syslog2iptables - Version 1.12

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

syslog2iptables

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 bucket count which is added to the leaky bucket for that ip address when a matching line is read from that syslog file.

Each ip address has an associated leaky bucket, which leaks one token per second. 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. 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.

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

1.12

Name

syslog2iptables.conf -- configuration file for syslog2iptables

syslog2iptables.conf

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

Sample

```
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/cisco.log" {
   pattern "Internet_Firewall denied (tcp|udp) ([^(]*)" {
      index 2;   // zero based
      bucket 200;
      message "cisco firewall blocked packet";
   };
};

file "/var/log/secure" {
```

```
   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";
   };
};

file "/var/log/httpd/access_log" {
   pattern "(.*) - - .* /cgi-bin" {
      index 1;   // zero based
      bucket 400;
      message "apache cgi-bin reference";
   };
   pattern "(.*) - - .*/index2.php" {
      index 1;   // zero based
      bucket 400;
      message "apache index2.php reference";
   };
   pattern "(.*) - - .*/main.php" {
      index 1;   // zero based
      bucket 400;
      message "apache main.php reference";
   };
};

file "/var/log/maillog" {
   pattern "lost input channel from .* \[(.*)\] .* after mail" {
      index 1;   // zero based
      bucket 200;
      message "sendmail spammer dropping connection";
   };
};

file "/var/log/messages" {
   pattern "sshd.pam_unix.*authentication failure.*rhost=(.*) user=" {
      index 1;   // zero based
      bucket 300;
      message "ssh failed password";
   };
   pattern "sshd.pam_unix.*authentication failure.*rhost=(.*)$" {
      index 1;   // zero based
      bucket 300;
      message "ssh failed password";
   };
};
```

Version

1.12