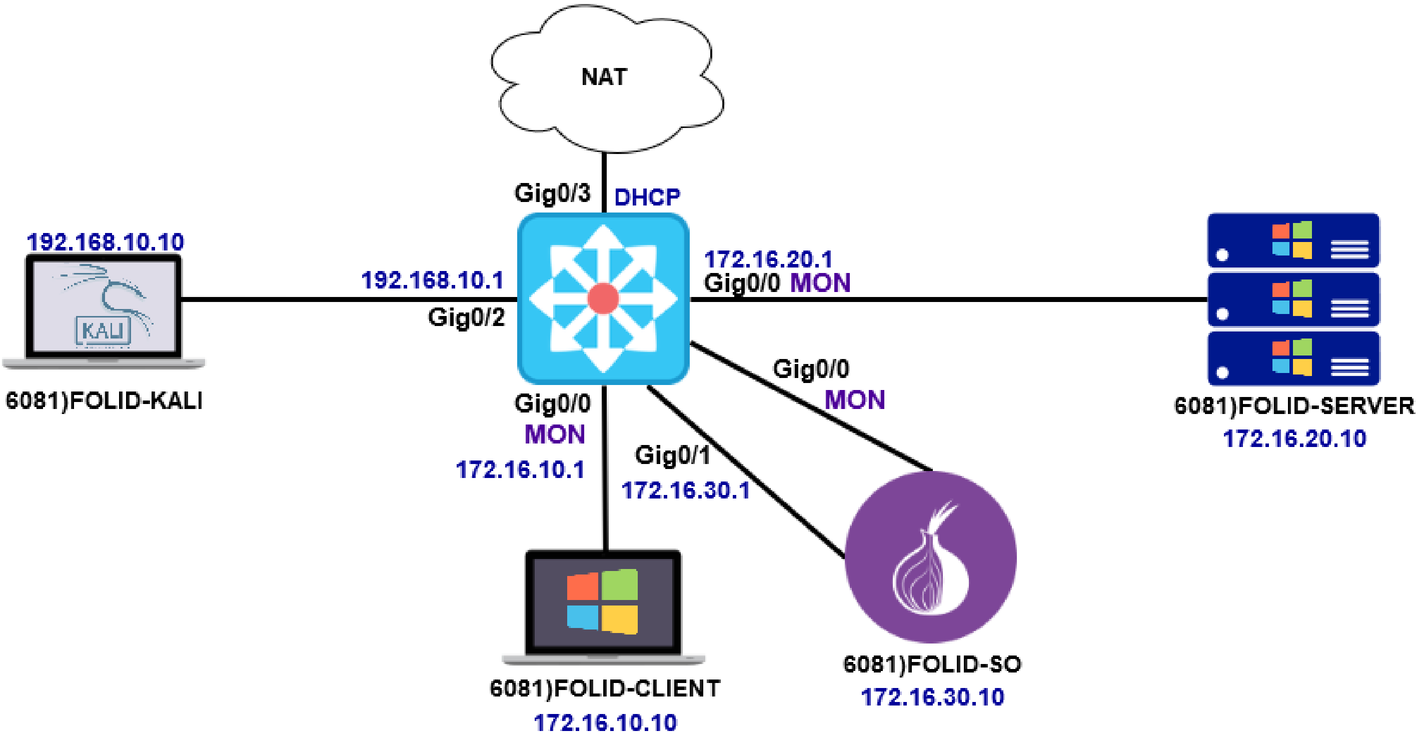
Lab 3 – Maintaining Security Onion

Lab Topology and Learning Goals



In this lab you learn how to perform regular maintenance and management tasks on a security onion host.

Required Resources

- **VMware Workstation 15**

Active Hosts

- **6081)Router**
- **6081)FOLID-SO**

Submission Instructions

Submit your completed lab to the appropriate lab quiz on FOL

- You can attempt the quiz multiple time, but only the last attempt will be graded
- Submissions are accepted until 11:59 PM of the same day
- Submissions by email will not be accepted
- All screenshots must include you FOLID (where FOLID is your FOL username)

Lab 3 – Maintaining Security Onion

Updating Security Onion

At the end of the last lab, you should have taken a running snapshot of your SO VM. Instead of powering on this VM, simply restore the VM snapshot to return the VM to a running state.

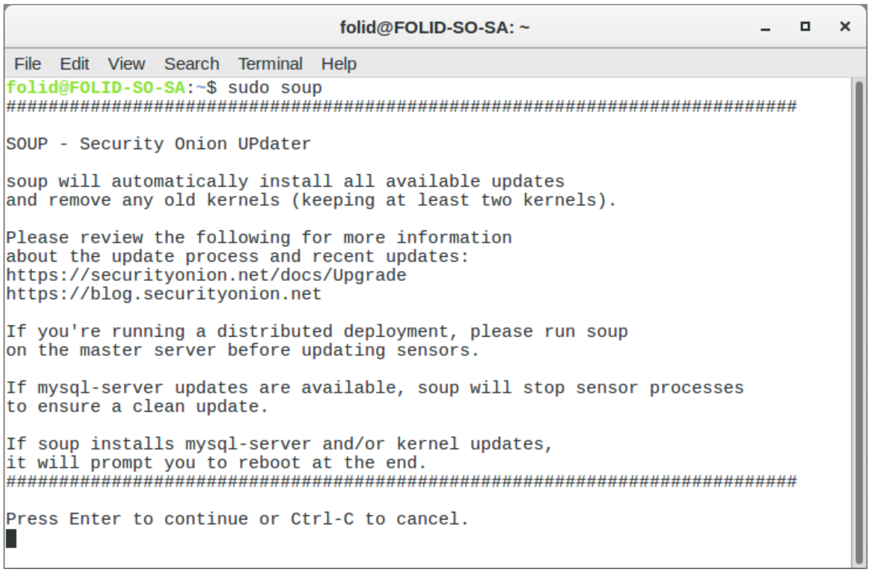
Security Onion contains several custom scripts to manage updating the components of the setup.

Ensure that you have internet connectivity before continuing

Soup

The Security Onion Update Script (soup) is to update the components of the operating system, docker images and installed applications. When executing the script, pay attention to the output, as you may be required to take actions.

To run the script, execute:



```
folid@FOLID-SO-SA:~$ sudo soup
###########################################################################

SOUP - Security Onion UPdater

soup will automatically install all available updates
and remove any old kernels (keeping at least two kernels).

Please review the following for more information
about the update process and recent updates:
https://securityonion.net/docs/Upgrade
https://blog.securityonion.net

If you're running a distributed deployment, please run soup
on the master server before updating sensors.

If mysql-server updates are available, soup will stop sensor processes
to ensure a clean update.

If soup installs mysql-server and/or kernel updates,
it will prompt you to reboot at the end.
###########################################################################

Press Enter to continue or Ctrl-C to cancel.
```

user@hostname:~$ sudo soup

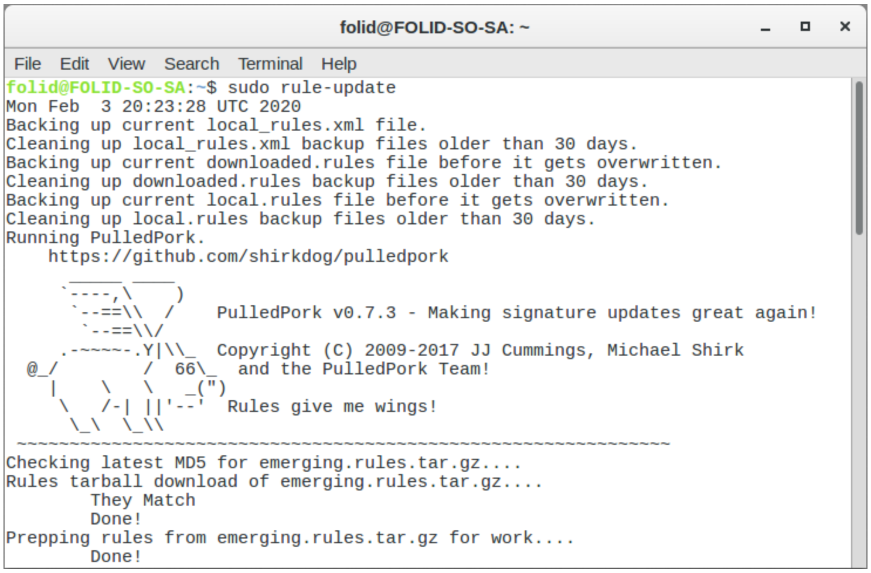
Review the results of the update. What components were updated?

When prompted, reboot the machine

Intrusion Detection System (IDS) Rule Update

The **rule-update** script is used to update the rule sets for SNORT and Suricata. This script should be regularly executed to ensure that the IDS is scanning for newly discovered threats. When executed, the **snort.conf** and **suricata.yaml** files have the .bak extension appended and the file is replaced with a new version. Attributes in the **HOME_NET** and **EXTERNAL_NET** variables are migrated to the new file. Any customizations to the configuration files must be manually migrated to the new file. As we have not enabled Suricata, it will not be updated.

To run the script, execute:



```
folid@FOLID-SO-SA:~$ sudo rule-update
Mon Feb  3 20:23:28 UTC 2020
Backing up current local_rules.xml file.
Cleaning up local_rules.xml backup files older than 30 days.
Backing up current downloaded.rules file before it gets overwritten.
Cleaning up downloaded.rules backup files older than 30 days.
Backing up current local.rules file before it gets overwritten.
Cleaning up local.rules backup files older than 30 days.
Running PulledPork.
    https://github.com/shirkdog/pulledpork

      _____ ____
     `----,\    )
      `--==\\  /    PulledPork v0.7.3 - Making signature updates great again!
       `--==\\/
     .-~~~~-.Y|\\_  Copyright (C) 2009-2017 JJ Cummings, Michael Shirk
  @_/        /  66\_  and the PulledPork Team!
    |    \   \   _(")
     \   /-| ||'--'  Rules give me wings!
      \_\  \_\\
 ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
Checking latest MD5 for emerging.rules.tar.gz....
Rules tarball download of emerging.rules.tar.gz....
        They Match
        Done!
Prepping rules from emerging.rules.tar.gz for work....
        Done!
```

user@hostname:~$ sudo rule-update

Review the results of the update. Notice that affected services were automatically restarted.

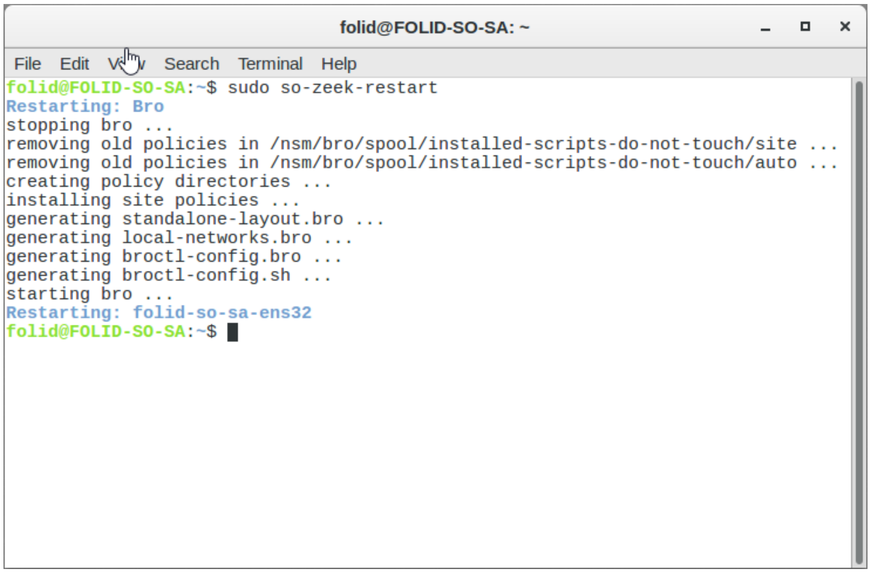
Add a screenshot showing the updated rule stats and the restarted services to the Lab 3 quiz

Zeek

Zeek (formerly known as bro; however, packages and configurations still reference the bro name) is a network analysis framework that monitors network traffic and creates logs for items such as:

- TCP/UDP/ICMP connections
- DNS activity
- SSL/TLS Handshake information
- HTTP/FTP activity
- Email activity
- Syslog events

Updating Zeek packages will attempt to migrate configuration customizations. It is recommended to confirm custom configurations (such as email inspection) are intact and restart he service:



```
folid@FOLID-SO-SA:~$ sudo so-zeek-restart
Restarting: Bro
stopping bro ...
removing old policies in /nsm/bro/spool/installed-scripts-do-not-touch/site ...
removing old policies in /nsm/bro/spool/installed-scripts-do-not-touch/auto ...
creating policy directories ...
installing site policies ...
generating standalone-layout.bro ...
generating local-networks.bro ...
generating broctl-config.bro ...
generating broctl-config.sh ...
starting bro ...
Restarting: folid-so-sa-ens32
folid@FOLID-SO-SA:~$
```

```
user@hostname:~$ sudo so-zeek-restart
```

Elastic Stack

Elastic stack should automatically restart, if it fails to appear in Kibana, the following command can be used to correct the setup:

```
user@hostname:~$ sudo so-elastic-configure-kibana
```

Managing OS Accounts

Security Onion makes use of a number of OS and application accounts to provide services to users. Based on Ubuntu, SO has the root account disabled by default; however, the user created during setup has sudo privileges granted at setup.

Change OS Account Password (not required)

The following command can be used to change the OS user account password:

```
user@hostname:~$ passwd
```

Add a New OS Account

To add a new OS user account, use the **adduser** command. Create a new user using the value of your FOLID as the username:

```
user@hostname:~$ sudo adduser folid
```

When prompted, use **Windows1** as the password, answer other prompts as required (you can leave some blank).

To grant sudo privileges to the user, execute the following command (where folid is the newly created username):

```
user@hostname:~$ sudo usermod -aG sudo folid
```

To configure SO to automatically login with the new user account (only acceptable in a lab environment), edit the GDM3 configuration file:

```
user@hostname:~$ sudo nano /etc/lightdm/lightdm.conf
```

Change the value listed for the attribute **autologin-user** to the user you created

Reboot the host and ensure that you are auto-logged in with the user you just created

List All OS Accounts

On Linux distributions, user accounts are store in the **/etc/passwd** file. A short list of user accounts can be displayed from this list by executing the command:

```
user@hostname:~$ cut -d: -f1 /etc/passwd
```

Add a screenshot showing the end of the output (displaying your newly created user) to the Lab 3 quiz

Disable an OS Account

To disable an OS account, you can set the account as expired and prevent future logins. Execute the following command to disable the **administrator** account

```
user@hostname:~$ sudo usermod --expiredate 1 administrator
```

Managing Single Sign On (SSO) Accounts

SO uses the account that is created for the squil service as a single sign on account for most functions (squirt, kibana, etc.). This user is created during the configuration portion of the installation; however, organizations with multiple users will require multiple accounts.

Change SSO Account Password (not required)

The following command can be used to change the SO SSO account password:

```
user@hostname:~$ sudo so-user-passwd
```

Add a New SSO Account (not required)

To add a new SO SSO account, execute the following command:

```
user@hostname:~$ sudo so-user-add
```

List All SSO Accounts

To list all SO SSO accounts, execute the following command:

```
user@hostname:~$ sudo so-user-list
```

Disable an SSO Account (not required)

To disable a SO SSO account, execute the following command:

```
user@hostname:~$ sudo so-user-disable
```

Re-Enable an SSO Account (not required)

To reinstate a previously disabled SSO account, execute the following command:

```
user@hostname:~$ sudo so-user-passwd
```

MySQL (not required)

Upon installation, a random password is generated for the MySQL instance. If manual management of the database is required, you must clear the generated defaults and manually configure the settings. The following command returns the auto-generated settings to default:

```
user@hostname:~$ sudo mysql --defaults-file=/etc/mysql/debian.cnf
```

Managing SO Services

SO includes easy to use scripts that automate much of the required service management and their tasks. Some of the key scripts include:

Get the status of all services:

```
user@hostname:~$ sudo so-status
```

Start all services:

```
user@hostname:~$ sudo so-start
```

Stop all services:

```
user@hostname:~$ sudo so-stop
```

Restart all services:

```
user@hostname:~$ sudo so-restart
```

The **so-*** portion of the command can also be used to control individual services:

Get the status of sguild:

```
user@hostname:~$ sudo so-sguild-status
```

Get the status of zeek:

```
user@hostname:~$ sudo so-zeek-status
```

Get the status of the elastic stack:

```
user@hostname:~$ sudo so-elastic-status
```

Likewise, individual services can be supplemented into the **start**, **stop** and **restart** commands.

Managing Firewall Services

On standalone deployments, SO defaults to allow only connections on **TCP port 22 (SSH).**

On distributed deployments, SO allows **TCP ports 22 (SSH), 4505/6 (SALT), and 7736 (SGUIL).**

To allow connections on alternative ports, the **so-allow** utility can be used to create additional rules.

Similarly, the **so-disallow** utility can revoke access.

Managing Log Retention

When dealing with the high volumes of traffic that retaining network logs creates, it is important to automate log retention. SO has default retention values for each individual service that can be customized if necessary.

Sguil

To configure sguil's database retention, edit the security onion configuration:

```
user@hostname:~$ sudo nano /etc/nsm/securityonion.conf
```

Find the attribute **DAYSTOKEEP** and observe the default value.

You can also manually purge the sguil database with the command:

```
user@hostname:~$ sudo sguil-db-purge
```

netsniff-ng (Full Packet Capture)

To configure retention of full packet captures, edit the security onion configuration:

```
user@hostname:~$ sudo nano /etc/nsm/securityonion.conf
```

Find the attribute **CRIT_DISK_USAGE** and observe the default value.

Elastic Search Logs

To configure retention of Elastic Search Logs, edit the security onion configuration:

```
user@hostname:~$ sudo nano /etc/nsm/securityonion.conf
```

Find the attribute **LOG_SIZE_LIMIT** and observe the default value.

On the terminal, execute the following command: grep 'DAYSTOKEEP\|CRIT_DISK_USAGE\|LOG_SIZE_LIMIT' /etc/nsm/securityonion.conf

Add a screenshot showing the output to the Lab 3 quiz

Lab 3 – Maintaining Security Onion

Take a running snapshot of your **SO** host called **Lab 3 Complete**, then shutdown.

Shutdown the other hosts and take a snapshot called **Lab 3 complete**

Submit your completed **Lab 3** quiz

References

- Security Onion Solutions. (2020). Security Onion: Security Onion Documentation. Evans, GA: Author.