syslog2iptables - Version 1.16

Packages

The various source and binary packages are available at http://www.five-ten-sg.com/syslog2iptables/packages/
The most recent documentation is available at http://www.five-ten-sg.com/syslog2iptables/

A Mercurial [http://www.selenic.com/mercurial/wiki/] source code repository for this project is available at http://hg.five-ten-sg.com/syslog2iptables/.

Name

syslog2iptables — a simple adaptive firewall

Synopsis

`syslog2iptables [-c][-d n]`

Description

syslog2iptables is a simple adaptive firewall. It maintains the INPUT chain of the iptables(1) firewall set based on syslog entries. These syslog entries are typically generated by your hardware firewall, but they could come from any source. Any syslog entry that contains a host name or ip address can be used as input to this package.

The syslog2iptables.conf(5) file specifies the syslog files to be monitored, and the regular expressions (regex(7)) to be applied to new lines in those files. Each regular expression needs an INDEX to specify the matching substring that contains either an ip address or host name, and a DELTA which is used to modify the leaky bucket count for that ip address when a matching line is read from that syslog file.

If the DELTA is negative, the leaky bucket count is set to that DELTA value, any existing blocking for that ip address is removed, and new blocking is prevented until that bucket leaks upward to zero.

If the DELTA is positive and the current leaky bucket count is not negative, that DELTA value is added to the leaky bucket count for that ip address. Once the bucket contains more than a configurable THRESHOLD number of tokens, that ip address is added to the INPUT chain with a DROP target.

Each ip address has an associated leaky bucket, which leaks one token per second so the count moves toward zero. When the bucket is drained to zero, that ip address is removed from the INPUT chain.

The discussion has focused on syslog files, but any ascii text file can be used, so long as some other process appends lines to that file, and those lines containing hostname or ip addresses can be matched with some regular expression.

Considering syslog files in particular, these are normally rotated via logrotate. **syslog2iptables** properly detects and handles this case by closing the old file, and reopening the newly created file.

With the default config file, you can manually unblock an ip address with **logger -p authpriv.info "manual unblock 1.2.3.4"** and you can manually block an ip address with **logger -p authpriv.info "manual block 1.2.3.4"**

Options

-c Load the configuration file, print a cannonical form of the configuration on stdout, and exit.

-d *n* Set the debug level to *n*.

Usage

syslog2iptables -d 2

Configuration

The configuration file is documented in syslog2iptables.conf(5). Any change to the config file will cause it to be reloaded within three minutes.

TODO

The following ideas are under consideration.

Add a configuration option for the iptables table name in the pattern statement. This implies handling multiple tables, so each table needs its own map of ip addresses and bucket values.

Copyright

Version

1.16

Name

syslog2iptables.conf — configuration file for syslog2iptables

Synopsis

```
syslog2iptables.conf
```

Description

The **syslog2iptables.conf** configuration file is specified by this partial bnf description. The entire config file is case sensitive. All the keywords are lower case.

```
CONFIG    = {CONTEXT ";"}+
CONTEXT   = "context" NAME "{" {STATEMENT}+ "}"
STATEMENT := (THRESHOLD | ADD-CMD | REM-CMD | IGNORE | FILE) ";"
THRESHOLD := "threshold" THRESHOLD-INTEGER-VALUE
ADD-CMD   := "add_command" IPT-CMD
REM-CMD   := "remove_command" IPT-CMD
IGNORE    := "ignore" "{" IG-SINGLE+ "}"
IG-SINGLE := IP-ADDRESS "/" CIDR-BITS
FILE      := "file" FILENAME "{" PATTERN+ "}"
PATTERN   := "pattern" REGULAR-EXPRESSION "{" {INDEX | BUCKET | MESSAGE}+ "};"

INDEX     := "index" REGEX-INTEGER ";"
DELTA     := "bucket" BUCKET-DELTA-INTEGER ";"
MESSAGE   := "message" REASON ";"
REASON    := string to appear in syslog messages
IPT-CMD   := string containing exactly one %s replacement token for
             the ip address
```

Sample

```
context general {
   threshold 550;

   add_command   "/sbin/iptables -I INPUT --src %s --jump DROP";
   remove_command "/sbin/iptables -D INPUT --src %s --jump DROP";

   ignore {
      127.0.0.0/8;      // localhost
   };

   file "/var/log/secure" {
      pattern "manual unblock (.*)" {
         index 1;   // zero based
         bucket -5000;
         message "manual unblock";
      };
      pattern "sshd.*Failed password .* from ::ffff:(.*) port" {
```

```
            index 1;   // zero based
            bucket 400;
            message "ssh failed password";
         };
         pattern "sshd.*Failed password .* from (.*) port" {
            index 1;   // zero based
            bucket 400;
            message "ssh failed password";
         };
         pattern "sshd.*authentication failure; .* rhost=(.*) " {
            index 1;   // zero based
            bucket 400;
            message "ssh failed password";
         };
         pattern "sshd.*Did not receive identification string from (.*)" {
            index 1;   // zero based
            bucket 400;
            message "ssh failed password";
         };
         pattern "proftpd.*no such user found from (.*) \[" {
            index 1;   // zero based
            bucket 400;
            message "ftp failed password";
         };
         pattern "proftpd.* authentication failure; .* rhost=(.*) " {
            index 1;   // zero based
            bucket 400;
            message "ftp failed password";
         };
         pattern "vsftpd.* authentication failure; .* rhost=(.*) " {
            index 1;   // zero based
            bucket 400;
            message "ftp failed password";
         };
         pattern "dovecot.* authentication failure; .* rhost=::ffff:(.*) " {

            index 1;   // zero based
            bucket 100;
            message "dovecot failed password";
         };
         pattern "dovecot.* authentication failure; .* rhost=(.*) " {
            index 1;   // zero based
            bucket 100;
            message "dovecot failed password";
         };
      };

      file "/var/log/messages" {
         pattern "dovecot.* authentication failure; .* rhost=(.*) " {
            index 1;   // zero based
            bucket 100;
            message "dovecot failed password";
         };
         pattern "kernel.*local-net-to.*SRC=(.*) DST=.*DPT=" {
```

```
            index 1;   // zero based
            bucket 400;
            message "kernel firewall blocked packet";
         };
         pattern "kernel.*outside-net-from.*SRC=(.*) DST=.*DPT=" {
            index 1;   // zero based
            bucket 400;
            message "kernel firewall blocked packet";
         };
      };

      file "/var/log/maillog" {
         pattern "lost input channel from.* \[(.*)\] .* after (mail|rcpt|auth)" {

            index 1;   // zero based
            bucket 100;
            message "sendmail spammer dropping connection";
         };
         pattern " \[(.*)\].* possible SMTP attack" {
            index 1;   // zero based
            bucket 100;
            message "sendmail authentication attack";
         };
         pattern "rejecting commands from.* \[(.*)\] due to pre-greeting traffic" {

            index 1;   // zero based
            bucket 1800;
            message "sendmail pre-greeting";
         };
         pattern "authentication failure: checkpass failed, .*\[(.*)\]" {
            index 1;   // zero based
            bucket 100;
            message "sendmail authentication failed";
         };
         pattern "dovecot.*Aborted login .* rip=(.*)," {
            index 1;   // zero based
            bucket 100;
            message "dovecot failed password";
         };
         pattern "dovecot.*Login: .* rip=(.*)," {
            index 1;   // zero based
            bucket -5000;
            message "dovecot good authentication";
         };
         pattern "sendmail.*AUTH=server, .*\[(.*)\]," {
            index 1;   // zero based
            bucket -5000;
            message "sendmail good authentication";
         };
      };
   };
```

Version

1.16