Getting Started

Note: The assumption for the exercise is that Bro is already installed or run in a VM. In this exercise, "<PREFIX>" represents the Bro install directory. Basic Linux knowledge, e.g., less, grep, man, etc, is a prerequisite.
Advanced users who want more of a challenge may skip directly to part D and solve the second exercise marked **Level advanced**.

A) BroControl

BroControl is an interactive shell for easily operating/managing Bro installations on a single system or even across multiple systems in a traffic-monitoring cluster. To learn more about BroControl please refer to the documentation. The following exercise is to start and use BroControl.

Exercise 1

Level beginner
Start up BroControl:

```
broctl
```

The first time that you run BroControl, you must install the BroControl configuration:

```
    [BroControl] > install
```

Now type:

```
    [BroControl] > help
```

Use the BroControl help to achieve the following tasks:

- Start a Bro instance and check if it is running.
- Find out which nodes are running. Which interfaces are monitored?
- What is the current packet count of the first node?
- Find all Bro processes running.
- What type is your Bro instance?
- Stop Bro and check if it has stopped.
- Exit BroControl.

B) Run Bro directly

If you don’t want to use BroControl, then you can run Bro directly. When you run Bro directly, it creates its log files in the current working directory. Therefore, it is a good idea to create a temporary directory so that you can more easily see which files are generated by Bro. Using Bro directly is especially useful when you want to analyze packet capture (pcap) traffic files.

Exercise 2

Level beginner
Use the ”bro” command with "--help" to find the option that allows you to read a pcap file. Read the pcap file http.pcap and examine the log files that are created. `--help` also tells you how to use policy script in the direct mode. Analyze the same pcap again using the script `extract-all.bro`. Examine the logs. What is different now?

C) Bro log files -- warm up

After completing the last two exercises you should now have two different log sets. Logs created by BroControl are stored in the "<PREFIX>/logs" directory. The logs directory contains a subdirectory named with today’s date (in the form of YYYY-MM-DD). This date-named subdirectory contains various log files (all are gzipped) which are copied into this subdirectory when Bro is stopped. Running Bro directly writes the logs into your working directory, as discussed above. These files are not zipped. One more notable difference between BroControl and running Bro directly is that BroControl loads local.bro scripts, whereas running Bro directly only runs "base" scripts.

Exercise 3

Level beginner
This exercise gives you an overview of the default logs generated by Bro. Find all the logs created during the previous exercises. Examine both log sets, the one created by BroControl and the one created using Bro directly. What is the difference? Why are they different?
The command

```
gunzip -dc <PREFIX>/logs/<date>/<log> | less
```

allows you to look inside the zipped logs. The other not-zipped logs can be examined using "less".
Answer the following questions by scanning through the logs. You can select one of the log sets or go through both.

- Which scripts were loaded by Bro?
- How can you find out if there were any network connections?
- Identify source and destination port of a connection.
- Look in the log weird.*. Is there anything? What is it? Why does Bro consider it weird?

- If so, use the UID of the first connection to find the same connection in the connection log. What could have happened there?

D) Bro log files -- digging deeper

In the following exercise a pcap file is provided that potentially contains real malware. **Do not execute files or follow links with a real system.** Use the Bro training VM, Bro-Live, try.bro.org, or whatever training environment you are using for these exercises. The objective of this exercise is to learn more about what can be done with Bro logs. There are two parts: one for beginners and one for more advanced Bro users.
The pcap used in this exercise was provided by http://forensicscontest.com/ and is their property.
For this exercise we assume a certain set of skills:

- Basic Unix tools, e.g., cat, sort, etc.
- Basic awk (or Perl) to process files
- Basic networking

Exercise 4

Level beginner
Produce log files the same way as above, using the file infected.pcap.

- Use bro-cut with the "--help" option learn how to extract the 4-tuple of a connection, the UID of the log record, and the duration.
- Find the top five connections with the longest duration. Why could it be useful to filter out the longest connections?
- Find the connection with the largest data transfer. What was transferred? From where to where? List the URL.
- There is one connection that uses a hardcoded IP address, which means there is no DNS request. Which IP is it?

Advanced Bro users continue here:
The background to this exercise can be found at http://forensicscontest.com/2010/04/01/ms-moneymanys-mysterious-malware. We present only a selection of the questions asked there. Use Bro to examine infected.pcap.

Exercise 5

Level advanced
The given pcap contains malware. Answer the following questions using Bro only:

- What is the name of one of the two .jar files that implemented the Java applet downloaded during the incident? Ambitious people can try to find the second .jar file name, too.
- What was the username on the infected Windows system that was used while the incident happened?

- What is the MD5 hash of the downloaded malicious executable?
- What URL did the user most likely click to start the whole incident?