
```

Verify user information

```
# getent passwd <LDAP_USER>
```

Kerberos Identify

Kerberos config: /etc/krb5.conf

Service config: /etc/krb5.keytab

Get a ticket

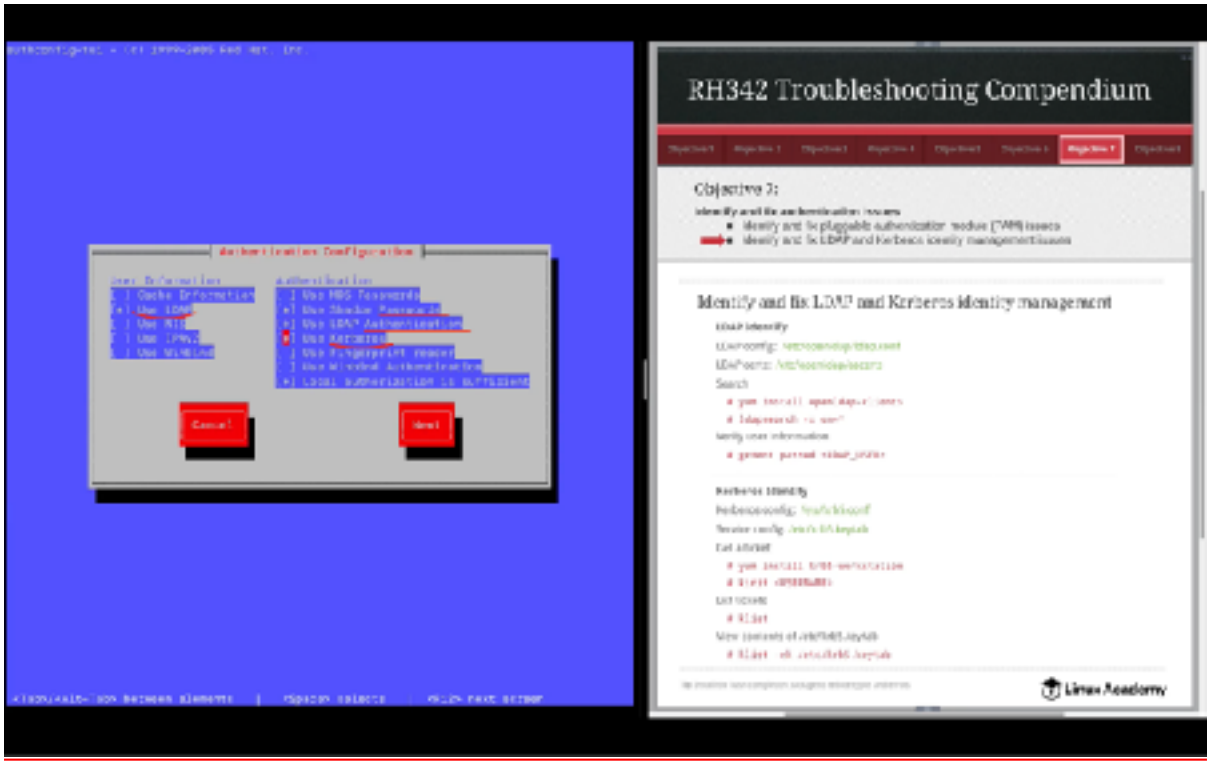
```
# yum install krb5-workstation
# kinit <USERNAME>
```

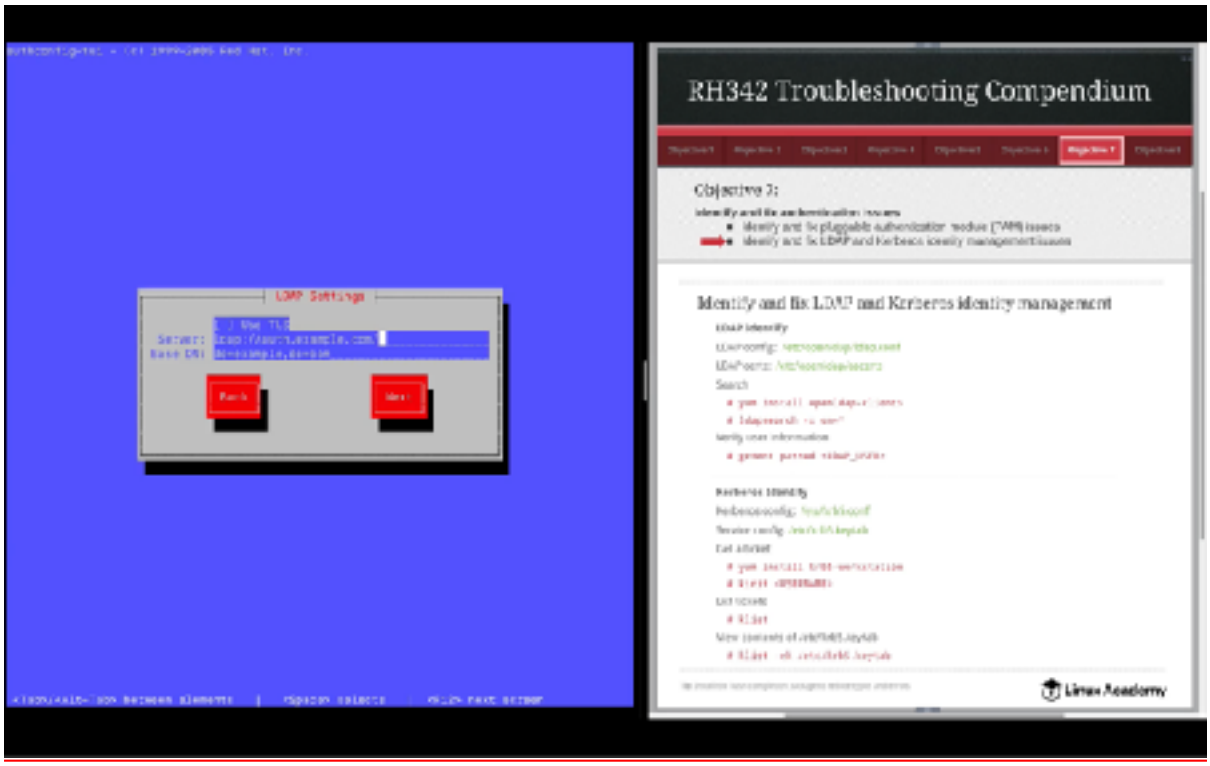
List tickets

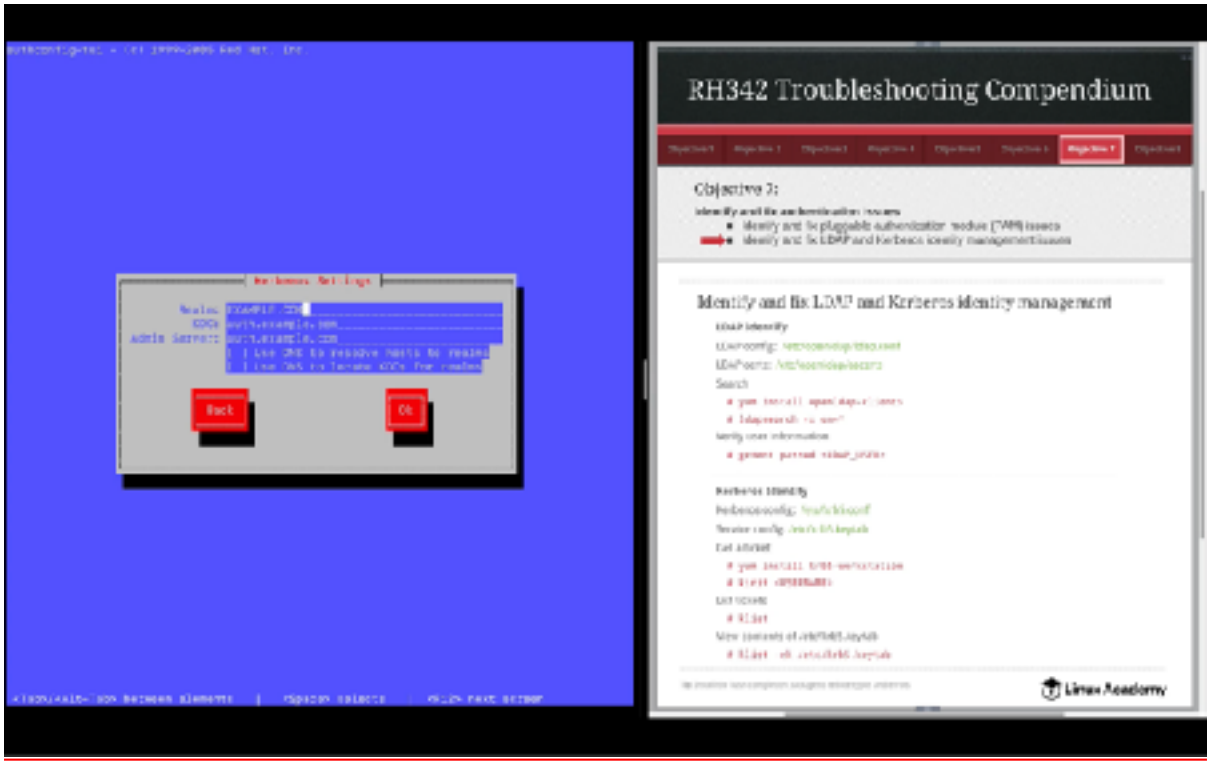
```
# klist
```

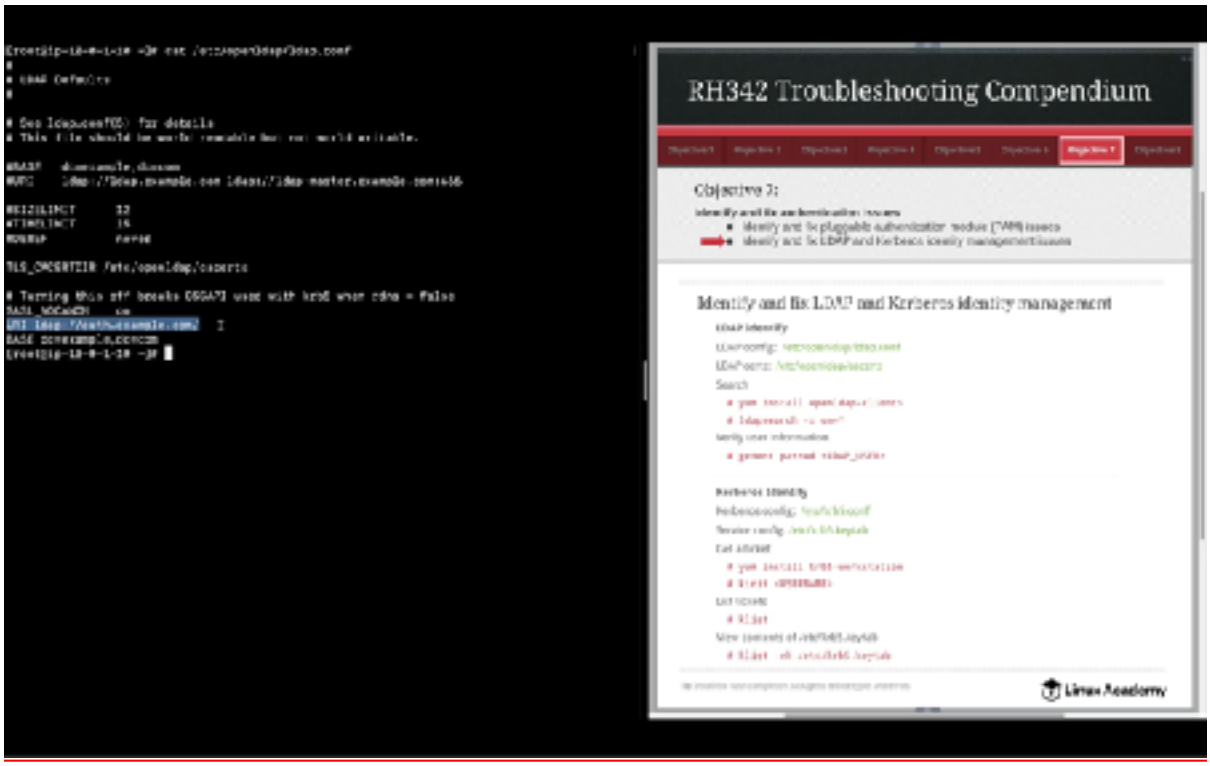
View contents of /etc/krb5.keytab

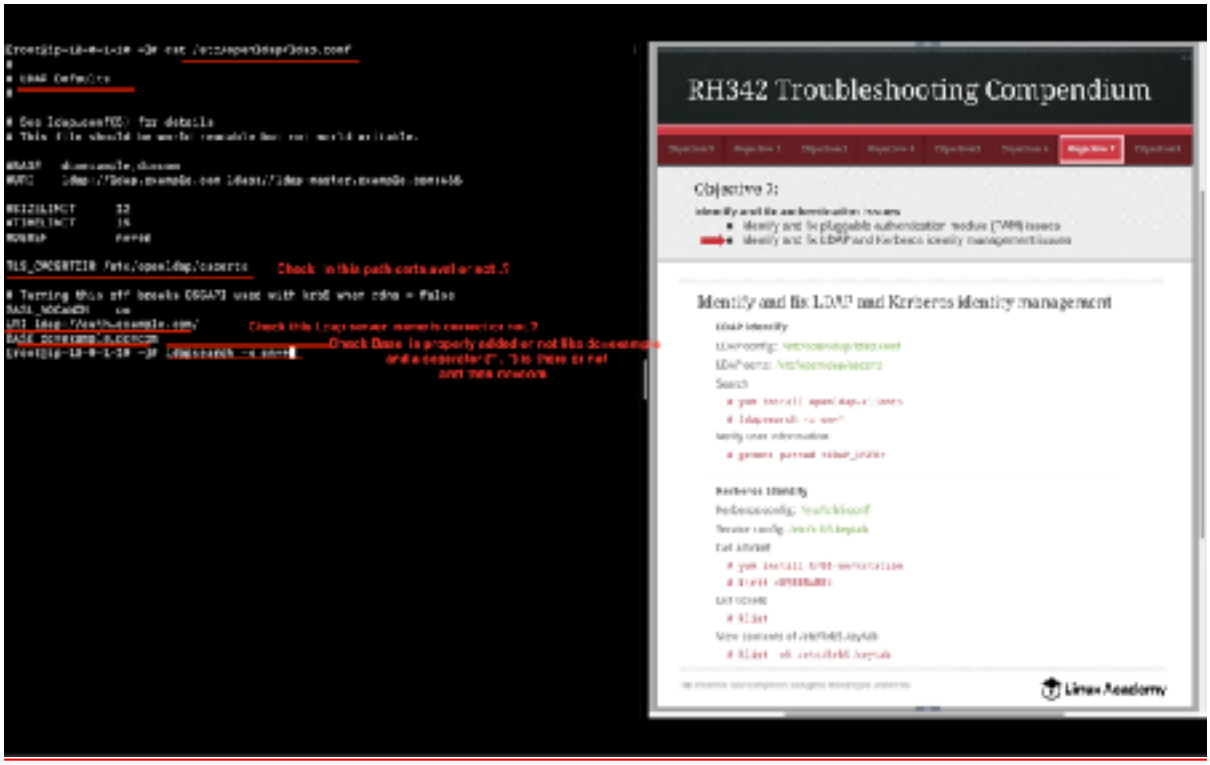
```
# klist -ek /etc/krb5.keytab
```

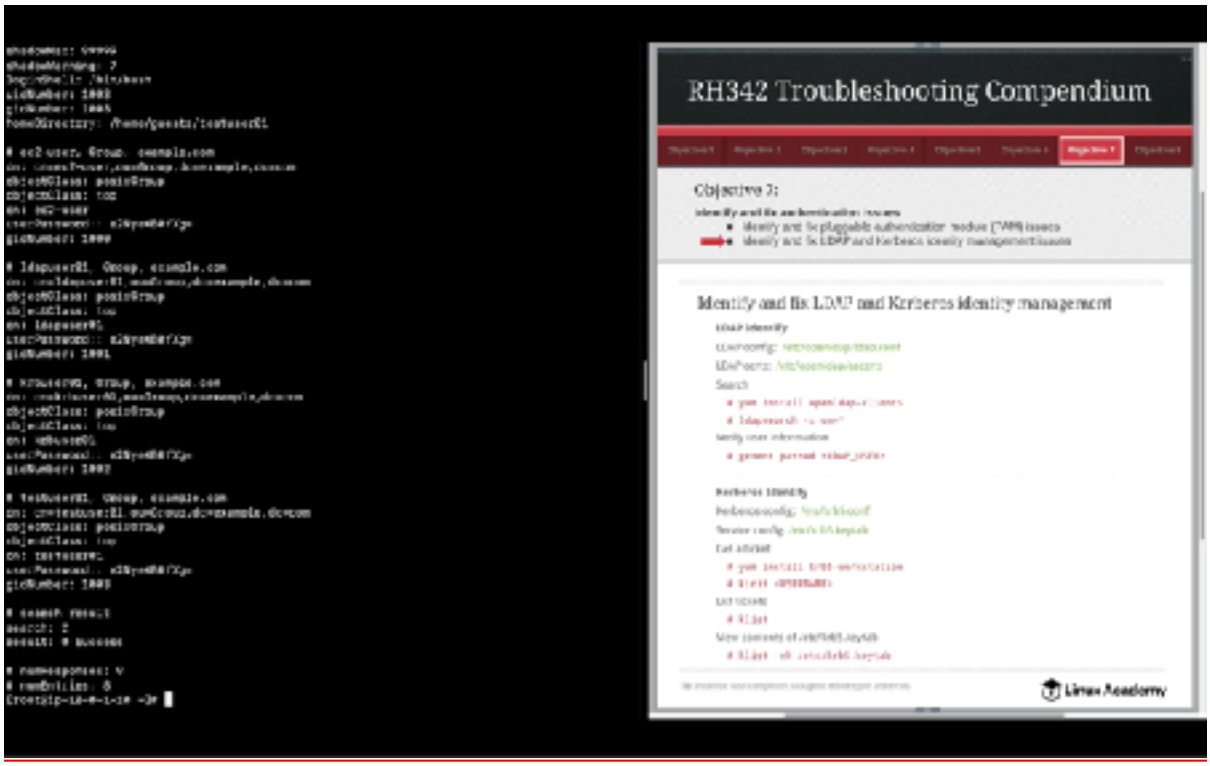

RH342 Troubleshooting Compendium
Objective 3:
Identify and fix LDAP and Kerberos identity management
Linux Academy

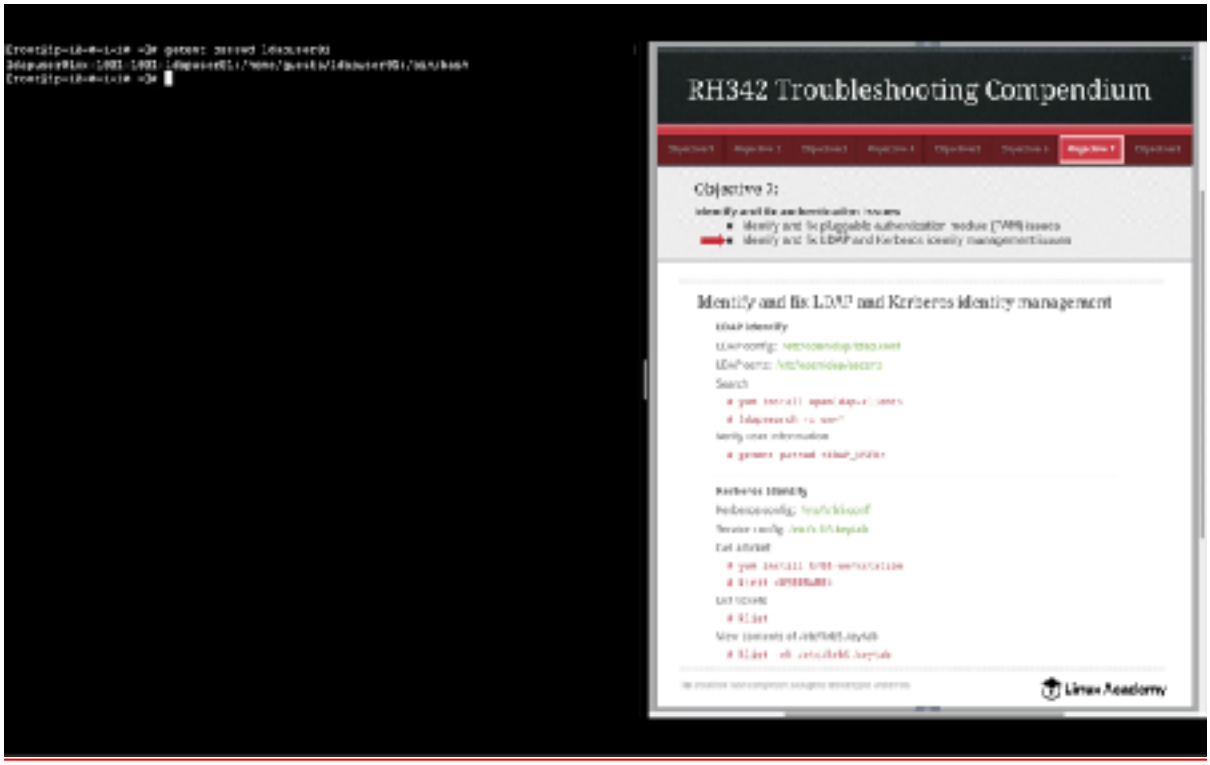

LDAP Settings
RH342 Troubleshooting Compendium
Objective 3:
Identify and fix LDAP and Kerberos identity management
Linux Academy

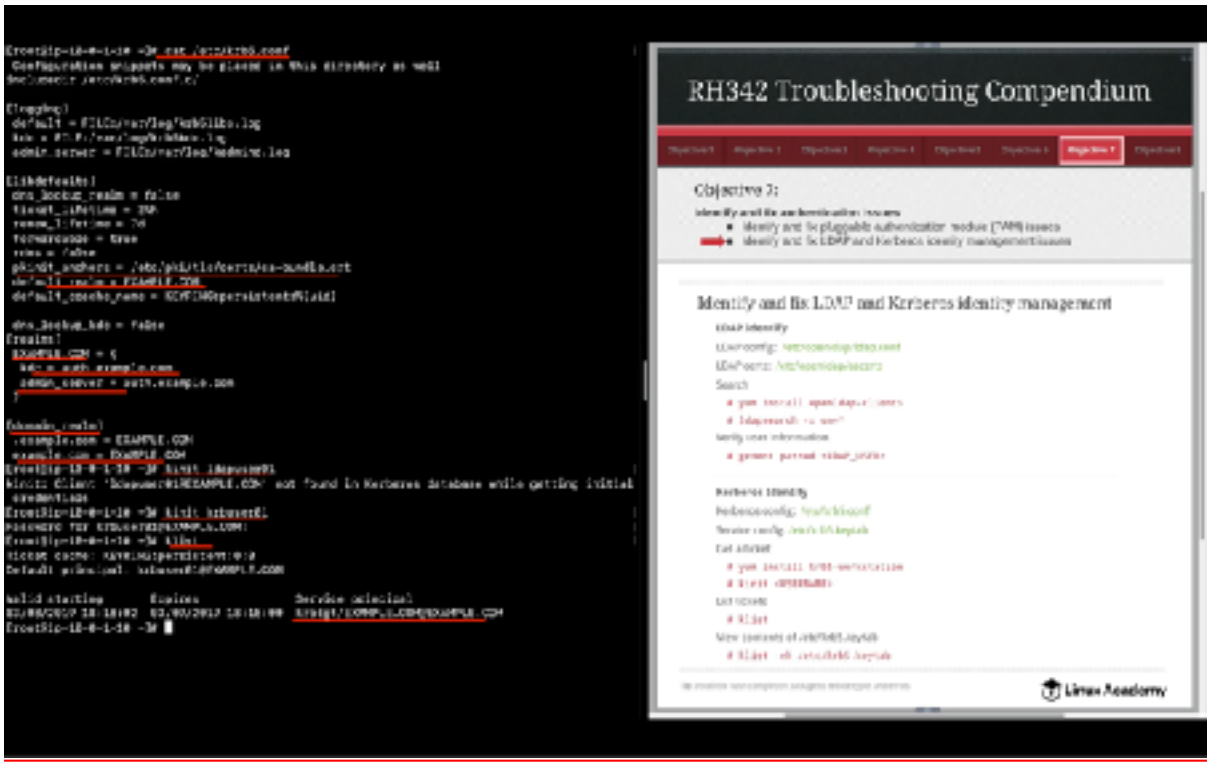

RH342 Troubleshooting Compendium
Objective 3:
Identify and fix LDAP and Kerberos identity management
Back
OK
Linux Academy

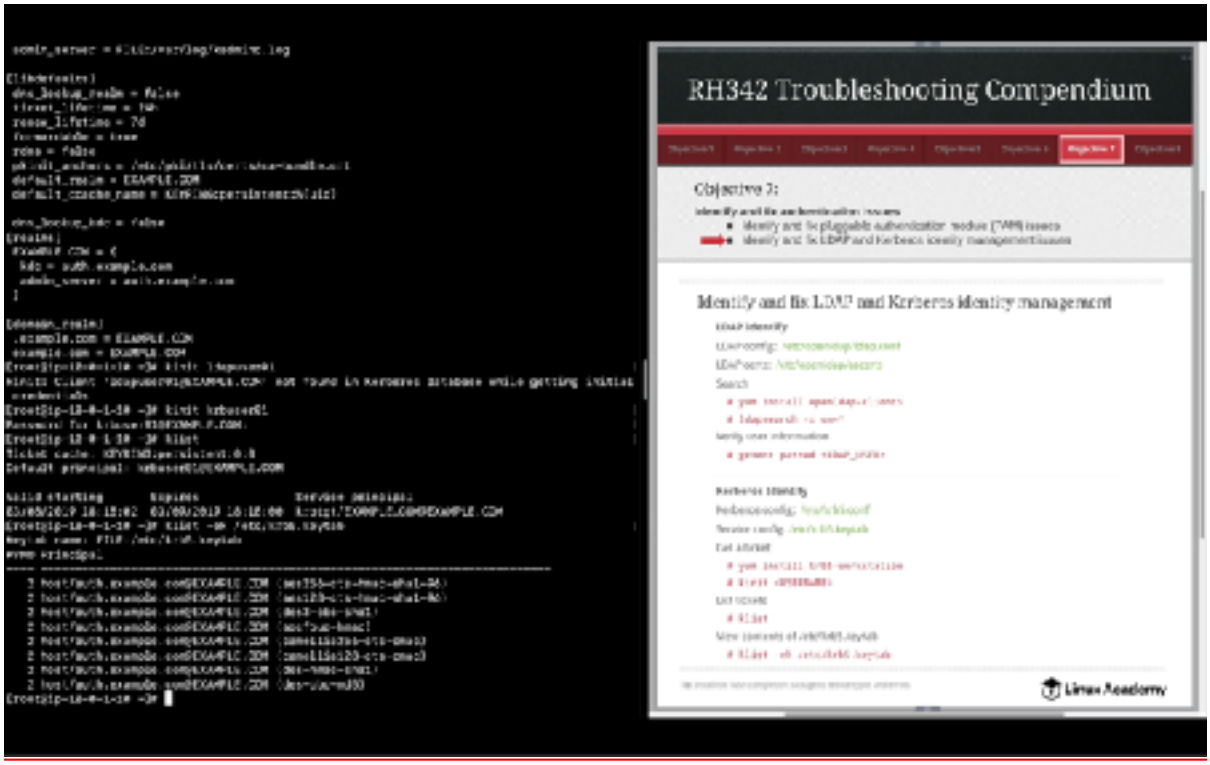

RH342 Troubleshooting Compendium
Objective 3:
Identify and fix LDAP and Kerberos identity management
Linux Academy

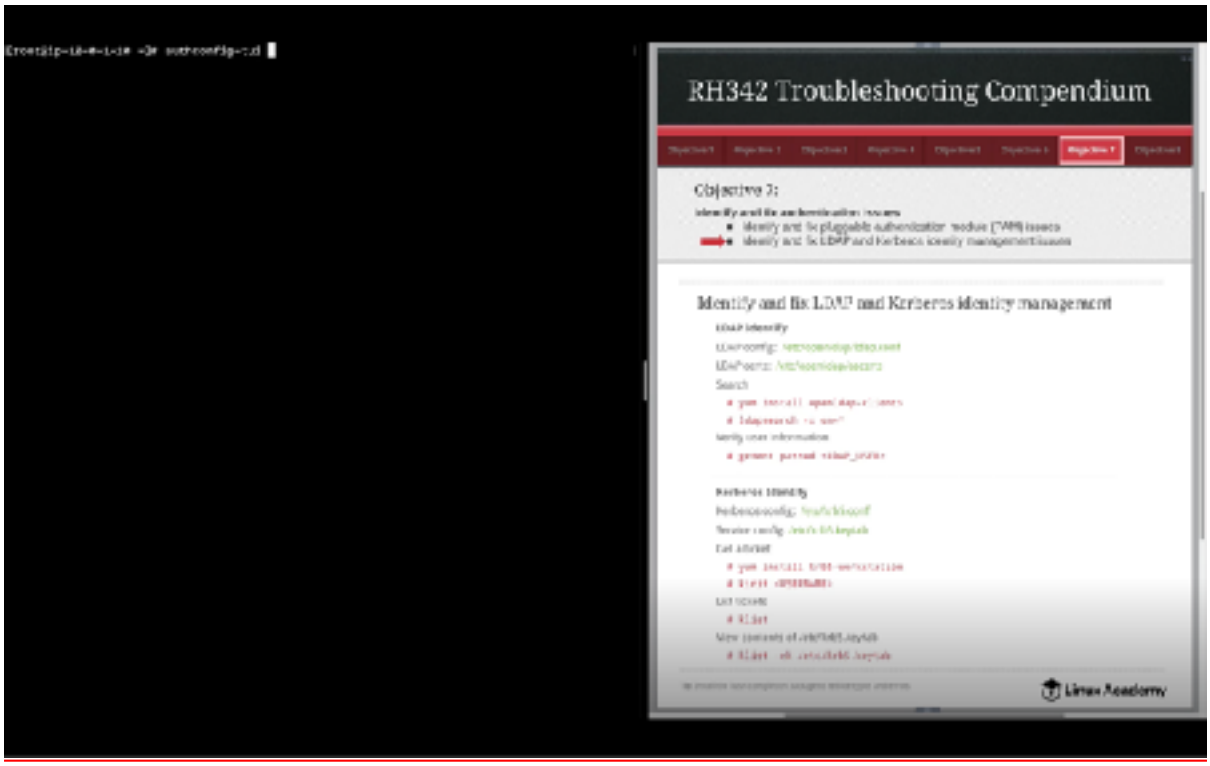

Check in this path certs and/or ext.?
Check this ldap server name is correctly
Check Base is properly added or not like dc=example
and a separator (",") is there or not
and has a value
RH342 Troubleshooting Compendium
Objective 3:
Identify and fix LDAP and Kerberos identity management
Linux Academy

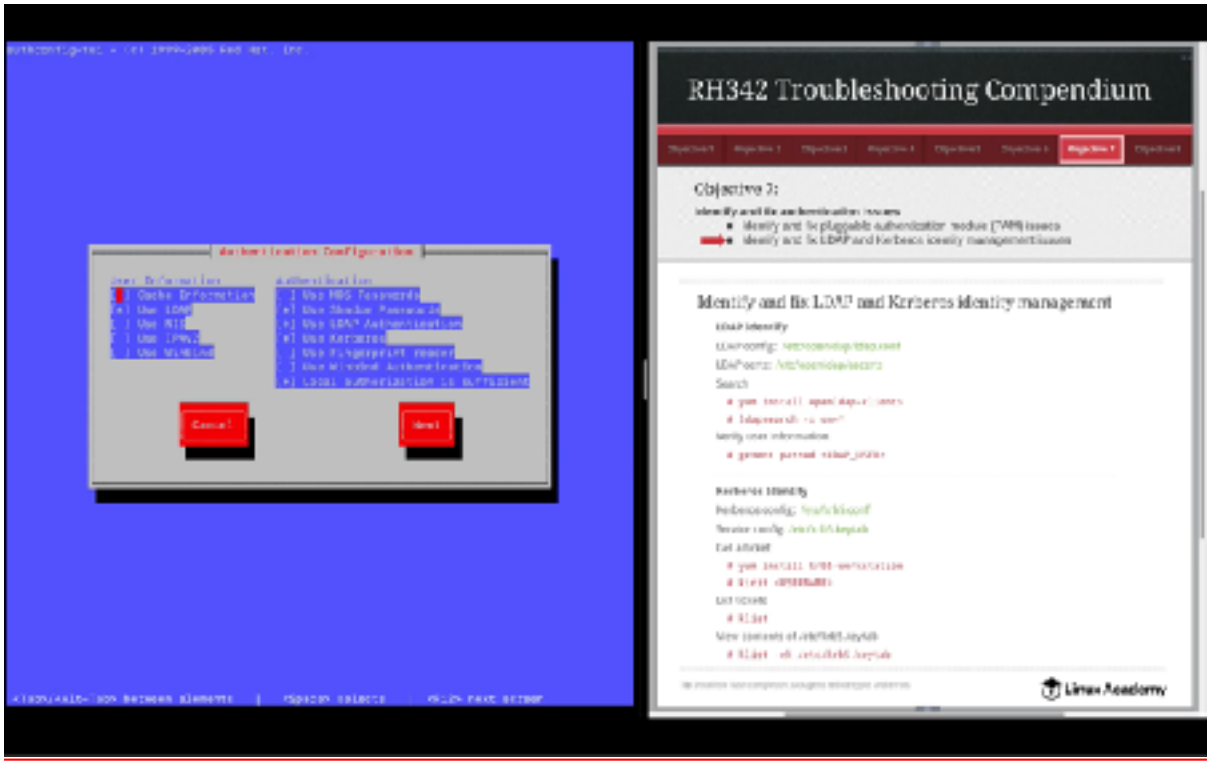

RH342 Troubleshooting Compendium
Objective 3:
Identify and fix LDAP and Kerberos identity management
Linux Academy

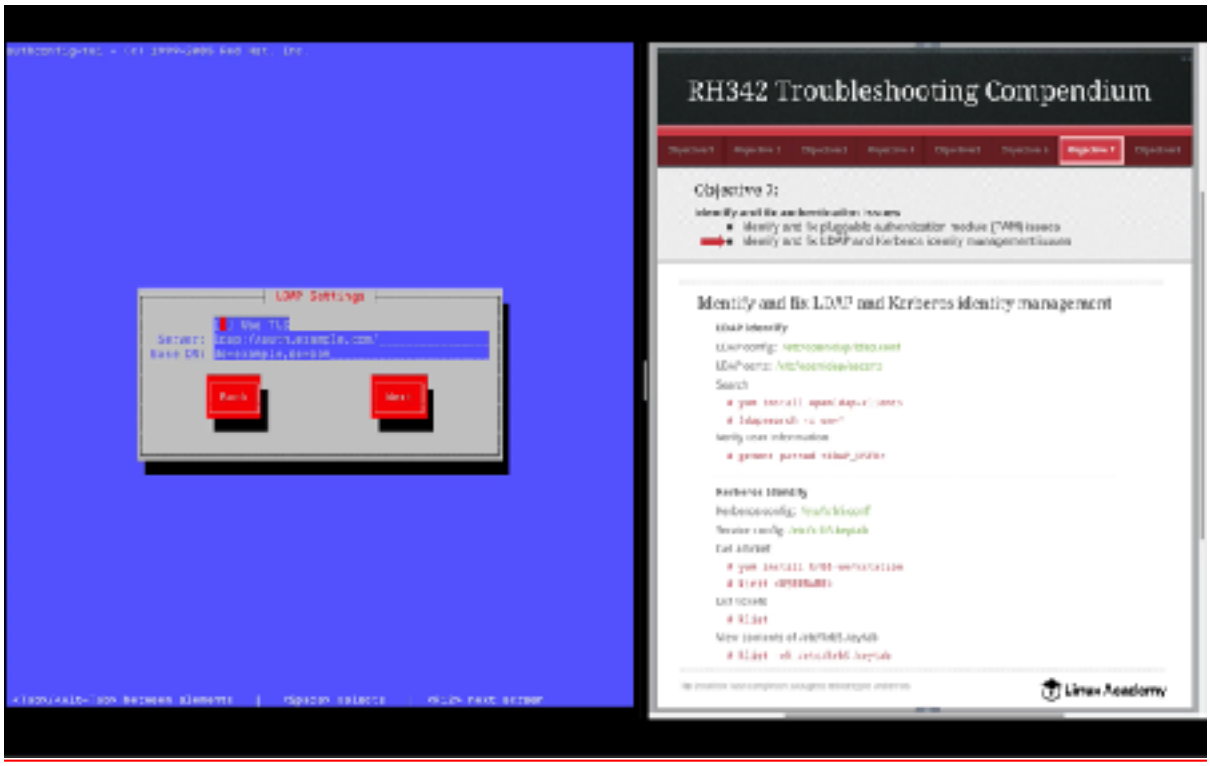

RH342 Troubleshooting Compendium
Objective 3:
Identify and fix LDAP and Kerberos identity management
Linux Academy

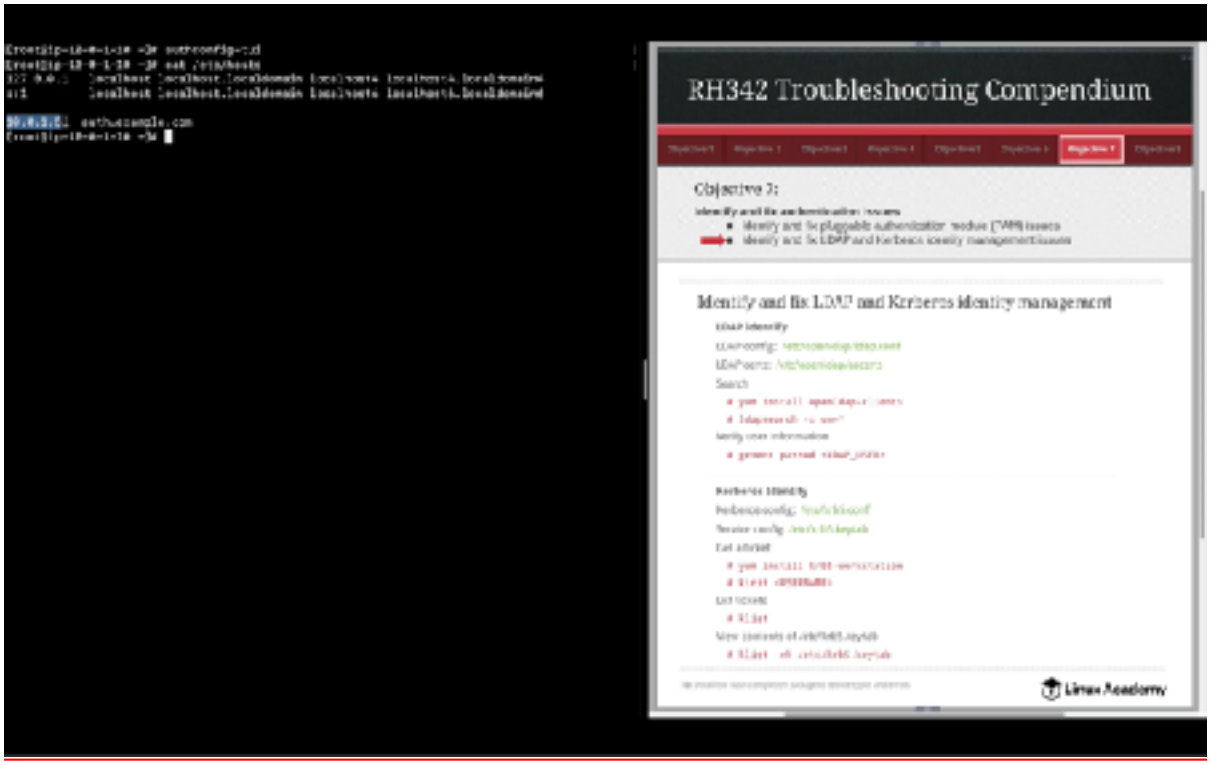

RH342 Troubleshooting Compendium
Objective 3:
Identify and fix LDAP and Kerberos identity management
Linux Academy

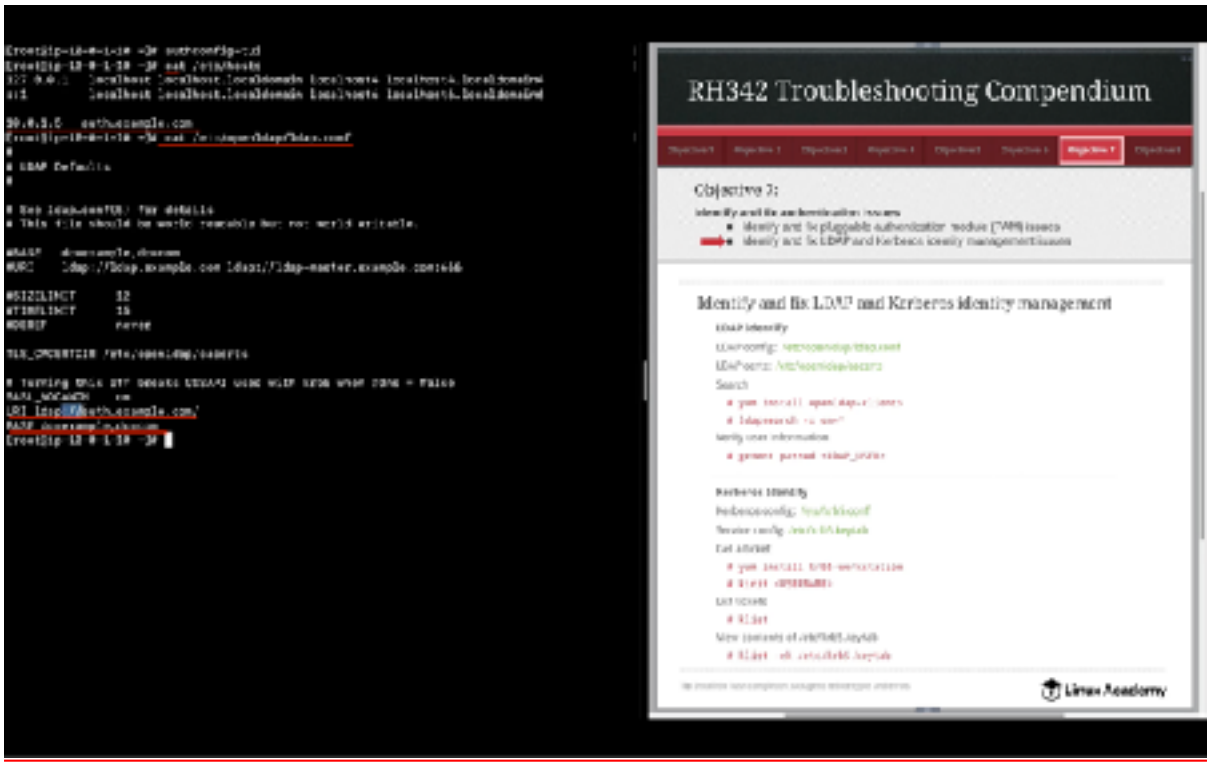

RH342 Troubleshooting Compendium
Objective 3:
Identify and fix LDAP and Kerberos identity management
Linux Academy

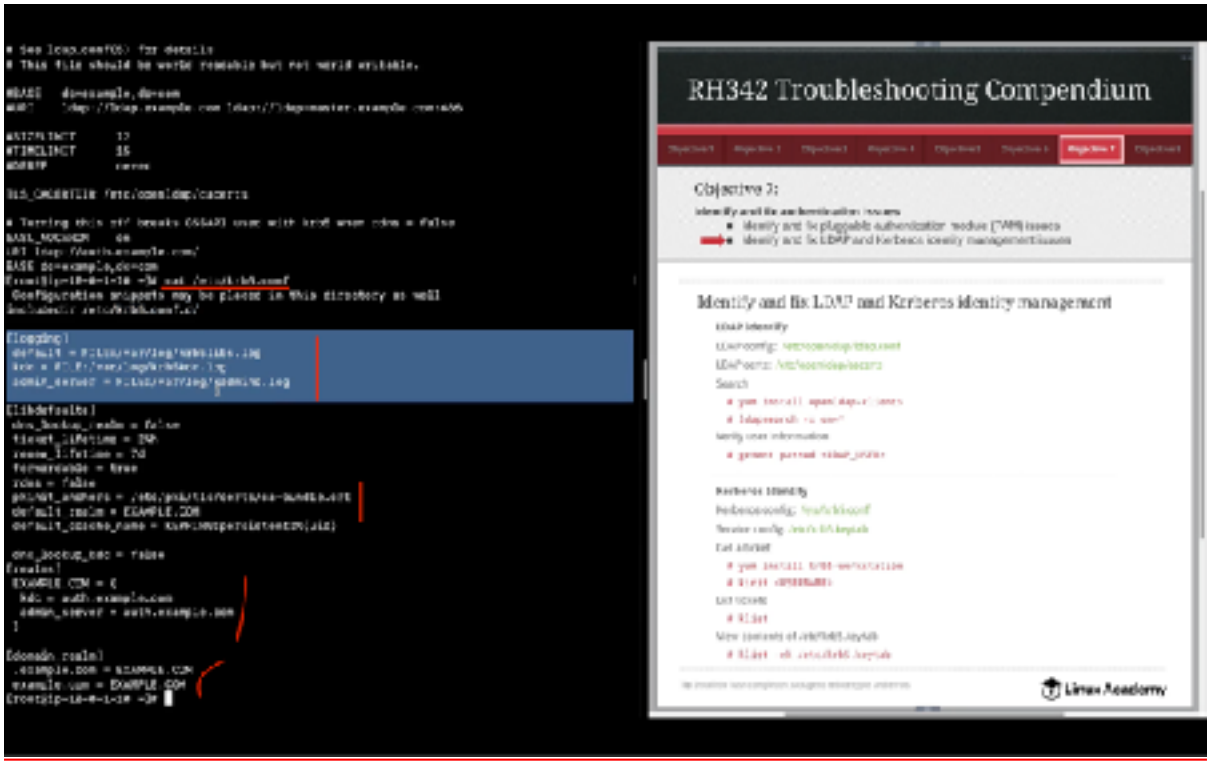

RH342 Troubleshooting Compendium
Objective 3:
Identify and fix LDAP and Kerberos identity management
Linux Academy


RH342 Troubleshooting Compendium
Objective 3:
Identify and fix LDAP and Kerberos identity management
Linux Academy

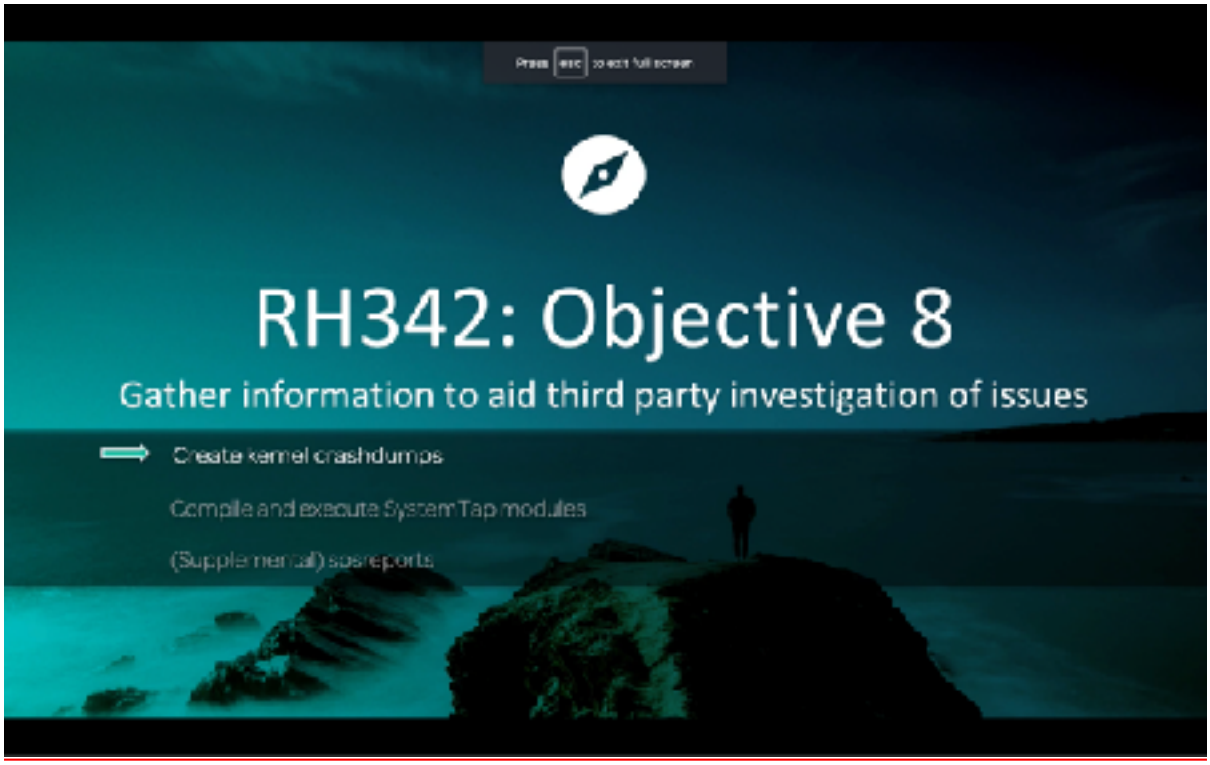

LDAP Settings
RH342 Troubleshooting Compendium
Objective 3:
Identify and fix LDAP and Kerberos identity management
Linux Academy


RH342 Troubleshooting Compendium
Objective 3:
Identify and fix LDAP and Kerberos identity management
Linux Academy


RH342 Troubleshooting Compendium
Objective 3:
Identify and fix LDAP and Kerberos identity management
Linux Academy



Creating Kernel crashDumps


Press esc to exit full screen
RH342: Objective 8
Gather information to aid third party investigation of issues
Create kernel crashdumps
Compile and execute SystemTap modules
(Supplemental) sosreports

Objective 8:

Gather information to aid third party investigation of issues

- Create kernel crashdumps
- Compile and execute SystemTap modules
- *Supplemental* sosreports

Create kernel crashdumps

kdump

```
# systemctl enable kdump && systemctl start kdump
```

Configuration file: /etc/kdump.conf

Remote crash dumps

kdumpctl to send crashdumps to another host

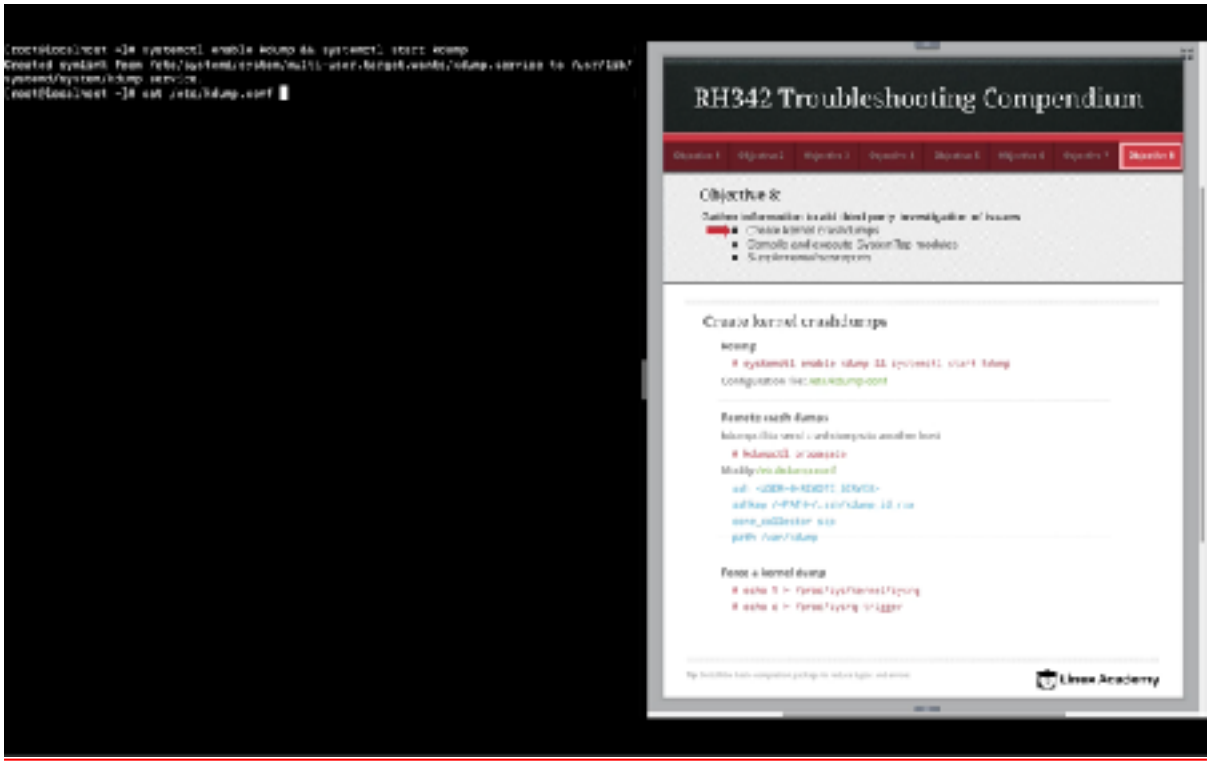
```
# kdumpctl propagate
```

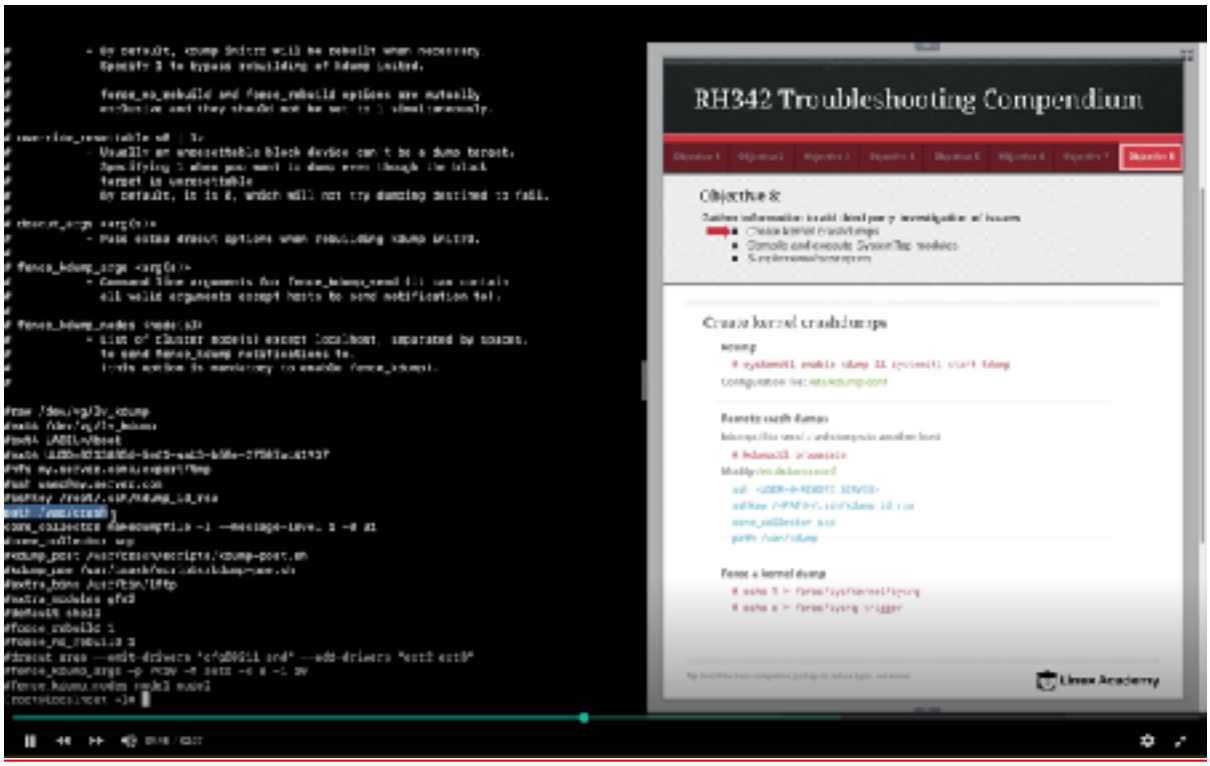
Modify /etc/kdump.conf

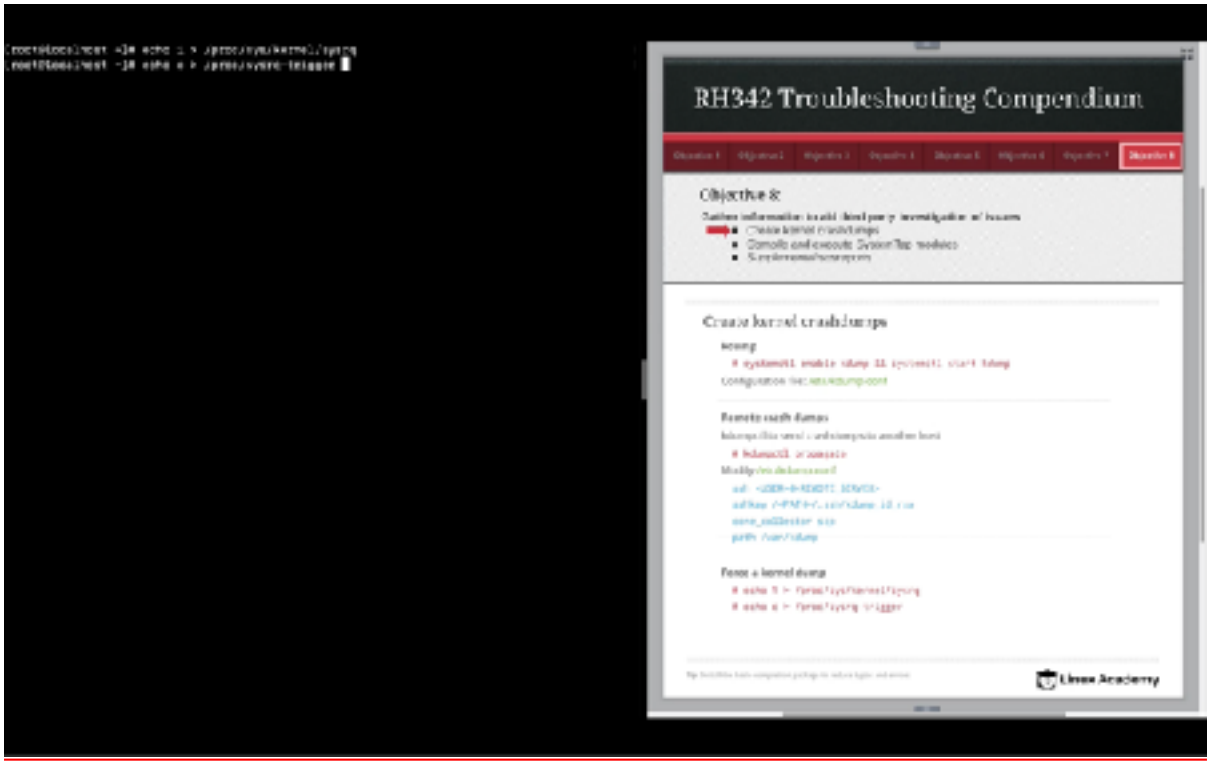
```
ssh <USER>@<REMOTE_SERVER>
sshkey /<PATH>/.ssh/kdump_id_rsa
core_collector scp
path /var/kdump
```

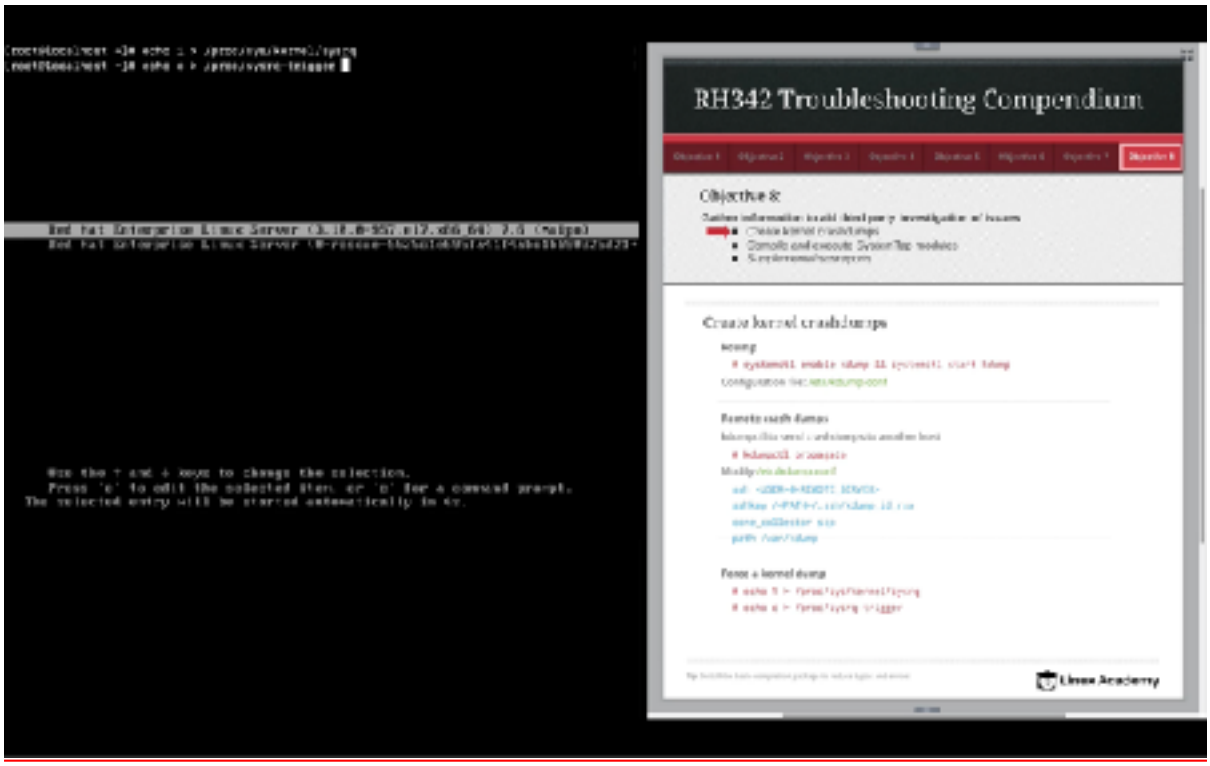
Force a kernel dump

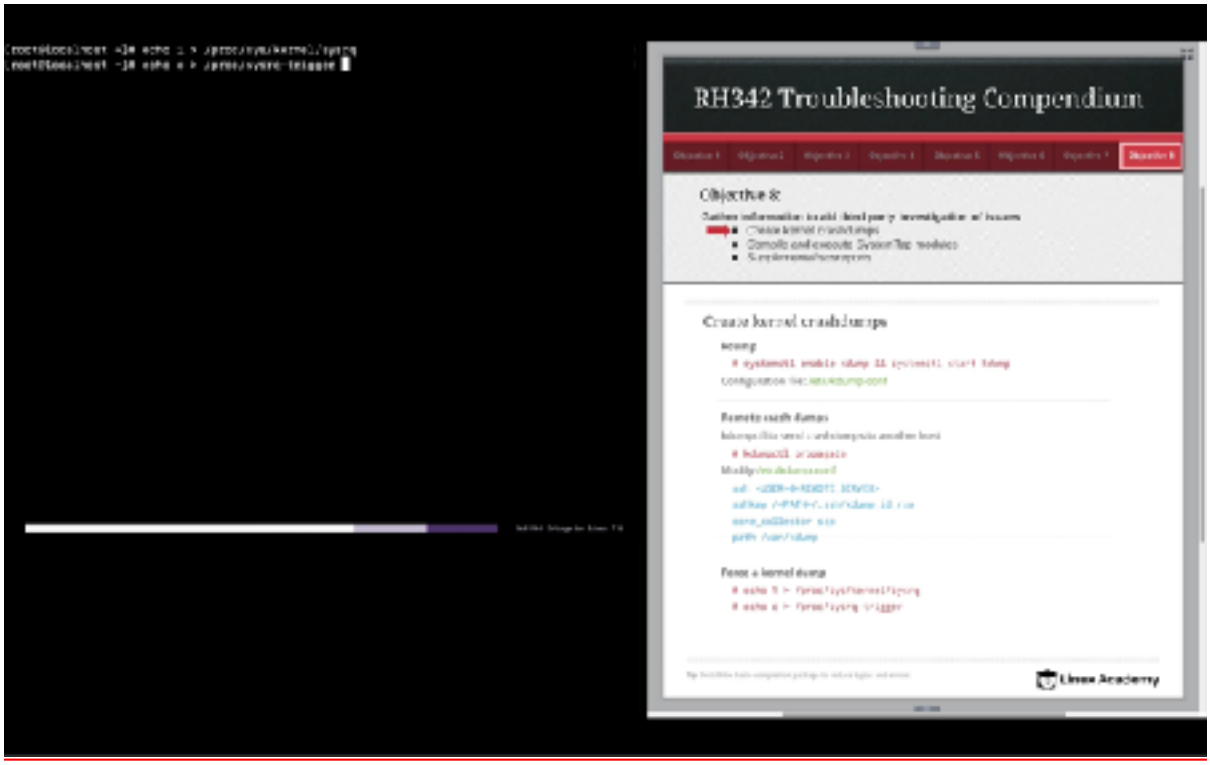
```
# echo 1 > /proc/sys/kernel/sysrq
# echo c > /proc/sysrq-trigger
```

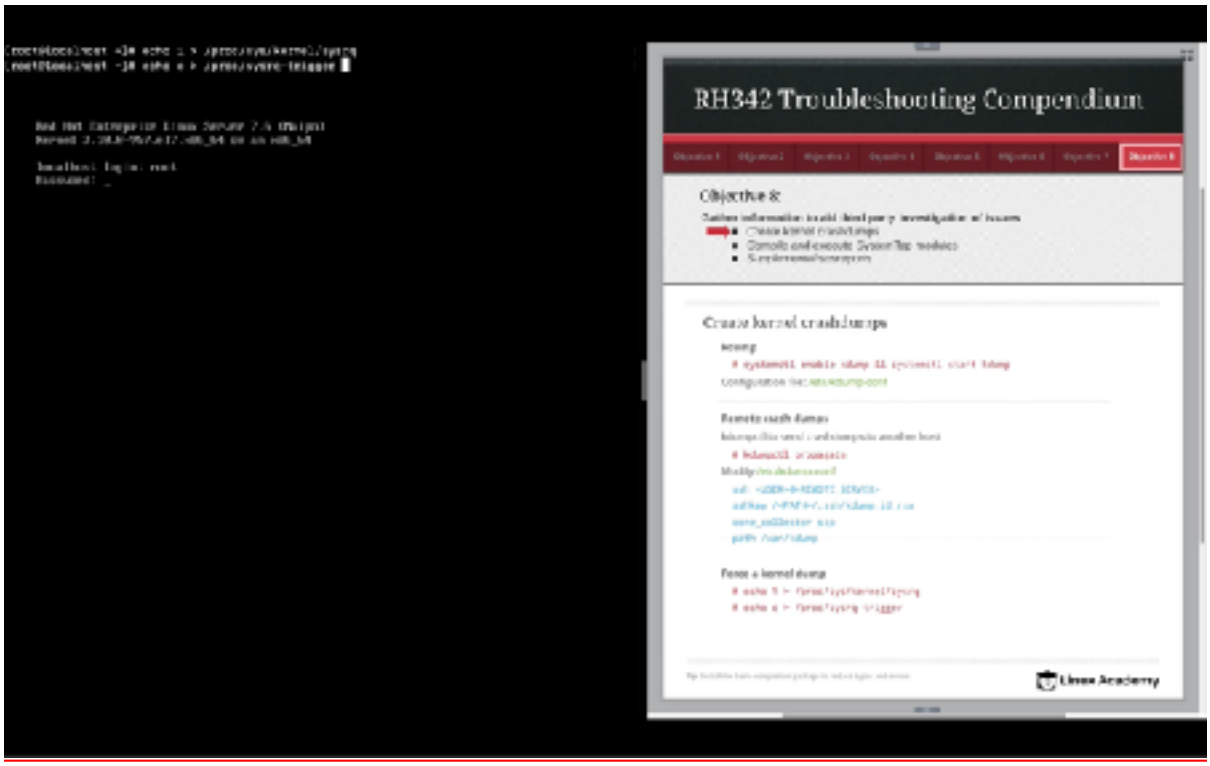

RH342 Troubleshooting Compendium
Objective 8:
Create kernel crashdumps
Linux Academy

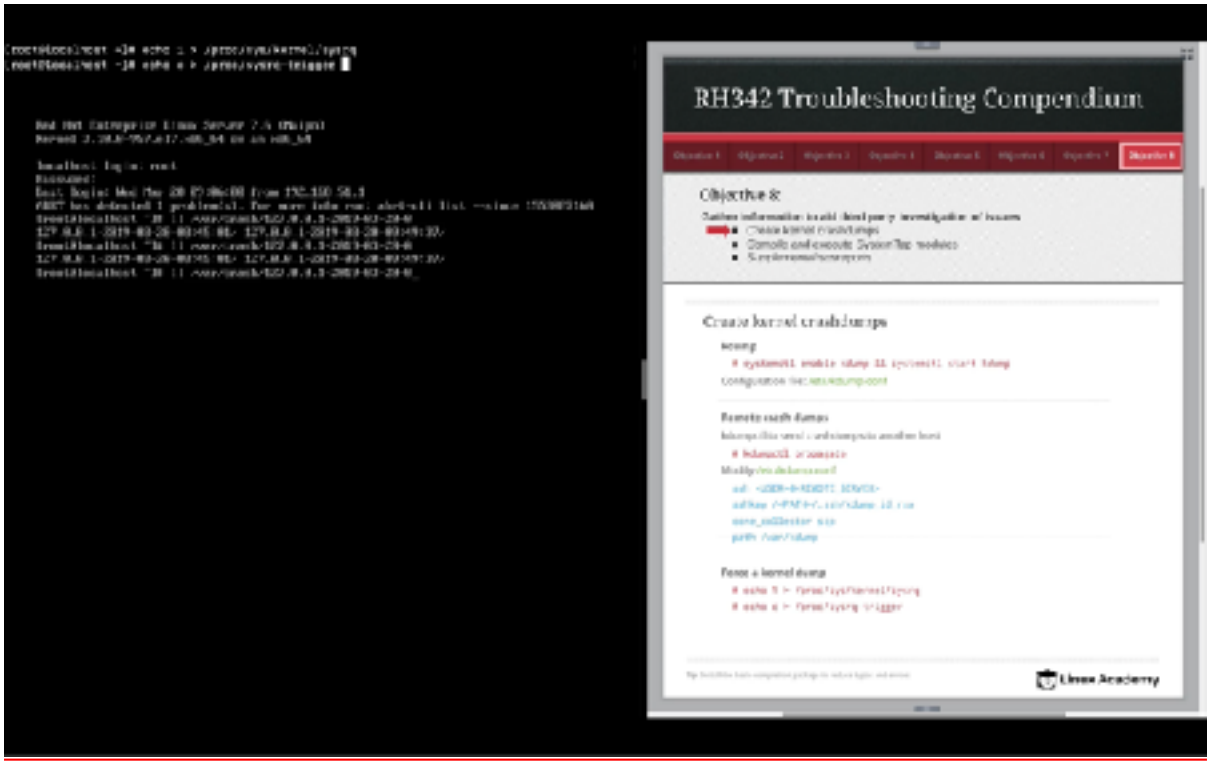

RH342 Troubleshooting Compendium
Objective 8:
Create kernel crashdumps
Linux Academy


RH342 Troubleshooting Compendium
Objective 8:
Create kernel crashdumps
Linux Academy


RH342 Troubleshooting Compendium
Objective 8:
Create kernel crashdumps
Linux Academy


RH342 Troubleshooting Compendium
Objective 8:
Create kernel crashdumps
Linux Academy


RH342 Troubleshooting Compendium
Objective 8:
Create kernel crashdumps
Linux Academy



Compile and execute SystamTap modules

Objective 8:

Gather information to aid third party investigation of issues

- Create kernel crashdumps
- Compile and execute SystemTap modules
- *Supplemental* sosreports

Compile and execute SystemTap modules

Documentation

Example scripts: /usr/share/systemtap/tapset/

Prerequisites

kernel-debuginfo (this package requires 'rhel-7-server-debug-rpms')

kernel-devel

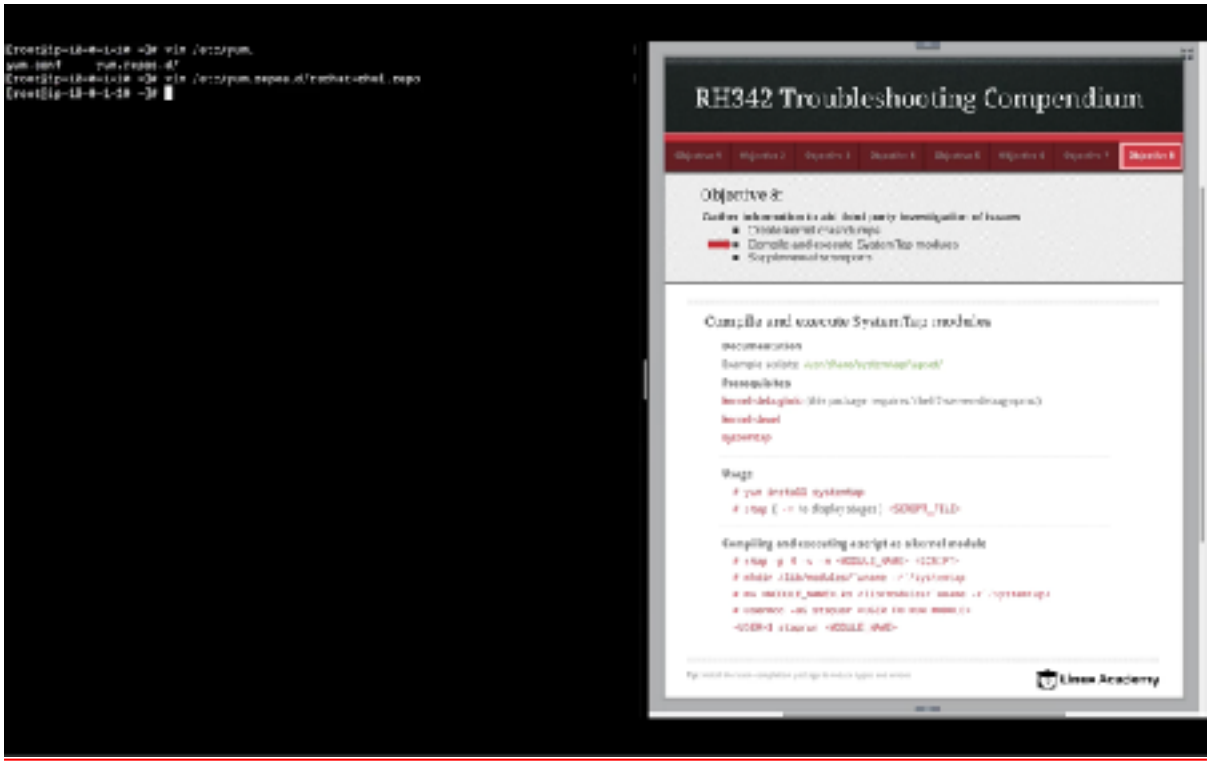
systemtap

Usage

```
# yum install systemtap
# stap ( -v to display stages ) <SCRIPT_FILE>
```

Compiling and executing a script as a kernel module

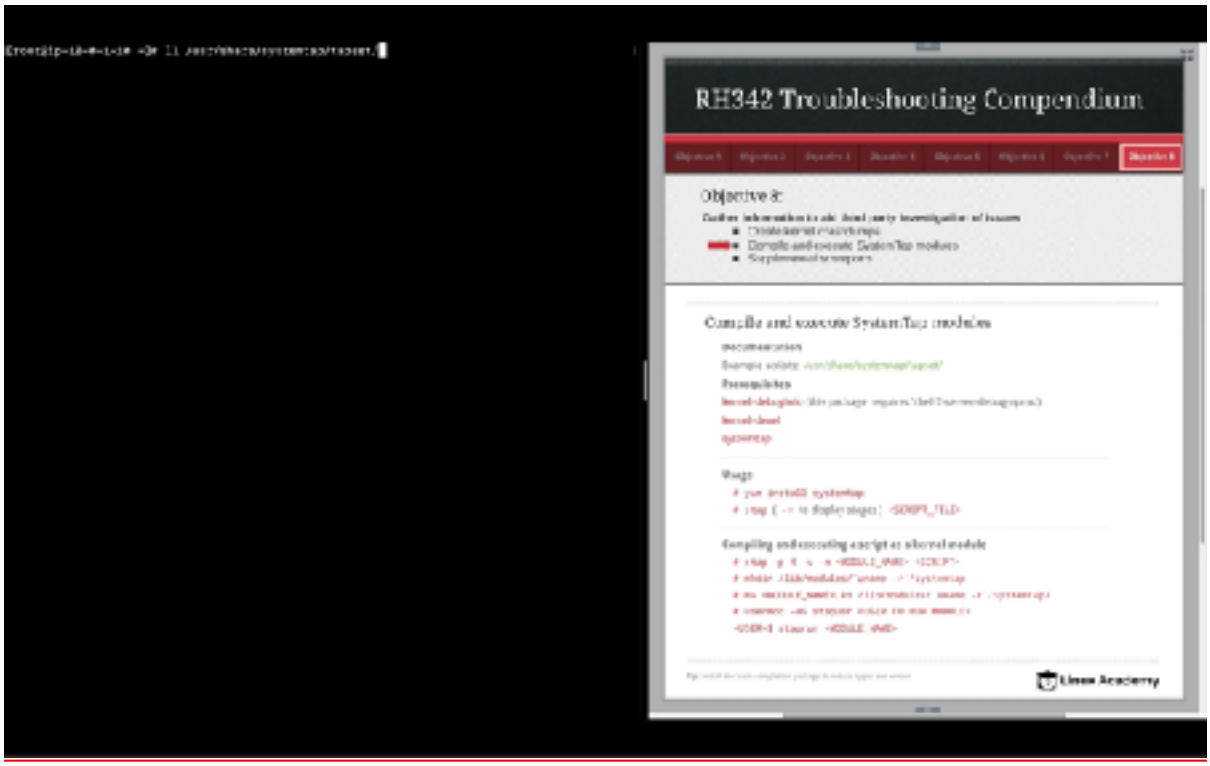
```
# stap -p 4 -v -m <MODULE_NAME> <SCRIPT>
# mkdir /lib/modules/`uname -r`/systemtap
# mv <MODULE_NAME>.ko /lib/modules/`uname -r`/systemtap/
# usermod -aG stapusr <USER TO RUN MODULE>
<USER>$ staprun <MODULE_NAME>
```


RH342 Troubleshooting Compendium
Objective 8:
Compile and execute SystemTap modules
Usage
Linux Academy

RH342 Troubleshooting Compendium
Objective 8:
Compile and execute SystemTap modules
Linux Academy

RH342 Troubleshooting Compendium
Objective 8:
Compile and execute SystemTap modules
Linux Academy

RH342 Troubleshooting Compendium
Objective 8:
Compile and execute SystemTap modules
Linux Academy


RH342 Troubleshooting Compendium
Objective 8:
Compile and execute SystemTap modules
Linux Academy

RH342 Troubleshooting Compendium

Objective 8:

Compile and execute SystemTap modules

Linux Academy

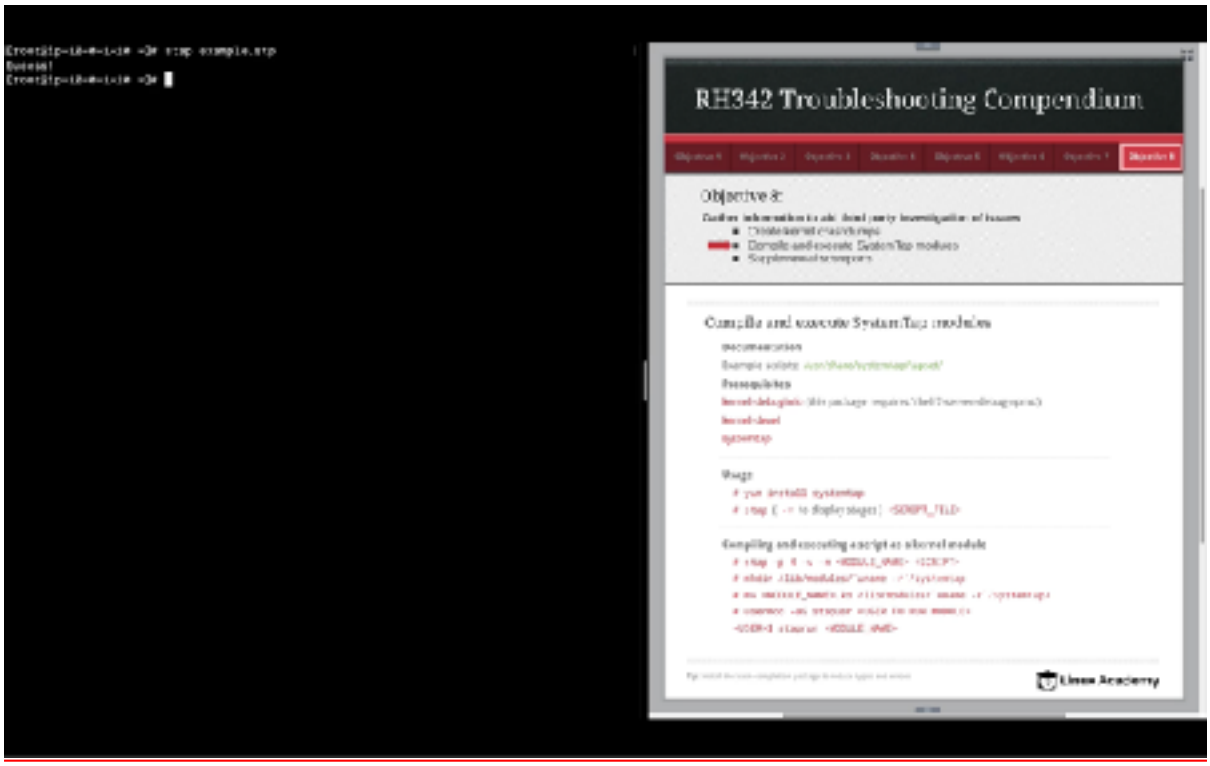
RH342 Troubleshooting Compendium

Objective 8:

Compile and execute SystemTap modules

Linux Academy

RH342 Troubleshooting Compendium
Objective 8:
Compile and execute SystemTap modules
Linux Academy

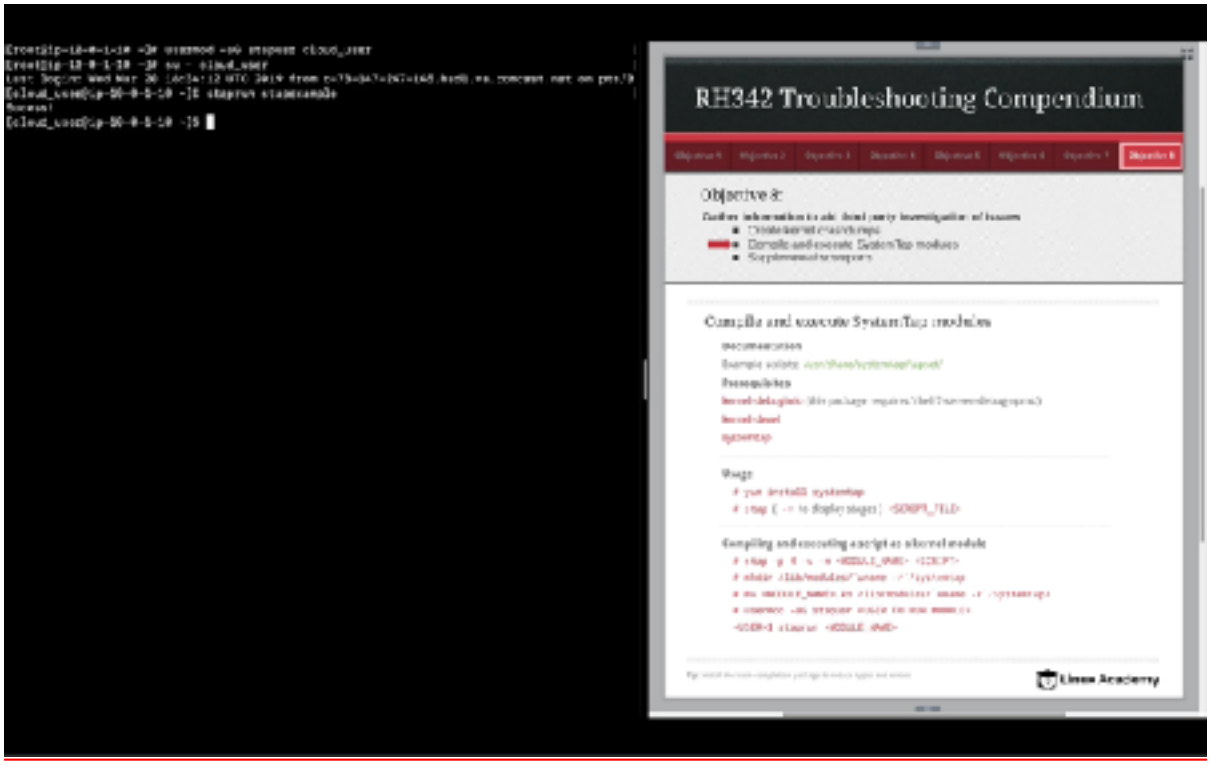

RH342 Troubleshooting Compendium
Objective 8:
Compile and execute SystemTap modules
Linux Academy

RH342 Troubleshooting Compendium
Objective 8:
Compile and execute SystemTap modules
Usage
Compiling and executing a script as a kernel module
Linux Academy

RH342 Troubleshooting Compendium
Objective 8:
Compile and execute SystemTap modules
Usage
Compiling and executing a script as a kernel module
Linux Academy

RH342 Troubleshooting Compendium
Objective 8:
Compile and execute SystemTap modules
Linux Academy

RH342 Troubleshooting Compendium
Objective 8:
Compile and execute SystemTap modules
Linux Academy


RH342 Troubleshooting Compendium
Objective 8:
Compile and execute SystemTap modules
Linux Academy

SOSReports

Objective 8:

Gather information to aid third party investigation of issues

- Create kernel crashdumps
- Compile and execute SystemTap modules
- *Supplemental* sosreports

Sosreports

Usage

List modules

```
# sosreport -l
```

Run wihout a module

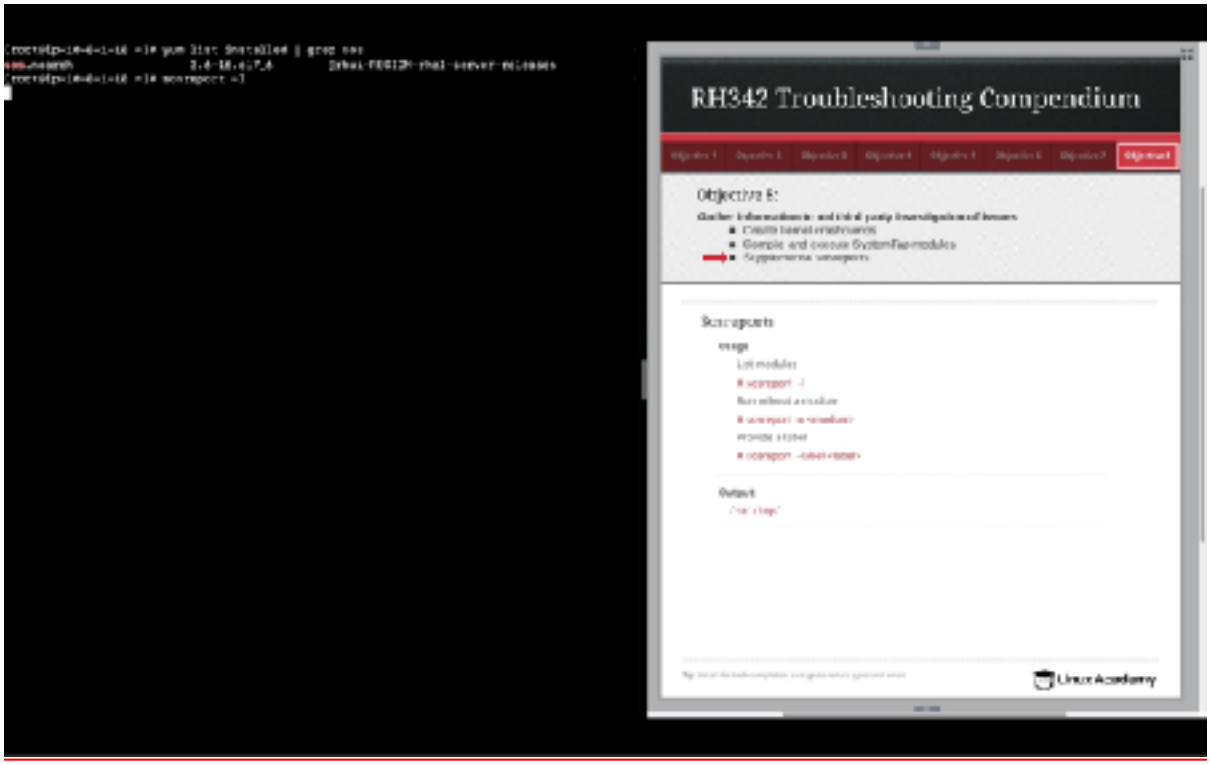
```
# sosreport -n <module>
```

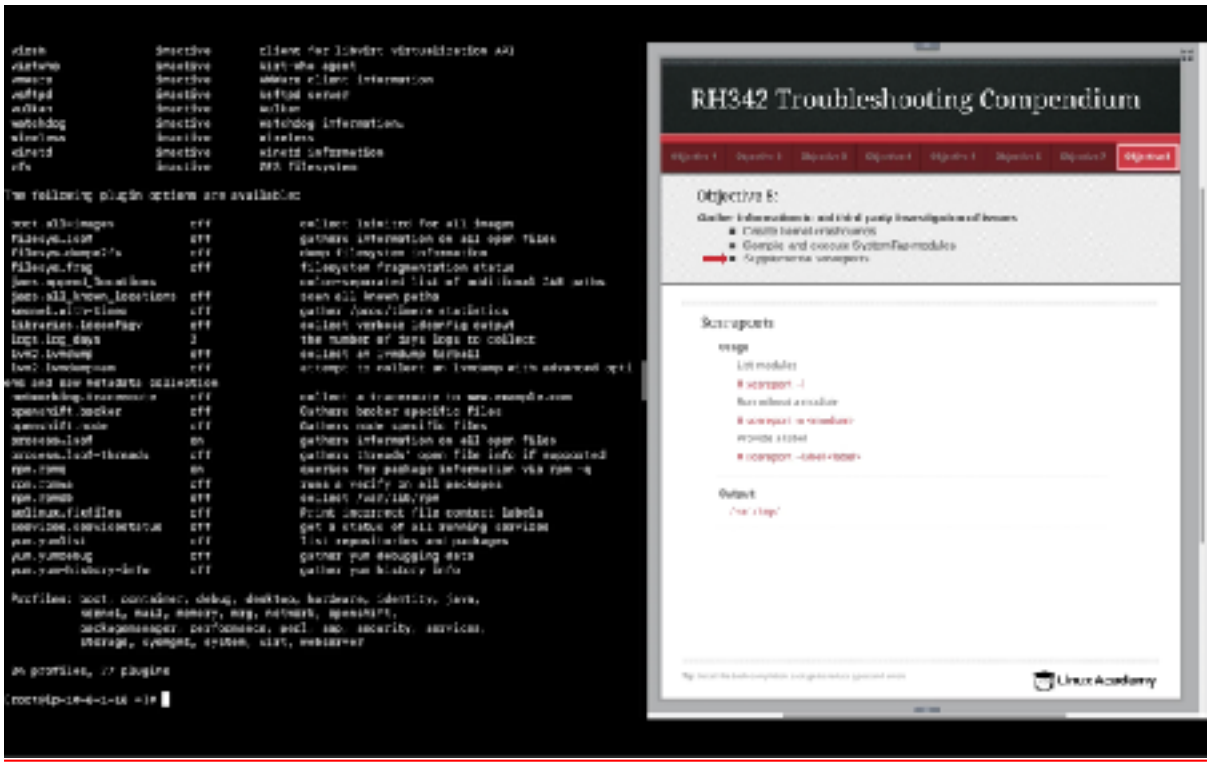
Provide a label

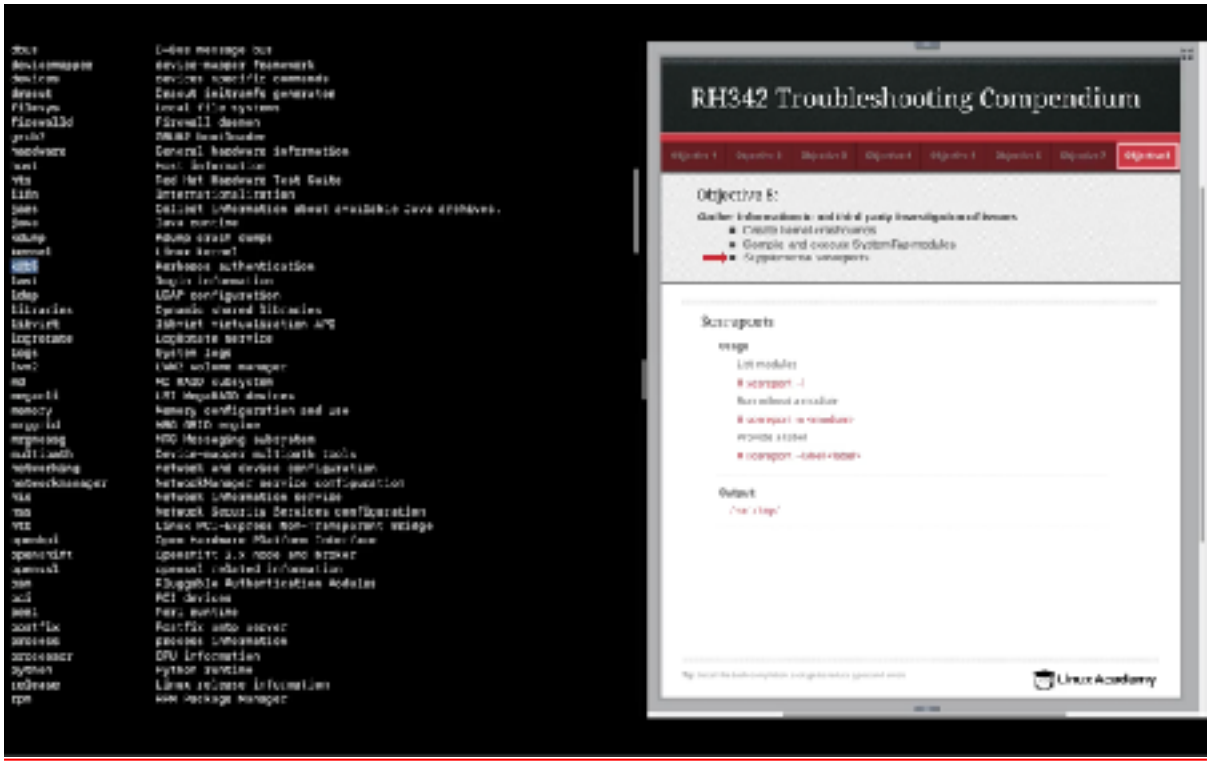
```
# sosreport --label <label>
```

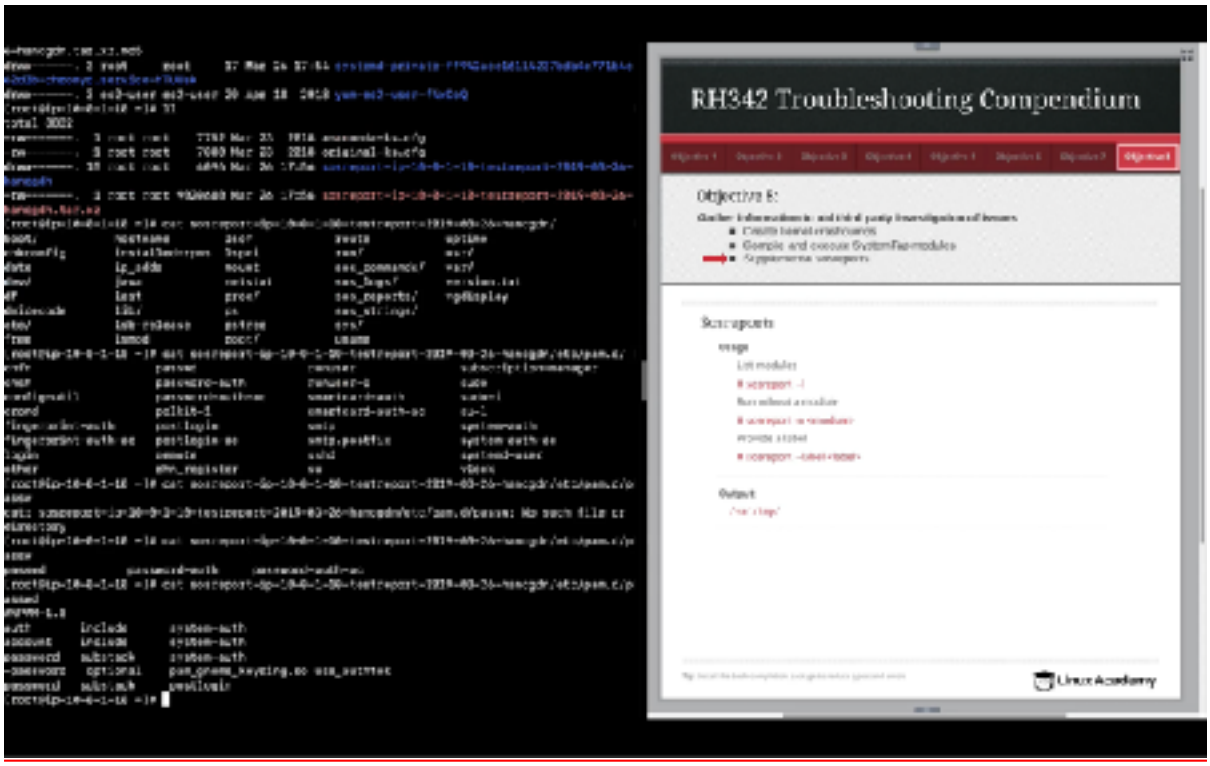
Output

```
/var/tmp/
```


RH342 Troubleshooting Compendium
Objective 5:
Scrapports
Output
Linux Academy


RH342 Troubleshooting Compendium
Objective 5:
Scrapports
Output
Linux Academy





Troubleshooting the current system load

Objectives:

Learning Objectives

Collect system information.

You should begin by logging in to the host and obtaining the following information to verify the user complaints:

- Current system load
- Current disk usage
- Swap usage

Performance Co-Pilot is installed. You should use it to help assess the problems.

- Enable `pmcd`
- Start `pmcd`

Resolve disk space issues.

You will need to find out what's consuming all of the disk space and see if it's something that can be removed.

Resolve CPU load issues.

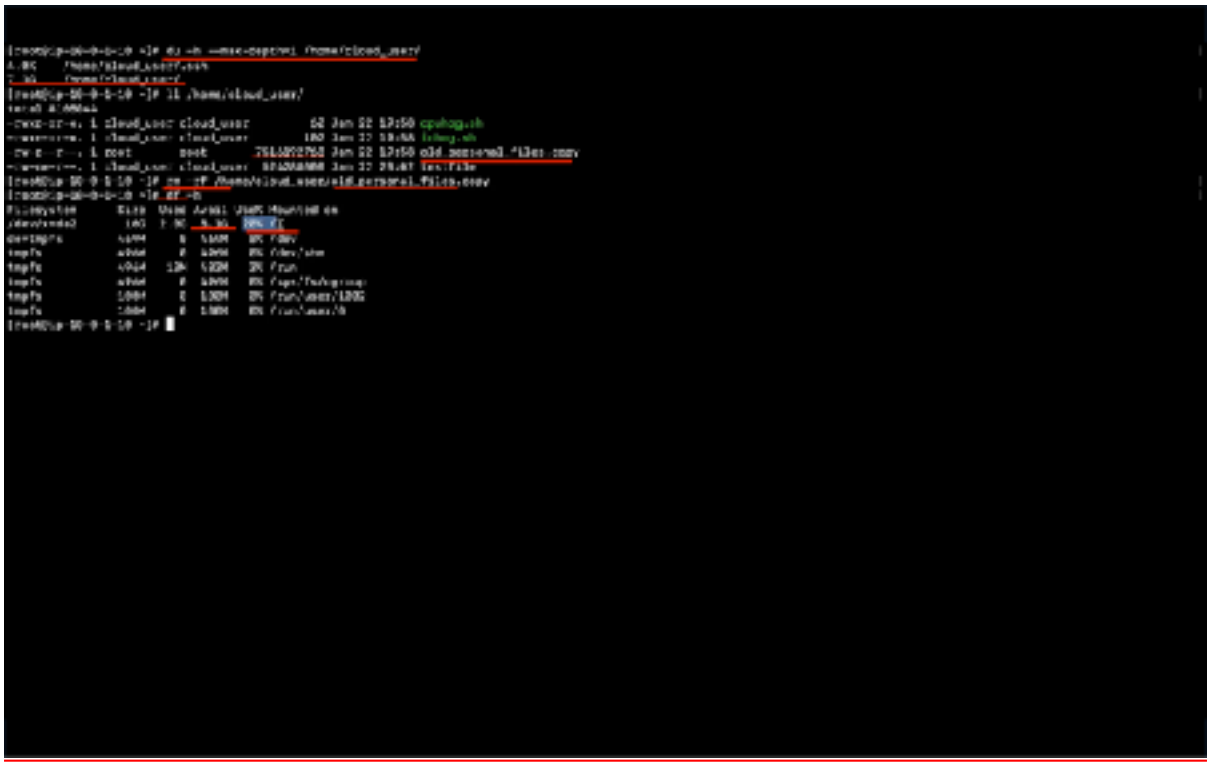
You will need to investigate the cause of CPU load issues and resolve them.

Resolve disk I/O issues.

You will need to investigate the cause of disk I/O issues and resolve it.

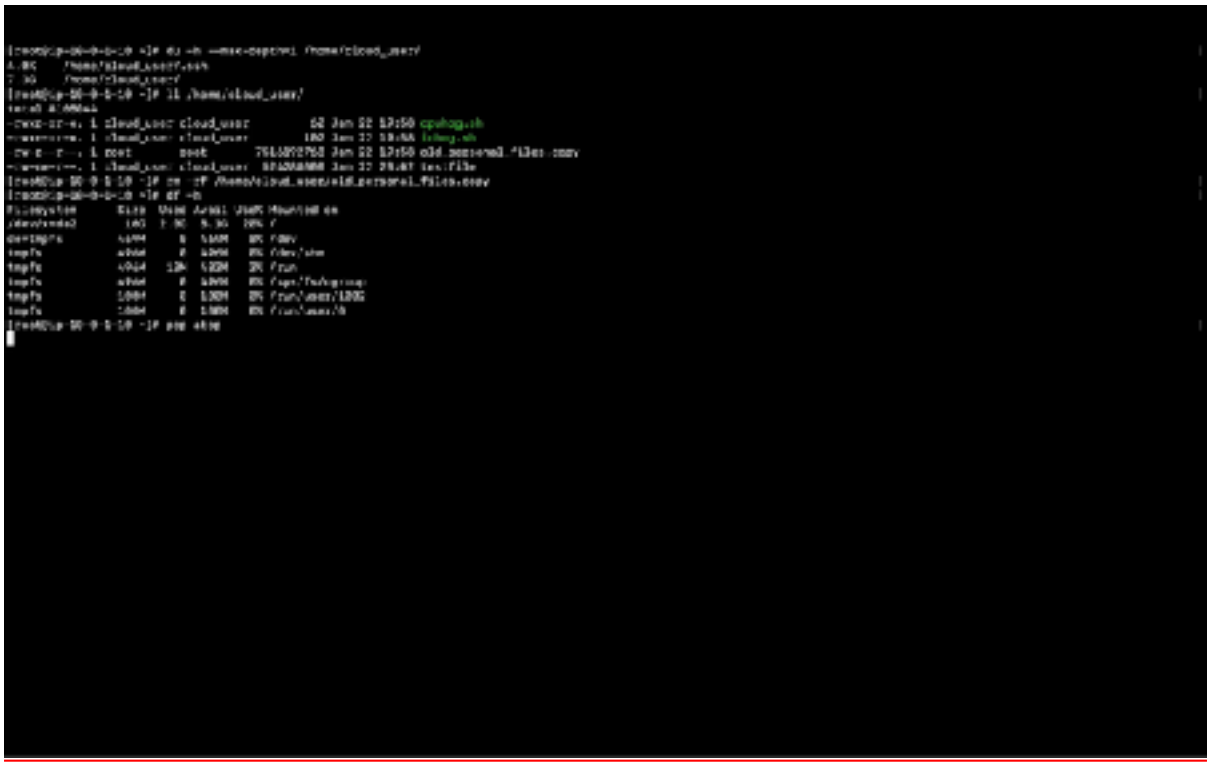
```
[redacted] password for cloud_user:
[redacted]
[redacted]
[redacted] -lf uptime
 [redacted] up 7 min,  1 user,  load average: [redacted]
[redacted] -lf df -h
Filesystem      Size  Used Avail Use% Mounted on
/dev/xvda2      [redacted]
devtmpfs        [redacted]  0  [redacted]  0% /dev
tmpfs           [redacted]  0  [redacted]  0% /dev/shm
tmpfs           [redacted]  [redacted]  [redacted] /run
tmpfs           [redacted]  0  [redacted]  0% /sys/fs/cgroup
tmpfs           [redacted]  0  [redacted]  0% /run/user/1001
[redacted] -lf ntpq
-bash: ntpq: command not found
[redacted] -lf uptime
[redacted]
[redacted] -lf free -m
              total        used        free      shared  buff/cache   available
Mem:           [redacted]         [redacted]         [redacted]          12         [redacted]         [redacted]
Swap:             0           0           0
[redacted] -lf systemctl enable pmod && systemctl start pmod
Created symlink from /etc/systemd/[redacted]-user.target.wants/pmod.service to /usr/lib/systemd/system/pmod.service.
```

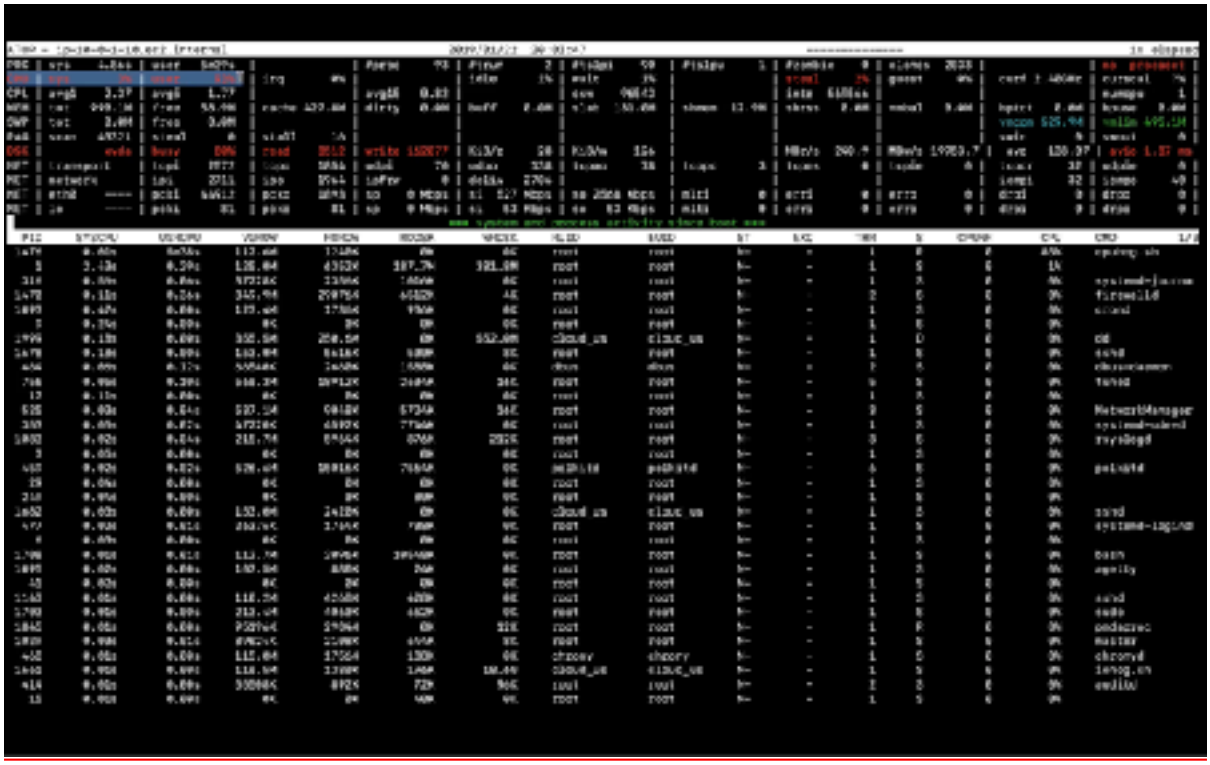
```
[redacted] -lf df -h
Filesystem      Size  Used Avail Use% Mounted on
/dev/xvda2      [redacted]
devtmpfs        [redacted]  0  [redacted]  0% /dev
tmpfs           [redacted]  0  [redacted]  0% /dev/shm
tmpfs           [redacted]  [redacted]  [redacted] /run
tmpfs           [redacted]  0  [redacted]  0% /sys/fs/cgroup
tmpfs           [redacted]  0  [redacted]  0% /run/user/1001
tmpfs           [redacted]  0  [redacted]  0% /run/user/0
[redacted] -lf
```

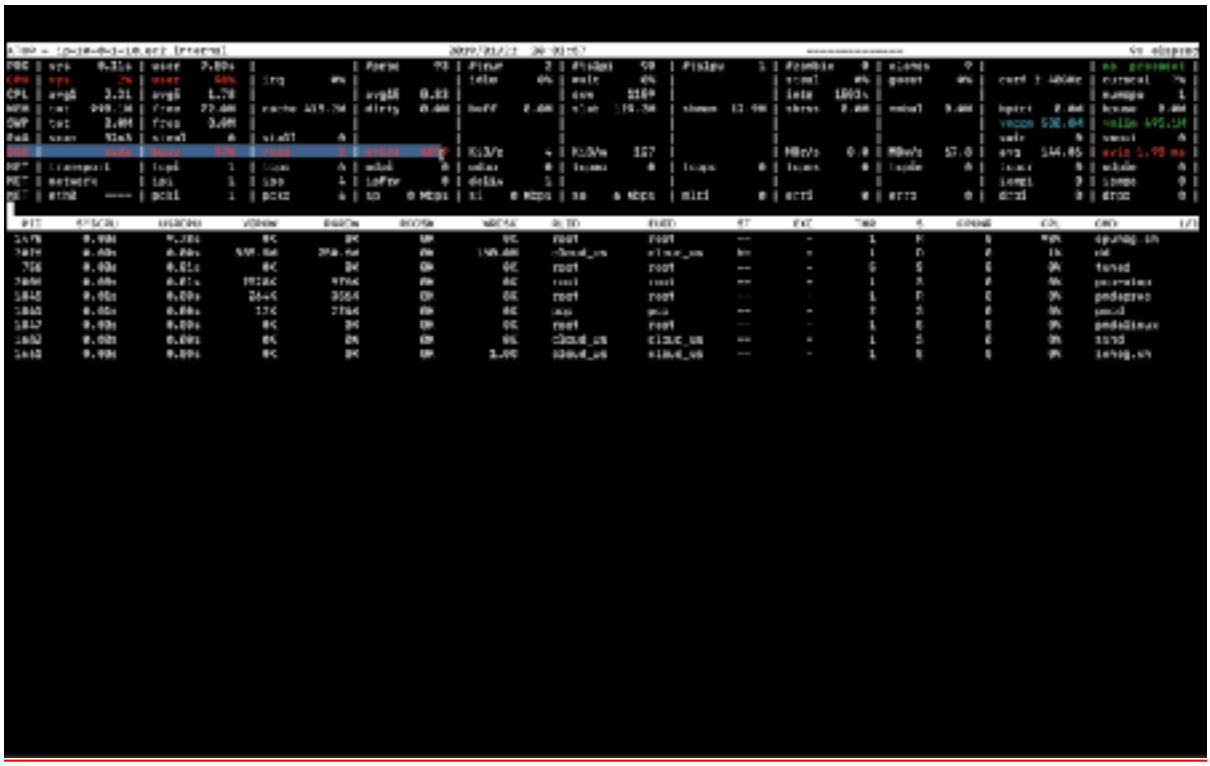


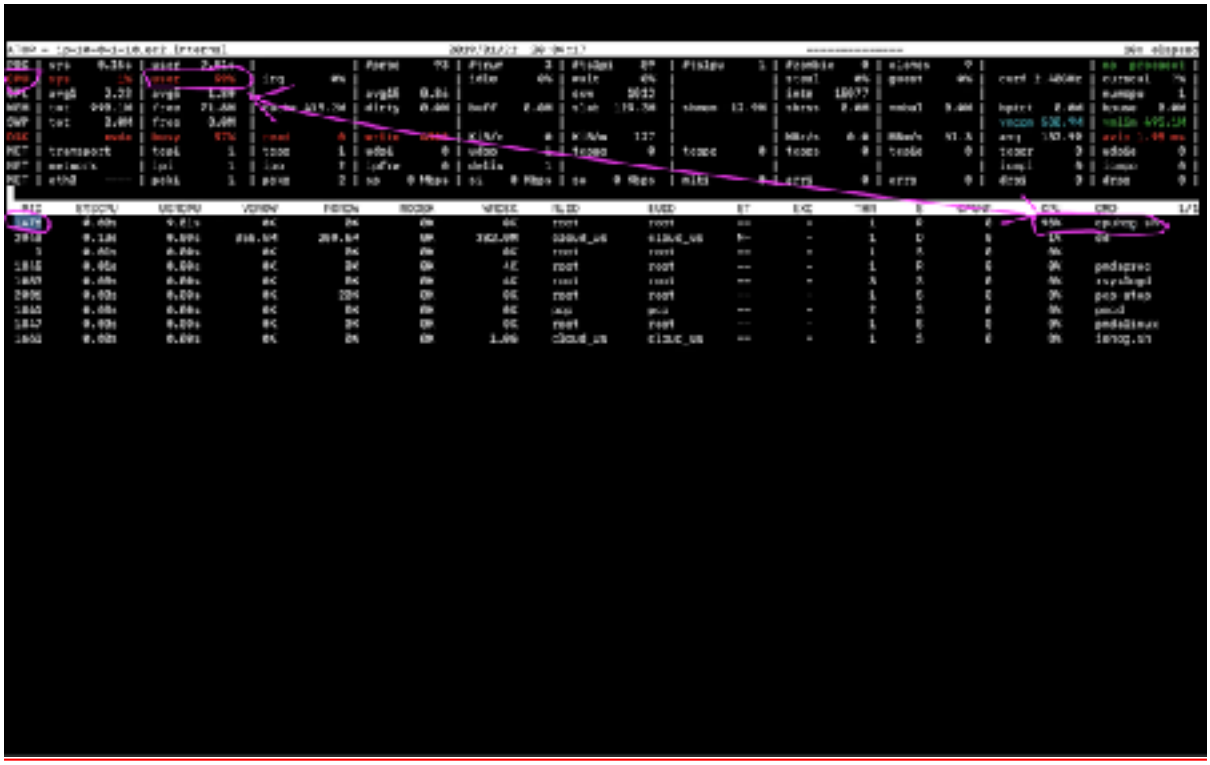
Use “atop” to understand with system(atop) recommandetations

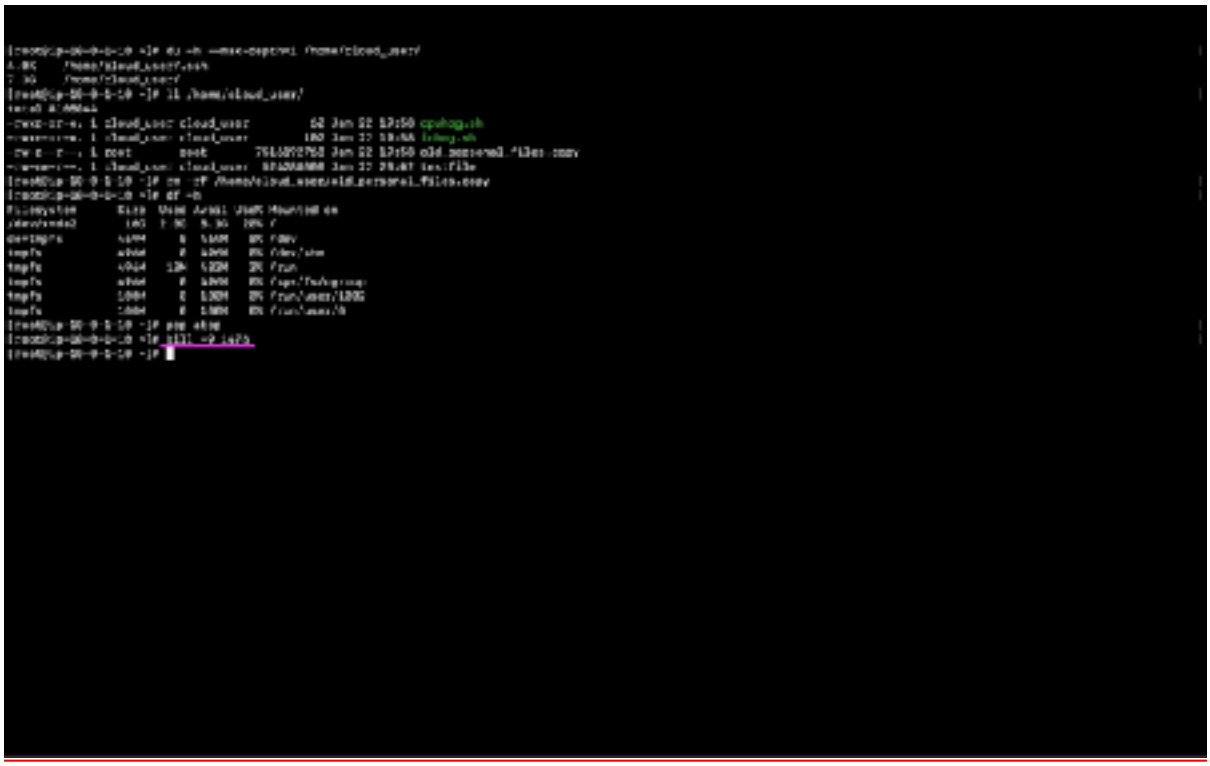
Danger zone shows in RED color higlighted

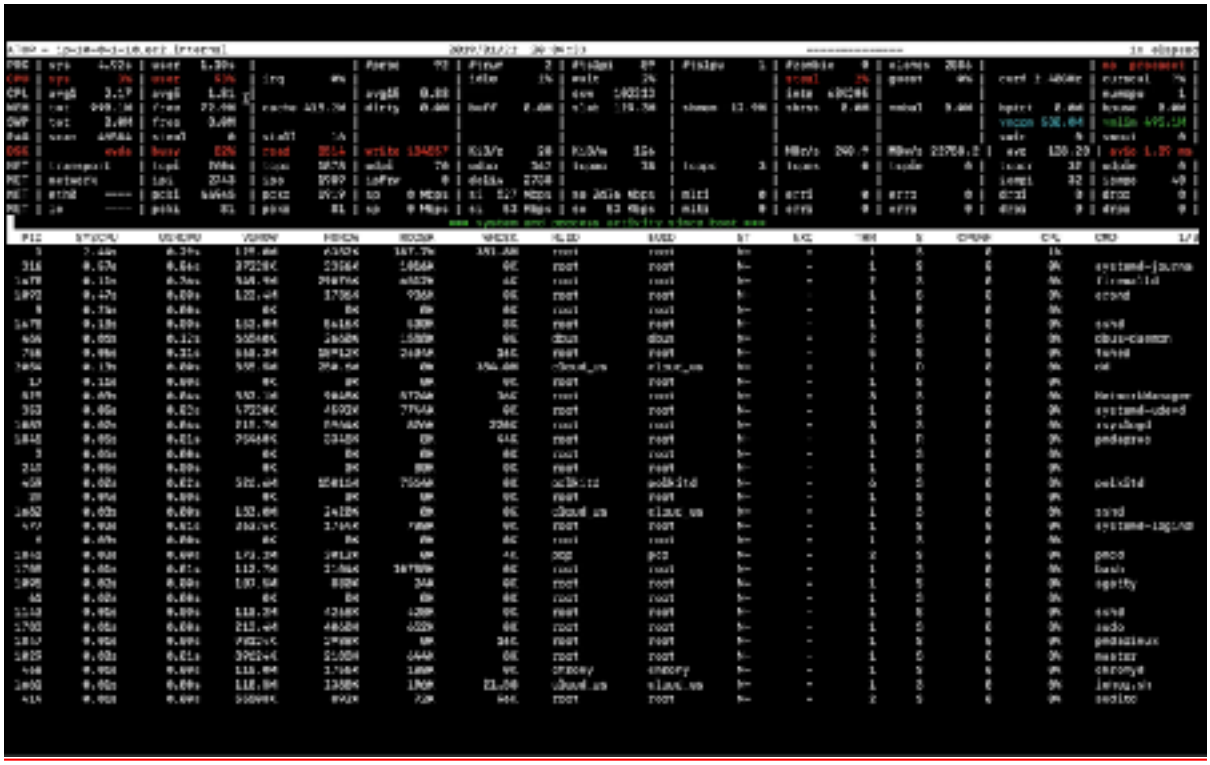


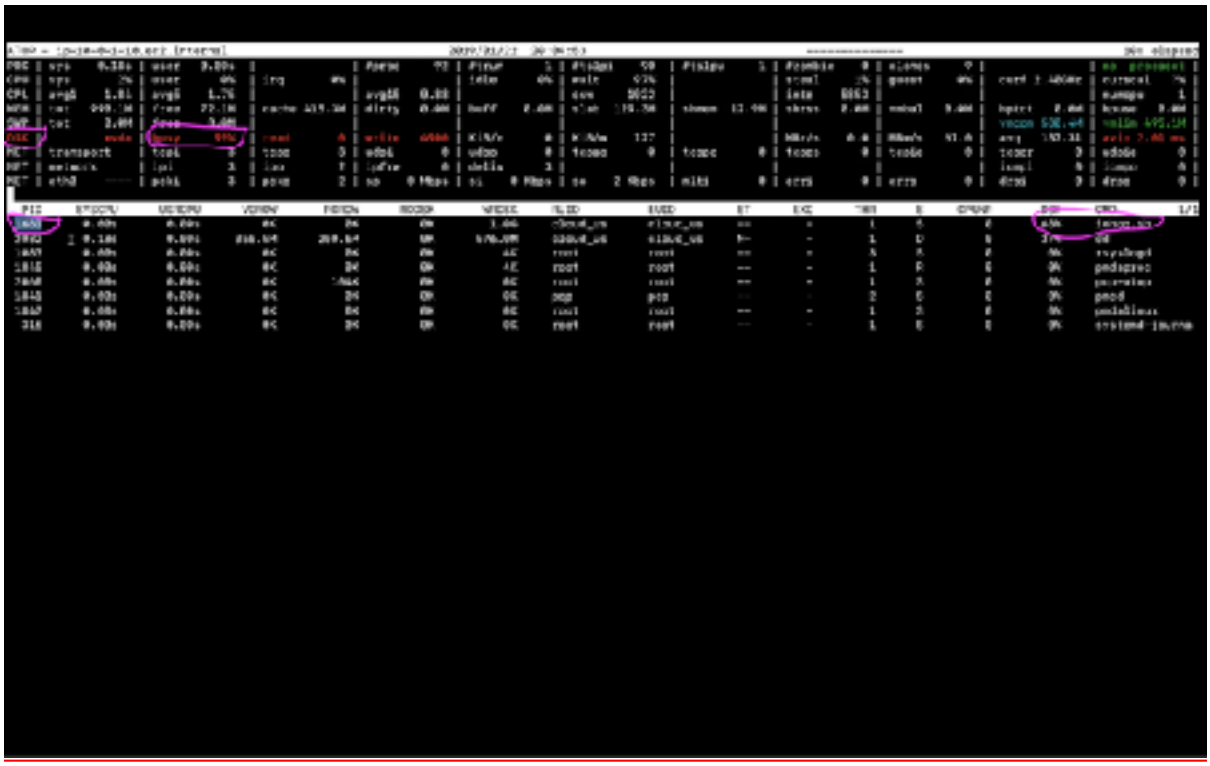












```
[root@ip-10-0-0-10 ~]# du -h --max-depth=1 /home/cloud_user/
4.0K    /home/cloud_user/.ssh
7.3G    /home/cloud_user/
[root@ip-10-0-0-10 ~]# ll /home/cloud_user/
total 8.0KB
-rwxr-xr-x. 1 cloud_user cloud_user        62 Jan 22 17:50 cpuhog.sh
-rwxrwxrwx. 1 cloud_user cloud_user       180 Jan 22 18:06 iohog.sh
drwxr-xr-x. 1 root       root       7516192768 Jan 22 17:50 old_personal_files.zip
-rwxrwxrwx. 1 cloud_user cloud_user  8589934080 Jan 22 20:07 test.file
[root@ip-10-0-0-10 ~]# rm -rf /home/cloud_user/old_personal_files.zip
[root@ip-10-0-0-10 ~]# df -h
Filesystem      Size  Used Avail Use% Mounted on
/dev/xvda2       10G  7.7G  9.3G  45% /
devtmpfs        4.0M     0  4.0M   0% /dev
tmpfs           4.0M     0  4.0M   0% /dev/shm
tmpfs           494M  12M  482M   2% /run
tmpfs           4.0M     0  4.0M   0% /sys/fs/cgroup
tmpfs           100M     0  100M   0% /run/user/1000
tmpfs           100M     0  100M   0% /run/user/0
[root@ip-10-0-0-10 ~]# ps aux
[root@ip-10-0-0-10 ~]# kill -9 1475
[root@ip-10-0-0-10 ~]# ps
ps aux
[root@ip-10-0-0-10 ~]# kill -9 1461
[root@ip-10-0-0-10 ~]# uptime
 20:04:22 up 9 min,  1 user,  load average: 0.07, 1.38, 0.88
[root@ip-10-0-0-10 ~]# df -h
Filesystem      Size  Used Avail Use% Mounted on
/dev/xvda2       10G  7.7G  2.0G  79% /
devtmpfs        4.0M     0  4.0M   0% /dev
tmpfs           4.0M     0  4.0M   0% /dev/shm
tmpfs           494M  12M  482M   2% /run
tmpfs           494M     0  494M   0% /sys/fs/cgroup
tmpfs           100M     0  100M   0% /run/user/1000
tmpfs           100M     0  100M   0% /run/user/0
[root@ip-10-0-0-10 ~]#
```

```
top - 12:34:56 up 3:18, 2 users,  load average: 0.42, 0.38, 0.31                    Tasks: 142 total,   1 running, 141 sleeping,   0 stopped,   0 zombie
%Cpu(s):  0.8 us,  0.2 sys,  0.0 ni, 98.9 id,  0.1 wa,  0.0 hi,  0.0 si,  0.0 st     KiB Mem :  1024.0 total,   312.4 free,   512.8 used,   198.8 buff/cache
KiB Swap:  2048.0 total,  2048.0 free,     0.0 used.   421.6 avail Mem

  PID USER      PR  NI    VIRT    RES    SHR S  %CPU  %MEM     TIME+ COMMAND
 1234 root      20   0  412356  18240   9216 S   0.3   1.7   0:12.04 pmdaproc
    1 root      20   0  169432   9876   6540 S   0.0   0.9   0:03.21 systemd
 2172 root      20   0  245108  21340   8192 S   0.3   2.0   0:08.55 pcp-atop
 1243 root      20   0  198764  14208   7168 S   0.0   1.3   0:02.17 pmcd
 1317 root      20   0  156320  12480   6144 S   0.0   1.2   0:01.03 pmdalinux
 1240 root      20   0  134512   9216   5120 S   0.0   0.8   0:00.41 sshd
```

Collect system information

You should begin by logging in to the host and obtaining the following information to verify the user complaints:

- Current system load:

```
uptime
```

- Note the load average shown and compare it to how many processors this server has:

```
nproc
```

- Current disk usage:

```
df -h
```

- Swap usage:

```
free -m
```

Performance Co-Pilot is installed. You should use it to help assess the problems.

- Enable `pmcd`

```
systemctl enable pmcd
```

- Start `pmcd`

```
systemctl start pmcd
```

Resolve disk space issues

1. Start by using `du` to find large directories in `/`:

```
du -h --max-depth=1 /
```

2. We have narrowed down which directory is consuming the largest amount of space. Let's try and narrow it down a little more:

```
du -h --max-depth=1 /home/
```

3. Narrow it down a little further:

```
du -h --max-depth=1 /home/cloud_user/
```

4. The space is being consumed by a file in this directory. Let's list the files in this directory:

```
ll /home/cloud_user/
```

5. We can see a rather large file called `old_personal_files.copy` in this directory. Let's delete this file:

```
rm -rf /home/cloud_user/old_personal_files.copy
```

6. Verify the disk space issue is now resolved:

```
df -h
```

Resolve CPU load issues

1. Since we have `pmcd` running, let's use one of it's tools to see what is causing our CPU load issue:

```
pcp atop
```

2. The list of processes are sorted by CPU usage. This makes it easy to see that the `cpuhog.sh` process is using the largest amount of CPU. Copy the process ID (PID) for the `cpuhog.sh` process and close `pcp atop`.

3. Now we can kill that process. Replace `PID` in the following command with the PID of the process:

```
kill -9 PID
```

4. Verify that the CPU issue is resolved by running `pcp atop` again. After a moment, we should see the CPU usage return to a normal value.

```
pcp atop
```

Resolve disk I/O issues

1. With `pcp atop` still running, sort the list of processes by disk I/O by pressing **SHIFT+D**.
2. We can see an `iohog.sh` command is causing the issue. Same as before, copy the PID for that process and close `pcp atop`.
3. We can now kill that process. Replace `PID` in the following command with the PID of the process:

```
kill -9 PID
```

4. Verify that the disk I/O issue is resolved by running `pcp atop` again. After a moment, we should see disk I/O return to normal.

```
pcp atop
```

Troubleshooting network issues

Objectives

Learning Objectives

Verify Server1.

Verify ***Server1*** is running Apache:

```
systemctl status httpd
```

Verify Apache is listening on port 80:

```
ss -lntp | grep 80
```

Verify the content can be accessed locally:

```
curl localhost
```

Verify the firewall is permitting port 80 traffic:

```
firewall-cmd --list-all
```

Permit inbound traffic to Apache:

```
firewall-cmd --permanent --add-service=http
```

Reload the firewall to pickup the change:

```
firewall-cmd --reload
```

Verify Server2.

On ***Server2*** (10.0.1.11)

Attempt to pull the web content from ***Server1*** via hostname:

```
curl ip-10-0-1-10.ec2.internal
```

Attempt to pull the web content from ***Server1*** via IP

```
curl 10.0.1.10
```

Check to see what DNS hosts are being used:

```
cat /etc/resolv.conf
```

Verify resolution via the specified host:

```
dig @10.0.0.2 ip-10-0-1-10.ec2.internal
```

Verify `/etc/nsswitch.conf`:

```
vim /etc/nsswitch.conf
```

Attempt to pull the web content from ***Server1*** via hostname:

```
curl ip-10-0-1-10.ec2.internal
```

Ping `ip-10-0-1-10.ec2.internal`:

```
ping ip-10-0-1-10.ec2.internal
```

Review `/etc/hosts`:

```
vim /etc/hosts
```

Attempt to pull the web content from ***Server1*** via hostname:

```
curl ip-10-0-1-10.ec2.internal
```

Install `elinks`:

```
yum install -y elinks
```

Use `elinks` to view the content:

```
links -dump http://ip-10-0-1-10.ec2.internal
```

Quiestion1:

1. Performance Co-Pilot.

a. Install pcp and pcp-system-tools.

b. Enable and start the pmcd and pmlogger services

```
[root@ip-10-0-1-10 ~]# systemctl  enable pmcd && systemctl  start pmcd && systemctl status pmcd

● pmcd.service - Performance Metrics Collector Daemon


[root@ip-10-0-1-10 ~]# systemctl enable pmlogger
 [root@ip-10-0-1-10 ~]# systemctl start pmlogger
[root@ip-10-0-1-10 ~]# systemctl status pmlogger
```

2. Take a baseline of CPU load.

Take a baseline of the kernel.all.load metric for 10 seconds and put this into the file /home/cloud_user/kernel.all.load.txt.

You can do this using the pmval or pmrep command:

<Learn about pcp & pmval , pmrep, diskload , cpuload comands>