Fail2ban

ban hosts that cause multiple authentication errors

Fail2ban monitors log files (e.g. /var/log/auth.log, /var/log/apache/access.log) and temporarily or persistently bans failure-prone addresses by updating existing firewall rules. Fail2ban allows easy specification of different actions to be taken such as to ban an IP using iptables or hostsdeny rules, or simply to send a notification email.

By default, it comes with filter expressions for various services (sshd, apache, qmail, proftpd, sasl etc.) but configuration can be easily extended for monitoring any other text file. All filters and actions are given in the config files, thus fail2ban can be adopted to be used with a variety of files and firewalls.

Installation

First we need to install fail2ban via aptitude

ROOT
aptitude update # aptitude **install** fail2ban

Configuration

You will find the config files in the directory /etc/fail2ban/.

To avoid merges during upgrades please do not modify the file jail.conf. Instead copy the jail.conf file to jail.local and make your changes in that file.

jail.local

The following is a working jail.local which is tested on a Debian Wheezy system. You could copy and paste the content to your **/etc/fail2ban/jail.local**.

```
# Fail2Ban configuration file.
#
# This file was composed for Debian systems from the original one
#  provided now under /usr/share/doc/fail2ban/examples/jail.conf
#  for additional examples.
#
# Author: Yaroslav O. Halchenko <debian@onerussian.com>
#
# $Revision$
#

# The DEFAULT allows a global definition of the options. They can be
```

```
overridden
# in each jail afterwards.

[DEFAULT]

# "ignoreip" can be an IP address, a CIDR mask or a DNS host
ignoreip = 127.0.0.1/8
findtime = 600
bantime  = 600
maxretry = 3

# "backend" specifies the backend used to get files modification. Available
# options are "gamin", "polling" and "auto".
# yoh: For some reason Debian shipped python-gamin didn't work as expected
#      This issue left ToDo, so polling is default backend for now
backend = auto

#
# Destination email address used solely for the interpolations in
# jail.{conf,local} configuration files.
destemail = root@localhost

#
# ACTIONS
#

# Default banning action (e.g. iptables, iptables-new,
# iptables-multiport, shorewall, etc) It is used to define
# action_* variables. Can be overridden globally or per
# section within jail.local file
banaction = iptables-multiport

# email action. Since 0.8.1 upstream fail2ban uses sendmail
# MTA for the mailing. Change mta configuration parameter to mail
# if you want to revert to conventional 'mail'.
mta = sendmail

# Default protocol
protocol = tcp

# Specify chain where jumps would need to be added in iptables-* actions
chain = INPUT

#
# Action shortcuts. To be used to define action parameter

# The simplest action to take: ban only
action_ = %(banaction)s[name=%(__name__)s, port="%(port)s",
protocol="%(protocol)s", chain="%(chain)s"]

# ban & send an e-mail with whois report to the destemail.
```

```
action_mw = %(banaction)s[name=%(__name__)s, port="%(port)s",
protocol="%(protocol)s", chain="%(chain)s"]
              %(mta)s-whois[name=%(__name__)s, dest="%(destemail)s",
protocol="%(protocol)s", chain="%(chain)s"]

# ban & send an e-mail with whois report and relevant log lines
# to the destemail.
action_mwl = %(banaction)s[name=%(__name__)s, port="%(port)s",
protocol="%(protocol)s", chain="%(chain)s"]
               %(mta)s-whois-lines[name=%(__name__)s, dest="%(destemail)s",
logpath=%(logpath)s, chain="%(chain)s"]

# Choose default action.  To change, just override value of 'action' with
the
# interpolation to the chosen action shortcut (e.g.  action_mw, action_mwl,
etc) in jail.local
# globally (section [DEFAULT]) or per specific section
action = %(action_)s

#
# JAILS
#

[ssh]

enabled  = true
port     = ssh
filter   = sshd
logpath  = /var/log/auth.log
maxretry = 6

[ssh-ddos]

enabled  = true
port     = ssh
filter   = sshd-ddos
logpath  = /var/log/auth.log
maxretry = 6

#
# HTTP servers i-MSCP customer sites
#

[apache]

enabled  = true
port     = http,https
filter   = apache-auth
logpath  = /var/log/apache2/*/error.log
maxretry = 6
```

```
# default action is now multiport, so apache-multiport jail was left
# for compatibility with previous (<0.7.6-2) releases
[apache-multiport]

enabled  = true
port     = http,https
filter   = apache-auth
logpath  = /var/log/apache2/*/error.log
maxretry = 6

[apache-noscript]

enabled  = true
port     = http,https
filter   = apache-noscript
logpath  = /var/log/apache2/*/error.log
maxretry = 6

[apache-overflows]

enabled  = true
port     = http,https
filter   = apache-overflows
logpath  = /var/log/apache2/*/error.log
maxretry = 2



#
# HTTP servers i-MSCP Control Panel
#

[imscp]

enabled  = true
port     = 8080,4443
filter   = nginx-http-auth
logpath  = /var/log/nginx/*error.log
maxretry = 6



#
# FTP servers
#

[proftpd]

enabled  = true
port     = ftp,ftp-data,ftps,ftps-data
filter   = proftpd
logpath  = /var/log/auth.log
maxretry = 6
```

```
#
# Mail servers
#
#
# Mail servers authenticators: might be used for smtp,pop3,imap servers, so
# all relevant ports get banned
#

[dovecot]

enabled  = true
port     = smtp,ssmtp,submission,imap2,imap3,imaps,pop3,pop3s
filter   = dovecot
logpath  = /var/log/mail.log
maxretry = 8


#
# Webmail
#
#
# Webmail authenticators: Don't forget to comment the unused webmailers out
#

[roundcube]
enabled  = true
port     = 8080,4443
filter   = roundcube
logpath  = /var/www/imscp/gui/public/tools/webmail/logs/errors
maxretry = 6

[rainloop]
enabled  = true
port     = 8080,4443
filter   = rainloop
logpath  = /var/log/nginx/*access.log
maxretry = 6
```

nginx-http-auth.conf

Please check if the file **/etc/fail2ban/filter.d/nginx-http-auth.conf** is available. If not, please create the file with the following content:

```
# fail2ban filter configuration for nginx


[Definition]


failregex = ^ \[error\] \d+#\d+: \*\d+ user "\S+":? (password mismatch|was
```

```
not found in ".*"), client: <HOST>, server: \S+, request: "\S+ \S+
HTTP/\d+\.\d+", host: "\S+"\s*$

ignoreregex =

# DEV NOTES:
# Based on samples in https://github.com/fail2ban/fail2ban/pull/43/files
# Extensive search of all nginx auth failures not done yet.
#
# Author: Daniel Black
```

roundcube.conf

Now create a new file **/etc/fail2ban/filter.d/roundcube.conf** and copy the following content into the file:

```
# roundcube configuration file
#


[Definition]

# Option:  failregex
# Notes.:  regex to match the password failures messages in the logfile. The
#          host must be matched by a group named "host". The tag "<HOST>"
can
#          be used for standard IP/hostname matching and is only an alias
for
#          (?:::f{4,6}:)?(?P<host>\S+)
# Values:  TEXT
#
failregex = .*Error: Login failed for .* from <HOST>\..*

# Option:  ignoreregex
# Notes.:  regex to ignore. If this regex matches, the line is ignored.
# Values:  TEXT
#
ignoreregex =
```

---

rainloop.conf

Now create a new file **/etc/fail2ban/filter.d/rainloop.conf** and copy the following content into the file:

```
# rainloop configuration file
#
```

```
[Definition]

# Option:  failregex
# Notes.:  regex to match the password failures messages in the logfile. The
#          host must be matched by a group named "host". The tag "<HOST>"
can
#          be used for standard IP/hostname matching and is only an alias
for
#          (?:::f{4,6}:)?(?P<host>\S+)
# Values:  TEXT
#
failregex = ^<HOST> -.*POST /rainloop/index.php\?/Ajax/0/ HTTP/1.1" 200

# Option:  ignoreregex
# Notes.:  regex to ignore. If this regex matches, the line is ignored.
# Values:  TEXT
#
ignoreregex =
```

Now restart fail2ban and test if all is working:

ROOT
service fail2ban restart

Test & Debug

To test your current config use fail2ban-regex. Here an example for dovecot:

ROOT
fail2ban-regex /var/log/mail.log /etc/fail2ban/filter.d/dovecot.conf

Links

Fail2ban offical website –> http://www.fail2ban.org

From:
https://wiki.i-mscp.net/ - **i-MSCP Documentation**

Permanent link:
https://wiki.i-mscp.net/doku.php?id=start:howto:fail2ban&rev=1443454833

Last update: **2015/09/28 16:40**