Bro Quick Start Guide

version 0.9, 11-15-2004, **DRAFT**

Vern Paxson, Jim Rothfuss, Brian Tierney
Contact: vern@icir.org
http://www.bro-ids.org/

This the Quick Start Guide for Bro version 0.9.

For further information about this notice, contact:

Vern Paxson email: vern@icir.org

Table of Contents

1 Overview of Bro

1.1 What is Bro?

Bro is a Unix-based Network Intrusion Detection System (IDS). Bro monitors network traffic and detects intrusion attempts based on the traffic characteristics and content. Bro detects intrusions by comparing network traffic against rules describing events that are deemed troublesome. These rules might describe activities (e.g., certain hosts connecting to certain services), what activities are worth alerting (e.g., attempts to a given number of different hosts constitutes a "scan"), or signatures describing known attacks or access to known vulnerabilities. If Bro detects something of interest, it can be instructed to either issue a log entry or initiate the execution of an operating system command.

Bro targets high-speed (Gbit/second), high-volume intrusion detection. By judiciously leveraging packet filtering techniques, Bro is able to achieve the performance necessary to do so while running on commercially available PC hardware, and thus can serve as a cost effective means of monitoring a site's Internet connection.

1.2 Bro features and benefits

- **Network Based**

  Bro is a network-based IDS. It collects, filters, and analyzes traffic that passes through a specific network location. A single Bro monitor, strategically placed at a key network junction, can be used to monitor all incoming and outgoing traffic for the entire site. Bro does not use or require installation of client software on each individual, networked computer.

- **Custom Scripting Language**

  Bro policy scripts are programs written in the Bro language. They contain the "rules" that describe what sorts of activities are deemed troublesome. They analyze the network activity and initiate actions based on the analysis. Although the Bro language takes some time and effort to learn, once mastered, the Bro user can write or modify Bro policies to detect and alert on virtually any type of network activity.

- **Pre-written Policy Scripts**

  Bro comes with a rich set of policy scripts designed to detect the most common Internet attacks while limiting the number of false positives, i.e., alerts that confuse uninteresting activity with the important attack activity. These supplied policy scripts will run "out of the box" and do not require knowledge of the Bro language or policy script mechanics.

- **Powerful Signature Matching Facility**

  Bro policies incorporate a signature matching facility that looks for specific traffic content. For Bro, these signatures are expressed as regular expressions, rather than fixed strings. Bro adds a great deal of power to its signature-matching capability because of its rich language. This allows Bro to not only examine the network content, but to understand the context of the signature, greatly reducing the number of false positives. Bro comes

with a set of high value signatures policies, selected for their high detection and low false positive characteristics.

- **Network Traffic Analysis**

  Bro not only looks for signatures, but can also analyze network protocols, connections, transactions, data amounts, and many other network characteristics. It has powerful facilities for storing information about past activity and incorporating it into analyses of new activity.

- **Detection Followed by Action**

  Bro policy scripts can generate output files recording the activity seen on the network (including normal, non-attack activity). They can also send alarms to event logs, including the operating system syslog facility. In addition, scripts can execute programs, which can, in turn, send e-mail messages, page the on-call staff, automatically terminate existing connections, or, with appropriate additional software, insert access control blocks into a router's access control list. With Bro's ability to execute programs at the operating system level, the actions that Bro can initiate are only limited by the computer and network capabilities that support Bro.

- Snort **Compatibility Support**

  The Bro distribution includes a tool, snort2bro, which converts Snort signatures into Bro signatures. Along with translating the format of the signatures, snort2bro also incorporates a large number of enhancements to the standard set of Snort signatures to take advantage of Bro's additional contextual power and reduce false positives.

1.3 Getting more Information

- **Reference manual**

  An extensive reference manual is provided detailing the Bro Policy Language

- **FAQ**

  Several Frequently Asked Questions are outlined in the Bro FAQ. Do you have a question that's not in the FAQ, send it to us and we'll add it.

- **Tutorial**

  Over a hundred tutorial viewgraphs covering every aspect of Bro set up and use.

- **E-mail list**

  Send questions on any Bro subject to Bro@bro-ids.org The list is frequented by all of the Bro developers, including the primary author of Bro, Dr. Vern Paxson.

  You can subscribe by going to the website:
  http://bro-ids.org/mailman/listinfo/bro,
  or by placing the following command in either the subject or the body of a message addressed to Bro-request@bro-ids.org.

  ```
  subscribe [password] [digest-option] [address=<address>]
  ```

A password must be given to unsubscribe or change your options. Once subscribed to the list, you'll be reminded of your password periodically. The 'digest-option' may be either: 'nodigest' or 'digest' (no quotes!) If you wish to subscribe an address other than the address you use to send this request from, you may specify "address=<email address>" (no brackets around the email address, no quotes!)

- **Website**

  The official Bro website is located at: http://www.bro-ids.org. It contains all of the above documentation and more.

2 Requirements

2.1 Network Tap

A network tap must be installed to provide Bro with access to live network traffic. For Bro to be most effective, access to the network must be full-bandwidth (no bandwidth limitations) and full-duplex. A passive tap is recommended to ensure minimal impact on network operations.

Normally the network tap for Bro should be placed behind an external firewall and on the DMZ (the portion of the network under the control of the organization but outside of the internal firewall), as shown in the figure below. Some organizations might prefer to install the network tap before the firewall in order to detect all scans or attacks. Placing Bro before the firewall will allow the organization to better understand attacks, but will produce a much high number of alarms and alerts. Another option is to place Bro inside the internal firewall, allowing it to detect internal hosts with viruses or worms. In addition to the connection to the network tap, a separate network connection is required for management of Bro and access to log files.

For more information on taps and tap placement see the Netoptics White paper titled *Deploying Network Taps with Intrusion Detection Systems* (http://www.netoptics.com/products/pdf/Taps-and-IDSs.pdf).

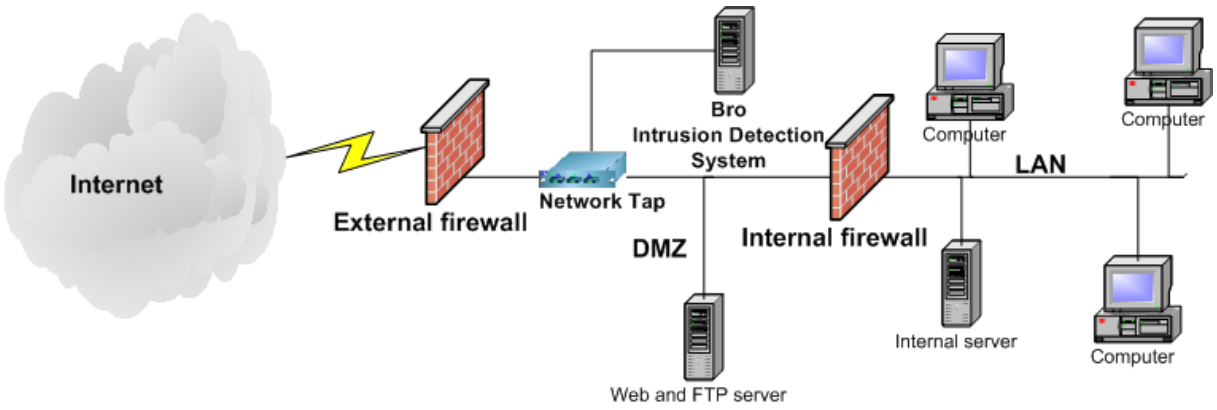

Figure 2.1: Typical location for network tap and Bro system

2.2 Hardware and Software Requirements

Bro requires no custom hardware, and runs on low-cost commodity PC-style system. However, the Bro monitoring host must examine every packet into and out of your site, so depending on your sites network traffic, you may need a fairly high-end machine. If you are trying to monitor a link with a large number of connections, we recommend using a second system for report generation, and run only Bro on the capture host.

Item	Requirements

Processor	1 GHz CPU (for 100 BT Ethernet with average packet rate <= 5,000 packets/second) 2 GHz CPU (for 1000 BT Ethernet with average packet rate <= 10,000 packets/second) 3 GHz CPU (for 1000 BT Ethernet with average packet rate <= 20,000 packets/second) 4 GHz CPU (for 1000 BT Ethernet with average packet rate <= 50,000 packets/second) (Note: these are **very** rough estimates, and much depends on the types of traffic on your network (e.g.: http, ftp, mail, etc.). See the Performance chapter of the Bro User Guide for more information)
Operating System	FreeBSD 4.10 (http://www.freebsd.org/) Bro works with Linux and Solaris as well, but the performance is best under FreeBSD. In particular there are some performance issues with packet capture under Linux. See the User Guide chapter on Bro and Linux for more information. FreeBSD 5.x should work, but may have performance issues. For sites with very high traffic loads, contact us for information on a FreeBSD 4.x patch to do *bpf bonding*
Memory	1 GB RAM is the minimum needed, but 2-3 GB is recommended
Hard disk	10 GByte minimum, 50 GByte or more for log files recommended
User privileges	*superuser* to install Bro, then Bro runs as user *bro*
Network Interfaces	3 interfaces are required: 2 for packet capture (1 for each direction), and 1 for host management. Capture interfaces should be identical.
Other Software	- Apache 1.3 http server (http://httpd.apache.org) - Perl version 5.6 or higher (http://www.perl.org) - libpcap version 0.8 or higher (http://www.tcpdump.org) - tcpdump version 3.8 or higher (http://www.tcpdump.org) Note: FreeBSD comes with older versions perl, libpcap, and tcpdump. Bro requires newer versions of these tools.

3 Installation and Configuration

3.1 Download

Download Bro from: http://www.bro-ids.org/

You can unpack the distribution anywhere except into the directory you plan to install into. To untar the file, type:

```
tar xvzf bro-0.9a6.6.tar.gz
```

3.2 Install

You'll need to collect the following information before beginning the installation.

- localnets: a list of local subnets for your network. Bro needs to know which networks are "internal" and which are "external".
- interface names: the names of the capture interfaces in your host (e.g. sk0 or en1). Use `ifconfig -a` to get the list of all network interfaces on your Bro host.

If you want to use Bro's periodic email report feature, you'll also need:

- email list: a list of email addresses to send the reports to.
- pgp keys: if you want to encrypt all email reports, the location of the GPG keyring of all recipients.

Bro is very easy to install. Just log in as `root`, and type:

```
./configure
```

or to install Bro in a location other than '`/usr/local/bro`', use:

```
./configure --prefix=/path/to/bro
```

and then type:

```
make
make install
```

To update an existing Bro installation with new binaries and standard policy file, instead of `'make install'` do a `'make update'`. This will preserve all your local customizations.

3.3 Configuration

The *Bro-Lite* configuration script can be used to automatically configure Bro for you. It checks your system's BPF settings, creates a 'bro' user account, installs a script to start bro at boot time, and installs a number of `cron` jobs to checkpoint bro every night, run perioidic reports, and manage log files.

To run this configuration script type:

```
make install-brolite
```

This will run the script `bro_config`, which creates the file '`$BROHOME/etc/bro.cfg`'. `bro_config` will ask a number of simple questions.

Sample output of `bro_config`, along with explanation, is shown below:

```
Running Bro Configuration Utility
Checking interfaces ....  Done.
Reading /usr/local/bro/etc/bro.cfg.example for defaults.
```

The `bro_config` script looks first at ./bro.cfg, then /usr/local/bro/etc, for default values to use below.

```
Bro Log archive location [/usr/local/bro/archive]
```

This is the directory where log file archives are kept. If you expect the log files to be very large, it is recommended to put these in a separate disk partition.

```
User id to install and run Bro under [bro]
```

`bro_config` will create a new user account with this username if the user does not exist.

```
Interface names to listen on. [en1,en2]
```

`bro_config` looks for all network interfaces and does a short test to determine which interfaces see the most traffic, and selects these interfaces as the default.

```
Site name for reports (i.e. LBNL, FOO.COM, BAZ.ORG) []
Starting Report Time [0600]
Report interval (in hours) [24]
Email addresses for internal reports [bro@localhost]
Do you want to send external reports to a incident
                reporting org (e.g.: CERT, CIAC, etc) (Y/N)
Y
Email addresses for external reports []
```

Daily reports will be created. Enter the site name you want to appear at the top and in the subject of all email reports. The 'start time' and 'interval' define the window of network activity that the daily report will cover, starting at 'Starting Report Time' and lasting through 'Report interval'. The start time should be entered using 24hr clock notation. For example: 12:30am = 0030, 2pm = 1400

Two types of reports will be generated, "internal" and "external". Internal reports contain the same basic information as the external reports, along with traffic statistics and more detailed information on incidents. Both internal and external reports will be sent to the "internal" email address list. External reports are only sent if you answer "Y" and enter an external email address. (Note: currently only internal reports are generated)

```
Do you want to encrypt the email reports (Y/N) [N]
Y
```

If you want the email reports encrypted, you will need to set up GPG (http://www.gnupg.org) and create a

GPG keyring containing the public keys of all email recipients. Instructions for this are in Section 3.4 [Encrypted Reports], page 8.

```
Running script to determine your local subnets ...
Your Local subnets [198.129.224.1/32]
```

Bro needs to know a list of your local subnets. `bro_config` runs a tool that attempts to discover this automatically. You should always verify the results of this tool. The format is a list of subnet/significant bits of address. For example: 131.243.0.0/16, 198.128.0.0/18, 198.129.224.1/32

This information will be stored in the file `$BROHOME/site/local.site.bro`

```
Saving settings to file: /usr/local/bro/etc/bro.cfg
Bro configuration finished.
To change these values, you can rerun bro_config at any time.
```

Indicates that the script finished successfully.

For site monitoring very high traffic rates on Gigabit ethernet, there is some additional system tuning that should be done. See the Bro User Guide for more details.

To reconfigure Bro, just type:

```
bro_config
```

This will update your '`/usr/local/bro/etc/bro.cfg`' file. You can also edit this file using your favorite editor if you prefer.

For other site customizations, you can edit the file $BROHOME/site/local.site.bro. For example, to tell bro to not look at traffic for host 198.162.44.66, add:

```
redef restrict_filters += { ["ignore host 198.162.44.66 "] = "not (host 198.162.44.66)
```

Or to disable alarms for "WeirdActivity", you can add this:

```
redef notice_action_filters += { [[WeirdActivity]] = ignore_notice, };
```

Any changes you make in $BROHOME/site will not be touched during an upgrade or reinstall of Bro. You should avoid editing files in $BROHOME/policy, as these will be overwritten.

More details are available in the Bro user guide.

3.4 Encrypted Reports

Bro can use GPG (`http://www.gnupg.org/`) to encrypt the reports that it sends. To have Bro encrypt your reports you must have said 'yes' to the bro_config question to encrypt your reports. For information on configuring GPG for Bro reports, see the Bro User Manual.

4 Running Bro

4.1 Starting Bro

Bro is automatically started at boot time via the `bro.rc` script, (located in /usr/local/bro/etc and /usr/local/etc/rc.d on FreeBSD or /usr/init.d on Linux)

To run this script by hand, type:

```
bro.rc start
```

or

```
bro.rc checkpoint
```

or

```
bro.rc stop
```

Use `checkpoint` to restart Bro, loading a new policy file.

4.2 Bro Scripts

Installing Bro automatically creates the following `cron` jobs, which are automatically run on a specified interval.

- `site-report.pl`: generates an email report of all alarms and alerts
- `mail_reports.sh`: send email reports

These scripts can also all be run by hand at any time.

Bro log files can get quick large, and it is important to make sure that the Bro disk does not fill up. Bro includes some simple scripts to help manage disk space. Most sites will want to customize these for their own requirements, and integrate them into their backup system to make sure files are not removed before they are archived.

- `check_disk.sh`: check for low disk space, and send email
- `bro_log_compress.sh`: removes/compresses old log files

These scripts can be customized by editing their settings in `$BROHOME/etc/bro.cfg`. The settings are as follows:

- `check_disk.sh`:
  - `diskspace_pct`: when disk is >= this percent full, send email
  - `diskspace_watcher`: list of email addresses to send mail to
- `bro_log_compress.sh`:
  - `Days2deletion`: remove files more than this many days old (default = 60)
  - `Days2compression`: compress files more than this many days old (default = 30)

4.3 Sending (E-mail) Bro Reports

A daily 'internal' report is created that covers three sets of information:

- Incident information
- Operational status of Bro
- General network traffic information

If the local organization is asked to report incidents to another incident analysis organization (i.e. CERT, CIAC, FedCIRC, etc.) an auxiliary 'external' report can be created that only contains the incident information. These reports are stored in $BRODIR/reports.

The two reports will be mailed to the e-mail addresses specified during Bro installation. These e-mail addresses can be changed by re-running the bro_config script or by editing $BROHOME/etc/bro.cfg directly. Each report has it's own set of e-mail addresses. If it is desired to send the auxiliary report directly to the external incident analysis organization without inspection, enter their e-mail address directly. Otherwise, have the external e-mail sent to someone who can inspect and forward it appropriately.

4.4 Reading a Bro Report

The report is divided into three parts, the summary, incidents, and scans. The summary includes a rollup of incident information, Bro operational statistics, and network information. The incidents section has details for each Bro alarm. The scans section gives details about scans that Bro detected.

4.4.1 Parts of a Report

Summary

Report Period: The beginning and ending date/times that define the window of network data used to produce the report.

Incident Count: The number of each type of incident that are detailed in the report period

System Statistics: Operating system statistics that give some idea of the 'health' of Bro's operation.

Traffic Statistics: Statistics gathered by Bro that may or may not have significant value in evaluating intrusions, but are useful in understanding the network environment.

Incidents

Incident: Each incident generated by the Bro installation is assigned a unique identification number. This number is unique for all incidents, not just to the daily report.

Incident Type: Bro can detect attacks, but cannot make a definitive judgment if an attack is successful without further investigation and/or knowledge of the

unique network environment. Bro uses an expert knowledge algorithm to make a determination if an incident is 'Likely Successful', 'Unknown' (not enough information to make a guess), or 'Likely Unsuccessful'.

Local Host: The local computer involved in the incident; usually the victim.

Remote Host: The remote computer involved in the incident; usually the attacker.

Alarm(s:) The network event(s) that Bro detected and identified as probable attacks.

Successful Connections: Connections where one host initiates a network request and the other host participates in the subsequent requested transactions.

Unsuccessful Connections: Connections where one host initiates a network request and the other host refuses the request.

Unknown Connections: Connections where one host initiated a network request, but it is unclear if the other host participated in a successful transaction.

Connections History: A summary tabulation of successful and unsuccessful connections made in specific time periods. The tabulations are accumulative. That is, the connections counted under 3 days will also be counted in each subsequent column.

Scans

Scans are repetitive (similar) probes, searching several victim hosts for vulnerabilities. The scan section gives the attack host instigating the scan, the date/time of the scan, and the ports that were probed.

4.4.2 Example Report:

```
Bro Report                                                  Organization Name
=============================================================================
Summary                        July 28, 2004 17:01 to July 29, 2004 17:00
=============================================================================
 Incident       Likely Successful          1
 Summary        Unknown                    0
                Likely Unsuccessful        0
                Scans                     10

 System         Bro disk space:   <% at time of report generation>
 Statistics     Bro Process cpu:  <time>
                Bro restarts:     <date/time>
                System reboots:   <date/time>

 Traffic        Number of packets:       <count>
```

```
 Statistics     Number of valid packets: <count>  <% of total>
                Protocol summary
                Http: <count>   <% of total>
                SSH : <count>   <% of total>
                SMTP: <count>   <% of total>
                Etc.
                Average bandwidth:
                Peak bandwidth:
=============================================================================
Incident Details
                      legend for connection type
                > connection initiated by remote host
                < connection initiated by local host
                # number corresponds to alarm triggered by the connection
                * successful connection, otherwise unsuccessful
=============================================================================
Incident     ORGCODE-000002                                LIKELY SUCCESSFUL
---------------------
Remote Host: 84.136.138.21   p54877614.dip.hacker.net
 Local Host: 124.333.183.162 pooroljoe.dhcp.org.com

Alarm(s) 1 MS-SQL xp_cmdshell - program execution
           Jul 29 12:43 84.135.118.20 -> 128.3.183.62
         2 TFTP Get Runtime.exe
           Jul 29 12:43 128.3.183.62 -> 84.135.118.20

Connections (only first 25 after alarm are listed)
-----------
                  time      byte   remote       local    byte
 date   time    duration  transfer  port  type   port  transfer  protocol
----- -------- -------- --------- -----  ---- ------ --------- ----------
07/29 12:43:31        ?     566 b  4634  1  >  1433      467 b  tcp/MSSQL
07/29 12:43:31        0         ?  2318  2 <     69       20 b  udp/tftp
07/29 12:43:32    265.7       4 b  4638  * <   2318      3.0kb  udp
07/29 12:48:56        ?         ?  4640    >  2362          ?  tcp
07/29 12:50:05        ?    11.4kb  4639  * <   3333      8.6kb  tcp
07/29 12:53:00        0         ?  4684  *  >  2362          ?  tcp
07/29 12:53:07        ?         ?  4685  *  >  2362          ?  tcp
07/29 12:53:59        ?         ?  4689  *  >  2362          ?  tcp
07/29 12:54:14      6.1         0  4693  * <   2380     94.2kb  tcp
07/29 12:54:21       .5      50 b  4694    >  2381          0  tcp
07/29 12:54:23       .7         ?  4695   <   2382          0  tcp
07/29 12:54:25       .5      51 b  4696  *  >  2383          0  tcp
07/29 12:54:27       .5      61 b  4697  *  >  2384          0  tcp
07/29 12:54:28       .7      39 b  4698    >  2385          0  tcp
07/29 12:54:31       .5      41 b  4699  *  >  2386          0  tcp
07/29 12:54:33      1.2    4.9 kb  4700    >  2387          0  tcp
```

```
07/29 12:54:35     12.8  195.0 kb  4701  * <   2388          0  tcp
07/29 12:54:53       .2         ?  4703    <   2390          0  tcp
07/29 12:54:54       .5      37 b  4704     >  2391          0  tcp
07/29 12:54:56      3.4      23 b  4705  *  >  2392          0  tcp
07/29 12:55:04     21.4  308.7 kb  4706     >  2393          0  tcp
07/29 12:55:27     50.7         ?  4707     >  2394          ?  tcp
07/29 12:59:23        ?         ?  4775     >  1433          ?  tcp
07/29 12:59:25        ?         ?  4774  *  >  3333          ?  tcp

Remote Host Connection History (all successful/unsuccessful to site)
      24 hrs     |      3 days      |      7 days     |      30 days
-------------------------------------------------------------------------
       14/10     |        0/0       |        0/0      |        0/0
-------------------------------------------------------------------------
  Total since remote host first seen on 07/29/04: 14/10

=========================================================================
Scans
=========================================================================
==
Date Dropped            Host                                  Port Scanned
-------------------------------------------------------------------------
Jul 29 13:14 n219077002119.netvigator.com                      (3128/tcp)
Jul 29 13:23 node1.lbnl.nodes.planet-lab.org                   (49702/tcp)
Jul 29 13:30 213-145-189-50.dd.nextgentel.com                  (4899/tcp)
Jul 29 13:32 211.55.52.67                                      (1034/tcp)
Jul 29 13:52 user-69-1-11-116.knology.net                      (3128/tcp)

*************************************************************************
```

Index